



中華科技大學

電子工程研究所碩士班

碩士學位論文

基於 AdaBoost 演算法之人臉識別研究

**The study of face recognition based on AdaBoost
algorithm**

研究生：陳銘修

指導教授：陳梧桐 博士

中華民國 104 年 7 月



中華科技大學

電子工程研究所碩士班

碩士學位論文

基於 AdaBoost 演算法之人臉識別研究

**The study of face recognition based on AdaBoost
algorithm**

研究生：陳銘修

指導教授：陳梧桐 博士

中華民國 104 年 7 月

誌謝

時間過得很快，碩士的旅程結束了即將前往人生另一個旅程。在碩士的旅程裡，首先非常感謝指導教授陳梧桐教授及系上的楊耀波教授與周瑞雄教授在我碩士學程中孜孜不倦的教誨與指導，使得我在課業或是人生道路中獲益良多。其次感謝百忙之中蒞臨指導的口試委員劉慶華博士給予我撰寫的論文建議及修改，讓本論文更加完善。

在碩士的旅程中，不可或缺的還有同學即是夥伴：劉宸豪、曾少卜、王聯升、鍾駿綸同學，感謝你們在課業討論與問題提供幫助。在此也謝謝無線通訊組已畢業的學長姐與讓我拍照建立圖片資料庫的人們。

感謝的人實在太多，但最大感謝還是要給我的家人們，謝謝家人的支持與鼓勵，讓我在這兩年的旅程中沒有後顧之憂與其它問題能好好學習，謹以此論文獻給最敬愛家人。

謝謝所有關心我、鼓勵我的朋友，謝謝你們！

研究生：陳銘修

民國 104 年 7 月

Abstract

In positioning algorithm using the image feature to locate the image position, the pattern recognition algorithm proposed by Viola is one of the positioning algorithms with highest correct rates. Viola's positioning algorithm is the algorithm using the AdaBoost algorithm with the coefficients of Haar function expansion. Viola's positioning algorithm can be used in the case of real-time location positioning. Although Viola's positioning algorithm positioning speed is very quickly, but in the training stage it takes a very long time.

In order to improve the speed of training stage of the Viola's algorithms and can have the similar accuracy with the original Viola positioning algorithm, this study is firstly adjustment the training images to an appropriate size and convert these images to gray scale. Then, take Haar function expansions for these gray images, and select the appropriate functions to reduce the number of Haar function using in the training stage. Experimental results show that the propose method can reduce the time during the training stage.

Key words: AdaBoost, Haar function, Face recognition

摘要

在使用影像特徵比對的影像定位演算法中，以 Viola 所提到的圖形識別定位演算法是定位演算法中正確率最高的演算法之一。Viola 定位演算法是使用 AdaBoost 演算法及 Haar 函數展開式係數的演算法，其定位速度已可用在即時定位。雖然 Viola 定位演算法的定位速度很迅速，但是在訓練時卻需要耗費很長的時間。

為了改善和提高演算法訓練的速度且能有原 Viola 定位演算法相近的正確率，本研究將原為彩色照片的影像調整為適當大小並轉為灰階圖，經過調整以後的影像使用 Haar 函數展開式，選取適當的 Haar 函數以減少訓練時 Haar 函數的使用數量，再讓 AdaBoost 做訓練，如此一來 AdaBoost 演算法即可提高不少的訓練速度。

關鍵詞：AdaBoost 演算法、Haar 函數、人臉辨識

目 錄

Abstract	i
摘 要	ii
目 錄	iii
表目錄	v
圖目錄	vi
第一章 緒論	1
第一節 研究動機	1
第二節 文獻探討	2
第三節 論文架構	5
第二章 Haar 函數	6
第一節 Haar 函數轉換	6
壹、 Haar 函數說明	6
貳、 Haar 函數係數展開式	9
第三章 AdaBoost 演算法	10
第一節 AdaBoost 簡介	10
第二節 AdaBoost 演算法	11
第四章 實驗過程	13
第一節 圖片壓縮	13
第二節 影像灰階化	14
第三節 建立人臉樣本	15
第四節 積分影像	16
第五節 Haar 函數數量選擇	18
第六節 Haar 函數展開係數	19
第七節 AdaBoost 訓練分類器	20
壹、 建立弱分類器	20
貳、 設定權重	21

參、 計算錯誤率與更新所有樣本權重值.....	21
肆、 判斷分類器數量.....	21
伍、 強分類器.....	21
第五章 實驗結果.....	23
第一節 實驗結果.....	23
壹、 裁剪與裁剪拼圖方式.....	23
貳、 減少 Haar 函數展開式係數與選擇弱分類器數量 T	24
第六章 結論與未來研究方向.....	44
第一節 結論.....	44
第二節 未來研究方向.....	44
參考文獻.....	45
附 錄 圖片資料庫.....	47
作者簡介.....	53

表目錄

表 1.1 相關文獻.....	2
表 5.1 不裁剪影像與裁剪影像的訓練時間.....	23
表 5.2 裁剪拼圖的訓練時間.....	24
表 5.3 裁剪拼圖 3×3 人臉比對正確率	24
表 5.4 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，弱分類器數量=30 結果彙 整表.....	25
表 5.5 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，弱分類器數量=50 結果彙 整表.....	25
表 5.6 Haar 函數 0 到 4 階層，Haar 函數展開係數>20，弱分類器數量=30 結果彙 整表.....	35

圖目錄

圖 1.1 結合 Haar 函數與 AdaBoost 演算法之識別流程圖	1
圖 2.1 Haar 函數的矩形框特徵類別，由左到右分別為第零類，第一類，第二類及第三類.....	6
圖 2.2 每階層 Haar 函數不為零的區域的大小	7
圖 2.3(a) 第零階 (k, j) 皆為 $(0,0)$ 位置的 Haar 函數	8
圖 2.3(b) 第一階、第一類 $(k, j) = (0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數 ..	8
圖 2.3(c) 第一階、第二類 $(k, j) = (0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數 ...	8
圖 2.3(d) 第一階、第三類 $(k, j) = (0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數 ..	9
圖 3.1 弱分類器組合強分類器示意圖	10
圖 3.2 AdaBoost 流程圖	12
圖 4.1 圖片處理與訓練流程圖	13
圖 4.2 影像原圖	13
圖 4.3 等比例縮小圖	13
圖 4.4 影像原圖	14
圖 4.5 等比例縮小圖	14
圖 4.6 等比例縮小圖影像	15
圖 4.7 等比例縮小灰階化影像	15
圖 4.8 調整比例灰階化與人臉樣本影像	15
圖 4.9 影像積分之所有像素和	16
圖 4.10(a) 積分影像矩陣計算概念圖綠色範圍	17
圖 4.10(b) 積分影像矩陣計算概念圖藍色範圍	17
圖 4.10(c) 積分影像矩陣計算概念圖	17
圖 4.11 計算紅色區塊像素和面積圖	18
圖 4.12 使用 4095 個 Haar 函數之人臉圖	18
圖 4.13 使用 393 個 Haar 函數重組之人臉圖	19

圖 4.14 使用 876 個 Haar 函數重組之人臉圖	19
圖 4.15 Haar 函數展開係數計算加權方向示意圖	20
圖 4.16 AdaBoost 分類訓練與辨識流程圖	20
圖 4.17 強分類器搜尋人臉示意圖	22
圖 5.1 灰階裁剪圖之不同人影像九宮格拼圖與對應之人臉樣本	23
圖 5.2 人臉樣本與辨識比對結果	24
圖 5.3 使用 80 個 Haar 函數重組之人臉圖	25
圖 5.4 使用 124 個 Haar 函數重組之人臉圖	25
圖 5.5 使用 213 個 Haar 函數重組之人臉圖	26
圖 5.6 使用 236 個 Haar 函數重組之人臉圖	26
圖 5.7 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	27
圖 5.8 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	27
圖 5.9 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	28
圖 5.10 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	28
圖 5.11 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	29
圖 5.12 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	29
圖 5.13 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	30
圖 5.14 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，定位更加準確	30
圖 5.15 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	31
圖 5.16 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	31
圖 5.17 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	32
圖 5.18 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	32
圖 5.19 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	33
圖 5.20 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	33
圖 5.21 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	34
圖 5.22 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，偏移變準確	34
圖 5.23 使用 393 個 Haar 函數重組之人臉圖	35
圖 5.24 使用 615 個 Haar 函數重組之人臉圖	35

圖 5.25 使用 687 個 Haar 函數重組之人臉圖	35
圖 5.26 使用 876 個 Haar 函數重組之人臉圖	35
圖 5.27 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確	36
圖 5.28 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確	37
圖 5.29 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確	37
圖 5.30 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確	38
圖 5.31 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確	39
圖 5.32 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確	39
圖 5.33 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確	40
圖 5.34 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	40
圖 5.35 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	41
圖 5.36 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	41
圖 5.37 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	42
圖 5.38 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	42
圖 5.39 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確	43

第一章 緒論

跟隨著時代的變遷、科技的進步，有很多的科技產品以及便利性的服務都問世了，如：平板電腦、智慧型手機、網路繳款、...等，人們的生活都變得更加快樂又方便。而便利的生活使的人們必須保護自己隱私或個資，所以早期的保護機制是設定密碼來保護，但如密碼被盜取了，這樣會讓使用者造成損失，所以人們就開始使用了生物認證的技術，如：虹膜辨識、語音辨識、人臉辨識、...等來保護自己。

第一節 研究動機

在一套的辨識系統裡面，人臉識別可說是近年來比較熱門的研究項目。人臉識別系統能分為人臉偵測與人臉辨識，第一步是人臉的偵測，如果人臉偵測時偵測錯誤，那麼第二步的人臉識別的錯誤率即會變高。本文使用的方法為 Viola 所提 AdaBoost 演算法[1][2]，是定位演算法正確率最高的演算法之一，它的定位速度以可用在即時定位，雖然 AdaBoost 演算法定位速度很快速，但在樣本訓練的時間太過冗長。

本文所提方法為了提高 AdaBoost 演算法訓練時速度，主要使用 Haar 函數將參數點減少，在訓練 AdaBoost 時可以減少弱分類器的數量，加快訓練速度。圖 1.1 為結合 Haar 函數與 AdaBoost 演算法之識別流程。並以實驗結果來證明本文所提方法能有很高的辨識正確率。

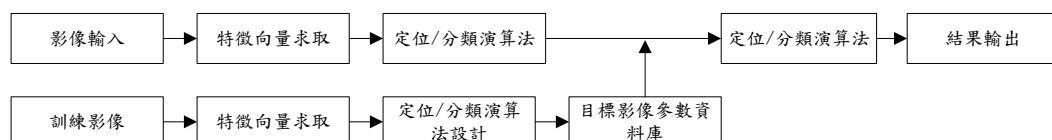


圖 1.1 結合 Haar 函數與 AdaBoost 演算法之識別流程圖

第二節 文獻探討

關於 Haar 函數、AdaBoost 演算法與人臉辨識方法的相關文獻

表 1.1 相關文獻

2010	中華科技大學-陳柏霖[6]	以 Adaboost 演算法之車牌定位研究與實作
概要： 提出結合Haar函數及Adaboost演算法的刻練過程，使演算法訓練速度提升減少時間並能與原演算法有相似的定位正確率。		
2010	中華科技大學-辛俊昇[7]	於 HAAR 函數轉換空間之臉部識別演算法
概要： 使用Haar函數做線性組合，並用較少特徵參數進行識別比對，證明減少特徵參數也能有不錯的識別率。		
2011	明志科技大學-陳致豪[8]	以膚色偵測加速 AdaBoost 人臉偵測
概要： 提出結合Adaboost演算法並用膚色偵測來減少檢索區塊，達到縮短偵測時間及減少錯誤偵測。		
2010	國立屏東教育大學-詹忻澂[9]	混合 HAAR-LIKE 特徵與特徵相對位置之人臉偵測
概要： 以Haar-like特徵為基礎，將人臉特徵位置加入Haar-like特徵相對位置，達到簡化偵測以提高偵測速度。		
2012	銘傳大學-賴昆煒[10]	基於Radon轉換之人臉辨識系統

<p>概要：</p> <p>使用OpenCV來定位出人臉位置及臉部大小，再透過Radon Transfofm判斷人臉是否歪斜來達到提高人臉歪斜的辨識正確率。</p>		
2009	國立臺灣科技大學-吳旻峰[11]	基於像素導向之階層式特徵與統計式遮罩Adaboost人臉偵測
<p>概要：</p> <p>提出迴旋式遮罩與階層式特徵配合 Haar-like 和 Adaboost 演算法達到減少特徵點達到加速 Adaboost 演算法訓練速度。</p>		
2008	亞洲大學-翁偉鐘[12]	利用 Adaboost 演算法於數位影像中進行文字定位
<p>概要：</p> <p>以 Adaboost 演算法為基礎建立擷取文字的系統，使用連通原件標記影像中的文字做定位，接著用離散小波與紋理特徵去進行文字分類判斷。</p>		
2013	逢甲大學-楊攸淇[13]	基於 Adaboost 演算法使用多重感興趣區域以提升偵測距離之多重物件偵測
<p>概要：</p> <p>以 Adaboost 演算法訓練行人偵測的特徵，再使用多重 ROI 演算法與多重物件偵測，偵測距離得遠近提高物件正確率。</p>		
2012	義守大學-莊啟鴻[14]	使用五官特徵之人臉辨識
<p>概要：</p> <p>提出一套新的人臉五官特徵且可調整特徵數量，組合多組人臉特徵模版進行人臉辨識。</p>		
2012	大同大學-黃韋瀚[15]	結合五官特徵與深度資訊之人臉辨識系統

<p>概要：</p> <p>使用Adaboost及ASM偵測人臉特徵，配合使用三維人臉的資訊進行人臉偵測與辨識。</p>		
2011	國立台灣師範大學-李昇龍[16]	基於增量學習之人臉辨識研究
<p>概要：</p> <p>提出增量學習的人臉辨識方法，使用影片辨識人臉，長期觀察人臉些微的變化並新增人臉變化的特徵來辨識人臉。</p>		
2014	中華大學-陳宣有[17]	結合特徵點與雙向辨識演算法的人臉辨識技術
<p>概要：</p> <p>提出使用兩階段的人臉辨識技術，第一階段是人臉區塊位置給予加權的WLVP模組，第二是階段是以特徵點為基礎的雙向辨識演算法，使用特徵點匹配與點之間的幾何結構進行最後辨識。</p>		
2012	明新科技大學-王士維[18]	複雜背景下之人臉偵測系統之設計
<p>概要：</p> <p>利用色彩空間使得膚色分割，再用二值化的特性膨脹與侵蝕與人臉特徵濾除背景近於膚色和人臉形狀之物件後進行人臉偵測。</p>		
2013	義守大學-陳柏州[19]	臉部特徵抽取之人臉辨識
<p>概要：</p> <p>使用人臉五官辨識方法當基礎，加入眉毛為新特徵使人臉的辨識率、正確率得以提升。</p>		

第三節 論文架構

本文共分為六章，各章內容概述如下：

第一章 緒論：

說明本文研究動機、文獻探討及論文架構。

第二章 Haar函數：

介紹 Haar 函數 4 個種類與 Haar 函數係數展開式。

第三章 Adaboost演算法：

介紹 Adaboost 演算法之分類訓練與架構流程。

第四章 實驗過程：

介紹實驗過程流程圖、圖片識別前置處理、圖片灰階化、影像積分與建立人臉識別樣本。

第五章 實驗結果：

說明硬體設備及開發工具、訓練樣本時間、正確率與不同參數條件下反覆測試結果。

第六章 結論與未來研究方向：

說明結論及未來研究方向。

第二章 Haar 函數

本章介紹 Haar 函數 4 個種類與 Haar 函數係數展開式。

第一節 Haar 函數轉換

壹、 Haar 函數說明

Haar 函數可以分為四類，如圖 2.1 所示，由左到右分別為第零類，第一類，第二類及第三類。

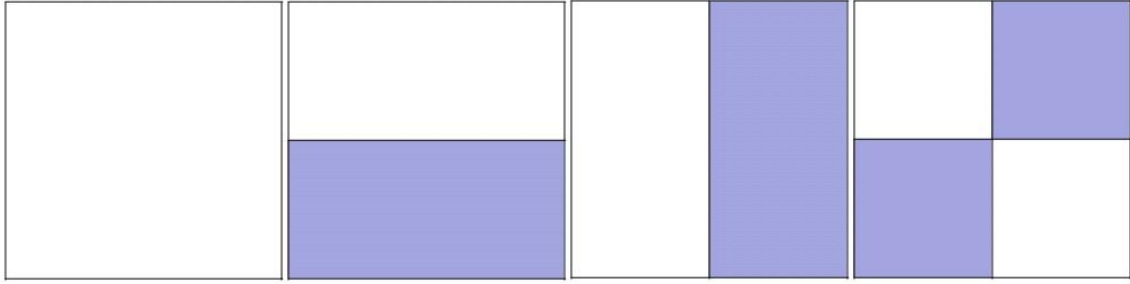


圖 2.1 Haar 函數的矩形框特徵類別，由左到右分別為第零類，第一類，第二類及第三類

本研究設定人臉大小為 $N \times N = 2^6 \times 2^6 = 64 \times 64$ 的大小，使用的 Haar 函數 $H_{n(k,j)}^{N,L}(y,x)$ 分為四個種類，函數中之 $n=0,1,2,3$ 。

第零類， $n=0$ ：

$$H_{0(k,j)}^{N,L}(y,x) = 1$$

第一類， $n=1$ ：

$$H_{1(k,j)}^{N,L}(y,x) = \begin{cases} 1, k \frac{2^N}{2^{L-1}} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ -1, k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ 0, otherwise \end{cases}$$

第二類， $n=2$ ：

$$H_{2(k,j)}^{N,L}(y,x) = \begin{cases} 1, k \frac{2^N}{2^{L-1}} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ -1, k \frac{2^N}{2^{L-1}} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ 0, otherwise \end{cases}$$

第三類， $n=3$ ：

$$H_{3(k,j)}^{N,L}(y,x) = \begin{cases} 1, k \frac{2^N}{2^{L-1}} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ 1, k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}} \\ -1, k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} \\ -1, k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} + 1 \leq y \leq k \frac{2^N}{2^{L-1}} + \frac{2^N}{2^{L-1}}, j \frac{2^N}{2^{L-1}} + 1 \leq x \leq j \frac{2^N}{2^{L-1}} + \frac{2^N}{2^L} \\ 0, otherwise \end{cases}$$

其 L 為每個不同的階層，如圖 2.2 所示，不同階層 Haar 函數的量也會隨 L 越來越大而越多。

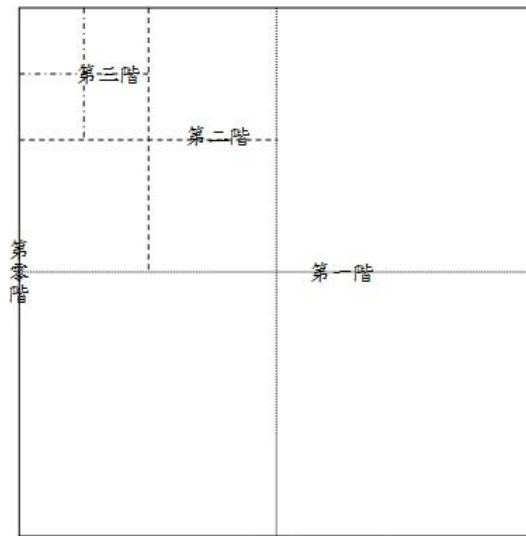


圖 2.2 每階層 Haar 函數不為零的區域的大小

而 (k, j) 為對應 Haar 所在的位置， (k, j) 的範圍為 $0 \leq j \leq 2^L - 1, 0 \leq k \leq 2^L - 1$ 。
 如圖 2.3 所示(a)為第零階 Haar 函數，由左至右為第一類、第二類及第三類 (k, j) 皆為 $(0,0)$ 之位置，而(b)(c)(d)為第一階，第一類、第二類及第三類之 Haar 函數，由左至右為 $(k, j)=(0, 0), (0, 1), (1, 0), (1, 1)$ 之位置。

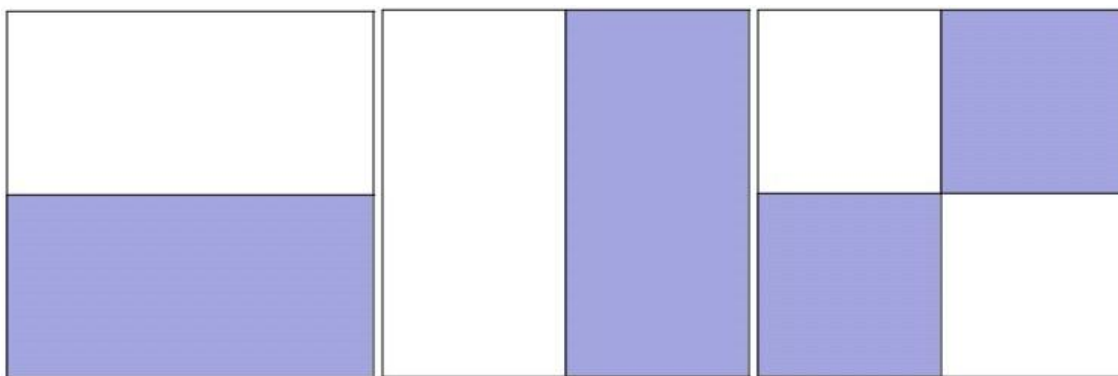


圖 2.3(a) 第零階 (k, j) 皆為 $(0,0)$ 位置的 Haar 函數

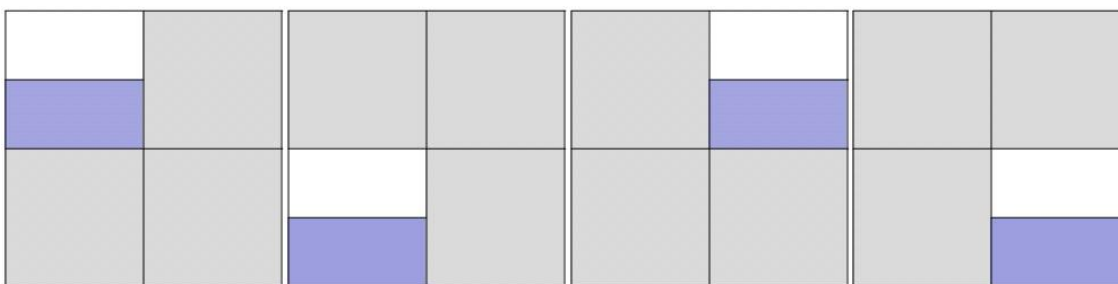


圖 2.3(b) 第一階、第一類 $(k, j)=(0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數

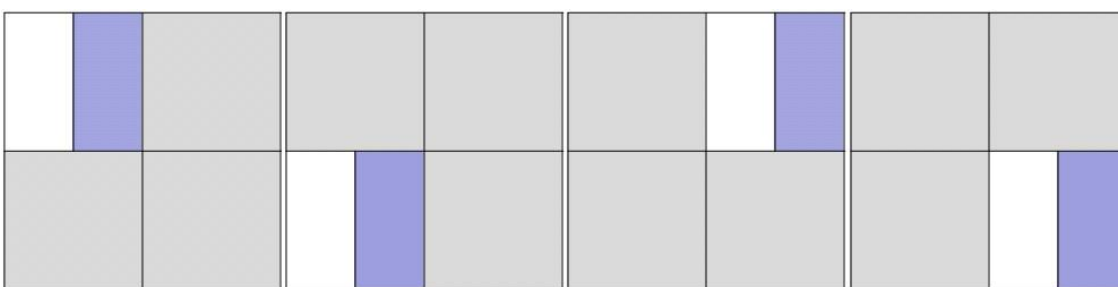


圖 2.3(c) 第一階、第二類 $(k, j)=(0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數

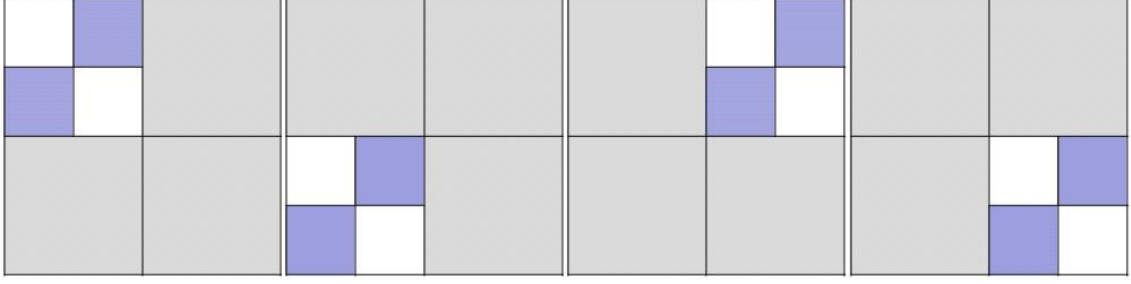


圖 2.3(d) 第一階、第三類 $(k, j) = (0, 0), (0, 1), (1, 0), (1, 1)$ 位置的 Haar 函數

貳、 Haar 函數係數展開式

$$H(y, x) = \alpha_{0000} H_{0(0,0)}^{N,0}(y, x) + \sum_{L=1}^N \sum_{k=0}^{2^L-1} \sum_{j=1}^{2^L-1} \sum_{n=1}^3 \alpha_{Lkn} H_{n(k,j)}^{NL}(y, x)$$

其 $\alpha_{L'k'j'n'}$ 為 $H(y, x)$ 的展開式，加權係數為：

$$\alpha_{L'k'j'n'} = \frac{H \circ H_{n'(k',j')}^{N,L'}}{H_{n'(k',j')}^{N,L'} \circ H_{n'(k',j')}^{N,L'}} = \frac{\sum_{y=0}^{(2^N-1)} \sum_{x=0}^{(2^N-1)} H(y, x) H_{n'(k',j')}^{N,L'}(y, x)}{\sum_{y=0}^{(2^N-1)} \sum_{x=0}^{(2^N-1)} (H_{n'(k',j')}^{N,L'}(y, x))^2}$$

第三章 AdaBoost 演算法

本章介紹 Adaboost 演算法與其架構流程。

第一節 AdaBoost 簡介

AdaBoost(Adaptive Boosting)是一種疊代算法，其概念是保存一訓練集與訓練集中元素的加權機率用以訓練弱分類器，在每一次的疊代中 AdaBoost 會調整弱分類器的機率分佈並計算出其錯誤率。因每次的疊代資料中難以分類正確的樣本也能得到較大的機率加權，而當這些弱分類器的數量夠多能夠組成高正確率的分類器時，AdaBoost 演算法的結果即可使用多個弱分類器結合成一個強分類器，如圖 3.1 所示。

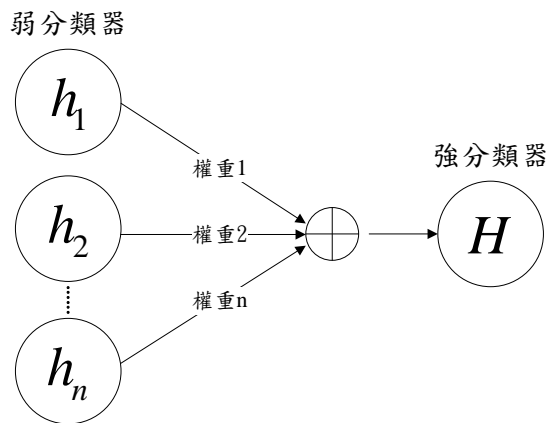


圖 3.1 弱分類器組合強分類器示意圖

Boosting 算法將前一個弱分類器分錯的樣本會被用來訓練下一個弱分類器，在每一輪中加入一個新的弱分類器直到預定的錯誤率。Boosting 算法無法將訓練出來的弱分類器組合成一個強分類器而 AdaBoost 每個訓練的樣本初始給予的權重都一樣，若某個樣本點已被某個弱分類器準確的分類，那麼在下一個弱分類器被選中的機率就會降低，相反如果某樣本點沒有被準確的分類，那麼樣本點的權重就會提高。最後將每次訓練的弱分類器結合起來，作為最終分類器（強分類器）。

第二節 AdaBoost 演算法

AdaBoost 演算法流程如下：

弱分類器： $h_j(x)$

$$h_j(x) = \begin{cases} 1, & p_j f_j < p_j \theta_j \\ 0, & \text{otherwise} \end{cases}, \text{ 其中 } p_j \text{ 為不等式的位置, } f_j \text{ 為特徵向量的位置, } \theta_j \text{ 為閾}$$

值是訓練最小錯誤率的值。

給定樣本 $(x_1, y_1, w_{1,1}), (x_2, y_2, w_{1,2}), \dots, (x_N, y_N, w_{1,N})$ ，其中 y_i 用來區分正負樣本，當 $y_i=1$ 時為正樣本， $y_i=0$ 時為負樣本， $w_{t,i}$ 為權重。

設 T 為組成強分類器的弱分類器總數量 t 為弱分類器初始為 1。

1. 設定初始權重值為 $w_{1,i} = \begin{cases} \frac{1}{2m}, & y_i = 1 \\ \frac{1}{2l}, & y_i = 0 \end{cases}$ ，其中 m 為正樣本總數， l 為負樣本總數。
2. 權重值正規化： $w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^N w_{t,j}}$
3. 對每個弱分類器 $h_j(x)$ 計算其錯誤率： $E_j = \sum_i w_{t,i} |h_j(x_i) - y_i|$
4. 在所有弱分類器裡選擇具有最小錯誤率 E_t 的一個分類器 h_t
5. 更新所有樣本的權重值： $w_{t+1,i} = w_{t,i} \beta_t^{1-\varepsilon_i}$ 其中 $\beta_t = \frac{E_t}{1-E_t}$ 當 $\varepsilon_i = 0$ 時，表示 x_i 被分類正確，否則 $\varepsilon_i = 1$
6. 若 $t < T$ ，則 $t = t + 1$ ，回到 2. 繼續訓練弱分類器，否則進入 7. 完成強分類器
7. 訓練完成的強分類器： $h(x_i) = \begin{cases} 1, & \sum_{t=1}^T \alpha_t h_t(x_i) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, & \text{otherwise} \end{cases}$ ，其中 $\alpha_t = \log \frac{1}{\beta_t}$

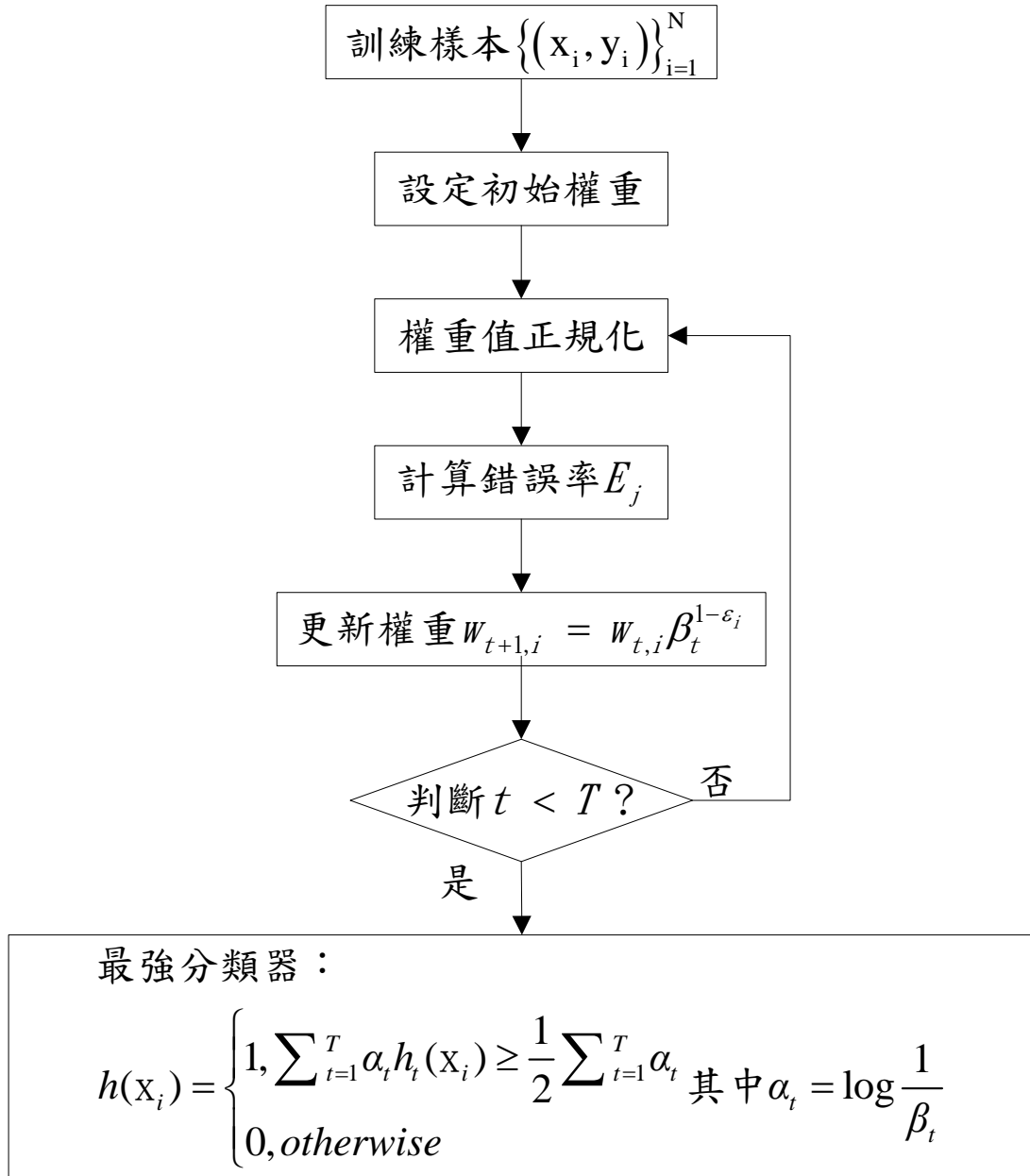


圖 3.2 AdaBoost 流程圖

第四章 實驗過程

AdaBoost 訓練分類器時，需要參數進行弱分類器分類，故先將圖片做前置處理並產生資料庫以便於後續訓練。圖 4.1 為前置處理與訓練流程。

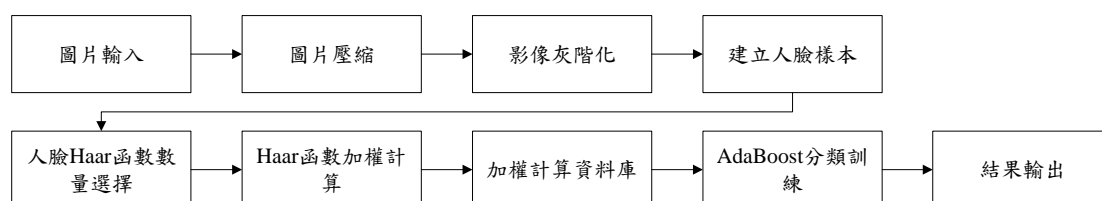


圖 4.1 圖片處理與訓練流程圖

第一節 圖片壓縮

每張照片因不同的拍攝器材、拍攝的遠近，會影響到每張照片的人臉大小與位置。故將每張照片做等比例縮小至每張人臉大小一致，人臉大小範圍為 64 像素點×64 像素點，在下列兩張圖中圖 4.3 是將圖 4.2 做 0.095 倍率等比例縮小，圖 4.3 中很明顯的背景有像素點改變造成影像的變化。



圖 4.2 影像原圖



圖 4.3 等比例縮小圖

在下列兩張圖中，圖 4.5 是將圖 4.4 做 0.265 倍率等比例縮小。



圖 4.4 影像原圖



圖 4.5 等比例縮小圖

第二節 影像灰階化

使用 RGB 訊號處理影像時，可能會隨著人臉附近的飾品、背景環境過於複雜或相近等而造成誤判的情況，故將彩色圖片灰階化且能加快電腦運算速度。在一般彩色影像中，影像由三維的紅色(R)、綠色(G)、藍色(B)三原色光所混合而成，而三原色光中每種顏色都有 0~255 之色彩變化亦為色彩空間。將三維 RGB 的數值都調為 0 的顏色為黑色、255 為白色，不同的 RGB 像素值乘上不同的加權係數使三維的 RGB 色彩空間變為一維的直線，其轉換公式(4.1)。直線兩端分別為白色與黑色，灰階值在直線上做處理，靠近白色那端圖片就越亮、黑色那端就越暗，使得影像沒有顏色之分只有明暗的變化。

$$Y=0.2989R + 0.5870G + 0.1140B \quad (4.1)$$

式(4.1)中 Y 為灰階亮度的變化。

將圖 4.6 彩色影像轉換為圖 4.7 灰階影像。



圖 4.6 等比例縮小圖影像



圖 4.7 等比例縮小灰階化影像

第三節 建立人臉樣本

將不同大小的照片人臉比例調成一致如圖 4.8 中左圖，且灰階化後如中圖，再剪裁成人臉範圍如右圖。正樣本範圍為初始樣本及初始樣本之上、下、左、右偏移 1 點像素點，共 3×3 點，而上、下、左、右偏移 8 點共 15×15 像素點捨棄不加入訓練，其餘皆為負樣本。有了正負樣本範圍即可進行 AdaBoost 的樣本訓練。



圖 4.8 調整比例灰階化與人臉樣本影像

第四節 積分影像

使用 Haar 函數時，因參數矩陣數值太過繁多若要計算其各個矩陣之像素和將會相當耗時。積分影像可快速求取任何大小的矩陣特徵計算，計算出矩陣之像素和，即可加快影像處理速度。

積分影像或稱總和面積表可使用快速查表的方式計算出影像中任意垂直之矩形區塊的像素和。如圖 4.9 所示。其為從點(0,0)到(x,y)之所有像素和，以數學式表示為(4.2)。

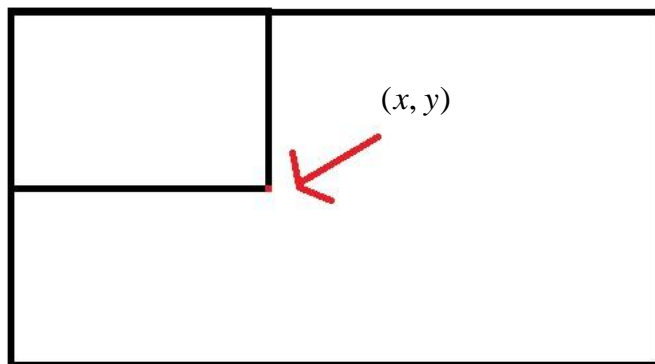


圖 4.9 影像積分之所有像素和

$$Sum(x, y) = \sum_{x=0}^x \sum_{y=0}^y image(x, y) \quad (4.2)$$

積分影像的計算如圖 4.10。假設若要計算出紅色區塊之像素和，可用咖啡色區塊減掉綠色區塊再減掉藍色區塊，因藍色區塊與綠色區塊有重疊，重疊大小為黃色區塊故最後加回黃色區塊。如圖 4.11 所示。

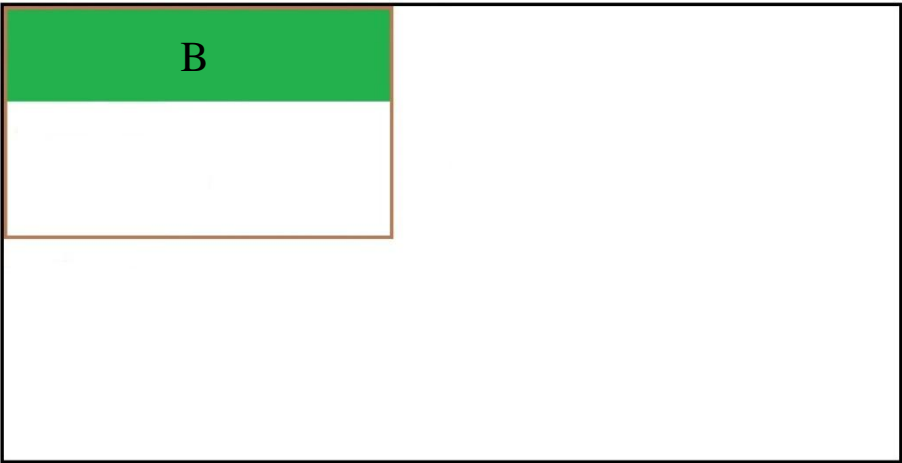


圖 4.10(a) 積分影像矩陣計算概念圖綠色範圍

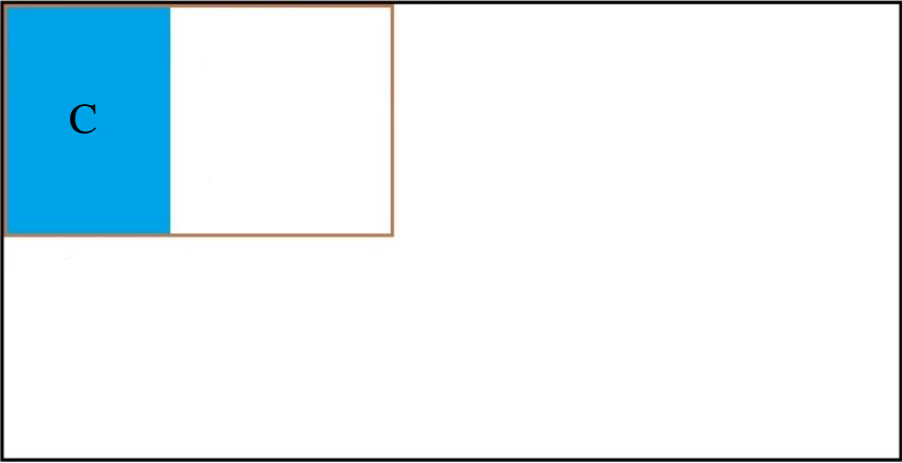


圖 4.10(b) 積分影像矩陣計算概念圖藍色範圍

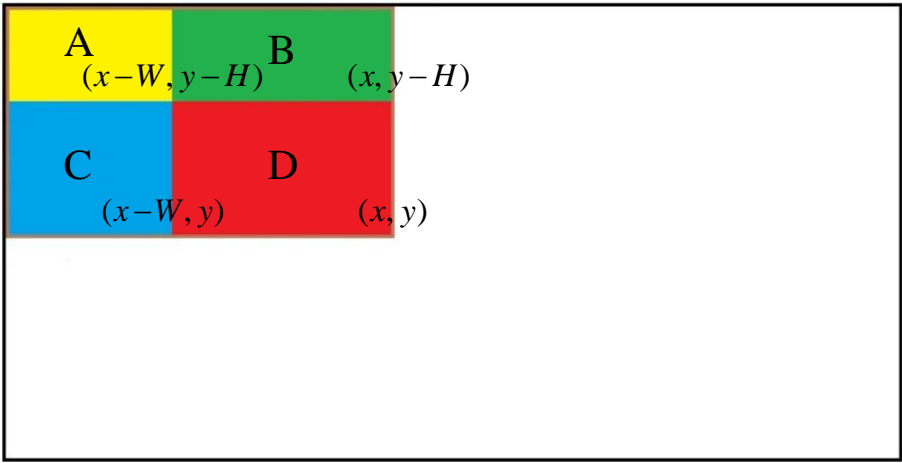


圖 4.10(c) 積分影像矩陣計算概念圖

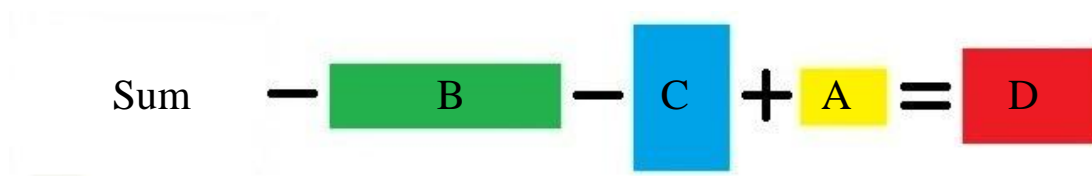


圖 4.11 計算紅色區塊像素和面積圖

從計算中可得知只需知道紅色區塊的四個頂點座標即可求出想找的任意區塊面積之像素和，即 (4.3) 公式。

$$D(x, y) = Sum(x, y) - B(x, y - H) - C(x - W, y) + A(x - W, y - H) \quad (4.3)$$

第五節 Haar 函數數量選擇

由於使用全部 Haar 函數會使訓練過久，故藉由基準值 Gamma 與 Haar 函數之階層的設定，在所有的 Haar 函數係數中篩選出滿足大於基準值與階層的 Haar 函數係數作為基本 Haar 函數樣本數量，爾後有新的且樣本沒有的 Haar 函數係數即加入到基本 Haar 函數樣本數量，可使得每張圖片都有能辨識之 Haar 函數係數。

Haar 函數階層越多其 Haar 函數數量則越多。圖 4.12 為全部 Haar 函數數量；圖 4.13 選擇 Haar 函數第零階到第 4 階 Haar 函數展開係數 > 20 的 Haar 函數數量；圖 4.14 為圖 4.13 加入新的 Haar 函數數量。

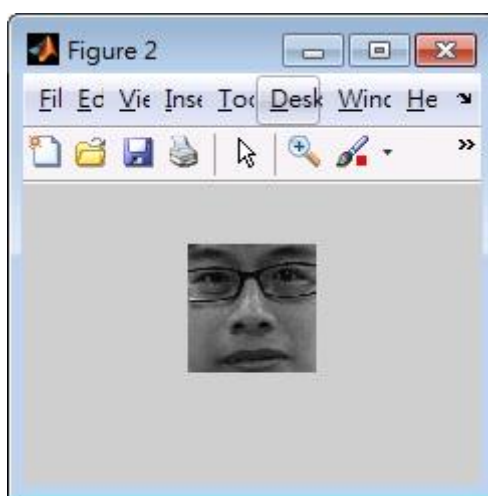


圖 4.12 使用 4095 個 Haar 函數之人臉圖

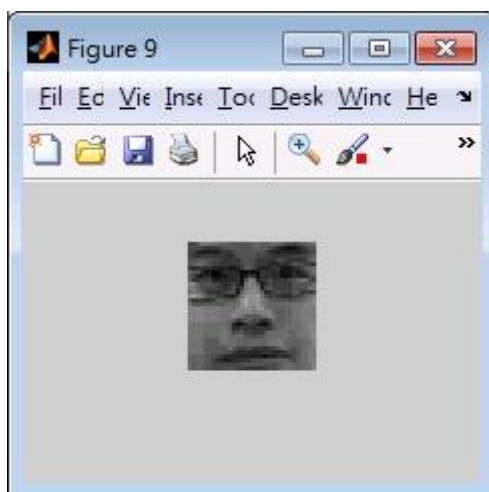


圖 4.13 使用 393 個 Haar 函數重組之人臉圖

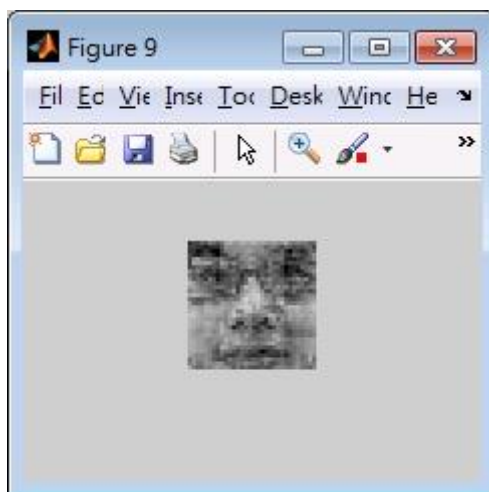


圖 4.14 使用 876 個 Haar 函數重組之人臉圖

選取 Haar 函數階層為 0 到 4 階與 Haar 函數展開係數 >20 的原因主要是因為其餘 Haar 函數展開係數解析度太過於細緻，對於人臉識別來說相當於是雜訊的存在，故選擇比較大的 Haar 函數展開係數作為樣本數量。

第六節 Haar 函數展開係數

設定好人臉大小為 64×64 後，使用 Haar 函數展開式將這個大小的方塊由左至右由上至下的計算出每一點加權值，如圖 4.15。計算出的加權數值即為 AdaBoost 演算法之 f_j 。

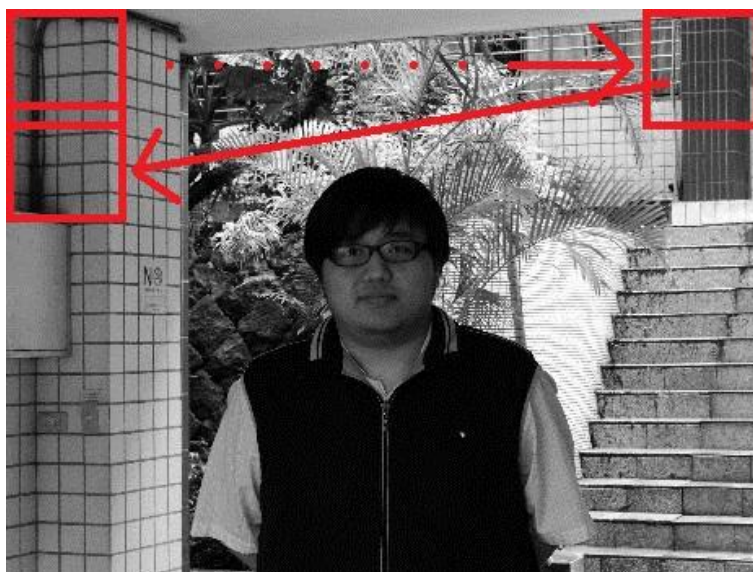


圖 4.15 Haar 函數展開係數計算加權方向示意圖

第七節 AdaBoost 訓練分類器

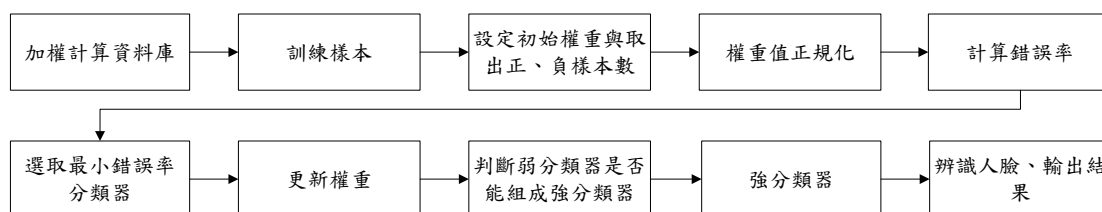


圖 4.16 AdaBoost 分類訓練與辨識流程圖

壹、建立弱分類器

用訓練樣本即 f_j 建立弱分類器， $h_j(x) = \begin{cases} 1, & p_j f_j < p_j \theta_j \\ 0, & \text{otherwise} \end{cases}$ ， θ_j 為正樣本 f_j 總合

平均加上負樣本 f_j 總合平均除以 2，由於不清楚正負樣本方向故用 p_j 來控制。

當正樣本 f_j 總合平均小於負樣本 f_j 總合平均時 p_j 大於 0 為正確而當正樣本 f_j 總合平均大於負樣本 f_j 總合平均時 p_j 小於 0 才為正確。

貳、設定權重

每個訓練樣本皆有初始權重，初始權重值為 $w_{1,i} = \begin{cases} \frac{1}{2m}, y_i = 1 \\ \frac{1}{2l}, y_i = 0 \end{cases}$ ，其中 m 為正

樣本總數， l 為負樣本總數。將每個初始權重做權重值正規化 $w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^N w_{t,j}}$ 。

參、計算錯誤率與更新所有樣本權重值

對每個弱分類器計算其錯誤率 $E_j = \sum_i w_{t,i} |h_j(x_i) - y_i|$ ，並在所有弱分類器裡選擇具有最小錯誤率 E_t 的分類器 h_t 。選取具有最小錯誤率弱分類器後，其餘弱分類器會更新權重值 $w_{t+1,i} = w_{t,i} \beta_i^{1-\varepsilon_i}$ 其中 $\beta_i = \frac{E_t}{1-E_t}$ 當 $\varepsilon_i = 0$ 時，表示 x_i 被分類正確，否則 $\varepsilon_i = 1$ ，然後繼續選取下一個具有最小錯誤率的弱分類器。

肆、判斷分類器數量

當弱分類器 t 不夠組成強分類器 T 時，即重複貳至參的步驟繼續訓練弱分類器，若分類器數量夠組成強分類器即到下一個步驟。

伍、強分類器

訓練完成的強分類器 $h(x_i) = \begin{cases} 1, \sum_{t=1}^T \alpha_t h_t(x_i) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0, otherwise \end{cases}$ ，其中 $\alpha_t = \log \frac{1}{\beta_t}$ ，

有了強分類器後，即可使用強分類器對圖片進行人臉辨識。

使用強分類器進行人臉識別的方向，如圖 4.17 所示。



圖 4.17 強分類器搜尋人臉示意圖

第五章 實驗結果

本文使用 CPU 為 Intel(R) Core(TM) i7 2600 3.4GHz 記憶體為 16G 的桌上型電腦進行 Haar 函數加權計算與 AdaBoost 分類器運算進行之硬體設備。

第一節 實驗結果

為了加速 AdaBoost 演算法訓練速度，本文研究了多種方法，其中有裁剪圖片做拼圖方式訓練、減少 Haar 函數展開式係數與選擇組合成強分類器的弱分類器數量，用少數圖片當訓練張數辨識全部圖片。

壹、 裁剪與裁剪拼圖方式

裁剪的人臉大小為 32×32 ，因為裁剪的圖為人臉範圍附近的上、左 50 像素點、下、右 80 像素點，若使用 64×64 的人臉大小則拼起來的圖會過大，這樣反而不會加快訓練速度。裁剪拼圖如圖 5.1。

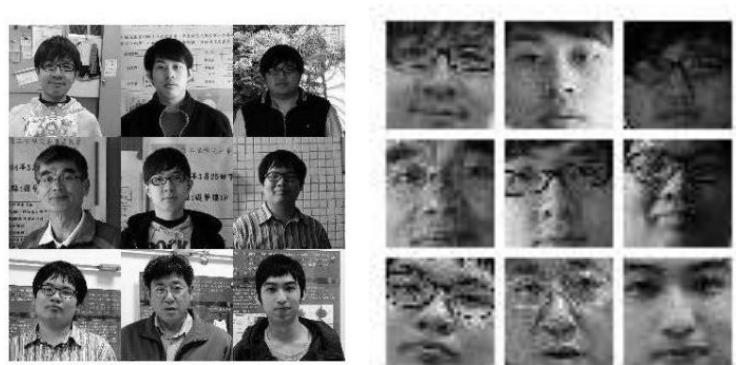


圖 5.1 灰階裁剪圖之不同人影像九宮格拼圖與對應之人臉樣本

表 5.1 不裁剪影像與裁剪影像的訓練時間

不裁剪與裁剪影像	訓練張數	訓練時間
不裁剪影像	1	約 13.5 天
不裁剪影像	9	預估 123 天
裁剪影像	1	約 2.5 天
裁剪影像	9	約 22.8 天

表 5.2 裁剪拼圖的訓練時間

訓練張數(拼圖張數)	訓練時間
1(9)	約 2.4 天
1(16)	約 7.8 天
1(25)	約 17.9 天

由表 5.1 與 5.2 可得知拼圖方式確實可以減少 AdaBoost 演算法訓練樣本的時間。其原因為將多張影像拼貼成一張大的影像，在訓練時可以減少對硬碟的存取，加快訓練速度。比原先一張一張圖處理需要較多次由硬碟存取各圖資料的方式快速。

表 5.3 裁剪拼圖 3×3 人臉比對正確率

弱分類器數量	訓練時間	正確率
150	約 16 小時	0%
300	約 1.2 天	0%
1023 (全部)	約 2.4 天	67%

在圖 5.2 中左為人臉樣本右為辨識結果，辨識正確率為 67%，錯誤辨識為(3)、(4)與(6)。

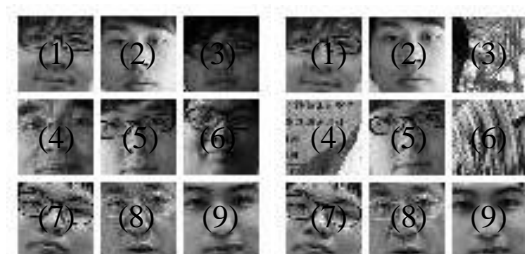


圖 5.2 人臉樣本與辨識比對結果

雖然訓練速度加快不少，但是由於裁剪了圖片使得圖片能訓練的負樣本數量大幅減少，而因負樣本的減少所以弱分類器無法充分的訓練負樣本，故在人臉識別的正確率上會非常的低。

貳、減少 Haar 函數展開式係數與選擇弱分類器數量 T

因全部 Haar 函數展開式係數放入訓練時間太過冗長，並不是本文的目的。故將所有 Haar 函數展開式係數中篩選出少許的 Haar 函數展開式係數，再將這少許 Haar 函數展開式係數中選擇分類器數量，實驗是否可以正確識別人臉，以提高 AdaBoost 演算法訓練速度。

實驗比較：

表 5.4 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，弱分類器數量=30 結果彙整表

訓練張數	2	5	10
Haar 函數展開係數數量	124	213	236
特徵求取時間	43.4 秒	195.6 秒	438.8 秒
弱分類器 T 數量	30	30	30
訓練分類器時間	3756 秒 (1 小時)	27706 秒 (7.7 小時)	61367 秒 (17 小時)
150 張人臉辨識率 正確/總數	42/150	22/150	44/150

表 5.5 Haar 函數 0 到 3 階層，Haar 函數展開係數>30，弱分類器數量=50 結果彙整表

訓練張數	2	5	10
Haar 函數展開係數數量	124	213	236
特徵求取時間	43.4 秒	195.6 秒	438.8 秒
弱分類器 T 數量	50	50	50
訓練分類器時間	6680 秒 (1.8 小時)	49343 秒 (13.7 小時)	116557 秒 (1.3 天)
150 張人臉辨識率 正確/總數	62/150	25/150	56/150

由表 5.4、5.5 得知 Haar 函數 0 到 3 階層 Haar 函數展開係數>30 的數量訓練張數少時，其 Haar 函數展開係數數量加入樣本的數量相對的少，圖 5.3 為使用 80 個 Haar 函數樣本數量。圖 5.4 為加入 1 張圖片新 Haar 函數數量。

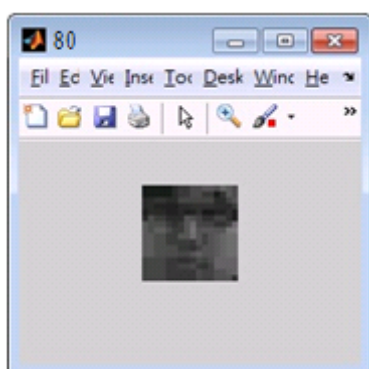


圖 5.3 使用 80 個 Haar 函數重組之人臉圖

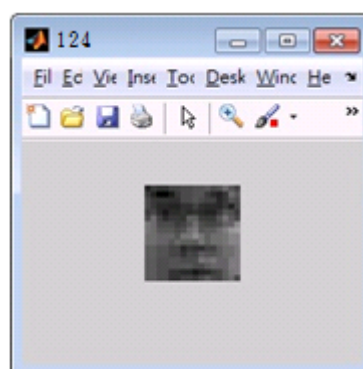


圖 5.4 使用 124 個 Haar 函數重組之人臉圖

圖 5.5 為加入 5 張圖片新 Haar 函數數量。圖 5.6 為加入 10 張圖片新 Haar 函

數數量。從圖 5.4 來看，圖片還能看出人臉輪廓，故計算 Haar 函數的加權時，還是能辨識出相似圖形。而因 Haar 函數數量較少，其訓練的分類器較為粗糙，能辨識人臉的程度還是有所限制。

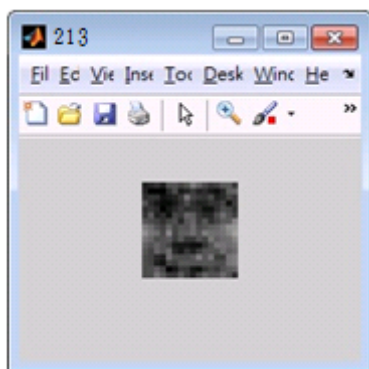


圖 5.5 使用 213 個 Haar 函數重組之人臉圖

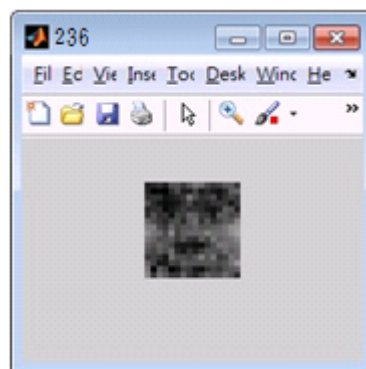


圖 5.6 使用 236 個 Haar 函數重組之人臉圖

從圖 5.5、5.6 來看，已經分辨不出人臉輪廓。但加入了新的 Haar 函數可以計算圖片 Haar 函數加權值，使得分類器變得細緻讓本來能辨識的人臉位置更為準確，有些偏移的人臉能定位準確，而因為分類器變為比較細緻也會使得有些原本能辨識的人臉會產生偏移。

Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，訓練張數分別為 1、10 張，雖然 Haar 函數與分類器數量減少，但訓練張數較多的結果還是能將訓練張數較少且有辨識到人臉的影像定位更加準確，如圖 5.7 至 5.14。



圖 5.7 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確



圖 5.8 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確



圖 5.9 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

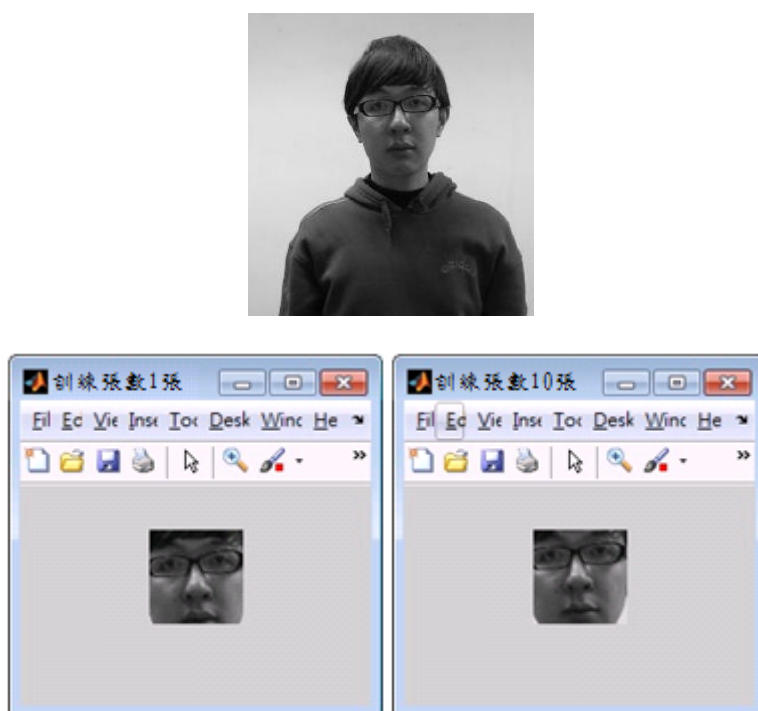


圖 5.10 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

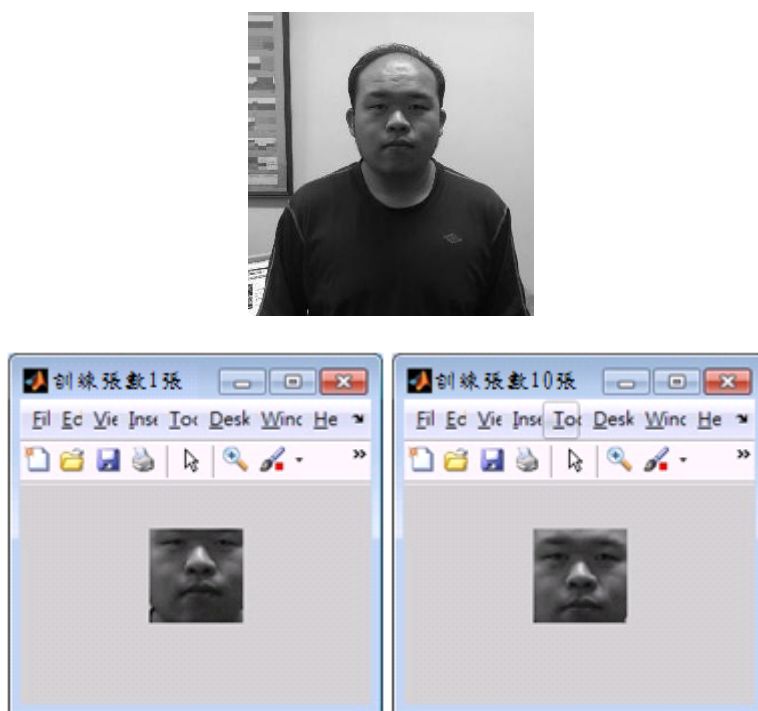


圖 5.11 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

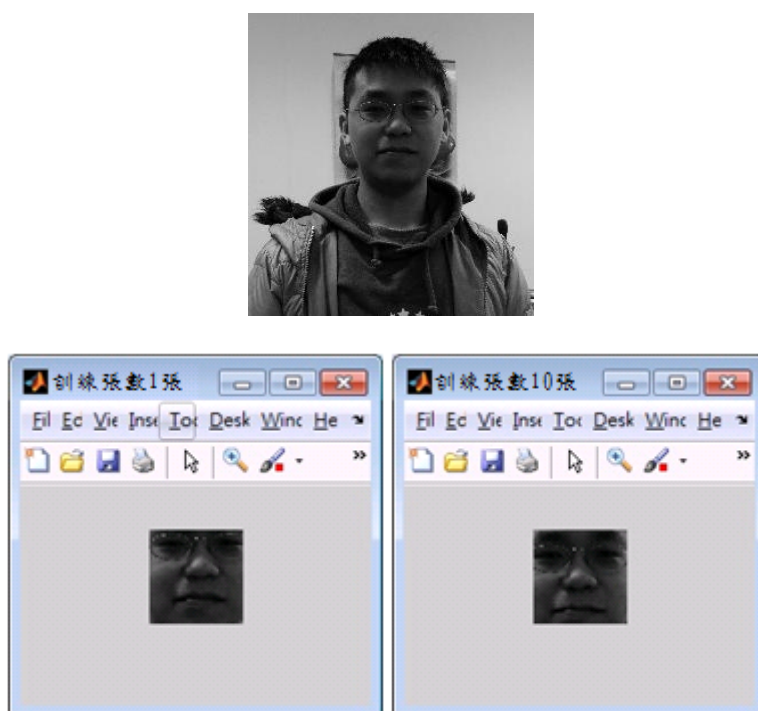


圖 5.12 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

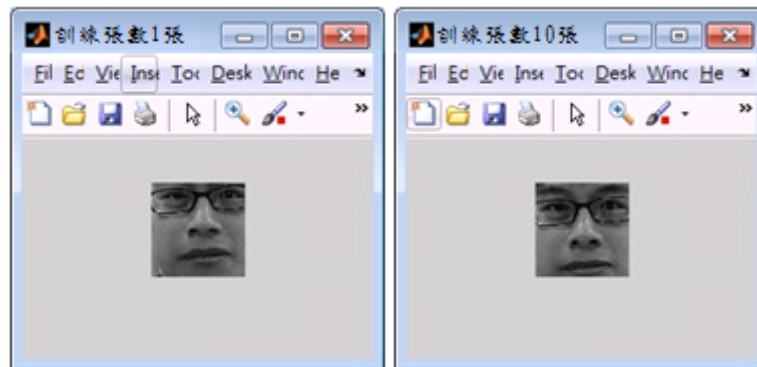


圖 5.13 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

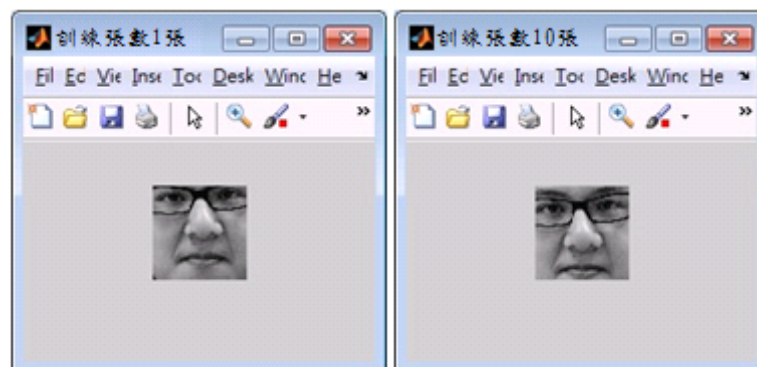


圖 5.14 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，定位更加準確

Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，訓練張數分別為 5、10 張，由表 5.5、5.6 可得知 Haar 函數數量增加而分類器數量不變時，能使訓練結果有些偏移的影像變為準確，如圖 5.15 至 5.22。

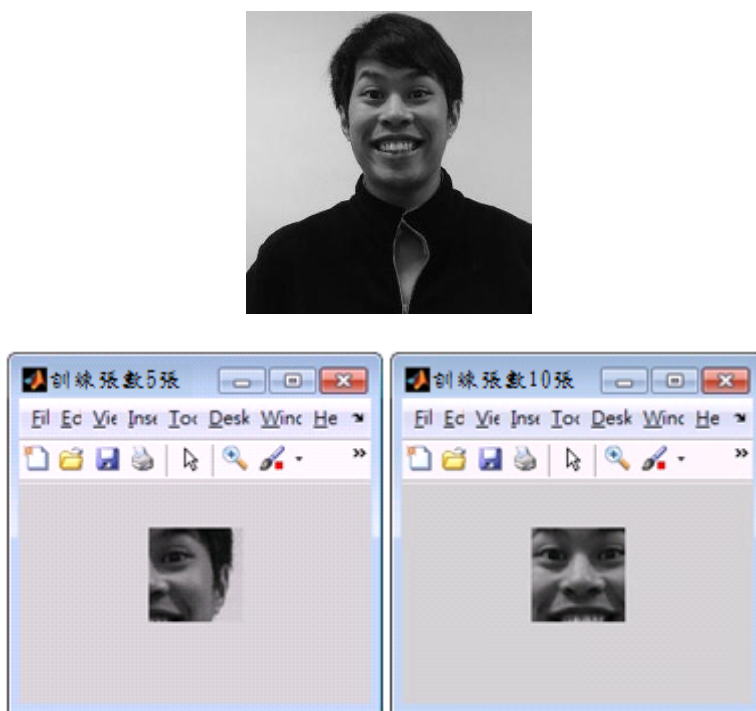


圖 5.15 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確



圖 5.16 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確



圖 5.17 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確



圖 5.18 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確



圖 5.19 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確



圖 5.20 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確

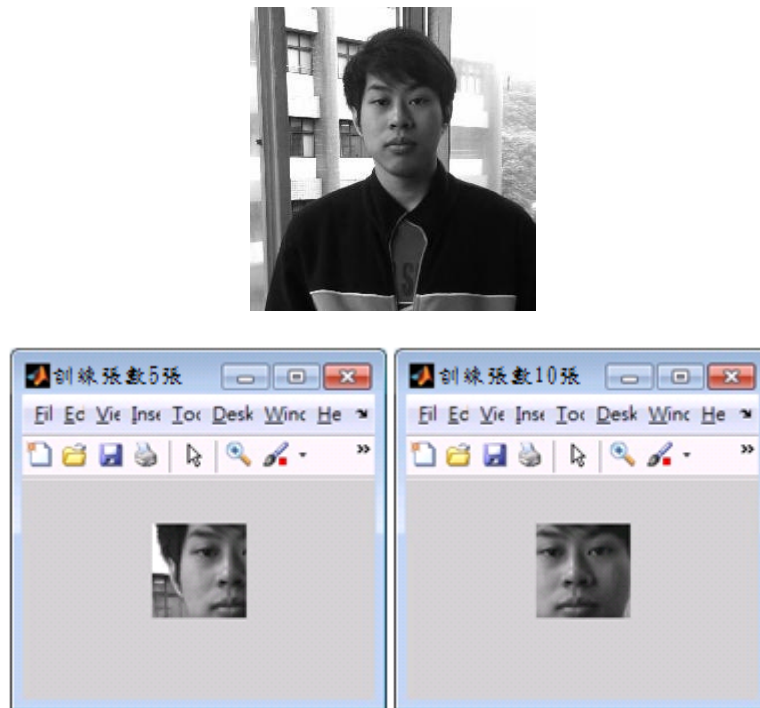


圖 5.21 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確

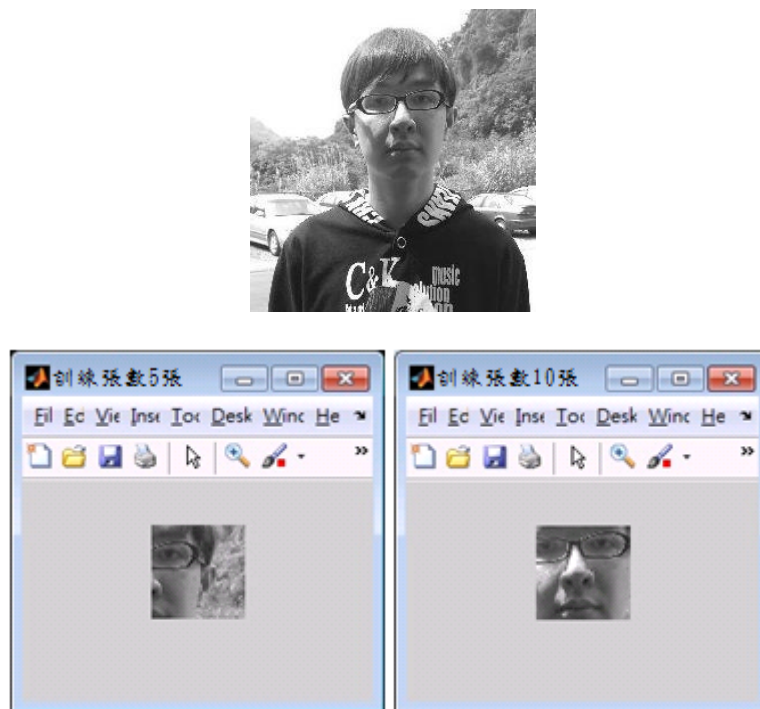


圖 5.22 Haar 函數 0 到 3 階層，Haar 函數展開係數 >30 ，偏移變準確

因上述的實驗結果中訓練 2 張圖的正確率比多張圖還高，故在下列再一次進行實驗以驗證實驗結果是否正確。此次實驗的 Haar 函數將提高，使得樣本 Haar 函數與新加入 Haar 函數時，一樣能夠清楚辨識出是人臉輪廓，如圖 5.23 至 5.26。

表 5.6 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，弱分類器數量=30 結果彙整表

訓練張數	1	2	3	6	9
Haar 函數展開係數數量	393	615	687	838	876
特徵求取時間	85.7 秒	275.4 秒	437.9 秒	805.7 秒	1379.3 秒
弱分類器 T 數量	30	30	30	30	30
訓練分類器時間	54751 秒 (15 小時)	120232 秒 (1.3 天)	173927 秒 (2 天)	446507 秒 (5.2 天)	780708 秒 (9 天)
150 張人臉辨識率 正確/總數	25/150	34/150	34/150	40/150	43/150

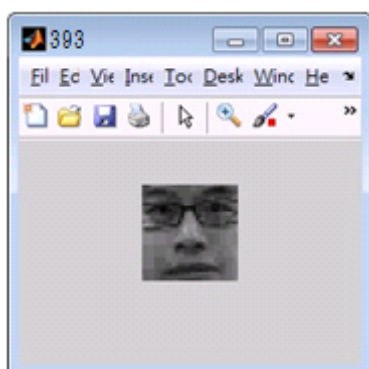


圖 5.23 使用 393 個 Haar 函數重組之人臉圖

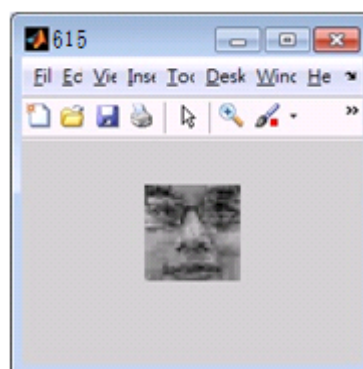


圖 5.24 使用 615 個 Haar 函數重組之人臉圖

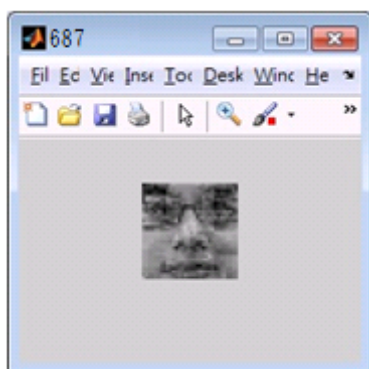


圖 5.25 使用 687 個 Haar 函數重組之人臉圖

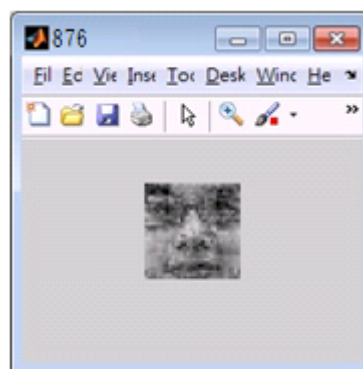


圖 5.26 使用 876 個 Haar 函數重組之人臉圖

以此實驗結果證明訓練 2 張圖並不會比多張圖的正確率還要高。

Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，訓練張數分別為 2、3 張，雖然將 Haar 函數數量增加但其特性還是不變，訓練張數較多的結果還是能將訓練張數較少且有辨識到人臉的影像定位更加準確，如圖 5.27 至 5.30。



圖 5.27 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確



圖 5.28 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確



圖 5.29 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確

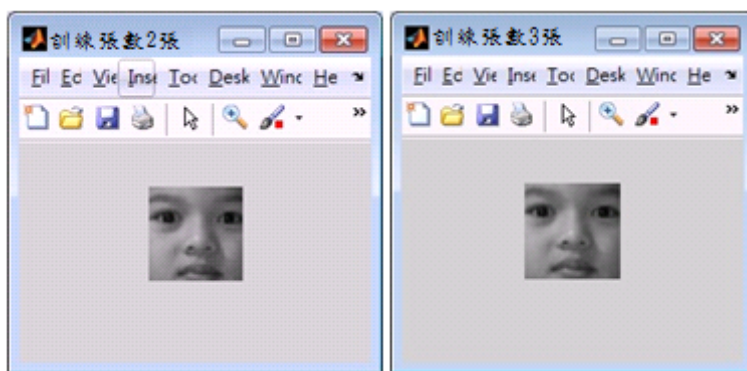


圖 5.30 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，定位更加準確

Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，訓練張數分別為 1、3 張，由表 5.7 可得知 Haar 函數數量增加而分類器數量不變時，能使訓練結果有些偏移的影像變為準確，如圖 5.31 至 5.33。



圖 5.31 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確



圖 5.32 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確

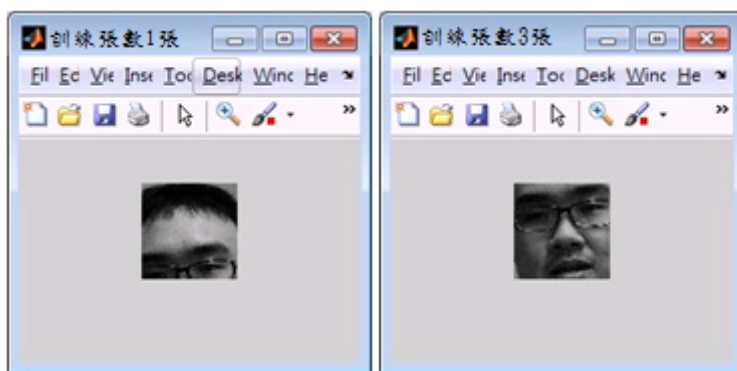


圖 5.33 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，偏移變準確

Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，訓練張數分別為 1、3 張，由表 5.6 可得知 Haar 函數數量大幅增加而分類器數量不變時，能使訓練結果錯誤辨識的人臉影像變為準確，如圖 5.34 至 5.39

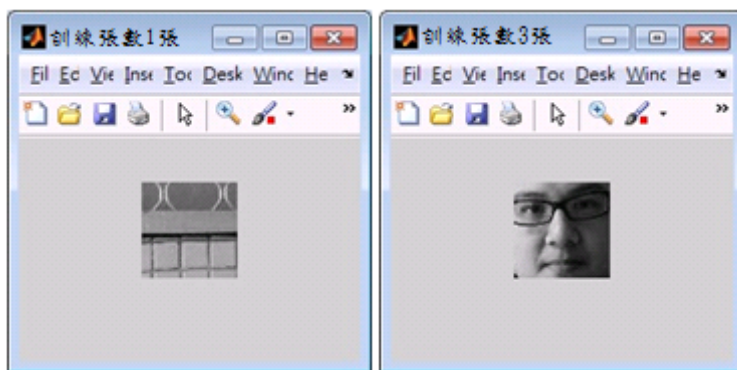


圖 5.34 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確



圖 5.35 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確



圖 5.36 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確

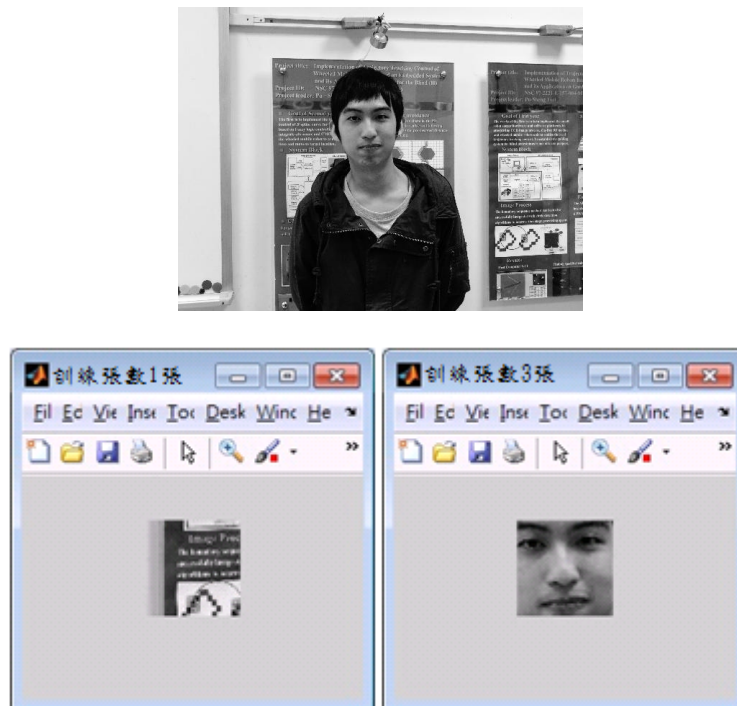


圖 5.37 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確

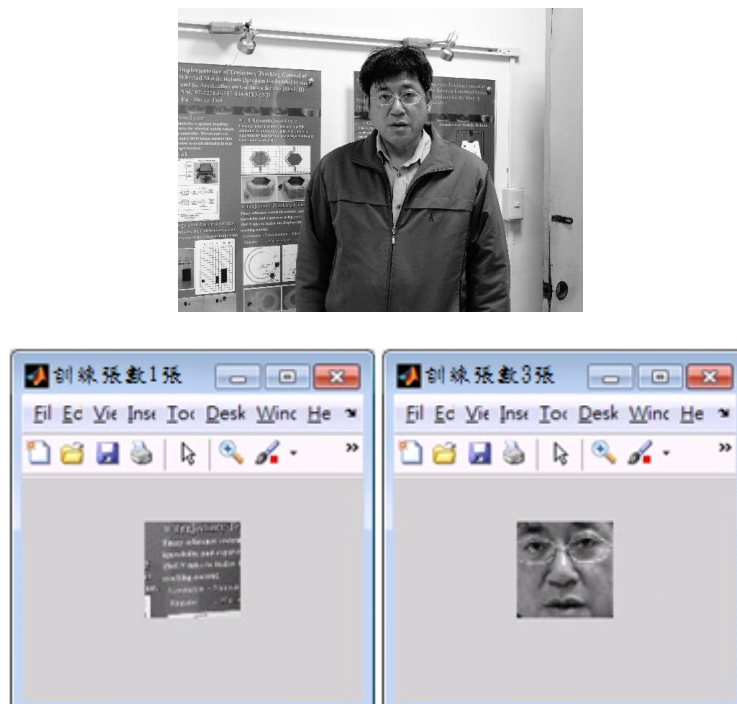


圖 5.38 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確



圖 5.39 Haar 函數 0 到 4 階層，Haar 函數展開係數 >20 ，錯誤變準確

第六章 結論與未來研究方向

第一節 結論

由實驗彙整表可發現，當 Haar 函數數量增加越多其正確辨識的影像就越多，故數量相差較多時，其人臉辨識的正確率會相差比較多，相差較少時雖無法增加正確率但能將辨識到的人臉定位得更加準確，而因為訓練的張數能使得弱分類器越來越細緻，所以有時候會將原本定位到的人臉產生一點偏移，若要解決此問題則使用更多影像加入訓練即可。

因圖片資料庫中，每張影像的光源背景、人臉明暗度、拍攝的角度與人臉的傾斜度皆不相同，故很容易造成誤判而形成辨識有誤。從實驗結果來看，雖然正確率不高，但是只要將訓練張數或弱分類器數量進行增加調整，其正確率還是有慢慢增加的趨勢，這樣就能不需要將所有數據都加入 AdaBoost 訓練。故將 Haar 函數數量與分類器數量減少以提高 AdaBoost 演算法訓練速度是可行的。

第二節 未來研究方向

在本文人臉辨識比對中，其未來能研究的方向如下：

1. 在本文將影像放入 AdaBoost 訓練時，因為拍攝的器材不同使得影像必須進行調整。而未來可發展使用任何拍攝器材都無須調整影像即可進行訓練辨識。
2. 在 AdaBoost 訓練時因加入新的參數使得每次的辨識都需再重新訓練樣本，未來可以研究直接將舊的訓練結果與新參數做結合即可不用每次都重新訓練。
3. 因為影像的明暗度、人臉明暗度與拍攝的角度等，很容易造成誤判而形成辨識有誤，未來可在人臉上加入明顯特徵，這樣就不會被影像影響辨識的正確率。

參考文獻

- [1] Paul Viola, Michael Jones. "Rapid object detection using a boosted cascade of simple features", Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 511-518, 2001.
- [2] Paul Viola and Michael Jones. "Robust Real-time Object Detection", IEEE International Journal of Computer Vision, vol. 57, no. 2, pp. 137-154, May 2004.
- [3] Y. Freund, R. E. Schapire. "Experiments with a new boosting algorithm", Machine Learning: Proceedings of the Thirteenth International Conference, Morgan Kaufman, San Francisco, pp. 148-156, 1996.
- [4] Ji Zhu, Saharon Rosset, Hui Zou, Trevor Hastie. "Multi-class AdaBoost", Technical report, Dept. of Statistics, University of Michigan, 2005.
- [5] V. P. Kumar, T. Poggio. "Learning-based approach to real time tracking and analysis of faces", MIT A. I. Memo, 1999, No. 1672.
- [6] 陳柏霖，以 Adaboost 演算法之車牌定位研究與實作，中華科技大學電子工程研究所碩士論文，民國九十九年七月。
- [7] 辛俊昇，於 HAAR 函數轉換空間之臉部識別演算法，中華科技大學在職進修碩士專班碩士論文，民國九十九年十二月。
- [8] 陳致豪，以膚色偵測加速 AdaBoost 人臉偵測，明志科技大學電子工程研究所碩士論文，民國一百零一年七月。
- [9] 詹忻澂，混合 HAAR-LIKE 特徵與特徵相對位置之人臉偵測，國立屏東教育大學資訊科學系碩士論文，民國九十九年七月。
- [10] 賴昆煒，基於 Radon 轉換之人臉辨識系統，銘傳大學電腦與通訊工程學系碩士班碩士論文，民國一百零一年七月。
- [11] 吳旻峰，基於像素導向之階層式特徵與統計式遮罩 Adaboost 人臉偵測，國立臺灣科技大學電機工程系碩士學位論文，民國九十八年六月二十九日。
- [12] 翁偉鐘，利用 Adaboost 演算法於數位影像中進行文字定位，亞洲大學資訊與設計學系碩士班碩士論文，民國九十七年七月。
- [13] 楊攸淇，基於 Adaboost 演算法使用多重感興趣區域以提升偵測距離之多重

- 物件偵測，逢甲大學電子工程學系碩士班碩士論文，民國一百零二年六月。
- [14] 莊啟鴻，使用五官特徵之人臉辨識，義守大學資訊管理學系碩士論文，民國一百零一年六月。
- [15] 黃韋瀚，結合五官特徵與深度資訊之人臉辨識系統，大同大學資訊工程研究所碩士學位論文，民國一百零一年七月。
- [16] 李昇龍，基於增量學習之人臉辨識研究，國立台灣師範大學資訊工程研究所碩士論文，民國一百年六月。
- [17] 陳宣有，結合特徵點與雙向辨識演算法的人臉辨識技術，中華大學資訊工程學系碩士班碩士論文，民國一百零三年七月。
- [18] 王士維，複雜背景下之人臉偵測系統之設計，明新科技大學電子工程研究所碩士學位論文，民國一百零一年七月。
- [19] 陳柏州，臉部特徵抽取之人臉辨識，義守大學資訊管理研究所碩士論文，民國一百零二年七月。
- [20] 繆紹綱，數位影像處理，普林斯頓國際有限公司，2004 年 10 月。
- [21] 洪維恩，MATLAB 7 程式設計，旗標出版股份有限公司，2009 年 3 月。

附
錄
圖
片
資
料
庫

圖片資料庫:

