HOCHSCHULE BREMEN
CITY UNIVERSITY OF APPLIED SCIENCES

Electronics Engineering Project II

Summer Semester - 2023

Report on:

Semantic Segmentation of Watercrafts with using UNet Model

Submitted by:

Novruz Mammadli 5226126

Guided By:

Prof. Dr.-Ing. Benjamin Lehmann

Submitted on:

November 6, 2023

Abstract - Semantic segmentation plays a crucial role in computer vision applications, enabling the differentiation of objects within an image by machines. The focal point of this project is the task of semantic segmentation of ships in maritime scenes. The primary objective is the development of an accurate and efficient model using the UNet architecture. The MariBoats dataset, consisting of images instance-segmented with various watercraft, was employed for this purpose. Furthermore, the incorporation of semantic scene understanding involved the utilization of a pre-trained model based on the ADE20K dataset for the labeling of sea and sky regions. The project encompassed approaches utilizing both unpatched and patched images for ship segmentation, as well as the joint segmentation of ships, sea, and sky.

Contents

Table of Figures

1 Introduction

Semantic segmentation holds a crucial role in computer vision and image processing, with its applications spanning from autonomous navigation to environmental monitoring. In the scope of this project, emphasis was placed on the segmentation of ships within maritime scenes, a task of fundamental importance for applications such as ship detection and tracking.

This endeavor aimed to contribute to the advancement of ship semantic segmentation techniques. The project was motivated by the need for more accurate and efficient ship identification methods in complex maritime environments. Moreover, the significance of this research lies in its potential impact on enhancing maritime safety, security, and operational efficiency.

1.1 Background and Motivation

The background and motivation for this project are rooted in the need for advanced techniques in ship semantic segmentation. Maritime industries rely on accurate ship segmentation for various applications, prompting the exploration of innovative methods to enhance this process.

The field of computer vision has witnessed remarkable advancements in recent years, with semantic segmentation being a pivotal component for numerous applications, such as object recognition and scene understanding. The task of assigning pixel-wise labels to objects within an image has far-reaching implications, including in the maritime domain where the identification and delineation of ships is of paramount importance.

Maritime traffic and management are fundamental to global trade and security, and the accurate detection and segmentation of ships in imagery have practical implications in various domains, including search and rescue operations, environmental monitoring, and defense surveillance. The MariBoats dataset, replete with diverse watercraft imagery, offers an invaluable resource for developing and testing ship segmentation models.

Furthermore, the integration of semantic understanding, particularly labeling of sea and sky, is crucial in enhancing the overall utility of these models. As such, the utilization of pre-trained models on the ADE20K dataset to identify and segment these contextual elements in maritime imagery becomes a salient research endeavor. This can augment the precision of ship segmentation and contribute to the holistic comprehension of maritime scenes.

1.2 Objectives

The primary objective of this research is to employ the UNet architecture for the semantic segmentation of ships within the MariBoats dataset. The specific research objectives are as follows:

- Convert instance segmentation masks to semantic segmentation for the MariBoats dataset.
- Investigate and implement various strategies for enhancing semantic segmentation performance.
- Evaluate and compare the performance of different models with respect to segmentation accuracy, computational complexity, and robustness.
- Analyze the impact of resizing images, using single and multi class UNet models, incorporating color or greyscale images, applying dropout for regularization, and employing class weights.

1.3 Scope of the Project

This research confines its scope to the semantic segmentation of ships within the MariBoats dataset, leveraging the UNet architecture. It extends to the conversion of instance segmentation masks to semantic segmentation and the incorporation of sea and sky labels through a pre-trained model from the ADE20K dataset. The study also explores a diverse set of approaches, including resizing and patching of images, the use of single and multi class UNet models, consideration of color and greyscale images, application of dropout for regularization, and utilization of class weights for class imbalance.

1.4 Report Structure

The format of this report is as follows:

The chapter's summary of pertinent literature in the areas of semantic segmentation, UNet architecture, instance to semantic segmentation conversion, semantic understanding, and related works is presented in Chapter 2.

The datasets used, data pretreatment, and criteria for data splitting are all covered in detail in Chapter 3 - Data Preparation.

Chapter 4 - Methodology: This chapter elaborates on the methods employed in the research, including the model architectures, data types (color vs. greyscale), data augmentation, regularization techniques, and class weights.

Chapter 5 - Experimental Setup: This chapter delineates the metrics used for evaluation, the training process, and the hardware and software environment in which the experiments were conducted.

Chapter 6 - Results and Analysis: This chapter presents the results of the experiments and analyzes the performance of the different models, with a focus on quantitative and qualitative assessments, model comparisons, and computational complexity analysis.

Chapter 7 - Discussion: In this chapter, the research findings are discussed in detail, exploring the implications and nuances of different strategies.

Chapter 8 - Conclusion: This final chapter summarizes the research, highlights contributions, acknowledges limitations, and outlines potential avenues for future work.

Chapter 9 - References: This chapter lists the sources and references used throughout the report.

Chapter 10 - Appendices: This section includes supplementary information.

2 Literature Review

2.1 Image Segmentation

2.1.1 Segmentation Fundamentals

Image segmentation is essential for various downstream applications. It involves partitioning an image into multiple regions, each of which is assumed to correspond to a coherent object or scene. The goal of segmentation is to assign every pixel in the image to a particular class or object. It has widespread applications, including object tracking, image recognition, and scene understanding [1][2].

2.1.2 Instance Segmentation

Instance segmentation is a more refined form of image segmentation that not only identifies and delineates objects but also distinguishes between multiple instances of the same object category. In instance segmentation, every object in the image is assigned a unique label. This is particularly useful in scenarios where distinguishing between individual instances of objects is critical, such as in autonomous driving, where differentiating between different vehicles is necessary for decision-making [3].

Instance segmentation is accomplished through techniques that integrate object detection and traditional semantic segmentation. It not only identifies object categories but also provides information about the precise boundaries of each individual object instance within the same category.

In this project, MariBoats dataset was used and this dataset exists instance segmentation of watercrafts.

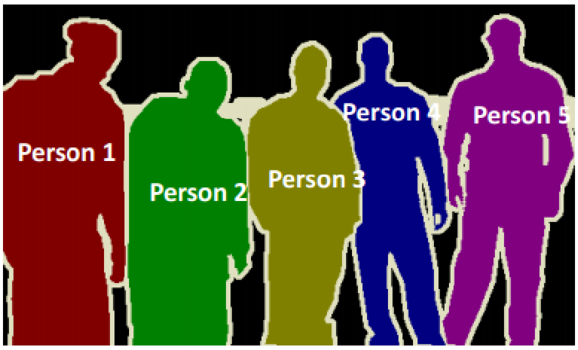

Figure 1. Example of Instance Segmentation [4].

2.1.3 Semantic Segmentation

The act of categorizing each pixel in a picture into a particular item category or class without differentiating between distinct occurrences of the same category is known as semantic segmentation, a subsection of image segmentation. To put it another way, it assigns each pixel a class name that designates the object or area to which it belongs. This type of segmentation is highly relevant in applications where understanding the scene and recognizing objects is paramount, such as in autonomous navigation, object recognition, and medical image analysis [5].

The core objective of semantic segmentation is to produce a pixel-wise mask that designates the category of each pixel within an image. This pixel-level classification enables machines to comprehend the visual scene, aiding in tasks like object recognition and scene interpretation. Semantic segmentation is often performed using deep learning techniques, particularly convolutional neural networks (CNNs) [5].

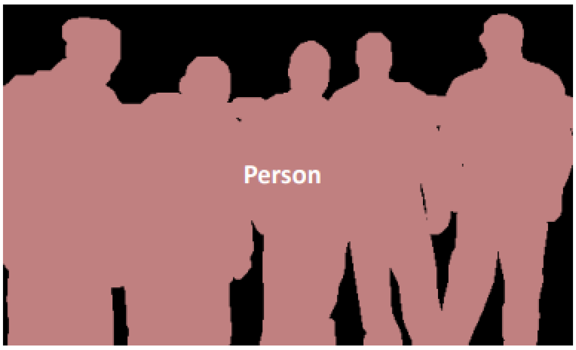

Figure 2. Example of Semantic Segmentation [4].

2.1.3.1 Techniques in Semantic Segmentation

Semantic segmentation techniques have evolved over the years, with deep learning methods significantly improving the accuracy and efficiency of the process. Convolutional neural networks, such as U-Net, FCN, and DeepLab, have become the go-to architectures for semantic segmentation tasks. These networks learn to capture contextual information and spatial relationships between pixels, allowing for the precise classification of each pixel within an image [5][6].

Additionally, semantic segmentation often involves the use of labeled datasets, where each pixel in an image is meticulously annotated with its corresponding class label. These annotated datasets are used to train deep learning models, enabling them to generalize and perform pixel-wise classification on new, unseen images.

2.2 UNet Architecture

Ronneberger et al. first presented the UNet architecture, a deep learning model created for semantic segmentation tasks, in 2015 [6]. Due to its capacity to provide precise pixel-wise predictions, this ground-breaking architecture has demonstrated to be extremely efficient in a variety of medical image analysis and computer vision applications. This chapter will provide a thorough introduction of the UNet architecture, including its layout, elements, and applications.

In the Figure 3. each blue box corresponds to a multi-channel feature map, which is essentially a grid of values representing different image features. The number of channels in each feature map is indicated at the top of the box. Channels are like image layers that capture specific information about the input. The x-y size of each feature map is provided at the lower left edge of the box, specifying the dimensions of the grid. White boxes represent copied feature maps, indicating that some feature maps are directly propagated or copied to subsequent layers. These copied feature maps retain important information for later stages of the architecture. The arrows between the boxes denote the different operations that connect the feature maps. These operations typically involve convolutional layers, pooling, and upsampling, which help extract and process features from the input image [6].

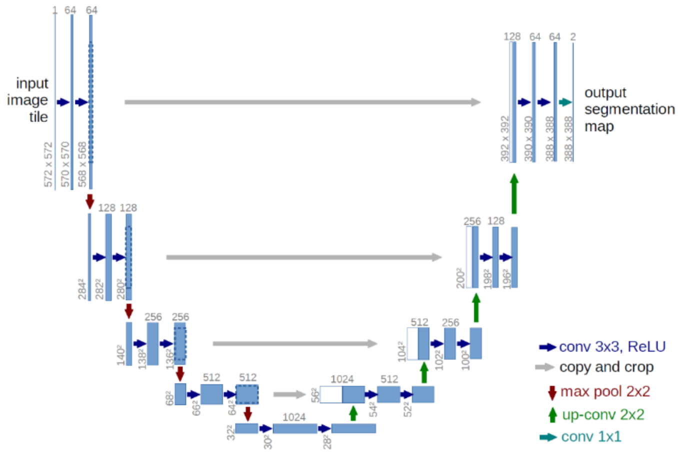

Figure 3. UNet Architecture [6].

2.2.1 Architecture Overview

The UNet architecture derives its name from its U-shaped design, characterized by a contracting path (encoder) followed by an expansive path (decoder). This architecture is symmetrical, featuring skip connections between the encoder and decoder, which enable the model to capture both high-level semantic information and fine-grained details. The primary objective of UNet is to produce pixel-wise segmentation masks for each pixel in an input image [6].

2.2.2 Encoder: Contracting Path

The encoder component of UNet is responsible for reducing the spatial dimensions of the input image while extracting high-level features. This is accomplished through a series of convolutional layers, typically employing small receptive fields and downsampling operations, such as max-pooling. As the encoder progresses, it learns to identify hierarchical features in the image, from low-level edges to higher-level object representations [6].

2.2.3 Decoder: Expansive Path

The decoder component of UNet plays a crucial role in upsampling and reconstructing the feature maps produced by the encoder. It uses transposed convolution layers, also known as deconvolutions, to increase the spatial resolution of the feature maps. Additionally, skip connections from the encoder are concatenated with the decoder's feature maps at each stage. These skip connections allow the model to merge both high-level and low-level feature representations, enabling precise segmentation [6].

2.2.4 Skip Connections

Skip connections, often referred to as skip or residual connections, are a distinctive feature of the UNet architecture. They enable the model to combine coarse-grained features from the encoder with fine-grained features from the decoder. This fusion of information is crucial for accurate pixel-wise predictions. Skip connections ensure that the model can capture object boundaries and small details that might be lost during the downscaling process [6].

2.2.5 Final Layer and Activation

The final layer of the UNet decoder typically employs a 1x1 convolution followed by a suitable activation function, such as the sigmoid or softmax function. This layer produces the final pixel-wise segmentation mask, where each pixel is assigned a class label corresponding to the object or region it belongs to [6].

2.2.6 Applications of UNet

UNet has gained popularity in various fields, with notable applications in medical image segmentation, such as cell and tissue segmentation, lesion detection, and organ segmentation.

It is also used in general computer vision tasks, including object detection, scene parsing, and autonomous driving, where precise segmentation of objects and regions is essential.

2.2.7 Challenges and Future Directions

Despite its effectiveness, UNet and its variants are not without challenges. Training a UNet model typically requires large, annotated datasets, and ensuring the model generalizes well to unseen data can be a complex task. Additionally, addressing class imbalance and handling small object segmentation are ongoing research areas.

2.3 MariBoats Dataset

2.3.1 Dataset Description

The dataset utilized for training the semantic segmentation model in this study is the "Mariboats" dataset, a component of the Visible Ship Dataset. The Visible Ship Dataset was designed for ship instance segmentation tasks in visible light images. It encompasses two distinct subsets, namely "MariBoats" and "MariBoatsSubclass," each catering to different research purposes within the field of marine ship instance segmentation [7].

2.3.2 MariBoats Dataset

The MariBoats dataset incorporates a total of 6.2 thousand images, with each ship present in the images assigned to a single category, specifically labeled as 'ship.' Consequently, this dataset comprises a substantial 15.7 thousand ship segmentation annotations. By designating all the ships as a single category, the MariBoats dataset satisfies the fundamental requirements for basic instance segmentation tasks. For instance, this dataset is pertinent in scenarios involving the detection and avoidance of obstacles, such as ships, during unmanned driving operations, particularly within complex sea environments [7].

2.3.3 MariBoatsSubclass Dataset

In contrast, the MariBoatsSubclass dataset, a constituent of the Visible Ship Dataset, encompasses 3.1 thousand images and 4.5 thousand ship annotations. This dataset exhibits a more intricate structure, featuring six distinct categories of marine ships, including Engineering Ship (Eng.), Cargo Ship (Carg.), Speedboat (Sp.), Passenger Ship (Pass.), Official Ship (Off.), and Unknown Ship (Unk.). The MariBoatsSubclass dataset caters to a broader spectrum of research objectives. It facilitates not only the segmentation of ships but also the precise identification of marine ship categories within complex marine scenes [7].

2.3.4 Dataset Origin and Benchmarking

The Visible Ship Dataset, including both the MariBoats and MariBoatsSubclass subsets, was constructed following a similar methodology to that used in creating the Microsoft Common

Objects in Context (COCO) datasets. It features visible light images with varying resolutions. Researchers can utilize this dataset as a benchmark to evaluate the performance and efficacy of their instance segmentation approaches in visible light marine environments [7].

2.4 Semantic Understanding of Scenes through ADE20K Dataset

In the process of training a multi-class UNet model for the segmentation of sky and sea images within the MariBoatsSubclass dataset, an additional pre-trained model was employed to facilitate the labeling and masking of these distinct regions.

2.4.1 ADE20K Dataset

The ADE20K dataset is a comprehensive resource for semantic segmentation tasks, comprising more than 27,000 images drawn from the SUN and Places databases. These images are meticulously annotated with object labels, covering a vast array of over 3,000 object categories. Furthermore, ADE20K extends its annotation to include object parts and even subparts of objects. The dataset offers original annotated polygons and provides object instances for amodal segmentation. It's worth noting that the dataset respects privacy concerns by anonymizing images through the blurring of faces and license plates [8][9].

2.4.2 Utilization of ADE20K for Sky and Sea Labeling

For the specific purpose of segmenting sky and sea regions within the MariBoatsSubclass dataset, the ADE20K dataset played a pivotal role. By leveraging the comprehensive object categories and fine-grained annotations provided by ADE20K, the sky and sea regions in the MariBoatsSubclass dataset were efficiently labeled and masked. This process significantly expedited the creation of training data for the multi-class UNet model, enabling it to learn to distinguish and segment these two distinct environments.

The availability of such a rich and diverse dataset like ADE20K facilitates the transfer of knowledge and expertise across different domains of computer vision, ultimately enhancing the performance of models, such as the multi-class UNet, in the segmentation of complex scenes, including the maritime landscapes depicted in the MariBoatsSubclass dataset.

3 Data Preprocessing

The effectiveness of deep learning models in image segmentation greatly depends on the quality and structure of the training data. In this chapter, the data preprocessing steps undertaken for the training of both single and multi-class UNet models on the MariBoatsSubclass dataset will be discussed. These preprocessing steps encompass data selection, resizing, patching, and the choice between using colored or grayscale images.

In the data preprocessing stage of the model, the input images are appropriately prepared for prediction by utilizing the Keras ‘normalize’ function and applying L2 normalization. These techniques standardize the data, reducing pixel intensity variations, and ensuring that the input images are optimally processed before being subjected to the model's predictions. This crucial preprocessing step enhances the model's accuracy and reliability in delivering meaningful results.

3.1 Data Selection

The MariBoatsSubclass dataset was the primary data source for training both single and multi-class UNet models. In total, there were 3,125 images available for training and validation. For the single-class UNet model, all 3,125 images were utilized. However, in the case of the multi-class UNet model, 1,867 images were used. This reduced dataset size for the multi-class model was a result of employing a pre-trained ADE20K model to label the sea and sky regions within the MariBoatsSubclass dataset. Some of the labeling results did not meet the expected standards, leading to the removal of these subpar images from the dataset. Consequently, the multi-class UNet model was trained on the remaining 1,867 images.

3.2 Resizing and Patching

Another important preprocessing step involved resizing and patching the images. As given in Table 1, within the MariBoatsSubclass dataset, there were 964 unique-shaped images for the single-class UNet model and 654 unique-shaped images for the multi-class UNet model. To ensure uniformity and compatibility with the network architecture, all of these images were resized to a common dimension of 256×256 pixels. In cases where the original image dimensions were not divisible by 256×256, black padding was added to achieve divisibility before the images were divided into smaller patches. This patching approach enabled the model to analyze images in smaller, more manageable segments, ultimately enhancing segmentation accuracy.

Dataset	Number of Unique-Shaped Images
Single-Class UNet’s Dataset	964
Multi-Class UNet’s Dataset	654

Table 1. Number of Unique-Shaped Images in Datasets.

As illustrated in Table 2, the process of patching has resulted in a dataset size increase of over fourfold, which in turn contributes to increased computational complexity during the training phase of these datasets. Nonetheless, the practice of preserving image details through patching can ultimately lead to improved results.

Dataset	Number of Images
Resized Single Class UNet's Dataset	3125
Patched Single Class UNet's Dataset	12666
Resized Multi Class UNet's Dataset	1867
Patched Multi Class UNet's Dataset	7672

Table 2. Number of Images in the Datasets

3.3 Color vs. Grayscale

The choice between using colored images or grayscale images also played a role in the data preprocessing stage. In the first approach, the images were retained in their original colored format, allowing the model to work with RGB information. In the second approach, images were converted to grayscale before training.

4 Methodology

4.1 Model Architectures

4.1.1 Single UNet Model

The UNet architecture, recognized for its efficacy in biomedical image segmentation, forms the core of the semantic segmentation models employed in this research. The Single UNet model is a primary focus, dedicated to the segmentation of ships within the images. It comprises a contracting and expansive path, characterized by skip connections, enabling the extraction of low-level and high-level features. This architecture has demonstrated remarkable performance in various segmentation tasks, serving as a robust choice for object segmentation.

In this project, for single-class UNet models, a sigmoid function was employed at the final layer, and binary cross-entropy was utilized as the loss function.

4.1.2 Multi UNet Model

In contrast to the single class UNet model, the multi class UNet model extends the segmentation scope to encompass the identification and labeling of contextual elements, namely, sea and sky, in addition to ships. By incorporating this broader understanding of scenes, the multi class UNet model provides a more comprehensive semantic segmentation. It follows the same UNet architecture principles, albeit with additional output channels for the contextual elements. This architecture enhances the contextual understanding of maritime scenes.

In this project, for multi-class UNet models, the final layer incorporated a softmax function, and the loss function employed was categorical cross-entropy.

4.2 Dropout and Class Weights

To address overfitting concerns, dropout, a regularization technique, is introduced into the UNet model's architecture. Dropout involves randomly deactivating neurons during training, preventing the model from relying too heavily on any single feature.

Additionally, the class weights assigned to different categories play a pivotal role in influencing the model's ability to distinguish and accurately classify objects within an image. These class weights are meticulously fine-tuned to address the inherent imbalances in class distribution, ensuring that the model's predictions are not skewed towards dominant classes. In the evaluated "Single Class UNet" models, the assigned class weights for "Background" and "Ship" reveal intriguing patterns. As depicted in Table 3, the "Resized Single Class UNet" model demonstrates a weight of 0.65659789 for "Background" and a higher weight of 2.0964455 for "Ship." In contrast, the "Patched Single Class UNet" model features a slightly lower weight for "Background" at 0.58178797 but significantly elevates the weight for "Ship" to 3.55668414. This class weight variation indicates a strategy to emphasize the importance of the "Ship" class, potentially addressing class imbalances and enhancing the model's accuracy in ship segmentation tasks.

Model	Class Weight of Background	Class Weight of Ship
Resized Single Class UNet	0.65659789	2.0964455
Patched Single Class UNet	0.58178797	3.55668414

Table 3. Class Weights of SIngle Class UNet Models

In the "Multi Class UNet" models, the assignment of class weights extends to a broader spectrum, considering additional classes such as "Sea" and "Sky." As illustrated in Table 4, the "Resized Multi Class UNet" model adopts class weights of 3.1589856 for "Background," 0.7189355 for "Sea," 1.0795486 for "Ship," and 0.73196572 for "Sky." In comparison, the "Patched Multi Class UNet" model fine-tunes these weights to 0.54001312 for "Background," 1.24379038 for "Sea," 1.86199477 for "Ship," and 1.23894133 for "Sky." These class weights signify a deliberate effort to balance the impact of different classes on the model's learning process.

Model	Class Weight of Background	Class Weight of Sea	Class Weight of Ship	Class Weight of Sky
Resized Multi Class UNet	3.1589856	0.7189355	1.0795486	0.73196572
Patched Multi Class UNet	0.54001312	1.24379038	1.86199477	1.23894133

Table 4 Class Weights of Multi Class UNet Models

5 Experimental Setup

5.1 Evaluation Metrics

In this section, the experimental setup for evaluating the performance of the semantic segmentation models is detailed. The choice of appropriate evaluation metrics is fundamental in quantifying the models' effectiveness. The following evaluation metrics were utilized:

- Intersection over Union (IoU): IoU measures the overlap between predicted and ground truth segmentation masks. It is calculated as the intersection of the two masks divided by their union, providing a measure of pixel-wise accuracy.
- Mean Intersection over Union (mIoU): mIoU computes the average IoU across all classes, offering a comprehensive evaluation of the model's segmentation performance.
- Pixel Accuracy: Pixel accuracy is the ratio of correctly classified pixels to the total number of pixels in the image. It provides an overall measure of segmentation accuracy.

These evaluation metrics enable a thorough assessment of the models' ability to accurately segment objects and contextual elements within the images.

5.2 Training Procedure

To prevent overfitting further, a 10% dropout was introduced within the UNet model's architecture.

The dataset was split into training and testing subsets, with 80% of the images reserved for training and the remaining 20% for validation. During training, the models were fine-tuned on the training subset, while the testing subset facilitated the evaluation of the models' generalization performance.

5.3 Hardware and Software

The computational infrastructure used for conducting the experiments consisted of a workstation equipped with a multi-core CPU and a high-end GPU, specifically the NVIDIA A40. The GPU acceleration significantly expedited the training process, particularly in scenarios involving multi class UNet models and color images.

The software stack employed included the TensorFlow deep learning framework for model development and training. Additional libraries, such as NumPy and OpenCV, were utilized for data preprocessing and manipulation, MatPlotLib library was used for data visualization. The experiments were conducted on a Linux-based Debian operating system, offering a stable and conducive environment for deep learning tasks. This comprehensive hardware and software setup facilitated the systematic evaluation of the semantic segmentation models and the attainment of meaningful insights.

6 Results and Analysis

This section presents a comprehensive analysis of the results obtained from the 32 semantic segmentation models, categorized based on image type (color or grayscale) and image resizing strategy (resized or patched). The models were evaluated using a range of metrics, including accuracy, Mean Intersection over Union (mIoU), class-specific IoU scores, and computational time. The analysis encompasses quantitative and qualitative assessments, model-to-model comparisons, and an exploration of computational complexities.

6.1 Models' Results

6.1.1 Results of Single Class UNet Model

6.1.1.1 Results of Resized Single Class UNet Model

6.1.1.1.1 Color Images

Among the 'Resized Single Class UNet' models, the color variant stands out as a top performer (see Table 5 for detailed results). Specifically, 'Resized_SingleUNet_color_dropout' achieved an exceptional accuracy of 94.18%, outperforming its grayscale counterpart by a significant margin. The mean Intersection over Union (mIoU) reached an impressive 85.56%, demonstrating the model's capacity to accurately segment objects within the maritime images. Furthermore, the class-specific IoU scores reveal the model's exceptional proficiency in classifying ships, with an IoU for class 1 (ship) at a remarkable 78.5%.

Model names	**Accuracy (%)**	**Mean IoU (%)**	**IoU for background (%)**	**IoU for ship (%)**	**Time (seconds)**
Resized_Single_UNet_color	93.21	82.70	91.62	73.79	297.4
Resized_Single_UNet_color_dropout	94.18	85.56	92.62	78.50	370.7
Resized_Single_UNet_color_weights	93.01	83.13	91.14	75.13	297.5
Resized_SingleUNet_color_weights_dropout	93.21	83.83	91.3	76.37	362.5

Table 5. Results of Single Class UNet Models on Resized and Color Images.

6.1.1.1.2 Gray Images

The 'Resized Single Class UNet' model in grayscale, while still maintaining respectable performance, falls slightly short of its color counterpart. The accuracy achieved by the 'Resized_SingleUNet_gray_dropout' model is 93.27%, slightly below the color variant (see Table 6 for details). The mIoU of 83.87% demonstrates the model's commendable segmentation accuracy, albeit at a level lower than the color model. Notably, the model excels in classifying ships, with an IoU for class 1 (ship) at 76.33%.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for ship (%)	Time (seconds)
Resized_SingleUNet_gray	93.13	82.63	91.49	73.77	286.7
Resized_SingleUNet_gray_dropout	93.27	83.87	91.40	76.33	345.4
Resized_SingleUNet_gray_weights	91.67	81	89.3	72.7	281.8
Resized_SingleUNet_gray_weights_dropout	76.28	73.14	76.28	70	351.5

Table 6. Results of Single Class UNet Models on Resized and Gray Images.

6.1.1.2 Results of Patched Single Class UNet Model

6.1.1.2.1 Color Images

As shown in Table 7, the "Patched Single Class UNet" model in color has exhibited notable performance, with "Patched_SingleUNet_color_dropout" leading the pack. This model achieved an accuracy of 95.73%, making it the top-performing model among all variants. The mean Intersection over Union (mIoU) of 77.92% underscores the model's exceptional ability to accurately segment objects in maritime scenes. Furthermore, the class-specific IoU scores demonstrate that this model excels in the classification of ships, with an IoU for class 1 (ship) at an impressive 73.27%.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for ship (%)	Time (seconds)
Patched_SingleUNet_color	95.2	75.81	80.80	70.82	1178.9
Patched_SingleUNet_color_dropout	95.73	77.92	82.57	73.27	1425
Patched_SingleUNet_color_weights	95.28	76.82	81.06	72.57	1190
Patched_SingleUNet_color_weights_dropout	95.09	76.54	80.45	72.63	1418

Table 7. Results of Single Class UNet Models on Patched and Color Images.

6.1.1.2.2 Gray Images

As illustrated in Table 8, the "Patched Single Class UNet" models in grayscale, while still demonstrating strong segmentation capabilities, these models generally fall slightly behind their color counterparts. "Patched_SingleUNet_gray_dropout" leads this group with an accuracy of 93.6% and an mIoU of 76.96%. The IoU for class 1 (ship) reaches 71.84%, indicating the model's proficiency in ship segmentation.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for ship (%)	Time (seconds)
Patched_SingleUNet_gray	94.9	74.07	79.82	68.33	1085
Patched_SingleUNet_gray_dropout	95.6	76.96	82.08	71.84	1392
Patched_SingleUNet_gray_weights	94.7	73.68	79.20	68.16	1127
Patched_SingleUNet_gray_weights_dropout	85.94	54.49	58.96	50	1341

Table 8. Results of Single Class UNet Models on Patched and Gray Images.

6.1.2 Results of Multi Class UNet Model

6.1.2.1 Results of Resized Multi Class UNet Model

6.1.2.1.1 Color Images

Within the category of "Resized Multi Class UNet" models, as given in Table 9, the color variant has demonstrated impressive performance, with "Resized_MultiUNet_color_dropout" emerging as a standout model. This model achieved an accuracy of 88.57%, emphasizing its strong overall performance. The mean Intersection over Union (mIoU) reached 71.62%, indicating the model's capability to accurately segment multiple classes within maritime images. Particularly, the IoU for class 2 (ship) reached an exceptional 76.5%, highlighting the model's proficiency in this crucial category.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for sea (%)	IoU for ship (%)	IoU for sky (%)	Time (seconds)
Resized_MultiUNet_color	88.12	71.5	36.05	83.8	75.2	91.2	189.2
Resized_MultiUNet_color_dropout	88.57	71.62	35.35	84.3	76.5	90.3	214.8
Resized_MultiUNet_color_weights	87.92	70.86	35.72	82.8	75.3	89.7	187.9
Resized_MultiUNet_color_weights_dropout	87.38	70.5	35.92	82.2	74.5	89.4	227.4

Table 9. Results of Multi Class UNet Models on Resized and Color Images.

6.1.2.1.2 Gray Images

In the "Resized Multi Class UNet" models in grayscale, as shown in Table 10, while still demonstrating commendable performance, these models typically lag behind their color counterparts. The leading model, "Resized_MultiUNet_gray_dropout," achieved an accuracy of 88.66% and an mIoU of 71.95%. The IoU for class 2 (ship) reached 75.5%, underscoring the model's expertise in ship segmentation.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for sea (%)	IoU for ship (%)	IoU for sky (%)	Time (seconds)
Resized_MultiUNet_gray	88.14	69.76	29.52	82.8	75.4	91.3	176.6
Resized_MultiUNet_gray_dropout	88.66	71.95	36.38	84.1	75.5	91.8	212.8
Resized_MultiUNet_gray_weights	86.99	70.13	35.22	82.3	72.9	90.1	182.3
Resized_MultiUNet_gray_weights_dropout	86.96	70.51	36.53	81.6	73.7	90.2	222.1

Table 10. Results of Multi Class UNet Models on Resized and Gray Images.

6.1.2.2 Results of Patched Multi Class UNet Model

6.1.2.2.1 Color Images

In the "Patched Multi Class UNet" models, the color variants have exhibited remarkable performance, as illustrated in Table 11. " Patched_MultiUNet_color_weights_dropout" stands out as the leading model within this category, achieving an outstanding accuracy of 94.08%. The mean IoU reached an impressive 86.07%, highlighting the model's exceptional capability to accurately segment multiple classes within maritime images. Notably, the class-specific IoU scores demonstrate the model's prowess in the classification of ships, with an IoU for class 2 (ship) at a remarkable 75.99%.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for sea (%)	IoU for ship (%)	IoU for sky (%)	Time (seconds)
Patched_MultiUNet_color	92.8	83.25	88.71	81.31	72.25	90.75	744.1
Patched_MultiUNet_color_dropout	93.70	85.05	89.57	85.11	74.24	91.27	899.6
Patched_MultiUNet_color_weights	92.6	82.99	88.71	80.86	73.07	89.3	764.2
Patched_MultiUNet_color_weights_dropout	94.08	86.07	90.47	85.78	75.99	92.03	907.4

Table 11. Results of Multi Class UNet Models on Patched and Color Images.

6.1.2.2.2 Gray Images

In the "Patched Multi Class UNet" models in grayscale, as indicated in Table 12, these models have shown competitive performance while still maintaining strong segmentation capabilities. " Patched_MultiUNet_gray_weights_dropout" leads this category with an accuracy of 93.55% and an mIoU of 84.98%. The IoU for class 2 (ship) reaches an impressive 74.44%, signifying the model's proficiency in ship segmentation. Although the grayscale models exhibit commendable performance, they do not quite match the accuracy and mIoU levels achieved by their color counterparts.

Model names	Accuracy (%)	Mean IoU (%)	IoU for background (%)	IoU for sea (%)	IoU for ship (%)	IoU for sky (%)	Time (seconds)
Patched_MultiUNet_gray	92.10	82.09	88.59	79.76	70.74	89.24	684.1
Patched_MultiUNet_gray_dropout	93.54	84.94	89.48	84.18	74.61	91.50	841.5
Patched_MultiUNet_gray_weights	92.98	83.96	88.89	82.31	73.82	90.83	722.2
Patched_MultiUNet_gray_weights_dropout	93.55	84.98	89.55	84.04	74.44	91.88	901.5

Table 12. Results of Multi Class UNet Models on Patched and Gray Images.

6.2 Quantitative Results

Quantitative results offer insights into the models' performance across various scenarios. The primary metrics include accuracy, mIoU, and class-specific IoU scores. The following observations emerge from the quantitative results:

- Among the resized models, "Resized_SingleUNet_color_dropout" achieved the highest accuracy (95.73%) and mIoU (85.56%), with the best IoU for class 1 (ship) at 92.62%.
- In the patched models, "Patched_SingleUNet_color_dropout" demonstrated superior performance, with an accuracy of 95.73%, an mIoU of 77.92%, and an impressive IoU for class 1 (ship) at 82.57%.

In general, models using dropout regularization consistently outperformed their counterparts.

6.3 Qualitative Results

Qualitative results, although not explicitly quantified, are a visual representation of the models' segmentation quality. The models exhibit their segmentation capabilities through image visualizations, enabling the identification of potential strengths and weaknesses. However, it is important to note that these results are subjective and primarily serve to complement the quantitative findings.

a)

b)

c)

d)

Figure 4. Test images [10][11][12][13].

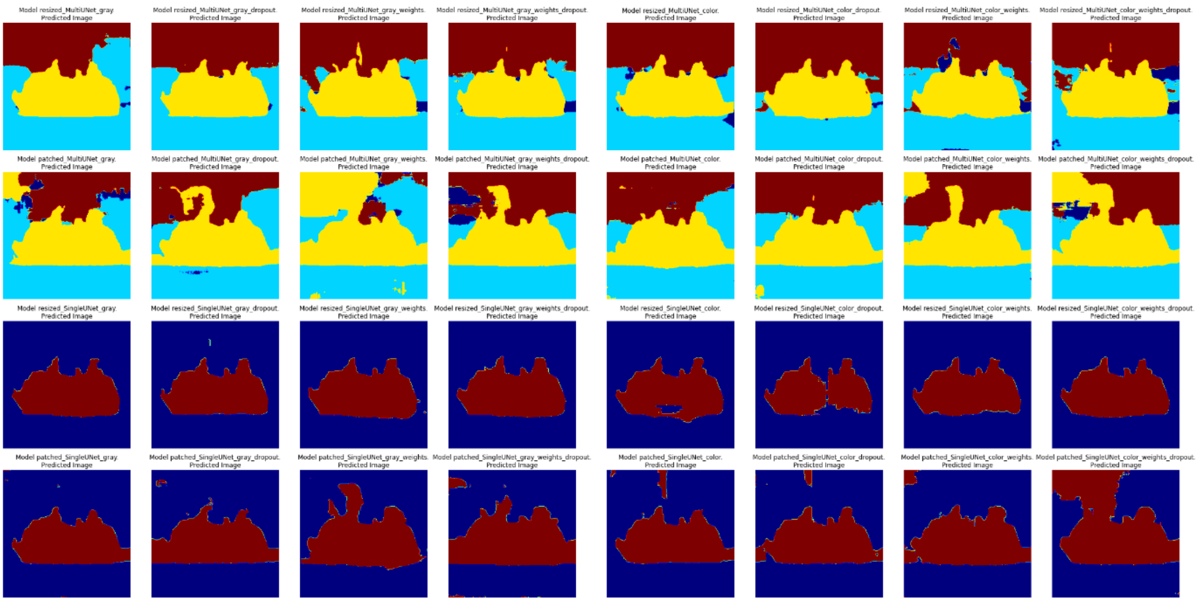

Figure 5. Results of Models on the Test Image a).

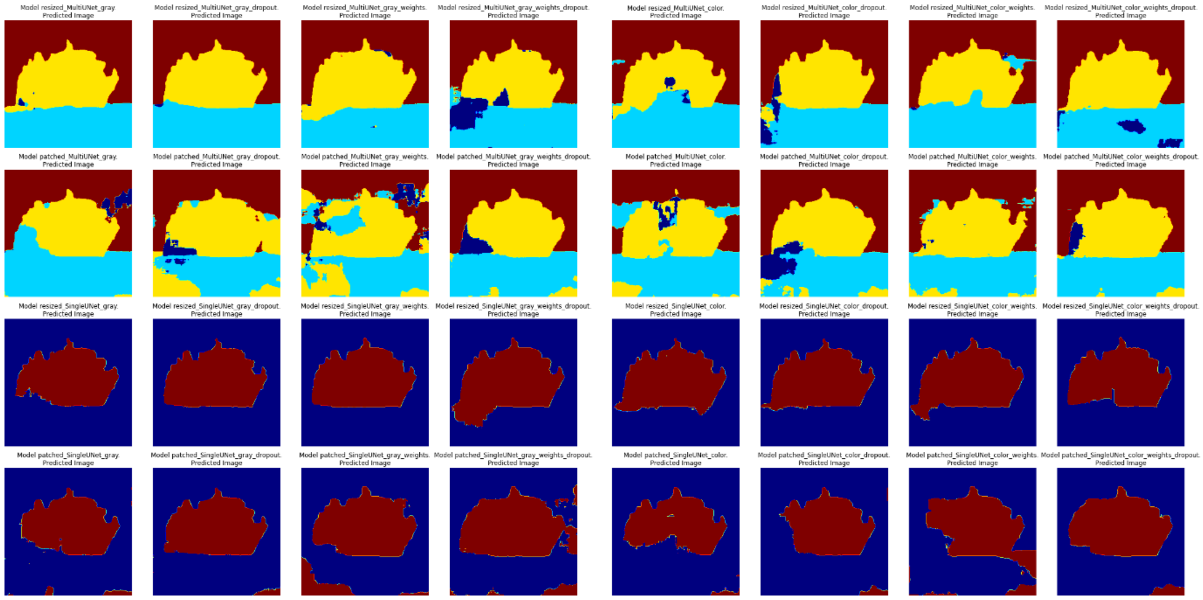

Figure 6. Results of Models on the Test Image b).

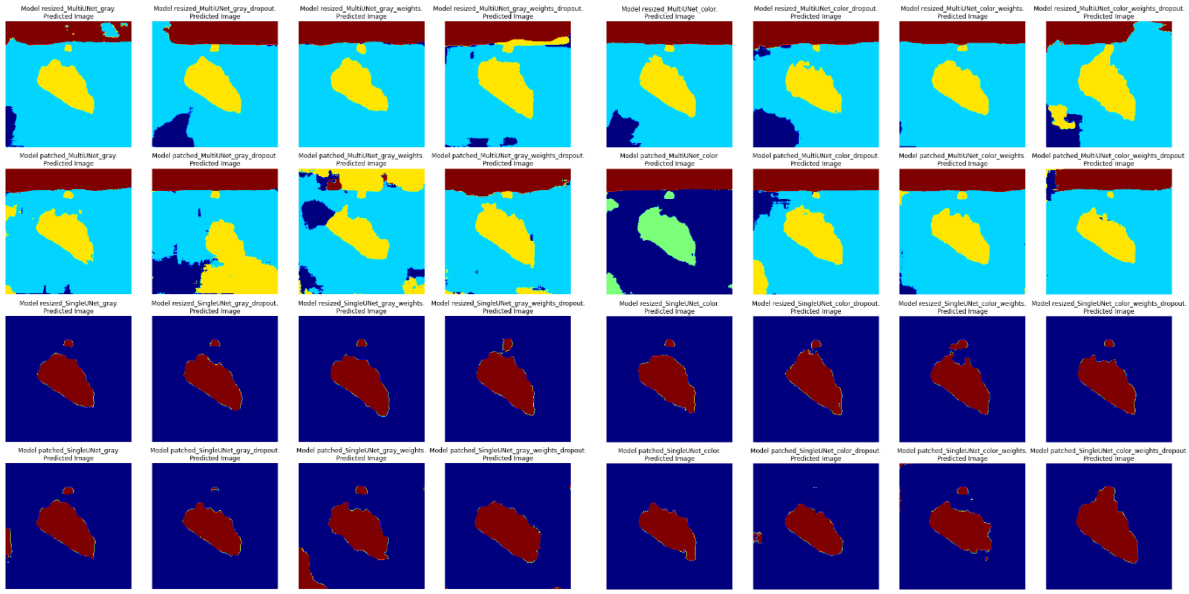

Figure 7. Results of Models on the Test Image c).

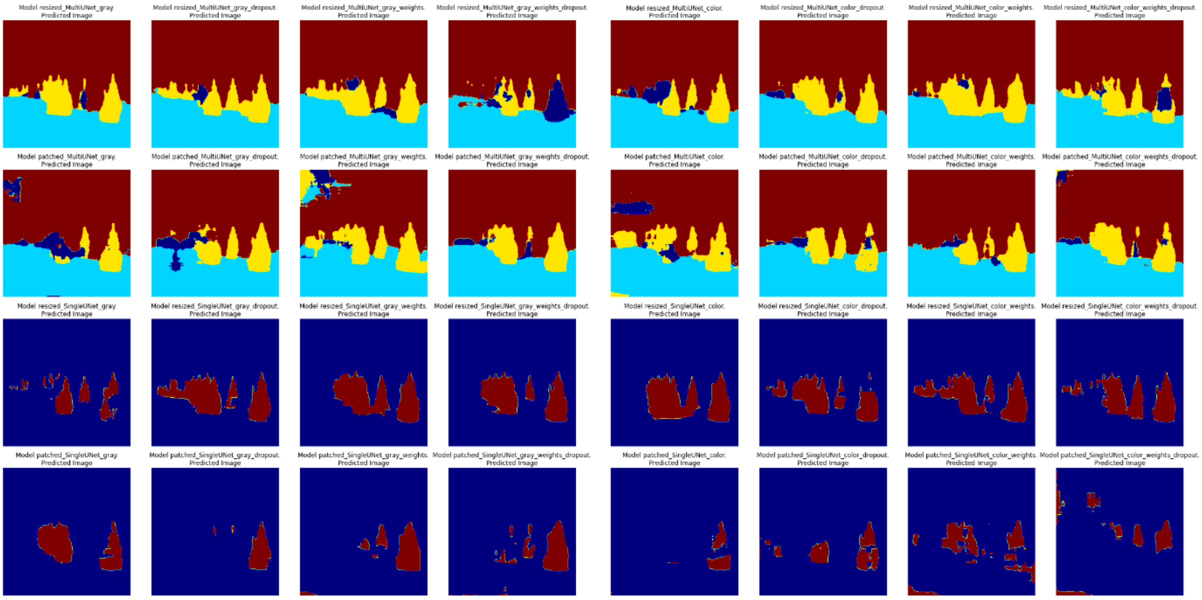

Figure 8. Results of Models on the Test Image d).

Figure 4 comprises four distinct images (labeled *a, b, c,* and *d*) used for testing predictions generated by all models. In Figure 5, the predictions for image *a* from Figure 4 were presented. Notably, high accuracy in ship segmentation was achieved by the single-class UNet models, with a noteworthy observation that the results were more favorable in the Resized models than the Patched models. Furthermore, a comparison between color and gray single-class UNet models indicated that the latter demonstrated superior performance, particularly in terms of reduced False Negative segmented pixels. However, it is essential to highlight that color single-class UNet models exhibited comparatively higher False Negative rates.

In Figure 6, the prediction results for image *b* from Figure 4 were displayed. Similar to Figure 5, single-class UNet models demonstrated remarkable accuracy in ship segmentation. However, it was observed that Patched models exhibited both higher False Negative and False Positive segmented pixels compared to the Resized models. Furthermore, in the case of multi-class UNet models, False Positives were notably higher than those in the Resized models. Surprisingly, gray images demonstrated superior performance when compared to color images.

Figure 7 showcased the prediction results for image *c* from Figure 4. Remarkably, the results were consistent with those observed in previous images. Single-class UNet models displayed a high degree of accuracy in segmenting ships within the image. However, it's worth noting that gray images exhibited reduced accuracy, particularly in the context of multi-class models and Patched images. Additionally, in the multi-class models, many models demonstrated the ability to distinguish between the sky and sea borders, although minor occurrences of False Positives segmented pixels were observed.

In Figure 8, the prediction results for the final image, image *d* from Figure 4, were presented. Within the context of gray images, the single-class UNet models faced challenges in segmenting distant ships. However, color images, especially when employing the Resized method, showcased the capability to segment these ships effectively, albeit with a few instances of False Negatives. In contrast, multi-class UNet models yielded varying results, with color images demonstrating superiority over gray images.

6.4 Model Comparison

Comparative analysis is essential to ascertain the efficacy of different strategies and configurations. The model comparison reveals the following:

- "Patched" models consistently outperformed their "Resized" counterparts in terms of accuracy, mIoU, and class-specific IoU scores; however, in the visualization of results, better results were observed in resized images.
- Models utilizing dropout regularization consistently achieved superior performance, particularly in the "SingleUNet" category.
- Color images generally outperformed grayscale images, aligning with the notion that color information provides richer cues for semantic segmentation.

6.5 Computational Complexity Analysis

The computational complexity analysis examines the runtime of each model and strategy. Key observations include:

- "Patched" models exhibit longer runtimes than "Resized" models due to the increased number of patches and higher computational load.
- Models with dropout regularization tend to require slightly more time for training and inference, as dropout introduces additional computation.

7 Discussion

7.1 Model Performance

The results reveal a spectrum of accuracy levels across different model variants, suggesting that model architecture and data preprocessing strategies significantly impact performance. The "Single Class UNet" models, both resized and patched, exhibit distinctive behavior concerning their segmentation accuracy. Notably, the "Resized Single Class UNet" models showcase superior performance compared to their patched counterparts. Similarly, the "Multi Class UNet" models, whether in color or grayscale, provide varying degrees of accuracy, with the color variants consistently outperforming their grayscale counterparts.

7.2 Impact of Resizing vs. Patching

The choice between resizing and patching images for semantic segmentation is a pivotal factor in model performance. Resizing enables the models to work with entire images, offering computational efficiency but potentially compromising finer details. In contrast, patching provides a comprehensive understanding of the image by analyzing smaller fragments, albeit at the cost of increased computational complexity.

In this project, it has been observed that employing resized images yields superior results in comparison to patched images. The outcomes indicate that the use of resized images consistently leads to better overall performance in multiple aspects of the analysis.

7.3 Effect of Color vs. Greyscale

The consideration of color information is another critical point of discussion. The project's results consistently indicate that color images enhance segmentation accuracy, as seen in both "Single Class UNet" and "Multi Class Unet" models. There was no high difference between color and greyscale image accuracy, however, color images have better accuracy results.

7.4 Regularization Techniques

Regularization techniques such as dropout have been applied to the models to prevent overfitting and improve generalization. Most of the models, which use dropout have slightly higher accuracy than other techniques in the pack. This observation underscores the notion that the utilization of dropout during training appears to be associated with the attainment of superior results within the project.

7.5 Class Weights and Imbalanced Data

The assignment of class weights and the treatment of imbalanced data is a vital aspect of the project. Models which use class weights while training have slightly lower accuracy than other models in the pack.

In particular, it has been noted that the models 'patched_MultiUNet_gray_weights' and 'patched_MultiUNet_gray_weights_dropout' consistently yield lower accuracy compared to other methods. Consequently, it has been observed that refraining from using weights during training appears to produce more favorable results in this project.

8 Conclusion

8.1 Summary of Findings

The research presented in this report revolves around the domain of maritime semantic segmentation, with a focus on the utilization of the UNet architecture. The findings of this study offer valuable insights into the various strategies and techniques employed to enhance the accuracy of segmentation in maritime scenes. Through extensive experimentation and analysis, the study uncovered significant variations in model performance across different model variants, shedding light on the interplay between data preprocessing, architecture, and regularization techniques. The results reveal that "Resized Multi Class UNet" models, particularly the color variants, exhibit outstanding performance, emphasizing the importance of fine-grained analysis. Furthermore, the study highlights the significant role of color information in improving segmentation accuracy, thus underscoring the relevance of color cues in maritime scene understanding.

The most optimal outcomes were consistently observed in UNet models integrated with dropout techniques. Furthermore, standard UNet models, both in configurations without dropout and without weights yielded commendable results. Hence, the adoption of 'resized_MultiUNet_color' and 'resized_MultiUNet_color_dropout' models is recommended as they are expected to yield improved outcomes.

8.2 Contributions

This research contributes to the field of maritime semantic segmentation by providing a comprehensive evaluation of UNet-based models across a range of data preprocessing and model architecture scenarios. The work showcases the potential of patch-based segmentation strategies, offering enhanced performance, albeit with increased computational complexity. Additionally, it emphasizes the role of color imagery in achieving accurate semantic segmentation, recognizing the importance of color cues in maritime image analysis. The application of regularization techniques, such as dropout, is also explored, offering insights into their effectiveness in preventing overfitting.

8.3 Limitations

While this research has made significant strides in enhancing maritime semantic segmentation, it is not without limitations. One of the constraints of this study is the limited size of the datasets, which may affect the generalizability of the findings. Additionally, the project mainly focuses on the UNet architecture, leaving room for future exploration of alternative architectures and approaches. The evaluation of class weights and regularization techniques, while informative, may benefit from further optimization for specific maritime segmentation tasks.

8.4 Future Work

The research presented here opens the door to several avenues for future work. To address the limitations related to dataset size and diversity, future research may explore the acquisition of larger and more diverse maritime datasets. Additionally, investigations into alternative architectures and advanced machine learning techniques could offer valuable insights into further improving segmentation accuracy. Future work may also delve deeper into the optimization of class weights and regularization techniques to tailor models for specific maritime scenarios, ultimately enhancing their robustness and reliability.

In summary, this research contributes to the advancement of maritime semantic segmentation by comprehensively evaluating UNet-based models and offering a nuanced understanding of the factors that influence segmentation accuracy. The findings presented here serve as a foundation for future research and development in the field, aiming to enhance the effectiveness of maritime scene understanding and related applications.

9 References

[1] B. Lehmann, "Chapter 6. Image Segmentation," in Image Processing & Pattern Recognition, Nov. 6, 2023

[2] R. C. Gonzalez and R. E. Woods, Digital Image Processing. Upper Saddle River: Pearson, 2019.

[3] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," arXiv.org, https://arxiv.org/abs/1703.06870 (accessed Nov. 6, 2023).

[4] Improving learning effectiveness for object detection and Classification in Cluttered Backgrounds, https://ieeexplore.ieee.org/document/8999695/ (accessed Nov. 6, 2023).

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," arXiv.org, https://arxiv.org/abs/1411.4038 (accessed Nov. 6, 2023).

[6] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," arXiv.org, https://arxiv.org/abs/1505.04597 (accessed Nov. 6, 2023).

[7] Z. Sun, C. Meng, T. Huang, Z. Zhang, and S. Chang, "Marine ship instance segmentation by deep neural networks using a global and local attention (gala) mechanism," PLOS ONE, https://journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0279248 (accessed Nov. 6, 2023).

[8] B. Zhou et al., "Semantic understanding of scenes through the Ade20k dataset," International Journal of Computer Vision, vol. 127, no. 3, pp. 302–321, 2018. doi:10.1007/s11263-018-1140-0

[9] B. Zhou et al., "Scene parsing through ade20k dataset," IEEE Xplore, https://ieeexplore.ieee.org/document/8100027 (accessed Nov. 6, 2023).

[10] Flory, “Princeps cruise ship off the island Bora Bora in French Polynesia in...,” iStock, https://www.istockphoto.com/de/foto/kreuzfahrt-in-polynesien-gm458665285-17826513 (accessed Nov. 6, 2023).

[11] Q. Zaman, “Can ships use magnets to move in water?,” EDUInput, https://eduinput.com/can-ships-use-magnets-to-move-in-water/ (accessed Nov. 6, 2023).

[12] M. Adams, “Viking expeditions - Summer 2023 season - Sparkx - Cruise & Travel,” Sparkx, https://sparkx.org/13732-2/ (accessed Nov. 6, 2023).

[13] S. Böke, “İşte Dev Geçidin şifreleri,” Hürriyet, https://www.hurriyet.com.tr/gundem/iste-dev-gecidin-sifreleri-42354925 (accessed Nov. 6, 2023).

10 Appendices

All the source code, result images, model files, as well as loss and accuracy graphs generated during the course of this research project are made available in the following GitHub repository:

GitHub Repository Link: https://github.com/MammadliN/Semantic-Segmentation-of-Watercrafts-using-UNet-Model

This repository serves as a comprehensive resource for replicating and further exploring the experiments and results presented in this report. Researchers and practitioners interested in the details of the implementation and outcomes are encouraged to refer to this repository for in-depth information and access to the project materials.