

國立臺灣大學電機資訊學院電子工程學研究所

碩士論文

Graduate Institute of Electronics Engineering  
College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis



基於簡單資料擴增學習之快速非共平面人臉偵測  
Fast Out-of-Plane Face Detection Based on Learning with  
Simple Data Augmentation

劉信宏

Hsin-Hung Liu

指導教授：簡韶逸博士

Advisor: Shao-Yi Chien, Ph.D.

中華民國 105 年 7 月

July, 2016



國立臺灣大學碩士學位論文  
口試委員會審定書

基於簡單資料擴增學習之快速非共平面人臉偵測  
Fast Out-of-Plane Face Detection Based on Learning  
with Simple Data Augmentation

本論文係劉信宏君 (P97921001) 在國立臺灣大學電子工程學研究所完成之碩士學位論文，於民國 105 年 7 月 28 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

簡 紹 造

王 鈺 強

蔣 志 璋

系主任、所長

劉 信 宏







## 誌謝

在校的求學生涯即將進入尾聲，在此我必須要感謝我的指導教授、簡韶逸博士，在方向擬定、調查執行、實驗與論文撰寫的期間，給予關鍵的支持、多樣的資源、必要的指導、與揮灑的空間，使得本篇論文得以順利完成。同時也要感謝諸位口試委員細心審查論文內容並提供許多寶貴的建議，讓我在此次階段性學業的終點、儘可能完善研究成果。儘管一路走來甘苦無法言喻，我仍要再次感謝在校其間給予過任何協助的教授、行政人員、同學、朋友們，讓我有機會走完學業全程，帶著不一樣的自己重回業界，再次感謝大家！





## 摘要

隨著行動裝置的普及化，各式相關應用不斷地被開發以滿足使用者所需。然而受限於可攜性要求之先天限制，無法無止盡地擴增其計算能力去應付眾多需求。基於此一考量，本論文針對人臉偵測提出一種新的訓練方式，藉由簡單資料擴增的學習模式讓行動裝置在僅增加少額計算量的情形下強化非共平面人臉的偵測能力，經由多種數據集的實驗比對確認為可行做法。其有效率的功能強化在提升性能的同時亦保留計算能力允許行動裝置同時執行更多功能以利開發更多豐富的應用。





# Abstract

With the popularity of mobile devices, all kinds of related applications are constantly being developed to meet user needs. However, subject to the requirements of portability restrictions, we can not endlessly amplified its computing performance to cope with the many demands. Based on this consideration, this paper proposes a new training methods for face detection. The learning by simple data augmentation allows mobile devices to strengthen the out-of-plane face detection capabilities with little increase of computed consumption. According to the experiments of various data sets, the proposed method is recognized as a viable approach. The functional enhancement with efficiency not only improves the performance but also allows mobile devices to perform more functions simultaneously to facilitate the development of more rich applications.





# Contents

口試委員會審定書	iii
誌謝	v
摘要	vii
<b>Abstract</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Related Works</b>	<b>3</b>
2.1 Weak Classifier . . . . .	3
2.1.1 LBP . . . . .	3
2.1.2 Haar-like Feature、HOG . . . . .	4
2.2 Strong Classifier . . . . .	5
2.2.1 Adaboost . . . . .	5
2.2.2 SVM、Neural Network . . . . .	5
2.3 Cascade Classifier . . . . .	6
<b>3 Proposed Algorithm</b>	<b>9</b>
3.1 Problems . . . . .	9
3.2 Training Flow . . . . .	9
3.3 Samples Preparation . . . . .	10
3.4 Samples Expansion and Mixing . . . . .	14
3.5 Training Results . . . . .	18

<b>4 Experiment</b>	<b>23</b>
4.1 Test Method . . . . .	23
4.2 Test Results . . . . .	24
4.3 Analysis . . . . .	26
<b>5 Conclusion</b>	<b>33</b>
<b>Bibliography</b>	<b>35</b>







# List of Figures

1.1	側臉檢測 . . . . .	1
2.1	Local Binary Pattern . . . . .	3
2.2	Haar-like Feature . . . . .	4
2.3	Histograms of Oriented Gradients . . . . .	4
2.4	Gentle Adaboost . . . . .	5
2.5	Support Vector Machine . . . . .	6
2.6	Neural Network . . . . .	6
2.7	Cascade Classifier . . . . .	7
2.8	OpenCV Xml Format . . . . .	8
3.1	Training Flow . . . . .	10
3.2	Labeled Faces in the Wild-a (LFW-a) . . . . .	12
3.3	MIT CBCL Face Data . . . . .	12
3.4	XnView 批次轉換 . . . . .	13
3.5	示意圖，將人頭視為圓柱體將正臉拉平展開 . . . . .	14
3.6	程式碼範例，將人頭視為圓柱體將正臉拉平展開 . . . . .	15
3.7	示意圖，正負樣本混合 . . . . .	16
3.8	正負樣本混合後的等效視角 . . . . .	16
3.9	程式碼範例，正負樣本混合 . . . . .	17
3.10	正負樣本混合前後的特徵取樣情形 . . . . .	17
3.11	串接式分類器負樣本衰減變化情形 . . . . .	19
3.12	LFW-a 延展版訓練結果, LBP 實際分布情形 . . . . .	20
3.13	MIT CBCL 延展版訓練結果, LBP 實際分布情形 . . . . .	21

4.1	XnView 建立縮圖表 . . . . .	23
4.2	測試結果 . . . . .	25
4.3	人臉樣本對照圖, 成功大學機器人實驗室與 MIT CBCL . . . . .	26
4.4	LFW-a 非延展測試, Pointing'04 ICPR Workshop . . . . .	27
4.5	LFW-a 延展測試, Pointing'04 ICPR Workshop . . . . .	27
4.6	LFW-a 非延展測試, 成功大學機器人實驗室 . . . . .	28
4.7	LFW-a 延展測試, 成功大學機器人實驗室 . . . . .	28
4.8	LFW-a 非延展測試, FEI Face Database . . . . .	29
4.9	LFW-a 延展測試, FEI Face Database . . . . .	29
4.10	LFW-a 測試對照表, Pointing'04 ICPR Workshop . . . . .	30
4.11	LFW-a 測試對照表, 成功大學機器人實驗室 . . . . .	30
4.12	LFW-a 測試對照表, FEI Face Database . . . . .	30
4.13	MIT CBCL 測試對照表, Pointing'04 ICPR Workshop . . . . .	31
4.14	MIT CBCL 測試對照表, 成功大學機器人實驗室 . . . . .	31
4.15	MIT CBCL 測試對照表, FEI Face Database . . . . .	31





# List of Tables

3.1	訓練結果, LFW-a . . . . .	18
3.2	訓練結果, MIT CBCL . . . . .	18





# Chapter 1

## Introduction



Figure 1.1: 側臉檢測, 取自 [38]

隨著科技的進步，不論大型主機或是行動裝置，其計算能力的提升早已不可同日而語。多個演算法平行處理所複合出的各種應用，像是多種物件的同步辨識可擴展成情境的分析、甚至是近來備受重視的汽車自動駕駛都將成為可能。然而行動裝置受限於可攜性要求之先天限制，無法無止盡地擴增其計算能力去應付眾多需求，因此針對行動裝置開發各式高效率的演算法便成為值得發展的研究課題。其中尤以人臉偵測或識別的相關技術長久以來始終備受關注，在眾多的演算法中，Paul Viola *et al.* [22] 基於 Integral Image、Haar-like Feature 與 Adaboost 所發表的串接式分類器 (Cascade Classifier) 兼具即時性及準確度，Shengcai Liao *et al.* [18] 所提出的 Multi-scale Block LBP Feature 在大致維持其準確度的同時大幅降低了訓練時間，自此串接式分類器開始進入適合行動裝置的實用階段，而然其先天屈居弱勢的多角度識別能力至今鮮少有人投入研究，甚為可惜。基於此一考量，本論

文針對人臉偵測提出一種新的訓練方式，藉由簡單資料擴增的學習模式讓行動裝置在僅增加少額計算量的情形下強化非共平面人臉的偵測能力，經由多種數據集的實驗比對確認為可行作法。其有效率的功能強化在提升性能的同時亦保留計算能力允許行動裝置同時執行更多功能以利開發更多豐富的應用。





# Chapter 2

## Related Works

物件偵測 (Object Detection) 技術的發展由來已久，本章將針對部份的主流做法簡單描述其優缺點，並依據分析選取最適合在可攜式裝置上執行的方法。

### 2.1 Weak Classifier

#### 2.1.1 LBP

弱分類器 (Weak Classifier) 為識別物件的基本單元，顧名思義它是種解析能力微弱的分類器。在這裡我們先介紹 LBP，LBP 為 Local Binary Pattern 的縮寫，其內容相當簡單，即是將待分析區分成等大小的 9 等份，以中心區塊為基準值跟外圍區域比對，大於或等於基準值者設為 1、小於則設為 0。初期是針對紋理分析所設計的特徵，但後續的研究顯示對於人臉偵測也有不錯的表現。



Figure 2.1: Local Binary Pattern

## 2.1.2 Haar-like Feature、HOG

其它兩個常見的特徵還有 Haar-like Feature 及 HOG (Histograms of Oriented Gradients)。Haar-like Feature 是最先配合 Adaboost 物件辨識所設計的特徵，由於外形類似 Haar Wavelet、因此得名，其特徵的計算為先求出黑白兩區塊的個別像素和再計算兩者差值。在實際測試上顯示其效果略優於 LBP，然而在數據處理上因為必須再做光影的標準差修正，使得 Haar-like Feature 最終轉為浮點數，導致不利於可攜式裝置執行。另外一個困擾為以 Haar-like Feature 為基礎的樣本訓練相當費時，可能得耗時數天方可完成、不利於參數調校，為其兩大缺點。

至於 HOG 則為方向梯度的統計直方圖，對於邊緣特徵分析有較佳的效果，故較適合像是行人偵測等比較強調輪廓識別等方面的應用。但是因為其高維度數據導致計算量龐大，加上 OpenCV 3 [26] 已不支援此特徵於串接式分類器中使用，故不再深入討論。

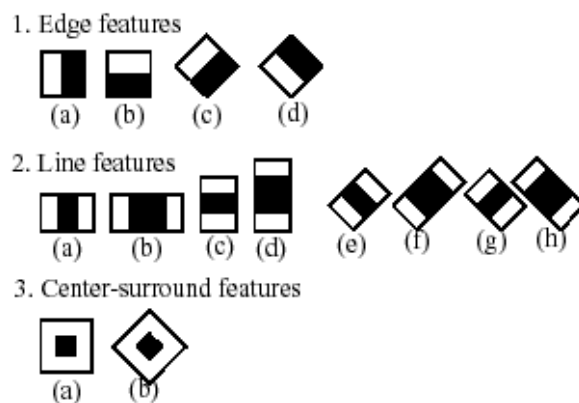
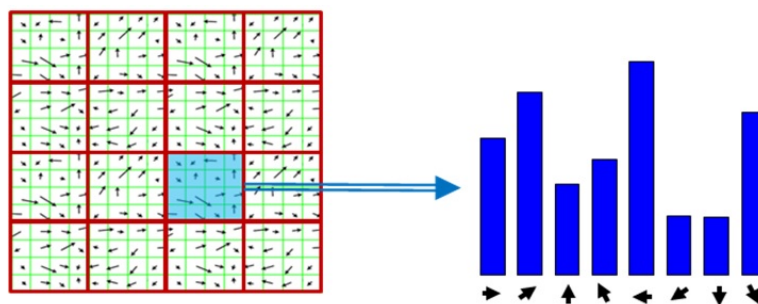


Figure 2.2: Haar-like Feature, 取自 [23]





## 2.2 Strong Classifier

### 2.2.1 Adaboost

Adaboost 為 Adaptive Boosting 縮寫，字面上意義”逐步適應提升”頗能代表其精神，也就是逐步選取最佳的弱分類器、一步步提升分類能力後組合成單一的強分類器 (Strong Classifier)。目前最常用的 Adaboost 強分類器演算法為 Gentle Adaboost，這種 boosting 並不強調立即辨認出最多的正負樣本，而是藉由計算 regression function 逐級找出整體誤差最小的特徵，並根據被選取的特徵更新樣本權重，最後再由這些特徵組合出強分類器。

#### Gentle AdaBoost

1. Given  $N$  examples  $(x_1, y_1), \dots, (x_N, y_N)$  with  $x \in \mathcal{R}^k, y_i \in \{-1, 1\}$
2. Start with weights  $w_i = 1/N, i = 1, \dots, N$ .
3. Repeat for  $m = 1, \dots, M$ 
  - (a) Fit the regression function  $f_m(x)$  by weighted least-squares of  $y_i$  to  $x_i$  with weights  $w_i$
  - (c) Set  $w_i \leftarrow w_i \cdot \exp(-y_i f_m(x_i))$ ,  $i = 1, \dots, N$ , and renormalize weights so that  $\sum_i w_i = 1$ .
4. Output the classifier  $\text{sign}\left[\sum_{m=1}^M f_m(x)\right]$

Figure 2.4: Gentle Adaboost, 取自 [23]

### 2.2.2 SVM、Neural Network

其它兩個常見的強分類器則是 SVM (Support Vector Machine) 及 Neural Network。SVM 的做法是把訓練樣本丟入高維空間後，嘗試在空間內找出能明確切割正負樣本群組的超平面 (Hyperplane)。其優點是不需要大量的訓練樣本，只要樣本數量足以建立有效的數學模型即可，但其優點同時也造成了無可避免的缺點，導致爾後所有待測物都必須套入此數學模型才能判定是否為目標物件，以致檢測速度無法和 Adaboost 所訓練出的強分類器相比擬。至於藉由模擬神經系統而發展出的 Neural Network 其計算量大參數又多，先天就不利於可攜式裝置的執行，故不在此討論。

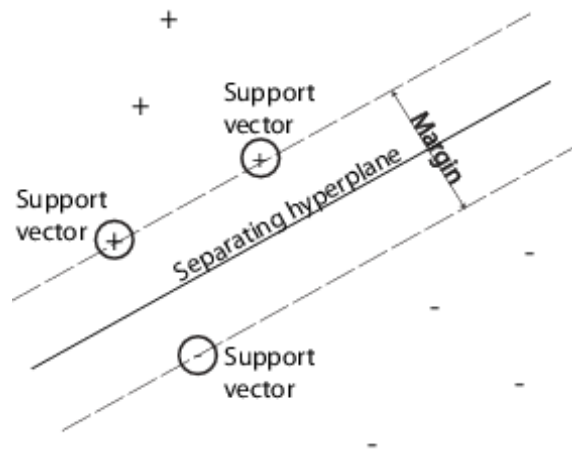


Figure 2.5: Support Vector Machine, 取自 [40]

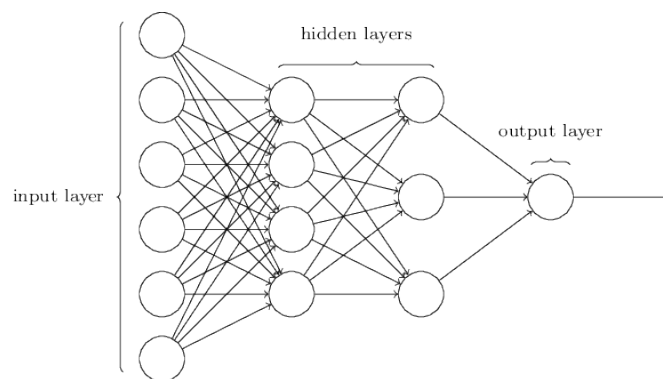


Figure 2.6: Neural Network, 取自 [41]

## 2.3 Cascade Classifier

為了加速檢測的收斂並增進執行的效率，我們將強分類器逐級串接成串接式分類器 (Cascade Classifier)。相較於設計成單一強分類器，待測物不必歷經完整的檢查即可由串接式分類器逐步汰除非目標物進而有效降低整體的計算量。以一般的設定 Hit Rate (正確抓出目標物的比率) 趨近於 100% 而 False Alarm Rate (將非目標物誤判為目標物的比率) 為 50% 以下為例，理論上每一級強分類器會掃除上一級留下來的非目標物至少一半以上，反過來若改以單一強分類器的結構來處理，那些原本會被串接式分類器中途丟棄的非目標物將無可避免地被嚴格檢查，這些不必要的計算白白浪費了可攜式裝置寶貴的資源，因此以串接式分類器建構物件辨識系統確實為必要之做法。

最後我們來看看 OpenCV 如何使用 Xml 檔案描述串接式分類器。如 Figure 2.8 所示，Xml 內部被分成三個部份，第一部份是記錄了訓練參數的設定 (Figure 2.8

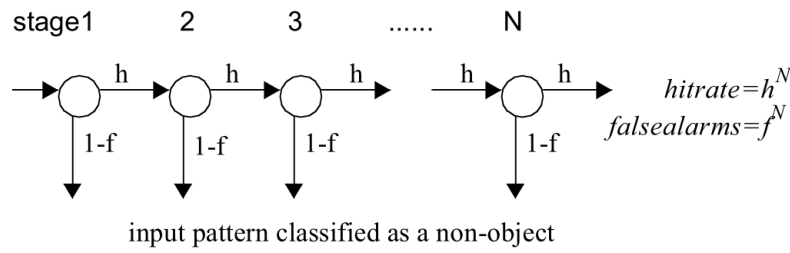


Figure 2.7: Cascade Classifier, 取自 [23]

並未列出)，像是選用 Adaboost 的種類、Hit Rate、False Alarm Rate 等。第三部份是特徵列表，以綠色框所選的第 1 個特徵來說 (特徵從 0 開始計數)，3、2、9、7 分別表示 X 起始位置為 3、Y 起始位置為 2、長為 9、寬為 7 的矩形框為 LBP 的第一個區塊，並依序計算出等大小的其它 8 個區塊，得到最終 8-bit 的 LBP 數據。然後我們再回頭審視第二部份，Stage 15 表示串接式分類器中從第 0 個開始算起的第 15 個強分類器，interNodes 表示各級強分類器裡所使用的弱分類器，前兩個數據”0, -1”表示是最單純的樹狀結構，第 3 個數據”1” (綠框所示) 表示選用特徵列表裡的第 1 個特徵，其餘的 8 組數據則組合成 256-bit 的 LUT(Look Up Table)，根據 LBP 數據到此 LUT 查表，來判斷是否為我們需要的物件，若是目標物，取出右側的 leafValues，反之則出左側的 leafValues，並於最後將所有的 leafValues 加總在一起，倘若其值大於 stageThreshold，表示待測物通過檢測，將其送入下一級強分類器做更進一步的解析，反之則直接丟棄不再處理，如此周而復始的運作直至整個檢測作業完成為止。



```

<!-- stage 15 -->
<_>
  <maxWeakCount>9</maxWeakCount>
  <stageThreshold>-1.0141675174236298e-001
  </stageThreshold>
  <weakClassifiers>
    <_>
      <internalNodes>
        0 -1 50
        -43553 1710194655 -43937 -541102097
        -168826897 -184549921 -1532542655 -4258609
      </internalNodes>
      <leafValues>
        -4.1064536571502686e-001 5.4582500457763672e-001
      </leafValues></_>
    <_>
      <internalNodes>
        0 -1 1
        -123734569 -728579295 -1065295872 -2067596094
        -2080257359 -2130605428 -48921708 -189349953
      </internalNodes>
      <leafValues>
        -5.0367355346679688e-001 4.4452941417694092e-001
      </leafValues></_>
  </weakClassifiers>
</_>

```

the content of LUT

```

<features>
  <_>
    <rect>
      3 2 3 2</rect></_>
  <_>
    <rect>
      3 2 9 7</rect></_>
</features>

```

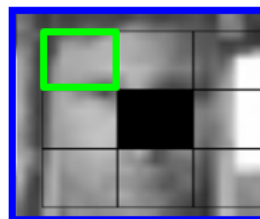


Figure 2.8: OpenCV Xml Format



## Chapter 3

# Proposed Algorithm

### 3.1 Problems

根據上一章的介紹與分析，我們選定 Adaboost 配合 LBP 做為本論文演算法的原型。考量訓練多角度人臉的串接式分類器會有下列的問題：

1. 訓練樣本以正臉為大宗，多角度側臉樣本極少。
2. 正臉、側臉分別訓練，會迫使系統於予個別檢測，增加計算負擔。
3. 正臉、側臉合併訓練，會模糊特徵，增加檢測難度。

在本章我們將提出方法嘗試克服上述的問題，並於下一章於予測試驗證。

### 3.2 Training Flow

Figure 3.1 的流程圖描述了完整的訓練步驟，簡要說明如下：

1. 準備合適的正負樣本，第 3.3 節會有詳細的說明與建議。
2. 分析評估正樣本的立體形狀，將正樣本圖面套上此立體面，藉此將正樣本集轉換成可旋轉的近似模型。
3. 正負樣本混合處理(局部挖取展開圖，空白部份隨機貼上負樣本。)，目的在於產生旋轉後的近似特寫，用以建立最終的正樣本集。

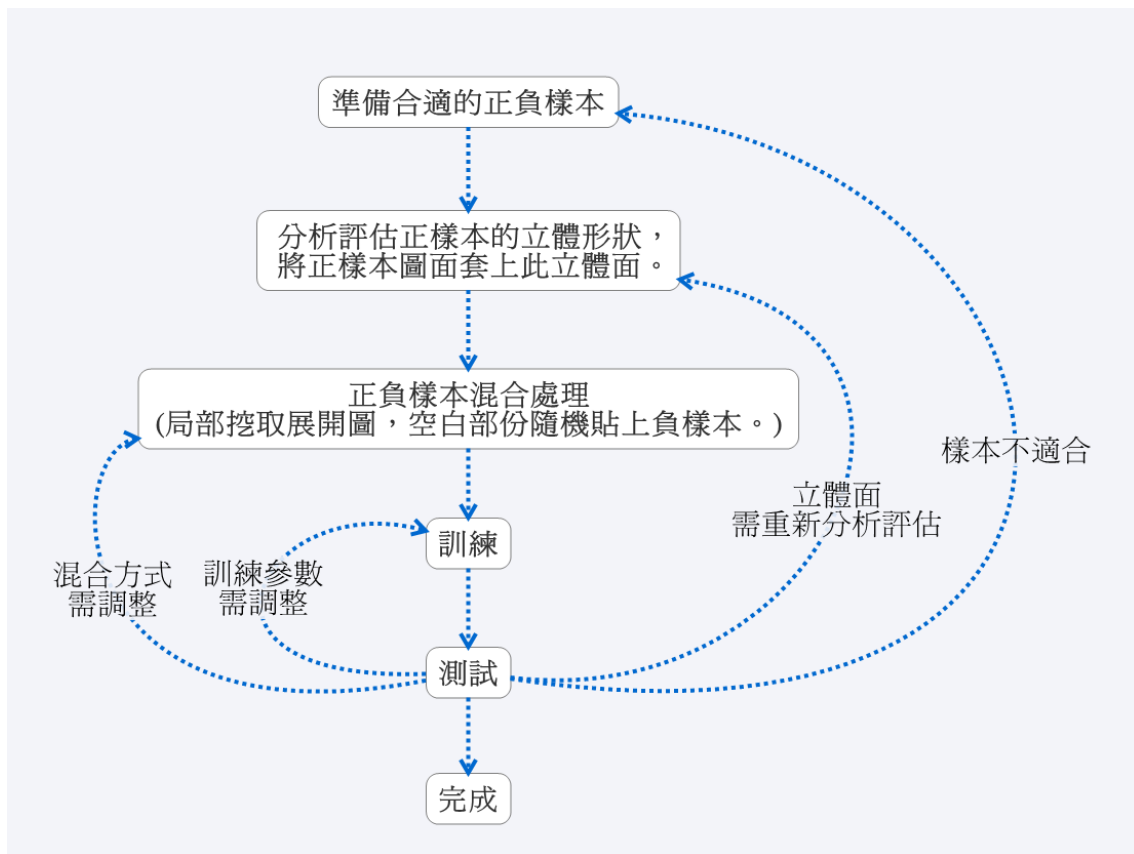


Figure 3.1: Training Flow

4. 使用新的正樣本集進行訓練。以上三點的實際操作過程將在第 3.4 節於以解釋。
5. 訓練結果測試，實測的結果在第 4 章會有進一步的討論。

### 3.3 Samples Preparation

OpenCV 訓練的實際操作方法，由於 OpenCV 官方網站及相關討論網頁已提供完整且豐富的說明，故不在本篇論文內重覆贅述。不過在開始訓練前的正負樣本準備作業，若能留意下面所述的注意事項，將有助於提升串接式分類器的性能，並降低異常訓練結果的發生機會。

正樣本的建置原則 (以人臉為例)：

1. 固定每張樣本的大小是必要條件。

2. Adaboost 的訓練必須仰賴大量樣本的篩檢始可產生足夠強度的分類器。因此在實務上，正樣本至少應有上千張以上不同的人臉以確保分類器訓練的品質。
3. 每張樣本儘可能完全排除背景、只保留人臉的部份。
4. 除非是刻意強化對人臉旋轉的容忍度，否則請儘量讓五官固定於圖像的特定位置，將有助於程式找出辨識能力最強的特徵，加速訓練的收斂。
5. 若不得已必須保留背景資訊時(例如欲保留完整的頭髮圖像時，自然會夾入部份背景)，必須儘可能確保背景的多樣性，以避免規律的背景在訓練過程中被誤認為是人臉的一部份導致建立錯誤的判斷機制。



負樣本的建置原則：

1. 負樣本儘可能符合現實並多樣化，例如不太可能出現裸臉的深海環境就不是好的負樣本圖像來源。
2. 負樣本的內容完全不包含任何局部或完整的正樣本。
3. 圖像的數量不可少(至少數百張)，圖面的大小必須大於正樣本、而且愈大愈好，以便程式可以隨機取樣產生足量的多樣性負樣本。相關細節將於第 3.5 節做更進一步的探討。

正樣本的部份，本次研究使用了 Labeled Faces in the Wild-a (LFW-a) [27]。此人臉資料庫有經過調校處理，將眼睛、鼻子與嘴巴做過定位校正，是非常適合用來訓練人臉的樣本。此外我們亦收集了 MIT CBCL Face Data [30] 做為另一組比對用的正樣本訓練集，至於負樣本則使用一份存放於 Google Code Archive 的圖庫 [31]。由於收集到的正樣本往往必須先行預處理方可使用，建議可善用免費軟體 XnView [37] 的”批次轉換”功能裁取我們需要的部份(界面如 Figure 3.4 所示)，可節省不少作業時間。

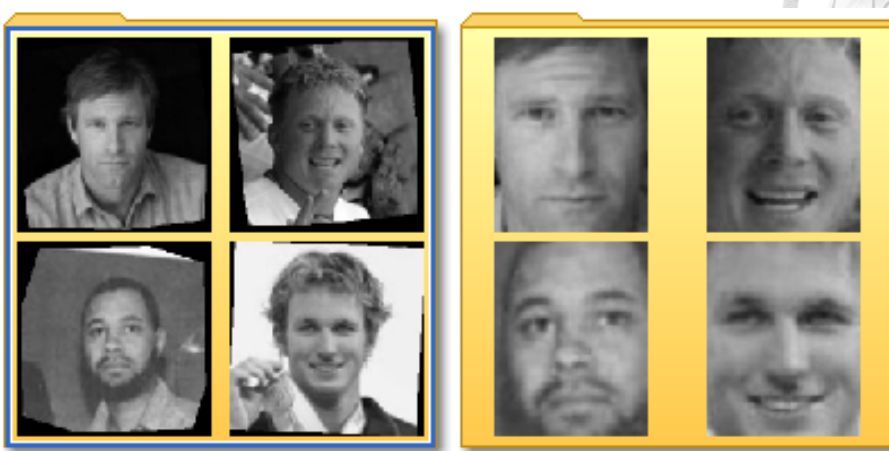


Figure 3.2: Labeled Faces in the Wild-a (LFW-a)

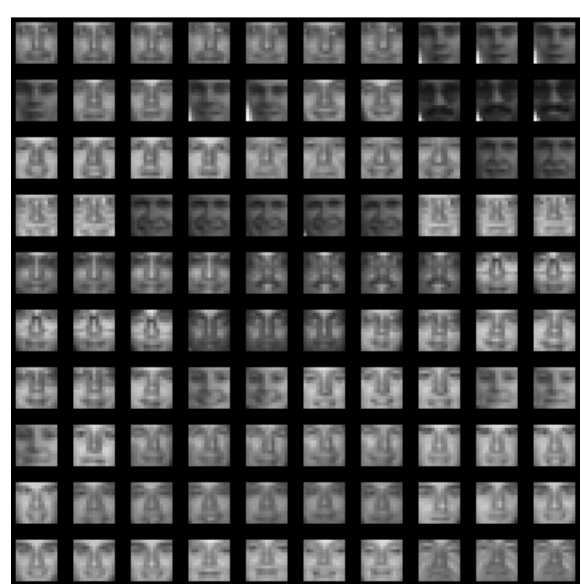


Figure 3.3: MIT CBCL Face Data



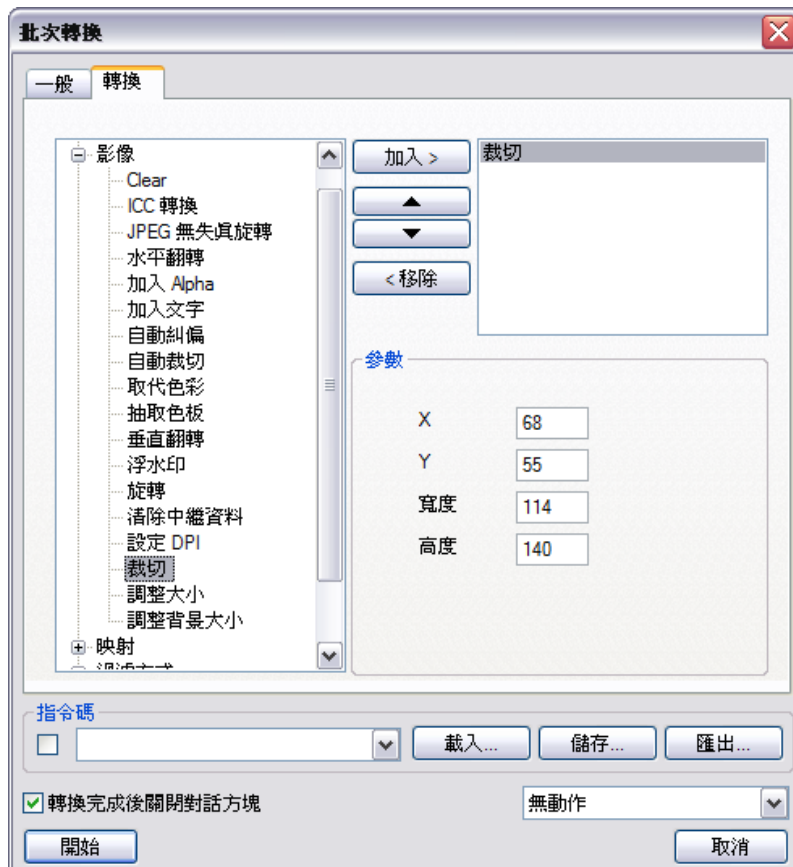


Figure 3.4: XnView 批次轉換

### 3.4 Samples Expansion and Mixing



在正式開始訓練前，必須對正樣本進行關鍵的正負樣本混合處理，根據多角度人臉辨識的需求，我們規劃的方式如下：

1. 將人頭視為圓柱體，用程式將正臉拉平展開。
2. 依正視圖的“長寬”挖取展開圖，得到五份變形的正樣本。空白部份隨機挖取負樣本來補上。

Figure 3.5 是將平面人臉映射到半周長圓柱形弧面的示意圖。首先我們先定義好弧面上每一個像素的位置，將每個紅點視為新展開圖的 X 軸座標整數值，然後用簡易的三角函數計算出每一個紅點投影到正視圖的位置  $x_l$  與  $x_r$ ，並依公式 3.1 至 3.6 計算相鄰像素的權重再以線性插入的方法求得  $x_l$  與  $x_r$  的像素值。如此反覆處理直到完成所有的像素計算為止。至於實際引用 OpenCV 模組所編寫出的程式碼可參考 Figure 3.6。

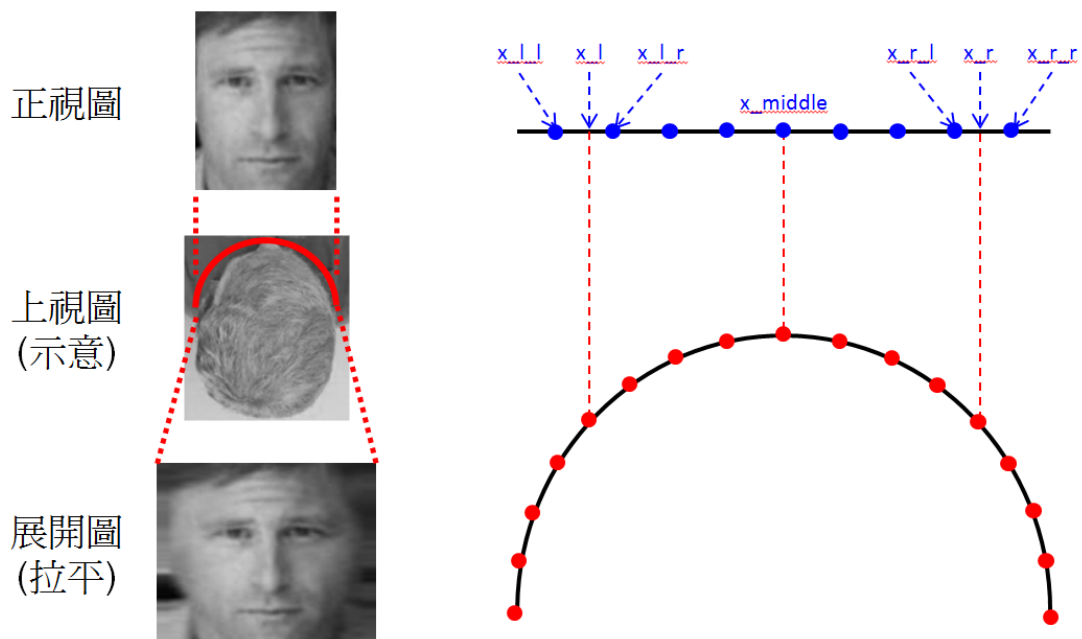


Figure 3.5: 示意圖，將人頭視為圓柱體將正臉拉平展開

$$x_l = x_{l_l} * w_{x_{l_l}} + x_{l_r} * w_{x_{l_r}} \quad (3.1)$$

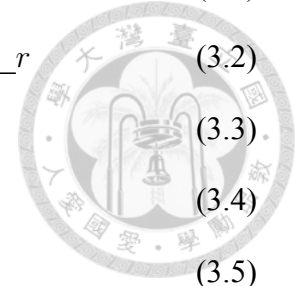
$$x_r = x_{r_l} * w_{x_{r_l}} + x_{r_r} * w_{x_{r_r}} \quad (3.2)$$

$$w_{x_{l_l}} = |x_l - x_{l_r}| \quad (3.3)$$

$$w_{x_{l_r}} = |x_l - x_{l_l}| \quad (3.4)$$

$$w_{x_{r_l}} = |x_r - x_{r_r}| \quad (3.5)$$

$$w_{x_{r_r}} = |x_r - x_{r_l}| \quad (3.6)$$



將正樣本展開完成後，下一步就要進行正負樣本的實際混合。因為延展後

```

21 // 正樣本延展(2D轉3D再攤平)開始。Y軸為一對一對應，不用調整。
22 for (y=0; y<=posimg.rows-1; y++)
23 {
24     // x軸自中心位置起左右一併處理
25     for (i=0; i<=posimg_ex_cols/2-1; i++)
26     {
27         // x軸的中心位置
28         float x_middle = posimg.cols/2-0.5;
29         // x軸距離中心的偏移量
30         float x_offset = sin((i+0.5)/(posimg.cols/2))*(posimg.cols/2);
31         // 中心位置左側對應到的轉換前位置，
32         // 疑似有小數點計算誤差、故以 conditional expression 修正。
33         float x_l = x_middle - x_offset < 0 ? 0 : x_middle - x_offset;
34         // 中心位置右側對應到的轉換前位置，
35         // 疑似有小數點計算誤差、故以 conditional expression 修正。
36         float x_r = x_middle + x_offset >= posimg.cols-1
37             ? posimg.cols-1 : x_middle + x_offset;
38         // 中心位置左側對應到的轉換前位置之相鄰左側像素
39         float x_l_l = floor(x_l);
40         // 中心位置左側對應到的轉換前位置之相鄰右側像素
41         float x_l_r = floor(x_l)+1;
42         // 中心位置右側對應到的轉換前位置之相鄰左側像素
43         float x_r_l = ceil(x_r);
44         // 中心位置右側對應到的轉換前位置之相鄰右側像素
45         float x_r_r = ceil(x_r)-1;
46         // linear interpolation 權重計算
47         float w_x_l_l = abs(x_l - x_l_r);
48         // linear interpolation 權重計算
49         float w_x_l_r = abs(x_l - x_l_l);
50         // linear interpolation 權重計算
51         float w_x_r_l = abs(x_r - x_r_r);
52         // linear interpolation 權重計算
53         float w_x_r_r = abs(x_r - x_r_l);
54         // 依權重計算儲存新圖像素值
55         posimg_ex.at<uchar>(y, posimg_ex_cols/2-1-i)
56             = int(posimg.at<uchar>(y, x_l_l)*w_x_l_l
57                 + posimg.at<uchar>(y, x_l_r)*w_x_l_r);
58         // 依權重計算儲存新圖像素值
59         posimg_ex.at<uchar>(y, posimg_ex_cols/2 +i)
60             = int(posimg.at<uchar>(y, x_r_l)*w_x_r_l
61                 + posimg.at<uchar>(y, x_r_r)*w_x_r_r);
62     }
63 }

```

Figure 3.6: 程式碼範例，將人頭視為圓柱體將正臉拉平展開

正樣本的寬度擴展為 1.57 倍，若依原始正樣本的大小重新在延展的正樣本取樣，分成五次位移、每次移動 0.14 倍評估為合理的做法，即不會太過密集產生過量的新樣本，也不致於過度跳躍造成圖面不連續。根據幾何計算，此做法

可以將原本的人臉正視圖擴展出約-30,-15,+15,+30 四種額外的角度(精確的角度是-32,-16,+16,+32)，然後連同 0 度的擴展圖共五組視為新的正樣本並開始訓練。至於正樣本以外的區域不可留白，必須以亂數取樣的方式自負樣本集內取出圖片貼上，以迫使程式在訓練過程中無法自這些區域找出任何規則，如此便不會干擾到正樣本區域的訓練作業。以上的做法具有下列的優點：

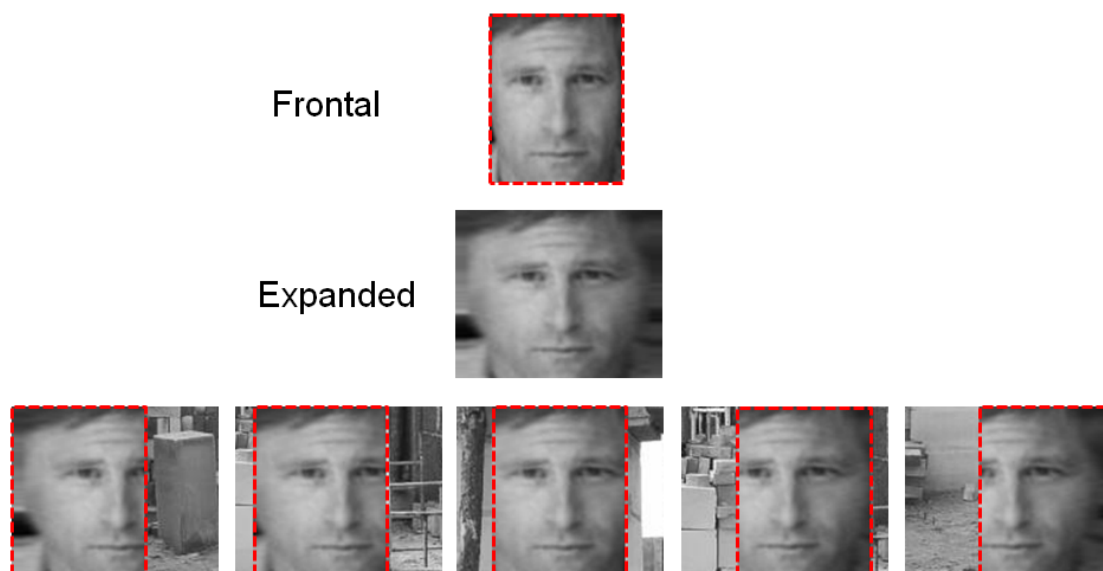


Figure 3.7: 示意圖，正負樣本混合

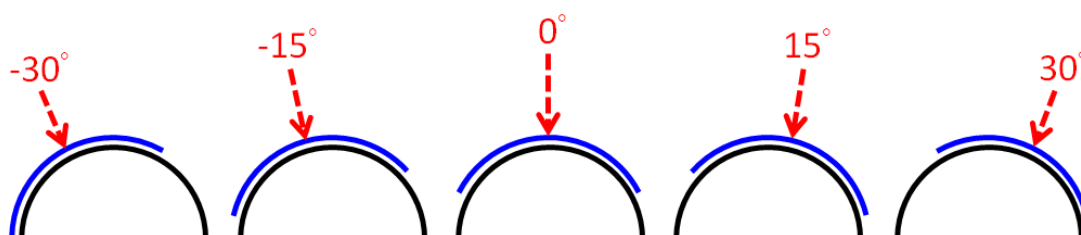


Figure 3.8: 正負樣本混合後的等效視角

1. 一張單一角度的正樣本直接生出五個角度的變形，夾帶的負樣本區域則會在訓練時被忽視，不影響訓練結果。
2. 同位置的特徵只要落在正樣本區域內皆等效，可有效節省特徵取樣，不像傳統訓練方法其同一位置特徵無法在不同角度的樣本內一致對應，進而被迫各自尋找合適自己的特徵、增加計算負擔。

```

86 // 五次迴圈，分別處理五次正負樣本的合併與存檔
87 for (i=1; i<=5; i++)
88 {
89     // 負樣本隨機設定y軸起始位置
90     int y_o=rand()%((negimg.rows - posimg_ex.rows)+1);
91     // 負樣本隨機設定x軸起始位置
92     int x_o=rand()%((negimg.cols - posimg_ex.cols)+1);
93     // 負樣本取樣開始
94     for (y=y_o; y<=y_o+posimg_ex.rows-1; y++)
95     {
96         for (x=x_o; x<=x_o+posimg_ex.cols-1; x++)
97         {
98             // 負樣本取樣
99             posnegimg.at<uchar>(y-y_o,x-x_o) = negimg.at<uchar>(y,x);
100         }
101     }
102     // 正樣本取樣開始
103     for (y=0; y<=posimg.rows-1; y++)
104     {
105         for (x=0; x<=posimg.cols-1; x++)
106         {
107             if (i==1) // 最左側正樣本取樣
108                 posnegimg.at<uchar>(y,x)
109                 = posimg_ex.at<uchar>(y,x);
110             else if (i==2) // 偏左側正樣本取樣
111                 posnegimg.at<uchar>(y,x+x_step)
112                 = posimg_ex.at<uchar>(y,x+x_step);
113             else if (i==3) // 正中央正樣本取樣
114                 posnegimg.at<uchar>(y,x+x_step*2)
115                 = posimg_ex.at<uchar>(y,x+x_step*2);
116             else if (i==4) // 偏右側正樣本取樣
117                 posnegimg.at<uchar>(y,x+posimg_ex_cols-posimg.cols-x_step)
118                 = posimg_ex.at<uchar>(y,x+posimg_ex_cols-posimg.cols-x_step);
119             else // 最右側正樣本取樣
120                 posnegimg.at<uchar>(y,x+posimg_ex_cols-posimg.cols)
121                 = posimg_ex.at<uchar>(y,x+posimg_ex_cols-posimg.cols);
122         }
123     }

```

Figure 3.9: 程式碼範例，正負樣本混合

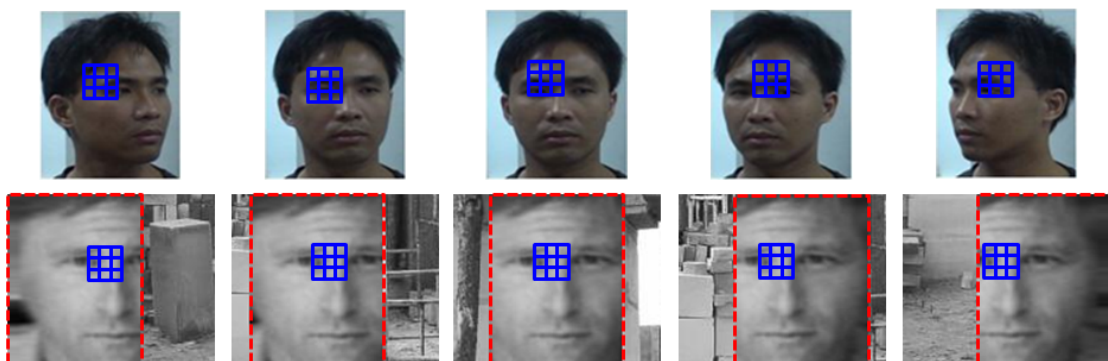


Figure 3.10: 正負樣本混合前後的特徵取樣情形, 取自 [42]

### 3.5 Training Results



	non-Expansion	Expansion	Expansion+
Positive	2000	10000	10000
Negative	1000	1000	1000
Stage	17	17	17
minHitRate	0.995	0.995	0.99995
Feature	80	96	159
Xml	28.9KB	34.3KB	59.9KB
Time	13'10"	14'22"	11'43"

Table 3.1: 訓練結果, LFW-a

	non-Expansion	Expansion	Expansion+
Positive	2000	10000	10000
Negative	1000	1000	1000
Stage	17	17	17
minHitRate	0.999	0.999	0.99995
Feature	85	104	115
Xml	32.8KB	39.0KB	45.2KB
Time	14'25"	23'21"	15'50"

Table 3.2: 訓練結果, MIT CBCL

Table 3.1 3.2 分別為 LFW-a 與 MIT CBCL 的訓練結果，各自有三種版本，分別為非延展版 (non-Expansion)、延展版 (Expansion) 與延展加強訓練版 (Expansion+)，非延展版為原始的樣本集直接訓練，延展版則將樣本集拉平展開混入負樣本後再進行訓練，至於延展加強訓練版的組態同延展版，只不過將每一級 (Stage) 的最小擊中率 (Minimal Hit Rate) 提升至 0.99995。由數據可看出即便延展版比非延展版多出了四倍的正樣本，但是由於正樣本間高度相關，所需的特徵 (Feature) 並未大量增加。訓練時間也都落在數十分鐘等級，並未出現異常費時的情況。

然而必須留意的是，雖然 [23] 有特別提及 OpenCV 串接式分類器裡的每一級強分類器其正負樣本的使用量皆相同，但不代表各級強分類器皆使用完全相同的正負樣本。依照 [22] 的原始理論定義，在設定 False Alarm Rate 為 50% 的情況下，每級強分類器丟棄負樣本的強度必須達到 50% 以上。參照 Figure 3.11 的說明，假設一組 20 級的串接式分類器要確保每級都有負樣本可供訓練、不會因為耗盡負樣本而被迫中斷訓練，那麼在其入口處的負樣本最少必須準備 1,048,576 張以上才

有機會完成訓練。儘管 OpenCV 為了減輕計算量不會真的動用到 1,048,576 張負樣本，但我們也可以確定 Table 3.1 3.2 所標示的負樣本 1000 張絕不會是整個訓練過程裡負樣本的最大使用量。這也是為何本章節一開始在”負樣本的建置原則”裡提及至少需準備數百張大面積的負樣本圖集，以確保 OpenCV 可以截取出足夠數量的負樣本供訓練之用。

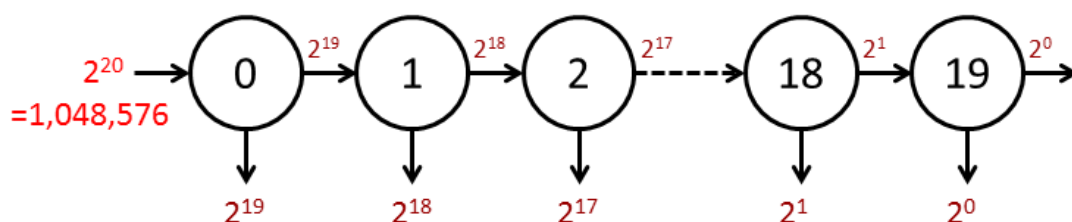


Figure 3.11: 串接式分類器負樣本衰減變化情形

Figure 3.12 3.13 分別是 LWF-a 及 MIT CBCL 延展版的訓練結果圖示說明，每一組圖分別代表各級強分類器所使用的特徵。觀察其內容可發現絕大多數的特徵皆落於中心區域，此現象跟當初預設負樣本區域會在訓練過程中被忽視的推測是相符的，至於實際的人臉檢測效果將於下一章做進一步的確認。



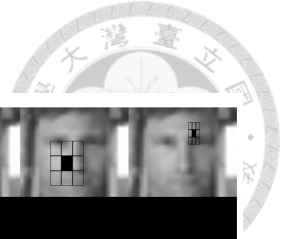


Figure 3.12: LFW-a 延展版訓練結果, LBP 實際分布情形, 使用 [36] 分析



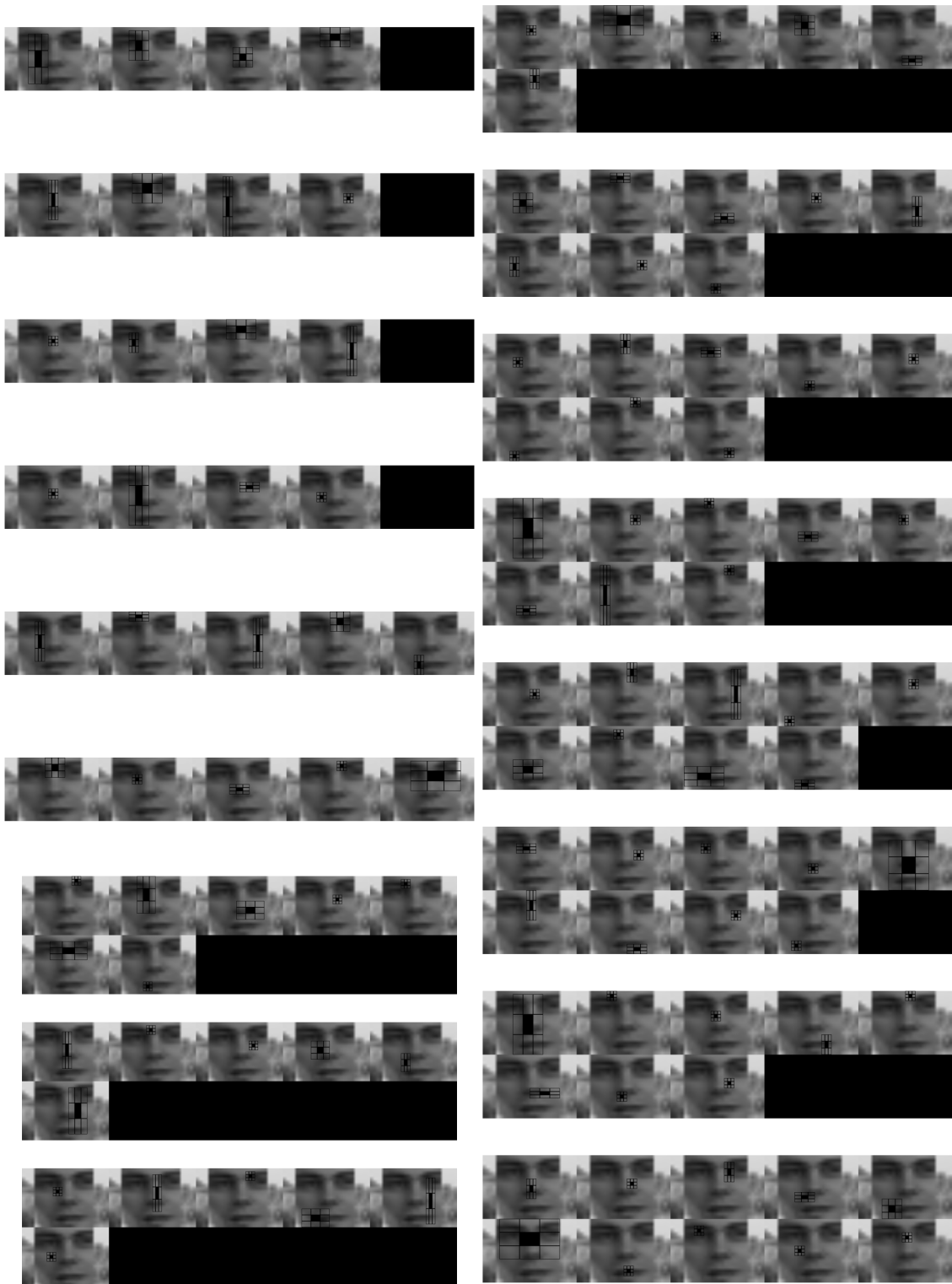
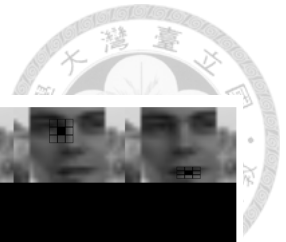


Figure 3.13: MIT CBCL 延展版訓練結果, LBP 實際分布情形, 使用 [36] 分析





# Chapter 4

## Experiment

### 4.1 Test Method

為了對訓練樣本延展前後檢測率的改善情形進行量化，我們必須儘可能蒐集多角度的人臉資料庫並整理成如 Figure 4.4 所示的測試圖集，此時善用 XnView 先前已提過的”批次轉換”功能與 Figure 4.1 所示的”建立縮圖表”功能可以大幅縮短處理的時間。

至於檢測的程式可以直接使用 OpenCV 所提供的 facedetect.cpp，指令如下所



Figure 4.1: XnView 建立縮圖表

示：

```
facetedetect.exe 縮圖表.jpg --cascade=cascade.xml --scale=0.4
```

其中”縮圖表.jpg”是由 XnView 合併多角度人臉資料庫所產生。cascade.xml 即為待檢測的串接式分類器 Xml 檔，scale=0.4 則表示將”縮圖表.jpg”縮小 0.4 倍後才開始測試(也就是放大 2.5 倍)，至於縮放的幅度則視”縮圖表.jpg”內的人臉大小做適當的調整。

順便一提的是，facetedetect.exe 預設並不允許將圖片放大檢測，也就是說 scale 小於 1 者皆屬無效設定。必須自行遮蔽 facetedetect.cpp 的第 56、57 行再重新編譯方可使用，如下列所示。

```
//if (scale < 1)  
  
// scale = 1;
```

## 4.2 Test Results

Figure 4.2 是 LFW-a 正樣本延展前後的測試比較結果，一如預期側臉的檢測率有明顯的提升，此外亦可觀察到連同正面人臉的檢測率也一併被強化。同時我們也收集了 Pointing'04 ICPR Workshop [32]、成功大學機器人實驗室 [33] 及 FEI Face Database [34] 所提供的多角度人臉資料庫、整理成 Figure 4.4 至 4.9 的測試圖表。由左至右從 -90 度到 +90 度的旋轉量分別由 13 或 11 張人臉所表示，並利用此一一系列的測試圖表來評估檢測率的提升情形。

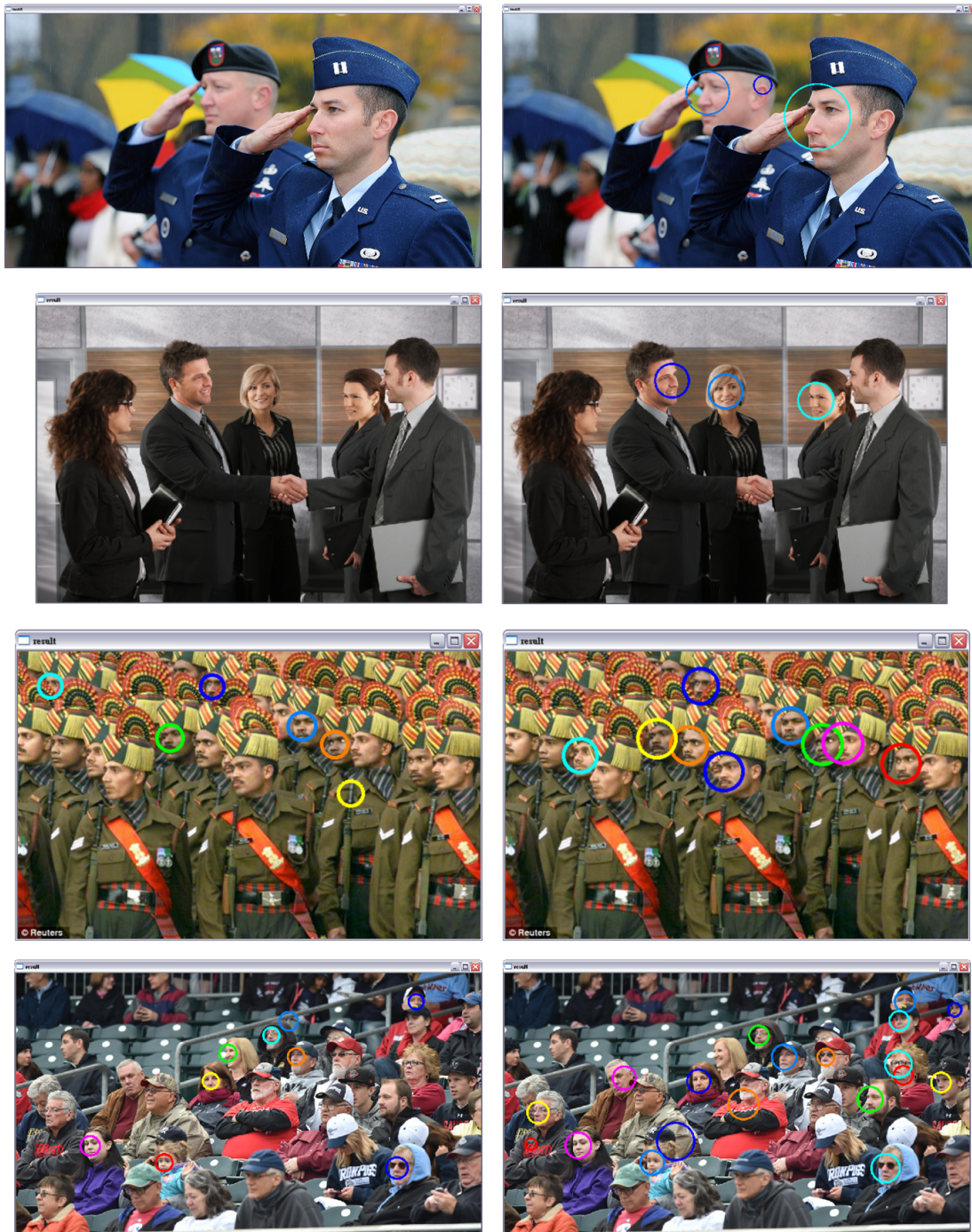


Figure 4.2: 測試結果, 取自 [43–46]

## 4.3 Analysis

根據 Figure 4.4 至 4.9 的測試圖表，我們將其整理成 Figure 4.10 至 4.15 的統計結果，其中 Hit Rate 的提升效果一併於圖表中以直條圖表示。根據測試結果我們發現：

1. 對角度變化的容忍度有所增加是明確的。
2. 原則上整體的檢測率是提升的。
3. 不同的訓練樣本會導致不同的測試結果。
4. 提升訓練時的 Minimal Hit Rate 原則上可提高測試的準確度。但本來測試結果就已不差者若再過度提升 Minimal Hit Rate 則可能會連帶提升 False Alarm Rate、反而帶來反效果導致誤判增多，也就是說這方面的調整可能必須伴隨更多的測試及參數調校。
5. 成功大學機器人實驗室的樣本測試結果不如其它兩個資料庫，尤以 MIT CBCL 特別明顯，藉由對 Figure 3.13 的觀察，我們發現絕大多數的 LBP 都圍繞著眼鼻口打轉，懷疑問題疑似起因於東方人鼻子眼睛的輪廓不如西方人明顯，才導致檢測率提升的效果受限，亦可參考 Figure 4.3 比較兩個樣本集的差異。



Figure 4.3: 人臉樣本對照圖, 成功大學機器人實驗室與 MIT CBCL



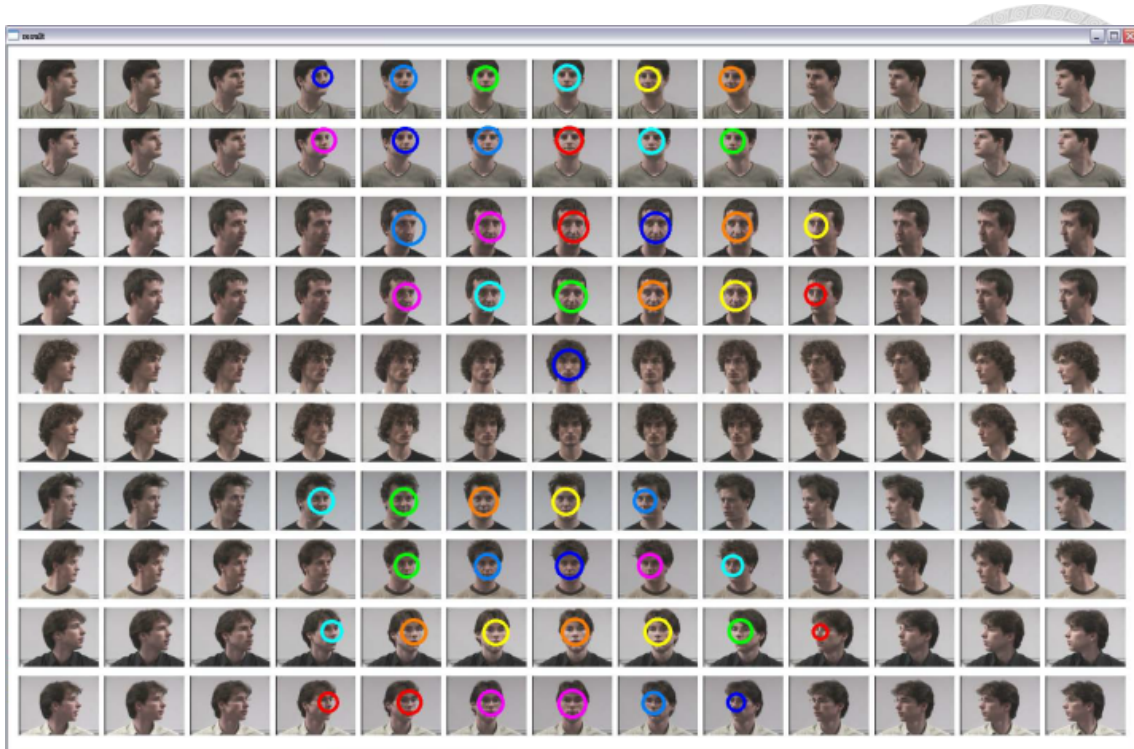


Figure 4.4: LFW-a 非延展測試, Pointing'04 ICPR Workshop



Figure 4.5: LFW-a 延展測試, Pointing'04 ICPR Workshop

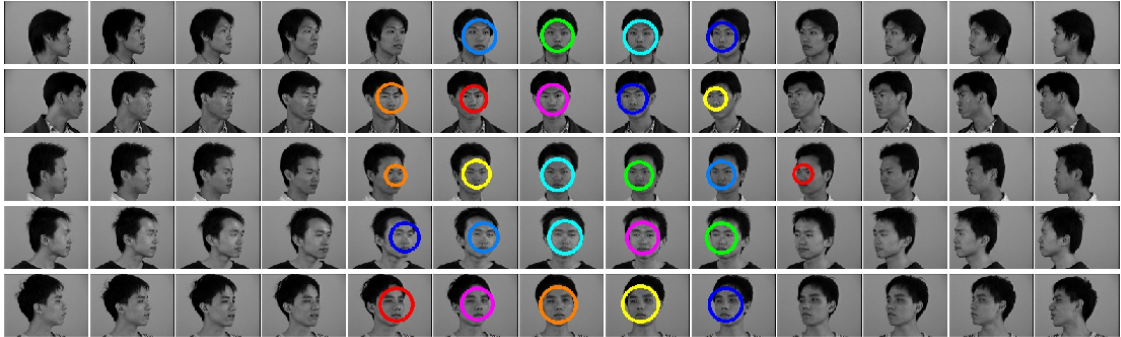


Figure 4.6: LFW-a 非延展測試, 成功大學機器人實驗室

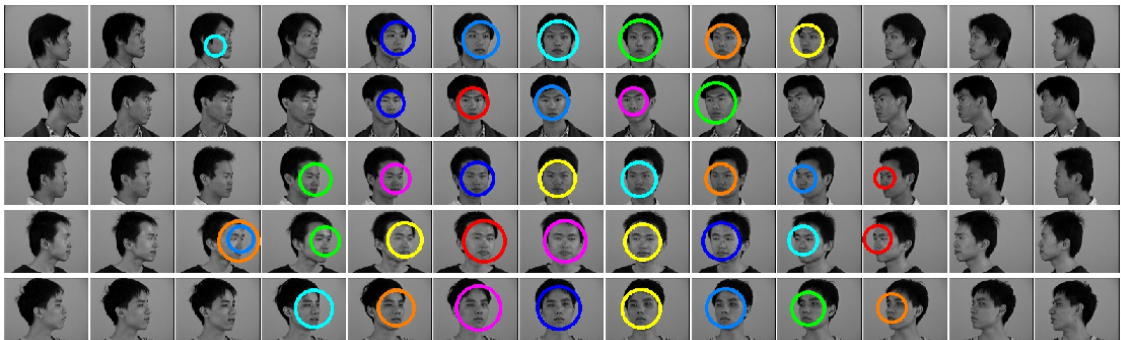


Figure 4.7: LFW-a 延展測試, 成功大學機器人實驗室





Figure 4.8: LFW-a 非延展測試, FEI Face Database



Figure 4.9: LFW-a 延展測試, FEI Face Database

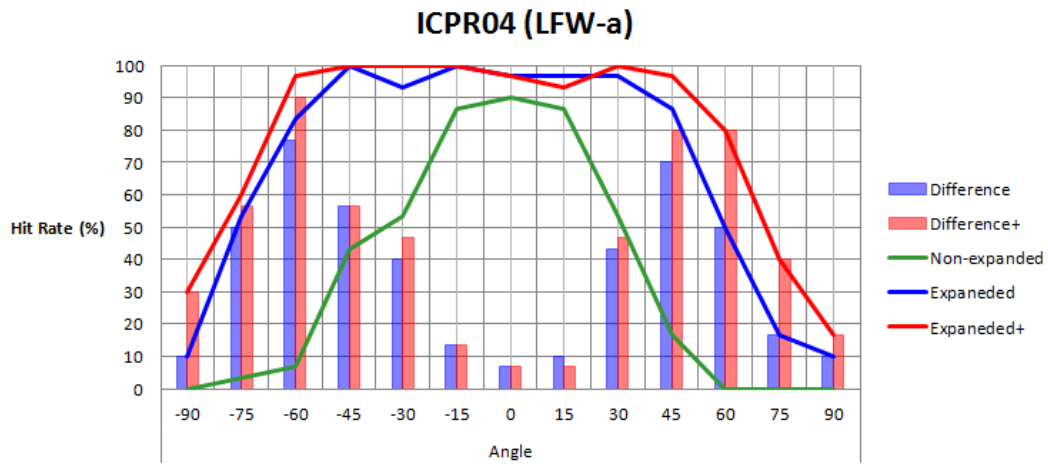


Figure 4.10: LFW-a 測試對照表, Pointing'04 ICPR Workshop

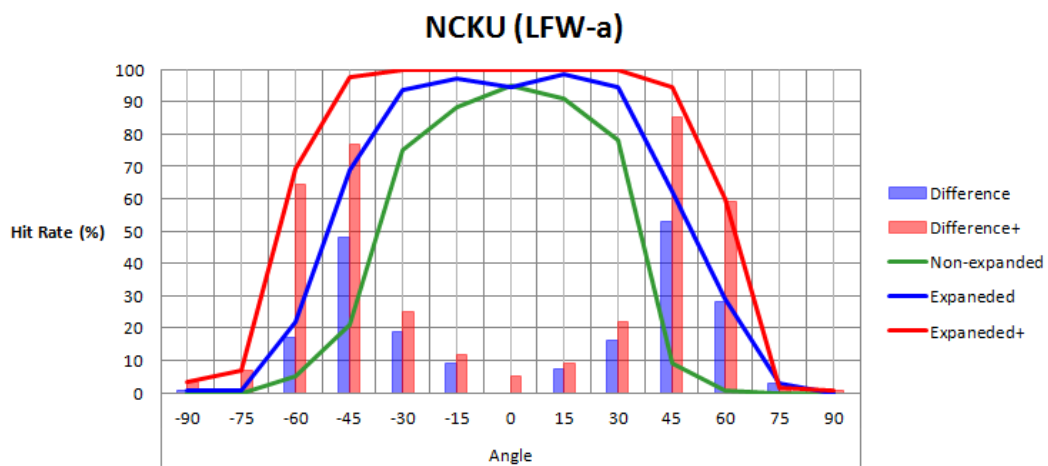


Figure 4.11: LFW-a 測試對照表, 成功大學機器人實驗室

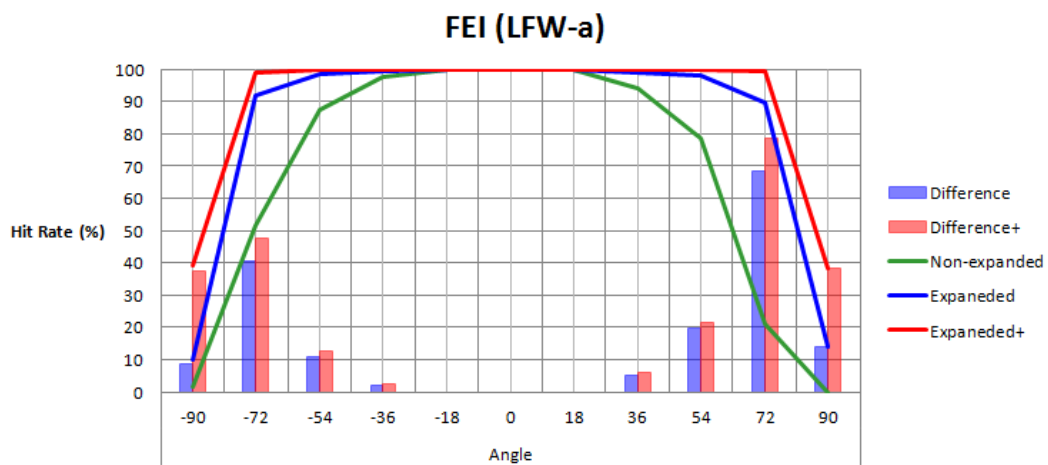


Figure 4.12: LFW-a 測試對照表, FEI Face Database

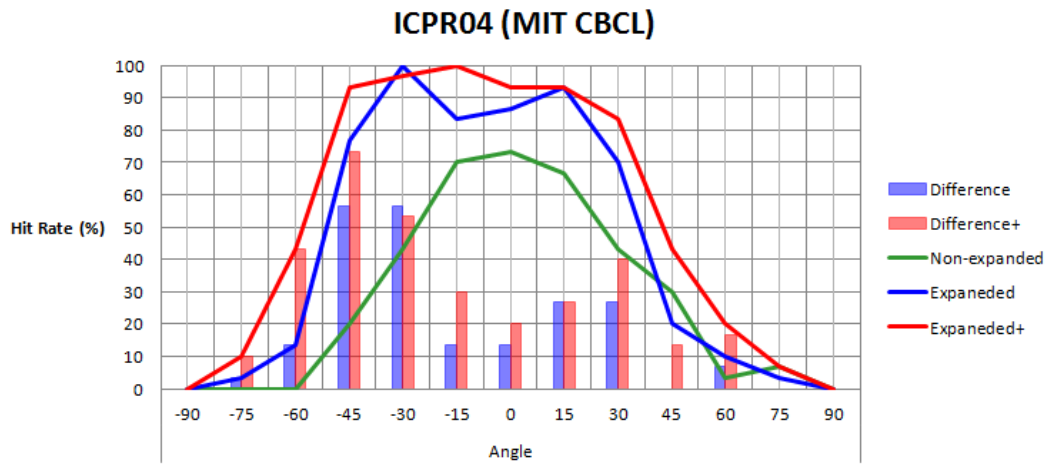


Figure 4.13: MIT CBCL 測試對照表, Pointing'04 ICPR Workshop

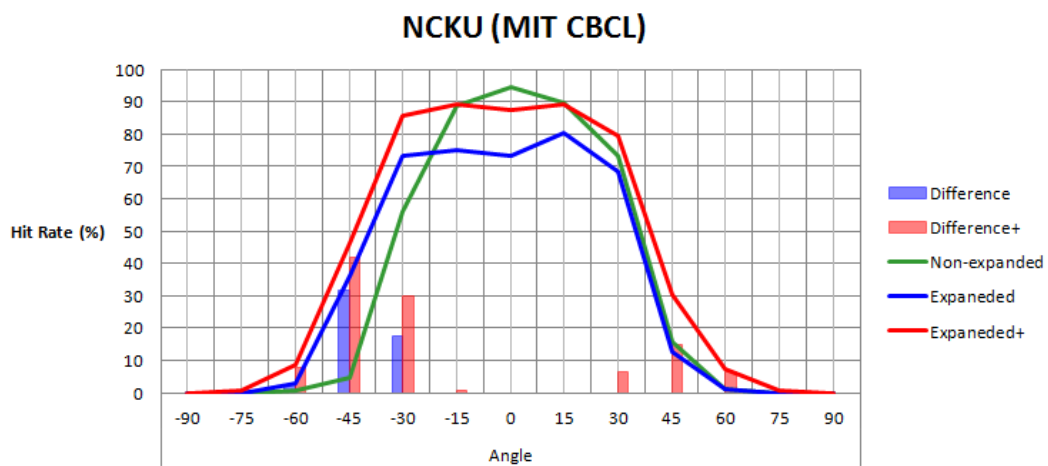


Figure 4.14: MIT CBCL 測試對照表, 成功大學機器人實驗室

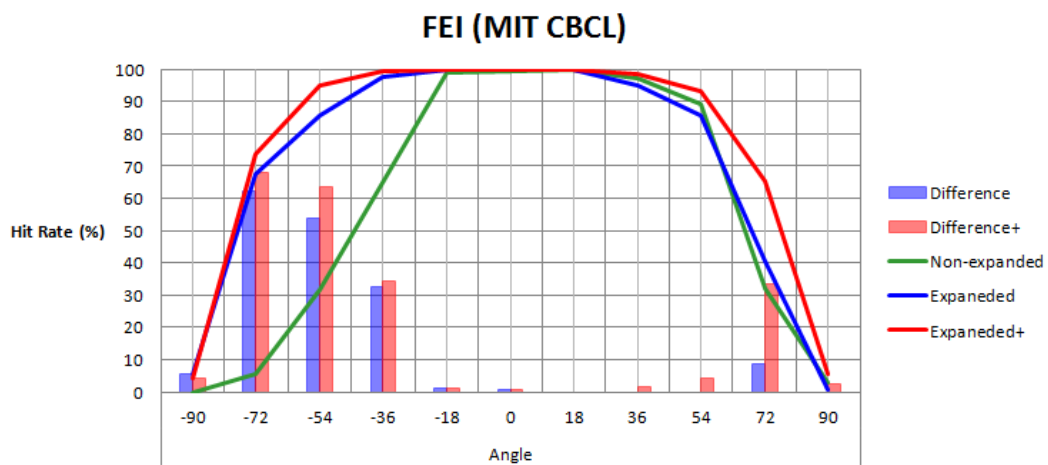


Figure 4.15: MIT CBCL 測試對照表, FEI Face Database





## Chapter 5

### Conclusion

根據以上章節的說明，我們在此簡要總結演算法的執行步驟：

1. 評估待測物的形狀、將訓練用正樣本展開圖面。
2. 評估旋轉情形、自展開圖面分段截取出新的正樣本，並於空白處隨機貼入負樣本(即正負樣本混合)。
3. 開始訓練。

如此可帶來的效益有：

1. 提升整體的檢測率。
2. 增加對角度變化的容忍度。
3. 未明顯增加訓練與檢測負擔。

在本次理論建立、實驗開發的過程中，我們亦發現了下列的延伸需求尚待更深入的研究去克服：

1. 樣本集蒐集的困難，為串接式分類器於實用化考量上的一大瓶頸。也就說如何從網路資源裡有效蒐集整理出我們需要的樣本是個值得深究的議題。
2. 本次的研究僅探討 1 維的非共平面變化，但理論上也可擴及 2 維的旋轉，然而人臉的變形會變得更為嚴重，更為困難的延展處理尚需更多的實驗驗證。


3. 依照推斷，因為東西方人臉輪廓差異造成的判斷弱化，除了找尋、建立東方人臉資料庫外，如何弱化輪廓差異造成的影響或是找出其它方法解決也是待開發的方向之一。

但也如前面章節所提及，供以 Adaboost 訓練的正樣本集至少要上千個以上才會有較佳的效果，在實驗過程中本欲找尋合適的多角度人臉正樣本集做為訓練效果的對照參考但未果(數量皆不足以供訓練之用)。換言之本篇論文在考量行動裝置計算能力受限的現實前題下所提交之演算法，在某種程度也解決了多角度人臉正樣本集不足的問題，且經測試證明效果明顯並具有實用性，相信未來可依此做法延伸出更多深入的應用。




## Bibliography

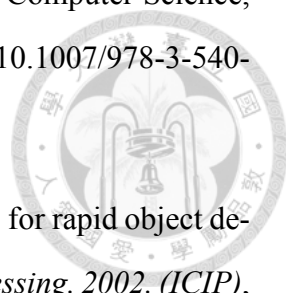
- [1] N. McCarroll, A. Belatreche, J. Harkin, and Y. Li, "Bio-inspired hierarchical framework for multi-view face detection and pose estimation," in *Proc. 2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2015.
- [2] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5325–5334, June 2015.
- [3] S. S. Farfade, M. Saberian, and L.-J. Li, "Multi-view Face Detection Using Deep Convolutional Neural Networks," *arXiv:1502.02766 [cs]*, Feb. 2015. arXiv: 1502.02766.
- [4] K. Kadir, M. K. Kamaruddin, H. Nasir, S. I. Safie, and Z. A. K. Bakti, "A comparative study between LBP and Haar-like features for Face Detection using OpenCV," in *Proc. 2014 4th International Conference on Engineering Technology and Entrepreneurship (ICE2T)*, pp. 335–339, Aug. 2014.
- [5] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," *arXiv:1407.4023 [cs]*, July 2014. arXiv: 1407.4023.
- [6] Z. Chai, Y. Zhang, Z. Du, D. Wang, and H. Méndez-Vázquez, "Learning Flexible Block based Local Binary Patterns for unconstrained face detection," in *Proc. 2014 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2014.

- 
- [7] J. R. Barr, K. W. Bowyer, and P. J. Flynn, “The effectiveness of face detection algorithms in unconstrained crowd scenes,” in *Proc. IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1020–1027, Mar. 2014.
- [8] D. Han, J. Kim, J. Ju, I. Lee, J. Cha, and J. Kim, “Efficient and fast multi-view face detection based on feature transformation,” in *Proc. 16th International Conference on Advanced Communication Technology (ICACT)*, pp. 682–686, Feb. 2014.
- [9] S. Liu, Y. Dong, W. Liu, and J. Zhao, “Multi-view face detection based on cascade classifier and skin color,” in *Proc. 2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems (CCIS)*, vol. 01, pp. 56–60, Oct. 2012.
- [10] K. Ma and J. Ben-Arie, “Vector array based Multi-View Face Detection with compound exemplars,” in *Proc. 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3186–3193, June 2012.
- [11] A. K. Makhtar, H. Yussof, H. Al-Assadi, L. C. Yee, Y. M. Mustafah, and A. W. Azman, “Out-of-Plane Rotated Object Detection using Patch Feature based Classifier,” in *Proc. International Symposium on Robotics and Intelligent Sensors 2012 (IRIS)*, vol. 41, pp. 170–174, Jan. 2012.
- [12] L. Xu, J. Li, and K. Wang, “Real-time and multi-view face tracking on mobile platform,” in *Proc. 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1485–1488, May 2011.
- [13] Y. Ying, H. Wang, and J. Xu, “An automatic system for multi-view face detection and pose estimation,” in *Proc. 2010 11th International Conference on Control Automation Robotics Vision (ICARCV)*, pp. 1101–1108, Dec. 2010.
- [14] Y. M. Mustafah, A. Bigdeli, A. W. Azman, F. Dadgostar, and B. C. Lovell, “Square Patch Feature: Faster weak-classifier for robust object detection,” in *Proc. 2010 11th International Conference on Control Automation Robotics Vision (ICARCV)*, pp. 2073–2077, Dec. 2010.



- 
- [15] Y. Li, S. Wang, and X. Ding, "Person-independent head pose estimation based on random forest regression," in *Proc. 2010 IEEE International Conference on Image Processing (ICIP)*, pp. 1521–1524, Sept. 2010.
- [16] B. Ma and T. Wang, "Head Pose Estimation Using Sparse Representation," in *Proc. 2010 Second International Conference on Computer Engineering and Applications (ICCEA)*, vol. 2, pp. 389–392, Mar. 2010.
- [17] C. Huang, H. Ai, Y. Li, and S. Lao, "High-Performance Rotation Invariant Multiview Face Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 671–686, Apr. 2007.
- [18] S. Liao, X. Zhu, Z. Lei, L. Zhang, and S. Z. Li, "Learning Multi-scale Block Local Binary Patterns for Face Recognition," in *Advances in Biometrics* (S.-W. Lee and S. Z. Li, eds.), no. 4642 in Lecture Notes in Computer Science, pp. 828–837, Springer Berlin Heidelberg, 2007.
- [19] R. Niese, A. Al-Hamadi, and B. Michaelis, "A Stereo and Color-based Method for Face Pose Estimation and Facial Feature Extraction," in *Proc. 18th International Conference on Pattern Recognition (ICPR)*, vol. 1, pp. 299–302, 2006.
- [20] C. Huang, H. Ai, Y. Li, and S. Lao, "Vector boosting for rotation invariant multi-view face detection," in *Proc. Tenth IEEE International Conference on Computer Vision, 2005. (ICCV)*, vol. 1, pp. 446–453 Vol. 1, Oct. 2005.
- [21] B. Wu, H. Ai, C. Huang, and S. Lao, "Fast rotation invariant multi-view face detection based on real Adaboost," in *Proc. Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. (FG)*, pp. 79–84, May 2004.
- [22] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *Int. J. Comput. Vision*, vol. 57, pp. 137–154, May 2004.
- [23] R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection," in *Pattern Recognition*

(B. Michaelis and G. Krell, eds.), no. 2781 in Lecture Notes in Computer Science, pp. 297–304, Springer Berlin Heidelberg, Sept. 2003. DOI: 10.1007/978-3-540-45243-0\_39.

- 
- [24] R. Lienhart and J. Maydt, “An extended set of Haar-like features for rapid object detection,” in *Proc. 2002 International Conference on Image Processing. 2002. (ICIP)*, vol. 1, pp. I–900–I–903 vol.1, 2002.
- [25] H. Schneiderman and T. Kanade, “A statistical method for 3d object detection applied to faces and cars,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2000. (CVPR)*, vol. 1, pp. 746–751 vol.1, 2000.
- [26] “OpenCV.” available: <http://opencv.org/>.
- [27] “Labeled Faces in the Wild-a.” available: <http://www.openu.ac.il/home/hassner/data/lfwa/>.
- [28] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” Tech. Rep. 07-49, University of Massachusetts, Amherst, October 2007.
- [29] L. Wolf, T. Hassner, and Y. Taigman, “Effective Unconstrained Face Recognition by Combining Multiple Descriptors and Learned Background Statistics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 1978–1990, Oct. 2011.
- [30] “MIT CBCL Face Data.” available: <http://cbcl.mit.edu/software-datasets/FaceData2.html>.
- [31] “Google Code Archive.” available: <https://code.google.com/archive/p/tutorial-haartraining/source/default/source>.
- [32] “Pointing’04 | Face pointing.” available: <http://www-prima.inrialpes.fr/Pointing04/data-face.html>.

- [33] “Databases for Face Detection.” available: [http://robotics.csie.ncku.edu.tw/Databases/FaceDetect\\_PoseEstimate.htm#Our\\_Database\\_](http://robotics.csie.ncku.edu.tw/Databases/FaceDetect_PoseEstimate.htm#Our_Database_).
- [34] “Carlos Eduardo Thomaz - Personal Web Page.” available: <http://fei.edu.br/~cet/facedatabase.html>.
- [35] J. Howse, S. Puttemans, Q. Hua, and U. Sinha, *OpenCV Blueprints*. Packt Publishing - ebooks Account, Nov. 2015.
- [36] available: [https://github.com/OpenCVBlueprints/OpenCVBlueprints/tree/master/chapter\\_5/source\\_code/visualize\\_models/](https://github.com/OpenCVBlueprints/OpenCVBlueprints/tree/master/chapter_5/source_code/visualize_models/).
- [37] “XnView Software.” available: <http://www.xnview.com/en/>.
- [38] “05-07-1970-04-march-down-court-street-in-athens-ohio-6.jpg.” available: <http://www.capecentralhigh.com/>.
- [39] “Histograms of Oriented Gradients.” available: <https://github.com/pavitrakumar78/Python-Custom-Digit-Recognition>.
- [40] “svmhyperplane.png.” available: <http://www.mathworks.com/help/stats/svmhyperplane.png>.
- [41] “tikz11.png.” available: <http://neuralnetworksanddeeplearning.com/images/tikz11.png>.
- [42] “input.JPG.” available: [http://robotics.csie.ncku.edu.tw/Students/PHD/93/WuJinYi/Web\\_Pic/3DFaceRec/input.JPG](http://robotics.csie.ncku.edu.tw/Students/PHD/93/WuJinYi/Web_Pic/3DFaceRec/input.JPG).
- [43] “Military3.jpg.” available: <http://du1ux2871uqvu.cloudfront.net/sites/default/files/article/Military3.jpg>.
- [44] “Public-Relations.jpg.” available: <http://selworthy.com/wp-content/uploads/2016/04/Public-Relations.jpg>.



[45] “article-2260148-00249ee100000258-886\_634x393.jpg.” available: [http://i.dailymail.co.uk/i/pix/2013/01/10/article-2260148-00249EE100000258-886\\_634x393.jpg](http://i.dailymail.co.uk/i/pix/2013/01/10/article-2260148-00249EE100000258-886_634x393.jpg).

[46] “mc-pictures-faces-in-the-crowd-ironpigs-vs-bis-006.” available: <http://www.trbimg.com/img-57254967/turbine/mc-pictures-faces-in-the-crowd-ironpigs-vs-bis-006/1050/1050x591>.

