

第十六屆盛群盃 HOLTEK MCU 創意大賽複賽報告

運用視覺 AI 之人機協作遙控系統

參賽編號：C-16

參賽隊員：蔡定維

日期：2021 年 11 月 17 日

零、摘要

本作品是運用視覺 AI 技術來追蹤人的手勢，透過藍牙無線通訊模組，以無線方式操控機器手，以及使用人臉辨識技術以解鎖登入管理系統，實現人機協作且具有人員使用權限管制的系統。利用攝影機先進行人臉辨識登入，登入後搭配 MediaPipe 手勢追蹤技術，辨識操作人員手指彎曲變化的角度，再利用藍牙模組傳送角度數據至機器手控制平台，控制各個伺服馬達轉動，以模仿手部動作。

壹、前言

目前市面上大部機器手都是以自動化方式，重複做一些標準化流程動作，雖然可減少人力，但靈活應變能力遠遠不及人類，因此，現在許多機器手都加上人機協作功能，利用搖桿或具有彎曲感測器的手套來進行控制，再加上人機介面以便監控。

但若使用手套來控制，在維修、清潔及成本方面會是個困擾，而在控管使用權也不好管理，因此，本次作品想運用近幾年討論度高的視覺 AI 技術，追蹤辨識人的手部動作，讓機器手與人結合實現人機協作，以增加機器手的靈活性，並加入人臉辨識登入系統，以便管理使用權限。

貳、工作原理

一、手勢 AI 追蹤系統

手勢追蹤系統是利用 Google MediaPipe Hand Tacking 技術，Google 採用了一個由 3 個模型組成的框架，包括：手掌識別模型 BlazePalm（用於識別手的整體框架和方向）、Landmark 模型（識別立體手部節點）、手勢識別模型（將識別到的節點分類成一系列手勢）。

其中 BlazePalm 可為 Landmark 模型提供準確建材的手掌圖像，這大大降低了對旋轉、轉化和縮放等數據增強方式的依賴，設法將更多計算能力用在提高預測準確性上。

判斷手指彎曲是透過 Hand Landmark 模型辨識出人的 21 點關節點，如圖 1，再利用讀取後關節點，分別將每隻手指的 x,y 座標經過運算識別手指是否有彎曲，例如判斷食指彎曲，如圖 2，先讀取 Point 8 及 Point 6 的座標點，再利用 y 軸數值相減($y_8 - y_6$)，若相減後數值大於 0，則判斷手指為彎曲狀態，反之小於 0 為張開狀態。

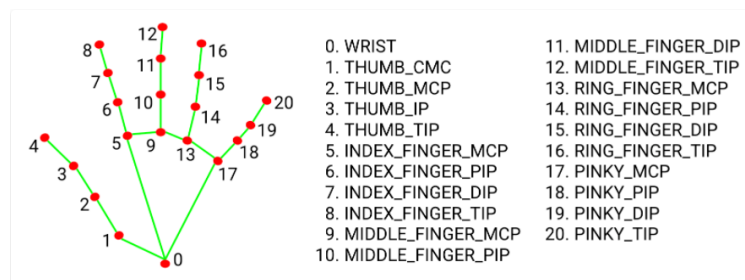


圖.1 21 點關節點

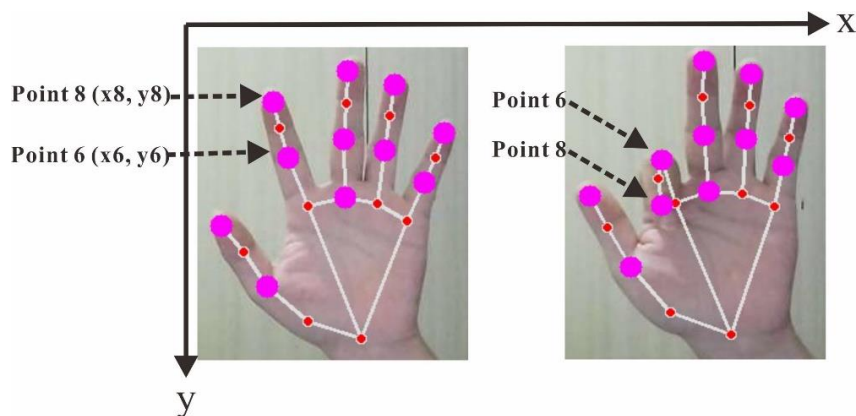


圖.2 判斷手指彎曲

二、機器手及主控制平台

機器手控制平台的核心是使用 HT66F2390 8-bit 微控器，機器手手指頭的部分是使用五顆 SG90 伺服馬達來進行控制抓取，另外還使用一顆 MG995 伺服馬達控制手臂的抬升，以及一顆步進馬達控制滑軌左右移動，六顆伺服馬達都是以 PMW 方式進行控制，步進馬達也是由 PWM 訊號搭配 M542C 步進馬達驅動模組進行控制。

機器手控制平台與電腦之間是利用藍牙無線通訊去傳送各個馬達的角度數值，所以在控制平台這邊還加上一顆 HC-06 藍牙模組，其與微控器之間是使用 UART 連接。

二、人臉辨識登入系統

人臉辨識登入系統使用的人臉辨識作法是利用 Dlib 程式庫的 68 點人臉特徵點檢測，在人臉檢測的基礎上進行，對人類臉上的特徵點，如圖 3 所示，利用檢測後的 68 點特徵點透過點與點之間的向量距離計算出 128 維的特徵向量，並使用歐式距離來判斷其相似性，並設定一個閾值進而判斷是否為同一個人。



圖.3 68 點特徵點

人臉辨識登入系統內包含新增使用者、查看使用權人員及刪除使用者資料三種功能，前兩者的功能需由使用權限為管理員方可使用。新增使用者須先填寫使用者帳戶名、密碼及擷取使用者人臉照片，註冊後會將資料上傳至 Firebase 雲端資料庫儲存。

參、作品架構

一、硬體架構

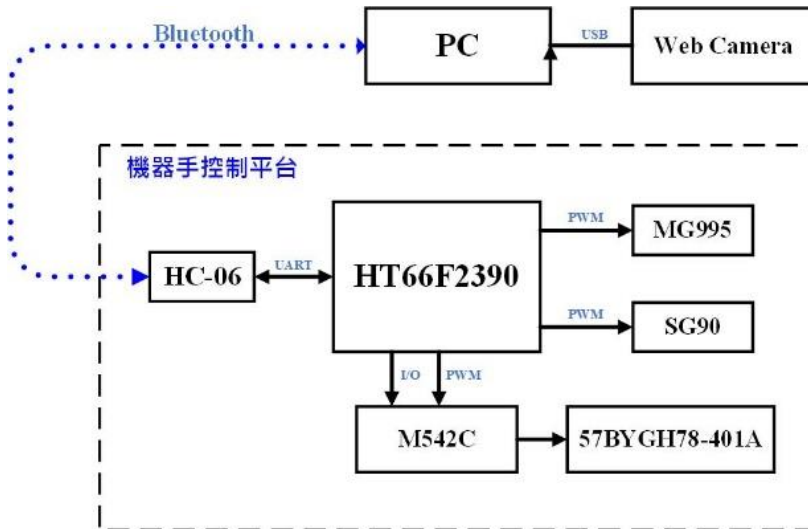


圖.4 硬體架構

二、軟體架構

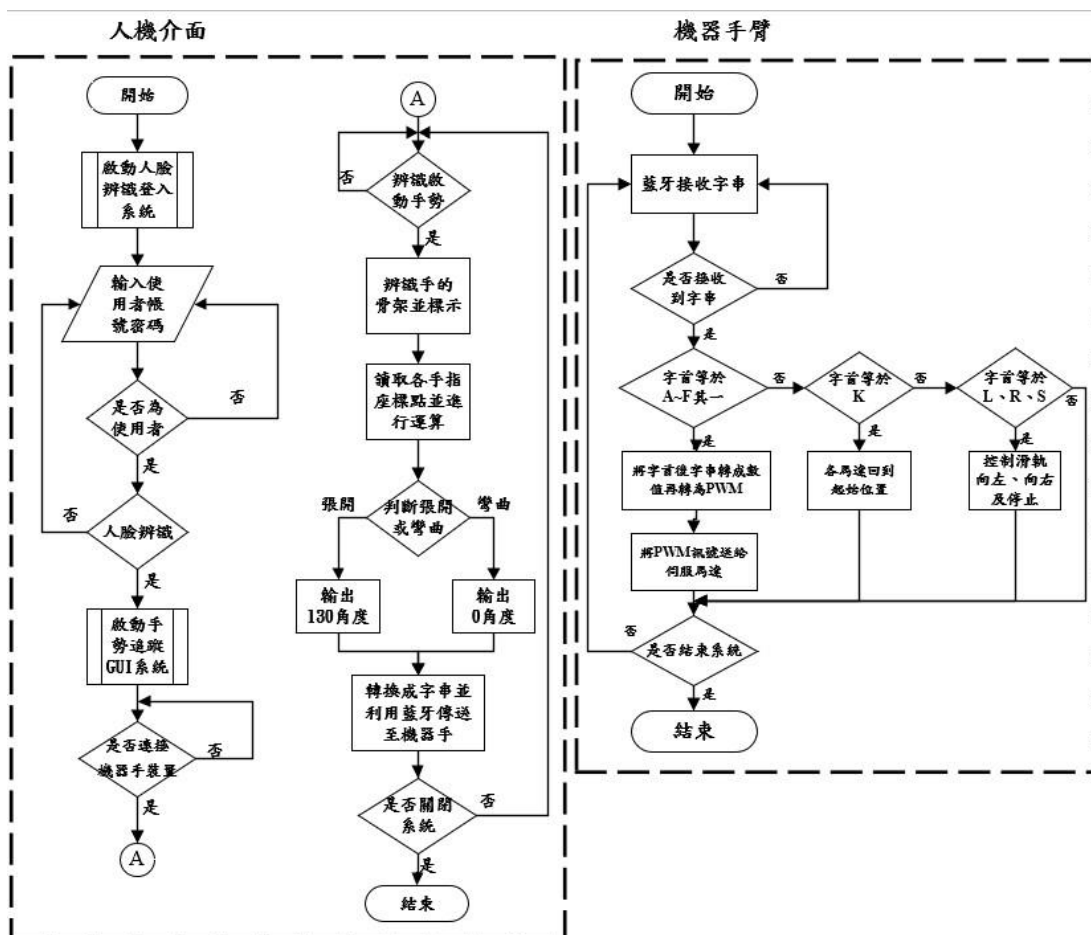
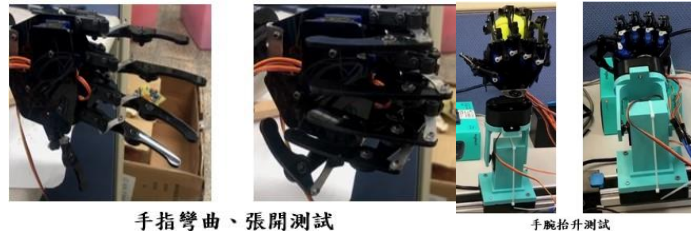


圖.5 軟體架構

肆、測試方法

一、控制板與機器手

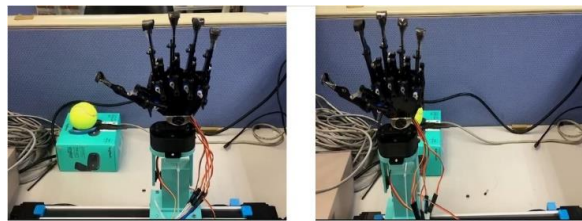
測試機器手控制板是使用 HT66F2390 8-bit 微控器進行控制，測試項目分別為：五根手指的彎曲動作、手腕的抬升及底座滑軌左右移動，如圖 6、7。



手指彎曲、張開測試

手腕抬升測試

圖.6 手指測試及手腕測試



滑軌左右移動測試

圖.7 滑軌測試

二、控制板與電腦藍牙連接

利用 HT66F2390 8-bit 微控器搭配 HC-06 藍牙模組與電腦藍牙連接，HC-06 與微控器之間是由 UART 進行連接。在 Baudrate 為 9600 時，由電腦端傳送特定字元或字串給微控器端，接收到資料後比對預先設定的字元，並依照各字元進行不同的動作，如圖 8。

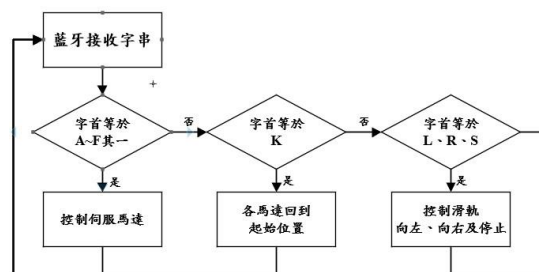


圖.8 藍牙資料傳送測試

三、手勢追蹤系統

手勢追蹤系統是利用 Google MediaPipe Hand Tracking 搭配上 OpenCV 進行處理，當影像偵測到人手會先辨識手掌位置，利用 Hand Landmark 模型，標出手的 21 點關節點，顯示在人機介面上並同步改變介面旁的拉條，如圖 9、10。

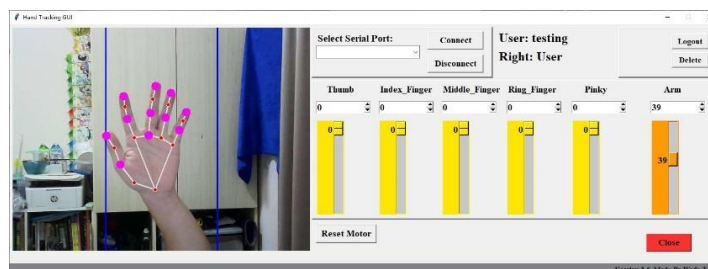


圖.9 手勢追蹤系統(張開)



圖.10 手勢追蹤系統(彎曲)

四、人臉辨識登入系統

啟動人臉辨識登入系統，須先填寫使用者帳戶及密碼，填寫完後會與雲端資料庫進行比對是否擁有使用權限，若有使用權限則會利用人臉辨識進行二次確認，辨識確認後才進入到手勢追蹤系統，如圖 11。

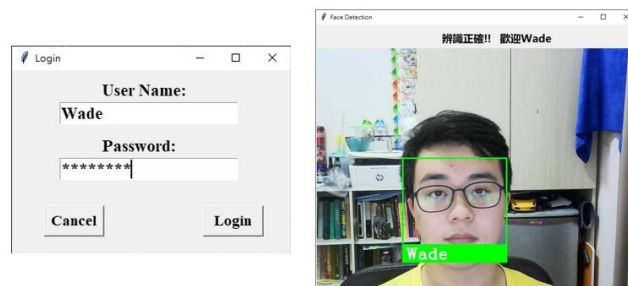


圖.11 人臉辨識系統

五、雲端資料上傳、刪除及查詢

Firebase 雲端資料庫可將剛新增使用者的帳戶名稱、密碼及使用權限上傳至雲端儲存，如圖 12，若要刪除使用者可透過人機介面進行刪除，而查詢功能與新增用戶只有系統管理者可進行，如圖 13。

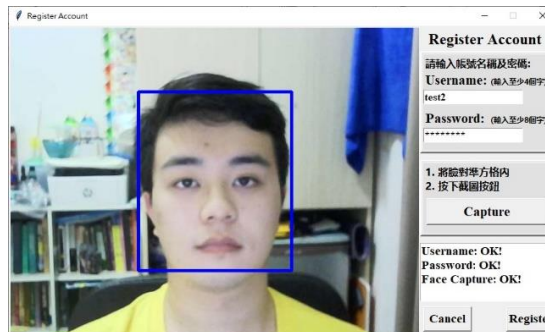
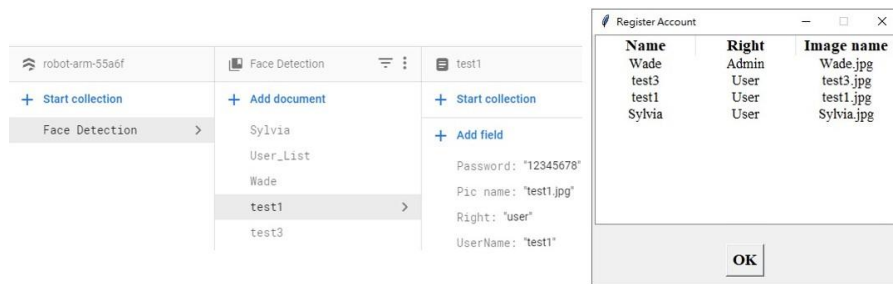


圖.12 新增使用者介面



Firebase雲端資料庫

人機介面查詢資料庫名單

圖.13 使用者資料查詢

伍、參考資料

- [1] GOOGLE LLC, MediaPipe Hands, 2020
- [2] Yanwei Liu, Python 影像辨識筆記(四):使用 dlib 辨識器, 2019
- [3] dlib 人臉特徵點檢測, 2019
- [4] 基於深度學習的人臉識別綜述, 2018
- [5] 張銘, 使用深度學習進行人臉辨識: Triplet loss, Large margin loss(ArcFace)
- [6] Face Detection - OpenCV, Dlib and Deep Learning(C++/Python), 2018
- [7] python dlib 學習(五):比對人臉, 2019
- [8] Python dlib: face landmarks detection, 2021
- [9] 人臉辨識系統 Face Recognition 開發紀錄 (OpenCV / Dlib), 2020
- [10] Face detection with OpenCV and deep learning, 2018
- [11] OpenCV Face Recognition, 2018
- [12] Deep Learning with OpenCV, 2017