



機械手臂的應用

解決營造業人力不足問題

組員：古若喬 / 陳品修 / 謝慈芯

2021/11/17

目錄

CONTENTS

1

背景

營建業缺工問題 / 連帶影響

2

目的

主要問題 / 解決方法

3

應用情境

情境一：語音 / 情境二：手勢

4

未來發展

應用想像

5

追加訓練

Google Colab / Kaggle

6

成果結語

影片成果 / 未來展望

營建業缺工問題 / 連帶影響

背景 01

營建業缺工問題

- 根據營建署發布，營造業勞工缺工總人數已達4萬8,809人
- 又以**鋼筋工**、**模板工**、**水泥工**、**電焊工**等基層勞工的缺口最大。
- 市場需求暴增、**移工進不來**、**人工成本上升**都是隱憂。
- 台灣工地出現**人力斷層**。一些像是**搬磚塊**、**綁鋼筋**、**釘模板**的**粗重工作**根本找不到新人投入。
- 那是辛苦又危險的工作，現在都只剩6、70歲的老師傅在做。



連帶影響

營造業者無法找到足夠的工人

營造業面臨合約逾期罰款壓力

導致建案無法推出/公共工程流標

基層營造工人的流動率高

工程進度落後

營造廠因人力不足不接案

主要問題 / 解決方法

目的 02

目的

- 主要問題：

1. 解決台灣營造業勞工嚴重不足
2. 解決許多粗工（例如：釘模板、切割模板、綁鋼筋等等）勞工不願意做的問題

- 解決方式：

1. 使用機械手臂來代替人力，透過對機械手臂的**控制與設計**其運作**腳本**取代大量人力工作。
2. 且利用機械手臂可以**抬起較重物件**的優勢來解決粗工無人力的情況。

情境一：語音 / 情境二：手勢

應用情境 03

應用情境

- 情境一、材料批發**工廠**環境：

- 機械手臂**搬貨系統** --- **語音辨識**

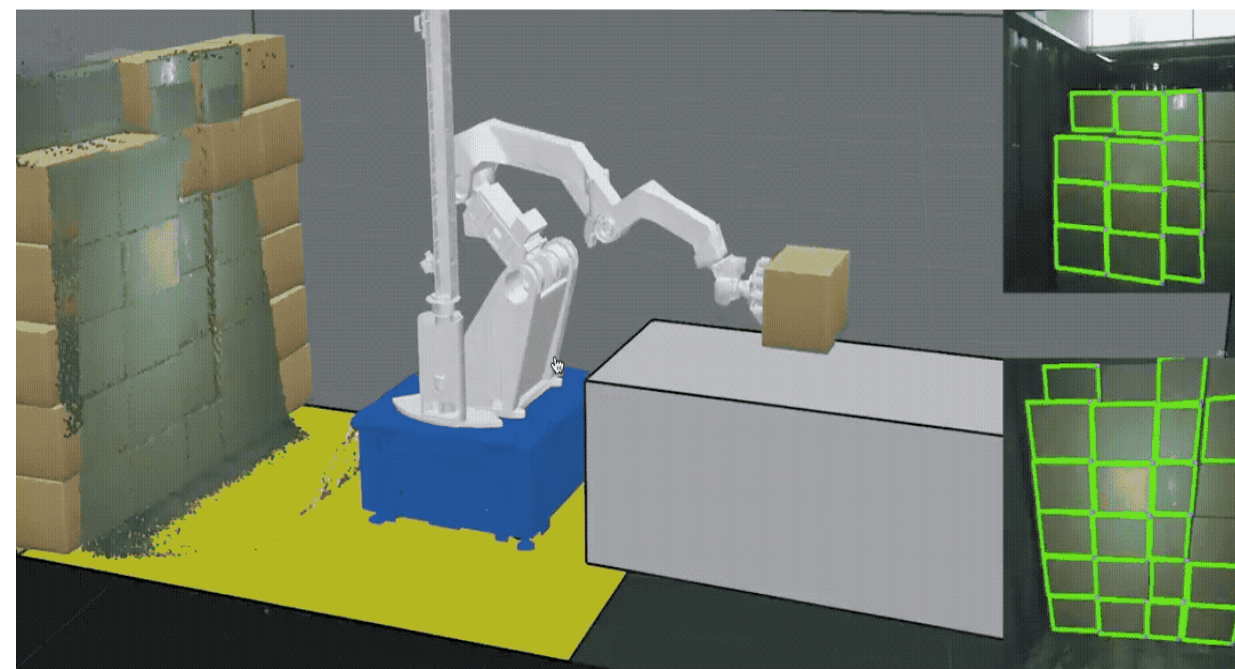
由於**工廠**搬運過程中手需要做其他作業，手部行動較不便，因此這種情況可以用**語音指示**機械手臂搬運建材，做兩點間 的來回搬運。

- 情境二、**工地現場**環境：

- 機械手臂**建構系統** --- **手勢辨識**

相對吵雜的**工地現場**環境由於無法良好的收音，因此這種情況可以用**手勢辨識**執行寫好的腳本，例如堆疊磚塊的路徑。

情境一、工廠搬貨情境示意圖



情境一、工廠搬貨情境

- 機械手臂之設計：
 - 目的：於**工廠**中需要的是將材料從**A地**搬移至**B地**。
因此設計利用語音輸入運動控制命令，且著重在“點對點”的機械運動控制函式。
 - 語音命令：如右圖
 - 情況模擬：假設現在有一個運作情境是需將材料從**A地** ($x=10,y=10,z=0$) 搬移至**B地** ($x=20,y=20,z=0$)。
 - 操作流程：1. 於使用者介面手動輸入材料點與目的點位置，並點選servo on等初始化指令。
2. 等機械設備準備就緒後用麥克風依據以下順序輸入：

“移動至材料點” → “下降” → “暫停” → “鬆開” → “夾取” → “移動至目的點” → “下降” → “暫停” → “鬆開”

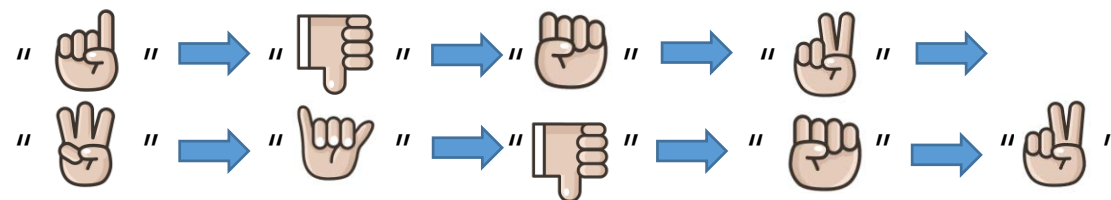
語音命令	命令目的
移動至材料點	機械手臂移動至搬移材料的起始位置
移動至目的點	機械手臂移動至堆放材料的目的地位置
開始操作	啟動機械手臂(servo on)
結束操作	關閉機械手臂(servo off)
下降	將機械手臂下降至材料位置
夾取	闔上機械手臂夾子
鬆開	打開機械手臂夾子
確認	確定執行語音指令
暫停	暫停當前指令動作
取消	取消當前語音指令並重新辨識

情境二 、工地現場建構情境示意圖



情境一、工地現場建構情境

- 機械手臂之設計：
 - 目的：於**工地**中由於環境較為吵雜，為避免收音之效果會不佳，本企劃之設計是利用手勢輸入運動控制命令，且著重在“點對點”的機械運動控制函式。
 - 手勢命令：如右圖
 - 情況模擬：假設現在有一個運作情境是需將材料從**A地** ($x=10,y=10,z=0$) 搬移至**B地** ($x=20,y=20,z=0$)。
 - 操作流程：1. 於使用者介面手動輸入材料點與目的點位置，並點選servo on等初始化指令。
2. 等機械設備準備就緒後於攝影機前依據以下順序輸入：



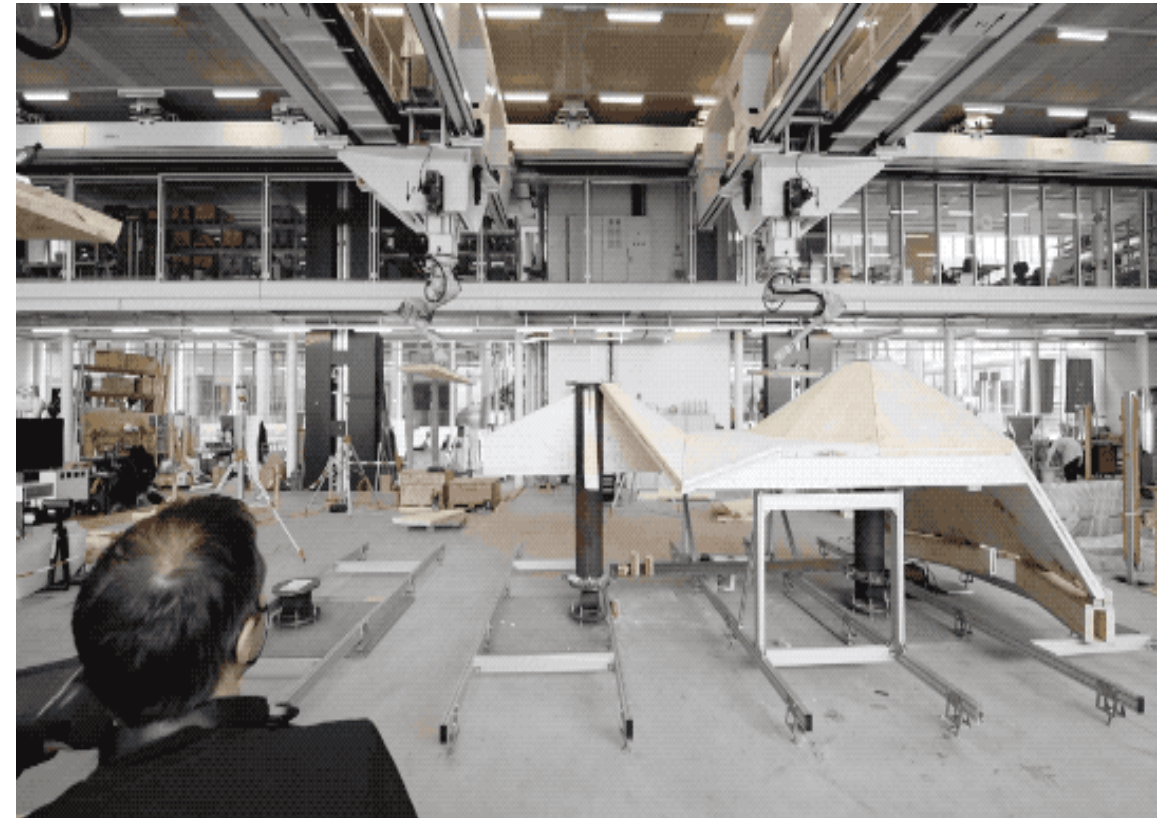
手勢命令	命令目的
移動至材料點 	機械手臂移動至搬移材料的起始位置
移動至目的點 	機械手臂移動至堆放材料的目的地位置
開始操作 	啟動機械手臂(servo on)
結束操作 	關閉機械手臂(servo off)
下降 	將機械手臂下降至材料位置
夾取 	闔上機械手臂夾子
鬆開 	打開機械手臂夾子
確認 	確定執行語音指令
暫停 	暫停當前指令動作
取消 	取消當前語音指令並重新辨識

應用想像

未來發展 04

未來發展想像

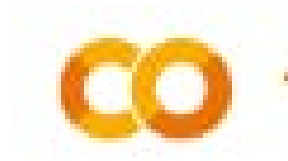
1. 營建數位化 --- 走出傳統建築思維
 - 讓營建可以更數位化發展，減少人力本支出。
2. 營建精緻化 --- 蓋房更精準有智慧
 - 透過機械手臂使營建更精確，也可以提升建築品質與穩固性。
3. 營建智慧化 --- 視覺/聽覺感測
 - 持續發展語音與手勢辨識，讓營建系統可以更加智慧化，具有視覺與聽覺的感測。



Google Colab / Kaggle

追加訓練 05

Google Colab



優點:

- 透過瀏覽器執行。Colab提供可執行python檔(.py)以及jupyter notebook(.ipynb檔)之環境，讓使用者可省去安裝大量函式庫之步驟，並體驗日前最為熱門的深度學習與人工智慧等多種Python函式庫。
- Colab本身也提供多種圖像dataset，以及多種機器訓練的範例程式碼以供測試與參考。
- 可與Google Drive連結，直接存取/修改Google Drive裡的檔案。
- 提供免費GPU使用

缺點:

- 執行時需保持瀏覽器窗口開啟，且須定期確認是否還留在線上
- 免費GPU效能並不算相當好
- GPU有「限量」，其上限不明。當Google認為你使用過量GPU時，會暫時限制該帳號無法使用GPU，且解除時間同樣不明(根據搜尋結果，少則半天，長則一個星期皆有可能)

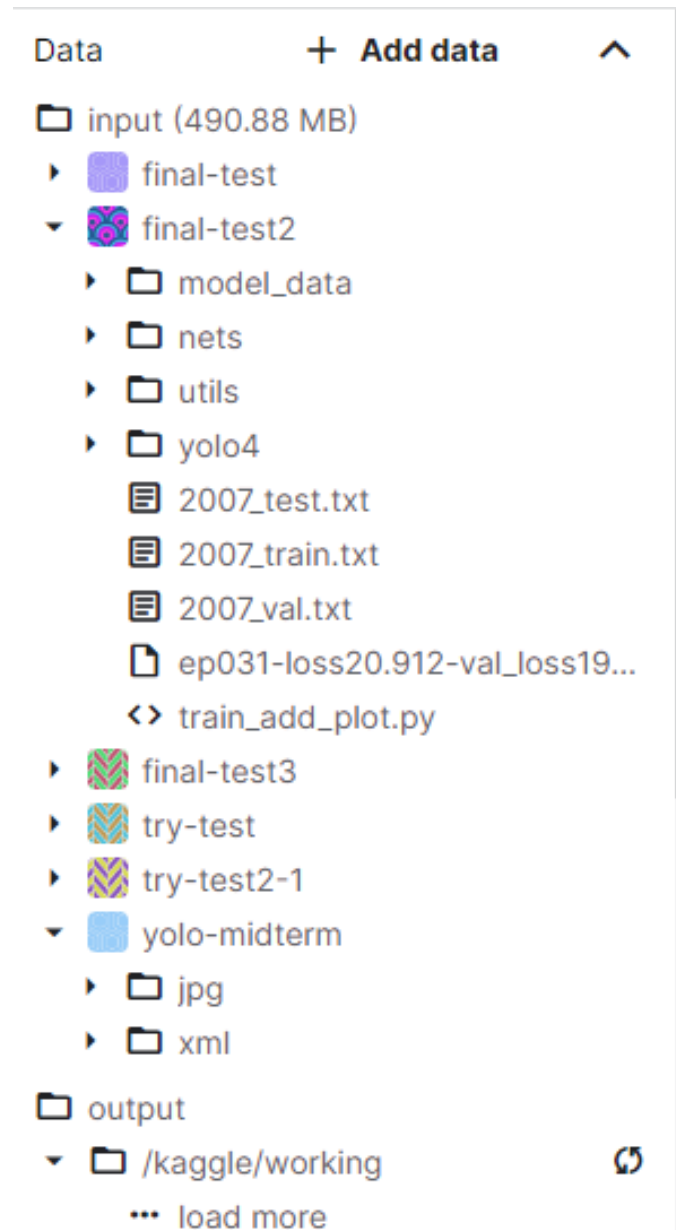
Kaggle kaggle

優點:

- 與Colab相同，提供可執行python檔案之環境，讓使用者可不必安裝大量資料庫便能執行訓練。
- Kaggle原為競技用網站，其所提供之dataset，數量與種類皆遠大於Colab，擴充資料庫時相當方便。
- 同樣提供免費GPU，一個帳號限額為39小時(開啟GPU後開始計算)，每個禮拜補充30小時。
- 可不必一直開著視窗，確認程式在執行後便能關閉網站，之後在上去確認結果就好。

缺點:

- 不像Colab可與Drive連動，Kaggle僅能透過上傳之方式更新資料集，若有檔案需要修改，則只能於本地端修改完之後再次上傳。



Colab訓練yolo模型-1

- 登錄Google Drive，取得image，xml與py檔
- 準備對應環境

The screenshot displays the Google Colab web interface. On the left, the 'Files' pane shows a directory structure with folders like 'simpsons', 'yolo', and '1110智能加工上課'. Under 'yolo', there is a folder named '1110智能加工上課' containing subfolders 'VOCdevkit', 'font', 'logs', 'model_data', 'nets', 'utils', and 'yolo4'. The 'yolo4' folder contains files like '0003.jpg', '2007_test.txt', '2007_train.txt', '2007_val.txt', 'Gesture_video_a...', 'Gesture_yolo_an...', 'Gesture程式註...', 'LICENSE', 'Object_video_an...', and 'Object_yolo_and...'. The main code editor area shows two code cells. The first cell contains the code to mount Google Drive:

```
from google.colab import drive
drive.mount('/content/gdrive')
```

 Below the code, it says 'Mounted at /content/gdrive'. The second cell contains the code to install and configure the environment:

```
!pip uninstall tensorflow -y
!pip uninstall tensorflow-gpu -y
!pip install tensorflow==1.13.2
!pip install tensorflow-gpu==1.13.2
!pip install keras==2.1.5

!pip uninstall h5py -y
!pip install h5py==2.10.0
```

 The output of the second cell shows the installation process, including warnings about ignoring invalid distributions and the successful installation of tensorflow-estimator 2.7.0. The bottom status bar indicates '34.87 GB 可用'.

Colab訓練yolo模型-2

- 修改train_add_plot.py檔案中對應的file path
- 直接執行train_add_plot.py

```

[ ] cd /content/gdrive/MyDrive/yolo/1110智能加工上課/1110智能加工上課

/content/gdrive/MyDrive/yolo/1110智能加工上課/1110智能加工上課

[ ] #記得修改 .py 裡面的路徑
#還有圖片，檔案，權重跟資料夾都要上傳到google drive，圖片可以壓成zip上傳
!python train_add_plot.py

/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_qint8 = np.dtype([('qint8', np.int8, 1)])
/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_quint8 = np.dtype([('quint8', np.uint8, 1)])
/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_qint16 = np.dtype([('qint16', np.int16, 1)])
/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_quint16 = np.dtype([('quint16', np.uint16, 1)])
/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_qint32 = np.dtype([('qint32', np.int32, 1)])
/usr/local/lib/python3.7/dist-packages/tensorflow/python/framework/dtype
_np_resource = np.dtype([('resource', np.ubyte, 1)])
Using TensorFlow backend.
Create YOLOv4 model with 9 anchors and 6 classes.
WARNING:tensorflow:From /usr/local/lib/python3.7/dist-packages/tensorflow
Instructions for updating:
Colocations handled automatically by placer.
2021-11-24 10:16:10.939689: I tensorflow/core/platform/cpu_feature_guard
2021-11-24 10:16:10.945099: I tensorflow/core/platform/profile_utils/cpu
2021-11-24 10:16:10.945359: I tensorflow/compiler/xla/service/service.cc
2021-11-24 10:16:10.945413: I tensorflow/compiler/xla/service/service.cc
2021-11-24 10:16:11.090716: I tensorflow/stream_executor/cuda/cuda_gpu_e

```

```

train_add_plot.py X
135
136 if __name__ == "__main__":
137     # 标签的位置
138     annotation_path = '/content/gdrive/MyDrive/yolo/txt/2007_train.txt'
139     # 获取classes和anchor的位置
140     classes_path = 'model_data/voc_classes.txt'
141     anchors_path = 'model_data/yolo4_anchors.txt'
142     weights_path = '/content/gdrive/MyDrive/yolo/1110智能加工上課/1110智能加工上課/logs/ep031-loss2
143     # 获得classes和anchor
144     class_names = get_classes(classes_path)
145     anchors = get_anchors(anchors_path)
146     # 一共有多少类
147     num_classes = len(class_names)
148     num_anchors = len(anchors)
149     # 训练后的模型保存的位置
150     log_dir = 'logs/'
151     # 输入的shape大小
152     # 显存比较小可以使用416x416
153     # 现存比较大可以使用608x608
154     input_shape = (416, 416)
155     mosaic = True
156     Cosine_scheduler = True
157     label_smoothing = 0.01
158
159     # 清除session
160     K.clear_session()
161
162     # 输入的图像为
163     image_input = Input(shape=(None, None, 3))
164     h, w = input_shape

```

Colab訓練yolo模型-結果

- 執行前訓練(凍結訓練)時，1個epoch所需時間大約12分鐘，並於19個epoch提前結束訓練

```
[ ] 221/221 [=====] - 744s 3s/step - loss: 28.5174 - acc: 0.0000e+00 - val_loss: 32.0854 - val_acc: 0.0000e+00
Epoch 7/25
221/221 [=====] - 740s 3s/step - loss: 27.2846 - acc: 0.0000e+00 - val_loss: 24.8882 - val_acc: 0.0000e+00
Epoch 8/25
221/221 [=====] - 744s 3s/step - loss: 26.0638 - acc: 0.0000e+00 - val_loss: 30.6965 - val_acc: 0.0000e+00
Epoch 9/25
221/221 [=====] - 741s 3s/step - loss: 26.3254 - acc: 0.0000e+00 - val_loss: 24.8326 - val_acc: 0.0000e+00
Epoch 10/25
221/221 [=====] - 743s 3s/step - loss: 25.9869 - acc: 0.0000e+00 - val_loss: 24.3836 - val_acc: 0.0000e+00
Epoch 11/25
221/221 [=====] - 749s 3s/step - loss: 25.5879 - acc: 0.0000e+00 - val_loss: 24.0117 - val_acc: 0.0000e+00
Epoch 12/25
221/221 [=====] - 747s 3s/step - loss: 25.3418 - acc: 0.0000e+00 - val_loss: 23.3843 - val_acc: 0.0000e+00
Epoch 13/25
221/221 [=====] - 739s 3s/step - loss: 24.3204 - acc: 0.0000e+00 - val_loss: 22.8992 - val_acc: 0.0000e+00
Epoch 14/25
221/221 [=====] - 734s 3s/step - loss: 24.4160 - acc: 0.0000e+00 - val_loss: 23.5048 - val_acc: 0.0000e+00
Epoch 15/25
221/221 [=====] - 733s 3s/step - loss: 24.7160 - acc: 0.0000e+00 - val_loss: 23.2650 - val_acc: 0.0000e+00
Epoch 16/25
221/221 [=====] - 734s 3s/step - loss: 25.1054 - acc: 0.0000e+00 - val_loss: 23.5475 - val_acc: 0.0000e+00
Epoch 17/25
221/221 [=====] - 726s 3s/step - loss: 24.7975 - acc: 0.0000e+00 - val_loss: 23.4733 - val_acc: 0.0000e+00
Epoch 18/25
221/221 [=====] - 717s 3s/step - loss: 24.6411 - acc: 0.0000e+00 - val_loss: 23.2077 - val_acc: 0.0000e+00
Epoch 19/25
221/221 [=====] - 717s 3s/step - loss: 24.4154 - acc: 0.0000e+00 - val_loss: 22.9635 - val_acc: 0.0000e+00
Epoch 00019: early stopping
<Figure size 640x480 with 1 Axes>
<Figure size 640x480 with 1 Axes>
```

Colab訓練yolo模型-結果2

- 執行後訓練時，1個epoch所需時間大約15~20分鐘。
- 於32個epoch開始移至Kaggle進行訓練

Epoch 26/50

2021-11-24 15:08:37.697357: I tensorflow/stream_executor/dso_loader.cc:152] successfully opened CUDA library libcublas.so.10.0 locally

2021-11-24 15:08:39.012836: W tensorflow/core/common_runtime/bfc_allocator.cc:211] Allocator (GPU_0_bfc) ran out of memory trying to allocate 2.17GiB. The caller indicates tha

2021-11-24 15:08:39.388738: W tensorflow/core/common_runtime/bfc_allocator.cc:211] Allocator (GPU_0_bfc) ran out of memory trying to allocate 1.16GiB. The caller indicates tha

2021-11-24 15:08:39.388817: W tensorflow/core/common_runtime/bfc_allocator.cc:211] Allocator (GPU_0_bfc) ran out of memory trying to allocate 2.17GiB. The caller indicates tha

442/442 [=====] - 958s 2s/step - loss: 23.4923 - acc: 0.0000e+00 - val_loss: 21.6747 - val_acc: 0.0545

Epoch 27/50

442/442 [=====] - 922s 2s/step - loss: 22.8505 - acc: 0.0000e+00 - val_loss: 21.3094 - val_acc: 0.1182

Epoch 28/50

442/442 [=====] - 914s 2s/step - loss: 22.3120 - acc: 0.0000e+00 - val_loss: 20.6660 - val_acc: 0.2364

Epoch 29/50

12/442 [.....] - ETA: 12:39 - loss: 21.5412 - acc: 0.0000e+00Traceback (most recent call last):

File "train_add_plot.py", line 308, in <module>

callbacks=[logging, checkpoint, reduce_lr, early_stopping])

File "/usr/local/lib/python3.7/dist-packages/keras/legacy/interfaces.py", line 91, in wrapper

return func(*args, **kwargs)

File "/usr/local/lib/python3.7/dist-packages/keras/engine/training.py", line 2224, in fit_generator

Kaggle訓練yolo模型-1

- 準備對應環境
- Kaggle預設之GPU驅動版本為Cuda 11.0.0，需安裝Cuda 10.0.0以供tensorflow-gpu 1.13.2使用

```
[ ]: !apt-get install -y libcublas10 cuda-libraries-10.0 libcudnn7=7.6.5.32-1+cuda10.1 cuda-cudart-10.0
```

⌵ Expand

```
[ ]: !pip uninstall tensorflow -y
!pip uninstall tensorflow-gpu -y
!pip uninstall keras -y
!pip install tensorflow==1.13.2
!pip install tensorflow-gpu==1.13.2
!pip install keras==2.1.5

!pip uninstall h5py -y
!pip install h5py==2.10.0
```

+ Code

+ Markdown

Kaggle訓練yolo模型-2

- 先上傳image, xml以及train.txt檔, 取得dataset的filepath
- 於本地端更新train_add_plot.py檔並上傳
- 於Kaggle上執行train_add_plot.py檔

```
if __name__ == "__main__":  
    # 标签的位置  
    annotation_path = '../input/try-test2-1/2007_train.txt'  
    # 获取classes和anchor的位置  
    classes_path = '../input/try-test/model_data/voc_classes.txt'  
    anchors_path = '../input/try-test/model_data/yolo4_anchors.txt'  
    weights_path = '../input/final-test2/ep031-loss20.912-val_loss19.559.h5'
```

```
Init_epoch = 31
```

```
# 後訓練
```

```
if Init_epoch>=25:  
    Freeze_epoch = Init_epoch  
    Epoch = 50  
    batch_size = 4
```

Data + Add data ^

input (490.88 MB)

- final-test
- final-test2
 - model_data
 - nets
 - utils
 - yolo4

2007_test.txt

2007_train.txt

2007_val.txt

ep031-loss20.912-val_loss19...

train_add_plot.py

- final-test3
- try-test
- try-test2-1
- yolo-midterm
 - jpg
 - xml

Kaggle訓練yolo模型-結果

✕ Viewing Version 4: ✓ Save & Run All • November 25, 2021, 1:45 PM

Go to Viewer




t instead: <https://pip.pypa.io/warnings/venv>

In [4]:

```
!python ../input/final-test3/train_add_plot.py
```

```
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:526: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
rsion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype [("qint8", np.int8, 1)]
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:527: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
rsion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype [("qint8", np.int8, 1)]
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:528: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
rsion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype [("qint16", np.int16, 1)]
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:529: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
rsion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype [("qint16", np.int16, 1)]
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:530: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
rsion of numpy, it will be understood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype [("qint32", np.int32, 1)]
/opt/conda/lib/python3.7/site-packages/tensorflow/python/framework/dtypes.py:535: Future
Warning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future ve
```

Version History

	Version 4 Save & Run All • Diff: +1 -1 Ran in 3 hours and 16 minutes	2d ago ...
	Version 3 Save & Run All • Diff: +1 -1 Cancelled after 3 minutes and 59 seconds	2d ago ...
	Version 2 Save & Run All • Diff: +9 -9 Cancelled after 1 hour and 46 minutes	2d ago ...
	Version 1 Save & Run All • Diff: +32 -0 Ran in 26 seconds	2d ago ...

Kaggle訓練yolo模型-結果2

- 後訓練部分，一個epoch約7分鐘上下，20個epoch約花了3個小時執行(含環境安裝)

```
val_acc: 0.6380













11102.9s    2    Epoch 50/50

11470.2s    3    1/884 [.....] - ETA: 4:20 - loss: 16.5455 - acc: 0.0000e+00 2/884
[.....] - ETA: 4:39 - loss: 15.2620 - acc: 0.0000e+00 3/884
[.....] - ETA: 4:51 - loss: 14.9197 - acc: 0.0000e+00 4/884
[.....] - ETA: 4:49 - loss: 15.1262 - acc: 0.0000e+00 5/884
[.....] - ETA: 4:45 - loss: 14.8999 - acc: 0.0000e+00 6/884
[.....] - ETA: 4:48 - loss: 14.8296 - acc: 0.0000e+00 7/884
[.....] - ETA: 4:46 - loss: 14.7719 - acc: 0.0000e+00 8/884
[.....] - ETA: 4:45 - loss: 15.4861 - acc: 0.0000e+00 9/884
[.....] - ETA: 4:48 - loss: 15.3480 - acc: 0.0000e+00 10/884
[.....] - ETA: 4:48 - loss: 15.1904 - acc: 0.0000e+00 11/884
[.....] - ETA: 4:47 - loss: 15.1801 - acc: 0.0000e+00 12/884
[.....] - ETA: 4:46 - loss: 15.1757 - acc: 0.0000e+00 13/884
[.....] - ETA: 4:46 - loss: 15.1043 - acc: 0.0000e+00 14/884
[.....] - ETA: 5:13 - loss: 15.0114 - acc: 0.0000e+00 15/884
[.....] - ETA: 6:17 - loss: 14.9448 - acc: 0.0000e+00 16/884
[.....] - ETA: 6:53 - loss: 14.9784 - acc: 0.0000e+00 17/884
[.....] - ETA: 7:22 - loss: 14.9079 - acc: 0.0000e+00 18/884
[.....] - ETA: 7:34 - loss: 14.9197 - acc: 0.0000e+00 19/884
[.....] - ETA: 7:36 - loss: 14.8689 - acc: 0.0000e+00 20/884
[.....] - ETA: 7:39 - loss: 14.8330 - acc: 0.0000e+00 21/884
[.....] - ETA: 7:40 - loss: 14.9253 - acc: 0.0000e+00 22/884
[.....] - ETA: 7:41 - loss: 14.8815 - acc: 0.0000e+00 23/884
[.....] - ETA: 7:43 - loss: 14.8681 - acc: 0.0000e+00 24/884
- - -
```

Kaggle訓練yolo模型-結果3

- Output可下載

logs

 ep033-loss20.509-val_... 256.95 MB	 ep034-loss20.093-val_... 256.95 MB	 ep035-loss19.355-val_I... 256.95 MB	 ep036-loss19.347-val_I... 256.95 MB
 ep037-loss18.623-val_I... 256.95 MB	 ep038-loss18.295-val_I... 256.95 MB	 ep039-loss18.391-val_I... 256.95 MB	 ep040-loss17.577-val_I... 256.95 MB
 ep041-loss17.195-val_lo... 256.95 MB	 ep042-loss17.001-val_I... 256.95 MB	 ep043-loss17.155-val_I... 256.95 MB	 ep044-loss16.639-val_I... 256.95 MB

Output

- logs
 - ep033-loss20.509-v...
 - ep034-loss20.093-v...
 - ep035-loss19.355-v...
 - ep036-loss19.347-v...
 - ep037-loss18.623-v...
 - ep038-loss18.295-v...
 - ep039-loss18.391-v...
 - ep040-loss17.577-v...
 - ep041-loss17.195-val...
 - ep042-loss17.001-va...
 - ep043-loss17.155-va...
 - ep044-loss16.639-v...
 - ep045-loss16.200-v...
 - ep046-loss15.704-v...
 - ep047-loss15.255-v...
 - ep048-loss14.859-v...
 - ep049-loss14.657-v...
 - ep050-loss14.578-v...
 - events.out.tfevents.1...
 - last1.h5

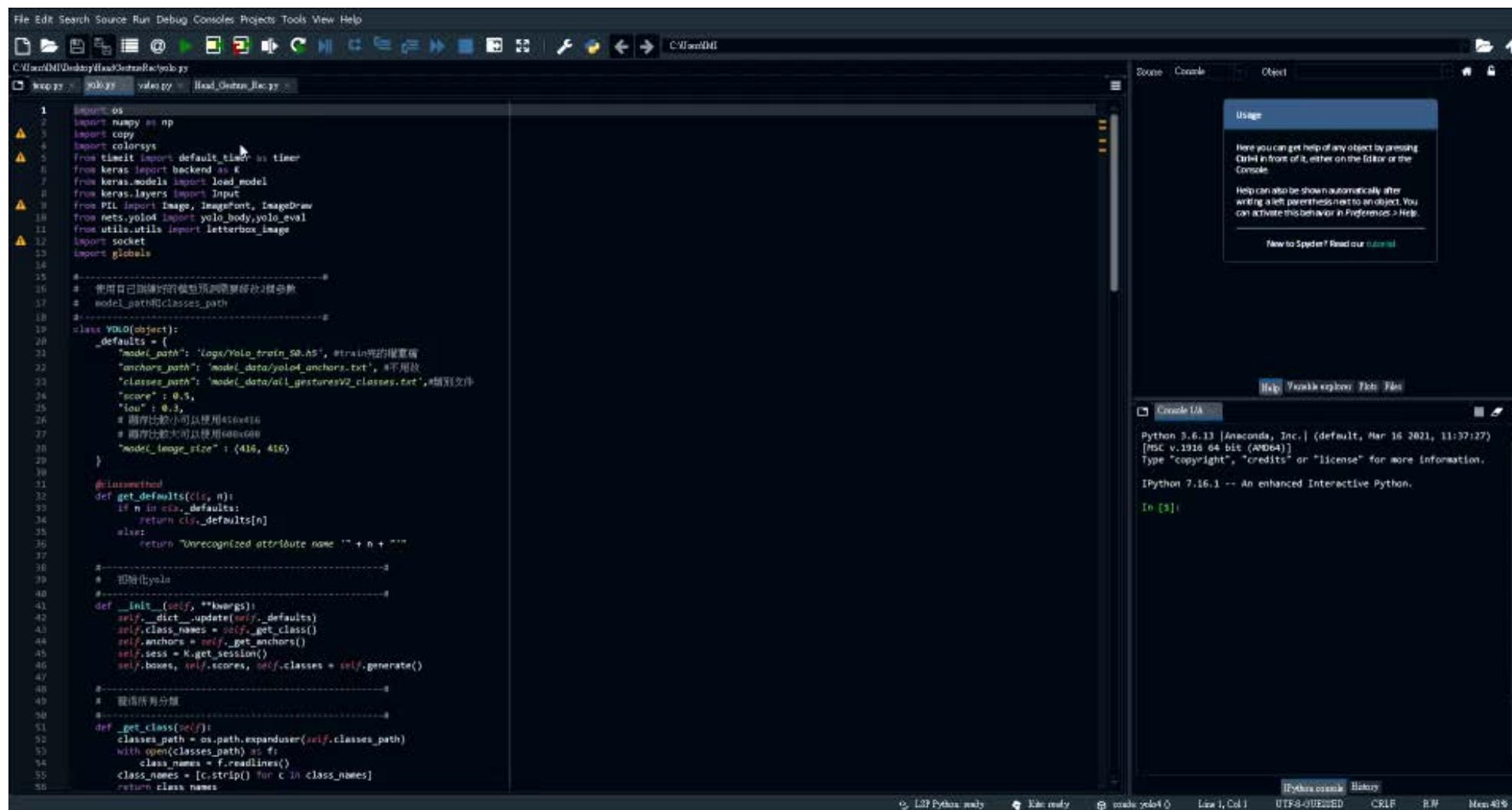
影片成果 / 未來展望

成果結語

06

執行成果

- 開啟並 run yolo.py (先) 和 video.py (後)



```
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\HUI\Desktop\HandGesturesRac\yolo.py
yolo.py yolo.py Hand_Gestures_Rac.py
1 import os
2 import numpy as np
3 import copy
4 import colorsys
5 from timeit import default_timer as timer
6 from keras import backend as K
7 from keras.models import load_model
8 from keras.layers import Input
9 from PIL import Image, ImageFont, ImageDraw
10 from nets.yolo4 import yolo_body, yolo_eval
11 from utils.utils import letterbox_image
12 import socket
13 import glob
14
15 #-----#
16 # 使用自己訓練好的模型預測視頻錄放2個參數
17 # model_path和classes_path
18 #-----#
19 class YOLO(object):
20     _defaults = {
21         'model_path': 'logs/Yolo_train_50.h5', #訓練好的模型檔
22         'anchors_path': 'model_data/yolo4_anchors.txt', #不用放
23         'classes_path': 'model_data/all_gesturesV2_classes.txt', #類別文件
24         'score': 0.5,
25         'iou': 0.3,
26         # 儲存比較小可以使用600x600
27         # 儲存比較大則可以使用600x600
28         'model_image_size': (416, 416)
29     }
30
31     @classmethod
32     def get_defaults(cls, n):
33         if n in cls._defaults:
34             return cls._defaults[n]
35         else:
36             return "Unrecognized attribute name '%s'" % n
37
38 #-----#
39 # 初始化yolo
40 #-----#
41 def __init__(self, **kwargs):
42     self._dict_.update(self._defaults)
43     self.class_names = self.get_class()
44     self.anchors = self.get_anchors()
45     self.sess = K.get_session()
46     self.bboxes, self.scores, self.classes = self.generate()
47
48 #-----#
49 # 讀出所有分類
50 #-----#
51 def get_class(self):
52     classes_path = os.path.expanduser(self.classes_path)
53     with open(classes_path) as f:
54         class_names = f.readlines()
55         class_names = [c.strip() for c in class_names]
56         return class_names
```

Usage

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.

New to Spyder? Read our tutorial

Python 3.6.13 [Anaconda, Inc.] (default, Mar 16 2021, 11:37:12)
[MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [3]:

辨識成果

- 執行 video.py 跑出 demo 影片



結語

- 前半使用Colab，須維持網頁開啟，導致筆電執行受到些許影響。後半移至Kaggle訓練解決該問題。
- 因為並非一次執行到底，因此無法透過history參數畫圖，且plt.plot在log結果中無法顯示。若從頭使用Kaggle訓練一次，並將plot改成saveimg，應能解決該問題，獲得acc與loss之圖片。
- Colab與Kaggle之環境版本皆比yolo訓練所需的版本新，因此會出現function參數不一致導致error之問題，需加上安裝對應版本之指令，將tensorflow與keras降級後才能執行。
- Colab之GPU限量不明確，曾有兩個帳號都被限制無法使用GPU的情況。

未來展望

- 可以再新增資料集
 1. 手勢資料集太單純，辨識結果有限也不夠準確，需要在白色背景以及某種角度手勢才會辨識出來。
 2. 目前模型是訓練的手勢資料不夠充足，未來也可以把更多不同人的手勢也加入，使訓練更準確。



THANKS FOR LISTENING

感謝聆聽

組員：古若喬 / 陳品修 / 謝慈芯

2021/11/17