第九屆旺宏科學獎

成果報告書

參賽編號：SA9-262

作品名稱：The Automatic ODCR Award-checking System using Machine Learning --
自動發票對獎系統

姓名：楊翔宇

關鍵字：影像辨識、機器學習、機器人

作品名稱

The Automatic ODCR Award-checking System using Machine Learning --

自動發票對獎系統

摘要

此研究以設計自動化發票對獎的系統（Award-checking System,簡稱 ACS）為目標。架設網路攝影機拍攝發票上的數字，經由自製的二值化演算法、雜訊處理以及字元切割方法，將數字逐一取出，擷取影像中特徵向量，經過訓練完成的「支持向量機」模組判讀影像中的數字，進而對獎。其中為了達到自動化對獎的效果，以微控制器操控機械手臂翻取發票，與電腦端使用遠端程序呼叫的通訊協定，進行系統整合。

實驗後得知於此系統上，以顏色為依據的二值化方法之正確率為 Otsu（最大類間方差法）演算法的 2.88 倍；且自製的光學數字字符識別（Optical Digital Character Recognition, 簡稱 ODCR）較一般 OCR(Optical Character Recognition)程式高出約 18%的正確率。

壹、 研究動機

一、 每回看見慈善機構擺設路邊勸捐發票的箱子，裡頭數千數百張的發票，必須花費不少人力去核對，「對發票」這個動作其實是反覆的、有規則的，正是應當交給電腦處理的顯著特徵。

二、 使用 OCR 程式所得的結果往往不如預期，欲提高辨識率必須經過某些特定的影像處理，而這些處理又可能因拍攝環境而有所不同。

三、 此一研究中，運用了數學上矩陣、微積分、三角函數、向量…等，以及物理上的光學、電學、靜力學知識，是把所學實際運用的難得機會。

貳、 研究目的

一、 創造一個適用於發票紙張上，對於 OCR 程式最佳的拍攝環境，進而推導至紙張上的數字辨識環境與程序。

二、 歸納進入攝影機的影像內容，找出一個能正確圈選、切割影像中數字的方法。

三、 於分割每一個數字為不同個體後，擷取特徵向量訓練機器學習(Machine Learning)模組，進而自製數字字元的 OCR 程式。

四、 設計一台可以自動對發票的機器，無須人為翻動發票。

參、 研究問題

一、 影像前置處理部分，使用不同的雜訊消除演算法對於 ODCR 的正確率提升程度如何？

二、 除了最常用的 Otsu（最大類間方差）二值化方式，有沒有更適合 ODCR 的二值化演算法？

三、 如何安排特徵向量的抓取，使之最能表達該圖形的特徵？

四、 機器設計上之光源、位置應該如何安排以增加影像的穩定性？

肆、 研究設備與器材

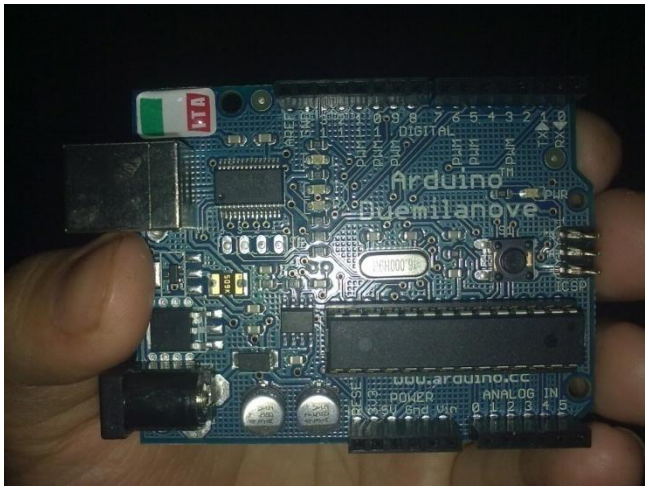
一、 硬體

（一） 電腦

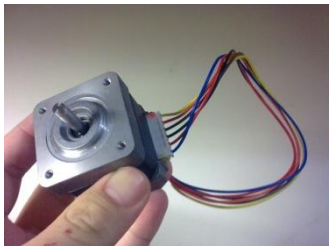
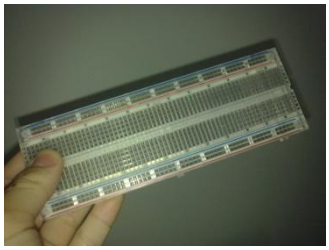
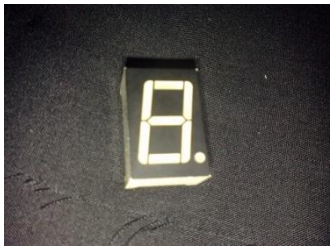
1. CPU：Intel(R) Core(TM)2 DUO CPU P8600 2.4GHz

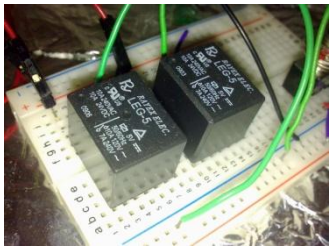
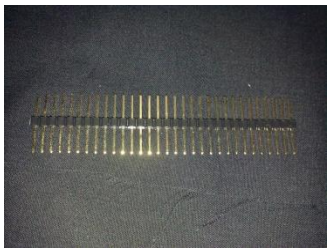
2. RAM：4GB

（二） AVR 微控制器：Arduino Duemilanove (使用 ATmega328p)



（三） 其他電子設備

		
網路攝影機(Logitech C300，1280*1024 像素)	伺服馬達(servo motor)：Futaba S3003	步進馬達：TAJIMA KH42JM2B128
		
麵包板	七段式顯示器	12V LED（白、綠、紅）

		
繼電器	變壓器	杜邦線
		
針腳	紙箱	發票(民國 98 年 5-6 月、7-8 月)

二、 軟體

（一） 作業系統：Fedora 11(Kernel Linux 2.6.30.10-105,2,13,fc11,i686.PAE , GNOME 2.26.3)

（二） python 相關軟體

1. python 2.6
2. ipython 0.9.1
3. vpython 5.13
4. pyserial 2.4

（三） opencv-python 1.0.0

（四） opencv 1.0.0

（五） convert ImageMagick 6.5.1-2 2010-01-06 Q16

（六） tesseract 2.04

（七） gcc 4.4.1

（八） fswebcam

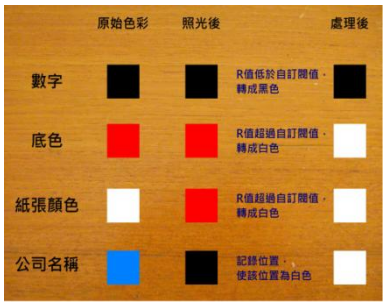
（九） libsvm 2.9

（十） 單晶片控制相關軟體：avrdude 5.8、avr-gcc 4.3.3

伍、 實驗過程與方法

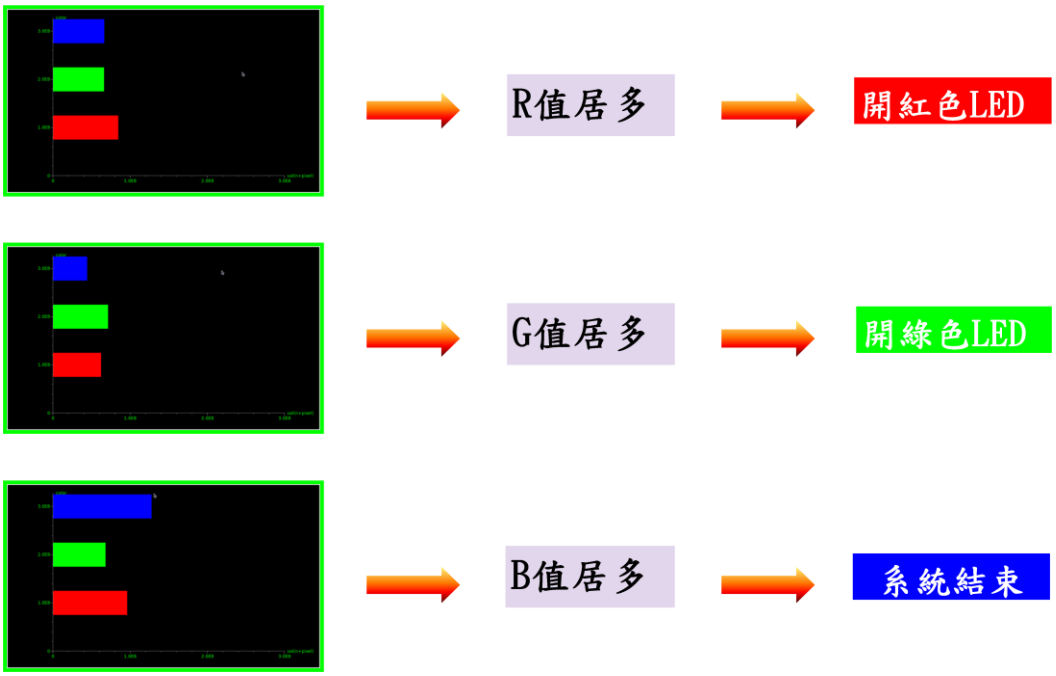
一、 光線安排

一般我們看到自然光源下的發票有白、藍、黑、紅，相當雜，而將光源改成強烈的紅光，如此一來，單純化影像內容，最終只剩下黑與紅。記錄白光下出現藍色商店名稱的位置，並在紅光下的同一位置直接去除黑點，影像中呈現黑色的只剩下欲求的發票號碼。

<table>
<tr><td> 
</td><td> 
</td></tr>
<tr><td>原始圖（白光、自然光）</td><td>原始圖（紅光）</td></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td>去除藍色</td><td>二值化示意圖</td></tr>
</table>

二、 影像色彩比例分析

發票底色主要為紅色系與綠色系，依照底色而打上不同的光，如此可以單純化影像中的資料，並依色彩比例的不同做出不一樣的反應，同時也安排結束記號(藍色)。



三、 剪裁

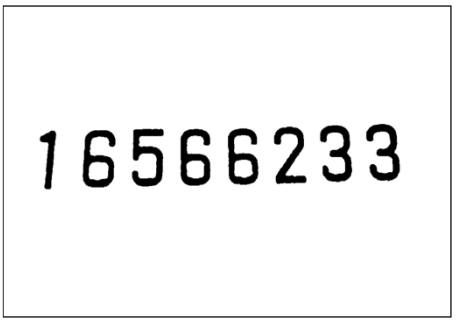
影像中，數字置於中央，故將周圍易產生陰影或是光線不均勻處除去。

剪裁前

剪裁後

四、 二值化

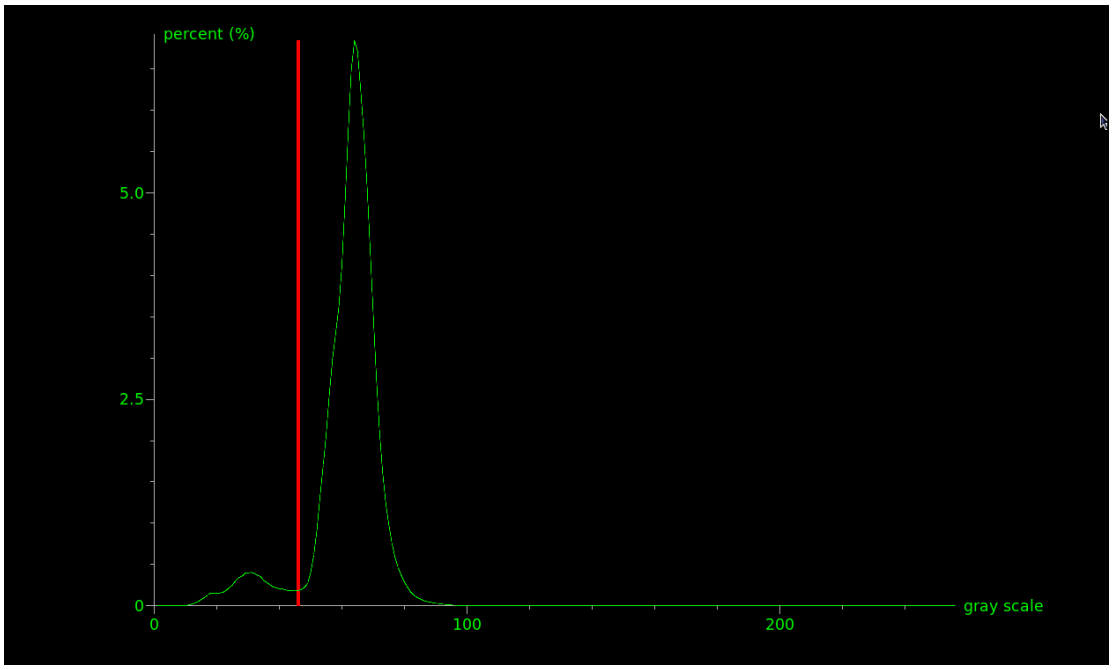
Otsu 演算法是最常用的二值化之閥值(threshold)取得方法，使用最大類間方差，可得到一個大部分人都會滿意的 threshold，然而，Otsu 對目標物大小敏感、遇到多個峰（指灰階值與其所佔的比例之圖上）或雜訊時，表現變得差勁，於 ACS 中，設計依賴色彩資訊之監督式二值化方式與 Otsu 比較、分析。



（一） Otsu 演算法

1. 介紹

Otsu 為一個非監督式(unsupervised)的二值化演算法（灰階影像轉黑白影像），也就是說此方法無須人為手動調整參數，線性地取最大類間方差便可以自動化取得一個合理的閥值，區分前景與背景，進而二值化。



2. 原理

上圖表示各灰階(0~255)出現的機率，此演算法將取兩座山峰中的最低處。

實際撰寫程式上，我們變數設定如下：

(1)p[灰階值]：為一陣列，代表灰階值（索引值）出現的機率

(2)

整個影像	前景	背景	代表意義
N	n1	n2	畫素總數
Sum	Csum	sum-csum	質量矩（灰階值*其像素數）
	m1	m2	平均灰度（質量矩/畫素總數）

使 k 為前景與背景的臨界值，從 0 至 255 一維線性的方式於迴圈中運算，求最大類間方差，也就是使前景與後景有最大的方差。

設前景、後景像素數佔全影像的比例為 w1,w2，圖像中總平均灰度為 u，而得：

$$u = w1 \times m1 + w2 \times m2$$

前景和背景的方差：l

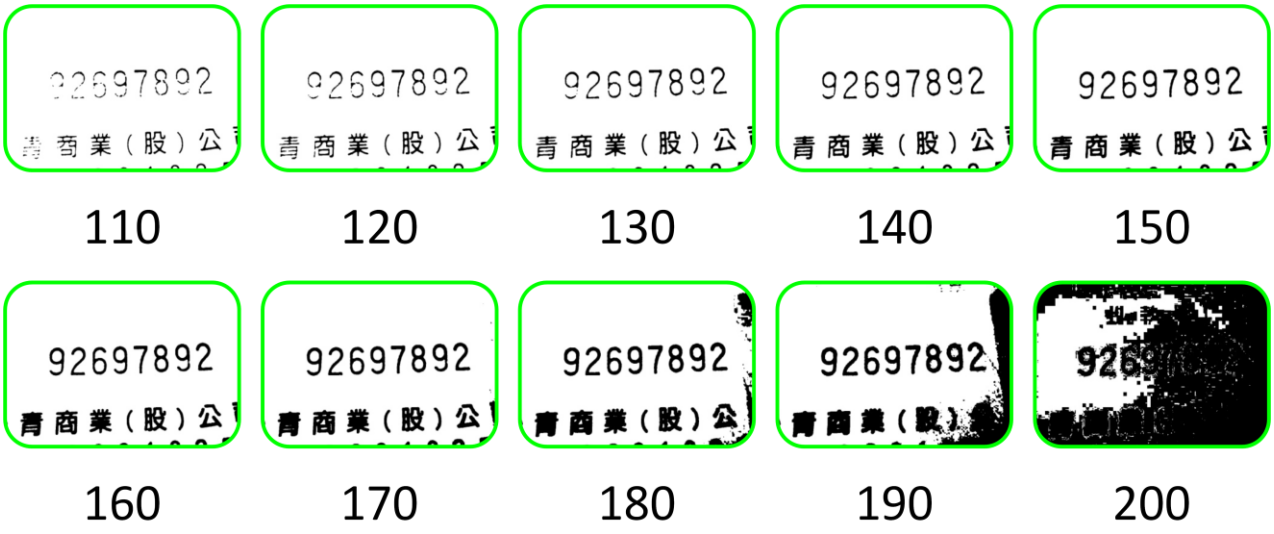
$$g = w1 \times (m1 - u)^2 + w2 \times (m2 - u)^2$$

參照機率的公式可得：

$$g = w1 \times (m1 - u)^2 + w2 \times (m2 - u)^2$$
$$= w1 \times w2 \times (u1 - u2)^2$$

（二） 依賴色彩資訊之監督式二值化

Otsu 演算法中是於灰階分布表中運算，已經失去了彩色的訊息，而發票上，數字底色與數字本身之顏色是截然不同的，然而往往取灰階值時導致顏色不同的 pixel 灰階值卻變得與部分陰影、雜訊相近，且 ACS 中以強烈的光線照射下，不同於光源的值並沒有意義，例如：紅光照射下，便是希望影像中只有紅與黑，於是 G 值與 B 值此時並沒有意義，可視為雜訊，故我們閥值只針對 R 值。



針對 R 值調整閥值

五、 消除雜訊

		
原始圖	中值濾波	高斯模糊

（一） 中值濾波

1. 理論：此為基於排序理論中有效抑制椒鹽噪音和斑點噪音的經典技術，其原理是每一個點的值以其周遭某區域內中的中位數代替，如此可以將孤立的雜訊點消除。
2. 實踐方法：

(1)將區域內的值取出，進行排序。

(2)排序後用中位數去取代該區域內的所有值。

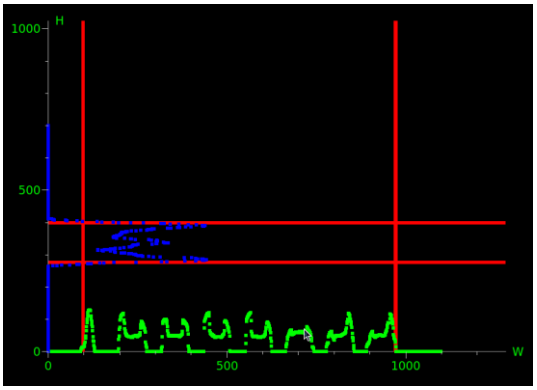
（二） 高斯模糊

為一種模糊濾波器，使每一個像素做常態分佈（高斯分佈）， 如此低能量的雜訊可被清除，但高能量的前景也會變淡。

六、 切割字元

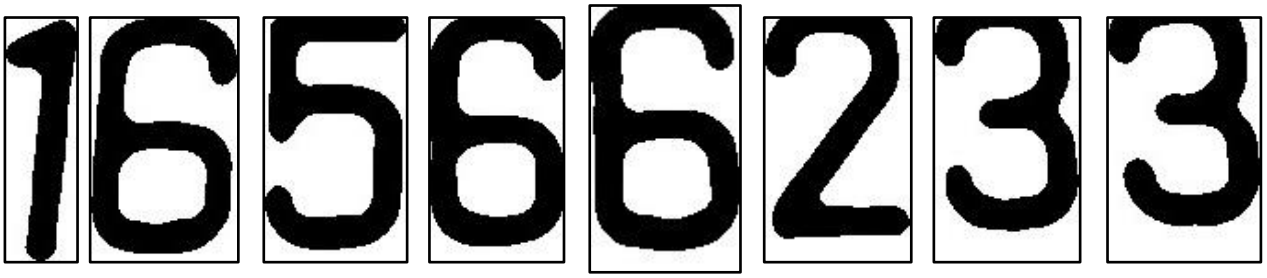
（一） 步驟一：去除數字以外的影像內容

使一條水平線由上而下掃描，若是黑點的分佈為等差，則表示為數字的位置，該位置設為起線，往下繼續掃描直到沒有黑點的出現為止，如此便可以圈出數字的位置。另外判斷抓到的前兩位數字在中線的右邊或後兩位數字在中線的左邊者，該狀況略過，如此可以減少雜訊的影響。



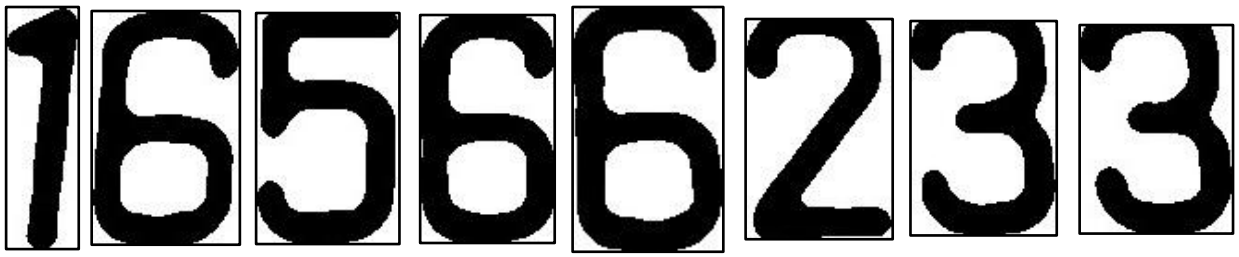
（二） 步驟二：切割出每一個數字

於步驟一分割後的影像中，使一條垂直線由左往右掃描，黑點出現時設為起點，黑點結束出現時設為終點，由此分割出八個數字。



（三） 步驟三：邊界貼齊數字

由於之後 ODCR 將對從數字影像中擷取特徵，其中包含對稱性等，使邊界貼齊數字，盡可能使每一張圖片的影像狀態相近，擷取後的特徵較具有意義。



七、 機器學習

（一） 簡介：

機器學習(Machine Learning)乃機器模擬或實現人類學習行為的技術，經由特定的學習，從學習的數據中提取規則或模式，面對未遇過的問題便可以做出合理的反應，是人工智慧的子領域。機器學習將面對的問題可分為：回歸與分類。

1. 回歸：像函式的概念，期待自變數 X 可求得對應的應變數 Y。
2. 分類：將輸入值歸類至所屬的區域

學習的方法有很多種，沒有絕對好壞，可分為：監督式、非監督式、半監督式…等。

（二） 常見學習方法

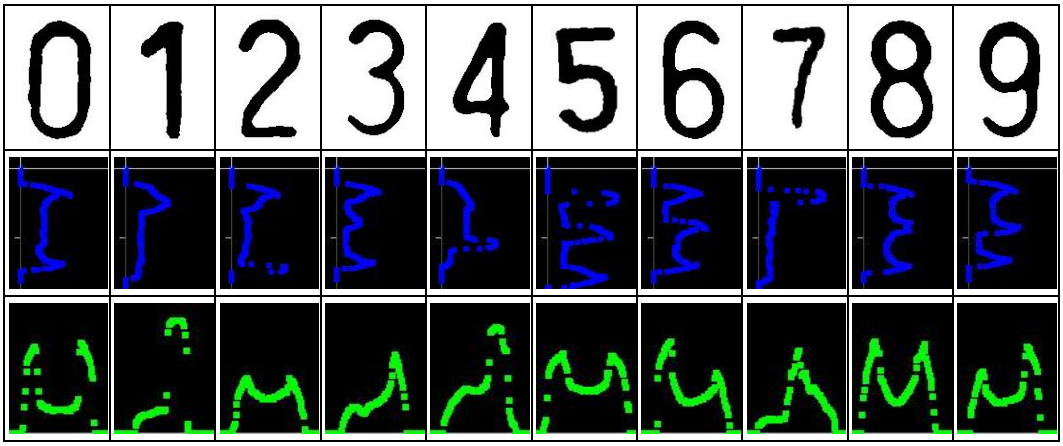
1. 類神經網路(Neural Network)：此為一個模擬人類神經的運算模型，由許多節點組成。
2. 決策樹：為一監督式的分類器，僅有單一種輸出，速度快。
3. 支持向量機(SVM)：為監督式學習，像是線性分類器的推廣，數據不再限制於平面向量，而將向量映射到更高維的空間裡，於分隔數據的超平面兩邊各建一個平行超平面，當兩個超平面有最大距離時，該平面則為「最大間隔超平面」，即為所求。ACS 使用 SVM。

（三） 擷取影像中特徵向量

機器學習的訓練(train)、測試(test)與預測(predict)上，我們都必須將一張圖檔的訊息轉成數值，而一連串有序的數值成為向量，此一由特定影像所對應的向量稱之為特徵向量。

此研究中，將設計針對數字影像擷取特徵向量，目前規劃以下幾種特徵向量：

1. 上下對稱性
2. 左右對稱性
3. 8 條鉛直分割線所切到的黑點數所佔的比例
4. 16 條水平分割線所切到的黑點數所佔的比例
5. 將影像分割成 3*5 的區塊，每一個區塊黑點數所佔的比例



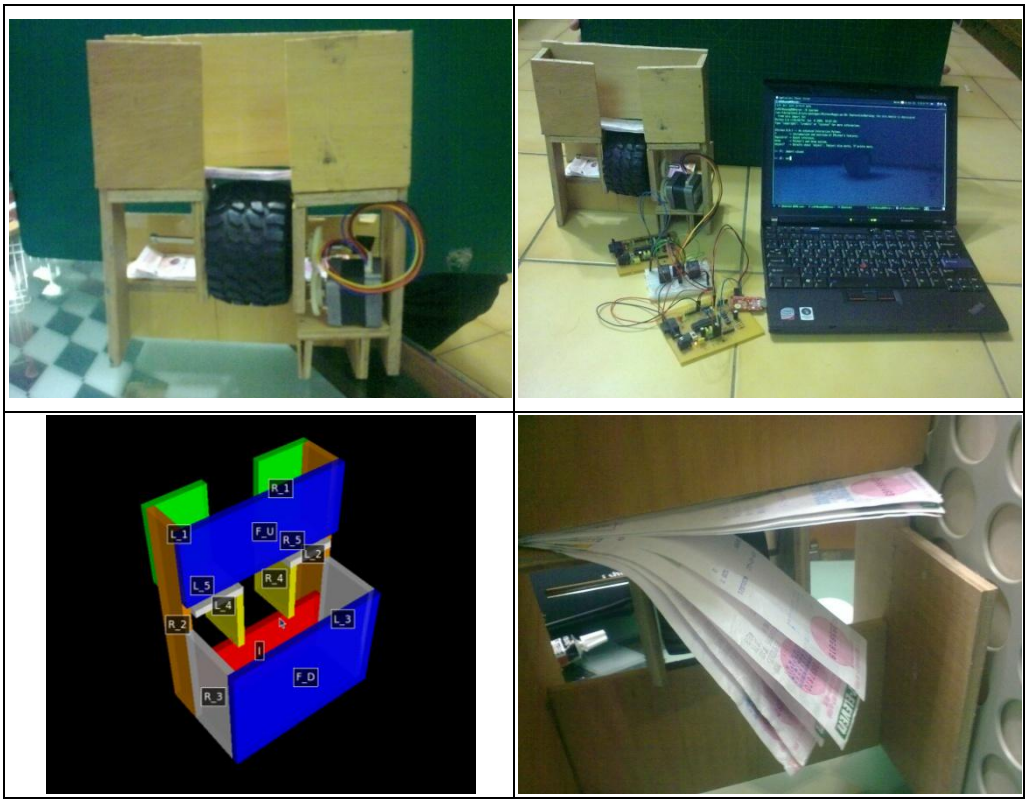
此表格中藍色的波表示一水平線由上而下掃描，每一水平線所擁有的黑點數，綠色的波則為使用垂直掃描，上述特徵向量(3)、(4)則是對這兩種波進行採樣。

八、 機器部分

（一） 設計目的：使發票依序出現於攝影機前。

（二） 設計目標：拍攝環境穩定（光線穩定、距離固定、位置固定）。

（三） 第一代至第三代機器：模擬點鈔機運作模式



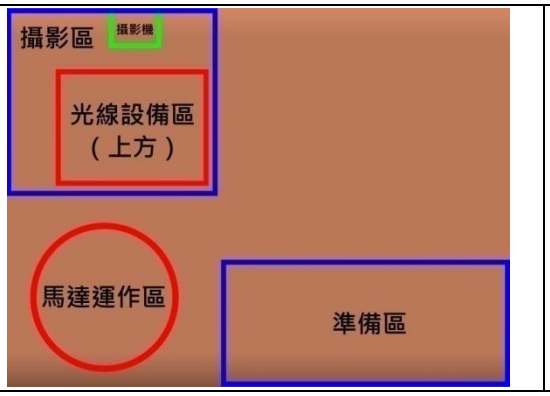
由於發票之間的摩擦力不固定、紙張大小不盡相同、機器材質選擇等問題，導致發票出來的狀況難控制，有「同時張多出來」、「轉不出來」、「斜向出來」等問題出現。

（四） 第四代機器

發票之間的摩擦力是前三階段沒有成功的最大問題，而造成此摩擦力的主要原因是重力或是夾住的力量造成的，因此若是由上而下抓取發票，

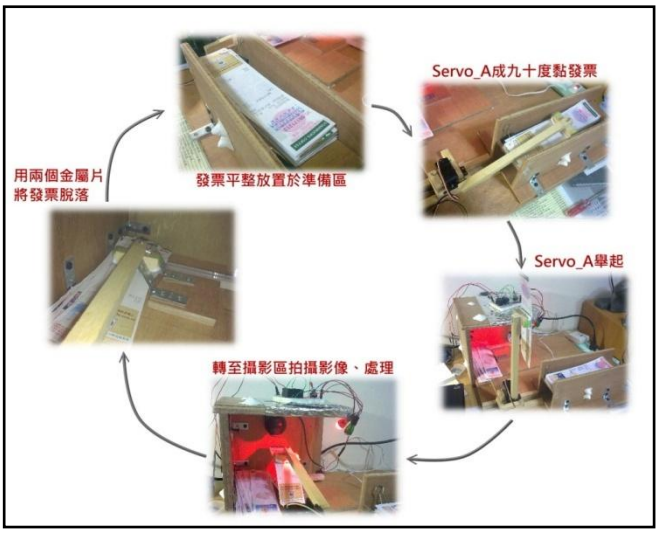
便可以避免，故設計一個簡易的機器手臂提取發票。

為了達到可以垂直提取的動作，便安排一個伺服馬達處理垂直的動作；為了讓發票可以旋轉至攝影機下方，設計第二個伺服馬達負責做水平旋轉的動作。處理重複黏貼的問題，我們使用具有此性質的便利貼，實際操作上可以使用上百次。



主要分為四個區域：

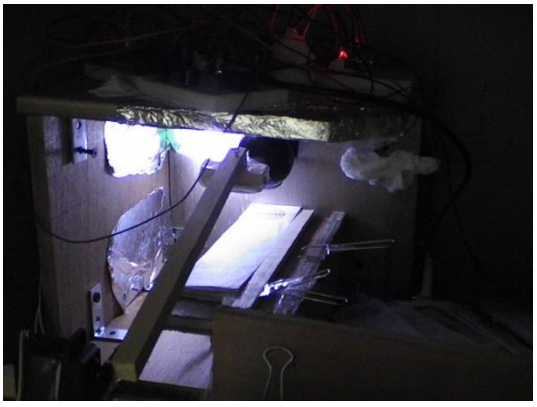
1. 準備區：尚未對獎發票堆疊處。
2. 攝影區：發票於此呈現在攝影機前，此區也是光線設計的重要地方。
3. 光線設備區：放置照光設備，包含 LED、繼電器、開關、電源，位於攝影區上方。
4. 馬達運作區：馬達旋轉的區域。

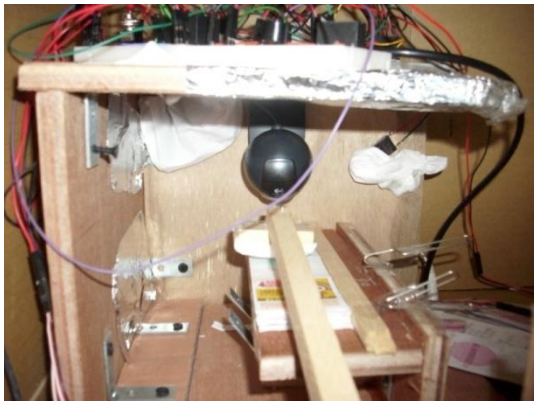


機器運作流程與方式示意圖

（五） 第五代機器

減少機械手臂運動的範圍，達到省時、高速的效果，發票堆疊於攝影機下方，再依序抽去。影像也較第四代穩定，亦改善第四代機器因為機械手臂旋轉路程遙遠導致發票下垂、掉落等狀況。





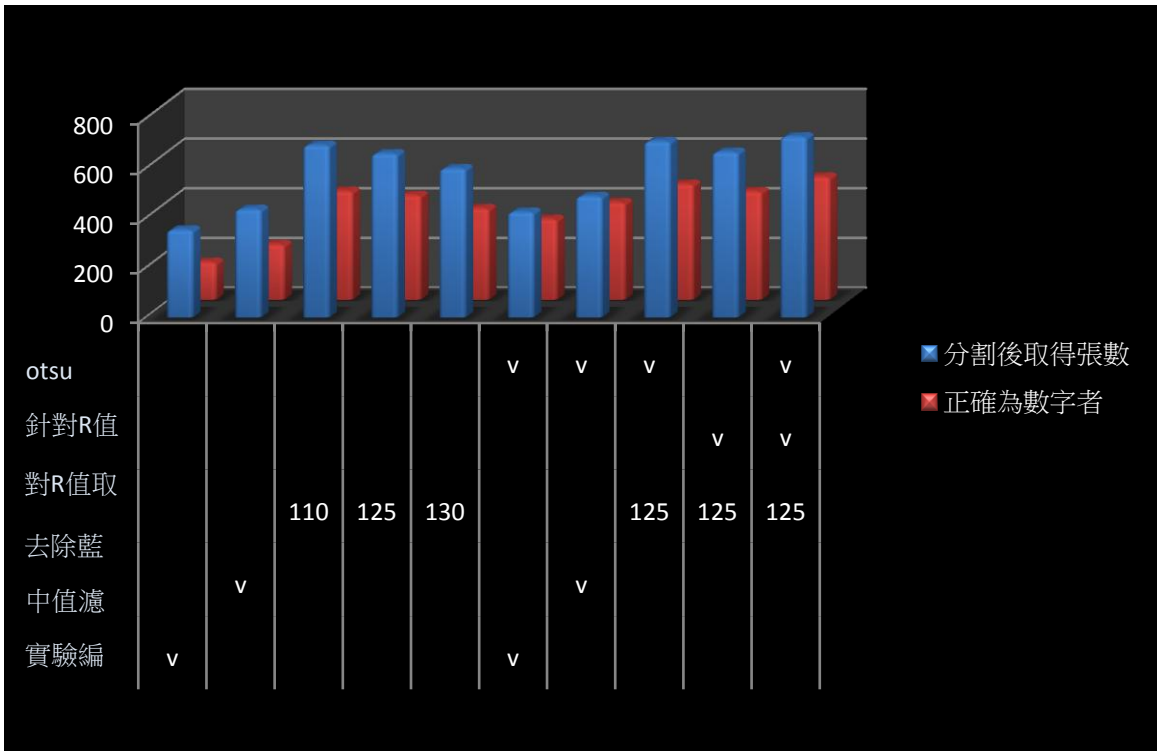
陸、 討論

一、 影像處理部分

（一） 數據

1. 第四代機器得：

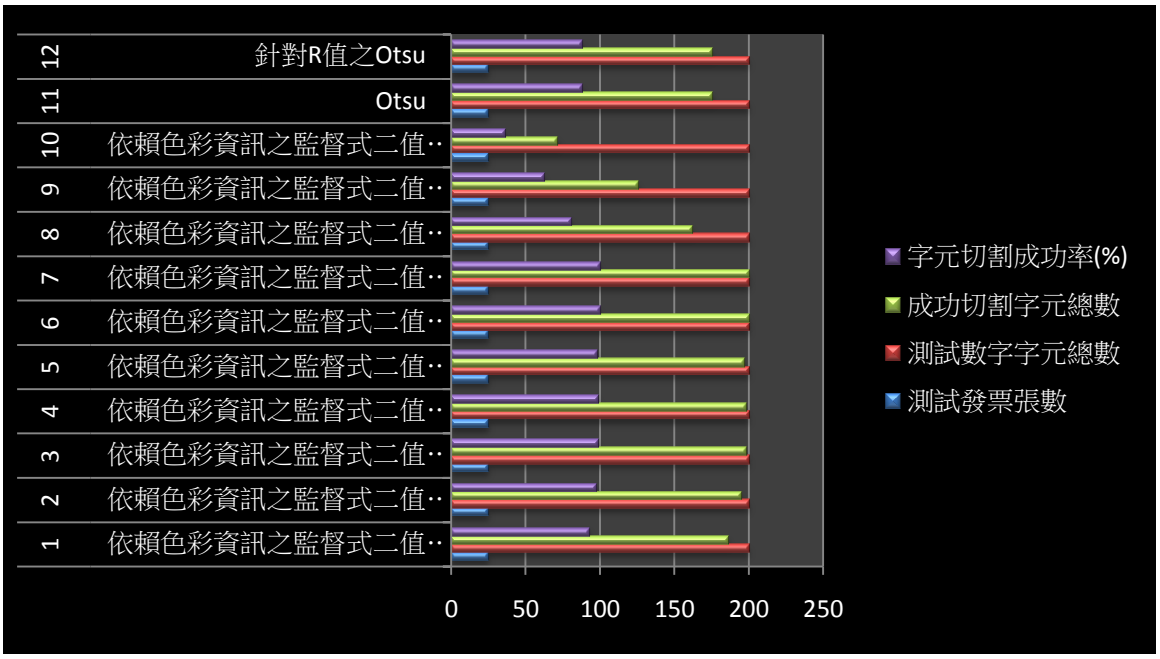
編號	otsu	針對 R 值之 otsu	對 R 值取閥值	去除藍色	中值濾波	分割後取得張數	正確為數字者
1	v					353	152
2		v				436	223
3			110			694	438
4			125			660	421
5			130			600	368
6	v				v	425	325
7		v			v	488	392
8			125		v	709	466
9			125	v		666	436
10			125	v	v	728	495



2. 第五代機器得：

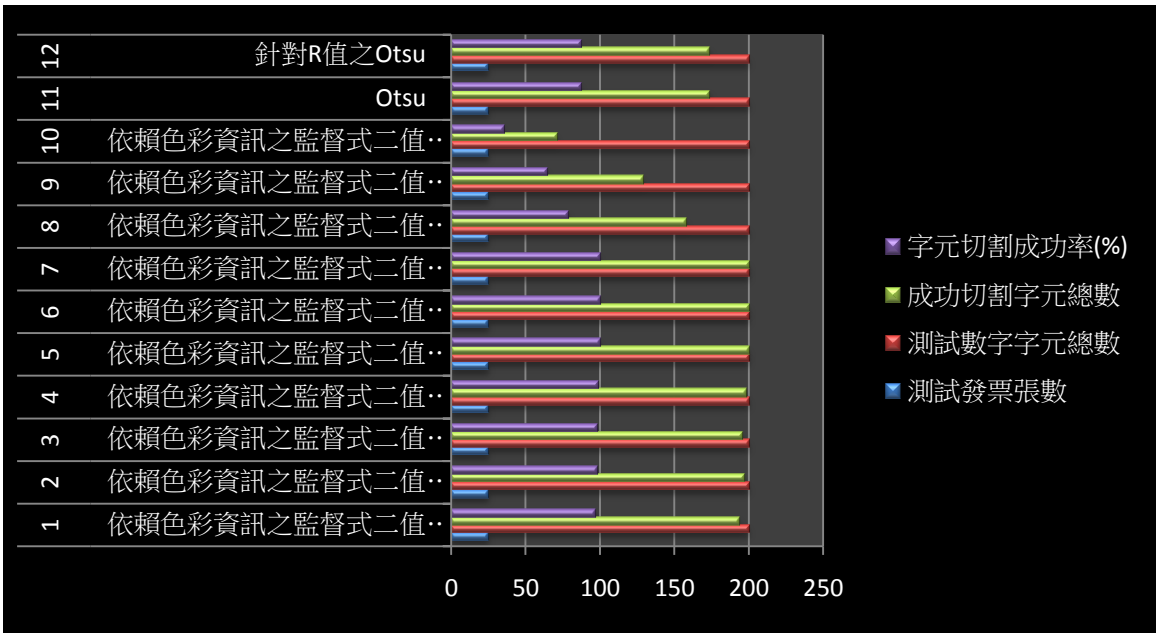
使用中值濾波：

編號	二值化方式	測試發票張數	測試數字字元總數	成功切割字元總數	字元切割成功率(%)
1	依賴色彩資訊之監督式二值化，閥值=110	25	200	186	93
2	依賴色彩資訊之監督式二值化，閥值=120	25	200	195	97.5
3	依賴色彩資訊之監督式二值化，閥值=130	25	200	198	99
4	依賴色彩資訊之監督式二值化，閥值=140	25	200	198	99
5	依賴色彩資訊之監督式二值化，閥值=150	25	200	197	98.5
6	依賴色彩資訊之監督式二值化，閥值=160	25	200	200	100
7	依賴色彩資訊之監督式二值化，閥值=170	25	200	200	100
8	依賴色彩資訊之監督式二值化，閥值=180	25	200	162	81
9	依賴色彩資訊之監督式二值化，閥值=190	25	200	126	63
10	依賴色彩資訊之監督式二值化，閥值=200	25	200	72	36
11	Otsu	25	200	176	88
12	針對 R 值之 Otsu	25	200	176	88



使用高斯模糊：

編號	二值化方式	測試發票張數	測試數字字元總數	成功切割字元總數	字元切割成功率(%)
1	依賴色彩資訊之監督式二值化，閥值=110	25	200	194	97
2	依賴色彩資訊之監督式二值化，閥值=120	25	200	197	98.5
3	依賴色彩資訊之監督式二值化，閥值=130	25	200	196	98
4	依賴色彩資訊之監督式二值化，閥值=140	25	200	198	99
5	依賴色彩資訊之監督式二值化，閥值=150	25	200	200	100
6	依賴色彩資訊之監督式二值化，閥值=160	25	200	200	100
7	依賴色彩資訊之監督式二值化，閥值=170	25	200	200	100
8	依賴色彩資訊之監督式二值化，閥值=180	25	200	158	79
9	依賴色彩資訊之監督式二值化，閥值=190	25	200	129	64.5
10	依賴色彩資訊之監督式二值化，閥值=200	25	200	71	35.5
11	Otsu	25	200	174	87
12	針對 R 值之 Otsu	25	200	174	87

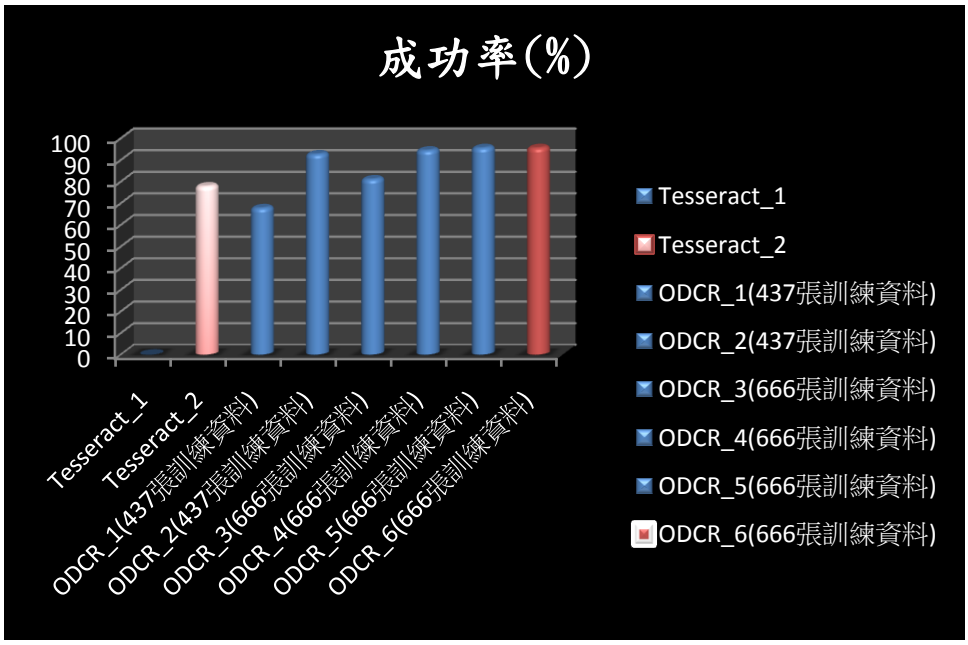


（二） 數據討論：

1. 第四代機器所得數據中，去除藍色可以提升約 2~3%的正確率。雖然此方法可以有效去除影像中的商店名稱，但由於我們取數字的方式是由上而下掃描，故掃描到數字之後變結束，於是下方的店家名稱影像並不大。
2. 第四代機器所得數據中，中值濾波約可以提升 5~20%的正確率。有效去除雜訊之後，可以避免在抓數字位置時受到影像。
   第五代機器得知，中值濾波、高斯模糊對整體辨識率的貢獻相近，高斯模糊略高一些。
3. 第四代機器所得數據中，若只針對 R 值做 Otsu 二值化，比一般 Otsu 的效果好 1.467 倍，因為在紅光下，只需針對 R 值，其餘值可斷定為雜訊。然而以上第五代機器所得之數據中，不論是否只對 R 值，其正確率是一樣的，可見改善機器後，雜質極少。
4. 關於影像數字的濃淡，除了可藉由二值化時閥值的調整外，機器中光源的強弱，還有攝影機亮度(brightness)參數的調整都會影響二值化後數字的濃淡，實驗中調整三個值的大小後，尋到一個合適的值即可。
5. Otsu 演算法為非監督式，可以廣泛適用於多種場合並且普遍有良好的表現，但在環境固定下，閥值固定的方式往往有更好的表現。另外，Otsu 演算法已經失去顏色的意義，而顏色在某些情境下卻是可以容易分辨出前景、後景的依據。實驗得知，自製依色彩為依據之二值化方法的成功抓取率高 Otsu 演算法的 12%。

二、 辨識部分

	參數	切割出每一個數字	成功率(%)
Tesseract_1	*	X	0
Tesseract_2	*	V	77.88
ODCR_1(437 張訓練資料)	*	V	67.78
ODCR_2(437 張訓練資料)	-c 20	V	92.67
ODCR_3(666 張訓練資料)	*	V	81.11
ODCR_4(666 張訓練資料)	-c 128	V	94.44
ODCR_5(666 張訓練資料)	-c 128 -g 0.5	V	95.56
ODCR_6(666 張訓練資料)	-c 8 -g 0.5	V	95.56



註 1：Tesseract 為 Google 公司所製作的 OCR 程式

註 2：以 180 張影像測試

一般 OCR 程式針對較廣、變化較多的字體、語言環境設計，且在判斷上有「參考資料庫是否有此一單字」…等其他方式提高正確率，自己設計的 ODCR 為 ACS 環境所設計，訓練時也是對此環境下的影像學習，故在此環境下表現可達 95.56% 的正確率，較一般 OCR 高約 18%。

柒、 結論

一、 第四代機器得：

（一） 去除藍色可以提升約 2~3%的正確率。

（二） 中值濾波約可以提升 5~20%的正確率。

（三） 自製依色彩為依據之二值化方法的正確率為 Otsu 演算法的 2.88 倍。

（四） 若只針對 R 值做 Otsu 二值化，比一般 Otsu 的效果好 1.467 倍，因為在紅光下，只需針對 R 值，其餘值可斷定為雜訊。

二、 第五代機器得：

（一） 拍攝影像穩定性改善之後，二值化演算法的成功率皆提高。

（二） 對灰階值、對 R 值之 Otsu 二值化效果相同，表示環境中雜訊極少。

（三） 依色彩為依據之二值化方法的正確率較 Otsu 演算法高 12%。

（四） 中值濾波、高斯模糊對整體辨識率的貢獻相近，高斯模糊略高一些。

（五） 自製 ODCR 較 Google 公司的 Tesseract OCR 程式成功率高約 18%。

捌、 參考資料及其他

一、 參考資料

（一） AlpaydinEthem. (2009 年 06 月). 机器学习导论. 机械工业出版社.

（二） Arduino. (無日期). 擷取自 Arduino: http://www.arduino.cc/

（三） BradskiGary. (2009 年 9 月). 学习 OpenCV（中文版）. 清华大学出版社.

（四） Google. (無日期). tesseract-ocr . 擷取自 tesseract-ocr : http://code.google.com/p/tesseract-ocr/

（五） PikeW. Kernighan and RobBrian. (1984). The Unix Programming Environment. Prentice Hall, Inc.

（六） 林弘德. (無日期). piaip's Using (lib)SVM Tutorial. 擷取自 piaip 的 (lib)SVM 簡易入門: http://www.csie.ntu.edu.tw/~piaip/docs/svm/

（七） 张广军. (2005 年 6 月). 机器视觉. 科学出版社.

二、 未來期望

（一） 增設更多的特徵向量截取、更多機器學習所需的影像以提升判斷正確率。

（二） 整合系統，並針對不完善的影像輸入做出對應的反應。

（三） 推廣紙張上的數字辨識，如辨識信件上的郵遞區號，並可以用機械手臂進行分類。