Lab 12 - FCN

Jhony A. Meja
Universidad de Los Andes
Biomedical Engineering Department
ja.mejia12@uniandes.edu.co

1. Materials and methods

Only the 32s_Train script was run because of memory problems. This script took more than 4 hours to run only 1 epoch (aprox. 12000 iterations) (Fig. 1). The estimated number of epochs for fully running the database was 12 epochs, which means that at this pace approximately in 2 days the first script will be run. That process was killed because it was taking too long, which lead to not saving the model because the process was interrupted. That is why changes in the script were tried modifying the batch size and reshaping the images and labels but errors were shown.

From the 32s Train script only 3 validation images were obtained (one image each 4000 iterations). It is important to say that the script was run using Jupyter Notebook, which can be a cause of the excess of time that it took. Another possible explanation for the low pace of computation can be the presence of other processes at the same time. A possible solution for fastening up the process could be running directly from python and not from jupyter.

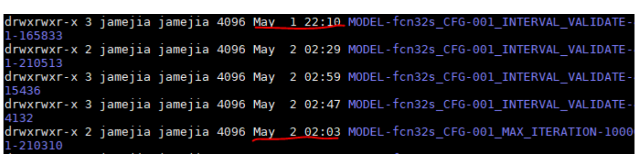

Figure 1. Time spent running 1 epoch (12000 iterations approx).

2. Results

The initialized weights shown good qualitative performance on the validation set (as expected). As the iterations went on, artifacts of miss-segmentation appeared, but the general shape of the main segmentation's was conserved.

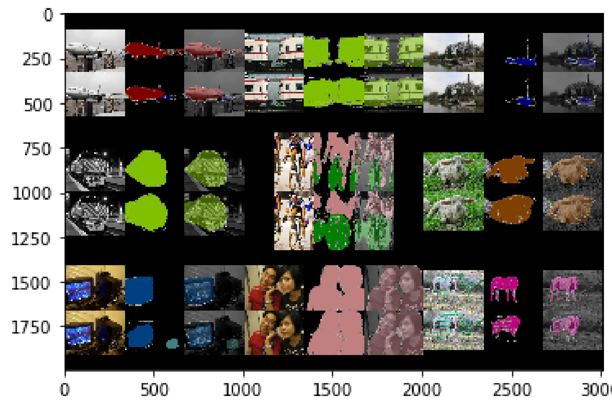

Figure 2. Validation qualitative results after 4000 iterations. Ground truth (top right of each mosaic image), prediction (bottom right of each mosaic image).

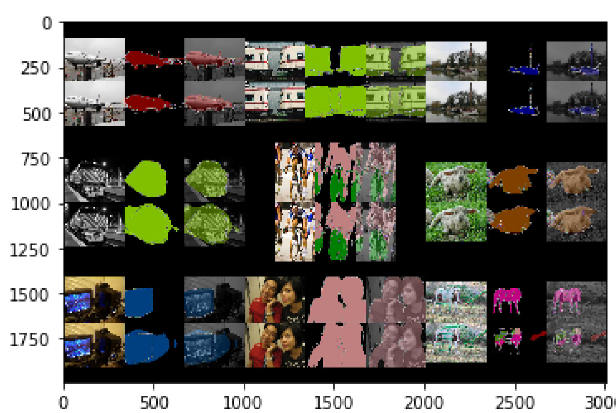

Figure 3. Validation qualitative results after 8000 iterations.

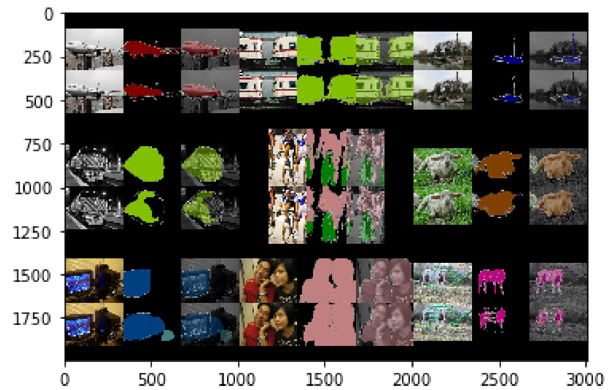

Figure 4. Validation qualitative results after 12000 iterations.

As it can be seen in Figs 2, 3 and 4, predictions show more false positives as the iterations went on. The most noticeable mistakes are seen in over segmentation of train and cow. As expected, the results obtained in the cyclist have a good recall but a bad precision. The rest of the categories seem to show results extremely similar to the ground truth expected. The good performance on this category can be due to low scale variation (big image in the center of the image with little noise).

3. Expected results

In theory 16s is expected to produce finer segmentations. However, the cost to pay is a larger computational cost and possible apparitions of false positives segmentations. On the other hand, 32s produce more general segmentations, which leads to a great recall but a low precision compared with 16s. 16s is extremely useful for detecting fine particles. That is why 16s should be preferred when dealing with multi-scale, multi-category images. On the other hand, datasets with small scale variation might be addressed in an acceptable way using 32s.

In theory the use of pre-trained weights from vgg16 should produce better segmentations based on a more robust model. Building models from scratch might require a lot more iterations to achieve similar results. Finally, the use of pre-trained weights from 32s in 16s should produce even better results. However, the final weights will not be as robust as the ones obtained by using weights from other databases.

4. Conclusions

Training a FCN can be extremely expensive in terms of memory and time. Alternative strategies for fastening up that process must be considered.

The use of pre-trained weights is expected to improve the results obtained, compared with scratch weights. Additionally, the use of weights trained in the same database should improve even more the results obtained, but the replication of it in other databases can be lower. The use of weights from other databases is expected to produce more robust models.