Report

0756104 李家揚

* Github link: <https://github.com/LeeJohnn/Visual-Recognition-HW2>
* Reference: <https://github.com/penny4860/tf2-eager-yolo3?fbclid=IwAR31sk6V2uAKaspia4M7GhMrV9FhNQQPVjeOO553_ACg4vGVWLXnD3chPes>, <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1640804&casa_token=zrsGRXyN6roAAAAA:AJnSYQLPdN9_WvtNBWnET1AtVnecgpPnFQctCUTvOOYSCAc8abPXsD8vsHGAp2EtfAuPTm2JMAc&tag=1>
* Speed benchmark
* Introduction

Using yolo-v3 to implement digit detector on svhn dataset.

* Methodology

At first, using the code provided by TA to parse the annotation in .h5 file. There are two parts for digit detector. The first one is classifier for the object detection, I saw the reference and trace the code to implement a CNN architecture for detecting character. In this network, it would classify text and non-text. The architecture is as follows:

* + INPUT: [32x32x1]
  + CONV3-32: [32x32x32]
  + CONV3-32: [32x32x32]
  + POOL2: [16x16x32]
  + CONV3-64: [16x16x64]
  + CONV3-64: [16x16x64]
  + POOL2: [8x8x64]
  + FC: [1x1x1024]
  + FC: [1x1x2]
  + Training Accuracy : 97.91%
  + Test Accuracy : 96.98%

Then the second part is the classifier for digit recognition. I also using the reference then implement a CNN architecture for recognizing numbers. The architecture is same except for the number of class. The detector will run in 2 step. First, the detector finds candidate region proposed by the MSER algorithm. And then the classifier determines whether or not it is a number in the proposed region. For the result, it will show the bounding box of the object and label the number and the score of the prediction.

* Summary

In this assignment, I have learned the yolo-v3 tool and the Google Colab application. Although it is hard to trace the code, it is very interesting and benefited a lot. For training yolo, I have to figure out the data size and the coordinate of data annotation should be rearrange from 0 to 1. I also found that training yolo in Google Colab will take lots of time on synchronizing the data, and may stop at some iteration.