

中華民國第 61 屆中小學科學展覽會 作品說明書

高級中等學校組 電腦與資訊學科

佳作

團隊合作獎

052509

最堅強的醫護人員—防疫機器人

學校名稱：臺中市華盛頓高級中學

作者： 高二 廖好捷 高二 劉柏鋒 高二 蔡承廷	指導老師： 張賢榮 許瑞中
-----------------------------------	---------------------

關鍵詞：防疫、醫療機器人、臉部辨識

摘要

本研究致力於改善在疫情底下的醫療場所分流機制，減輕第一線醫療人員的人力負擔。利用 Arduino 模組與臉部口罩辨識技術，串連「口罩辨識、體溫量測、酒精消毒」等三項主要關卡，最後製作出一個集所有功能於一身的「防疫機器人」。

防疫機器人將設置於醫療場所入口，在語音提示下，進行口罩辨識、酒精噴灑與溫度感測，如有突發狀況可與保全人員聯繫。待所有檢測完成後，溫度超標和正常的人會以不同方式進入醫院，降低交叉感染的風險，全面性的守護第一線的人員。

本研究成果能為醫療場所的分流達到「降低人力需求」及「安全」的目標，甚至可以拓展至其他同樣需要分流機制的場域。

壹、研究動機

新冠肺炎(COVID-19)自 2019 年爆發以來，造成許多人草木皆兵，人人皆害怕自己成為被感染的對象。發生之初，除了口罩的配給問題外，風險最高的便是那群守在最前線，不斷為民眾的健康把關的醫護人員。作為疫情爆發後的第一道防線，他們需要大量且頻繁的接觸病患，是染上新冠肺炎的高風險群。在 2021 年 5 月，台灣爆發新冠肺炎(COVID-19)的社區感染，大量確診民眾湧入醫院進行治療，不但對醫療場域產生威脅，也讓院內感染的事件逐漸增加。面對這個看不見的敵人，我們都無法保證防護的完善性與縝密度是否足以維護醫護人員。

在這次全球災難中，備受大家關注的便是如無人超商、服務機器人……等，以 AI 人工智慧取代傳統人力的運作模式，避免人與人近距離接觸的新興產業應運而生。面對這樣的情況，我們也開始思考，是否能夠改變居於首要防線的抗疫人力資源，以「機器人」或其他系統裝置取代人工，大幅降低醫護人員感染病毒的風險？如今，我們可以利用紅外線測量體溫，網路的革新使傳送影像更加快速，諸多的人工智慧結合，便能以更安全且更快速的解決我們生活上的問題。因此我們利用心智圖進行腦力激盪，設想出「防疫機器人」應具備的功能（如下圖一），並利用 Arduino 的概念，將想法模型化，根據實驗結果證明我們的想法是否可能執行。我們試圖設計一個可以代替第一線醫療人員的機器人，在醫院參與防疫工作，並結合體溫量測、口罩辨識、消毒與人流控管…等功能於一身，這不只是保護醫護人員，亦是守護其他未感染的民眾，更希望能透過本項研究與概念，協助世人一起度過這次疫情難關。



▲圖一：防疫機器人腦力激盪心智圖（資料來源：研究者自製）

貳、研究目的

一、研究防疫機器人的運作概念與可行性

- (一) 設計防疫機器人運作架構與流程
- (二) 測試與調校防疫機器人運作機構
- (三) 建構 Google 表單…等數據蒐集之處理

二、探討口罩辨識的技術與實際場域的影像配戴口罩技術

- (一) 了解判斷是否有配戴口罩的參數
- (二) 設計於群眾中仍準確判斷的模式

三、利用 Arduino 模組（紅外線測溫傳感模組）設計應用程式

- (一) 紅外線感溫系統於實際場域的溫度自動量測裝置探討
- (二) 紅外線感溫系統回傳數據與分流判斷之探討

四、利用 Arduino 模組（紅外線循跡感測器模組）啟動酒精消毒／口罩遞送

- (一) 能否正確偵測到人的手部

五、測試 WiFi 遙控車的靈敏度以及是否能換手之功能






- (一) 能否由後台控制人員精準控制

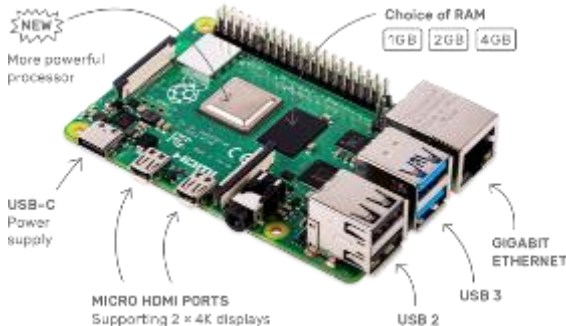
六、結合防疫機器人並搭配 SOP 流程

- (一) 標準流程是否完整無誤
- (二) 各部分的硬體結構是否能相互配合

參、研究設備及器材

一、主要材料：

名稱	圖片	用途	數量
Arduino UNO	 <p>圖二：Arduino UNO (資料來源：研究者自行拍攝)</p>	主控版，連接周邊電路，以及最主要的程式整合位置	1
非接觸式紅外線測溫傳感器模塊	 <p>圖三：MLX90614 紅外線測溫傳感器模塊 (資料來源：研究者自行拍攝)</p>	利用紅外線進行遠距的體溫測量，並傳遞數據	1
TCRT5000 紅外線循跡感測器模組	 <p>圖四：TCRT500 紅外線循跡感測器模組 (資料來源：研究者自行拍攝)</p>	利用紅外線偵測前方是否有物體	2
HC-SR04 超音波測距模組	 <p>圖五：HC-SR04 超音波測距模組 (資料來源：研究者自行拍攝)</p>	利用超音波偵測前方是否有物體	1
ISD1820 錄、放音語音模組	 <p>圖六：ISD1820 錄、放音語音模組 (資料來源：研究者自行拍攝)</p>	錄音及放音，可以搭配其他模組並適時播放語音訊息	

<p>樹莓派 4 代 Raspberry Pi 4</p>	 <p>圖七：樹莓派 4 代 Raspberry Pi 4 (資料來源：台灣物聯科技)</p>	<p>作為進行口罩辨識的微型主機，連接鏡頭，並處理數據傳給後台</p>	<p>1</p>
-----------------------------------	---	-------------------------------------	----------

表一：主要使用材料列表

二、其他材料：壓克力板、筆電、程式語言（Arduino、Python）、Google Colab、電池、行動電源、安卓手機、Arduino 相關用品（麵包板、杜邦線、傳輸線、LED 燈、電阻…等）、喇叭(0.5W)。

肆、研究過程與方法

一、醫療機器人介紹

「對於未來醫療來說，醫療機器人將會被應用到更多的醫療場景中，並將成為改變醫療商業模式的新動力。」(王豫，2020)。「醫療機器人」將成為未來醫衛系統中不可或缺的力量，不但能減輕醫療人力的負擔，也能避免與病患近距離接觸的機會，降低各種感染的可能與風險。

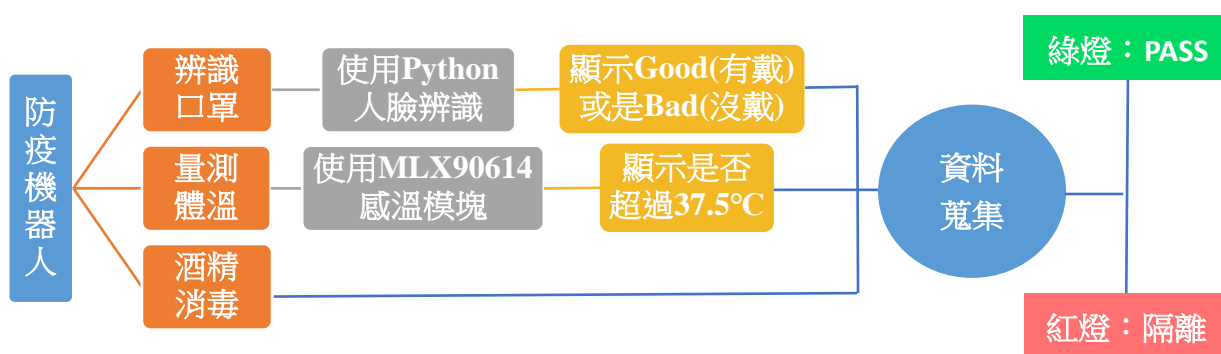
而根據國際機器人聯盟(International Federation of Robotics；IFR)的定義：「醫療機器人能夠編制操作計畫，並依據實際情況確定動作程序，然後把動作轉換成操作過程的機械運動。」(楊正瑀，2017) 而其中醫療機器人又分成四種，「包括：診斷系統用途機器人、外科手術輔助機器人與放射線治療機器人、復健系統用途機器人、其他類型醫療機器人。」(陳賜賢，2015)，本項裝置研究的「防疫機器人」，功能較類似其他類型醫療機器人中的「遠端臨場醫療機器人」。雖然沒有進行遠距治療行為，卻可以在遠端監控進入醫院場域的人員，改善行政工作流程並透過初步診斷，發揮「防疫機器人」的用途。

二、防疫機器人運作概念與流程：

本項研究主要的目標是要串連目前可行的防護措施，並且利用相關人工智慧方式，

發展出一個有效的「防疫機器人」，可以代替第一線的醫療人員，也能有效杜絕相關病毒直接或間接感染，使病毒的侵略於第一時間即被阻擋在外，提高國人安全。

因此，我們將防疫機器人的設計著重在負責醫院入口處進行防疫與管制，降低醫療人員與病患之間接觸所造成的風險。與一般單一功能的醫療機器人不同的是，本研究的醫療機器人集結三項功能於一身：透過「口罩有無」、「人流管制」、「溫度高低」.....等層層的把關，將高風險的人隔離，讓無風險的人跟隨著機器人的指引進入醫院。從一開始進入醫院人員的篩檢，到進入醫院後的路線規劃，我們希望可以完成一整套任務，並且透過一次次的修正來讓任務執行更加精準。如此，不但可以找出具有潛在風險的患者，也可以保護醫護人員與其他院內病患的安全，更能使人潮分流，降低群聚感染的機率。另外，我們也利用 google 表單來蒐集發燒病人的防疫相關資訊，並將所有的溫度資料上傳至雲端，不但希望可以利用資料來統計入院者的數據，也可以透過表單來加速發燒病患的症狀蒐集，以利後續的隔離與治療。



▲圖八：研究架構（資料來源：研究者自製）

（一）配套措施

為了讓防疫機器人的效益達到最大值，在執行任務的過程中我們也找到了施行任務可能的缺點，並且加以修正：如果出現不願配合防疫措施的人，像是不願配戴口罩、不願意被消毒、量體溫，機器人便會連線至保全中心請專業的保全來處理，並且要求他配合防疫措施。但如果進入醫院者並未佩戴口罩，機器人將會引導病人走另一條走道，到販賣部進行購買。

(二) 回復機制

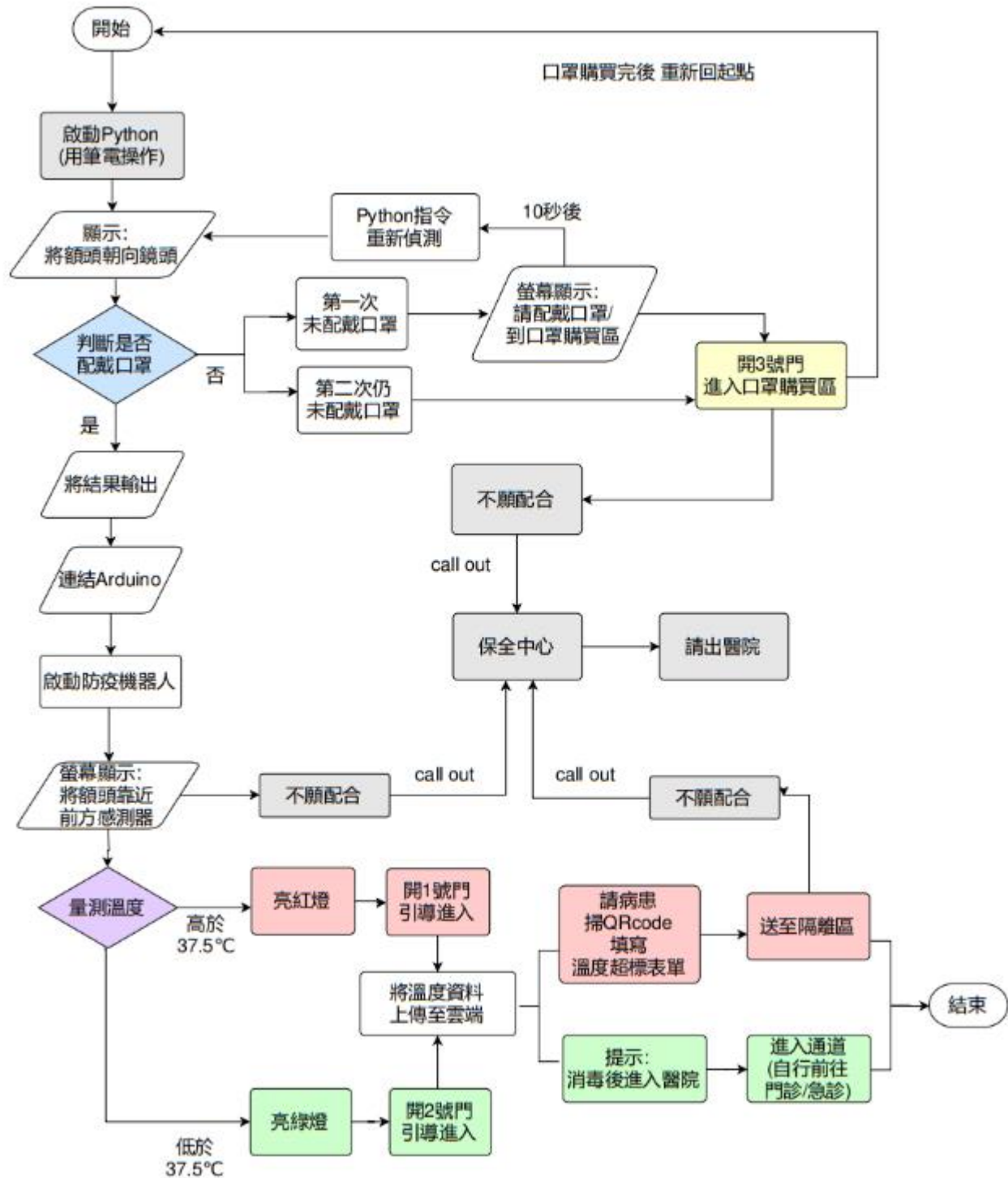
為了讓機器人的任務可以不斷循環，因此我們也在任務結束後進行相關的回復措施。因為我們的防疫機器人需要兼顧消毒的工作，但機器人能盛裝的消毒液也有限，因此我們在機器人底下加裝 Wifi 遙控車，可以從遠端的控制台進行操控，不但提高它的機動性，也可以讓新的防疫機器人替換值勤中的防疫機器人，讓原本執勤的機器人在任務後進行清潔與消毒液的補充。

(三) 標準作業流程(SOP)

我們最主要的目標是要減少醫護人員在防疫層面的人力，用機器人的方式來達成人流管制的任務，因此我們也致力於將整個防疫機器人的作業流程標準化，不只是擁有單一功能，而是能完成所有原本由醫護人員進行的各項檢查。因此，我們訂定完整的 SOP(標準程序流程)，讓機器人可以有順序性的進行每項任務，而透過我們對於各步驟的排序，更是希望可以將防疫的效率達到最高，也將提高機器人本身的功能性。

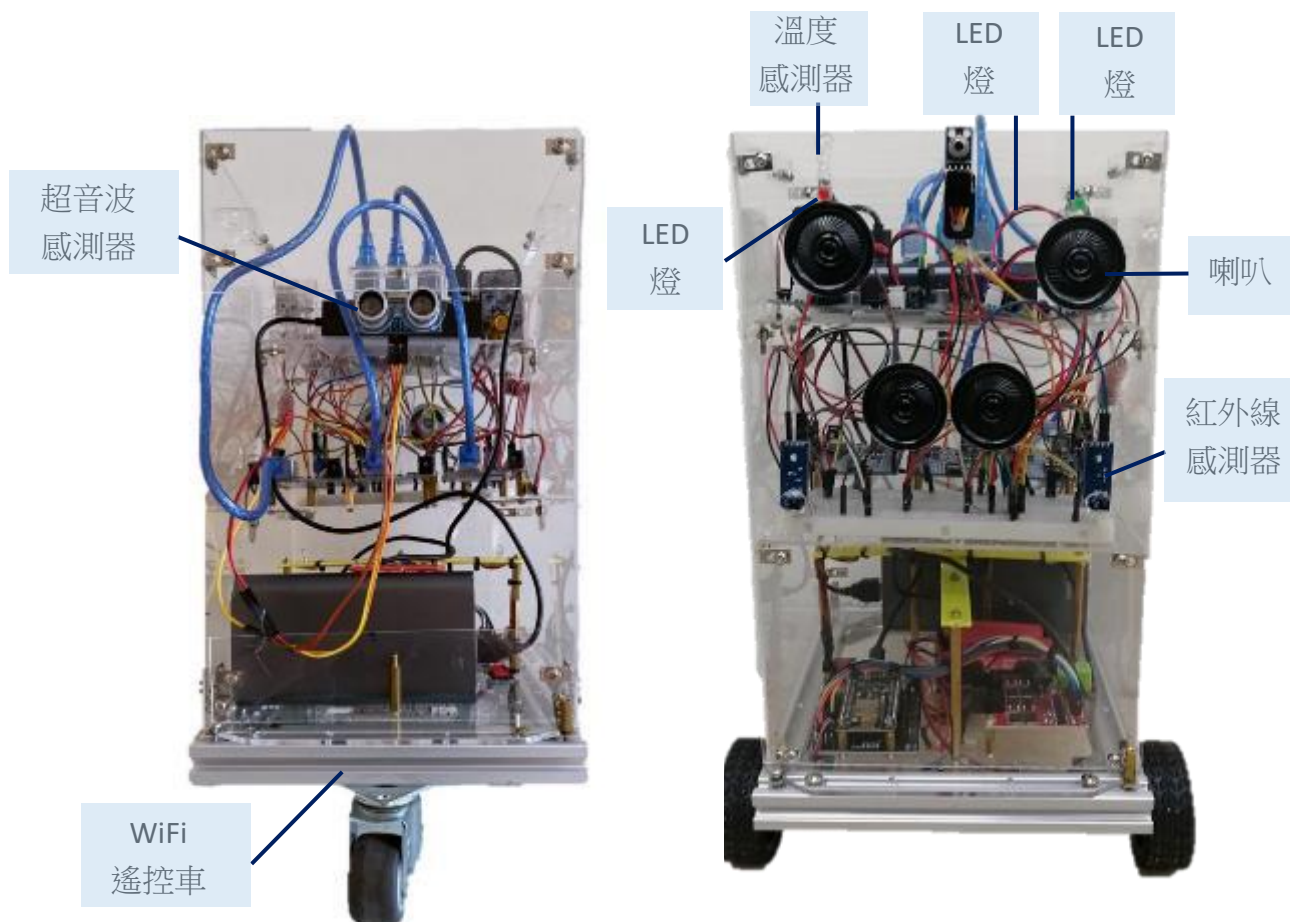
1. 第一步：筆記型電腦的螢幕會顯示提醒，請入院者將臉部指向拍照的鏡頭，以進行口罩的辨識。
2. 第二步：電腦開始判斷面前的病人是否有配戴口罩，若有便會引導入院者進入防疫走道，進行下一步的體溫測量以及消毒……等，但若沒有配戴口罩則會先提醒入院者配戴口罩並重新辨識；若入院者沒有攜帶口罩，則會引導病人從 3 號門進入口罩購買區購買口罩，再重新排隊、重新辨識口罩。
3. 第三步：螢幕提示入院者將額頭靠近紅外線體溫感測器以測量體溫。
4. 第四步：若測量出的體溫經判斷高於 37.5°C，機器人身後通往隔離區的 1 號門會開啟，並利用語音要求病人即刻前往隔離所進行隔離，並且在通道後填寫溫度超標表單，將症狀及旅遊史傳至雲端，以利後續的篩檢；若溫度在正常範圍內，則通往醫院的 2 號門會開啟。(在我們目前的設計中，將以 LED 燈展示，高於 37.5°C 以紅燈表示，溫度在正常範圍內則以綠燈表示)。
5. 第五步：在機器人身上我們會配置自動的酒精噴灑器，但我們對於消毒與否並不強求，而是自由讓病人使用。
6. 備註：

- (1) 在五個步驟中，若病人沒有配合或不清楚指示，則機器人會立刻通知警衛或保全人員，請他們親自透過語音系統溝通或前往現場處理。
- (2) 當完成溫度感測，所有溫度資料將會回傳至雲端，以便於資料的分析。
- (3) 須逐步完成每個步驟。檢查通過或完成指定動作才可進入下一個步驟



▲圖九：防疫機器人任務流程圖（資料來源：研究者自製）

三、作品成果



▲圖十：防疫機器人正面
(來源：研究者自行拍攝)

▲圖十一：防疫機器人背面
(來源：研究者自行拍攝)

(一) 正面：

1. 超音波感測器：用於感測是否有人接近機器人，當距離夠接近，便會開始啟動後續的動作

(二) 背面：

1. 紅外線感溫模組：測量進入醫院者的體溫。我們預先設定的標準為 37.5°C ，當偵測到的體溫大於或等於 37.5°C 時會亮起紅燈，開啟 1 號門並引導入院者進入隔離區，若是偵測到的體溫高於 33°C 且低於 37.5°C 時會亮起綠燈，開啟 2 號門並引導入院者通過走道進入醫院中，並可進行酒精消毒；如果偵測到的溫度低於 30°C ，則會亮起黃燈，代表量測到的只有空氣，因此不會開任何一道門，確保開門在入院者尚未被偵測時處於關閉的狀態。
2. 酒精噴灑器：當進入醫院者的手部接近紅外線感測模組時，酒精噴灑器便會啟動進行噴灑消毒

3. 喇叭：過程中的每一個步驟，機器人都會配合語音系統進行提醒。當超音波感測器感測到有人靠近時，語音系統會播放：「請走防疫路線。」；當溫度過高而亮起紅燈時，語音系統會播放：「溫度過高，請走一號門。」；當溫度正常而亮起綠燈時，語音系統會播放：「溫度正常，請走二號門。」；當左側的紅外線感測模組感測到手靠近時，語音系統會播放：「請取口罩。」；當右側的紅外線感測模組感測到手靠近時，語音系統會播放：「請使用酒精。」
4. WiFi 遙控車：提供醫療機器人必要的機動性，讓後臺的監控人員可以在必要情況時遠端操控機器人，也達到我們對於防疫保持距離的要求

四、數據收集

我們不只是希望可以利用完整的 SOP 流程執行醫院分流的任務，我們也希望可以將這些量測的資訊上傳至雲端，建構一個完整數據平台，蒐集所以防疫的數據，讓醫院內部掌握所有進出的人的健康狀況。

1. 溫度超標者：填寫表單

我們會在溫度超標得入院者進入隔離區前，先掃 QRcode 條碼（如下圖），填寫相關的防疫及實名制的資料，其中包括姓名、電話、異常溫度、旅遊史、接觸史、其他相關症狀、其他應註明的事項…等，不但希望可以在進入隔離前就先做好所有資料的收集，以利後續篩檢及治療的工作，也希望將這些資料上傳至雲端，讓疫調與其他健康管理的工作可以執行地更加快速。

2. 入院者：溫度資料回傳至雲端

當有戴好口罩、進入防疫路線並量測體溫者，我們都會利用程式回傳且紀錄，將所有的溫度結果用資訊的方式直接上傳至雲端，進行後續的數據分析。



圖十二：溫度超標表單 QRcode（資料來源：研究者自行製作）

防疫相關資訊

由於您的體溫超過正常標準，而目前國內新冠病毒肺炎疫情仍盛行，因此我們希望您能藉由這份表單，了解您的相關資訊，以便醫護人員可以為您作詳細的治療或相關追蹤。

*必填

姓名*

您的回答 _____

連絡電話*

您的回答 _____

籍籍*

您的回答 _____

過去14天是否曾出國*

是
 否

過去14天是否曾因身體不適而前往診所及醫院*

是
 否

同住親友是否有人收到居家隔離的通知*

是
 否

過去14天內是否曾有咳嗽、喉痛、流鼻水的症狀*

是
 否

過去14天內是否曾有嗅覺或味覺喪失的症狀*

是
 否

過去14天是否有發熱的症狀*

是
 否

過去14天是否有肌肉痠痛或四肢無力的症狀*

是
 否

提交

圖十三：溫度超標表單（資料來源：研究者自行製作）

伍、研究結果

一、分區組裝結構簡介：

名稱	圖片	用途	數量
WiFi 遙控車	 <p>圖十四：WiFi 遙控車 (資料來源：研究者自行拍攝)</p>	透過 WiFi 連接，本體是遙控車，我們可以遠端透過連接對應的 IP 位置來遙控車子，同時為承載整體系統的車體，光是目前小型的機身就可以承受相對巨大的重量，未來若放大機身，也可承載更多醫療器具。	1
紅外線 循跡系統	 <p>圖十五：紅外線循跡系統 (資料來源：研究者自行拍攝)</p>	目前以三組為一個單位製作而成的 IR，可以在偵測到有人靠近一定距離，且同時觸發三個紅外線循跡模組才會啟動整個系統，可以有效避免系統的誤觸率。	1
紅外線 感溫系統	 <p>圖十六：紅外線感溫系統 (資料來源：研究者自行拍攝)</p>	在程式設計中，我們利用紅色和綠色的 LED 燈，模擬在測量完病人的體溫之後，高於或低於 37.5 度分別的分流路徑。若大於或等於 37.5 度，則利用專屬的道路進入隔離區，以減少和其他人的接觸。若溫度介於 33 度至 37 度，則走由綠色燈代表的路徑，直接進入醫院內。若 30 度以下，則為空閒狀態。	1

表二：組裝完成結構

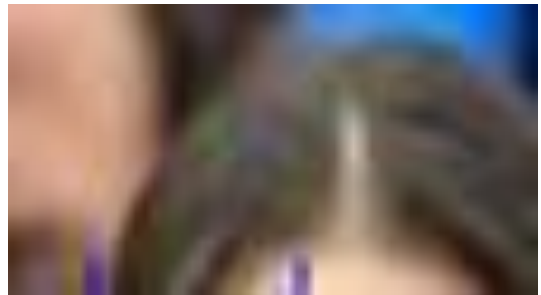
二、軟體(口罩辨識系統)

(一)利用 Google Colab 進行初步測試

在程式的撰寫中，我們希望設計出一套具有辨識口罩的功能，並能夠加以判斷鏡頭下的人是否有配戴口罩的系統，於是我們首先從類神經網路方面開始進行平台層面的測試，希望能證明口罩辨識的可行性。我們使用 Google Colab 作為我們的平台，試著從圖片中找出人臉，找到人臉特徵後進行分析。而我們所希望的結果，是這項功能可以將有戴口罩的人與沒有戴口罩的人做出區別，以作為我們的防疫機器人在檢查口罩時最重要的功能。而在我們完成程式的編寫後，我們便開始讓 AI 進行大量的訓練，以提高它的準確度。(在表三中的圖十七至圖二十二，因為隱私權的問題，故已將照片中人臉使用馬賽克處理。)



圖十七 (資料來源：
<https://www.cna.com.tw/news/firstnews/202003010226.aspx>)



圖十八 (資料來源：
<https://lvcnn.com/cn/news.php?id=29767>)



圖十九 (資料來源：
<https://dailyview.tw/Popular/Detail/3283>)



圖二十 (資料來源：
<https://www.businesstoday.com.tw/article/category/183029/post/202101290028/>)

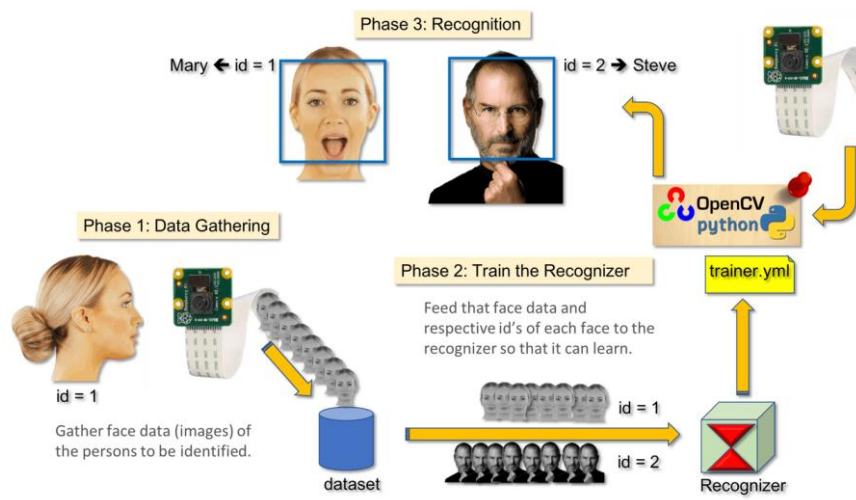
 <p>圖二十一（資料來源： https://ctee.com.tw/amp/news/policy/212388.html）</p>	 <p>圖二十二（資料來源： https://ctee.com.tw/amp/news/policy/212388.html）</p>
--	---

表三：口罩辨識測試【照片中人臉皆已用馬賽克處理】（資料來源：研究者自製）

由圖十六及圖十九可以看出我們所設計的防疫機器人，可以辨別鏡頭前的人是否有戴口罩；而在圖十五及圖二十的測試中，顯示即使有數人同時在鏡頭前，也能捕捉到照片中每個人的臉部及口罩。在圖十五的測試中，如果是距離鏡頭較遠且未將臉部朝向鏡頭的照片，我們的辨識系統便不會將其納入檢查範圍。

(二)使用 Raspberry Pi（樹莓派）

為了讓程式的運作效率提升，也考量到我們要將程式運用在實體的部分，我們開始尋找其他相較於利用 Google Colab 更為簡便的方式，其中，我們查了許多有關樹莓派的資料，也實際嘗試過使用樹莓派來進行口罩的辨識：



圖二十三：人臉辨識過程（資料來源：<https://www.hackster.io/mjrobot/real-time-face-recognition-an-end-to-end-project-a10826#code>）

根據圖三十，是運用 Raspberry Pi 來進行人臉的辨識，我們在分析完過程之後，將整個程序主要分為三個階段：

1. 偵測臉部和資料收集
2. 訓練偵測器
3. 臉部辨識

我們蒐集了許多使用樹莓派進行人臉辨識的方法，皆是透過以上這三個階段進行。在我們將鏡頭與樹莓派連接之後，便會將程式庫寫入系統當中，接著我們會測試相機，以確保鏡頭與我們寫入的程式沒有衝突。在這裡我們使用的方法，是利用哈爾特徵，讓接下來所有的影像簡單化，讓 AI 可以更快速有效的辨識圖形。

接著是最重要的訓練過程，在這個階段，我們需要大量的素材讓 AI 進行影像的辨別，而我們將影像分為陰性（無配戴口罩）和陽性（有配戴口罩）兩種。並輸入各類圖片，讓 AI 針對每張照片進行判斷，隨著我們輸入的照片漸多，其在辨識的精準度上也漸漸提升。

訓練完成後，最後的步驟是使用 GUI 介面標記臉部。在這一步我們需要做的，是去設定矩形大小的參數並以藍色邊框的方式呈現。之後我們設計程式讓 AI 可以在找到人臉之後，自動標定出長寬值，而且矩形可以隨著鏡頭捕捉到的影像移動，讓 AI 在偵測過程中，即使病人移動，也不會中斷辨識。

(三)使用 Python 技術

因為考量到口罩辨識程式的運算量可能過於龐大，導致樹莓派的 CPU 過熱，我們考慮了第三種較為簡便的方法：利用筆電進行影像處理。在筆電端我們利用 Pyserial 來溝通 Python 和 Arduino，在事前我們會先安裝 Anaconda 來進行 debug，並且安裝 OpenCV 的套件。

在筆電端的優點是我們可以利用筆電內建的鏡頭來進行口罩辨識，若是運用筆電也可以使用我們較熟悉的資料夾存放模式，而且在捕捉到影像之後，我們可以直接將圖片存放到 Google 雲端硬碟，就不必再多撰寫程式讓系統的負荷量增加，且若是未來需要維護時也較為簡便。

樹莓派	筆電
適合自動化，較差的環境	一般室內
有 GPIO，適用多數感測或輸出模組	若有特殊功能／模組，需要串接開發板
要外接攝影機	維護介面較熟悉
運算量太大要考慮散熱	運算量大時，可以選較好的筆電

表四：口罩辨識技術種類比較（資料來源：研究者自製）

1. 自動拍照程式部分

```
#!/usr/bin/env python
# coding: utf-8

# In[13]:

#設定定時拍照
import cv2
import time
from datetime import datetime

#now = datetime.now()
cv2.namedWindow("frame")#影像視窗

for i in range(2):
#拍2張照片

    current_time = datetime.today().strftime('%Y-%m-%d-%H:%M:%S')
    #print("Current Time =", current_time)

    cap = cv2.VideoCapture(0)

    while(cap.isOpened()):
        time.sleep(3)
        #每張間隔3秒
        ret,img=cap.read()
        cv2.imshow("frame",img)
        cv2.imwrite("WHSRecord/"+current_time+"_"+str(i)+"_"+Mask.jpg", img)
        #PRecord為資料夾名稱，記得加斜線
        break
    cap.release()
    cv2.destroyAllWindows()
    cv2.waitKey(1)
    cv2.waitKey(1)
```

2. 偵測口罩程式部分

```
#!/usr/bin/env python
# coding: utf-8

# In[5]:

import cv2
import numpy as np
from matplotlib import pyplot as plt

net = cv2.dnn.readNetFromDarknet("yolov3-tiny-obj.cfg","yolov3-tiny-obj_10000.weights")
#載入模型，可更改權重檔名
output_layers = [layer_names[i] - 1 for i in net.getUnconnectedOutLayers()]
classes = [line.strip() for line in open("obj.names")]#讀類別
colors = [(0,0,255), (255,0,0), (0,255,0)]

cv2.namedWindow("frame")
cap = cv2.VideoCapture(0)#0表示內建攝影機，數字1為外接式
```

```
if i in indexes:
    x, y, w, h = boxes[i]
    label = str(classes[class_ids[i]])
    color = colors[class_ids[i]]
    cv2.rectangle(img, (x, y), (x + w, y + h), color, 1)
    cv2.putText(img, label, (x, y - 5), font, 1, color, 1)
    print(label)#將類別結果輸出
plt.rcParams['figure.figsize'] = [15, 10]
img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

plt.imshow(img_rgb)
```

3. Pyserial 程式部分

```

#!/usr/bin/env python
# coding: utf-8

# In[ ]:

#https://swf.com.tw/?p=1188

import serial
from time import sleep
import sys

COM_PORT = 'COM21' #請自行修改序列埠名稱
BAUD_RATES = 9600
ser = serial.Serial(COM_PORT, BAUD_RATES)

try:
    while True:
        # 接收用戶的輸入值並轉成小寫
        choice = input('按a開燈、按b關燈、按e關閉程式 ').lower()

        if choice == 'a':
            print('傳送開燈指令')
            ser.write(b'LED_ON\n') # 訊息必須是位元組類型
            sleep(0.5) # 暫停0.5秒，再執行底下接收回應訊息的迴圈
        elif choice == 'b':
            print('傳送關燈指令')
            ser.write(b'LED_OFF\n')
            sleep(0.5)
        elif choice == 'e':
            ser.close()
            print('再見！')
            sys.exit()
        else:
            print('指令錯誤...')

        while ser.in_waiting:
            mcu_feedback = ser.readline().decode() # 接收回應訊息並解碼
            print('控制板回應：', mcu_feedback)

except KeyboardInterrupt:
    ser.close()
    print('再見！')

```

4. 網路爬蟲

```

import requests
import json

import gspread
import time #待會會取時間
#import datetime
from oauth2client.service_account import ServiceAccountCredentials

```

```

def auth_gss_client(path, scopes):
    credentials = ServiceAccountCredentials.from_json_keyfile_name(path, scopes)
    return gspread.authorize(credentials)

auth_json_path = 'pexcel-6a003967f3ac.json' #由剛剛建立出的憑證, 放置相同目錄以供引入
gss_scopes = ['https://spreadsheets.google.com/feeds'] #我們想要取用的範圍
gss_client = auth_gss_client(auth_json_path, gss_scopes) #呼叫我們的函式

#從剛剛建立的sheet, 把網址中 https://docs.google.com/spreadsheets/d/{key}/edit 的 {key} 的值代入
spreadsheet_key_path = '1z_ic_0t4jkq-GiUOmZnLbx1jaKQF1HChrv3Wv-dfpps'

#我們透過open_by_key這個method來開啟sheet
wks = gss_client.open_by_key(spreadsheet_key_path)
sheet = wks.get_worksheet(0)

#sheet = wks.sheet1

#today = time.strftime("%c")

#today = datetime.date.today()
today = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime())

sheet.update_cell(5, 2, today) #更新時間

A1 = sheet.cell(9, 2) #抓值用
dirvalue = A1.value
print(dirvalue)

```

圖二十四～二十七：口罩辨識程式（資料來源：研究者自製）

這個部分我們利用到了 google cloud platform 上面的 API 服務，在這裡我們用的是 google sheets API，因為我們將利用 google 的試算表讓我們的資料可以保存至雲端，以便我們隨時取用病患資料，並且較不需擔心我們的資料存放問題。我們希望可以透過網路爬蟲的方式，透過 Python 去抓取病患填完的表單資料。之後透過抓取到的體溫、旅遊史、接觸史等等，讓我們的防疫機器人去控制要開啟的閘門，以及讓醫師可以進行初步的判斷，以便促進醫療人員的效率最大化。

三、紅外線感溫系統測試

我們將 MLX90614 模組(紅外測溫傳感器模塊)與 Arduino Uno 板串聯起來，製作出一個能夠量測溫度並且輸出數值的系統。我們以兩種實驗進行驗證：

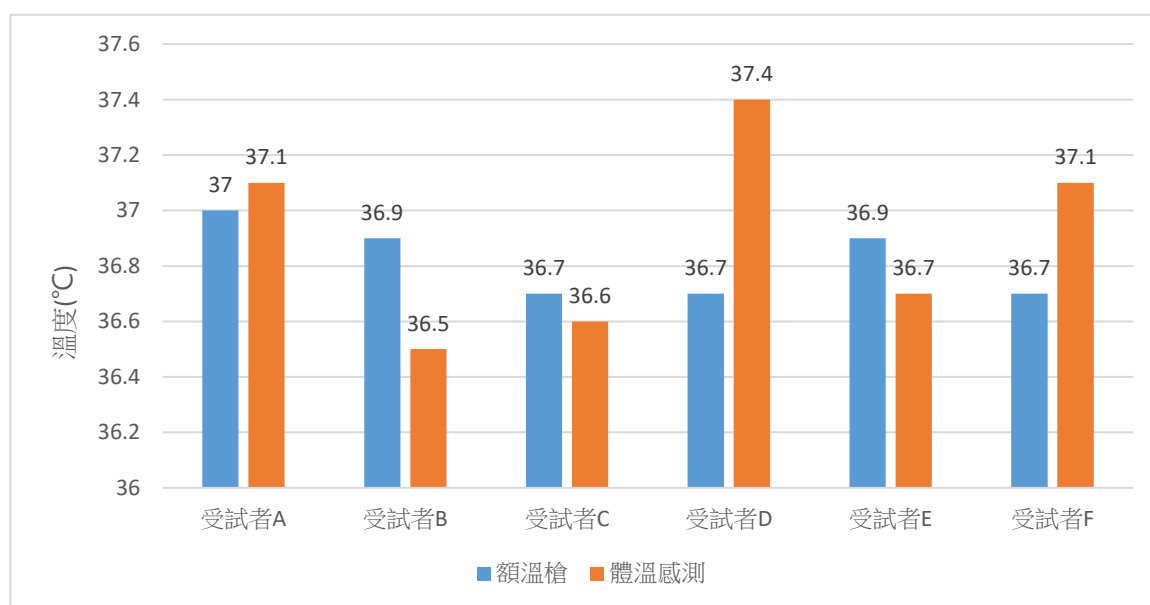
(一) 實驗一：紅外線感溫系統於實際場域的溫度自動量測裝置探討

我們利用紅外線感溫系統進行人體額溫量測，並且隨即使用校正過後的額溫槍來作為真實的溫度，作為對照組，比較紅外線感溫系統量測溫度的準確性：

測試 受試者	第一次		第二次		第三次		平均		差值
	額溫 槍	溫度 感測	額溫 槍	溫度 感測	額溫 槍	溫度 感測	額溫 槍	溫度 感測	
A	37.0	37.2	37.0	36.7	37.1	37.3	37.0	37.1	0.1
B	37.1	37.2	36.8	36.0	36.9	36.4	36.9	36.5	-0.4
C	36.8	36.6	36.6	36.4	36.8	36.9	36.7	36.6	-0.1
D	36.5	36.5	36.7	37.3	36.8	37.6	36.7	37.1	0.4
E	37.0	37.5	36.9	36.2	36.9	36.3	36.9	36.7	-0.2
F	36.7	37.0	36.8	37.7	36.6	36.5	36.7	37.1	0.4

表五：紅外線感溫系統與額溫槍之溫度比較（資料來源：研究者繪製）

（溫度單位：°C，數值四捨五入至小數點後第一位）



圖二十八：紅外線感溫系統與額溫槍之平均溫度比較（資料來源：研究者繪製）

根據實驗結果，六次實驗差值的平均為 0.2°C，可推論紅外線感溫模組仍具有一定的準確性與穩定性。雖然在某些受試者（如受試者 D）出現了一些差距，但至少證實了防疫機器人具有量測體溫的可行性。它的缺點則是在量測距離的方面不如熱像儀來的遠，只限於短距離的量測。

(二) 實驗二：紅外線感溫系統回傳數據之探討

本項裝置係利用 Arduino 開放式平台為介面，利用類 C 語言完成程式之撰寫與應用，串聯紅外線感溫模組。測量完溫度後，防疫機器人在面對體溫符合標準及過高的病人，將會分別開起兩道不同的閘門。我們預先設定的標準為 37.5°C，當偵測到的體溫大於或等於 37.5°C 時會開啟的隔離區走道，若是偵測到的體溫高於 33°C 且低於 37.5°C 時，則可以通過走道進入醫院中；如果偵測到的溫度低於 30°C，則代表量測到的只有空氣，因此不會開任何一道門，確保閘門在入院者尚未被偵測時處於關閉的狀態（在我們目前的設計中，將以 LED 燈展示，大於或等於 37.5°C 以紅燈表示，介於 33°C-37.5°C 則以綠燈表示，低於 30°C 時則以黃燈表示）。利用這種方式，能大幅減少發燒的患者和其他人的接觸，也讓駐守於隔離的醫護人員能控制人流量，更可以精準的判斷數據，並將結果上傳至雲端中。



圖二十九：紅外線感溫測試與 LED 顯示（資料來源：研究者拍攝）

由上圖實驗結果證實，此程式的邏輯並無出現錯誤，在判斷溫度也非常精準。而透過程式數值的輸出可得知，在防疫機器人身上的紅外線感溫模組十分準確，在測量時不只測量感測到的溫度，更把人體和環境的溫度分開，讓我們可以更明確的判斷病患本身的體溫是否超高。而我們也找了六位受試者進行溫度測量的實驗，而根據實驗結果，溫度感測模組對於六位受試者的溫度都能做出精確的分流判斷，而就算沒有受試者時，溫度感測模組也能確實判斷出來，並且亮出黃色燈號。

測試 受試者	第一次			第二次			第三次		
	溫度 感測	理論 顏色	實際 顏色	溫度 感測	理論 顏色	實際 顏色	溫度 感測	理論 顏色	實際 顏色
A	37.2	■	■	36.7	■	■	37.3	■	■
B	37.2	■	■	36.0	■	■	36.4	■	■
C	36.6	■	■	36.4	■	■	36.9	■	■
D	36.5	■	■	37.3	■	■	37.6	■	■
E	37.5	■	■	36.2	■	■	36.3	■	■
F	37.0	■	■	37.7	■	■	36.5	■	■
無	27.7	■	■	28.1	■	■	29.2	■	■

表六：LED 理論顏色與理論顏色之比較（資料來源：研究者繪製）

(三) 熱顯像儀

本研究小組曾考慮過使用熱顯像儀來代替紅外線感溫系統，其優點是它可以架設在一個定點，並且可以有效且精確地捕捉範圍內人體的溫度，就可以除去利用溫度計需要配合病人的身高調整高度的這個缺點。但是礙於熱顯像儀的費用過於昂貴，因此我們不將它納入考量，若未來有機會我們可以將其升級。


```

#include <Wire.h>
#include <Adafruit_MLX90614.h>

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

void setup() {
  Serial.begin(9600);
  Serial.println("Adafruit MLX90614 test");
  pinMode(8, OUTPUT); //紅燈
  pinMode(9, OUTPUT); //綠燈
  pinMode(10, OUTPUT); //黃燈
  mlx.begin();
}

void loop() {
  Serial.print("Ambient = ");
  Serial.print(mlx.readAmbientTempC());
  Serial.print(" *C\tObject = ");
  Serial.print(mlx.readObjectTempC());
  Serial.println(" *C");
  Serial.print("Ambient = ");
  Serial.print(mlx.readAmbientTempF());
  Serial.print(" *F\tObject = ");
  Serial.print(mlx.readObjectTempF());
  Serial.println(" *F");
  if(mlx.readObjectTempC() > 37.5 )
  {
    digitalWrite(8, HIGH); //紅燈亮或啟動繼電器又或聲音電路，告知來者(溫度異常，請走1號門)
  }
  else
  {
    digitalWrite(8, LOW);
  }

  if(mlx.readObjectTempC() > 33 and mlx.readObjectTempC() < 37.5)
  {
    digitalWrite(9, HIGH); //綠燈亮或啟動繼電器又或聲音電路，告知來者(溫度正常，請走2號門)
  }
  else
  {
    digitalWrite(9, LOW);
  }

  if(mlx.readObjectTempC() < 30) //當來者尚未開始量測時，黃燈表示可接受量測溫度功能
  {
    digitalWrite(10, HIGH);
  }
  else
  {
    digitalWrite(10, LOW);
  }

  Serial.println(); //將溫度感測量測之數字顯示於監控螢幕中
  delay(500);
}

```

圖三十：紅外線感溫模組相關程式碼（資料來源：研究者拍攝）

四、酒精噴灑裝置/口罩遞送裝置

另一個防疫機器人必備的功能就是噴灑酒精。我們目前以 TCRT500 模組（紅外線循跡感測器模組）作為感測裝置，組裝成一套系統，當感測到手部靠近時，便會驅動酒精噴灑，為入院者進行消毒。但這項步驟並不會被要求，而是讓病人自行決定是否要消毒。

```
//利用三個感測器同時偵測達到條件時則有三種狀態出現

int sensorPin1 = A0;      // 感測器腳位設定
int sensorPin2 = A1;
int sensorPin3 = A2;

int ledPin1 = 2;         // LED顯示設定，可設計觸發繼電器或者是聲音電路
int ledPin2 = 3;
int ledPin3 = 4;

int sensorValue1=0;     // 設定感測器的初始值為0
int sensorValue2=0;
int sensorValue3=0;

void setup ()           //設定程式
{
  pinMode (ledPin1, OUTPUT);      //設定led為輸出
  pinMode (ledPin2, OUTPUT);
  pinMode (ledPin3, OUTPUT);
}

void loop ()
{
  sensorValue1 = analogRead (sensorPin1);    //讀取感測器的值
  sensorValue2 = analogRead (sensorPin2);
  sensorValue3 = analogRead (sensorPin3);

  if (sensorValue1>800)                //若第一感測器的值大於800則LED熄滅
  {
    digitalWrite (ledPin1,LOW); //LED顯示設定，可設計讓繼電器不動作或者是聲音電路不動作
  }
  else
  {
    digitalWrite (ledPin1, HIGH); //否則第一感測器的值小於800則LED亮，或者是繼電器動作又或者是聲音電路被觸發
  }

  if (sensorValue2>800)                //若第二感測器的值大於800則LED熄滅，可設計讓繼電器2不動作或者是聲音電路2不動作
  {
    digitalWrite (ledPin2,LOW);
  }
  else
  {
    digitalWrite (ledPin2, HIGH); //否則第二感測器的值小於800則LED亮，可設計讓繼電器2動作或者是聲音電路2動作
  }

  if (sensorValue3>800)                //若第三感測器的值大於800則LED熄滅，可設計讓繼電器3動作或者是聲音電路3不動作
  {
    digitalWrite (ledPin3,LOW);
  }
  else
  {
    digitalWrite (ledPin3, HIGH); //否則第二感測器的值小於800則LED亮，可設計讓繼電器32動作或者是聲音電路3動作
  }
}
```

圖三十一：酒精噴灑裝置示意圖（資料來源：研究者拍攝）

五、WiFi 遙控車運作測試

我們將前述的作品固定在 WiFi 遙控車上，使之能夠隨著車子移動。我們使用安卓手機，在遠端透過連接對應的 IP 位置來遙控車子，可以順利進行前進、後退、左右轉的任務，在移動上具有高機動性。其載重能力也很高，光是目前小型的模型就可以承受相對巨大的重量，未來若放大機身，想必能承載更多醫療器具。



圖三十二、三十三：使用安卓手機操控機器人移動（資料來源：研究者自行拍攝）

陸、討論

- 一、在我們防疫機器人的運作流程中，我們將偵測口罩作為第一個步驟，是因為考量到目前各場所，皆被規定要確實佩戴口罩，尤其是在醫院，這類病毒聚集更密集的区域，我們認為偵測口罩會是整個流程的當務之急。而我們將溫度測量置於最後，是考量整體檢查過程的流暢度，在測量完體溫後，再判斷數據並引領病人進入隔離區或醫院，這樣的安排，便可以免除過程中過多的程序。
- 二、考量到機器人可能需要適時的移動，因此我們在機器人主體的下方加裝了 WiFi 遙控車，讓後臺的操作人員可以精準的用手機操控，而無論是在速度或是接收度上，我們設計的 WiFi 遙控車都能達到我們的要求，甚至是在載重上，我們的 WiFi 車能夠承受的重量更是大大超過我們的機器人，因此在未來精進中，我們認為也可以將簡單的醫療設備裝置的遙控車上，成為另類的機器人，在醫院中負責運送的功能。
- 三、在感測體溫的功能中，我們採用的是紅外線測溫傳感器模塊，因為在不侵入性的感測溫

中，除了額溫槍，紅外線屬於較準確也較符合我們的防疫機器人設計的方法。而在我們的程式設計及實驗證實下，我們所運用的紅外線測溫傳感器模塊和平時額溫槍所測量出的溫度僅有著微小的差距，而我們認為這並不影響測溫功能。而未來，我們更希望可以利用熱像儀代替紅外線感溫器，解決目前只能近距離測量的問題。

四、為了讓忘記配戴口罩的病人也可以保護自己與他人，我們設置了口罩購買的區域，讓沒有攜帶口罩的病人可以即時購買並配戴，保護自己也保護他人。口罩的辨識在我們的防疫機器人中扮演最重要的角色，我們利用類神經網路模擬程式、訓練並利用各類照片進行測試，除了單一的個人照，我們更嘗試加入許多人聚集的照片，就是為了讓我們的口罩辨識功能可以更敏銳，即使面對群眾依然可以準確辨識配戴口罩的有無。另外，我們也嘗試使用樹莓派及 Python 作為我們口罩辨識的一環，在經過適當的評估之後，我們認為 Python 較符合我們的能力所及，在資料的處理與操作上也較為便利，因此我們使用它做為我們口罩辨識的工具與技術。

五、Arduino 自動統計體溫超過標準以及合格的人數，以及利用 google 表單，讓體溫超過標準的病人的填下相關接觸史、旅遊史……等資料，讓醫院的醫療人員有數據可以分析每日進入醫院的人流狀況以及風險高低，以便醫院內部針對數據做出不同的應對措施。

六、我們將所有的系統以及功能串連在一起，最終完成了我們防疫機器人的模型。

七、未來展望

(一) 起初，我們想要使用熱顯像儀來代替溫度感測儀，因為我們認為它可以架設在一個定點，並且可以更有效且精確的捕捉範圍內人體的溫度，更可以去掉利用紅外線感溫系統衍生出的，需要配合病人調整高度的問題，但是礙於熱顯像儀的成本過高，因此在這次的模擬中，我們使用紅外線感溫系統代替。若未來有機會精進我們的防疫機器人，我們便希望可以利用熱顯像儀作為感溫的工具。

(二) 在進行口罩辨識的時候我們會進行 count 數的統計，我們會將畫面中的人數上限設為兩人，讓行動不便的人士有至多一名的陪同者可以一起進行辨識。在辨識完成之後，會將資料傳給閘門系統，讓閘門的 count 數設定為兩人，以利行動不便人士的感測，也加速隊伍的進行。

柒、結論

本小組設計的防疫機器人在流程的操作上已趨於完善，各類功能也在實驗的測試中得到符合理想的成果，我們透過了多次的模型模擬後，皆呈現準確且流暢的結果，結合了口罩辨識、酒精噴灑、紅外線感溫……等功能，這幾類原本需要醫護人員親自操作的檢查程序，看似簡單卻隱藏了諸多的風險，但之後，透過我們的防疫機器人，醫護人員可以避免第一線接觸病人，降低了他們感染新型冠狀病毒甚至未來其他流行性疾病的可能。而之後，我們也會繼續針對這台防疫機器人進行進一步的開發，包括增進口罩辨識的功能以及將各步驟更完整的結合，讓機器人的工作更有效率也更精確，還有加強和後端人員遠端連線的系統，彌補機器人在面對人類情緒與情感的不足。我們期許，在我們設計出更完善的防疫機器人後，可以駐紮在醫院門口，讓大家可以放心的將檢查工作交給機器人完成。

捌、參考資料及其他

- 一、蔣榮先(2020)。從 AI 到智慧醫療。台灣：商周。
- 二、王豫(2020)。醫療機器人：產業未來新革命。大陸：機械工業。
- 三、柯博文(2016)。Raspberry Pi 最佳入門與實戰應用。2016 年 4 月，台北：基峰。
- 四、Jack Creasey(2015)。Raspberry Pi 專案實作。2016 年 11 月，台北：基峰。
- 五、楊正瑀（2017）。拓展醫療領域之新興應用。2017 年 01 月 16 日，取自
<https://www.ctimes.com.tw/DispArt-tw.asp?O=HK11G6GM4KCARASTD5>
- 六、陳賜賢（2015）。醫療機器人專利探勘與商機剖析。2015 年 03 月 11 日，取自
<https://mic.iii.org.tw/AISP/FreeS.aspx?id=3057>
- 七、衛生福利部疾病管制署（2020 年 4 月 16 日）。醫療機構因應嚴重特殊傳染性肺炎感染管制措施指引。取自：
https://www.cdc.gov.tw/Category/ListContent/NO6oWHDwvVfwb2sbWzvHWQ?uaid=UDXo5Wd2jDnm_vJUU9PjQ
- 八、潘乃欣(2020 年 03 月 03 日)。銘傳大學開發校園防疫 AI 提醒沒戴口罩的學生。聯合

報・取自 <https://udn.com/news/story/120960/4385219>

九、新竹市政府衛生局（2020 年 2 月 28 日）・新竹市馬偕醫院啟動「疫病門診」 就醫分流及早攔截高危險群[公告]・取自：

https://www.hccg.gov.tw/ch/home.jsp?id=30122&parentpath=0,16,30121&mcustomize=municipalnews_view.jsp&dataserno=202002280001&aplistdn=ou=municipalnews,ou=message,ou=ap_root,o=hccg,c=tw&toolsflag=Y&mserno=201601300023

十、Real-Time Face Recognition: An End-to-End Project. (2000, August 22) Hasker.io. Retrieved from <https://www.hackster.io/mjrobot/real-time-face-recognition-an-end-to-end-project-a1082>

【評語】 052509

本研究利用 Arduino 模組與臉部口罩辨識技術，串連「口罩辨識、體溫量測、酒精消毒」等三項主要 關卡來實作「防疫機器人」的嵌入式系統。目前雖僅是原型，但基本功能已經具備，未來可以進一步擴充功能。目前有關「口罩辨識」這部分所採用的機器學習方法沒有說明，且尚未進行實驗來量測在不同環境中的辨識錯誤率，未來可以進行更深入的探討。

作品簡報

科 別：電腦與資訊學科

組 別：高中職組

作品名稱：最堅強的醫護人員- 防疫機器人

關鍵詞：防疫、醫療機器人、臉部辨識



動機

新冠肺炎(COVID-19)自2019年爆發以來，造成許多人草木皆兵，人人皆害怕自己成為被感染的對象。但風險最高的一直是那群守在最前線，不斷為民眾的健康把關的醫護人員。作為疫情爆發後的第一道防線，他們需要大量且頻繁的接觸病患，是染上新冠肺炎的高風險群。而今年五月，台灣又再次爆發社區感染，更是讓醫護人員再次深陷病毒的危險中，我們希望可以盡一份力守護未受感染的民眾以及辛苦的醫護人員，因此試圖結合所學的Arduino知識，設計出一個可以代替第一線醫療人員的機器人，在醫院參與防疫工作，協助大家一起度過這次病毒的難關。

研究目的

研究防疫機器人
應具備的功能

測試各功能
的可行性
並進行組裝

建構出完整的
防疫SOP

精簡
醫護人員
人力

節省時間
防疫作業
更加精準

醫院分流
降低交叉感染
風險



Annual Primary & High School Science Fair
中華民國中小學科學展覽會

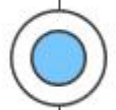


超音波感測系統

紅外線感溫系統

紅外線循跡系統

錄、放音系統



防疫機器人

- 辨識口罩
- 量測體溫
- 酒精消毒

使用Python
人臉辨識

使用MLX90614
感溫模塊

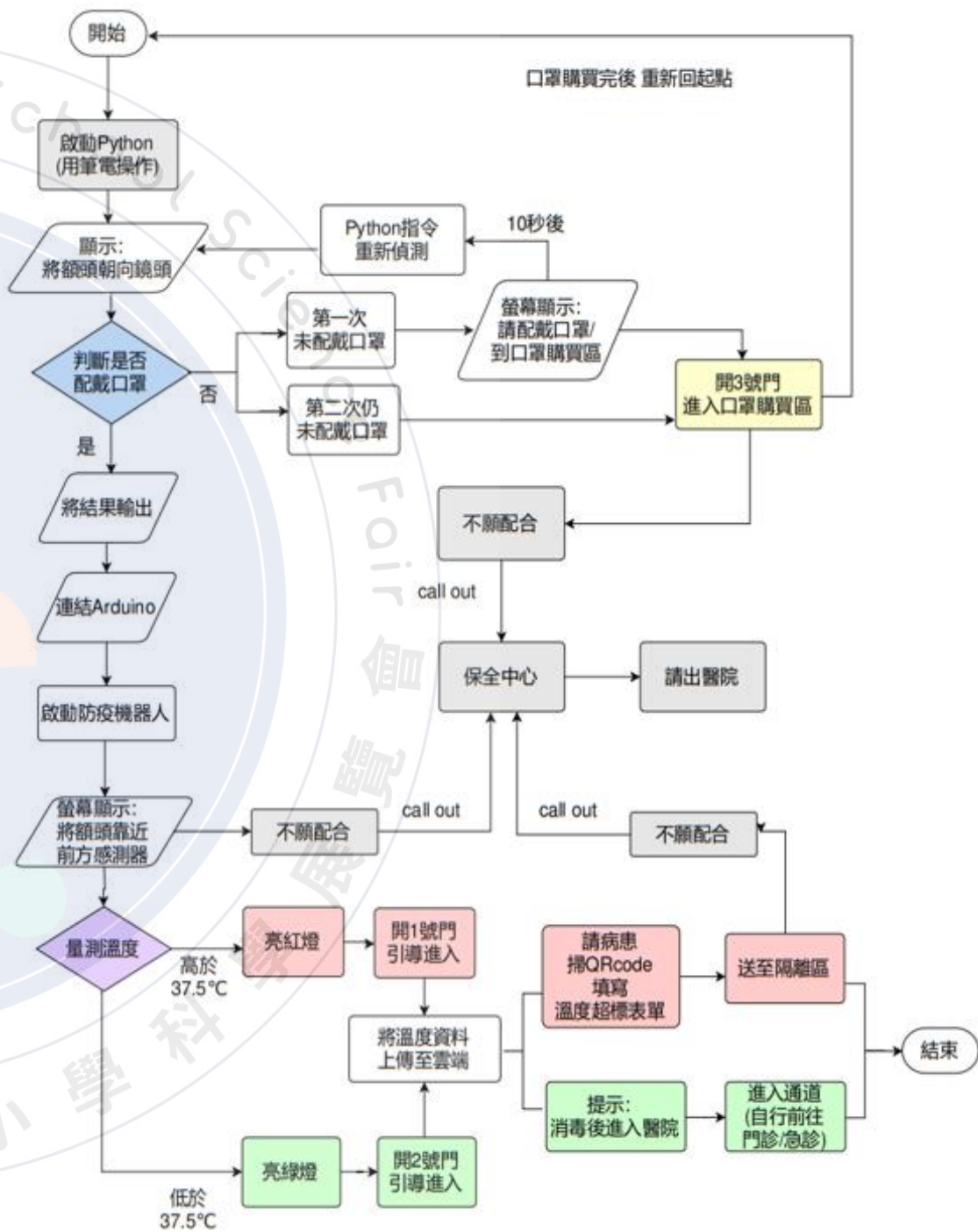
顯示Good(有戴)
或是Bad(沒戴)

顯示是否
超過37.5°C

資料蒐集

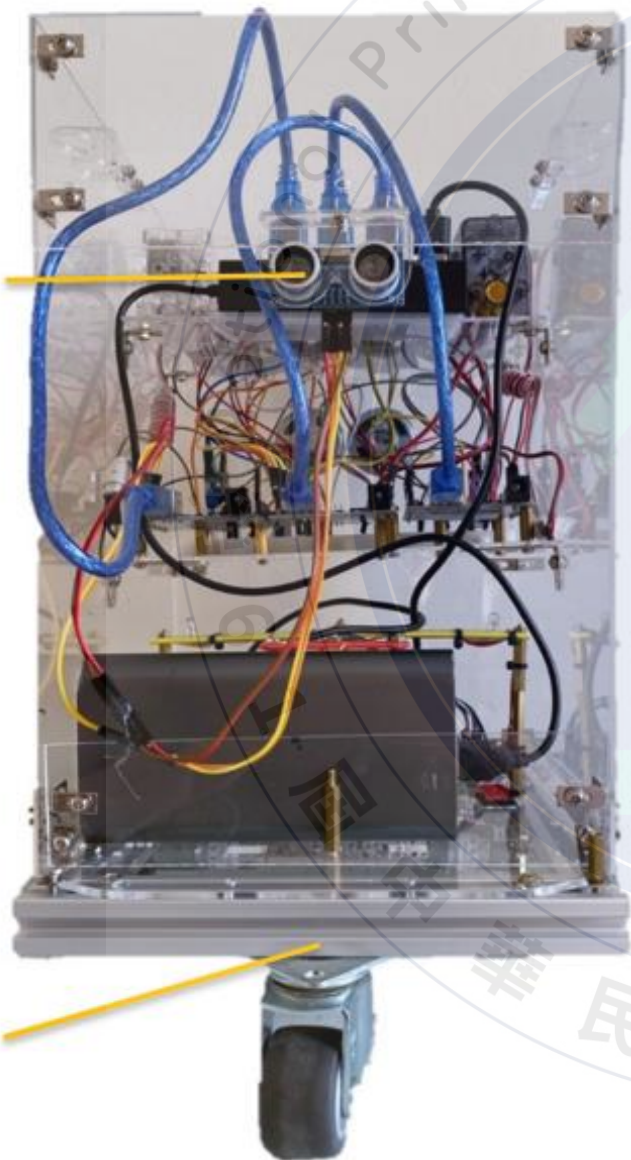
綠燈：PASS

紅燈：隔離



作品成果

超音波感測器



WiFi
遙控車

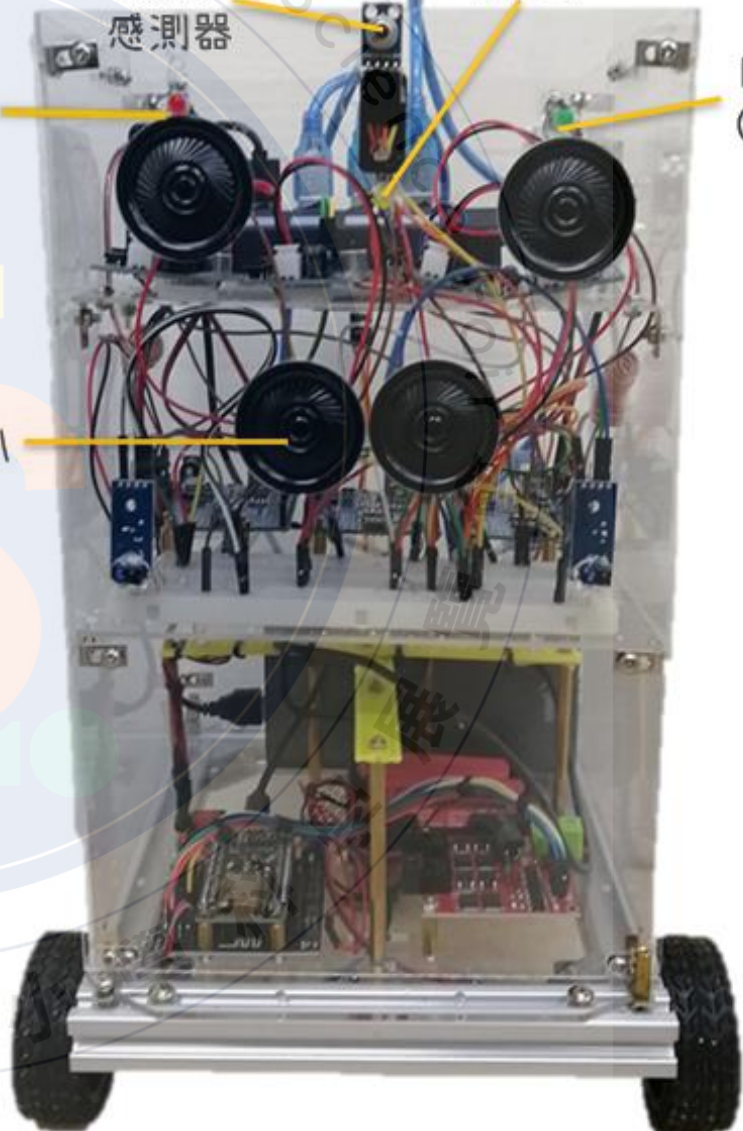
溫度感測器

LED燈
(黃色)

LED燈
(綠色)

LED燈
(紅色)

喇叭



中華民國中

作品成果

防疫相關資訊

由於您的體溫超過正常標準，而目前國內新冠肺炎疫情仍盛行，因此我們希望可以藉由這張表單，了解您的相關資訊，以便醫護人員可以為您作即時的治療或相關通報。

*必填

姓名 *

您的回答

連絡電話 *

您的回答

體溫 *

您的回答

過去14天是否曾出國 *

- 是
 否

過去14天是否曾因身體不適而前往診所及醫院 *

- 是
 否

同住親友是否有人收到居家隔離的通知 *

- 是
 否

過去14天內是否曾有咳嗽、喉嚨痛、流鼻水的症狀 *

- 是
 否

過去14天內是否有嗅覺或味覺喪失的症狀 *

- 是
 否

過去14天是否有腹瀉的症狀 *

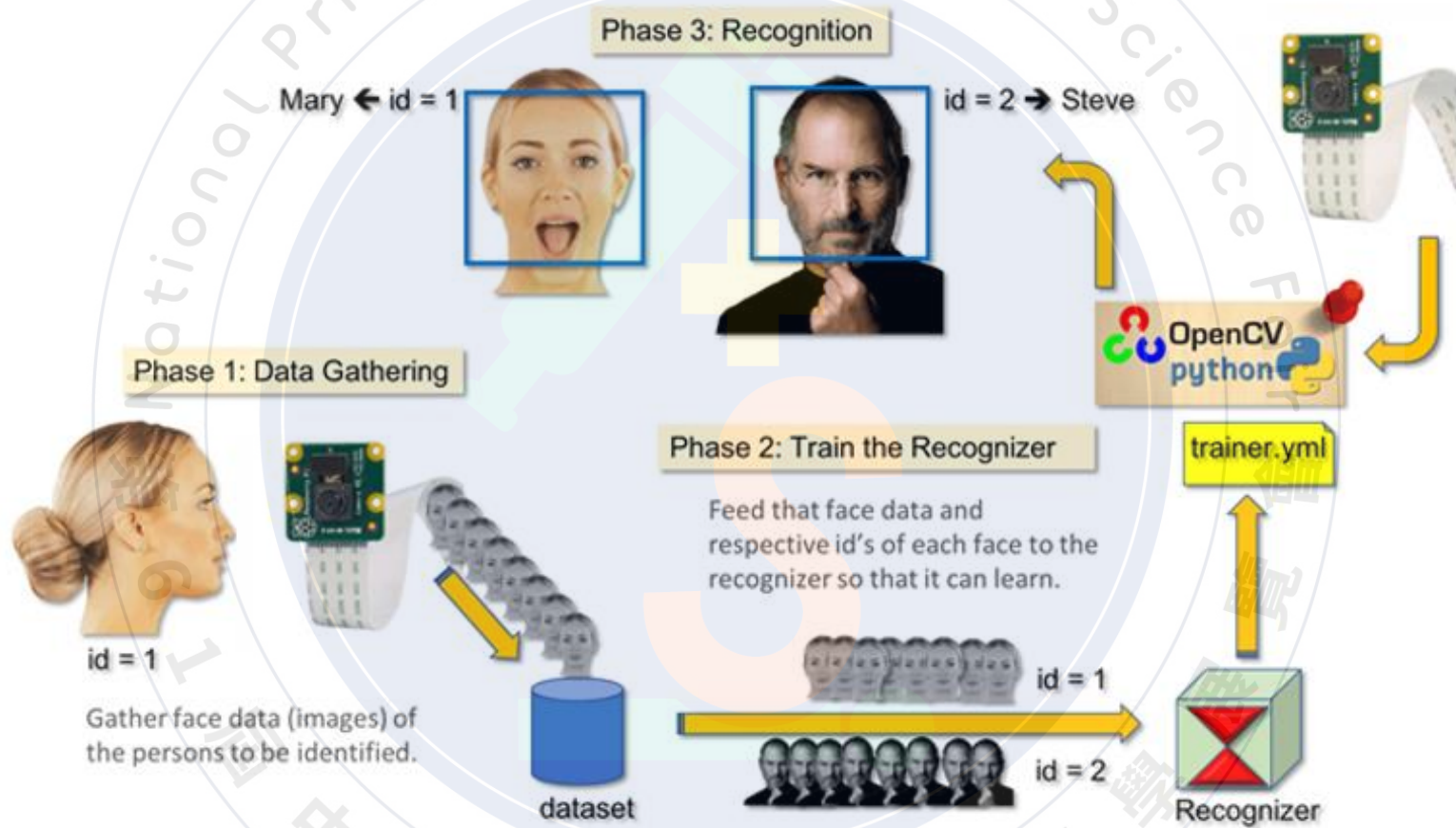
- 是
 否

過去14天是否有肌肉痠痛或四肢無力的症狀 *

- 是
 否

提交

人臉辨識



圖片來源：<https://www.hackster.io/mjrobot/real-time-face-recognition-an-end-to-end-project-a10826#code>

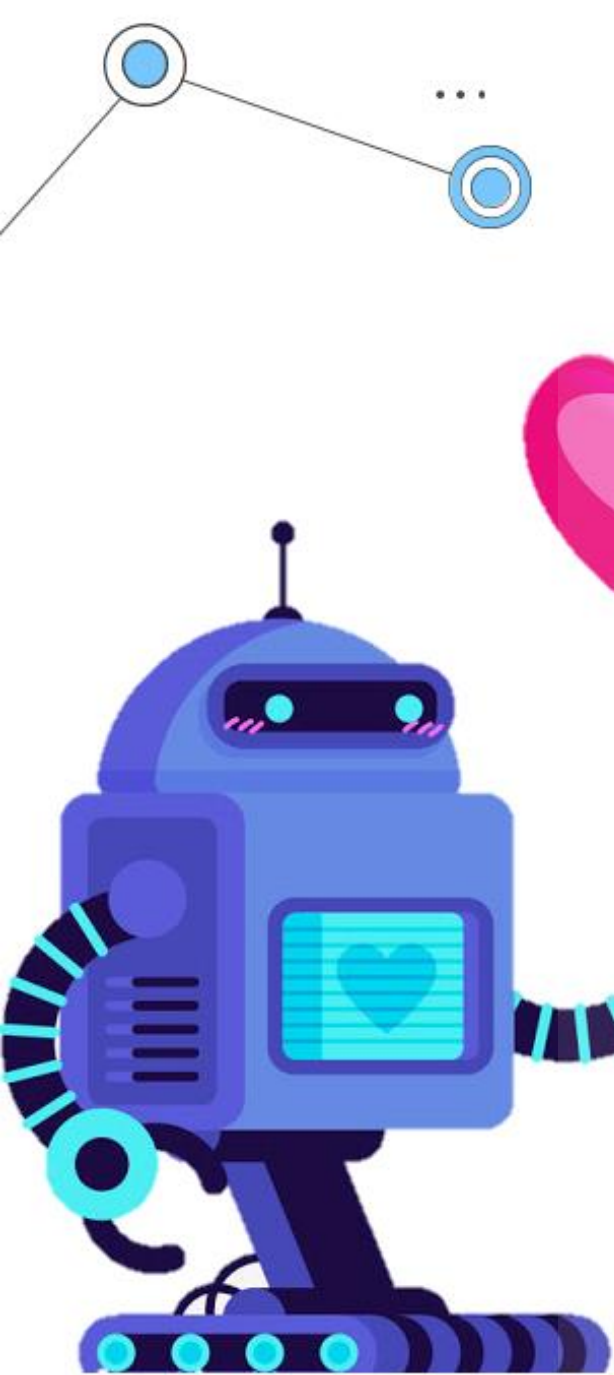
硬體考量

樹莓派	筆電
適合自動化，較差的環境	一般室內
有GPIO，適用多數感測或輸出模組	若有特殊功能／模組，需要串接開發板
要外接攝影機	維護介面較熟悉
運算量太大要考慮散熱	運算量大時，可以選較好的筆電

網路爬蟲



圖片來源：<https://tammy-discovery.com/best-webcrawler-resources/>



最堅強的醫護人員—
防疫機器人

謝謝大家

