Journal of
Marine Science and Engineering

MDPI

Article

Enhanced YOLOv7 for Improved Underwater Target Detection

Daohua Lu [1,2,*], Junxin Yi [1] and Jia Wang [1]

[1] School of Mechanical Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, China; 15161085985@163.com (J.Y.); wjjzhb@just.edu.cn (J.W.)
[2] Marine Equipment and Technology Institute, Jiangsu University of Science and Technology, Zhenjiang 212003, China
* Correspondence: ludaohua_just@126.com

Abstract: Aiming at the problems of the underwater existence of some targets with relatively small size, low contrast, and a lot of surrounding interference information, which lead to a high leakage rate and low recognition accuracy, a new improved YOLOv7 underwater target detection algorithm is proposed. First, the original YOLOv7 anchor frame information is updated by the K-Means algorithm to generate anchor frame sizes and ratios suitable for the underwater target dataset; second, we use the PConv (Partial Convolution) module instead of part of the standard convolution in the multi-scale feature fusion module to reduce the amount of computation and number of parameters, thus improving the detection speed; then, the existing CIou loss function is improved with the ShapeIou_NWD loss function, and the new loss function allows the model to learn more feature information during the training process; finally, we introduce the SimAM attention mechanism after the multi-scale feature fusion module to increase attention to the small feature information, which improves the detection accuracy. This method achieves an average accuracy of 85.7% on the marine organisms dataset, and the detection speed reaches 122.9 frames/s, which reduces the number of parameters by 21% and the amount of computation by 26% compared with the original YOLOv7 algorithm. The experimental results show that the improved algorithm has a great improvement in detection speed and accuracy.

Keywords: underwater target detection; YOLO7; loss function; attention mechanism

check for updates

Citation: Lu, D.; Yi, J.; Wang, J. Enhanced YOLOv7 for Improved Underwater Target Detection. *J. Mar. Sci. Eng.* **2024**, *12*, 1127. https://doi.org/10.3390/jmse12071127

Academic Editors: Yan-Tsung Peng, Wenqi Ren, Jingchun Zhou and Qiuping Jiang

Received: 6 June 2024
Revised: 28 June 2024
Accepted: 3 July 2024
Published: 4 July 2024

1. Introduction

Occupying a large part of the earth's area, the ocean is an important source of oil, natural gas, and mineral resources, attracting extensive attention from some adventurers and researchers, which inevitably leads to an increasing number of ocean exploration activities [1]. As an important technology for marine exploration activities, underwater target detection is widely used in archaeology, marine environment monitoring, underwater navigation, fish farming, and other fields, and has received continuous attention [2–7]. However, the harsh marine environment makes underwater target detection still face various challenges.

The main challenges we are currently facing in underwater target detection are as follows: first, we have found from existing underwater datasets and images from real applications for which wavelength-dependent absorption and scattering [8] degrade the quality of underwater images, causing visibility, weak contrast, and color variations, which make the underwater targets blurry. Then, because some underwater organisms have swarming habits and prefer to attach to objects such as mud, sand, and coral reefs, this causes the underwater target and the background area to obscure each other, making the detection accuracy suffer. Finally, because some embedded devices have limited computational power, which leads to the fact that some network models with large parameter counts cannot be applied to underwater embedded devices, so people are also actively pursuing underwater detection devices that can be applied to large parameters [9].

In recent years, underwater target detection methods based on deep learning have been widely studied and gradually applied in the field of underwater biological detection. Deep learning-based underwater target detection is based on learning features iteratively, taking the output of each layer as the input of the next layer, and transforming the detailed features of the lower layer into the detailed features of the higher layer through nonlinear mapping between the lower layer and the higher layer. However, due to the relatively small dataset for underwater object detection, the available learning features are limited, and most underwater organisms occur in groups and are relatively small and fuzzy. Current deep learning-based detection algorithms are unable to effectively detect small objects, which poses a great challenge to the target detection task. To cope with this challenge, the YOLO series of algorithms now exist, including YOLOv1 [10], YOLO9000 [11], YOLOv3 [12], YOLOv4 [13], YOLOv5 [14], YOLOv6 [15], YOLOv7 [16], YOLOv8 [17], etc. The YOLO (You Only Look Once) series comprises typical single-stage target detection algorithms, which have faster detection speed and wider application scope than two-stage target detection algorithms such as RCNN [18], Fast-RCNN [19], Faster RCNN [20], Mask-RCNN [21], Cascade-RCNN [22], and SSD [23]. In order to adapt to different application scenarios and improve the detection accuracy for different detection objects, many researchers have successively proposed a variety of improved detection methods based on the YOLO series of algorithms. The literature [24] has proposed an underwater target detection algorithm, YOLOv5s-CA, based on an improved YOLOv5, to improve the detection accuracy by embedding a Coordinate Attention (CA) module and a Squeeze-and-Excitation (SE) module. The literature [25] also proposes a lightweight algorithm based on an improved YOLOv4, aiming to make the model have a smaller number of parameters and smaller size by combining MobileNetv2 and depth-separable convolution.

However, the target size of the underwater target detection studied in this paper is relatively small, and the previous YOLOv3, YOLOv4, and YOLOv5 algorithms are less effective at detecting small targets, and only the YOLOv7 algorithm has a better detection performance in small target detection at present. Therefore, for the different characteristics of underwater organisms, an underwater target detection algorithm based on the improved YOLOv7 is proposed, with the following four main improvements:

(1) The K-Means algorithm is used to re-cluster the underwater target dataset, and the anchor frames produced by the clustering are more in line with the smaller size of the underwater targets studied in this paper, which accelerates the convergence speed of the model and improves the detection accuracy of small-sized targets.

(2) Pconv (Partial Convolution) is introduced to optimize the multi-scale feature fusion module, which reduces the parameter count and computational workload while improving the detection speed.

(3) The ShapeIou_NWD loss function is used instead of the CIou loss function, which effectively solves the problem of slow convergence and reduced prediction accuracy during the training process by ignoring the shape and scale of the bounding box itself, as well as the possibility of detecting targets that are too small.

(4) Introducing the SimAM attention mechanism after the multi-scale feature fusion module enhances the detection accuracy of the network model without parameter increase.

Finally, several comparative experiments and ablation experiments are conducted to prove the effectiveness of the improved algorithm in this paper.

2. Materials and Methods

2.1. YOLOv7 Model

Compared with its variants YOLOv7-D6, YOLOv7-E6, YOLOv7-E6E, YOLOv7-W6, and YOLOv7-X, YOLOv7 has the smallest number of parameters, the fastest detection speed, and is most suitable to be applied to underwater target detection. Therefore, YOLOv7 is selected as the base algorithm to be improved in this paper. The network structure of YOLOv7 is shown in Figure 1.

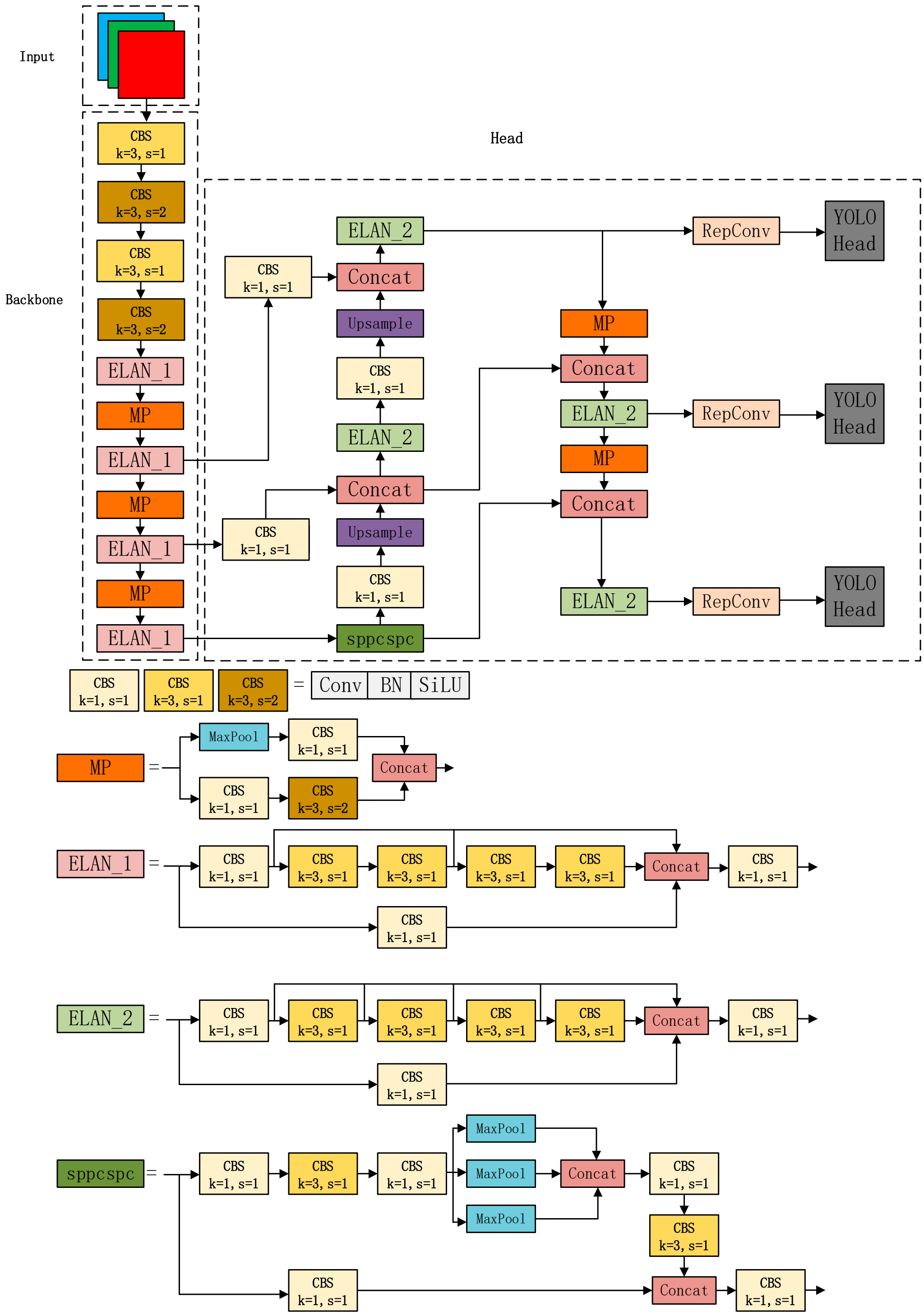


Figure 1. Illustration of the YOLOv7 network structure.

As can be seen from Figure 1, the network structure of YOLOv7 consists of three parts: Input, Backbone, and Head. Input scales the image to be detected to a fixed size after inputting the image to fulfill the input demands of the Backbone, and then carries out the feature extraction through the Backbone, which contains three structures: CBS (Conv + BN + SiLU), ELAN_1, and MP. The CBS mainly uses convolutional layers for feature extraction, and ELAN_1, as a kind of efficient layer aggregation network, greatly improves the learning ability of the detected objects and enriches the diverse feature learning. The MP structure adds a MaxPool layer on top of the CBS, and the upper and lower layers are contacted. The MP structure adds the MaxPool layer on top of the CBS, which greatly improves the feature extraction ability by contacting the upper and lower layers for feature fusion of the features extracted from the branches. In the Head part, the SPPCSPC pyramid structure is employed to broaden the receptive field and make the Head network suitable for multi-size inputs, and then the bottom information is passed upwards from the bottom to the top through the pyramid network structure, so as to realize the fusion of feature information of different scales, and after fusing semantic information and spatial information at multiple scales, three target feature layers of different scales are outputted, which are respectively reused through the RepConv structure. The three obtained feature layers are reparametrized by the RepConv structure to adjust the number of channels, then the preliminary prediction results are obtained by the YOLO Head, and finally the preliminary prediction results are subjected to some post-processing operations, such as confidence filtering, NMS (Non-Maximum Suppression), and so on, in order to obtain the final results.

2.2. Kmeans

The a priori frame sizes used in YOLOv7 were obtained by clustering on the MS COCO dataset using the K-Means [26] algorithm. The MS COCO dataset contains about 41% small targets (area < 32 × 32), and the rest are medium vs. large targets. However, in the marine life dataset, marine organisms are predominantly small- to medium-sized targets, which is significantly different from the size of the objects in the COCO dataset. Therefore, the original anchor frame data size is not suitable for the dataset in this paper. In order to improve the matching probability between the underwater targets and anchor frames, the anchor frame size was redesigned using the K-means clustering algorithm. The basic process is described as follows:

- Randomly select k samples from the dataset as initial clustering centers $c = \{c_1, c_2, \ldots, c_k\}$.
- For each sample in the dataset, calculate its distance from the K cluster centers and assign it to the class corresponding to the cluster center with the smallest distance.
- For each category c_i, recalculate its clustering center $c_i = \frac{1}{|c_i|}\sum_{x \in c_i} x$ (i.e., the center of mass of all samples belonging to the class).
- Repeat steps 2 and 3 until the position of the clustering center no longer changes.

Compared with the K-Means++ [27] algorithm, the K-Means algorithm is simple in principle and easy to implement, only one hyperparameter k needs to be adjusted, and the K-Means++ algorithm relies on the already selected centroids each time the next centroids are selected.

Therefore, in this paper, the K-Means algorithm is selected to recluster the anchor frames. The anchor frames after reclustering using K-Means are presented in Table 1 below:

Table 1. Optimized K-Means algorithm for clustering anchor frame parameters.

Feature Map Size	80 × 80	40 × 40	20 × 20
YOLOv7	(21, 17) (28, 24) (35, 32)	(52, 35) (44, 46) (57, 56)	(80, 51) (82, 79) (140, 117)

2.3. ELAN_PC

ELAN [28], mainly composed of VoVNet [29] and CSPNet [30], is an efficient layer aggregation network. It facilitates the network to acquire more features by regulating the shortest and longest gradient pathways. ELAN is mainly composed of two branches, the first branch mainly doing the change of channel number through the convolution of 1 × 1, and the second branch firstly doing the change of channel number through the convolution of 1 × 1, then doing the feature extraction through the four convolutional modules of 3 × 3, and finally superimposing the results of the two branches together. The ELAN can then effectively alleviate the problem of gradient disappearance when the model reaches a certain depth. However, the optimization of the ELAN network in terms of the number of parameters and the amount of computation is not ideal, so in this paper, while ensuring the structural integrity of the ELAN network, PConv is introduced to construct the ELAN_PC_1 and ELAN_PC_2 network modules, which use PConv to replace the convolution kernel of 3 × 3 convolutional layers in the ELAN network. The ELAN_PC_1 and ELAN_PC_2 network structures are shown in Figure 2.

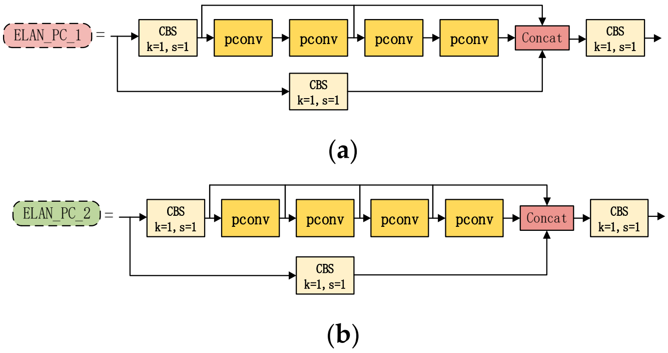


Figure 2. Improved ELAN structure; (**a**) ELAN_PC_1; (**b**) ELAN_PC_2.

PConv [31] is a new lightweight convolution module that reduces computational redundancy while reducing memory accesses. In this module, we employ a convolutional operation on a subset of the input channels to avoid excessive redundancy in the feature map. This approach, illustrated in Figure 3, ensures that the convolutional operation is applied to a select set of channels while the remaining channels remain unchanged.

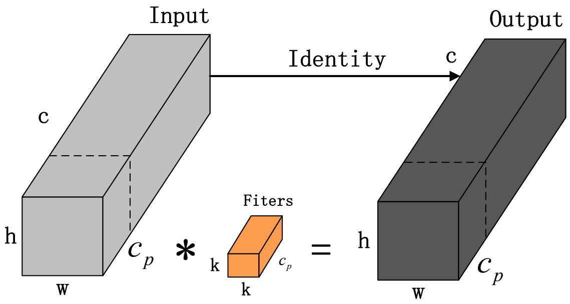


Figure 3. The structure diagram of PConv module.

For continuous or regular memory accesses, the input or output c_p channels are computed as representative of the entire feature map. The PConv is computed as $h \times w \times k^2 \times c_p^2$, while the regular convolution is computed as $h \times w \times k^2 \times c^2$, the memory accesses for the PConv are $h \times w \times 2c_p$, and the memory accesses for the regular convolution are $h \times w \times 2c$. If $c_p = c/4$, then the PConv is computed as 1/16 of the regular convolution, and the memory accesses are 1/4.

It can be seen that the introduction of the PConv convolution module into the ELAN module can significantly reduce the amount of computation and memory access, thus making the model lightweight and speeding up the inference.

2.4. ShapeIoU_NWD

The YOLOv7 model employs the CIOU [32] loss function, which is designed to account for the overlap area, centroid distance, and aspect ratio of the bounding box regression. However, this loss function does not consider the influence of the inherent properties of the bounding box itself, such as shape and scale. Consequently, the CIOU loss function may prevent the model from optimizing the similarity when the aspect ratio is the same [33]. To solve this problem, this paper adopts ShapeIoU [34], a bounding box regression method that focuses on the shape and scale of the bounding box itself. This method can calculate the loss by focusing on the shape and scale of the bounding box itself, which makes the bounding box regression more accurate. The parameter schematic of ShapeIoU is presented in Figure 4.

$$\text{IoU} = \frac{\left|B \cap B^{gt}\right|}{\left|B \cup B^{gt}\right|} \tag{1}$$

$$ww = \frac{2 \times \left(w^{gt}\right)^{scale}}{\left(w^{gt}\right)^{scale} + \left(h^{gt}\right)^{scale}} \tag{2}$$

$$hh = \frac{2 \times \left(h^{gt}\right)^{scale}}{\left(w^{gt}\right)^{scale} + \left(h^{gt}\right)^{scale}} \tag{3}$$

$$\text{distance}^{shape} = hh \times \left(x_c - x_c^{gt}\right)^2 / c^2 + ww \times \left(y_c - y_c^{gt}\right) / c^2 \tag{4}$$

$$\Omega^{shape} = \sum_{t=w,h} \left(1 - e^{-\omega_t}\right)^{\theta}, \theta = 4 \tag{5}$$

$$\begin{cases} \omega_w = hh \times \frac{\left|w - w^{gt}\right|}{\max\left(w, w^{gt}\right)} \\ \omega_h = ww \times \frac{\left|h - h^{gt}\right|}{\max\left(h, h^{gt}\right)} \end{cases} \tag{6}$$

where B represents the prediction frame, B^{gt} represents the real frame, w^{gt} and h^{gt} represent the width and height of the real frame respectively, scale is the scale factor, which is related to the scale of the target in the dataset, ww and hh are the weight coefficients in the horizontal and vertical directions, respectively, and their values are related to the shape of the real frame. Its corresponding bounding box regression loss is as follows:

$$L_{ShapeIoU} = 1 - IoU + \text{distance}^{shape} + 0.5 \times \Omega^{shape} \tag{7}$$

In order to further improve the accuracy of underwater small target detection, we have introduced the Normalized Wasserstein Distance [35]. In this approach, the Gaussian distributions N_a and N_b, modeled for the two bounding boxes $A = (c_{xa}, c_{ya}, w_a, h_a)$ and $B = (c_{xb}, c_{yb}, w_b, h_b)$, can be expressed as follows:

$$W_2^2(N_a, N_b) = \left\| \left(\left[cx_a, cy_a, \frac{w_a}{2}, \frac{h_a}{2} \right]^T, \left[cx_b, cy_b, \frac{w_b}{2}, \frac{h_b}{2} \right]^T \right) \right\|_2^2 \tag{8}$$

However, since $W_2^2(N_a, N_b)$ is a distance measure and not a similarity measure, it cannot be directly used as such. Therefore, in order to address this limitation, it is necessary to normalize the distance by transforming $W_2^2(N_a, N_b)$ into a value between 0 and 1. This process results in the normalized Gaussian Wasserstein distance (NWD), which enables more effective and meaningful comparisons and evaluations in the context of detecting small underwater objects. The formula is as follows:

$$NWD(\mathrm{N_a}, N_b) = \exp\left(-\frac{\sqrt{W_2^2(\mathrm{N_a}, N_b)}}{C} \right) \tag{9}$$

Here, C is a constant related to the dataset.

Therefore, the loss of our *ShapeIoU_NWD* can be obtained by taking the form of a weighted summation based on the introduced $_{ShapeIoU}$ loss function and NWD metric with the following formula:

$$L_{ShapeIoU\_NWD} = (1 - r) \times (1 - NWD(N_a, N_b)) + r \times L_{ShapeIoU} \tag{10}$$

where r is a scaling factor.

In this paper, we use the ShapeIoU_NWD loss function instead of the CIOU loss function, which takes into account both the effects of changes in the shape and scale of the bounding box itself, as well as the sensitivity of small target displacements, and provides a significant improvement in the detection accuracy of small- and medium-sized targets.

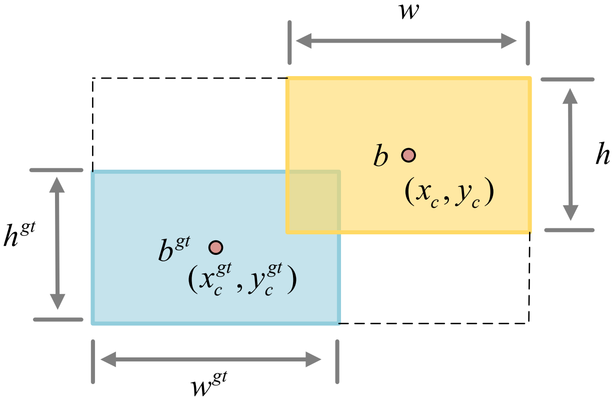


Figure 4. ShapeIou parameter diagram.

2.5. SimAM

The size of underwater targets is relatively small, and because of the presence of some mud, sand, and some other interfering background information, it occupies less information in the whole image. In order to improve the accuracy of small target detection and reduce the interference of background information, this paper adopts the attention mechanism to adaptively focus on the detail information related to the small target, and reduces the attention to other interference information. In contrast to the SE [36] (Squeeze-and-Excitation) channel attention mechanism, which solely considers the internal channel information, and the CBAM [37] (Convolutional Block Attention Module), which focuses on the local spatial location range, the SimAM [38] attention mechanism does not introduce additional network parameters for feature maps, thereby deriving 3D attention weights. Furthermore, it is a plug-and-play feature that can be integrated into any position within the model. Consequently, in this paper, three SimAM attention mechanisms are incorporated after the multi-scale feature fusion module with the objective of enhancing the detection accuracy of the model. The structure of the SimAM attention mechanisms is depicted in Figure 5.

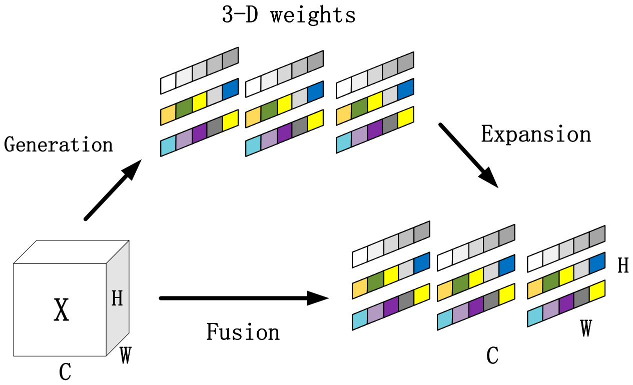


Figure 5. SimAM attention mechanism structure.

The core idea of SimAM is based on the local self-similarity of images. In an image, neighboring pixels usually have strong similarity to each other, while the similarity between distant pixels is weak. SimAM takes advantage of this property to generate attention weights by calculating the similarity between each pixel in the feature map and its neighboring pixels. The formula is shown below:

$$\dot{X} = \text{sigmoid}\left(\frac{1}{E}\right) \odot X \tag{11}$$

$$E = \frac{4\left(\sigma^2 + \lambda\right)}{(t-\mu)^2 + 2\sigma^2 + 2\lambda} \tag{12}$$

$$\mu = \frac{1}{M}\sum_{i=1}^{M} x_i \tag{13}$$

$$\sigma^2 = \frac{1}{M}\sum_{i=1}^{M}(x_i - \mu)^2 \tag{14}$$

where X is the input feature, E is the energy function on each channel. In order to prevent the possibility of too large a value for E, a sigmoid function is used to limit it; t is the value of the input feature, λ is a constant value, which is 1×10^{-4}, μ and σ^2 denote the mean and variance on each channel in X, respectively, and M = H × W denotes the number of values on each channel.

2.6. Proposed Improved Algorithm

In comparison to other one-stage target detection algorithms, such as YOLOv3 and YOLOv4, YOLOv7 exhibits superior performance in detecting small targets. However, the detection effect is less satisfactory when there are problems such as occlusion and low contrast of the target. Therefore, the following improvements are made to YOLOv7 to address these problems. Figure 6 illustrates the network structure of the improved YOLOv7 algorithm.

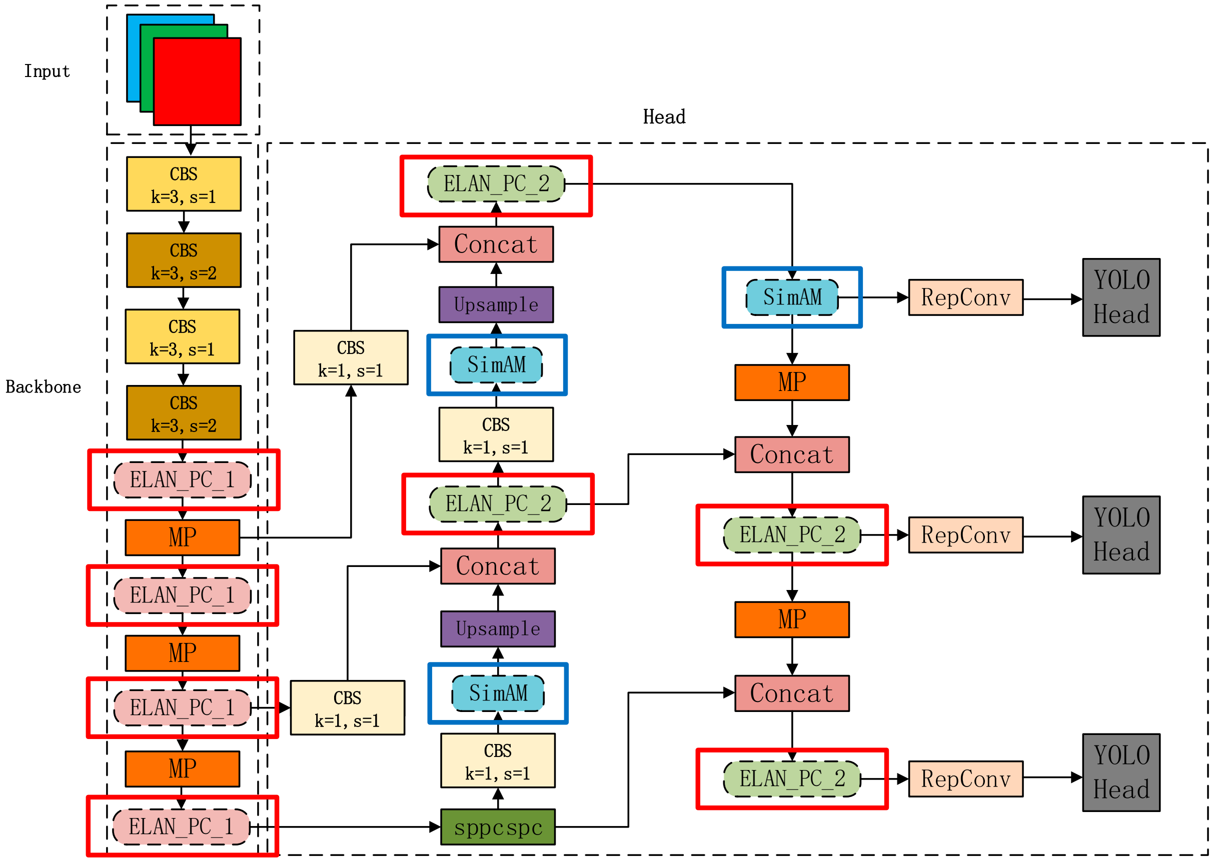


Figure 6. Improved YOLOv7 network structure diagram.

3. Experiments

3.1. Experimental Dataset

The images for underwater target detection studied in this paper are from the 2020 National Underwater Robotics Professional Competition (URPC) dataset, which consists of 5543 images, with an image size of 640 × 640. Based on the definition of small, medium, and large targets in the COCO dataset, each type of target was divided into small, medium, and large targets, where the number of holothurian tags was 5537, in which the number of small targets was 61, the number of medium targets was 3487, and the number of large targets was 1989; the number of echinus tags was 22,343, in which the number of small targets was 2788, the number of medium targets was 13,696, and the number of large targets was 5859; the number of scallop tags was 6720, with the number of small targets 603, the number of medium targets was 1359, and the number of large targets 4758; and the number of starfish tags 6841, with the number of small targets 588, the number of medium targets 4059, and the number of large targets 2194. The results are shown in Figure 7.

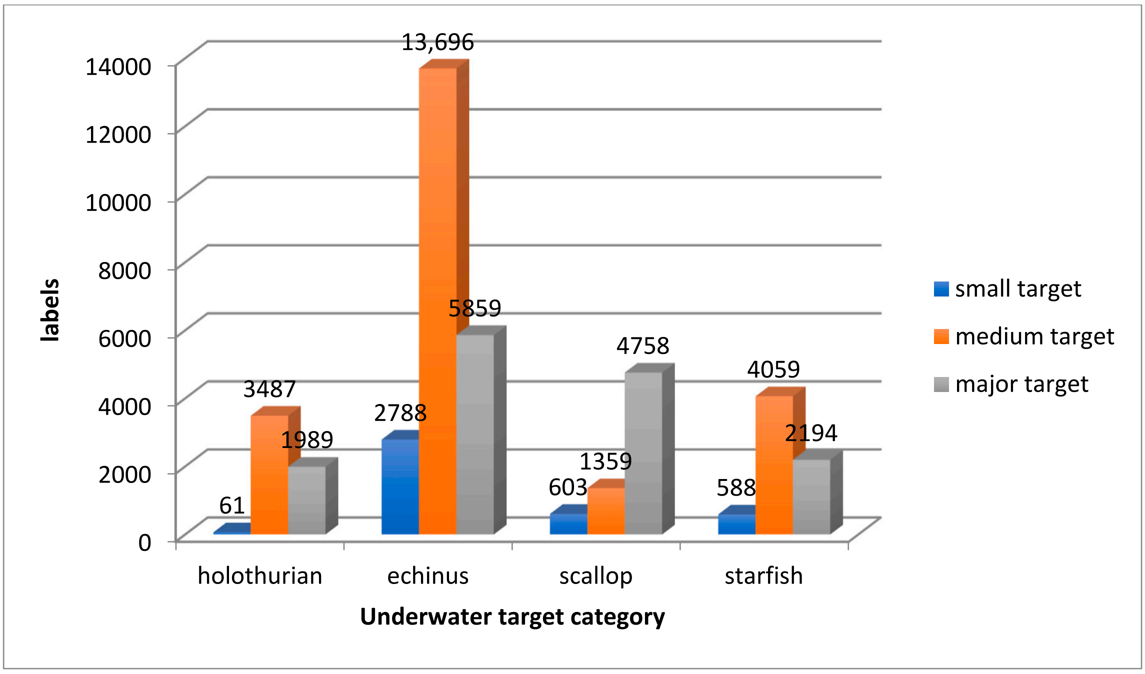


Figure 7. Number of tags for each type of target in the underwater target data set.

The sample images, labeled with location and category information, were subsequently stored in PASCAL VOC format. The dataset was randomly divided into a training set of 4434 images and a validation set of 1109 images according to the corresponding marine organism category in a ratio of 8:2.

3.2. Experimental Environment and Parameterization

This experiment is implemented in the ubuntu20.04 operating system based on the PyTorch deep learning framework, the GPU selection for the size of the memory 12 G NVIDIA GeForce RTX 3080Ti, the CPU configuration is a 12-core AMD Ryzen 9 5900X, PyTorch version 2.0.1, CUDA version 11.7, and the python language environment is 3.8.17.

The hyperparameters for this experiment are configured as follows: in the model training phase, the parameters are tuned using the SGD optimizer, with the initial learning rate set to 0.01 and momentum to 0.937, and the learning rate decayed using warm up, with the weight decay coefficient set to 0.0005. In addition, the batch size is set to 8, and a total of 150 rounds are trained.

3.3. Experimental Design

3.3.1. Evaluation Metrics

This experiment focuses on underwater target detection algorithms, which should satisfy both real-time and good detection performance, so the values of FPS, FLOPS, Params, mAP, precision and recall are used as evaluation indexes to compare the algorithms. FPS represents the frame rate, indicating the number of images processed per second. FLOPS represents the number of floating-point operations executed, serving as a measure of the model's computational complexity, Params is the sum of the model parameters, which are used to evaluate the size of the model, and mAP is the average value of the precision of each type of underwater target detection. The formula for mAP is as follows:

$$mAP = \frac{\sum\limits_{i=0}^{n} AP(i)}{n} \tag{15}$$

where n is the number of underwater target classes and AP (Average Precision) is the precision value of each class.

Additionally, the precision and recall values are derived through the following calculations:

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

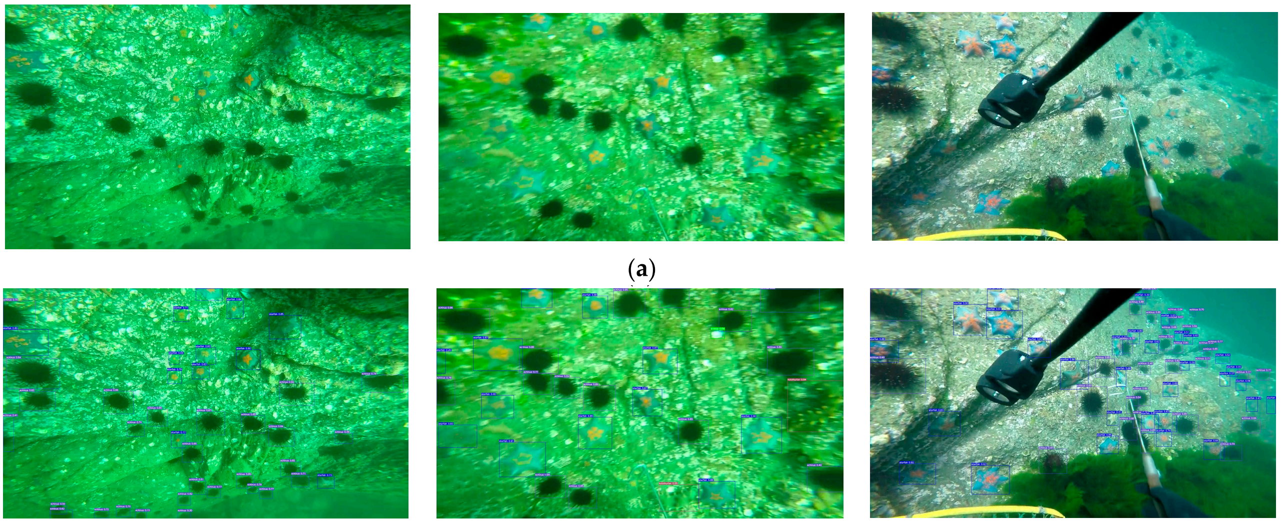
$$Recall = \frac{TP}{TP + FN} \tag{17}$$

where TP is the number of positive samples predicted to be positive (true positive); FP is the number of negative samples predicted to be positive (false positive); FN is the number of positive samples predicted to be negative (false negative).

3.3.2. Experimental Results

Three images are extracted from the dataset for comparison. The three original images extracted are shown in Figure 8a. Using YOLOv7 and this paper's algorithm to detect these three pictures, the results will be detected for comparison, and the comparison results are presented in Figure 8b,c.

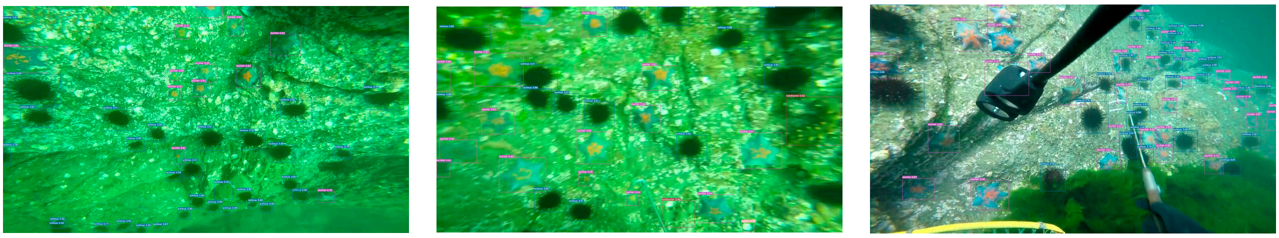
As illustrated in Figure 8, the YOLOv7 algorithm is more prone to miss detection and partially occluded cases are not recognized or recognized with lower confidence values compared to this paper's algorithm, whereas this paper's algorithm is able to detect as accurately as possible all types of underwater targets on the map.



(**a**)

(**b**)

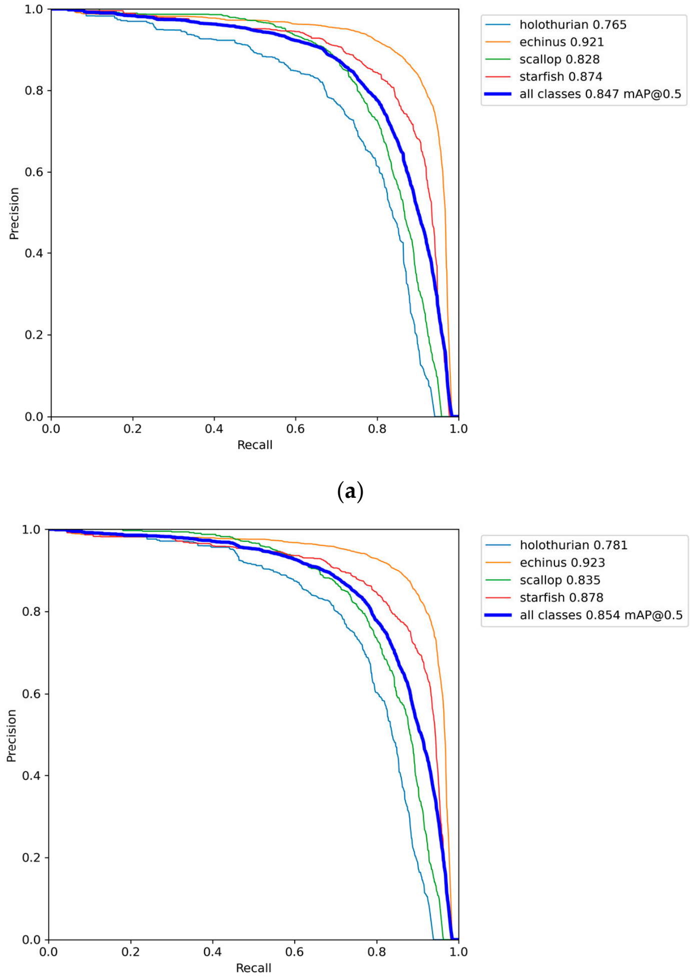
Figure 8. *Cont.*



(**c**)

Figure 8. Comparison chart of detection results of two algorithms; (**a**) The raw image; (**b**) YOLOv7 algorithm detection results; (**c**) Improved YOLOv7 algorithm detection results.

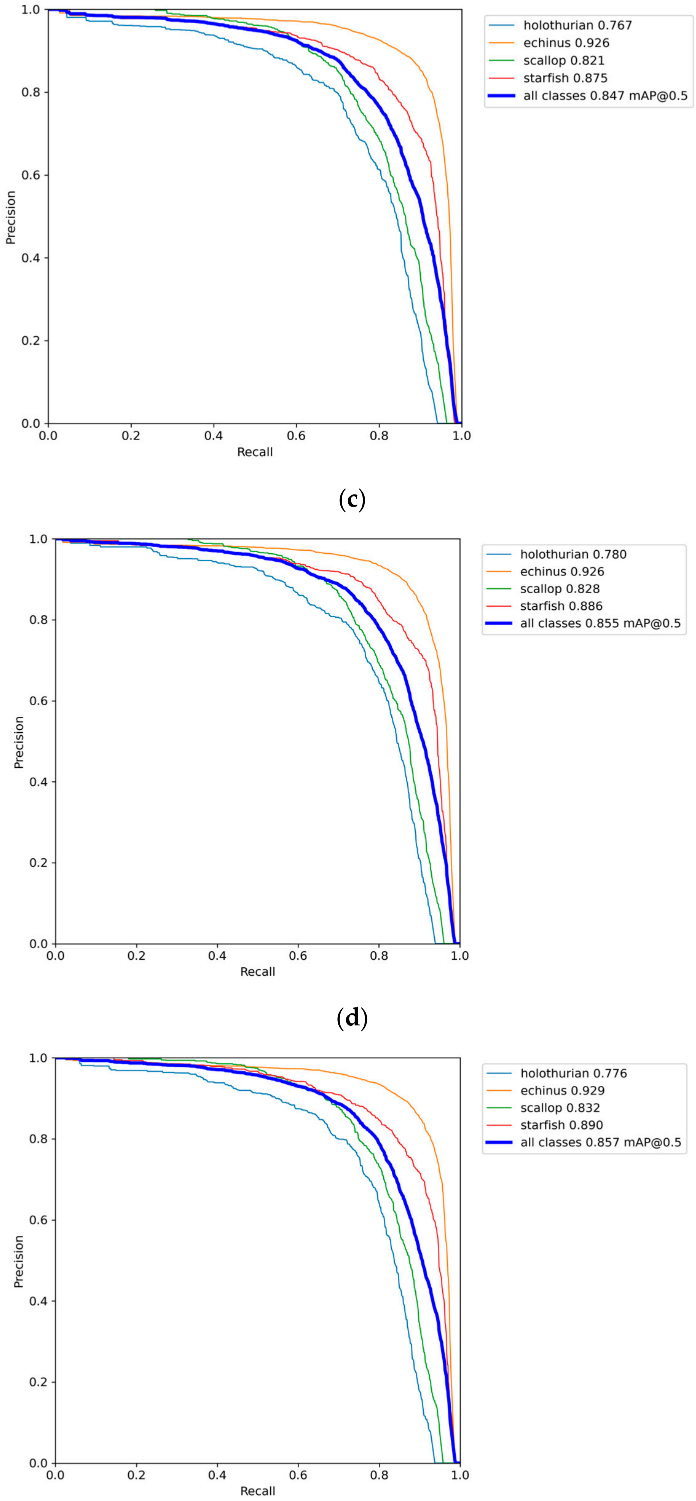
Figure 9 illustrates the precision-recall (P-R) curves of the original YOLOv7 algorithm and the modified YOLOv7 algorithm after improving each module, which were trained on the identical underwater target dataset. Specifically, Figure 9a illustrates the P-R curves of the YOLOv7 algorithm, Figure 9b represents the P-R curves of the YOLOv7_kmeans, Figure 9c represents the P-R curves of the YOLOv7_kmeans with the addition of the ELAN_PC module, and Figure 9d represents the P-R curves of the YOLOv7_kmeans with the addition of the ELAN_PC module and the P-R curve of the ShapeIou NWD loss function. Figure 9e represents the P-R curve for the final addition of the SimAM attention mechanism. As can be seen from Figure 9a,e, the AP of holothurian increased by 1.1%, the AP of echinus increased by 0.8%, the AP of scallop increased by 0.4%, and the AP of starfish increased by 1.6%.



(**a**)

(**b**)

Figure 9. *Cont.*


(**c**)

(**d**)

(**e**)

Figure 9. (**a**) The P-R curves of the original YOLOv7 algorithm; (**b**) P-R curve for the YOLOv7_kmeans; (**c**) P-R curve of the ELAN_PC module added to YOLOv7_kmeans; (**d**) P-R curves with the ELAN_PC module and the ShapeIoU_NWD loss function added to the YOLOv7_kmeans; (**e**) Final improved YOLOv7 algorithm.

4. Discussion

4.1. Ablation Experiments

CBAM, CPCA, CA, GAM, and SimAM are introduced after the multi-scale feature fusion module, and the network is trained on the basis of the K-Means algorithm, the ELAN_PC structure, and the ShapeIou_NWD, the obtained detection results are compared according to the average accuracy value, and the comparison results are presented in Table 2.

As illustrated in Table 2, it is evident that the introduction of attention mechanisms such as CBAM, CPCA, CA, and GAM reduced the average accuracy value compared to the original algorithms that did not incorporate an attention mechanism, and the 85.5% accuracy value of the algorithms that did not incorporate an attention mechanism was reduced by 0.2%, 1.1%, 0.2%, and 2.7%, respectively. By integrating the SimAM attention mechanism, the average accuracy value witnessed a 0.2% increase. In summary, the utilization of the SimAM attention mechanism demonstrates a more pronounced impact on enhancing the detection accuracy of small targets.

Table 2. The results of the attention mechanism ablation experiments.

Attention Mechanism	Holothurian (%)	Echinus (%)	Scallop (%)	Starfish (%)	mAP (%)
CBAM	77.3	92.7	83.1	88.3	85.3
CPCA	77.8	92.1	80.1	87.8	84.4
CA	77.5	92.3	83.1	88.5	85.3
GAM	77.3	85.6	79.8	88.6	82.8
SimAM	77.6	92.9	83.2	89.0	85.7

Through the ablation experiments, the impact of the different modules in the improved algorithm was examined, and the results are summarized in Table 3.

Table 3. The results of the ablation experiments.

Algorithm	Params (M)	FLOPs (G)	mAP (%)	FPS (frame/s)
YOLOv7	36.5	103.2	84.7	112.5
YOLOv7 + Kmeans	36.5	103.2	85.4	113.2
YOLOv7 + Kmeans + ELAN_PC	28.3	75.4	84.7	126.3
YOLOv7 + Kmeans + ELAN_PC + ShapeIoU_NWD	28.3	75.4	85.5	125.3
YOLOv7 + Kmeans + ELAN_PC + ShapeIoU_NWD + SimAM	28.7	76.6	85.7	122.9

As can be seen from Table 3, by adding the K-means clustering algorithm, the average accuracy is improved by 0.7%; by adding the ELAN_PC structure, although the accuracy value decreases by 0.7%, the number of parameters decreases by 22%, the amount of computation decreases by 27%, and the speed of detection is improved by 12%; and later on the ShapeIou_NWD loss function is utilized to increase the accuracy of detection to 85.5% by keeping the number of parameters and the amount of computation constant. Finally, the detection accuracy is improved to 85.7% using the SimAM attention mechanism. Therefore, compared with the original YOLOv7 algorithm, the whole improved algorithm increases the average accuracy value of underwater target detection by 1% and the detection speed by 9.2%.

4.2. Comparative Experiments

The improved YOLOv7 model was compared to these algorithms including Faster RCNN, SSD, YOLOv5, YOLOv7, YOLOv7-W6, YOLOv7-E6E and YOLOv8 algorithms on an underwater target dataset using the aforementioned evaluation metrics. The comparison results are presented in Table 4.

Table 4. Comparison of the improved algorithm with several mainstream algorithms.

Algorithm	Holothurian (%)	Echinus (%)	Scallop (%)	Starfish (%)	mAP (%)	FPS (frame/s)
Faster RCNN	56.0	82.0	52.0	79.0	67.5	40.7
SSD	63.0	77.0	46.0	74.0	65.0	167.9
YOLOv5	73.7	91.4	80.6	85.1	82.7	270.0
YOLOv7	76.5	92.1	82.8	87.4	84.7	112.5
YOLOv7-W6	74.2	90.8	75.2	87.1	81.8	114.2
YOLOv7-E6E	74.6	89.7	49.2	87.6	75.3	51.9
YOLOv8	74.9	91.4	80.3	88.3	83.7	131.9
Ours	77.6	92.9	83.2	89.0	85.7	122.9

From Table 4, we can see that our improved YOLOv7 algorithm has the highest detection accuracy in each class of underwater targets and the highest mAP value in all classes, which is 18.2% higher than the two-stage Faster RCNN algorithm, compared with the single-stage algorithms SSD, YOLOv5, YOLOv7, YOLOv7-W6, YOLOv7- E6E, and YOLOv8, and the mAP of the improved algorithm is improved by 20.7%, 3%, 1%, 3.9%,10.4%, and 2%, respectively. In addition, although the detection speed of the SSD algorithm, the YOLOv5 algorithm and the YOLOv8 algorithm is slightly higher than that of the improved YOLOv7 algorithm, the detection accuracy is much different from the improved algorithm in this paper, so the comprehensive comparison of this paper's algorithm has a great advantage over other algorithms.

From Table 5, we can see that the total accuracy of our improved algorithm is 38.8% higher compared to the two-stage Faster RCNN algorithm, and for each class of underwater targets, the accuracy value of our improved algorithm is higher. In addition, compared to other one-stage algorithms, other algorithms, such as YOLOv5, YOLOv7, YOLOv7-W6, and YOLOv7-E6E, are 1.1%, 1%, 3%, and 11.2% less accurate than ours, respectively, except for the SSD algorithm and the YOLOv8 algorithm, which are 1% and 0.6%, respectively, more accurate than our algorithm.

Table 5. Precision comparison results for different algorithms.

Algorithm	Holothurian (%)	Echinus (%)	Scallop (%)	Starfish (%)	All (%)
Faster RCNN	35.4	53.2	37.8	51.5	44.5
SSD	80.6	89.0	83.1	84.4	84.3
YOLOv5	76.5	88.3	83.4	80.5	82.2
YOLOv7	75.1	88.0	84.6	81.5	82.3
YOLOv7-W6	75.5	86.3	79.7	79.6	80.3
YOLOv7-E6E	79.9	88.6	34.8	85.2	72.1
YOLOv8	77.8	89.0	85.6	83.4	83.9
Ours	78.0	88.7	85.1	81.6	83.3

As can be seen from Table 6, the improved YOLOv7 algorithm has the largest value of Total Recall, improving 4% over the two-stage Faster RCNN algorithm and 32.9%, 1.9%, 0.6%, 3.2%, 6%, and 3.6% over the one-stage SSD, YOLOv5, YOLOv7, YOLOv7-W6, YOLOv7-E6E, and YOLOv8, respectively.

Table 6. Recall comparison results for different algorithms.

Algorithm	Holothurian (%)	Echinus (%)	Scallop (%)	Starfish (%)	All (%)
Faster RCNN	65.6	88.2	61.5	85.1	75.1
SSD	50.5	51.4	27.7	55.1	46.2
YOLOv5	70.1	86.0	73.3	79.5	77.2
YOLOv7	71.8	86.4	72.9	82.8	78.5

Table 6. *Cont.*

Algorithm	Holothurian (%)	Echinus (%)	Scallop (%)	Starfish (%)	All (%)
YOLOv7-W6	68.5	85.9	66.5	82.7	75.9
YOLOv7-E6E	66.3	82.4	64.1	79.6	73.1
YOLOv8	68.4	83.1	68.4	82.0	75.5
Ours	73.3	87.4	72.5	83.3	79.1

5. Conclusions

In this paper, an underwater target detection algorithm based on an improved YOLOv7 is proposed for underwater targets with some smaller sizes, complex backgrounds, and low contrast. By introducing the SimAM attention mechanism, the improved ELAN_PC structure and the ShapeIou_NWD loss function are used to improve the ability to extract features for smaller underwater targets and to reduce the loss of feature information. Meanwhile, in order to enhance the robustness of the algorithm in this paper, the K-Means anchor frame clustering algorithm is used to enhance the performance of the algorithm. Several comparison experiments and ablation experiments show that the improved YOLOv7 target detection algorithm achieves an average accuracy value of 85.7%, which is 1% higher than the original YOLOv7, and the speed reaches 122.9 frames/s, which is a good balance between detection speed and accuracy. Afterwards, the main feature extraction part of the model will be improved by using a lighter weight module to allow the algorithm to increase the inspection accuracy of underwater targets while being lighter weight.

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

Funding: This research was made possible by the Key Research and Development Program of Jiangsu Province (SBE2022010137).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Related improvements have been posted at https://github.com/hero259/yolov7-main-underwater (accessed on 1 June 2024).

Acknowledgments: Thanks to each of the authors for their dedication to the experiment and technical support.

Conflicts of Interest: The authors confirm that they have no conflicts of interest to report in relation to the present study.

References

1. Zhou, X.; Ding, W.; Jin, W. Microwave-assisted extraction of lipids, carotenoids, and other compounds from marine resources. In *Innovative and Emerging Technologies in the Bio-Marine Food Sector*; Academic Press: Cambridge, MA, USA, 2022; pp. 375–394.
2. Wang, S.; Liu, X.; Yu, S.; Zhu, X.; Chen, B.; Sun, X. Design and Implementation of SSS-Based AUV Autonomous Online Object Detection System. *Electronics* **2024**, *13*, 1064. [CrossRef]
3. Yu, G.; Cai, R.; Su, J.; Hou, M.; Deng, R. U-YOLOv7: A network for underwater organism detection. *Ecol. Inform.* **2023**, *75*, 102108. [CrossRef]
4. Jia, R.; Lv, B.; Chen, J.; Liu, H.; Cao, L.; Liu, M. Underwater Object Detection in Marine Ranching Based on Improved YOLOv8. *J. Mar. Sci. Eng.* **2023**, *12*, 55. [CrossRef]
5. Wu, B.; Liu, C.; Jiang, F.; Li, J.; Yang, Z. Dynamic identification and automatic counting of the number of passing fish species based on the improved DeepSORT algorithm. *Front. Environ. Sci.* **2023**, *11*, 1059217. [CrossRef]
6. Zhao, H.; Cui, H.; Qu, K.; Zhu, J.; Li, H.; Cui, Z.; Wu, Y. A fish appetite assessment method based on improved ByteTrack and spatiotemporal graph convolutional network. *Biosyst. Eng.* **2024**, *240*, 46–55. [CrossRef]
7. Liu, Y.; An, D.; Ren, Y.; Zhao, J.; Zhang, C.; Cheng, J.; Liu, J.; Wei, Y. DP-FishNet: Dual-path Pyramid Vision Transformer-based underwater fish detection network. *Expert Syst. Appl.* **2024**, *238*, 122018. [CrossRef]

8. Sahu, P.; Gupta, N.; Sharma, N. A survey on underwater image enhancement techniques. *Int. J. Comput. Appl.* **2014**, *87*, 333–338. [CrossRef]
9. Christensen, L.; de Gea Fernández, J.; Hildebrandt, M.; Koch, C.E.S.; Wehbe, B. Recent advances in ai for navigation and control of underwater robots. *Curr. Robot. Rep.* **2022**, *3*, 165–175. [CrossRef]
10. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
11. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
12. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
13. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
14. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788.
15. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Wei, X. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv* **2022**, arXiv:2209.02976.
16. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696.
17. Jocher, G. YOLOv8 by Ultralytics. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 15 February 2023).
18. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J.; Berkeley, U.C.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
19. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
20. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
21. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
22. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162.
23. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Proceedings Part I 14, Amsterdam, The Netherlands, 11–14 October 2016; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
24. Wen, G.; Li, S.; Liu, F.; Luo, X.; Er, M.-J.; Mahmud, M.; Wu, T. Yolov5s-ca: A modified yolov5s network with coordinate attention for underwater target detection. *Sensors* **2023**, *23*, 3367. [CrossRef] [PubMed]
25. Zhang, M.; Xu, S.; Song, W.; He, Q.; Wei, Q. Lightweight Underwater Object Detection Based on YOLO v4 and Multi-Scale Attentional Feature Fusion. *Remote Sens.* **2021**, *13*, 4706. [CrossRef]
26. Sinaga, K.P.; Yang, M. Unsupervised K-Means Clustering Algorithm. *IEEE Access* **2020**, *8*, 80716–80727. [CrossRef]
27. Arthur, D.; Vassilvitskii, S. K-means++: The advantages of careful seeding. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, LA, USA, 7–9 January 2007; pp. 1027–1035.
28. Zhang, X.; Zeng, H.; Guo, S.; Zhang, L. Efficient long-range attention network for image super-resolution. In Proceedings of the Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, 23–27 October 2022; Part XVII. Springer: Cham, Switzerland, 2022; pp. 649–667.
29. Lee, Y.; Hwang, J.W.; Lee, S.; Bae, Y.; Park, J. An Energy and GPU-Computation Efficient Backbone Network for Real-Time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Long Beach, CA, USA, 16–17 June 2019.
30. Wang, C.Y.; Mark Liao, H.Y.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of cnn. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2020), Washington, DC, USA, 14–19 June 2020; pp. 390–391.
31. Chen, J.; Kao, S.-H.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, Don't walk: Chasing higher FLOPS for faster neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
32. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation. *arXiv* **2020**, arXiv:2005.03572. [CrossRef] [PubMed]
33. Zhang, Y.-F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and Efficient IOU Loss for Accurate Bounding Box Regression. *arXiv* **2021**, arXiv:2101.08158. [CrossRef]
34. Zhang, H.; Zhang, S. Shape-IoU: More Accurate Metric considering Bounding Box Shape and Scale. *arXiv* **2023**, arXiv:2312.17663.

35. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2021**, arXiv:2110.13389.
36. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
37. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. *arXiv* **2018**, arXiv:1807.06521.
38. Yang, L.; Zhang, R.Y.; Li, L.; Xie, X. SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In Proceedings of the International Conference On Machine Learning (ICML), Virtual Event, 18–24 July 2021; pp. 11863–11974.

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