Multimedia Tools and Applications (2024) 83:20003–20032
https://doi.org/10.1007/s11042-023-16311-y

Criss-cross global interaction-based selective attention in YOLO for underwater object detection

Xin Shen[1] · Huibing Wang[1] · Yafeng Li[1] · Tianzhu Gao[1] · Xianping Fu[1,2]

Received: 3 September 2022 / Revised: 11 June 2023 / Accepted: 12 July 2023 /
Published online: 29 July 2023

Abstract

With the development of computer vision, object detection has attracted wide attention and achieved exciting results in most situations. However, facing underwater environments, object detection's performance degrades severely due to multiple ineluctable factors, including poor underwater imaging quality, underwater objects with protective colors, etc. These lead to strong interference of underwater backgrounds and the weak discriminability of underwater object features, which make underwater object detection become an extremely challenging task and cry out for reliable solutions. In order to reduce the underwater background interference and improve underwater object perception, we first propose the criss-cross global interaction strategy (CGIS). CGIS consists of two criss-cross structures, where feature decomposition and feature extraction are performed sequentially according to different criss-cross shapes in each structure. For information interaction, our strategy simultaneously avoids the destruction of direct information correspondence and the lack of global information interaction. According to different parameter allocation strategies, CGIS is further divided into standard criss-cross global interaction strategy (SCGIS) and efficient criss-cross global interaction strategy (ECGIS). We then design the criss-cross global interaction-based selective attention in different target dimensions. Our selective attention efficiently perceives global underwater information and rationally allocates precious computing resources to important underwater regions. We finally combine the designed selective attention with YOLO detectors, where attention modules are added to both ends of the feature fusion. The experimental results show that our work makes important progress in achieving efficient underwater object detection. Our selective attention shows good robustness in various YOLO detectors and exhibits ideal generalization in different detection tasks.

Keywords Underwater object detection · Information interaction · Selective attention · YOLO detectors

✉ Xianping Fu
fxp@dlmu.edu.cn

Extended author information available on the last page of the article

1 Introduction

Underwater object detection is an interesting and challenging task in computer vision, which is a basic premise for ocean exploration and autonomous grasping by underwater robots. Due to the absorption and scattering of light by water, the imaging quality of the underwater image is poor, such as blur, low contrast, color distortion, and so on. In order to avoid attacks, underwater objects have evolved protective colors, which can make full use of complex underwater environments such as sand or rocks as shelters to hide themselves. The above phenomena lead to strong interference of underwater backgrounds and the weak discriminability of underwater object features, which greatly aggravate the difficulty of underwater object detection. Although popular object detection algorithms [4, 30, 50, 51] using deep learning have achieved encouraging results, it is not ideal to apply these algorithms directly to the underwater environment. Obviously, common methods to improve the performance of neural networks, such as directly increasing the depth [13], width [34], and cardinality [41] in the network, cannot effectively solve the problem of underwater object detection.

At present, underwater detection algorithms tend to improve the underwater detection performance from two different perspectives: 1. Data enhancement techniques [21, 45], such as splicing and overlapping, are adopted to improve the dataset quality. 2. Network construction techniques [43, 48], such as residual connection and feature pyramid, are used to improve the network performance. This simple performance gain is mainly due to the improvement of dataset quality and network performance. The core problems of strong underwater background interference and weak underwater object perception have not been solved effectively. In practical underwater applications, underwater detection algorithms [6, 22, 33, 42] still have some problems, such as poor robustness and weak generalization.

It is worth noting that the attention mechanism has been widely applied in computer vision recently [1, 5, 7, 25, 37, 52], which can extract more valuable information from massive information. In order to reduce the underwater background interference and improve underwater object perception, we plan to focus on the attention mechanism in this paper.

At present, various modules with the attention mechanism have been proposed in computer vision, such as the Squeeze-and-Excitation module (SEM) [16], Bottleneck attention module (BAM) [26], Convolutional block attention module (CBAM) [40], Efficient channel attention module (ECAM) [39], Coordinate attention module (CoAM) [14], Spatial group-wise enhance module (SGEM) [20], Style-based recalibration module (SRM) [19], Frequency channel attention module (FCAM) [27], Shuffle attention module (ShAM) [49], Criss-Cross attention module [15] and so on. According to the different target dimensions, these attention modules are divided into spatial attention modules [15, 20], channel attention modules [16, 19, 27, 39], and hybrid attention modules [14, 26, 40, 49]. It is worth noting that the information interaction is the most important process in attention modules, which is responsible for capturing global or local dependencies in the target dimension. The attention activation is carried out on the basis of the interactive result. The quality of the interactive result directly determines the performance of the attention module. Although major breakthroughs have been made in various attention modules, there are still some problems.

First, some attention modules use the dimensionality reduction interaction strategy [14, 16, 26, 27, 40]. This strategy refers to the use of a bottleneck structure in information interaction, which first compresses the input channel dimension and then expands the output channel dimension. Although these modules can perceive global information, they cause confusion in the correspondence between information. Second, some attention modules use the local interaction strategy [19, 20, 26, 39, 40, 49]. This strategy refers to the use of local thinking in information interaction, which focuses on capturing local dependencies in the spatial dimension or channel dimension. Although these modules can ensure the direct correspondence of information, they ignore the importance of perceiving global information. In summary, the destruction of direct information correspondence and the lack of global information interaction will all reduce the quality of information interaction, thereby reducing attention performance. The negative effects of the above problems will be magnified in the complex underwater environment.

In this paper, our work is dedicated to solving the problems of strong underwater background interference and weak underwater feature discriminability. We first propose the criss-cross global interaction strategy (CGIS), which can avoid the deficiencies brought by the dimensionality reduction interaction strategy and the local interaction strategy in information interaction. The proposed strategy is mainly composed of two criss-cross structures. In each criss-cross structure, we perform corresponding feature extraction on the decomposed features according to different criss-cross shapes. After passing through two criss-cross structures in turn, each calculated feature can efficiently perceive the global underwater information. According to different parameter allocation strategies in feature extraction, our strategy can be divided into standard criss-cross global interaction strategy (SCGIS) and efficient criss-cross global interaction strategy (ECGIS). Compared with SCGIS, ECGIS can achieve better underwater information interaction with fewer parameters. On the basis of SCGIS and ECGIS, we then design the corresponding selective attention in the spatial, channel, and hybrid dimensions, respectively. Finally, we combine designed attention modules with YOLO detectors. The main reason for choosing the YOLO series here is that the YOLO algorithms belong to one-stage detection algorithms. They can better balance the detection speed and detection accuracy, which are more suitable for complex underwater environments. Our attention modules are added to both ends of the feature fusion in YOLO. In this paper, the main contributions of our work are summarized as follows:

- We first propose the criss-cross global interaction strategy, which simultaneously avoids the destruction of direct information correspondence and the lack of global information interaction. Our strategy fully perceives underwater global information and achieves efficient underwater information interaction.
- We then design the criss-cross global interaction-based selective attention for reducing the underwater background interference and improving the underwater object perception. On the basis of high-quality underwater interaction results, our selective attention extracts important underwater information from complex underwater environments.
- We finally combine the designed attention modules with YOLO detectors, which satisfies both high-precision and real-time requirements for underwater object detection.

The remainder of this paper is organized as follows. In Section 2, we review related work on attention mechanisms and object detection. In Section 3, we introduce the proposed method in detail. Experiments and analyses are provided in Section 4. The conclusion about our work is summarized in Section 5.

2 Related works

2.1 Attention mechanism

With the unremitting efforts of researchers, the attention mechanism in deep learning has made significant breakthroughs. According to different design ideas, attention mechanisms can be divided into selective attention mechanisms [14, 16, 19, 20, 26, 27, 39, 40, 49] and enhanced attention mechanisms [15, 24, 38, 44]. Selective attention can highlight important features and suppress unimportant features according to the calculated importance of each feature. The enhanced attention can enhance each feature according to the strength of the calculated correlation among all features. In order to better reduce the underwater background interference and improve underwater object perception, here we focus on selective attention.

Hu et al. [16] proposed the Squeeze-and-Excitation Network, where SEM learned the importance of each channel by modeling the interdependence among channels. The bottleneck structure was used in information interaction to reduce parameters and computations. This dimensionality reduction interaction strategy destroys the direct information correspondence, which will reduce attention effectiveness, especially in complex underwater environments. Wang et al. [39] proposed the Efficient Channel Attention Network, where ECAM used the local interaction strategy without dimensionality reduction in the channel dimension. This strategy reduced the interaction cost while ensuring direct information correspondence. A method for adaptively selecting the kernel size of 1D convolution was developed to better determine the coverage of local cross-channel interaction. They argue that the lack of global information interaction has little effect on attention performance. However, this is very disadvantageous for applying selective attention in complex underwater environments. Hou et al. [14] proposed the coordinate attention for an efficient mobile network, where CoAM generated the selective attention that captured both spatial and channel information by embedding spatial location into channel attention. To avoid the missing position information in the spatial dimension caused by 2D global pooling, they used two 1D global pooling to capture vertical spatial information and horizontal spatial information. Their work is extremely innovative and helps the network more accurately locate objects of interest.

BAM and CBAM are proposed in [26] and [40], respectively. BAM combined channel attention and spatial attention in parallel and successively used multiple dilated convolutions to expand the receptive field in the spatial dimension. CBAM combined channel attention and spatial attention in series and used both max pooling and average pooling to enrich receptive fields in different dimensions. For BAM and CBAM, there is still the problem of direct correspondence destruction on the channel branch and the problem of global interaction lack on the spatial branch. The selective attention that ensures direct information correspondence and achieves global information interaction is crucial for reducing underwater background interference and improving underwater object perception.

SGEM [20] used global average pooling and normalization to collect spatial information. Spatial-wise information interaction was further achieved by assigning a weight and a bias to each channel grouping. SRM [19] used global average pooling and global standard deviation pooling to collect two different spatial information. Channel-wise information interaction was further achieved by assigning two different learnable parameters to each channel dimension. For SGEM and SRM, the local spatial interaction and the local channel interaction degrade the selective attention performance in underwater detection environments. FCAM [27] used different DCT priors to capture the intrinsic information of channel grouping and used fully connected layers with a bottleneck structure to achieve global information interaction. The dimensionality reduction interaction strategy leads to the destruction of direct information correspondence, which is not conducive to underwater detection tasks. ShAM [49] mainly consisted of two attention branches. In information interaction, the channel attention branch and the spatial attention branch assign weight and bias parameters to each channel dimension, which realizes the local interaction of channel information and spatial information. Although the local interaction strategy avoids the destruction of direct information correspondence, it causes a lack of global information interaction.

2.2 Object detection

With the unremitting efforts of researchers, the object detection task in computer vision has made significant breakthroughs. According to different processing procedures, object detection algorithms can be divided into one-stage algorithms [2, 18, 23] and two-stage algorithms [11, 12, 31]. Compared with the two-stage detection algorithms, the one-stage detection algorithms have a greater advantage in inference speed. In order to better complete the underwater object detection task, here we focus on the detection algorithms in YOLO (You Only Look Once) series [2, 10, 17, 28–30, 36, 46, 47].

Redmon et al. proposed YOLOV1 [28], YOLOV2 [29] and YOLOV3 [30]. The core idea of the YOLO series is to directly input the entire image at the input end and directly output the position and corresponding category of the bounding box at the output end. YOLOV1 used GoogleLeNet as the backbone, which had ideal inference speed and generalization ability. YOLOV2 used DarkNet19 as the backbone and introduced the idea of anchor boxes. The model convergence speed was improved by adding the batch normalization layer after the convolution layer. The multi-scale training method improved the robustness of YOLOV2 on images with different sizes. The backbone used by YOLOV3 was DarkNet53. YOLOV3 applied the residual structure to extract features better and applied feature pyramid networks (FPN) for feature fusion. The multi-scale prediction strategy was used to better detect objects with different scales. Compared with YOLOV1 and YOLOV2, YOLOV3 can achieve a better balance of speed and accuracy.

Bochkovskiy et al. [2] proposed YOLOV4, which combined various tricks in deep learning. YOLOV4 introduced mosaic data augmentation and cross mini-batch normalization at the input. CSPDarkNet53, Mish activation function, and DropBlock regularization were used in the backbone. The spatial pyramid pooling (SPP) module and path aggregation network (PAN) structure were borrowed in the neck. In the head, the loss computation and non-maximum suppression were performed based on complete-intersection over union (CIOU) and distance-intersection over union (DIOU), respectively. Compared with the previous versions, YOLOV4 has a stronger performance. YOLOV5 was proposed in [17], which had a similar network structure to YOLOV4. In the backbone, YOLOV5 added Focus and SPP structures and tweaked the implementation details, which can be called modified CSPDarkNet. The cross-stage partial (CSP) structure is further used in the neck to strengthen the feature fusion ability of the network. Adaptive anchor box calculating and adaptive image scaling were applied at the input. YOLOV5 has stronger flexibility, which can achieve rapid deployment.

YOLOv6 [46] designed the EfficientRep backbone and the Rep-PAN neck based on the RepVGG style. The decoupled head is further optimized by reducing overhead. YOLOv6 adopted the anchor-free training strategy and the SimOTA label assignment strategy to further improve detection accuracy. For YOLOv7 [36], the extended efficient long-range attention network (Extended-ELAN) improved model learning ability without destroying the original gradient path. The concatenation-based model scaling method maintained the optimal structure of the model design. The planned re-parameterized convolution effectively increased model inference speed. The dynamic label assignment strategy with coarse-to-fine guidance provided better dynamic targets for different branches.

Ge et al. [10] proposed YOLOX based on YOLOV3. YOLOX used an anchor-free strategy to reduce the complexity of the detection head and used the decoupled head to improve the model convergence speed. The SimOTA strategy was applied to the loss computation, which is able to dynamically match positive samples for objects with different sizes. In general, YOLOX has superior performance in terms of speed and accuracy. YOLOV8 [47] was an improved version based on YOLOV5, which has the best speed and detection performance at present. The backbone still used the CSP idea and achieved further lightweight. The neck still used the PAN idea and tweaked the convolutional structure. The head used both decoupled-head and anchor-free strategies. Task-aligned assigner was used in sample matching.

3 Proposed method

In this section, we first introduce the main idea of a criss-cross global interaction strategy (CGIS). We then provide detailed processes for standard criss-cross global interaction strategy (SCGIS) and efficient criss-cross global interaction strategy (ECGIS). Next, we design criss-cross global interaction-based selective attention in the spatial, channel, and hybrid dimensions, respectively. Finally, our attention modules are combined with YOLO algorithms.

3.1 Criss-cross global interaction strategy (CGIS)

CGIS is mainly composed of two criss-cross structures. In each criss-cross structure, feature decomposition and feature extraction are performed in sequence. In feature decomposition, we select features located at the corresponding positions according to different criss-cross shapes. In feature extraction, we perform convolution calculations on these selected features. It is worth noting here that, in CGIS, the input features $\mathbf{X} \in \mathbb{R}^{H \times W}$ cannot yet achieve information interaction, the features $\mathbf{Y} \in \mathbb{R}^{H \times W}$ obtained after the first criss-cross structure can achieve local information interaction, and the output features $\mathbf{Z} \in \mathbb{R}^{H \times W}$ obtained after the second criss-cross structure can achieve global information interaction. The general form of CGIS is formulated as follows:

$$\mathbf{Y} = E_1\left(D_1\left(\mathbf{X}\right)\right), \tag{1}$$

$$\mathbf{Z} = E_2\left(D_2\left(\mathbf{Y}\right)\right), \tag{2}$$

where D_1 and E_1 respectively represent feature decomposition and feature extraction in the first criss-cross structure, and D_2 and E_2 respectively represent feature decomposition and feature extraction in the second criss-cross structure. Figure 1 shows the main idea of CGIS, where the height H and the width W are set to 3 respectively, and the features at different positions are represented by circles of different colors. In order to better emphasize the difference in the degree of information interaction at different stages, we draw the input features $\mathbf{X}$ at the initial stage, the features $\mathbf{Y}$ at the intermediate stage, and the output features $\mathbf{Z}$ at the final stage with lines of different thickness. Obviously, if the line used to draw the circle in Fig.1 is thicker, it indicates that the degree of information interaction at this stage is stronger. Next, we will focus on two specific implementations of CGIS, including SCGIS and ECGIS.

3.1.1 Standard criss-cross global interaction strategy (SCGIS)

For SCGIS, the first criss-cross structure is mainly composed of two processes: the first feature decomposition and the first feature extraction. The first feature decomposition can be completed in two steps. First, we generate HW masks with different criss-cross shapes, where each mask has $H \times W$ size. Second, according to these generated masks $\mathbf{M} \in \mathbb{R}^{HW \times (H \times W)}$,

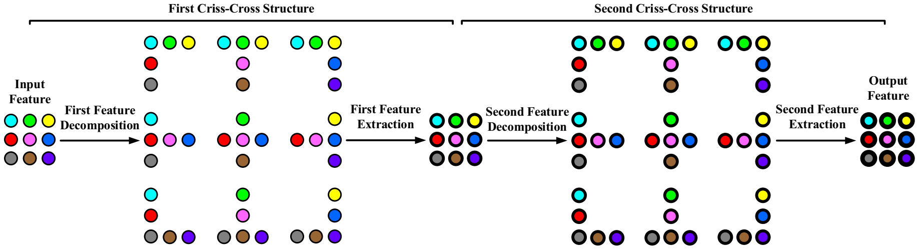
First Criss-Cross Structure
Second Criss-Cross Structure
Input Feature
First Feature Decomposition
First Feature Extraction
Second Feature Decomposition
Second Feature Extraction
Output Feature

Fig. 1 The main idea of criss-cross global interaction strategy (CGIS). For the convenience of drawing, the width and height are set to 3 here. Circles with different colors represent features located at different locations. In different stages, the stronger the information interaction intensity, the thicker the drawn circle lines

we perform the mask operation on the input features $\mathbf{X} \in \mathbb{R}^{H \times W}$ to obtain the corresponding features $\mathbf{P} \in \mathbb{R}^{HW \times (H+W-1)}$, where the value at the true position is retained, and the value at the fasle position is discarded. The first feature decomposition is formulated as follows:

$$\mathbf{P} = D_1(\mathbf{X}) = f_{\{\mathbf{M}\}}(\mathbf{X}), \tag{3}$$

where $f_{\{\mathbf{M}\}}$ represents the mask operation. The mask operation in the first feature decomposition is defined as:

$$f_{\{\mathbf{M}\}}(\mathbf{X}) = f_{\{\mathbf{m}_n\}}(\mathbf{X}) = \begin{cases} x_{ij} & if\ m_{nij} = True \\ \text{none} & if\ m_{nij} = False \end{cases}, \tag{4}$$

where $n = [1, ..., HW]$, $i = [1, ..., H]$, and $j = [1, ..., W]$. $x_{ij} \in \mathbb{R}$ represents the feature at the ith row and j th column in $\mathbf{X}$, $\mathbf{m}_n \in \mathbb{R}^{H \times W}$ represents the n th mask in $\mathbf{M}$, and $m_{nij} \in \mathbb{R}$ represents the Boolean value at the i th row and j th column in $\mathbf{m}_n$.

After the first feature decomposition, we next perform the first feature extraction. The first feature extraction can be completed in three steps. First, we generate a 1-dimensional (1D) group convolution layer with parameters $\mathbf{\Omega} \in \mathbb{R}^{HW \times (H+W-1)}$, where the number of input channels and output channels are all set to HW, and the kernel size is set to $H + W - 1$. Second, we put the features $\mathbf{P}$ into the configured layer for 1D group convolution, where features $\mathbf{P}$ with $HW \times (H + W - 1)$ size can be processed into features with $HW \times 1$ size. Third, these features with $HW \times 1$ size are reshaped into features $\mathbf{Y}$ with $H \times W$ size. At this time, the features $\mathbf{Y}$ obtained after the first criss-cross structure have achieved local information interaction according to different criss-cross shapes in the input features $\mathbf{X}$. The first feature extraction is formulated as follows:

$$\mathbf{Y} = E_1(\mathbf{P}) = f_{\{H \times W\}}\left(f_{\{\mathbf{\Omega}\}}(\mathbf{P})\right), \tag{5}$$

where $f_{\{H \times W\}}$ represents the reshape operation that can reshape the size from $HW \times 1$ to $H \times W$, and $f_{\{\mathbf{\Omega}\}}$ represents the convolution operation. The convolution operation in the first feature extraction is defined as:

$$f_{\{\mathbf{\Omega}\}}(\mathbf{P}) = \mathbf{p}_n (\mathbf{\Omega}_n)^T, \tag{6}$$

where $n = [1, ..., HW]$. T represents the transpose operation, $\mathbf{p}_n \in \mathbb{R}^{1 \times (H+W-1)}$ represents features at the nth row in $\mathbf{P}$, and $\mathbf{\Omega}_n \in \mathbb{R}^{1 \times (H+W-1)}$ represents parameters at the nth row in $\mathbf{\Omega}$.

For SCGIS, the second criss-cross structure is similar to the first criss-cross structure, mainly composed of two processes: the second feature decomposition and the second feature extraction. The second feature decomposition can be completed on the basis of the first feature decomposition. We directly use the masks $\mathbf{M} \in \mathbb{R}^{HW \times (H \times W)}$ generated in the first feature decomposition and use the same mask operation $f_{\{\mathbf{M}\}}$ to decompose features $\mathbf{Y} \in \mathbb{R}^{H \times W}$. The second feature decomposition is formulated as follows:

$$\mathbf{U} = D_2(\mathbf{Y}) = f_{\{\mathbf{M}\}}(\mathbf{Y}), \tag{7}$$

where $\mathbf{U} \in \mathbb{R}^{HW \times (H+W-1)}$ represents the feature obtained after the second feature decomposition. The mask operation in the second feature decomposition is defined as:

$$f_{\{\mathbf{M}\}}(\mathbf{Y}) = f_{\{\mathbf{m}_n\}}(\mathbf{Y}) = \begin{cases} y_{ij} & if\ m_{nij} = True \\ \text{none} & if\ m_{nij} = False \end{cases}, \tag{8}$$

where $n = [1, ..., HW]$, $i = [1, ..., H]$, and $j = [1, ..., W]$. $y_{ij} \in \mathbb{R}$ represents the feature at the i th row and jth column in $\mathbf{Y}$. The meanings of $\mathbf{m}_n$ and m_{nij} are explained in (4).

The second feature extraction can be performed after the second decomposition. First, we regenerate a 1D group convolution layer with parameter $\mathbf{\Phi} \in \mathbb{R}^{HW \times (H+W-1)}$, where the setting strategy is the same as the first feature extraction. Second, we perform 1D group convolution on the features $\mathbf{U}$. Third, by using the same reshape operation $f_{\{H \times W\}}$ in the first feature extraction, the size of the features obtained after convolution is reshaped from $HW \times 1$ to $H \times W$. The second feature extraction can be completed after the above three steps. The second feature extraction is formulated as follows:

$$\mathbf{Z} = E_2(\mathbf{U}) = f_{\{H \times W\}}\left(f_{\{\mathbf{\Phi}\}}(\mathbf{U})\right), \tag{9}$$

where $f_{\{\mathbf{\Phi}\}}$ represents the convolution operation. The convolution operation in the second feature extraction is defined as:

$$f_{\{\mathbf{\Phi}\}}(\mathbf{U}) = \mathbf{u}_n(\mathbf{\Phi}_n)^T, \tag{10}$$

where $n = [1, ..., HW]$. $\mathbf{u}_n \in \mathbb{R}^{1 \times (H+W-1)}$ represents features at the nth row in $\mathbf{U}$, and $\mathbf{\Phi}_n \in \mathbb{R}^{1 \times (H+W-1)}$ represents parameters at the nth row in $\mathbf{\Phi}$. At this point, the output features $\mathbf{Z}$ obtained after the two criss-cross structures have achieved global information interaction.

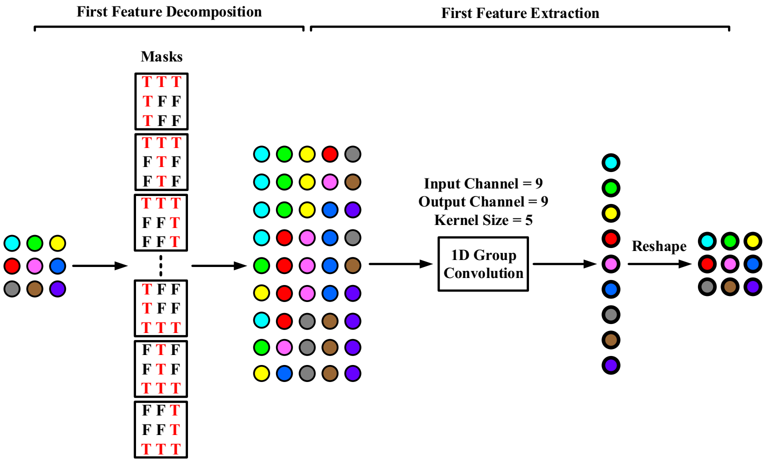

Fig. 2 The first criss-cross structure in standard criss-cross global interaction strategy (SCGIS)

In order to better show the design process of SCGIS, here we focus on the first criss-cross structure in SCGIS as an example, where the second criss-cross structure in SCGIS is similar. According to Fig. 1, the first feature decomposition and the first feature extraction of SCGIS are shown in Fig. 2.

3.1.2 Efficient criss-cross global interaction strategy (ECGIS)

For ECGIS, the feature decomposition in the first criss-cross structure can be completed in two steps. First, we generate three different types of masks, including masks $\mathbf{M} \in \mathbb{R}^{HW\times(H\times W)}$, mask $\mathbf{M}^{'} \in \mathbb{R}^{HW\times(H+W-1)}$ and mask $\mathbf{M}^{''} \in \mathbb{R}^{HW\times(H+W-1)}$. The purpose of generating masks $\mathbf{M}$ is to extract corresponding features according to different criss-cross shapes. The purpose of generating mask $\mathbf{M}^{'}$ is to further extract the required features according to the row direction of different criss-cross shapes. The purpose of generating mask $\mathbf{M}^{''}$ is to further extract the required features according to the column direction of different criss-cross shapes. Second, the features $\mathbf{P} \in \mathbb{R}^{HW\times(H+W-1)}$ are obtained by combining the input features $\mathbf{X}$ with the masks $\mathbf{M}$. We further combine features $\mathbf{P}$ with masks $\mathbf{M}^{'}$ and $\mathbf{M}^{''}$ to obtain features $\mathbf{P}^{'} \in \mathbb{R}^{HW\times W}$ and $\mathbf{P}^{''} \in \mathbb{R}^{HW\times(H-1)}$, respectively. The first feature decomposition is formulated as follows:

$$\mathbf{P}^{'}, \mathbf{P}^{''} = D_1(\mathbf{X}) = \begin{cases} f_{\left\{\mathbf{M}^{'}\right\}}\left(f_{\{\mathbf{M}\}}(\mathbf{X})\right) \\ f_{\left\{\mathbf{M}^{''}\right\}}\left(f_{\{\mathbf{M}\}}(\mathbf{X})\right) \end{cases}, \tag{11}$$

where $\mathbf{P} = f_{\{\mathbf{M}\}}(\mathbf{X})$. $f_{\{\mathbf{M}\}}$, $f_{\left\{\mathbf{M}^{'}\right\}}$, and $f_{\left\{\mathbf{M}^{''}\right\}}$ represent different mask operations. $f_{\{\mathbf{M}\}}(\mathbf{X})$ is defined in (4). $f_{\left\{\mathbf{M}^{'}\right\}}(\mathbf{P})$ and $f_{\left\{\mathbf{M}^{''}\right\}}(\mathbf{P})$ are defined as:

$$f_{\left\{\mathbf{M}^{'}\right\}}(\mathbf{P}) = \begin{cases} p_{nk} \;\; if \;\; m_{nk}^{'} = True \\ \text{none} \; if \;\; m_{nk}^{'} = False \end{cases}, \tag{12}$$

$$f_{\left\{\mathbf{M}^{''}\right\}}(\mathbf{P}) = \begin{cases} p_{nk} \;\; if \;\; m_{nk}^{''} = True \\ \text{none} \; if \;\; m_{nk}^{''} = False \end{cases}, \tag{13}$$

where $n = [1, ..., HW]$ and $k = [1, ..., H + W - 1]$. $p_{nk} \in \mathbb{R}$ represents the feature at the n th row and kth column in $\mathbf{P}$. $m_{nk}^{'} \in \mathbb{R}$ and $m_{nk}^{''} \in \mathbb{R}$ represent the Boolean values at the nth row and kth column in $\mathbf{M}^{'}$ and $\mathbf{M}^{''}$, respectively.

The feature extraction in the first criss-cross structure can be completed in three steps. First, by using the reshape operation $f_{\{W\times HW\}}$, the features $\mathbf{P}^{'} \in \mathbb{R}^{HW\times W}$ are reshaped into the features $\mathbf{P}^{'\boldsymbol{\Delta}} \in \mathbb{R}^{W\times HW}$. By using the reshape operation $f_{\{H\times W(H-1)\}}$, the features $\mathbf{P}^{''} \in \mathbb{R}^{HW\times(H-1)}$ are reshaped into the features $\mathbf{P}^{''\boldsymbol{\Delta}} \in \mathbb{R}^{H\times W(H-1)}$. Second, a 1D group convolution layer with parameters $\alpha \in \mathbb{R}^{W\times W}$ is generated to process $\mathbf{P}^{'\boldsymbol{\Delta}}$, where the number of input and output channels, the kernel size and the stride are all set to W. The 1D group convolution layer with parameters $\beta \in \mathbb{R}^{H\times(H-1)}$ are generated to process $\mathbf{P}^{''\boldsymbol{\Delta}}$, where the number of input and output channels are all set to H, and the kernel size and the stride are all set to $H - 1$. Third, after the 1D group convolution for $\mathbf{P}^{'\boldsymbol{\Delta}}$ and $\mathbf{P}^{''\boldsymbol{\Delta}}$, the features with $W \times H$ size and the features with $H \times W$ size are obtained respectively. We transpose the features with $W \times H$ size and sum them with the features with $H \times W$ size to get features

$\mathbf{Y} \in \mathbb{R}^{H\times W}$. The first feature extraction is formulated as follows:

$$\begin{aligned}\mathbf{Y} &= E_1\left(\mathbf{P}^{'}, \mathbf{P}^{''}\right)\\ &= f_{\{\alpha\}}\left(f_{\{W\times HW\}}\left(\mathbf{P}^{'}\right)\right)^T + f_{\{\beta\}}\left(f_{\{H\times W(H-1)\}}\left(\mathbf{P}^{''}\right)\right)\end{aligned}, \tag{14}$$

where $\mathbf{P}^{'\boldsymbol{\Delta}} = f_{\{W\times HW\}}\left(\mathbf{P}^{'}\right)$ and $\mathbf{P}^{''\boldsymbol{\Delta}} = f_{\{H\times W(H-1)\}}\left(\mathbf{P}^{''}\right)$. $f_{\{\alpha\}}$ and $f_{\{\beta\}}$ represent the convolution operations. $f_{\{\alpha\}}\left(\mathbf{P}^{'\boldsymbol{\Delta}}\right)$ and $f_{\{\beta\}}\left(\mathbf{P}^{''\boldsymbol{\Delta}}\right)$ are defined as:

$$f_{\{\alpha\}}\left(\mathbf{P}^{'\boldsymbol{\Delta}}\right) = f_{\{\alpha_j\}}\left(\mathbf{p}_j^{'\boldsymbol{\Delta}}\right) = \mathbf{p}_{ja}^{'\boldsymbol{\Delta}}\left(\alpha_j\right)^T, \tag{15}$$

$$f_{\{\beta\}}\left(\mathbf{P}^{''\boldsymbol{\Delta}}\right) = f_{\{\beta_i\}}\left(\mathbf{p}_i^{''\boldsymbol{\Delta}}\right) = \mathbf{p}_{ib}^{''\boldsymbol{\Delta}}\left(\beta_i\right)^T, \tag{16}$$

where $i = [1, ..., H]$, $j = [1, ..., W]$, $a = [1, ..., H]$, and $b = [1, ..., W]$. $\mathbf{p}_j^{'\boldsymbol{\Delta}} \in \mathbb{R}^{1\times HW}$ and $\mathbf{p}_i^{''\boldsymbol{\Delta}} \in \mathbb{R}^{1\times W(H-1)}$ represent the features at the jth row and the ith row in $\mathbf{P}^{'\boldsymbol{\Delta}}$ and $\mathbf{P}^{''\boldsymbol{\Delta}}$, respectively. $\mathbf{p}_{ja}^{'\boldsymbol{\Delta}} \in \mathbb{R}^{1\times W}$ and $\mathbf{p}_{ib}^{''\boldsymbol{\Delta}} \in \mathbb{R}^{1\times (H-1)}$ represent the features belonging to the ath stride range and the bth stride range in $\mathbf{p}_j^{'\boldsymbol{\Delta}}$ and $\mathbf{p}_i^{''\boldsymbol{\Delta}}$, respectively. $\alpha_j \in \mathbb{R}^{1\times W}$ and $\beta_i \in \mathbb{R}^{1\times(H-1)}$ represent the features at the jth row and the ith row in α and β, respectively.

For ECGIS, the feature decomposition in the second criss-cross structure can be completed in two steps. First, $\mathbf{Y} \in \mathbb{R}^{H\times W}$ are decomposed into $\mathbf{U} \in \mathbb{R}^{HW\times(H+W-1)}$ by using the mask operation $f_{\{\mathbf{M}\}}$ defined in the first feature decomposition. Second, we further use the mask operations $f_{\left\{\mathbf{M}^{'}\right\}}$ and $f_{\left\{\mathbf{M}^{''}\right\}}$ defined in the first feature decomposition to decompose $\mathbf{U} \in \mathbb{R}^{HW\times(H+W-1)}$ into $\mathbf{U}^{'} \in \mathbb{R}^{HW\times W}$ and $\mathbf{U}^{''} \in \mathbb{R}^{HW\times(H-1)}$. The second feature decomposition is formulated as follows:

$$\mathbf{U}^{'}, \mathbf{U}^{''} = D_2\left(\mathbf{Y}\right) = \begin{cases} f_{\left\{\mathbf{M}^{'}\right\}}\left(f_{\{\mathbf{M}\}}\left(\mathbf{Y}\right)\right) \\ f_{\left\{\mathbf{M}^{''}\right\}}\left(f_{\{\mathbf{M}\}}\left(\mathbf{Y}\right)\right) \end{cases}, \tag{17}$$

where $\mathbf{U} = f_{\{\mathbf{M}\}}\left(\mathbf{Y}\right)$. $f_{\{\mathbf{M}\}}\left(\mathbf{Y}\right)$ is defined in (8). $f_{\left\{\mathbf{M}^{'}\right\}}\left(\mathbf{U}\right)$ and $f_{\left\{\mathbf{M}^{''}\right\}}\left(\mathbf{U}\right)$ are defined as:

$$f_{\left\{\mathbf{M}^{'}\right\}}\left(\mathbf{U}\right) = \begin{cases} u_{nk} \;\; if \;\; m_{nk}^{'} = True \\ \text{none} \; if \;\; m_{nk}^{'} = False \end{cases}, \tag{18}$$

$$f_{\left\{\mathbf{M}^{''}\right\}}\left(\mathbf{U}\right) = \begin{cases} u_{nk} \;\; if \;\; m_{nk}^{''} = True \\ \text{none} \; if \;\; m_{nk}^{''} = False \end{cases}, \tag{19}$$

where $n = [1, ..., HW]$ and $k = [1, ..., H + W - 1]$. $u_{nk} \in \mathbb{R}$ represents the feature at the nth row and kth column in $\mathbf{U}$. The meanings of $m_{nk}^{'}$ and $m_{nk}^{''}$ are explained in (12) and (13).

The feature extraction in the second criss-cross structure can be completed in three steps. First, $\mathbf{U}^{'} \in \mathbb{R}^{HW\times W}$ and $\mathbf{U}^{''} \in \mathbb{R}^{HW\times(H-1)}$ are reshaped into $\mathbf{U}^{'\boldsymbol{\Delta}} \in \mathbb{R}^{W\times HW}$ and $\mathbf{U}^{''\boldsymbol{\Delta}} \in \mathbb{R}^{H\times W(H-1)}$ respectively by using the same reshape operations $f_{\{W\times HW\}}$ and $f_{\{H\times W(H-1)\}}$ in the first feature extraction. Second, two 1D group convolution layers with parameters $\delta \in \mathbb{R}^{W\times W}$ and $\varphi \in \mathbb{R}^{H\times(H-1)}$ are generated to process $\mathbf{U}^{'\boldsymbol{\Delta}}$ and $\mathbf{U}^{''\boldsymbol{\Delta}}$, where the setting strategies are the same as the first feature extraction. Third, the processing for $\mathbf{U}^{'\boldsymbol{\Delta}}$ and $\mathbf{U}^{''\boldsymbol{\Delta}}$

in the second feature extraction is the same as the processing for $\mathbf{P}'^{\boldsymbol{\Delta}}$ and $\mathbf{P}''^{\boldsymbol{\Delta}}$ in the first feature extraction. The second feature extraction is formulated as follows:

$$\begin{aligned}\mathbf{Z} &= E_2\left(\mathbf{U}', \mathbf{U}''\right) \\ &= f_{\{\delta\}}\left(f_{\{W\times HW\}}\left(\mathbf{U}'\right)\right)^T + f_{\{\varphi\}}\left(f_{\{H\times W(H-1)\}}\left(\mathbf{U}''\right)\right),\end{aligned} \tag{20}$$

where $\mathbf{U}'^{\boldsymbol{\Delta}} = f_{\{W\times HW\}}\left(\mathbf{U}'\right)$ and $\mathbf{U}''^{\boldsymbol{\Delta}} = f_{\{H\times W(H-1)\}}\left(\mathbf{U}''\right)$. $f_{\{\delta\}}$ and $f_{\{\varphi\}}$ represent the convolution operations. $f_{\{\delta\}}\left(\mathbf{U}'^{\boldsymbol{\Delta}}\right)$ and $f_{\{\varphi\}}\left(\mathbf{U}''^{\boldsymbol{\Delta}}\right)$ are defined as:

$$f_{\{\delta\}}\left(\mathbf{U}'^{\Delta}\right) = f_{\{\delta_j\}}\left(\mathbf{u}_j'^{\Delta}\right) = \mathbf{u}_{ja}'^{\Delta}\left(\delta_j\right)^T, \tag{21}$$

$$f_{\{\varphi\}}\left(\mathbf{U}''^{\Delta}\right) = f_{\{\varphi_i\}}\left(\mathbf{u}_i''^{\Delta}\right) = \mathbf{u}_{ib}''^{\Delta}\left(\varphi_i\right)^T, \tag{22}$$

where $i = [1, ..., H]$, $j = [1, ..., W]$, $a = [1, ..., H]$, and $b = [1, ..., W]$. $\mathbf{u}_j'^{\boldsymbol{\Delta}} \in \mathbb{R}^{1\times HW}$ and $\mathbf{u}_i''^{\boldsymbol{\Delta}} \in \mathbb{R}^{1\times W(H-1)}$ represent the features at the jth row and the ith row in $\mathbf{U}'^{\boldsymbol{\Delta}}$ and $\mathbf{U}''^{\boldsymbol{\Delta}}$, respectively. $\mathbf{u}_{ja}'^{\boldsymbol{\Delta}} \in \mathbb{R}^{1\times W}$ and $\mathbf{u}_{ib}''^{\boldsymbol{\Delta}} \in \mathbb{R}^{1\times(H-1)}$ represent the features belonging to the ath stride range and the bth stride range in $\mathbf{u}_j'^{\boldsymbol{\Delta}}$ and $\mathbf{u}_i''^{\boldsymbol{\Delta}}$, respectively. $\delta_j \in \mathbb{R}^{1\times W}$ and $\varphi_i \in \mathbb{R}^{1\times(H-1)}$ represent the features at the jth row and the ith row in δ and φ, respectively.

In order to better show the design process of ECGIS, here we focus on the first criss-cross structure in ECGIS as an example, where the second criss-cross structure in ECGIS is similar. According to Fig. 1, the first feature decomposition and the first feature extraction of ECGIS are shown in Fig. 3. For SCGIS, we assign different parameters to these features selected from different criss-cross shapes, and these parameters are not shared with each other during training. For ECGIS, we use a parameter sharing method in feature extraction, which can further reduce parameters and achieve more effective global information interaction. Our parameter sharing method is shown in Fig. 4, where the position of the star symbol indicates the intersection of row and column directions in the criss-cross shape. We focus on the locations of these intersections, and divide features with the same position state according to row and column directions, respectively. The features enclosed by the same color box share parameters during training.

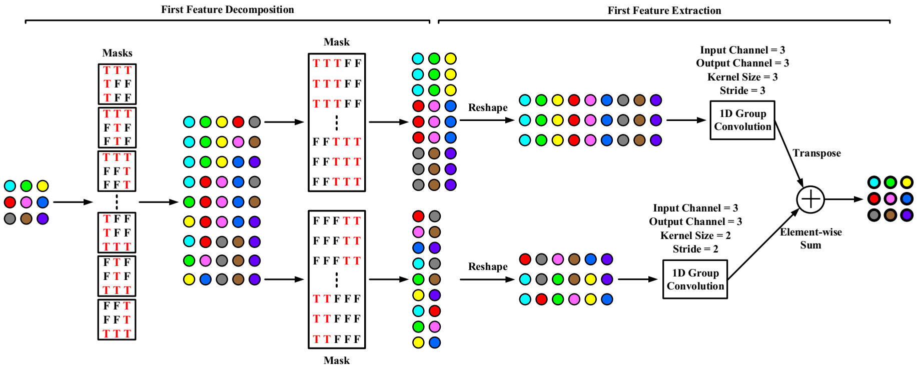

Fig. 3 The first criss-cross structure in efficient criss-cross global interaction strategy (ECGIS)

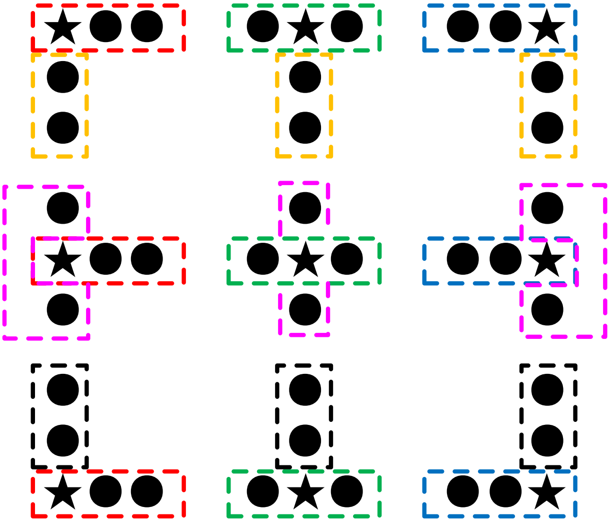

Fig. 4 The parameter allocation strategy in efficient criss-cross global interaction strategy (ECGIS). Features with the same location state share the same trainable parameters, which are boxed by the same color

3.2 Criss-cross global interaction-based selective attention

In this subsection, we plan to design selective attention using the criss-cross global interaction strategy in information interaction. There are three things to note here. First, according to different implementations, CGIS can be divided into SCGIS and ECGIS. Second, according to different target dimensions, selective attention can be divided into spatial attention, channel attention and hybrid attention. Third, the structural design of selective attention using different interaction strategies on the same target dimension is exactly the same. For the convenience of introduction, the subsequent content is organized as follows.

We first introduce the criss-cross global interaction-based spatial attention module (CGI-SAM), which includes standard criss-cross global interaction-based spatial attention module (SCGI-SAM) and efficient criss-cross global interaction-based spatial attention module (ECGI-SAM). We then introduce the criss-cross global interaction-based channel attention module (CGI-CAM), which includes standard criss-cross global interaction-based channel attention module (SCGI-CAM) and efficient criss-cross global interaction-based channel attention module (ECGI-CAM). We finally introduce the criss-cross global interaction-based hybrid attention module (CGI-HAM), which includes standard criss-cross global interaction-based hybrid attention module (SCGI-HAM) and efficient criss-cross global interaction-based hybrid attention module (ECGI-HAM).

3.2.1 Criss-cross global interaction-based spatial attention module (CGI-SAM)

The structural design of CGI-SAM is shown in Fig. 5. When SCGIS is specifically used in information interaction, CGI-SAM is denoted as SCGI-SAM. When ECGIS is specifically used in information interaction, CGI-SAM is denoted as ECGI-SAM. As can be seen from Fig. 5, CGI-SAM is mainly composed of three processes, including information preprocessing, information interaction and attention activation. First, information preprocessing is responsible for processing input features into the features required for subsequent operations.

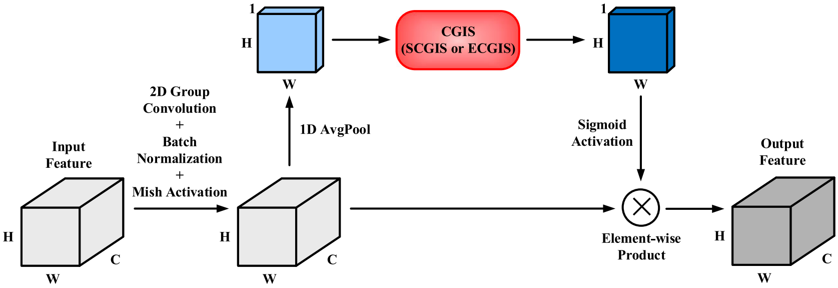

Fig. 5 The structural design of criss-cross global interaction-based spatial attention module (CGI-SAM). According to different interaction strategies, CGI-SAM can be denoted as standard criss-cross global interaction-based spatial attention module (SCGI-SAM) or efficient criss-cross global interaction-based spatial attention module (ECGI-SAM)

We use the 2-dimensional (2D) group convolution with parameters $\mathbf{W} \in \mathbb{R}^{C\times 1\times 1}$ to quickly process the input features $\mathbf{A} \in \mathbb{R}^{C\times H\times W}$, where the number of input and output channels are set to C, and the kernel size is set to 1×1. This convolution operation has low computational cost and can appropriately reduce the similarity of deep features, which can make the subsequent selective attention can play a better screening role. We then use batch normalization and Mish activation on the features obtained after convolution to get features $\mathbf{B} \in \mathbb{R}^{C\times H\times W}$. $\mathbf{B}$ is further processed into features $\mathbf{C} \in \mathbb{R}^{1\times H\times W}$ after 1D global average pooling. Second, information interaction is responsible for capturing feature dependencies in the target dimension. Passing $\mathbf{C}$ into CGIS can get features $\mathbf{D} \in \mathbb{R}^{1\times H\times W}$ that effectively perceive global information in the spatial dimension. CGIS is detailed in Subsection 3.1, which includes the introduction to SCGIS and ECGIS. Third, attention activation is responsible for activating the features obtained after information interaction to obtain the attention map and combining the attention map with the target features to finally play the role of the attention mechanism. We use Sigmoid activation for $\mathbf{D}$ to obtain the spatial attention map with $1 \times H \times W$ size, and use the element-wise product to combine spatial attention with $\mathbf{B}$ to get output features $\mathbf{E} \in \mathbb{R}^{C\times H\times W}$.

3.2.2 Criss-cross global interaction-based channel attention module (CGI-CAM)

We show the structural design of CGI-CAM in Fig. 6. When SCGIS and ECGIS are used in information interaction, respectively, CGI-CAM can be denoted as SCGI-CAM and ECGI-

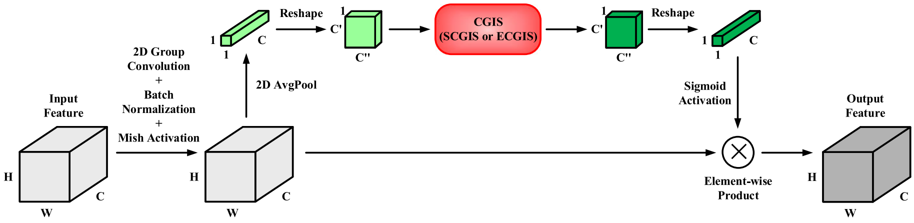

Fig. 6 The structural design of criss-cross global interaction-based channel attention module (CGI-CAM). According to different interaction strategies, CGI-CAM can be denoted as standard criss-cross global interaction-based channel attention module (SCGI-CAM) or efficient criss-cross global interaction-based channel attention module (ECGI-CAM)

CAM, respectively. CGI-CAM also consists of three processes. In information preprocessing, 2D group convolution with parameters $\mathbf{W} \in \mathbb{R}^{C\times 1\times 1}$, batch normalization and Mish function are used to process input features $\mathbf{A} \in \mathbb{R}^{C\times H\times W}$ to get the features $\mathbf{B} \in \mathbb{R}^{C\times H\times W}$. We then perform 2D global average pooling on $\mathbf{B}$ to obtain features with $C \times 1 \times 1$ size, and further reshape the obtained features into features $\mathbf{C} \in \mathbb{R}^{1\times C'\times C''}$, where $C = C' \times C''$. In information interaction, SCGIS or ECGIS can also be chosen to process $\mathbf{C}$. We then reshape the features obtained after CGIS into the features $\mathbf{D} \in \mathbb{R}^{C\times 1\times 1}$. In attention activation, the channel attention map with $C \times 1 \times 1$ size is obtained by performing the Sigmoid function on $\mathbf{D}$. Using the element-wise product to combine channel attention with $\mathbf{B}$ can get output features $\mathbf{E} \in \mathbb{R}^{C\times H\times W}$.

3.2.3 Criss-cross global interaction-based hybrid attention module (CGI-HAM)

On the basis of CGI-SAM and CGI-CAM, we further design CGI-HAM according to different combinations, as shown in Fig. 7(a)-(c). Figure 7(a) shows the combination of CGI-SAM and CGI-CAM in series, which is abbreviated as CGI-HAM1. Figure 7(b) shows the combination of CGI-CAM and CGI-SAM in series, which is abbreviated as CGI-HAM2. Figure 7(c) shows the combination of CGI-CAM and CGI-SAM in parallel, which is abbreviated as CGI-HAM3. To be more specific, if we use SCGIS or ECGIS in information interaction, CGI-HAM1 can be further divided into SCGI-HAM1 or ECGI-HAM1. The same is true for CGI-HAM2 and CGI-HAM3.

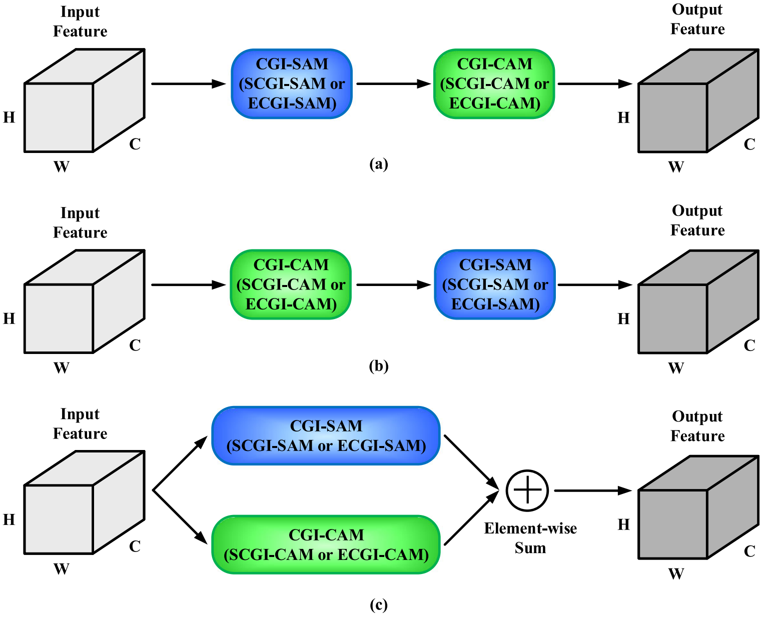
Input Feature
H
W
C
CGI-SAM (SCGI-SAM or ECGI-SAM)
CGI-CAM (SCGI-CAM or ECGI-CAM)
Output Feature
H
W
C
(a)
Input Feature
H
W
C
CGI-CAM (SCGI-CAM or ECGI-CAM)
CGI-SAM (SCGI-SAM or ECGI-SAM)
Output Feature
H
W
C
(b)
Input Feature
H
W
C
CGI-SAM (SCGI-SAM or ECGI-SAM)
CGI-CAM (SCGI-CAM or ECGI-CAM)
Element-wise Sum
Output Feature
H
W
C
(c)

Fig. 7 The criss-cross global interaction-based hybrid attention module (CGI-HAM). (a) The combination of CGI-SAM and CGI-CAM in series, where CGI-HAM1 includes SCGI-HAM1 and ECGI-HAM1. (b) The combination of CGI-CAM and CGI-SAM in series, where CGI-HAM2 includes SCGI-HAM2 and ECGI-HAM2. (c) The combination of CGI-SAM and CGI-CAM in parallel, where CGI-HAM3 includes SCGI-HAM3 and ECGI-HAM3

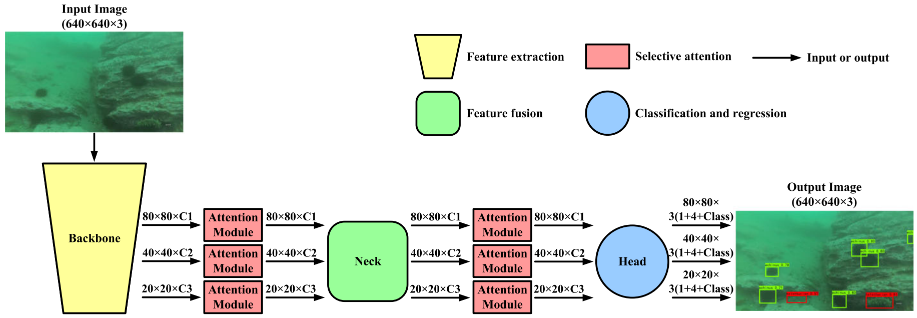

Fig. 8 Combining attention modules with YOLO detectors for underwater object detection

3.3 Application of attention modules

In order to meet the accuracy and real-time requirements of underwater object detection, we combine designed modules with YOLO algorithms, as shown in Fig. 8. Here, our attention modules refer to CGI-SAM, CGI-CAM, and CGI-HAM. The YOLO detectors refer to YOLOV3 [30], YOLOV4 [2], YOLOV5 [17], YOLOV6 [46], YOLOV7 [36], YOLOV8 [47] and YOLOX [10]. It is worth noting that the YOLO series have a similar network architecture, which is mainly composed of a backbone, neck, and head. The backbone is responsible for extracting image features, which can obtain high-level semantic information. The neck is responsible for fusing features at different scales, which can further enhance semantic information. The head is responsible for classifying and regressing the enhanced features at different scales, which can obtain the object category and bounding box position. We add the attention modules to both ends of the YOLO neck. Our selective attention can devote limited computing resources to more important underwater regions, which is crucial for reducing underwater background interference and improving underwater object perception.

In order to make the application environment of YOLO algorithms more flexible, YOLO detectors often have multiple different network sizes. Here we focus on YOLOX as an example. YOLOX can be divided into YOLOX-Nano, YOLOX-Tiny, YOLOX-S, YOLOX-M, YOLOX-L, and YOLOX-X according to different network scales. Based on the above sequence, C_1 can be taken as 64, 96, 128, 192, 256, and 320, respectively. C_2 can be taken as 128, 192, 256, 384, 512, and 640, respectively. C_3 can be taken as 256, 384, 512, 768, 1024, and 1280, respectively.

4 Experiments and analyses

In order to validate our work, we conduct comprehensive experiments on the underwater image dataset [3, 35] and the PASCAL VOC dataset [8, 9] and analyze the experimental results in detail. In this section, we first provide the training details of the network model used in the detection task. Then, ablation experiments and comparative experiments are performed on the underwater image datasets. Finally, the experiments are performed on the PASCAL VOC dataset. We use single-class average precision (AP) and multi-class mean average precision (mAP) to measure the detection accuracy and use frames per second (FPS) to measure the detection speed. The parameters, giga floating-point operations per second (GFLOPs), and memory consumption were used to measure the network scale and operation cost. On the basis of the above objective evaluation indicators, in order to better visualize the results, we further use detection maps, attention maps, epoch-mAP graphs, and epoch-loss graphs to comprehensively demonstrate the subjective performance of the proposed attention.

4.1 Training details

In this paper, our work is mainly based on URPC and Brackish datasets. The underwater image dataset (URPC 2017-2019) consists of URPC 2017(17655), URPC 2018(2901) and URPC 2019(4757), which has a total of 22577 images and 4 categories (Holothurian, Echinus, Scallop, and Starfish) after removing duplicate images. The underwater image dataset (Brackish 2019) has a total of 10,995 images and 6 categories: fish, small fish, crab, shrimp, jellyfish, and starfish. The PASCAL VOC dataset consists of VOC 2007 test, VOC 2007 trainval, and VOC 2012 trainval, which has a total of 21503 images and 20 categories. We first divide the dataset into trainval set and test set in a 9:1 ratio. The trainval set is further divided into a training set and validation set in a 9:1 ratio. For URPC 2017-2019, the training set, validation set, and test set have 18287, 2032, and 2258 images, respectively. For Brackish 2019, the training set, validation set, and test set have 8905, 990, and 1100 images, respectively. For the PASCAL VOC dataset, the training set, validation set, and test set have 17418, 1935, and 2150 images, respectively. During training, the input image size is set to 640×640. We use the stochastic gradient descent (SGD) optimizer with a weight decay of 5e-4, a momentum of 0.9, and a mini-batch size of 16. All models are trained within 500 epochs by setting the initial learning rate to 0.01, which is decreased by a factor of 0.5 per 50 epochs. All experiments are run on a personal computer with NVIDIA GeForce RTX 3090/PCle/SSE2 and Intel(R) Core(TM) i9-10980XE CPU @ 3.00GHz×36.

4.2 Underwater image dataset

In order to better study the problem of underwater object detection, our work is mainly based on URPC and Brackish datasets in this paper. URPC is a public Chinese underwater detection dataset, where underwater images are captured by underwater robots and divers in the near-shallow sea. The underwater image dataset (URPC 2017-2019) consists of URPC 2017(17655), URPC 2018(2901) and URPC 2019(4757), which has a total of 22577 images and 4 categories (Holothurian, Echinus, Scallop, and Starfish) after removing duplicate images. Brackish is a public European underwater detection dataset, where underwater images are collected by using a fixed-point camera and light source in strait waters. The underwater image dataset (Brackish 2019) has a total of 10,995 images and 6 categories: fish, small fish, crab, shrimp, jellyfish, and starfish. Many underwater work studies are based on these two underwater datasets. Figure 9 shows the underwater images in real marine environments and provides the underwater detection results of our work. Obviously, underwater images have low contrast, color cast, texture distortion, and so on. Underwater objects have protective colors and strong concealment capabilities. All of the above phenomena greatly increase the difficulty of underwater object detection. In this paper, our goal is to reduce underwater background interference, improve underwater object perception, and ultimately achieve efficient underwater object detection. As can be seen from the above experimental results, our work has good robustness, adaptability, and generalization in complex underwater detection tasks.

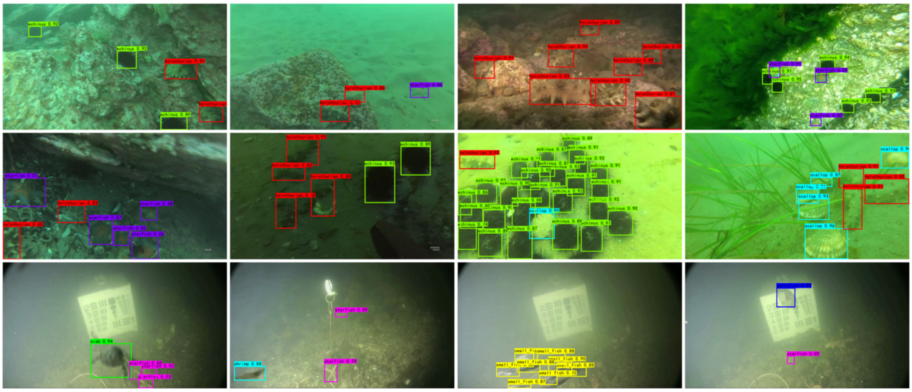

Fig. 9 Various underwater images in real marine environments, where the three lines represent the detection results of YOLOV5+ECGI-HAM1 in URPC 2017-2019, YOLOX+ECGI-HAM1 in URPC 2017-2019, and YOLOV8+ECGI-HAM1 in Brackish 2019, respectively

4.3 Experiments on underwater image dataset

It is worth noting here that the experiments on spatial attention, channel attention, hybrid attention, final design, and attention robustness are based on URPC 2017-2019, and the experiments on training loss and training accuracy are based on Brackish 2019.

Spatial Attention In order to explore more effective spatial attention calculation methods in underwater object detection, we conduct the ablation experiment based on YOLOX-M. We focus on BAM-Var, CBAM-Var, ShAM-Var, SGEM, SCGI-SAM and ECGI-SAM in Table 1, where BAM-Var, CBAM-Var and ShAM-Var represent the spatial attention branches in BAM [26], CBAM [40] and ShAM [49], respectively. SGEM [20] is a lightweight spatial attention module. SCGI-SAM and ECGI-SAM are our attention modules designed in the spatial dimension. These attention modules are applied in the same way, as shown in Fig. 8. It can be seen from Table 1 that other modules are weaker than our modules in improving accuracy. The reason is that although BAM-Var and CBAM-Var use techniques such as the continuous use of multiple dilated convolutions and the combined use of multiple pooling operations in information interaction to enrich the spatial receptive field, they can still only perceive local spatial information. In the information interaction, although ShAM-Var and SGEM avoid the destruction of direct information correspondence by assigning weight and bias to each channel dimension and each channel grouping, it causes the lack of global information interaction. Obviously, it is necessary to perceive global information in complex underwater environments. Compared with SCGI-SAM, ECGI-SAM can compute spatial attention more efficiently, which benefits from the ECGIS we use in information interaction. Compared with SCGIS, ECGIS uses a parameter sharing method in feature extraction, which can more efficiently capture underwater global dependencies and further reduce the number of parameters. Building on Grad-CAM [32], we successfully achieve attention visualization in underwater detection tasks. We focus on selecting the attention layer on the middle branch after YOLOX_M neck for visualization. Figure 10 shows the attention visualization results in complex underwater environments for CBAM-Var, SGEM, SCGI-SAM and ECGI-SAM.

Channel Attention In order to explore more effective channel attention calculation methods in underwater object detection, we also make the ablation experiment based on YOLOX-M.

Table 1 Comparison of different spatial attention modules

Architecture	Param.	GFLOPs	FPS	AP(%)				mAP(%)
				Holothurian	Echinus	Scallop	Starfish	
YOLOX-M	25.28M	73.49	41.96	74.86	76.18	80.09	71.61	75.69
+BAM-Var [26]	25.59M	74.05	37.12	75.35	76.53	80.37	72.17	76.11(+0.42)
+CBAM-Var [40]	25.28M	73.50	39.02	75.60	76.72	80.51	72.45	76.32(+0.63)
+ShAM-Var [49]	25.28M	73.49	41.03	75.61	76.64	80.46	72.35	76.27(+0.58)
+SGEM [20]	25.28M	73.49	40.21	75.68	77.02	80.59	72.53	76.46(+0.77)
+SCGI-SAM(Ours)	28.93M	73.62	36.47	75.84	76.88	80.65	72.73	76.53(+0.84)
+ECGI-SAM(Ours)	25.46M	73.62	35.10	76.20	77.14	80.86	73.14	76.84(+1.15)

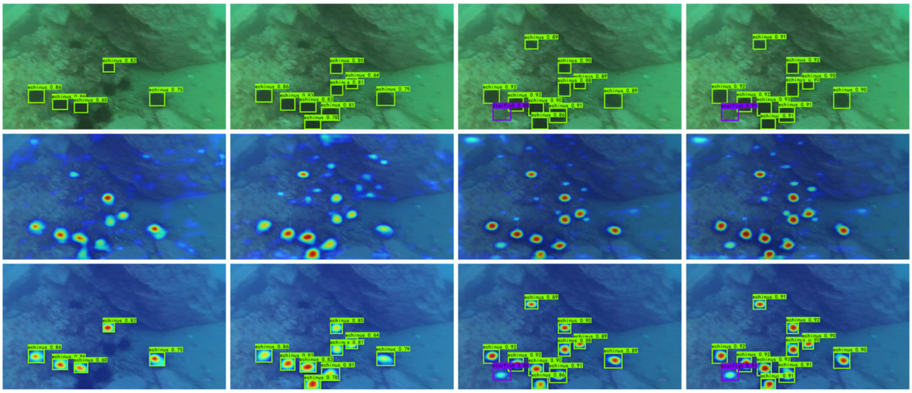

Fig. 10 Attention visualization results. The four columns represent the use of CBAM-Var, SGEM, SCGI-SAM and ECGI-SAM in underwater object detection, respectively

In this experiment, we focus on SCGI-CAM and ECGI-CAM, and compare them with other channel attention modules, including SEM [16], SRM [19], ECAM [39] and FCAM [27]. The experimental results are reported in Table 2. The accuracy gains brought by our attention modules are higher than SEM and FCAM, which indicates that ensuring the direct information correspondence in information interaction is crucial for the application of attention mechanism in underwater object detection. Our attention modules outperform SRM and ECAM, which indicates that achieving global information interaction in the channel dimension is also crucial for underwater object detection. Compared with SCGI-CAM, ECGI-CAM can generate channel attention more suitable for underwater object detection. Like Fig. 10, Fig. 11 shows the attention visualization results for ECAM, SRM, SCGI-CAM and ECGI-CAM.

Hybrid Attention In order to explore more efficient hybrid attention computation methods in underwater object detection, ablation experiments are performed on the basis of YOLOX-M. After research on spatial attention and channel attention, we find that the attention module designed based on ECGIS can show stronger performance in underwater object detection. Therefore, we focus on various hybrid attention modules composed of ECGI-SAM and ECGI-CAM, which include ECGI-HAM1, ECGI-HAM2 and ECGI-HAM3. Table 3 reports the experimental results on BAM [26], CBAM [40], CoAM [14], ShAM [49], and designed attention modules. Compared with BAM and CBAM, designed attention modules can achieve greater gains in improving the accuracy of underwater object detection, which mainly benefits from two points. First, our modules avoid the lack of global information interaction in the spatial dimension. Second, our modules avoid the destruction of direct information correspondence in the channel dimension. Compared with CoAM, designed attention modules also perform better. Although CoAM successfully captures the global location information by aggregating features along two spatial directions respectively, our criss-cross structure can capture the global location information more directly and efficiently in the spatial dimension. Compared with ShAM, proposed attention modules achieve the global interaction of channel information and spatial information. For our hybrid attention modules, we find that ECGI-HAM1 performs the best. Compared with other combinations, the combination of spatial attention and channel attention in series can achieve better performance gains in underwater object detection. Figure 12 shows the attention visualization results for CBAM, CoAM, SCGI-HAM1 and ECGI-HAM1. As can be seen from Figs. 10, 11 and 12, our designed

Table 2 Comparison of different channel attention modules

Architecture	Param.	GFLOPs	FPS	AP(%)				mAP(%)
				Holothurian	Echinus	Scallop	Starfish	
YOLOX-M	25.28M	73.49	41.96	74.86	76.18	80.09	71.61	75.69
+SEM [16]	25.48M	73.50	40.21	75.05	76.45	80.40	71.76	75.92(+0.23)
+SRM [19]	25.29M	73.49	41.23	75.09	76.67	80.64	71.90	76.08(+0.39)
+ECAM [39]	25.28M	73.49	40.87	75.14	76.58	80.54	71.84	76.03(+0.34)
+FCAM [27]	25.48M	73.50	39.07	75.00	76.29	80.35	71.74	75.85(+0.16)
+SCGI-CAM(Ours)	25.59M	73.62	38.62	75.24	76.72	80.71	71.92	76.15(+0.46)
+ECGI-CAM(Ours)	25.40M	73.62	37.16	75.31	76.81	80.82	71.97	76.23(+0.54)

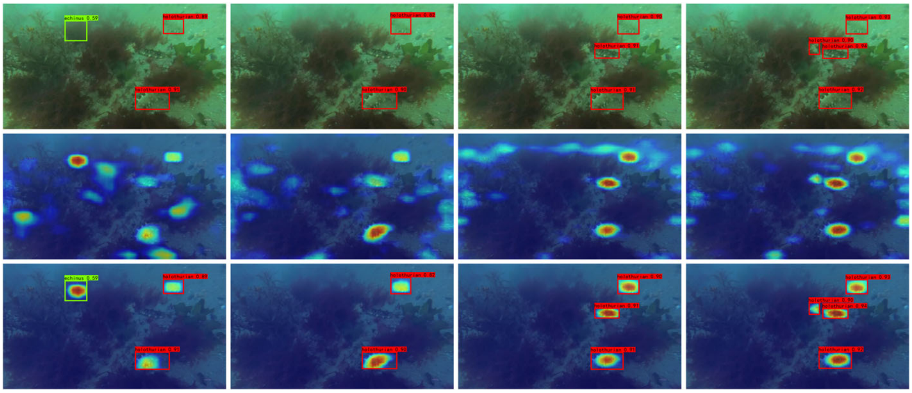

Fig. 11 Attention visualization results. The four columns represent the use of ECAM, SRM, SCGI-CAM and ECGI-CAM in underwater object detection, respectively

selective attention can play a better role in reducing underwater background interference and improving underwater object perception. Obviously, avoiding the destruction of direct information correspondence and the lack of global information interaction in the attention module are crucial for underwater object detection.

Final design After the above research on spatial attention, channel attention and hybrid attention, we finally believe that, compared with other designed modules, ECGI-HAM1 composed of ECGI-SAM and ECGI-CAM in series is more suitable for application in complex underwater environments. To further prove this point, we do some experiments based on YOLOX-S and YOLOX-L. Tables 4 and 5 report the application results of various selective attentions designed using SCGIS and ECGIS for underwater object detection, respectively. It can be seen from Tables 4 and 5 that the detection accuracy can be improved under different network scales by using spatial attention or channel attention, and the gain brought by spatial attention is greater. When we use hybrid attention at different network scales, the detection accuracy is further improved and the combination of spatial attention and channel attention in series can be more beneficial to deal with underwater visual information. Compared with Table 4, the results in Table 5 are generally better. The above experiments prove our point. ECGI-HAM1 shows the best performance in complex underwater environments, and can be plug-and-play in different network scales. Here, we focus on selecting the attention layer on the middle branch after YOLOX_S neck for visualization. Figure 13 shows the attention visualization results in different marine environments for ECGI-HAM1. As can be seen from Fig. 13, for underwater object detection, our work achieves ideal experimental results in different marine environments.

Attention Robustness To verify attention robustness, we integrate ECGI-HAM1 into a wide variety of YOLO detectors. The specific application of attention modules is elaborated in subsection 3.3. In order to achieve a fair comparison, all detectors are experimented with similar network sizes. Table 6 reports the underwater detection performance of ECGI-HAM1 on YOLOV3, YOLOV4, YOLOV5-L and YOLOX-L. For YOLOV3 and YOLOV4, the input image size is set to 608 × 608. For other YOLO detectors, the input image size is set to 640 × 640. Table 7 reports the underwater detection performance of ECGI-HAM1 on YOLOV5-X,

Table 3 Comparison of different hybrid attention modules

Architecture	Param.	GFLOPs	FPS	AP(%)				mAP(%)
				Holothurian	Echinus	Scallop	Starfish	
YOLOX-M	25.28M	73.49	41.96	74.86	76.18	80.09	71.61	75.69
+BAM [26]	25.79M	74.05	35.36	75.79	76.99	80.86	72.58	76.56(+0.87)
+CBAM [40]	25.67M	73.50	36.41	76.24	77.38	81.24	73.04	76.98(+1.29)
+CoAM [14]	25.44M	73.51	35.83	76.75	77.83	81.66	73.58	77.46(+1.77)
+ShAM [49]	25.28M	73.49	39.77	77.07	77.85	81.54	73.84	77.58(+1.89)
+ECGI-HAM1(Ours)	25.48M	73.64	33.05	77.33	78.35	82.15	74.19	78.01(+2.32)
+ECGI-HAM2(Ours)	25.48M	73.64	32.62	77.13	78.17	81.98	73.98	77.82(+2.13)
+ECGI-HAM3(Ours)	25.48M	73.64	32.34	76.92	77.98	81.81	73.75	77.62(+1.93)

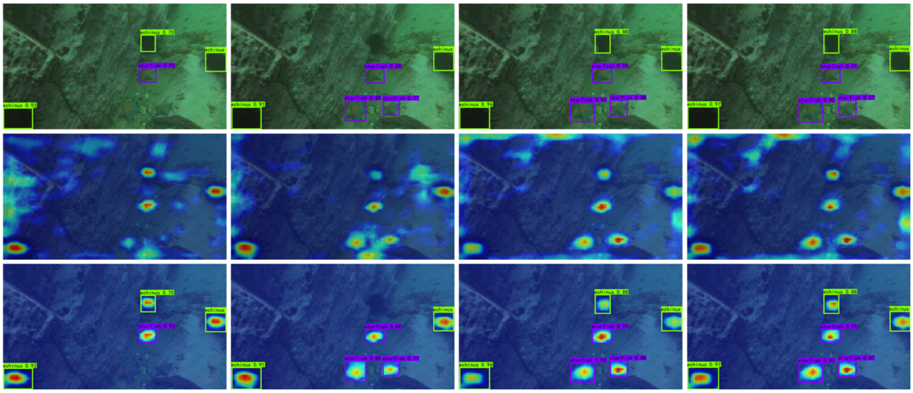

Fig. 12 Attention visualization results. The four columns represent the use of CBAM, CoAM, SCGI-HAM1 and ECGI-HAM1 in underwater object detection, respectively

Table 4 Comparison of various attention modules based on SCGIS under different network scales

Architecture	Param.	GFLOPs	FPS	mAP (%)
YOLOX-S	8.94M	26.64	56.97	70.58
+ SCGI-SAM	12.58M	26.75	49.93	71.59(+1.01)
+ SCGI-CAM	9.13M	26.75	53.01	71.13(+0.55)
+ SCGI-HAM1	12.73M	26.77	46.87	72.69(+2.11)
+ SCGI-HAM2	12.73M	26.77	46.73	72.41(+1.83)
+ SCGI-HAM3	12.73M	26.77	46.40	72.35(+1.77)
YOLOX-L	54.15M	155.29	32.54	80.76
+ SCGI-SAM	57.80M	155.42	29.53	81.45(+0.69)
+ SCGI-CAM	54.60M	155.42	30.57	81.14(+0.38)
+ SCGI-HAM1	58.20M	155.45	27.56	82.21(+1.45)
+ SCGI-HAM2	58.20M	155.45	27.42	82.00(+1.24)
+ SCGI-HAM3	58.20M	155.45	27.10	81.96(+1.20)

Table 5 Comparison of various attention modules based on ECGIS under different network scales

Architecture	Param.	GFLOPs	FPS	mAP (%)
YOLOX-S	8.94M	26.64	56.97	70.58
+ ECGI-SAM	9.06M	26.75	48.48	71.90(+1.32)
+ ECGI-CAM	9.00M	26.75	51.43	71.21(+0.63)
+ ECGI-HAM1	9.07M	26.77	44.51	73.18(+2.60)
+ ECGI-HAM2	9.07M	26.77	44.39	72.89(+2.31)
+ ECGI-HAM3	9.07M	26.77	43.96	72.64(+2.06)
YOLOX-L	54.15M	155.29	32.54	80.76
+ ECGI-SAM	54.33M	155.42	28.10	81.67(+0.91)
+ ECGI-CAM	54.28M	155.42	30.06	81.18(+0.42)
+ ECGI-HAM1	54.35M	155.45	26.62	82.60(+1.84)
+ ECGI-HAM2	54.35M	155.45	26.51	82.36(+1.60)
+ ECGI-HAM3	54.35M	155.45	26.23	82.17(+1.41)

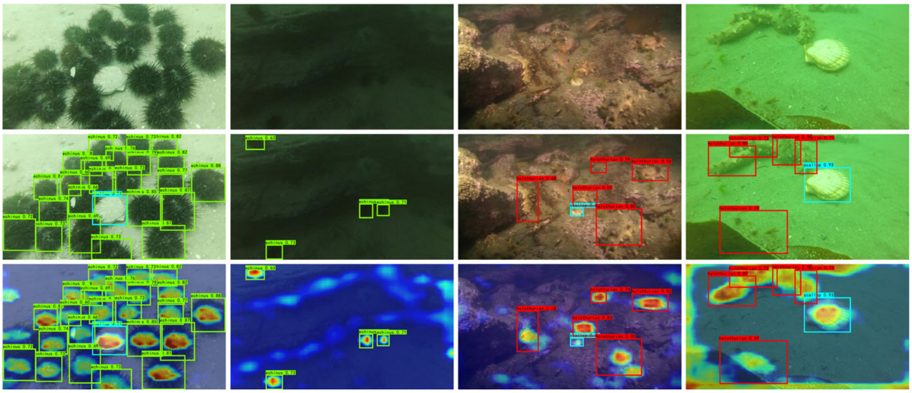

Fig. 13 Attention visualization results in different marine environments for ECGI-HAM1

YOLOV7-X and YOLOX-X. As can be seen from Tables 6 and 7, our attention module can show good robustness in different YOLO detectors. Our selective attention significantly improves the underwater detection accuracy. Although the designed ECGI-HAM1 brings additional parameters and computations, the negative impact of these costs is negligible. To further demonstrate the advantages of ECGI-HAM1, we focus on testing on the basis of YOLOX in complex underwater environments, as shown in Fig. 14. From the subjective comparison results, it can be seen that the underwater detection accuracy is significantly improved by combining our selective attention. ECGI-HAM1 can indeed effectively reduce the underwater background interference and improve the underwater object perception, which meets the real-time and accuracy requirements of underwater detection tasks.

In order to further prove the rationality and effectiveness of our work, we provide the graphs of training loss and training accuracy. Figure 15(left) shows the epoch-loss result of YOLOV8+ECGI-HAM1 during the learning process. It can be seen from train loss and val loss that our training process is stable and convergent, and there is no overfitting problem. Figure 15(right) shows the epoch-mAP results of YOLOV8 and YOLOV8+ECGI-HAM1 in the Brackish underwater dataset. It can be seen from the experimental results that our proposed attention can effectively improve the accuracy of underwater detection.

Table 6 The underwater detection performance of ECGI-HAM1 on YOLOV3, YOLOV4, YOLOV5-L and YOLOX-L

Detector	Param.	FLOPs	FPS	mAP(%)
YOLOV3 [30]	61.54M	139.76G	14.32	75.18
+ECGI-HAM1	61.72M	140.05G	9.27	76.39(+1.21)
YOLOV4 [2]	63.95M	127.60G	20.93	77.27
+ECGI-HAM1	64.14M	127.89G	16.69	78.28(+1.01)
YOLOV5-L [17]	46.15M	108.04G	34.15	79.03
+ECGI-HAM1	46.34M	108.32G	28.14	80.35(+1.32)
YOLOX-L [10]	54.15M	155.29G	32.54	80.76
+ECGI-HAM1	54.35M	155.45G	26.62	82.60(+1.84)

Table 7 The underwater detection performance of ECGI-HAM1 on YOLOV5-X, YOLOV7-X and YOLOX-X

Detector	Param.	FLOPs	Memory	mAP(%)
YOLOV5-X [17]	86.24M	204.14G	1248.91M	83.24
+ECGI-HAM1	86.53M	204.45G	1318.98M	83.54(+0.30)
YOLOV7-X [36]	70.83M	188.46G	885.92M	84.01
+ECGI-HAM1	70.91M	188.59G	920.96M	84.12(+0.11)
YOLOX-X [10]	99.00M	281.46G	1497.85M	84.32
+ECGI-HAM1	99.29M	281.77G	1567.92M	84.77(+0.45)

4.4 Experiments on PASCAL VOC dataset

In Table 8, We integrate ECGI-HAM1 with different YOLO detectors and test the performance of original detectors and attention detectors on the VOC detection task. All detectors

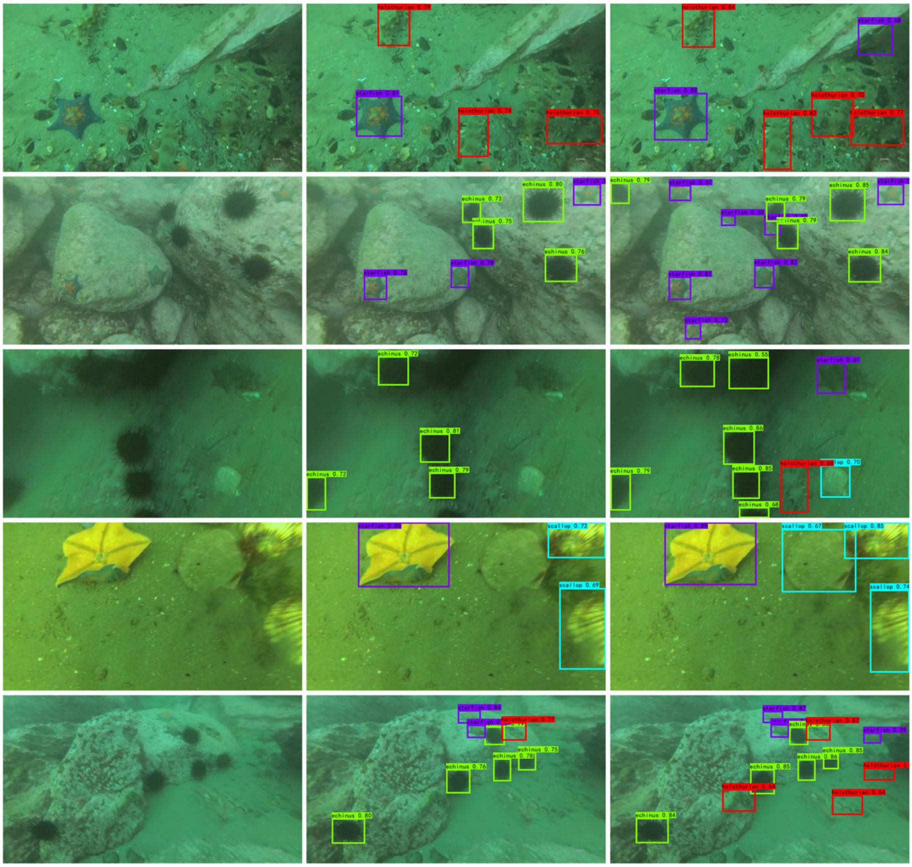

Fig. 14 Subjective comparative experiments on underwater detection tasks (URPC 2017-2019). The three columns show the original images, the YOLOX detector results and our detector results (YOLOX+ECGI-HAM1), respectively

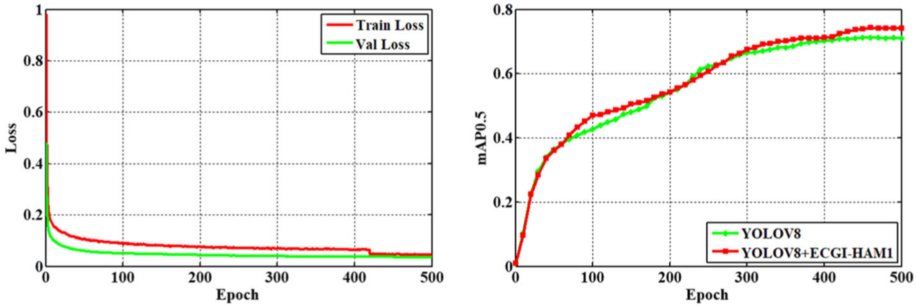

Fig. 15 The training loss and training accuracy of the proposed work(YOLOV8+ECGI-HAM1) on Brackish 2019

used here are the same as in Table 6. Compared with the original detectors, the attention detectors achieve ideal performance gains in detection accuracy. It is worth noting that the detection performance enhancement is not due to the simple capacity increase but due to the designed selective attention ECGI-HAM1. The experimental results in Table 8 demonstrate the generalization ability of ECGI-HAM1 on different detection tasks. After further analysis of Tables 6 and 8, we find that the combination of ECGI-HAM1 and YOLO has a stronger performance gain in our underwater detection task than in the PASCAL VOC detection task. This suggests that our designed selective attention is more needed in complex underwater environments. In other words, ECGI-HAM1 can make a greater contribution to solving the problems of strong underwater background interference and weak underwater feature discriminability.

5 Conclusion

In this paper, we first proposed the criss-cross global interaction strategy (CGIS). Our strategy can effectively avoid the destruction of direct information correspondence caused by the dimensionality reduction interaction strategy and the lack of global information interaction caused by the local interaction strategy. According to different parameter allocation methods, CGIS was further divided into standard criss-cross global interaction strategy (SCGIS) and efficient criss-cross global interaction strategy (ECGIS). Compared with SCGIS, ECGIS

Table 8 Comparison results of different detectors on the PASCAL VOC dataset

Detector	Param.	FLOPs	FPS	Memory	mAP(%)
YOLOV3 [30]	61.63M	140.01G	17.21	942.39M	77.80
+ECGI-HAM1	61.81M	140.30G	13.17	992.98M	78.40(+0.60)
YOLOV4 [2]	64.04M	127.85G	22.33	1295.63M	80.62
+ECGI-HAM1	64.22M	128.14G	17.22	1346.22M	81.17(+0.55)
YOLOV5-L [17]	46.24M	108.19G	36.41	850.84M	83.71
+ECGI-HAM1	46.42M	108.48G	32.19	906.90M	84.13(+0.42)
YOLOX-L [10]	54.16M	155.36G	34.47	1030.10M	85.33
+ECGI-HAM1	54.31M	155.59G	28.01	1086.15M	85.96(+0.63)

achieved better information interaction with fewer parameters. We then designed the criss-cross global interaction-based selective attention in different target dimensions. Specifically, there are SCGI-SAM and ECGI-SAM in the spatial dimension, SCGI-CAM and ECGI-CAM in the channel dimension, and SCGI-HAM1, SCGI-HAM2, SCGI-HAM3, ECGI-HAM1, ECGI-HAM2, and ECGI-HAM3 in the hybrid dimension. Our selective attention can effectively reduce underwater background interference and improve underwater object perception. We finally combined the designed attention modules with YOLO detectors. The combination of ECGI-HAM1 and YOLO achieved a good balance of accuracy and real-time in underwater detection tasks. Our work provided a more significant performance gain for underwater detection tasks and brought some performance improvements for other detection tasks. The experimental results show that our work makes important progress in achieving efficient underwater object detection. Our selective attention shows good robustness in various YOLO detectors and exhibits ideal generalization in different detection tasks.

In general, perceiving global underwater information content and ensuring direct underwater information correspondence are crucial for underwater detection tasks. Our criss-cross interaction structure and parameter-sharing strategy can effectively capture global underwater dependencies and further reduce the underwater attention parameters. At different network scales, the use of spatial attention or channel attention can improve underwater detection accuracy, and spatial attention brings greater underwater performance gain. Hybrid attention can further improve underwater detection performance, and the series combination of spatial attention and channel attention is more conducive to processing underwater visual information. In underwater detection environments, our work has demonstrated excellent performance in terms of parameters, computations, memory consumption, and detection accuracy. However, our high-strength detail design may lead to excessive feature calibration and introduce some performance interference in simple detection tasks.

In the future, we will continue to work on exploring the potential of attention mechanisms in underwater object detection. We plan to further combine enhanced attention and selective attention to propose a more powerful attention module, which will achieve both global correlation-based information enhancement and global importance-based information selection. We hope that the underwater detection task can be better accomplished by combining stronger attention modules with state-of-the-art detectors.

Acknowledgements The authors gratefully acknowledge the financial supports from the National Natural Science Foundation of China under Grant 61370142, Grant 61802043, Grant 61272368, Grant 62176037 and Grant 62002041, in part by the Fundamental Research Funds for the Central Universities under Grant 3132016352 and Grant 3132021238, in part by the Dalian Science and Technology Innovation Fund under Grant 2018J12GX037, Grant 2019J11CY001 and Grant 2021JJ12GX028, in part by Liaoning Revitalization Talents Program under Grant XLYC1908007, in part by the Liaoning Doctoral Research Start-up Fund Project Grant 2021-BS-075, and in part by the China Postdoctoral Science Foundation under Grant 3620080307.

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request. URPC dataset: http://www.cnurpc.org/. Brackish dataset: https://www.kaggle.com/datasets/aalborguniversity/brackish-dataset.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Bhaumik G, Verma M, Govil MC, Vipparthi SK (2022) Hyfinet: hybrid feature attention network for hand gesture recognition. Multimedia Tools and Applications, 1–20
2. Bochkovskiy A, Wang CY, Liao HYM (2020) Yolov4: Optimal speed and accuracy of object detection. arXiv:2004.10934
3. Brackish dataset. https://www.kaggle.com/datasets/aalborguniversity/brackish-dataset (2023)
4. Cai Z, Vasconcelos N (2018) Cascade r-cnn: Delving into high quality object detection. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 6154–6162
5. Cao P, Xie FX, Zhang SC, Zhang ZP, Zhang JF (2022) Msanet: Multiscale attention networks for image classification. Multimedia Tools and Applications, 1–20
6. Chen L, Zhou FX, Wang SK, Dong JY, Li N, Ma HP, Wang X, Zhou HY (2022) Swipenet: Object detection in noisy underwater scenes. Pattern Recognit 132:108926
7. Chen Y, Xia SX, Zhao JQ, Zhou Y, Niu Q, Yao R, Zhu DJ, Chen H (2022) Adversarial learning-based skeleton synthesis with spatial-channel attention for robust gait recognition. Multimedia Tools and Applications, 1–16
8. Everingham M, Gool LV, Williams CKI, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88(2):303–338
9. Everingham M, Eslami SMA, Gool LV, Williams CKI, Winn J, Zisserman A (2015) The pascal visual object classes challenge: A retrospective. Int J Comput Vis 111(1):98–136
10. Ge Z, Liu S, Wang F, Li Z, Sun J (2021) Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430
11. Girshick R (2015) Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp 1440–1448
12. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 580–587
13. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 770–778
14. Hou Q, Zhou D, Feng J (2021) Coordinate attention for efficient mobile network design. In: Proc IEEE/CVF Conf Comput Vis Pattern Recognit (CVPR), pp 13713–13722
15. Huang Z, Wang X, Huang L, Huang C, Wei Y, Liu W (2019) Ccnet: Criss-cross attention for semantic segmentation. In: Proc IEEE/CVF International Conference on Computer Vision (ICCV), pp 603–612
16. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 7132–7141
17. Jocher G et al (2021) Yolov5. https://github.com/ultralytics/yolov5
18. Kong T, Sun F, Liu H, Jiang Y, Li L, Shi J (2020) Foveabox: Beyound anchor-based object detection. IEEE Trans on Image Processing 29:7389–7398
19. Lee H, Kim HE, Nam H (2019) Srm: A style-based recalibration module for convolutional neural networks. In: Proc IEEE/CVF International Conference on Computer Vision (ICCV), pp 1854–1862
20. Li X, Hu XL, Yang J (2019) Spatial group-wise enhance: Improving semantic feature learning in convolutional networks. arXiv preprint arXiv:1905.09646
21. Lin WH, Zhong JX, Liu S, Li T, Li G (2020) Roimix: proposal-fusion among multiple images for underwater object detection. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 2588–2592. IEEE
22. Liu CW, Wang ZH, Wang SJ, Tang T, Tao YL, Yang CF, Li HJ, Liu X, Fan X (2021) A new dataset, poisson gan and aquanet for underwater object grabbing. IEEE Transactions on Circuits and Systems for Video Technology 32(5):2831–2844
23. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: Single shot multibox detector. In: European Conference on Computer Vision, pp 21–37. Springer
24. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: Hierarchical vision transformer using shifted windows. In: Proc IEEE/CVF International Conference on Computer Vision (ICCV), pp 10012–10022
25. Mao YX, Zhang TZ, Fu B, Thanh DN (2022) A self-attention based wasserstein generative adversarial networks for single image inpainting. Pattern Recognition and Image Analysis 32(3):591–599
26. Park J, Woo S, Lee JY, Kweon IS (2018) Bam: Bottleneck attention module. In: Proceedings of the British Machine Vision Conference (BMVC)
27. Qin Z, Zhang P, Wu F, Li X (2021) Fcanet: Frequency channel attention networks. In: Proc IEEE/CVF International Conference on Computer Vision (ICCV), pp 783–792
28. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 779–788

29. Redmon J, Farhadi A (2017) Yolo9000: better, faster, stronger. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 7263–7271
30. Redmon J, Farhadi A (2018) Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767
31. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: Towards realtime object detection with region proposal networks. Advances in neural information processing systems 28
32. Selvaraju RR, Cogswell M, Das RA, Vedantam PD, Batra D (2017) Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proc IEEE International Conference on Computer Vision (ICCV), pp 618–626
33. Song PH, Li PT, Dai LH, Wang T, Chen Z (2023) Boosting r-cnn: Reweighting r-cnn samples by rpn's error for underwater object detection. Neurocomputing 530:150–164
34. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proc IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1–9
35. Underwater robot picking contest. http://www.cnurpc.org/ (2023)
36. Wang CY, Bochkovskiy A, Liao HY (2022) Yolov7: Trainable bag-offreebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696
37. Wang HB, Jiang GQ, Peng JJ, Deng RX, Fu XP (2022) Towards adaptive consensus graph: Multi-view clustering via graph collaboration. IEEE Transactions on Multimedia
38. Wang X, Girshick R, Gupta A, He K (2018) Non-local neural networks. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 7794–7803
39. Wang Q, Wu B, Zhu P, Li P, Hu Q (2020) Eca-net: Efficient channel attention for deep convolutional neural networks. In: Proc IEEE/CVF Conf Comput Vis Pattern Recognit (CVPR)
40. Woo S, Park J, Lee JY, Kweon IS (2018) Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 3–19
41. Xie S, Girshick R, Dollár P, Tu Z, He K (2017) Aggregated residual transformations for deep neural networks. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 1492–1500
42. Xu SB, Zhang MH, Song W, Mei HB, He Q, Liotta A (2023) A systematic review and analysis of deep learning-based underwater object detection. Neurocomputing
43. Xu FQ, Wang HB, Peng JJ, Fu XP (2021) Scale-aware feature pyramid architecture for marine object detection. Neural Comput & Applic 33:3637–3653
44. Yang J, Li C, Zhang P, Dai X, Xiao B, Yuan L, Gao J (2021) Focal self-attention for local-global interactions in vision transformers. arXiv preprint arXiv:2107.00641
45. Yeh CH, Lin CH, Kang LW, Huang CH, Lin MH, Chang CY, Wang CC (2021) Lightweight deep neural network for joint learning of underwater object detection and color conversion. IEEE Transactions on Neural Networks and Learning Systems 33(11):6129–6143
46. Yolov6: a single-stage object detection framework dedicated to industrial applications. https://github.com/meituan/YOLOv6 (2022)
47. Yolov8 (2023) https://github.com/ultralytics/ultralytics
48. Yu HF, Li XB, Feng YK, Han S (2023) Multiple attentional path aggregation network for marine object detection. Appl Intell 53(2):2434–2451
49. Zhang QL, Yang YB (2021) Sa-net: Shuffle attention for deep convolutional neural networks. In: Proc IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 2235–2239. IEEE
50. Zhang S, Wen L, Bian X, Lei Z, Li SZ (2017) Mask r-cnn. In: Proc IEEE International Conference on Computer Vision (ICCV), pp 2961–2969
51. Zhang S, Wen L, Bian X, Lei Z, Li SZ (2018) Single-shot refinement neural network for object detection. In: Proc IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 4203–4212
52. Zhao HS, Jia JY, Koltun V (2020) Exploring self-attention for image recognition. In: Proc IEEE/CVF Conf Comput Vis Pattern Recognit, pp 10076–10085

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Xin Shen[1] · Huibing Wang[1] · Yafeng Li[1] · Tianzhu Gao[1] · Xianping Fu[1,2]

Xin Shen
shenxin@dlmu.edu.cn

Huibing Wang
huibing.wang@dlmu.edu.cn

Yafeng Li
yfl@dlmu.edu.cn

Tianzhu Gao
brain@dlmu.edu.cn

[1] The School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China

[2] The Peng Cheng Laboratory, Shenzhen 518000, China