DRL-YOLO: A Small Target Detection Model in Fuzzy Scenes Combining Multi-Perspective Feature Fusion with a Lightweight Detection Head

Kexin Zhang, Ziwei Zhou*

Abstract—**This research presents a new target detection network model, the DRL-YOLO model, in response to the YOLO network's limitations in identifying small targets in uncertain settings. The DRL-YOLO network model is based on the YOLO architecture and integrates dynamic snake convolution in the Backbone component. This offers the model a strong approach for feature extraction through a multi-perspective feature fusion strategy. In the Neck section, the DRL-YOLO model incorporates the RepViT Block module, utilizing the self-attention mechanism of ViT to enhance the feature extraction efficiency of the CNN for improved feature processing. The Head section introduces PLDetect, an innovative detection head that markedly reduces the model's computational complexity while maintaining accuracy through structural innovation and replacing the original convolution module. The DRL-YOLO model exhibited improvements in mAP of 1.7% and 1.3% on the DUO and URPC2020 datasets, respectively, compared to the baseline model, alongside a significant 40.1% decrease in GFLOPs. The experimental results validate that DRL-YOLO provides an optimal solution.**

Index Terms—**lightweight, multi-view feature fusion, small target detection, YOLOv8s**

I. Introduction

Detection of small targets in ambiguous environments is a significant research focus in computer vision, with extensive applications in several fields, including security surveillance, medical imaging, and biological detection. The primary challenge in locating small targets is their minimal spatial presence in the image and the scarcity of discernible features. Traditional target identification algorithms can be deceived by background noise while attempting to identify small targets, diminishing their accuracy and dependability. In intricate fuzzy environments, the challenge of detecting small targets is exacerbated by numerous additional considerations. Occlusion occurrences in the scene might partially or conceal the target's feature information, hindering the detector's ability to reliably identify the target. The blurring effect diminishes image quality, renders the edge and texture details of the target less discernible, and complicates feature extraction. Variations in lighting conditions will affect the visual performance of the target; specifically, brightness and contrast will diminish in low-light environments, while overexposure may arise in intense lighting conditions, both of which will negatively influence the detection of small targets. In practical small target identification applications, real-time mobile devices are preferred due to their intricate settings. Although conventional big target detection models exhibit excellent recognition accuracy, they are computationally demanding, possess several parameters, and pose challenges for deployment on mobile devices or embedded systems.

Deep learning is ineffective in detecting small objects in complex scenarios because of its extensive training samples, protracted training duration, and intricate network architecture, which complicate its application on standard devices. This research introduces the novel network model DRL-YOLO to address these difficulties. DRL-YOLO achieves an optimal equilibrium between model complexity and accuracy, enhancing precision while diminishing computing speed.

This paper's primary contributions are as follows:

The Backbone part incorporates C2f_DynamicConv to augment a segment of the basic C2f architecture. The application of the quantization technique using C2f_DynamicConv enhances the understanding of gradient flow while maintaining a lightweight model. This accelerates the model's convergence and enhances training efficacy.

A refined C2f module, derived from RepViT, partially substitutes the original module in the Neck portion. RepViT Blocks alter the sequence of the 3x3 depth separable convolutions within the MobileNetV3 module and consolidate them into a unified branch. This accelerates feature representation and processing.

This work proposes PLDetect, a lightweight detection head based on PConv, in the head portion. By meticulously streamlining the original framework, PLDetect enhances detection accuracy while markedly accelerating operational speed. This enhancement renders the detecting head more appropriate for real-time, demanding application settings while preserving efficient performance.

II. Related works

In recent years, various methods have arisen to tackle intricate application scenarios, constrained resources, and

Manuscript received January 8, 2025; revised March 22, 2025.

K. X. Zhang is a postgraduate student at the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China (e-mail: 582694941@qq.com).

Z. W. Zhou is a Professor at the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan 114051, China (corresponding author, phone: 86-13941255680; e-mail: 381431970@qq.com).

additional obstacles. Although the conventional R-CNN [1] attains superior accuracy, it incurs a trade-off in terms of speed and resource utilization. Conversely, the YOLO [2-7] series and SSD [8] are better suited for real-time applications. Recent model enhancement strategies have facilitated small target detection in ambiguous settings. Wang et al. [9] proposed the UTD-Yolov5 approach, an enhanced YOLOv5 target detection algorithm. It emphasizes enhancing detection flexibility and accuracy by substituting the original Backbone with a two-phase cascaded CSP (CSP2) and incorporating a visual channel attention mechanism module, SE, among other modifications. Zhou et al. [10] introduced a method utilizing an enhanced YOLOv4, which includes the Multi-scale Retina Algorithm (MSRCR) for picture enhancement alongside an updated Spatial Pyramid Pooling (SPP) module. Ge et al. [11] introduced a lightweight model, UW_YOLOv3, to mitigate computational energy and storage resource limitations in intricate application contexts. Huo et al. [12] introduced an enhanced Ghost module developed by a feature reuse concept to augment the accuracy and robustness of biometric identification and detection in intricate circumstances. Resource identification has emerged as a significant aspect of tiny target detection in intricate circumstances, with numerous studies contributing to this field. Wu et al. [13] developed the YOLOv5-fish detection algorithm, which employs an autonomous MSRCR algorithm to enhance blurred images. This advancement accelerates and increases the accuracy of fish target identification in obscured environments by optimizing critical components of the YOLOv5 model. Liang et al. [14] introduced the C3 module, which employs depth-separable convolutions and Ghost convolutions, alongside the structurally parameterized RepVgg module, to enhance the model's detection accuracy and inference time. Yi et al. [15] introduced a compact target identification method utilizing YOLOv7, incorporating the SENet attention mechanism and improving the FPN network architecture. Liu et al. [16] introduced MarineYOLO, enhancing target localization accuracy and stability through the utilization of upgraded EC2f and EMA modules, together with the incorporation of CBAM and Wise-IoU loss functions. Liu et al. [17] introduced YoLoWaternet (YWnet), enhancing the detection accuracy of tiny targets with the use of CBAM, CRFPN, and SRC3 modules, alongside EIoU loss functions and decoupling heads. Zhang et al. [18] introduced a biometric method utilizing an enhanced lightweight YOLOv5, selecting EfficientNetV2-S as the lightweight backbone network, thereby decreasing the computational load of network parameters and enhancing recognition speed. Qu et al. [19] introduced the YOLOv8-LA model, an innovative neural network designed for small target detection. This model enhances performance through the implementation of the LEPC module and the AP-FasterNet architecture, along with the integration of the CARAFE upsampling operation, ensuring high detection accuracy while preserving real-time processing capabilities. These enhancement modules are essential for augmenting the precision of small target detection in ambiguous contexts and signify the swift advancement of fuzzy target detection. Contemporary research methodologies inadequately address the unique challenges presented by complex ecosystems. They either inadequately address these issues or render computations excessively costly in the pursuit of enhanced performance. Both of these challenges hinder the practical application of these technologies in real-world scenarios. Achieving a balance between detection accuracy and model complexity remains challenging; hence, small target detection in ambiguous circumstances continues to encounter difficulties in efficient application with constrained resources.

III. Methods

A. Network Model

YOLOv8 (You Only Look Once version 8) represents the most recent advancement in the YOLO series of object detection models developed by the Ultralytics team, maintaining the series' esteemed standards of efficiency, rapidity, and precision. The Ultralytics team classifies YOLOv8 into five model sizes: n, s, m, l, and x. This research uses YOLOv8s as the primary model. YOLOv8s is acknowledged for its exceptional detection precision, rapid computational efficiency, and diminished parameter quantity. It is recognized for its simple deployment on mobile devices and its ability to maintain a high level of detection accuracy. The Backbone component of YOLOv8 enhances feature extraction via a sophisticated amalgamation of three modules: Conv, C2f, and SPPF. The Conv module comprises multiple convolutional layers, batch normalization, and SiLU activation functions that improve model performance while maintaining computing efficiency. The C2f module utilizes skip-layer connections and split operations in a novel manner. This facilitates the network in attaining an optimal balance between depth and width, improving gradient propagation and expediting information transfer. The SPPF module utilizes a method called "spatial pyramid pooling" to facilitate the model's extraction of relevant features across several scales. This augments its ability to adjust to objectives across several scales. In the Neck section, YOLOv8 has improved feature fusion by removing redundant convolutional layers found in YOLOv5, leading to a more efficient and streamlined network. In the Head region, YOLOv8 has offered a notable innovation by employing a decoupled architecture that separates target classification from localization tasks, thereby resolving the conflicts present in conventional frameworks. This innovative adjustment improves the model's performance in classification and localization accuracy.

YOLOv8 has achieved extensive use in target detection owing to its remarkable versatility. Nonetheless, the identification of small objects in ambiguous environments continues to be hindered by issues such as intricate backdrops, inadequate illumination, and diminutive, indistinct targets, which increase the likelihood of missing and erroneous detections. This research proposes a novel DRL-YOLO network model (Fig. 1) that guarantees precise detection in particular environments while maintaining a compact and lightweight design.

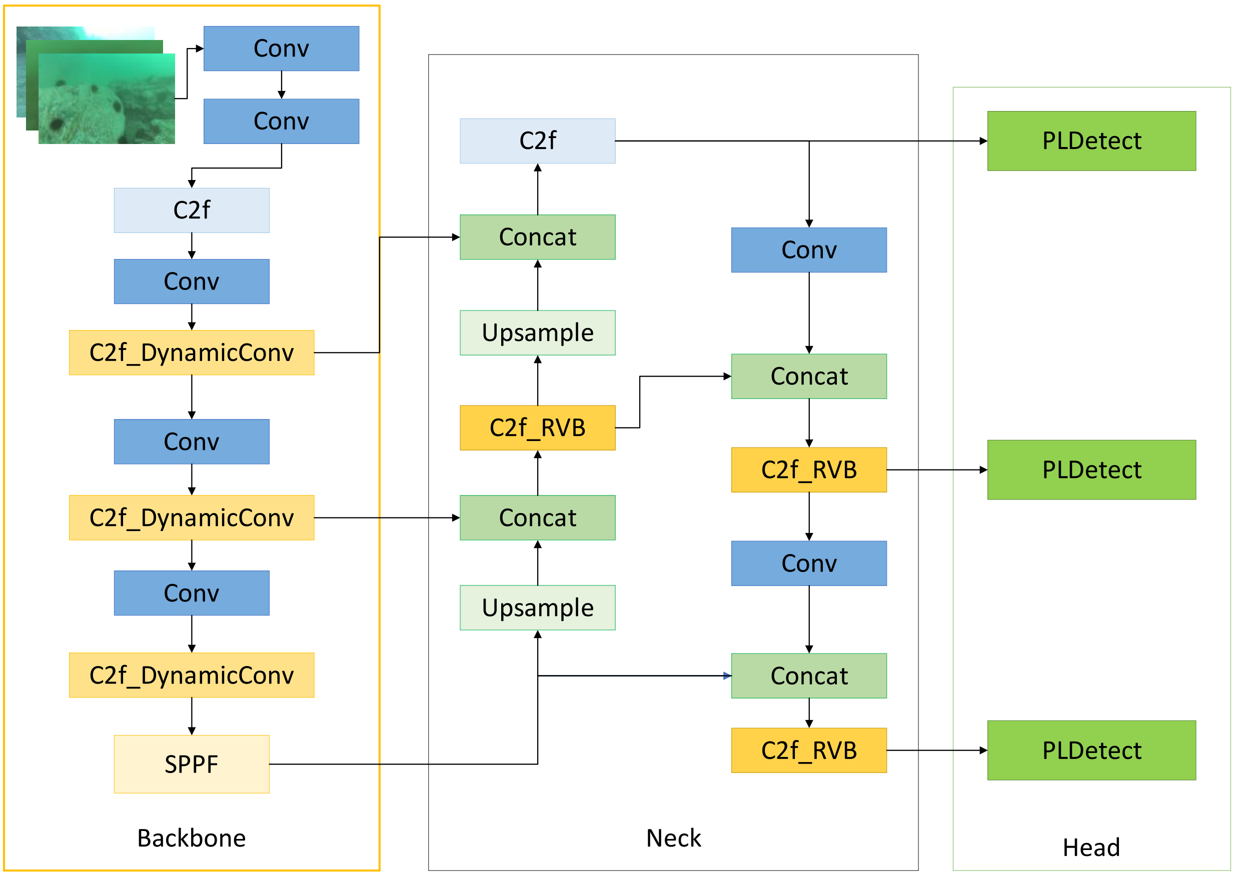

Fig. 1. A network structure diagram of DRL-YOLO.

DRL-YOLO utilizes various unique methodologies to enhance computing efficiency, detection precision, memory utilization, and model resilience. The optimization methodologies are as follows:

Employing a static convolutional kernel for input features in a conventional convolutional layer may lead to extended image processing durations, especially when the dimensions of the image and the number of features fluctuate. This paper introduces the C2f_DynamicConv module within the Backbone section as a resolution to this problem. It alters the dimensions and arrangement of the convolutional kernel based on differing input features. This improves feature extraction efficiency and accelerates the training and inference procedures.

The gradient may diminish or information may be compromised in YOLOv8's traditional convolutional network, especially during the deep feature extraction stage. This research introduces the C2f_RVB module, which enables the transfer of feature information between layers using residual connections, so effectively mitigating information loss and accelerating the training process. This improves the model's adaptability to diverse object sizes and complex environments.

This study proposes a novel PLDetect detection module to enhance detection efficiency and accuracy, employing a serial configuration of PConv and Conv. Pconv circumvents the computation of invalid regions by convolving solely the valid areas, thereby substantially decreasing the computational load, particularly in intricate landscapes with increased occlusion and absent components. The serial architecture additionally diminishes memory usage and enhances the model's performance.

B. Backbone Network Improvements

The C2f in the Backbone segment of the YOLOv8 model has a fixed-size convolutional kernel, potentially limiting its applicability to targets of varying shapes and sizes. For objects with intricate geometries or atypical proportions, a static convolutional kernel may prove ineffective in capturing their characteristics. DSConv (Dynamic Snake Convolution) enhances the capture of local characteristics in an image by dynamically altering the convolution kernel, which is crucial for target detection. This study replaces the usual convolution in the bottleneck section of C2f to improve adaptability for feature extraction in complicated situations and better target identification accuracy. Figure 2 depicts the configuration of the enhanced C2f_DynamicConv module.

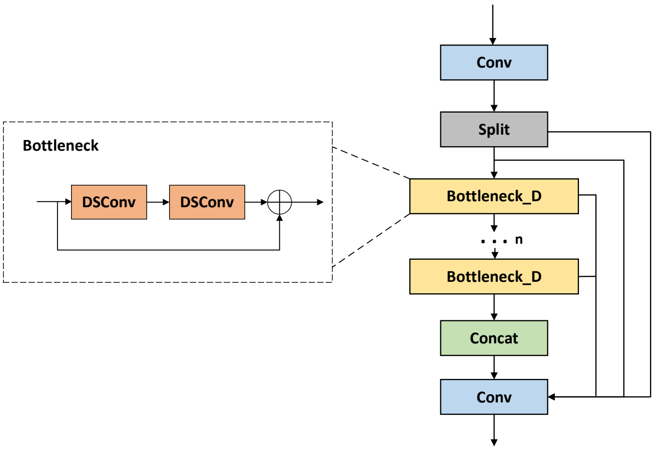

Fig. 2. The C2f_DynamicConv module.

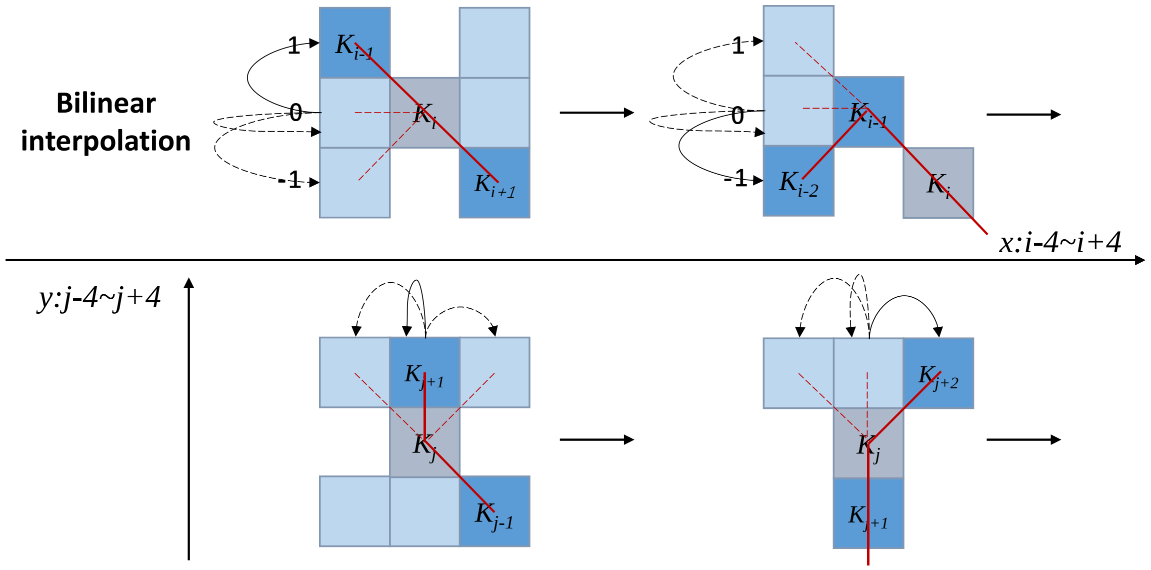

Fig. 3. Calculation process of DSConv coordinates.

The DSConv (Dynamic Snake Convolution) operation is a dynamic convolution operation designed to segment tubular structures that adapt to the target's geometry. It works best on thin, winding tubular structures like blood vessels and roads. DSConv can focus on features from different perspectives through the multi-perspective feature fusion strategy, ensuring that important information from different global patterns is retained. Although DSConv provides more complex feature extraction, it still maintains a high computational efficiency, which is necessary for real-time or near-real-time image processing applications, and satisfies the goal of fuzzy target detection to be deployed on limited resources. The calculation of the DSConv coordinates is shown in Figure 3.

This paper modifies the usual convolution kernel along the x-axis and y-axis to enhance DSConv (Dynamic Serpentine Convolution) for modeling tubular structures. A convolution kernel measuring 3x3 is utilized as an example to demonstrate the process: In the x-axis direction, each grid point of the convolution kernel is represented in this work as $K_{i\pm c}= (x_{i\pm c}, y_{i\pm c})$, where c might assume the values {0, 1, 2, 3, 4}, indicating the horizontal distance from the central grid. The selection of each grid place in the convolution kernel is a cumulative procedure. Commencing with the central location K_i, the subsequent position K_{i+1} of each grid is determined by the preceding position K_i augmented by an offset $\Delta=\{\delta \mid \delta \in [-1,1]\}$. To preserve a linear morphological structure in the convolution kernel, the offsets Σ must be aggregated to allow for dynamic adjustment of the kernel along the local structure of the target. The variation of the convolution kernel along the x-axis and y-axis is illustrated in equations (1) and (2), respectively.

$$K_{i\pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_i + c, y_i + \Sigma_i^{i+c}\Delta y), \\ (x_{i-c}, y_{i-c}) = (x_i - c, y_i + \Sigma_{i-c}^{i}\Delta y), \end{cases} \quad (1)$$

$$K_{j\pm c} = \begin{cases} (x_{j+c}, y_{j+c}) = (x_j + \Sigma_j^{j+c}\Delta x, y_j + c), \\ (x_{j-c}, y_{j-c}) = (x_j + \Sigma_{j-c}^{j}\Delta x, y_j - c), \end{cases} \quad (2)$$

The bilinear interpolation approach, as delineated in equation (3), is employed to transform the input feature map according to the new coordinate mapping, resulting in the deformed feature map. Finally, the DSConv convolutional layer analyses the distorted feature map.

$$K = \Sigma_{K'} B(K', K) \cdot K' \quad (3)$$

C. Neck Network Improvements

The neck architecture in the YOLOv8 model is crucial as it amalgamates feature maps at many sizes to improve target detection precision. The initial neck component of YOLOv8 insufficiently examines the intricate interrelationship among several feature layers in the realm of feature fusion. This suggests that the combined traits possess limited expressive potential. This may lead to a deterioration in the model's efficacy when tackling targets with complex backgrounds or multi-scale variability. Classic necks may result in increased computing costs when processing high-resolution feature maps, hence affecting the model's overall efficiency, especially on resource-constrained systems. The two challenges are particularly crucial in fuzzy target detection within resource limitations. This study seeks to improve the depth and quality of feature fusion while reducing computational resource consumption. This paper integrates the RVB (RepViT Block) with the original C2f module to improve the network's performance. The C2f_RVB module amalgamates traditional convolutional processes with the RepViT Block module, which is based on RepViT (To Look Back at Mobile CNN From A ViT Point Of View). The RepViT Block module aims to optimize network performance, decrease computing demands and parameter quantity, and boost model dependability by allowing C2f_RVB to adapt more proficiently to varying picture conditions and target discrepancies via multi-scale and multi-path feature processing. This method allows the model to provide strong feature extraction and fusion capabilities while being lightweight. Figure 4 depicts the structure of the improved C2f_RVB module.

The RepViT Block (Revisiting Mobile CNN From

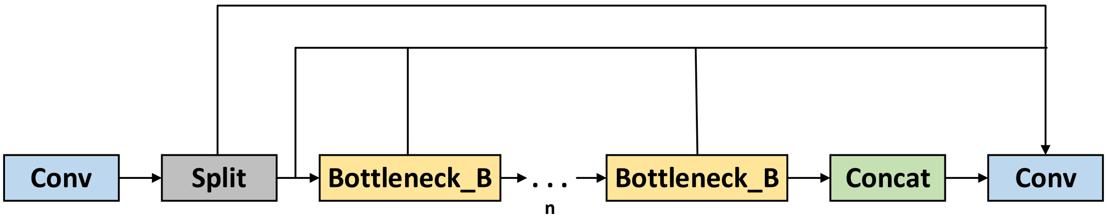

Fig. 4. The C2f_RVB module.

ViT Perspective Block) is an innovative network module that enhances the efficacy of conventional convolutional neural networks (CNNs) by integrating the advantages of Vision Transformer (ViT), which performs more effectively in resource-constrained environments, such as mobile devices. The purpose of the RepViT Block is to utilize the self-attention mechanism in ViT to improve the feature extraction efficacy of CNNs.The Vision Transformer (ViT) effectively captures global dependencies via the self-attention mechanism, whereas Convolutional Neural Networks (CNNs) specialize in local feature extraction. The RepViT Block integrates the advantages of both methodologies to enhance feature processing efficiency. In contrast to the conventional MobileNetV3 block that closely integrates spatial mixing and channel mixing, the RepViT Block implements multiple measures to delineate these processes. This work repositions the 3×3 deep convolution (DW) to precede the 1×1 extended convolution. Furthermore, the Squeeze-and-Excitation (SE) layer is repositioned after the 3x3 depthwise convolution, as it requires spatial information that has been previously processed by the 3x3 depthwise convolution. This modification enables the paper to distinctly differentiate the spatial mixer from the channel mixer within the MobileNetV3 block. This paper employs a conventional structural reparameterization technique to enhance the 3x3 DW layer. This facilitates the model's learning during the training process. This article utilizes the structural reparameterization technique to avoid the additional computational and memory burdens linked to skip connections during the inference phase, which is particularly crucial for resource-limited mobile devices. Figure 5 illustrates the configuration of the RepViT Block module.

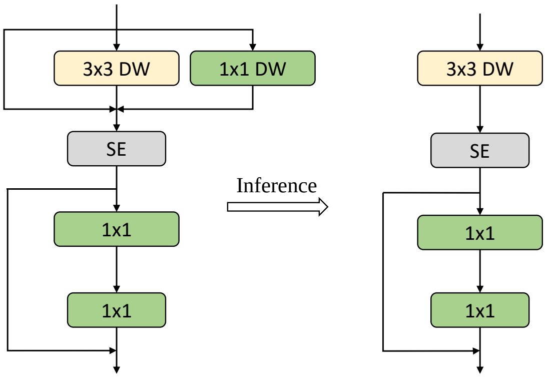

Fig. 5. RepViT Block

The self-attention mechanism in the RepViT Block significantly improves feature extraction capabilities. The self-attention technique allows the model to dynamically allocate different weights to features across several channels throughout the processing of the input feature map. This enables the recognition of salient characteristics. This mechanism is particularly proficient in fuzzy target recognition, as the complexity and dynamic unpredictability of certain circumstances require the model's adaptability in recognizing and responding to various visual cues. The C2f_RVB module enhances the model's ability to recognize multi-scale targets by effectively amalgamating feature maps from various scales. In the identification of small targets within obscured surroundings, where target size and shape display considerable variability, multi-scale feature fusion improves the model's capacity to accurately identify and localize these targets. The RepViT Block significantly enhances feature extraction by the integration of an advanced self-attention mechanism, multi-scale feature fusion, and deep separable convolution. The structure-intensive parameterization method and carefully designed network architecture improve lightweight efficiency while maintaining excellent performance, which is essential for resource-constrained fuzzy target recognition applications.

D. Head Network Improvements

The detection head of YOLOv8 predicts bounding boxes and category probabilities from the feature map. It comprises two branches, each employing two 3x3 convolutions and one 1x1 convolution to extract information from the input, subsequently computing Bbox.loss and Cls.loss. The detecting head constitutes approximately one-fifth of the model's computation. In conventional convolution, every channel of the input feature map must be processed. A large feature map dimension might result in substantial computational demands (FLOPs), prolonging inference time. Structurally, parallel convolutional layers augment computational complexity and memory consumption. In this study, we present the novel detection head PLDetect to address the computational burden while enhancing detection accuracy. Figure 6 illustrates the configuration of the PLDetect detection head:

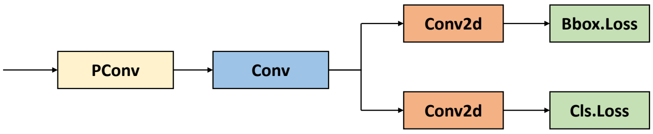

Fig. 6. PLDetect structure

In PLDetect, the initial 3x3 standard convolution layer is substituted with a PConv layer (Partial Convolution), which executes convolution operations on select channels of the input feature map while preserving the other channels, hence diminishing computational load and memory access.

Substituting the 3x3 convolution in the second layer with a 1x1 convolution diminishes computational load and parameter count, preserving the spatial dimensions of the feature map, hence rendering the model more efficient while sustaining detection efficacy. These modifications transform the detecting head from an original parallel configuration to a serial configuration. The serial architecture facilitates a more efficient integration of feature information across various scales through the sequential processing of features. This layer-by-layer feature fusion enables the model to discern the intricacies of multi-scale targets, hence enhancing detection accuracy. In contrast to the parallel structure, the serial structure necessitates fewer parameters and computations. The serial structure compresses and integrates characteristics at each stage, hence minimizing duplicate calculations. In deep neural networks, particularly in convolutional neural networks (CNNs), the feature map sometimes includes substantial redundant information. This redundancy is evident in the similarity of features across many channels. PConv is predicated on this finding and diminishes computational demands by executing convolutional operations on select channels while preserving the efficacy of feature extraction. The PConv operation can be characterized as a standard convolutional operation applied to a subset of the input feature map. PConv structure in Figure 7.

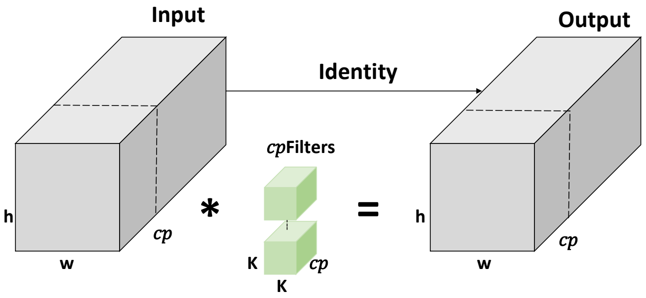

Fig. 7. PConv structure.

PConv processes only a subset of the c channels in an input feature map, rather than convolving all of them (e.g., c_p channels). This operation can be mathematically represented as follows:

$$\mathrm{PConv}(I) = \mathrm{Conv}(I_{cp}) \tag{4}$$

I_{cp} represents the c_p channels chosen from the input feature map I, whereas Conv signifies the standard convolution procedure. The PConv operation significantly reduces the number of floating point operations (FLOPs) by operating solely on a subset of the input feature map's channels. The formula for FLOPS in PConv is as follows:

$$h \times w \times k^2 \times c_p^2 \tag{5}$$

PConv is an innovative method for employing convolutional processes to enhance the speed and efficiency of neural networks. It accomplishes this by replicating feature maps to reduce unnecessary computations. This method decreases computational complexity, optimizes memory access patterns, and facilitates quicker inference. The new PLDetect model, enhanced by the incorporation of PConv and structural modifications, has notable lightweight properties, rendering it more appropriate for operation on resource-constrained devices to fulfill the requirements of real-time detection in ambiguous situations.

IV. Experiment

A. Experimental Environment and Configuration

This experiment establishes the necessary environment on a computer operating with the Ubuntu system, configured as detailed in Table I below:

TABLE I
Experimental environment

Parameters	Configuration
GPU	NVIDIA GeForce RTX 4060 Ti
GPU memory size	16GB
Operating systems	Unbuntu20.04
Python	Python 3.8.10
CUDA	12.6

In the training phase, the input image size is 640x640, the epoch is 100 rounds, and the batch size is 16.

B. Model Evaluation Metrics

This experiment employs a set of established evaluation metrics in target detection to assess model performance: precision (P), recall (R), mean average precision (mAP), floating point operations (GFLOPs), model parameters, and frame rate (FPS). These indicators collectively indicate the model's accuracy, efficiency, and resource utilization.GFLOPs (Giga Floating Point Operations Per Second) quantify the volume of floating-point operations a computing device can execute in one second and are frequently employed to assess the computational efficacy of a computer or processor. The formula for computing GFLOPs is as follows:

$$\mathrm{GFLOPs}=\frac{\mathrm{FLOPs}}{10^9} \tag{6}$$

The term FLOPs refers to the quantity of floating point operations executed per second. The mean Average Precision (mAP) is a widely utilized performance evaluation statistic in target detection, assessing the average precision across all target categories of the model. mAP is derived by computing the average precision (AP) for the predicted outcomes of each category and subsequently averaging the APs across all categories. To calculate mAP, generate the precision-recall (PR) curve for each category and determine the area beneath the curve. Consequently, mAP may thoroughly represent the model's detection efficacy across many categories. The formula for calculating mAP (Mean Average Precision):

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{AP}_i \tag{7}$$

N is the number of categories, and AP_i is the average precision of the ith category.The formula for AP is:

$$\mathrm{AP} = \sum_{i=1}^{n-1}(r_{i+1} - r_i)P_{\mathrm{inter}}(r_{i+1}) \tag{8}$$

Where r_1, r_2, ..., and r_n are the recall values corresponding to the first interpolation at the first interpolation of the precision interpolation segment in ascending order.

C. Introduction to The Dataset

In ambiguous situations, the undersea milieu serves as a quintessential special scenario, encompassing numerous characteristics that might depict the intricate environment. The DUO dataset [20] comprises data gathered from URPC contests, containing 7,782 precisely tagged photos, with 6,671 allocated for training and 1,111 for assessment. The photos in the DUO dataset exhibit common characteristics, including inconsistent lighting, blurriness, elevated noise levels, and other traits typical of indistinct images, which significantly illustrate the challenges encountered in small target detection within authentic fuzzy environments. Figure 8 displays representative images from the DUO dataset.

The collection comprises four groups of underwater organisms: holothurian, echinus, scallop, and starfish. Figure 9(a) illustrates the distribution of sample quantities, with echinus representing the highest proportion and scallops exhibiting a lower percentage. Figure 9(b) illustrates the spatial distribution of the items, indicating that the detected objects are predominantly clustered in the center of the image. Figure 9(c) depicts the allocation of sample sizes within the dataset. The scatter points suggest that the DUO dataset comprises a higher quantity of small samples and diminutive targets.

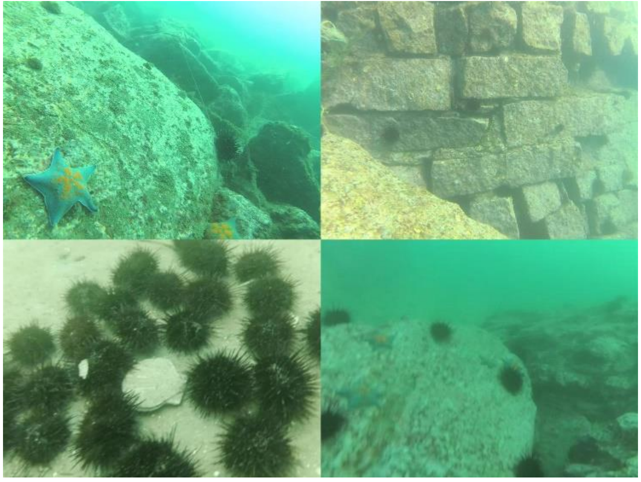

Fig. 8. DUO dataset.

The URPC2020 dataset comprises 5,543 photos extensively utilized in Chinese underwater robotics competitions, featuring four distinct marine organisms: holothurian, echinus, scallop, and starfish. Of these, 4434 are allocated for training, while 1109 are designated for testing. Figure 10 illustrates the exact distribution of the dataset.

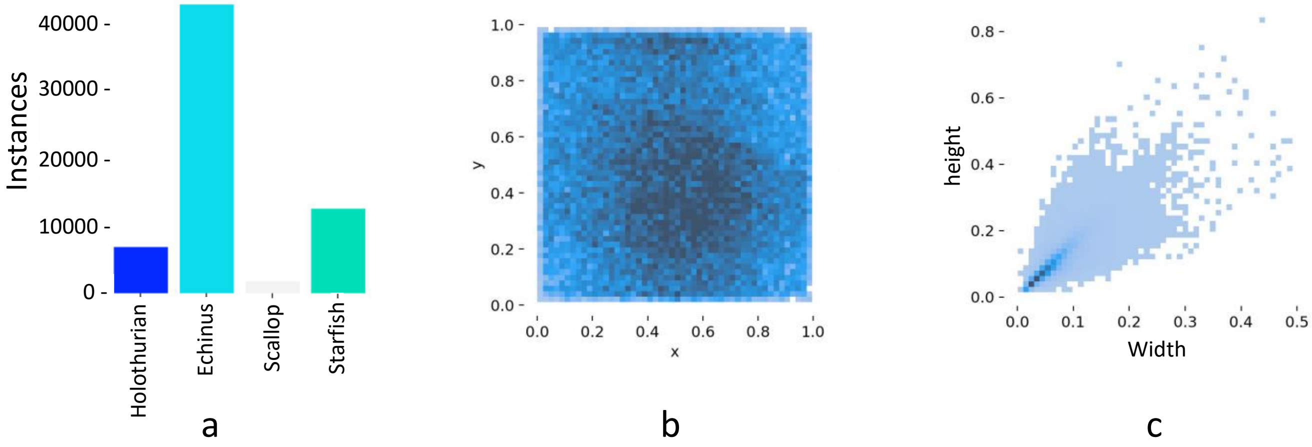

Fig. 9. Distribution of the DUO dataset, where a is the number of samples, b is the sample location, and c is the sample size.

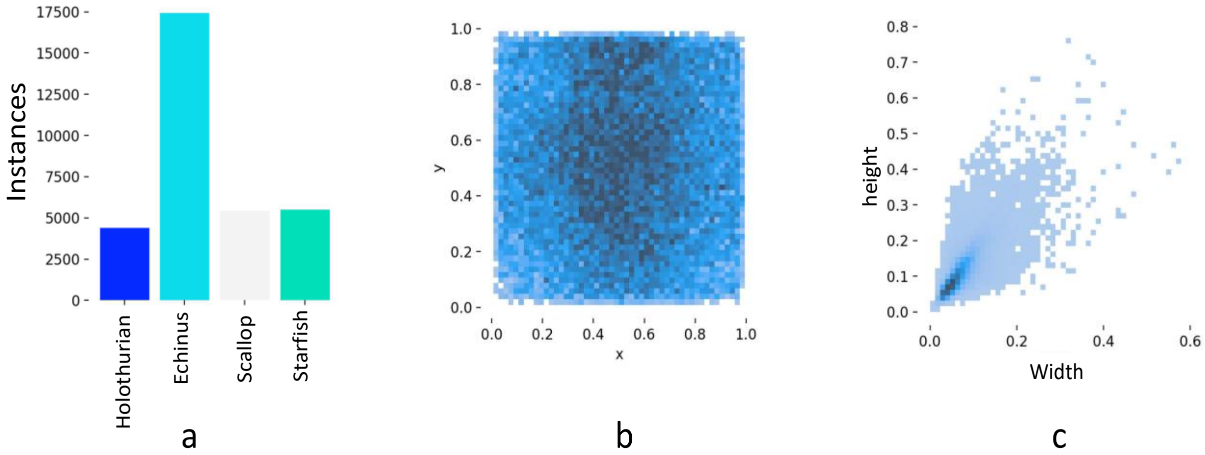

Fig. 10. Distribution of URPC2020 dataset where a is the number of samples, b is the sample location, and c is the sample size.

D. Ablation Experiment

This study uses YOLOv8s as a baseline model to evaluate the efficacy of the enhanced model. It accomplishes this by including several modules in the DUO dataset and evaluating the efficacy of the enhanced algorithmic model. All experiments in the ablation study commenced under identical settings, with the experimental environment detailed in Table I. The outcomes of the ablation experiments conducted on the DUO dataset are presented in Table II.

Experiment 1 functions as a benchmark for the original YOLOv8s model and offers a comparative reference for later experiments. Experiment 2 employs the C2f_DynamicConv method. The mAP50 has increased to 83.9%, the FPS has risen to 106.4, and the FLOPS/G has slightly decreased relative to the original model. This illustrates that the dynamic convolution technique can significantly enhance the model's detection accuracy and operational efficiency. Experiment 3 additionally presents C2f_RVB, resulting in a mAP50 increase of 84.3%, an FPS rise of 112.7, and a decrease in FLOPS/G to 23.9. This outcome underscores the beneficial impact of the RVB module in improving model performance, particularly in diminishing computational complexity. In Experiment 4, PLDetect was incorporated into the model, resulting in an enhancement of mAP50 to 84.7%, an increase in FPS to 115.4, and a reduction in FLOPS/G to 20.1. The PLDetect technology markedly enhances detection accuracy and efficiency. In Experiment 5, the integration of C2f_RVB and PLDetect yielded a mAP50 of 84.9%, an FPS of 102.9, and a FLOPS/G of 26.3. The experimental findings demonstrate that the integration of these two strategies can significantly improve the model's detection ability. In Experiment 6, the concurrent application of C2f_DynamicConv, C2f_RVB, and PLDetect yielded a mAP50 of 84.3%, an FPS of up to 120.7, and a FLOPS/G of 19.3. This arrangement attains maximal computing efficiency while preserving elevated detection accuracy. Experiment 7 and Experiment 8 were optimized according to Experiment 6, with Experiment 8 attaining the highest mAP50 of 84.8%, the lowest FLOPS/G of 17.0, and the highest FPS of 124.8. Experiment 8's design yielded the highest performance among all experiments, offering an effective solution for small target recognition in uncertain situations. Table III displays the ablation experiments on the URPC2020 dataset.

Through ablation trials, this study validates the effectiveness of the suggested augmentation approach. While maintaining a lightweight design, the use of methods like C2f_DynamicConv, C2f_RVB, and PLDetect significantly improves detection accuracy and operating efficiency.

TABLE II

RESULTS OF ABLATION EXPERIMENTS ON THE DUO DATASET

Experiment	C2f_DynamicConv	C2f_RVB	PLDetect	mAP50(%)	FLOPS/G	FPS
1	×	×	×	83.1	28.4	96.8
2	√	×	×	83.9	26.1	106.4
3	√	√	×	84.3	23.9	112.7
4	√	×	√	84.7	20.1	115.4
5	×	√	×	84.9	26.3	102.9
6	×	√	√	84.3	19.3	120.7
7	×	×	√	84.6	21.5	119.5
8	√	√	√	**84.8**	**17.0**	**124.8**

TABLE III

RESULTS OF ABLATION EXPERIMENTS ON THE URPC2020 DATASET

Experiment	C2f_DynamicConv	C2f_RVB	PLDetect	mAP50(%)	FLOPS/G	FPS
1	×	×	×	82.7	28.4	100.7
2	√	×	×	83.1	26.1	108.5
3	√	√	×	83.6	23.9	102.1
4	√	×	√	83.7	19.1	114.7
5	×	√	×	83.6	26.3	105.6
6	×	√	√	83.5	19.3	112.5
7	×	×	√	83.9	21.5	115.9
8	√	√	√	**84.0**	**17.0**	**121.6**

Table IV

THE EXPERIMENTAL RESULTS WERE COMPARED WITH MAINSTREAM MODELS ON THE DUO DATASET

Model	AP（%）				mAP50(%)	FLOPs (GFLOPs)
	Starfish	Scallop	Echinus	Holothurian		
Faster R-CNN[1]	78.9	48.3	77.9	69.7	68.7	210.3
SSD[8]	75.1	39.6	75.1	72.9	65.7	62.7
ResNet[21]	89.6	55.4	90.0	77.3	78.1	64.5
YOLOv5s	93.0	45.8	91.7	81.1	77.9	**15.8**
YOLOv7	91.0	54.7	90.2	79.3	80.1	105.2
YOLOvX-s	92.4	65.3	92.2	77.9	81.9	21.8
YOLOv8s	92.2	64.6	91.8	83.8	83.1	28.4
YOLOv10s	90.5	66.5	92.8	89.5	84.6	21.3
DRL-YOLO	92.8	66.4	92.6	87.3	**84.8**	**17.0**

TABLE V

EXPERIMENTAL RESULTS OF COMPARISON WITH MAINSTREAM MODELS ON THE URPC2020 DATASET

Model	AP（%）				mAP50(%)	FLOPs (GFLOPs)
	Starfish	Scallop	Echinus	Holothurian		
Faster R-CNN	81.4	75.1	87.9	67.5	78.0	210.3
SSD	-	-	-	-	76.2	62.7
DETR	-	-	-	-	60.3	188.7
YOLOv5s	-	-	-	-	79.5	**15.8**
YOLOv7	83.3	77.8	86.6	70.3	79.5	105.2
YOLOv8s	88.8	80.4	90.2	71.4	82.7	28.4
DRL-YOLO	**89.1**	**80.7**	**91.2**	**75.1**	**84.0**	**17.0**

E. Comparison Experiment

This article contrasts DRL-YOLO with the leading deep learning object identification models currently available, including Faster R-CNN, SSD, ResNet, YOLOv5s, YOLOv7, and YOLOv10s, utilizing the DUO dataset. The findings indicate that DRL-YOLO performs exceptionally effectively. The assessment metrics employed in this comparative experiment are AP (%), mAP50 (%), and FLOPs (GFLOPs) for each category. Table IV demonstrates that the DRL-YOLO model has superior detection capabilities while maintaining a lightweight design in comparison to other models. It is also more effective at utilizing limited resources for detection in complex environments. The mAP50(%) of DRL-YOLO surpasses that of all compared models, while its GFLOPs are significantly lower than those of most models, demonstrating superior detection accuracy and computing efficiency compared to the current YOLOv10s model.DRL-YOLO offers superior accuracy and average precision relative to alternative models, all while maintaining a cheap computational cost. This research employs uniform assessment criteria and experimental settings to thoroughly assess the generalisability of the models, conducting comparison tests on the URPC2020 dataset, with the results presented in Table V.

In the URPC2020 dataset, DRL-YOLO attained a mAP50 of 84.0 and a computational cost of 17.0 GFLOPs. Relative to the baseline, there was a 1.3% enhancement in the mAP50 and a 40.1% reduction in the FLOPs (GFLOPs). The experimental findings exhibited the robust generalization capability of DRL-YOLO.

To achieve a compromise between detection accuracy and processing efficiency, DRL-YOLO employs modules that preserve detection precision while being lightweight. This research contrasts the Backbone structure of DRL-YOLO by partially substituting C2f_DynamicConv with the Fusion Backbone, currently the predominant lightweight detection framework, with the experimental results presented in Table VI.

The experimental findings indicate that the enhanced backbone in DRL-YOLO is lightweight, however, the mAP score increases by 0.8% relative to the baseline. In comparison to previous models, it continues to exhibit good accuracy despite a reduction in computational complexity. This feature enables the DRL-YOLO model to adjust for real-time detection while enhancing detector quality.

TABLE VI

COMPARATIVE EXPERIMENTAL RESULTS OF REPLACING OTHER LIGHTWEIGHT BACKBONE WITH DRL-YOLO BACKBONE ON THE DUO DATASET

Backbone	mAP50(%)	FLOPs (GFLOPs)
Starnet[22]	80.8	17.3
Mobilenetv3[23]	79.0	16.3
Fasternet[24]	83.0	21.7
Efficientnet[25]	83.1	22.0
Improved Backbone	**83.9**	26.1

A. Results

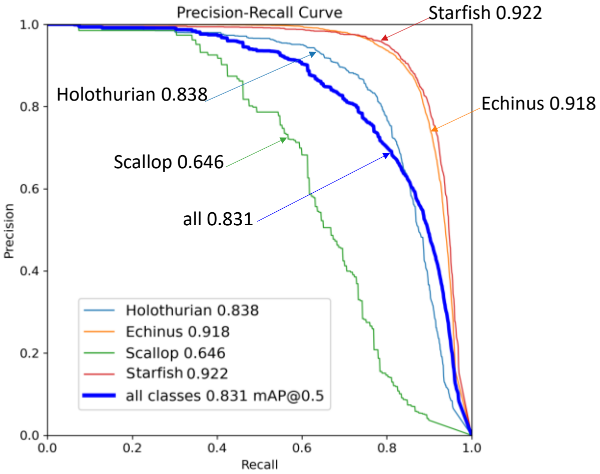

Fig. 11. shows the YOLOv8 PR plot on the DUO dataset.

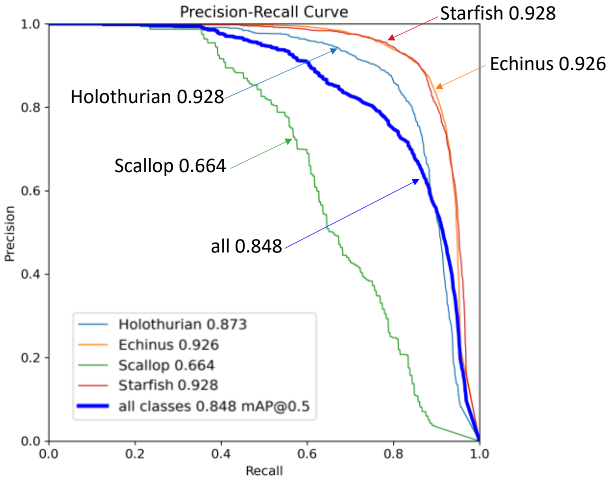

Fig. 12. A DRL-YOLO PR plot on the DUO dataset.

The PR plot indicates that the enhanced DRL-YOLO achieves a mean average precision (mAP50) score of 84.8%. This represents a significant increase from the baseline score of 83.1. The enhanced DRL-YOLO demonstrates superior accuracy compared to YOLOv8 across all categories, indicating that the optimization of the model architecture and the refinement of the training approach have a substantial impact. PR plots of the baseline versus the improved DRL-YOLO model are shown in Fig.11 and Fig.12.

V. CONCLUSIONS

This study presents DRL-YOLO, an innovative micro-target detection model for uncertain scenarios based on YOLOv8s. DRL-YOLO exhibits exceptional performance due to enhancements in the model architecture through the integration of deep convolution and dynamic convolution. It exceeds conventional detection models and current micro-target detection methods in terms of accuracy and computational complexity under uncertain conditions. It provides substantial advantages in feature performance, processing efficiency, and generalization capability. The test findings indicate that DRL-YOLO excels in detecting small objects within complex surroundings. It achieved mAP scores of 84.8% and 84.0% on the DUO and URPC2020 datasets, respectively. This indicates an enhancement of 1.7% and 1.3% relative to the baseline. An optimal balance of lightness and precision was achieved. GFLOPs dropped by 40.1% relative to the baseline. These changes render DRL-YOLO suitable for implementation in intricate contexts with constrained resources. DRL-YOLO's robust adaptability to intricate scenarios facilitates precise target recognition in real-time detection. The forthcoming research may investigate multimodal learning approaches to integrate many input sources, such as visuals and sounds, for the identification of small targets, given the complexities of real application contexts. DRL-YOLO is anticipated to enhance accuracy and robustness in outcome detection while augmenting the model's adaptability in intricate contexts. Considering DRL-YOLO's proficiency in achieving an ideal equilibrium between accuracy and computing complexity in intricate contexts, the small target detection methodology may be applicable across a broader spectrum of real-world scenarios in the future. It can be utilized in ambiguous conditions, such as inclement weather, to identify automobiles on dusty roads, among other applications. This will augment the critical function of small target-detection technologies in ambiguous environments for social and economic advancement.

REFERENCES

[1]. S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149. DOI:10.1109/tpami.2016.2577031.

[2]. Z. Ge, S. Liu and F. Wang et al., “YOLOX: Exceeding YOLO Series in 2021,” Computer Vision and Pattern Recognition (cs.CV), DOI:10.48550/arXiv.2107.08430.

[3]. J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” Computer Vision and Pattern Recognition (cs.CV), DOI:10.48550/arXiv.1804.02767.

[4]. A. Bochkovskiy, C. Wang and H. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” Computer Vision and Pattern Recognition (cs.CV), DOI: 10.48550/arXiv.2004.10934.

[5]. C. Li and H. Jiang et al., “YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications,” Computer Vision and Pattern Recognition (cs.CV), DOI: 10.48550/arXiv.2209.02976.

[6]. C. Wang, A. Bochkovskiy and H. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” Computer Vision and Pattern Recognition (cs.CV), DOI:10.48550/arXiv.2207.02696.

[7]. A. Wang, H. Chen and L. Liu et al., “YOLOv10: Real-Time End-to-End Object Detection,” Computer Vision and Pattern Recognition (cs.CV), DOI:10.48550/arXiv.2405.14458.

[8]. W. Liu, D. Anguelov et al., “SSD: Single Shot MultiBox Detector,” Computer Vision – ECCV 2016: Computer Vision–ECCV 2016, pp. 21-37. DOI:10.1007/978-3-319-46448-0_2.

[9]. J. Wang and N. Yu, “UTD-Yolov5: A Real-time Underwater Targets Detection Method based on Attention Improved YOLOv5,” Computer Vision and Pattern Recognition (cs.CV), DOI:10.48550/arXiv.2207.00837.

[10]. J. Zhou, Q. Yang, H. Meng, and D. Gao, “An underwater target recognition method based on improved YOLOv4 in complex marine environment,” Systems Science & Control Engineering, pp. 590-602. DOI:10.1080/21642583.2022.2082579.

[11]. H. Ge, Y. Dai, Z. Zhu, and R. Liu, “A Deep Learning Model Applied to Optical Image Target Detection and Recognition for the Identification of Underwater Biostructures,” Machines 2022, DOI:10.3390/machines10090809.

[12]. J. Huo and Q. Jiang, “IG-YOLOv5-based underwater biological recognition and detection for marine protection,” Open Geosciences, vol. 15, pp.20220590. DOI:10.1515/geo-2022-0590.

[13]. F. Wu, Z. Cai and S. Fan et al., “Fish Target Detection in Underwater Blurred Scenes Based on Improved YOLOv5,” IEEE Access, vol. 11, pp. 122911-122925. DOI: 10.1109/ACCESS.2023.3328940.

[14]. H. Liang and T. Song, “Lightweight marine biological target detection algorithm based on YOLOv5,” IEEE Access, DOI:10.3389/fmars.2023.1219155.

[15]. W. Yi and B. Wang, “Research on Underwater Small Target Detection Algorithm Based on Improved YOLOv7,” IEEE Access, vol. 11, pp. 66818-66827. DOI: 10.1109/ACCESS.2023.3290903.

[16]. L. Liu, C. Chu, C. Chen and S. Huang, “MarineYOLO: Innovative deep learning method for small target detection in underwater environments,” Alexandria Engineering Journal, vol. 104, pp. 423-433. DOI:10.1016/j.aej.2024.07.126.

[17]. P. Liu, W. Qian, and Y. Wang, “YWnet: A convolutional block attention-based fusion deep learning method for complex underwater small target detection,” Ecological Informatics, vol. 79, DOI:10.1016/j.ecoinf.2023.102401.

[18]. L. Zhang, J. Fan, and Y. Qiu, “Marine zoobenthos recognition algorithm based on improved lightweight YOLOv5,” Ecological Informatics, vol. 80, DOI:10.1016/j.ecoinf.2024.102467.

[19]. S. Qu, C. Cui, J. Duan, Y. Lu and Z. Pang, “Underwater small target detection under YOLOv8-LA model,” Scientific Reports, DOI:10.1038/s41598-024-66950-w.

[20]. C. Liu, H. Li, and S. Wang, “A Dataset and Benchmark of Underwater Object Detection for Robot Picking,” 2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), pp. 1-6. DOI:10.1109/ICMEW53276.2021.9455997.

[21]. S. Targ, D. Almeida, and K. Lyman, “Resnet in Resnet: Generalizing Residual Architectures,” Machine Learning (cs.LG), DOI:10.48550/arXiv.1603.08029.

[22]. X. Ma, X. Dai and Y. Bei et al., “Rewrite the Stars,” Computer Vision and Pattern Recognition (cs.CV), pp. 5694-5703. DOI:10.48550/arXiv.2403.19967.

[23]. A. Howard, M. Sandler and G. Chu et al., “Searching for MobileNetV3,” IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1314-1324. DOI: 10.1109/ICCV.2019.00140.

[24]. J. Chen, S. Kao and H. He et al., “Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks,” Computer Vision and Pattern Recognition (cs.CV), pp.12021-12031. DOI:10.48550/arXiv.2303.03667.

[25]. B. Koonce et al., “EfficientNet,” Convolutional neural networks with swift for Tensorflow: image recognition and dataset categorization, pp. 109-123. DOI:10.1007/978-1-4842-6168-2_10.