frontiers | Frontiers in Marine Science

TYPE Original Research
PUBLISHED 23 March 2023
DOI 10.3389/fmars.2023.1153416

OPEN ACCESS

EDITED BY
Xuemin Cheng,
Tsinghua University, China

REVIEWED BY
Peng Ren,
China University of Petroleum (East China), China
Qiqi Zhu,
China University of Geosciences Wuhan, China

*CORRESPONDENCE
Heng Li
12309119@kust.edu.cn

SPECIALTY SECTION
This article was submitted to Ocean Observation, a section of the journal Frontiers in Marine Science

RECEIVED 29 January 2023
ACCEPTED 13 March 2023
PUBLISHED 23 March 2023

CITATION
Zhang C, Zhang G, Li H, Liu H, Tan J and Xue X (2023) Underwater target detection algorithm based on improved YOLOv4 with SemiDSConv and FIoU loss function. *Front. Mar. Sci.* 10:1153416. doi: 10.3389/fmars.2023.1153416

Underwater target detection algorithm based on improved YOLOv4 with SemiDSConv and FIoU loss function

Chengpengfei Zhang[1], Guoyin Zhang[1], Heng Li[1*], Hui Liu[1], Jie Tan[2] and Xiaojun Xue[1]

[1]Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China, [2]College of Engineering, Tongren Polytechnic College, Tongren, China

Underwater target detection is an indispensable part of marine environmental engineering and a fast and accurate method of detecting underwater targets is essential. Although many target detection algorithms have achieved great accuracy in daily scenes, there are issues of low-quality images due to the complex underwater environment, which makes applying these deep learning algorithms directly to process underwater target detection tasks difficult. In this paper, we presented an algorithm for underwater target detection based on improved You Only Look Once (YOLO) v4 in response to the underwater environment. First, we developed a new convolution module and network structure. Second, a new intersection over union loss was defined to substitute the original loss function. Finally, we integrated some other useful strategies to achieve more improvement, such as adding one more prediction head to detect targets of varying sizes, integrating the channel attention into the network, utilizing K-means++ to cluster anchor box, and utilizing different activation functions. The experimental results indicate that, in comparison with YOLOv4, our proposed algorithm improved the average accuracy of the underwater dataset detection by 10.9%, achieving 91.1%, with a detection speed of 58.1 frames per second. Therefore, compared to other mainstream target detection algorithms, it is superior and feasible for applications in intricate underwater environments.

KEYWORDS

deep learning, underwater detection, YOLO, convolutional neural network, loss function

1 Introduction

Underwater target detection technology has been widely used in marine biodiversity monitoring, marine ecosystem health assessment, and smart mariculture (Akkaynak and Treibitz, 2019). Due to the difficulties in data acquisition and the intricate underwater environment, underwater target detection has been an important and challenging task when it comes to detecting targets. The existing research on underwater target detection

methods can be broadly classified into two types: one is the traditional approach based on using hand-crafted features and shallow classifiers, and the other is a deep learning approach based on automatic feature extraction. Traditional target detection algorithms usually use a sliding window approach to delineate the region of interest on the input picture that may contain the target. Then, features will be extracted from the region-of-interest by using feature extraction algorithms, such as histogram of oriented gradient(HOG) (Dalal and Triggs, 2005), oriented fast and rotated brief(ORB)(Rublee et al., 2011), and scale-invariant feature transform(SIFT)(Lowe, 2004). Finally, classifiers such as adaboost (Yoav and Schapire, 1997), support vector machine (SVM) (Cortes and Vapnik, 1995), and deformable part model(DPM) (Felzenszwalb et al., 2008). are used to classify the extracted features. However, traditional target detection algorithms have many disadvantages, such as their poor robustness, low efficiency, and limited accuracy, which makes it difficult to meet the current demand. For the past few years, deep convolutional neural networks(DCNN) have been widely used in many fields such as medical image semantic segmentation (Wang Z. et al., 2022), urban land-use planning (Zhu et al., 2022), and autonomous driving (Li and Jin, 2022), with satisfactory results. Many approaches based on DCNN principles have been devised, and their effectiveness has been proven in a variety of domains, including in underwater target detection.

Target detection methods based on DCNN are gradually evolving in two directions due to the divergent focus on detection accuracy and detection speed. One is a region proposal-based target detection algorithm, also called the two-stage algorithm. Among all these algorithms, the R-CNN series is the most representative. R-CNN (Girshick et al., 2014) was presented by R. Girshick et al. in 2014, and it significantly outperformed the mainstream algorithm on the Pascal VOC dataset. It applies a selective search method to engender region proposals and uses CNN to extract features. After that, features are classified using SVM. Based on R-CNN, Fast R-CNN(Girshick, 2015), Faster R-CNN (Ren et al., 2017), and Mask R-CNN (He et al. 2018), many other two-stage methods have been gradually proposed and achieved better accuracy and speed. However, these two-stage algorithms have high computation time, which makes it difficult to meet the needs for real-time target detection. In order to resolve this issue, the regression-based target detection algorithm, also called the one-stage algorithm, was proposed. You Only Look Once (YOLO) (Redmon et al., 2016) was first introduced by J. Redmon et al.in 2015. When it was proposed, it attracted a lot of attention. YOLO's core idea is to use the whole picture as the input to the CNN and output the result of bounding box prediction. (Zhang et al., 2022) Because of this, YOLO has fast detection speed. Since its development, one-stage algorithms such as single shot multibox detector (SSD) (Liu et al., 2016) and RetinaNet (Lin et al., 2017). Were gradually proposed, and one-stage target detection algorithms were developed rapidly.

Although most of the algorithms mentioned above have achieved good performance in daily scenes, applying these deep learning algorithms directly to process underwater target detection tasks still has some problems. Firstly, the targets have a relatively large variation in scale due to the shooting distance. Secondly, underwater images are generally low-quality due to the complex and changing underwater environment, which means models have a low target localization accuracy in the underwater target detection assignment. Finally, looking at the research on underwater target recognition based on deep learning, although most of the existing detection methods have high recognition precision, the real-time performance of many of them is insufficient due to their high complexity, large number of parameters, and large scale. Therefore, it is essential to develop an underwater target detection algorithm that meets the needs for real-time detection while ensuring recognition accuracy.

In this paper, we presented an algorithm for underwater target detection based on improved YOLOv4 (Bochkovskiy et al., 2020) to solve the above-mentioned issues. In terms of network structure, we followed the original version, used CSPDarknet53 (Wang et al., 2020) as the backbone, and introduced channel attention block into it to emphasize useful informative features. Then, we constructed a new convolution module by integrating the traditional convolution, the depthwise separable convolution (DSC), and channel shuffle (Zhang et al., 2018), named SemiDSConv for convenience. This module can ensure the performance similar to a traditional convolution network, reduce the computational cost, and speed up the inference while solving the channel information separation problem caused by DSC. Based on this new module, inspired by CSPNet, we further designed the SemiDSCSP module, and applied it with the SemiDSConw module to the neck part of the model to replace the original convolution network and further reduce the inference time. In the head part, we added a prediction head to help the model deal with large changes in the targets' scale. Meanwhile, we defined a new intersection over union (IoU) loss function, FIoU, which boosts the localization accuracy and the convergence speed of the model. In comparison with the original YOLOv4, our improved YOLOv4 can better deal with underwater target detection tasks. For the dataset of URPC, the mAP was increased by 10.9% with the baseline and the inference speed reaching 58.1 frames per second (FPS). Overall, the presented algorithm demonstrates good results with a quick speed. The contributions of our work can be summed up as follows:

1. Developed a new convolution module named SemiDSConv. This module's performance is close to the traditional convolution network, but with less computation and faster inference speed. Based on it, the SemiDSCSP module was then designed and replaced the traditional convolution in the neck part;
2. Defined a new IoU loss, FIoU, that obtains superior localization accuracy and faster convergence speed;
3. Integrated some other useful tricks, such as introducing the channel attention block which can help the network to extract useful informative features more easily, adding a new prediction head to deal with dramatic changes in the scale of the underwater targets, using Mish as activation function, and using the K-means++ clustering algorithm to cluster anchor boxes;
4. On the URPC dataset, the proposed method achieved 91.1% mAP, outperforming the baseline by 10.9% with 58.1 FPS.

2 Related work

2.1 YOLOv4

Since the YOLO algorithm was first presented by J. Redmon et al. in 2015, it has received great attention among researchers. YOLOv4 was introduced in 2020 and is one of the state-of-the-art object detection algorithms. It greatly improved the detection accuracy and computational speed of YOLOv3 (Redmon and Farhadi, 2018). On COCO target detection dataset, YOLOv4 improves YOLOv3's FPS by 12%. Compared to other one-stage algorithms, such as SSD, YOLOv4 has a detection accuracy that far exceeds theirs while having the speed to meet real-time detection requirements. Compared to YOLOv5 and v7, it is lighter and has a faster detection speed when handling underwater target detection tasks with not much difference in accuracy. Thus, YOLOv4 is suitable for real-time target detection tasks.

YOLOv4 mainly consists of three sections: the backbone, the neck, and the head. YOLOv4 takes CSPDarknet53 as the backbone network. CSPDarknet53 is composed of five large residual blocks which contains one, two, eight, eight, and four residual units in them, respectively. Each residual unit consists of 3*3 and 1*1 convolutional layers. This architecture can help the network to get richer gradient information while reducing the amount of calculation needed. In the neck part, YOLOv4 uses PANet(Liu et al., 2018) to fuse the feature information from different-size feature maps to enhance the ability of the model to detect objects of various sizes. Meanwhile, Yolov4 adds the SPP block into the network which can expand the receptive field, prevent overfitting, and improve scale-invariance. In the end, the extracted multi-scale feature maps are sent into the YOLOv3 detection head for detection.

2.2 Channel attention

Channel attention mechanisms have shown their utility across many tasks. For the underwater image, typically, targets only occupy a fraction of the whole image, and the rest is background information. In order to minimize the distractions of background information and highlight the target, channel attention can be used to help distinguish the target from the background as channel attention focuses on what is meaningful given an image (Woo et al., 2018). SENet(Squeeze-and-Excitation Network) (Hu et al., 2018) was proposed by Jie Hu et al., which is a prominent representative of channel attention. It is composed of two parts: a squeeze operation and an excitation operation. The squeeze operation uses global average pooling to aggregate the summarized information from each channel, and the excitation operation adjusts the relevance of each channel according to its weight. Therefore, the introduction of the SE block can enhance the feature extraction capability of the model. The structure of the SE block is indicated in Figure 1.

2.3 Activation functions

The activation Function is one of the crucial factors influencing the performance of a neural network. The rectified linear unit (ReLU) (Glorot et al., 2011) was proposed by Vinod Nair et al. in 2011. Its formula is defined in Equation (1).

$$\mathbf{f_{ReLU}(x) = max\,(0, x)}\ , x \in \mathbf{R}\,. \tag{1}$$

Due to its low computational cost and easy optimization characteristics, ReLU is widely used in neural networks. However, it is not without weaknesses. As shown in Equation (1), ReLU grows unbounded and is directly truncated at negative values. The former would lead to excessive differences in weights, resulting in reduced accuracy. The latter would result in a Dead ReLU problem, i.e. if the input is a negative value, the output of ReLU and the gradient will become zero. Finally, the network parameters will not be updated. Alex Krizhevsky proposed ReLU6 (Krizhevsky and Hinton, 2010) to address the former issue, which is formulated in Equation (2).

$$\mathbf{f_{ReLU6}(x) = \min\,(6, \max\,(0, x))}\ , x \in R\,. \tag{2}$$

But it still does not solve the Dead ReLU problem. In 2019, Diganta Misra et al. presented Mish activation function (Misra, 2019), which can be defined as:

$$\mathbf{f_{Mish}(x) = xtanh(\ln\,(1 + e^{x}))}, x \in R\,. \tag{3}$$

Compared with ReLU, Mish is non-monotonic, smoother, and allows a few negative weight inflow. Figure 2 shows visually the differences between ReLU, ReLU6, and Mish. Better expressivity and information flow are facilitated by these properties, and these properties also make the network avoid saturation.

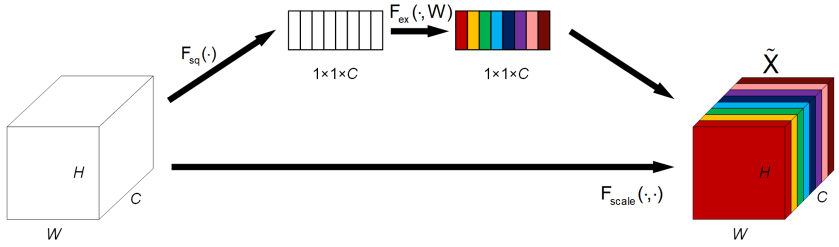

FIGURE 1
The structure of the SE block.

2.4 General target detection

Since the rise of convolutional neural networks, many researchers have continued to propose new methods and ideas due to the need for various target detection tasks. Aiming to improve the assignment of anchor labels in the current anchor-based model, Kim (Kim and Lee, 2020) et al. proposed a probabilistic model for assigning labels to anchors - Probabilistic Anchor Assignment(PAA), the assignment criteria of which depend on the combination of classification accuracy and IoU, rather than IoU alone. Redundant hyperparameters such as IoU threshold and number of positive samples, are then discarded to improve the performance and stability of the model. Yang (Yang et al., 2022) et al. proposed the Cascade Sparse Query (CSQ) mechanism, where Query represents using the query passed in the deeper-level (higher-level feature with lower resolution) layer to guide the detection of small targets in this layer, and then predicting the query in this layer to be further passed to the next layer. Sparse represents the significant reduction of the computational overhead of the detection head on the low-level feature layer by using sparse convolution. Li(Li et al., 2022) et al. improved Multiscale Vision Transformers which incorporates decomposed relative positional embeddings, proposed MViTv2, and optimized the pooling attention in the network using residual structures. After that, many experiments have been conducted to verify the superiority of the proposed algorithm in the fields of classification, detection, and video tracking. To address the problem of sample scarcity in the dataset, Hou(Hou et al., 2022) et al. creatively proposed a new idea to explore the relationship between samples and help the network to learn by focusing on the batch dimension and introducing the Transformer structure in it. The proposed BatchFormer has achieved good performance in a large number of experiments.

2.5 Underwater target detection

In the past few years, with the evolution of deep learning-based target detection algorithms, more and more researchers have been implementing this technology in the underwater environment. In 2019, Moniruzzaman (Moniruzzaman et al., 2019) et al. constructed a Halophila ovalis dataset that consists of 2,699 underwater photographs of Halophila ovalis and presented Inception V2-based Faster R-CNN network to detect seagrass. Experimentally, the proposed network achieved a high mAP of 0.3464 on laboratory images. In 2021, Zeng (Zeng et al., 2021) et al. presented a method to introduce the adversarial occlusion network (AON) to the Faster R-CNN algorithm and the resulting model achieves better robustness in terms of underwater seafood. In the same year, Wang (Wang et al., 2021) et al. introduced YOLOv5 for underwater target detection and conducted a lot of detailed experiments and comparisons based on this, and finally used the experimental results as the YOLOv5 baseline for underwater target detection. For the task of underwater sea cucumber target detection, Peng (Peng et al., 2021) et al. proposed the Shortcut Feature Pyramid Network (S- FPN) and Piecewise Focal Loss (PFL), which improved the multi-scale feature fusion approach of the network and balanced the positive and negative samples, enabling the mAP to achieve a high accuracy of 94%. Yeh (Yeh et al., 2021) et al. proposed an underwater target detector with joint image color conversion for the problem of underwater image color absorption, which converts underwater color images to grayscale images, and improved the performance of the target detector with low computational cost. In 2022, Hong (Hong et al., 2022) et al. used a parameter calibration strategy to fine-tune the parameters of the Mask RCNN model to detect and locate shrimp better. Cai (Cai et al., 2022) et al. proposed a weakly supervised learning framework for underwater object detection, using two detectors trained

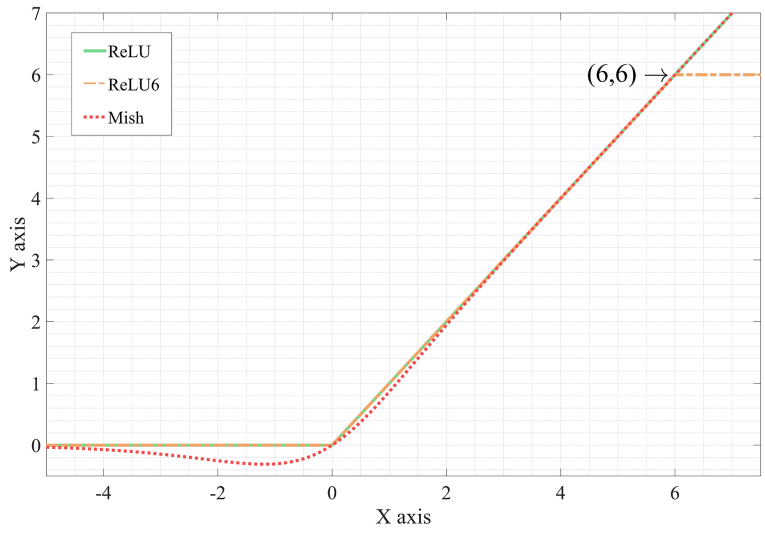

FIGURE 2
Comparison between ReLU, ReLU6, and Mish.

simultaneously and learning from each other to select cleaner samples, which eventually achieved good performance. Chen (Chen et al., 2022) et al. proposed the Sample-WeIghted hyPEr Network (SWIPENET) and a novel training paradigm called Curriculum Multi-Class Adaboost (CMA) to address both problems simultaneously for the case of ambiguous underwater targets and the presence of many small targets, which eventually achieved good performance. In 2023, Wang (Wang et al., 2023) et al. proposed a new underwater target detection algorithm based on reinforcement learning and image enhancement, which automatically learns and adjusts the combined sequence of underwater image enhancement methods by a neural network in order to help the network's detector achieve the best performance. Although these works achieved quite a high degree of detection accuracy, there are still some limitations to them, namely the low detection speed. Therefore, how to ensure a high detection accuracy with real-time rapid detection is still a research issue worthy of study.

3 Proposed model

3.1 Network structure

Considering the speed requirements of real-time detection tasks, we chose the best-known and the most used one-stage algorithm—YOLOv4-as our baseline. The framework of the improved YOLOv4 is shown in Figure 3. We introduced a new convolution module and a bottleneck structure based on it to speed up the network inference. A new IoU loss function was developed to enhance detection precision and the velocity of convergence. A new prediction head was added to deal with the large differences in underwater target scales. The prediction head we added uses mainly high-resolution and shallow features to predict, which makes it sensitive to small targets. Therefore, the newly added prediction head and the original prediction heads form a four-head structure that can better handle the drastic changes in the size of underwater targets. The channel attention module was introduced into the backbone to encourage the network to retain more useful features. In addition, we used Mish activation function to replace ReLU. It solves the Dead ReLU problem, avoids network convergence slowdown, and, at the same time, improves the accuracy of the network. Although it slightly increases the computational cost, we deem it worthwhile.

3.2 SemiDSConv module

The depthwise separable convolution (DSC) is composed of two parts: depthwise convolution and pointwise convolution. Depthwise convolution convolves each channel of the input feature map separately. If the amount of input channels is N, after convolving each of the N channels, these feature maps are collocated together to get an output feature map of channel N. Pointwise convolution is a 1×1 convolution. The pointwise convolution in DSC is mainly used to allow DSC to freely change the number of output channels and to perform channel fusion on the output feature map of depthwise convolution. The ratio of the computational cost of DSC to conventional convolution is illustrated in Equation (4)

$$\frac{k \cdot k \cdot n \cdot s \cdot s + n \cdot m \cdot s \cdot s}{k \cdot s \cdot n \cdot m \cdot s \cdot s} = \frac{1}{m} + \frac{1}{k^2} \tag{4}$$

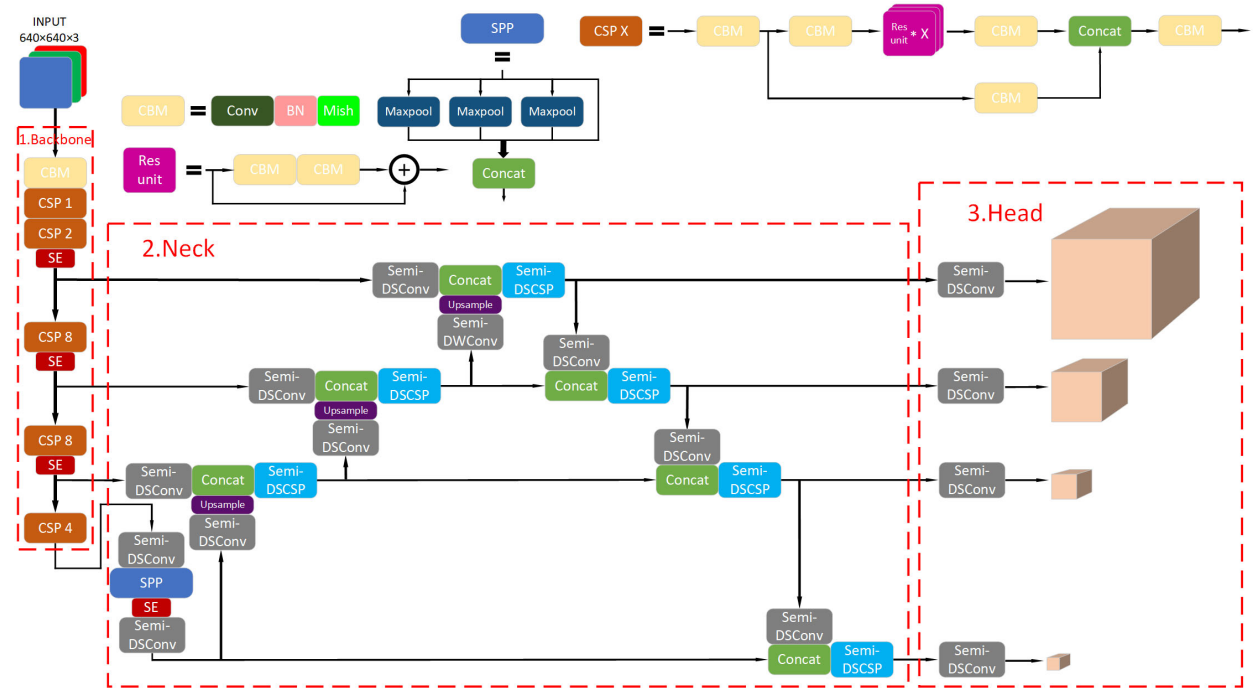

FIGURE 3
The improved YOLOv4 network structure.

Where *k·k.* is the convolution kernel size. *n* and *m* denote the input and output channels, separately. *s·s* represents the size of the feature map. From the equation, it is clear that the computational cost of DSC is much less than that of traditional convolution.

However, due to the characteristics of DSC, channel information is computed separately from each other, resulting in a significant reduction in its capability to extract and fuse features, much weaker than traditional convolution. To overcome this issue, the SemiDSConv module was designed. The structure of the SemiDSConv module is indicated in Figure 4.

The SemiDSConv module first uses a 1*1 convolution kernel to fuse the input features maps, while achieving channel dimensionality reduction to reduce the computational cost of subsequent convolution operations. After that, the feature maps are computed through the traditional convolution and the depthwise separable convolution, respectively. The channels are then concatenated together. It then performs shuffle operations so that the information between the channels is completely fused. The SemiDSConv module effectively maintains the advantages of DSC while minimizing the negative impact of its shortcomings on the network.

Based on this, inspired by the CSPNet, we also designed the SemiDSCSP module, which enables the network to better extract and fuse the feature information. The structure of the SemiDSCSP module is indicated in Figure 5.

It is worth mentioning that if all traditional convolutions in the network are replaced with SemiDSConv, the number of network layers will be too deep. This would make the resistance of data flow too high and increase the inference time significantly. In the Neck part, the feature map is extracted by the backbone, with smaller width and height, less redundant repetitive information, and shorter inference time. Therefore, we replaced traditional convolutions only in the Neck to achieve good performance.

3.3 FIoU loss function

Due to differences in the network structure and the basic idea, YOLO has its natural disadvantage in localization precision compared with a two-stage algorithm. Therefore, the authors of the YOLO series and other researchers have been exploring strategies to address this issue. Among the various improvement strategies, improving the loss function is the most effective and direct strategy. YOLOv4 includes three types of loss functions: confidence loss, category loss, and localization loss (also called the loss of bounding box coordinates). Different from YOLOv3, YOLOv4 substitutes Complete-IoU (CIoU) (Zheng et al., 2021) loss for cross entropy loss in YOLOv3 as the localization loss function and obtains better convergence speed and accuracy (Jiao et al., 2022). The CIoU loss was improved from Distance-IoU (DIoU) (Zheng et al., 2020) loss. The DIoU loss and the CIoU loss is defined in Equations (5)-(9):

$$\mathcal{L}_{DIoU}=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2} \tag{5}$$

$$\mathcal{L}_{CIoU}=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2}+\alpha v. \tag{6}$$

$$IoU=\frac{|A\cap A_{gt}|}{|A\cup A_{gt}|}. \tag{7}$$

$$\alpha=\frac{v}{(1-IoU)+v}. \tag{8}$$

$$v=\frac{4}{\pi^2}(\arctan\frac{w_{gt}}{h_{gt}}-\arctan\frac{w}{h})^2. \tag{9}$$

where p and p_{gt} are the central points of the predicted box and the ground-truth box. d is the diagonal length of the minimum bounding rectangle. ρ (p, p_{gt}) indicates the Euclidean distance between p and p_{gt}. A denotes the predicted box whereas A_{gt} denotes the ground-truth box. w, w_{gt},h, and h_{gt} respectively represent the width of the predicted box and ground-truth box and the height of the two boxes.

As shown in Equations (6), (8), and (9), the newly added penalty term av is to measure the discrepancy of aspect ratio between the predicted box and the ground-truth box. The experimental results indicate that, compared with previous IoU loss functions (GIoU and DIoU) (Rezatofighi et al., 2019), the localization accuracy and the convergence speed of the CIOU loss have substantially increased. However, CIoU still has certain limitations. Specifically, when $\{w= kw_{gt} = kh_{gt}|k\in R^+\}$ is satisfied, v becomes zero and the loss function will degrade to DIoU loss. This drawback renders the convergence speed slow in some cases. For the underwater target detection task, the slow convergence of the loss function may cause the network to fail and to converge quickly

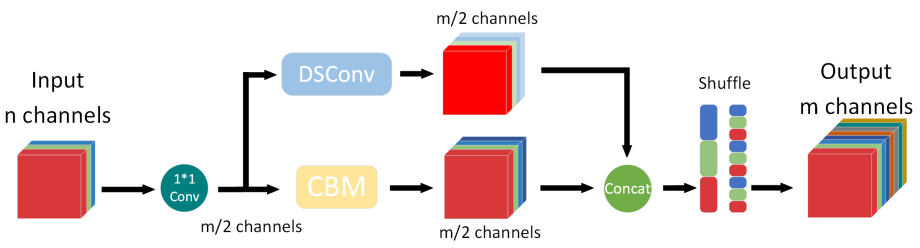

FIGURE 4
The structure of the SemiDSConv module.

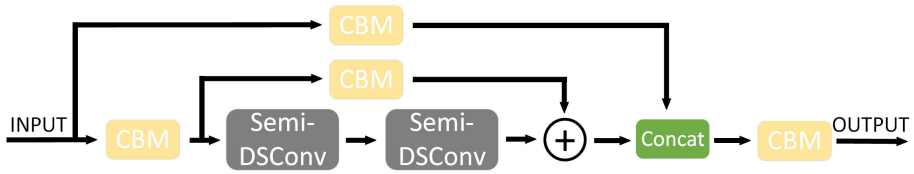

FIGURE 5
The structure of the SemiDSCSP module.

in a limited number of epochs due to the small number of samples. It may also lead to overfitting if the training epochs are extended for model convergence.

In order to address this situation, we designed a new loss function that inherited some properties from CIoU loss and added proper penalty terms to it. We call it Fast-IoU(FIoU); the specific formula is shown as follows:

$$\mathcal{L}_{FIoU}=L_{IoU}+L_D+L_R+L_L=1-IoU+\frac{\rho^2(p,p_{gt})}{d^2}+\alpha v+\frac{\rho^2(h,h_{gt})}{l_h^2}+\frac{\rho^2(w,w_{gt})}{l_w^2}. \quad (10)$$

where, l_h and l_w are the height and width of the minimum bounding rectangle. As shown in Equation (10), we divide the whole loss function into four parts: the IoU loss L_{IoU}, the distance loss L_D, the aspect ratio loss L_R and the side length loss L_L. Generally, L_R and L_L function together to optimize the similarity between two boxes. If $\{w = kw_{gt}, h = kh_{gt} | k \in R^+\}$ is satisfied, although L_R becomes zero, L_L it is still minimizing the difference between the two boxes' width and height. The convergence process of the CIoU and the FIoU is shown in Figure 6.

In order to verify the effect of different loss functions on the network model performance, we evaluate FIoU loss function by replacing CIoU with FIoU in the original YOLOv4 algorithm. Figure 7 shows the training loss curves of two models in the URPC dataset. As can be seen, the FIoU decreased more quickly than CIoU in epochs 0 to 30. After 30 epochs, the curve of FIoU loss functions was stable while CIoU was not. Although after 45 epochs, both the FIoU and the CIoU loss functions were stabilized, FIoU was still well below CIoU. It verifies that the FIOU loss function has a quicker convergence rate and better regression accuracy than the CIOU loss function.

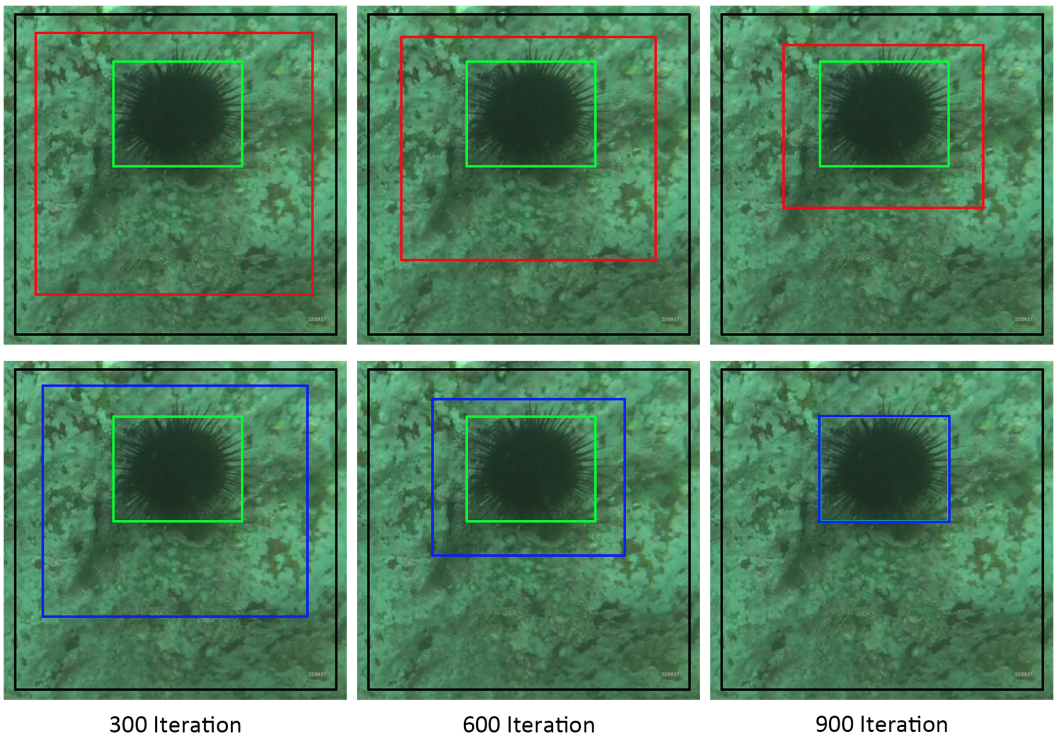

FIGURE 6
The diagrams of prediction box regression in the first and second row respectively represent the prediction box regression process of CIoU and FIoU. The green box refers to the ground truth box. The black box refers to the anchor box, and the red and blue one is the prediction boxes of CIoU and FIoU, respectively.

Overall, compared to CIoU loss, FIoU can get better localization accuracy and convergence speed. This enables the YOLOv4 network using FIoU as the loss function to have a higher performance than the network using CIoU as the loss function. We substitute FIoU loss for CIoU loss in YOLOv4, hoping to render it better for the underwater target detection task.

4 Experiments

4.1 Dataset

The dataset adopted in the paper was from the Target Recognition Group of China Underwater Robot Professional Competition (URPC), which includes four categories: echinus, holothurian, scallop, and starfish. The dataset contained 4757 images in total. The dataset is a sequence of frames from multiple video segments with a continuous distribution and a large similarity between neighboring frames. Therefore, we shuffled the dataset randomly and split the dataset into a training and test set at a ratio of 4:1, then labeled the targets. In order to better simulate the real situation in the underwater environment, we kept the images without targets detected in the training set and test set. The finally obtained training set contains 3806 images and the test set contains 951 images. One practical issue deserves mention: the resolution of images and the number of individual category samples are very unbalanced in the dataset. This would bring challenges to the training of the network.

4.2 Model evaluation metrics

In the field of target detection, Average Precision(AP) is the metric most commonly used to evaluate the performances of the model. Before introducing AP, we present a brief overview of precision (P) and recall (R), which are computed by Equations (11) and (12):

$$\mathbf{P}=\frac{\mathbf{TP}}{\mathbf{TP+FP}}\times\mathbf{100\%}. \tag{11}$$

$$\mathbf{R}=\frac{\mathbf{TP}}{\mathbf{FN+TP}}\times\mathbf{100\%} \tag{12}$$

where *TP*, *FP* and *FN* refers to the positive samples predicted to be positive by the model, the negative samples predicted by the model to be positive, and the positive samples predicted to be negative by the model, respectively.

Because P and R are interactive, to combine the two metrics, AP is introduced to evaluate the goodness of the detection accuracy of the model, as defined in Equation (13):

$$\mathbf{AP}=\int_0^1 \mathbf{P(R)dR}. \tag{13}$$

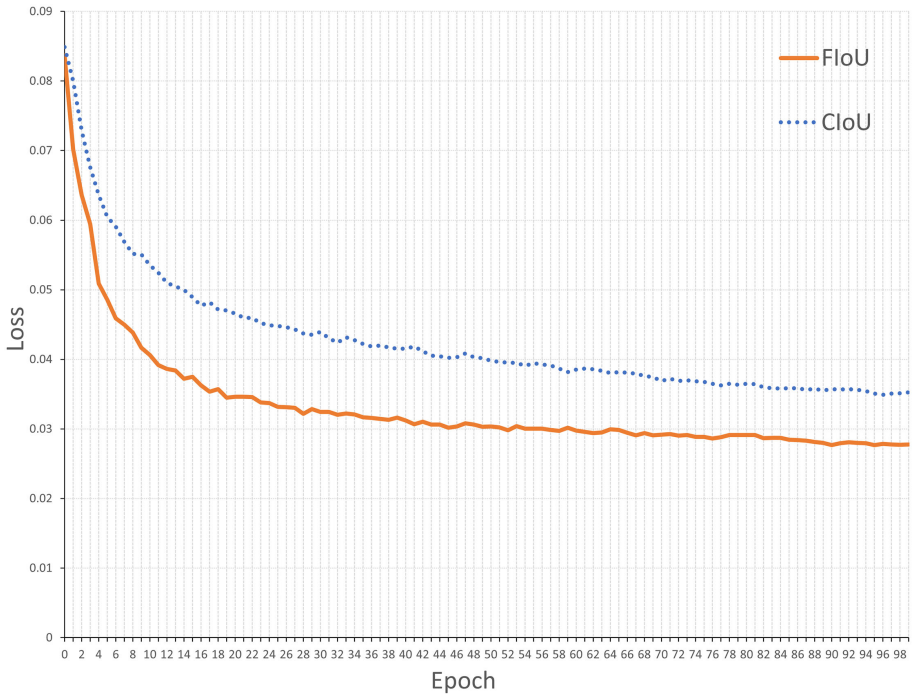

FIGURE 7
Curves of the FIoU and CIoU loss values with the epoch increasing.

In multi-class target detection tasks, mean Average Precision is commonly used to evaluate the overall model performance. Namely, AP values were averaged for each category. The equation for calculating mAP as defined below:

$$\mathbf{mAP}=\frac{1}{n}\sum_{i=0}^{n}\mathbf{AP}. \quad (14)$$

where n refers to the number of types.

4.3 Experimental environment and parameter settings

We implement the proposed method on Python 3.9.7 and Pytorch 1.8.1. All the methods were trained and tested using an NVIDIA RTX3090 GPU and an Intel Xeon E7-4809 v3 CPU.

During the training phase, we set the initial training hyperparameters for each group of experiments to be the same to ensure the fairness of our experiments. The resolution of the input images were consistently set to 640 × 640. To prevent the gradients from exploding when the learning rate was high, the learning rate was tuned based on the cosine annealing strategy (Loshchilov and Hutter, 2016).

YOLOv4 algorithm expands the anchor mechanism. Setting a predefined prior frame can well represent the original state of the target to be detected and get a more reasonable potential distribution of data sample bounding boxes. The high-quality anchor can play an optimal role in the process of small target detection and post-processing prediction. Therefore, when training underwater data, it is very important to set appropriate anchors according to the characteristics of the underwater dataset. In this paper, we used the K-means++ (Arthur and Vassilvitskii, 2007) clustering algorithm to cluster anchor boxes in the URPC dataset. Finally, we obtain the anchor parameters' fit among the underwater targets. The clustered anchor boxes are (17,14), (24,21), (31,28), (37,39), (48,32), (54,46), (69,62), (92,89), and (144,129).

The specific settings of the other hyperparameters are shown in Table 1.

The loss function curves of the proposed method are demonstrated in Figure 8, which contains three parts: localization loss, classification loss, and confidence loss. From the figure, it can be noted that all losses steadily decrease with the number of epochs. The model converged in under 100 epochs.

In the testing stage, all the resolutions of the input image were consistently set to 640 × 640. The IoU threshold was set to 0.4. All other parameters were the same. During the test, only one GPU was used uniformly for testing. The average of the 10 test results for the entire test set test time was considered as the final prediction time.

4.4 Experimental results and analysis

4.4.1 Ablation experiments

To verify the effectiveness of the proposed model or every submodule, we present ablation experiments in this paper.

Table 2 shows the results of the ablation experiments. As listed in Table 2, Model 1(baseline) was the original YOLOv4 network structure. Model 2 replaced the ReLU activation function in Model 1 with the Mish activation function. Model 3 replaced the CIoU loss function in Model 2 with our proposed FioU loss function. Model 4 was model 3 with SemiDSConv and SemiDSCSP. Model 5 was the proposed four-head structure based on model 4 and model 6 was the model in which the SE channel attention mechanism module was embedded into Model 5.

The results showed that both Model 2 and Model 5 have improved performance separately to varying degrees compared to the previous model. In comparison to Model 2, Model 3, which used FIoU loss function, increased the mAP by 4.3%. The proposed Model 4 increased the mAP by 3.7% and also improved the detection speed by about 14 FPS. After embedding SE channel attention into the network, the proposed Model 6 attained the best performance. Compared to the original YOLOv4 algorithm (Model 1), Model 6 's mAP increased from 80.2% to 91.1%, an increase of 10.9%.

It may be noted that the presented model not only reduces the computational cost and improves the detection speed, but also achieves good performance compared to the baseline.

4.4.2 Detection results comparison

To demonstrate the superiority of the proposed method in the detection of underwater targets, we compared it with the original YOLOv4 algorithm and six other methods: YOLOv5, YOLOv7 (Wang CY, et al., 2022), Tiny YOLOv4, YOLO-Fish(Al Muksit et al., 2022), Faster R-CNN, and SSD. All tests were performed on the URPC dataset. The results of these experiments are shown in Table 3.

It can clearly be seen from Table 3 that the presented method has the highest mAP, while the detection speed is faster than the baseline, meeting the demand for real-time detection.

Figure 9 indicates the visualization experimental result of YOLOv4, Tiny-YOLOv4, YOLOv5, YOLOv7, and our method for underwater detection on the URPC dataset. As can be discerned

TABLE 1 Hyperparameter settings.

Training Epochs	Batch Size	Learning Rate	Weight Decay	Momentum	Cosine Annealing
100	8	0.00522	0.00044	0.98	0.114
Translate (Image **Translation**)	Scale (Image Scale)	Fliplr **(Image Flip Left-Right)**	Flipud (Image Flip **Up-Down**)	Mosaic	Mixup
0.0726	0.9	0	0.5	0.932	0

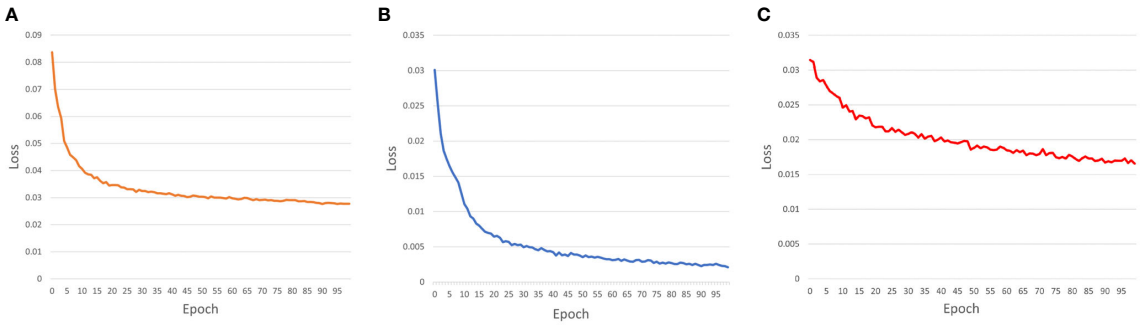

FIGURE 8
The curves of the loss values: **(A)** localization loss; **(B)** classification loss; **(C)** confidence loss.

TABLE 2 Results of ablation experiments.

Model	Method						mAP(%)*	Speed (FPS)
	Baseline	Mish	FIoU	Semi-DSConv	New Head	SE		
Model1	√						80.2	51.2
Model2		√					80.6(+0.4)	49.3
Model3		√	√				84.9(+4.3)	49.3
Model4		√	√	√			88.6(+3.7)	63.4
Model5		√	√	√	√		90.5(+1.9)	58.5
Model6		√	√	√	√	√	91.1(+0.6)	58.1

*The value within the bracket denotes the improvement compared to the previous model

TABLE 3 Experimental results of different algorithms on the URPC dataset.

Method	mAP (%)	Scallop (%)	Starfish (%)	Holothurian (%)	Echinus (%)	Model Size (MB)	Speed (FPS)
YOLOv4	80.2	73.5	87.2	77.7	82.3	204.8	51.2
YOLOv5	80.4	72.9	87.4	76.3	84.8	243.2	44.7
YOLOv7	80.5	73.6	89.7	73.7	85.1	186.0	48.9
YOLO-Fish	77.5	69.1	86.7	71.6	82.6	234.8	45.6
Tiny YOLOv4	63.7	58.5	70.8	56.5	69.0	23.0	114.9
Faster R-CNN	84.4	78.2	93.3	79.1	86.9	419.2	4.8
SSD	61.5	59.3	68.4	56.0	62.2	36.4	72.3
Ours	91.1	86.2	93.2	89.7	95.2	182.7	58.1

from Figure 9, the detection result of our method was better than YOLOv4, and considerably better than the Tiny YOLO v4.

To better demonstrate the detection results of our proposed algorithm with other algorithms, we compared our proposed algorithm with YOLOv5 and YOLOv7 in detail. Figure 10 shows the detection results of the three algorithms. As shown in the figure, the targets marked with red-dashed boxes in the figure have obscure and blurred edges, which are difficult to distinguish from the background, for which our algorithm can still identify and label well. At the same time, many targets underwater are easily misidentified due to the complex environment, and the yellow-dashed boxes in the figure mark the targets that are misidentified by the algorithm. As can be seen, our proposed algorithm has a low false detection rate and is suitable for using in complex underwater environments.

All the experimental results show that our proposed method achieves a good trade-off between detection accuracy and detection speed, which means that it is considered superior for underwater target detection.

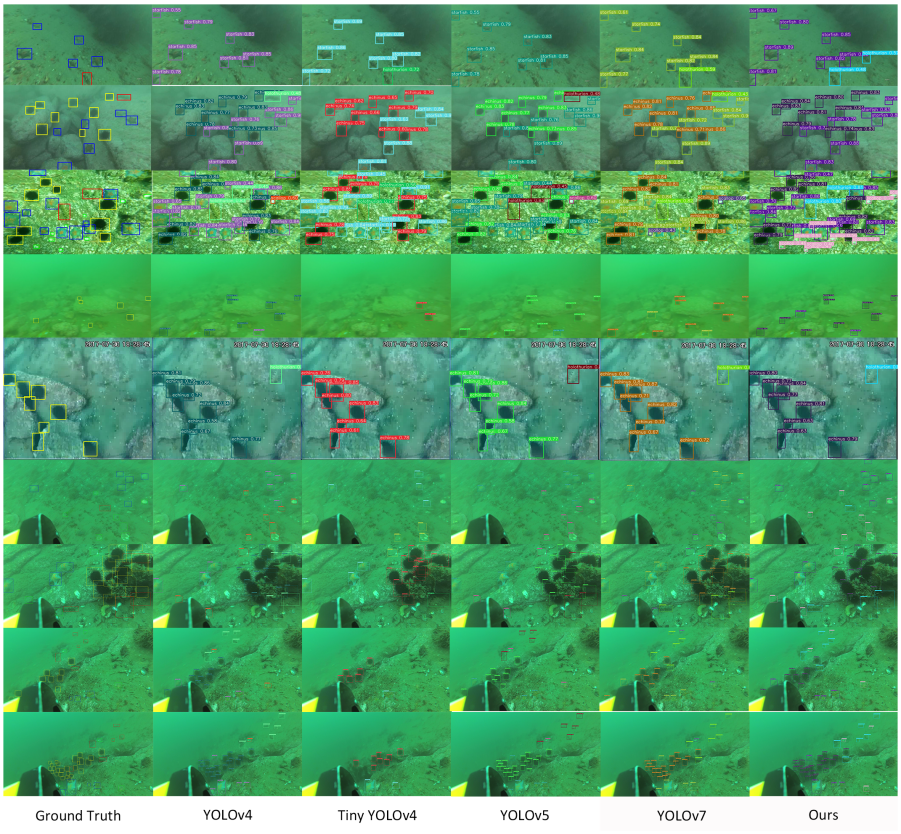

FIGURE 9
Visualization comparison of detection results with YOLO v4, Tiny YOLO v4, YOLOv5, YOLOv7, and ours.

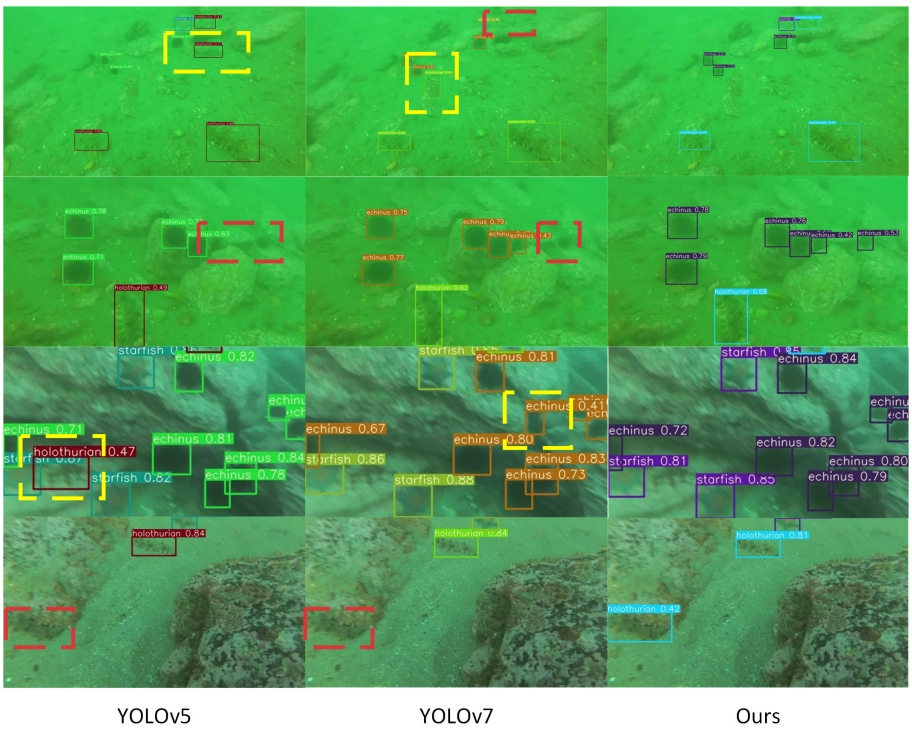

FIGURE 10
More detailed visualization comparison of detection results with YOLOv5, YOLOv7, and ours.

5 Conclusions

Detecting targets with good accuracy and fast detection speed in underwater environments is a challenging problem. In this paper, we presented a real-time underwater target detection algorithm based on improved YOLOv4. In our work, we first developed a new convolutional module and network structure to enhance the feature extraction capability for the model, reduce the computational effort, and speed up the model inferencing. Then, we defined a new IoU loss that improves the target detection performance and the convergence speed of the network. Meanwhile, we optimized the network model and made some other small improvements. We added a new prediction head to handle dramatic changes in the scale of the underwater targets and embedded the channel attention block in the network, which makes the detection and classification of the network more accurate. Experiments show that the presented model achieves 91.1% mAP and 58.1 FPS detection speed on the URPC dataset, outperforming the other listed algorithms in terms of combined performance, which indicates that the proposed model has significant advantages in handling underwater target detection tasks and is more robust in complex underwater environments.

In our future work, how to compress model size to design a more lightweight network and make it applicable to small, embedded devices while maintaining accuracy is an issue that merits further research.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material. Further inquiries can be directed to the corresponding author.

Author contributions

CZ conceived, planned, and performed the designs and drafted this paper. GZ, HEL, JT and HUL provided guidance and reviewed this paper. XX provided the design ideas and edited this paper. All authors contributed to the article and approved the submitted version.

Funding

This research was funded by National Natural Science Foundation of China, grant number 61863018, and the Applied Basic Research Foundation of Yunnan Province, grant number 202001AT070038.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Akkaynak, D., and Treibitz, T. (2019). "Sea-Thru: A method for removing water from underwater images," in *2019 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 1682–1691.

Arthur, D., and Vassilvitskii, S. (2007). "K-means++ the advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms* (Society for Industrial and Applied Mathematics 3600 University City Science Center Philadelphia, PA United States), 1027–1035.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv*. 10934. doi: 10.48550/arXiv.2004.10934

Cai, S., Li, G., and Shan, Y. (2022). Underwater object detection using collaborative weakly supervision. *Comput. Electrical Eng.* 102, 108159. doi: 10.1016/j.compeleceng.2022.108159

Chen, L., Zhou, F., Wang, S., Dong, J., Li, N., Ma, H., et al. (2022). SWIPENET: Object detection in noisy underwater scenes. *Pattern Recognition* 132, 108926. doi: 10.1016/j.patcog.2022.108926

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20 (3), 273–297. doi: 10.1007/BF00994018

Dalal, N., and Triggs, B. (2005). "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition* (New York: IEEE Press), 886–893.

Felzenszwalb, P., McAllester, D., and Ramanan, D. (2008). "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 1–8.

Freund, Y., and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. System Sci.* 55 (1), 119–139. doi: 10.1006/jcss.1997.1504

Girshick, R. (2015). "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision* (New York: IEEE Press), 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 580–587.

Glorot, X., Bordes, A., and Bengio, Y. (2011). "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. (Brookline: Microtome Publishing). 315–323.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2018). Mask r-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2), 386–397. doi: 10.1109/TPAMI.2018.2844175

Hong, K. T., Abdullah, S. N. H. S., Hasan, M. K., and Tarmizi, A. (2022). Underwater fish detection and counting using mask regional convolutional neural network. *Water* 14 (2), 222. doi: 10.3390/w14020222

Hou, Z., Yu, B., and Tao, D. (2022). "BatchFormer: Learning to explore sample relationships for robust representation learning," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 7246–7256.

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 7132–7141.

Huang, H., Zhou, H., Yang, X., Zhang, L., Qi, L., and Zang, A. Y. (2019). Faster r-CNN for marine organisms detection and recognition using data augmentation. *Neurocomputing* 337, 372–384. doi: 10.1016/j.neucom.2019.01.084

Jiao, W., Cheng, X., Hu, Y., Hao, Q., and Bi, H. (2022). Image recognition based on compressive imaging and optimal feature selection. *IEEE Photonics J.* 14 (2), 1–12. doi: 10.1109/JPHOT.2022.3155489

Kim, K., and Lee, H. S. (2020). "Probabilistic anchor assignment with iou prediction for object detection," in *Computer vision–ECCV 2020: 16th European conference* (Germany: Springer International Publishing), 355–371.

Krizhevsky, A., and Hinton, G. (2010) *Convolutional deep belief networks on cifar-10*. Available at: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=bea5780d621e669e8069f05d0f2fc0db9df4b50f (Accessed February 26, 2023).

Li, P., and Jin, J. (2022). "Time3D: End-to-End joint monocular 3D object detection and tracking for autonomous driving," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 3875–3884.

Li, Y., Wu, C. Y., Fan, H., Mangalam, K., Xiong, B., Malik, J., et al. (2022). "MViTv2: Improved multiscale vision transformers for classification and detection," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 4794–4804.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2018). Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (2), 318–327. doi: 10.1109/iccv.2017.324

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., et al. (2016). "Ssd: Single shot multibox detector002E," in *Computer vision–ECCV 2016: 14th European conference* (Germany: Springer International Publishing), 21–37.

Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). "Path aggregation network for instance segmentation," in *2018 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 8759–8768.

Loshchilov, I., and Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts. *arXiv*. 03983. doi: 10.48550/arXiv.1608.03983

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60 (2), 91–110. doi: 10.1023/b:visi.0000029664.99615.94

Misra, D. (2019). Mish: A self regularized non-monotonic neural activation function. *arXiv*, 08681. doi: 10.48550/arXiv.1908.08681

Moniruzzaman, M., Islam, S. M. S., Lavery, P., and Bennamoun, M. (2019). "Faster r-CNN based deep learning for seagrass detection from underwater digital images," in *2019 digital image computing: Techniques and applications* (New York: IEEE Press), 1–7.

Muksit, A., Hasan, F., Hasan Bhuiyan Emon, M. F., Haque, M. R., Anwary, A. R., and Shatabda, S. (2022). YOLO-fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecol. Inform* 72, 101847. doi: 10.1016/J.ECOINF.2022.101847

Peng, F., Miao, Z., Li, F., and Li, Z. (2021). S-FPN: A shortcut feature pyramid network for sea cucumber detection in underwater images. *Expert Syst. Appl.* 182, 115306. doi: 10.1016/j.eswa.2021.115306

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You only look once: Unified, real-time object detection," in *2016 IEEE conference on computer vision and pattern recognition* (New York: IEEE Press), 779–788.

Redmon, J., and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv*, 02767. doi: 10.48550/arXiv.1804.02767

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39 (6), 1137–1149. doi: 10.1109/tpami.2016.2577031

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). "Generalized intersection over union: A metric and a loss for bounding box regression," in *2019 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 658–666.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). "ORB: An efficient alternative to SIFT or SURF," in *2011 international conference on computer vision* (New York: IEEE Press), 2564–2571.

Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv*, 02696. doi: 10.48550/arXiv.2207.02696

Wang, Z., Li, T., Zheng, J. Q., and Huang, B. (2022). "When cnn meet with vit: Towards semi-supervised learning for multi-class medical image semantic segmentation," in *Computer vision–ECCV 2022 workshops* (Cham: Springer Nature Switzerland), 424–441.

Wang, C.-Y., Mark Liao, H.-Y., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). "CSPNet: A new backbone that can enhance learning capability of CNN," in *2020 IEEE/CVF conference on computer vision and pattern recognition workshops* (New York: IEEE Press), 1571–1580.

Wang, H., Sun, S., Bai, X., Wang, J., and Ren, P. (2023). A reinforcement learning paradigm of configuring visual enhancement for object detection in underwater scenes. [Preprint]. Available at: https://ieeexplore.ieee.org/document/10058092 (Accessed March 15, 2023).

Wang, H., Sun, S., Wu, X., Li, L., Zhang, H., Li, M., et al. (2021). "A yolov5 baseline for underwater object detection," in *OCEANS 2021* (San Diego Porto: IEEE Press), 1–4.

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018b). "CBAM: Convolutional block attention module," in *Computer vision–ECCV 2018: 15th European conference* (Germany: Springer International Publishing), 3–19.

Yang, C., Huang, Z., and Wang, N. (2022). "QueryDet: Cascaded sparse query for accelerating high-resolution small object detection," in *2022 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 13658–13667.

Yeh, C. H., Lin, C. H., Kang, L. W., Huang, C. H., Lin, M. H., Chang, C. Y., et al. (2021). Lightweight deep neural network for joint learning of underwater object detection and color conversion. *IEEE Trans. Neural Networks Learn. Syst.* 33 (11), 6129–6143. doi: 10.1109/TNNLS.2021.3072414

Zeng, L., Sun, B., and Zhu, D. (2021). Underwater target detection based on faster r-CNN and adversarial occlusion network. *Eng. Appl. Artif. Intell.* 100, 104190. doi: 10.1016/j.engappai.2021.104190

Zhang, Y. F., Ren, W., Zhang, Z., Jia, Z., Wang, L., and Tan, T. (2022). Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* 506, 146–157. doi: 10.1016/j.neucom.2022.07.042

Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *2018 IEEE/CVF conference on computer vision and pattern recognition* (New York: IEEE Press), 6848–6856.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D. (2020). Distance-IoU loss: Faster and better learning for bounding box regression. *Proc. AAAI Conf. Artif. Intell.* 34 (07), 12993–13000. doi: 10.1609/aaai.v34i07.6999

Zheng, Z., Wang, P., Ren, D., Liu, W., Ye, R., Hu, Q., et al. (2021). Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybernetics*. 52 (8), 8574–8586. doi: 10.1109/TCYB.2021.3095305

Zhu, Q., Lei, Y., Sun, X., Guan, Q., Zhong, Y., Zhang, L., et al. (2022). Knowledge-guided land pattern depiction for urban land use mapping: A case study of Chinese cities. *Remote Sens. Environ.* 272, 112916. doi: 10.1016/j.rse.2022.112916