remote sensing

MDPI

Article

A Detection Approach for Floating Debris Using Ground Images Based on Deep Learning

Guangchao Qiao [1,2], Mingxiang Yang [2,3,*] and Hao Wang [2,3]

[1] College of New Energy and Environment, Jilin University, Changchun 130021, China
[2] China Institute of Water Resources and Hydropower Research, Beijing 100038, China
[3] State Key Laboratory of Simulation and Regulation of Water Cycle in River Basin, Beijing 100038, China
* Correspondence: yangmx@iwhr.com

Abstract: Floating debris has a negative impact on the quality of the water as well as the aesthetics of surface waters. Traditional image processing techniques struggle to adapt to the complexity of water due to factors such as complex lighting conditions, significant scale disparities between far and near objects, and the abundance of small-scale floating debris in real existence. This makes the detection of floating debris extremely difficult. This study proposed a brand-new, effective floating debris detection approach based on YOLOv5. Specifically, the coordinate attention module is added into the YOLOv5 backbone network to help the model detect and recognize objects of interest more precisely so that feature information of small-sized and dense floating debris may be efficiently extracted. The previous feature pyramid network, on the other hand, summarizes the input features without taking into account their individual importance when fusing features. To address this issue, the YOLOv5 feature pyramidal network is changed to a bidirectional feature pyramid network with effective bidirectional cross-scale connection and weighted feature fusion, which enhances the model's performance in terms of feature extraction. The method has been evaluated using a dataset of floating debris that we built ourselves (SWFD). Experiments show that the proposed method detects floating objects more precisely than earlier methods.

Keywords: floating debris detection; riverine litter; deep learning; object detection; YOLOv5

Citation: Qiao, G.; Yang, M.; Wang, H. A Detection Approach for Floating Debris Using Ground Images Based on Deep Learning. *Remote Sens.* **2022**, *14*, 4161. https://doi.org/10.3390/rs14174161

Academic Editor: Alberto Refice

Received: 25 July 2022
Accepted: 19 August 2022
Published: 24 August 2022

1. Introduction

The existence of rivers serves as the foundation for human survival and development. They offer water for domestic needs, agriculture, and industry for people. Additionally, they perform functions associated with the growth of modern civilization, such as shipping, flood control, tourism, and climate regulation.

However, with the development of human society and the acceleration of urbanization, the occurrence of apparent pollution incidents in rivers frequently occur. Among them, floating debris has attracted more and more attention, including domestic garbage discarded by humans and dead animals and plants. The Parisian public sanitation service (SIAAP) has been using a network of floating debris-retention booms since 1990 to prevent any visible pollution in the Seine River. According to six years of monitoring (2008–2013), the average total mass of retrieved floating rubbish is 1937 tons, which has seriously harmed the river's aesthetics and aquatic ecosystem [1]. A lot of land garbage is washed into the Nakdong River in the rainy season, discharging 3000 tons of debris every year, which greatly impacts the water ecology [2]. A number of sources of floating debris are connected to highly populated areas, and some studies have revealed that the quantity of debris downstream of bigger metropolitan centers has grown [3,4]. Perishable and smelly materials of various sizes are mixed up with the floating debris, along with plastics and foams that are difficult to degrade. In June 2020, Fujian Province suffered continuous severe rains, which swept a considerable amount of debris and withered branches and leaves from the

surrounding environment into Nanpu Stream, having a significant influence on the river ecosystem. According to Zheng in Tianjin's investigation on the state of floating objects in the Haihe River, if rubbish is not cleaned on time, organic pollution in the Haihe River would worsen [5]. For example, there is leftover food in the fast-food box, residual protein, sugar, and other chemicals in the trash beverage bottle, and water-soluble contents in waste containers and packaging. If these floating objects are not salvaged, they will contribute to organic contamination in bodies of water. Currently, cleaning surface water is mostly performed manually. However, because of the huge water area and complicated geographic distribution of the water, manual operation is not only cumbersome and ineffective, but it also makes it impossible to see tiny floating objects. The task of cleaning up floating debris will gradually be performed by autonomous ships with a greater level of intelligence and automation instead of manual operations as the urban smart water conservation process advances. As a result, reliable detection of floating debris is essential for unmanned ships to clear up floating objects effectively.

Video surveillance networks are evolving rapidly in the direction of "high-definition, networking, and intelligence" due to the fast development of 5G communication technology and artificial intelligence, and machine vision-based intelligent detection systems are extensively employed in numerous domains, including face recognition [6], autonomous driving [7], and pedestrian detection [8]. If the approach of machine vision is used to recognize the floating debris in the surface water, the objects of conventional size can be effectively located and recognized, which greatly improves the work efficiency. However, the floating debris in the water area often has the phenomenon that the objects' size is too small, the features are easy to lose, and the objects cannot be located and recognized. The reason is that small objects have fewer pixels, fewer available features, and high positioning accuracy. All existing object detectors based on deep learning inevitably face the scaling problem, and the differences between different images, or even between the objects to be examined in the same image, are extremely vast, and there are a lot of tiny objects used in practical applications. Therefore, small object detection has numerous potential applications and is crucial in a variety of fields, including defect detection [9], aerial image analysis [10], and smart medical care [11].

Early methods for surface water debris detection contained background fractionation, frame difference, and image segmentation [12]. In 2019, Lin [13] realized the detection of river floating objects based on UAV by combining time difference detection and background subtraction, but it is easy to cause detection errors due to the change in water surface characteristics. In terms of convolutional neural network-based deep learning methods, Zhang [14] used the improved RefineDet to detect floating objects on the water surface in real-time, and the detection accuracy in the dataset was above 80%. Lin [15] presented the improved YOLOv5s (FMA-YOLOv5s), which obtained 79.41% mAP and 42FPS on the test dataset by adding a feature map attention (FMA) layer at the end of the backbone. These detectors show good performance in detecting large/conventional scale floating debris, but poor performance when the debris to be detected is small. The ability to convey abstraction gradually grows as the convolution neural network is deepened, but shallow spatial information is lost, resulting in the deep feature map being unable to offer fine-grained spatial information, making it unable to precisely find objects. At the same time, the semantic information of small objects is gradually lost in the process of downsampling.

A considerable amount of research has indicated that the improved YOLOv5 has a wide range of applications. Zhu [16] introduced CBAM and ECA-Net to YOLOv5 to improve the detection of boulders in planetary images. Jin [17] proposed a YOLOv5 with an attention mechanism to detect ships. Shi [18] detects flying birds at airports by introducing a channel attention mechanism in YOLOv5. From the current point of view, most of these improved methods have certain defects. For example, YOLOv5 tries to introduce location information on the channel by global pooling, but this approach can only capture local information and cannot obtain long-range dependent information. In response to the above issues, we improved the existing great YOLOv5. Coordinate attention

encodes the channel information of the feature map along the horizontal and vertical spatial directions by introducing the recent and innovative coordinate attention method into the backbone network. Long-term spatial direction dependencies are acquired, precise location information is preserved, and the network's global receptive field is enlarged. Furthermore, because the previous feature pyramid network ignores the importance of different input characteristics when fusing information, it just summaries them indiscriminately. To alleviate this problem, the bidirectional feature pyramid network with efficient bidirectional cross-scale connection and a weighted feature fusion is used to replace the feature pyramid module in YOLOv5, enhancing the model's feature extraction ability. This improves YOLOv5 in two ways to improve its detection of tiny and dense objects.

We delve into great depth about our methodology in the sections that follow. We delve into greater depth about the proposed approach and the information utilized in the article in Section 2. Detailed experimental results and discussion are given in Section 3. The conclusion of this study may be found in Section 4.

2. Materials and Methods

2.1. Dataset

The study is based on the surface water floating debris image dataset (SWFD dataset) established by our team (floating debris includes plastic bottles, plastic bags, foam, branches, and floating algae). This dataset was gathered by Internet download and on shooting. Crawling images of floating debris in surface water from the Internet and downloading them to a local disk using crawler technology. By manually examining the images, users may then choose the images with the best quality; the on-site shooting method was filmed by our team to the first section of the North Canal in Beijing, China (this refers to the river section from Beiguan Barrier Gate to Tongji Road and Bridge, which is important for flood control, drainage, and the landscape river in the sub-center of the city. The task of water monitoring is heavy, and daily manual operations are used to salvage floating debris on the river surface, which is heavy and inefficient) by high-definition cameras. There are on-site simulations of various scenarios in actual business, and RGB color high-definition images are obtained. The collected samples should originate from as many different scenarios as possible in order to make the collected image samples broader and assist the model to completely learning to utilize these data. For example, varied viewing angles, different lighting, and different weather might present problems with floating debris. To improve the representation features the model extracts for these issues, the camera angle is changed several times, and images are acquired at various times. The resolutions of these images vary from 416 $\times$ 416 to 1920 $\times$ 1080, with a total of 16,000 images and a total of 67,344 of all objects.

It is important to note that in real business scenarios, the camera shooting distance is long, and most of the floating objects in the video images are small in size, which brings great challenges to the detection of floating debris. In the SWFD dataset, according to the pixel ratio of the object mask relative to the image, it is found that more than 95% of the objects have a pixel ratio of less than 10%. Figure 1 shows the analysis visualization result of the SWFD dataset, Figure 1a shows the position distribution of the object center point, the horizontal and vertical coordinates represent the position of the center point, Figure 1b shows the object size distribution, and the horizontal and vertical coordinates represent the width and height of the object.

2.2. Method

2.2.1. YOLOv5

On the PASCAL VOC dataset, the R-CNN algorithm [19] defeated the peak of the traditional object detection algorithm DPM [20] by an absolute advantage in 2014. It has reached a new milestone in the field of object detection using deep learning. Deep learning algorithms have established an absolutely dominant position in the field of object detection since then, and this has persisted to this day. Deep learning-based object detection

algorithms are often classified into two types: one-stage object detectors and two-stage object detectors. The R-CNN series, which includes Fast R-CNN [21], Faster R-CNN [22], R-FCN [23], and Libra R-CNN [24], is the most typical two-stage object detector. The most prominent models for one-stage object detectors are YOLO [25–28], SSD [29], and RetinaNet [30]. Although the two-stage approach starting with Faster R-CNN has completed end-to-end training, it still falls short of the real-time needs of realistic application scenarios, whereas YOLO accomplishes true real-time object detection. YOLO returns all detection results based on the complete input image at once. Ultralytics LLC proposed YOLOv5 in 2020, after several years of updated versions.

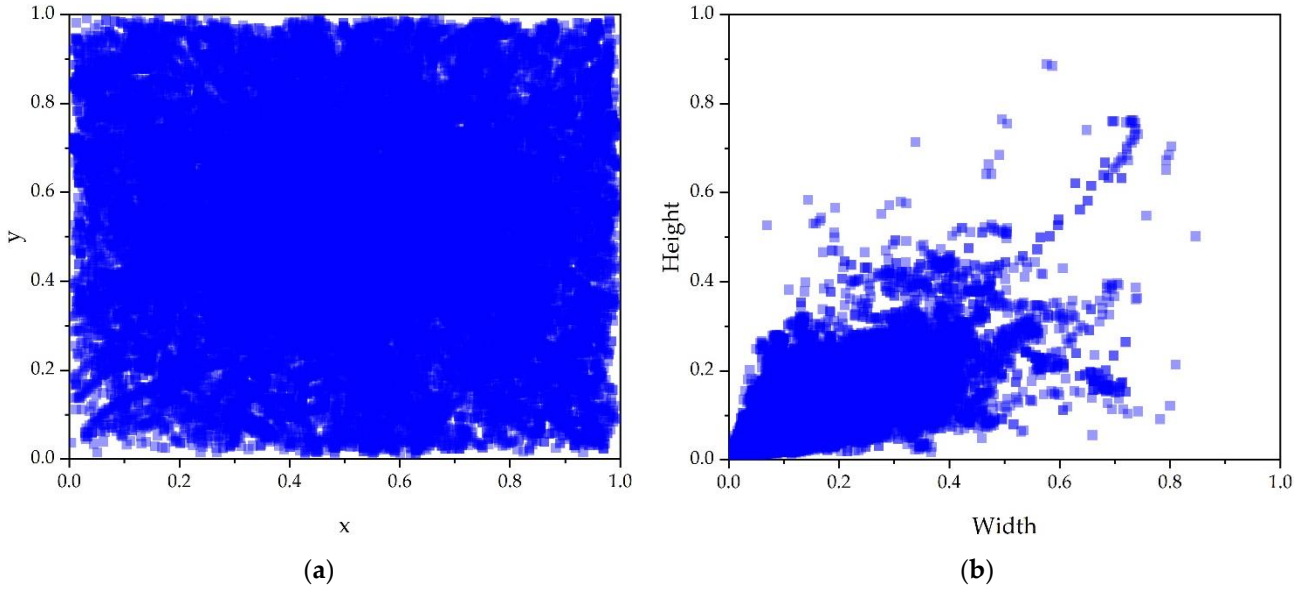


Figure 1. SWFD dataset visualization. (**a**) Position distribution of object center points, (**b**) object size distribution.

On the foundation of YOLOv4, YOLOv5 incorporates several notable academic accomplishments in recent years. It provides improved detection accuracy and speed, as well as more network deployment flexibility, making it a suitable alternative for real-time and mobile deployment scenarios. YOLOv5 is divided into four categories based on the size of the model: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. These four variations of the model increase in weight, width, and depth in succession, and they are all made up of input, backbone, neck, and head. The component of input, for example, preprocesses the image using mosaic data augmentation, auto-learning bounding box anchors, and image scaling. To extract image feature information, the backbone component employs focus downsampling, enhanced BottleneckCSP, and SPP (Spatial Pyramid Pooling) structures. The neck component combines the feature pyramid structure (Feature Pyramid Networks, FPN) with the path aggregation network structure (Path Aggregation Network, PAN) [31], allowing feature information from objects of various sizes to be transferred and solving the challenge of multi-scale object detection. The head component calculates classification, localization, and confidence loss using three loss functions, and enhances the network's accuracy using NMS (Non-Maximum Suppression).

Despite the good engineering practicability of YOLOv5, there are still challenges with the detection of tiny and dense objects. Consequently, YOLOv5 is improved by this research. By using the coordinate attention module and the bidirectional feature pyramid network, it is possible to more efficiently extract the feature data of tiny and dense objects, as well as to improve the multi-scale fusion of the prediction feature layer, which enhances the model's detection performance. Figure 2 demonstrates the improved YOLOv5′s structure; the improved modules are marked with red boxes and red lines in the figure.

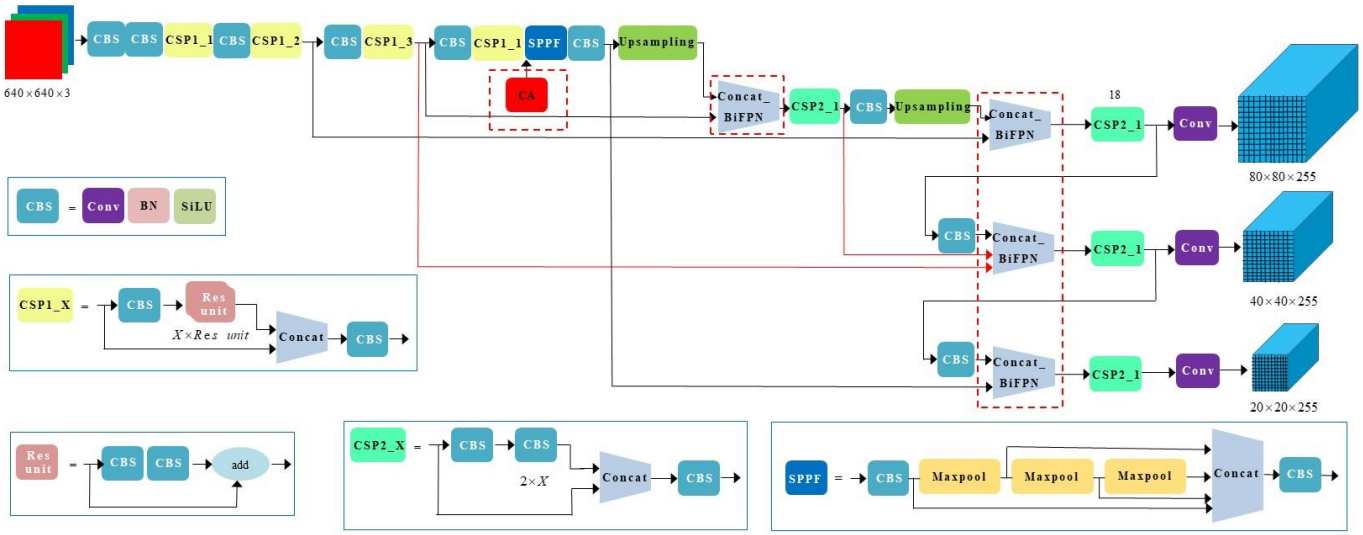


Figure 2. The structure of improved YOLOv5.

2.2.2. Improved Backbone Network

Due to the long shooting distance and broad range of video surveillance cameras, the target pixels in the picture account for a tiny fraction, and the available features are fewer. This study introduces the coordinate attention (CA) [32] mechanism, which can effectively extract the feature information of small objects and dense objects, and further improve the detection accuracy. CA embeds the location information into the channel attention so that the network can obtain the information of a larger area without introducing significant overhead. Previous channel attention mechanisms, such as SE-Net (Squeeze-and-Excitation) [33], only took into account internal channel information and ignored positional information. BAM [34] and CBAM [35] attempted to introduce location information by global pooling on the channel, but this approach could only gather local information and not long-range dependent information. The CA module encodes the channel information of the feature map along with the horizontal and vertical spatial directions, which can not only obtain the long-term dependencies of the spatial direction but also save the precise location information, and at the same time expand the global receptive field of the network.

As demonstrated in Figure 3, CA encodes channel relationships and long-term dependencies via accurate location information, which involves two steps: coordinate information embedding and coordinate attention generation. A coordinate attention module can be regarded as a computing unit that enhances the ability to express features. It can take any intermediate feature vector $X = [x_1, x_2, \ldots, x_C] \in R^{C \times H \times W}$ (C is the number of channels, and H and W are the height and width of the image, respectively) as input and output a tensor $Y = [y_1, y_2, \ldots, y_C]$ of the same size with enhanced representation ability.

To facilitate the attention module to spatially acquire long-range dependencies with precise location information, global pooling is divided into one-to-one dimensional feature encoding operations. For an input feature map X of $C \times H \times W$, each channel is encoded along the horizontal and vertical coordinates using pooling kernels $(H, 1)$ and $(1, W)$ with two spatial ranges, respectively, the output of the Cth channel of height H and the output of the Cth channel of width W are:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le I < W} x_c(h, i) \tag{1}$$

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le I < H} x_c(j, w) \tag{2}$$

Equations (1) and (2) aggregate features in two spatial directions, respectively, to generate a pair of direction-aware feature maps, which facilitates the network to more accurately locate the target to be inspected.

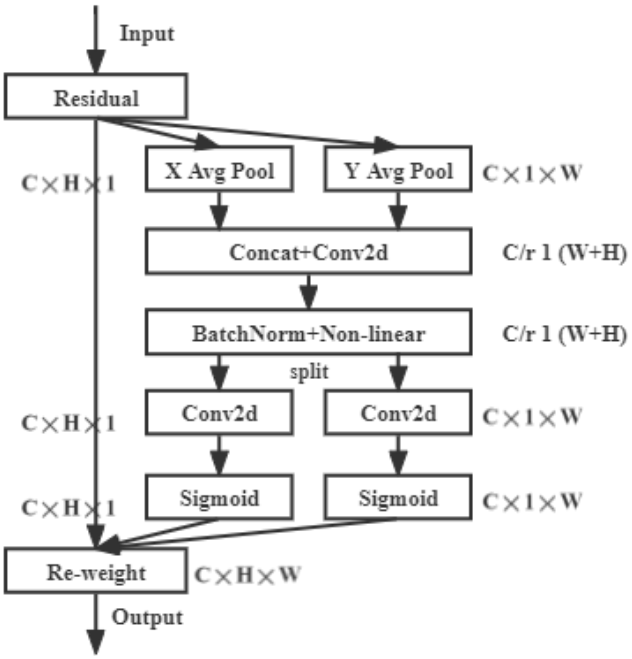


Figure 3. The structure of CA module.

As for coordinate attention generation, first there is splicing of the above two feature maps with specific direction information, and then using the 1×1 convolution transformation function F_1 with weight sharing to obtain f, as shown in Equation (3).

$$f = \delta(F_1([z^h, z^w])) \tag{3}$$

where δ is the nonlinear activation function Sigmoid, $f \in R^{C/r \times (H+W)}$ is the intermediate feature map of spatial information in the horizontal and vertical directions, and r is the scale of downsampling. Next, split f into two tensors $f^h \in R^{C/r \times H}$ and $f^w \in R^{C/r \times W}$ along the spatial dimension, and then convert f^h and f^w into tensors with the same number of channels as input X through two 1×1 convolution transformation functions F_h and F_w, respectively. As shown in Equations (4) and (5):

$$g^h = \sigma(F_h(f^h)) \tag{4}$$

$$g^w = \sigma(F_w(f^w)) \tag{5}$$

Expand g^h and g^w separately and use them as attention weights. The output of the attention module is shown in Equation (6):

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times y_c^w(j) \tag{6}$$

The CA module is a novel and efficient attention mechanism that can improve the accuracy of the network without incurring any additional overhead. The original author of CA applied it to EfficientNet and MobileNet, and they have both seen positive outcomes. To this purpose, this study adds it to the backbone of YOLOv5. The comparison of the detection results is presented in Figure 4 below. Figure 4a demonstrates the YOLOv5's detection findings. Three false negative objects can be found in the lower left and center of the image, and Figure 4b displays the outcomes after integrating the coordinate attention mechanism. The floating debris was precisely recognized in the middle and the bottom left

corner of the image. It can be observed that the attention mechanism can cause the network model to pay attention to tiny objects across a wider range, improving the network's detection ability.

Figure 4. Comparison of detection effects. (**a**) The detection effect of YOLOv5, (**b**) the detection effect of YOLOv5 introducing the CA module.

2.2.3. Multi-Scale Feature Fusion with BiFPN

Bidirectional feature pyramid network (BiFPN) [36] is a new feature fusion method proposed by the Google Brain team. It adopts efficient bidirectional cross-scale connection and weighted feature fusion. It is developed from feature pyramid networks (FPN). Figure 5a shows the structure of FPN. This structure establishes a top-down path for feature fusion and uses the fused feature layer with more semantic and geometric information for prediction. Although FPN transfers deep semantic information to shallow layers, the location information of deep features is still relatively poor. Therefore, path aggregation networks (PANet) establish a bottom-up route on the basis of FPN, and communicates the underlying location information to the prediction feature layer, so that the prediction feature layer has both the semantic information of the top layer and the position information of the bottom layer. This can greatly improve object detection accuracy.

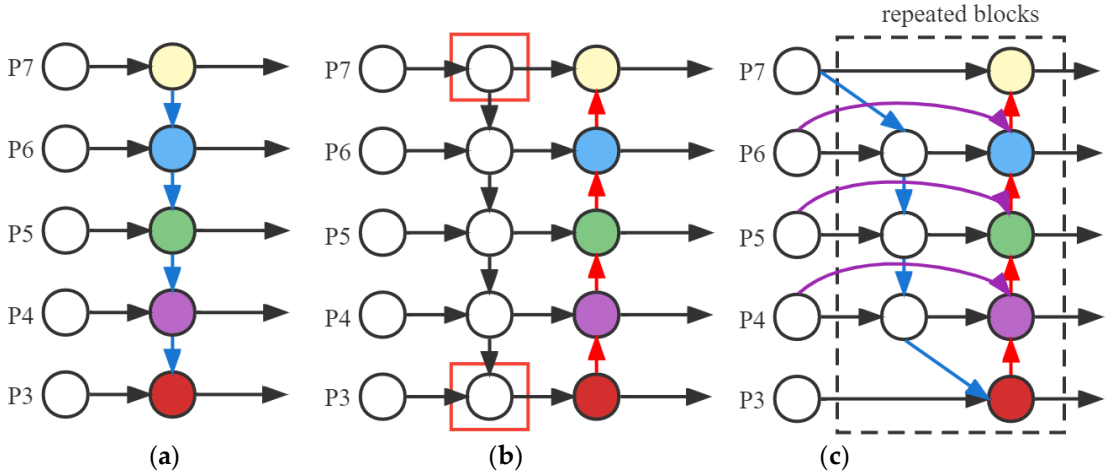


Figure 5. FPN, PANet, and BiFPN structure. (**a**) The structure of FPN, (**b**) The structure of PANet, (**c**) The structure of BiFPN.

When fusing diverse input features, most prior approaches essentially summarize them arbitrarily. However, as these diverse input features have varied resolutions, they produce uneven contributions to the fused output features. BiFPN uses learnable weights to learn the importance of various input features while repeatedly performing top-down and bottom-up multi-scale feature fusion. Its structure is shown in Figure 5c, and the

first is to remove the unit in the red box in PANet and connect it directly to the next layer. Since there are no fused objects, it is not necessary to add additional convolutions to this one-sided connection; otherwise, it will introduce features at different levels and increase the difficulty of gradient backhaul.

Then, in order to fuse more features without incurring additional costs, an edge is created between the original input node and the output node. Finally, the top-down and bottom-up routes are fused into one module so that stacking may be repeated for higher-level feature fusion. When fusing features with differing resolutions, the fast normalized fusion approach is utilized for weighted fusion. This method normalizes each weight to between 0 and 1, which improves the computation speed, as shown in Equation (7).

$$O = \sum_i \frac{w_i}{\varepsilon + \sum_j w_j} \cdot I_i \tag{7}$$

where $w_i \geq 0$, I_i represents the input feature.

Based on the above advantages, this paper replaces the PAN module in YOLOv5s with the BiFPN module to strengthen feature fusion and improve the detection speed.

2.2.4. Improved YOLOv5

Although YOLOv5 is an excellent detector at present and has good engineering practicability, and there is still room for improvement. Through the theoretical analysis and research in the previous chapters, the structure of YOLOv5 after introducing the coordinate attention mechanism and BiFPN is shown in Table 1.

Table 1. Improved YOLOv5 structure.

ID	From	N	Params	Module	Arguments
0	−1	1	3520	Conv	[3, 32, 6, 2, 2]
1	−1	1	18,560	Conv	[32, 64, 3, 2]
2	−1	1	18,816	C3	[64, 4, 1]
4	−1	1	73,984	Conv	[64, 128, 3, 2]
5	−1	2	115,712	C3	[128, 128, 2]
7	−1	1	295,424	Conv	[128, 256, 3, 2]
8	−1	3	625,152	C3	[256, 256, 3]
10	−1	1	1,180,672	Conv	[256, 512, 3, 2]
11	−1	1	1,182,720	C3	[512, 512, 1]
12	−1	1	25,648	CA	[512, 512, 32]
13	−1	1	656,896	SPPF	[512, 512, 5]
14	−1	1	131,584	Conv	[512, 256, 1, 1]
15	−1	1	0	Upsample	[None, 2, ‘nearest’]
16	[−1, 6]	1	65,794	Concat_BiFPN	[256, 256]
17	−1	1	296,448	C3	[256, 256, 1, False]
18	−1	1	33,024	Conv	[256, 128, 1, 1]
19	−1	1	0	Upsample	[None, 2, ‘nearest’]
20	[−1, 4]	1	16,514	Concat_BiFPN	[128, 128]
21	−1	1	74,496	C3	[128, 128, 1, False]
22	−1	1	295,424	Conv	[128, 128, 3, 2]
23	[−1, 14, 6]	1	65,795	Concat_BiFPN	[256, 256]
24	−1	1	296,448	C3	[256, 256, 1, False]
25	−1	1	590,336	Conv	[256, 256, 3, 2]
26	[−1, 11]	1	65,794	Concat_BiFPN	[256, 256]
27	−1	1	1,051,648	C3	[256, 512, 1, False]

The “From” represents the number of layers the input comes from, the “−1” in the column represents the output from the previous layer, the “N” represents the number of times this module is stacked, the “Params” column represents the size of the parameter, the “Module” represents the name of the module used, and the “Arguments” column is represented as module parameter information.

3. Results and Discussion

We detail the experimental setup, assessment criteria, and parameter settings in this section, which includes extensive experiments on the proposed approach. The accurate prediction outcomes of these models for recognizing partial images, as well as quantitative results, are revealed when we compare our proposed method with current state-of-the-art models. At the same time, we analyze and discuss the results.

3.1. Comparative Methods and Metrics

We compare the proposed network's floating object detection performance to that of several other state-of-the-art approaches, including Faster R-CNN, SSD, YOLOv3, and YOLOv5s. The Faster R-CNN is a typical two-stage object detector, while the others are single-stage object detectors. To ensure fairness, when training YOLOv5 and improving YOLOv5, the parameter settings for both are same. Both use the SGD optimizer to train the network on the SWFD dataset; the momentum is 0.937, the weight decay is 0.0005, the batch size is 16, and the learning rate is 0.01. It should be mentioned that all models employ the same training and test sets, as well as 300 iterations. Experiments with floating debris detection were carried out on a PC with an Intel Xeon Gold 5218 @ 2.30 GHz, Tesla T4, and 64 GB of RAM. Our model is built on the Pytorch deep learning framework, using Python as the programming language, and CUDA11.4 and CUDNN8.2.2 as the GPU accelerators. The loss comparison curve of YOLOv5 and the improved YOLOv5 is obtained from the log files saved during the training process, as shown in Figure 6.

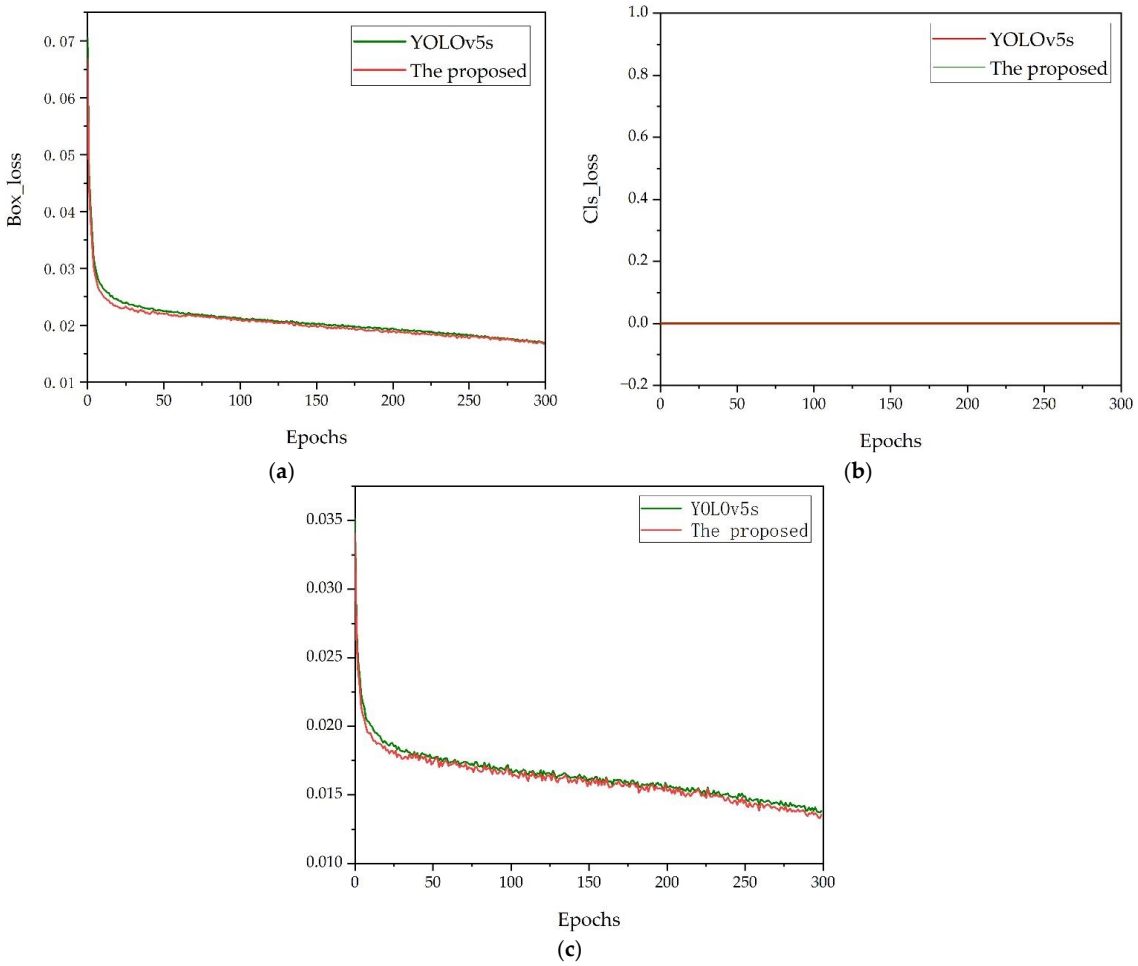


Figure 6. Training loss comparison. (**a**) Bounding box location loss, (**b**) classification loss, (**c**) confidence loss.

To assess the detector's performance, we use four widely accepted metrics: recall, AP (Average Precision), AP_{50}, and AP_{75}. The recall is defined as follows:

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

where TP and FN refer to the number of true positives and false negative pixels of all the images in the test set, respectively.

AP refers to the area enclosed by the P–R curve (which is drawn on the horizontal axis of recall and the vertical axis of precision), as defined in the following Equation (9).

$$AP = \int_0^1 p(r)dr \tag{9}$$

The difference between AP, AP_{50}, and AP_{75} is the difference in intersection over union (IoU), which is IoU = 0.50:0.05:0.95, IoU = 0.50, and IoU = 0.75, respectively.

3.2. Results on SWFD Dataset

On the SWFD test dataset, Figure 7 demonstrates the detection results of our method and other models. We only detected floating debris on the surface of the water in the experiment, and we did not distinguish between different categories of debris. As shown in Figure 7, all strategies can provide better results for large and clear debris (shown in the first column). However, our method generates results that are closer to the ground truth than the other four methods for small objects (as shown in the second column), extremely small objects (as shown in the third and fourth columns), and dense objects (as shown in the fifth column). YOLOv5s (see Figure 7f) is currently an excellent detection algorithm with good engineering practicability, but it still has significant shortcomings in the recognition of small and even extremely minute objects. By combining the feature information of multiple layers, YOLOv3 provides three feature maps of different scales for reasoning, however it still delivers unsatisfactory results for small object detection, with many false positive and false negative results. Furthermore, Faster R-CNN, and SSD produced disappointing results, including some false positive and false negative results. Our approach is particularly exciting since it discovers objects that are not in the ground truth. With a higher overlap rate between our detections and the ground truth, these results demonstrate the strength of our method. Moreover, the SWFD dataset has a large number of small objects, enabling our method to detect small and extremely small objects.

We compare quantitatively using four indicators, and Table 2 shows the complete quantitative results of experiments using the SWFD dataset. Our method outperforms all other methods on all evaluation metrics, as shown in Table 2. In comparison to SSD, Faster R-CNN has achieved comparable performance in numerous evaluation indicators, including a 12.2% improvement in recall from 77.9% to 90.1%, and a 10% rise in AP, AP_{50}, and AP_{75}. The file size of the weight file is bigger. Under the same settings, YOLOv5s exceeds all one-stage detectors. Our proposed method, on the other hand, has the best results, with a recall of 96.5% and AP, AP_{50}, and AP_{75} of 95.8%, 97.9%, and 97.1%, respectively. The recall of our method is 2.6% higher than that of YOLOv5s, and the AP is 3.4% higher. Our method is also closer to the ground truth because the AP_{50} and AP_{75} are larger. These results indicate that the coordinate attention mechanism and BiFPN play a significant role in helping our model to make more accurate predictions.

3.3. Ablation Studies

We include a coordinate attention mechanism and BiFPN into the network to improve the prediction performance of small and dense objects. In this part, we will perform ablation experiments to see how these two approaches impact the network. The experiment's results are shown in Table 3.

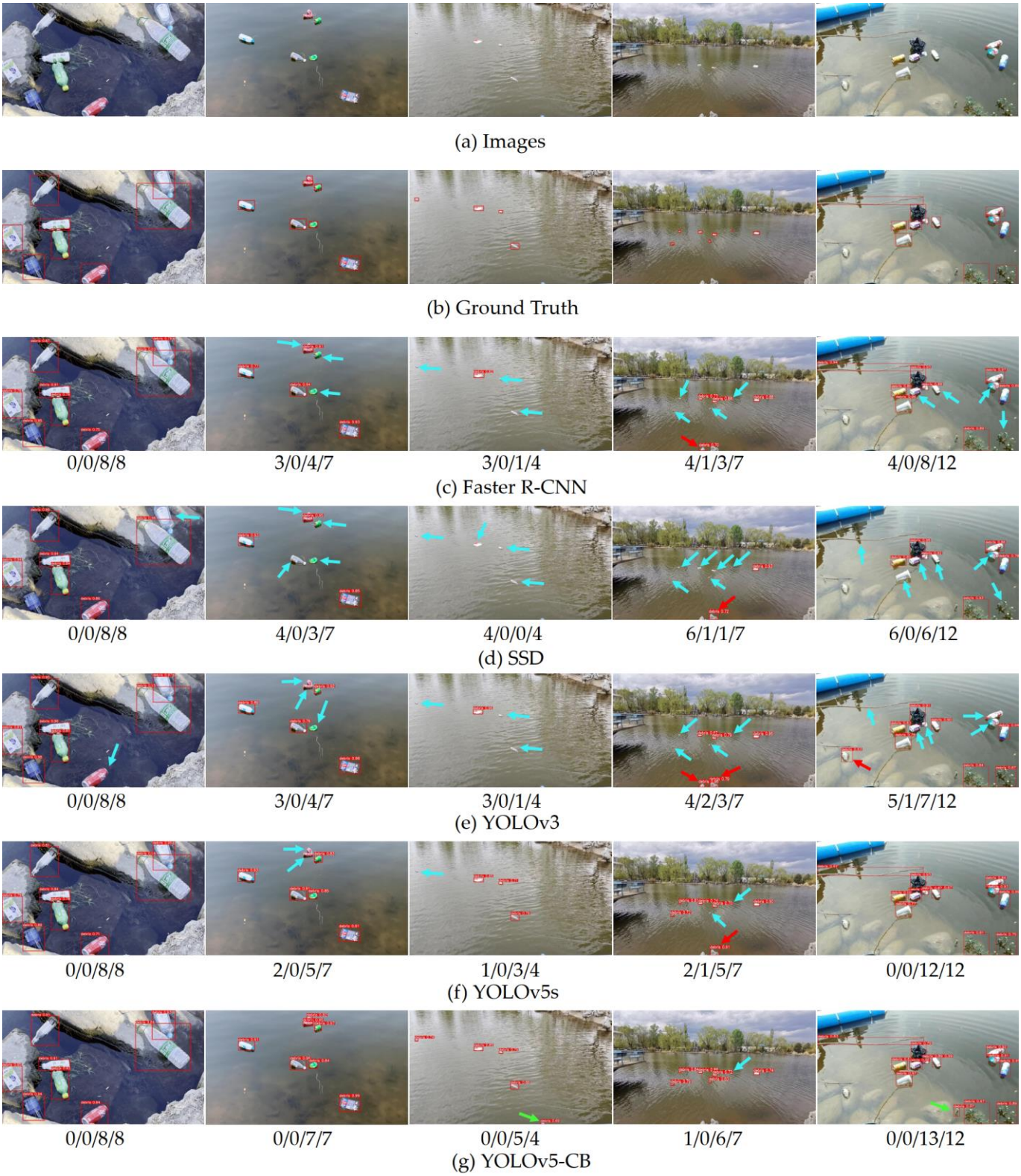


Figure 7. Examples of floating debris detection results for five methods on the SWFD dataset, where the red bounding box represents the result of debris detection, blue arrows highlight some false negative results, red arrows highlight some false positive results, and green arrows highlight some true positives that are not annotated in ground truth. The bottom of the subgraph of each row marks the detected object state, which are false negative/false positive/true positive/total numbers of objects to be checked.

Table 2. Quantitative results based on the SWFD dataset. The most impressive results are highlighted in bold.

Label	Methods	Recall (%)	AP (%)	AP_{50} (%)	AP_{75} (%)	Weights/MB
debris	Faster R-CNN	90.1	90.2	92.8	90.7	158.5
	SSD	77.9	80.8	84.3	81.4	102.3
	YOLOv3	83.8	88.6	91.7	89.9	322.5
	YOLOv5s	93.9	92.4	95.1	93.8	14.1
	Proposed	**96.5**	**95.8**	**97.9**	**97.1**	**15.9**

Table 3. The impact of different improvement strategies on the network based on the SWFD dataset.

Label	Methods	CA	BiFPN	Recall (%)	AP (%)
debris	YOLOv5s			90.1	92.4
	CA-YOLOv5	√		92.2	91.4
	Bi-YOLOv5		√	93.1	92.9
	Proposed	√	√	96.5	95.8

√ indicates an imported module.

Compared with YOLO v5s, CA-YOLOv5 with coordinate attention mechanism and Bi-YOLOv5 with BiFPN achieved better results in terms of Recall and AP, and AP increased by 1.2% and 2.7%, respectively. These results also demonstrate that each optimization method in the improved YOLO v5 is effective. In addition, the proposed method has the characteristics of high precision and small size, and can be quickly transplanted into mobile devices to improve the efficiency of debris detection.

3.4. Validate with the Published System

Finally, we put the proposed approach to the test by setting up a system. The system adopts a B/S architectural design, and the Springboot and Vue frameworks are used to implement the back-end and front-end, respectively. To meet the playback requirements of various clients to the greatest extent, we simultaneously built a streaming media service based on the idea of cloud-edge collaboration, and implemented the streaming media protocol stack typically used in RTSP (Real-Time Streaming Protocol), RTMP (Real-Time Messaging Protocol), HTTP (Hyper Text Transfer Protocol), WS (Websocket), and supported real-time forwarding and synthesis in two ways, choosing different ways based on different scenarios, as shown in Figure 8. For real-time floating debris detection and video encoding and decoding, the intelligent terminal fully utilizes GPU (Graphics Processing Unit) computational capability and includes streaming media protocol stacks such as RTSP/RTMP. The protocol is to send orders to the smart terminal via MQTT (Message Queuing Telemetry Transport), pull the camera's video feed, and simultaneously push the stream in real-time to the server. The video stream is delivered to the web side over HTTP or WS association.

The video stream is delivered to the web side using the HTTP or WS protocol. Taking a lake in the Yanqing Experimental Base of the China Institute of Water Resources and Hydropower Research as the research area, and the floating debris on the lake surface was tested. It should be mentioned that just the functions connected to the detection of floating debris are discussed here since it is a sub-function of the system. The system's ability to recognize floating debris is tested in this section using test data taken from the camera's real-time video feed.

Figure 9 shows the real-time detection of video streams by YOLOv5 and YOLOv5-CB. The detection results are sent to the server and brought to the front-end when the edge machine has finished its detection task. When floating debris is discovered, an automated cleaning task is created and sent to the unmanned cleaning vessel or the accountable party. It has been confirmed that the system enables the smart integration of video recorder equipment access, edge machine inference calculation, and front-end visualization, which

significantly boosts the monitoring and cleaning efficiency of floating debris and serves as a certain benchmark for the design of cloud-edge collaboration architecture.

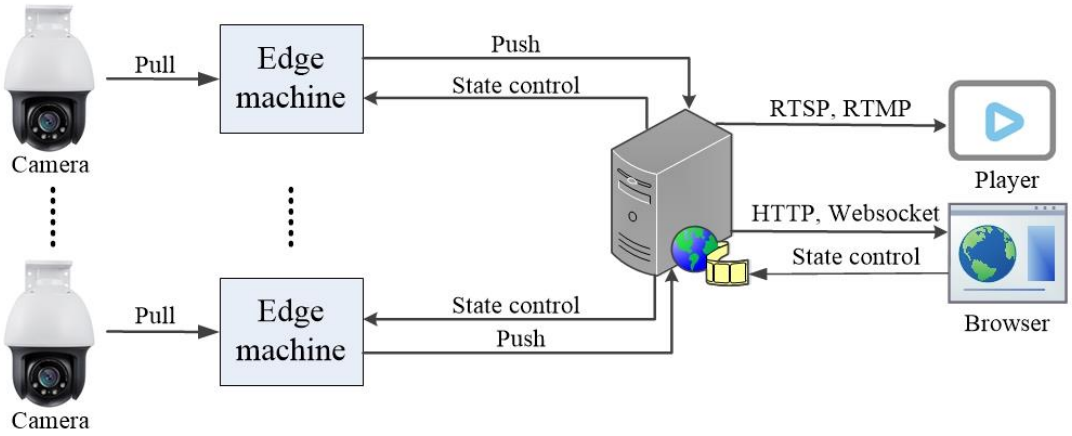
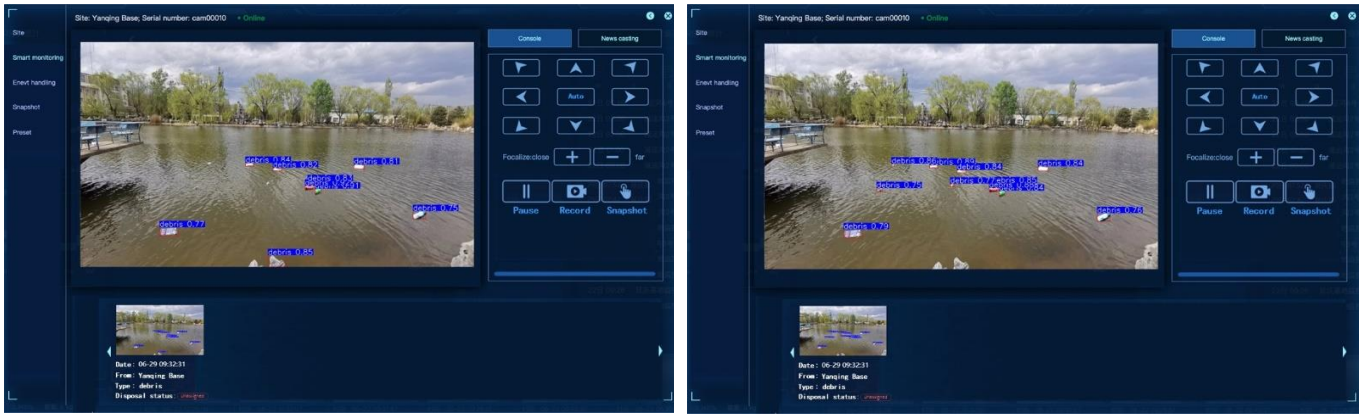


Figure 8. Cloud-edge collaboration architecture.



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

Figure 9. Floating debris detection function of intelligent river patrol system. (**a**) The effect of YOLOv5s real-time detection; (**b**) the effect of YOLOv5-CB real-time detection.

3.5. Importance of Analysis of Lightweight Models in Floating Debris Detection

A timely cleanup of floating debris may significantly improve the water's quality, which is crucial for preserving the natural environment while also enhancing the aesthetic of the surrounding waterways. The river's detection of floating items mostly depends on the human eye, which is a labor-intensive and ineffective process, as was discovered during the inspection and inquiry. Our proposed approach is a lightweight model that is easy to deploy on embedded devices and places little demand on hardware and computing capacity.

4. Conclusions

This study offered an innovative and efficient method for detecting floating debris on surface water. We introduce a coordinate attention mechanism into the YOLOv5 backbone network to include position information into channel attention, enabling the network to acquire information across a larger area. Meanwhile, the BiFPN module replaces the FPN module, and weighted feature fusion and efficient bidirectional cross-scale connection are applied to boost the detection accuracy of tiny floating debris and very small floating debris. The method's efficacy is validated by comparison to other state-of-the-art techniques. The experimental results show that the approach outperforms the other four classical detectors in our self-built floating debris dataset (SWFD).

Author Contributions: Conceptualization, G.Q. and M.Y.; methodology, G.Q.; software, G.Q.; validation, G.Q. and M.Y.; formal analysis, G.Q. and M.Y.; investigation, G.Q. and M.Y.; resources, M.Y.; data curation, G.Q. and M.Y.; writing—original draft preparation, G.Q.; writing—review and editing, G.Q.; visualization, G.Q.; supervision, M.Y. and H.W.; project administration, M.Y. and H.W.; funding acquisition, M.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Beijing Science and technology planning project, grant number Z201100001820022 and National Natural Science Foundation of China, grant number U1865102.

Data Availability Statement: Not applicable.

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

References

1. Gasperi, J.; Dris, R.; Bonin, T.; Rocher, V.; Tassin, B. Assessment of floating plastic debris in surface water along the Seine River. *Environ. Pollut.* **2014**, *195*, 163–166. [CrossRef]
2. Jang, S.W.; Kim, D.H.; Seong, K.T.; Chung, Y.H.; Yoon, H.J. Analysis of floating debris behaviour in the Nakdong River basin of the southern Korean peninsula using satellite location tracking buoys. *Mar. Pollut. Bull.* **2014**, *88*, 275–283. [CrossRef] [PubMed]
3. Grbić, J.; Helm, P.; Athey, S.; Rochman, C.M. Microplastics entering northwestern Lake Ontario are diverse and linked to urban sources. *Water Res.* **2020**, *174*, 115623. [CrossRef] [PubMed]
4. Wagner, S.; Klöckner, P.; Stier, B.; Römer, M.; Seiwert, B.; Reemtsma, T.; Schmidt, C. Relationship between Discharge and River Plastic Concentrations in a Rural and an Urban Catchment. *Environ. Sci. Technol.* **2019**, *53*, 10082–10091. [CrossRef]
5. Zheng, W.; Han, Z.; Zhao, Z. A study on the current situation of floating debris in Haihe River of Tianjin and the Counter-measures. *Environ. Sanit. Eng.* **2001**, *3*, 123–126.
6. Jeevan, G.; Zacharias, G.C.; Nair, M.S.; Rajan, J. An empirical study of the impact of masks on face recognition. *Pattern Recogn.* **2022**, *122*, 108308. [CrossRef]
7. Liu, L.; Lu, S.; Zhong, R.; Wu, B.; Yao, Y.; Zhang, Q.; Shi, W. Computing Systems for Autonomous Driving: State of the Art and Challenges. *IEEE Internet Things J.* **2021**, *8*, 6469–6486. [CrossRef]
8. Xie, J.; Pang, Y.; Khan, M.H.; Anwer, R.M.; Khan, F.S.; Shao, L. Mask-Guided Attention Network and Occlusion-Sensitive Hard Example Mining for Occluded Pedestrian Detection. *IEEE Trans. Image Process.* **2021**, *30*, 3872–3884. [CrossRef]
9. Zeng, Z.; Liu, B.; Fu, J.; Chao, H. Reference-Based Defect Detection Network. *IEEE Trans. Image Process.* **2021**, *30*, 6637–6647. [CrossRef]
10. Han, P.; Ma, C.; Li, Q.; Leng, P.; Bu, S.; Li, K. Aerial image change detection using dual regions of interest networks. *Neurocomputing* **2019**, *349*, 190–201. [CrossRef]
11. Tsai, J.; Hung, I.Y.; Guo, Y.L.; Jan, Y.; Lin, C.; Shih, T.T.; Chen, B.; Lung, C. Lumbar Disc Herniation Automatic Detection in Magnetic Resonance Imaging Based on Deep Learning. *Front. Bioeng. Biotechnol.* **2021**, *9*, 708137. [CrossRef] [PubMed]
12. Ojha, S.; Sakhare, S. Image processing techniques for object tracking in video surveillance—A survey. In Proceedings of the 2015 International Conference on Pervasive Computing (ICPC), Pune, India, 8–10 January 2015.
13. Lin, Y.; Zhu, Y.; Shi, F.; Yin, H.; Yu, J.; Huang, P.; Hou, D. Image Processing Techniques for UAV Vision-Based River Floating Contaminant Detection. In Proceedings of the 2019 Chinese Automation Congress (CAC2019), Hangzhou, China, 22–24 November 2019; pp. 89–94.
14. Zhang, L.; Wei, Y.; Wang, H.; Shao, Y.; Shen, J. Real-Time Detection of River Surface Floating Object Based on Improved RefineDet. *IEEE Access* **2021**, *9*, 81147–81160. [CrossRef]
15. Lin, F.; Hou, T.; Jin, Q.; You, A. Improved YOLO Based Detection Algorithm for Floating Debris in Waterway. *Entropy* **2021**, *23*, 1111. [CrossRef] [PubMed]
16. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–18 June 2013.
17. Zhu, L.; Geng, X.; Li, Z.; Liu, C. Improving YOLOv5 with Attention Mechanism for Detecting Boulders from Planetary Images. *Remote Sens.* **2021**, *18*, 3776. [CrossRef]
18. Jin, S.; Sun, L. Application of Enhanced Feature Fusion Applied to YOLOv5 for Ship Detection. In Proceedings of the 33rd Chinese Control and Decision Conference (CCDC 2021), Kunming, China, 22–24 May 2021.
19. Shi, X.; Hu, J.; Lei, X.; Xu, S. Detection of Flying Birds in Airport Monitoring Based on Improved YOLOv5. In Proceedings of the 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021; pp. 1446–1451.
20. Chai, E.H.; Zhi, M. Rapid Pedestrian Detection Algorithm Based on Deformable Part Model. In Proceedings of the Ninth International Conference on Digital Image Processing (ICDIP 2017), Hong Kong, China, 19–22 May 2017; Volume 10420.
21. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
22. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal.* **2017**, *39*, 1137–1149. [CrossRef]

23. Dai, J.F.; Li, Y.; He, K.M.; Sun, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. In *Advances in Neural Information Processing Systems 29 (NIPS 2016), Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain, 5–10 December 2016*; Lee, D.D., Sugiyama, M., Luxburg, U.V., Guyon, I., Garnett, R., Eds.; NeurIPS: San Diego, CA, USA, 2016; Volume 29, p. 29.
24. Pang, J.; Chen, K.; Shi, J.; Feng, H.; Ouyang, W.; Lin, D. Libra R-CNN: Towards Balanced Learning for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
25. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. *arXiv* **2016**, arXiv:1612.08242.
26. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
27. Bochkovskiy, A.; Wang, C.; Liao, H.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
28. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
29. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *arXiv* **2015**, arXiv:1512.02325.
30. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.M.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal.* **2020**, *42*, 318–327. [CrossRef]
31. Liu, S.; Qi, L.; Qin, H.F.; Shi, J.P.; Jia, J.Y. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
32. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. *arXiv* **2021**, arXiv:2103.02907.
33. Jie, H.; Li, S.; Gang, S.; Albanie, S. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal.* **2020**, *42*, 2011–2023.
34. Park, J.; Woo, S.; Lee, J.; Kweon, I.S. BAM: Bottleneck Attention Module. *arXiv* **2018**, arXiv:1807.06514.
35. Woo, S.H.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In *Computer Vision—ECCV 2018, PT VII, Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland, 2018; pp. 3–19.
36. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. *arXiv* **2019**, arXiv:1911.09070.