Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.054188

ARTICLE

Ghost-YOLO v8: An Attention-Guided Enhanced Small Target Detection Algorithm for Floating Litter on Water Surfaces

Zhongmin Huangfu, Shuqing Li*[*] and Luoheng Yan

School of Information Engineering, North China University of Water Resources and Electric Power, Zhengzhou, 450046, China

*Corresponding Author: Shuqing Li. Email: lishuqing@stu.ncwu.edu.cn

Received: 21 May 2024 Accepted: 27 July 2024 Published: 12 September 2024

ABSTRACT

Addressing the challenges in detecting surface floating litter in artificial lakes, including complex environments, uneven illumination, and susceptibility to noise and weather, this paper proposes an efficient and lightweight Ghost-YOLO (You Only Look Once) v8 algorithm. The algorithm integrates advanced attention mechanisms and a small-target detection head to significantly enhance detection performance and efficiency. Firstly, an SE (Squeeze-and-Excitation) mechanism is incorporated into the backbone network to fortify the extraction of resilient features and precise target localization. This mechanism models feature channel dependencies, enabling adaptive adjustment of channel importance, thereby improving recognition of floating litter targets. Secondly, a 160 × 160 small-target detection layer is designed in the feature fusion neck to mitigate semantic information loss due to varying target scales. This design enhances the fusion of deep and shallow semantic information, improving small target feature representation and enabling better capture and identification of tiny floating litter. Thirdly, to balance performance and efficiency, the GhostConv module replaces part of the conventional convolutions in the feature fusion neck. Additionally, a novel C2fGhost (CSPDarknet53 to 2-Stage Feature Pyramid Networks Ghost) module is introduced to further reduce network parameters. Lastly, to address the challenge of occlusion, a new loss function, WIoU (Wise Intersection over Union) v3 incorporating a flexible and non-monotonic concentration approach, is adopted to improve detection rates for surface floating litter. The outcomes of the experiments demonstrate that the Ghost-YOLO v8 model proposed in this paper performs well in the dataset Marine, significantly enhances precision and recall by 3.3 and 7.6 percentage points, respectively, in contrast with the base model, mAP@0.5 and mAP@0.5:0.95 improve by 5.3 and 4.4 percentage points and reduces the computational volume by 1.88 MB, the FPS value hardly decreases, and the efficient real-time identification of floating debris on the water's surface can be achieved cost-effectively.

KEYWORDS

YOLO v8; surface floating litter; target detection; attention mechanism; small target detection head; ghostnet; loss function

1 Introduction

In the context of advancements in industrial and agricultural sectors, the release of considerable quantities of wastewater and waste has imparted substantial harm to the ecological integrity of watersheds. Key contributors to water pollution include surface floaters [1], water pollution itself [2], and eutrophication of water bodies. These factors not only contaminate and deteriorate water resources, but also present a serious hazard to human safety and wellbeing. Therefore, the immediate detection and subsequent removal of floating litter from water surfaces is of paramount importance in achieving sustainable green development.

With the wide application of deep neural network architectures [3] in fields such as image processing [4] and target detection [5] deep learning techniques have been rapidly developed. Among them, YOLO (You Only Look Once) v8 [6], the latest iteration of the YOLO algorithm family, excels in target detection, classification, and instance segmentation tasks, as a classical real-time target detection algorithm, shows significant application potential in target detection with its high accuracy and efficiency. For this paper, we have selected the smallest yet most accurate variant, YOLO v8n, for target detection. However, the complexity and variability of the water surface environment, such as changes in weather conditions can cause images to be blurred, foamed, or incompletely displayed, which in turn hinders the model's proficiency in extracting and accurately recognizing features of the target, while noise interference causes the model to have difficulty in accurately extracting features of the target by interfering with the useful information in the image. The process of identifying floating trash on water using the YOLO v8 algorithm is hindered by issues like misidentifications and failures to detect. These challenges pose constraints on the ability of the algorithm to consistently deliver accurate and trustworthy results in real-world applications, and thus further research and improvements are needed to overcome these problems.

Addressing the aforementioned issues, this paper introduces a lightweight Ghost-YOLO v8 water surface floating litter detection model that fuses the attention mechanism and a specialized small target detection unit to overcome challenges of low precision and bloated model dimensions. in the detection of water surface floating litter by traditional networks. The principal contributions of this research paper are listed here:

1. Incorporating the SE (Squeeze-and-Excitation) [7] attention module within the backbone network enhances its capability for feature extraction and improves the model's precision in target localization., improving the model's anti-interference capability and robustness.
2. Introducing a 160 × 160 pixel small target detection layer, the combination of deep semantic information and shallow semantic information is strengthened, and the ability of small target feature expression is improved.
3. Incorporate the Ghostconv module [8] and design a new C2fGhost (CSPDarknet53 to 2-Stage Feature Pyramid Networks Ghost) module, with the aim of significantly decreasing the network's parameter count to enhance the network model's operational speed and achieve a lightweight effect.
4. Use WIoU (Wise Intersection over Union) v3 [9] , a flexible and non-monotonic concentration approach instead of CIoU (Complete Intersection over Union) [10] in the original model as the model localization loss function for bounding boxes, overcoming the challenge of low detection rates for waterborne litter, hindered by obstruction, and enhancing the model's detection capabilities.

2 Related Work

With the advent of deep learning, a CNN (Convolutional Neural Network) [11] based approach for water surface floating litter detection has emerged. This method, thanks to its robust feature extraction capabilities and straightforward rule-setting, offers an effective and reliable means of identifying and classifying floating litter by extracting, training, and learning the salient characteristics of such debris. Target detection algorithms leveraging CNNs are broadly categorized into two: two-stage algorithms, exemplified by R-CNN (Region-CNN) [12], Fast R-CNN [13], and Faster R-CNN

[14], and single-stage algorithms, such as the YOLO [15] series and SSD (Single Shot MultiBox Detector) [16]. The research on single-stage detection algorithms primarily focuses on enhancing model structure and optimizing data preprocessing techniques.

In the realm of optimizing the model structure for single-stage target detection algorithms, several advancements have been made. Gai et al. [17] proposed to improve YOLO-V4 algorithm to detect cherries by adding DenseNet network combined with CSPDarknet53, optimizing the a priori frame for circular markers, and enhancing feature extraction and a framework for cherry fruit detection, optimized for both speed and accuracy. Wang et al. [18] proposed UAV-YOLO v8 (UAV-YOLO) model to optimize UAV aerial target detection with Wise-IoU (Intersection over Union) v3 to improve positioning accuracy, BiFormer to enhance the key information attention, and Focal FasterNet block fusion features to significantly reduce the small target leakage rate and improve the detection performance. Yuan et al. [19] addressed the limitations of CIoU by employing the GCBlock structure, the innovative GSConv convolution method, and the SIoU (Soft Intersection over Union) loss function. This strategy enhanced feature extraction, reduced computational requirements, and raised detection accuracy. Qiao et al. [20] introduced the coordinate attention mechanism, enabling the network to capture broader contextual information. By utilizing the BiFPN module in lieu of the FPN (Feature Pyramid Network) module, weighted feature fusion and efficient bidirectional cross-scale connectivity were achieved, improving the detection precision of tiny and very small floating litter. Lin et al. [21] bolstered the feature extraction capabilities of the lightweight target detection backbone through the addition of a FMA (Feature Mapping Attention) layer, alongside data augmentation techniques and an extended training dataset, thereby enhancing the model's generalizability. Hou et al. [22] augmented the model's generalization capabilities by introducing the HorBlock module and employing a genetic algorithm for evolution. Additionally, the extension of the underwater dataset via offline data augmentation bolstered feature extraction, ultimately improving the model's detection accuracy and speed in complex environments.

In terms of data preprocessing improvement for single-stage target detection algorithms, Li et al. [23] proposed an adaptive underwater image enhancement method RGHS (Relative Global Histogram Stretching), which improves the YOLO v5 model by introducing a convolutional triple-attention mechanism module, and improves the ability to detect minute objects. Yue et al. [24] augmented the Flow-Img dataset to avoid overfitting, incorporated a dedicated layer for detecting tiny targets into YOLO v5s, removed the large target detection head, and introduced CBAM (Convolutional Block Attention Module) to enhance target feature capture, combining the IoU (Intersection over Union) with the NWD (Normalized Wasserstein Distance) loss function, to enhance the exactness and reliability of the water surface floating bottle recognition. Li et al. [25] cropped and optimised the YOLO v5s, and added a specific image preprocessing module, and embedded the SOC (System On Chip) embedded in a network camera to achieve edge computing deployment, which is capable of real-time processing of video data. Lei et al. [26] compared various deep learning algorithms by migration learning on collected floating object data and concluded that SSD is better than other algorithms in terms of accuracy, while Faster R-CNN is more detailed in terms of prediction details. Yang [27] used meanshift algorithm to segment the image and extracted the features of colour distance of floats and energy decomposition coefficients after grey scale wavelet transform processing to achieve the estimation of information such as the type and degree of pollution. Chen et al. [28] introduced a method for detecting surface floating objects utilizing a persistent unsupervised domain adaptation technique. This method enhances the feature extraction capability for small-scale floating objects by eliminating low-resolution feature maps and augmenting high-resolution maps. Li et al. [29] formulated a river floating litter dataset sourced from UAV aerial images and introduced targeted enhancements to the YOLO v5s target detection algorithm, addressing issues such as category imbalance and diminished accuracy in small target detection. Additionally, Li et al. [30] developed a small-sample water surface floating debris dataset model grounded in AlexNeT [31]. By employing gradient descent for model fine-tuning and incorporating fused light correction, the method effectively identifies common pollutants, achieving a nearly 15% improvement in recognition rate.

3 Methods

The network architecture of YOLO v8n comprises four key components: the input layer for data augmentation, the backbone responsible for image feature extraction, the neck-end for multi-scale feature fusion, and the decoupled output header that separates classification and detection functions. The schematic representation of this algorithm's model structure is depicted in Fig. 1. Within this architecture, The Conv block executes convolution operations, followed by BN (Batch Normalization) and the SiLU activation function on the incoming image. By efficiently capturing residual features, the C2f module empowers YOLO v8n to retain rich gradient details while keeping its design lightweight. The SPPF (Spatial Pyramid Pooling Fast) module transforms feature maps of varying sizes into fixed-sized feature vectors, thereby reducing computational costs. Finally, the detect module, comprising a classifier and regressor, forecasts the location and classification of the target, drawing upon the neck's output features.

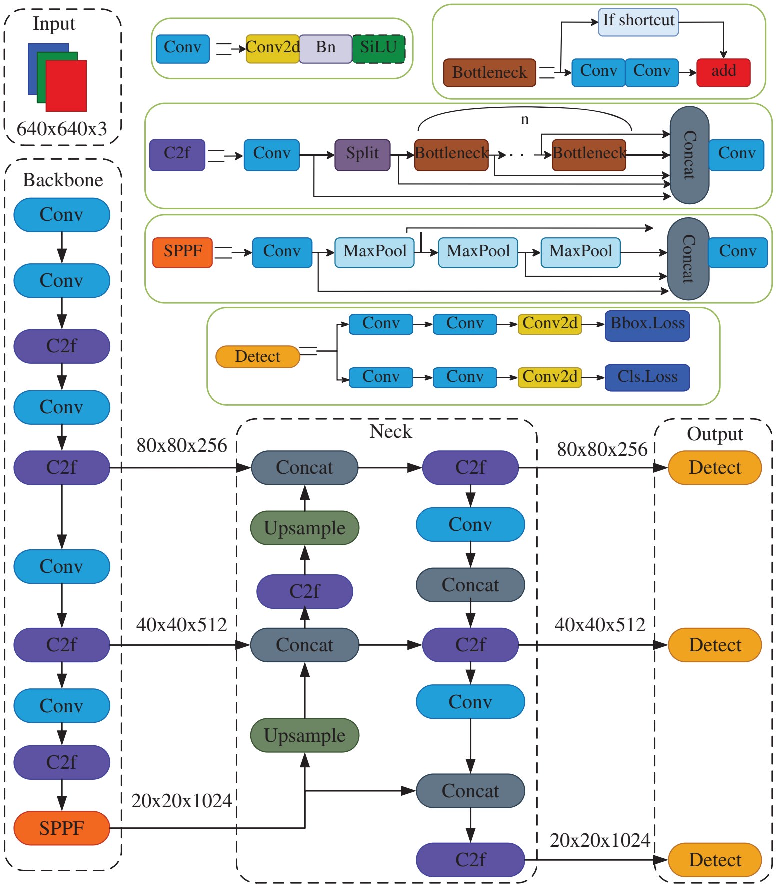


Figure 1: Model structure of the YOLO v8 algorithm

3.1 YOLO v8n Algorithm Improvement Strategy

Addressing the issues of low detection rates for floating litter on water surfaces and the excessive size and parameters of traditional networks, this paper proposes a tailored lightweight variant of the Ghost-YOLO v8 algorithm for surface water litter detection, as shown in Fig. 2. This algorithm integrates an attention mechanism with a small-sized target detection unit and comprises four key improvements. The SE attention mechanism is embedded in the backbone network to bolster the model's emphasis on detecting small targets, along with a 160 × 160 small-target detection layer at the neck-end (as shown in the red dashed box in Fig. 2). Additionally, while ensuring the precision remains intact, we minimize the computational complexity and the model's parameter count by adopting a co-merging strategy with GhostNet in the backbone network of the YOLO v8 architecture: optimizing the convolution operation (the GhostConv module, Red serial number ① in Fig. 2) and using a more efficient module (the C2fGhost module, Red serial number ② in Fig. 2) to achieve the model's lightweight, the WIoU v3 loss function is utilized to optimize the bounding box regression accuracy.

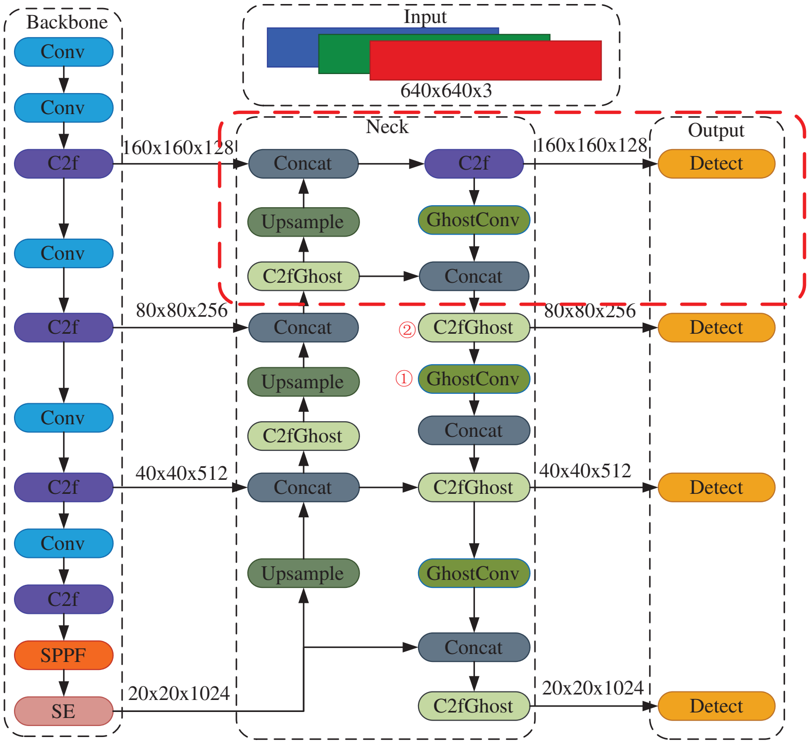


Figure 2: Model structure of the Ghost-YOLO v8 algorithm

3.2 SE-Backbone

Owing to the targets' compact arrangement and diminutive dimensions in the water surface floating litter image dataset, the baseline YOLO v8n model exhibits limited feature extraction capability and detection accuracy for small targets, failing to meet the desired detection accuracy for water surface floating litter images. To address this, the fixed-length feature vectors generated by the SPPF module are used by us to process images with diverse scales, as inputs to the SE attention mechanism module. This module focuses on essential data pertinent to the ongoing task by adaptively learning the relevance and importance of feature channels, suppressing irrelevant information, and enhancing the representational capacity of the network model. Exhibited in Fig. 3, the backbone network has been integrated with the SE attention mechanism module, forming the SE-Backbone, that improves the ability to discern and extract unique information from images on the water surface by weighting the importance of feature extraction images. Our conclusion comes from previous research [32].

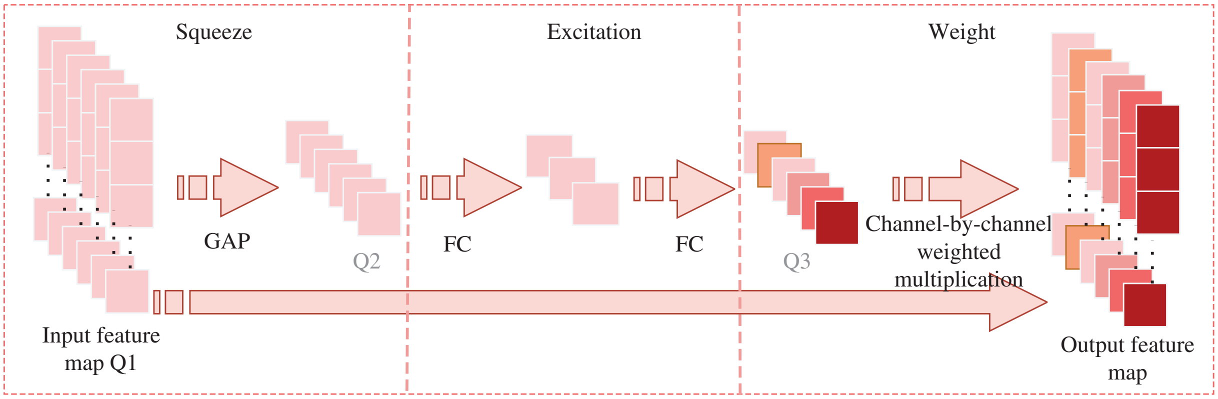


Figure 3: SE attention mechanism module

3.3 Small Target Detection Head

YOLO v8 uses an anchor-free detection head to predict the bounding box by removing some unnecessary optimisation operations such as convolution to achieve faster detection speed [33]. However, Owing to the diminutive nature of the samples in the floating litter on the water surface and the large downsampling multiplier of YOLO v8, the intricacy of obtaining feature details from diminutive targets in deeper layers of feature maps hinders achieving a satisfactory detection outcome. In contrast, the FPN (Feature Pyramid Network) structure has an obvious advantage in dealing with complex visual scenes, and its unique pyramid structure allows the model to perform feature fusion at multiple scales, thereby empowering the model to appreciate both the fine-grained specifics and the encompassing context present in the image. Therefore, this paper suggests adding a layer for detecting minute targets, as shown by the red box in Fig. 2, to add a 160 × 160 scale small target detection header in the bottom network of FPN and facilitate feature integration with the C2f layer located at the base of the backbone network to better extract detailed information from images, thereby strengthening the semantic content and feature portrayal of small targets within floating waste on water surfaces.

3.4 GhostNet Lightweight Architecture

Deep convolutional neural networks usually involve CNN consisting of a large number of convolutions, which leads to high computational cost. Inspired by the lightweight network GhostNet, we adopt the GhostConv module instead of ordinary convolutions and design the new C2fGhost module instead of the C2f module, it alleviates the heavy reliance on 3 × 3 regular convolutions in the base structure, resulting in a compressed network model with reduced parameters and lower computational demands. This allows the model to be more easily deployed on mobile for real-time identification of debris floating atop the water.

The GhostConv module creates numerous feature maps via cost-effective arithmetic processes, which reduces the learning cost of non-critical features, and thus lowers memory requirements during transitional scaling. The GhostConv module enhances the conventional convolution module by splitting the regular convolution operation into two separate steps: Firstly, the input information undergoes a regular convolution to produce a subset of feature maps. Secondly, linear operations are executed on these primary feature maps to yield supplementary, redundant feature maps. Lastly, the results from these two steps are concatenated together. As shown in Fig. 4, the GhostConv module is compared with the regular convolution module.

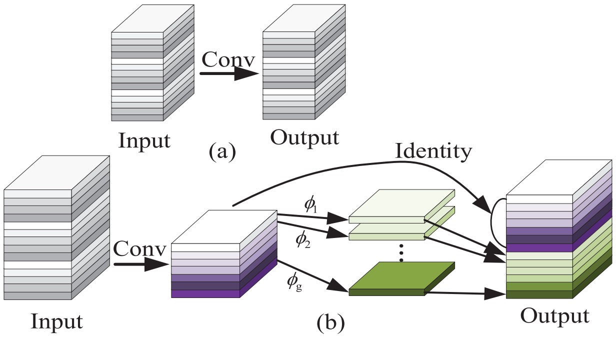


Figure 4: Comparison between ordinary convolutional module and GhostConv module, (a) is ordinary convolutional module, (b) is GhostConv module

The new C2fGhost module outlined in this paper is shown in Fig. 5, utilizing the strategy of feature fusion across stages and the truncated gradient flow technique, it augments the variability of the features learned by different layers within the network, and implements the GhostBottleneck in place of all bottlenecks within the C2f module of the original network, effectively reducing the influence of redundant gradient data and further amplifying the network's learning prowess.

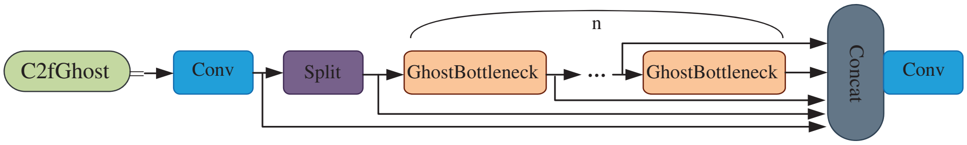


Figure 5: C2fGhost structure

3.5 *WIoU v3 Loss Function*

For the regression of detection frames, the YOLO v8 model utilizes CIoU as the loss function, which factors in the overlapping zone, centroid predicted spacing and form factor and real frames in the bounding box regression, i.e.,

$$\left\{\left(W = kW_{gt}, H = kH_{gt}\right) | k \in R^{+}\right\} \tag{1}$$

where W and H are the width and height of the predicted bounding box; W_{gt} and H_{gt} measure the actual width and height of the bounding box, respectively. However, CIoU has some ambiguities based on the relative metrics outlined by the aspect intersection ratio and does not consider the equilibrium between challenging and straightforward samples. Therefore, in this paper, we select WIoU v3 utilizing an alternative to CIoU for bounding box loss computation method for the Ghost-YOLO v8 algorithm.

The varying and non-steady targeting system in WIoU v3 can effectively avoid detrimental effects of suboptimal samples during training by weighing the proportion of high-quality and substandard samples, focusing the results of the bounding box regression on the target object, and solving the problem of floating litter on the surface of the water being difficult to be detected due to the occlusion. v3 is based on the addition of the adaptive and non-uniform focusing dynamics to WIoU v1 with the expression of Eqs. (2)–(5).

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

$$L_{WIoUv1} = R_{WIoU} L_{IoU} \tag{3}$$

$$L_{WIoU\,v3} = r L_{WIoUv1} \tag{4}$$

$$r = \frac{\beta}{\delta \alpha^{\beta-\delta}} \tag{5}$$

where W_g and H_g correspond to the minimal width and height closure region of the prediction frame and the true frame; $*$ denotes the separation of W_g and H_g from the graph, locating constants and preventing the creation of gradients that impede convergence; and the mapping of the non-monotonic focusing coefficients r and the outliers β is controlled by the hyper-parameters α and δ.

4 Experiment

The testing setup described in this paper comprises the following: A Linux Ubuntu 16.04.4 LTS operating system, an NVIDIA GeForce RTX 3090 GPU equipped with 24 GB of graphics memory, utilizing YOLO v8n as the baseline model. The batch size hyperparameter is configured at 8, with 150 training iterations set. The optimization algorithm selected is SGD, and an initial learning rate of 0.01 is applied, the channel depth is 0.33, the channel width is 0.25, and max_channels is 1024.

4.1 Floating Litter Dataset on Water Surface

In this paper, we use Marine dataset [34], which contains a total of 1335 water surface floating litter data as the dataset for training, and the uncrossed 130 data as the dataset for validation. With a view to overcome the overfitting problem that occurs when the dataset is too small, and the effect of light and reflection on water surface floating garbage detection, the following data enhancement is applied to create three versions of each source image: (1) 50% horizontal flip probability; (2) 90-degree rotation with the same probability in the following three ways: none, clockwise, and counterclockwise; (3) randomly cropping the image from 0% to 20%; and (4) random rotation between −15 and +15 degrees, which improves the diversity of the image information and allows the model to learn a more comprehensive range of floating trash features during training.

4.2 Evaluation Metrics

In terms of model detection performance, the parameter (params) count and the amount of computation (GFLOPs) are used as evaluation metrics for model size. Among them, params denotes the number of parameters inside the schema of the network, that aligns with the spatial intricacy and dictates the size of utilization; GFLOPs denotes computational rate in floating-point operations, that aligns with the temporal complexity and determines the length of the network execution time. Frames Per Second (FPS) denotes the model's image processing rate per second, crucial for assessing real-time target detection efficiency. The calculation formula is Eq. (6).

$$\text{FPS} = \frac{1/\left(\text{T}_{\text{pre}} + \text{T}_{\text{in}} + \text{T}_{\text{post}}\right)}{1000} \tag{6}$$

where T_{pre}, T_{in}, T_{post}, indicate respectively the time taken for preprocessing, inference, and post-processing.

In terms of model detection accuracy, we choose P (precision rate), R (recall rate) and mAP (mean average precision) as the evaluation indexes of accuracy. The calculation formulas are Eqs. (7)–(11).

$$\mathrm{P} = \mathrm{P_T} / (\mathrm{P_T} + \mathrm{P_F}) \tag{7}$$

$$\mathrm{R} = \mathrm{P_T} / (\mathrm{P_T} + \mathrm{N_F}) \tag{8}$$

$$AP@0.5 = \frac{1}{\mathrm{n}} \sum_{i=1}^{n} P_i = \frac{1}{n} P_1 + \frac{1}{n} P_2 + \cdots + \frac{1}{n} P_n \tag{9}$$

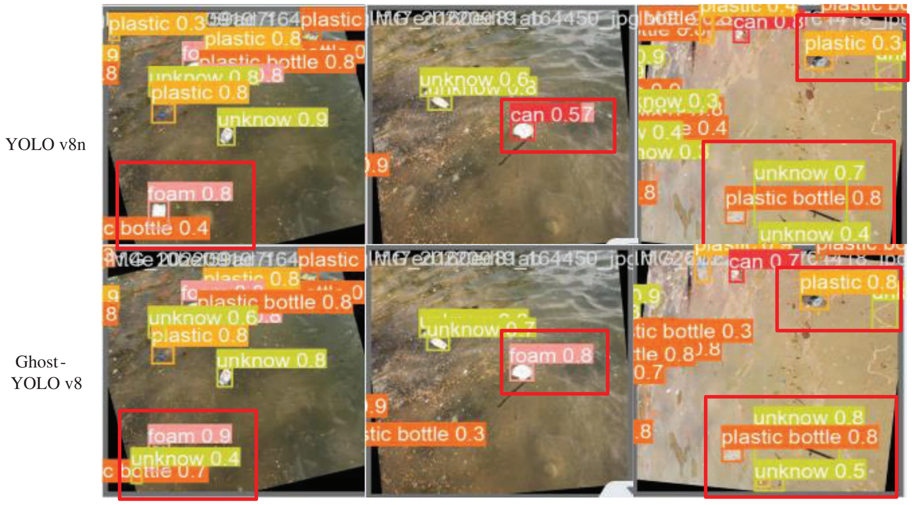
$$mAP@0.5 = \frac{1}{N} \sum_{k=1}^{N} AP@0.5_k \tag{10}$$

$$mAP@0.5:0.95 = \frac{1}{10} mAP@0.5 + \frac{1}{10} mAP@0.55 + \cdots + \frac{1}{10} mAP@0.95 \tag{11}$$

where the number P_T denotes the samples that are predicted accurately, while P_F stands for the mispredicted samples; N_F represents the number of unpredicted samples; $AP@0.5$ denotes the mean accuracy for this sample class, with an IoU threshold of 0.5 in the confusion matrix; $mAP@0.5$ is the average of the $AP@0.5$ values of samples taken across all categories; N is the total number of categories; the computed mAP score, encompassing IoU thresholds from 0.5 to 0.95, evaluated in steps of 0.05, is referred to as $mAP@0.5:0.95$.

4.3 Visualisation Results

According to the experimental findings that the enhanced Ghost-YOLO v8 model presented in this paper surpasses the YOLO v8n baseline in detecting small objects, Fig. 6 demonstrates comparison of detection effectiveness of baseline models YOLO v8n and the Ghost-YOLO v8 model.



(a) (b) (c)

Figure 6: Comparison of the detection results of YOLO v8n and Ghost-YOLO v8 models, (a) represents the comparison of missed detection (b) represents the comparison of false detection (c) represents the comparison of confidence level

As clearly illustrated in the comparison graph in Fig. 6, the Ghost-YOLO v8 model surpasses the baseline YOLO v8n in detecting objects, especially small targets such as unknowns. Not only does Ghost-YOLO v8 detect more of these objects, but it also enhances the confidence level for the same target. From Fig. 6a, it is evident that in scenarios where the objects are minute in size, the YOLO v8n model overlooks the presence of unrecognized object targets, resulting in a missed detection situation, while the Ghost-YOLO v8 algorithm does not have a missed detection situation; from Fig. 6b, it is evident that the YOLO v8n model experiences a case of misdetection, which misclassifies the floating foam as a jar, while the Ghost-YOLO v8 algorithm does not suffer from misdetection; from Fig. 6c it is noticeable that concerning the detection of the same small target, the Ghost-YOLO v8 algorithm generally detects it with higher confidence than the benchmark model.

4.4 Improvement Methods and Results

4.4.1 SE-Backbone Improvement Experiment

Enhancement of the backbone network is achieved by appending the SE attention mechanism module, in addition to Self [35] (Self-Attention Mechanism), CBAM [36], GAM [37] (Global Attention Mechanism) module, and MHSA [38] (Multi-Head Self-Attention), respectively, the enhancement is directed towards the 10th level of the backbone framework (beginning with layer 0) within the YOLO v8n model. Within identical experimental settings, the inclusion of the attention module leads to varying levels of improvement in the model's detection accuracy, in particular, on the mAP@0.5 metric, Self, CBAM, GAM, and SE attention mechanisms brought 1.5%, 0.7%, 1.5%, and 1.7% accuracy improvements, respectively, and on them AP@0.5:0.95 metric, Self, GAM, and SE attention mechanisms brought 0.7%, 1%, and 0.5% accuracy improvements, respectively and the performance comparison is shown in Table 1, with the highest detection benefit by integrating of the SE attention module. Without increasing the computational effort, there was a 1% improvement in precision and a 2.4% jump in recall. Similarly, the mAP@0.5 enhanced by 1.7%, and the mAP@0.5:0.95 incremented by 0.5%.

Table 1: Contrasting the model's efficiency with the attention mechanism integrated into the backbone

Model	mAP@0.5/%	mAP@0.5:0.95/%	Precision/%	Recall/%	GFLOPs
YOLO v8n	90.3	61.1	91	82.3	8.1
YOLO v8n+Self	91.8	61.8	92	84.4	8.2
YOLO v8n+CBAM	91	60.7	91.7	82.5	8.1
YOLO v8n+GAM	91.8	62.1	92.6	84.2	9.4
YOLO v8n+MHSA	90.3	60	90.7	82.5	8.2
YOLO v8n+SE	92	61.6	92	84.7	8.1

4.4.2 Small Target Detection Head Improvement Experiments

By adding 160 × 160 scale Miniature target identification head in the bottom network of FPN and convergence the features with C2f at the bottom of the backbone network (as shown in the red box in Fig. 2), the detailed information in the image can be better extracted, and in this way, the explanatory data and feature portrayal of the tiny entities in the floating litter on the water surface can be enhanced. The comparison of performance outcomes is depicted in Table 2, mAP@0.5 showcasing enhancements of 0.6 and 4.2 percentage points in precision and recall, resulting in a 3.1% and 1.4% enhancement in the mAP@0.5:0.95 score, respectively, and the amount of parameters is reduced by 0.11 MB.

Table 2: Comparison of model performance with the addition of Head detection head at neck-end

Model	Params/MB	GFLOPs	Precision/%	Recall/%	mAP@0.5/%	mAp@0.5:0.95/%
YOLO v8n	11.47	8.1	91	82.3	90.3	61.1
+Head	11.36	12.5	91.6	86.5	93.4	62.5
Self+Head	11.87	12.6	91.9	86.5	93.3	63.1
GAM+Head	17.62	13.8	92	85.7	93	63
SE+Head	11.39	12.5	91.9	87	93.6	63.4

By analyzing the data from the backbone improvement experiments, the detection performance can be optimized further by using the Self module, the GAM module, and the SE attention module in combination with the Head detection head. Table 2 presents the experimental outcomes, the SE attention module combined with head-detection has the best detection effect, demonstrating a 3.3% and 2.3% enhancement in mAP@0.5 and mAP@0.5:0.95 respectively without any increase in parameter count.

4.4.3 GhostNet Lightweight Structure Improvement Experiments

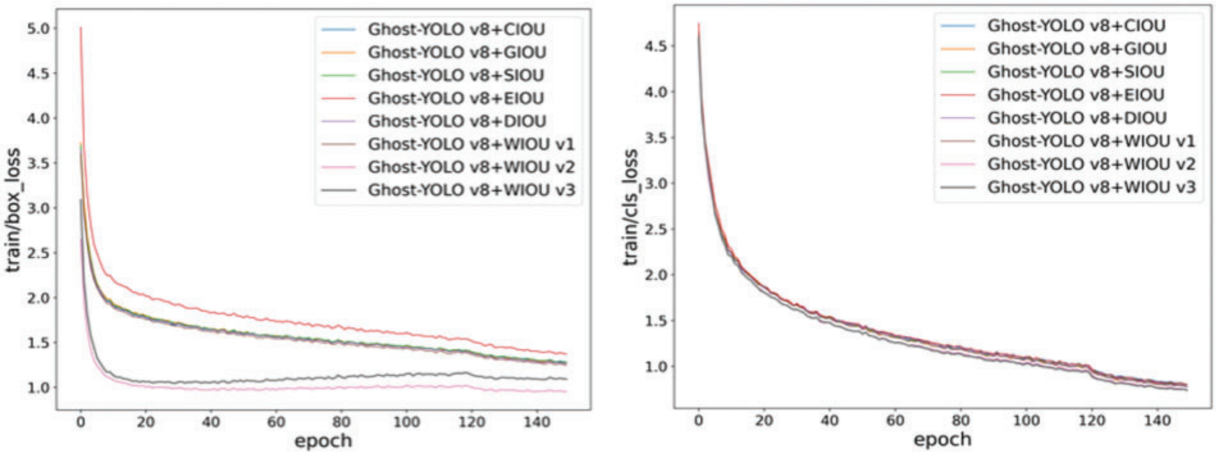
Utilizing diverse convolutional strategies, the model's neck-end in the YOLO v8n model is streamlined for efficiency and the C2f module constructed by this convolutional approach to assess and evaluate the effect of various streamlined designs on detection accuracy. The comparisons of performance outcomes are presented in Table 3, and the experimental findings indicate that the DSConv [39], GSConv [40] and GhostConv convolution methods with their constructed C2f modules enables a range of reductions in the model's computational burden. Since the lightweight model structure trades detection accuracy for speed, the DSConv+DSConv2D structure and the GSConv+VoV-GSCSPC [41] structure cannot satisfy both point and speedup. In this paper, we advocate utilizing GhostConv as an alternative to conventional convolution in the neck-end of the feature fusion process, and replace the lightweight structure of ordinary C2f module with the newly designed C2f Ghost module, that enhances the model's precision by 0.6 percentage points and its recall by 1.1 percentage points, respectively, improves the mAP@0.5 by 0.9%, and decreases the parameter count and the computation volume by 2.03 MB and 12.34%, which really achieves both point and speedup.

Table 3: Comparison of model performance of neck-end fusion lightweight structure

Model	Params/ MB	GFLOPs	Precision/%	Recall/%	mAP@0.5/%	mAP@0.5: 0.95/%
YOLO v8n	11.47	8.1	91	82.3	90.3	61.1
DSConv+DSConv2D	9.12	6.2	91.5	83	90.9	59
GSConv+VoV-GSCSPC	10.77	7.3	91	81.8	90.1	58.5
GhostConv+C2fGhost	9.44	7.1	91.6	83.4	91.2	59.8

4.4.4 WIoU v3 Loss Function Improvement Experiments

To validate the WIoU loss function within the assignment of spotting floating trash targets atop water, the commonly used IoU [42] loss function is compared on the Marine dataset to compare and analyse the effects of CIoU [43], GIoU, SIoU [44], EIoU, DIoU and WIoU on the model performance. Among them, box_loss (box regression loss) is employed to quantify the discrepancy between the anticipated and actual bounding box positions, the lower the box_loss, the more accurately the model predicts the position of the target; cls_boss (classification loss) assesses the model's performance in correctly classifying diverse groups, the lower the cls_boss, the more accurate the model is in distinguishing different categories. accuracy of the model. As evident from the comparative analysis of the loss function depicted in Fig. 7, WIoU v3 exhibits the minimal loss and superior model fitting capabilities. The outcomes of the experiments indicate that the model using the WIoU loss function (Ghost-YOLO v8+WIoU v3) improves mAP@0.5 and mAP@0.5:0.95 by another 1.8% and 1.6%, respectively, over the original model (Ghost-YOLO v8+CIoU).



(a) (b)

Figure 7: Comparison of loss functions (a) is the comparison of the bounding box loss function during training; (b) is the comparison of class loss function during training process

4.5 Ablation Experiment

To validate the efficacy of the enhancement module, YOLO v8n is employed as the baseline model. A series of ablation studies are performed, exploring various permutations of multiple improvement components, utilizing precision, recall, mAP@0.5, mAP@0.5:0.95, parameter count, and computational load as evaluation metrics. The outcomes of these ablation experiments are presented in Table 4.

Table 4: Ablation experiment results

Head	SE	GhostConv +C2fGhost	WIoU v3	FPS	Params/ MB	GFLOPs	P/%	R/%	mAP@ 0.5/%	mAP@0.5: 0.95%
				294.1	11.47	8.1	91	82.3	90.3	61.1
√				196.7	11.36	12.5	91.6	86.5	93.4	62.5

(Continued)

Table 4 (continued)

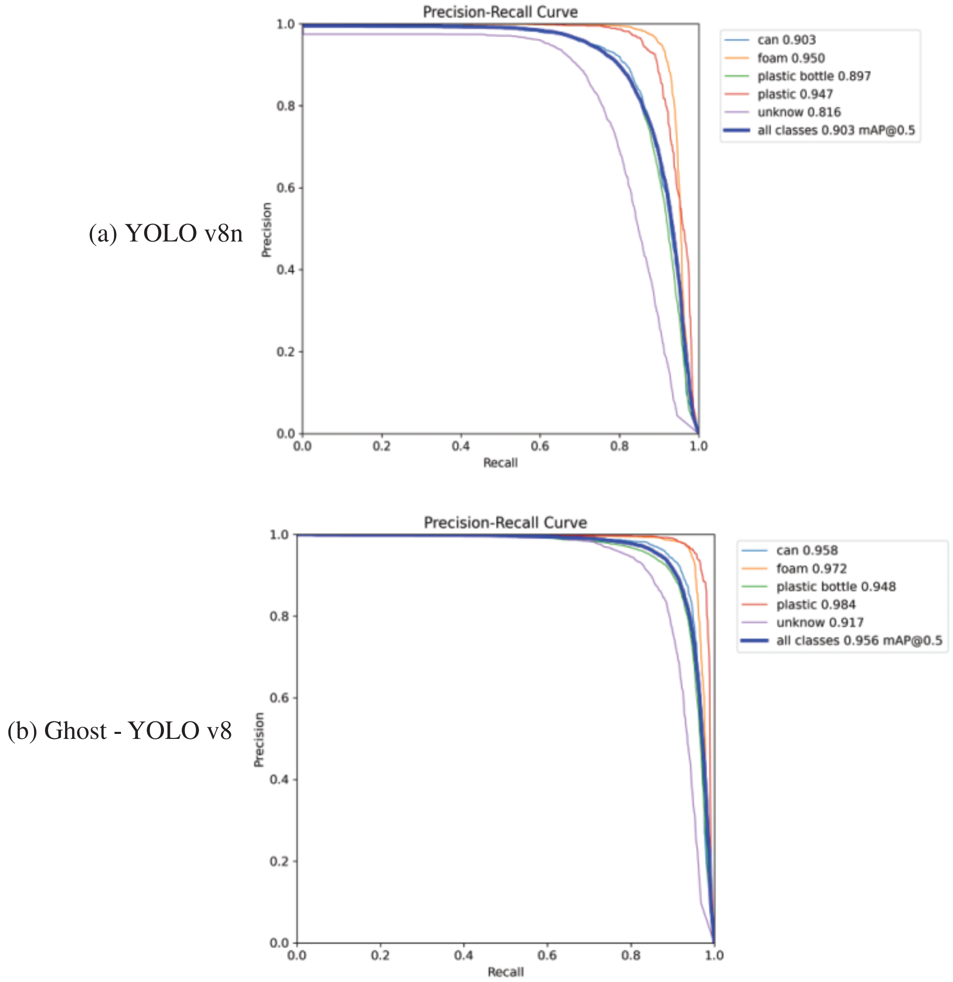
Head	SE	GhostConv +C2fGhost	WIoU v3	FPS	Params/ MB	GFLOPs	P/%	R/%	mAP@ 0.5/%	mAP@0.5: 0.95%
	√			303	11.5	8.1	92	84.7	92	61.6
		√		303	9.4	7.1	91.6	83.4	91.2	59.8
√	√			217.1	11.39	12.5	91.9	87	93.6	63.4
√	√	√		244.9	9.59	11.6	91.9	87.2	93.8	63.9
√	√	√	√	224.9	9.59	11.6	94.3	89.9	95.6	65.5

According to Table 4, the following key observations can be drawn:

By incorporating a 160 × 160 Minute target recognition layer in the feature fusion neck, precision and recall improved by 0.6% and 4.2%, respectively, there was an increase of 3.1% in mAP@0.5 and a 1.4% improvement in mAP@0.5:0.95. Incorporating the SE Attention Module into the backbone architecture led to a 1% boost in precision and a 2.4% increase in recall, respectively, and increased mAP@0.5 and mAP@0.5:0.95 by 1.7% and 0.5%. Implementing the lightweight GhostConv structure with the C2fGhost module in the neck boosted mAP@0.5 by 0.9% while reducing parameters and computations by 2.07 MB and 12.34%, respectively. The integration of the Head detection head and SE Attention Module led to gains of 3.3% and 2.3% in mAP@0.5 and mAP@0.5:0.95, respectively. The combined use of the Head detection head, SE attention mechanism, and lightweight structure lifted mAP@0.5 and mAP@0.5:0.95 by 3.5% and 2.8%, respectively, with a 1.88 MB reduction in parameters. With the addition of these three modules and the adoption of WIoU v3 as the loss function, precision and recall surged by 3.3% and 7.6%, respectively, and there was a 5.3% rise in mAP@0.5 and a 4.4% improvement in mAP@0.5:0.95, while computations were reduced by 1.88 MB and FPS almost unchanged. In summary, our proposed Ghost-YOLO v8 model maintaining superior real-time capabilities, while enhancing detection accuracy.

The trained network architecture's performance in detecting targets is showcased in Fig. 8. Specifically, Fig. 8a illustrates the results of the experimental tests conducted on the baseline model, YOLO v8n, while Fig. 8b showcases the outcomes of the Ghost-YOLO v8 model. Notably, across all categories, the Ghost-YOLO v8 model exhibits superior detection accuracy compared to the YOLO v8n baseline. achieving improvements of 5.5%, 2.2%, 3.7%, and 10% in the "can," "foam," "plastic bottle," "plastic," and "unknown" categories, respectively. When the IoU threshold is set to 0.5, the Ghost-YOLO v8 model attains a mAP@0.5 of 95.6%, outperforming the baseline model by 5.3 percentage points.

Incorporating the SE attention module, a small target detection header, and the lightweight GhostConv+C2fGhost structure, the refined Ghost-YOLO v8 algorithm exceeds the YOLO v8n baseline in both accuracy and parameter efficiency. Respectively, there is a 5.3% gain in mAP@0.5 and a 4.4% gain in mAP@0.5:0.95 on the Marine dataset, while achieving a 1.88 MB reduction in parameter size, ultimately enhancing network speed and precision.


(a) YOLO v8n

(b) Ghost - YOLO v8

Figure 8: Comparison of Precision-Recall curves (a) mAP detection effect of YOLO v8n algorithm (b) mAP detection effect of Ghost-YOLO v8 algorithm

4.6 Comparative Analysis of Detection Performance Among Different Models

To conduct a thorough assessment of the Ghost-YOLO v8 model's performance, this paper selects seven YOLO v3n [45], YOLO v5n, SSD, YOLO v6n [46], DETR [47], Fast R-CNN, and YOLO v8n [48] representative models are used as comparison objects. Table 5 reveals that in terms of the mAP@0.5 measure, the experimental results demonstrate, the Ghost-YOLO v8 model improves by 3.1 percentage points over the YOLO v3n model, outperforming the YOLO v5n model by 5.5 percentage points, surpassing the SSD model by 14 percentage points, outperforming the YOLO v6n model by a margin of 14 percentage points, surpassing the DETR model by 11.2 percentage points, outperforming the Fast R-CNN model by 8.2 percentage points, and also has a 5.3 percentage point improvement. The Ghost-YOLO v8 model also performs well in the mAP@0.5:0.95 metric, improving by 2.7 percentage points compared to the YOLO v3n model, 7.9 percentage points compared to the YOLO v5n model, 14.5 percentage points compared to the SSD model, 14.4 percentage points compared to the YOLO v6n model, 10.2 percentage points compared to the DETR model, 8.7 percentage points compared to the Fast R-CNN model, and a 4.4 percentage point improvement compared to the YOLO v8n. In addition, the mAP@0.5 values of the Ghost-YOLO v8 model reached their maximum values in all five categories of cans, foams, plastics, plastic bottles, and unknowns, which were 95.8%, 97.2%, 94.8%, 98.4%, and 91.7%, respectively, which further proves the advantages of the model in small target detection.

Table 5: Detection results of different algorithms on Marine dataset

Network	Can	Foam	Plastic bottle	Plastic	Unknow	mAP@0.5/%	mAp@0.5:0.95/%
YOLO v3n	92.7	93.8	92.1	96.6	87.1	92.5	62.8
YOLO v5n	88.6	94.4	90.7	94.3	82.6	90.1	57.6
SSD	88.7	78.4	87	82.9	70.5	81.5	51
YOLO v6n	82.7	89.8	83.1	83.9	68.5	81.6	51.6
DETR	85.7	80.1	91.6	86.7	78.2	84.4	55.3
Fast R-CNN	91.2	85.4	90.5	89.1	80.7	87.4	56.8
YOLO v8n	90.3	95	89.7	94.7	81.6	90.3	61.1
Ghost-YOLO v8	95.8	97.2	94.8	98.4	91.7	95.6	65.5

4.7 Comparative Results of Validation Tests on Different Datasets

To validate the authenticity and reliability of the enhanced algorithm, this paper is based on the VisDrone2019 dataset. The dataset was collected by the laboratory team of Tianjin University through UAVs in the diverse environments of 14 domestic cities and towns, and contains 10 types of common traffic targets, which is an authoritative dataset for evaluating the performance of small target detection. When baseline against the YOLO v8n, the Ghost-YOLO v8 model's average detection accuracy improves by 4.4%, as evidenced by the test results. Compared with classical algorithms such as DetNet59 [48], CornerNet [49], Fast R-CNN [50], CenterNet [51] and Mixed YOLO v3-LITE [52], Ghost-YOLO v8 demonstrates superior performance in small target detection, significantly proving the notable advancement and proven effectiveness of this paper's method within the realm of small target detection in Table 6.

Table 6: Target detection results for different models on the VisDrone2019 dataset (%)

Networks	Pedestrain	People	Car	Van	Truck	Tricycle	Bus	Motor	mAP@0.5
DetNet59	15.3	4.1	36.1	17.3	20.9	13.5	26	10.9	15.3
CornerNet	20.4	6.6	40.9	20.2	20.5	14	24.4	12.1	17.4
Fast R-CNN	21.4	15.6	51.7	29.5	19	13.1	31.4	20.7	21.7
CenterNet	22.6	20.6	59.7	24	21.3	20.1	37.9	23.7	26.2
Mixed YOLO v3-LITE	34.5	23.4	70.8	31.3	21.9	15.3	40.9	32.7	28.5
YOLO v8n	33.5	27.3	75.5	37.9	27.7	21.6	45	35.5	32.5
Ghost-YOLO v8	40.2	32.7	77.3	45.2	32.9	25.1	56.4	42.2	36.9

5 Conclusion

This paper presents a Ghost-YOLO v8 model that is both efficient and resource-friendly for detecting and identifying floating litter on water surfaces. Incorporating the SE attention module within the backbone network, the model's feature extraction and target localization accuracies are enhanced. The inclusion of a 160 × 160 layer designed for detecting small targets optimizes the fusion of deep and shallow semantic information, bolstering small target feature representation. The GhostConv module and a novel C2fGhost module significantly reduce network parameters, achieving a lightweight design. Furthermore, the dynamic non-monotonic focusing mechanism of WIoU v3 replaces CIoU as the loss function, improving the model's bounding box regression performance. Experimental results demonstrate that the Ghost-YOLO v8 model outperforms the baseline by 3.3% and 7.6% in precision and recall, respectively, while achieving improvements of 5.3% and 4.4% in mAP@0.5 and mAP@0.5:0.95, while keeping the FPS almost the same. Additionally, the model reduces computational requirements by 1.88 MB compared to the baseline, exhibiting a superior detection effect. The goal of our future work is to enhance the model's detection accuracy and speed even further and optimize its porting and deployment on edge mobile platforms.

Acknowledgement: I would like to express my sincerest gratitude to all those who contributed to this study by providing invaluable comments and suggestions to improve our thesis, and whose contributions were one of the key factors in the success of this study.

Funding Statement: Supported by the fund of the Henan Province Science and Technology Research Project (No. 242102210213).

Author Contributions: The authors confirm their contributions to the paper as follows: Zhongmin Huangfu and Shuqing Li conceived and designed the study; data collection: Shuqing Li; analysis and interpretation of results: Shuqing Li, Luoheng Yan; draft manuscript preparation: Shuqing Li, Luoheng Yan. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The dataset we use is a public dataset. Dataset link: https://universe.roboflow.com/marine-debris/marine-debris-i2ge3 (accessed on 4 December 2022).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Zhang, Y. Wei, H. Wang, Y. Shao, and J. Shen, "Real-time detection of river surface floating object based on improved refinedet," *IEEE Access*, vol. 9, no. 1, pp. 81147–81160, Sep. 2021. doi: 10.1109/ACCESS.2021.3085348.

[2] K. Zhang, R. K. Amineh, Z. Dong, and D. Nadler, "Microwave sensing of water quality," *IEEE Access*, vol. 7, no. 1, pp. 69481–69493, May 2019. doi: 10.1109/ACCESS.2019.2918996.

[3] R. Miikkulainen *et al.*, "Evolving deep neural networks," in *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. Elsevier, pp. 269–287, 2024. doi: 10.48550/arXiv.1703.00548.

[4] S. V. Walt *et al.*, "Scikit-image: Image processing in Python," *PeerJ*, vol. 2, no. 2, Jun. 2014, Art. no. e453. doi: 10.7717/peerj.453.

[5] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, “Object detection in 20 years: A survey,” *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, Mar. 2023. doi: 10.1109/JPROC.2023.3238524.

[6] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, “A review of Yolo algorithm developments,” *Procedia Comput. Sci.*, vol. 199, no. 1, pp. 1066–1073, Jan. 2022. doi: 10.1016/j.procs.2022.01.135.

[7] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, May 2018, pp. 7132–7141. doi: 10.48550/arXiv.1709.01507.

[8] K. Han, Y. Wang, Q. Tian, J. Guo, and C. Xu, “GhostNet: More features from cheap operations,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Mar. 2020, pp. 1580–1589. doi: 10.48550/arXiv.1911.11907.

[9] Z. Tong, Y. Chen, Z. Xu, and R. Yu, “Wise-IoU: Bounding box regression loss with dynamic focusing mechanism,” Jan. 2023. doi: 10.48550/arXiv.2301.10051.

[10] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye and D. Ren, “Distance-IoU loss: Faster and better learning for bounding box regression,” in *Proc. AAAI Conf. Artif. Intell*, Apr. 2020, vol. 34, pp. 12993–13000. doi: 10.1609/aaai.v34i07.6999.

[11] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: An astounding baseline for recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. workshops*, Jun. 2014, pp. 806–813.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Mar. 2017, pp. 2961–2969. doi: 10.48550/arXiv.1703.06870.

[13] R. Girshick, “Fast R-CNN,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[14] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *Adv. in Neural Inform. Process. Syst.*, vol. 39, pp. 91–99, 2015. doi: 10.1109/TPAMI.2016.2577031.

[15] M. Hussain, “YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection,” *Machines*, vol. 11, no. 7, Jun. 2023, Art. no. 677. doi: 10.3390/machines11070677.

[16] W. Liu *et al.*, “SSD: Single shot multibox detector,” in *Proc. Comput. Vis.-ECCV 2016: 14th European Conf.*, Amsterdam, Oct. 2016, vol. 9905, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[17] R. L. Gai, N. Chen, and H. Yuan, “A detection algorithm for cherry fruits based on the improved YOLO-v4 model neural computing and applications,” vol. 35, no. 19, pp. 13895–13906, May 2021. doi: 10.1007/s00521-021-06029-z.

[18] G. Wang, Y. Chen, P. An, H. Hong, J. Hu and T. Huang, “UAV-YOLOv8: A small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios,” *Sensors*, vol. 23, no. 16, Aug. 2023, Art. no. 7190. doi: 10.3390/s23167190.

[19] H. Yuan and L. Tao, “Detection and identification of fish in electronic monitoring data of commercial fishing vessels based on improved Yolov8,” *J. Dalian Ocean Univ.*, vol. 38, pp. 533–542, 2023. doi: 10.16535/j.cnki.dlhyxb.2022-354.

[20] G. Qiao, M. Yang, and H. Wang, “A detection approach for floating debris using ground images based on deep learning,” *Remote Sens.*, vol. 14, no. 17, Aug. 2022, Art. no. 4161. doi: 10.3390/rs14174161.

[21] F. Lin, T. Hou, Q. Jin, and A. You, “Improved YOLO based detection algorithm for floating debris in waterway,” *Entropy*, vol. 23, no. 9, Aug. 2021, Art. no. 1111. doi: 10.3390/e23091111.

[22] C. Hou, Z. Guan, Z. Guo, S. Zhou, and M. Lin, “An improved YOLOv5s-based scheme for target detection in a complex underwater environment,” *J. Mar. Sci. Eng.*, vol. 11, no. 25, May 2023, Art. no. 1041. doi: 10.3390/jmse11051041.

[23] Y. Li, X. Bai, and C. Xia, “An improved YOLOV5 based on triplet attention and prediction head optimization for marine organism detection on underwater mobile platforms,” *J. Mar. Sci. Eng.*, vol. 10, no. 9, Aug. 2022, Art. no. 1230. doi: 10.3390/jmse10091230.

[24] X. S. Yue, J. Li, Y. H. Wang, P. H. Zhu, Z. X. Wang and X. H. Xu, “Surface floating small target detection algorithm based on improved YOLOv5s,” (in Chinese), *China Ship Res.*, vol. 19, pp. 1–9, Jun. 2024. doi: 10.19693/j.issn.1673-3185.03689.

[25] H. Li, H. Jia, R. Zhang, S. Yang, Q. Qi and T. Liu, "Detection method of river floating objects based on edge computing," *Proc. J. Phy.: Conf. Series*, Mar. 2023. doi: 10.1088/1742-6596/2456/1/012035.

[26] L. Y. Lei, J. Ai, J. Peng, and D. Yao, "Deep learning based target detection and evaluation of floating objects on water surface," *Environ. Develop.*, vol. 31, no. 6, pp. 117–120, Jun. 2019. doi: 10.16647/j.cnki.cn15-1369/X.2019.06.071.

[27] P. Yang, *Research on Intelligent Monitoring of Water Surface based on Computer Vision*. Guizhou University for Nationalities, China, Jul. 2015.

[28] R. F. Chen, J. Wu, Y. Peng, Z. Li, and H. Shang, "Solving floating pollution with deep learning: A novel SSD for floating objects based on continual unsupervised domain adaptation," *Eng. Appl. Artif. Intell.*, vol. 120, no. 1, 2023, Art. no. 105857. doi: 10.1016/j.engappai.2023.105857.

[29] D. Li, Z. Yan, and J. Sun, "Research on river floating rubbish classification and detection technology based on UAV vision," *Metal Mine*, vol. 9, pp. 199–205, 2021. doi: 10.19614/j.cnki.jsks.202109028.

[30] N. Li, Y. Wang, S. Xu, and L. Shi, "Recognition of floating objects on water surface with small sample based on AlexNet," *Comput. Appl. and Softw.*, vol. 36, no. 2, pp. 251–257, 2019.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017. doi: 10.1145/3065386.

[32] Z. M. Huangfu and S. Q. Li, "Lightweight you only look once v8: An upgraded you only look once v8 algorithm for small object identification in unmanned aerial vehicle images," *Appl. Sci.*, vol. 13, no. 22, Nov. 2023, Art. no. 12369. doi: 10.3390/app132212369.

[33] A. Aboah, B. Wang, U. Bagci, and Y. Adu-Gyamfi, "Real-time multi-class helmet violation detection using few-shot data sampling technique and YOLOv8," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2023, pp. 5349–5357. doi: 10.1109/CVPRW59228.2023.00564.

[34] Marine Debris, "Marine Debris Dataset," 2022. Accessed: Jul. 11, 2023. https://universe.roboflow.com/marine-debris/marine-debris-i2ge3/dataset/49

[35] Z. Lin *et al.*, "A structured self-attentive sentence embedding," Mar. 2017. doi: 10.48550/arXiv.1703.03130.

[36] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. European Conf. Comput. Vis. (ECCV)*, 2018, vol. 11211, pp. 3–19. doi: 10.1007/978-3-030-01234-2.

[37] Y. Liu, Z. Shao, and N. Hoffmann, "Global attention mechanism: Retain information to enhance channel-spatial interactions," Dec. 2021. doi: 10.48550/arXiv.2112.05561.

[38] H. Tan, X. Liu, B. Yin, and X. Li, "MHSA-Net: Multihead self-attention network for occluded person re-identification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 8210–8224, 2022. doi: 10.1109/TNNLS.2022.3144163.

[39] M. G. D. Nascimento, R. Fawcett, and V. A. Prisacariu, "DSConv: Efficient convolution operator," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Nov. 2019, pp. 5148–5157. doi: 10.48550/arXiv.1901.01928.

[40] H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan and Q. Ren, "Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles," Jul. 2014. doi: 10.1007/s11554-024-01436-6.

[41] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang, "UnitBox: An advanced object detection network," in *Proc. 24th ACM Int. Conf. on Multimed.*, Aug. 2016, pp. 516–520. doi: 10.1145/2964284.2967274.

[42] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Visi.*, Feb. 2018, pp. 2980–2988. doi: 10.48550/arXiv.1708.02002.

[43] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 658–666. doi: 10.1109/CVPR.2019.00075.

[44] Z. Gevorgyan, "SIoU loss: More powerful learning for bounding box regression," May 2022. doi: 10.48550/arXiv.2205.12740.

[45] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," Apr. 2018. doi: 10.48550/arXiv.1804.02767.

[46] C. Li *et al.*, "YOLOv6: A single-stage object detection framework for industrial applications," Sep. 2022. doi: 10.48550/arXiv.2209.02976.

[47] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov and S. Zagoruyko, “End-to-end object detection with transformers,” in *Comput. Vis.–ECCV 2020(ECCV 2020)*, Glasgow, UK, May 2020, pp. 213–229. doi: 10.48550/arXiv.2005.12872.

[48] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, “TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Montreal, BC, Canada, Oct. 2021, pp. 2778–2788. doi: 10.1109/ICCVW54120.2021.00312.

[49] X. Zhang, Y. Feng, S. Zhang, N. Wang, and S. Mei, “Finding nonrigid tiny person with densely cropped and local attention object detector networks in low-altitude aerial images,” *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 15, pp. 4371–4385, 2022. doi: 10.1109/JSTARS.2022.3175498.

[50] W. Chen, X. Jia, X. Zhu, E. Ran, and X. Hao, “Target detection in unmanned aerial vehicle images based on DSM-YOLO v5,” *Comput. Eng. Appl.*, vol. 59, no. 18, pp. 226–233, Sep. 2023. doi: 10.3778/j.issn.1002-8331.2302-0324.

[51] R. Zhang, Z. Shao, and J. Wang, “Multi-scale dilated convolutional neural network for object detection in UAV images,” *Geomatics and Inform. Sci. of Wuhan Univ.*, vol. 6, pp. 895–903, 2020.

[52] X. Chen, D. Peng, and G. Yu, “Real-time object detection for UAV images based on improved YOLOv5s,” *Opto-Electron Eng.*, vol. 49, no. 3, 2022, Art. no. 210372. doi: 10.12086/oee.2022.210372.