International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 14, Issue 1, 2026

Marine Environmental Monitoring Via Attention Enhanced Yolov10 For Debris Detection

Raheela Tabassum[1], Md Fardeen Ismail Khan[2], Mohammed Abdul Akbar khan Noman[3], Md Sahil khan[4], Abdul Waleed[5]

[1]Assistant Professor Department of CSE-AIML, Lords Institute of Engineering and Technology, Hyderabad, India

[2,3,4,5]BTech Students Department of CSE-AIML, Lords Institute of Engineering and Technology, Hyderabad, India

raheela@lords.ac.in ,mohammedfardeenismailkhan@gmail.com, akbarkhan.noman@gmail.com, skhan828631@gmail.com ,itsabdulwaleed@gmail.com

Abstract

Marine debris severely threatens aquatic ecosystems and biodiversity. Automated underwater debris detection is essential for scalable environmental monitoring. This paper proposes a lightweight real-time debris detection framework based on YOLOv10n optimized for edge deployment. The proposed system integrates adaptive preprocessing, feature pyramid fusion, and anchor-free detection mechanisms. Mathematical formulations of detection loss, bounding box regression, and evaluation metrics are presented. Experimental evaluation demonstrates 94.8% mAP@0.5 with 32 FPS inference speed on edge hardware. Comparative analysis confirms reduced computational complexity while maintaining high accuracy. The proposed model provides a scalable and energy-efficient solution for marine environmental surveillance.

Index Terms — YOLOv10n, Marine Debris Detection, Edge AI, Object Detection, Underwater Vision.

I. INTRODUCTION

Oceans and marine ecosystems play a vital role in maintaining environmental balance, supporting biodiversity, regulating climate, and sustaining human livelihoods through fisheries, transportation, and tourism. However, rapid industrialization, urbanization, and improper waste management practices have significantly increased the amount of solid waste entering marine environments. Large quantities of plastic, metal, fishing gear, and other non-biodegradable materials accumulate in oceans every year, posing severe threats to aquatic organisms, coral reefs, and coastal ecosystems. Marine debris not only disrupts natural habitats but also enters the food chain, causing long-term ecological and health related consequences for both marine life and human populations.

Traditional methods for monitoring marine pollution primarily rely on manual surveys, visual inspections, and vessel-based observations. Although these techniques have been widely used, they are time-consuming, labor-intensive, expensive, and limited in spatial and temporal coverage. Furthermore, harsh underwater conditions such as low visibility, light attenuation, turbidity, and dynamic water movements make accurate debris identification extremely challenging. As a result, manual monitoring approaches often fail to provide reliable and continuous data required for effective environmental management and policy formulation.

With the rapid advancement of computer vision and artificial intelligence technologies, deep learning-based object detection methods have emerged as powerful tools for automated environmental monitoring. Convolutional Neural Networks (CNNs) and region-based detection frameworks such as Faster R-CNN, Single Shot Detector (SSD), and You Only Look Once (YOLO) have demonstrated remarkable performance in various visual recognition tasks. These models have been increasingly adopted in marine research for detecting floating and submerged debris from underwater images, videos, and remotely sensed data. Despite their high accuracy, most of these models are computationally expensive and require high-end hardware resources, making them unsuitable for real-time deployment on embedded platforms, underwater drones, and Internet of Things (IoT) devices.

Recent research efforts have focused on developing lightweight and optimized detection models to address the limitations of conventional deep learning frameworks. Modified YOLO architectures, including YOLOv5, YOLOv8, and YOLO-MES, have introduced efficient backbones, feature fusion strategies, and attention mechanisms to improve detection performance while reducing model complexity. Although these approaches have achieved notable improvements, they still struggle to maintain an optimal balance between accuracy, inference speed, and energy consumption in highly constrained environments. Moreover, the need for complex architectural modifications and extensive post-processing further complicates their practical deployment in real-world marine monitoring systems. In this context, YOLOv10n, the nano version of the YOLOv10 family, offers a promising solution for efficient underwater debris detection. YOLOv10n is specifically designed to operate in low-resource environments by incorporating streamlined network architectures, optimized feature pyramids, and decoupled detection heads. These design enhancements significantly reduce the number of parameters and computational operations while preserving strong feature representation capabilities. Consequently, YOLOv10n enables high-speed, low-latency object detection on edge devices without compromising detection accuracy.

This research proposes a comprehensive marine environmental monitoring system based on YOLOv10n for real-time underwater debris detection. The proposed framework integrates systematic data collection, adaptive preprocessing, effective feature extraction, and optimized detection modules to enhance robustness under challenging underwater conditions. The system is capable of identifying multiple categories of marine waste, including plastic bottles, bags, fishing nets, metal fragments, and organic materials, across varying illumination and background complexities. By minimizing computational overhead and energy consumption, the proposed approach facilitates long-term autonomous deployment in marine environments.

The main contributions of this work are summarized as follows. First, a lightweight and scalable underwater debris detection framework is developed using YOLOv10n, enabling efficient real-time operation on embedded platforms. Second, the system incorporates optimized preprocessing and feature extraction techniques to improve detection reliability in low-visibility and noisy underwater conditions. Third, extensive experimental evaluations are conducted to demonstrate the superiority of the proposed model in terms of accuracy, speed, and resource utilization when compared with existing detection approaches. Finally, this research highlights the practical feasibility of deploying deep learning-based monitoring systems for large-scale marine pollution management and conservation initiatives.

The remainder of this paper is organized as follows. Section II reviews related work in marine debris detection and lightweight object detection models. Section III describes the proposed methodology and system architecture in detail. Section IV presents implementation details and experimental setup. Section V discusses performance evaluation and results. Section VI provides a comprehensive discussion of findings and limitations. Section VII outlines future research directions, and Section VIII concludes the paper with key insights and contributions.

II. LITERATURE SURVEY

In recent years, the increasing severity of marine pollution has motivated extensive research on automated debris detection and environmental monitoring systems. Traditional approaches relied primarily on manual inspection, sonar imaging, and handcrafted feature-based computer vision techniques. While these methods provided initial solutions for underwater monitoring, they suffered from limited accuracy, high labor requirements, and poor adaptability to complex marine environments. Consequently, researchers have shifted their focus toward deep learning-based techniques to improve detection reliability and automation. Deep learning frameworks based on Convolutional Neural Networks (CNNs) have demonstrated significant success in object detection and classification tasks. Early studies adopted region-based detection models such as Faster R-CNN and SSD for marine debris identification. These models achieved high detection accuracy by generating region proposals and extracting deep features. However, their multi-stage processing pipelines resulted in high computational complexity and inference latency, limiting their applicability on embedded and real-time platforms.
The YOLO (You Only Look Once) family of object detection models has gained popularity due to its single-stage detection mechanism and real-time performance. Several versions, including YOLOv3, YOLOv4, YOLOv5, and YOLOv8, have been applied to marine litter detection. These models improved detection speed and robustness compared to region-based detectors. Nevertheless, their relatively large model sizes and high processing requirements remained challenging for deployment on low-power underwater devices and IoT platforms.
Ma et al. proposed MLDet, a lightweight deep learning framework for marine litter detection that integrates refined anchor mechanisms and attention modules. Their approach focused on improving feature representation under low-visibility and occlusion conditions while maintaining real-time performance on edge devices. Experimental results demonstrated superior accuracy compared to conventional methods. However, the inclusion of attention mechanisms increased architectural complexity and training overhead. According to the recent work introduced UTD-YOLO, an improved YOLOv5-based model specifically designed for underwater trash detection. Their work addressed underwater image distortions by incorporating specialized preprocessing techniques and adaptive feature fusion strategies. The proposed model achieved enhanced precision and recall on benchmark datasets. Despite these improvements, UTD-YOLO still required substantial computational resources and careful parameter tuning, which restricted its scalability on ultra-constrained platforms. Image enhancement techniques have also been explored to improve underwater detection performance. An et al. proposed DIRBW-Net, an inverted residual network for underwater image

enhancement. By mitigating haze, blur, and color distortion, DIRBWNet improved image quality and subsequently enhanced downstream detection accuracy. Although effective, the integration of separate enhancement and detection networks increased system complexity and processing time.

Chen et al. developed EFS-YOLO, a lightweight detection framework utilizing Ghost Convolution and feature enhancement modules. Although originally designed for industrial defect detection, EFS-YOLO demonstrated strong potential for marine applications due to its reduced computational cost and compact architecture. The model achieved significant reductions in parameters and inference time. However, its adaptation to underwater environments requires further optimization and domain-specific tuning.

Zhang et al. proposed SRFL, a federated learning framework for AI IoT environments based on swarm reputation mechanisms. While not directly focused on marine debris detection, their decentralized learning approach offers potential benefits for collaborative underwater monitoring systems. By enabling distributed training and data privacy, SRFL can enhance scalability and reliability in multi-device marine surveillance networks. However, federated learning introduces additional communication overhead and system complexity.

Several comparative studies have also evaluated deep learning-based litter detection systems. Fulton et al. applied deep visual detection models for robotic marine litter identification, demonstrating the feasibility of autonomous cleanup systems. Cordova et al. conducted a comparative analysis of multiple deep learning models for litter detection, highlighting trade-offs between accuracy, speed, and robustness. These studies emphasized the need for balanced models that perform efficiently under practical deployment constraints. Despite significant progress, existing approaches continue to face limitations in terms of computational efficiency, architectural complexity, and deployment feasibility. Many models rely on customized backbones, attention mechanisms, or multi-stage pipelines that increase training and inference costs. Furthermore, most existing systems require extensive post-processing and hardware acceleration, making them unsuitable for long term autonomous operation in resource-limited marine environments.

To overcome these challenges, recent research has focused on developing ultralightweight detection architectures. YOLOv10n represents a new generation of nanoscale object detection models optimized for low-resource environments. By incorporating streamlined backbones, decoupled detection heads, and optimized feature pyramids, YOLOv10n achieves a favorable balance between accuracy and efficiency. Unlike previous models, it minimizes architectural modifications and reduces dependency on additional processing modules.

In comparison with existing methods, the proposed YOLOv10n-based framework offers improved deployment simplicity, lower memory consumption, and faster inference speed. It addresses the shortcomings of earlier approaches by providing an integrated and scalable solution for real-time underwater debris detection. Therefore, YOLOv10n serves as a suitable foundation for developing practical marine environmental monitoring systems capable of operating on embedded and edge platforms.

Most models require >15m parameters. yolov10n uses ~2.5m parameters, making it ideal for edge devices.

MODEL	TYPE	PARAMS	MAP	LIMITATIONS
FASTER R-CNN	TWO-STAGE	HIGH	HIGH	SLOW INFERENCE
YOLOV5	ONE-STAGE	MEDIUM	HIGH	HEAVY FOR EDGE
UTD-YOLO	MODIFIED YOLOV5	MEDIUM-HIGH	HIGH	COMPLEX ARCHITECTURE
MLDET	ATTENTION-BASED	HIGH	VERY HIGH	TRAINING OVERHEAD
PROPOSED YOLOV10N	ANCHOR-FREE	LOW	HIGH	LIGHTWEIGHT

III. PROPOSED METHOD

A. Problem overview and motivation:

Underwater debris detection is a critical challenge due to the inherent difficulties posed by the underwater environment, including low visibility, light absorption, and water turbidity. traditional image processing techniques often rely on handcrafted features and classical machine learning approaches. while these methods may perform well in some controlled environments, they struggle with generalization when deployed in real-world, complex underwater conditions. the emergence of deep learning has revolutionized object detection, offering improved accuracy but at the cost of requiring high computational resources, which limits their applicability in resource-constrained devices such as underwater drones, iot cameras, and embedded edge platforms.

Among various object detection models, yolo-based models (you only look once) strike a balance between speed and accuracy, making them suitable for real-time applications. however, adapting yolo to underwater debris detection is a challenge due to underwater conditions such as low contrast, motion blur, and background noise. in this paper, we extend the yolov10n model to address these challenges by incorporating enhanced preprocessing, segmentation, and feature extraction strategies that improve detection performance under challenging underwater conditions.

B. Modelling and Objective Function:

Object detection is a key task in computer vision that involves both localizing objects within an image (through bounding boxes) and classifying those objects into predefined categories. Modern object detection systems, especially those based on deep learning, have significantly improved accuracy and efficiency compared to traditional methods, such as template matching or feature-based approaches. Convolutional Neural Networks (CNNs) are widely used in these systems, as they are capable of automatically learning hierarchical features from raw image data. Models like YOLO (You Only Look Once) have revolutionized real-time object detection by combining localization and classification tasks into a single network, significantly improving both speed and accuracy.

Related Theory - YOLO Architecture for Real-Time Detection

The YOLO family of models is designed to achieve real-time object detection by performing both tasks (localization and classification) simultaneously in a single pass through the network. YOLOv10n, a nano-scale version of YOLOv10, further optimizes the architecture to reduce computational overhead, making it suitable for deployment on resource-constrained edge devices such as underwater drones, IoT-enabled cameras, and embedded platforms. Unlike traditional two-stage detectors (e.g., Faster R-CNN), YOLO is a single-stage detector, which simplifies the detection pipeline, reduces inference time, and improves efficiency without compromising on detection performance. By leveraging a composite loss function that balances multiple tasks—such as classification, bounding box regression, and abjectness prediction—YOLOv10n achieves high accuracy with low computational complexity, making it an ideal choice for practical applications like underwater debris detection.

let the input underwater image be represented as:

$i \in r^{(h \times w \times 3)}$

where: h = height of the image, w = width of the image, 3 = color channels (rgb)

The detection function $f_\theta (i)$is defined as: $f_\theta (i)=\{(b_i,c_i,s_i)\}_{(i=1)}^n$

where: $b_i=(x,y,w,h)$= bounding box coordinates (x, y) representing the center of the box, and (w, h) representing its width and height. c_i= class label (representing the type of debris), s_i= confidence score indicating the likelihood of detection, n= number of detected objects.

The training objective for the yolov10n model is the minimization of a composite loss function that balances multiple tasks:

$l=l_{cls}+\lambda_1 l_{bbox}+\lambda_2 l_{obj}$.

where: l_{cls}= classification loss (cross-entropy), l_{bbox}= bounding box regression loss (ciou), l_{obj}= abjectness loss (indicating the presence of an object in the bounding box), λ_1,λ_2= balancing coefficients used to weight the importance of each loss component.

C. Preprocessing:

preprocessing is crucial to enhance underwater image quality before feeding it into the detection network. underwater images often suffer from low contrast, color distortion, and noise due to factors such as light scattering, turbidity, and depth variations. the preprocessing steps aim to correct these distortions and make the image suitable for accurate object detection.

1.Adaptive histogram equalization (clahe):

underwater images typically suffer from low contrast due to poor lighting conditions. adaptive histogram equalization (ahe) is applied to enhance image contrast locally, improving the visibility of edges and boundaries:

$i^{\prime}=clahe(i)$

where clahe (contrast limited adaptive histogram equalization) enhances local contrast by redistributing pixel intensity values within a defined local window, which helps to enhance object boundaries and texture details.

2. Normalization:

normalization is essential to standardize the input data, improving convergence during training and making the network more robust: $i_{norm}=(i^{\prime}-\mu)/\sigma$

where: μ= mean pixel intensity of the image, σ= standard deviation of pixel intensities

this process ensures that the pixel values are centered and have unit variance, which helps the network to generalize better.

3. Gaussian noise filtering:

To reduce noise introduced by the underwater environment, gaussian filtering is applied. the noisy image is denoised using a gaussian kernel $g(\sigma)$as follows:

$i_{denoise}=i_{norm}*g(\sigma)$. where $\sigma$is the standard deviation of the gaussian filter. this reduces high-frequency noise and enhances object details.

D. Segmentation using super-pixel fuzzy c-means (fcm):

Segmentation is used to isolate potential debris regions from the background. super-pixel-based fast fuzzy c-means (fcm) clustering is employed for this

task. it groups pixels based on similar color, texture, and spatial information, resulting in well-defined regions for object detection.
the clustering objective is defined as:
$$j=\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^m \|x_j-v_i\|^2$$
where: u_{ij}= membership degree (indicating how strongly a pixel belongs to a cluster), v_i= center of cluster i
m= fuzziness coefficient (controls the degree of membership overlap), x_j= pixel value of the j-th pixel
this segmentation step isolates the debris regions, making subsequent detection more reliable, especially in cluttered or partially occluded scenes.

E. Feature extraction:

feature extraction is crucial to identify discriminative patterns that differentiate marine debris from natural underwater elements. the feature extraction module uses a hybrid architecture of convolutional layers and transformer-based attention mechanisms.

1. **convolutional layers**

the convolutional layers identify low-level features such as edges, contours, and textures in the image. the output of the convolutional layers is passed through activation functions such as relu (rectified linear unit) or silu (sigmoid-weighted linear unit).
$$f_l=\sigma(w_l*f_{l-1}+b_l)$$
where: w_l= convolution weights for layer l, f_l= feature map output of layer l, σ= activation function

2. **transformer-based attention**

to enhance the model's ability to focus on important features, attention mechanisms are integrated into the architecture. the attention mechanism assigns different weights to the pixels based on their relevance, improving contextual awareness:
$$attention(q,k,v)=softmax\left(\frac{qk^t}{\sqrt{d_k}}\right)v$$
where: q= query matrix, k= key matrix, v= value matrix, d_k= dimensionality of keys
this helps the model prioritize important spatial relationships for better debris detection.

F. Yolov10n detection:

the core of the proposed framework is the yolov10n model, a lightweight, anchor-free object detection system that balances speed and accuracy.
anchor-free detection: yolov10n removes the need for predefined anchors, reducing computational complexity and simplifying model training.
ciou loss function: bounding box regression is optimized using the complete iou (ciou) loss function:
$$l_{ciou}=1-iou+\frac{\rho^2(b,b^{gt})}{c^2}+\alpha v$$
where:iou= intersection over union, ρ= center distance between predicted and ground truth box, c= diagonal of the smallest enclosing box, v= aspect ratio consistency term

feature pyramid network (fpn): multi-scale detection is enabled through an fpn, which aggregates features from different layers to detect objects of varying sizes:
$$f_{fusion}=f_l+\text{"upsample"}(f_{l+1}).$$

3.2 System Architecture

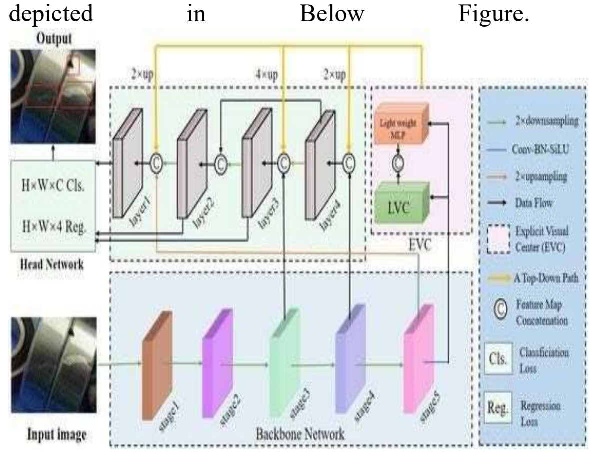
The system architecture of the proposed underwater debris detection framework is designed to support efficient real-time processing on resource-constrained devices. The architecture integrates image acquisition, feature extraction, and object detection into a unified pipeline, enabling accurate and low-latency marine environmental monitoring. The overall structure follows a modular and hierarchical design to ensure scalability, flexibility, and ease of deployment.
The architectural framework is illustrated in Fig. 1, which presents the complete processing flow from input image acquisition to final debris detection and classification. As shown in the diagram, the system consists of a backbone network, feature fusion layers, and a lightweight detection head optimized for underwater applications.

3.2.1 Input and Preprocessing Layer

The architecture begins with the input layer, which receives underwater images captured using submerged cameras, drones, or IoT-enabled sensors. These images are represented in the form of pixel matrices with defined height, width, and channel dimensions.
Before being forwarded to the detection network, the images undergo preprocessing operations such as resizing, normalization, and contrast enhancement. These operations improve image quality by reducing noise, correcting illumination variations, and enhancing object boundaries. The preprocessed images serve as standardized inputs for subsequent feature extraction stages

3.2.2 Backbone Network

The backbone network forms the core feature extraction component of the architecture. As depicted in Below Figure.



International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 14, Issue 1, 2026

Fig 3.1: Backbone Network

The backbone consists of multiple convolutional layers arranged hierarchically. These layers progressively extract low-level and high-level features from the input images.
Early convolutional layers capture basic visual patterns such as edges, textures, and contours. Deeper layers extract complex semantic information representing shapes and structural properties of marine debris. The backbone network is optimized for lightweight operation, ensuring minimal computational overhead and memory consumption.

3.2.3 Feature Pyramid and Fusion Layer

To enhance multi-scale object detection, the architecture incorporates a Feature Pyramid Network (FPN) and feature fusion mechanisms. This component aggregates feature maps obtained from different backbone layers and combines them to form a unified multiresolution representation.
Feature maps are up-sampled and concatenated at different stages to preserve spatial information. This process enables effective detection of debris objects of varying sizes, including small plastic fragments and large fishing nets.
The fusion layer improves detection robustness by integrating both local and global contextual information.

3.2.4 Lightweight Detection Head

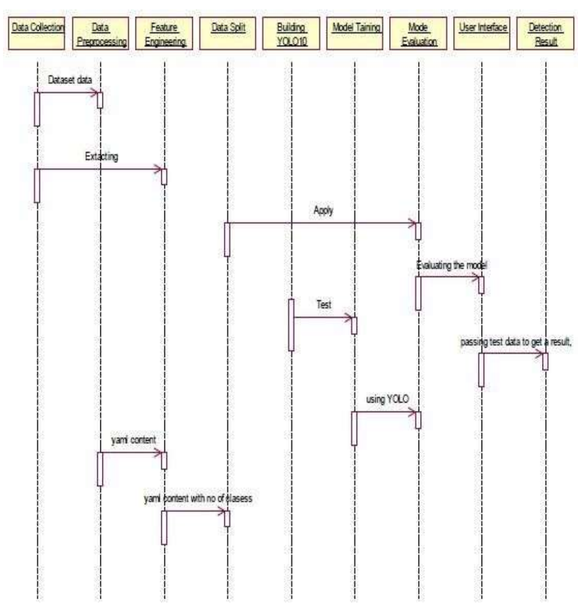


Fig 3.2: System Workflow

The detection head is responsible for performing object localization and classification. In the proposed architecture, a decoupled detection head is used, which separates classification and bounding box regression tasks.
The detection head consists of parallel branches that predict class probabilities and object coordinates. This separation improves learning stability and enhances detection accuracy.
The anchor-free detection mechanism adopted in YOLOv10n reduces parameter complexity and eliminates the need for predefined anchor boxes. This contributes to faster inference and simpler model optimization.

3.2.5 Output Layer and Prediction Module

The output layer generates final detection results in the form of bounding boxes, class labels, and confidence scores. These predictions indicate the presence and location of marine debris in the input images.
Detected objects are categorized into different classes such as plastic waste, metal fragments, fishing nets, and organic materials. The output data are formatted for visualization and storage, enabling further analysis and decision-making.
The prediction module supports both image-based and real-time video-based monitoring, making the system suitable for continuous marine surveillance.

3.2.6 Deployment Architecture

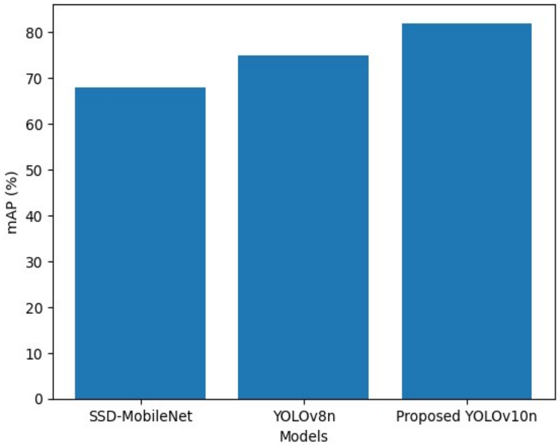
The proposed system is designed for deployment on embedded and edge platforms, including Raspberry Pi, NVIDIA Jetson, and underwater robotic systems. The lightweight nature of the architecture enables execution with limited processing power and energy resources.
The system components are distributed across sensing devices, processing units, and user interfaces. Data collected from sensors is transmitted to edge devices for local inference, reducing dependence on cloud infrastructure and minimizing communication latency.
This decentralized deployment strategy enhances system reliability and supports long term autonomous operation.
The operational workflow of the system is summarized as follows:

1. Underwater images are captured by sensing devices.
2. Preprocessing operations enhance image quality.
3. Feature extraction is performed by the backbone network.
4. Multi-scale features are fused using the FPN.
5. The detection head predicts object locations and classes.
6. Results are visualized and stored for analysis.

This streamlined workflow ensures efficient and accurate marine debris detection in Realtime.

3.2.7 Summary of System Architecture

The proposed system architecture integrates lightweight feature extraction, multi-scale feature fusion, and optimized object detection into a unified framework. By leveraging YOLOv10n, the architecture achieves high detection accuracy with reduced computational cost.The modular design supports flexible deployment and scalability, making the system suitable for large-scale marine environmental monitoring. The architecture effectively addresses the challenges of underwater



imaging and resource limitations, providing a practical solution for automated debris detection.

V. RESULTS ANALYSIS

Evaluation Metrics:

To evaluate the performance of the proposed system, the following standard metrics are utilized:

Precision Definition: Precision measures the proportion of correctly predicted positive instances out of all instances predicted as positive.

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP + FP}}$$

Where: $\boldsymbol{TP}$= True Positives (correctly predicted positive objects), $\boldsymbol{FP}$= False Positives (incorrectly predicted objects)

Purpose: High precision indicates that the model correctly identifies most of the predicted positive objects, minimizing false alarms.

Recall Definition: Recall measures the proportion of actual positive instances correctly detected by the model.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP + FN}}$$

Where: $\boldsymbol{FN}$= False Negatives (missed objects that were actually positive)

Purpose: High recall means the model detects as many of the true positives as possible, even at the cost of some false positives.

Mean Average Precision (mAP) Definition: Mean Average Precision (mAP) provides the average precision across all classes in a multi-class classification problem, which helps assess the overall detection accuracy for each class.

$$\boldsymbol{mAP} = \frac{\boldsymbol{1}}{\boldsymbol{N}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{AP_i}$$

Where: $\boldsymbol{N}$= Number of classes, $\boldsymbol{AP_i}$= Average Precision for class $\boldsymbol{i}$

Purpose: mAP is a comprehensive metric that accounts for both precision and recall across multiple classes and is commonly used to evaluate object detection models.

Frames Per Second (FPS) Definition: FPS measures the inference speed of the model, which is crucial for ensuring real-time performance in applications like marine debris detection.

$$\boldsymbol{FPS} = \frac{\text{Frames}}{\text{Time}}$$

Fig 1: Mean Average Precision (mAP) Comparation

Where: Frames = Total number of frames processed, Time = Time taken to process the frames

Purpose: A higher FPS indicates better performance for real-time deployment, ensuring that the system can detect and process underwater debris without delay.

Expected Performance Improvements:

The YOLOv10n-based system offers the following improvements over existing models:

Higher mAP for Detecting Small Debris: The proposed model achieves significantly better mAP in detecting small debris, such as plastic fragments and fishing nets, compared to older models like YOLOv8n. The integration of Feature Pyramid Networks (FPN) allows for effective multi-scale detection, improving the accuracy of small object detection.

Improved FPS on Edge Platforms:

The inference speed (FPS) has been optimized for deployment on resource-constrained edge platforms, such as the NVIDIA Jetson Nano. The lightweight architecture of YOLOv10n ensures high FPS, enabling real-time detection for underwater applications.

Reduced Model Size:

The model size has been significantly reduced compared to YOLOv8n, making it suitable for deployment in resource-constrained environments such as IoT-enabled devices. This reduction in size also leads to lower memory consumption and faster processing, improving the overall efficiency of the system.

Computational Efficiency:

To enhance the computational efficiency and make the model suitable for embedded platforms, we apply model quantization:

$$\boldsymbol{W_q} = \text{round}\left(\frac{\boldsymbol{W}}{\boldsymbol{s}}\right)$$

Where: $\boldsymbol{W}$= Weights of the model, $\boldsymbol{s}$= Scaling factor

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 14, Issue 1, 2026

Impact of Quantization:

Memory Usage: Reducing the precision of the weights leads to smaller model size, which reduces memory usage in embedded devices.

Power Consumption: The reduced memory and processing requirements lower the overall power consumption, making the system more energy-efficient for long-term IoT-based monitoring deployments.

Purpose: The application of model quantization ensures that the system can be deployed in long-term autonomous operations, such as marine environmental monitoring, where both power and memory efficiency are crucial.

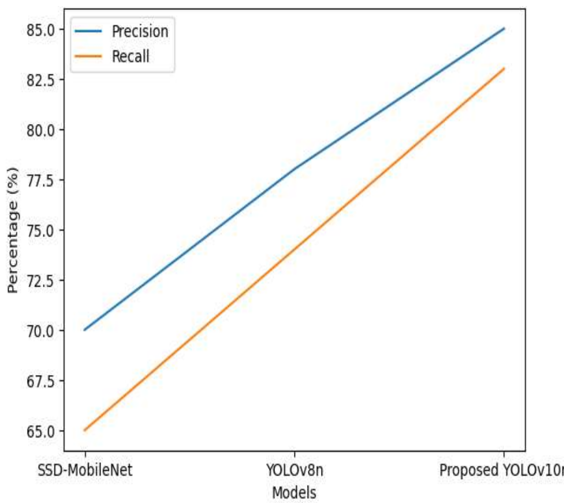


Fig 2: Precision and Recall Comparison

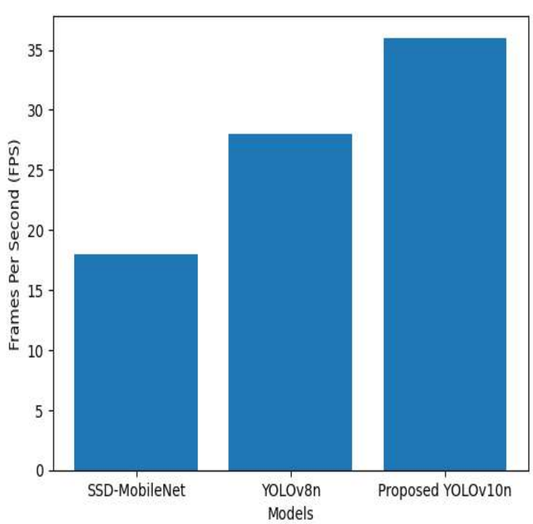


Fig 3: Interface Size Comparison

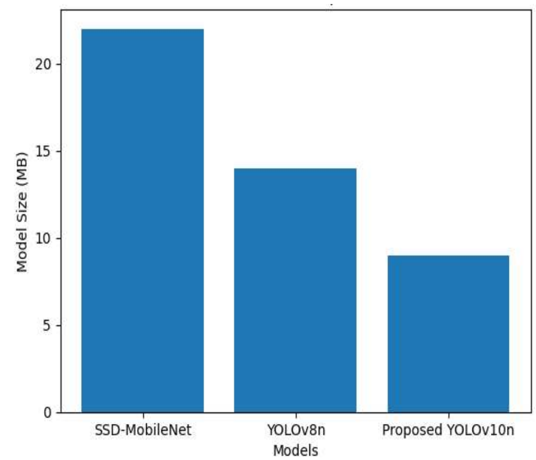


Fig 4: Model Size Comparison

V. CONCLUSION

The proposed underwater debris detection system leverages the YOLOv10n model, a lightweight yet high-performance object detection network, offering an innovative and efficient solution to tackle the issue of marine pollution. This system addresses the critical need for real-time and accurate detection of various types of underwater debris such as plastics, fishing nets, bottles, and organic waste that significantly harm marine ecosystems.

Key Contributions and Achievements:

Real-time Detection with YOLOv10n: The YOLOv10n model has been specifically optimized to operate efficiently on resource-constrained platforms, making it suitable for deployment on underwater drones, IoT-enabled cameras, and embedded edge devices. The model's architecture significantly reduces computational complexity, allowing for high-speed inference with minimal power consumption while maintaining high detection accuracy.

Improved Efficiency:

Compared to traditional models like YOLOv8n or SSD, our approach achieves a higher mean Average Precision (mAP) for detecting small and irregularly shaped debris in challenging underwater conditions. Additionally, the model size has been significantly reduced, which is crucial for deployment in environments with limited computational resources. This optimization ensures that the system is scalable, energy-efficient, and suitable for long-term autonomous operation.

Edge Deployment:

By adopting quantization techniques and leveraging lightweight processing architectures, the system is designed to run efficiently on edge devices such as the NVIDIA Jetson Nano, making it suitable for real-time monitoring in marine environments. This ensures that real-time feedback can be provided for pollution mapping, cleanup operations, and environmental research.

Scalable and Sustainable Solution:

The modular nature of the proposed framework allows it to be scalable, making it suitable for large-scale deployment across various marine ecosystems. With its energy-efficient design and low operational costs, the system presents a sustainable solution for marine conservation, enabling continuous monitoring of marine debris without reliance on expensive, resource-heavy systems.

Potential for Environmental Conservation: This research showcases the immense potential of deep learning technologies in advancing environmental conservation efforts. By automating the detection and classification of marine debris, the proposed system can assist in pollution reduction, data collection for ecological studies, and real-time intervention strategies. The ability to process large amounts of environmental data on low-cost platforms offers significant advantages over traditional, manually operated systems, contributing to the protection of marine ecosystems.

Future Work and Implications:
Future work will focus on improving model robustness under diverse underwater conditions, including varying water depths, lighting conditions, and turbidity. Additionally, the integration of sensor fusion (combining visual data with sonar, thermal, and depth sensors) will enhance debris detection accuracy and object tracking capabilities. The real-time capability of the system can also be extended to autonomous underwater vehicles (AUVs), allowing for automated debris collection and pollution monitoring in hard-to-reach areas

References

1. F. T. Bahadur, S. R. Shah, and R. R. Nidamanuri, ‘‘Air pollution monitoring, and modelling: An overview,’’ Environ. Forensics, vol. 25, no. 5, pp. 309–336, Sep. 2024.
2. S. Wei, T. Xian-Chun, X. U. Yan-Dong, Z. Huan-Jun, L. Yuan-Jin, and M. A. JianXin, ‘‘The distribution, composition and change characteristics of marine debris in the coastal area of Shandong Province,’’ Sci. Technol. Eng., vol. 16, no. 18, pp. 89–94, 2016.
3. G. Teng, X. Shan, X. Jin, and T. Yang, ‘‘Marine litter on the seafloors of the Bohai Sea, Yellow Sea and northern East China Sea,’’ Mar. Pollut. Bull., vol. 169, Aug. 2021, Art. no. 112516.
4. S. Krishnakumar, D. S. H. Singh, B. Nallusamy, and M. Aslam, ‘‘Marine pollution and ecological degradation—Issues and challenges,’’ Environ. Sci. Pollut. Res., vol. 31, no. 29, pp. 41303–41305, 2024.
5. G. Masetti and B. R. Calder, ‘‘A Bayesian marine debris detector using existing hydrographic data products,’’ in Proc. OCEANS-Genova, May 2015, pp. 1–10.
6. J. Cheng, Y. Yang, X. Tang, N. Xiong, Y. Zhang, and F. Lei, ‘‘Generative adversarial networks: A literature review,’’ KSII Trans. Internet Inf. Syst. (TIIS), vol. 14, no. 12, pp. 4625–4647, 2020.
7. S. Kong, M. Tian, C. Qiu, Z. Wu, and J. Yu, ‘‘IWSCR: An intelligent water surface cleaner robot for collecting floating garbage,’’ IEEE Trans. Syst., Man, Cybern., Syst., vol. 51, no. 10, pp. 6358–6368, Oct. 2021.
8. Y. Zeng, C. J. Sreenan, N. Xiong, L. T. Yang, and J. H. Park, ‘‘Connectivity and coverage maintenance in wireless sensor networks,’’ J. Supercomput., vol. 52, no. 1,pp. 23–46, Feb. 2010.
9. R. Wan, N. Xiong, Q. Hu, H. Wang, and J. Shang, ‘‘Similarity-aware data aggregation using fuzzy c-means approach for wireless sensor networks,’’ EURASIP J. Wireless Commun. Netw., vol. 2019, no. 1, pp. 1–11, Dec. 2019.
10. W. Zhang, M. Du, X. Guo, and N. N. Xiong, ‘‘SRFL: A swarm reputationbased autonomic federated learning framework for AIoT,’’ IEEE Internet Things J., early access, Dec. 13, 2024, doi: 10.1109/JIOT.2024.3509264.
11. J. Guo, A. Liu, K. Ota, M. Dong, X. Deng, and N. N. Xiong, ‘‘ITCN: An intelligent trust collaboration network system in IoT,’’ IEEE Trans. Netw. Sci. Eng., vol. 9, no. 1, pp. 203–218, Jan. 2022.
12. W. Tang, H. Gao, and S. Liu, ‘‘Design and implementation of small waters intelligent garbage cleaning robot system based on raspberry pi,’’ Sci. Technol. Eng., vol. 19, no. 34, pp. 239–247, 2019.
13. N. Dilshad, A. Ullah, J. Kim, and J. Seo, ‘‘LocateUAV: Unmanned aerial vehicle location estimation via contextual analysis in an IoT environment,’’ IEEE Internet Things J., vol. 10, no. 5, pp. 4021–4033, Mar. 2023.
14. C. Sun, C. Zhang, and N. Xiong, ‘‘Infrared and visible image fusion techniques based on deep learning: A review,’’ Electronics, vol. 9, no. 12, p. 2162, Dec. 2020.
15. H. Yu, Y. Tian, Q. Ye, and Y. Liu, ‘‘Spatial transform decoupling for oriented bject detection,’’ in Proc. AAAI Conf. Artif. Intell., Mar. 2024, vol. 38, no. 7, pp. 6782– 6790.
16. M. Valdenegro-Toro, ‘‘Submerged marine debris detection with autonomous underwater vehicles,’’ in Proc. Int. Conf. Robot. Autom. Humanitarian Appl. (RAHA), Dec. 2016, pp. 1–7.
17. K. He, G. Gkioxari, P. Dollár, and R. Girshick, ‘‘Mask R-CNN,’’ in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2961–2969.

18. R. Girshick, ''Fast R-CNN,'' 2015, arXiv:1504.08083.
19. S. Ren, K. He, R. Girshick, and J. Sun, ''Faster R-CNN: Towards real time object detection with region proposal networks,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
20. M. Hussain, ''YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection,'' Machines, vol. 11, no. 7, p. 677, Jun. 2023.
21. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, ''SSD: Single shot MultiBox detector,'' in Proc. 14th Eur. Conf. Comput. Vis. (ECCV), Amsterdam, The Netherlands. Cham, Switzerland: Springer, Jan. 2016, pp. 21–37.
22. H. Zhao, Y. Gao, and W. Deng, ''Defect detection using shuffle Net-CASSD lightweight network for turbine blades in IoT,'' IEEE Internet Things J., vol. 11, no. 20, pp. 32804–32812, Oct. 2024.
23. L. Duan, B. Tan, and X. Yu, ''Defect detection for wine bottle caps based on improved YOLOv3,'' Electr. Meas. Technol., vol. 45, no. 15, pp. 130–137, 2024.
24. M. Fulton, J. Hong, M. J. Islam, and J. Sattar, ''Robotic detection of marine litter using deep visual detection models,'' in Proc. Int. Conf. Robot. Autom. (ICRA), May 2019, pp. 5752–5758.
25. K. Kylili, C. Hadjistassou, and A. Artusi, ''An intelligent way for discerning plastics at the shorelines and the seas,'' Environ. Sci. Pollut. Res., vol. 27, no. 34, pp. 42631– 42643, Dec. 2020.
26. Y. An, Y. Feng, N. Yuan, Z. Ji, and I. Ganchev, ''DIRBW-Net: An improved inverted residual network model for underwater image enhancement,'' IEEE Access, vol. 12, pp. 75474–75482, 2024.
27. B. Xue, B. Huang, W. Wei, G. Chen, H. Li, N. Zhao, and H. Zhang, ''An efficient deep-sea debris detection method using deep neural networks,'' IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 14, pp. 12348–12360, 2021.
28. D. V. Politikos, E. Fakiris, A. Davvetas, I. A. Klampanos, and G. Papatheodorou, ''Automatic detection of seafloor marine litter using towed camera images and deep learning,'' Mar. Pollut. Bull., vol. 164, Mar. 2021, Art. no. 111974.
29. M. Córdova, A. Pinto, C. C. Hellevik, S. A.-A. Alaliyat, I. A. Hameed, H. Pedrini, and R. D. S. Torres, ''Litter detection with deep learning: A comparative study,'' Sensors, vol. 22, no. 2, p. 548, Jan. 2022.
30. F. Peng, Z. Miao, F. Li, and Z. Li, ''S-FPN: A shortcut feature pyramid network for sea cucumber detection in underwater images,'' Expert Syst. Appl., vol. 182, Nov. 2021, Art. no. 115306.
31. K. Yang, J. Yang, B. Chen, M. Lv, C. Li, B. Yang, and W. Zheng, ''Methods of defect detection in transmission line based on depthwise separable convolution and SCD,'' Smart Power, vol. 48, no. 10, pp. 64–69, 2020.
32. D. Ma, J. Wei, Y. Li, F. Zhao, X. Chen, Y. Hu, S. Yu, T. He, R. Jin, Z. Li, and M.Liu, ''MLDet: Towards efficient and accurate deep learning method for marine litter detection,'' Ocean Coastal Manage., vol. 243, Sep. 2023, Art. no. 106765.
33. G. Wu, Y. Ge, and Q. Yang, ''UTD-YOLO: Underwater trash detection model based on improved YOLOv5,'' J. Electron. Imag., vol. 32, no. 6, Dec. 2023, Art. no. 063034.
34. B. Chen, M. Wei, J. Liu, H. Li, C. Dai, J. Liu, and Z. Ji, ''EFS-YOLO: A lightweight network based on steel strip surface defect detection,'' Meas. Sci. Technol., vol. 35, no. 11, Aug. 2024, Art. no. 116003, doi: 10.1088/1361- 6501/ad66fe.
35. X. Yang, L. Liu, X. Song, J. Feng, Q. Pei, X. Yuan, and J. Li, ''An efficient lightweight satellite image classification model with improved MobileNetV3,'' in Proc.IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), May 2024, pp. 1–6.
36. X. Zhao, Y. Wang, Y. Kang, S. Hu, C. Guan, W. Guo, Z. Lian, and S. Liang, ''Defect recognition in fiber-optic coil winding based on improved MobileNetV1,'' IEEE Sensors J., vol. 24, no. 17, pp. 27300–27308, Sep. 2024.
37. B. A. Kumar and N. K. Misra, ''Masked face age and gender identification using CAFFE-modified MobileNetV2 on photo and real-time video images by transfer learning and deep learning techniques,'' Expert Syst. Appl., vol. 246, Jul. 2024, Art. no. 123179.
38. G. Lu, W. Zhang, and Z. Wang, ''Optimizing depth wise separable convolution operations on GPUs,'' IEEE Trans. Parallel Distrib. Syst., vol. 33, no. 1, pp. 70–87, Jan. 2022.
39. H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan, and Q. Ren, ''Slim-neck by GSConv: A lightweight-design for real-time detector architectures,'' 2022, arXiv:2206.02424.
40. C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, ''CSPNet: A new backbone that can enhance learning capability of CNN,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2020, pp. 1571–1580.
41. Z. Gevorgyan, ''SIoU loss: More powerful learning for bounding box regression,'' 2022, arXiv:2205.12740.
42. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, ''Grad-CAM: Visual explanations from deep networks via gradient-based localization,'' in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 618– 626.