AN2DL - Second Homework Report
GLM2

Giulio Tamburini, Luca Bonini, Mirko Manset, Mario Russo

giuliot, lucabonini, mirkomanset, russomarioxyz

252427, 247808, 232142, 226591

December 14, 2024

1 Introduction

In this report, we will describe the techniques used to develop a neural network designed to address a semantic segmentation problem.

The dataset for this project consists of real images of Martian terrain, each paired with a corresponding mask that assigns a class label to every pixel. Specifically, each pixel is classified into one of the following five categories: *Background*, *Soil*, *Bedrock*, *Sand* or *Big Rock*.

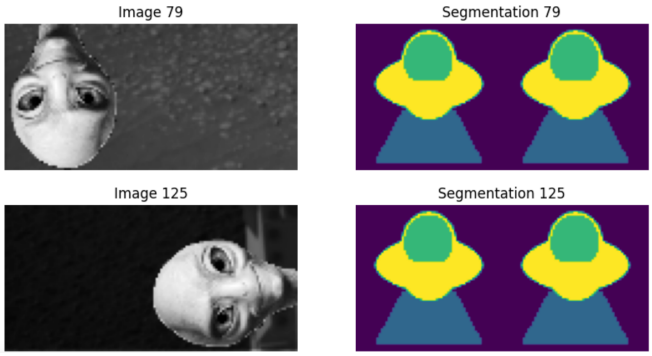

Figure 1: Outliers examples

2 Problem Analysis

2.1 Dataset characteristics

The training dataset comprises over 2,500 grayscale images, each with dimensions of 64x128 pixels. During the initial inspection phase, we identified several **outliers**—images that were corrupted by the presence of an alien figure. Although these corrupted images differed visually, their corresponding segmentation masks were all the same. To address this issue, we developed a function to automatically remove all samples with identical segmentation masks from the dataset. This cleaning process ensured that the dataset no longer included corrupted images. The cleaned dataset was then saved to avoid reprocessing the outliers each time the notebook is used.

After the data cleansing process, we split the **cleaned dataset** into training and validation sets with an 85/15 ratio.

2.2 Main challenges

We believe that the primary challenge of this project was designing an appropriate model to address the complex task of semantic segmentation. Furthermore, the scarcity of training data posed a significant issue, which we attempted to mitigate by applying data augmentation strategies (see 3.1) designed to augment not only the images but also their corresponding masks.

Numerous experiments were conducted to find the best combination of techniques and configurations. However, the training process for these models was particularly challenging due to the **lim-**

ited availability of GPU and RAM resources, which caused delays in the development process.

3 Model development

Starting from a basic *U-Net* model, several attempts on modification of its layers and general architecture were performed. Furthermore many different losses were considered to evaluate the performance during the training.

3.1 Data augmentation

In order to improve the performances it was used an **online image transformation approach**. By using the *Albumentation* library each sample and the respective segmentation mask belonging to the cleaned dataset underwent to a series of geometrical and elastic transformations giving back 3 different modified samples. The augmentation pipeline was composed by: a *OneOf* among the **geometrical transformations** such as *RandomCrop*, *Rotate*, *Translate*, *HorizontalFlip*, *VerticalFlip*, a second *OneOf* among **color-based transformations** like *RandomBrightnessContrast* and *HueSaturationValue*, ending with a moderate intensity **elastic distortion** with *ElasticTransform*. In order to cope with the class imbalance of the class *Big Rock* it was done an **oversampling** of the images containing that class, by increasing by a *oversampling factor* the number of augmentations performed on that specific images.

3.2 Model definition

The classical *U-Net* consists of a contracting path (encoder) and an expanding path (decoder). The encoder extracts hierarchical features through convolutional and max pooling layers, while the decoder upsamples the features and combines them with corresponding features from the encoder.

One of main building block of the *U-Net* is the *U-Net block* which typically consists of a series of convolutional layers followed by a normalization layer and an activation function. It is used both in downsampling and upsampling paths.

In our experiments we tried to enhance this specific building block trying different **normalization layers** such Batch Normalization, Layer Normalization and Group Normalization to help to stabilize the training process.

We also incorporated **residual connections** to enhance training stability and prevent overfitting. These connections allow the model to learn residual mappings, which can help to accelerate training and improve generalization.

Finally we introduce **Squeeze-and-Excitation mechanism** that enables the network to adaptively weight the importance of different feature channels and emphasize the relevant ones improving the network's ability to capture discriminative information.

To further enhance the performance of the *U-Net*, we explored various architectural modifications. We experimented with different layer configurations in both the downsampling and upsampling paths, including the use of dilated convolutions and transposed convolutions for upsampling. Additionally, we investigated the impact of varying the depth and width of the network on the overall performance.

To improve the model's ability to focus on relevant features, we also incorporated **attention mechanisms** into the *U-Net* architecture. Specifically, we employed attention gates, which allow the network to selectively focus on the most informative regions of the feature maps. This mechanism helps the model to better capture fine-grained details and reduce the impact of noise.

Additionally, we applied **regularization techniques** such as L1-L2 regularization and dropout to the network's weights.

We also investigated the potential benefits of a ***dual U-Net*** architecture, which combines a coarse network with a finer network. The coarse network provides a rough segmentation map, while the finer network refines the details. We explored different strategies for combining the outputs of the two networks, including concatenation, weighted summation, and attention-based fusion. However, our experiments did not yield significant performance improvements.

3.3 Losses

The choice of the loss function proved to be a critical aspect of the training process, significantly influencing the model's performance. Several loss functions, as well as combinations of them, were tested to address the specific challenges of the semantic segmentation task.

The first loss used was the *SparseCategorical-CrossEntropy*, chosen for its ability to optimize pixel-wise classification and establish a strong baseline. However, this loss struggled with imbalanced datasets, prompting the exploration of other approaches. The *Focal Loss* was utilized to address this issue by focusing on harder-to-classify pixels, reducing the impact of well-classified regions and improving model performance on minority classes. The *Dice Loss* was employed to optimize for overlap between predicted masks and ground truth, making it particularly effective for tasks with imbalanced class distributions. A combination of *Dice Loss* and *Focal Loss* was also tested, leveraging their complementary strengths to enhance segmentation accuracy. Finally, the *Focal Loss Tversky* was used to handle extreme class imbalance by prioritizing hard-to-predict regions and providing fine-grained control over false positives and false negatives, making it well-suited for the task at hand.

4 Results

In this section are reported the most relevant models with respect to their architecture and the loss used.

Table 1: Experiments on models, validation and test Mean Intersection Over Union achieved

Model	Validation MIOU	Test MIOU
Basic U-Net	42,61%	42,19%
Dual U-Net aug	46,19%	44,72%
U-Net D-Aug	50,45%	49,43%
Final Model	51,05%	50,28%

- **Basic U-Net**: single U-Net of depth 2
- **Dual U-Net aug**: dual U-net (depth 2 + depth 4), static augmentation, focal loss
- **U-Net D-aug**: single U-net (depth 3), dynamic augmentation, focal loss
- **Final Model**: single U-net (depth 4), dynamic augmentation, oversampling minority class, focal Tversky loss, enhanced U-net block (the one described in the model definition section).

Other models with different configurations and loss functions were also tested during this study. However, their performance metrics were consistently below those achieved by the models presented above, reaffirming the efficacy of the chosen architectures and training strategies.

5 Conclusions

In conclusion, this project tackled the challenging task of semantic segmentation on Martian terrain, addressing various types of issues. Despite these challenges, we developed a neural network and implemented a range of techniques to enhance the model's performance and robustness.

As part of our efforts to further enhance the model's performance, we integrated a **hyperparameter optimization** framework using **Optuna**. Unfortunately, due to time and computational cost constraints, we were unable to wait for sufficient iterations to achieve a reliable and comprehensive result. Future work could build on this foundation by completing the optimization process and exploring the potential improvements it could bring to the model's performances.

6 Contributions

- **Giulio Tamburini**: Data cleaning process; Development and training of the basic U-Net and dual model; hyperparameter optimization; contribution to the report.
- **Luca Bonini**: Training and development of unets and dual unets variants; Improvements of the unet block; Implementation of the Focal loss Tversky; contribution to the report.
- **Mirko Manset**: Training and development of unets and dual unets variants; Implementation of Focal Loss, Dice Loss and Combined Loss; contribution to the report.
- **Mario Russo**: Implementation of the dynamic data augmentation approach; Training and development of unets and dual unets variants; contribution to the report.