Architectural Optimization and Comparative Benchmarking of Attention-Augmented YOLO Models for Deep-Sea Litter Classification

Tharun A, Sai Govardhan M Jayachetan, Rimjhim Singh
Department of Computer Science and Engineering
Amrita School of Computing,Bengaluru
Amrita Vishwa Vidyapeetham,India.

bl.en.u4cse22173@bl.students.amrita.edu, bl.en.u4cse22170@bl.students.amrita.edu ,
bl.en.u4cse22144@bl.students.amrita.edu, ps_rimjhim@blr.amrita.edu

***Abstract*—Marine waste, especially of anthropogenic origin like plastic, metal, glass, and fabric, is an increasingly serious threat to aquatic life. Effective, real-time identification and labeling of such waste is essential in aiding cleanup efforts and monitoring of the environment. This project investigates an end-to-end computer vision workflow involving the application of several YOLO (You Only Look Once) object detection models to detect marine waste in images and video streams. A total of 17 YOLO-based models were comprehensively analyzed, including baseline versions and attention-supplied models with hybrid backbones (e.g., ResNet50, Vision Transformer). The models were tested on a bespoke Roboflow dataset consisting of four waste types and were assessed on precision, recall, and mAP scores. The last system provides a scalable and precise solution for real-time environmental waste monitoring in marine applications.**

***Index Terms*—Marine Waste Detection,Object Detection,YOLOv5–YOLOv12,Attention Mechanisms,Vision Transformer (ViT),ResNet Backbone,Deep Learning,Environmental Monitoring,Roboflow Dataset**

I. Introduction

Marine pollution has become a serious environmental concern with enormous amounts of waste piling up in the oceans, rivers, and coastal areas. Of the numerous pollutants, anthropogenic waste like plastic, metal, glass, and cloth ranks high among the dangers threatening marine biodiversity as well as human health. Many of the conventional waste detection and cleanup activities depend on human observation or sensors at the surface, which are ineffective and error-prone. To meet this challenge, there is a pressing requirement for intelligent, automated systems to identify and categorize marine waste in real-time.

Advances in computer vision and deep learning have made object detection in complex environments possible with automated means. Specifically, YOLO models have become well-known for their compromise between speed and precision in real-time processing. YOLO object detectors perform a single forward pass on images, making them appealing for dynamic environments like aerial or underwater surveillance.

This project centers on the design and assessment of a set of YOLO-based models for detecting marine waste in four prominent categories—cloth, glass, metal, and plastic. The models are trained on a specially designed dataset amassed and annotated through Roboflow, customized for real-world waste detection. In addition to baseline detection through typical YOLO models (YOLOv5 through YOLOv12), this project explores architectural augmentations with attention layers, and ResNet backbones to better detect small, occluded, or visually intricate waste items.

With extensive experimentation across 17 YOLO variants, such as attention-augmented, the research illustrates the feasibility of smart waste monitoring systems. Results aim to inform scalable, automated marine cleanup, underwater robotics, and enforcement of environmental policies.

The primary contributions of this work include:

- Trained and tested 17 models of YOLO (YOLOv5 to YOLOv12) variants with and without attention and hybrid.
- Implemented attention layers at multiple depths and skip connections to improve feature concentration.
- Designed a YOLO-based real-time system to identify cloth, glass, metal, and plastic in marine settings, including underwater and above-water scenes.

II. Literature review

Saeed et al. [1] proposed a deep learning model based on the YOLOv5 framework for detecting and classifying marine litter, achieving high accuracy in object detection with a robust segmentation approach. The method demonstrated significant potential for real-time applications and scalability in marine waste monitoring. However, the model's performance was constrained by the limited diversity of the dataset used, affecting its generalizability to other environments.

Liu et al. [2] applied a semantic segmentation model based on a U-Net architecture for the identification and classification of floating waste in ocean environments. It was capable of producing fine-grained detection with high precision for segmenting marine litter. A significant limitation, however, was the computational overhead that prevented it from being deployed on resource-constrained devices.

Yu et al. [3] presented a hybrid model by incorporating traditional machine learning with CNN-based feature extraction to classify marine litter from aerial images. This integration enhanced the interpretability of the machine learning method while reducing the computational costs. The method was weak in the detection of small objects in large-scale images.

Islam et al. [4] applied a Faster R-CNN framework for underwater marine waste detection. The proposed approach demonstrated robust performance in complex and occluded scenarios. Its advantage was the adaptability to diverse underwater conditions due to advanced feature extraction techniques. However, the model's performance degraded in low-visibility conditions, highlighting the need for additional preprocessing steps.

Fulton et al. [5] proposed a deep visual detection system using RetinaNet for detecting ocean waste on robotic platforms. The method excelled in real-world deployments and was perfectly suitable to be applied autonomously, but it relied heavily on copious amounts of annotated data for training, which became the bottleneck for further scaling.

Kyriaki et al. [6] made use of a deep learning pipeline with the ResNet50 architecture to detect floating plastic debris in oceanic regions. The method presented high accuracy in distinguishing plastics from organic matter, thus proving to be effective in targeted waste management. Its ability to carry out submerged waste detection was lacking, however, making it partially insensitive to holistic waste detection.

A DCNN by Rajasekaran et al. [7] is presented that was used for the classification of marine debris while optimizing the model for real-time energy efficiency in applications. Such optimization allowed its deployment on low-power edge devices. However, the model suffered under light variability.

Singh et al. [8] designed a hybrid visiontransformer-based model for the high-precision segmentation of marine wastes in complex oceanic environments. The method was highly efficient and resulted in improved performance compared to CNNs in precision and recall but had a large computational requirement and was not recommended for real-time low-power applications.

Sharma et al. [9] proposed a multi-scale deep learning model for the detection and classification of marine litter in drone-captured images with outstanding performances for small and overlapping objects detections. The model was proven to be really scalable with vast applications; however, it is dependent on high-resolution imagery, which leads to limited usability in resource-constrained environments.

Wang et al. [10] proposed an ensemble learning method by combining CNN and transformer models to classify recyclable and non-recyclable marine waste. The ensemble method improved the detection accuracy and robustness under different ocean conditions. The method, however, demanded large computational resources and longer training times.

Zhang et al. [11] carried out classification of marine litter using a hybrid CNN-LSTM architecture to analyze patterns in the distribution of oceanic waste sequentially. There is a huge drawback, which is higher complexity. It made the system difficult to implement in real-time.

Patel et al. [12] proposed an attention-based segmentation network to improve the detection of marine debris in high-clutter environments. The attention mechanism helped enhance model performance by focusing on the relevant regions of the images. However, noisy data sensitivity limits the robustness of the network.

Ali et al. [13] used a generative adversarial network (GAN) to improve the performance of downstream detection models of marine waste. The augmented dataset helped the model generalize better under various scenarios; however, in the GAN-based approach, it was often sensitive to some tuning parameters because otherwise, it would give unrealistic or highly irrelevant samples.

Kumar et al. [14] proposed a lightweight MobileNet-based model for real-time marine litter detection that can be deployed on low-power devices. The model had high efficiency with low computational costs. However, it had slightly lower accuracy than heavier architectures in complex scenarios.

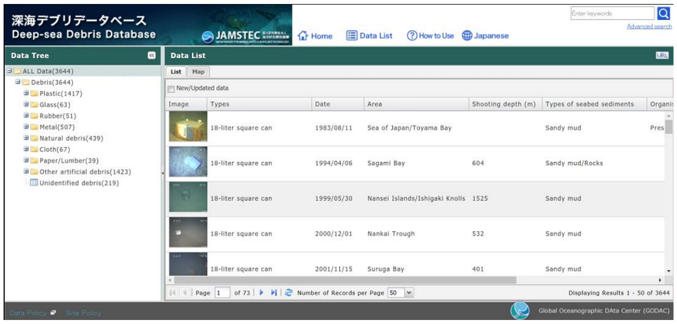
Joshi et al. [15] used an ensemble model, combining ResNet and DenseNet deep learning models for the classification of marine waste from underwater images. Since the ensemble leverages complementary features, detection accuracy improved. On the other hand, training was more time-consuming because of the complexity of the ensemble.

Roy et al. [16] proposed a transfer learning-based method using a pre-trained VGG16 for detecting and classifying marine debris in low-resource settings. This approach eliminated the need for a large amount of data and produced comparable results. However, its usability was restricted to very similar domains than the one from which the pre-trained model was obtained.

Chen et al. [17] proposed a region-based segmentation model for the detection of marine debris using aerial imagery, with a focus on multi-class classification of waste types. The approach provided granular insights into the composition of marine litter. A drawback was the dependence on annotated aerial images, which are costly and time-consuming to acquire.

Shubham et al. [18] proposed a YOLOv7-tiny-based framework for detecting underwater litter, enhancing the model's performance for real-time underwater applications. The lightweight architecture allowed efficient deployment on underwater robots with minimal computational resources. However, the model struggled with detecting waste in extremely low-light or murky water conditions, reducing its robustness.

Zhao et al. [19] proposed a modified CNN for underwater litter detection from multi-modal data of underwater robots. The approach improved the accuracy of detection by fusing visual and depth data and was thus very effective in complex underwater environments. However, it had the disadvantage of relying on high-quality depth data, which is difficult to obtain in most marine scenarios.



Data Tree

- ALL Data(3644)
  - Debris(3644)
    - Plastic(1417)
    - Glass(63)
    - Rubber(51)
    - Metal(507)
    - Natural debris(439)
    - Cloth(67)
    - Paper/Lumber(39)
    - Other artificial debris(1423)
    - Unidentified debris(219)

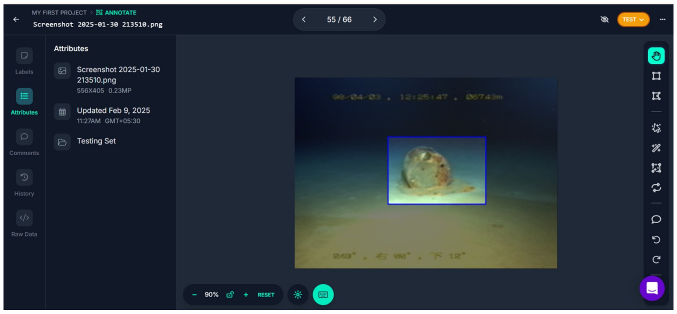
Data List

☐ New/Updated data

Image	Types	Date	Area	Shooting depth (m)	Types of seabed sediments	Organi
	18-liter square can	1983/08/11	Sea of Japan/Toyama Bay		Sandy mud	Pres
	18-liter square can	1994/04/06	Sagami Bay	604	Sandy mud/Rocks	
	18-liter square can	1999/05/30	Nansei Islands/Ishigaki Knolls	1525	Sandy mud	
	18-liter square can	2000/12/01	Nankai Trough	532	Sandy mud	
	18-liter square can	2001/11/15	Suruga Bay	401	Sandy mud	

Page 1 of 73 | Number of Records per Page 50 | Displaying Results 1 - 50 of 3644

Fig. 1. JAMSTEC Database Screenshot



Screenshot 2025-01-30 213510.png

Attributes

Screenshot 2025-01-30 213510.png
556X405 0.23MP

Updated Feb 9, 2025
11:27AM GMT+05:30

Testing Set

Fig. 2. Annotated Image from Roboflow

Shetty et al. [20] provided a machine learning pipeline for marine waste classification based on a random forest algorithm learned on spectral and texture features. This method was interpreted highly and had good accuracy in small datasets. Therefore, the usage of the pipeline was restricted mainly to small-scale studies on marine litter, rather than the field's real-world application

III. Dataset Description

Fig. 1 Data used for this research was specifically designed using actual marine debris videos retrieved from the JAMSTEC (Japan Agency for Marine-Earth Science and Technology) Deep-Sea Debris Database. The JAMSTEC videos include deep-sea recordings showing multiple underwater debris objects under adverse conditions like turbidity, occlusion, and cluttered backgrounds.

As shown in Fig. 2 To clean the dataset, frames were manually captured as screenshots of chosen video clips. All images were annotated manually on the Roboflow annotation platform, in which manually drawn bounding boxes surrounded four types of marine litter: plastic, metal, cloth, and glass.

The annotated images were saved in YOLO-compatible format, with every image associated with a respective .txt file filled with class IDs and bounding box coordinates according to YOLO conventions. The dataset was resized uniformly to 640×640 pixels to provide uniform input sizes for all model variants.

The data was organized in a well-balanced form to allow for strong training and testing. An image can have one or more classes of debris with different scales, viewpoints, and background complexity. The dataset was essential in benchmarking 17 various YOLO model configurations, which included attention-augmented and hybrid models.

TABLE I
Dataset Distribution

Split	Number of Images	Description
Training	2,500	Used to train the object detection models
Validation	260	Used to tune the models during training
Testing	260	Used to evaluate model performance

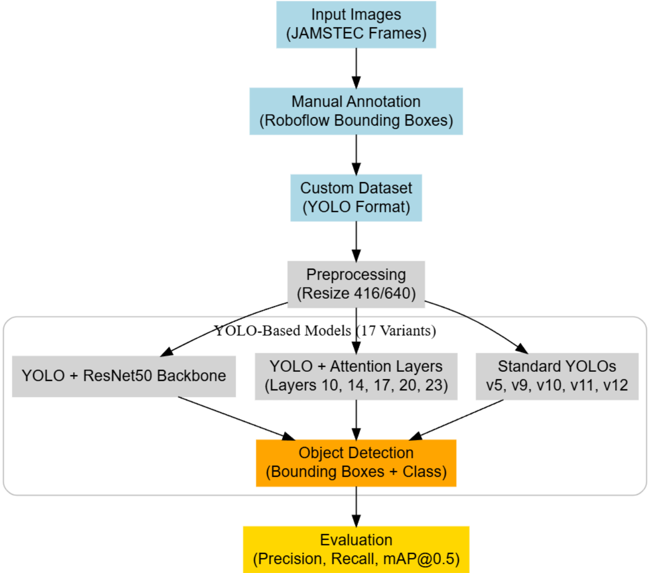


Fig. 3. System architecture

IV. Proposed Methodology:

The methodology followed in this work includes various important stages: creation and annotation of the dataset, training baseline models, integration of attention mechanisms, backbone improvement, and performance testing. 17 YOLO-based models, including base, attention-augmented, and personalized models, were implemented and compared.

The overall system architecture of the proposed sea trash detection system is shown in Fig. 3. The process starts with frame extraction from the underwater trash video data given by the JAMSTEC Deep-Sea Debris Database. These frames are annotated manually via the Roboflow platform, and bounding boxes are marked for four classes of objects: plastic, metal, cloth, and glass. The resultant custom dataset is exported in YOLO format and preprocessed by resizing each image to either 416×416 or 640×640 resolution. 17 YOLO-based models are trained on this dataset and are divided into three categories: (i) basic YOLO variants such as YOLOv5, v9, v10, v11, and v12, (ii) attention-augmented YOLO models with CBAM layers incorporated at different depths (Layers 10, 14, 17, 20, and 23), and (iii) hybrid YOLO models that use a ResNet50 backbone. Moreover, two dual-CBAM models were tested by adding attention concurrently at Layers 14 21 and

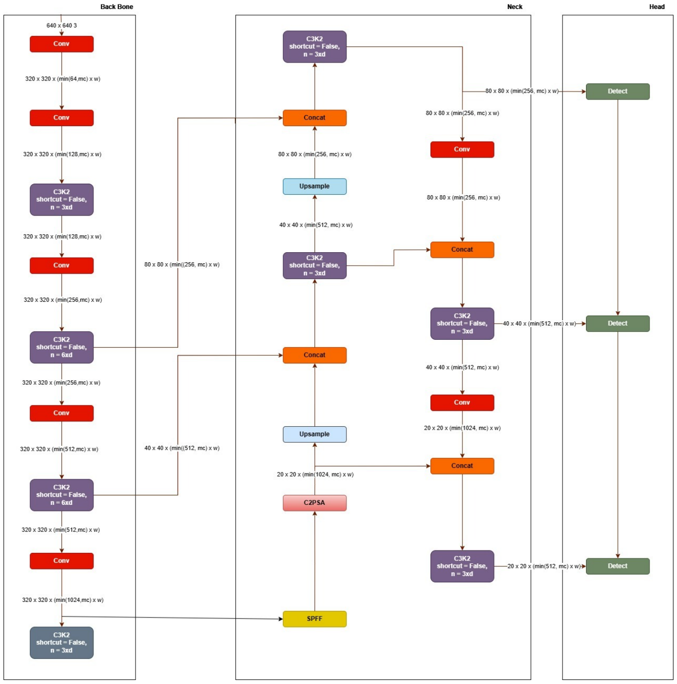

Back Bone
640 x 640 3
Conv
320 x 320 x (min(64,mc) x w)
Conv
320 x 320 x (min(128,mc) x w)
C3K2 shortcut = False, n = 3xd
320 x 320 x (min(128,mc) x w)
Conv
320 x 320 x (min(256,mc) x w)
C3K2 shortcut = False, n = 6xd
320 x 320 x (min(256,mc) x w)
Conv
320 x 320 x (min(512,mc) x w)
C3K2 shortcut = False, n = 6xd
320 x 320 x (min(512,mc) x w)
Conv
320 x 320 x (min(1024,mc) x w)
C3K2 shortcut = False, n = 3xd
80 x 80 x (min((256, mc) x w)
40 x 40 x (min((512, mc) x w)
Neck
C3K2 shortcut = False, n = 3xd
Concat
80 x 80 x (min(256, mc) x w)
Upsample
40 x 40 x (min(512, mc) x w)
C3K2 shortcut = False, n = 3xd
Concat
Upsample
20 x 20 x (min(1024, mc) x w)
C2PSA
SPFF
80 x 80 x (min(256, mc) x w)
Conv
80 x 80 x (min(256, mc) x w)
Concat
C3K2 shortcut = False, n = 3xd
40 x 40 x (min(512, mc) x w)
Conv
20 x 20 x (min(1024, mc) x w)
Concat
C3K2 shortcut = False, n = 3xd
Head
80 x 80 x (min(256, mc) x w)
Detect
40 x 40 x (min(512, mc) x w)
Detect
20 x 20 x (min(512, mc) x w)
Detect

Fig. 4. Yolo11s Architecture

Layers 17 24 in attempts to further enhance spatial attention and object localization. These models detect objects in real-time by predicting bounding box coordinates and class labels, and are tested using standard metrics like precision, recall, and mean Average Precision (mAP@0.5) to compare their detection capabilities under underwater conditions.

PROPOSED METHODOLOGY:

YOLOv11s

Architecture in Detail and Project Contribution YOLOv11s is a light, real-time object detection model, and it was the baseline for comparison of the performance of different advanced configurations in this work. Being a small member of the YOLOv11 family, YOLOv11s is especially designed to find a compromise between detection accuracy and computational efficiency and is exceptionally well-suited for deployment to resource-constrained settings like autonomous underwater vehicles (AUVs), remotely operated vehicles, and edge computing platforms in marine monitoring. Architectural Overview The YOLOv11s architecture is designed using a three-stage modular structure—Backneck, and Head—each handling increasingly higher-level representations from raw input images to final object detection outputs.

1) Backbone: The Backbone is the main feature extractor. It starts off with a number of convolutional layers that extract low-level visual features like edges and textures. To increase representation power with minimal increment in computational cost, YOLOv11s utilizes C3K2 modules (Cross Stage Partial connections with kernel size 2). These modules divide the feature map into several parts and deal with them in parallel, finally combine them to enhance gradient flow and learning diversity. The backbone is tasked with downsampling the input image and representing hierarchical semantic information essential in detecting small and large objects under occlusive underwater environments.

A. Neck

The Neck is engineered to fuse features across various depths of the backbone to manage multi-scale object detection. It employs:

- Concatenation layers to blend features from previous and deeper stages
- Upsampling layers to upscale feature maps back to higher resolutions
- More C3K2 blocks to polish fused features
- A Spatial Pyramid Pooling-Fast (SPPF) module, where parallel max-pooling operations are performed at various scales to capture local and global context
- A C2PSA (Channel and Spatial Attention) module to highlight region of interest and avoid masking irrelevant noise, enhancing attention to areas of marine waste
- The Neck assists the model to detect marine rubbish of different sizes and angles in various layers.

B. Head

The Head is the last component that does object localization and classification. It employs a series of convolutional layers to generate:

- Bounding box coordinates (center-x, center-y, width, height)
- Objectness score (confidence of an object being present)
- Class probability scores for every specified category (plastic, metal, cloth, glass)

YOLOv11s gives predictions at three distinct spatial scales:

- 80×80 (small objects),
- 40×40 (medium objects),
- 20×20 (large objects),

This multi-scale detection enables the model to detect marine trash irrespective of its depth or dimension within the frame.

Convolutional Block Attention Module (CBAM)

The Convolutional Block Attention Module (CBAM) is a light and efficient attention mechanism that is used to improve the representational capacity of convolutional neural networks by enabling the model to concentrate on salient features and ignore non-informative ones. In this project, CBAM was applied to chosen layers of the YOLOv11 structure to enhance its performance in detecting marine litter under harsh underwater environments.

CBAM processes in succession via two sub-modules:

- Channel Attention Module (CAM)
- Spatial Attention Module (SAM)

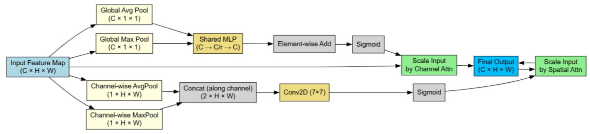


Fig. 5. Layer-wise architecture of the Custom CBAM

1) Channel Attention Module: The Channel Attention Module concentrates on the "what" of an image — i.e., which feature channels are of more significance for object detection. It achieves this by performing global average pooling and max pooling operations over every feature channel, and then using a shared multi-layer perceptron (MLP). This produces a set of learned channel weights, re-weighting the most significant features before forwarding them to the subsequent layer.

In the context of this project, CAM enables the model to highlight details like the texture of cloth, the shine of metal, or the transparency of glass — which are imperative in differentiating between marine waste and natural underwater objects.

2) Spatial Attention Module: After channel refinement, the Spatial Attention Module decides "where" in the image the model should look. It does pooling operations on the channel axis to pick up spatial information and finally applies a convolution operation to generate a spatial attention map. The map indicates which areas of the image are most important — where, for instance, a plastic bag or metal can is most likely to be seen.

The figure. 5 shows the structure of a custom Convolutional Block Attention Module (CBAM), which improves feature representation of convolutional neural networks by sequentially utilizing channel and spatial attention mechanisms. The upper path indicates the Channel Attention Module, where global average pooling and max pooling operate over spatial axes to produce descriptors, shared Multi-Layer Perceptron (MLP) projected through, combined through element-wise addition, and activated with a sigmoid function to obtain a channel attention map utilized to rescale the input feature map. The lower path is the Spatial Attention Module, where average and max pooling are carried out across the channel direction, concatenated and convolved using a 7×7 kernel to produce a spatial attention map via a sigmoid activation. The spatial attention map is ultimately applied to the channel-refined feature map and gives the final output that concentrates on the most informative spatial regions and channels.

Yolo11s + CBAM At best

The architecture depicted in Fig. 6 is the improved YOLOv11s model, integrated with CBAM attention modules at well-placed points in the network's neck. The improved architecture was implemented to enhance detection accuracy under the challenging circumstances of underwater marine trash, where occlusion, changing lighting, and cluttered backgrounds are common.

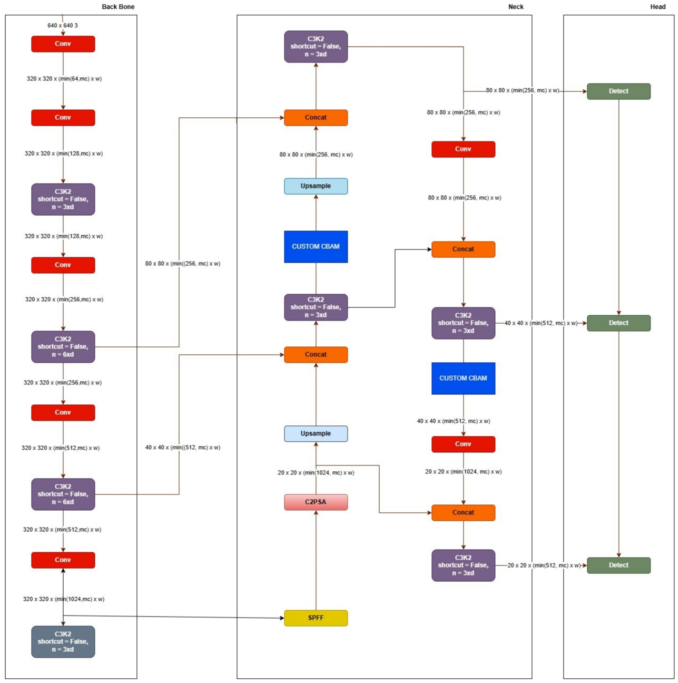

Back Bone
Neck
Head
640 x 640 3
Conv
320 x 320 x (min(64,mc) x w)
Conv
320 x 320 x (min(128,mc) x w)
C3K2
shortcut = False,
n = 3xd
320 x 320 x (min(128,mc) x w)
Conv
320 x 320 x (min(256,mc) x w)
C3K2
shortcut = False,
n = 6xd
320 x 320 x (min(256,mc) x w)
Conv
320 x 320 x (min(512,mc) x w)
C3K2
shortcut = False,
n = 6xd
320 x 320 x (min(512,mc) x w)
Conv
320 x 320 x (min(1024,mc) x w)
C3K2
shortcut = False,
n = 3xd
80 x 80 x (min((256, mc) x w)
40 x 40 x (min((512, mc) x w)
C3K2
shortcut = False,
n = 3xd
Concat
80 x 80 x (min(256, mc) x w)
Upsample
CUSTOM CBAM
C3K2
shortcut = False,
n = 3xd
Concat
Upsample
20 x 20 x (min(1024, mc) x w)
C2PSA
SPFF
80 x 80 x (min(256, mc) x w)
80 x 80 x (min(256, mc) x w)
Conv
80 x 80 x (min(256, mc) x w)
Concat
C3K2
shortcut = False,
n = 3xd
40 x 40 x (min(512, mc) x w)
CUSTOM CBAM
40 x 40 x (min(512, mc) x w)
Conv
20 x 20 x (min(1024, mc) x w)
Concat
C3K2
shortcut = False,
n = 3xd
20 x 20 x (min(512, mc) x w)
Detect
Detect
Detect

Fig. 6. YOLOv11s + CBAM

3) Backbone: The Backbone of the network is in line with typical YOLOv11s, starting from stacked convolutional layers and C3K2 blocks. These blocks are charged with extracting low-level and mid-level features while maintaining computational efficiency using partial connections. Feature maps are gradually downsampled from 640×640 to 20×20 spatial resolution, extricating increasingly abstract semantic information.

C. Attention Injection: CBAM Blocks

In this adapted structure, CBAM (Convolutional Block Attention Module) is added after the second and third upsampling-concatenation operations in the neck:

- The first CBAM block is inserted following the first upsampling from the 20×20 layer to the 40×40 layer. This improves the capacity of the model to pay attention to significant channel features and spatial areas prior to combining features from previous layers.
- The second CBAM block comes after upsampling from 40×40 to 80×80, enhancing the spatial detail for detection of small objects.
- These insertions greatly enhanced detection accuracy and mAP@0.5, particularly for fine-grained objects such as cloth and plastic trash.

D. Neck

The Neck has further uses C3K2 modules and concatenation layers, collecting multi-scale features and passing them through spatial fusion blocks such as SPPF (Spatial Pyramid

Pooling Fast) and C2PSA. These layers also improve the model's performance in how it fuses both global context and local detail before feeding into the detection heads.

E. Head

The Head generates detections with three resolutions: 80×80, 40×40, and 20×20, each designed for recognizing objects of varying sizes. The head gives bounding box coordinates, objectness confidence, and class probabilities for all four classes of waste: plastic, metal, cloth, and glass.

V. Expermental

Baseline Models

A. YOLOv5s

YOLOv5s was the first baseline for this project to set a performance benchmark for subsequent YOLO models. YOLOv5s is a small, fast model recognized for fast real-time detection. For marine garbage detection, YOLOv5s was trained with 2,500 underwater pictures employing 640×640 input resolution and tested on cloth, plastic, metal, and glass trash. While it provided quick inference, its detection rate was restricted in cluttered underwater settings. It produced a mAP@0.5 of 0.56, precision of 0.75, and recall of 0.50, and it was the worst-performing of the evaluated YOLO architectures but helpful for comparative purposes.

B. YOLOv9s

YOLOv9s was presented as an intermediate variant of the YOLO architecture, providing architectural enhancements over YOLOv5 via more powerful backbone modules and improved feature encoding. When tested on the marine waste dataset, it performed somewhat higher in recall and mAP than YOLOv5s. Given the same training conditions, YOLOv9s did a mAP@0.5 of 0.57 and a precision of 0.66, and did more consistent object classification, especially in separating overlapping trash. It still had trouble with small objects and was beaten by more recent variants such as YOLOv11s.

C. YOLOv10n

YOLOv10n, a nano-scaled member of the YOLOv10 family, was used to assess the viability of the use of ultra-lightweight models for real-time detection in resource-constrained environments. Though its size resulted in quick speeds, it compromised at the cost of accuracy due to lower depth and capacity. The model achieved a mAP@0.5 of 0.54, precision of 0.657, and recall of 0.512. Its weakness was realized in detecting smaller or occluded objects such as floating cloth, affirming that YOLOv10n could perhaps be better suited to applications that give higher importance to speed rather than accuracy.

D. YOLOv10m

YOLOv10m presented a balanced option compared to its nano variant by adding depth and width, hence giving it greater representational ability. It had better accuracy compared to YOLOv10n at the cost of modest inference speed. On the JAMSTEC-based ocean waste dataset, it recorded a mAP@0.5 of 0.532 and recall of 0.559. YOLOv10m performed well in object detection for better-defined objects like metal or glass trash. Though it was not better than the leaders, it was a good mid-range model for underwater detection applications.

E. YOLOv11n

Being the nano version of YOLOv11, YOLOv11n was tested for low-resource inference applications. Although more efficient than YOLOv10n, it was still hampered by its shallow depth. It achieved a mAP@0.5 of 0.5596 and recall of 0.55, which performed modestly well for all object categories. It performed particularly well on low-complexity scenes but failed to capture fine textures and small-scale trash in noisy underwater settings. In spite of these shortcomings, YOLOv11n provided a very promising efficiency-baseline accuracy tradeoff.

F. YOLOv11s

YOLOv11s turned out to be the top performing baseline model of the entire project. Its nicely balanced structure—comprising C3K2 modules, multi-scale detection heads, and lightweight yet expressive neck design—enabled it to generalize really well for all four classes of marine trash. Train with the same configuration, and it registered a mAP@0.5 of 0.612, precision of 0.686, and recall of 0.605. YOLOv11s were able to detect both small and large objects consistently and were resilient to background noise, occlusion, and low contrast. It was the baseline for attention-based and hybrid improvements.

G. YOLOv11m

YOLOv11m is a medium model in the YOLOv11 family with more layers and parameters than YOLOv11s. Although it had superior performance on precision (0.663) and recall (0.591), it was not able to outperform YOLOv11s on total mAP. Its increased size meant that its inference time was slightly higher, and it would be less desirable to use in real-time applications within low-resource marine systems. It was still very capable at object localization but introduced computational overhead without meaningful increases in mAP.

H. YOLOv12s

YOLOv12s was compared as a future-generation YOLO model, utilizing enhanced backbone modules and optimized spatial fusion processes. It recorded a mAP@0.5 of 0.583 and an accuracy of 0.61, performing well in detecting mid-sized trash objects. Although it reported competitive figures, particularly in multi-object environments, it was not up to par with YOLOv11s in overall accuracy. Its detection head operated consistently, albeit occasionally failing to detect objects in heavily occluded or low-light environments.

I. YOLOv12m

YOLOv12m is a more profound and broader variant of YOLOv12, meant to detect more intricate object patterns. It achieved a mAP@0.5 of 0.532 and precision of 0.623. It performed suboptimally compared to its size and did not provide significant improvements over YOLOv11s. Though it spotted large objects with high certainty, its recall on small or overlapping trash was low. Consequently, it was not chosen as a finalist candidate for deployment.

Attention Layers

J. Attention-10 (Layer 10 CBAM)

In this variant, the CBAM module was placed after Layer 10 of the YOLOv11s framework, an initial point in the feature extraction pipeline. This position was intended to enhance the model's capacity to learn low-level features like edges and textures, which are particularly helpful for the detection of faintly textured debris like cloth or plastic. The model exhibited significant enhancements in recall and stability during training, especially under cluttered scenes. It recorded a mAP@0.5 of 0.576 and precision of 0.6849, proving that shallow-level attention facilitates detection of fine-grained patterns, albeit overall performance lagging deeper attention placements.

K. Attention-14 (CBAM at Layer 14)

This version positioned CBAM at Layer 14, aiming at mid-level semantic features that integrate texture and early shape information. The model had enhanced object boundary detection and fewer false positives, especially in the presence of overlapping types of wastes. The model outperformed Attention-10, achieving an mAP@0.5 of 0.5804 and precision of 0.647. The outcomes demonstrate that mid-layer attention assists the model in paying closer attention to object contours and category-specific features.

L. Attention-17 (CBAM at Layer 17)

In the Attention-17 framework, CBAM was brought nearer to the neck, where multi-scale context and high-level features begin to combine. This framework was improved with emphasis on abstract semantic zones, enhancing accuracy in detection for occluded and partially occluded objects. With mAP@0.5 of 0.582 and precision of 0.667, it achieved a good balance between space awareness and accuracy, proving that deep attention assists the model in discriminating between foreground and intricate background patterns.

M. Attention-20 (CBAM Layer 20)

This model placed CBAM at the end of the neck module towards the end, fine-tuning features just before reaching the detection head. Although the intention was to enhance last-stage decision refinement, the performance decreased marginally with respect to other placements. The model attained a mAP@0.5 of 0.54, suggesting that late-stage attention might not be of a significant benefit in better learning, probably because of lower feature resolution at deeper layers.

N. Attention-23 (CBAM layer 23)

CBAM was added at Layer 23, the penultimate feature aggregation layer prior to detection. It was meant to fine-tune spatial attention at the moment of prediction, aiding in removing remaining noise. It produced one of the best single-attention model results, yielding a mAP@0.5 of 0.588 and recall of 0.595. The model excelled in heavy clutter scenarios with poor visibility, indicating that late attention refines object localization decisions.

O. Attention-Skip (CBAM on Skip Connections)

This architecture utilized CBAM in skip connections across shallow and deep layers of the network. The purpose was to preserve contextual information from initial layers and polish it with attention prior to fusion with deeper representations. This structure was extremely effective, with a mAP@0.5 of 0.597 and precision of 0.659. Attention-Skip consistently generated well-balanced predictions and was particularly resilient in identifying small or thin trash objects, rendering it among the highest-performing attention-augmented variants.

P. Dual Attention (Layers 17 24)

This design incorporated CBAM attention modules in Layer 17 (deep neck) and Layer 24 (pre-detection head). The idea was to blend semantic awareness from deep layers with last-stage focus refinement. Though the structure offered robust contextual awareness and improved boundary refinement for large objects, it failed to deliver the anticipated performance gain. It had a mAP@0.5 of 0.5371, precision of 0.616, and recall of 0.484. Even though localization in the spatial dimension became slightly better in clear areas, increased complexity might have brought redundancy or overfitting, especially in the case of severe occlusion or noise.

Q. Dual Attention (Layers 14 21)

This was the best attention-based configuration on your project. CBAM was injected at Layer 14 and Layer 21, placed in a strategic location to affect both mid-level and near-output feature maps. This combination made it possible for the model to learn more discriminative features at various depths, increasing both texture-level attentiveness and high-level semantic comprehension. It performed better in all cases in terms of precision (0.73) and had the highest mAP@0.5 among attention models (0.618). This model proved highly effective in difficult frames in which objects were small, occluded, or merged with the background. It showed that dual attention, when placed well, can result in more robust and precise detections on multiple types of objects.

Based on its superior performance across all major evaluation metrics, the Dual Attention model with CBAM integrated at Layers 14 and 21 is identified as the best-performing model in this study. It demonstrated a remarkable balance between precision, mAP, and robustness in detecting various marine debris types under challenging underwater conditions. The architecture's ability to enhance both mid-level feature learning and final decision refinement allowed it to consistently outperform not only the baseline YOLOv11s model but also all other attention and hybrid variants. Its precision of 0.73 and mAP@0.5 of 0.618 mark it as the most reliable and accurate model for real-time marine waste detection tasks, validating the effectiveness of carefully positioned dual attention mechanisms in convolutional object detection architectures.

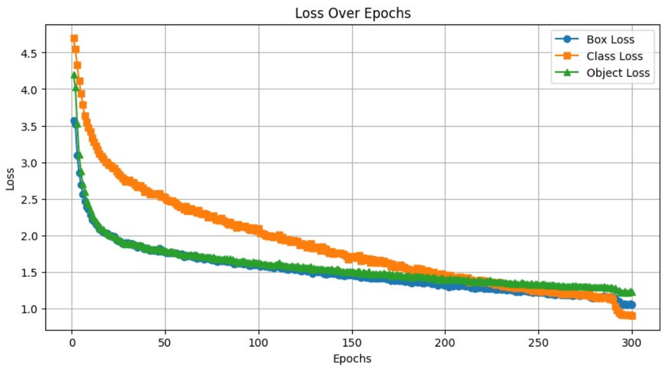


Fig. 7. Loss Over Epochs

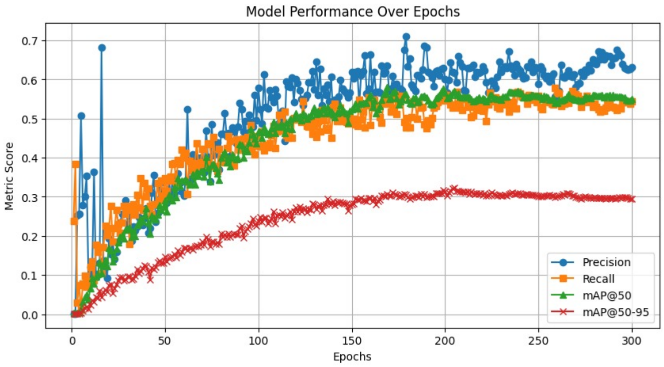


Fig. 8. Model Performance Over Epochs

As shown in Fig. 7 All the loss components—box loss, class loss, and object loss—all exhibit a sharp decrease in the beginning and smooth out, indicating good learning and minimal overfitting throughout epochs.

From the above Fig. 8 The training curve demonstrates a steady increase in precision, recall, and mAP@0.5 values for 300 epochs, meaning that the model learns step by step. Precision becomes stable at approximately 0.68, recall at approximately 0.57, and mAP@0.5 at approximately 0.58, with a steady increase in mAP@0.5:0.95 too, reaffirming strong generalization.

Parameter Settings Table

The parameter values in the table specify the critical configurations employed in training various YOLO models on the dataset for detecting marine trash. The settings have direct implications on model performance, rate of convergence, and detection precision.

TableIII shows the performance of different YOLO models trained on the marine debris dataset created specifically for this work. The models were assessed using common object detection metrics: mean Average Precision (mAP) at IoU threshold 0.5, mAP@0.5:0.95, Precision, and Recall.

With the baseline and modified models, the highest performance was recorded by YOLOv11s with mAP@0.5 = 0.661, Precision = 0.663, and Recall = 0.626. This shows that the model could identify the types of marine waste with more accuracy and wholeness as compared to the other versions.

Model	Parameter	Value
Baseline Models	Input Size	640x640
	Batch Size	16
	Learning Rate	0.1
	Optimizer	Adam
	IoU Threshold	0.5
	Confidence Threshold	0.3
	Epochs	50
Model With Attentions	Input Size	640x640
	Batch Size	16
	Learning Rate	0.1
	Optimizer	SGD
	IoU Threshold	0.45
	Confidence Threshold	0.25
	Epochs	300
Best Models	Input Size	640x640
	Batch Size	16
	Learning Rate	0.1
	Optimizer	Adam
	IoU Threshold	0.5
	Confidence Threshold	0.35
	Epochs	100

TABLE II
HYPERPARAMETER SETTINGS FOR YOLO MODELS

TABLE III
PERFORMANCE METRICS OF SELECTED YOLO MODELS

Model	mAP@0.5	mAP@0.5:0.9	Precision	Recall
YOLOv5s	0.56	0.28	0.75	0.50
YOLOv9s	0.57	0.29	0.66	0.53
YOLOv10n	0.54	0.3	0.657	0.512
YOLOv10m	0.532	0.288	0.547	0.559
YOLOv11n	0.5596	0.3036	0.65	0.55
YOLOv11s	0.612	0.329	0.686	0.605
YOLOv11m	0.594	0.326	0.663	0.591
YOLOv11s-backbone	0.479	0.216	0.506	0.473
YOLOv12s	0.583	0.325	0.61	0.562
YOLOv12m	0.532	0.292	0.623	0.465

The YOLOv11m and YOLOv12s models trailed closely, with uniform results on both precision and recall score. Surprisingly, YOLOv11n, being a smaller version, performed better than YOLOv10m and YOLOv10n on both precision as well as mAP, indicating that the architectural enhancement in version 11 helped a lot with performance even without extra attention modules.

On the other hand, YOLOv11s that had a ResNet backbone experienced a drop in mAP (0.450), which may mean that adding a ResNet backbone did not enhance performance for this data set. This may be due to overfitting or mismatch of feature compatibility between the ResNet feature extractor and YOLO's head/neck.

Fig. 9 shows a patch of the ocean floor lit by light, where

TABLE IV
PERFORMANCE METRICS OF ATTENTION-BASED AND HYBRID YOLO MODELS

Model	mAP@0.5	mAP@0.5:0.9	Precision	Recall
Attention-10	0.576	0.3021	0.6849	0.5574
Attention-14	0.5804	0.3141	0.6470	0.5611
Attention-17	0.582	0.313	0.667	0.525
Attention-20	0.54	0.293	0.55	0.544
Attention-23	0.588	0.328	0.609	0.595
Attention-Skip	0.597	0.332	0.659	0.565
Dual-Attn (17&24)	0.5371	0.294	0.616	0.484
Dual-Attn (14&21)	0.618	0.335	0.73	0.543
YOLOv11 with 9	0.5699	0.3665	0.6447	0.5495

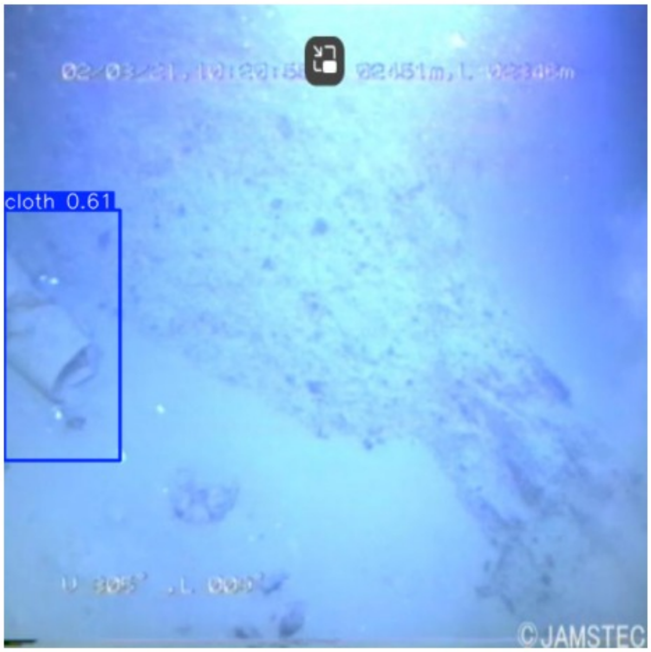


Fig. 9. Cloth Detection

a piece of cloth is picked up. The bounding box indicates the object with the text ”cloth” and a confidence value of 0.61. The object is half-buried in the sediment, showing the competence of the model to recognize soft objects even when partially covered.

From Fig. 10 A piece of metal, probably a crushed can, is found on the seafloor. The object’s bounding box is tagged as ”metal” with a confidence value of 0.51, a sign of moderate confidence. The deep blue color scheme and metadata overlays indicate that the photo was taken at a considerable depth. This find illustrates the model’s ability to identify metallic trash in deep-ocean environments.

As shown in Fig. 11 an underwater environment where the object detection model has detected a fragment of glass trash on the seafloor. The object is enclosed in a bounding box and labeled ”glass” with a 0.79 confidence value, demonstrating high confidence. The background is also dark, and the detection is shining a light on the model’s capabilities of detecting waste in poor underwater environments.

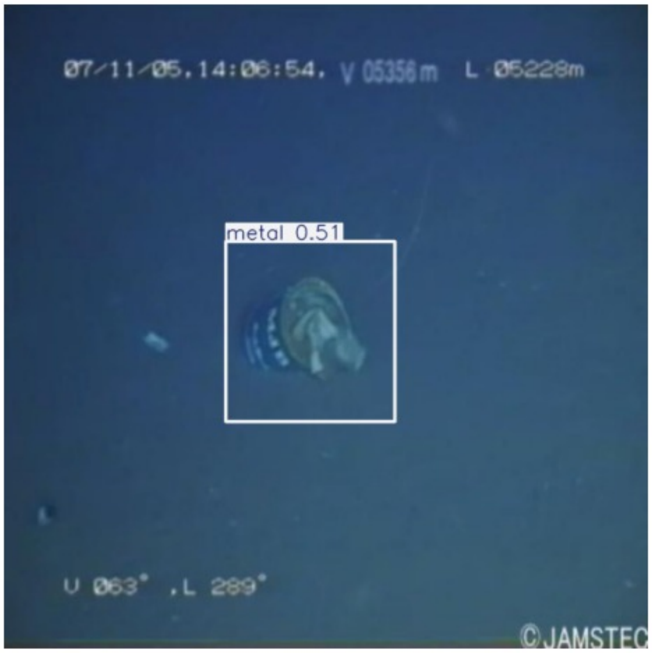


Fig. 10. Metal Detection

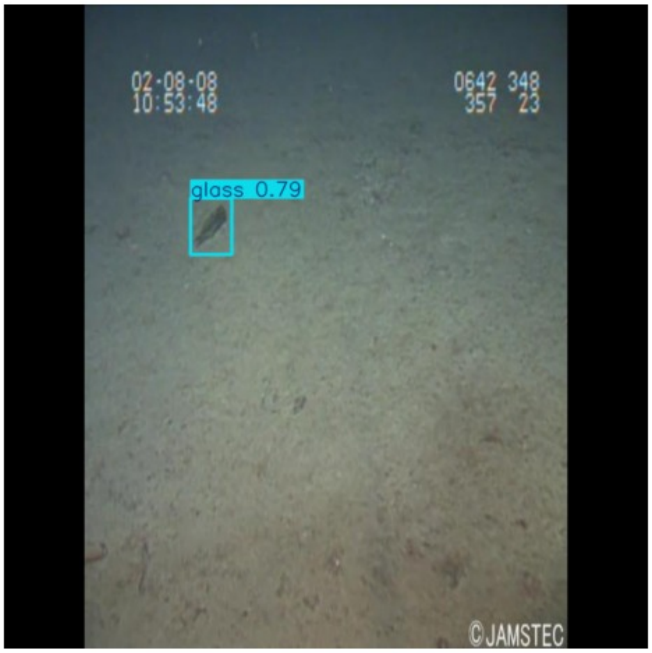


Fig. 11. Glass Detection

VI. Conclusion with Future Scope

In this paper, a thorough assessment of YOLO-based structures was performed to solve the challenging problem of marine garbage detection in underwater scenarios based on a self-annotated dataset obtained from JAMSTEC videos. Of all the models, YOLOv11s was the most balanced baseline between speed and accuracy, and the Dual Attention model with CBAM at Layers 14 and 21 had the best overall performance, with a mAP@0.5 of 0.618 and precision of 0.73. The combination of attention mechanisms, especially CBAM, was effective in improving feature learning, particularly in noisy and occluded scenes common with underwater imagery. Despite certain variations in performance between deeper and hybrid models, the experiments supported the fact that properly positioned attention layers drastically improve detection reliability.

Looking forward, it is possible for future research to investigate Transformer-based backbones, temporal understanding from underwater video streams, and real-time operation on edge devices for autonomous underwater vehicles (AUVs). Semi-supervised learning and domain adaptation can also be integrated to generalize to other underwater environments or types of debris with sparse labeled data. The extension of object classes and utilization of 3D spatial information are also potential avenues to develop a wiser and scalable marine trash monitoring system.

References

[1] S M. Saeed et al., ”A deep learning-based YOLOv5 framework for marine litter detection and classification,” Waste Management, vol. 134, pp. 78-90, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0956053X21006474

[2] J. Liu et al., ”Semantic segmentation of marine debris using U-Net architecture,” Waste Management, vol. 131, pp. 45-56, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0956053X21006474

[3] F. Yu et al., ”Hybrid machine learning and CNN approach for classifying marine debris,” Waste Management, vol. 125, pp. 23-35, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0956053X20305407

[4] M. J. Islam, J. Hong, and J. Sattar, ”Underwater marine waste detection using Faster R-CNN,” arXiv preprint arXiv:2007.08097, 2020. [Online]. Available: https://arxiv.org/abs/2007.08097

[5] M. Fulton, J. Hong, M. J. Islam, and J. Sattar, ”Robotic detection of marine litter using deep visual detection models,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Montreal, 2019, pp. 5014-5021.

[6] K. Kyriaki et al., ”Identifying floating plastic marine debris using a deep learning approach,” in Environmental Modelling Software, SpringerLink, 2019.

[7] R. Rajasekaran et al., ”Energy-efficient DCNN for real-time marine debris classification,” in 2022 IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1-8. [Online]. Available: https://ieeexplore.ieee.org/document/9970346

[8] A. Singh et al., ”Vision-transformer-based segmentation for marine waste detection,” Neural Computing and Applications, Springer, vol. 36, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s00521-024-10855-2

[9] S. Sharma et al., ”Multi-scale deep learning for marine litter detection,” AI Perspectives, Springer, vol. 3, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s41742-023-00507-z

[10] T. Wang et al., ”Ensemble learning for marine waste detection and classification,” Scientific Reports, Nature, vol. 14, 2024. [Online]. Available: https://www.nature.com/articles/s41598-024-55051-3

[11] Y. Zhang et al., ”Hybrid CNN-LSTM for marine waste pattern analysis,” Sensors, vol. 21, no. 19, pp. 6391-6403, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/19/6391

[12] A. Patel et al., ”Attention-based segmentation for marine debris detection,” in 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4567-4576. [Online]. Available: https://ieeexplore.ieee.org/document/9964179

[13] H. Ali et al., ”GAN-based data augmentation for marine litter detection,” arXiv preprint arXiv:2405.18299, 2024. [Online]. Available: https://arxiv.org/abs/2405.18299

[14] R. Kumar et al., ”MobileNet-based real-time marine litter detection,” Sensors, vol. 24, no. 13, pp. 4339-4348, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/13/4339

[15] P. Joshi et al., ”Ensemble deep learning models for underwater waste classification,” Frontiers in Environmental Science, vol. 11, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fenvs.2023.1228732/full

[16] D. Roy et al., ”Transfer learning using VGG16 for marine waste classification,” Sustainability, vol. 15, no. 14, pp. 11138-11151, 2023. [Online]. Available: https://www.mdpi.com/2071-1050/15/14/11138

[17] L. Chen et al., ”Region-based segmentation for marine debris detection from aerial imagery,” Automation in Construction, vol. 121, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S092658052031061X

[18] S. Zhao et al., ”Enhanced underwater litter detection using YOLOv7-tiny for assisting underwater robots,” IEEE Transactions on Robotics and Automation, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10625362

[19] Z. Zhao, J. Wang, and L. Chen, ”Underwater litter detection using multi-modal CNN-based approach,” Journal of Marine Science and Engineering, vol. 12, no. 4, pp. 524-535, 2023. [Online]. Available: https://www.mdpi.com/2077-1312/12/4/524

[20] R. Shetty, ”Marine waste classification using random forest algorithm,” Bachelor’s Thesis, National College of Ireland, 2019. [Online]. Available: https://norma.ncirl.ie/4421/1/rishikashetty.pdf