**Abstract**

Deep learning methodologies have been extensively applied in a substantial portion of ongoing research within the realm of remote sensing. The release of the benchmark results containing open-source dataset Marine Debris Archive (MARIDA), created new opportunities for the application of deep learning algorithms for the problem of debris identification and segmentation in marine environments. The attention-based segmentation method shown in this paper performs better than the state-of-the-art outcomes achieved with MARIDA. The Original study introduces a brand-new spatially aware encoder and decoder architecture that preserves the images' sparse ground truth patches' structure and contextual information. Keeping the idea from the original work, this report discusses the usage of different machine learning data processing and modelling algorithms. It is anticipated that the obtained results will open new avenues for deep learning research with remote sensing imagery. You may get the code at

**Introduction**

Marine debris is a growing concern for various reasons, including environmental, economic, and human health considerations [Fig01]. Plastic waste poses a significant problem as it can remains floating in the ocean for extended periods and has been discovered in numerous locations worldwide. To address this issue, a range of methods has been developed to detect marine debris, leveraging both open-source and commercial data sources such as satellite imagery and autonomous underwater vehicles.

A map of the world with cities

Description automatically generated

Fig01 - Red spots indicate the presence of Marine Debris around the world.

Earth observation data from public and commercial satellite programs play a crucial role in identifying debris sites. To better understand the spectral characteristics of marine debris, researchers have utilized indices like the Floating Debris Index (FDI) and Plastic Index (PI). However, distinguishing floating debris from other bright features like waves, sunlight, clouds, ships, and foam presents challenges due to the complex properties of plastics, including variations in colour, chemical composition, size, and submergence level in the water. Previous datasets primarily focused on detecting objects such as vessels, clouds over sea areas, and macroalgae. To address the existing challenges in this field, the Marine Debris Archive (MARIDA) has been introduced as an open-access benchmark dataset. MARIDA contains images extracted using Sentinel-2 multispectral satellite data, aiming to provide a valuable resource for further research and development in the detection and monitoring of marine debris.

**EU AI Regulation**

The EU AI Regulation, proposed by the European Commission, sets out rules for AI use in the EU. In the Marida project, this means assessing the ML system's risks to users and the environment, ensuring clear explanations of how it detects marine debris, complying with data privacy laws like GDPR, addressing biases for fair detection, and holding the project team accountable for system performance. Overall, the regulation will shape how risks, transparency, ethics, and compliance are managed in the Marida Marine Debris ML project.

**Requirement and Data Analysis**

The Marine Debris ML Project is crucial for protecting our oceans and promoting human safety. By using machine learning, it helps identify and monitor marine debris, particularly plastics, which threaten marine life and habitats. This technology enables targeted clean-up efforts and reduces the risk of accidents for ships navigating through debris-filled waters. Moreover, it provides valuable data on plastic pollution trends, empowering policymakers, and environmental organizations to make informed decisions and allocate resources efficiently. The project also highlights the potential of artificial intelligence in environmental conservation, encouraging further research and innovation in sustainable technologies. In this project, we required to go through the existing work. Perform data visualization, apply some machine learning algorithms to enhance the result. Idea is to understand the basic principles of designing ML and Deep learning Models.

**Marida – Dataset**

MARIDA holds a collection of 1381 patches saved as geo-tif files, each accompanied by its pixel-wise segmentation masks and confidence scores stored in a JSON file. These patches were gathered from 63 scenes captured by the Sentinel S2 Satellite mission and cover study sites across 11 countries, including Honduras, Guatemala, Haiti, and others.

The dataset, presented in WGS'84/UTM projection, comprises 15 different classes, with details on their distribution provided in [Fig02]. The table also points out the need for better representation of certain classes, a challenge addressed through Weighted Cross Entropy loss, explained in an upcoming section. Moreover, the dataset allows for a simpler approach to marine debris detection by converting the problem into a binary classification task, distinguishing between plastic and non-plastic classes. This approach streamlines the process and yields better results compared to the baseline results outlined in MARIDA.

A table of numbers and symbols

Description automatically generated

Fig01 – Table contain all the 15 classes.

**Data Processing and Visualization**

1. **Data loading with Labels**

MARIDA Dataset provide collection of labelled images. Code [fig03] takes us through this collection one by one. For each image, it tells us the filename and its associated label.

A screen shot of a computer

Description automatically generated

Fig03 – Output shows the filename and respective.

1. **Loading some images**

Fig04 – is the output of random selection and visualization of images stored in a collection. Code sets up a layout for displaying 18 images in a grid format with 6 columns and 3 rows. By randomly sampling images from the collection, it ensures a diverse selection for viewing. Each selected image is processed and displayed within its designated subplot, with the image name serving as its title. The code utilizes NumPy to handle image data, particularly for grayscale images where only the first colour channel is considered. Upon execution, the code generates a plot showcasing the randomly selected images, offering a convenient way to explore and examine image content.

A collage of different images

Description automatically generated

Fig04 – Output from code section 11 in the provide project file.

1. **Loading random images with respective Histogram**

Here, we subplot grid with a specified number of rows and columns to display images and their corresponding histograms. It selects a random sample of images from a collection and iterates over them. For each image, it converts the image data into a NumPy array, ensuring compatibility for visualization. If the image has multiple colour channels, it selects the first one. Then, it extracts the image name from its file path and calculates the row and column indices for the current subplot. Within each subplot, it displays the image using a grayscale colormap and adds a title. Additionally, it creates a histogram to show the distribution of pixel intensities within the image. Finally, it adjusts the spacing between subplots for better presentation and displays the plot containing both images and histograms. This code provides a convenient way to visualize images along with their pixel intensity distributions for analysis.

A collage of images of a variety of objects

Description automatically generated

Fig05 – Output from code section 12 in the provide project file.

1. **Distribution of images with respect to different classes**

Below visualizes the distribution of images across different classes using a vertical bar plot created with the Seaborn library. First, it sets the Seaborn style to "whitegrid" for a clean appearance. Then, it sorts the counts of images for each class in descending order to ensure clarity in the plot. Using this sorted data, it creates a DataFrame containing the class names and the corresponding number of images. The code then generates a vertical bar plot where each bar represents a class, and its height indicates the number of images belonging to that class. Additionally, annotations displaying the exact number of images are added to each bar for clarity. Finally, labels and a title are added to the plot to provide context, and adjustments are made for better readability. Overall, this visualization offers insights into the distribution of images across different classes in the dataset.

A graph with numbers and a bar

Description automatically generated with medium confidence

Fig06 – Output section 24 from the provided project file.

**System Engineering**

A diagram of a flowchart

Description automatically generatedBelow is the simple block diagram which is explaining that the images from the Marida Dataset will be processed, and the output will be a classified image or model accuracy in case of this project scope.

**Algorithm Design**

Based on the scope of this project and researched, this report used RandomForestClassifier and KNeighborsClassifier Algorithms. We will go through in details for these separately.

1. **Random Forest**

**Motivation behind using RandomForestClassifier**

Random Forest is a popular choice for image classification due to its versatility and effectiveness. One key advantage is its ability to handle high-dimensional data, making it suitable for processing the vast amount of information present in images. By employing an ensemble of decision trees, Random Forest combines multiple classifiers to make more accurate predictions. Moreover, it can automatically assess the importance of different image features, enabling it to effectively distinguish between classes based on relevant characteristics. Another strength lies in its ability to capture non-linear relationships within image data, providing flexibility in modelling complex patterns. Random Forest also tends to be less prone to overfitting, particularly when trained with many trees and appropriately tuned hyperparameters. Additionally, its built-in out-of-bag evaluation allows for unbiased performance estimation without the need for a separate validation set. These factors, coupled with its ease of implementation and minimal tuning requirements, make Random Forest a compelling choice for image classification tasks across various domains.

Resources

[Artificial Intelligence Act: MEPs adopt landmark law | News | European Parliament (europa.eu)](https://www.europarl.europa.eu/news/en/press-room/20240308IPR19015/artificial-intelligence-act-meps-adopt-landmark-law)

[21\_ofwg\_wksp\_002.pdf (apec.org)](https://mddb.apec.org/Documents/2021/OFWG/WKSP/21_ofwg_wksp_002.pdf)