Deep Learning

Project 3

Madi Bokishev

2018320252

Encoder-Decoder Implementation

The objective of this project was to implement and train two models: Unet and Resnet-encoder-Unet. For this task the PascalVOC dataset has been used. This report is conducted to describes the main point of implementation process, results of the training and observations on the models.

The first task of the project was solved by completing the given main_skeleton.py document. All the training and validations were accomplished using GPU as a processing device. Due to the lack of the graphic memory, I had to decrease the value of the batch size to 1 and 4 for Unet and Resnet Encoder respectively. With the given checkpoints for each of the models, the training was done for 1 epoch and stored into the **UNet_final.pth** and **resnet_encoder_unet_final.pth** files. The outputs after the testing were saved in the corresponding **history** folder.

Unet model was the first implemented part of the project. For completing its code, a modified from original model of Unet architecture was used as a reference for fitting the convolutions with their number of expected channels. As well as that, the concatenation of them was accomplished with the help of the pytorch function **pytorch.cat()**.

After finishing the coding part of the Unet model, the training and validation process were successfully implemented in the modules_skeleton.py document. For this process, **Adam optimizer** and **cross Entropy Loss** function were used. The resulted **accuracy** is attached below:

```
epoch 1 train loss :  2.3113843824182236 train acc :  0.6723044259207589
epoch 1 val loss :  2.6460281178432026 val acc :  0.6850606387423486
```

For the second model, Resnet_Encoder_Unet its corresponding code file has been completed. The Residual Block has a similar structure as a Resnet model. Also, the concatenations of the convolutions were implemented as well. This model passed the similar training, and the results of accuracies can be seen here:

```
Training
epoch 1 train loss :  1.1626246326553473 train acc :  0.7130061455492703
epoch 1 val loss :  1.2088168664384134 val acc :  0.6995433532085615
```

As it was mentioned before, the results of the dataset images segmentation are kept in the history folder. There you can see some examples from it: