Object Detection in an Urban Environment

1. Project Overview

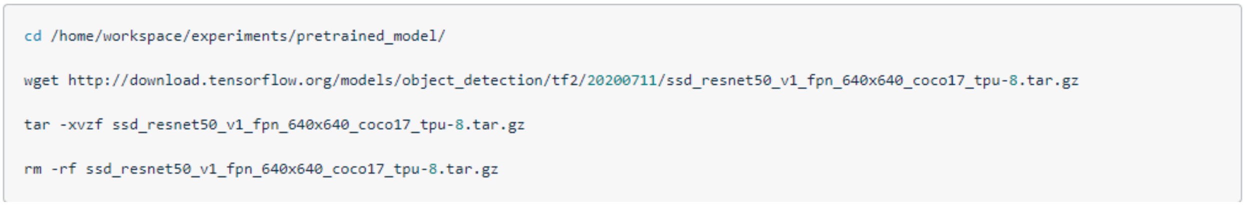
The goal of the project is to detect object in an urban environment. For this project, the waymo open dataset is used for sensor data. The waymo dataset is stored into .tfrecord files. A dataset of images of urban environments containing annotated cyclists, pedestrians and vehicles were provided. In the first task, we need to perform an Exploratory Data Analysis (EDA), like object occlusions, label distribution etc. Later we need to train a Single Shot detector, a pertained model of SSD Resnet 50 640x640 model is given. At first we need to train and evaluate the pre-trained model and then based on the results we need to improve the performance of object detection by modifying the hyper-parameters of the CNN in config files.

The modifications to the config file need to reduce the loss of model and better detection. At last step, after satisfactory results, the model will be exported and inference video is created.

2. Set Up

i) The jupyter notebook is launched using the script ./launch_jupyter.sh

ii) Download the pretrained model and move to the /home/workspace/experiments/pretrained_model/ folder and extract the tar.gz as described in the below steps



```
cd /home/workspace/experiments/pretrained_model/

wget http://download.tensorflow.org/models/object_detection/tf2/20200711/ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz

tar -xvzf ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz

rm -rf ssd_resnet50_v1_fpn_640x640_coco17_tpu-8.tar.gz
```

iii) Edit the config file so a new config file is created with name ***pipeline_new.config*** . Move this file to the ***/home/workspace/experiments/reference/ .***

cd /home/workspace/
python edit_config.py --train_dir /home/workspace/data/train/ --eval_dir/home/workspace/data/val/ --batch_size 2 --checkpoint/home/workspace/experiments/pretrained_model/ssd_resnet50_v1_fpn_640x640_coco17_tpu-8/checkpoint/ckpt-0 --label_map/home/workspace/experiments/label_map.pbtxt

iv) Training Process

python experiments/model_main_tf2.py --model_dir=experiments/reference/ --pipeline_config_path=experiments/reference/pipeline_new.config

Monitor the process by executing the below command.

```
python -m tensorboard.main --logdir experiments/reference/
```

Evaluation Process

```
python experiments/model_main_tf2.py --
model_dir=experiments/reference/ --
pipeline_config_path=experiments/reference/pipeline_new.config --
checkpoint_dir=experiments/reference/
```

v) Export the trained model

```
python experiments/exporter_main_v2.py --input_type image_tensor --
pipeline_config_path experiments/reference/pipeline_new.config --
trained_checkpoint_dir experiments/reference/ --output_directory
experiments/reference/exported/
```

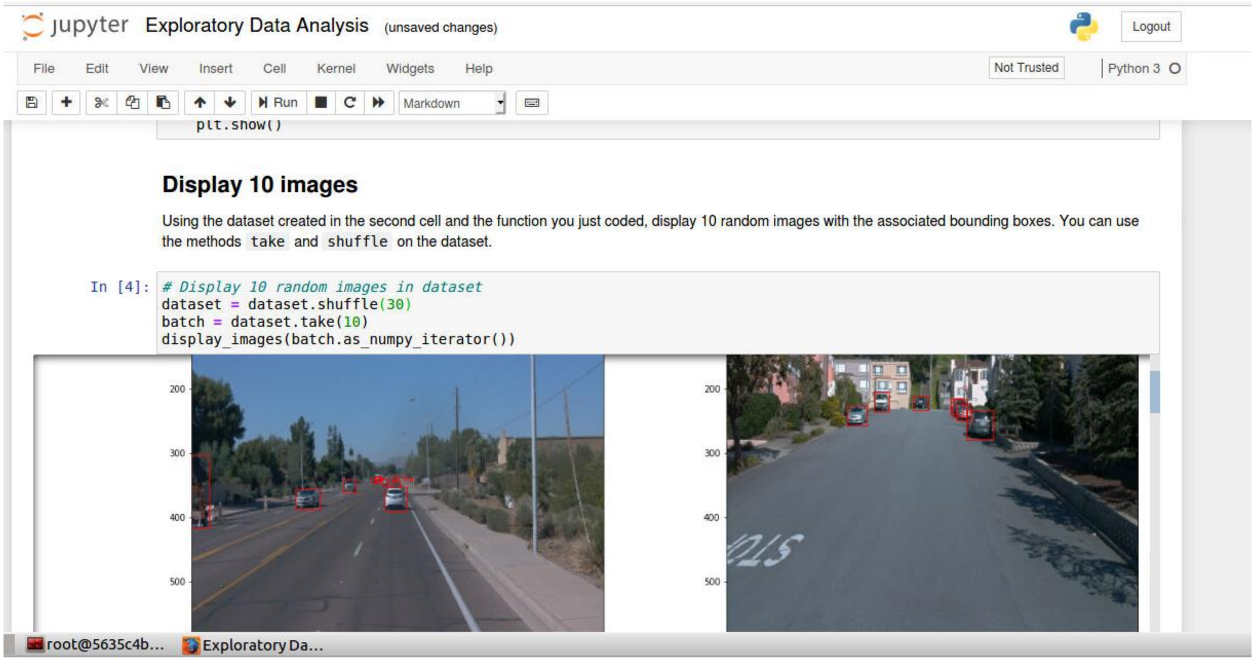
Modify the config file and train and export the trained model when you find the results are satisfactory.

3. Dataset

- Dataset Analysis:

  The dataset is divided into train, test, val and stored into respective folders. We use the train folder to do dataset analysis.

  Below are some images of with its bounding boxes

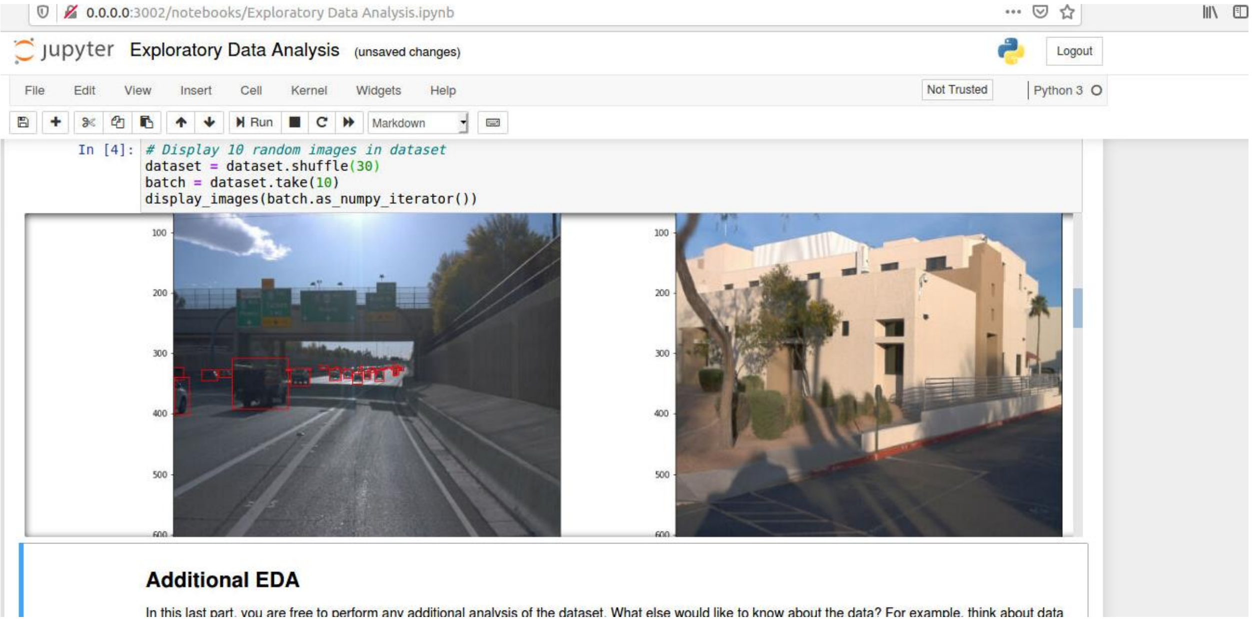


```
    plt.show()
```

Display 10 images

Using the dataset created in the second cell and the function you just coded, display 10 random images with the associated bounding boxes. You can use the methods `take` and `shuffle` on the dataset.

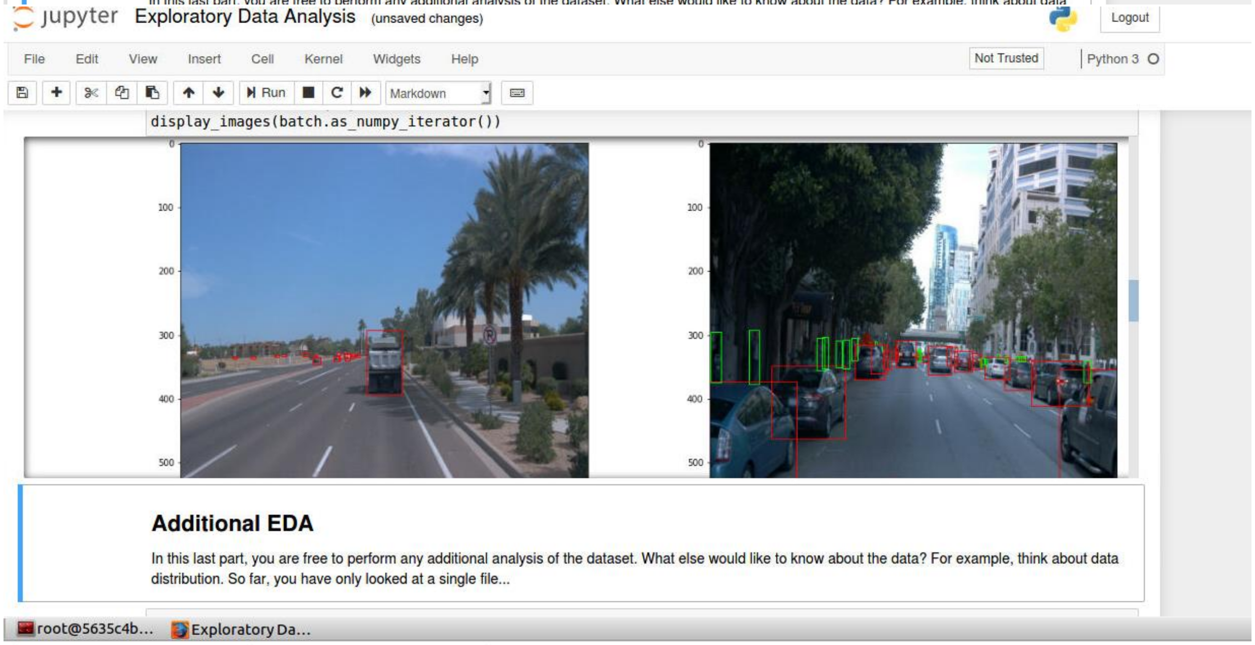
```
In [4]: # Display 10 random images in dataset
        dataset = dataset.shuffle(30)
        batch = dataset.take(10)
        display_images(batch.as_numpy_iterator())
```



```
In [4]: # Display 10 random images in dataset
        dataset = dataset.shuffle(30)
        batch = dataset.take(10)
        display_images(batch.as_numpy_iterator())
```

Additional EDA

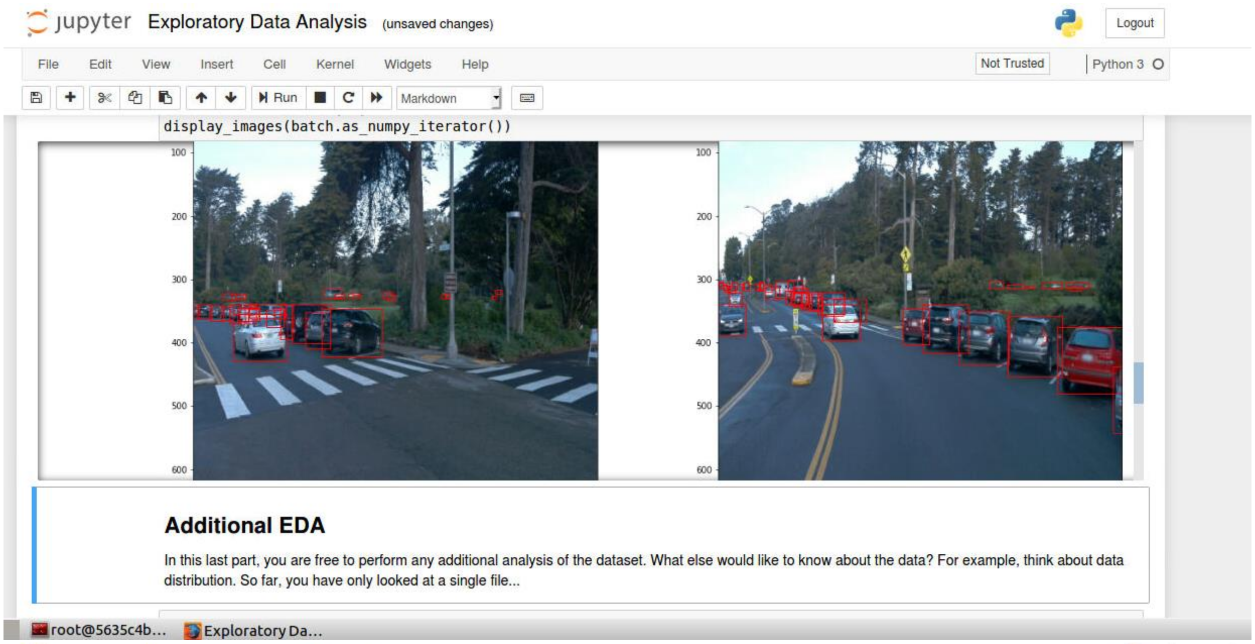
In this last part, you are free to perform any additional analysis of the dataset. What else would like to know about the data? For example, think about data



```
display_images(batch.as_numpy_iterator())
```

Additional EDA

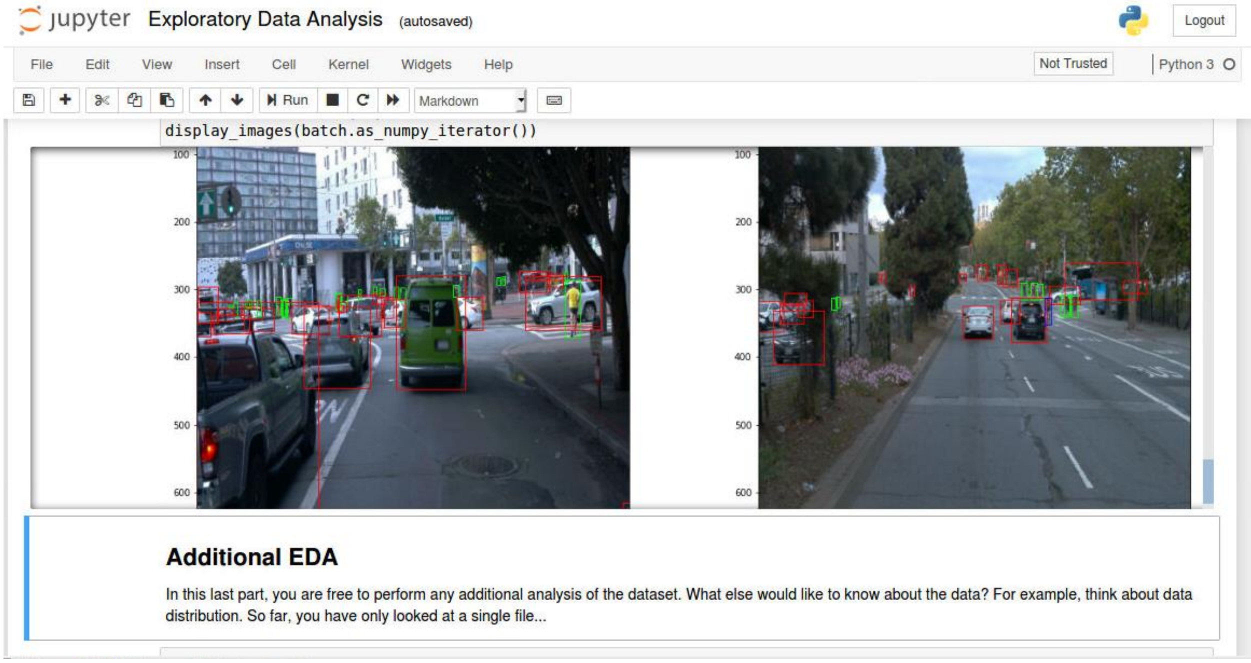
In this last part, you are free to perform any additional analysis of the dataset. What else would like to know about the data? For example, think about data distribution. So far, you have only looked at a single file...


jupyter
Exploratory Data Analysis (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
100
200
300
400
500
600
root@5635c4b…
Exploratory Da…

```
display_images(batch.as_numpy_iterator())
```

Additional EDA

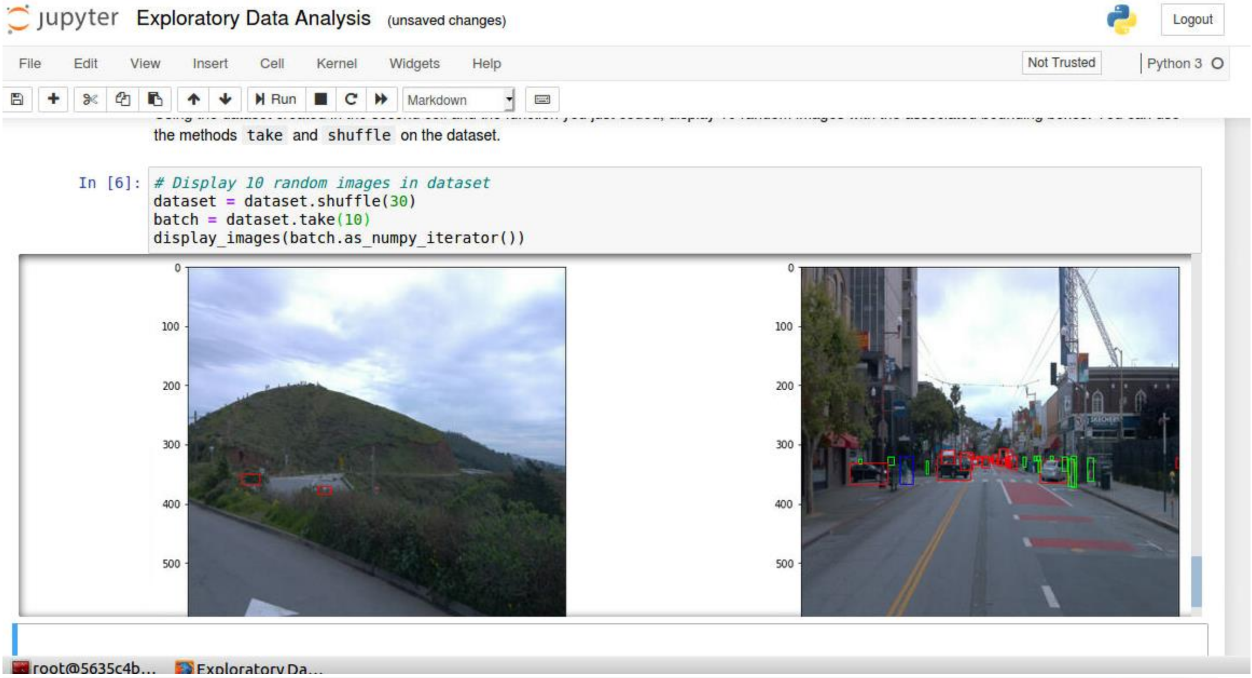
In this last part, you are free to perform any additional analysis of the dataset. What else would like to know about the data? For example, think about data distribution. So far, you have only looked at a single file...


jupyter
Exploratory Data Analysis (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
100
200
300
400
500
600

```
display_images(batch.as_numpy_iterator())
```

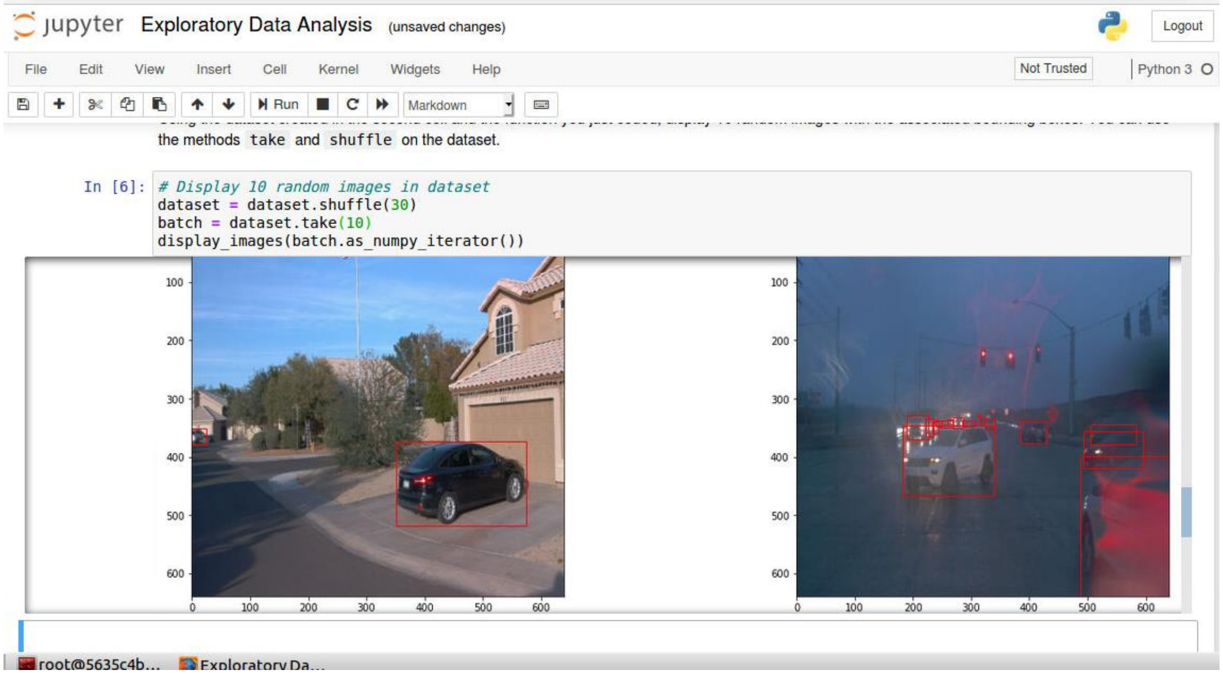
Additional EDA

In this last part, you are free to perform any additional analysis of the dataset. What else would like to know about the data? For example, think about data distribution. So far, you have only looked at a single file...


jupyter Exploratory Data Analysis (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
0
100
200
300
400
500
root@5635c4b... Exploratory Da...

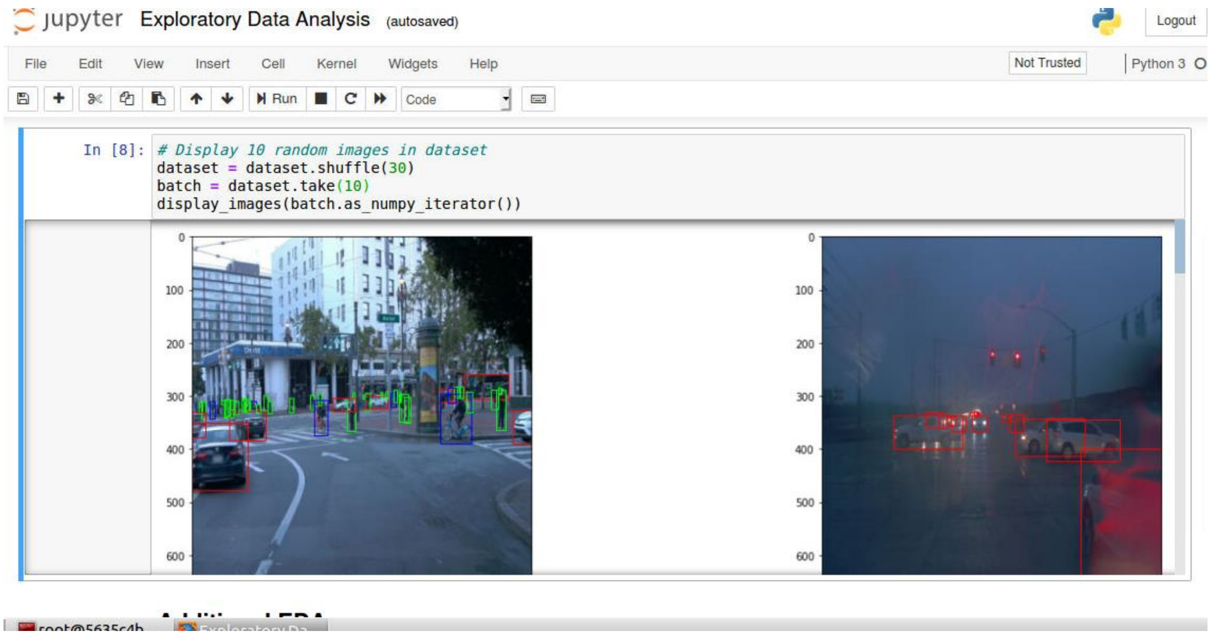
Using the dataset created in the second cell and the function you just coded, display 10 random images with the associated bounding boxes. You can use the methods `take` and `shuffle` on the dataset.

```
In [6]: # Display 10 random images in dataset
        dataset = dataset.shuffle(30)
        batch = dataset.take(10)
        display_images(batch.as_numpy_iterator())
```


jupyter Exploratory Data Analysis (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Markdown
100
200
300
400
500
600
0
root@5635c4b... Exploratory Da...

Using the dataset created in the second cell and the function you just coded, display 10 random images with the associated bounding boxes. You can use the methods `take` and `shuffle` on the dataset.

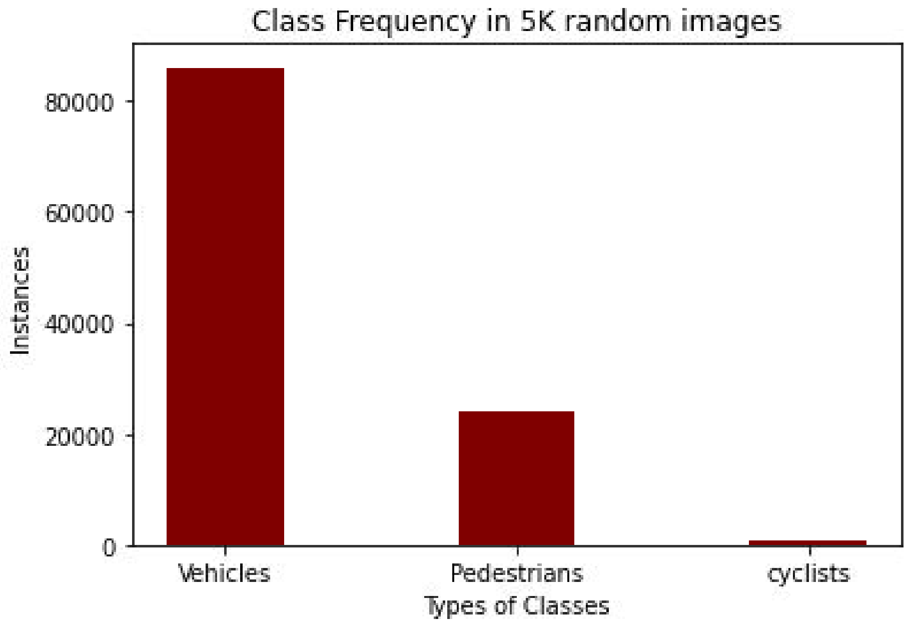
```
In [6]: # Display 10 random images in dataset
        dataset = dataset.shuffle(30)
        batch = dataset.take(10)
        display_images(batch.as_numpy_iterator())
```



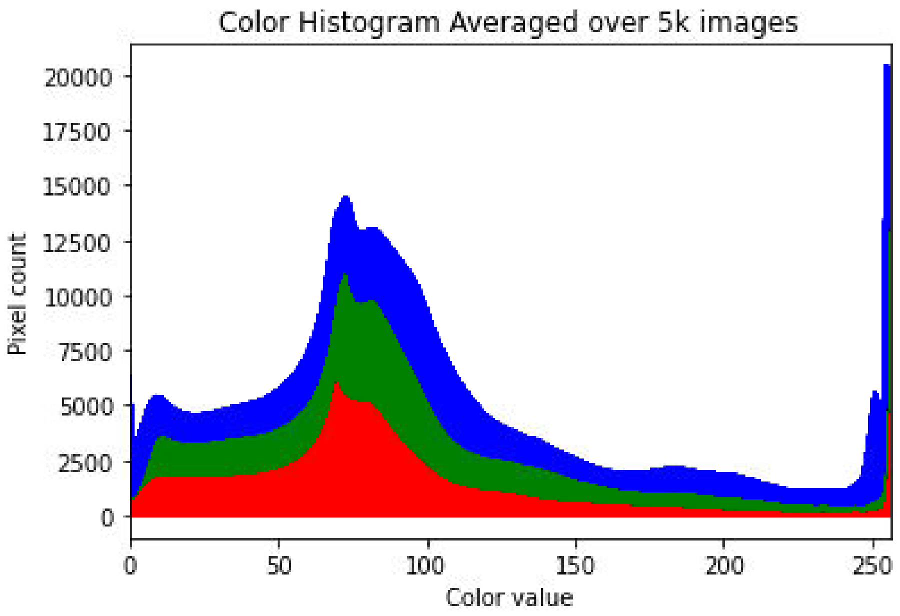
```
In [8]: # Display 10 random images in dataset
        dataset = dataset.shuffle(30)
        batch = dataset.take(10)
        display_images(batch.as_numpy_iterator())
```

The dataset consists of images with
Traffic: heavy traffic, no traffic, no detection of any vehicle
Types of roads: Highways, hill areas, Signal junctions, Residential area, Zebra crossing
Environment: variation of sun light, Foggy, Night etc.
Heavy Vehicles, Light vehicles, bicycles etc.

- *Additional EDA:*
  - Below Graph indicates the class instances out of 5K random image samples out of which 86090 were vehicles, 24004 were pedestrians, 613 were cyclists.



  - Below graph is the average pixel count of the color value in RGB channels individually.

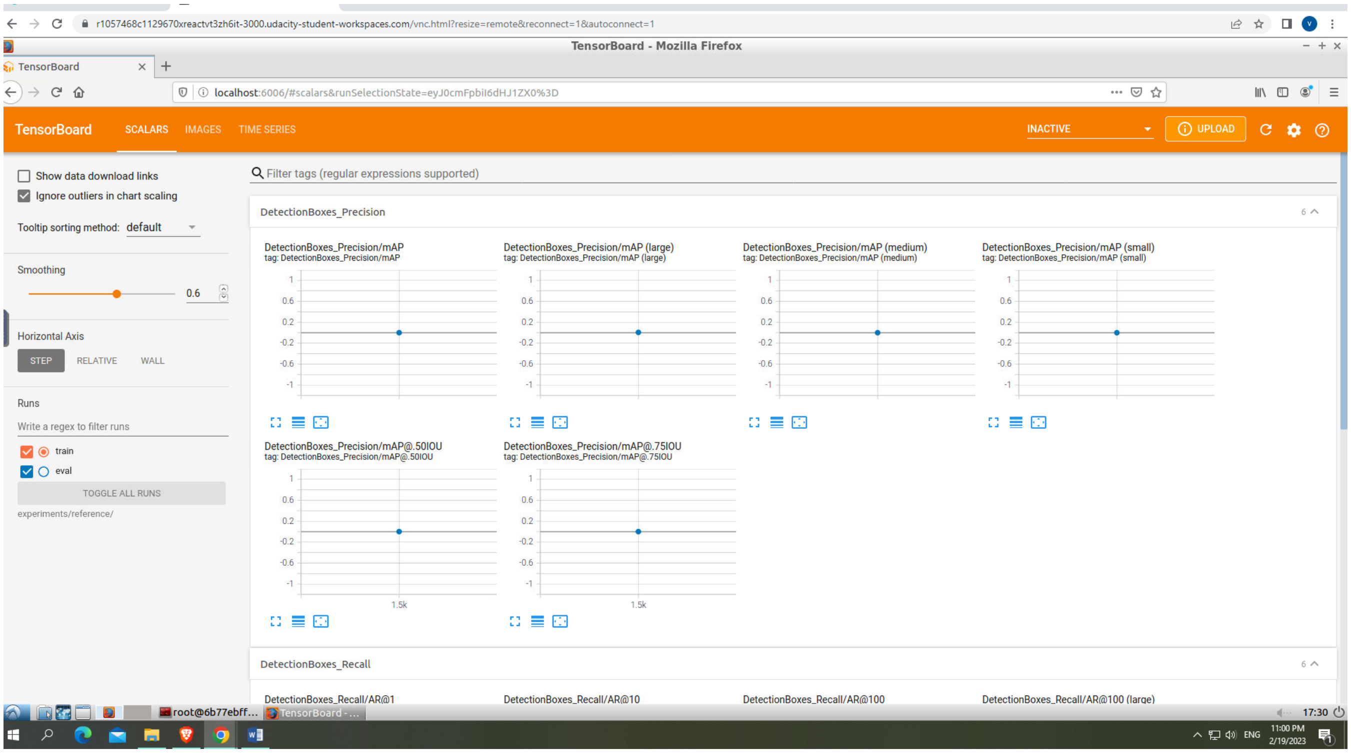


`

4. Training

i) Reference Experiment: The loss and learning rate charts of the pre-trained model training and evaluation results shown below. Based on the observation, the more number of iterations, the more drop in the learning rate of the model and high increase in the loss.

The classification loss and localization loss decreased with more number of iterations. Where as the normalized total loss, regularization loss has steep increase after 800 steps and slowly decreasing afterwards. The learning rate has been decreasing after 200 steps. The evaluation process is shown in blue dot as it ran for on epoch only.


r1057468c1129670xreactvt3zh6it-3000.udacity-student-workspaces.com/vnc.html?resize=remote&reconnect=1&autoconnect=1

TensorBoard - Mozilla Firefox

localhost:6006/#scalars&runSelectionState=eyJ0cmFpbiI6dHJ1ZX0%3D

☐ Show data download links
☑ Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing 0.6
Horizontal Axis: STEP
Runs
☑ train
☑ eval
experiments/reference/

DetectionBoxes_Precision

DetectionBoxes_Precision/mAP
tag: DetectionBoxes_Precision/mAP

DetectionBoxes_Precision/mAP (large)
tag: DetectionBoxes_Precision/mAP (large)

DetectionBoxes_Precision/mAP (medium)
tag: DetectionBoxes_Precision/mAP (medium)

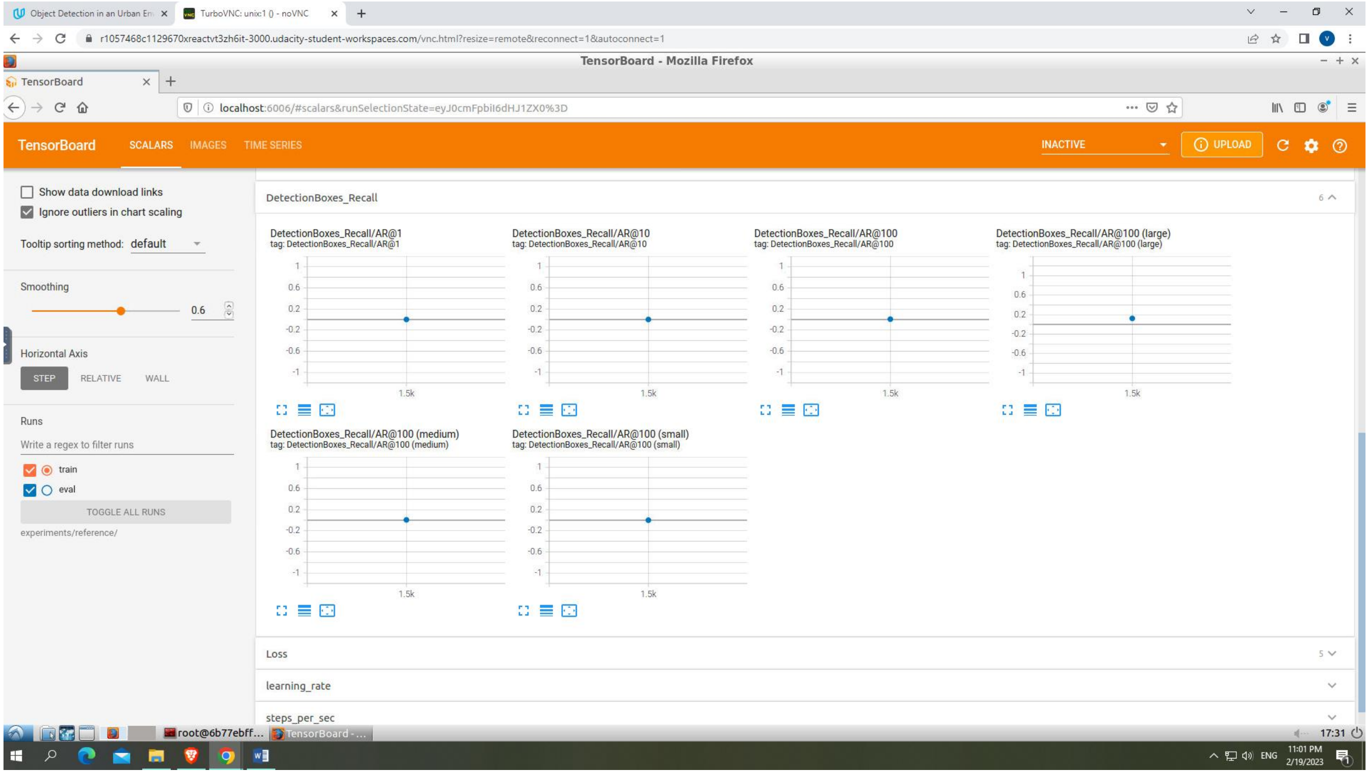
DetectionBoxes_Precision/mAP (small)
tag: DetectionBoxes_Precision/mAP (small)

DetectionBoxes_Precision/mAP@.50IOU
tag: DetectionBoxes_Precision/mAP@.50IOU

DetectionBoxes_Precision/mAP@.75IOU
tag: DetectionBoxes_Precision/mAP@.75IOU

DetectionBoxes_Recall

DetectionBoxes_Recall/AR@1 DetectionBoxes_Recall/AR@10 DetectionBoxes_Recall/AR@100 DetectionBoxes_Recall/AR@100 (large)


Object Detection in an Urban En
TurboVNC: unix:1 () - noVNC

r1057468c1129670xreactvt3zh6it-3000.udacity-student-workspaces.com/vnc.html?resize=remote&reconnect=1&autoconnect=1

TensorBoard - Mozilla Firefox

localhost:6006/#scalars&runSelectionState=eyJ0cmFpbiI6dHJ1ZX0%3D

☐ Show data download links
☑ Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing 0.6
Horizontal Axis: STEP
Runs
☑ train
☑ eval
experiments/reference/

DetectionBoxes_Recall

DetectionBoxes_Recall/AR@1
tag: DetectionBoxes_Recall/AR@1

DetectionBoxes_Recall/AR@10
tag: DetectionBoxes_Recall/AR@10

DetectionBoxes_Recall/AR@100
tag: DetectionBoxes_Recall/AR@100

DetectionBoxes_Recall/AR@100 (large)
tag: DetectionBoxes_Recall/AR@100 (large)

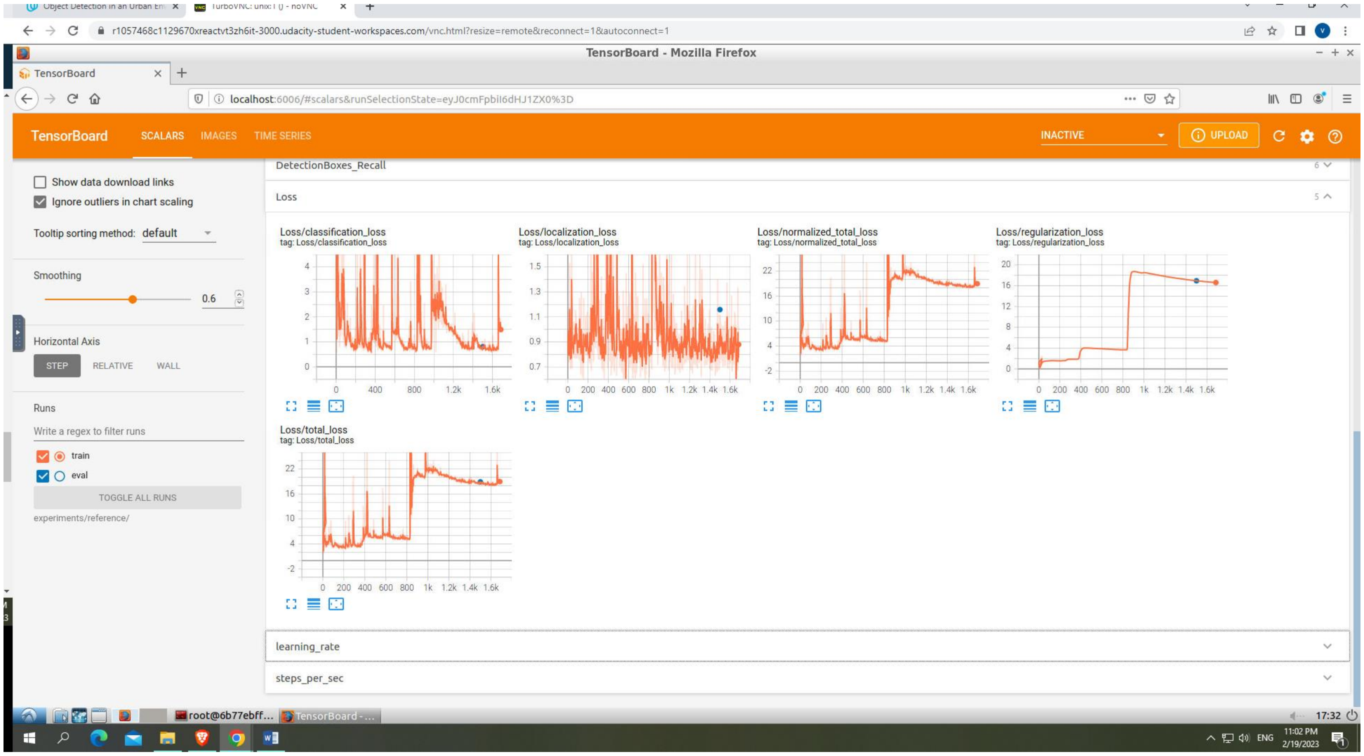
DetectionBoxes_Recall/AR@100 (medium)
tag: DetectionBoxes_Recall/AR@100 (medium)

DetectionBoxes_Recall/AR@100 (small)
tag: DetectionBoxes_Recall/AR@100 (small)

Loss

learning_rate

steps_per_sec



□ Show data download links

☑ Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis: STEP RELATIVE WALL

Runs

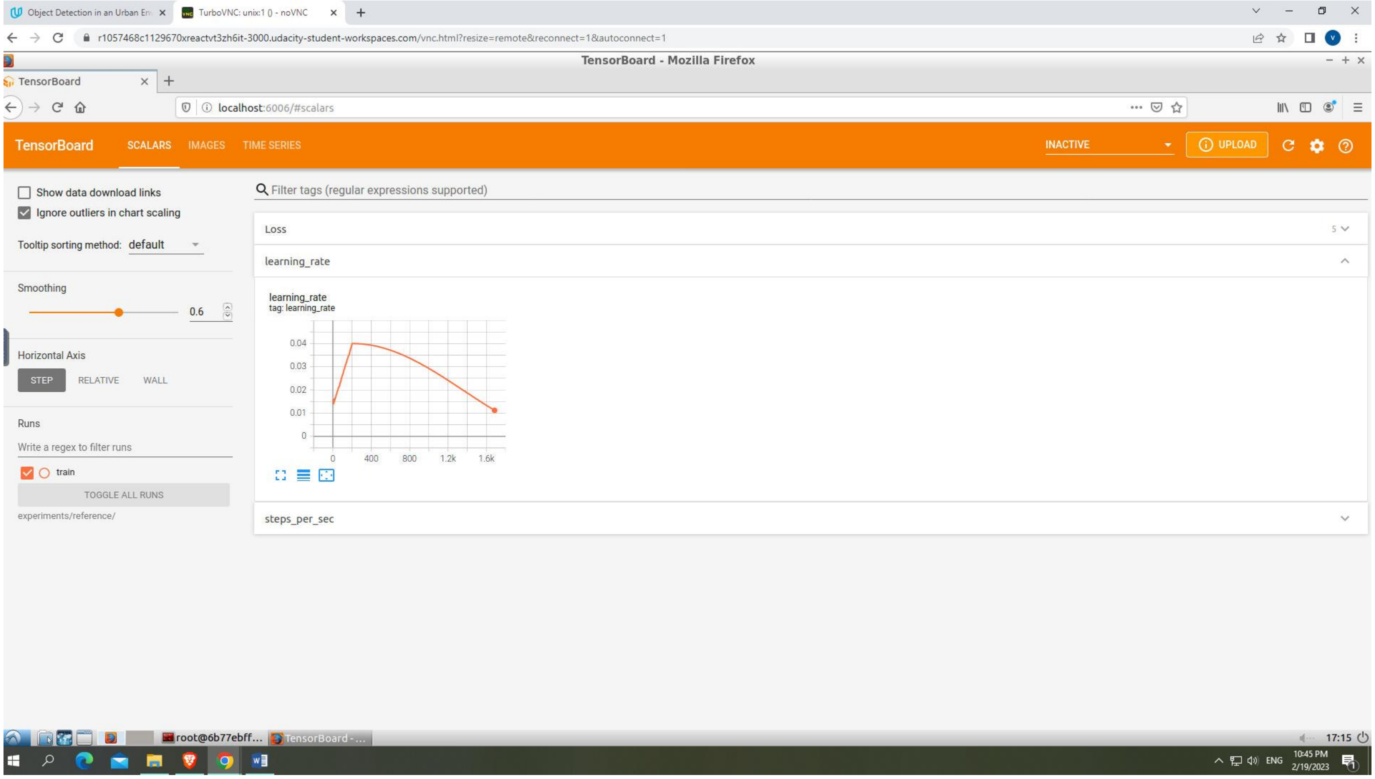
Write a regex to filter runs

☑ train

☑ eval

TOGGLE ALL RUNS

experiments/reference/


Object Detection in an Urban En
TurboVNC: unix:1 () - noVNC
r1057468c1129670xreactvt3zh6it-3000.udacity-student-workspaces.com/vnc.html?resize=remote&reconnect=1&autoconnect=1
TensorBoard - Mozilla Firefox
TensorBoard
localhost:6006/#scalars
TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
Filter tags (regular expressions supported)
Loss
5
learning_rate
learning_rate
tag: learning_rate
0.04
0.03
0.02
0.01
0
0
400
800
1.2k
1.6k
steps_per_sec
root@6b77ebff...
TensorBoard - ...
17:15
ENG
10:45 PM
2/19/2023

□ Show data download links

☑ Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis: STEP RELATIVE WALL

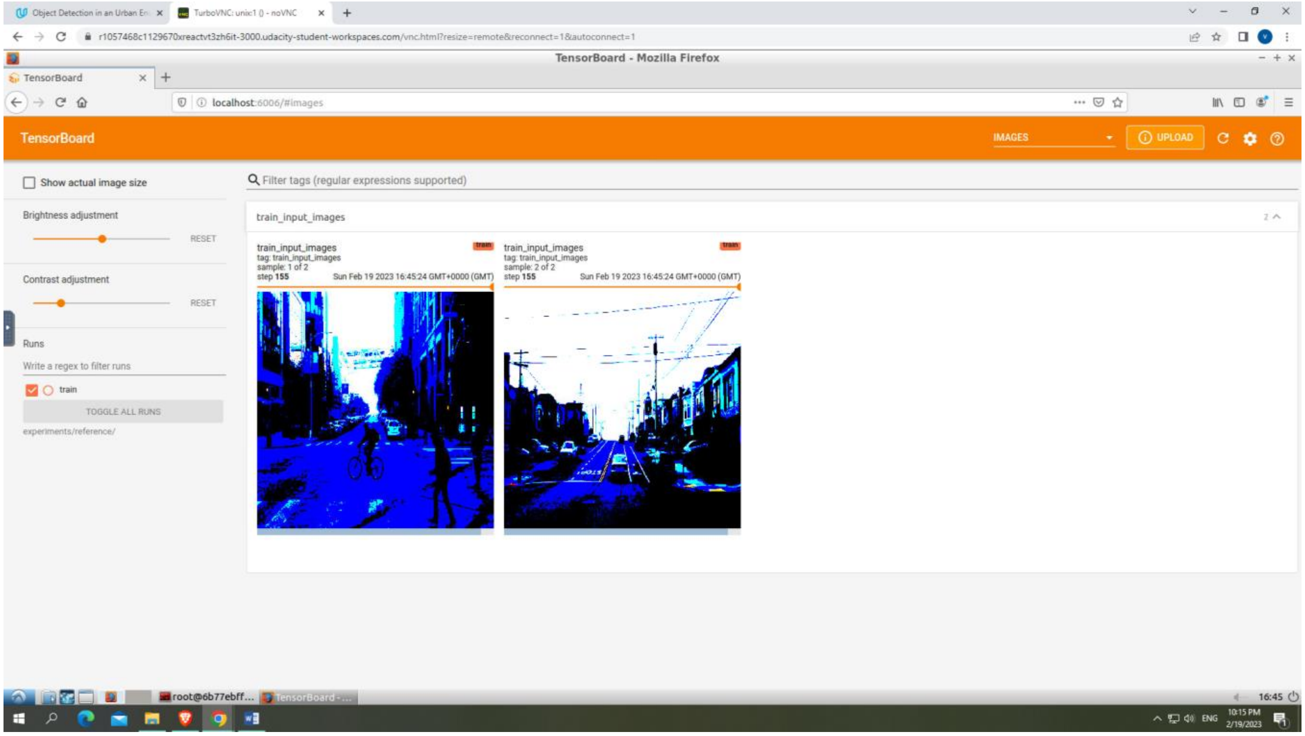
Runs

Write a regex to filter runs

☑ train

TOGGLE ALL RUNS

experiments/reference/



TensorBoard - Mozilla Firefox

localhost:6006/#images

☐ Show actual image size

Brightness adjustment

Contrast adjustment

Runs

☑ ○ train

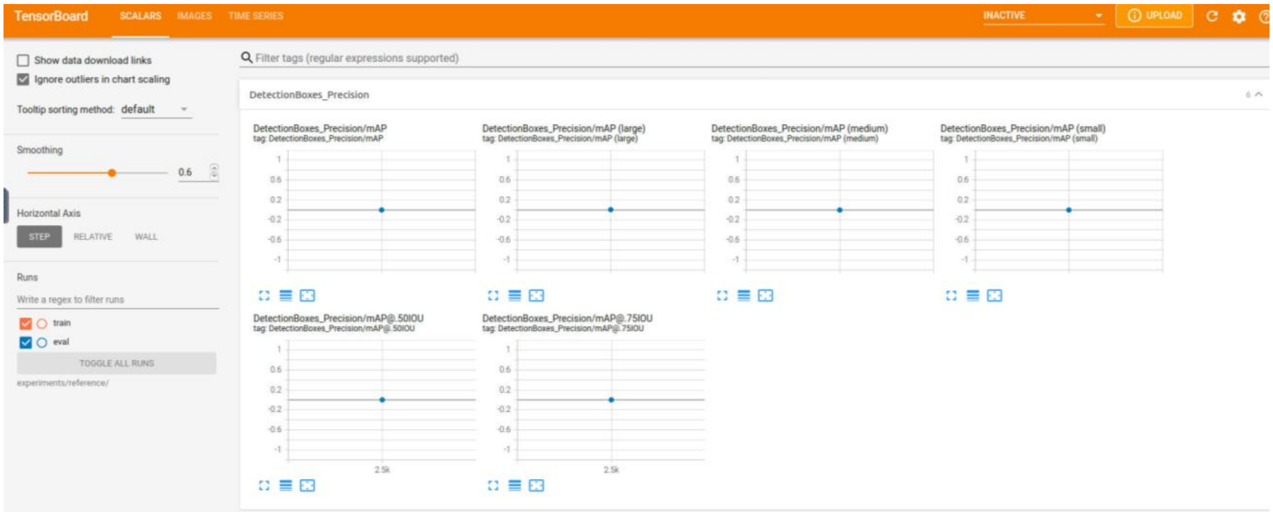
experiments/reference/

train_input_images

train_input_images
tag: train_input_images
sample: 1 of 2
step 155 Sun Feb 19 2023 16:45:24 GMT+0000 (GMT)

train_input_images
tag: train_input_images
sample: 2 of 2
step 155 Sun Feb 19 2023 16:45:24 GMT+0000 (GMT)

Object Detection in an Urban En
TurboVNC: unix:1 () - noVNC
r1057468c1129670xreactvt3zh6it-3000.udacity-student-workspaces.com/vnc.html?resize=remote&reconnect=1&autoconnect=1
TensorBoard
IMAGES
UPLOAD
Filter tags (regular expressions supported)
RESET
RESET
Write a regex to filter runs
TOGGLE ALL RUNS
train
train
root@6b77ebff...
TensorBoard - ...
16:45
ENG
10:15 PM
2/19/2023



TensorBoard

☐ Show data download links
☑ Ignore outliers in chart scaling

Tooltip sorting method: default

Smoothing 0.6

Horizontal Axis: STEP

Runs

☑ ○ train
☑ ○ eval

experiments/reference/

DetectionBoxes_Precision

DetectionBoxes_Precision/mAP
tag: DetectionBoxes_Precision/mAP

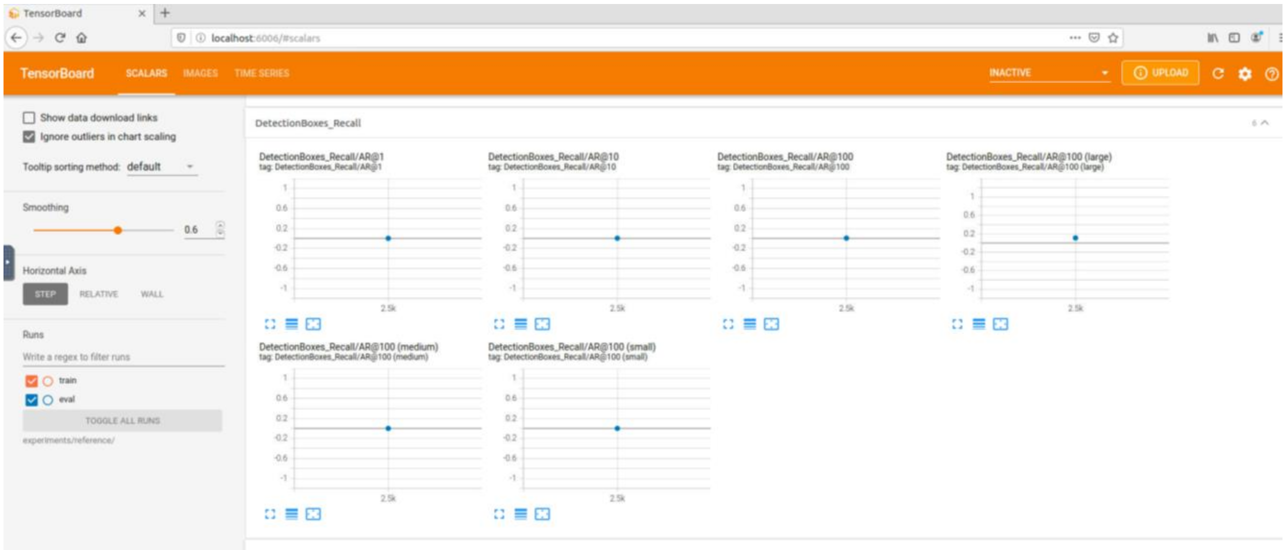
DetectionBoxes_Precision/mAP (large)
tag: DetectionBoxes_Precision/mAP (large)

DetectionBoxes_Precision/mAP (medium)
tag: DetectionBoxes_Precision/mAP (medium)

DetectionBoxes_Precision/mAP (small)
tag: DetectionBoxes_Precision/mAP (small)

DetectionBoxes_Precision/mAP@.50IOU
tag: DetectionBoxes_Precision/mAP@.50IOU

DetectionBoxes_Precision/mAP@.75IOU
tag: DetectionBoxes_Precision/mAP@.75IOU



TensorBoard
localhost:6006/#scalars
TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD

☐ 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

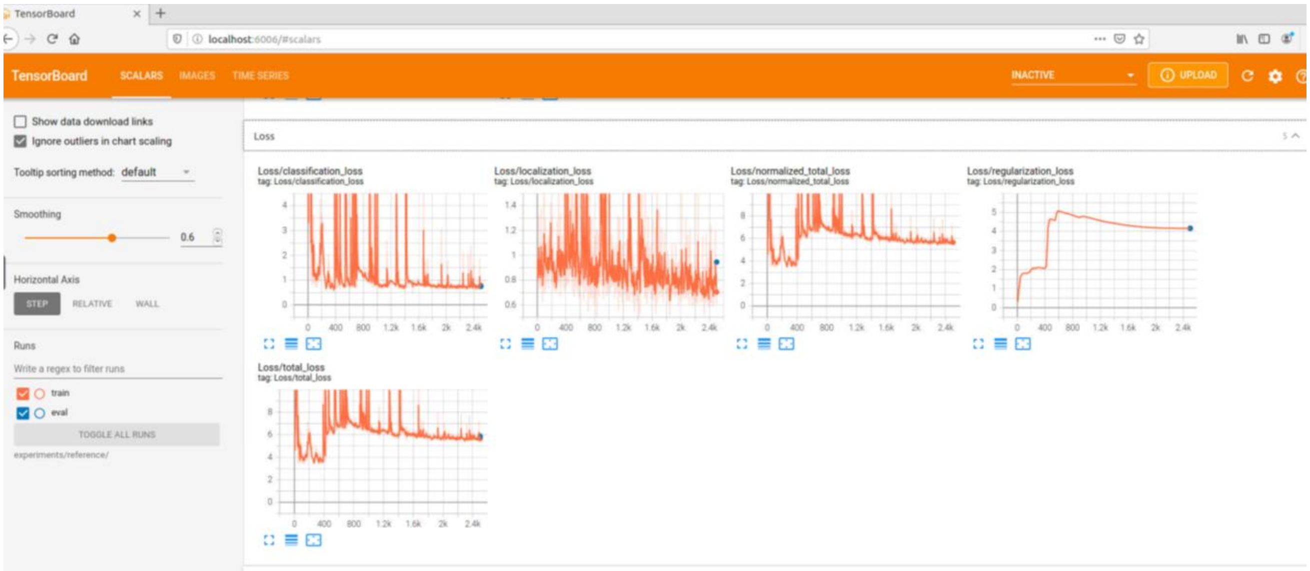
☑ ○ eval

TOGGLE ALL RUNS

experiments/reference/

DetectionBoxes_Recall

DetectionBoxes_Recall/AR@1
tag: DetectionBoxes_Recall/AR@1
DetectionBoxes_Recall/AR@10
tag: DetectionBoxes_Recall/AR@10
DetectionBoxes_Recall/AR@100
tag: DetectionBoxes_Recall/AR@100
DetectionBoxes_Recall/AR@100 (large)
tag: DetectionBoxes_Recall/AR@100 (large)
DetectionBoxes_Recall/AR@100 (medium)
tag: DetectionBoxes_Recall/AR@100 (medium)
DetectionBoxes_Recall/AR@100 (small)
tag: DetectionBoxes_Recall/AR@100 (small)
1
0.6
0.2
-0.2
-0.6
-1
2.5k



☐ 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

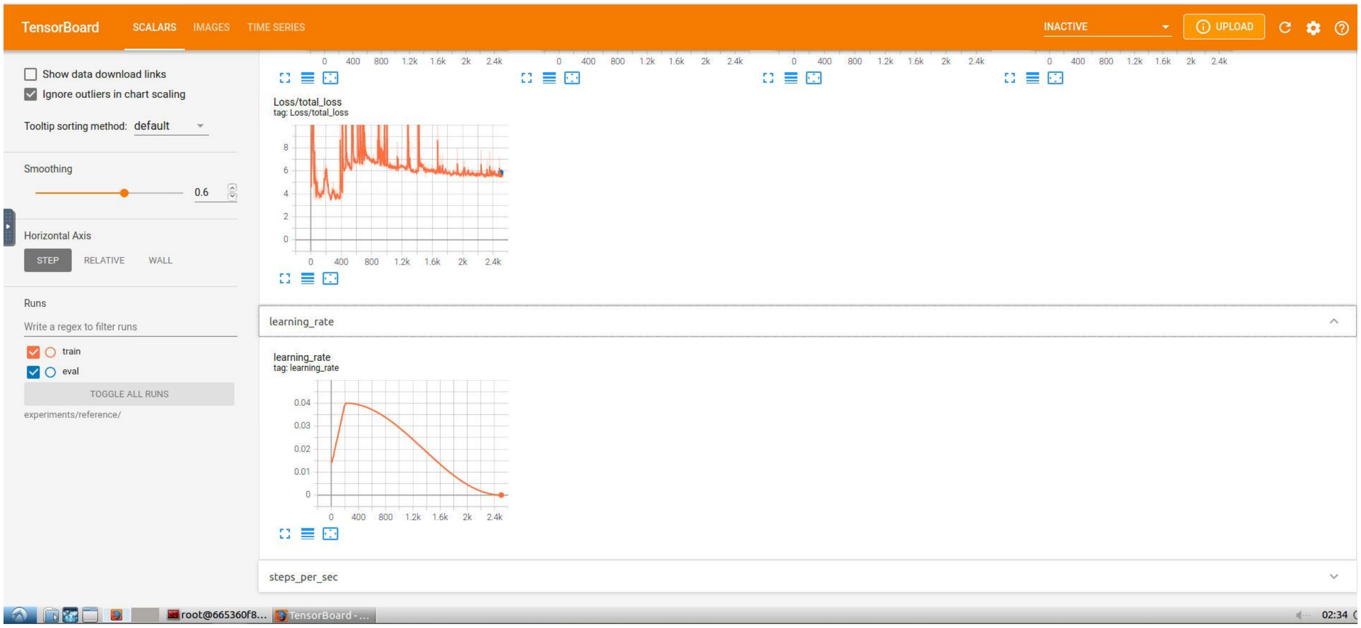
☑ ○ train

☑ ○ eval

TOGGLE ALL RUNS

experiments/reference/

Loss



☐ Show data download links
☑ Ignore outliers in chart scaling

Tooltip sorting method: default

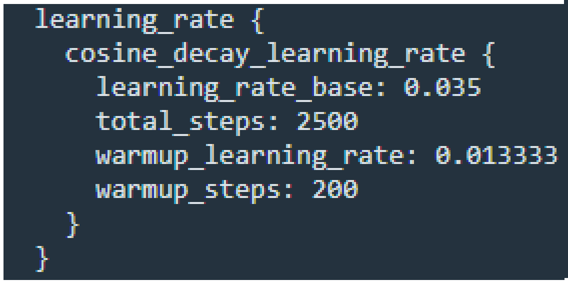
Smoothing 0.6

Horizontal Axis: STEP RELATIVE WALL

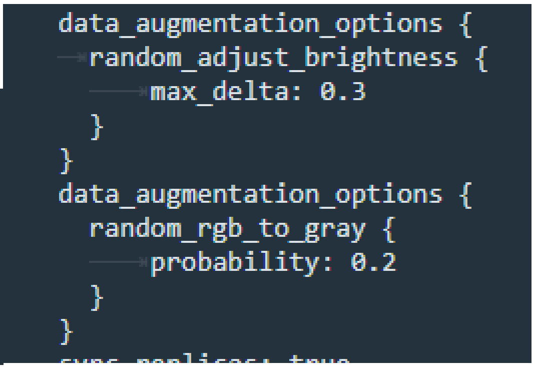
Runs
Write a regex to filter runs
☑ ○ train
☑ ○ eval
experiments/reference/

ii) Improve on the Reference:

In the config file added the following data augmentations and decreased the learning rate. The loss is reduced ~3.



```
learning_rate {
  cosine_decay_learning_rate {
    learning_rate_base: 0.035
    total_steps: 2500
    warmup_learning_rate: 0.013333
    warmup_steps: 200
  }
}
```



```
data_augmentation_options {
  random_adjust_brightness {
    max_delta: 0.3
  }
}
data_augmentation_options {
  random_rgb_to_gray {
    probability: 0.2
  }
}
sync_replicas: true
```

Data augmentation is a set of techniques to increase the amount of data artificially by generating new data points from existing data.

I have considered two data augmentations as shown above

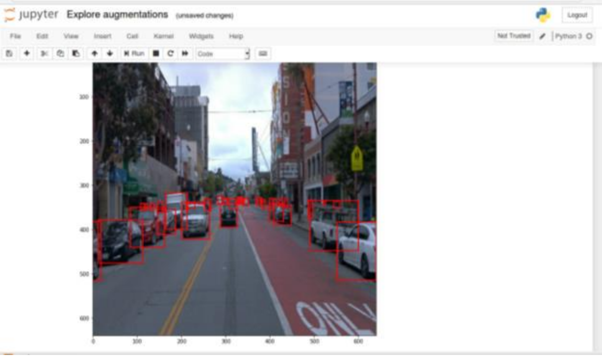
1. Random Adjust Brightness

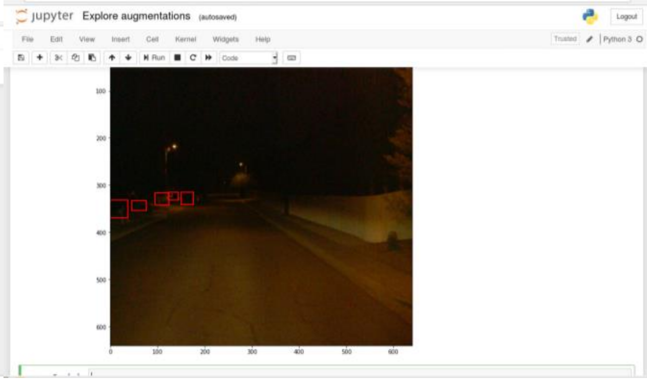
This data augmentation will increase the brightness of images randomly with maximum delta of 0.3, which seems effective in increasing the efficiency of the training model. This augmentation will help to overcome the difficulty of detecting objects against glare, varying brightness.

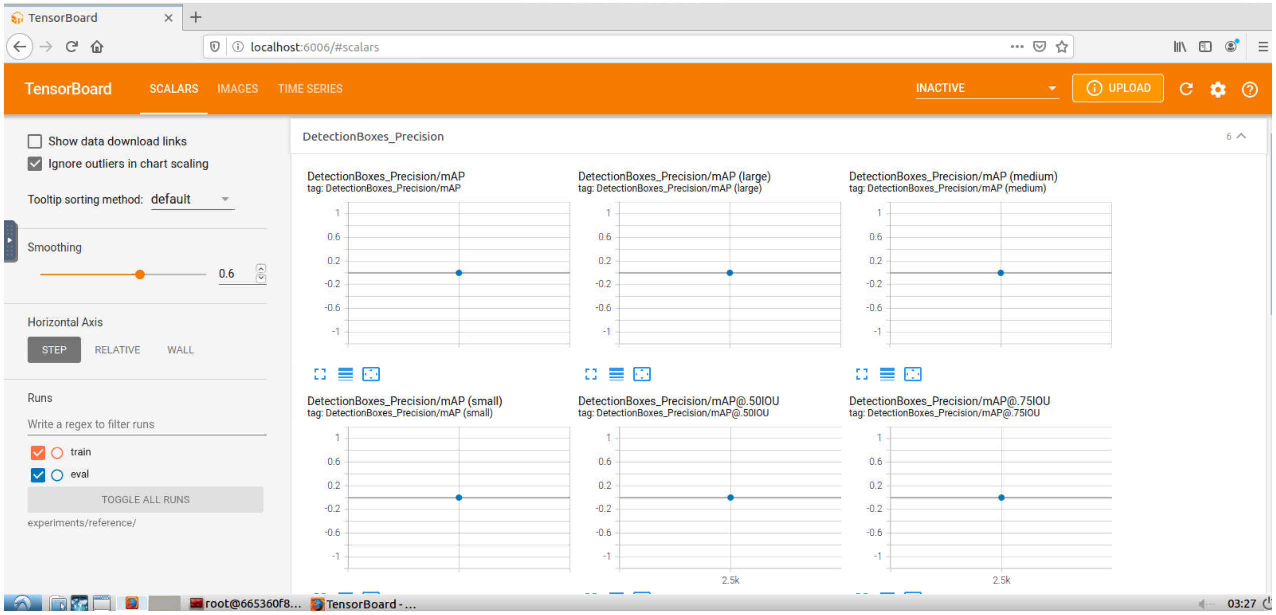
2. Random RGB to gray

This data augmentation converts the RGB image into gray scale, which reduces the color complexity and can negate the shadows resulting in better object detection.

Reduced the Learning Rate so that model gets trained to fit better with data, also its important to select a value so that it doesn't over fit.


jupyter Explore augmentations (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
100
200
300
400
500
600
0
100
200
300
400
500
600


jupyter Explore augmentations (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
100
200
300
400
500
600
0
100
200
300
400
500
600



☐ 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

☑ ○ eval

TOGGLE ALL RUNS

experiments/reference/

DetectionBoxes_Precision 6

DetectionBoxes_Precision/mAP
tag: DetectionBoxes_Precision/mAP

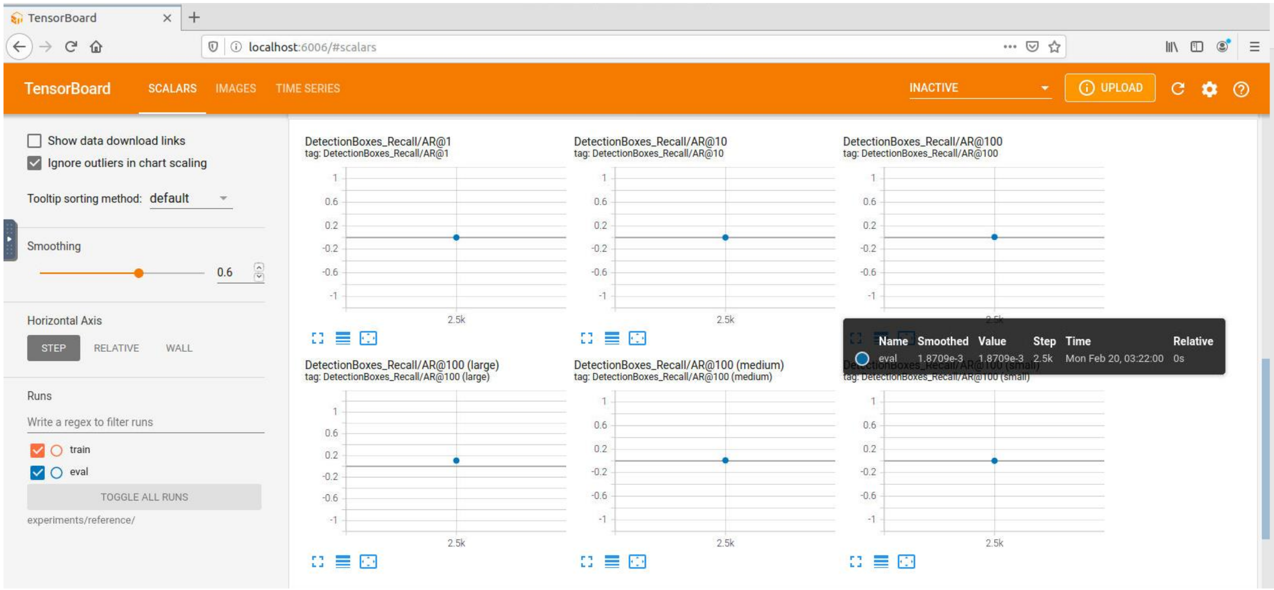
DetectionBoxes_Precision/mAP (large)
tag: DetectionBoxes_Precision/mAP (large)

DetectionBoxes_Precision/mAP (medium)
tag: DetectionBoxes_Precision/mAP (medium)

DetectionBoxes_Precision/mAP (small)
tag: DetectionBoxes_Precision/mAP (small)

DetectionBoxes_Precision/mAP@.50IOU
tag: DetectionBoxes_Precision/mAP@.50IOU

DetectionBoxes_Precision/mAP@.75IOU
tag: DetectionBoxes_Precision/mAP@.75IOU



☐ 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
☑ ○ eval

TOGGLE ALL RUNS

experiments/reference/

DetectionBoxes_Recall/AR@1
tag: DetectionBoxes_Recall/AR@1

DetectionBoxes_Recall/AR@10
tag: DetectionBoxes_Recall/AR@10

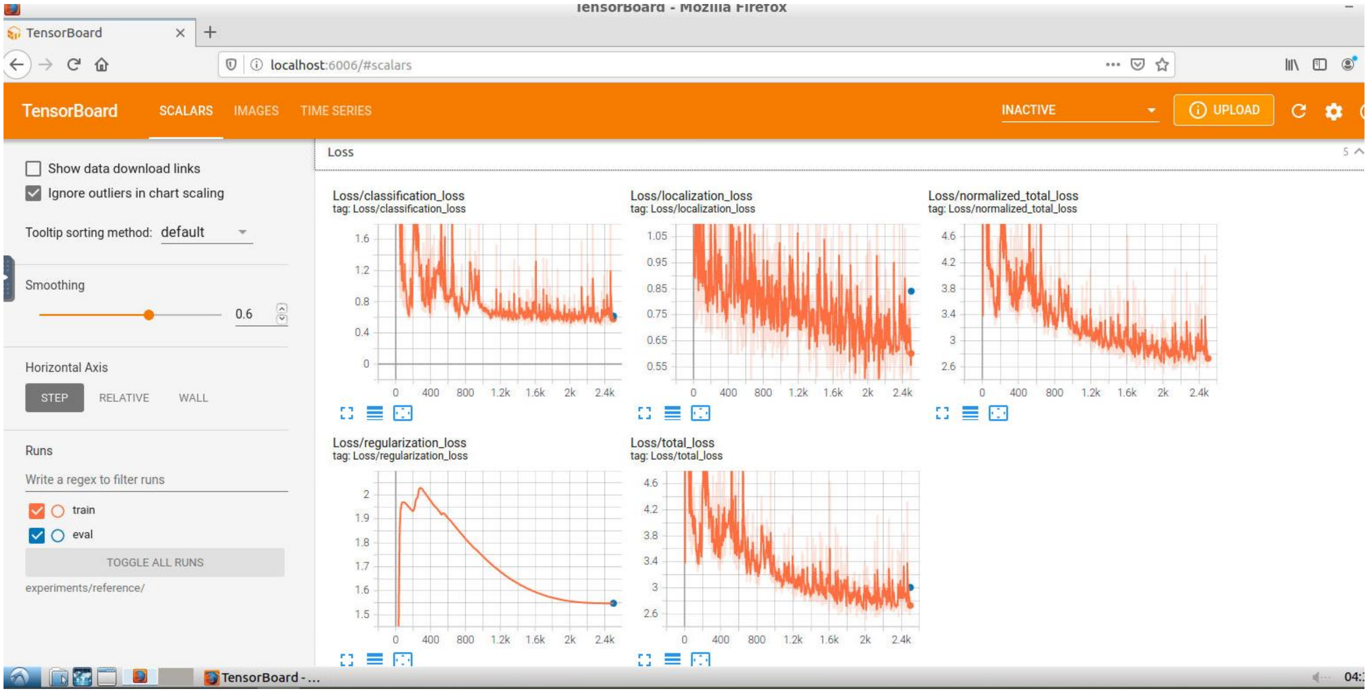
DetectionBoxes_Recall/AR@100
tag: DetectionBoxes_Recall/AR@100

DetectionBoxes_Recall/AR@100 (large)
tag: DetectionBoxes_Recall/AR@100 (large)

DetectionBoxes_Recall/AR@100 (medium)
tag: DetectionBoxes_Recall/AR@100 (medium)

DetectionBoxes_Recall/AR@100 (small)
tag: DetectionBoxes_Recall/AR@100 (small)

```
Name  Smoothed   Value      Step  Time                   Relative
eval  1.8709e-3  1.8709e-3  2.5k  Mon Feb 20, 03:22:00   0s
```



☐ 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
☑ ○ eval

TOGGLE ALL RUNS

experiments/reference/

Loss

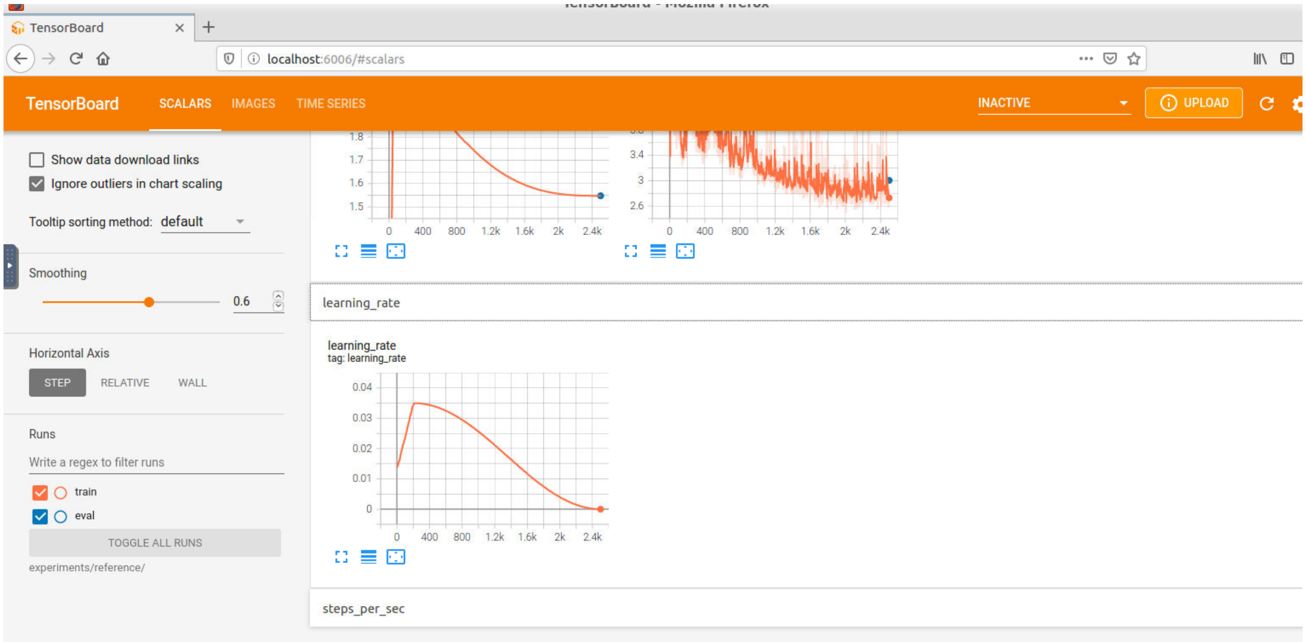
Loss/classification_loss
tag: Loss/classification_loss

Loss/localization_loss
tag: Loss/localization_loss

Loss/normalized_total_loss
tag: Loss/normalized_total_loss

Loss/regularization_loss
tag: Loss/regularization_loss

Loss/total_loss
tag: Loss/total_loss



☐ 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

☑ ○ eval

experiments/reference/

learning_rate

learning_rate
tag: learning_rate

steps_per_sec