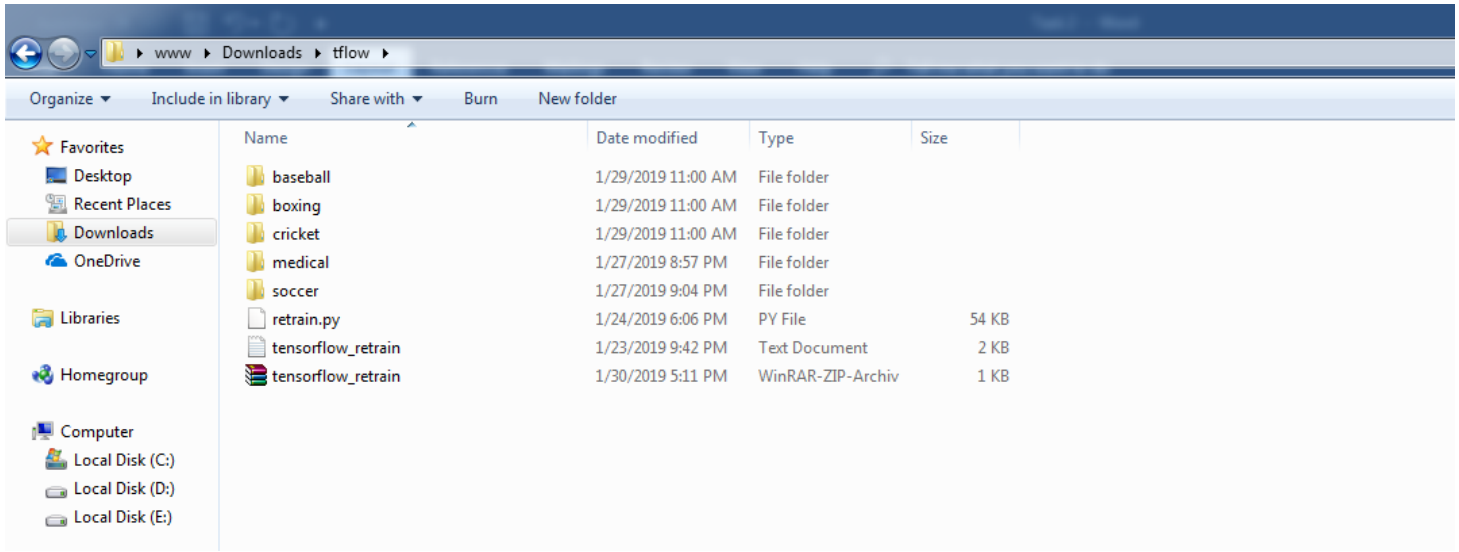
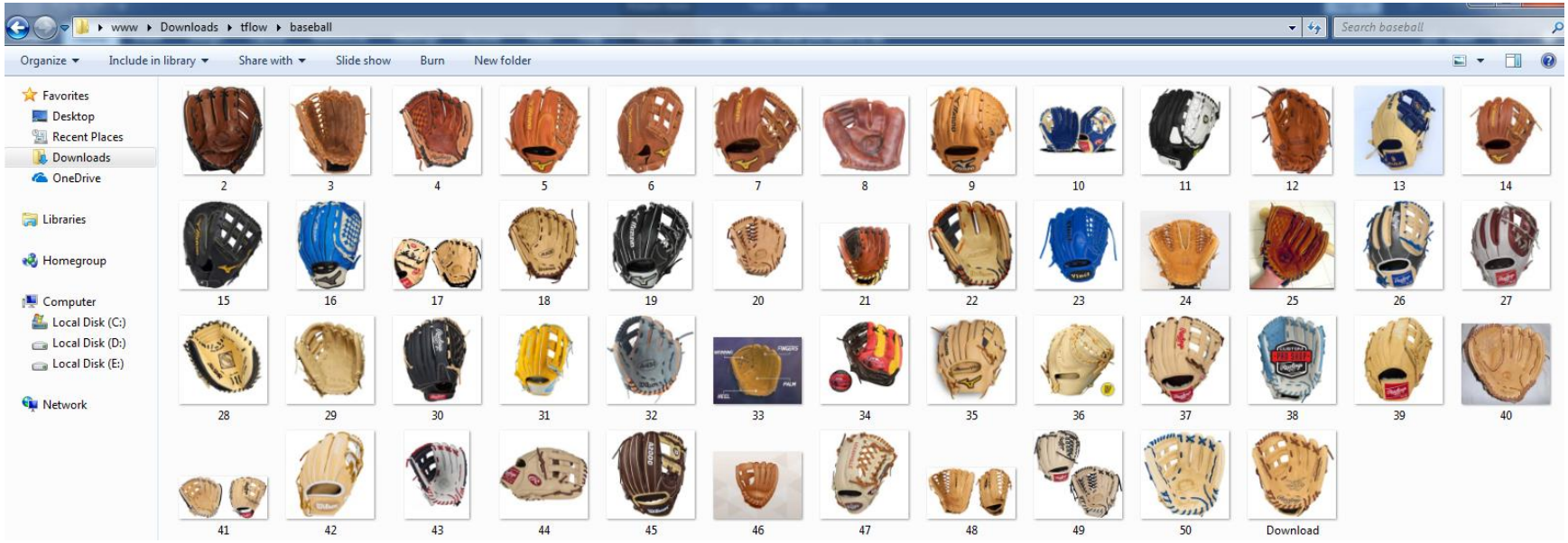
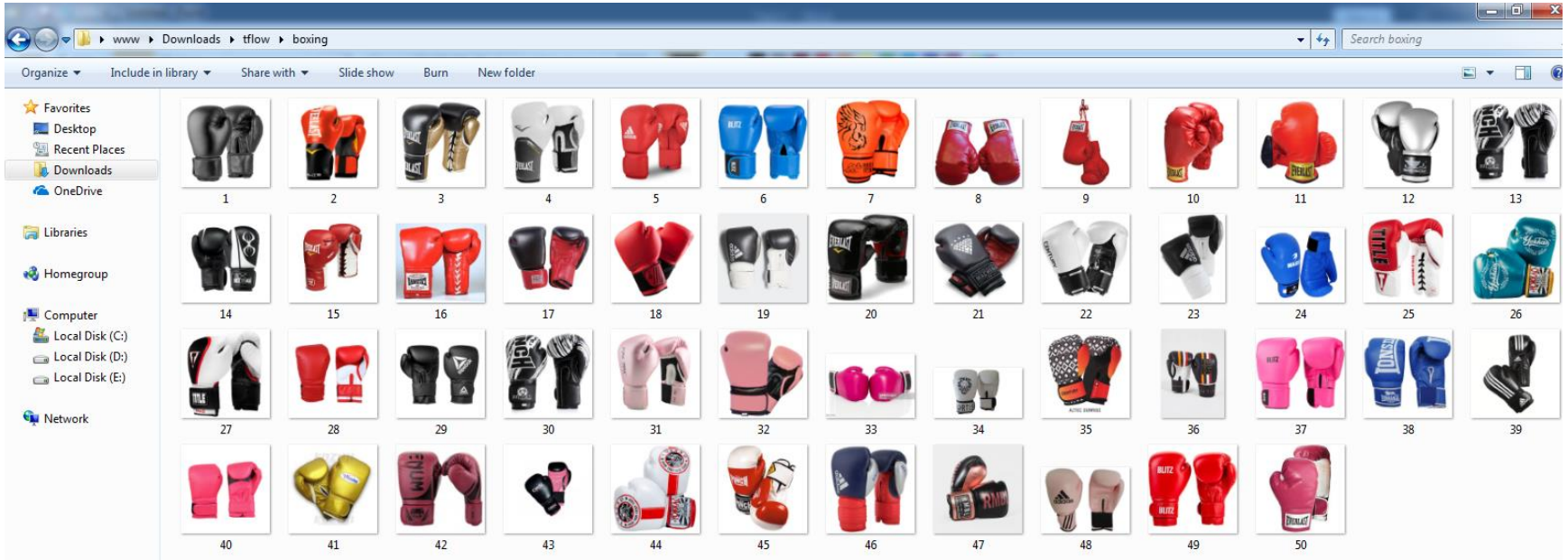
Task 2: Tensorflow [10%] Dataset: Select a topic (candies, building, …) with at least five different categories and at least 50 pictures in each ccategory
Instructions: Please deliver all commands in your documentation (use a word-document and convert it later into a pdf);

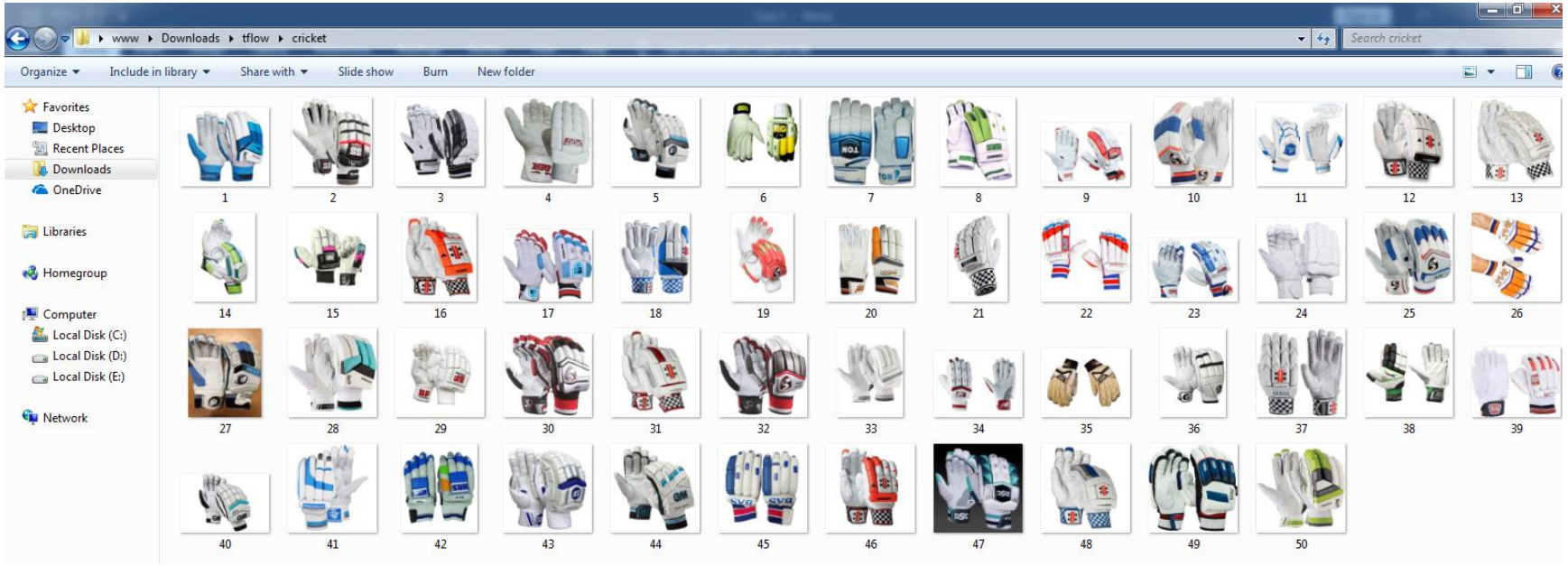
1. Select a topic and collect the pictures

    a. Download the pictures and put them in five folders named after the categories.




www ▸ Downloads ▸ tflow ▸ baseball
Search baseball
Organize
Include in library
Share with
Slide show
Burn
New folder
Favorites
Desktop
Recent Places
Downloads
OneDrive
Libraries
Homegroup
Computer
Local Disk (C:)
Local Disk (D:)
Local Disk (E:)
Network
2 3 4 5 6 7 8 9 10 11 12 13 14
15 16 17 18 19 20 21 22 23 24 25 26 27
28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50 Download

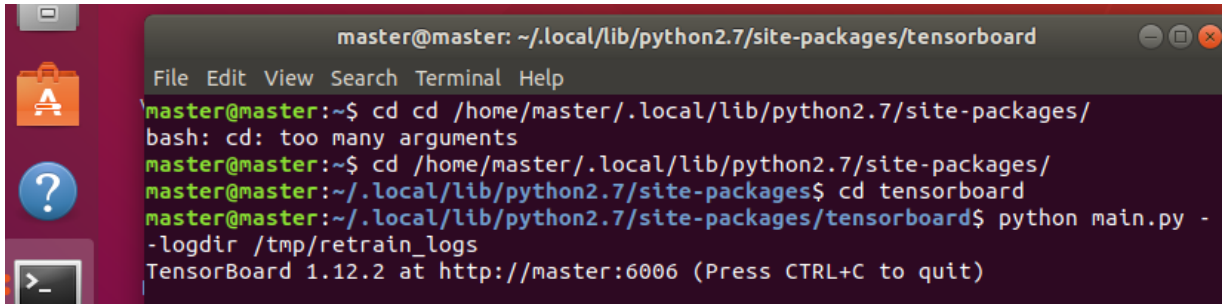

www ▸ Downloads ▸ tflow ▸ boxing
Search boxing
Organize
Include in library
Share with
Slide show
Burn
New folder
Favorites
Desktop
Recent Places
Downloads
OneDrive
Libraries
Homegroup
Computer
Local Disk (C:)
Local Disk (D:)
Local Disk (E:)
Network
1 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 31 32 33 34 35 36 37 38 39
40 41 42 43 44 45 46 47 48 49 50

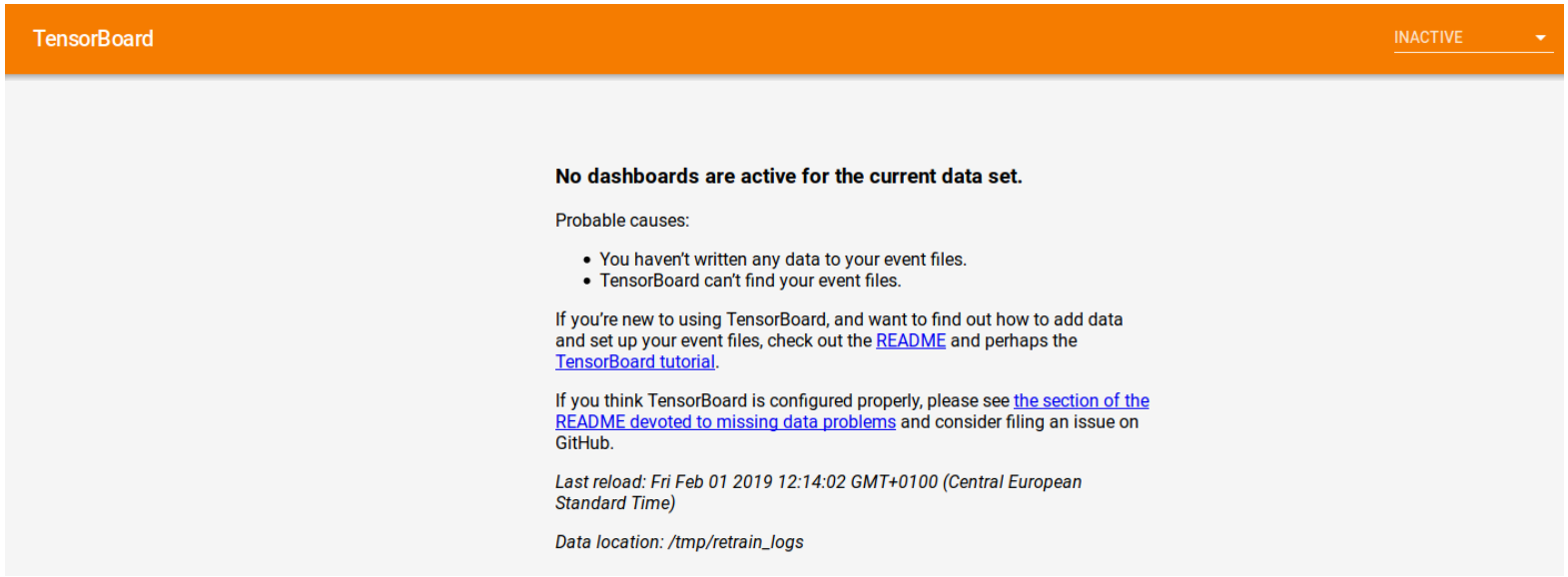

www ▸ Downloads ▸ tflow ▸ cricket
Search cricket
Organize
Include in library
Share with
Slide show
Burn
New folder
Favorites
Desktop
Recent Places
Downloads
OneDrive
Libraries
Homegroup
Computer
Local Disk (C:)
Local Disk (D:)
Local Disk (E:)
Network
1 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 31 32 33 34 35 36 37 38 39
40 41 42 43 44 45 46 47 48 49 50


www ▸ Downloads ▸ tflow ▸ medical
Search medical
Organize
Include in library
Share with
Slide show
Burn
New folder
Favorites
Desktop
Recent Places
Downloads
OneDrive
Libraries
Homegroup
Computer
Local Disk (C:)
Local Disk (D:)
Local Disk (E:)
Network
1 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 29 30 31 32 33 34 35 36 37 38 39
40 41 42 43 44 45 46 47 48 49 50

2. Monitoring: Tensorboard

a. Start Tensorboard to monitor the progress

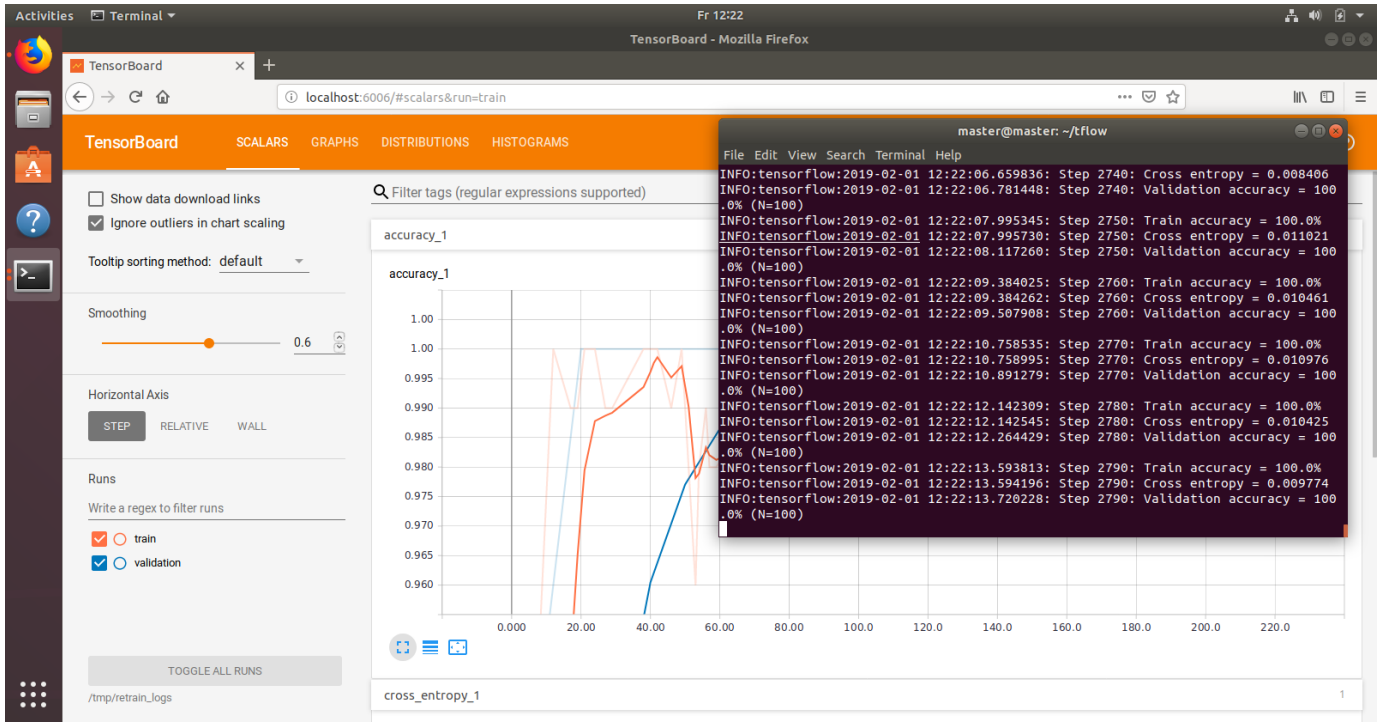


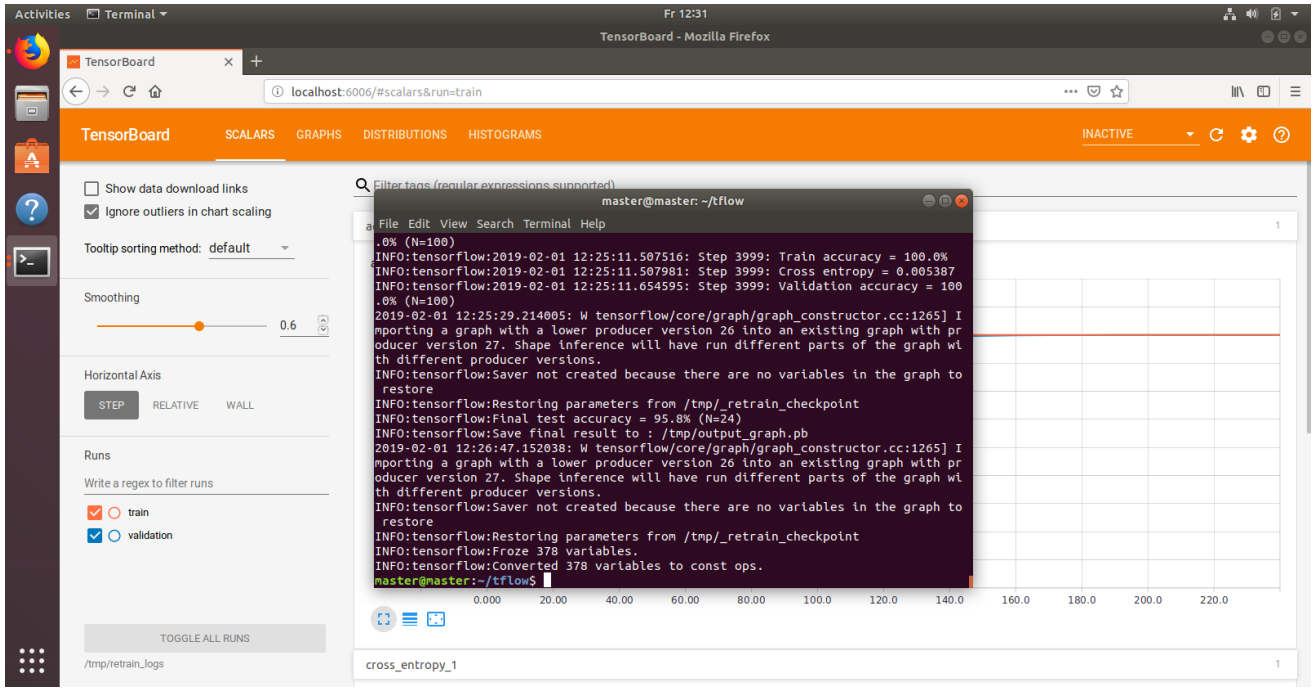


3. Model Training

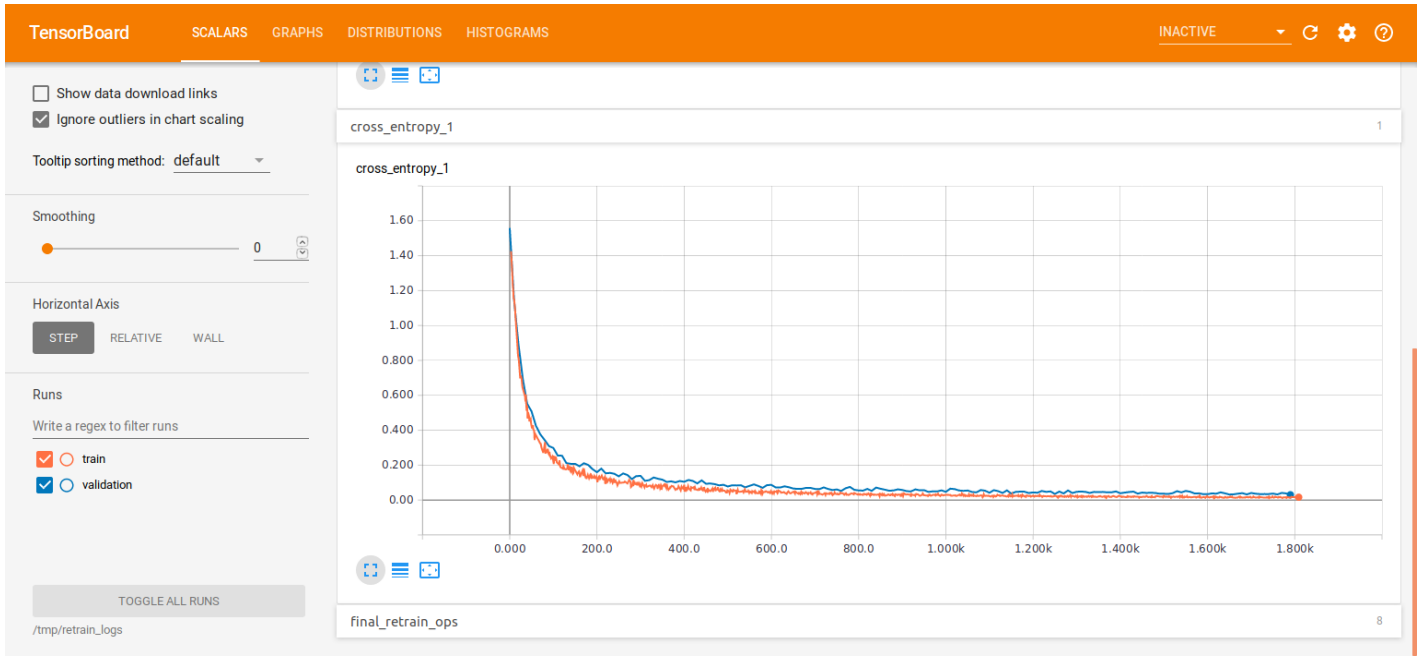
a. Train the model using your pictures

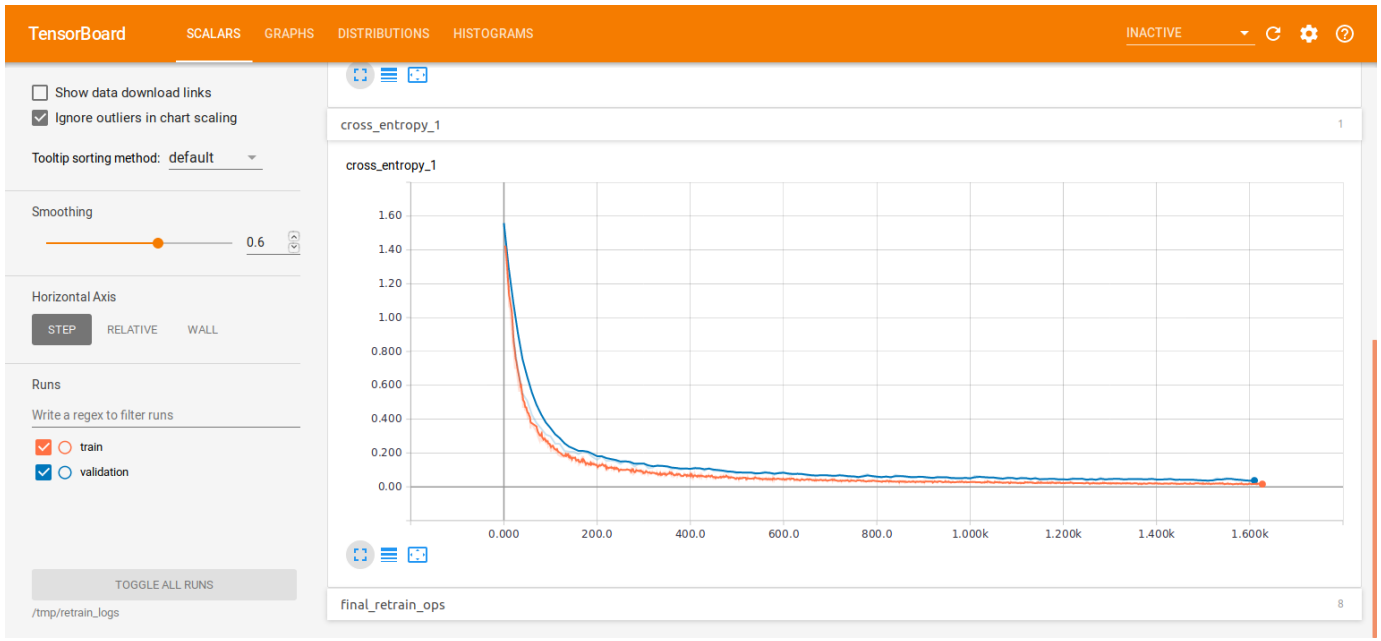
```
master@master:~/tflow$ python retrain.py --image_dir ~/tflow_photos
INFO:tensorflow:Looking for images in 'baseball'
INFO:tensorflow:Looking for images in 'boxing'
INFO:tensorflow:Looking for images in 'cricket'
INFO:tensorflow:Looking for images in 'medical'
INFO:tensorflow:Looking for images in 'soccer'
INFO:tensorflow:Using /tmp/tfhub_modules to cache modules.
INFO:tensorflow:Downloading TF-Hub Module 'https://tfhub.dev/google/imagenet/inc
eption_v3/feature_vector/1'.
```


Activities
Terminal
Fr 12:22
TensorBoard - Mozilla Firefox
TensorBoard
localhost:6006/#scalars&run=train
TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.6
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
validation
TOGGLE ALL RUNS
/tmp/retrain_logs
Filter tags (regular expressions supported)
accuracy_1
cross_entropy_1
master@master: ~/tflow
File Edit View Search Terminal Help
INFO:tensorflow:2019-02-01 12:22:06.659836: Step 2740: Cross entropy = 0.008406
INFO:tensorflow:2019-02-01 12:22:06.781448: Step 2740: Validation accuracy = 100
.0% (N=100)
INFO:tensorflow:2019-02-01 12:22:07.995345: Step 2750: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:22:07.995730: Step 2750: Cross entropy = 0.011021
INFO:tensorflow:2019-02-01 12:22:08.117260: Step 2750: Validation accuracy = 100
.0% (N=100)
INFO:tensorflow:2019-02-01 12:22:09.384025: Step 2760: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:22:09.384262: Step 2760: Cross entropy = 0.010461
INFO:tensorflow:2019-02-01 12:22:09.507908: Step 2760: Validation accuracy = 100
.0% (N=100)
INFO:tensorflow:2019-02-01 12:22:10.758535: Step 2770: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:22:10.758995: Step 2770: Cross entropy = 0.010976
INFO:tensorflow:2019-02-01 12:22:10.891279: Step 2770: Validation accuracy = 100
.0% (N=100)
INFO:tensorflow:2019-02-01 12:22:12.142309: Step 2780: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:22:12.142545: Step 2780: Cross entropy = 0.010425
INFO:tensorflow:2019-02-01 12:22:12.264429: Step 2780: Validation accuracy = 100
.0% (N=100)
INFO:tensorflow:2019-02-01 12:22:13.593813: Step 2790: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:22:13.594196: Step 2790: Cross entropy = 0.009774
INFO:tensorflow:2019-02-01 12:22:13.720228: Step 2790: Validation accuracy = 100
.0% (N=100)

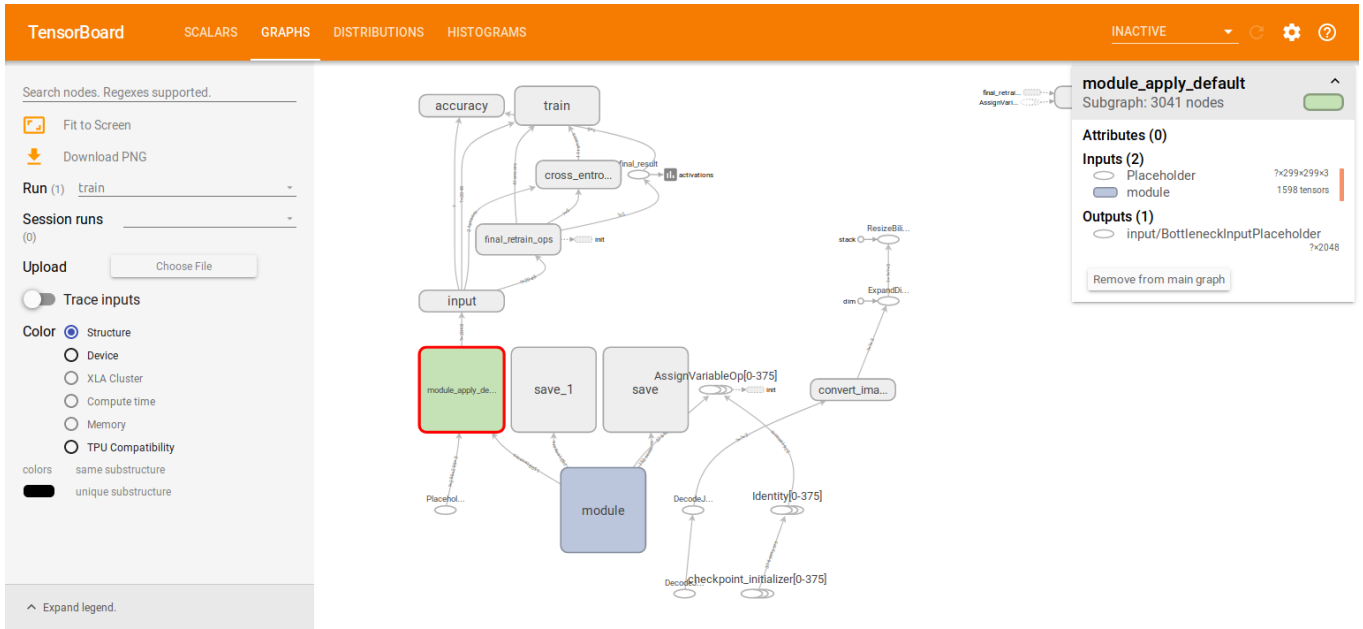

Activities
Terminal
Fr 12:31
TensorBoard - Mozilla Firefox
TensorBoard
localhost:6006/#scalars&run=train
TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
INACTIVE
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.6
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
validation
TOGGLE ALL RUNS
/tmp/retrain_logs
cross_entropy_1
master@master: ~/tflow
File Edit View Search Terminal Help
.0% (N=100)
INFO:tensorflow:2019-02-01 12:25:11.507516: Step 3999: Train accuracy = 100.0%
INFO:tensorflow:2019-02-01 12:25:11.507981: Step 3999: Cross entropy = 0.005387
INFO:tensorflow:2019-02-01 12:25:11.654595: Step 3999: Validation accuracy = 100
.0% (N=100)
2019-02-01 12:25:29.214005: W tensorflow/core/graph/graph_constructor.cc:1265] I
mporting a graph with a lower producer version 26 into an existing graph with pr
oducer version 27. Shape inference will have run different parts of the graph wi
th different producer versions.
INFO:tensorflow:Saver not created because there are no variables in the graph to
 restore
INFO:tensorflow:Restoring parameters from /tmp/_retrain_checkpoint
INFO:tensorflow:Final test accuracy = 95.8% (N=24)
INFO:tensorflow:Save final result to : /tmp/output_graph.pb
2019-02-01 12:26:47.152038: W tensorflow/core/graph/graph_constructor.cc:1265] I
mporting a graph with a lower producer version 26 into an existing graph with pr
oducer version 27. Shape inference will have run different parts of the graph wi
th different producer versions.
INFO:tensorflow:Saver not created because there are no variables in the graph to
 restore
INFO:tensorflow:Restoring parameters from /tmp/_retrain_checkpoint
INFO:tensorflow:Froze 378 variables.
INFO:tensorflow:Converted 378 variables to const ops.
master@master:~/tflow$

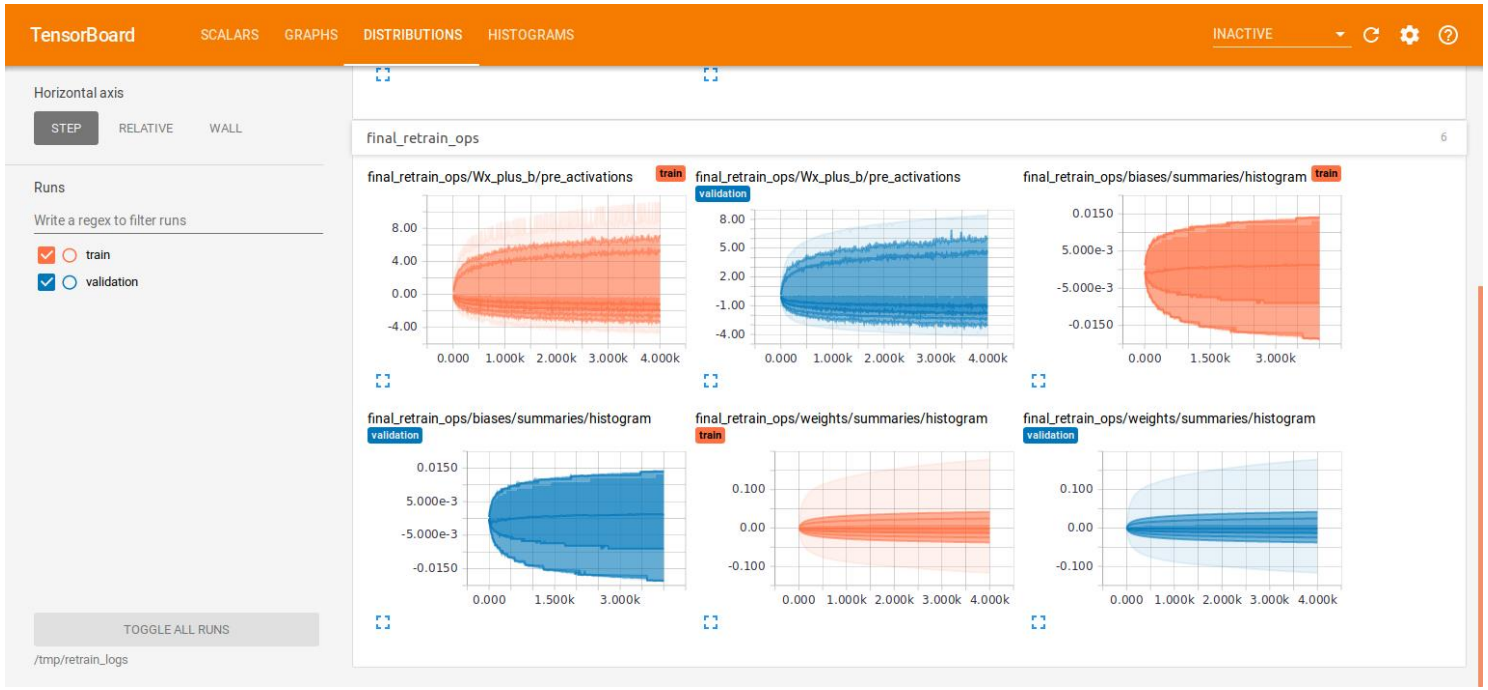
b. Control the training using Tensorboard (tune the smoothing factor)

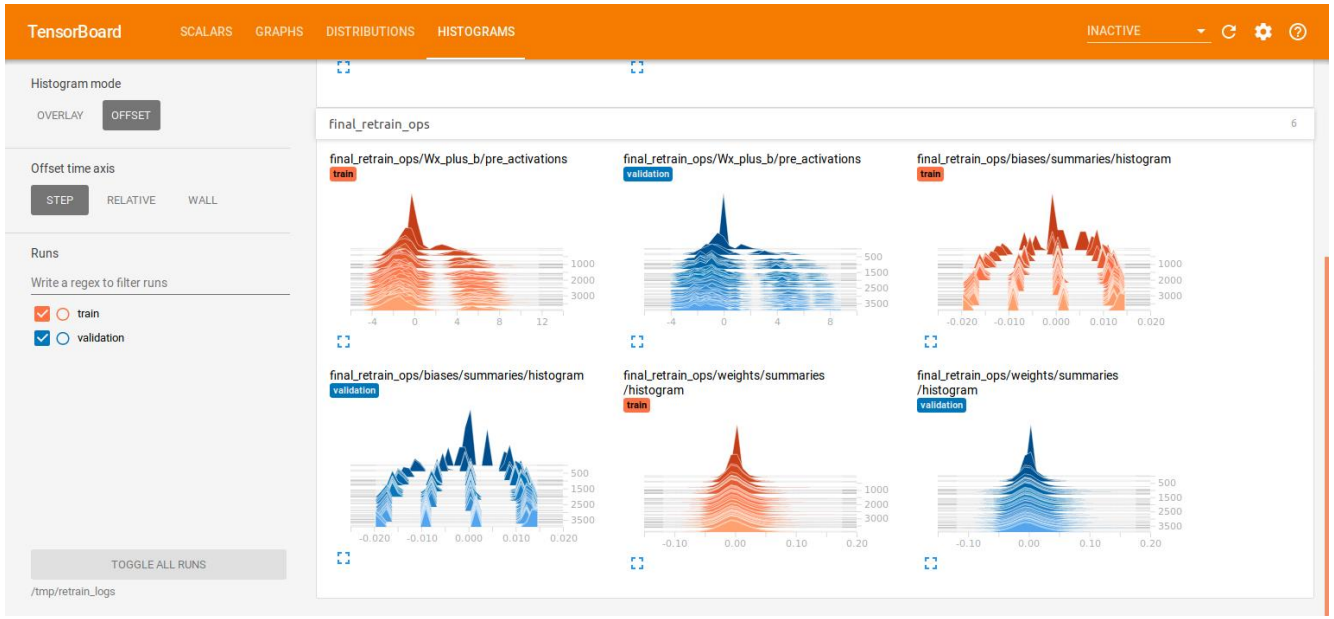




c. Adapt the model training command

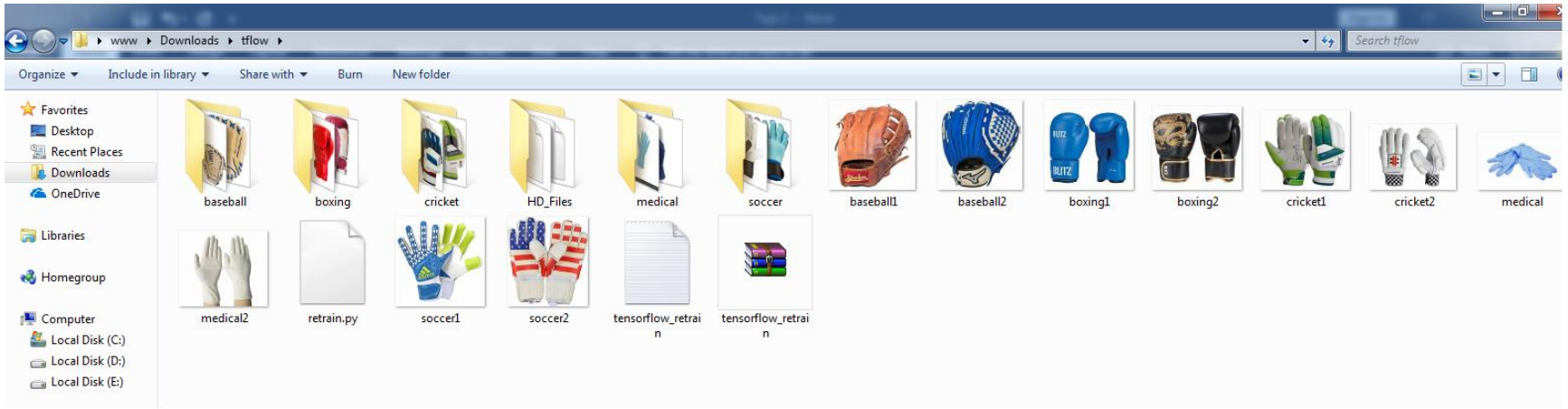



TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
INACTIVE
Horizontal axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
validation
TOGGLE ALL RUNS
/tmp/retrain_logs
final_retrain_ops
6
final_retrain_ops/Wx_plus_b/pre_activations
train
final_retrain_ops/Wx_plus_b/pre_activations
validation
final_retrain_ops/biases/summaries/histogram
train
final_retrain_ops/biases/summaries/histogram
validation
final_retrain_ops/weights/summaries/histogram
train
final_retrain_ops/weights/summaries/histogram
validation

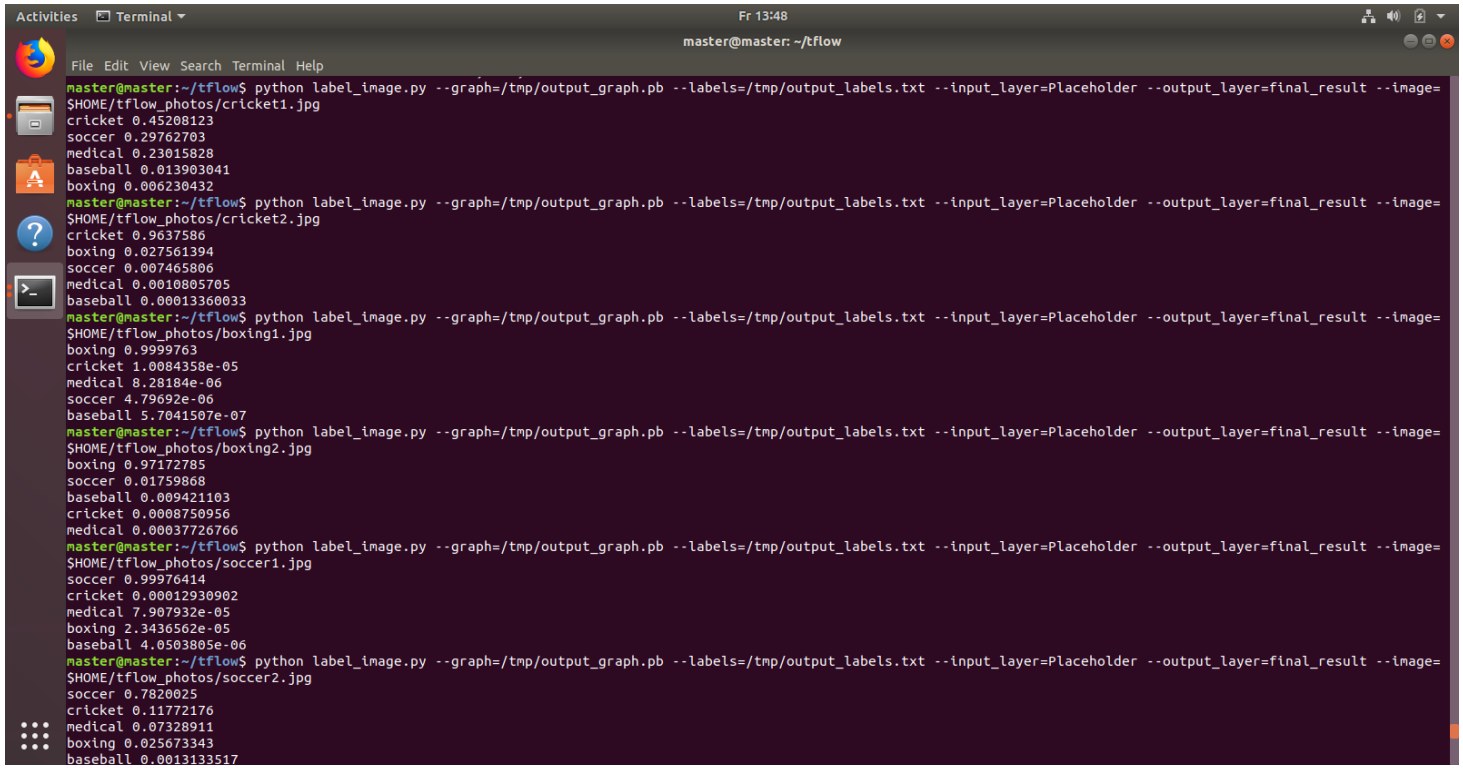

TensorBoard
SCALARS
GRAPHS
DISTRIBUTIONS
HISTOGRAMS
INACTIVE
Histogram mode
OVERLAY
OFFSET
Offset time axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
validation
TOGGLE ALL RUNS
/tmp/retrain_logs
final_retrain_ops
6
final_retrain_ops/Wx_plus_b/pre_activations
train
final_retrain_ops/Wx_plus_b/pre_activations
validation
final_retrain_ops/biases/summaries/histogram
train
final_retrain_ops/biases/summaries/histogram
validation
final_retrain_ops/weights/summaries/histogram
train
final_retrain_ops/weights/summaries/histogram
validation

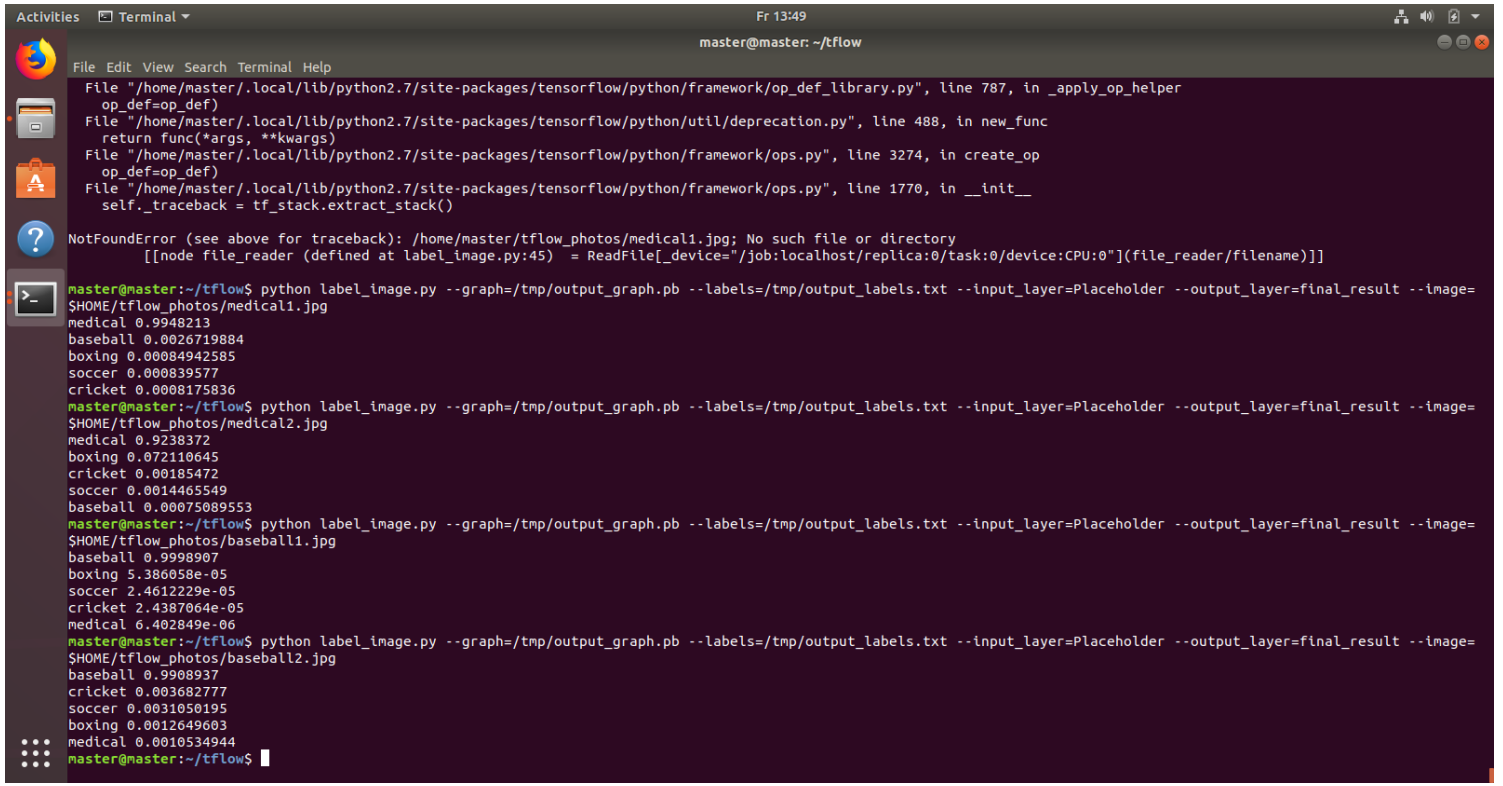
4. Prediction

a. Use at least two pictures of each category



b. Categorize the pictures and put screenshots with the results in your report





- 400 training steps were used.
- 5 sub categories different types of ‘hand gloves’ were used. Soccer, Cricket, Boxing, Medical and Baseball.
- Image files should be in format jpeg, jpg, JPEG or JPG.
- Retrain.py script was used to train above images (total 250). Outputs of this script stored in /tmp/* folders
- Smoothing factor of 0.6 was used to smooth out the noise in the pixel data.
- Random_crop, random_scale, flpip_left_right, random_brightness parms were used to smooth the images inside the script.
- Jpeg, jpg, bmp, gif, png files can be predicted using the model created from retrain step.
- Bottleneck tensors were used to minimize loss function.
- 95.8% final accuracy was achieved in retraining step.
- Test data prediction gave 45%, 96%, 99%, 97%, 99%, 78%, 99%, 92%, 99%, 99%.