專題報告

題目：自動發牌機
2019/02~2019/09

105062303 黃允亮

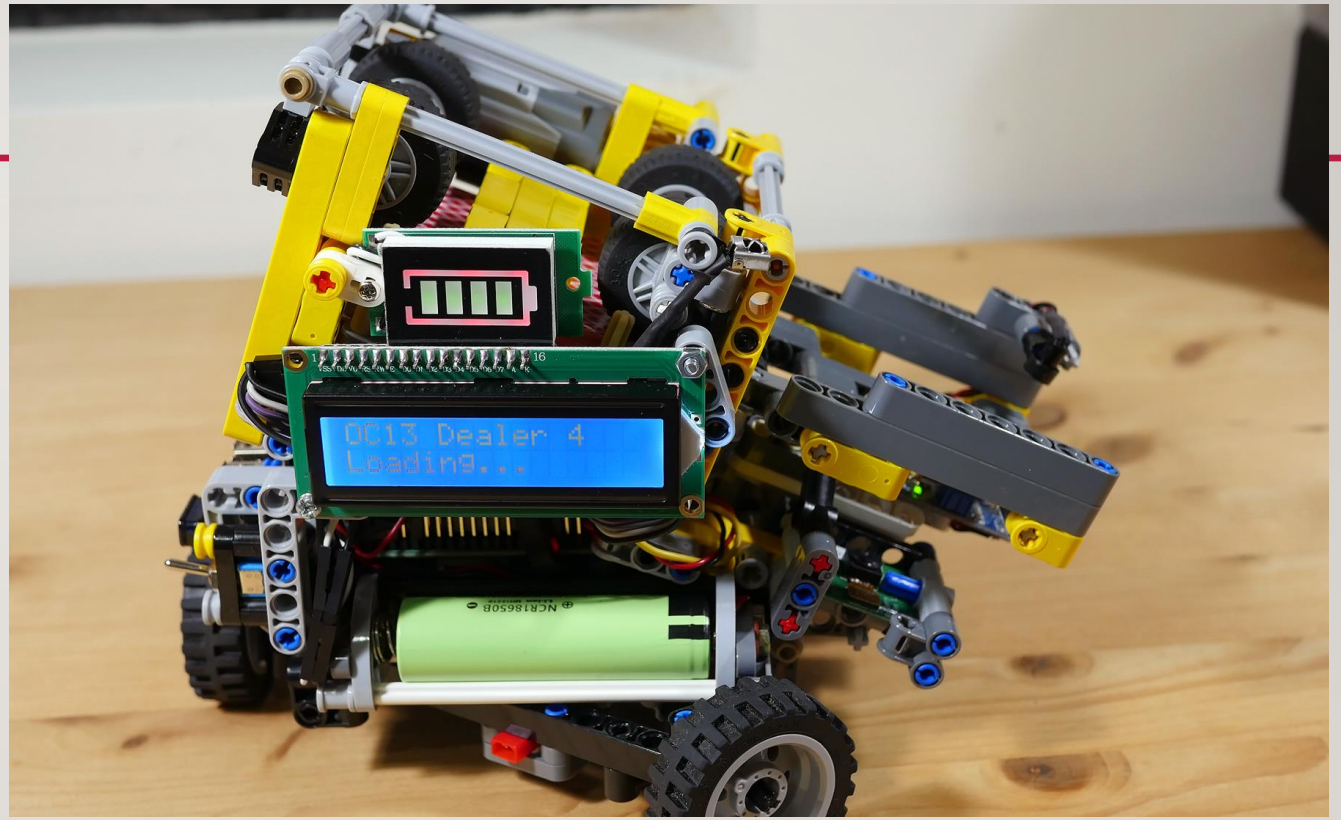
105062338 周廷駿

OUTLINE

- 自動發牌機簡介
- 已完成進度（發牌機基本架構）
- 已完成進度（視覺辨識）
- 結合視覺辨識至 pi 上
- 總結與心得

自動發牌機簡介

自動發牌機



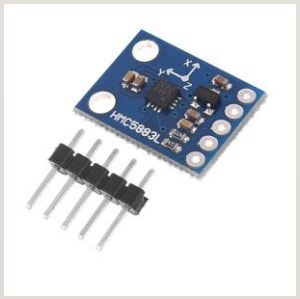
製作動機

- 親朋好友間在玩撲克牌時總會有誰要發牌的問題，因此我們用物聯網的概念設計了一個自動發牌機。

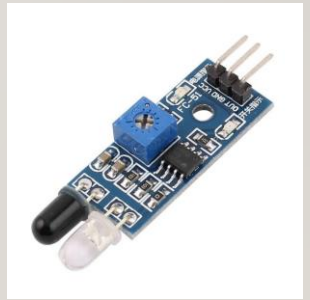
發牌機結構與特色

- 發牌機主要是由樹莓派與電子元件進行控制，它最大的特點是結合了深度學習的模型，在開始發牌前會利用視覺辨識，辨別出玩家的位置，並透過電子羅盤記住。再來透過馬達驅動發牌機旋轉，朝玩家的位置發射牌張，發射牌張的同時也仍須透過感光元件控制發牌的張數，避免一次發兩張或以上的牌。

Raspberry pi 3



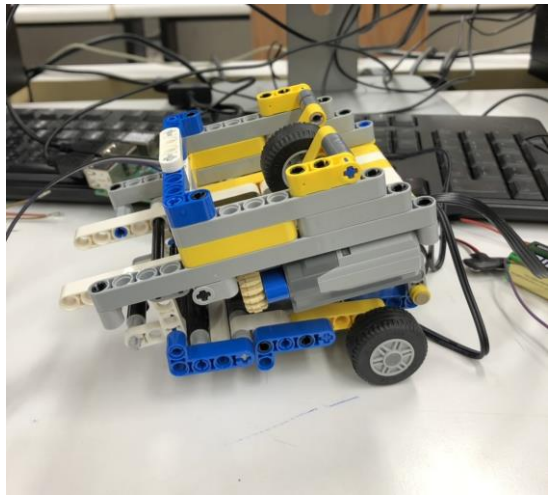
電子羅盤

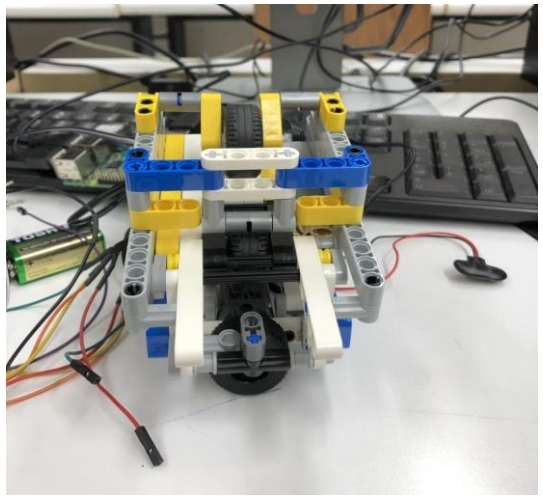


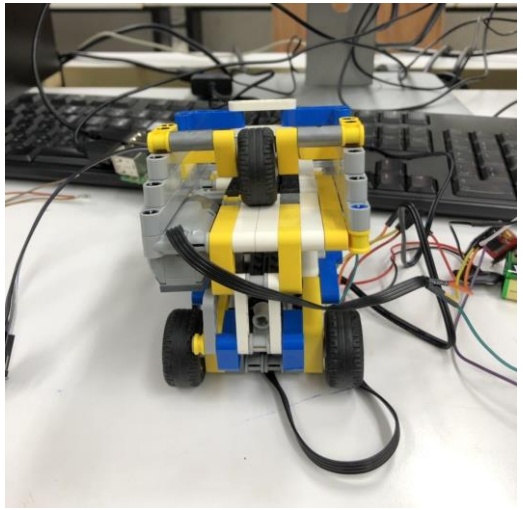
紅外線避障模組

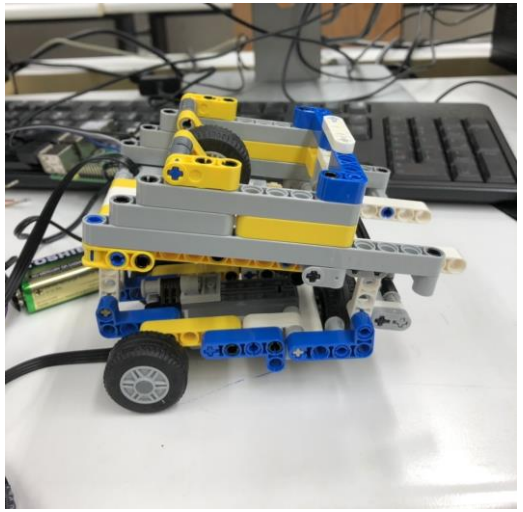
已完成進度（發牌機基本架構）

組裝全數完成





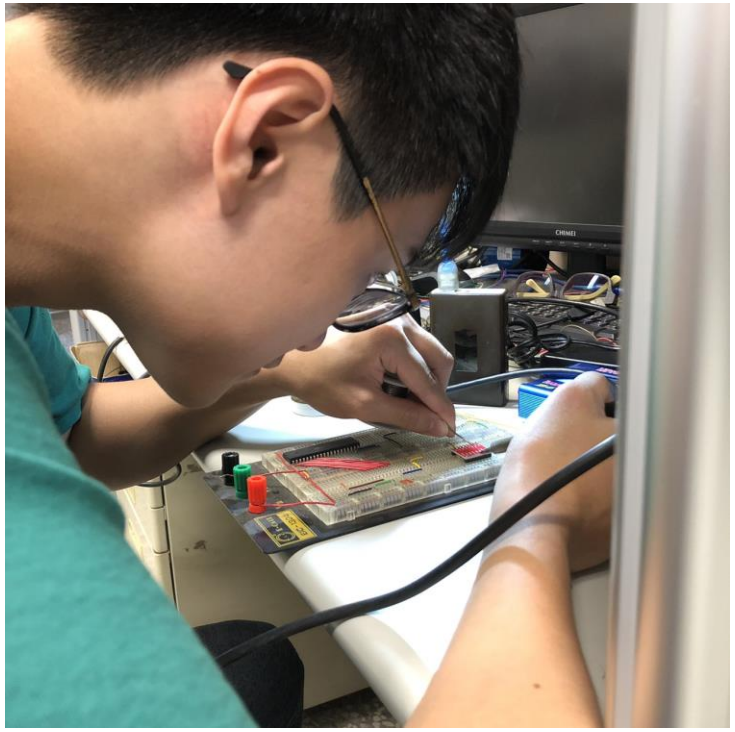




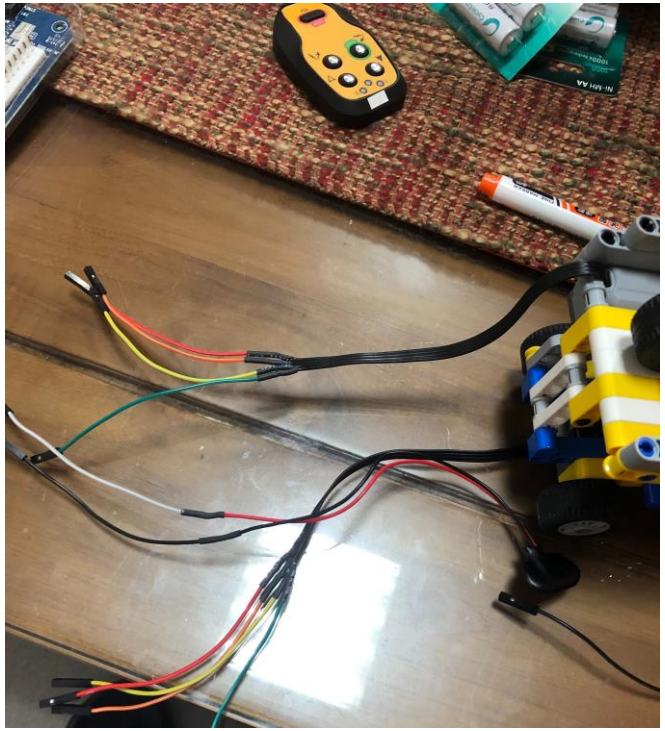
材料備齊

- TB6612驅動模組
- QMC5883L電子羅盤
- 紅外線避障模組
- 紅外線感測器、遙控器
- 9V 電池

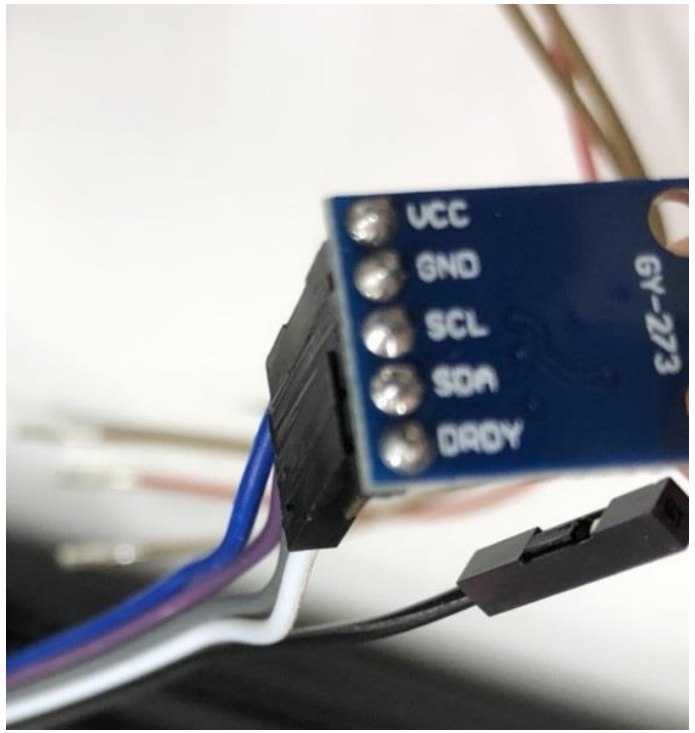
焊接電路板



馬達線壓杜邦端子、裝杜邦頭

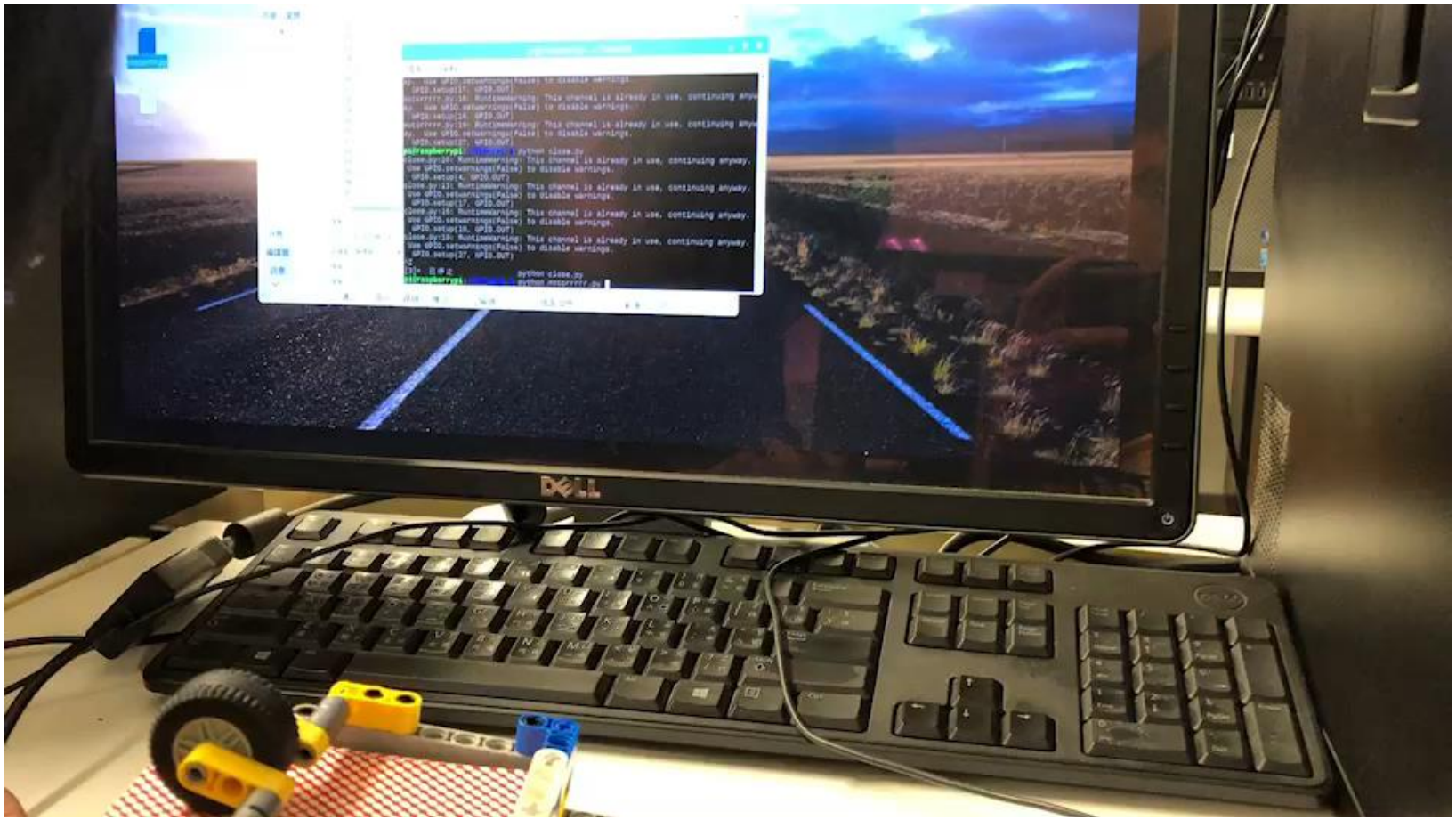


電子羅盤定位測試

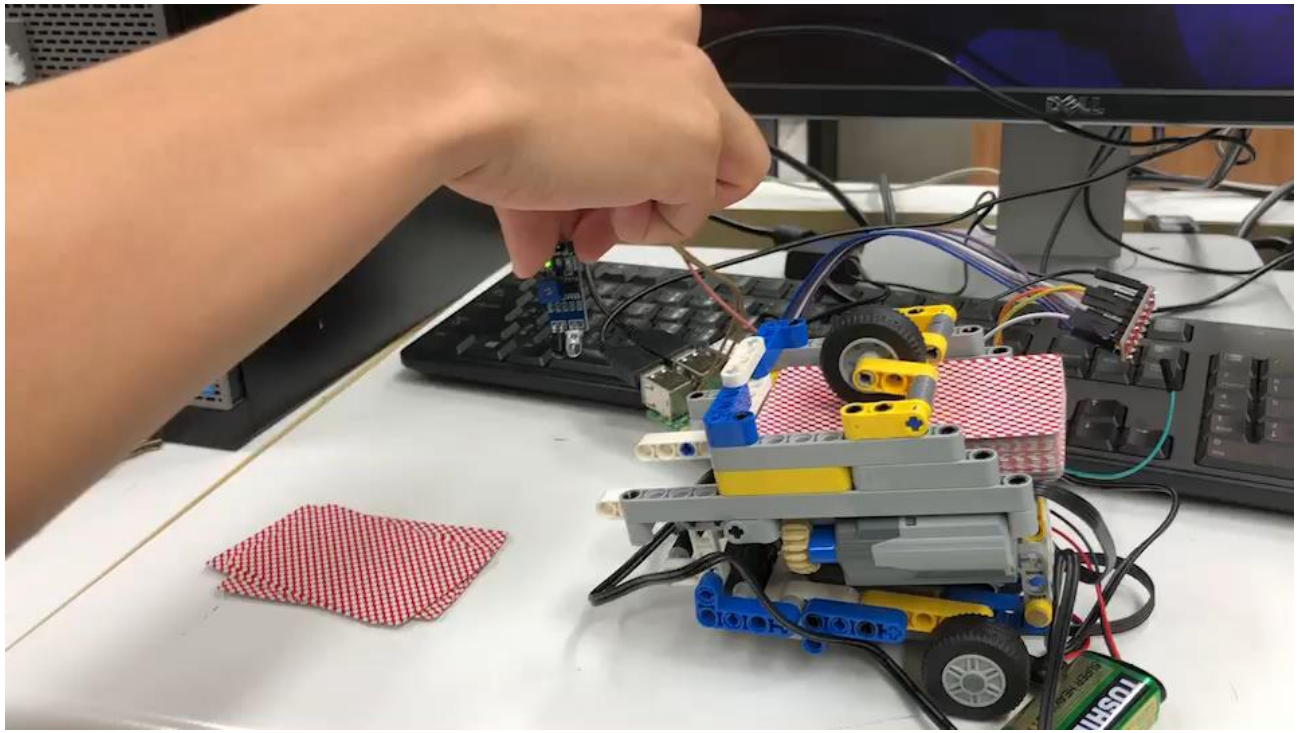



```
pi@raspberrypi:~/Desktop/py-qmc5883l-master $ python qmc_test.py
[-1837.0, 1485.0]
[-1832.0, 1465.0]
[-1797.0, 1485.0]
[-1792.0, 1465.0]
[-1725.0, 1455.0]
[-1647.0, 1485.0]
[-1635.0, 1455.0]
[-1630.0, 1475.0]
[-1590.0, 1517.0]
[-1507.0, 1527.0]
[-1400.0, 1507.0]
[-1377.0, 1517.0]
```

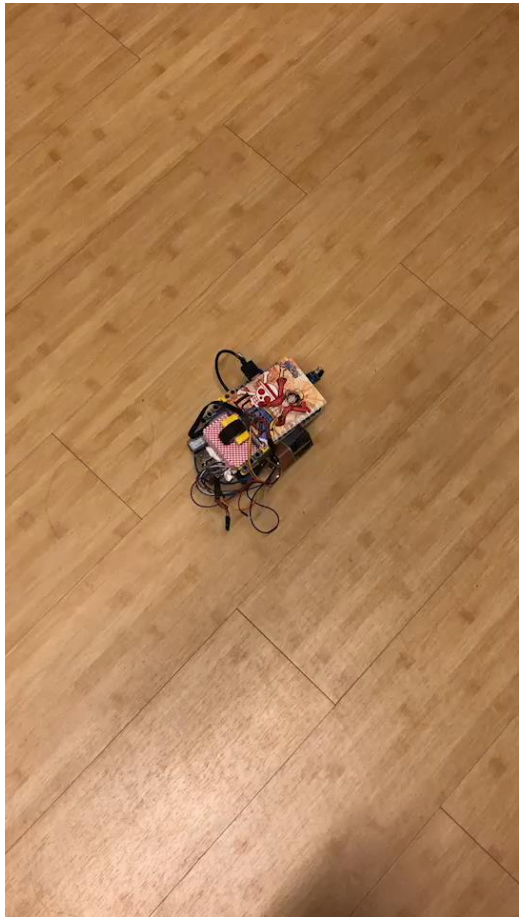
發牌動作



紅外線避障模組使用



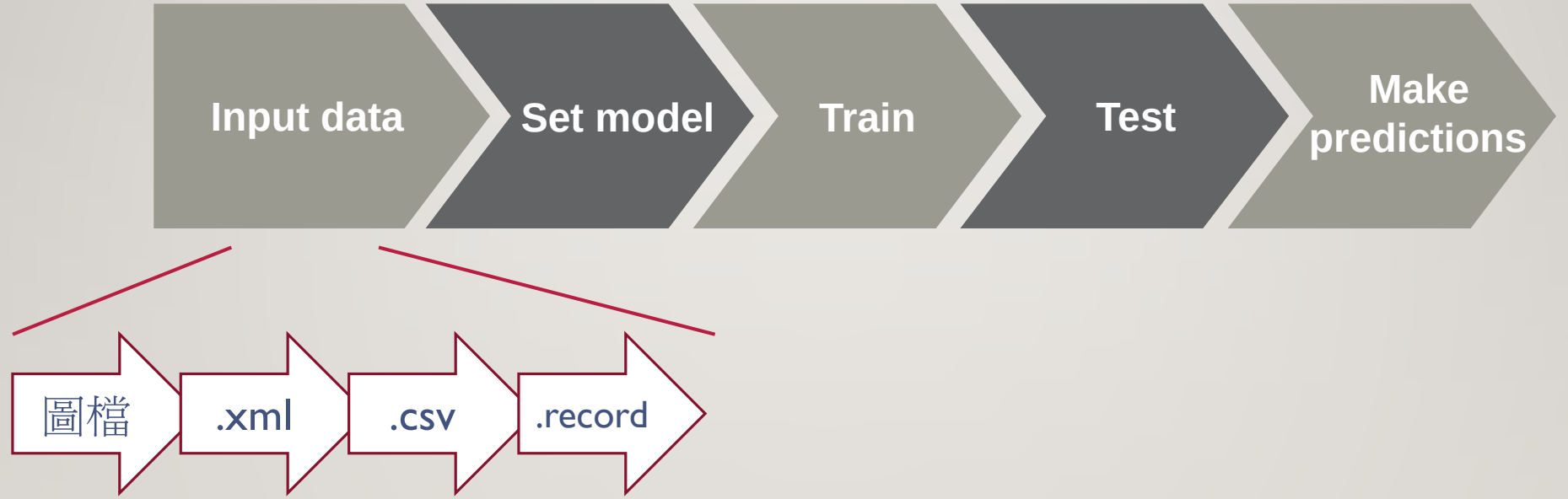
3人發牌 (雙馬達轉動)



4人發牌 (雙馬達轉動)

已完成進度（視覺辨識）

學習使用 TENSORFLOW



目標：實作單一物件偵測


kite: 99%
kite: 97%
kite: 82%
kite: 90%
kite: 85%
kite: 54%
kite: 92%
person: 76%
person: 54%
pers
person: 97%
person: 86%
person: 99%
person: 99%

目標立牌

訓練模型

① 拍 100 張目標立牌照片

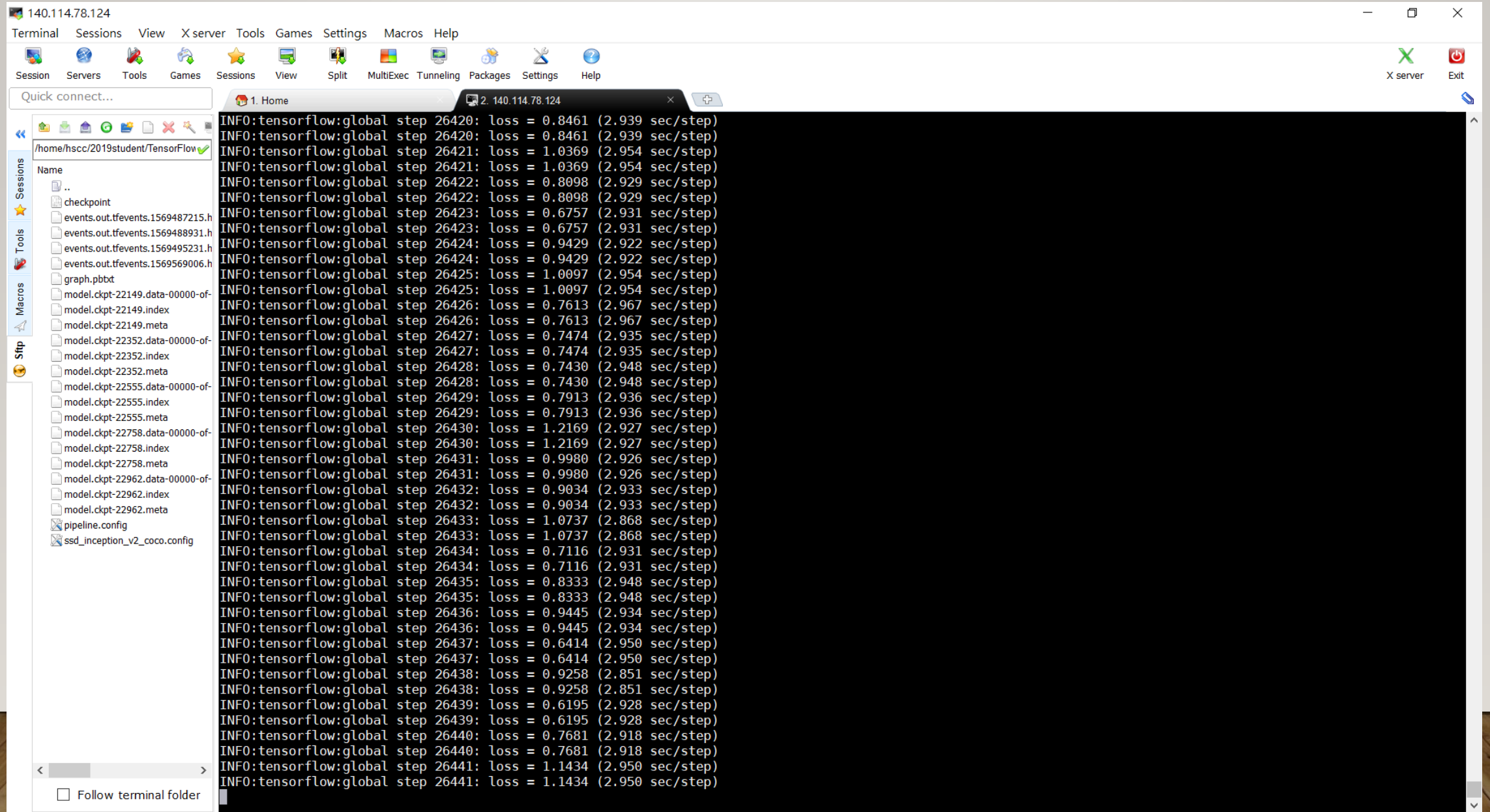
② 用 labelImg 框出照片內的目標物

③ 以 9:1 的比例分配照片至 train 及 test 資料夾

④ 將照片轉檔 (.jpg -> .xml -> .csv -> .record)

⑤ 訓練模型

訓練模型

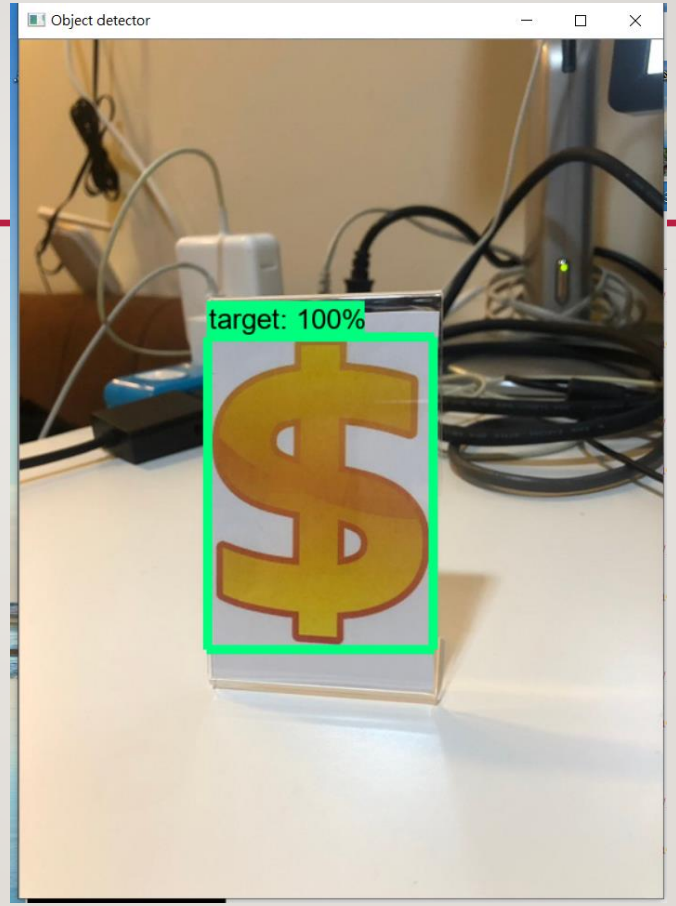

140.114.78.124
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
1. Home
2. 140.114.78.124
/home/hscc/2019student/TensorFlow
Name
..
checkpoint
events.out.tfevents.1569487215.h
events.out.tfevents.1569488931.h
events.out.tfevents.1569495231.h
events.out.tfevents.1569569006.h
graph.pbtxt
model.ckpt-22149.data-00000-of-
model.ckpt-22149.index
model.ckpt-22149.meta
model.ckpt-22352.data-00000-of-
model.ckpt-22352.index
model.ckpt-22352.meta
model.ckpt-22555.data-00000-of-
model.ckpt-22555.index
model.ckpt-22555.meta
model.ckpt-22758.data-00000-of-
model.ckpt-22758.index
model.ckpt-22758.meta
model.ckpt-22962.data-00000-of-
model.ckpt-22962.index
model.ckpt-22962.meta
pipeline.config
ssd_inception_v2_coco.config
Follow terminal folder
Sessions Tools Macros Sftp
INFO:tensorflow:global step 26420: loss = 0.8461 (2.939 sec/step)
INFO:tensorflow:global step 26420: loss = 0.8461 (2.939 sec/step)
INFO:tensorflow:global step 26421: loss = 1.0369 (2.954 sec/step)
INFO:tensorflow:global step 26421: loss = 1.0369 (2.954 sec/step)
INFO:tensorflow:global step 26422: loss = 0.8098 (2.929 sec/step)
INFO:tensorflow:global step 26422: loss = 0.8098 (2.929 sec/step)
INFO:tensorflow:global step 26423: loss = 0.6757 (2.931 sec/step)
INFO:tensorflow:global step 26423: loss = 0.6757 (2.931 sec/step)
INFO:tensorflow:global step 26424: loss = 0.9429 (2.922 sec/step)
INFO:tensorflow:global step 26424: loss = 0.9429 (2.922 sec/step)
INFO:tensorflow:global step 26425: loss = 1.0097 (2.954 sec/step)
INFO:tensorflow:global step 26425: loss = 1.0097 (2.954 sec/step)
INFO:tensorflow:global step 26426: loss = 0.7613 (2.967 sec/step)
INFO:tensorflow:global step 26426: loss = 0.7613 (2.967 sec/step)
INFO:tensorflow:global step 26427: loss = 0.7474 (2.935 sec/step)
INFO:tensorflow:global step 26427: loss = 0.7474 (2.935 sec/step)
INFO:tensorflow:global step 26428: loss = 0.7430 (2.948 sec/step)
INFO:tensorflow:global step 26428: loss = 0.7430 (2.948 sec/step)
INFO:tensorflow:global step 26429: loss = 0.7913 (2.936 sec/step)
INFO:tensorflow:global step 26429: loss = 0.7913 (2.936 sec/step)
INFO:tensorflow:global step 26430: loss = 1.2169 (2.927 sec/step)
INFO:tensorflow:global step 26430: loss = 1.2169 (2.927 sec/step)
INFO:tensorflow:global step 26431: loss = 0.9980 (2.926 sec/step)
INFO:tensorflow:global step 26431: loss = 0.9980 (2.926 sec/step)
INFO:tensorflow:global step 26432: loss = 0.9034 (2.933 sec/step)
INFO:tensorflow:global step 26432: loss = 0.9034 (2.933 sec/step)
INFO:tensorflow:global step 26433: loss = 1.0737 (2.868 sec/step)
INFO:tensorflow:global step 26433: loss = 1.0737 (2.868 sec/step)
INFO:tensorflow:global step 26434: loss = 0.7116 (2.931 sec/step)
INFO:tensorflow:global step 26434: loss = 0.7116 (2.931 sec/step)
INFO:tensorflow:global step 26435: loss = 0.8333 (2.948 sec/step)
INFO:tensorflow:global step 26435: loss = 0.8333 (2.948 sec/step)
INFO:tensorflow:global step 26436: loss = 0.9445 (2.934 sec/step)
INFO:tensorflow:global step 26436: loss = 0.9445 (2.934 sec/step)
INFO:tensorflow:global step 26437: loss = 0.6414 (2.950 sec/step)
INFO:tensorflow:global step 26437: loss = 0.6414 (2.950 sec/step)
INFO:tensorflow:global step 26438: loss = 0.9258 (2.851 sec/step)
INFO:tensorflow:global step 26438: loss = 0.9258 (2.851 sec/step)
INFO:tensorflow:global step 26439: loss = 0.6195 (2.928 sec/step)
INFO:tensorflow:global step 26439: loss = 0.6195 (2.928 sec/step)
INFO:tensorflow:global step 26440: loss = 0.7681 (2.918 sec/step)
INFO:tensorflow:global step 26440: loss = 0.7681 (2.918 sec/step)
INFO:tensorflow:global step 26441: loss = 1.1434 (2.950 sec/step)
INFO:tensorflow:global step 26441: loss = 1.1434 (2.950 sec/step)

DEMO

- 靜態畫面實測：

✓左圖有目標立牌，可看到模型正確將其框出。

✓右圖沒有目標立牌，因此模型未框出任何物體。



DEMO

- 動態畫面實測：

結合視覺辨識至 pi 上

ALGORITHM

- 將 pi 的鏡頭裝在發牌機上，每隔10度拍一張照片，拍完一圈共拍36張。
- 將36張照片全數做目標偵測並做編號，假設連續偵測到 n 張照片存在目標的機率 >90%，取第 n/2 張的照片用電子羅盤紀錄其目標位置。（第 n/2 張照片的位置最貼近玩家位置）

總結與心得

- 黃允亮：忽然間大學專題就到了尾聲，從一開始毫無頭緒的發想，經過每兩週一次的meeting，我們理想中的自動發牌機大致上是圓滿完成。
- 其中讓我印象最深刻的是研究視覺辨識，因為我先前完全沒接觸過深度學習相關課程，需要不斷在網路上蒐集教學資料及各式錯誤的解決方法，教授也常在開會時提到自我學習的重要性，包括安裝一個先前完全陌生的軟體，到實際學會應用。
- 這次專題製作過程中，每次實踐出新功能時都讓人十分有成就感，也很慶幸有隊友能互相協助。我很幸運地學習到物聯網以及人工智慧這兩大近年來熱門的技術，相信對未來一定是有極大的幫助，最後要非常感謝教授和助教們的指導及幫忙。

Thanks for listening!