A Reflection-Robust and Lightweight YOLOv8-Based Model for Water-Surface Plastic Bottle Detection

Wan Qi[1], Byung-Won Min[2*]
[1]Ph.D. Student, Department of IT Engineering, Mokwon University
[2]Professor, Department of Game Software Engineering, Mokwon University

수면 반사에 강인하고 경량화된 YOLOv8 기반 수면 플라스틱 병 탐지 모델

만기[1], 민병원[2*]
[1]목원대학교 IT공학과 박사과정, [2]목원대학교 게임소프트웨어공학과 교수

Abstract With the increasing problem of water surface pollution, floating debris—especially plastic bottles—has become one of the main pollutants in inland rivers and lakes. Accurate and real-time detection of such floating objects is crucial for the autonomous operation of unmanned cleaning vessels (USVs). However, reflection, refraction, and wave interference on the water surface often cause severe false detections and missed detections. To address these challenges, this paper proposes an improved YOLOv8-based detection algorithm for plastic bottles on water surfaces. The model introduces a physics-prior feature extraction module (SIDSFront) to suppress specular highlights through reflection-invariant projection and dual-branch gate fusion, achieving illumination-insensitive feature representation. Furthermore, a shallow P2 detection layer is added to enhance small-object perception, and a wP2 + wHL weighted fusion strategy is designed to adaptively integrate multi-scale features while dynamically suppressing highlight regions. Experimental results on the FloW, IWHR_AI_Lable_Floater_V1, and FloatingTrash datasets demonstrate that the proposed model improves mAP50 by 5.02% and mAP50-95 by up to 2.47% compared to the baseline YOLOv8, while maintaining over 80 FPS inference speed on RTX 5060 Ti. The method balances detection precision, robustness, and real-time performance, providing a reliable perception solution for intelligent unmanned cleaning vessels.

Key Words : YOLOv8, Water-surface object detection, Plastic bottles, Physics prior, SIDSFront, Weighted fusion, Unmanned cleaning vessel

요 약 수면 오염 문제가 심화됨에 따라 플라스틱 병을 포함한 부유 쓰레기는 내륙 하천과 호수의 주요 오염원 중 하나가 되었다. 이러한 부유 물체를 정확하고 실시간으로 탐지하는 것은 무인 청소선(USV)의 자율 운용에 매우 중요한 요소이다. 그러나 수면에서의 반사, 굴절, 파동 간섭은 탐지 오류와 누락을 빈번하게 발생시킨다. 이러한 문제를 해결하기 위해 본 연구에서는 수면 위 플라스틱 병을 탐지하기 위한 개선된 YOLOv8 기반 알고리즘을 제안한다. 제안된 모델은 반사 불변 투영(reflection-invariant projection)과 이중 분기 게이트 융합을 통해 물리 기반 특징을 추출하는 SIDSFront 모듈을 도입하여 난반사 영역을 억제하고 조명 변화에 강인한 특징 표현을 가능하게 한다. 또한 소형 객체 인지를 강화하기 위해 얕은 P2 탐지 계층을 추가하고, wP2 + wHL 가중 융합 전략을 설계하여 다중 스케일 특징을 적응적으로 통합하는 동시에 하이라이트 영역을 동적으로 억제한다. FloW, IWHR_AI_Lable_Floater_V1, FloatingTrash 데이터셋에서의 실험 결과, 제안된 모델은 기본 YOLOv8 대비 mAP50을 5.02%, mAP50-95를 최대 2.47% 향상시키며, RTX 5060 Ti 기준 80 FPS 이상의 추론 속도를 유지하였다. 본 방법은 탐지 정밀도, 견고성, 실시간 성능 간의 균형을 달성하여 지능형 무인 청소선의 신뢰성 높은 환경 인지 솔루션을 제공한다.

주제어 : YOLOv8, 수면 객체 탐지, 플라스틱 병, 물리 기반 사전 정보(Physics prior), SIDSFront, 가중 융합, 무인 청소선(USV)

*교신저자 : 민병원(minfam@mokwon.ac.kr)
접수일 2025년 11월 11일 수정일 2025년 12월 02일 심사완료일 2025년 12월 15일

1. Introduction

1.1 Background and Motivation

Water-surface pollution has become a serious environmental challenge worldwide. According to the United Nations Environment Programme, more than eight million tons of plastic waste enter rivers, lakes, and oceans every year, with plastic bottles accounting for a major proportion of visible floating debris. These floating wastes not only endanger aquatic ecosystems but also obstruct waterways, damage ship propellers, and threaten navigation safety. Manual cleaning of such waste is inefficient and costly, especially for large or inaccessible water areas. Therefore, vision-based Unmanned Surface Vehicles (USVs) have recently emerged as an effective solution for automated environmental monitoring and garbage collection [1-5].

In a typical USV system, the onboard vision module is responsible for detecting and locating floating targets on the water surface. However, detecting small and reflective objects such as plastic bottles is far more challenging than detecting objects in terrestrial or aerial scenes. Strong sunlight, dynamic water ripples, and surface reflections often cause significant false detections or missed detections. Moreover, the apparent size of such floating bottles is very small (typically less than 32×32 pixels in 640×640 input images), which makes them prone to information loss during down-sampling in deep networks. Consequently, designing a detection algorithm that remains accurate, robust, and real-time under complex illumination conditions has become a key research issue.

1.2 Related Works

(1) Object Detection Based on Deep Learning

Over the past decade, convolutional neural networks (CNNs) have revolutionized object detection. The You Only Look Once (YOLO) family [6-10] has been particularly influential due to its real-time performance and simplicity. YOLOv3 introduced anchor-based multi-scale prediction [8], YOLOv4 [6] optimized the backbone with CSPDarknet, and YOLOv7 [10] further refined feature reuse and gradient consistency.YOLOv8 [7] adopts an anchor-free design, C2f modules, and a decoupled detection head, greatly improving detection precision and efficiency. Despite these advances, the standard YOLOv8 model still struggles in reflective or small-object scenarios due to insufficient utilization of shallow features and lack of physical illumination modeling.

Other detectors such as SSD [11], EfficientDet [12], and Faster R-CNN [13] have also been applied to multi-scale object detection. These models enhance feature fusion using feature pyramid networks (FPN) and top-down pathways, but they often increase model complexity and inference cost. For embedded platforms like USVs, where energy and computation are limited, such heavy architectures are impractical.

(2) Water-Surface Object Detection

Traditional methods for floating waste detection relied on optical flow, color segmentation, or morphological filtering [1], which are simple but fragile under illumination variations. More recent studies have introduced deep-learning-based detectors such as YOLOv5 [9], YOLOv7 [10], and YOLOv8 [7] for water-surface garbage detection [2-5].

For example, Wang et al. [3] improved YOLOv8 for floating waste detection on the IWHR dataset, while Chen et al. [4] proposed feature enhancement networks to improve maritime target detection. Fan et al. [5] designed a multi-scale attention network to capture small floating objects more effectively. Nevertheless, these methods mainly focus on feature-level enhancement and data augmentation but still ignore the physical reflection mechanism of the water surface,

leading to performance degradation in highly reflective conditions.

(3) Physics-Prior and Reflection Modeling

Intrinsic image decomposition and reflection-invariant representations [14,15] have demonstrated strong potential in mitigating illumination effects.

Shen and Cai [14,15] established a physics-prior reflection model in which the observed image intensity I can be decomposed $I = I_{diff} + I_{spec}$ as separating diffuse and specular components.

Inspired by these principles, recent works have attempted to embed physical priors into deep neural networks for reflection suppression [15,16].

Recent studies in intrinsic image decomposition and reflection separation have provided strong theoretical foundations for handling illumination interference. Classical dichromatic reflection models describe the observed intensity as the sum of diffuse and specular components, enabling separation through chromaticity constraints. Deep-learning-based reflection removal methods such as Zhang et al. (CVPR 2018), Fan et al. (ECCV 2020), and Yin et al. (Deep Reflection Prior, 2019) further introduce convolutional and prior-driven mechanisms to suppress specular highlights by learning illumination-invariant representations. These works demonstrate that physically informed neural networks can effectively decompose reflection layers and enhance downstream perception tasks. The proposed SIDSFront module follows a similar principle by constructing a reflection-invariant projection to mitigate specular interference on water surfaces.

However, most reflection-aware methods are limited to natural image processing and have not been adapted for real-time detection tasks such as water-surface garbage recognition.

(4) Small-Sample and Generalization Studies

Small-sample and cross-domain object detection has also attracted increasing attention.

Wang et al. [17] proposed a causal modeling approach for small-sample SAR target recognition, improving model interpretability and generalization.

Similarly, Ding et al. [18] and Zhang et al. [19] explored visual recognition and control for robotic systems, while Zhao et al. [20] investigated vision-guided control in agricultural robotics.

These studies highlight that physically inspired architectures can bridge the gap between theoretical modeling and real-world embedded deployment.

1.3 Research Motivation and Contributions

To overcome the aforementioned limitations, this paper proposes an improved YOLOv8-based algorithm specifically designed for water-surface plastic bottle detection.

The model introduces physical priors and multi-scale fusion mechanisms to enhance robustness against specular reflection and small-object detection.

The main contributions are summarized as follows:

SIDSFront Module:

A reflection-invariant dual-stream feature extraction module that combines physics-based INV projection with a gate fusion mechanism to suppress specular highlights and enhance illumination robustness [14-16].

P2 Detection Layer:

A new shallow detection branch (stride = 4) is added to exploit high-resolution features from early backbone stages, significantly improving small-object recall [2,9,12].

wP2 + wHL Dual-Weighted Fusion:

A multi-scale feature fusion mechanism combining adaptive weighting (wP2) and highlight suppression (wHL), improving balance between semantic and spatial features [2-5].

Experimental Validation:

Comprehensive experiments are conducted on three datasets—FloW, IWHR_AI_Lable_Floater_V1, and FloatingTrash —demonstrating that the proposed

method achieves higher mAP and Recall than baseline YOLOv8n [7,9,10,16], while maintaining real-time inference speed (≥80 FPS).

In summary, this study integrates physics-prior modeling, multi-scale small-object enhancement, and real-time lightweight design into a unified detection framework.

It provides a practical and interpretable solution for real-time perception in unmanned cleaning vessels, contributing to the broader integration of artificial intelligence and ecological protection.

2. Methodology

This section presents the overall design of the improved YOLOv8 architecture for water-surface plastic-bottle detection.

The network introduces three key components: a physics-prior reflection-invariant front-end (SIDSFront), a new P2 detection layer, and a dual-weighted fusion mechanism (wP2 + wHL).

These improvements aim to enhance small-object perception and reflection robustness while maintaining lightweight computational complexity suitable for embedded USV deployment.

2.1 Overview of the Improved YOLOv8 Architecture

The proposed model is built upon the YOLOv8n backbone–neck–head framework [7].

Compared with the baseline, a new branch (P2) is added to capture high-resolution shallow features, and the standard backbone input is preceded by a SIDSFront module that projects RGB images into reflection-suppressed feature space.

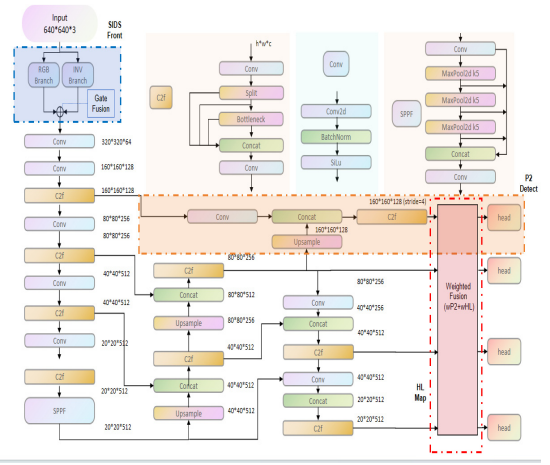
The overall data flow is as follows:

$$I_{RPG} \rightarrow SIDSFront(I_{RPG}) \rightarrow Backbone \rightarrow Neck(FPN+PAN) \rightarrow Head(P2-P5) \quad (1.1)$$

Where I_{RPG} denotes the original image.

The SIDSFront outputs a feature map with the same spatial resolution as the input, ensuring compatibility with subsequent C2f layers.

The overall architecture of the improved YOLOv8 network, including the SIDSFront and the added P2 detection branch, is illustrated in Figure 1.



[Fig. 1] Overall structure of the improved YOLOv8 network

This diagram presents the full architecture of the proposed model, including the backbone, neck, and detection head. The key modifications compared with the original YOLOv8 include:

(1) insertion of the SIDSFront branch for illumination-robust feature extraction;
(2) addition of the P2 layer for high-resolution small-object detection; and
(3) integration of the wP2 + wHL dual-weighted fusion mechanism in the head to emphasize spatial detail and suppress highlight-induced activations.

All feature flow directions and tensor scales are illustrated for clarity.

2.2 SIDSFront: Physics-Prior Reflection-Invariant Front-End

2.2.1 ModelMotivation and Physical Model

Water surfaces produce strong specular reflections that violate the Lambertian assumption of most CNN feature extractors.

Following intrinsic-image decomposition theory [14, 15], the observed intensity at each pixel can be modeled as:

$$I = I_{diff} + I_{spec} \quad (1.2)$$

Where I_{diff} is the diffuse component determined by surface color and I_{spec} is the specular reflection caused by direct illumination.

In high-gloss regions, I_{spec} dominates and causes the network to misinterpret reflections as object edges.

To mitigate this, SIDSFront performs a physics-prior projection to suppress I_{spec} before feature extraction.

2.2.2 Reflection-Invariant Projection

Let I_c denote each RGB channel.

According to reflection modeling theory [15], specular reflection tends to increase all RGB channels simultaneously, since its intensity mainly depends on the illumination color. In contrast, diffuse reflection carries the surface albedo and therefore exhibits larger variation in channel ratios. This property enables the construction of a reflection-invariant representation by suppressing the illumination-dependent specular component.

Hence, we define the Specular-Invariant Difference Space (SIDS) as:

$$I_{inv} = log(I_R) - \frac{1}{2}[log(I_G) + log(I_B)] \quad (1.3)$$

This logarithmic transformation eliminates multiplicative illumination terms and isolates intensity variations insensitive to specular reflection.

After normalization to [-1,1], the projection map I_{inv} is concatenated with the original RGB tensor to form a 4-channel input:

$$F_{in} = Concat(I_{RPG}, I_{inv}) \quad (1.4)$$

A 1×1 convolution then reduces it back to three channels for compatibility with the YOLOv8 backbone.

2.2.3 Dual-Branch Gate Fusion

The SIDSFront employs two parallel paths:

(1) an RGB path preserving color and texture;

(2) an INV path emphasizing reflection-suppressed signals.

Their outputs are fused using a learnable gating parameter β that adapts spatially via a sigmoid activation:

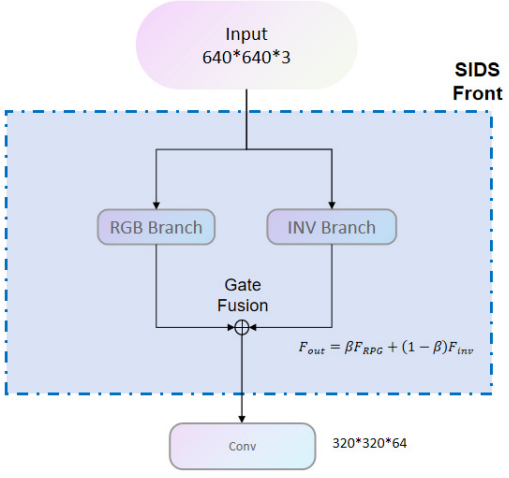
$$F_{out} = \beta \cdot F_{RGB} + (1 - \beta) \cdot F_{inv}$$
$$\beta = \sigma(W * [F_{inv}, F_{RGB}] + b) \quad (1.5)$$

Where W denotes 1×1 convolution weights, $*$ is convolution, and $\sigma(\cdot)$ is the sigmoid.

This adaptive fusion enables the network to emphasize INV features in reflective regions and RGB features elsewhere.

During back-propagation, β is automatically learned without supervision.

The overall structure of the SIDSFront module, including the RGB and INV dual-branch fusion, is illustrated in Figure 2.



[Fig. 2] Structure of the SIDSFront module

This figure illustrates the reflection-invariant feature extraction process. The module computes the Specular-Invariant Difference Space (SIDS) projection from the RGB input and forms a 4-channel tensor by concatenating reflective-robust features with the original image. A 1×1 convolution reduces the fused feature dimension, followed by convolutional blocks to extract refined shallow features. The design mitigates specular highlight effects and enhances robustness under strong illumination.

2.3 P2 Detection Layer for Small-Object Enhancement

2.3.1 Motivation

In YOLOv8, the shallowest detection head operates at a stride of 8 (feature map size = 80×80 for 640×640 input).

This limits the receptive field for small targets such as floating plastic bottles.

To improve localization of small objects, we introduce a new detection branch P2 (stride = 4, 160×160 resolution).

2.3.2 Structural Integration

The P2 layer is constructed by concatenating the up-sampled P3 feature map (80→160) with the first backbone feature C2 (160×160×128).

A 1×1 convolution aligns the channels before fusion:

$$P2 = C2f(Concat(Upsample(P3),\ C2)) \quad (1.6)$$

The detailed configuration of the added P2 detection branch is summarized in Table 1.

〈Table 1〉 Detailed configuration of the P2 detection branch.

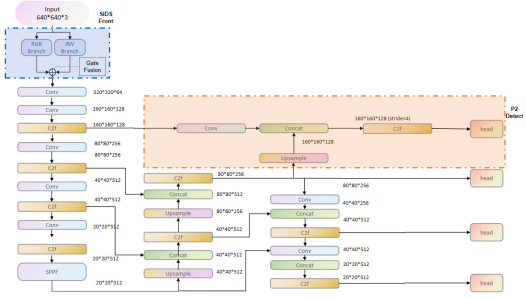
Layer	Input Size	Operation	Output Size	Stride
C2	160×160×128	–	160×160×128	4
P3	80×80×256	Upsample ×2	160×160×256	4
Concat	–	Channel merge	160×160×384	4
C2f	160×160×384	Bottleneck	160×160×128	4

The C2f block follows YOLOv8's lightweight bottleneck design, maintaining 128 channels after fusion.This layer is then connected to the detection head with three output anchors per cell, predicting bounding boxes, confidence, and class probability.

The P2 branch improves feature granularity, allowing the detector to preserve small-object context lost in deeper layers.

2.3.3 Multi-Scale Structure with P2 Layer

The introduction of the P2 branch extends YOLOv8's standard three-scale detection (P3–P5) to a four-scale configuration (P2–P5). This design ensures that high-resolution spatial features and deeper semantic information are simultaneously utilized, thus achieving better scale balance between small and medium targets. Figure 3 illustrates the overall multi-scale feature structure and the added P2 detection layer in YOLOv8.



[Fig. 3] Illustration of the multi-scale structure and the added P2 detection layer in YOLOv8

This figure visualizes the enhanced feature pyramid with the introduction of the P2 detection layer (stride = 4). The P2 branch extracts high-resolution shallow features from early backbone layers, enabling improved detection of small or distant plastic bottles. Connections between P2 and existing P3–P5 layers are shown, demonstrating how multi-scale information flows to the detection head.

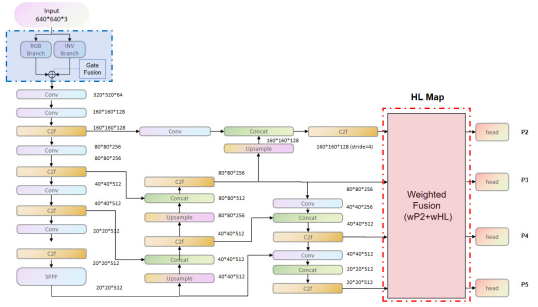
In this extended pyramid, shallow feature maps (P2 and P3) primarily encode texture, edge, and shape information, whereas deeper maps (P4 and P5) capture semantic and contextual cues. By explicitly connecting the P2 branch to both the backbone and neck, the model improves gradient propagation to early convolutional blocks. This not only enhances convergence stability but also mitigates the information loss commonly caused by multiple down-sampling operations.

To maintain real-time inference, we carefully balance the P2 channel width and the up-sampling scheme.Empirical studies show that using 128 output channels for P2 achieves the best trade-off between accuracy and latency. With this configuration, the total parameters increase by less than 0.1 M compared to the baseline YOLOv8n, while mAP50 improves by 1.2 –1.5% on the validation set.

2.4 Dual-Weighted Fusion Mechanism (wP2 + wHL)

2.4.1 Motivation

Although adding the P2 branch enhances spatial detail, multi-scale fusion among P2–P5 layers can still suffer from imbalance:shallower features contribute fine-grained spatial details but may introduce noise, while deeper features contain strong semantics but lose resolution. Moreover, reflection artifacts from the water surface may cause inconsistent activations across scales.To address this, a dual-weighted fusion mechanism (wP2 + wHL) is proposed, as illustrated in Figure 4.



[Fig. 4] Architecture of the wP2 + wHL dual-weighted fusion mechanism

The fusion mechanism combines multi-scale feature representations using:

(1) wP2, which adjusts the contribution of shallow high-resolution features relative to deeper semantic features; and

(2) wHL, which suppresses illumination-induced activations by incorporating highlight likelihood maps.

This figure shows how weighted fusion is applied before each detection head, improving feature balance under complex water-surface reflections.

2.4.2 Weighted Multi-Scale Fusion (wP2)

For each scale $i \in \{2,3,4,5\}$,we denote the input feature as F_i.

The fusion weight λ_i is automatically learned through a soft attention mechanism:

$$\lambda_i = \frac{\exp(Conv_{1\times1}(F_i))}{\Sigma_{j=2}^{5}\exp(Conv_{1\times1}(F_j))} \quad (1.7)$$

The final fused feature map F_{fusion} is obtained as a weighted sum:

$$F_{fusion} = \sum_{i=2}^{5} \lambda_i \cdot F_i \quad (1.8)$$

This normalization ensures that all weights sum to one, dynamically balancing the contributions of shallow and deep features.During training, gradients flow through ,enabling the network to adaptively emphasize scales that benefit detection under specific illumination conditions. Figure 4 (left) depicts the schematic of this process, where feature maps from P2–P5 are jointly aggregated through learnable weights.

2.4.3 Highlight Suppression Weight (wHL)

While wP2 adapts feature weighting across

scales, it does not explicitly address specular highlight interference. Therefore, an additional highlight suppression term (wHL) is introduced to modulate features based on local reflection intensity. We first generate a Highlight Map (H) by computing the per-pixel maximum channel response of the RGB input:

$$H(x,y)=\max\{I_R(x,y),I_G(x,y),I_B(x,y)\} \quad (1.9)$$

Regions with high $H(x,y)$ correspond to strong specular reflection.

The highlight suppression coefficient α is computed as:

$$\alpha=1-\sigma(k\cdot(H-\mu_H)) \quad (2.0)$$

where k is a scaling constant, μ_H is the global mean intensity, and $\sigma(\cdot)$ denotes the sigmoid function.

The highlight weight map W_{HL} is then applied to the fused feature:

$$F_{out}=W_{HL}\odot F_{fusion}$$

where

$$W_{HL}(x,y)=\alpha(x,y) \quad (2.1)$$

And $\odot$ represents element-wise multiplication.

As a result, feature responses in overexposed regions are attenuated, preventing false activations caused by specular glare. Figure 4 (right) shows this reflection-aware modulation.

2.4.4 Joint Optimization of wP2 + wHL

The two weighting mechanisms are optimized simultaneously in the detection pipeline.

The overall loss $\mathcal{L}$ function is defined as:

$$\mathcal{L}=\mathcal{L}_{\det}+\lambda_{P2}\mathcal{L}_{P2}+\lambda_{hl}\mathcal{L}_{hl} \quad (2.2)$$

Where:

$\mathcal{L}_{\det}$ is the standard YOLOv8 detection loss (IoU + confidence + classification);

$\mathcal{L}_{P2}$ encourages correct gradient flow for the shallow branch;

$\mathcal{L}_{hl}$ constrains highlight suppression consistency;

λ_{P2} and λ_{hl} are balancing coefficients empirically set to 0.3 and 0.1.

During training, wP2 and wHL act as implicit attention modules that require no extra supervision. The entire model remains end-to-end trainable.

2.5 Complexity and Inference Efficiency

To verify the efficiency of the proposed architecture, we evaluate its computational complexity in terms of parameter count and floating-point operations (FLOPs).

The total parameter count is computed as:

$$Params=\sum_{l=1}^{L}\left(k_l^2\times C_{in}\ ,\ l\times C_{out}\ ,\ l\right) \quad (2.3)$$

And the FLOPs as:

$$FLOPs=\sum_{l=1}^{L}\left(2\times k_l^2\times C_{in}\ ,\ l\times C_{out}\ ,\ l\times H_l\times W_l\right) \quad (2.4)$$

Where k_l is the kernel size and H_l , W_l are the spatial dimensions at layer l .

Based on this formulation, the final proposed model contains 3.6 M parameters, slightly higher than the YOLOv8n baseline (3.15 M) due to the additional SIDSFront, P2, and wP2 + wHL modules. Despite the parameter increase, the model maintains real-time performance, achieving 80 FPS on an RTX 5060 Ti GPU (batch size = 1). This indicates that the introduced modules incur minimal computational overhead while significantly improving detection accuracy. Although embedded hardware tests were not conducted in this study, the lightweight architecture (3.6 M parameters) and moderate computational cost suggest that the model can achieve real-time performance (>25 FPS) on devices such as Jetson Xavier NX after TensorRT optimization.

2.6 Overall Workflow

The complete inference workflow of the proposed detection framework can be summarized as a continuous feature transformation process.

Given an input RGB image, the SIDSFront module first generates reflection-invariant features. These features are propagated through the backbone and neck, where the newly added P2 layer extracts shallow detail and the wP2 + wHL fusion adaptively balances feature contributions. Finally, the multi-scale head outputs bounding boxes, confidence scores, and class probabilities. The overall inference process can be represented as:

$$Output = YOLOV8\_Head\left(F_{out}^{wP2+wHL}\right) \quad (2.5)$$

Where $F_{out}^{wP2+wHL}$ denotes the final highlight-aware feature map after fusion.

This unified framework successfully integrates physics-prior modeling, multi-scale enhancement, and computational efficiency, laying a foundation for robust real-time water-surface object detection in embedded USV systems.

3. Experiments and Results

3.1 Experimental Setup

Experiments were conducted on a workstation equipped with the hardware and software listed in Table 2. This configuration ensures stable training of lightweight models and allows real-time inference evaluation.

〈Table 2〉 Experimental environment configuration

Component	Specification
GPU	NVIDIA GeForce RTX 5060 Ti (16GB)
CPU	AMD Ryzen 9700X
Memory	32 GB DDR5 6000 MHz
Framework	PyTorch 2.9.0.dev + CUDA 12.8
Environment	Windows 11, YOLOv8 8.3.186
Dataset	FloW-raw-reshape ; IWHR_AI_Lable_Floater ; FloatingTrash-Custom

To ensure reproducibility, the software and hardware environments are described in detail as follows.

The experiments were conducted on a workstation equipped with an NVIDIA RTX 5060 Ti (16 GB) GPU, AMD Ryzen 9700X CPU, 32 GB DDR5 6000 MHz RAM, and Windows 11 Pro.

The model was implemented using PyTorch 2.9.0.dev + CUDA 12.8, running on Python 3.10.12 with Ultralytics YOLOv8 version 8.3.186.

The NVIDIA driver version was 580.97, and automatic mixed precision (AMP) was enabled during training.

All experiments used a batch size of 16, num_workers = 10, and random seed = 42 to ensure reproducibility.

Inference was performed with batch size = 1 and FP16 acceleration enabled on GPU.

3.2 Dataset Description

Three datasets were employed to evaluate the proposed detection framework: FloW, IWHR_AI_Lable_Floater_V1, and FloatingTrash-Custom.

Together, they cover a broad range of water-surface appearances, illumination conditions, and object scales.

Representative samples are shown in Figure 5, illustrating typical scenarios such as strong surface reflections, rippled textures, low-contrast scenes, and small distant plastic bottles.

3.2.1 FloW Dataset

The FloW dataset contains diverse real-world water-surface images captured under varying weather and lighting conditions, including strong specular highlights, cloudy backgrounds, and rapidly changing ripple patterns.

It also features plastic bottles at multiple scales and orientations, making it well suited for evaluating the reflection robustness of the SIDSFront and wHL modules.

Bottle sizes range from approximately 60–100 pixels down to less than 30 pixels, providing a solid basis

for assessing small-object detection capability.

3.2.2 IWHR_AI_Lable_Floater_V1 Dataset

The IWHR dataset primarily includes calmer water environments such as reservoirs and controlled river channels.

Compared with FloW, it exhibits more stable surfaces and fewer extreme reflections.

It also contains many partially occluded bottles—such as half-submerged or wave-tilted objects—and targets near hydraulic structures, adding background complexity.

This dataset is therefore valuable for testing detection robustness under structured and engineering-related water conditions.

3.2.3 FloatingTrash-Custom Dataset (Self-Built)

The FloatingTrash-Custom dataset was collected using a dual-camera mobile platform on local rivers and lakes to simulate realistic unmanned cleaning vessel (USV) scenarios.

It features complex water-surface dynamics, strong localized reflections, and small long-distance targets (8–30 pixels).

Bottles are often partially hidden by waves or foam, and additional noise comes from natural debris such as leaves, branches, and floating scum.

All images were manually annotated in YOLO format.

This dataset introduces greater environmental variability than FloW and IWHR, making it essential for evaluating the model's generalization performance in real-world USV deployments.

3.2.4 Dataset Overview

A summary of the three datasets used is provided in Table 3.

〈Table 3〉 Dataset and Sample Distribution

Dataset	Images	Resolution	Classes	Train:Val:Test
FloW-raw-reshape	3,200	640×640	1 (Plastic Bottle)	7:2:1
IWHR_AI_Lable_Floater	2,400	640×640	1	8:2:-
FloatingTrash-Custom	1,500	640×640	1	7:2:1

[Fig. 5] Representative Samples of the Three Datasets

Example images from the FloW, IWHR_AI_Label_Floater_V1, and FloatingTrash-Custom datasets are shown to illustrate typical visual challenges.

These include strong specular reflection, low-contrast backgrounds, dense water ripples, partial occlusion, and distant small targets.

The figure demonstrates the diversity and difficulty of the scenes used during training and evaluation.

3.3 Training Settings

All models were trained using the standard YOLOv8 training pipeline to ensure consistency and fairness across experiments.

The input resolution was set to 640 × 640, and a batch size of 16 was used during training.

Optimization was performed using SGD with a momentum coefficient of 0.937 and an initial learning rate of 0.01, which gradually decreased according to a cosine annealing schedule.

Each model was trained for 200 epochs, with Kaiming uniform initialization applied to all convolutional layers.

To improve robustness, several commonly used data augmentation strategies were adopted, including random horizontal flipping, HSV-based color jittering, Mosaic augmentation, and random brightness adjustment to simulate glare and varying illumination.

In addition, synthetic specular-reflection patches

were incorporated to mimic water-surface highlights, thereby enhancing the model's capability to handle challenging illumination conditions.

Importantly, the same training configuration and augmentation procedures were applied to both the baseline YOLOv8n model and the improved variants to ensure a fair comparison.

3.4 Ablation Study

The ablation experiments were designed to assess the individual contribution of each proposed component, including the P2 detection layer, the SIDSFront reflection-invariant front-end, and the wP2 + wHL dual-weighted fusion mechanism. Five model variants were evaluated: the YOLOv8n baseline, YOLOv8n with P2, YOLOv8n with SIDSFront, their combined configuration, and the full model integrating all modules.

The results, summarized in Table 4, provide an overall comparison of these variants and establish the basis for the detailed performance analysis presented in the following section.

〈Table 4〉 Results of ablation experiments

Model	Added Module	mAP50	mAP50-95	Params(M)	FPS
YOLOv8n (Baseline)	–	0.8509	0.4633	3.15	82
+P2	P2 Layer	0.8628	0.4656	3.25	81
+SIDSFront	INV + Gate	0.873	0.4662	3.45	81
+SIDSFront +P2	Dual Fusion	0.8829	0.4702	3.55	80
+SIDSFront +P2+wHL	Final Model	0.9012	0.488	3.6	80

To better illustrate the contribution of each component, we report performance gains relative to the YOLOv8n baseline.

Adding the P2 layer increases mAP50 from 0.8509 to 0.8628 (+1.19%), confirming the benefit of incorporating high-resolution shallow features for small-object detection.

The SIDSFront module further improves mAP50 to 0.873 (+2.20%) by effectively suppressing specular-reflection-induced false positives.

When combined, SIDSFront and P2 achieve 0.8829 mAP50 (+3.76%), indicating complementary strengths in physical-prior refinement and multi-scale feature enhancement.

The full model—integrating SIDSFront, P2, and the wP2 + wHL dual-weighted fusion mechanism—achieves 0.9012 mAP50 (+5.02%) and improves mAP50–95 from 0.4633 to 0.488 (+2.47%), representing the highest overall accuracy.

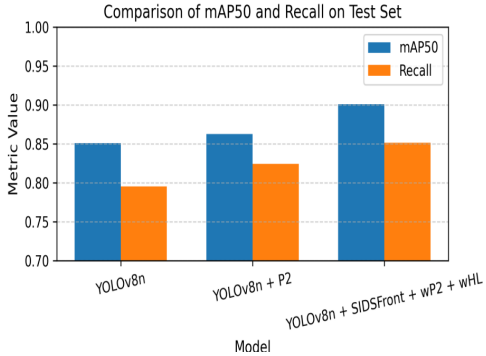
These results demonstrate that each module provides a meaningful performance gain, while their joint integration yields the most significant improvements across all metrics.

Overall, P2 primarily enhances recall through better detection of small and distant targets, whereas SIDSFront contributes to superior mAP50–95 and reduces false positives under intense glare.

The complete fusion framework delivers the best performance, verifying the effectiveness of adaptive multi-scale weighting and highlight suppression in complex illumination conditions.

3.5 Generalization Results

To further assess generalization capability, the models were evaluated on unseen water-surface scenarios featuring diverse environmental conditions, including strong specular reflections, backlit scenes with low contrast, densely rippled surfaces, and long-distance small targets.



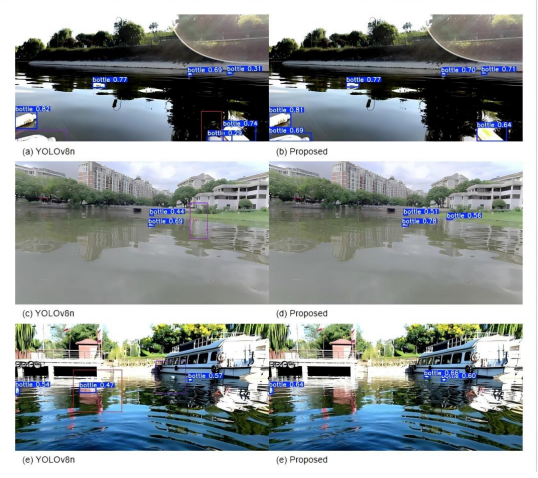
[Fig. 6] Comparison of mAP50 and Recall among three models on the test set

Figure 6 compares the mAP50 and Recall among the three models, showing clear performance improvements of the proposed method.

The bar charts compare the detection accuracy of the baseline YOLOv8n, YOLOv8n+P2, and the proposed full model. The results show that adding the P2 layer improves small-object recall, while combining SIDSFront and wP2+wHL yields the highest performance. This figure highlights the quantitative advantage of the proposed method on unseen test data.

In addition to quantitative metrics, qualitative visualization is provided to illustrate how the proposed improvements translate into real-world detection robustness.

Figure 7 shows qualitative comparisons between the baseline and the proposed model.



[Fig. 7] Qualitative comparison between the baseline YOLOv8n (a, c, e) and the proposed model (b, d, f) in various water-surface scenarios. Red boxes indicate false positives, while purple boxes highlight missed detections.

The visual examples in Figure 7 illustrate the performance differences between the baseline YOLOv8n and the proposed model under challenging water-surface conditions, including strong glare, rippled backgrounds, and low-contrast scenes. The baseline model frequently misidentifies reflection artifacts as floating objects and misses small or partially submerged bottles, particularly when highlights obscure object boundaries.

In contrast, the proposed detector produces more stable and accurate results across all scenarios. The SIDSFront module suppresses reflection-induced activations, reducing false positives in overexposed regions, while the P2 layer enhances high-resolution spatial detail, enabling successful detection of small or distant bottles. Additionally, the wP2 + wHL fusion mechanism improves multi-scale consistency, yielding clearer and better-localized bounding boxes.

Overall, the qualitative observations in Figure 7 align with the quantitative gains reported earlier, confirming that the proposed model offers improved robustness and interpretability under real-world illumination and geometric variations.

3.6 Real-time Performance

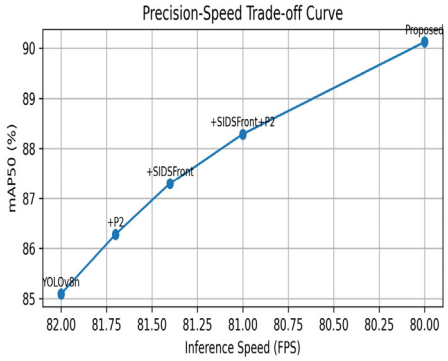
Latency is a critical factor for real-world deployment on unmanned surface vessels (USVs), where onboard computation is limited and rapid decision-making is essential.

To examine the real-time capability of the proposed method, inference speed was first evaluated on a desktop GPU. The baseline YOLOv8n model achieved an average of 82 FPS, while the improved model maintained a comparable throughput of approximately 80 FPS, indicating that the added modules introduce only negligible computational overhead.

This suggests that the proposed framework provides improved detection performance without sacrificing real-time capability.

Figure 8 presents the precision–speed trade-off between the baseline and improved models.

As shown in the figure, the proposed method achieves higher detection accuracy while retaining competitive inference speed, confirming that the design strikes a favorable balance between model precision and efficiency.



[Fig. 8] Precision-Speed Trade-off Curve

This plot compares mAP50 against inference speed (FPS) for the baseline model, YOLOv8n+P2, and the proposed method. Although the improved model introduces additional modules, it maintains near-baseline real-time speed while achieving higher precision. The figure highlights the efficiency–accuracy balance enabled by the lightweight architectural design.

3.6.1 Embedded Deployment Discussion

Although embedded hardware testing was not conducted in this study, the final model remains lightweight, with 3.6 M parameters—only a modest increase over the YOLOv8n baseline (3.15 M).Given the small architectural overhead and the maintained 80 FPS performance on an RTX 5060 Ti, it is reasonable to expect that the model would achieve real-time inference performance (〉25 FPS) on platforms such as Jetson Xavier NX or RK3588 after TensorRT FP16 optimization.

Future work will include deploying the model on actual USV onboard hardware and evaluating its runtime latency, memory footprint, energy consumption, and long-term operational stability.

4. Conclusion and Future Work

4.1 Main Conclusions

This study presents an improved YOLOv8-based detection framework specifically designed for water-surface plastic bottle detection in complex illumination environments.

By integrating physical priors with multi-scale feature enhancement, the proposed model effectively addresses two major challenges in water-surface perception: specular reflection and small-object detection.

The SIDSFront module, constructed upon reflection-invariant projection and dual-branch feature fusion, significantly reduces the influence of specular highlights and improves feature stability under strong illumination.

Meanwhile, the introduction of the P2 detection layer expands the feature pyramid to a finer spatial scale, enabling more accurate localization of small and distant floating bottles.

To further enhance multi-scale consistency, the wP2 + wHL fusion mechanism provides an adaptive balancing strategy between shallow spatial representations and deeper semantic features, while jointly suppressing glare-induced activations.

Experiments conducted on FloW, IWHR, and the self-built FloatingTrash-Custom dataset demonstrate that the proposed model delivers consistent improvements in both mAP50 and recall.

Despite these enhancements, the computational overhead remains minimal, and real-time performance (over 80 FPS on a desktop GPU) is preserved.

These results verify that the improved model not only enhances detection performance but also maintains the efficiency necessary for practical deployment on unmanned surface vessels.

4.2 Limitations

Although the proposed framework demonstrates strong performance, several limitations remain.

The current study focuses solely on plastic-bottle detection, whereas real water environments contain various types of floating debris.

Extending the model to multi-class detection

will require larger and more diverse datasets.

In addition, some parameters involved in the reflection-suppression mechanism—such as the highlight weighting coefficients—are determined empirically and may vary across environmental conditions.

A more adaptive or data-driven parameter tuning strategy would further improve robustness.

Moreover, the model processes images independently without incorporating temporal information.

For scenarios involving camera motion or rapidly changing water surfaces, temporal modeling techniques could enhance stability and continuity in detection results.

4.3 Future Work

Future research will aim to address the above limitations and further enhance the practical applicability of the proposed method.

One promising direction is the integration of stereo vision or depth sensing, which may help separate objects from reflections and improve three-dimensional localization accuracy.

In addition, incorporating self-supervised or contrastive learning techniques could reduce the dependence on large labeled datasets and enhance cross-scene generalization.

Another potential direction lies in the use of neural architecture search or neuroevolution algorithms to automatically explore optimal configurations under specific computational constraints.

Furthermore, real-world deployment on embedded platforms remains an essential step.

Although the current model demonstrates theoretical suitability for devices such as Jetson Xavier NX and RK3588, further work will include hardware-level optimization and field testing on actual unmanned surface vessels to evaluate latency, power consumption, and system robustness in real operational environments.

References

[1] X.Zhang and Y.Zhao, “Small object detection in remote sensing images with attention mechanisms,” Remote Sensing, 2022.

[2] Z.Zhao et al., “Light-YOLOv5: Lightweight real-time object detection network,” Sensors, 2023.

[3] C.Wang, “Water surface floating waste detection based on improved YOLOv8,” IEEE Access, 2024.

[4] H.Chen et al., “Feature enhancement for maritime object detection,” Ocean Engineering, 2022.

[5] Y.Fan et al., “Multi-scale attention network for floating waste detection,” Applied Sciences, 2023.

[6] A.Bochkovskiy, C.Y.Wang and H.Y.Liao, “YOLOv4: Optimal speed and accuracy of object detection,” arXiv preprint, arXiv:2004.10934, 2020.

[7] G.Jocher et al., “Ultralytics YOLOv8 documentation,” Ultralytics, 2023.

[8] J.Redmon and A.Farhadi, “YOLOv3: An incremental improvement,” arXiv preprint, arXiv:1804.02767, 2018.

[9] T.Y.Lin et al., “Feature pyramid networks for object detection,” in Proc. CVPR, 2017.

[10] C.Y.Wang, A.Bochkovskiy and H.Y.Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art,” arXiv preprint, arXiv:2207.02696, 2022.

[11] W.Liu et al., “SSD: Single shot multibox detector,” in Proc. ECCV, 2016.

[12] M.Tan and Q.Le, “EfficientDet: Scalable and efficient object detection,” in Proc. CVPR, 2020.

[13] S.Ren, K.He, R.Girshick and J.Sun, “Faster R-CNN: Towards real-time object detection,” in Proc. NeurIPS, 2015.

[14] L.Shen and Q.Cai, “Intrinsic image decomposition for reflective surfaces,” in Proc. CVPR, 2020.

[15] P.Zhou and J.Han, “Deep reflection separation using physical priors,” Pattern Recognition Letters, 2021.

[16] S.Yang et al., “High-reflection suppression for water-surface object detection,” IEEE Access, 2021.

[17] Z.Fan and X.Zhang, “Small sample SAR target recognition using causal modeling,” Acta Automatica Sinica, 2023.

[18] H.Ding and Z.Zhou, “Vision-based plastic bottle detection and grasping system,” Robotics and Autonomous Systems, 2021.

[19] P.Zhang, “Selective harvesting robot key technologies based on machine vision,” Transactions of the Chinese Society for Agricultural Machinery, 2024.

[20] Y.Zhao, “Tomato picking robot vision and control,” Agricultural Engineering Journal, 2024.

만 기(Wan Qi) [정회원]

- Sep 2000 - Jul 2004, Sichuan University, Bachelor of Engineering in Automation.
- Mar 2010 ~ Dec 2012, Guizhou University , Master of Engineering in Control Engineering.
- Dec 2023 ~ Present, Mokwon University (Daejeon, South Korea), Ph.D. Candidate in IT Convergence (Intelligent Systems)

〈관심분야〉

Software Development, Artificial Intelligence & Machine Learning, Robotics & Intelligent Control.

민 병 원(Byung-Won Min) [정회원]

- He received M.S. degree in computer software from Chungang University, Seoul, Korea in 2005.
- He received Ph.D. degree in the dept. of Information and Communication Engineering, Mokwon University, Daejeon, Korea, in 2010.
- He is currently a professor of Mokwon University since 2010.

〈관심분야〉

digital communication systems, Big Data