# deepLab训练

## 一 deeplab训练环境搭建

* 1. 参考

|  |
| --- |
| <https://lijiancheng0614.github.io/2018/03/13/2018_03_13_TensorFlow-DeepLab/> |

* 1. 数据集

准备[VOCtrainval\_11-May-2012.tar](http://geeekvr.com:18129/edit/work/deeplab/datasets/pascal_voc_seg/VOCtrainval_11-May-2012.tar)

* 1. 下载code

|  |
| --- |
| git clone <https://github.com/sycophant-stone/tf_base.git> |

精简模式

|  |
| --- |
| git clone --depth=1 https://github.com/sycophant-stone/tf\_base.git |

* 1. 把VOC放到research/deeplab/datasets中
  2. 运行[download\_and\_convert\_voc2012.sh](http://geeekvr.com:18129/edit/work/deeplab/datasets/download_and_convert_voc2012.sh)
  3. 使用slim的env环境设置

首先先解压缩(注意这里-o的用法)

|  |
| --- |
| root@df4112054141:/work/gi/tf\_base/research# **unzip -o protobuf.zip** |

然后运行工具脚本

|  |
| --- |
| #!/bin/bash  echo "export"  **export PYTHONPATH=$PYTHONPATH:`pwd`:`pwd`/slim**  echo "protoc"  /home/julyedu\_433249/work/tf\_base/research/bin/protoc object\_detection/protos/\*.proto --python\_out=. |

* 1. 下载pretrain模型

|  |
| --- |
| # From deeplab/datasets/pascal\_voc\_seg/  mkdir init\_models  cd init\_models  wget http://download.tensorflow.org/models/deeplabv3\_pascal\_train\_aug\_2018\_01\_04.tar.gz  tar zxf ssd\_mobilenet\_v1\_coco\_11\_06\_2017.tar.gz |

下载dataset

|  |
| --- |
| http://host.robots.ox.ac.uk/pascal/VOC/voc2012/VOCtrainval\_11-May-2012.tar |

* 1. train用的命令

|  |
| --- |
| python deeplab/train.py --logtostderr --training\_number\_of\_steps=30000 --train\_split="train" --model\_variant="xception\_65" --atrous\_rates=6 --atrous\_rates=12 --atrous\_rates=18 --output\_stride=16 --decoder\_output\_stride=4 --train\_crop\_size=513 --train\_crop\_size=513 --train\_batch\_size=1 --dataset="pascal\_voc\_seg" --tf\_initial\_checkpoint=deeplab/init\_models/deeplabv3\_pascal\_train\_aug/model.ckpt --train\_logdir=deeplab/trainlog --dataset\_dir=deeplab/datasets/pascal\_voc\_seg/tfrecord |
|  |

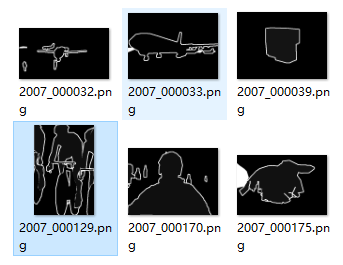
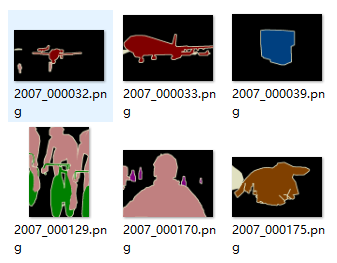
* 1. win10上搭建
     1. 搭建数据集

|  |
| --- |
| Gitbash中运行  ./download\_and\_convert\_voc2012.sh  利用其解压缩的方法. |

* + 1. 整理数据集

利用class来生成raw class.

|  |
| --- |
| cd D:\work\stuff\modules\misc\sprd\_camera\alg\july\tf\_base\research\deeplab\datasets  **python remove\_gt\_colormap.py** --original\_gt\_folder="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\VOCdevkit\\VOC2012\\SegmentationClass" --output\_dir="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\VOCdevkit\\VOC2012\\SegmentationClassRaw" |



利用tf生成tfrecord

|  |
| --- |
| (tensorflow) D:\work\stuff\modules\misc\sprd\_camera\alg\july\tf\_base\research\deeplab\datasets>  **python** **build\_voc2012\_data.py** --image\_folder="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\VOCdevkit\\VOC2012\\JPEGImages" --semantic\_segmentation\_folder="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\VOCdevkit\\VOC2012\\SegmentationClassRaw" --list\_folder="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\VOCdevkit\\VOC2012\\ImageSets\\Segmentation" --image\_format=jpg --output\_dir="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\tfrecord" |

* + 1. 可用的环境总结

|  |
| --- |
|  |

|  |
| --- |
| tf\_initial\_checkpoint = "D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\init\_models\\deeplabv3\_pascal\_train\_aug\\model.ckpt" |

pyCharm使用调试golden的命令:

|  |
| --- |
| python deeplab/train.py --logtostderr --training\_number\_of\_steps=30000 --train\_split="train" --model\_variant="xception\_65" --atrous\_rates=6 --atrous\_rates=12 --atrous\_rates=18 --output\_stride=16 --decoder\_output\_stride=4 --train\_crop\_size=513 --train\_crop\_size=513 --train\_batch\_size=1 --dataset="pascal\_voc\_seg" --tf\_initial\_checkpoint="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\init\_models\\deeplabv3\_pascal\_train\_aug\\model.ckpt" --train\_logdir="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\output" --dataset\_dir="D:\\work\\stuff\\modules\\misc\\sprd\_camera\\alg\\july\\tf\_base\\research\\deeplab\\datasets\\pascal\_voc\_seg\\tfrecord" |

二 deeplab的evaluation和test

2.1 eval方法

|  |
| --- |
| # From tensorflow/models/research/  python deeplab/eval.py \  --logtostderr \  --eval\_split="val" \  --model\_variant="xception\_65" \  --atrous\_rates=6 \  --atrous\_rates=12 \  --atrous\_rates=18 \  --output\_stride=16 \  --decoder\_output\_stride=4 \  --eval\_crop\_size=513 \  --eval\_crop\_size=513 \  --dataset="pascal\_voc\_seg" \  --checkpoint\_dir=${PATH\_TO\_CHECKPOINT} \  --eval\_logdir=${PATH\_TO\_EVAL\_DIR} \  --dataset\_dir=${PATH\_TO\_DATASET} |

2.2 可视化方法

|  |
| --- |
| # From tensorflow/models/research/  python deeplab/vis.py \  --logtostderr \  --vis\_split="val" \  --model\_variant="xception\_65" \  --atrous\_rates=6 \  --atrous\_rates=12 \  --atrous\_rates=18 \  --output\_stride=16 \  --decoder\_output\_stride=4 \  --vis\_crop\_size=513 \  --vis\_crop\_size=513 \  --dataset="pascal\_voc\_seg" \  --checkpoint\_dir=${PATH\_TO\_CHECKPOINT} \  --vis\_logdir=${PATH\_TO\_VIS\_DIR} \  --dataset\_dir=${PATH\_TO\_DATASET} |

## 二 可用的gpu环境

2.1 极客云

|  |
| --- |
| ssh -p 8065 [root@geeekvr.com](mailto:root@geeekvr.com)  ssh -L 16006:127.0.0.1:6006 -p 8065 [root@geeekvr.com](mailto:root@geeekvr.com) |

## 三 tensorboard使用方法

3.1 tensorboard映射

|  |
| --- |
| ssh -L 16006:127.0.0.1:6006 julyedu\_433249@101.132.121.198  tensorboard --logdir="/path/to/log-directory"  最后，在本地访问地址：http://127.0.0.1:16006/ |

## 四 terminology

Tf.TFRecordReader

slim.dataset\_data\_provider

tf.train.MomentumOptimizer

tf.image.resize\_images

## 五 答疑

### 5.1 VOC图片原尺寸,以及crop后的尺寸

原尺寸:

Crop后尺寸可调,目前是513,513

### 六 实验

6.1 数据读取实验

描述: 从VOC格式数据,整理成tfrecord模式,并读入到tf的queue中.

6.2 数据增强实验

描述: 采用多种方式对train数据做数据增强.

6.3 xception网络搭建实验.

6.4 aspp网络搭建

6.5 decoder网络搭建

6.6 multi-scale网络实现

6.7 损失函数及评价函数实验

2018-10-12: 损失函数思路摸排好了. 评价函数还没找到.

6.8 summary实验

### 附录1 win10上搭建tenserflow-gpu版本

F1.1 安装anaconda.

F1.2 安装tensorflow-gpu

python install tensorflow-gpu

F1.3 安装对应的CUDA比如9.0

F1.4 然后尝试import tensorflow

会有提示,缺少一个` Could not find cudnn64\_7.dll`,下载对应的cudnn

F1.4 下载cudnn

<https://developer.nvidia.com/rdp/cudnn-download>

解压之后,把对应的cudnn64\_7.dll放到CUDA的bin下,应为CUDA的bin你也放到path路径上了.