



嶺東科技大學
LING TUNG UNIVERSITY

資訊管理系

影像辨識

指導教授：陳健忠 教授

組員名單：呂鎮遠 A78C005

賴昭廷 A78C019

劉至城 A78C054

許家誠 A78C062

中 華 民 國 1 1 1 年 0 5 月

110
學
年
度

嶺東科技大學
資訊管理系

影像辨識



嶺東科技大學
LING TUNG UNIVERSITY

資訊管理系專題口試委員審定書

影像辨識

指導教授： 陳健忠 教授

組員名單： 呂鎮遠 A78C005

賴昭廷 A78C019

劉至城 A78C054

許家誠 A78C062

指導教授： _____

口試委員： _____

中 華 民 國 1 1 1 年 0 5 月

謝 誌

本專題報告得以順利完成，首先要感謝恩師陳健忠老師細心引導我們，耐心的協助我們，克服研究過程中所面臨的困難，給予我們最大的協助，使本專題得以順利完成。

研究報告口試期間，感謝黃光宇老師、林甘敏老師不辭辛勞、細心審閱，不僅給予我們指導，並且提供寶貴的建議，使我們的專題內容以更臻完善，在此由衷的感謝。

在此致上最誠摯的謝意,謹以本專題帶著說不盡的感激、感動與喜悅，獻給所有關心我們的教授及組員。

呂鎮遠

賴昭廷

劉至城

許家誠 謹誌

中華民國111年05月於嶺東

摘要

本研究目的為設計應用於校園行政上使用的影像辨識系統。當今學校行政上要查詢學生的基本資料以及其他相關資訊時，都必須以文字搜尋的方式才能從學校資料庫中拉取。本專題目的為設計影像辨識結合校園行政的系統，利用人臉辨識來辨別是否為本校的學生，將校園資料庫抽取該名學生的基本資料，直接顯示在電腦螢幕上省略查詢步驟，提升學校行政效率。

為此利用人臉辨識系統應用於校園行政作業，以提升學校行政效率，縮短時間；研究方法主要利用攝像頭進行拍攝，將畫面傳輸至電腦中，再利用 Anaconda 平台內架設 Python 虛擬環境並安裝的 Open CV 以及 Dlib 套件，進行人臉特徵提取尋找出畫面中的人臉，並且比對 MySQL 資料庫的學生人臉特徵資料表，藉由影像辨識系統，把影像檔進行人臉特徵提取，與雲端資料庫比對出特徵相符的學生，最後在電腦中顯示影像辨識結果最符合相似資料的學生。

關鍵詞：影像辨識、人臉辨識、資料庫、辨識分析

目 錄

摘 要	I
目 錄	II
表目錄	IV
圖目錄	V
第壹章 緒論	1
1.1 研究背景	1
1.2 研究動機	1
1.3 研究問題	1
1.4 研究目的	1
第貳章 文獻回顧與探討	2
2.1 文獻探討	2
2.1.1 校園行政管理	2
2.1.2 影像辨識系統	2
2.1.3 人臉辨識	2
2.1.4 資料庫	2
2.1.5 關於網站建置	2
2.2 相較其他辨識的優、缺點	3
2.3 人臉辨識的相關技術	3
第參章 研究方法	4
3.1 研究架構	4
3.1.1 硬體系統架構流程圖	5
3.1.2 軟體系統架構流程圖	6
3.1.3 甘特圖	7
3.2 研究流程	8
3.3 研究安裝步驟	9
3.4 系統流程	10
3.5 創建資料庫	12
第肆章 系統分析與設計	13
4.1 系統設計	13
4.2 系統模擬	14
4.3 模型訓練	16
4.4 系統建置硬體	17
4.4.1 JSING WB1 視訊攝影機 Webcam	17

4.4.2 電腦.....	18
4.5 系統建置軟體	20
4.5.1 Anaconda.....	20
4.5.2 Python3	21
4.5.3 MySQL.....	21
4.5.4 XAMPP	22
4.5.5 Open CV.....	22
4.5.6 DLIB	23
4.5.7 HTML	24
4.5.8 PHP	24
4.6 人臉影像辨識套件比較.....	25
4.6.1 模組套件一	25
4.6.2 模擬程式套件二.....	27
4.7 網站架構.....	28
4.7.1 E-R model 圖	29
4.7.2 關聯式資料表.....	30
4.7.3 資料表欄位.....	31
4.8 功能展示	32
第伍章 結論與未來展望.....	33
5.1 結論	33
5.2 未來展望	33
參考文獻.....	34

表目錄

表 2.1 人臉辨識相較於虹膜、指紋、聲音辨識系統的優、缺點	3
表 4.1 系統設計所需功能及用途.....	13
表 4.2 視訊攝影機規格.....	17
表 4.3 電腦1規格.....	18
表 4.4 電腦2規格.....	19
表 4.5 軟體設備	20
表 4.6 沒有戴口罩下擷取和辨識人臉測試表.....	25
表 4.7 有戴口罩下擷取和辨識人臉測試表.....	26
表 4.8 學生資料表	31
表 4.9 學期成績資料表.....	32

圖目錄

圖 3.1	研究流程圖	4
圖 3.2	硬體系統架構流程圖	5
圖 3.3	軟體系統架構流程圖	6
圖 3.4	甘特圖	7
圖 3.5	影像抓取情境圖	8
圖 3.6	人臉辨識步驟	8
圖 3.7	人臉辨識系統流程圖	10
圖 3.8	創建資料庫	12
圖 4.1	68點個特徵點	14
圖 4.2	辨識人臉圖片	15
圖 4.3	模型訓練三階段	16
圖 4.4	視訊攝影機	17
圖 4.5	電腦1 ASUS X560UD	18
圖 4.6	電腦2 ASUS VivoBookS510U	19
圖 4.7	Anaconda	20
圖 4.8	Python3	21
圖 4.9	MySQL	21
圖 4.10	XAMPP	22
圖 4.11	Open CV	22
圖 4.12	Dlib	23
圖 4.13	HTML	24
圖 4.14	PHP	24
圖 4.15	模擬程式套件一執行圖	25
圖 4.16	模擬程式套件二執行圖	27
圖 4.17	人臉辨識後輸出至網站的介面	28
圖 4.18	E-R model	29
圖 4.19	關聯式資料表圖	30
圖 4.20	成果展示圖	32

第壹章 緒論

1.1 研究背景

隨著網際網路應用日漸普及、電子化交易時代的來臨，許多資料應用在網路與電腦上，須獲得個人資料就必須具備一定程度的認證方式，剛開始是個人密碼、晶片卡...等認證方式，不過在近年電腦計算機性能超越以往的運算速度，使密碼在網路上發生被破解或是他人盜用的情形，且難以確保認證的安全性，使用智慧卡或晶片卡也不安全，因為卡片可能被偽造、竊取、遺失、或無法讀取資料，想找出不需要攜帶，也不會遺失的安全認證。

在網路發達的世代，科技硬體與軟體技術的突破，若能使用到安全認證，使我們提高保障。而利用到最新人體特徵的辨識系統，是做為防護系統最有效且安全的方法。而特徵辨識系統，例如：聲音、指紋、虹膜和人臉...等辨識系統，稱生物辨識，其中人臉辨識是最直接的辨認方式，依照個人在臉部影像所呈現的特徵做身份辨識。

1.2 研究動機

現在科技進步的時代，除了硬體的功能汰舊換新，軟體系統也逐漸日新月異，往創新、快速、安全的方向發展，讓生活日常中便有了更進步的使用，從以前的鑰匙開鎖，進步到輸入電子密碼或是智慧卡...等，在安全性與性能上有了大幅度的進展，但到了大學生活後發現，學生因壓力大時常出現記憶力不好與怠惰的狀況，且智慧卡片遺失、忘記攜帶的問題，讓學生在學習的動力上減少信心，在這些情況下我們認識到有企業運用人臉辨識系統進行身分確認，現在學生到學校辦公都要帶著學生證，才能確認該名學生身分，辦理學校事務，因此想考慮利用此方法來辨識學生身分。

1.3 研究問題

根據研究動機中，推斷出以下四種研究問題

- 1.影像辨識系統是否能準確辨別每一位學生？
- 2.人在移動過程中是否能準確提取人臉特徵？
- 3.人臉影像辨識套件差異比較

1.4 研究目的

因考量到新生和其他有些遺失學生證的人無法辦理公務，延誤他們想辦理公務的時間，希望可透過本專題之人臉辨識，即可確認該學生身分，且讓整體行政運作更有效率。

- 1.透過影像系統比對找出影像中的人確認身分。
- 2.透過影像辨識軟體比對人臉資料，確認身分後連結資料庫並搜尋學生資料。
- 3.分別以兩種模組軟體，比較人臉影像辨識套件

第貳章 文獻回顧與探討

2.1 文獻探討

2.1.1 校園行政管理

根據此研究資料運用到行政電腦化，因影像辨識系統預想應用至校園行政處理上，為使校園行政人員可以正常操作使用，在軟體的操作與資料顯示必須簡單易懂，進而提高作業效率，讓影像辨識系統[1] 成為下個校園系統。

2.1.2 影像辨識系統

在台灣與販賣菸酒須達到指定歲數，利用影像辨識軟體辨識出人臉的身分，拍攝系統能夠利用影像辨識系統[2]來識別人臉是否到達法定年齡，影像系統會與資料庫比對出人臉真偽和提取人臉相關的基本資料來完成確認。

2.1.3 人臉辨識

科技隨著時間的移動快速翻新，電腦影像軟體技術近年來已漸趨成熟，已經可以進行即時辨識人臉系統。在影像辨識上，使用 dilb、opencv 等影像相關軟體[7]在影像畫面中抓取人臉，立即找出人臉並分析特徵[4]，再把特徵存入資料庫系統0。

2.1.4 資料庫

對於創建出學生人臉資料庫步驟，預先建立學生人臉資料庫[6]，連接到學生基本資料，再將學生拍攝的標準檔案照片，取出臉部各特徵數據，經影像基本處理，使用特殊軟體檢測，取出各影像特徵邊線與圖示，經正規化處理，取得正規化特徵數據，以建立特徵資料庫以便之後進行比對查詢相關學生資料。

2.1.5 關於網站建置

網站建置[9]主要在於傳達資訊給使用者，然而在網站中顯示的文字與圖必須讓人一目了然，關於本專題網站的功能主要是連結人臉辨識系統[10]，將資料連接資料庫，顯示資料於網頁中自動更新讓使用者操作知道學生資料與相關資料，在網站中可以尋找與註冊組相關文件，申請各式文件資料、歷年成績單、在學證明等等來加速辦理流程。

2.2 相較其他辨識的優、缺點

現在社會中有許多種類的人體生物辨識如表2.1，像是指紋、虹膜、聲音辨識，像是高級住宅推出虹膜辨識進行身分驗證，或指紋辨識使用在我們日常生活中的手機上等等，不同的辨識都有不同的優、缺點與專題中的人臉辨識進行比較。

表 2.1 人臉辨識相較於虹膜、指紋、聲音辨識系統的優、缺點

優點	1. 可與人事出勤考核系統整合，甚至連有帶口罩也可以識別人臉，可避免代打卡弊端。 2. 辨識時間小於1秒，使用方便簡易快速。 3. 非接觸式系統，辨識距離彈性大，不會有衛生問題。 4. 自動記錄辨識成功及失敗影像與時間，方便查詢提升形象與安全保護。 5. 人臉辨識無法以照片作假，不會受帽子，眼鏡或其他外物影響。 6. 應用面廣泛，人臉辨識的技術已經充斥在生活中，大幅提升生活效率及品質，包括機場出入關、智慧型手機解鎖、上下班打卡、智能ATM、智慧社區住宅、規模較大的公司門禁管理等。
缺點	1. 人臉的五官並不像指紋一成不變，隨著時間、年齡增長，人臉中的特徵都會有些許改變，或表情、整容等等，都有可能影響辨識的結果。(需隨時更新資料) 2. 可能受環境光線角度的影響。(本專題的攝影機只要使用，就會有自動照明的功能) 3. 恐侵犯隱私權，人臉的數據也屬於個人資料，單從一張照片就可知道某人的身分、姓名、年齡、背景以及電話，那麼一個人的隱私將不復存在。(要先經過學生同意，才可以收集人臉相片資料)

2.3 人臉辨識的相關技術

人臉辨識的相關技術，包含以下幾種類型：

1.人臉偵測：透過五官的位置特徵和幾何關係的數據，來判斷此影像是否為一張人臉，其中包括了眉毛、眼睛、鼻子、嘴巴、臉形等器官的特徵，若符合條件的話，則回傳相關的人臉數據。

2.人臉圖像預處理：電腦獲取的原始圖像，可能會受到許多條件限制和環境的干擾，無法直接使用，必須先進行處理，其中包含濾波、均衡化、銳化、人臉對準、灰度校正，人臉圖像增強以及規一化的動作，目的是為了利於電腦做識別。

3.人臉圖像匹配與識別：將輸入的影像與資料庫中已存在的人臉特徵數據進行比對，根據相似程度來進行識別。可分為兩類，一類稱為人臉偵測，是一對一比較的過程，根據相似度來判斷是否為同一個人，會以某個門檻值作為基準，若達門檻值，則結果就會判定為同一人。另一類稱之為辨識數據比對，是一對多進行比較的過程，將輸入的影像與多張人臉特徵數據進行比較，如果找到其中最相似的臉，就會回傳結果來鑑定此人的身分。

4.歐式距離：「歐氏距離」也稱「歐基里德距離」或「歐基里德度量歐式距離」，它是 m 維空間任兩點之間的距離，而人臉辨識是將五官分為多個維度，利用歐式距離計算出得到的解，即可當人臉辨識判斷的因子。

第參章 研究方法

3.1 研究架構

研究架構以圖3.1分為規劃、分析、設計、建置4個階段

1.規劃：

運用人臉辨識在校園行政管理，放置攝像頭於校園處室門口，當有學生進入時，能快速辨識學生身份，並顯示學生身資料於電腦上，提升校園行政速度，使用方便簡易且有效率，非接觸式系統，辨識距離彈性大，不會有衛生問題。

2.分析：

查詢網路上已有的人臉辨識程式碼，以及套件的運用方式，找出人臉特徵位置的座標，將其數據化顯示，以便資料庫內輸入相關資料，讓辨識的速度更快。

3.設計：

使用 Open CV、Dlib 兩種軟體套件將人臉特徵的位置點找出，在使用時會連線到攝影機，將照到的影像與資料庫的圖像比對。

4.建置：

主要是使用 Python 語言來撰寫，使用 Anaconda 內建來創虛擬環境，安裝 Open CV、Dlib 和其他套件，找出人臉特徵點的位置，連線到資料庫進行圖像比對，測試辨識一次所需的時間，進行程式方面的修改，使其達到快速且有效率的工作速度。

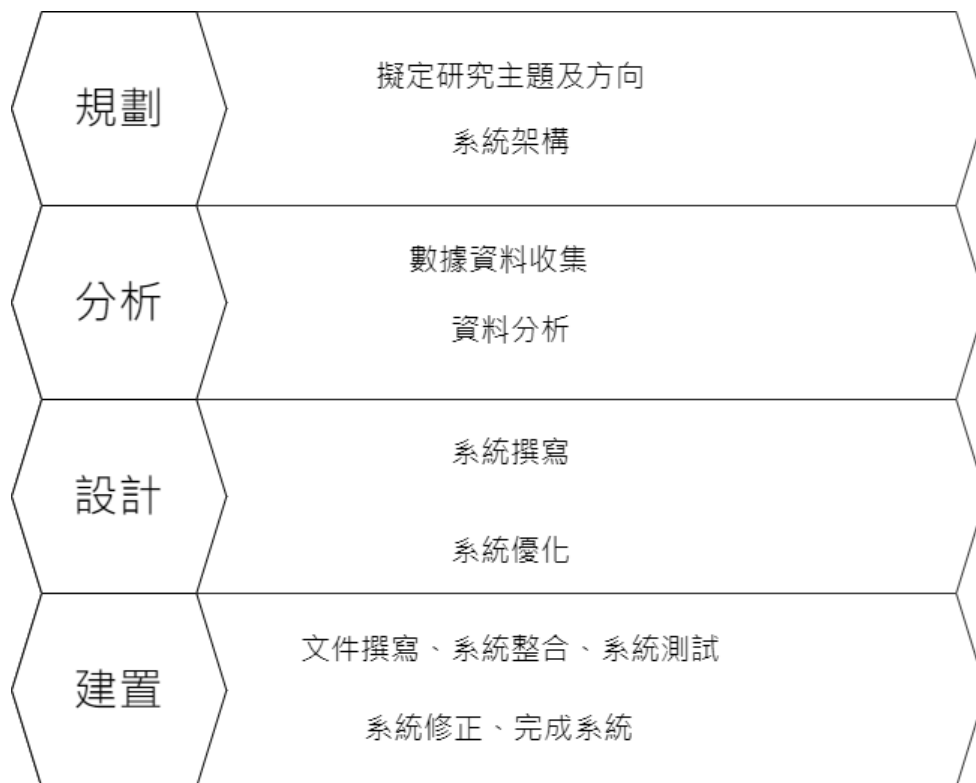


圖 3.1 研究流程圖

3.1.1 硬體系統架構流程圖

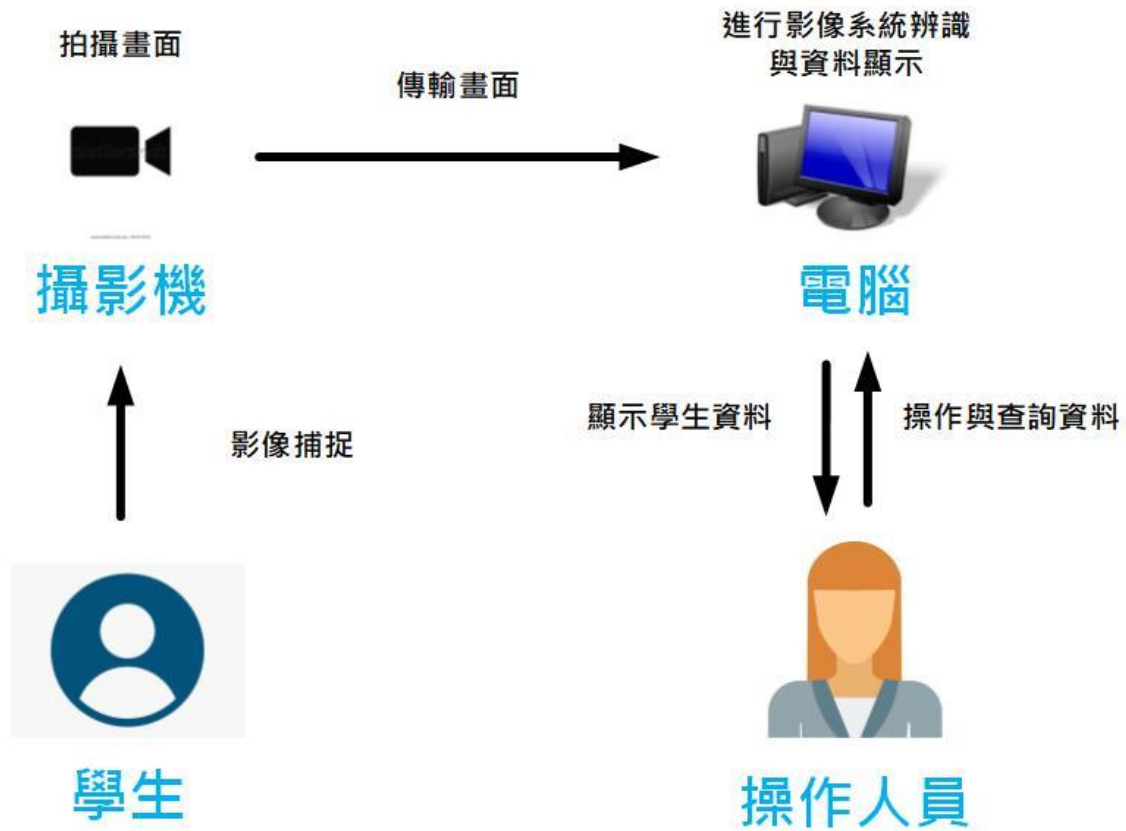


圖 3.2 硬體系統架構流程圖

圖3.2為硬體系統架構流程圖，學生站在攝影機前，將會立即拍攝到人臉，並將截取照片傳回主控電腦，使用電腦已儲存的人臉資料與截取照片進行辨識找尋相關學生，操作人員這時可透過電腦看到學生的基本資料，也可進行操作與查詢資料，學生也可以辦理自己的事務。

3.1.2 軟體系統架構流程圖



圖 3.3 軟體系統架構流程圖

圖3.3為軟體系統架構流程圖，經由主控電腦中的 Anconda 作為架設 Python 虛擬環境的平台，Python 作為主要撰寫程式的語法，My-SQL 建立學生基本資料表以及學期成績資料表資料，XAMPP 中的資料庫與網頁伺服器並進行兩者的連線，傳輸需要資料至網頁中顯示出來，本研究使用兩種人臉辨識的軟體 Open CV 以及 Dlib 進行系統運作，Open CV 將拍到的人臉進行人臉框架鎖定，並使用 Open CV 內部的套件中的哈爾特徵檢測法以及 OpenCV 神經網路檢測法，進行人臉辨識，而 Dlib 是由套件中的演算法 HOG（histogram of oriented gradient）及迴歸樹進行運作鎖定人臉進行人臉特徵提取。

HTML 用來連結開啟辨識到人臉後的網站，php 是在伺服器端執行的指令碼語言，尤其適用於 Web 開發並可嵌入 HTML。

資料庫中的特徵資料，先以新生或在校學生繳交的人臉照片的方式，藉由 Open CV 以及 Dlib 進行特徵提取，並且儲存於資料庫中作為比對之用途，將拍攝系統傳輸至主控電腦所產生的特徵數據與存放本地數據進行比對，若比對度達80以上將輸出名字便可搜尋從資料庫中提取其他相關資料，若沒有將會顯示沒有該名人員，是否將該名人員特徵資料新增至資料庫。

3.1.3 甘特圖

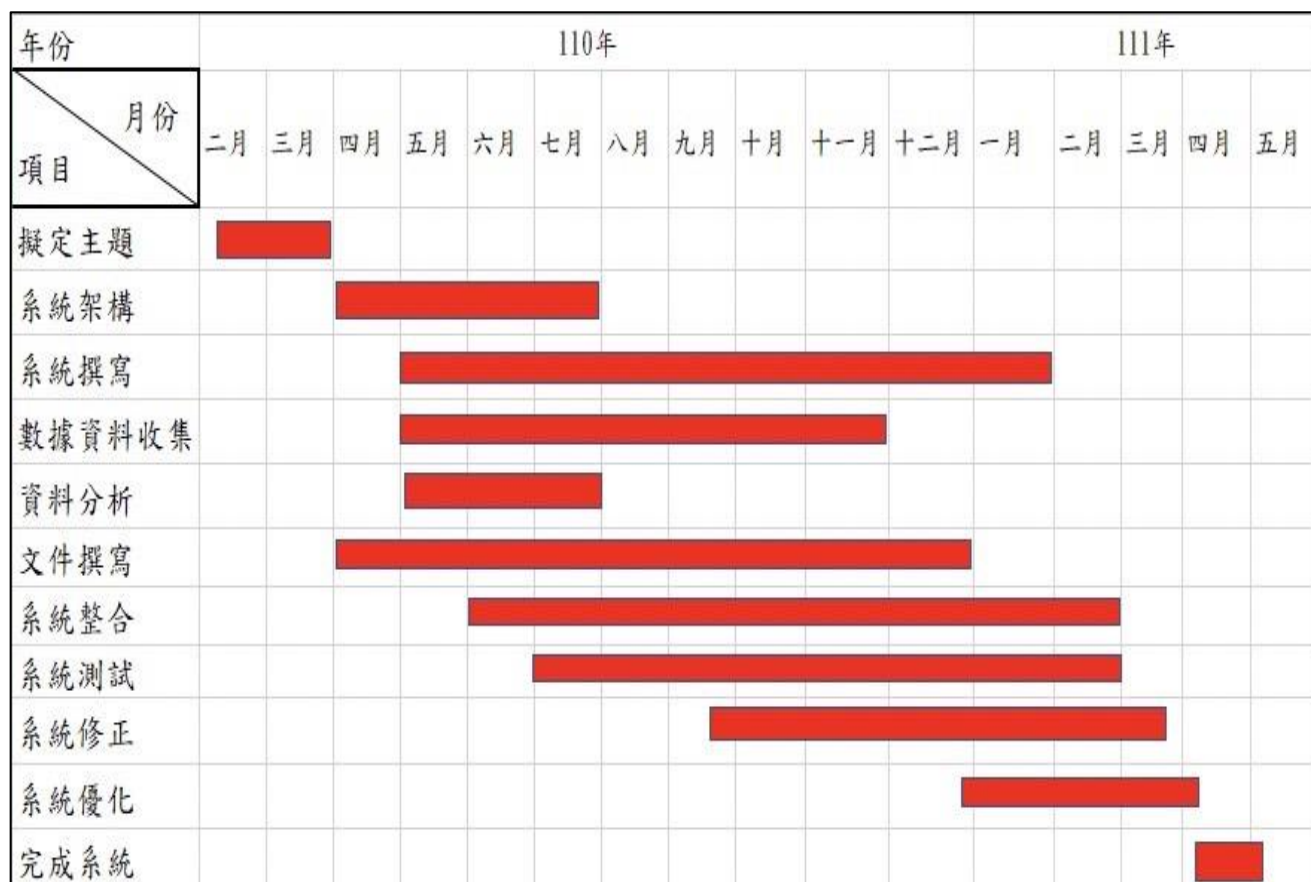


圖 3.4 甘特圖

3.2 研究流程

當本計畫系統開發工具除了基本的電腦硬體與辦公軟體(Office)之外，以及所需要的軟體使用，研究的步驟為以圖3.5與圖3.6作為合併共同說明。

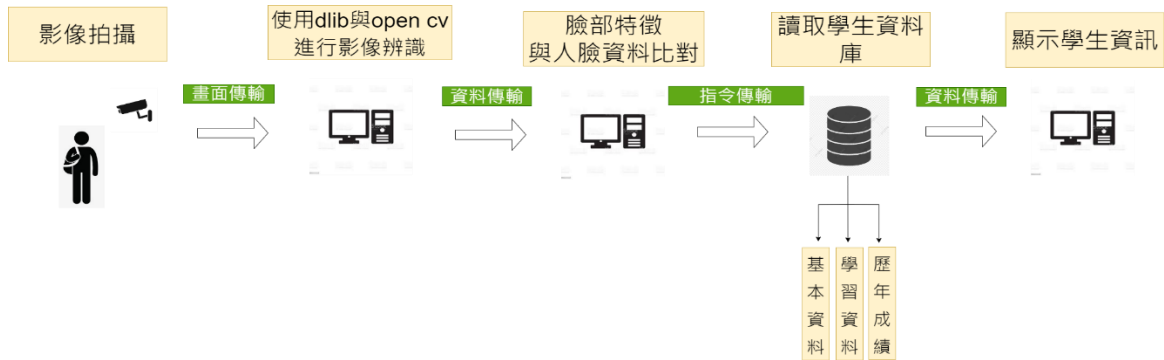


圖 3.5 影像抓取情境圖

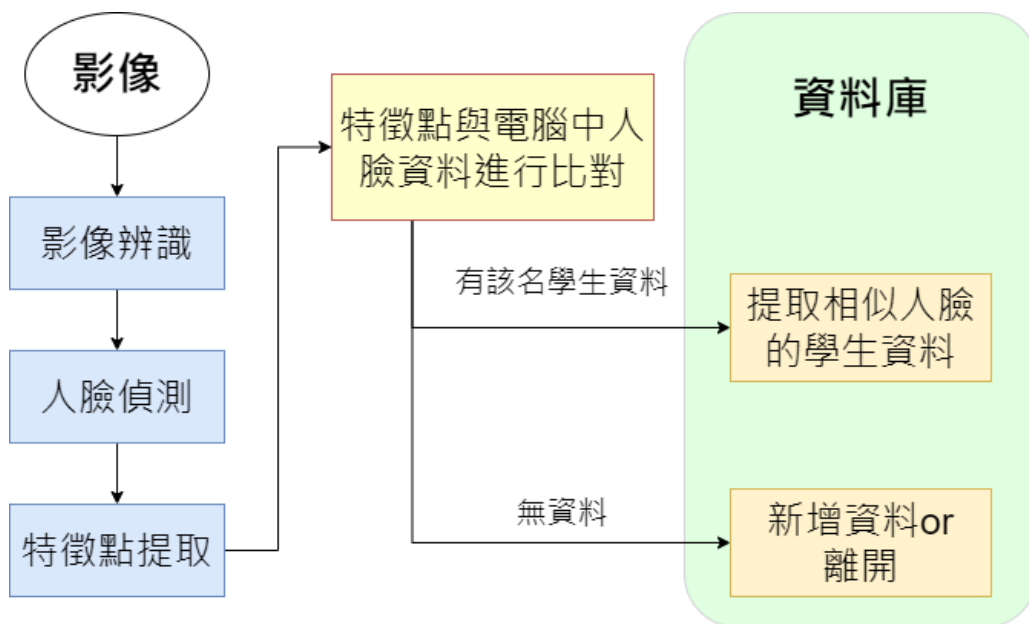


圖 3.6 人臉辨識步驟

1. 每當開學或是註冊時，將會有新生或轉學生入學註冊並將照片上傳。
2. 如圖3.5所示，當新生或轉學生註冊時，所上傳的該名學生照片會先使用 Open CV、Dlib 進行人像偵測以及特徵值抽取，將辨識結果建置為人像特徵資料與目前在校生的人像資料，一同存放置 MySQL 的人像辨識資料庫中。
3. 如圖3.6所示，只要有陌生人士走進外部攝影機的拍攝範圍內，攝影機將進行人像照片的抓取，當影像拍攝資料獲取時，藉由 Open CV、Dlib 將外部拍攝影像資料進行人像特徵提取與內部人臉特徵資料庫的辨識結果，資料進行人臉辨識與特徵值比對。

4. 當影像比對人像偵測以及特徵值高達80%以上時，將從 MySQL 學生基本資料庫中提取該名學生的相關基本資料，輸出至行政人員的電腦中。

5. 若比對結果為 `unknown`，表示人像資料庫中無該名人士的人像特徵的相關資料，將提醒行政人員是否將該名人士資料輸入至人像資料庫當中。

3.3 研究安裝步驟

STEP 1: 首先安裝 anaconda

安裝完後打開命令提示字元，創建虛擬環境輸入：`conda create -n opencv-env python=3.6.8`

STEP 2: 開始 Install OpenCV

(1) 先進入剛創建好的虛擬環境輸入：`conda activate opencv-env`

(2) 安裝套件 接著依序安裝 OpenCV 相依套件

輸入：`pip install numpy scipy matplotlib scikit-learn jupyter`

`pip install opencv-python`

`pip install dlib`

Numpy：基於 Python 的 n 維數值計算擴展。

Imutils：一系列使得 opencv 便利的功能，包括圖像旋轉、縮放、平移，骨架化、邊緣檢測、顯示

matplotlib 圖像（`imutils.opencv2matplotlib(image)`）。

(3) 測試安裝

可使用 Python 直譯器做簡單測試

```
>>>import cv2
```

```
>>>cv.__version__
```

```
>>>import dlib
```

```
>>>dlib.__version__
```

如果看到版本的話代表安裝成功，最後到官網人臉68特徵點模型下載開源數據集

`shape_predictor_68_face_landmarks.dat`，下載完後解壓縮到程式所在根目錄底下即可。

3.4 系統流程

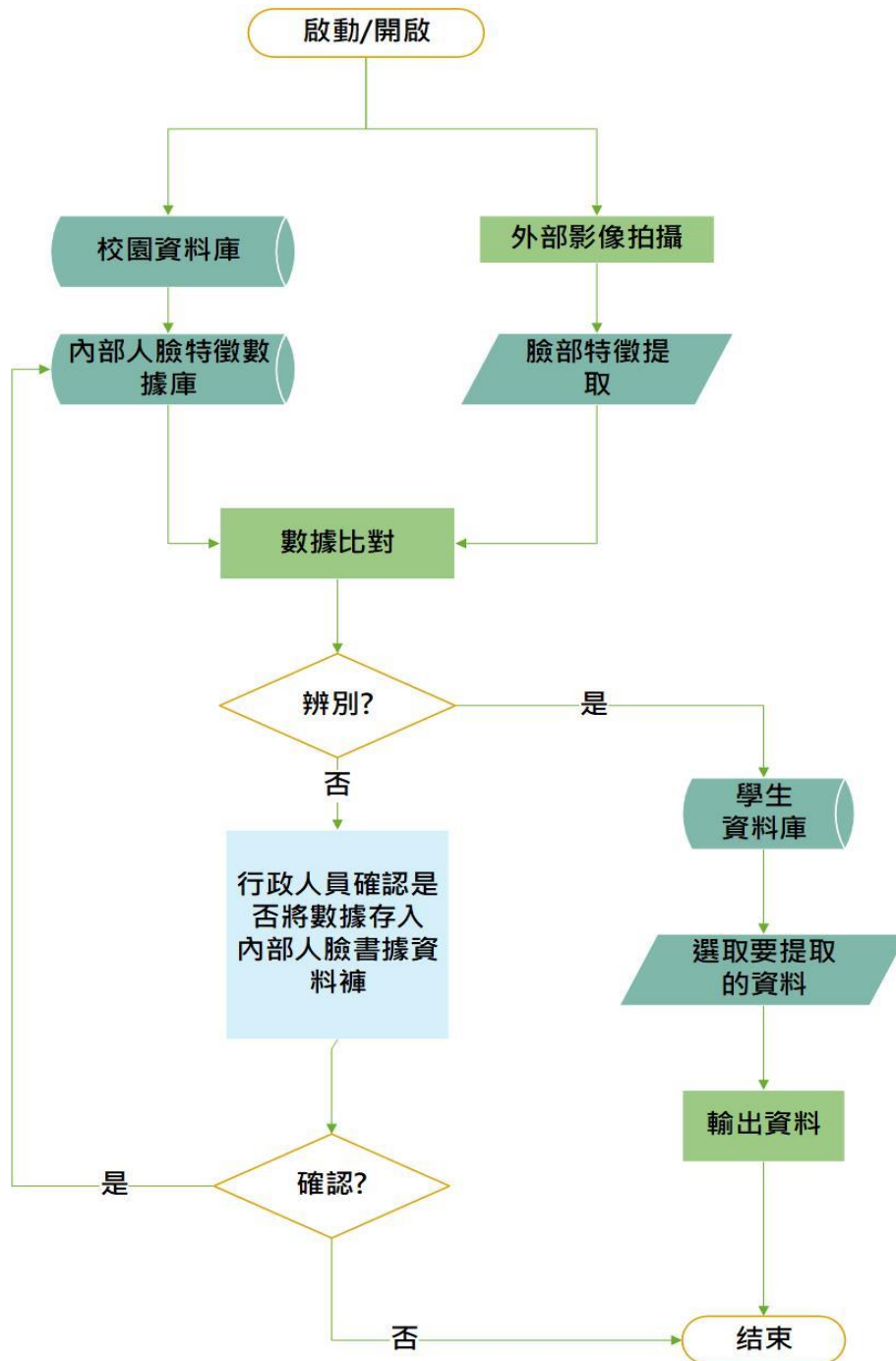


圖 3.7 人臉辨識系統流程圖

以註冊組為例:當學生要進入註冊組辦理業務時如圖3.7，學生面對門口的攝像頭拍攝到人時，並開啟校園資料庫中的內部人臉特徵資料庫，藉由外部攝像頭當有人進入拍攝範圍之後，將會進行該名學生影像拍攝，進行人像的特徵提取，與內部人像資料庫中進行特徵比對判別，比對結果後相似率高達80%以上時，將提取該名

學生在內部人像資料庫中的基本後，會另外開啟一則網頁可下拉式選單，也可供教職人員和學生查看，且詢問是否再提取另一個基本資料庫中的其他相關資料供使用者選取，之後將提取的資料輸出，若是比對後相似率接近於0的時候將會提示說該名人士未有資料再內部人像資料庫中，詢問是否將該名人士的人像特徵資料加入到內部人像資料庫中。

3.5 創建資料庫

學生基本資料(basic)、學期成績資料表(sheet110)、在學生基本資料表中含有姓名、學號、畢業高中、住址、註冊狀態、緊急聯絡人，則學期成績資料表中含有各科目成績、平均分數、班級名次。

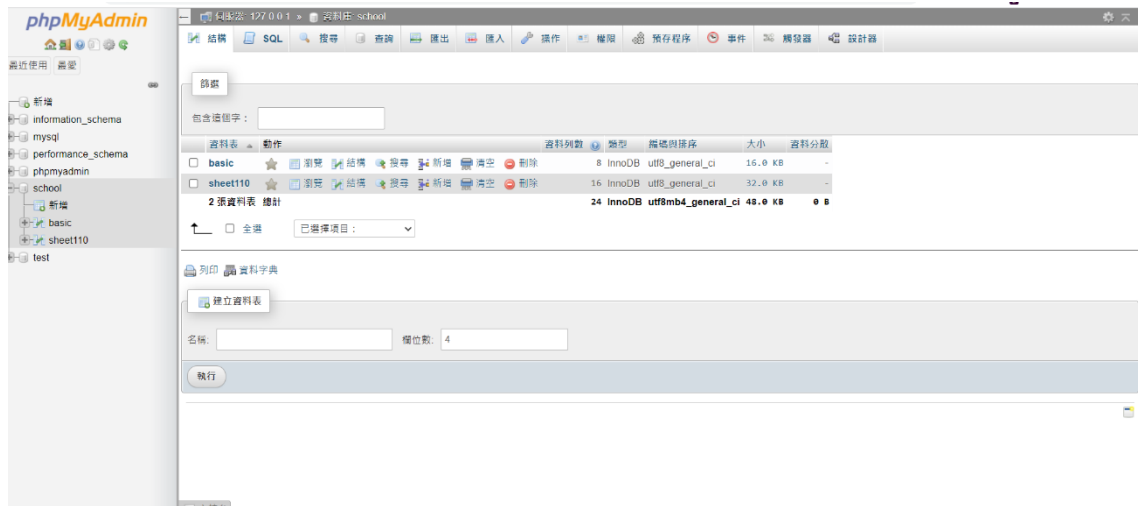


圖 3.8 創建資料庫

第肆章 系統分析與設計

4.1 系統設計

對於本專題訂定的人臉辨識系統，經我們收集資料整理後，所須條件如表4.1系統設計所需的軟、硬體中使用到的功能及用途。

表 4.1 系統設計所需功能及用途

硬、軟體	功能說明	用途
視訊攝影機	接受訊號開啟照相功能	進行人像拍攝
模組一 電腦	存儲程式和控制程式	人臉辨識系統運作
模組二 電腦	存儲程式和控制程式	人臉辨識系統運作
anaconda	語法為 Python 的軟體平台	安裝 Open CV 和 dlib 套件的程式平台，用來執行特徵提取、存取資料庫特徵、人臉比對結果
Python3	做為系統的語法功能	將外部影像與內部資料庫中提取的該人像資料進行特徵比對，以及提取或是輸入資料庫
XAMPP/MySQL	透過關聯式資料庫系統把內部資料進行連結，方便快速查詢，用來整合網頁伺服器，並創建資料庫	建立人像辨識資料庫以及相關基本資料，使用 XAMPP 中的資料庫與網頁伺服器，並進行兩者的連線，傳輸需要資料至網頁中顯示出來
Open CV	運用 LBP 運算元提取圖片特徵，這樣可以獲取整幅影像的 LBP 編碼影像	將攝影機拍攝影像進行辨識轉為人像辨識資料
Dlib	使用 HOG 以及迴歸樹演算法	提供訓練好的模型，運用此套件在電腦上，可以辨識出人臉的68的特徵點
html	來使用在網站上的程式語言，編輯網站畫面與設計排版來顯示學生資料等等	用來連結開啟辨識到人臉後的網站
php	用來收尋以及傳輸資料庫的數據	即“超文字前處理器”，是在伺服器端執行的指令碼語言，尤其適用於 Web 開發並可嵌入 HTML

4.2 系統模擬

以註冊組為例來展示，我們啟動攝像頭放置於距離門前2公尺內，當學生踏進處室時，攝像頭回傳影像至系統自動偵測人臉，於主控電腦內安裝的 Anaconda、Python3、MySQL、Dlib、Open CV 等軟體，進行系統運作提取人臉特徵如圖 4.1，和預先儲存在本地的資料庫進行數據比對，經辨識身分認證後，擷取學生姓名或學號搜尋 MySQL 中的學生資料，供使用者選取相關資料。

效果展示:

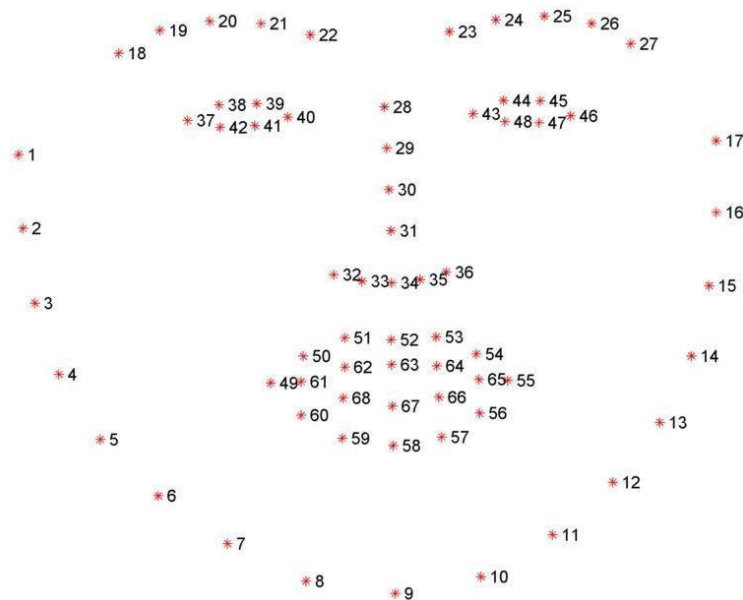


圖 4.1 68點個特徵點

如圖 4.1 為 68 特徵點包括鼻子、眼睛、眉毛，以及嘴巴等，如圖 7 紅點就是偵測出人臉的 68 個特徵點。



圖 4.2 辨識人臉圖片

模組一使用 Open CV 軟體套件進行人臉辨識，圖4.2為 camera 攝影機視窗部分，影像中人臉有被軟體繪製的框架選取，在綠色框架左上角顯示辨識到的人的資訊。

先在 Anaconda 內創設 python 的環境以及安裝套件 OpenCV3(人臉辨識)、pillow(顯示文字)、numpy(有效率地操作多維陣列的函式與運算子)，以及在 C 槽新增一個專門放影像辨識的資料夾、(haarcascade_frontalface_default.xml(人臉識別分類器))，例如:(C:\cv\face)這樣的建立方式，在 face 內新增人臉擷取到的照片(可設定擷取張數)的資料夾、放訓練檔的資料夾。

人臉擷取出的照片檔案位置要標明清楚，程式執行後會先出現文字輸入框，只能先輸入數字(輸入非數字訓練時會出現錯誤)，從0開始輸入，這時就會開始擷取。

下一步是訓練，利用程式碼將存放照片檔案的位置標示出來，程式會自動訓練，訓練完，會在資料夾內新增一個訓練檔。

最後進行人臉辨識，將訓練檔的位置標示出，利用 pillow 套件顯示文字，中文字的 ttf 檔位置校詳細標示出(顯示中文需要的)，辨識出的人臉會輸出到結果以及畫面上。

4.3 模型訓練

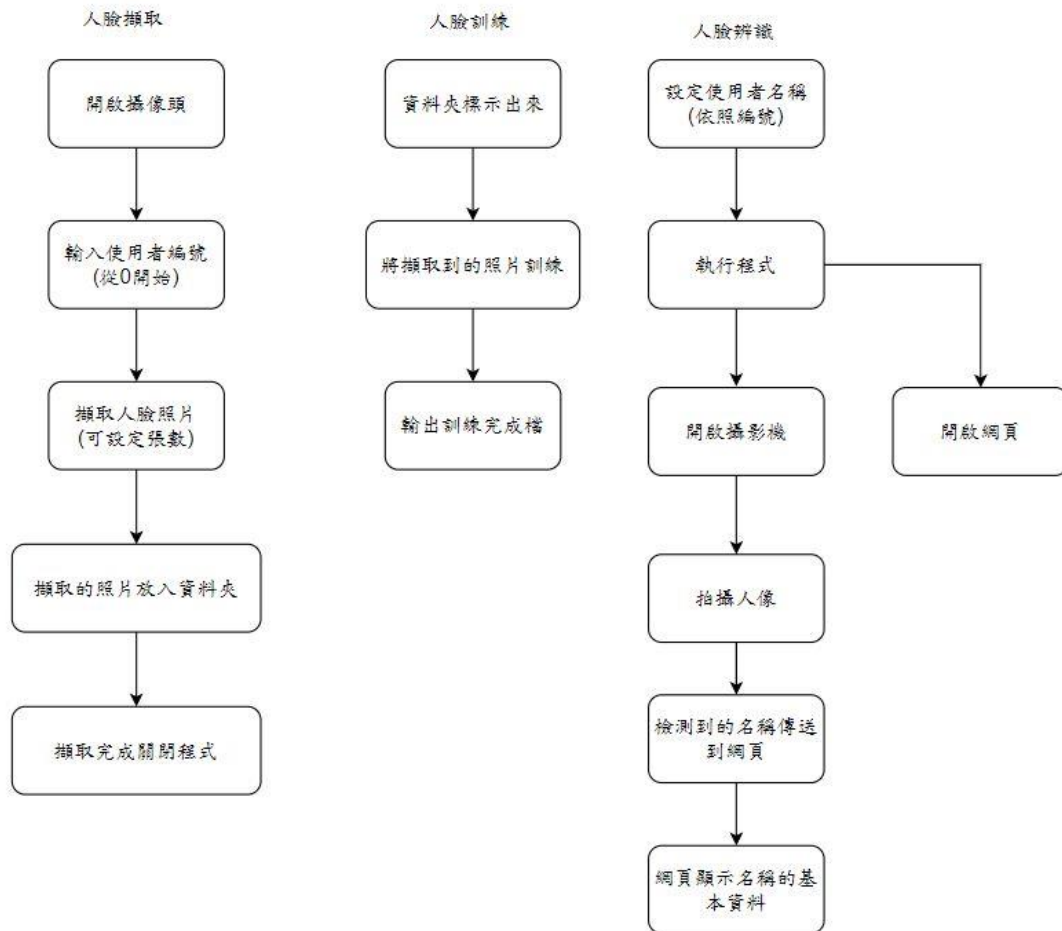


圖 4.3 模型訓練三階段

opencv 程式的執行流程分為三階段，人臉照片擷取、人臉訓練、人臉辨識(可顯示中文)，最主要的是擷取及訓練，是這個程式碼進行人臉辨識最重要的東西

opencv 程式的執行流程分為三階段，人臉照片的擷取、訓練、辨識(可顯示中文)，最主要的是擷取及訓練，是這個程式碼進行人臉辨識最重要的環節

4.4 系統建置硬體

4.4.1 JSING WB1 視訊攝影機 Webcam

(1) 運用攝影機透過即時拍攝或視訊擷取到的人員影像與掃描（且驗證過）的身分證件進行比對，藉此驗證人員身分。

(2) 參照於該網站 <https://24h.pchome.com.tw/prod/DCAS2V-A900B7S6L>

表 4.2 視訊攝影機規格

品 牌:JSING	連接方式:USB 供電
解析度:2560*1440，30fps	供 電:5V
產品尺寸:95*56*70mm	線 長:1.5m
輸出格式:YUV、MJPG、H.264	產品重量:150g
作業系統:WinXP/Win7/Win8/ Win10/MacOS	保 固:一年
用途	進行人像拍攝



圖 4.4 視訊攝影機

4.4.2 電腦

考慮到不同情況，故利用兩台電腦安裝不同的人臉辨識套件做測試，分別為模組套件一和模組套件二。

模組套件一電腦規格：用來安裝 Open CV 套件，來進行人臉辨識應用，結合了人工智慧的技術，讓電腦視覺的效用提升，以影像分析為素材，並透過深度學習訓練，再利用演算法歸納出其關聯來識別出結果，以及架設資料庫。

表 4.3 電腦1規格

產品型號	ASUS X560UD
SSD	256 G
RAM	4G DDR4
顯卡型號	GTX 1050
處理器	Intel® Core™ i5-8250U
用途	人臉辨識系統運作



圖 4.5 電腦1 ASUS X560UD

模組套件二電腦規格:用來安裝 Dlib 套件，來進行人臉辨識應用，結合了人工智慧的技術，讓電腦視覺的效用提升，以影像分析為素材，並透過深度學習訓練，再利用演算法歸納出其關聯來識別出結果。

表 4.4 電腦2規格

產品型號	VivoBookS510U
HDD	1TB
RAM	8.00 GB (7.88 GB 可用)
顯卡型號	NVIDIA GeForce MX150
處理器	Intel® Core™ i5-8250U
用途	人臉辨識系統運作



圖 4.6 電腦2 ASUS VivoBookS510U

4.5 系統建置軟體

表 4.5 軟體設備

軟體	說明
Anaconda	架設 Python 虛擬環境並安裝 Open CV 和 Dlib 套件的操作平台，用來執行特徵提取、存取資料庫特徵、人臉比對結果。
Python3	將外部影像與內部資料庫中提取的該人像資料進行特徵比對，以及提取或是輸入資料庫。
MySQL	建立學生基本資料和學期成績資料表的虛擬資料。
XAMPP	使用 XAMPP 軟體架設網站與創建資料庫。
Open CV	將攝影機拍攝影像進行辨識轉為人像辨識資料。
Dlib	提供訓練好的模型，運用此套件在電腦上，可以辨識出人臉的68的特徵點。
HTML	使用 HTML 來使用在網站上的程式語言，編輯網站畫面與設計排版來顯示學生資料等等。
PHP	用來收尋以及傳輸資料庫的數據。

4.5.1 Anaconda

Anaconda 作為架設 Python 虛擬環境並安裝 Open CV 和 Dlib 套件的操作平台，為本專題研究系統主體，進行一系列人臉辨識、身分確認、資料提取。



圖 4.7 Anaconda

4.5.2 Python3

主要是進行資料比對的部分，將外部影像與 MySQL 人像資料庫中提取的該人像資料進行特徵比對，若比對相同，將從 MySQL 基本資料庫中提取該名人士的相關資料，如果沒有此學生的資料，會請對方輸入自己的照片到資料庫。



圖 4.8 Python3

4.5.3 MySQL

建立學生基本資料和學期成績資料表的虛擬資料，以便本地比對完人臉並輸出資料後，並搜尋存放在 MySQL 內的相關資料。



圖 4.9 MySQL

4.5.4 XAMPP

XAMPP 是一個可以簡單整合 Apache(網頁伺服器)、PHP(伺服器端語言)、Perl(程式語言)及 MariaDB(資料庫)的軟體包，透過 XAMPP 就可以方便快捷架站。

PHP 主要在後端執行，專題中使用到為資料庫架設的程式與連結網頁伺服器中資料的傳輸連結也是透過 PHP 進行，在資料庫中因為語法的關係會使程式收尋與編排上方便好用。



圖 4.10 XAMPP

4.5.5 Open CV

影像辨識的軟體之一，將攝像頭拍攝到的影像進行辨識出人臉，Open CV 辨識出的影像將會自動捕捉人臉並框出，顯示在連結畫面上。



圖 4.11 Open CV

4.5.6 DLIB

套件中的演算法 HOG 及迴歸樹進行運作，核心原理 HOG 將圖像 Hog 特徵來表示人臉，藉由迴歸樹進行資料分類，進而產生人臉特徵數據。



圖 4.12 Dlib

HOG 定義：

又稱方向梯度直方圖（Histogram of Oriented Gradient，HOG）是一種在計算機視覺和圖像處理中用來進行物體檢測的特徵描述方法（特徵描述方法：就是通過從圖像中提取有用信息和丟棄無關信息來簡化圖像，並將信息運用於算法）。

在一副圖像中，局部目標的表象和形狀（appearance and shape）能夠被梯度或邊緣的方向密度分佈很好地描述。首先將圖像分成小的連通區域，我們把它叫細胞單元。然後採集細胞單元中各像素點的梯度的或邊緣的方向直方圖。最後把這些直方圖組合起來就可以構成特徵描述器。

迴歸樹：一般用於資料的分類，其基本思想是將原資料集按照某些屬性值進行二拆分，有點類似構建二叉查詢樹的過程。樹的葉子節點可能是一個資料點的值，也可能是一些資料點的平均值，這個可通過引數控制。

4.5.7 HTML

現在網頁呈現上都會使用到 HTML 語法進行畫面上的顯示與設計，本專題使用到 html 的部分，針對網頁進行資料庫學生資料的顯示、設計修改、下拉式選項等等事項顯示。



圖 4.13 HTML

4.5.8 PHP

PHP 主要在後端執行，專題中使用到為資料庫架設的程式與連結網頁伺服器中資料的傳輸連結也是透過 PHP 進行，在資料庫中因為語法的關係會使程式收尋與編排上方便好用。

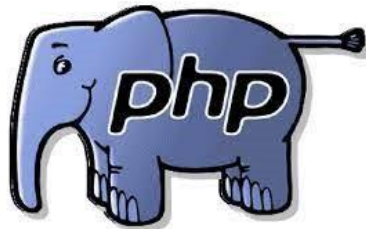


圖 4.14 PHP

4.6 人臉影像辨識套件比較

本專題人臉影像辨識分別有模組一 Open cv 套件與模組二 Dlib 套件，我們使用兩個辨識套件進程式製作，在完成後進行優、缺點的比較。

4.6.1 模組套件一

模組程式 Open cv 套件電腦一：如圖4.14首先到電腦前擷取該名學生的100張~500張照片，在經由電腦訓練，最後在進行人臉影像辨識。

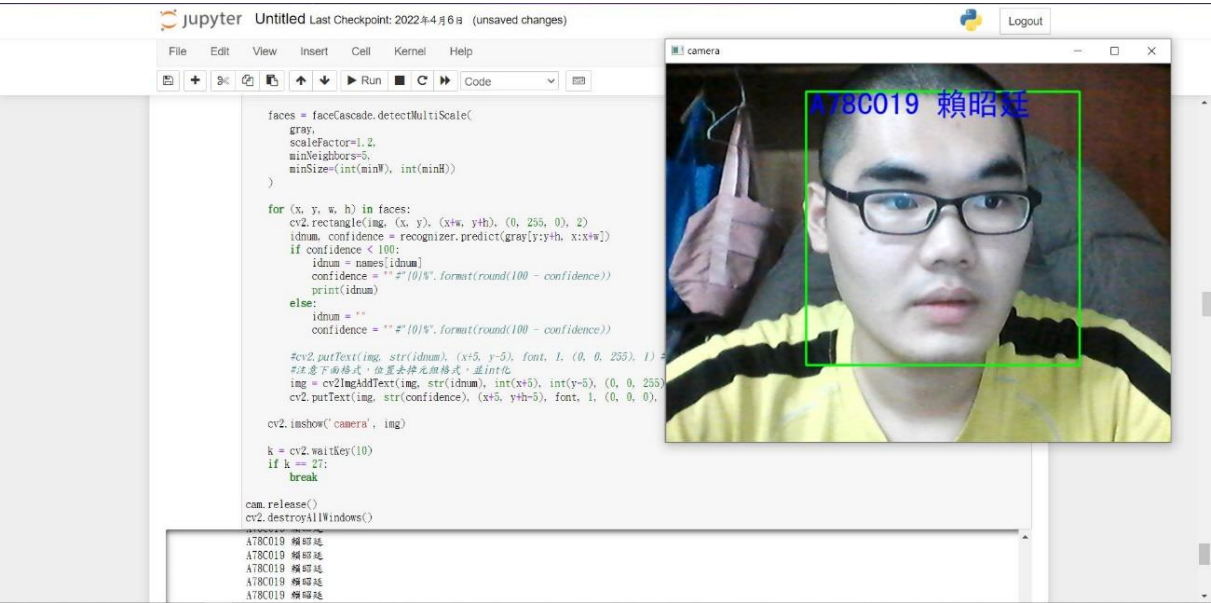


圖 4.15 模擬程式套件一執行圖

優點：

- (1) Anaconda 有 OpenCV 套件可安裝，不須在網路上尋找套件下載資源
- (2) 收集人臉的不同方向照片進行訓練
- (3) 可以檢測不同比例的人臉

缺點：

- (1) 每人需要先擷取多張(100張以上)的照片才可做訓練
- (2) 在極端的角度下無法確認是否有人臉
- (3) 訓練、擷取比 Dlib 的程式花費更多時間

表4.6為沒有戴口罩的情況下所做的測試表，攝影機擷取圖片的張數、擷取人臉時間、訓練人臉時間以及信心度。

表 4.6 沒有戴口罩下擷取和辨識人臉測試表

擷取圖片/項目	擷取人臉時間	訓練人臉時間	信心度
---------	--------	--------	-----

100張	大約1分鐘，4個人	大約20秒，4個人	大約30~40% (容易辨識錯誤姓名)
200張	大約1分鐘30秒，4個人	大約30秒，4個人	大約40~50% (要特定角度才會辨識到姓名)
300張	大約2分鐘，4個人	大約1分鐘，4個人	大約50~60%
400張	大約2分鐘30秒，4個人	大約1分鐘20秒，4個人	大約50~63%
500張	大約3分鐘，4個人	大約1分鐘20秒，4個人	大約52~65% (容易辨識錯誤)

PS:會因辨識的距離遠近造成信心度增加或減少，有的人戴口罩

表4.7為有戴口罩的情況下所做的測試表，攝影機擷取圖片的張數、擷取人臉時間、訓練人臉時間以及信心度。

表 4.7 有戴口罩下擷取和辨識人臉測試表

擷取圖片/項目	擷取人臉時間	訓練人臉時間	信心度
100張	大約1分鐘，4個人	大約20秒，4個人	大約30~40% (容易辨識錯誤姓名)
200張	大約1分鐘30秒，4個人	大約30秒，4個人	大約40~50% (要特定角度才會辨識到姓名)
300張	大約2分鐘，4個人	大約1分鐘，4個人	大約50~60%
400張	大約2分鐘30秒，4個人	大約1分鐘20秒，4個人	大約65~70%
500張	大約3分鐘，4個人	大約1分鐘20秒，4個人	大約52~65% (容易辨識錯誤)

4.6.2 模擬程式套件二

模組程式 Dlib 套件電腦一：如圖4.15Dlib 套件可需使用1張人臉正面照並使用人臉模組訓練套件，就可進行人臉辨識。

優點: (1)可以在 CPU 上實時工作

(2)簡單的架構

(3)可以檢測不同比例的人臉

(4)相較於範例1程式碼，每人只需匯入一張大頭貼照片即可做人臉特徵擷取、訓練

(5)在訓練時間比較快速節省時間，準確度也辨識的很精準

缺點: (1)相較於模組套件一較不耗時間，也因筆電規格配備，而影響筆電的效能

(2) 會出現大量的把非人臉預測為人臉的情況

(3) 不適用於非正面人臉影像

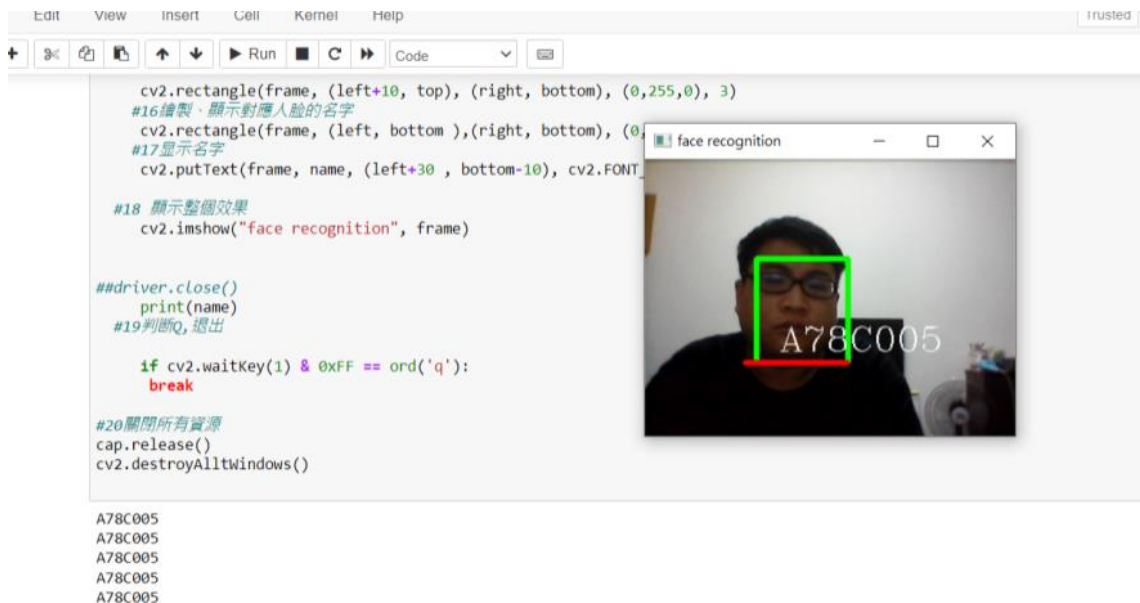


圖 4.16 模擬程式套件二執行圖

4.7 網站架構

網站結合了學生資料庫與影像辨識，影像辨識結合校園行政系統，人臉辨識可以迅速的辨識人臉影像，並傳輸人臉資料到網站登入，進入網頁後將會由人臉資料顯示在網頁中，網站首頁包含了學生的基本資料，基本資料有姓名、性別、班級、學號、座號、註冊狀態，還有下拉是選項歷年成績與在學證明連接雲端資料庫，運用人臉辨識系統應用於校園行政作業，以提升學校行政效率，縮短時間。

學生資料查詢

← → ↺ ⌂ localhost/1/1.php

註冊組 請選擇查尋資料

學生資料(大類別):

相關資料(小類別):

學生資料

姓名	性別	班級	學號	電話	戶籍地	地址	註冊狀態	家長姓名	家長電話	畢業高中
劉至城	男	資管4A	A78C054	0968248267	臺中市	中區綠川里臺灣大道一段1號	未註冊	劉爸爸	0981956709	台中一中

Sat Apr 23 2022 22:18:06 GMT+0800 (台北標準時間)

圖 4.17 人臉辨識後輸出至網站的介面

4.7.1 E-R model 圖

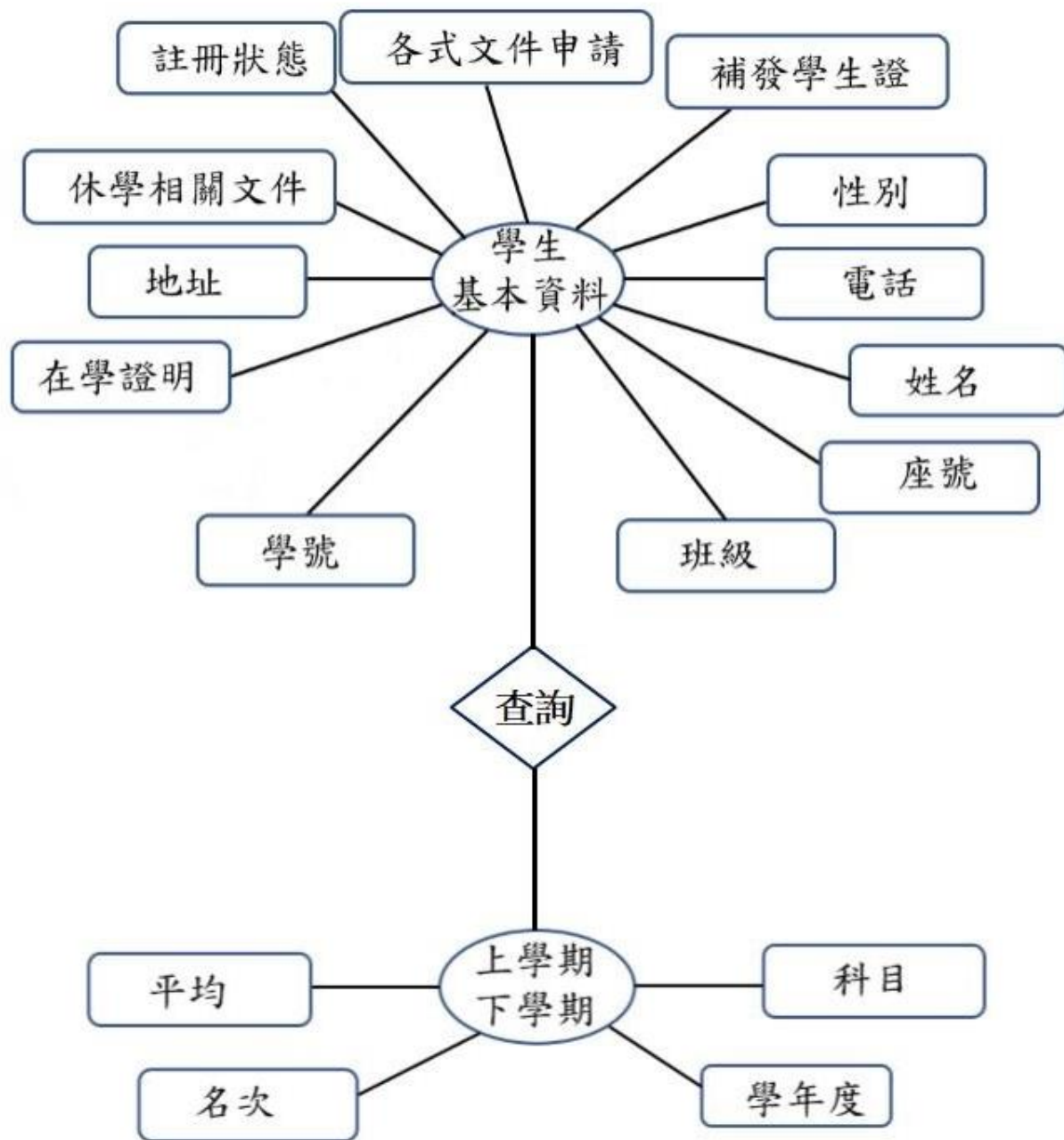


圖 4.18 E-R model

4.7.2 關聯式資料表

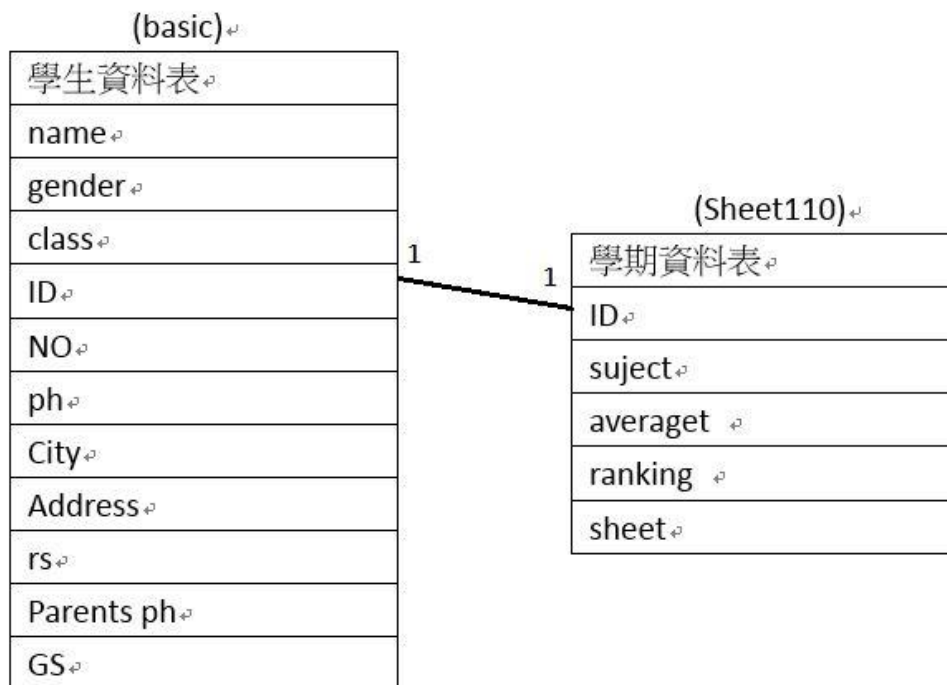


圖 4.19 關聯式資料表圖

4.7.3 資料表欄位

學生資料表

表 4.8 學生資料表

欄位	資料型態	限制	中文備註
name	varchar(10)	NOT NULL	姓名
gender	varchar(10)	NOT NULL	性別
class	varchar(10)	NOT NULL	班級
ID	varchar(10)	PRIMARY KEY	學號
NO	Int(11)	NOT NULL	座號
ph	varchar(10)	NOT NULL	電話
City	varchar(10)	NOT NULL	縣市
Address	varchar(10)	NOT NULL	地址
rs(registration status)	varchar(10)	NOT NULL	註冊狀態
parents	varchar(10)	NOT NULL	家長
parents ph	varchar(10)	NOT NULL	家長電話
GS	varchar(10)	NOT NULL	畢業高中

學期成績資料表

表 4.9 學期成績資料表

欄位	資料型態	限制	中文備註
ID	varchar(10)	PRIMARY KEY	學號
Subject	varchar(10)	NOT NULL	科目
Average	varchar(10)	NOT NULL	平均
Ranking	varchar(10)	NOT NULL	名次
Sheet	varchar(10)	NOT NULL	學年度

4.8 功能展示

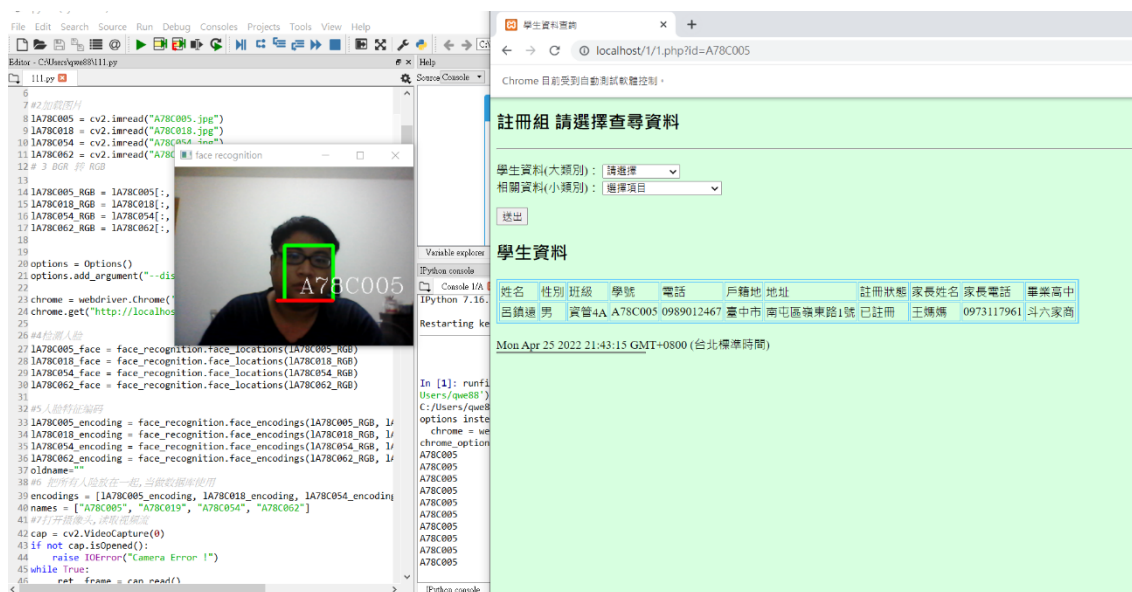


圖 4.20 成果展示圖

第伍章 結論與未來展望

5.1 結論

現在科技每天都不斷地在進步，本專題達到以下目的有，透過影像辨識系統找出影像中的人臉及人名，和設計出使用於校園行政註冊組影像辨識之流程，且透過影像辨識軟體比對人臉資料，確認身分後連結資料庫抓取學生資料，在讓學生辦理相關行政業務更快速與有效率。

5.2 未來展望

人臉辨識的出現，取代了傳統的簽到打卡、現金付款、解鎖，現在生活中處處可見人臉辨識的應用。人臉辨識的技術不僅提高了效率，也結合各個產業和領域大幅提升人類的生活品質，帶來不少的經濟價值。如有科技突破或新技術出現，問題也必然隨之而來，其中照片和真人的分辨，目前仍有部分的技術尚未完全克服，隱私權的問題也是人臉辨識領域的一大挑戰。未來展望是能透過加強演算法，降低相片樣本數量而達到更高的辨識準確度。另外，進一步加強資訊安全，讓人們在人臉辨識的使用上能夠更安心、更信任這項技術。

假設辨識到的人沒有在資料庫，就會啟動補充方案新增人臉辨識的資料。

參考文獻

- [1] 張國保(1992)，大學總務行政電腦化之研究。國立政治大學教育研究所碩士論文。
https://ah.nccu.edu.tw/item?item_id=101797
- [2] 李政昕(2020)，應用於自動販賣機之影像辨識系統。中興大學電機工程學系所學位論，020 年 (2020 / 01 / 01) ， P1 - 26
<https://hdl.handle.net/11296/23z6x6>
- [3] 王磊，圖片人臉檢測—Dlib 版（四）;2018/04/27
<https://iter01.com/70063.html>
- [4] 吳明芳、炳聰、育誠 (2007)，掌形特徵於身份確認上的應用。Application of palm characteristics to identity confirmation；崑山科技大學電機工程研究所學位論文，2007 年(2007 / 07 / 30)，P53~68
http://ir.lib.ksu.edu.tw/bitstream/987654321/12212/2/6.p53~p68.A20_%E6%8E%8C%E5%BD%A2%E7%89%B9%E5%BE%B5%E6%96%BC%E8%BA%AB%E4%BB%BD%E7%A2%BA%E8%AA%8D%E4%B8%8A%E7%9A%84%E6%87%89%E7%94%A8.pdf
- [5] 林俊傑(2012)，即時多人臉辨識之影像資料庫系統
Real Time Multiple Face Recognition System on Image Database
中興大學資訊網路多媒體研究所學位論文，2012 年 (2012 / 01 / 01) ， P1 - 58
<https://www.airitilibrary.com/Publication/alDetailedMesh?docid=U0005-2305201222443600>
- [6] 呂芳懌(Fang-Yie Leu)、少棠(Shau-Tang Yu)、閔雄(Min-Shiung Shiu)、晁立(Chau-Li Lin)，影像資料庫在人臉辨識上的應用
東海科學期， (1999 / 07 / 01) ， P103 - 122
<https://www.airitilibrary.com/Publication/alDetailedMesh?docid=10238719-199907-201107090011-201107090011-103-122>
- [7] 張鈞名，基於 python 語言使用 OpenCV 搭配 dlib 實作人臉偵測與辨識，2018/10/15 09:00:00
<https://www.tpisoftware.com/tpu/articleDetails/950>
- [8] 林珮雯(Peiwen Lin)，電子商務網站介面設計研究－美感與使用性(The Study of E-commerce Website Interface Design - Aesthetics v.s. Usability)，2005
bin/g32/g3web.cgi?randomimg=F3KoPc_1649600428&validpath=%2Ftmp%2F%5Enclcdr_doschk%2FF3KoPc_1649600428_NTEzNzYx&validinput=511761&check=%E7%A2%BA%E5%AE%9A

- [9] 陳珮汝，網站設計原則運用於城市行銷網站之廣告效果研究；台中教育大學數位內容科技學系碩士在職專班碩士論文，2012/ 1/1，P1 – P2
<http://ntcuir.ntcu.edu.tw/bitstream/987654321/3932/1/099NTCTC620030-001.pdf>
- [10] 人臉識別系列二，FisherFace，LBPH 演算法及 Dlib 人臉檢測，2019/ 12 /9
<https://codingnote.cc/zh-tw/p/35918/>
- [11] 陳泓亦，XAMPP 安裝及介紹 - 前言，2022/1/16
<https://hackmd.io/@cmrdb/SJQC1GlaY>

附錄

模擬程式 Open CV 套件一程式碼:

程式碼分三階段人臉擷取、訓練、辨識

第一階段 人臉數據收集

```
import cv2
import os
# 調用筆電內置攝影機，所以參數為0，如果有其他的攝影機可以調整參數為1，2
cap = cv2.VideoCapture(0)
pathf = r'c:\cv\face\haarcascade_frontalface_default.xml'
face_detector = cv2.CascadeClassifier(pathf)

face_id = input('\n enter user id:')
print('\n Initializing face capture. Look at the camera and wait ...')
count = 0

while True:
    # 從攝影機擷取圖片
    success, img = cap.read()
    # 轉為灰階圖片
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    # 檢測人臉
    faces = face_detector.detectMultiScale(gray, 1.3, 5)
    for (x, y, w, h) in faces:
        cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0))
        count += 1
        # 保存圖像
        cv2.imwrite("C:/cv/face/Facedata/User." + str(face_id) + '.' + str(count) + '.jpg',
            gray[y:y+h, x:x+w])
        cv2.imshow('image', img)
        # 保持畫面的持續。
        k = cv2.waitKey(1)
        if k == 27: # 通過 esc 鍵退出影像擷取
            break
        elif count >= 1000: # 得到1000個樣本後退出影像擷取
            break

# 關閉攝影機
cap.release()
cv2.destroyAllWindows()
```

第二階段 人臉數據訓練

```
import numpy as np
from PIL import Image #需安裝 Pillow 安裝指令為 pip install Pillow
import os
import cv2
```

```

# 人臉數據路徑
path = 'C:/cv/face/Facedata'
pathf = r'c:\cv\face\haarcascade_frontalface_default.xml'

recognizer = cv2.face.LBPHFaceRecognizer_create()
detector = cv2.CascadeClassifier(pathf)

def getImagesAndLabels(path):
    imagePaths = [os.path.join(path, f) for f in os.listdir(path)] # join 函數的作用？
    faceSamples = []
    ids = []
    for imagePath in imagePaths:
        PIL_img = Image.open(imagePath).convert('L') # convert it to grayscale
        img_numpy = np.array(PIL_img, 'uint8')
        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)
        for (x, y, w, h) in faces:
            faceSamples.append(img_numpy[y:y + h, x: x + w])
            ids.append(id)
    return faceSamples, ids

print("Training faces. It will take a few seconds. Wait ...")
faces, ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

recognizer.write(r'face_trainer\trainer.yml')
print("{0} faces trained. Exiting Program".format(len(np.unique(ids))))

```

第三階段人臉檢測(可顯示中文)

```

# -*- coding: utf-8 -*-
"""
Created on Sun Apr 17 20:51:06 2022

@author: user
"""

from flask import Flask, render_template, render_template_string, Response
#face_recognition 人臉檢測，PIL 法顯示中文人名
import cv2
#---增加的 PIL 法顯示中文---
import numpy
from PIL import Image, ImageDraw, ImageFont
import multiprocessing
import threading
import time
import os

```

```

from selenium import webdriver
from selenium.webdriver.chrome.options import Options

#定義一個函數
def cv2ImgAddText(img, text, left, top, textColor=(0, 255, 0), textSize=40):
    if (isinstance(img, numpy.ndarray)): # 判斷是否 OpenCV 圖片類型
        img = Image.fromarray(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
    # 創建一個可以在給定圖像上繪圖的對象
    draw = ImageDraw.Draw(img)
    # 字體的格式，自己下載華文仿宋字體，放在根目錄下
    fontStyle = ImageFont.truetype(
        "C://SimHei.ttf", 40, encoding="utf-8")
    # 繪製文本
    draw.text((left, top), text, textColor, font=fontStyle)
    # 轉換回 OpenCV 格式
    return cv2.cvtColor(numpy.asarray(img), cv2.COLOR_RGB2BGR)

options = Options()
options.add_argument("--disable-notifications")

chrome = webdriver.Chrome("c:\\chromedriver.exe", chrome_options=options)
chrome.get("http://127.0.0.1/1.php")

cam = cv2.VideoCapture(0)

pathf = r'c:\cv\face\haarcascade_frontalface_default.xml'
recognizer = cv2.face.LBPHFaceRecognizer_create()

recognizer.read(r'c:\cv\face\face_trainer\f500.yml')

cascadePath = pathf
faceCascade = cv2.CascadeClassifier(cascadePath)
font = cv2.FONT_HERSHEY_SIMPLEX
#這裡為0或者1都沒有關係
idnum = 0

names = ['A78C019 賴昭廷', 'A78C062 許家誠', 'A78C054 劉至城', 'A78C005 呂鎮遠']
key = ['A78C019', 'A78C062', 'A78C054', 'A78C005']
#names 中存儲人的名字，若該人 id 為0則他的名字在第一位，id 位1則排在第二位，以此類推

```

```

minW = 0.1*cam.get(3)
minH = 0.1*cam.get(4)
def fun1():
    oldname=""
    while True:
        ret, img = cam.read()
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

        faces = faceCascade.detectMultiScale(
            gray,
            scaleFactor=1.2,
            minNeighbors=5,
            minSize=(int(minW), int(minH))
        )

        for (x, y, w, h) in faces:
            cv2.rectangle(img, (x, y), (x+w, y+h), (0, 255, 0), 2)
            idnum, confidence = recognizer.predict(gray[y:y+h, x:x+w])
            if confidence < 100:
                idnums = key[idnum]
                idnum = names[idnum]
                print(idnums)

                if (oldname != idnums):
                    urls="http://127.0.0.1/1.php?id="+idnums
                    chrome.get(urls)
                    oldname=idnums

                confidence = ""#{0}%".format(round(100 - confidence))
            else:
                idnum = ""
                confidence = ""#{0}%".format(round(100 - confidence))
            #cv2.putText(img, str(idnum), (x+5, y-5), font, 1, (0, 0, 255), 1) #不能顯
示中文
            #注意下面格式，位置去掉元組格式，並 int 化
            img = cv2.ImgAddText(img, str(idnum), int(x+5), int(y-5), (0, 0, 255),20)#
顯示為中文 PIL 法
            cv2.putText(img, str(confidence), (x+5, y+h-5), font, 1, (0, 0, 0), 1)

        cv2.imshow('camera', img)

        k = cv2.waitKey(10)
        if k == 27:
            break

    cam.release()
    cv2.destroyAllWindows()

def fun2():

```

```

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('ttt.html',
                           title=idnum)

#if __name__ == '__main__':
#    app.run()

if __name__ == "__main__":
    app.run(host='0.0.0.0')

if __name__ == '__main__':
    sing_thread = threading.Thread(target=fun1)
    song_thread = threading.Thread(target=fun2)

    sing_thread.start()
    song_thread.start()
模拟程式 Dlib 套件二程式碼:
import dlib
import numpy as np
import face_recognition
from selenium import webdriver
from selenium.webdriver.chrome.options import Options

#2 加载图片
lA78C005 = cv2.imread("A78C005.jpg")
lA78C018 = cv2.imread("A78C018.jpg")
lA78C054 = cv2.imread("A78C054.jpg")
lA78C062 = cv2.imread("A78C062.jpg")
# 3 BGR 转 RGB

lA78C005_RGB = lA78C005[:, :, ::-1]
lA78C018_RGB = lA78C018[:, :, ::-1]
lA78C054_RGB = lA78C054[:, :, ::-1]
lA78C062_RGB = lA78C062[:, :, ::-1]

options = Options()
options.add_argument("--disable-notifications")

chrome = webdriver.Chrome('c:\\chromedriver.exe', chrome_options=options)
chrome.get("http://localhost/1/1.php")

#4 检测人脸
lA78C005_face = face_recognition.face_locations(lA78C005_RGB)
lA78C018_face = face_recognition.face_locations(lA78C018_RGB)
lA78C054_face = face_recognition.face_locations(lA78C054_RGB)

```

```

1A78C062_face = face_recognition.face_locations(1A78C062_RGB)

#5 人脸特征编码
1A78C005_encoding = face_recognition.face_encodings(1A78C005_RGB,
1A78C005_face)[0]
1A78C018_encoding = face_recognition.face_encodings(1A78C018_RGB,
1A78C018_face)[0]
1A78C054_encoding = face_recognition.face_encodings(1A78C054_RGB,
1A78C054_face)[0]
1A78C062_encoding = face_recognition.face_encodings(1A78C062_RGB,
1A78C062_face)[0]
oldname=""
#6 把所有人脸放在一起,当做数据库使用
encodings = [1A78C005_encoding, 1A78C018_encoding, 1A78C054_encoding,
1A78C062_encoding]
names = ["A78C005", "A78C019", "A78C054", "A78C062"]
#7 打开摄像头,读取视频流
cap = cv2.VideoCapture(0)
if not cap.isOpened():
    raise IOError("Camera Error !")
while True:
    ret, frame = cap.read()
    frame = cv2.resize(frame, (0, 0), fx=0.5, fy=0.5)
    #8 BGR 传 RGB

    frame_RGB = frame[:, :, ::-1]
    #9 人脸检测
    faces_locations = face_recognition.face_locations(frame_RGB)
    #10 人脸特征编码
    faces_encodings = face_recognition.face_encodings(frame_RGB, faces_locations)
    #11 与数据库中的所有人脸进行匹配
    for (top, right, bottom, left), face_encoding in zip(faces_locations,
faces_encodings):
        #12 进行匹配
        matches = face_recognition.compare_faces(encodings, face_encoding)
        #13 计算距离
        distances = face_recognition.face_distance(encodings, face_encoding)
        min_distance_index= np.argmin(distances) # 0, 1, 2
        #14 判断:如果匹配,获取名字
        name = "Unknown"
        if matches[min_distance_index]:
            name = names[min_distance_index]

    #15 绘制人脸矩形框

    cv2.rectangle(frame, (left+10, top), (right, bottom), (0,255,0), 3)
    #16 绘制、显示对应人脸的名字
    cv2.rectangle(frame, (left, bottom ),(right, bottom), (0,0,255), 3)

```

```

#17 显示名字
    cv2.putText(frame, name, (left+30, bottom-10),
cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 1)
#18 显示整个效果
    cv2.imshow("face recognition", frame)
    if(oldname !=name):
        urls="http://localhost/1/1.php?id="+name
        chrome.get(urls)
        oldname=name

    print(name)
#19 判断 Q,退出
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
#20 关闭所有资源
cap.release()
cv2.destroyAllWindows()

```

網站程式碼:

```

<?php
//連線資料庫並讀取資料表
$servername = "localhost"; //伺服器連線名
$username = "school"; //資料庫使用者名稱
$password = "school"; //資料庫密碼
$dbname = "school"; //資料庫名

if (isset($_GET["id"]))
    $id=$_GET["id"];
else
    $id="";

$conn = new mysqli($servername, $username, $password, $dbname); //連線資料庫
echo "<table border='2' bordercolor='#66ccff'>";
if (!$conn) {
    die("連線失敗：" . mysqli_connect_error()); //連線資料庫失敗則殺死程序
}
$i=0;
$list_arr=array();
if ($id=="")
    $sql = "select * FROM basic";//傳到這個
else
    $sql = "select * FROM basic WHERE ID='$id'";//傳到這個

$result = mysqli_query($conn, $sql);

```

```

if (mysqli_num_rows($result) > 0) {
    echo "<tr>";
    echo "<td>姓名</td>";
    echo "<td>性別</td>";
    echo "<td>班級</td>";
    echo "<td>學號</td>";
    echo "<td>電話</td>";
    echo "<td>戶籍地</td>";
    echo "<td>地址</td>";
    echo "<td>註冊狀態</td>";
    echo "<td>家長姓名</td>";
    echo "<td>家長電話</td>";
    echo "<td>畢業高中</td>";
    echo "</tr>";

    while ($row = mysqli_fetch_assoc($result)) {
        $i++;
        $array[] = $row;
        echo "<tr>";
        echo "<td>" . $row['name'] . "</td>";
        echo "<td>" . $row['gender'] . "</td>";
        echo "<td>" . $row['class'] . "</td>";
        echo "<td>" . $row['ID'] . "</td>";
        echo "<td>" . $row['ph'] . "</td>";
        echo "<td>" . $row['City'] . "</td>";
        echo "<td>" . $row['Address'] . "</td>";
        echo "<td>" . $row['rs'] . "</td>";
        echo "<td>" . $row['parents'] . "</td>";
        echo "<td>" . $row['parents ph'] . "</td>";
        echo "<td>" . $row['GS'] . "</td>";
        echo "</tr>";
    }

    //echo json_encode($array,JSON_UNESCAPED_UNICODE);
    $a7=json_encode($array,JSON_UNESCAPED_UNICODE);
    //print_r($a7)."</br><br>";

} else {

}

mysqli_close($conn); //關閉資料庫
?>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

```

```

<meta http-equiv="refresh" />
<meta name="viewport" content="width=device-width, user-scalable=no, initial-
scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

<title>學生資料查詢</title>

<link href="a.css" rel="stylesheet" type="text/css">
</head>
<body>
    <meta http-equiv="refresh" content="40">
    <div id="root" style="overflow:-x: auto;">

    <body bgcolor="#D7FFE">
    <figure></figure>
    <h2 align="left ">註冊組 請選擇查尋資料</h2>
    <hr align="center">

    <!-- <nav class="sc-fzomuh gHzJiR"> //這個是查詢學生連結
        <a          aria-current="page"          class="sc-fzAui          eoGDzK          active"
href="https://mykirito.com/">查詢學生</a>
        <a          class="sc-fzqAui          eoGDzk"
href="/https://forum.gamer.com.tw/C.php?bsn=60076&snA=5702507">使用者表<a>
        <a class="sc-fzqMdD fZQoNm">
        </a>
        -->
    <script>

    department=new Array();
    department[0]=["姓名","性別","班級","學號","座號","電話","地址","註冊狀態"]; //
    台北市
    department[1]=["姓名","性別","班級","學號","座號","電話","地址","註冊狀態"]; //
    新北市
    department[2]=["上學期成績","下學期成績"];          // 桃園市
    department[3]=["學生證"];
    department[4]=["在學證明.pdf"];
    department[5]=["在學證明.pdf"];

    function renew(index){
        for(var i=0;i<department[index].length;i++)
            document.myForm.member.options[i]=new Option(department[index][i],
            department[index][i]); // 設定新選項
        document.myForm.member.length=department[index].length; // 刪除多餘的選
        項
    }
    </script>
<script>

```

```

function print_value() {
    <!-- 將 select 的值在印出 -->
    document.getElementById("result").innerHTML
document.getElementById("number").value
}
</script>
<form method="get" name="myForm" target="_self" accept-charset="UTF-8"
action="3.4.php">
    學生資料(大類別) :
    <select name="a" style="width:150px,height:120px"
onChange="renew(this.selectedIndex);">

        <option value="">請選擇</option>
        <option value="basic">學生資料更新</option>
        <option value="sheet110">歷年成績</option>
        <option value="basic">補發學生證</option>
        <option value="basic">註冊狀態</option>
        <!--<option value="basic">補發學生證</option>
        <option value="basic">休學、退學、復學及保留入學資格等之申請</option>
        <option value="basic">項學生學籍管理、登記及報表</option>
        <option value="basic">修業證明書之申請</option>
        <option value="basic">畢業生課程學分之審查</option>
        <option selected="selected"></option>
        -->
    </select>
    <br>

    相關資料(小類別) :
    <select name="member" style="width:160px">
        <option value="">選擇項目</option>
    </select>
    <br>

    <p>
        <input type="submit" name="button" id="button" value="送出" />
    </p>
</form>
<script>
    </script>
<h2 align="left "> 學生資料</h2>
<table width="200" border="1">

    <p>
</body>
    <script language="JavaScript">
function ShowTime(){
    document.getElementById('showbox').innerHTML = new Date();

```

```

        setTimeout('ShowTime()',1000);
    }
</script>
<body onload="ShowTime()">
<div id="showbox"></div>
</body>
</body>
</html>

```

創建成績資料表

```

import mysql.connector

from mysql.connector import Error

MYSQL_HOST = '127.0.0.1'
MYSQL_DB = 'school'
MYSQL_USER = 'school'
MYSQL_PASS = 'school'

def connect_mysql(): #連線資料庫
    global connect,cursor
    connect = mysql.connector.connect(host = MYSQL_HOST, db = MYSQL_DB,
user = MYSQL_USER, password = MYSQL_PASS,
        charset = 'utf8', use_unicode = True)
    cursor = connect.cursor()

    dropdb="drop table if exists basic;"
    cursor.execute(dropdb)

    sqldb="""CREATE TABLE `basic` (
`name` varchar(10) NOT NULL,
`gender` varchar(10) NOT NULL,
`class` varchar(10) NOT NULL,
`ID` varchar(10) NOT NULL,
`NO` int NOT NULL,
`ph` varchar(10) NOT NULL,
`City` varchar(10) NOT NULL,
`Address` varchar(10) NOT NULL,
`rs` varchar(10) NOT NULL,
`parents` varchar(10) NOT NULL,
`parents ph` varchar(10) NOT NULL,
`GS` varchar(10) NOT NULL) ENGINE=InnoDB DEFAULT CHARACTER
SET=utf8"""
    cursor.execute(sqldb)

connect_mysql()

```

創建基本資料表

```
import mysql.connector
```

```
from mysql.connector import Error
```

```
MYSQL_HOST = '127.0.0.1'
```

```
MYSQL_DB = 'school'
```

```
MYSQL_USER = 'school'
```

```
MYSQL_PASS = 'school'
```

```
def connect_mysql(): #連線資料庫
```

```
    global connect,cursor
```

```
    connect = mysql.connector.connect(host = MYSQL_HOST, db = MYSQL_DB,  
user = MYSQL_USER, password = MYSQL_PASS,  
    charset = 'utf8', use_unicode = True)
```

```
    cursor = connect.cursor()
```

```
    dropdb="drop table if exists sheet110;"
```

```
    cursor.execute(dropdb)
```

```
    sqldb=""CREATE TABLE `sheet110` (
```

```
    `ID` varchar(10) NOT NULL,
```

```
    `Chi` varchar(10) NOT NULL,
```

```
    `Eng` varchar(10) NOT NULL,
```

```
    `Maths` varchar(10) NOT NULL,
```

```
    `society` varchar(10) NOT NULL,
```

```
    `nature` varchar(10) NOT NULL,
```

```
    `average` varchar(10) NOT NULL,
```

```
    `ranking` varchar(10) NOT NULL,
```

```
    `sheet` varchar(10) NOT NULL) ENGINE=InnoDB DEFAULT CHARACTER
```

```
SET=utf8""
```

```
    cursor.execute(sqldb)
```