Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.047028

ARTICLE

An Underwater Target Detection Algorithm Based on Attention Mechanism and Improved YOLOv7

Liqiu Ren, Zhanying Li*, Xueyu He, Lingyan Kong and Yinghao Zhang

College of Information Science and Engineering, Dalian Polytechnic University, Dalian, 116034, China

*Corresponding Author: Zhanying Li. Email: lizy@dlpu.edu.cn

Received: 22 October 2023 Accepted: 22 December 2023 Published: 27 February 2024

ABSTRACT

For underwater robots in the process of performing target detection tasks, the color distortion and the uneven quality of underwater images lead to great difficulties in the feature extraction process of the model, which is prone to issues like error detection, omission detection, and poor accuracy. Therefore, this paper proposed the CER-YOLOv7(CBAM-EIOU-RepVGG-YOLOv7) underwater target detection algorithm. To improve the algorithm's capability to retain valid features from both spatial and channel perspectives during the feature extraction phase, we have added a Convolutional Block Attention Module (CBAM) to the backbone network. The Reparameterization Visual Geometry Group (RepVGG) module is inserted into the backbone to improve the training and inference capabilities. The Efficient Intersection over Union (EIoU) loss is also used as the localization loss function, which reduces the error detection rate and missed detection rate of the algorithm. The experimental results of the CER-YOLOv7 algorithm on the UPRC(Underwater Robot Prototype Competition) dataset show that the mAP(mean Average Precision) score of the algorithm is 86.1%, which is a 2.2% improvement compared to the YOLOv7. The feasibility and validity of the CER-YOLOv7 are proved through ablation and comparison experiments, and it is more suitable for underwater target detection.

KEYWORDS

Deep learning; underwater object detection; improved YOLOv7; attention mechanism

1 Introduction

The ocean occupies 70% of the earth's area and its marine products are abundant. Among them, sea cucumbers, sea urchins, and many other marine organisms have high nutritional value [1,2], and their popularity in the diet and medicine industries is increasing. Presently, holothurian and echinus are mainly caught by manual salvage, and there are many dangers in the course of fishing. To reduce the risk of manual fishing, underwater robots can be used for marine organism capture methods to achieve the purpose of protecting fishermen.

Underwater target detection allows underwater robots to autonomously discover and target the location of underwater organisms, thus reducing labor costs and achieving a higher degree of automation for underwater robots [3,4]. Due to the complex underwater environment, underwater images generally exhibit color distortion and image blurring, leading to difficulties in extracting underwater creature features, which poses a huge challenge for underwater object detection [5].

To solve the problems of difficult underwater image feature extraction and low accuracy of underwater target detection, we improved the YOLOv7 [6] algorithm and proposed the CER-YOLOv7 target detection algorithm. The attention mechanism Convolutional Block Attention Module (CBAM) [7] and structural Reparameterization Visual Geometry Group (RepVGG) [8] are introduced into the backbone of the YOLOv7, while Efficient Intersection over Union (EIoU) loss [9] is used as the localization loss.

The subsequent sections are organized as follows. Section 2 engages in an exploration of pertinent prior research in the field. In Section 3, a comprehensive exposition of the algorithm's enhancements is presented, including all key points of improvement. In Section 4, the metrics employed in experimentation are introduced, alongside a thorough presentation of the results of the experiment. In Section 5, the algorithms introduced in this paper are briefly summarized.

This paper will contribute to the following three aspects. Firstly, we augment the YOLOv7 network's backbone with the integration of both the CBAM block and RepVGG block, thereby bolstering the algorithm's capability to extract salient features. Secondly, we advocate for the replacement of the conventional Complete Intersection over Union (CIoU) loss [10] function with the Efficient Intersection over Union (EIoU) loss, resulting in enhanced accuracy across algorithmic predictions. Lastly, through systematic comparison with alternative underwater target detection algorithms, we verify the superior detection efficacy of the proposed CER-YOLOv7 algorithm by its excellent performance on the UPRC dataset.

2 Related Works

The research of underwater target detection can be divided into two main research directions. One is from the perspective of image pre-processing to enhance the image and make it clear. The other one is to improve the target detection algorithm.

Unlike the images in general target detection scenarios, underwater target images exhibit low image quality. Wang et al. [11] proposed an effective image enhancement method based on Retinex, which enhances the captured underwater images and enables the robot to have a clearer view of the underwater scene. Han et al. [12] combined the MAX-RGB method with the grayscale shading method for color correction of underwater images. Uplavikar et al. [13] proposed a domain adversarial learning-based underwater image enhancement algorithm that can effectively handle the diversity of images and transform them into clear images. Image enhancement methods can effectively enhance the quality of underwater images and increase the accuracy of underwater target detection. However, the detection speed tends to be slow after adding image enhancement methods. So, there are limitations in underwater scenarios with high real-time requirements.

Convolutional Neural Networks (CNN) based target detection algorithms have undergone rapid advancements because of the continuous research in deep learning [14] over the years. The algorithms can be categorized into two main branches: two-stage and single-stage. The two-stage algorithm first generates candidate regions in the initial stage and performs prediction in the next stage. Ren et al. [15] proposed a Faster Region-based Convolutional Network (Faster-RCNN), which uses Region Proposal Network (RPN) to generate region suggestions, followed by a classification and bounding box regression using CNN. Rather than extracting candidate regions independently, while the single-stage algorithm approaches the detection task as a regression problem and has slightly lower precision than the two-stage method, it compensates by offering faster detection speeds. Redmon et al. [16] proposed YOLO, which is the first single-stage target detection algorithm, while the YOLO series also contains: YOLO9000 [17], YOLOv3 [18], YOLOv4 [19], YOLOv5, YOLOx [20], YOLOv6, YOLOv7. Liu et al. [21] proposed a Single Shot Multi-Box Detector (SSD), which introduces multi-reference and multi-resolution techniques that greatly increase the detection accuracy of single-stage algorithms.

In the domain of underwater target detection, the inherent challenges posed by low image quality exacerbate the intricacies of feature extraction from images. To address this, Chen et al. [22] introduced a pioneering solution termed the Sample-Weighted Super Network (SWIPELENT), complemented by an innovative training paradigm named Curriculum Multi-Class Adaboost (CMA). This combined approach effectively mitigates the issues associated with blurred underwater images. Similarly, Song et al. [23] put forth a distinctive contribution involving a two-stage underwater detection algorithm, grounded in the Region Convolutional Network (RCNN) framework. The outcomes of this effort yielded an impressive accuracy level of 85.5%. However, it is imperative to underscore that while two-stage target detection algorithms exhibit high precision, they often contend with limitations in real-time processing capabilities, thereby imposing constraints on operational efficiency. To be faster in detection speed, Zhang et al. [24] used an attentional feature fusion module for underwater detection based on the YOLOv4 algorithm, a model that balances accuracy and real-time performance. Wang et al. [25] added the CBAM module to the backbone network of YOLOv5 to enhance the feature extraction capability. Qing et al. [26] proposed RepVGG-YOLO to improve feature extraction capability based on an improved RepVGG as a backbone extraction network. Therefore, we present an enhanced algorithm for the YOLOv7 backbone network based on the CBAM with the RepVGG.

The loss function is employed to measure the level of disparity between the predicted results and the ground truth, serving as an estimation of inconsistency. Whether the loss function is selected appropriately or not can affect the goodness of the algorithm model to a certain extent. Li et al. [27] replaced the Binary Cross Entropy (BCE) loss with Focal Loss as the localization loss function and adopted EIoU loss instead of Intersection over Union (IoU) loss as the confidence loss function within the YOLOx algorithm. These modifications effectively address the problem of positive and negative sample imbalance, resulting in improved convergence speed and enhanced detection accuracy for target frames. Yue et al. [28] proposed to utilize EIoU loss to overcome the challenge of detecting frame sample unbalance, which often leads to accuracy degradation and slow convergence. Li et al. [29] proposed to use EIoU loss as the localization loss function for the problem of possible non-convergence in training on fish and small target detection datasets when using Generalized Intersection over Union (GIoU) loss as the loss function. Therefore, better results are obtained by using the EIoU loss function as the localization loss function.

3 Methods

This section introduces the novel CER-YOLOv7 algorithm, which centers on the integration of an attention mechanism via the CBAM onto the original YOLOv7. This augmentation consists of systematically concatenating the channel attention and spatial attention mechanisms, sequentially used to train the YOLOv7. This approach culminates in two-dimensional channel weights, enhancing the algorithm's feature extraction capabilities.

A further innovation is introduced through the integration of the Structure Reparameterization RepVGG block into the model's backbone, ensuring collaboration between structures for seamless training and inferencing. This augmentation maintains real-time applicability while enhancing detection accuracy.

Another notable advancement pertains to the transition of the localization loss function to the EIoU loss. This addresses issues of proportional adjustments in width and height when predicting boxes, surmounting challenges posed by linear aspect ratios. This modification contributes to increased accuracy.

This section showcases the holistic enhancements within the CER-YOLOv7 algorithm. The introduction of CBAM, RepVGG block integration, and recalibrated localization loss collectively enhance feature extraction, detection accuracy, and real-time efficiency.

3.1 CER-YOLOv7

The main structure of the CER-YOLOv7 is shown in Fig. 1 and is divided into five main sections: Input, Backbone, Neck, Head, and Output. Our improved part is the red area in the figure.

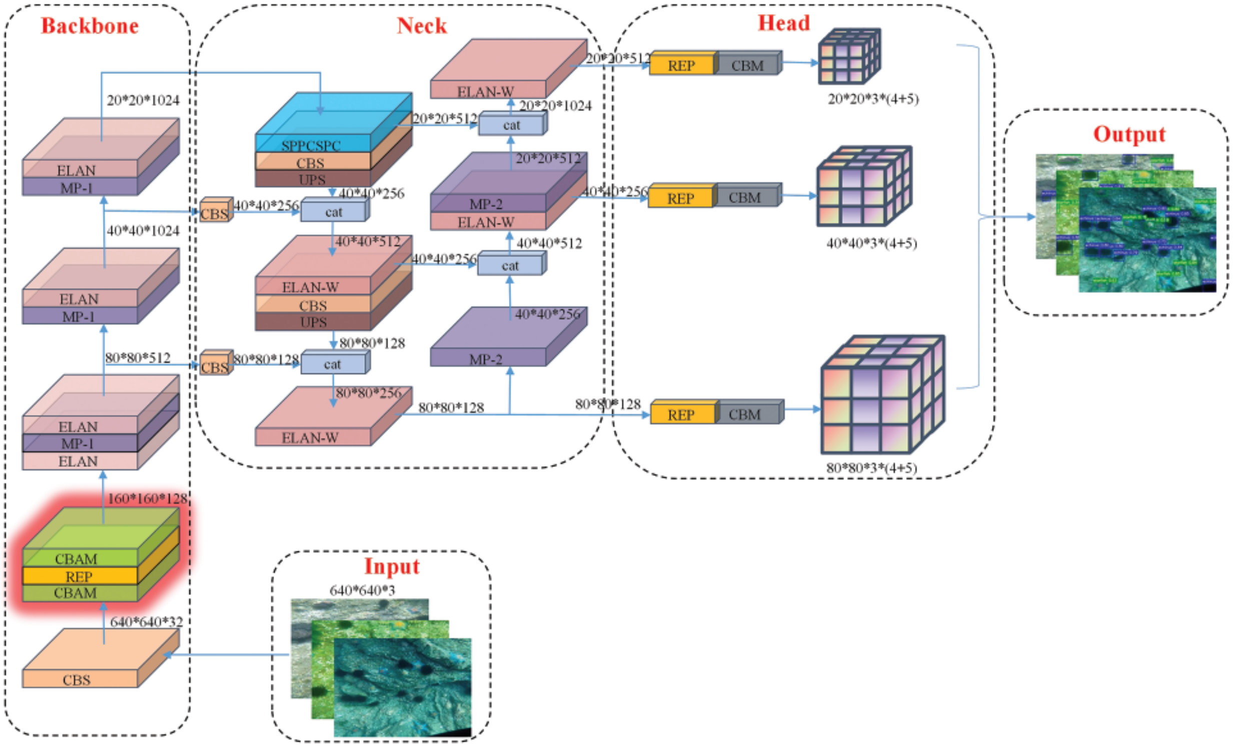


Figure 1: Overall structure of the CER-YOLOv7 algorithm

Before the input, we perform data enhancement and distortion-free affine transformation operations on all the experimental data to obtain the images with a size of 640 × 640 as the input to the network.

After the images are input into the network, firstly, they enter the Backbone module for feature extraction and output a two-fold down-sampled feature map through the CBS block. To better retain the effective features during downsampling, we insert the CBAM block here. And the RepVGG block is inserted next for better feature extraction.

After entering the Extended-Efficient Layer Aggregation Networks (E-ELAN) and the MP block to extract features alternately. The output multiscale features of the three E-ELAN blocks are finally used as the input of the Neck module. In the Neck module, the multiscale feature information is first input to the feature mapping of different sizes by the Spatial Pyramid Pooling Cross-Stage Partial (SPPCSPC) block, and then the multiscale feature information is fused to broaden the perceptual field, which expands the feature information to some extent. Then the output is fused by the PAFPN block which consists of a modified Feature Pyramid Network (FPN) [30] and Path Aggregation Network (PANet) [31] structure. Finally, the input image is sent to the Head section for output by integrating through 1 × 1 size of convolution after two different degrees and scales of feature extraction by Backbone and Neck modules.

3.2 CBAM Attention Mechanism

The attention mechanism first appeared in Recurrent Neural Networks (RNN) [32]. As the name means attention mechanism is to acquire more effective features and suppress the invalid features in the target scene. It is a plug-and-play module with a negligible parameter, thus enabling the model accuracy to be improved with almost no increase in parameters.

In the field of target detection, common attention modules include the channel attention Squeeze-and-Excitation (SE) [33], the CBAM, the Coordinate Attention (CA) [34], the Efficient Channel Attention (ECA) [35], and the non-reference attention Simple parameter-free Attention Module (SimAM) [36]. The structure of the CBAM module utilized in this paper is shown in Fig. 2, which incorporates the Channel Attention Module (CAM) and the Spatial Attention Module (SAM) and uses a serial approach to process the input feature layers through these two modules separately.

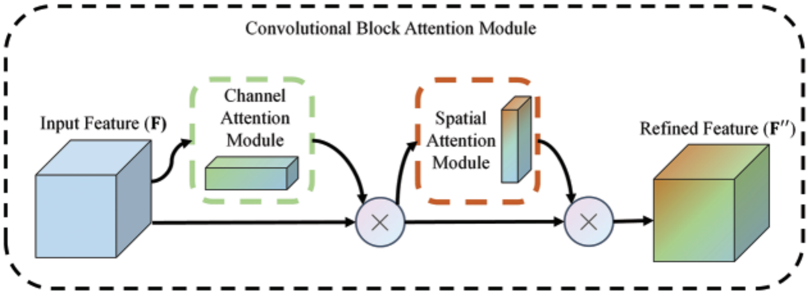


Figure 2: Overall structure of attention mechanism CBAM module

The overall formula of CBAM is shown in the following Eqs. (1) and (2):

$$F' = M_C(F) \otimes F \tag{1}$$

$$F'' = M_S(F') \otimes F' \tag{2}$$

The input feature map F is used as the first stage input. The one-dimensional attention channel map $M_C(F)$ is obtained by the channel attention mechanism and then F' is obtained by multiplying $M_C(F)$ with F. F' as the second-stage input feature map, and the two-dimensional spatial attention map $M_S(F')$ is obtained through the spatial attention mechanism, and then the final output F'' is obtained by multiplying $M_S(F')$ with F'.

The CAM and SAM modules are introduced separately in the following part.

3.2.1 CAM Module

The CAM module is more concerned with what input features are meaningful, and the overall structure is shown in Fig. 3a. After inputting the feature map F, the results from the global MaxPool and AvgPool channels respectively are sent to the fully connected layer (Shared MLP). The results from the fully connected layer are then summed and the Sigmoid activation function is used to obtain the channel attention feature map of the input layer $M_C(F)$ above. The CAM formula is described in the following Eq. (3), and W_0, W_1 denote the weights of the two fully connected layers of the MLP.

$$M_C(F) = \sum\left(MLP(AvgPool(F)) + MLP(Maxpool(F))\right) = \sum\left(W_1\left(W_0\left(F^c_{avg}\right)\right) + W_1\left(W_0\left(F^c_{\max}\right)\right)\right) \tag{3}$$

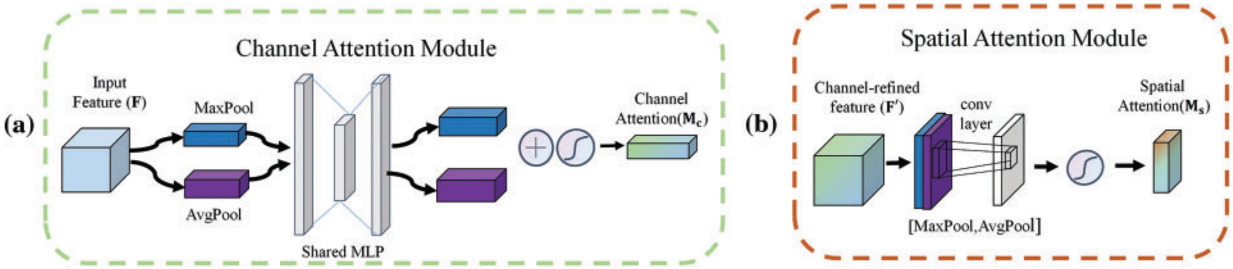


Figure 3: CBAM module detailed explanation

3.2.2 *SAM Module*

Unlike the CAM module, the SAM module is more concerned with what part of the input feature's information is more important. The overall structure is shown in the following Fig. 3b.

The input Channel-refined feature map F' through global MaxPool and AvgPool to get two $H \times W \times 1$ channels for concat operation, and then a size of 7 × 7 convolution reduces the dimension to 1 channel, that is, $H \times W \times 1$ channel. Finally, by Sigmoid activation function to get spatial attention feature map as $M_S(F')$ above. The SAM formula is described in the following Eq. (4), and $f^{(7\times7)}$ denotes a convolution kernel of size 7 × 7.

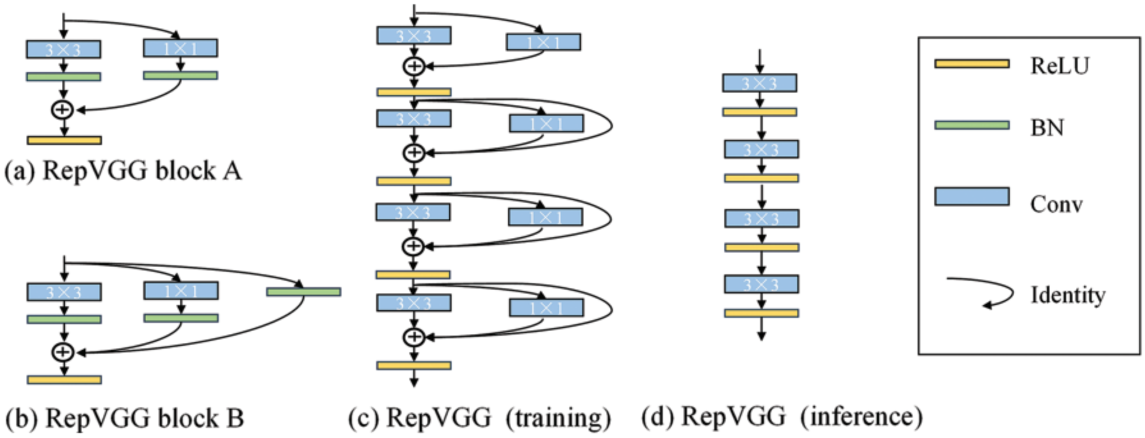
$$M_S(F) = \sum\left(f^{(7\times7)}\left(|AvgPool(F); MaxPool(F)\right)\right) = \sum\left(f^{(7\times7)}\left(\left|F^s_{avg}; F^s_{\max}\right|\right)\right) \tag{4}$$

3.3 *RepVGG*

Addressing overfitting gradient disappearance and the explosion of convolutional neural networks in underwater target detection, we adopted the RepVGG block to improve the model. RepVGG is a simple but powerful convolutional neural network, that incorporates ideas from VGG [37] with ResNet [38]. During training, the network is carried out in a multi-branch topology, and during the inference, the parameters of the branches are re-parameterized and combined into a single branch for faster and more accuracy. The RepVGG network structure and structural re-parameterization are described separately below.

3.3.1 *RepVGG Block*

The structure of RepVGG is shown in Fig. 4. The multi-branch topology of ResNet is inherited and improved in the training network, and the parallel multi-branch can increase the feature capability of the model and make detection accuracy higher. Two residual structures RepVGG block A and B are used for the training network. Block A contains a size of 3 × 3, a size of 1 × 1 convolution kernel, and a Rectified Linear Unit (ReLU) activation function. Block B contains a size of 3 × 3, 1 × 1 convolution kernel, an identity branch, and a ReLU. In the inference, we refer to the VGG structure by transforming the multi-branch into a single-branch through structural reparameterization, and the whole structure is composed of a 3 × 3 size convolutional kernel and ReLU activation function in series to make inference faster.



(a) RepVGG block A

(b) RepVGG block B

(c) RepVGG (training)

(d) RepVGG (inference)

Figure 4: Part structure of the RepVGG network

3.3.2 Structural Reparameterization

The structural reparameterization of RepVGG is the process of transforming the RepVGG training into RepVGG inference in the above Fig. 4, and the whole process is divided into the following 4 steps: (1) Combine Conv and Batch Normalization (BN); (2) Transform the convolution kernel size from 1 × 1 to 3 × 3; (3) Transform BN to 3 × 3 convolution; (4) Fuse multiple 3 × 3 convolution branches into one branch. The overall steps of structural reparameterization are shown in Fig. 5 below. The process of Conv and BN in step 1 is described by the following Eqs. (5) and (6):

$$Conv\,(x) = W\,(x) + b \tag{5}$$

$$BN\,(x) = \gamma \frac{(X - \mu)}{\sqrt{\sigma^2 + \varepsilon}} + \beta \tag{6}$$

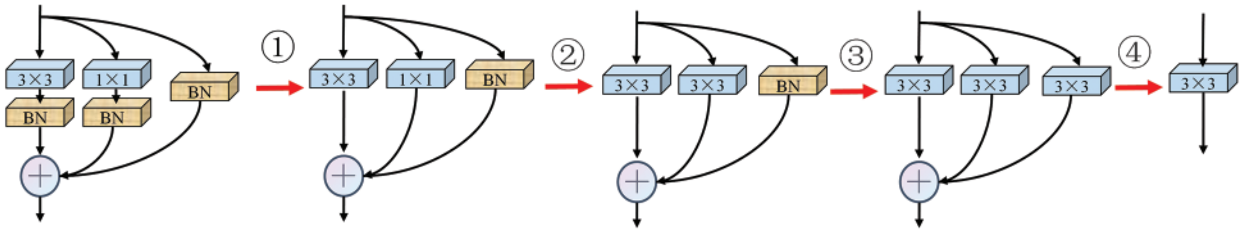


Figure 5: Sketch of structural reparameterization steps

Step 1 is represented in the following Eq. (7):

$$BN\,(conv\,(x)) = \gamma \frac{(W\,(x) + b - \mu)}{\sqrt{\sigma^2 + \varepsilon}} + \beta = \left(\frac{\gamma\, W\,(x)}{\sqrt{\sigma^2 + \varepsilon}}\right) + \left(\frac{\gamma\, \mu'}{\sqrt{\sigma^2 + \varepsilon}} + \beta\right) \tag{7}$$

In the above three equations γ is the weight factor, $W(x)$ is the convolution kernel operation, b is the convolution kernel bias, μ is the channel mean, $\mu' = b - \mu$ is the cumulative mean, σ is the variance, ε is a very small constant that prevents the denominator from being zero, and β is the BN layer bias. The new weight and deviation formulas are generated after passing step 1 as follows Eqs. (8)–(10):

$$W_i' = \frac{\gamma_i W_i}{\sigma_i} \tag{8}$$

$$B_i' = -\frac{\gamma_i \mu_i'}{\sigma_i} + \beta_i \tag{9}$$

$$BN(conv(x)) = W_i'(x) + B_i' \tag{10}$$

Step 2 only needs to pad in the original 1 × 1 × 3 convolution, the padded part convolution kernel parameters set to 0. Notably, the padding of the input feature map needs to be set to 1 to ensure that the width and height of the input and output feature maps are consistent. The difference between step 3 and step 2 is that the BN branch does not have a convolution layer, so a 3 × 3 convolution layer with an identity mapping needs to be constructed. In other words, the BN layer can be fused with the constructed 3 × 3 convolution layer with constant input and output. Step 4 is to sum the parameters of the 3 convolution layers to obtain the final 3 × 3 convolution.

3.4 EIoU Loss

CIoU as the loss function in the YOLOv7 algorithm with the following Eq. (11), in which we find that the CIoU adds the aspect ratio of the bounding box as a penalty term to the bounding box loss function. It can accelerate the convergence to a certain extent, but the aspect ratio is a relative value and there is uncertainty, which easily causes errors and inaccurate experimental results. To solve the above problem, we use the EIoU loss function, which is based on CIoU loss, and split the aspect ratio in Eq. (13) to calculate the width and height separately. gradient vanishing. To address the issue of gradient vanishing, a BN (Batch Normalization) layer is employed. The BN layer normalizes the input data by subtracting the mean and dividing it by the variance on a per-layer basis. This normalization process ensures a more uniform distribution of input data, thereby reducing the occurrence of gradient disappearance. By standardizing the output of each layer to have consistent mean and variance, the BN layer eliminates the influence of weight scaling, which resolves both gradient vanishing and explosion problems [39].

$$L_{CIoU} = 1 - IoU + \left(\frac{\rho^2(b, b^{gt})}{c^2} + \alpha\upsilon\right) \tag{11}$$

$$\alpha = \frac{\upsilon}{(1 - IoU) + \upsilon} \tag{12}$$

$$\upsilon = \frac{4}{\pi^2}\left(\arctan\frac{w_{gt}}{h_{gt}} - \arctan\left(\frac{w}{h}\right)\right)^2 \tag{13}$$

The EIoU obtained after improved CIoU is shown in the following Eq. (14):

$$L_{EIoU} = L_{IoU} + L_{dis} + L_{asp} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \frac{\rho^2(w, w^{gt})}{c_w^2} + \frac{\rho^2(h, h^{gt})}{c_h^2} \tag{14}$$

In the above equation, c_w and c_h are the width and height of the smallest external rectangle obtained by merging predicted box and ground truth, and ρ represents the Euclidean distance between the two points. In Eq. (14), the EIoU contains four components: L_{IoU} overlap loss, L_{dis} distance loss, width, and height loss of L_{asp}. The initial two components of the EIoU follow a similar approach to the CIoU. The height-width loss aims to directly minimize the disparity between the predicted box's height and width compared to the ground truth, making our model have faster convergence speed and better localization results in our experiments.

4 Experiments and Results

In this section, the experimental environment, parameters, dataset, and metrics are provided. Ablation experiments are performed to evaluate and compare the performance of YOLOv7 and its improved algorithms. The CER-YOLOv7 is compared with the mainstream detection algorithms. The detection results of each algorithm are evaluated by a combination of quantitative and qualitative analyses.

4.1 Experimental Environment and Parameters

In our network, CBAM and RepVGG block and EIoU loss functions are used to improve the YOLOv7 network. All experiments are implemented on a desktop machine equipped with an Intel Core i7-12700F CPU and 32.0-GB RAM, and NVIDIA GeForce RTX3080 GPU (10-GB memory) for acceleration.

To control the experimental variables and ensure experiment fairness, identical parameters are set for each group of experiments. The resolution of the input images resized to 640 × 640, the optimizer momentum is 0.937, the initial learning rate is 0.001, and the weight decay coefficient is 0.0005. Adam was selected as a gradient descent optimizer for updating the convolution kernel parameters. Considering the parameters and Floating Point of Operations (Flops) are different for each model, we set the batch size to 8 through preliminary experiments on each model to take into account that all models do not exceed the GPU memory. For better analysis of the training process and results, the epoch was set to 300. All other parameters were in keeping with the YOLOv7 network.

4.2 Dataset Process

In the experiments, the China Underwater Robot Professional Contest (URPC) dataset was used. There are 5 categories in the original dataset: holothurian, echinus, scallop, starfish, and seaweed. Seaweed is a disturbance item and a total of 82 images exist in the dataset. We picked 4000 images from the original 5543 images randomly and deleted the category of seaweed. The ratio of the training and test set was divided into 7:3, corresponding to 2800 training images and 1200 test images, respectively. After sampling, we counted the category information, category ratio, target location distribution, and target size again, so that the data distribution of the training and test set can be relatively matched to achieve the purpose of division.

To meet the requirements needed for the experiment, all sample images were preprocessed following the VOC2007 sample format. In this process, we found that some images in this dataset have problems such as chromatic aberration, low contrast, occlusion, and dense targets. There are differences in image sizes and the number of species, which cause difficulties in the process of training the model.

4.3 Evaluation Metrics

In the field of underwater detection, a set of metrics finds purpose in quantifying the efficacy of algorithms. Within the CER-YOLOv7 algorithm, a comprehensive framework of evaluation is established, comprising seven distinctive metrics.

Precision (P), denoting the proportion of predicted positive instances accurately aligned with actual positives, the definition of P is defined as follows Eq. (15):

$$P\,(Precision) = \frac{TP}{TP + FP} \tag{15}$$

Recall (R), signifying the ratio of true positive predictions to the entirety of true positive samples, the equation for R is defined as follows Eq. (16):

$$R\,(Recall) = \frac{TP}{TP + FN} \tag{16}$$

In the above equation, True Positive (TP) denotes the count of positive samples correctly classified as positive; False Positive (FP) signifies the count of negative samples erroneously classified as positive; and False Negative (FN) reflects the count of positive samples incorrectly classified as negative.

Average Precision (AP), capturing the area enclosed beneath the Precision-Recall (PR) curve, the definition of AP is defined as follows Eq. (17):

$$AP = \int_0^1 P\,(r)\,dr \tag{17}$$

Meanwhile, mean Average Precision (mAP) means the averaging of AP for each category. The mAP takes values in the interval [0,1], and the higher the value, the more accurate the model. The formula for mAP is simplified as follows Eq. (18):

$$mAP = \frac{1}{N}\sum_{n=1}^{N} AP_n \tag{18}$$

The metrics of Frame Per Second (FPS) and Giga Floating-point Operations Per Second (GFLOPS) serve as significant determinants for real-time applicability, where FPS quantifies image processing speed and GFLOPS measures the computational intricacy of the model. It is important to note that computational complexity does not necessarily equate to operational speed. Lastly, the parameters metric pertains to the cumulative numerical count of model parameters, furnishing an intuitive means of measuring model size.

In summation, this array of metrics provides a multidimensional evaluative lens for the CER-YOLOv7 network's performance in the domain of underwater target detection, affording insights into diverse aspects of its functionality and effectiveness.

4.4 Ablation Experiments

The CBAM module, the RepVGG module, and the EIoU loss function injected into the model are described according to the previous comprehensive description in Section 3. To substantiate the enhanced efficacy of the augmented YOLOv7 algorithm and discern the apparent contributions of each improved module within the architecture, a systematic manipulation of variables was processed.

The ensuing experimental analysis entailed a quantification of results, wherein the mAP and FPS values were meticulously measured across diverse model configurations.

To effectuate a rigorous and discriminating comparative assessment, eight distinct experiments were meticulously contrived within the framework of this ablation experiment. Each experiment configuration was intricately designed to offer incremental insights into the additive impact of the augmentations. Specifically, Exp.1 elucidates the foundational YOLOv7 model. Thereafter, a progressively augmentative approach was undertaken, successively introducing the CBAM block (Exp.2), RepVGG block (Exp.3), EIoU loss function (Exp.4), a combination of CBAM and RepVGG blocks (Exp.5), CBAM block and EIoU loss function (Exp.6), RepVGG block and EIoU loss function (Exp.7). Eventually, it was fully improved in Exp.8.

Table 1 provides a meticulous table of the experimental data and results, and a comprehensive and clear presentation of the main conclusions of the ablation experiments. Comparing Exp.1 with Exp.2, the mAP score improvement is after introducing the CBAM module, which indicates that the attention mechanism module can effectively suppress useless features after inserting the YOLOv7 network. The combination of the spatial and channel attention mechanisms allows the model to focus more on meaningful information about the target feature, leading to a notable enhancement in accuracy. Yet, this improvement comes at the cost of increased model complexity, resulting in a reduction in inference speed. In Exp.3, the mAP score is improved after introducing the RepVGG module. Through the single-way architecture of RepVGG and the 3 × 3 convolution kernel, the model enables the inference speed close to the original model while increasing the model complexity, and improving the model accuracy. According to Exp.4, replacing the CIoU loss function with EIoU increases the mAP score, leading to accelerated convergence of the model and enhanced accuracy in regression. With the same parameters and complexity, the model inference speed and accuracy are improved. Comparing Exp.1 with Exp.5, Exp.6, and Exp.7, we did ablation experiments and showed different degrees of improvement in mAP in both two combinations.

Table 1: Results of the ablation study algorithm

No.	Method			AP (%)	AP (%)	AP (%)	AP (%)	mAP (%)	FPS
	CBAM	RepVGG	EIoU	Holothurian	Echinus	Starfish	Scallop		
Exp.1				75.5	88.6	87.5	84.2	83.9	102
Exp.2	✓			78.6	88.5	87	85.1	84.8	87
Exp.3		✓		77.3	88.6	87	85.3	84.5	101.2
Exp.4			✓	78	89.4	86.9	84.6	84.7	**103.9**
Exp.5	✓	✓		78.1	89.7	**88.6**	85.5	85.5	85.5
Exp.6	✓		✓	78.3	88.5	86.9	85.2	84.8	88.2
Exp.7		✓	✓	78.1	89.3	87.3	85.5	85.1	101.8
Exp.8	✓	✓	✓	**79.2**	**90.8**	88.5	**86.1**	**86.1**	88.5

Comparing Exp.1 with Exp.8, there is an improvement in the mAP score and the speed of inferencing is slightly decreased. Indicating that the YOLOv7 network with CBAM block, RepVGG block, and the replacement of EIoU loss function has a slight decrease in inference speed, but a large improvement in model accuracy. By comparing with the YOLOv7 network, our model has superior overall performance. According to the data in the table, we can conclude that different categories of AP and mAP values are all improved, which can be concluded that our model performed much better than YOLOv7 on the UPCR dataset. Meanwhile, we compare the data according to Fig. 6, where the horizontal axis indicates the experiment serial number, the main vertical axis indicates mAP (%), and the secondary vertical axis indicates FPS. In the figure, we can conclude that when adding the CBAM module, there is a significant reduction in the inference speed compared to all other algorithms. However, there is no obvious influence of the RepVGG module and the EIoU loss function on the inference speed. All three modules have different degrees of improvement for mAP values, so we use a combination of all three for our final algorithm.

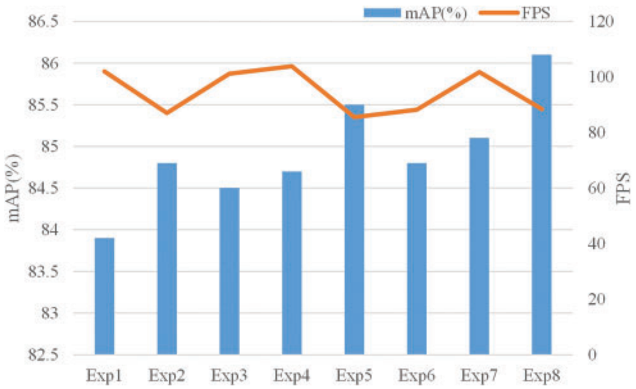


Figure 6: Results of the ablation study algorithm

4.5 *Qualitative Analysis*

To analyze the detection results of each algorithm more intuitively, we detected the targets with SSD, Faster-RCNN, YOLOv5-s, YOLOx-s, YOLOv6-s, YOLOv7, and YOLOv8-s, respectively. To imitate the underwater detection environment under real light sources, we used the test dataset divided into daylight, blue light, and green light images for testing. For the camera position irregularity during the detection, we detect in the front and side, respectively, and detect whether there is an obscuring effect in the dense state of targets. The detection effect is shown in the following Fig. 7. The red detection box indicates holothurian, the blue detection box indicates echinus, the yellow detection box indicates starfish and the purple detection box indicates scallop.

According to the feedback in the figure, each algorithm has different degrees of error and omission in detection, among which SSD, Faster R-CNN, and YOLOx-s perform slightly worse than other target detection algorithms. The low error and omission rates of CER-YOLOv7 make it stand out from the crowd of target detection models.

To better analyze the performance of CER-YOLOv7, we compared it to the test results of YOLOv7. Fig. 8a shows the detection results under different light conditions with similar target sizes. In the top left corner of the daylight image, YOLOv7 has a missed detection phenomenon. Misdetections of the scallop target are present in both images in the green light image, but the number of misdetections was lower in CER-YOLOv7. There is no obvious difference between the two images in the blue light image. Fig. 8b represents the detection results under different light for different target sizes. YOLOv7 identifies the clumped seaweed as an echinus in the top-left corner of the daylight images and exhibits a higher rate of false detection for small targets than CER-YOLOv7. There are false and missed detections of scallop targets in YOLOv7 in green light images. In the blue light image, the two images are not different. Fig. 9 indicates the accuracy comparison of the two networks when the targets are easier to identify, and it is evident that CER-YOLOv7 has improved detection accuracy for most targets.

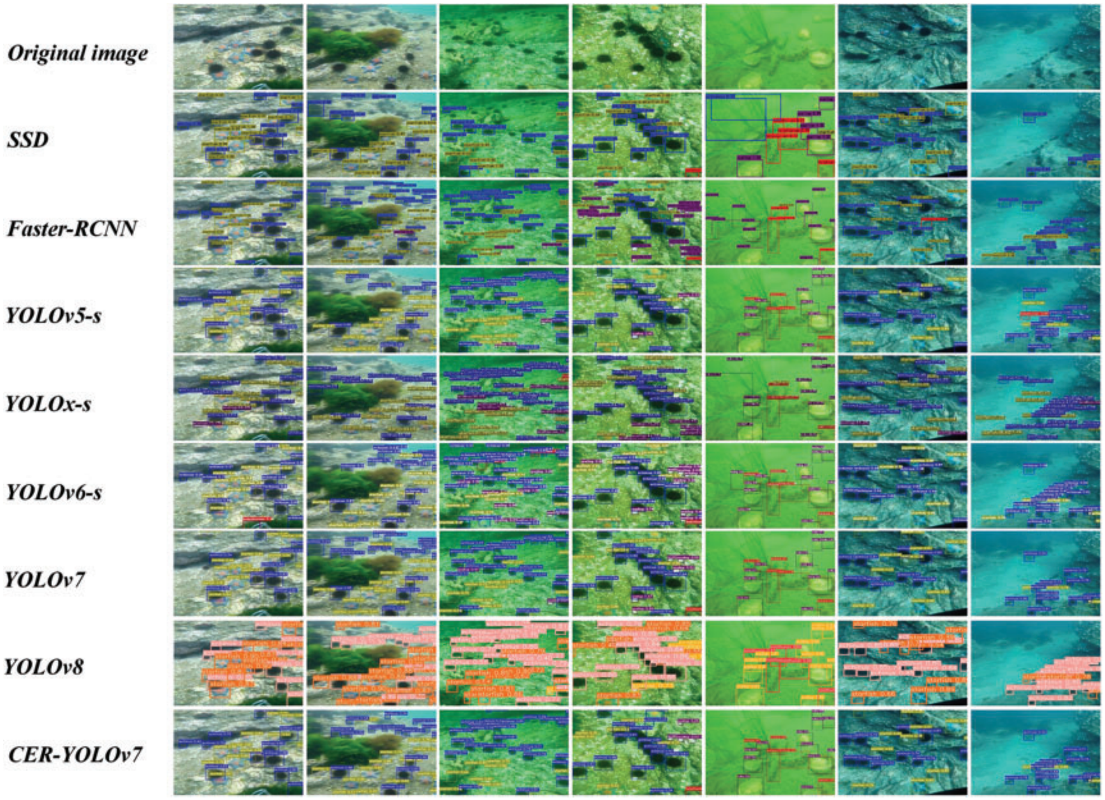


Figure 7: Detection images of different algorithms and original images

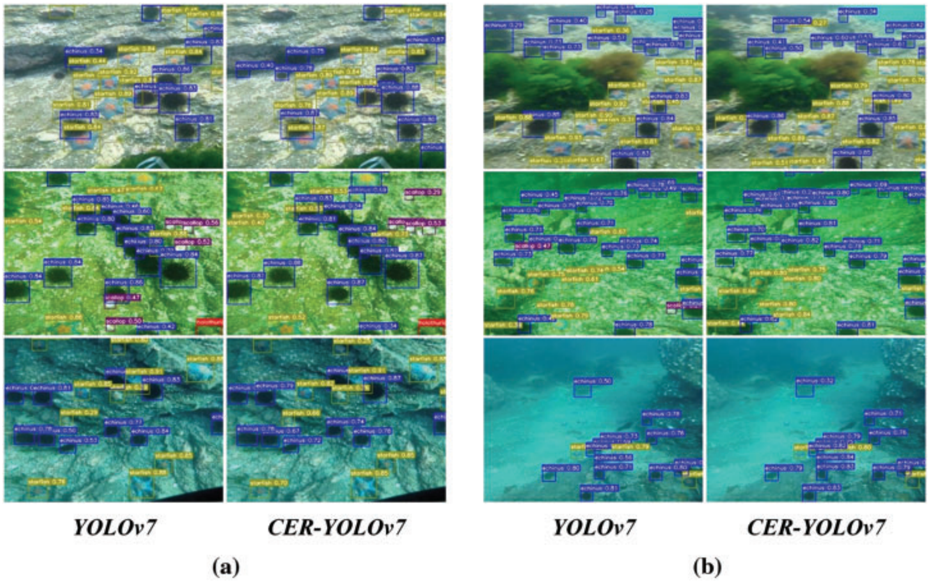
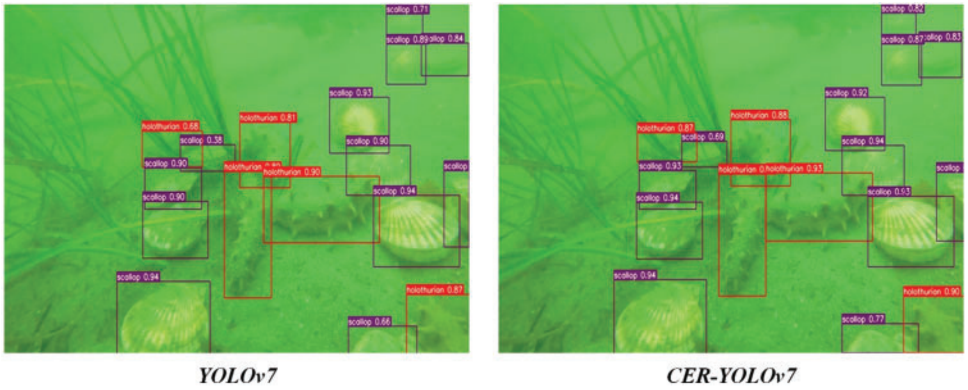


Figure 8: Comparison images of detection results. (a) Detection results with similar target size (b) detection results with different target sizes



YOLOv7 *CER-YOLOv7*

Figure 9: Comparison chart of detection accuracy

4.6 Quantitative Analysis

Meanwhile, we compared CER-YOLOv7 with the current popular target detection algorithms such as SSD, Faster-RCNN, YOLOv5-s, YOLOx-s, YOLOv6-s, YOLOv7 and YOLOv8-s in our experiments. Experiment results are recorded in Table 2. All experiments use the same configuration, parameter settings, data sets, and evaluation metrics.

Table 2: Detection results of different algorithms

Model	Map (%)	FPS	Params (M)	Flops (G)
SSD	54.4	119.4	26.29	62.75
Faster-RCNN	73.2	22.7	28.48	941.17
YOLOx-s	81.9	62.9	8.94	26.76
YOLOv5-s	80.8	**147.1**	**6.74**	**16.5**
YOLOv6-s	82.1	84.5	18.5	45.3
YOLOv7	83.9	102	35.49	105.2
YOLOv8-s	83.7	82.6	11.2	28.6
CER-YOLOv7	**86.1**	88.5	35.5	106.2

The addition of CBAM and RepVGG modules significantly enhances the ability to extract features, while preserving a higher number of effective features. Furthermore, we have optimized the loss function. The EIoU loss function re-calculates the localization function to enable more accurate detection of target locations. This improvement has greatly enhanced our target detection accuracy. However, it is worth noting that while these refinements have indeed improved performance, also increased the model's complexity and parameters due to the inclusion of the CBAM and RepVGG modules. This has directly resulted in a decrease in inference speed.

The above systematic evaluation of the CER-YOLOv7 leads to the following conclusions: (a) compared with the YOLOv7, the precision is raised by 2.2%, showing that the algorithm in underwater target detection accuracy is higher than other mainstream algorithms; (b) the algorithm's capability to ensure consistent performance, even under fluctuating conditions in terms of both the volume and status of FPS; and (c) the significant improvement in the mAP value proves that the algorithm can predict accurately in different scenarios.

When epoch = 300, CER-YOLOv7 is much better than traditional SSD and Faster-RCNN in terms of accuracy, real-time, parameters, and complexity. Therefore, we can conclude that the CER-YOLOv7 demonstrates superior performance in complex underwater scenarios. Compared to other mainstream algorithms, CER-YOLOv7 boasts higher accuracy while maintaining moderate speed. Additionally, the issue of decreased detection speed guides our future research direction, which is to optimize the algorithm through lightweight methods while maintaining model accuracy, thereby reducing model complexity and parameters, and ultimately enhancing detection speed.

5 Conclusion

For the problem of difficulty in extracting effective features from underwater images and the localization error of the loss function, we proposed the CER-YOLOv7 target detection network in this paper. The following improvements are made to the original YOLOv7 network.

First, the CBAM module is inserted in the Backbone part to make the down-sampled image retain more effective features from both channel and space dimensions. Second, the RepVGG block is inserted after the CBAM module, thus greatly enhancing the feature extraction capability. The capability of training and inference of the network is also improved, so that the accuracy is improved when the algorithm maintains the same detection speed. Last, replacing the original localization loss function of the YOLOv7 network with the EIoU loss function effectively solves the error generated by the original localization loss function in calculating the aspect ratio, so that the algorithm can better localize the target, thus reducing the false and missed detection rate. We have performed ablation studies on the improved part of the CER-YOLOv7 and compared the detection results of different networks. In the experimental results, the mAP score is 86.1%, which is a 31.7% and 2.2% increase over SSD and YOLOv7, respectively. Then we concluded that the detection accuracy of CER-YOLOv7 in the complex underwater environment has been greatly improved. However, there is still much room for improvement in terms of model complexity and number of parameters, which also guides the direction for our next work.

Since underwater images are more difficult to obtain and limited in number and species of underwater organisms in shallow water areas. We have only experimented on one dataset and the underwater images are distorted in color and poor in quality. So, in future work, we will collect datasets extensively for experiments, perform data augmentation techniques on underwater images, and continue researching in the direction of real-time detection as well as model lightweight.

Acknowledgement: We would like to express our heartfelt appreciation to the Scientific Research Fund of Liaoning Provincial Department of Education for providing the necessary financial support to carry out this research project.

Funding Statement: Scientific Research Fund of Liaoning Provincial Education Department (No. JGLX2021030): Research on Vision-Based Intelligent Perception Technology for the Survival of Benthic Organisms.

Author Contributions: Methodology: Liqiu Ren, Zhanying Li; data collection: Xueyu He, Yinghao Zhang; analysis of results: Liqiu Ren, Zhanying Li; draft preparation: Liqiu Ren, Lingyan Kong, Xueyu He. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data will be available on request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Bordbar, F. Anwar, and N. Saari, "High-value components and bioactives from sea cucumbers for functional foods—A review," *Mar. Drugs*, vol. 9, no. 10, pp. 1761–1805, 2011. doi: 10.3390/md9101761.

[2] A. Sibiya, J. Jeyavani, J. Sivakamavalli, C. Ravi, M. Divya and B. Vaseeharan, "Bioactive compounds from various types of sea urchin and their therapeutic effects—A review," *Reg. Stud. Mar. Sci.*, vol. 44, pp. 101760, 2021.

[3] D. Lee, G. Kim, D. Kim, H. Myung, and H. T. Choi, "Vision-based object detection and tracking for autonomous navigation of underwater robots," *Ocean Eng.*, vol. 48, pp. 59–68, 2012. doi: 10.1016/j.oceaneng.2012.04.006.

[4] H. Huang *et al.*, "A review on underwater autonomous environmental perception and target grasp, the challenge of robotic organism capture," *Ocean Eng.*, vol. 195, pp. 106644, 2020. doi: 10.1016/j.oceaneng.2019.106644.

[5] M. J. Er, J. Chen, Y. N. Zhang, and W. X. Gao, "Research challenges, recent advances, and popular datasets in deep learning-based underwater marine object detection: A review," *Sens.*, vol. 23, no. 4, pp. 1990, 2023. doi: 10.3390/s23041990.

[6] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, 2023, pp. 7464–7475.

[7] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proc. ECCV*, Munich, Germany, 2018, pp. 3–19.

[8] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding and J. Sun, "RepVGG: Making VGG-style convnets great again," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, 2021, pp. 13733–13742.

[9] F. Y. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang and T. Tan, "Focal and efficient iou loss for accurate bounding box regression," *Neurocomput.*, vol. 506, pp. 146–157, 2022. doi: 10.1016/j.neucom.2022.07.042.

[10] Z. Zheng *et al.*, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," *IEEE Trans. Cybern.*, vol. 52, no. 8, pp. 8574–8586, 2021. doi: 10.1109/TCYB.2021.3095305.

[11] Y. Wang *et al.*, "Real-time underwater onboard vision sensing system for robotic gripping," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2020. doi: 10.1109/TIM.2021.3123218.

[12] F. Han, J. Yao, H. Zhu, and C. Wang, "Underwater image processing and object detection based on deep CNN method," *J. Sens.*, vol. 2020, pp. 1–20, 2020. doi: 10.1155/2020/6707328.

[13] P. M. Uplavikar, Z. Wu, and Z. Wang, "All-in-one underwater image enhancement using domain-adversarial learning," in *Proc. of the 2019 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 1–8.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. doi: 10.1038/nature14539.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Adv. Neural Inf. Process. Syst.*, vol. 28, pp. 1137–1149, 2015.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788.

[17] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 6517–6525.

[18] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv:1804.02767, 2018.

[19] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv:2004.10934, 2020.

[20] G. Zheng, S. T. Liu, F. Wang, Z. M. Li and J. Sun, "YOLOX: Exceeding YOLO series in 2021," arXiv:2107.08430, 2021.

[21] W. Liu *et al.*, "SSD: Single shot multibox detector," in *2016 Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, Netherlands, 2016, pp. 21–37.

[22] L. Chen *et al.*, "SWIPENET: Object detection in noisy underwater scenes," *Pattern Recognit.*, vol. 132, pp. 108926, 2022. doi: 10.1016/j.patcog.2022.108926.

[23] P. Song, P. Li, L. Dai, T. Wang, and Z. Chen, "Boosting R-CNN: Reweighting R-CNN samples by RPN's error for underwater object detection," *Neurocomput.*, vol. 530, pp. 150–164, 2023. doi: 10.1016/j.neucom.2023.01.088.

[24] M. Zhang, S. Xu, W. Song, Q. He, and Q. Wei, "Lightweight underwater object detection based on YOLO v4 and multi-scale attentional feature fusion," *Remote Sens.*, vol. 13, no. 22, pp. 4706, 2021. doi: 10.3390/rs13224706.

[25] Q. F. Wang, M. Cheng, S. Huang, Z. J. Cai, J. L. Zhang and H. B. Yuan, "A deep learning approach incorporating YOLO v5 and attention mechanisms for field real-time detection of the invasive weed Solanum rostratum Dunal seedlings," *Comput. Electron. Agric.*, vol. 199, pp. 107194, 2022. doi: 10.1016/j.compag.2022.107194.

[26] Y. Qing, W. Liu, L. Feng, and W. Gao, "Improved YOLO network for free-angle remote sensing target detection," *Remote Sens.*, vol. 12, no. 11, pp. 2171, 2021. doi: 10.3390/rs13112171.

[27] D. Li, Z. Zhang, B. Wang, C. Yang, and L. Deng, "Detection method of timber defects based on target detection algorithm," *Meas.*, vol. 203, no. 7, pp. 111937, 2022. doi: 10.1016/j.measurement.2022.111937.

[28] L. Yue, H. Ling, J. Yuan, and L. Bai, "A lightweight border patrol object detection network for edge devices," *Electron.*, vol. 11, no. 22, pp. 3828, 2022. doi: 10.3390/electronics11223828.

[29] J. Li, C. Liu, X. Lu, and B. Wu, "CME-YOLOv5: An efficient object detection network for densely spaced fish and small targets," *Water*, vol. 14, no. 15, pp. 2412, 2022. doi: 10.3390/w14152412.

[30] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 2117–2125.

[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 8759–8768.

[32] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," arXiv:1409.2329, 2014.

[33] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 7132–7141.

[34] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, 2021, pp. 13713–13722.

[35] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, 2020, pp. 11534–11542.

[36] L. Yang, R. Y. Zhang, L. Li, and X. Xie, "SimAM: A simple, parameter-free attention module for convolutional neural networks," in *Int. Conf. Mach. Learn. (PMLR)*, 2021, pp. 11863–11874.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556, 2014.

[38] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778.

[39] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. Mach. Learn. (PMLR)*, 2015, pp. 448–456.