



國立雲林科技大學
電子工程系
Department of Electronic Engineering

教育部補助AI應用領域系列課程-
人工智慧計算晶片設計和應用人才培育

LAB7-Deeplab

國立雲林科技大學

夏世昌 特聘教授 / 電子工程系

王斯弘 助理教授 / 前瞻學位學士學程

2022, Fall Semester



YunTech 國立雲林科技大學
National Yunlin University of Science & Technology



dataset

從雲端下載 .h5檔、dataset和deeplabv3-plus-tf2-main.zip

把.h5檔與dataset放到deeplabv3-plus-tf2-main的指定資料夾

名稱	修改日期	類型	名稱	類型	壓縮大小
backbone_weights_of_mobilenetv2.h5	2022/12/6 上午 06:16	H5 檔案	datasets	檔案資料夾	
backbone_weights_of_xception.h5	2022/12/6 上午 06:18	H5 檔案	img	檔案資料夾	
deeplabv3_mobilenetv2.h5	2022/12/6 上午 06:15	H5 檔案	logs	檔案資料夾	
deeplabv3_xception.h5	2022/12/6 上午 06:18	H5 檔案	model_data	檔案資料夾	
VOCdevkit.zip	2022/12/6 上午 06:18	ZIP 檔案	nets	檔案資料夾	
			utils	檔案資料夾	
			VOCdevkit	檔案資料夾	
			.gitignore	Git Ignore 來源檔案	
			deeplab.py	PY 檔案	
			get_miou.py	PY 檔案	
			json_to_dataset.py	PY 檔案	
			LICENSE	檔案	
			predict.py	PY 檔案	



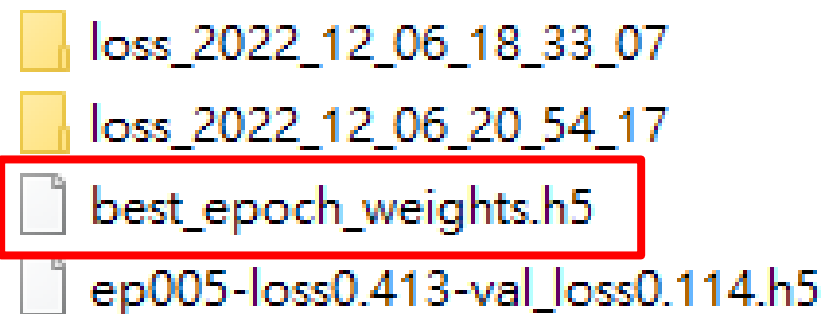
訓練

>python voc_annotation.py

文件生成對應的txt

>python train.py

訓練完會生成 .h5，在logs資料夾底下



修改Deeplab.py的路徑

修改訓練完權重檔路徑

```
class DeeplabV3(object):  
    _defaults = {  
        #-----  
        # model_path 指向logs文件夹下的权值文件  
        # 训练好后logs文件夹下存在多个权值文件，选择验证集损失较低的即可。  
        # 验证集损失较低不代表miou较高，仅代表该权值在验证集上泛化性能较好。  
        #-----  
        "model_path" : 'logs/best_epoch_weights.h5',  
        #-----  
        # 所需要区分的类的个数+1  
        #-----  
        "num_classes" : 21,  
    }
```

儲存權重檔路徑

```
self.model.save('./export/deeplab.h5')  
self.model.summary()
```



預測

>python predict.py

在紅框地方輸入img/street.jpg

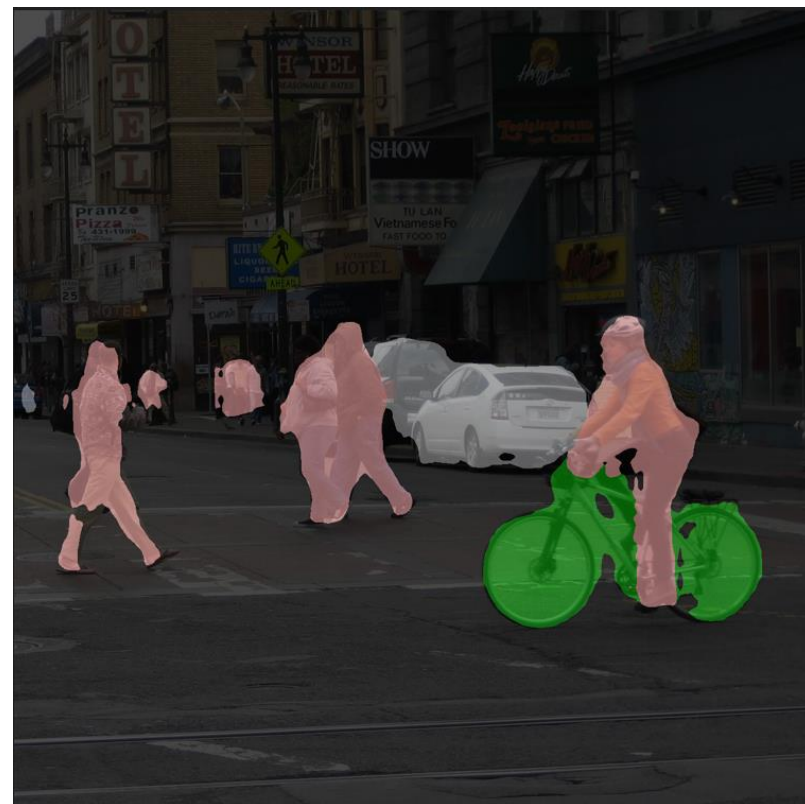
```
Total params: 2,758,597
Trainable params: 2,719,813
Non-trainable params: 38,784
```

Configurations:

keys	values
model_path	logs/best_epoch_weights.h5
num_classes	21
backbone	mobilenet
input_shape	[512, 512]
downsample_factor	16
mix_type	0

Input image filename:

預測結果



將.h5檔轉成tflite

雲端提供deeplab_tflite.ipynb，記得修改圖片路徑與.h5檔的路徑

```
IMAGE_SIZE = 512
def representative_data_gen():
    dataset_list=glob.glob('VOCdevkit/VOC2007/JPEGImages/*')
    dataset_list_index=np.random.choice(range(len(dataset_list)),100)
    for i in range(100):
        image = dataset_list[dataset_list_index[i]]
        print(image)
        image = tf.io.read_file(image)
        image = tf.io.decode_jpeg(image,channels=3)
        image = tf.image.resize(image,[IMAGE_SIZE, IMAGE_SIZE])
        image = tf.cast(image,tf.float32)
        image = tf.expand_dims(image,0)
        #with tf.Session() as sess:
        #    image = sess.run(image)
        yield [image]
```

```
if tf.__version__[0]=='2':
    converter = tf.lite.TFLiteConverter.from_keras_model(save_keras_model)
elif tf.__version__[0]=='1':
    converter = tf.lite.TFLiteConverter.from_keras_model_file('./export/deeplab.h5')
converter.optimizations = [tf.lite.Optimize.DEFAULT]
converter.target_spec.supported_ops = [tf.lite.OpsSet.TFLITE_BUILTINS_INT8]
converter.inference_input_type = tf.uint8
converter.inference_output_type = tf.uint8
converter.representative_dataset = representative_data_gen
tflite_model = converter.convert()
```





在ubuntu轉edgetpu.tflite檔

雲端有提供訓練完的tflite檔

```
(base) hongyu@ubuntu:~/Downloads$ edgetpu_compiler deeplab.tflite
Edge TPU Compiler version 16.0.384591198
Started a compilation timeout timer of 180 seconds.

Model compiled successfully in 2107 ms.

Input model: deeplab.tflite
Input size: 743.07KiB
Output model: deeplab_edgetpu.tflite
Output size: 1.06MiB
On-chip memory used for caching model parameters: 805.00KiB
On-chip memory remaining for caching model parameters: 6.17MiB
Off-chip memory used for streaming uncached model parameters: 0.00B
Number of Edge TPU subgraphs: 1
Total number of operations: 72
Operation log: deeplab_edgetpu.log
```



在樹梅派上執行

雲端有提供semantic_segmentation.py

將以下兩個檔案放置版端

1.自己的edgetpu.tflite

2.semantic_segmentation.py

接上鏡頭

樹梅派執行

>python3 semantic_segmentation.py --model 自己的edgetpu.tflite

B-ground	Aero plane	Bicycle	Bird	Boat	Bottle	Bus
Car	Cat	Chair	Cow	Dining-Table	Dog	Horse
Motorbike	Person	Potted-Plant	Sheep	Sofa	Train	TV/Monitor





END

