

# 臺北市立大安高級工業職業學校專題製作競賽

## 作品說明書封面



類別：電機電子群

作品名稱：圖書館雜誌借閱模擬系統

關鍵詞：OpenCV、影像辨識、圖書館

## 目錄

壹、摘要 .....	1
貳、研究動機 .....	1
參、研究方法 .....	1
一、討論並確定可行性 .....	1
二、蒐集資料 .....	2
三、決定功能 .....	2
四、選定軟體及平台 .....	4
五、影像辨識程式測試 .....	4
六、整合測試 .....	7
肆、研究結果 .....	11
伍、討論 .....	12
陸、結論 .....	14
柒、參考資料及其他 .....	15

## 壹、摘要

物聯網是未來的趨勢，它可以用電腦對機器、裝置、人員進行集中管理、控制，也可以對家庭裝置、汽車進行遙控，以及搜尋位置、防止物品被盜等。本專題也是以物聯網為基礎進行影像辨識，後將結果傳至資料庫，再用網頁查看雜誌借閱的情況。

## 貳、研究動機

在 2017 年 Google I/O 大會的尾聲，Google 發表了結合人工智慧及機器視覺技術的影像辨識技術 Google Lens，可以辨識花種、自動連網還能提供店家評分。剛好想到圖書館無法外借的那些雜誌，是否也能用類似的技術來統計被人閱讀的次數，並把統計資料顯示在網頁上，供讀者及館方參考。這樣不僅能知道哪些熱門雜誌是熱門的，哪些是乏人問津的，還能使購買雜誌的資金效益最大化。也能，讓讀者知道哪些書正在被閱讀。因此，我們決定著手進行研究影像辨識及資料庫。

## 參、研究方法

研究流程圖，由發想到蒐集資料，在決定功能及平台後，開始實作，在經過測試成功後及完成，如【圖 1】所示。

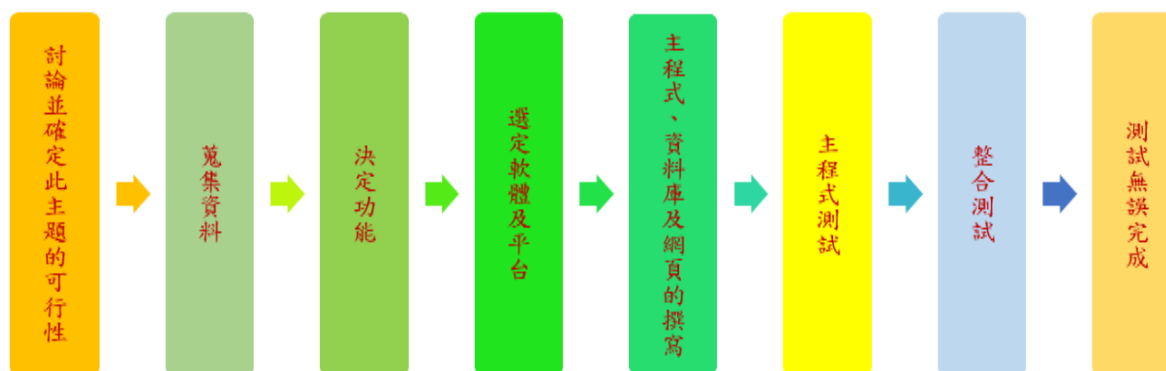


圖 1 研究流程圖

### 一、討論並確定可行性

近年來，物聯網(IoT)成為業界最熱門的話題。並隨著智慧型手機的普及，使物聯網的影響力更加深遠。因此，我們決定把影像辨識結合物聯網，做為借閱系統的核心，如【圖 2】所示。

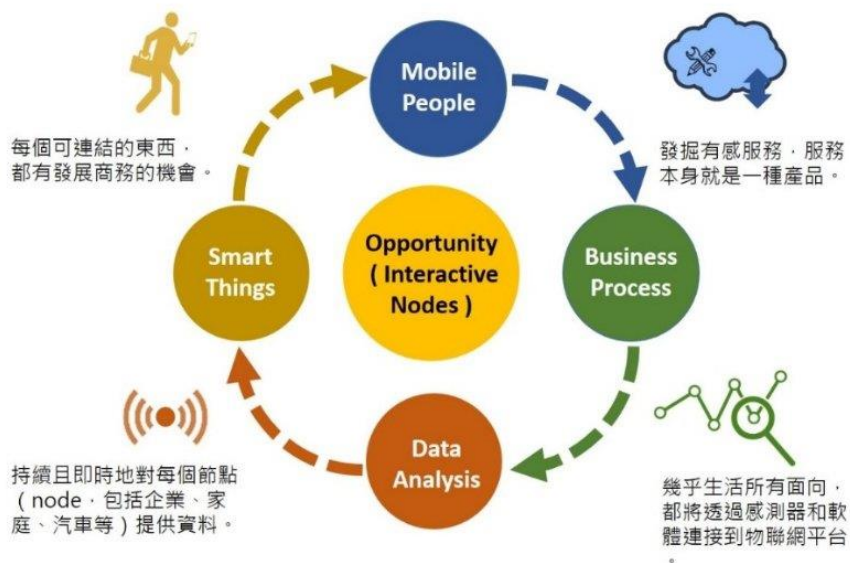


圖 2 物聯網示意圖

## 二、 蒐集資料

當方向確定後，我們尋找到了許多函式能幫助我們降低影像辨識程式碼撰寫的難度。但，如何將辨識結果傳給資料庫，到是費了一番工夫研究。最後，才找到適合的函式來搭建橋樑，成功的傳輸。

## 三、 系統功能

系統功能的部分，我們分成兩個部分完成；一塊是影像辨識，另一塊則是資料庫。首先，我們先將影像辨識的處理程序決定好。如【圖 3】所示。

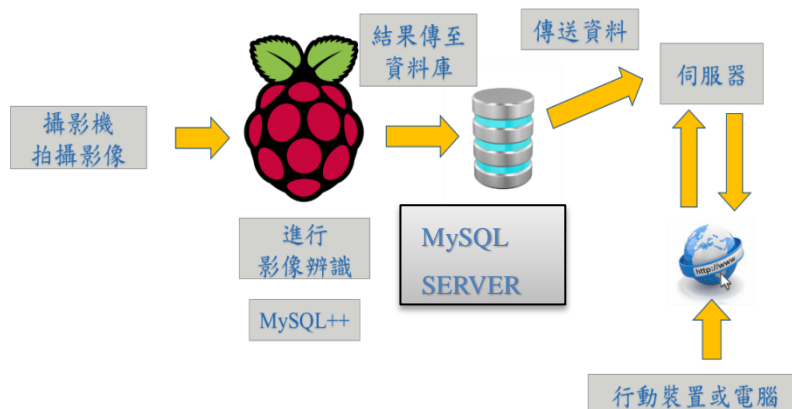


圖 3 影像辨識流程圖

第一塊是影像辨識，我們決定先經由攝影機拍攝畫面，再判斷是否有 15 張圖形。如果有，則代表書沒有被借走，直接書出結果。如果沒有，則代表有書被借走繼續進行影像處理，再經由圖形檢索，找出哪幾本被借走，如【圖 4】所示。用這種演算法的好處是，不管書有沒有放回原位，只要在攝影機的有效拍攝範圍內，都能辨識的出來。

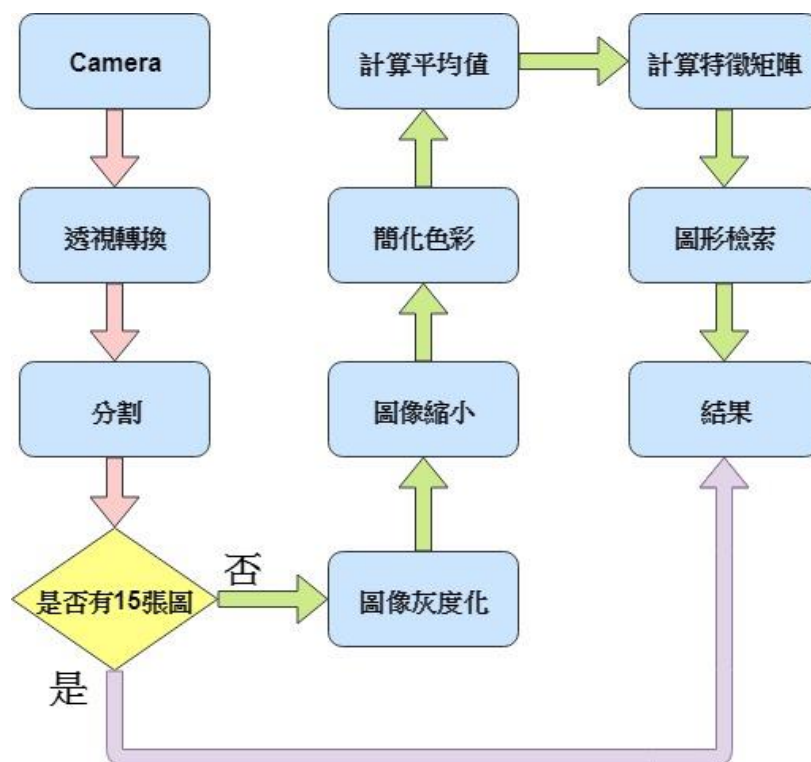


圖 4 影像辨識流程圖

再來就是，資料庫的部分。如【圖 5】所示。收到來自主程式輸出的 CSV 檔後，依據其內容更新舊有的資料。使用者，只要透過網頁，即可查看當前雜誌被借閱的情況，及被借閱的次數。

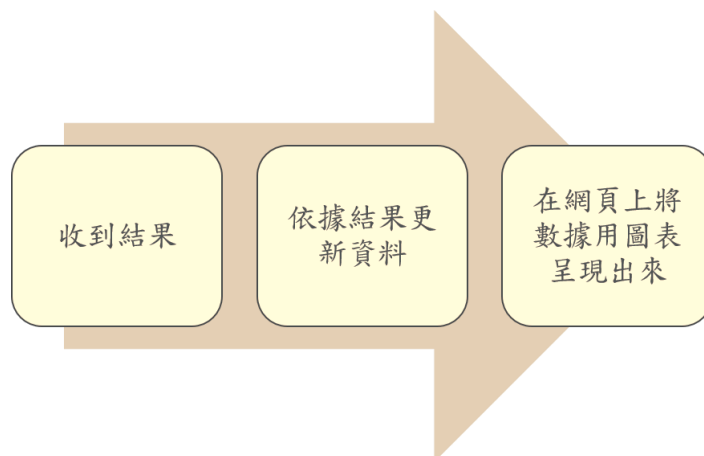


圖 5 資料庫運行架構

#### 四、 選定軟體及平台

一開始我們想使用近幾年滿火紅的 Raspberry Pi 3【圖 6】來當作主機使用，並在其自由度相當高的作業系統 Raspbian 上進行開發。但，在最後的測試中發現，再我們使用 1080p 畫質的相機後，它已經無法負荷如此龐大的運算，時常當機。為解決此問題，我們決定用一般的桌機來代替，也將平台改至 Windows。



圖 6 原先使用的 Raspberry Pi 3 外觀

#### 五、 影像辨識程式測試

撰寫完成試後，我們進行了測試，看是否能夠照著我們預想的流程圖走，如【圖 7】所示。

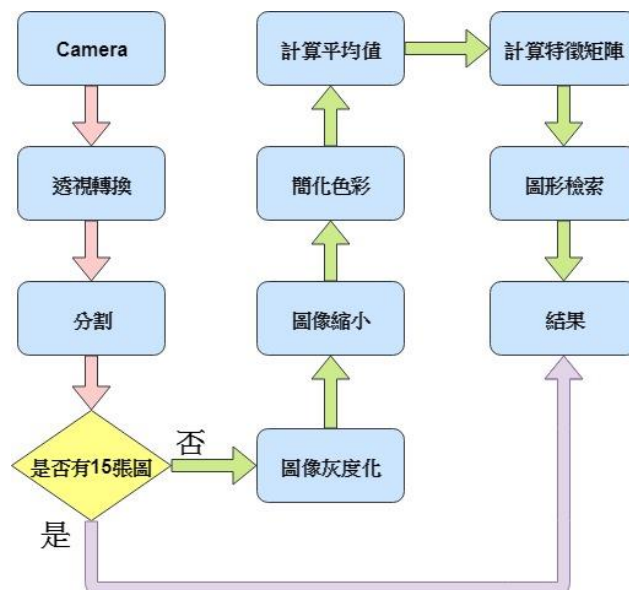


圖 7 影像辨識流程圖

### (一) 透視轉換

因為圖書館攝影機預定架設的位置是在天花板，所以會由往下拍，影像會變成類似梯形的矩形。而透視轉換，是將原本梯形的圖片拉平，好進行接下來影像處理，如【圖 8】所示。

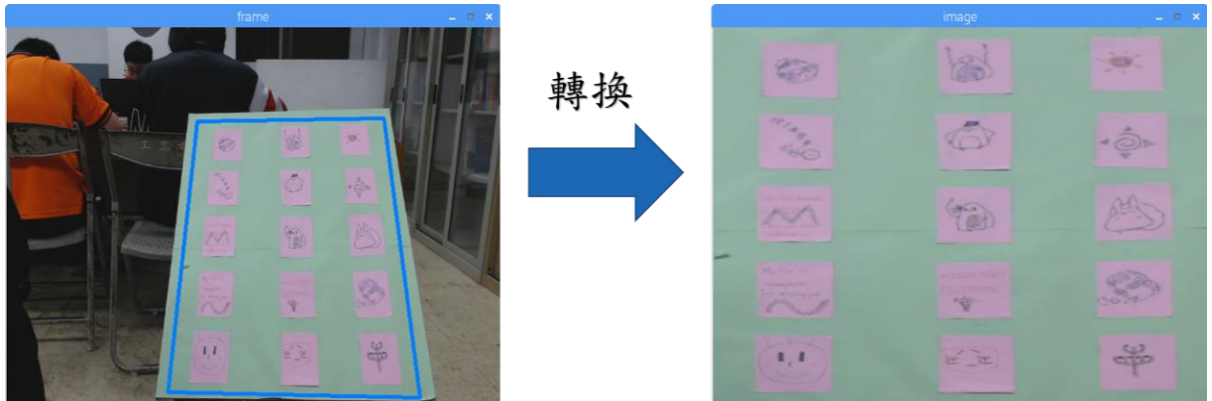


圖 8 透視轉換

### (二) 分割

是將我們所需要的圖塊截出來，方便之後的影像處理。首次執行完再次執行時，會先看有沒有 15 張圖，也就是雜誌架上的雜誌數。有，則表示沒有書被借走，直接輸出結果。沒有，則表示有書被借走，繼續往下處理，如【圖 9】所示。

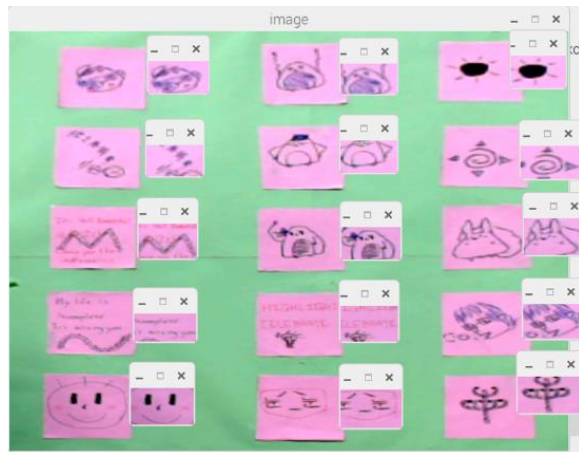


圖 9 分割

### (三) 圖像灰度化

圖像灰度化是為了，降低 cpu 負擔，加快運算的速度，如【圖 10】所示。

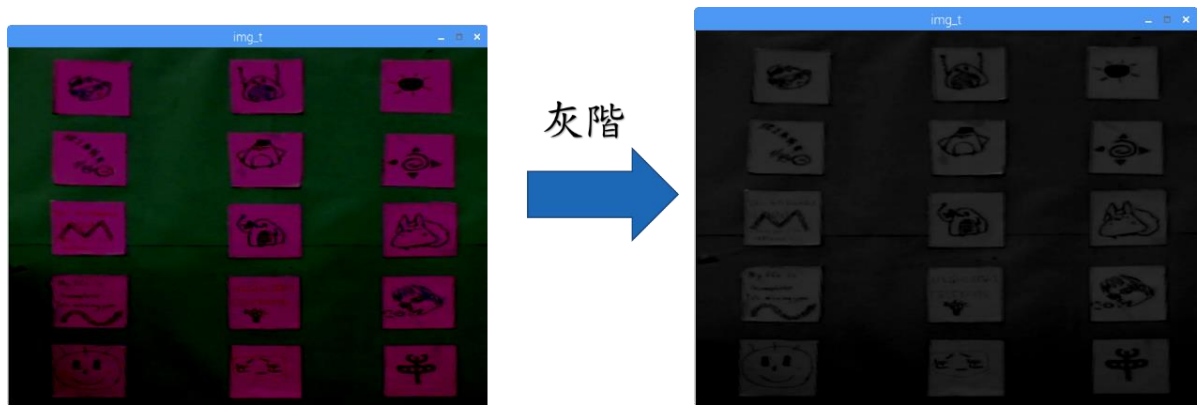


圖 10 灰階

#### (四) 圖像縮小

將已經截取大小為 400x150 的 15 張圖，縮小成 20x7.5，劃分成 20x20 大小互不相交的小區塊，計算每一個小區塊的平均值，將該值作為縮小影像對應位置的像素值。

#### (五) 簡化色彩

簡化色彩是為了允許在待檢索影像和靶心圖表面之間存在一定的色彩差異，能更進一步的加強系統的穩定性。經過灰階化後，目前的色彩是 256 級，簡化色彩後，影像的灰階級能固定在更小的範圍內。方便之後的特徵值計算。

#### (六) 計算特徵矩陣

當獲得縮小影像後，將所有的像素點的像素值相加求和，將結果除以像素點的總個數，即為像素平均值。在首次進行時，會將 15 張圖的特徵值存在一個資料夾內，構成一個圖庫，讓圖形檢索有一個比較的對象。

#### (七) 圖形檢索

即是將當前所截取並處理過的圖像特徵值，透過歐式距離計算，去尋找與圖庫中特徵值接近的圖形。並找出，哪幾本沒有被當前的特徵值所對應，即是被借走，不在架上的書，如【圖 11】所示。



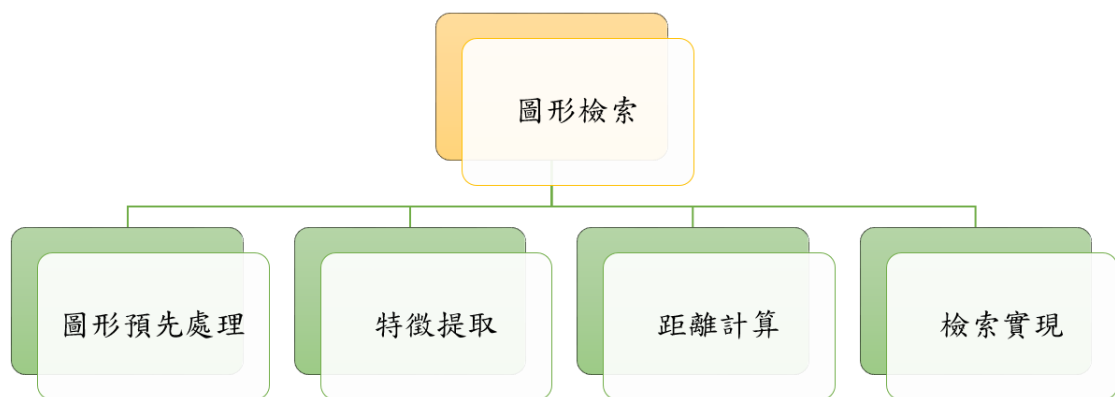


圖 11 圖形檢索流程

## (八)結果

和月初，初次執行所載的 15 張圖片，進行圖形檢索，檢查哪幾本被借走。並將結果，輸出成 CSV 檔。

## 六、 整合測試

確認影像辨識程式沒問題後，就將程式結合資料庫，進行最後的測試。首先，利用攝影機拍攝照片，再使用影像透視法，把它拉平，如【圖 12】所示。

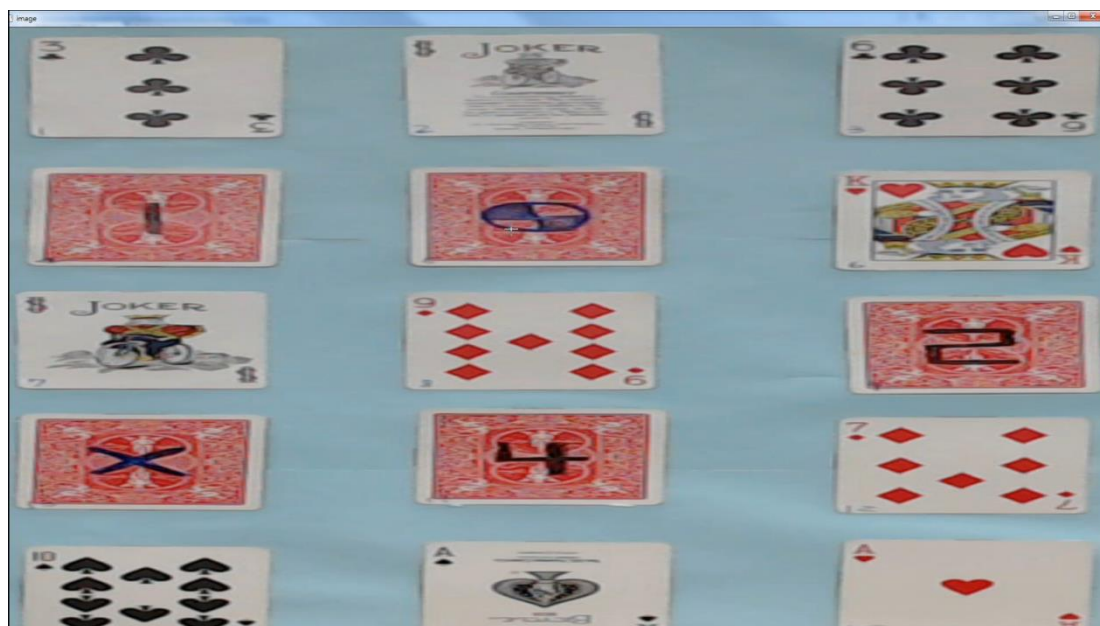


圖 12 程式開啟後，攝影機所拍攝到的畫面。

再來，將 x 記號的卡拿走，來模擬 1 本雜誌被借走。如【圖 13】所示。

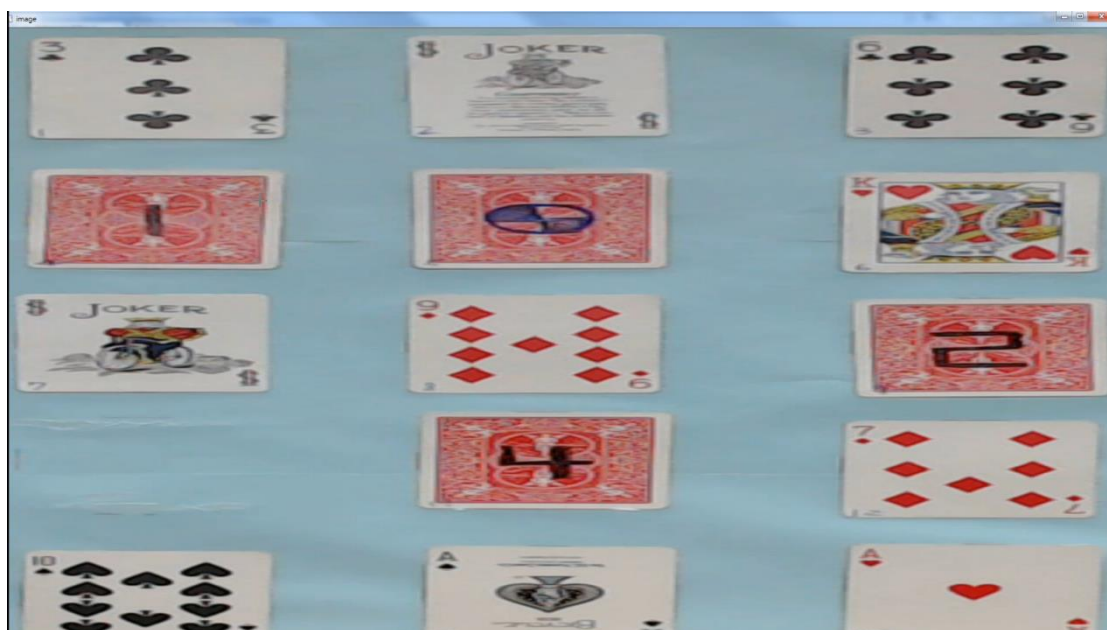


圖 13 第 2 張被貼掉後圖像。

等待運算完後，去我們架設的網站看一下數據有沒有更著變化。如【圖 14】所示。

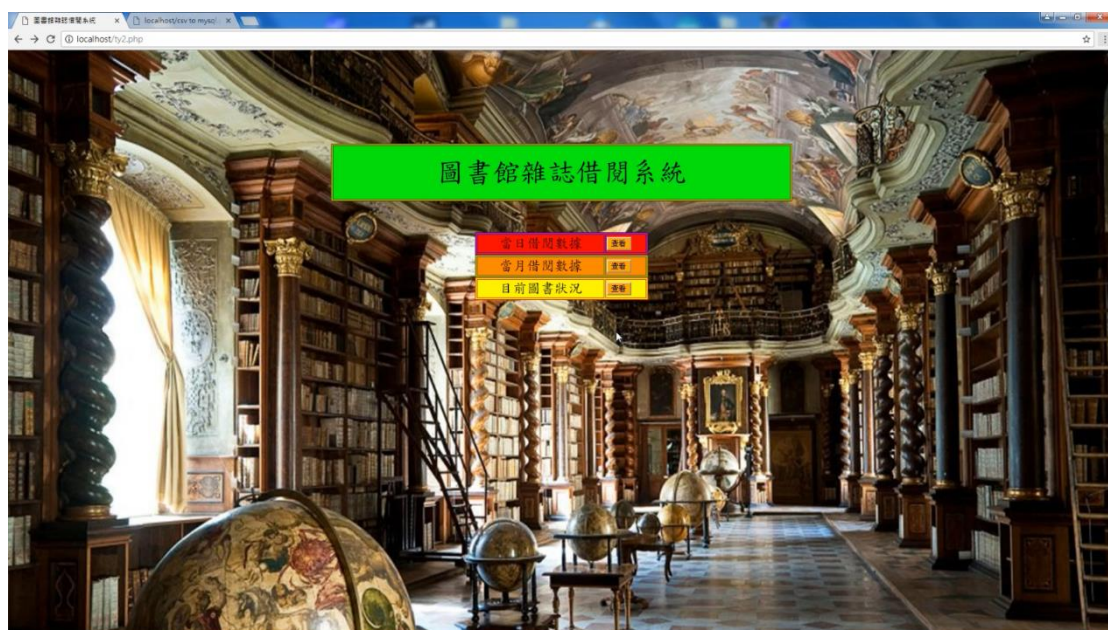


圖 14 我們所架設的網站首頁

當日借閱數據我們是用長條圖來呈現的，也可以正確的看出第 9 本書借閱量有 5。因為是測試為了方便觀察，書本被借閱一次，所上升的數據都被我們從 1 調整成 5。如【圖 15】所示。

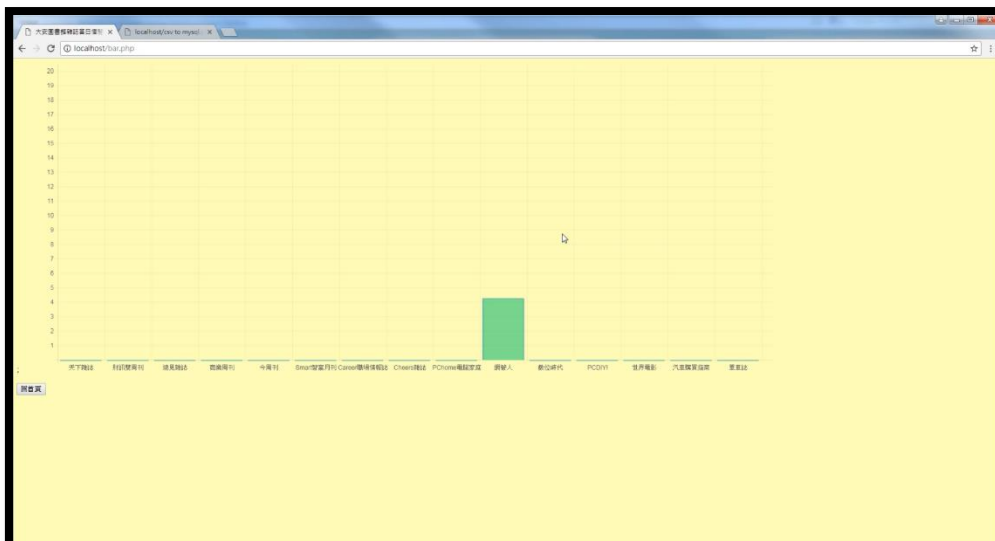


圖 15 當日借閱數據長條圖

目前的圖書狀況，我們則是用表來呈現。可以看出，第 9 本書的借閱次數有 5，狀態則是 x，代表正被借閱。如【圖 16】所示。

編號	名稱	借閱次數	行
0	天下雜誌	0	1
1	財訊雜誌	0	2
2	遠見雜誌	0	3
3	商業周刊	0	4
4	今周刊	0	5
5	Smart智富月刊	0	6
6	Career職涯情報誌	0	7
7	Cheers雜誌	0	8
8	PCHome電腦家庭	0	9
9	讀者人	5	X
10	數位時代	0	4
11	PCDIY	0	4
12	世界電影	0	5
13	汽車雜誌	0	5
14	星島	0	5

圖 16 目前圖書狀態表

接下來，我們要測試，雜誌亂放之後，系統能不能找到它在哪裡，並在表格中呈現。所以，我們隨便挑了 2 張牌對調位子，方塊 9(第 7 張)與 X 牌(第 9 張)，如【圖 17】所示。

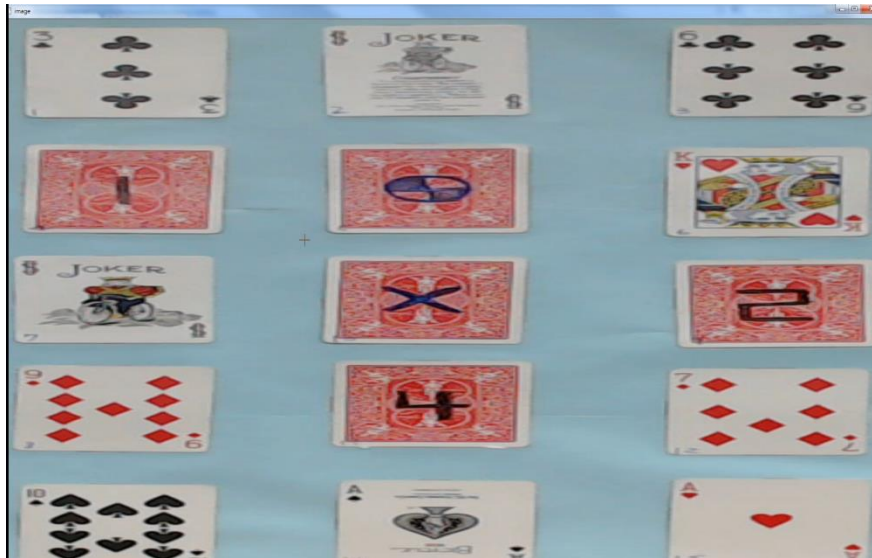


圖 17 交換位子後的樣子

然後去看看我們的表格的第 7 與第 9。可以看到他們現在的位置分別是，第 4 列第 1 行，第 3 列第 2 行，如【圖 18】所示。再看一下對照表，確認是否正確，如【圖 19】所示。

編號	名稱	借閱次數	列	行
0	天下雜誌	0	1	1
1	財訊雙周刊	0	1	2
2	遠見雜誌	0	1	3
3	商業周刊	0	2	1
4	今周刊	0	2	2
5	Smart智富月刊	0	2	3
6	Career 職場情報誌	0	3	1
7	Cheers 雜誌	5	4	1
8	PChome 電腦家庭	0	3	3
9	網管人	5	3	2
10	數位時代	0	4	2
11	PCDIY!	0	4	3
12	世界電影	0	5	1
13	汽車購買指南	0	5	2
14	單車誌	0	5	3

回首頁

圖 18 放大後的表格

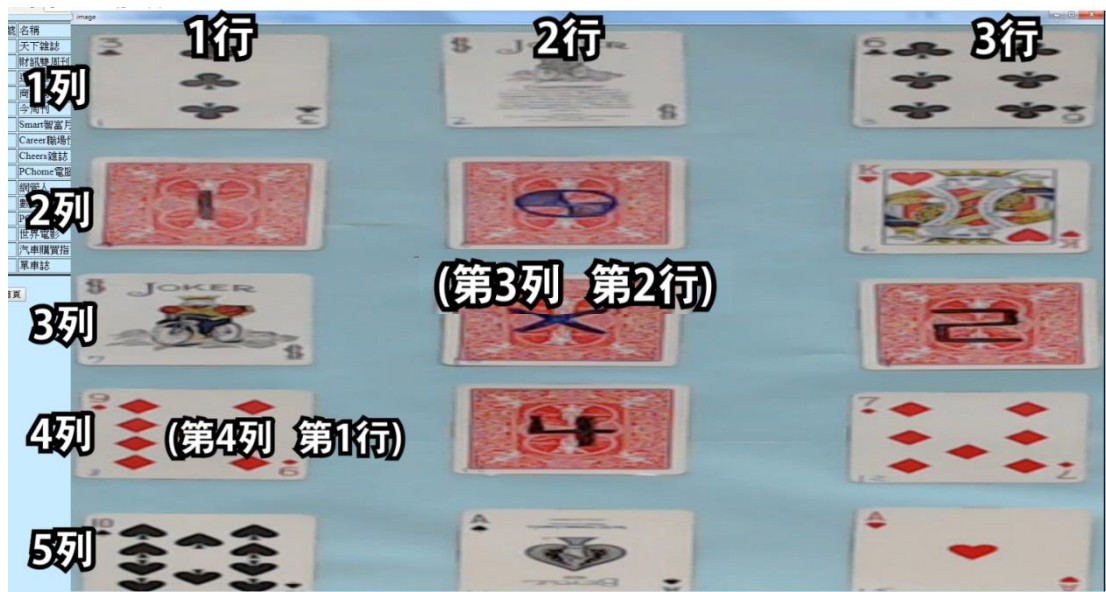


圖 19 列行對照表

在模擬成功後，我們前往圖書館實測。當然，我們失敗了，畢竟我們做的只是模擬系統。實際上，還有許多因素需要考慮，像是燈光和雜誌架的檔桿等。如【圖 20】所示。

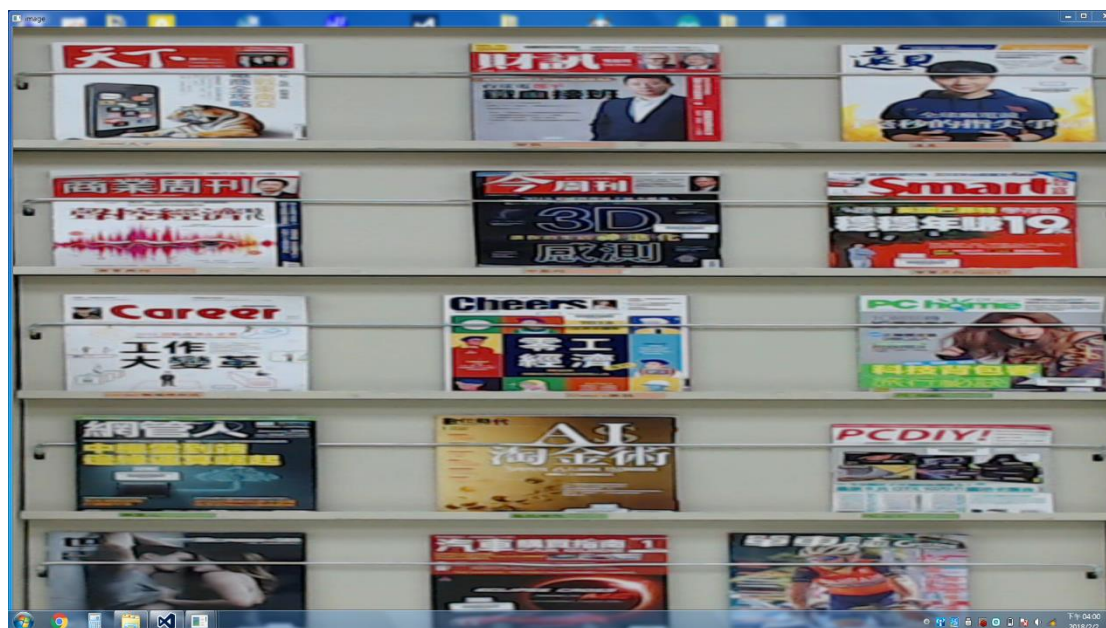


圖 20 圖書館實際測試

## 肆、研究結果

在這次的研究中，將影像辨識與資料庫做整合，完成了圖書館雜誌借閱模擬系統，雖然還無法在圖書館實際運行，但我相信在不久的未來一定能夠達到我們的3個目標，如【圖 21】所示。



圖 21 成品能達到的 3 大出發點

## 伍、討論

### 一、硬體方面所遭受的問題

#### (一) 攝影機

一開始使用的是原廠附的相機模組，解析度只有 640\*480，【圖 22】所示。礙於攝影機實地架設的距離會有點遠，怕畫質會太差，導致辨識上的錯誤。再與廠商討論後，拿了 1920\*1080 的攝影機來代替，如【圖 23】所示。



圖 22 Raspberry Pi 原廠設影機模組



圖 23 更換後的高畫質攝影機

## (二) 主機

原先使用的樹梅派，在 640\*480 的畫質下，運作還算順暢。但在更換完攝影機後，時常當機，硬體效能已經不堪負荷。所以，只能找效能相對強大許多的桌機或筆電來取代之。

## 二、 軟體方面所遭受的問題

### (一) 網頁程式

以前沒接觸過這方面的程式，所以得一步步摸索，剛開始只能用最基礎的白底黑字來製作，為了能讓我們的結果用圖表呈現，去尋找了許多的範例，來了解該如何製作，但要讓圖表跟數據同步，所以必須擁有與資料庫連線的功能，同時也要將影像辨識的結果上傳至資料庫才能達成，我們在上傳資料這點遇到了一些困難，最後是將影像辨識的結果輸出成 CSV 檔，再利用網頁程式將值上傳來解決這項問題。

## 三、 其它所遭受的問題

### (一) 圖書館雜誌架

圖書館的雜誌架中間有一條桿子，用來固定雜誌不滑落。而，那根桿子在實際測試時，也的確會影響辨識的準確度。另外，它的底色也與雜誌的顏色有些相近，導致偵測不到雜誌的情況。目前還在與學校討

論解決方針中，如【圖 21】。



圖 24 有明顯的欄杆，背景過白故加黑布測試。

## 陸、結論

我們在研究的過程中，原以為一些簡單的功能，難度不會太高。實際操作後才發現很多事情比我們想像中的難很多。導致程式撰寫及與資料庫連結上，遇到了許多困難，最後在我們嘗試無數次的實驗後，達成了目標。雖然，只能在模擬實驗上運行，實際上在圖書館還需要克服很多因素，像是燈光、雜誌櫃等等。不過，在做專題的過程中，我們了解到物聯網的概念，也終於知道為甚麼它會成為未來的一大趨勢。

由於我們出生在一个資訊爆炸的時代，使我們能透過網路找到許多相關的資訊，再加以應用，完成了這次的專題。製作過程中，雖然遇到了很多問題，但組員們都發揮各自的專長，一同克服了問題。最後，還得感謝我們的指導老師們和贊助商。當我們迷失在黑暗中，不知道該怎麼做時，老師們就像一盞燈，給了我們方向。再我們資金匱乏時，也有廠商願意贊助我們設備，盡力的提供我們實驗所需的設備。也因為這個專題的存在，我們學會了團隊合作，這類課本沒教的事。

未來，我們後續的目標是使該系統能夠在校內的圖書館裡運行。並從原本的網路攝影機(Webcam)，改為使用廠商提供的 IP Cam。前者採用有線連接，後者則是無線的。這樣就能夠使安裝更為方便，不用再拉一堆線。再來，就是使本系統的功能更趨近於完備，讓使用者介面更加簡潔美觀，完成一套能夠應用在全國各地圖書館的「雜誌借閱系統」。



## 柒、參考資料及其他

### 一、書籍

[1]望熙榮、望熙貴(2015)：OpenCV with Visual Studio 影像辨識處理。新北：博碩文化出版社。

[2]李立宗(2017)：OpenCV 開發一本搞定。臺北：佳魁數位發行。

### 二、網路參考資料

[1]正投影原理。民 106 年 10 月 25 日，取自：  
<http://el.fotech.edu.tw/localuser/c1216/web1/graph1/ch9.html>。

[2]透視轉換基本原理。民 106 年 10 月 25 日，取自：  
[http://140.112.94.11/~ttl/course01/lecture\\_notes/c1lecture\\_note03.htm#伍、影像幾合\(Image Geometry\)](http://140.112.94.11/~ttl/course01/lecture_notes/c1lecture_note03.htm#伍、影像幾合(Image Geometry))。

[3] [WDC] 淺談 Web Server 及 Application Server (上)。民 106 年 11 月 08 日，取自：<http://blog.ericisk.org/archives/662>

[4] MySQL 教程。民 106 年 11 月 08 日，取自：  
<http://www.runoob.com/mysql/mysql-tutorial.html>

[5]HTML5 Tricks。民 106 年 12 月 13 日，取自：  
<http://www.html5tricks.com/tag/html5%E5%9B%BE%E8%A1%A8/>

[6]阿洲的程式教學。OpenCV 教學。民 106 年 12 月 23 日，取自：  
[http://monkeycoding.com/?page\\_id=12](http://monkeycoding.com/?page_id=12)