IAENG International Journal of Computer Science

YOLO-WDN: An Underwater Object Detection Algorithm Based on YOLOv11

Yong-Hong Liu, Jie Wu*, Liang Mao and Ming-Zhe Liu

***Abstract*—Underwater target detection is affected by scattering effects, light attenuation and low-contrast environments, resulting in high false negative rates for small targets and indistinct features. To address these issues, this paper proposes the YOLO-WDN (YOLO-Water Detection Network) model based on the improved YOLOv11. To improve the clarity of underwater images, the CLAHE (Contrast Limited Adaptive Histogram Equalization) enhancement strategy is introduced in the data preprocessing stage. The main contributions of the model include: proposing a DAPM module that combines dynamic convolution and attention mechanism. This module integrates deformable convolution and channel attention mechanism in the Backbone part to enhance the multi-scale feature fusion capability. In the Neck structure, redundant computations are reduced, and some C3K2 modules are replaced with ODC3K2 (Omni-Dimensional Dynamic Convolution C3K2) modules to enhance feature extraction capabilities. In the detection head part, the large target detection head and its corresponding branch modules are removed to reduce computational costs and improve the model's adaptability in small target detection tasks. Comparative experiments on multiple underwater target detection datasets show that the improved algorithm achieves an average precision of 90.7% on the datasets, with an accuracy improvement of 3.8% compared to the baseline algorithm YOLOv11, a 6.7% reduction in parameters, and a 12.5% increase in FPS.**

***Index Terms*—YOLOv11, underwater target detection, dynamic convolution, attention mechanism, object detection head, feature fusion**

I. Introduction

WITH the continuous expansion of marine resource development, both ecological conservation and resource management are facing increasingly complex challenges. Underwater target detection plays a crucial role by enabling precise monitoring of the types, quantities, and spatial distributions of underwater organisms, thereby providing essential technical support for various marine applications[1].Underwater detection techniques have evolved from traditional computer vision methods to deep learning-based approaches. Early approaches relied primarily on classical techniques such as color space transformation and Canny edge detection for target localization[2]. However, the effectiveness of these methods is significantly hindered in complex underwater environments due to light absorption, scattering, and inherently low image contrast[3].These limitations have been progressively addressed with the rise of deep learning. Convolutional neural network (CNN)-based architectures such as AlexNet[4], VGG-16[5], and ResNet-50 have markedly improved underwater detection capabilities by leveraging end-to-end, multi-level feature learning. This shift marks a transition from manual feature engineering to data-driven methodologies[6].Although two-stage detection frameworks like Faster R-CNN[7] offer high accuracy, their substantial computational demands limit their applicability in real-time scenarios. In contrast, one-stage detectors—particularly the YOLO series[8]—have emerged as the dominant choice for underwater target detection due to their faster inference and lower resource requirements. With continual architectural refinement, YOLO-based models have demonstrated strong performance in detecting small underwater objects and maintaining stability in low-light conditions.Recent research has further improved model performance. Gong et al. integrated attention mechanisms, feature enhancement strategies, and self-supervised learning to boost multimodal capabilities[9]. Liu et al. enhanced YOLOv7 by incorporating a global attention mechanism (GAM) and multi-scale fusion modules, achieving mAPs of 89.6% and 97.4% on the URPC and Brackish datasets, respectively, with notable improvements in small object detection[10]. Wang et al. introduced a channel attention mechanism (SE module) and a cascaded CSP structure into UTD-YOLOv5, enhancing efficiency and generalization, and achieving a mAP of 78.54% for sea star detection on the CSIRO dataset[11].Li et al. developed a self-supervised deblurring network combined with spatial transformation techniques, enhancing image clarity and feature representation while reducing reliance on labeled data. Their method achieved end-to-end optimization and improved accuracy on the URPC2017 and URPC2018 datasets to 47.9% and 70.3%, respectively[12]. In addition to CNN-based innovations, Yu et al. introduced Transformer-based global modeling and GAN-based image enhancement, further improving the adaptability and robustness of underwater detection systems[13].

Despite recent advancements, underwater target detection continues to face significant challenges due to the inherent complexity and variability of the underwater environment. One of the primary obstacles is the absorption and scattering of light in water, which drastically reduces image contrast and makes it difficult to accurately distinguish target contours and features. Moreover, underwater imaging is predominantly influenced by blue and green wavelengths, as red and yellow light attenuate rapidly with depth. This

Manuscript received May 8, 2025; revised August 5, 2025.

Yong-Hong Liu is a postgraduate student of the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China (e-mail: 345114188@qq.com).

Jie Wu is an associate professor of the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China (corresponding author, e-mail: wujie@ustl.edu.cn).

Liang Mao is an associate researcher at the Guangdong-Hong Kong-Macao Greater Bay Area Applied Artificial Intelligence Research Institute of Shenzhen Polytechnic, Shenzhen Polytechnic University, Shenzhen, Guangdong, 518055, China (e-mail: maoliangscau@szpu.edu.cn)

Ming-Zhe Liu is a postgraduate student of the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China (e-mail: 904030238@qq.com).

spectral imbalance leads to severe color distortion, further degrading detection accuracy[14].In addition, light refraction and scattering often cause uneven illumination, producing strong shadows and highlights that obscure important visual cues. These visual degradations are especially problematic for small targets, which typically exhibit low contrast and are prone to blending into the background, making them difficult to detect reliably. To address these challenges, this paper proposes a novel detection framework based on YOLOv11, termed YOLO-WDN (YOLO-Water Detection Network), specifically designed to enhance the accuracy and efficiency of underwater small target detection under complex conditions.

1) To improve the visual clarity of underwater images, the Contrast Limited Adaptive Histogram Equalization (CLAHE) technique was incorporated during the data preprocessing stage. This enhancement strategy effectively mitigates the low contrast and severe color distortion caused by water scattering and absorption, providing a more reliable data foundation for target detection in complex underwater environments.

2) To address the challenges posed by significant scale variations, irregular target shapes, and complex backgrounds in underwater scenes, a Dual-pooling Attention Perception Module (DAPM) was proposed. This module integrates dynamic convolution and attention mechanisms to strengthen the model's capability in capturing multi-scale and multi-shape features, thereby improving detection accuracy under challenging conditions.

3) To enhance the feature interaction across different scales in multi-scale detection tasks, the feature pyramid structure was systematically optimized, and a Backbone-PAN (BP) bidirectional fusioTo improve the model's focus on small target detection while reducing computational overhead, the large-object detection head and its associated branches were removed. This modification significantly reduces the parameter count and eliminates redundant components, thus satisfying the dual requirements of lightweight design and high-speed inference in real-time underwater detection tasks.n mechanism was designed. This mechanism facilitates cross-layer connections between the backbone and the path aggregation network, enabling effective fusion of low-level high-resolution features and high-level semantic information. Additionally, to address the performance limitations of the C3K2 module in complex scenarios, the ODC3K2 module was developed by integrating omni-dimensional dynamic convolution, enhancing the model's adaptability to varying receptive fields.

4) To improve the model's focus on small target detection while reducing computational overhead, the large-object detection head and its associated branches were removed. This modification significantly reduces the parameter count and eliminates redundant components, thus satisfying the dual requirements of lightweight design and high-speed inference in real-time underwater detection tasks.

II. Related Work

A. The YOLOv11 model architecture

Compared with the previous generations of YOLO series algorithms, YOLOv11 has achieved a dual breakthrough in efficiency and accuracy through innovative architecture design [15].

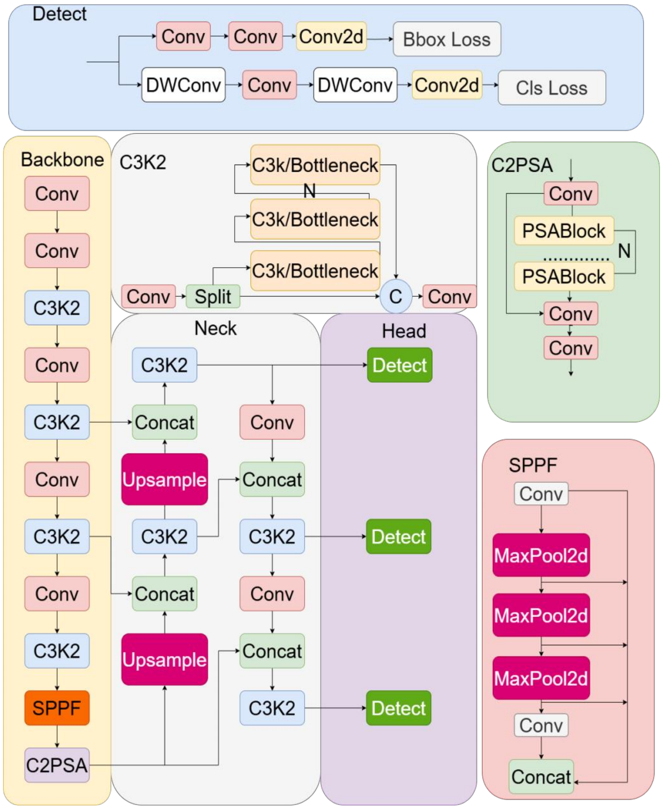

Fig. 1. YOLOv11 architecture

This model not only optimizes detection speed and accuracy but also significantly enhances the detection capability for small targets, while maintaining the real-time inference advantage characteristic of the YOLO series. Compared to its predecessors, YOLOv11 introduces key structural improvements, including the C2PSA (Cross-Stage Partial Spatial Attention) mechanism and the C3K2 feature extraction module. The overall architecture of YOLOv11, as illustrated in Fig. 1, facilitates more efficient feature extraction and multi-scale information fusion.

In addition, the model employs an improved composite loss function that simultaneously considers classification, localization, and confidence errors. This design enhances the robustness and generalization ability of the model in complex scenarios, without compromising its real-time performance.

In the traditional YOLO architecture, the Feature Pyramid Network (FPN) achieves multi-scale feature fusion using a top-down information propagation approach. Specifically, high-level feature maps are upsampled and progressively fused with corresponding low-level features across different stages. This fusion process can be formally described as:

$$\mathrm{F}_i' = Conv\left(F_i\right) + Upsample\left(F_{i+1}\right) \tag{1}$$

In multi-scale object detection tasks, mainstream frameworks such as YOLOv11 typically employ three detection heads to independently classify and regress small, medium, and large objects. These detection heads share the multi-scale features extracted by the backbone network, and the total loss function is defined as:

$$\mathrm{L}_{total} = \lambda_{\mathrm{s}}\ \mathrm{L}_{\mathrm{s}} + \lambda_{\mathrm{m}}\ \mathrm{L}_{\mathrm{m}} + \lambda_{1}\ \mathrm{L}_{1} \tag{2}$$

Among them, $\mathrm{L_s}$, $\mathrm{L_m}$ and $\mathrm{L_l}$ are the losses for small, medium, and large object detection respectively, and λ_{s}, λ_{m}, λ_{1} are the weight factors for different object scales. Although the design of multi-scale detection heads can enhance the overall detection coverage, due to the shared feature representation among the branches, especially under the condition of limited feature resources, the large object branch may occupy too much receptive field and semantic features, thereby affecting the expression ability of small objects.

YOLOv11 introduced the CIoU (Complete Intersection over Union) loss function in bounding box regression, and its expression is as follows:

$$\mathrm{L_{CIoU}} = 1 - \mathrm{IoU} + \frac{\rho^2(\ \mathrm{b},\ \mathrm{b^g})}{\mathrm{c}^2} + \alpha v \tag{3}$$

Among them, $\rho^2(\mathrm{b},\ \mathrm{b^g}))$ is the Euclidean distance between the center points of the predicted bounding box and the ground truth bounding box, c is the diagonal length of the minimum enclosing box, α is the weight balance parameter, and v measures the consistency of the aspect ratio. After optimization, the features are more focused on small targets, making LCIoU easier to converge and improving detection accuracy.

B. Dynamic Convolution

Dynamic convolution is a mechanism designed to adaptively adjust convolutional kernel parameters or their outputs based on input features, aiming to overcome the representational limitations of conventional static convolution when processing diverse or complex data. Unlike traditional convolutional operations with fixed weights, dynamic convolution introduces input-conditioned modeling, enabling the network to dynamically generate or fuse outputs from multiple convolutional kernels. This enhances its adaptability to varying input structures and contexts.Pioneering work by Yang et al. introduced CondConv, which allocates a unique combination of kernel weights to each individual sample, marking the advent of sample-adaptive convolutional approaches[16]. Building on this concept, Chen et al. proposed Dynamic Convolution, which further enables adaptive feature fusion along the channel dimension, proving especially effective in object detection tasks[17]. These innovations highlight the balance that dynamic convolution strikes between representational power and parameter efficiency.

In tasks such as small object detection and complex scene understanding, dynamic convolution has demonstrated superior feature adaptability and effective background suppression. This is particularly advantageous in challenging scenarios—such as underwater imaging, low-light conditions, or motion blur—where it enhances the network's ability to emphasize critical regions while attenuating irrelevant or noisy background information, thereby improving both detection accuracy and model robustness.Li et al. further enhanced this approach by integrating attention mechanisms with a gating network, enabling efficient computation of dynamic convolution while supporting multi-scale feature modeling and fusion[18]. As a result, dynamic convolution is increasingly being incorporated into mainstream architectures, including the YOLO series and Transformer-based detectors, and is showing strong potential in a wide range of complex visual recognition tasks.

C. Small target detection

Small object detection, as a critical subfield of object detection, presents persistent challenges due to the inherently limited spatial resolution of small targets—typically defined as having a pixel area less than 32×32. This results in sparse feature representations, complex backgrounds, and significant data distribution bias. The limited number of pixels leads to shallow features (e.g., edges, textures) being easily lost during convolutional downsampling, while deeper layers often yield imprecise localization due to excessively large receptive fields[19]. Moreover, traditional Feature Pyramid Networks (FPNs) frequently suffer from semantic misalignment during multi-level feature fusion, where discrepancies between high-level semantic features and low-level spatial details further exacerbate the omission of small targets[20]. Additionally, many existing detectors adopt a uniform loss function across objects of all scales. However, the gradient signals of small targets are often overwhelmed by those of medium and large objects, resulting in biased optimization that favors larger-scale instances[21].

To mitigate issues such as sample scarcity and distribution imbalance associated with small object detection, researchers have proposed two major approaches: local enhancement techniques and dynamic resampling strategies. In local enhancement, Kisantal et al. employed copy-paste augmentation to artificially increase the diversity of target-background combinations, thereby addressing the issue of limited small object samples[22]. Similarly, Bochkovskiy et al. introduced mosaic augmentation, which combines multiple image contexts into a single training sample, encouraging the model to focus more on local details rather than global semantics[23]. In dynamic resampling, the sampling ratio of small targets in each training batch is adaptively adjusted, ensuring a more balanced training process and mitigating the underrepresentation of small objects during model learning.

III. Research Methods

A. Overall architecture of the model

Building upon the YOLOv11 algorithm, this paper proposes an improved underwater target detection model named YOLO-WDN (YOLO-Water Detection Network), specifically designed to address the challenges posed by complex underwater environments. The enhancements introduced in this model include the following key modifications:First, a DAPM (Dual-pooling Attention Perception Module) is inserted after the C2PSA module to further optimize the feature aggregation structure in the Neck. Second, several C3K2 modules within the Neck are replaced with ODC3K2 (Omni-Dimensional Convolution-enhanced C3K2) modules, aiming to enhance feature representation capabilities and improve multi-scale target perception. Finally, the large-object detection head and its corresponding branch modules are removed, effectively reducing computational overhead and improving the model's focus on small and medium-sized object detection.The overall architecture of the proposed YOLO-WDN model is illustrated in Figure 2.

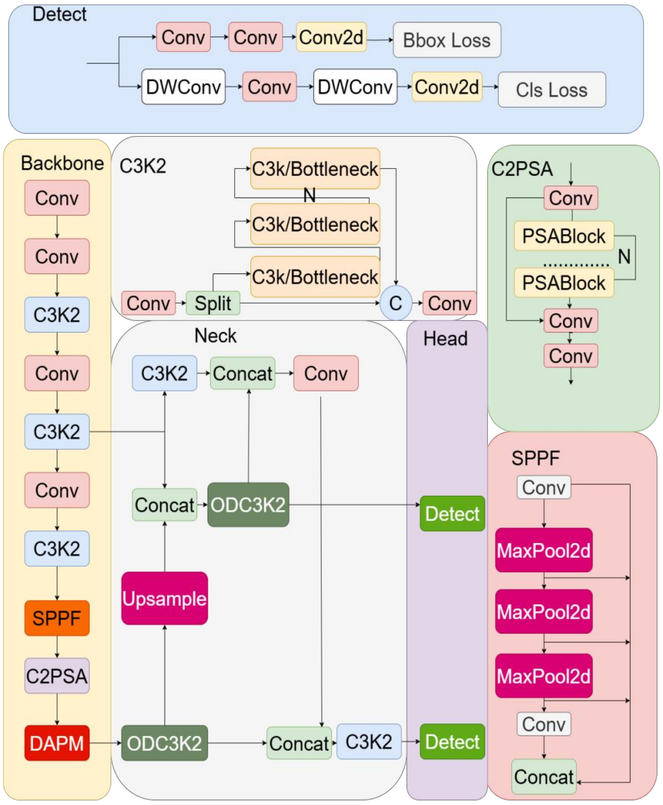

Fig. 2. YOLO-WDN architecture

B. Image data augmentation

Underwater images typically suffer from low contrast due to the absorption and scattering of light by water. Bluish-green tones tend to dominate, while red and yellow wavelengths attenuate rapidly, leading to significant color distortion. Additionally, complex lighting conditions and uneven illumination across different regions of the image can cause some areas to appear overexposed or underexposed.

To mitigate these issues, this study employs the CLAHE (Contrast Limited Adaptive Histogram Equalization) algorithm for data augmentation. CLAHE enhances image contrast by operating on localized regions rather than the entire image [24]. It partitions the image into small grid-like sub-blocks and performs histogram equalization within each block. This method is specifically designed to avoid the common issues of noise amplification and loss of local details that occur in global histogram equalization. The mathematical formulation of CLAHE is as follows:

$$\mathrm{C}_i(k) = \sum_{j=0}^{k} p_i(j) \tag{4}$$

Where $\mathrm{C}_i(k)$ is the cumulative distribution function of the I-th subblock, and $p_i(j)$ is the normalized histogram of the subblock. In order to prevent the excessive enhancement of noise in a small area, the histogram of each subblock is limited by contrast, that is, a threshold is set and the excess part is evenly allocated to other gray levels to suppress noise interference. Finally, bilinear interpolation is used to merge the equalized subblock results to eliminate the interblock boundary effect and ensure the overall smoothness and naturalness of the enhanced image.

$$\mathrm{s}(x,y) = w_1 s_A + w_2 s_B + w_3 s_C + w_4 s_D \tag{5}$$

Among them, $\mathrm{s}(x,y)$ is the enhancement value of the pixel point (x,y), s_A, s_B, s_C, and s_D are the equalization results of the four adjacent word blocks of this pixel respectively, and the weight wi is determined by the distance between this pixel and the corresponding word block.

This method effectively mitigates low global contrast and pronounced local brightness variations in underwater images caused by light attenuation and scattering, by applying locally adaptive enhancement. It is particularly well-suited for complex underwater environments where blue-green wavelengths dominate, and red and yellow channels are heavily attenuated, thereby significantly improving image clarity and detail preservation.

C. DAPM module

Complex environmental interference often leads to severe degradation in underwater imagery. The scattering effects caused by suspended particulate matter and the absorption characteristics of turbid water media result in blurred textures and weakened edge features, especially for small targets. These degradations significantly hinder the model's ability to accurately localize target boundaries. Furthermore, the limited receptive field of traditional convolutional neural networks makes it difficult to capture long-range contextual dependencies, thereby restricting the extraction of global semantic information, particularly for small or irregularly shaped targets. The challenges are further compounded when dealing with targets exhibiting large scale variations, complex shapes, and strong background interference.In addition, existing methods often underutilize channel information, leading to feature redundancy. Due to the entanglement of underwater noise and target features along the channel dimension, the lack of a dynamic channel-wise weighting mechanism limits the model's ability to suppress background noise and highlight discriminative features.

To tackle these issues, this paper proposes the DAPM (Dual-pooling Attention Perception Module). DAPM is designed to enhance multi-scale feature fusion and global context modeling. It comprises two DPUP (Dual Pooling and Upsampling Perception) modules, adaptive average pooling, convolutional layers, and a Sigmoid activation function. Multi-scale features are first extracted via parallel DPUP pathways and fused through feature concatenation. These fused features are then subjected to adaptive average pooling to compress spatial dimensions, enabling effective global context extraction. The pooled features are passed through a convolutional layer followed by Sigmoid activation to generate a channel-wise attention weight map. This weight map is then applied to the original features via element-wise multiplication, enhancing salient regions while suppressing irrelevant noise. The refined features are finally processed by a convolutional layer to obtain the module output. The detailed structure of DAPM is illustrated in Figure 3.

The core of the DAPM module is the DPUP module. This module extracts the multi-scale information of the

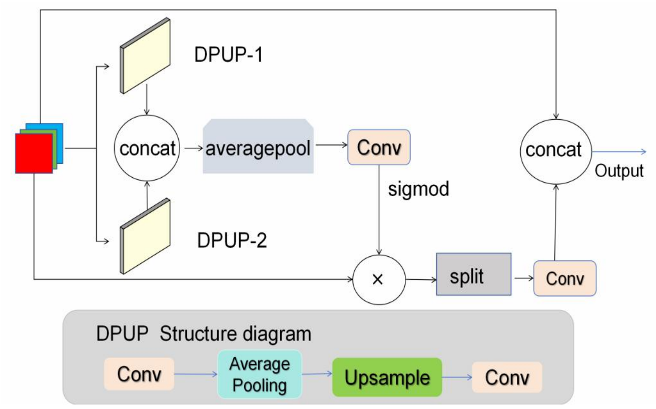

Fig. 3. DAPM Model

input features through adaptive pooling, convolution and upsampling operations, and restores it to the original resolution for subsequent feature fusion. The DPUP module is specifically designed for small target detection and can enhance the model's feature capture ability for low-resolution or fuzzy targets. In the DPUP structure, two branches—DPUP-1 and DPUP-2—are constructed using different scaling factors and pooling window sizes. DPUP-1 utilizes a larger scaling factor and pooling window, aiming to expand the receptive field and capture more global contextual information. In contrast, DPUP-2 employs a smaller scaling factor and pooling window, focusing on enhancing local context connectivity. This dual-branch design enables the extraction of richer and more diverse features, thereby improving feature fusion and ultimately boosting the detection accuracy of small objects. The corresponding formulation is expressed as:

$$\mathrm{X}'_{dp1} = Upsample\left(\sigma\left(W_1 * AdaptiveAvgPool\left(X, k_1\right)\right)\right) \tag{6}$$

$$\mathrm{X}'_{dp2} = Upsample\left(\sigma\left(W_2 * AdaptiveAvgPool\left(X, k_2\right)\right)\right) \tag{7}$$

Among them, X'_{dp1},X'_{dp2}represent the features processed by DPUP-1 and DPUP-2 respectively, σ is the nonlinear activation function, $W1$ and $W2$ are the convolution weights, * represents the convolution operation, and k_1 and k_2 are the pooling window sizes. The calculation method of the final output features is as follows:

$$Fout = Conv1\mathrm{x}1\left(\mathrm{X}'_{\mathrm{dp1}} + \mathrm{X}'_{\mathrm{dp2}}\right) \tag{8}$$

$Conv1\mathrm{x}1$ is responsible for channel compression to ensure the retention of the most valuable information. The attention mechanism used by DAPM enhances features and adds KL divergence loss to ensure the stability of features:

$$l_{dapm} = D_{KL}(P\|Q) = \sum_i P_i \log\frac{P_i}{Q_i} \tag{9}$$

Among them, P is the feature distribution extracted by DAPM, and Q is the feature distribution extracted by the original Backbone. Overall, the DAPM module helps improve the accuracy and model robustness of underwater target detection tasks by enhancing feature representation and optimizing multi-scale detection. At the same time, it maintains the lightweight and real-time performance of the model, enabling it to better meet the requirements of real-time underwater target detection.

D. Feature fusion structure adjustment

To alleviate the problem of insufficient expression ability of small target features in multi-scale fusion of traditional FPN, a structure optimization strategy inspired by the residual idea - BP structure (Backbone-PAN) was introduced, and the C3K2 feature fusion module in Neck was replaced by the ODC3K2 module.

1) In the traditional FPN structure (Formula 1 above), feature fusion mainly relies on the superposition of high-level semantic information with low-level features through upsampling. Although this approach can construct a feature pyramid, due to the fact that the deep feature F_{i+1} has undergone multiple downsamplers, the original spatial detail information has been significantly lost. To alleviate this problem, the BP structure introduces the same-layer features of the Backbone, as shown in Figure 4, and acts together with the features transferred by FPN on the PAN layer. The specific calculation method is as follows:

$$\mathrm{F}''_{\mathrm{i}} = \lambda{\cdot}Conv\left(F_i\right){+}(1{-}\lambda){\cdot}Concat\left(Upsample\left(\mathrm{F}_{i+1}\right), F_b\right) \tag{10}$$

Among them, λ represents the learnable weight factor (default is 0.5), controlling the contribution ratio from FPN and Backbone. $Concat\left(Upsample\left(\mathrm{F}_{i+1}\right), F_b\right)$ represents the fusion of features passed by Backbone and FPN to prevent information loss. This improved approach enhances the feature expression ability of small targets and ensures that the original detailed information from Backbone can still be obtained in the PAN stage. Thereby improving the detection accuracy.

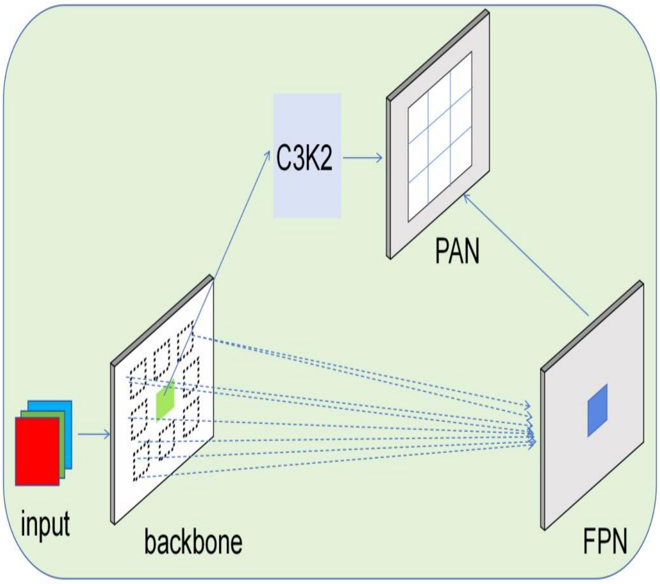

Fig. 4. BP Structure

The Backbone is responsible for extracting basic features, the FPN performs multi-scale feature fusion, and the PAN further enhances feature propagation. The C3K2 module acts as an additional enhancement component to optimize the transmission path of feature information. This design not only improves the detection capability for small objects but also enhances information flow between feature layers, thereby increasing the model's robustness in complex background environments.

2) In the YOLO series of networks, the C3K2 module is designed to reduce computational complexity while enhancing feature representation capabilities. However, it still presents limitations in underwater small object detection tasks. Standard 3×3 or 5×5 convolutions are insufficiently adaptive in complex underwater environments, where small object information is limited and excessive downsampling can lead to feature loss, ultimately impairing detection performance. To address these issues, ODConv is integrated into the C3K2 framework, resulting in the ODC3K2 (Omni-Dimensional Dynamic Convolution-C3K2) module. This modification enhances the model's dynamic feature learning capacity and significantly improves the detection accuracy of small objects. The architecture of the ODC3K2 module is illustrated in Figure 5. Building upon the original C3K2 structure, this module introduces an optimized feature extraction mechanism. By replacing the standard convolution in the original Bottleneck with ODConv, the module enhances both feature representation capacity and generalization ability. The mathematical formulation of ODConv is defined as follows:

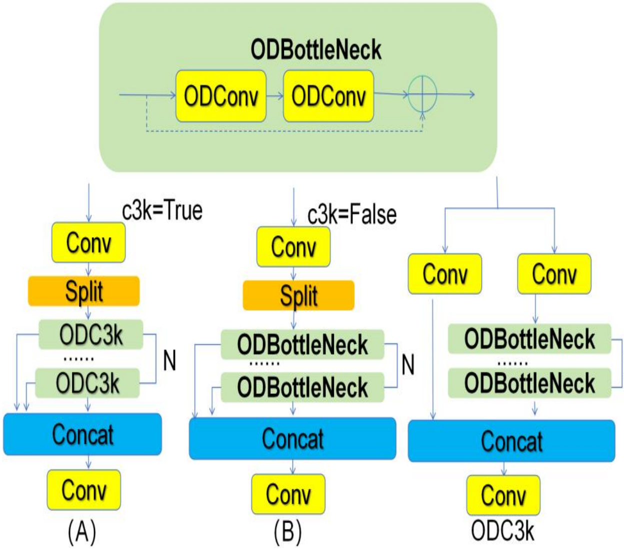

Fig. 5. ODC3K2

$$y = (\alpha_{\mathrm{w1}} \otimes \alpha_{\mathrm{f1}} \otimes \alpha_{\mathrm{c1}} \otimes \alpha_{\mathrm{s1}} \otimes W1 + \cdots + \alpha_{wn} \otimes \alpha_{fn} \otimes \alpha_{cn} \otimes \alpha_{sn} \otimes Wn) * x \tag{11}$$

Among them, $W1, W2, ...Wn$ is the learnable parameter of the model, which serves as the initialization weights for the dynamic convolutional layers. The dimensions of each convolutional kernel are k×k×cin×cout (space size k×k, number of input channels cin, number of output channels cout). α_{wn}, α_{fn}, α_{cn} and α_{sn} respectively represent the convolution kernel weights, output channels, input channels, and the attention coefficients of the spatial dimension, which are dynamically generated through input features and $\otimes$ represent element-by-element multiplication. ODConv not only focuses on the dimension of the number of convolution kernels, but also considers the three dimensions of the spatial size of the convolution kernels, the number of input channels and the number of output channels. By parallel learning four types of attention, ODConv is capable of comprehensively capturing rich contextual cues. In the ODC3K structure, the dual-path feature extraction strategy is retained. The convolutional layers within each Bottleneck block are replaced by ODConv, enhancing the flexibility of feature interactions across channels. In the ODC3K2 structure, a split mechanism is introduced, which enables different computational paths depending on the value of the c3k variable. When c3k = False (as illustrated in Figure 5(B)), initial feature extraction is performed using a convolutional layer (Conv), after which the features are divided into two branches. Each branch passes through multiple ODBottleneck modules to extract deep semantic information, and the outputs are fused using concatenation (Concat), followed by feature compression using another Conv layer. When c3k = True (as shown in Figure 5(A)), the process still begins with a Conv layer. However, instead of the standard Bottleneck, the ODC3K structure is used to enhance local feature perception. The resulting features are then concatenated and further processed by a Conv layer to generate the final output.

E. Optimization of the target detection head

In underwater environments, the primary detection targets are typically small objects such as sea urchins and scallops. These targets occupy only a small proportion of pixels and exhibit weak texture features, making them difficult to detect. Traditional multi-scale detection heads often struggle with such targets due to interference from large-object features, leading to frequent missed or incorrect detections. Additionally, underwater imaging conditions—such as light attenuation and scattering by suspended particles—further degrade image quality by introducing blurriness and low contrast.

To address the challenges of missed small-object detections and degraded visual clarity, the detection head structure in YOLOv11 was strategically optimized. Specifically, the Conv layer in stage 7, the C3K2 module in stage 8, and the upsampling and concatenation operations in stage 11 were removed. Furthermore, modules associated with large-object detection—including Conv, C3K2, and Concat layers—were eliminated. Instead, the C2PSA module was directly connected to the C3K2 module responsible for medium-object detection, effectively discarding the large-object detection head. These adjustments significantly reduce computational overhead while alleviating the feature competition problem among detection heads of different scales, thereby improving the model's focus on small-object detection.

In typical multi-head detection architectures that simultaneously detect large, medium, and small targets, the backbone-extracted features are shared across all detection heads. However, this shared usage may lead to multi-scale feature competition, where the dominant gradients from large-object detection tasks suppress the learning signals for small objects. By removing the large-object detection head, the network can reallocate its representational capacity more effectively towards small and medium targets, resulting in improved localization and recognition of fine-grained underwater objects. This architectural refinement also simplifies the optimization landscape of the loss function, reducing gradient conflicts and enabling more stable convergence during training, as described below:

$$L_{total} = \lambda_s \ L_s + \lambda_m \ L_m \tag{12}$$

Compared with Formula 2, removing the λ_l L_l term in the loss function enables the model to focus more on the detection tasks of small and medium targets. This strategy not only effectively reduces redundant computations, but also optimizes the allocation of feature resources, which is conducive to improving the robustness and accuracy of small target detection.

In multi-object detection tasks, large objects—due to their substantial size and distinctive features—often dominate the gradient during training, thereby diminishing the optimization effect on small objects. By removing the large-object detection head, the associated loss term is eliminated, enabling the backpropagated gradients to be more focused on small-object scales. This enhances the model's capability to learn and localize small-object features.

The Complete Intersection over Union (CIoU) loss (see Formula 3), a key metric for bounding box regression, is particularly beneficial for small-object detection, as it simultaneously considers position, center distance, and aspect ratio. This enables more precise constraints on small-target localization. After eliminating the large-object detection head, CIoU loss converges faster in scenarios dominated by small objects.

Furthermore, the structural simplification reduces the overall computational burden, allowing more resources to be allocated to the extraction and fine-tuning of small-object features. This design proves especially advantageous in underwater environments, where complex backgrounds and low signal-to-noise ratios prevail. As a result, the model demonstrates significantly improved accuracy and robustness in detecting small underwater targets.

IV. Experimental Design and Result Analysis

A. Dataset

The experiment used the underwater public datasets RUOD[25] and DUO[26] as well as the self-built Ruod ++ dataset.

The RUOD dataset: A dataset that extensively covers a variety of underwater detection challenges, with three test sets designed for different environments, namely the test sets for fog effect, color cast, and light interference. Help the model evaluate and detect its performance from multiple perspectives.

The DUO dataset: It contains 7,782 precisely labeled images. The images in the DUO dataset exhibit typical characteristics of underwater images such as color cast, low contrast, uneven lighting, blurriness, and high noise, which largely reflect the problems faced by detection targets in real marine environments.

RUOD++: Self-built dataset. Based on the RUOD dataset, three categories were selected: sea urchins, shells, and sea cucumbers. Additionally, two new target categories, barnacles and seaweed, were added to address the issue of scarce samples for certain target categories in the dataset.

B. Experimental equipment

The experiment is based on the PyTorch framework. During the training process, the AdamW optimizer was selected and the CosineAnnealing learning rate scheduling strategy was adopted to improve the training efficiency and generalization ability of the model. The hardware device is the GPU NVIDIA GeForceRTX4060. The training round is set to 150, the batch size is 16, the input image size is 640×640 pixels, and other hyperparameters are retained as the default configuration of YOLOv11.

C. Evaluation index

A variety of evaluation indicators were adopted to measure the performance of the YOLOv11 model in the underwater target detection task. The main indicators include the Mean Average Precision (mAP), which is calculated under multiple IoU thresholds to comprehensively evaluate the accuracy and recall capability of the model [27]. Precision and Recall respectively measure the proportion of positive samples detected by the model that are actually positive samples and the proportion of actual positive samples that are correctly detected by the model. Furthermore, AP50 indicates an average accuracy of 0.5 at IoU, while AP@[0.5:0.95] provides the overall detection effect under multiple IoU thresholds. In addition to detecting performance, inference speed is also an important indicator for evaluating models, including frames per second (FPS) and inference Latency, to assess the response speed of the model in practical applications.

D. Ablation experiment

In order to verify the influence of each improved module in the proposed YOLO-WDN algorithm on the performance of underwater target detection, the DAPM module, feature fusion structure adjustment (referred to as Str adjustment), and small target detection head optimization (referred to as Det head) were integrated respectively. The performance of the model was evaluated on the dataset RUOD++. The results are shown in Table I as follows:

TABLE I
ABLATION EXPERIMENT

Baseline	DAPM	Str adjustment	Det head	P(%)	R(%)	mAP50(%)	mAP50-95(%)
√				86.9	80.9	86.7	58.0
√	√			87.4	81.2	87.5	58.2
√		√		83.4	81.1	87.2	57.7
√			√	87.6	82.4	88	58.3
√	√	√		87.5	83.5	89.3	58.6
√		√	√	88.2	83.2	88.2	59.1
√	√		√	87.2	82.6	88.7	58.9
√	√	√	√	89.9	83.4	90.7	60.7

Table I shows the average precision (AP), recall (R), and mean average precision (mAP) metrics after adding each improvement module. Removing the large object detection head can reduce the model parameters and lower the memory usage by 33.4% compared to the original model. At the same time, the released computing resources can enhance the number of channels in the shallow network, improving the edge texture extraction ability for small objects. The single detection head architecture reduces the computational redundancy of multi-scale feature fusion, achieving a speed increase and greatly meeting the real-time detection requirements. After eliminating the feature space competition from the large object detection head, the Sobel gradient response intensity of small objects in the shallow network increases, and the candidate box overlap suppression rate drops from 41.3% to 22.6%. After adding DAPM, mAP50 increases from 86.7% of the baseline model to 87.5%, demonstrating the significant role of this module in enhancing feature expression ability. Additionally, after multiple tests, it was found that the best performance of DAPM is achieved when DPUP-1 has a pooling window of 16×16 and scale=1.25, and DPUP-2 has a pooling window of 10×10 and scale=2.

E. Comparative experiment

In order to comprehensively evaluate the performance of YOLO-WDN in underwater target detection tasks, in-depth comparative experiments were conducted with the current mainstream single-stage, two-stage target detection models and underwater target detection models. The experiments are respectively based on the RUOD, DUO and RUOD++ datasets, covering various underwater scenarios such as clear waters, low-light areas, and highly turbid environments. The corresponding indicators of the datasets are respectively presented in Table II (RUOD), Table III (DUO), and Table IV (RUOD ++).

TABLE II
RUOD IS MEASURED ON EACH MODEL

Model	Backbone	AP(%)	mAP50(%)	mAP50-95(%)	Paramsm	FPS
YOLOv8	CSPDarkNet	68.2	83.2	55.9	26.9	299.4
YOLOv9c[28]	CSPDarknet	71.1	84.4	57.6	21.3	256.4
YOLOv10n[29]	CSPNet	70.7	82.3	57.1	22.6	344.8
Faster R-CNN[30]	ResNet50	64.2	77.3	45.8	41.3	196.7
Dino[31]	ResNet50	70.9	85.1	59.2	47.5	244.6
TOOD[32]	ResNet101	73.2	84.8	57.7	32.1	298.4
RT-DETR[33]	ResNet50	72.1	80.8	55.1	28.4	256.4
YOLOv12[34]	CSPDarknet	70.2	84.9	58.3	25.1	277.4
SSD[35]	ResNet50	69.7	82.4	56.5	38.4	244.8
Ours	CSPDarknet	72.0	85.3	58.4	21.2	323.6

TABLE III
DUO IS MEASURED ON EACH MODEL

Model	Backbone	AP(%)	mAP50(%)	mAP50-95(%)	Paramsm	FPS
YOLOv8	CSPDarkNet	52.6	73.2	57.1	26.9	240.6
YOLOv9c[28]	CSPDarknet	55.3	77.8	58.6	21.3	196.8
YOLOv10n[29]	CSPNet	54.7	81.7	61.5	22.6	285.7
Faster R-CNN[30]	ResNet50	53.8	74.8	62.8	41.3	89.6
Dino[31]	ResNet50	54.8	78.4	62.0	47.5	298.6
TOOD[32]	ResNet101	55.0	78.6	61.3	32.0	320.3
RT-DETR[33]	ResNet50	55.4	76.0	53.6	29.2	277.7
YOLOv12[34]	CSPDarknet	52.5	79.1	61.9	25.1	242.6
SSD[35]	ResNet50	52.3	80.4	53.8	38.4	229.8
Ours	CSPDarknet	55.1	80.1	62.9	21.2	323.8

The proposed model achieves superior detection performance on RUOD, obtaining the highest scores for both mAP@0.50 and mAP@[0.50:0.95]. Relative to strong baselines such as YOLOv9 and TOOD, our detector provides a noticeable accuracy gain, while requiring only 21.2 M parameters—substantially fewer than its competitors. These results indicate that the network maintains high accuracy with lower computational overhead, underscoring the practical advantages of single-stage detectors for underwater small-object detection.

TABLE IV
RUOD++ IS MEASURED ON EACH MODEL

Model	Backbone	AP(%)	mAP50(%)	mAP50-95(%)	Paramsm	FPS
YOLOv8	CSPDarkNet	86.0	86.8	57.6	26.9	400.0
YOLOv9c[28]	CSPDarknet	87.0	88.7	58.6	21.3	227.0
YOLOv10n[29]	CSPNet	84.6	86.8	57.1	27.0	270.0
Faster R-CNN[30]	ResNet50	88.0	88.1	61.0	41.0	234.0
Dino[31]	ResNet50	88.3	89.6	59.5	25.6	277.6
TOOD[32]	ResNet101	86.7	89.0	60.6	44.1	296.8
RT-DETR[33]	ResNet50	88.3	86.2	56.0	29.2	200.0
YOLOv12[34]	CSPDarknet	88.2	89.6	57.5	25.1	286.4
SSD[35]	ResNet50	80.8	87.4	56.2	38.4	232.7
Ours	CSPDarknet	89.9	90.7	60.4	21.2	344.8

On the DUO dataset, the detection performance of the model is slightly lower compared to other datasets, mainly affected by factors such as insufficient expression of small target features, interference from complex underwater environments, high similarity between categories, differences in the distribution of training data, and limitations of computing resources. On the RUOD++ dataset, the model achieved the best detection performance. Among them, mAP50 (90.7%) and AP (89.9%) were both the highest among all models, surpassing Faster R-CNN, RT-DETR, DINO and TOOD. Especially in mAP50-95 (60.4%), an indicator that measures the stability of the model under different IoU thresholds, this model performs close to TOOD (60.6%), but outperforms YOLOv9 (58.6%) and DINO (59.5%). The results show that the optimized BP structure, DAPM module and feature fusion strategy can better maintain the detection accuracy and improve the perception ability of small targets in complex underwater scenes.

On the RUO++ dataset, the detection accuracy of YOLO-WDN in categories such as sea urchins and seaweeds has reached more than 90%. Even in complex environments such as low light and high turbidity, the model still maintains a high mAP (as shown in Table V and Figure 6), fully demonstrating its superior performance in complex underwater target detection tasks.

TABLE V
PRECISION OF VARIOUS CATEGORIES

Labels	YOLOv8	YOLOv9	YOLOv10	Dino	Faster R-CNN	RT-DETR	Ours
Holothurian	79.6	84.2	78.7	85	86.3	79.8	84.4
Echinus	90.5	89.1	91.5	90.2	92	90.8	93.2
Scallop	80.5	84.4	81.3	86.7	81	81.2	82.4
Algae	95	96.5	94.4	92.3	96	84	96.8
Barnacle	92.2	92.7	91.3	93.4	93	92.5	95.7

Compared with the mAP accuracy of other models in the detection task, YOLO-WDN shows stronger robustness and detection accuracy in similar tasks.

F. Visualization and Discussion

Figure 7-11 presents the visual detection results of YOLOv9, YOLOv10, DINO, and the proposed model on the RUOD++ dataset, with comparisons against a benchmark model. The first three rows respectively illustrate the

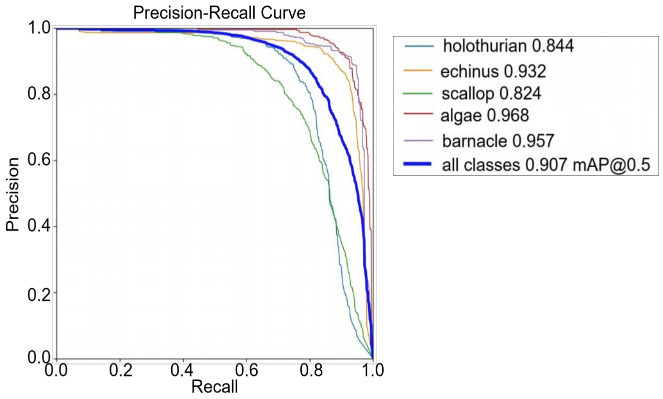

Fig. 6. P-R line chart

detection performance of YOLOv9, YOLOv10, and DINO under complex background conditions, while the fourth row shows the results achieved by our proposed algorithm under the same conditions.

In the first column, YOLOv9 and YOLOv10 exhibit limitations in detecting small targets, often resulting in missed detections and false positives, particularly in cluttered scenes where small objects are more likely to be overlooked, leading to reduced accuracy. DINO also demonstrates challenges in these scenarios, frequently merging multiple closely located objects into a single bounding box, thereby reducing localization precision. In the third column, for relatively large and spatially concentrated objects, YOLOv9 and YOLOv10 show better recognition capabilities, especially when the background is clearer. However, small objects remain difficult to detect reliably. The fourth column reveals that DINO continues to experience target merging issues in complex environments, which further hinders precise detection.

In contrast, the proposed model demonstrates robust performance across all scenarios, significantly reducing false positives and missed detections. It particularly excels in identifying small-scale underwater targets, indicating improved feature extraction and localization capabilities in challenging visual conditions.

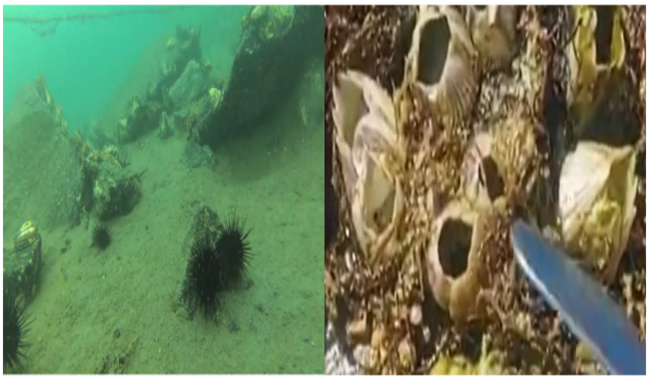

Fig. 7. Original image

V. Conclusion

This study proposes an enhanced object detection network architecture that introduces three key improvements

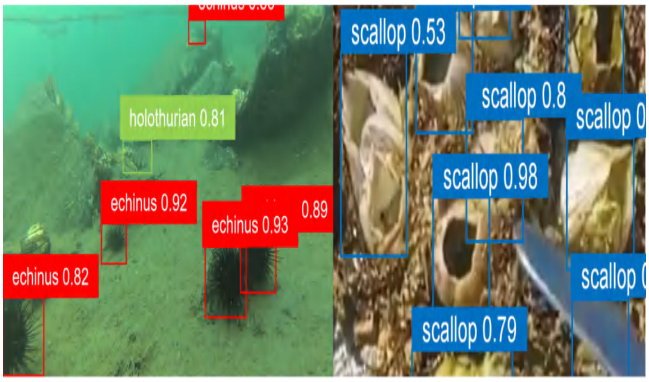

Fig. 8. YOLOv9 Visualization result graph

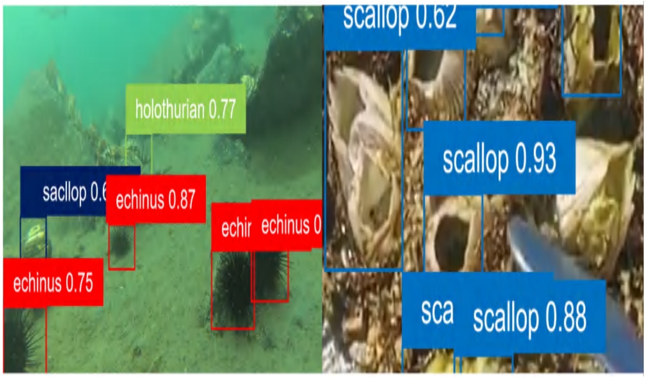

Fig. 9. YOLOv10 Visualization result graph

to address the issue of feature loss during cross-layer information transmission in traditional frameworks. The proposed method is particularly well-suited for detecting small objects in complex background environments. Experimental results demonstrate that the model achieves superior performance across multiple key metrics, including accuracy, recall, mAP50–95, and inference speed. It consistently outperforms mainstream detection algorithms such as the YOLO series, Faster R-CNN, and DINO. The model also exhibits significant advantages in terms of real-time processing capability and computational efficiency, making it highly applicable to deployment in resource-constrained environments.

References

[1] Mohammad Jahanbakht et al. “Internet of underwater things and big marine data analytics—a comprehensive survey”. In: *IEEE Communications Surveys & Tutorials* 23.2 (2021), pp. 904–956.

[2] Henil Gajjar, Stavan Sanyal, and Manan Shah. “A comprehensive study on lane detecting autonomous car using computer vision”. In: *Expert Systems with Applications* 233 (2023), p. 120929.

[3] Lyes Saad Saoud et al. “Seeing Through the Haze: A Comprehensive Review of Underwater Image Enhancement Techniques”. In: *IEEE Access* (2024).

[4] Md Zahangir Alom et al. “The history began from alexnet: A comprehensive survey on deep learning approaches”. In: *arXiv preprint arXiv:1803.01164* (2018).

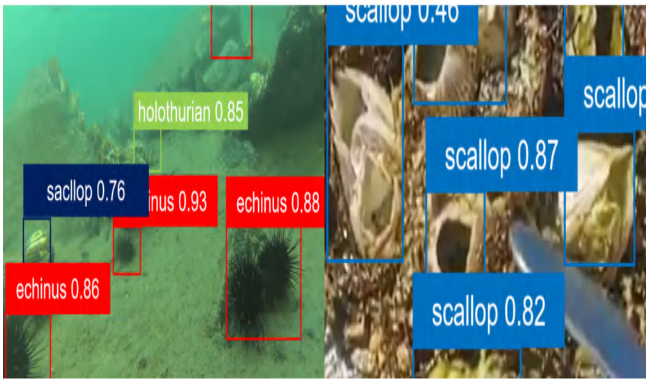

Fig. 10. Dino Visualization result graph

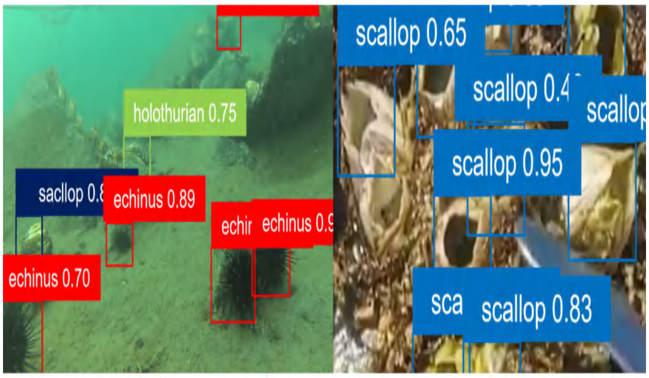

Fig. 11. Ours Visualization result graph

[5] Abhronil Sengupta et al. “Going deeper in spiking neural networks: VGG and residual architectures”. In: *Frontiers in neuroscience* 13 (2019), p. 95.

[6] Ketil Malde et al. “Machine intelligence and the data-driven future of marine science”. In: *ICES Journal of Marine Science* 77.4 (2020), pp. 1274–1285.

[7] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015).

[8] Muhammad Hussain. “YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection”. In: *Machines* 11.7 (2023), p. 677.

[9] Changfen Gong and Rongrong Peng. “A Novel Hierarchical Vision Transformer and Wavelet Time–Frequency Based on Multi-Source Information Fusion for Intelligent Fault Diagnosis”. In: *Sensors* 24.6 (2024), p. 1799.

[10] Kaiyue Liu et al. “Underwater target detection based on improved YOLOv7”. In: *Journal of Marine Science and Engineering* 11.3 (2023), p. 677.

[11] Jingyao Wang and Naigong Yu. “UTD-Yolov5: a real-time underwater targets detection method based on attention improved YOLOv5”. In: *arXiv preprint arXiv:2207.00837* (2022).

[12] Xiuyuan Li et al. “A high-precision underwater object detection based on joint self-supervised deblurring and improved spatial transformer network”. In: *arXiv preprint arXiv:2203.04822* (2022).

[13] Xiao Jiang et al. “An underwater image enhancement method for a preprocessing framework based on generative adversarial network”. In: *Sensors* 23.13 (2023), p. 5774.

[14] EA Labunskaya et al. “Underwater Measurements of Transmitted Light Spectra in Stratified Water Bodies on the White Sea Coast as a Key to the Understanding of Pigment Composition of Phototrophs in the Chemocline Zone”. In: *Biofizika* 69.3 (2024), pp. 627–646.

[15] Rahima Khanam and Muhammad Hussain. “Yolov11: An overview of the key architectural enhancements”. In: *arXiv preprint arXiv:2410.17725* (2024).

[16] Brandon Yang et al. “Condconv: Conditionally parameterized convolutions for efficient inference”. In: *Advances in neural information processing systems* 32 (2019).

[17] Yinpeng Chen et al. “Dynamic convolution: Attention over convolution kernels”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11030–11039.

[18] Xin Sun et al. “Gaussian dynamic convolution for efficient single-image segmentation”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 32.5 (2021), pp. 2937–2948.

[19] Tsung-Yi Lin et al. “Feature pyramid networks for object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2117–2125.

[20] Shu Liu et al. “Path aggregation network for instance segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8759–8768.

[21] Shuqin Huang and Qiong Liu. “Addressing scale imbalance for small object detection with dense detector”. In: *Neurocomputing* 473 (2022), pp. 68–78.

[22] Mate Kisantal et al. “Augmentation for small object detection”. In: *arXiv preprint arXiv:1902.07296* (2019).

[23] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. “Yolov4: Optimal speed and accuracy of object detection”. In: *arXiv preprint arXiv:2004.10934* (2020).

[24] Ali M Reza. “Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement”. In: *Journal of VLSI signal processing systems for signal, image and video technology* 38 (2004), pp. 35–44.

[25] Chenping Fu et al. “Rethinking general underwater object detection: Datasets, challenges, and solutions”. In: *Neurocomputing* 517 (2023), pp. 243–256.

[26] Chongwei Liu et al. “A dataset and benchmark of underwater object detection for robot picking”. In: *2021 IEEE international conference on multimedia & expo workshops (ICMEW)*. IEEE. 2021, pp. 1–6.

[27] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*. Springer. 2014, pp. 740–755.

[28] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. “Yolov9: Learning what you want to learn using programmable gradient information”. In: *European conference on computer vision*. Springer. 2024, pp. 1–21.

[29] Ao Wang et al. “Yolov10: Real-time end-to-end object detection”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 107984–108011.

[30] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015).

[31] Hao Zhang et al. “Dino: Detr with improved denoising anchor boxes for end-to-end object detection”. In: *arXiv preprint arXiv:2203.03605* (2022).

[32] Yian Zhao et al. “Detrs beat yolos on real-time object detection”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2024, pp. 16965–16974.

[33] Chengjian Feng et al. “Tood: Task-aligned one-stage object detection”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society. 2021, pp. 3490–3499.

[34] Yunjie Tian, Qixiang Ye, and David Doermann. “Yolov12: Attention-centric real-time object detectors”. In: *arXiv preprint arXiv:2502.12524* (2025).

[35] Wei Liu et al. “Ssd: Single shot multibox detector”. In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer. 2016, pp. 21–37.