2019/11/28
0780828
Alisher Mukashev (穆安里)

HOMEWORK #3

Goals

Train the digit detection model based on the StreeView House Number (SVHN) dataset.

Introduction

In this homework, I have trained a digit detector using RetinaNet [1]. Given SVHN dataset contains 33402 training images and 13068 test images. Example of a train image is shown on Figure 1:

Figure 1: Train image

Deep learning model must be trained and be able to recognize all classes and must be not only accurate, but also fast. The backbone for RetinaNet is Resnet50.

The code based on Github repository [2].

Methodology

Preprocessing

To train the model we should use image and annotation data. The annotation data represented in PASCAL VOC Annotation Format [3].

Model Architecture

In this homework, I am going to use Retina-Net with Resnet50 backbone.

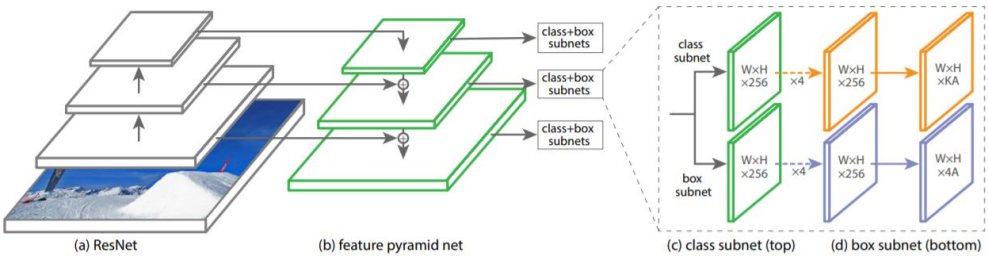

Figure 2: RetinaNet network architecture

Hyperparameters

The learning rate 0.0001 was selected. The batchsizes 4, 8, 16 and 32 were tested and finally 32 was chosen. The batchsize 64 caused memory problems. The number of epochs in the range (5:10) were checked and the best results were shown when choosing 4.

Results

The program was written in Google Colab and was inferenced by TA of this class. The most common metric for object detection is mean Average Precision (mAP). It was used to evaluate the performance of obtained model. To evaluate the speed Google Colab GPU was used.

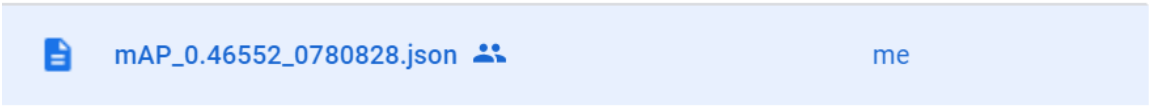

Figure 3: Accuracy evaluation; map 0.46552

```
[ ] %%timeit
    _, _, _ = model.predict_on_batch(np.expand_dims(image, axis=0))

    The slowest run took 253.78 times longer than the fastest. This could mean that an intermediate result is being cached.
    1 loop, best of 3: 25.5 ms per loop
```

Figure 4: Speed evaluation; 25.5 ms per loop

Summary

Before I was working only with YOLO network. In this homework I used RetinaNet and it also showed good results. The Github link to my repository [4].

There was always a problem with memory allocation and that is why the maximum batch size that I used is 32. I think If I used batch size 64 and more I would get better results. Hope In the future I will find out other approaches to avoid a memory problem.

References

1. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. ´ Focal loss for dense object detection. arXiv preprint arXiv:1708.02002, 2017.
2. https://github.com/penny4860/retinanet-digit-detector
3. https://github.com/penny4860/svhn-voc-annotation-format
4. https://github.com/alishsuper/Selected-Topics-in-Visual-Recognition-using-Deep-Learning/tree/master/HW3