AN2DL - Second Homework Report
ANNamo bene

Michelangelo Stasi, Nicolò Tocalli, Elia Pontiggia

Myke01, angelo ajax, pontig

250299, 247366, 247274

December 14, 2024

1 Introduction

This project involves the implementation of a fully-Convolutional Neural Network (CNN) for image segmentation. Contrary to the previous homework, this time the Neural Network has to be implemented completely from scratch.

2 Problem Analysis

The task of this project focuses on correctly classifying different types of terrains represented in images taken from the Mars drone Curiosity.

2.1 Dataset Description

The provided dataset contains around 2,600 images for the train set and more than 10,000 images for the test set. Each image is a grayscale image of 64x128 pixels. Every image in the test set is associated with a mask of the same size, where each pixel is labeled with a value from 0 to 4, representing the terrain type (0: Background, 1: Soil, 2: Bedrock, 3: Sand, 4: Big Rock)

2.2 Data Preprocessing

During an initial inspection of the training set, we identified and removed approximately 800 pairs of images and masks that were outliers. These images were straightforward to eliminate as they shared identical masks.

Regarding data augmentation, we employed a less aggressive strategy compared to the first challenge. Specifically, we applied basic geometric transformations (rotation, horizontal flipping, zooming, and shifting) to the images and masks. Additionally, we introduced random brightness adjustments to the images to enhance the model's robustness to varying lighting conditions.

3 Method

For this image segmentation problem, we had a wide range of options for the loss function. We initially chose the sparse categorical cross-entropy loss function, a common choice for multi-class classification problems. Subsequently, we incorporated an ensemble of other loss functions better suited to image segmentation tasks (see details later).

For the optimizer, we selected the AdamW [1] optimizer with its default learning rate of 0.001. Additionally, we implemented a learning rate scheduler to adjust the learning rate dynamically during training.

To mitigate overfitting, we incorporated dropout layers and applied early stopping during training. Moreover, each convolutional layer had a kernel regularizer with a value of 1e-4.

Given the computational demands of training, we decided against using k-fold cross-validation, despite its advantages for small datasets. Instead, all local results were evaluated on a single validation set, consisting of 300 images randomly selected from the training set

4 Experiments

4.1 Standard U-Net

All our models were built upon a U-Net backbone, a fully convolutional neural network widely used for image segmentation tasks.

We began with a simple U-Net model to establish a baseline for comparison. From there, we attempted to enhance the model by incorporating various blocks:

- Attention gates in the skip connections

- Residual blocks

- Dense connections between the encoder and the decoder

- Bottleneck enhancements: squeeze-and-excitation blocks, global context modules, and parallel dilated convolutions

- (Learnable) weighted feature fusion between the encoder and the decoder

- `Conv2DTranspose` instead of `UpSampling2D` in the decoder

- Multiscale blocks

- Deep supervision

For each added block, we trained the model with different depths of the "U" portion of the network and compared the results.

Additionally, we developed a custom loss function combining multiple loss functions. Specifically, we implemented Focal Loss, Dice Loss, Sparse Categorical Cross-Entropy Loss, and Boundary Loss. We experimented with different weight combinations for these loss functions during training.

4.2 Researched Architectures

After experimenting with various configurations, we discovered a paper describing a specialized U-Net architecture [2] that achieved excellent results on Medical Image Segmentation tasks. We implemented and trained this architecture on our dataset to evaluate its potential to improve our results.

We also explored another advanced architecture, the Multiscale U-Net [3], a U-Net variant initially applied to Medical Image Segmentation tasks. We implemented this architecture to assess its applicability to our problem.

The most relevant models and their performances are summarized in Table 4.2. (*Note*: Each model was trained multiple times with different hyperparameter configurations, and the table reflects the best performances achieved.)

Model	Validation	Test
Baseline, depth=3	.6130	.3510
Baseline, depth=4	.3725	.3576
+ LR scheduler and dropout	.5016	.4403
+ Attention gates	.5617	.4803
+ Residual blocks	.5682	.4848
+ Dense connections	.4321	.4176
− Dense connections		
Update loss with ignore 0	.7951	.6613
+ Bottleneck improvement		
+ Loss function ensemble	.8947	.7070
Bagging ensemble	n/a	.6300
+ Weighted feature fusion	.6335	.4709
− Weighted feature fusion		
U-net 3+ architecture	.8680	.7180
+ Post-processing	.8203	.7178
+ Multiscale	.8529	.6641
+ Deep supervision	.9122	.6796
Ensemble of the best models	n/a	.7387

5 Discussion

Similar to the first challenge, our models performed well on the validation set but showed slight overfitting, as evidenced by their lower performance on Kaggle compared to local machines. This was expected, given the absence of k-fold cross-validation. Nevertheless, the techniques we employed to prevent overfitting helped mitigate this issue.

Overall, we observed greater instability in the

models trained for this challenge compared to those from the first challenge. Minor changes in hyperparameters, such as the weights of the loss functions, often led to significant performance variations on both the validation and test sets. This instability can likely be attributed to the limited size of the dataset and the fact that our models were developed entirely from scratch, without leveraging pre-trained weights commonly used in transfer learning scenarios.

Interestingly, increasing the complexity of the models did not always result in better performance. For instance, the addition of dense connections to the U-Net architecture and the inclusion of squeeze-and-excitation blocks failed to improve performance. Consequently, we decided to exclude from the final model those blocks that did not contribute to performance enhancement.

In our initial experiments, we mistakenly computed the loss function across all classes, including the background class. This led to suboptimal results, as the model was minimizing loss for the background class, which was irrelevant for the mean Intersection over Union (IoU) computation. After excluding the background class from the loss computation, the results improved significantly.

For this task, data augmentation did not consistently enhance model performance. In some cases, models trained without data augmentation outperformed those trained with it. As a result, we opted for minimal geometric data augmentation and applied it only to models that demonstrated improved performance with augmentation.

We also experimented with a standard bagging ensemble for the U-Net model with an improved bottleneck—the only model feasible to train multiple times due to its low computational cost. However, the results were unsatisfactory, and we chose not to pursue this approach further.

A notable improvement in test set predictions was achieved through post-processing. Specifically, we applied Conditional Random Fields and morphological operations, such as dilation, which enhanced the final results.

To further improve the robustness of our predictions, we combined the outputs of the best-performing models through majority voting. This ensemble approach resulted in an improved score on the test set.

Other methods we explored in the final phases of the challenge that unfortunately did not yield sig-
nificant improvements included test-time augmentation based on the confidence of the model's predictions during inference, the use of deep supervision and the use of `Conv2DTranspose` instead of `UpSampling2D` in the decoder.

6 Conclusions

In this project, we implemented a fully convolutional neural network for image segmentation. The task presented several challenges, starting with the need to build the model from scratch without utilizing pre-trained weights. The limited size of the dataset added further complexity to the problem.

To address these challenges, we began with a simple U-Net model, progressively incorporating various blocks and analyzing their impact on the model's performance.

Through this process, we gained valuable insights into the critical role of selecting an appropriate loss function for image segmentation tasks. We experimented with different loss functions and their combinations, which significantly improved the model's performance.

Additionally, we learned that increasing model complexity does not always yield better results. In some cases, a simpler model outperformed more complex architectures, highlighting the importance of balancing complexity with effectiveness.

In the end, we managed to achieve a good score on the first split on the test set, and we also managed to improve the score by applying post-processing to the predictions and by merging the predictions of the best models we trained via majority voting. We cannot say anything about the generalization of our model, as we only tested it on the first split of the test set, but we are confident that our model is sufficiently robust to generalize to the other splits.

Contributions

Michelangelo: Base U-net, weighted feature fusion, experiments with augmentation, U-net3+, Multiscale, Deep supervision

Nicolò: Paper research, Ensemble and post processing.

Elia: Base U-net, LR scheduler, attention gates, residual blocks, custom loss, bottleneck improvements, bagging, experiments with augmentation.

References

[1] K. documentation `https://keras.io/api/optimizers/adamw/`. Adamw optimizers.

[2] H. Huang, L. Lin, R. Tong, H. Hu, Q. Zhang, Y. Iwamoto, X. Han, Y.-W. Chen, and J. Wu. Unet 3+: A full-scale connected unet for medical image segmentation. 2020.

[3] R. Su et al. Msu-net: Multi-scale u-net for 2d medical image segmentation. *Frontiers in Genetics*, 2021.