Journal of
Marine Science and Engineering

MDPI

Article

Underwater Target Detection Lightweight Algorithm Based on Multi-Scale Feature Fusion

Liang Chen *, Yuyi Yang, Zhenheng Wang, Jian Zhang, Shaowu Zhou and Lianghong Wu

School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China
* Correspondence: kentchen@hnust.edu.cn

Abstract: The performance of underwater target detection algorithms is affected by poor imaging quality in underwater environments. Due to the arithmetic power limitation of underwater devices, existing deep learning networks are unable to provide efficient detection processes with high detection accuracy. Lightweight CNN models have been actively applied for underwater environment detection, yet their lite feature fusion networks cannot provide effective fusion effects and reduce the detection accuracy. In this paper, a lightweight algorithm based on multi-scale feature fusion was proposed, with the model parameters greatly reduced, improving the target detection accuracy. The forward propagation memory overhead is reduced by using multi-scale shared convolutional kernels and pooling operations to co-construct the query matrix in the Tansformer encoding stage. Then, the feature fusion path is optimized in order to enhance the connection of multi-scale features. A multiscale feature adaptive fusion strategy is used to enhance the detection performance and reduce the dependence on the complex feature extraction network. The feature extraction network is also reparameterized to simplify the operation. Using the UPRC offshore dataset for validation, the study results have demonstrated that the statistical mAP metrics validate the detection accuracy. Compared with SSD, RetinaNet and YOLOv5-s improved by 13%, 8.6%, and 0.8%, while the number of parameters decreased by 76.09%, 89.74%, and 87.67%. In addition, compared with the YOLOv5-lite model algorithm with the same parameter volume, the mAP is improved by 3.8%, which verifies the accuracy and efficiency of the algorithm in this paper.

Keywords: underwater target detection; multi-scale fusion; transformer; YOLOv5; lightweight

Citation: Chen, L.; Yang, Y.; Wang, Z.; Zhang, J.; Zhou, S.; Wu, L. Underwater Target Detection Lightweight Algorithm Based on Multi-Scale Feature Fusion. *J. Mar. Sci. Eng.* **2023**, *11*, 320. https://doi.org/10.3390/jmse11020320

Academic Editors: Dimitrios V. Lyridis and Charis Ntakolia

Received: 29 December 2022
Revised: 13 January 2023
Accepted: 16 January 2023
Published: 2 February 2023

1. Introduction

There are plentiful application scenarios for underwater environment perception, which could provide information support for underwater pipeline inspection, marine pasture management and underwater navigation perception [1–3]. Due to its relatively low cost and the ability to maintain a high degree of accuracy within a certain range, the optical vision detection method has become an attractive research object.

In contrast to the two-stage target detection algorithm, the single-stage detection algorithm utilizes an end-to-end inference process to pinpoint the target position directly, enhancing system operational efficiency and reducing post-maintenance costs [4]. It has subsequently evolved into the mainstream studies on algorithms for visual detection in underwater devices. To adapt to the complex underwater environment, single-stage networks commonly use feature pyramid structures for feature extraction. In addition, it is possible to selectively train detectors for multi-scale feature maps, which can improve the accuracy of detecting targets of different scales.

However, underwater targets are often densely distributed and interfere with each other, which prevents the detection algorithm from effectively locating the targets, causing the detection accuracy to be compromised. Image pyramids are commonly used to extract multi-scale image features to resolve this issue, but the detection accuracy cannot be effectively enhanced because the target features cannot be connected by existing methods.

Therefore, literature [5] proposed a single-stage algorithm called MD-SSD using MobileNet to extract multi-scale feature maps for prediction. However, there is little correlation within multi-scale feature maps, which lead to the algorithm's detection accuracy improvement being unsatisfactory. RetinaNet performs feature fusion using pyramids that span multiple scales and semantic information layers, which improves its multiscale information flow capability to enhance detection accuracy [6]. The literature [7] employed the YOLOv4 network, which is commonly used in industry to detect underwater organisms. The combination with PANet [8] has provided additional bottom-up path enhancement feature fusion capability [9], which demonstrated bi-directional path fusion's effectiveness. However, because of the lack of smoothness in simple bidirectional fused features as well as the poor connection between multi-scale features, the combined network only results in a limited improvement in detection accuracy. As a further enhancement, in the literature [10], ASFF was used in conjunction with an attention mechanism to control the contribution between features at multiple scales, which is applied to underwater sonar images to improve the degree of association of multi-scale features. The algorithm improves the smoothness of feature fusion, but the complicated fusion path and the nonnegligible computational cost is challenging to adapt to the minimal storage and the computational requirements of underwater detection devices.

The above processing techniques have failed to consider the regional correlation between feature elements, which makes it challenging to effectively increase the gradient weights of target regions when processing complex underwater environment images. The literature [11] used a Transformer to improve the feature region correlation in YOLOv5 to enhance the detection accuracy of small targets, but excessively relies on complex backbone network feature extraction, which cause real-time operation tasks difficulty for detection robots. Therefore, it is complicated to find a balance between the accuracy and efficiency of the above algorithms.

To resolve that problem, our paper proposes a lightweight Transformer integrated object detection algorithm based on cross-scale feature fusion with enhanced multi-scale feature fusion. The fusion path is improved by an adaptive weighted feature fuser, and the correlation of local features of the algorithm is also improved with better detection accuracy. At the same time, the proposed model's lightweight implementation achieves a lightweight network for detecting low-quality underwater image targets while maintaining high operational efficiency.

2. Related Work

2.1. YOLOv5-lite Detection Framework

The lite network is adopted to replace the conventional standard convolution with a computationally intensive and read-simplified structure based on large convolutional detection networks. To make it more flexible to the operation and deployment of mobile embedded devices, the network model's depth and width are compressed with an acceptable performance float. Based on Ultralytics LLC's YOLOv5 optimization, the network structure of YOLOv5-lite is divided into three major parts based on its design concept: Backbone network, multi-scale feature fusion network (Neck), and predictor network (Head). Moreover, YOLOv5-lite was used in conjunction with ShuffleNetv2 to compress the backbone network [12], making it shallower in depth and more efficient to operate.

The structural framework of the YOLOv5-lite algorithm is shown in Figure 1.

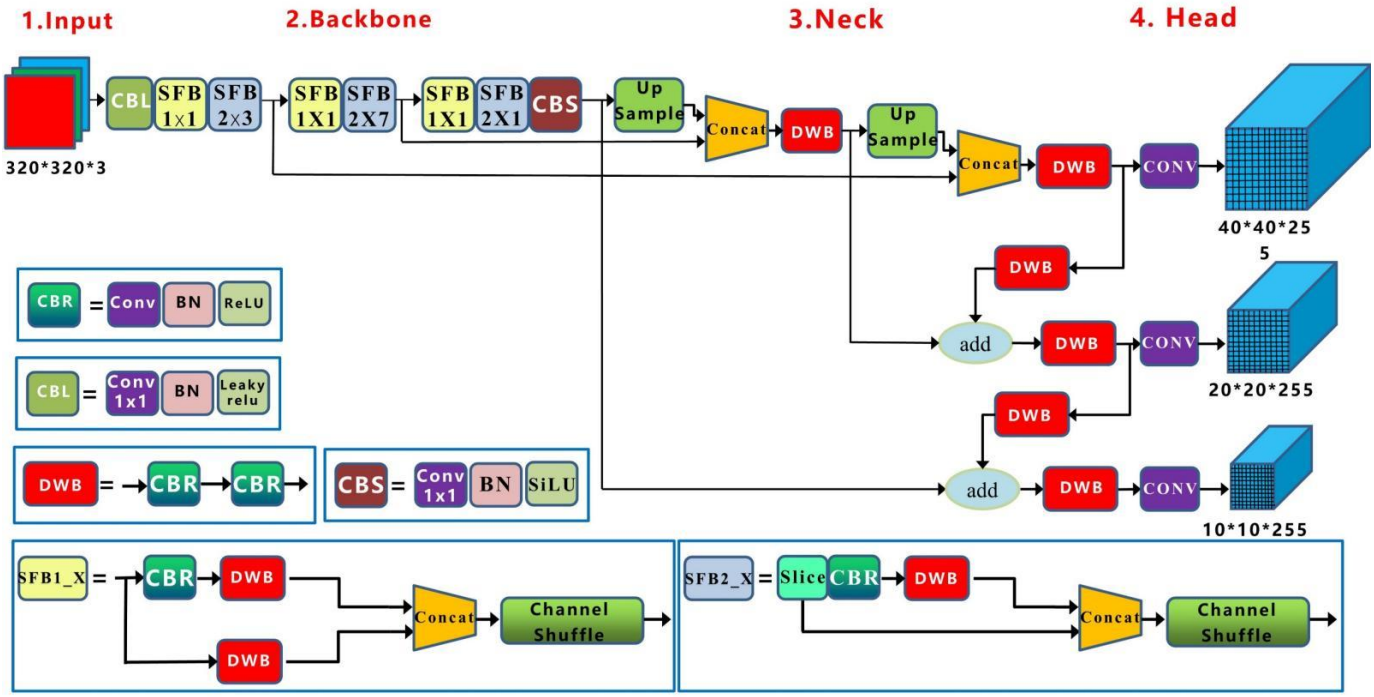

Figure 1. YOLOv5-lite overall structure.

The YOLOv5-lite network described above first feeds the online enhanced image data into the feature extraction network backbone, which performs multi-scale feature extraction on images. According to experimental results published in the literature [13], YOLOv5's frequent use of the random residual structure (CSP Bottleneck) consumes more cache memory and slows down the inference speed. In order to integrate the multiplexed module branches and enhance network parallelism, the Lite network adopted ShuffleBlock rather than the original structure module. The memory overhead is reduced, causing channel changes in the hidden layer. The multi-scale feature map was then passed through the PAN structure used for feature fusion to realize multiscale feature information interaction. Finally, point-wise convolution was used to statistically analyze the multi-resolution semantic feature maps in order to complete target location prediction and category evaluation.

To sum up, YOLOv5-lite optimized element-level operations by merging an efficient convolution structure with equal channels. The model fragmentation structure was reduced, which could achieve the goal of compressing the model depth, making the algorithm easy to deploy. However, due to the simplification of the original complex model, the algorithm has also diminished the anti-disturbance ability in complicated scenarios. The network's simple bi-directional path feature fusion structure has a low regional correlation within feature elements extracted from dense multi-scale targets. Therefore, the underwater detection accuracy of the algorithm could not be guaranteed.

2.2. Vision Transformer Models

Convolutional neural networks still play a dominant role in target detection, and ViT [14] relies on a self-attentive mechanism that works well for other tasks such as NLP. ViT adaptively divides images and adopts an architecture without convolutional modules to achieve effective results, but the uncorrelated nature between patches during initial training makes it difficult for the network to converge. DeiT [15] improves the efficiency of training Vit, enabling large networks training with better training direction. However, it cannot adapt to the storage capacity of underwater devices, which require the same model size as CNN. MobileFormer [16] uses MobileNetV2 [17] and parallel branching of ViT, which makes the model size compressed and connects local and global information through a bridging module. However, with the presence of a large amount of low-frequency noise in

the underwater environment, the detection and recognition of targets should emphasize the capture of high-frequency information at the shallow level of the model and the union of low-frequency information at the high level of the model. In this paper, we combine with Transformer in several different stages of the CNN deep layer and enhance the feature fusion capability of neck structure in lite type network to improve the underwater target detection performance.

3. YOLOv5-lite Detection Framework with Multi-Scale Feature Fusion

Since single-stage target detection algorithms that rely solely on complex backbone networks are unable to fully exploit multi-scale features, in this paper, we optimize the inference structure of the backbone network to ensure stable operation and rapid detection of underwater intelligent devices. Moreover, considering that the multiscale information in the feature pyramid implies spatial distance association information, this paper improves the feature fuser of YOLOv5-lite in combination with Transformer to enhance the multiscale feature association of the Neck structure. The feature fusion capability is enhanced by improving the adaptive fusion path, which ensures the robustness of the algorithm in detecting underwater targets of various scales.

3.1. Backbone Network Design

To reduce storage costs, YOLOv5-lite compresses the network and downscales using a two-branch structure. To ensure stable training and potential generalization, the operation requires data distribution adjustment of the features using batch size regularization (BN) after the convolution operation. While BN and data batch size are directly correlated, as data batch size scales get smaller, regularization errors increase with network depth. Consequently, the distribution of the data at inference deviates from the distribution of the data at training, which has a negative impact on test performance. In order to improve the batch normalization after 3×3 convolution in the ShuffleBlock residual structure and improve the architectural anti-perturbation performance, batch-free normalization (BFN), which has been shown to be effective in the literature [18], is applied in this paper. The layer normalization (LN) can be expressed as Equation (1).

$$\hat{x_j} = LN(x_j) = \frac{x_j - \mu}{\sqrt{\sigma^2 + \varepsilon}}, j = 1, 2, \ldots, d \tag{1}$$

where, $\mu = \frac{1}{d}\sum_{i=1}^{d} x_j$ and $\sigma^2 = \frac{1}{d}\sum_{i=1}^{d} (x_j - \mu)^2$ represent the mean and variance of the data samples, respectively. LN can be generalized to group normalization (GN) by dividing neurons into groups, making it possible to use it for small batch training. Performance degradation could be minimized by ensuring that the inputs at higher levels have almost identical distributions.

The proposed network structure reduces data distribution shifts brought about by small batches of training data and mitigates information loss during downsampling. In order to lessen the loss of inference efficiency caused by multi-branch structure, this paper optimizes ShuffleBlock based on the concept of structural reparameterization that has been suggested in the literature [19]. To increase the network's detection accuracy during training, the training and inference processes of forward inference are decoupling. This decoupling process is shown in Figure 2. To increase network inference speed and reduce hardware throughput data overhead while maintaining the same network performance, the extracted structure parameters are fused during the inference process.

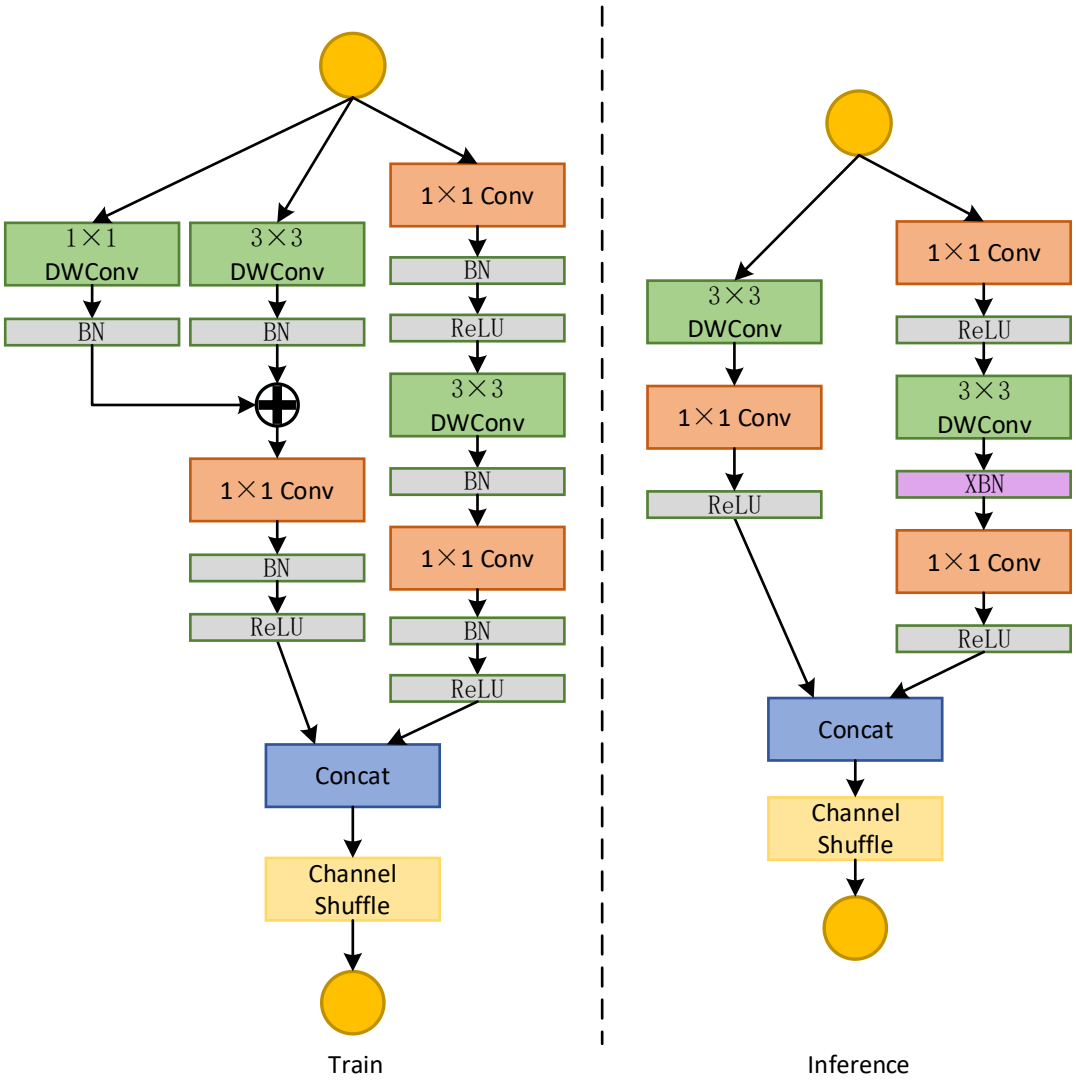

Figure 2. Training and inference module.

When the model is trained, the convolution and batch normalization operations are represented as Equation (2):

$$O = \gamma_j \frac{\sum_{i=1}^{C} w_i^{(j)} \times x_i + b_j - \mu_j}{\sqrt{\sigma_j{}^2 - \varepsilon}} + \beta_j \tag{2}$$

where $w_i^{(j)}$ is the convolution parameter of the jth batch of data, b is the network parameter bias, γ is the normalized scale factor, and β is the bias term. After the input features are convolved with 3×3 and 1×1 convolutions and the corresponding batch normalization operations, the model nonlinearity capability is enhanced using ReLU activation. The two branches are then stacked with channels rather than using linear computing to increase the network operation rate, and finally shuffle channels are used to enhance the exchange of feature information.

In addition, in order to reduce the number of parameters, regularization manipulation and convolution are fused at the time of modular inference. The convolution and batch normalization manipulation can be broken down into weight terms and bias terms using the information gathered from the above equation. These terms are then used as new

convolution weights and bias data for sharing network parameters. The model inference structure that results from this can be stated as Equations (3) and (4):

$$\hat{w}_i = \frac{\gamma_j}{\sqrt{\sigma_j{}^2 - \varepsilon}} \sum_{i=1}^{C} w_i^{(j)} \tag{3}$$

$$\hat{b}_j = \frac{\gamma_j}{\sqrt{\sigma_j{}^2 - \varepsilon}} (b_j - \mu_j) + \beta_j \tag{4}$$

where $\hat{w}_i$ denotes the new convolutional kernel weights utilized for the parameters, and $\hat{b}_j$ indicates its parameter bias. The re-parameter utilization mechanism could save nearly 36% of the number of forward propagation parameters for each residual shuffle unit, improving the network operation efficiency.

3.2. Transformer Modules Design

The ViT processes images as text to improve regional correlation, by encoding feature local blocks as feature vectors as input to the Transformer module as a way to obtain the degree of autocorrelation between feature local blocks. To calculate the vector similarity based on self-attentiveness, the standard Transformer needs three projection matrices, W_q, W_k and W_v, respectively. However, the large feature map will cause a threefold increase in the Transformer's encoding space, making it difficult for the capacity and computing power of small hardware devices to meet their requirements of accuracy and efficiency. In this paper, we obtain the linkage weight between local blocks of images by self-attentive operation based on sliding window. The self-attention formula in each window can be expressed as Equations (5) and (6):

$$y_i = \sum_{j \in N(i)} \alpha_{i \to j} W_v x_j \tag{5}$$

$$\alpha_{i \to j} = \frac{e^{(W_q x_i)^T W_k x_j}}{\sum_{z \in N(i)} e^{(W_q x_i)^T W_k x_z}} \tag{6}$$

where the similarity $\alpha_{i \to j}$ is calculated from the query matrix W_q, key matrix W_k. The scalar between [0,1] is obtained as the feature local block vector linkage weights. Thereafter, Softmax is used to normalize $\alpha_{i \to j}$, controlling the contribution of this vector in the spatial location.

To simplify the computation, Transformer uses multi-headed self-attentiveness to be responsible for different coding vectors separately. However, the above mechanism only establishes spatial association of feature local blocks through a single layer encoder, ignoring the effect of detailed local information such as image texture on target detection, which could bring significant performance degradation when the number of model parameters is reduced to small networks.

Multi-scale features are beneficial for detection or semantic segmentation. Large neural networks use multi-scale design both in the feature layer hierarchy and single-stage feature extraction. The multi-scale design can help the network to extract structural features from different perspectives, hence we establish the convolution of multi-scale shared kernel parameters to extract the query matrix W_q of feature local block encoding vectors. The AtrousFormer module described in literature [20], which uses pooling operations combined with convolution operations to build local perceptual fields, allows the query vector to be built aggregating detailed information of the local feature space.

The multi-scale self-attentive query structure is shown in Figure 3.

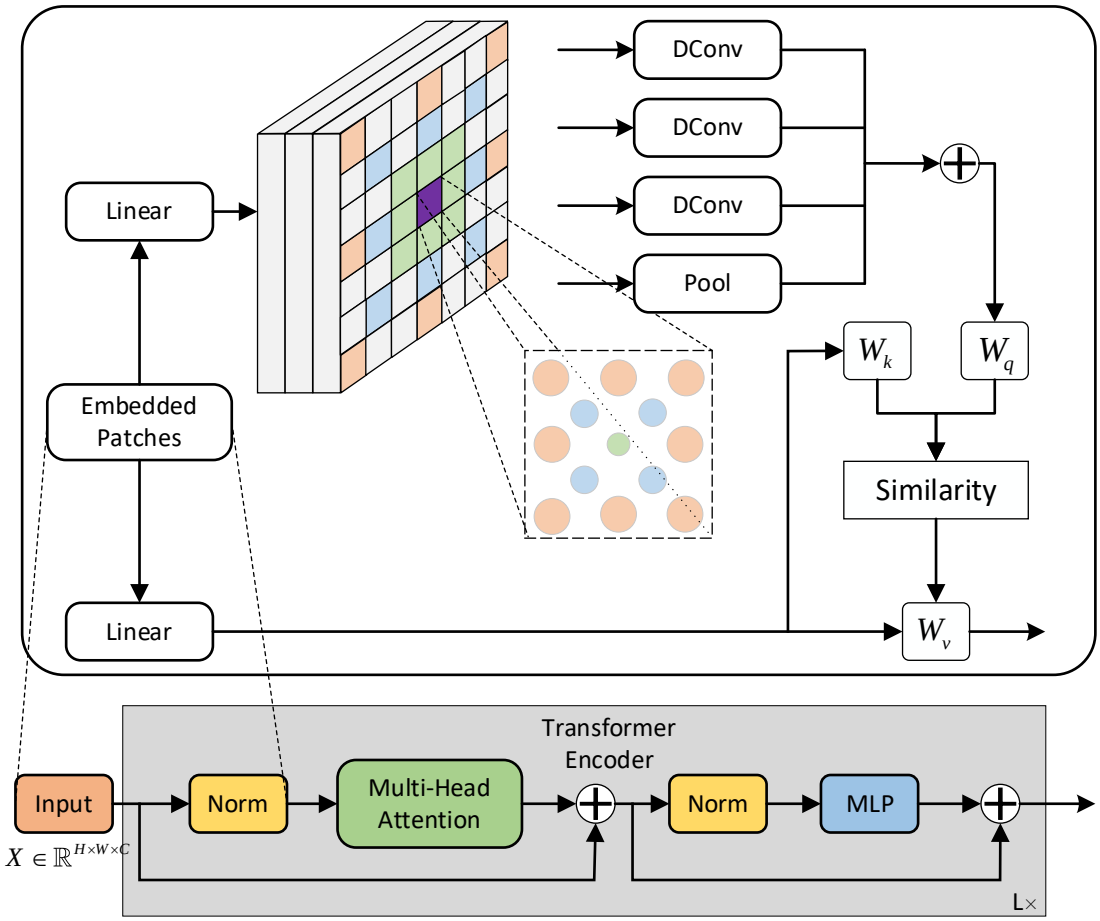

Figure 3. Multiscale query Transformer.

The given input features $X \in \mathbb{R}^{H\times W\times C}$ are encoded using a multilayer perceptron to map them by dimensional channels, after which the features are spanned by dimension to obtain N ($N = H \times W$) graph feature vectors. Additionally, the number of channels is divided into channel dimensions by a custom number of multi-headed self-attention, and finally the encoded feature vector $X_{emb} \in \mathbb{R}^{N\times num\_head\times \frac{C}{num\_head}}$ is obtained.

Unlike classification, spatial constraints need to be considered when performing underwater detection tasks. Thus, the spatial template relationship unique is introduced to convolution instead of the position encoding of the standard transformer. By sharing convolution kernel parameters applied to multi-scale convolution kernels, the amount of spatial location information of features could be increased while reducing the number of parameters and memory overhead.

In addition, there is morphological occlusion of the target, which could easily lead to feature confusion as the target features are disturbed by underwater environmental noise. Building the query matrix by convolution kernels with learnable parameters may only lead to the problem of feature gradient disappearance. Therefore, in this paper, the ability of query features to resist spatial displacement bias is boosted by adding feature pooling to the matrix building process. Equation (7) is an expression of the computational process.

$$F_q = \sum_{r\in\mathbb{R}} SiLU(DConv_r(F_e, W_q^{k=3}, g = d)) + avgpool(F_e) \quad (7)$$

where F_e is the feature map deformed by encoding, $SiLU(\cdot)$ is the activation function, $DConv_r(\cdot)$ is the expansion convolution operation with r as the expansion rate, $W_q^{k=3} \in \mathbb{R}^{k^2\times d\times d/g}$ is the kernel weight to obtain multi-scale queries of feature local block vectors

by setting different expansion rates, and avg$pool(\cdot)$ is the average pooling corresponding to the kernel size to enhance the generalization ability of multi-headed self-attentiveness by aggregating vector data features.

Shallow features have more spatial information and larger feature map resolution. Since the direct application of Transformer is unable to balance the forward propagation parameter memory overhead, shallow features are calculated in our paper by implementing a convolutional self-attentive module (CSAFormer), which could directly predict the similarity between the query and key value matrix as the weight of the attention values. The structure is shown in Figure 4.

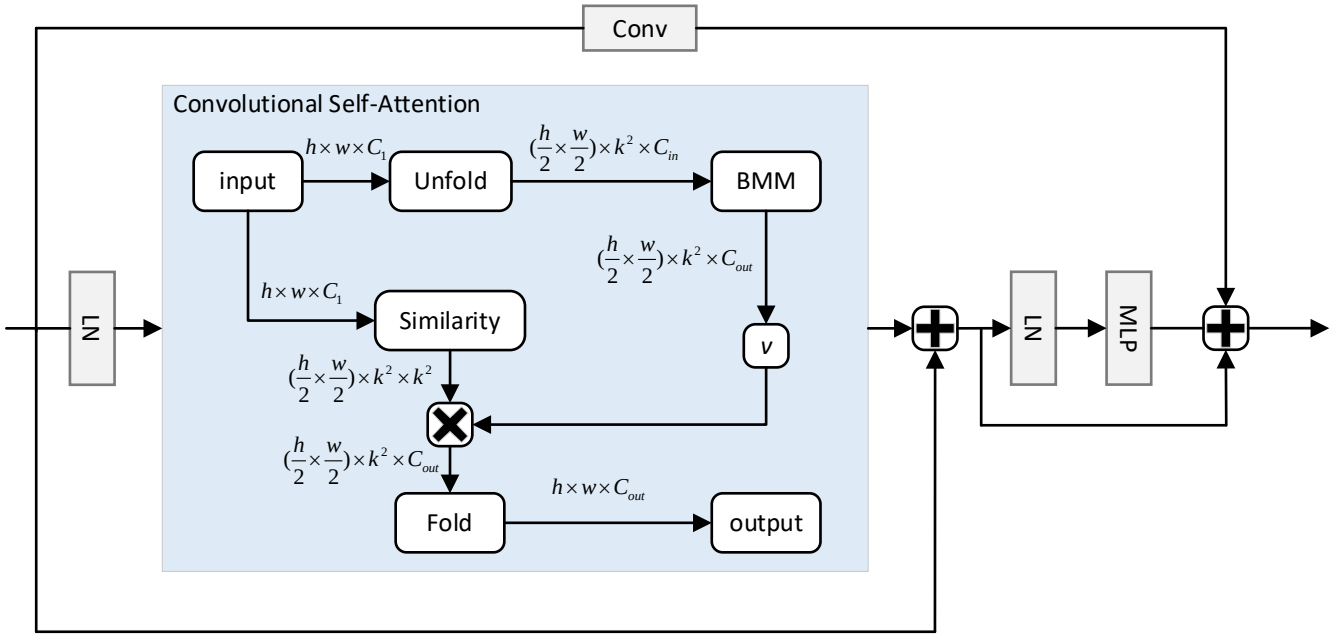

Figure 4. Convolutional self-attention module.

In order to directly predict the similarity matrix, the 3 × 3 sliding kernel is implemented inside the convolutional self-attentive module and encoded with a fully connected layer mapping, which is proved to have better performance when the sliding kernel is small. The convolutional multiplication operation is represented by the Batch matrix multiplication (BMM) as the batch matrix multiplication carried out by the divided local feature blocks. Each kernel parameter has a corresponding similarity weight matrix consisting of k^2 elements. Finally, the summation operation of the k^2 different self-attention weights associated with the input values is performed. The entire structure includes learnable filters as well as internal self-attentive dynamic kernels, which improve the spatial adaptability of shallow information. To implement the CSAFormer module, internal residual connections are added in this paper.

3.3. Multi-Scale Neck Structure Improvement

For low quality underwater images, shallow backbone networks are unable to extract important features. It is a common strategy to increase the complexity of the backbone network, which could introduce too many parameters, making the deployment of underwater devices difficult. The literature [21] claimed that detection algorithms would also achieve better results by relying mainly on multi-scale feature fuser. Therefore, in this paper, we extract the shallow backbone network of YOLOv5-lite, and potential features for fusion in the Neck module could be utilized. By making multi-scale features process high-level linguistic and low-level spatial information with the same priority at an early stage, the network's detection performance should be improved. An adaptive path weight

in multi-path fusion is presented to reduce single fusion nodes and improve the fusion capability of the Neck structure. Additionally, we combine the Generalized FPN applied to the internal block connection of YOLOv5-lite in order to address the gradient growth issue that may arise as a result of the enhanced Neck structure emphasizing only feature fusion. The improved Neck structure fusion unit is shown in Figure 5.

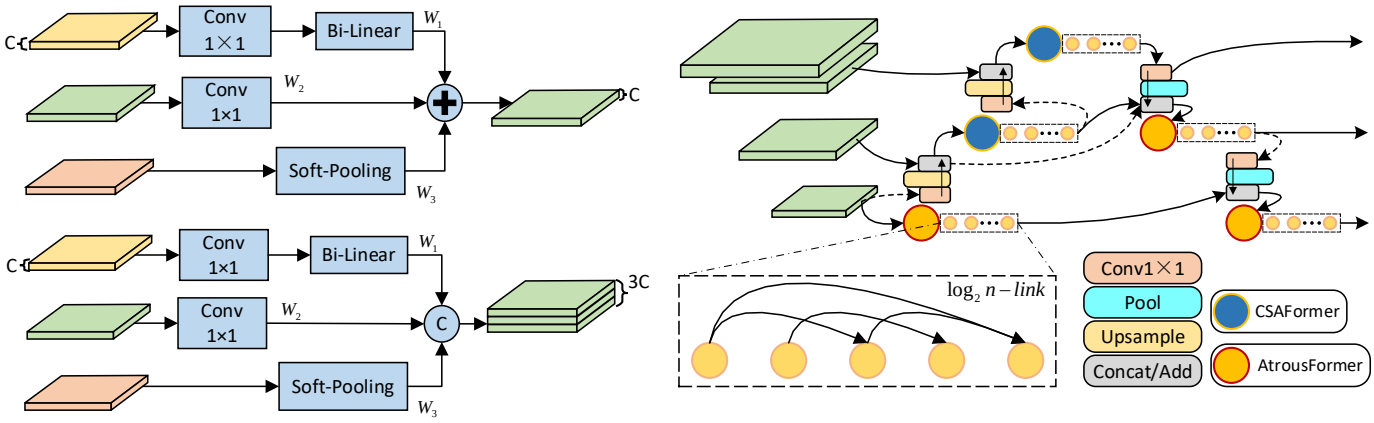

Figure 5. Adaptive weight fusion module and multi-scale fusion Neck structure.

As shown in the right part of Figure 5, the enhanced Neck structure $\log_2 n - link$ utilizes 3 layers of feature layers at the same scale as input. The feature maps are connected between the same scales using jumping layers to reduce the gradient disappearance problem during the back-propagation of the complex Neck structure. This connection allows the lth layer to receive at most $\log_2 l + 1$ layers of feature information. Compared with dense-link, it reduces the computational cost while maintaining the performance.

The cross-scale feature maps are fused in the Neck structure after up-sampling and down-sampling transformations, as shown in the left part in Figure 5. The up-sampling uses bilinear interpolation to ensure the information association between elements, and the down-sampling operation in this paper uses the Soft-Pool proposed by Stergiou A (2021) to enhance the weight ratio of dominant information in down-sampling [22], which can be expressed as Equation (8).

$$\widetilde{p} = \sum_{i \in R} \frac{e^{p_i}}{\sum\limits_{j \in R} e^{p_j}} \times p_i \tag{8}$$

where p is a pixel in the feature map and j is an index within the pooling kernel. The risk of losing information by down sampling in the high-dimensional feature space is reduced by retaining the dominant weights at high activation value locations. However, this approach has significant computational cost, which is expressed in this paper by simplifying the computation as Equations (9) and (10).

$$\widetilde{p} = \sum_{i \in R} \frac{HardSwish(p_i) + 1}{\varepsilon + \sum\limits_{j \in R} (HardSwish(p_j) + 1)} \times p_i \tag{9}$$

$$HardSwish(x) = x \frac{\mathrm{ReLU6}(x+3)}{6} \tag{10}$$

where $HardSwish(\cdot)$ is the ReLU activation function improvement function with limited upper bound to prevent the noise gradient from affecting the pooling result and enhance the advantage of high-dimensional features in the deeper layers of the network, which is a very small value protection calculation. This method saves 0.47 times the computation time while ensuring performance and is more appropriate for lite type networks.

After that, cross-layer fusion is performed by adjusting feature scales and using learnable parameters to balance the importance between multi-scale features, which allows

the network to maintain a sufficient amount of information exchange even when the network scales over large distances, improving the detection accuracy of the network detector.

4. Experimental Verification and Analysis

4.1. Experimental Data Set

In this paper, optical vision target detection is applied to offshore underwater robots. The multi-scale feature fuser's fusion ability is enhanced with the model keeping lightweight, enabling the shallow lightweight network to detect targets effectively in challenging underwater environments. To train the model and validate generalization, this paper uses 2900 actual underwater images from the underwater robot professional contest (URPC) (https://www.curpc.com.cn/, accessed on 1 September 2020). Target types include holothurian, echinus, scallops and starfish. These include targets that are morphologically overlapping, densely distributed, and various sizes and scales. Some image examples are shown in Figure 6.

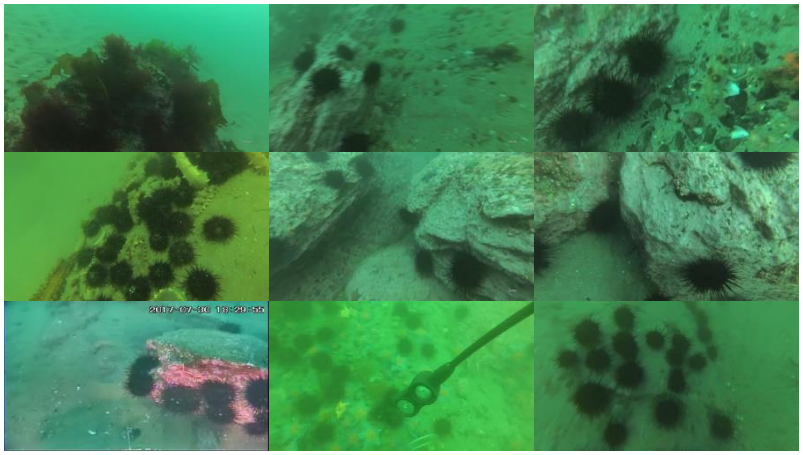

Figure 6. Example of a dataset image.

4.2. Experimental Parameter Setting

The training environment in this paper is on an Ubuntu system, and the training device is a graphics card RTX2080-11G, RAM size of 16 G, using pytorch framework. The ImageNet pre-trained network model is transferred, the batch size is set to 16, the image size is 416 × 416 and iterative training is performed for 500 epochs. The dataset was divided in 8:2 for training and testing. The number of class instances, real bounding boxes visualization, real frame label centroids, and aspect ratio scatter plots of the dataset are shown in Figure 7.

The first image of Figure 7 demonstrated that the experimental dataset suffers from instance category imbalance. There were 2797 instances of holothurian, 10,995 instances of echinus, 1160 instances of scallops, and 3269 instances of starfish. The minimum category is less than 1/10 of the maximum category. Therefore, in our paper, the category loss function is replaced with Focal-loss and the category loss is weighted for the dataset with the weighting parameters of 3.26, 0.83, 7.85, and 2.79. The second column of images in Figure 7 shows that the labeled boxes are relatively balanced in length and width. The first image in row 2 shows that the center points are also distributed mainly based on the image center. Thus, take the default anchor distribution.

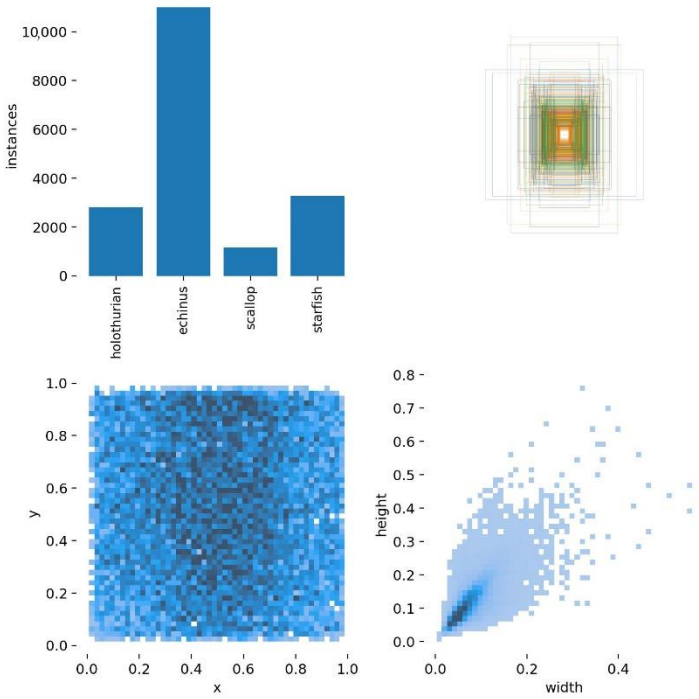

Figure 7. Data set label analysis.

4.3. Ablation Experiments

To analyze the effect of the improved multiscale feature fuser, Neck adaptive fusion, Cross Stage Partial replacement to Transformer is improved in two volumes of YOLOv5-lite network, respectively. YOLOv5s-lite and YOLOv5g-lite are employed by Rep-Shuffle and Rep-Convolution module as the backbone network convolution. The number of model parameters and computation are observed in Table 1, while the intermediate modules are ablated for the evaluation index of mean Average Precision (mAP), which measures the accuracy of the model.

Table 1. Comparison of ablation experiments.

Generalized-FPN	CSA Former	Arous Former	Soft Pool	mAP (%)	Total Parameters of the Model	Total Model Calculation (GFLOPS)
				73.8	5,574,845	16.2
✓				74.1	5,640,642	16.3
	✓			74.2	6,200,397	17.7
		✓		76.3	11,561,597	17.8
✓	✓	✓		77.1	5,773,330	16.9
✓	✓	✓	✓	77.5	5,740,690	19.7

By comparing the mAP values of the network with the above improved modules and the original YOLOv5-lite, it is evident that the mAP of the improved Neck with the Generalized-FPN feature fusion method, the replacement of the original convolution operation with Arous Former and CSA Former, and the improvement of the downsampling method with Soft Pool in detecting underwater targets increased by 0.3%,0.4%,2.5%, and 3.7% respectively. The above results show that the proposed algorithm in this paper has higher accuracy and recall rate in underwater target detection. On the other hand, if the fusion structure in the neck is replaced directly by CSAFormer, the number of model parameters will increase by 559755; if the fusion structure is simply replaced by ArousFormer, the number of model parameters will increase by 5920955, while our proposed network distinguishes deep features from shallow features to improve the detection accuracy while balancing the parameters and computational effort. In addition, the backbone network

reparameterization method implemented in this paper can save 9192 model parameters and reduce the computational effort by 0.1 GFLOPs while maintaining the mAP value, which makes the model easier to reason about.

Feature pooling can be used in convolutional neural networks to aggregate feature representations, reduce computational requirements and allow for the creation of larger and deeper architectures. As shown in Figure 8, the results are compared by comparing images undergoing common pooling operations with the soft pool optimized in this paper.

Figure 8. Results of different pooling methods.

The output results for the 9 × 9 size pooling kernel for the 255 × 255 image are shown in the figure. We observe that the image looks patchy after the maximum pooling operation, emphasizing the main features but lacking in specifics. Edge contours are missing from the average pooling, which filters out the peak information. When aggregating the image data in our proposed algorithm and Soft Pool method, the focused information is afforded with more weight, which is advantageous for the transfer of feature information during down sampling. However, during inference, we observe that the feature values in Neck are small and sparse, so the exponential operation is discarded and has a higher inference speed than the SoftPool method.

4.4. Intermediate Characteristics Analysis

Transformer enhances the detection accuracy of the model in complex environments by finding the intrinsic associations between feature vectors. In this paper, we obtain the potential connections between scales by directly integrating Transformer in the multi-scale feature fuser. It enables the multi-scale features to be fully fused and boosts the key feature weights, so that the algorithm structure can be more robust to low quality targets in complex scenes. For example, the recall rate is higher, while accuracy is maintained in scenes with occlusion between targets and dense target distribution.

By visualizing the intermediate feature layer after Transformer in Neck, generating attentional heat map can visualize the change of network gradient values (Selvaraju R R, 2020). The position of the attention of the intermediate layer of the network during the fusion of the target features was analyzed and visualized, as shown in Figure 9.

Intermediate layer feature maps exist in high dimensional feature space with more semantic information between dimensions [23,24]. Columns 2 to 6 of Figure 9 show the different semantic feature maps extracted from the middle layer feature maps. The feature map visualization shows that our proposed algorithm has higher gradient values at the target location in the target dense scene among stones and less activation of the same class gradient values in the background. In mud-obscured scenes, this algorithm can effectively correlate the gradient activation between the same class, but YOLOv5s-lite is unable to do so, which leads to low detection confidence. In complex terrain, the algorithm in this paper can effectively highlight dense target activations in the terrain, while YOLOv5s-lite activations cannot obtain effective target locations. The algorithm in this paper also effectively distinguishes water plants from terrain on the background semantic dimensional map, leading to a higher recall rate for the algorithm.

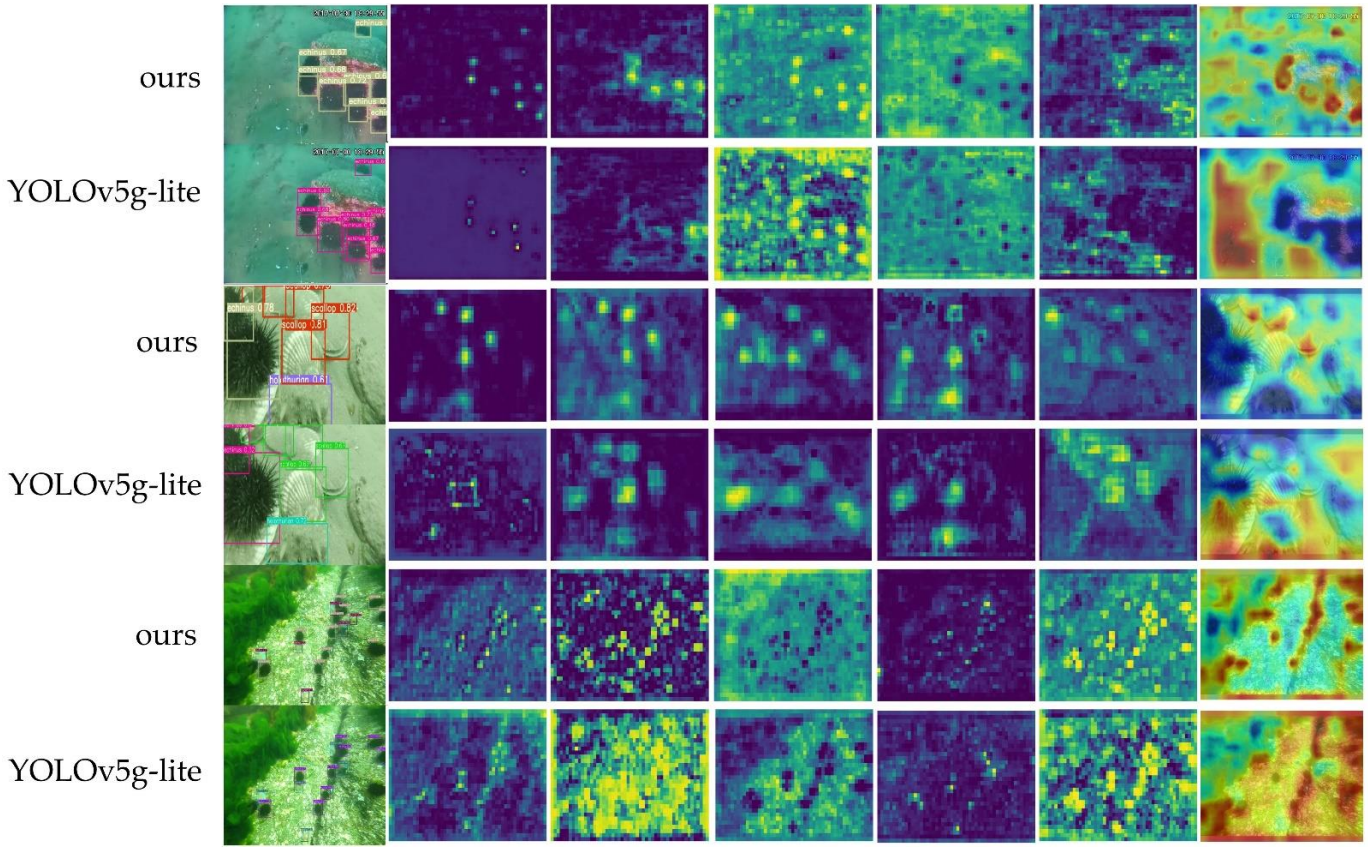

Figure 9. Detection effect visualization, feature visualization and attention visualization results.

It can be seen from the attentional heat map in column 7 of Figure 9 that the algorithm in this paper can effectively focus on the target location region in the dense scene of stones. In the mud-obscured scenes, the attention of this paper's algorithm for scallop class detection is in the scallop salient region, and there is a gradient change connected domain between the same kind, which proves that this paper's algorithm effectively associates the inter-target connection. YOLOv5s-lite is affected by mud and sand and primarily focuses on regional details as the detection basis. In complex terrain, our proposed algorithm focuses on separating the foreground and background and focuses more on the target region than YOLOv5s-lite, resulting in a more reliable detection of the target

4.5. Algorithm Comparison Experiment

To validate the performance advantage of the algorithm in this paper, the existing popular single-stage target detection neural network is presented as a reference for the experiment. The mAP index of other networks is tested with the same hardware platform and the same data set. The experimental results are summarized in Table 2.

Comparing the metric results of different algorithms in the table, compared to the SSD algorithm [25], which uses predictors directly for prediction based on multi-scale feature maps, the algorithm in this paper shows a 13% improvement in detection results and a 76.09% reduction in the number of parameters. Compared to the RetinaNet algorithm [26], a positive and negative sample optimization network with integrated feature pyramids, the detection result of this paper's algorithm is improved by 8.6 and the parameter consideration is reduced by 89.74%. Compared to YOLOv5, the computational and parameter volumes are reduced by 81.98% and 87.67%, while the accuracy is even slightly higher than YOLOv5s mAP when YOLOv5g-lite is used, with an improvement of 0.8% and 0.3%. On a YOLOv5-lite network of the same volume, the mAP improvement of 3.7% on YOLOv5g-lite and 3.3% on YOLOv5s-lite using this paper's algorithm indicates that the improvement of

this paper's algorithm on the Neck is more important. The experimental data show that the algorithm in this paper balances the number of parameters and performance accuracy, and the algorithm is more efficient for underwater target detection and more suitable for underwater device deployment.

Table 2. Comparison of detection performance and number of parameters of different algorithms.

Method	Backbone	Size	mAP (%)	Parameters
SSD	VGG	300 × 300	64.5	24,013,232
RetinaNet	ResNet50	600 × 600	68.9	55,938,472
YOLOv4	DarkNet-CSP	416 × 416	78.0	64,363,101
YOLOv5s	DarkNet	416 × 416	76.7	7,071,633
YOLOv5l	DarkNet	416 × 416	77.2	46,563,790
YOLOv5s-lite	DarkNet	416 × 416	70.1	1,649,853
YOLOv5s-lite (our)	DarkNet	416 × 416	73.4	1,432,930
YOLOv5g-lite	DarkNet	416 × 416	73.8	5,574,845
YOLOv5g-lite (our)	DarkNet	416 × 416	77.5	5,740,690

The UAV real-time detection algorithm is widely used in the application deployment of unmanned intelligent devices [11], and the performance of this paper's algorithm on real-time detection is reflected by comparing it with this detection algorithm. In addition, the corresponding module implementation is replaced with the standard Transformer for real-time experimental comparison, reflecting the detection efficiency of this algorithm in terms of module improvement. 260 underwater images are selected for testing. The average test time and frame rate per second (fps) are calculated separately. The results are shown in Table 3.

Table 3. Comparison of the real-time performance.

Models	Average Test Time (ms)	fps
YOLOv5-lite-s-tran (*)	32.0	30.98
TPH-YOLO	24.5	40.75
YOLOv5-lite-g(ours)	20.7	48.28
YOLOv5-lite-s-tran	18.6	53.58
YOLOv5-lite-s(ours)	18.4	54.39

The results show that the algorithm in this paper improves the single image processing speed by 15.5% and the detection frame rate by 18.47% over TPH-YOLO on Average Test Time. It is also more efficient than YOLOv5-lite-s, which replaces some convolutional layers with standard Transformer. In the individual module (*) replacement test, the algorithm in this paper improves the detection performance while improving the detection rate by 42.5% and the detection frame rate by 75.5%.

Regarding the ability to identify each category compared with the baseline algorithm, i.e., YOLOv5, YOLOv5s-lite, YOLOv5g-lite and the performance of this paper's algorithm in the test images. The results in Table 4 show that the algorithm in this paper improves the detection accuracy for different categories of targets, so the model can improve the detection performance in each target type.

Table 4. Comparison of target type identification.

Models	AP				mAP
	Holothurian	Echinus	Scallops	Starfish	
YOLOv5-lite-s	58.0	88.5	55.3	78.6	70. 1
YOLOv5-lite-s(ours)	64.0	91.5	57.2	80.7	73.4
YOLOv5-lite-g	65.2	90.2	60.0	79.8	73. 8
YOLOv5-lite-g(ours)	69.9	91.4	64.6	84.1	77.5

The visualization of the detection results of the algorithm in this paper is depicted as Figure 10.

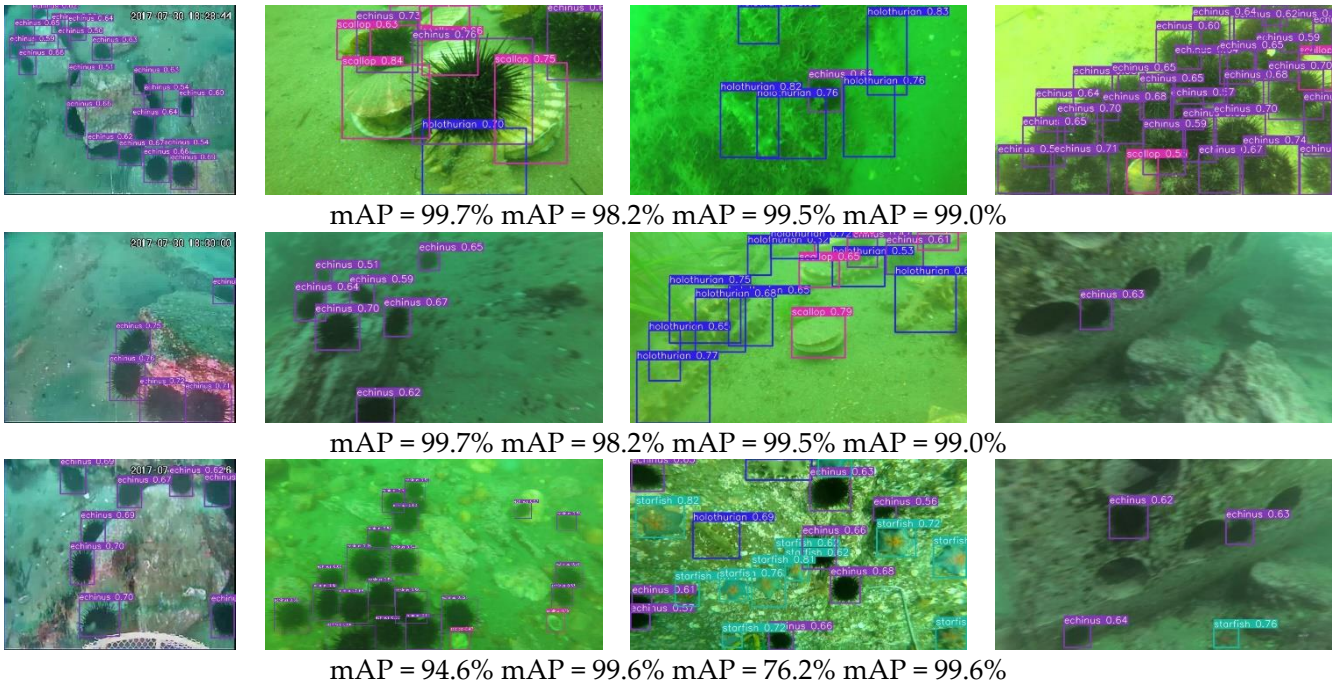

Figure 10. Visualization of the detection results of the algorithms in this paper.

By applying our proposed algorithm to different underwater environments for verification, it can be seen from the visualization result graph that the underwater targets have a different scale size distribution. By improving the multi-scale feature fusion in this paper, targets of different sizes can be effectively detected. In the case of unclear underwater fogging, this algorithm can identify the sea cucumber targets hidden in the water plants. The high recall rate is also maintained in dense scenes.

5. Discussion

Underwater environmental sensing devices require lightweight network models, but lightweight designs tend to focus on the backbone, which hinders the information exchange of the extracted features [27–29]. Low frequency noise that interferes with target detection is difficult to eliminate due to the complex optical properties of the water body. The absorption and scattering of light underwater leads to blurred targets and low contrast, compromising the target detection process. Enhancing the backbone as in atmospheric environments is not appropriate, and detection should incorporate with underwater multi-scale features. Through experiments we verified the effectiveness of enhancing feature fusion behind a light backbone, outperforming large network models in terms of detection performance. It is obvious that the effective mechanism is the improvement of the network normalization in backbone, which makes the cumulative error at the time of training is mitigated. The shallow high-resolution feature maps in the network are effectively fused with the deep low-resolution feature maps. The deep feature map is filtered several times to eliminate the high frequency information [30], and the underwater interference that is considered to be mainly present is smoothed, which can guide the shallow feature map to detect and locate the target, while the Transformer structure module implemented separately for different scale feature maps in this paper can strengthen the connection of spatial features and improve the detection accuracy, which can be seen from the ablation experiment. In addition, the corresponding improvements on the large model obtain slightly higher performance gains than the small model, probably mainly because the deep feature maps on the large model have more dimensions and have a better information base

when performing fusion. This allows Transformer to perform spatial attention modeling with longer vector lengths, representing a stronger representation and ultimately resulting in better model performance. However, the lightweight network still has difficulty in convergence during training. In this paper, we set the downsampling multiplier to 8, 4, and 0 to balance the corresponding computational cost. It is a challenge to balance and adjust the hyperparameters during training for our further research.

6. Conclusions

For the optical image target detection and localization needs of underwater inspection robots, we propose a lightweight target detection model with improved multi-scale feature fusion. The inference process is simplified by using the re-parameterization mechanism and optimizing the data normalization method to improve the operational efficiency of the model feature extraction. Then the model parameters are optimized and the weight adaptive path is designed to reconstruct the Neck, which enhances the information exchange of multi-scale features. Combined with the improved Transformer, the local feature correlation is enhanced to compensate for the scale information in the deep feature space. The underwater environment perception capability of the algorithm is improved at the multi-scale level. Eventually, the detection efficiency is improved while maintaining the operational efficiency of the algorithm. Presently, the proposed network improves the detection algorithm for the Neck structure, but does not fuse different scale feature layers in terms of feature correlation between multi-scale feature extraction layers, which will be the direction of future optimization.

Author Contributions: Conceptualization, L.C. and Y.Y.; Methodology, L.C.; Software, Y.Y.; Validation, Y.Y. and L.C.; Formal Analysis, J.Z.; Investigation, J.Z.; Resources, J.Z.; Data Curation, Z.W.; Writing—Original Draft Preparation, Y.Y.; Writing—Review and Editing, L.C. and Z.W; Visualization, Y.Y.; Supervision, S.Z. and L.W; Project Administration, L.W.; Funding acquisition, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 62271199 and Science and Technology Talents Sponsorship Program by Hunan Association for Science and Technology, China, grant number 2022TJ-Q03.

Institutional Review Board Statement: Not available.

Informed Consent Statement: Not available.

Data Availability Statement: Not available.

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

References

1. Al Muksit, A.; Hasan, F.; Emon, M.F.H.B.; Haque, M.R.; Anwary, A.R.; Shatabda, S. YOLO-Fish: A robust fish detection model to detect fish in realistic underwater environment. *Ecol. Inform.* **2022**, *72*, 101847. [CrossRef]
2. Zhou, J.; Xu, T.; Guo, W.; Zhao, W.; Cai, L. Underwater occlusion object recognition with fusion of significant environmental features. *J. Electron. Imaging* **2022**, *31*, 023016. [CrossRef]
3. Ntakolia, C.; Moustakidis, S.; Siouras, A. Autonomous path planning with obstacle avoidance for smart assistive systems. *Expert Syst. Appl.* **2023**, *213*, 119049. [CrossRef]
4. Sun, Z.; Lv, Y. Underwater attached organisms intelligent detection based on an enhanced YOLO. In Proceedings of the 2022 IEEE International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 25–27 February 2022; pp. 1118–1122. [CrossRef]
5. Yao, Y.; Qiu, Z.; Zhong, M. Application of improved MobileNet-SSD on underwater sea cucumber detection robot. In Proceedings of the 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chengdu, China, 20–22 December 2019; Volume 1, pp. 402–407. [CrossRef]
6. Wei, Q.; Chen, W. Underwater Object Detection of an UVMS Based on WGAN. In Proceedings of the 2021 China Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 702–707. [CrossRef]
7. Hao, W.; Xiao, N. Research on Underwater Object Detection Based on Improved YOLOv4. In Proceedings of the 2021 8th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), Beijing, China, 10–12 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 166–171.

8. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018. [CrossRef]
9. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
10. Fan, X.; Lu, L.; Shi, P.; Zhang, X. A novel sonar target detection and classification algorithm. *Multimed. Tools Appl.* **2022**, *81*, 10091–10106. [CrossRef]
11. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788.
12. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
13. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural net-work for mobile devices. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
14. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 $\times$ 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
15. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distil-lation through attention. In Proceedings of the International Conference on Machine Learning, Online, 13–15 December 2021; pp. 10347–10357.
16. Chen, Y.; Dai, X.; Chen, D.; Liu, M.; Dong, X.; Yuan, L.; Liu, Z. Mobileformer: Bridging mobilenet and transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; Volume 2, p. 6.
17. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.
18. Huang, L.; Zhou, Y.; Wang, T.; Luo, J.; Liu, X. Delving into the Estimation Shift of Batch Normalization in a Network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 753–762. [CrossRef]
19. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. Repvgg: Making vgg-style convnets great again. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13733–13742.
20. Yang, C.; Wang, Y.; Zhang, J.; Zhang, H.; Wei, Z.; Lin, Z.; Yuille, A. Lite Vision Transformer with Enhanced Self-Attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11988–11998. [CrossRef]
21. Jiang, Y.; Tan, Z.; Wang, J.; Sun, X.; Lin, M.; Li, H. GiraffeDet: A Heavy-Neck Paradigm for Object Detection. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
22. Stergiou, A.; Poppe, R.; Kalliatakis, G. Refining activation downsampling with SoftPool. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 10357–10366.
23. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Locali-zation. *Int. J. Comput. Vis.* **2020**, *128*, 336–359. [CrossRef]
24. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2921–2929.
25. 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; Springer: Cham, Switzerland, 2016; pp. 21–37.
26. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
27. Zhou, T.; Si, J.; Wang, L.; Xu, C.; Yu, X. Automatic Detection of Underwater Small Targets Using Forward-Looking Sonar Images. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 4207912. [CrossRef]
28. Pang, J.; Liu, W.; Liu, B.; Tao, D.; Zhang, K.; Lu, X. Interference Distillation for Underwater Fish Recognition. In Proceedings of the Asian Conference on Pattern Recognition, Macau SAR, China, 4–8 December 2022; pp. 62–74. [CrossRef]
29. Chen, L.; Zhou, F.; Wang, S.; Dong, J.; Li, N.; Ma, H.; Wang, X.; Zhou, H. SWIPENET: Object detection in noisy underwater scenes. *Pattern Recognit.* **2022**, *132*, 108926. [CrossRef]
30. Paul, S.; Chen, P.Y. Vision transformers are robust learner. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 26–27 February 2022; Volume 36, pp. 2071–2081.