Learning Report

Oscar Galindo

Exercise U-Net and Segmentation Dataset

April 26th, 2021

Dr. Olac Fuentes

Learning

During this activity I learned many things. I will start by indicating that I learned about the specific details of the U-Net implementation. I had seen this network being mentioned before as benchmarks on the task of segmentation but I had never fully gotten a picture about the implementation details. Those details include the concatenation of the result of applying convolution to a corresponding layer performing the transposed convolution on the other side of the "U" shape. From this last I learned that transposed convolutions were a "thing". Transposed convolutions rely on the principle of learning parameters on a sort-of regression procedure in order to predict the content of the upsampled image based on an individual element of the max-pooled image. The transposed convolution is effectively trying to learn parameters to reverse the kernels obtained in the first convolution layers of U-Net.

In relation to training, I refreshed the fact that loss is the differentiable "result" that we use to train the network, I think this was not as clear because I always get confused with that. I also refreshed on what loss represents which is the error and about how overfitting is sometimes represented by a lack of overfitting in the training set while at the same time the validation data has some loss. Furthermore, I practiced how to include optimizers with custom learning rates, which although hard to believe is not that clear how to do from the documentation online. In addition, I observed how changing the learning rate of the network could quickly determine that we overfit from the beginning by observing what happened after Dr. Fuentes asked that we trained the network with a tenth of the original learning rate of 1e-3.

In terms of results the best loss I achieved was around 0.18 for training and 0.45 for validation. This numbers are similar to those the rest of the class reported and I also observed the correct segmentation of a dog image after I trained the network.

Here is the result: