Journal of Soft Computing Exploration
Homepage: htttps://shmpublisher.com/index.php/joscex
p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Enhancing YOLO performance with attention module for plastic and non-plastic waste detection on water surfaces

Adri Priadana[1], Aris Wahyu Murdiyanto[2], Muhammad Ichwandar Akrianto[3], Heru Cahyono[4]
[1]Department of Informatics, Universitas Jenderal Achmad Yani Yogyakarta, Indonesia
[2]Department of Information System, Universitas Jenderal Achmad Yani Yogyakarta, Indonesia
[3]Department of Information Technology, Universitas Tidar, Indonesia
[4]Department of Digital Business, Universitas AKPRIND Indonesia, Indonesia

Article Info

Article history:

Received Apr 06, 2026
Revised Apr 08, 2026
Accepted May 05, 2026

Keywords:

Waste detection
YOLO
Attention modules
Waste on water surfaces
Object detection

ABSTRACT

The rapid accumulation of plastic waste in aquatic environments poses serious threats to ecosystems, water management systems, and human health. This growing concern creates an urgent need for efficient and accurate detection methods. To address this challenge, this work proposes an approach to enhance YOLO performance by integrating attention modules for plastic and non-plastic waste detection on water surfaces. A comprehensive evaluation is conducted on the Plastic on Water dataset, considering detection accuracy, computational complexity, and inference speed. The results identify YOLO11n as the most effective baseline, achieving a mean Average Precision (mAP) of 96.3% with 2,590,230 parameters, 6.4 GFLOPs, and an inference speed of 18.58 FPS. To further improve performance, several attention modules are integrated into the YOLO11n architecture. Among them, the Convolutional Block Attention Module (CBAM) yields the best performance, achieving an mAP of 96.7% with 2,598,520 parameters and 6.5 GFLOPs, while maintaining real-time performance at 18.26 FPS. The results demonstrate improved detection capability, particularly for small and less prominent objects, with negligible additional computational cost. These findings highlight the effectiveness of attention mechanisms, especially CBAM, in enhancing lightweight object detection models for real-time aquatic waste monitoring.

Corresponding Author:

Adri Priadana,
Department of Informatics,
Universitas Jenderal Achmad Yani Yogyakarta,
Jl. Siliwangi, Ringroad Barat, Banyuraden, Gamping, Sleman, Yogyakarta 55293, Indonesia.
Email: adripriadana3202@gmail.com
https://doi.org/10.52465/joscex.v7i2.44

1. INTRODUCTION

Waste pollution has become an increasingly critical global environmental challenge, particularly in developing countries such as Indonesia. Rapid population growth and rising consumption patterns have significantly increased the volume of solid waste, especially plastic waste, which is persistent and difficult to degrade. According to reports from the World Bank, inefficient waste management systems remain a major

issue in many developing regions, resulting in substantial amounts of unmanaged waste entering natural ecosystems [1]. A considerable portion of this waste is transported into aquatic environments, where rivers act as the primary pathways carrying plastic debris into the oceans [2]. In Indonesia, this problem is further exacerbated by uneven waste management infrastructure, leading to the accumulation of floating debris in rivers and other water bodies. This condition not only threatens aquatic ecosystems but also poses long-term risks to human health through microplastic contamination in the food chain [3]–[5].

Beyond ecological impacts, floating waste also disrupts hydrological systems by obstructing water flow and increasing flood risk, especially in densely populated urban areas. Despite the importance of continuous monitoring, existing practices largely rely on manual observation, which is time-consuming, labour-intensive, and prone to human error. Moreover, such approaches are not scalable for real-time and large-area monitoring. These limitations highlight the need for automated and reliable detection systems capable of identifying waste in aquatic environments under complex and dynamic conditions.

Recent advances in computer vision and deep learning have provided promising solutions for automated environmental monitoring. Among various object detection approaches, the You Only Look Once (YOLO) architecture has gained significant attention due to its ability to perform fast and accurate real-time detection [6]. Continuous improvements in YOLO, from earlier versions to more recent versions, have enhanced both detection performance and computational efficiency. As a result, YOLO has been widely adopted in various real-world applications, including environmental monitoring and waste detection.

However, despite these advancements, deploying YOLO-based models in practical scenarios introduces significant challenges, particularly in terms of computational efficiency. Real-world applications often operate under constrained computational resources, where limitations in memory, processing power, and energy consumption become critical factors [7]. While high-capacity models can achieve superior performance, they are often unsuitable for efficient deployment [8]. Consequently, the development of lightweight models has emerged as a key research direction to balance performance and efficiency applied in real-time scenario [9].

Lightweight variants of YOLO, such as nano-scale models, have been specifically designed to reduce the number of parameters and computational complexity, enabling faster inference and lower resource consumption. Prior studies on deep learning architecture have shown that network architecture optimization can effectively reduce computational costs without substantially degrading performance [10], [11]. This indicates that, beyond adopting lightweight designs, further architectural optimization remains a promising approach to enhance model efficiency. However, in practice, such optimization must be carefully balanced, as lightweight object detection models often experience performance degradation, particularly in complex environments such as water surfaces, where objects are irregularly shaped and influenced by dynamic backgrounds, reflections, and lighting variations. Recent object detection surveys indicate that lightweight models tend to experience performance degradation in visually complex scenarios [12]. This trade-off between efficiency and detection performance remains a fundamental challenge in the development of object detection systems.

To address this limitation, attention mechanisms have been increasingly integrated into deep learning architectures to enhance feature representation. Attention modules enable models to selectively focus on the most relevant spatial and channel-wise features, thereby improving detection performance without significantly increasing computational complexity. Several attention mechanisms, including Squeeze-and-Excitation (SE) [13], Convolutional Block Attention Module (CBAM) [14], Coordinate Attention (CA) [15], and Efficient Channel Attention (ECA) [16], have demonstrated effectiveness in improving model performance across various computer vision tasks [17].

Previous studies have shown that integrating attention mechanisms into object detection models can lead to notable improvements in detection performance while introducing only minimal additional computational overhead [18]–[20]. This suggests that attention mechanisms provide a promising strategy for enhancing lightweight models, particularly in scenarios where both efficiency and detection performance are critical. Moreover, several studies have also explored YOLO-based models for water related object detection tasks such as Mussel-YOLO [21], UDD-YOLO [22], SFD-YOLO [23], RBL-YOLO [24], CRAB-YOLO [25], [26], V9-Benthos [25], and for waste detection tasks in aquatic environments, including the integration with attention mechanisms. Liu et al. [27] proposed FP-YOLO based on YOLOv13n [28] for floating plastic detection, incorporating specialized modules to enhance feature representation and mitigate visual disturbances such as reflections. Their model achieved a mean Average Precision (mAP) of 87.8% and demonstrated strong robustness under various image degradation conditions, highlighting the importance of handling complex

water-surface characteristics. Similarly, Yang et al. [29] developed YOLOv5_CBS by integrating coordinate attention and architectural modifications, achieving an mAP of 92.18%. This study demonstrated that attention mechanisms can improve detection performance in river environments.

In addition, Li et al. [30] introduced PAR-YOLO for water-surface waste detection, focusing on both efficiency and detection performance through architectural enhancements. The proposed model achieved an mAP of 85.53% while maintaining real-time performance. However, the approach primarily relies on complex architectural modifications rather than lightweight attention integration. Furthermore, Yan et al. [31] proposed EFD-YOLO based on YOLOv8n [32], incorporating attention mechanisms to improve detection performance. Their results showed an increase in mAP compared to the baseline YOLOv8n, indicating that lightweight models can benefit from attention integration without significant increases in computational complexity. Despite the extensive exploration of attention mechanisms, their integration into recent YOLO versions, particularly nano-scale variants, for waste detection in aquatic environments remains limited, while systematic evaluation of these versions to establish an optimal baseline prior to enhancement has not been sufficiently explored.

The lack of evaluation of recent YOLO version on nano-scale models and their integration with attention mechanisms for waste detection in complex aquatic environments remains an open research gap. This study aims to address this gap by exploring and proposing an enhanced YOLO-based model that integrates an attention mechanism into a YOLO nano model for detecting plastic and non-plastic waste on water surfaces. The proposed approach is designed to improve detection performance while maintaining computational efficiency, enabling practical deployment in real-time environmental monitoring systems. By focusing on the balance between detection performance and efficiency, this research contributes to the development of robust and scalable solutions for automated waste detection in complex aquatic environments. The main contributions of this work are summarized as follows:

1) This study presents a comparative analysis of several recent YOLO nano versions and identifies YOLO11n [33] (nano variant) as the most effective model, achieving the highest mAP on the Plastic on Water dataset [34]. This finding establishes YOLO11n [33] as a strong baseline for waste detection in aquatic environments.
2) This work investigates the effect of incorporating attention mechanisms into the YOLO11n [33] as the baseline model. Experimental results demonstrate that integrating CBAM [14] with YOLO11n [33] yields the highest mAP compared to other evaluated attention modules, establishing it as the most effective integration for improving detection performance.

2. METHOD

This study proposes an improved YOLO11n [33] model by incorporating an attention mechanism into the baseline network. Specifically, CBAM [14] is integrated within the last C3k2 block of the YOLOv11n [33] backbone, forming a modified block referred to as C3k2A, as shown in Figure 1. This design aims to enhance feature representation by enabling the network to focus on more informative spatial and channel-wise features. The resulting architecture maintains computational efficiency, with a total of 2,598,520 parameters and 6.5 GFLOPs.

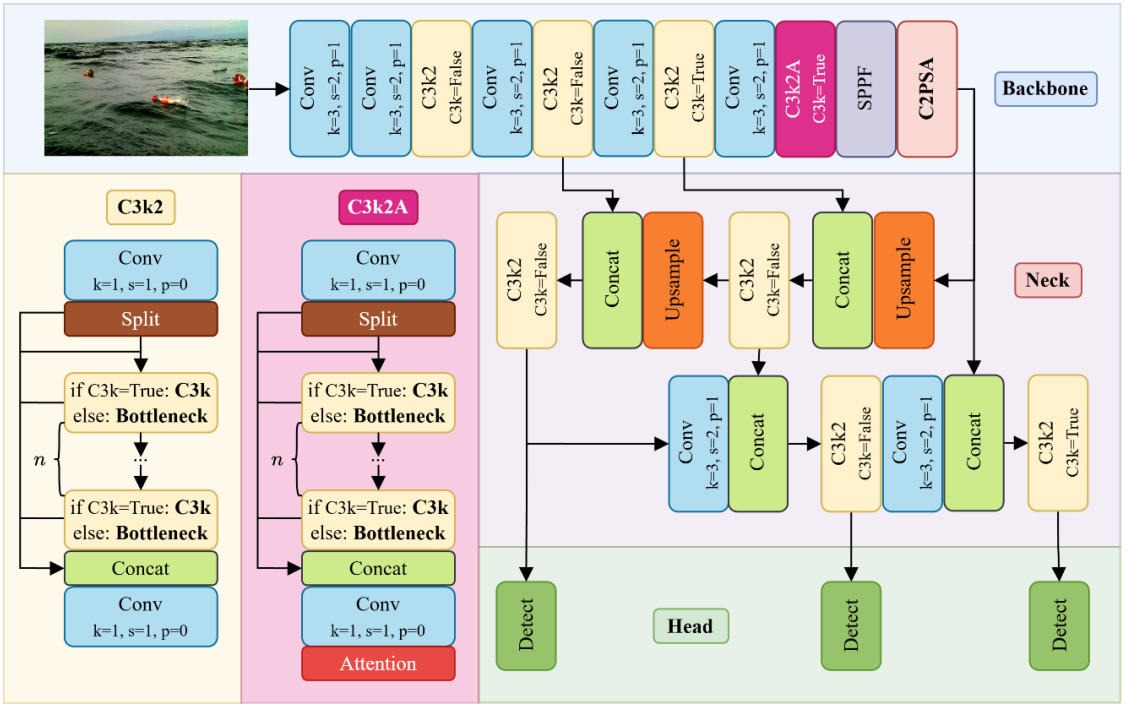

Figure 1. The proposed improved YOLO11n [33] model by incorporating an attention mechanism within the last C3k2 block of the YOLOv11n backbone, forming a modified block referred to as C3k2A

Overall Network

Built upon the foundation of YOLOv8 [32], the YOLO11 [33] architecture introduces a refined backbone design through the C3k2 module, which replaces the earlier C2f block. This module maintains the core principle of cross-stage partial connections while introducing a configurable internal structure governed by the C3k setting. Specifically, the input feature map is first partitioned into two channel groups, where only a selected portion undergoes a sequence of bottleneck transformations, while the remaining channels act as a shortcut path to preserve original information. The transformed branch consists of stacked convolutional layers combined with normalization and activation, forming either a standard bottleneck or an expanded variant with additional convolutional depth when the C3k configuration is activated.

The C3k module splits the input feature map into two branches. The one passes through a sequence of convolutional operations and lightweight bottleneck layers. The second serves as a shortcut path to preserve gradient flow. These outputs from both branches are then concatenated and fused. It enables the network to capture both refined and original features. This design improves representation capability with reduced redundancy, which makes it suitable for real-time object detection tasks. The bottleneck block is a compact unit that consists of two of 3×3 convolutional layers. The first one reduces the number of channels (compression) and the second one restores the channel dimension (expansion) to match the input. It incorporates a residual (skip) connection that adds the input directly to the output, helping to stabilize training and mitigate gradient vanishing. These bottleneck units within C3k2 are stacked in a lightweight manner to deepen the network while keeping the parameter count low. This approach enables efficient learning of discriminative features without significantly increasing computational cost.

This selective processing reduces redundancy while enabling deeper feature extraction. Notably, earlier layers in the backbone allocate a smaller fraction of channels to the transformation branch to prioritize efficiency, whereas deeper layers adopt a more balanced split, allowing richer semantic representation. The outputs from both branches are subsequently concatenated and fused, ensuring effective feature reuse and stable gradient propagation across the network. As an improvement, an attention mechanism is integrated and located as the last layer of C3k2, forming a modified block referred to as C3k2A.

Complementing the backbone improvements, YOLO11 [33] incorporates the C2PSA module as an advanced feature refinement unit that integrates attention mechanisms into the partial connection framework. Unlike conventional attention modules that operate on the entire feature map, C2PSA selectively embeds a

Position-Sensitive Attention mechanism within a split feature pathway. After dividing the input channels, one branch is processed through an attention-enhanced pipeline consisting of multi-head self-attention and feed-forward transformations, while the other branch bypasses these operations. The attention mechanism captures long-range spatial dependencies by computing relationships between different feature positions, enabling the model to focus on structurally important regions. Meanwhile, the feed-forward network further enriches feature representations through non-linear transformations. Residual connections are applied within the attention branch to stabilize optimization and mitigate gradient degradation. After processing, both branches are recombined, allowing the model to benefit from both preserved low-level details and context-aware high-level features, ultimately improving detection performance in complex scenes.

Finally, the training objective of YOLO11 [33] is formulated as a composite loss function that integrates localization, distribution, and classification components as defined in Equation (1).

$$L = \lambda_{\text{CIoU}} L_{\text{CIoU}} + \lambda_{\text{DFL}} L_{\text{DFL}} + \lambda_{\text{Cls}} \tag{1}$$

The weighting coefficients λ_{CIoU}, λ_{DFL}, and λ_{Cls} control the contribution of each component, ensuring a balanced optimization between accurate localization and reliable classification. L_{CIoU} (Complete Intersection over Union loss) [35] measures the detection performance of predicted bounding boxes by considering overlapping area, center distance, and aspect ratio consistency, which is described in Equation (2).

$$L_{\text{CIoU}} = 1 - \text{IoU} + \frac{\rho^2(\mathbf{b}, \mathbf{b}^{gt})}{c^2} av. \tag{2}$$

IoU is the Intersection over Union between the predicted bounding box and the ground truth. $\mathbf{b}$ and $\mathbf{b}^{gt}$ denote the center coordinates of the predicted and ground-truth bounding boxes, respectively. $\rho(\cdot)$ represents the Euclidean distance between two points. c is the diagonal length of the smallest enclosing box covering both predicted and ground-truth boxes. v measures the consistency of aspect ratio between the two boxes, defined in Equation (3).

$$v = \frac{4}{\pi^2}\left(arctan\frac{w^{gt}}{h^{gt}} - \frac{w}{h}\right)^2, \tag{3}$$

where w, h, and w^{gt}, h^{gt} are the width and height of the predicted and ground-truth boxes, respectively. α is a positive trade-off parameter defined in Equation (4).

$$\alpha = \frac{v}{(1 - \text{IoU}) + v}. \tag{4}$$

The L_{DFL} (Distribution Focal Loss) [36] refines bounding box regression by modeling the probability distribution of discrete bounding box coordinates, leading to more precise localization, which is defined in Equation (5).

$$L_{\text{DFL}} = -\sum_{i=1}^{N}\sum_{k=0}^{K} y_{i,k} \log\left(p_{i,k}\right), \tag{5}$$

where N is the number of bounding box predictions. K is the number of discrete bins used to model bounding box offsets. $y_{i,k}$ is the ground-truth distribution (usually soft labels between two adjacent bins). $p_{i,k}$ is the predicted probability for bin k.

The L_{Cls} represents the classification loss, typically implemented using binary cross-entropy or focal loss to handle class imbalance, which is defined in Equation (6).

$$L_{\text{Cls}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)\right], \tag{6}$$

where N is the number of samples (or anchors). $y_i \in \{0,1\}$ is the ground-truth class label. $p_i \in [0,1]$ is the predicted probability for the positive class.

Incorporating attention mechanisms has been shown to improve detection performance in many studies [18]–[20]. Motivated by this, the present work integrates an attention module into the network. Specifically, in

the YOLO11n [33] model, the final C3k2 block in the backbone is replaced with a modified C3k2A block that embeds an attention module to enhance feature representation.

Convolutional Block Attention Module (CBAM)

CBAM [14] is a lightweight attention mechanism that refines feature representations by sequentially applying channel and spatial attention as shown in Figure 2.

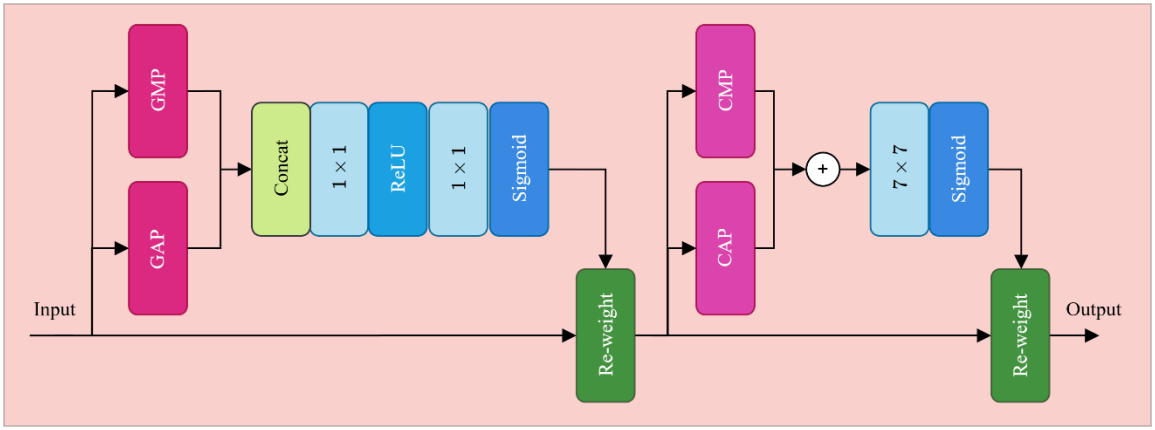

Figure 2. Convolutional block attention module (CBAM) [14]

Given an input feature map $F \in \mathbb{R}^{C \times H \times W}$, the channel attention module first aggregates spatial information using Global Average Pooling (GAP) and Global Max Pooling (GMP), producing two descriptors F_{avg}^{c} and F_{max}^{c}, each of size $C \times 1 \times 1$. These descriptors are passed through two shared 1×1 convolution layers with a reduction ratio r, where the first convolution reduces the channel dimension from C to C/r, followed by a non-linear activation, and the second convolution restores it back to C. The outputs from both pooling branches are then summed up and activated by a sigmoid function to produce the channel attention map shown in Equation (7).

$$M_c(F) = \sigma\left(W_1\left(\delta\left(W_0(F_{avg}^{c})\right)\right) + W_1\left(\delta(W_0(F_{max}^{c}))\right)\right), \tag{7}$$

where $F_{avg}^{c} \in \mathbb{R}^{C \times 1 \times 1}$ is obtained by applying Global Average Pooling (GAP) over spatial dimensions and $F_{max}^{c} \in \mathbb{R}^{C \times 1 \times 1}$ is obtained via Global Max Pooling (GMP). $W_0 \in \mathbb{R}^{C \times \frac{C}{r}}$ and $W_1 \in \mathbb{R}^{\frac{C}{r} \times C}$ are learnable weight matrices corresponding to two 1×1 convolutions. r is the channel reduction ratio. $\delta(\cdot)$ denotes the Rectified Linear Unit (ReLU) activation function and $\sigma(\cdot)$ denotes the Sigmoid activation function. The refined feature is obtained via channel-wise multiplication shown in Equation (8).

$$F' = M_c(F) \odot F. \tag{8}$$

where $\odot$ denotes element-wise multiplication, with $M_c(F)$ broadcast along spatial dimensions.

Subsequently, the spatial attention module focuses on "where" important information is located. The feature F' is compressed along the channel axis using average pooling (CAP) and max pooling (CMP), resulting in two spatial maps F_{avg}^{s} and F_{max}^{s} of size $1 \times H \times W$. These maps are concatenated and passed through a convolution layer with a 7×7 kernel, followed by a Sigmoid activation to generate the spatial attention map shown in Equation (9).

$$M_s(F') = \sigma\left(f^{7\times7}\left(\left[F_{avg}^{s}; F_{max}^{s}\right]\right)\right). \tag{9}$$

where $F_{avg}^{s} \in \mathbb{R}^{1 \times H \times W}$ is obtained by applying average pooling along the channel dimension and $F_{max}^{s} \in \mathbb{R}^{1 \times H \times W}$ is obtained via max pooling along the channel dimension. $[\cdot;\cdot]$ denotes channel-wise concatenation,

resulting in a feature map of size $2 \times H \times W$. $f^{7\times7}(\cdot)$ represents a convolution operation with a kernel size of 7×7, producing a spatial attention map. Finally, the spatial attention map is applied to the feature as shown in Equation (10).

$$F'' = M_s(F') \odot F', \tag{10}$$

where $\odot$ denotes element-wise multiplication, with $M_s(F')$ broadcast along channel dimensions. Through this sequential attention mechanism, CBAM adaptively emphasizes informative features along both channel and spatial dimensions, leading to improved feature representation.

Although YOLO11 already incorporates attention through the C2PSA module, integrating an additional attention module can still improve performance because both operate in a complementary manner. [14] C2PSA mainly captures global contextual dependencies using self-attention, whereas CBAM focuses on lightweight channel and spatial recalibration, emphasizing "what" and "where" features are important at a finer scale. Moreover, placing CBAM [14] in the backbone (e.g., within the final C3k2 block) allows early refinement of high-level features before multi-scale aggregation, while C2PSA typically acts at later stages. This multi-stage attention strategy enhances feature representation progressively, combining efficient local refinement with global reasoning, which can increase the detection performance.

Implementation Configuration

All experiments are implemented using the PyTorch library with the latest version of YOLO framework environment provided by Ultralytics. The training process is conducted on a system equipped with an NVIDIA GeForce 1080Ti GPU featuring 11 GB of memory. The network is trained for 300 epochs, with a batch size empirically determined to be 8. Following the default settings, the initial learning rate is set to 10^{-2} with an automatic scheduler applied as defined in the framework and a momentum coefficient of 0.9 to accelerate convergence while maintaining training stability. Input images are resized to 640 × 640 during both training and inference. Data augmentation strategies provided by the framework are utilized, including geometric and photometric transformations such as mosaic, Hue, Saturation, and Value (HSV), flipping, scaling, with default probabilities and parameters. A fixed random seed of 0 is used to ensure experimental reproducibility. During inference, the confidence threshold is set to 0.25, while Non-Maximum Suppression (NMS) is applied with an IoU threshold of 0.7 to eliminate redundant detections. The model is not initialized using any pretrained weights. Stochastic Gradient Descent with Muon (MuSGD) is used as an optimizer.

In terms of loss configuration, the weighting factors are kept at their default settings, assigning values of 7.5 for bounding box regression, 0.5 for classification, and 1.5 for DFL. In this study, the evaluation is conducted using Average Precision (AP) at a fixed Intersection over Union (IoU) threshold of 0.5. To evaluate real-time performance, inference speed is measured on a separate system powered by an Intel Core i7-9750H CPU running at 2.60 GHz. The model efficiency is then assessed based on average frames per second (FPS), providing insight into its suitability for real-time deployment scenarios.

3. RESULTS AND DISCUSSIONS

The experimental results begin with a comparative evaluation of several recent YOLO nano variants on the Plastic on Water dataset [34], considering both detection performance and computational efficiency. The Plastic on Water dataset [34], sourced from the Roboflow platform, comprises 3,597 images divided into three subsets: 2,877 images for training, 360 for validation, and 360 for testing. It includes two categories, namely plastic and non-plastic. As shown in Table 1, earlier lightweight models such as YOLOv5n [37] and YOLOv6n [38] achieve mAP values of 95.6% and 95.0%, respectively, with 2,508,854 and 4,238,342 parameters. More advanced architectures, including YOLOv8n [32] and YOLOv10n [39], further improve performance to 96.0% and 96.1%, while maintaining moderate computational costs. Among all evaluated models, YOLO11n [33] achieves the highest mAP of 96.3% with only 2,590,230 parameters and 6.4 GFLOPs. This result demonstrates that YOLO11n [33] provides the best trade-off between detection performance and efficiency and is therefore selected as the baseline model for subsequent experiments in aquatic waste detection.

Although some variants such as YOLO26n [40] achieve competitive performance (96.0% mAP) with lower computational complexity (5.8 GFLOPs), they do not surpass YOLO11n [33] in terms of detection performance. Similarly, YOLO12n [41] and YOL13n [28] show slightly lower mAP values of 95.7% and 95.2%, respectively, despite comparable or reduced model sizes. YOLO13n [20] has been utilized as a baseline in several recent studies [27], [39]–[45], suggesting its growing adoption for benchmarking object detection models. These observations indicate that architectural improvements do not necessarily guarantee better performance, further justifying the selection of YOLO11n as a strong and reliable baseline.

Building upon this baseline (96.3% mAP, 2,590,230 parameters, and 6.4 GFLOPs), the impact of incorporating several attention mechanisms into the modified C3k2A block of YOLO11n [33] is evaluated under consistent architectural settings. The integration of the SE [13] module results in a slight decrease in

performance to 96.2% (−0.1% mAP), while increasing the number of parameters to 2,598,422 (+8,192) with no significant change in GFLOPs. The ECA [16] module maintains baseline performance at 96.3% (±0.0% mAP) without introducing significant additional parameters or computational cost, demonstrating its efficiency but limited effectiveness in improving detection performance.

In contrast, the CA [15] module improves detection performance to 96.5% (+0.2% mAP), with a marginal increase in parameters to 2.60M (+6,680) and no significant increase in GFLOPs. This indicates that CA provides a favorable trade-off between detection performance improvement and computational overhead. Most notably, the integration of CBAM [14] achieves the highest performance, reaching 96.7% mAP (+0.4% compared to the baseline). This improvement is obtained with only a negligible increase in complexity, resulting in 2,598,520 parameters (+8,290) and 6.5 GFLOPs (+0.1 GFLOPs).

Table 1. Experimental result comparison on plastic on water dataset [34]

Models	Parameters	GFLOPs	Average Speed (FPS)	mAP
YOLOv3-tiny [46]	12,133,156	19.0	10.93	93.3
YOLOv5n [33]	2,508,854	7.2	22.02	95.6
YOLOv6n [38]	4,238,342	11.8	20.93	95.0
YOLOv8n [32]	3,011,238	8.2	20.68	96.0
YOLOv10n [47]	2,707,820	8.4	19.61	96.1
YOLO11n [33]	2,590,230	6.4	18.58	**96.3**
YOLOv12n [41]	2,568,438	6.5	15.76	95.7
YOLOv13n [28]	2,460,301	6.4	10.73	95.2
YOLO26n [40]	2,504,580	5.8	18.52	96.0
YOLO11n [33] with SE [13]	2,598,422	6.4	18.35	96.2
YOLO11n [33] with ECA [16]	2,590,233	6.4	18.53	96.3
YOLO11n [33] with CA [15]	2,596,910	6.4	18.36	96.5
YOLO11n [33] with CBAM [14]	2,598,520	6.5	18.26	**96.7**

These results clearly demonstrate that while all attention mechanisms introduce minimal computational overhead, their impact on detection performance varies significantly. Among them, CBAM [14] provides the most substantial improvement, enhancing feature representation through combining spatial and channel attention. Therefore, the YOLO11n [33] with CBAM [14] configuration is established as the most effective model in this study, achieving the highest mAP with only a marginal increase in parameters and computational cost, making it well-suited for real-time aquatic waste detection.

The inference speed of several recent YOLO nano versions is evaluated in terms of average FPS to assess their suitability for real-time applications. As shown in Table 1, YOLOv5n [33] achieves the highest speed at 22.02 FPS, followed by YOLOv6n [38] and YOLOv8n [32] with 20.93 FPS and 20.68 FPS, respectively. More recent models such as YOLOv10n [47] and YOLO26n [40] exhibit comparable performance at 19.61 FPS and 18.52 FPS. The selected baseline, YOLO11n [33], operates at 18.58 FPS, which is slightly lower than earlier nano versions but remains within the real-time range. In contrast, YOLOv12n [41] and YOLOv13n [28] show reduced speeds of 15.76 FPS and 10.73 FPS, respectively, indicating a trade-off between architectural modifications and inference efficiency. Despite not being the fastest model, YOLO11n [33] provides a favorable balance between detection performance and inference speed, supporting its selection as the baseline model.

Building upon this baseline, the impact of integrating attention mechanisms into YOLO11n [33] is further analyzed in terms of inference speed. The results show that the addition of attention modules introduces only minimal overhead. Specifically, YOLO11n [33] with ECA [16] achieves 18.53 FPS (−0.05 FPS compared to the baseline), while YOLO11n [33] with SE [13] and CA [15] result in 18.35 FPS (−0.23 FPS) and 18.36 FPS (−0.22 FPS), respectively. The proposed YOLO11n [33] with CBAM [14] model operates at 18.26 FPS, representing a slight decrease of only 0.32 FPS compared to the baseline YOLO11n [33]. Despite this minor reduction in speed, the CBAM-integrated model delivers the highest mAP, demonstrating that the proposed approach effectively enhances detection performance with only negligible impact on inference speed. This confirms that the YOLO11n [33] with CBAM [14] configuration maintains real-time capability while achieving superior detection performance.

Analyzing per-class AP shows that the YOLO11n model with an attention mechanism integrated provides varying degrees of improvement across two classes as shown in Table 2. For plastic, the CBAM-enhances model even reaches 94.7 AP, which outperforms the baseline, and other attention integrations showing its advantage in focusing on more discriminative features for plastic waste detection. By comparison, the non-

plastic class is less variable across models with SE marginally ahead at 98.9 AP, and others clustering closely around 98.6–98.7 AP for this class, suggesting that it is already well learnt by the baseline model.

Table 2. Per-class detection performance result on water dataset [34]

Models	Plastic AP	Non-Plastic AP	mAP
YOLO11n [33]	93.9	98.6	**96.3**
YOLO11n [33] with SE [13]	93.6	98.9	96.2
YOLO11n [33] with ECA [16]	94.0	98.7	96.3
YOLO11n [33] with CA [15]	94.3	98.7	96.5
YOLO11n [33] with CBAM [14]	94.7	98.6	**96.7**

Further analysis of the confusion matrix, as shown in Figure 3, highlights that all models show no direct misclassification between the plastic and non-plastic classes, as indicated by a zero value for cross-class predictions. This indicates that the models are highly effective in distinguishing between the two object categories. However, differences appear in terms of missed detections (false negatives) and false positives from the background. For the plastic class, the YOLO11n [33] model with CBAM [14] achieves the highest correct classification rate (0.93), with relatively low missed detections (0.07), while the ECA [16] variant shows the lowest performance (0.90) with increased background neglect (0.10). A similar trend is observed for the non-plastic class, where most models maintain a high true positive rate (0.98–0.99). It indicates that this class is consistently easier to detect. In terms of background confusion, a marked variation is observed. SE [13] model produces the highest false positives against the plastic class (0.63). It indicates a tendency to over detect plastic objects in the background region. In the other hand, ECA [16] and CA [15] show a more balanced behavior with lower false positives against plastic (0.43). CBAM [14] model maintains a competitive balance, matching the baseline in plastic and non-plastic detection while slightly improving background predictions against plastic (0.53). Overall, CBAM [14] offers a stable trade-off between detection accuracy and background suppressions.

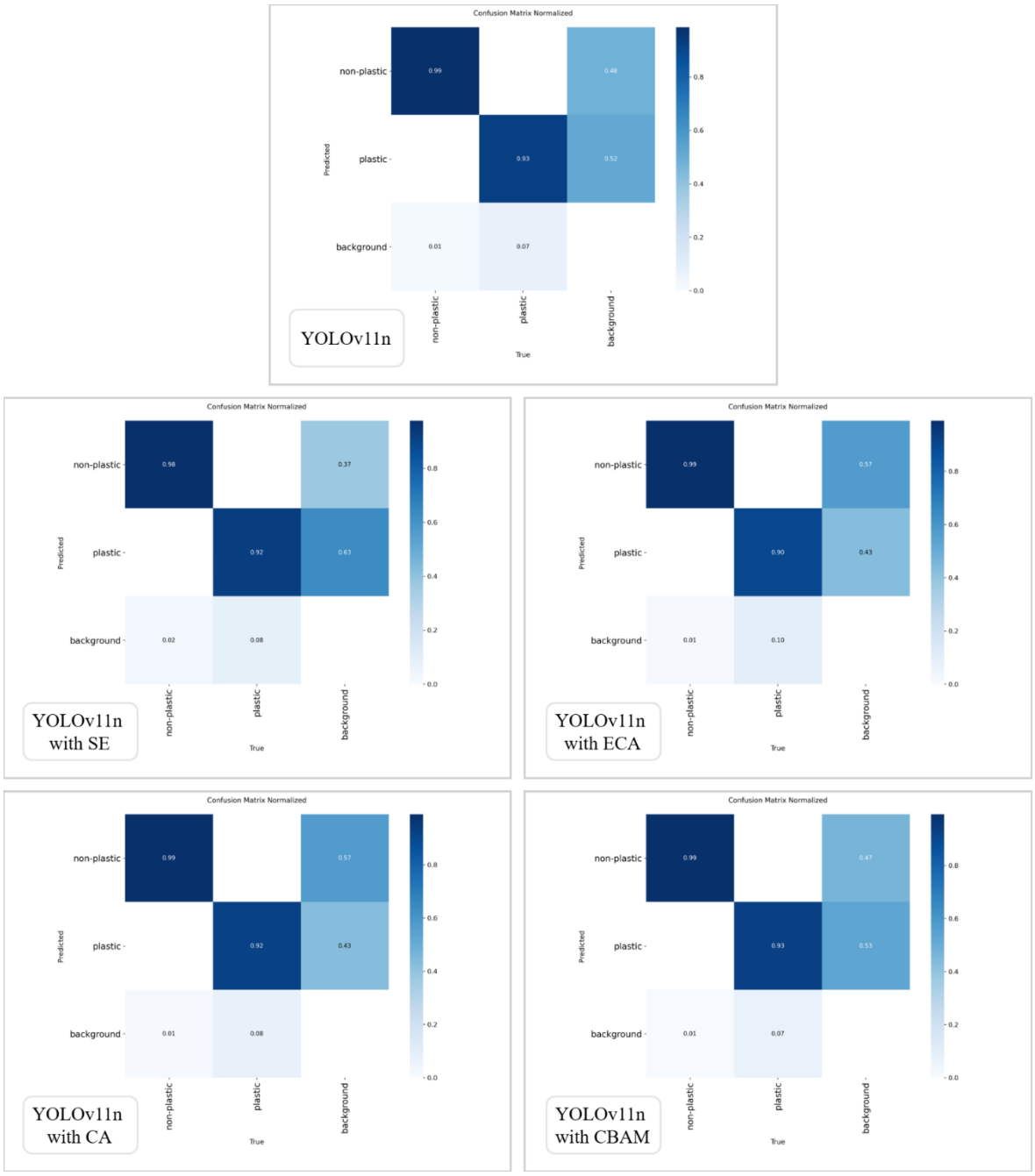

Figure 3. Confusion matrix comparison

To further analyze the contribution of the attention mechanism, an ablation study is conducted by decomposing the CBAM [14] module into its individual components. As shown in Table 3, the baseline YOLO11n achieves an mAP of 96.3. When only channel attention is applied, the performance decreases to 95.3, while spatial attention alone results in 95.9. In contrast, the full CBAM [14] module achieves the highest performance of 96.7. These results indicate that neither channel attention nor spatial attention alone is sufficient to improve detection performance in this task. However, when combined, they provide complementary benefits by enhancing both feature representation across channels and spatial localization. This complementary effect enables CBAM [14] to outperform the baseline, demonstrating that the integration of both attention mechanisms is essential for achieving performance gains.

Table 3. Ablation study on water dataset [34]

Models	mAP
YOLO11n [33]	96.3
YOLO11n [33] with only channel attention of CBAM [14]	95.3
YOLO11n [33] with only spatial attention of CBAM [14]	95.9
YOLO11n [33] with CBAM [14]	96.7

Figure 4 presents a qualitative comparison between the baseline YOLO11n [33] model (left) and the improved YOLO11n [33] with CBAM [14] (right) in detecting plastic and non-plastic objects on the water surface. As illustrated, several selected examples highlight cases where the baseline YOLO11n [33] model fails to detect certain plastic and non-plastic objects, particularly when the objects are small or less visually prominent. In these same scenarios, the enhanced YOLO11n [33] with CBAM [14] successfully detects the missed objects, demonstrating improved detection capability. This enhancement is especially noticeable for small-scale objects, suggesting that the integration of CBAM [14] enables the model to better focus on informative features. Overall, these qualitative results support the quantitative findings, indicating that the adding CBAM [14] improves detection performance in challenging scenarios.

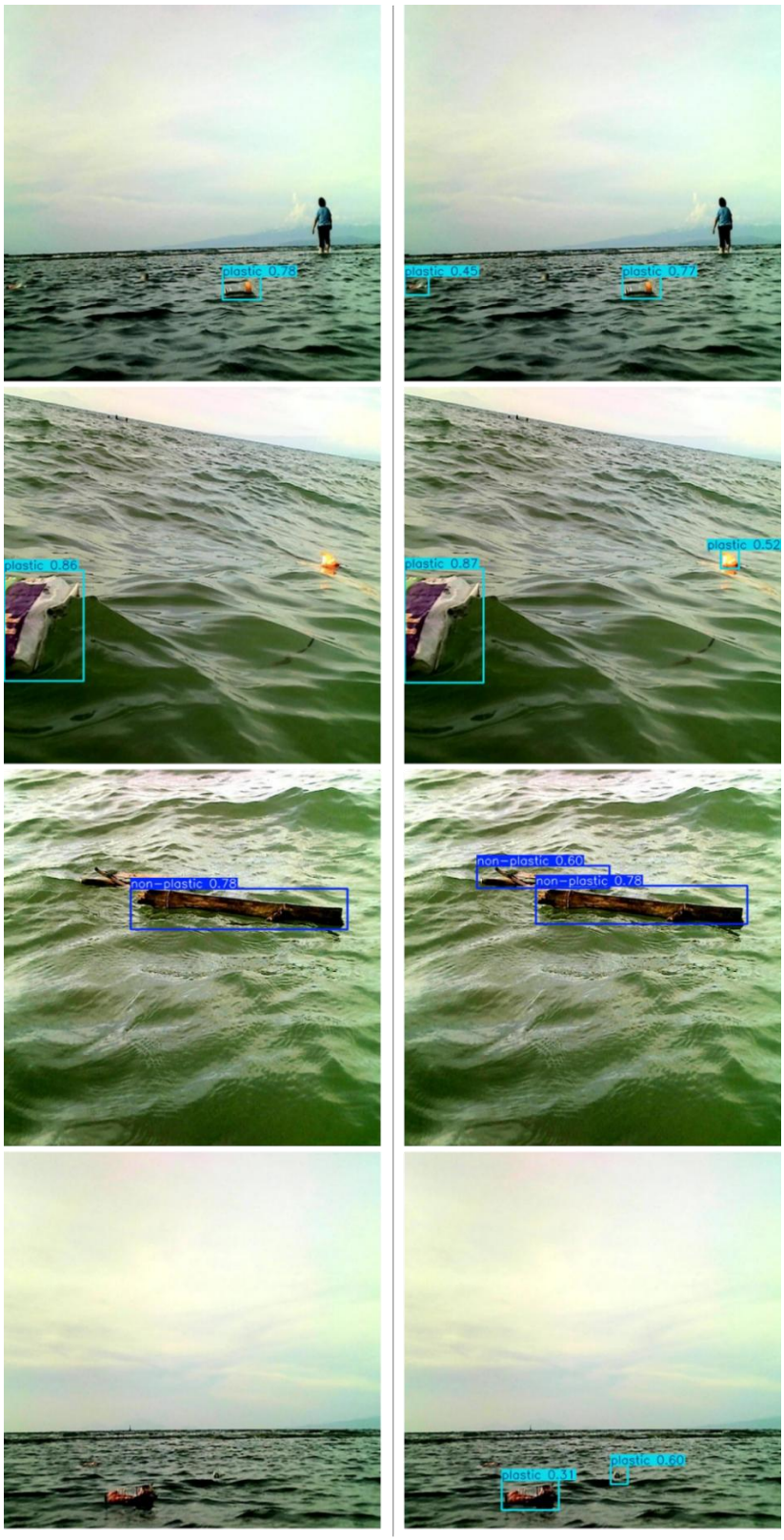

Figure 4. The qualitative results of the improved YOLO11n model with CBAM (right) compared to the YOLO11n model (left) as the baseline

.

Qualitative examples are provided to show the challenges of detecting waste objects in complex environments such as water surfaces, where objects are irregularly shaped and influenced by dynamic backgrounds, reflections, and lighting variations as shown in Figure 5 (top). These factors reduce object visibility and introduce significant background interference, which makes the detection task more difficult. Despite these challenges, the model is still able to correctly detect plastic and non-plastic objects. Nevertheless, the improved YOLO11n with CBAM model still struggles to detect several small objects in some cases. Figure 5 (bottom) shows situations where objects are either missing detected. These errors typically occur when objects are partially occluded or have low contrast with the background. These observations indicate that, although the proposed approach improves overall detection performance, it still faces limitations under highly complex scenarios and requires further improvement.

Figure 5. Qualitative results of the proposed YOLO11n with CBAM model. The top row illustrates detection results under challenging environmental conditions, including irregular object shapes, dynamic water surfaces, reflections, and varying lighting. The bottom row presents representative failure cases, including missed detections caused by difficulties in detecting small or low-contrast objects

4. CONCLUSION

This study presents a comprehensive evaluation of several YOLO nano variants for waste detection on the Plastic on Water dataset, considering detection performance, computational complexity, and inference speed. The experimental results demonstrate that YOLO11n achieves the best overall performance, attaining the highest mAP of 96.3% while maintaining a lightweight architecture with 2,590,230 parameters and 6.4 GFLOPs. In terms of inference speed, YOLO11n operates at 18.58 FPS, which remains suitable for real-time applications. Although not the fastest model, YOLO11n provides a favorable balance between detection performance and efficiency, making it a strong baseline for aquatic waste detection. These findings highlight the importance of architectural design in achieving an optimal trade-off between performance and efficiency in lightweight object detection models.

Extending this baseline, this study further investigates the integration of attention mechanisms within YOLO11n model. The results show that incorporating the CBAM into the modified C3k2A block yields the best performance, achieving an improved mAP of 96.7% while maintaining efficient computation with 2,598,520 parameters and 6.5 GFLOPs. The model operates at an inference speed of 18.26 FPS, representing only a slight reduction compared to the baseline. Both quantitative and qualitative evaluations confirm that the YOLO11n model with CBAM shows improved detection in some challenging cases, including less prominent objects. Therefore, the proposed approach offers an effective and efficient solution for real-time waste detection in aquatic environments. Future work will focus on extending the evaluation to multiple datasets and more diverse data sources to assess cross-dataset generalization. In addition, further experiments will be conducted

to evaluate model robustness under challenging real-world conditions, such as strong reflections, dynamic water surfaces, occlusions, and low-light environments. To improve the reliability of the findings, future studies will also incorporate repeated training runs and statistical analysis, including variance and confidence intervals. Moreover, through improved data analysis and the use of datasets with more explicit scale variation to evaluate small-object versus large-object behavior. Finally, further investigation into the trade-off between detection performance and computational efficiency, as well as the development of more task-specific attention mechanisms, will be explored to enhance real-time deployment in practical applications.

REFERENCES

[1] S. Kaza, L. C. Yao, P. Bhada-Tata, and F. Van Woerden, *What A Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*. World Bank Publications, 2018.

[2] L. J. J. Meijer, T. van Emmerik, R. van der Ent, C. Schmidt, and L. Lebreton, "More than 1000 rivers account for 80\% of global riverine plastic emissions into the ocean," *Sci. Adv.*, vol. 7, no. 18, 2021.

[3] A. Al Mamun, T. A. E. Prasetya, I. R. Dewi, and M. Ahmad, "Microplastics in human food chains: Food becoming a threat to health safety," *Sci. Total Environ.*, vol. 858, p. 159834, 2023.

[4] A. Jyoti, D. Kumar, P. Rasane, S. Ercisli, and J. Singh, "Health implications of microplastic exposure and sustainable solutions," *Environ. Sci. Eur.*, vol. 38, no. 1, p. 33, 2026.

[5] K. Dulta and others, "Microplastic: From marine pollution to the human food chain," *Food Biosci.*, vol. 73, p. 107613, 2025.

[6] R. Sapkota and M. Karkee, "Ultralytics YOLO Evolution: An Overview of YOLO26, YOLO11, YOLOv8 and YOLOv5 Object Detectors," *arXiv Prepr.*, 2026.

[7] R. Cordova-Cardenas, D. Amor, and A. Gutierrez, "Edge AI in Practice: A Survey and Deployment Framework for Neural Networks on Embedded Systems," *Electronics*, vol. 14, no. 24, p. 4877, 2025.

[8] K. Sun, X. Wang, X. Miao, and Q. Zhao, "A review of AI edge devices and lightweight CNN and LLM deployment," *Neurocomputing*, vol. 614, p. 128791, 2025.

[9] 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, 2023.

[10] J. Chen and others, "Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12021–12031.

[11] A. Priadana, D. L. Nguyen, X. T. Vo, M. D. Putro, G. Cao, and K. H. Jo, "A High-Accuracy and Faster Face Recognizer Supporting Biometric Continuous Authentication," *IEEE Trans. Ind. Informatics*, vol. 21, no. 8, pp. 6220–6229, 2025.

[12] L. Liu and others, "Deep Learning for Generic Object Detection: A Survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, 2019.

[13] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, 2020.

[14] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *ECCV*, 2018, pp. 3–19.

[15] Q. Hou, D. Zhou, and J. Feng, "Coordinate Attention for Efficient Mobile Network Design," in *CVPR*, 2021, pp. 13708–13717.

[16] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient Channel Attention for Deep CNN," in *CVPR*, 2020, pp. 11531–11539.

[17] M.-H. Guo and others, "Attention mechanisms in computer vision: A survey," *Comput. Vis. Media*, vol. 8, no. 3, pp. 331–368, 2022.

[18] J. An, M. D. Putro, A. Priadana, Y. Lee, J. Kim, and K. H. Jo, "YOLOv5 with Combination of Coordinate Attention and CBAM," in *IECON*, 2023.

[19] M. Yurdakul and S. Tasdemir, "BC-YOLO: MBConv-ECA based YOLO framework for blood cell detection," *Signal, Image Video Process.*, vol. 19, no. 9, 2025.

[20] A. Priadana and others, "HFD-YOLO: Improved YOLO Network for Human Fall Detection," *IEEE Access*, vol. 13, pp. 41248–41258, 2025.

[21] F. Zhao and others, "Mamba-based super-resolution and YOLOv10 for freshwater mussel detection," *Ecol. Inform.*, vol. 90, p. 103324, 2025.

[22] J. Tao and others, "Diffusion-Enhanced Underwater Debris Detection via YOLOv12n," *Remote Sens.*, vol. 17, no. 23, p. 3910, 2025.

[23] J. Guo and others, "SFD-YOLO: Subsidence funnels detection," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 140, p. 104605, 2025.

[24] F. Zhao and others, "Riverbed litter monitoring using deep learning," *Mar. Pollut. Bull.*, vol. 209, p. 117030, 2024.

[25] F. Zhao and others, "Deep learning-based super-resolution reconstruction and improved YOLOv9 for efficient benthos detection," *Remote Sens. Ecol. Conserv.*, 2026.

[26] F. Zhao and others, "Enhanced hermit crabs detection using YOLOv8," *Mar. Environ. Res.*, vol. 210, p. 107313, 2025.

[27] Z. Liu and others, "Water-aware real-time detection of floating plastic debris via an enhanced YOLOv13 framework," *Expert Syst. Appl.*, vol. 313, p. 131552, 2026.

[28] M. Lei and others, "YOLOv13: Real-Time Object Detection with Hypergraph-Enhanced Adaptive Visual Perception," *arXiv Prepr.*, 2025.

[29] X. et al. Yang, "Detection of River Floating Garbage Based on Improved YOLOv5," *Mathematics*, vol. 10, no. 22, p. 4366, 2022.

[30] N. Li, M. Wang, H. Huang, B. Li, B. Yuan, and S. Xu, "PAR-YOLO: a precise and real-time water surface garbage detection model," *Earth Sci. Informatics*, vol. 18, no. 1, 2025.

[31] Y. Yan, Z. Liang, C. Liu, and T. Zou, "EFD-YOLO: An Improved YOLOv8 Network for River Floating Debris Detection,"

IEEE Geosci. Remote Sens. Lett., vol. 23, 2026.
[32] R. Varghese and M. Sambath, "YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance," in *International Conference on Advances in Data Engineering and Intelligent Computing Systems*, 2024.
[33] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," *arXiv Prepr.*, 2024.
[34] Roboflow Universe, "Plastic on Water Dataset." 2026.
[35] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," no. 2, 2019.
[36] X. Li and others, "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes," in *Advances in Neural Information Processing Systems*, 2020, vol. 33, pp. 21002–21012.
[37] R. Khanam and M. Hussain, "What is YOLOv5: A deep look into the internal features," *arXiv Prepr.*, 2024.
[38] C. Li and others, "YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications," *arXiv Prepr.*, 2022.
[39] Z. Wang and others, "YOLOv13-Cone-Lite: Traffic Cone Detection Algorithm," *Appl. Sci.*, vol. 15, no. 17, p. 9501, 2025.
[40] R. Sapkota, R. H. Cheppally, A. Sharda, and M. Karkee, "YOLO26: Key Architectural Enhancements and Benchmarking," *arXiv Prepr.*, 2025.
[41] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-Centric Real-Time Object Detectors," *arXiv Prepr.*, 2025.
[42] Y. Sanjalawe and others, "HyperFallNet: Human Fall Detection Using YOLOv13," *IEEE Access*, vol. 13, pp. 177111–177126, 2025.
[43] Z. Feng and F. Liu, "IFEM-YOLOv13 for Robust Underwater Object Detection," *Symmetry (Basel).*, vol. 17, no. 9, p. 1531, 2025.
[44] J. Zhang, R. Sun, B. Li, D. Kong, D. Zhao, and J. Zhang, "DAS-YOLOv13: Dual-Axis Attention Model for Defect Detection," *Sensors*, vol. 26, no. 5, p. 1574, 2026.
[45] S. Zhang and others, "DE-YOLOv13-S: Biomimetic Vision-Based Model," *Biomimetics*, vol. 10, no. 11, p. 724, 2025.
[46] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv Prepr. arXiv1804.02767*, 2018.
[47] A. Wang and others, "YOLOv10: Real-Time End-to-End Object Detection," *Adv. Neural Inf. Process. Syst.*, vol. 37, 2024.