# Homework 2

Kuan-Chung Lin 0856735

Project repository: <a href="https://github.com/purpleFar/street-view-house-numbers-detection">https://github.com/purpleFar/street-view-house-numbers-detection</a>

## Introduction

The proposed challenge is a object detection task with street view house numbers. The dataset contains 33,402 images of 10 classes of digitals.

### Environment

• System: Windows 10

● CPU: Intel® Core<sup>TM</sup> i5-10300H CPU @ 2.50GHz 2.50GHz

• GPU: NVIDIA GeForce GTX 1660 Ti

• Language: Python-3.6.12

• Major module: mmdetection

• mmdetection github link: https://github.com/open-mmlab/mmdetection

## **Data Processing**

First, I split the train dataset into 32,387 for training and 1015 for testing. And I transfer the original .mat file to the coco format.

I thought this task is different with other object detection task. Because the digitals are not symmetrical. I didn't use random flipping in data processing. In fact, I didn't used most image augmentation methods. Because the object detection task is not as simple as object classification task, the model is hard to fit the training data.

### Model

Because I use mmdetection to make my own model and it was easy to deployed. I tried a lot of model to get my best score, for instance, YOLOv3 and Faster R-CNN with different backbone. At the end, I used Faster R-CNN with backbone ResNet-34.

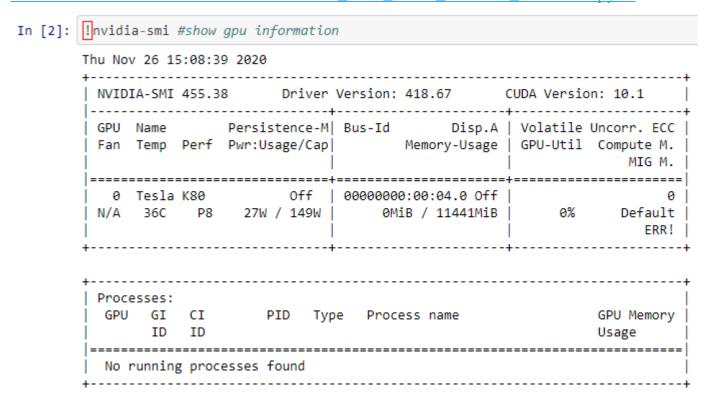
## Summary of Results

I got mAP 58.08% use my Faster R-CNN with backbone ResNet-34.



mAP\_0.58082\_0856735.json

Besides, I run my model on Colab to test the speed. Note that I didn't transfer the weight in to that model in Colab. Because I think whether I use my pre-trained weight or not the testing speed will be the same. Here is the link you can see my result: <a href="https://github.com/purpleFar/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection.ipynb">https://github.com/purpleFar/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection/blob/master/street-view-house-numbers-detection-numbers-detecti



Run 50 loops, use 4.449 second in model inference. (Not included images loading times) fps:11.24, 88ms per loop