Submission – Assignment 9

Marta Gulida – matriculation number 5585808 – mg776
Tillman Heisner – matriculation number 4517815 – th273
Erik Bode – matriculation number 4505199 – kb301

Coding tasks (comments)

1) (ConvNet2) Describe what happens when you change the number of filters. Do more or fewer perform better?

If we increase the number of filters we are generating more feature maps, thus learn more meaningful representations. This leads to a better performance. E.g., for ConvNet1 we have 33% of accuracy and for ConvNet2 we have 38% of accuracy.

2) (ConvNet3) Describe what happens. What are the benefits of using smaller filters?

We have the same accuracy as for ConvNet2, which is equal to 38%. With a smaller number of filters we have less parameters to train and still learn all important features that are necessary for our network.

3) (Batch normalization) Insert batch normalization after the convolution in ConvNet3 and describe what happens. Is the network performing better? If so, why?

The accuracy increases, and now we have 48% of accuracy while using batch normalization. It improves the gradient flow and helps to speed up the process of learning the parameters.

4) (Transfer learning) Briefly discuss your observations.

In the ModifiedResNet1 we are replacing the linear layer with a new one and then retrain a complete network. It has the highest accuracy (58%) as we train every layer to fit our data. As it trains every parameter again it should take more time in comparison to the others.
In the ModifiedResNet2 we freeze all the layers beside the new linear layer in the end. It has 37% accuracy, which is lower than the first one, but it most probably trains much faster.
In the ModifiedResNet3 we allow the second convolutional layer of the second BasicBlock in layer4 of ResNet to be trained. It has 43% accuracy, which leads to an increase in accuracy and a small increase in the training time in comparison to the ModifiedResNet2.
In the ModifiedResNet4 we also reset the second convolutional layer of the second BasicBlock in layer4 of ResNet allowing it to learn new representations leading to the slightly better accuracy (44%).
In the ModifiedResNet5 we allow the entire layer4 of ResNet to be trained, while freezing all other layers and not reinitializing the second convolutional layer. This again increased the accuracy to 49%.

5) You may have noticed while implementing the CNN models that the number of input features to the linear layer at the end depends on the output of the convolutional layer before it. E.g., a convolutional layer which produces an output of shape (5, 3, 10, 10) (in BCHW

format) will result in the linear layer after it having 3*10*10 input features. Further, the height and width of this output (10 and 10, in this case) also depend on the size of the input image. **How is the ResNet model able to take images of variable sizes as inputs?**

Between the last convolution layer and feed-forward layer there is a global average pooling layer. A global average pooling layer ensures that the last layer always gets the same amount of inputs.