

國立勤益科技大學資訊管理系

四技學生實務專題報告書

熱影像結合人臉偵測暨網頁瀏覽系統

組別:第八組

專題生:楊智傑(JHIH-JIE,YANG)

夏功霖(GONG-LIN,SIA)

蕭凱夫(KAI-FU, HSIAO)

巫宇哲(YU-JHE,WU)

指導老師:董俊良 博士

中華民國一一零年六月四日

一、摘要

隨著新冠肺炎疫情爆發，普羅大眾對於防疫意識也逐漸完善，人們對於出入安全顯得更為重視，從過去採用體溫機須配合人力使用，到現今逐漸採用熱影像系統，並逐漸加入一些生物特徵做為參考依據，如人臉偵測，都更為安全且便捷。

本研究在於改善現行之作業流程，將舊有的接觸式測量，改為使用熱影像技術測量法，提高安全性，也更快速便捷，本系統採用紅外線熱影像，並輔以人臉識別技術以改善系統溫度抓取之準確性，並透過黑體來校正，使溫度誤差縮小符合人體之溫度，透過後端時序資料庫掌握每一筆偵測之溫度，使用 grafana 將資料以圖表的方式呈現，使管理人員能夠快速且有效率的了解資訊，最後透過以整合好之網頁，使本系統達到異地閱覽之功能。

本研究希望能建立一個安全且有效率之體溫偵測系統，以符合使用者及監測人員需求，並藉由遠端控制達成零接觸、零感染安全有效之系統。

二、研究方法與過程

(一) 專題描述

本研究目的在於透過紅外線熱影像系統，結合人臉偵測技術與 wordpress 軟體，達到提升體溫測量安全性與準確性，本研究主要分為三大階段，影像處理階段、資料傳輸階段及網頁建制階段。

(二) 系統架構及系統流程圖

系統架構如圖 1 所示

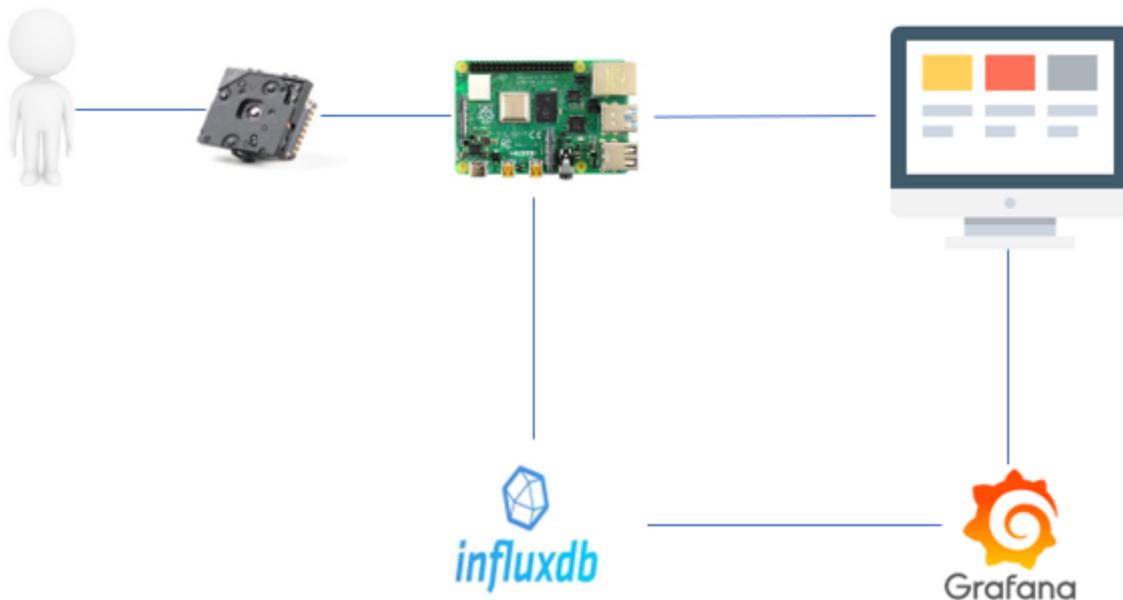


圖 1 系統架構圖

(三) 影像處理階段

在此階段首先須解決熱影像相關問題後才能與人臉識別技術進行整合，針對熱影像須了解市面上各種設備器材，在做需求評估決定硬體設備如圖 2 紅外線熱影像設備所示。



圖 2 紅外線熱影像設備

本研究最終選用 FLIR Lepton 3.5 做為紅外線熱影像設備，因其擴充性佳且其解析度為 160x120 使用在體溫偵測上可觀察較精準溫度，圖 3 設備接線圖。

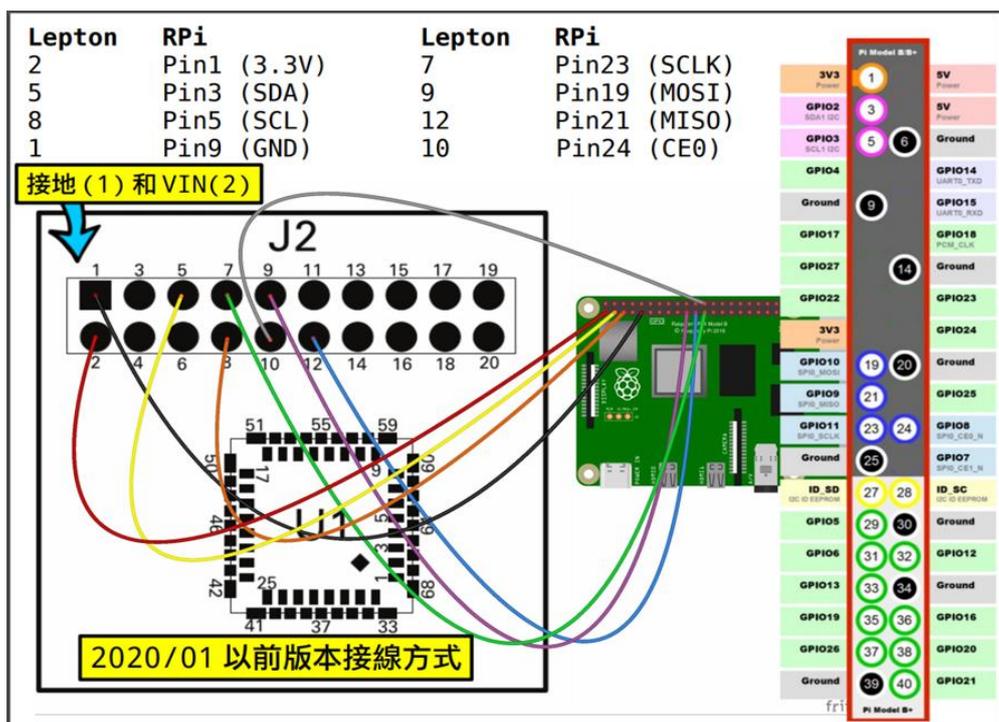


圖 3 設備接線圖

透過硬體廠商所提供範例程式了解到 c++版本與 python 版本各自的優缺點，c++版本與設備協同性較佳，不易產生失真的狀態且穩定性較佳，如圖 4 c++版本程式執行畫面，python 版本則是由 c++版本衍生而來，雖來 python 語言擴充性較佳，但是必會產生相容性問題，則穩定度較差亦產生失真問題，如圖 5 python 版本失真畫面，顧本研究選用 c++做為主要的程式語言。

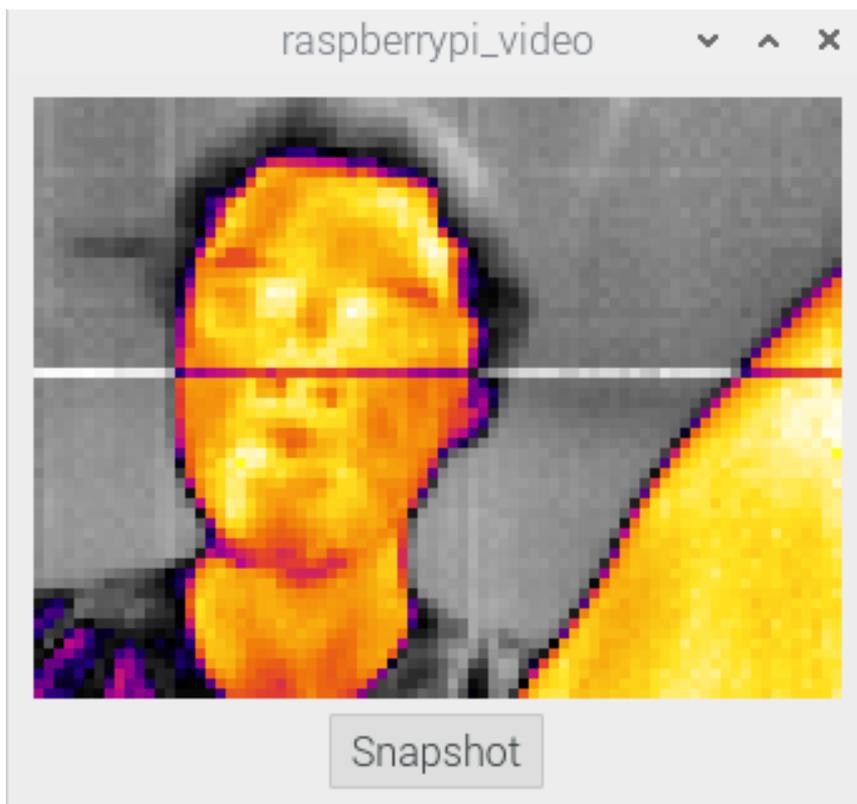


圖 4 c++版本程式執行畫面



圖 5 python 版本失偵畫面

解決熱影像相關問題後，仍須與人臉偵測進行整合，因熱影像占據電腦使用空間較大，使本研究必須降低人臉偵測對設備所造成的負擔，故我們選用較不占空間的人臉偵測程式，如圖 6 和圖 7 為人臉偵測圖。

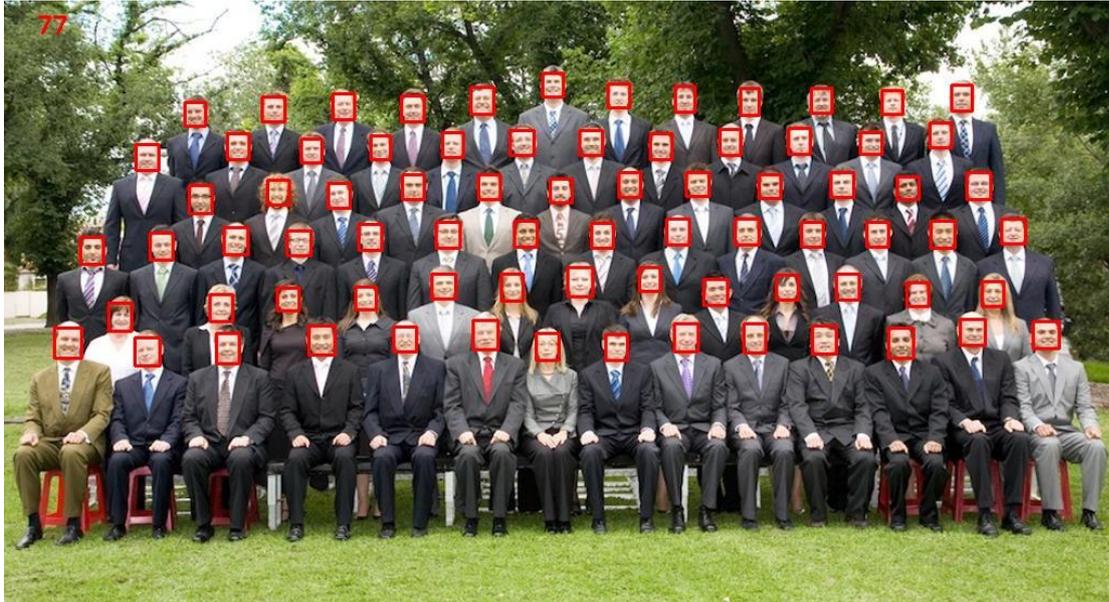


圖 6 人臉偵測圖 1

利用 MNN(Mobile Neural Network)深度學習框架，並用 widerface 執行訓練模型



圖 7 人臉偵測圖 2

本研究的人臉偵測模型訓練，是採用由 widerface 數據集合進行隨機取樣 10% 進行訓練，本次模型訓練包含了 3,221 張圖片與 29,160 人臉區域，該數據集合包含多種狀況及不同解析度圖片。在評估物件偵測模型準確度時，預測結果為真且實際結果也為真則稱 TP、實際為假則稱 FP， $(TP+FP)$ 與所有偵測到的物件應為真並且成功預測為真的比例 (TF)，最後在各類別 m 結果加總平均，即 mAP (mean Average

Precision)，如公式(2)所示，通常會加入一個限定條件來衡量預測出的物件框的精準度，透過真實結果的物件框範圍 gt^{area} 與預測結果的物件框範圍 p^{area} 交集與聯集的比例作為依據，(3)式所示；本研究在 IoU (Intersection over Union)大於 0.5 作為基準，mAP(mean Average Precision)為 84.29%

$$mAP = \frac{1}{m} \sum \frac{TP^m}{TP^m + FP^m} \quad (2)$$

$$IoU = \frac{gt^{area} \cap p^{area}}{gt^{area} \cup p^{area}} \quad (3)$$

圖 8 為媒合熱影像畫面與一般影像畫面程式碼，整合好人臉偵測技術與熱影像技術後，仍然需校正溫度，使偵測溫度為正常值，故本研究使用，單次黑體修正，使其溫度保持在誤差 $\pm 5^{\circ}C$ 內，同時由於兩顆鏡頭解析度也不相同，故在重疊熱影像畫面與一般影像畫面時需視各自解析度差修正，藉由框選黑體計算誤差值，並做補正動作，使設備溫度校正回歸至正常值，圖 9 黑體與校正畫面。

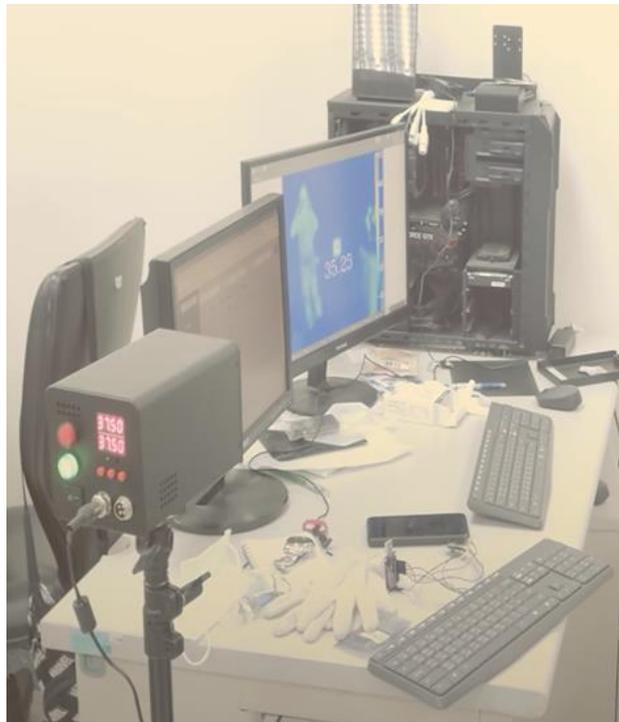


圖 9 黑體與校正畫面

```

roi_x = int(vw / 8)
edge_x1 += roi_x
edge_x2 -= roi_x
x1 = int((raw_x1 + roi_x) / 8)
y1 = int(raw_y1 / 8)
x2 = int((raw_x2 - roi_x) / 8)
y2 = int(raw_y2 / 8)
shift_raw = a[y1:y2, x1:x2]
print(self.x_bb,self.y_bb,self.x_ba,self.y_ba)
black_x1 = int(((self.x_bb)/8)+5)#+1
black_y1 = int((self.y_bb/8)-8)#-8
black_x2 = int(((self.x_ba)/8)+5)#+1
black_y2 = int((self.y_ba/8)-8) #-10 #1
print(black_x1,black_y1,black_x2,black_y2)

```

圖 8 媒合熱影像畫面與一般影像畫面程式碼

Jia-Wei 發現當處於幅射模式時來自 lepton3.5 的 14-bit 會被穩定並標準化，攝像頭發出的信號稱為 flux-linear，因為他與 lepton 3.5 光譜帶內的幅射值成線性關係，該線性訊號通過普朗克曲線與場景溫度有關：

$$S = \int_{\lambda_1}^{\lambda_2} \frac{2\pi hc^2}{\lambda^5} \frac{1}{\exp\left(\frac{hc}{\lambda k T_k}\right) - 1} R(\lambda) * \delta\lambda \quad (4)$$

(4) 式中 S 為攝像頭輸出的信號， λ_1 與 λ_2 為定義的光譜帶，h 為普朗克常數 c 為光速 k 是玻爾茲曼常數， $R(\lambda)$ 為攝像機的反應性， δ 函數常稱為單位脈衝符號或單位脈衝函數， T_k 代表絕對溫度，以開爾文為單位，但由於上式在硬體中無法使用，將 (4) 式轉換成 (5) 式

$$s = \frac{R}{\exp\left(\frac{B}{T_k}\right) - F} + O \quad (5)$$

S 同樣為攝像頭輸出的信號 R、B、F、O 是硬體在校準期間所生成的參數， T_k 代表絕對溫度，已開爾文為單位，在將 (5) 式變換為 (6) 式

$$T_k = \frac{B}{\ln\left(\frac{R}{S-O} + F\right)} \quad (6)$$

在本研究中我們使用 Lepton3.5 像機以獲取熱影像，但是在出廠設定中，Lepton3.5 的輸出值為不穩定的 14-bit，為了獲取實際溫度值，我們首先須做幅射度校正，同時根據參考文獻我們可得知 R、B、F、O 個別參數的範圍，以下會解釋個參數的用處

R=[10000, 1000000]
B=[1200, 1700]
F=[0.5, 3]
O=[-16384, 16383]

Jia-Wei 提到 F 參數值通常不需改變，除非測量環境溫度非常高，超出了 Lepton3.5 的範圍，由參考文獻得知 B 參數的常用值為 1428，如果 F 參數與 B 參數都具有固定值不需變動，那需要校正便只剩下 R 參數與 O 參數，R 參數代表 Lepton3.5 的響應值，O 參數則為偏移量，由圖 10 得出擬合回歸線(fitting the regression line)得知 R 值為 231159.5，O 值為 6094.248，最終帶入公式可知(7)

$$T = \frac{1428}{\ln\left(\frac{231159.5}{S - 6094.248} + 1\right)} - 237.15 \quad (7)$$

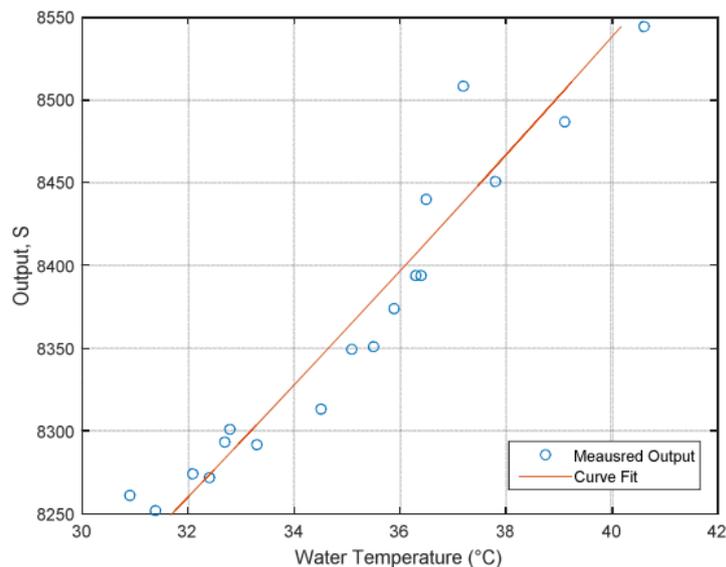


圖 10 測量出的擬合結果

調整完參數後得到圖 11 為影像處理階段最終畫面



圖 11 影像處理階段最終畫面

(四)資料傳輸階段

本研究使用時序資料庫 influxdb 做為資料庫負責儲存所偵測到的溫度，並使用 docker 軟體包裝 influxdb 除了可以將應用程式快速建置之外，也可以減少成本，圖 13 為將溫度存到資料庫的程式碼。

```
influxdb_cpp::server_info si("127.0.0.1",8086,"thermal");
influxdb_cpp::builder()
.meas("degree")
.tag("topic","temperature")
.field("value",temp_d)
.post_http(si);
```

圖 13 溫度存到資料庫的程式碼

最終透過 Grafana 串接 influxdb 就可以呈現當下偵測到的溫度為攝氏 35.8 度如 圖 14，還能夠顯示當下往回推半小時所偵測到的溫度如 圖 15，以表格呈獻近三筆偵測

到的溫度如圖 16，計算溫度被偵測到的次數如圖 17，和當溫度超過自己所設定的值時會發送警告如圖 18，並且發送警告到 line 軟體上如圖 19。



圖 14 當前偵測溫度



圖 15 顯示近一小時偵測到的溫度

Panel Title	
Time	degree.mean (0; , topic: temperature)
2021-05-07 17:00:00	38.0
2021-05-07 17:20:00	37.1
2021-05-07 20:20:00	35.8

圖 16 以表格呈獻近三筆偵測到的溫度

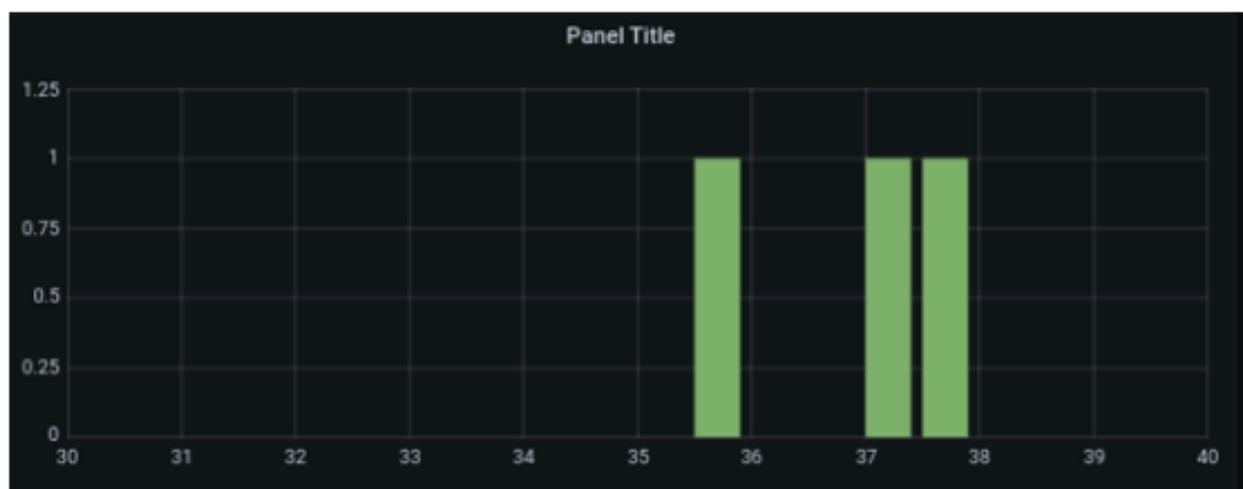


圖 17 計算溫度被偵測到的次數

Panel Title	
message  PENDING degree.mean=38.75	May 7, 2021 17:24:25
message  OK	May 7, 2021 17:24:23
message  ALERTING degree.mean=38.79	May 7, 2021 17:24:19
message  PENDING degree.mean=38.75	May 7, 2021 17:24:17

圖 18 溫度超過自己所設定的值時會發送警告

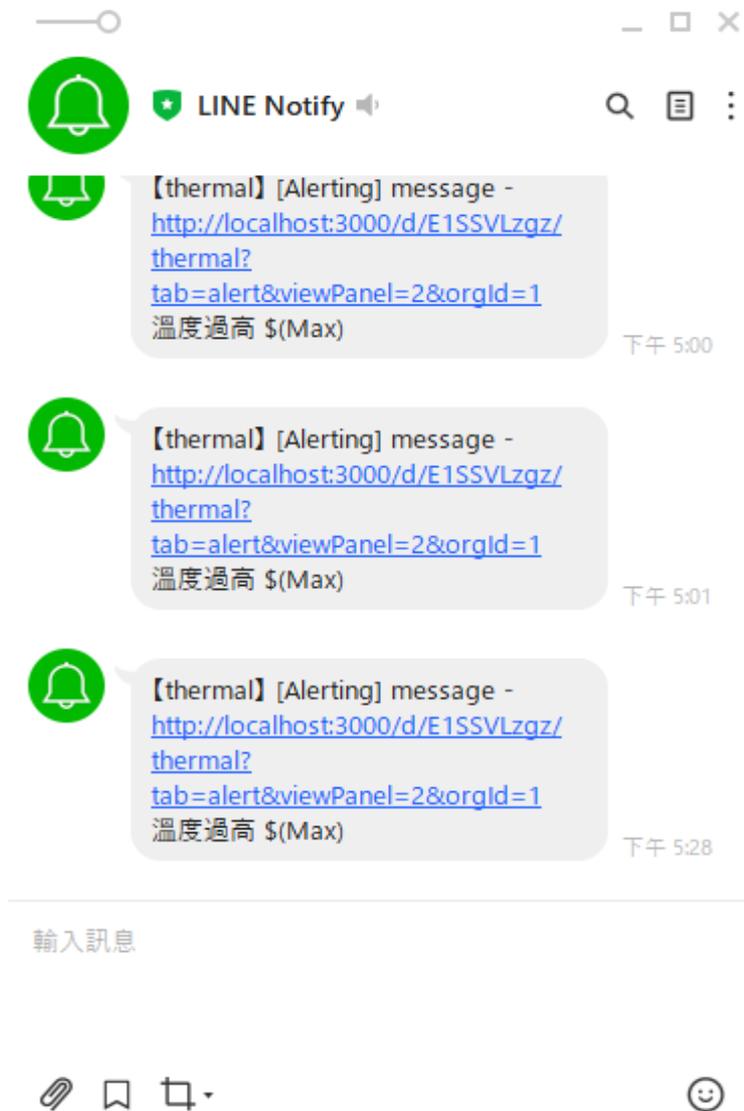


圖 19 發送警告到 line 軟體上

(五)網頁建制階段

本研究使用 MJPG-streamer 將執行於電腦上的程式，以 HTTP 的方式進行訪問，如圖 20，並使用 wordpress 軟體架設網站，透過 iframe 連結並整合程式及後端圖表，如圖 21 最終系統樣貌

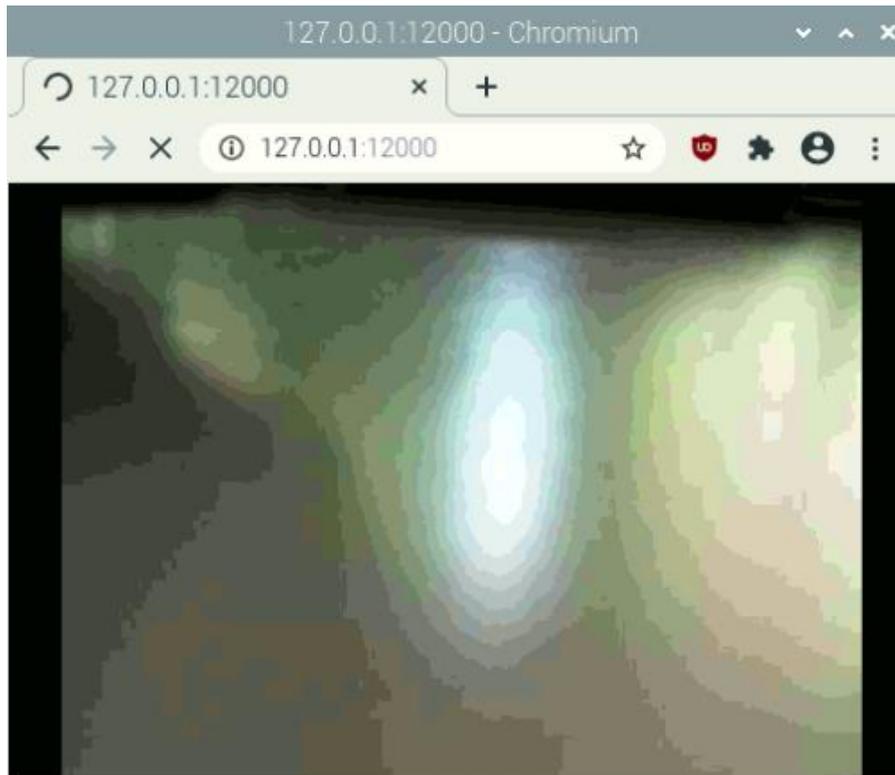


圖 20 以 HTTP 方式訪問程式

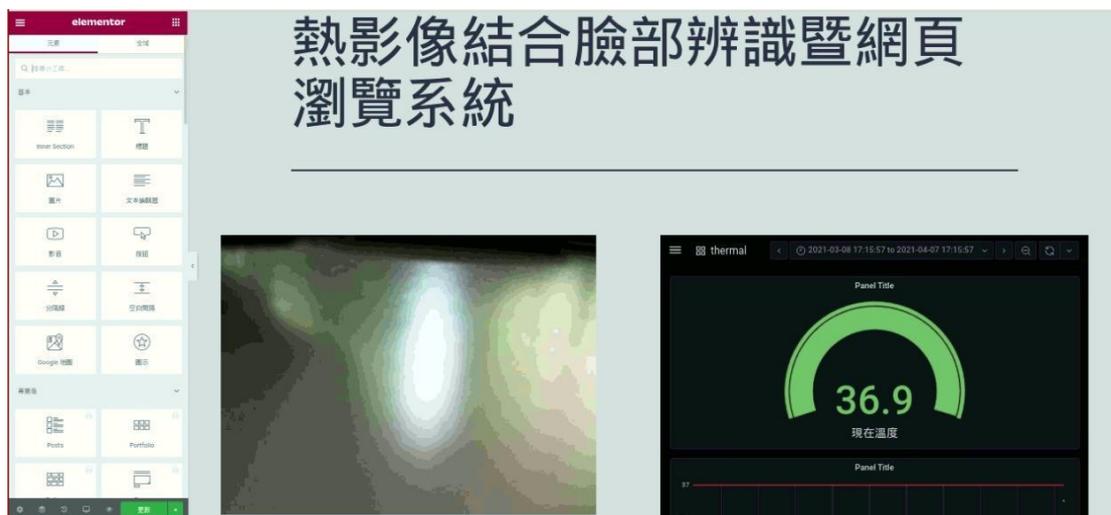


圖 21 最終系統樣貌

三、結論

本研究開發及設計出熱影像結合人臉偵測暨網頁瀏覽系統，該系統可在網站上即時閱覽當前監控畫面，可使得操作人員與畫面中人群保持安全距離，以降低染病風險，並透過以整合好的熱影像與人臉偵測功能，達到快速並準確辨識人臉及當前體溫，降低複雜的體溫測量流程，簡化每一位人員的工作手續，並保證人員的生命安全，並針對熱影像體溫誤差缺失加以改善，本系統仍有待改善之處，是未來研究努力的目標。

(一)研究成果

1. 數據成果

- (1)使用 raspberry pi 作為搭載平台，使得系統體積大幅縮小
- (2)將熱影像設備的誤差從 $\pm 5^{\circ}\text{C}$ 修正至 $\pm 1^{\circ}\text{C}$
- (3)結合人臉辨識系統使得體溫偵測不受到高溫物體的熱影響
- (4)修改人臉辨識的特徵值使其能偵測配戴口罩人員之體溫
- (5)透過 MJPG 使其能在 WEB 上瀏覽系統之影像
- (6)將偵測後的體溫繪製成圖表，以利後續統計之用
- (7)當體溫異常會產生提醒音效

藉由熱影像結合人臉偵測暨網頁瀏覽系統的運作，讓監管人員與受測人員可以達到下列成效：

2. 監控人員方面：

- (1)減少設備所佔用體積，使空間更寬暢
- (2)減少工作之負擔，並簡化相關流程
- (3)保持安全距離使人員安全更受保障
- (4)主管可透過網頁觀看當前畫面與過去測量體溫記錄統計之圖表
- (5)即使人員離開電腦，當溫度有異常也能透過通訊軟體得知

3. 受測人員：

- (1)加快檢疫速度，使其走過便能留下體溫紀錄

(二)後續研究

本研究目前因時間及資金限制，導致開發不夠完善，未來希望能以下幾點作出改善的方向：

1. 採用固定黑體校正，使其誤差降至 $\pm 0.1^{\circ} \text{C}$
2. 重新設計外觀讓系統更符合現代美學
3. 新增計算人次功能
4. 增加人臉識別功能使其功能更加完善