POLITECNICO
MILANO 1863

AN2DL - Second Homework Report
The Backpropagators

Arianna Procaccio, Francesco Buccoliero, Kai-Xi Matteo Chen, Luca Capoferri

ariii123, frbuccoliero, kaiximatteoc, lucaacapoferri

246843, 245498, 245523, 259617

December 14, 2024

This is a report of the second homework for the course **Artificial Neural Network and Deep Learning** at Politecnico di Milano.[1]

The focus of this homework is to correctly assign to each pixel of a given **Mars terrain grayscale image** one of five labels, representing different types of terrain. This is a *Semantic Segmentation* problem. No pretrained models were allowed, hence all networks had to be trained from scratch.

1 Problem Analysis

The dataset provided contains 2615 **training examples**, each comprised of a $(64, 128, 1)$ grayscale image and a corresponding ground truth mask with labels in $\{0, 1, 2, 3, 4\}$, where **0 is background** and the other 4 represent different Mars terrain classes (soil, bedrock, sand, big rock). A separate test set of 10022 images (no masks) was provided for submission.

Just like H1, following a quick inspection of images and masks, we noticed the presence of some *outliers* containing images of aliens and completely mismatching labels. After identifying these outliers, we discarded them to avoid compromising the training process. See Fig.1

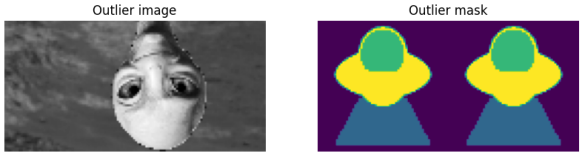

Figure 1: An outlier and its mask

The dataset presented a roughly even distribution among classes 0 to 3 but class 4 reportedly contained a very small amount of samples, accounting for just 0.13% of the total pixels. The evaluation metric for the leaderboard is the **mean Intersection over Union (mIoU)**, with the background class being ignored during evaluation, see Eq.1

$$\frac{1}{|C|} \sum_{c \in C} \frac{\mathbf{1}(y = c) \wedge \mathbf{1}(\hat{y} = c)}{\mathbf{1}(y = c) \vee \mathbf{1}(\hat{y} = c)} \tag{1}$$

2 Method

Our methodology maintained the principle of **modularity** as it was fundamental in the previous run to easily switch between architectures, optimizers, augmentation techniques, and training configurations.

We implemented a variety of segmentation architectures: standard **U-Nets**, variants with *Attention*, *Group Normalization*, *ASPP* blocks, and even

[1] `http://chrome.ws.dei.polimi.it/index.php?title=Artificial_Neural_Networks_and_Deep_Learning`

more experimental "transformer-like" modules. Despite the robustness promised by these approaches, we faced non negligible difficulties in achieving significant results with them: training from scratch without pretrained weights and with *limited computational resources* made it challenging to **properly tune hyperparameters** and to run extensive searches.

2.1 Preprocessing and augmentation

The data was cleaned from junk and split into train,validation and test set. Later we also developed a more advanced splitting technique that mantains a balance between *mostly-non-segmented* and *well-segmented* images, based on *expert-driven* metrics, see Eq.3

$$\text{disp(img)} = \text{std}\left(\frac{[n_0, n_1, n_2, n_3, n_4]}{\sum_{i=0}^{4} n_i}\right) \tag{2}$$

$$\text{type(img)} = \begin{cases} \text{Well-seg.} & disp(img) < 0.45 \\ \text{Non-seg.} & disp(img) \geq 0.45 \end{cases} \tag{3}$$

We tested various **data augmentation** pipelines (involving geometric transforms, flipping and elastic deformations). Contrary to H1, augmentation had a less dramatic effect, likely due to the simpler and more uniform nature of the dataset. Still, applying mild augmentation improved generalization slightly, especially when combined with original, unmodified data. Overly complex or *aggressive augmentations* did not yield significant benefits.

2.2 The model

We tried a variety of models:

- **Basic U-Net**[8] architectures: served as a strong baseline. Straightforward and relatively quick to train.
- **U-Net variants** (with ASPP[1], Attention[7], Transformer-like blocks, Group Normalization[10]): We integrated more complex layers and attention mechanisms. These aimed at improving feature capture of subtle terrain structures.
- **Heavier SOTA custom architectures**: Inspired by advanced literature, we attempted modular blocks combining dilated convolutions, windowed attention, and multi-branch feature fusion. Some examples include **RockSeg**[3], **Light4Mars**[11], and even a custom **TurkeySeg**, built trying to follow the suggestions from Prof.Lomurno's logbook.

Unfortunately, while conceptually promising, these more advanced architectures often failed to outperform simpler baselines. We believe that, with additional time, more extensive hyperparameter tuning, and careful initialization, their full potential could have been realized. As a result, we chose **U-NET-L** a U-NET model that employs group normalization, with regularization achieved through a combination of spatial dropout and an L2 weight regularizer.

2.3 Optimizers and Losses

We experimented with all the optimizers already seen in H1: **SGD**, **Adam**[4], **AdamW**[6],**Lion**[2], and even **Ranger**[9]. Ultimately, once again, we found that stable optimizers like Adam performed better.

In terms of loss functions, we extended our search beyond the classic *Sparse Categorical Crossentropy* by including *Dice*, *Focal* [5], and various combined loss strategies to better address class imbalance and improve segmentation boundary accuracy. Our experiments demonstrated that combining multiple loss functions led to improvements in performance.

Therefore, the loss function we adopted integrates **Dice**, **Focal**, **Crossentropy**, and **Boundary** losses, along with a penalty term specifically targeting errors in the prediction of the background class. This addition proved crucial, as the network struggled to learn the background class effectively. By explicitly penalizing these errors, the loss function guided the model toward better overall segmentation accuracy. Additionally, the loss calculation incorporates class weights derived from the class distribution, ensuring a balanced contribution from all classes during training.

Table 1: Sample results from different configurations. Best results are highlighted in **bold**.

Model	Train mIoU	Validation mIoU	Platform Test mIoU
Random predictor (expected)	-	-	0.0989
Basic U-Net	0.5921	0.5243	0.49276
U-Net + ASPP	0.6531	0.6252	0.58961
U-Net + Attention	0.6902	0.6381	0.59113
U-Net-L + Complex Aug	0.6513	0.6227	0.61386
U-Net-L (mild aug)	**0.7303**	**0.6817**	**0.6400**

3 Experiments and Results

We conducted many training experiments, focusing on different *model topologies* and *augmentation strategies*. Due to the limited time available we tried to insist on the configurations that were more promising, both in results and upon *further inspection of the predicted masks*, making the needed adjustments to try to improve the results.

From Table 1, we see:

- **No single "complex" architecture** managed to outperform the simpler baselines on platform test mIoU, as previously discussed. While these architectures theoretically have the potential for superior performance, our results consistently fell short of those expectations.

- **Mild augmentation** improved validation performance, though its impact was less pronounced compared to H1. In this case, the test dataset was already relatively clean, making lighter augmentation sufficient. The most notable improvements were achieved using *rigid deformations*. Mismatches between training, validation, test, and platform results were minimal. While we observed some local test improvements by fine-tuning the best models with images augmented through rigid transformations, these gains did not translate consistently to the platform test.

- Refining the **loss function** yielded more substantial improvements by directly addressing the dataset's class imbalance. This adjustment allowed the model to better represent minority classes, improving overall performance in a balanced and measurable way.

4 Conclusions

In this homework, we tackled a semantic segmentation task using Mars terrain images. Key takeaways:

- **Advanced architectures without proper tuning** do not guarantee better results. Simpler U-Nets performed similar to more complex networks due to insufficient fine-tuning.

- **Augmentation remains important**, though its impact was less pronounced compared to H1. Mild, targeted augmentations provided slight improvements in generalization.

- **Modularity proved valuable once again**, enabling us to seamlessly switch between models and strategies.

For future work, we plan to focus more on hyperparameter optimization for the advanced architectures and explore task-specific augmentation strategies. Additionally, investigating alternative regularization methods or incorporating pseudo-labeling techniques may help achieve state-of-the-art performance for this task.

The group division followed a similar structure to H1: **Luca** and **Arianna** focused on implementing complex state-of-the-art models and conducting most of the experiments, while **Matteo** and **Francesco** worked on the foundational aspects of the notebook, ensuring smooth progress for the entire team. Matteo once again served as the group leader, and Francesco wrote the backbone of the report. However, all team members contributed to the codebase, model experiments, and report writing, ensuring that everyone gained significant hands-on experience.

References

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, 2017.

[2] X. Chen, C. Liang, D. Huang, E. Real, K. Wang, Y. Liu, H. Pham, X. Dong, T. Luong, C.-J. Hsieh, Y. Lu, and Q. V. Le. Symbolic discovery of optimization algorithms, 2023.

[3] L. Fan, J. Yuan, X. Niu, K. Zha, and W. Ma. Rockseg: A novel semantic segmentation network based on a hybrid framework combining a convolutional neural network and transformer for deep space rock images. *Remote Sensing*, 15(16), 2023.

[4] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

[5] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection, 2018.

[6] I. Loshchilov and F. Hutter. Decoupled weight decay regularization, 2019.

[7] D. Pal, P. B. Reddy, and S. Roy. Attention uw-net: A fully connected model for automatic segmentation and annotation of chest x-ray. *Computers in Biology and Medicine*, 150:106083, 2022.

[8] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[9] L. Wright. Ranger - a synergistic optimizer. `https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer`, 2019.

[10] Y. Wu and K. He. Group normalization, 2018.

[11] Y. Xiong, X. Xiao, M. Yao, H. Cui, and Y. Fu. Light4mars: A lightweight transformer model for semantic segmentation on unstructured environment like mars. *ISPRS Journal of Photogrammetry and Remote Sensing*, 214:167–178, 2024.