Exercise for MA-INF 2218 Video Analytics SS18
Submission on 30.05.2018
Two Stream Neural Networks

In this assignment we will train a two stream convolutional neural network for action recognition. The two stream CNN contains two neural networks: the spatial stream and the temporal stream. Both networks are trained independently, the output of networks is fused later. The spatial stream uses individual frames of the video whereas the temporal stream uses motion information across the frames in the form of optical flow. You have to implement your solution using python2.7. For your implementation, you are allowed to use OpenCV 2.x, numpy, scikit-learn to its full extend. In this assignment you have to use PyTorch for training the neural networks.

1. Train the spatial stream on the provided dataset. Use the VGG16 network which is already available in PyTorch. You should use Imagenet weights as initialization (Also available in PyTorch). After training, report your accuracy on the test set and provide a plot of the learning curve (objective function value over number of iterations) of the training set. You should plot the learning curve using the TensorBoard.
   (2 Points)

2. Train the temporal stream using a stack of optical flows as input. For each frame f_t in the video, calculate optical flows between $(f_t,\ f_{t+1})$, ..., (f_{t+9}, f_{t+10}). **Note:** The optical flows are already computed and available alongside the provided dataset. You should convert the stack of flows to an image with the required number of channels (You can consider each flow frame as a channel). As the given Imagenet model is trained for a 3 channel input image in the first convolutional connection or layer, to make it work with a input image with channel number more than 3, you have to modify the weight matrix of the first convolutional connection. For modification, average each kernel across the input channels, and copy the averaged kernel as much as needed. *(3 Points)*

3. Fuse the learned feature vectors (the output of second last layer) from both spatial and temporal streams. Train a linear SVM on the fused feature vectors and report the accuracy on test set.
   Hint: The output feature vectors from both networks are frame level feature vectors. To get feature vector per video you have to average frame level feature vectors. *(4 Points)*

4. Design an end-to-end architecture using both spatial and temporal streams. You are allowed to use any desired technique for fusing the output feature vectors of the streams. Train your new model and report the accuracy on test set. *(3 Points)*
   Hint: You can use the trained weights from the previous section. Also, you can fix the streams and fine-tune the new layers.

5. Implement and use a non-local-block layer (dot-product version) as described in [1].

   - Modify the spatial stream by adding the non-local-block layers before the 2nd, 3rd, and 4th maxpooling layers.
   - Train the spatial stream with the updated configuration.
   - Output the learning curve.

- Compare the obtained learning curve with the learning curve of task 1.

(8 Points)

Network Schema: For this sheet, you need to use a GPU.
In case of any question please reach me at **fayyaz@iai.uni-bonn.de**

[1] X. Wang, R. Girshick, A. Gupta, K. He: Non-local Neural Networks