

物聯網期末報告

疲勞識別系統

資工系3年A班 學號 1091438 郁宸瑋

資工系3年C班 學號 1090953 陳元彬

資工系3年C班 學號 1093322 廖廷緯

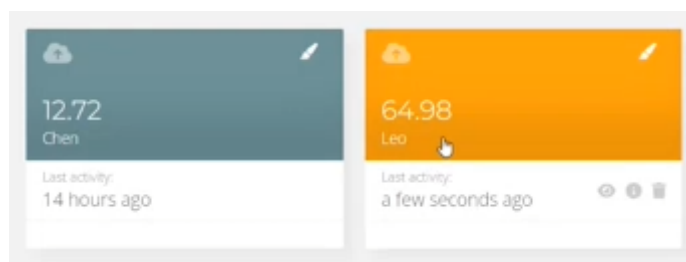
一、專題功能說明

在本次專題中，我們以透過樹莓派連接攝影鏡頭以及喇叭和LED來實作我們的疲勞駕駛偵測系統，其中也搭配了Ubidots中的功能來實作。

系統一開始會先透過預先訓練好的人臉模型來判斷該位使用者是否為模型中所存在的用戶，如果是的話便會通過檢測進入到主要判斷階段實時檢測駕駛的疲勞狀態，其中我們透過計算眼睛閉合程度來判斷駕駛是否閉眼和透過嘴巴張合的程度來判斷駕駛是否在打哈欠，如果在兩秒內，這兩種情形發生的程度高達八成，會觸發喇叭發出警示訊息，LED燈也會閃爍(模擬出車子行進時的閃黃燈，提醒附近駕駛注意此車)，最後此系統每分鐘都會上傳駕駛在這60秒內的總疲勞率到Ubidots上，透過內建的功能畫出疲勞率的折線圖，且當疲勞率超過50%時還會寄送email到指定的信箱內，來通知這個駕駛可能有過度疲勞的風險。這個功能可以用來通知駕駛親屬或大眾運輸工具司機的管理團隊，可以提前得知自己家屬的疲勞駕駛風險，藉此預防更多意外的發生。

二、專題操作與執行流程

1. 程式一開始會透過預先訓練好的人臉辨識模型來辨識使用者
2. 確認使用者後，會根據使用者是誰，選定 UserName 以及 Ubidots 上的Device，確認後是指定的使用者後，顯示 Verify pass，非指定使用者則會顯示 Verify fail
3. 進入檢測畫面後每兩秒會檢測確認一次人臉的眼皮以及嘴巴的閉合率超過一定比率就會觸發警示系統
4. 觸發警示系統後會使喇叭說出"Warring! Detect you are yawning"或"Warring! wake up"提醒使用者注意，並觸發 LED 燈閃爍模擬行駛在路上時可以警示附近的駕駛注意此車輛
5. 當檢測時間達到一分鐘後會將這個時間內的總疲勞率上傳到 Ubidots 上使用者的Device內畫出圖表(總疲勞率:疲勞張數/每秒總張數)
6. 當上傳到 Device 內的數值超過50時，代表這一分鐘內駕駛的疲勞率超過了50%此時會觸發 Ubidots 內部設置的事件寄送警示郵件給指定信箱，提醒使用者須嚴加注意



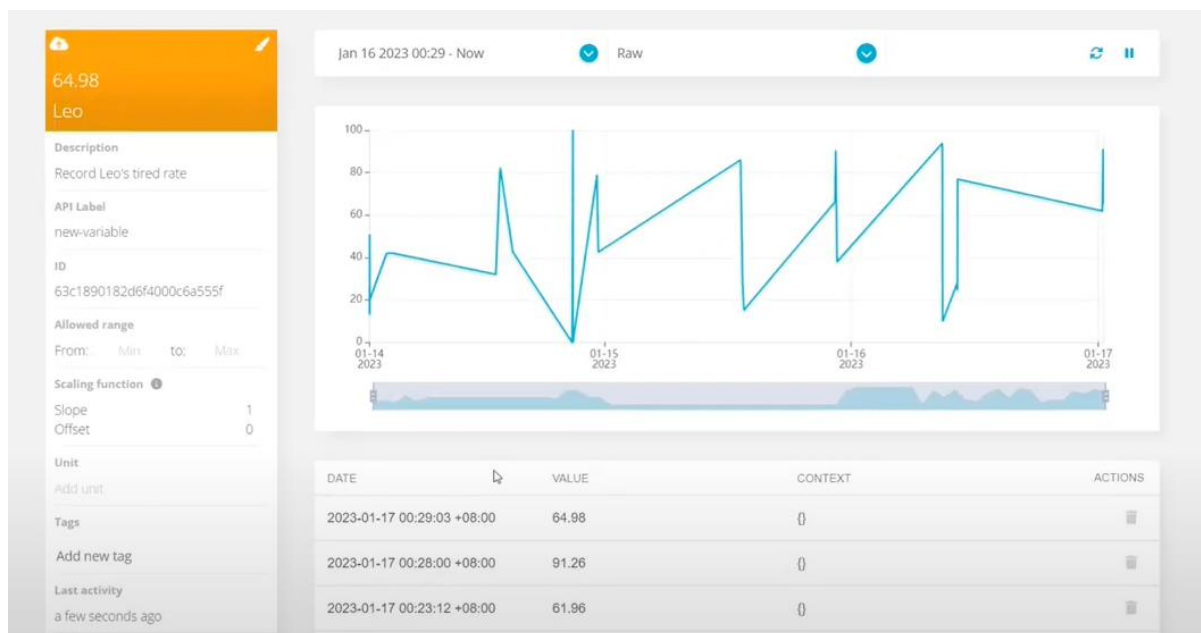
Ubidots區隔使用者內容

LED燈亮/暗，提醒道路其他使用者遠離

檢測時間達到一分鐘後，會將一分鐘內的總疲勞率上傳到

Ubidots 上使用者的Device內畫出圖表

(總疲勞率:疲勞張數/每秒總張數)



當上傳到 Device 內的數值超過50%時

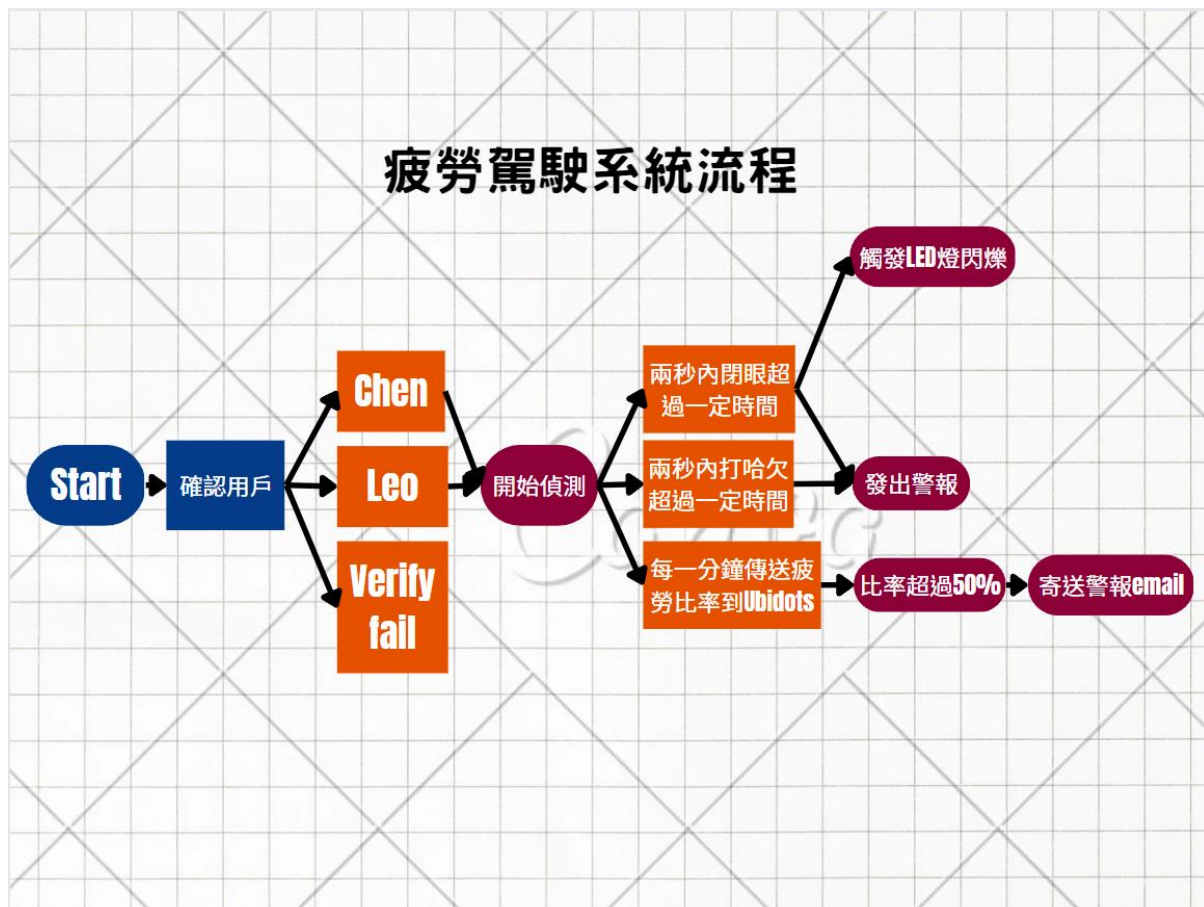
代表這一分鐘內駕駛的疲勞率超過了50%

此時會觸發 Ubidots 內部設置的事件

寄送警示郵件給指定信箱，提醒使用者須嚴加注意



三、軟體程式執行流程圖



四、參考的課程實驗或是網路資源

參考的課程實驗:

1. Lec 02 - RPi Basics
2. Lec 03 - GPIO (透過LED燈的閃爍，模擬閃黃燈)
3. Lec 06 - Network&Cloud(透過Ubidots完成寄信，以及疲勞統計圖表)
4. Lec 07 - Camera(透過外接Camera連接電腦，辨識並檢測使用者疲勞程度)
5. Lec 10 - Smart Speaker(透過喇叭發出警示給以疲勞的使用者)

網路資源:

1. <http://vision.fe.uni-lj.si/cvww2016/proceedings/papers/05.pdf>
(眼皮以及嘴巴閉合率的計算公式參考)
2. <https://steam.oxxostudio.tw/category/python/ai/ai-face-recognizer.html> (人臉辨識模型訓練方法參考)

五、開發最耗時的部分和原因

在這次的期末專題的開發過程中，除了一些課堂上用過的基礎代碼外，可以說是每一個階段都是一次未知的挑戰。雖然不管是哪一個過程單獨拿出來講都耗費大量的時間，但其中最耗費時間的還是功能設計以及查找資料上費了最大時間與精神。

在實作的過程中每次想要多新增功能時，都要去查找資料來確保實作這些功能的可行性，從一開始設想一個跟主功能很適配的額外功能時就需要大家集思廣益一段時間，但即便想出了之後也會因為使用工具的限制或資料不足而無法實現，最後導致許多感覺不錯的點子石沉大海，而這些時間也跟著耗費掉了。其他就算最後有實作出來也耗費了一定的時間或因為套件限制導致功能必須有所刪減，像是查找官方API、套件官方網站社群Q&A、語法、設定以及參考其它人實作過怎樣的功能來讓自己有更多新功能的靈感。

其次便是在觀察與測試人臉偵測程式的過程以及Debug，就算有了計算公式也要花時間觀察自己臉部變化時參數是如何變化的，並以此來設定判斷的標準值，還要觀察所擷取的影像有沒有其他干擾偵測的因素或是有沒有其他因素可以加速運算讓每秒能檢測的幀數更多進而得到更細微的數據。

六、組員名單與分工

學號	姓名	工作內容
1090953	陳元彬	樹梅派相關硬體安裝、 查找資料
1091438	郁宸瑋	樹梅派相關硬體安裝、 查找資料
1093322	廖廷緯	影像處理相關的Code (人臉辨識)

七、完整專題操作流程之展示影片

Youtube連結：https://youtu.be/TBz_t0KP9rc