Object Detection Writeup

1. **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?

In this project, we are doing object detection. This is important because we want to detect objects on the road from the background. We want to detect cars, bicyclist, pedestrians, and objects on the road so that our algorithm can avoid it.

2. **Set up**: This section should contain a brief description of the steps to follow to run the code for this repository.

I followed all the steps in the instructions. So first I used a pretrained fpn resnet model, and then I used it to train the waymo open dataset. I also changed some parts in the config file to fine-tune the parameters for better performance. Lastly, I did data augmentation to improve performance.

3. **Dataset**

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

    b. *Cross-validation*: This section should detail the cross-validation strategy and justify your approach.

The dataset contains images of streets with road users, completed with the bounding-box labels. The files - ex1.png and ex2.png - shows some examples of the images and bounding-boxes. Here, different classes, such as pedestrians and cars are bounded in different color boxes to differentiate them. I also calculated the distribution of cars, pedestrians, and cyclist in the file graph.png, and found that there are about 16000 cars, 4000 pedestrians, and less than 500 cyclists. Lastly, I also have tensorboard charts which shows the performance of both experiments that I did (more on that in question 4). For cross-validation, the model is trained on a portion of the data (the training set) and then evaluated on the remaining portion (the test set).

4. **Training**

a. *Reference experiment*: This section should detail the results of the reference experiment. It should include training metrics, Tensorboard charts, and a detailed explanation of the algorithm's performance.

b. *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.

First, I ran the reference experiment. The results were very underwhelming, with the loss only converging to about 250 (the goal should be below 1). This amount of loss is horrible. The tensorboard charts can be found in reference.png file. Here, we can see that the loss is very big and not going down much. This is probably because the learning rate decays too quickly, and there is no data augmentation.

In this improvement, I changed the batch size to 4, the number of steps to 4000, the learning rate to 0.4, and added many data augmentations (example is in ex_aug.png). After training this improved model, I got the final error about 1.0 and 1.1, which is around the target! The charts can be found in improved.png file.