ISSN NO: 2249-3034

Marine Trash Detection and Segmentation using Convolutional Neural Networks

Dr.THOMAS FELDMAN,PROF.KALAM NARREN

[1] Assistant Professor, Potti Sriramulu Chalavadi Mallikarjuna Rao College of Engineering and Technology (Autonomous Institution), Vijayawada.

[2],[3],[4],[5] Student,Potti Sriramulu Chalavadi Mallikarjuna Rao College of Engineering and Technology (Autonomous Institution),Vijayawada

ABSTRACT— The Trash in the marine is now becoming a serious concern in today's world, which has a dangerous threat to the marine life due to the persistence of debris for long periods. This problem is also impact on human beings due to the consumption of marine food. This problem can be solved by detecting the Trash in the marine and must be cleansed either by humans or autonomous robot systems. The job done by humans may not give the best results in terms of productivity. So ultimately choosing the autonomous robot systems plays a very important and crucial role in these types of situations which increases the productivity by reaching our goals. Our model is trained using Convolutional Neural networks to detect the marine trash which increases the accuracy involved in detection, by using YOLOv8 which is more accurate and faster when compared to other versions of YOLO. Here we also added Labelling and Segmentation to it to achieve the task in unexpected moments like proper detection and evaluation. Segmentation gives its importance when underwater detection is complex by giving the dimensions of the body and tells us whether it is a trash or not. The detection in marine field involves complex tasks and algorithms due to presence of low light in the underwater areas where the objects will not be Clear. So finally, our model can be utilized to deploy in Autonomous robot system to perform such complex tasks in Marine trash detection and Instance Segmentation due to evolvement of technology in the field of Autonomous Robot Systems or Autonomous Underwater Vehicles (AUV).

I. INTRODUCTION

In recent times, the issue of marine trash pollution has gained significant attention due to its mischievous goods on marine ecosystems and littoral communities. The presence of marine debris poses serious pitfalls to submarine life, mortal health, and the overall environmental balance. Effective operation and mitigation strategies bear accurate discovery and segmentation of marine trash, which can be a grueling task in large- scale marine surroundings. To address this challenge, convolutional neural networks (CNNs) have surfaced as important tools for automated image analysis and object discovery tasks. using their capability to learn hierarchical features from raw input data, CNNs have shown promising results in colorful computer vision operations, including object recognition, segmentation, and bracket. In this paper, we propose a new approach for marine trash discovery and segmentation using CNNs. Our system aims to directly identify and delineate marine debris from aquatic images, enabling effective monitoring and operation of marine pollution. By training a CNN on a different dataset of annotated marine images, we demonstrate the effectiveness of our approach in directly detecting and segmenting different types of marine trash across varying environmental conditions.

The remainder of this paper is organized as follows in Section II, we give a comprehensive review of affiliated work in the field of marine debris discovery and CNN- grounded image segmentation. Section III details the methodology employed in our approach, including dataset collection, network armature, training procedure, and performance evaluation. Section IV details about the mathematical analysis such as equations and expressions used in this model from top to bottom. Experimental results are presented in Section V, followed by Conclusion and perceptivity in Section VI.

A. PROBLEM DESCRIPTION AND OVERVIEW

Problem description Marine debris is a major environmental problem that has a negative impact on marine life, ecosystems, and mortal health. It can also be a nautical hazard for vessels and boats. The quantum of marine debris is adding every time, and it is estimated that there are over 150 million tons of marine debris in the world's abysses. Overview YOLO (You Only Look Once) is a real- time object discovery algorithm that can be used to descry marine debris in images and vids. YOLO is a one- stage object discovery algorithm, which means that it can descry objects in a single pass through the image. This makes YOLO important faster than two- stage object discovery algorithms, similar as R- CNN and Fast R- CNN.

B.OBJECTIVES

The ideal of marine trash discovery and segmentation using YOLO is to identify and descry marine debris in images and vids. This can be used to

● Reduce the number of marine debris in the abysms. By relating and locating marine debris, it can be collected and disposed of properly. This helps to cover marine life and ecosystems.

● Meliorate the safety of marine navigation. Marine debris can be a navigational hazard for vessels and boats. By relating and locating marine debris, it can be avoided.

● Increase public awareness of the marine debris problem. By making marine debris visible, it can raise awareness of the problem and encourage people to take action to reduce it. YOLO is an important tool that can be used to achieve these objects. It's a fast and effective object discovery algorithm that can be used to descry marine debris in real time. This makes it a good option for associations that want to develop their own marine trash discovery system.

II. LITERATURESURVEY

Upon analyzing all conference papers, numerous challenges were identified by Girshicketal in 2014, encountered issues with dataset redundancy, particularly concerning the supplementary dataset transitioning between training and testing datasets [2]. In their exploration of object detection, Redmon etal in 2016 introduced YOLO, a novel approach that consolidates object discovery tasks into a single network, promising efficiency and end-to-end performance. The authors assert that YOLO achieves impressive real-time processing speeds, capable of handling images at 45 frames per second. Compared to alternative object detection algorithms, YOLO demonstrates superior localization accuracy while maintaining a low false positive rate on background elements [3]. Building upon this framework, Redmon and Farhadi in 2018 introduced YOLOv3, incorporating minor modifications [6]. Notably, YOLOv3 utilizes Logistic Regression to predict objectness scores for each bounding box, ensuring a score of 1 when the ground truth overlaps with the box. Furthermore, YOLOv3 employs a new network architecture called Darknet-53, consisting of 53 convolutional layers. Darknet, written in C or CUDA, serves as the foundational framework for training neural networks and forms the backbone of YOLO.

In 2019, Valdenegro-Toro also faced challenges despite achieving good discovery sensitivity; the system struggled to categorize debris types, although it could detect their presence in images. The training dataset used in this study lacked representative debris shapes typically found in marine environments. Nonetheless, the experimenter remains optimistic that training on similar datasets will yield desired results. Recognizing the critical importance of marine conservation, various industries such as fishing and tourism hold significant potential. However, progress in these sectors has been hindered by technological limitations, particularly in underwater exploration [10]. YOLOv3, an advanced object detection algorithm, was introduced to address complex backgrounds in submarine environments and detect marine life and debris effectively (Watanabe etal., 2019). Additionally, emerging deep learning algorithms, including DeepMultiBox proposed by Erhan etal in 2014, are gaining popularity for training sensors to identify objects in images. Wang etal in 2016, developed the SOAR robot, a vision surveillance system integrated with an android smartphone and robotic fish, designed specifically for real-time detection of marine debris in submarine terrains. The SOAR robot's success in detecting and addressing marine debris underscores its efficacy in navigating challenging underwater environments.

In 2021, Bing Xue and Baoxiang Huang and others, proposed a Deep-Sea Debris Identification using Deep convolutional Neural Networks. The main objective among them is to determine the whether deep neural networks can distinguish between real debris and natural sea environment. So, they mainly they gathered a dataset for the classification of images and construction of hybrid Shuffle Xception network for classification of different images. Finally, they experimented on the above procedure [14]. In 2022, Wei Zhou, Fujian Zheng and others proposed a Deep learning Marine Debris Detection Network called YOLO Trashcan, consists of feature enhancement and feature fusion defined for (Efficient Channel Attention) ECA_DO-Conv_CSPDarknet53 and (Dilated Parallel Module) DPMs_PixelShuffle

ISSN NO: 2249-3034

_PANET which is only for size of the network 214MB. They validated their model on Trashcan 1.0 dataset [16]. In 2022, Shailendra Kumar, Abhinav Gautam and others, proposed a Deep learning Framework for Macro Marine Litter Classification and Quantification which tells about the type and quantity of several types of marine waste for the better management. For this purpose, they proposed a line sweep algorithm to find the approximate area cover of the marine litter. They mainly focus on management and monitoring to help the government [17]. In 2022, Takuya Kiyokawa, Jun Takamatsu and others discussed the challenges for future Robotic Sorters of Mixed Industrial waste [18]. In 2022, Haruna Abdu, and Mohd Halim Mohd Noor, conducted a survey regarding waste detection and classification using Deep learning. They described the art of deep learning models that insight into the research areas that can still be explored. They reviewed different image classification and waste detection models providing an analysis with precise and organized representation [19]. In 2023, Shu-Min Tsai, Ming-Lin Chuang and others discussed an application of Beach Trash Detection System based on Deep learning. To keep the beach clean, their work proposes an automatic beach trash removing system that integrates trash detection and unmanned vehicle using deep learning. A single chip system is used to control the unmanned vehicle [20].

A. EXISTING METHOD

Crowdsourcing platforms, analogous as The Ocean Cleanup and Marin Eye, allow stoners to upload images and vids of marine debris. These images and vids are also analyzed by impositions to identify and classify marine debris. audial discovery aural discovery systems, analogous as MARLIN and MARLIN- Acoustic, can be used to descry marine debris by listening for the sound of swells hitting the debris. These systems are not as considerably used as other systems, but they can be effective in detecting large pieces of debris. Remote seeing Remote seeing technologies, analogous as satellites and drones, can be used to descry marine debris from a distance. These systems are not as accurate as other systems, but they can be used to survey large areas of water for marine debris. mortal observers' mortal observers can be used to descry marine debris by visually examining the water. This system is not as effective as other styles, but it can be effective in detecting small pieces of debris that may be missed by other styles.

B. PROPOSED METHOD

YOLO (You Only Look formerly) is a real- time object discovery algorithm that can be used to descry marine trash in images and videos. YOLO is a one-stage object discovery algorithm, which means that it can descry objects in a single through the image. This makes YOLO important faster than two- stage object discovery algorithms, similar as R- CNN and Fast R-CNN. YOLO can be used to descry a wide range of marine debris, including plastic bottles, plastic bags, fishing nets, and other objects. It's also fairly easy to train and emplace, making it a good option for associations that want to develop their own marine trash discovery system. Then are some of the benefits of using YOLO for marine trash object discovery.

● Real- time discovery YOLO can descry marine debris in real time. This means that it can be used to cover marine surroundings for marine debris on a nonstop base.

● Wide range of discovery YOLO can descry a wide range of marine debris, including plastic bottles, plastic bags, fishing nets, and other objects.

● Easy to use YOLO is easy to use and emplace. This makes it a good option for associations that don't have a lot of experience with object discovery.

III. METHODOLOGY

A.PROJECT REQUIREMENTS

Also are some of the design conditions for marine trash discovery and segmentation using YOLOv8 Data collection A large dataset of images and vids is demanded to train a YOLOv8 model for marine trash discovery and segmentation. This dataset should include a variety of marine debris, including plastic bottles, plastic bags, fishing nets, and other objects. The dataset should also be representative of the different environmental conditions that YOLOv8 will be used in, analogous as low- light conditions and water with high turbidity. We used Trash- ICRA19 A Bounding Box Labeled Dataset of Submarine Trash Model training YOLOv8 can be trained using a variety of machine knowledge fabrics, analogous as TensorFlow and PyTorch. The training process can be time- consuming, and it may bear a important computer to train the model on a large dataset.

B.PROCEDURE

1. Data Collection:

The study relied upon the Japan Agency for Marine-Earth Science and Technology Deep Debris Database, a comprehensive repository comprising real-life videos and captured images portraying underwater debris. Notably, this dataset encompasses the intricate realities of submarine environments, where lighting conditions can vary significantly, affecting image quality. Despite these inherent challenges, the dataset offers a substantial collection of 6008 training illustrations and 1204 validation cases. This wealth of diverse data not only enriches the study's analysis but also underscores its robustness and relevance in addressing marine debris detection and management.

2. Data Classification:

There are a total of 16 classes in the dataset. The following are the classes, along with their label indices. 1: 'plant',
2: 'animal fish',
3: 'animal_starfish',
4: 'animal_shells',
5: 'animal_crab',
6: 'animal_eel',
7: 'animal_etc',
8: 'trash_etc',
9: 'trash_fabric',
10: 'trash_fishing_gear',
11: 'trash_metal',

ISSN NO: 2249-3034

12: 'trash_paper',
13: 'trash_plastic',
14: 'trash_rubber',
15: 'trash_wood',

3. Data Preparation:

Data preparation is not merely a procedural step but a pivotal aspect of research, laying the groundwork for comprehensive analysis and meaningful insights. In the case of the JAMSTEC Deep-sea Debris dataset, the initial data structure presented a challenge, as it was in a list format rather than the requisite image format. To address this, sophisticated web scraping techniques were employed, harnessing the capabilities of Parse Hub. Section 5 provides an exhaustive exposition on how Parse Hub was adeptly utilized for web scraping, offering a detailed glimpse into the meticulous process of converting the dataset into an image-centric format. This transformation not only enhances the dataset's accessibility but also unlocks its full analytical potential, empowering researchers to glean deeper insights into underwater debris dynamics.

4. Modelling:

Modelling is an essential initial phase encompassing the development of a deep learning model, its implementation, and the evaluation process employed to assess results. Within this framework, the deep learning model YOLO, which stands for "You Only Look Once," is utilized for object detection. Accompanying YOLO is Darknet, primarily a Neural Network Framework employed for training the sensor. The synergy between YOLO and Darknet is pivotal in this research endeavor, facilitating the detection of objects both in isolation and within larger contexts.

C. YOLOv8 INSTANCE SEGMENTATION

In this section, we present the methodology for YOLOV8 instance segmentation, detailing the network architecture, training procedure, and evaluation metrics.

1. Network Architecture

The YOLOv8 instance segmentation model is built upon the YOLO (You Only Look Once) architecture, which is known for its efficiency and real-time performance. The backbone of the network consists of a series of convolutional layers followed by max-pooling operations, enabling the extraction of hierarchical features from input images. Unlike traditional object detection frameworks, YOLOv8 utilizes a single convolutional network to simultaneously predict bounding boxes and segmentation masks for multiple object instances in a single pass.

To achieve instance segmentation, YOLOv8 incorporates additional layers for pixel-wise classification and segmentation
mask prediction. Specifically, a series of up sampling and
convolutional layers are employed to generate dense prediction maps corresponding to object boundaries and regions of interest. By leveraging skip connections and feature fusion techniques, the network effectively captures fine-grained spatial information and semantic context, facilitating accurate instance segmentation

across diverse object categories.

2. Training Procedure

The training procedure for YOLOv8 instance segmentation involves several key steps to optimize model performance and generalization capability. Firstly, a large-scale annotated dataset containing diverse object instances is collected and preprocessed to ensure uniformity and consistency. Data augmentation techniques such as random cropping, rotation, and color jittering are applied to augment the training dataset and improve model robustness.

Next, the YOLOv8 model is initialized with pretrained weights on a large-scale image classification dataset such as ImageNet. Fine-tuning is performed using the annotated instance segmentation dataset, where the network parameters are updated through backpropagation and stochastic gradient descent optimization. To mitigate overfitting, regularization techniques such as dropout and weight decay are employed, along with early stopping criteria based on validation performance.

During training, the loss function is computed based on both object detection and segmentation objectives, incorporating terms for bounding box regression, objectness score prediction, and pixel-wise segmentation accuracy. Multi-task learning is employed to jointly optimize these objectives, ensuring consistent and coherent predictions across different tasks.

3. Evaluation Metrics

oTo evaluate the performance of YOLOv8 instance segmentation, a comprehensive set of metrics is employed to quantify detection accuracy, segmentation quality, and computational efficiency. Standard metrics such as mean Average Precision (mAP) and Intersection over Union (IoU) are used to assess object detection performance, measuring the precision and recall of predicted bounding boxes relative to ground truth annotations.

oFor segmentation quality evaluation, pixel-wise metrics such as Pixel Accuracy, Mean IoU, and F1 Score are computed to quantify the similarity between predicted segmentation masks and ground truth annotations. Additionally, runtime performance metrics including inference time and memory footprint are measured to evaluate the computational efficiency of the proposed method.

oThe evaluation of the YOLOv8 model using the confusion matrix revealed valuable insights into its performance. By analyzing metrics such as precision, recall, and F1 score derived from the confusion matrix, we were able to assess the model's ability to correctly classify objects across various classes. Our results indicated high precision and recall for certain classes, while others exhibited lower performance. Through a detailed examination of the confusion matrix, we identified specific areas where the model struggled, such as detecting small or occluded objects. These findings highlight the importance of fine-tuning model parameters and training data augmentation techniques to improve the YOLOv8 model's performance in challenging scenarios.

oThe following figures shows the confusion matrix for nano and medium models of project to detect and segment:

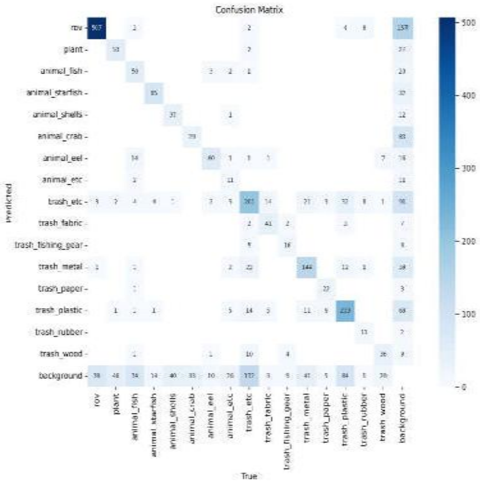


Fig.1(a): Confusion matrix for detection using medium model.

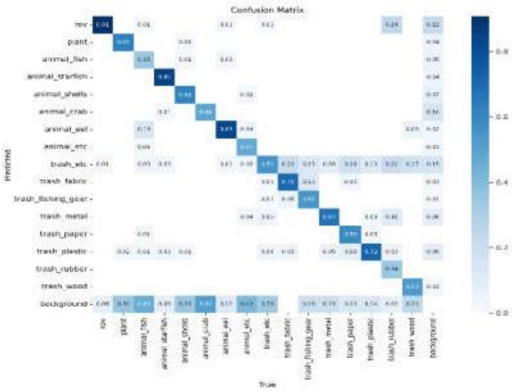


Fig.1(b): Confusion matrix for segmentation using medium model.

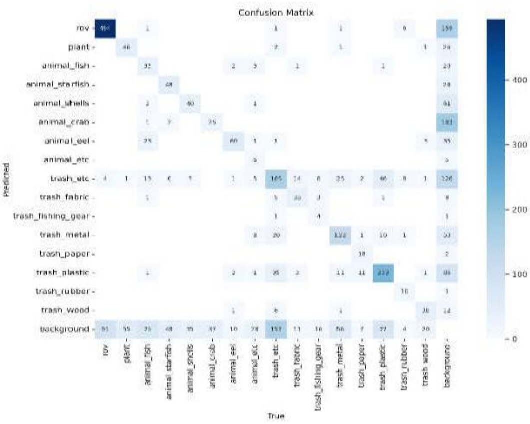


Fig.1(c): Confusion matrix for detection using nano model.

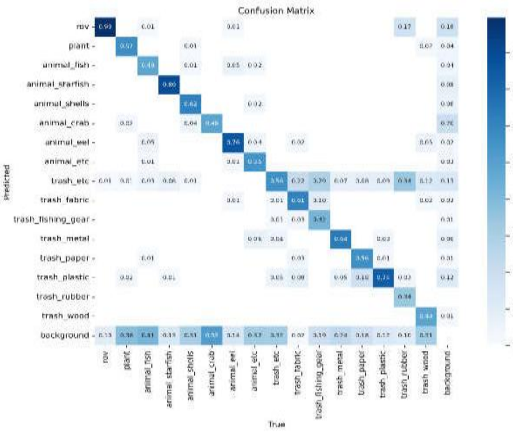


Fig.1(d): Confusion matrix for segmentationusing nano model.

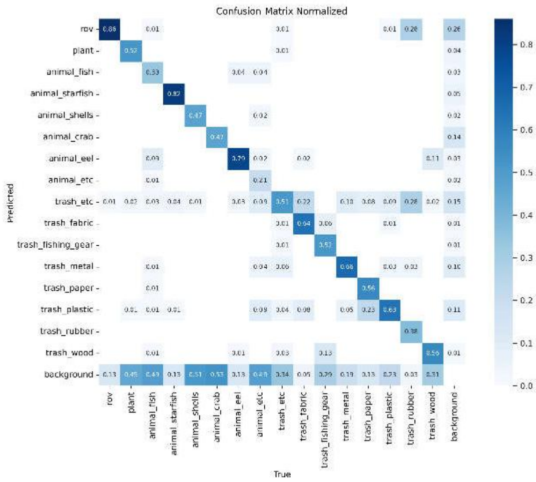


Fig.1(e): Confusion matrix of normalization detection using medium model.

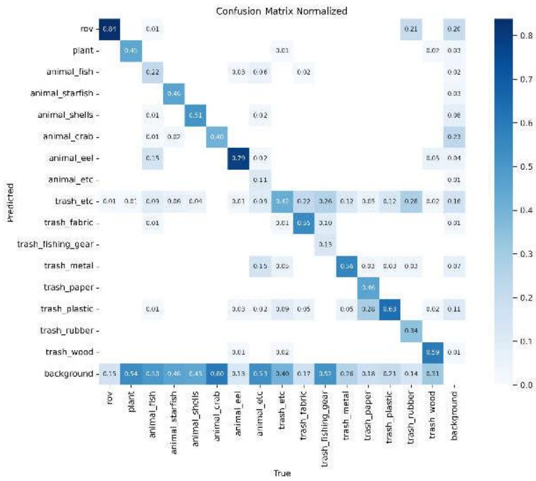


Fig.1(f): Confusion matrix of normalization detection using nano model.

D. IMPLEMENTATION

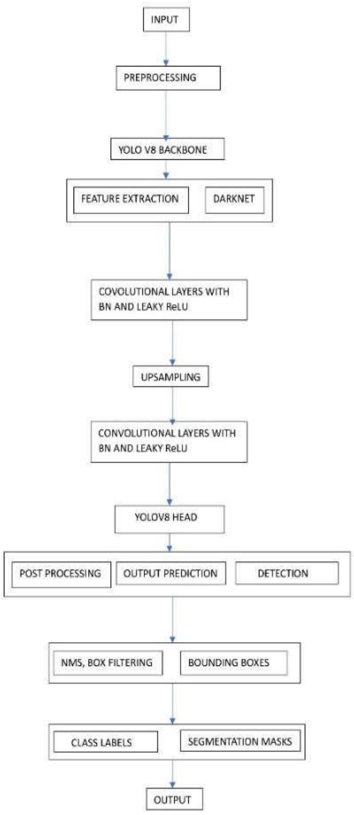


Fig.2 Depicting the implementation procedure of YOLOv8 model.

o Input Image: This is the starting point of the process. The image data is fed into the system.

o Preprocessing: This stage prepares the image data for the YOLOv8 model. It might involve resizing the image, scaling the pixel values, or applying random transformations like cropping, flipping, or color jittering to create more diverse training data and improve the model's generalization capability.

oYOLOv8 Backbone (Feature Extraction Backbone): This is the initial stage of the YOLOv8 model that's responsible for extracting features from the input image. It consists of multiple convolutional layers, each with a Batch Normalization layer and a Leaky ReLU activation layer. Similar to YOLOv3, here's a breakdown of what each layer does:

oConvolutional layer: Extracts features from the image by applying filters and performing convolutions.

oBatch Normalization layer: Speeds up learning and improves the stability of the training process.

oLeaky ReLU activation layer: Introduces non-linearity into the network, allowing it to learn more complex features.

oYOLOv8 Head (Detection Head): This stage of the model takes the extracted features from the backbone and uses them to detect objects. It consists of multiple convolutional layers with Batch Normalization and Leaky ReLU activation layers, followed by up sampling layers to increase the resolution of the feature maps.

oPost-processing: This stage refines the predictions made by the model. It might involve techniques like non-max suppression (NMS) to remove duplicate bounding boxes and box filtering to improve the accuracy of the bounding boxes.

oOutput Prediction: This is the final stage of the process where the model outputs the detected objects with bounding boxes, class labels, and potentially segmentation masks.

IV. MATHEMATICAL EQUATIONS AND EXPRESSIONS

YOLOv8 incorporates convolutional neural networks (CNNs) at its core for feature extraction. The mathematical operation essential for this process is convolution. Each element (i, j) in the output feature map is determined by calculating convolutional operations. This integral process lies at the heart of YOLOv8's ability to detect and classify objects within images accurately.

1. Convolution:

Output [i, j] = Σ (Input [m, n] * Kernel [i - m, j - n]) + Bias

Whereas,

oInput is given by a tensor representing the image data (often with width, height, and channels).

oKernal is a filter with a smaller tensor that slides across the input, capturing local features.

om, n are Iterators for positions within the kernel.

oBias is a scalar value added to the element-wise product for each position.

2. This model explores how the core functionality can be utilized for this marine trash detection and segmentation.

3. Classification (Trash vs non-trash):

Score(trash) = W_c^T * x + b_c[t]

Whereas,

oW_c be the weight matrix for the class layer

ob_c[t] be the bias vector.

4. Activation Function:

This function transforms the score into a probability between 0 (not trash) and 1 (likely trash).

This function introduces non-linearity, allowing the network to learn complex patterns. Common activation functions include:

oLeaky ReLU (Rectified Linear Unit): **max (0, Input)**

oSigmoid: **1 / (1 + exp(-Input))**

5. Pooling:

Pooling layers, such as max pooling, play a crucial role in reducing the dimensionality of data while retaining essential features. An example of this is outlined in the formula provided: Pooled Output [i, j] equals the maximum value within a specified region of the input, determined by the stride. Here, the stride parameter defines the increment between successive pooling operations as the window traverses the input. This approach ensures that crucial features are preserved while down sampling the data. Max pooling effectively condenses the input information, facilitating more efficient computation in subsequent layers of the neural network, all while maintaining the integrity of significant features essential for accurate learning and classification tasks.

Pooled Output [i, j] = max (Input [i * Stride: (i + 1) * Stride, j * Stride: (j + 1) * Stride])

6. Prediction (Bounding Boxes):

Output Channels: YOLOv8 predicts multiple bounding boxes and their associated class probabilities within a single image. The number of output channels in the convolutional layers dedicated to bounding box prediction would depend on the model configuration (e.g., number of predicted boxes per grid cell).

Bounding Box Parameters: These channels encode parameters for each potential bounding box, often as offsets from a pre-defined grid within the feature map. Let's denote predicted box parameters for centre coordinates (cx, cy), width (w), and height (h) as (t_cx, t_cy, t_w, t_t). The mathematical form might be specific to YOLOV8's implementation, but it would involve convolutions and potentially scaling based on grid cell size.

7. Segmentation Mask (Adaptation):

The simplified view of adaptation is given by Decoder

Layers which take the feature maps containing bounding box information as input, Up sampling Layers which increase the resolution of the feature maps to match the original image size for mask prediction, and Convolutional Layers which learn to generate a binary mask for each instance within a bounding box. The mask prediction can be formulated as:

$$M\_i\ (x, y) = \sigma\ (W\_m * F\ (x, y) + b\_m)$$

Whereas,

oM_i (x, y) represent the predicted mask value (0 or 1) for pixel location (x, y) in mask i (corresponding to the i-th bounding box).

oW_m is Weight matrix for the mask prediction layer.

oF (x, y) is Feature vector at pixel location (x, y) in the unsampled feature map.

ob_m is Bias vector for the mask prediction layer.

oΣ is Sigmoid activation function, converting the output to a probability between 0 (background) and 1 (foreground - belonging to the trash instance).

8. YOLOv8 likely uses a combination of these operations with additional layers like batch normalization for training.

9. This model explores how the core functionality can be utilized for this marine trash detection and segmentation.

10. YOLOv8 Loss function components:

11. Classification Loss (L_cls):

This penalizes errors in predicted class probabilities (trash vs. non-trash). A common choice is the binary cross-entropy loss:

$$L\_cls\ (t, \sigma(z\_t)) = - (t * \log(\sigma(z\_t)) + (1 - t) * \log(1 - \sigma(z\_t)))$$

Whereas,

oT is Ground truth label (1 for trash, 0 for non-trash).

oσ(z_t) is Predicted probability of the "trash" class from the model's output (sigmoid activation typically used).

12. Bounding Box Loss (L_box):

This penalizes errors in predicted bounding box parameters (center coordinates, width, height). The Intersection over Union (IoU) loss is a common choice:

$$L\_box = 1 - IoU\ (bbox\_pred, bbox\_gt)$$

Whereas,

obbox_pred: Predicted bounding box parameters (cx, cy, w, h).

obbox_gt: Ground truth bounding box parameters for a trash object.

oIoU: Intersection over Union function (various formulations exist).

13. Mask Loss (L_mask):

It penalizes errors in the predicted segmentation mask. A common choice is the binary cross-entropy loss:

$$L\_mask = - (mask\_gt * \log(\sigma\ (M\ (x, y))) + (1 - mask\_gt) * \log(1 - \sigma\ (M\ (x, y))))$$

Whereas,

omask_gt is Ground truth binary mask for a trash instance (1 for object pixels, 0 for background).

oM (x, y) is Predicted mask value (0 or 1) for pixel location (x, y) in the predicted mask.

oΣ is Sigmoid activation function.

14. Complete Loss Function:

The total loss function combines these components, weighted by hyperparameters (λ):

L_total = λ_cls * L_cls + λ_box * L_box + λ_mask * L_mask

oλ_cls, λ_box, λ_mask are Hyperparameters controlling the importance of each loss term.

15. Evaluation Metrics:

To assess model performance, you can use various metrics:

oAverage Precision (AP):

This metric considers both precision (percentage of predicted trash objects that are truly trash) and recall (percentage of actual trash objects that are correctly detected). It's often used as the primary metric for object detection tasks.

oPrecision and Recall:

These metrics are building blocks of mAP. They can be calculated for each class (trash) separately.

Precision = True positive/Total “yes” predictions

Recall = True positive/Actual “yes” predictions

oF1 Score:

This metric combines precision and recall into a single measure:

F1 score = 2 * (Precision * Recall) / (Precision + Recall)

oMean Average Precision (mAP@0.5 IoU):

This variation of mAP focuses on detections with a minimum IoU threshold (often 0.5) between predicted and ground truth bounding boxes.

$$\mathbf{mAP} = \frac{1}{\_}\sum_{=1}^{N} AP$$

whereas,

AP is average precision

IoU=Area of intersection/Area of union

- Segmentation Quality Metrics:

Evaluation of the quality of the predicted segmentation masks for trash instances are done by metrics like Intersection over Union (IoU) or pixel-wise accuracy.

V. RESULTS AND DICUSSIONS

Model performance Metrics:

Throughout training and confirmation, we observed the model's performance on various criteria across periods. This data provides precious perceptivity into our model’s literacy process and its capability to understand the handed information. By assaying these criteria, we can identify areas for enhancement and make adaptations to enrich its overall effectiveness. Also, this information serves as a literal record, allowing us to track patterns and implicit anomalies in the model’s evaluation over time. This can be pivotal for diagnosing problems with the model or the training process itself.

train/box_loss	1.015799
train/cls_loss	1.274201
train/dfl_loss	1.2076
metrics/precision(B)	0.521053
metrics/recall(B)	0.427858
metrics/mAP50(B)	0.422862
metrics/mAP50-95(B)	0.26517
val/box_loss	1.2546
val/cls_loss	1.77543
val/dfl_loss	1.31093

Table.1 Shows the average values of performance metrics of YOLOv8 medium model during the training for detection for10 epochs.

train/box_loss	0.731885
train/seg_loss	1.2694939
train/cls_loss	0.594128
train/dfl_loss	1.008484
metrics/precision(B)	0.694007
metrics/recall(B)	0.5593647
metrics/mAP50(B)	0.5864221
metrics/mAP50-95(B)	0.4034978
metrics/precision(M)	0.6897922
metrics/recall(M)	0.545595
metrics/mAP50(M)	0.569687
metrics/mAP50-95(M)	0.324712
val/box_loss	1.066093
val/seg_loss	2.00374
val/cls_loss	1.314054
val/dfl_loss	1.155961

Table.2 Displaying the average values of performance metrics of YOLOV8 medium model during the training for segmentation for 100 epochs.

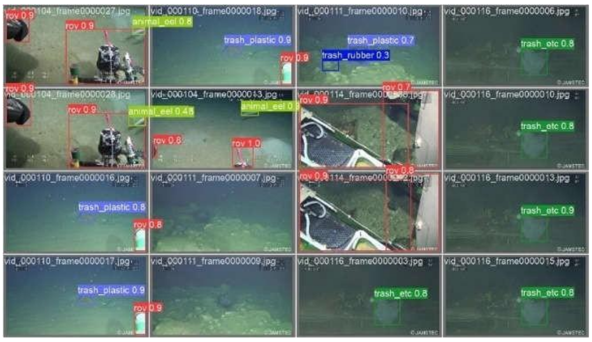
train/box_loss	1.058465
train/cls_loss	1.72381
train/dfl_loss	1.15606
metrics/precision(B)	0.487845
metrics/recall(B)	0.364933
metrics/mAP50(B)	0.357602
metrics/mAP50-95(B)	0.228501
val/box_loss	1.25835
val/cls_loss	1.95589
val/dfl_loss	1.21311

Table.3 Exhibiting the average values of performance metrics of YOLOv8 nano model during the training for detection for10 epochs.

train/box_loss	0.869203
train/seg_loss	1.4971
train/cls_loss	0.8609
train/dfl_loss	1.0342
metrics/precision(B)	0.6516
metrics/recall(B)	0.5076
metrics/mAP50(B)	0.5336
metrics/mAP50-95(B)	0.3558
metrics/precision(M)	0.6503
metrics/recall(M)	0.4934
metrics/mAP50(M)	0.516
metrics/mAP50-95(M)	0.284
val/box_loss	1.1407
val/seg_loss	1.9551
val/cls_loss	1.4605
val/dfl_loss	1.1408

ISSN NO: 2249-3034

Table.4 Representing the average values of



performance metrics of YOLOv8 nano model during the training for segmentation for 100 epochs

Fig.2 showcasing the output images for detection of underwater trash.

It consists of eight sub-images, each a snapshot from a video frame. The algorithm identifies various objects, including a Remotely Operated Vehicle (ROV), marine life, and different categories of trash. Each detected object is enclosed in a bounding box and assigned a confidence score, ranging from 0.3 to 1.0. The ROV, appearing mechanical with attached equipment, is detected with high confidence in the top left frames. Marine life is detected in the top two frames of the second column, while trash, categorized into plastic, rubber, and others, is detected across all frames, except where the ROV is dominant. This analysis demonstrates the potential of such algorithms in environmental monitoring

and cleanup efforts, particularly in identifying and classifying underwater waste.

Fig.3 showcasing the output images for segmentation of underwater trash.

It is a collage of eight frames, each extracted from a video and annotated by the algorithm. The algorithm identifies and segments various objects such as “trash_plastic”, “trash_etc”, “animal_eel”, and “rov”, each enclosed within a bounding box and associated with a confidence score ranging from 0.6 to 1.0. The background of each frame depicts an underwater environment with varying visibility. This analysis underscores the potential of such algorithms in environmental monitoring, particularly in the detection and classification of underwater waste. The algorithm’s ability to distinguish between different types of objects, even in complex underwater environments, is evident.

VI.CONCLUSION AND FUTURE SCOPE

Marine Trash Detection and Segmentation Using Convolution Neural Network is discussed in this article, the escalating issue of marine trash, which poses a significant threat to marine life and indirectly to humans, necessitates effective detection and cleanup strategies. The limitations of human-led efforts underscore the importance of autonomous robot systems in addressing this problem. The model developed in this study leverages Convolutional Neural Networks and YOLO v8 for accurate and efficient detection of marine trash. The addition of labelling and segmentation enhances the model’s performance, particularly in complex underwater environments with low light conditions. The model’s ability to discern trash from other objects and provide dimensional information through segmentation proves crucial. Therefore, the model holds promise for deployment in Autonomous Robot Systems or Autonomous Underwater Vehicles (AUVs), marking a significant advancement in the field. This could revolutionize marine trash detection and cleanup, contributing to the preservation of marine ecosystems. The study thus underscores the transformative potential of integrating advanced technology with environmental conservation efforts.

The future scope of this project is given below

1. The factors such as Stability, accuracy and speed of the model can be improved in terms of Detection by implementing efficient and perfectly trained models coming in future.
2. The use of AI in this project can bring huge advancements like real time classification of trash as the AI is capable of learning from its mistakes in real time and make decisions faster than deep learning.
3. The advancements of this project can be done in terms of proper training by implementing new methods of training the model.

ACKNOWLEDGMENT

The authors express their deepest gratitude to JAMSTEC for their generous provision of the deep-sea debris database, an invaluable asset that significantly enriched the depth and scope of this research. Additionally, the authors extend their heartfelt appreciation to the anonymous reviewers whose meticulous scrutiny and insightful feedback have immensely elevated the quality and impact of this manuscript. Special recognition is also extended to Ultralytics and the esteemed contributors of the BIG VISION blog for their unwavering support and expertise in custom dataset training, which has played an indispensable role in refining and optimizing our model to achieve its full potential.

REFERENCES

[1] H. Yin and C. Cheng. (2010) “Monitoring methods study on the great Pacific Ocean garbage patch,” in Proc. Int. Conf. Manage. Service Sci., Aug. 2010, pp. 1–4. DOI:10.1109/ICMSS.2010.5577101

[2] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 IEEE Conference on Computer Vision and PatternRecognition. doi:10.1109/cvpr.2014.81

[3] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2016.91

[4] M. Valdenegro-Toro, “Submerged marine debris detection with autonomous underwater vehicles,” 2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA). DOI: 10.1109/RAHA.2016.7931907

[5] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN:
Towards real-time object detection with region proposal networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 39, Issue: 6, 01 June 2017). DOI:10.1109/TPAMI.2016.2577031

ISSN NO: 2249-3034

[6] Joseph Redmon, Ali Farhadi, (2018). YOLOv3: An Incremental Improvement. 2018 IEEE conference on Computer Vision and Pattern Recognition (CVPR). doi.org/10.48550/arXiv.1804.02767

[7] Sanae Chiba a b, Hideaki Saito c, Ruth Fletcher b, Takayuki Yogi d, Makino Kayo d, Shin Miyagi d, MoritakaOgido d, Katsunori Fujikura e, "Human footprint in the abyss: 30-year records of deep-sea plastic debris," Mar. Policy, vol. 96, pp. 204–212, Oct. 2018. Doi: 10.1016/j.marpol.2018.03.022

[8] Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J., 2018. Unet++: A nested u-net architecture for medical image segmentation. in: Deep learning in medical image analysis and multimodal learning for clinical decision support, Springer, 3–11, 2018.

[9] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 4510–4520.

[10] Matias Valdenegro-Toro. (2019). Deep Sub-Ensembles for Fast Uncertainty Estimation in Image Classification. Machine Learning (cs. LG); Computer Vision and Pattern Recognition (cs.CV); Machine Learning (stat.ML). doi.org/10.48550/arXiv.1910.08168.

[11] Choi, D., Shallue, C.J., Nado, Z., Lee, J., Maddison, C.J., Dahl, G.E., 2019. On empirical comparisons of optimizers for deep learning, metrics likearXiv preprint arXiv:1910.05446. Bisong, E., 2019. Google Collaboratory, in: Building Machine Learning and Deep Learning Models on Google Cloud Platform, Springer, 59–64, 2019.

[12] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 318–327, Feb. 2020.

[13] Main-Knorn, M., Pflug, B., Louis, J., DeBacker, V., Müller-Wilm, U., Gascon, F., 2017. Sen2Cor for sentinel-2, in: Image and Signal Processing for Remote Sensing XXIII, vol. 10427, International Society for Optics and Photonics, 1042704, 2017. Chasant, M., 2020. Plastic pollution in Ghana: urban trash heroes, https://www. muntaka.com/plastic-pollution-in-ghana/, accessed: 2020-08-10, 2020.

[14] Bing Xue; Baoxiang Huang; Weibo Wei; Ge Chen; Haitao Li; Nan Zhao; Hongfeng Zhang. (2021). An Efficient Deep-Sea Debris Detection Method Using Deep Neural Networks. 2021 IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (Volume: 14). 10.1109/JSTARS.2021.3130238.

[15] M. Kremezi et al., "Pansharpening PRISMA data for marine plastic litter detection using plastic indexes," IEEE Access, vol. 9, pp. 61955–61971, 2021.

[16] Wei Zhou; Fujian Zheng; Gang Yin; Yiran Pang; Jun Yi. (2022). YOLOTrashCan: A Deep Learning Marine Debris Detection Network. IEEE Transactions on Instrumentation and Measurement (Volume: 72). DOI:10.1109/TIM.2022.3225044.

[17] Shailender Kumar; Abhinav Gautam; Khushbu Mehlawat; Kunal Rawat. (2022). Deep Learning Framework for Macro Marine Litter Classification and Quantification. 2022 3rd International Conference for

Emerging Technology (INCET). DOI:10.1109/INCET54531.2022.9825334

[18] Takuya Kiyokawa; Jun Takamatsu; Shigeki Koyanaka. (2022). Challenges for Future Robotic Sorters of Mixed Industrial Waste: A Survey. IEEE Transactions on Automation Science and Engineering (Volume: 21, Issue: 1, January 2024). DOI: 10.1109/TASE.2022.3221969.

[19] Haruna Abdu; Mohd Halim Mohd Noor. (2022). A Survey on Waste Detection and Classification Using Deep Learning. IEEE Access (Volume: 10). DOI: 10.1109/ACCESS.2022.3226682.

[20] Shu-Min Tsai; Ming-Lin Chuang; Jheng-Yong Gao; Cheng-Han Liu. (2023). Application of Beach Trash Detection System Based on Deep Learning. 2023 International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan). DOI: 10.1109/ICCE-Taiwan58799.2023.10226908.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 4510–4520.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015, arXiv:1409.1556.

[23] J. I. Watanabe, Y. Shao, and N. Miura, "Underwater and airborne monitoring of marine ecosystems and debris," J. Appl. Remote Sens., vol. 13, no. 4, 2019, Art. no. 044509.

[24] I. M. J. van den Beld, B. Guillaumont, L. Menot, C. Bayle, S. ArnaudHaond, and J.-F. Bourillet, "Marine litter in submarine canyons of the Bay of Biscay," Deep-Sea Res. II—Topical Stud. Oceanography, vol. 145, pp. 142–152, Nov. 2017.

[25] K. Topouzelis, D. Papageorgiou, A. Karagaitanakis, A. Papakonstantinou, and M. A. Ballesteros, "Plastic litter project 2019: Exploring the detection of floating plastic litter using drones and Sentinel 2 satellite images," in Proc. IEEE Int. Geosci. Remote Sens. Symp., Sep. 2020, pp. 6329–6332.

Bibliography

Mr. K. Sundar Srinivas is currently working as an Associate Professor in the Department of Electronics and Communication Engineering at PSCMR College of Engineering and Technology, Vijayawada, Andhra Pradesh, India. He dedicates his research efforts to the fields of wireless communications and embedded systems.

Mr. R. Balaji is progressing through his final year in the Department of Electronics and Communication engineering at PSCMR College of Engineering and Technology, Vijayawada, Andhra Pradesh, India. He sustains an active membership with IETE. His research focuses on Scientific Theories, Artificial Intelligence, and VLSI.

Mr. V. H. V. Satya Sai is progressing through his final year of Electronics and Communication engineering at PSCMR College of Engineering and Technology, located in Vijayawada, Andhra Pradesh, India. He maintains an active membership with IETE. His research focuses on PLC & AI, VLSI, and the integration of Autonomous Vehicle Technology with PLC capabilities, which he pursues with great interest.

Ms. T. Padmavathi is advancing through her final year in the Department of Electronics and Communication Engineering at PSCMR College of Engineering and Technology, Vijayawada, Andhra Pradesh, India. Her research concentrates on the fields of VLSI and Embedded Systems.

Ms. M. Srilakshmi is advancing through her final year in the Department of Electronics and Communication Engineering at PSCMR College of Engineering and Technology, Vijayawada, Andhra Pradesh, India. She maintains an active membership with the IETE organization. Her research concentrates on the fields of VLSI and Embedded Systems.