The Knowledge Engineering Review

RESEARCH ARTICLE

Open Access

https://doi.org/10.48130/ker-0026-0019

The Knowledge Engineering Review **2026**, 41: e018

UWD-YOLO for underwater small object detection

Jiebin Pi, Li Mo*, Hongping Yu*, Rong Yuan, Huangliang Yang, Shikai Li and Duo Wang

School of Mechanical Engineering, Chengdu University, Chengdu, Sichuan 610106, China

* Correspondence: moli@cdu.edu.cn (Mo L); hpyu354@cdu.edu.cn (Yu H)

Abstract

Traditional underwater detection methods struggle with low contrast and background interference, hindering accurate small object detection. To address these challenges, we propose UWD-YOLO, an improved YOLOv11-based model tailored for complex underwater environments. It integrates three key enhancements: a Spatial-Attentive Downsampling Module (SADA) to preserve spatial features in low-contrast regions, a Lightweight Shared Detection Head (LSP-Head) to improve multi-scale accuracy while reducing computational costs, and a Dynamic Modeling Enhancer (C3K2_DME) to filter severe background noise. Extensive experiments demonstrate UWD-YOLO's superiority. On a custom underwater dataset, it achieves 66.3% mAP@0.5. To rigorously validate cross-dataset generalization and statistical stability, we evaluated it on the public UTDAC2020 dataset, where it secured a state-of-the-art mean mAP@0.5 of 84.1% with an exceptionally low variance (±0.9%). Most notably, UWD-YOLO obtains an impressive 35.3% APS, significantly mitigating feature degradation for tiny marine targets. These results confirm UWD-YOLO as a highly stable, precise, and robust solution for underwater small object detection.

Keywords: Underwater object detection; Spatial attention downsampling; Lightweight shared detection head; Dynamic enhancement

Citation: Pi J, Mo L, Yu H, Yuan R, Yang H, et al. 2026. UWD-YOLO for underwater small object detection. *The Knowledge Engineering Review* 41: e018 https://doi.org/10.48130/ker-0026-0019

1 Introduction

With the growing attention to the development and sustainable utilization of marine resources, as well as the protection of marine ecosystems, underwater object detection has become a crucial component in intelligent marine perception systems. It plays a vital role in a variety of applications such as underwater robotics, seabed observation, fishery monitoring, and autonomous underwater exploration[1,2]. In particular, the accurate detection of small underwater objects such as fish, starfish, and sea cucumbers is of great significance for ecological monitoring and resource management[3]. However, due to their small size, weak features, and complex environments, detecting small underwater objects remains extremely challenging[4]. Underwater scenes are often affected by light attenuation, color distortion, blur, low contrast, and background interference, which significantly degrade image quality and make feature extraction more difficult[5,6]. These factors lead to high false detection rates and missed detections when traditional algorithms (such as SIFT and HOG) are applied[7,8].

In recent years, deep learning-based object detection methods have achieved remarkable success in natural scenes. Representative approaches such as the YOLO series, Faster R-CNN, and SSD have demonstrated outstanding performance in terms of accuracy and efficiency[9,10]. Among them, the YOLO family has been widely adopted for real-time detection tasks due to its fast inference speed and end-to-end optimization capability. Nevertheless, directly applying these mainstream methods to underwater small-object detection still faces several difficulties. The weak feature representation of small underwater targets, insufficient multi-scale feature fusion, and complex environmental interference severely limit their detection accuracy and robustness[11–13]. To alleviate these problems, many studies have explored image enhancement and color correction techniques to improve visibility, and attention-based modules such as FPN, EIoU, and AFF to enhance feature representation[14,15]. In addition, lightweight and efficient variants (e.g., YOLOv5s, YOLOv7-tiny) have been proposed to improve inference efficiency. Furthermore, some works have utilized Generative Adversarial Networks (GANs) and semi-supervised learning to augment data and improve model generalization[16–18]. However, these methods still face challenges in handling complex underwater imaging conditions and maintaining robustness in low-quality visual environments[19].

To explicitly address the aforementioned challenges, our research motivation stems directly from observing the specific failure cases of baseline models in complex underwater environments. As intuitively compared in Fig. 1, traditional detectors frequently suffer from severe background interference, often confusing background textures (such as rocks) with actual marine organisms, leading to obvious false positives (as seen in Fig. 1b). In contrast, our proposed model effectively suppresses these background distractions and achieves precise localization (Fig. 1c). Driven by these empirical observations, the architecture of our UWD-YOLO is explicitly designed with three targeted improvements to resolve these exact bottlenecks: the SADA module to mitigate spatial information loss, the LSP-Head to improve precise multi-scale perception, and the C3K2_DME to dynamically capture spatial dependencies and filter out background noise. These improvements provide a robust and highly accurate solution for underwater small-object detection.

The key contributions of this paper are as follows:

• We propose the Spatial-Attentive Downsampling with Aggregation (SADA) module, which combines group convolution, Softmax-weighted aggregation, and multi-scale attention mechanisms to mitigate spatial information loss caused by traditional downsampling techniques. This improves the model's ability to focus on target regions and enhances perceptual accuracy, particularly for small objects in low-contrast environments.

• We design a Lightweight Shared Precise Detection Head (LSP-Head), which leverages shared feature extraction and a decoupled branch structure, allowing classification and regression tasks to be modeled independently. This leads to more accurate small-object detection, significantly improving stability and precision while reducing computational complexity.

© The Author(s)

https://doi.org/10.48130/ker-0026-0019

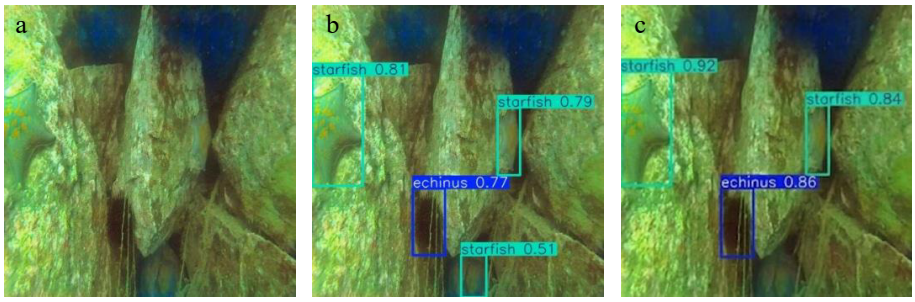


Fig. 1 Comparison of detection results in a complex underwater environment. (a) Raw image. (b) The baseline YOLOv11 generates a false positive due to background interference (misidentifying a rock as a starfish). (c) Our UWD-YOLO effectively suppresses the background noise and avoids the false positive.

• We propose a Dynamic Modeling Enhancer (C3K2_DME), which extends the original C3K2 module by incorporating multi-branch dynamic convolution, an Inception-style expression structure, and DropPath regularization. This module strengthens the model's spatial modeling capacity and enhances the depth of contextual feature representation for intermediate layers, further boosting the model's adaptability to complex underwater environments.

Extensive experiments on both a custom dataset and the public UTDAC2020 dataset thoroughly demonstrate the superior accuracy, efficiency, and robust cross-dataset generalization of our proposed approach.

2 Related work

2.1 YOLO-based lightweight detection

With the continuous development of computer vision, lightweight detection methods like the YOLO series have become the paradigm for tasks requiring high real-time performance and efficiency, such as agricultural and maritime surveillance[20,21]. As the latest iteration, YOLOv11 introduces several architectural improvements to balance computational efficiency and multi-scale feature representation[22,23]. It employs modules like C3K2 to reduce redundant computations[24] and integrates lightweight spatial attention (PSA) and dynamic convolution heads to focus on key regions while maintaining a low computational burden[25,26].

However, while these generic YOLO-based improvements and classic lightweight neural networks (e.g., SqueezeNet, ShuffleNet) achieve remarkable success in standard datasets through techniques like depthwise separable convolutions, their direct application to complex environments is limited. Specifically, to reduce computational costs, these traditional lightweight models often use simple pooling or strided convolutions during feature downsampling[27–29]. When deployed for underwater detection, these rudimentary operations result in a catastrophic loss of semantic information, rapidly degrading detection accuracy. Therefore, while this paper adopts YOLOv11 as the highly efficient baseline, substantial domain-specific redesigns are imperative to overcome its inherent bottlenecks in complex scenarios.

2.2 Underwater small object detection

Complex underwater environments, characterized by severe light attenuation, scattering blur, and low contrast, present significant challenges for object detection algorithms. Small object detection in such environments is particularly difficult due to the inherently limited resolution and weak semantic representation of tiny targets. Recent research has attempted to improve underwater detection by integrating attention mechanisms into the backbone or utilizing GAN-based image enhancement[30,31]. While these methods improve global feature aggregation under low-light conditions, they still fall short in preserving fine-grained spatial details during downsampling. Furthermore, recent advancements have explored diverse architectures to tackle specific underwater challenges. For instance, SU-YOLO[32] introduced a spiking neural network approach to achieve efficient underwater object detection, while YOLO-Starfish[33] specifically focused on learning complex underwater features to improve the detection accuracy of marine organisms.

Moreover, single-stage object detectors have continuously evolved their detection heads (e.g., employing decoupled structures or focal loss) to handle multi-scale outputs and class imbalance[34,35]. Despite these advancements, conventional detection heads still struggle with low-quality underwater images and low-resolution objects. They typically face a dilemma: relying on densely parameterized heads introduces massive computational redundancy, whereas conventional shared architectures often sacrifice multi-scale perception, leading to severe accuracy degradation for small objects.

2.3 Dynamic feature modeling and substantial innovations

To enhance modeling flexibility in complex scenarios, dynamic convolution structures (e.g., Dynamic Con, CondCon) have been proposed to adaptively adjust kernel weights based on input features[36–38]. Concurrently, multi-branch structures inspired by Inception[39] and hybrid modeling approaches like MLP-Mixer[40], alongside regularization techniques like DropPath[41,42], have been adopted to expand receptive fields and increase feature diversity. However, traditional static convolutions and even generic dynamic implementations struggle to achieve sufficient semantic modeling when facing dense small-object distributions and heavy underwater background interference.

While the aforementioned literature has extensively explored attention mechanisms, shared detection heads, and dynamic convolutions, these techniques are predominantly designed for general terrestrial scenarios. The substantial innovation of this UWD-YOLO lies in overcoming the fundamental representation learning limitations of generic vision models when subjected to severely degraded underwater visual fields, rather than superficially stacking existing engineering components. The underlying computer vision mechanisms behind our structural reformulations are explicitly defined as follows:

• Underlying mechanism in Spatial Downsampling (SADA): Standard spatial compression operations, such as strided convolutions or max-pooling, assume high visual contrast and inherently act as generic high-pass filters. In low-contrast underwater scenarios, this causes an irreversible collapse of spatial entropy for micro-scale boundaries. SADA fundamentally redefines this paradigm by coupling spatial attention mathematically into the downsampling transformation. By implementing a Softmax-normalized dynamic aggregation over grouped spatial pathways, SADA preserves critical topological invariants and structural target priors before they are lost during resolution reduction.

• Latent Space Alignment in Detection Heads (LSP-Head): Conventional shared detection heads blindly compress parameters, which induces severe gradient interference between the optimization landscapes of classification and regression tasks under high backscatter noise. LSP-Head breaks this trade-off by introducing a domain-tailored decoupled structure that separates task-specific distributions. Crucially, by integrating a learnable scale modulation factor into the bounding box distribution function, it explicitly regularizes scale variance within the latent space, enabling stable multi-scale localization even under severe boundary ambiguity.

• Dynamic Manifold Filtering for Background Noise (C3K2_DME): Generic dynamic convolutions adjust weights based on clean terrestrial statistics but fail when encountering dense, deceptive underwater background noise (e.g., textures of marine rocks). C3K2_DME establishes a dynamic feature manifold modeling framework. By embedding Inception-style multi-branch expressions with adaptive receptive fields and DropPath regularization, it dynamically modulates the nonlinear pathways of intermediate layers. This mathematical formulation enables the model to actively distort its latent representation space to segregate and suppress stationary environmental clutter while selectively amplifying the weak semantic signatures of camouflaged marine targets.

Consequently, UWD-YOLO shifts from a superficial engineering stacking framework to a holistically optimized, domain-specific architecture rooted in solving the underlying feature representation bottlenecks of underwater vision.

3 Methodology

3.1 UWD-YOLO structure

To address the issues of insufficient feature representation, inefficient feature fusion, and computational redundancy in underwater object detection, we propose an improved model called UWD-YOLO, based on YOLOv11. In this work, we replace the original C3K2 modules in YOLOv11 with our proposed C3K2_DME (dynamic modeling enhancer), which enhances the spatial modeling ability and contextual representation of intermediate features. Additionally, the strided convolution layers originally used for downsampling are replaced by our SADA (spatial-attentive downsampling with aggregation) module, which reduces information loss and improves the model's focus on target regions. For the detection head, we replace the standard decoupled detection heads in YOLOv11 with the proposed LSP-Head (lightweight shared precise detection head), which incorporates a shared pathway and scale adjustment factors to significantly improve detection accuracy for multi-scale small objects. Figure 2 illustrates the overall improved architecture, where the function of each module is as follows: the C3K2_DME module enhances the model's ability to perceive underwater images through multi-branch convolutions and dynamic feature adjustments; the SADA module preserves more crucial features through spatial attention and downsampling aggregation; and the LSP-Head increases detection accuracy for small objects while maintaining a lightweight design. The following sections will provide detailed descriptions of the design and optimization of each module.

3.2 Spatial-attentive downsampling with aggregation

In small-object detection tasks, the ability to extract spatial details is critical for accurate recognition. Conventional spatial compression methods, such as 2 × 2 strided convolution or max pooling, tend to discard essential information during the downsampling process. This issue is particularly pronounced in underwater environments characterized by low contrast and blurriness, where feature extraction becomes extremely challenging. To address the significant information loss caused by traditional max pooling or strided convolution in lightweight detection networks (the empirical evidence and visualizations of which are detailed in Section 4.4), this paper proposes a lightweight adaptive downsampling module that integrates a spatial attention mechanism—referred to as SADA. This module simultaneously reduces the spatial dimensions while encouraging the network to focus on more meaningful regional features, thereby enhancing the perception of small objects and improving overall detection accuracy.

The SADA module consists of two parallel branches: a feature downsampling path and an attention weight path. The feature downsampling path performs spatial compression using a 3 × 3 group

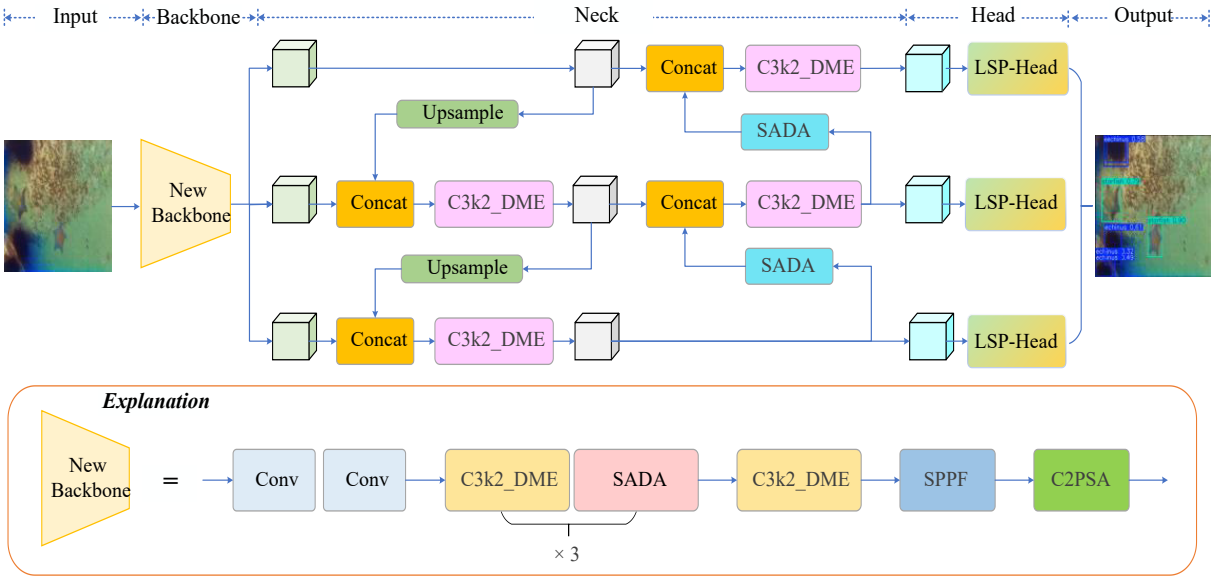


Fig. 2 UWD-YOLO structure. The diagram illustrates the network architecture of the UWD-YOLO model, which consists of three key components: the backbone, neck, and head. The backbone is responsible for extracting features, the neck fuses these features, and the head generates the final detection results.

https://doi.org/10.48130/ker-0026-0019

convolution with a stride of two, and expands the number of channels to four times the original, resulting in four weighted feature branches. The attention path first applies 3 × 3 average pooling on the input feature map to capture contextual information, followed by a 1 × 1 convolution to generate an attention map with the same number of channels as the input. This attention map is then reshaped to produce a weight distribution over the four channel branches at each spatial location and is normalized using a softmax function. The outputs of the two branches are multiplied element-wise along the fifth dimension (⊙) and then summed to complete the spatially weighted downsampling operation. This process can be formalized as follows in Eq. (1):

$$x_{out} = \sum_{i}^{4} (Conv_i(x) \odot Softmax_i(Conv_i(AvgPool(x)))) \quad (1)$$

Here, xout denotes the output feature, and Convi(x) represents the output of the i[th] downsampling branch. The Softmax function provides the corresponding spatial attention weights. The attention pathway is constructed using average pooling followed by a 1 × 1 convolution, which compresses the channel dimension and extracts global spatial information. The Softmax function is then applied to generate spatially normalized attention weights. The output shape is B × C × H/2 × W/2 × 4, where B is the batch size, C is the number of channels, and H and W denote the height and width of the feature map, respectively. The number 4 represents the spatial division factor, corresponding to a 2 × 2 partitioning of the feature map. The downsampling path uses a group convolution with stride = 2 to produce a feature map of shape B × 4C × H/2 × W/2, which is reshaped to match the attention output's dimensions. After fusion, the two outputs are multiplied element-wise and summed across branches to produce the final downsampled result, preserving richer semantic information. Compared to the Focus module in YOLOv5 or conventional strided convolution, the SADA module offers a more flexible and lightweight structure. It can adaptively focus on high-response regions, significantly enhancing small-object detection performance, especially in scenarios with limited image quality during downsampling. The structure of the SADA module is illustrated in Fig. 3.

3.3 Lightweight shared detection head

Traditional multiscale detection heads often suffer from redundant feature representations, high computational costs, and weak perception of small objects. To address these issues, we propose the LSP-Head. As illustrated in the overall architecture of Fig. 4, LSP-Head effectively enhances multiscale semantic representation and bounding box regression accuracy through a shared feature extraction pathway, decoupled subbranches, and a scale-aware modulation mechanism.

Initially, LSP-Head aligns the channel dimensions of multi-scale input features (P3, P4, P5) to C using a 1 × 1 convolution and Group Normalization (GN), denoted as the 1 × 1 Conv-GN blocks in Fig. 4. GN is specifically chosen to ensure training stability under small batch sizes. This process is formulated as Eq. (2):

$$P'_i = GN(Conv_{1\times1}(P_i)), i \in \{3,4,5\} \quad (2)$$

During the Training mode, these features are progressively upsampled and fused to extract a highly compressed, shared spatial feature map FS, which significantly mitigates computational redundancy. Specifically, P5 and P4 are upsampled and concatenated with P3, followed by two consecutive 3 × 3 convolutions (represented by the DeConv-GN 3 × 3 modules in the diagram), computed as Eq. (3):

$$F_s = Conv_{3\times3}\left(Conv_{3\times3}\left(Concat\left(P'_3, Up(P'_4), Up\left(P'_5\right)\right)\right)\right) \quad (3)$$

To prevent gradient interference between different tasks while maintaining parameter efficiency, FS is subsequently fed into functionally independent classification (Conv Cls) and regression (Conv Reg) subbranches. Each branch utilizes two consecutive 3 × 3 convolutions to learn its respective task-specific representation. The classification and regression outputs are defined in Eqs. (4) and (5), respectively:

$$x_i^{cls} = Conv2(Conv1(F_s)) \quad (4)$$

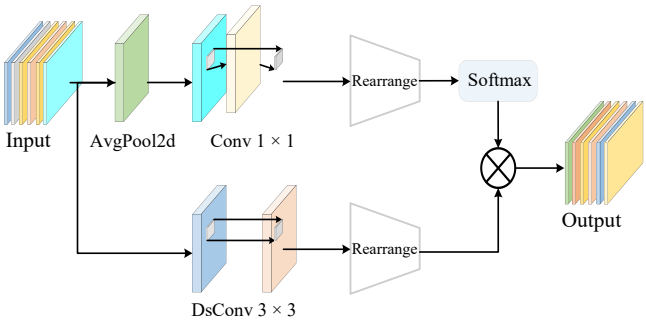


Fig. 3 The SADA downsampling module. It combines feature downsampling and attention-weighted paths to improve small-object detection by reducing information loss during downsampling.

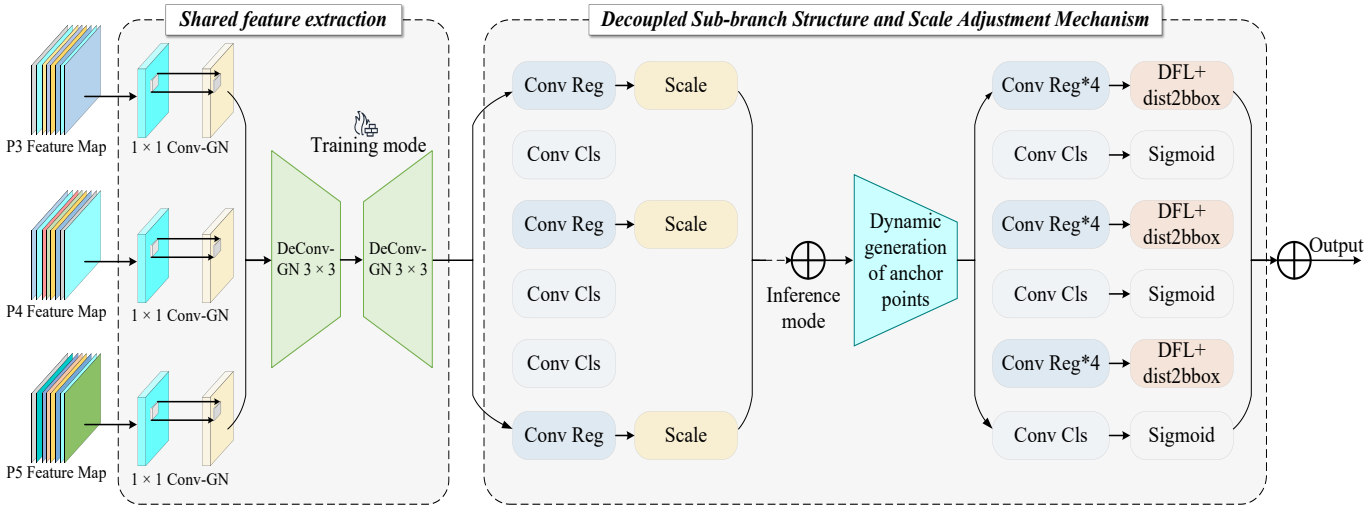


Fig. 4 The LSP-Head detection head module. It improves multi-scale detection by using shared feature extraction and decoupled sub-branch structures for classification and regression tasks.

$$x_i^{reg} = Conv2(Conv1(F_s)) \tag{5}$$

Crucially, to improve the spatial perception of multiscale targets—particularly small objects we introduce a learnable scale factor γi (highlighted by the Scale modules in Fig. 3) into the regression branch at each detection layer i ∈ {1,2,3}. This factor adaptively modulates the regression output to yield a scale-adjusted expected bounding box distribution, as shown in Eq. (6):

$$E(y_i) = \gamma_i \sum_{k=1}^{r} p_{ik} \cdot k \tag{6}$$

where, $E(y_i)$ denotes the expected value, p_{ik} is the predicted discrete probability distribution, and r is the number of discrete bins. This mechanism enables the model to effectively learn scale-sensitive feature responses.

Finally, during the Inference mode, a Dynamic generation of anchor points module adaptively proposes candidate boxes based on high-confidence regions to improve localization generalization. The classification output (processed via Sigmoid) and the regression output (processed by Distribution Focal Loss, DFL) are then concatenated (represented by the symbol) to form the complete detection Output $x^{out}{}_i$, as expressed in Eq. (7):

$$x_i^{out} = Concat\left(Sigmoid\left(x_i^{cls}\right),\ DFL\left(x_i^{reg}\right)\right) \tag{7}$$

This design achieves an optimal balance between modular decoupling and robust multiscale detection performance.

3.4 Dynamic enhancement module for intermediate feature modeling

In object detection tasks, the modeling capability of intermediate features by the backbone and neck modules directly affects the network's ability to represent spatial structures, semantic details, and contextual relationships. To improve the modeling depth, receptive field width, and channel interaction capacity of intermediate layers, we design an enhanced structure named C3K2_DME, based on the original C3K2 module in the YOLO series. This module aims to enhance intermediate feature representation and improve the network's adaptability to multi-scale object detection.

C3K2_DME retains the overall residual framework of the original C3K2 module, while replacing the original stacked convolution structure with multiple dynamic modeling submodules (Dynamic Modeling Blocks). Each submodule integrates dynamic convolutional branches, multi-scale receptive fields (Inception-style), lightweight MLP-based channel modeling, and DropPath regularization to enhance spatial awareness and nonlinear modeling capabilities. The overall structure is illustrated in Fig. 5.

Specifically, each sub-module in C3K2_DME takes the input feature $x \in \mathrm{R}^{B\times C\times H\times W}$ as the base and first applies normalization in Eq. (8):

$$x_1 = BN_1(x) \tag{8}$$

Then, the normalized features are fed into the Dynamic Inception Mixer module, which employs multiple parallel convolutional branches—such as 1 × 1, 3 × 3, and 5 × 5 convolutions—to extract features at different scales. These multi-scale features are then dynamically fused using weights adaptively generated from the input, as shown in Eq. (9), yielding the output feature z_1:

$$z_1 = DynamicInceptionMixer(x_1) \tag{9}$$

This output is adjusted using DropPath and a learnable scaling factor γ_1, and then added to the original residual input to obtain the first-stage output in Eq. (10):

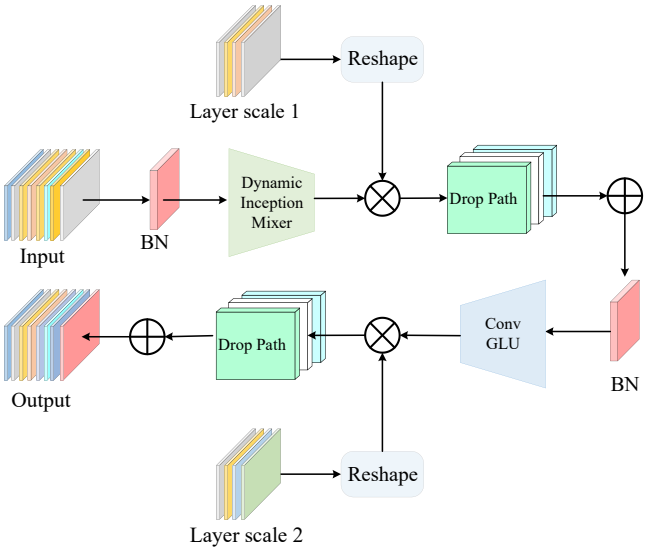


Fig. 5 Structure of the C3K2_DME module. It enhances the intermediate feature representation by integrating dynamic convolutions, multi-scale receptive fields, and lightweight MLPs. This module significantly improves spatial modeling and feature interaction capabilities, making it suitable for multi-scale object detection tasks.

$$y_1 = x + DropPath(\gamma_1 \cdot z_1) \tag{10}$$

Subsequently, y_1 is passed through a second normalization layer and then fed into a lightweight MLP branch, which consists of two consecutive 1 × 1 convolutions and GELU activation functions, as shown in Eq. (11), in order to enhance channel modeling capability:

$$z_2 = MLP(BN_2(y_1)) = W_2 GELU(W_1 \cdot y_1) \tag{11}$$

The final output of the module is Eq. (12):

$$y = y_1 + DropPath(\gamma_2 \cdot z_2) \tag{12}$$

Here, $\gamma_1, \gamma_2 \in \mathrm{R}^C$ are learnable scaling factors, which are initialized with small values at the beginning of training to enhance convergence stability.

C3K2_DME is an enhanced dynamic modeling structure proposed on top of the original C3K2 module, designed to replace the dynamic convolution components in the YOLO series. By integrating multi-branch dynamic convolutions (Inception-style), lightweight channel-wise MLPs, normalization, and residual mechanisms, it significantly improves the spatial modeling and semantic interaction capabilities of intermediate features, making it a suitable substitute for the original C3K2 convolutional blocks.

4 Experiments

To comprehensively evaluate the performance of the proposed improved YOLO model on underwater object detection tasks, we conducted a series of comparative experiments under a unified experimental environment. This section introduces the hardware platform, training parameters, optimizer, as well as the underwater image augmentation and preprocessing strategies used in the experiments. The detailed configuration is summarized in Table 1.

4.1 Dataset

This study utilizes a dataset primarily composed of images collected from publicly available online resources, marine science image repositories, research image libraries, and underwater photography

https://doi.org/10.48130/ker-0026-0019

Table 1. Experimental environment configuration.

Name	Configuration
Operating system	Ubuntu 22.04
GPU	NVIDIA RTX 3090-24GB
CPU	Intel(R) Xeon(R) Gold 6430
Programming language	Python 3.8.20
Framework	PyTorch 2.5.1
CUDA version	12.4
CUDNN version	8.2.1

archives. The dataset covers typical underwater scenes from both natural marine environments and artificial settings. To ensure data quality and diversity, the images underwent strict screening and preprocessing, with low-quality samples (such as duplicates, blurred images, and those with no identifiable targets) removed. The dataset was annotated frame-by-frame using the open-source labeling tool Labelme, with each target instance precisely labeled using rectangular bounding boxes, adhering to high accuracy and consistency principles.

The dataset contains five common underwater target objects: sea urchins, sea cucumbers, scallops, starfish, and aquatic plants. These objects are often characterized by small size, strong camouflage, blurred boundaries, and high sensitivity to background interference, making object detection more challenging. Representative raw images and examples of each species are illustrated in Fig. 6. The dataset consists of 7,600 images, with 73.2% of the targets being small objects (defined as those with a pixel area smaller than 32 × 32), emphasizing small object detection. The dataset is divided into training (5,320 images), validation (1,520 images), and test sets (760 images), with a 7:2:1 ratio, ensuring balanced category distribution and scene diversity, thus enhancing the model's generalization ability.

In order to test the generalization capability of our model, we also utilized the UTDAC2020 dataset. This dataset was provided as part of the Underwater Target Detection Algorithm Competition 2020 and contains 6,460 annotated underwater images. These images are categorized into four types of underwater objects: sea urchins, sea cucumbers, scallops, and starfish. The resolution of the images is 640 × 640 pixels, which maintains the original aspect ratio. The dataset is divided into two portions: 80% of the images are used for training, while the remaining 20% are designated for validation. The variety and complexity of the UTDAC2020 dataset make it an excellent resource for evaluating the model's performance in a wide range of underwater environments.

4.2 Evaluation metrics

In this study, we use several evaluation metrics to comprehensively assess the performance of the UWD-YOLO model. These metrics include Precision, Recall, F1 score, Average Precision (AP), and Mean Average Precision (mAP). Additionally, we evaluate the computational complexity of the model using Parameter Count (Param) and Floating Point Operations (FLOPs). The specific evaluation metrics and their formulas are outlined below:

Precision measures the proportion of true positive predictions out of all predictions labeled as positive (target objects). The formula is as follows in Eq. (13):

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

Recall measures the proportion of true positive predictions out of all actual positive samples. The formula is given in Eq. (14):

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

True Positive (TP) is the number of correctly predicted positive samples, while FP (False Positive) is the number of negative samples incorrectly predicted as positive, and False Negative (FN) is the number of positive samples incorrectly predicted as negative.

The F1 score is the harmonic mean of Precision and Recall, commonly used to assess the balance between the two metrics, especially when there is an imbalance between positive and negative classes. The formula is given in Eq. (15):

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{15}$$

Average precision (AP) is a key metric in object detection, reflecting the precision of the model at different recall levels. The formula is given in Eq. (16):

$$AP = \int_0^1 P(R)\,dR = \sum_{i=0}^{n-1} P_i \Delta R(i) = \sum_i \frac{P_i}{\sum_r \Delta R} \tag{16}$$

where, $P(R)$ represents precision at different recall levels, P_i is the discrete precision value, R is recall, and n is the number of discrete points.

In object detection tasks, the IoU (Intersection over Union) threshold plays a critical role in evaluating model performance. To assess the model's performance more accurately, several evaluation metrics are used, including AP, AP50, AP75, APS, APM, and APL. Among them, AP measures the average precision of the model across all targets; AP50 represents the average precision when the IoU threshold is 0.5; AP75 reflects precision at a stricter IoU threshold of 0.75; and APS,

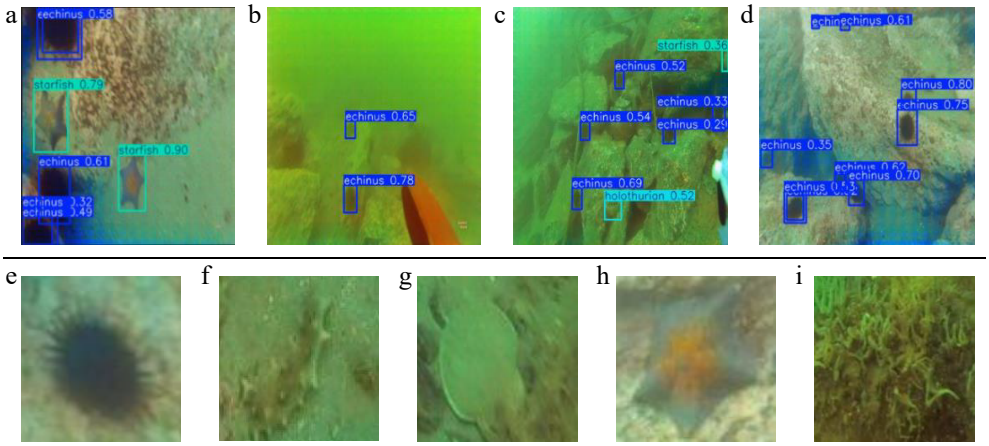


Fig. 6 Dataset and instance counts for each species. (a)–(d) Raw detection images; (e)–(i) examples of each species and corresponding instance counts.

APM, and APL correspond to the precision for small, medium, and large objects, respectively.

In addition, mean average precision (mAP) is a commonly used metric that averages the average precision (AP) values over different Intersection over Union (IoU) thresholds. The typical range of IoU thresholds spans from 0.5 to 0.95. mAP@0.5 refers to the average precision at an IoU threshold of 0.5, while mAP@0.5:0.95 calculates the average mAP across multiple IoU thresholds, ranging from 0.5 to 0.95 with a step size of 0.05. This range provides a more robust evaluation of the model's performance across various levels of overlap between predicted and ground truth boxes. The formula for mAP is given in Eq. (17):

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{17}$$

where, N is the total number of IoU thresholds (commonly ranging from 0.5 to 0.95), and AP_i is the average precision at each IoU threshold.

To explicitly evaluate the real-time processing capability for edge deployment, we also introduce frames per second (FPS) as a crucial speed metric. FPS measures the number of image frames the detection model can process per second during inference. A higher FPS indicates a faster reasoning speed, which is an essential prerequisite for ensuring smooth and real-time visual perception in practical underwater equipment.

Additionally, Param and FLOPs are used to assess the model's size and computational efficiency. Param represents the total number of trainable parameters in the model, reflecting its complexity and storage requirements. On the other hand, FLOPS indicates the number of floating-point operations the model performs per second during inference, providing insight into its computational burden and efficiency.

These evaluation metrics offer a comprehensive view of the model's performance across different scales, categories, and computational efficiency, enabling further optimization for practical deployment. In the experiments, we use these metrics to measure the detection accuracy, computational load, and overall performance under various conditions.

4.3 Comparative experiments

To comprehensively validate the effectiveness of UWD-YOLO in underwater environments, we first compared it with several conventional object detection models, including multi-stage detectors (such as Cascade-RCNN and Faster-RCNN) and single-stage detectors (such as GFL, TOOD, and ATSS_R50). Table 2 presents a comparison of UWD-YOLO with these traditional detection models across multiple evaluation metrics. As shown in the table, UWD-YOLO outperforms traditional models across multiple evaluation metrics. Compared to the multi-stage detectors (Cascade-RCNN and Faster-RCNN), UWD-YOLO achieves higher scores in AP (0.362), AP50 (0.66), and AP75 (0.378), and outperforms them in small object detection (APS), with a score of 0.353, surpassing Cascade-RCNN (0.292) and Faster-RCNN (0.305). These results demonstrate that UWD-YOLO excels not only in overall detection accuracy but also in small object detection in complex underwater environments.

To further validate our model, we compared it with YOLO series models, focusing on mAP@0.5 and mAP@0.5:0.95, as well as detection performance across various underwater object categories (such as Echinus, Holothurian, Scallop, Starfish, and WaterWeeds) on our self-built dataset. As shown in Table 3, UWD-YOLO outperforms all YOLO series models in both mAP@0.5 (0.663) and mAP@0.5:0.95 (0.365), significantly surpassing YOLOv8n (0.516 and 0.267), YOLOv9t (0.509 and 0.267), and YOLOv10n (0.503 and 0.265). UWD-YOLO also demonstrates superior detection accuracy across multiple underwater object categories, particularly in Echinus (0.866) and Starfish (0.775), exceeding the performance of YOLOv8n (0.809) and YOLOv9t (0.659). Specifically, we observe a dramatic performance jump in the extremely challenging 'WaterWeeds' category, where generic baselines (e.g., YOLOv10n) score nearly 0, whereas UWD-YOLO achieves an AP of 0.336. This phenomenon is not due to evaluation discrepancies, as all models were strictly trained and tested under identical settings. Instead, it stems from the inherent characteristics of water weeds: they possess highly irregular, non-rigid boundaries, severe camouflage, and typically suffer from sample imbalance (long-tail distribution). Consequently, standard models experience severe feature suppression, where the weak textural signatures of water weeds are entirely washed out during generic downsampling and overwhelmed by dominant classes. UWD-YOLO explicitly overcomes this failure mode through the SADA module, which preserves fragile topological textures during spatial compression, and the C3K2_DME module, which dynamically isolates these camouflaged signatures from background noise. This architectural synergy allows

Table 2. Comparison of UWD-YOLO with traditional object detection models.

Method	AP	AP50	AP75	APS	APM	APL	Param (M)	FLOPs (G)
cascade-rcnn[43]	0.314	0.58	0.307	0.292	0.384	0.42	69.213	162
faster-rcnn[44]	0.305	0.578	0.289	0.301	0.368	0.325	41.369	134
Gfl[35]	0.312	0.581	0.305	0.306	0.382	0.354	32.028	128
Tood[45]	0.316	0.585	0.311	0.315	0.384	0.364	32.268	123
atss_r50[46]	0.322	0.613	0.302	0.295	0.385	0.449	32.122	126
UWD-YOLO	**0.362**	**0.66**	**0.378**	**0.353**	**0.393**	**0.483**	**1.862**	**4.5**

Bold values represent the best performance for each evaluation metric.

Table 3. Comparison with state-of-the-art models on the custom dataset.

Method	mAP@0.5	mAP@0.5:0.95	APS	Echinus	Holothurian	Scallop	Starfish	WaterWeeds	Param (M)	FLOPs (G)	FPS
YOLOv8n[47]	0.516	0.267	0.317	0.809	0.448	0.384	0.653	0.012	3.006	8.1	**231**
YOLOv9t[48]	0.509	0.267	0.274	0.779	0.459	0.368	0.659	0.013	1.971	7.6	47
YOLOv10n[49]	0.503	0.265	0.314	0.799	0.417	0.382	0.641	0	2.266	6.5	192
YOLOv11n[50]	0.597	0.340	0.326	0.858	**0.638**	0.384	0.768	0.019	2.583	6.3	141
YOLO_GN[11]	0.632	0.354	0.327	0.836	0.615	0.523	0.762	0.107	2.278	5.2	76
Underwater-YCC[13]	0.627	0.351	0.293	0.799	0.579	0.448	0.732	0.009	**1.786**	4.6	238
UWD-YOLO	**0.663**	**0.365**	**0.353**	**0.866**	0.614	**0.723**	**0.775**	**0.336**	1.862	**4.5**	158

Bold values represent the best performance for each evaluation metric.

https://doi.org/10.48130/ker-0026-0019

UWD-YOLO to successfully recover the 'WaterWeeds' features from almost complete failure to a reliable detection state. Despite having significantly fewer parameters (1.862M) and lower computational cost (4.5G FLOPs) than the other YOLO models (e.g., YOLOv8n with 3.006M parameters and 8.1G FLOPs), UWD-YOLO still achieves superior detection accuracy and robustness.

This superiority is further supported by the Precision-Recall curve comparison with YOLOv11n (Fig. 7). UWD-YOLO's precision-recall curve significantly outperforms YOLOv11n (the officially recognized nano variant of the YOLOv11 series), demonstrating a better balance between precision and recall across multiple categories. Specifically, in the Echinus category, UWD-YOLO achieves a precision of 0.869, outperforming YOLOv11n's 0.858. Moreover, UWD-YOLO shows a remarkable improvement in detecting WaterWeeds, where YOLOv11n's curve is almost at the lowest level, whereas UWD-YOLO achieves a mAP@0.5 of 0.336, significantly surpassing YOLOv11n's 0.019. In summary, UWD-YOLO outperforms all YOLO series models in terms of both mAP@0.5 and detection accuracy across multiple categories, making it an excellent lightweight choice for real-world underwater object detection tasks.

To evaluate the effectiveness of the proposed algorithm visually in complex underwater environments, we provide a comparison of detection results using the original images, the YOLOv11n baseline model, and the proposed UWD-YOLO model under typical underwater scenarios, as illustrated in Fig. 8. As observed in the original image (Fig. 8a), underwater images often suffer from uneven lighting, low contrast, and occlusions, making it difficult to distinguish between the background and target objects. In the detection results of the baseline model YOLOv11n (Fig. 8b), some targets can be identified, but issues such as missed detections and inaccurate bounding boxes are common. In crowded areas or regions with complex backgrounds, the missed detection rate increases significantly. Compared to the baseline, the proposed UWD-YOLO model in Fig. 8c demonstrates significantly higher detection accuracy and robustness. It not only detects more small objects but also provides more precise localization, with bounding boxes that more closely align with the actual targets. Moreover, it effectively reduces false positives and missed detections in complex background regions and shows strong adaptability to multi-object scenarios, occlusions, and lighting variations. The visual results clearly indicate that UWD-YOLO achieves superior performance over the YOLOv11n baseline in challenging underwater environments, verifying its practical applicability.

To further validate the superiority of UWD-YOLO in specific underwater scenarios, we include recently proposed state-of-the-art models specifically designed for underwater object detection, namely YOLO_GN and Underwater-YCC, in our comparative analysis. Furthermore, given that the core focus of this paper is underwater small object detection, we explicitly incorporated the standard COCO evaluation metric for small objects (APS) into Table 3 to prevent small-object performance deficits from being masked by easier-to-detect large objects. As shown in Table 3, UWD-YOLO comprehensively outperforms these domain-specific models across all metrics. It achieves the highest overall mAP@0.5 of 0.663, surpassing both YOLO_GN (0.632) and Underwater-YCC (0.627). More importantly, UWD-YOLO demonstrates an overwhelming advantage in detecting small targets, achieving the highest APS score of 0.353, which significantly outperforms the baseline YOLOv11n (0.326) as well as the specialized models YOLO_GN (0.327) and Underwater-YCC (0.293). This advantage is particularly evident in detecting extremely challenging small objects with complex background interference, such as WaterWeeds, where UWD-YOLO achieves an AP of 0.336 compared to merely 0.107 for YOLO_GN and 0.009 for Underwater-YCC. Furthermore, UWD-YOLO accomplishes this superior detection performance while maintaining the lowest computational cost (4.5G FLOPs) among all compared specialized models. To explicitly address the critical trade-off between high-precision spatial modeling and real-time edge deployment, we also quantified the reasoning speed (Table 3). While the SADA and C3K2_DME modules enhance complex feature extraction, the proposed LSP-Head drastically cuts down multi-scale redundant parameters. Combined with its exceptionally

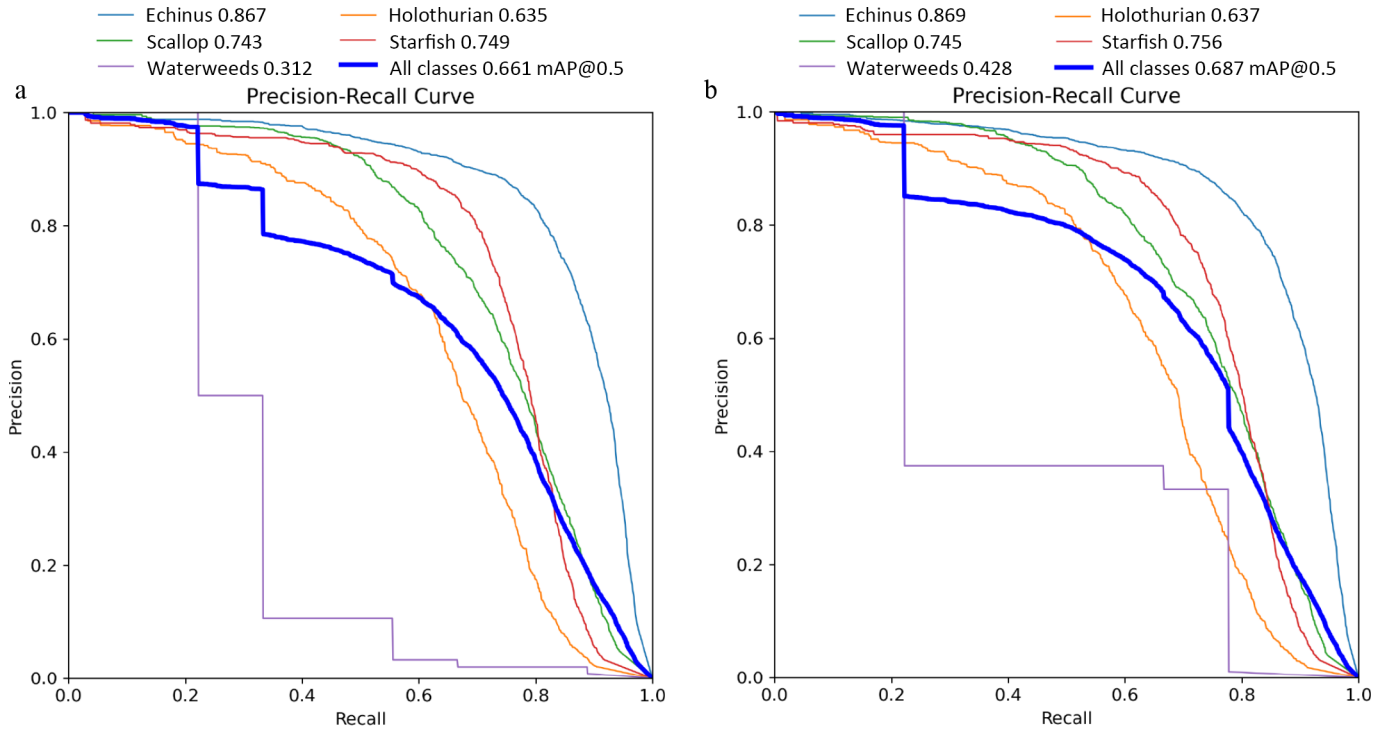


Fig. 7 Precision-Recall curve comparison between UWD-YOLO and YOLOv11 on the self-built dataset.

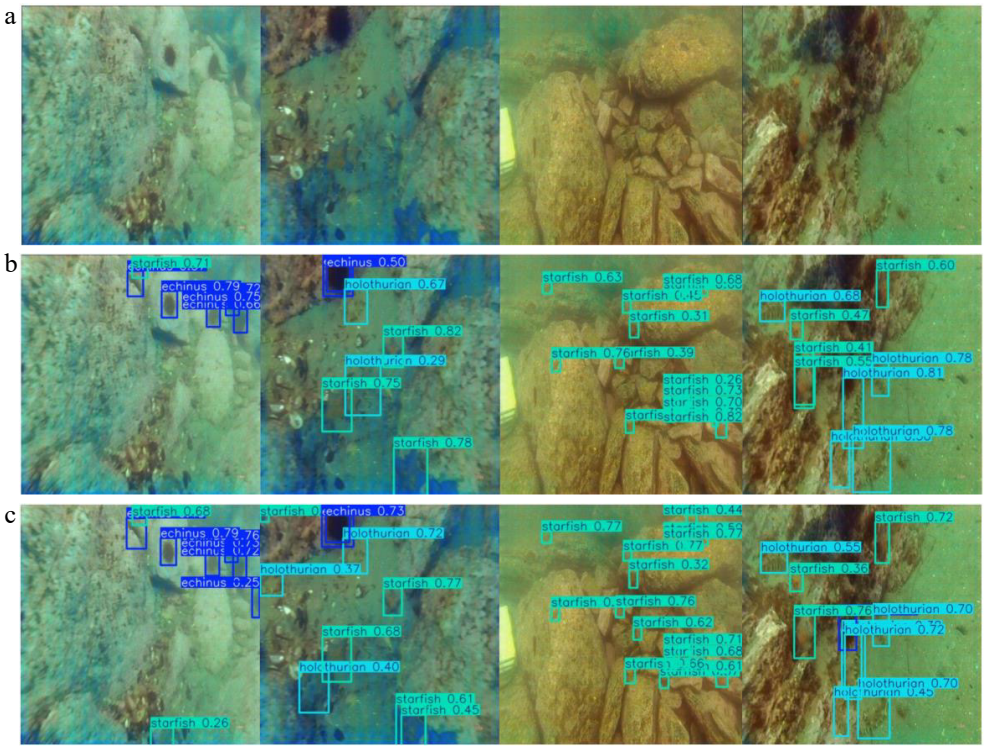


Fig. 8 Original images and detection results. (a) Original image. (b) Detection results obtained using the original YOLOv11n algorithm. (c) Detection results using the proposed algorithm.

low parameter count (1.862M), UWD-YOLO achieves a high inference speed of 158 FPS, easily surpassing the standard real-time threshold and even outperforming the YOLOv11n baseline (141 FPS). This robustly proves that UWD-YOLO not only serves as an efficient general detector but also stands as a highly competitive and lightweight SOTA solution specifically for complex underwater environments.

4.4 Ablation study

In this section, we conduct an ablation study to evaluate the contribution of each key module to the overall performance of UWD-YOLO. The modules tested include spatial-attentive downsampling (SADA), lightweight shared detection head (LSP-Head), and dynamic modeling enhancer (C3K2_DME). These modules are tested both independently and in various combinations to understand their individual and synergistic effects. The evaluation metrics include mean average precision (mAP@0.5, mAP@0.5:0.95), F1-score, model size (Params), and computational cost (FLOPs), as summarized in Table 4.

In the individual module experiments, the SADA module significantly improved mAP@0.5 from the baseline's 0.597 to 0.612, and mAP@0.5:0.95 from 0.34 to 0.349. Crucially, it elevated the small-object average precision (APS) from the baseline's 0.326 to 0.337. This quantitative improvement serves as direct empirical evidence that SADA effectively mitigates the spatial information loss typically incurred by traditional stride convolutions. It demonstrates that the spatial attention mechanism effectively helps the model retain crucial semantic features and focus on detecting small objects, particularly in complex underwater environments. Next, the LSP-Head, when used independently, achieved an mAP@0.5 of 0.609 and improved the APS to 0.330, improving detection stability while maintaining a relatively low computational cost (6.0G FLOPs). However, when C3K2_DME was used alone, the mAP@0.5 was 0.601, and the APS reached 0.332, indicating that while this module primarily enhances intermediate feature modeling, it has a limited direct improvement for small object detection when lacking robust feature inputs.

The dual-module combinations further reveal synergistic benefits, particularly for small objects. The SADA + LSP-Head combination achieved an mAP@0.5 of 0.630 and notably boosted the APS to 0.348, indicating that coupling spatial attention with the lightweight shared detection head significantly enhances small object detection and overall robustness in complex scenarios. The SADA + C3K2_DME combination resulted in an mAP@0.5 of 0.623 and an APS of 0.337, proving that integrating spatial attention with dynamic modeling effectively captures details and boosts performance. For the LSP-Head + C3K2_DME combination, the mAP@0.5 reached 0.616, and the APS reached 0.344, showing that this pairing improves detection accuracy and adaptability in cluttered, noisy backgrounds. Ultimately, the final model, integrating all three modules (SADA + LSP-Head + C3K2_DME), achieves the best detection performance with an mAP@0.5 of 0.663, mAP@0.5:0.95 of 0.365, an F1-score of 0.667, and an outstanding APS of 0.353. Despite these significant performance improvements, the final model maintains high computational efficiency, requiring only 1.862M parameters and 4.5G FLOPs. This systematically validates both the individual contributions and the synergistic advantages of the proposed architecture, especially for small object detection tasks.

The SADA + C3K2_DME combination resulted in an mAP@0.5 of 0.623 and mAP@0.5:0.95 of 0.343, proving that integrating spatial attention with dynamic modeling effectively captures details and boosts performance. For the LSP-Head + C3K2_DME combination, the mAP@0.5 reached 0.616 and mAP@0.5:0.95 reached 0.346, showing that this pairing improves detection accuracy and adaptability in cluttered, noisy backgrounds. Ultimately, the final model, integrating all three modules (SADA + LSP-Head + C3K2_DME), achieves the best detection performance with an mAP@0.5 of 0.663, mAP@0.5:0.95 of 0.365, and an F1-score of 0.667. Despite these

https://doi.org/10.48130/ker-0026-0019

Table 4. Ablation study on the effectiveness of each module on the dataset.

SADA	LSP-Head	C3K2_DME	mAP@0.5	mAP@0.5:0.95	APS	F1	Param (M)	FLOPs (G)
×	×	×	0.597	0.34	0.326	0.576	2.583	6.3
√	×	×	0.612	0.349	0.337	0.587	2.241	6.4
×	√	×	0.609	0.345	0.330	0.661	2.261	6
×	×	√	0.601	0.337	0.332	0.656	2.322	5.8
√	√	×	0.630	0.356	0.348	0.641	1.918	6.1
√	×	√	0.623	0.343	0.337	0.612	1.979	5.9
×	√	√	0.616	0.346	0.344	0.589	1.999	5.5
√	√	√	**0.663**	**0.365**	**0.353**	**0.667**	**1.862**	**4.5**

Bold values represent the best performance for each evaluation metric.

significant performance improvements, the final model maintains high computational efficiency, requiring only 1.862M parameters and 4.5G FLOPs. This systematically validates both the individual contributions and the synergistic advantages of the proposed architecture.

Beyond the quantitative metrics, to intuitively understand the internal mechanisms of these improvements under complex underwater conditions, we visualized the Grad-CAM heatmaps generated from different model configurations (Fig. 9). These configurations include the baseline model YOLOv11n (Base), the model with SADA added (Model A), the model with SADA + LSP-Head (Model B), and the full model with SADA + LSP-Head + C3K2_DME (Model C).

The general changes in the heatmaps clearly reveal that the Base model suffers from significant noise interference. Due to the information loss inherent in traditional downsampling methods, the baseline model fails to retain target-specific features, exhibiting large areas of high response in the background, with attention often mistakenly focused on non-target regions such as rocks, leading to failures in recognizing small targets like seaweed. After introducing the SADA module (Model A), the spatial information of small targets is much better preserved. The background noise is noticeably reduced, providing a clear visual contrast that illustrates how SADA prevents the loss of critical features during downsampling, although the model still lacks sufficient focus on the target regions, resulting in weak responses to actual objects and unstable detection. With the addition of the LSP-Head module (Model B), the response intensity in the target areas increases significantly; the model shows enhanced perception for marine organisms (e.g., seaweed, sea urchins, and starfish) while further suppressing background distractions. The final model (Model C) demonstrates the best activation performance, where high-response areas are accurately concentrated on the targets, and background interference is nearly eliminated.

Specific scenarios in Fig. 9 further illustrate these capabilities. Using the multi-object dense image (Fig. 9b) as an example, both the Base and Model A almost fail to detect the seaweed targets, showing no obvious activation near those regions. Model B begins to detect the seaweed, but the response intensity is low, and activation is concentrated on hard-to-detect areas. In contrast, the final Model C consistently detects the seaweed targets, with heatmaps clearly displaying high-response activation regions corresponding precisely to the seaweed. Similarly, for the background interference image (Fig. 9d), where rock regions produce excessively high responses, the Base, Model A, and Model B all suffer from severe false positives,

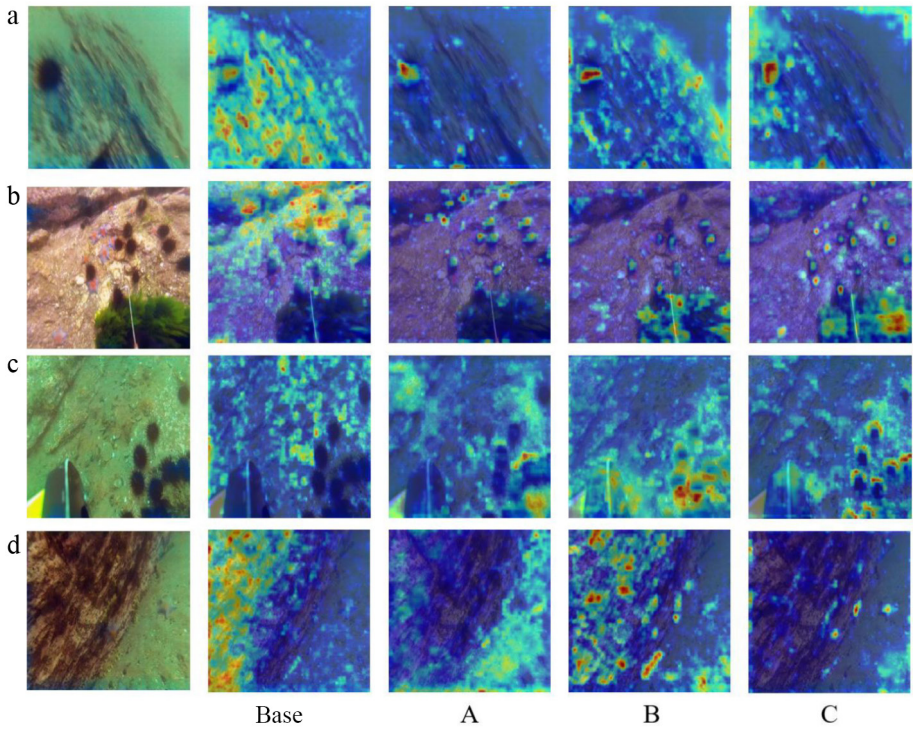


Fig. 9 Visualization of detection heatmaps from different models in typical underwater scenarios. Darker colors indicate better feature extraction in the corresponding image regions. (a) Blurred image; (b) images of different focal points; (c) images of the same target concentration; (d) background interference image. The columns represent different model configurations: 'base' is the YOLOv11n baseline; (A) is the model with SADA; (B) is the model with SADA + LSP-Head; and (C) is the full UWD-YOLO model (SADA + LSP-Head + C3K2_DME).

misidentifying rocks as target objects. However, Model C effectively suppresses this background rock interference and successfully identifies the true target (the starfish). Through this heatmap visualization analysis, the robust target focusing and noise suppression capabilities of UWD-YOLO are visually and definitively confirmed.

4.5 Evaluation of model generalization

To rigorously assess the generalization capability and statistical stability of the proposed UWD-YOLO, we conducted experiments on the public UTDAC2020 dataset using multiple random seeds. The experiments compared its performance with several other advanced object detection models. The evaluation metrics included mAP@0.5, mAP@0.5:0.95, and APS (average precision for small objects), with the main overall metrics reported as mean ± standard deviation to evaluate both absolute performance and robustness.

For the experimental setup, UWD-YOLO was trained for 200 epochs on our self-built dataset, with an additional 100 epochs in the generalization experiments on UTDAC2020 to further assess the model's ability to generalize across diverse underwater object categories. This extended training across different random seeds allowed us to comprehensively assess the model's ability to generalize across diverse and challenging underwater environments without succumbing to random initialization fluctuations.

As shown in Table 5, UWD-YOLO achieved the highest overall performance with a mean mAP@0.5 of 0.841 ± 0.009 and mAP@0.5:0.95 of 0.488 ± 0.016. More importantly, UWD-YOLO exhibited an exceptionally low variance (± 0.009) in mAP@0.5, which is significantly smaller than that of the baseline YOLOv11 (0.817 ± 0.045). This minimal fluctuation explicitly demonstrates that the substantial performance improvement of UWD-YOLO is highly statistically significant. Furthermore, by validating our specific structural designs for small targets, UWD-YOLO achieved the highest APS score of 0.353, prominently outperforming the baseline YOLOv11 (0.326) and effectively mitigating the severe feature degradation of tiny objects.

Specifically, UWD-YOLO demonstrated exceptional performance on the echinus and holothurian categories, with AP values of 0.526 and 0.554, respectively, indicating its strong capability to capture fine-grained features of highly camouflaged underwater organisms. In comparison, other advanced models such as RTMDet and NAS-FCOS achieved mean mAP@0.5 scores of 0.805 ± 0.018 and 0.823 ± 0.015, respectively. Although RTMDet showed competitive results in specific categories like scallop (0.470) and starfish (0.535), UWD-YOLO maintained the most optimal balance between overall high accuracy, extreme statistical stability, and superior small-object perception across diverse classes. Similarly, classic models like RetinaNet and SSD yielded lower overall performance (mAP@0.5 of 0.772 and 0.778, respectively) and exhibited higher instability. These results collectively highlight that UWD-YOLO is a highly robust and accurate architecture for underwater object detection.

4.6 Discussion on practical application

To demonstrate the practical utility of the proposed UWD-YOLO model in real-world marine environmental monitoring, we developed a proof-of-concept pipeline that integrates the visual detector with a large language model (LLM). Specifically, acting as a reliable visual perception front-end, UWD-YOLO extracts structured detection outputs (i.e., species categories and instance counts, e.g., {'Scallop': 12, 'Echinus': 9}) which are formatted into JSON and routed to the Qwen LLM via a REST API. Using standard prompt engineering workflows, the LLM translates these quantitative data into preliminary natural language ecological summaries to assist in evaluating benthic community density and habitat status. This downstream application showcases that UWD-YOLO can be seamlessly integrated into automated marine data analysis pipelines, effectively reducing the manual statistical workload for researchers.

5 Conclusions

This study proposes a novel underwater object detection model, UWD-YOLO, based on an improved YOLOv11 framework. By integrating the SADA, LSP-Head, and C3K2_DME modules, UWD-YOLO excels in small object detection and adaptability in complex scenarios. Experimental results show that UWD-YOLO achieved an mAP@0.5 of 66.3% on the custom multi-class dataset. Furthermore, extensive evaluations on the public UTDAC2020 dataset show that the model achieves an impressive mean mAP@0.5 of 0.841 with an exceptionally low variance of ± 0.009 across multiple random seeds. This confirms its remarkable statistical stability and robustness against severe underwater background interference. Notably, UWD-YOLO obtains an APS of 0.353, validating its superior capability in extracting features from highly degraded, tiny marine targets. Although detecting highly camouflaged organisms like scallops and starfish remains challenging, the model demonstrates strong competitiveness, high accuracy, and structural robustness, making it highly suitable for reliable underwater monitoring tasks. Future research will focus on further optimizing the model's detection capability for extremely dense small objects and exploring adaptive feature-extraction strategies for real-time underwater inference.

Author contributions

The authors confirm their contributions to the paper as follows: conceptualization: Pi J, Mo L, Yu H; methodology and project administration: Pi J, Mo L; software: Pi J, Yang H; validation: Pi J, Yu H, Mo L; formal analysis: Pi J, Yang H; investigation: Yu H, Yuan R; resources: Pi J, Yu H, Mo L; data curation: Yu H, Mo L, Yuan R; writing – original draft preparation: Pi J, Mo L, Yu H; visualization: Pi J, Li S, Wang D; supervision: Yu H, Mo L, Yuan R; funding acquisition: Yu H, Mo L, Yuan R. All authors reviewed the results and approved the final version of the manuscript.

Table 5. Comparison of UWD-YOLO with traditional object detection models.

Model	mAP@0.5	mAP@0.5:0.95	APS	Echinus	Holothurian	Scallop	Starfish
RTMDet[51]	0.805 ± 0.018	0.464 ± 0.022	0.322	0.458	0.385	**0.470**	**0.535**
NAS-FCOS[52]	0.823 ± 0.015	0.427 ± 0.011	0.347	0.431	0.331	0.466	0.483
RetinaNet	0.772 ± 0.022	0.415 ± 0.032	0.294	0.446	0.327	0.387	0.495
SSD	0.778 ± 0.018	0.401 ± 0.016	0.291	0.419	0.308	0.412	0.462
YOLOv11	0.817 ± 0.045	0.473 ± 0.052	0.326	0.525	0.548	0.360	0.508
UWD-YOLO	**0.841 ± 0.009**	**0.488 ± 0.016**	**0.353**	**0.526**	**0.554**	0.380	0.513

Bold values represent the best performance for each evaluation metric.

https://doi.org/10.48130/ker-0026-0019

Data availability

All key experimental methods and data analysis procedures are described in detail within the manuscript. The project's code and model implementation are publicly available at https://github.com/Art-Jiebin/UWD-YOLO11. Further data or materials are available from the corresponding author upon reasonable academic request.

Acknowledgments

This work was supported partly by the National Natural Science Foundation of China under Grant No. 52175130 and the Sichuan Provincial Outstanding Young Scientist Foundation under Grant No. 2024NSFJQ0033.

Conflict of interest

The authors declare that they have no conflict of interest.

Dates

Received 8 April 2026; Revised 18 July 2026; Accepted 16 September 2026; Published online 2 November 2026

References

[1] Dinakaran R, Zhang L, Li CT, Bouridane A, Jiang R. 2022. Robust and fair undersea target detection with automated underwater vehicles for biodiversity data collection. *Remote Sensing* 14:3680

[2] Hu Z, Cheng L, Yu S, Xu P, Zhang P, et al. 2025. Underwater target detection with high accuracy and speed based on YOLOv10. *Journal of Marine Science and Engineering* 13:135

[3] Er MJ, Chen J, Zhang Y, Gao W. 2023. Research challenges, recent advances, and popular datasets in deep learning-based underwater marine object detection: a review. *Sensors* 23:1990

[4] Sudhakar M, Meena MJ. 2021. An efficient interactive segmentation algorithm using color correction for underwater images. *Wireless Networks* 27:5435−5446

[5] Li Y, Xie P, Tang Z, Jiang T, Qi P. 2019. SVM-based sea-surface small target detection: a false-alarm-rate-controllable approach. *IEEE Geoscience and Remote Sensing Letters* 16:1225−1229

[6] Song P, Li P, Dai L, Wang T, Chen Z. 2022. Boosting R-CNN: reweighting R-CNN samples by RPN's error for underwater object detection. *arXiv* Preprint:2206.13728

[7] Li CY, Guo JC, Cong RM, Pang YW, Wang B. 2016. Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. *IEEE Transactions on Image Processing* 25:5664−5677

[8] Wu D, Luo L. 2024. SVGS-DSGAT: an IoT-enabled innovation in underwater robotic object detection technology. *Alexandria Engineering Journal* 108:694−705

[9] Hu K, Lu F, Lu M, Deng Z, Liu Y. 2020. A marine object detection algorithm based on SSD and feature enhancement. *Complexity* 2020:5476142

[10] Zeng L, Sun B, Zhu D. 2021. Underwater target detection based on Faster R-CNN and adversarial occlusion network. *Engineering Applications of Artificial Intelligence* 100:104190

[11] Chen X, Fan C, Shi J, Wang H, Yao H. 2024. Underwater target detection and embedded deployment based on lightweight YOLO_GN. *The Journal of Supercomputing* 80:14057−14084

[12] Chen X, Yuan M, Yang Q, Yao H, Wang H. 2023. Underwater-YCC: underwater target detection optimization algorithm based on YOLOv7. *Journal of Marine Science and Engineering* 11:995

[13] Zhang M, Xu S, Song W, He Q, Wei Q. 2021. Lightweight underwater object detection based on YOLO v4 and multi-scale attentional feature fusion. *Remote Sensing* 13:4706

[14] Bhalla S, Kumar A, Kushwaha R. 2024. Feature-adaptive FPN with multiscale context integration for underwater object detection. *Earth Science Informatics* 17:5923–5939

[15] Islam MJ, Xia Y, Sattar J. 2020. Fast underwater image enhancement for improved visual perception. *IEEE Robotics and Automation Letters* 5:3227−3234

[16] Hou C, Guan Z, Guo Z, Zhou S, Lin M. 2023. An improved YOLOv5s-based scheme for target detection in a complex underwater environment. *Journal of Marine Science and Engineering* 11:1041

[17] Liu M, Jiang W, Hou M, Qi Z, Li R, et al. 2023. A deep learning approach for object detection of rockfish in challenging underwater environments. *Frontiers in Marine Science* 10:1242041

[18] Yang H, Zhou T, Jiang H, Yu X, Xu S. 2024. A lightweight underwater target detection network for forward-looking sonar images. *IEEE Transactions on Instrumentation and Measurement* 73:2525113

[19] Ashraf H, Jeong Y, Lee CH. 2021. Underwater ambient-noise removing GAN based on magnitude and phase spectra. *IEEE Access* 9:24513−24530

[20] Wang CY, Bochkovskiy A, Liao HYM. 2023. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June, 2023*. USA: IEEE. pp. 7464–7475 doi: 10.1109/CVPR52729.2023.00721

[21] Zhao ZQ, Zheng P, Xu ST, Wu X. 2019. Object detection with deep learning: a review. *IEEE Transactions on Neural Networks and Learning Systems* 30:3212−3232

[22] Jocher G, Qiu J, Chaurasia A. 2025. *Ultralytics YOLO11*. https://github.com/ultralytics/ultralytics

[23] Yang X, Sufril Azlan Mohamed A, Wang C. 2025. ShadowFPN-YOLO: a real-time NMS-free detector for remote sensing ship detection. *IEEE Access* 13:55801−55814

[24] Zou C, Yu S, Yu Y, Gu H, Xu X. 2025. Side-scan sonar small objects detection based on improved YOLOv11. *Journal of Marine Science and Engineering* 13:162

[25] Chen J, Er MJ. 2024. Dynamic YOLO for small underwater object detection. *Artificial Intelligence Review* 57:165

[26] Han Q, Fan Z, Dai Q, Sun L, Cheng MM, et al. 2021. On the connection between local attention and dynamic depth-wise convolution. *arXiv* Preprint:2106.04263

[27] Han K, Wang Y, Tian Q, Guo J, Xu C, et al. 2019. GhostNet: more features from cheap operations. *arXiv* Preprint:1911.11907

[28] Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, et al. 2016. SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and < 0.5 MB model size. *arXiv* Preprint:1602.07360

[29] Zhang X, Zhou X, Lin M, Sun J. 2017. ShuffleNet: an extremely efficient convolutional neural network for mobile devices. *arXiv* Preprint:1707.01083

[30] Chen L, Yang Y, Wang Z, Zhang J, Zhou S, et al. 2023. Lightweight underwater target detection algorithm based on dynamic sampling transformer and knowledge-distillation optimization. *Journal of Marine Science and Engineering* 11:426

[31] Liu C, Wen J, Huang J, Lin W, Wu B, et al. 2024. Lightweight underwater object detection algorithm for embedded deployment using higher-order information and image enhancement. *Journal of Marine Science and Engineering* 12:506

[32] Li C, Liu W, Gong G, Ding X, Zhong X. 2025. SU-YOLO: spiking neural network for efficient underwater object detection. *arXiv* Preprint:2503.24389

[33] Gong R, Xu J, Zheng Z, Zhang D. 2026. YOLO-Starfish: fish object detection learning complex underwater features. *Scientific Reports* 16:13964

[34] Lin TY, Goyal P, Girshick R, He K, Dollár P. 2017. Focal loss for dense object detection. *arXiv* Preprint:1708.02002

[35] Li X, Wang W, Wu L, Chen S, Hu X, et al. 2020. Generalized focal loss: learning qualified and distributed bounding boxes for dense object detection. *arXiv* Preprint:2006.04388

[36] Chen Y, Dai X, Liu M, Chen D, Yuan L, et al. 2019. Dynamic convolution: attention over convolution kernels. *arXiv* Preprint:1912.03458

[37] Yang B, Bender G, Le QV, Ngiam J. 2019. CondConv: conditionally parameterized convolutions for efficient inference. *arXiv* Preprint:1904.04971

[38] Ding X, Zhang X, Ma N, Han J, Ding G, et al. 2021. RepVGG: making VGG-style ConvNets great again. *arXiv* Preprint:2101.03697

[39] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, et al. 2014. Going deeper with convolutions. *arXiv* Preprint:1409.4842

[40] Tolstikhin I, Houlsby N, Kolesnikov A, Beyer L, Zhai X, et al. 2021. MLP-mixer: an all-MLP architecture for vision. *arXiv* Preprint:2105.01601

[41] Touvron H, Cord M, Sablayrolles A, Synnaeve G, Jégou H. 2021. Going deeper with image transformers. *arXiv* Preprint:2103.17239

[42] Huang G, Sun Y, Liu Z, Sedra D, Weinberger KQ. 2016. Deep networks with stochastic depth. In *Computer Vision – ECCV 2016*, eds. Leibe B, Matas J, Sebe N, Welling M. Cham: Springer International Publishing. pp. 646–661 doi: 10.1007/978-3-319-46493-0_39

[43] Cai Z, Vasconcelos N. 2018. Cascade R-CNN: delving into high quality object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June, 2018*. USA: IEEE. pp. 6154–6162 doi: 10.1109/cvpr.2018.00644

[44] Ren S, He K, Girshick R, Sun J. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39:1137–1149

[45] Feng C, Zhong Y, Gao Y, Scott MR, Huang W. 2021. TOOD: task-aligned one-stage object detection. *2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October, 2021*. USA: IEEE. pp. 3490–3499 doi: 10.1109/ICCV48922.2021.00349

[46] Zhang S, Chi C, Yao Y, Lei Z, Li SZ. 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June, 2020*. USA: IEEE. pp. 9759–9768 doi: 10.1109/CVPR42600.2020.00978

[47] Varghese R, Sambath M. 2024. YOLOv8: a novel object detection algorithm with enhanced performance and robustness. *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), Chennai, India, 18–19 April, 2024*. USA: IEEE. pp. 1–6 doi: 10.1109/ADICS58448.2024.10533619

[48] Wang CY, Yeh IH, Mark Liao HY. 2025. YOLOv9: learning what you want to learn using programmable gradient information. In *Computer Vision – ECCV 2024*, eds. Leonardis A, Ricci E, Roth S, Russakovsky O, Sattler T, et al. Cham: Springer Nature Switzerland. pp. 1–21 doi: 10.1007/978-3-031-72751-1_1

[49] Wang A, Chen H, Liu L, Chen K, Lin Z, et al. 2024. YOLOv10: real-time end-to-end object detection. *Advances in Neural Information Processing Systems 37, Vancouver, Canada, 10−15 December, 2024*. San Diego, California: Neural Information Processing Systems Foundation, Inc. pp. 107984−108011 doi: 10.52202/079017-3429

[50] Rasheed AF, Zarkoosh M. 2025. YOLOv11 optimization for efficient resource utilization. *The Journal of Supercomputing* 81:1085

[51] Lyu C, Zhang W, Huang H, Zhou Y, Wang Y, et al. 2022. Rtmdet: an empirical study of designing real-time object detectors. *arXiv* Preprint:2212.07784

[52] Wang N, Gao Y, Chen H, Wang P, Tian Z, et al. 2020. NAS-FCOS: fast neural architecture search for object detection. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13−19 June, 2020*. USA: IEEE. pp. 11940−11948 doi: 10.1109/CVPR42600.2020.01196

Copyright: © 2026 by the author(s). Published by Maximum Academic Press, Fayetteville, GA. This article is an open access article distributed under Creative Commons Attribution License (CC BY 4.0), visit https://creativecommons.org/licenses/by/4.0/.