

(B05) 專題題目:圖像辨識自走車

I. 成員名單與分工:

- 1091541 陳律君: 模型訓練
- 1091550 劉佳華: 撰寫此次專題功能程式碼
- 1093305 陳鈺汶: 資料庫以及網頁架設

以上為大略分工, 但其實我們比較偏向互相幫忙, 很多時候遇到問題都是透過大家的努力一起解決的。

II. 專題功能說明:

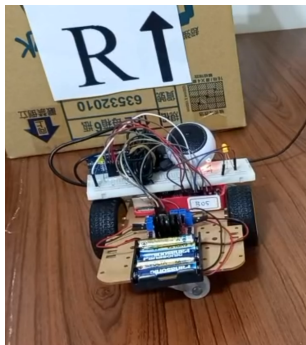
透過我們自己訓練的兩個模型讓自走車辨識特定標示做出相對應的動作。在辨識到一張紙上同時出現箭頭與字母(L或R)時車子會首先朝著箭頭的方向前行並且運用超聲波測距傳感器使其在距離紙張大約25公分處時停下再根據紙上字母樣式朝向正確方向轉彎, 其中L代表向左轉, R代表向右轉。

麵包板兩側的LED燈在這裡是作為方向燈使用, 與旋轉方向同側的LED燈會亮起呈現即將旋轉的方向並在完成旋轉後熄滅, 與此同時speaker會依旋轉方向喊出:「向左(右)旋轉」, 而車子經過的每一段路徑都將被儲存在資料庫中(結合SQLite), 紀錄的內容包含走過該路段所花的時間以及車子的轉向等等, 路段與路段是以轉彎做為區隔。

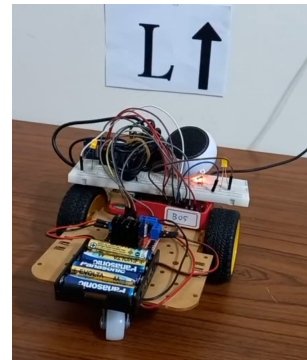
水滴形圖標代表目的地, 車子會在圖標前約5到10公分處停下, speaker喊出抵達終點並登記最後一段路徑資訊。我們有另外架設網頁顯示整個路程從開始到結束的所有紀錄(Flask)供使用者參考。

- 使用元件及硬體:
 1. 樹莓派
 2. 自走車
 3. Speaker
 4. 超聲波傳感器
 5. LED燈
- 模型(兩個):
 1. 辨識L和R
 2. 辨識箭頭與終點符號

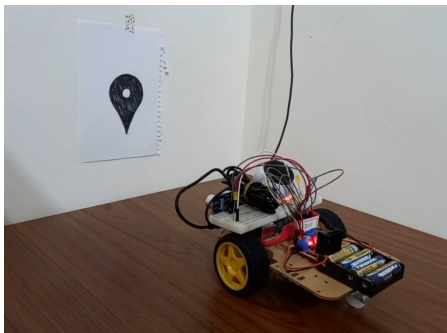
III. 專題操作與執行流程:



右轉彎(圖示1)



左轉彎(圖示2)



抵達終點(圖示3)

自走車行走紀錄

direction:

- 0: 左轉
- 1: 右轉
- 2: 抵達終點

ID	Now time	Spend time	direction
1	2023-01-18 01:42:22	9.56672215461731	1
1	2023-01-18 01:45:41	16.274681091308594	1
1	2023-01-18 01:47:39	7.49680228805542	1
2	2023-01-18 01:48:45	61.41791391372681	0
2	2023-01-18 01:51:19	9.668070316314697	1
3	2023-01-18 01:52:03	39.07472801208496	0
3	2023-01-18 01:56:58	7.305257558822632	1
4	2023-01-18 01:59:45	42.10233545303345	0
4	2023-01-18 02:02:39	33.11334204673767	0

自走車行走紀錄(圖示4)

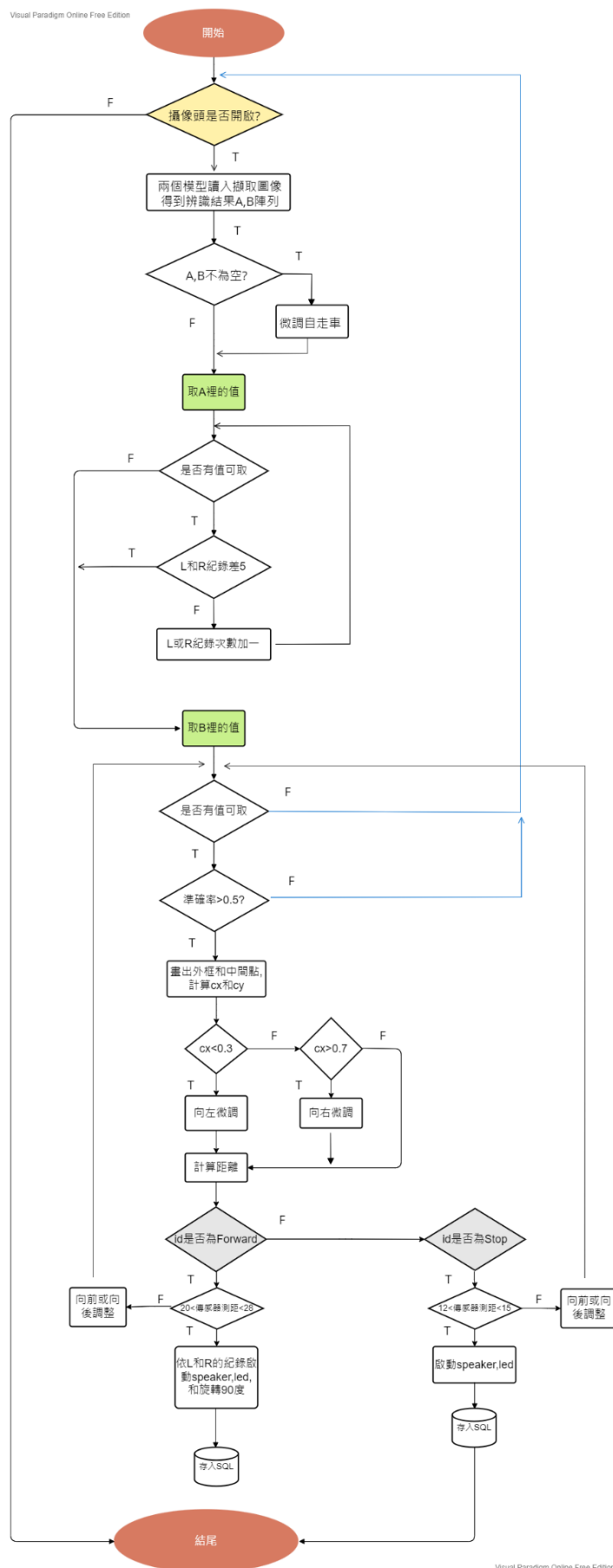
使用者由VNC執行主程式驅動自走車, 使自走車能辨識箭頭、R(圖示1)、L(圖示2)和終點(圖示3), 依照所看到的標示做出相對應的動作, 在自走車抵達終點後可藉由我們架設的網站(圖示4)去查看自走車行駛的各項紀錄。

IV. 硬體電路示意圖:



V. 軟體程式執行流程圖:

Visual Paradigm Online Free Edition



Visual Paradigm Online Free Edition

VI. 參考的課程實驗或是網路資源

- 參考課程實驗:
 1. lec03_gpio
 2. lec04_pwm&uart
 3. lec07_camera
 4. lec08_cv
 5. lec09_autocar
 6. lec10_smart_speaker
- 模型訓練:
 - https://www.tensorflow.org/install/source_windows
 - <https://github.com/tensorflow/models>
 - <https://github.com/protocolbuffers/protobuf/releases/download/v3.15.6/protocol-3.15.6-win64.zip>
 - <https://github.com/nicknochnack/GenerateTFRecord>
 - <https://github.com/nicknochnack/TFODCourse>
 - <https://colab.research.google.com/>
 - <http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/>
- 超聲波測距傳感器:
 - <https://www.gushiciku.cn/pl/p5ki/zh-tw>
 - https://atceiling.blogspot.com/2014/03/raspberry-pi_18.html#.VvoEs-J961t
- Windows 實現 sudo
 - <https://maplecheng.com/archives/271>
- TFLite:
 - https://hackmd.io/@flag23311033/B1ljFoTF?utm_source=preview-mode&utm_medium=rec
- SQLite
 - <https://docs.python.org/2/library/sqlite3.html>
- Flask
 - <https://blog.liang2.tw/posts/2015/09/flask-draw-member/>

VII. 開發最耗時的部份與原因

最耗時的部分是訓練模型，因為我們沒有這方面的經驗，所以花了很多時間摸索。整個過程可以說不太順利，從環境架設到正式訓練再到後面的轉檔以及應用於程式這整套過程發生了非常多的錯誤，包括環境架設中各種東西版本相衝、訓練出的模型無法使用或是準確率不高無法辨識物體、不成功的轉檔等等問題花了我們很多時間去解決。我們也嘗試了很多種方法訓練模型，從YOLOv5到Keras再到OpenCV到最終使用的TensorFlow。大多數的數據照片也都是由我們自己拍照而來，並使用Labelimg做框選製作數據集。以上種種原因即是為什麼訓練模型是我們耗費最多時間的部分。

VIII. 錄製完整專題操作流程之展示影片

<https://youtu.be/ESf0geBkUps>