Object Detection in an urban Environment

1- PROJECT OVERVIEW

The main objective of the Project is to detect objects in an Urban Environment. Exploratory data analysis was performed on the processed dataset. Three classes of objects have been detected in different colored Bounding boxes (Vehicles in Red, Pedestrians in Blue, Cyclist in Green). A function display_images have been implemented in Exploratory data Analysis Notebook to display images in the dataset. Later 10 images were randomly displayed out of the dataset. The model was trained using the Waymo dataset and the results were evaluated. The results of the training and evaluation have been plotted in TensorBoard. The graphs obtained in the first training did not yield good results, so, training was performed by changing some parameters. Later data augmentation techniques were used manipulate the images. Techniques like Random adjust Brightness, Random adjust contrast, Random adjust Hue etc were used. Later the model was tested in an urban environment.

Object detection is one of the most important functionalities in Autonomous Driving. It enables the car to see and understand the environment around It. The vehicle must stop if the detected object is near to the car otherwise the car may collide with the obstacle. If because of reasons like rain and fog, the computer fails to detect the obstacle, then it may lead to fatal accidents.

2- SETUP

Step1: The file named Exploratory data analysis.ipynb has the code required for display images function.

Step2: new_pipeline.config file has been stored in home/workspace/experiments/reference. It has the reference to various data augmentation techniques.

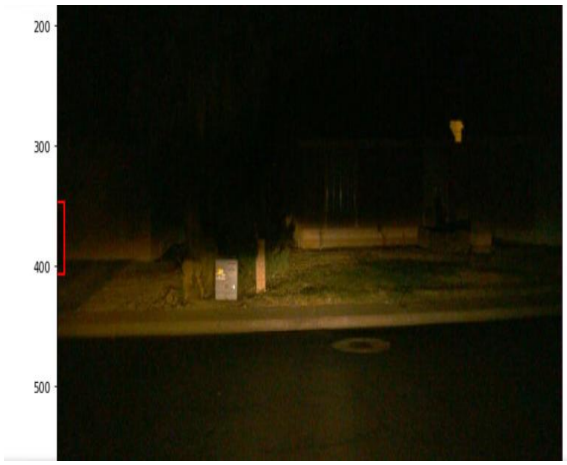
Step3: Explore Augmentations.ipynb file has the implementations for Data augmentations

3- DATASET

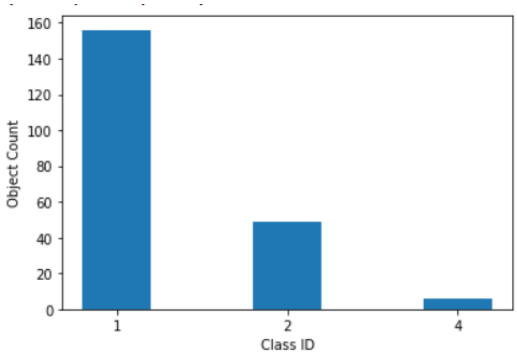
1- Dataset Analysis

The Waymo open data set has been used for the process. The data was present in /home/workspace/data folder.

The data was divided into 3 folders namely Train, Val and Test. The data was present in TF record format. The data set consisted of different types of areas like urban and semi urban. The dataset has images which had different classes of objects like cars, pedestrian, and bicyclists. There were objects which were distinct and Occluded. The objects were of different sizes like big, small and very small. The dataset consisted of images in different conditions of lighting. The objects were present in different weather conditions. Also, the objects detected were most of the time present at the center of the image. Object detections at the left and the right side of the image were not very common.


200
300
400
500


300
400
500



The dataset is very skewed as we can see from the graph above. This graph shows Class ID vs Object Count. The class 1(Vehicles) has the highest object count followed by Class 2 (Pedestrian) and Class 4 (Bicyclist) respectively.

2- Cross Validation

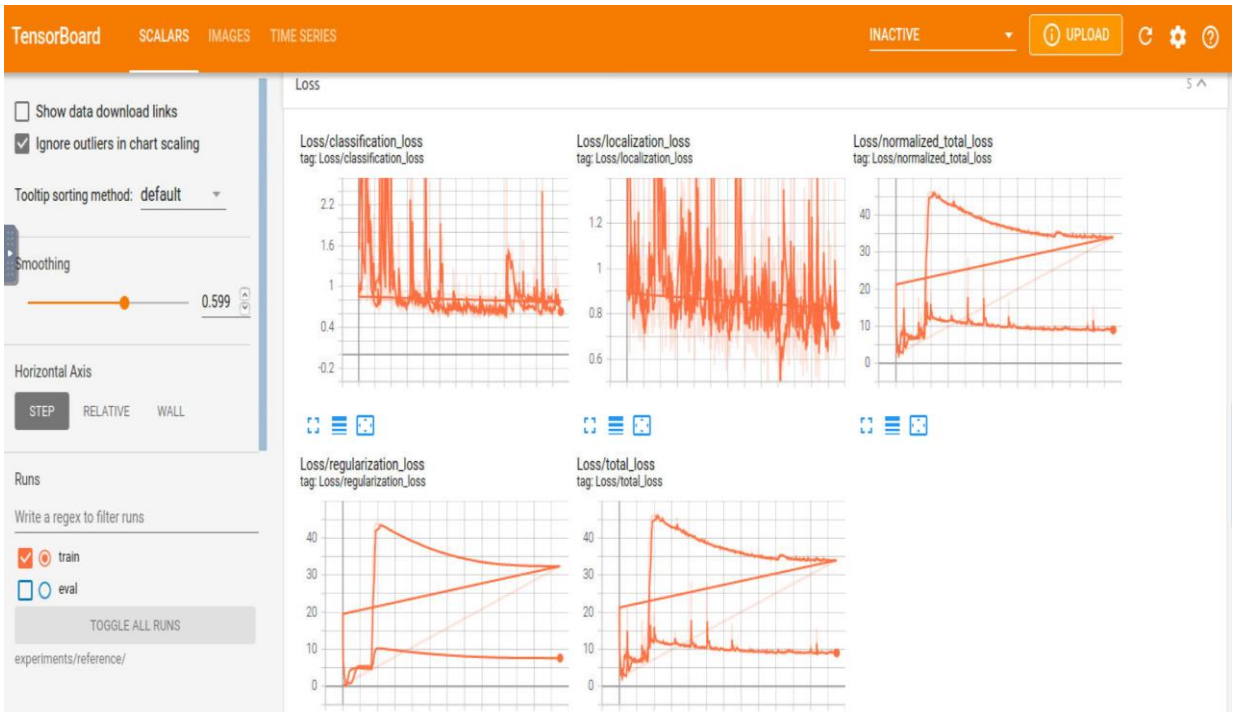
The data has already been split as training, Val and test folders. There are around 87 Tf Record files in Training, around 10 in Val and 3 in Test folder.

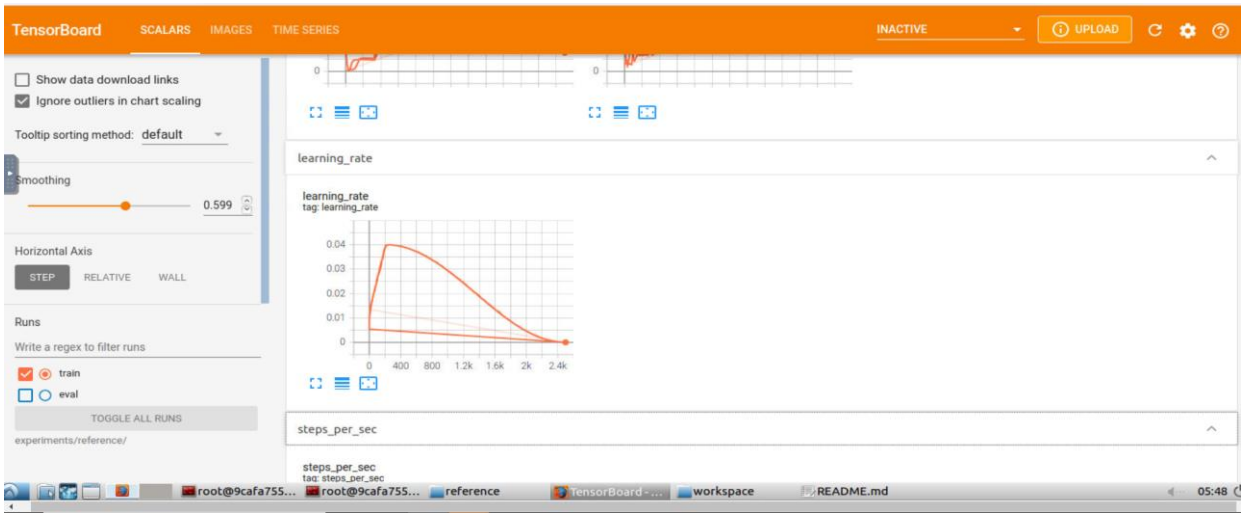
Based on the results obtained on the Cross-validation set, The model can be once again trained with augmented dataset so that the model performance can be improved.

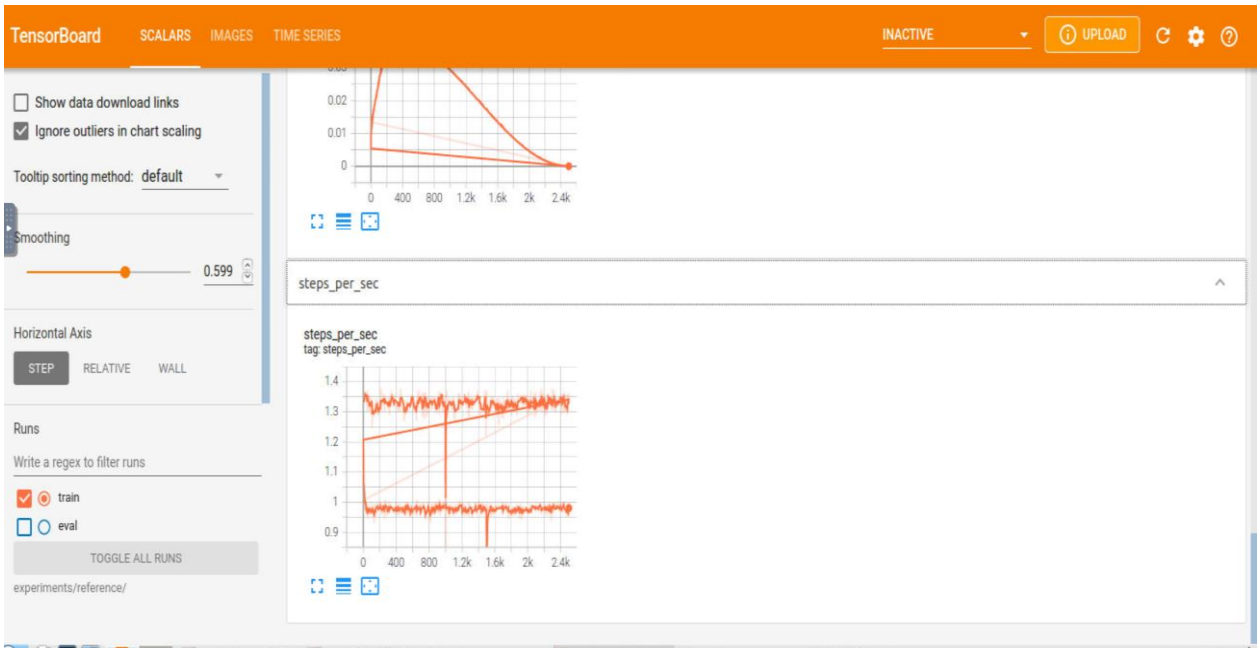
3- TRAINING

1- Reference Experiment (Phase 1)

These are the outputs from the tensor board after running the reference experiment.


TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.599
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


TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.599
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
eval
TOGGLE ALL RUNS
experiments/reference/
learning_rate
learning_rate
tag: learning_rate
steps_per_sec
steps_per_sec
tag: steps_per_sec
root@9cafa755...
root@9cafa755...
reference
TensorBoard - ...
workspace
README.md
05:48



My Observation: The graphs were obtained during the training Process. I observed that training for one more iteration improved the loss to some extent.

Note: I was Unable to detect the tensor board so I retrained the model with the same dataset. So I ended up training 2 times. They are marked as 1st and 2nd Iteration.

Classification Loss: Not much change in 1st and 2nd iteration

Localization Loss: Not Much Change

Normalized total Loss: Loss saturated at value 10 in the second iteration. In 1st iteration it was around 35.

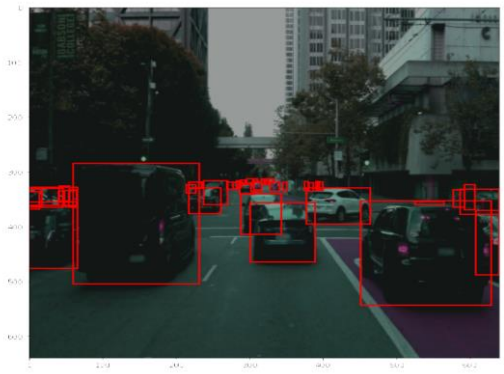
Regularization loss: It saturated at value 10 in the second iteration. In the first iteration it was around 35.

Total Loss: The loss saturated at around 10. In the first iteration it was around 35.

2- Improve on the Reference- DATA AUGMENTATIONS

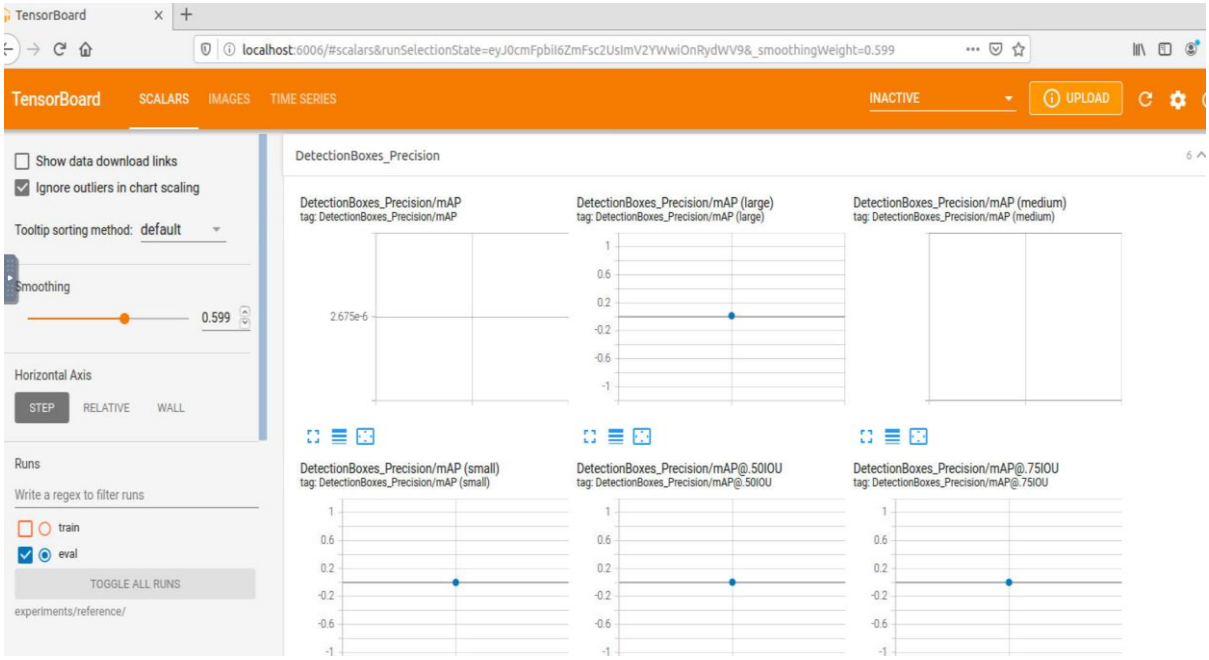
Since the output of the Evaluation is not very good. The data should be augmented so that It can be further used for the training. The following Augmentation techniques have been used and the new_pipeline.config has been modified.

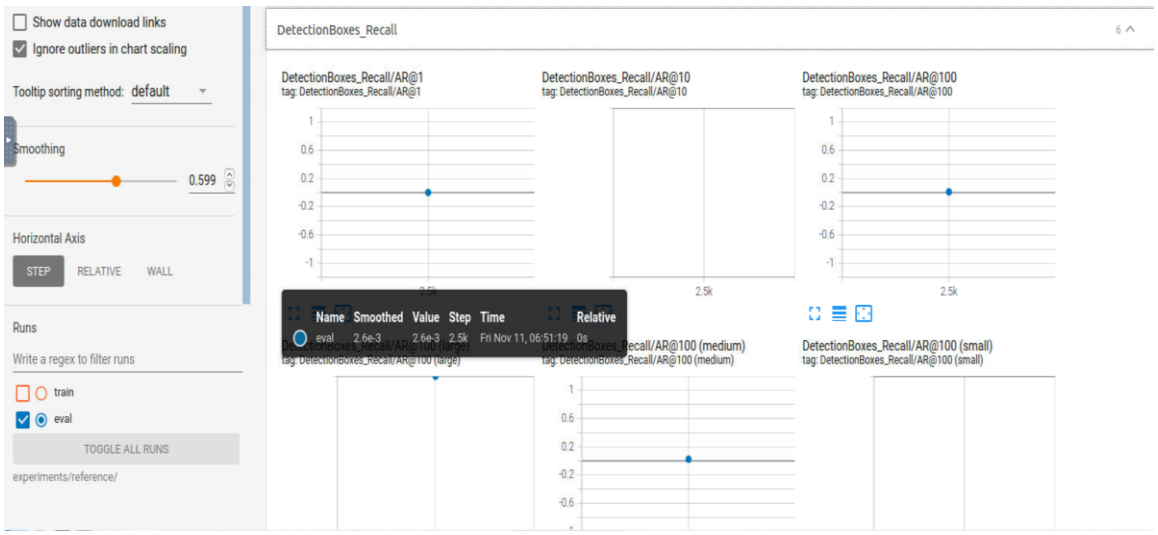
1- Random Crop Image
2- Random adjust Brightness
3- Random Adjust Contrast
4- Random Distort Colour
5- Random adjust Saturation
6- Random Adjust hue

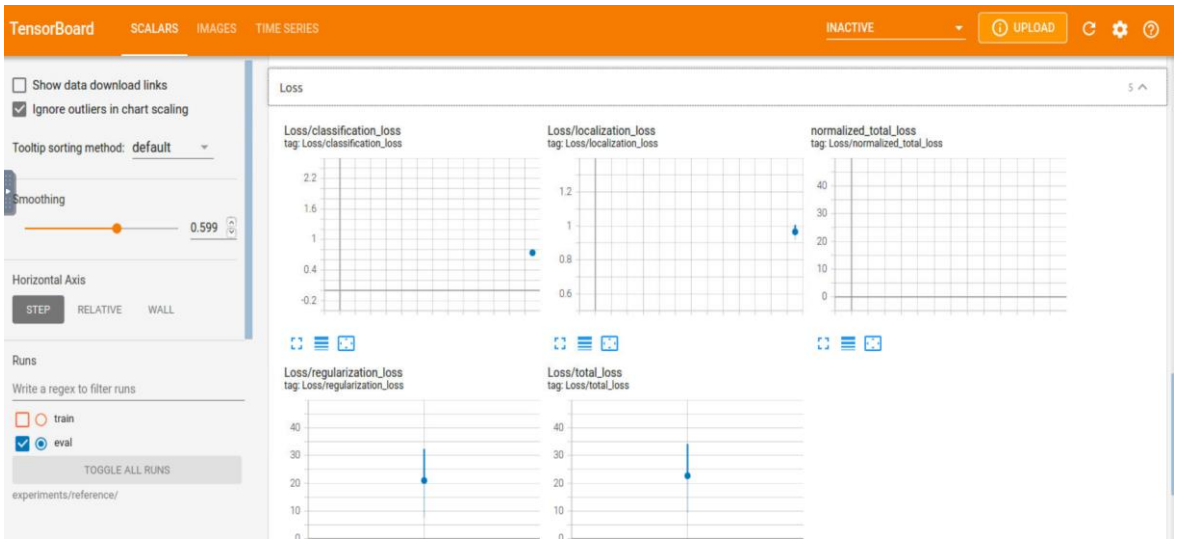


4- Cross- Validation (Phase 1)

There are around 86 TF record files for Model training. Around 10 TF Record for Model validation. After the model validation, the following graphs were obtained. They have been run only for 1 epoch.


TensorBoard
localhost:6006/#scalars&runSelectionState=eyJ0cmFpbil6ZmFsc2UsImV2YWwiOnRydWV9&_smoothingWeight=0.599
TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
Show data download links
Ignore outliers in chart scaling
Tooltip sorting method: default
Smoothing
0.599
Horizontal Axis
STEP
RELATIVE
WALL
Runs
Write a regex to filter runs
train
eval
TOGGLE ALL RUNS
experiments/reference/
DetectionBoxes_Precision
DetectionBoxes_Precision/mAP
tag: DetectionBoxes_Precision/mAP
2.675e-6
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
0.599
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
Name Smoothed Value Step Time Relative
eval 2.6e-3 2.6e-3 2.5k Fri Nov 11, 06:51:19 0s
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)

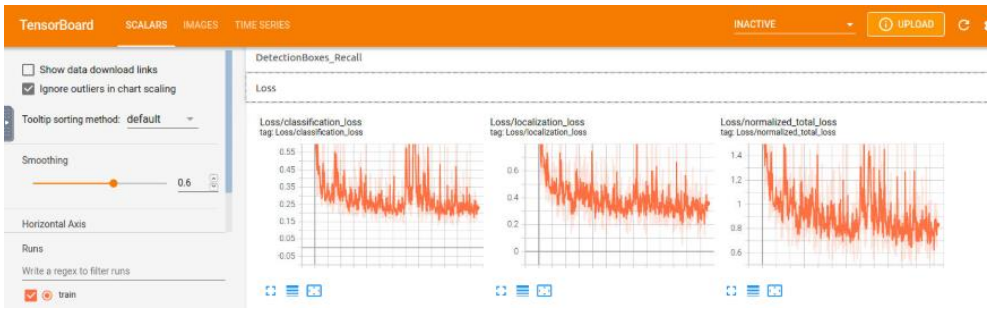


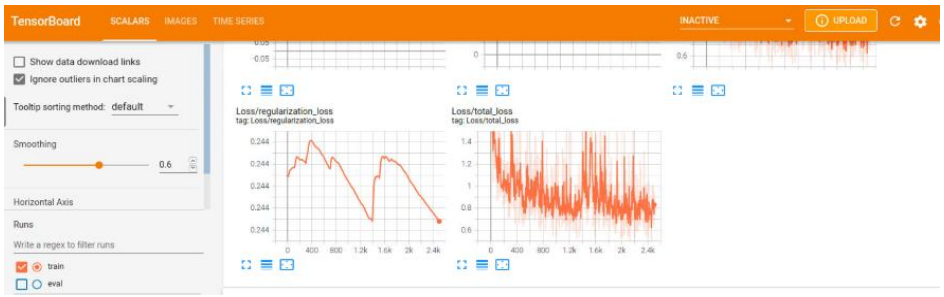
Phase 2 (Training and Evaluation graphs)

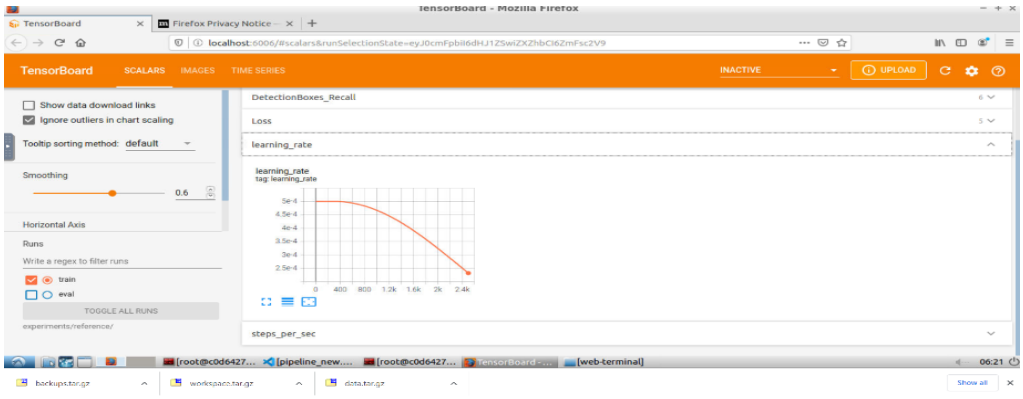
Since the graphs obtained were not very meaningful, training was performed once again based on the updated parameters in the pipeline_new_config file. Following parameters were changed

```
learning_rate_base: 0.0005
total_steps: 4500
warmup_learning_rate: 0.0005
warmup_steps: 300
```

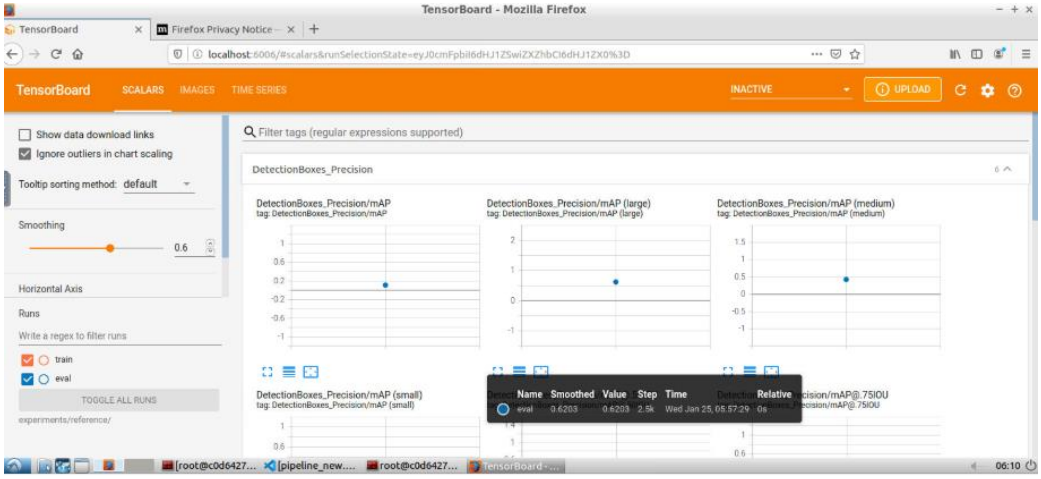
The graphs obtained were as follows for training and evaluation process:

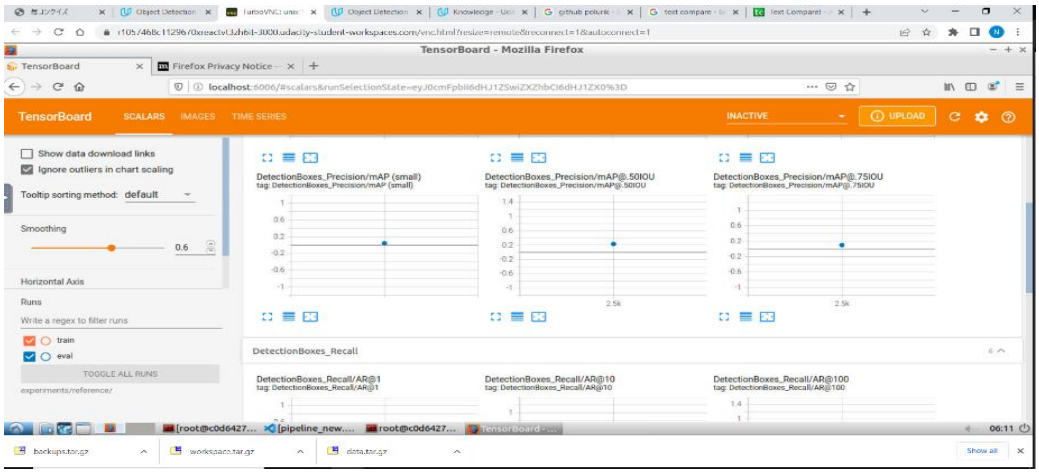


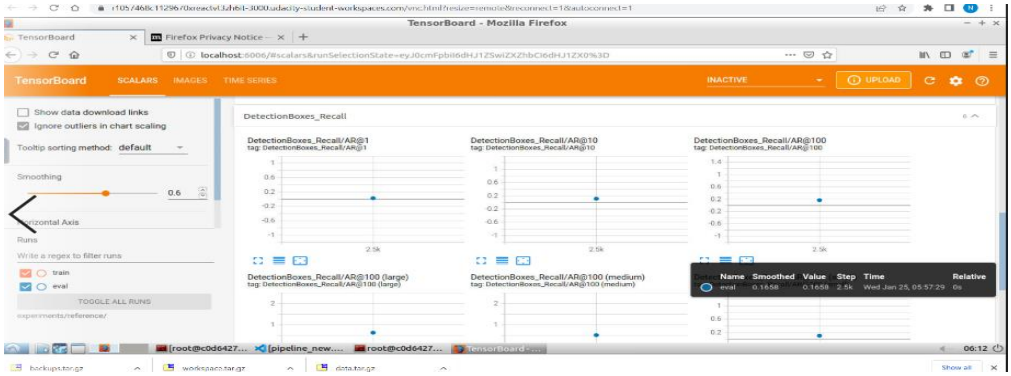


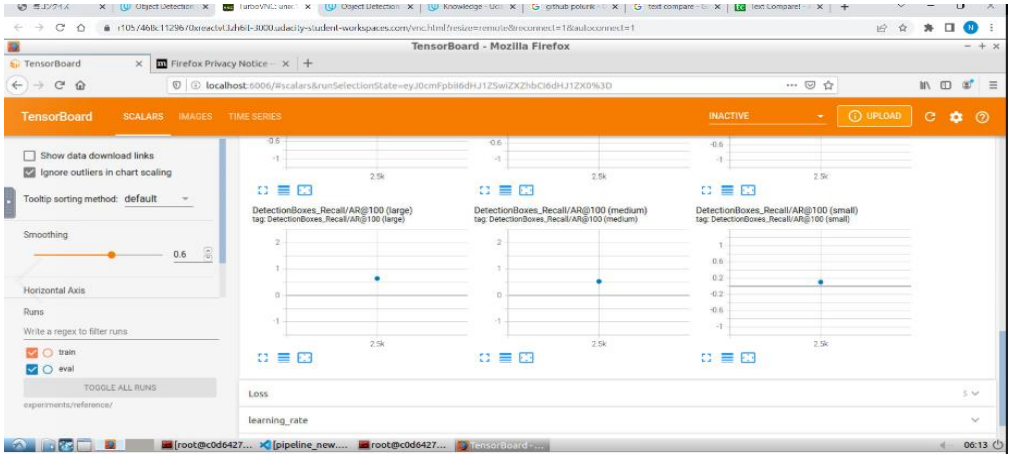


Evaluation graphs




TensorBoard - Mozilla Firefox
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
Runs
Write a regex to filter runs
train
eval
TOGGLE ALL RUNS
DetectionBoxes_Precision/mAP (small)
DetectionBoxes_Precision/mAP@.50IOU
DetectionBoxes_Precision/mAP@.75IOU
DetectionBoxes_Recall
DetectionBoxes_Recall/AR@1
DetectionBoxes_Recall/AR@10
DetectionBoxes_Recall/AR@100


TensorBoard - Mozilla Firefox
TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
DetectionBoxes_Recall
DetectionBoxes_Recall/AR@1
DetectionBoxes_Recall/AR@10
DetectionBoxes_Recall/AR@100
DetectionBoxes_Recall/AR@100 (large)
DetectionBoxes_Recall/AR@100 (medium)
Name Smoothed Value Step Time Relative
eval 0.1658 0.1658 2.5k Wed Jan 25, 05:57:29 0s


TensorBoard - Mozilla Firefox
TensorBoard
SCALARS
IMAGES
TIME SERIES
INACTIVE
UPLOAD
DetectionBoxes_Recall/AR@100 (large)
DetectionBoxes_Recall/AR@100 (medium)
DetectionBoxes_Recall/AR@100 (small)
Loss
learning_rate