Homework Assignment 2

Deep Learning, 23/24

Nikolay Vasilev
(63190338)

1 Introduction

In this assignment, we will focus on convolutional neural networks. In the first part, we will implement a simple 18-layer ResNet and will try to achieve at least 90% classification accuracy by adjusting the parameters. In the next part, we will expand this network in order to turn it into a fully convolutional network (FCN-32s) for semantic segmentation. We will compare this network with an implementation of the U-Net network by using the meanIOU metric on the test set and looking at some images of produced segmentation masks. In the last part of this assignment, we will use the U-Net implementation for the image colorization task. We will check what is the difference in the performance of the network when adding or omitting skip-connections.

2 Backbone architecture

ResNet is an architecture that uses residual learning to solve the degradation problem. That allows us to train deeper neural networks and improve performance. Residual learning is based on shortcut connections that perform identity mapping and their outputs are added to outputs of stacked layers. One Residual basic block contains convolutions, batch normalizations after each convolution, ReLU activations, and a single skip. We will follow the architecture described in the original ResNet paper in order to implement the 18-layer ResNet. To achieve the highest score we did the following things:

- Initialize the weights of all convolutions using the Kaiming initialization, which is proven to work well with neural networks that use the ReLU activation function.
- Used a scheduler that reduces the learning rate with 0.1 when the loss plateaus.
- Made sure we don't use softmax in the model since we are using the Cross Entropy as a loss function, and it already applies softmax. Anyway, we apply the softmax to calculate the accuracy and classify the images.

The best parameters proved to be an initial learning rate of 0.1, a regularization parameter of 1e-5, an SGD optimizer, and a batch size of 128. Training the network with these parameters for 20 epochs gave us an accuracy of 95% and a loss of 0.1776 on the test set. In Figure 1 we can see the accuracies and losses we calculated in the training process for all 20 epochs and on both the train and validation sets.

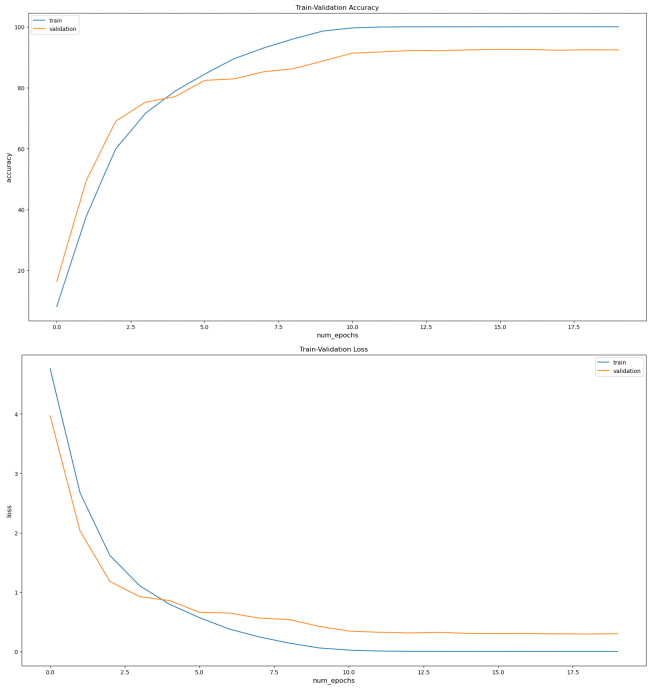

Figure 1: Train-Validation Loss and Accuracy over all 20 epochs

As we can see, we need only 10 epochs to achieve accuracy higher than 90% and the network seems to converge after that. We can also see that our model overfits a bit on our data.

3 Semantic segmentation

We will use the same residual blocks that we defined in the previous task, but we will extend it in order to create a Fully convolutional network (FCN-32s) for a semantic segmentation task. We will get rid of the average pooling and the fully connected layer and instead, we will first apply 1x1 convolution on the output of the last residual block to transform the number of channels into the number of classes. The result we will put through a batch normalization. To the output of that sequence, we will apply transposed convolution in order to transform the height and width of the feature maps to those of the input image and therefore assign each pixel to a specific class. We will also compare the performance of that

network with the performance of our implementation of the U-Net network, following the structure described in the original article. For this part of the task, we will be again using the Cross Entropy as a loss function and therefore we won't apply softmax to the output of the models. However, we will apply it when calculating the mean IOU metrics and when classifying the pixels for visualization.

After training both networks on the semantic segmentation task, we evaluated them on the test set using the mean IOU metric - The U-Net achieved 99.47%, while the ResNet FCN-32s achieved 96.99%. As we can see the difference is only 3%, but it is a fact that the FCN-32s network has a worse performance. This is expected since the U-Net allows skip connections that combine features from different scales. This multi-scale feature fusion property is one of the key strengths that enables better segmentation performance. FCN-32s also allows skip connections but primarily uses features from the last layer, which results in limited information being captured. One way to improve the performance of our ResNet FCN model would be to additionally implement different scales such as FCN-16s and FCN-8s and fuse them to get better predictions.

Let's visualize the images to see the difference in the predicted masks.

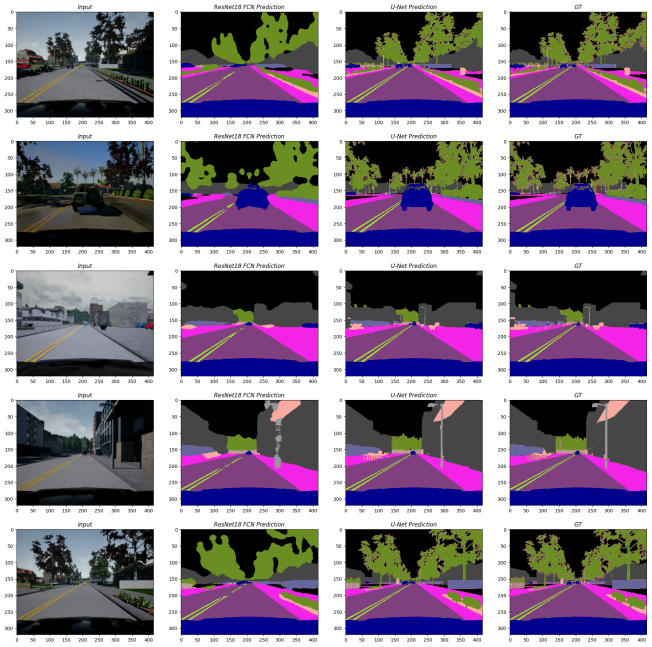

Figure 2: Comparison of semantic segmentation done for 5 images with both FCN-32s and U-Net

In Figure 2 we can see that both networks don't have problems predicting the classes, but the predictions done with ResNet18 FCN-32s are not as clear as the ones done with U-Net. Meanwhile the predictions done with U-Net are almost identical to the ground truth masks.

4 Colorization

In the last part of our assignment, we will check how the U-Net network works for a colorization task. We will also compare the U-Net that uses skip-connections with one that doesn't (simple encoder-decoder architecture). In both of these network implementations, we don't need softmax at all since we are trying to predict the values of the pixels and not classify them into classes. This is also why we are using the L1 loss when training the model.

First, in order to evaluate the performance of both networks, we used the metrics PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index) on the test set. These metrics are focused on pixel-wise accuracy and consider structural similarity, which makes them a good fit for the colorization task's performance evaluation. We got the following results:

Network	PSNR	SSIM
U-Net without skips	17.82	63.62%
U-Net with skips	20.01	92.18%

We can see that there is quite a difference between the performances of both networks. This is expected because the skip-connections allow feature reuse and therefore play a crucial role in capturing complex spatial relationships and information, which results in better performance. When no skip-connections are used in deep neural networks like U-Net, it can lead to a vanishing gradient problem, meaning that during the training gradients of the loss function become very small and therefore it is difficult to backpropagate and update the parameters effectively. This is why the U-Net that uses skip connections achieves better results.

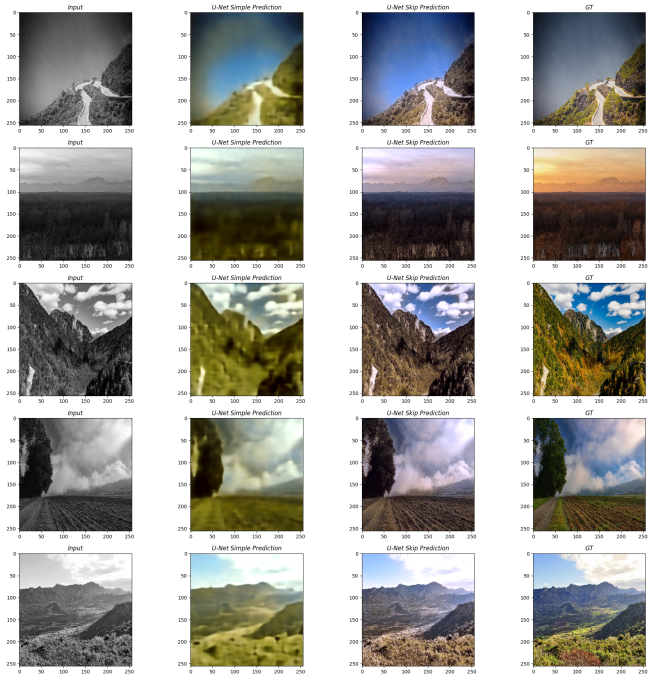

Figure 3: Comparison of colorization done for 5 images with both U-Net with and without skips

We can see in Figure 3 that the predictions that are done using the U-Net without skip-connections are still good and it is visible that the images are the same and the color is similar to the ground truth, but they are very blurry. Meanwhile for the U-Net that uses skip-connections, the predicted colored images are a lot clearer and very close to the ground truths color-wise.