Project Writeup: Object Detection in an Urban Environment

Project overview

This section should contain a brief description of the project and what we are trying to achieve. Why is object detection such an important component of self driving car systems?

We are trying to use camera detection to detect the objects in Waymo Open DataSet.

Object detection is important to self driving car, because the car first needs to see the environment around in an accurate and timely fashion, then it can react accordingly.

Set up

This section should contain a brief description of the steps to follow to run the code for this repository.

I used the official Workspace in Udacity course UI, so I did not set up the environment myself.

To run the code, follow the steps:
1, split the dataset
python create_splits.py --data_dir ./data/waymo

2, train the model
python experiments/model_main_tf2.py --model_dir=experiments/reference/
--pipeline_config_path=experiments/reference/pipeline_new.config

3, evaluate the model
CUDA_VISIBLE_DEVICES="" python experiments/model_main_tf2.py
--model_dir=experiments/reference/
--pipeline_config_path=experiments/reference/pipeline_new.config
--checkpoint_dir=experiments/reference/

4, monitor with TensorBoard
python -m tensorboard.main --logdir experiments/reference/

5, export the 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/

6, generate inference video
python inference_video.py --labelmap_path label_map.pbtxt --model_path experiments/reference/exported/saved_model --tf_record_path ./data/waymo/test/segment-10072231702153043603_5725_000_5745_000_with_camera_labels.tfrecord --config_path experiments/reference/pipeline_new.config --output_path animation_5725.gif

Dataset

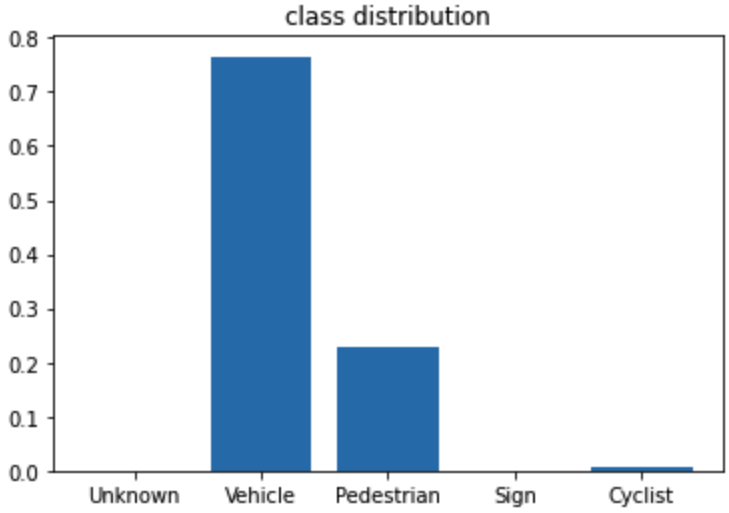
Dataset analysis

This section should contain a quantitative and qualitative description of the dataset. It should include images, charts and other visualizations.

The dataset contains 97 tfrecord files. The class distribution based on some sampled frames is listed below, the majority is vehicle, then pedestrian, then cyclist.

```
total_count 454951
labels ['Unknown', 'Vehicle', 'Pedestrian', 'Sign', 'Cyclist']
counts [     0 348286 103958      0   2707]
ratios [0.         0.76554618 0.22850373 0.         0.00595009]
```

Out[21]: `Text(0.5, 1.0, 'class distribution')`



Cross validation

This section should detail the cross validation strategy and justify your approach.

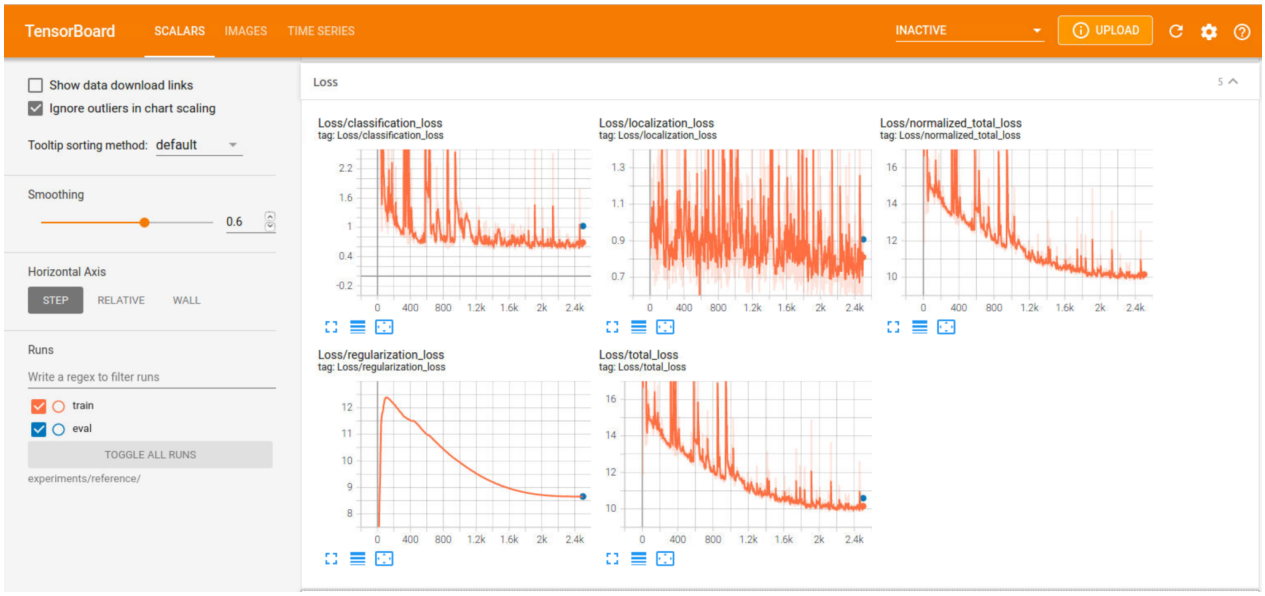
According to the project instruction, we only need to split the tfrecord files into train and validation. I randomly shuffled the file list, took the first 80% files as training, and the rest 20% files as validation.

Training

Reference experiment

This section should detail the results of the reference experiment. It should includes training metrics and a detailed explanation of the algorithm's performances.

I trained for more than 36000 steps. Initially the training loss is more than 16, and it gradually reduced to around 0.965. The validation loss also went down accordingly, at a similar level, but a bit higher.



Improve on the reference

This section should highlight the different strategies you adopted to improve your model. It should contain relevant figures and details of your findings.

At first I tried the default momentum optimizer and cosine decay learning rate (with warm up learing rate as 0.01, and learning rate base as 0.04), and trained fro 2500 steps, the but loss got to 10 and could not go down. So in the inference video, there was no bounding box displayed out.

So I tried multiple changes:
1, use random horizontal flip and brightness adjustment for Data Augmentation.

2, change to Adam optimizer.
3, change to the exponential learning rate, and a small learning rate 0.01. Also, added min learning rate to keep a min value.
4, increase the training steps to more than 36000 steps

Eventually got a training loss around 0.965, and validation loss at similar level. In the inference video, there are objects detected.