Comparison of Deep Neural Network Learning Algorithms for Mars Terrain Image Segmentation

Wojciech Wicki
SoftRocket,
Rumia, Poland *wickiwojtek@gmail.com*

Wiktor Burblis
Gdansk University of Technology,
Gdansk, Poland *wiktorbur@gmail.com*

Miłosz Tkaczeń
National Polytechnic Institute of Toulouse,
Toulouse, France *m.tkaczen@gmail.com*

Jerzy Demkowicz
Gdansk University of Technology
Gdansk, Poland *demjot@eti.pg.edu.pl*

Abstract

This paper is dedicated to the topic of terrain recognition on Mars using advanced techniques based on the convolutional neural networks (CNN). The work on the project was conducted based on the set of 18K images collected by the Curiosity, Opportunity and Spirit rovers. The data were later processed by the model operating in a Python environment, utilizing Keras and Tensorflow repositories. The model benefits from the pretrained backbones trained for analysis of the RGB images. The project achieves an accuracy of 83.5% when extending the scope of classification to unknown objects and 94.2% when omitting unknown results. The results were compared with related projects of Zooniverse and NASA's Jet Propulsion Laboratory scientific group. From amongst the evaluated configurations, the best results and resource utilization were achieved by applying the UNet architecture with resnext_50 backbone and Adam optimizer.

Keywords: Artificial Intelligence, machine learning, CNN, neural networks, optimization algorithms.

1. Introduction

Manual control over rovers on Mars presents a challenge of communication delays, which occur due to the vast distance between Earth and Mars, resulting in a significant signal lag time (3 to 22 minutes one way) to travel between the two planets [1]. This creates a demand for an effective object recognition tool to enhance the capabilities of navigation through diverse and unpredictable terrains, particularly the Deep Neural Networks (DNNs). Deep convolutional neural networks are often used for the task of image semantic segmentation, particularly network architectures like SegNet [2], U-Net [3], and DeepLab [4]. Some CNNs, tailored for semantic segmentation, predict multiple output labels (one for each pixel) within an image, namely U-Net and DeepLab. For the purposes of this project, the data were obtained from NASA's open access web archive [5]. The parameters of the AI4Mars dataset [5] consist of a total of 35,000 images, distributed into the chosen dataset (16,386), training dataset (12,851) and validation dataset (3,213). Training images were evaluated by around 10 individuals to ensure quality and consensus on the crowdsourced labels. Test images were annotated with labels by the rover planners and scientists affiliated with NASA's Mars Science Laboratory mission. Main drawback of the related approaches is that they are not trained with objects they have never seen and are not present in the training dataset. Our research focuses on preparing a solution that overcomes this issue by introducing a new class into which a pixel is classified as if it is not of any known class.

2. Methodology

In order to perform the task, the decision was made to use Python programming language together with GitHub version control. Main library used for neural network modeling is Keras. For image preprocessing, libraries pillow, numpy, tensorflow and pandas are used. Pretrained models were downloaded from the pretrained library segmentation_models and from tensorflow_examples. Visualization of the results, including printing of images and making statistics on the dataset, was made with the use of imageio and scikit-learn libraries. Gray-scale images of the planet Mars were matched with the pretrained backbones prepared for RGB images by applying a single convolutional layer with convolutions of size 1 by 1. Training set was split into the training and validation part in the ratio of 0.8. Categorical cross entropy was used as a loss function and a mean absolute error together with categorical accuracy were used as metrics. Detailed description of application in the form of code is available on git repository page[1].

3. Results

Throughout this experimental process, various network architectures were tested, hyperparameters were adjusted, and the impact of different factors on the model's quality was analyzed. Class weights were calculated based on the number of occurrences of a particular label on each pixel in the dataset. Weight is calculated as an inverse of the proportion of a particular class in the dataset divided by the number of occurrences.Weight for class null was lowered on purpose as we consider this class the least important. Based on used weights, versioning convention was used: v1 means no weights used, v2 uses calculated weights, v3 uses weights adjusted by us and other versions means other changes like changed optimizer or learning rate. Architectures for Platforms 1 and 2 were chosen based on related works in the field of image segmentation whereas Platform 3 was an exception as it was created from scratch. Architecture for each platform is shown below in Table 1.

Table 1. Comparison of models on the basis of their construction

Model name	Architecture	Backbone	Pretraining
Platform 1	Linknet	DenseNet-201	Yes
Platform 2	Unet	ResNeXt-50	Yes
Platform 3	Default	Default	No

Learning rate, batch size and optimizer were selected experimentally. The choice of Adagrad [6] and Adam [7] optimizers was supported by the fact that those optimizers offer the variable training rate throughout the training process. Imbalance of classes may result in models having high general accuracy, but still fail to predict certain classes as shown in the confusion matrix (Figure 2).

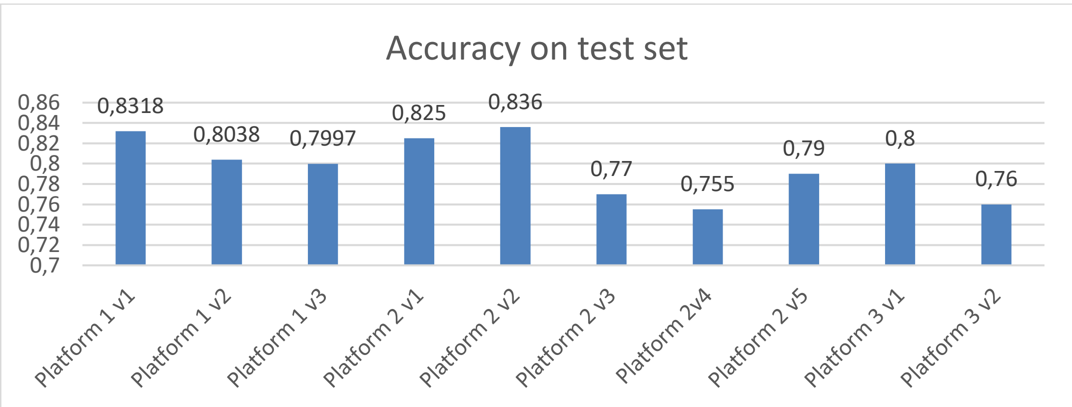

Fig. 1. Comparison of accuracies on test set for each model

[1] https://github.com/Lidbey/AI4Mars

Best performing model when assessing overall accuracy is Platform 2 v2. Achieved accuracy is 83.6% including NULL labels as to be classified (Figure 1). Imbalance of classes may result in models having high general accuracy, but still fail to predict certain classes, therefore confusion matrices were constructed (Figure 2). From these analyses, it was decided to choose two most promising models - one with a good overall accuracy (Figure 1) and one which correctly classified the biggest proportion of big rocks (Figure 2 on the right). The model that did perform better in big rock classification achieved an overall accuracy of 76% on the test set, which is considered acceptable, especially considering low resource requirements - this model is around 10 times smaller than the others. Big rock has shown much lower accuracy results than any other class due to a small representation of this label in the training data.

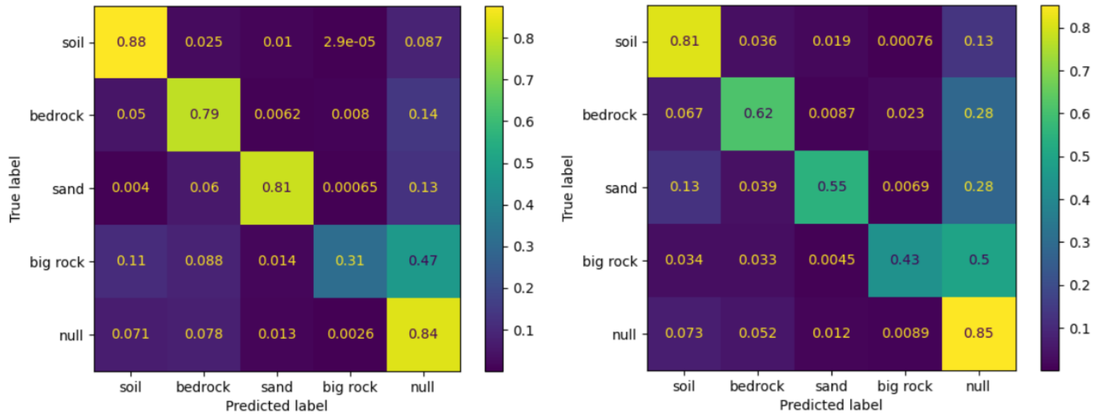

Fig. 2. Confusion matrix for platform 2 v2 (left) and for platform 3 v2 (right) models

This research focuses on the segmentation algorithms, but introduction of null class can be considered as a lack of detected object (negative example). Therefore other labels can be treated as positive examples. With data analyzed this way, more metrics like precision, recall, f1-score and accuracy within positive labels could be calculated (Table 2).

Table 2. Metrics of chosen models

Model name	Precision	Recall	F1	Accuracy within positive labels
Platform 2 v2	0.88	0.89	0.88	0.942
Platform 3 v2	0.88	0.78	0.83	0.878

In order to further test the results, a manual analysis was done on several photos. This allowed us to assess the performance independent of the provided dataset labels, which weren't perfect especially on the edges of classes. The results are satisfactory - predicted classes that differ from classes on the true label image are discussable because of the imperfections on the labels in the given dataset. Composition of Mars terrain image, true labels and predicted labels is shown at Figure 3.

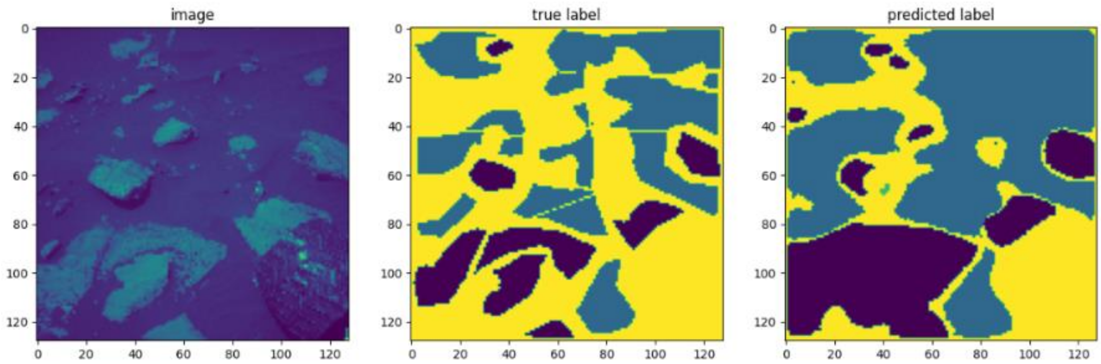

Fig. 3. Prediction of image NLB_537938493EDR_F0601266NCAM00653M1 using platform 2 v2 model

4. Discussion and Conclusions

Throughout the process of developing the neural network, Platform 2 v2 in particular gives the most promising results. It is utilizing the UNet architecture with a ResNeXt_50 backbone and exhibits a compelling balance between various metrics. The highest overall accuracy values (83.6%) and accuracy within positive samples (94.2%) on the test set is suggesting that the model accurately captures different features of the Mars terrain. Accuracy on big rock pixels for this model reached 31%. The only other model which was noteworthy is the one constructed from scratch (platform 3 v2), because of the lower hardware requirements and its ability to find the big rock class (43% of all big rock pixels were found correctly). Compared to the Zooniverse model SPOCv2 [8], both noted models had higher ability to find big rock class. SPOCv2 was able to find correctly only 13.33% of all big rock pixels but overall accuracy was higher (91.41%). NASA's JPL model based on ResNet_101 [5] had much higher overall accuracy (96.67%) and the big rock accuracy (93.24%) but it has higher hardware requirements (ResNeXt_50 is lighter weight architecture than ResNet_101). It's also noteworthy that the JPL had merged labels and used much more labeled data (40K images compared to 18K images). Additionally, the scope of the work included classifying an additional class - Null - which we find important in order to automatically classify different objects than those of a known class, which resulted in more complex tasks performed by trained models. One of the issues that was slowing down this process was high requirements of computing power, greatly extending the time required to finish the training of a single model. This paper may serve as a robust foundation for future work in the development of technologies related to Mars exploration. Paper shows that convolutional neural networks trained on image datasets publicly available can be used as a foundation for the rover navigation on different planets. It is also providing insights into possibilities of transfer learning from Earth datasets onto extraterrestrial environments.

Acknowledgements

We extend our thanks to a team that helped us develop the solution and use their resources to test more architectures. Special thanks to Julia Kalinowska, Marcin Bonk, Filip Smyk for comprehensive knowledge, assisting in the development of the solution and advice related to the work.

References

1. https://mars.nasa.gov/mars2020/spacecraft/rover/communications, Accessed 10.04.2024
2. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. CoRR, vol. abs/1511.00561 (2015)
3. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), ser. LNCS, vol. 9351. Springer, 2015, pp. 234–241, (available on arXiv:1505.04597 [cs.CV]).
4. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40 (4), 834–848 (2017)
5. Swan, R.M., Atha, D., Leopold, H.A., Gildner, M., Oij, S., Chiu, C., Ono, M.: AI4MARS: A dataset for terrain-aware autonomous driving on Mars. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1982-1991 (2021)
6. https://medium.com/konvergen/an-introduction-to-adagrad-f130ae871827, Accessed 09.12.2023
7. https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/, Accessed 19.10.2023
8. https://www.zooniverse.org/projects/hiro-ono/ai4mars/about/research, Accessed 20.06.2024