Selected Topics in Visual Recognition using Deep Learning

Homework 3

Kuan-Chung Lin 0856735

Project repository: https://github.com/purpleFar/instance-segmentation

Introduction

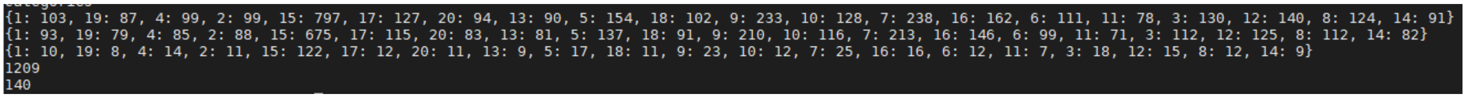
The proposed challenge is instance segmentation task with Tiny PASCAL VOC dataset. The dataset contains 1,449 images of 20 classes. The data is split into 1,349 training images and 100 testing images.

Environment

- System: Linux
- CPU: Intel(R) Xeon(R) Platinum 8260M CPU @ 2.40GHz
- GPU: NVIDIA Tesla T4
- Python: 3.6.12
- Extra modules: mmdetection

Data Processing

First, I split the train dataset into 1,209 for training and 140 for validation. The image bellow shows number of each classes. Each row represents the distribution of original data, new training data, validation data, number of new training data and number of validation data.



```
{1: 103, 19: 87, 4: 99, 2: 99, 15: 797, 17: 127, 20: 94, 13: 90, 5: 154, 18: 102, 9: 233, 10: 128, 7: 238, 16: 162, 6: 111, 11: 78, 3: 130, 12: 140, 8: 124, 14: 91}
{1: 93, 19: 79, 4: 85, 2: 88, 15: 675, 17: 115, 20: 83, 13: 81, 5: 137, 18: 91, 9: 210, 10: 116, 7: 213, 16: 146, 6: 99, 11: 71, 3: 112, 12: 125, 8: 112, 14: 82}
{1: 10, 19: 8, 4: 14, 2: 11, 15: 122, 17: 12, 20: 11, 13: 9, 5: 17, 18: 11, 9: 23, 10: 12, 7: 25, 16: 16, 6: 12, 11: 7, 3: 18, 12: 15, 8: 12, 14: 9}
1209
140
```

You can not that my split method is care about class distribution of original data. I try to make class distribution of new training set and validation set are same as original data.

Model

I use the best performing model in mmdetection, which is “cascade mask rcnn”. And the backbone what I used is also the best performing in mmdetection, which is XResnet-101. My hyperparameters you can see in https://github.com/purpleFar/instance-segmentation/blob/master/configs/cascade_rcnn/cascade_mask_rcnn_x101_32x4d_fpn_hw3.py. It do not have any special settings.

Summary of Results

I got mAP 54.26%

mAP_0.54268_0856735_3.json