Journal of Real-Time Image Processing (2024) 21:114
https://doi.org/10.1007/s11554-024-01494-w

RESEARCH

GST-YOLO: a lightweight visual detection algorithm for underwater garbage detection

Longyi Jiang[1] · Fanghua Liu[1] · Junwei Lv[1] · Binghua Liu[1] · Chen Wang[1]

Received: 15 April 2024 / Accepted: 7 June 2024 / Published online: 16 June 2024
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

Abstract
Underwater cleaning work primarily relies on human labor, but applying computer vision technology to Autonomous Underwater Vehicles can enhance cleaning efficiency. Considering that existing vision detection algorithms are difficult to deploy on resource-constrained embedded devices, this paper introduces a lightweight vision detection algorithm based on an improved YOLOv8-GST-YOLO. This algorithm integrates the lightweight Ghost network and prunes the model, overcoming the drawbacks of YOLOv8's high computational parameters and large size. It also features a GTR module and a bi-directional path aggregated feature pyramid guided by SimAM attention to enhance detection accuracy and global feature extraction capabilities. Experiments on a specially collected underwater trash image dataset show that GST-YOLO, while reducing the model size by 51% and increasing computational efficiency by 54%, improves the accuracy rate to 95.4%, surpassing the YOLOv8 algorithm. This demonstrates its potential as a crucial detection tool for underwater unmanned cleaning tasks, offering broad application prospects.

Keywords Underwater debris · Visual detection · YOLOv8 · Lightweight algorithm

1 Introduction

As human demand for plastic products continues to increase, plastic production has grown more than 22 times over several decades. However, as of 2015, the global plastic recycling rate was only 9% [7], with non-degradable plastic waste accumulating slowly while continuing to pose a significant threat to marine ecosystems [1, 13]. According to statistics from 2010, approximately 48 to 127 million tons of the 275 million tons of plastic waste generated by 192 coastal countries ultimately flowed into the oceans [11]. It is estimated that the current total amount of deep-sea garbage worldwide exceeds 142.85 million tons. This plastic waste poses a significant threat to marine ecosystems [14].

Clearly, marine cleanup efforts are urgently needed. Traditionally, underwater garbage cleanup has relied mainly on manual dredging or waiting for waves to wash it onto beaches before cleaning, a method that is not only slow to respond but may also cause secondary pollution and is difficult to comprehensively detect and clean underwater debris. We must explore a new approach to cleaning. In recent years, with the widespread application of deep learning algorithms in the field of computer vision, Convolutional Neural Networks (CNNs) have achieved significant success in object detection tasks. Unlike traditional machine learning methods that rely on fine manual feature design and extraction, deep learning algorithms can extract more abstract features without the need for human-defined features. Therefore, deep learning has gradually replaced traditional machine learning algorithms [30]. In the field of object detection, deep learning algorithms are mainly divided into one-stage and two-stage algorithms. One-stage algorithms, such as YOLO [23] (You Only Look Once) and SSD [18] (Single Shot Detector), aim to quickly detect objects at once and are suitable for real-time detection scenarios. In contrast, two-stage

✉ Fanghua Liu
15008347760@163.com

Longyi Jiang
1003000675@qq.com

Junwei Lv
2993124538@qq.com

Binghua Liu
1275315402@qq.com

Chen Wang
3785933228@qq.com

[1] Faculty of Mechanical Engineering, Jiangsu University of Science and Technology, Zhenjiang 212100, China

algorithms focus more on detection accuracy [21], such as SPP-Net [20](Spatial Pyramid Pooling Networks) and Faster R-CNN [24]. These algorithms can accurately distinguish different categories in images and precisely locate them, using bounding boxes to label each object.

Based on this, we believe that applying computer vision detection to Autonomous Underwater Vehicles (AUVs) can achieve more proactive, comprehensive, and rapid underwater garbage detection and recovery. By deploying well-trained CNN models on AUVs, real-time identification and positioning of various underwater debris can be achieved, thereby guiding AUVs to perform precise salvage tasks. Furthermore, by combining the advantages of one-stage and two-stage algorithms, hybrid models can be designed to be both fast and accurate, addressing the unique challenges of the underwater environment.

2 Related works

Constrained by complex underwater environments and the limited carrying conditions of underwater vehicles, the development of underwater garbage detection technology has been insufficient. However, in recent years, an increasing number of scholars have turned their attention to this field. In 2019, Kylili et al. [15] detected floating marine plastic debris using deep learning algorithms, created a dataset containing 4000 images, and trained it using the VGG16 model, achieving a validation accuracy of 86%. Fulton et al. [6] compared the performances of YOLOv2, Tiny-YOLO, Faster RCNN, and SSD in deep-sea plastic debris detection and found that Faster RCNN had the highest accuracy, while Tiny-YOLO, despite its lower precision, had a smaller model size and higher detection efficiency. In 2021, Politikos et al. [22] introduced a new target detection algorithm using Mask R-CNN, constructed an underwater image dataset containing 11 types of garbage categories, and obtained a 62% average precision score after training.

In 2022, Teng et al. [28] proposed an image classifier based on the YOLOv5 algorithm, targeting garbage in coastal beach areas, trained on a dataset of 2050 images, and achieved an average precision score of 89.4%. Moorton et al. [19] conducted a study to explore whether deep learning could successfully differentiate between marine life and underwater synthetic debris. They compared a simple convolutional neural network with the VGG-16 model, classifying aquatic life and underwater garbage using 1644 images. The study concluded that the potential to safely remove underwater debris without harming the marine ecosystem using computer image detection is substantial. Additionally, Sannigrahi et al. [25] developed an automatic detection system for marine floating plastic using high-resolution satellite images and advanced machine learning models, while Kako et al. [12] utilized drones based on deep learning to survey the amount of plastic waste on beaches, demonstrating the potential to estimate the quantity of beach garbage.

These studies not only highlight the potential applications of computer vision technology in the field of marine debris processing but also indicate the direction for future research, including further improvements in algorithm performance, enhancing the model's generalization capabilities, and developing more efficient training strategies, laying the foundation for broader applications in marine environments. Overall, intelligent detection based on deep learning provides an effective solution for underwater garbage cleaning and recovery, promising to significantly alleviate the issue of marine plastic pollution. Future research should focus on improving algorithm performance, enhancing the generalization ability of models, and developing more efficient training strategies.

This study addresses the current challenges in underwater garbage detection using computer vision and proposes a lightweight visual detection algorithm, GST-YOLO. The main contributions of this paper are as follows:

(1) By adopting the lightweight convolution GhostConv to replace the standard convolution in YOLOv8 and removing the C2f module in the Backbone part, this study streamlined the network layers and parameters, achieving model lightweighting. This improvement enables the algorithm to be effectively deployed on devices with limited computing power.
(2) Introducing the Transformer framework from the field of natural language processing and combining it with the lightweight convolution GhostConv, a CNN+Transformer hybrid module, GTR, was designed. This module greatly enhances feature extraction capability while maintaining low computational complexity and small model volume, reducing the demand for computational resources without sacrificing model accuracy.
(3) A Bidirectional Pathway Aggregated Feature Pyramid structure guided by the SimAM attention mechanism, GSBiFPN, was designed, replacing the FPN-PANet structure originally used by YOLOv8. This innovative approach from the perspective of feature fusion solves the problem of low detection accuracy in lightweight networks.
(4) A dataset containing hard-to-degrade garbage in complex underwater environments was organized and constructed, and the new algorithm was validated using this dataset, addressing the current challenges of underwater garbage detection.

The structure of this paper is as follows: Sect. 3 provides a detailed description of the dataset construction process and

the basic framework and key structures of the GST-YOLO algorithm; Sect. 4 describes the testing of the algorithm, comparative experimental processes, analysis, results, and evaluation; Sect. 5 discusses the experimental results in a comprehensive manner based on existing research; and Sect. 6 summarizes the research findings and provides an outlook on future research directions.

3 Materials and methods

In this section, we will describe the experimental dataset required for this study and the strategy for algorithm improvement.

3.1 Dataset construction

Our dataset images are primarily sourced from the Japan Agency for Marine-Earth Science and Technology (JAMSTEC). They encompass a variety of environmental conditions, depths, and illumination levels, fully meeting our criteria for constructing a high-quality dataset. After the selection process, our dataset includes, but is not limited to, underwater non-biodegradable waste such as plastic bags, plastic bottles, and masks, as illustrated in Fig. 1. In selecting the data, we focused on the diversity of the images and gave priority to those showing trash in areas frequently visited by marine life. This ensures that the trained detection network can effectively differentiate between underwater life and debris, fulfilling our goal of removing non-biodegradable waste and protecting the marine ecosystem.

In terms of dataset annotation, since our main objective is to identify non-biodegradable waste, we did not categorize the types of debris. Hence, it is uniformly labeled as "trash_plastic" in the dataset. This simplified labeling approach not only reduces the computational demands for algorithm training but also aids in enhancing training efficiency. Through selection, the final constructed dataset for underwater non-biodegradable garbage comprises 4750 images. These images are partitioned into training, validation, and testing sets in a ratio of 8:1:1.

3.2 Algorithm improvement

The GST-YOLO algorithm proposed in this study mainly makes improvements based on YOLOv8.

3.2.1 YOLOv8 network architecture

The overall structure of the YOLOv8 algorithm is divided into three main parts: the feature extraction Backbone network, the feature fusion Neck network, and the Head network, which is responsible for object classification and location regression, as shown in Fig. 2. It inherits the gradient bifurcation concept from its predecessors, employs downsampling through convolution, and switches to the C2f module for feature extraction (replacing the previously used C3 module). The features extracted by the Backbone are fused in the Neck through upsampling, forming a three-level feature pyramid rich in feature information. This structure adopts the FPN-PANet [17] concept, maintaining multi-level information of the feature maps, which achieves a good balance between detection accuracy and speed.

3.2.2 Improved algorithm GST-YOLO

The YOLOv8 algorithm's complex feature extraction network, primarily composed of CBS, C2f, and SPPF modules, leads to a significant number of parameters and relatively slower detection speed. This presents a notable obstacle in applications requiring lightweight yet high-precision underwater target detection. To address this, we introduce a lightweight detection algorithm based on an improved YOLOv8

Fig. 1 Dataset annotation

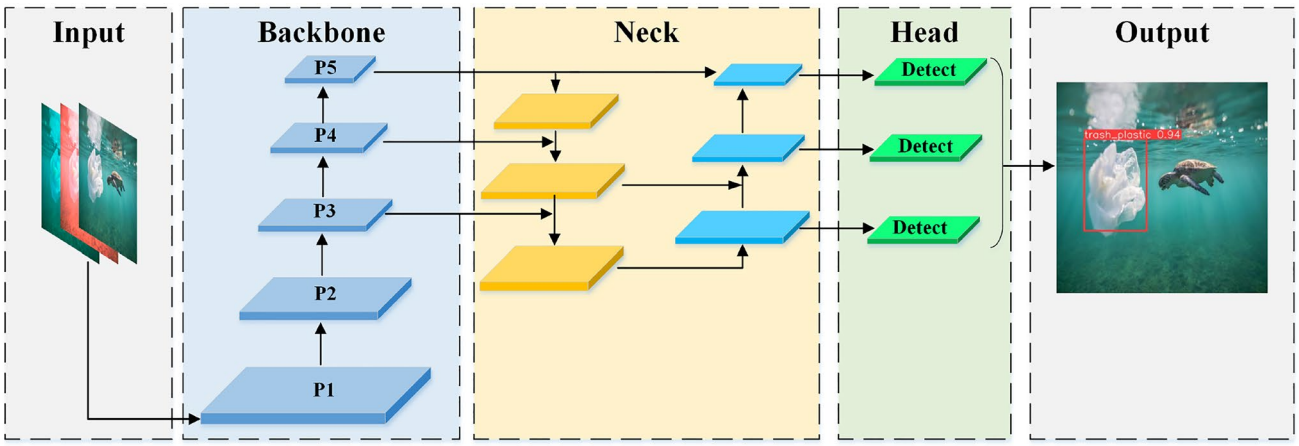
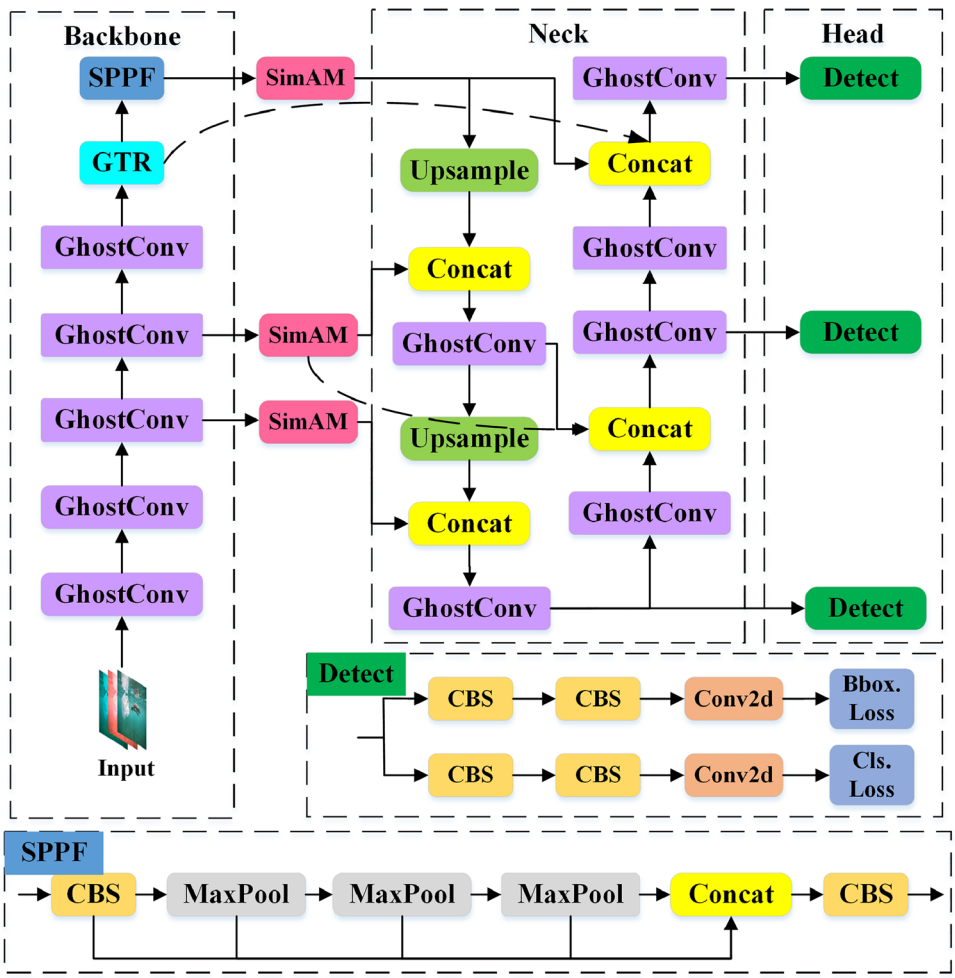


Fig. 2 YOLOv8 network architecture

- GST-YOLO, aiming to balance lightweight design with high detection accuracy (as shown in Fig. 3).

Firstly, we replace the CBS module in YOLOv8 with a lightweight feature extraction module, GhostConv, and eliminate the C2f module from the Backbone. This modification aims to reduce the number of parameters, thereby effectively lowering computational costs and enhancing the real-time performance of underwater target detection. Secondly, we introduce a GTR module that combines CNN and Transformer, placed at the end of the

Fig. 3 GST-YOLO



Backbone network, to ensure the model maintains high detection accuracy while being lightweight. Lastly, we design a feature pyramid structure, GSBiFPN, guided by SimAM attention, replacing the FPN-PANet structure used in the original YOLOv8. This innovative feature fusion approach addresses the issue of low detection accuracy in lightweight networks.

In GST-YOLO, prior to training, the model subjects image data to comprehensive augmentation and undergoes model fine-tuning to ensure its robustness and versatility across diverse and complex scenarios, thereby enhancing its performance. Upon entry into the backbone network, features are extracted from shallow to deep layers, and a three-dimensional dynamic attention mechanism is employed to concentrate on target features within the shallow, intermediate, and deep layers of the network, thereby reducing the influence of background interference and generating discriminative features to aid in better discerning the differences between targets and backgrounds. Subsequently, these features are fed into GSBiFPN for multi-scale feature fusion, ensuring the model's adept response to targets of varying sizes and proportions by extracting features at different network levels and amalgamating them. The Head module receives feature maps from the Neck, further merging and integrating information from diverse levels, and performs object detection predictions based on these amalgamated features. Detailed structures and principles of each module will be elaborated upon in subsequent sections.

3.2.3 Lightweight modules

Traditional convolution modules (CBS, as shown in Fig. 4a) tend to generate many redundant feature maps during feature extraction. However, experiments have proven that these feature maps play a crucial role in retaining comprehensive information from the input. In contrast, GhostConv aims to address this issue with a lower computational cost. Its core idea is not to avoid generating these feature maps but to produce new Ghost feature maps [8] based on an existing set of feature maps by performing a series of cost-effective linear transformations (as detailed in Fig. 4b). These maps are capable of effectively capturing the original feature information.

3.2.4 Attention mechanism

The fundamental concept of attention mechanism is to actively focus on key information during the feature extraction process, thereby reducing the influence of background interference. Its approach involves enhancing the feature extraction capability of the network model on specific spatial locations and channels. Current attention mechanisms include spatial attention, which focuses on important locations in feature maps, and channel attention, which emphasizes channels containing crucial information, with representative methods being DCN [2] and CA [9], respectively. There are also hybrid attention mechanisms that combine both, such as DANet [5] and CCNet [10].

While these attention methods have significantly improved the recognition accuracy of deep learning models,

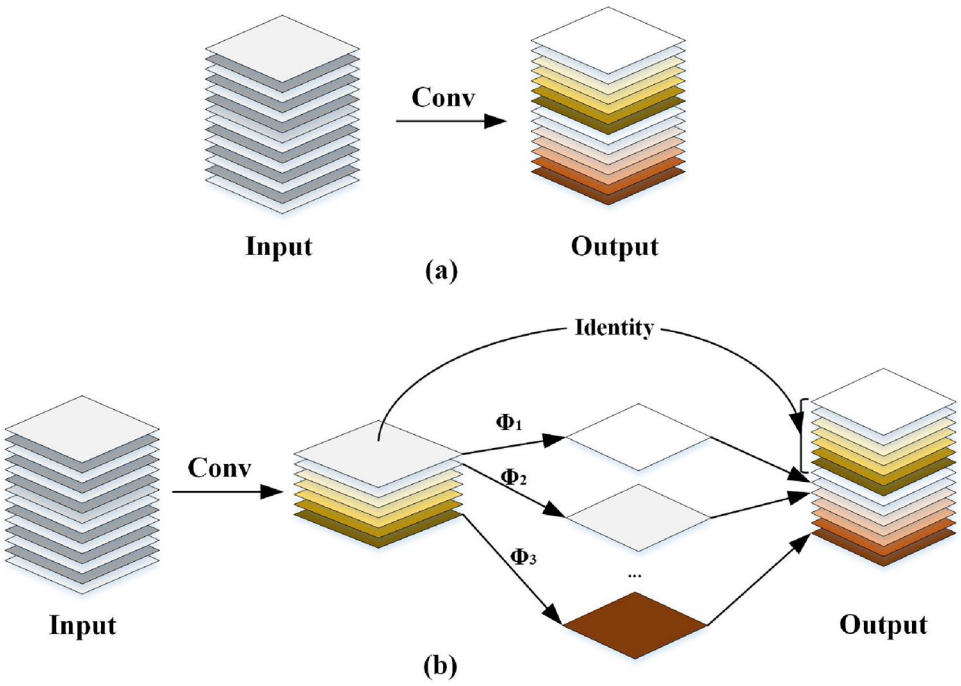


Fig. 4 Structures of convolutional basic units. (**a**) Convolution structure. (**b**) GhostConv structure

they refine features along either the channel or spatial dimensions, as shown in Fig. 5. This approach somewhat limits the model's flexibility in learning attention weights across channel and spatial variations.

SimAM [29] represents an innovative attention mechanism that, unlike previous methods, can directly infer three-dimensional attention weights (encompassing both spatial and channel dimensions) in the feature layer without adding extra parameters to the original network, as illustrated in Fig. 6, where different colors represent different channels. In essence, SimAM achieves a performance enhancement beyond traditional spatial and channel attention mechanisms at a lower computational cost.

3.2.5 Improvements in the backbone

In the Backbone section, YOLOv8 employs a pattern where CBS and C2f modules are nested to extract a multi-layer feature structure from shallow to deep. This design significantly enhances the model's feature extraction capability but also leads to model size expansion and a surge in computational load due to the repeated use of CBS and C2f modules. To address this issue, we replace the CBS module with the GhostConv module and remove the C2f module, subsequently introducing the SimAM attention mechanism to guide the feature flow towards the Neck for multi-scale feature fusion, as shown in Fig. 7, effectively reducing the Backbone's volume. This improvement strategy not only significantly reduces the model's volume but also maintains its excellent feature extraction and information fusion capabilities. Consequently, the model is better equipped to comprehend the differences between targets and backgrounds, ensuring its robustness and versatility across various real-world scenarios.

To enhance detection accuracy, at the end of the Backbone, we combine CNNs with Transformers and, incorporating the concept of multi-scale feature fusion, design the feature enhancement module GTR, as depicted in Fig. 8.

Originally designed for natural language processing tasks, the Transformer [26] introduces the multi-head self-attention mechanism, enabling effective long-distance dependency handling and efficient parallel computation. Building on this architecture, Dosovitskiy et al. [4] proposed the Vision Transformer, applying the Transformer to two-dimensional image processing, achieving notable results. In the domain of image processing, by segmenting images into patches and encoding and decoding these patches, interactions between image blocks are enabled, providing remote dependencies and global context associations, significantly enhancing target detection performance, as illustrated in Fig. 9.

In the feature enhancement module GTR, we apply an identity mapping to a portion of the input feature maps, while generating corresponding redundant feature maps through linear operations for another part. After feature concatenation, these features are further transformed into small block patches through a Split operation and processed

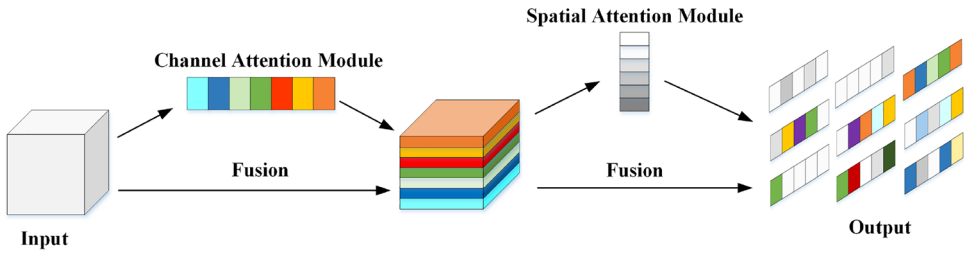


Fig. 5 Traditional channel spatial attention

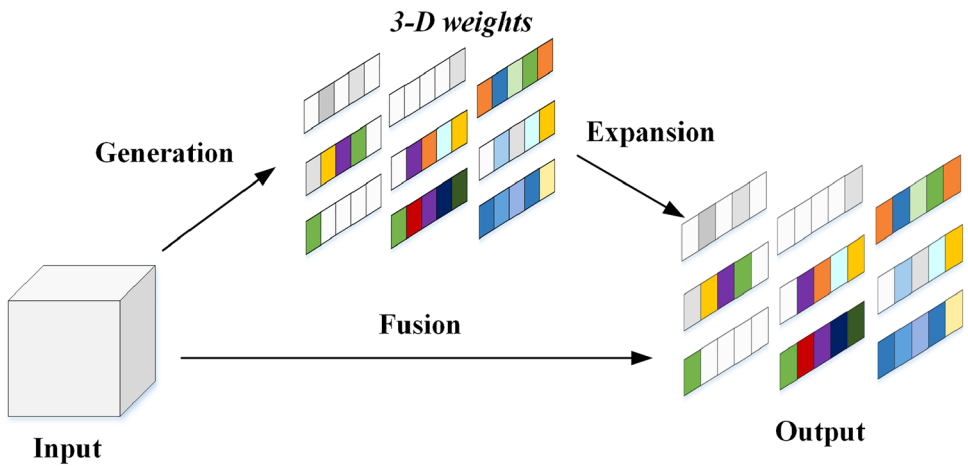


Fig. 6 SimAM attention

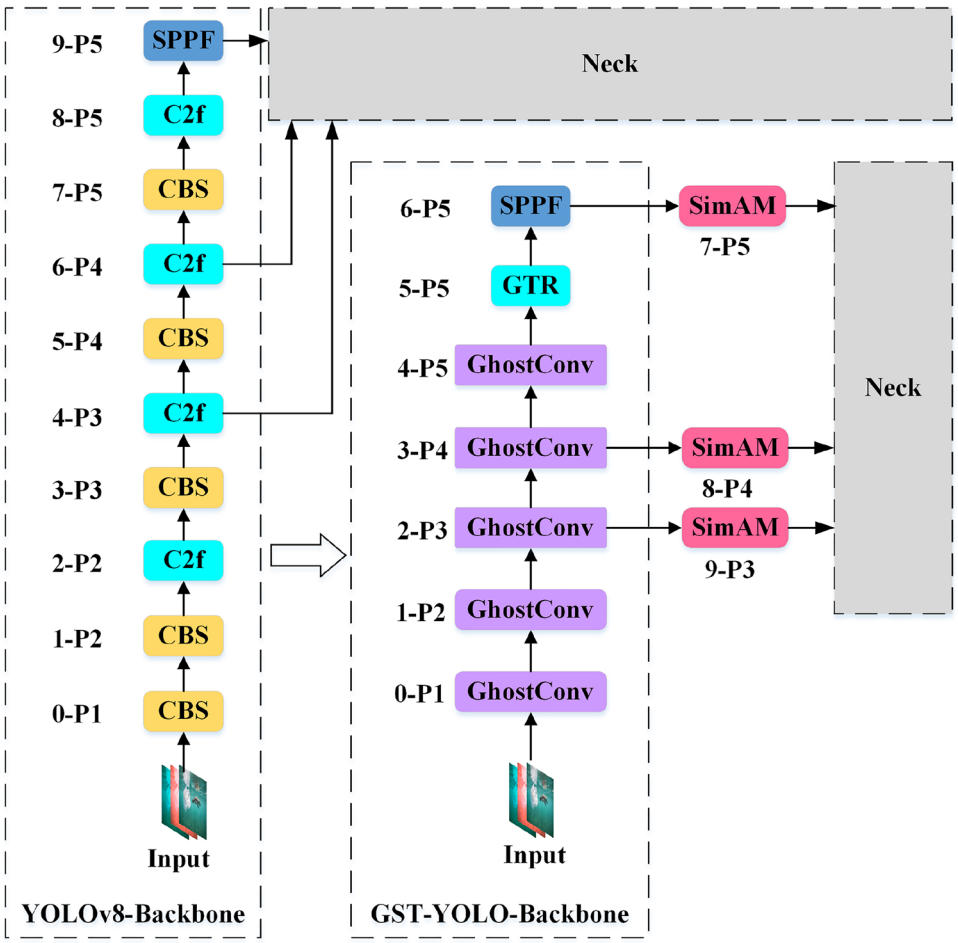


Fig. 7 Backbone improvement

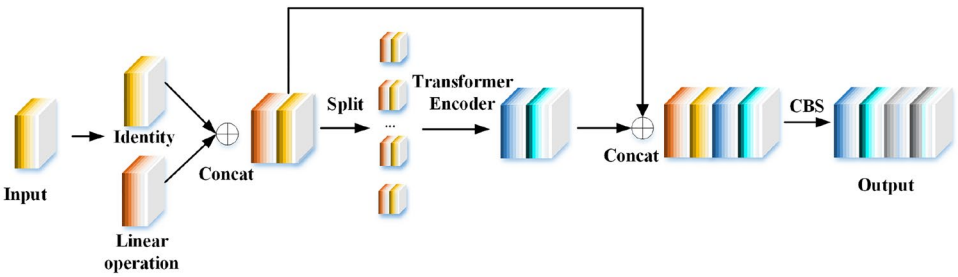


Fig. 8 GTR module

via Transformer encoding, then concatenated back with the original feature maps. In this manner, the GTR module leverages the Transformer's self-attention mechanism to effectively capture long-distance dependencies between different positions in the image. A subsequent standard convolution focuses on local features, enriching the feature information.

The core design of the GTR module involves splitting the image features into multiple sequential blocks with vertical associations, encoding each block, and extracting image features based on the correlations between blocks. The self-attention mechanism is one of the core components of this process, used to compute the weights of each element in the input sequence, thereby capturing the internal dependencies of the sequence. In the self-attention mechanism, each element interacts with other elements in the sequence, obtaining a weighted representation, where the weight of each element is determined by its relationship with other elements. The computation formula for the self-attention mechanism is as follows:

$$Attention(Q, K, V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right)V \quad (1)$$

where Q, K, and V represent matrices of Query, Key, and Value, with d_k representing the dimensionality of the keys.

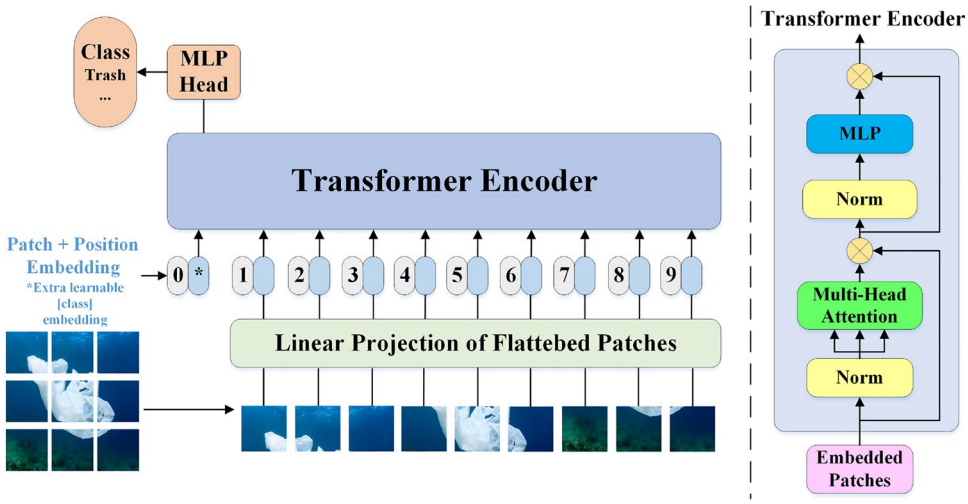


Fig. 9 Vision transformer

The softmax function transforms each element into a value between 0 and 1, indicating the weight.

The formulas for calculating Q, K, and V are as follows:

$$(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \mathrm{MatMul}(\mathrm{X}, (\mathrm{W}^{\mathrm{Q}}, \mathrm{W}^{\mathrm{k}}, \mathrm{W}^{\mathrm{V}})) \tag{2}$$

where X represents the input sequence, and W^Q, W^K, W^V are the weight matrices of the model. The specific computations of Q, K, and V are based on the multiplication of X with these weight matrices.

Additionally, between each layer of the encoder and decoder, there's a fully connected Multi-Layer Perceptron (MLP), used for non-linear transformation and feature extraction. The computation formula for the MLP is as follows:

$$\mathrm{FFN}(\mathrm{x}) = \mathrm{ReLU}(\mathrm{xW}_1 + \mathrm{b}_1)\mathrm{W}_2 + \mathrm{b}_2 \tag{3}$$

where x is the input vector, W_1 and W_2 are the weight matrices, b_1 and b_2 are the bias vectors, and ReLU stands for the Rectified Linear Unit activation function.

3.2.6 Guided SimAM bi-directional feature pyramid network

In deep learning algorithms, as the depth of the network increases, the extracted features become more hierarchical. Typically, shallow features are rich in detail information, capturing minor local changes in images, but lack a deep understanding of the overall semantics. Conversely, deep features are richer in expressing semantic levels, understanding the overall structure and semantic information of images, but have weaker detail expression capabilities [3]. Therefore, effectively extracting and fusing multi-level image features is crucial to improving the accuracy of detection algorithms.

Considering that the same target may exhibit different features at different scales, the Feature Pyramid Network [16] (FPN) adopts a bottom-up and top-down structure to build a feature pyramid, as shown in Fig. 10a, thereby achieving multi-scale feature acquisition and fusion, and supporting target detection or semantic segmentation at various scales. The Path Aggregation Network (PANet) further expands on this concept, introducing lateral paths, top-down aggregation paths, and bottom-up aggregation paths, as shown in Fig. 10b. The combination of these three paths enables the

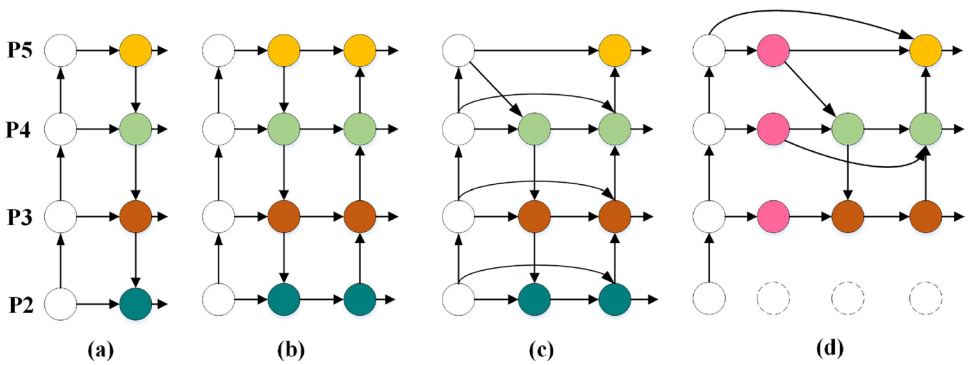


Fig. 10 Feature fusion networks. (**a**) FPN structure. (**b**) PANet structure. (**c**) BiFPN structure. (**d**) ours GSBiFPN

network to fuse information from feature maps of different scales at each level, adapting to targets of various sizes. The Bi-Directional Feature Pyramid Network [27](BiFPN), through residual connections and weighted fusion, optimizes the semantic interaction and scale contribution adjustment between features, enhancing the processing capability for targets of different scales, as shown in Fig. 10c.

In response to the insufficient cross-scale interaction and the degradation of semantic information for small targets in YOLOv8s multi-scale feature fusion, we propose a novel feature pyramid structure with pruning optimization. We introduce a Guided SimAM Bi-Directional Feature Pyramid Network (GSBiFPN) module, led by the SimAM attention mechanism and considering cross-layer connections and weighted feature fusion, as shown in Fig. 10d. Based on shallow, medium, and deep feature information, we construct bi-directional cyclic paths and cross-scale connections. Through the attention mechanism-guided weighted feature fusion technique, we not only emphasize the importance of feature information from different pathways but also enable the model to continuously focus on feature extraction in the target region at various scales. This approach achieves effective fusion of multi-scale feature information while maximizing the network's feature extraction capability, ensuring that the model can effectively handle disturbances such as overlap and occlusion, and produce robust responses to targets of different sizes and proportions. The specific implementation of this approach is illustrated in Fig. 11.

3.3 Experiment

To validate the practical effectiveness of our improved algorithm, we established an experimental platform using the PyTorch framework for deep learning. The hardware configurations and training environment for the experimental process are detailed in Table 1.

The experiment will employ two algorithms, YOLOv8 and GST-YOLO, for training and iteration on the previously mentioned dataset. Throughout the entire experimental process, apart from the algorithm differences, all other configurations will remain identical to ensure the validity of the comparative experiment and validate the superiority of the improved algorithm.

4 Results

4.1 Evaluation parameter

The evaluation of the algorithmic network utilizes two metrics: intersection over union (IoU) and mean average precision (mAP). IoU measures the degree of match between the predicted bounding box and the target location, that is, the ratio of the intersection to the union of the predicted box and the actual box. The value of IoU ranges from 0 to 1, and typically, if IoU is greater than a set threshold (e.g., 0.5), the detection result is considered correct. Specifically, during data analysis, we calculate the IoU of each predicted box output by the model

Table 1 Hardware configuration and training environment

Hardware configuration	Description	Software/library	Version
CPU	Intel(R) i7-12700 H	Operating System	Windows11
GPU	RTX 3060	PyTorch	1.7.1
Memory	16GB	CUDA	11.0

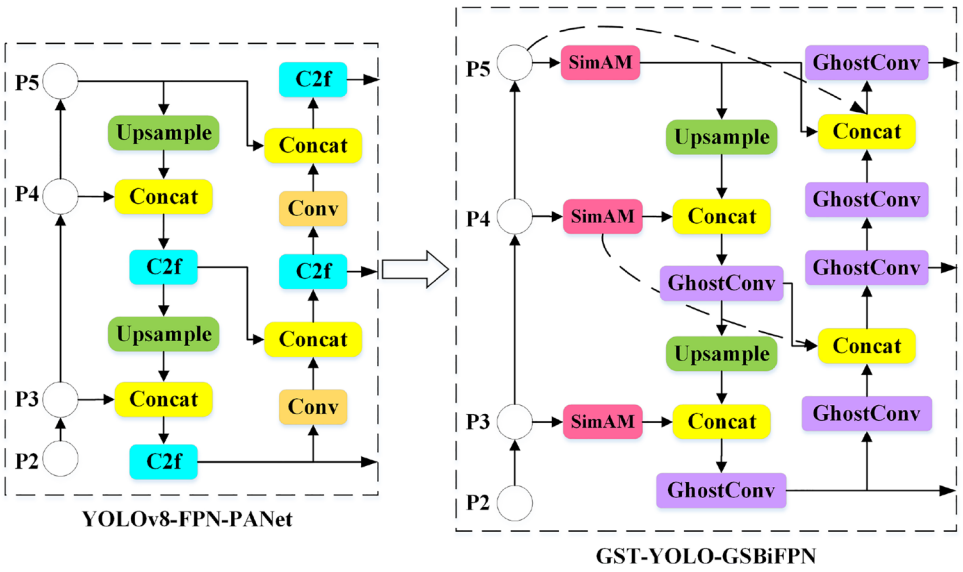


Fig. 11 Feature fusion enhancement

with all actual boxes and take the maximum value as MaxIoU. If MaxIoU is still less than the predefined threshold (usually 0.5), the predicted box is marked as a false positive (FP). If MaxIoU is greater than the predefined threshold, it indicates that there is a corresponding actual box for that box. At this time, if the predicted box and the actual box belong to the same category, the box is marked as a true positive (TP). If not, the box is also deemed an FP.

FN (False Negative): The number of instances incorrectly classified as negative, meaning they are predicted as negative but are actually positive.FP (False Positive): The number of instances incorrectly classified as positive, meaning they are predicted as positive but are actually negative.TN (True Negative): The number of instances correctly classified as negative, meaning they are predicted as negative and are indeed negative.

R (Recall), or the recall rate, indicates the proportion of actual positive samples in the predicted samples to all the actual samples; P (Precision), or precision rate, represents the proportion of actual positive samples in the predicted samples to all the predicted positive samples. AP (Average Precision) is calculated based on the area under the Precision-Recall curve; mAP (mean Average Precision) refers to the sum of the average precision of all categories divided by the number of categories. Here are their formulas:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{AP} = \int_0^1 P \mathrm{d}R \tag{6}$$

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

4.2 Training result

Based on the dataset annotated as described above and the training parameters configured, we conducted training on the model and selected YOLOv8s as the baseline for comparison. The training results are illustrated in Fig. 12.

From the precision curve comparison in Fig. 12a, we can observe that the precision of the GST-YOLO algorithm gradually surpasses that of the YOLOv8s algorithm with the increase in training steps. The specific surpassing value, as seen from the comparison in Table 2, is 2.1%. Achieving such an enhancement is quite significant, especially when the precision rate has already exceeded 90%. Figure 12b shows the corresponding precision-recall curve, where the area under the curve represents the average precision (AP). It is evident from the graph that the AP of the GST-YOLO algorithm is higher than that of YOLOv8s.

The advantages of the GST-YOLO algorithm over the original model are not limited to this. From the comparison in Table 2, we can see that the number of training parameters

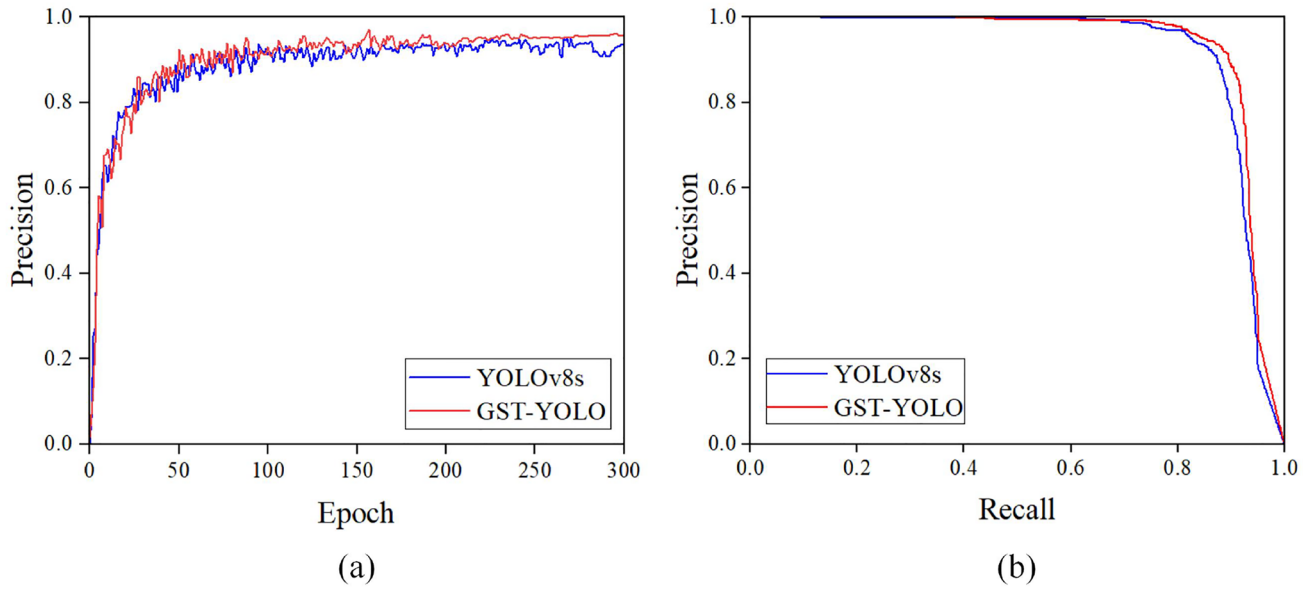


Fig. 12 Comparison of GST-YOLO and YOLOv8s training evaluation. (**a**) Precision curve. (**b**) Precision-recall curve

of the GST-YOLO algorithm has been halved, the model weight has also been reduced to half of the original, and the computational efficiency (GFLOPs) has been greatly improved (more than doubled), meeting the requirements for deployment on Autonomous Underwater Vehicles (AUVs). The experimental results fully affirm the correctness of our algorithm improvement theory.

As illustrated by the detection samples in Fig. 13, the GST-YOLO algorithm demonstrates high robustness and versatility across various real-world scenarios. In Fig. 13a–d, even with the presence of distractions (e.g., marine life, humans, or artificial devices) and occlusions, the algorithm network is capable of correctly identifying waste based on extracted features, thereby avoiding misidentifications and effectively preventing harm to marine life during underwater operations.

4.3 Ablation experiment

To delve into the impact of each improvement module on the overall performance of the algorithm and to verify the comprehensive effect of the improved algorithm, we conducted ablation experiments to compare the results of algorithm performance under different network structure influences, as shown in Table 3.

First, we carried out a maximized light-weighting experiment, applying the GhostConv module to fully replace similar modules, that is, completely replacing the Backbone and Neck layers of YOLOv8. As expected, we inferred that the model would be minimized in size, but there would be a significant drop in accuracy. Surprisingly, the experimental results exceeded our expectations. We underestimated the tri-scale fusion structure of YOLO itself and the feature extraction capability of the GhostConv module. The application of the GhostConv module not only reduced the model's size and the number of parameters but also had a relatively minor impact on accuracy. The cumulative effect of multiple GhostConv modules far exceeded their individual use. As shown in Table 3, the accuracy only decreased by 12.4%, but the model's parameters were reduced by more than half, and the training speed significantly increased. The final model

Table 2 Comparison of GST-YOLO and YOLOv8s Networks

Network	Layers	Parameters	FLOPs/G	Weight/M	P/%	R/%	mAP/%
YOLOv8s	168	11125971	28.4	21.4	93.3	88.6	93.1
GST-YOLO	136	5478739	13.1	10.6	95.4	84.5	93.9

Fig. 13 Test sample

Table 3 Ablation Experiment

GhostConv	GTR	SimAM	GSBiFPN	P/%	mAP50/%	Params	FLOPs/G	Inference time (ms)	Weight/M
				93.3	93.1	11.1×10^6	28.4	11.4	21.4
✓				80.9	75.2	4.56×10^6	13.0	7.0	8.6
✓	✓			90.2	86.3	5.23×10^6	13.2	9.3	9.4
✓		✓		91.3	88.5	5.12×10^6	13.2	7.6	9.2
✓	✓	✓		92.5	92.3	5.60×10^6	13.6	9.0	10.9
✓	✓	✓	✓	95.4	93.9	5.47×10^6	13.1	8.9	10.6

file size was only 40% of the original, successfully reducing the weight by 60%.

Next, we individually added the newly designed GTR module to the light-weighted Backbone layer, incorporating it into the fifth layer of multi-scale fusion, which is the bottom layer. This addition aimed to enhance the extraction of detailed features. The results showed that compared to the original YOLOv8 model, the accuracy only decreased by 3.1%, while the number of parameters decreased by 53%. Then, we added the SimAM module individually to the Backbone layer, also at the fifth layer of multi-scale fusion. The experimental results matched expectations: compared to the original YOLOv8 model, the accuracy only decreased by 2.0%, while the parameter count was reduced by 54%. Finally, when adding both GTR and SimAM modules to the Backbone, compared to the original YOLOv8 model, the accuracy only decreased by 0.8%, while the parameter count was reduced by 50%.

Following our theoretical framework, we constructed the GSBiFPN bidirectional path-aggregated feature pyramid and established the GST-YOLO model. The experimental results were encouraging: compared to YOLOv8, our GST-YOLO achieved a 2.1% increase in accuracy, a 51% reduction in the number of parameters, a 54% improvement in computational speed, and the final model file size was only 49% of the original, successfully reducing the volume by 51%.

To delve into the feature target localization ability of each network during training, we provide the attention maps of each network's training, as shown in Fig. 14. From Fig. 14a, it can be seen that in the original YOLOv8 network, there is no apparent focus of attention; its feature extraction is scattered and comprehensive, resulting in high precision of training results but also a large number of parameters and computational load. Figure 14b shows the network structure where YOLOv8 is fully replaced with GhostConv, which similarly lacks a distinct focus during feature extraction.

In Fig. 14c, where we incorporated the SimAM attention module, there is a clear focus in feature extraction, akin to the human brain's focus during learning. This improves learning efficiency, but the focus still encompasses a significant portion of non-target objects. Figure 14d combines the GTR module, which weakens the attention on non-target objects. Figure 14e presents our GST-YOLO network, which maintains focused learning on target objects without excessively concentrating on non-targets. This ensures a reduction in the number of parameters and computational load during training while maintaining strong learning capabilities for target objects, ensuring high precision in the final training outcomes.

The objective of this study is to design an algorithm that boasts high detection accuracy, a small model size, and fast detection speed. Figure 15 illustrates the superior performance of the algorithm more clearly, showcasing that GST-YOLO has high detection accuracy, short inference time, and a compact model size.

5 Discussion

In 2019, Fulton et al. compared four deep learning algorithms (YOLOv2, Tiny-YOLO, Faster RCNN, and SSD) for their effectiveness in detecting marine debris, especially plastic waste. This is considered one of the earliest studies in this field. At that time, limited by the development of deep learning algorithms, the study merely demonstrated the feasibility of applying deep learning algorithms for detection and deployment on Autonomous Underwater Vehicles (AUVs), without providing detailed information about the model size, as shown in the first four entries of Table 4. This research benefits from the advancements in deep learning algorithms over the years, achieving significant breakthroughs in detection accuracy and efficiency. To further illustrate the excellence of our model, we compared it with cutting-edge detection algorithms (YOLOv5, YOLOv6, YOLOv7, YOLOv8) using the dataset proposed in this paper. The data is presented in Table 4, which demonstrates that our GST-YOLO algorithm has surpassed previous research in terms of P, mAP, Params, and FPS. Furthermore, we compared the occlusion and overlap detection capabilities of several major models (Faster R-CNN, YOLOv7, YOLOv8,

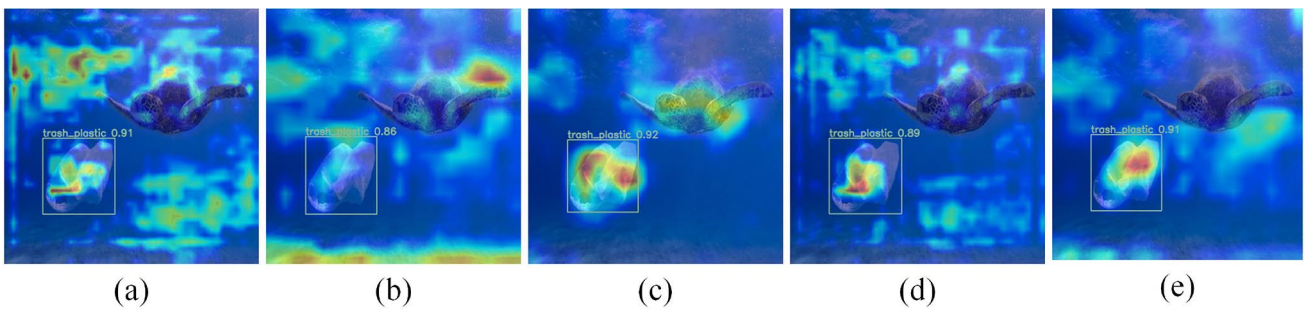


Fig. 14 Attention map

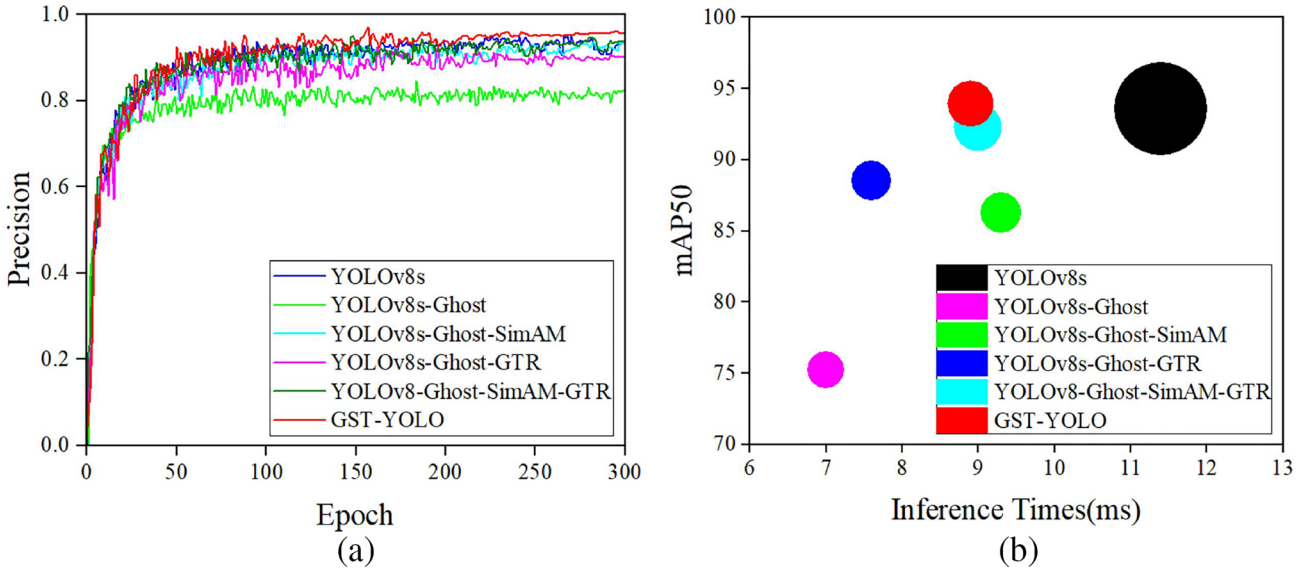


Fig. 15 Ablation study evaluation. (**a**) Precision curve. (**b**) Comprehensive model comparison (Color block size represents model size)

Table 4 Network comparison

Network	P/%	mAP/%	Params/M	FPS
YOLOv2 [6]	82.3	47.9	–	74
Tiny-YOLO [6]	70.3	31.6	–	205
Faster R-CNN [6]	83.3	81.0	–	18.75
SSD [6]	69.8	67.4	–	25.2
YOLOv5	92.5	92.8	7.2	89
YOLOv6	91.4	91.0	18.5	87
YOLOv7	92.7	92.1	31.4	93
YOLOv8	93.3	93.1	11.2	101.1
GST-YOLO	95.4	93.9	5.47	136.1

GST-YOLO). The results, shown in Fig. 16, indicate that GST-YOLO exhibits comprehensive detection of occlusions and overlaps, with only a few instances of missed detections for small and edge objects. This highlights its suitability for underwater garbage detection applications.

6 Conclusion and future work

In this paper, we propose a novel strategy to address the current challenges in underwater waste cleanup, namely the application of computer vision detection to Autonomous Underwater Vehicles (AUVs) for the active and unmanned detection and retrieval of marine debris. To validate the feasibility of this strategy, we selected images from an open-source database to establish a validation dataset targeting non-biodegradable waste such as plastic bags, bottles, and masks. We then chose the YOLOv8 algorithm as our baseline model and refined it for lightweight and precision, creating a detection network that is both accurate and lightweight, named GST-YOLO, suitable for direct deployment on AUVs or embedded devices. We trained and validated GST-YOLO using our dataset and the assessment results are encouraging, surpassing previous research. Compared to the original model, GST-YOLO achieved a 51% reduction in training parameters, a 49% decrease in model weight, a 54% improvement in computational efficiency (GFLOPs), a 2% increase in precision, and a 0.6% rise in mAP. Based on these results, we believe GST-YOLO is fully capable of being deployed on AUVs or embedded devices for underwater waste cleaning.

While the current results are promising, relying solely on our network model is insufficient to clean up all underwater debris. Our dataset is still limited in representing the full scope of underwater waste, stemming from our overall limited exploration of the ocean. However, we are optimistic that as time progresses, more underwater debris imagery will be captured, leading to a more comprehensive dataset. We envision GST-YOLO as an essential component of future underwater waste cleaning systems to address underwater pollution challenges.

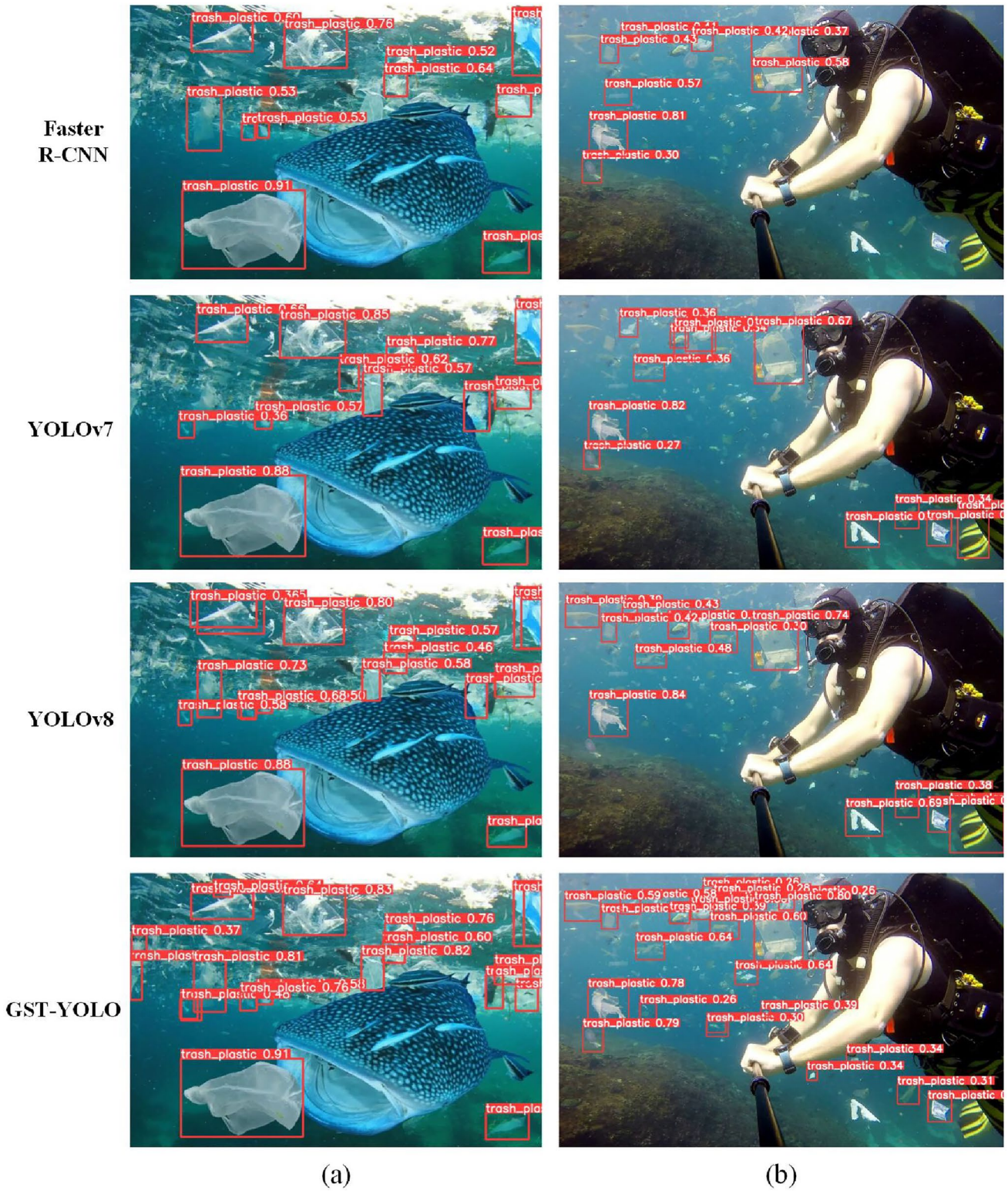


Fig. 16 Contrast experiment

Author Contributions Contributions Statement This research project was collaboratively completed by the following authors, with their contributions as follows: Longyi, Jiang designed the main framework of

the research and was responsible for leading the experiments and core data analysis. He drafted the initial manuscript and performed the final editing. Fanghua, Liu participated in the research of the main framework and was responsible for the final review of the initial manuscript draft. Junwei, Lü was in charge of data collection and processing, proposed key improvements to the data analysis methods, and participated in writing the manuscript. Binghua, Liu handled the maintenance and calibration of the experimental instruments and equipment and contributed to the creation of the graphics. Chen, Wang was responsible for the literature review, providing comprehensive theoretical support for the background of the study. All authors have read and agreed to the final submission of this manuscript. Each author confirms that their contributions to this paper meet the authorship criteria of the Journal of Real-Time Image Processing.

Data Availability No datasets were generated or analysed during the current study.

Conflict of interest The authors declare no Conflict of interest.

Code and Dataset Availability The code and dataset used in this paper will be provided according to the journal's requirements and can be accessed through the following link:https://github.com/yxyy67/GST-YOLO.

References

1. Chamas, A., Moon, H., Zheng, J., Qiu, Y., Tabassum, T., Jang, J.H., Abu-Omar, M., Scott, S.L., Suh, S.: Degradation rates of plastics in the environment. ACS Sustain Chem Eng **8**(9), 3494–3511 (2020). https://doi.org/10.1021/acssuschemeng.9b06635
2. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: 16th IEEE International Conference on Computer Vision (ICCV), pp. 764–773. Ieee, NEW YORK (2017). https://doi.org/10.1109/iccv.2017.89
3. Deng, H., Zhang, Y.: Fmr-yolo: Infrared ship rotating target detection based on synthetic fog and multiscale weighted feature fusion. IEEE Trans. Instrum. Meas. **73**, 1–17 (2024). https://doi.org/10.1109/TIM.2023.3336445
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020). https://doi.org/10.48550/arXiv.2010.11929
5. Fu, J., Liu, J., Tian, H.J.: Dual attention network for scene segmentation. In: 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3141–3149. Ieee Computer Soc, LOS ALAMITOS (2019). https://doi.org/10.1109/cvpr.2019.00326
6. Fulton, M., Hong, J., Islam, M.J., Sattar, J.: Robotic detection of marine litter using deep visual detection models. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 5752–5758 (2019). https://doi.org/10.1109/ICRA.2019.8793975
7. Geyer, R., Jambeck, J.R., Law, K.L.: Production, use, and fate of all plastics ever made. Sci. Adv. **3**(7), e1700782 (2017). https://doi.org/10.1126/sciadv.1700782
8. Han, K., Wang, Y., Tian, Q.: Ghostnet: More features from cheap operations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1577–1586. Ieee, NEW YORK (2020). https://doi.org/10.1109/cvpr42600.2020.00165
9. Hou, Q.B., Zhou, D.Q., Feng, J.S., Ieee Comp, S.O.C.: Coordinate attention for efficient mobile network design. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13708–13717. Ieee Computer Soc, LOS ALAMITOS (2021). https://doi.org/10.1109/cvpr46437.2021.01350
10. Huang, Z., Wang, X., Huang, L.: Ccnet: Criss-cross attention for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **45**(6), 6896–6908 (2023). https://doi.org/10.1109/tpami.2020.3007032
11. Jambeck, J.R., Geyer, R., Wilcox, C.: Plastic waste inputs from land into the ocean. Science **347**(6223), 768–771 (2015). https://doi.org/10.1126/science.1260352
12. Kako, S., Morita, S., Taneda, T.: Estimation of plastic marine debris volumes on beaches using unmanned aerial vehicles and image processing based on deep learning. Mar. Pollut. Bull. **155**, 111127 (2020). https://doi.org/10.1016/j.marpolbul.2020.111127
13. Kremezi, M., Kristollari, V., Karathanassi, V.: Increasing the sentinel-2 potential for marine plastic litter monitoring through image fusion techniques. Mar. Pollut. Bull. **182**, 19 (2022). https://doi.org/10.1016/j.marpolbul.2022.113974
14. Kuhn, S., van Franeker, J.A.: Quantitative overview of marine debris ingested by marine megafauna. Mar. Pollut. Bull. **151**, 110858 (2020). https://doi.org/10.1016/j.marpolbul.2019.110858
15. Kylili, K., Kyriakides, I., Artusi, A., Hadjistassou, C.: Identifying floating plastic marine debris using a deep learning approach. Environ. Sci. Pollut. Res. Int. **26**(17), 17091–17099 (2019). https://doi.org/10.1007/s11356-019-05148-4
16. Lin, T.Y., Dollár, P., Girshick, R., He, K.M., Hariharan, B., Belongie, S., Ieee: Feature pyramid networks for object detection. In: 30th IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 936–944. Ieee, NEW YORK (2017). https://doi.org/10.1109/cvpr.2017.106
17. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8759–8768. Ieee, NEW YORK (2018). https://doi.org/10.1109/cvpr.2018.00913
18. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: 14th European Conference on Computer Vision (ECCV), vol. 9905, pp. 21–37. Springer International Publishing Ag, CHAM (2016). https://doi.org/10.1007/978-3-319-46448-0_2
19. Moorton, Z., Kurt, Z., Woo, W.L.: Is the use of deep learning an appropriate means to locate debris in the ocean without harming aquatic wildlife? Mar. Pollut. Bull. **181**, 113853 (2022). https://doi.org/10.1016/j.marpolbul.2022.113853
20. Msonda, P., Uymaz, S.A., Karaagaç, S.S.: Spatial pyramid pooling in deep convolutional networks for automatic tuberculosis diagnosis. Traitement Du Signal **37**(6), 1075–1084 (2020). https://doi.org/10.18280/ts.370620
21. Patnaik, S.K., Babu, C.N., Bhave, M.: Intelligent and adaptive web data extraction system using convolutional and long short-term memory deep learning networks. Big Data Mining Anal. **4**(4), 279–297 (2021). https://doi.org/10.26599/BDMA.2021.9020012
22. Politikos, D.V., Fakiris, E., Davvetas, A., Klampanos, I.A., Papatheodorou, G.: Automatic detection of seafloor marine litter using towed camera images and deep learning. Mar. Pollut. Bull. **164**, 10 (2021). https://doi.org/10.1016/j.marpolbul.2021.111974
23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788. Ieee, NEW YORK (2016). https://doi.org/10.1109/cvpr.2016.91
24. Ren, S.Q., He, K.M., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2017). https://doi.org/10.1109/tpami.2016.2577031
25. Sannigrahi, S., Basu, B., Basu, A.S., Pilla, F.: Development of automated marine floating plastic detection system using

sentinel-2 imagery and machine learning models. Mar. Pollut. Bull. **178**, 113527 (2022). https://doi.org/10.1016/j.marpolbul.2022.113527
26. Srinivas, A., Lin, T.Y., Parmar, N., Shlens, J., Abbeel, P., Vaswani, A.: Bottleneck transformers for visual recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 16514–16524. Ieee Computer Soc, LOS ALAMITOS (2021). https://doi.org/10.1109/cvpr46437.2021.01625
27. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10778–10787 (2020). https://doi.org/10.1109/CVPR42600.2020.01079
28. Teng, C., Kylili, K., Hadjistassou, C.: Deploying deep learning to estimate the abundance of marine debris from video footage. Mar. Pollut. Bull. **183**, 114049 (2022). https://doi.org/10.1016/j.marpolbul.2022.114049
29. Yang, L., Zhang, R.Y., Li, L., Xie, X.: Simam: A simple, parameter-free attention module for convolutional neural networks (2021)
30. Zuo, C., Feng, S., Zhang, X., Han, J.: Deep learning based computational imaging: status, challenges, and future. Acta Opt Sin **40**, 11003 (2020). https://doi.org/10.3788/AOS202040.0111003

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Longyi Jiang is a master's student currently pursuing his degree at the School of Mechanical Engineering at Jiangsu University of Science and Technology. His main research focus is on underwater target identification and control of underwater robots. He has conducted in-depth research in the control and identification technologies of underwater robots.

Fanghua Liu is a master's supervisor and currently holds a position at the School of Mechanical Engineering at Jiangsu University of Science and Technology. Her research interests are primarily in mechatronics and control, multibody dynamics, and the research and development of robotic technologies. Professor Liu has published more than 10 SCI/EI papers and has participated in 2 major national projects and 15 provincial and ministerial projects.

Junwei Lv is a master's student currently studying at the School of Mechanical Engineering at Jiangsu University of Science and Technology. His research is focused on the development and application of pipeline dredging robots, particularly in robot design and performance optimization.

Binghua Liu is a master's student currently studying at the School of Mechanical Engineering at Jiangsu University of Science and Technology. His main research direction is hybrid and parallel rehabilitation robots, especially in-depth studies on the control systems and human-machine interaction technologies of rehabilitation robots.

Chen Wang is a master's student currently studying at the School of Mechanical Engineering at Jiangsu University of Science and Technology. His research area is mainly on the recovery device platforms for underwater submersibles, aiming to improve the efficiency and safety of submersible recovery operations.