Semantic Segmentation of Storms



Team name: Cm_pi_jr

Chosen theme: Life on Earth

Organisation name: CMG a SOŠPg

Country: Czech Republic

1. Introduction

We called our experiment Semantic Segmentation of Storms, because that is a good fitting title for what we were actually doing. Our final experiment is slightly different from what you saw in the original plan, which we started with. That is because we were trying hard to classify all types of clouds in the satellite imagery - all kinds of clouds are completely different from the height of 400 km and from view above as well. We decided to change our idea a little bit. So (after consultation with Czech Hydrometeorological Institute - CHMI) we ended up just classifying storms as they are relatively easy to find. Regarding our motivation to do this, we were especially interested in machine learning and cool stuff around it, but we were also curious about where storms are created the most, if it is around the earth's equator or if it is in regions beyond the Arctic Circle. Our question was also: Does it depends on longitude or is the latitude the only thing that matters?

2. Method

As for the part which was done aboard the ISS, it was only about taking images in the visible spectrum with a period of 15 seconds. While we were waiting for our data from ISS, we were developing a neural network for semantic segmentation. This was the crucial part of our whole experiment and, as we were completely new in this topic, we tried hard to get it working.

Firstly, we needed to find a good and user-friendly library, we were choosing from two alternatives: TensorFlow and PyTorch. We ended up using TF as there are a lot of tutorials out there. We have tested a lot of different architectures, such as well-known U-Net, VGG16 from Tensorflow's example library, or different versions of EfficientNet and FCNs. However nothing was suitable for our project - we were getting all-black or all-white masks (white colour indicates storm, and black background). Finally we found U-Net with attention in this paper.

Currently our CNN is not very precise, but we will try to make it as good as we can in future.

3. Experiment results

As we said above the results are not accurate. The model is predicting a storm where it is not even a cloud, or contrariwise predicting a background where it is clearly a storm in its mature stage. The link for our model and tutorial on how to run on your machine is here (it is on disk google, because the model is too large). From beginning to end we were constantly reminding our-selves that we have to write something to this template, but unfortunately we end up with just a model and nothing else.

The most funny thing about this is the role of ChatGPT - we started with a pre-trained encoder from EfficentNet and we were trying to find a suitable decoder for it. As we are very good at programming, we fired up ChatGPT, we let it code the decoder for us and it worked just fine. However in expectation of better results, we delete it. The catch is that this was the best model that we have ever had and we have never even come close to its accuracy.



Figure 1 - predicted mask

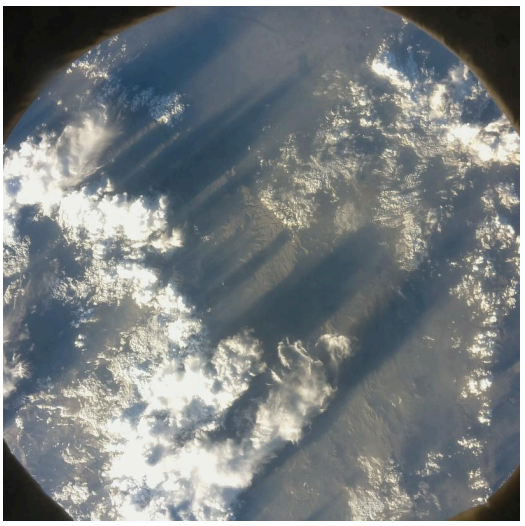


Figure 2 - image



Figure 3 - predicted mask

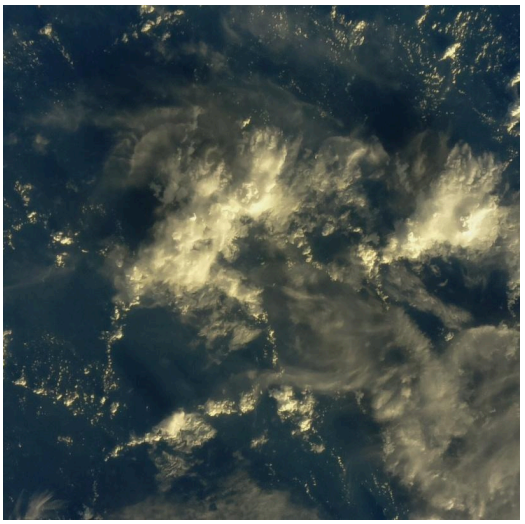


Figure 4 - image

4. Learnings

Our teamwork did not work as well as we expected, but we are currently developing a strategy for the next year, which contains a better division of work - mainly we are planning to divide the programming part into separated sections, on which every member can work parallelly with the others. As for this year, we have found out that machine learning is a very comprehensive task and its learning is very time-consuming, if you set it to only one person of the team. We learn a lot of stuff around neural networks, but we agreed, it was not an optimal way of learning. A coherent tutorial or more systematic learning might be a better option. Last but not least we learn that time is expensive and that you should spend it on something meaningful.

5. Conclusion

Throughout the year we experienced betrayal, but the pain you felt yesterday, is the power you feel today, so by overcoming this, we learnt a lot of cool things and skills. We acquired the basics of machine learning and developed our time-management skills. Finally we learn that the first thing to do, when you are doing an experiment like this, is what are you going to write in the final report, because this is nearly the only thing that you are looking for. We designed our work for an application-like result, so we found it difficult to fill in the report.

We will repeat our-selves: The model is not accurate at all. If it was and if we did not spend a lot of hours on just the model developing we could investigate storms more, we could compare the result with the meteorological radars, but time is running away.