Scott Gibson

October 16th, 2022

Capstone Project Proposal

The problem will be object detection and classification in images taken from a vehicle. Ideally the model will be able to detect things of importance for an autonomous vehicle and label it with a bounding box and descriptive word, like ‘car’. Objects of interest could be pedestrians, bicycles, and other cars and trucks for example. A simple proof of concept model would be able to learn to identify cars from a database of labeled images. It could then take in new images that a user submits to it and correctly find and label cars in that image.

In most of my internship experience I have worked with Computer Vision algorithms and web development. That along with my interest in Deep Learning inspired me to work on a project related to that topic, and to host that end result on the web. The industry of autonomous vehicles is one I’d be interested in working in sometime in my career, and I have a family member currently working at Zoox, who would often discuss some of the interesting problems they would run into.

There is a plentiful amount of labeled data on Roboflow relating to this problem. A large majority of the images provided have labels and bounding boxes describing cars specifically. So if we were to consider labeling pedestrians or stop-lights, the datasets are greatly unbalanced towards images of cars. And, on downloading some of the datasets, image preprocessing is applied, which increases the total quantity of the images via various transformations.

I plan on training a benchmark model on Google Cloud using their Vision API. However, the dataset I have must be transformed and reduced before I can submit it to their algorithms. Ideally, this will create a solid benchmark for me to compare against. To create my own model, I will research and use open-source image classification neural networks, and add layers or update parameters to attempt to fit it to my problem. Ideally, I would find an algorithm already trained on detecting or labeling cars in images, and then tweak it to fit the images that I supply.

I plan on the final deliverable being an application deployed on the web. The user interface would allow a user to submit images of their choice, and display output images with the detected objects boxed and labeled. This API would either be serviced by Google Cloud or AWS.

The processing power by CPU or GPU will likely need to be powerful if I end up using all the images in the datasets I've observed. Training a Deep Learning model on thousands of images is a very computational and memory expensive task. The current dataset I'm working with has roughly ten thousand images and more than 500 MB of storage on my machine. By filtering out all but around one thousand images of just cars, I believe I can make a powerful, but simple model using a tenth the amount of memory. Which will be better for my computer, and for uploading that dataset to the cloud.