

國立台灣師範大學
資訊工程研究所碩士論文

指導教授：方瓊瑤 博士

以 Graph cut 演算法為基礎的連續影像人臉偵測
系統

A Video Face Detection Method Using the Graph
Cut Algorithm

研究生：郭俊麟 撰

中華民國一百零一年七月

摘要

本論文提出一套可在各種教室情境中運作的人臉偵測系統，偵測對象為教室中的多名學生，主要應用在自動教室觀察與記錄系統中。本研究採用顏色做為人臉偵測時的特徵，且利用 graph cut 技術做為人臉偵測時主要的方法。

以顏色為特徵的人臉偵測有著較不受頭部轉動和傾斜影響的優點，因為在頭部轉動和傾斜時，膚色依然在人臉中佔有一定比例的面積；至於眼睛、嘴巴和鼻子等其他人脸特徵在頭部轉動和傾斜時(尤其是轉動)在影像中較不穩定。這個顏色特徵的優點對於在教室中進行自動人臉偵測來觀察與記錄學生的行為有很大的幫助，因為在課堂中學生頭部的姿勢變化常常都是有意義的，如疲憊時打瞌睡、表示贊同時點頭或心不在焉時將人臉轉向他處等，而這些變化也往往是教學觀察者們(教師、研究人員)所關心的現象。因此，本系統若能夠在各種頭部姿勢狀態下做人臉偵測，就能夠更進一步地去分析這些姿勢變化和其所代表的意義。

利用顏色特徵來偵測人臉必須選擇一個適當的色彩空間，並且決定人臉的膚色在該色彩空間中的範圍。然而，這類作法常會遇到兩個問題，一是不同的光線以及人種需定義不同的膚色範圍，二是在教室中有許多物體顏色接近膚色(如原木色課桌椅)，會降低人臉偵測的正確率。針對第一個問題，本研究提出一個動態的膚色範圍定義方式；而為了解決第二個問題，本研究提出一個穩定的方法在影像中擷取前景(即學生的部分)。此方法結合單點建模與 graph cut 的技術，可以得到完整不破碎的前景，在前景的範圍內擷取膚色，避免類膚色背景的干擾。

另一方面，利用膚色在 Hue 色彩空間中高度集中的特質，本研究再次以 graph cut 技術優化膚色區域的偵測結果，統計收集到的膚色像素、動態更新膚色範圍，以提高偵測的穩定性。

在實驗時，本研究架設單一攝影機來擷取影像，每張影像中均包含 4~6 位學

生。本研究假設初始教室沒有學生，系統首先進行背景建置，待學生進入教室，系統偵測到影像中有前景出現後，便會開始進行人臉偵測。實驗結果顯示，本研究提出的人臉偵測技術，較不受各種頭部轉動和傾斜角度之影響，並且能夠在低解析度影像下，維持高準確率。

關鍵字：圖分割、膚色偵測、人臉偵測、教室、前景擷取。

Abstract

We propose a face detection system which is used in classrooms with various environments. The targets are several students in class, whose behaviors are to be observed and recorded in a classroom observation system. The feature chosen for detection is “color”, and the kernel method is the well-known graph cut algorithm.

Color feature is robust against head pose changing because the area of skin regions changes little during head rotating or tilting, while other features like eyes, nose and mouth are unstable under these conditions. This character is useful for observing students’ behaviors in class since it’s usually meaningful when a student change his head pose. For example, one may doze off if he is tired, nod his head to show his agreement, or turn his head out when he is distracted. As a result, if we can perform face detection under various head poses, then the results can be used to detect such behaviors mentioned above for further researches.

To detect faces with color feature, we must choose a proper color space first, and determine the range of “skin color”. However, this kind of methods has two problems. First, the range changes with different lighting conditions and human races. Second, there are many non-human objects with skin like color which affects on the precision a lot. For the first problem, we propose a dynamic learning scheme to change the skin color range frame by frame. And to solve the second problem, we propose a robust background subtraction method to eliminate non-human objects. This method combining pixel based background modeling and the graph cut algorithm extracts complete foreground region from the input frame and thus avoids the effect of skin like background pixels.

On the other side, since the hue values of skin color pixels are distributed concentratedly in hue color band, we apply graph cut to improve the result of skin color detection, and then collect the skin pixels for learning new skin color range.

In the experiments, we set up a single camera, and there are 4~6 students in the image. Assuming an empty classroom in the beginning, the system constructs the background model first. Then, when objects appear in the image, the system will start to perform face detection. According to the experimental result, the technique proposed is robust under various head poses, and retain high precision in the low resolution images.

Keywords: graph cut, skin color detection, face detection, classroom, background subtraction.

誌謝

口試後一個月，定稿在即，漫長艱辛的研究工作總算暫時告一段落。碩士生活遠比想像中辛苦，而自己的能力與意志也比想像中薄弱許多，好在一切終能落幕，所有負面的感覺也將轉化回正向力量陪伴我往下一個階段前進。

這些日子裡，最需要感謝的有兩人，就是我的指導教授方瓊瑤教授和我的同學梁瑞宏。方老師雖然總說自己放牛吃草，但其實她對於學生的要求非常細膩，這對於我這種做事粗枝大葉，不拘小節的人來說著實受到相當大的震撼且難以適應。不過也就在這亦步亦趨的過程中，更了解自己個性上的盲點和一些待人處世應有的態度。雖然改進的幅度可能還不大，卻已產生了許多更堅定的想法和方向。並且，方老師本身自律甚嚴並且為人正派，讓人覺得十分欽佩，而這也是自己需要繼續努力的地方。

感謝梁瑞宏支援了我研究和課業上大大小小的瑣事，並且總是與我同在，不論是一起衝研究，或是一起吃美食、玩電玩發洩研究上產生的眾多不快。如果不是他的話，想必我在師大需要學的教訓、繳的學費還會要更多。而因為自己的關係，拖延了兩人口試的日期，在此便以祝福和感謝代替抱歉，希望他往後人生的路途能夠符合心中所嚮。

此外，最重要的就是感謝口試委員陳世旺老師和梁祐銘老師，在口試過程給予的許多建議以及包容，讓我順利通過。感謝實驗室的學弟妹陪伴我們度過這一年並且永遠給予我們支持；感謝家人默默的關心並在我需要的時候永遠敞開大門歡迎；感謝樂團的朋友們給予我前進的動力；最後，希望在下個人生階段結束時，能夠毫不遲疑地感謝自己的努力。

郭俊麟 2012. 8. 27

目錄

摘要.....	I
Abstract	III
誌謝.....	V
目錄.....	VI
圖目錄.....	VIII
表目錄.....	XII
第一章 緒論	1
1.1 研究背景與目的	1
1.2 文獻探討	5
1.2.1 前景擷取方法	5
1.2.2 Graph cut 演算法	10
1.2.3 膚色偵測	11
1.3 系統架構	15
第二章 影像前處理	18
2.1 差值影像	18
2.2 對 difference histogram 前景擷取演算法的改良	22
第三章 Graph cut 前景擷取技術	27
3.1 Graph cut 技術	27
3.1.1 名詞定義	27
3.1.2 影像標記(labeling)和成本函數(cost function)	28
3.1.3 最小化成本函數	30
3.1.4 相關定理證明	32
3.2 應用於影片前景擷取	34
第四章 Dynamic Graph Cut.....	39
4.1 Max-flow min-cut 演算法的時間複雜度	40

4.1.1 Ford-Fulkerson 與 residual graph.....	40
4.1.2 Graph cut 和 maximum flow 演算法.....	42
4.2 Dynamic graph cut	44
4.2.1 增加新的 hard constraint	44
4.2.2 Graph 再參數化(reparameterization).....	46
第五章 膚色區域偵測與人臉區域選取.....	52
5.1 膚色區域偵測.....	52
5.2 人臉區域選取.....	58
第六章 實驗結果與討論.....	61
6.1 去背景實驗結果.....	61
圖 6.18 去背景失敗之例子.....	72
6.2 膚色區域偵測實驗結果.....	72
6.3 人臉區域選取之結果.....	82
第七章 結論與未來工作.....	89
7.1 結論.....	89
7.2 未來工作.....	90
參考文獻.....	91

圖目錄

圖 1.1 以 Cr 屬於[133 173]、Cb 屬於[77 127]作為膚色範圍的失敗範例	5
圖 1.2 Temporal difference 示意圖	7
圖 1.3 單一位置像素 intensity 值的 histogram	7
圖 1.5 常見的兩種 HSV 色彩空間圖示	13
圖 1.6 膚色像素的 HSV 值在各種照明情況下之分布圖	15
圖 1.7 系統流程圖	16
圖 2.1 Difference histogram 和 turning point 示意圖	18
圖 2.2 Chiu 提出的前景擷取流程圖(Chiu, 2010)	19
圖 2.3 以差值影像進行前景擷取的結果	20
圖 2.4 Difference histogram 之範例	20
圖 2.5 實作 difference histogram 前景擷取的失敗範例	21
圖 2.6 Chiu 之論文所使用的影片範例(Chiu, 2010)	22
圖 2.7 背景更新失誤之範例	22
圖 2.8 改良後的 difference histogram 前景擷取流程圖	24
圖 2.9 取得最佳門檻值之示意圖	25
圖 2.10 改良 Chui 演算法之後前景擷取的結果	26
圖 3.1 Graph cut 範例	29
圖 3.2 由 I 所建立的 graph 之範例	31
圖 3.3 Feasible cut 之範例	33
圖 3.4 Hard constraint 賦予示意圖	35
圖 3.5 由 integral image 計算矩形內部二值化後其值為 1 之像素總和	36
圖 3.6 Graph cut 流程圖	37
圖 3.7 Graph cut 擷取前景的比較範例	38
圖 4.1 Residual graph 建構示意圖	41
圖 4.2 Boykov and Kolmogorov's algorithm 示意圖	43

圖 4.3 Boykov and Kolmogorov's algorithm 與其他 maximum flow 演算法處理三種影像的時間比較.....	44
圖 4.4 在 residual graph 上修改 t-link 容量之示意圖.....	47
圖 4.5 在 residual graph 上修改 n-link 容量之示意圖.....	48
圖 4.6 Dynamic graph cut 演算法的加速情形.....	49
圖 4.7 Temporary difference 之範例.....	50
圖 4.8 使用 dynamic graph cut 演算法且增加 temporary difference 判斷機制的實驗結果.....	51
圖 5.1 Hue channel 影像範例.....	53
圖 5.2 以 Zhang <i>et al.</i> 所提出之固定範圍偵測膚色之實驗範例.....	53
圖 5.3 經過 dynamic graph cut 增強之膚色區域.....	54
圖 5.4 各種教室和照明狀況下膚色之 hue channel 之 intensity 值分布.....	55
圖 5.5 在各種教室背景下膚色的 intensity 值區間.....	57
圖 5.6 Harris corner 在人臉膚色區域上的反應.....	59
圖 6.1 實驗所使用五間教室之背景圖.....	61
圖 6.2 影片 265 之輸入影像.....	64
圖 6.3 影片 265 之前景擷取實驗結果.....	64
圖 6.4 影片 266 之輸入影像.....	65
圖 6.5 影片 266 之前景擷取實驗結果.....	65
圖 6.6 影片 288 下之輸入影像.....	66
圖 6.7 影片 288 之前景擷取實驗結果.....	66
圖 6.8 影片 292 下之輸入影像.....	67
圖 6.9 影片 292 之前景擷取實驗結果.....	67
圖 6.10 影片 301 之輸入影像.....	68
圖 6.11 影片 301 之前景擷取實驗結果.....	68
圖 6.12 影片 303 之輸入影像.....	69
圖 6.13 影片 303 之前景擷取實驗結果.....	69
圖 6.14 影片 307 下之輸入影像.....	70

圖 6.15 影片 307 之前景擷取實驗結果	70
圖 6.16 影片 309 之輸入影像	71
圖 6.17 影片 309 之實驗結果	71
圖 6.18 去背景失敗之例子	72
圖 6.19 影片 265 之輸入影像	75
圖 6.20 影片 265 之膚色區域偵測結果	75
圖 6.21 影片 266 之輸入影像	76
圖 6.22 影片 266 之膚色區域偵測實驗結果	76
圖 6.23 影片 288 之輸入影像	77
圖 6.24 影片 288 之膚色區域偵測結果	77
圖 6.25 影片 292 之輸入影像	78
圖 6.26 影片 292 之膚色區域偵測結果	78
圖 6.27 影片 301 之輸入影像	79
圖 6.28 影片 301 之膚色區域偵測結果	79
圖 6.29 影片 303 之輸入影像	80
圖 6.30 影片 303 之膚色區域偵測結果	80
圖 6.31 影片 307 之輸入影像	81
圖 6.32 影片 307 之膚色區域偵測結果	81
圖 6.33 影片 288 之輸入影像	84
圖 6.34 影片 288 之人臉偵測結果	84
圖 6.35 影片 292 之輸入影像	85
圖 6.36 影片 292 之人臉偵測結果	85
圖 6.37 影片 301 之輸入影像	86
圖 6.38 影片 301 之人臉偵測結果	86
圖 6.39 影片 303 之輸入影像	87
圖 6.40 影片 303 之人臉偵測結果	87
圖 6.41 影片 309 之輸入影像	88

圖 6.42 影片 309 之人臉偵測結果 88

表目錄

表 1.1 FIAS 的口語互動類別	2
表 3.1 依據各 link 賦予適當的權重	32
表 4.1 使用者增添 hard constraint 前後 p 的 t-link 權重值	45
表 4.2 以“增加流量”為基礎所定義的 link 權重值	45
表 6.1 前景擷取之實驗結果	62
表 6.2 膚色區域偵測之實驗結果	73
表 6.3 人臉偵測之實驗結果	82

第一章 緒論

1.1 研究背景與目的

教室觀察為教育研究中一個重要的過程，其係指針對教室內外的情境脈絡、教師的各種教學行為和方法、學生的學習情形及教材的使用等方面，進行系統化的描述、記錄、解釋、分析及預測的歷程。陳美玉(陳美玉，2012)便指出，教室觀察是教師獲得實踐知識的重要來源，也是教師用以蒐集學生資料、分析教學方法的有效性，以及瞭解教與學行為的基本途徑。

一般而言，教室觀察大致可分為兩個取向：一為「量化觀察法」：以系統的規則、紀錄方式來進行的系統觀察；另一為「質化觀察法」：結合人種誌的觀察技巧(陳美玉，1998；Croll，1986)。以研究的角度來說，量化觀察法由於其客觀程度較高，觀察者的技術門檻較低，為研究的主流。不過，在某些情況下，基於可行性以及對於問題的定義，質化觀察法有其存在之必要性，或者應搭配量化觀察法，統整兩者，以達兼顧廣度與深度的研究成果。

至於教室觀察到底應該觀察什麼，則是眾說紛紜。Borich(Borich，1994)提出八項重要規準，包括：學習氣氛、班級經營、課程清晰、教學多樣、任務取向、學習參與、學習成效、成就表現與高層次的思考；王淑俐(王淑俐，1998)則認為國內的教室觀察項目應包含：1. 教學計畫；2. 班級管理；3. 教學呈現；4. 溝通技巧；5. 教室氣氛；6. 教學組織與策略；7. 教學評量；8. 學習環境；9. 教學改善與專業成長。實務方面，世界各地的學者和教育人員亦制定了各式的 check

list，如南華大學教學現場教室觀察施行辦法(民國 96 年 12 月 19 日本校 96 學年度第 1 學期第 2 次教務會議修訂通過)就提到教學現場教室觀察應該包含：

1. 基本資訊：現場學生人數、選用教材及教學方式等。
2. 教學方法：教學主題的詮釋與掌握、音量適中與清晰度及導引學生產生學習動機等。

此外，明尼蘇達大學也將該校制定的 classroom observation checklist 放在網頁中供師生下載，顯見教室觀察對現今的教育來說已是不可或缺。

在教室觀察的眾多項目當中，“師生互動”可謂其中一個的核心觀察項目。教育本身即是老師與學生交互作用之後的產物，扣除一些客觀條件(如採光、學生人數、麥克風音量及回音情況等)之後，教師與學生的互動情形便主導了整個教學過程的結果。關於師生互動的研究，有一個著名的分析系統是 FIAS(表 1.1)，它所要分析的是教室情境中師生雙方所說的語言。此一觀察系統的目的在於運用一套代碼系統(coding system)，記錄在教室中的師生口語互動情形，以分析教學行為，進而幫助教師改進教學行為。這套系統將師生互動進行抽象的分類，讓觀察者能夠以極少的主觀依據去做紀錄，可以想見，由此分析系統所得到的資料，將具備優良的穩定性與一致性，不因施測者不同而劇烈地影響觀察結果。

表 1.1 FIAS 的口語互動類別

教師 話語	學生 驅 動	<p>1.接納學生的情感：以一種不具威脅性的方式，接納及澄清學生的態度或情感語氣。學生的情感可能是正向的，也可能是負向的。這一類也包括預測或回想學生的情感。</p> <p>2.稱讚或鼓勵：稱讚或鼓勵學生的動作或行為。這一類也包括紓解緊張但不傷人的笑話；點頭、或說「嗯(Um hm)」、或說「繼續下去(Go on)」等。</p> <p>3.接受或利用學生的想法：澄清、擴大或發展學生所提出的意見或想法。這一類包括教師延伸學生的意見或想法，但是當老師呈現較多自己的意見或想法時，則屬於第五類。</p>
		<p>4.問問題：以教師的意見或想法為基礎，詢問學生有關內容或步驟的問題，並期待學生回答。</p>

	教師主動	5.演講：就內容或步驟提供事實或見解；表達教師自己的觀念，提出教師自己的解釋，或者引述某位權威者（而非學生）的看法。 6.指示：指示、指令或命令，此類行為具有期望學生服從的功能。 7.批評學生或維護權威：陳述的語句內容為企圖改變學生的行為，從不可接受的型態轉變為可接受的型態；責罵學生；說明教師為何採取這種作為；極端地自我參照。
學生話語	教師驅動	8.學生話語—教師驅動：學生為了回應教師所講的話。教師指定學生答問，或是引發學生說話，或是建構對話情境。學生自由表達自己的想法是受到限制的。
	學生主動	9.學生話語—學生主動：學生主動開啟對話。表達自己的想法；引起新的話題；自由地闡述自己的見解和思路，像是提出具思考性的問題；超越既有的架構。
靜止		10.安靜或混亂：暫時停頓、短時間的安靜或混亂，以致於觀察者無法瞭解師生之間的溝通。

師生互動情形除了可以透過語言來觀察，亦可由一些肢體動作取得資訊。其中，頭部姿勢的改變便是一類可以用來分析師生互動的肢體動作。例如點頭可能代表學生同意老師的說法與表示理解；低頭與轉頭可能代表學生心不在焉或是感到疲倦等。另外，頭部姿勢改變大多伴隨著視線的改变，而這也透漏出許多師生互動的情況，例如學生的視線有無面對老師能夠用來判斷該生是否有將注意力集中於老師的授課；在師生對話時，兩者的視線的交集情形也夠用來分析雙方的交流情況。總而言之，頭部的各種姿勢變化包含了許多關於師生互動情形的資訊，為一進行師生互動觀察的良好特徵。

本研究結合教室觀察原理與影像處理技術，希望能夠做到教室觀察的自動化，而教室觀察的內容則放在上段所提及的頭部姿勢變化分析方面。不過，由於人臉在影像中的特徵較為明顯，而且偵測到人臉的轉動就等於偵測到頭部轉動，因此本研究將主要的研究方向定位於教室中的人臉偵測以及追蹤，用以進一步偵測人臉轉動。

人臉偵測在影像處理領域之中的研究已有相當歷史，當中也不乏穩定快速的技術，只是大部分的人臉偵測技術都無法適應任意頭部姿勢變化。原因是前述方法採用的特徵在影像上多半會受五官造成的光影分布影響，所以在各種頭部姿勢變化之下的穩定性很低。以 Haar feature 為例，此特徵主要與邊緣(edge)在人臉上的分布情況有關，而正臉與側臉的邊緣分布情況又十分不同，因此被訓練來偵測正臉的 Haar feature 就無法正確地偵測出側臉。有另一類技術不是用低階的邊緣特徵，而是希望直接把人臉的眼睛、鼻子和嘴巴等高階特徵偵測出來，以決定人臉的位置及大小。但這類方法所使用的影像中大多只能處理單一人臉，並且臉部的大小不能夠太小，否則會因為特徵不夠明顯而大幅影響偵測正確率，這對於本研究的應用——在單一畫面中偵測多個人臉並不適用。

為了克服這個問題，本研究使用顏色作為人臉偵測的主要特徵，系統先在畫面中偵測膚色區域，然後篩選出屬於人臉的膚色區域。膚色在各種色彩空間中的分布範圍已被研究許久，而目前較常使用而且表現較為穩定的色彩空間為 HSV 色彩空間和 YCbCr 色彩空間。膚色在此二色彩空間中的分布較為集中，也有許多論文經由大量統計影像中的膚色像素，而定出膚色在此二者中的像素值範圍。

不過，顏色特徵並非完美無缺，由於亮度值容易受到環境光線影響，在不同教室以及不同的照明狀況之下，膚色在各維度的 intensity 值範圍不盡然相同，若以同一個膚色範圍在不同教室中偵測，必然無法獲得穩定的正確率。

更甚者，由於教室中本身就有許多物體的顏色接近膚色，如粉色窗簾和原木製桌椅等，若是單純於影像中使用某個膚色範圍來偵測膚色區域，這類物體相當容易干擾偵測的結果(圖 1.1)。圖 1.1 呈現以 Cr 屬於[133 173]、Cb 屬於[77 127]作為膚色範圍的失敗範例。其中左邊為輸入之原始影像，右邊為膚色擷取後的結果。由上排的例子可看出因照明情況偏白而使得膚色擷取結果破碎；下排例子則是顯示當背景中有許多膚色物體時會造成擷取範圍過大。因此，本研究致力於如何在畫面中取得完整的膚色區域，並且能夠在各種教室和照明狀況下維持正確率。



圖 1.1 以 Cr 屬於[133 173]、Cb 屬於[77 127]作為膚色範圍的失敗範例。左邊為輸入影像，右邊為膚色擷取後的結果。上排：因照明情況偏白而使得膚色偵測結果破碎；下排：由於背景中有許多膚色物體而造成擷取範圍過大。

(資料來源：<http://vision.caltech.edu>)

1.2 文獻探討

本研究分成三個部分：第一是初步的前景擷取；第二是以 graph cut 演算法強化前景擷取的結果，讓前景區域更為連續完整，並以 dynamic graph cut 增進其速度；第三是在前景區域中偵測膚色區塊，並且由多個膚色區塊中選取屬於人臉的部分。以下文獻探討也會以此順序排列之。

1.2.1 前景擷取方法

前景擷取時較直觀的想法是計算兩張連續影像之間對應位置像素的 intensity 差值(temporal difference)。由於前景物體在影像中往往是處於移動狀態，一個瞬間由背景變化為前景的像素，其 intensity 值必定會有相當程度的變動，可以藉由設定這個差值的門檻值偵測前景。假設 f_t 為時間 t 的輸入影像， $f_{t-1}(i, j)$ 與 $f_t(i, j)$

分別為位置 (i, j) 之像素在時間 $t-1$ 與 t 的 intensity 值，則前景的擷取公式如下。

$$|f_t(i, j) - f_{t-1}(i, j)| > T$$

其中， T 為一給定之門檻值。此公式考慮位置 (i, j) 之像素其前後兩張影像中之 intensity 差值絕對值，若該值夠大則將該像素視為前景。此方法的優點在於它可以適應各種環境的突發狀況(如照度改變)，但偵測區域破碎、無法偵測緩慢移動甚至是靜止的前景、輸出結果對於該門檻值的設定相當敏感等為其主要的缺點。如圖 1.2 所示，前後兩張影像相減後，因為雜訊很多，很難找到一個適當的門檻值來區分前景背景。圖 1.2(b) 為用 temporal difference 方法且取得適當門檻值之期望結果。而圖 1.2(c)、(d) 則分別顯示其門檻值設定過高與過低之結果。

另外有些研究者發展出建立一張靜止的背景以擷取前景的技術，改為取當前的輸入影像和此背景影像的差值作為分辨前景的依據。假設時間 t 之背景影像為 b_t ，則位置 (i, j) 像素被視為前景的條件可表為

$$|f_t(i, j) - b_t(i, j)| > T$$

T 為給定的門檻值。利用前景和背景的颜色差值去區分，可以有效解決前景移動程度不大，甚至靜止導致無法順利偵測前景的問題；前景擷取結果破碎的情況也可能改進。然而，此技術對於背景漸進改變這個現象較無抵抗能力，另外，輸出結果與門檻值的選取依然關係很大。簡而言之，背景影像如何建立，以及抵抗時間變換所造成的影響為此類方法首要面對的問題。

除了上述兩個基本技術之外，目前較為常用的前景擷取技術為背景機率模型建立。背景機率模型建立時，會對於影像中每一個像素，經由長時間的統計個別進行分析，進而建立背景機率模型。比較基本的做法是以 histogram 為主的 probability model(圖 1.3)。建好所有位置的模型之後，對於每一張輸入影像，拿取每一個位置的像素與該位置的模型做比對，依此判斷該像素值應屬於前景或背景。以機率模型代替單一像素值，可以有效避免一些因環境因素規律改變所產生的誤判，也較符合實際的狀況。

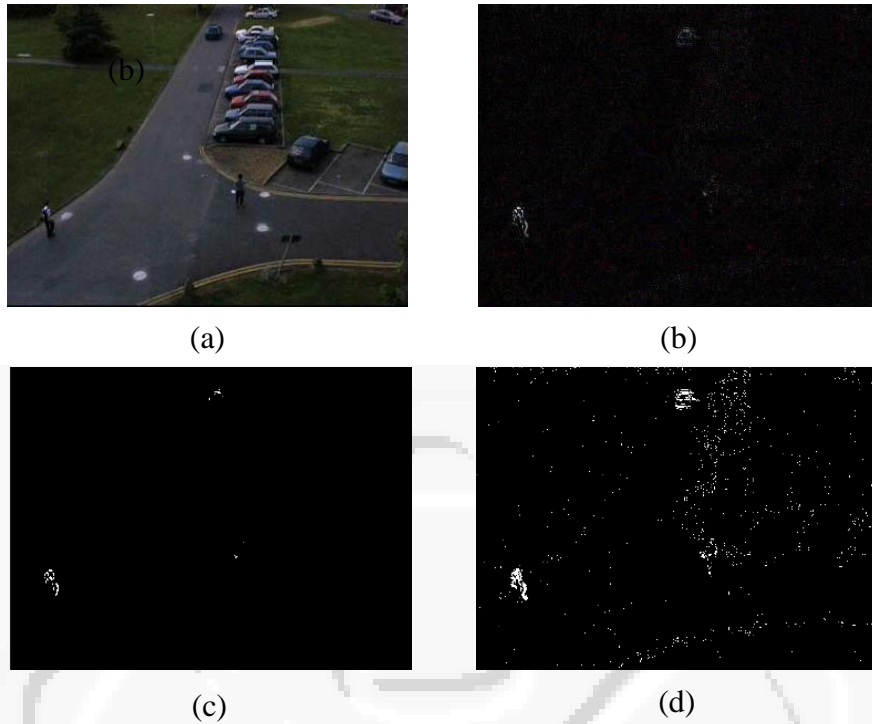


圖 1.2 Temporal difference 示意圖(a) 輸入影像(b) Temporal difference (c) thresholding too high(d) thresholding too low(Piccardi , 2004)

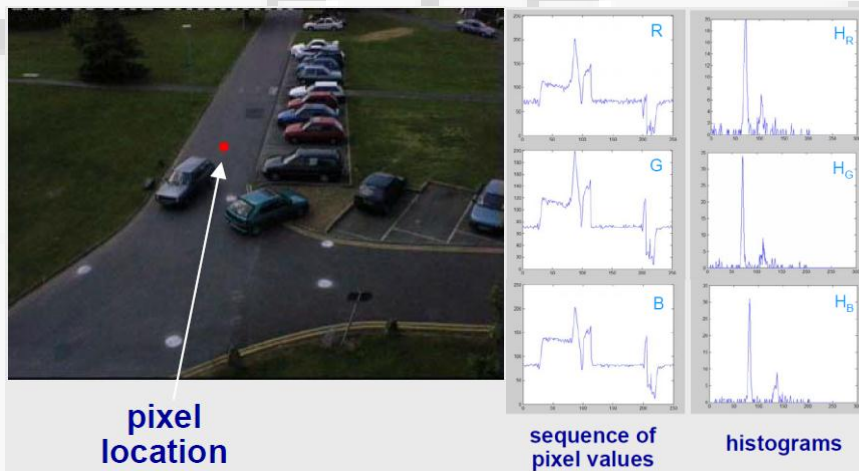


圖 1.3 單一位置像素 intensity 值的 histogram(Piccardi , 2004)

Mixture of Gaussian (以下簡稱 GMM) 的技術可以用來建立背景模型，它利用多個 Gaussian 模型 model 背景中每一個單一像點的 intensity 值，可以在規律性的背景變化(如晝夜變化) 下穩定的擷取前景。其缺點是該方法無法自動決定每一像素的最適當 Gaussian 函數個數。然而，由於其穩定且優越的結果，依然有大量的研究使用 GMM 為主的前景擷取方法(Stauffer , 2000)。

Elgammal *et al.*(Elgammal *et al.*, 2002) 以數張單純背景의影像為基礎，在影像中各個像素位置分別建立一個三維(即 R,G,B)的 Gaussian probability model，並以某像素屬於背景的機率值決定該像素是否屬於背景。此一技術的優點是能夠同時考慮 RGB，引進了這三個維度在機率上的相關性。若是將三個維度分別用一維建模，就只能視其為機率獨立，但這與現實狀況不太相符。不同於一維的 Gaussian distribution，多維 Gaussian distribution 的參數由原來的平均值和變異數改為平均向量 μ 和變異矩陣 Σ 。

Elgammal *et al.*(Elgammal *et al.*, 2002)假設一給定 GMM 中有 K 個 channels，每個 channel 皆為 Gaussian distribution。第 j 個 channel 其權重為 ω_j ，平均向量和變異矩陣分別為 μ_j 和 Σ_j ；若以 x_t 代表某像素的 intensity 值，則其分布函數可表示為

$$P_r(x_t) = \sum_{j=1}^K \frac{\omega_j}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (x_t - \mu_j)^T \Sigma_j^{-1} (x_t - \mu_j) \right\}$$

該篇作者在 02 年提出的另外一篇論文(Elgammal *et al.*, 2002)，考慮到並非所有的背景像素 intensity 值都符合 Gaussian distribution 的模型，而提出更一般的 kernel distribution 概念。此外，Elgammal 放棄前段所建立的三維模型，將三個維度分開建模之後以相乘的方式結合。假設此模型中每個像素除存有 N 個歷史向量，每個向量之維度為 d ，而第 j 個維度的 kernel function 為 K_{σ_j} ，其平均數為 0，變異數為 σ_j 。則對於一給定輸入向量，其背景之機率分布函數定義為

$$P_r(x_t) = \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^d K_{\sigma_j} \left(\frac{(x_t - x_i)_j}{\sigma_j} \right)$$

此方法雖然隱含了上段所提及的缺點，但機率獨立的假設除了在實作上可以提升速度之外(例如免除了 covariance 矩陣及其反矩陣的計算)，對於 noise 的容忍程度也會提高；在後續處理可以彌補此部分缺點的前提之下，機率獨立與否的假設

並不會嚴重影響最後的實驗結果。並且，該論文捨棄了 Gaussian 模型，而使用了更一般的 kernel 函數 K_{σ_j} ，如此能夠在真實情況不符合 Gaussian 模型的假設時降低錯誤率。

此外，這篇論文提出了以 normalized RGB 代替原來的 RGB 作為特徵，其對於光線變換的容忍程度有相當程度的提高，normalized RGB 的定義如下：

$$r = \frac{R}{R+G+B}; \quad g = \frac{G}{R+G+B}; \quad b = \frac{B}{R+G+B};$$

由這個定義來看，normalized RGB 將原來的三個特徵正規化到 [0, 1] 區間。因此，對於倍數和小幅平移式的光線變換都有一定的容忍程度。

Kae and Bow (Kae and Bow, 2001) 也利用三維的 Gaussian distribution 對各個位置像素的 intensity 分布情形建立模型，由於該方法不需要事先做建背景的动作，在該領域的研究中為一重大突破。他們假定單一位置的像素以背景出現的機率遠大於以前景出現的機率，Kae and Bow 以 K 個 Gaussian distribution $\eta(x; \theta_j)$ 來模擬像素的 intensity 值分布：

$$P(x_N) = \sum_{j=1}^K w_j \eta(x_N; \theta_j)$$

其中， θ_j 為每個 distribution 的參數，包含平均值和變異數， w_j 為各個 distribution 的權重值。根據上述假設，該論文設定一個門檻值 T ，將各個 Gaussian 函數以 w_j 的大小作排序並由最大值開始累加，當累加值超越 T ，便將這些計入累加的 Gaussian 函數作為背景的分佈函數。

$$B = \arg \min_b \sum_{j=1}^b w_j > T$$

此一技術用 GMM 建三維模型，相對細膩的表示了單一位置像素值的分布情況。在背景像素的 intensity 值所佔時間比例較大的基本假設之下，以一個有效率且不致偏頗的方式去估計背景的分佈。然而，該論文也提到，對於初始點為前景的像素，要收斂到完美的背景分布需要極長的時間，儘管該論文提出另一個 EM 的學

習方法來解決這一個問題，前景的分布情形依然是決定正確率的最大因素。

總而言之，單一像素的 intensity 值的背景模型建立為現今前景擷取的研究者主要的思考方向，但卻有幾個問題尚待解決：

1. 像素眾多，模型建立耗費大量空間與時間。
2. 此類技術皆假設，前景在影像中無論是所佔時間或所佔面積都比背景小很多，對於前景在影像中長時間靜止並佔有較大面積的情況，較不適用。
3. 單點建模著重於分辨單一位置的背景及前景像素分布，並未考慮像素與像素間的關係，這對於具平移可能性的應用潛藏危機。

在上述提及的三個問題之中，第一個問題可以用程式的平行化去解決，如 Cheng *et al.* (Cheng *et al.*, 2011)，該研究使用修正後的 SVM 作為建模的方法，用以抵抗背景隨時間的變化，並以 GPU CUDA library 平行化程式，兼顧 real-time 的效果。但第二和第三個問題，本質上並不是單點建模所能夠處理的情況。

1.2.2 Graph cut 演算法

在背景建立時，除了考慮單一位置像素不同時間的 intensity 值的垂直關係外，若能夠將橫向的相鄰像素之關係也加以考慮，就能夠改進單點建模的缺點。舉例來說，若某個像素的 intensity 值比較接近背景，但是其鄰近像素的 intensity 值卻比較接近前景，此時基於鄰近像素分類的一致性，將這個類似背景的像素視為前景是一個比較合理的結論。

以此概念為主的前景擷取方法也已經被提出，即 graph cut (Wu and Leahy, 1993) 演算法。該演算法以像素之 intensity 值做為特徵值 e 定義 energy function。基於最小化 energy function 的概念，可以在使用者提供某些 hard constraint (標記某些像素點為前景或背景) 之後 (圖 1.4)，以 max-flow min-cut 的演算法得到前景擷取的穩定結果。在圖 1.4(a)(b) 中，使用者在影像上以紅色劃記一些背景像素，以白色劃記某些前景像素之後，先以此建立機率模型，在考慮像素之間的位置關係，以取得完整的前景。

上述方法已經在單張影像中實現，在連續影像之中必然也有效果。然而，將

graph cut 技術擴展到連續影像上的主要議題有二：

- 一、如何用連續影像提供的自動的資訊取代單張影像所需要的手動的 hard constraint，也就是將系統由互動變為自動；
- 二、如何加速演算法使其能夠達到即時效果。

Ying *et al.* (Ying *et al.*, 2007)先將畫面切割成許多 blocks 之後，再經由類似 graph cut 的技術於影片中擷取前景。Graph cut 的優勢在於不需要每個像素位置皆儲存一個模型，大大降低記憶體佔有率。另外在該演算法中，顏色相近的鄰近像素之間有內聚力，會讓前景擷取結果較完整。不過，由於 graph cut 在實作上也需要用到 mixture of Gaussian 的模型，如此亦不能避免之前提及 GMM 的缺點。另外，這個技術對於背景與前景像素分布較接近的影像在理論上較無抵抗能力，而這卻是單點建模相對有優勢的地方，因為單點建模只考慮單一像素的 intensity 值分布情形，即使整體來看背景與前景色系相近，只要在單一位置能夠清楚劃分背景與前景即可。

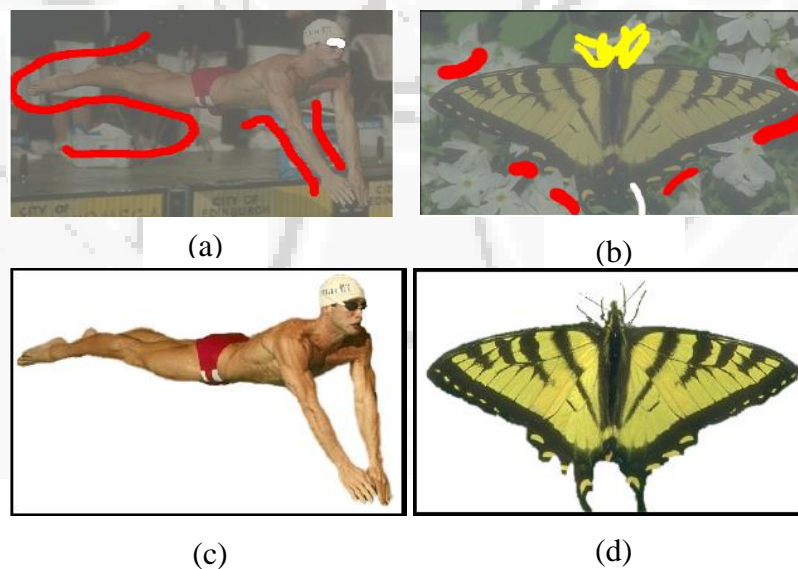


圖 1.4 Graph cut 前景擷取範例(a)(b)hard constraint(c)(d)前景擷取結果

1.2.3 膚色偵測

膚色偵測前須先決定一個色彩空間(color model)，並且決定膚色之 intensity 值在此色彩空間之中的分布範圍。色彩空間為一三維實數空間，用來將各種不同

的彩色向量化(即將不同顏色以不同的實數向量表示)。目前已被提出並且廣泛運用於各個領域的色彩空間有十幾種，例如 HSI、HSV、YUV、YCrCb 等。

最常用的色彩空間為 RGB 三維色彩空間，其中 R 為紅色(red)、G 為綠色(green)、B 為藍色(blue)。此三種顏色是肉眼所能感知最強烈的顏色，幾乎所有的顏色都能夠以此三色經由調整光線強度之後混合而得。在電腦硬體中，亦以此三色作為基本顏色，螢幕上配有此三種顏色的發光元件，並且將每種顏色各以 8bit(即範圍在 0~255 間的值)的數值區分強度，混合起來有 2^{24} 種組合(顏色)，又稱為 24bit 全彩。

同一顏色在各種不同的色彩空間中的表示向量其實是可以互相轉換的，但大部份的轉換公式都是經由 RGB 色彩空間來處理。許多研究發現膚色像素的表示向量在 HSV 和 YCrCb 兩種色彩空間之中具有高度集中的特質(Chai and Ngan, 1999) (Mahmoud, 2008)(Phung, 2002)，較利於界定膚色範圍。因此，以下將詳細介紹此兩種色彩空間，並且討論使用此二者做膚色偵測的相關文獻。

HSV 色彩空間的三個維度分別是 hue(色調)、saturation(色度)和 value(亮度)。其經由 RGB 色彩空間的轉換公式如下：

$$V = \max(R, G, B)$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)}$$

V 和 S 可直接由 R、G、B 三者的值求出，至於 H 則會因為 R、G、B 三者的最大值之不同而略有不同。

$$H = \begin{cases} \frac{G - B}{\max(R, G, B) - \min(R, G, B)} \times 60 & \text{if } \max(R, G, B) = R \\ \frac{B - R}{\max(R, G, B) - \min(R, G, B)} \times 60 + 120 & \text{if } \max(R, G, B) = G \\ \frac{R - G}{\max(R, G, B) - \min(R, G, B)} \times 60 + 240 & \text{if } \max(R, G, B) = B \end{cases}$$

上述 H 代表的是以可見光波長區分的各種純色，由波長最長的紅色，一直到波長最短的紫色。理論上會將 H 的範圍會在 $[0^\circ, 360^\circ]$ 之間，可以用一個色環

圖形來說明(如圖 1.5)。H 並非對於所有的 R、G、B 值皆有定義，當 $R = G = B$ 時(即顏色為灰色時)， $S = 0$ ，代入 H 的函式會發生除以零的情況，因此在色環中心部份，也就是灰色的部分，其 H 值之定義便會發生問題。以 opencv library 為例，在 $S = 0$ 時，H 之值與 V 相同，這樣的定法主要是為了確保 1 對 1 的性質。

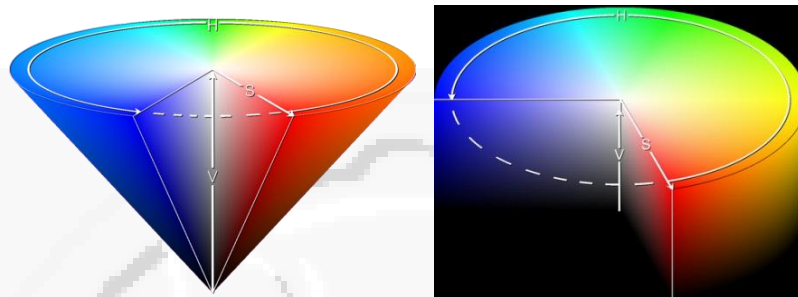


圖 1.5 常見的兩種 HSV 色彩空間圖示

(參考資料：<http://andrewharvey4.wordpress.com>)

YCrCb 色彩空間用 Y 來表示色彩的強度，以 Cr 和 Cb 表示 R 和 B 的色度。其經由 RGB 色彩空間的轉換公式如下：

$$Y = 0.299R + 0.587G + 0.114B$$

$$Cr = (R - Y) \times 0.713 + \delta$$

$$Cb = (B - Y) \times 0.564 + \delta$$

δ 是一個確保計算結果大於零的常數，如果 RGB 值為 0~255 之間， $\delta = 128$ ；若為 0~1 之間，則 $\delta = 0.5$ 。

這兩個色彩空間都有“亮度”這個維度(即維度 Y 與 V)，而除了亮度以外的另外兩個維度，理論上較不受光線變化的影響，而能夠單純的表達一個像素的“顏色”。因此人臉的膚色儘管以不同強度的光照射，或者由於人種不同而產生的先天膚色差異，其呈現在影像中之像素的 intensity 值在 H-S 和 Cb-Cr channels 中依然維持相當的集中度。以下列舉一些相關的研究工作。

Chai and Ngan (Chai and Ngan, 1999) 以數十張人臉影像統計與測試的結果中觀察到膚色的最佳定義範圍為 $77 \leq Cb \leq 127$ ， $133 \leq Cr \leq 173$ 。以此膚色範圍偵測出大概的膚色區域之後，再以形態學之技術來填補缺漏並濾掉雜訊，可以得到不

錯的結果。不過，此研究的測試影像均只包含單一人臉，且影像中主要的膚色區域也僅包含人臉，並不適用於本研究的情況。

Mahmoud(Mahmoud, 2008)一樣使用 YCrCb 色彩空間來偵測膚色，但除了限制 Cb 和 Cr 範圍外，同時限制亮度的範圍。其所提出的膚色範圍為 $Y > 80$ ， $85 < Cb < 135$ ， $135 < Cr < 180$ 。除了增加亮度的限制之外，在 Cb 和 Cr 的部分，範圍與 Chai 所提出的並沒有相差太多。這也說明膚色的分布在 YCbCr 色彩空間有一定的穩定性。

Phung(Phung, 2002)以 k -means 搭配 Mahalanobis distances 來學習膚色在 Cb 和 Cr 色彩空間的範圍。Phung 先使用一個固定範圍來快速的篩選掉非膚色像素，即 $75 \leq Cb \leq 135$ ， $130 \leq Cr \leq 180$ 。給定一個未分類像素，先檢查其對應的 intensity 值是否落在該固定範圍之內，若否，則歸為非膚色像素；若是，才計算此像素與訓練好的各群膚色中心點之 Mahalanobis distances，若存在與某群中心之距離小於一個門檻值，此像素才會被確認為膚色像素。

Sigal *et al.* (Sigal *et al.*, 2004)假設膚色的分布函數在不同照明之下的變動可以用同一函數的平移(translation)、縮放(scale)和旋轉(rotation)來描述，並稱其為 motion of distribution。他們以 HSV 為色彩空間，定義一組代表膚色分布函數變動的向量：

$$\xi = [T_H, T_S, T_V, S_H, S_S, S_V, \theta, \phi]^T$$

其中 T 代表色彩空間的平移， S 表示其縮放程度，而 θ, ϕ 則代表兩個旋轉角。

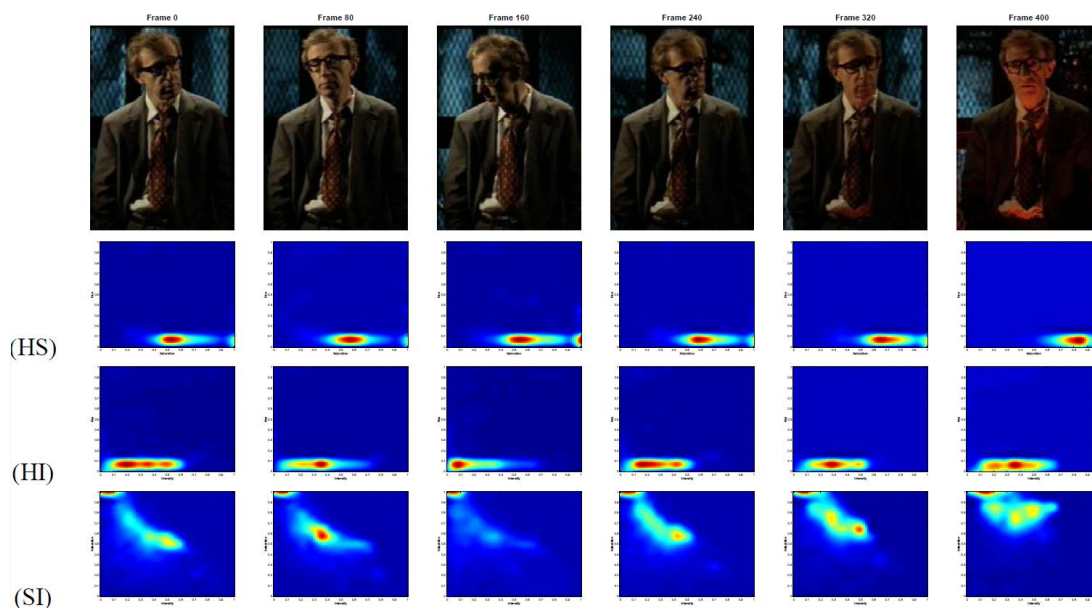


圖 1.6 膚色像素的 HSV 值在各種照明情況下之分布圖。(Sigal *et al.*, 2004)

圖 1.6 展示了該論文的核心想法，在不同照明情況下對膚色像素的 HSV 三個維度兩兩作圖。觀察圖中可以發現，除了分布的位置、大小以及方向略有變動之外，其分布形狀基本上是不變的。因此，該論文以二階馬可夫鏈作為模型，學習膚色分布的變動模式，如此既保有膚色分布的特徵，又能夠動態調整膚色範圍。

Zhang *et al.* (Zhang *et al.*, 2010) 在 HSV 色彩空間中使用一個固定的範圍擷取可能的膚色像素： $0^\circ \leq H \leq 50^\circ$ ， $0.2 \leq S \leq 0.68$ ， $0.35 \leq V \leq 1$ 。至於對固定 intensity 值範圍可能偵測失誤的部分，該論文以膚色補償來解決，根據影像中環境的顏色分布情況，估計光線的影響並作出調整，讓符合此範圍內的像素確實為膚色的機率提高。

總結來說，在單張影像的情況下，膚色在 HSV 和 YCrCb 兩個色彩空間中很容易以固定範圍偵測而得到大致正確的結果。不過，如果要能夠適用於各種照明環境，甚至用於影片，就必須要有動態學習並且修改膚色範圍的機制存在。

1.3 系統架構

本研究之系統流程圖如圖 1.7，對於每張輸入影像，經過背景去除(即前景擷取)步驟之後，在前景的範圍內偵測膚色區域。膚色區域經初步偵測後，再利用

graph cut 演算法將膚色區域完整化；最後，由多個膚色區域之中選擇屬於人臉的區域，並且重新統計膚色區域 intensity 值的分布情形以更新膚色範圍。

本系統的第一個步驟為背景去除，使用的方法以 graph cut 演算法為主。採用 graph cut 技術的好處是利用該演算法進行背景去除不會受限於背景在影像中需佔據大部分時間與面積的假設，符合本研究的精神；另外，graph cut 演算法除了考慮單一像素的分類之外，也將相鄰像素間的相似程度納入計算，因此能得到完整的前景擷取結果。不過，由於該演算法所需要的輸入包含了前景與背景的機率模型和若干需事先被手動標記的像素點，本研究另外提出了一個穩定快速的全域門檻值計算方法來去除背景並取得初步的前景擷取結果，以自動提供 graph cut 所需要的輸入。

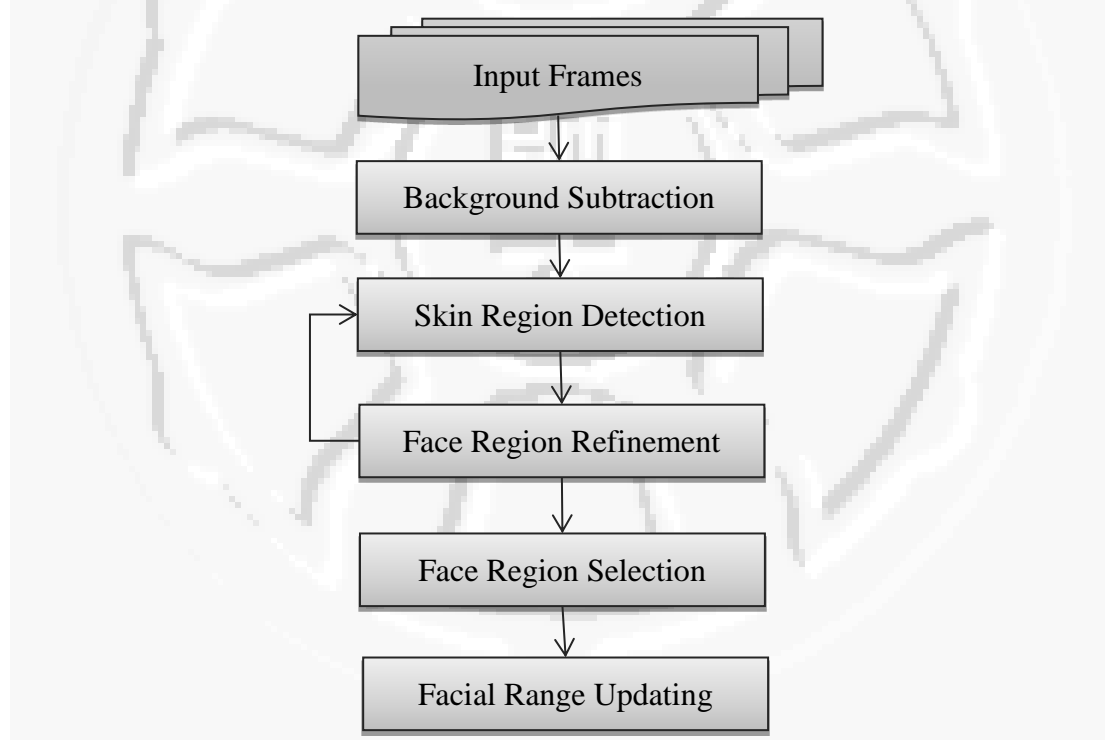


圖 1.7 系統流程圖。

背景去除之後，系統開始在前景區域中進行膚色偵測，選定的 channel 主要為 HSV 中的 hue channel。雖然膚色像素之 intensity 值範圍容易受到光線變化的影響，但是同一影像中的鄰近膚色像素，其在 hue channel 中的集中程度卻相當高。利用此一特性，系統在以固定範圍偵測膚色區域之後，再度利用 graph cut

演算法來提高膚色區塊的完整程度，並且重新統計膚色的 intensity 值來更新膚色在 hue channel 的偵測區間。

本論文將在以下章節介紹各個步驟所使用的方法。第二章介紹背景去除的初步工作；第三章介紹 graph cut 演算法的理論以及套用至連續影像的概念；第四章介紹增進 graph cut 速度的演算法——dynamic graph cut 演算法以及實作細節；第五章介紹膚色區域偵測、人臉區域選取以及膚色範圍更新的程序，第六章則為實驗結果以及結論。



第二章 影像前處理

2.1 差值影像

本研究在前景初步擷取的部分首先參考了 Chiu(Chiu, 2010)所提出的方法，該研究利用輸入影像和背景影像相減所計算出來的影像(簡稱差值影像)建機率模型，並且將左右兩邊離主峰最近的第一個反曲點定為 global threshold，以此做為區分前景以及背景的門檻值。

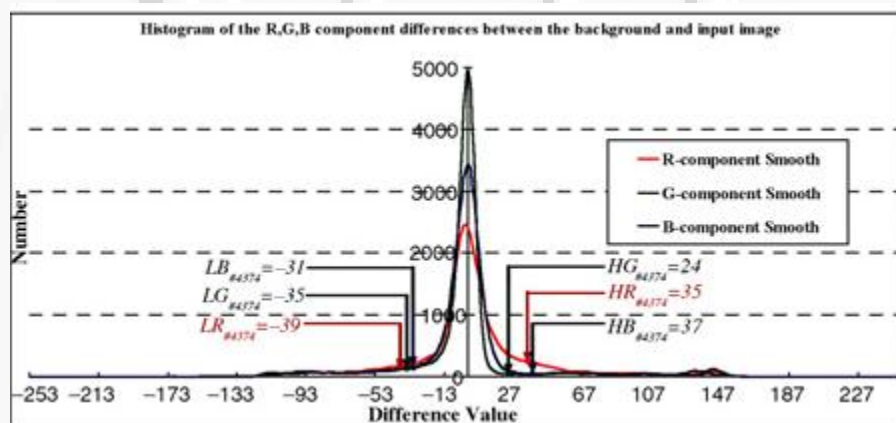


圖 2.1 Difference histogram 和 turning point 示意圖(Chiu, 2010)。

由圖 2.1 可以看出 Chiu 演算法的核心思維：差值影像中像素的 intensity 值大多趨近於零，因此，由該影像所建立的 R、G、B 三個彩色維度的 color difference histogram(以下簡稱 difference histogram)皆分別趨近 zero mean 的 Gaussian 分布。另一方面，前景像素的 intensity 差值因為大多為非零的值，會在 histogram 的兩側造成一些波峰。該研究認為第一個屬於前景波峰的起點，就可以做為前景和背景的分界點。

圖 2.2 為 Chiu 演算法流程圖，由 difference histogram 計算出 global threshold 之後，進行前景擷取。前景擷取之後，再進行一些型態學の後處理，包括 dilation, hollow filling 和 shadow removal 等，最後再將確定是背景的像素之 intensity 值與原來的背景影像對應像素的 intensity 值做線性內插(linear interpolation)處理，以

更新後背景像素的 intensity 值。

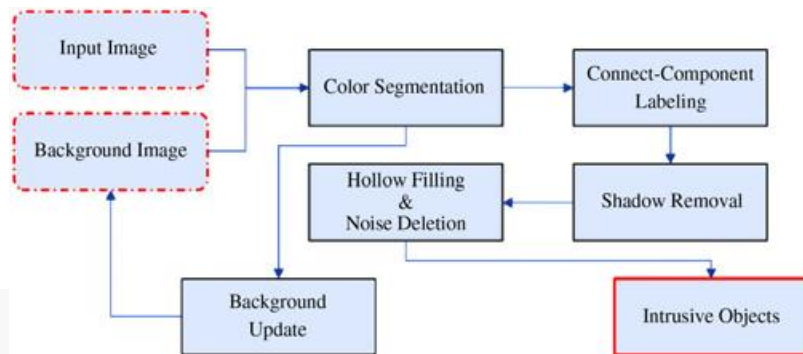


圖 2.2 Chiu 提出的前景擷取流程圖(Chiu, 2010)。

圖 2.3 展示了一般未做形態學處理的實驗結果，除了前景中有少許缺口之外，大致上是正確的，爾後輔以一些填補、去陰影的動作，在原論文提及的實驗結果中就已經能得到相當高的正確率。

圖 2.3 展示以差值影像實作前景擷取但尚未做形態學處理的實驗結果。由圖中可觀察得知除了前景中有少許缺口之外，前景擷取的結果大致上是正確的，爾後輔以一些填補、去陰影的動作，就能得到相當高的正確率。

圖 2.4 為一利用 difference histogram 進行前景擷取的例子。由圖中可觀察到，像素之 intensity 差值的波峰，除了接近零的大部分區塊以外，的確都屬於前景的區塊；從圖 2.4(b) 中 R、G、B channels 分別之 difference histograms 也可以發現，在零點右側皆有另一個明顯的波峰出現，這是因為影像中前景部分的像素值大部分都大於背景的像素值。



圖 2.3 以差值影像進行前景擷取的結果。(左)為原圖，(右)白色部分為前景

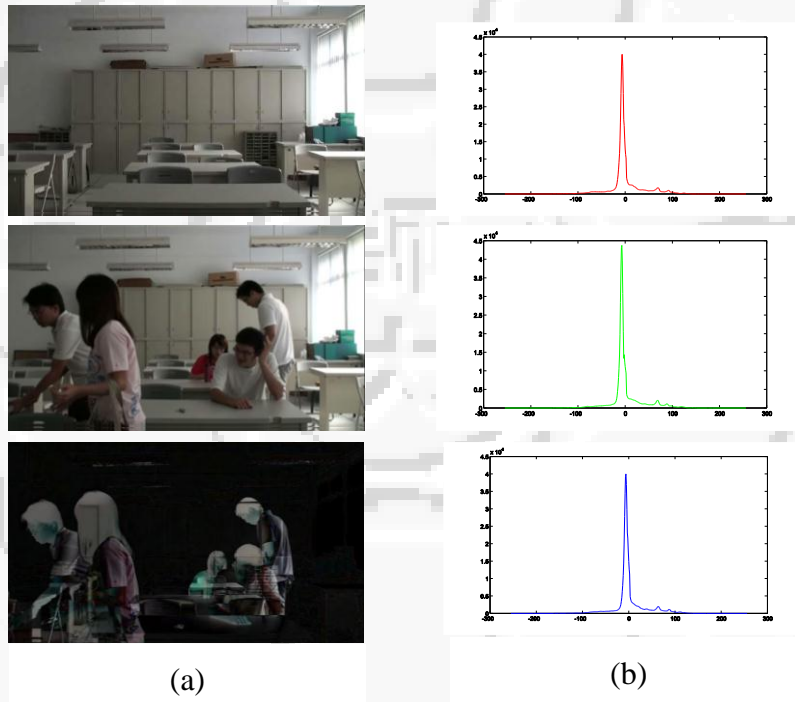


圖 2.4 Difference

(a)由上至下分別為背景影像、原圖以及差值
(b)分別之 difference histograms

由上述 difference histogram 來擷取前景相當自然且合理，而且不需要輸入一張影像，只需要配合一張背景影像、計算出 difference histogram 和一個 global threshold 就可以得到不錯的前景擷取結果，不論是在時間和空間上的成本都非常小。儘管如此，要將此方法套用至本研究上時卻發現了一些問題：

1. Difference histogram 的平滑度會影響 global threshold 的選取結果：理論上來

說，最佳的門檻值應定義為“difference histogram 左右兩側離中心點最近的極小值”(該值所在的位置稱為 turning point)，但在實作中 histogram 的曲線往往是粗糙的，極易混淆極小值的定義。因此 Chiu 對於 difference histogram 做了簡單的平滑化動作：將每一個點自己的 histogram 值與鄰近五個點的 histogram 值平均取代該點的 histogram 值。但這種平滑化的方式是否足以解決此問題，畢竟缺乏理論支持。

2. 極小值的操作型定義並不完全適用於該方法：極小值在微積分中的定義為“一次微分等於零，且二次微分大於零”之處。但 Chiu 對極小值的處理卻相當簡單，由零點往左右兩側出發，往右邊找到第一個 histogram 值遞增的點 ($H[i-1] < H[i] \& H[i] < H[i+1]$)；往左邊找亦找第一個 histogram 值遞增的點 ($H[i-1] > H[i] \& H[i] > H[i+1]$) 便視此兩點為極小值。這樣的作法對於前景面積較大的影像而言就會產生問題，因為其 difference histogram 相對來說比較不穩定。圖 2.5 即為實作 difference histogram 前景擷取的失敗範例。在該連續影像中，一名學生因正要入座而進入了影像中，由於該學生在畫面中所佔的面積比例相當大，有許多背景像素被遮蔽，因此在 difference histogram 接近零點附近的點其 histogram 值也產生了劇烈的變化。在這種情況下，很有可能會在非常接近零點附近找到極小值，而造成前景擷取包含背景的錯誤狀況。



圖 2.5 實作 difference histogram 前景擷取的失敗範例。此為連續的五張輸入影像之前景擷取結果。

3. RGB 三個門檻值的結合方式並不合理：由於差值影像分成 RGB 三個 channels，且分別計算出 difference histograms 和對應的門檻值。如何結合這三個 channels 的門檻值作為分類依據在 Chiu 的演算法中亦扮演重要角色。原論文取三個門檻值的絕對值之和為最終的門檻值，若差值影像的某個像素

其 RGB 絕對值總和大於這個門檻值，則該像素視為前景，反之則為背景。



圖 2.6 Chiu 之論文所使用的影片範例(Chiu, 2010)。

4. 背景所佔影像面積與出現時間比會影響背景更新的結果：原論文所用來測試的連續影像，背景皆佔據大部分的出現時間與影像面積(如圖 2.6)。因此即使有某些前景像素暫時被誤分為背景，只要連續影像時間夠長，原來背景的像素終究會被突顯出來。不過，在教室中的學生不僅可能長時間靜止，在本研究的應用中甚至會佔據大部分影像面積，若有被誤分為背景的前景像素，可能會因為前景的長時間靜止而一直被分為背景，造成背景影像無法正確收斂的結果。圖 2.7 為一背景更新失誤的範例，由圖 2.7(c)之背景影像可見許多理應屬於前景的顏色出現，即是因為前景像素長時間在畫面中靜止，累積錯誤分類的結果。

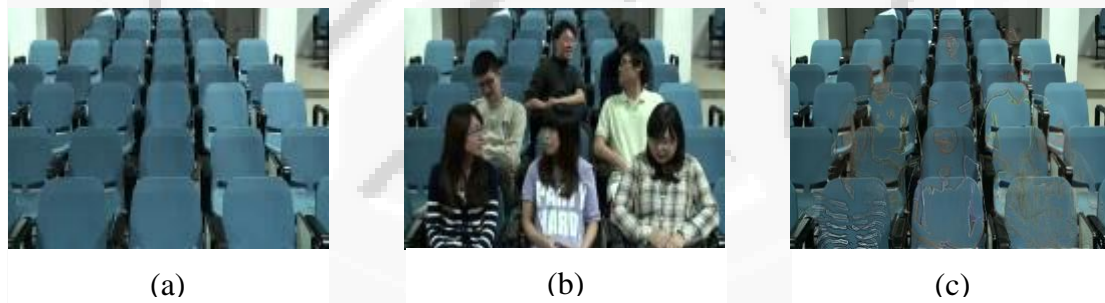


圖 2.7 背景更新失誤之範例。(a)初始背景影像(b)第 446 張輸入影像(c)第 446 張背景影像

2.2 對 difference histogram 前景擷取演算法的改良

在 2.1 節中提及將 difference histogram 為基礎的前景擷取方法套用至本研究上時所遭遇的四個主要問題。在這四個問題之中，以第 2、4 個問題對於前景擷取的結果較具決定性，而第 4 個問題又能夠有效的被 graph cut 演算法解決；因

此，本節將著重於討論如何解決第二個問題，即在 difference histogram 中取得較理想的“極小值”。

在 difference histogram 中要取得極小值有兩個方式，一是從數學上的定義下手，也就是 2.1 節所提到的“一次微分為零，二次微分小於零”之處。儘管該 difference histogram 並非 closed form 的函數，也非連續函數，基本上是不可微分，雖然能夠以差分的方式去逼近，得到最接近微分為零之處，不過其成效仍然視 difference histogram 的平滑程度而定。

另一種取得極小值的方法是從直觀的定義去下手，即“在某個範圍內的最小值”，此定義的好處是不論原函數的性質如何皆可套用，只是要在什麼範圍取最小值需要仰賴其他假設或知識而定。考慮到無法事先得知 difference histogram 的平滑程度，使用第一種方法的風險較高，因此本研究採取了第二種方式。

假設前後二張連續影像的前景比例不會相差太多，上一張輸入影像的前景比例就可以用來估計目前輸入影像的前景比例。如此就可以利用前一張影像前景比例的資訊，估計一個可能的前景背景分割處，再於該點附近幾個 bin 中尋找最小值，當成最後的門檻值。圖 2.8 為改良之後的 difference histogram 前景擷取流程圖，與 Chiu 演算法不同之處在於額外計算前一張的前景所佔面積比，藉此估計當前輸入影像的前景比例和門檻值。

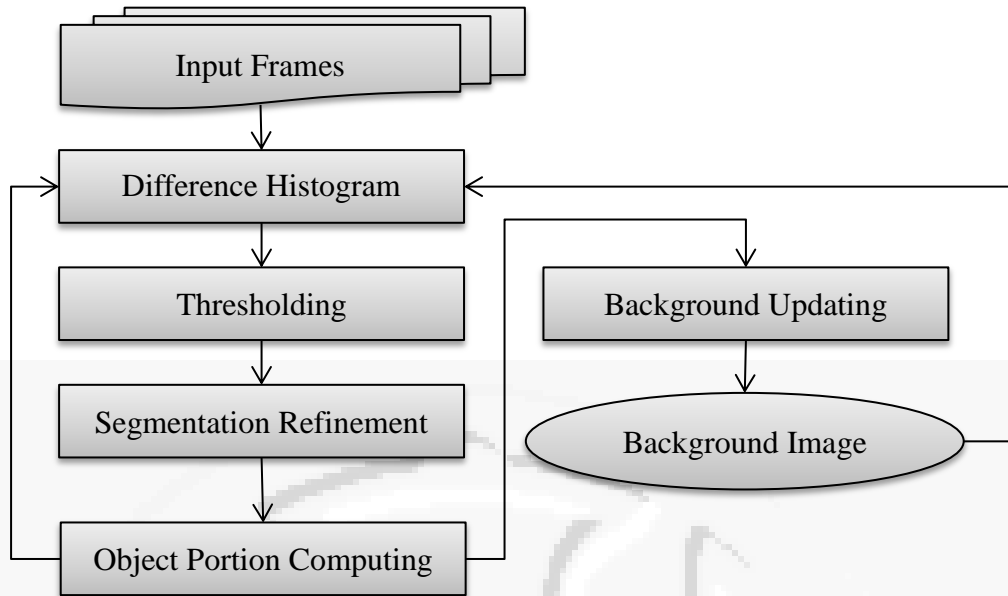


圖 2.8 改良後的 difference histogram 前景擷取流程圖。

另外，在統計前景所佔影像面積比例時，位於 difference histogram 零點右側和左側的前景面積比例是必須分開計算的。其原因在於，位於左右兩側的前景事實上代表著兩種不同類型的前景。右側是比背景亮的前景，而左側則相反。此兩者所佔比例視前景物體像素的 intensity 值分布情形可能會有極大差異；一言以蔽之，difference histogram 一般來說並非左右對稱。在此情況下，將左右兩側的前景比例分開計算，才會找到合理的初始門檻值。

另一方面，前景面積比例的計算是在 graph cut 之後，而 graph cut 演算法必然會加大前景的範圍。因此，在連續二張輸入影像之前景面積比例相差不多的前提下，可以預期計算出來的前景面積比例會大於實際的比例，因此估計的初始門檻值會小於理想值。

既然已經預期初始值會小於理想值，則比初始值小的部分就不需要再列入考慮，只要由初始值向外找即可。因此本研究將兩側初始門檻值往外 10 個 bins 中的 histogram 值最小處視為最恰當的門檻值，圖 2.9 為最佳門檻值取得示意圖。兩側的著色區塊代表在前一張影像中的前景面積比例。其中紅色區塊為前景像素之 intensity 值比背景小的部分，藍色區塊則是前景像素之 intensity 值大於背景像素值的部分。根據此二者的比例，系統能夠在零點兩側各找一個初始門檻值(即

兩垂直線所指之處)。最後，由這兩個初始門檻值為起點，往外搜尋 10 個 bins，找出 histogram 值最小的 bin 設定為最佳門檻值。

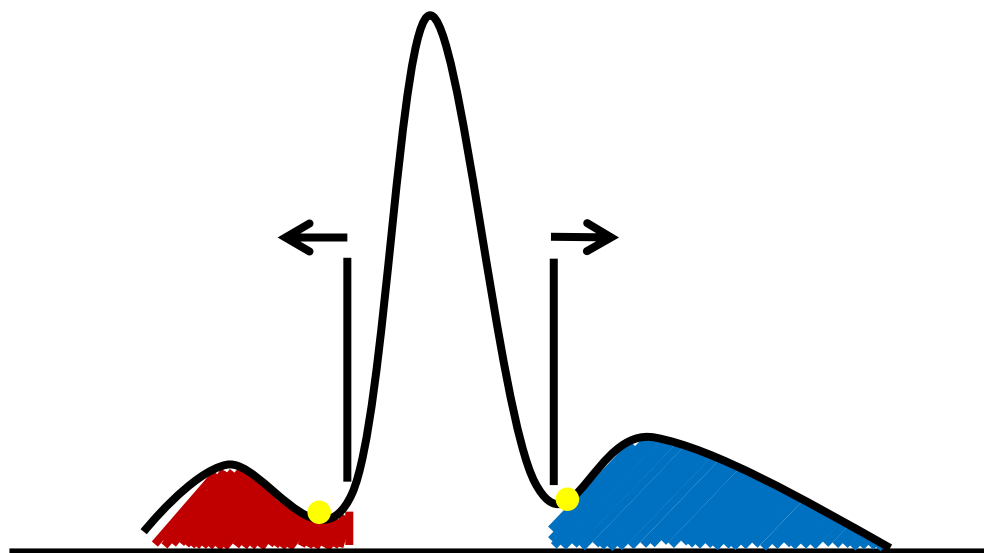


圖 2.9 取得最佳門檻值之示意圖。

Chui 之前景擷取演算法經過上述修正之後，系統得以預測的前景面積比例限制 global threshold 的搜尋範圍。即使 difference histogram 偶爾因為前景面積比例劇烈的變化而產生大量雜訊，系統依然可以得到相對合理的 global threshold，再經由 graph cut 演算法的增強，即可得到完整的前景擷取結果(圖 2.10)。由圖 2.10 中可見，取過 global threshold 的影像儘管在學生右側因為物體快速移動而在邊界產生偵測失誤，經過 graph cut 演算法的修正之後即得到相當正確的結果。以下將在第三章詳細介紹 graph cut 的原理和其演算法。



圖 2.10 改良 Chui 演算法之後前景擷取的結果。上排為取 global threshold 後之結果；下排為 graph cut 之後的結果。



第三章 Graph cut 前景擷取技術

本章主要介紹 graph cut 的前景擷取技術。Graph cut 技術除了考慮單一像素的分類外，相鄰像素間的關係也被納入考量，此時顏色相近的相鄰像素會傾向被分到同一類。這個概念使得前景擷取的結果會更完整。不過，Boykov and Jolly (Boykov and Jolly, 2001) 所提出的 graph cut 技術僅用於單張影像的前景擷取，並且還需使用者手動設定 hard constraint。本研究針對上述問題進行改良，以期使用 graph cut 技術自動處理連續影像前景擷取的問題。本章將在 3.1 節敘述 graph cut 的理論架構以及圖論背景知識，並在 3.2 節說明本系統將 graph cut 演算法應用於連續影像之步驟。

3.1 Graph cut 技術

本節首先定義相關的名詞，接著介紹 graph cut 的成本函數，最後則敘述 Boykov and Jolly 如何以 max-flow min-cut 的演算法去最小化成本函數並得到最佳的前景擷取結果。

3.1.1 名詞定義

以下定義在說明 graph cut 技術時所需的相關名詞：

- cut：給定一張 graph $G = (V, E)$ ，其中 V 為 vertices 所成集合， E 為 links 所成集合，每一個 link e 都賦予一個權重 w_e 。在 G 上可以定義一個 cut C ， C 為 E 的子集，使得 G 的導出圖(induced graph，也就是在 G 中刪除屬於 C 的 links) $G(C) = (V, E^C)$ 將 V 分為兩個不連通(disconnected)的子集。若要求這個 cut 必須把特定兩個點—source 和 terminal 分屬不同子集，則此 cut 又稱 st-cut。
- minimum cut：一個 cut C 的權重定義為該 cut 上所有邊的權重總和，即

$$|C| = \sum_{e \in C} w_e \quad (3.1)$$

任一 graph 上的 cut 並不唯一，而權重最小的 cut，就稱為該 graph 的 minimum cut。

- maximum flow：給定一個 graph $G = (V, E)$ ，則一包含 source 和 terminal 的子圖 f 若符合 G 的容量限制(capacity constraint，即 f 上的任一 link，其權重皆不得逾越該 link 在 G 上的容量)和守恆律(conservation law，即 f 上任一點的指出 link 權重和等於指入 link 權重和)，則稱 f 為 G 上的 flow。一個 flow 的權重為 f 中 source 所有指出 link 的權重總和；而一個 graph 中 link 權重總和最大的 flow，便稱為該 graph 的 maximum flow。
- max-flow min-cut 定理：對於任一 graph，maximum flow 的值等於 minimum cut 的值。

3.1.2 影像標記(labeling)和成本函數(cost function)

若給定一張影像 I ，其像素個數為 P ，則可以用一個二元向量 $A = (A_1, A_2, \dots, A_p)$ 來代表一個影像標記。其中 A_i 可以是 0 或 1，也可以標記為背景像素“bkg”或前景像素“obj”。Boykov and Jolly (Boykov and Jolly, 2001) 希望能夠以使用者劃定之某些必為前景和背景的区域(hard constraint)作為線索，配合影像中像素的 intensity 值來對一個影像標記定義其成本函數。由圖 3.1 可以略窺其意，當使用者定義了某些確定為前景和背景的区域之後，系統便能夠以這些区域中像素的 intensity 值所提供的資訊分別建立前景和背景的機率模型。某一像素被賦予一個標記時，所需付出的成本便和該像素與此標記類別的機率模型之符合程度有關。

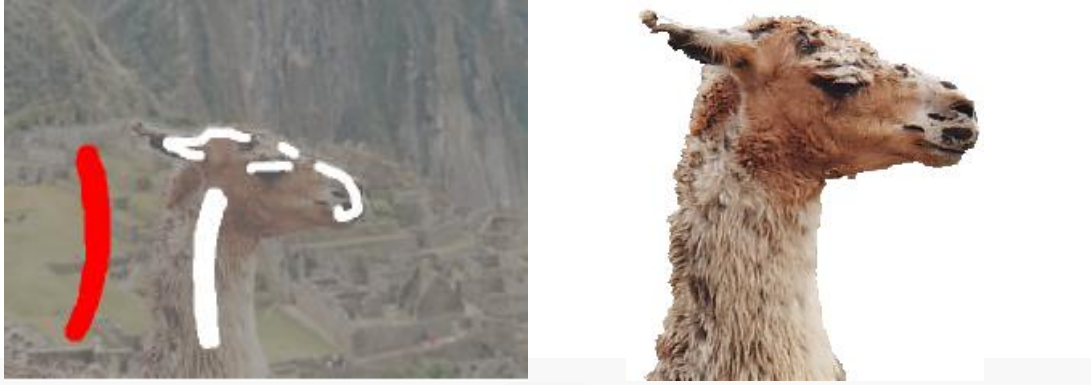


圖 3.1 Graph cut 範例(Rother, 2004)。左圖的紅色區域為使用者定義的背景部分，白色區域則為前景部分。右圖為影像前景擷取的結果。

前面提到對於一個影像標記 A 可以定義成本函數 $E(A)$ 。 $E(A)$ 定義如下：

$$E(A) = R(A) + \lambda B(A) \quad (3.2)$$

其中

$$R(A) = \sum_{p \in I} R_p(A_p) \quad (3.3)$$

$$B(A) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(A_p, A_q) \quad (3.4)$$

且

$$\delta(A_p, A_q) = \begin{cases} 1 & \text{if } A_p \neq A_q \\ 0 & \text{otherwise} \end{cases}$$

Boykov and Jolly 在該研究中將此成本函數分為區域項(region term) $R(A)$ 和邊界項(boundary term) $B(A)$ ，其中 λ 是一個參數，用以決定此兩者的相對重要程度。

區域項的概念在上段已略有描述，藉由使用者所定義的背景和前景區域，能夠得到兩個關於前景和背景 intensity 值的事前機率分布(prior probability distribution) $P(x|O)$ 和 $P(x|B)$ 。給定像素 p ，標記 p 為前景和背景之區域項成本定為

$$R_p(\text{"obj"}) = -\log P(I_p | O) \quad (3.5)$$

$$R_p(\text{"bkg"}) = -\log P(I_p | B) \quad (3.6)$$

由於機率值在 0 到 1 之間，且對數函數取負號會將 1 映至 0，將 0 映至無窮大。因此，若該像素為前景的機率越高，將其分類為前景所需的成本也就越低；反之亦然。

區域項成本是定義在每個被標記的像素上，但邊界項成本的部分是定義在每一對相鄰的像素之間。首先觀察 $\delta(A_p, A_q)$ ，當 $\delta(A_p, A_q) = 0$ 時， $A_p = A_q$ 。也就是說，被分為同一類的相鄰像素，並沒有邊界項的成本。若 $A_p \neq A_q$ ，即相鄰的像素 p 和 q 被標記為不同的兩個類別時，代表邊界項成本的 $B_{\{p,q\}}$ 函數才會產生影響，其中

$$B_{\{p,q\}} = \exp\{-(I_p - I_q)^2\} \quad (3.7)$$

$-(I_p - I_q)^2$ 必小於零，因此 $B_{\{p,q\}}$ 的值會落在 0 和 1 之間。假設某一對相鄰像素 p 和 q 的 intensity 值十分接近，卻又被標記為不同類別，其邊界項成本就會趨近於最大值 1；若此二者的 intensity 值差異很大，被標記為不同類別就只需要付出少量的成本。

綜合來說，一個低成本の影像標記符合以下兩個條件：1. 被標記的像素其 intensity 值必須相當符合其標記類別の機率模型。2. 被標記為不同類別の相鄰像素，其 intensity 值必須有很大的差異。

上節簡述了 graph cut 演算法的基本精神，亦即如何利用成本函數來衡量一影像標記的優劣程度。以下將說明如何利用 maximum flow 與 minimum cut 演算法來獲得成本最低の影像標記。

3.1.3 最小化成本函數

考慮前一小節提及の影像 I ，建立一張 graph $G = (V, E)$ 。其中， V 所代表的 vertices 包含 I 的所有像素以及兩個額外の vertices: source S (代表前景) 和 terminal T (代表背景)。另一方面， E 包含の edges 有兩種，為 t-link (terminal link) 和

n-link(neighborhood link)。每一個像素(vertex)均有兩條 t-link(分別連結到 S 和 T) 和四條 n-link(連結相鄰像素)，如圖 3.2 所示。

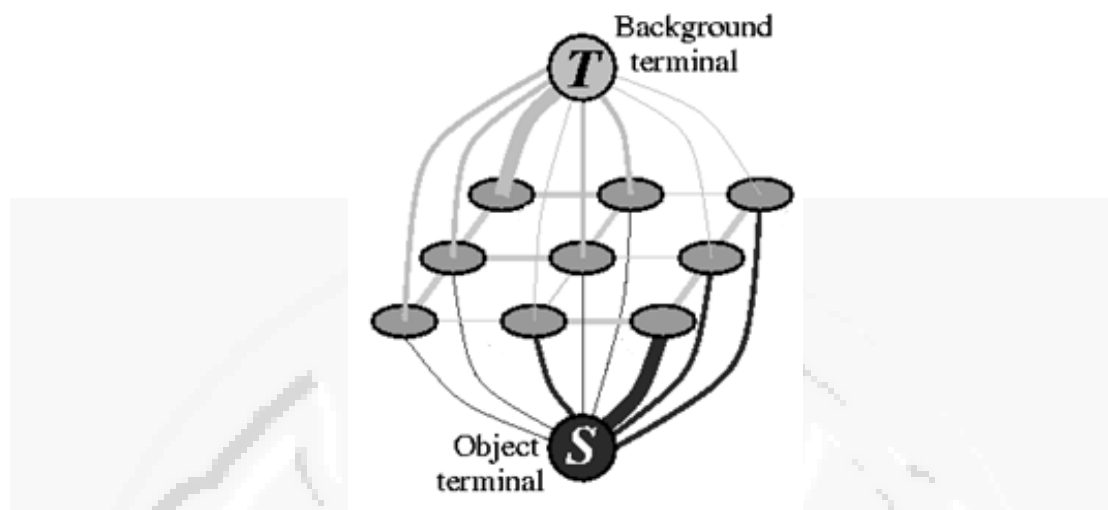


圖 3.2 由 I 所建立的 graph 之範例(Juan and Y. Boykov, 2006)。

表 3.1 中呈現各種 links 權重定義的方式。於一個 graph G 中之 vertex(像素) p 可分為三類：1. 使用者定義為背景。2. 使用者定義為前景。3. 其它待標記 vertices。Vertices 間 t-link 的權重賦予則跟 p 在 vertex p 的類別有關。若 p 為使用者定義的前景，則其與 T 相連的 t-link 權重值為 0，與 S 相連的 t-link 必須大於所有 p 之 n-link 之和。同理，若 p 為已知背景，其與 S 相連的 t-link 權重為零，另一條 t-link 權重必須大於所有 p 之 n-link 之和。假設 p 屬於第三類，即其為待標記 vertex，則 p 與背景和前景相連的 t-link 分別為前景和背景的区域項成本函數。另外，不論 p 為何種類別，其周圍四條 n-link 的權重定義為邊界項成本函數 $B_{\{p,q\}} = \exp\{-(I_p - I_q)^2\}$ ，其中 q 為與 p 為相鄰的像素。

權重給定後，此 graph 就形成了一個 graph $N = (G, S, T)$ ，接著就可以在該 graph 上找到最小的 st-cut C^* 。此 minimum cut C^* 將 1-1 對應於一個影像標記 A^* ，在此 cut 之下與 S 相連的 vertices 即對應前景像素，而與 T 相連的 vertices 則對應至背景像素。此時， C^* 的權重與 $E(A^*)$ 相同，且 A^* 即為成本最小的影像標記。

表 3.1 依據各 link 賦予適當的權重。其中 O 和 B 為使用者定義的前景和背景像素所構成的集合。

Edge	Weight	for
$\{p, q\}$	$\exp\{-(p - q)^2\}$	$\{p, q\} \in N$
$\{p, S\}$	0	$p \in B$
	$4\lambda + \varepsilon$	$p \in O$
	$-\lambda \log P(I B)$	otherwise
$\{p, T\}$	0	$p \in O$
	$4\lambda + \varepsilon$	$p \in B$
	$-\lambda \log P(I O)$	otherwise

3.1.4 相關定理證明

將上節的 graph 建立完成後，任一個 st-cut 都會將此 graph 的 vertices 一分為二，其中一群與 S 相連，另一群與 t 相連。若希望此標記結果能夠保有使用者定義(hard constraint)的部分，且能夠讓 minimum cut 的權重對應到最低成本的影像標記，則此一 cut 必須是一個 feasible cut。

給定一 graph 與其對應的任一 cut C ，若 C 符合以下四個條件則 C 稱為一 feasible cut：

1. 每一個像素對應的 vertex 僅有一條 t-link 包含於 C 。
2. $\{p, q\} \in C$ 若且唯若 p 和 q 之 t-link 連結至不同的 vertex。
3. 若 p 為已知前景，則 $\{p, T\} \in C$ 。
4. 若 p 為已知背景，則 $\{p, S\} \in C$ 。

圖 3.3 中顯示 feasible cut 的一個範例。由上述定義可知一個 graph 中的 feasible cut，其單一像素所對應的 vertex 的 t-link 僅有一條，因此可由這些 t-link 唯一決定一個影像標記。值得注意的是被分到不同類別的相鄰像素間不會有 n-link 之連

結，因為其間的連結會被 cut。接下來將證明 minimum cut C^* 為一 feasible cut，而且對應到最小成本的影像標記。

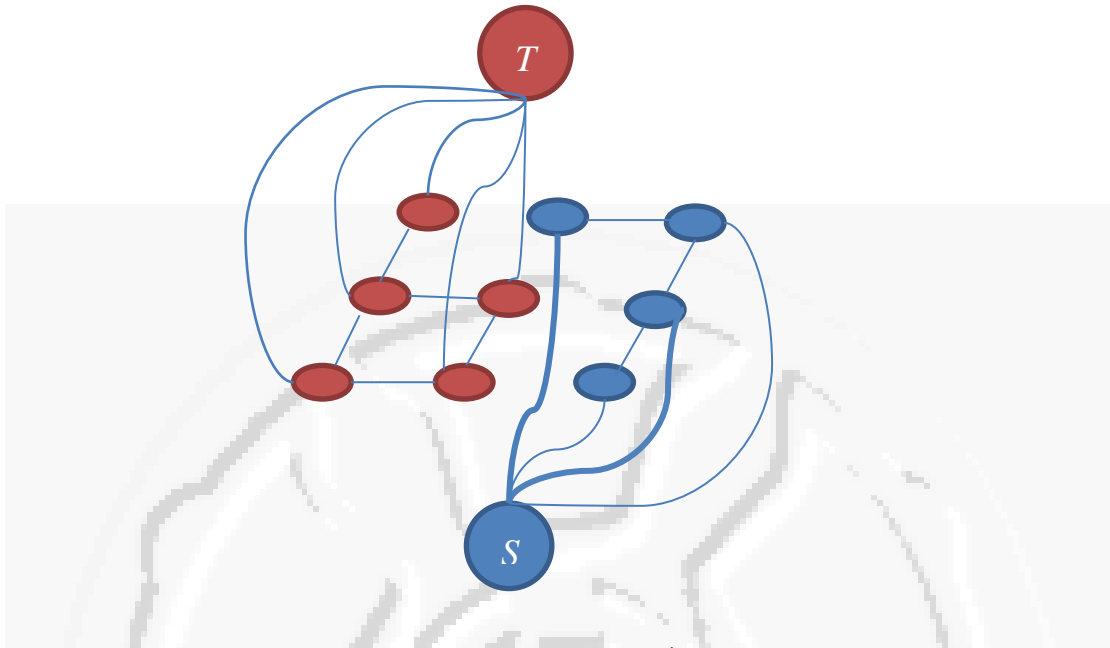


圖 3.3 Feasible cut 之範例。

● **引理：** C^* 為一 feasible cut。

證明：

- (i) 首先說明一給定 graph 經過 minimum cut 演算法後每個像素所對應的 vertex 僅餘一條 t-link。假設存在某個 vertex 留有兩條 t-link，則表示 source 和 terminal 間就仍然有路徑(path)存在，這違反了 cut 的定義。另一方面，假設存在另一 vertex 經過 cut 之後無任何 t-link，則此 cut 不可能為 minimum cut。因為，必然能夠在 graph 中加回該 vertex 的其中一條 t-link，即自 C^* 中刪除該條 t-link，來得到一個成本更小的 cut。
- (ii) 接下來證明 $\{p, q\} \in C$ 若且唯若 p 和 q 之 t-link 連結至不同的 vertex：

“ \Leftarrow ”：假設 p 和 q 之 t-link 連結至不同的 vertex，但 $\{p, q\} \notin C^*$ ，則 source 和 terminal 間有路徑存在，不符合 cut 定義；因此 $\{p, q\} \in C^*$ 。

“ \Rightarrow ”：在此證明此命題的逆否命題，即“若 p 和 q 之 t-link 連結至相同的 vertex，則 $\{p, q\} \notin C^*$ ”。假設 $\{p, q\} \in C^*$ 且 p 和 q 之 t-link 連結至相同的

vertex，則此 cut 不會為 minimum cut；因為將 $\{p, q\}$ 自 C^* 中去除，不會產生 source 至 terminal 的路徑但卻可降低 cut 的成本，得到另一個成本更小的 minimum cut，所以 $\{p, q\} \notin C^*$ 。

(iii) 若 p 為已知前景，則 $\{p, q\} \in C^*$ 。假設存在 $p \in O$ 但 $\{p, S\} \in C^*$ ，因之前表 3.1 權重的定義，可將 $\{p, S\}$ 接上，改將 p 之所有 n-link 和 $\{p, T\}$ cut 掉而得到一個成本更小的 cut。同樣道理可以證明 C^* 符合最後一個性質。

● **定理一：由 C^* 所對應的影像標記 A^* 為成本最小之影像標記**

證明：首先證明對於任何符合 hard constraint 之影像標記 A ，其成本函數值 $E(A)$ 與其對應的 feasible cut C 之權重相同。觀察表 3.1 可知

$$\begin{aligned} |C| &= \sum_{p \in O \cup B} \lambda R_p(A_p) + \sum_{\{p, q\} \in N} B_{\{p, q\}} \delta(A_p, A_q) \\ &= E(A) - \sum_{p \in O} \lambda R_p(\text{"obj"}) - \sum_{p \in B} \lambda R_p(\text{"bkg"}) = E(A) - \text{const} \end{aligned}$$

此 *const* 常數為使用者定義部分的區域項成本函數，對於任何滿足 hard constraint 的影像標記皆為相同的值。因此

$$E(A^*) = \min_A E(A) = \min_C E(A) = \min_C |C| + \text{const} = |C^*| + \text{const}$$

總而言之，在用 graph cut 技術進行前景擷取時，首先讓使用者在影像上定義 hard constraint，以收集關於前景與背景的事前知識；並在這個基礎之下，定義影像標記的成本函數。要解出成本最低的影像標記可利用該影像定義一個 graph，並且證明在這個 graph 上的 minimum cut 能夠對應到最低成本的影像標記。

3.2 應用於影片前景擷取

要將 graph cut 演算法應用於影片前景擷取，首要面對的問題是如何將原本手動的程序自動化，即自動取得 hard constraint 的訊息。由於本研究在前一個階段已經利用一個全域門檻值得到初步的前景擷取結果，因此我們利用這個結果來提供 graph cut 所需的事前資訊。

總結 graph cut 演算法所需要的輸入資訊有：1. 原始影像；2. 前景以及背景的

hard constraint；以及3.前景以及背景的事前機率模型。在hard constraint自動化的部分，由於背景在影像中的位置相較於前景而言較為聚集，所佔影像區域的面積也比較大，因此本系統在取過全域門檻值的二值化影像中搜尋，若以某一個像素 p 為中心 7×7 pixels 的 patch (如圖 3.4 之黃框所示) 全部都被標為背景的話， p 就會被加入成為背景的 hard constraint。另外，由於大部分的前景較背景破碎，所佔影像區域的面積也較小，因此本系統只取 3×3 pixels 的 patch (如圖 3.4 之紅框所示) 來決定必為前景的像素。



圖 3.4 Hard constraint 賦予示意圖。上排為原輸入影像，中排為取全域門檻值之前景擷取結果，下排為二值化影像與定義 hard constraint 之示意。其中黃色方框之中心會被定義為背景，紅色方框中心則會被定義為前景

實作方面，由於對每一個像素進行 7×7 pixels 的 patch 檢查太過耗時，本研究於此使用了 integral image 的概念 (Viola, 1994)。Integral image 原為該論文用以計算

長方形特徵的方法。在 integral image 中任一位置 (x, y) 之 intensity 值為二值化影像中所有在 (x, y) 左方及上方之像素其值的總和。

對一張二值化影像建構好 integral image 之後，若欲計算在此二值化影像中某一方塊內的其值為 1 之像素總和，只需自 integral image 中以該方塊四個角的總合值做簡單的加減便能夠完成，並且其計算量不受方塊大小的限制。以圖 3.5 為例，若要計算 ABCD 矩形內部二值化後其值為 1 之像素總和，只要將 $D - B - C + A$ 即可。更重要的是，建構 integral image 時只需要掃描原二值化影像一次就能夠完成，因此可以省去大量的加減計算。

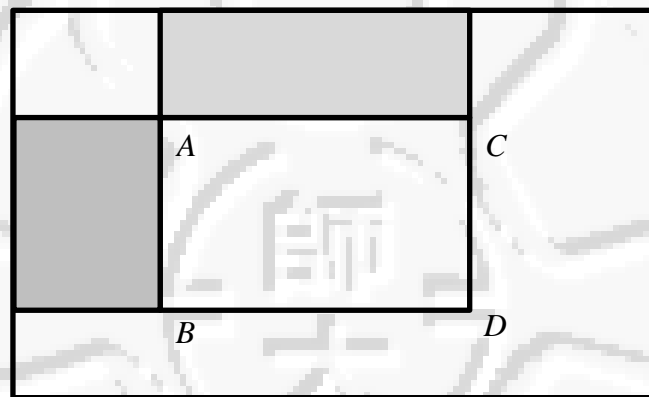


圖 3.5 由 integral image 計算矩形內部二值化後其值為 1 之像素總和。

擷取出 hard constraint 之後，系統還必須計算前景與背景的事前機率分布以利後續 graph cut 的處理。在單張影像的 graph cut 演算法中，此事前機率分布是以統計 hard constraint 的 intensity 值之分佈取得。因此，既然已經得到了 hard constraint，事前機率的部分當可以如法得之。不過，本研究在此有一優勢即系統中存有一張背景影像，故背景的事前機率分布完全可以由此背景影像統計而得，比起僅取 hard constraint 部分的統計方式更加完整。以下為本系統應用 graph cut 之流程圖(圖 3.6)。經由全域門檻值的計算，得到初步去背景結果之後，系統就能夠進行 hard constraint 的標記和前景背景事前機率的計算，以此提供建構 graph 所需要的輸入而能夠進行 graph cut 演算法。

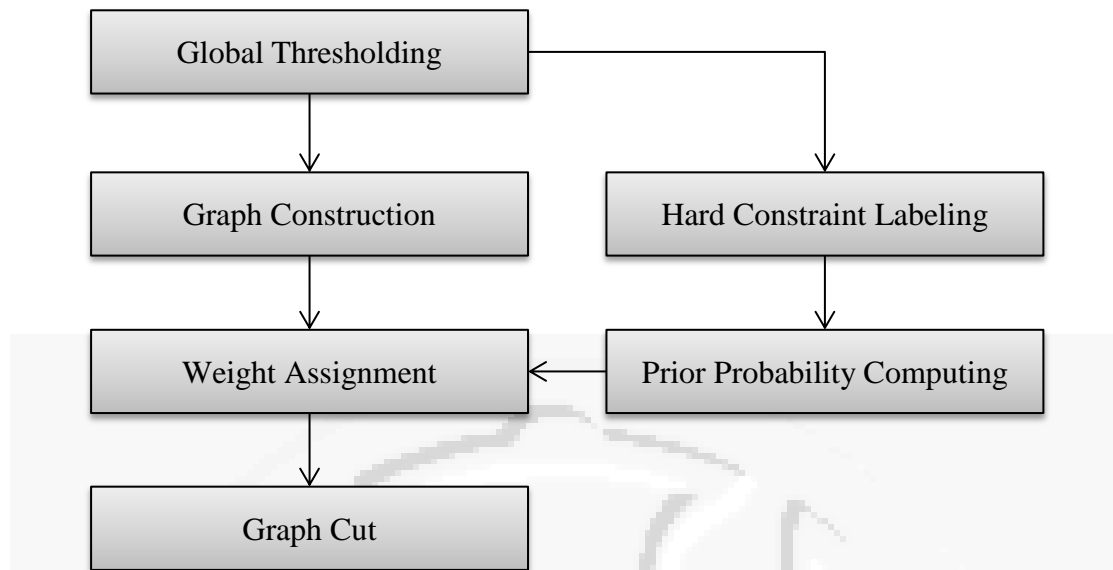


圖 3.6 Graph cut 流程圖。

圖 3.7 顯示使用 graph cut 擷取前景的比較範例。圖 3.7 中上排為原輸入影像，中排為僅用 global threshold 擷取前景之結果，下排為 global threshold 後再利用 graph cut 擷取前景之結果。由圖 3.7 中可以看到 graph cut 演算法優異的表現。即使因屬於背景的桌面顏色和人臉膚色過於接近而經由 global threshold 後在人臉附近產生許多前景擷取錯誤的像素，在經過 graph cut 的處理之後，仍然能夠以具同質性之像素間的內聚力將此錯誤修正。而且，利用 graph cut 技術所擷取出的前景非常完整，因此也不需要再做其他的形態學處理。

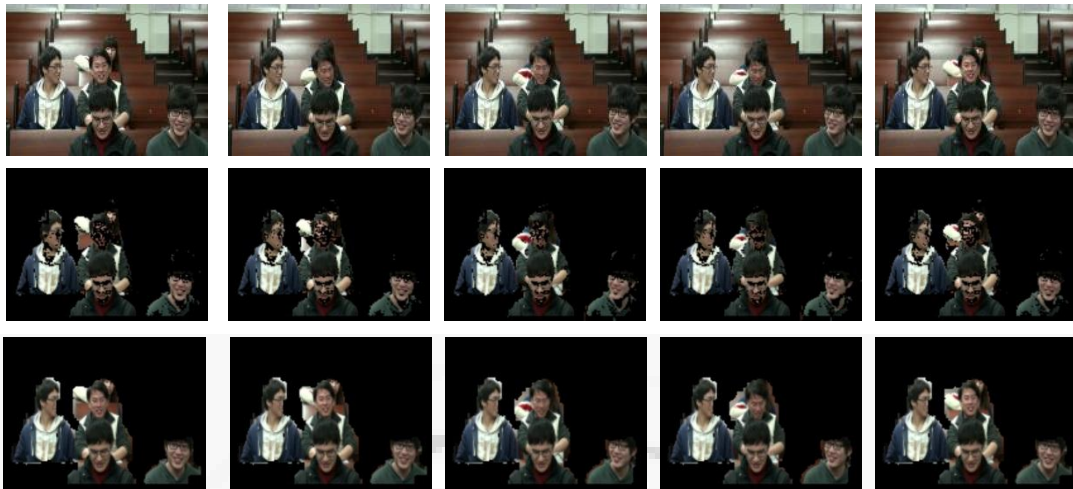


圖 3.7 Graph cut 擷取前景的比較範例。上排為原輸入影像，中排為 global threshold 擷取前景之結果，下排為 global threshold 後再利用 graph cut 擷取前景之結果。

總而言之，本研究利用 graph cut 演算法來進行前景擷取，即將某像素分類為前景或背景。而在下一個階段中亦將利用 graph cut 演算法來擷取前景的膚色區域，即將前景像素分為膚色和非膚色。Graph cut 不只考量了各個像素在色彩空間中的近似程度，更進一步將像素間的“距離”也納入考量，這樣的特性有利於影像中的像素分類，因此能夠達到本研究想要的結果。

第四章 Dynamic Graph Cut

第三章說明如何由 global threshold 去背景的結果提供 graph cut 演算法需要的輸入，包含“hard constraint”和前景背景的機率模型，以此實現 graph cut 演算法的自動化。實驗結果顯示，經過 graph cut 增強的去背景結果完整而且正確，為接下來的膚色偵測提供良好基礎。但缺點是該演算法的時間複雜度很高，拖慢了整體系統執行速度。而且對於每一張輸入影像皆須重新配置一張 graph，在記憶體的利用上也沒有效率。

近年有不少研究希望能夠改進 graph cut 演算法的速度和記憶體利用。其中，一改良的思考方向為：當連續影像輸入時，在不重新配置 graph 的條件下，利用前一張影像所配置的 graph 修改部分 links 的權重，即可計算新 graph 的 minimum cut 並得到正確的結果。Boykov and Jolly 在其論文(Boykov and Jolly, 2001)中就已經有提出一些類似的想法，假設使用者對於 graph cut 的前景擷取結果不滿意，繼續添加其他 hard constraints，他們希望能夠就使用者新增的 hard constraint 局部修改 link 的權重，再計算新的 minimum cut。

若要實現上述的想法，首要考慮的問題是，一個找完 minimum cut 的 graph 是處在有 maximum flow 通過的狀態，而使用者修改權重，即影響某些 links 的容量限制；假設對於這些容量的修改，破壞了原本 flow graph 的結構，其結果就會失去控制。

本章將就這個部分詳細討論。4.1 介紹 max-flow min-cut 的演算法及其時間複雜度；4.2 介紹相關研究和本論文最後使用的演算法—dynamic graph cut；4.3 則為實作的細節和實驗結果。

4.1 Max-flow min-cut 演算法的時間複雜度

根據 minimum cut 與 maximum flow 定理，一個 graph 的 minimum cut 權重和等於 maximum flow 權重和，加上目前為止並沒有直接解出 minimum cut 的演算法，只能透過 maximum flow 演算法取得 minimum cut，因此以下將就 maximum flow 演算法詳細討論。

4.1.1 Ford-Fulkerson 與 residual graph

Maximum flow 演算法的核心是根據在給定一個 graph 以及其上的一個 flow 的條件下建構 residual graph，然後不斷尋找 source 至 terminal 間的路徑增加流量一直到 residual graph 產生瓶頸為止。Residual graph 的定義如下：

Residual graph: 給定一 graph $G=(V,E)$ 和其上的一個 flow f ，令 $c(u,v)$ 為 link (u,v) 上的容量， $f(u,v)$ 為 (u,v) 上的流量。其 residual graph $N(f)$ 之 link 集合定義如下，對於任意 G 上的 link (u,v) ：

1. 若 $f(u,v) > 0$ ，則 $N(f)$ 有 link (v,u) ，權重為 $f(u,v)$ 。
2. 若 $c(u,v) > f(u,v)$ ，則 $N(f)$ 有 link (u,v) ，權重為 $c(u,v) - f(u,v)$ 。

圖 4.1 顯示 residual graph 建構示意圖。左圖中黑色箭頭的寬度表示容量大小、紅色箭頭的寬度表示流量大小，而右圖中綠色箭頭則表示 residual graph 之 link。如圖 4.1 所示，考慮 graph 上的某一 link (u,v) ，其對應的容量如左圖中的黑色箭頭所示，如果有 flow(其流量為 $f(u,v)$) 通過(即如左圖中之紅色箭頭所示)，則在其 residual graph 中就會產生一方向相反，權重大小同樣為 $f(u,v)$ 的 link(即如右圖中之深綠色箭頭所示)。另外，假設 $f(u,v)$ 未達容量上限，則在 residual graph 中會產生一條方向相同，大小為 $c(u,v) - f(u,v)$ (剩餘容量)的 link(即如右圖中之淺綠色箭頭所示)。

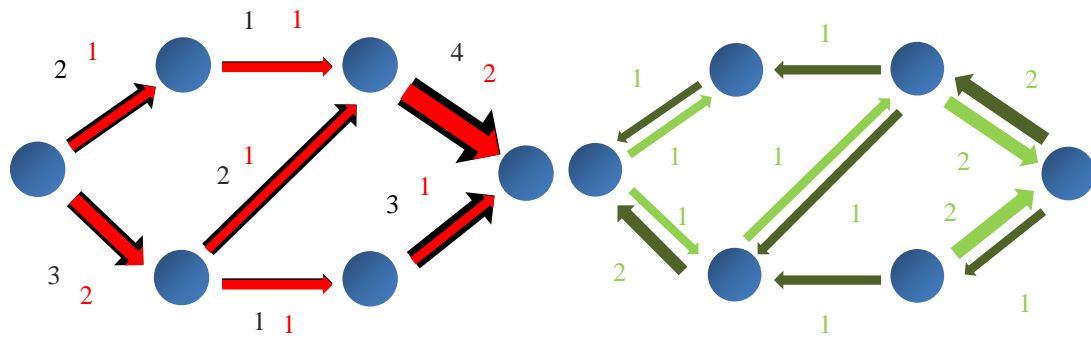


圖 4.1 Residual graph 建構示意圖。左圖中黑色箭頭的寬度為容量大小、紅色箭頭的寬度為流量大小，而右圖的 residual graph 中，深綠的 edge 為反向，與紅色 edge 大小相同；淺綠則為正向，為剩餘容量的大小。

Ford-Fulkerson 證明，只要在 residual graph 中沒有 source 到 terminal 間的路徑存在，此時的流量總和就是 maximum flow。而逆命題亦成立，即由 maximum flow 產生出來的 residual graph，不會有 source 到 terminal 的路徑存在。總而言之，給定一 flow，只需要判斷其 residual graph 中 source 和 terminal 間有無路徑存在，即可得知其是否為 maximum flow。

假設給定一 flow f ，而其 residual graph 中的確存在一條 source 及 terminal 間的路徑 P ，且 P 中最小的 link g 之權重為 $|g|$ ，則必然可以在原來的 flowgraph 中，順著 P 的路徑將 g 流過以得到一更大的 flow 其流量為 $|f + g|$ 。反覆操作，一直到 residual graph 不再有路徑為止，此時的流量就是 maximum flow。演算法如下：

Function Ford-Fulkerson(input $N = (G, S, T)$)

{ Initial flow $f = 0$; Residual graph $R = N$;

While (t is reachable from s) {

 Find a path P from s to t

 Let c be the minimum weight of links on P

 Update flow f by adding c units of flow along P to f

 Update residual graph R depending on new flow f }

```
return f }
```

Ford-Fulkerson 不僅提出了一個尋找 maximum flow 的演算法，其更大的貢獻是在於提出 residual graph 的概念和其與 maximum flow 之間的關係。

4.1.2 Graph cut 和 maximum flow 演算法

Ford-Fulkerson 演算法除了受限於容量必須是整數外，每次搜尋新路徑的時間複雜度為 $O(|E|)$ 。假設找到可能的最多條路徑，其數量為容量的最大值 F ，則在 worst case 下，時間複雜度為 $O(|E| \times F)$ 。因此許多研究者在以 residual graph 為基礎的 maximum flow 演算法中做了許多改善其限制與降低時間複雜度的努力。Edmonds and Karp algorithm 解決了容量必須是整數的限制，在 residual graph 中，每次只找 source 到 terminal 的最短路徑做擴充，最多只要找 $|V| \times |E|$ 次就能夠得到 maximum flow，worst case 下的時間複雜度為 $O(|V| |E|^2)$ ，由原本的指數複雜度降到了多項式時間複雜度。Dinic's algorithm 為 Edmonds-Karp algorithm 的加強版。每次將所有等長的最短路徑一起擴充，只需要做 $|V|-1$ 次就能夠找到 maximum flow，時間複雜度為 $O(|V|^2 |E|)$ 。

以 residual graph 為基礎的 maximum flow 演算法中，Dinic's Algorithm 是已知最效率的方法。然而，由於 graph cut 演算法所建構出來的 graph 有一些更方便的特性存在，例如每一個像素都和 source 與 terminal 恰有一邊相連，且每一個像素只和鄰近的點相連等。Boykov and Kolmogorov (Boykov and Kolmogorov, 2004) 因此發展出專門用於 graph cut 的 maximum flow 演算法，並透過實驗證明該方法在大多數情況中都比 Dinic's Algorithm 更有效率。

該演算法的基本想法可由圖 4.2 說明。首先建立兩顆樹，分別以 source 和 terminal 為根，每次更新 flow 的動作分為三個階段：

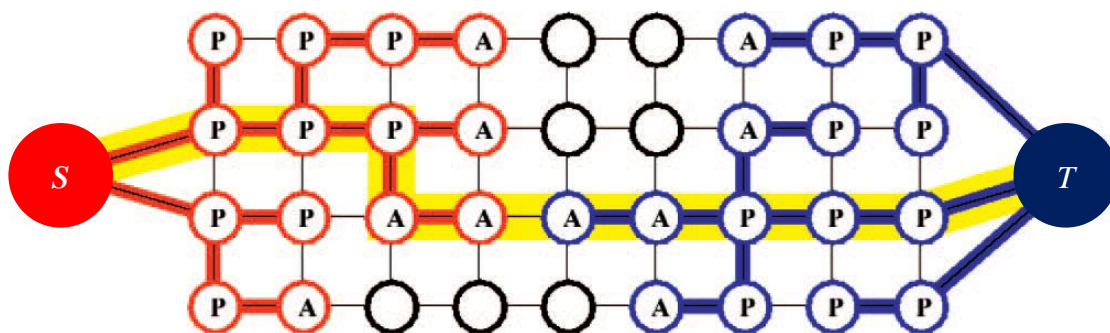


圖 4.2 Boykov and Kolmogorov's algorithm 示意圖

(Boykov and Kolmogorov, 2004)。

1. growth: 以 source 和 terminal 為根的兩棵樹各自擴張，一直到兩棵樹有接觸為止，同時會找到一條 source 和 terminal 間的路徑。
2. augmentation: 以該路徑的最小權重邊更新 flow 和 residual graph，此時兩棵樹會因為 residual graph 的更新而被分離成許多小樹。
3. adoption: 保留 source 及 terminal 為根的樹。

另外，在圖 4.2 中，除了 source 和 terminal 以外，各個 vertex 依照其被標記的情況可分為三類：

1. free: 沒有被標記的 vertex，表示該 vertex 既不屬於 source 樹；也不屬於 terminal 樹。
2. passive: 標示為“P”，表示該 vertex 附近的 vertices 都屬於同一棵樹。
3. active: 標示為“A”，表示該 vertex 附近有 free 的 vertices。

這些標記主要用於 growth 階段，兩顆樹各自從 active 的 vertices 擴張，將 free 的 vertices 納入樹中，並且將內部的 vertices 設為 passive。

觀察前述三個步驟可以發現，Boykov and Kolmogorov's algorithm 與其他演算法主要的不同點在於 $s-t$ 路徑的尋找。不同於 Ford-Fulkerson's algorithm 以 BFS 尋找路徑或者是 Edmonds and Karp algorithm 尋找最短路徑，Boykov and Kolmogorov's algorithm 是以兩棵樹的擴張與交會來找出路徑。雖然因為找到的這條路徑不再確定是最小，其 worst-case 的複雜度上界(upper bound)可能大於

Edmonds and Karp's algorithm 和 Dinic's algorithm 提出的上界。不過，由實驗來看，在各種影像大小之下，Boykov and Kolmogorov's algorithm 的效率都遠勝 Dinic's algorithm。圖 4.3 為 Boykov and Kolmogorov's algorithm 與其他 maximum flow algorithm 處理三種不同影像(Bell photo、Lung CT、與 Liver MR)的時間比較，時間單位為秒。由圖中可知 Boykov and Kolmogorov's algorithm (如圖中最後一列數據所示)在此三種不同影像的處理時間皆為最短。

method	2D examples					
	Bell photo (255x313)		Lung CT (409x314)		Liver MR (511x511)	
	N4	N8	N4	N8	N4	N8
DINIC	2.73	3.99	2.91	3.45	6.33	22.86
H.PRF	1.27	1.86	1.00	1.22	1.94	2.59
Q.PRF	1.34	0.83	1.17	0.77	1.72	3.45
Our	0.09	0.17	0.22	0.33	0.20	0.45

圖 4.3 Boykov and Kolmogorov's algorithm 與其他 maximum flow 演算法處理三種影像的時間比較(Boykov and Kolmogorov, 2004) (單位：秒)。

4.2 Dynamic graph cut

在 4.1 節中提到 graph cut 的 maximum flow algorithm 和 Ford-Fulkerson's algorithm 皆是以 residual graph 為基礎。理論上對一配置好的 graph 進行 graph cut 之後，會得到一個 $s-t$ 不連通的 residual graph。若此時加大某條路徑的容量(也就是流量限制)，則在 residual graph 中也會有相應的路徑出現，於是系統便可以在 residual graph 中繼續擴充 flow，進而找到新的 maximum flow 以及其對應的 minimum cut。

4.2.1 增加新的 hard constraint

Boykov and Jolly (Boykov and Jolly, 2001)在其論文中提供使用者在 graph cut 進行完畢之後，即時增加新的 hard constraint 並再次計算 maximum flow 的介面。考慮一張已完成 graph cut 的影像和一個不屬於 hard constraint 的像素 p 。如果使

用者額外標記此像素為 hard constraint，依照表 3.1 的定義，像素 p 的 t-link 修改前後的權重值應如表 4.1 所示。

表 4.1 使用者增添 hard constraint 前後 p 的 t-link 權重值

t-link	initial cost	new cost
$\{p, S\}$	$\lambda R_p(\text{"bkg"})$	K
$\{p, T\}$	$\lambda R_p(\text{"obj"})$	0

然而，如果此時已經有 flow 通過 $\{p, T\}$ ，將容量設為零就會破壞流量的一致性，產生流量小於容量的矛盾，因而無法得到正確結果。由此例可知，對於一個有 flow 通過的 residual graph 來說，減少容量是可能破壞流量一致性的動作；但相對來說，增加容量就完全沒有此風險。因此，Boykov and Jolly 將最後 link 的權重定義如表 4.2。

表 4.2 以“增加流量”為基礎所定義的 link 權重值

t-link	initial cost	add	new cost
$\{p, S\}$	$\lambda R_p(\text{"bkg"})$	$K + \lambda R_p(\text{"obj"})$	$K + c_p$
$\{p, T\}$	$\lambda R_p(\text{"obj"})$	$\lambda R_p(\text{"obj"})$	c_p

在此定義之下，由於 p 兩端的 t-link 權重值(容量)都增加，因此不會破壞流量的一致性。然而，這樣更新權重並繼續計算 minimum cut 的結果會與重新配置一張 graph 而後重新定義權重的結果相同，其關鍵在於，只要保持像素 p 兩端 t-link 容量的差值，其二者同時加上任意正數皆不會改變 minimum cut 的結果。表 4.3 中即呈現此一概念，既然將 t-link 分別設為 K 和 0 會破壞流量一致性，那就保持兩端的差值 K ，並同時加上 c_p 以維持流量一致性。

此一作法的原理在於，像素 p 兩端的 t-link 形成一條顯然的 $s-t$ 路徑。不管將兩端 t-link 的容量同時增加多少，都可以直接將這個平移數值當做流量通過此

路徑且不會影響其他路徑中的流量分布；而剩餘的容量，也就是此二容量間的差值，才是真正對於尋找 minimum cut 的解有貢獻的部分。

4.2.2 Graph 再參數化(reparameterization)

Boykov and Jolly 所提出 4.1.2 節的這個概念，對 t-link 的性質有較為深入的說明。簡單來說，某像素 p 其兩端的 t-link 權重只在差值的部分會對 minimum cut 的結果產生影響，在固定權重差值的情況下，將此二權重同加或同減一個值，並不會改變 minimum cut 的結果。因此，如果想要修改某像素 p 的 t-link 權重值，卻又不希望破壞流量的一致性，僅需要在兩條 t-link 上同時加上一個夠大常數即可。

Kohli and Torr(Kohli and Torr, 2007)提出了 dynamic graph cut 演算法，將前述在 residual graph 上修改 t-link 的想法延伸至 n-link，並將此修改 link 的容量而不會破壞流量一致性的方法統稱為 graph 再參數化(reparameterization)。

給定一個影像標記 A ，將一組相應的 link 權重定義以 θ 表示，並以 $E(A|\theta)$ 代表其成本函數值。給定兩組 link 的權重定義 θ_1 和 θ_2 ，若對於任意的 A ，皆符合 $E(A|\theta_1) = E(A|\theta_2)$ ，則此兩組 link 的權重定義可稱為對方之 graph 再參數化。

根據上述的說法，若 θ_1 和 θ_2 互為對方的 graph 再參數化，此二者之 minimum cut 結果必然相同，即

$$A^* = \arg \min_A E(A|\theta_1) = \arg \min_A E(A|\theta_2)$$

此概念在本論文中已有多次提及，而 Kohli and Torr 則以“再參數化”之命名賦予其較為嚴謹的數學定義。

接下來說明 dynamic graph cut 演算法的內容。Dynamic graph cut 演算法的目標是能夠修改 t-link 和 n-link 的容量，即便在修改之後與原來的 flow 產生衝突，也能夠透過 graph 再參數化的動作，解決此一衝突。

1. t-link 容量的修改方式：修改 t-link 之流程可以圖 4.4 說明。當一個 graph 完成 minimum cut 之後，應無 $s-t$ 路徑存在，由於 t-link $\{p, S\}$ 之剩餘容量為零，

代表有相當於其容量大小的流通過。在圖 4.4(b)中，首先將 t-link $\{p, S\}$ 的容量減 3，t-link $\{p, T\}$ 之容量減 2，因而 $\{p, S\}$ 的容量產生了負值，破壞了流量一致性。解法為，將 $\{p, S\}$ 和 $\{p, T\}$ 的容量同時加 3(此例中 $a=3$)，兩 t-link 的剩餘容量分別變成 0 和 5，流量即不再有衝突的現象發生。簡單來說，修改流量之後，只需要將 p 兩邊的 t-link 權重值一起平移一個正數以解決 t-link 的容量產生負值的情況即可。

2. n-link 容量的修改方式：修改 n-link 之流程可以圖 4.5 說明。圖 4.5(a)中，由於 n-link $\{q, p\}$ 之權重下修 3，因而得到 -1 的容量，破壞了流量一致性。解法是在 $\{q, p\}$ ， $\{p, T\}$ 和 $\{q, S\}$ 三者的容量加 a 使得負的容量變成 0(此例中 $a=1$)，等同讓 a 的流由 t 逆向流回 s 。最後，為了維持流量的結構，將 $\{q, p\}$ 的容量扣掉 a ，再參數化便完成。此時雖然做了容量相減的動作，但是由於 a 的大小為修改前後的容量差值，不可能大於原來的容量，因此減去 a 並不會使容量產生負值。

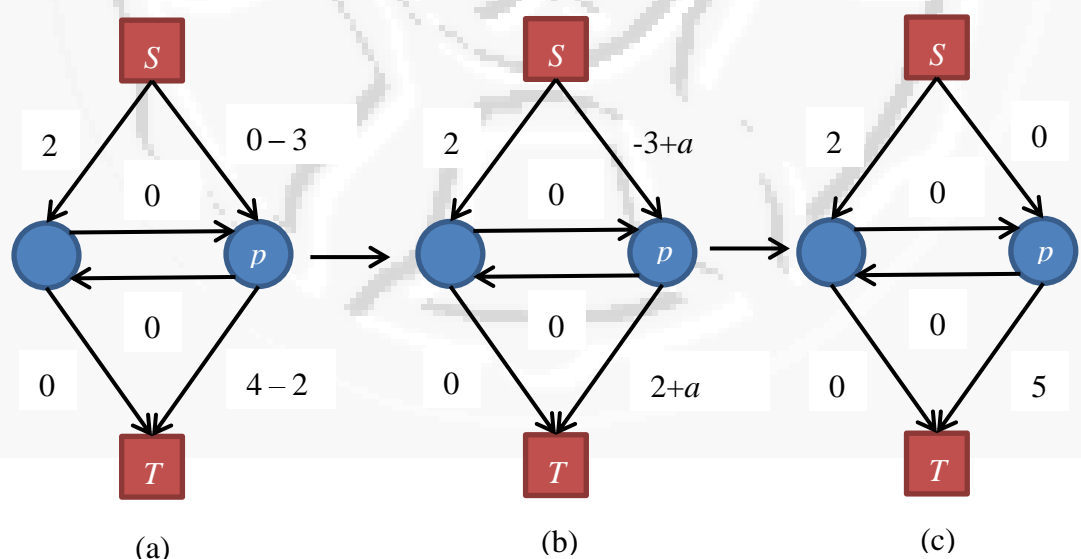


圖 4.4 在 residual graph 上修改 t-link 容量之示意圖(a)完成 minimum cut 之 residual graph(b)修改 t-link 容量後之 residual graph(c)再參數化後之 residual graph

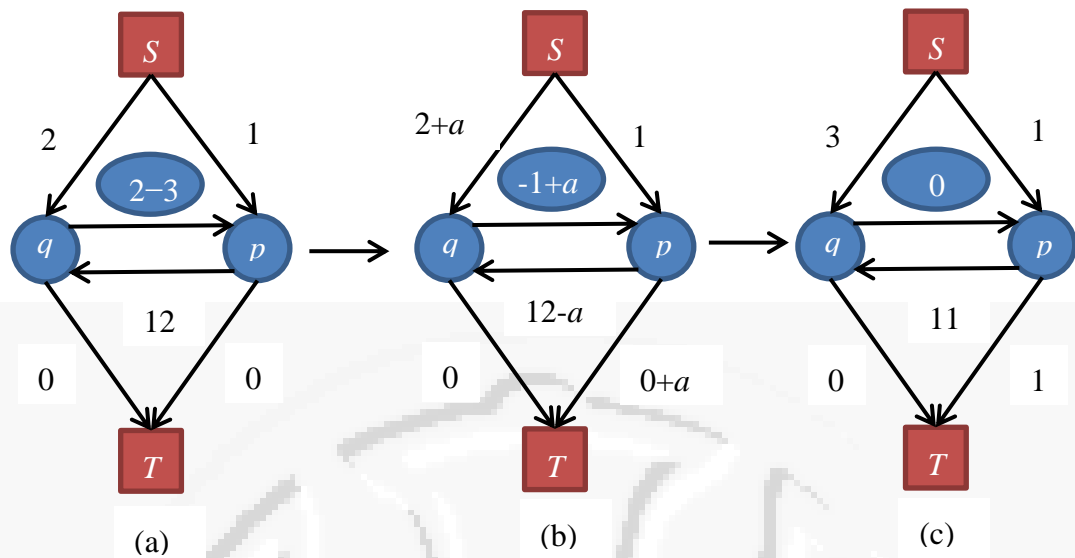


圖 4.5 在 residual graph 上修改 n-link 容量之示意圖(a)完成 minimum cut 之 residual network(b)修改 n-link 容量後產生負值之 residual graph(c)再參數化後之 residual graph

Dynamic graph cut 演算法提供了修改 n-link 和 t-link 的機制，這讓系統可以只在第一張輸入影像時建立 graph。第一張之後的影像只需要根據 dynamic graph cut 的規則依序修改各 link 的權重，便能夠再一次計算 minimum cut。

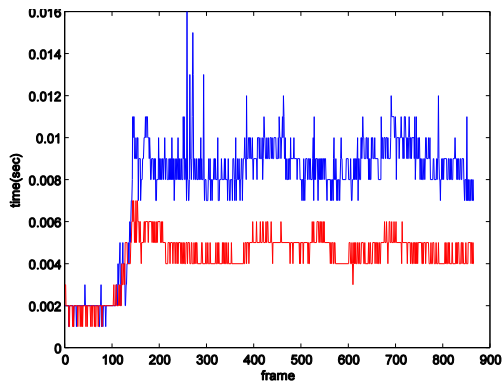
圖 4.6 為使用 dynamic graph cut 演算法的加速結果。圖 4.6 (a)(b)為兩個不同連續影像的第一張影像，而圖 4.6 (c)(d)分別為圖 4.6 (a)(b)中輸入影像編號執行 maximum flow 演算法所需執行時間。藍色部分為執行 graph cut 演算法所需時間，而紅色部分為執行 dynamic graph cut 演算法所需時間。由圖中可以觀察到，經過 dynamic graph cut 演算法的加速，每張輸入影像所需的尋找 maximum flow 的時間幾乎可以省去一半左右，而且，在不同影像中所花的時間也較穩定，變異較小。



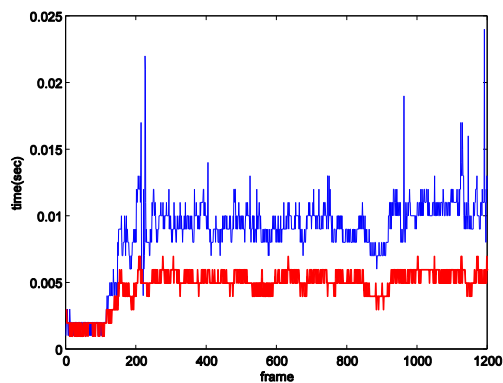
(a)



(b)



(c)



(d)

圖 4.6 Dynamic graph cut 演算法的加速情形。(a)(b)為兩個不同連續影像的第一張影像，(c)(d)分別為(a)(b)中輸入影像編號執行 maximum flow 演算法所需執行時間。藍色部分為執行 graph cut 演算法所需時間用，紅色部分為執行 dynamic graph cut 演算法所需時間。

雖然 dynamic graph cut 能夠有效的加快尋找 maximum flow 的速度，不過在每一張輸入影像中依然需要重新計算每一個像素的 t-link 和 n-link。如果能夠進一步增加一些條件限制，使得在影像變動不大的狀況下僅需修改部分 link 的權重，如此就可以節省更多的執行時間。

本研究以連續影像相減的 temporary difference 作為特徵，系統只修改相鄰兩張影像的同一像素其 intensity 差值夠大的 link 權重值，之後再進一步尋找 maximum flow。若相鄰兩張影像的同一像素其 intensity 差值不夠大，系統便視其為不需更動 link 權重值的像素；此門檻值很自然的選定前景擷取階段計算出來的全域門檻值。圖 4.7 下排影像中的白色像點即為需修改 link 權重值的像點。

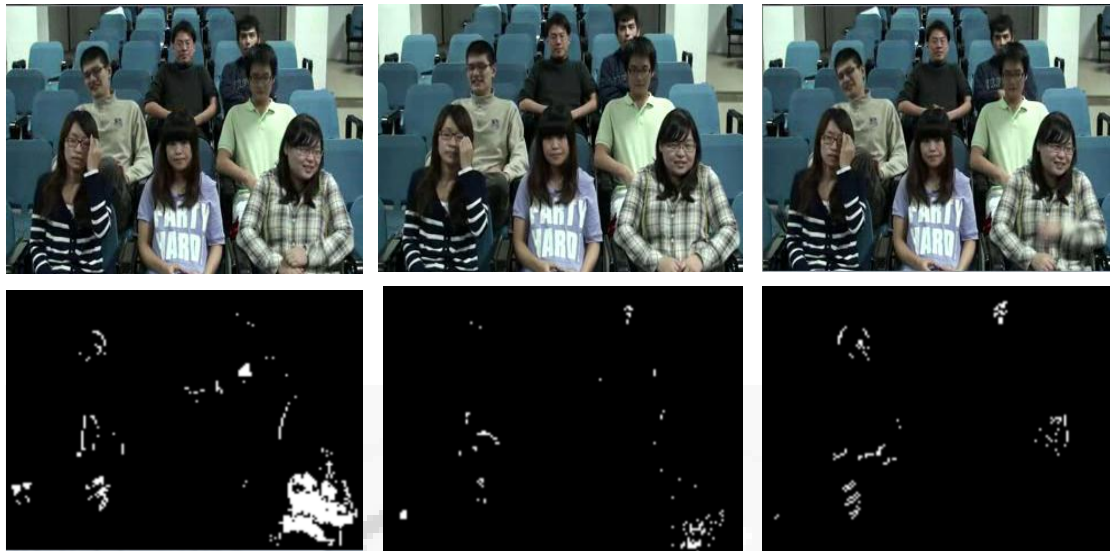


圖 4.7 Temporary difference 之範例。上排為原影像，下排為取過全域門檻值的 temporal difference 影像。

圖 4.8 為使用 dynamic graph cut 演算法且增加 temporary difference 判斷機制的實驗結果，其中紅色折線為使用 graph cut 演算法尋找 max-flow 的執行時間；綠色折線為使用 dynamic graph cut 演算法尋找 max-flow 的執行時間；藍色折線為使用 dynamic graph cut 演算法且增加 temporary difference 判斷機制的執行時間。經由計算結果，加入 temporary difference 的判斷機制之後雖然在總執行時間上有所節省，但是影像與影像間的執行時間變異反而增加。這是由於以全域門檻值決定欲修改的像素無可避免地產生些許雜訊，因此系統必須在欲修改的像素數量過大之時，重新進行整張影像的 dynamic graph cut 以穩定前景擷取結果。然而在這樣的切換過程中，會讓 max-flow 的尋找產生較大的 overhead，而為了系統的穩定性，此欲修改的像素數量之門檻值僅設為百分之一。圖 4.8 中，使用 dynamic graph cut 演算法尋找 max-flow(以綠色折線表示)和加入 temporal difference 判斷機制(以藍色折線表示)的平均執行時間比約為 6:5，約能達成百分之十六的加速。

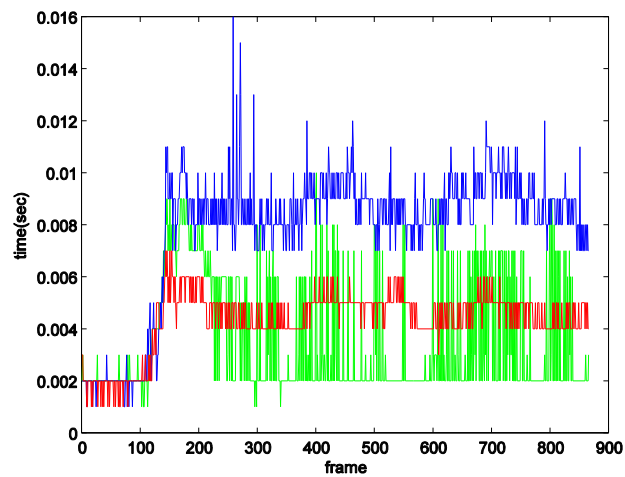


圖 4.8 使用 dynamic graph cut 演算法且增加 temporary difference 判斷機制的實驗結果。紅色為使用 graph cut 演算法尋找 max-flow；綠色為使用 dynamic graph cut 演算法尋找 max-flow；藍色為使用 dynamic graph cut 演算法且增加 temporary difference 判斷機制的實驗結果。

第五章 膚色區域偵測與人臉區域選取

以 dynamic graph cut 演算法得到前景區域之後，接下來必須在前景影像中偵測膚色區域並且進而選取人臉區域。在膚色區域的偵測中，本研究首先採取 Zhang *et al.*(Zhang *et al.*, 2010)所提出的固定範圍偵測膚色；然後根據膚色的 hue channel 在影像中高度集中的特質，以 dynamic graph cut 演算法將膚色區域完整化；最後，為了適應各種教室的光線變化，當膚色區域在影像中所佔比例夠大，也就是學生數量夠多之後，系統便會在做完 dynamic graph cut 演算法之後的膚色區域中進行統計，並且更新膚色的範圍。

在每一張輸入影像中系統會偵測到許多膚色區域，然而這些膚色區域並非全部都是人臉區域，也包含了腿部、手部和一些雜訊區域。根據本研究觀察，人臉膚色區域與其他膚色區域之最大不同，除了形狀、大小以外，最明顯之處在於人臉區域上的邊緣(edge)特徵無論是方向或大小的分布都較為複雜，而在其他的膚色區域之上邊緣的分布則相對較單純。視覺上來說，人臉區域由於高低起伏較為明顯，因此光影變化亦較豐富；非人臉區域則相對平坦。

因此，本研究在多個膚色區域中，除了以形狀大小等特徵濾掉非膚色區域之外，會在一塊膚色區域進行 Sobel 邊緣偵測，統計該膚色區域中每一個像素的 gradient 強度以及方向。最後，以膚色區域的平均邊緣強度以及方向的變異大小來決定某膚色區域是否為人臉膚色區域。

5.1 膚色區域偵測

本研究所參考的膚色範圍為 Zhang *et al.*(Zhang *et al.*, 2010)所提出 $0^\circ \leq H \leq 50^\circ$ ， $0.2 \leq S \leq 0.68$ ， $0.35 \leq V \leq 1$ 。選擇 HSV 色彩空間的原因主要有兩點：

1. Dynamic graph cut 演算法中使用的特徵亦為 hue channel，選擇此色彩空間的膚色範圍對於 dynamic graph cut 演算法來說較符合一致性。
2. 根據本研究觀察，膚色在 hue channel 的集中情形相當良好(圖 5.1)，同樣是膚色區域，雖然因為光影分布在 RGB 色彩空間產生了许多不一樣的顏色變異，但在 hue channel 表現出來的變異程度則相當低，符合 dynamic graph cut 演算法所假設的情境。

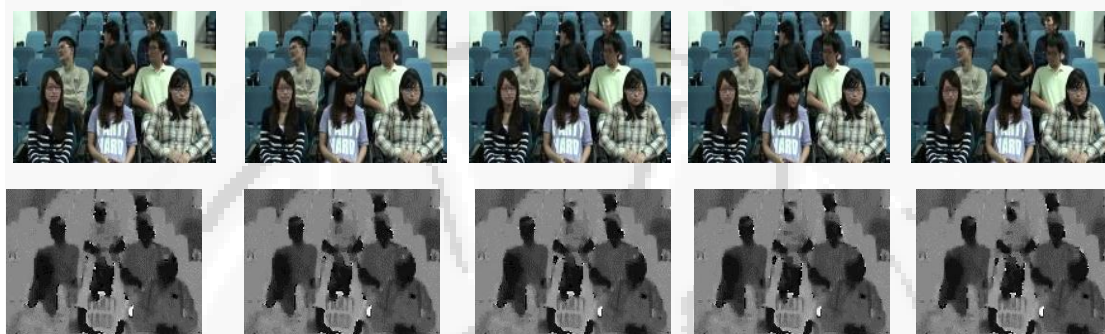


圖 5.1 Hue channel 影像範例。上排為原始影像，下排為 hue channel 之 intensity 值之影像。

當前景所占面積比例到達一定程度時，系統便將上述固定範圍套用於前景中以初步偵測膚色範圍。然而，由圖 5.2 可見此時偵測出來的膚色區域不完整且充斥雜訊。



圖 5.2 以 Zhang *et al.* 所提出之固定範圍偵測膚色之實驗範例。

接下來如同前景擷取的階段，系統必須提供 dynamic graph cut 演算法所需要的三個輸入——hard constraint 以及膚色和非膚色的機率模型。Hard constraint 的部分，由於人臉區域大多是直立長方形，因此本系統以 3×2 pixels 的 patch 來決定膚色的 hard constraint，並且在剩餘的前景中取 3×3 pixels 的 patch 當作非膚色的 hard constraint。

至於膚色的機率模型，本研究統計利用固定膚色範圍初步偵測到的膚色像素在 hue channel 的 intensity 值，建立 histogram 並且做簡單的平滑化動作，而得到膚色的機率模型，並將其記為 $P(x|skin)$ ；非膚色的機率模型，則單純以 $1 - P(x|skin)$ 取得。實驗範例如圖 5.3 所示，dynamic graph cut 除了能夠將膚色區域完整化之外，也能夠藉此消除極小的雜訊區域。由圖 5.3 可觀察得知完整化的人臉區域，保留了臉上的特徵和形狀，更利於接下來的人臉偵測。

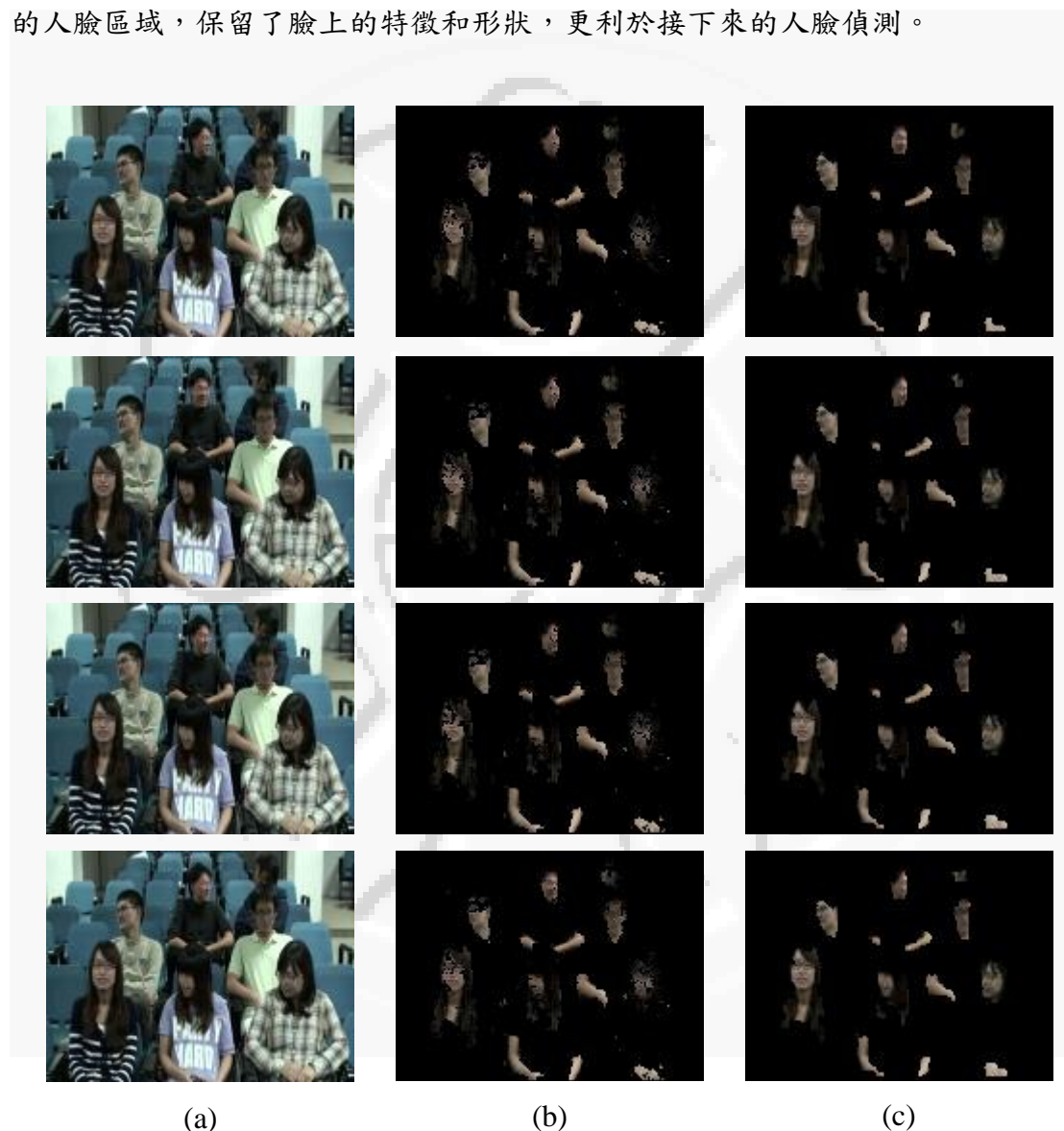


圖 5.3 經過 dynamic graph cut 增強之膚色區域。(a)原始輸入影像(b)固定範圍偵測之膚色區域(c)經由 dynamic graph cut 增強之膚色區域

儘管 dynamic graph cut 演算法能夠將偵測到的膚色區域完整化，然而，初始時所仰賴的固定膚色範圍畢竟無法適應各種教室以及天候的照明情況。因此，本

研究在此提出一個動態學習膚色範圍的方法，能夠隨著教室中環境的改變，更新膚色範圍，以提高膚色區域偵測的正確率。

根據本研究觀察，儘管在各種教室的照明情況下膚色的 hue channel 之 intensity 值分布不全然相同，在 0 到 100 度之間卻經常呈現類似 Gaussian distribution 的分布，如圖 5.4 所示。圖 5.4 為三種不同背景及照明狀況的教室，由圖 5.4 (b)可清楚看見這三間教室中的膚色以肉眼來看有很大的不同。上排的影像中膚色稍微偏綠褐色，中間則是略紅，下排影像中之膚色則偏向白色。在這三種不同類型的膚色 hue channel 之 intensity 分布中，左側峰的位置和集中程度皆不相同，也就是此三者分別近似於不同平均數和變異數的 Gaussian 分布。

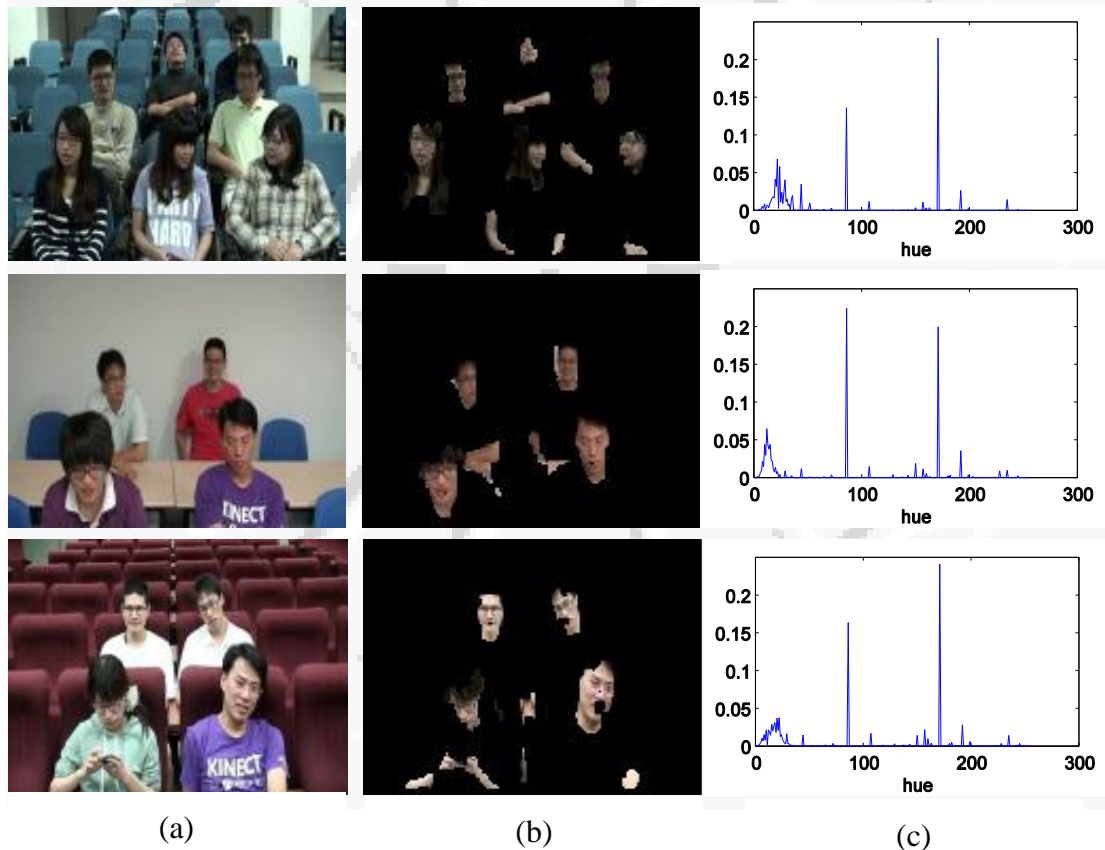


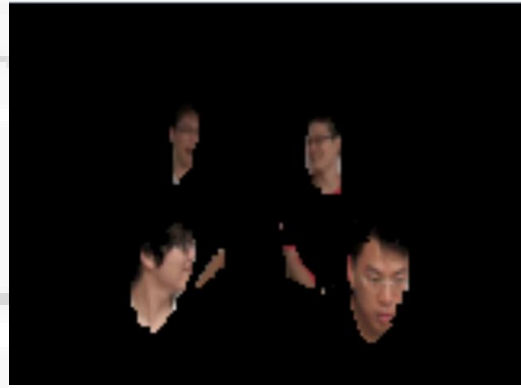
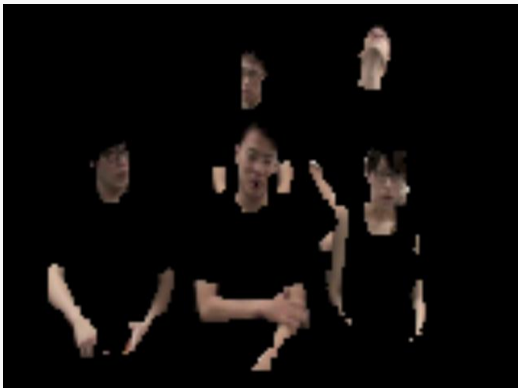
圖 5.4 各種教室和照明狀況下膚色之 hue channel 之 intensity 值分布。(a)原影像 (b)膚色區域(c)膚色區域之 hue channel 之 intensity 值分布圖

因此，本研究以估計圖 5.4(c)中左側 Gaussian 分布的平均數和變異數的方式，搭配信賴區間的概念來更新膚色範圍，其步驟如下：

1. 統計已完成 dynamic graph cut 之後膚色區域中像素的 hue channel 之 intensity 值，並建立 histogram。
2. 以左側峰中機率前三大的 bins 取平均，作為估計的平均值，記為 μ_s 。
3. 由平均值往左出發，統計 47.7% 的比例以估計變異數 σ_s 。
4. 以 $[\mu_s - 2\sigma_s, \mu_s + 2\sigma_s]$ ，即 95.4% 之信賴區間估計新區間。

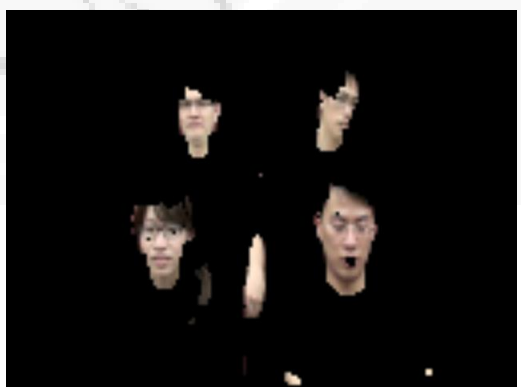
在實作中，由於 hue channel 之 intensity histogram 的統計並不穩定，如果僅以 $[\mu_s - 2\sigma_s, \mu_s + 2\sigma_s]$ 作為新的區間，很容易因為某幾張影像的雜訊使得區間迅速發散，讓結果失去控制。因此，對於區間的更新必須給予一些限制。本系統的作法為，將新區間之上下界與舊區間之上下界進行比較，舊區間之上界與下界一次都只能往新區間的方向移動一個 bin；更甚者，本系統限制區間的總寬度不能超過 60，若區間寬度已達到最大值，則更新方式改為比較新舊區間中點，舊區間中點往新區間中點平移一個 bin。

圖 5.5 展示了在各種教室照明狀況下動態學習而得的膚色範圍之偵測結果。圖 5.5 之(a) (b) (c)和(d)之最終膚色範圍分別為 $[-1, 46]$ 、 $[-9, 45]$ 、 $[7, 61]$ 、以及 $[-3, 51]$ 。由圖中可以看出上述的方法可以改良初始值固定膚色範圍的缺點，進而使得膚色範圍可依輸入影像之不同而適當的調整，且調整的結果有助於影像中膚色區域的偵測。本研究實驗了 5 間不同背景組成的教室，在一般照明情況之下，膚色區間會漸漸隨著學生人數穩定而趨向穩定。而在不同教室中，這些區間的上下界皆有顯著差異。



(a)

(b)



(c)

(d)

圖 5.5 在各種教室背景下膚色的 intensity 值區間。

(a)[-1, 46](b)[-9, 45](c)[7, 61](d)[-3, 51]

上述的結論與 Sigal *et al.* (Sigal *et al.*, 2004)所提出的實驗結果類似，即膚色在 HSV 色彩空間的分布不但有聚集現象，且在各種照明情況之下，其分布之形狀並不會產生太多結構上的變化(如分裂)，這個特性有利於膚色分布的追蹤以及調整。

本研究在膚色區域偵測的階段，使用 Gaussian distribution 來逼近膚色像素 intensity 值之分布，並利用估計出來的平均值和變異數計算膚色範圍。Dynamic graph cut 演算法在這個階段扮演了重要角色，首先是完整化膚色區域、去除雜訊；而後，經由該演算法增強之後的膚色區域能夠用來微調膚色像素 intensity 值之分布，以便於後續的膚色範圍更新等動作。經由實驗結果顯示，膚色範圍在實驗的影片中都會收斂到一個穩定的區間，這證明了在不同教室和照明情況下，膚色的 hue channel 之 intensity 值之分布改變可以大致以 Gaussian distribution 的平移和縮放來近似。

5.2 人臉區域選取

到目前為止，本系統在教室影像中偵測了多塊膚色區域，接下來的工作則是在多個膚色區域中挑選人臉的膚色區域。

人臉膚色區域與其他膚色區域顯著的不同點有以下幾項：

1. 形狀：人臉膚色區域的形狀大都可以橢圓近似，並且為實心聚集的形狀。
2. 長寬比：人臉膚色區域的長寬比不會太大，相對來說，手臂與腿部的長寬比則會有較大的數字出現。
3. 邊緣：人臉區域由於有五官分布，紋理較複雜；而手臂與腿部區域，一般來說較人臉平滑很多。

圖 5.6 以 Harris corner detector 來將上述第 3 點之概念視覺化，Harris corner detector 為一偵測特徵點的 filter。影像中特徵明顯的點(如眼角、嘴角)，在 Harris corner detector 時會得到較大的值。由圖中可以觀察到，人臉區域經由 Harris corner detector 後，呈現高強度 corner 反應的像素位置上分布較廣；反觀手臂與腿部區

域，僅在接近邊緣部分呈現較高強度 corner 的反應，大部分區域的 corner 皆不明顯。

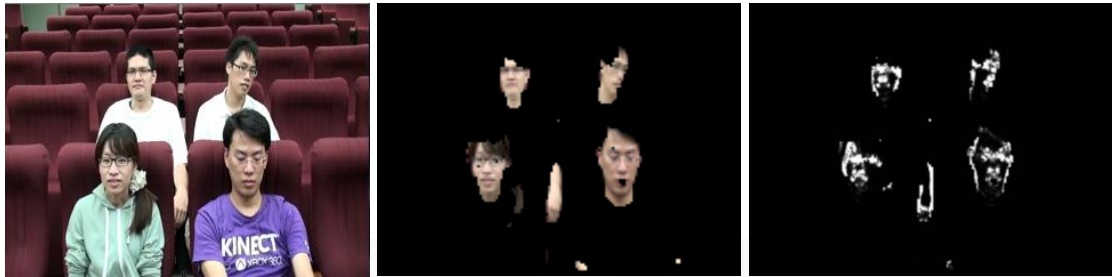


圖 5.6 Harris corner 在人臉膚色區域上的反應。

依據上述觀察，本系統先以形狀、大小和長寬比初步濾掉一些雜訊區域和明顯為手臂與腿部的區域，剩下的區域便為候選的人臉區域。在這些候選的人臉膚色區域中，系統希望能夠將表面較為平坦的區域去除，保留紋理較複雜的區域，藉此篩選出人臉區域。

1. 膚色區域表面的邊緣反應平均值：以 Sobel filter 統計某膚色區域內所有像素之水平和垂直方向的 gradient，並統計 gradient 的強度平均值。假設某膚色區域為 R ，像素為 p ， I_p 為 p 之灰階值，則該膚色區域之 gradient 強度平均值可以下式表示：

$$\mu = \frac{1}{|R|} \sum_{p \in R} \sqrt{\left(\frac{dI_p}{dx}\right)^2 + \left(\frac{dI_p}{dy}\right)^2}$$

2. 膚色區與表面的邊緣方向變異數：人臉區域除了邊緣強度較大之外，其邊緣的方向分布也較複雜。本系統以 $\frac{dI_p}{dy}$ 和 $\frac{dI_p}{dx}$ 之比值作為 gradient 之方向並計算其 arctangent 函數值，該值代表正負 π 間的角度，在此記為 θ_p 。為了抵抗雜訊，將這個角度乘上該像素 p 之 gradient 強度作為加權，並計算此值在該膚色區域中之變異數：

$$\sigma_\theta^2 = \frac{1}{|R|} \left(\sum_{p \in R} (w_p \theta_p)^2 - |R| \times \mu_\theta^2 \right)$$

其中 $w_p = \sqrt{\left(\frac{dI_p}{dx}\right)^2 + \left(\frac{dI_p}{dy}\right)^2}$ ， μ_θ 為 $w_p \theta_p$ 之平均值。

本系統於各個候選區域膚色中分別計算此二特徵，並分別取一適當門檻值將不符合條件的膚色區域淘汰，以完成人臉膚色區域選取，實驗結果將於第六章討論。



第六章 實驗結果與討論

本研究實驗於五種教室環境，教室編號分別為 S101、C209、B1 演講廳、B101 和 B103 五間教室，其中 B101 和 B103 為膚色背景，其餘三間則否。圖 6.1 為各間教室的背景圖，除了背景是否為膚色之外，教室的採光也扮演重要角色。B101 和 B103 接受的自然採光較多，因此在不同時間點拍攝的影片其照明情形也有較大差異。以下實驗結果依照去背景、膚色區域偵測和人臉區域選取分節討論之。

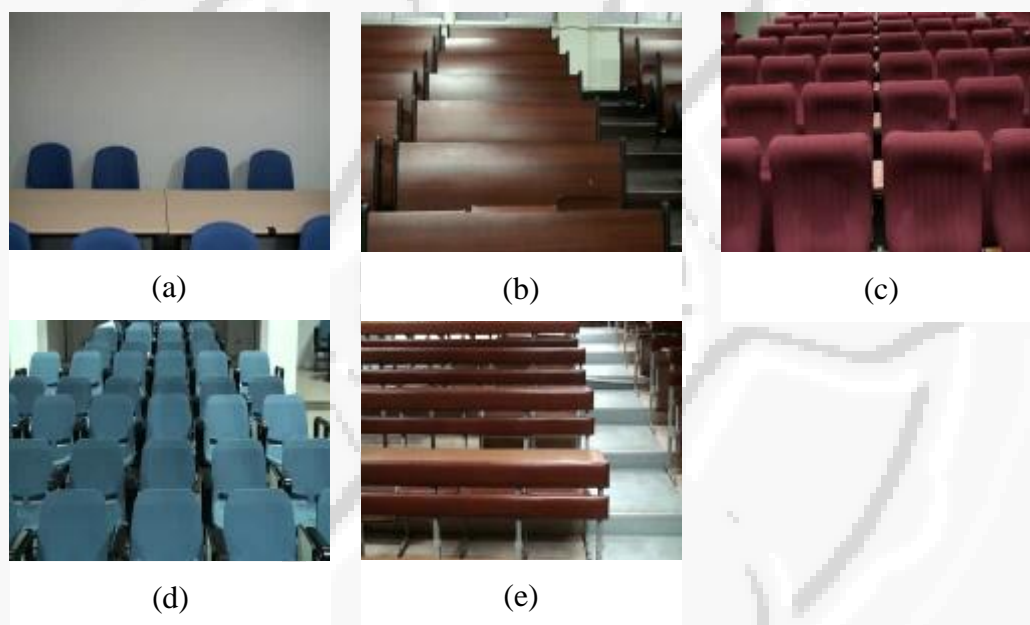


圖 6.1 實驗所使用五間教室之背景圖(a)C209(b)B103(c)S101(d)B1 演講廳(e)B101

6.1 去背景實驗結果

在此展示的去背景實驗影片一共八組，除了各間教室的正常採光情形之外，並且測試了 B1 演講廳、S101 和 B101 在較為陰暗的採光之下的實驗結果。正確率的計算方面，本研究採取以像素為基礎的 precision-recall 和 F-measure 計算，並以自製的 ground truth 做為參考標準。由於影像數量龐大，因此本研究依照各個影片的時間長短，以 50 或 100 張影像為間隔取樣製作 ground truth。

表 6.1 前景擷取之實驗結果


影片	教室	照明情況	影像張數	膚色背景	Precision	Recall	F-measure
 265	S101	陰暗	1304	否	91.11%	84.43%	0.8752
 266	B101	略暗	4488	是	92.07%	95.33%	0.9366
 288	B1 演講 廳	正常	866	否	85.65%	97.18%	0.9105
 292	B102	正常	1200	是	88.65%	91.13%	0.8984
 301	C209	正常	5641	否	88.91%	91.85%	0.9033
 303	S101	正常	5226	否	85.95%	97.00%	0.9110
 307	B1 演講 廳	略暗	6227	否	90.40%	95.36%	0.9264
 310	B103	正常	5343	是	91.27%	89.91%	0.9052

表 6.1 列舉了前景擷取所使用的實驗影片和其正確率，以下將依照各影片編號分別說明：

265: 此為 S101 教室中將前排日光燈盡數關閉所拍攝的影片，以本研究提出的前景擷取方法來說亦為難度最高的測資。由於光線極為陰暗，所擷取出的前景邊界較為模糊，也因此造成了 recall 偏低的情況。

- 266: B103 教室中的日光燈分為三排，此影片的照明情況為關閉左右兩排只剩中間，由於該教室的自然採光程度高，因此對於亮度的影響較小，反而是在色系方面更偏黃色。此影片的對比度高，色彩也較鮮明，對於前景擷取來說相對容易，僅是在影像右上方的背景與學生的衣服顏色較接近，因此在該處附近的實驗結果會較不穩定。
- 288: 該影片的人數最多，座位方式也最擁擠。由於 graph cut 演算法的特性，在前景中的細碎背景區塊都被吞噬，因此造成了 precision 較低的結果。
- 292: 與 266 一樣使用 B103 教室，因為日光燈全開，因此白平衡方面比較正常，前景擷取較困難的部分為右下角頭髮和階梯重疊的區域。
- 301: C209 教室的背景為純白色，因此去背景的正确率也較高，主要發生錯誤的地方為左上角學生的襯衫區域，由於顏色與背景太過接近所致。
- 303: S101 教室中日光燈全開的情形，白平衡偏白。由於顏色鮮明因此正确率較高。
- 307: B1 演講廳中關前排燈光的情形，模擬教師以投影片上課的情況。雖然因此整體亮度偏暗，不過並未劇烈影響正确率。
- 309: B101 教室的桌子高，但視覺上空隙甚多，因此在影像下半部的前景區域相當破碎，這對於本研究的方法來說是屬於比較困難的情況。因此，儘管顏色鮮明對比高，recall 依然是偏低。

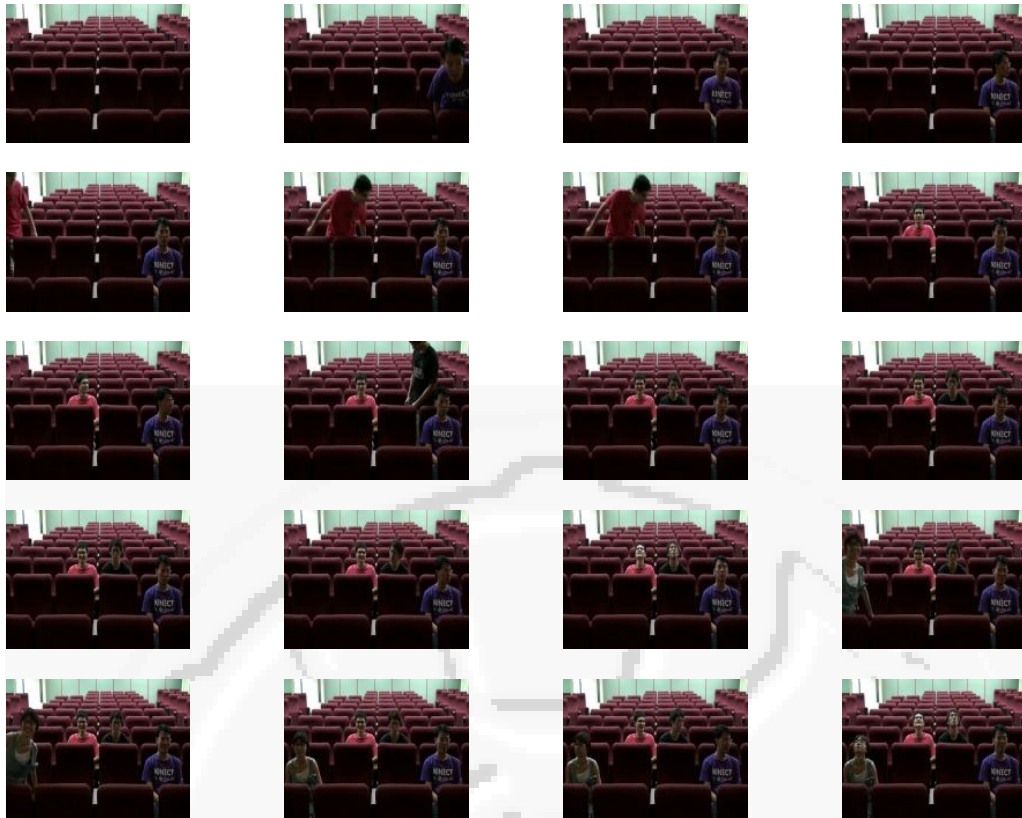


圖 6.2 影片 265 之輸入影像

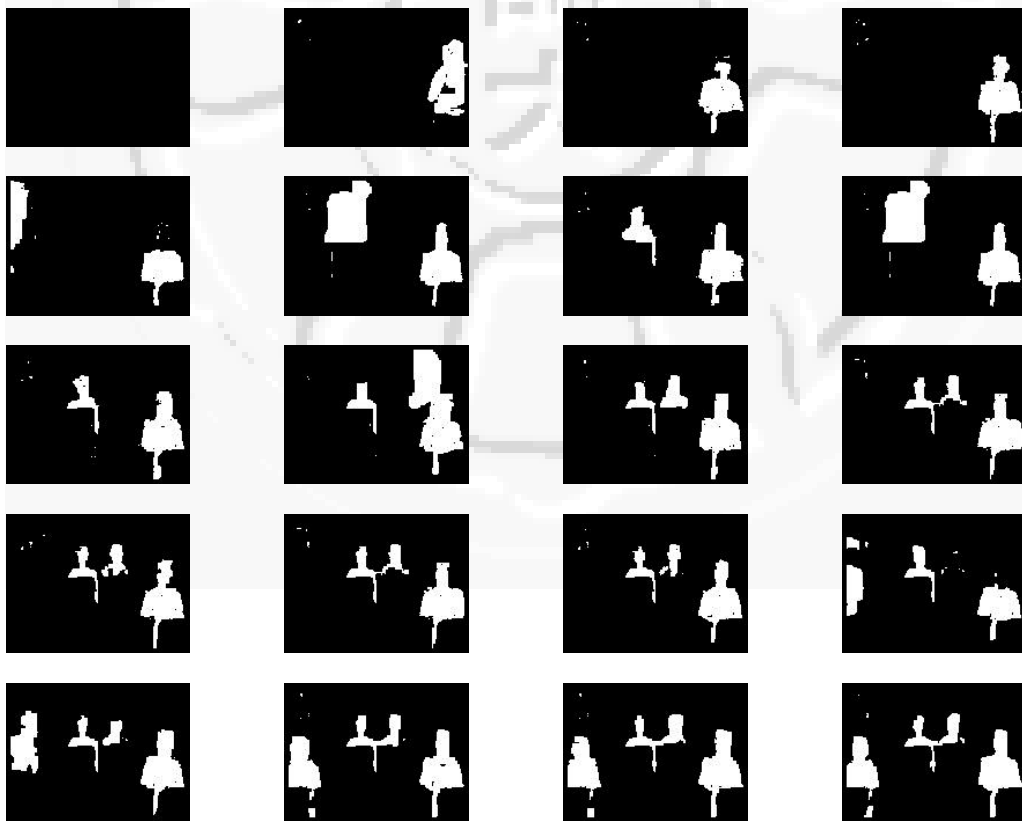


圖 6.3 影片 265 之前景擷取實驗結果

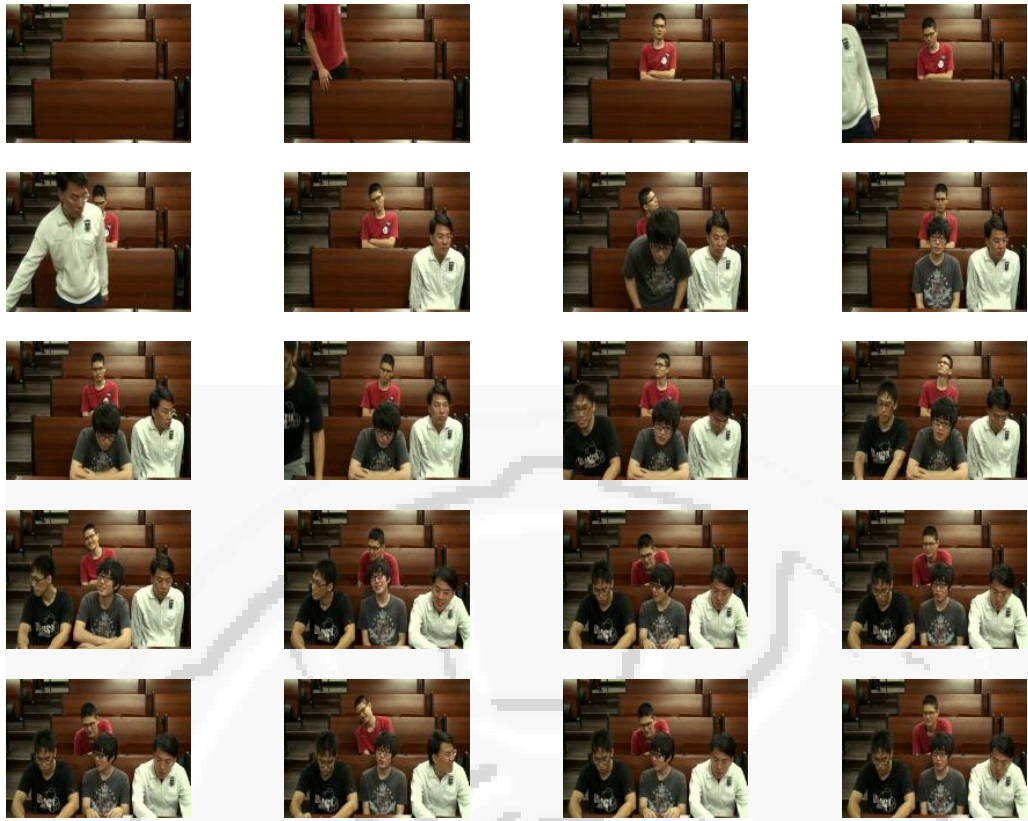


圖 6.4 影片 266 之輸入影像

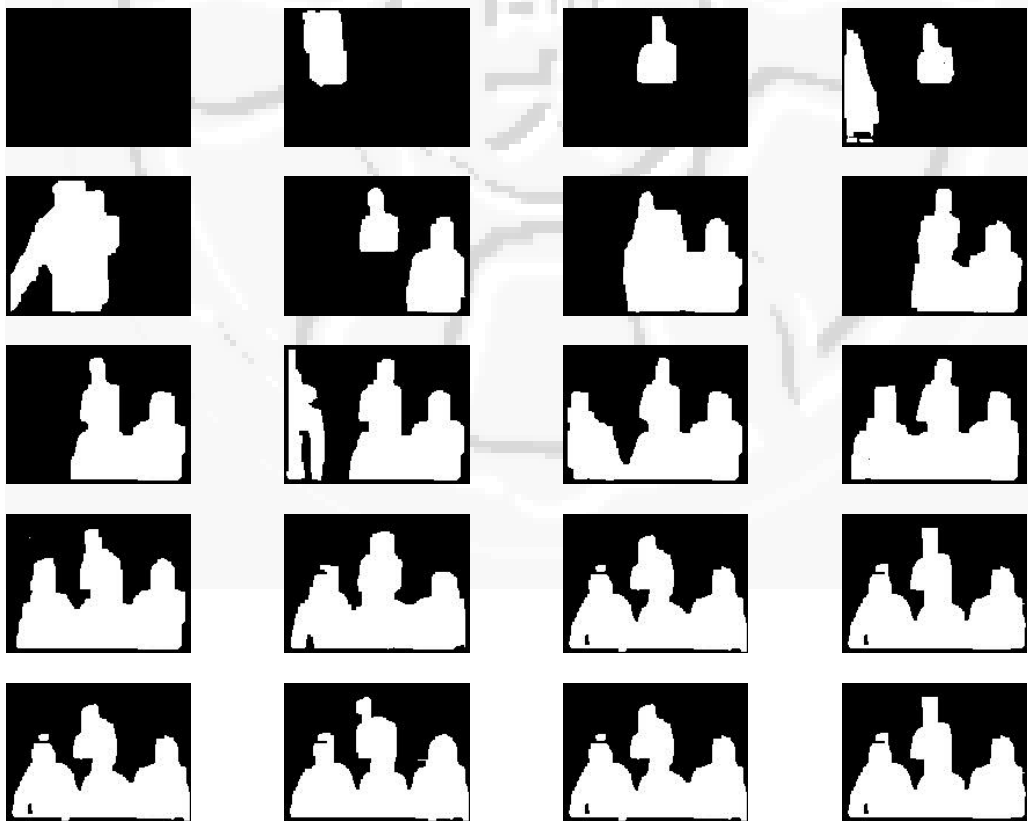


圖 6.5 影片 266 之前景擷取實驗結果

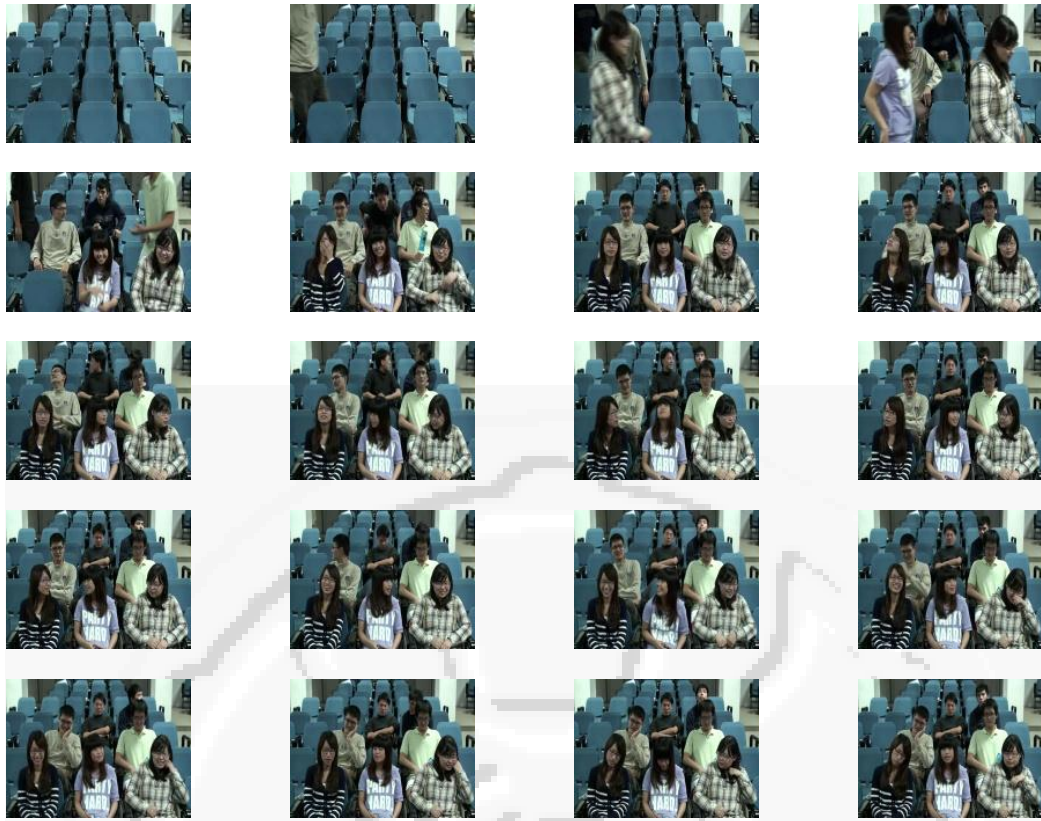


圖 6.6 影片 288 下之輸入影像

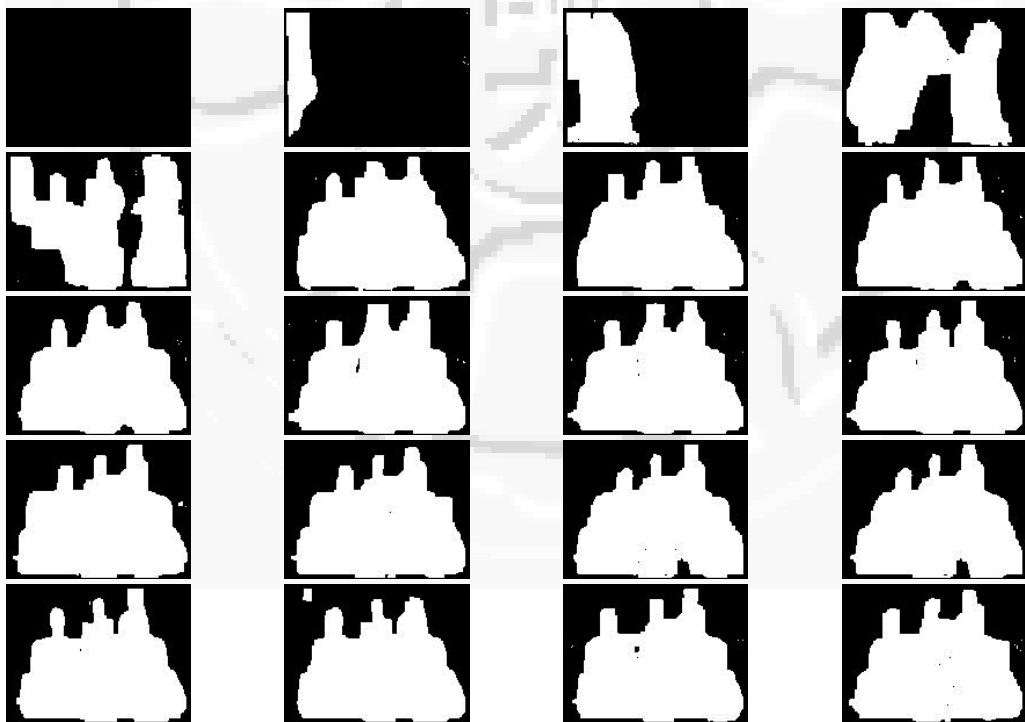


圖 6.7 影片 288 之前景擷取實驗結果

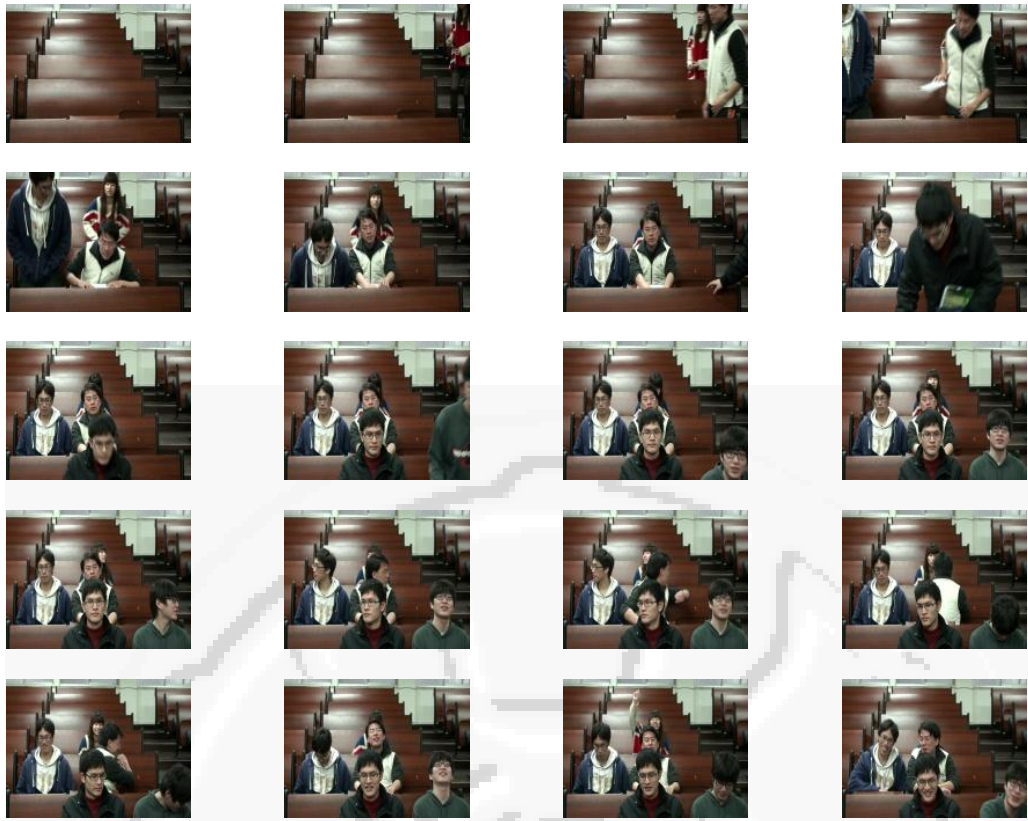


圖 6.8 影片 292 下之輸入影像

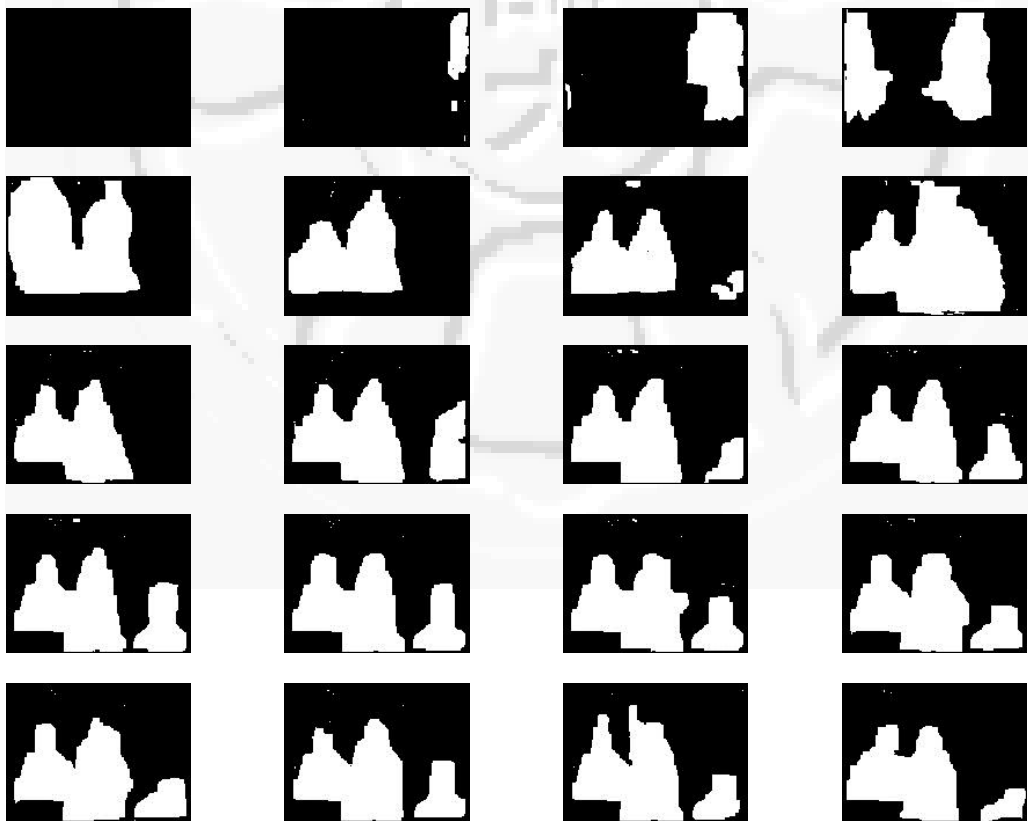


圖 6.9 影片 292 之前景擷取實驗結果

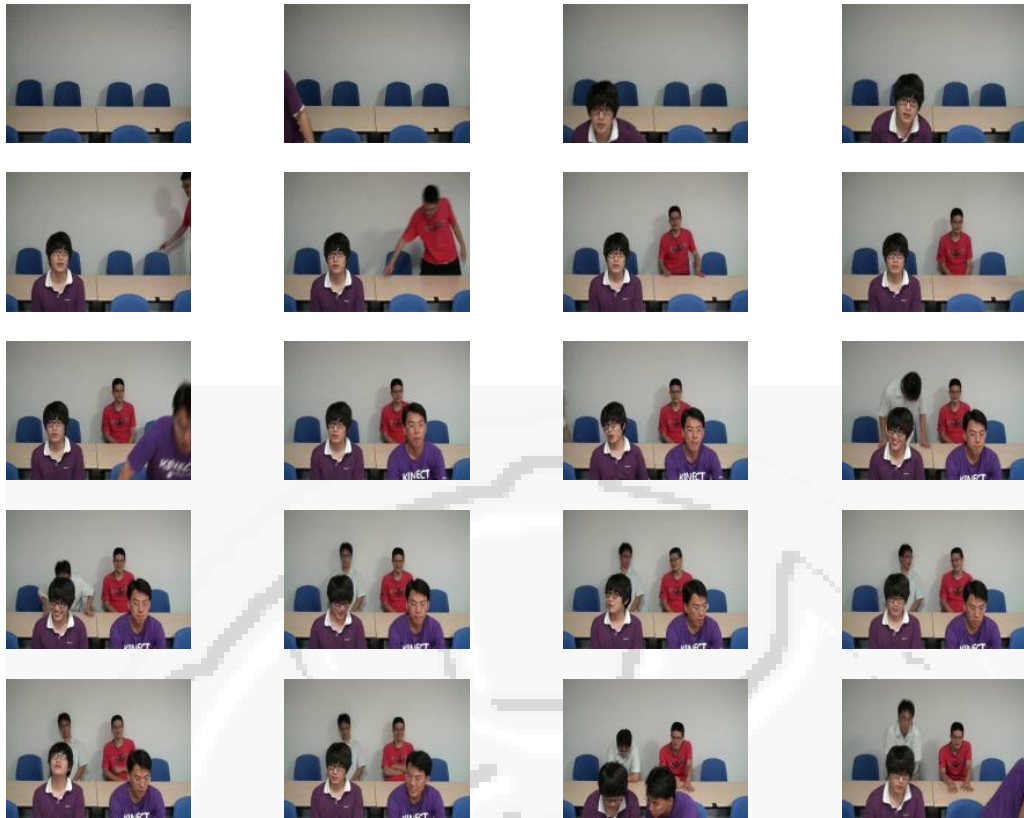


圖 6.10 影片 301 之輸入影像

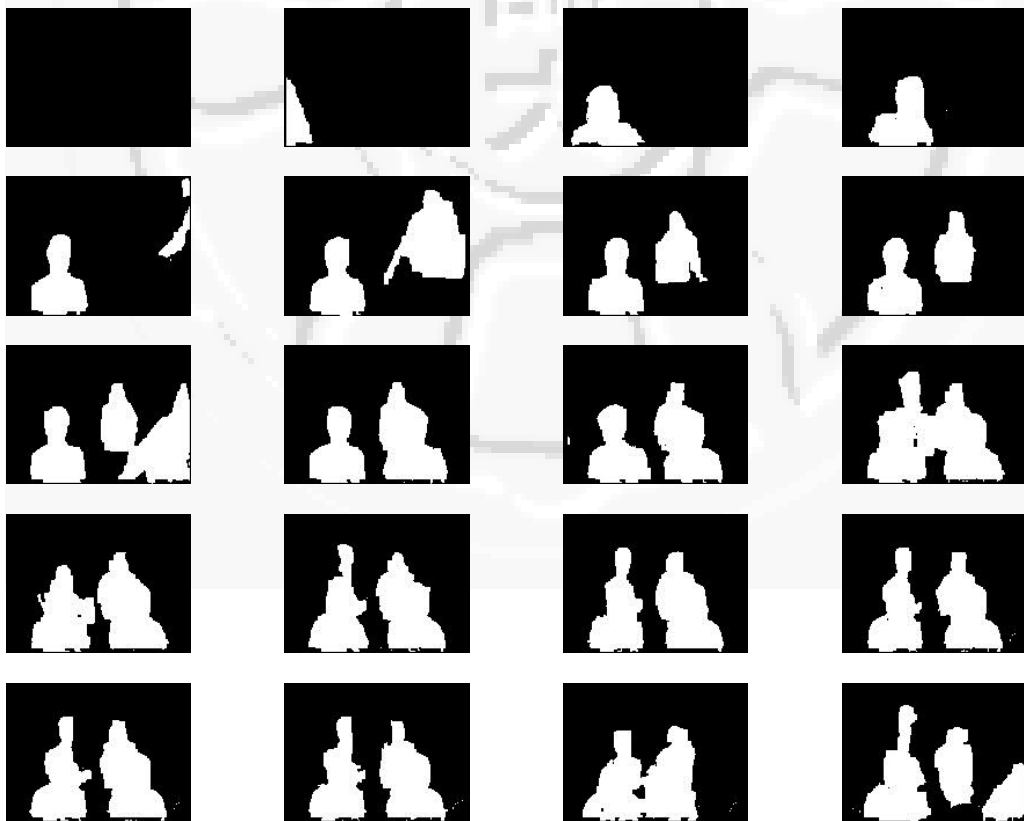


圖 6.11 影片 301 之前景擷取實驗結果

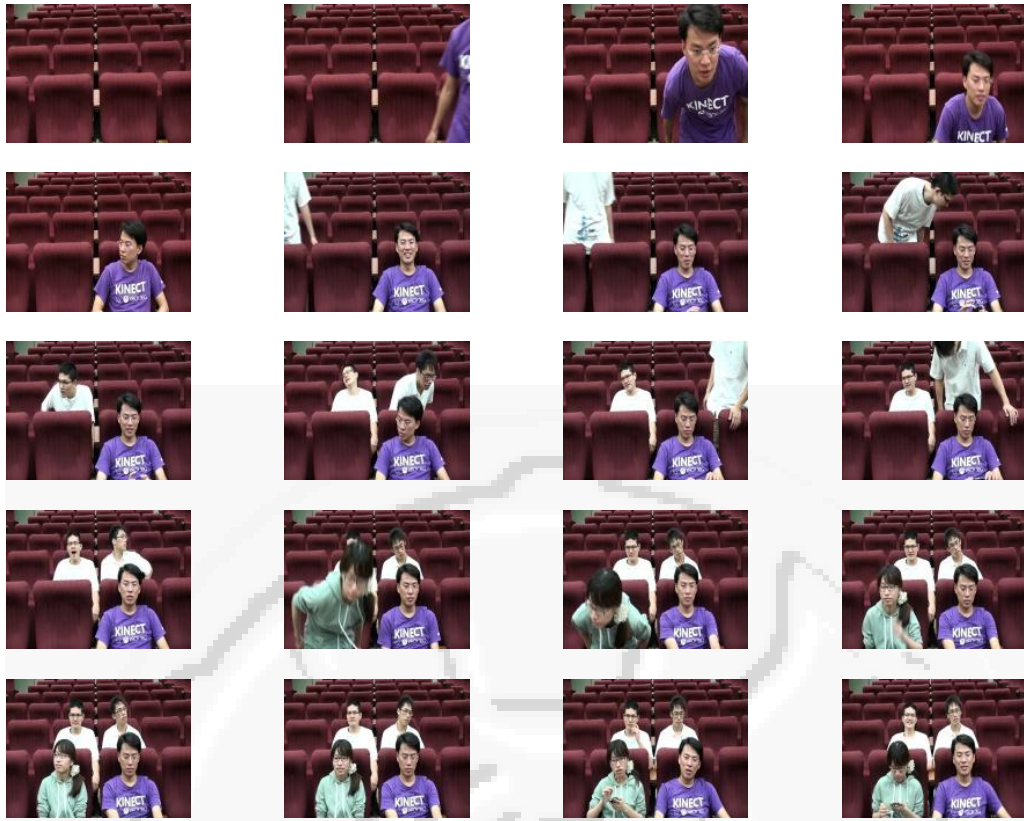


圖 6.12 影片 303 之輸入影像

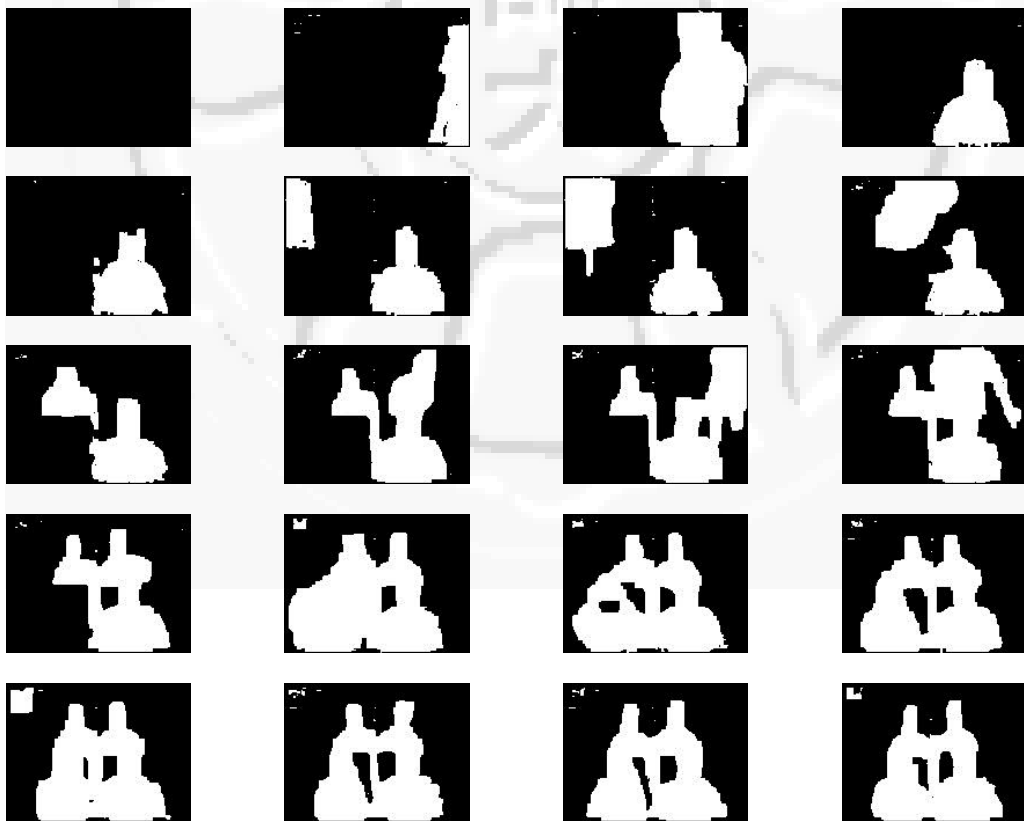


圖 6.13 影片 303 之前景擷取實驗結果

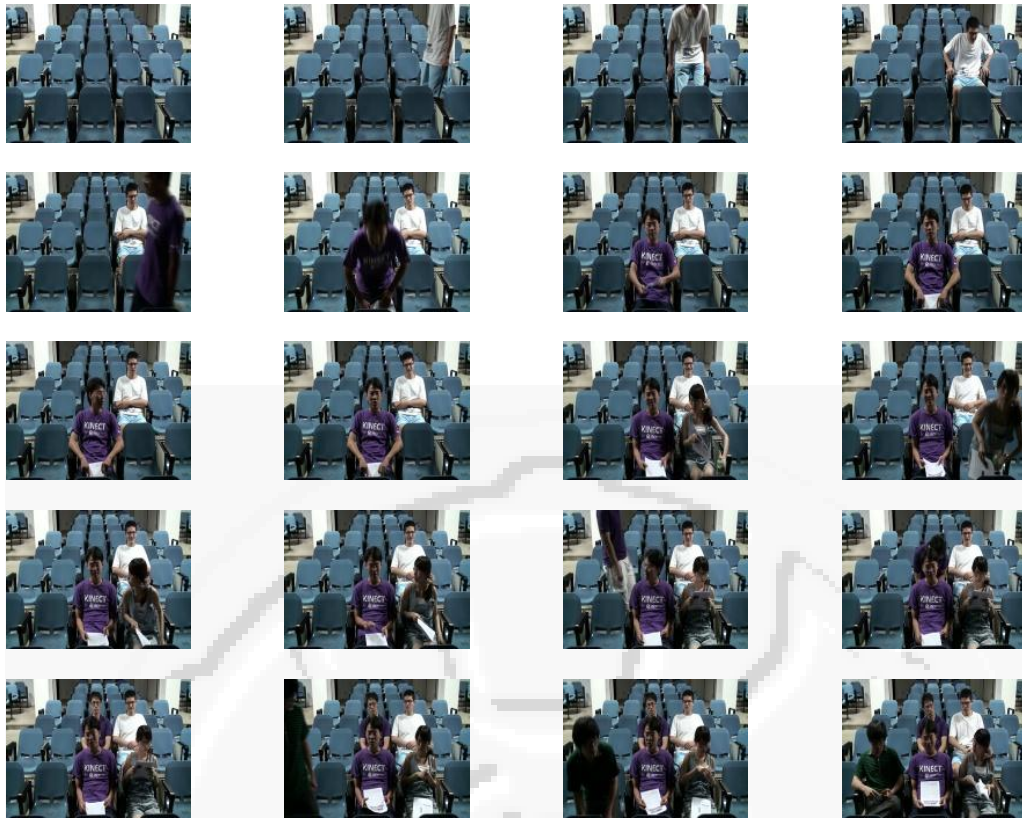


圖 6.14 影片 307 下之輸入影像

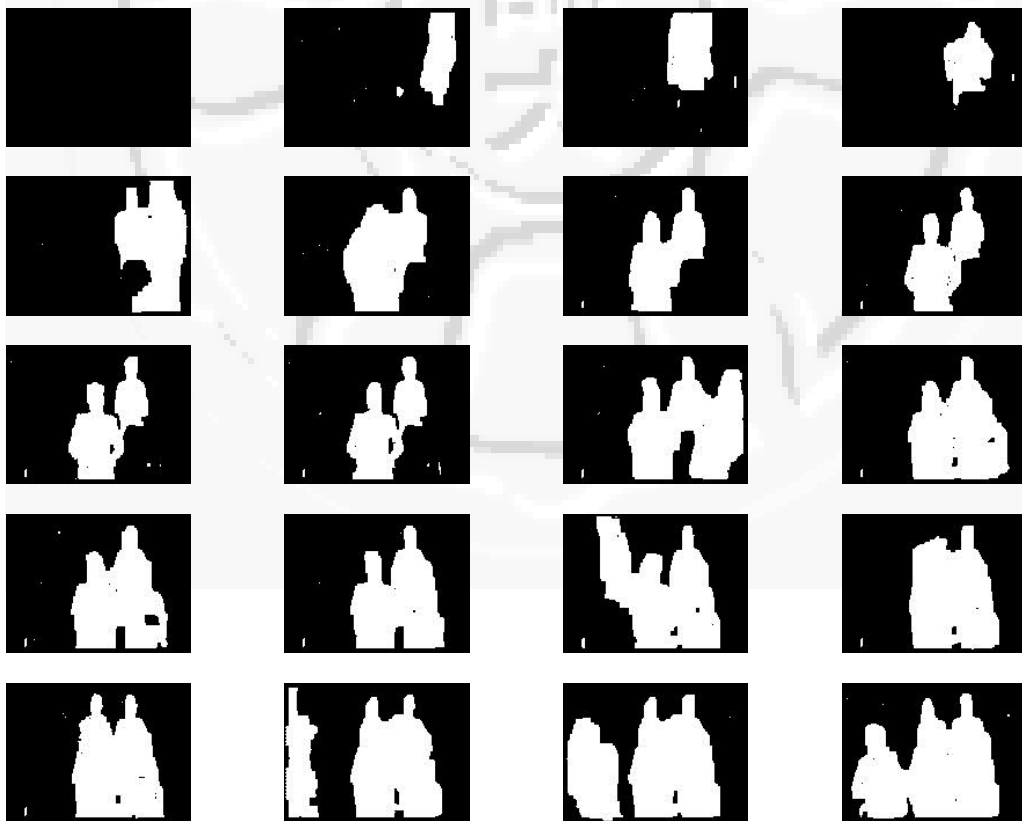


圖 6.15 影片 307 之前景擷取實驗結果

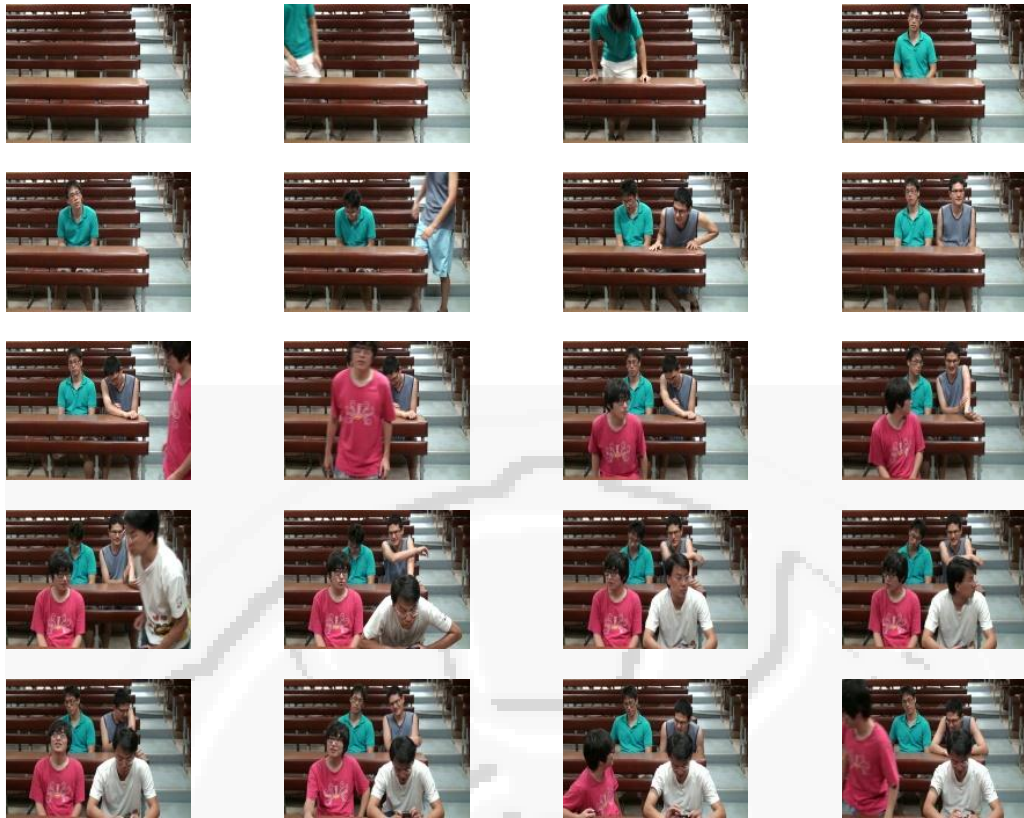


圖 6.16 影片 309 之輸入影像



圖 6.17 影片 309 之實驗結果

總結來說，前景擷取的結果正確率相當高，該方法對於各種不同的照明條件也有一定程度的抵抗能力。然而，由於此方法的實驗結果較偏於前景(precision 較 recall 低)，多偵測出來的部分若為膚色，就容易造成下一個階段的失誤。圖 6.18 為偵測錯誤的例子。若前景中有大片面積的顏色與背景過度相似的話，則有可能發生偵測錯誤的現象。



圖 6.18 去背景失敗之例子。

6.2 膚色區域偵測實驗結果

膚色區域之偵測，在膚色背景教室和非膚色背景教室中狀況相當不同。非膚色背景教室中，膚色顯示的顏色變異較大，若是在不同照明條件之下，顏色的分布就更為複雜。然而，經過膚色範圍的動態調整和 graph cut 演算法的增強，即使是在較為陰暗的非膚色背景教室，所偵測的膚色區域依然十分完整。

至於在膚色背景教室中，雖然膚色的分布較為集中，但由於背景幾乎都是膚色，經由 graph cut 的增強之後容易產生一些細碎的雜訊。這些雜訊和人臉區域若有重疊，會讓人臉區域的形狀變的不規則，而降低之後人臉偵測的正確率。這個缺點也是本研究未來主要努力的方向之一。

正確率的計算與前景擷取類似，以像素為基礎，並且以自製的 ground truth 做為比較標準，數據如表 6.2 所示。

表 6.2 膚色區域偵測之實驗結果

影片	教室	照明情況	影像張數	膚色背景	Precision	Recall	F-measure
 265	S101	陰暗	1304	否	88.47%	77.91%	0.8256
 266	B101	略暗	4488	是	67.33%	88.49%	0.7635
 288	B1 演講廳	正常	866	否	62.34%	66.43%	0.6393
 292	B102	正常	1200	是	78.03%	81.73%	0.7948
 301	C209	正常	5641	否	83.41%	87.31%	0.8526
 303	S101	正常	5226	否	86.93%	71.91%	0.7859
 307	B1 演講廳	略暗	6227	否	91.57%	66.99%	0.7722

由數據來看，整體的準確率比前景擷取低許多。除了膚色範圍的定義的確比前景背景的區分困難之外，主要的原因其實是在於，影像中的膚色所占面積不多，pixel based 的計算方法在這種情況下會放大小區域錯誤所佔的比例。另外，如前段所述，在膚色背景教室中，graph cut 演算法所造成的細碎雜訊較多，因此也顯著地拉低了正確率。以下為各影片的膚色偵測圖例以及說明：

- 265: 此影片的陰暗程度較高，所偵測到的膚色像素較少因此 recall 較低，不過由於膚色與非膚色的界線明顯，因此偵測錯誤的機率也較低，precision 反而較高。
- 266: 膚色背景的教室環境由於光反射的原理，會有類似膚色增強的效果，因此本影片的偵測結果中，recall 是相當高的，也就是說很少膚色 pixel 未偵測到。不過也因為這樣的效果，使得此影片之 precision 明顯低於非膚色背景教室。
- 288: 由於此間教室的背景為藍色與灰黑色，因此膚色顯現出偏綠的效果，對於本研究所提出的偵測方法來說是屬於較為不利的狀況，因此 precision 和 recall 都偏低。不過，對於實驗數據影響最大的應是左下角女同學的髮色被歸類為膚色所致。
- 292: 此為膚色背景教室，不過由於日光燈全開，白平衡並未受到膚色影背景影響太多，正確率在膚色背景教室相對高一些。
- 301: 非膚色背景教室，不過原木色的桌面亦為膚色，對於膚色偵測的結果也有些許影響。
- 303: 此影片之實驗結果顯現出本方法的優點，儘管膚色偏白，經由動態調整膚色範圍的過程，解決此一問題，並維持了較高的正確率。
- 307: 由於膚色偏綠的緣故，此間教室所偵測的膚色區域會較 ground truth 略為收縮，這個現象為 recall 偏低的主因。不過，缺少了染色頭髮的干擾，儘管偏暗，所得到的正確率仍然高於 288 甚多。

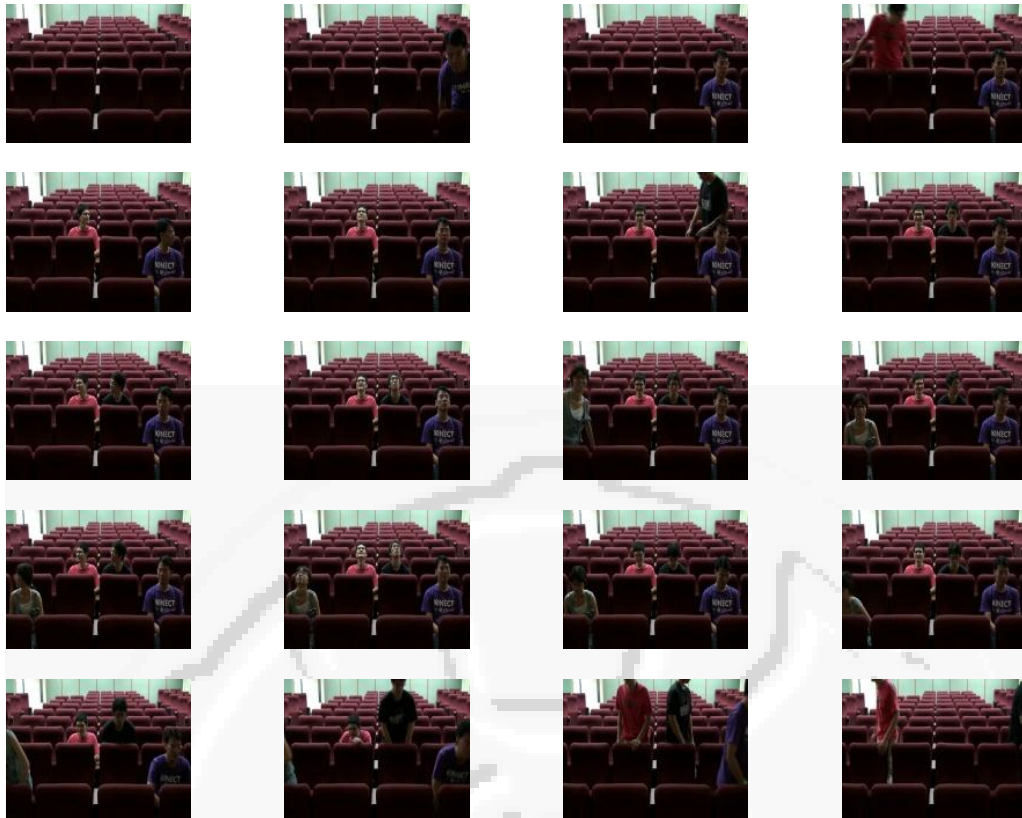


圖 6.19 影片 265 之輸入影像

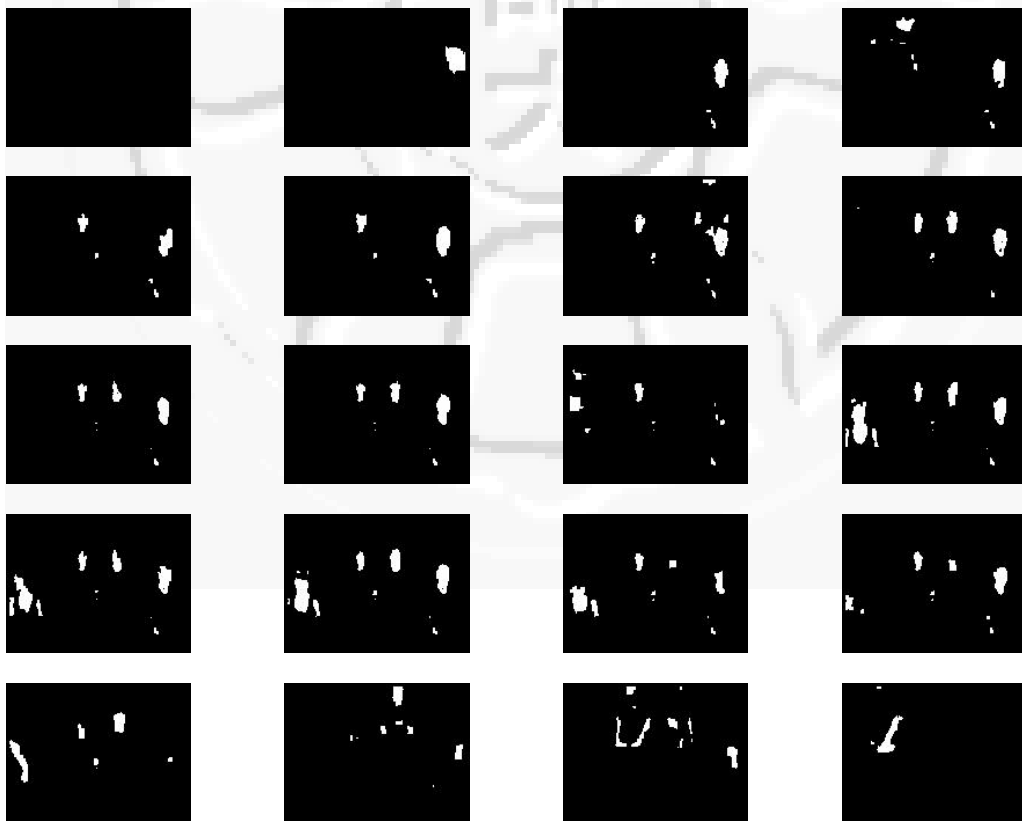


圖 6.20 影片 265 之膚色區域偵測結果

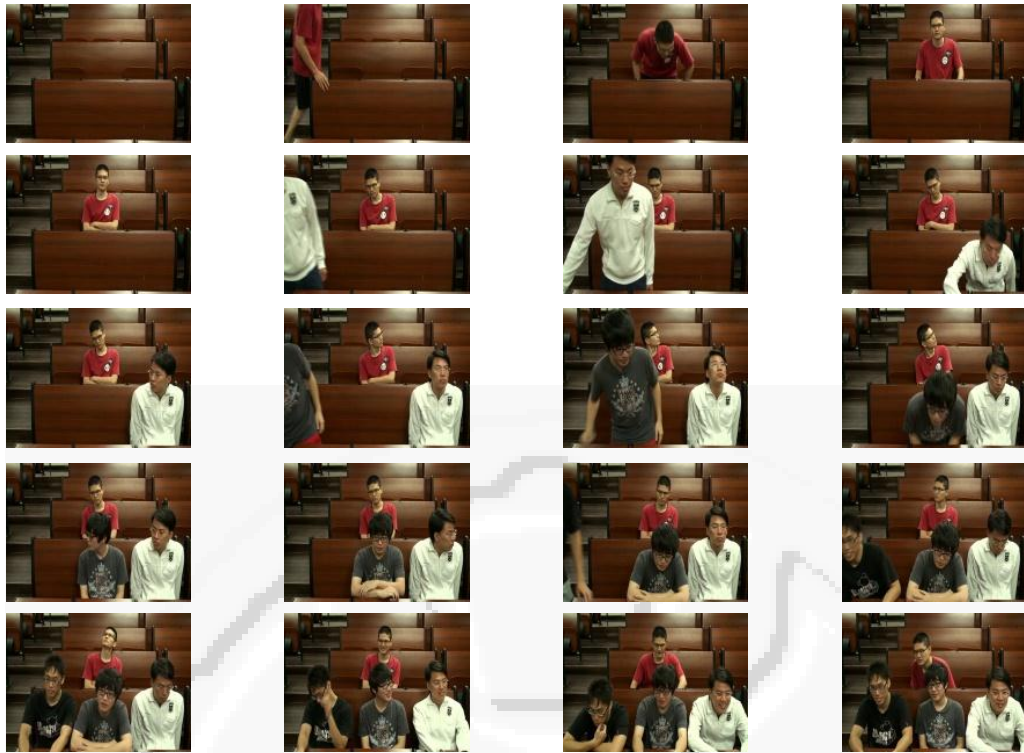


圖 6.21 影片 266 之輸入影像



圖 6.22 影片 266 之膚色區域偵測實驗結果

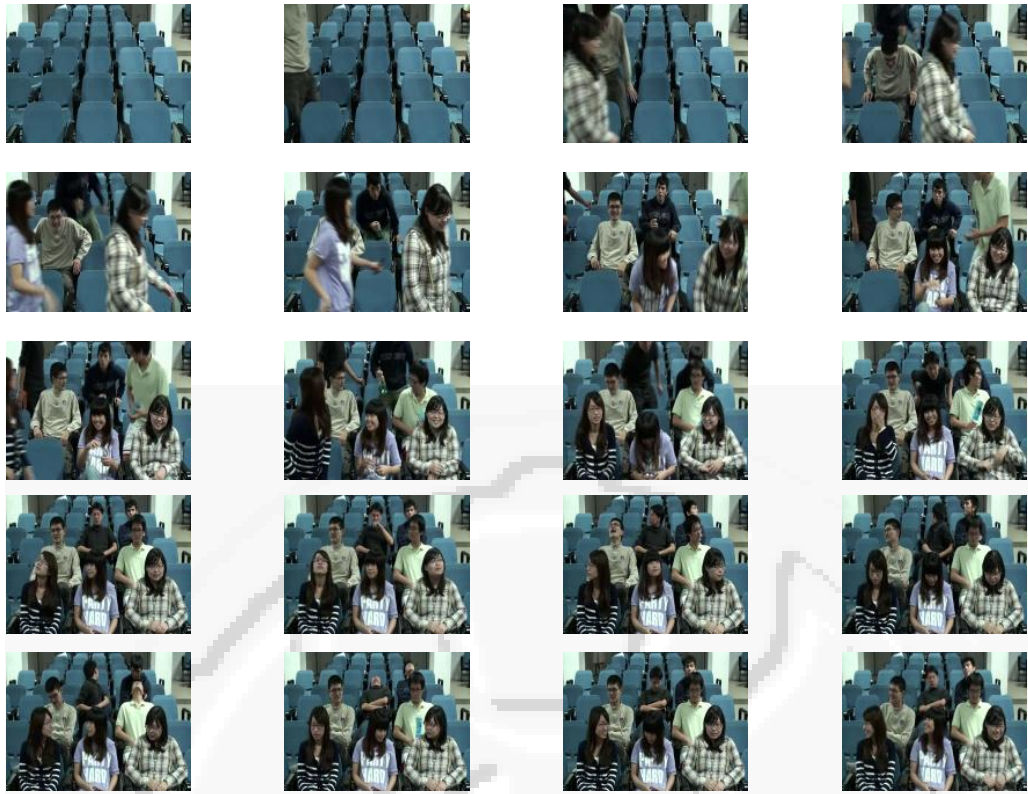


圖 6.23 影片 288 之輸入影像



圖 6.24 影片 288 之膚色區域偵測結果

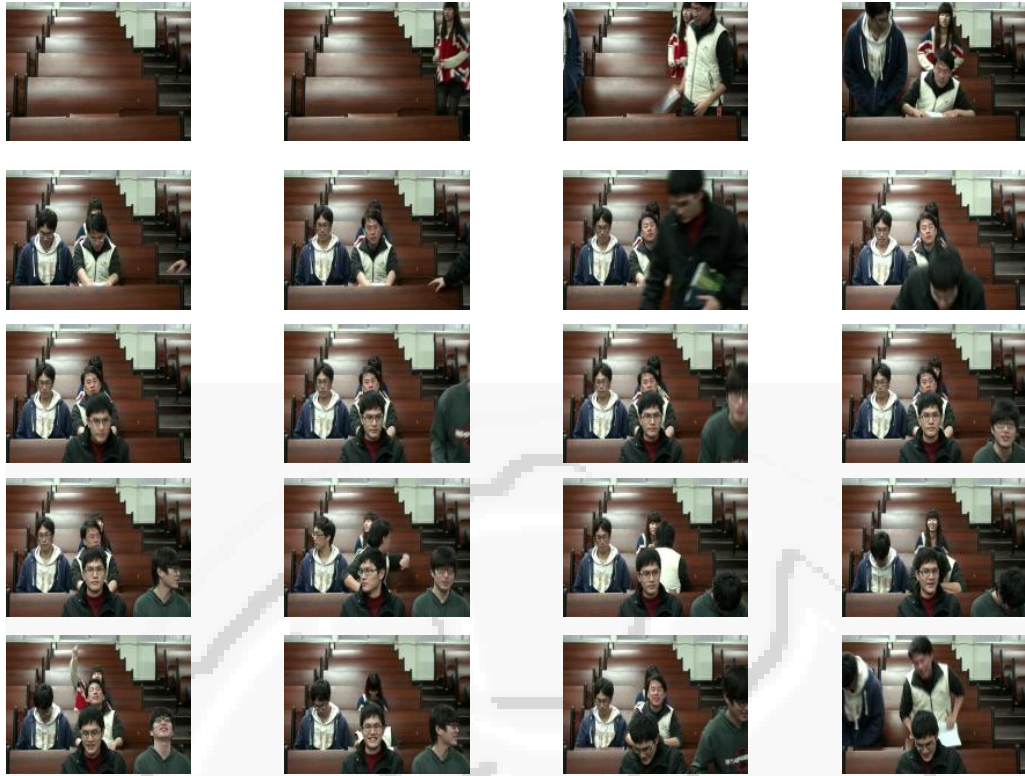


圖 6.25 影片 292 之輸入影像



圖 6.26 影片 292 之膚色區域偵測結果

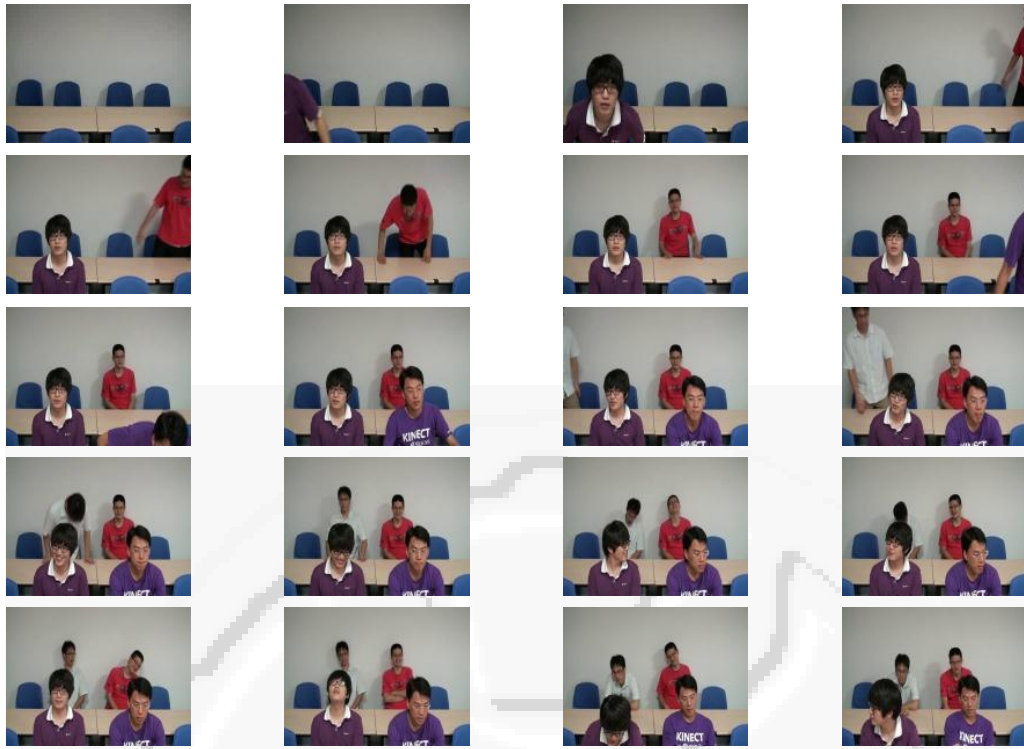


圖 6.27 影片 301 之輸入影像

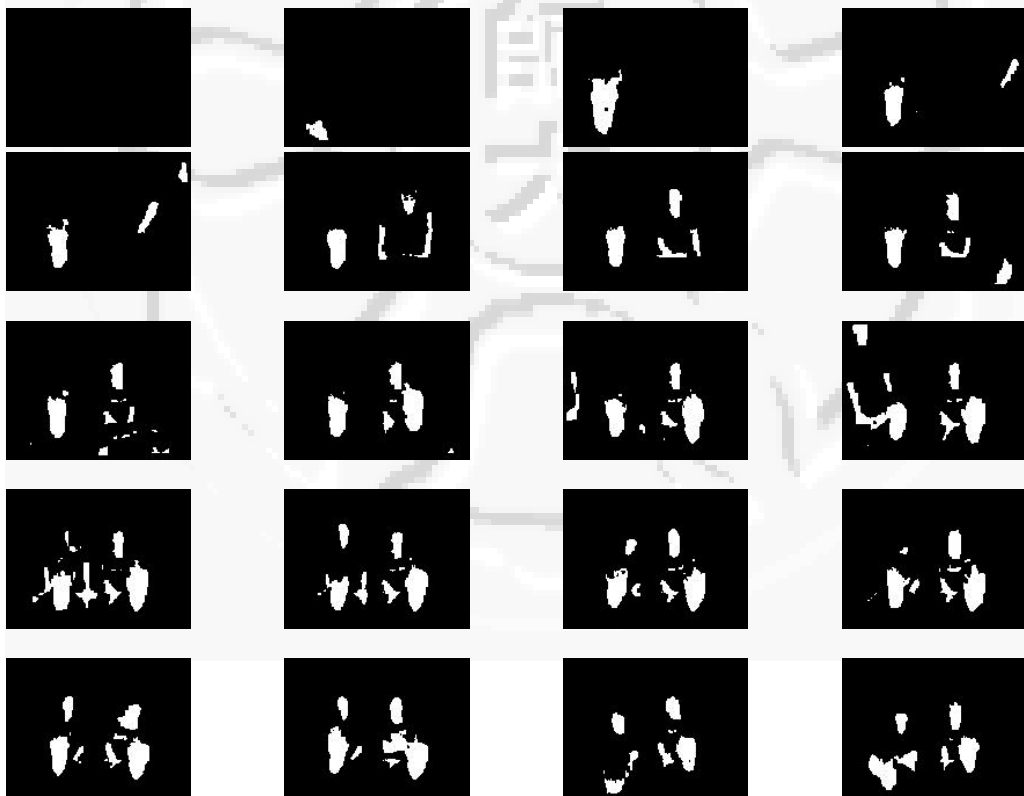


圖 6.28 影片 301 之膚色區域偵測結果

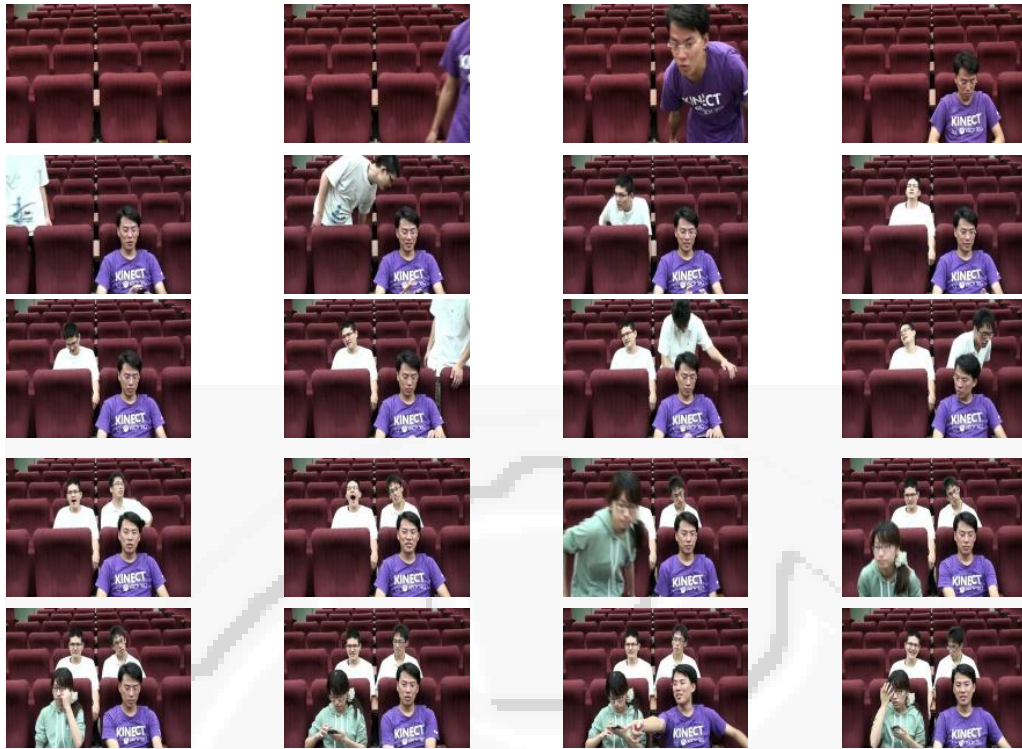


圖 6.29 影片 303 之輸入影像

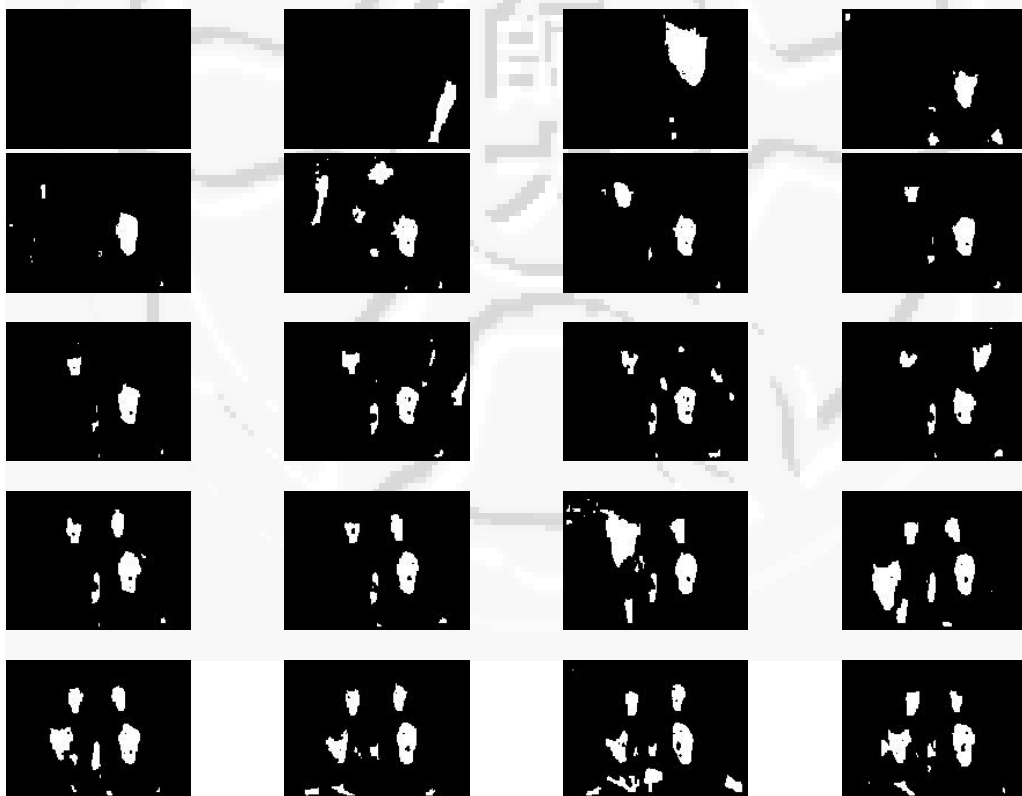


圖 6.30 影片 303 之膚色區域偵測結果

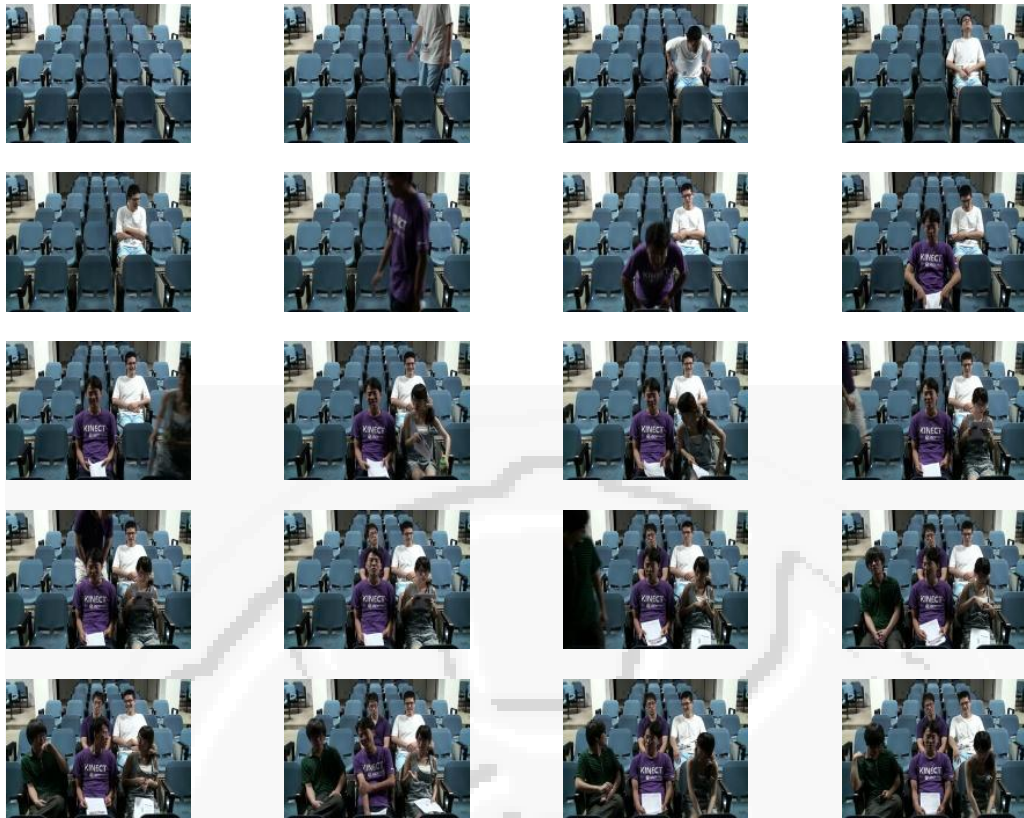


圖 6.31 影片 307 之輸入影像

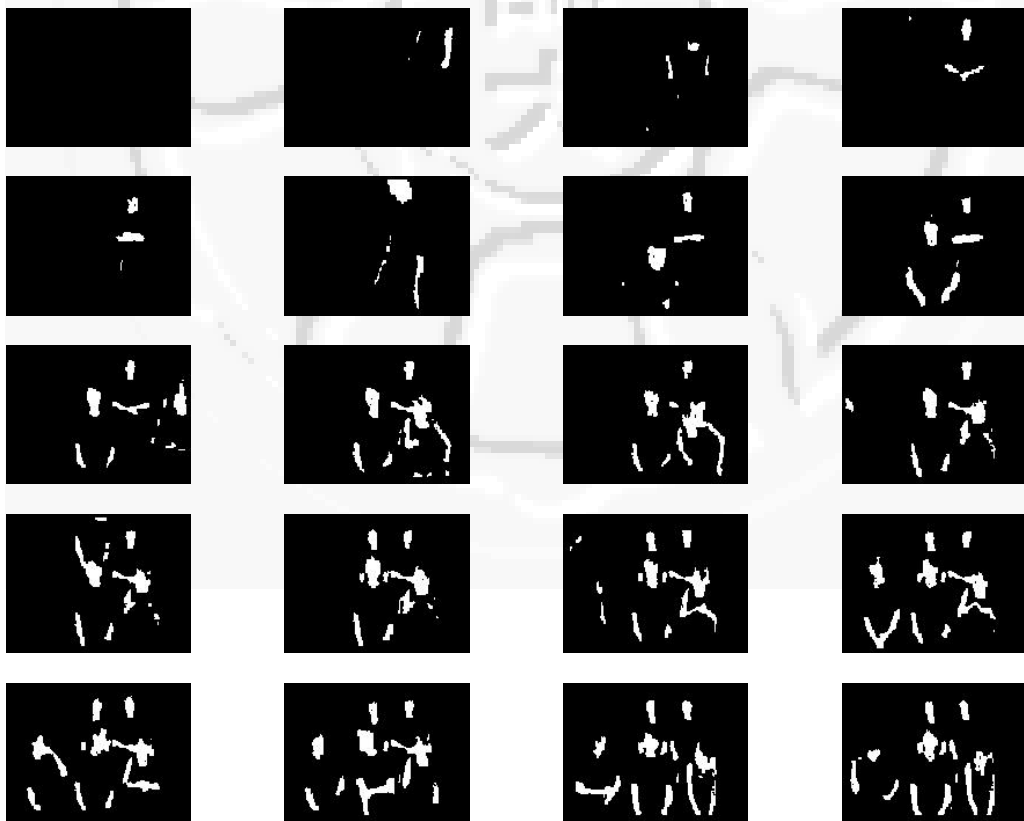


圖 6.32 影片 307 之膚色區域偵測結果

6.3 人臉區域選取之結果

人臉區域的選取，本系統主要以形狀、大小和表面邊緣的強度以及方向變異來過濾篩選掉非膚色區域，實驗數據如表 6.3 所示。

表 6.3 人臉偵測之實驗結果

影片	教室	照明情況	影像張數	膚色背景	Precision	Recall	F-measure
 288	B1 演講廳	正常	866	否	98.87%	59.84%	0.7455
 292	B102	正常	1200	是	95.74%	72.61%	0.8258
 301	C209	正常	5641	否	96.99%	74.59%	0.8432
 303	S101	正常	5226	否	99.06%	84.87%	0.9141
 309	B103	正常	5209	是	99.27%	57.56%	0.7286

由於在人臉偵測階段，也就是人臉區域選取階段，系統設定了許多固定門檻值，因此在 precision 和 recall 之間便會有 trade off 的現象產生。觀察表 6.3 便可知，各影片人臉偵測的 precision 都非常高，但 recall 則普遍偏低。欲解決此一現象，或是將各影片的拍攝條件限制多一些，或是發展出能夠自動學習門檻值的方法，這也是未來工作一個主要的努力方向。

另外，以本研究所發展的方法而言，尚未解決的問題便是膚色區塊重疊的問題。膚色區塊重疊會使多個膚色區塊以不規則的形式組合在一起，大大增加人臉

區域選擇的困難度，也是拉低正確率的主因。以下為各影片偵測結果之圖例及說明：

- 288: 非膚色教室，此影片的偵測困難度主要是在於眾多的人數，並且人臉大小的變化很大。造成了許多後方的人臉因為遮擋或大小太小的原因而偵測不到。
- 292: 膚色背景，主要困難度為學生排列不規則，遮擋的情形相當多，正中間學生的臉部大部分都偵測的到。不過若是與後方同學的手部重疊，便容易因為形狀的原因被濾掉。
- 301: 非膚色背景，人數也較少，因此正確率較上兩個影片來說相對較高。不過還是會受膚色區域重疊和膚色桌面影響正確率。
- 303: 此五個測試影片中正確率最高者，由於椅背較高，較少膚色重疊的現象，並且臉部區域多半夠大且清楚，因此能有較高的正確率。
- 309: 膚色背景嚴重地影響了偵測正確率，並且在此影片中的膚色區域重疊情況也較多，因此 recall 非常低。

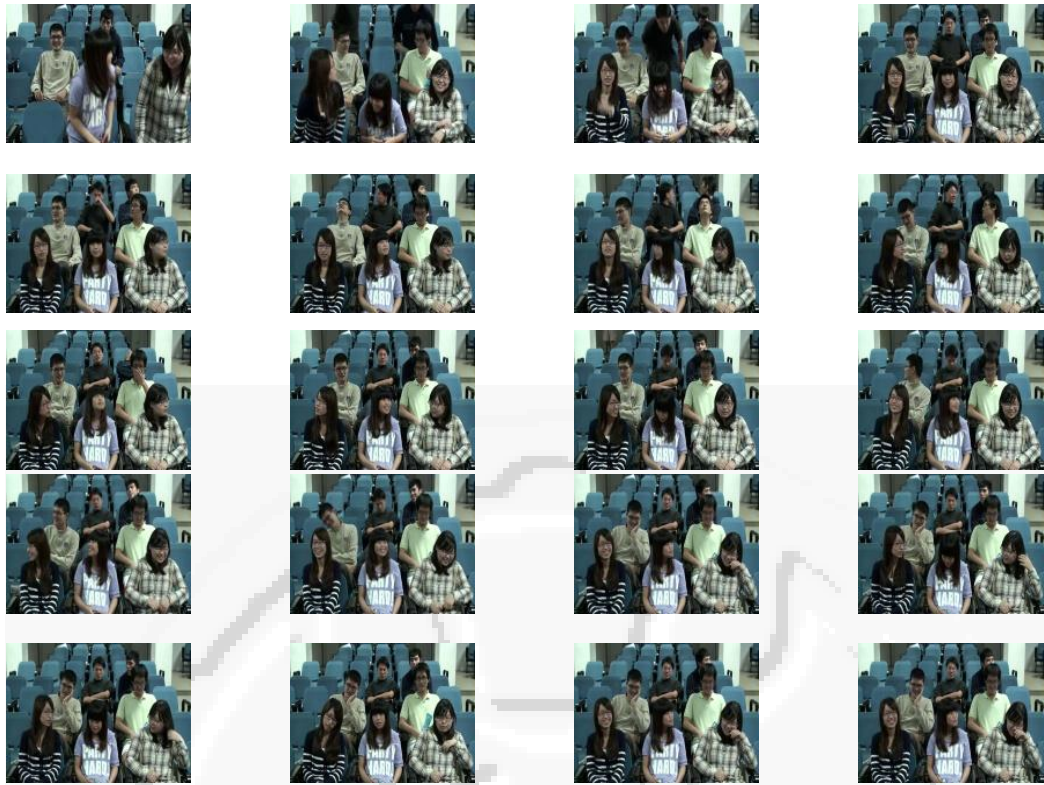


圖 6.33 影片 288 之輸入影像

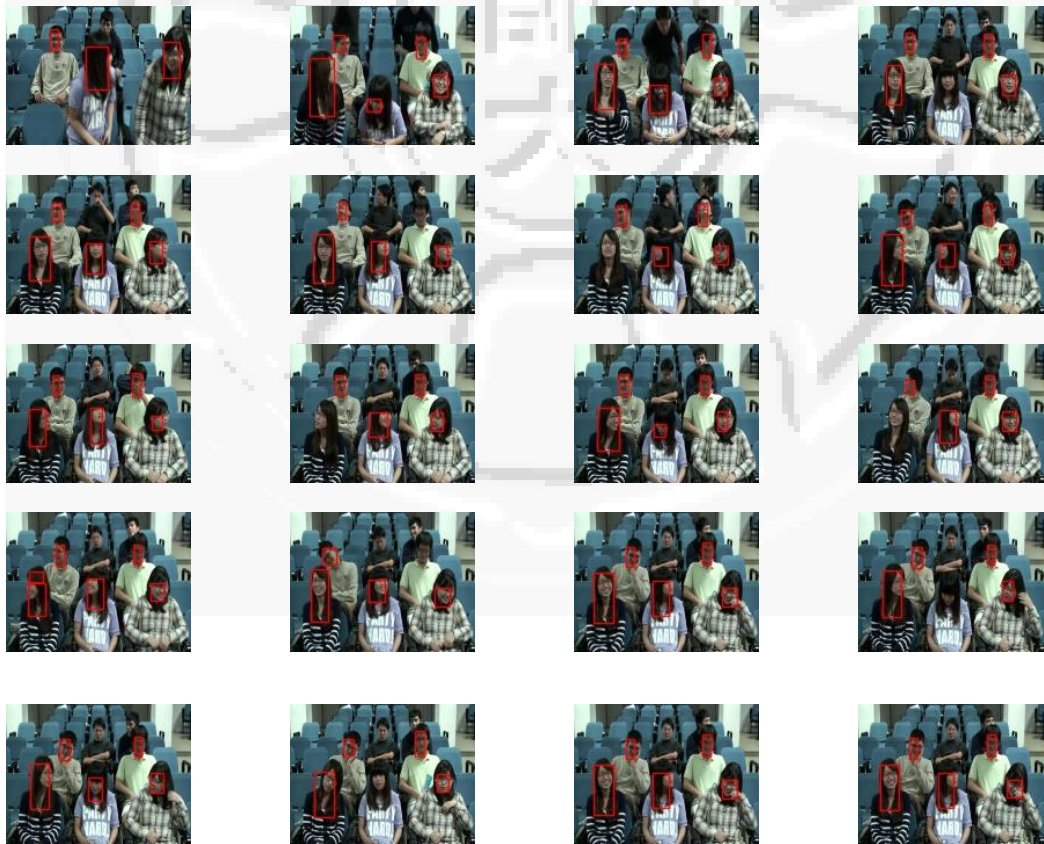


圖 6.34 影片 288 之人臉偵測結果

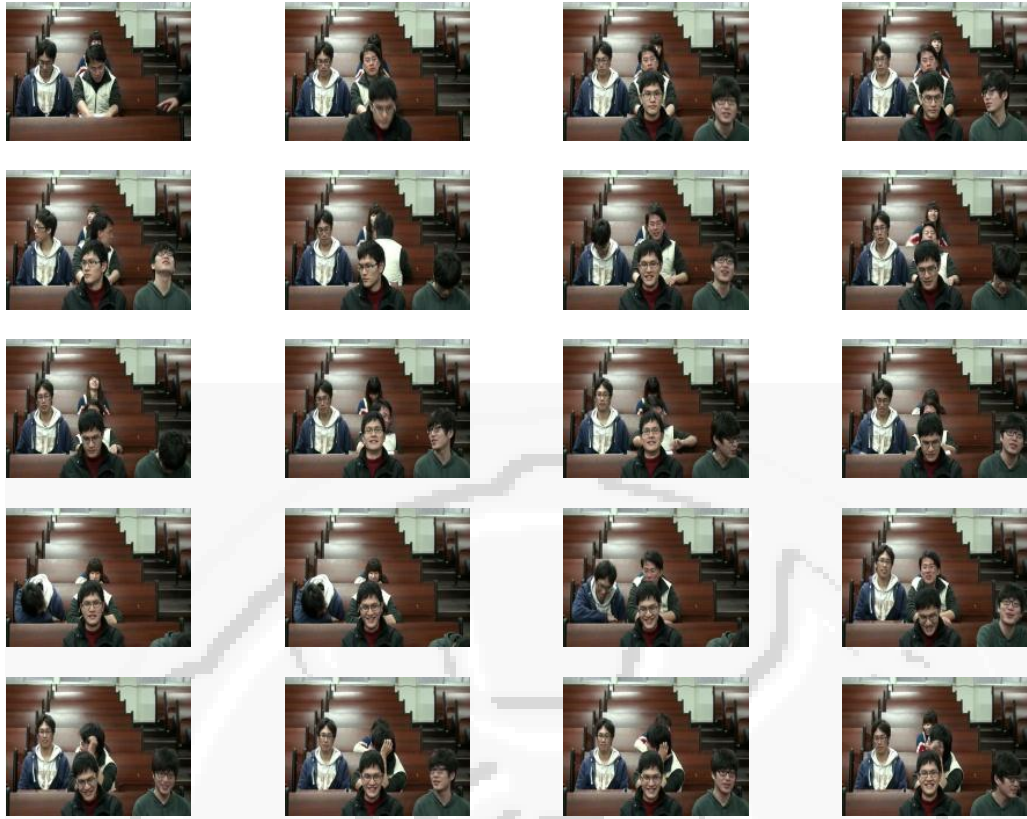


圖 6.35 影片 292 之輸入影像

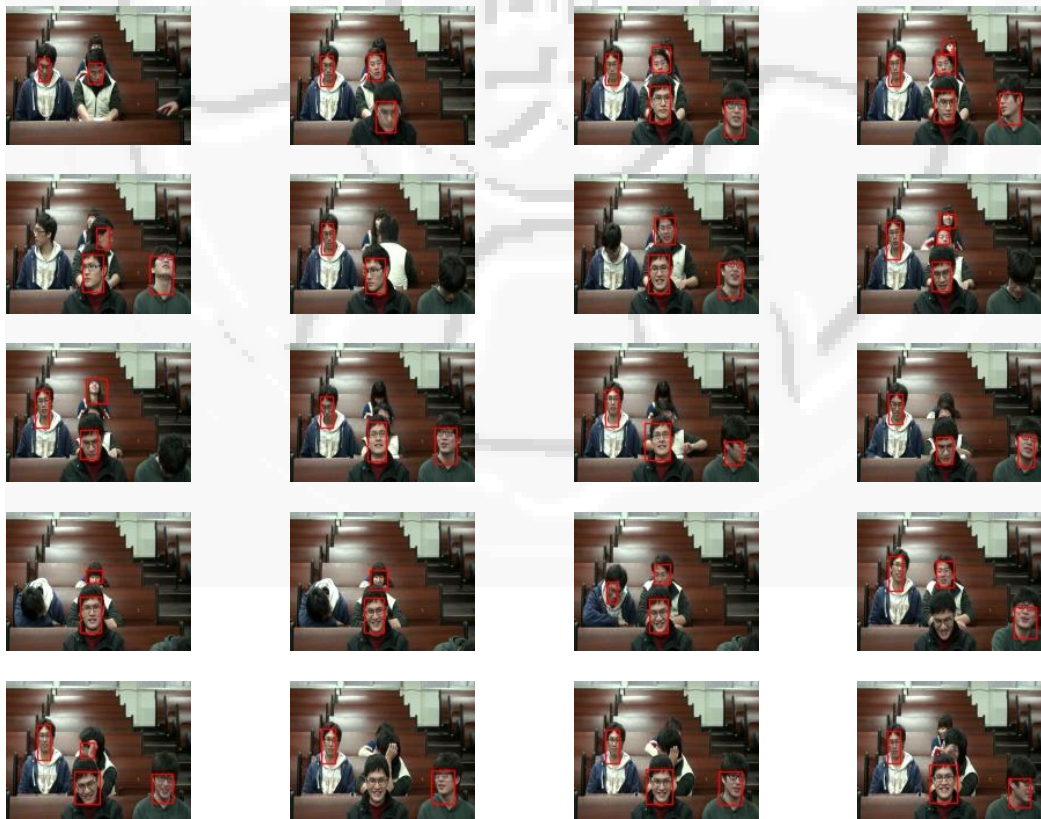


圖 6.36 影片 292 之人臉偵測結果

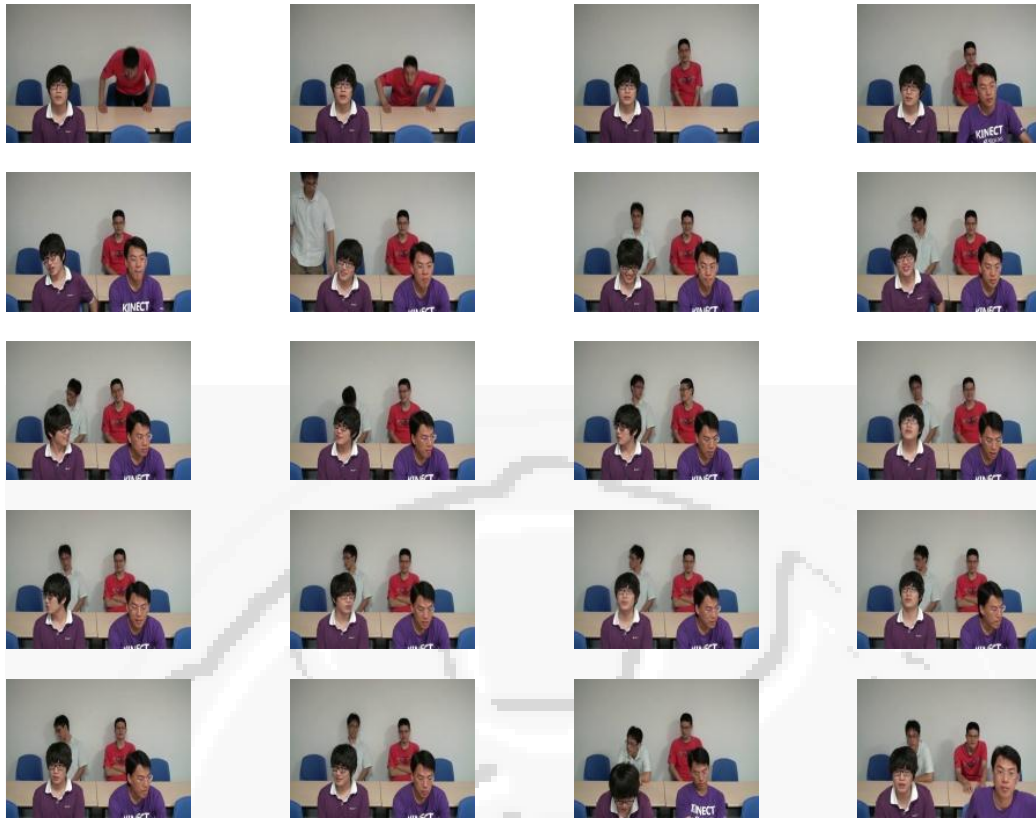


圖 6.37 影片 301 之輸入影像

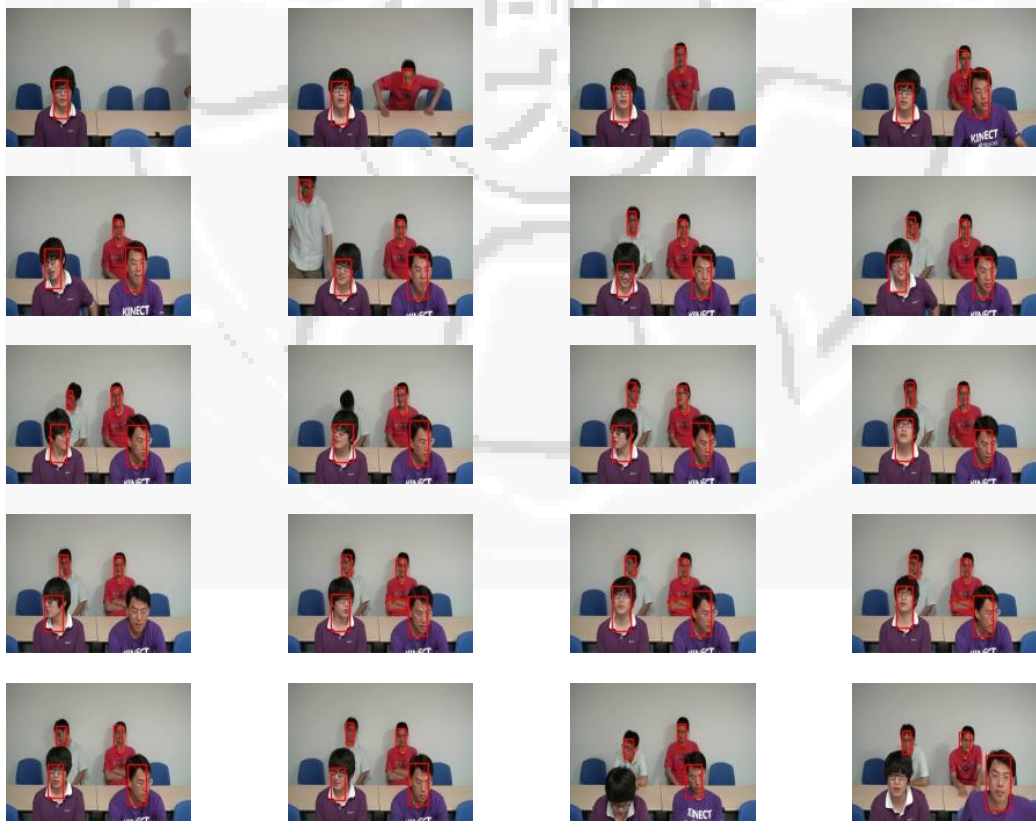


圖 6.38 影片 301 之人臉偵測結果

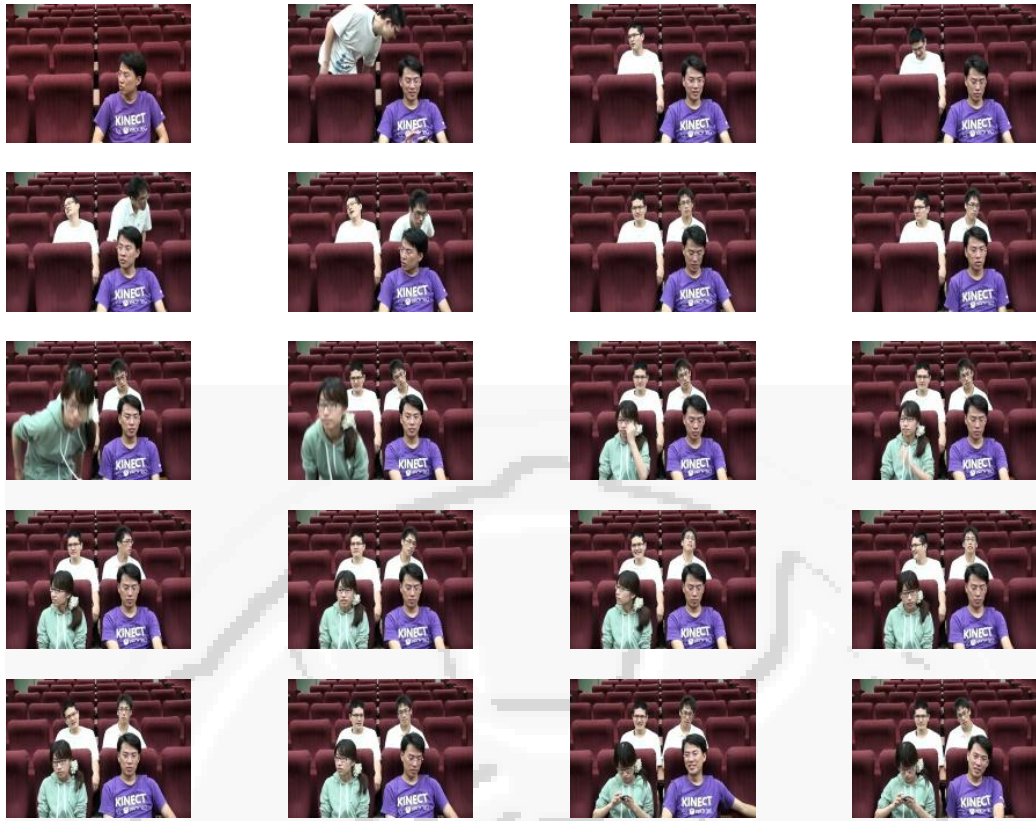


圖 6.39 影片 303 之輸入影像

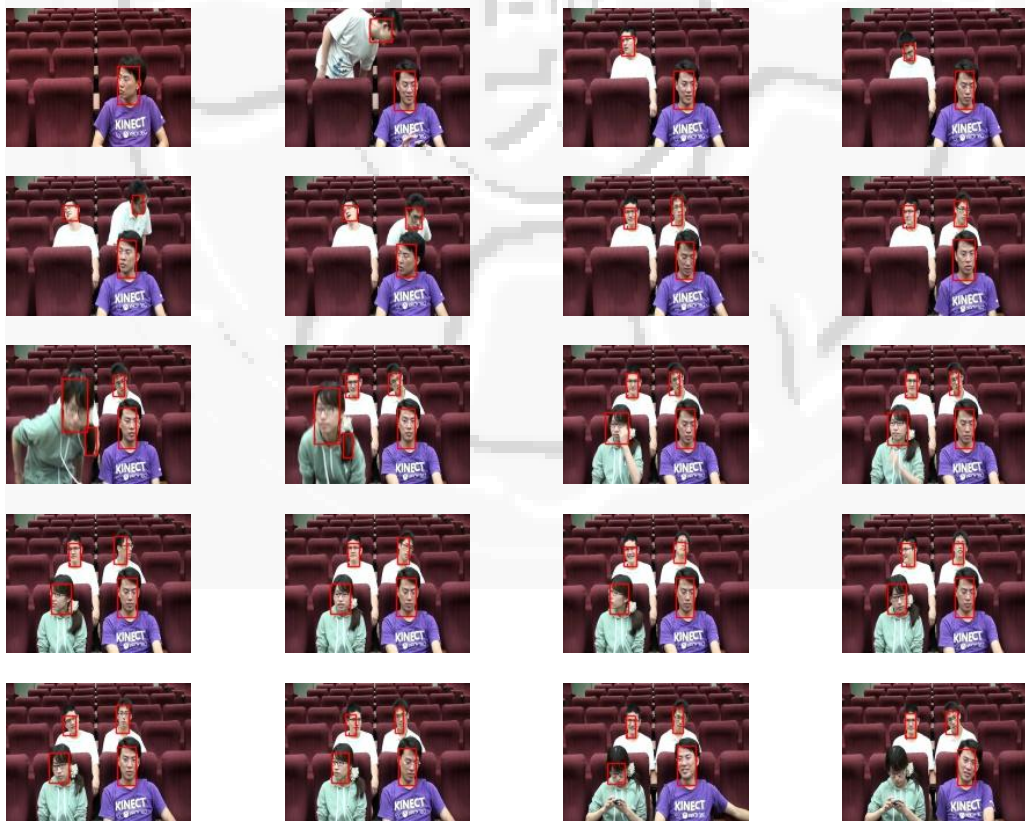


圖 6.40 影片 303 之人臉偵測結果

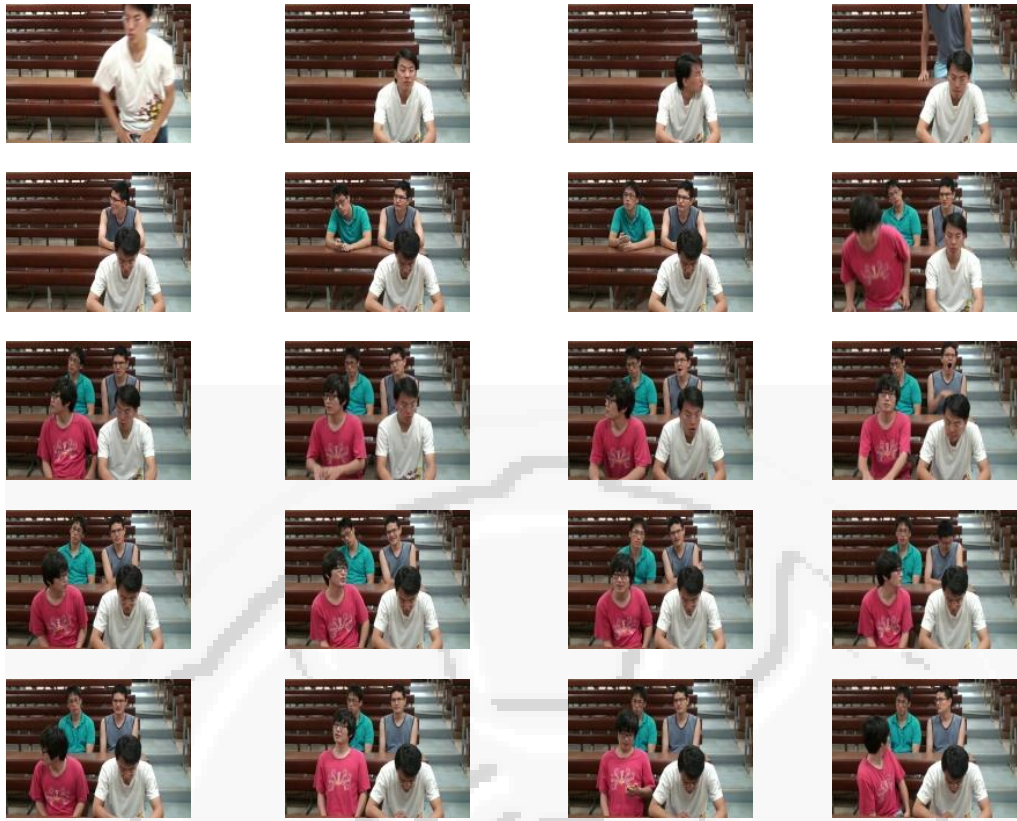


圖 6.41 影片 309 之輸入影像



圖 6.42 影片 309 之人臉偵測結果

第七章 結論與未來工作

7.1 結論

本研究以 graph cut 演算法為核心，試圖在連續影像中以顏色為特徵偵測人臉，並且希望能夠在學生的各種頭部姿勢之下偵測，以期未來能夠分析學生的行為。

以顏色為特徵偵測人臉，最大的缺陷在於此特徵極易受到環境影響。而且，膚色的顏色也並非只出現在人類身上，尤其在教室的背景中，木製的膚色物體相當多，如何避免掉這些雜訊，成為以顏色偵測人臉的最大問題。

而上述提到的兩個問題，本研究皆以 graph cut 演算法為基礎有效率地將其解決。首先，系統所使用的去背景方法，由於使用了 graph cut 演算法，使得偵測到的前景區塊正確而完整，有效的去除了影像中不屬於人類的膚色部分。另外，顏色特徵方面我們使用了較不受光影變化影響的 hue 和 saturation 兩個 channel。雖然膚色 intensity 值的分布容易受到光線環境影響，但其在 hue channel 之中仍然有聚集的現象。

利用這個特質，初步偵測膚色之後，先以 graph cut 解決膚色破碎的問題，之後再次統計膚色區域中的 hue intensity 值並更新區間。最後，在多個膚色區域中，系統簡單以形狀和表面紋理的複雜程度決定人臉的膚色區域。

根據實驗結果，本系統在非膚色背景的教室中，且膚色區域沒有交錯的情況下有著相當高的正確率。但在膚色背景教室中，膚色偵測的階段則會因為些許雜訊的摻入而使人臉區域的形狀不夠單純，讓系統誤判為非人臉區域。膚色區域有重疊時亦容易產生上述情形，亦為本研究未來主要的努力方向。

7.2 未來工作

1. 以更高階的特徵判斷人臉區域：本系統現階段所採取的兩個特徵，邊緣反應和 gradient 方向之變異數表現尚不夠穩定，在不同教室和影片中所產生的數值不盡相同。
2. 以橢圓逼近人臉：目前只做人臉膚色區域的選取，為本系統無法抵抗膚色區域重疊的主要原因。某些膚色區域不只包含人臉，可能也同時包含了後方學生的手部等膚色區域，如果能夠以橢圓逼近人臉，將人臉由重疊的膚色區塊中分離開來，便能夠有效率的提升偵測的正確率。
3. 膚色增強以及照明情況分析：對於某些特別陰暗的教室環境，為了能夠讓膚色完整，可以事先做膚色增強的動作。此外，系統也能夠對於背景影像事先做一些照明情況的分析，以對演算法進行微調的動作。
4. 人臉追蹤：以適當的特徵進行人臉追蹤，以增強人臉偵測的穩定性。

本研究在去背景與膚色偵測兩個部分提供了穩定且優良的基礎。未來期望能夠以此基礎，改良人臉偵測的分法，以期進一步偵測人臉的頭部姿勢與凝視角度，以夠分析學生們的行為，達到自動進行教室觀察的目的。

參考文獻

L. Cheng, M. Gong, D. Schuurmans, and T. Caelli, "Real-Time Discriminative Background Subtraction," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1401-1414, 2010.

Z. Wu and R. Leahy, "An Optimal Graph Theoretic Approach to Data Clustering: Theory and its Application to Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1101-1113, 1993.

O. Juan and Y. Boykov, "Interactive Graph Cuts," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, pp. 1023-1029, 2006.

P. Kohli, "Dynamic Graph Cuts for Efficient Inference in Markov Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2079-2088, 2007.

Y. Boykov, "An Experimental Comparison of Min-cut/Max-flow Algorithms for Energy Minimization in Vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124-1137, 2004.

P. Kae and R. Bow, "An Improved Adaptive Background Mixture Model for Real-Time Tracking with Shadow Detection," *Proceedings of the 2nd European Workshop on Advanced Video-Based Surveillance Systems*, London, 2001.

O. Carmichael and M. Hebert, "Shape-Based Recognition of Wiry Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.1537-1552, vol.26, no.12, Dec. 2004.

C. Stauffer and W. E. L. Grimson, "Learning Patterns of Activity using Real-Time Tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747-757, 2000.

A. M. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and Foreground Modeling using Nonparametric Kernel Density Estimation for Visual Surveillance," *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151-1163, 2002.

A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-Parametric Model for Background Subtraction," *Proceedings of the 6th European Conference on Computer Vision-Part II (ECCV '00)*, pp. 751-767, London, 2000.

K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-Time Foreground-Background Segmentation using Codebook Model," *RealTime Imaging*, vol. 11, no. 3, pp. 172-185, 2005.

R. Lienhart, L. Liang, and A. Kuranov, "A Detector Tree of Boosted Classifier for Real-Time Object Detection and Tracking," *Proceedings of International Conference on Multimedia and Expo (ICME '03)*, vol.2, pp. II-277-280, Baltimore, Maryland, 2003.

L. M. Huang, C. S. Ouyang, and S. J. Lee, "A Neuro-Fuzzy Approach for Multiple Human Objects Segmentation," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 2815-2820, Washington, 2003.

Y. Yacoob and L. S. Davis, "Detection and Analysis of Hair," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1164-1169, 2006.

K. C. Lee, D. Anguelov, B. Sumengen, and S. B. Gokturk, "Unsupervised Learning from Local Features for Video-Based Face Recognition," *Proceedings of the 8th IEEE International Conference on Automatic Face & Gesture Recognition (FG '08)*, pp. 1-6, Amsterdam, The Netherlands, 2008.

P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 2, no. 57, pp. 137-154, 2004.

A. H. Gee and R. Cipolla, "Determining the Gaze of Faces in Images," *Image and Vision Computing*, vol. 12, pp. 639-647, 1994.

J. G. Wang and E. Sung, "EM Enhancement of 3D Head Pose Estimated by Point at Infinity," *Image and Vision Computing*, vol. 25, no. 12, pp. 1864-1874, 2007.

P. Campadelli and R. Lanzarotti, "A Face Recognition System Based on Local Feature Analysis," *Proceedings of the 4th international conference on Audio- and video-based biometric person authentication(AVBPA'03)*, pp. 68-73, Springer-Verlag Berlin, Heidelberg, 2003.

E. Hjelmås and B. K. Low, "Face Detection: A Survey," *Computer Vision and Image Understanding*, vol. 83, no. 3, pp. 236-274, 2001.

J. Tu, Y. Fu, and T. S. Huang, "Locating Nose-Tips and Estimating Head Poses in Images by Tensorposes," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 1, pp. 90-102, 2009.

P. Martins and J. Batista, "Single View Head Pose Estimation," *Proceedings of the 15th IEEE International Conference on Image Processing (ICIP '08)*, pp. 1652-1655, San Diego, California, USA, 2008.

Q. Li, Y. Mao, Z. Wang, and W. Xiang, "Robust Real-Time Detection of Abandoned and Removed Objects," *Proceedings of the 5th International Conference on Image and Graphics (ICIG '09)*, pp. 156-161, Xi'an, China, 2009.

S. C. Huang, "An Advanced Motion Detection Algorithm with Video Quality Analysis for Video Surveillance Systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 1, pp. 1-14, 2011.

M. Vargas, J. M. Milla, S. L. Toral, and F. Barrero, "An Enhanced Background Estimation Algorithm for Vehicle Detection in Urban Traffic Scenes," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 3694-3709, 2010.

W. Wang, J. Yang, and W. Gao, "Modeling Background and Segmenting Moving Objects from Compressed Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, pp. 670-681, 2008.

K. Huang, S. Wang, T. Tan, and S. J. Maybank, "Human Behavior Analysis Based on a New Motion Descriptor," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 12, pp. 1830-1840, 2009.

H. Wang, Q. Lu, X. Wang, G. Lv, and L. Xu, "Method for video incident detection based on biological visual Mechanism," *Proceedings of IEEE International Conference on Automation and Logistics (ICAL '10)*, pp. 114-119, Hong Kong, 2010.

L. Wang, L. Wang, M. Wen, Q. Zhuo, and W. Wang, "Background Subtraction using Incremental Subspace Learning," *Proceedings of IEEE International Conference on Image Processing (ICIP '07)*, vol. 5, pp. 45-48, San Antonio, Texas, USA, 2007.

Y. Liu and Y. Li, "Robust Moving Object Region Extraction Using a Column and Row Projection Algorithm," *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE '10)*, vol. 6, pp. 168-171, Chengdu, China, 2010.

C. C. Chiu, M. Y. Ku, and L. W. Liang, "A Robust Object Segmentation System Using a Probability-Based Background Extraction Algorithm," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 518-528, 2010.

Y. Mu, H. Zhang, H. Wang, and W. Zuo, "Automatic Video Object Segmentation using Graph Cut," *Proceedings of IEEE International Conference on Image Processing (ICIP '07)*, vol. 3, pp. III-377-III-380, San Antonio, Texas, USA, 2007.

J. Park, A. Tabb, and A. C. Kak, "Hierarchical Data Structure for Real-Time Background Subtraction," *Proceedings of IEEE International Conference on Image Processing(ICIP '06)*, pp. 1849-1852, 8-11, Atlanta, GA, USA, 2006.

X. N. Zhang, J. Jiang, Z. H. Liang and C. L. Liu, "Skin Color Enhancement Based on Favorite Skin Color in HSV Color Space," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp.1789-1793, 2010.

D. Chai and K. N. Ngan, "Face Segmentation Using Skin-Color Map in Videophone Applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551-564, 1999.

Z. Liu, J. Yang and N. S. Peng, "An Efficient Face Segmentation Algorithm Based on Binary Partition Tree," *Proceedings of Signal Processing:Image Communication*, vol. 20, no. 4, pp. 295-314, Shanghai, 2005.

H. Sahbi and N. Boujemaa, "Coarse to Fine Face Detection Based on Skin ColorAdaption," *Proceedings of the International ECCV Workshop*, pp. 112-120, Copenhagen, 2002.

T. M. Mahmoud, "A New Fast Skin Color Detection Technique," *World Academy of Science, Engineering and Technology*, pp. 501-505, 2008.

S. K. Singh, D. S. Chauhan, M. Vatsa and R. Singh, "A Robust Skin Color Based Face Detection Algorithm," *Tamkang Journal of Science and Engineering*, vol. 6, no. 4, pp. 227-234, 2003.

S. A. Phung, "A Novel Skin Color Model in YCbCr Color Space And Its Application to Human Face Detection," *Proceedings of IEEE International Conference on Image Processing(ICIP '02)*, vol. 1, I-289 - I-292, New York, 2002.

L. Sigal, S. Sclaroff and V. Athitsos, " Skin Color-based Video Segmentation Under Time-varying Illumination," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 7, pp. 862-877, 2004.

M. Ying, "Automatic Video Object Segmentation Using Graph Cut," *Proceedings of IEEE International Conference on Image Processing(ICIP '07)*, vol. 3, pp. III-377 - III-380, Texas, Usa, 2007.

C. Rother, V. Kolmogorov, and A. Blake, "Grabcut—Interactive Foreground Extraction Using Iterated Graph Cuts," *ACM Transactions on Graphics (SIGGRAPH)*, pp.309-314, New York, 2004.

