# 個人專題研究報告

研究主題：

**基於深度學習 - 人臉識別考勤系統**

**研究生：梁家銘 執導教授：郭家旭**

[研究動機 2](#_Toc114913456)

[系統架構設計 4](#_Toc114913457)

[系統設計 4](#_Toc114913458)

[系統部署 5](#_Toc114913459)

[研究方法與設計 6](#_Toc114913460)

[**辨識畫面的呈現 6**](#_Toc114913461)

[影像處理 6](#_Toc114913462)

[人臉偵測 6](#_Toc114913463)

[**人臉關鍵點訓練 7**](#_Toc114913464)

[**人物識別 7**](#_Toc114913465)

[預期解果 8](#_Toc114913466)

[系統展示(目前進度) 8](#_Toc114913467)

[**已知人物的識別 8**](#_Toc114913468)

[**未知人物的識別 8**](#_Toc114913469)

[問題與解決方案(目前進度) 9](#_Toc114913470)

[**人物識別誤判問題 9**](#_Toc114913471)

[參考資料 10](#_Toc114913472)

研究動機：

隨著科技日新月異，深度學習在各種不同領域成效令人驚嘆，對於影像辨識的應用亦是普遍廣泛，如人臉辨識考勤系統，須多公司企業也陸續的將這門技術引入自家公司中使用，智能考勤系統相比傳統的打卡方式，數位化的人臉識別節省了須多不必要的浪費，如打卡紙，墨水等，同時人員出勤的數位化也為大型公司省下了不少資料整理時間。

根據目前市場上人臉識別考勤系統使用者給出的回饋表示，目前的系統技術任然還有改善的空間，人臉辨識在某些外在因素的影響下，還是很難正確的做出辨識，例如鏡頭角度、臉部表情、場景光線等影響，也因為如此或許我們能使用不同的模型訓練方法來提高辨識的成功率，改善目前市場上的問題。

**研究目的**：

* 學習人工智慧模型訓練的相關技術知識。
* 學習硬體對神經網路加速的相關知識。
* 學習人工智慧結合物聯網的相關應用。
* 改善市場上人臉識別考勤系統已知的問題。

系統架構設計：

系統架構分為Ｗeb Cam、Client、Server三端，主要以網路通信，如圖1-1所示。

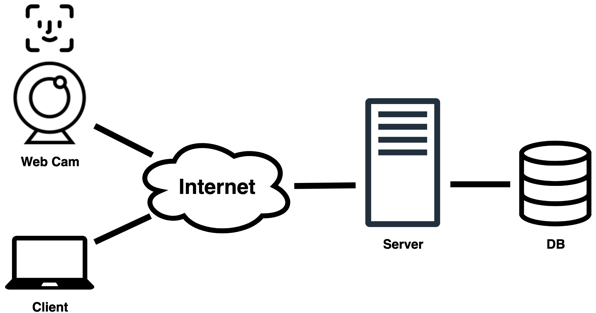


圖1-1:系統架構

(資料來源:本研究)

系統設計

* Web Cam：採用 C++ 語言開發，負責收集影像數據。設備使用 ESP32-CAM 無線攝像頭收集影像數據，並搭配 ILI9341 TFT LCD 螢幕呈現拍攝畫面。
* Client端：採用 React 網頁框架以 Typescript 語言開發，負責與 Server 的溝通，提供多種考勤資料下載的相關功能。
* Server端：採用 Python 語言開發，負責人物識別神經網路運算，並且提供識別運算與資料下載的相關接口。神經網路模型採用 Dlib 提供的人臉偵測模型與臉部識別模型，搭配 NVIDI GTX 165 8GB 顯示卡來加速運算。資料庫使用 MySQL 關聯式資料庫，更好的管理成員與部門的相關數據。

系統部署

本系統部署圖如圖1-2所示，系統各組件工作描述如圖1-3所示。

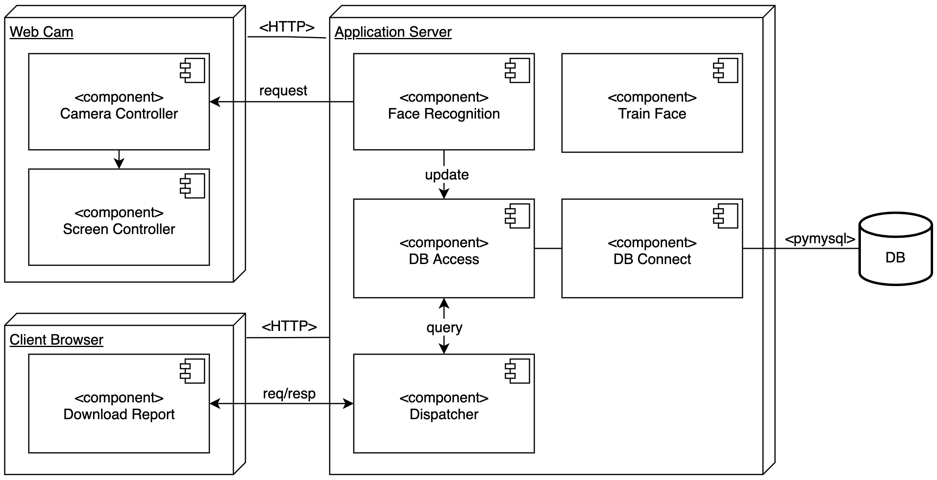


圖1-2:系統部署

(資料來源:本研究)

|  |  |
| --- | --- |
| **Component** | **Description** |
| Camera Controller | 負責影像數據的收集。 |
| Screen Controller | 負責拍攝影像畫面的呈現。 |
| Download Report | 提供考勤資料下載的功能。 |
| Face Recognition | 負責人臉識別神經網路運算。 |
| Train Face | 負責人臉識別模型的訓練。 |
| DB Access | 維護資料庫事務，提供封裝好的資料庫操作。 |
| DB connect | 負責資料庫的連接，提供資料庫增、刪、改、查操作。 |
| Dispatcher | 負責將Client端的請求分派給對應的資料庫操作方法。 |

圖1-3:系統組件工作描述

(資料來源:本研究)

研究方法與設計

辨識畫面的呈現

基於 **Ajax** 工具開發

Web Cam 端利用人物識別的API接口，將拍攝畫面送到到 Server辨識，並在完成辨識後，再將標註好辨識結果的圖片回傳給Web Cam 呈現畫面。

影像處理

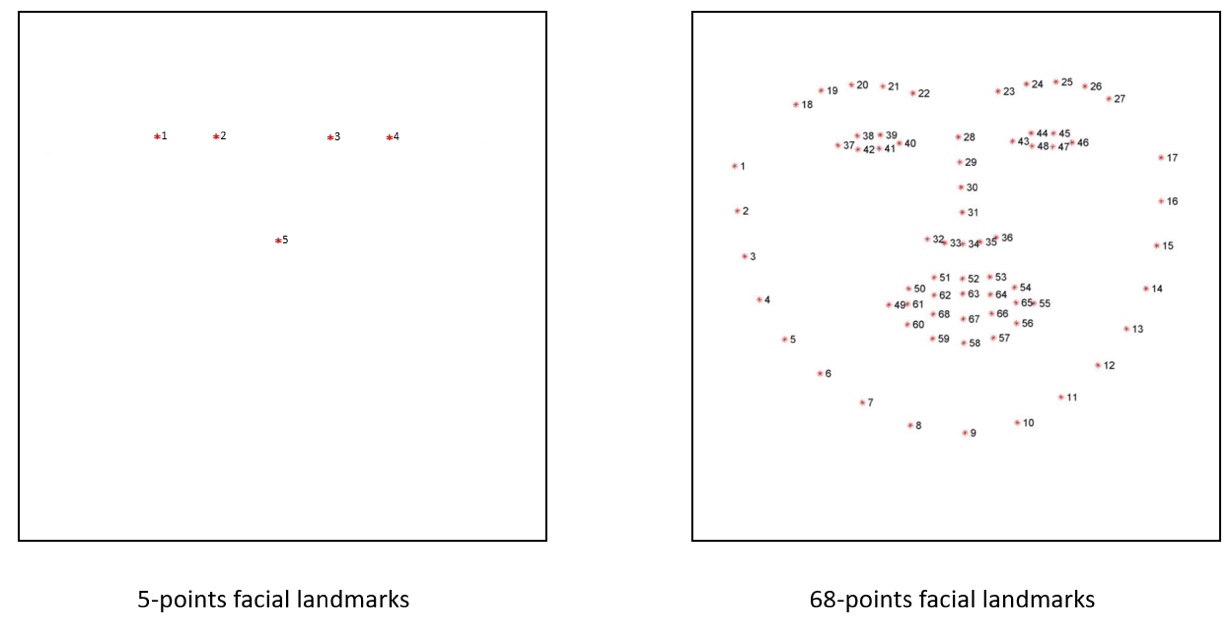
基於 **OpenCV** 工具開發

OpenCV 具有豐富的影像處理函式庫，能夠很好的幫助我們在影像上的操作，如解析度放大縮小、臉部辨識結果的畫面呈現。

人臉偵測

基於 **Dlib/OpenCV** 工具開發

使用 Dlib 所提供的深度卷積人臉偵測模型，此模型是在一個已標注好關鍵點的資料集上做訓練，最終達到能夠預測出人臉上五官的位置。

人臉關鍵點偵測分為以下兩種，如圖1-3所示。

|  |  |
| --- | --- |
| Facial Landmarks | Description |
| 5-points | 使用**左眼頭尾、右眼頭尾、鼻頭**這五個點來做辨識 |
| 68-points | 使用**外輪廓、眉毛、眼睛、鼻子、嘴巴**共68個點來做辨識 |

圖1-3:影像關鍵點

(資料來源: iT邦幫忙, 山姆大叔)

人臉關鍵點訓練

基於 **Dlib/Numpy/OpenCV** 工具開發

使用 Dlib 臉部識別模型。該模型在 Wild 基準的 Labeled Faces 上的準確率為 99.38%。本系統使用 68-points 關鍵點作為模型訓練參數如圖1-5所示，利用影像的抖動的方式增加訓練樣本，並訓練已知人物的關鍵點數據，最後再將訓練好關鍵點數據儲存下來。

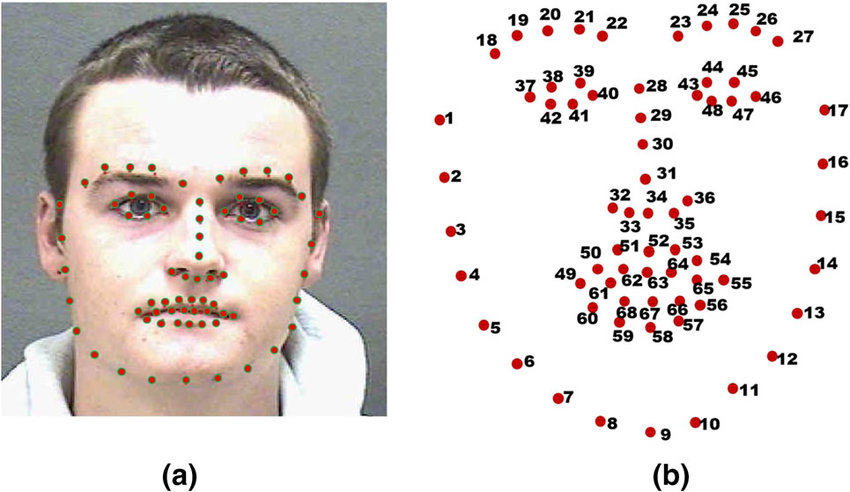


圖1-4:68-Points 臉部關鍵點圖

(資料來源:ResearchGate, [Ali Ahmed Elmahmudi](https://www.researchgate.net/profile/Ali-Elmahmudi))

人物識別

基於 **Dlib/Numpy/OpenCV** 工具開發

使用 Dilb人臉偵測模型與臉部辨識模型預測，分為以下三步驟：

* 人臉偵測：將 Web Cam 所拍攝到的畫面作為人臉識別模型的輸入，得出畫面中的人臉位置。
* 臉部識別：在將人臉畫面作為臉部識別模型的輸入，得出人物的臉部關鍵點數據。
* 人物識別：利用已知人物關鍵點數據減去拍攝畫面中的人臉關鍵點數據，得出誤差值，並在系統上設定一個識別閥值，若誤差小於該閥值，則視為已知人物，如圖1-6所示。

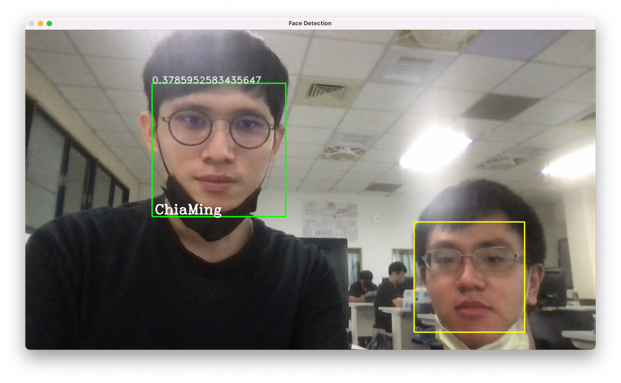


圖1-6:人物識別結果 綠框為已知人物 黃框為未知人物

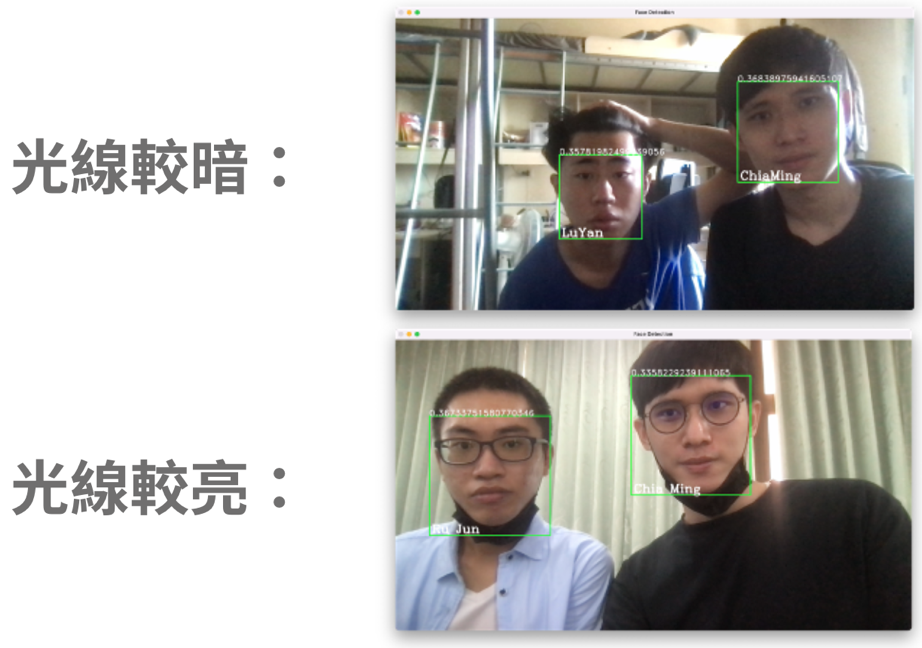
(資料來源:本研究)

預期解果

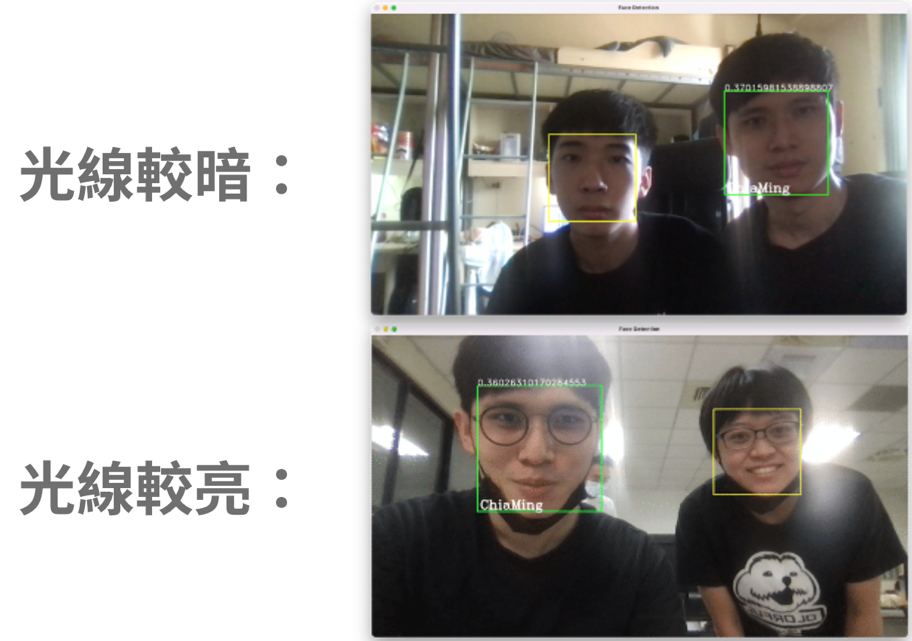
* 在鏡頭距離人物50 cm內能夠100%成功擷取人臉特徵
* 在光線充足的空間中能夠達到95%以上的成功辨識

系統展示(目前進度)

已知人物的識別



未知人物的識別



## 問題與解決方案(目前進度)

### 人物識別誤判問題

* 問題描述：在剛開始的時候，已知人物訓練資料採用單一樣本進行訓練，導致會把未知的人物也辨識為已知的某位人物。
* 解決方案：在訓練已知的人物時設置影像抖動參數，將訓練資料透過隨機偏移生成新的訓練資料，以此來增加訓練的樣本數量，並且降低關鍵點誤差的閥值。在設置抖動參數將單一樣本提升至 50 個樣本以及，將關鍵點誤差閥值由原本的 0.6 降至 0.4 後，解決了已知人物誤判的問題。

## 參考資料

[1] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks

https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf