
Exercise: TensorBoard (ca. 2h)

1. Learning goal

In the lecture we have introduced the CNN model and in the last exercise your task was to get some CNN implementation running. The CNN is actually a simple model. Nevertheless, it is complex regarding the number of its parameters and being mainly a black box. For this, visualizations are helpful for understanding what happens in a CNN during training. TensorBoard is a tool that comes with TensorFlow and allows producing visualizations of scalar values, histograms and images during training and testing of TensorFlow computation graphs.

2. Getting started with TensorBoard

Take your CNN implementation or my CNN code at

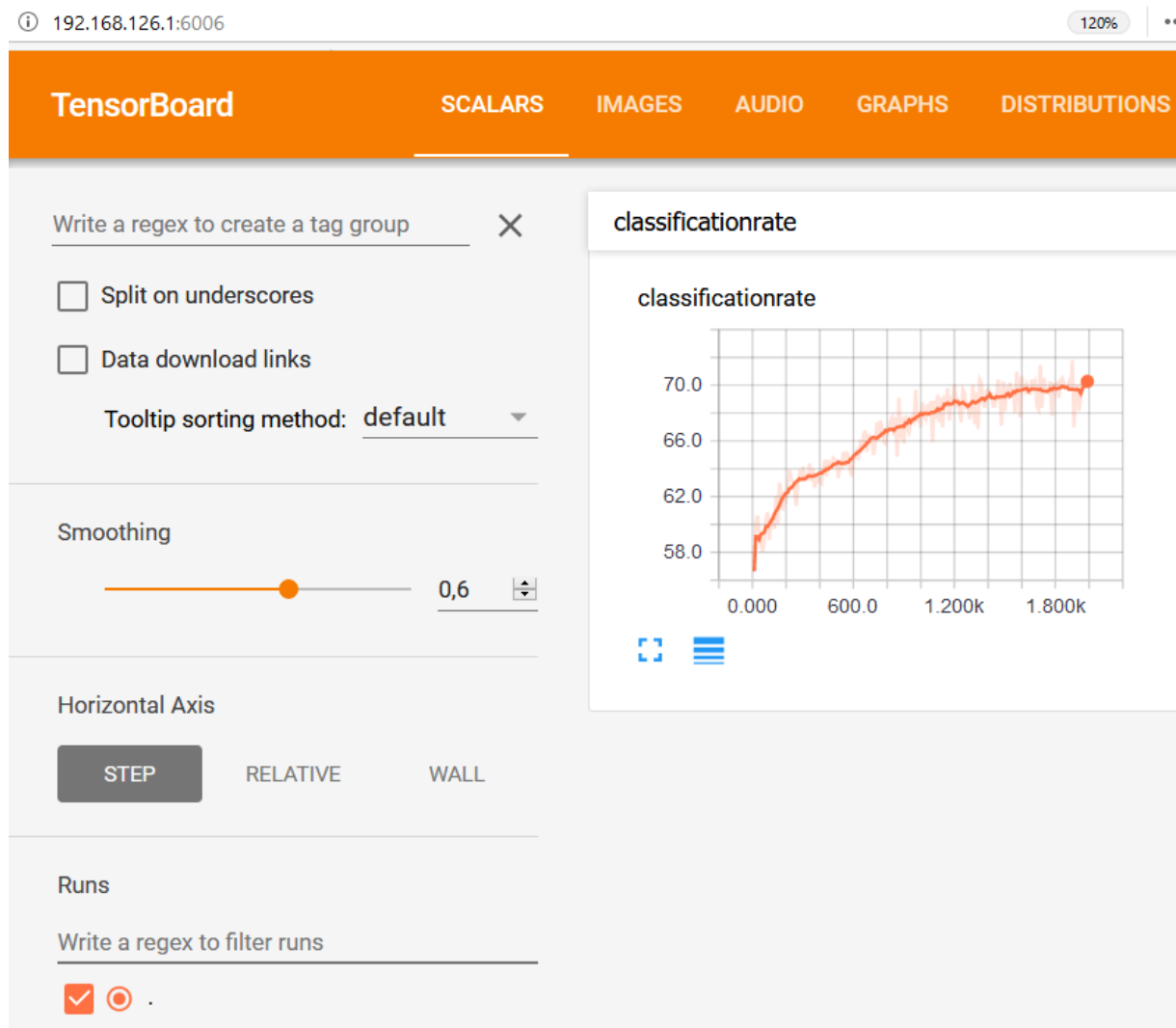
<https://github.com/juebrauer/Solutions-Exercises-MultimodalSensorSystems/tree/master/A%20first%20Convolutional%20Neural%20Network>

and read some TensorBoard introduction such as

https://www.tensorflow.org/programmers_guide/summaries_and_tensorboard

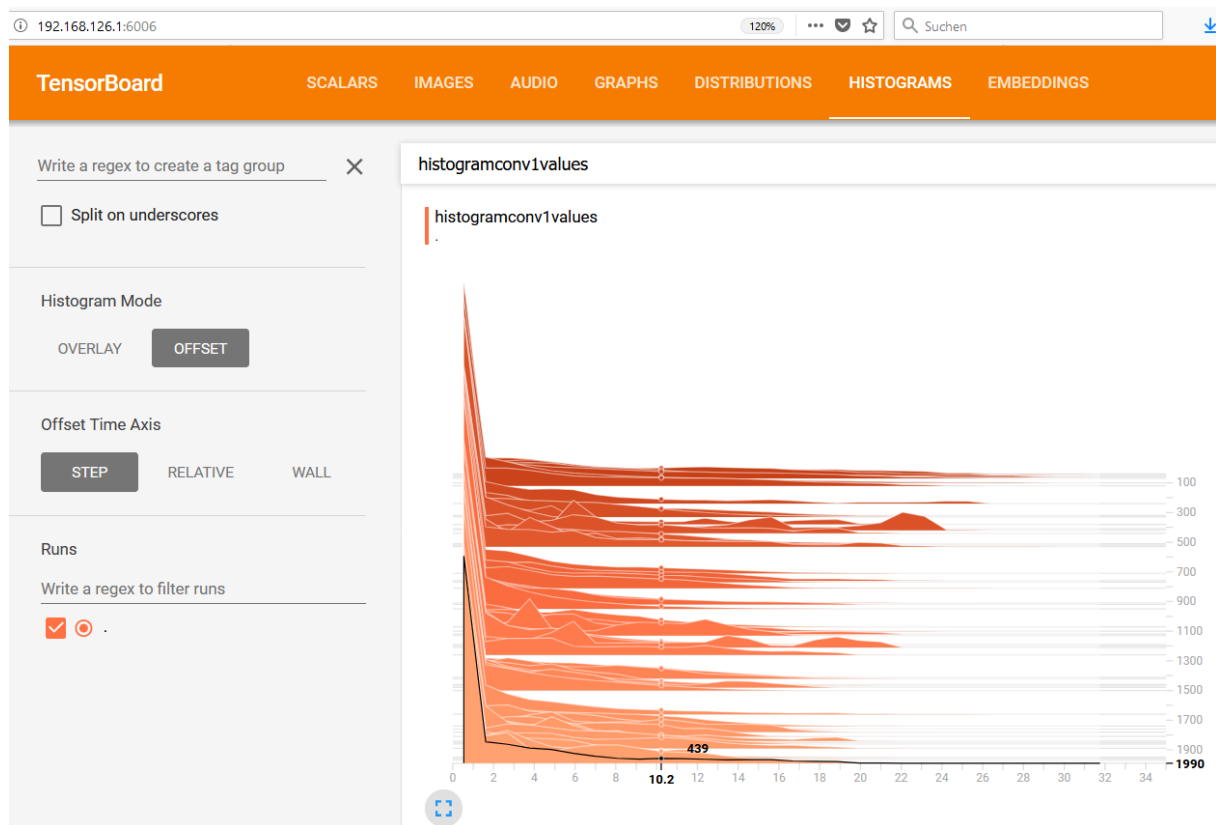
3. Log some scalar value

First adapt my/your CNN TensorFlow code, such that you can evaluate the classification performance on a test dataset every 50 training steps. Then augment your code, such that the classification performance is logged during training and visualize it using TensorBoard:



4. Log histograms over conv1 filter values

Next, choose some convolution layer, e.g., convolutional layer 1. An interesting question is in which range are the activation values of these neurons / filters (e.g., after the RELU step) and how often each value occurs. A compact answer for this question is a histogram of these activation values. For this, log the conv1 filter values (conv1 tensor) as well during each 50 training steps. A resulting view of the histograms over training steps could look like this:



5. Group ops for a better visualization of your computation graph

Add appropriate commands of the form

```
with tf.name_scope("test-name-123") : ...
```

for grouping ops such that the visualized computation graph is easier to read:

