Journal of
Marine Science and Engineering

MDPI

Article

YOLO-GE: An Attention Fusion Enhanced Underwater Object Detection Algorithm

Qiming Li * and Hongwei Shi

College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China; 202330310267@stu.shmtu.edu.cn
* Correspondence: qmli@shmtu.edu.cn

Abstract: Underwater object detection is a challenging task with profound implications for fields such as aquaculture, marine ecological protection, and maritime rescue operations. The presence of numerous small aquatic organisms in the underwater environment often leads to issues of missed detections and false positives. Additionally, factors such as the water quality result in weak target features, which adversely affect the extraction of target feature information. Furthermore, the lack of illumination underwater causes image blur and low contrast, thereby increasing the difficulty of the detection task. To address these issues, we propose a novel underwater object detection algorithm called YOLO-GE (GCNet-EMA). First, we introduce an image enhancement module to mitigate the impact of underwater image quality issues on the detection task. Second, a high-resolution feature layer is added into the network to improve the problems of missed detections and false positives for small targets. Third, we propose GEBlock, an attention-based fusion module that captures long-range contextual information and suppresses noise from lower-level feature layers. Finally, we combine an adaptive spatial fusion module with the detection head to filter out conflicting feature information from different feature layers. Experiments on the UTDAC2020, DUO and RUOD datasets show that the proposed method achieves an optimal detection accuracy.

Keywords: object detection; feature fusion; yolov8; attention mechanism

Citation: Li, Q.; Shi, H. YOLO-GE: An Attention Fusion Enhanced Underwater Object Detection Algorithm. *J. Mar. Sci. Eng.* **2024**, *12*, 1885. https://doi.org/10.3390/jmse12101885

Received: 21 September 2024
Revised: 9 October 2024
Accepted: 18 October 2024
Published: 21 October 2024

1. Introduction

Water covers about 71% of the Earth's surface, and humans have engaged in a wide range of production and research activities across various aquatic environments, including rivers, lakes and oceans. Due to inherent physiological limitations, underwater operations present greater challenges compared to activities conducted on the water's surface. As the foundation for subsequent operations, underwater object detection has garnered increasing attention from researchers in recent years. It aims to utilize various advanced technological methods to automatically recognize and locate objects in underwater environments. These objects may include aquatic organisms, sunken ships, underwater personnel and seabed equipment, and the ability to detect them is of significant importance for underwater resource exploration, water resource monitoring, aquaculture and marine biodiversity conservation. With the advancements in image acquisition hardware and improvements in computational capabilities, computer vision-based object detection has emerged as a major research direction and has made substantial progress, and has been widely applied across various industries on both land and water, including autonomous driving, intelligent video surveillance, ship detection, and so on. However, there are still many difficulties to be solved in underwater object detection. Firstly, poor lighting conditions have a serious impact on underwater optical imaging, and factors such as water quality and flow can also affect the clarity and quality of images. Secondly, underwater targets such as marine life and corroded shipwrecks exhibit diverse scales, shapes and appearances. Therefore, underwater object detection is a challenging yet promising research direction.

With ongoing technological advancements and innovations, it is anticipated that more efficient and accurate underwater object detection systems will be developed in the future, providing robust support for marine resource development and research.

2. Related Work

Object detection is a challenging task in the field of computer vision, aiming to extract target feature information from images or videos and achieve target localization. Over the past few decades, it is generally believed that object detection has undergone the following two periods: the traditional algorithms and the deep learning-based algorithms.

Traditional object detection algorithms primarily rely on manually extracted features, and the entire algorithmic process can be summarized into three steps. First, select the regions of interest, choosing areas that may contain objects. Second, extract features from the regions that may contain objects. Finally, perform detection and classification on the extracted features. P. Viola et al. [1] proposed the Viola–Jones (VJ) detector, which uses a sliding window approach to check if a target exists within the window. However, due to its massive computational demands, this detector has very high time complexity. To address this issue, the VJ detector significantly improved the detection speed by combining the following three key techniques: integral image, feature selection and detection cascade. N. Dalal et al. [2] proposed the Histogram of Oriented Gradients (HOG) detector. This method improves the detection accuracy by calculating overlapping local contrast normalization on a dense grid of uniformly spaced cells. Thus, the HOG detector is an algorithm that extracts feature histograms based on local pixel blocks, showing good stability under local deformations and lighting variations. P. Felzenszwalb et al. [3] proposed an object detection method based on a multi-scale, deformable parts model called the Deformable Parts Model (DPM), which can be seen as an extension of the HOG method, consisting of a root filter and several part filters. It improves the detection accuracy through techniques such as hard negative mining, bounding box regression and context initialization.

Deep learning-based object detection methods are mainly divided into single-stage and two-stage categories. Single-stage methods directly produce the final prediction results through a single forward propagation process. First, the input image is preprocessed and then target features are extracted through operations such as convolution and attention mechanisms. These features are fed into the object detection head, which performs target localization and classification and ultimately outputs the prediction results. Single-stage object detection algorithms focus more on speed and real-time performance. Two-stage methods, on the other hand, process in phases. The first stage generates candidate boxes, extracts regions that may contain objects by using a region proposal network and performs pooling operations to map the features to a fixed size. In the second stage, it extracts target feature information from the candidate regions by using a feature extraction network, and then performs target classification and localization. Finally, non-maximum suppression is used to remove redundant candidate boxes, and the final results are obtained. Currently, single-stage methods mainly include the SSD [4] and the YOLO series algorithms [5], while RCNN [6], Fast RCNN [7], Faster RCNN [8] and Mask RCNN [9] belong to the two-stage methods.

With the development of object detection technology, researchers are committed to exploring how to improve the accuracy of object detection in complex environments. Typically, attention mechanisms are used within networks to improve the extraction of target features, enhance semantic information and increase the precision of detection tasks. Woo et al. [10] designed a CBAM module combining channel and spatial attention mechanisms, effectively addressing some issues in attention mechanisms. The channel attention module weights channel features and the spatial attention module weights spatial features, enabling the network to more accurately extract important features and improve the detection accuracy. Cao et al. [11] proposed GCNet, which combines the long-range dependency modeling of NLNet [12] and the channel attention adjustment of SENet [13], enhancing the network's overall ability to model global context information. Lee et al. [14]

improved the SENet channel attention module and proposed the EffectiveSE attention module, effectively avoiding the loss of channel information and helping to enhance the representation of feature information. Misra et al. [15] introduced TripletAttention, which captures cross-dimensional interactions to calculate attention weights using a three-branch structure. This method processes input features through rotation and residual transformations, effectively establishes cross-dimensional dependencies, encodes channel and spatial features and maintains a low computational overhead. Chen et al. [16] proposed a new hybrid attention transformer, HAT, which combines channel attention and self-attention methods and leverages the complementary advantages of both in global statistics and local fitting capabilities. Ouyang et al. [17] proposed a novel attention module, EMA, which reshapes feature information on partial channel dimensions and groups them into multiple sub-features, preserving channel features and reducing computational resource waste. This method recalibrates the channel weights of each parallel branch by encoding global feature information and captures pixel-level relationships through cross-dimensional interactions. Wan et al. [18] introduced the MLCA attention module, which integrates local and global levels of spatial and channel feature information through mixed local channel attention, enhancing the network's expressive capability. Yu et al. [19] proposed an attention module called the MCA, which reduces the model size and improves the network accuracy through a three-branch structure of multi-dimensional collaboration. In the MCA module, dual cross-dimensional feature responses are merged through an adaptive combination mechanism and a gating mechanism is designed to extract local feature information interaction during excitation transformation.

In underwater image detection tasks, methods such as feature fusion and attention mechanisms are commonly used to improve the extraction of target feature information. Additionally, image enhancement techniques can address issues of insufficient lighting and poor image quality in underwater images, thereby enhancing the accuracy of detection tasks. Song et al. [20] proposed Boosting R-CNN, a two-stage detection method. This method employs a new region proposal network called RetinaRPN, which has strong capabilities to detect blurry and low-contrast underwater images. Moreover, the method introduces a probabilistic reasoning pipeline and Boosting reweighting, helping the detector to make more accurate predictions based on uncertainties when dealing with blurry objects in underwater images. Guo et al. [21] improved YOLOv8 [22] by introducing the FasterNet [23] module, combining a fast feature pyramid network with a lightweight C2f structure. This enhances the ability to extract target features from underwater images while reducing the network complexity. Lin et al. [24] proposed a network based on DETR [25], designing a learnable query recall mechanism to improve the network's convergence speed by adding supervision signals to the queries, thereby enhancing the underwater detection accuracy. Additionally, a lightweight adapter was introduced to extract target feature information, improving the detection capability for small and irregular underwater targets. Wang et al. [26] proposed DJL-Net, an end-to-end model, which uses a dual-branch joint learning network, combining underwater image enhancement and underwater object detection through multitask joint learning. DJL-Net uses enhanced images produced by the image processing module to supplement features lost due to the degradation of the original underwater images, thereby improving the detection accuracy. Liang et al. [27] proposed RoIAttn to improve the accuracy of general detectors in underwater environments. RoIAttn aims to effectively capture relationships at the region of interest (RoI) level, achieving the decoupling of regression and classification tasks through a dual-head structure. Considering the difficulty of convolution in accurately regressing coordinate information, RoIAttn introduces positional encoding in the regression branch to provide more precise regression box position information. Dai et al. [28] observed that the edges of underwater objects are distinctive and can be differentiated from low-contrast environments based on their edges. Therefore, they proposed an edge-guided representation learning network called ERL-Net, aiming to achieve distinctive representation learning and aggregation under the guidance of edge cues.

The current underwater object detection algorithms do not perform well in complex underwater environments, which is usually due to the network's insufficient feature extraction capability and the pervasive noise information within the network. Therefore, improving the accuracy and robustness of underwater object detection algorithms is a research direction worth exploring.

The proposed YOLO-GE in this study effectively enhances the network's feature extraction capability by introducing image enhancement techniques, high-resolution feature layers, an attention fusion enhancement module and an adaptive fusion detection head. This approach increases the focus on small objects, suppresses the propagation of noise information within the network and improves the detection accuracy of object detection algorithms in complex underwater environments.

3. Methodology

To address the issue of low detection accuracy caused by poor underwater image quality and the presence of many small aquatic organisms, an attention fusion enhancement model based on YOLOv8s, called YOLO-GE (GCNet-EMA), is proposed, as shown in Figure 1.

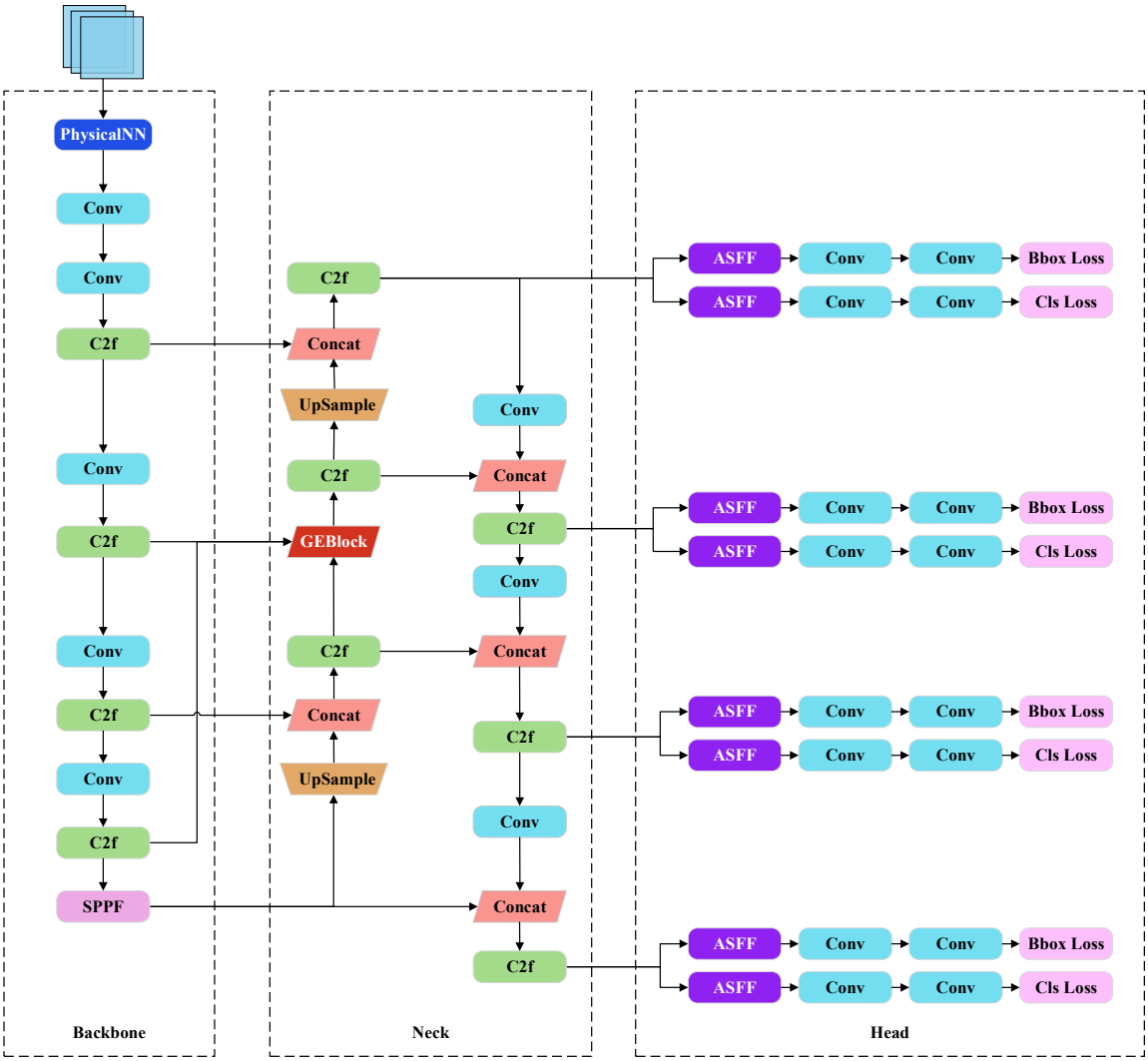


Figure 1. The network structure of YOLO-GE consists of the backbone, neck and head. The modules using white font are our main improvement points.

At the start of the backbone network, an image enhancement module (PhysicalNN) is introduced. In the neck network, an attention fusion enhancement module (GEBlock) is designed to incorporate high-resolution feature layers. In the detection head, an Adaptive Spatial Feature Fusion (ASFF) module is added. The number of trainable parameters of YOLO-GE is as follows: the backbone network has 5.1 M trainable parameters; the neck network has 4.3 M parameters; since the detection head integrates ASFF, the number of trainable parameters is relatively large, reaching 6.9 M.

3.1. Image Enhancement Module

Due to the lack of lighting in underwater images, the images become blurry and have a blue–green color cast, which reduces the image contrast and decreases the accuracy of the detection tasks. To address this issue, we introduce the image enhancement module PhysicalNN [29], as shown in Figure 2. First, the input image is passed through the Backscatter Estimation module to estimate the environmental illumination. Then, using the estimated environmental illumination and the input image, the Direct-transmission Estimation module estimates the direct transmission map. Finally, a reconstruction operation is performed on the estimation results. This module effectively mitigates the effects of the underwater environment, enriches the colors and enhances the image contrast, thereby significantly improving the detection accuracy. The visual effects of image enhancement using PhysicalNN are shown in Figure 3.

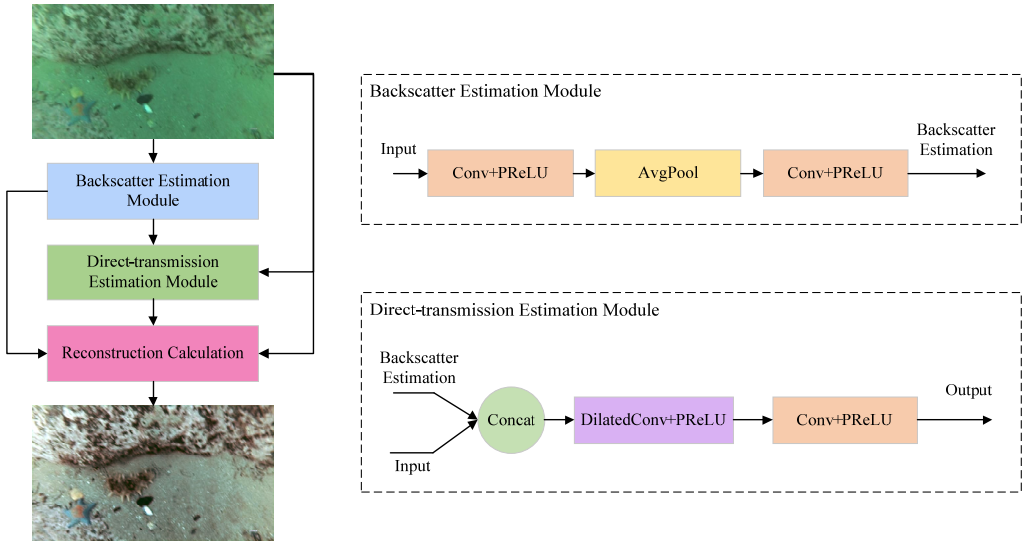


Figure 2. Structure of PhysicalNN. This module mainly consists of the Backscatter Estimation module, Direct-transmission Estimation Module and a reconstruction calculation.

Figure 3. Comparison of image enhancement effects with PhysicalNN. The first row shows the original images and the second row shows the images after enhancement.

3.2. High-Resolution Feature Layer

During the process of extracting object features in the network, the target feature information gradually weakens after several downsampling operations. For small objects, the feature information in the image is inherently weak and this information further diminishes, or even disappears, after multiple downsampling steps, resulting in poor detection performance for small targets. For example, if the input image size is 640 × 640 pixels, after 4 downsampling operations, the image size becomes 40 × 40 pixels. A target that originally measures 64 × 64 pixels in the input image would be reduced to 4 × 4 pixels after 4 downsampling steps, making feature extraction very challenging. Although high-level feature maps have a larger receptive field and rich semantic information, their lower resolution makes them less suitable for detecting small objects. On the other hand, shallow feature maps, which have a higher resolution, are more favorable for extracting and detecting small object features. In underwater object detection tasks, there are many small aquatic organisms in the underwater environment. In the UTDAC2020 dataset, the target size distribution is shown in Table 1, where the smallest target in the Echinus category occupies only 2 pixels and the smallest target in the Scallop category occupies only 12 pixels. Small targets are mainly concentrated in the Echinus category. Therefore, adding high-resolution feature layers helps capture the features of small objects and improves the overall performance of the network. We introduced a high-resolution feature layer, P2, with a resolution of 160 × 160 pixels into the network, as shown in Figure 4. The feature information from the B2 layer of the feature extraction network is fused with the feature information from the P3 layer of the neck to obtain the P2 layer, and then the feature information from the P2 layer is fed into the detection head for object localization and classification.

Table 1. Distribution of target sizes in the UTDAC2020 dataset. Max, Min and Mean represent the maximum pixel value, minimum pixel value and average pixel value of targets in the corresponding category. S/M/L indicate the number of small, medium and large targets in the corresponding category, respectively.

Category	Max (px)	Min (px)	Mean (px)	S/M/L
Echinus	1,920,240	2	17,487	2796/13,973/7737
Starfish	1,241,536	200	23,942	591/4191/4004
Holothurian	1,283,205	198	26,682	62/3529/2552
Scallop	1,825,000	12	38,195	602/1428/5220

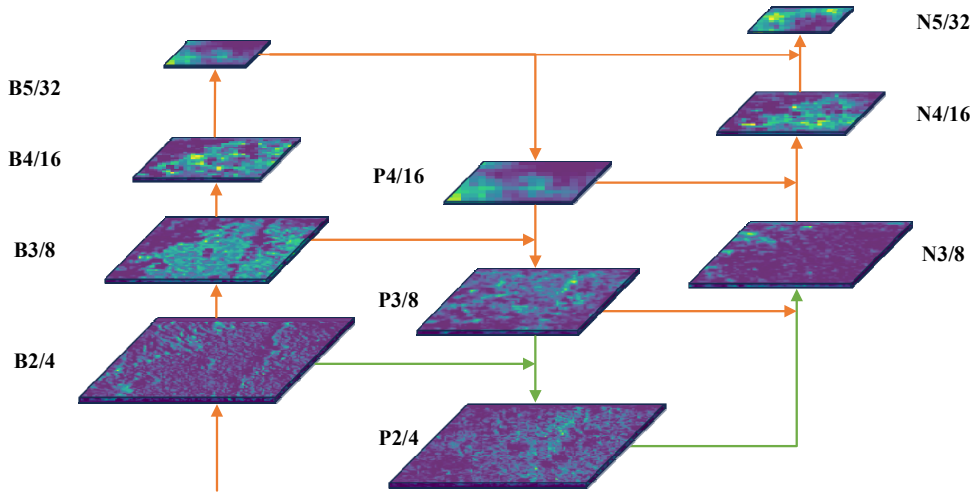


Figure 4. Adding the P2 high-resolution feature layer. B represents the feature extraction network, P represents FPN and N represents PAN. The orange connection lines represent the feature information transmission paths in the original structure, while the green connection lines indicate the feature information transmission paths introduced by the high-resolution feature layers.

3.3. Attention Fusion Enhancement

YOLOv8 [22] uses a PAN-FPN structure, which simply fuses feature information from different scales. However, this simple fusion has obvious issues: it cannot effectively differentiate the importance of feature information from different scales, resulting in fused feature information that does not highlight the important information at each level. Additionally, the fusion of high-level and low-level features may introduce more noise, reducing the network performance. To address this problem, we designed an attention fusion enhancement module (GEBlock), and Figure 5 shows the overall structure of this module. The GE module integrates the GCNet [11] and EMA [17] attention modules, effectively enhancing the ability to capture long-range dependencies and utilizing cross-dimensional interactions to extract pixel-level relationships. After introducing the GE module into the network, it effectively suppresses the noise information from the lower-level feature layers during feature fusion, thereby improving the overall detection performance. The feature information for this module comes from high-level feature information in the backbone network, low-level feature information and the previous layer's feature information from the GE module. Figure 1 shows the position of the GE module in the network and the sources of its feature information inputs.

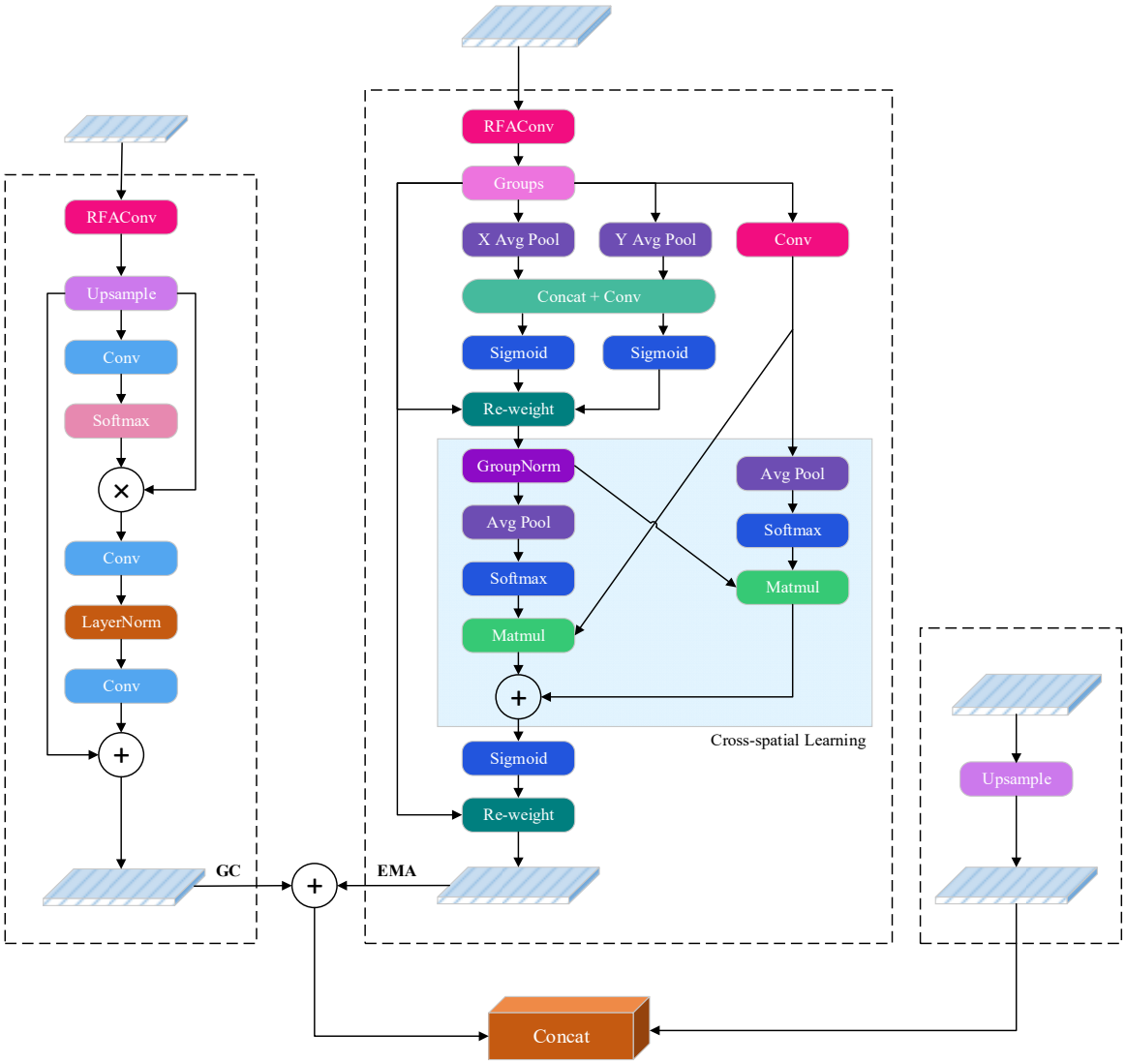


Figure 5. Structure of the GE module. The input feature information of the GE module comes from three different feature layers, which are processed by the GC attention, EMA attention and upsampling operations, and are finally fused together.

Equations (1)–(3) provide the mathematical representation of the GE module, where X_1 represents the high-level feature layer information, X_2 represents the low-level feature layer information and X_3 represents the feature information from the previous layer of the GE module. X_G and X_E are intermediate variables and Y is the final output. *RFAConv* represents receptive field attention convolution operations, *GCNet* represents global context attention operations and *EMA* represents multi-scale attention operations. *Concat* denotes the concatenation of feature maps, *ADD* denotes the addition of feature maps and *UpSample* denotes the upsampling operation.

$$X_G = GCNet(UpSample(RFAConv(X_1))) \tag{1}$$

$$X_E = \mathrm{EMA}(RFAConv(X_2)) \tag{2}$$

$$Y = Concat(ADD(X_G, X_E), UpSample(X_3)) \tag{3}$$

In the GE module, the receptive field attention convolution (RFAConv) [30] is also introduced, which considers long-range information through global pooling and solves the problem of convolution kernel parameter sharing in traditional convolutions. This helps to highlight feature information at different locations in the image. The structure of RFAConv is shown in Figure 6. First, the input feature map is passed through average pooling to reduce the spatial dimensions. Then, the pooled feature map is processed through three different group convolutions and a Softmax activation function to generate three attention maps. Next, these attention maps are used to reweight the feature map that has been processed through group convolution, allowing different parts of the feature map to be adjusted based on the importance indicated by the attention maps. After reweighting, the feature map is reshaped and then subjected to a convolution operation to obtain the final output.

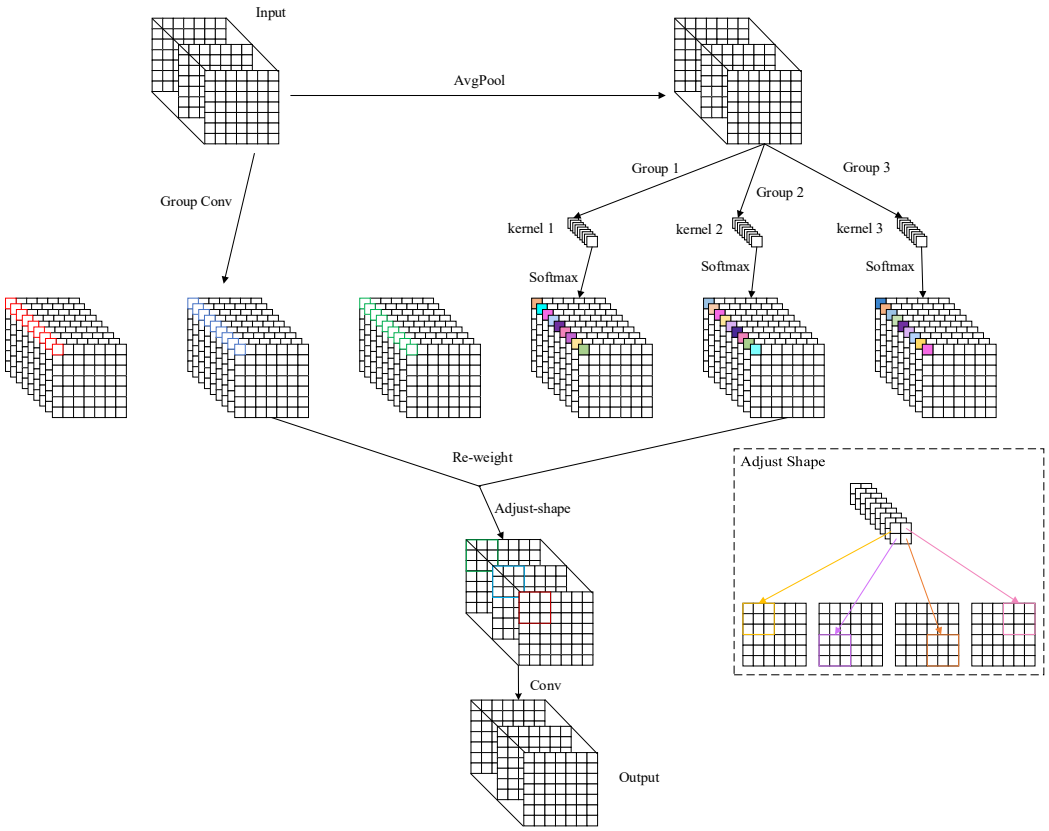


Figure 6. Structure of RFAConv. RFAConv emphasizes the importance of different features within the receptive field sliding window, addressing the issue of convolutional kernel parameter sharing.

For the high-level feature layer, the feature map size is 20 × 20 pixels. In the GE module, the input feature information first passes through RFAConv, which not only reduces the dimensionality of the input features but also effectively highlights feature information from different positions in the image. Then, the downsampled feature information undergoes an upsampling operation, enlarging the image to 80 × 80 pixels, which helps to improve the performance of small-object detection. The upsampled feature map is then processed by the global context attention mechanism, GCNet. In traditional self-attention models, each position can only interact with other positions in the sequence, whereas the global context attention mechanism allows each position to directly interact with the entire sequence, thus better capturing global semantic information. By introducing the global context attention mechanism, GCNet, the model can more effectively handle long-range dependencies, model global context information, enhance the understanding of global information and improve the performance of detection tasks.

For the low-level feature layer in the feature extraction network, the feature map size is 80 × 80 pixels. When input to the GE module, it first undergoes the RFAConv operation to reduce the feature dimensions and then is processed by the EMA. This effectively suppresses noise information from the low-level feature layer and enriches the semantic information. EMA is a high-performance multi-scale attention mechanism that, compared to traditional channel or spatial attention mechanisms, not only retains feature information of each channel but also reduces the computational burden. Its key lies in reshaping some channels into batch dimensions and grouping the channel dimensions into multiple sub-features, allowing spatial semantic features to be better distributed within each feature group. Specifically, the channel weights are recalibrated by global information in one parallel branch, while the output features from the two parallel branches are further aggregated through cross-dimensional interaction to capture pixel-level pairwise relationships.

After being processed by RFAConv and the attention mechanism, the low-level and high-level features from the backbone network have their feature maps merged through an addition operation within the GE module. This not only retains more detailed information from the low-level features but also effectively reduces the propagation of noise within the network. The input from the previous layer of the GE module contains richer semantic information. First, it undergoes an upsampling operation to enlarge the image to 80 × 80 pixels and then it is concatenated with the feature map obtained from the addition operation, further enriching the semantic information.

3.4. ASFFHead

In YOLO-GE, we use feature information from four different scales for object detection and localization. To handle potential conflicts between these four feature layers, we introduce ASFF [31]. ASFF effectively filters out conflicting information and retains useful information for combination, thereby enhancing scale invariance. ASFF integrates and adjusts features from other levels to the same resolution at a specific level, then finds the optimal fusion method through training to achieve adaptive spatial fusion. This approach helps us to achieve more effective fusion among multi-scale features, improving the detection performance.

In YOLO-GE, we integrate ASFF into the detection head to form the ASFFHead, as shown in Figure 7. During the detection task, the ASFFHead effectively reduces conflicts between feature information at different scales, enriching semantic information and thereby improving the accuracy of the object detection. The four different scale feature layers from PAN-FPN are used as detection layers and the red marked box in Figure 7 shows the fusion method of the four different scale features in ASFF-4. These four scales of features are represented as $X^{1\to4}$, $X^{2\to4}$, $X^{3\to4}$ and $X^{4\to4}$, and these features are multiplied by their corresponding weights and summed together. Finally, the output of ASFF-4 is obtained, with its mathematical representation shown in Equation (4), as follows:

$$y_{ij}^{l} = \alpha_{ij}^{4} \cdot x_{ij}^{1\to l} + \beta_{ij}^{4} \cdot x_{ij}^{2\to l} + \gamma_{ij}^{4} \cdot x_{ij}^{3\to l} + \eta_{ij}^{4} \cdot x_{ij}^{4\to l} \tag{4}$$

where y_{ij}^{l} represents the (i, j) vector of the output feature map y^{l} between channels. The weight parameters of the four different scale feature maps are given by α_{ij}^{4}, β_{ij}^{4}, γ_{ij}^{4} and η_{ij}^{4}, respectively. l denotes the level layer and, in YOLO-GE, $l \in \{1, 2, 3, 4\}$.

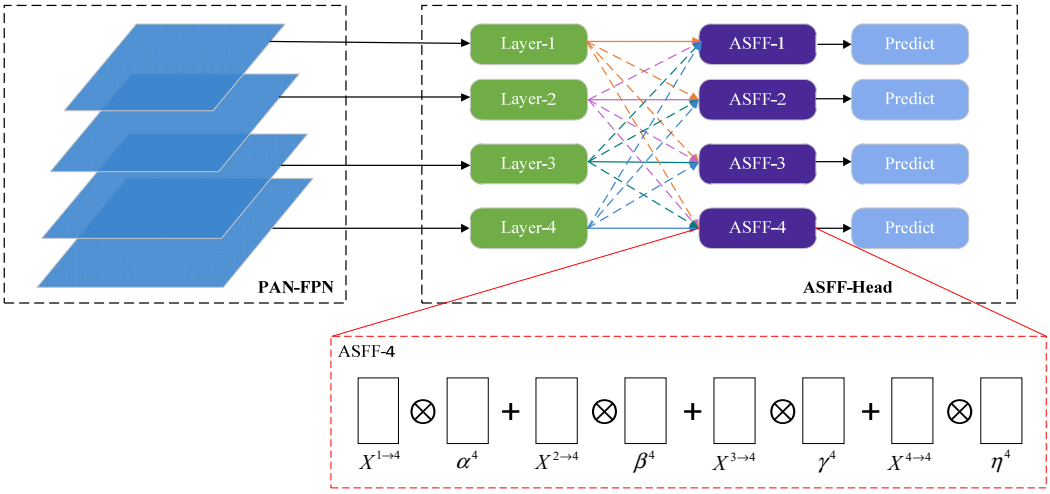


Figure 7. Structure of ASFF. The red marked box below shows the fusion method of feature information from four different scales in ASFF-4. The orange, purple, green, and blue arrows represent the feature information originating from layers 1 to 4, respectively.

4. Experiments

We conducted a large number of experiments to validate the effectiveness of the proposed method, and no pre-trained weights were used in any of the experiments.

4.1. Dataset

The datasets used in this study include the following: the UTDAC2020 dataset, DUO dataset [32] and RUOD dataset [33]. The UTDAC2020 dataset consists of 6,461 images covering the following four categories: sea urchins, starfish, sea cucumbers and scallops. This dataset contains images of the following five different resolutions: 3840 × 2160, 1920 × 1080, 720 × 405, 586 × 480 and 704 × 576. The DUO dataset contains 7782 images covering the following four categories: sea urchin, starfish, sea cucumber and scallop. Most of the images in this dataset have a resolution of 720 × 405 pixels. The RUOD dataset consists of 14,000 images, with the majority of images having a resolution of 1920 × 1080 pixels. This dataset includes the following 10 common aquatic categories: sea cucumber, sea urchin, scallop, starfish, fish, coral, diver, squid, sea turtle and jellyfish. The quantity distributions of species categories in the three datasets are shown in Figure 8.

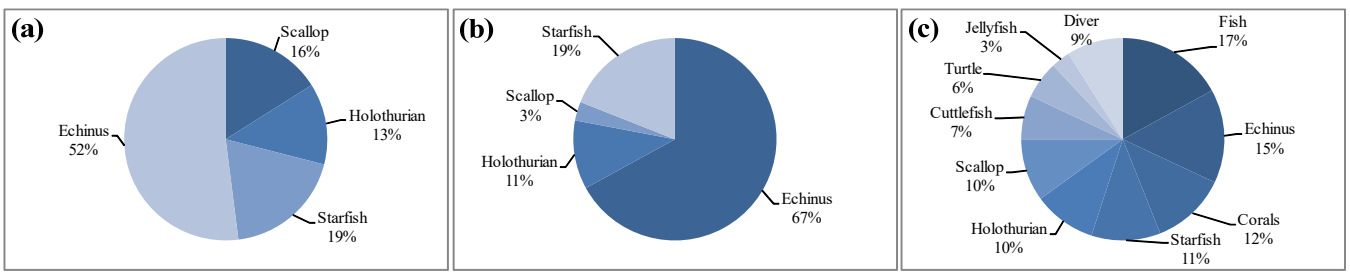


Figure 8. Distribution of the number of samples in each category within the datasets. (**a**) UTDAC2020 dataset, (**b**) DUO dataset and (**c**) RUOD dataset.

4.2. Experimental Settings

The experimental environment is as follows: the operating system is Linux (Ubuntu 20.04), the processor is a seven-core Intel(R) Xeon(R) CPU E5-2680 v4, the graphics card is an NVIDIA RTX 3080 (20 GB) and the memory is 30 GB. We used Python 3.8 to implement the methods and conducted experiments in the PyTorch 2.0.0 and CUDA 11.8 environments. Due to GPU memory limitations, we set the batch size to eight and chose the SGD optimizer. To prevent overfitting, we conducted preliminary experiments to adjust the number of training epochs on different datasets. On the UTDAC2020 dataset, the loss function converged after 150 epochs, so we set the training epochs for this dataset to 150. On the DUO and RUOD datasets, the loss function converged after 300 epochs, so we set the epochs to 300. Additionally, we used Mosaic data augmentation, which further helped us to avoid overfitting. All other parameters were set to the default values of YOLOv8.

4.3. Evaluation Metrics

Because this research mainly focused on the improvement of the detection accuracy, we did not introduce the discussion of the computation amount and model complexity. The mean average precision (mAP) was used as the evaluation metric. This metric measures the average performance of the detection method across all categories. We evaluated the performance using the following three different Intersection over Union (IoU) thresholds for the mAP metric: mAP_{50}, mAP_{75} and $mAP_{50:95}$. The formula for the mean average precision (mAP) is as follows:

$$AP = \int_0^1 PdR \tag{5}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{6}$$

In this formula, AP_i represents the average precision (AP) for the i-th category and N denotes the number of categories in the dataset. The average precision measures the performance of the model for a specific category, while the mean average precision (mAP) is the average of average precisions across all categories, providing a single value for the overall performance of the model across all categories.

In Formula (5), P represents precision and R represents recall. Their mathematical representations are as follows:

$$P = \frac{TP}{TP + FP} \times 100\% \tag{7}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{8}$$

In this context, TP (true positive) refers to the number of correctly predicted positive samples by the model. FP (false positive) indicates the number of negative samples that the model incorrectly predicted as positive.

4.4. Results and Discussion

In order to validate the effectiveness of the improvements employed, we conducted ablation experiments on YOLO-GE. Additionally, we compared our method with the latest approaches on the UTDAC2020, DUO and RUOD datasets. Through extensive experiments, we demonstrated the superiority of the proposed method.

4.4.1. Ablation Experiment

We conducted ablation experiments on the UTDAC2020 dataset to validate the effectiveness of the proposed improvement methods. In Table 2, Method A refers to the introduction of the image enhancement module, Method B refers to the inclusion of the high-resolution feature layer P2, Method C refers to the use of the attention fusion enhancement module and Method D refers to the use of ASFFHead.

Table 2. Ablation study results.

Method	mAP_{50}(%)	mAP_{75}(%)	$mAP_{50:95}$(%)
YOLOv8s	83.9	53	49.6
+A	(+0.4)84.3	(+0.9)53.9	(+0.4)50
+A+B	(+0.2)84.5	(+1.2)55.1	(+0.6)50.6
+A+B+C	(+0.3)84.8	(+0.2)55.3	(+0.3)50.9
+A+B+C+D	(+0.3)85.1	(+2.2)57.5	(+0.8)51.7

Firstly, to address the issue of images appearing overall bluish-green due to factors such as inadequate lighting, we introduced an image enhancement module. Secondly, to improve the accuracy of small-object detection in underwater images, we incorporated the high-resolution feature layer P2. However, the inclusion of the high-resolution feature layer might cause low-level feature noise to propagate through the network. To address this issue, we introduced the Attention Fusion Enhancement GE module, which effectively prevents noise propagation and improves the network's feature extraction capabilities. Finally, to eliminate potential feature information conflicts caused by the introduction of the high-resolution feature layer, we introduced ASFFHead, which effectively helps the network filter out conflicting information and further improves the detection accuracy. Compared to the baseline model YOLOv8s, our proposed method improves mAP_{50}, mAP_{75} and $mAP_{50:95}$ by 1.2%, 4.5% and 2.1%, respectively.

4.4.2. Comparison with the Benchmark Model

We compared our method with the baseline model YOLOv8s on the UTDAC2020, DUO and RUOD datasets, and partial qualitative comparison visualization results are shown in Figure 9. It can be seen that, in terms of the detection accuracy, YOLO-GE is significantly superior to YOLOv8s, effectively improving the detection effect in complex environments.

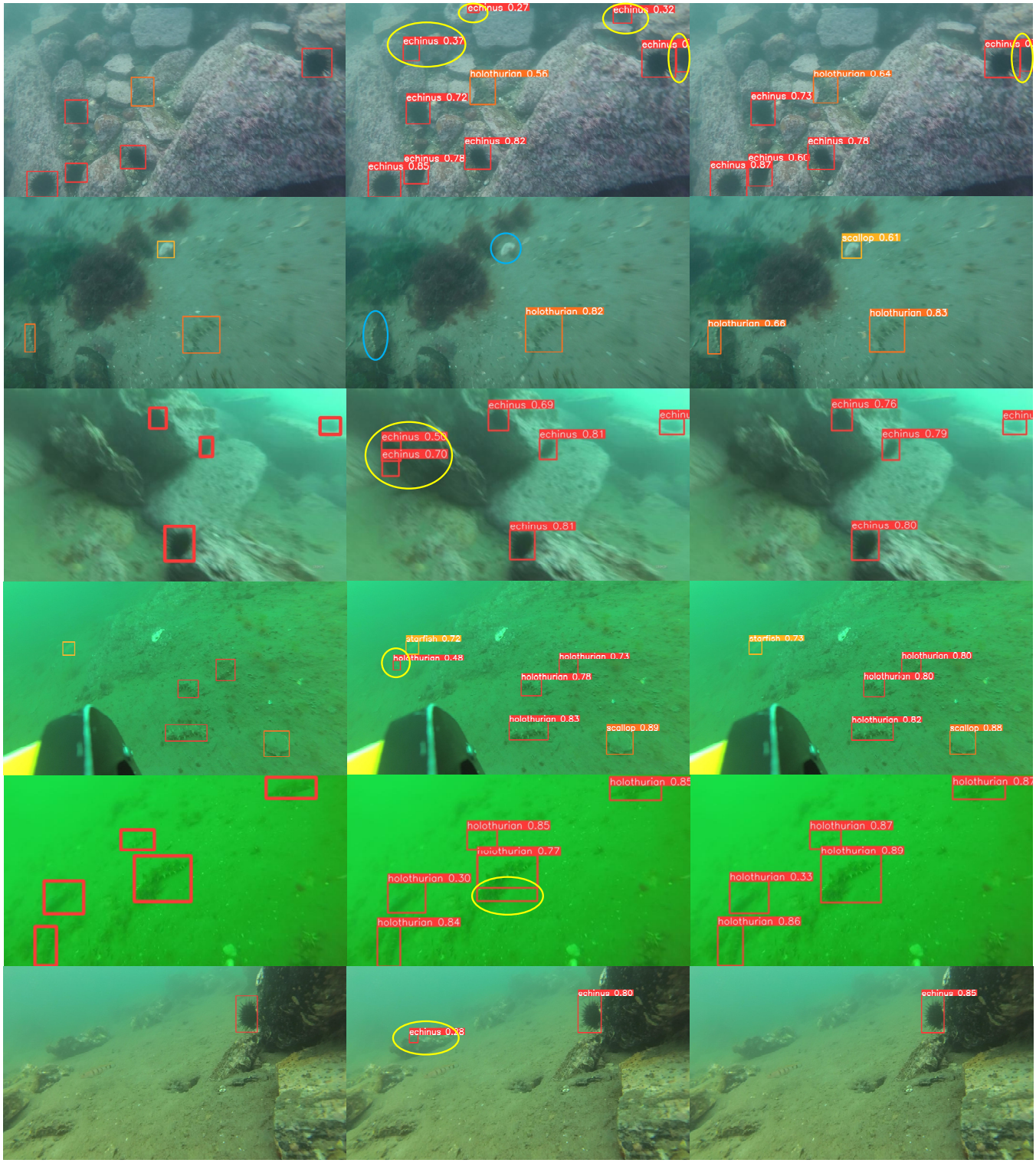
In addition, to better demonstrate the improvement effects of the proposed method, we used the LayerCAM algorithm [34] to visualize the results with heatmaps and compared the YOLO-GE method with the YOLOv8s method. The visualization results are shown in Figure 10. It is evident that YOLO-GE highlights the important features of the target more effectively, leading to an optimal detection performance.

The quantitative comparison results are shown in Table 3. It can be observed that YOLO-GE significantly outperforms YOLOv8s in detection accuracy, with particularly notable improvements in high-precision detection.

Table 3. Comparison of the experimental results with the baseline model.

Dataset	Method	mAP_{50}(%)	mAP_{75}(%)	$mAP_{50:95}$(%)
UTDAC	YOLOv8s	83.9	53	49.6
	YOLO-GE	(+1.2)85.1	(+4.5)57.5	(+2.1)51.7
DUO	YOLOv8s	86.5	76.5	68.7
	YOLO-GE	(+0.2)86.7	(+1.3)77.8	(+1.6)70.3
RUOD	YOLOv8s	86.8	70.2	63.9
	YOLO-GE	(+0.3)87.1	(+2.1)72.3	(+1.6)65.5

However, the improvements of YOLO-GE on the DUO and RUOD datasets are relatively smaller compared to the UTDAC dataset. This is because the DUO dataset has more pronounced class imbalance and more underwater object occlusions than the UTDAC dataset. As for the RUOD dataset, it contains more classes and includes many clearer underwater images with larger underwater objects. Since our proposed improvements are more focused on dealing with turbid underwater images and small objects, the enhancement on this dataset is not as significant.



(a) (b) (c)

Figure 9. Partial detection results. The yellow circles mark false-positive targets, while the blue circles mark missed targets. In the images from rows 1 to 3 and 6, the red annotation box represents echinus, the orange annotation box represents holothurian, and the yellow annotation box represents scallop. In the images from rows 4 and 5, the red annotation box represents holothurian, the orange annotation box represents scallop, and the yellow annotation box represents starfish. (**a**) Ground truth. (**b**) YOLOv8s method. (**c**) YOLO-GE method.

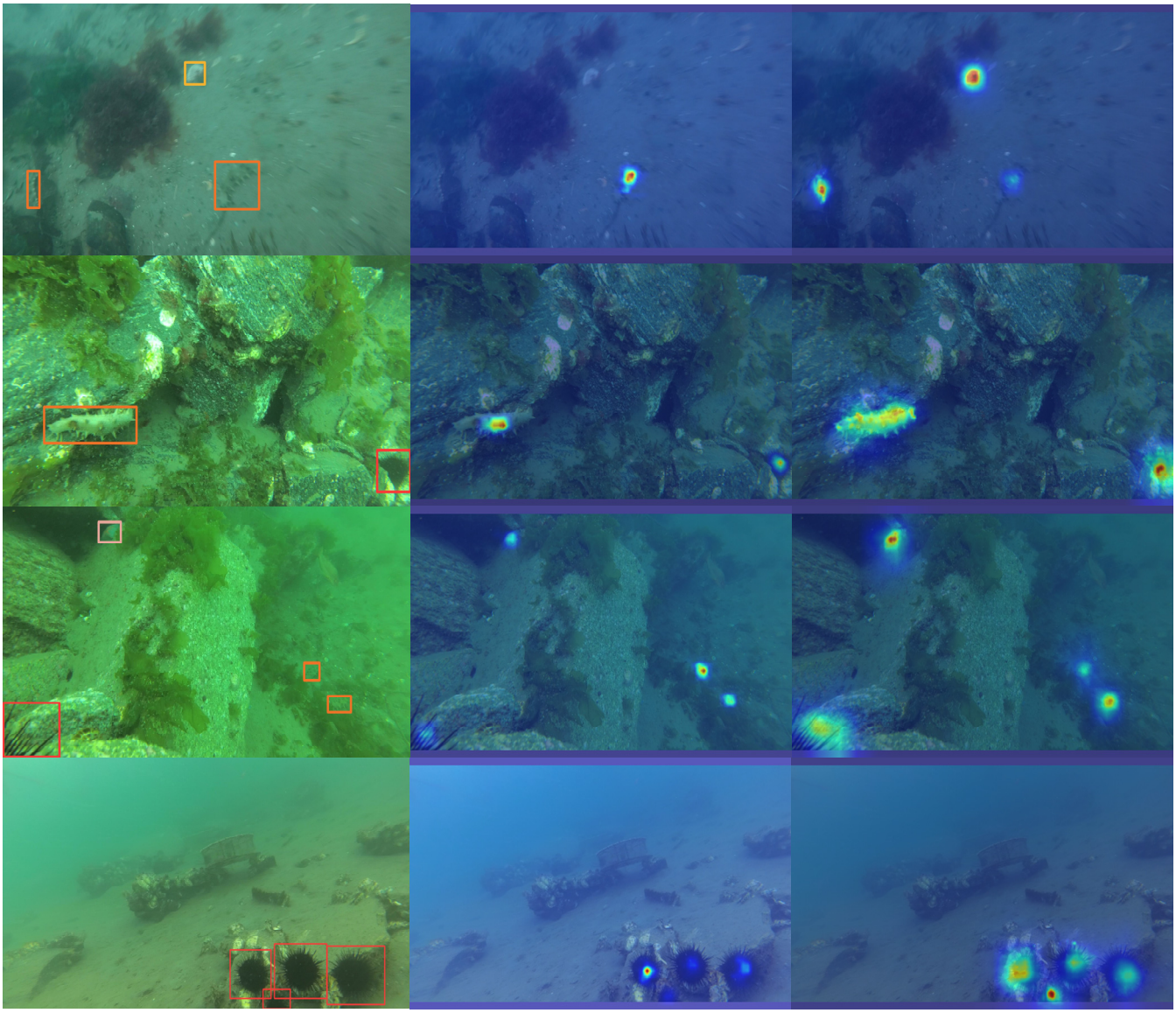


Figure 10. Heatmap visualization. The red box represents echinus, the orange box represents holothurian, the yellow box represents scallop, and the pink rectangular box represents starfish. (**a**) Ground truth; (**b**) YOLOv8s method; (**c**) YOLO-GE method.

4.4.3. Comparison with Other Methods

To comprehensively evaluate the performance of YOLO-GE and validate its generalization ability, we conducted comparative experiments with the latest methods on the UTDAC2020, DUO and RUOD datasets. It can be seen from Table 4 that, on the three datasets, our proposed method achieves the best performance in mAP_{50}, mAP_{75} and $mAP_{50:95}$, with particularly significant improvements in high-precision detection.

The outstanding performance of YOLO-GE on the UTDAC2020, DUO and RUOD datasets demonstrates its strong capability in handling various underwater scenarios, particularly in more challenging and diverse underwater environments. The results indicate that YOLO-GE is highly effective in dealing with turbid images and small objects, as evidenced by its performance on these datasets. Universal detectors like YOLOv8s and DetectoRS perform well across the datasets but generally lag behind YOLO-GE. This highlights YOLO-GE's design advantages, especially for underwater detection in complex or challenging environments.

Table 4. Comparison with other methods on the UTDAC2020, DUO and RUOD datasets.

Method	UTDAC2020			DUO			RUOD		
	mAP_{50}	mAP_{75}	$mAP_{50:95}$	mAP_{50}	mAP_{75}	$mAP_{50:95}$	mAP_{50}	mAP_{75}	$mAP_{50:95}$
Universal Detector									
YOLOv8s [22]	83.9%	53%	49.6%	86.5%	76.5%	68.7%	86.8%	70.2%	63.9%
VFNet [35]	79.3%	44.1%	44.0%	80.8%	68.4%	61.1%	81%	57.5%	53.6%
Faster R-CNN [8]	80.9%	44.1%	44.5%	75.9%	63.1%	54.8%	81.4%	56.4%	51.9%
Deformable DETR [36]	84.1%	47.0%	46.6%	79.0%	62.7%	55.7%	66.8%	39.4%	38.7%
DetectoRS [37]	82.8%	49.9%	47.6%	81.7%	70.5%	62.8%	82.8%	61.9%	56.6%
Cascade R-CNN [38]	81.4%	48.3%	46.0%	79.6%	40.8%	60.4%	81.5%	59.4%	54.4%
RetinaNet [39]	80.4%	42.9%	43.9%	78.2%	64.4%	57.3%	80.6%	55.3%	51.7%
YOLOv10s [40]	83.3%	53.1%	49.1%	85.7%	75.6%	67.5%	85.9%	68.2%	61.9%
YOLO11s [41]	84.4%	53.1%	49.8%	86.2%	76.1%	68.7%	86.8%	69.9%	63.8%
Underwater Detector									
Boosting R-CNN [20]	82.4%	52.5%	48.5%	78.4%	66.1%	57.3%	80.6%	59.5%	53.9%
ERL-Net [28]	82.8%	52.2%	48.4%	81.4%	69.5%	61.2%	83.1%	60.9%	54.8%
RoIAttn [27]	82.0%	47.5%	46.0%	79.5%	66.5%	58.7%	81.7%	57.3%	52.9%
YOLO-GE (Ours)	85.1%	57.5%	51.7%	86.7%	77.8%	70.3%	87.1%	72.3%	65.5%

5. Conclusions

Underwater object detection research is significant for the development and protection of marine resources, as it helps scientists and engineers monitor marine life, underwater terrain and environmental changes more accurately. Additionally, this technology is crucial in the military field, where it can enhance the efficiency and safety of underwater combat and defense systems. Due to the lack of lighting in underwater environments, underwater images often suffer from blurriness and low contrast, increasing the difficulty of underwater object detection tasks. Additionally, the presence of numerous small aquatic organisms, combined with factors like water currents and poor water quality, weakens target features, leading to missed detections and false positives. To address these challenges and improve the detection accuracy, this paper proposes a YOLOv8-based Attention Fusion Enhancement underwater object detection model, named YOLO-GE. First, to improve the overall image quality and enhance the image contrast, an image enhancement module is introduced. Second, a high-resolution feature layer is incorporated to boost the detection capability for small targets. Third, an Attention Fusion Enhancement module is introduced to effectively prevent the propagation of noise from low-level feature layers through the network, and RFAConv is employed to further improve the target feature extraction. Finally, to resolve potential feature information conflicts among the four detection layers, the Adaptive Spatial Feature Fusion Detection Head is introduced, which further enhances the small-object detection capabilities and improves the network's detection accuracy. The effectiveness of the proposed method is validated on the UTDAC2020, DUO and RUOD datasets, achieving optimal mAP performance across all three datasets.

However, the proposed method still has room for improvement. For instance, its detection performance is poor when there is occlusion between underwater objects or when the object surfaces are corroded. Additionally, when there is a class imbalance in the dataset, the improvement in the detection performance is not significant. In future work, we will continue to explore more effective algorithms and techniques to enhance the detection capabilities in more complex underwater environments.

The proposed model can help researchers in the field of marine science to monitor and study marine life and their habitats more accurately. Additionally, it can assist rescue personnel in the swift search and recovery of missing persons underwater. The development of this model is significant for fields such as aquaculture and underwater rescue operations.

Author Contributions: Conceptualization, Q.L. and H.S.; methodology, Q.L. and H.S.; software, H.S.; validation, Q.L. and H.S.; formal analysis, Q.L. and H.S.; investigation, Q.L.; resources, Q.L.;

data curation, H.S.; writing—original draft preparation, H.S.; writing—review and editing, Q.L. and H.S.; visualization, H.S.; supervision, Q.L.; project administration, Q.L.; funding acquisition, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key R&D and Transformation Projects of Xizang (Tibet) Autonomous Region Science and Technology Program, grant number XZ202401ZY0004.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are openly available in the UTDAC2020, DUO and RUOD datasets at https://aistudio.baidu.com/datasetdetail/215376 (accessed on 21 March 2024), https://github.com/chongweiliu/DUO (accessed on 31 March 2024) and https://github.com/dlut-dimt/RUOD (accessed on 21 March 2024), respectively.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Viola, P.; Jones, M. Rapid Object Detection Using a Boosted Cascade of Simple Features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, HI, USA, 8–14 December 2001.
2. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005.
3. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A Discriminatively Trained, Multiscale, Deformable Part Model. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008.
4. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
5. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
6. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
7. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
8. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
9. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
10. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
11. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond. In Proceedings of the 2019 IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019.
12. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-Local Neural Networks. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
13. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
14. Lee, Y.; Park, J. CenterMask: Real-Time Anchor-Free Instance Segmentation. In Proceedings of the 2020 IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
15. Misra, D.; Nalamada, T.; Arasanipalai, A.U.; Hou, Q. Rotate to Attend: Convolutional Triplet Attention Module. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision, Virtual, 5–9 January 2021.
16. Chen, X.; Wang, X.; Zhou, J.; Qiao, Y.; Dong, C. Activating More pixels in Image Super-Resolution Transformer. In Proceedings of the 2023 IEEE Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
17. Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient Multi-Scale Attention Module with Cross-Spatial Learning. In Proceedings of the 2023 IEEE International Conference on Acoustics, Speech and Signal Processing, Rhodes Island, Greece, 4–10 June 2023.
18. Wan, D.; Lu, R.; Shen, S.; Xu, T.; Lang, X.; Ren, Z. Mixed Local Channel Attention for Object Detection. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106442. [CrossRef]
19. Yu, Y.; Zhang, Y.; Cheng, Z.; Song, Z.; Tang, C. MCA: Multidimensional Collaborative Attention in Deep Convolutional Neural Networks for Image Recognition. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107079. [CrossRef]
20. Song, P.; Li, P.; Dai, L.; Wang, T.; Chen, Z. Boosting R-CNN: Reweighting R-CNN Samples by RPN's Error for Underwater Object Detection. *Neurocomputing* **2023**, *530*, 150–164. [CrossRef]

21. Guo, A.; Sun, K.; Zhang, Z. A Lightweight YOLOv8 Integrating FasterNet for Real-Time Underwater Object Detection. *J. Real-Time Image Process.* **2024**, *21*, 49. [CrossRef]
22. Ultralytics YOLO. Available online: https://github.com/ultralytics/ultralytics (accessed on 24 April 2024).
23. Chen, J.; Kao, S.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023.
24. Lin, X.; Huang, X.; Wang, L. Underwater object detection method based on learnable query recall mechanism and lightweight adapter. *PLoS ONE* **2024**, *19*, e0298739. [CrossRef] [PubMed]
25. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
26. Wang, B.; Wang, Z.; Guo, W.; Wang, Y. A Dual-Branch Joint Learning Network for Underwater Object Detection. *Knowl.-Based Syst.* **2024**, *293*, 111672. [CrossRef]
27. Liang, X.; Song, P. Excavating RoI Attention for Underwater Object Detection. In Proceedings of the 2022 IEEE International Conference on Image Processing, Bordeaux, France, 16–19 October 2022.
28. Dai, L.; Liu, H.; Song, P.; Tang, H.; Ding, R.; Li, S. Edge-Guided Representation Learning for Underwater Object Detection. *CAAI Trans. Intell. Technol.* **2024**. *Early View*. [CrossRef]
29. Chen, X.; Zhang, P.; Quan, L.; Yi, C.; Lu, C. Underwater Image Enhancement Based on Deep Learning and Image Formation Model. *arXiv* **2021**, arXiv:2101.00991.
30. Zhang, X.; Liu, C.; Yang, D.; Song, T.; Ye, Y.; Li, K.; Song, Y. RFAConv: Innovating Spatial Attention and Standard Convolutional Operation. *arXiv* **2023**, arXiv:2304.03198.
31. Liu, S.; Huang, D.; Wang, Y. Learning Spatial Fusion for Single-Shot Object Detection. *arXiv* **2019**, arXiv:1911.09516.
32. Liu, C.; Li, H.; Wang, S.; Zhu, M.; Wang, D.; Fan, X.; Wang, Z. A Dataset and Benchmark of Underwater Object Detection for Robot Picking. In Proceedings of the 2021 IEEE International Conference on Multimedia & Expo Workshops, Shenzhen, China, 5–9 July 2021.
33. Fu, C.; Liu, R.; Fan, X.; Chen, P.; Fu, H.; Yuan, W.; Zhu, M.; Luo, Z. Rethinking General Underwater Object Detection: Datasets, Challenges, and Solutions. *Neurocomputing* **2023**, *517*, 243–256. [CrossRef]
34. Jiang, P.-T.; Zhang, C.-B.; Hou, Q.; Cheng, M.-M.; Wei, Y. LayerCAM: Exploring Hierarchical Class Activation Maps for Localization. *IEEE Trans. Image Process.* **2021**, *30*, 5875–5888. [CrossRef] [PubMed]
35. Zhang, H.; Wang, Y.; Dayoub, F.; Sünderhauf, N. VarifocalNet: An IoU-Aware Dense Object Detector. In Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 10–25 June 2021.
36. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv* **2020**, arXiv:2010.04159.
37. Qiao, S.; Chen, L.-C.; Yuille, A. DetectoRS: Detecting Objects With Recursive Feature Pyramid and Switchable Atrous Convolution. In Proceedings of the 2021 IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 10–25 June 2021.
38. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
39. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
40. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
41. Ultralytics YOLO. Available online: https://github.com/yt7589/yolov11 (accessed on 7 October 2024).