**中 國 文 化 大 學**

**資 訊 工 程 學 系**

**資 訊 系 統 專 題**

**智能衣櫃**

**學 生：黃 懷 萱**

**鍾 亮 節**

**詹 惠 茹**

**指導教授：林 世 崧**

**中 華 民 國 111 年 10 月**

**智能衣櫃**

**專題學生：黃懷萱、鍾亮節、詹惠茹**

**指導教授：林世崧 博士**

**中國文化大學 資訊工程學系**

# 摘要

因應現代技術的自動化，同時為了緩解忙碌的早晨，本研究計畫完成一台智能衣櫃。其中技術項目包含衣物辨識，讓使用者放入衣物至檢視口後不用手動輸入項目細節便可存入衣物資訊，完畢後再自動運轉將衣物整理擺放至位置，只需要將衣物放入衣櫃就可以在面板上查看所有的衣物資訊，再者是使用時會自動抓取天氣資訊並自動計算出當時最適合的衣物組合，透過UI介面選取心儀的衣物組合，接著會由衣櫃內部的馬達運轉自動推送到使用者面前，不需要翻箱倒櫃的搜尋衣服，利用機器來記錄並完成所有事物。

**關鍵詞**：衣物辨識、聯網家電、模擬分析、馬達控制

指導教授 (簽名)

**目 錄**

[摘要 I](#_Toc105401267)

[表 目 錄 IV](#_Toc105401268)

[圖 目 錄 V](#_Toc105401269)

[第 1 章 研究動機與研究問題 1](#_Toc105401270)

[1.1 研究動機 1](#_Toc105401271)

[1.1.1 節省時間 2](#_Toc105401272)

[1.1.2 了解衣物使用狀況 2](#_Toc105401273)

[1.2 研究問題 3](#_Toc105401274)

[1.2.1 圖樣辨識 3](#_Toc105401275)

[1.2.2 衣物穿搭推薦演算法 3](#_Toc105401276)

[1.2.3 UI(User Interface)介面整合 3](#_Toc105401277)

[1.2.4 馬達控制系統 4](#_Toc105401278)

[1.2.5 研究流程 4](#_Toc105401279)

[第 2 章 文獻探討 5](#_Toc105401280)

[第 3 章 研究方法與工作項目 7](#_Toc105401281)

[3.1 各項研究方法 7](#_Toc105401282)

[3.1.1 衣物辨識 7](#_Toc105401283)

[3.1.2 天氣擷取 14](#_Toc105401284)

[3.1.3 天氣演算法 18](#_Toc105401285)

[3.1.4 UI介面 18](#_Toc105401286)

[3.1.5 資料庫規格書 19](#_Toc105401287)

[3.2 系統軟硬體架構之設計與整合 29](#_Toc105401288)

[3.3 操作步驟 30](#_Toc105401289)

[3.3.1 存放 31](#_Toc105401290)

[3.3.2 拿取 32](#_Toc105401291)

[第 4 章 人力配置 34](#_Toc105401292)

[第 5 章 預期結果 35](#_Toc105401293)

[第 6 章 執行進度 36](#_Toc105401294)

[參考文獻 37](#_Toc105401295)

# 表 目 錄

[表3.1 衣服的種類及訓練樣本 8](#_Toc105332813)

[表3.2 依HSV辨識出來的顏色表 10](#_Toc105332814)

[表3.3 clothes\_information的table欄位 20](#_Toc105332815)

[表3.4 weather\_score的table欄位 21](#_Toc105332816)

[表3.5 category的table欄位 21](#_Toc105332817)

[表3.6 v\_category\_clothes的table欄位 22](#_Toc105332818)

[表3.7 v\_clothes\_information的table欄位 23](#_Toc105332819)

[表3.8 v\_clothes\_graph\_edge的table欄位 24](#_Toc105332820)

[表3.9 city的table欄位 26](#_Toc105332821)

[表3.10 station的table欄位 26](#_Toc105332822)

[表3.11 v\_station的table欄位 27](#_Toc105332823)

[表3.12 user\_dashborad的table欄位 28](#_Toc105332824)

[表4.1參與本項研究之人員及工作內容 34](#_Toc105332825)

# 圖 目 錄

[圖1.1 研究流程魚骨圖 4](#_Toc105357124)

[圖3.1 訓練模型AlexNet style圖 7](#_Toc105357125)

[圖3.2 Kaggle訓練集網站 8](#_Toc105357126)

[圖3.3 訓練模型的流程圖 9](#_Toc105357127)

[圖3.4 辨識出黑色外套的結果 12](#_Toc105357128)

[圖3.5 辨識出白色T-Shirt的結果 12](#_Toc105357129)

[圖3.6 進行衣服辨識的流程圖 13](#_Toc105357130)

[圖3.7 中央氣象局所API提供站別 14](#_Toc105357131)

[圖3.8 中央氣象局所提供之JSON格式 15](#_Toc105357132)

[圖3.9 存入資料庫流程圖 16](#_Toc105357133)

[圖3.10 獲得天氣資訊流程圖 17](#_Toc105357134)

[圖3.11 爬蟲與資料庫抓取速度比對圖 17](#_Toc105357135)

[圖3.12 EEL運作 19](#_Toc105357136)

[圖3.12 v\_category\_clothes 資料預覽 22](#_Toc105357137)

[圖3.14 v\_clothes\_information資料預覽 24](#_Toc105357138)

[圖3.15 v\_station的資料預覽 28](#_Toc105357139)

[圖3.16 外部架構圖 29](#_Toc105357140)

[圖3.17 存放時的架構圖 30](#_Toc105357141)

[圖3.18 拿取時的架構圖 30](#_Toc105357142)

[圖3.19 存放時檢視口流程圖 31](#_Toc105357143)

[圖3.20 拿取時檢視口流程圖 33](#_Toc105357144)

[圖6.1 執行進度甘特圖 36](#_Toc105357145)

# 研究動機與研究問題

## 研究動機

不論是學生或上班族，面對緊湊的生活步調，在出門前往往需要花費許多時間決定日常生活的衣物穿搭，除了要找尋適合自己當日外出場合的服裝之外，也需要利用手機查詢天氣概況，避免因為穿著不夠保暖，受風寒而影響身體健康；又或者穿著太多，身體過熱感到不適。因此，如何在出門前，快速地選定適合自己心情的穿搭，並且兼顧美觀與身體舒適是一個值得研究的議題。

為了解決上述問題，本計畫擬設計並開發一個智能衣櫃，此衣櫃具備一個主控系統、一個螢幕、一組馬達控制系統與攝像頭。首先，使用者將先添購的衣物放進衣櫃時，系統會自動驅動攝像頭將該衣物進行拍照，並透過服飾影像識別演算法自動識別衣物類別，存入資料庫的同時也會啟動馬達將衣物收納擺放好。當使用者要出門時，系統會抓取當地氣候資訊，並依據自訂的演算法選定穿搭衣物，也可以經由使用者定義。最後，當使用者決定該穿搭衣物後，系統會驅動馬達控制系統，將衣物送到方便使用者拿取的位置，使用者拿取後即可穿搭出門。

## 節省時間

* + - 1. 自動天氣預報

讓使用者能夠在使用衣櫃的同時，也可以迅速獲得天氣狀況，不僅節省時間更能讓沒有習慣看天氣預報的使用者，有更好的生活體驗。

* + - 1. 有效利用時間

將所有需要進行瑣碎步驟的程序進行統整，從辨識衣物開始至存放，到使用者真正使用，從選取衣物到提出衣服，其實並不需要使用者進行太多的操作，皆可讓一切自動化，最後結果一切都由電腦自動演算得出，來達成節省時間的目的。

透過演算法統包成自動運行，再結合利用行動裝置的應用軟體來讓使用者可以隨時隨地使用。

### 了解衣物使用狀況

先透過圖樣辨識等技術將衣物資料存儲至資料庫當中，其中包含存入時間、使用次數、類別、顏色、及在實體衣櫃中儲存的位置欄，可達到以下效果：

1. 衣物使用狀況

透過人機介面可察看存放至資料庫內紀錄衣物的各項資訊，如使用次數的多寡，能得出實用性及喜好程度不高衣物，來記錄使用者偏好的衣物樣式，減少購買不必要的衣物節省空間的使用，也可以減少使用者支出的問題，更能了解使用者對於衣物使用的喜好，使用者也可選擇使用次數較少的衣服，來達到使用率平均，減少衣物耗損。

1. 降低衣物耗損

透過使用資料庫查看使用次數，知曉使用者衣物使用頻率、存放時間及存放環境的觀測，對衣物有更好的控管，也可提醒使用者衣物保存的狀況如何，避免過度存放，時間過太久的衣物容易發霉等諸多問題產生，建議汰換。

## 研究問題

### 圖樣辨識

如何將衣物的分類區別(如：短袖、長袖、帽T、短褲等)，取出該衣物的顏色。當有新的衣物要加入到衣櫃中，會先將衣物進行辨識，辨識出它是什麼樣的衣物、屬於哪一個分類以及是什麼顏色，再將它放入指定的衣架上，等下次要進行穿搭衣服的時候，可以優先推薦新加入的衣物(通常會因為喜歡而購買衣物並加入到衣櫃中)。

### 衣物穿搭推薦演算法

如何聯網至中央氣象中心的Application Interface中獲取有關於天氣的資訊，其中包含當日濕度、溫度、最高溫及最低溫。在每一次放入衣物時將圖樣辨識所獲取的資訊存入資料庫後，建立一無向圖，將不同分類的衣物進行組合，並且記算每件衣服的天氣適配程度，如天氣熱可搭配短袖或短褲，依此類推。接著將不同的衣物進行組合，為不同風格進行評分，參考衣物色彩學，如同色系，互補色等。

### UI(User Interface)介面整合

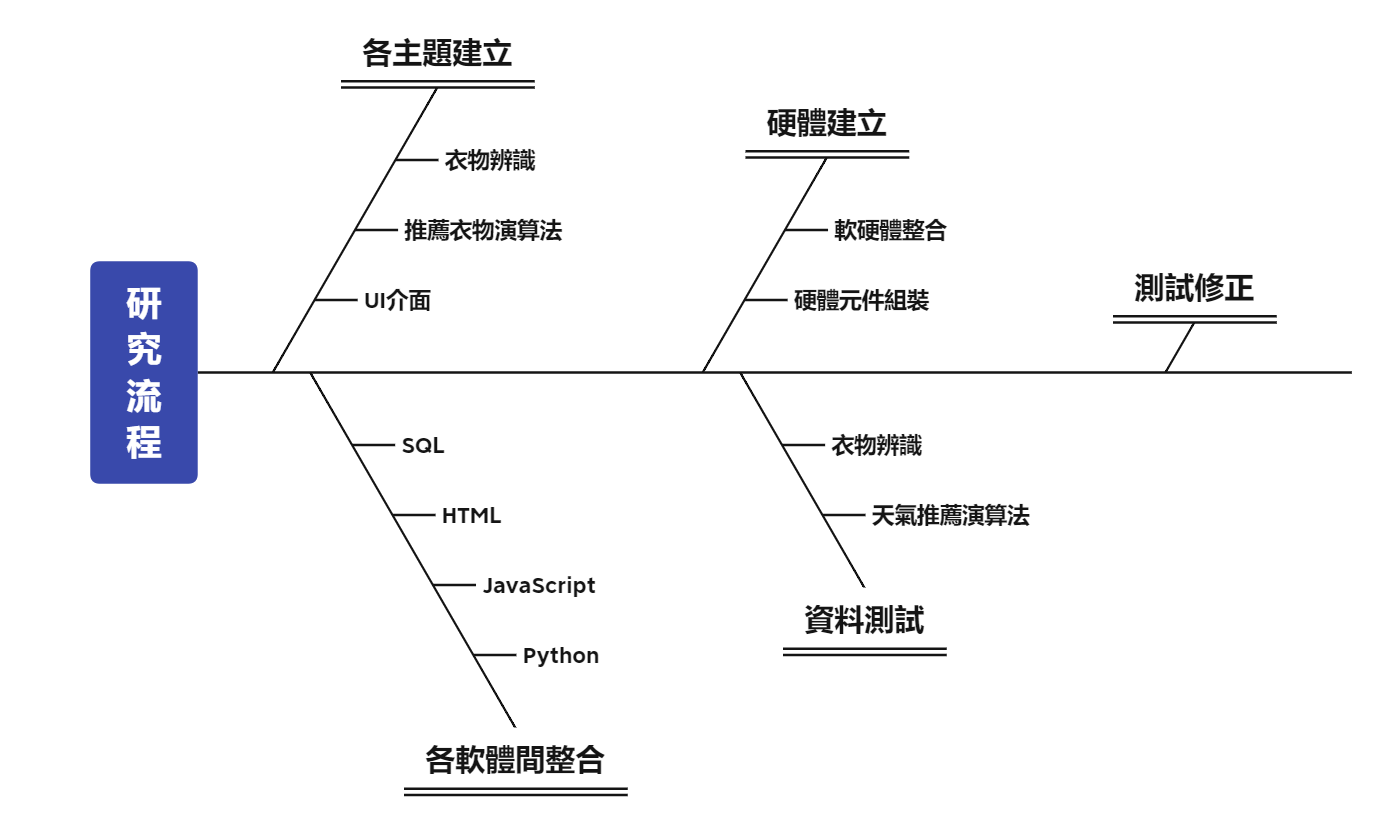
使用者開啟主機後，UI介面會顯示系統完成衣物穿搭推薦後的組合，可點選想穿著的衣物組合後，會自動搜索該衣物的位置資訊並利用馬達控制系統將衣物推出至檢視口方便使用者拿取，也能自訂義想穿著的衣物，無須必定使用系統所推薦的組合。此外也可再設定頁面調整使用者個人需求，將產品逐漸個性化，因應不同使用者不同需求。

### 馬達控制系統

馬達控制系統包含微控器、感測器與馬達控制，可以讓衣櫃具備自動收取功能。架設衣櫃的模型後，由UI介面傳送資料至電路板上，採取相對應的動作，收放、推送等。

### 研究流程

將預計研究的主題分為三部分研究，軟體層、硬體層、軟硬體整合。先是將軟體的各自主題建立，待流程順利後便將各軟體間進行的應用進行整合，最後再建立IOT設備。由於衣物辨識和天氣演算法需透過不斷測試調整參數來提高準確率，最後流程為重複進行測試修正。



**圖1.1 研究流程魚骨圖**

# 文獻探討

基於第1章所述之研究動機與目的，在智能家電中還未有出現智能衣櫃相關的產品過，大多都是屬於智能除溼、除臭等功能的衣櫃廣受大家認知。此研究的重點在於想法上的結合，撇去繁重的手動新增，使用Python所擁有的套件，達成智能化的效果。

首先建立人工智慧之圖樣辨識等方式自動辨識衣物類別，為了有更好的訓練成功率，在這裡是使用網路上公開的資料集，由於數據較多所以測出來的會較為準確，由Kaggle中Clothing\_Dataset大量圖片作為訓練集[[1]](#_參考文獻)並訓練其模型[[2]](#_參考文獻)，訓練過後的模型可轉為HDF模型[[3]](#_參考文獻)方便後續獲得樣式類別。透過微動開關[[4]](#_參考文獻)確認感應衣物重量，開啟鏡頭對影像進行分析，先是將圖片利用OpenCV感測中心點進行HSV量質化表偵測[[5]](#_參考文獻)，獲得一定參數內的數值[[6]](#_參考文獻)，可獲得該衣物顏色，在獲取顏色後結束攝像頭並擷取圖片儲存，並使用HDF模型對其圖片進行辨識，獲得樣式後回傳至資料庫[[7]](#_參考文獻)進行儲存，辨識後的衣物可自動新增至SQL Server[[8]](#_參考文獻)達到衣物管理的效果。

演算法利用requests與BeautifulSoup4進行網頁爬蟲[[9]](#_參考文獻)，負責抓取中央氣象局所提供之現存測站[[10]](#_參考文獻)，由於[[11]](#_參考文獻)提供範例，因此為了提高演算速度事先將縣市與站別分為兩個部分存入資料庫再取用會較為快速。不定時更新的站別資料導致不穩定性，因此寫成一演算法會在固定時間內進行站別更新，隨後利用中央氣象局所提供API[[12]](#_參考文獻)來獲取天氣資訊，所獲得之資訊為JSON格式檔案，因此利用正規表示法[[13]](#_參考文獻)即可自動獲取當日當時所需天氣資訊。

自動推薦演算法是利用26度穿衣法則[[14]](#_參考文獻)並加以修改，透過演算得出今日最合適衣物結果，由於後端核心都由Python架構，而前端所需的HTML與JS需要有套件結合，因此所使用的函式庫會與第三方開發程式EEL[[15]](#_參考文獻)所串接，結合EEL與前端HTML和JavaScript，並使用前端的開發程式Angular[[16]](#_參考文獻)對前端進行管理。

最後結合IOT設備進行實體操作，先是利用直流馬達與微動開關建立可操控式履帶，伺服馬達[[17]](#_參考文獻)負責控制將衣物送至相對應儲存格。

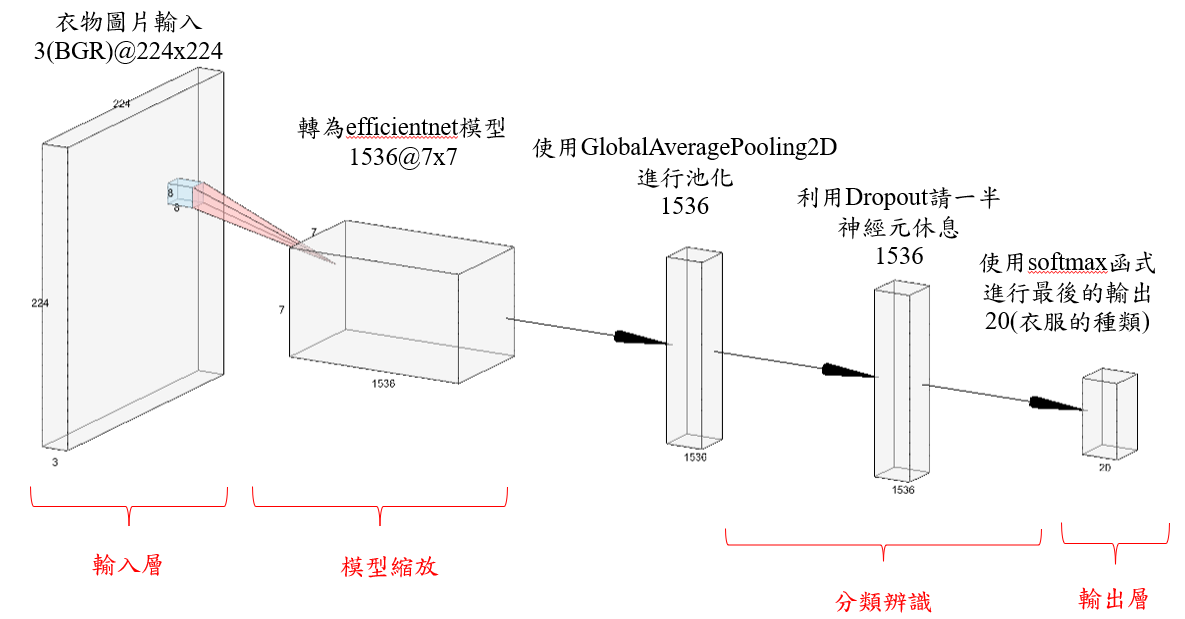
# 研究方法與工作項目

## 各項研究方法

### 衣物辨識

1. 訓練模型

一開始先在Kaggle網站找到衣物的彩色衣物的訓練集，全部的訓練集大約五千多，其中尤其上半身的訓練圖片約有2500張，而下半身的訓練圖片落在1500張左右，而訓練的標籤有20種，使用TensorFlow的套件將這些衣物進行訓練，最後會訓練出一個模型，將這個模型存成HDF5檔儲存起來，這麼做可以不需要每次辨識衣物時還要再跑一次訓練模型。

訓練模型使用到四層神經元分別如下:

**圖3.1 訓練模型AlexNet style圖**

以下是所使用的訓練集以及訓練標籤與樣本:

一張含有 文字 的圖片

自動產生的描述

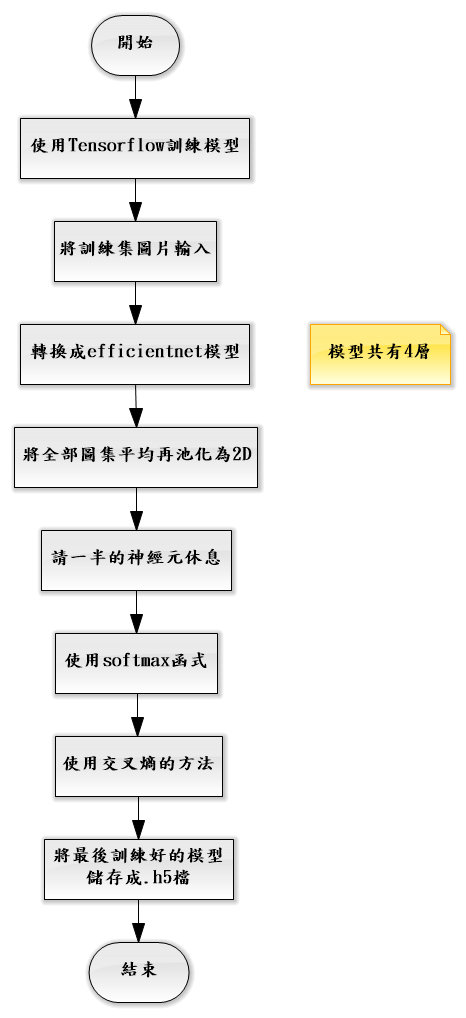
**圖3.2 Kaggle訓練集網站**

所使用的衣物種類標籤與訓練樣本數，上半身不包含外套的種類有7種，下半身有3種，外套類有2種，其他占比5種，共20項，如下表：

**表3.1 衣服的種類及訓練樣本**

|  |  |
| --- | --- |
| 衣物種類(label) | 訓練樣本數(items) |
| T-Shirt | 1011 |
| Long Sleeve | 699 |
| Pants | 692 |
| Shoes | 431 |
| Shirt | 378 |
| Dress | 357 |
| Outwear | 312 |
| Shorts | 308 |
| Hat | 171 |
| Skirt | 155 |
| Polo | 120 |
| Undershirt | 118 |
| Blazer | 109 |
| Hoodie | 100 |
| Body | 69 |
| Top | 43 |
| Blouse | 23 |

訓練完的模型成功率在86%，損失率落在47%。



**圖3.3 訓練模型的流程圖**

1. 辨識衣服

接下來會進行實際衣服的辨識，主要分為兩個部分

1. 衣服顏色

進行顏色辨識及衣服辨識要開啟攝影機時會使用到OpenCV的套件，將RGB轉成HSV，這麼做可以使攝影機只需要對一個維度(Hue)的顏色辨識，而且在打開攝影機的時候，辨識HSV顏色數據都會一直更新，直到按下截圖的瞬間，程式會抓最後辨識到的HSV顏色的數值，並判斷是什麼顏色，將判斷完顏色的衣物儲存至系統中。

**表3.2 依HSV辨識出來的顏色表**

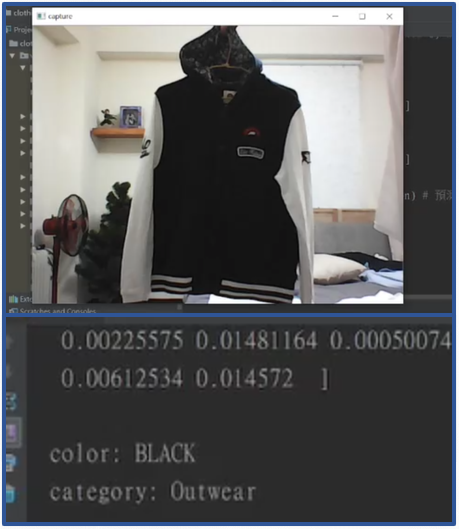
|  |  |  |  |
| --- | --- | --- | --- |
| Hue | Saturation | Value | Color |
| 不列入 | 不列入 | V < 50 | BLACK |
| 不列入 | S < 20 | 不列入 | WHITE |
| H < 5 H >= 167 | S >= 20 | V > 50 | RED |
| H < 22 | S >= 20 | V > 50 | ORANGE |
| H < 33 | S >= 20 | V > 50 | YELLOW |
| H < 94 | S >= 20 | V > 50 | GREEN |
| H < 131 | S >= 20 | V > 50 | BLUE |
| H < 167 | S >= 20 | V > 50 | PURPLE |

1. 衣服種類

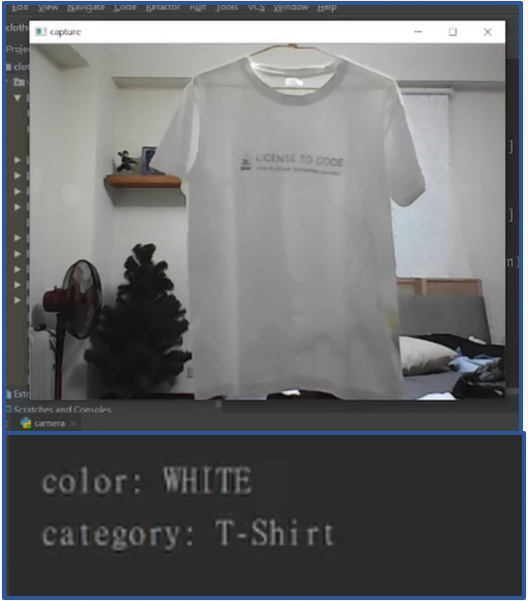
辨識衣服什麼種類，單靠攝影機是無法辨識出來，因此就必須要使用事先所準備的已模型訓練檔案，HDF5檔，將攝影機拍到的照片當作測試集來進行辨識，而衣物的種類共有20種，因此會有20種不同數值的可能性在陣列當中，將剛拍下來的照片進行辨識，在預測結果後可取得這20種不同的可能性，當中最高的數值也就是最有可能的結果，顯示在主控後台上。

最後得出來的衣服顏色及衣服種類用物件的方式傳入SQL內加入資料，並開啟馬達控制系統將衣物送進衣櫃內存放。

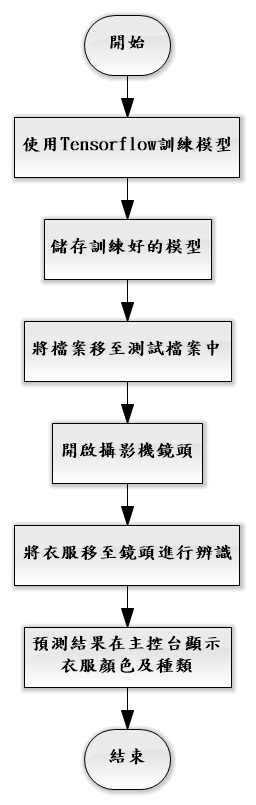
試驗樣本共15件，成功樣本11件，與模型的預估結果相符合，成功率約在七成左右，以下是本研究測試模型的過程：



**圖3.4 辨識出黑色外套的結果**



**圖3.5 辨識出白色T-Shirt的結果**



**圖3.6 進行衣服辨識的流程圖**

### 天氣擷取

* + - 1. 設計方法

使用環境為Python

首先是利用爬蟲(BeautifulSoup4)及請求套件(Requests)模擬使用者上網的樣式讓程式可自動查看網頁內容，透過HTML中的Tags擷取所需項目，所擷取的項目站別資料包含所屬縣市、區域等，並將資料分別存入資料庫中如表3.11。

再來申請中央氣象局的會員資訊，獲得授權碼，並使用Requests套件獲得中央氣象局所提供之API的詳細資料，使用方法為將預計查詢資料的網域網址與授權碼結合後，加上參數，即可獲得當時的天氣JSON格式資料。所獲得得JSON格式資料，再利用正規表示法(regular expression套件)擷取所需項目如圖2。

本研究計畫所使用的項目為：溫度、濕度、最高溫、最低溫、最高溫時間段、最低溫時間段。

一張含有 桌 的圖片

自動產生的描述

**圖3.7 中央氣象局所API提供站別**

一張含有 桌 的圖片

自動產生的描述

**圖3.8 中央氣象局所提供之JSON格式**

* + - 1. 限制

1. 多重判斷

氣象站分為兩種站別「自動氣象站別」及「局屬氣象資料站」，其分別為前者屬於自動偵測氣象站，後者為有人員管理之氣象站，兩站分別所管理的站別不會重疊。

解決方法：因此需要透過演算法判別在執行搜查動作，先是利用資料庫存取不同站別的數據，將資料保存下來後再傳到前端給使用者使用

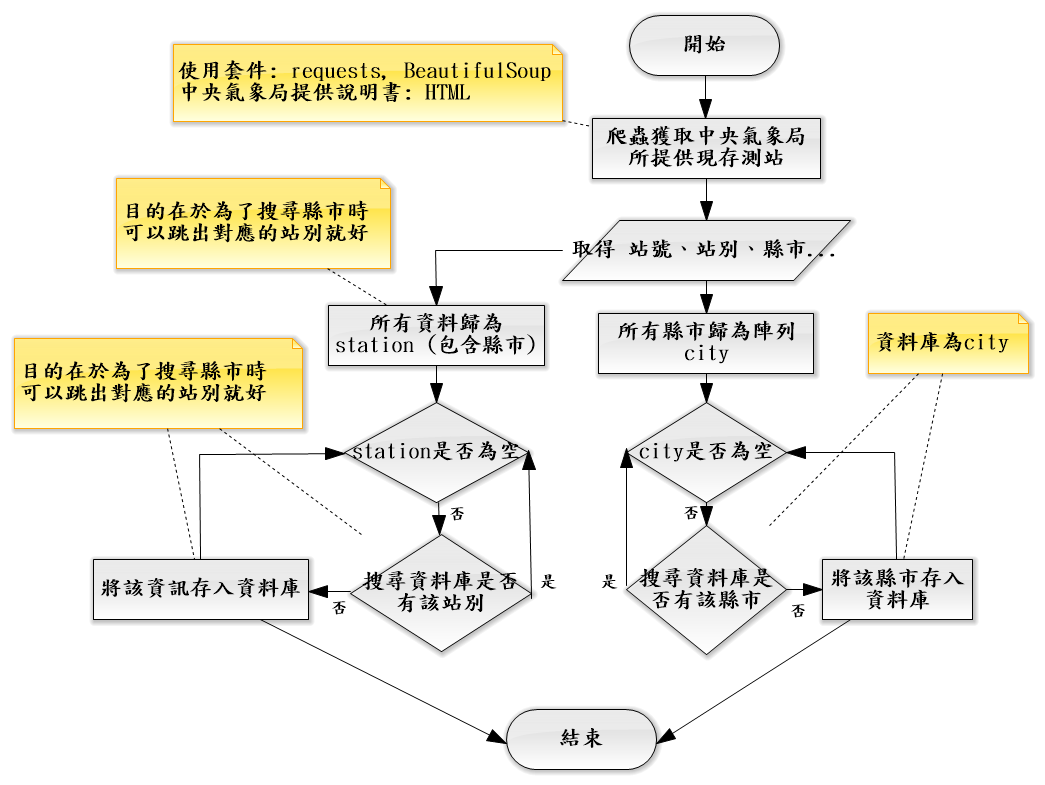
1. 參數不滿足

由於目前站別並沒有提供所有其他相關參數，如「風速」、「體感溫度」等，因此數據上會有偏差。  
解決方法：為利用個人偏好程度來調整個體差異，建造專屬化衣櫃，如體質怕冷、怕熱或偏好衣物等。

* + - 1. 實際操作

為了更好的令前端更簡易的搜索資料，將資料分為兩個部分，第一部分為縣市資料庫如city如表1，第二部分是將站別存入station如表2，為此可以在前端搜索的狀況利用資料庫的提升速度。

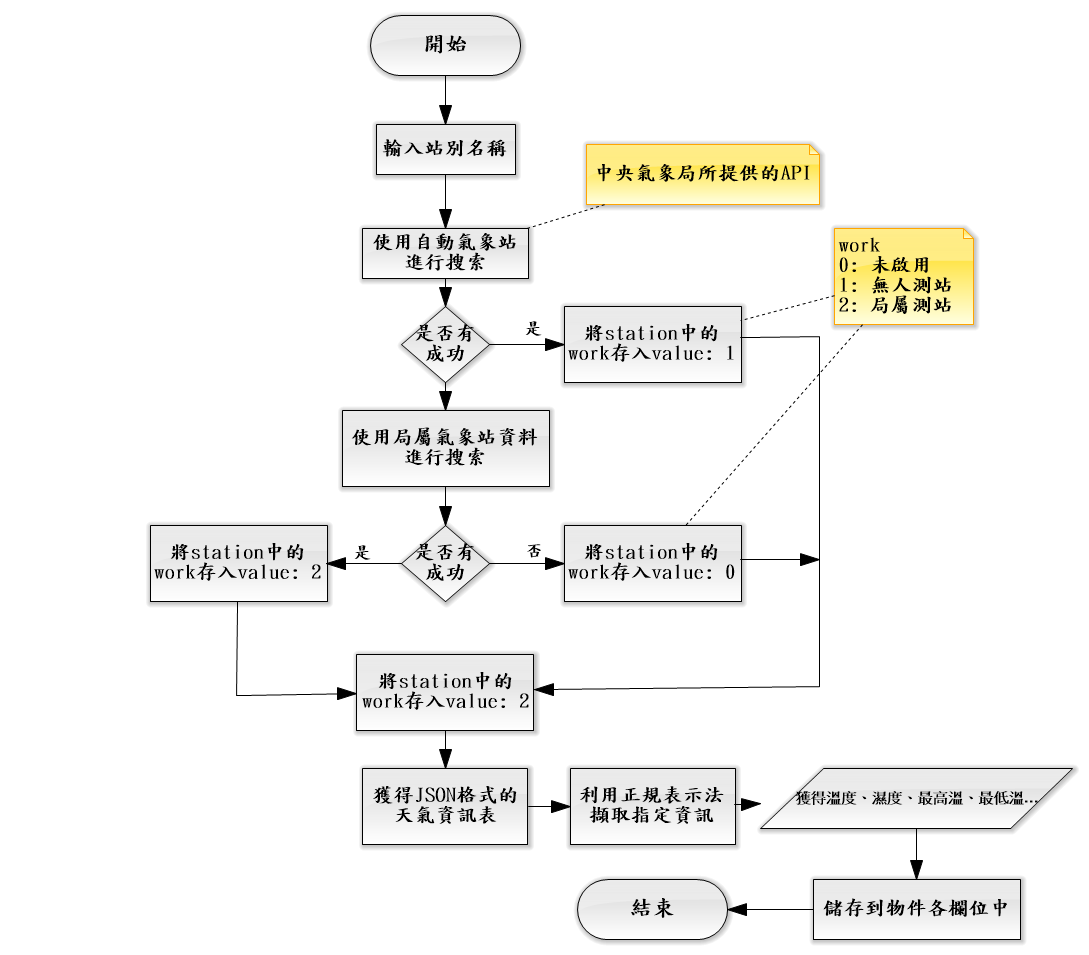
以下是將站別資訊存入資料庫的流程圖說明：



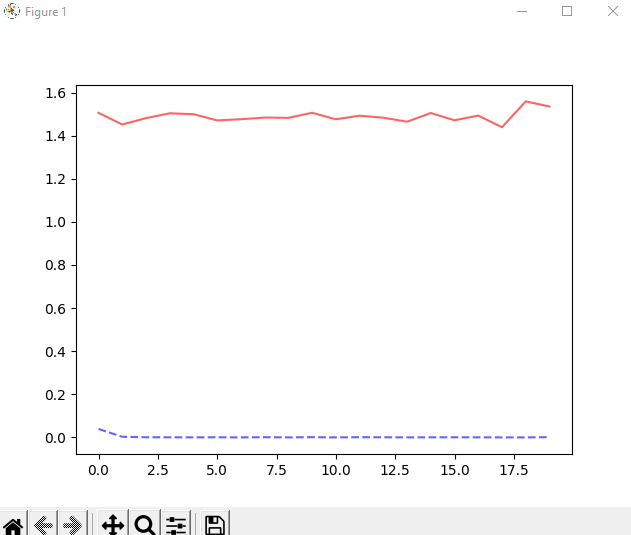
**圖3.9 存入資料庫流程圖**

獲得站別名稱後便可以獲得天氣資訊，為了避免重複作業，因此在資料庫的Work欄位上會記錄搜索過的結果，並將此存儲下來，以便後續使用時可以提升速度。

以下是天氣資訊擷取的流程圖說明：



**圖3.10 獲得天氣資訊流程圖**



**圖3.11 爬蟲與資料庫抓取速度比對圖**

### 天氣演算法

獲得衣物樣式後透過資料庫進行管理，並利用視圖檢視表來做到節點與節點之間的關聯性，達到類圖形化效果。對於推薦衣物的公式參考自二十六度穿衣法則並修正，計算如下列公式：，T為26度標準值，可利用使用者喜好設定值U進行加減，N為目前溫度與每件衣物的S值進行相加後平方即可獲得衣物的穿搭準則。

此演算的資料皆由已事先準備的資料庫中取得，S為v\_clothes\_infromation中的Score中提供，此資料是由weather\_score中的Id與clothes\_information的WeatherScoreId以inner join而來的資料，更方便的管理個資料的管控，詳細計畫規格書如表x，而U則為user\_dashborad中的WeatherLike中取得，如表3.12。

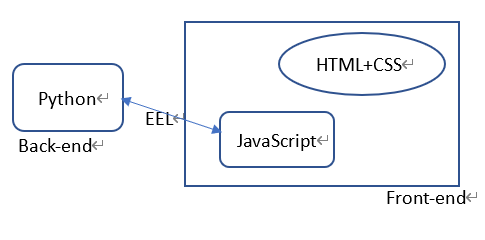
### UI介面

本研究分為後端與前段應用，前端使用JavaScript進行介面管理控制，而後端層面使用Python來進行演算、辨識、硬體設備呼叫及資料庫請求等動作，其串接的方法是使用Python所提供的EEL套件來為兩者進行聯通，EEL是一項可以讓Python與前端JavaScript連接的第三方開發程式，並在JavaScript利用async等待連接，待Python啟用便會同時啟用JavaScript，同樣的道理也可以利用JavaScript呼叫Python來進行資料的引用、計算等。

JavaScript中負責管理HTML資料的套件為前端開發套件為Angular，相較於其他開發套件，Angular的擁有更完整的開發框架，並且支援各種平台的該發，且Angulare在資料的管理上更直觀，因此本研究使用這套開發框架來完善計畫，對變數等傳輸進行控制。

後端與前端的來回作用分別為三種，對資料庫、Arduino及演算法等請求回傳。

在站別的資料庫回傳中先是利用Python主動呼叫資料庫並傳至JavaScript，並使用Angular前端管理套件來對HTML的各項節點內資料傳輸或是回傳，由於使用開發工具，前端的頁面將會更為靈活，同時也支援從Python所開啟的本地端伺服器。待使用者選擇完指定站別後也可以從JavaScript中回傳該參數給Python並進行請求等動作來獲得需求的資料。



**圖3.12 EEL運作**

### 資料庫規格書

資料庫表名稱採用小寫底線區分，欄位一律大寫開頭採取駝峰式命名法，使用的資料庫管理工具為SQL Server，語言為SQL。View table，作用將資料整合讀取，不能新增修改刪除查詢等。

1. 衣物管理

clothes\_information在資料庫中負責存取有關於衣物節點有相關的資訊，在攝像頭辨識完結果後會存到此資料庫中，其目的在於為了更好的管理衣物，並且做到紀錄，可以做到衣物管控等，也方便後續演算進行運算，如下表：

**表3.3 clothes\_information的table欄位**

|  |  |  |  |
| --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 |
| Id | bigint | Not Null | 編號 |
| Position | int | Null | 放在衣櫃裡面的位  (0~9) |
| WeatherScoreId | int | Not Null | 衣物分類(weather\_score.id) |
| Color | varchar(50) | Not Null | 衣物顏色 |
| UserPreferences | int | Null | 使用者喜好程度 |
| CategoryId | varchar(255) | Not Null | 衣物種類(category.id) |
| ClothesStyle | varchar(255) | Null | 衣物風格 |
| UsageCounter | int | Not Null | 使用次數 |
| CreateTime | datetime | Not Null | 創建時間 |
| ModifyTime | datetime | Not Null | 異動時間 |
| FilePosition | text | Not Null | 圖片檔案位置 |

weather\_score負責存取衣物種類名稱及分數之間的關聯性，以及統一衣物的名稱，這些資料與辨識的標籤有關，因此辨識的標籤有多少此資料就會有多少，除此之外並會對這些進行模擬分析，為後續天氣演算法計算天氣識配衣物有很大的幫助，由資料庫負責管理也可讓程式更好優化及除錯，以便後續的更新。

**表3.4 weather\_score的table欄位**

|  |  |  |  |
| --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 |
| Id | bigint | Not Null | 編號 |
| CategoryId | bigint | Not Null | 衣物分類 |
| ClothesType | varchar(255) | Not Null | 衣物種類 |
| Score | bigint | Not Null | 天氣分數 |
| Name | nvarchar(255) | Null | 衣物種類名稱 |

category負責管理衣服種類，其中衣物種類分為上半身、下半身、外套、洋裝、配件、鞋子等…為了辨識的標籤有不同的需求，衣物層級優先級，最高級上半身與下半身屬於必穿著的類型，而外套屬於第二層級，將會在最高層級後有需求才穿著。

**表3.5 category的table欄位**

|  |  |  |  |
| --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 |
| Id | bigint | Not Null | 編號 |
| CategoryName | varchar(50) | Not Null | 衣物分類 |
| Level | int | Not Null | 衣物層級優先級 |

v\_category\_clothes整合category和weather\_score，為衣物更完善了階層級和天氣分數便於後續要計算最佳衣物組合時。（註：ws代表weather\_score，cat代表category。）

**表3.6 v\_category\_clothes的table欄位**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 | 來自 |
| Id | bigint | Not Null | 編號 | ws |
| CategoryId | bigint | Not Null | 衣物分類 | ws |
| CategoryName | varchar(50) | Not Null | 衣物名稱 | cat |
| ClothesType | varchar(255) | Not Null | 衣物種類 | ws |
| Score | bigint | Not Null | 天氣分數 | ws |
| Name | nvarchar(255) | Null | 衣物種類名稱 | ws |
| Level | int | Not Null | 衣物層級 | cat |



**圖3.13 v\_category\_clothes 資料預覽**

v\_clothes\_information 負責作為查詢使用，更好的管理所有資訊的集合，排除不必要的項目。（註：ci代表clothes\_information，ws代表weather\_score，cat代表category。）

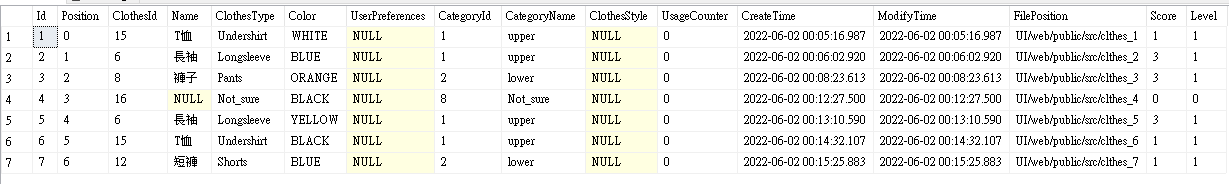
**表3.7 v\_clothes\_information的table欄位**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 | 來自 |
| Id | bigint | Not Null | 編號 | ci |
| Position | int | Null | 衣櫃內位子  (0~9) | ci |
| ClothesId | varchar(50) | Not Null | 衣物分類編號 | ci |
| Name | nvarchar(255) | Null | 衣物名稱 | ws |
| ClothesType | varchar(255) | Not Null | 衣物種類 | ws |
| Color | varchar(50) | Not Null | 衣物顏色 | ci |
| UserPreferences | int | Null | 使用者喜好程度 | ci |
| CategoryId | bigint | Not Null | 分類編號 | ci |
| CategoryName | varchar(50) | Not Null | 衣物分類 | cat |
| ClothesStyle | varchar(255) | Null | 衣物風格 | ci |
| UsageCounter | int | Not Null | 使用次數 | ci |
| CreateTime | datetime | Not Null | 創建時間 | ci |

(待續)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ModifyTime | datetime | Not Null | 異動時間 | ci |
| FilePosition | text | Not Null | 圖片檔案位置 | ci |
| Score | bigint | Not Null | 天氣分數 | ws |
| Level | int | Not Null | 衣物層級 | cat |

**表3.7 v\_clothes\_information的table欄位(續)**



**圖3.14 v\_clothes\_information資料預覽**

v\_clothes\_graph\_edge此table負責存取有關於衣物與衣物之間的關聯性，用無向圖的概念製作，主要核心利用inner join來達成，其目的在於可以對衣物進行各種演算。

**表3.8 v\_clothes\_graph\_edge的table欄位**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 | 來自 |
| Id | bigint | Not Null | 編號 | 自計算 |
| Clothes1Position | int | Not Null | 衣服  一號位置 | ci1 |
| Clothes2Position | int | Not Null | 衣服  二號位置 | ci2 |

(待續)

**表3.8 v\_clothes\_graph\_edge的table欄位(續)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Clothes1CategoryId | bigint | Not Null | 分類編號 | ci1 |
| Clothes2CategoryId | bigint | Not Null | 分類編號 | ci2 |
| Clothes1ClothesName | varchar(50) | Not Null | 衣物分類 | w1 |
| Clothes2ClothesName | varchar(50) | Not Null | 衣物分類 | w2 |
| Clothes1WS | bigint | Not Null | 天氣分數 | w1 |
| Clothes2WS | bigint | Not Null | 天氣分數 | w2 |
| AdaptationScore | bigint | Null | 總分 | Score  相加 |
| Clothes1Preferences | int | Null | 衣服一使用者喜好 | ci1 |
| Clothes2Preferences | int | Null | 衣服二使用者喜好 | ci2 |
| TotalPreferences | int | Null | 喜好程度總分 | 喜好度  相加 |

1. 站別管理

city負責存取有關於台灣目前開放中央氣象局所提供之API站別的所有縣市，資料由爬蟲所填入。

**表3.9 city的table欄位**

|  |  |  |  |
| --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 |
| Id | bigint | Not Null | 編號 |
| CityName | varchar (50) | Null | 城市名稱 |

station負責存取有關於台灣目前開放中央氣象局所提供之API站別的所有站號名稱及其資訊，資料由爬蟲所填入。

**表3.10 station的table欄位**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | | 是否為空 | | 註解 |
| Id | | bigint | | Not Null | 編號 |
| StationNumber | | varchar(50) | | Null | 站號 |
| StationName | | nvarchar(50) | | Null | 站名 |
| CityId | | bigint | | Not Null | 城市編號 |
| Address | | nvarchar(255) | | Null | 地址 |
| Remark | | nvarchar(255) | | Null | 備註 |
| CreateTime | | date | | Null | 資料起始日期 |

(待續)

**表3.10 station的table欄位(續)**

|  |  |  |  |
| --- | --- | --- | --- |
| ModifyTime | Date | Not Null | 異動時間 |
| Work | Int | Null | 作用區域  0: 不作用,  1: 自動氣象站,  2: 人工氣象站 |

v\_station負責結合city與station欄位，作用於查詢站別間的關係，在UI介面首頁可選擇。（註：st代表station）

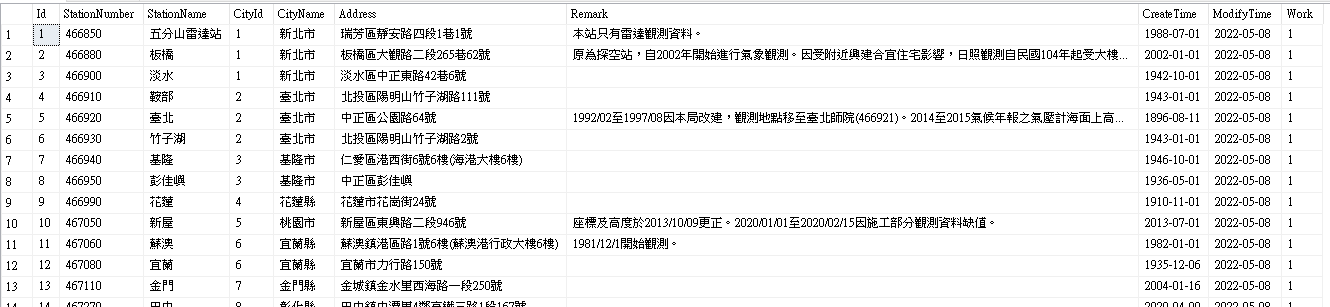
**表3.11 v\_station的table欄位**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | 是否為空 | 註解 | 來自 |
| Id | bigint | Not Null | 編號 | st |
| StationNumber | varchar(50) | Null | 站號 | st |
| StationName | nvarchar(50) | Null | 站名 | st |
| CityId | bigint | Not Null | 城市編號 | st |
| CityName | varchar (50) | Null | 城市名稱 | city |
| Address | nvarchar(255) | Null | 地址 | st |
| Remark | nvarchar(255) | Null | 備註 | st |
| CreateTime | date | Null | 資料起始日期 | st |

(待續)

**表3.11 v\_station的table欄位(續)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ModifyTime | Date | Not Null | 異動時間 | st |
| Work | Int | Null | 作用區域 | st |



**圖3.15 v\_station的資料預覽**

1. 使用者管理

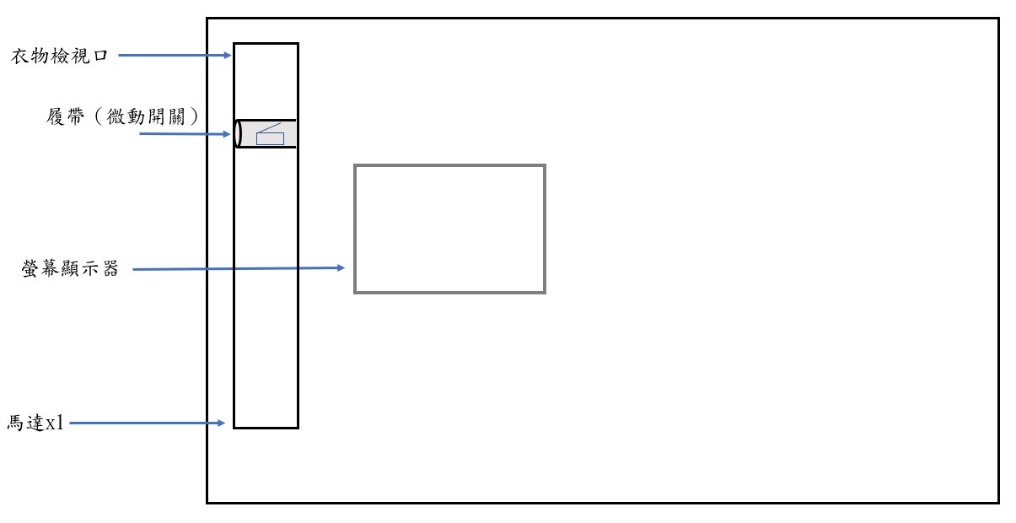
user\_dashborad 負責儲存使用者自己的使用習慣，例如比較怕冷或是怕熱，喜歡穿什麼類型的衣物，使用習慣等等。

**表3.12 user\_dashborad的table欄位**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 欄位名稱 | 欄位型別 | | 是否為空 | | 註解 |
| Id | | bigint | | Not Null | 編號 |
| UserName | | nvarchar(50) | | Null | 使用者名稱 |
| WeatherLike | | Int | | Not Null | 喜歡的天氣  (5為最熱) |
| ModifyTime | | datetime | | Not Null | 異動時間 |
| NowCity | | nvarchar (50) | | Null | 可以記錄使用者  目前所在城市 |

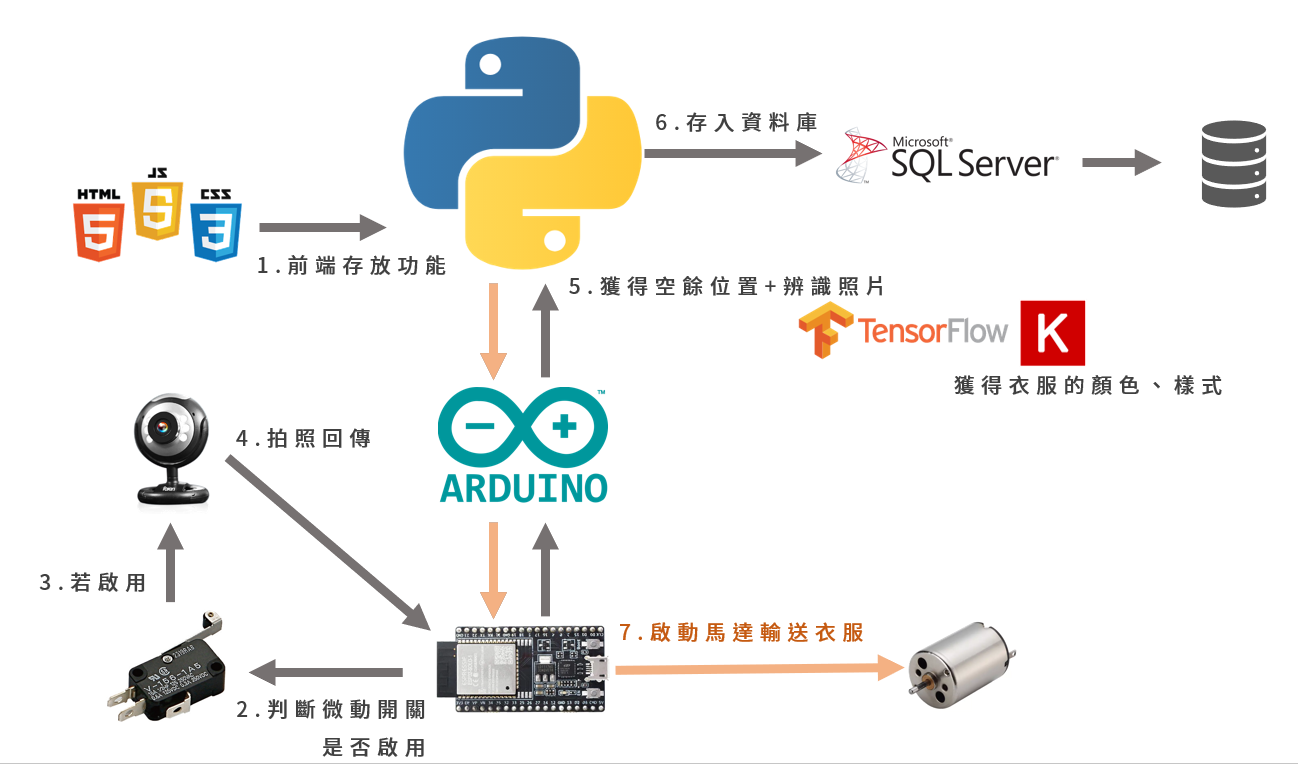
## 系統軟硬體架構之設計與整合

以使用者的面向來看，透過圖3.17的外部架構圖的螢幕顯示器上點選所需指令，包含存放與拿取，若選擇存放，將衣物擺放至衣物檢視口，電腦會確認微動開關是否被按下，若是將會啟動攝像頭將衣物進行辨識後將資訊存入資料庫並存放至空餘的指定位置。同理，點選拿取時，電腦將自動使用演算法抓取天氣資訊並得出最佳穿衣組合，從儲存位置點送出至檢視口，達到自動化的效果。

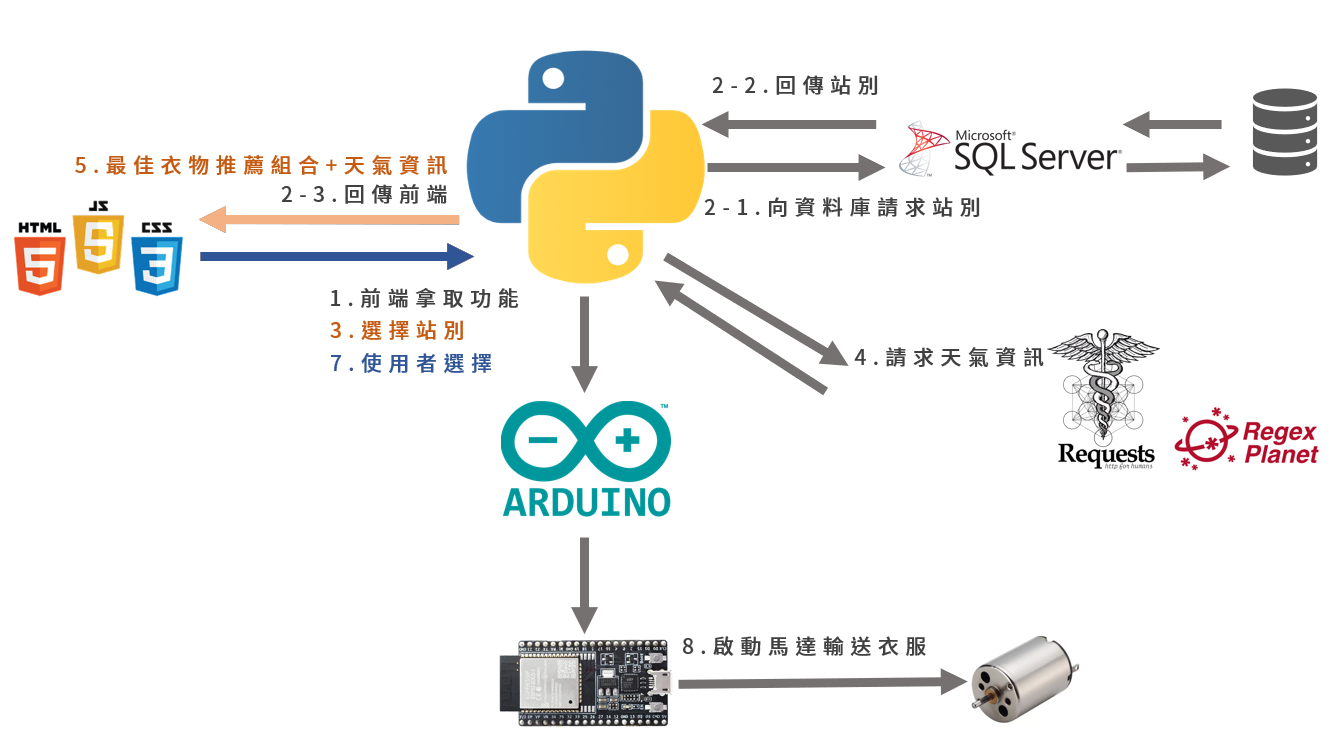


**圖3.16 外部架構圖**

下兩張圖是完整的軟硬體間的系統架構圖，分別以兩種模式拆開說明，其中兩者皆會使用的操作介面，是皆由網頁開發式架構來建立應用程式的前端畫面顯示，主要溝通核心在Python的各式演算上，而硬體的運作皆由系統自己判斷，不需要使用者逐步操作。



**圖3.17 存放時的架構圖**



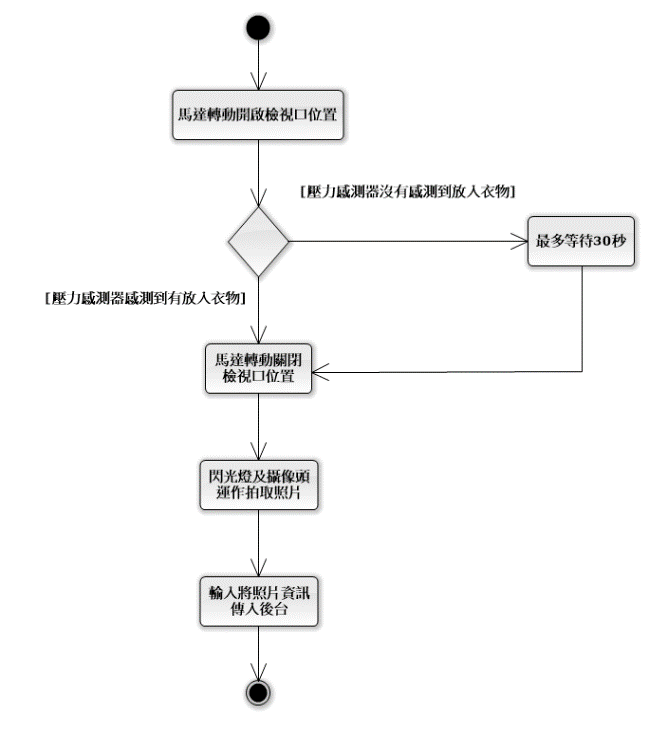
**圖3.18 拿取時的架構圖**

## 操作步驟

分為兩大流程，一是存放衣物，為準備的必要流程；二是拿取衣物，準備完畢後的使用過程。

### 存放

為在正式使用之前的準備過程，衣櫃上有一處存放衣物的檢視口，將衣物放入檢視口，透過微動開關確認是否有衣物放在該位置上，若有則關閉檢視口，避免光線影響辨識結果，檢視口的正中間會設置一台攝像頭，上方會使用照明裝置，待確認後拍照上傳至儲存裝置，若無則會等待三十秒後自動關閉。在放入後關上出口，利用Tensorflow人工智慧的辨識，獲得衣物樣式，上下衣物、長短及顏色等，並讀取衣櫃內的各項資訊，得出空餘位置後，回傳至主控面板的ESP32上，啟動馬達驅使綠帶轉動至編號位置，進行存放。而使用者同時也可在人機介面上查看目前衣物狀況。



**圖3.19 存放時檢視口流程圖**

### 拿取

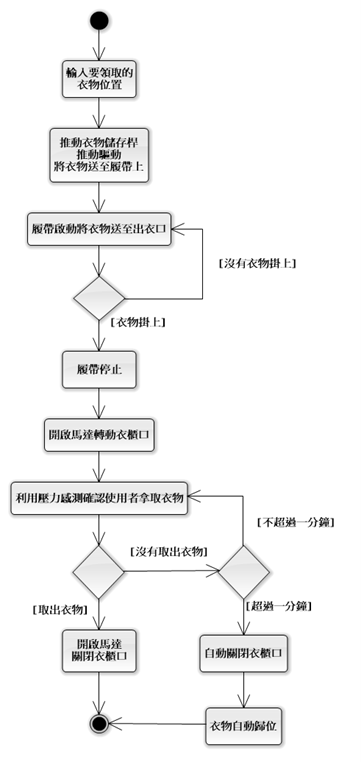
選擇「拿取」的選項後透過ESP32面板中的Wifi透過程式碼自動聯網至氣象局所開放的API中，抓取當日當地的平均溫度(TEMP)、濕度(HUMD)以及最高最低溫度(D\_TX、D\_TN)，及最高問溫時間段與最低溫時間段(D\_TXT、D\_TNT)。

螢幕顯示已抓取的天氣資訊，並且透過演算法演示出系統最推薦的四樣組合，除了可以選擇系統推薦的組合外，也可以讓使用者自定義喜好的搭配，客製化。

演算法的演算模式為根據抓取的資料，其中是利用資料庫模擬無向圖並連接每一條相關線路，使用者可表達對此衣物的喜好與不喜好將可降低衣物組合出現的機率，若不喜歡排列的所有組合，也可以自由性的選擇當日想穿搭衣服，可分別選擇當日想選擇的上衣及下褲即可。

確認今日穿搭組合後點選確定，所選擇的衣物位置會回傳到主控機台上，對應到該位置，推送到出口給使用者。

獲取資料後的機器透過ESP32版連接至該馬達輸出位置，等待運送桿到達將儲存桿上的衣物透過衣架取出，運送桿將衣物從儲存桿中提出後再運送至履帶上，並由履帶送至出衣口。



**圖3.20 拿取時檢視口流程圖**

# 人力配置

參與本研究之工作人員及預定之工作內容如表4.1所示。

**表4.1參與本項研究之人員及工作內容**

一張含有 桌 的圖片

自動產生的描述

# 預期結果

本計畫運用上述所描述的裝置與技術概念，預期做出一個可以將有關於早晨換衣的動作流程透過包裝，呈現迅速且一體的樣貌。以Arduino ESP32為硬體主控核心，利用SQL將使用者與各項數據的資料存儲起來及結合ESP32的WIFI模組來自動搜尋並運行演算法，於介面顯示演算結果，確認後即可由馬達自動推送該衣物，做到自動化等目的，並且在準備階段使用衣物辨識技術減少使用者手動輸入等狀況，一切皆由機器自己完成。相信本計畫的成果，可以大幅減少出門穿搭衣物的時間，並作為未來發展智能衣櫃的有用參考。

# 執行進度

一張含有 桌 的圖片

自動產生的描述

**圖6.1 執行進度甘特圖**

# 參考文獻

1. Ololo. (2020). Clothing Dataset (Full, High Resolution). Kaggle. <https://www.kaggle.com/datasets/agrigorev/clothing-dataset-full>
2. Hyunsoo. (2021). DL\_pro. Kaggle. <https://www.kaggle.com/code/hhs1516/dl-pro>
3. I code so i am. (2017, December 14). Day 04：關於 Keras 的一些小技巧 -- 組態、模型存檔與實驗. IT邦幫忙. <https://ithelp.ithome.com.tw/users/20001976/ironman>
4. 張元菘、蔣旻翰(2017)。腿部復健機構製作專題報告書。修平科技大學電機工程學系專題。未出版，台中市。<http://ir.hust.edu.tw/bitstream/310993100/6191/1/34-%E8%85%BF%E9%83%A8%E5%BE%A9%E5%81%A5%E6%A9%9F%E6%A7%8B%E8%A3%BD%E4%BD%9C-%E5%B0%88%E9%A1%8C%E5%A0%B1%E5%91%8A%E6%9B%B8.pdf>
5. Pysource. (2021, October 19). Simple Color Recognition with Opencv and Python. Youtube. <https://www.youtube.com/watch?v=t71sQ6WY7L4>
6. 駱易辰 (民96)。HSV 色彩空間前景物體抽取及其於人體動作辨識系統應用。國立交通大學電機與控制工程研究所碩士論文。已出版。新竹市。p14-15
7. 陳會安. (2020). SQL Server 2019/2017資料庫設計與開發實務. 碁峰.
8. Mike. (n.d.). SQL教學. Fooish程式技術. <https://www.fooish.com/sql/>
9. 彭彭的課程. (2019, January 11). Python 網路爬蟲 Web Crawler 基本教學. Youtube. <https://www.youtube.com/watch?v=9Z9xKWfNo7k>
10. 中央氣象局. (2019, January 11). 現存測站. <https://e-service.cwb.gov.tw/wdps/obs/state.htm>
11. 曹永忠. (2020, April 23). 【物聯網環控系統開發#2】 測試氣象局OPEN DATA的API KEY. MakerPRO科技創新實作社群媒體. <https://makerpro.cc/2020/04/test-api-key-of-cwb-open-data/>
12. 中央氣象局. (n.d.). 中央氣象局開放資料平臺之資料擷取API. Swagger. <https://opendata.cwb.gov.tw/dist/opendata-swagger.html>
13. 余晟. (2019). Regular Expression 橫刃萬解程式的關鍵語言：正規表示法. 深石.
14. 小郑杰西卡. (2021, November 15). 什么是「26 度穿衣法」？寒潮来袭怎么穿衣更合适？. 知乎. <https://www.zhihu.com/question/496588292>
15. Proladon. (2019, December 24). 【Proladon】Python Eel GUI - EP2. Python Eel 安裝與基本專案建置. Youtube. https://www.youtube.com/watch?v=FNPW2ZFksCQ&t=
16. 黃俊壹 (阿壹). (2022). 圖像 Angular 開發入門：打造高靈活度的網頁應用程式（iT邦幫忙鐵人賽系列書）. 博碩.
17. 趙英傑 . (2020). 超圖解 Arduino 互動設計入門 第四版. 旗標.