

逢 甲 大 學

資 訊 工 程 學 系 專 題 報 告

浮水印在 MPEG2 的應用

學生： 方柏文(四丙)
柯政鋒(四丙)

指導教授：李維斌 老師

中華民國九十三年五月

目 錄

圖表目錄	IV
摘要	V
第一章 導論	1
1.1 研究動機與目的	1
1.2 成果簡介	1
第二章 浮水印簡介	2
2.1 何謂浮水印	2
2.1.1 浮水印的特色	3
2.1.2 浮水印的分類	3
2.1.3 好的浮水印的要求	5
2.1.4 浮水印的主要用途	6
2.2 常見的浮水印技術	7

2.2.1	簡單系統法	7
2.2.2	頻域轉換(frequency transform) 法	8
2.2.3	展頻(Spread spectrum)	9
第三章	MPEG2 簡介	12
3.1	MPEG2 導論	12
3.2	MPEG2 壓縮及基本架構	14
3.3	MPEG-2 壓縮格式、編碼跟解碼	19
3.4	MPEG2 的應用	24
3.5	專題的應用	25
第四章	系統的實做	26
4.1	使用的工具	26
4.2	藏浮水印的基本構想	27

4.3	取浮水印的基本構想.....	29
4.4	實作中遇到的困難.....	29
4.5	採用的浮水印技術的簡介...	31
4.6	浮水印必要條件的比較.....	32
4.7	實驗中的一些其他做法和其缺點...	34
第五章	結論與心得.....	35
	參考資料.....	38
	附錄	
	圖表目錄	
圖 1-1	成果基本架構圖.....	2
圖 2-1	看的見的浮水印.....	4
圖 2-2	看不見的浮水印.....	4
圖 3-1	MPEG2 的基本架構.....	14

圖 3-2	MPEG2 中的一個 Group of picture·····	20
圖 3.3	壓縮流程圖·····	23
圖 4.1	Mpeg2decode·····	28
圖 4.2	Mpeg2encode·····	28
圖 4.3	Three-layer pyramid wavelet decomposition···	32

摘要

隨著電子化社會的來臨，大量的文字，繪畫與傳統媒體等均轉換成數位檔案作儲存，而數位檔案最大的優點就是傳輸方便與複製容易，這可以說是繼畢昇之刻版印刷以來資料傳播最劃時代的改變。

資訊科技的快速發展，電子數位產品不斷的出現，再加上透過網際網路無遠弗屆的便利性，使得所有數位資料得以快速簡便的複製與傳遞。而數位複製的技術又使得所有的數位圖像能大量複

製，如此則容易導致使用者有意無意中去觸犯到智慧財產權擁有者的權利，爲了能保護原著作者的權利，故在數位圖檔裡頭加入宣告擁有者資訊也就相對顯得重要。

現在的資訊時代是一個多媒體的時代,影像壓縮技術就相對的重要,DVD 是未來影像格式的主流,而 MPEG2 則是他的壓縮技術,所以其地位不得不重要,所以研究 MPEG2 的浮水印技術,可望有效杜絕盜版的暢行。



第一章 導論

1.1 研究動機與目的

現在社會上盜版的行爲四處可見,尤其在台灣盜版根本是司空見慣,所以根據這個現象,我們就浮現了一個靈感,何不來研究防止盜版的浮水印技術,但是我們去了解後,發覺浮水印應用的地方如此的多,要如何去選擇呢?我們就想到未來影像的主流---DVD,覺得這種一定很值得去研究,所以就決定做浮水印在 MPEG2 上的應用.

1.2 成果簡介

整個系統分爲兩部分:藏浮水印和取浮水印.

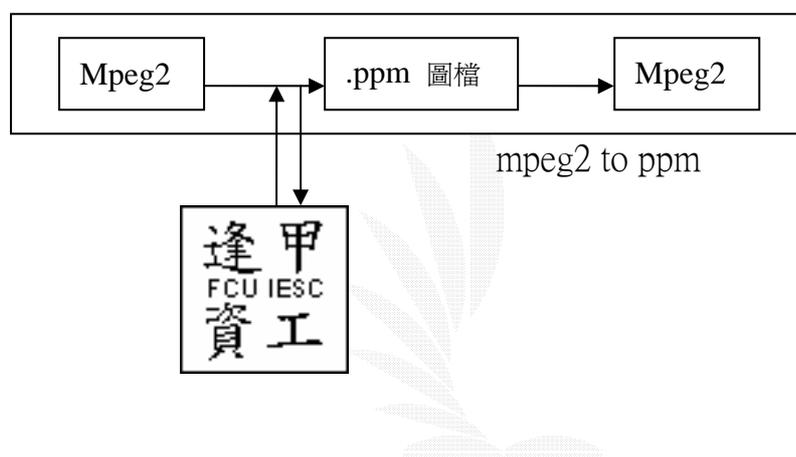
第一部分:

可將一張灰階圖片藏進一個 mpeg2 格式影片中,不會嚴重影響畫質,使加完浮水印的影片跟原始影片沒有什麼兩樣.

第二部分:

從影片中取出此圖片,而且從取出的圖片可以辨識出是否
為所藏的圖片.

簡單的架構如下:



第二章 浮水印簡介

2.1 何謂浮水印

在現在這個資訊爆炸的時代,所有的資料都轉成數位的東西來傳送閱覽,這使得資料容易被傳閱和修改,可是也形成盜版的問題源源不絕的發生,所以爲了保障原著者的權利,在產品中加入可辨識的圖案是

必須也必要的。而浮水印就是爲了這個目的做出來的一種技術。

2.1.1 浮水印的特色

1. 以廣義來說，就是「數位/類比簽章」，以和原來的資料作區別。
2. 主要用於版權和身份確認。
3. 版權擁有者嵌入浮水印，並能夠從資料中偵測（detect）並解出（decode）浮水印，用來作為版權控制，而浮水印便是「版權所有」的證明。
4. 只有加上浮水印的版權擁有者，有解出浮水印的方法。

2.1.2 浮水印的分類

第一類是看得見的浮水印，以直接在數位媒體上能看見浮水印，其缺點是這個浮水印已破壞了數位媒體原有的品質，但是使用者在看見這種浮水印時也就不敢隨意非法使用這個數位媒體了。如：



圖 2-1 看的見的浮水印

第二類是看不見的浮水印，強調植入浮水印後的數位媒體，無

法直接以肉眼看見，植入浮水印後的數位媒體和原來的數位媒體看

起來並沒有任何不同，因此浮水印具備了保密性。如：



圖 2-2 看不見的浮水印

這次我們要做的就是第二類型的。

2.1.3 好的浮水印的要求

1.隱藏性(Imperceptibility):

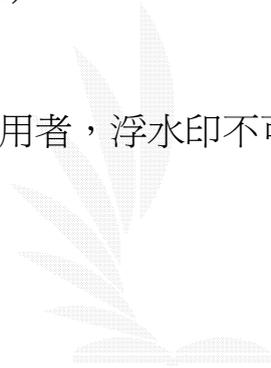
加入的浮水印，需肉眼看不見或聽不見難以被人察覺，能對於原影像或聲音品質的影響應減至最低。

2.不可去除性(No removable) :

對於未經過授權的非法使用者，浮水印不可輕易被剪下或移除。

3.強健性(Robustness) :

在經過一些影像處理的動作或壓縮後，浮水印仍能存在於影像資料中維持其功能或浮水印能忍受各種數位處理與攻擊，至少在其被破壞前原始影像已嚴重失真。



4.可輕易解讀性(Decode ability) :

對於經過授權的合法使用者，必須要能輕易地將不可見的浮水印抽取出來，不需再經過和原始影像比對。

5.安全性(Security) :

即使使用者知道加入的浮水印的程序也無法讓未經授權者移除所加入的浮水印。

而這些要求間,有時會互相衝突,有所矛盾,如隱藏性(Imperceptibility)和強健性(Robustness)兩者之間就有矛盾,為符合強健性我們必須把浮水印加在影像較獨特的地方,可是這樣去不容易滿足隱藏性。

2.1.4 浮水印的主要用途

- 1.當作保密用途
- 2.傳遞秘密資訊

3.證明媒體來源出處為何?合法性的鑑別

4.防止被盜用

5.著作財產權的保護

2.2 常見的浮水印技術

2.2.1 簡單系統法

1.最低位元(Least Significant Bit)法 :加入浮水印的位元設在圖片區塊位元的最低位元,是不易被人眼所觀察出來的。例如: 灰色圖素的值是 90 轉換結果值將為 01011010 選擇圖素的 Least significant bit 插入 watermark 一個位元轉換成新的資料。但是這些系統都易被敵人以一些方法來破壞浮水印資料。例如利用過濾處理都會改變一些最低位元的值。

2.資料多餘(Redundancy)法 : 它只是將一個浮水印重覆加入好幾次,

但對大部份 檔案都會經過壓縮處理,這對資料多餘法會有相當的破壞。

2.2.2 頻域轉換(frequency transform)法

1. Discrete cosine transform (不連續餘弦轉換)

不連續餘弦轉換是屬於區段基礎的處理，自從大部分影像及圖像壓縮藉由 JPEG， MPEG 或 H.261/263 標準來處理，這些方法是利用 DCT 的基礎技術，使用 DCT 來作電子浮水印處理是相當的合宜，當我們利用 DCT 轉換作浮水印嵌入，一個圖檔會被分割成兩個變動大小的同質性區塊，使用 DCT 作浮印處理的好處是具有非常強韌性的資料而且不會降低視覺的品質。

2. Wavelet transform(小波轉換)

在應用數位小波轉換之後原始資料將被轉換成數個波段，這些波段

包括低-低波段，低-高波段，高-低波段，高-高波段，利用小波轉換來作浮印處理我們可以得到的好處是強化 MPEG 的程式及重作和多重嵌入浮印處理的能力。

這是我們這次所運用的技術。

2.2.3 展頻(Spread spectrum)

目前在電信通訊及人造衛星系統都使用展頻技術來增加資料安全的能力及消除雜訊人為干擾，自從資料能夠應用延展技術相對的資料復原比率也相對增加，舉例：我們傳送一個” 1” 如果我們接收到的也是一個” 1” ，我們可以說這傳送是正確的，如果將” 1” 延展 3 次變成” 111” ，這時接收到的可能是” 011” ，” 101” ，” 110” ，” 111” ，藉由多數或最大可能性的原理我們得到原來正確的資料也是” 1” ，最近幾年來展頻技術

已經應用在浮水印處理技術上，無論如何最常使用的一個方法是利用直接展頻。

1.直接展頻(Direct-Sequence Spread)：浮水印信號係藉由相位

調變原信號(載波信號)與送收兩方都知道的虛擬號碼順序所隱藏。

2.區塊展頻技術(Block spread spectrum)

區塊展頻技術是應用在區塊及周期性的遞歸處理上，如果利用 least significant bit 將 logo X 嵌入在圖檔 Y 裡面，即將 Y 分割成許多區塊再將 logo X 分散的隱藏在各區塊間，在這裡，圖檔將被區分為多少個區塊呢?這個答案將根據不同的 logo 和不同的圖檔特性來決定。

3.重複展頻(Duplication spreading)

重複展頻的基礎只是應用反覆的嵌入處理，將 logo 展開的二位元資料流嵌入圖檔的每一個圖素裡，重複執行，直到圖檔的最後一個圖

素。

應用重複展頻技術可提高資料還原及安全性，此方法僅較優於直接

序列展頻。



第三章 MPEG2 簡介

3.1 MPEG2 導論

MPEG 是動態視訊專家小組 (Moving Picture Experts Group) 的簡稱。MPEG 又可分成 MPEG1, MPEG2, MPEG3, MPEG4。因為 MPEG3 隨著 MPEG2 HDTV 的發展已經被 MPEG2 所取代,而 MPEG4 又是以視訊會議為主要應用,所以我們專題將是以 MPEG2 為重心。在 1988 年 ISO/IEC(International Organization for Standardization/International Electrotechnical Commission)對動態視訊與語音壓縮標準提出的壓縮準則,如今已廣泛地被使用在 CD-ROM 的視訊應用。說仔細一點就是一個訂定 bitstream 格式的標準。那什麼是 bitstream 呢? Bitstream 就是將視訊及音訊記錄到介質(比如說錄影機, DVD-digital video disk, 電纜或無線傳輸)上一連串的 0 與 1。為了使這一連串的 0 與 1 有意義,我們就必須對它編碼,其實

bitstream 一旦制定之後，可以說連解碼器的架構也差不多訂定了。

可是只要編碼器產生的 bitstream 標準符合 MPEG 的標準，你要怎樣

壓縮產生是各個廠商的事。然而當初 MPEG-1 制定時，其解像度只

有 $350 * 240 * 30$ ，以此解像程度每當用於點腦螢幕上播放時，由於

電腦螢幕的要求過高，所以顯示出的效果都不夠讓使用者所滿意，

故其並不適合做視訊傳播的應用，因此 ISO/IEC 在 1990 年又開始制

定 MPEG-2 的視訊壓縮標準，增加了視訊應用範圍的彈性，其中包

含了支援不同的影像的解析度、增強畫面比率化的計算、加強時間

和空間的資料壓縮性能，最大的改善在於提高數位率和使用調整位

元率(VBR) 的技術，主要的應用包含了數位電視視訊的傳播、高畫

質數位電視(HDTV)、及數位儲存媒體的應用，例如 DVD (Digital

Versatile Disk)。而 MPEG-2 壓縮標準最後於 1995 年完成。然而視訊

壓縮有很多不同的壓縮標準，適用於不同的應用範圍，MPEG 算是

最普遍地被應用於數位視訊服務及廣泛地被工業界接受的一個視訊壓縮標準。

3.2 MPEG2 壓縮及基本架構

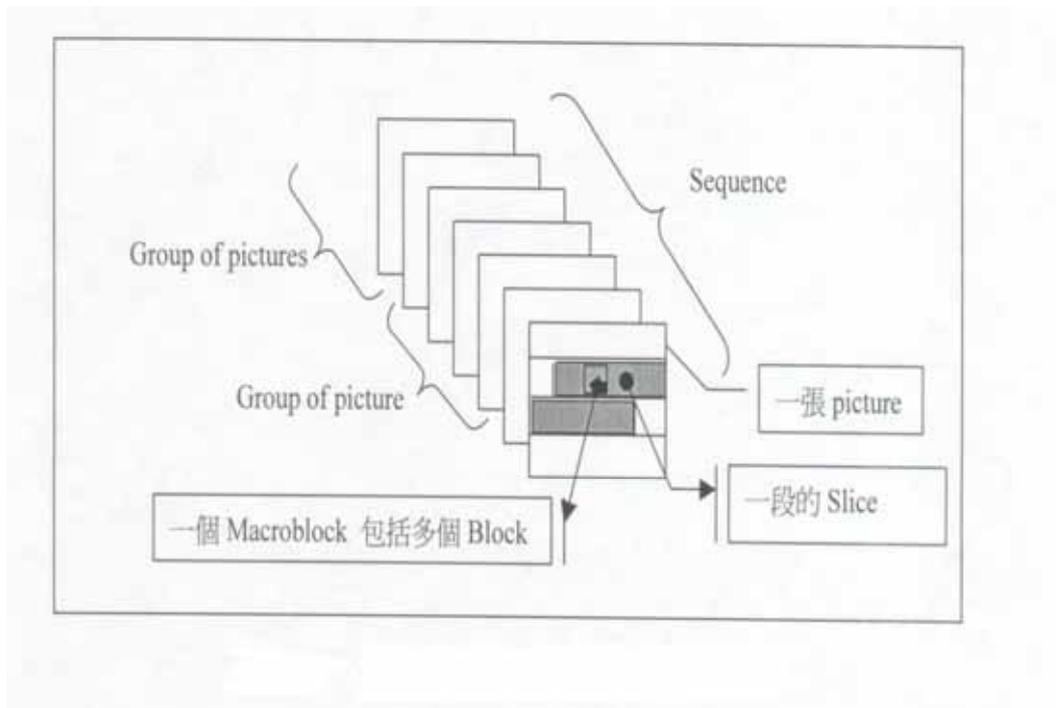


圖 3-1 MPEG2 的基本架構

基本架構分為以下這六大部份：

1. 視訊序列 (Video sequence) :

由數個影像或影像群組所組成，然而一部影片可以只由一個視訊序列組成，也可以由數個視訊序列所組成。

2. 影像群組 (Group of pictures : GOP) :

由一張 I 畫面及數張 P 及 B 畫面所組成。然而一 GOP 也是視訊序列隨機存取的基本單位。

3. 影像 (Picture) :

由數個片段 (Slice) 所組成，為最主要的編碼單位，主要有三種影像編碼的型態 I、P、B。

4. 片段 (Slice) :

由數個大區塊 (Macroblock) 所組成，主要將 Picture 作水平且固定



單位的切割，描述了在影像中的垂直位置，是訊號同步及錯誤控制的最小單位。

5. 大區塊 (Macroblock) :

由四個亮度信號 (Luminance) 區塊 (Block) 及數個色彩 (Chrominance) 信號區塊所組成，例如：4:4:4、4:2:2、4:2:0。然而色彩信號區塊的數目是依據影像數位化的取樣格式所決定。大區塊是移動估測及移動補償的基本單位，描述了編碼型態、移動向量及在片段中距離第一個大區塊的相對位址。

6. 區塊 (Block) :

包含量化後的畫面資料，由 8×8 的像素所組成。MPEG 定義了三種畫面壓縮模式 I 畫面 (Intra frame)、B 畫面 (Bi-directional frame) 跟 P 畫面 (Predicted frame)。

那以下一一來介紹 I、P、B

(1) I 畫面 (Intra-coded pictures) : 不需要考慮與其他畫面之間的關係，

其所儲存的是一張完整的畫面，所以在解碼時可以做為解碼的

切入點，供隨意擷取開始位置，不需參考其他畫面的資料，可

以單獨進行解碼。且 I 畫面是一視訊序列或一畫面群組的第一

張，隨後在畫面群組裡的 P 畫面與 B 畫面都會參考到它的資

料，所以在網路傳輸時需要特別保護其畫面資料的遺失，以免

造成隨後畫面影像品質的損害。由於編碼獨立，不須參考其他

畫面，享受消除時間軸上多餘性之好處，因此壓縮率較差。

(2) P 畫面 (Predictive coded pictures) :

P 畫面是以前面的 I 畫面或 P 畫面作為參考圖像，而參考的位置就

是以移動估測所產生的移動向量來表示，若找不到最適合的大區塊

時，則使用 Intra 模式編碼。P 畫面是由數個 Intra 模式編碼與預測模式編碼的大區塊（Predictive coded macroblocks）所組成。由於參考前一 I 畫面或 P 畫面且又以動態補償方式預測編碼，其編碼效率較高，也就是說畫面中有重複的部份就不要儲存，只儲存不一樣的地方。

(3) B 畫面（Bidirectionally predicted pictures）：

B 畫面的原理跟 P 畫面的原理相同，唯一的差別在於 B 畫面可以參考前面的 I 畫面，也可以參考後面的 P 畫面。如同 P 畫面一樣，畫面資訊在參考畫面找不到相似的大區塊時，會使用 Intra 模式編碼。參考前後畫面做動態補償預測編碼，擁有最高的編碼效率，本身不再做為其他預測編碼用 而這三類的畫面在 Group of

picture 中並沒有有一定的安排次序。但是排列方式會影響視訊壓縮效率與隨機處理的能力。

3.3 MPEG-2 壓縮格式、編碼跟解碼

爲了取得高壓縮比效果，MPEG 採用了複合式多種壓縮技巧，首先是以前一畫面爲基礎的動態補償 (block-based motion compensation) 方法，利用前一畫面至目前畫面內容之預測 (prediction)，或是由前一畫面其下移畫面至目前畫面內容之內插預測 (interpolation prediction)，再利用 DCT 轉換除去空間上的相關性，並配合量化 (quantized) 程序略除不重要的資訊，最後經由 VLC (variable length coding) 方式編碼後與動態向量複合產生視訊壓縮編碼。

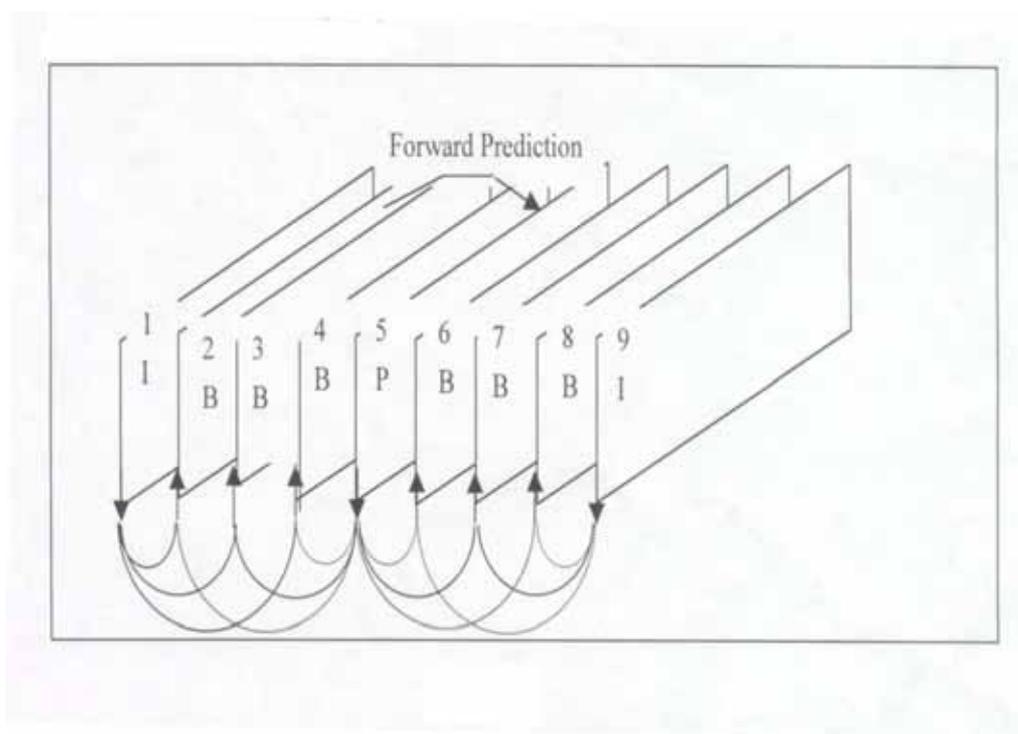


圖 3-2 MPEG2 中的一個 Group of picture

我們對於上述的 (block-based motion compensation) 方法、預測 (prediction)方法以及使用內插預測(interpolation prediction)的方法舉一實例作說明。如上圖所示，我們列出一 Group of picture (包含圖像 I、圖像 B 及圖像 P)：在這範例中，可以發現第一張圖像 I 為獨立編碼，它與其它圖像沒有任何關連。第五張圖像 I 是參考第一張圖像 I，此種編碼方式為順向預測。而第二張圖像 B 不僅參考

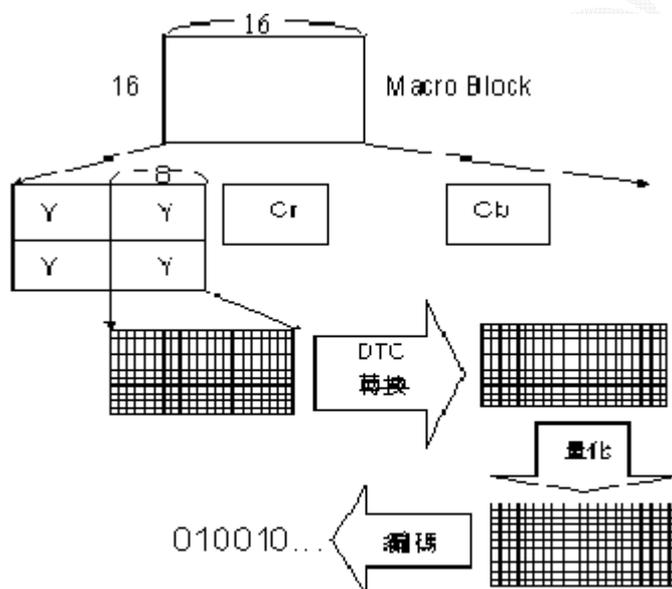
了第一張圖像 I 還包括了未來第五張圖像 P，此種作法稱為內插預測 (interpolation prediction)的方法。Group of picture 中其他部份可按不同圖形型態依此類推。透過動態補償的方式，消除畫面間時間和空間的關聯性，僅用最小的資料量來記錄彼此間的差異，也就是利用 Motion Vector 來記錄，如此可大大的增加壓縮的效率。

MPEG2 是數位電視的標準，主要被用來製作 DVD，其提供兩種解析度 720×480 與 1280×720，每秒可播放 60 張的影像。MPEG 檔案在播放時，一秒中總是有數十張的影像快速閃過，故中間有許多影像會有重複的部分，壓縮時就是壓縮這一部分，除了重複的以外，也可以紀錄影像移動的位移，其中；在 I 畫面中，我們將畫面分割為 16×16 的區塊來處理，這些區塊稱為 macro block。每個 macro block 又由 4 個 8×8 的 Y 區塊、一個 Cr 區塊及一個 Cb

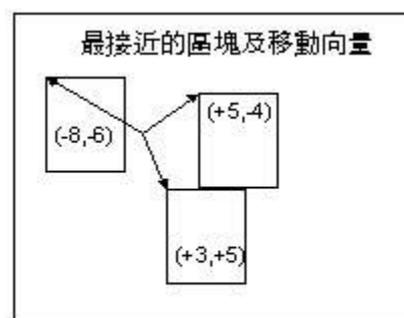
區塊組成。這裡的 Cr、Cb 區塊都是由 16×16 做部份取樣而成 8×8 的區塊，也就是用 YUV411 的格式，如圖(一)。接下來的步驟就先將每個 8×8 的區塊做 DCT 轉換，然後量化，再以 Run-Length 及 Huffman 編碼的方式壓縮。在 P 畫面中，所要儲存的資料主要是該畫面和參考畫面(前一個 I 畫面)的不同處。在同一張畫面中的任何一部份，往往可以在前一張畫面中的某一個位置找到，我們只要記錄那一個部份是從前一張畫面的那一部份移動過來的，就可以使需要儲存的畫面資訊減少許多，這種技巧稱為運動補償 (motion compensation)。在 P 畫面中裡同樣是以 macro block 為單位，每一個 macro block 照理可以在參考畫面中找到。

如圖(二)，我們在參考畫面的某一個範圍內尋找和該 macro block 最接近的，也就是誤差最小的區塊，這個作法稱為區塊比對

(block matching)。如果區塊比對時找到最接近的，就只要記錄該區塊在兩個畫面中的位移，也就是所謂的運動向量 (motion vector)，以及誤差的部份。至於 B 畫面則和 P 畫面的原理相同，只是它的每一個 macro block 的參考畫面可以是從前面的 I 畫面 (或 P 畫面)，也可以是後面的 P 畫面(或 I 畫面)，或者是兩者的平均。因為是雙向的，所以稱為 Bi-directional。I、P、B 三種畫面的壓縮過程如圖(三)所示。



圖(一)



圖(二)

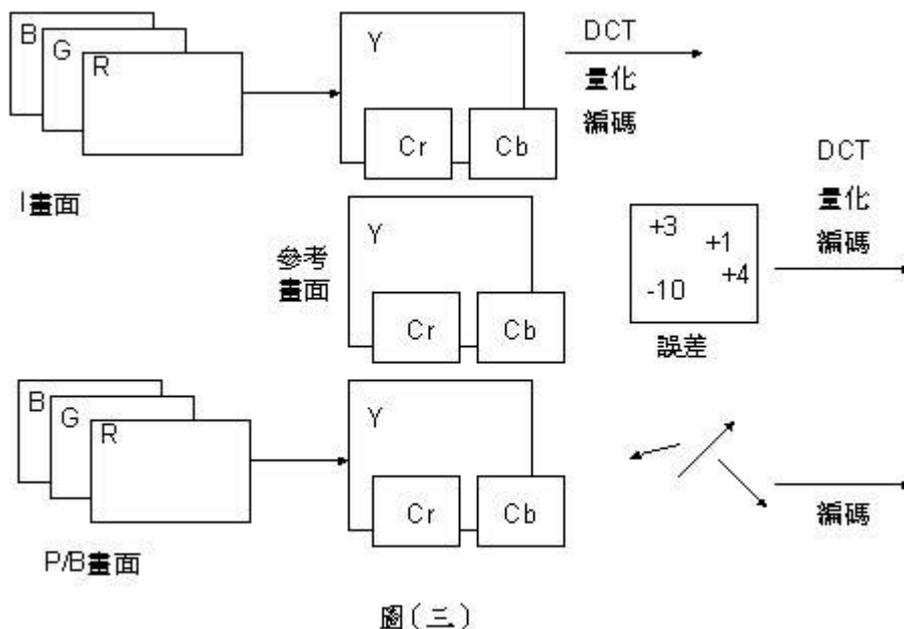


圖 3.3 壓縮流程圖

3.4 MPEG2 的應用

MPEG-2 除了做為 DVD 的指定標準外，MPEG-2 還可用於為廣播，有線電視網，電纜網絡以及衛星直播(Direct Broadcast Satellite)提供廣播級的數字視頻。而 MPEG-2 的另一特點是，其可提供一個較廣的範圍改變壓縮比，以適應不同畫面質量，儲存容量，以及頻

寬的要求。對於最終用戶來說，由於現存電視機分辨率限制，MPEG-2 所帶來的高清晰度畫面質量(如 DVD 畫面)在電視上效果並不明顯，倒是其音頻特性(如加重低音，多聲道等)更引人注目。

3.5 專題的應用

在以上說了那麼多有關 MPEG2 的基本架構、壓縮格式、編碼及解碼的原理。都只是爲了讓大家多了解一些有關 MPEG2 的原理。但是，我們的專題只用作 MPEG2 的 PLUG-IN(外掛程式)，所以，對於和我們專題沒有直接有關的技術，實際上，我們並沒有去鑽研。

那我們專題只對於 MPEG 的 I 畫面做 Water Mark。在 1.2 時我們有說過 I 畫面是壓縮率最差的，也就是說，它失真的程度也最少。而且 P、B 畫面的產生都是根據 I 畫面而來的，所以，我們要在 MPEG2 中去加入 Water Mark，就必須對 I 畫面下手。

第四章 系統的實做

4.1 使用的工具

1. 分割Mpeg2的程式(Visual c++ 6.0)

可以把一個mpeg2的影片分割成多個檔案,檔案格式包含:

YUV , PPM圖檔 等... , 而且又可把分割完的檔案重新結合成

mpeg2的影片,我們就是利用這個特性來藏浮水印,我們選用的檔案

格式是 ppm圖檔. (以下稱此程式為mpeg2 to ppm)

2.藏浮水印的程式(Borland c++ builder 5.0)

可以將8位元的灰階圖藏在另一個8位元灰階圖或24位元的

彩色圖,我們採用的是藏在24位元的方法.

4.2 藏浮水印的基本構想

根據 mpeg2 的製作流程圖,我們可以將整個 mpeg2 的製作切割成下面兩部分,DECODE 和 ENCODE ,而根據我們所得到的程式,更可以將其得出來的 coded video 的 I frame 以.ppm 的圖檔格式存成一張一張的圖檔,所以我們就設想程式裡一定有一個步驟是將每個 I frame 分解成 R,G,B 三個頻道,然後才存成 ppm 圖檔的,所以我們就決定找出這三個頻道的輸出,並把他截住,在程式中做完浮水印的處理,在繼續剩下的步驟,存成 ppm 圖檔,然後再把 ppm 圖檔組合回 mpeg2 的影片,這樣就完成藏的动作.

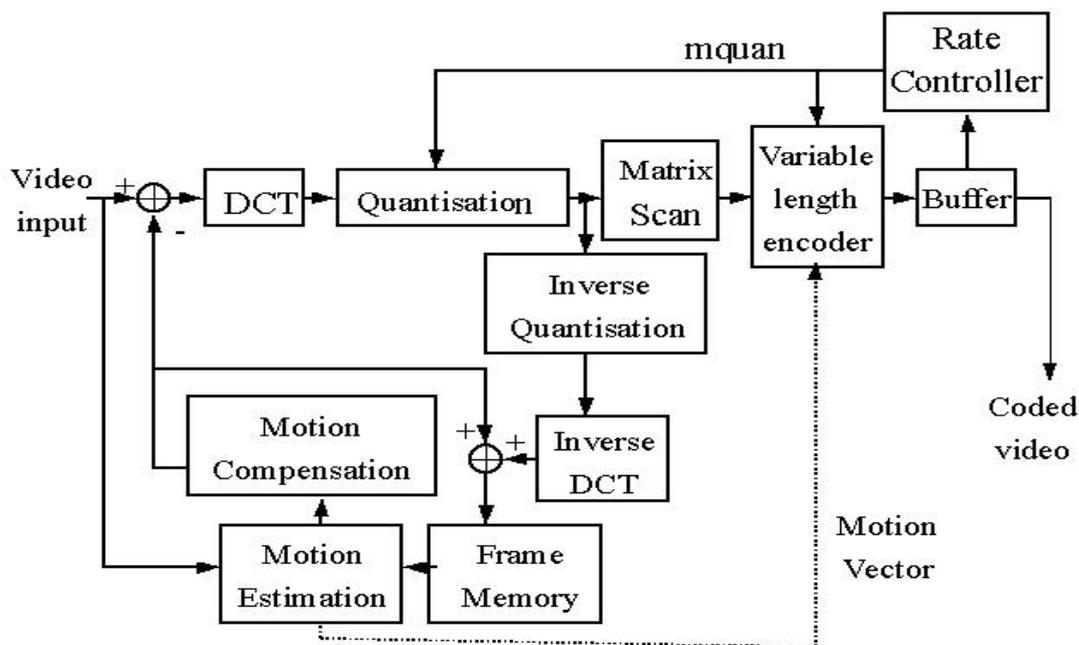


圖 4.1 DECODE

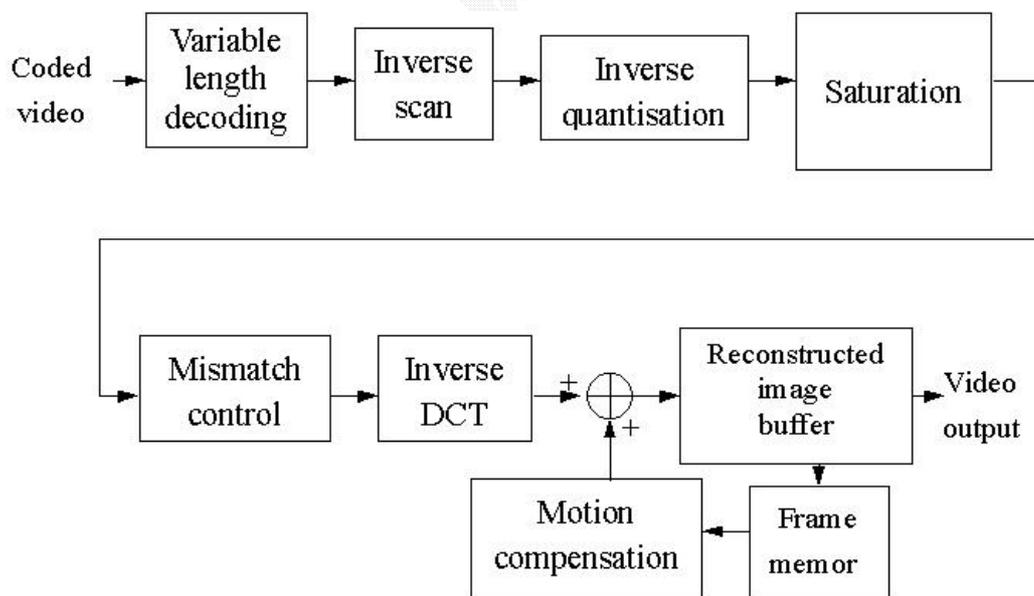


圖4.2 ENCODE

4.3 取浮水印的基本構想

因為我們發現 mpeg2 to ppm 這個程式不管重複執行幾次,存成的圖片都不變,合起來的影片依然完美,所以我們就假設他 decode 和 encode 的所執行的路徑每次都相同,也就是只要輸入的影片沒變,解出的圖片順序就不變,而輸入的圖片不變,合起來的影片就是一樣的,因此我們就在借用這個程式,把加了浮水印的影片輸入,看是在第幾張藏的,就截住那一張的 RGB,抓出所藏的浮水印,這樣就完成取的動作.

4.4 實作中遇到的困難

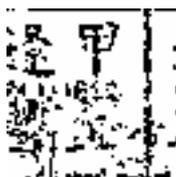
- 1.尋找輸出 RGB 的地方
- 2.浮水印程式的改寫
- 3.浮水印的載入和輸出
- 4.浮水印對影片的破壞

剛開始時,畢竟上述兩點都是猜想的,所以在尋找驗證上花了不少時間,後來找到輸出點,怎麼加.取浮水印成了另一個問題,浮水印的程式是用 Borland c++ builder 5.0 寫的,而 mpeg2 to ppm 則是用 Visual C++, 兩者函式不互通 , 所以根本不能直接套用,只能重新改寫,這時真的深感自己程式功力的薄弱,除此之外,在 載入和輸出浮水印更遇到困難,因為只要是圖檔都有檔頭,那就還要再去辨識檔頭才能載入或輸出浮水印,後來就聽從助教的忠告,將格式改成 RAW 檔(沒有檔頭,只有一堆原始資料),問題就解決啦.而程式剛寫完後發現另一個問題,就是要使取出來的圖清晰的 RobustValue 太高,要定大概 30-50 才能辨認出,所以導致影片被嚴重破壞,所以我們就想了一個方法,把浮水印只存在某一張圖片中,這樣一閃而過,對影片就沒有破壞,而我們只要知道是藏在哪一張,一樣可以輕易的取出.



Robust Value

20



Robust Value

40



原圖

4.5 採用的浮水印技術的簡介

這次專題我們採用的是小波理論,也就是將一張圖片分成四部份, HH , HL , LH , LL 四種頻率,(圖4.3 為作三次小波理論的圖) HH為最不影響人眼的高頻區,LL則是最容易影響的低頻區,正常而言浮水印應該藏在HH區,因為他最不會造成視覺上的影響,但是這次專題我們卻不這樣做,反而藏在最影響視覺得低頻區,原因就在於MPEG2本身也是一種壓縮技術,而他的壓縮也是運用頻域轉換的原理,所以他壓縮的地方也在高頻區,我們如果將浮水印藏在高頻,那將會遭受MPEG2壓縮的破壞,導致取出的浮水印根本不能辨認,故我們

只能藏在低頻區,而這樣的做法就導致分割成單張圖片時,浮水印相當明顯,也使的我們只能用藏影片中的某些片段的方法,使浮水印不那么明顯。

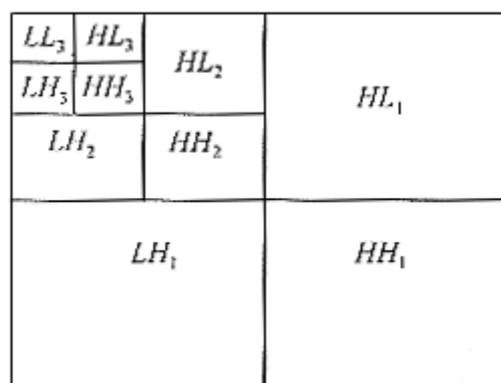


圖4.3 Three-layer pyramid wavelet decomposition

4.6 浮水印必要條件的比較

1.隱藏性(Imperceptibility):

這個程式可以決定藏的張數和順序,可以使在漫長的影片中,浮水印出現不到一秒的時間,不易為人察覺,故具隱藏性.

2.不可去除性(No removable) :

在數千.數萬張裡只存在少數幾張,在沒有取浮水印程式且不知道是藏的順序和張數下,是很難去除的.

3.強健性(Robustness) :

在經過 mpeg2 反覆的 decode 和 encode 下,依然能保有辨識度,可見具有強健性

4.可輕易解讀性(Decode ability) :

當有取浮水印程式和知道藏的順序和張數下,可以輕易取出浮水印.

5.安全性(Security) :

當一個人拿到已加浮水印的影片,可藏和可取的程式以及藏的順序時,他依然不能去除掉他,因為這個程式只可以藏浮水印和另取浮水印出來,而不能改影片裡的浮水印.

4.7 實驗中的一些其他做法和其缺點

1.將浮水印藏在每一張分割的圖

缺點: 浮水印太明顯,嚴重破壞影片品質

修正方法:從中挑幾張來藏

2.每一張PPM圖只藏一張浮水印

失敗處:因時間關西對小波理論未深入了解,所以無法取出

而且對影片破壞跟上一點差不多

3.將浮水印切割成數小段再分別藏入各PPM圖

缺點: 因本身取出浮水印的方法本身就會有點小誤差

取出圖樣會有點小偏移和模糊,所以當我們將藏的各段取出

再結合時,那些小誤差構成的圖,已經是不能辨認了,這是

沒辦法忽略的地方,不然這各方法是最不會破壞影片品質的

方法。

第五章 結論與心得

組員:柯政鋒

這一次的專題製作，真的讓我學到很多以前不曾遇到或是沒機會遇到的寶貴經驗。我們小組從大三下就開始準備籌畫整個專題。從一開始，我們的計劃是打算作出在 mpeg4 的環境下能把影片中的人物物件上的置換。到最後只能對專題的要求向下修改，改成能在 mpeg2 的架構下作 watermark。所以若能一開始在題目企劃能考量多一點，比如在題目的深度上要配合小組的能力是能真的 handle(處理)，這樣之後在整個專題系統進度上將會較好推動。不會因為在系統中的某個環結出了問題而無法如預期中解決，例如:因為當初設想可以從學習中逐一解決，後來才發現問題的複雜度比想像中還困難。

在來就是在將來若有在接專題在開發文件(Document)上要多下

點功夫，因為在我們這次的專題中途有因為小組人員的改制，從 4 人變成 2 人，所以在交接上若有軟件的開發文件詳細記錄，能在系統開發的過程讓分工合作的團隊人員也能在詳讀文件後對系統不熟的部份很快就能上手。

最後要感謝李維斌老師對我們的鼓勵，畢竟軟體開發的過程遇到的困難有時不一定只是在程式上的功力，對我而言，也改變了我對分析問題的角度、看法，和應有的作事態度有了長進。

組員:方柏文

這次的專題不只是讓我學習到新的知識,還讓我學習到一些做人處世和團隊合作的道理,這專題我們從大三就開始做了,一開始定的題目是在 MPEG4 上做圖片的修改,起先大家興致勃勃的收集眾多資料,但隨著資料的齊全,越發覺自己的實力低落,終於大家就決定修改成目前這個題目,然後資料就重新收集,但是研究到切 frame 的階

段時,大家又卡住了,而且大家的懈怠之心又浮現,就這樣卡了一學期,而學期末時,一個組員又不幸退學,我們對專題就更加沒有信心了,還好李維斌老師的一句話”不拼一拼怎麼知道趕不上發表”讓我們重新再奮鬥起來。

但是因為對 java 的不熟悉和大家的鬆懈,我們還是沒能在上學期發表,接下來就開始了我們悲慘的經歷,從剩三個人變成組成八個人一組的大結合,最後剩下我們兩個繼續在李維斌老師下堅持下去,最後證明這個選擇是正確的.

這次做這個專題,感覺受到很多外力的幫助,本來在切 frame 上是要無中生有的,後來剛好有學長在搞這方面,就在機緣巧合間拿到了那個分割 mpeg2 的程式,但是那時拿到時,感覺這程式很龐大,好像不是我們所能理解的,就又要換 jpeg200 的題目,還好李維斌老師堅持在我們不了解為什麼要放棄這個題目前,絕不給我們改題目,讓

我們繼續對 mpeg2 研究下去,也讓我體驗到原來在還沒有花心思下去之前,認為很難的都是假的,當我開始硬著頭皮去看程式時,發覺其實並不是那麼難,而且謹記著老師給我們的忠告,”有些地方你只要知道怎麼用就好,不用去搞懂他”讓我不至於對不需要的地方去鑽牛角尖,讓我可以更大範圍的去了解整個程式是怎麼跑的,進而去改寫程式,這真是讓我受益不淺,堅持留下來,真的是正確的選擇.

¹ 參考資料

<http://moon.tkcna.tku.edu.tw/~ele/HW/VGACARD/mpeg2.txt>

http://www.jasoncash.idv.tw/forum/topic.asp?TOPIC_ID=7030

<http://www.hoyo.idv.tw/data/mpeg.htm>

<http://home.kimo.com.tw/a8550269/MPEG2.htm>

<http://watermarking.unige.ch/video/mpeg2.htm>

http://home.kimo.com.tw/oms_com/edit_a.htm

<http://www.mpeg.org>

