

國立交通大學

電機與控制工程學系

碩士論文

應用於 MPEG-4 標準協定之高效率視訊物件擷取技術

A High Efficient Video Segmentation Method and
Application to MPEG-4 Protocol



研究生：許弘昕

指導教授：林進燈 博士

中華民國 九十四 年 七 月

應用於 MPEG-4 標準協定之高效率視訊物件擷取技術

A High Efficient Video Segmentation Method and Application
to MPEG-4 Protocol

研 究 生：許弘昕

Student : Hung-Shin Hsu

指 導 教 授：林進燈 博士

Advisor : Dr. Chin-Teng Lin

國立交通大學

電機與控制工程學系



Submitted to Department of Electrical and Control Engineering

College of Engineering and Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master

in

Electrical and Control Engineering

July 2005

Hsinchu, Taiwan, Republic of China

中華民國 九十四 年 七 月

應用於 MPEG-4 標準協定之高效率視訊物件 擷取技術

研究生：許弘昕

指導教授：林進燈博士

國立交通大學電機與控制工程研究所

摘要

本論文主要提出一個新的視訊物件分割演算技術，應用於 MPEG-4 標準協定中。這個視訊物件分割演算技術主要分為兩部分：初始前景/背景建立與移動物件追蹤。在第一個部分，首先使用多次的改變偵測(Change detection)步驟建立出可信賴的初始前景/背景資訊。接著針對上述步驟所產生的未遮蔽背景區域，利用每個像素位置在每次改變偵測過程中累積的變化資訊，找出並消除屬於未遮蔽背景的區域，改進了只參考最初與最後移動物體位置資訊所造成判斷上的缺失。在第二個部分，我們使用邊緣運算器追蹤移動物件的邊緣，並透過改變偵測和背景預測的方法移除未遮蔽背景的邊緣，進而擷取出完整的移動物件。實驗證明在改進過未遮蔽背景消除方法之後，我們可以較有效率地偵測出未遮蔽背景區域，進而更精確地分割出移動物件與背景。

之後我們將所提出的視訊物件分割演算法加入現今的 MPEG-4 標準編/解碼器中，進而建立出以物件輪廓為基礎的編/解碼架構。在編碼器(Encoder)架構部分，首先將影像利用所提出的視訊分割技術分離出移動的視訊物件與不動的背景，接著在編碼過程中只編碼一次最初的背景資訊，不需要如以往每張編碼影像都帶有背景資訊；在解碼器(Decoder)架構部分再將背景資訊填補到每個解碼影像中，

由於在填補過程中會造成重複區域(Overlap region)與無資訊區域(Non-information region)的產生，故我們分別使用前景優先性(Foreground priority)與背景補償(Background compensation)方式消除上述的問題。實驗證明當影片內移動物體移動程度越大時，所提出的方式與標準 MPEG-4 編/解碼架構比較，更能夠有效地減少所需傳輸的資訊量，即能夠提高影片壓縮率。而在解碼影像畫質方面，與使用傳統的解碼方式比較也能維持在一定的水準。



A High Efficient Video Segmentation Method and Application to MPEG-4 Protocol

Student: Hung-Shin Hsu

Advisor: Dr. Chin-Teng Lin

Department of Electrical and Control Engineering
National Chiao-Tung University

Abstract

In this thesis, first we propose a new method to eliminate the uncovered-background regions in the proposed video object segmentation algorithm. The video object segmentation algorithm mainly includes two parts: initial foreground/background construction and moving object tracking. In the first part, we construct the reliable initial foreground/background by using several times of change detection step. For the uncovered-background regions, which are produced in the previous procedure. According to the collected change information of each pixel during change detection steps, we can resolve the lacks of previous proposed method, which only makes use of the initially and finally located information of moving object as the reference to detect the uncovered-background region. Finally the initial foreground/background is updated. In the secondary part, we use edge operator to extract the edge of the moving object, the uncovered-background edge can be eliminated by using the change detection and background prediction methods. Finally the complete moving object region can be extracted. The experimental results show that the updated method of uncovered-background elimination can detect these regions more efficiently and the

moving object and background are separated more precisely.

Next, we combine the segmentation algorithm into current MPEG-4 standard to build an object-shape based coding architecture. In the encoder architecture, firstly the segmentation algorithm separates the encoding frame into moving object and monotonous background. Then, the background information is only encoded once in the first frame of the encoding procedure, and the following encoding frames will only include the moving object information. Therefore, identical background information will not be repeatedly included in each encoded frame. In the decoder architecture, the background addition step is used to add the background information of first decoding frame into each following decoding frames. Then, we use the foreground priority and background compensation methods to resolve the overlap regions and non-information regions problems respectively during the background addition step. The experimental results show that as the moving objects of video contents change more obviously, the proposed coding architecture will provide better compression ratio compared with the traditional MPEG-4 coding architecture, and the qualities of encoded images can also get good performance.

誌謝

轉眼兩年之間碩士求學生涯已過，也即將要步出陪伴我走過大學與碩士生涯的交大校園。首先感謝我的指導教授林進燈老師，不僅在研究上給予細心的指導，使得本論文能夠順利完成；另外在為人處事與求學態度上給予啟蒙與指導，受益良多。

另外要感謝實驗室鶴章和朝暉學長的細心指導與建議，讓我在研究過程中增進了許多的專業知識。也感謝在這兩年碩士生涯陪伴著我的同學們，不管是朝夕相處的實驗室同學們，或是大學同窗們，很感謝你們不管是在快樂或困難的時刻都能夠與我一起分享與承擔，讓我的生活充滿了溫馨與感動；真的很感謝你們。

最後要感謝父母的支持，在我遇到生活上的困境時，能夠教導我如何面對並解決問題，讓我能夠專心於課業上的研究，順利完成學業。很感謝你們的犧牲與付出，謝謝!!



目錄

中文摘要.....	i
英文摘要.....	iii
致謝.....	v
目錄.....	vi
圖目錄.....	vii
表目錄.....	ix
第一章 序論.....	1
第二章 相關理論.....	3
2-1 各種影像物件擷取的理論與方法.....	3
2-1-1 空間資訊的擷取方式：.....	3
2-1-2 時間資訊的擷取方式：.....	6
2-2 未遮蔽背景區域(Uncovered-Background Region)的產生.....	6
2-3 MPEG-4 標準協定：.....	7
第三章 動態視訊物件擷取技術與未遮蔽背景之消除方法.....	10
3-1 改變偵測及初始背景/前景之建立.....	10
3-2 未遮蔽背景區域的消除(Uncovered-background elimination).....	13
3-3 移動物體追蹤.....	18
第四章 以物件輪廓為基礎之 MPEG-4 編/解碼架構.....	21
4-1 編碼器架構.....	21
4-2 解碼器架構.....	23
4-3 背景填補架構所面對的問題及其解決方式.....	24
4-3-1 重疊區域與無資訊區域的產生.....	25
4-3-2 前景優先性與背景補償(Foreground Priority and Background Compensation).....	26
第五章 實驗結果.....	30
5-1 動態物件擷取演算法.....	30
5-2 提出的 MPEG-4 編/解碼架構.....	35
第六章 結論與未來展望.....	44
參考文獻.....	46

圖目錄

Fig. 2-1	Threshold based segmentation.	4
Fig. 2-2	分水嶺理論流程圖.	5
Fig. 2-3	Three kinds of areas in a video scene.	6
Fig. 3-1	初始前景/背景建立流程圖	10
Fig. 3-2	Weather sequence (a)~(i) 9 個連續改變偵測(change detection)結果.	11
Fig. 3-3	初始移動物體擷取結果	12
Fig. 3-4	設定不同 UB_{th} 所得到的未遮蔽背景區域消除的結果(a)當 $UB_{th}=3$ 時 ; (b)當 $UB_{th}=5$ 時 ; (c)當 $UB_{th}=7$ 時 ; (d)當 $UB_{th}=9$ 時.....	15
Fig. 3-5	The 8-direction connected component filter.	16
Fig. 3-6	(a)水平標記記錄 ; (b)垂直標記記錄.....	16
Fig. 3-7	等值關係矩陣	17
Fig. 3-8	經由反射性與傳遞性步驟所得到的最後等值關係矩陣.....	17
Fig. 3-9	最後水平標記記錄結果	18
Fig. 3-10	將 Fig. 3-4(a) 經過 connected-component 與 dilation 步驟所得到 初始前景結果.....	18
Fig. 4-1	提出的 MPEG-4 編碼器架構	21
Fig. 4-2	所提出編碼器位元串流架構圖.....	22
Fig. 4-3	(a)解碼器端的背景填補架構 ; (b)經由背景填補架構所得到的完整 解碼影像。	23
Fig. 4-4	(a) I -frame 中所包含的背景資訊 ; (b) P -frame 中所擁有的移動物體資 訊 ; (c)背景填補過程後產生的重疊區域與無資訊區域現象。	25
Fig. 4-5	重疊區域(白色區域)與無資訊區域(黑色區域)現象.....	26
Fig. 4-6	將 Fig. 4-5 經由前景優先性步驟所得到的解碼影像.....	27
Fig. 4-7	(a)編碼的影像資訊內容 ; (b)經由移動物件擷取演算法所得到的 每張影像背景資訊內容。	28
Fig. 4-8	將 Fig. 4-5 經由前景優先性與背景補償步驟所得到的解碼影像....	29
Fig. 5-1	(a)輸入的#5、#25、#45、和#65 影像 ; (b)使用先前之演算法所擷取 出的#5、#25、#45、和#65 移動物件影像 ; (c)使用改進之演算法所 擷取出的#5、#25、#45、和#65 移動物件影像。	32
Fig. 5-2	(a)使用先前之演算法所分離出的初始背景影像 ; (b)使用改進之演算 法所分離出的初始背景影像。	31
Fig. 5-3	(a)輸入的#10、#30、#50、和#70 影像 ; (b)使用先前之演算法所擷 取出的#10、#30、#50、和#70 移動物件影像 ; (c)使用改進之演算法 所擷取出的#10、#30、#50、和#70 移動物件影像。	34

Fig. 5-4	(a)使用先前之演算法所分離出的初始背景影像；(b)使用改進之演算法所分離出的初始背景影像。.....	33
Fig. 5-5	Akiyo sequence (a)使用原本編/解碼架構所得到的#15、#35、#55、#75 影像輸出；(b)使用提出的編/解碼架構所得到的#15、#35、#55、#75 影像輸出。.....	37
Fig. 5-6	Weather sequence (a)使用原本編/解碼架構所得到的#35、#55、#75、#95 影像輸出；(b)使用提出的編/解碼架構所得到的#35、#55、#75、#95 影像輸出。.....	39
Fig. 5-7	Salesman sequence (a)使用原本編/解碼架構所得到的#40、#60、#80、#100 影像輸出；(b)使用提出的編/解碼架構所得到的#40、#60、#80、#100 影像輸出。.....	41
Fig. 5-8	Hall Monitor sequence (a)使用原本編/解碼架構所得到的#27、#77、#117、#167 影像輸出；(b)使用提出的編/解碼架構所得到的#27、#77、#117、#167 影像輸出。.....	43



表目錄

Table5-1	Akiyo sequence壓縮率比較.....	38
Table5-2	Weather sequence壓縮率比較.....	40
Table5-3	Salesman sequence壓縮率比較.....	42
Table5-4	Hall Monitor sequence壓縮率比較.....	44



第一章 序論

近年來，愈來愈多類的數位內容出現在通訊與多媒體應用領域，例如：影像電話，視訊會議，數位互動電視...等，資訊的編碼與壓縮逐漸變成一個令人不可忽視的課題。由於傳統的影音壓縮標準如 MPEG-1[17]，MPEG-2[18]是以”框架(frame)”作為編碼的基礎，在面對這類應用時已經無法滿足以上的需求。另外，傳統以框架作為壓縮的方式往往會在壓縮過後的資料上帶有許多不必要或是多餘的資訊，例如：無變動的背景。這些都是造成壓縮資料中存在許多不必要資訊的原因。隨著現今多媒體逐漸以視訊物件作為基礎(video-object-based)的應用方向，新的影音編碼與壓縮架構也就變得更加需要。

為了達到上述的需求，一種新的影音編碼架構：MPEG-4[19][20]因而產生。不像以往傳統的編碼方式，MPEG-4 不僅能提供強大的編碼效益，針對多媒體應用，例如以視訊內容為基礎的互動性(interactivity)與可變性(scalability)，亦能提供各種不同的新功能。與傳統編碼方式最大的不同就是 MPEG-4 能將在同一場景的不同內容分割為多樣的物件。例如：固定的背景，移動中的車輛，...等。每一個物件都稱做是一個單一的視訊物件平面(Video Object Plane)。接著，這些物件可以針對其特性分別被編碼與解碼。藉由這樣的概念，我們針對較有興趣的物件內容，可以有效地傳送、儲存、並且呈現這些多媒體資訊。

雖然 MPEG-4 標準提供了多種以視訊內容為基礎的編碼功能，但視訊物件平面的擷取在 MPEG-4 編碼架構之中並沒有一定的標準，因此使用者可以針對自身的需求加以規定所需的物件擷取方式。然而由於視訊物件平面的擷取並無法單獨只由單一的內容特徵來取得，例如：飽和度(intensity)、色彩資訊、移動資訊...等。因此，在現今的視訊物件平面的擷取方式往往只能利用半自動或是人工方式來得到。在我們所提出的視訊物件擷取架構之中，針對固定的背景狀況之

下，能夠挖取出移動的視訊物件。其架構包括了初始前景/背景的建立與移動物件的追蹤。然而，在影片起始部份，因為物件的移動，造成在初始前景/背景建立的過程之中，某些應屬於背景的部份被誤判成移動物件區域的現象產生。此類判斷錯誤的區域稱作未遮蔽背景區域(Uncovered-background regions)[1][2]。這些區域會在接下來的移動物件追蹤步驟中造成移動物件區域上的誤判。為了要挖取出更精確的移動物件，我們提出一個針對此類現象偵測與消除的改進方式。另一方面，針對所提出的視訊物件擷取技術，我們將與現今的 MPEG-4 編/解碼架構結合，使其在編碼過程中只傳送一次的背景資訊即可，不需要如以往每張框架都帶有背景資訊；接著在解碼架構部分再將背景資訊補償。以達到減少在已編碼資料中不必要的資訊。

本篇論文中，第二章介紹關於動態影像擷取技術的各種理論與方法，另外針對未遮蔽背景區域的產生原因及其影響做說明，另外也針對現今 MPEG-4 架構特性做詳細的介紹。第三章則是介紹我們所提出的移動物件擷取技術並詳述所改進的未遮蔽背景區域消除架構。第四章則是針對提出的視訊移動物件擷取技術與現今的 MPEG-4 標準結合，並說明其編/解碼原理。第五章則是針對我們所提出的移動物件擷取技術與改進的未遮蔽背景方法做效果之比較；另外，我們也將其改進的 MPEG-4 編/解碼架構與傳統的 MPEG-4 架構針對編碼過後的檔案資料大小與解碼之後的影像結果做討論與說明。最後，在第六章做結論和未來展望。

第二章 相關理論

2-1 各種影像物件擷取的理論與方法

在影像中物件擷取的領域中大約可以分成兩大方式：以空間資訊為主的擷取方式 (Spatial-based segmentation) 與以時間資訊為主的擷取方式 (Temporal-based segmentation)。空間資訊上的擷取方式主要是以特徵擷取 (feature extraction) 為基礎。例如：以統計方式處理之後再取門檻值 (histogram threshold)，邊緣資訊擷取 (edge extraction) 等。而時間資訊上的擷取方式主要是以時間上兩相鄰的影像所產生的變化資訊為主。

2-1-1 空間資訊的擷取方式：



空間資訊的擷取方式主要是以單張影像作為處理。傳統上的空間資訊擷取方式主要有以下方法：

(1) 以門檻值為基礎的擷取方式(threshold based segmentation)[3]:

此類方式(如圖 Fig.2-1 所示)在影像擷取之中是最基本的方式之一，並且經常應用於灰階的影像中。此類方法以選擇一個門檻值 T 作為挖取物件的依據。一張經由此方式所得到的影像可由 Eq.(2.1) 表示

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T \end{cases} \quad (2.1)$$

其中 $g(x, y)$ 代表屬於前景部份區域， $f(x, y)$ 代表影像的灰階值；但是由於影像的灰階分布往往是分散的，並且不同區域的灰階值也會部分重疊，

因此此種方式應用在真實狀況中並不是如此完美。

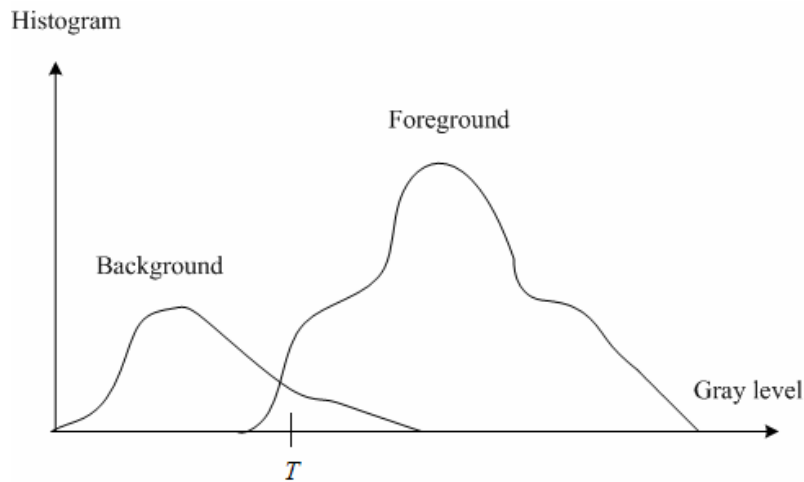


Fig.2-1 Threshold based segmentation.

(2) 以邊緣資訊為基礎的擷取方式(Edge Based Segmentation)

邊緣是由位於兩區域輪廓(boundary)間的像素群所形成的集合，因此我們可以利用許多邊緣偵測技術[3]來偵測物件區域。但是由於色彩與亮度等因素，這些偵測到的邊緣資訊往往是不連續或是因為過度萃取所得到的無用邊緣資訊。然而，真實的物件區域輪廓必須是完整且封閉的，因此，經由現今存在的邊緣偵測方式所萃取出來的邊緣資訊，針對物件區域輪廓的擷取往往只能作為參考之用。

(3) 以區域資訊為基礎的擷取方式(Region Based Segmentation)

在一張影像之中，相同的區域往往都具有相似的視覺特徵，例如：灰階值、色彩、或是材質等。在這類的方法之中，最常用到的概念就是分割與合併(split and merge)的技術[4]-[6]。此類方法的效果取決於所選擇的同種性(homogeneity)決定方式。而其中最受歡迎的就是分水嶺理論(watershed algorithm)[7]，其模型如 Fig.2-2 所示。分水嶺理論主要是以最

小值(minima)作為標記點(marker)，而標記點一般則是某區域的起始位置。此方法主要有以下步驟：

步驟一：針對所處理的影像先取型態上的梯度運算(morphological gradient operation)。

步驟二：針對每個像素所得到的梯度值，由低至高逐一處理。

步驟三：(1)針對已被標記區域(labeled region)附近的像素優先排列為此區域的一部份(圖中白球附近的灰球區域)
(2)尋找未被標記區域附近的像素並且給予新的標記(圖中黑球附近)

步驟四：隨著不同標記所形成的不同區域彼此間的邊緣即是此影像的分水嶺部份

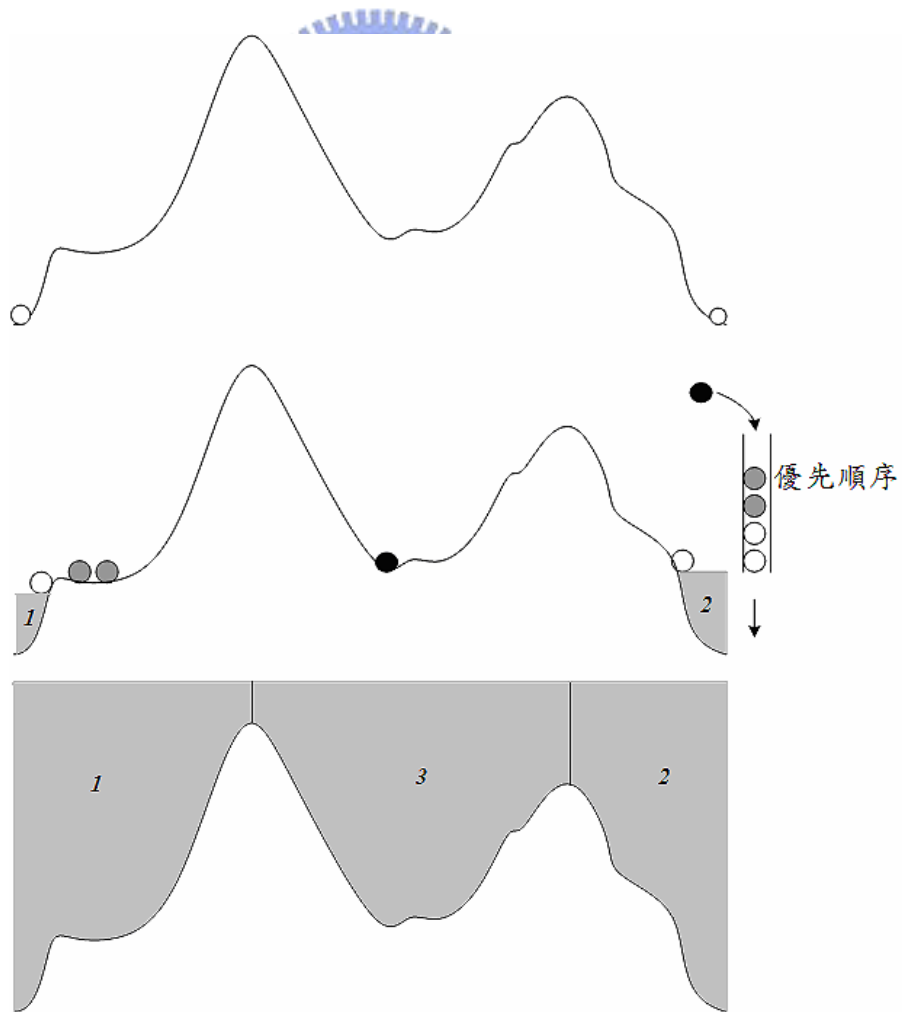


Fig.2-2 分水嶺理論流程圖.

2-1-2 時間資訊的擷取方式：

此方式主要是以在時間中相鄰的兩張影像資訊作為主要的擷取依據。由於移動物件通常包含了飽和度(intensity)上的變化，因此我們能夠利用此類資訊來偵測移動物體的位置並且擷取出來。其中最主要的就是以移動資訊為主的擷取方式(motion-based segmentation)。其中，改變偵測(Change detection)技術[8]-[13]是其中最不耗費運算能力的一種擷取方式。此種方法藉由時間上兩相鄰的影像能夠將時間上的變動與不變資訊分隔出來，因此移動物件即可從被判定為變動的區域擷取出。利用此方式擷取出來的效果主要取決於兩因素：第一是門檻值(threshold)的選擇，其決定分離變動區域與不變區域部分。第二則是有一合理的機制能夠消除小區域的雜訊，例如在改變區域很大的範圍之中所帶有的未變動區域。

2-2 未遮蔽背景區域(Uncovered-Background Region)的產生

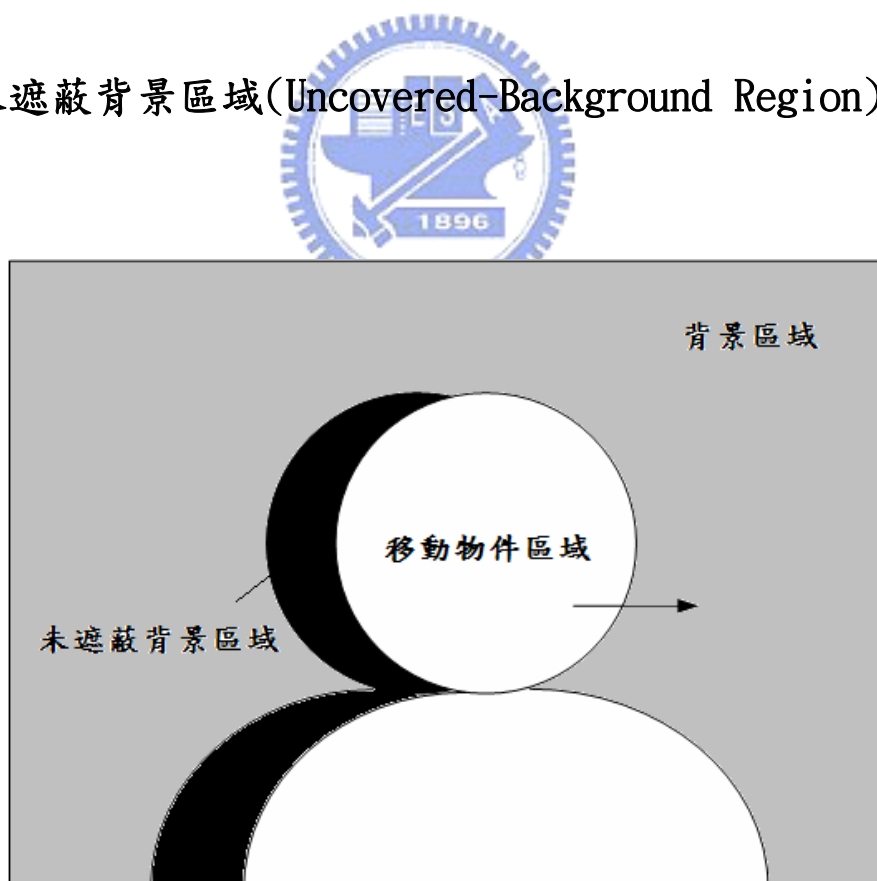


Fig.2-3 Three kinds of areas in a video scene.

我們以 Fig.2-3 說明在偵測移動物體區域時，未遮蔽背景區域產生的原因：

首先，當物體由畫面中的左方移動到右方時，原本在物體後方的背景資訊顯露出來，但是由於在動態物體擷取過程之中，此區域的飽和度有變化，再加上新的背景資訊來不及更新，因此會造成此區域被誤判成移動物體。而這些原本應是屬於背景資訊卻誤判成移動物體區域即稱作未遮蔽區域。

2-3 MPEG-4 標準協定：

針對不同的多媒體服務，MPEG-4 能夠提供各種不同的內建工具來加強不同類型的應用，主要包括下列三大特點：

(1) 壓縮效率：

MPEG-4 是以先前 MPEG-1、MPEG-2 標準為基礎，但提供更高編碼效率，增加 MPEG-4 應用的市場接受度。

(2) 內容導向互動性 (content-based interactivity):

把視訊當成物件，而非視訊圖框，即可讓內容導向應用(content-based applications)付諸實現；此時只須提供更高效率的物件表示法、物件操控、位元串流編輯和物件導向延展能力(object-based scalability)，就能將內容互動性帶至更高水準。

(3) 適用於所有傳輸媒介：

就算在容易發生錯誤的環境，MPEG-4 也很強健可靠，故可用於各種傳輸媒介，包括行動網路和實體連接線路。

至於在 MPEG-4 內建的多媒體應用工具中，主要具有四大特性：

(1) 視訊壓縮工具

視訊編碼解碼器(video codec)可以除去空間和時間的冗餘性，達到壓縮視訊的目的。Intra Coded VOPS (I-VOPs)會利用視訊物件平面包含的資訊進行編碼，這能移除部份的空間冗餘性，圖框間編碼(inter coding)則會透過移動估算和補償來利用圖框間的時間冗餘性。圖框間編碼有兩種模式，第一種是根據前面的視訊物件平面進行預測，這種模式稱為P-VOP；第二種則根據前面和後面的視訊物件平面進行預測，稱為B-VOP。這兩種編碼技術都是以過去的視訊標準為基礎，MPEG-4 則會提供額外的工具來增加壓縮效率、抗錯性和不同類型視訊物件的編碼能力。

(2) 形狀編碼工具 (Shape Coding Tools)

MPEG-4 提供許多工具，可對各種形狀的物件進行編碼。二位元形狀(binary shape)資訊可用來定義特定時間點上，物件的那些部份(像素)屬於視訊物件；二位元形狀資訊還採用移動補償式區塊導向(motion compensated block-based)編碼技術，能同時支援無失真(lossless)和失真(lossy)編碼方式。灰階形狀(gray scale shape)資訊與二位元形狀資訊非常類似，但每個像素或矩陣元素不再是簡單的二位元值，而是在某個範圍內的數值(通常是0到255)，代表該像素的透明度。灰階形狀資訊也採用區塊導向的移動補償技術來協助編碼，但是只支援無失真編碼方式。

(3) Sprite 編碼

在整段視訊內容(video segment)中，視訊物件的某些部份會出現在場景內，sprite 就是由這些區域所組成。「背景 sprite」(background sprite)是很典型的範例，包含攝影機跟拍鏡頭(camera-panning sequence)的所有

背景像素；它基本上是靜態影像，只需在傳輸開始時傳送一次，sprite 能在這些情形下提供很高的壓縮效率，這也是 MPEG-4 包含它們的主要原因。sprite 編碼技術不但適用於合成物件，也能用於自然場景中的剛體(rigid)運動物件。

(4) 延展性

MPEG-4 利用多個視訊物件層(VOL)來實現空間延展性(spatial scalability)和時間延展性(temporal scalability)，例如我們可以利用兩個視訊物件層來提供空間延展性，一個是基礎層(base-layer)，另一個則是加強層(enhancement-layer)，加強層可以改善基礎層所提供視訊物件平面的空間解析度；同樣的，只要利用時間延展性，當目標圖框速率高於基礎層提供的速率時，就可以利用加強層進行解碼；換言之，時間延展性會讓畫面的動作更平順。



第三章 動態視訊物件擷取技術與未遮蔽背景之消除方法

一般若單獨使用移動資訊(motion information)來擷取移動物體時，由於移動物體區域並非皆是移動的狀態，往往會造成無法擷取完整的移動物體。因此我們利用不動的背景資訊找出完整的移動物體區域。但在場景中要出現完整的背景資訊是不容易的，因此在我們的演算法中將整個擷取架構分成兩大部分：初始前景/背景建立與移動物件輪廓追蹤。

3-1 改變偵測及初始背景/前景之建立

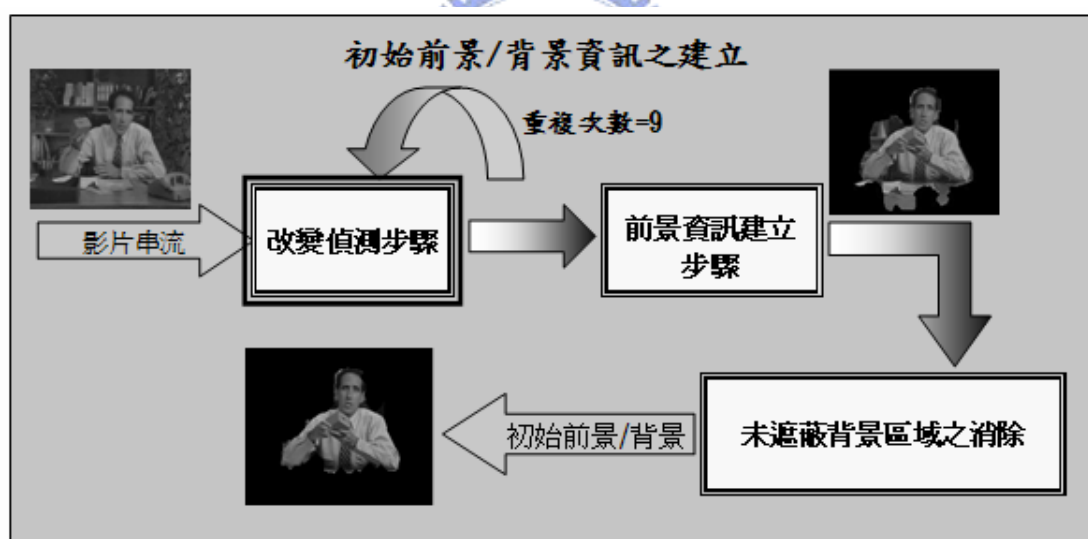


Fig.3-1 初始前景/背景建立流程圖

在初始背景建立部分，如 Fig.3-1 所示。我們主要利用改變偵測(Change Detection)技術先將約略的移動物體區域分離出來，其表示式如 Eq.3-1 所示：

$$D_{ij}(m,n) = \begin{cases} 1 & , \text{if } |f(m,n,t_i) - f(m,n,t_j)| \geq Vth \\ 0 & , \text{Otherwise} \end{cases} \quad (3-1)$$

其中 $f(m,n,t_i)$ 與 $f(m,n,t_j)$ 分別為在時間 t_i 與 t_j 的影像，而 V_{th} 則是門檻值，若 $D_{ij}(m,n)=1$ 時此像素則認為是屬於移動區域，反之則屬於背景。然而僅單一次的改變偵測並無法將完整的移動物體挖取出來，在物件內部往往會有誤判的區域；因此在此論文中，我們設定重複上述改變偵測的步驟 9 次，進而能夠累積較多的移動物件空間資訊，將其移動物件內部缺少部分補齊。其每次改變偵測結果分別如 Fig.3-2(a)~(i)所示。

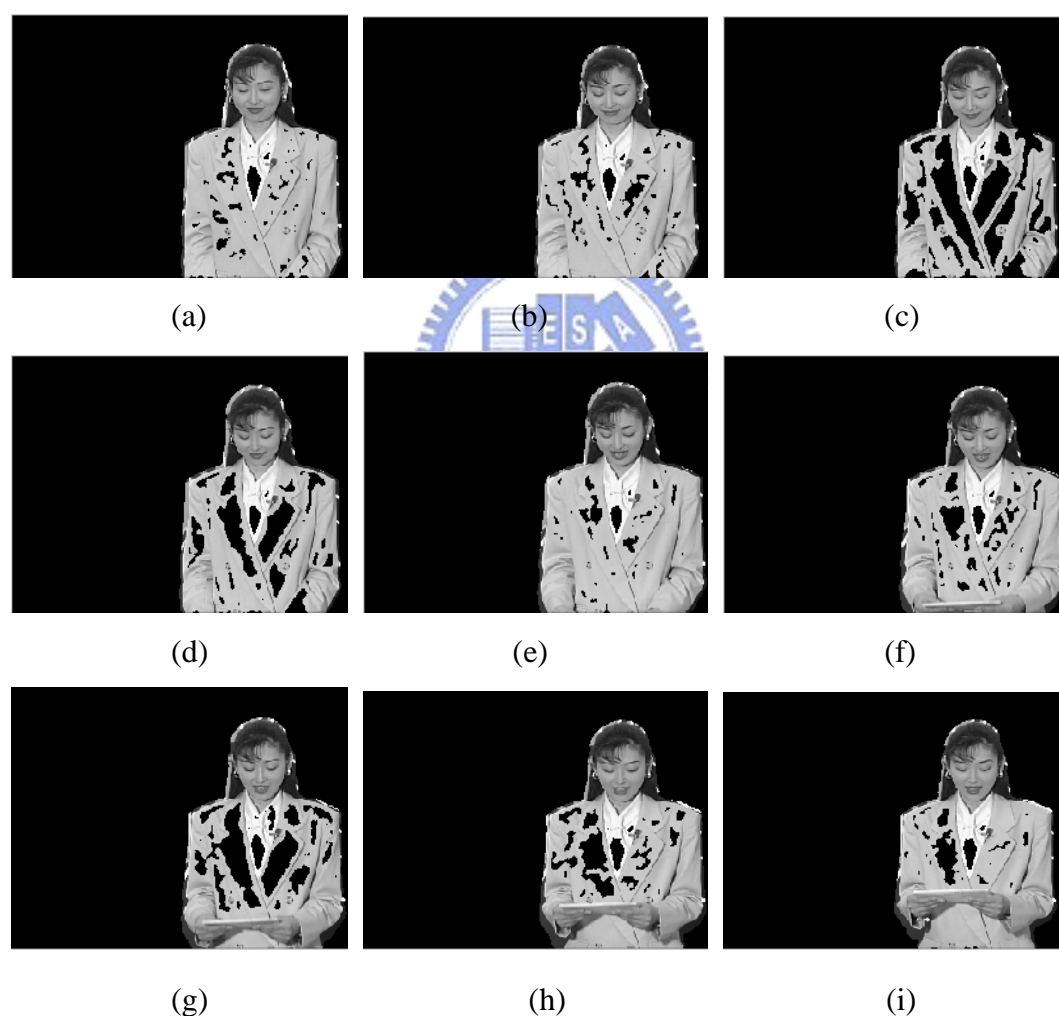


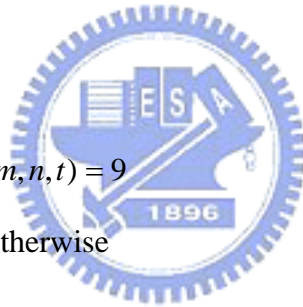
Fig.3-2 Weather sequence (a)~(i) 9個連續改變偵測(change detection)結果

接著，根據之前累積的位置資訊，我們能夠將初始前景/背景分離出來。在初始前景/背景建立的架構中，主要有兩步驟：首先我們求得每一個像素位置的背景累積次數，其方法如 Eq.3-2 所示：

$$BGI(m,n,t_i) = \begin{cases} BGI(m,n,t_j)+1 & , \text{if } D_{ij}(m,n)=0 \\ 0 & , \text{Otherwise} \end{cases} \quad (3-2)$$

其中 $BGI(m,n,t_j)$ 代表的是每經過一次改變偵測之後此像素被認為是背景的累計次數。在此我們設定若某像素在這 9 次的改變偵測過程之中，皆被判斷屬於背景區域時，則此像素確定是屬於背景區域，否則為移動物體區域。其可由 Eq.3-3 表示：

$$M(m,n,t) = \begin{cases} 0 & , \text{if } BGI(m,n,t) = 9 \\ f(m,n,t) & , \text{Otherwise} \end{cases} \quad (3-3)$$




其中 $f(m,n,t)$ 為此時影像，而 $M(m,n,t)$ 則是最後被建立出來的移動物體影像，反之其餘的區域則是被認為是背景區域；其擷取出來的移動物體如 Fig.3-3 所示：



Fig.3-3 初始移動物體擷取結果

然而，由於是利用改變偵測作為累積移動物體位置的方式，若物體移動範圍過大，則未遮蔽背景的現象就極有可能產生。以 Fig.3-3 為例：由於在這 9 次的改變偵測過程中物體由左方移動到右方，因此部份區域的 BI 值無法達到符合背景區域的要求，造成這些區域在此時應屬於背景卻被誤判成為移動區域，而這些區域即是未遮蔽背景區域。因此為了挖取出精確的物件區域，必須要有一處理步驟將此現象消除。

3-2 未遮蔽背景區域的消除 (Uncovered-background elimination)



先前所提出的未遮蔽背景消除架構中，主要是利用移動物體區域上的位置變動資訊作為偵測未遮蔽背景的依據。其方法首先利用影像簡化方式(Image Simplification)將建立出來的初始移動區域分成不同的區塊。接著利用第一次與最後一次的改變偵測所分別擷取出來的移動物體區域互相比較；若是第一次擷取出來的物件區域，在最後一次時卻不再是屬於物件區域時，此範圍在初始移動區域所對應的區塊則是屬於未遮蔽背景的区域。

由於此方法必須要先將移動區域利用類似分水嶺技術將其分成不同的區塊，因此必須要花費較多的運算時間；另外，由於此種方式只有比較第一次與最後一次改變偵測所變動的物件區域位置，若是發生未遮蔽背景區域是在其餘的改變偵測步驟時，此種方法將無法順利消除此現象。

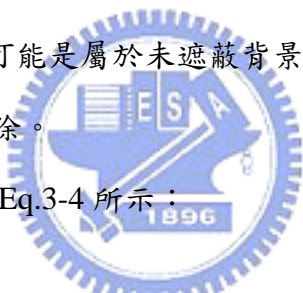
為了改善上述方式的缺失，我們參考經由多次改變偵測對於每個像素所累積的變動資訊來判斷未遮蔽背景區域，改善了之前只有使用部分改變偵測資訊所造

成的缺失。我們所提出的方法如下：

由於未遮蔽背景區域是由於 BGI 值無法達到符合背景區域的要求所造成；另外，未遮蔽背景區域一般會發生在移動物體輪廓的邊緣；因此，我們可以試著從這兩點特性找出消除的方式。

首先，我們針對影像中的每個像素所累積的 BGI 值作為探討的對象。在多次的改變偵測過程中，若某一區域一直是屬於背景的區域，在其範圍內的像素，在最後所累積的 BGI 值應會是屬於較高的值，以我們的方式大約在 7~9 之間。反之，若是此區域一直都是屬於移動物體的區域，則其 BGI 值應會較低，大約在 0~2 之間。因此若其區域的 BGI 值是介於這兩者範圍以外，並且其位置是在移動物體輪廓邊緣，則極有可能是屬於未遮蔽背景區域。因此若能找出一個適當的門檻值則可以將此現象消除。

未遮蔽背景消除方法可以以 Eq.3-4 所示：


$$M'(m,n,t) = \begin{cases} 0, & \text{if } BGI(m,n,t) \geq UBth \\ M(m,n,t), & \text{Otherwise} \end{cases} \quad (3-4)$$

其中 $M'(m,n,t)$ 是經過未遮蔽背景消除之後的移動物體區域，而 $UBth$ 則是其門檻值。若有一像素其累積後的 BGI 值大於設定的門檻值，此像素則被判斷為背景或是未遮蔽背景等區域，反之則屬於移動物體區域。

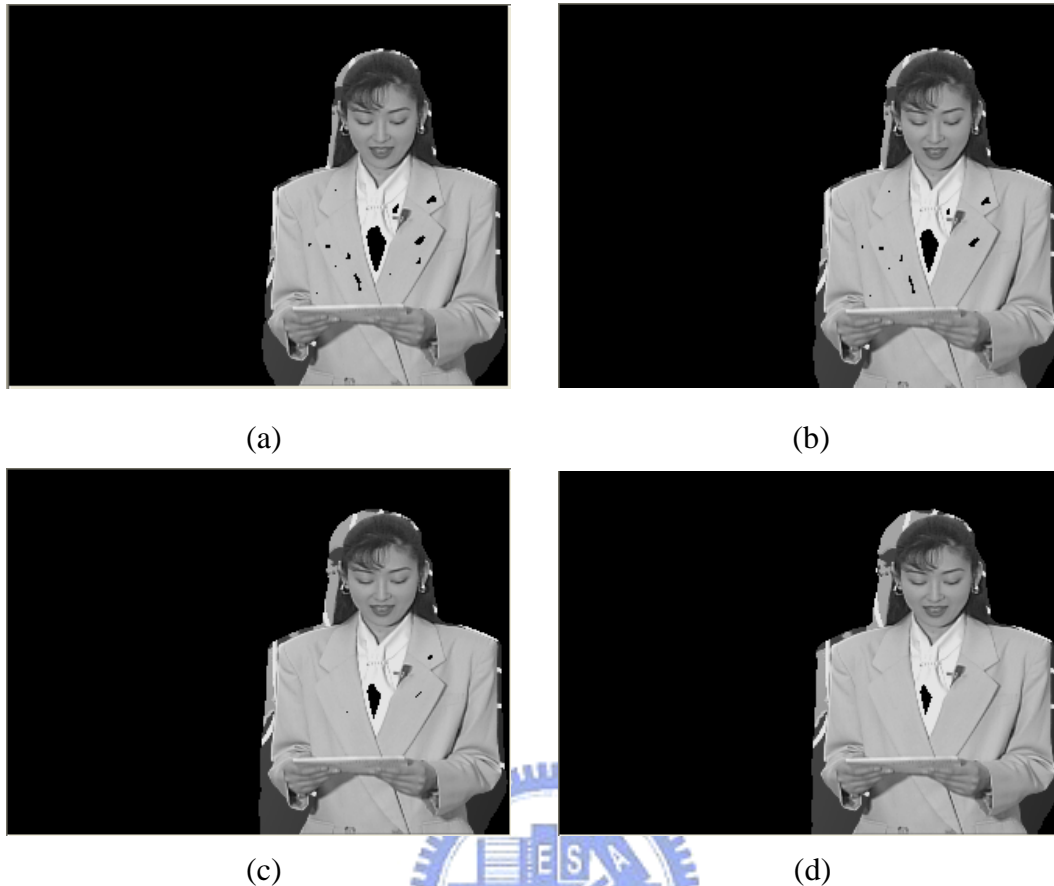


Fig.3-4 設定不同 $UBth$ 所得到的未遮蔽背景區域消除的結果 (a)當 $UBth=3$ 時(b)當 $UBth=5$ 時(c)當 $UBth=7$ 時 (d)當 $UBth=9$ 時

Figure.3-4 顯示出當設定不同門檻值時所得到的未遮蔽背景區域消除結果；由結果可發現當門檻值設定較小時，其未遮蔽背景區域消除能力會愈好；但另一問題則是由於在移動物件內部，其飽和度(intensity)變動量不像邊緣如此明顯，因此部份區域的 BGI 值會較大；若是將門檻值設定過低，則會將上述區域誤判成背景區域，造成移動物件內部有缺洞產生。針對上述的問題，我們可以利用後處理的方式將這類缺洞填補，以得到完整的初始移動物體輪廓與背景。

針對後處理的部份，我們主要是以 connected-component 濾波器[14]來加以消除，其主要的功用是能對於不同物件發掘並分類其各別相關的組成區域。接著，根據事先已設定好的門檻值，就能消除較大範圍的雜訊區域。在此，我們使

用 8-direction connected-component 來針對一張影像中的每一像素作逐一的掃描與處理，其 8 個方向標示如 Fig.3-5 所示。

3	2	1
4		0
5	6	7

Fig.3-5 The 8-direction connected component filter

首先，我們先使用水平方向與垂直方向(Fig.3-5中的0、2、4、6)針對影像中的像素作逐一的掃描，如此我們就能分別建立出水平標記記錄(horizontal label buffer)與垂直標記記錄(vertical label buffer)。標記的方法是在一二元影像(binary image)中，假設P為在掃描過程中正在處理的像素，假設P像素的二進位值為1，且其右側像素二進位值也為1，此右側像素則與P像素標記同一編號，否則則標記另一新的編號，如此則能建立出水平掃描之後的標記記錄(如Fig.3-6(a)所示)。與上述方法相似，針對在P像素下方的像素也是以此方式標記，如此則能建立出水平掃描之後的標記記錄(如Fig.3-6(b)所示)。接下來將水平標記記錄與垂直標記記錄所得到的資訊加以合併以建立出等值關係矩陣(equivalent relation matrix)。

0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	2	2	0	1	3	6	10	14
0	3	3	3	3	3	0	2	4	7	11	15
0	4	5	5	5	5	0	2	5	8	12	16
0	6	7	8	8	8	0	2	5	9	13	17

(a)

(b)

Fig.3-6 (a)水平標記記錄；(b)垂直標記記錄

將水平標記記錄與垂直標記記錄互相比較，我們可以發現到在水平標記記錄中，編號3與編號4應該屬於同一群，編號4與編號6應該屬於同一群，編號5與編號7應該屬於同一群。因此，我們可以藉由二位元矩陣(binary matrix)將上述

的現象建立出一等值關係式。所建立出的等值關係矩陣如 Fig.3-7 所示。

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0
4	0	0	1	0	0	1	0	0
5	0	0	0	0	0	0	1	0
6	0	0	0	1	0	0	0	0
7	0	0	0	0	1	0	0	0
8	0	0	0	0	0	0	0	0

Fig.3-7 等值關係矩陣

接著，由於等值關係矩陣必須要符合反射性(reflexivity)、對稱性(symmetry)、和傳遞性(transition)，因此我們必須要針對等值關係矩陣內容作上述的處理。為了符合反射性，我們將矩陣內由左上方到右下方所形成的對角線，其中的每個矩陣單元都設定為1。為了符合傳遞性，我們使用到Floyd-Warshall演算方式，其內容可用Eq. 3-5表示：

```

For j = 1 to n
  For i = 1 to n
    If L[i,j] = 1 then
      For k = 1 to n
        L[i,k] = L[i,k] OR L[j,k]

```

(3-5)

經過上述處理過後，我們可以建立起最後的等值關係矩陣(如Fig.3-8所示)。接著依照所得到的等值關係矩陣，我們將水平標記記錄結果再加以修正，最後所得到的水平標記記錄結果如Fig.3-9所示。

	1	2	3	4	5	6	7	8
1	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	0	0	1	1	0	1	0	0
4	0	0	1	1	0	1	0	0
5	0	0	0	0	1	0	1	0
6	0	0	1	1	0	1	0	0
7	0	0	0	0	1	0	1	0
8	0	0	0	0	0	0	0	1

Fig.3-8 經由反射性與傳遞性步驟所得到的最後等值關係矩陣

0	0	0	0	0	0
0	0	0	0	0	0
0	1	1	1	2	2
0	3	3	3	3	3
0	3	4	4	4	4
0	3	4	5	5	5

Fig.3-9 最後水平標記記錄結果

在完成水平與垂直的標記步驟之後，我們接下來針對對角線方向(1、3、5、7)的像素做類似上述的步驟，如此可將影像中個每個區域各別標記出來。在完成標記步驟之後，我們分別計數在相同標記編號區域中的像素總數，若數量超過一定的門檻值之後，此區域則被歸類成移動物件區域的一部份，否則屬於雜訊區域，若這些雜訊區域是在移動物件內部時，我們則另外利用影像處理上的dilation步驟將其填補。經過上述後處理步驟所得到的初始前景影像如Fig.3-10所示。

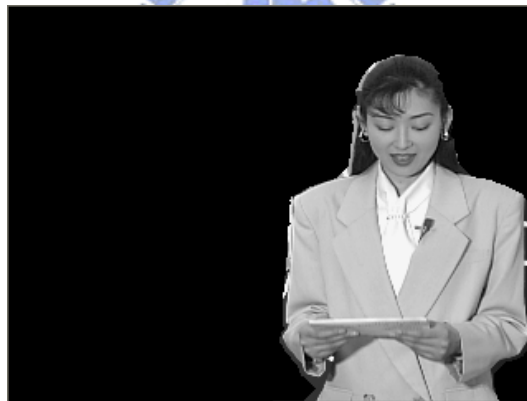


Fig.3-10 將 Fig.3-4(a)經過 connected-component 與 dilation 步驟所得到的初始前景結果

3-3 移動物體追蹤

在擷取出初始的移動物體區域與背景之後，下一步則是追蹤移動物體的移動位置與區域。為了達到運算量的節省，我們以邊緣資訊(edge information)作為判斷移動物體區域與範圍的主要根據。其流程圖如 Fig.3-11 所示：

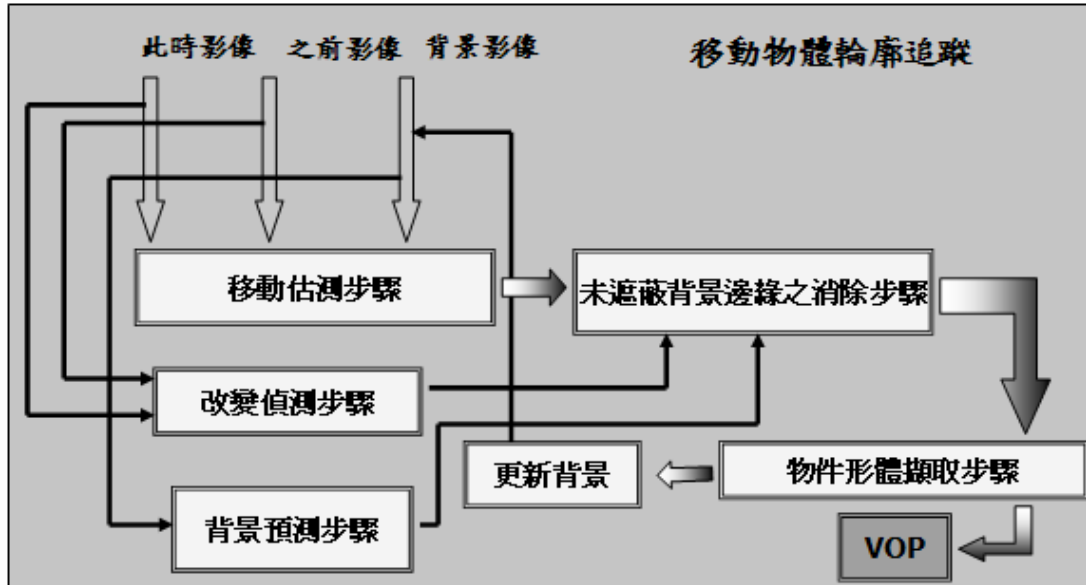


Fig.3-11 移動物體追蹤流程圖

首先是移動估測步驟，此部份主要是建立移動物件的完整輪廓。其方式是利用此時的影像與之前的影像所得到的差異值，與此時的影像與建立出來的背景影像所得到的差異值，分別帶入輪廓偵測器(edge detector)中來找出完整的物件輪廓。我們選擇肯尼輪廓偵測器(Canny edge detector)[15][16]作為尋找輪廓的工具。其表示式如 Eq.3-6 所示：

$$\begin{aligned}
 ME &= [(\phi(|f_i - f_j|)) \cup (\phi(|f_i - BG|))] \\
 &= [(\theta(\nabla G * |f_i - f_j|)) \cup (\theta(\nabla G * |f_i - BG|))] \quad (3-6)
 \end{aligned}$$

將兩種不同輸入所找到的輪廓取其聯集之後的結果後即可得到完整的物件輪廓。

由於上述步驟中，仍然會有些許的背景輪廓(background edge)因為輪廓偵測器的判斷錯誤而被誤判成物件輪廓，而這些錯誤的資訊對於接下來的物件形體擷取(object shape extraction)步驟有很大的影響。若不加以消除，這些背景輪廓亦會造成未遮蔽背景區域的產生。因此我們利用改變偵測與背景預測兩種方式將誤判的背景輪廓清除；當背景是複雜的狀況下，改變偵測方式可以很有效地消除被誤判的背景輪廓。而針對背景較單純且背景與移動物體亮度(luminance)相近的

情況下，使用背景預測方式可以較有效消除未遮蔽背景輪廓。

利用上述提到的方式可以將完整的物件輪廓找出並且將會造成未遮蔽背景現象的錯誤輪廓消除，最後將我們所要的動態物件完整擷取出來。我們首先利用輪廓的區域資訊先概略將物件在場景中的位置框出，接著再利用掃描的方式將其物件輪廓內部掃描出物件的詳細區域，最後再將其灰階值對應到相對的像素位置中，即可完整挖出移動的視訊物件平面(VOP)。

由於擷取出來的視訊物件平面，其他的區域即是此時影像中的背景資訊，因此我們可以將此類資訊重新作為移動估測步驟時輸入的背景影像資訊，以作為下一影像擷取動作的參考資料，如此將可建立一循環式移動物件偵測機制。



第四章 以物件輪廓為基礎之 MPEG-4 編/解碼架構

我們已經提出了一種新的動態物件擷取演算法，接下來的步驟就是將其與現今的 MPEG-4 標準協定做結合。在現今所存在的多種 MPEG-4 編/解碼器中，我們選擇 XviD[21][22]編/解碼器作為結合的平台。由於在我們的擷取演算法中，背景的內容並不會變動，針對此種特性，在所提出的新編解碼架構中，可以使其減少背景資訊上的傳輸資料量。在 4-1 中，我們將介紹所提出的編碼器(encoder)架構，並且說明如何減少在已編碼資料(encoded data)中屬於背景資訊所需的資料量。在 4-2 中，我們將逐一介紹在所提出的解碼器(decoder)架構中，針對未帶有背景資訊的框架(frame)，如何將其填補。在 4-3 中，我們將針對在背景填補這方面所遇到的問題與解決方式做詳細的介紹。

4-1 編碼器架構

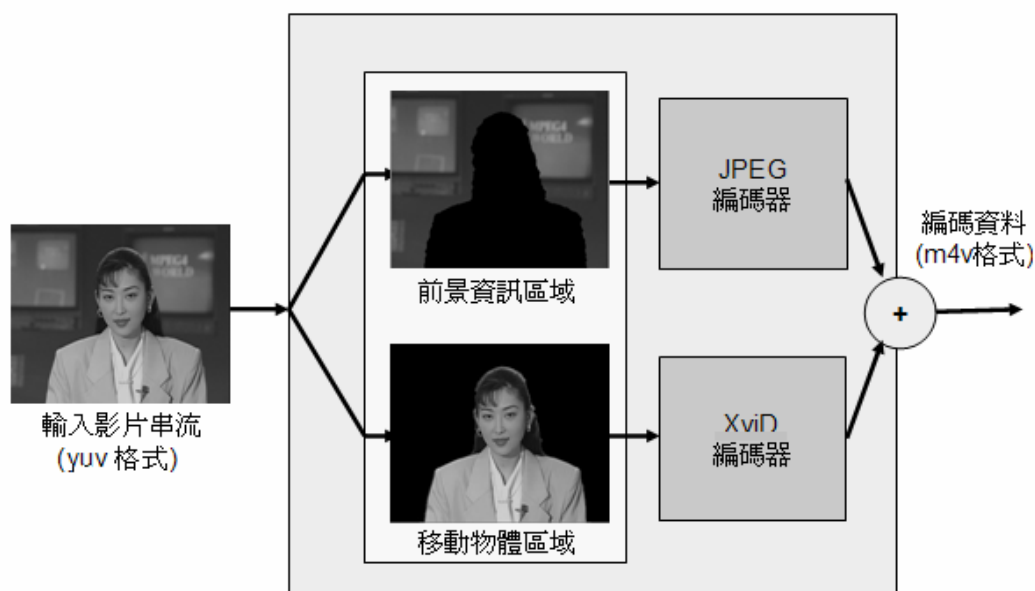


Fig.4-1 提出的 MPEG-4 編碼器架構

我們所提出的編碼器架構圖如 Fig.4-1 所示。首先，輸入的影片藉由提出的擷取演算法先行分割為移動的物件區域與不動的背景區域。接著，我們使用 XviD 與 JPEG[23]編碼器分別針對移動的物件區域與背景做編碼的動作。由於背景資訊內容並不改變，所以我們只需要在編碼過程剛開始時傳送一次背景資訊即可。這樣的編碼方式可以使編碼過的資料內只存有移動物體資訊與少量的背景資訊。

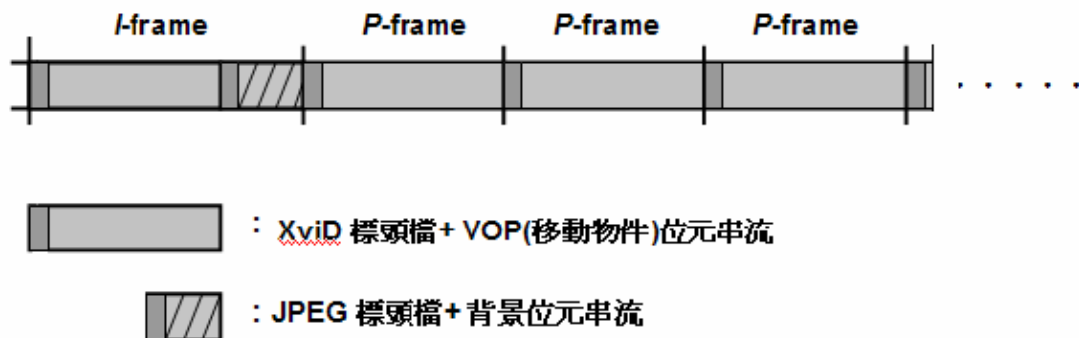


Fig.4-2 所提出編碼器位元串流架構圖

在 MPEG-4 的編碼架構中，若現在的影像內容與上一張影像變化量達到一定程度時，此張影像則被編碼為 *I*-frame(又稱為 *Key-frame*)。而在 *I*-frame 之間的影像則稱做 *P*-frame。*I*-frame 儲存整張影像資訊，而 *P*-frame 只紀錄與上一張影像的差異量。在我們所提出的編碼位元串流(bit-stream)架構中(如圖 Fig.4-2 所示)，背景資訊只包含在 *I*-frame 中，其餘的 *P*-frame 只包含移動物體資訊。因此，*I*-frame 位元串流包含了移動物件位元串流資訊與背景位元串流資訊，而 *P*-frame 只包含了移動物件位元串流資訊。

4-2 解碼器架構

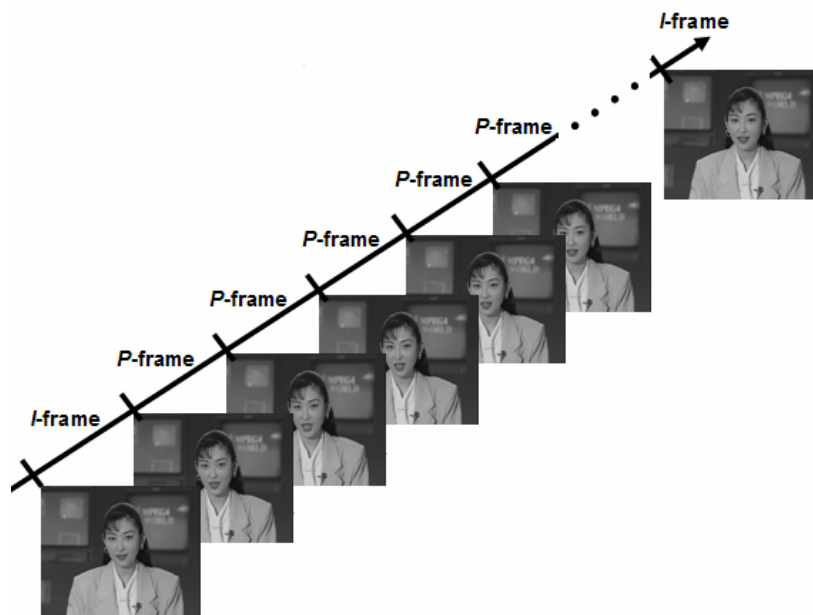
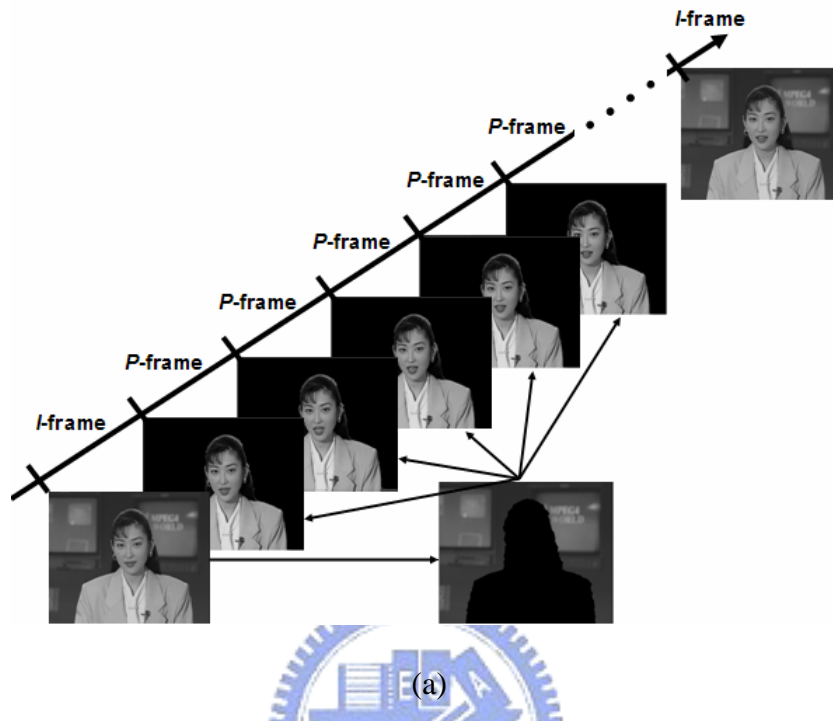


Fig.4-3 (a)解碼器端的背景填補架構；(b)經由背景填補架構所得到的完整解碼影像。

在所提出的解碼器架構中，我們根據不同的標頭檔(header)內容(如圖 Fig.4-2 所示)來使用不同的解碼器。若是屬於 XviD 標頭檔，則使用 XviD 解碼器將移動物件位元串流資訊解碼；反之若是 JPEG 標頭檔，則使用 JPEG 解碼器將背景位元串流資訊解碼。由於在已編碼檔案中只有 *I*-frame 影像含有背景資訊，因此在解碼過程中我們必須要將其餘的 *P*-frame 影像內容增加背景的資訊，以建立出完整的場景。其解決的方式是在解碼的過程中，若是解碼到 *I*-frame 影像時，我們將其中的背景資訊填補到接下來的 *P*-frame 影像內容中，如此 *P*-frame 影像將擁有背景資訊(如 Fig.4-3(a)所示)。其表示式可由 Eq.4-1 表示之。

$$F_{P-frame}(x, y, t_n) = M_{P-frame}(x, y, t_n) + B_{I-frame}(x, y, t_0) \quad \text{with } t_n > t_0 \quad (4-1)$$

其中 $F_{P-frame}(x, y, t_n)$ 代表第 n 個 *P*-frame 的解碼影像內容， $M_{P-frame}(x, y, t_n)$ 代表第 n 個 *P*-frame 的移動物體影像，而 $B_{I-frame}(x, y, t_0)$ 代表 *I*-frame 的背景影像，即解碼器架構中背景填補步驟時唯一填補的背景資訊。經由此步驟，我們即可得到完整的解碼資料(如 Fig.4-3(b)所示)。



4-3 背景填補架構所面對的問題及其解決方式

如之前所述，利用此方式可以在編碼過程中有效地減少所需傳送背景資訊。然而，由於背景資訊只有傳送一次，因此在解碼架構中，由於物件的移動，在背景填補步驟時會造成移動物件輪廓上的錯誤。產生的錯誤主要分成有重疊區域(overlap region)以及無資訊區域(non-information region)。其產生的原因與解決方式將分別在4-3.1與4-3.2中說明。

4-3-1 重疊區域與無資訊區域的產生

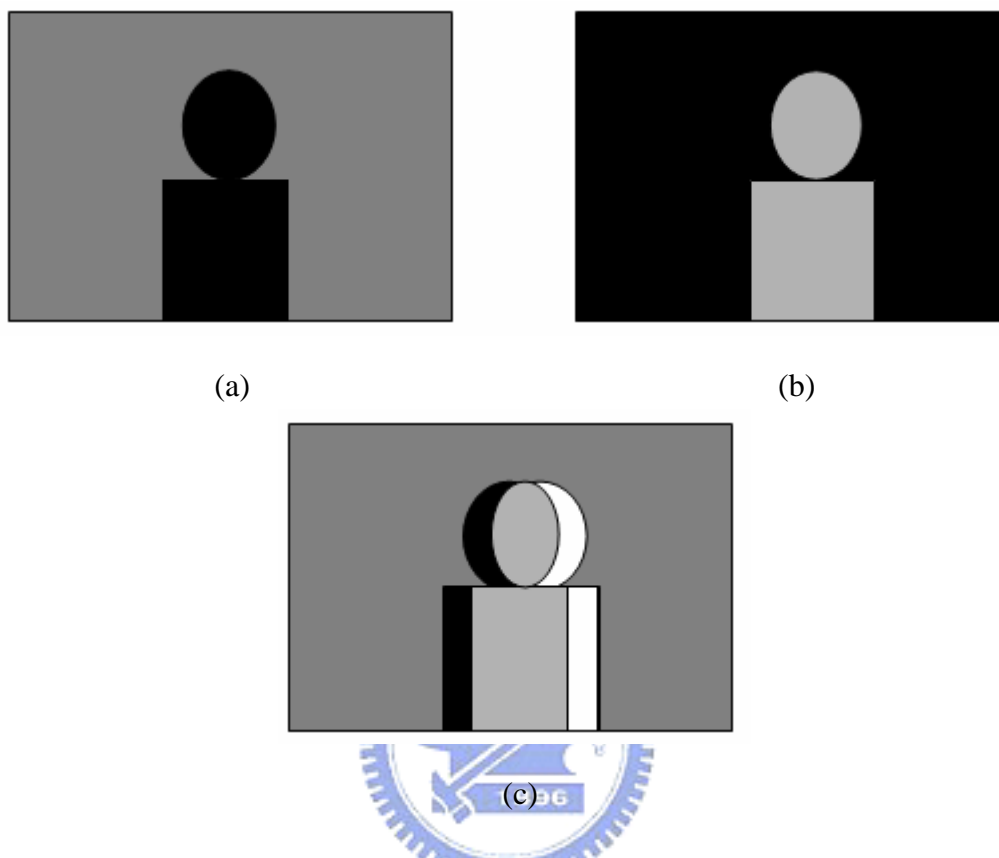


Fig.4-4 (a) *I*-frame 中所包含的背景資訊 (b) *P*-frame 中所擁有的移動物體資訊 (c)背景填補過程後產生的重疊區域與無資訊區域現象

我們以Fig.4-4簡單說明重疊區域與無資訊區域的產生原因。假設Fig.4-4(a)為*I*-frame所帶有的背景資訊；其中黑色的區域由於是屬於移動物件區域，因此其像素位置並無包含相對應的灰階值資訊。而Fig.4-4(b)則是代表之後*P*-frame所擁有的移動物體資訊；相反地，此時將只有移動物件區域所包括的像素位置含有相對應的灰階值資訊。擁有以上資訊之後，接下來則是針對*P*-frame內容執行背景填補動作。但由於移動物體在此時已經有稍許移動，若我們只是單純地將之前所得到的背景資訊加入到現有的前景移動物體資訊中，則會造成部份移動物體位置與之前的背景資訊位置有重疊的現象產生(如Fig.4-4(c)中的白色區域)，而這些區

域即是重疊區域。而另一方面，由於物件的移動與背景資訊並未補償，所以亦會造成部份缺少背景資訊與移動物體資訊的區域產生(如Fig.4-4(c)中的黑色區域)，這些區域即是無資訊區域。Fig.4-5即是上述現象的真實情形。



Fig.4-5 重疊區域(白色區域)與無資訊區域(黑色區域)現象

4-3-2 前景優先性與背景補償(Foreground Priority and Background Compensation)

為了解決上述的問題，我們分別針對其個別產生的原因，提出相對應的解決方式。針對重疊區域我們在解碼器架構部分提出前景優先性的方式能修正重疊區域的問題；另外針對無資訊區域的問題我們可在編碼器部分利用背景補償方式將此問題消除。其個別解決方式將在以下作詳細地說明：

(1)前景優先性:

重疊區域由於是因為當移動物體位置與背景區域位置重疊時，若以灰階影像來說，其兩者灰階值相加之後會增加，亦有很大的機會超過最大灰階值255，於

是會有灰階值改變或是白色區域產生。針對此種現象，我們可以從影像前景與背景的概念，在解碼步驟中的背景填補方式上做些許的改變。

由於在一般場景中，移動物件往往是屬於前景的部份，因此，當背景資訊與移動體資訊在相同位置上重疊時，針對此位置，我們只要呈現移動物體資訊即可消除重複區域問題的產生。我們可在解碼架構的背景填補步驟中建立以上的判斷方法，其表示式可由Eq.4-2與Eq.4-3表示之：

$$OI_{P-frame}(x, y, t_n) = MI_{P-frame}(x, y, t_n) \cap BI_{I-frame}(x, y, t_0) \quad \text{with } t_n > t_0 \quad (4-2)$$

$$F_{P-frame}^{moving}(x, y, t_n) = \begin{cases} M_{P-frame}(x, y, t_n), & \text{if } MI_{P-frame}(x, y, t_n) \cup OI_{P-frame}(x, y, t_n) \\ 0, & \text{otherwise} \end{cases} \quad (4-3)$$

其中 $OI_{P-frame}(x, y, t_n)$ 與 $MI_{P-frame}(x, y, t_n)$ 分別代表第 n 個 P -frame 中之重複區域與移動物體區域像素位置。而 $BI_{I-frame}(x, y, t_0)$ 則代表在 I -frame 中所擷取出的背景區域像素位置；而 $F_{P-frame}^{moving}(x, y, t_n)$ 則代表在第 n 個解碼影像中，屬於移動物體區域內的每個像素灰階值。依照上述步驟我們可以順利消除此問題。其結果如 Fig.4-6 所示。



Fig.4-6 將 Fig.4-5 經由前景優先性步驟所得到的解碼影像

(2) 背景補償

另外針對無資訊區域的狀況，由於此區域是當物體移動到別處之後，原本物體所在的部份區域在填補的背景資訊中並沒有其相對應的資訊存在，因此這些區域在此時應屬於背景區域的一部份。因此為了能夠依然在只傳送一次背景資訊前提之下，在編碼架構部分中，對於每個要傳送的*P*-frame內容，原本只擁有變動物體資訊，可以加入部分的背景補償資訊，如此則可以將這類的問題解決。

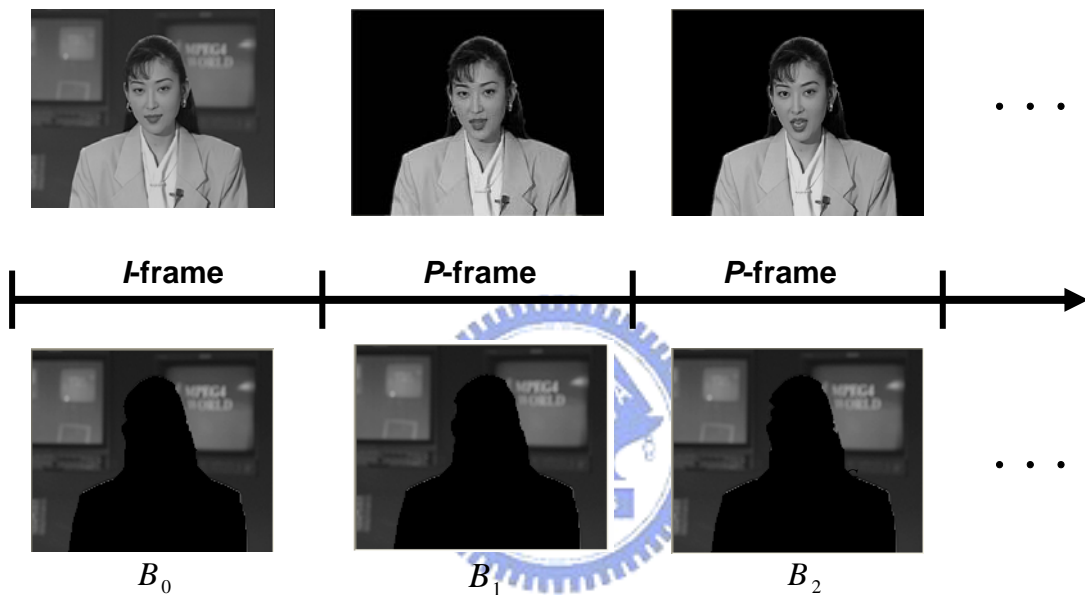


Fig.4-7 (a)編碼的影像資訊內容；(b)經由移動物件擷取演算法所得到的每張影像背景資訊內容。

在所提出的編碼器架構中，我們利用移動物件擷取演算法將每一張要編碼的影像分割成移動物體與不動背景，之後將第一張影像(*I*-frame)中完整的變動物件與背景資訊編碼，之後的影像(*P*-frame)則只編碼變動物件資訊(如Fig.4-7(a)所示)。因此整個編碼資料中只有第一張的背景資訊(如Fig.5-7中的 B_0)，即背景填補步驟中唯一使用的背景資訊。但無資訊區域的產生是因為物體的變動與填補的背景資訊無法相符合所造成的缺洞。針對這些區域，我們可以將從第二張影像後的背景資訊分別與第一張的背景資訊比較，其方式可由Eq.4-4作為表示：

$$NOI_n(x, y, t_n) = \begin{cases} B_n(x, y, t_n) & , \text{if } B_n I(x, y, t_n) = 1 \ \& \ B_0 I(x, y, t_0) = 0 \\ 0 & , \text{otherwise} \end{cases} \quad (4-4)$$

其中 $NOI_n(x, y, t_n)$ 代表的是在第 n 個 P -frame 中屬於無資訊區域的位置及其所對應的每個像素灰階值； $B_n(x, y, t_n)$ 則是代表第 n 個 P -frame 中屬於背景區域的位置及其所對應的每個像素灰階值； $B_n I(x, y, t_n)$ 與 $B_0 I(x, y, t_0)$ 分別代表第 n 個 P -frame 與唯一的 I -frame 中含有背景資訊的位置。

我們的背景補償概念是利用第一張編碼影像(I -frame)中的背景資訊，作為唯一背景填補動作的參考資訊 $B_0(x, y, t_0)$ ，與之後編碼影像(P -frame)擷取出來的背景資訊 $B_n(x, y, t_n)$ ，兩者互相比較；若某一像素在 P -frame 中是屬於背景區域，即 $B_n I(x, y, t_n) = 1$ ，但在最初的 I -frame 中卻不屬於背景區域，即 $B_0 I(x, y, t_0) = 0$ ，如此可確定此像素原屬於移動物體區域但因為物體變動此時已屬於背景區域，造成無資訊區域的產生。因此我們針對有以上現象的區域，在編碼器架構裡，先將其背景資訊補償到 P -frame 中，使得 P -frame 中除了移動物體資訊外另外帶有少許的背景補償資訊。如此在解碼器中，經過背景填補過後則能組成一完整畫面。其結果如 Fig.4-8 所示：



Fig.4-8 將 Fig.4-5 經由前景優先性與背景補償步驟所得到的解碼影像

第五章 實驗結果

實驗結果主要分兩大部分：第一部分是針對所提出的動態物件擷取技術和改進的未遮蔽背景區域消除方法在效果上的呈現與比較。第二部份則是將所提出的MPEG-4編/解碼架構與傳統的MPEG-4架構，針對編碼壓縮後的檔案大小與解碼畫面做分別的比較與討論。

5-1 動態物件擷取演算法

我們主要針對所改進的未遮蔽背景區域消除方法與之前所提出的方式做擷取效果上的比較。所使用的測試影片為MPEG-4標準測試檔[24]，其檔案格式為YUV格式，大小為CIF(352x288)與QCIF(176x144)；測試平台為Visual C++.net，電腦作業系統與設備分別為Microsoft Window XP，AMD AthlonXP 2G系統。

在比較所提出的未遮蔽背景區域消除方法與效果時，有兩大主要探討的對象：第一是經由初始前景/背景建立步驟所得到出來的初始前景/背景分離區域，而第二則是利用上述步驟所得到的資訊，經由移動物件追蹤所擷取的移動物件平面(VOP)。我們將針對上述結果作為討論與比較的內容。

(1) Salesman sequence:

首先我們使用”Salesman”影片檔作為測試。在初始前景/背景分離方面，由於利用改變偵測來累積前景/背景資訊時前景人像不斷地在改變，尤其是在手的移動部份，因此未遮蔽背景現象就會明顯產生。利用之前所提出的消除方式只能參考第一次與最後一次改變偵測所分離出的移動物體資訊來消除此區域，當未遮蔽背景現象發生於其餘改變偵測的過程中時，此種方式將無法完全消除；如Fig.5-1 (a)在桌面部份所產生的誤判區域。使用我們改進的消除方式可以完全利用到改變偵測所得到的資訊，也就能順利將此現象消除。

接著是在經過移動物件追蹤步驟之後，所擷取出的VOP結果。由於使用改進過的演算法可以將較精確的前景/背景區域分離出來，之後將這些資訊帶入到移動估測步驟之中，則可以成為較好的移動物體追蹤參考資料；因此所擷取出來的VOP區域精確度會比使用之前的方式要來的更好(如Fig.5-2所示)。

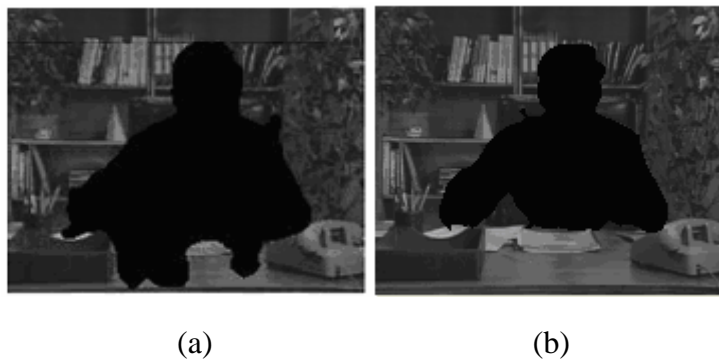


Fig.5-1 (a)使用先前之演算法所分離出的初始背景影像；

(b)使用改進之演算法所分離出的初始背景影像。



Fig.5-2 (a) 輸入的#5、#25、#45、和#65 影像；

(b) 使用先前之演算法所擷取出的#5、#25、#45、和#65 移動物件影像；

(c) 使用改進之演算法所擷取出的#5、#25、#45、和#65 移動物件影像。

(2) Weather sequence :

接下來我們使用”Weather”影片檔來作為測試。在初始前景/背景建立過程中，由於移動物體從左側移動到右側，之後頭與肩膀部份又稍微往左移動，造成所得到的初始移動物件區域在身體左側與頭部部份有未遮蔽背景現象產生，如Fig.3-3所示。若使用之前的未遮蔽背景消除方式將只能消除物件左側的未遮蔽背景區域；反之，若使用改進過的消除方式亦能將物體頭部與肩膀部份的未遮蔽背景區域消除，因此所建立出來的初始背景資訊將較為精確(如Fig.5-3所示)。接著將所得到的初始前景/背景區域資訊帶入到移動物件追蹤架構中；由於已建立較精確的區域資訊，在視訊物件平面(VOP)的擷取效果上也就比使用之前方法要來的準確。(如Fig.5-4所示)



Fig.5-3 (a)使用先前之演算法所分離出的初始背景影像；

(b)使用改進之演算法所分離出的初始背景影像。



(a)

(b)

(c)

Fig.5-4 (a) 輸入的#10、#30、#50、和#70 影像；

(b) 使用先前之演算法所擷取出的#10、#30、#50、和#70 移動物件影像；

(c) 使用改進之演算法所擷取出的#10、#30、#50、和#70 移動物件影像。

5-2 提出的 MPEG-4 編/解碼架構

我們將所提出的移動物件分割演算法加入到MPEG-4編/解碼流程中。在編碼器架構部分，由於只傳送一次背景資訊，如此可以減少重覆的編碼資訊，達到減少所需壓縮的資料量。在解碼器架構部分，使用背景填補的方式將背景資訊加入到每張影像之中，使其具有完整的移動物件與背景資訊。之後利用前景優先性與背景補償等後處理步驟分別解決重疊區域與無資訊區域等現象。

在此有兩個主要部份作為討論與比較的對象：第一部份是我們使用相同的影片測試檔，分別使用所提出的MPEG-4編/解碼架構與原本的架構對於編碼檔案壓縮率上的比較。第二部分針對解碼影片畫質作個別的比較與分析。在測試影片的選擇上，我們使用不同移動程度的移動物體作為測試的對象，藉以了解移動資訊的多寡對於編碼過的檔案容量所造成的影響；若壓縮之後的檔案大小愈小，代表壓縮率愈高，反之則代表壓縮率愈低。另外，由於我們的訴求是在編碼影片時只有第一張影像編碼為*I-frame*，之後的皆編碼為*P-frame*；為了符合上述條件，在與原先的編碼架構比較時，我們先將*I-frame interval*設定大於此測試影片所包含的frame總數，藉以在編碼時只有一次*I-frame*產生。

我們所使用的測試影片為YUV格式的MPEG-4標準測試檔，大小為CIF(352x288)與QCIF(176x144)；測試平台為Visual C++.net，電腦作業系統與設備分別為Microsoft Window XP，AMD AthlonXP 2G系統。

(1) Akiyo sequence :

我們首先從移動物體變動較小的影片開始測試。在”Akiyo”影片測試檔中，移動物體只有在頭部有稍微變動過，其餘部分幾乎沒有變動。由於此類狀況使用原本的編碼方式並不需要在P-frame中紀錄大量的可變動資訊，可以很有效地減少所需的編碼資料量，因此並沒有發揮出我們新編碼架構所擁有的優勢。

在解碼過後的影像畫質方面，經過背景填補步驟後，可以將背景資訊加入到缺少背景資訊的P-frame影像中。之後再利用前景優先性與背景補償等步驟將重疊區域與無背景區域等現象給消除。由實驗結果方面，除了在頭部與手軸部份有少許因背景填補所造成的不明顯接合痕跡之外，其餘填補狀況算是很完整。因此可知使用上述方式可以準確將完整的解碼影像資訊建立出來(如Fig5-5(b)所示)。

Table5-1 Akiyo sequence 壓縮率比較

	原來編碼方式	提出的編碼方式
輸入檔案大小	43.5MB	
影像大小	CIF(352x288)	
Frame 總數	300	
壓縮檔案大小	719KB	837KB
壓縮率	61.95	53.21



Fig.5-5 Akiyo sequence (a)使用原本編/解碼架構所得到的#15、#35、#55、#75 影像輸出；(b)使用提出的編/解碼架構所得到的#15、#35、#55、#75 影像輸出。

(2) Weather sequence :

接下來在”Weather”影片測試檔中，影片中移動物體大部分都維持在一定的移動範圍，因此變動的資訊並不大。但在影片最後階段物體由原先所在的位置迅速向
右方影像邊緣移動，此時若使用之前的編碼方式將會造成需要記錄大量的變動資
訊在每個編碼的*P*-frame中。而我們所使用的編碼方式是將每張移動物體的資訊
都儲存在每張*P*-frame中，因此物體的移動變化程度對我們的編碼方式影響並不
如以原先編碼方式要來的大。因此在此測試檔中兩者的壓縮率差距減少。

在解碼過後的影像畫質方面，除了在右側肩膀與背景接合部份有稍許的雜訊
產生之外，其餘背景填補部份算是非常完整，與使用原先編/解碼所得到的解碼
影像畫質相較不遠。

Table5-2 Weather sequence 壓縮率比較

	原來編碼方式	提出的編碼方式
輸入檔案大小	32.9MB	
影像大小	320x240	
Frame 總數	300	
壓縮檔案大小	1.2MB	1.24MB
壓縮率	27.41	26.53



Fig.5-6 Weather sequence (a)使用原本編/解碼架構所得到的#35、#55、#75、#95 影像輸出；(b)使用提出的編/解碼架構所得到的#35、#55、#75、#95 影像輸出。

(3) Salesman sequence :

在”Salesman”影片測試當中，雖然移動物體一直位於同一位置，但身體部份與手部分卻不斷地變動。對於這樣不斷變動的影像，使用原先的編碼架構就必須隨時紀錄大量的變動資訊在P-frame中，也就造成編碼壓縮過後的檔案大小加大。因此在此測試結果中，我們所提出的編碼架構其編碼壓縮之後的檔案會比其要來的小，即壓縮率提高。

在解碼影像畫質方面，由於此測試檔在移動物件擷取演算法中可以很精確地將移動物體區域與背景區域分離出來，因此在解碼過程中，背景補償與其後處理可以將背景精確填補到所對應的位置而不會有錯誤的狀況產生。與使用原先解碼架構所得到的解碼影像並無明顯差別。

Table5-3 Salesman sequence 壓縮率比較

	原來編碼方式	提出的編碼方式
輸入檔案大小	12MB	
影像大小	QCIF(176x144)	
Frame 總數	331	
壓縮檔案大小	543KB	508KB
壓縮率	22.63	24.18

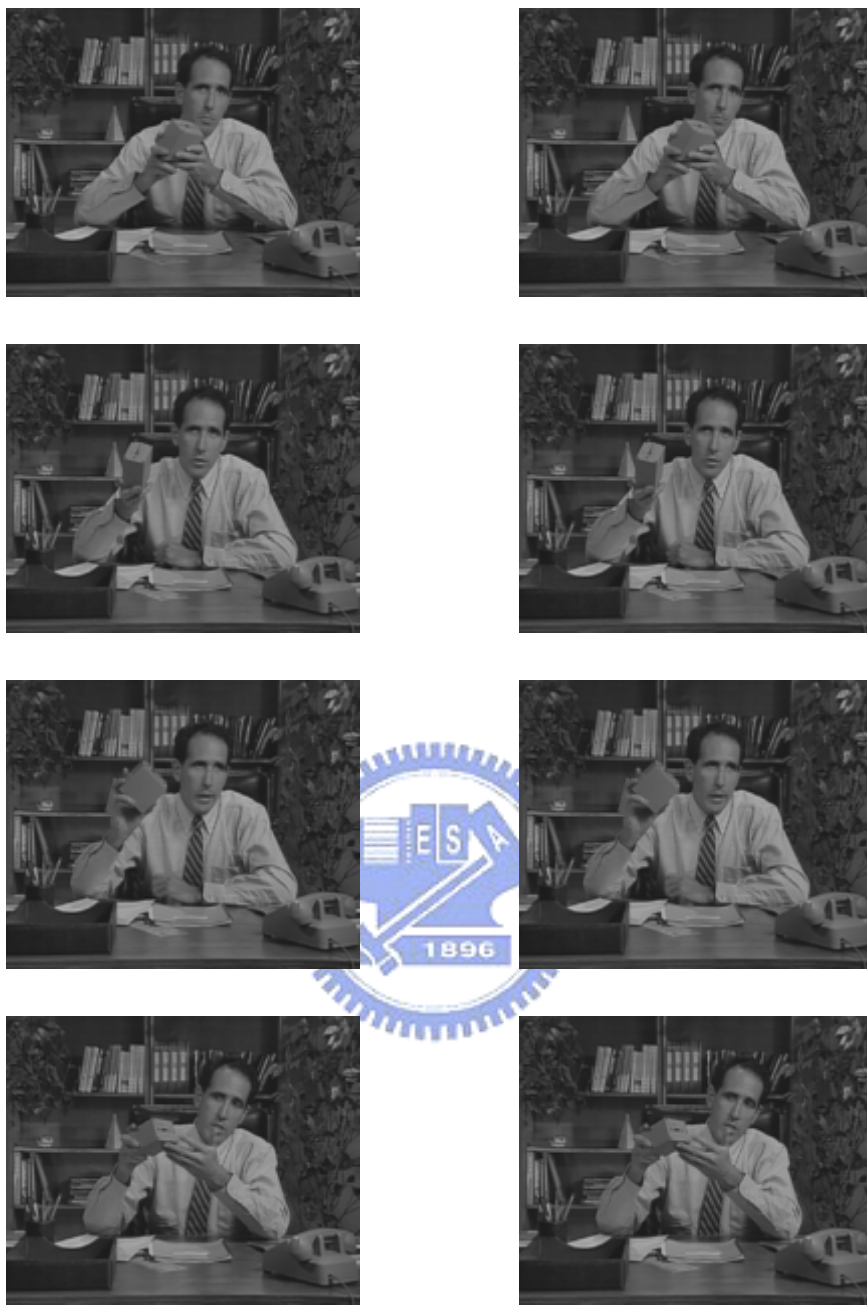


Fig.5-7 Salesman sequence (a)使用原本編/解碼架構所得到的#40、#60、#80、#100
 影像輸出；(b)使用提出的編/解碼架構所得到的#40、#60、#80、#100 影
 像輸出。

(4) Hall Monitor sequence :

最後是”Hall Monitor”影片測試檔。此影片有以下的特點:第一是由開始到結束移動物體皆不斷地變動位置且移動範圍很大;第二是此影片在一剛開始的部份並沒有出現移動物體,換句話說所傳送的編碼背景資訊是非常完整的一整張影像;第三是在影片中有原本屬於移動物體一部份的區域最後不變動了,因此造成背景資訊上的變化。我們分別針對以上特點,對於其編碼壓縮檔案與解碼影像作個別的討論。

在第一點方面,由於此影片中的移動物體在位置與形狀皆有劇烈的變化,因此若使用原本的編碼方式將紀錄大量的變化資訊於每張P-frame中,造成編碼後的檔案大小會比使用改進過的編碼方式要來的大,壓縮率也就會較低。而第二點方面,由於傳送的編碼背景資訊是很完整的整張影像,因此在解碼過程中,無資訊區域現象將不會產生,因此背景補償步驟在此並不需要。而在第三點方面,在影片中,原本屬於移動物體的公事包被置放在箱子上不動,因此被歸類為背景的一部分,造成背景資訊變動。但由於傳送給編/解碼器中的背景資訊是最初的背景資訊,此時背景並不包含公事包,因此若是用此資訊作為在解碼架構中背景填補步驟的背景資訊時,則會造成解碼過後影像中放置的公事包消失(如Fig.5-8(b) #167解碼影像中的紅色圓圈部分)。

Table5-4 Hall Monitor sequence壓縮率比較

	原來編碼方式	提出的編碼方式
輸入檔案大小	43.5MB	
影像大小	CIF(352x288)	
Frame 總數	300	
壓縮檔案大小	2.13MB	1.05MB
壓縮率	20.42	41.42



Fig.5-8 Hall Monitor sequence (a)使用原本編/解碼架構所得到的#27、#77、#117、
#167 影像輸出；(b)使用提出的編/解碼架構所得到的#27、#77、#117、#167
影像輸出。

第六章 結論與未來展望

在本篇論文中，我們首先提出一種針對影片內容中移動物件的擷取演算法，另外針對在擷取過程中會產生的未遮蔽背景區域現象(Uncovered-background region)，我們也提出一種可以完全利用到在初始前景/背景建立時所累積的資訊，之後並加以判斷與消除此現象的方法；與之前的技術相比在消除效果上有明顯地改進。

接下來我們將所提出的擷取演算法整合到現今的MPEG-4編/解碼架構中，使其建立以物件輪廓內容為基礎之編/解碼架構。由於影片中擁有大量不動的背景資訊，為了減少編碼資料中多餘且重複的資訊，我們提出的編/解碼架構將只需在影片開始部分編碼一次背景資訊，之後的編碼影像內容只包含移動物體資訊即可。之後在解碼部分再將同一背景資訊填補到只包含移動物體資訊的影像中，以建立完整的影像內容。之後利用前景優先性與背景補償方式分別解決背景填補過程中產生的重疊區域與無資訊區域等現象。由實驗結果得知，當影片中的移動物體變動程度愈大時，比較使用改進的編/解碼方式與原本的方式，在編碼資料壓縮效率上會愈明顯。另外在解碼架構方面，與使用原來的解碼方式比較，所得到的解碼影像並不會有太大的差別。

雖然我們所提出的移動物件擷取演算法已經能準確地分離出影片中的移動物體，但仍然有些許可以繼續改進的地方。首先是由於我們的演算法主要是針對不變動的背景情況下，因此攝影機必須維持不動；為了避免因為拍攝時的些許震動造成背景資訊的變化，可以在攝影機中加入數位影像穩定技術。另外我們的演算法主要是以影像灰階值的變化作為判斷的依據，但若是針對當移動物體與背景彼此的亮度差距不大的情況，可能會造成擷取效果變差；因此可以在擷取演算法中加入色彩資訊作為參考，以提高擷取時的準確率。

在另一方面，我們所提出的MPEG-4編/解碼架構仍有必須要改進的地方。由於在解碼架構中用來做背景填補的資訊是影片最初所建立的初始不動背景，因此當影片中背景內容有改變時，填補的背景資訊並不會跟著改變，造成解碼影像在背景內容改變區域有錯誤產生。為了解決這樣的狀況，我們必須在編/解碼架構中有另一背景內容更新機制來偵測背景內容的變化並加以更新，以消除所造成的錯誤現象。



參考文獻

- [1] N. Mukawa and H. Kuroda, "Uncovered background prediction in interframe coding," *IEEE Trans. on. Communication*, vol. COM-33, pp. 1227–1231, Nov. 1985.
- [2] Andre' Kaup and Til Aach, "Efficient prediction of uncovered background in interframe coding using spatial extraction" *Institute for Communication Engineering Aachen University of Technology, 52056 Aachen ,Germany*.
- [3] R. C. Gonzalez and R E. Woods, "Digital Image Processing", Addison-Wesley Publishing Company, 1992.
- [4] R. M. Haralick and L.G. Shapiro, "Survey: Image segmentation techniques" *Comput. Vis. Graph. Image Process*, vol. 29, pp. 100-132, 1985.
- [5] Y. -L. Chang and X. Li, "Adaptive image region-growing" *IEEE Trans. on Image processing*, vol.3, pp.868-872,1994.
- [6] S. A. Hijjatoleslami and J. Kittler, "Region growing: A new approach," *IEEE Trans. on Image Processing*, vol.7, pp.1079-1084,1998.
- [7] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations" *IEEE Trans. on Pattern Anal. Machine Intell*, vol.13, pp.583-598, 1991.
- [8] J. G. Choi et al, "A vop generation tool: automatic segmentation of moving objects in image sequences based on spatial temporal information," *IEEE Trans. on circuit and systems for video technology*, vol. 9, pp. 1216-1226, 1999.
- [9] R. Mech and M. Wollborn, "A noise robust method for 2D shape estimation of moving objects in video sequences considering a moving camera," *Signal Processing*, vol. 66, pp. 203–217, 1998.
- [10] T. Aach, A. Kaup, and R. Mester, "Statistical model-based change detection in moving video," *Signal Processing*, vol. 31, pp. 165–180, 1993.
- [11] Robert Thoma and Matthias Bierling. Motion Compensating Interpolation

Considering Covered and Uncovered Background. *Signal Processing: Image Communication* 1 (1989) 191-212.

- [12] A. Neri, S. Colonnese, G. Russo, and P. Talone, "Automatic moving object and background separation," *Signal Processing*, vol. 66, pp. 219-232, 1998.
- [13] R. Mech and M. Wollborn, "A noise robust method for 2D shape estimation of moving objects in video sequences considering a moving camera," *Signal Processing*, vol. 66, pp. 203-217, 1998.
- [14] Rosenfeld A., Pfaltz J.L., "Sequential operations in digital processing" in *JACM*, vol. 13, 1996, pp.471-494.
- [15] J. Canny, "computational approach to edge detection," *IEEE Transactions on Pattern Anal. Machine Intell*, vol. PAMI-8, pp. 679-698, 1986.
- [16] Canny Edge detector : <http://www.cee.hw.ac.uk/hipr/html/canny.html>
- [17] Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 mbit/s, ISO/IEC 11 172, 1993.
- [18] Information Technology—Generic Coding of Moving Pictures and Associated Audio Information, ISO/IEC 13 818, 1994.
- [19] MPEG-4 video verification model version 18.0, ISO/IEC JTC1/SC29/WG11 N3093, 2001
- [20] Overview of the MPEG-4Standard :
<http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm>
- [21] XviD Homepage: : <http://www.xvid.org>
- [22] XviD的歷史與特點 : http://easylearn.bhes.tpc.edu.tw/high/xvid_history.htm
- [23] JPEG Homepage : <http://www.jpeg.org>

- [24] MPEG-4 標準測試檔：(1) <http://media.xiph.org/video/derf>
(2) <http://www.cipr.rpi.edu/resource/sequences/sif.html>
(3) <http://trace.eas.asu.edu/yuv/>

