Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.057655

ARTICLE

Lightweight Underwater Target Detection Using YOLOv8 with Multi-Scale Cross-Channel Attention

Xueyan Ding[1,2], Xiyu Chen[1], Jiaxin Wang[1] and Jianxin Zhang[1,2,*]

[1]School of Computer Science and Engineering, Dalian Minzu University, Dalian, 116600, China

[2]Research Center of Multimodal Information Perception and Intelligent Processing, Dalian Minzu University, Dalian, 116600, China

*Corresponding Author: Jianxin Zhang. Email: jxzhang0411@163.com

Received: 23 August 2024 Accepted: 23 October 2024 Published: 03 January 2025

ABSTRACT

Underwater target detection is extensively applied in domains such as underwater search and rescue, environmental monitoring, and marine resource surveys. It is crucial in enabling autonomous underwater robot operations and promoting ocean exploration. Nevertheless, low imaging quality, harsh underwater environments, and obscured objects considerably increase the difficulty of detecting underwater targets, making it difficult for current detection methods to achieve optimal performance. In order to enhance underwater object perception and improve target detection precision, we propose a lightweight underwater target detection method using You Only Look Once (YOLO) v8 with multi-scale cross-channel attention (MSCCA), named YOLOv8-UOD. In the proposed multi-scale cross-channel attention module, multi-scale attention (MSA) augments the variety of attentional perception by extracting information from innately diverse sensory fields. The cross-channel strategy utilizes RepVGG-based channel shuffling (RCS) and one-shot aggregation (OSA) to rearrange feature map channels according to specific rules. It aggregates all features only once in the final feature mapping, resulting in the extraction of more comprehensive and valuable feature information. The experimental results show that the proposed YOLOv8-UOD achieves a mAP50 of 95.67% and FLOPs of 23.8 G on the Underwater Robot Picking Contest 2017 (URPC2017) dataset, outperforming other methods in terms of detection precision and computational cost-efficiency.

KEYWORDS

Deep learning; underwater target detection; attention mechanism

1 Introduction

One of the key factors for success in ocean exploration and the precise autonomous operation of intelligent robots is the availability of highly accurate underwater target detection technology that is capable of real-time processing [1]. However, certain uncontrollable factors render underwater target detection an exceedingly challenging task. For instance, underwater environments are inherently uncertain, and factors such as randomly distributed sand, rocks, and seaweed underwater may interfere with correctly identifying targets. Furthermore, underwater organisms have evolved over time and often have protective colors that blend in with their environment, rendering them visually

challenging to identify and distinguish. Also, because water absorbs and scatters light, underwater environments can lead to low contrast, color distortion, texture degradation, and blurred details in images, directly affecting the image's quality. The combination of these challenges creates extreme background interference, making detecting underwater targets difficult [2,3].

The rapid advancement of deep learning has dramatically accelerated the promotion of object detection technology. Current target detection methods mainly evolve into two categories: region proposal-based methods (two-stage detection methods) [4] and regression-based methods (one-stage detection methods) [5,6]. As a significant example of one-stage detection algorithms, the You Only Look Once (YOLO) series [7] have been extensively researched and widely adopted in computer vision tasks due to its high precision, rapid detection speed, and flexible architecture advantages. The latest YOLOv8 method [8] not only outperforms earlier versions, such as YOLOv5, in terms of detection performance, but also adopts a more optimized model architecture than YOLOv7, which effectively reduces redundant computations [9]. Therefore, it has been successfully applied to underwater vehicle detection and many other fields.

For the underwater target detection task, considering the detection accuracy and processing speed of the model, we chose the YOLOv8 network underwater target detection baseline network. However, YOLOv8 neglects the enhancement and refinement of multi-scale features for densely distributed underwater objects, leading to inaccurate localization and classification [10]. In addition, there are many improvements to YOLOv8 as a baseline that only consider the detection accuracy of the model without considering the mount of computation, which may lead to the loss of the original lightweight advantage of the model.

In this paper, we propose a multi-scale cross-channel attention-guided underwater target detection method, YOLOv8-UOD, to achieve precise and efficient underwater target detection. Multi-scale cross-channel attention uses multi-scale and cross-channel strategies to reduce computational burden and improve detection accuracy. YOLOv8-UOD demonstrates better performance when dealing with common challenges in underwater imaging, such as poor lighting conditions and unclear visual information. The critical contributions of our work are outlined below:

1. This paper proposes YOLOv8-UOD, a lightweight underwater target detection method designed based on the YOLOv8 model. It is designed to meet underwater target detection's increased precision and lightweight needs.

2. We propose a multi-scale cross-channel attention mechanism that incorporates both multi-scale and cross-channel strategies. The multi-scale approach applies fine, medium, and coarse-grained attention to enhance the variety of attention perception. The cross-channel strategy uses channel shuffling and one-shot aggregation cascade methods to reorganize and aggregate all features. We integrate multi-scale cross-channel attention (MSCCA) into the YOLOv8 to achieve the heightened precision requirements of detectors for underwater target detection tasks.

3. The experiments show that the YOLOv8-UOD method demonstrates better performance in the evaluations performed on the Underwater Robot Picking Contest 2017 (URPC2017) dataset, and it has significant advantages in terms of precision and detection speed compared to YOLOv8 and other methods.

2 Related Works

Deep learning-based underwater target detection methods mainly fall into two categories: two-stage and one-stage methods. This section outlines recent progress in these research fields.

2.1 Two-Stage Underwater Target Detection Methods

Two-stage target detection methods decompose the task into two steps: extracting candidate regions followed by classifying and accurately localizing the targets. These detection methods surpass one-stage methods in terms of detection precision and localization precision, but their detection speed is generally lower than that of the one-stage target detection method. Therefore, many researchers have focused on improving classic algorithms like region-based convolutional neural network (R-CNN) and Faster Region-based Convolutional Neural Networks (Faster R-CNN) in recent years [11].

For example, Liu et al. [12] replaced the backbone network of Faster R-CNN with a Transformer structure while introducing a path aggregation network, enabling better integration of deep and shallow feature maps. Zeng et al. [13] introduced the Faster Region-based Convolutional Neural Network with Attention to Object Norms (Faster R-CNN-AON), which integrates an adversarial occlusion network. This approach improves the detection performance when the sample data is limited. To address the overlapping and occlusion issues of underwater organisms, Lin et al. [14] proposed RoIMix, an augmentation technique applied to Faster R-CNN. RoIMix simulates the overlapping and occlusion of underwater organisms by fusing regions of interest extracted from multiple images, thus enabling Faster R-CNN to better detect dense objects.

Additionally, Shi et al. [15] replaced the backbone network of Faster R-CNN with the Residual Neural Network (ResNet). They introduced a Bidirectional Feature Pyramid Network (Bi-FPN) structure while applying the K-means++ clustering algorithm for anchor box generation, enhancing multi-scale feature integration and increasing target localization precision. Wang et al. [16] used Res2Net101 to replace the feature extraction module of Faster R-CNN and introduced Online Hard Example Mining (OHEM) to address the issue of imbalance between positive and negative samples in bounding boxes, enabling more accurate and effective detection of underwater objects. Song et al. [17] introduced Boosting R-CNN, an innovative two-stage detector designed for underwater scenarios, which tackles the difficulties of detecting underwater targets by incorporating uncertainty modelling and mining challenging samples.

2.2 One-Stage Underwater Target Detection Methods

One-stage target detection methods convert the detection problem into an end-to-end regression problem without generating candidate regions. Compared to two-stage detection methods, single-stage detection methods have a slight gap in detection precision. However, they can achieve near-comparable precision while significantly enhancing detection speed, making them more suitable for intelligent devices like underwater robots. The main one-stage target detection methods include the YOLO algorithm [18] and SSD algorithm [19], which have been widely applied in the field of underwater target detection due to their excellent performance and high precision.

SSD is one of the essential representatives of one-stage target detection methods, and many researchers use it for underwater target detection tasks. Ma et al. [20] proposed the MobileNet-SSD, utilizing a 13-layer depthwise separable convolution as the core feature extractor, achieving rapid and accurate detection. Li et al. [21] introduced the XC-SSD model, which incorporates a channel-space attention mechanism to improve the semantic content of high-level feature maps while minimizing false negatives and false positives.

YOLO is one of the most important representatives of one-stage target detection methods and a popular research direction for realizing underwater target detection tasks. Chen et al. [22] integrated channel attention and feature pyramid over a YOLOv4 [23] backbone network to extract and isolate the most significant weighted multi-scale features, subsequently utilizing these features for

underwater bio-detection. Cai et al. [24] combined a weakly supervised learning method based on the YOLOv5 [25] method using two YOLOv5 detectors for training. Using this dual training mechanism not only reduces the consumption of computational resources but also improves recognition precision. Hang et al. [26] suggested the implementation of a global attention mechanism within the YOLOv5 model to strengthen the backbone network's feature extraction ability for essential regions, along with a multi-branch reparameterization structure to enhance the fusion of multi-scale features. Zhang et al. [27] proposed the CGC-YOLO network, which incorporates a cross-stage partial convolution block attention module (CSPCBAM), ghost modules, and cluster non-maximum suppression (Cluster-NMS), achieving efficient processing of blurred objects while maintaining lower computational costs and faster inference speeds. Lou et al. [28] proposed a new down-sampling method and integrated it into YOLOv8, which can retain contextual feature information better. Meanwhile, they improved the feature fusion method of YOLOv8 so that the network maintains more comprehensive information in the feature extraction process.

3 Proposed Method

In this section, we present YOLOv8-UOD, a lightweight underwater object detection method based on the YOLOv8 model. This method improves the precision of underwater object detection and minimizes the computational cost by introducing our multiscale cross-channel attention module.

The network structure of YOLOv8-UOD is depicted in Fig. 1. YOLOv8-UOD consists of three modules: Backbone, Neck, and Head. The Backbone is responsible for feature extraction, capturing essential semantic information. The Neck enhances the expression of semantic information through feature fusion, while the Head is tasked with generating object categories and bounding box positions. Within the Backbone and Neck, we integrate the multi-scale cross-channel attention module, which combines multi-scale attention mechanisms with cross-channel strategies. This allows the approach to precisely identify and locate targets in challenging underwater environments, enhancing the adaptability of the YOLOv8-UOD method for underwater detection tasks.

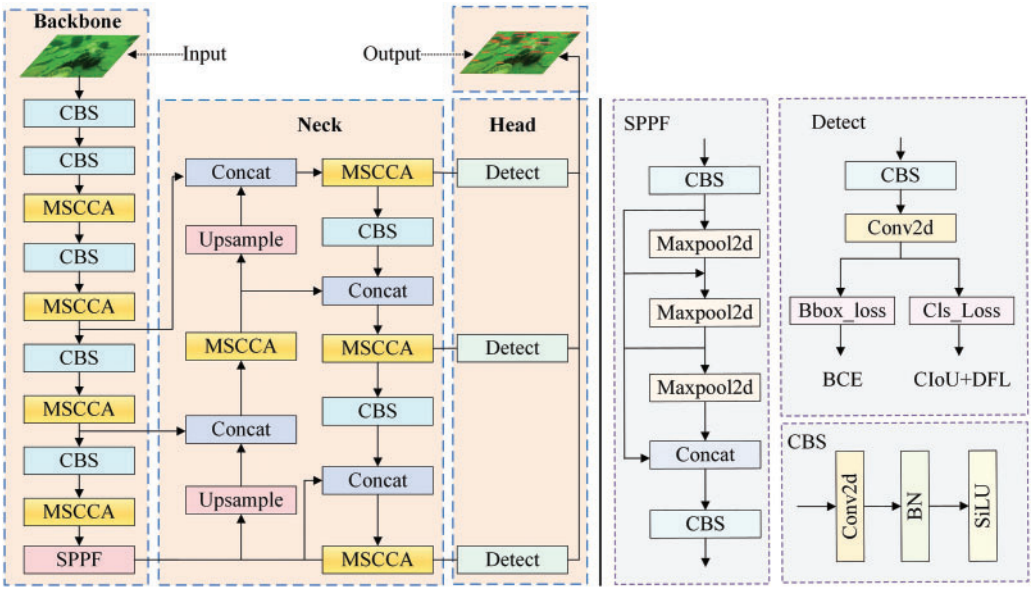

Figure 1: YOLO-UOD network structure

3.1 Multi-Scale Cross-Channel Attention

In the MSCCA module, in order to make it capable of capturing the overall and detailed features of multi-scale underwater targets based on the input information, we introduce multi-scale attention to capture the local and overall details of multi-scale underwater targets, as well as the information between the target and the background. In addition, we incorporate a cross-channel strategy to enhance the information flow between different channels across neighboring feature layers. This involves merging the constant mappings within each block, thereby enhancing the model's adaptability to various features., which helps the precise location information to propagate quickly across the feature layers of both the backbone and neck networks. The structure of MSCCA is shown in Fig. 2.

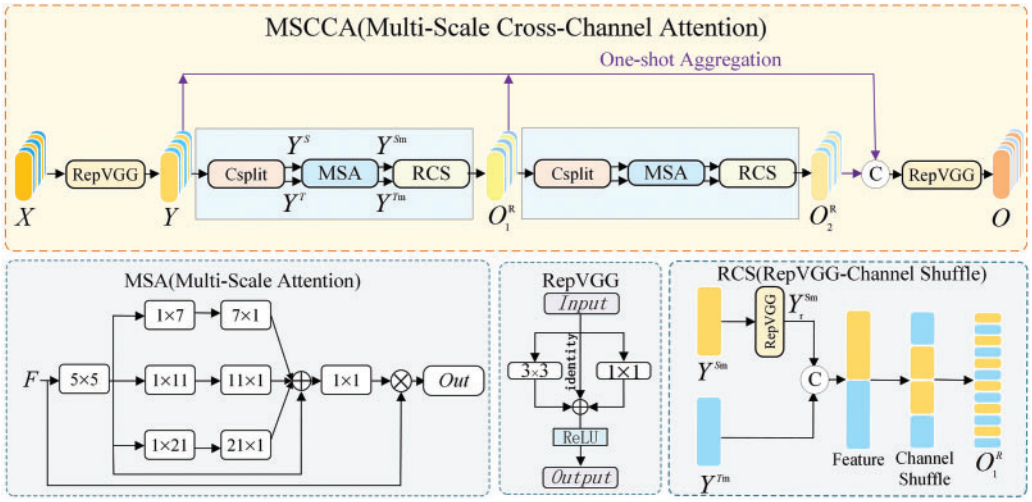

Figure 2: MSCCA network structure diagram

Fig. 2 illustrates the details of the MSCCA's components and specific processes. First, the input image X is passed through a RepVGG [29] to generate the feature map Y. Next, the Csplit module segments the feature map Y into Y^S and Y^T. These segmented feature maps are fed into the MSA module, which performs convolution operations using multiple convolution kernels (e.g., 7×1, 1×7, 1×11, 11×1, 1×21, 21×1) and combines these features through a weighting mechanism to merge these features and generate the output Y^{Sm} and Y^{Tm}. Different sizes of convolution kernels in multiscale convolution are designed to handle the irregular shapes, localized details, and long-edge structures of underwater targets. Together, these convolutions improve the model's feature extraction capabilities and enhance the interaction and refinement of features across different scales.

Next, the features go back into the RCS [30] module to further realize the feature extraction and channel interaction operations. The RCS module consists of the RepVGG module and channel shuffling, where the RepVGG consists of a 3×3 convolution layer and a 1×1 convolution layer along with the ReLU activation function, and the channel shuffling rearranges the output of the RepVGG. In practice, the RCS module receives features Y^{Sm} and Y^{Tm} from the MSA module. Y^{Sm} is then processed by a RepVGG module to generate the intermediate feature Y_r^{Sm}. Then, Y^{Tm} and Y_r^{Sm} are channel aggregated to form new features. Next, a channel shuffling operation rearranges the channels of the features to output the final feature O_1^R. Channel shuffling ensures the global interaction of feature information by rearranging channels to ensure computational efficiency and enhance feature

fusion. Utilizing this design allows the model to comply more with the lightweight requirements of underwater target detection.

Subsequently, O_1^R is again subjected to channel splitting, and the steps of multi-scale attention, RepVGG, and channel shuffling are repeated to obtain O_2^R. Finally, through a one-time aggregation method, Y, O_1^R, and O_2^R are concatenated and passed through a RepVGG module for the final fusion and integration to produce the final output result O.

Multi-scale attention effectively captures features at different scales and is particularly suitable for detecting underwater irregularly shaped targets. Channel shuffling enhances the flow of information between feature channels and suppresses redundant information, thus optimizing the detection efficiency of the model. Combining the two, the YOLOv8 demonstrates enhanced detection capabilities and robustness in complex underwater environments.

3.2 Multi-Scale Attention

For enhancement and refinement between multi-scale features of underwater objects, we implemented multi-scale attention using only some simple convolutional structures to improve the model's ability to capture and extract multi-scale features [31]. The composition of the multi-scale attention module consists of three parts. First, the module uses a 5×5 convolution to perform the convolution operation to converge the local information in order to ensure the spatial continuity and richness of the features. Secondly, as shown by the MSA module in Fig. 2, we used multi-branch strip convolution. Three pairs of 1×7 and 7×1, 1×11 and 11×1, and 1×21 and 21×1 convolution are included here to form a multiscale convolution. During training, the network assigns a learnable weight to the outputs of these convolutional operations, and by learning the weights of different convolutional operations, the network can better fuse features at different scales and suppress redundant information. There are two main reasons for using multi-branch strip convolution instead of standard 2D convolution. On the one hand, this is more in line with the lightweight design of the model; for example, a standard 2D convolution of 7×7 can be approximated by a pair of 7×1 and 1×7 convolution, which reduces computational complexity. On the other hand, in the underwater target detection application scenarios, there are a large number of striped and irregularly shaped objects (e.g., sea cucumbers, starfish, etc.), and strip convolution can better capture the edge and directional features of these objects., make up for the shortcomings of mesh convolution, and improve the network model's ability to extract features [32]. Lastly, the 1×1 convolution is used to capture the interdependencies among various channels, and its output is directly used as the attention weights to reweight the module's inputs to enhance essential features dynamically [33]. MSA can be expressed as:

$$Att = Conv_{1\times1}(\sum_{i=0}^{3} Scale_i(DWConv(Conv_{5\times5}(F)))), \tag{1}$$

$$Out = Att \otimes F. \tag{2}$$

where Eq. (1) showcases the computation process of the attention map, which begins by applying depth-wise convolution ($DWConv$) to the input feature F, followed by a weighted sum operation through the $Scale_i$ branches, and finally integrating them through the 1×1 convolutional layer. Here, $Scale_i$ denotes different scale branches that correspond to kernels of different sizes to capture features of different scales. The output feature is calculated using Eq. (2), which $\otimes$ denotes element-wise matrix multiplication. The attention map is applied element-wise to the input feature through multiplication, resulting in a weighted output feature that further strengthens the model's emphasis on significant features.

3.3 Cross-Channel Operations

To promote interaction among different features and improve the model's generalization capability, we introduce the idea of cross-channel in the MSCCA module. In the MSCCA module, the input feature map X with dimensions $C \times H \times W$ (C denotes the number of channels, while H and W indicate the height and width of the feature map, respectively) is accepted as input, and further feature extraction is performed by RepVGG to obtain the feature map Y. Subsequently, Y is split into two sub-features Y^S and Y^T, providing support for cross-channel information interaction. Here, the input feature map Y and the two sub-features Y^S and Y^T can be represented as $Y \in \mathbb{R}^{C\times H\times W}$, $Y^S \in \mathbb{R}^{C\times H\times W}$, and $Y^T \in \mathbb{R}^{C\times H\times W}$, respectively. The channel splitting operation can be expressed as:

$$Y^S,\ Y^T = Csplit\,(Y) \tag{3}$$

where $Y = [Y_1, \ldots, Y_C]$, $Y^S = [Y_1^S, \ldots, Y_{C/2}^S]$, and $Y^T = [Y_1^T, \ldots, Y_{C/2}^T]$. Subsequently, Y^S and Y^T are simultaneously passed through the multi-scale attention module. The formula for the multi-scale feature extraction stage can be expressed as:

$$Y^{Sm} = MSA\left(Y^S\right) \tag{4}$$

$$Y^{Tm} = MSA\left(Y^T\right) \tag{5}$$

Here after processing by the multi-scale attention module, each sub-feature is able to capture more detailed contextual information at different spatial scales. This allows the model to understand and extract both local and global dependencies of features, thereby enhancing the model's perception of various scale structures in images.

Next, in order to preserve the original information to enrich the features in the cross-channel operation, we do not perform any operation on Y^{Tm}. Meanwhile, Y^{Sm} is transferred to the RepVGG block for further feature extraction to obtain Y_r^{Sm}. This design ensures that the network retains some of the original signal while increasing complexity, enhancing the generalization of the model. The formula can be expressed as:

$$O_1^R = RepVGG\left(Y_r^{Sm}\right) + Y^{Tm} \tag{6}$$

Here, RepVGG is a Visual Geometry Group (VGG) based on structural reparameterization, which contains 3×3, 1×1 convolution and an identity branch. Y_r^{Sm} denotes the result after the RepVGG module. O_1^R denotes the result of the cascade of Y_r^{Sm} and Y^{Tm}.

Next, Y^{Tm} and Y_r^{Sm} are combined to form a fused feature O_1^R. This is followed by a Channel Shuffle operation, which promotes information exchange between different channels by rearranging their order, further improving the network's capability for feature extraction and representation across multiple levels.

Cross-channel cooperation ensures effective feature fusion at deeper levels of the model, even after feature segmentation and independent processing by channel shuffling. This not only provides rich feature information for subsequent detection tasks but also improves the generalization capability of the model.

4 Experiments and Results

To verify the effectiveness of our approach, we conduct comprehensive detection experiments, followed by an in-depth analysis of the results. The experimental evaluation assesses detection precision using mAP (mean Average Precision) calculated at specified IoU (Intersection over Union)

thresholds. We utilize mAP50 (mAP at IoU = 0.5) to measure algorithm performance, while mAP50-95 (mAP across IoU thresholds from 0.5 to 0.95 in 0.05 increments) is the challenge metric. Alongside precision, recall is evaluated to measure the model's ability to detect positive cases correctly. The network's size and computational complexity are quantified using Params (parameters) and FLOPs (floating-point operations).

4.1 Experimental Details

In this study, we utilized the publicly available The Underwater Robot Picking Contest 2017 (URPC2017) [34] dataset, which was captured in a real underwater environment by professionals using an underwater video camera and is widely used for underwater object detection tasks. The dataset was downloaded from the official repository and stored in a structured directory format for easy access during the training and evaluation phases. As shown in Fig. 3, the dataset consists of 18,638 images, each with a resolution of 720 × 405, containing annotations for multiple underwater object classes such as sea urchin, sea cucumber, and scallop. Each image is annotated with bounding boxes and category labels to accurately indicate the location of the objects.

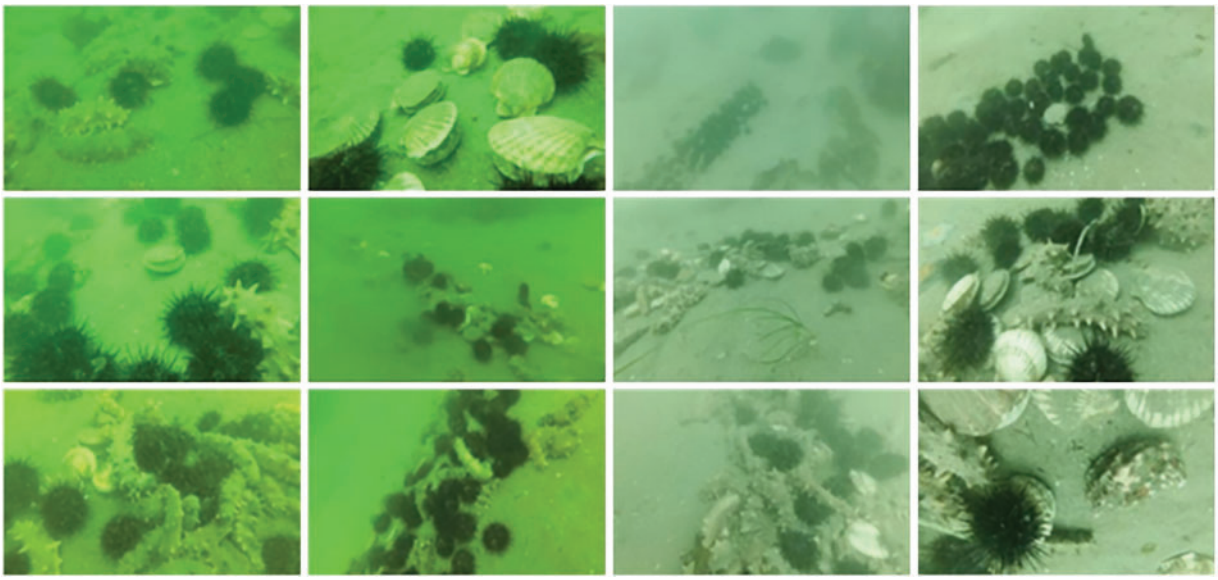

Figure 3: Samples from the URPC2017 dataset

For the experiment, the dataset was divided into training and testing sets, with the training images into training and validation sets in the ratio of 8:2. We also implemented data augmentation strategies, including random scaling, cropping, rotation, and color adjustments, to enhance the model's robustness throughout the training process. The raw data of the dataset is preprocessed, resized to 640 × 640 pixels, and normalized to ensure consistency across all images before it is fed into the model for training and evaluation.

The approach utilizes the efficient PyTorch framework, operating within the Python 3.8.0 environment. Table 1 describes the detailed experimental setup.

Table 1: Experimental environment

Experimental details	Detailed information
Graphics processing unit	NVIDIA RTX3090Ti (24 GB)
Processing unit	Intel (R) Xeon (R) Silver 4210R CPU @ 2.40 GHz
Operating system	Ubuntu 20.04.5
Random access memory	64 GB

During the model training phase, we set 300 epochs for the training period and implemented an early stopping mechanism, which terminates training prematurely if no performance advancement is observed over 50 epochs. For all the experiments, we use a uniform hyperparameter, and the specific experimental settings are detailed in Table 2.

Table 2: Experimental parameters

Parameter	Setting
Batch size	8
Optimizer	Stochastic Gradient Descent (SGD)
Initial learning rate	0.01
Minimum learning rate	0.0001 (1% of the initial learning rate)
Learning rate decay method	Cosine annealing

4.2 Ablation Experiments

To verify the unique contribution of each module in YOLOv8-UOD, we propose ablation experiments. We conducted experiments where the model is integrated with all modules, i.e., including the MSCA module, Channel Shuffle, and RepVGG. Then, we conducted ablation experiments between modules by gradually removing different modules to analyze each module's role in the model and gain insights into each part's impact on overall performance. The experimental results are summarized in Table 3.

Table 3: Ablation experiments of different modules

Methods	Params	Precision	Recall	mAP50	FLOPs
Full (All modules)	7.88 M	**96.19%**	92.12%	**95.67%**	23.8 G
-MSA	7.88 M	95.78%	**92.26%**	95.56%	23.8 G
-Channel Shuffle	7.88 M	96.16%	92.10%	95.63%	23.8 G
-RepVGG	7.88 M	96.17%	92.10%	95.63%	**23.7 G**

In the complete configuration, utilizing MSA, RepVGG, and Channel Shuffle, the model exhibits optimal performance, achieving a mAP50 of 95.67% and a recall of 92.12% while maintaining FLOPs at 23.8 G. This result demonstrates the importance of using a combination of these modules to improve the model's detection precision and processing speed. MSA dynamically weights the feature map to

emphasize salient regions, enhancing the network's adaptability to image features of different scales and complexities. The removal of MSA results in a slight decrease in mAP50 to 95.51%, with recall slightly increasing to 92.26%, which shows that it is optimized for multi-scale feature extraction and improving detection precision.

Channel Shuffle enhances the flow of information between channels by rearranging the order of channels in convolution layers, improving feature extraction capability and generalization of the network. Its removal leads to a slight performance decrease in mAP50 from 95.67% to 95.63%, and recall slightly decreases to 92.10%, indicating its positive impact on optimizing the model's information processing and feature learning processes. These observations confirm the importance of these components in enhancing the model's overall performance. The proposed modules all improve the precision and also achieve good performance compared to YOLOv8.

4.3 Experimental Analysis

To further validate the performance benefits of the attention module in underwater object detection, we compared YOLOv8-UOD with the original YOLOv8 network architecture and other versions of the YOLO series, including YOLOv5 and YOLOv7. The experimental results are presented in Table 4.

Table 4: The comparison of the proposed method with other methods

Methods	Params	Precision	Recall	mAP50	mAP50-95	FLOPs
YOLOv5 [25]	**2.39 M**	95.61%	91.03%	94.82%	69.13%	**7.8 G**
YOLOv7 [35]	35.48 M	**97.87%**	**94.46%**	95.59%	**74.32%**	105.1 G
YOLOv8 [8]	2.88 M	95.50%	91.38%	95.12%	70.10%	8.2 G
YOLOv8-UOD	7.88 M	96.19%	92.12%	**95.67%**	71.76%	23.8 G

From Table 4, it is evident that YOLOv8-UOD excels in precision, mAP50, recall, and mAP50-95 metrics, outperforming all other outperforms all other versions of the YOLO method. Although the YOLOv7 model has a slight edge in precision and recall, there's a reason for this: it's the positive impact of its design with an anchored frame, but it also makes it somewhat limited in processing speed. The anchor strategy constructs many anchor boxes, directly allowing the network to carry out object classification and bounding box regression. Nevertheless, this method necessitates configuring multiple hyperparameters, including scale and aspect ratio, which are difficult to optimize and may affect detection accuracy. Moreover, it results in numerous overlapping boxes, escalating computational demands. Experimental data in Table 4 also shows that YOLOv7's Params and FLOPs are nearly five times larger than those of YOLOv8-UOD.

Fig. 4 shows the precision-recall curves for YOLOv8-UOD compared to YOLOv8. The coloured lines represent the precision-recall curves for each category, and the dark blue lines represent the average precision-recall curves for all categories. The phenomenon illustrated in Fig. 4 shows that as the recall increases, the model causes a significant increase in the number of false positives to detect all the positive cases as much as possible, which leads to a significant decrease in the precision or even close to zero. This is due to the balanced between recall and precision. The figure reveals a higher precision-recall curve for our YOLOv8-UOD method, suggesting that it can achieve higher precision while maintaining the same recall. This confirms the stability of our method in complex underwater environments, as well as its adaptability and efficiency in handling underwater target detection tasks.

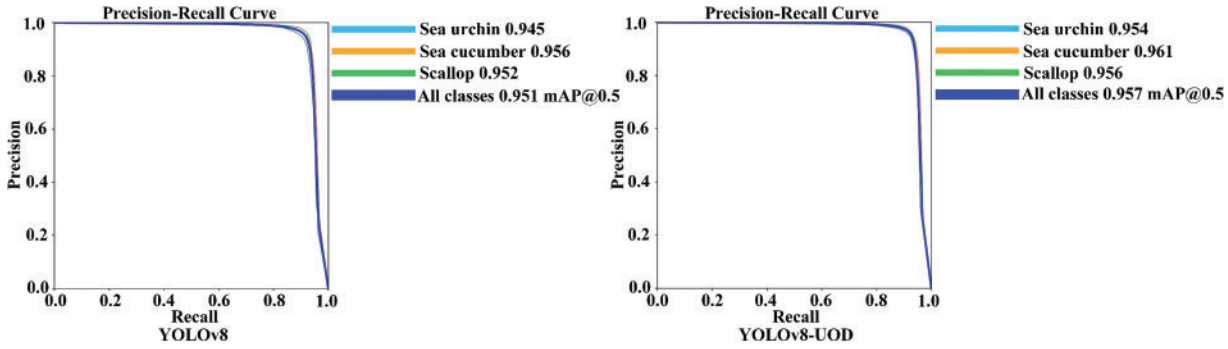

Figure 4: Precision-recall curves of YOLOv8-UOD and YOLOv8 on URPC2017 dataset

To validate the effectiveness of the proposed MSCCA, we incorporated a vision transformer with bi-level routing attention (BiFormer) [36] and deformable large kernel attention (DLKA) [37], which are recently proposed effective modules, into YOLOv8 for comparison, respectively. The experimental results are presented in Table 5.

Table 5: Comparing several modules to enhance YOLOv8

Methods	Params	Precision	Recall	mAP50	mAP50-95	FLOPs
YOLOv8	**2.88 M**	95.56%	91.38%	95.12%	70.13%	**8.2 G**
YOLOv8+BiFormer	2.89 M	96.18%	90.99%	95.15%	70.01%	8.9 G
YOLOv8+DLKA	4.36 M	95.94%	91.45%	95.21%	70.70%	13.4 G
YOLOv8-UOD	7.88 M	**96.19%**	**92.12%**	**95.67%**	**71.76%**	23.8 G

In this study, we integrate the BiFormer and DLKA modules into the YOLOv8 and determine the impact that both approaches have on YOLOv8 by conducting experiments with the same training strategy. While the BiFormer module slightly increases mAP50 (from 95.12% to 95.15%), it also resulted in a decrease in the mAP50-95 and recall, which reflects a potential problem with optimizing precision in specific scenarios that may affect generalization ability. The DLKA module improves mAP50, mAP50-95, and recall to 95.21%, 70.70%, and 91.45%, respectively, demonstrating that the large convolutional kernel combined with deformable convolution presented in this module can enhance the performance of YOLOv8 to a certain extent.

Although YOLOv8 is improved by integrating both BiFormer and DLKA modules, the positive impact of these two methods is not as large as that of our proposed YOLOv8-UOD. YOLOv8-UOD, with the introduction of the multi-scale cross-channel module, not only achieves a mAP50 of 95.67%, but also improves mAP50-95 to 71.76% and recall to 92.12%. Our proposed method's false detection rate is lower than the other methods in the table, proving that it is suitable for complicated underwater environments. Despite the increase in Params and FLOPs, the model reaches a favorable balance between processing speed and precision, making it a more compelling choice for underwater target detection.

We applied the weights of the trained model to the test set of images from the URPC2017 dataset. Fig. 5 presents the original and manually labeled images from the test set and the results obtained using the YOLOv8-UOD model. These images include challenging environments such as turbid water,

overlapping target objects, and occlusions of unknown items. Compared to the original annotated images, the YOLOv8-UOD model achieves accurate detection results without deviations from the annotations and demonstrates the capability to identify targets that lack prior annotation information.

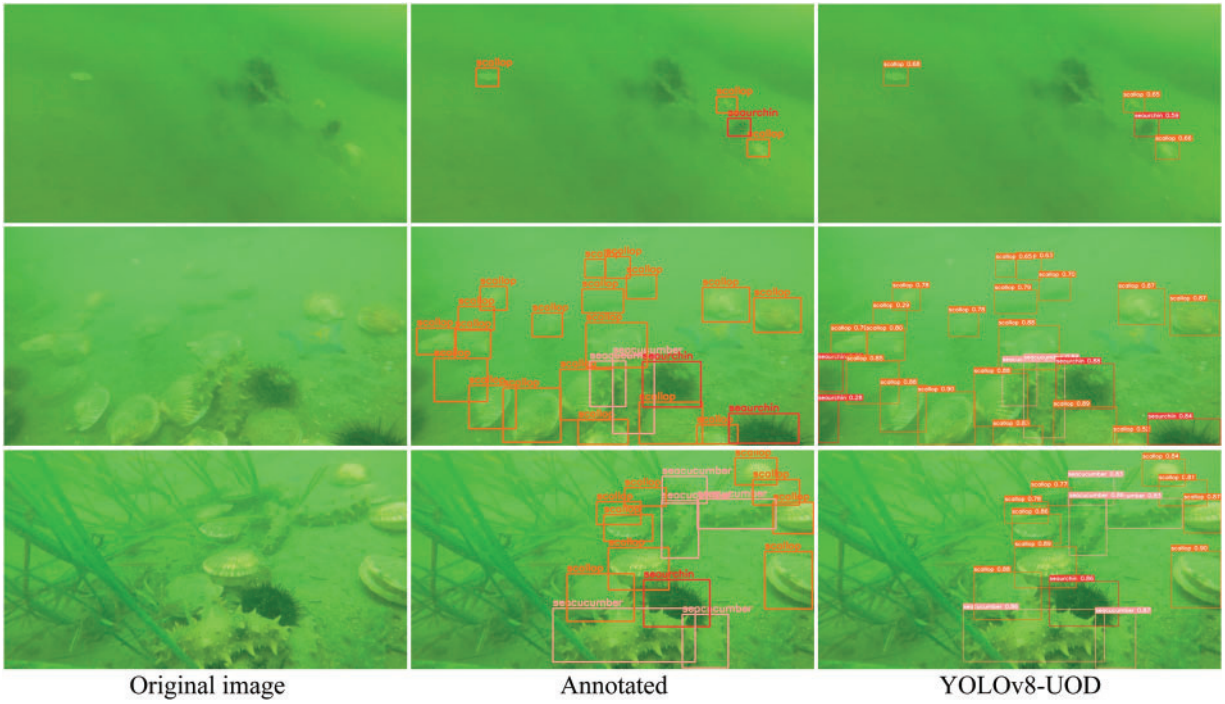

Figure 5: Detection results and labeled images

5 Conclusion

The YOLOv8-UOD proposed in this paper is a lightweight underwater object detection method that effectively enhances the precision of target detection in complex underwater environments. The proposed MSCCA mechanism captures features at various scales through an MSA module, providing the network with more comprehensive visual information. The cross-channel strategy effectively integrates cross-channel information through RCS and OSA, extracts more comprehensive and valuable feature information, and significantly improves network performance and computational efficiency. The experimental results show that the method improves the precision of underwater target detection, and achieves a suitable balance in the number of parameters, computation, and memory consumption.

In future research, we intend to advance and perfect the YOLOv8-UOD method, striving for superior performance and efficiency. Achieving this will necessitate further exploration of strategies for model compression, designing more lightweight network architectures, and adapting them for deployment on small embedded devices without compromising precision. For instance, model pruning can decrease the model's size by removing redundant neurons and connections in the network while maintaining performance. In addition, knowledge distillation is an effective strategy for training compact "student" models to imitate the behavior of larger "teacher" models, thus achieving similar performance with limited resources. We expect to develop accurate and efficient underwater target detection models by combining these approaches.

Acknowledgement: The authors wish to express their sincere gratitude to the anonymous reviewers and the editor, whose valuable suggestions have significantly improved the quality of this manuscript.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China Grants 62402085, 61972062, and 62306060, the Liaoning Doctoral Research Start-Up Fund 2023-BS-078, the Dalian Youth Science and Technology Star Project 2023RQ023, and the Liaoning Basic Research Project 2023JH2/101300191.

Author Contributions: The authors confirm their contribution to the paper as follows: study conception and design: Xiyu Chen, Xueyan Ding; data collection: Xiyu Chen, Jiaxin Wang; analysis and interpretation of results: Xiyu Chen, Xueyan Ding, Jiaxin Wang; draft manuscript preparation: Xiyu Chen, Xueyan Ding, Jianxin Zhang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The authors do not have permission to share the data.

Ethics Approval: Not applicable.

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

References

[1] X. Shen, X. Sun, and H. Wang, "Multi-dimensional, multi-functional and multi-level attention in YOLO for underwater object detection," *Neural Comput. Appl.*, vol. 35, no. 27, pp. 19935–19960, 2023. doi: 10.1007/s00521-023-08781-w.

[2] S. Xu, M. Zhang, W. Song, H. Mei, Q. He and A. Liotta, "A systematic review and analysis of deep learning-based underwater object detection," *Neurocomputing*, vol. 527, pp. 204–232, 2023. doi: 10.1016/j.neucom.2023.01.056.

[3] J. Gao, Y. Zhang, X. Geng, H. Tang, and U. A. Bhatti, "PE-Transformer: Path enhanced transformer for improving underwater object detection," *Expert. Syst. Appl.*, vol. 246, Jan. 2024, Art. no. 123253. doi: 10.1016/j.eswa.2024.123253.

[4] O. Hmidani and E. M. Ismaili Alaoui, "A comprehensive survey of the R-CNN family for object detection," in *2022 5th Int. Conf. Adv. Commun. Technol. Netw. (CommNet)*, Marrakech, Morocco, 2022, pp. 1–6. doi: 10.1109/CommNet56067.2022.9993862.

[5] H. Tang, A. Peng, D. Zhang, T. Liu, and J. Ouyang, "SSD real-time illegal parking detection based on contextual information transmission," *Comput. Mater. Contin.*, vol. 62, no. 1, pp. 293–307, 2020. doi: 10.32604/cmc.2020.06427.

[6] M. Hussain, "YOLOv1 to v8: Unveiling each variant—A comprehensive review of YOLO," *IEEE Access*, vol. 12, pp. 42816–42833, 2024. doi: 10.1109/ACCESS.2024.3378568.

[7] J. Terven, D. -M. Córdova-Esparza, and J. -A. Romero-González, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Mach. Learn. Knowl. Extraction*, vol. 5, no. 4, pp. 1680–1716, 2023. doi: 10.3390/make5040083.

[8] R. Varghese and M. Sambath, "YOLOv8: A novel object detection algorithm with enhanced performance and robustness," in *Proc. Int. Conf. Adv. Intell. Syst. Comput.*, Chennai, India, 2024, pp. 1–6. doi: 10.1109/ADICS58448.2024.10533619.

[9] Y. Zhao, F. Sun, and X. Wu, "FEB-YOLOv8: A multi-scale lightweight detection model for underwater object detection," *PLoS One*, vol. 19, no. 9, Sep. 2024, Art. no. e0311173. doi: 10.1371/journal.pone.0311173.

[10] R. Jia, B. Lv, J. Chen, H. Liu, L. Cao and M. Liu, "Underwater object detection in marine ranching based on improved YOLOv8," *J. Mar. Sci. Eng.*, vol. 12, no. 1, p. 55, 2024. doi: 10.3390/jmse12010055.

[11] G. Priyadharshini and D. R. Judie Dolly, "Comparative investigations on tomato leaf disease detection and classification using CNN, R-CNN, Fast R-CNN and faster R-CNN," in *Proc. Int. Conf. Adv. Comput. Commun. Syst.*, Coimbatore, India, 2023, pp. 1540–1545. doi: 10.1109/ICACCS57279.2023.10112860.

[12] J. Liu, S. Liu, S. Xu, and C. Zhou, "Two-stage underwater object detection network using swin transformer," *IEEE Access*, vol. 10, pp. 117235–117247, 2022. doi: 10.1109/ACCESS.2022.3219592.

[13] L. Zeng, B. Sun, and D. Zhu, "Underwater target detection based on faster R-CNN and adversarial occlusion network," *Eng. Appl. Artif. Intell.*, vol. 100, no. 4, 2021, Art. no. 104190. doi: 10.1016/j.engappai.2021.104190.

[14] W. -H. Lin, J. -X. Zhong, S. Liu, T. Li, and G. Li, "ROIMIX: Proposal-fusion among multiple images for underwater object detection," in *IEEE Int. Conf. Acoustics Speech Signal Process.*, Barcelona, Spain, 2020, pp. 2588–2592. doi: 10.1109/ICASSP40776.2020.9053829.

[15] P. Shi, X. Xu, J. Ni, Y. Xin, W. Huang and S. Han, "Underwater biological detection algorithm based on improved faster-RCNN," *Water*, vol. 13, no. 17, 2021, Art. no. 2420. doi: 10.3390/w13172420.

[16] H. Wang and N. Xiao, "Underwater object detection method based on improved faster RCNN," *Appl. Sci.*, vol. 13, no. 4, 2023, Art. no. 2746. doi: 10.3390/app13042746.

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

[18] X. Hu *et al.*, "Real-time detection of uneaten feed pellets in underwater images for aquaculture using an improved YOLO-V4 network," *Comput. Electron. Agric.*, vol. 185, 2021, Art. no. 106135. doi: 10.1016/j.compag.2021.106135.

[19] Z. Jiang and R. Wang, "Underwater object detection based on improved single shot MultiBox detector," presented at the Proc. 2020 3rd Int. Conf. Algorithms Comput. Artif. Intell., Sanya, China, 2021. doi: 10.1145/3446132.3446170.

[20] K. Ma, B. Huang, and H. Yin, "Underwater sea cucumbers detection based on improved SSD," in *Proc. IEEE Int. Conf. Power, Intell. Comput. Syst.*, Shenyang, China, 2019, pp. 343–347. doi: 10.1109/ICPICS47731.2019.8942503.

[21] Q. Li, M. Joo Er, L. Li, J. Chen, and J. Wu, "Underwater object detection based on improved SSD with convolutional block attention," in *Proc. Int. Conf. on Intell. Auto. Syst.*, Dalian, China, 2022, pp. 37–42. doi: 10.1109/ICoIAS56028.2022.9931319.

[22] L. Chen *et al.*, "Underwater object detection using Invert Multi-Class Adaboost with deep learning," in *Proc. Int. Joint Conf. Neural Netw.*, Glasgow, UK, 2020, pp. 1–8. doi: 10.1109/IJCNN48605.2020.9207506.

[23] A. Bochkovskiy, C. -Y. Wang, and H. -Y. M. J. A. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020. doi: 10.48550/arXiv.2004.10934.

[24] S. Cai, G. Li, and Y. J. C. E. E. Shan, "Underwater object detection using collaborative weakly supervision," *Comput. Electr. Eng.*, vol. 102, 2022, Art. no. 108159. doi: 10.1016/j.compeleceng.2022.108159.

[25] G. Jocher *et al.*, "Yolov5," 2021. Accessed: Aug. 15, 2024. [Online]. Available: https://github.com/ultralytics/yolov5

[26] Z. Hang, L. Fan, K. Ping, G. Xiaofeng, H. Mingyun and T. Heng, "Small object detection algorithm based on context information and attention mechanism," in *Proc. Int. Comput. Conf. Wavelet Active Med. Technol. Inform. Process.*, Chengdu, China, 2022, pp. 1–6. doi: 10.1109/ICCWAMTIP56608.2022.10016586.

[27] Z. Zhang, Q. Tong, and X. Huang, "An efficient YOLO network with CSPCBAM, ghost, and cluster-NMS for underwater target detection," *IEEE Access*, vol. 12, pp. 30562–30576, 2024. doi: 10.1109/ACCESS.2024.3368878.

[28] H. Lou *et al.*, "DC-YOLOv8: Small-size object detection algorithm based on camera sensor," *Electronics*, vol. 12, no. 10, 2023, Art. no. 2323. doi: 10.3390/electronics12102323.

[29] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding and J. Sun, "RepVGG: Making VGG-style ConvNets great again," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Nashville, TN, USA, 2021, pp. 13728–13737. doi: 10.1109/CVPR46437.2021.01352.

[30] M. Kang, C. M. Ting, F. F. Ting, and R. C. W. Phan, "RCS-YOLO: A fast and high-accuracy object detector for brain tumor detection," in *Proc. Med. Image Comput. Comput. Assist. Interv.*, Vancouver, BC, Canada, 2023, pp. 600–610. doi: 10.1007/978-3-031-43901-8_57.

[31] M. H. Guo, C. Z. Lu, Q. Hou, Z. Liu, M. M. Cheng and S. M. Hu, "SegNeXt: Rethinking convolutional attention design for semantic segmentation," *Adv. Neural Inform. Process. Syst.*, vol. 35, pp. 1140–1156, 2022. doi: 10.48550/arXiv.2209.08575.

[32] Q. Hou, L. Zhang, M. -M. Cheng, and J. Feng, "Strip pooling: Rethinking spatial pooling for scene parsing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2020, pp. 4002–4011. doi: 10.1109/CVPR42600.2020.00406.

[33] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large Kernel matters—Improve semantic segmentation by global convolutional network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 1743–1751. doi: 10.1109/CVPR.2017.189.

[34] National Natural Science Foundation of China. "Underwater robot picking contest," 2017. Accessed: Sep. 15, 2024. [Online]. Available: http://2017.urpc.org.cn/index.html

[35] 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. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Vancouver, BC, Canada, 2023, pp. 7464–7475. doi: 10.1109/CVPR52729.2023.00721.

[36] L. Zhu, X. Wang, Z. Ke, W. Zhang, and R. Lau, "BiFormer: Vision transformer with bi-level routing attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, Vancouver, BC, Canada, 2023, pp. 10323–10333. doi: 10.1109/CVPR52729.2023.00995.

[37] R. Shu, L. Chen, L. Su, T. Li, and F. Yin, "DLCH-YOLO: An object detection algorithm for monitoring the operation status of circuit breakers in power scenarios," *Electronics*, vol. 13, no. 19, Oct. 2024, Art. no. 3949. doi: 10.3390/electronics13193949.