AN2DL - Second Homework Report
NeuralNinjas

David Rojas, Felix Rojas, Subhadip Banerjee, Satvik Bisht

davidfrojasc, felixrojasc, subhadipbanerjee09, satvikbisht2602

274685, 275030, 271077, 223892

February 4, 2025

1 Introduction

This project focuses on training a segmentation model for analyzing 64x128 grayscale images of Mars terrain. The objective is to classify each pixel into one of five distinct classes, making this a multi-class semantic segmentation problem.
The key objectives of this project are:

- Develop a robust deep learning model for semantic segmentation.
- Enhance model performance using appropriate preprocessing techniques and data augmentation.
- Fine-tune the model to mitigate overfitting and achieve optimal performance.
- Detect and handle class imbalance or outliers through data inspection.
- Split the dataset into training and validation subsets for generalization testing and hyper-parameter tuning.
- Implement overfitting prevention strategies like early stopping and dropout layers.
- Train the best-optimized model on the full dataset to make final predictions for unseen data.

2 Dataset Characteristics

The dataset consists of training and test images with the following specifications:

- Training images shape: $64 \times 128 \times 1$.
- Five unique classes for pixel labels.

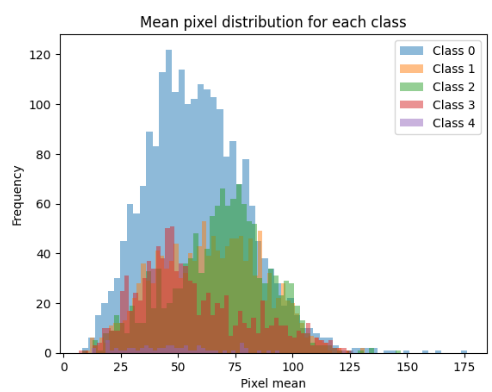

Figure 1: Mean pixel distribution in the dataset

3 Methodology

3.1 Data Preprocessing

When plotting images, we noticed that the dataset had some aliens, when we deleted those images our final cleaned dataset contained 2,504 images.

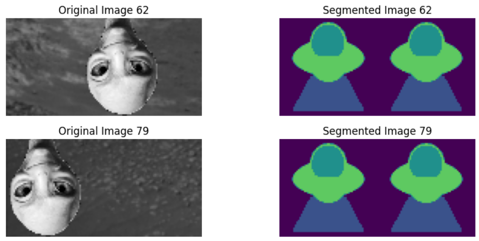

Figure 2: Aliens with their masks

We also implemented the following steps:

- Early stopping to prevent overfitting.
- Outlier removal using mean pixel values.
- Normalization to scale pixel values to $[0, 1]$.
- Adding a channel dimension for model compatibility.

3.2 Data Augmentation

We performed data augmentation for all the images in the dataset. We used random horizontal flips, and making sure that this augmentation was done to both the image and the mask. This improved the model's ability to generalize and learn features from data to correctly associate them to classes.

3.3 Loss Function

We utilized Cross Entropy Loss as our primary loss function, which ignores the background class to focus on meaningful segments and automatically handles the multi-class nature of the segmentation task. Also a built-in softmax activation is applied before computing the loss.

The loss function works in conjunction with the MulticlassJaccardIndex [2] metric:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

This monitors segmentation quality during training specifically for IoU (Intersection over Union), and to weigh rare and common classes equally since each class IoU is computed independently before averaging.

3.4 Model Architecture

We used a UNet architecture, which follows a symmetric encoder-decoder structure with skip connections. The encoder path consists of four downsampling blocks, each containing dual convolutional layers with 32, 64, 128, and 256 filters respectively. Each convolution is followed by batch normalization and ReLU activation. There is also a bottleneck layer which works lowest resolution with 512 filters, creating a better feature representation. The decoder and encoder have four upsampling blocks, using transposed convolutions to restore spatial dimensions.

The network employs the following optimization techniques:

- Batch normalization for stable training.
- Early stopping with a patience of 30 epochs.

4 Experiments

Table 1: Comparison of different models. Best results are highlighted in **bold**.

Method	Val. IOU%
Focal Loss - AdamW	39.51
Dice Loss - AdamW	42.98
CrossEntropy Loss - AdamW	43.89
CrossEntropy Ignoring Index 0 - SGD	62.09
Background Neural Network - AdamW	63.66
CrossEntropy Ignoring Index 0 - AdamW	**64.39**

We conducted a series of experiments to evaluate model performance under different configurations. Our primary focus was on optimizing loss functions, augmentation techniques, and architectural enhancements.

We tested various optimizers (Adam, AdamW, SGD), schedulers (ReduceLROnPlateau, OneCycleLR), model architectures and loss functions (CrossEntropy Loss, Dice Loss, and Focal Loss). The best loss function in our case was CrossEntropy, yielding validation IoU scores approximately 5% higher.

Data augmentation played a key role in model generalization. We utilized the Albumentations library [1] to introduce vertical flips, random rotations, elastic transformations, and grid distortions. Surprisingly, more complex augmentations did not yield performance gains, with validation IoU dropping by about 4%. This suggests that certain transformations might introduce unnecessary variability or noise, which negatively affects learning.

We also implemented gated skip connections and deep supervision for grayscale images. These modifications aimed to strengthen feature propagation and improve convergence.

A significant boost in Intersection over Union (IoU) scores was achieved by introducing `ignore_index=0` in the CrossEntropy Loss and multiclass Jaccard Loss functions. This adjustment ensured that the background class (label 0) was excluded from loss calculations and evaluation metrics. As a result, performance evaluations focused solely on the foreground classes, leading to more accurate and meaningful IoU scores.

5 Results

Finally, we trained the best optimized model with all the data with the number of epochs where the model does not overfit as we acknowledged using early stopping technique.
Key performance metrics:

- Training loss: 0.019, IoU: 84.13%.
- Validation IoU: 69.63%.
- Kaggle IoU: 64.615%.

6 Discussion

6.1 Strengths

- Deep Learning is a highly effective approach for solving image segmentation tasks due to its ability to learn complex spatial hierarchies.
- The U-Net architecture captures spatial information efficiently, enabling precise pixel-level segmentation.
- The use of data augmentation improved model generalization, enhancing performance on unseen data.
- The combination of early stopping and batch normalization contributed to more stable training and reduced overfitting.

6.2 Limitations

- Limited GPU resources constrained our ability to experiment with larger models, more complex architectures, and longer training times.
- Excessive data augmentation may have introduced noise, potentially decreasing performance.
- The exclusion of the background class, while improving IoU, could have led to minor misclassifications in edge cases where background pixels shared features with target classes.
- The fixed size of the input images (64x128) limited the ability to capture larger spatial patterns that could exist in higher-resolution images.

7 Conclusions

- The U-Net architecture demonstrated robust performance in image segmentation, achieving high accuracy and effective spatial feature learning.
- Strategic use of data augmentation and hyperparameter tuning were instrumental in achieving competitive results, improving the model's robustness to unseen data.
- Excluding the background class from loss calculations and evaluation metrics significantly improved Intersection over Union (IoU) scores, highlighting the importance of tailored loss function design.
- Future work could explore the impact of larger image sizes, more computational resources, and advanced architectural variations to further boost performance.

7.1 Contributions

David Rojas: Code and report; Felix Rojas: Code and report; Subhadip Banerjee: Code and Report draft; Satvik Bisht: Model testing.

References

[1] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin. Albumentations: fast and flexible image augmentations. *Information*, 11(2):125, 2020.

[2] Lightning AI. Jaccard index - torchmetrics documentation, 2024.