

國立交通大學

資訊科學與工程研究所

碩士論文

非監督式主播影像偵測於新聞故事分段之研究

The Study of Unsupervised Anchorperson Detection for  
News Story Segmentation

研究生：吳玉善

指導教授：傅心家 教授

中華民國九十六年七月

非監督式主播影像偵測於新聞故事分段之研究  
The Study of Unsupervised Anchorperson Image  
Detection for News Story Segmentation

研究生：吳玉善

Student：Yu-Shan Wu

指導教授：傅心家

Advisor：Prof. Hsin-Chia Fu

國立交通大學  
資訊科學與工程研究所  
碩士論文

A Thesis

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Computer Science

July 2007

Hsinchu, Taiwan, Republic of China

中華民國九十六年七月

# 非監督式主播影像偵測於新聞故事分段之研究

研究生:吳玉善

指導教授:傅心家 教授

國立交通大學資訊科學與工程研究所

## 摘要

建立一個全自動的新聞故事分段系統是一個重要且富挑戰性的工作。一則新聞故事段落是由主播播報與外景採訪所組成，所以若能夠知道主播出現的時段就能將一個連續的新聞影片自動分段。本篇論文提出一個以偵測人臉為基礎的主播偵測方式。首先偵測每張新聞影像的人臉區域，利用人臉區域取出特徵，以此特徵做分群，假設最大群是主播，進而篩選出主播人臉區域。因為只注意影像中有人臉的區域，不會受到複雜背景的影響，沒有偵測鏡頭轉換的問題；還有在判斷那張影像有主播的過程並沒有使用主播影像模型，所以不需要為了每個主播去調整模型。我們以 5 個小時不同電視台的整點新聞影片進行主播影像偵測與新聞故事切割的實驗，驗證所提的方法能正確的找出主播出現的時段。論文最後更進一步將所提的方法整合至一個已有的新聞系統中並成功的應用在東森晚間新聞故事的切割上。

# The Study of Unsupervised Anchorperson Image Detection for News Story Segmentation

Student: Yu-Shan Wu

Advisor: Prof. Hsin-Chia Fu

Institute of Computer Science and Engineering  
National Chiao Tung University

## Abstract

Building an automatic system for news story segmentation is an important and challenging task. A news story is composed of an anchorperson shot and a news footage shot, we can segment a news video into several stories if we know when the anchorperson shows up. This paper presents a method for anchorperson detection based on face detection. First, detecting human faces region in every news frame. Then, extracting features by the face region, and clustering on all features. Suppose that the biggest cluster is presented for anchorperson. This method would not be effected by the complex background because it focuses only on the face region. And because of its unsupervised nature, the algorithm does not need to adjust model for different anchorpersons. The efficacy of the proposed method is tested on 5 h of news programs. Moreover, we integrate the proposed method to an existed news video library system and segmenting on the ETT news programs successfully.

## 誌謝

謝謝傅老師在我研究所兩年的生涯給予我的指導和照顧，並幫助我的論文找到研究方向，學習到做研究的方法與態度，才得以完成此篇論文。同時，感謝實驗室博士後研究以及博士班學長，永煜、柏伸、政龍、岳宏、士賢，還有學弟逸凡、昭翰以及學妹佳蓁平常在生活上及學業上的建議與指教。特別感謝柏伸學長在論文上的極大幫助，讓我認識影像方面的知識也幫助我解決困難並修改論文，讓論文更為完美。感謝大學同學以及朋友在生活上的鼓勵。最後，感謝爸爸、媽媽、姊姊、妹妹一直在背後支持我，給我無憂無慮的生活，讓我可以專注在學業上，才得以順利完成學業。



# 目錄

|                           |     |
|---------------------------|-----|
| 摘要.....                   | i   |
| Abstract.....             | ii  |
| 誌謝.....                   | iii |
| 目錄.....                   | iv  |
| 表目錄.....                  | vi  |
| 圖目錄.....                  | vii |
| 第一章 前言.....               | 1   |
| 1.1 研究動機.....             | 1   |
| 1.2 研究目標.....             | 1   |
| 1.3 研究方向.....             | 2   |
| 1.4 章節介紹.....             | 3   |
| 第二章 人臉偵測之相關研究.....        | 4   |
| 2.1 由上而下的以知識為基礎的方法.....   | 4   |
| 2.2 由下而上的以特徵為基礎的方法.....   | 5   |
| 2.3 樣板比對方法.....           | 8   |
| 2.4 以表面跡象為基礎的方法.....      | 9   |
| 第三章 主播影像偵測與新聞故事切割.....    | 13  |
| 3.1 人臉區塊偵測與追蹤.....        | 15  |
| 3.1.1 人臉偵測.....           | 16  |
| 3.1.2 以人臉追蹤來加速人臉偵測.....   | 23  |
| 3.2 主播人臉區塊篩選.....         | 26  |
| 3.2.1 根據人臉區塊的特性抽取特徵.....  | 26  |
| 3.2.2 基於最大群原則的主播區域篩選..... | 27  |

|                          |    |
|--------------------------|----|
| 3.3 新聞故事切割.....          | 28 |
| 第四章 實驗結果.....            | 29 |
| 4.1 主播影像偵測實驗與結果分析.....   | 29 |
| 4.1.1 人臉偵測實驗與結果分析.....   | 30 |
| 4.1.2 分群實驗與結果分析.....     | 32 |
| 4.2 新聞故事切割實驗與結果分析.....   | 38 |
| 4.3 新聞故事切割系統應用與結果分析..... | 40 |
| 第五章 結論與未來展望.....         | 43 |
| 5.1 結論.....              | 43 |
| 5.2 未來展望.....            | 43 |
| 參考文獻.....                | 45 |



## 表目錄

|                                      |    |
|--------------------------------------|----|
| 表 4-1: 人臉偵測結果(沒有用人臉追蹤).....          | 30 |
| 表 4-2: 主播人臉誤失結果(沒有用人臉追蹤).....        | 31 |
| 表 4-3: 人臉偵測結果(有用人臉追蹤).....           | 31 |
| 表 4-4: 主播人臉誤失結果(有用人臉追蹤).....         | 31 |
| 表 4-5: 有無人臉追蹤的人臉偵測效能比較.....          | 32 |
| 表 4-6: 分群實驗的結果並列.....                | 38 |
| 表 4-7: 主播段偵測的實驗結果.....               | 39 |
| 表 4-8: 主播段偵測的實驗結果(考慮各種評量標準).....     | 40 |
| 表 4-9: 新聞主播影像偵測實驗結果.....             | 41 |
| 表 4-10: 東森新聞主播段偵測實驗結果.....           | 42 |
| 表 4-11: 東森新聞主播段偵測實驗結果(考慮各種評量標準)..... | 42 |



## 圖目錄

|                                |    |
|--------------------------------|----|
| 圖 1-1: 電視新聞節目的結構.....          | 2  |
| 圖 2-1: 人臉樣板示意圖.....            | 9  |
| 圖 3-1: 主播影像偵測與新聞故事切割流程.....    | 14 |
| 圖 3-2: 人臉偵測與追蹤流程.....          | 15 |
| 圖 3-3: 人臉偵測流程.....             | 16 |
| 圖 3-4: 區域成長演算法範例.....          | 19 |
| 圖 3-5: 膚色區域偵測.....             | 20 |
| 圖 3-6: 在候選人臉區域中偵測眼睛.....       | 21 |
| 圖 3-7: 由眼睛配對所產生的人臉驗證候選區域.....  | 22 |
| 圖 3-8: 眼睛區塊比對的人臉追蹤演算法.....     | 24 |
| 圖 3-9: 驗證人臉區域的人臉追蹤演算法.....     | 25 |
| 圖 3-10: 估計衣服區域.....            | 26 |
| 圖 3-11: 新聞故事架構.....            | 28 |
| 圖 4-1: 東森 1 小時衣服資料的分群實驗結果..... | 33 |
| 圖 4-2: 中天 1 小時衣服資料的分群實驗結果..... | 34 |
| 圖 4-3: 民視 1 小時衣服資料的分群實驗結果..... | 35 |
| 圖 4-4: 華視 1 小時衣服資料的分群實驗結果..... | 36 |
| 圖 4-5: 三立 1 小時衣服資料的分群實驗結果..... | 37 |

# 第 1 章

## 前言

### 1.1 研究動機

一般我們在觀看電視新聞時，通常是隨意的瀏覽各電視新聞台，不知道會看到怎樣的新聞，大部分的新聞都是自己沒興趣的，且因為電視台新聞是連續播放，無法將其快轉，往往要等個半小時或一小時才能看到想看的新聞。如果有一個全自動的新聞系統來整理分析新聞影片，我們就能從中搜尋感興趣的新聞。而自動化新聞系統的核心是新聞故事切割元件，所以我們想發展一個效能不錯的新聞故事切割系統。

### 1.2 研究目標

一則新聞故事通常是由主播播報與外景採訪所組成，如圖 1-1。有了主播出現的時間，我們就能將新聞影片作分段。因此本論文的研究目標是要在一小時的整點新聞影片中，找到主播出現的時間。

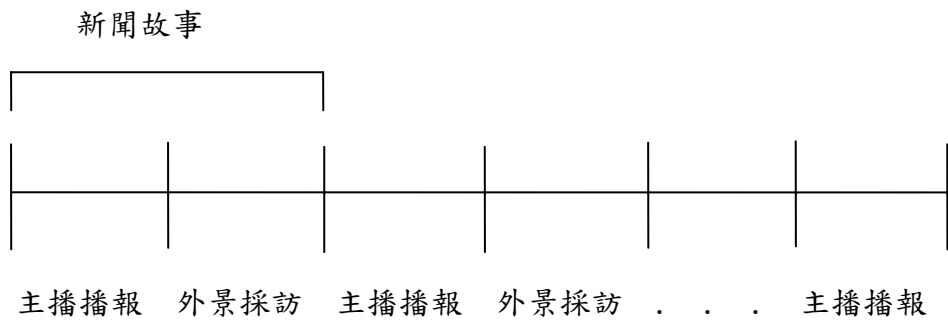


圖 1-1 電視新聞節目的結構

### 1.3 研究方向

偵測新聞主播的方式有許多種，Gao 等人【1】利用影像來偵測主播，首先偵測新聞影片中鏡頭轉換的時間點將影片分成許多個鏡頭，判斷這些鏡頭中那些是屬於主播播報，那些是屬於外景採訪。判斷的方法是從每個鏡頭中各取一張影像當代表，假設有主播的影像有共同的特徵—相同且乾淨的背景、明顯的主播人形，根據這些特徵來判斷那些影像有主播，進而確定該影像所代表的鏡頭是否屬於主播播報。這個方法在當時提出有很好的效果，但目前的電視新聞主播在播報新聞時，背景會播放該新聞的影片，所以此方法用在目前的新聞影片會出問題，不只會把同一個主播鏡頭分成好幾個鏡頭，更嚴重的是在之後的鏡頭判斷中會出錯誤，因為原本的假設條件（主播鏡頭都有相同且乾淨的背景）已經不符合了。

除了影像之外，鄭【2】利用聲音來偵測主播。首先在一段連續的新聞語音中偵測語者交換點，根據語者交換點把新聞語音分成好幾個語音段，假設每個語音段只有一個語者。對這些語音段做分群，假設主播的語音段是最多的，因此最大的那一群就是主播。這個方法在主播音段沒有背景音樂時會成功，但目前的新聞台主播在播報新聞時通常有背景音樂，所以也會出問題。

基於上述的原因，本論文的研究方向在提出一個更可靠的主播偵測方式，仍然是以影像為基礎，但不偵測鏡頭轉換，而是偵測影像中的人臉區域，所以不會

受到複雜背景的影響；且在判斷那個人是主播的作法是先對每張影像偵測人臉，取出每個人的特徵，利用此特徵做分群，假設最大群是主播，在這過程中沒有使用主播影像模型，不需為了不同的主播調整模型；還有是以影像為基礎，自然沒有背景音樂的問題。

## 1.4 章節介紹

在以下章節中，第二章首先介紹人臉偵測與其相關研究，從中選擇一個適當的人臉偵測演算法；第三章介紹本論文如何以人臉偵測為基礎來偵測主播影像，同時介紹如何利用人臉追蹤來加速人臉偵測的時間，還有在偵測主播影像之後如何做新聞故事切割；第四章是第三章所提出的新聞主播影像偵測與新聞故事切割方法的實驗結果，以驗證其可行性與效能；第五章是結論及對未來的展望。



## 第 2 章

# 人臉偵測之相關研究

在一張影像中偵測人臉是一個被研究多年的題目，已有許多種方法被提出。Yang 等人【3】將這些方法分成四類。



### 2.1 由上而下的以知識為基礎的方法 (Knowledge-Based Top-Down Method)

以知識為基礎的方法建立在研究者對人臉的認識，一個人臉通常有兩個對稱的眼睛，一個鼻子跟一個嘴巴等特徵，這些特徵間有相對的位置跟距離，可當成人臉驗證的規則，例如眼睛、嘴巴還有鼻子排列成一個T字形。在影像中找出這些人臉特徵，驗證特徵間的位置與距離是否符合已經定義好的規則就能確定找到的特徵是否組成一張人臉，同時也知道其所在。

以知識為基礎的方法困難的地方在於如何決定驗證人臉特徵的規則，如果規則訂的太嚴格，則會有許多人臉偵測不到，但如果訂的太鬆，則許多不是人臉的物件會被判斷成人臉。而且還有一個困難在於規則的決定主要是針對正面的人臉，那麼對於低頭及側面的人臉就會不適用。然而以知識為基礎的方法對於正面

且周圍沒有複雜背景的人臉是偵測得到的。

Kotropoulos 和 Pitas 【4】提出了一個以知識為基礎的人臉偵測方法。假設一張  $m \times n$  的影像在位置  $(x, y)$  的灰階值為  $I(x, y)$ ，定義水平投影為  $HI(x) = \sum_y I(x, y)$ ，垂直投影為  $VI(y) = \sum_x I(x, y)$ 。在水平投影中若有劇烈變化的兩個位置可能是頭的左右兩側，在垂直投影中若有劇烈變化的位置可能是眼睛與嘴巴。利用水平投影與垂直投影找出臉部特徵的位置後，再去判斷這些臉部特徵的相對位置與距離是否符合事先定義好的規則，若符合則認為偵測到人臉。

## 2.2 由下而上的以特徵為基礎的方法 (Feature-Based Bottom-Up Method)

人臉通常不是正面，有可能有不同的姿勢，例如側面，低頭等，還有不同影像的光強度可能不同，這些情況有可能讓以知識為基礎的方法會偵測不到人臉。所以研究者想尋找更穩定的特徵來偵測人臉，例如膚色或眼睛、鼻子、嘴巴、睫毛等，可用邊緣偵測器(edge detector)或型態上(morphology)的方法來偵測，然後再利用預先建立好的統計模型來驗證找到的特徵是否組成一個人臉。以特徵為基礎的方法所遭遇的困難是如果所選的特徵在影像中被雜訊、光照影響或其他物件遮蔽使得這些特徵沒辦法被找到，那麼影像中某些是人臉的區域就會偵測不到。至於臉部特徵的選擇有幾種，以下分別介紹：

### (1) 膚色(Skin Color)

膚色已經被證實可當成是偵測與追蹤人臉的有效特徵，藉由判斷影像中那些地方可能是膚色可縮小人臉偵測的範圍，因為只要在是膚色的區域中去驗證是否有人臉的其他特徵，像是眼睛、鼻子、嘴巴等就可以偵測到人臉，不過只有彩色影像才能用膚色這個特徵。雖然不同人種的膚色看起來差異極大，但有許多的研究顯示這些差異主要來自亮度而非色度 【5】。很多顏色空間都能用來表示膚色，像是 RGB 【6】、YCbCr 【7】、HSV 【8】、YIQ 【9】、CIE LUV 【10】。



選好代表膚色的顏色空間後，接下來就是要建立膚色模型，最簡單的膚色模型就是直接規定顏色範圍【11】，例如當選用的顏色空間為 YCbCr，統計收集的膚色像素來規定 Cb 的範圍是  $[Cb_1, Cb_2]$ ，Cr 的範圍是  $[Cr_1, Cr_2]$ 。如果一個測試像素的 Cb 與 Cr 值都落在這兩個範圍內，則認為此像素的顏色是膚色。

Crowley 和 Coutaz【12】用長方條代表的統計圖 (histogram) 來建立膚色模型，令  $h(r, g)$  代表在 RGB 顏色空間中收集的膚色像素的 RG 值等於  $(r, g)$  的長方條高度，同時規定長方條高度的門檻值  $\tau$ 。若測試像素的 RG 值為  $(r_1, g_1)$  且  $h(r_1, g_1) > \tau$ ，則認為此測試像素的顏色是膚色。

除了非參數的方法外，單一高斯模型【13】與高斯混合模型【6】也有人用來建立膚色模型。單一高斯模型的參數只有一個平均值與一個共變異數矩陣，這兩個參數可由收集的膚色像素計算得到，至於高斯混合模型的參數有高斯元件的個數、每個高斯元件的平均值與共變異數矩陣與權重，這些參數可用 EM 演算法來計算得到。至於用長方條代表的統計圖 (histogram) 或用高斯混合模型來建立膚色模型比較好沒有一定的標準答案，陳等人【14】則認為當收集的膚色像素比較少且具代表性時，用高斯混合模型來建立膚色模型較好；當收集的膚色像素很多且在顏色空間的範圍很廣時，用長方條代表的統計圖來建立膚色模型較好；Jones 等人【15】則認為用長方條代表的統計圖 (histogram) 來建立膚色模型較好，並有佐以實驗證明。

另外，Chow 等人【16】則認為一個像素一個像素的去判斷是不是膚色並不適當，因為眼睛、眉毛、頭髮等不是膚色的地方加上臉部有些小地方可能不會被判斷成膚色，造成判斷出的人臉區域會是破碎分離的，這時還要再用形態上的運算 (morphological operation) 去做處理，把破碎的人臉區域合併成一個完整的人臉區域，但是形態上的運算有其缺點，因為若破碎的人臉區域分隔的有點遠，則無法被合併，又或是有些小區域根本不屬於人臉區域也被合併到人臉區域。所以 Chow 等人提出了一個以區域為主 (Region-Based) 的膚色判斷方法，首先把測試影像做降取樣 (downsampling)，例如從  $640 \times 480$  的大小降取樣成  $80 \times 60$ ，然

後以 Comaniciu 等人 【17】提出的均值移動分段 (Mean Shift Segmentation) 把影像分割成好幾個顏色相同的區域，然後再做升取樣(upsampling)回  $640 \times 480$  的大小，降取樣(downsampling)的目的除了加速影像分段(Image Segmentation)外，最重要的作用是把眼睛、鼻子以及嘴巴區域都併入膚色，因為這些臉部特徵在降取樣的影像中會變的很小，以至於在影像分段(Image Segmentation)時會被合併到與臉部其他部分屬於同一個分段(區域)。將測試影像做完分段後，因為每個分段(區域)有相同的顏色，所以測試影像中同個分段可同時用膚色模型去判斷是否為膚色。

## (2) 臉部特徵(Facial Feature)

Chetverikov 和 Lerch 【18】提出一個簡單的人臉偵測方法，他們認為人臉可用兩團暗點代表眼睛，三團亮點代表兩頰與鼻子，並且可用條紋來代表人臉、眉毛和嘴唇的輪廓，所謂的條紋是由相近的邊界(edge)所組成。首先降低影像的解析度並通過拉普拉斯濾波器(Laplacian Filter)，接下來偵測影像中所有聚集成一團的亮點或暗點，然後取任意兩團暗點與三團亮點，判斷兩個條件，第一個條件是這兩團暗點是否與其中一團亮點形成一個三角形，第二個條件是這三團亮點是否形成另一個三角形且這兩個三角形同方向；如果兩個條件都成立的話，最後再驗證三件事，第一件事是三團亮點中是否只有一團亮點被條紋所包圍，第二件事是兩團暗點的上方是否都有細長狀的條紋，第三件事是是否有一個橢圓形的條紋包圍這五團亮點與暗點，若全都符合的話則認為這兩團暗點與三團亮點形成人臉。

Han 等人 【19】認為人臉中最不易被其他因素影響而可靠的特徵是眼睛，因為眼睛部份的灰階值比其附近的灰階值低，所以影像中在包含眼睛與其附近的區域的灰階值形成一個山谷，Han 等人利用形態上的運算(morphological operation)，包括關閉(closing)、相減(clipped difference)、門檻值(thresholding)等取出可能是眼睛的像素，然後把相鄰的眼睛像素連接形成眼睛部份，將所有的眼睛部份兩個兩個配對，在配對的同時可估計人臉的範圍區域，



最後用一個訓練好的類神經網路去驗證此範圍區域是否為人臉。

## 2.3 樣板比對方法(Template Matching)

樣板比對方法使用一個事先準備好的人臉樣板，人臉樣板的取得是以人工的方式從灰階影像中剪下人臉的區域(通常是正面的人臉)，有其固定的大小。利用這人臉樣板與測試影像中可能是人臉的區域做比對，計算相關值，若相關值越高，則越有可能是人臉，反之，則不是。樣板比對方法的優點在於實作簡單，不過當測試影像中人臉的姿勢、大小、形狀與樣板不同時就有可能偵測不到。

Sakai 等人【20】使用樣板比對的方法偵測人臉，他把人臉樣板分成眼睛、鼻子、嘴巴、人臉輪廓四個樣板，其中人臉輪廓樣板主要是由線段組成的人臉輪廓。首先在影像中尋找邊界(edge)，邊界就是灰階值有明顯變化的地方，接下來用人臉輪廓樣板去比對所有找出的邊界，如果相關值很高，則該線段圍成的區域很有可能是人臉，進一步的用眼睛、鼻子與嘴巴樣板去比對該區域中可能是眼睛、鼻子與嘴巴的位置，如果計算出的相關值都很高，則認為該區域的確是人臉。

Sinha【21】認為不應該直接使用灰階影像中的人臉當成樣板，因為這樣當測試影像中的人臉被光照改變亮度時人臉樣板就不能用了。他把樣板分成好幾個相鄰的區域，只定義相鄰區域的灰階值關係，如圖 2-1。圖 2-1 把人臉分成 16 個區域，箭頭方向代表相鄰區域的明暗關係，被箭頭指到的區域表示該區域較暗。當影像中的人臉被光照改變亮度時，使用此種新定義的人臉樣板仍能偵測到影像中的人臉是因為人臉中各個區域的亮度相對關係不會受到光照的影響。

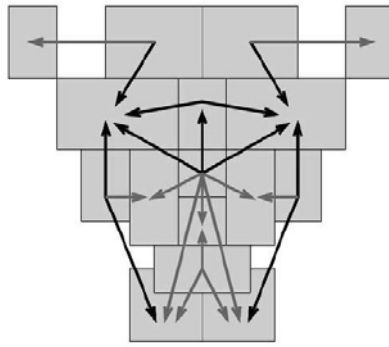


圖 2-1 人臉樣板示意圖

當測試影像中的人臉角度或大小與事先定義的人臉樣板不同時，樣板比對的人臉偵測方法可能會失效。為了解決這個問題，Miao 等人【22】把測試影像從 -20 度每次轉 5 度到 20 度且調整測試影像成幾個不同的大小。這個方法很直覺，可是很花時間，因為本來是只要偵測一張影像中的人臉，現在等於要偵測好幾張影像中的人臉。



## 2.4 以表面跡象為基礎的方法(Apppearance-Based Method)

以表面跡象為基礎的方法也有使用樣板，不過所用的樣板是從許多訓練影像中學習得來，而且還有用非人臉樣板。一般來說，以表面跡象為基礎的方法主要是從訓練影像中進行統計分析來學得人臉樣板與非人臉樣板的特性。這些特性以機率模型或區別函數(discriminant function)來表示後可用來做人臉偵測。

一張  $n \times m$  的人臉影像或非人臉影像可當成是一個  $n \times m$  的特徵向量(feature vector)，如果以機率模型來表示人臉影像或非人臉影像的分布，則一個特徵向量可看成是一個隨機變數  $x$ ，令  $p(x|face)$  和  $p(x|nonface)$  分別代表人臉影像與非人臉影像的條件機率函數，用貝式分類器或最大相似度(maximum likelihood)即可判斷在測試影像中的某個位置是否為人臉。不過，代表人臉影

像或非人臉影像的特徵向量維度通常很高使得  $p(x|face)$  和  $p(x|nonface)$  兩個函式的參數型不易求得，所以只能以  $p(x|face)$  和  $p(x|nonface)$  的近似參數型來實作判斷一張影像是否為人臉的貝式分類器。

除了以機率模型來表示人臉影像或非人臉影像的分佈外，找出人臉影像或非人臉影像在特徵向量空間的區別函數(discriminant function) 【23】，或稱決定表面(decision surface)，有了決定表面後就可以由測試影像是位於決定表面的那一邊來判斷該測試影像是不是人臉。實作上為了計算速度上的考量，一般照慣例的作法是先把人臉影像和非人臉影像的特徵向量投影到低維度空間，然後去尋找區別函數(discriminant function)。不過在低維度空間常無法找出明確的區別函數來區分人臉影像或非人臉影像，這時要用多層次的類神經網路(multilayer neural network) 【24】或支撐向量機器(support vector machine)

【25】的方法來解決，這兩個方法的中心思想是再把低維度的特徵向量投影到高維度空間(當然維度比原本降維前的特徵向量還低，不然一開始的降維就沒意義了)，因為在低維度空間無法區分的兩類特徵向量(人臉和非人臉)，到了高維度空間後就有機會能夠區分了。

Turk 和 Pentland 【23】提出了一個以表面跡象為基礎的人臉偵測方法，首先從一些訓練灰階影像中框出  $x$  個人臉影像與  $y$  個非人臉影像，調整這些影像到  $n \times m$  的大小，於是這些影像可看成是一個  $n \times m$  維的向量。接下來對  $x$  個人臉影像與  $y$  個非人臉影像計算一個共變異數矩陣(covariance matrix)，並計算此共變異數矩陣(covariance matrix)的固有值(eigenvalue)與對應的固有向量(eigenvector)，選擇最大的  $p$  個固有值所對應的固有向量， $p < n \times m$ ，把每個  $n \times m$  維的人臉影像再與這  $p$  個固有向量做內積可得到  $p$  個係數，於是原本  $n \times m$  維的人臉影像降成  $p$  維的向量；非人臉影像也以同樣的方式降成  $p$  維。這種降維的方式稱為主要成分分析(principle component analysis)，主要想法在於使得資料向量降維後與降維前的平均平方差(mean square error)最小。 $x$  個人臉影像降維後計算平均值， $y$  個非人臉影像也在降維後計算平均值，這兩個平均值就

分別代表人臉影像與非人臉影像。以測試影像中的每個位置為中心所形成的  $n \times m$  的影像區域都去做降維然後與人臉影像和非人臉影像的平均值計算距離，如果離人臉影像較近，則認為此位置有人臉。

Sung 和 Poggio 【26】認為各用一個平均值代表人臉影像與非人臉影像並不適當，首先他們以人工收集了 4150 張人臉影像與 43210 張非人臉影像，將每張影像調整到  $19 \times 19$  的大小，所以每個人臉影像與非人臉影像可看成是一個 361 維的向量，以 k 均值(k-means)演算法將人臉影像與非人臉影像各分成 6 群，每一群都假設其為一個高斯分佈，有平均值與共變異數矩陣。接下來 Sung 和 Poggio 定義了測試影像與人臉影像和非人臉影像的距離公式，他們認為有兩種距離要算。要計算第一種距離之前，先對 12 群的人臉影像與非人臉影像計算一個共變異數矩陣  $C$ ，然後利用  $C$  的固有向量做主要成分分析(principle component analysis)，把全部的影像從 361 維的向量降成 75 維的向量，每一群重新計算平均值與共變異數矩陣，然後測試影像要與某一群計算距離時，先用  $C$  的固有向量把測試影像降成 75 維的向量，然後再計算與那一群降維後的 Mahalanobis 距離，所算出來的距離即為第一種距離；測試影像與某一群的第二種距離的計算方式是先把測試影像用  $C$  的固有向量降成 75 維，然後後面補零再擴充成 361 維，最後與測試影像降維前的向量計算歐式距離，此即為第二種距離。最後用一個類神經網路來代表一個人臉分類器，首先把 4150 張人臉影像與 43210 張非人臉影像，每張影像去計算 24 種距離(因為總共 12 群，每一群可算兩種距離)，這 24 種距離可看成是一個 24 維的向量，此即為類神經網路的輸入；類神經網路的輸出只有兩種數字，1 代表人臉，-1 代表非人臉。有了輸入與輸出後，此類神經網路的參數就可從訓練影像訓練得來，訓練的演算法是倒傳遞(backpropagation)演算法。訓練好人臉分類器後，當給一個測試影像時，以一個  $19 \times 19$  的滑動視窗從左上到右下掃過整張測試影像，滑動視窗在測試影像中的每個位置都先與那 12 群人臉影像與非人臉影像計算 24 種距離，然後把這個 24 維的向量輸入人臉分類器，如果輸出比較接近 1，則代表測試影像的該位置有人臉。

Henry 等人【24】認為 Sung 和 Poggio 的方法有一個可改進的地方，即非人臉影像的收集。人臉影像的收集沒問題，只要去找不同人種、各年齡層、不同性別的人臉。可是要怎樣收集具有代表性的非人臉影像，Henry 等人提出一個 bootstrap 演算法來解決此問題，演算法如下：

(步驟一)

首先隨機產生 1000 張  $19 \times 19$  的影像，這 1000 張影像中任一位置的灰階值都是隨機給與，把這 1000 張影像當成是非人臉影像。

(步驟二)

將非人臉影像與收集好的人臉影像去訓練一個類神經網路，神經網路的輸出是 1 表示是人臉影像，-1 表示非人臉影像，訓練的方法同 Sung 和 Poggio。

(步驟三)

接下來把一部份完全沒有人臉的風景影像輸入此類神經網路，收集輸出是 1 的風景影像(也就是辨認錯誤的風景影像)。

(步驟四)

從辨認錯誤的風景影像中隨機選取 250 張加入非人臉影像中，回到步驟二。此過程反覆直到沒有人臉的風景影像都曾經通過步驟三。

其實這四類人臉偵測方法的邊界很模糊，例如 2.3 節介紹過有一種樣板比對方法不是用真的人臉影像，而是去定義人臉各區域之間的明暗關係，至於明暗關係的定義靠的是一般人對人臉的認識，所以這就跟以知識為基礎的方法很像；還有有的人臉偵測法不只屬於某一類，例如有一種樣板比對方法先使用一個人臉輪廓樣板在測試影像中尋找可能是人臉的區域，接著用眼睛、鼻子、嘴巴樣板去尋找眼睛、鼻子、嘴巴，這方法看起來同時也是以特徵為基礎的方法，因為它有去尋找眼睛、鼻子、嘴巴等臉部特徵；另外對於人臉偵測法有其他的分類，例如只分成兩類，第一類是有偵測眼睛、嘴巴等細部的臉部特徵；第二類是只定義好人臉的樣板模式，一旦符合樣板模式即認為偵測到人臉，完全不管眼睛、嘴巴等臉部特徵。



## 第 3 章

### 主播影像偵測與新聞故事分段

本章敘述所提的以影像偵測新聞主播的方法，還有說明在偵測到新聞主播之後如何做新聞故事的切割。圖 3-1 是主播影像偵測與新聞故事切割的流程圖，錄製新聞影像後依序對每張影像偵測與追蹤人臉，依據人臉區塊的特性抽取特徵，利用此特徵做分群，假設最大群為主播，就能篩選出主播人臉區塊，進而找出那些影像中有主播，最後判斷主播出現的時段作新聞故事切割。於本章 3.1 節介紹人臉區塊偵測與追蹤；3.2 節介紹主播人臉區塊篩選；3.3 節介紹如何在知道主播影像後做新聞故事切割。

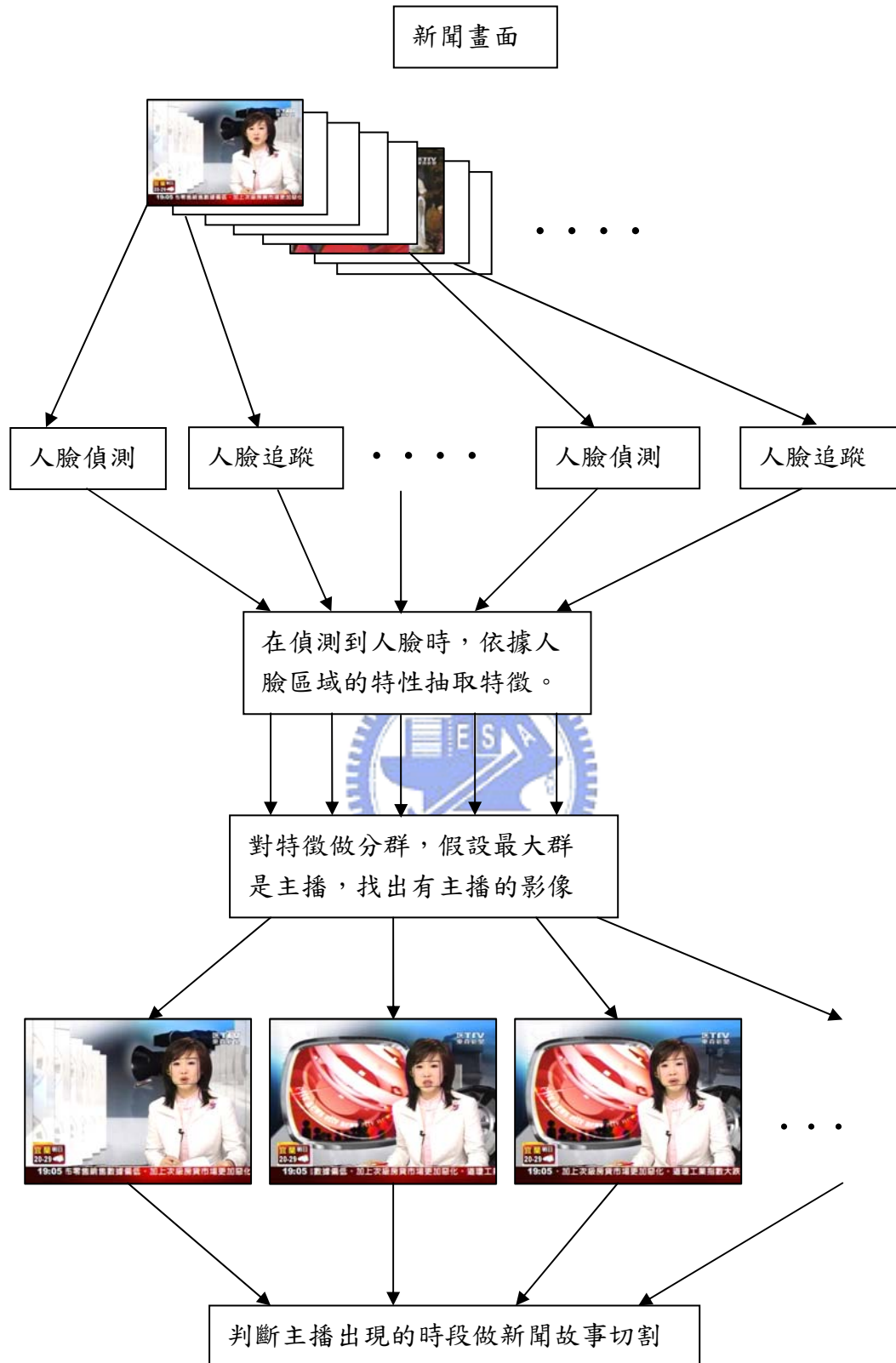


圖 3-1 主播影像偵測與新聞故事切割流程

### 3.1 人臉區塊偵測與追蹤

本節敘述如何偵測與追蹤人臉區塊，圖 3-2 是人臉區塊偵測與追蹤的流程圖，首先以每秒一張的速率擷取新聞影像，依序對每張影像做完整的人臉偵測，若有偵測到人臉，以兩種人臉追蹤的方式對接下來的影像偵測人臉，如果追蹤失敗則重做一次完整的人臉偵測。在 3.1.1 小節中介紹人臉偵測，在 3.1.2 小節中會介紹這兩種人臉追蹤方法。

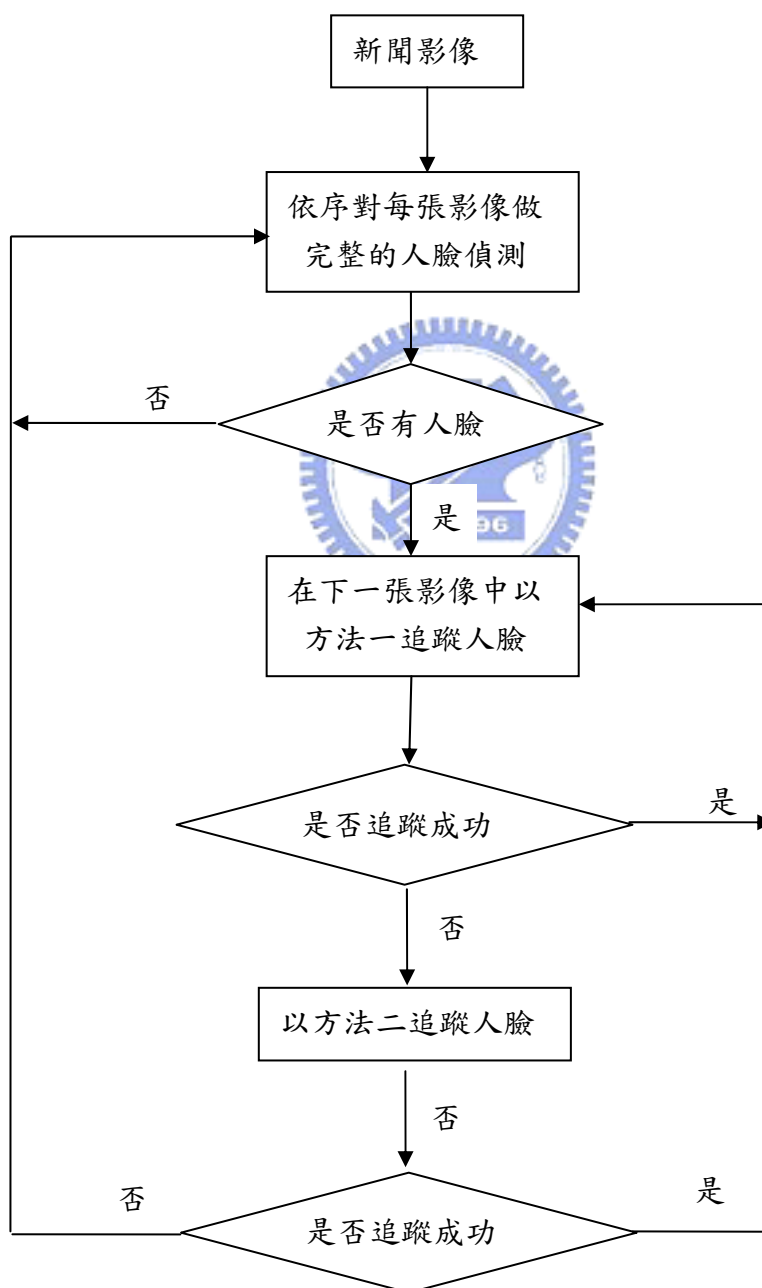


圖 3-2 人臉偵測與追蹤流程



### 3.1.1 人臉偵測

人臉偵測的方法有很多種，在第二章已介紹過，本論文所用的人臉偵測方法是整合第二章所提的方法。演算法如圖 3-3，首先將影像作均值移動分段以利之後的膚色判斷，以一個預先建立好的膚色模型找出膚色區域，接下來在每個膚色區域中去尋找眼睛，最後用一個人臉模型去驗證找到的眼睛是否屬於同一張臉。演算法的各步驟詳述如下：

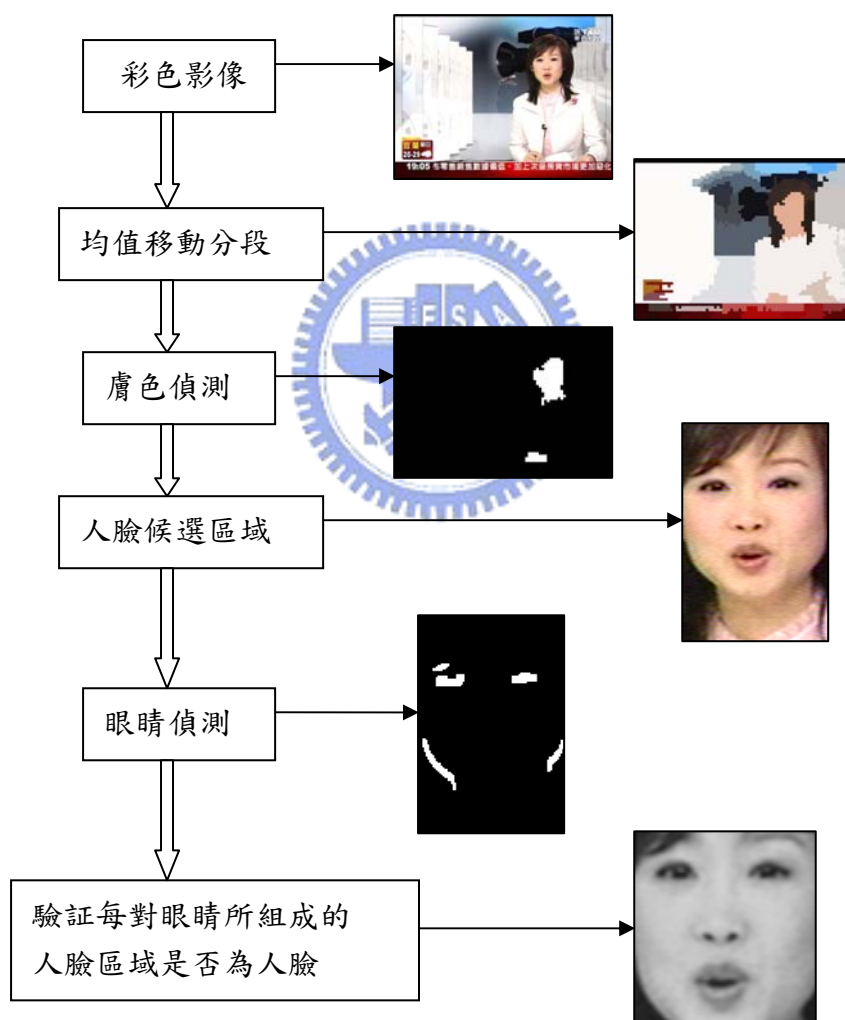


圖 3-3 人臉偵測流程

### 均值移動分段(Mean Shift Segmentation)(步驟一)：

將影像做均值移動分段的目的，是為了接下來的膚色判斷。因為以像素為單位做膚色判斷有許多缺點，像是眼睛、嘴巴、眉毛等不是膚色的地方，還有即使是臉部膚色的像素也有可能被膚色模型認為不是膚色，這些原因都會造成判斷出來的臉部區域是破碎且不完整的，在這種情況下通常使用形態上的運算(morphological operation)來處理使得分離的臉部區域合併，但型態上的運算(morphological operation)有可能造成一些不想要發生的情況，像是臉部各區域距離太遠以致無法合併，還有不屬於臉部的區域有可能被合併到臉部區域中。基於這些原因，本論文採用以區域為基礎(region-based)的膚色判斷，因此要對測試影像做均值移動分段(Mean Shift Segmentation)，演算法如下：

1. 將影像從  $640 \times 480$  的大小降取樣(downsampling)成  $80 \times 60$ ，並將影像轉換到 YUV 顏色空間，降取樣的目的除了加速之後的處理外，更重要的是在降取樣的影像中，眼睛、眉毛等臉部特徵會變得很小，如此在接下來的影像分段的過程中會被併入到臉部膚色區域中。

2. 對降取樣的影像做均值移動濾波(Mean Shift Filtering)，令  $x_i = (x_i^s, x_i^r)$  與  $z_i = (z_i^s, z_i^r)$  都是一個 5 維的向量，分別代表輸入影像與做完均值濾波後輸出影像的每個像素的空間 XY 座標與 YUV 值， $i=1, \dots, 4800$ ，其中  $x_i^s$  是一個 2 維的向量代表像素的空間 XY 座標， $x_i^r$  是一個 3 維的向量代表像素的 YUV 值。令  $y_{i,j}$  是一個 5 維的向量，代表  $x_i$  在第  $j$  次的均值移動(mean shift)後變成多少。均值移動濾波(Mean Shift Filtering)的演算法如下：

(a) 設  $j=1$  且  $y_{i,1} = x_i$ 。

(b) 由公式(1)計算  $y_{i,j+1}$  直到  $y_{i,j} = y_{i,j+1}$ ，此時令  $y_{i,c} = y_{i,j}$ ，其中

$$y_{i,j+1} = \frac{\sum_{i=1}^{480} x_i^s g\left(\left\|\frac{y_{i,j}^s - x_i^s}{h_s}\right\|^2\right) g\left(\left\|\frac{y_{i,j}^r - x_i^r}{h_r}\right\|^2\right)}{\sum_{i=1}^{480} g\left(\left\|\frac{y_{i,j}^s - x_i^s}{h_s}\right\|^2\right) g\left(\left\|\frac{y_{i,j}^r - x_i^r}{h_r}\right\|^2\right)} \quad (1)$$

$$g(x) = 1, \text{ if } 0 \leq x \leq 1.$$

$$g(x) = 0, \text{ if } x > 1.$$

$$h_s = 4, h_r = 8.$$

(c) 設定  $z_i = (x_i^s, y_{i,c}^r)$ 。

(d) 對  $i = 1, \dots, 4800$  都要做(a)、(b)、(c)。

3. 對均值移動濾波後的影像用區域成長(region growing)演算法找出各個相鄰且顏色相近的像素，給它們一個共同的編號。以一個  $5 \times 5$  的灰階影像說明區域成長演算法，如圖 3-4，格子中的數值代表像素的灰階值，數值的上標代表像素的編號，規定灰階值的差距在 2 以內才算相似。演算法的各步驟說明如下：

- (a) 將影像中所有像素的編號都設定成 -1，如圖 3-4(a)。
- (b) 以影像中最左上角的像素為起點，設定其編號為 1，判斷其右邊與下面的像素顏色是否與自己相近，若是，則設定其編號為 1，再從右邊與下面的像素向外發展，途中遇到顏色相近的像素都設定其編號為 1，直到找不到相鄰且顏色相近的像素為止，如圖 3-4(b)。
- (c) 若影像中還有編號為 -1 的像素，從中選擇最上方且最靠近左邊的像素為起點，設定其編號為 2，重複(b)中的尋找過程，並將所遇到的像素中編號為 -1 且顏色相近的像素，設定其編號為 2，如圖 3-4(c)。
- (d) 若影像中還有編號為 -1 的像素，從中選擇最上方且最靠近左邊的像素為起點，設定其編號為 3，重複(b)中的尋找過程，並將所遇到的像素中編號為 -1 且顏色相近的像素，設定其編號為 3，如圖 3-4(d)。
- (e) ... 依此類推，直到影像中沒有編號為 -1 的像素，如圖 3-4(e)。

(f) 從編號 1 為起點，每次將編號加 1，找出編號相同的像素，從中選擇最上方且最靠近左邊的像素，把所有其他編號相同的像素的顏色值都設定成這個像素的顏色值。

4. 再做一次 3.。最後把影像升取樣(upsampling)回  $640 \times 480$ ，我們把經過分段的影像稱為分段影像。

|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| $9^1$ | $9^1$ | $9^1$ | $3^1$ | $3^1$ |
| $9^1$ | $9^1$ | $6^1$ | $6^1$ | $3^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |

(a)

|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| $9^1$ | $9^1$ | $9^1$ | $3^1$ | $3^1$ |
| $9^1$ | $9^1$ | $6^1$ | $6^1$ | $3^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |

(b)

|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| $9^1$ | $9^1$ | $9^1$ | $3^2$ | $3^2$ |
| $9^1$ | $9^1$ | $6^1$ | $6^1$ | $3^2$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |
| $6^1$ | $6^1$ | $6^1$ | $0^1$ | $0^1$ |

(c)

|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| $9^1$ | $9^1$ | $9^1$ | $3^2$ | $3^2$ |
| $9^1$ | $9^1$ | $6^3$ | $6^3$ | $3^2$ |
| $6^3$ | $6^3$ | $6^3$ | $0^1$ | $0^1$ |
| $6^3$ | $6^3$ | $6^3$ | $0^1$ | $0^1$ |
| $6^3$ | $6^3$ | $6^3$ | $0^1$ | $0^1$ |

(d)

|       |       |       |       |       |
|-------|-------|-------|-------|-------|
| $9^1$ | $9^1$ | $9^1$ | $3^2$ | $3^2$ |
| $9^1$ | $9^1$ | $6^3$ | $6^3$ | $3^2$ |
| $6^3$ | $6^3$ | $6^3$ | $0^4$ | $0^4$ |
| $6^3$ | $6^3$ | $6^3$ | $0^4$ | $0^4$ |
| $6^3$ | $6^3$ | $6^3$ | $0^4$ | $0^4$ |

(e)

圖 3-4 區域成長演算法範例

### 建立膚色模型且用建立好的膚色模型判斷影像中的膚色區域（步驟二）：

首先收集東森、民視、華視、中天、三立五家電視台，每家各兩個新聞主播，共十個主播的臉部膚色像素，總共約 22000 個像素。將這些像素的顏色從 RGB 顏色空間轉換到 YCbCr 顏色空間，用 2.2 節所提過的以長方條代表的統計圖 (histogram) 來建立膚色模型，令  $h(Cb_1, Cr_1)$  代表 CbCr 平面上 CbCr 值等於  $(Cb_1, Cr_1)$  的長方條高度，同時規定長方條高度的門檻值  $\tau$ 。若測試像素的 CbCr 值為  $(Cb_2, Cr_2)$  且  $h(Cb_2, Cr_2) > \tau$ ，則認為此測試像素的顏色是膚色，用這方法判斷分段影像中的每個位置是不是膚色，就可以做出跟原影像 (未經過任何處理) 一樣大的一個膚色二元圖，如圖 3-5(c)，1 表示該位置是膚色，0 則表示不是膚色。接著在膚色二元圖中找出所有的 8 連接成分 (8-connected component)，屬於同一個 8 連接成份是指二元圖中所有是 1 且有相鄰或對角相鄰連接的關係，每一個 8 連接成份可看成是一個膚色區域。

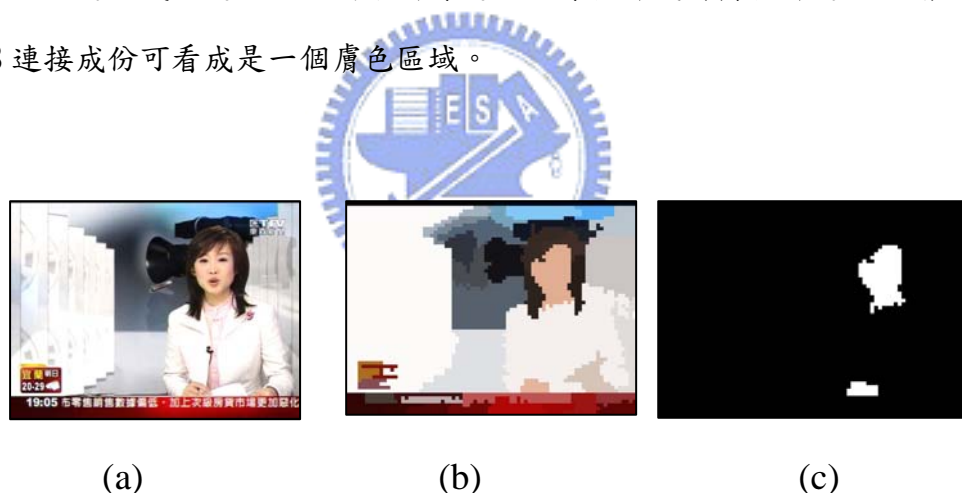


圖 3-5 膚色區域偵測。(a)圖代表原影像，(b)圖代表分段影像，(c)圖代表膚色區域。

### 判斷每個膚色區域是否有眼睛（步驟三）：

用一個最小的矩形包圍一個膚色區域，在原影像 (未經過任何處理) 中找出這個矩形區域，此步驟就是要在這個矩形區域中找出眼睛。觀察眼睛區域的像素可發現虹膜的灰階值比膚色低，眼白的灰階值比膚色高；虹膜的 Cb 值比膚色高，

Cr 值比膚色低。利用這些特性就能找出眼睛的位置，如圖 3-6，演算法如下：

1. 定義  $\text{EyeMapC} = \frac{1}{3} \left\{ (Cb^2) + (\tilde{Cr})^2 + (Cb/Cr) \right\}$ ，其中  $Cb^2$ ， $Cb/Cr$ ， $(\tilde{Cr})^2$  都要調整到  $[0,255]$  這個範圍內， $\tilde{Cr} = 255 - Cr$ 。
2. 定義  $\text{EyeMapL} = \frac{Y(x,y) \oplus g(x,y)}{Y(x,y) \ominus g(x,y) + 1}$ ，其中  $Y$  表示矩形區域的灰階影像， $\oplus$  代表膨脹(dilation)運算， $\ominus$  代表侵蝕(erosion)運算。令  $R(x,y) = \sqrt{x^2 + y^2}$ ，  
if  $R \leq |\sigma|$ ， $g(x,y) = |\sigma| \cdot \left( \left| 1 - (R(x,y)/\sigma)^2 \right|^{0.5} - 1 \right)$ ；else  $g(x,y) = -\infty$ 。
3. 定義  $\text{EyeMap} = \text{EyeMapC} * \text{EyeMapL}$ ，其中  $*$  代表點對點相乘。
4.  $\text{EyeMap}$  是個灰階影像，設定一個門檻值(threshold)使其變為二元影像，1 代表該位置是眼睛，0 代表該位置不是眼睛。
5. 在  $\text{EyeMap}$  中找 4 連接成份(4-connected component)，屬於同一個 4 連接成份是指二元圖中所有是 1 且有相鄰連接的關係，每一個 4 連接成份可看成是一個眼睛區域。

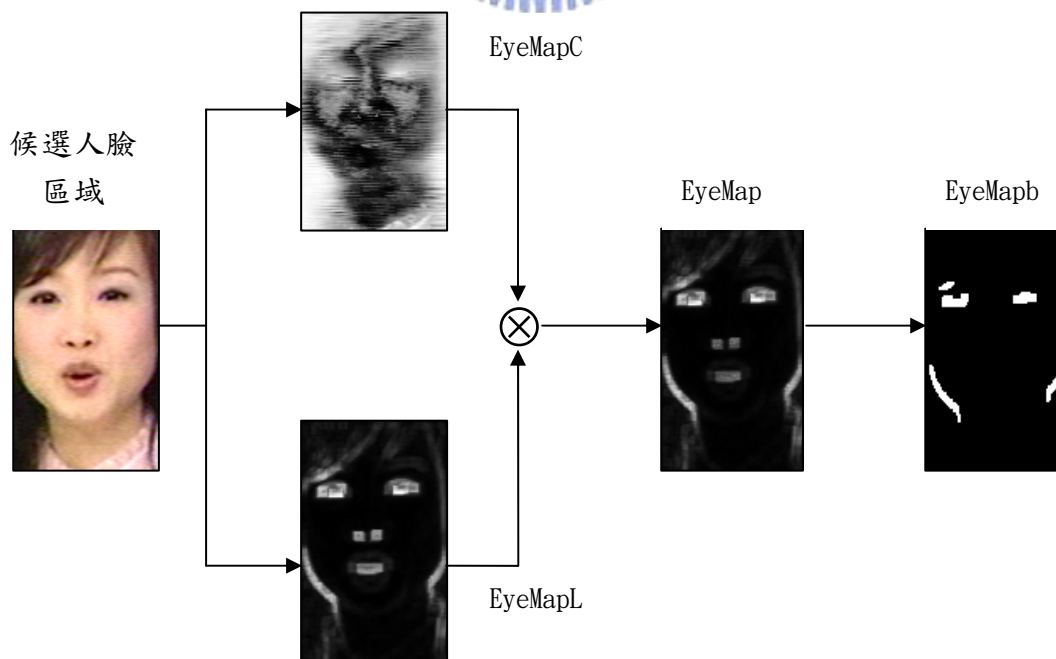


圖 3-6 在候選人臉區域中偵測眼睛



#### 訓練人臉樣板與驗證每對眼睛是否屬於同一張臉（步驟四）：

因為找出的眼睛不一定是真正的眼睛，像是鼻孔或頭髮的邊緣也有可能被認為是眼睛，所以需要進一步的驗證，又因為眼睛是一對的，所以是兩個眼睛一組來驗證，如圖 3-7 所示有好幾個人臉驗證候選區域。

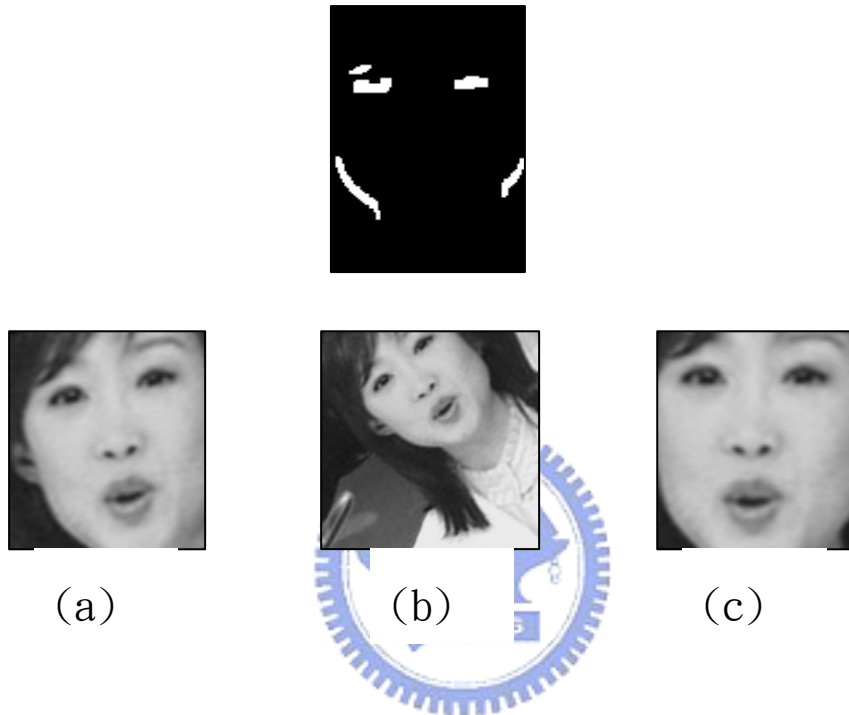


圖 3-7 由眼睛配對所產生的人臉驗證候選區域

在驗證之前需要有一個人臉樣板，首先收集東森、民視、華視、中天、三立五家電視台，每家各兩個新聞主播，共十個主播，每個主播 7 張人臉影像，總共 70 張，將這 70 張人臉影像都調整到  $20 \times 20$  的大小，每張人臉影像可看成是一個 400 維的向量，用主要成分分析(principle component analysis)的方法將 400 維的向量降成 10 維，假設降維後的人臉影像為一個高斯分佈，算出其平均值與共變異數矩陣。如此人臉樣板可表示成一個高斯分佈的機率模型。有了人臉樣板後，我們就能驗證找到的眼睛是否屬於同一張人臉，演算法如下：

1. 計算這兩個眼睛的方向，如果這兩個眼睛的方向差的很多，例如一個眼睛方向是右上到左下，另一個眼睛的方向是左上到右下，則這兩個眼睛明顯不是

同一個人的眼睛，此時繼續驗證下一組眼睛配對。反之，兩個眼睛的方向一致的話則繼續下一步 2.。

2. 以這兩個眼睛的中心位置估計一個四邊形的人臉區域，在原影像的灰階影像中取出這個人臉區域。
3. 將這個人臉區域調整到  $20 \times 20$  的大小，此時可把這個人臉區域看成是一個 400 維的向量，用所收集的 70 張人臉影像的共變異數矩陣的特徵向量做主要成分分析(principle component analysis)將其降成 10 維的向量。
4. 把這 10 維的向量代入人臉樣板的高斯模型中，計算其與高斯模型的近似值(likelihood)，如果有超過規定的門檻值(threshold)，則認為這個人臉區域確實是人臉，否則不認為是人臉。
5. 重覆 1.2.3.4. 驗證一個膚色區域中所有的眼睛配對，選擇一個近似值(likelihood)最高且超過門檻值的人臉區域，認為此人臉區域的確是一個人臉。



### 3.1.2 以人臉追蹤加速人臉偵測

觀察幾天的新聞影像後，我們發現主播在播報同一則新聞或外景的某個人物在接受訪問時，下半身通常是靜止不動，而頭部只會有輕微的晃動。根據這個特性，我們不需要每張影像做完整的人臉偵測，只要在偵測到人臉的同時對接下來的影像在某一個範圍內去追蹤這個人臉，如此一來可加速人臉偵測的步驟。本節介紹論文中所用的兩個人臉追蹤演算法。

#### (方法一)眼睛區塊比對的人臉追蹤演算法

演算法如圖 3-8，各步驟說明如下：

1. 對第一張影像偵測人臉，若有找到人臉，紀錄左右兩眼的中心位置，把左眼與右眼區塊剪下，當成要在第二張影像中搜尋比對的目標區塊。



2. 在第二張影像中，以第一張影像的左右兩眼的中心位置為準，各定出一個搜尋範圍，在此範圍內搜尋比對眼睛區塊，如果左眼與右眼區塊都有找到，則人臉追蹤成功。
3. 兩區塊比對的方式是先設定一個門檻值。然後兩區塊灰階值點對點相減，取絕對值後相加，如果有小於門檻值，則認為比對成功。
4. 在搜尋區域內可能會找到好幾個小於門檻值的位置，此時我們選擇差距最小的那個位置，而不是一找到某個小於門檻值的位置即停止。

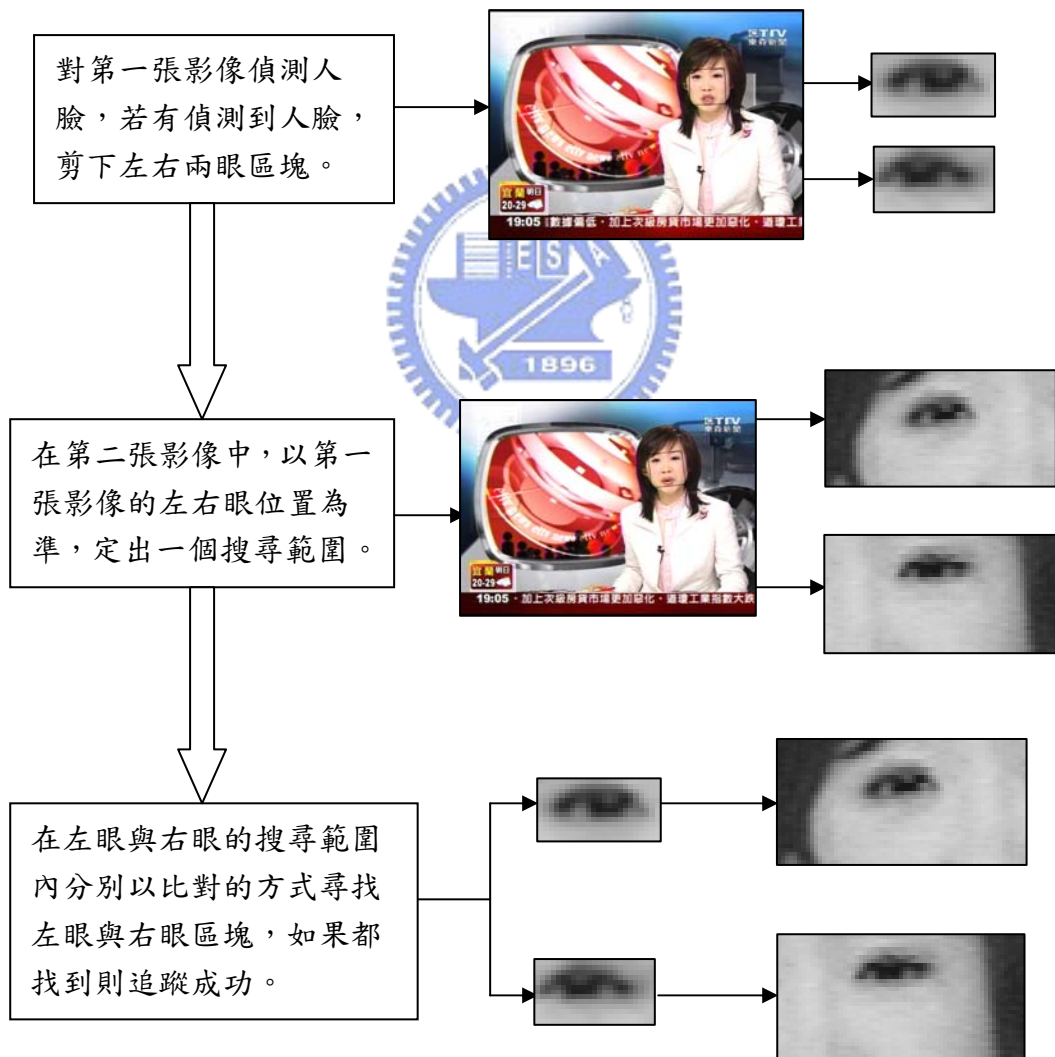


圖 3-8 眼睛區塊比對的人臉追蹤演算法

## (方法二) 驗證人臉區域的人臉追蹤演算法

演算法如圖 3-9，各步驟說明如下：

1. 對第一張影像偵測人臉，若有找到人臉，紀錄左右兩眼的中心位置。
2. 在第二張影像中，以第一張影像的左右兩眼的中心位置為準，各定出一個搜尋範圍，在此範圍內搜尋眼睛，搜尋的方法與 3.1.1 節中介紹的方法一樣。
3. 將左右兩眼的搜尋範圍中找出的眼睛兩兩配對，再用 3.1.1 節中介紹的方法去驗證每對眼睛是否屬於同一張臉。

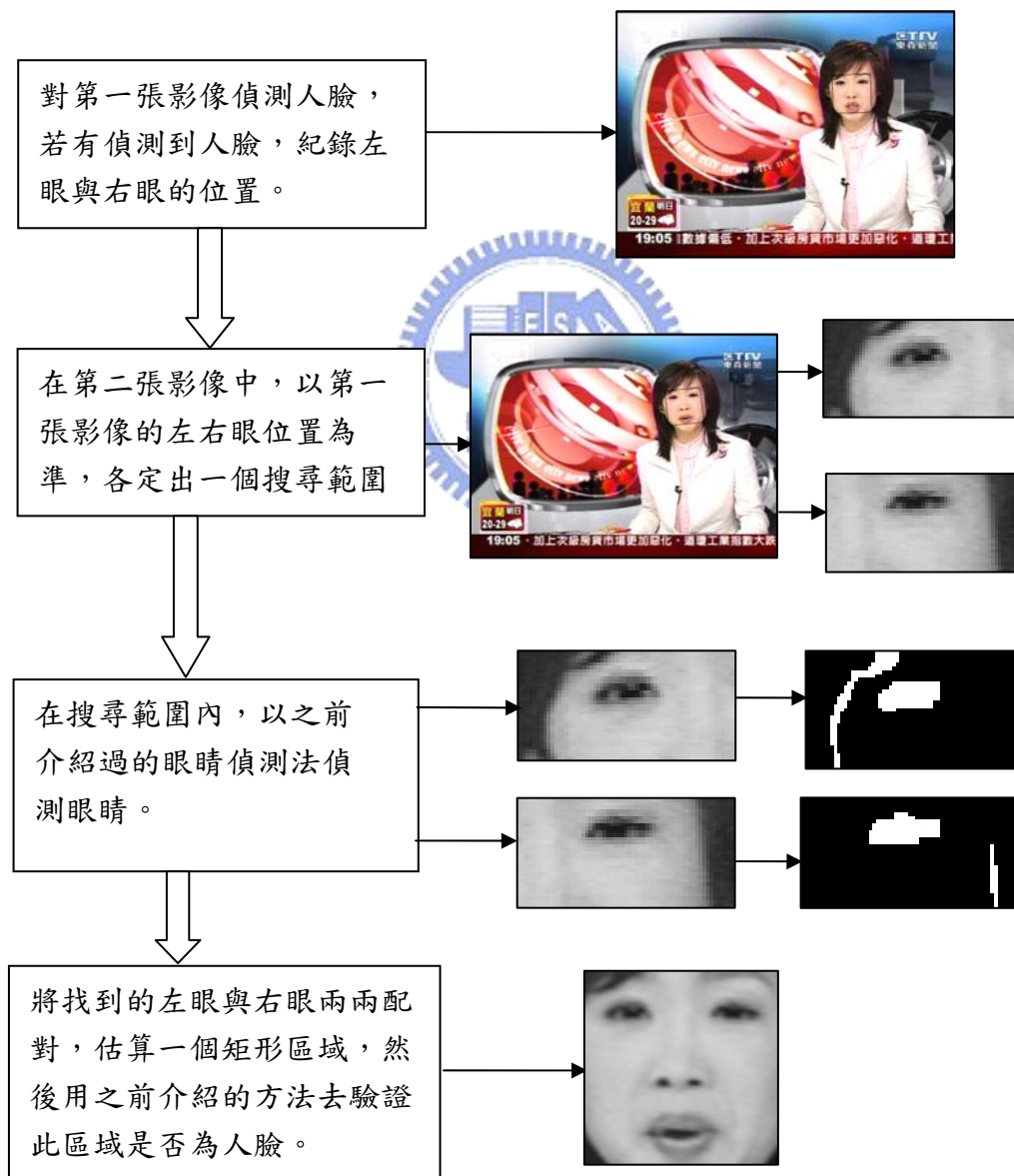


圖 3-9 驗證人臉區域的人臉追蹤演算法

在實作時是先用方法一追蹤，沒有追蹤到的話再用方法二追蹤。如果都沒有追蹤到人臉或追蹤到的人臉數與前一張影像不同，則對整張影像重新做完整的人臉偵測。

## 3.2 主播人臉區塊篩選

在偵測完新聞影像中所有的人臉區塊後，接下來從中找出主播人臉區塊，我們就能知道那些影像有主播。首先根據每個人臉區塊的特性抽取特徵，對此特徵作分群，假設主播為最大群來篩選出主播區塊。於 3.2.1 小節說明如何由人臉區塊的特性抽取特徵；3.2.2 小節中說明如何做分群。

### 3.2.1 根據人臉區塊的特性抽取特徵

我們所選的特徵是衣服顏色，由人臉區塊的中心位置與長寬來估計衣服的区域，如圖 3-10。統計在衣服範圍內那個顏色出現最多，以此顏色當成衣服顏色。另外，為了避免顏色過多以致於同一件衣服在不同的影像中找出的代表色不同，將顏色以比較少的位元數來表示，例如本來是 8 位元，改成 5 位元。還有為了使得視覺上看起相近的顏色，在顏色空間上的數值也很接近，我們將衣服範圍內的顏色由 RGB 顏色空間轉換到 Lab 顏色空間，使用 Lab 顏色空間的距離公式。



(a)



(b)

圖 3-10 估計衣服區域。(a)人臉偵測結果(b)由人臉中心位置估計的衣服區域。

### 3.2.2 基於最大群原則的主播區域篩選

我們以衣服顏色當作每個人的特徵，利用此特徵來分群，假設在新聞影片中主播出現最多次，因此分群完後最大的那一群即是主播，分群的演算法有許多種，本論文使用四種分群法做實驗，選擇效能最好的分群法。假設  $X = \{x_1, x_2, x_3, \dots, x_n\}$  是全部衣服顏色的資料點集合，以下介紹這四種分群法。

**循序式分群法：**

1.  $\{x_1\}$  是一群，令其為  $C_1$ ，並設定一個兩群間距離的門檻值  $T$ 。
2. 依序對  $x_i$ ， $2 \leq i \leq n$ ，找一個距離最近的群  $C_m$ ，如果此距離小於  $T$ ，則把  $x_i$  加入  $C_m$ ，更新  $C_m$  的群中心，反之， $x_i$  自己形成一個新的群。群中心的計算方式為群內所有點的平均。

**階層式凝聚分群法(使用平均鏈結)：**

1.  $X$  中每個資料點各自是一群，並設定一個兩群間距離的門檻值  $T$ 。
2. 假設  $A = \{x_i, \dots, x_j\}$  是一群， $B = \{x_m, \dots, x_k\}$  是一群， $1 \leq i < j < m < k \leq n$ ，則  $A$  與  $B$  之間的距離是  $\frac{1}{(k-m) + (j-i) + 2} \sum_{a=i}^j \sum_{b=m}^k dist(x_b, x_a)$ ，其中  $dist(x_a, x_b)$  就是 Lab 顏色空間中兩個點的距離。
3. 從  $X$  中選擇最相近的兩群，若此兩群的距離小於  $T$ ，則將此兩群合併，否則分群完成。
4. 重複步驟 3。

**階層式凝聚分群法(使用完全鏈結)：**

演算法與階層式凝聚分群法(使用平均鏈結)一樣，只不過距離計算方式不同。假設  $A = \{x_i, \dots, x_j\}$  是一群， $B = \{x_m, \dots, x_k\}$  是一群， $1 \leq i < j < m < k \leq n$ ，則  $A$  與  $B$  之間的距離是  $\max_{i \leq a \leq j, m \leq b \leq k} dist(x_a, x_b)$ 。

**階層式凝聚分群法(使用單鏈結)：**

演算法與階層式凝聚分群法(使用平均鏈結)一樣，只不過距離計算方式不

同。假設  $A = \{x_i, \dots, x_j\}$  是一群， $B = \{x_m, \dots, x_k\}$  是一群， $1 \leq i < j < m < k \leq n$ ，則  $A$  與  $B$  之間的距離是  $\min_{i \leq a \leq j, m \leq b \leq k} dist(x_a, x_b)$ 。

### 3.3 新聞故事切割

在一小時 3600 張的新聞影像中，知道那幾張影像有主播後，我們就能做新聞故事切割，方法很簡單，就是要找出主播出現的時間序列，演算法如下：

1. 在有主播出現的影像中，找出時間上連續的主播影像序列。
2. 假設外景的時間不可能少於兩秒，所以合併間隔在兩秒(兩張影像)內的主播影像序列，使其成為一個更長的主播影像序列。
3. 假設主播播報新聞的時間不可能少於三秒，所以刪除長度在三秒(三張影像)內的主播影像序列。
4. 有了主播出現的時間序列後，每則新聞故事的時間就是該則主播出現的時間到下一個主播開始出現的時間，如圖 3-11。

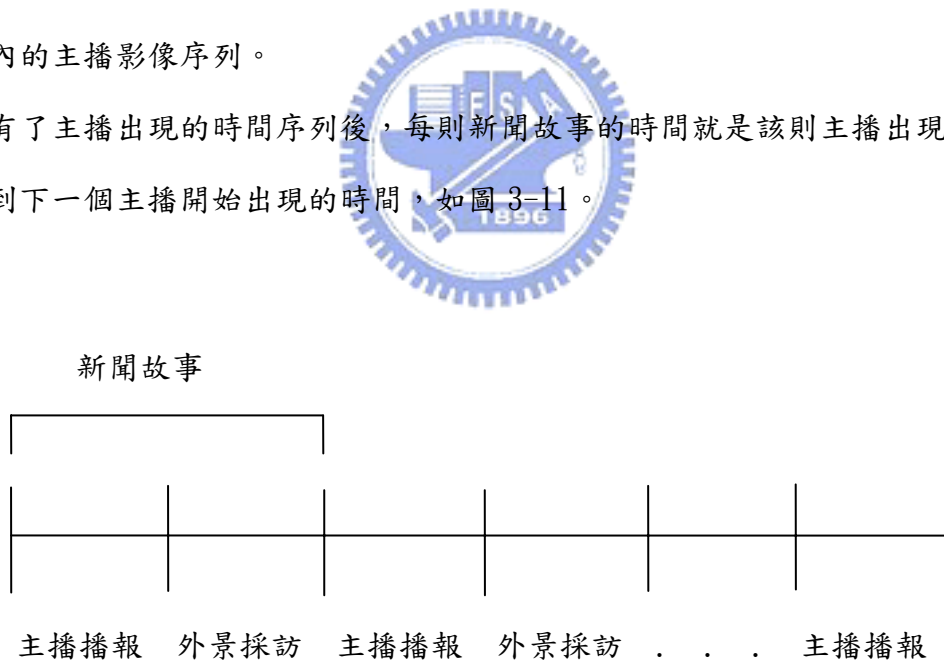


圖 3-11 新聞故事架構

## 第 4 章

### 實驗結果

在這章中對於第三章所提的主播影像偵測方法與新聞故事切割方法，加以實作，並設計實驗來評估此方法的效能。對於實驗的平台，在硬體方面使用了以 Intel Pentium-4 2.4Ghz 的時脈速率的中央處理器的個人電腦，搭配有 1Gigabytes 主記憶體，並接有電視影像擷取卡，作業系統為 Microsoft Windows XP 專業版。

#### 4.1 主播影像偵測實驗與結果分析

本論文所提的主播影像偵測方法的第一步是偵測與追蹤每張影像的人臉，可是不能保證使用追蹤的人臉偵測方式是否能達到預期的加速效果且效能不會降低。所以 4.1.1 節的實驗會去比較有無人臉追蹤的人臉偵測效能；新聞主播影像偵測方法的最後是對衣服顏色做分群，分群好後最大的那一群為主播，可是不知道要用那種分群法較好，所以 4.1.2 節的實驗會去比較 4 個不同分群法的分群效果。資料來源為東森、中天、民視、華視、三立五家電視新聞台，各一個小時的新聞影像，取得的方式是用電視擷取卡以每秒一張的速率擷取整點新聞的影像，每次 3600 張影像，用於主播影像偵測。



### 4.1.1 人臉偵測實驗與結果分析

本小節的實驗要比較有無人臉追蹤的人臉偵測效能，評量的標準有兩個，第一個是人臉偵測的效能，第二個是主播的人臉在人臉偵測的過程是否被誤失，結果如表 4-1，4-2，4-3，4-4，4-5。

表 4-1 是沒有用人臉追蹤的人臉偵測結果，第一欄是電視台名稱，第二欄是在 3600 張影像中真正的人臉數，第三欄是程式偵測到的人臉數，第四欄是正確偵測到的人臉數，第五欄是正確率，第六欄是召回率；表 4-2 是沒有用人臉追蹤的主播人臉誤失結果，第一欄是電視台名稱，第二欄是主播的人臉數，第三欄是主播的人臉在人臉偵測的過程中有多少被誤失，第四欄是誤失率；表 4-3 是有用人臉追蹤的人臉偵測結果，各欄位的意義同表 4-1；表 4-4 是有用人臉追蹤的主播人臉誤失結果，各欄位的意義同表 4-2；表 4-5 是將平均的結果並列。

表 4-1 人臉偵測結果(沒有用人臉追蹤)

| 電視台 | 真正的人臉數 | 程式偵測到的人臉數 | 正確偵測到的人臉數 | 正確率 | 召回率 |
|-----|--------|-----------|-----------|-----|-----|
| 東森  | 1051   | 1308      | 756       | 58% | 72% |
| 中天  | 1122   | 1397      | 724       | 52% | 65% |
| 民視  | 1251   | 1563      | 871       | 56% | 70% |
| 華視  | 774    | 997       | 539       | 54% | 70% |
| 三立  | 946    | 1258      | 679       | 54% | 72% |
| 平均  |        |           |           | 55% | 70% |

表 4-2 主播的人臉在人臉偵測的過程中的誤失率(沒有用人臉追蹤)

| 電視台 | 主播的人臉數 | 在人臉偵測的過程中誤失的人臉數 | 誤失的比率 |
|-----|--------|-----------------|-------|
| 東森  | 576    | 9               | 2%    |
| 中天  | 505    | 23              | 1%    |
| 民視  | 628    | 92              | 15%   |
| 華視  | 408    | 5               | 1%    |
| 三立  | 699    | 49              | 7%    |
| 平均  |        |                 | 6%    |

表 4-3 人臉偵測結果(有用人臉追蹤)

| 電視台 | 真正的人臉數 | 程式偵測到的人臉數 | 正確偵測到的人臉數 | 正確率 | 召回率 |
|-----|--------|-----------|-----------|-----|-----|
| 東森  | 1051   | 1299      | 780       | 60% | 74% |
| 中天  | 1122   | 1431      | 755       | 53% | 67% |
| 民視  | 1251   | 1471      | 906       | 61% | 72% |
| 華視  | 774    | 1016      | 554       | 55% | 72% |
| 三立  | 946    | 1233      | 749       | 61% | 79% |
| 平均  |        |           |           | 58% | 73% |

表 4-4 主播的人臉在人臉偵測的過程中的誤失率(有用人臉追蹤)

| 電視台 | 主播的人臉數 | 在人臉偵測的過程中誤失的人臉數 | 誤失的比率 |
|-----|--------|-----------------|-------|
| 東森  | 576    | 9               | 2%    |
| 中天  | 505    | 17              | 3%    |
| 民視  | 628    | 69              | 10%   |
| 華視  | 408    | 1               | 0.1%  |
| 三立  | 699    | 47              | 7%    |
| 平均  |        |                 | 5%    |



表 4-5 有無人臉追蹤的人臉偵測效能比較

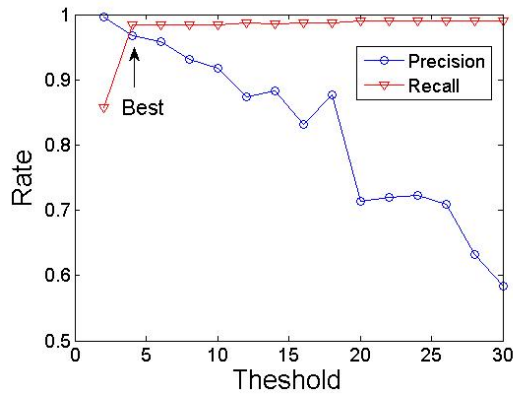
|        | 人臉偵測正確率 | 人臉偵測召回率 | 主播人臉的誤失率 |
|--------|---------|---------|----------|
| 沒用人臉追蹤 | 55%     | 70%     | 6%       |
| 有用人臉追蹤 | 58%     | 73%     | 5%       |

從表 4-1 與 4-2 來看，雖然人臉偵測的效能不好，不過主播人臉在人臉偵測的過程中被誤失的比率很小，所以可以繼續進行接下來的取衣服顏色及分群處理；從表 4-5 來看沒用人臉追蹤的人臉偵測效果與有用人臉追蹤的效果差不多，而且主播人臉在人臉偵測的過程中誤失率也差不多，不過有用人臉追蹤的人臉偵測速度約比沒用人臉追蹤的速度快上 30%，所以的確可使用人臉追蹤來加速人臉偵測。

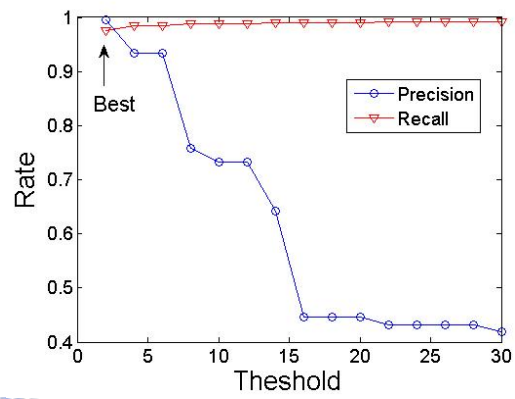
#### 4.1.2 分群實驗與結果分析

本小節的實驗要比較循序式分群法、階層式凝聚分群法（使用平均鏈結）、階層式凝聚分群法（使用完全鏈結）、階層式凝聚分群法（使用單鏈結）這四種分群法對衣服特徵的分群效果。由於分群好後程式會指出那些影像有主播，然後再跟正確答案做比對，計算正確率與召回率，藉由這兩個指標來比較四個分群法的效能。

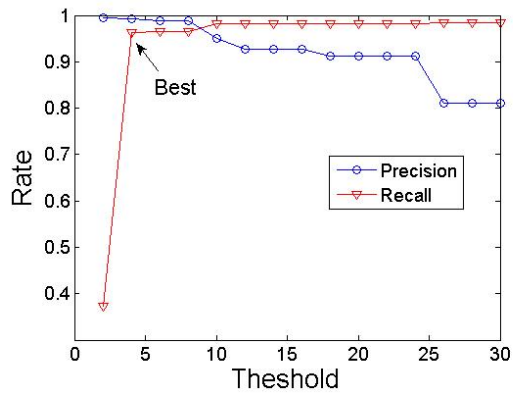
圖 4-1 是東森 1 小時衣服資料的實驗結果，當中的 (a)、(b)、(c)、(d) 小圖分別是循序式分群法、階層式凝聚分群法（使用平均鏈結）、階層式凝聚分群法（使用完全鏈結）、階層式凝聚分群法（使用單鏈結）這四種分群法的結果。每個小圖中的橫座標是門檻值，縱座標是比率，圖中的的兩條線分別代表正確率與召回率，箭頭所指代表正確率加召回率最高的地方。圖 4-2、圖 4-3、圖 4-4、圖 4-5 分別代表中天、民視、華視、三立 1 小時衣服資料的實驗結果。



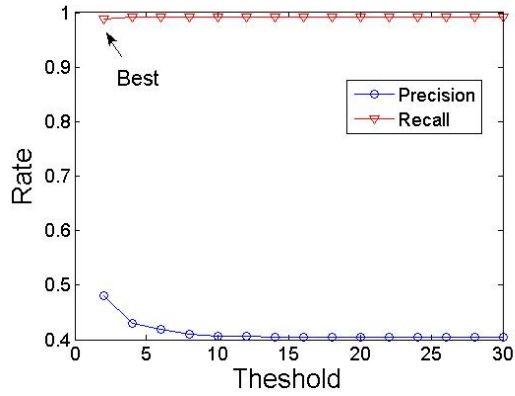
(a)



(b)

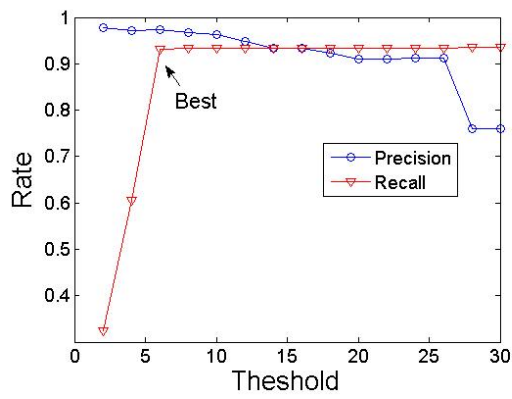


(c)

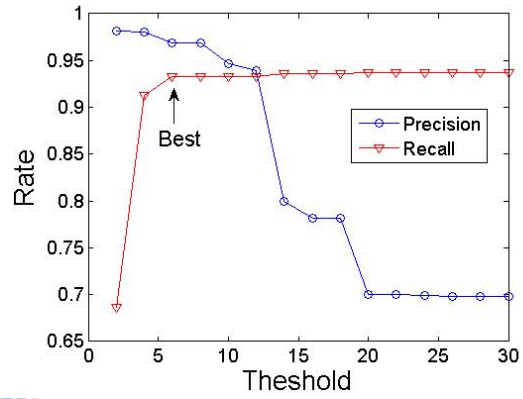


(d)

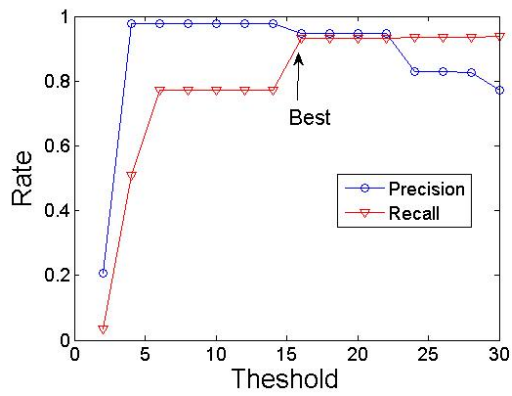
圖 4-1 東森 1 小時衣服資料的分群實驗結果



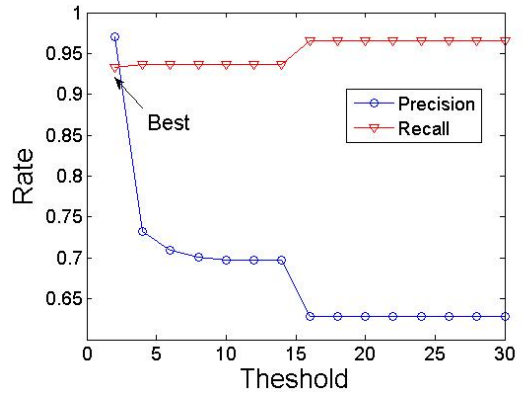
(a)



(b)

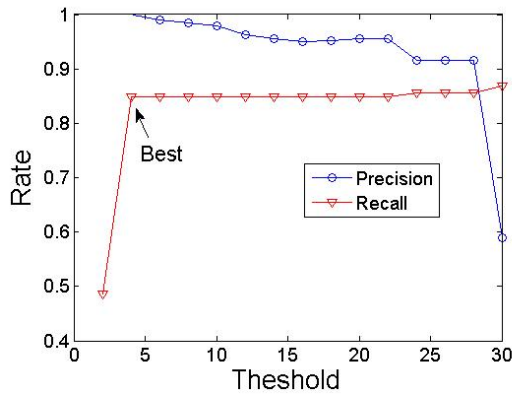


(c)

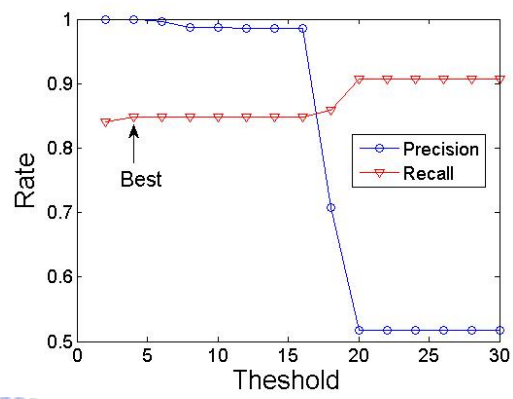


(d)

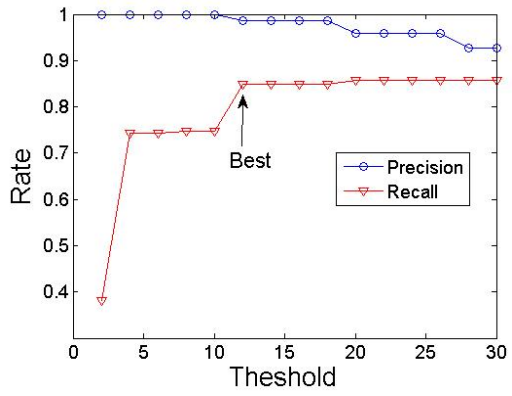
圖 4-2 中天 1 小時衣服資料的分群實驗結果



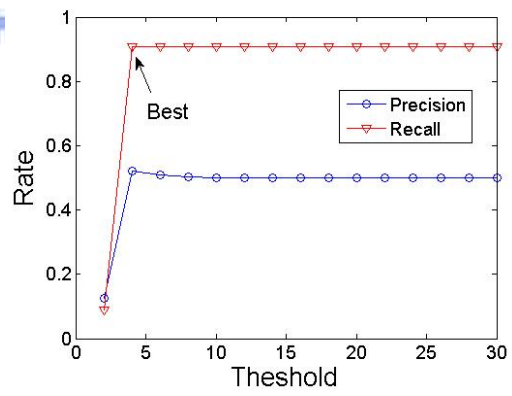
(a)



(b)

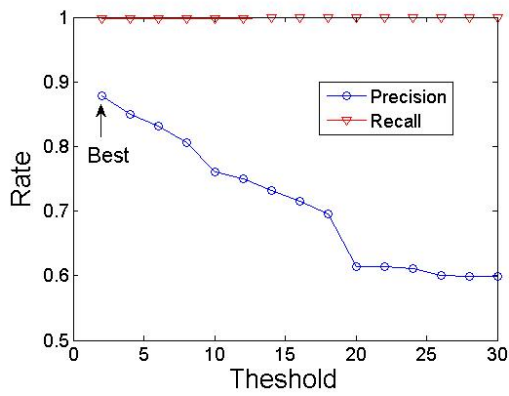


(c)

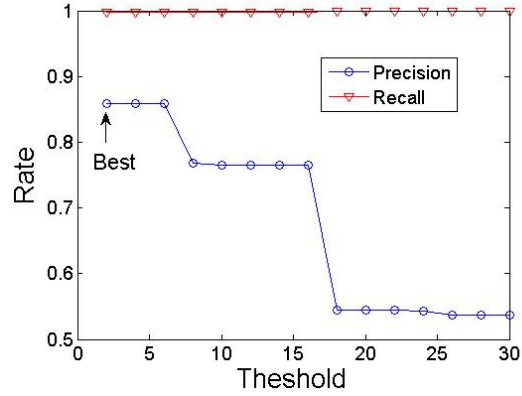


(d)

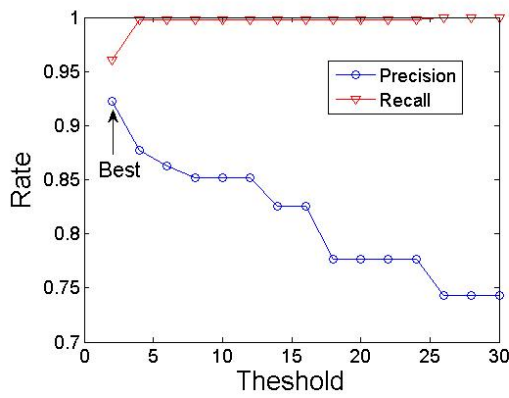
圖 4-3 民視 1 小時衣服資料的分群實驗結果



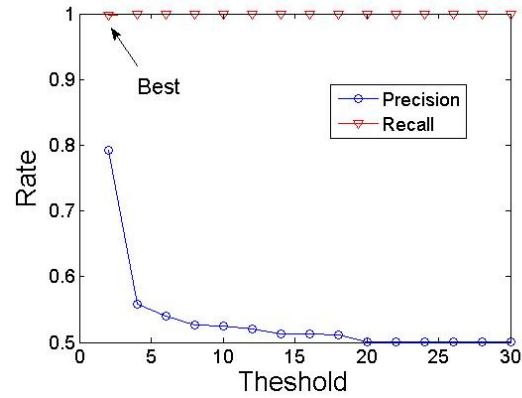
(a)



(b)

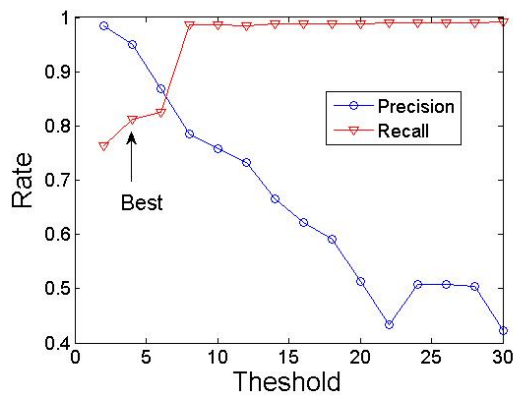


(c)

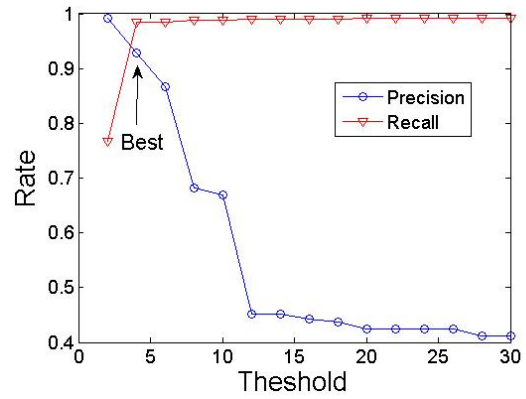


(d)

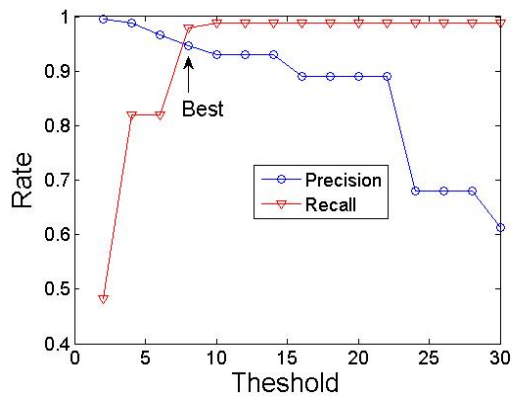
圖 4-4 華視 1 小時衣服資料的分群實驗結果



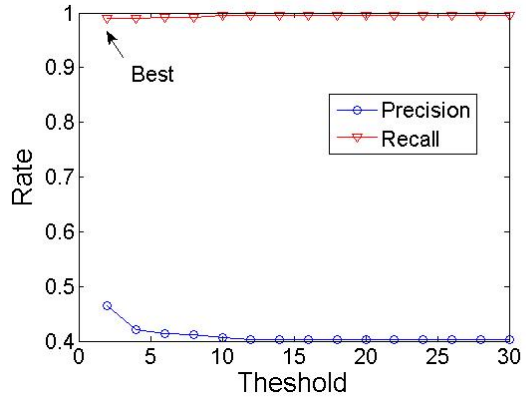
(a)



(b)



(c)



(d)

圖 4-5 三立 1 小時衣服資料的分群實驗結果

表 4-6 是把結果並列，第一欄是電視台名稱，第二、三、四、五欄分別是 4 種分群法在不同的門檻值下所能找出最好的正確率與召回率，從結果來看，似乎是階層式凝聚分群法（使用平均鏈結）較好，雖然比起循序式分群法和階層式凝聚分群法（使用完全鏈結）並沒有顯著的效能提升，不過階層式凝聚分群法（使用平均鏈結）的效能較為穩定，所以本論文使用階層式凝聚分群法（使用平均鏈結）來對衣服特徵做分群。

表 4-6 分群實驗的結果並列

|    | 循序式分群 |       | 階層式群凝聚分群<br>(使用平均鏈結) |       | 階層式凝聚分群<br>(使用完全鏈結) |       | 階層式凝聚分群<br>(使用單鏈結) |       |
|----|-------|-------|----------------------|-------|---------------------|-------|--------------------|-------|
|    | 正確率   | 召回率   | 正確率                  | 召回率   | 正確率                 | 召回率   | 正確率                | 召回率   |
| 東森 | 96.7% | 98.4% | 99.4%                | 97.5% | 99.3%               | 96.3% | 47.9%              | 98.9% |
| 中天 | 78.4% | 98.7% | 92.9%                | 98.5% | 94.7%               | 97.9% | 46.4%              | 99.0% |
| 民視 | 100%  | 84.9% | 100%                 | 84.9% | 98.7%               | 84.9% | 52.0%              | 90.7% |
| 華視 | 87.9% | 99.7% | 85.8%                | 99.7% | 92.2%               | 96.0% | 79.2%              | 99.7% |
| 三立 | 97.4% | 93.1% | 96.8%                | 93.3% | 94.7%               | 93.3% | 97.0%              | 93.3% |
| 平均 | 92.1% | 94.9% | 95.0%                | 94.8% | 95.9%               | 93.7% | 74.2%              | 96.2% |

## 4.2 新聞故事切割實驗與結果分析

知道那些影像有主播後，用 3.3 節介紹的方法就能找出主播出現的時段，進而做新聞故事切割，本節所做的實驗就是要驗證所提的方法是否能正確的找出所有的主播段。一個主播段有兩個時間斷點，就是開頭與結尾，評量的方式是誤差



在一秒以內算正確，因為本論文是以每秒一張的速率擷取新聞影像。例如有一個主播時段的正確答案是 5~15 秒，而程式認為是 4~17 秒，則算對一個斷點，錯一個斷點。

表 4-7 是 4.1 節中所用的 5 個小時的新聞資料主播段偵測結果，第一欄是真正有多少主播斷點，第二欄是程式偵測到的主播斷點數，第三欄是正確偵測到的主播斷點，第四欄是正確率，第五欄是召回率。從表 4-7 來看，本論文所提的方法確實能找出主播段。

表 4-7 主播段偵測的實驗結果

| 電視台 | 真正的主播斷點數 | 程式偵測到的主播斷點數 | 正確偵測到的主播斷點數 | 正確率  | 召回率  |
|-----|----------|-------------|-------------|------|------|
| 東森  | 62       | 62          | 62          | 100% | 100% |
| 中天  | 48       | 48          | 46          | 96%  | 96%  |
| 民視  | 62       | 64          | 59          | 92%  | 95%  |
| 華視  | 50       | 56          | 50          | 89%  | 100% |
| 三立  | 64       | 62          | 56          | 90%  | 88%  |
| 平均  |          |             |             | 93%  | 96%  |

剛剛所提的主播段偵測的評量標準是以誤差在 1 秒以內算正確，不過對於一則新聞故事影片來說，多差個幾秒也能看到主要的新聞故事；另外就是主播播報的結束時間不會影響新聞故事的切割，這點從新聞故事的結構就能看出。表 4-8 就是綜合考慮這兩點所做的主播段偵測實驗結果。第一欄是誤差標準，第二欄是考慮主播段的開始與結束時間所得到的正確率與召回率，第三欄是只考慮主播段的開始時間所得到的正確率與召回率。表中每個正確率與召回率都是 5 個小時資料的平均。

表 4-8 主播段偵測的實驗結果(考慮各種評量標準)

| 誤差標準 | 同時考慮主播段的<br>起始與結束時間 |       | 只考慮主播段<br>的起始時間 |       |
|------|---------------------|-------|-----------------|-------|
|      | 正確率                 | 召回率   | 正確率             | 召回率   |
| 1 秒  | 93.5%               | 95.7% | 94.5%           | 96.6% |
| 2 秒  | 94.3%               | 96.4% | 94.5%           | 96.6% |
| 3 秒  | 94.3%               | 96.4% | 94.5%           | 96.6% |
| 4 秒  | 94.3%               | 96.4% | 95.1%           | 97.3% |
| 5 秒  | 94.3%               | 96.4% | 95.1%           | 97.3% |

### 4.3 新聞故事切割系統應用實驗

本節將論文中所提的新聞故事切割系統整合至本實驗室已有的一套新聞系統中，並以兩個星期的東森晚間新聞資料來做驗證，時間是 2007 年 6 月中到 6 月底。表 4-9 是主播影像偵測實驗結果，第一欄是日期，第二欄是真正的主播影像張數，第三欄是程式偵測到的主播影像張數，第四欄是正確偵測到的主播影像張數，第五欄是正確率，第六欄是召回率。表 4-10 是偵測主播段的實驗結果，第一欄是日期，其他欄位的意義與表 4-7 相對應的欄位相同。表 4-11 與表 4-8 相對應的欄位意義相同，唯一不同的地方只有實驗資料。

表 4-9 東森新聞主播影像偵測實驗結果

| 日期       | 真正的主播<br>影像張數 | 程式偵測到的<br>主播影像張數 | 正確偵測到的<br>主播影像張數 | 正確率 | 召回率 |
|----------|---------------|------------------|------------------|-----|-----|
| 20070629 | 651           | 839              | 644              | 77% | 99% |
| 20070628 | 667           | 837              | 659              | 79% | 99% |
| 20070627 | 656           | 724              | 629              | 87% | 95% |
| 20070625 | 612           | 810              | 606              | 75% | 99% |
| 20070624 | 582           | 678              | 571              | 84% | 98% |
| 20070623 | 593           | 612              | 585              | 95% | 98% |
| 20070622 | 558           | 791              | 554              | 70% | 99% |
| 20070621 | 531           | 565              | 524              | 92% | 98% |
| 20070620 | 611           | 773              | 605              | 78% | 99% |
| 20070619 | 620           | 783              | 609              | 78% | 98% |
| 20070618 | 606           | 742              | 600              | 81% | 99% |
| 20070617 | 662           | 799              | 658              | 82% | 99% |
| 20070616 | 505           | 512              | 488              | 95% | 97% |
| 20070615 | 681           | 772              | 665              | 86% | 97% |
| 平均       |               |                  |                  | 83% | 98% |

表 4-10 東森新聞主播段偵測實驗結果

| 日期       | 真正的主播斷點數 | 程式偵測到的<br>主播斷點數 | 正確偵測到的<br>主播斷點數 | 正確率   | 召回率   |
|----------|----------|-----------------|-----------------|-------|-------|
| 20070629 | 60       | 76              | 54              | 71%   | 90%   |
| 20070628 | 62       | 74              | 53              | 72%   | 85%   |
| 20070627 | 58       | 66              | 53              | 80%   | 91%   |
| 20070625 | 54       | 68              | 46              | 68%   | 85%   |
| 20070624 | 46       | 62              | 45              | 73%   | 98%   |
| 20070623 | 58       | 64              | 56              | 88%   | 97%   |
| 20070622 | 56       | 76              | 53              | 70%   | 95%   |
| 20070621 | 58       | 62              | 55              | 89%   | 95%   |
| 20070620 | 56       | 70              | 52              | 74%   | 93%   |
| 20070619 | 58       | 76              | 52              | 68%   | 90%   |
| 20070618 | 56       | 66              | 50              | 76%   | 89%   |
| 20070617 | 56       | 70              | 51              | 73%   | 91%   |
| 20070616 | 58       | 58              | 57              | 98%   | 98%   |
| 20070615 | 50       | 62              | 49              | 79%   | 98%   |
| 平均       |          |                 |                 | 76.9% | 92.5% |


表 4-11 東森新聞主播段偵測實驗結果(考慮各種評量標準)

| 誤差標準 | 同時考慮主播段的<br>起始與結束時間 |       | 只考慮主播段<br>的起始時間 |       |
|------|---------------------|-------|-----------------|-------|
|      | 正確率                 | 召回率   | 正確率             | 召回率   |
| 1 秒  | 76.9%               | 92.5% | 75.6%           | 90.8% |
| 2 秒  | 79.7%               | 95.8% | 79.3%           | 95.2% |
| 3 秒  | 81.1%               | 97.4% | 80.9%           | 97.3% |
| 4 秒  | 81.6%               | 98.7% | 81.5%           | 98.0% |
| 5 秒  | 81.8%               | 98.3% | 81.7%           | 98.2% |

## 第 5 章

# 結論與未來展望

### 5.1 結論



本文提出了以人臉偵測為基礎的偵測新聞主播方法來進行新聞故事切割，由於使用的是視訊上的特徵所以不會像【2】一樣會受到背景音樂的干擾；且因為只偵測影像中的人臉區域所以不會像【1】一般容易受到複雜背景的干擾。此外本方法是透過主播影像出現最頻繁的假設來偵測主播報導新聞的時間，所以不需要針對每個主播調整模型。

我們利用本方法實作了一個能自動切割東森新聞的系統，實作出的系統證實了本方法確實的切割出了可接受的新聞段落。

### 5.2 未來展望

在本論文的研究與實驗和應用中，發現有數個主題是我們可以繼續改進的重點，在此說明如下：

1. 本論文是以每秒一張的速率抽取影像，所以判斷主播出現時間的最小單位是

秒，不過影像檔每秒有 29.97 張，若能以跟影像播放相同的速率抽取影像來偵測主播的話，就能提高判斷主播出現時間的精確度。

2. 本論文實作的人臉偵測與人臉追蹤所花的時間比一般論文所寫的時間多，可能的原因是在人臉偵測的過程中多了影像分段這個步驟；還有在 3.1.2 節所提的人臉追蹤在搜尋區域中要比對完所有的位置，從中選擇一個最好的位置，這個步驟也是會花許多時間，若能在不失精確度的情形下減少時間的花費，則 1. 中所提的建議就能實現。
3. 本論文只取衣服顏色當成每個人的特徵，這步驟的風險在於萬一有其他人穿的衣服顏色與主播所穿的衣服顏色相同，則此人也會被認為是主播，若能找到其他更有分別性的特徵則可以大大提升主播段辨識的正確率。



## 參考文獻

- 【1】 Xinbo Gao and Xiaoou Tang, “Unsupervised Video-Shot Segmentation and Model-Free Anchorperson Detection for News Video Story Parsing,” IEEE Trans. Circuits and System for Video Technology, pp. 756-776, Sep. 2002
- 【2】 鄭士賢, “Model-based learning for Gaussian Mixture Model and its application on Speaker Identification,” 國立交通大學, 資訊工程研究所碩士論文, 民國九十一年
- 【3】 Ming-Hsuan Yang, David J. Kriegman and Narendra Ahuja, “Detecting Faces in Images: A Survey,” IEEE Tran. Pattern Analysis and Machine Intelligence, vol. 24, Issue 1, pp. 34-58, Jan. 2002
- 【4】 C. Kotropoulos and I. Pitas, “Rule-Based Face Detection in Frontal Views,” Proc. Int’ l Conf. Acoustics, Speech and Signal Processing, vol. 4, pp. 2537-2540, 1997
- 【5】 H. P. Graf, T. Chen, E. Petajan and E. Cosatto, “Locating Faces and Facial Parts,” Proc. First Int’ l. Workshop Automatic Face and Gesture Recognition, pp. 41-46, 1995
- 【6】 T. S. Jebara and A. Pentland, “Parameterized Structure from Motion for 3D Adaptive Feedback Tracking of Faces,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 144-150. 1997
- 【7】 H. Wang and S.-F. of Chang, “A Highly Efficient System for Automatic Face Region Detection in MPEG Video,” IEEE Trans. Circuits and System for Video Technology, vol. 7, no. 4, pp. 615-628, 1997
- 【8】 D. Saxe and R. Foulds, “Toward Robust Skin Identification in Video Images,” Proc. Second Int’ l Conf. Automatic Face and Gesture Recognition, pp. 379-384, 1996
- 【9】 Y. Dai and Y. Nakano, “Extraction for Facial Images from Complex



Background Using Color Information and SGLD Matrices,” Proc. IEEE Conf. Computer Vision & Image Processing, vol. 1, pp. 137-141 Oct 1994

- 【10】 M.-H. Yang and N. Ahuja, “Detecting Human Faces in Color Images,” Proc. IEEE Int’l Conf. Image Processing, vol. 1, pp. 127-130, 1998
- 【11】 D. Chai and K.N. Ngan, “Locating Facial Region of a Head-and-Shoulders Color Images,” Proc. Third Int’l Conf. Automatic Face and Gesture Recognition, pp. 124-129, 1998
- 【12】 J.L. Crowley and J.M. Bedrune, “Integration and Control of Reactive Visual Processes,” Proc. Third European Conf. Computer Vision, Vol. 2, pp. 47-58, 1994
- 【13】 I. Cai, A. Goshtasby and C. Yu, “Detecting Human Faces in Color Images,” Proc. 1998 Int’l Workshop Multi-Media Database Management Systems, pp. 124-131, 1998
- 【14】 陳鍛生和劉政凱, “膚色檢測技術綜述,” 計算機學報, Chinese Journal of Computers, 02期, 2006
- 【15】 Michael J. Jones and James M. Rehg, “Statistical Color Models with Application to Skin Detection,” IEEE Computer Society Conference, Computer Vision and Pattern Recognition, vol. 1, pp. 23-25. June, 1999
- 【16】 Tze-Yin Chow, Kin-Man Lam and Kwok-Wai Wong, “Efficient color face detection algorithm under different lighting conditions,” Journal of Electronic Imaging, vol 15, pp. Jan, 2006
- 【17】 Dorin Comaniciu and Peter Meer, “Mean Shift: A Robust Approach Toward Feature Space Analysis,” IEEE Trans Pattern Analysis and Machine Intelligence, vol. 24, Issue. 5, pp. 603-619. May, 2002
- 【18】 D. Chetverikov and A. Lerch, “Multiresolution Face Detection,” Theoretical Foundations of Computer Vision, vol. 69, pp. 131-140,

1993

- 【19】** C.-C. Han, H.-Y.M. Liao, K.-C. Yu and L.-H. Chen, “Fast Face Detection via Morphology-Based Pre-Processing,” Proc. Ninth Int’ l Conf. Image Analysis and Processing, pp. 469-476, 1998
- 【20】** T. Sakai, M. Nagao and S. Fujibayashi, “Line Extraction and Pattern Detection in a Photograph,” Pattern Recognition, vol. 1, pp. 233-248, 1969
- 【21】** P. Sinha, “Object Recognition via Image Invariants: A Case Study,” Investigation Ophthalmology and Visual Science, vol. 35, no. 4, pp. 1735-1740, 1994
- 【22】** J. Miao, B. Yin, K. Wang, L. Shen and X. Chen, “A Hierarchical Multiscale and Multiangle System for Human Face Detection in Complex Background Using Gravity-Center Template,” Pattern Recognition, vol. 32, no. 7, pp. 1237-1248, 1999
- 【23】** M. Turk and A. Pentland, “Eigenfaces for Recognition,” J. Cognitive Neuroscience, vol. 141, pp. 245-250, 1991
- 【24】** H. Rowley, S. Baluja and T. Kanade, “Neural Network-Based Face Detection,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no.1, pp. 23-38, Jan. 1998
- 【25】** E. Osuna, R. Freund and F. Girosi, “Training Support Vector Machines: An Application to Face Detection,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 130-136, 1997
- 【26】** K-K. Sung and T. Poggio, “Example-Based Learning for View-Based Human Face Detection,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 39-51, Jan. 1998