Research on marine flexible biological target detection based on improved YOLOv8 algorithm

Yu Tian*, Yanwen Liu*, Baohang Lin and Peng Li

College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, China

* These authors contributed equally to this work.

ABSTRACT

To address the challenge of suboptimal object detection outcomes stemming from the deformability of marine flexible biological entities, this study introduces an algorithm tailored for detecting marine flexible biological targets. Initially, we compiled a dataset comprising marine flexible biological subjects and developed a Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm, supplemented with a boundary detection enhancement module, to refine underwater image quality and accentuate the distinction between the images' foregrounds and backgrounds. This enhancement mitigates the issue of foreground-background similarity encountered in detecting marine flexible biological entities. Moreover, the proposed adaptation incorporates a Deformable Convolutional Network (DCN) network module in lieu of the C2f module within the YOLOv8n algorithm framework, thereby augmenting the model's proficiency in capturing geometric transformations and concentrating on pivotal areas. The Neck network module is enhanced with the RepBi-PAN architecture, bolstering its capability to amalgamate and emphasize essential characteristics of flexible biological targets. To advance the model's feature information processing efficiency, we integrated the SimAM attention mechanism. Finally, to diminish the adverse effects of inferior-quality labels within the dataset, we advocate the use of WIoU (Wise-IoU) as a bounding box loss function, which serves to refine the anchor boxes' quality assessment. Simulation experiments show that, in comparison to the conventional YOLOv8n algorithm, our method markedly elevates the precision of marine flexible biological target detection.

INTRODUCTION

With the rapid advancement of target detection algorithms, the detection accuracy for common tasks in simple scenarios, like pedestrian detection, now exceeds 90%. However, for targets with unique characteristics, like the flexible biological entities discussed herein, the algorithm's performance remains unsatisfactory. Consequently, it is necessary to customize and improve the algorithm according to these unique characteristics to improve the detection results.

Observing flexible marine biology offers a valuable method for assessing marine biological diversity and forecasting environmental shifts. Recently, underwater target detection employing machine vision has gained traction in marine studies. Utilizing visual autonomous perception technology to precisely identify and quantify various marine species, including sea cucumbers and octopuses in aquatic farms, aids seafood cultivators in real-time monitoring of marine life growth and environmental alterations, while also minimizing human labor and avoiding high-risk tasks. Despite this potential, four significant challenges hinder effective detection of flexible biological entities in marine settings (*Lin & Zhao, 2020*):

1) Flexible marine biological objects exhibit considerable variability in shape and posture due to their soft, deformable bodies.

2) Detection backgrounds in marine environments are often complex and variable.

3) Marine organisms excel in camouflage and concealment compared to inanimate objects.

4) In underwater environments, light absorption, scattering, and water quality significantly affect underwater imagery, often resulting in color distortion, blurring, and detail loss.

Based on the preceding analysis, the challenges can be categorized into two primary areas (*Chen et al., 2023a*): underwater image processing and object detection algorithms. Underwater image processing encompasses methods for image enhancement and image restoration (*Wang et al., 2019*). Enhancement techniques include prior model-based approaches such as ULAP and IBLA (*Song et al., 2018*; *Peng & Cosman, 2017*), which rely on predetermined knowledge or statistical assumptions about the environment. However, these assumptions may not hold in complex scenarios. Conversely, non-physical model-based methods, like the CLAHE algorithm, utilize histogram equalization to improve image clarity. Another example is the Relative Global Histogram Stretching (RGHS) algorithm, which enhances shallow water images by adjusting contrast and color, thereby achieving superior image quality with reduced noise (*Huang et al., 2018*). Recently, deep learning-inspired methods involving convolutional neural network (CNN) and generative adversarial network (GAN) like Water-net, have emerged. These methods employ a network's self-learning capability to conduct weighted fusion of input images, resulting in enhanced underwater imagery (*Li et al., 2019*).

To tackle underwater image restoration, *He, Sun & Tang (2020)* introduced the Dark Channel Prior (DCP) algorithm, leveraging prior knowledge for image restoration. Chiang's team (*Raveendran, Patil & Birajdar, 2021*) applied DCP to underwater contexts and developed a restoration algorithm that incorporates wavelength compensation and deblurring. Beyond DCP, *Wang, Zheng & Zheng (2017)* presented the Adaptive Attenuation-Curve Prior (AACP) algorithm for similar purposes. In the realm of object detection, algorithms are bifurcated into traditional methods and those predicated on deep learning (*Wei et al., 2023*). Traditional approaches (*Zou et al., 2023*), such as Cascade +Haar, DP (Deformable Parts Model), and Selective Search (SIFT+SVM), are becoming

outdated due to technological advancements and increasing data volumes. These methods often suffer from low computational efficiency due to the sliding window technique and are hampered by the manual intensity required for feature design and selection, affecting their accuracy, objectivity, robustness, and generalizability. Deep learning-based object detection algorithms are categorized into two types: a) two-stage algorithms, which separate detection into region proposal extraction and proposal classification and regression, with RCNN (*Girshick et al., 2014*), Fast region-based convolutional neural network (R-CNN) (*Sabri & Li, 2021*), Faster R-CNN (*Wang & Xiao, 2023*) and Mask R-CNN (*Rashid et al., 2022*) as prime examples; b) one-stage algorithms, which concurrently perform detection and classification, outputting class probabilities and object coordinates directly, exemplified by *Cao et al.*'s *(2020)* SSD and the YOLO series (*Jiang et al., 2022*) algorithms.

The performances of object detection algorithms, including those based on deep learning like YOLO and SSD (*Chen, 2023*; *Dong et al., 2022*), often diminishes with poor imaging quality and complex or varied detection backgrounds. Furthermore, the detection of flexible marine biological objects (*Liu, Liu & Jiang, 2023*), characterized by their soft bodies and significant shape changes, demands an algorithm with enhanced modeling capabilities for geometric transformations and a robust distinction between the foreground and background of an image.

Accordingly, this article introduces a flexible marine biological object detection algorithm using an enhanced YOLOv8. "Improvements of the CLAHE Underwater Image Enhancement Algorithm" discusses the incorporation of the CLAHE image enhancement algorithm with a boundary detection enhancement module to improve the underwater image dataset and delineate the boundaries between the image's foregrounds and backgrounds. "The Structure of Yolov8n Model and Improvement Strategy" outlines the comprehensive framework of the object detection algorithm based on YOLOv8n, detailing improvements in the backbone, neck, and loss function, and integrating the attention mechanism SimAM. "Experiment Research and Result Analysis" describes the assembly of a flexible marine biological dataset, the establishment of an experimental environment, and the execution of ablation studies along with comparative analyses of different models. "Conclusions" concludes the article.

IMPROVEMENTS OF THE CLAHE UNDERWATER IMAGE ENHANCEMENT ALGORITHM

CLAHE (*Ting et al., 2023*) is an advanced image enhancement technique evolved from Adaptive Histogram Equalization (AHE). The fundamental concept of CLAHE is to partition the image into multiple segments (or tiles), for example, $8 \times 8 = 64$ tiles. Unlike AHE, where histogram equalization is applied to the entire image, CLAHE performs this operation individually on each tile. While AHE excels in enhancing local image contrast and delineating image boundaries, it falls short by intensifying image noise. To alleviate this problem, CLAHE imposes a contrast limitation on each tile to prevent the excessive

noise amplification associated with AHE. The operational principle of CLAHE is outlined as follows:

Assuming that the sliding window size of the CLAHE algorithm is $M \times M$, therefore the local mapping function is

$$m(i) = \frac{255 \times C_{DF}(i)}{M \times M}. \tag{1}$$

$C_{DF}(i)$ is the cumulative distribution function of the sliding window local histogram; the slope S of the local mapping function is

$$S = \frac{d(m(i))}{di} = H_{ist}(i) \times \frac{255}{M \times M}. \tag{2}$$

According to Formulas (1) and (2), by adjusting S, the height $H_{ist}(i)$ can be modified, which in turn adjusts the image contrast. If the maximum value of S is represented as S_{max}, the maximum value of the histogram is

$$H_{max} = S_{max} \times \frac{M \times M}{255}. \tag{3}$$

To maintain a consistent overall area of the histogram, the part of the histogram that exceeds H_{max} is truncated and evenly distributed within the entire histogram range. In practice, this truncation is achieved by setting a threshold T (instead of H_{max}), which results in the entire image's histogram increasing by L. Consequently,

$$H_{max} = T + L. \tag{4}$$

Finally, the improved histogram is

$$H_{ist}(i) = \begin{cases} H_{ist}(i) + L, H_{ist}(i) < T \\ H_{max}, H_{ist} \geq T \end{cases}. \tag{5}$$

To summarize, by adjusting S_{max}, histograms with different maximum heights H_{max} can be enhanced to varying degrees. To augment the algorithm's capacity for delineating foreground and background boundaries in addition to enhancing underwater images, a Boundary Detection Enhancement Module (BEM) (*Guan et al., 2022*) is integrated into the CLAHE algorithm to focus the network's attention on contour representation. This module leverages both high-level and low-level features to isolate boundary details and eliminate irrelevant information. Its operational mechanism is depicted in Fig. 1. The image information input into the image preprocessing algorithm enters both the CLAHE algorithm channel and the boundary detection enhancement channel. The CLAHE algorithm channel adjusts image contrast, while the front and rear edge enhancement channel concentrates on extracting and intensifying boundary information within the image. This process not only enhances the boundary information between the foreground and background but also accentuates the shape details of the target object. Subsequently,
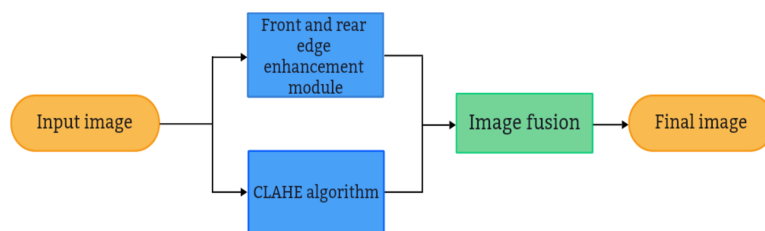
Figure 1 The working mechanism of the CLAHE underwater image enhancement algorithm with foreground and background boundary enhancement module.
Full-size ⬛ DOI: 10.7717/peerj-cs.2271/fig-1

the outputs from both channels are merged, effectively enhancing the underwater image alongside the object's shape and boundary details concurrently.

THE STRUCTURE OF YOLOV8N MODEL AND IMPROVEMENT STRATEGY

YOLOv8n

YOLOv8, the most recent advancement (*Wang et al., 2023*) in the YOLO series, is capable of performing both detection and classification tasks. This SOTA model builds upon previous iterations, incorporating novel features and enhancements. Notable innovations include a redesigned backbone network, an anchor-free head, and an updated loss function. This study utilizes the YOLOv8n variant as the foundational model.

As depicted in Fig. 2, the YOLOv8n architecture comprises four main components: Input, Backbone, Neck, and Head (*Oh & Lim, 2023*). The Input segment employs Mosaic for data augmentation, which is deactivated during the final 10 epochs. The anchor-free mechanism predicts the object's center directly, eliminating the need for anchor box offset predictions and thereby streamlining the non-maximum suppression (NMS) process.

The Backbone serves as the primary feature extraction module, encompassing Conv, C2f, and SPPF modules. The Conv module applies convolution, BN, and SiLU activation to the input image, maintaining a similar structure to YOLOv5's backbone. Adopting the CSP concept, the C3 module is substituted with a C2f structure to enhance gradient flow, with channel numbers tailored to different model scales. Additionally, YOLOv8 retains the SPPF module from the YOLOv5 (*Wu et al., 2021*) design, enabling the conversion of variable-sized feature maps into fixed-size feature vectors.

The Neck's primary role is to amalgamate multi-scale features and construct a feature pyramid, utilizing the PANet architecture (*Liu et al., 2023*). This integral structure comprises two main components: the feature pyramid network FPN and the path aggregation network PAN. The synergy between FPN and PAN enables comprehensive integration of bidirectional information flow within the network, enhancing its detection capabilities.

The Head segment fragment extracts the classification and positioning details of objects of different sizes from feature maps of different sizes. In YOLOv8n, the Head has been
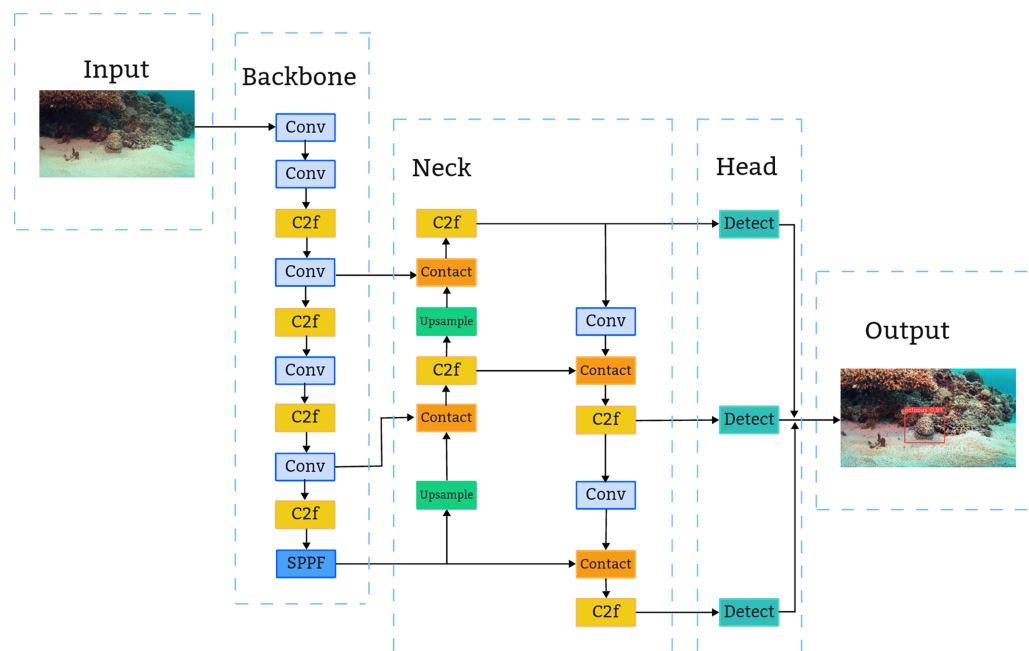
Figure 2 Structure of YOLOv8n network. Full-size ⬚ DOI: 10.7717/peerj-cs.2271/fig-2

updated to the prevalent decoupled head structure, which segregates the classification and detection functions and transitions from an Anchor-Based to an Anchor-Free approach.

Deformable convolutional network (DCN) module

Given the flexible nature and variable shapes of marine biological objects, object detection algorithms need to enhanced capabilities to model geometric transformations. Traditional convolution operations partition the feature map into segments matching the convolution kernel's size, with each segment's position on the feature map being static. This approach is less effective for objects undergoing complex deformations. As illustrated in Fig. 3A, the fixed size and shape of a 3 × 3 convolution kernel limit its capacity to model geometric transformations during convolution. To address this, researchers like *Dai et al. (2017)* introduced an innovative concept of integrating an offset to the standard convolution kernel, as detailed in Fig. 4, showcasing the principles of deformable convolution operations. Figures 3B–3D depict how each element of the deformable convolution kernel is adjusted by a specific offset, altering the sampling positions (*Nguyen et al., 2023*) and thereby augmenting the convolutional neural network's capability to model geometric transformations and improving the model's proficiency in detecting flexible marine objects.

Traditional two-dimensional convolution first uses the sampling grid $\mathcal{R}$ to sample on the input feature map x (*Li & Liu, 2023*), and then uses the weight w to weight the sampled values. The sampling grid determines the size of receptive field. For
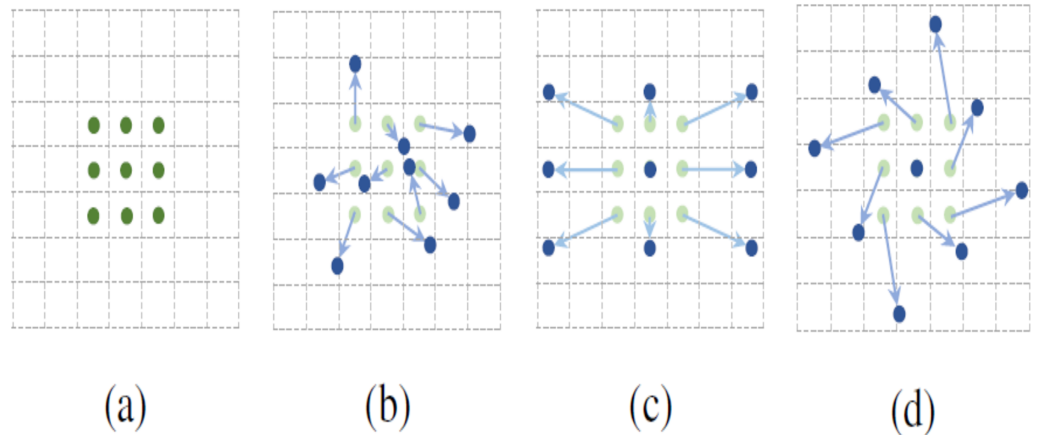
Figure 3 (A–D) Sampling point location map of deformable convolution.
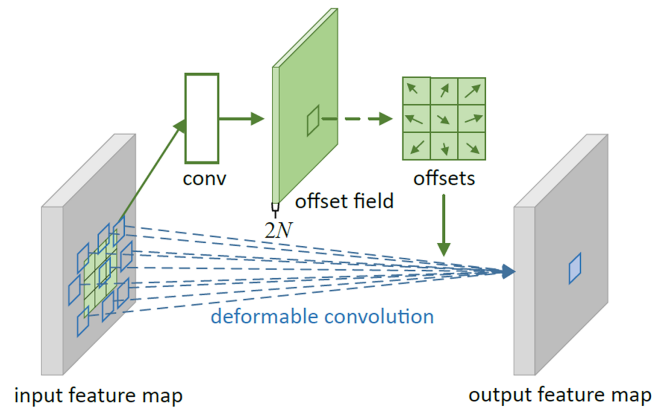Full-size ⬜ DOI: 10.7717/peerj-cs.2271/fig-3

Figure 4 Principles of deformable convolution operation.
Full-size ⬜ DOI: 10.7717/peerj-cs.2271/fig-4

example, the sampling grid of a regular two-dimensional convolution of size 3×3 can be expressed as:

$$\mathcal{R} = [(-1,-1), (-1,0), \dots, (0,1), (0,1)]. \tag{6}$$

For each position p_0 in the output feature map y, there is

$$y(p_0) = \sum_{p_n} (p_n) \cdot x(p_0 + p_n) \cdot \tag{7}$$

In Formula (7), p_n denotes the enumeration of positions in $\mathcal{R}$. In deformable convolution, the regular grid $\mathcal{R}$ is offset by $\{\delta_{p_0} = n = 1, \dots, N\}$, in which $N = |\mathcal{R}|$. Formula (7) then becomes the following form:

$$y(p_0) = \sum_{p_n} (p_n) \cdot x(p_0 + p_n + \delta_{p_0}) \cdot \tag{8}$$

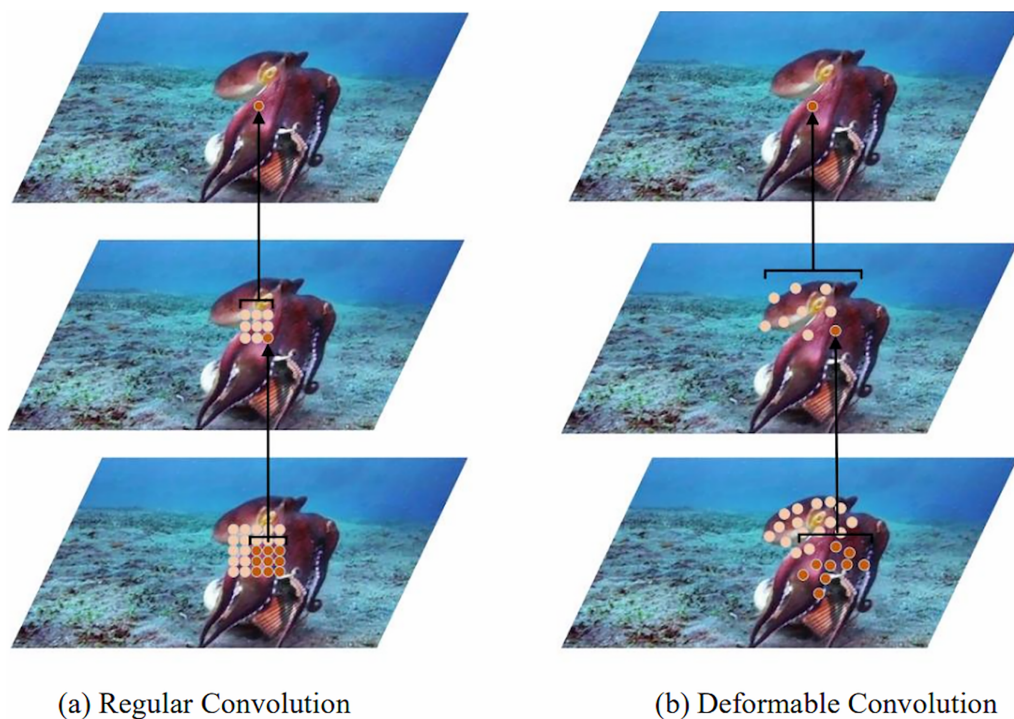(a) Regular Convolution   (b) Deformable Convolution

Figure 5 (A–D) A comparison of receptive field and positions of sampling points of regular convolution and deformable convolution operation. Full-size 🖼 DOI: 10.7717/peerj-cs.2271/fig-5

This adjustment causes the sampling points to assume irregular and shifted positions. In deformable convolution, δ_{p_0} is determined by convolving on x. The process of learning δ_{p_0} and generating the convolution kernel in deformable convolution is depicted in Fig. 4. Furthermore, the incorporation of multiple deformable convolutional layers significantly influences the receptive field and composite deformation, as illustrated in Fig. 5. Figure 5A employs standard convolution, whereas Fig. 5B utilizes deformable convolution. Both figures share identical top-level feature maps and sampling points. However, through successive convolutional layers, it is evident that deformable convolution allows for the adaptive adjustment of sampling points in response to the target's scale and shape.

Convolutional neural networks that incorporate deformable convolution are referred to as deformable convolutional neural networks (DCNN). From the discussed content, it is clear that a DCNN model equipped with deformable convolution can effectively modify the receptive field, enrich the feature map's useful feature information, and enhance the network model's capacity to model the geometric transformation of the detected target based on its size and shape.

Compared to traditional convolution, deformable convolution introduces additional parameters and greater flexibility to the network model, but this can complicate the convergence of the model. Improper design or placement of the deformable convolution module within the network might lead to training collapse.
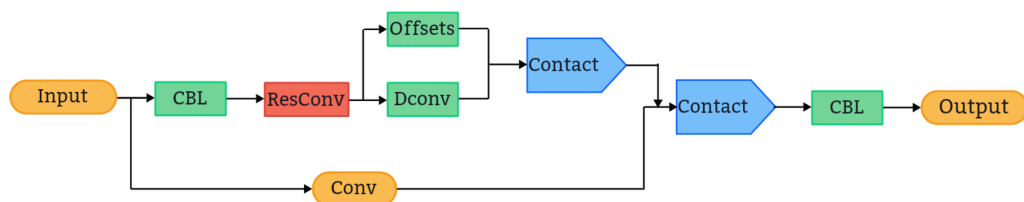
Figure 6 The structure of deformable convolution module.
Full-size 🖼 DOI: 10.7717/peerj-cs.2271/fig-6

To mitigate the risk of network collapse during training, this study draws on the classic ResNet (*Shafiq & Gu, 2022*) structure to inform the design of the deformable module unit, including its stacking method and placement. A skip connection is employed between two deformable convolution layers to enhance network convergence, as depicted in Fig. 6, which illustrates the deformable convolution basic module structure utilized in this experiment. CBL module refers to "convolutional block laver", which is a basic convolutional block composed of a series of convolutional layers, batch normalization layers and activation function layers.

Consequently, the DCN module is more adept at managing non-rigid deformations and positional variations of underwater targets, enhancing detection accuracy. Through learning deformable convolution parameters, the model dynamically adjusts the convolution kernel's sampling positions based on the target's specific shape and location, enabling more precise target characterization. This enhancement bolsters the underwater target detection model's performance, increasing its adaptability to the complexities and variability of underwater environments.

RepBi-PAN neck network

The Neck network plays a crucial role in applying transformation and concatenation operations to the feature map, enriching and diversifying its feature information while enhancing its robustness. In YOLOv8n, this network incorporates the PANet architecture, which introduces an additional bottom-up path to FPN, facilitating quicker signal transmission from low-level to high-level features and *vice versa*, thus enhancing the propagation of precise low-level feature signals. Numerous scholars have developed innovative neck network designs based on FPN and PAN, with RepBi-PAN standing out as a notable example.

RepBi-PAN features a Bi-directional Concatenation (BiC) module, depicted in Fig. 7. This module amalgamates feature maps from three adjacent depths by applying 1×1 convolutions for dimensionality reduction on feature maps of identical scale and depth. For larger-scale feature maps from shallower layers, it employs 1×1 convolution for dimensionality reduction followed by a 3×3 convolution with a stride of 2 for down-sampling. Conversely, for smaller-scale feature maps from deeper layers, it uses a 2×2 transposed convolution followed by up-sampling. Afterward, it concatenates the three feature maps and applies another round of 1×1 convolution for dimensionality reduction. Through this process, the BiC structure assimilates feature information across adjacent
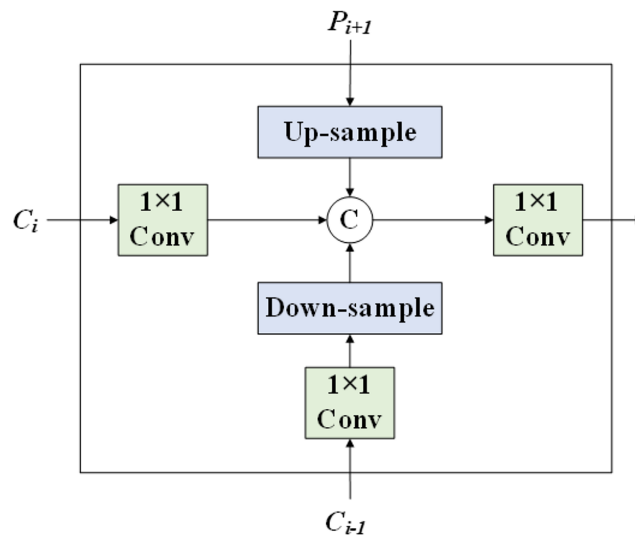
Figure 7 The structure of BiC.   Full-size ⬜ DOI: 10.7717/peerj-cs.2271/fig-7
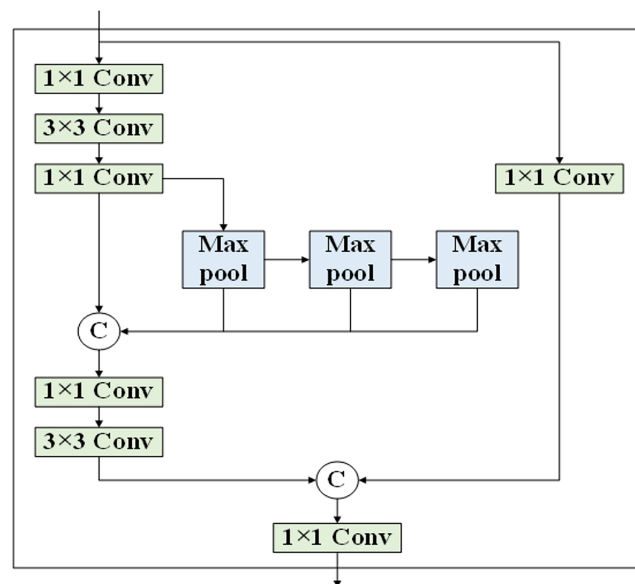

Figure 8 The structure of SlimCSPSPPF.   Full-size ⬜ DOI: 10.7717/peerj-cs.2271/fig-8

layers, facilitating cross-layer and cross-channel feature fusion, enhancing the interaction between deep and shallow layers, and significantly improving the neck network's feature fusion capability. Furthermore, this module aids in preserving precise positioning signals, which is particularly crucial for accurately locating smaller objects.

In addition to the BiC structure, RepBi-PAN enhances the SSP structure (*Tang et al., 2023*) by streamlining the SPPF block into a version akin to CSP, termed the SlimCSPSPPF block. Figure 8 illustrates the refined SlimCSPSPPF structure. Upon receiving the deepest layer's feature map, SlimCSPSPPF bifurcates it into two pathways. One pathway engages in multi-scale transformation of the feature map through compound convolution and spatial
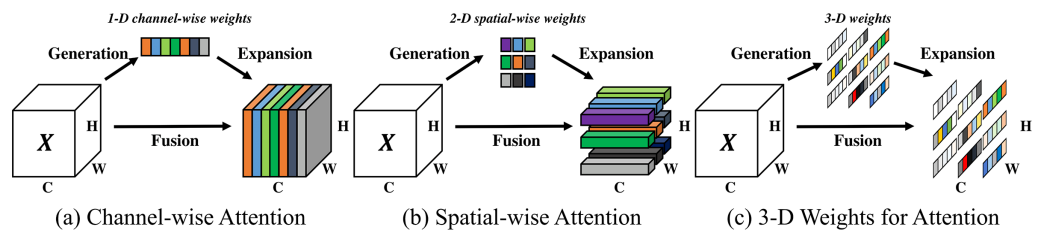
Figure 9 (A–C) The comparison of the principles of SimAM attention structure, channel attention structure and spatial attention structure. Full-size ◪ DOI: 10.7717/peerj-cs.2271/fig-9

pyramid pooling, while the other employs a skip connection to preserve SlimCSPSPPF's integrity. The SlimCSPSPPF structure surpasses the original SPPF in terms of feature representation, efficiency, and processing speed.

RepBi-PAN integrates the BiC structure to link convolutional layers across four different depths at the Backbone's second and third levels of convolutional depth and implements the SlimCSPSPPF structure at the deepest convolutional layer. By refining the original YOLOv8n's PANet structure, RepBi-PAN substantially boosts the model's capacity to amalgamate multi-scale features pertinent to flexible marine organisms, thereby enhancing the detection accuracy of the algorithm for these subjects.

The SimAM attention structure

In tasks involving the detection of marine life, where the target closely resembles the background, incorporating an attention mechanism prior to the YOLOv8n network's head can significantly mitigate the impact of extraneous information. This article employs SimAM, a three-dimensional, parameter-free attention mechanism (*Zhang, Liu & Jiang, 2022*).

Traditional attention mechanisms, such as SE, ECA, and SOCA, are constrained by their intrinsic designs, typically refining features along either channel or spatial dimensions and generating one- or two-dimensional weights. This constrains the model's capacity to concurrently learn weights across both channels and spatial dimensions and escalates the model's parameter count. Unlike these mechanisms, SimAM introduces a parameter-free approach to generate full 3D weights, achieving three-dimensional weight derivation without increasing the original network's parameter load. Figure 9 compares the operational principles of the SimAM attention structure with those of channel and spatial attention structures.

Figure 9 illustrates that channel attention generates one-dimensional weights for each channel, refining the feature map's channel weights while treating all spatial positions within a channel uniformly (*Zhang, Li & Zhang, 2023*). Spatial attention generates two-dimensional spatial weights, refines the weights of different positions within the same channel of the feature map, and treats all channels equally. Conversely, the 3D weights generated by SimAM can refine different positions across various channels. Hence, it surpasses both channel attention and spatial attention. The mechanism through which the SimAM attention structure produces 3D weights is detailed below.

The formula for calculating 3D weights is presented as:

$$\dot{X} = sigmoid\left(\frac{1}{E}\right) \odot X. \tag{9}$$

X is the input feature, and the sigmoid function is applied to constrain any excessive values in E. E represents the energy function on each channel. The calculation process of E is

$$E = \frac{4(\sigma^2 + \lambda)^2}{(t - \mu)^2 + 2\sigma^2 + 2\lambda}. \tag{10}$$

In Formula (10), t denotes the value of the input feature, $t \in X$. The lower the energy E is lower, greater the difference between the target neuron t and other neurons x_i in the corresponding dimension, thereby increasing the relative importance. Consequently, the significance of the neuron can also be denoted by $1/E$. λ is the constant $1e - 4$, μ and σ^2 denote the mean and variance of each channel in X respectively, with the computation process outlined as follows:

$$\mu = \frac{1}{M} \sum_{i=1}^{M} x_i \tag{11}$$

$$\sigma^2 = \frac{1}{M} \sum_{i=1}^{M} (x_i - \mu)^2. \tag{12}$$

M denotes the number of neurons, which equals the product of the width (W) and the height (H) of the feature map. x_i represents all the neurons in the dimension corresponding to the target neuron. By manipulating these neurons, the network can assign greater weights to those containing crucial information, thereby enhancing detection and positioning accuracy without the need for additional network parameters.

It is considered that the detection network of flexible marine biological targets is easily affected by background interference similar to the target, incorporating the SimAM attention mechanism into the detection network allows for a comprehensive and efficient evaluation of feature weights and the fusion of target information across multiple dimensions. This process not only amplifies the information pertaining to flexible marine biological targets but also diminishes the influence of natural background information, ensuring a focused target detection. Furthermore, the SimAM attention structure dose not introduce extra network parameter calculations, thus minimally affecting the model's running speed.

The WIoU loss function

In current detection algorithms, researchers designing the bounding box loss function often presume that the training dataset's bounding box annotations are of high quality, focusing on improving the bounding box regression function's fitting capabilities.

However, low-quality bounding box annotations are prevalent in many datasets. Enhancing the fitting ability of the bounding box regression function for these low-quality images can adversely affect the model's target positioning performance. The training data in this study, annotated by the authors, inevitably include such low-quality annotations, which can diminish the model's generalization capability.

The baseline model of YOLOv8 employs the CIoU loss function, which considers the overlapping area, center point distance, and aspect ratio. However, the aspect ratio is represented as a relative value, introducing a degree of ambiguity. Additionally, it fails to address the balance issue between challenging and straightforward samples, potentially leading to slow convergence and reduced prediction accuracy. Given the presence of low-quality samples in our dataset, factors like aspect ratio and distance can intensify penalties for these samples, undermining the model's generalization capabilities. When there is significant overlap between the target and anchor boxes, reducing the emphasis on geometric factors can enhance model generalization. Consequently, this study adopts the WIoU loss function over the CIoU loss function. The WIoU loss function more effectively addresses sample imbalance, improving the model's convergence rate.

WIoU employs a dynamic, non-monotonic focusing mechanism to assess the anchor box's quality and leverages gradient gain to bolster the efficacy of high-quality anchor boxes while mitigating the effects of detrimental gradients (*Chen et al., 2023b*), thereby enhancing the algorithm's overall performance. WIoU introduces a dual-layer attention mechanism to avert slow convergence, augment convergence accuracy, and bolster model generalization. It is posited that the corresponding position of (x, y) in the target frame is (x_{gt}, y_{gt}), R_{WIoU} denotes the loss of high-quality anchor boxes. The WIoUv1 formula is presented as:

$$\mathcal{L}_{WIoUv1} = R_{WIoU}\mathcal{L}_{IoU} \tag{13}$$

In which

$$R_{WIoU} = \exp\left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{\left((w_c)^2 + (h_c)^2\right)^*}\right) \tag{14}$$

To circumvent the generation of substantial harmful gradients by low-quality samples, WIoUv3 is devised utilizing β and WIoUv1. The formula is delineated as follows:

$$\mathcal{L}_{WIoUv3} = r\mathcal{L}_{WIoUv1}. \tag{15}$$

In which

$$r = \frac{\beta}{\alpha^{\beta-\delta}}, \ \beta = \frac{\mathcal{L}_{IoU}^*}{\mathcal{L}_{IoU}} \in [0, +\infty). \tag{16}$$

This study employs the WIoUv3 loss function for experimental analyses. This function incorporates a dynamic, non-monotonic focusing mechanism (*Wang et al., 2023*). It
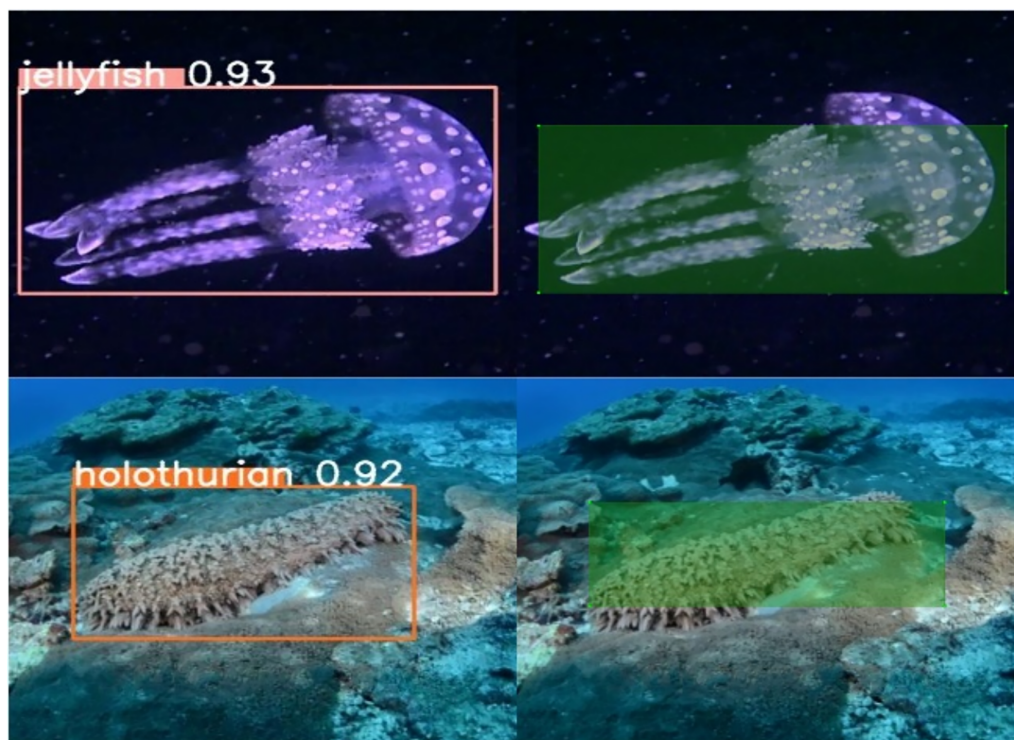
Figure 10 Low-quality bounding box annotations in the training and ideal predictions of Wise-IoU.
Full-size 🖼 DOI: 10.7717/peerj-cs.2271/fig-10

dynamically computes the "outlier degree" to serve as a metric for evaluating the quality of anchor boxes and implements a gradient gain allocation strategy. A lower outlier degree indicates a higher-quality anchor box, to which a reduced gradient gain is allocated, refocusing the bounding box on anchor boxes of average quality. Conversely, assigning diminished gradient gains to anchor boxes with higher outlier degrees effectively mitigates the impact of low-quality examples by limiting their potential to generate large, detrimental gradients. Furthermore, the calculation of the outlier degree is dynamic throughout the training process, enabling WIoU to continually adapt its gradient gain distribution strategy to optimize performance in the prevailing context.

As demonstrated in Fig. 10, Wise-IoU can reduce the influence of these low-quality labels and achieve better prediction when the labeling frames of jellyfish and holothurian in training data are quite different from the real ones. The dataset for this study was developed through the authors' independent collection and annotation of images, encompassing over 7,000 annotated boxes set against a complex seabed environment background.

Given the inevitability of missing and low-quality annotations, incorporating WIoU into the detection algorithm significantly reduces the adverse effects of substandard training data, boosts the bounding box regression function's adaptability, and enhances the algorithm's performance.
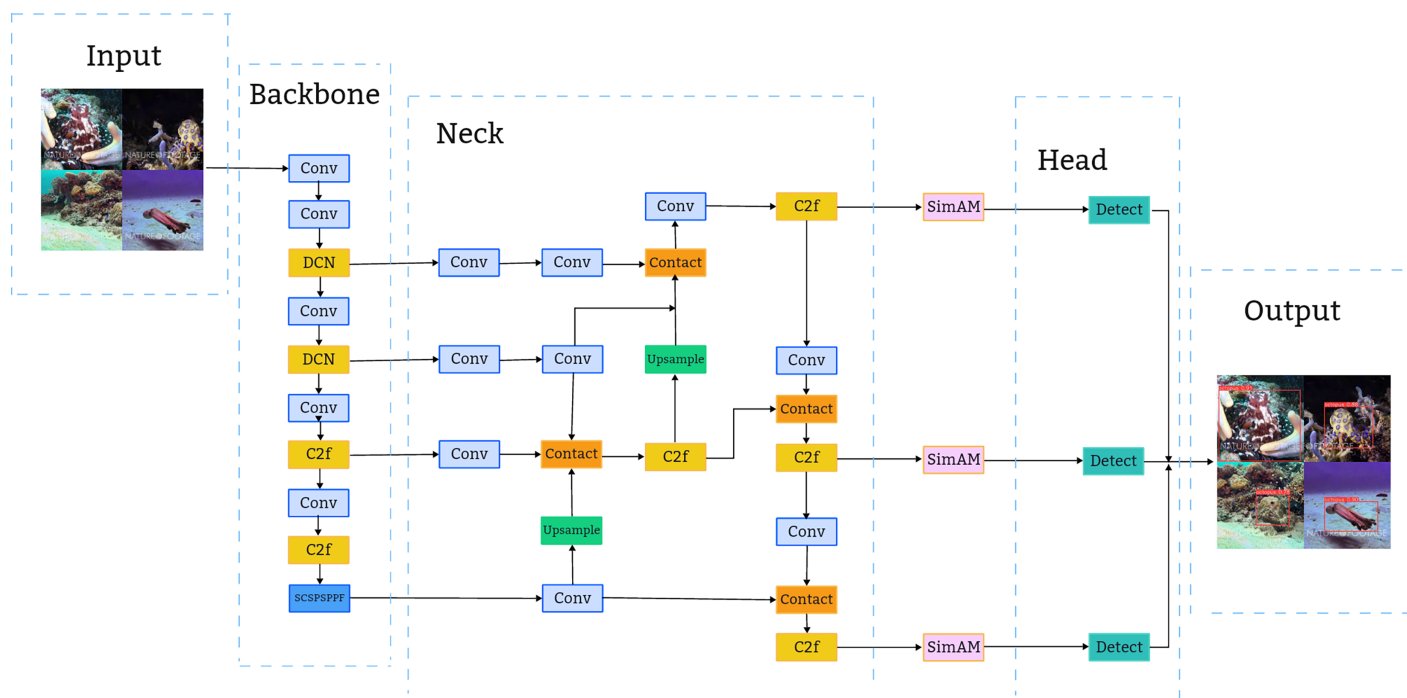
Figure 11 The overall structure of the improved algorithm model.

The overall framework of marine flexible biological target detection algorithm based on YOLOv8n

Following the evaluation of the target dataset and application context, enhancements were made to the original detection algorithm. Figure 11 illustrates the comprehensive architecture of the refined flexible marine biological object detection algorithm network, based on YOLOv8n. This architecture combines the original YOLOv8 model with the improvements mentioned above.

EXPERIMENT RESEARCH AND RESULT ANALYSIS

Configuration of the experiment

The experimental environment comprises Ubuntu 20.04 64-bit, PyCharm 2023.1.2, Pytorch 2.0.1, CUDA 11.8 (for training acceleration), and Python 3.11. The hardware setup includes an AMD Radeon 7-5700X CPU and an RTX4080 GPU. These processors are utilized for image preprocessing and model training, respectively. The enhanced algorithm's outcomes are benchmarked against those of leading target detection algorithms using GPU-accelerated training.

Evaluating index of the model

The evaluation metrics are precision, recall, and mean average precision (mAP).

Precision: the ratio of correctly predicted positive observations to the total predicted positives.

$$Precision = \frac{TP}{TP + FP}. \tag{17}$$

Recall: the ratio of correctly predicted positive observations to all observations in actual class.

$$Recall = \frac{TP}{TP + FN}. \tag{18}$$

Average precision: By plotting recall on the x-axis and precision on the y-axis, a curve is generated. The area under this curve represents the AP.

$$AP = \int P(R)dR. \tag{19}$$

AP represents a detailed mean accuracy, computed at intervals of 0.05 across a range from 0.5 to 0.95. Due to its complexity, the currently commonly used index is AP^{50}, which refers to the average accuracy when IoU = 0.5. AP^{50} generally refers to the average detection accuracy of a single category of targets. The weighted average AP^{50} value of multiple categories is also called mAP^{50} (%) (mean average precision). mAP^{50} (%) offers a comprehensive assessment of the model's detection accuracy across all categories, serving as the experimental evaluation metric in this study. The mean average precision is delineated in Formula (20), where N signifies the number of target categories, and AP represents the average accuracy of a single category target.

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N}. \tag{20}$$

Dataset collection

a) Dataset. To ensure the dataset's representations of biological object postures, locations, and scenarios accurately reflect real-world conditions and closely align with the actual data distribution, this study sourced extensive footage of authentic oceanic environments from documentaries, employing frame-by-frame annotation to develop a dataset of flexible marine biological entities. The utilized documentary footage is publicly available, making it suitable for this research's dataset needs. Figure 12 displays the distribution of target categories within the dataset, which comprises a total of 6,749 images featuring prevalent flexible marine species, such as octopus, holothurian, starfish, and jellyfish. Images were annotated using Labelimg. The dataset is divided into a training set with 5,341 images and a validation set consisting of 1,408 images. It encompasses a diverse array of images captured across various seasons, showcasing the organisms' body postures during different activities and habitats, under varying light conditions and environmental settings. Figure 13 presents a selection of images from the dataset.
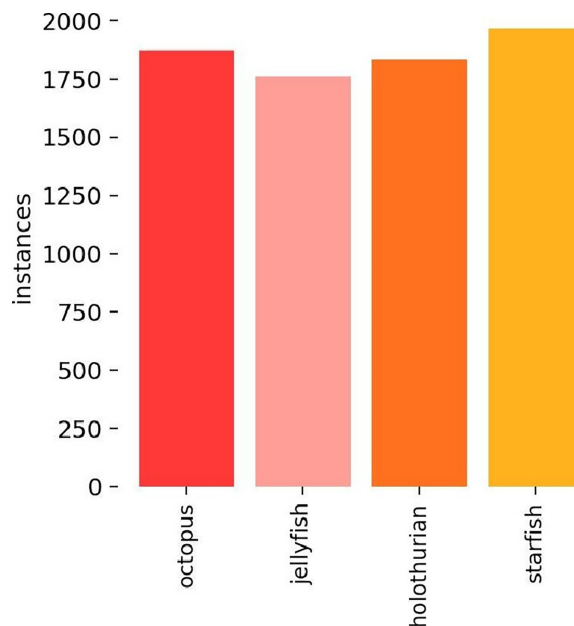
Figure 12 The histogram of categories of flexible marine biological objects.
Full-size ☑ DOI: 10.7717/peerj-cs.2271/fig-12

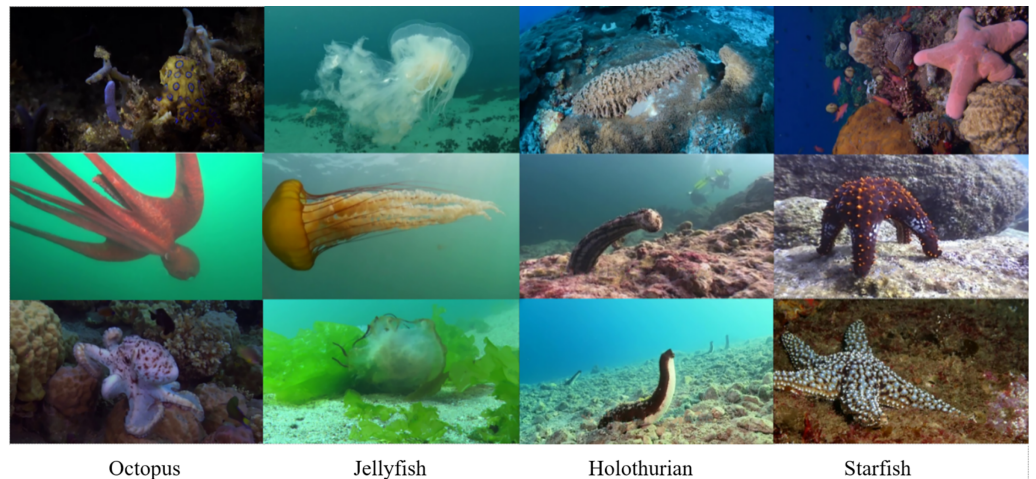| Octopus | Jellyfish | Holothurian | Starfish |

Figure 13 Some images in the dataset.   Full-size ☑ DOI: 10.7717/peerj-cs.2271/fig-13

b) Dataset image enhancement. Given that the subjects of this dataset are flexible biological entities inclined to camouflage and blend into their surroundings, they often appear in environments closely resembling themselves. Consequently, this study introduces targeted enhancements to the standard CLAHE underwater image enhancement algorithm, incorporating boundary detection into the enhancement module to accentuate the distinction between foreground and background while adjusting contrast. As illustrated in Fig. 14, the contrast in images processed with HE and CLAHE has notably improved, with a clear enhancement in detail visibility. However, some HE-processed images exhibit color shift issues, particularly noticeable in the second image.
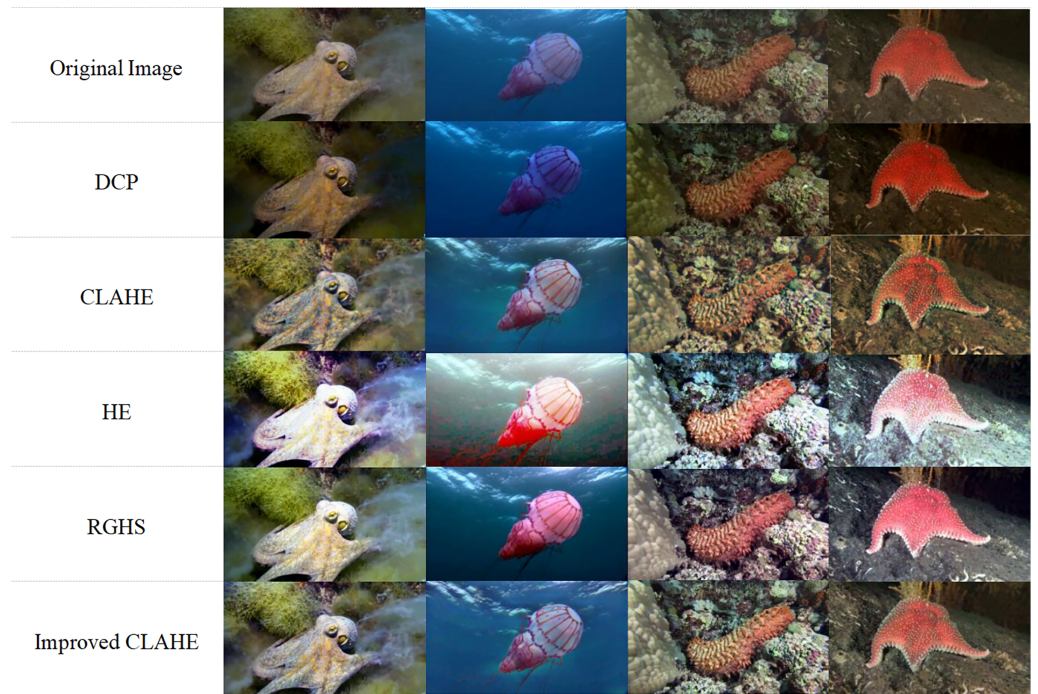
Figure 14 Effects of different image processing algorithms.
Full-size ☑ DOI: 10.7717/peerj-cs.2271/fig-14

Table 1 Performance index of different models comparison table.

Methods	AP^{50} (%)				mAP^{50} (%)
	Holothurian	Starfish	Jellyfish	Octopus	
Original image	66.5	77.9	80.5	78.5	75.9
DCP	67.9	78.1	80.3	79.7	76.2
CLAHE	67.7	77.7	82.2	79.2	76.7
HE	68.1	77.2	80.4	79.8	76.4
RGHS	69.0	78.1	79.9	80.5	76.5
Improved CLAHE	68.5	75.5	81.1	81.2	77.1

DCP-processed images show reduced contrast and intensified blue-green color shifts, alongside diminished overall brightness. While RGHS processing improves image brightness, it can lead to distortion and a reddish color tint. In contrast, images enhanced with the modified CLAHE algorithm display enhanced detail clarity without significant color shifts or localized over-enhancement. Overall, the image quality significantly benefits from the enhancements applied by the refined CLAHE algorithm.

To evaluate the impact of various image enhancement algorithms on model detection outcomes, this study utilized images processed by different algorithms for model training and testing. The experimental results, presented in Table 1.

Table 2 Model training parameters.

Name of parameter	Meaning	Setting
Pretraining weight	Initial weights of the model	YOLOv8n.pt
Optimizer	Parameter updating algorithm of the model	Adam
Image-size	Input image size of the model	640×640
lr0	Initial learning rate	0.01
Batch size	Batch size	16
Epoch	Training times of all training sets	200

The experimental results reveal that, relative to the original image, model detection accuracy has improved following preprocessing with CLAHE, RGHS, HE, and DCP. Notably, there was a significant enhancement in the detection quality of octopus. The mAP^{50} (%) for the enhanced CLAHE algorithm increased by 1.2% (relative to the original image), followed by HE at 0.5%, RGHS at 0.6%, DCP at 0.3%, and standard CLAHE at 0.8%. These findings convincingly demonstrate the effectiveness of the enhanced CLAHE algorithm in augmenting underwater images, prompting its selection for future experiments in this study.

Model training

Table 2 outlines the model training parameter settings. The dataset is partitioned into a training set comprising 5,341 images and a validation set with 1,408 images. Moreover, to mitigate overfitting during training, a label smoothing technique is employed.

Experiment results and analysis

Effectiveness verification of the improved algorithm

To assess the efficacy of the improved algorithm module, this study employs the original YOLOv8n model as the baseline model and mAP^{50} (%) as the evaluation metric. Ablation studies were performed using various combinations of enhancement modules. Compared to the baseline YOLOv8n model, the following five enhancements each contributed to an increase in model detection accuracy, as detailed in Table 3. Post-processing with the enhanced CLAHE algorithm resulted in a 1.2% increase in mAP^{50} (%). Substituting the original C2f module with deformable convolution (DCN) led to a 1.3% rise in mAP^{50} (%). Incorporation of the RepBi-PAN network model into the neck improved feature fusion, while the introduction of the SimAM attention module augmented the algorithm's feature processing capabilities. Subsequently, adopting the WIoU bounding box loss function in place of the original CIoU further elevated mAP^{50} (%). Of all modifications, the DCN module exhibited the most substantial enhancement, showing the most pronounced effect. Collectively, these modifications elevated the YOLOv8n-Improve's mAP^{50} (%) by 4.6% compared to the original YOLOv8n, indicating significant improvement.

Figure 15 presents the P-R diagram, illustrating the sequential addition of improvement methods. The detection accuracy for sea cucumbers, starfish, octopus, and jellyfish has

Table 3 Ablation experiment result.

Model	CLAHE	DCN	RepBi-PAN	SimAM	WIoU	Recall (%)	Precision (%)	mAP^{50} (%)
YOLOv8n						70.7	82.5	75.9
	√					69.8	84.8	77.1
	√	√				71.1	86.6	78.4
	√	√	√			70.8	87.3	79.1
	√	√	√	√		72.2	87.9	80.0
	√	√	√	√	√	71.9	89.8	80.5
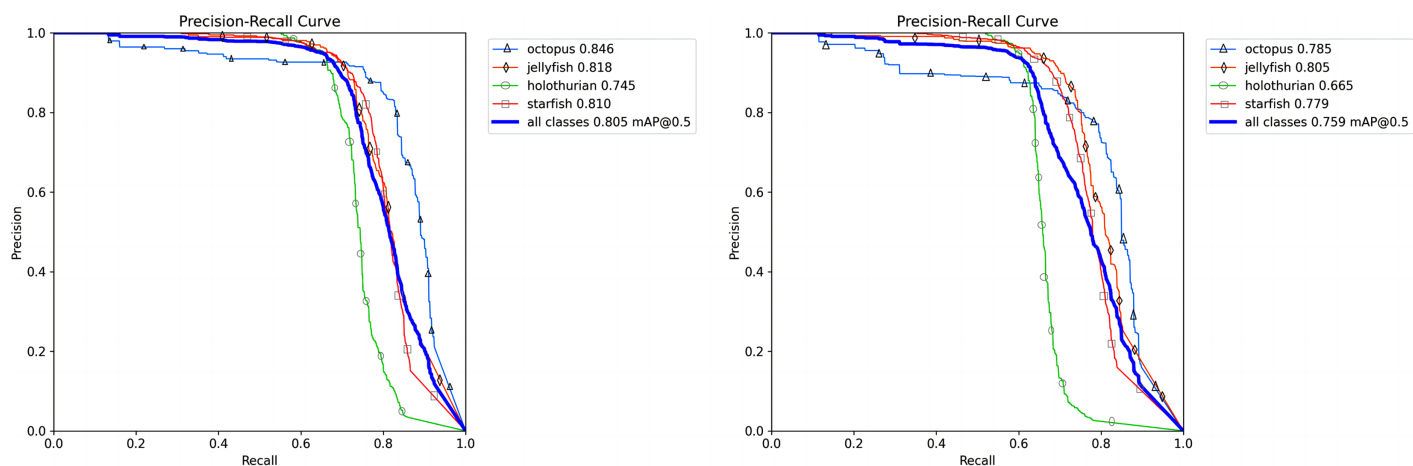
Figure 15 The P-R diagram in which the original YOLOv8 and the improving methods is added. Full-size ▣ DOI: 10.7717/peerj-cs.2271/fig-15

seen overall enhancements through these methods. Specifically, the detection accuracy improvements are as follows: sea cucumbers by 8%, octopus by 6.1%, starfish by 3.1%, and jellyfish by 1.3%. Consequently, the final detection accuracies are 74.5% for sea cucumbers, 84.6% for octopus, 81.0% for starfish, and 81.8% for jellyfish, indicating substantial improvement.

Figure 16 illustrates that YOLOv8n-Improve converges more rapidly, and its loss function values are lower compared to YOLOv8n. The integration of the DCN module enhances feature extraction efficiency, while the inclusion of RepBi-PAN ensures more comprehensive feature fusion across layers, thereby improving bounding box regression accuracy and reducing loss values. In Fig. 17, YOLOv8n-Improve exhibits superior precision and recall relative to YOLOv8n. This improvement is attributed to the SimAM attention module's robust feature information processing capabilities, which effectively address the challenge of feature information overload due to the diversity of flexible biological objects' features. It emphasizes the crucial features of marine biological entities while minimizing background noise. Furthermore, the adoption of the WIoU bounding box loss function mitigates the adverse effects of low-quality annotations in the dataset.
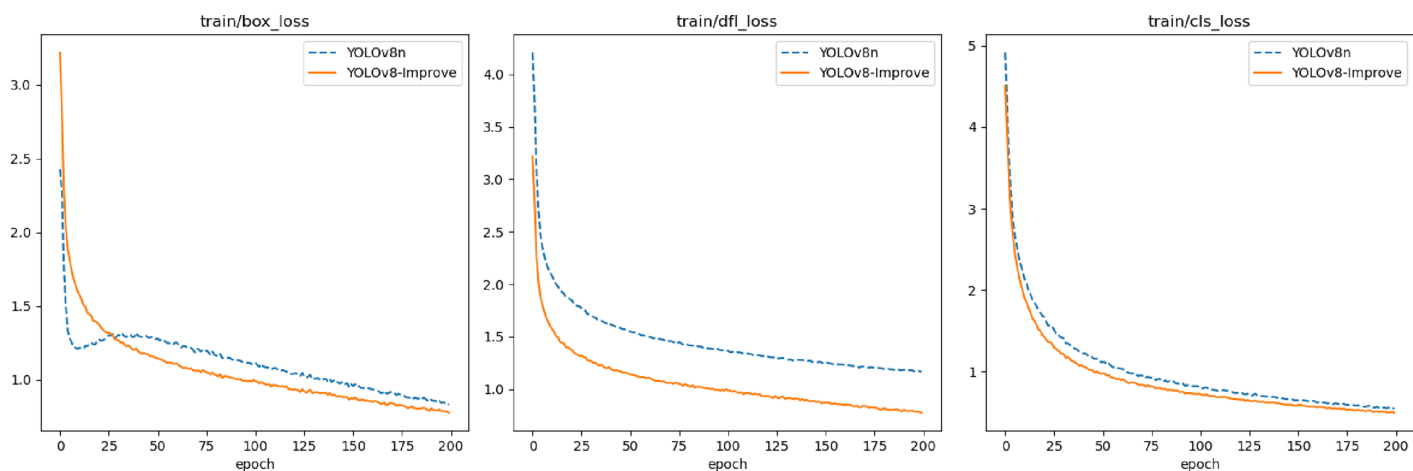
Figure 16 Curve of loss functions of YOLOv8n and YOLOv8-Improve.

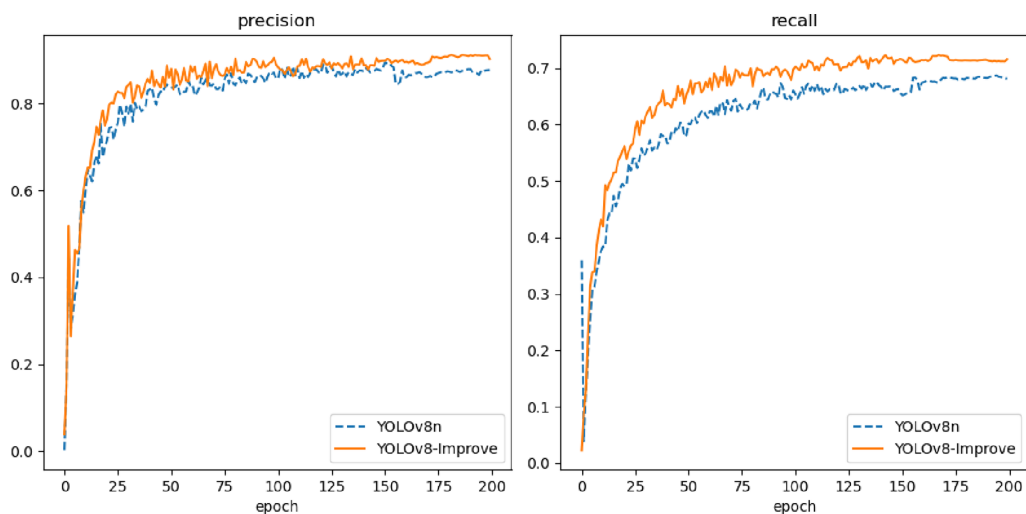Full-size ⊡ DOI: 10.7717/peerj-cs.2271/fig-16

Figure 17 Curve of precision and recall of YOLOv8n and YOLOv8-Improve.

Full-size ⊡ DOI: 10.7717/peerj-cs.2271/fig-17

To visually assess the impact of the improved algorithm on the model's detection performance, the detection confusion matrices of the baseline algorithm and the YOLOv8-Improve algorithm are compared, as illustrated in Fig. 18.

Figure 18 demonstrates that, following the application of the YOLOv8n-Improve algorithm, the recall rates for jellyfish, sea cucumbers, and octopus have increased, with the exception of starfish. Additionally, the missed detection rates for all four flexible biological targets have decreased to some extent. Both Table 3 and Fig. 18 corroborate the efficacy of the proposed algorithm.

Comparison of different models' detection results

To further assess the algorithm's effectiveness, this study compares the performance of YOLOv8-Improve with other leading object detection algorithms using a dataset of flexible
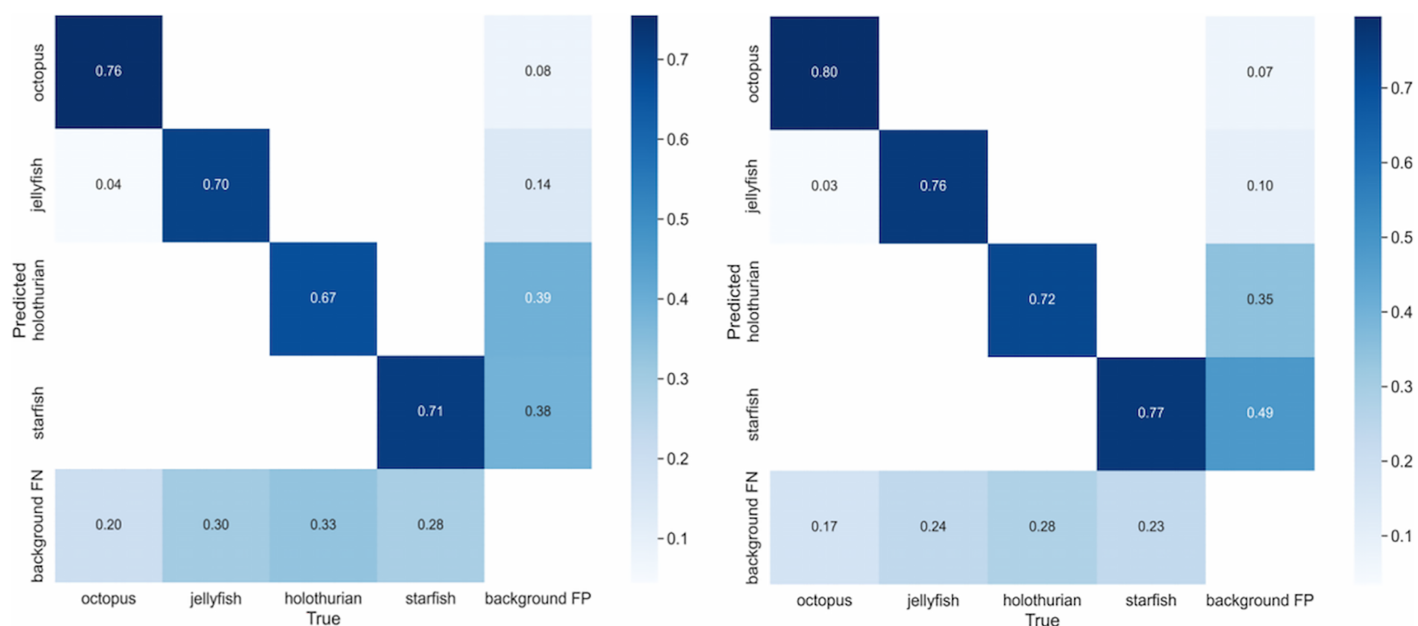
Figure 18 The YOLOv8-Improve algorithm is used to process the contrast graph of the confusion matrix.
Full-size ⊡ DOI: 10.7717/peerj-cs.2271/fig-18

Table 4 Comparison of different models' detection results.

Model	mAP^{50} (%)	Weights (M)	GFLOPs	Single image detection time (ms)
SSD	71.5	97.6	273.4	22.1
YOLOv5s	77.5	14.4	16.0	12.1
YOLOv7-tiny	77.1	12.5	13.2	14.8
Faster-RCNN	75.1	109.5	370.2	25.2
YOLOv8n	75.9	6.3	8.2	7.2
RT-DETR	80.3	32.8	110.2	15.8
Literatures (*Chen et al., 2023b*)	80.2	16.6	20.5	26.8
YOLOv8n-Improve	80.5	7.6	9.1	9.1

marine biological entities. The comparisons, detailed in Table 4, show that YOLOv8n-Improve offers comparable accuracy to the newly proposed Compared with the newly-proposed RT-DETR (*Lv, Xu & Zhao, 2023*) and the target detection algorithm for Marine environment design in literature (*Lei, Tang & Li, 2022*), they have similar accuracy, but YOLOv8-Improve has faster detection speed. Against the two-stage algorithm Faster-RCNN, YOLOv8n-Improve enhances the mAP^{50} (%) by 5.4%, with significant reductions in single-image detection time, Weights (M), and GFLOPs. The SSD algorithm underperforms on this dataset, achieving the lowest detection accuracy of 71.5% and subpar results in other metrics. Additionally, YOLOv8n-Improve is benchmarked against
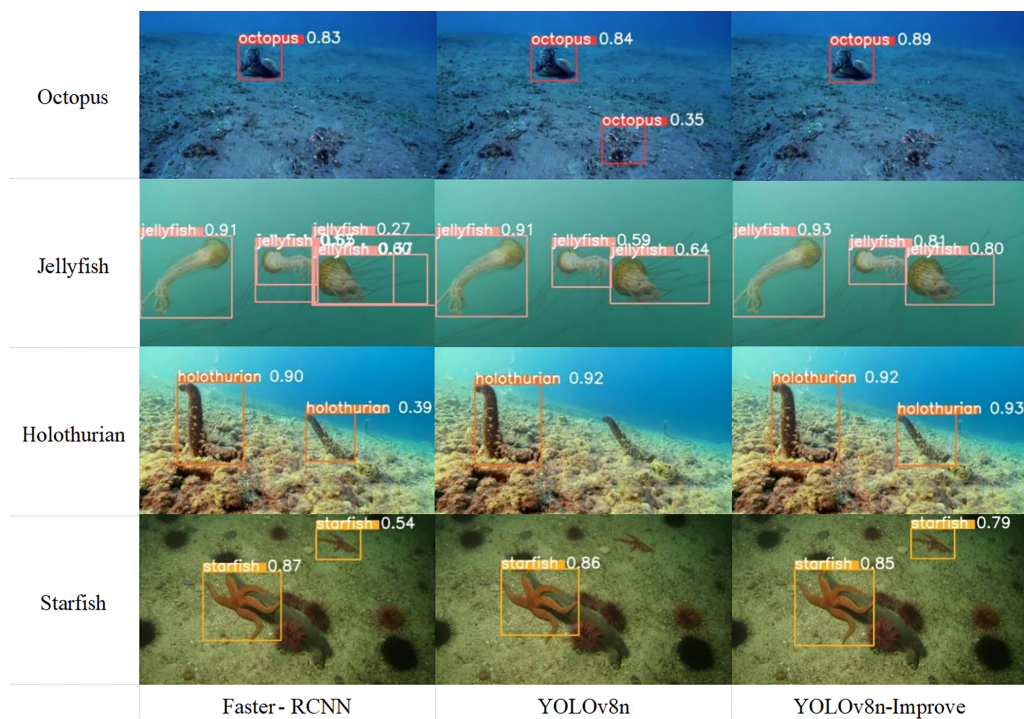
Figure 19 Visual comparison of different models' detection results.
Full-size ☒ DOI: 10.7717/peerj-cs.2271/fig-19

lighter YOLO series networks like YOLOv5n, YOLOv7-tiny, and YOLOv8n. Table 4 reveals that YOLOv8n-Improve's mAP^{50} (%) is 80.5%, which is 4.6% higher than the unenhanced YOLOv8n, with minimal impact on detection speed. The model's detection rate of 109.8 fps surpasses the 60 fps threshold, suitable for real-time scene detection. It shows slight improvements in Weights (M) and GFLOPs over YOLOv7-tiny and YOLOv5s, with mAP^{50} (%) gains of 3.4% and 3.0%, respectively. Furthermore, its single-image detection time of 9.1 ms is more efficient than YOLOv7-tiny's 14.8 ms and YOLOv5s's 12.1 ms. In summary, YOLOv8n-Improve demonstrates commendable detection performance.

To more vividly illustrate the algorithm's detection capabilities, this study compares the detection outcomes of Faster-RCNN, YOLOv8n, and the developed YOLOv8n-Improve for various marine biological entities. Figure 19, which displays the bounding boxes and confidence levels for detections of octopus, jellyfish, starfish, and sea cucumber, shows that YOLOv8n-Improve outperforms the other models in terms of bounding box accuracy and confidence. Specifically, YOLOv8n occasionally misses smaller objects like starfish and sea cucumbers. Moreover, in scenes with multiple objects or significant object deformation, Faster-RCNN and YOLOv8n sometimes produce false detections, evident in the jellyfish and octopus instances. These observations affirm the superior detection efficacy of YOLOv8n-Improve on flexible marine organisms.

CONCLUSIONS

This study addresses the challenge of detecting flexible marine biological objects, which are characterized by their soft bodies, ease of deformation, large deformation scales, and varied postures. Initially, a dataset specific to these organisms was created. Subsequently, the CLAHE underwater image enhancement algorithm was employed, augmented with a module to enhance foreground and background boundaries for image processing. Considering the unique traits of these organisms, the YOLOv8 algorithm was selected as the baseline for object detection experiments, leading to the development of an enhanced YOLOv8-based detection algorithm. This enhancement includes a deformable convolution module to improve the feature extraction network's ability to model the geometric variations of the organisms. Additionally, the RepBi-PAN network structure was incorporated to bolster the feature aggregation capability of the algorithm, while the SimAM attention mechanism was integrated to refine feature processing and reduce hardware demands. The introduction of the WIoU bounding box loss function aims to mitigate the effects of poor-quality labels. Although the proposed algorithm exhibits a slight decrease in detection speed compared to the baseline, it maintains adequate real-time performance and achieves a 4.6% increase in detection accuracy, demonstrating its effectiveness for detecting flexible marine biological entities.

ADDITIONAL INFORMATION AND DECLARATIONS

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Yu Tian conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.
- Yanwen Liu conceived and designed the experiments, authored or reviewed drafts of the article, and approved the final draft.
- Baohang Lin analyzed the data, prepared figures and/or tables, and approved the final draft.
- Peng Li conceived and designed the experiments, performed the experiments, analyzed the data, authored or reviewed drafts of the article, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The data is available at figshare: Tian, Yu (2024). Dataset.zip. figshare. Dataset. https://doi.org/10.6084/m9.figshare.24447487.v1.

REFERENCES

Cao S, Zhao D, Liu X, Sun YP. 2020. Real-time robust detector for underwater live crabs based on deep learning. *Computers and Electronics in Agriculture* **172(4)**:105339 DOI 10.1016/j.compag.2020.105339.

Chen Q. 2023. *Underwater target detection method based on the deep learning research*. Beijing: Beijing Institute of Petrochemical Industry.

Chen L, Yin T, Zhou S, Yi G, Fan D, Zhao J. 2023a. Underwater target detection algorithm based on feature fusion enhancement. *Electronics* **12(13)**:2756 DOI 10.3390/electronics12132756.

Chen X, Yuan M, Yang Q, Yao H, Wang H. 2023b. Underwater-YCC: underwater target detection optimization algorithm based on YOLOv7. *Journal of Marine Science and Engineering* **11(5)**:995 DOI 10.3390/jmse11050995.

Dai J, Qi H, Xiong Y, Zhang G, Hu H, Wei Y. 2017. Deformable convolutional networks. In: *Proceedings of the IEEE International Conference on Computer Vision. 2017*. Piscataway: IEEE, 764–773 DOI 10.1109/ICCV.2017.89.

Dong JN, Yang M, Xie ZR, Cai LP. 2022. Review of underwater image target detection data sets and detection algorithms. *Journal of Marine Technology* **41(5)**:60–72.

Girshick R, Donahue J, Darrell T, Malik J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *IEEE Computer Society* **2014**:580–587 DOI 10.1109/CVPR.2014.81.

Guan Q, Chen Y, Wei Z, Heidari AA, Hu HG, Yang XH, Zheng JW, Zhou QW, Chen HL, Chen F. 2022. Medical image augmentation for lesion detection using a texture-constrained multichannel progressive GAN. *Computers in Biology and Medicine* **145**:105444 DOI 10.1016/j.compbiomed.2022.105444.

He K, Sun J, Tang X. 2020. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33(12)**:2341–2353 DOI 10.1109/CVPR.2009.5206515.

Huang D, Wang Y, Song W, Sequeira J, Mavromatis S. 2018. Shallow-water image enhancement using relative global histogram stretching based on adaptive parameter acquisition. *MultiMedia Modeling* **10704(13)**:453–465 DOI 10.1007/978-3-319-73603-7_37.

Jiang P, Ergu D, Liu F, Cai Y, Ma B. 2022. A review of yolo algorithm developments. *Procedia Computer Science* **199(11)**:1066–1073 DOI 10.1016/j.procs.2022.01.135.

Lei F, Tang F, Li S. 2022. Underwater target detection algorithm based on improved YOLOv5. *Journal of Marine Science and Engineering* **10(3)**:310 DOI 10.3390/jmse10030310.

Li C, Guo C, Ren W, Cong RM, Hou JH, Kwong S, Tao DC. 2019. An underwater image enhancement benchmark dataset and beyond. *IEEE Transactions on Image Processing* **29**:4376–4389 DOI 10.1109/TIP.2019.2955241.

Li B, Liu P. 2023. ADFireNet: an anchor-free smoke and fire detection network based on deformable convolution. *Sensors* **23(16)**:7086 DOI 10.3390/s23167086.

Lin S, Zhao Y. 2020. Review on key technologies of target exploration in underwater optical images. *Laser & Optoelectronics Progress* **57(6)**:060002 DOI 10.3788/LOP57.060002.

Liu Q, Gong X, Li J, Wang H, Liu R, Liu D, Zhou R, Xie T, Fu R, Duan X. 2023. A multitask model for realtime fish detection and segmentation based on YOLOv5. *PeerJ Computer Science* **9**:e1262 DOI 10.7717/peerj-cs.1262.

Liu XJ, Liu Y, Jiang SX. 2023. DCN-YOLOv5 underwater target detection based on SimAM attention mechanism [J/OL]. *Journal of Chongqing Technology and Business University (Natural Science Edition)* **37(5)**:1–9 DOI 10.13382/j.jemi.B2205968.

Lv WY, Xu SL, Zhao Y. 2023. DETRs beat YOLOs on real-time object detection. *ArXiv preprint* DOI 10.48550/arXiv.2304.08069.

Nguyen HB, Duong VH, Tran Thi AX, Nguyen QC. 2023. Efficient keyword spotting system using deformable convolutional network. *IETE Journal of Research* **69(7)**:4196–4204 DOI 10.1080/03772063.2021.1946438.

Oh G, Lim S. 2023. One-stage brake light status detection based on YOLOv8. *Sensors* **23(17)**:7436 DOI 10.3390/s23177436.

Peng YT, Cosman PC. 2017. Underwater image restoration based on image blurriness and light absorption. *IEEE Transactions on Image Processing* **26(4)**:1579–1594 DOI 10.1109/TIP.2017.2663846.

Rashid U, Javid A, Khan AR, Liu L, Ahmed A, Khalid O, Saleem K, Meraj S, Iqbal U, Nawaz R. 2022. A hybrid mask RCNN-based tool to localize dental cavities from real-time mixed photographic images. *PeerJ Computer Science* **8(9)**:e888 DOI 10.7717/peerj-cs.888.

Raveendran S, Patil MD, Birajdar GK. 2021. Underwater image enhancement: a comprehensive review, recent trends, challenges and applications. *Artificial Intelligence Review* **54(7)**:5413–5467 DOI 10.1007/s10462-021-10025-z.

Sabri ZS, Li Z. 2021. Low-cost intelligent surveillance system based on fast CNN. *PeerJ Computer Science* **7(11)**:e402 DOI 10.7717/peerj-cs.402.

Shafiq M, Gu Z. 2022. Deep residual learning for image recognition: a survey. *Applied Sciences* **12(18)**:8972 DOI 10.3390/app12188972.

Song W, Wang Y, Huang D, Tjondronegoro D. 2018. A rapid scene depth estimation model based on underwater light attenuation prior for underwater image restoration. *Advances in Multimedia Information Processing, PT1. 2019* **1164**:678–688 DOI 10.1007/978-3-030-00776-8_62.

Tang H, Liang S, Yao D, Qiao Y. 2023. A visual defect detection for optics lens based on the YOLOv5-C3CA-SPPF network model. *Optics Express* **31(2)**:2628–2643 DOI 10.1364/OE.480816.

Ting L, Xianchun Z, Ying Z, Cai Y, Ma B. 2023. Underwater image enhancement based on IMSRCR and CLAHE-WGIF. *Instrumentation* **10(2)**:19–29 DOI 10.1016/j.procs.2022.01.135.

Wang G, Chen Y, An P, Hong H, Hu J, Huang T. 2023. UAV-YOLOv8: a small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* **23(16)**:7190 DOI 10.3390/s23167190.

Wang G, Chen Y, An P, Hong H, Hu J, Huang T. 2023. UAV-YOLOv8: a small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* **23(16)**:7190 DOI 10.3390/s23167190.

Wang Y, Song W, Fortino G, Qi LZ, Zhang WQ, Liotta A. 2019. An experimental-based review of image enhancement and image restoration methods for underwater imaging. *IEEE Access* **7**:140233–140251 DOI 10.1109/ACCESS.2019.2932130.

Wang H, Xiao N. 2023. Underwater object detection method based on improved faster RCNN. *Application Science* **13(4)**:2746 DOI 10.3390/app13042746.

Wang N, Zheng H, Zheng B. 2017. Underwater image restoration via maximum attenuation identification. *IEEE Access* 5:18941–18952 DOI 10.1109/ACCESS.2017.2753796.

Wei J, Liu G, Liu S, Xiao Z. 2023. A novel algorithm for small object detection based on YOLOv4. *PeerJ Computer Science* 9(2):e1314 DOI 10.7717/peerj-cs.1314.

Wu W, Liu H, Li L, Long YL, Wang XD, Wang ZH, Li JL, Chang Y. 2021. Application of local fully convolutional neural network combined with YOLOv5 algorithm in small target detection of remote sensing image. *PLOS ONE* 16(10):e0259283 DOI 10.1371/journal.pone.0259283.

Zhang L, Li J, Zhang F. 2023. An efficient forest fire target detection model based on improved YOLOv5. *Fire-Switzerland* 6(8):291 DOI 10.3390/fire6080291.

Zhang LQ, Liu ZT, Jiang CS. 2022. An improved SimAM based CNN for facial expression recognition. In: *Chinese Control Conference, CCC, 2022-July*. 6510–6514.

Zou Shi, Guo Y, Shi ZW, Guo YH, Ye JP. 2023. Object detection in 20 years: a survey. *Proceedings of the IEEE* 111(3):257–276 DOI 10.1109/JPROC.2023.3238524.

Tian et al. (2024), *PeerJ Comput. Sci.*, DOI 10.7717/peerj-cs.2271

27/27