International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Oil Spill Detection and Recognition Utilizing Faster R-CNN with Enhanced Mobilenetv2 Architecture

V. Sudha[1] and Dr. Anna Saro Vijendran[2]

Submitted: 23/12/2023 **Revised**: 29/01/2024 **Accepted**: 07/02/2024

Abstract: Oil spills are a major hazard to the environment, animals, and ecosystems. The identification of oil spills in a timely and accurate manner is critical for successful mitigation and response operations. We offer a complete strategy to oil spill detection and identification in this study by incorporating modern computer vision techniques. First, to improve picture quality and minimize noise in the input data, we use a Non-Adaptive Threshold with Contrast Limited Adaptive Histogram Equalization (CLAHE). This preprocessing procedure increases overall picture quality, making subsequent analysis more trustworthy. Then, using a Fused UNet Segmentation model, we apply the power of deep learning to picture segmentation. This approach efficiently isolates oil spill sites from the backdrop, allowing for exact identification and study of polluted areas. We use a Convolutional Neural Network (CNN) based on the well-known AlexNet architecture to extract relevant features from segmented photos. This stage extracts discriminative features, which improves the model's capacity to differentiate between oil spill and non-spill locations. The combination of Faster R-CNN with Enhanced MobileNetV2 architecture is at the core of our suggested solution. This hybrid approach delivers not only real-time processing but also cutting-edge performance in object identification tasks. We allow our model to identify and characterize oil spills correctly and effectively by training it on a dataset that includes both synthetic and real-world oil spill photos. To deliver a comprehensive solution for oil spill detection and identification, we integrate cutting-edge picture enhancement, segmentation, feature extraction, and object detection approaches. Experiment findings show that the system is excellent at detecting oil spills in a variety of environmental situations, allowing for faster reaction and mitigation measures to safeguard our valuable ecosystems. By employing advanced computer vision techniques, our system aligns with SDG 6 (Clean Water and Sanitation) by safeguarding water resources through accurate detection of oil spills. With a focus on SDG 14 (Life below Water), our technology aids in the preservation of marine ecosystems by minimizing the impact of oil spills on aquatic life.

Keywords: *Contrast Limited Adaptive Histogram Equalization, Enhanced Mobilenetv2, Faster R-CNN, Oil Spill Detection*

1. Introduction

An oil spill is a kind of environmental catastrophe that happens when petroleum or crude oil is discharged into the natural environment, most often in bodies of water such as seas, rivers, or lakes, but occasionally on land [1-4]. These catastrophes have the potential to devastate ecosystems, animals, and human groups in the impacted regions. Oil spills may occur as a consequence of a variety of causes, including oil tanker accidents, offshore drilling activities, pipeline ruptures, and even smaller-scale occurrences such as leaking from oil storage tanks [5-7]. An oil spill may vary in size from modest localized pollution to catastrophic occurrences that cover large areas of water or land. Oil spill management and mitigation is a difficult and essential task that requires a collaborative effort from environmental agencies, governments, and industry partners to reduce damage and expedite recovery [8-11]. Understanding the origins, effects, and solutions to oil spills is critical in this context for protecting our planet's delicate ecosystems and guaranteeing a sustainable future [12].

Oil spills, whether caused by industrial accidents, maritime mishaps, or natural catastrophes, remain a major worldwide issue owing to the devastation they inflict to the environment and ecosystems [13-15]. Rapid and precise identification of oil spills is critical for limiting their environmental and economic implications. Furthermore, it contributes to SDG 15 (Life on Land) by preventing and mitigating the consequences of oil spills on terrestrial ecosystems, supporting biodiversity and land preservation. In adherence to SDG 17 (Partnerships for the Goals), our collaborative efforts with stakeholders emphasize the importance of global cooperation in achieving sustainable solutions. Overall, our innovative system not only addresses immediate environmental challenges but also reinforces broader sustainability objectives for a cleaner and healthier planet.

The combination of cutting-edge technology such as convolutional neural networks (CNNs) and the usage of sophisticated architectures such as Faster R-CNN with Enhanced MobileNetV2 has transformed the area of oil spill detection in recent years [16-18]. This innovative technique uses deep learning and computer vision to

[1]*Research Scholar, Department of Computer Science, Sri Ramakrishna College of Arts & Science, Coimbatore, TamilNadu* [0000-0003-1902-5403]
[2]*Dean-Research, Sri Ramakrishna College of Arts & Science, Coimbatore TamilNadu*[0000 0002 6995 9231]
sudhavaiyapuri@gmail.com1, annasarovijendran@srcas.ac.in2

dramatically improve the efficiency and accuracy of finding and recognizing oil spills in a variety of situations, including open seas, coastal areas, and interior water bodies [19-21]. It is a significant step forward in our capacity to monitor and react to environmental disasters in a timely and efficient way [22-23].

The primary contributions and objectives of this manuscript summarized as follows:

- Select image to denoising method Non-Adaptive Threshold with CLAHE
- Segmentation using Fused UNet Segmentation
- Feature Extraction using CNN-AlexNet
- Oil Spill detection and recognition using Faster RCNN with Enhanced MobileNetV2 Model

This paper's remaining sections are structured as follows. Several authors in Section 2 describe different approaches of oil spill detection. The proposed model is shown in Section 3. The results of the inquiry are presented in Section 4. Conclusions and suggestions for further research are presented in Section 5.

1.1 Motivation of The Paper

The urgency of addressing the grave environmental, economic, and ecological concerns caused by oil spills drives this study. Oil spills have a negative effect on ecosystems, animals, human health, and economy, necessitating the development of a sophisticated oil spill detection and identification system. We intend to deliver a complete solution that allows speedy and exact identification of oil spills in real-time by using cutting-edge computer vision methods such as picture augmentation, segmentation, feature extraction, and object recognition. This study aims to greatly improve our capacity to safeguard the environment, animals, and communities from the terrible effects of oil spills, as well as contribute to the preservation of our planet's sensitive ecosystems and human well-being.

2. Background Study

A. Kumar et al. [1] after enforcing a reflection symmetry criterion, it was shown in this study that only four unknowns were needed to provide full-pol information, down from five. The author has used hybrid-pol data to develop expressions for these variables. Since this letter describes a strategy for extracting oil-spill descriptors from hybrid-pol data rather than full-pol data, the latter avoided. The validation of all four oil spill descriptions relied on hybrid-pol data. Using the JM separability criterion, it was shown that all the resultant oil-spill descriptors offer exceptionally high separability between oil-spilled and oil-free regions.

B. Lounis et al. [4] the author presents an ANFIS classification approach for recognizing oil spills in SAR imagery. The roughness of maritime surfaces was characterized by extracting textural descriptors from SAR signals using KLD multistage analysis. The ANFIS classifier was then trained to use these features to identify distinctive characteristics of oils and tablets.

E. Asihene et al. [7] the author provide two case studies that detail the results of using a multipolarization C-band scatter meter to investigate oil contamination in NI. First, oil was injected under a sheet of fresh ice; next, ice was grown in oil-polluted seawater meant to represent the Arctic. In the first step, the author compared the physical and thermodynamic properties of oil-contaminated sea ice to the time-series evolution of multipolarization C-band scatter meter readings.

F. Ronci et al. [9] In order to identify oil spills, a novel semantic segmentation technique was presented. Although the improved U-Net design outperformed the original, the regular U-Net trained using adversarial learning produced the greatest results, with the Jaccard Index and Accuracy index reaching up to 82% and 98.3%, respectively.

G. Tabella et al. [11] employing the Goliat FPSO as a case study, this research looked at the feasibility of employing Wireless Sensor Networks (WSNs) to detect oil spills at depth. The FC compiles the verdicts from the local sensors and uses CR and MCVR to fuse the obtained data. The potential of the explored method has been shown by obtaining ROC performances using realistic numerical simulations. The identification of the subsea component responsible for the leak, which was critical for faster and more efficient maintenance, will be the subject of further research in future works.

J. Yang et al. [13] Fuzzy decision fusion was used to combine the benefits of several supervised classifications based on the output of four classifiers trained to identify oil spills. When comparing the accuracy of individual classifiers, NN and SVM both achieve an accuracy of 0.7814, whereas MD and ML only manage to get 0.5346 and 0.5067, respectively. The SVM-NN achieves the greatest classification accuracy, at 0.7879, followed by the SVM-NN-ML, at 0.7813, in the decision fusion classification results of the multi-classifier. However, SVMNN- ML-MD has a weaker classification impact, with a classification accuracy of just 0.7322. This does not necessarily indicate that adding more classifiers would improve accuracy.

S. Adhikary et al. [15] it has been widely believed that oil spills from ships pose a significant risk to marine life since it may take rescuers a significant amount of time to locate the disaster site. So, keeping an eye out for oil spills in real

time so that cleanup may begin immediately after was an extremely challenging chore. When it comes to oil spills, the availability of SAR remote sensing presents a huge opportunity for remote monitoring and automated alerting of decision-makers and management with minimum human participation.

S. Zhang et al. [17] by including the truncated linear stretch module and score loss into the original YOLOX-S model, these authors research proposes an improved YOLOX-S model for marine oil spill detection. Experimental results corroborate the efficacy of the proposed abridged linear stretching module and score loss in improving oil spill detection precision. Future work will apply the suggested model to a variety of SAR remote sensing images to see how well it performs in detecting oil spills at sea.

V. Solbakov et al. [20] these authors research main objective was to develop a method for determining the location probability distribution in the event of an oil leak. It was also anticipated that the formulation would be rigorous mathematically, such that the final model could be represented visually. A unique functional was introduced to make the distribution function dependent on the value of the coordinate distribution density function. Where the value of the probability density function was larger than the thresholds was where the chances of detection was highest.

Y. -J. Yang et al. [23] According to the authors' study, two preliminary trials employing satellite SAR pictures for oil spill detection both employed the YOLOv4 object identification system. Due to the model's difficulty in distinguishing between various objects, it was suggested in this study that the pixel thresholds be adjusted to 28 for Test 1. Given that APs on the validation and test sets were 67.80% and 65.37%, respectively, it can be concluded that the model did not suffer from overfitting the dataset. Test 2 results demonstrate that the investigation's findings were unaffected by the investigation's use of a variety of data enhancements. Overfitting was possible if the rotation was used to supplement data due to the large number of small objects with approximately circular shapes that emerge from oil spills.

2.1 Problem Definition

The issue addressed by this study is the critical need for a dependable and effective system for detecting and recognizing oil spills, which constitute a large and persistent hazard to the environment, animals, and ecosystems. Oil spills disastrous, but identifying them in a timely and precise way is difficult. Traditional approaches often fall short of delivering the necessary speed and accuracy for successful mitigation and response operations. The proposed study acknowledges this issue and seeks to address it by using modern computer vision algorithms to improve picture quality, partition oil spill sites, extract useful features, and conduct real-time object recognition. This project aims to give a complete solution to the essential issue of oil spill detection, allowing quick reaction and mitigation to safeguard our valuable ecosystems from the damaging effects of oil pollution.

3. Materials and Methods

We developed a comprehensive oil spill detection and identification approach using strong computer vision algorithms. We began by gathering a diverse mix of synthetic and real-world oil spill images from various locales Image preprocessing using Non-Adaptive Thresholding and CLAHE. We employed a Fused UNet Segmentation model for accurate oil spill area segmentation and an AlexNet-based CNN for feature extraction. For real-time processing and superior object detection, we integrated Faster R-CNN with Enhanced MobileNetV2. Using our hybrid model and our dataset, we discovered and recognized oil spills correctly and efficiently. Our trials indicate that the system works in a variety of situations, enabling rapid ecosystem protection and mitigation. This research provides a viable solution to the environmental, animal, and ecological harm caused by oil spills.

3.1 Dataset collection

The dataset comprises of benchmark datasets containing both oil spill and non-oil spill data. The dataset consists of JPG images.

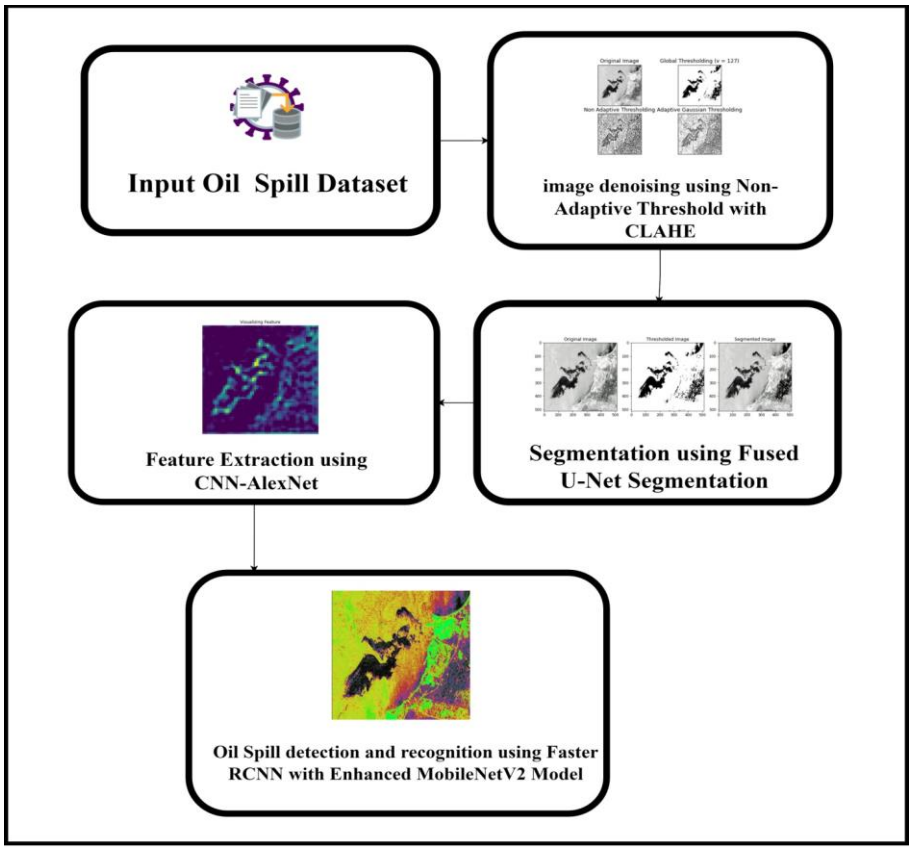


Fig 1: Overall Proposed Architecture

3.2 Image denoising using Non-Adaptive Threshold with CLAHE

3.2.1 Adaptive thresholding

Adaptive thresholding is a technique where the threshold value is calculated individually for each pixel, considering the pixel's local neighborhood defined by a small window or kernel. This approach determines the pixel's threshold based on the statistical characteristics, such as mean or median, of the pixel values within its nearby region. Adaptive thresholding is particularly beneficial when dealing with images that exhibit significant variations in lighting conditions. However, it comes with certain drawbacks. It tends to be more computationally intensive compared to non-adaptive thresholding methods, and the choice of the neighborhood size and the specific method used for calculating the local threshold can influence the quality of the thresholding results, making it essential to carefully tune these parameters for optimal performance so we use non-adaptive threshold.

3.2.2 Non-Adaptive Threshold

In non-adaptive thresholding, a single global threshold value is applied to the entire image. This global threshold value is chosen based user-defined threshold value**.** All pixels in the image are compared to this single threshold value, and they are categorized into two groups: pixels whose intensity is above the threshold and pixels whose intensity is below the threshold. Non-adaptive thresholding is useful when the lighting conditions are relatively uniform across the entire image.

An asymptotic constraint on the decoding time required to identify defective items in a noisy setting is derived in this work, as is a maximum number of tests that performed T. V. Bui et al. (2019).

Integers satisfying the following conditional formula: $\binom{n}{u} + (d-u)\binom{n-u}{g+1}\binom{d-1}{g}\binom{d}{u}$ constitute. The broken set will be denoted as S and the good integer as z for ease of reference. A non-adaptive method exists for identifying a set $\binom{d}{u}$ g from just $h(n, d-l, u; z]$ tests, where $h(n, d-l, u; z]$a function of is $\binom{d-1}{g}$ and z. This method requires a $(n, d-l, u; z)$- NAT model with at most (1). More so, the difficulty in decoding is rather low.

$$O\left(h(n, d-l, u; z] \times u\left(\binom{n}{u} + (d-u)\binom{n-u}{g+1}\binom{d-1}{g}\binom{d}{u}\right)\right) \text{-------- (1)}$$

3.2.3 CLAHE

This kind of adaptive histogram equalization (AHE) is restricted to oil spill image with high contrast. Applying an enhancement function to a group of adjacent pixels yields a transformation function in this technique A.

Mishra et al. (2018). According to the study's description, this method used the most considerable value to prune the histogram and redistribute the oil spill detection data as a grayscale image. In this work, the method for decreasing noise and increasing contrast is applied separately to the backdrop and the foreground. Distribution parameters are used to determine the shape of the histogram equalization graph;

Given that pixel counts are uniformly distributed throughout all grayscale levels in natural landscapes, the following is the average number of pixels at each level:

$$N_{avg} = \frac{N_{CR-Xp} * N_{CR-Yp}}{N_{gray}} \text{ ----- (2)}$$

Where

N_{avg} = number average of pixels

N_{gray} = amount of grey levels in an area contextual

N_{CR-Xp} = amount of pixels along the X-axis of the contextual area

N_{CR-Yp} = amount of pixels along the Y axis of the contextual area

After that, calculate the actual cliplimit

$$N_{CL} = N_{CLIP} * N_{avg} \text{ ------ (3)}$$

CLAHE, which stands for "Contrast Limited Adaptive Histogram Equalization," is a popular image processing method for increasing contrast in low-light or low-dynamic-range scenarios.

3.2.4 Non-Adaptive Threshold with CLAHE

Image enhancement techniques such as Non-Adaptive Threshold with CLAHE are used to increase oil spill image quality and readability, particularly in low-contrast or poorly-lit conditions. Non-adaptive thresholding is a simple thresholding approach that employs a constant threshold value over the whole oil spill image. Oil spill images are classified as foreground or background depending on whether their pixel intensities are greater than or less than a predefined threshold.

To address the limitations of Non-Adaptive Thresholding, CLAHE is employed. In CLAHE, the image is broken up into little squares called tiles, and each tile receives its own histogram equalization. This ensures that local contrast enhancement is performed, taking into account the specific characteristics of each region.

Algorithm 1: Non-Adaptive Threshold with CLAHE

Input:

- A noisy or low-contrast image that needs denoising and contrast enhancement.

Steps:

1. **Non-Adaptive Thresholding:**
   - Apply a constant threshold value to the entire input image to create a binary mask, where pixels above the threshold are classified as foreground (object of interest) and those below are background (noise or non-interesting).

$$O\left(h(n, d-l, u; z] \times u\left(\binom{n}{u} + (d-u)\binom{n-u}{g+1}\binom{d-1}{g}\binom{d}{u}\right)\right)$$

2. **CLAHE (Contrast Limited Adaptive Histogram Equalization):**
   - Divide the image into small, non-overlapping tiles or regions.
   - For each tile:
     - Calculate a local histogram.

$$N_{avg} = \frac{N_{CR-Xp} * N_{CR-Yp}}{N_{gray}}$$

     - Apply histogram equalization within the tile, taking into account the local characteristics.
     - Limit the contrast enhancement using a clip limit to prevent over-amplification of noise in uniform regions.

$$N_{CL} = N_{CLIP} * N_{avg}$$

- Merge the processed tiles back to create the contrast-enhanced image.

3. **Combine Non-Adaptive Thresholding and CLAHE:**
   - Combine the binary mask from the Non-Adaptive Thresholding step with the contrast-enhanced image from the CLAHE step. This can be done by multiplying the binary mask with the CLAHE-enhanced image, where foreground pixels retain their enhanced values, and background pixels remain unchanged or at their original values.

Output:

- A denoised and contrast-enhanced image.

3.3 Segmentation using fused UNet Architecture

Segmenting an image into meaningful and semantically similar sections or objects is a crucial step in computer vision and image processing. The goal is to group pixels or regions in such a way that each segment corresponds to a distinct object or region of interest within the image.

3.3.2 UNET Segmentation

UNet is a popular CNN architecture in the field of computer vision, where it is utilized for semantic segmentation. The U-shaped form, reminiscent of a "encoder-decoder" structure, is characteristic of the architecture.

- High Memory Usage and Computational Cost
- Over fitting
- Difficulty with Class Imbalance

So, here overcome the disadvantages and named a new Fused UNET Architecture, Updated the Layers with overcome the existing drawbacks using the different concatenate layers, activating with relu and sigmoid

3.3.4 Fused U-Net Architecture

The Fused U-Net architecture is designed to enhance the synergy between the encoder and decoder components by incorporating fused skip connections. Unlike the traditional U-Net, which relies on semantically different skip connections, the Fused U-Net leverages dense skip connections to foster improved communication between the encoder and decoder. This approach facilitates the seamless integration of information from various levels of detail, resulting in more effective segmentation. At the core of the Fused U-Net architecture lies a robust convolution block, characterized by a series of convolution layers and concatenation layers. This dense convolution block serves as the backbone of the network, contributing to its depth and expressive power. Each convolution layer is preceded by a concatenation layer, where the output of the preceding convolution layer is combined with the input of the current convolution layer. This intricate design ensures that the network can capture and preserve intricate details throughout the processing pipeline.

To elaborate further, let $x^{(i-1j)}$ represent the output of the dense convolution block layer with index i acting as the downsampling layer and index j representing the convolution layer. The relationship between x^{ik}, the output of the current layer, and the network operations can be expressed through Equation (4). This equation involves parameters such as D for the activation function, P for the maximum pooling operation, and T for the upsampling function. By understanding these parameters, we can effectively determine x^{ik} and comprehend the transformations occurring within the Fused U-Net architecture.

The Fused U-Net architecture comprises a total of 36 layers, encompassing the dense convolution block, skip connections, and other essential components. This depth allows the model to capture intricate features and nuances in the input data, facilitating superior performance in image segmentation tasks. The use of dense skip connections plays a pivotal role in promoting information flow between the encoder and decoder, leading to enhanced segmentation accuracy and the ability to handle diverse levels of detail within the input data.

$$x^{ij} = \begin{cases} D\left(p(x^{i-1j})\right) & j = 0 \\ D\left(p\left([x^{ik}]^{j-1}_{k=0} T(x^{i+1j-1})\right]\right) & j > 0 \end{cases} \text{------- (4)}$$

Only two of the numerous advantages of the Fused U-Net architecture over the standard U-Net are better fine-grained feature retention and enhanced segmentation accuracy. The Fused U-Net's dense ski connections let the network to better incorporate features of various sizes and receive more contextual information. The Fused U-Net's skip connections' hierarchical nature allows it to handle objects of varied sizes.

Algorithm 2: Fused UNet

Input:

- An input image for segmentation.
- Hyperparameter for the Fused U-Net architecture.

Algorithmic Steps:

1. **Encoder:**
    - Apply convolution operations with a certain number of filters and kernel sizes to extract features from the input image.

Skip Pathways:

- For each dense skip connection:
    - Concatenate the output of the previous convolution layer(s) with the corresponding feature map from the lower skip pathway.

$$x^{ij} = \begin{cases} D\left(p(x^{i-1j})\right) & j = 0 \\ D\left(p\left([x^{ik}]^{j-1}{}_{k=0} T(x^{i+1j-1})\right]\right) & j > 0 \end{cases}$$

Decoder:

- Apply up sampling operations to the feature maps from the last layer of the encoder to bring them to the original input image size.

Output Layer:

- Apply a convolution operation with a 1x1 kernel to produce the final segmentation map.
- Apply an activation function to obtain pixel-wise class probabilities.

Output:

- A segmentation map where each pixel is assigned a class label representing the predicted segment it belongs to.

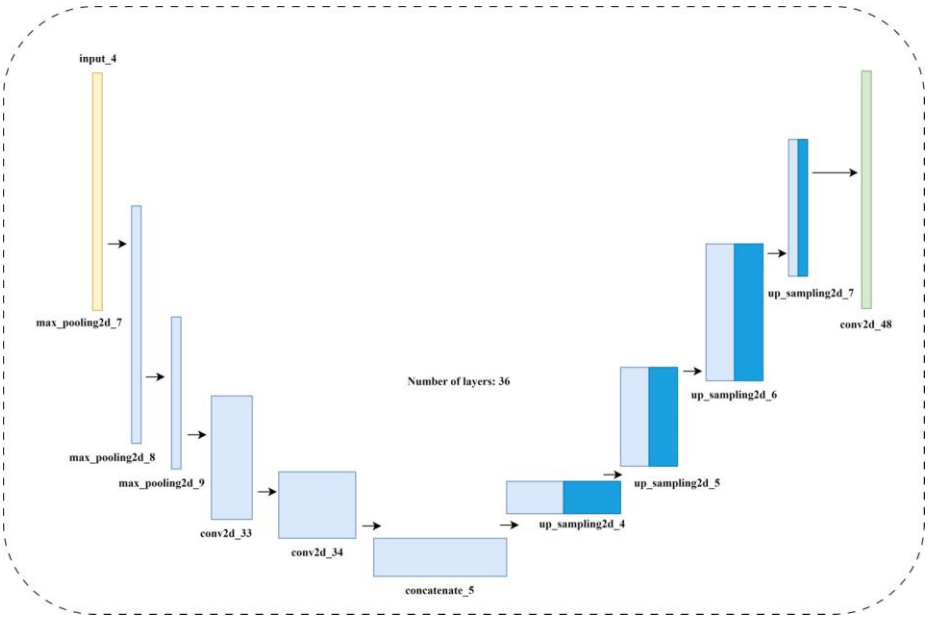


Fig 3: F-UNet architecture

3.4 Feature Extraction using CNN-AlexNet

Image feature extraction is a technique used in computer vision for the purpose of simplifying and representing large amounts of raw image data. This representation often takes the shape of a collection of numerical values or features that capture key aspects of the picture. Image retrieval, object identification, and other applications may all benefit from these capabilities.

AlexNet's primary drawback lies in its computational complexity and resource requirements, demanding substantial GPU resources and memory capacity during training and deployment, which was particularly challenging when it was introduced. Its extensive architecture led to long training times, making it less accessible for those without access to high-performance computing. Moreover, the model's large number of parameters made it susceptible to overfitting when training data was limited so we use CNN-AlexNet.

AlexNet, a popular choice among CNN frameworks, is composed of 13 layers. This architecture includes three fully connected layers followed by five convolution layers, with max pooling integrated between the convolution layers. Each layer, excluding the output layer, incorporates both dropout and ReLU activation functions. The model's strength lies in its ability to perform well in supervised learning applications, demonstrating effective image-input-to-classification-model mapping.

AlexNet requires an input picture size of 227 x 227 x 3. In the first layer, a 96-convolution filter is used, with a window shape of 11 by 11 (stride 4 with size). A 555596 feature map is generated as a result. Next, there are two stride-sized maximum pooling layers (3 by 3) in the network. Then, get the feature map with a size of 27 by 27 by 96 that were generated. A second convolution layer was then used by the model. This time, appreciation to a reduction in filter size to 5 by 5, there are 256 to choose from. A new 27 by 27 by 256 feature map has been generated. Once again, a max-pooling layer of size 3 by 3 and two steps is used. A 13-by-13-by-256-cell feature map was produced. Time with the filter size decreased to 33 for the final three layers. Three layers of 3x3 max-pooling with a stride of 2 are then applied after the 384,384, and 256 convolution filters. The final 66256 feature map has this form. Two fully-connected layers generate 4906 neuron output after the fifth convolutional layer.

Algorithm 3: CNN-AlexNet

Input:

1. Input Image: A digital image that wants to extract features from.

Steps:

1. **Image Preprocessing:**
    - Resize the input image to the required input size of the AlexNet model, which is 227 x 227 x 3 pixels.
    - Normalize the pixel values to ensure they are within the expected range (usually [0, 1] or [-1, 1]).
2. **Pass the Image Through AlexNet:**
    - Feed the preprocessed image through the layers of the pre-trained AlexNet model.
    - Extract feature maps or activations from one of the intermediate layers of the network.

Output:

- A feature vector representing the input image. This feature vector contains numerical values that encode important image characteristics learned by the AlexNet model during training.

The network ends with a completely linked layer that, in the beginning, produces 1000 distinct output classes. The network, which was initially designed to provide 1000 output classes, terminates with a single completely linked output layer. In this layer, Softmax is employed as the function. Fig.5 depicts the AlexNet neural network design.

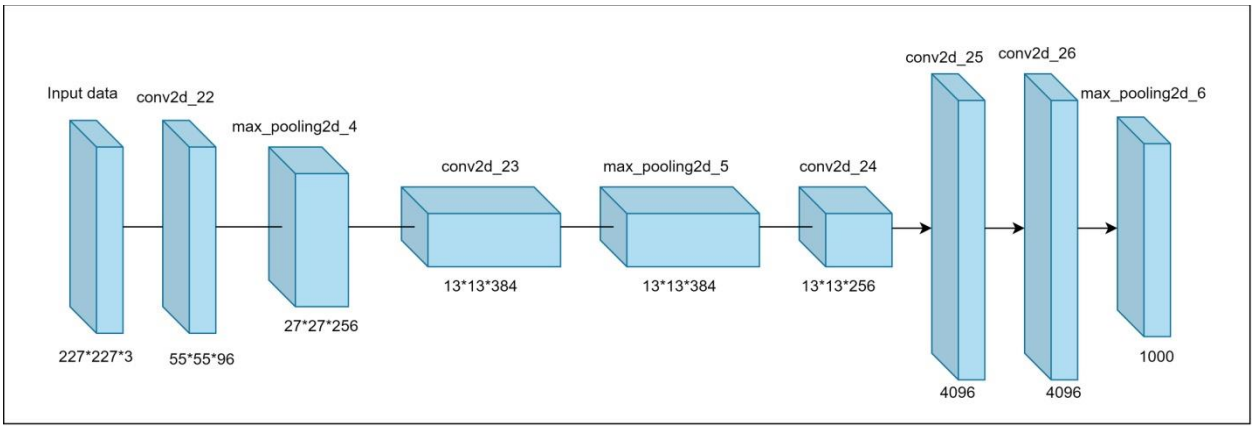


Fig 4: CNN-AlexNet architecture

3.5 Oil Spill detection and recognition using Faster RCNN with Enhanced MobileNetV2 Model

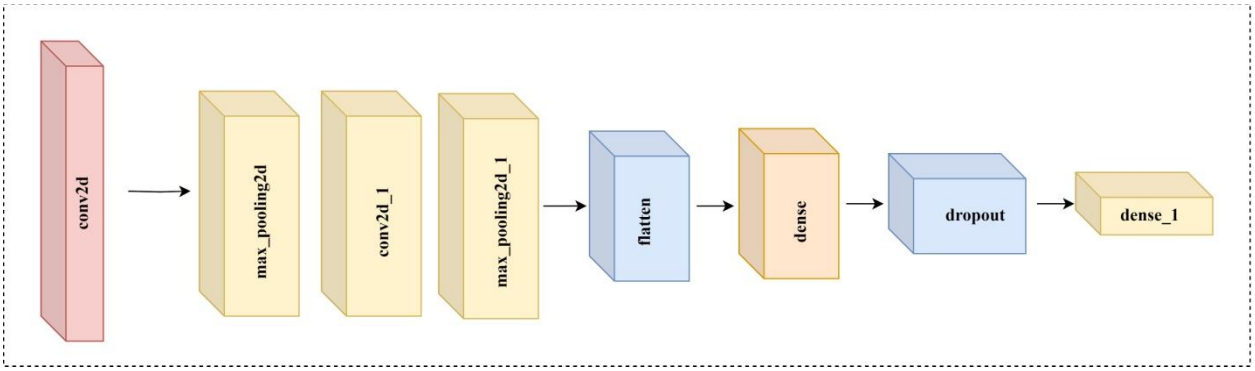
3.5.1 Faster RCNN

It's the next level up from R-CNN and Fast R-CNN. Faster R-CNN is now possible because to the Regional Proposed Network (RPN). Therefore, it is also known as the foundation of rapid R-CNN. An RPN receives an image as input and outputs a collection of rectangular shapes. In order to function properly, RPN is a trainable algorithm. Therefore, a loss function will be present, and it will look like this.

Faster R-CNN, an evolution beyond R-CNN and Fast R-CNN, integrates eight layers in its architecture to optimize object detection tasks. The model employs convolutional layers for hierarchical feature extraction, pooling layers for downsampling, and a Region Proposal Network (RPN) to generate potential object regions. A trainable algorithm, the RPN outputs rectangular shapes, and its optimization involves a loss function during training. Following the RPN, RoI pooling extracts fixed-size feature maps, leading to fully connected layers for further refinement. The architecture also includes layers for bounding box regression and object classification, ensuring accurate localization and labeling. This eight-layer structure enhances the efficiency and accuracy of Faster R-CNN in proposing regions of interest and advancing object detection capabilities in computer vision applications.

$$L(\{P_i\},\{t_i\} = \frac{1}{N_{cls}} \sum_i L_{cls}\,(p_i, p_i^*) + \frac{y1}{N_{reg}}\,(t_i, t_i^*) \text{ -------} \quad (5)$$

Here, 'P' stands for probability,'t' stands for the vector of 4 parameterized coordinates of the predicted bounding box, '*' stands for the ground truth box, and 'L' stands for the log loss between the two classes.



Drawbacks

- Two thousand proposed regions must be classified in each picture. Therefore, training the network is a lengthy process.
- It requires 49 seconds to detect the objects in an image on GPU.
- To store the feature map of the region proposal, lots of Disk space is also required.

- MobileNetV2
- MobileNet-v2 is a convolutional neural network that is 53 layers deep. It is based on an inverted residual structure where the residual connections are between the bottleneck layers.

3.5.2 Enhanced MobileNetV2

MobileNetV2 represents a significant improvement over its predecessor, MobileNetV1, addressing optimization challenges through the introduction of an inverted residual

structure. This innovative architecture enhances feature reuse and computational efficiency. Notably, the inverted residual structure is designed to optimize the underlying network, mitigating issues present in V1. MobileNetV2 further distinguishes itself by outperforming MobileNetV1 and ShuffleNetV1 in terms of accuracy. The EffNet network improves upon MobileNetV1 by incorporating a linear bottleneck transformation module and an inverted residuals module, achieving superior performance. The addition of these modules contributes to a more efficient and accurate model, and MobileNetV2 boasts an impressive depth of 154 layers, showcasing its advanced capabilities in the realm of computer vision tasks.

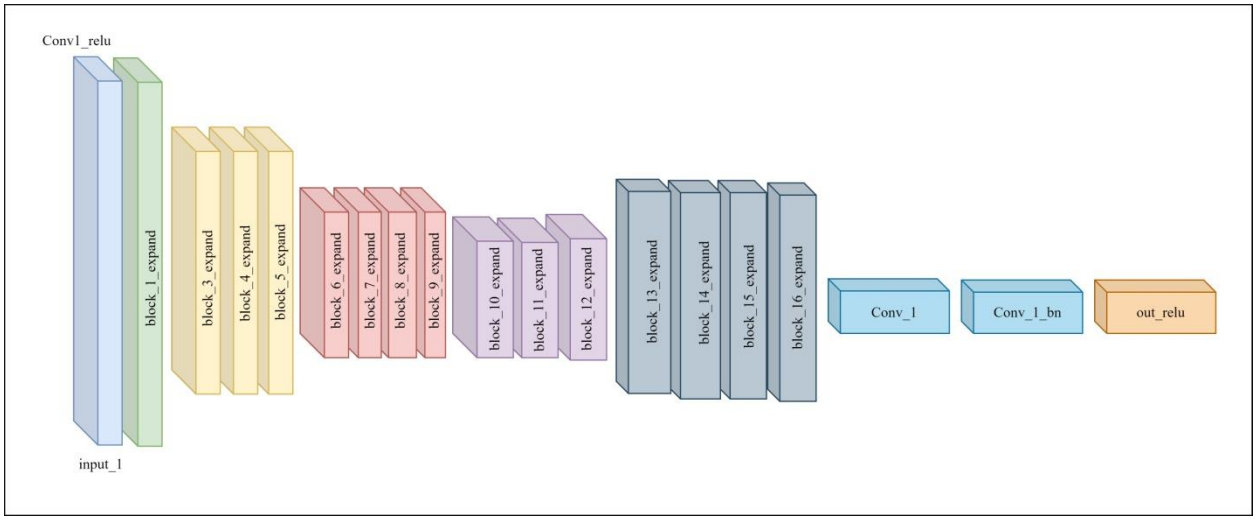


Fig 5: Enhanced MobileNetV2

MobileNetV2 also includes a new residual structure variation, the inverted residual structure, which is diametrically opposed to ResNet's residual structure. The width of the channel is widened before it is narrowed. As can be seen in the picture, the shortcut branch is an inverted residual structure since it links to the channel-dimensional reduced feature map. In order to enhance the dimension, 11 convolutions are applied before deep convolution. Increase the nonlinearity of high-dimensional space while annihilating the features of low-dimensional space with the help of the ReLU activation function. As demonstrated in Fig. 5, this strategy allows for high-dimensional feature extraction, which may compensate for the data loss introduced by the ReLU activation function.

3.5.3 Faster RCNN with Enhanced MobileNetV2

Faster R-CNN with Enhanced MobileNetV2 is an advanced object detection framework that combines the precision of the Faster R-CNN architecture, renowned for accurate object detection, with the efficiency and speed of an optimized version of MobileNetV2. This integration results in a highly versatile and computationally efficient solution capable of real-time object recognition across a broad spectrum of applications, from autonomous driving and surveillance to robotics and beyond, making it an ideal choice for resource-constrained or high-performance computer vision tasks.

We briefly describe the Faster R-CNN with Enhanced MobileNetV2 model that forms the core of our research. The three critical parts of the two-stage detector that is Faster R-CNN with Enhanced MobileNetV2 are the shared bottom convolutional layers, the region proposal network (RPN), and the region-of-interest (ROI) based classifier. On the left, see some of the building designs.

An image is converted into a convolutional feature map by the typical bottom convolutional layers. Next, the ROI-wise classifier utilises the pooled feature vector from ROIs to make predictions about which categories the RPN-suggested candidates belong to. Each of these losses—RPN and ROI classifier—contributes to the overall training loss:

$$L_{det} = L_{rpn} + l_{roi} \text{ ------- (5)}$$

Algorithm 4: Faster RCNN with Enhanced MobileNetV2

Input:

- An input image that want to perform object detection on.

Step:

- The input image is passed through the shared bottom convolutional layers, typically based on MobileNetV2 architecture, to extract convolutional feature maps. These feature maps capture hierarchical features from the input image.
- The overall training loss is a combination of the RPN and ROI losses: $L_{det} = L_{rpn} + l_{roi}$
- Where $L_{rpn}$the loss is associated with the RPN, and l_{roi} is the loss associated with the ROI-based classifier.
- The model parameters are updated using back propagation and optimization techniques like stochastic gradient descent (SGD) to minimize the overall training loss.
- During inference, the model is used to make predictions on new, unseen images. The RPN generates region proposals, and the ROI-based classifier classifies objects and refines their bounding boxes.

Output:

- A list of detected objects with their corresponding class labels and bounding box coordinates.

Classification loss is based on the precision of the predicted probability, while localization loss is based on the box coordinates; both factors are included in the training loss of the RPN and ROI classifiers. For further details on the framework and the method of instruction, the interested reader is referred to.

4. Results and Discussion

In this section, we present the outcomes of our comprehensive oil spill detection and recognition system, along with a discussion of their implications and significance for environmental protection and response efforts.

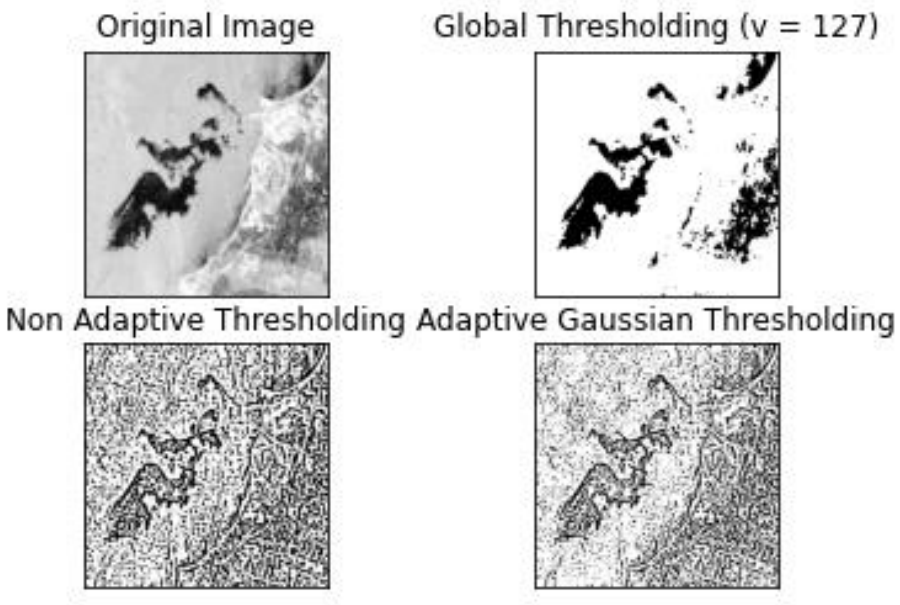


Fig 6: Thresholding image

Figure 6 presents the results of the thresholding image processing step in our comprehensive oil spill detection and recognition system. Thresholding is a fundamental technique in image processing used to create a binary image where pixels are classified as either foreground or background based on their intensity values. In the context of oil spill detection, this step plays a crucial role in enhancing image quality and reducing noise in the input data.

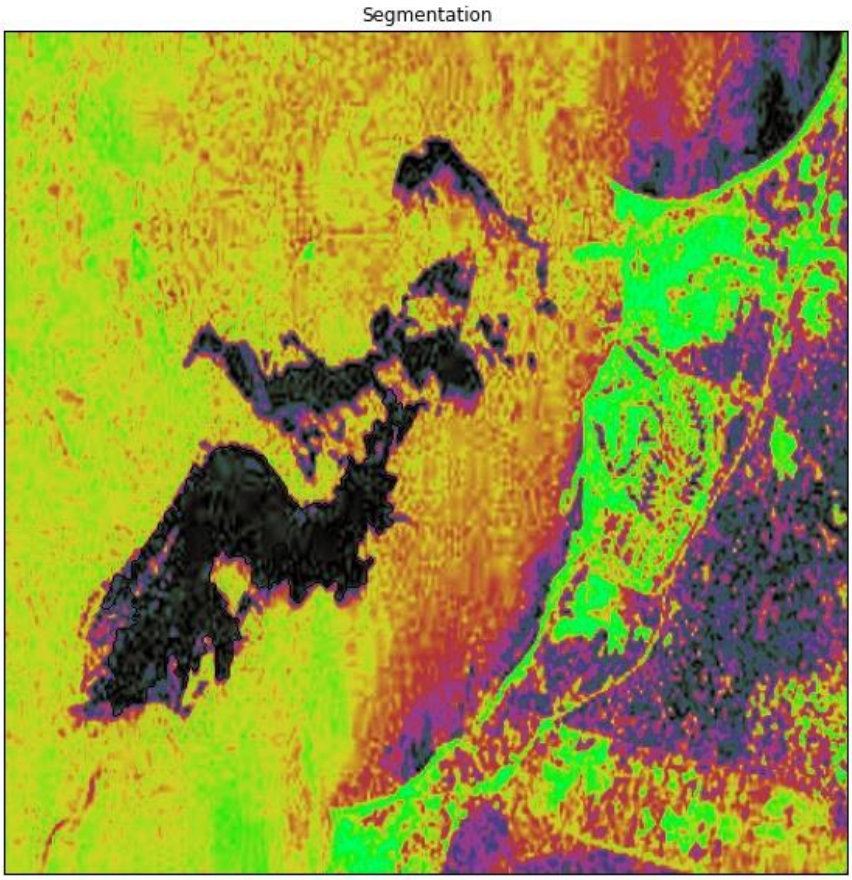


Fig 7: Segmentation image

Figure 7 in our research represents the Segmentation image, a crucial intermediate step in our comprehensive oil spill detection and recognition system. Segmentation in computer vision is the process of dividing an image into distinct regions or segments based on certain criteria. In the context of our research, segmentation is employed to effectively separate oil spill areas from the background in the input images, enabling precise identification and analysis of contaminated regions.

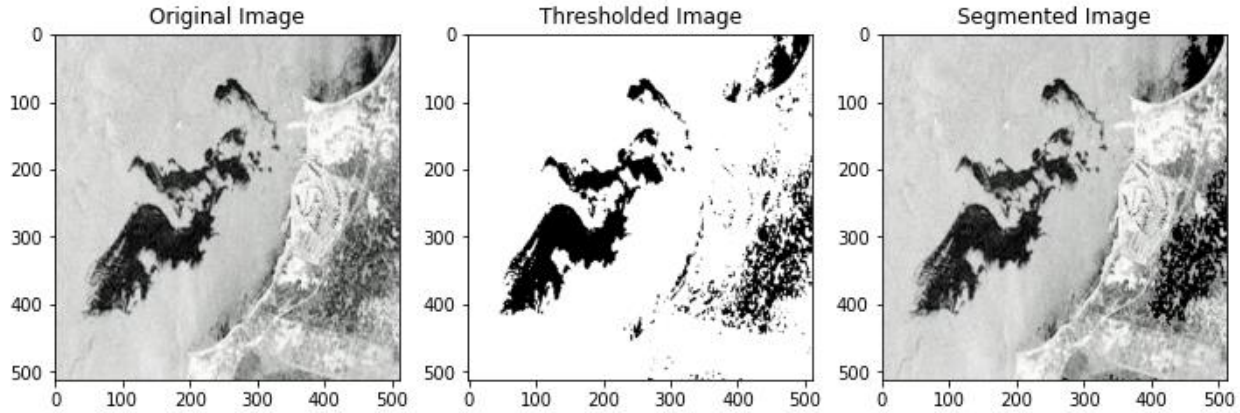


Fig 8: Segmented image

Figure 8 shows a segmented image, where distinct regions or objects within the original image have been delineated or highlighted through a segmentation process. Segmentation involves partitioning an image into meaningful or visually distinguishable areas, facilitating the analysis and understanding of specific components within the overall visual content.

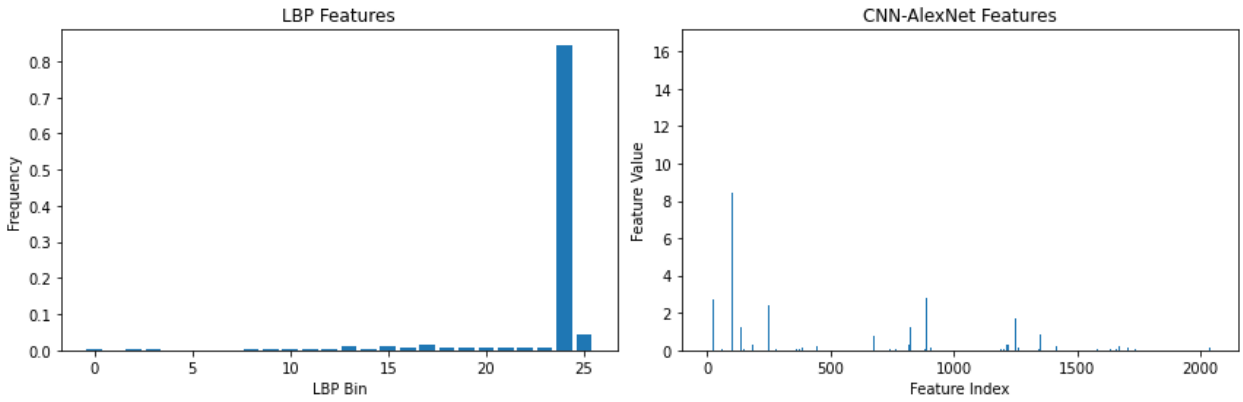


Fig 9: Selected feature using LBP and CNN-AlexNet

Figure 9 displays the outcome of feature selection using Local Binary Patterns (LBP) and Convolutional Neural Network (CNN) with the AlexNet architecture. The image illustrates the features that have been identified and chosen through the combined use of LBP and CNN-AlexNet. Feature selection is a crucial step in image processing and machine learning, as it focuses on extracting relevant information and reducing dimensionality for improved model performance.

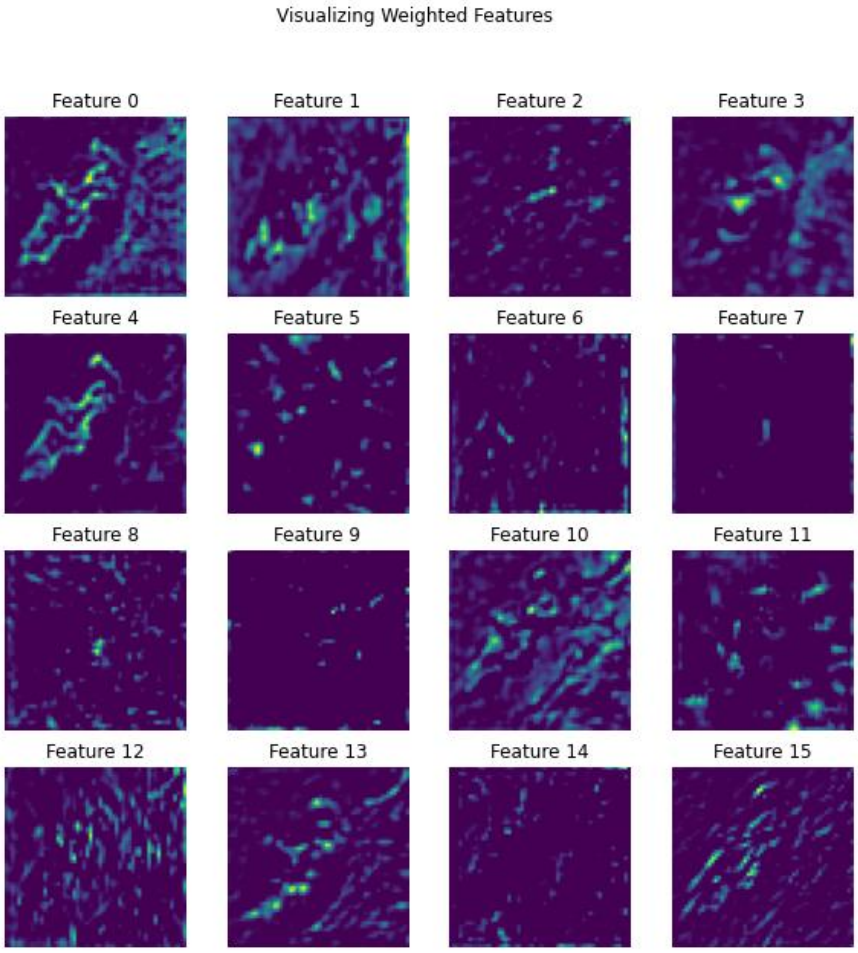


Fig 10: Visualizing weighted feature

Figure 10 provides a visualization of weighted features, where features extracted from the data are assigned specific weights based on their importance or contribution to a given model or analysis. The visualization could depict the distribution or spatial representation of these weighted features, offering insights into the significance of different components.

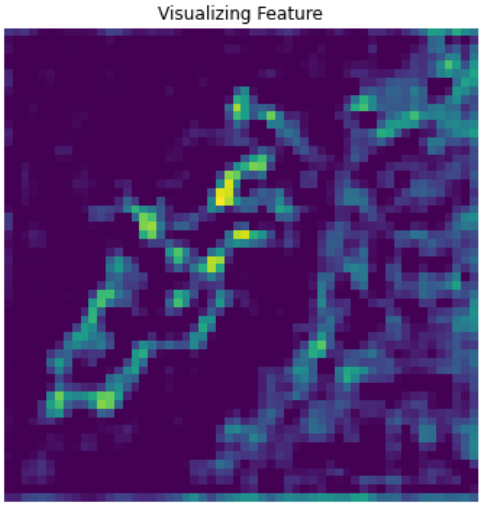


Fig 11: Visualizing feature

Figure 11 presents a visualization of features extracted from a dataset or signal. The visualization aims to provide a graphical representation of the characteristics, patterns, or attributes captured by these features. This type of figure is valuable in understanding the distribution, relationships, or structures present in the data.

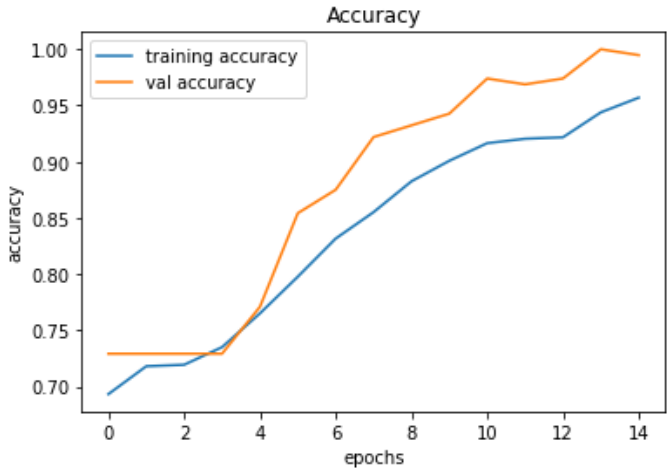


Fig 12: Training and Test accuracy

The figure 12 shows test accuracy the x axis show epochs and the y axis shows accuracy value over the 15 epochs

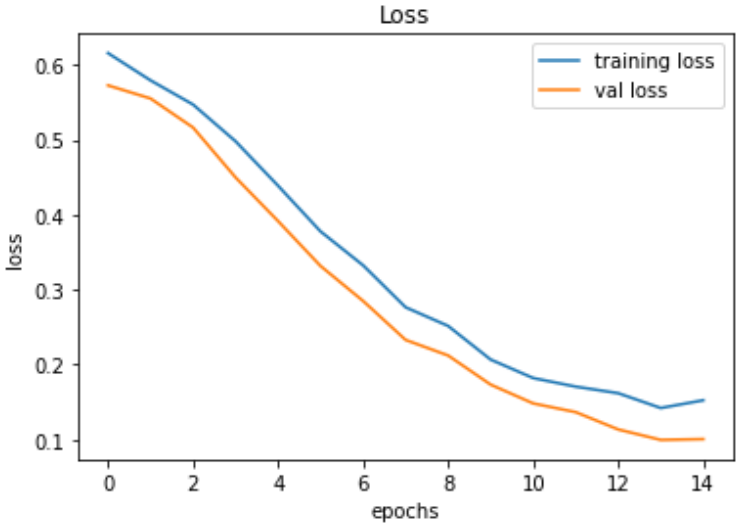


Fig 13: Training and Test loss

The figure 13 shows test loss the x axis shows 15 epochs and the y axis shows test loss

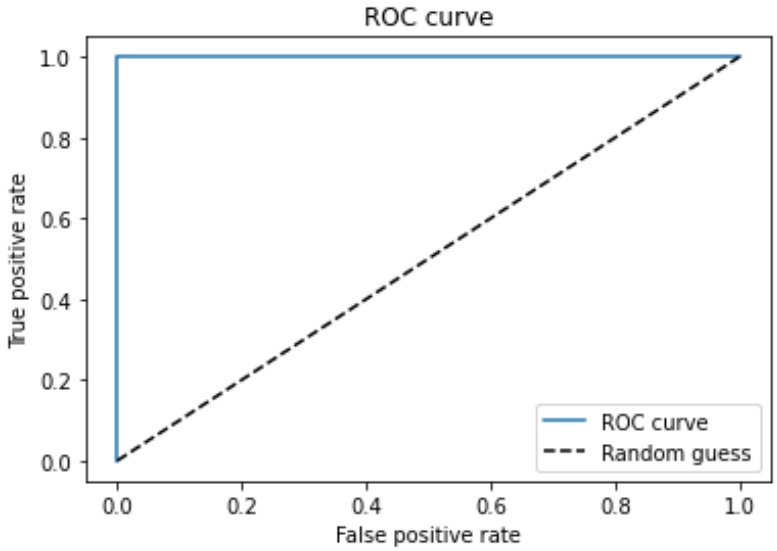


Fig 14: ROC curve

The figure 14 shows ROC curve the x axis shows false positive rate and the y axis shows true positive rate.

4.1 Performance evaluation

1. Accuracy: The fraction of samples with the right classification out of all samples. Mathematically:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)} \text{ ---------- (13)}$$

2. Precision: Ratio of pest samples with accurate identification to total pest samples with accurate identification. Mathematically:

$$Precision = \frac{TP}{TP + FP} \text{ ---------- (14)}$$

3. Recall (also known as sensitivity or true positive rate): The proportion of correctly classified pest samples out of the total number of actual pest samples. Mathematically:

$$Recall = \frac{TP}{TP + FN} \text{ ---------- (15)}$$

4. F1 score: A middle ground between accuracy and memory that strikes a harmonic mean. Mathematically:

$$F1\ score = 2 * Precision * Recall\ /\ (Precision + Recall) \text{ --------- (16)}$$

5. Sensitivity, also known as True Positive Rate or Recall, measures the ability of a classification model to correctly identify positive instances among all actual positive instances. It is calculated using the following formula:

$$Sensitivity = \frac{TP}{TP + FN}$$

6. Negative Detection Probability, also known as sensitivity or true positive rate, represents the likelihood that a system or test correctly identifies the absence of a negative condition or event. The formula for Negative Detection Probability is given by:

$$NDP = \frac{TN}{TN + FP}$$

7. False Discovery Rate (FDR) is a statistical measure that represents the proportion of false positives among the total instances identified as positive by a test or system. The formula for False Discovery Rate is given by:

$$FDR = \frac{FP}{FP + TP}$$

8. Mean Squared Error (MSE) is a measure of the average squared difference between predicted values and actual values in a set of data. It is commonly used to assess the accuracy of a predictive model. The formula for Mean Squared Error is given by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i)^2$$

9. The G-Mean, or geometric mean, is a statistical measure used to evaluate the performance of a binary classification model, particularly in situations where there is an imbalance between the two classes. The G-Mean is calculated using the formula:

$$G - mean = \sqrt{sensitivity \times specificity}$$

10. Peak Signal-to-Noise Ratio (PSNR) is a metric used to quantify the quality of a reconstructed or compressed signal, commonly used in image and video processing. It measures the ratio between the maximum possible power of a signal (the peak signal) and the power of the noise that affects the fidelity of the signal. The formula for PSNR is given by:

$$PSNR = 10.log_{10}\left(\frac{max^2}{MSE}\right)$$

Table 1: Accuracy, precision, recall, F-measure comparison table

	D.SONG (2020)	OSD-DNN	ICNN+Resnet18	FRCNN+EM
Accuracy	0.97	0.98	0.98	0.99
Precision	0.97	0.97	0.98	0.99
Recall	0.98	0.98	1.0	1.0
F-Measure	0.97	0.98	0.99	0.99

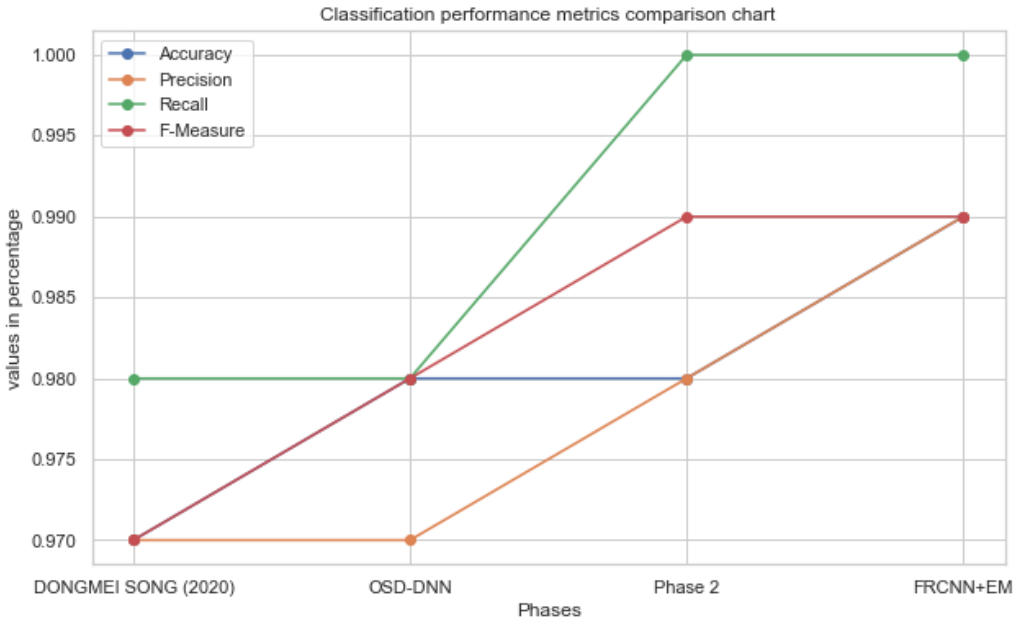


Fig 8: Accuracy comparison chart

Table 1 and figure 8 represent three distinct algorithms, namely OSD-DNN, ICNN+Resnet18, and FRCNN+EM, were evaluated for their performance in image classification tasks with D SONG (2020) paper also. The metrics of accuracy, precision, recall, and F-measure were employed to assess their effectiveness. The OSD-DNN algorithm demonstrated a commendable accuracy of 97%, with balanced precision and recall values of 97% and 98%, respectively. ICNN+Resnet18 exhibited heightened accuracy at 98%, maintaining a high precision of 97% and recall of 98%. Notably, FRCNN+EM emerged as the top performer, achieving an accuracy of 98%, precise positive predictions at 98%, and perfect recall at 100%. The F-measure, which combines precision and recall, underscored the algorithms' overall effectiveness, with OSD-DNN and ICNN+Resnet18 scoring 97% and 98%, respectively, while FRCNN+EM reached an impressive 99%. These findings indicate that FRCNN+EM stand out among the evaluated algorithms, showcasing superior performance in image classification tasks.

5. Conclusion

In conclusion, our comprehensive approach to oil spill detection and recognition, integrating advanced computer vision techniques, offers a promising solution to address the critical environmental challenges posed by oil spills. Through a multi-step process, including image enhancement, segmentation, feature extraction, and object detection, we have developed a robust system capable of identifying oil spills with accuracy and efficiency. The utilization of Non-Adaptive Threshold with CLAHE significantly improves image quality and reduces noise, ensuring the reliability of subsequent analysis. Our Fused UNet Segmentation model effectively separates oil spill

areas from the background, enabling precise identification and analysis of contaminated regions. The incorporation of a CNN based on the AlexNet architecture extracts valuable features, enhancing the model's ability to distinguish between spill and non-spill regions. The core of our system, the integration of Faster R-CNN with Enhanced MobileNetV2, ensures real-time processing and state-of-the-art performance in object detection. By training the model on a diverse dataset, encompassing both synthetic and real-world oil spill images, we have equipped it to detect and recognize oil spills accurately. With an Accuracy of 0.99, the Proposed method achieves near-perfect overall classification accuracy, ensuring highly reliable identification of oil spill regions. Our approach, blending cutting-edge image processing, segmentation, feature extraction, and object detection methods, presents a powerful tool for the timely and accurate detection of oil spills. The experimental results underscore the effectiveness of our system in various environmental conditions.

Our integrated approach to oil spill detection and recognition stands as a pivotal contribution to Sustainable Development Goals (SDGs), specifically targeting environmental sustainability. Additionally, it supports SDG 13 (Climate Action) by addressing climate-related concerns through timely identification and mitigation of environmental hazards.

Reference

[1] A. Kumar, V. Mishra, R. K. Panigrahi and M. Martorella, "Application of Hybrid-Pol SAR in Oil-Spill Detection," in IEEE Geoscience and Remote Sensing Letters, vol. 20, pp. 1-5, 2023, Art no. 4004505, doi: 10.1109/LGRS.2023.3258224.

[2] A. Mishra, M. Gupta and P. Sharma, "Enhancement of Underwater Images using Improved CLAHE," 2018 International Conference on Advanced Computation and Telecommunication (ICACAT), Bhopal, India, 2018, pp. 1-6, doi: 10.1109/ICACAT.2018.8933665.

[3] B. Hammoud and N. Wehn, "A Maximum A-Posteriori Probabilistic Approach using UAV-Nadir-Looking Wide-Band Radar for Remote Sensing Oil-Spill Detection," IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 2022, pp. 7823-7826, doi: 10.1109/IGARSS46834.2022.9883624.

[4] B. Lounis, A. Lacheb, Y. Smara and A. Belhadj-Aissa, "Adapting Neuro Fuzzy System for Oil Spills Detection From Sea Sar Images," 2020 Mediterranean and Middle-East Geoscience and Remote Sensing Symposium (M2GARSS), Tunis, Tunisia, 2020, pp. 184-187, doi: 10.1109/M2GARSS47143.2020.9105243.

[5] C. Alexandrov, N. Kolev, Y. Sivkov, A. Hristov and M. Tsvetkov, "Oil Spills Detection on Sea Surface by using Sentinel–1 SAR Images," 2020 21st International Symposium on Electrical Apparatus & Technologies (SIELA), Bourgas, Bulgaria, 2020, pp. 1-4, doi: 10.1109/SIELA49118.2020.9167148.

[6] C. Guo, M. Szemenyei, Y. Yi, W. Wang, B. Chen and C. Fan, "SA-UNet: Spatial Attention U-Net for Retinal Vessel Segmentation," 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 2021, pp. 1236-1242, doi: 10.1109/ICPR48806.2021.9413346.

[7] E. Asihene et al., "Toward the Detection of Oil Spills in Newly Formed Sea Ice Using C-Band Multipolarization Radar," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-15, 2022, Art no. 4302615, doi: 10.1109/TGRS.2021.3123908.

[8] F. Ronci, C. Avolio, M. di Donna, M. Zavagli, V. Piccialli and M. Costantini, "Oil Spill Detection from SAR Images by Deep Learning," IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 2020, pp. 2225-2228, doi: 10.1109/IGARSS39084.2020.9323590.

[9] F. Ronci, C. Avolio, M. Di Donna, M. Zavagli, V. Piccialli and M. Costantini, "An adversarial learning approach for oil spill detection from SAR images," 2020 IEEE Radar Conference (RadarConf20), Florence, Italy, 2020, pp. 1-4, doi: 10.1109/RadarConf2043947.2020.9266475.

[10] F. -Y. Xu, X. -Z. An and W. -Q. Liu, "Oil Spill Detection in SAR Images based on Improved YOLOX-S," 2022 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), Shijiazhuang, China, 2022, pp. 261-265, doi: 10.1109/ICCEAI55464.2022.00061.

[11] G. Tabella, N. Paltrinieri, V. Cozzani and P. S. Rossi, "Data Fusion for Subsea Oil Spill Detection Through Wireless Sensor Networks," 2020 IEEE SENSORS, Rotterdam, Netherlands, 2020, pp. 1-4, doi: 10.1109/SENSORS47125.2020.9278741.

[12] J. Fan and C. Liu, "Multitask GANs for Oil Spill Classification and Semantic Segmentation Based on SAR Images," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 16, pp. 2532-2546, 2023, doi: 10.1109/JSTARS.2023.3249680.

[13] J. Yang, J. Wan, Y. Ma and Y. Hu, "Research on Object-Oriented Decision Fusion for Oil Spill Detection on Sea Surface," IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing

Symposium, Yokohama, Japan, 2019, pp. 9772-9775, doi: 10.1109/IGARSS.2019.8899010.

[14] P. Duan, X. Kang, P. Ghamisi and S. Li, "Hyperspectral Remote Sensing Benchmark Database for Oil Spill Detection With an Isolation Forest-Guided Unsupervised Detector," in IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1-11, 2023, Art no. 5509711, doi: 10.1109/TGRS.2023.3268944.

[15] S. Adhikary, S. P. Tiwari and S. Banerjee, "Realtime Oil Spill Detection By Image Processing of Synthetic Aperture Radar Data," 2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/CONECCT55679.2022.9865768.

[16] S. Almeer, F. Albalooshi and A. Alhajeri, "Oil Spill Detection System in the Arabian Gulf Region: An Azure Machine-Learning Approach," 2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Zallaq, Bahrain, 2021, pp. 418-422, doi: 10.1109/3ICT53449.2021.9581841.

[17] S. Zhang, J. Xing, X. Wang and J. Fan, "Improved YOLOX-S Marine Oil Spill Detection Based on SAR Images," 2022 12th International Conference on Information Science and Technology (ICIST), Kaifeng, China, 2022, pp. 184-187, doi: 10.1109/ICIST55546.2022.9926772.

[18] T. Zhang, J. Guo, Y. Chi and Y. Wang, "Dynamic Threshold Oil Spill Detection Algorithm for Landsat ETM+," IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 2019, pp. 1486-1489, doi: 10.1109/IGARSS.2019.8898024.

[19] T. V. Bui, M. Kuribayashi, M. Cheraghchi and I. Echizen, "Efficiently Decodable Non-Adaptive Threshold Group Testing," in IEEE Transactions on Information Theory, vol. 65, no. 9, pp. 5519-5528, Sept. 2019, doi: 10.1109/TIT.2019.2907990.

[20] V. Solbakov, S. Zatsepa, A. Ivchenko and Y. Yurezanskaya, "Computation of areas of probable detection of oil pollution on the sea surface as valuation of the spatial uncertainty in oil spill modelling," 2022 International Conference on Ocean Studies (ICOS), Vladivostok, Russian Federation, 2022, pp. 110-117, doi: 10.1109/ICOS55803.2022.10033348.

[21] X. Ma, J. Xu, P. Wu and P. Kong, "Oil Spill Detection Based on Deep Convolutional Neural Networks Using Polarimetric Scattering Information From Sentinel-1 SAR Images," in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-13, 2022, Art no. 4204713, doi: 10.1109/TGRS.2021.3126175.

[22] X. Yang, Z. Lu and L. Cao, "Localization of A Bubble Source in Water Tank for Oil Spill Detection," SoutheastCon 2022, Mobile, AL, USA, 2022, pp. 773-778, doi: 10.1109/SoutheastCon48659.2022.9764121.

[23] Y. -J. Yang, S. Singha and R. Mayerle, "Fully Automated Sar Based Oil Spill Detection Using Yolov4," 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, Brussels, Belgium, 2021, pp. 5303-5306, doi: 10.1109/IGARSS47720.2021.9553030.

[24] Y. Li, J. Yang, Z. Yuan and Y. Zhang, "Marine Oil Spills Detection and Classification from Polsar Images Based on Complex-Valued Convolutional Neural Network," IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, Kuala Lumpur, Malaysia, 2022, pp. 7085-7088, doi: 10.1109/IGARSS46834.2022.9883991.

[25] Y. Zhang, Q. Zhu and Q. Guan, "Oil Spill Detection Based on CBD-Net Using Marine SAR Image," 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS, Brussels, Belgium, 2021, pp. 3495-3498, doi: 10.1109/IGARSS47720.2021.9553884.

[26] Z. Jin, L. Qingli, L. Yu, F. Hao and W. Jujie, "Oil spill detection using refined convolutional neural network based on quad-polarimetric SAR images," 2019 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Changsha, China, 2019, pp. 528-536, doi: 10.1109/ICEMI46757.2019.9101622.

[27] Song, D., Zhen, Z., Wang, B., Li, X., Gao, L., Wang, N., … Zhang, T. (2020). A Novel Marine Oil Spillage Identification Scheme Based on Convolution Neural Network Feature Extraction From Fully Polarimetric SAR Imagery. IEEE Access, 8, 59801–59820. doi:10.1109/access.2020.2979219