**Car Object Detection Using Tensorflow**

**References**

Nicholas Renotte Course and Code: https://www.youtube.com/watch?v=yqkISICHH-U (https://github.com/nicknochnack/TFODCourse)

Jett Heaton Troubleshooting TF, CUDA, cuDNN Setup: https://www.youtube.com/watch?v=OEFKlRSd8Ic

**Setup**

Create Virtual Environment

Link Virtual Environment to Jupyter Notebook (see video 2)

Visual Studio

Cuda, cuDNN (to use GPU to train models, much faster than CPU)

Add to Environment Variables -> Path

A screenshot of a computer program

Description automatically generated

In order to use your GPU to train, you need to downgrade Python and Tensorflow (<https://www.tensorflow.org/install/pip#windows-native>). I used Python 3.9.17 and Tensorflow 2.10.0. You may also need to download Tensorflow-gpu 2.10.0.

**Image Collection**

Use Nicholas’s file 1 or alternatively screenshot or download images separately. Make sure to have different positions/colors/time/etc.

**Image Labeling**

>>>cd Tensorflow/labelimg

>>>python labelImg.py

Create tight boxes around your target

**Training**

Pick a pretrained model from Detection Model Zoo (https://github.com/tensorflow/models/blob/master/research/object\_detection/g3doc/tf2\_detection\_zoo.md). Models vary based on time to train and accuracy.

Make sure to confirm the GPU is showing and will be used for training.

**Evaluation**

Run the model on the Test group

Use Tensorboard to evaluate performance

Input specific images or feed live webcam video to see how the model actually works

**Improvements**

Add more training/testing data

Use a different pre-trained model

Modify classes

**Coding Tips**

Install packages Python -m pip install ipykernel

Install packages conda install ipykernel

Check package version pip freeze | findstr tensorflow

Clear Terminal Output cls

Quit Ctrl + C

Monitor GPU usage nvidia-smi --query-gpu=utilization.gpu --format=csv --loop=1

nvidia-smi

**To Access Terminal**

run miniconda3 as admin

cd C:\Users\Matt Matsuo\Documents\tf\_final

conda activate tf\_final

jupyter notebook

**TEST IF GPU IS CORRECTLY HOOKED UP AND USABLE WITH CUDA**

python

import tensorflow as tf

print(tf.\_\_version\_\_)

len(tf.config.list\_physical\_devices('GPU'))>0

**Car Project Specifics**

Documents/tf\_final

Other Train and Test – used this data when attempted multiple classes. Raw video here.

Tensorflow/workspace/annotations – label\_map shows categories, test and train records are the compiled images and these files will be used for train/test

Tensorflow/workspace/images – images with their corresponding label files

Tensorflow/workspace/models

my\_ssd\_mobnet

my\_ssd\_mobnet\_v2

ssd\_resnet50\_v1\_fpn\_640\_v1

efficientdet\_d1\_car\_only\_v1

Tensorflow/workspace/pre-trained-models – downloaded models from the Zoo

**Results**

The best model was **efficientdet\_d1\_car\_only\_v1**

Example results are in **Final Model Live Webcam Footage.mp4**

This model only attempted to classify cars. I dropped the trailer, truck, bus, and van tags.

A screen shot of a computer program

Description automatically generated

A screenshot of a graph

Description automatically generatedA graph with numbers and a dot

Description automatically generated

A screenshot of a graph

Description automatically generated

A car parked on a street

Description automatically generated

**Future Steps**

Train for trailers, trucks, buses, and vans

Collect data on motorcycles, bicycles, and pedestrians

Classify and train a model on vehicles travelling from right to left and left to right

Try other models in Zoo as well as other methods such as YOLO (you only look once)

Collect data on vehicle speed

Collect data on traffic (volume) by time of day and day of the week