

國立交通大學

電機學院電機產業研發碩士班

碩士論文

結合小波轉換、方塊截短編碼、嵌入式零樹小波與
資料探勘等技巧的新體積型影像壓縮方法

A New Volumetric Image Compression Method:
Combining Techniques of WT, BTC, EZW, and
Data Mining

研究生：林慶安

指導教授：林昇甫 博士

中華民國九十七年一月

結合小波轉換、方塊截短編碼、嵌入式零樹小波與資料探勘等技巧的新
體積型影像壓縮方法

A New Volumetric Image Compression Method:
Combining Techniques of WT, BTC, EZW, and Data Mining

研究生：林慶安

Student: Ching-Ann Lin

指導教授：林昇甫 博士

Advisor: Dr. Sheng-Fuu Lin

國立交通大學

電機學院電機產業研發碩士班

碩士論文

A Thesis

Submitted to College of Electrical and Computer Engineering

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of Master

in

Industrial Technology R & D Master Program on

Electrical Engineering

January 2008

Hsinchu, Taiwan, Republic of China

中華民國九十七年一月

國立交通大學
電機學院電機產業研發碩士班

論文口試委員會審定書

本校 電機學院電機產業研發碩士班 林慶安 君

所提論文：

(中文) 結合小波轉換、方塊截短編碼、嵌入式零樹小波
與資料探勘等技巧的新體積型影像壓縮方法

(英文) A New Volumetric Image Compression Method:
The WT, BTC, EZW, and Data Mining Techniques
Combined

合於碩士資格水準、業經本委員會評審認可。

口試委員：張 翔 林昇甫

潘晴財 _____

何銀成 _____

指導教授：林昇甫

專班主任：柯明達

中 華 民 國 97 年 1 月 10 日

結合小波轉換、方塊截短編碼、嵌入式零樹小波 與資料探勘等技巧的新體積型影像壓縮方法

研究生：林慶安

指導教授：林昇甫 博士

國立交通大學電機學院產業研發碩士班

摘 要

本篇論文提出一個結合四種技巧的壓縮方法，本方法主要是結合小波轉換 (WT)、方塊截短編碼 (BTC)、嵌入式零樹小波(EZW)、資料探勘 (data mining)技巧而成。我們知道WT和BTC是兩種完全不同的壓縮方法，選擇結合這兩方法的原因是因為影像經小波轉換後，其所得的係數矩陣，再經由BTC原理加以分類後，將可獲得更高效應。但使用BTC分類後，相對的也會帶來位元圖產生之現象，然而將資料探勘技巧用於選擇位元圖上，可使所選的位元圖更具代表性，且能有效的減少影像間所須傳送之位元圖數量。在前述一連串之動作後，所須傳送的係數明顯減少了許多，但如以固定長度位元來代表係數時，會使壓縮效果減低，所以在傳輸係數時，是使用了EZW演算法內的係數重建之方式來完成整個壓縮方法。而我們所完成的新方法經實驗證實確實有達到高壓縮比及高品質之優點。但在壓縮上所需的時間，比其它演算法來的差，由此可知若在壓縮時間上能有更進一步改善的話，這個新方法是個相當不錯的方法。本論文的主要貢獻有三，第一，利用BTC原理將小波轉換後之係數矩陣加以分類，使其所需之資料量減少。第二，利用資料探勘技巧有效的找出具代表性之位元圖及重覆之位元圖。第三，完成一個新的壓縮方法，實驗結果顯示此方法在壓縮上能夠達到不錯的成效。

A New Volumetric Image Compression Method: Combining Techniques of WT, BTC, EZW, and Data Mining

Student: Ching-Ann Lin Advisor: Dr. Sheng-Fuu Lin

Industrial Technology R & D Master Program of
Electrical and Computer Engineering College
National Chiao-Tung University

Abstract

This thesis proposes an image compression method which mainly combines four techniques including wavelet transformation(WT), block truncation coding(BTC), embedded zerotree wavelet(EZW), and data mining. They are two completely different types of compression method between WT and BTC. But, after the images were transformed through wavelet, the coefficient matrix being classified by BTC will result in higher effects; as a result, this thesis combines these two extremely different types of method, WT and BTC. However, using BTC to classify also brings about the problem of images generated by bitmap. Applying data mining to choose bitmap will not only make the chosen bitmap more representative but also effectively reduce the bitmap numbers during transforming the images. It is obvious that the above methods can decrease number of coefficient greatly. If we use the bits with fixed length to represent the coefficient, the performance of compression is poor. As a result, during transforming coefficients, the coefficients reconstruction method of EZW is adopted throughout the compression process. The new method in this thesis surely has high compression rate and quality. Nevertheless, the proposed method is more time-consuming if it compares to other algorithms. Therefore, if the problems of processing time can be improved, this new method is quite satisfactory. There are three major contributions of the present study. First, using BTC to classify the coefficient matrix obtained from wavelet transformation reduces data redundancy. Second, data mining techniques can effectively find out the representative and repetitive bitmap. Third, this thesis proposes a novel compression method and the experiment result shows the proposed method has good ability in compression.

誌 謝

首先要以最誠摯的謝意，感謝我的指導教授林昇甫博士，感謝他在研究上的啟迪與指導，以及細心校閱本論文的謬誤之處。同時也要感謝本實驗室的學長建焜、士哲、晉嘉、國育、弦澤、松傑、永吉、逸章、啟耀；一起同甘共苦同學坤龍、宥瑾以及學弟們，由於他們熱心的協助與討論，使得研究過程中的困境能夠迎刃而解。

最後要感謝我親愛的父母，感謝他們的養育之恩以及在我求學生涯中給我最大的鼓勵與支持，使我得以在精神與生活上無後顧之憂，順利完成學業。



A New Volumetric Image Compression Method: Combining Techniques of WT, BTC, EZW, and Data Mining

Student: Ching-Ann Lin Advisor: Dr. Sheng-Fuu Lin

Industrial Technology R & D Master Program of
Electrical and Computer Engineering College
National Chiao-Tung University

Abstract

This thesis proposes an image compression method which mainly combines four techniques including wavelet transformation(WT), block truncation coding(BTC), embedded zerotree wavelet(EZW), and data mining. They are two completely different types of compression method between WT and BTC. But, after the images were transformed through wavelet, the coefficient matrix being classified by BTC will result in higher effects; as a result, this thesis combines these two extremely different types of method, WT and BTC. However, using BTC to classify also brings about the problem of images generated by bitmap. Applying data mining to choose bitmap will not only make the chosen bitmap more representative but also effectively reduce the bitmap numbers during transforming the images. It is obvious that the above methods can decrease number of coefficient greatly. If we use the bits with fixed length to represent the coefficient, the performance of compression is poor. As a result, during transforming coefficients, the coefficients reconstruction method of EZW is adopted throughout the compression process. The new method in this thesis surely has high compression rate and quality. Nevertheless, the proposed method is more time-consuming if it compares to other algorithms. Therefore, if the problems of processing time can be improved, this new method is quite satisfactory. There are three major contributions of the present study. First, using BTC to classify the coefficient matrix obtained from wavelet transformation reduces data redundancy. Second, data mining techniques can effectively find out the representative and repetitive bitmap. Third, this thesis proposes a novel compression method and the experiment result shows the proposed method has good ability in compression.

誌 謝

首先要以最誠摯的謝意，感謝我的指導教授林昇甫博士，感謝他在研究上的啟迪與指導，以及細心校閱本論文的謬誤之處。同時也要感謝本實驗室的學長建焜、士哲、晉嘉、國育、弦澤、松傑、永吉、逸章、啟耀；一起同甘共苦同學坤龍、宥瑾以及學弟們，由於他們熱心的協助與討論，使得研究過程中的困境能夠迎刃而解。

最後要感謝我親愛的父母，感謝他們的養育之恩以及在我求學生涯中給我最大的鼓勵與支持，使我得以在精神與生活上無後顧之憂，順利完成學業。



目錄

中文摘要	i
英文摘要	ii
誌謝	iii
目錄	iv
圖目錄	vi
表目錄	ix
	
第一章 緒論	1
1.1 研究動機	2
1.2 相關研究	4
1.3 論文架構	4
第二章 基礎知識與相關研究探討	5
2.1 影像壓縮處理	5
2.1.1 小波編碼	5
2.1.2 方塊截短編碼	10

2.1.3 嵌入式零樹小波編碼	13
2.1.4 算術編碼	19
2.2 資料探勘	20
第三章 新體積型影像壓縮方法	23
3.1 影像編碼概述	23
3.2 影像前處理	24
3.3 小波係數分類	25
3.3.1 小波係數切割	26
3.3.2 小波係數分類與處理	27
3.4 位元圖選取方式	32
3.5 小波係數重建編碼	35
3.6 序列影像位元圖組更新	38
第四章 實驗結果與分析	41
4.1 實驗機制	41
4.1.1 實驗設備	41
4.1.2 效能評估	44
4.2 實驗結果	44
4.2.1 改良式與傳統式資料探勘效能比較	44
4.2.2 單張影像效能比較	48
4.2.3 體積型影像效能比較	49
4.3 實驗分析	55
第五章 結論與未來展望	58
參考文獻	I



圖目錄

2.1 影像壓縮與解壓縮流程圖。	5
2.2 一維影像 P 示意圖。	6
2.3 比例函數示意圖。(a) 基底向量 ϕ_0^1 示意圖。(b) 基底向量 ϕ_1^1 示意圖。	7
2.4 一維影像 P 之兩基底向量組合示意圖。	7
2.5 正交基底向量示意圖。(a) 正交基底向量 ψ_0^1 。(b) 正交基底向量 ψ_1^1 。	7
2.6 L 和 H 二個頻帶示意圖。	8
2.7 四個頻帶示意圖。	8
2.8 七個頻帶示意圖。	8
2.9 原始影像。	9
2.10 二階段小波轉換結果示意圖。	10
2.11 BTC 之編碼及解碼流程圖。(a) 編碼端示意圖。(b) 解碼端示意圖。	11
2.12 位元圖。	11
2.13 三階段小波係數矩陣。	13
2.14 父子對應的三種關係示意圖。(a) 第一種父子對應關係。(b) 第二種父子對應關係。(c) 第三種父子對應關係。	14
2.15 Z 字型掃瞄方式。	16
2.16 第二回合前輸入係數矩陣。	18
3.1 WT-BTC 系統架構示意圖。	24
3.2 原影像與 YCbCr 影像對照圖。(a) RGB 平面。(b) Y 平面。(c) Cb 平面。(c) Cr 平面。	25
3.3 小波係數切割之流程圖。	26
3.4 高度相關之 BTC 係數圖。	28

3.5 高度相關係數之重建還原圖。	28
3.6 中度相關之 BTC 係數圖。	29
3.7 2×2 及 4×4 位元圖之流程圖。	30
3.8 中度相關之位元圖。	30
3.9 中度相關係數之重建還原圖。	31
3.10 低度相關之 BTC 係數圖。	31
3.11 位元圖組示意圖。(a) 個數為 2 的位元圖。(b) 個數為 2 的位元圖。(c) 個數為 6 的位元圖。(d) 個數為 6 的位元圖。	33
3.12 決定最低門檻值的流程圖。	34
3.13 位元圖組成長機制流程圖。	35
3.14 位元圖組成長機制示意圖。	36
3.15 重要係數重建編碼流程圖。	38
3.16 位元圖組更新流程圖。	40
4.1 Google Earth 空照圖。(a) 山脈。(b) 道路。(c) 有雲霧。(d) 都市。	42
4.2 核磁共振的醫學斷層掃描影像。(a) 第一層。(b) 第二層。(c) 第四層。 (d) 第七層。	43
4.3 LENA 圖。	44
4.4 claire-orig 影片圖。	44
4.5 第 15 張空照影像壓縮還原圖。(a) 參數值 5 無成長壓縮還原圖。(b) 參數 值 5 無成長壓縮還原圖。(c) 參數值 7 無成長壓縮還原圖。(d) 參數值 7 有成長壓縮還原圖。	45
4.6 參數值 z 與位元圖組之關係圖。	46
4.7 第 15 張空照影像壓縮還原圖。(a) 原圖。(b) 參數值為 5 的資料探勘法 壓縮還原圖。(c) 參數值為 7 的資料探勘法壓縮還原圖。(d) 改良式資 料探勘法壓縮還原圖。	47
4.8 LENA 原圖。(a) 原圖。(b) 2D PROGRES 壓縮還原圖。(c) 2D SPIHT 壓 縮還原圖。(d) WT-BTC 壓縮還原圖。	48
4.9 單張影像壓縮之 $PSNR$ 與 C_R 關係圖。	49
4.10 第 8 張空照影像壓縮還原圖。(a) 原圖。(b) 3D SPIHT 壓縮還原圖。(c) 3D PROGRES 壓縮還原圖。(d) WT-BTC 壓縮還原圖。	50

4.11 體積型之空照影像的 $PSNR$ 與 C_R 關係圖。	51
4.12 第二層醫學斷層掃描圖。 (a) 原圖。 (b) 3D SPIHT 壓縮還原圖。 (c) 3D PROGRES 壓縮還原圖。 (d) WT-BTC 壓縮還原圖。	52
4.13 體積型之斷層掃描影像的 $PSNR$ 與 C_R 關係圖。	53
4.14 體積型影像序列區間與 $PSNR$ 關係圖。	54
4.15 3D SPIHT 與 WT-BTC 在影片中 $PSNR$ 變化圖。	54
4.16 三階小波係數組合圖。 (a) 2D SPIHT 演算法。 (b) WT-BTC 演算法。	55



表目錄

2.1 第一階段後小波係數與對應符號表。	16
2.2 重要係數修正表(第一回合)。	17
2.3 資料探勘分類表。	20
3.1 第三階小波係數之切割方式與 <i>PSNR</i> 關係表。	27
3.2 小波轉換係數分類表。	27
3.3 相關係數分類規格表。	37
3.4 各影像所需位元圖組表。(單位：個)	39
4.1 實驗影像分類表。	42
4.2 <i>PSNR</i> 與壓縮率比較表。	46
4.3 傳統式與改良式資料探勘之比較表。	47
4.4 單張影像之位元率與 <i>PSNR</i> 比較表。(單位：dB)	49
4.5 空照體積型影像位元率與 <i>PSNR</i> 比較表。(單位：dB)	51
4.6 醫學體積型影像位元率與 <i>PSNR</i> 比較表。(單位：dB)	53
4.7 EZW 演算法重建疊代次數與 <i>PSNR</i> 關係表。	56
4.8 不同影像之 <i>PSNR</i> 、位元率與位元圖組。	56
4.9 位元率與位元圖組之關係表。	56
4.10 相同位元率下 WT-BTC 與 2D SPIHT 所需時間表。(單位：秒)	57
4.11 相同位元率下 WT-BTC 與 3D SPIHT 所需時間表。(單位：秒)	57

第一章

緒論

對影像壓縮這技術可以追溯到超過 40 年前。起初這領域的研究工作重點是放在降低視訊傳輸頻寬的類比方法，稱為頻寬壓縮(bandwidth compression)。隨著時代進步，影像壓縮技術逐漸轉變成為數位化的方式處理，並伴隨著幾個主要的國際影像壓縮標準被採用，從此影像壓縮也被認為是可行的技術。

除此之外，影像壓縮在許多方面均扮演著重要的角色，包含醫學影像診斷、視訊會議、傳真、及辨識系統、倒車雷達警示、3D 影像重建等。而影像壓縮技術又可分為兩大類：資訊保持(information preserving)類及耗損(lossy)類。而兩種影像壓縮皆各有特色。在醫學影像方面，比較需要有細節的影像資訊來作為是否有疾病的判斷依據，所以大都使用資訊保持的方式來實行。而其他如視訊會議方面，因呈現的方式是即時的，所以在資料的傳遞上需比較快。相對的，如果資料壓縮率越高，在相同時間內所能傳送的資料量就越多，故往往會使用耗損類壓縮方式來執行。

在本章節中，將在 1.1 節介紹本論文的研究動機；在 1.2 節介紹相關研究；最後在 1.3 節介紹本論文的架構。

1.1 研究動機

在上述的應用介紹裡，其中一項 3D 影像重建，則為此篇論文的主要研究動機。在 3D 影像重建的領域中，有許多的方法廣為人知，如照度差立體法(photometric stereo)，shape-from-texture，立體影像作距離計測法，光投影法，體積型影像重建法，而其中體積型影像重建方法，為醫學上主要透過段層掃描方式所得之影像，一張張如體積般疊起所作出來的重建模型，而體積型影像與一般序列影像最大差異，在於其一張張連續影像其彼此差值都是頗大的，無法如視訊壓縮一樣預測下一張影像，然而如此差異大的序列影像壓縮，就成為本論文研究的動機。根據上述動機，本論文研究的目的為藉由近年來影像壓縮技術，及結合資料探勘 (data mining) [1]及基因演算法[2][3]所組合而成的改良式方法。在一序列差異大的影像中，找出其資料冗餘性 (data redundancy)，其資料冗餘性的相關研究，將於本論文的 1.2 節內說明。

1.2 相關研究



資料冗餘性是數位影像壓縮比的重要一環。如果用 n_1 與 n_2 代表同一個資訊之兩組資料的資料量，則第一組資料集合 (n_1 代表第一組) 的相對資料冗餘性 (relative data redundancy) R_D 可定義成

$$R_D = 1 - \frac{1}{C_R}, \quad (1.1)$$

其中 C_R 通常稱為壓縮率 (compression ratio)

$$C_R = \frac{n_1}{n_2}. \quad (1.2)$$

在一般而言， R_D 與 C_R 分別落在開區間 $(0, \infty)$ 與 $(-\infty, 1)$ 這兩區間內。對於一個像是壓縮率為 20 的資料而言，意味著已壓縮的資料內的每一個單元，可以對應未壓縮資料內 20 個資料量 (比如說位元)。相對的冗餘性 0.95 意味第一組資料中有 95% 是冗餘的。

在數位影像壓縮中，參考文獻[4]內有提到三種基本的資料冗餘性是被公認及可被利用的：編碼（coding）冗餘性、像素間（interpixel）冗餘性及視覺心理（psychovisual）冗餘性。當減少及消除這些冗餘性後就可以達到資料壓縮的效果。

因為當資料壓縮後及解壓縮還原後，感興趣的資訊可能丟失，所以很需要一個方法以衡量資訊丟失的特性與程度。而這種評估準則分為兩大類：(1) 客觀保真度準則（objective fidelity criterion）(2) 主觀保真度準則（subjective fidelity criteria）。當損失的程度可以表示成輸入影像經壓縮及解壓縮後輸出影像的函數時，這就是客觀的保真度準則。輸入與輸出影像之間的均方根（rms）誤差就是一個好例子。假設 $f(x, y)$ 代表輸入影像， $\hat{f}(x, y)$ 代表經壓縮與解壓縮後所得的輸出影像，在任意的 x 或 y 值下，我們對 $f(x, y)$ 與 $\hat{f}(x, y)$ 之間的誤差 $e(x, y)$ 定義為

$$e(x, y) = f(x, y) - \hat{f}(x, y), \quad (1.3)$$

因此，影像間的總誤差為

$$\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\hat{f}(x, y) - f(x, y)]^2, \quad (1.4)$$

其中影像大小為 $M \times N$ 。而兩張影像的均方根誤差（root-mean-square error） e_{rms}

$$e_{rms} = \left[\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\hat{f}(x, y) - f(x, y)]^2 \right]^{1/2}. \quad (1.5)$$

而另一個與客觀保真度準則有緊密相關的為均方訊號雜訊比（mean-square signal-to-noise ratio），用 SNR_{rms} 表示

$$SNR_{rms} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \hat{f}(x, y)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\hat{f}(x, y) - f(x, y)]^2}. \quad (1.6)$$

雖然客觀保真度準則是一種簡單且方便的評估損失機制，但在解壓縮完後的影像觀

察者還是人。因此用一群人來評估衡量解壓縮後的影像品質會更適合。其做法為：找一群適當的觀察者，讓他們使用一個絕對等級尺度與解壓縮後的影像作比較。而可以用如 $\{-3,-2,-1,0,1,2,3\}$ 的數量化來代表主觀評價{很差、較差、稍差、相同、稍好、較好、很好}。

近年來，除了使用均方根誤差 (e_{rms}) 來作客觀的保真度準則評估機制外，另一個評估機制 峰值信噪比 ($PSNR$) 及 位元率 (bpp or b/p) -- 壓縮率 (C_R) 漸漸成為評斷的主流。

$$PSNR = 10 \log_{10} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \frac{255^2}{[\hat{f}(x, y) - f(x, y)]^2} dB. \quad (1.7)$$

1.3 論文架構

本論文章節的安排如下，在第二章中將介紹論文會使用到的相關技術以及方法，例如對小波轉換編碼[5][6]、方塊截短編碼[7][8]、EZW 編碼[9]、算術編碼[10]及資料探勘。第三章則針對我們提出的 WT-BTC 壓縮方法來討論。第四章會顯示實驗結果和可行性，最後第五章則說明對未來後續研究方向及建議。

第二章

影像相關知識及理論

在這一章中我們將介紹本論文中將使用到的影像處理技術及影像壓縮處理流程如圖 2.1，並在 2.1 節內說明小波編碼(wavelet coding)，方塊截短編碼(block truncation coding)和嵌入式零樹小波(embedded zerotree wavelet)之編碼方式及目前其各研究，2.2 節則簡略介紹資料探勘(data mining)，2.3 節則介紹我們在壓縮及解壓縮最後所做的算術編碼(arithmetic coding)。

2.1 影像壓縮處理

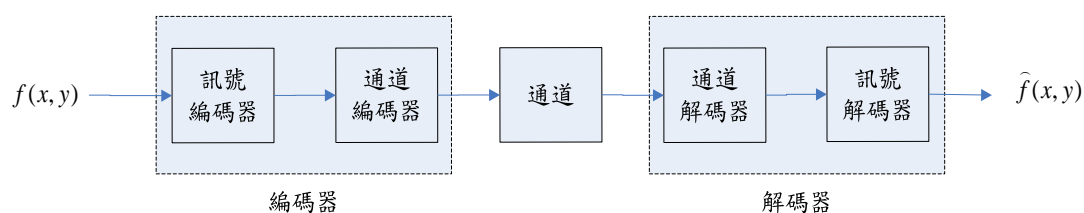


圖 2.1 影像壓縮與解壓縮流程圖。

2.1.1 小波編碼

在解釋小波是什麼之前，我們就對一些在小波研究上常用到的觀念及名詞，來舉例

作解釋。我們對一串取樣值(訊號)使用一個 Haar 階梯函數(step function)來說明。

1.Haar 比例函數(scaling function)

$$\phi(x) = \begin{cases} 1, & x \in [0,1), \\ 0, & x \notin [0,1), \end{cases} \quad (2.1)$$

其中， ϕ 滿足所謂的比例方程式

$$\phi(x) = \sum_{i \in \mathbb{Z}} C_i \phi(2x - i), \quad (2.2)$$

給一個向量空間 V^k ，其基底向量可由 2^k 個比例函數構成，這 2^k 個比例函數分別用

$\phi_0^k, \phi_1^k, \phi_2^k \dots \phi_{2^k-1}^k$ 來表示。在這些基底向量中，我們使用下式來檢查其正交性

(orthogonality)

$$\langle \phi_i^k, \phi_j^k \rangle = \int_0^1 \phi_i^k \phi_j^k dx = 0. \quad (2.3)$$

接下來我們再找一個與 V^k 正交且互補的新向量空間 $\bar{V}^k (\perp V^k)$ 且 $\bar{V}^k \subset V^{k+1}$ 。而形成 \bar{V}^k

基底向量用 $\psi_0^k, \psi_1^k, \psi_2^k \dots \psi_{2^k-1}^k$ 表示，而構成 \bar{V}^k 向量空間的基底向量則稱為小波

集(wavelet sets)。為了更清楚的解釋小波的定義，所以我們以兩個比例函數 ϕ_0^1 與 ϕ_1^1 來

取代一維影像 P 的例子來做說明。一維影像 P 如圖 2.2 所示，比例函數如圖 2.3(a)、(b)

所示，取代之示意圖如圖 2.4 所示。

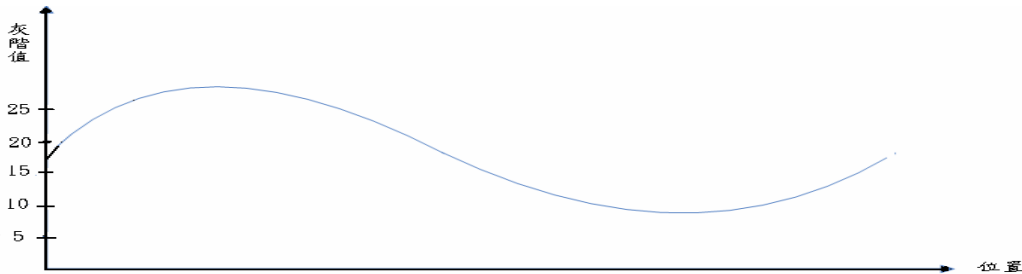


圖 2.2 一維影像 P 示意圖。

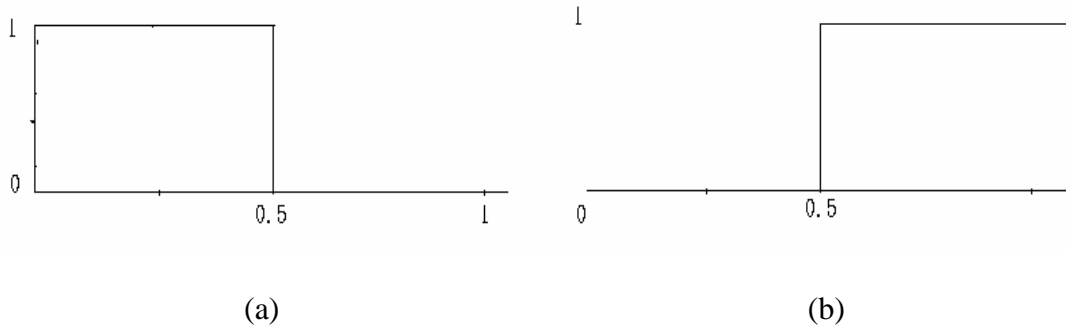


圖 2.3 比例函數示意圖。(a) 基底向量 ϕ_0^1 示意圖。(b) 基底向量 ϕ_1^1 示意圖。

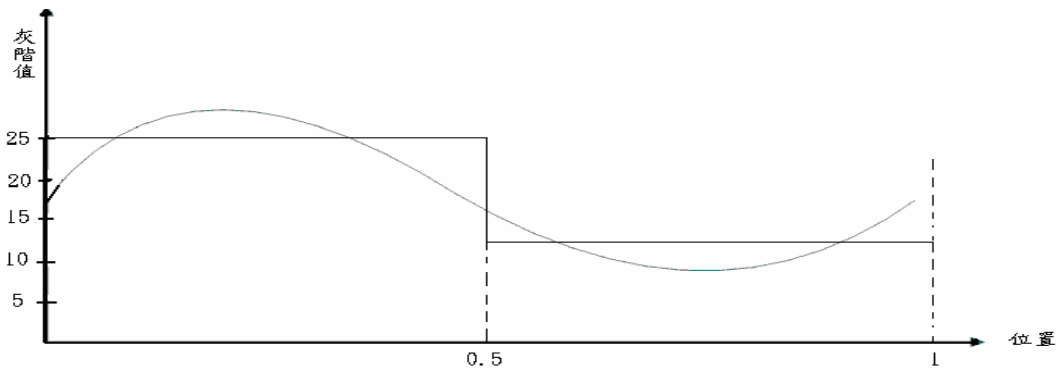


圖 2.4 一維影像 P 之兩基底向量組合示意圖。

由圖 2.2-2.4 知，一維影像 P 可以由 ϕ_0^1 及 ϕ_1^1 基底向量作線性組合來表示。若將 ϕ_0^1 與 ϕ_1^1 兩組成一維影像 P 之基底向量，改以圖 2.5(a)、(b) 所示之正交基底向量取代 ϕ_0^1 與 ϕ_1^1 ，則此動作稱為小波轉換。

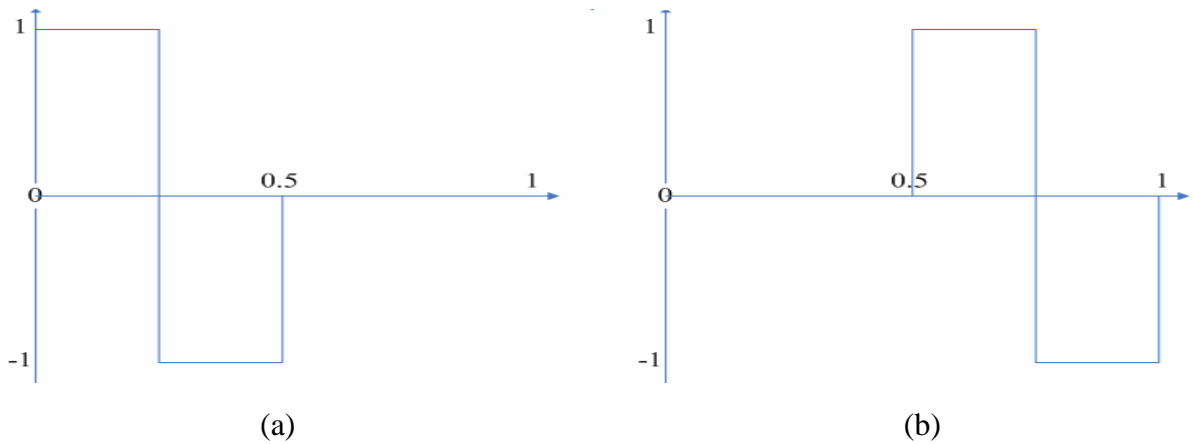


圖 2.5 正交基底向量示意圖。(a) 正交基底向量 ψ_0^1 。(b) 正交基底向量 ψ_1^1 。

由圖 2.5(a)、(b)可知正交向量 ψ_0^1 、 ψ_1^1 是構成 ϕ_0^1 及 ϕ_1^1 的基底向量，所以如果一維影像 P ，其 P 數量化為 $P=(3,5,2,8)$ 並用 ϕ_0^2 、 ϕ_1^2 、 ϕ_2^2 和 ϕ_3^2 四組的線性組合來表示，則 $P=3\phi_0^2+5\phi_1^2+2\phi_2^2+8\phi_3^2$ 。而若基底向量 ϕ_2^2 與 ϕ_3^2 改用正交基底向量 ψ_0^1 與 ψ_1^1 表示時，我們可以得一維影像 $P=3\phi_0^2+5\phi_1^2+(-1\psi_0^1)+(-3\psi_1^1)$ ，而前面係數則變換成較小的係數。

仿照上述，一張二維影像我們先進行行方向的小波轉換後，接著再進行列方向的小波轉換後可得如圖 2.6 所示，在圖 2.6 中，L 代表低頻帶而 H 代表高頻帶。

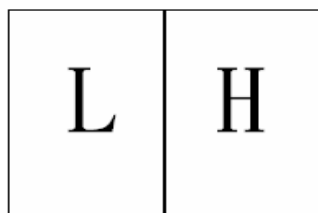


圖 2.6 L 和 H 二個頻帶示意圖。

若我們繼續對 L 及 H 進行平均(averaging)和相差(differencing)的運算，則可得圖 2.7 的四個頻帶。

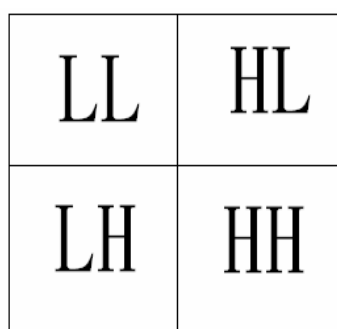


圖 2.7 四個頻帶示意圖。

同理，依照上述對四個頻帶繼續作平均和相差，但由於 LH、HL 和 HH 含蠻多零的值，

所以對只對 LL 頻帶進行平均和相差的運算。圖 2.8 為七個頻帶示意圖，或者稱為二階段的小波轉換。

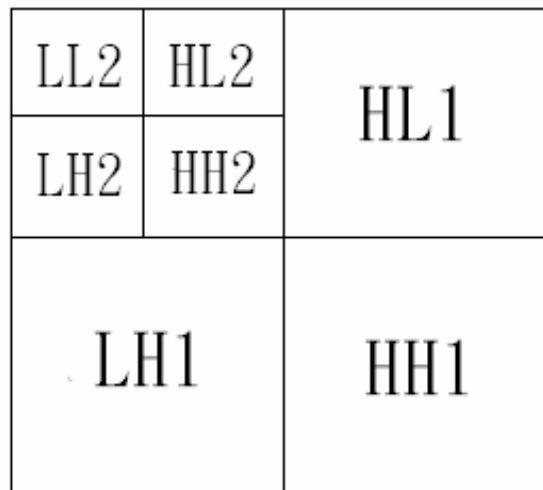


圖 2.8 七個頻帶示意圖。

我們以一個實際的例子來表示，給定一輸入影像如圖2.9所示，圖2.10為二階小波轉換之結果。



圖2.9 原始影像。

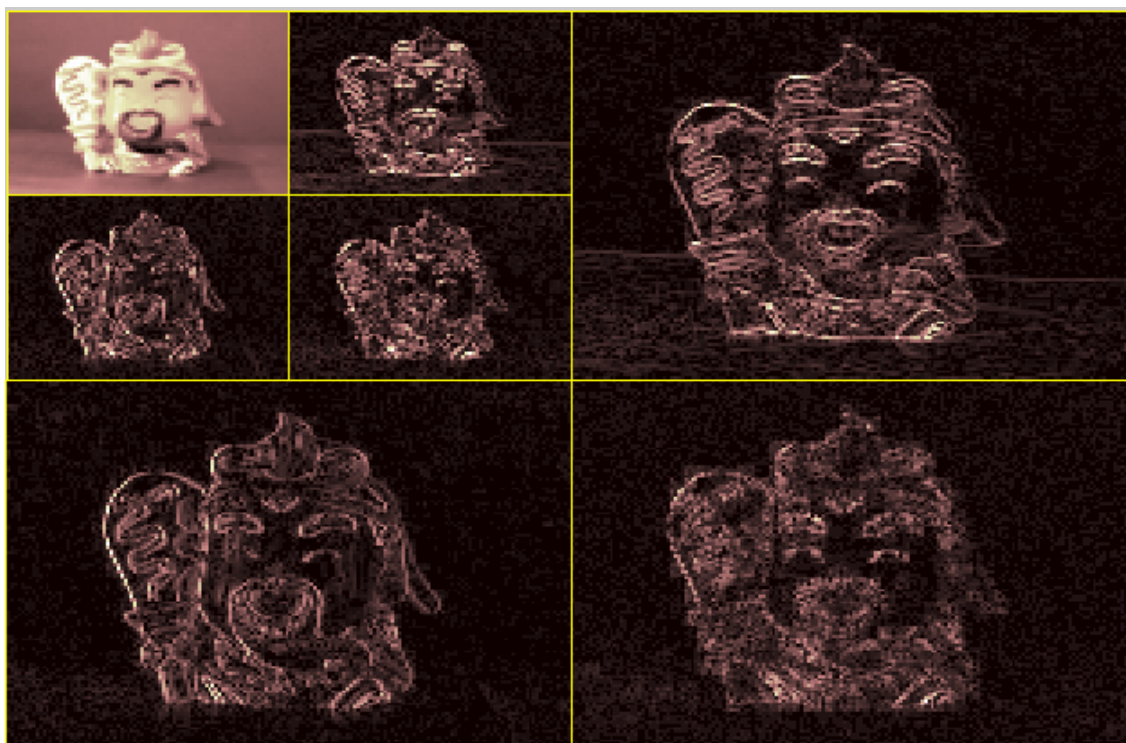


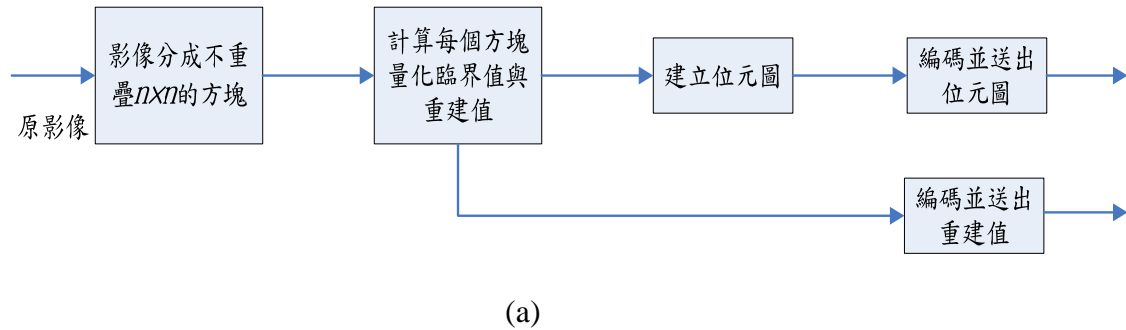
圖2.10 二階段小波轉換結果示意圖。

而在小波編碼領域中，除了Haar濾波外還有知名的9/7浮點濾波器[11]和5/3整數小波[12]可使轉換後的影像有不同的效果，以及除改變濾波器外，也有從小波係數所推展出的壓縮演算法如知名的SPIHT[13]、EBCOT[14][15][16]等。

2.1.2 方塊截短編碼

方塊截短編碼，簡稱 **BTC**。在 **BTC** 系統中，影像首先被分割成 $n \times n$ 方塊(一般都分成 4×4)，每個方塊上面的所有像素原先可能有各種值，經過 **BTC** 編碼後，只會剩下兩個值。對於任意一個方塊，我們均以一個位元圖(bit-map)及兩個重建值， a 和 b 來表示該方塊。位元圖及重建值的計算程序是這樣子的:首先我們計算原方塊中每一個像素的量化臨界值(通常取該方塊之平均值)，令其為 X_{th} ;然後將原方塊中每一像素值與 X_{th} 作比較，如果小於其相對應之位元圖位置設置為 1;否則設為 0。而設置為 1 之重建值則為 b ，反之則為 a 。以此為原則完成 **BTC** 編碼，其流程圖如圖 2.11 所示。

編碼端



解碼端

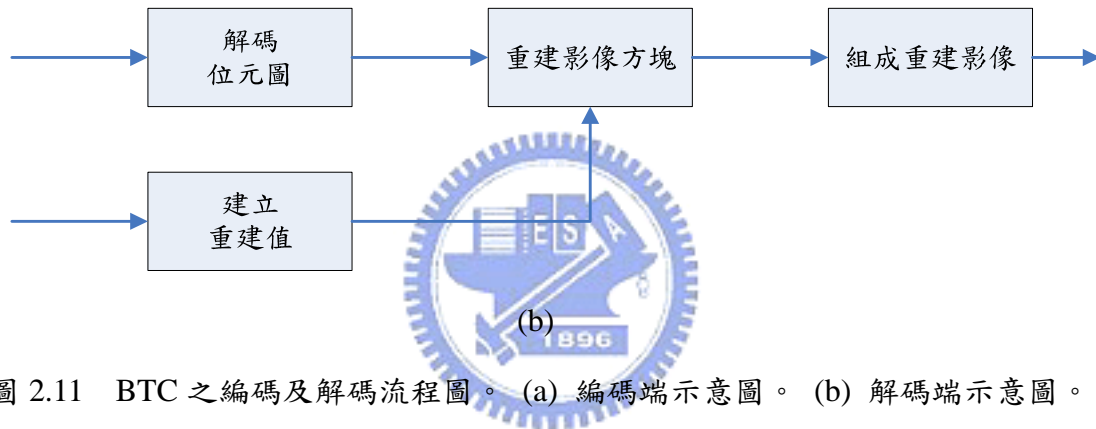


圖 2.11 BTC 之編碼及解碼流程圖。(a) 編碼端示意圖。(b) 解碼端示意圖。

舉例說明，如下所示為原影像取出之一個 4×4 方塊：

$$X = \begin{bmatrix} 60 & 102 & 127 & 167 \\ 30 & 40 & 43 & 53 \\ 30 & 33 & 30 & 28 \\ 21 & 23 & 23 & 23 \end{bmatrix},$$

而其平均值 $\bar{X} = 52.06$ ，所有低於平均值的位置則設定為 0 大於設定為 1 所成位元圖

如圖 2.12 所示。

$$Y = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

圖 2.12 位元圖。

再經位元圖所示，於位元圖中其 0 與 1 之位置，對應回原係數方塊之係數，其 0 位置之係數平均值，稱為暗部平均值以 a 表示。而 1 位置之係數平均值，稱為亮部平均值以 b 表示。依上述之定義，在本例中兩平均值之值為 $a=30$ 、 $b=102$ 。在 BTC 系統中會把 a 和 b 兩重建值與 Y 位元圖一起送到接收端。當接收端收到後，依據位元圖分佈在分別將 a 或 b 填入相對應位置可得到以下的重建方塊。

$$X = \begin{bmatrix} 102 & 102 & 102 & 102 \\ 30 & 30 & 30 & 102 \\ 30 & 30 & 30 & 30 \\ 30 & 30 & 30 & 30 \end{bmatrix}.$$

由此上例可知 BTC 系統的編碼與解碼，均較其他壓縮方式來的簡單。但其效能會受量化臨界值的選擇所影響，而位元圖數量的多少也是一個影響壓縮品質好壞與壓縮率高低的關鍵之一。在最早提出 BTC 系統之研究中，其量化器之設計是保留方塊之平均值 \bar{X} 及方差值 σ 為主。

公式中的 m 值為原影像切出的 m 個 $n \times n$ 方塊，其平均值及方差值公式如下

$$\bar{X} = \frac{1}{m} \sum_{i=1}^m X_i, \quad (2.1)$$

$$\overline{X^2} = \frac{1}{m} \sum_{i=1}^m X_i^2, \quad (2.2)$$

$$\sigma^2 = \overline{X^2} - \bar{X}^2. \quad (2.3)$$

由式(2.3)中看出，只要 \bar{X} 與 $\overline{X^2}$ 無改變，則方差值也能維持不變。接下則決定重建值 a 與 b 在[6]中提到，因其希望保留 \bar{X} 與 $\overline{X^2}$ 值，所以給定一個 q 值，其 q 值為方塊中大於臨界值數目的像素個數。再把量化臨界值設為平均值 \bar{X} 則可解得 a 與 b


$$a = \bar{X} - \sigma \sqrt{\frac{q}{m-q}}, \quad (2.4)$$

$$b = \bar{X} + \sigma \sqrt{\frac{m-q}{q}}. \quad (2.5)$$

由式(2.4)、式(2.5)可得 $a=24$ 、 $b=114$ 兩重建值，雖然 b 比較接近原始的值但 a 部分明顯的比原方塊提高許多，並且在計算 σ 時需費時故後來有提出所謂的 AMBTC[17]及適應性 AMBTC 等方法。除此之外，其位元圖的數量也是一個可以改良的缺失，因如果依原 BTC 系統架構，所產生之位元圖數量會依照方塊切割大小($n \times n$)而出現 2^n 個，所以在本論文之第三章會提出一個方法來解決上述之問題。

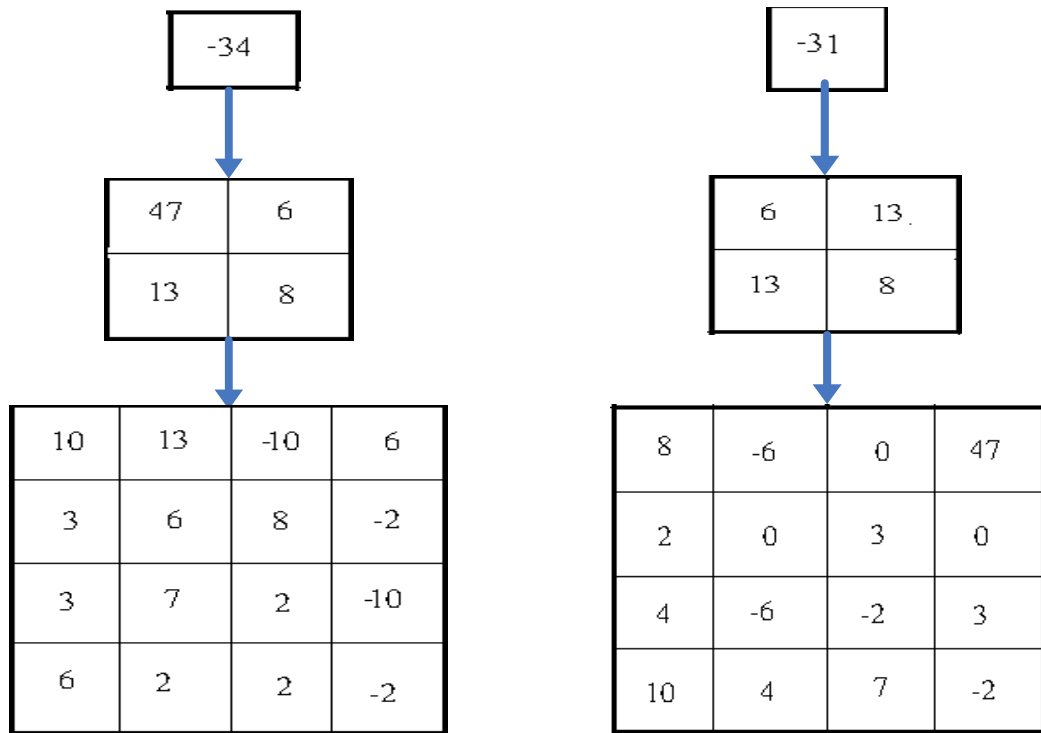
2.1.3 嵌入式零樹小波編碼

嵌入式零樹小波編碼，簡稱 EZW。而其編碼方式將會在本節中介紹，首先是將一個 8×8 的三階小波係數矩陣如圖 2.13 所示，分為三種父子對應關係如圖 2.14(a)、(b)、(c)所示。



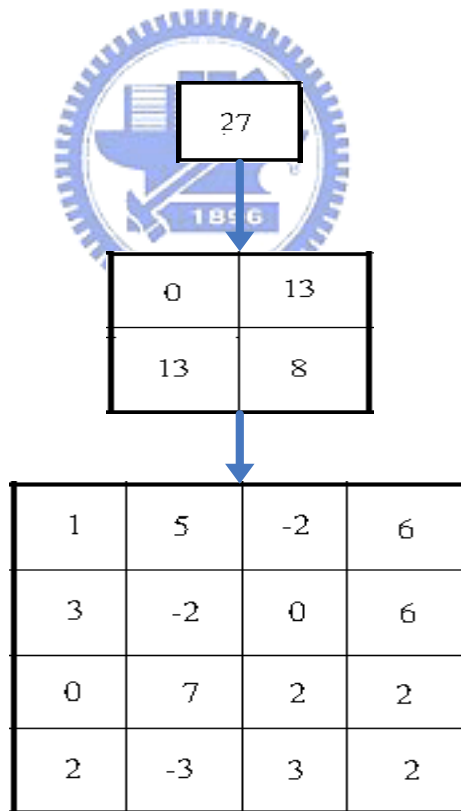
57	-34	47	6	10	13	-10	6
-31	27	13	8	3	6	8	-2
6	13	0	13	3	7	2	-10
13	8	13	8	6	2	2	-2
8	-6	0	47	1	5	-2	6
2	0	3	0	3	-2	0	6
4	-6	-2	3	0	7	2	2
10	4	7	-2	2	-3	3	2

圖 2.13 三階段小波係數矩陣。



(a)

(b)



(c)

圖 2.14 父子對應的三種關係示意圖。(a) 第一種父子對應關係。(b) 第二種父子對應關係。(c) 第三種父子對應關係。

當分為三種父子關係後，再回頭檢查原始係數矩陣，由於原始係數矩陣是小波轉換係數矩陣，依小波特性和可知，低頻帶係數之絕對值會大於高頻帶係數值。故在 EZW 演算法中使用到幾個專有名詞。

專有名詞定義如下：

定義 2.1

門檻值 T_0

$$T_0 = 2^{\lceil \log(\max(Y_{ij})) \rceil} \quad (2.6)$$

$\max(Y_{ij})$ 為小波係數矩陣中係數取絕對值後最大者，以圖 2.13 為例 $T_0=32$ 。

定義 2.2

ZTR(zerotree root): 有一小波係數之絕對值小於某一門檻值 T 時，且該小波係之子孫係數值皆小於 T 時，該係數稱為 ZTR 類係數。

定義 2.3

IZ(isolated zero): 有一小波係數之絕對值小於 T ，但其子孫中，存有至少一個大於 T 之係數時，則稱該小波係數為 IZ 類係數。

定義 2.4

POS(positive significant): 有一正的小波係數，其值大於 T ，則該小波係數稱為 POS 類係數。

定義 2.5

NEG(negative significant): 有一負的小波係數，其絕對值大於 T ，則該小波係數稱為 NEG 類係數。

定義 2.6

Z(zero): 該係數無子孫，且該係數之絕對值小於 T 及祖先非 ZTR。

以 $T_0=32$ 為例，在圖 2.14(a) 中係數 -34 絕對值大於 32 所以該係數 -34 歸為 NEG 類係數；在圖 2.14(b) 中係數 -31 絕對值小於 32，但子孫中存在一係數 47 比 32 大，所以係數 -31 歸為 IZ 類係數；在圖 2.14(c) 中係數 27 小於 32 且子孫之係數皆小於 32，故係數 27 歸為

ZTR 類係數。在 EZW 編碼中是採用 Z 字型掃瞄方式，如圖 2.15 所示。

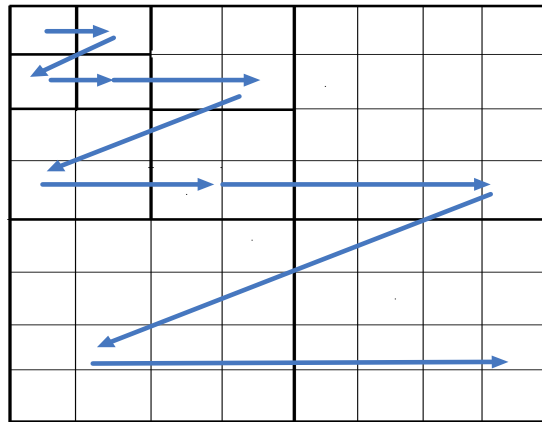


圖 2.15 Z 字型掃瞄方式。

在 EZW 演算法中，如其係數值是大於 T 但小於 $2T$ 時，給予重建值 $3T/2$ ，而屬於 ZTR 類之係數則將列為不重要而不編碼處理。故依以上之原則可得表 2.1

表 2.1 第一階段後小波係數與對應符號表。

小波係數	符號	重建值
57	POS	48
-34	NEG	-48
-31	IZ	0
27	ZTR	0
47	POS	48
6	ZTR	0
13	ZTR	0
8	ZTR	0
6	ZTR	0
13	IZ	0

13	ZTR	0
8	ZTR	0
10	Z	0
13	Z	0
3	Z	0
6	Z	0
0	Z	0
47	POS	48
3	Z	0
0	Z	0

經歷第一回合後，四個重要係數都被找到，而這四個重要係數的重建工作也完成。然而接下則是作重要值修正工作。如小波係數值 57 是介於 32 及 64 之間，而修正工作是以 $(32+64)/2=48$ 為界線，當 57 大於 48 時，則輸出布林代數 1，反之則為 0，而修正值為 $40=(32+48)/2$ 或 $56=(64+48)/2$ ，故可得表 4.2 之關係。

表 2.2 重要係數修正表(第一回合)。

小波係數	布林代數	修正值
57	1	56
-34	0	-40
47	0	40
47	0	40

由表 2.1 及 2.2 可得，第一回合之輸出 $DP(1)=POS,NEG,IZ,ZTR,POS,ZTR,ZTR,ZTR,IZ,ZTR,ZTR,ZTR,ZTR,ZTR,ZTR,ZTR,POS,ZTR,ZTR$ 。若 POS 以二位元 00 編碼代表，

NEG 以 01、IZ 以 10、ZTR 以 11 代表，則 DP(1)可表示為式(2.7)

$$DP(1)=00011011001111111110111111111111001111 \quad (2.7)$$

而修正輸出 RP(1)可表示為式(2.8)

$$RP(1)=1000 \quad (2.8)$$

完成了式(2.7)和式(2.8)後，圖 2.13 中內的四個重要係數也完成編碼。此時，第二回的編碼其輸入係數矩陣為，第一回合之中重要係數之重建修正後之差額放回對應位置所得之矩陣，如圖 2.16 所示，其圈圈內則為第一回合之重要係數。

1	6	7	6	10	13	-10	6
-31	27	13	8	3	6	8	-2
6	13	0	13	3	7	2	-10
13	8	13	8	6	2	2	-2
8	-6	0	7	1	5	-2	6
2	0	3	0	3	-2	0	6
4	-6	-2	3	0	7	2	2
10	4	7	-2	2	-3	3	2

圖 2.16 第二回合前輸入係數矩陣。

而其新門檻值 T 則降為 $16(=32/2)$ 外，其餘動作如第一回合編碼。在第二回合編碼同時，為了將第一回合之重要係數作更進一步修正，所以分別又在(48,64)及(30,48)區間中各取中間值 56 及 40 當判斷準則，如重要係數 57 是落於(48,64)中且大於 56 則布林代數輸出為 1，反之為 0;重要係數-34 之絕對值落於(32,48)區間中但小於 40，故布林代數

為 0；重要係數 47 落於(32,48)中且大於 40，所以布林代數為 1。所以第二回合，除如同第一回合之修正輸出(10)外還須增加四個重要係數修正輸出(1011)，以此類推。

2.1.4 算術編碼

算術編碼法與霍夫曼編碼法[18]都是屬於無失真資料壓縮，亦即壓縮過的資料可以完整重建成原來的模樣。這兩種方法所依據的理論基礎相當類似，都是利用出現機率較高的符號，使用較少位元的字碼來表示，出現機率較低的符號，則使用較多位元的字碼來表示。

霍夫曼編碼法是利用二元編碼樹的方式來編碼資料，它會將整個編碼空間以最佳整數個位元的方式來分配。而算術編碼則是用一個實數來表示要壓縮的資料，換句話說，它將整個編碼空間以精確至小數的方式來分配，以達到更佳的編碼效果。算術編碼法最大的特點在於不需要每個符號都編碼並且送出結果(霍夫曼法每個輸入符號都要編碼成整數個位元)，算術編碼用一個介於 0 到 1 之間的實數來代表編碼到目前的情形，當輸入符號持續進入編碼時，這個實數會跟著改變，由於存在 0 到 1 之間的實數共有無限多個，因此任何待壓縮檔案都能用一個獨一無二的實數來代表它。

假設資料源中，一共含有 n 個不同的輸入符號，輸入符號集 A 可表示成 $A = \{s_1, s_2, \dots, s_n\}$ ，

累積密度函數(cumulative density function) C 定義如下：

$P(s_k)$ 表示第 k 個來源符號的機率值，且 $C(k)$ 表示前面 k 個來源符號的機率值累積總和
算術編碼演算法：(假設資料源中，含有 n 個不同的符號)

步驟一：計算資料源中每一個符號的出現機率，並求符號的累積機率值。

步驟二：令回合計數 $i=0$ ，初始編碼區間定為(大於等於 0 但小於 1)，其中代表初始編碼區間的下限(lower limit)，則代表初始編碼區間的上限(upper limit)。

步驟三： $i=i+1$ ，若 x 代表目前要編碼的輸入符號，依照符號們的機率值來分割目前的編碼區間，並且找出輸入符號集 A 內和 x 相等的那個符號。假設是符號 s_k ，則將 s_k 對應的符號機率區間當成一個新的編碼區間。

實際計算新編碼區間的過程，可用下式來實現：

代表讀入此回合輸入符號 x 前的編碼區間下限，則代表讀入此回合輸入符號 x 前的編碼區間上限。代表 s_k 前面 $k-1$ 個來源符號的累積機率值，代表前面 k 個來源符號(含 s_k) 的累積機率值。

步驟四：重複步驟三，每次讀入一個輸入符號，並根據該符號適當地調整編碼區間，一直到所有輸入符號都處理完畢為止。

步驟五：從最後的編碼區間內任選出一個實數值，稱為標籤(label)，即，將標籤值當成編碼的結果輸出。

2.2 資料探勘

資料探勘是指從大量資料中擷取出有意義的資訊，以便達成有效業務統計的一種資料分析方法。資料探勘可以劃分為五種功能、六種模式分別如表 2.3 所示。

表 2.3 資料探勘分類表。

項次	主要功能	主要模式
一	分類(classification)	分類(classification)
二	推估(estimation)	回歸(regression)
三	預測(prediction)	時間序列(time series)
四	關聯分組(affinity grouping)	同質分群(clustering)
五	同質分組(clustering)	聯合(association)
六	無	序列(sequence)

在表 2.3 中，其各項項次我們分以兩段來依序介紹其定義，第一段為主要功能(分類、推估、預測、關聯分組、同質分組)各項次的定義，第二段為主要模式(分類、回歸、時間序列、同質分群、聯合、序列)各項次的定義。

主要功能

1.分類

依照分析的對象的屬性來劃分定義，建立類組。例如將貸款人信用風險屬性，區分為高度風險貸款者、中度風險貸款者、低度風險貸款者。使用的技巧有使用的技巧有決策樹(decision tree)，記憶基礎推理(memory - based reasoning)等。

2.推理

根據既有連續性數值之相關屬性資料，以獲致某一屬性未知之值。例如按照信用申請者之教育程度、行為別來推估其信用卡消費量。使用的技巧包括統計方法上之相關分析、迴歸分析及類神經網路方法。

3.預測

根據對象屬性之過去觀察值來推估該屬性未來之值。例如由顧客過去之刷卡消費量預測其未來之刷卡消費量。使用的技巧包括迴歸分析、時間數列分析及類神經網路方法。

4.關聯分組

從所有物件決定那些相關物件應該放在一起。例如超市中相關之盥洗用品(牙刷、牙膏、牙線)，放在同一間貨架上。在客戶行銷系統上，此種功能係用來確認交叉銷售(cross selling)的機會以設計出吸引人的產品群組。

5.同質分組

將異質母體中區隔為較具同質性之群組。同質分組相當於行銷術語中的區隔化(segmentation)，但是，假定事先未對於區隔加以定義，而資料中自然產生區隔。使用的技巧包括 k-平均值法及結塊(agglomeration)法。

主要模式

1.分類

是根據一些變數的數值做計算，再依照結果作分類。(計算的結果最後會被分類為幾個少數的離散數值，例如將一組資料分為可能會回應或是可能不會回應兩類)。分類 常常

被用來處理如前面說到的郵寄對象篩選的問題。我們會用一些已經分類的資料來研究它們的特徵，然後再根據這些特徵對其他未經分類或是新的資料做預測。這些我們用來尋找特徵的已分類資料可能是來自我們的現有的歷史性資料，或是將一個完整資料庫做部份取樣，再經由實際的運作來測試；譬如利用一個大的郵寄對象資料庫的部份取樣來建立一個分類模組，以後再利用這個模組來對資料庫的其他資料或是新的資料作預測。

2. 回歸

是使用一系列的現有數值來預測一個連續數值的可能值。

3. 時間序列

與回歸很像，只是它是用現有的數值來預測未來的數值。時間序列不同點在於它所分析的數值都與時間有關。時間序列的工具可以處理有關時間的一些特性，譬如時間的階層性（例如每個禮拜五個或六個工作天）、季節性、節日、以及其他的一些特別因素如過去與未來的關連性有多少。



4. 同質分組

是將資料分為幾組，其目的是要將組與組之間的差異找出來，同時也要將一個組之中的成員的相似性找出來。同質分組與分類不同的是，你不曉得它會以何種方式或根據什麼來分類。所以你必須要有一個分析師來解讀這些分類的意義。

5. 聯合

是要找出在某一事件或是資料中會同時出現的東西。聯合主要是要找出下面這樣的資訊：如果項目 A 是某一事件的一部份，則項目 B 也出現在該事件中的機率有 X %。（例如：如果一個顧客買了低脂乳酪以及低脂優酪乳，那麼這個顧客同時也買低脂牛奶的機率是 85%。）

6. 序列

與聯合關係很密切，所不同的是序列中相關的項目是以時間區分開來（例如：如果做了 X 手術，則 Y 病菌在手術後感染的機率是 45%。又例如：如果 A 股票在某一天上漲 12%，而且當天股市加權指數下降，則 B 股票在兩天之內上漲的機率是 68%）。

第三章

新體積型影像壓縮方法

影像壓縮標準可追溯 1980 年 CCITT 所建立的國際數位傳真 Group 3 及 Group 4 編碼標準。而影像壓縮的形成，起初是為了解決影像傳遞時，所必須消耗的大量儲存空間及傳送的時間，而衍生出來的技術。近年來壓縮技術的進步，使人們在壓縮影像的同時也嘗試把影像應用於各方面，由第一章的探討可以知道目前最熱門的影像技術，莫過於如何透過一些影像處理把整個物體重建成為一個 3D 模型。而經由 2D 影像重建出來的 3D 模型(如醫學的斷層掃描影像)，皆會受 2D 影像本身所含的資訊量的多寡而有所差異。但一個品質很好的 3D 模型，往往所需的 2D 影像張數大都是一個蠻龐大的數目。故本論文設計出一個體積型彩色影像壓縮方法，可以依照使用者的需求、而製作出不同品質的影像，詳細的內容將在此章節說明。

在章節 3.1 中會概述單張影像編碼的流程。在 3.2 節中會介紹影像前處理。在 3.3 節會探討編碼分類及排列方式。3.4 節會討論位元圖組選取方法。3.5 節中會介紹影像係數值編碼方法。在 3.6 節中會介紹體積型影像系統位元圖組更新的方法。

3.1 影像編碼概述

圖 3.1 為單張影像編碼的架構，在流程圖 3.1 中，第三階小波轉換區域以 2×2 方式進行切割，其餘部份以 4×4 方式進行切割，令第三階小波係數區域稱為 X 集合，第一、二階小波係數區域稱為 Y 集合，其 X 與 Y 集合定義如下：

X 集合: $X = \{LL3, LH3, HL3, HH3\}$ 。

Y 集合: $Y = \{LH2, HL2, HH2, LH1, HL1, HH1\}$ 。

在定義完 X 與 Y 集合後，將 X 集合內全部元素切割成 2×2 係數方塊，並將 Y 集合內全部元素切割成 4×4 係數方塊，之後再依 3.3 節所定義之分類原則分為三類並分別處理，其中係數編碼方式將於 3.5 節中介紹。而在流程圖中的中度相關之係數方塊所得之位元圖，將會依 3.4 節所介紹方式進行篩選。

而本論文所提出之 WT-BTC 壓縮演算法其編碼的流程大致可以分成五個部份。第一部份為影像前處理。第二部份為小波轉換後係數分類。第三部份為選取適當的位元圖組。第四部份為係數編碼。第五部份為位元圖組更新。

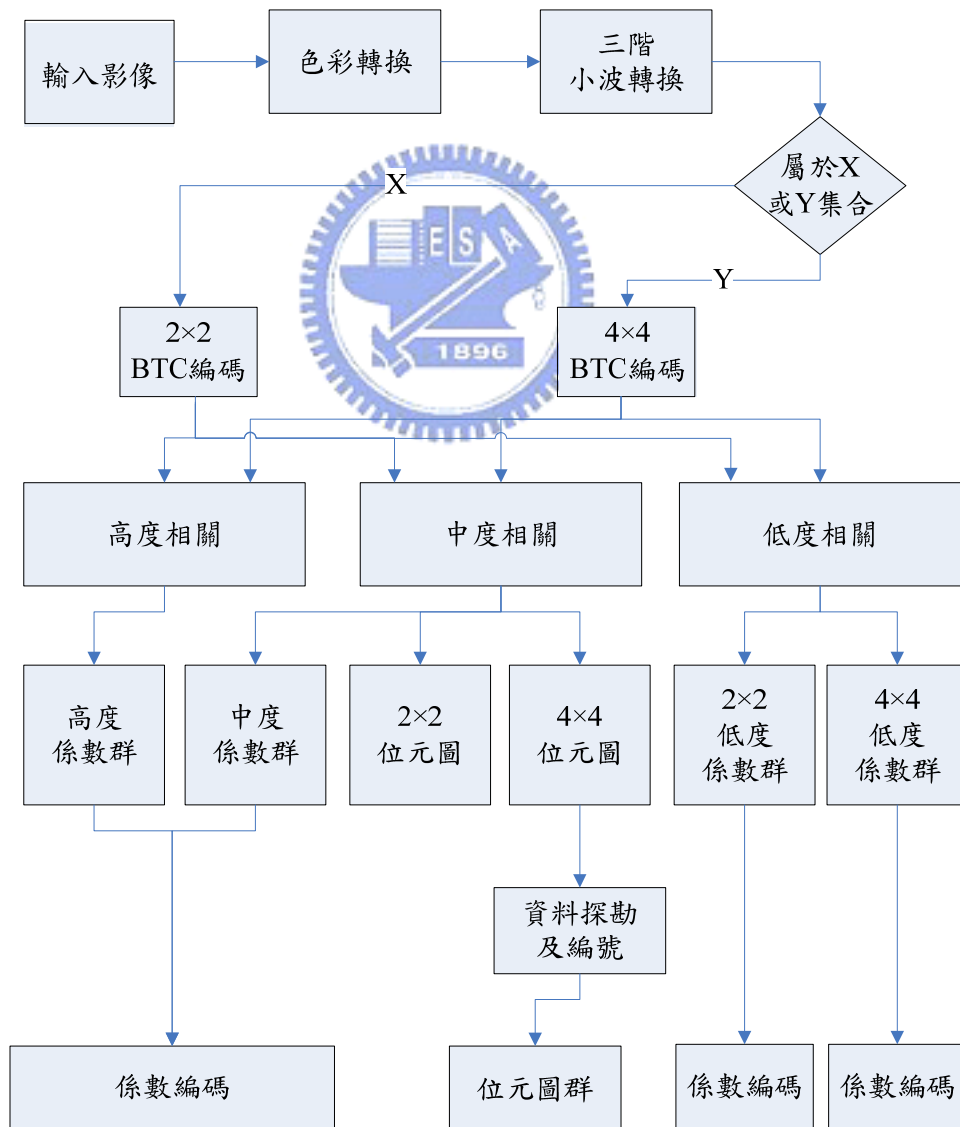


圖 3.1 WT-BTC 系統架構示意圖。

3.2 影像前處理

彩色影像一般皆為 RGB 系統，而本論文為了減少彩色影像的資料量，所以在論文中是使用 YCbCr 系統來處理輸入的彩色影像。在 YCbCr 系統中，Y 代表灰階亮度，Cb 和 Cr 代表彩度，而任兩種彩色系統之間都存在聯繫彼此的數學關係[19]。在 YCbCr 系統中，由於人眼對亮度的敏感度高於彩度(如圖 3.2(b)、(c)、(d))，所以在取樣時可以依 4:2:2 的方式對 YCbCr 取樣，如此一來，彩色影像的資料量相對也減少許多。



圖 3.2 原影像與 YCbCr 影像對照圖。(a) 原圖。(b) Y 平面。(c) Cb 平面。(d) Cr 平面。

3.3 小波係數分類

在本節中會分兩部分來介紹，首先在 3.3.1 節中會介紹小波係數切割的方式，並說明選用此種切割方式之原因，在 3.3.2 節中會介紹切割完後之係數方塊分類及處理方式。

3.3.1 小波係數切割

在原始影像經色彩轉換及三階 Haar 小波轉換後，所得之係數矩陣依流程圖 3.3 所示進行切割，在第三階小波係數 X 集合時，採用 2×2 的切割方式，在 Y 集合時，則採用 4×4 的切割方式進行切割，其如此切割之主要原因如下的例子作說明。以 512×512 的 LENA 圖來當作輸入影像，第三階小波係數若採用 4×4 的切割方式來進行切割，雖然以大方塊的切割方式可以減少系統在處理係數方塊的數量，但也因為所切割的方塊越大，方塊內的係數的差異也越大，故導致該係數方塊原若以 2×2 的切割方式，可以使四個小方塊分別落於高度及中度相關之分類，但採用 4×4 切割後該方塊卻必須被歸類到低度相關之分類，而並無節省到空間如表 3.1 所示，然而在第一、二階小波係數所採取的切割方式也因上述之因素而有所影響，故本論文才選用圖 3.3 所示之方式進行切割。

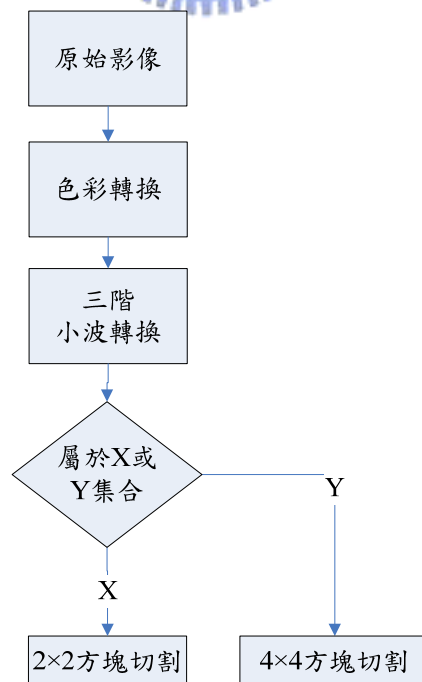


圖 3.3 小波係數切割之流程圖。

表 3.1 第三階小波係數之切割方式與 PSNR 關係表。

小波係數範圍	2×2 切割	4×4 切割
LL3(個)	3635	4096
LH3(個)	2698	3268
HL3(個)	2998	3715
HH3(個)	2452	2991
PSNR(dB)	42.62	40.61

3.3.2 小波係數分類與處理

在上述之切割完成後，其每一個係數方塊之最大值則稱為 Ma 、最小值為 Mi 。當取得係數方塊之最大值及最小值後，其餘動作依表 3.2 內所述，依其相關度定義分類，而在表 3.2 相關度定義中的中度及低度相關定義項目所描述之 A 與 B 集合其定義如下：

A 集合:係數方塊內大於 $(Ma-12)$ 值之係數集合。

B 集合:係數方塊內小於等於 $(Mi+12)$ 值之係數集合。

在表 3.2 內的係數方塊之處理方式項目中，我們將以高度相關影像圖 3.4、中度相關影像圖 3.6 與低度相關影像圖 3.10 當例子來說明各種相關係數方塊之處理方式。

表 3.2 小波轉換係數分類表。

項目 \ 類別	高度相關	中度相關	低度相關
係數方塊之相關度定義	$(Ma-Mi) \leq 12$	$(Ma-Mi) > 12$ 且 A、B 集合內個數和等於方塊內個數和	$(Ma-Mi) > 12$ 且 A、B 集合內個數和不等於方塊內個數和
係數方塊之處理方式	方塊平均值取代	建立位元圖 亮、暗部平均值取代	若是 2×2 係數方塊 轉換成 1×4 向量 若是 4×4 係數方塊 轉換成 1×16 向量

一. 高度相關

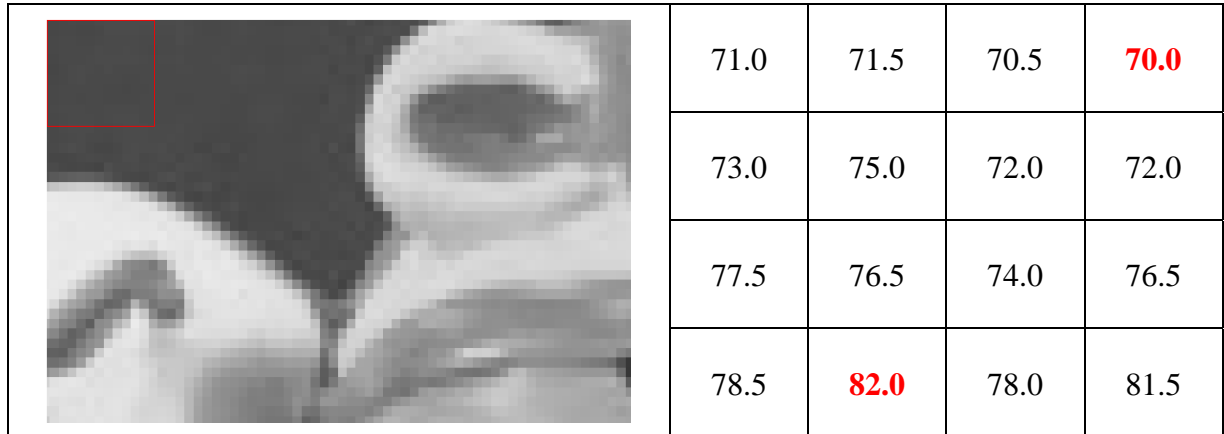


圖 3.4 高度相關之 BTC 係數圖。

由高度相關影像圖 3.4 知，左為 Y 平面之原始圖片、右為圖片中其紅色方塊經 Haar 小波函數運算後之係數。從右邊表知其最小係數為 **70.0**、最大係數為 **82.0**，其 $(Ma-Mi)$ 為 **12** 等於臨界值 12，故選取平均值 **75.0** 方式來代表此一方塊值，如高度相關係數之重建還原圖 3.5 所示。

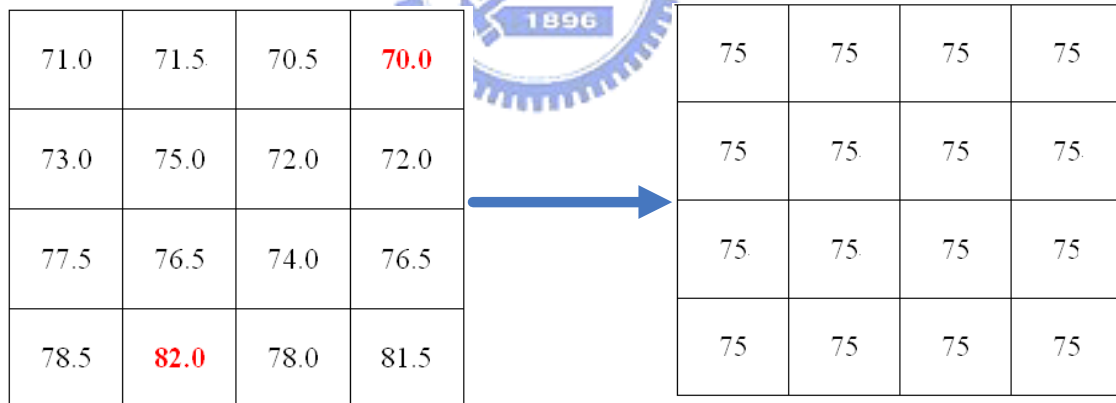


圖 3.5 高度相關係數之重建還原圖。

二. 中度相關

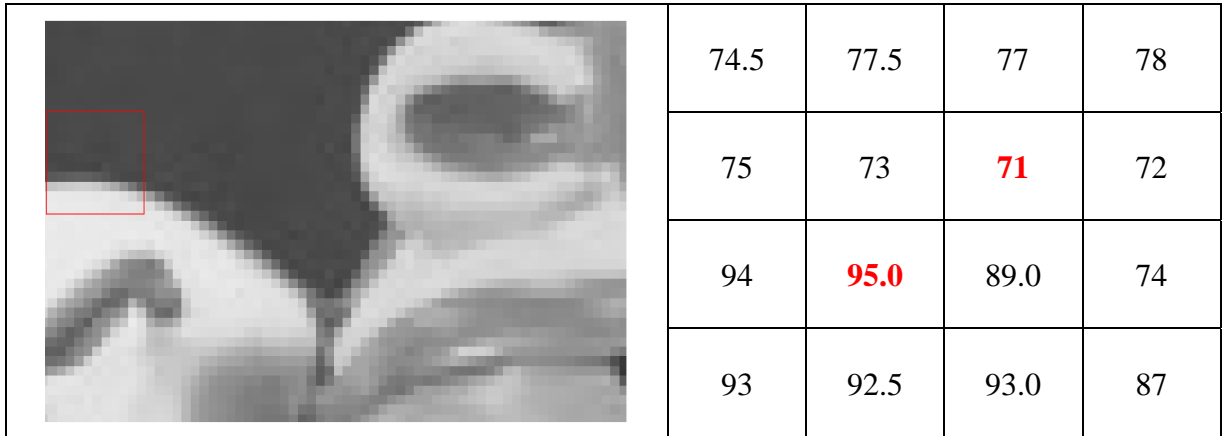


圖 3.6 中度相關之 BTC 係數圖。

由中度相關影像圖 3.6 中可得知，其 BTC 編碼之方塊內最小轉換係數值為 71.0、最大為 95.0 其 $(Ma-Mi)$ 差值為 24.0 大於 12 屬於中或低度相關，A 集合為方塊內大等於 83(=95-12)之係數群； $\{A\}=\{94、95、89、87、93、92.5、93\}$ 共 7 個。B 集合為方塊內小於等於 83(=71+12)之係數群； $\{B\}=\{74.5、77.5、77、78、75、73、71、72、74\}$ 共 9 個，A、B 集合內個數和 16(=7+9)等於 4x4 方塊和，故此為中度相關。並依中度相關之處理方式建立位元圖，其流程如圖 3.7 所示。

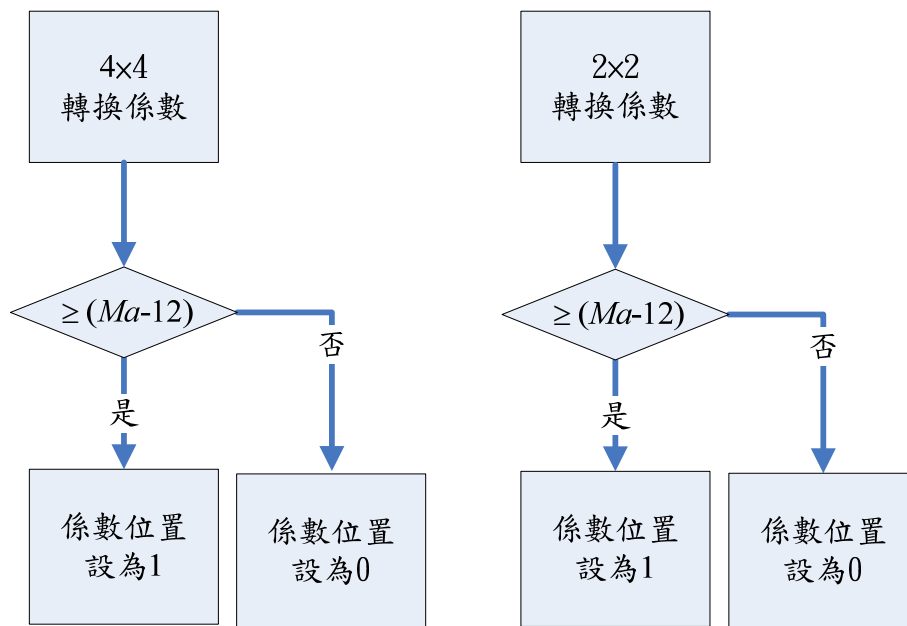


圖 3.7 2x2 及 4x4 位元圖之流程圖。

在圖 3.6 中得知，其平均值為 82.2 所以大於平均值的係數稱為亮部、反之稱為暗部，如中度相關之位元圖 3.8 所示，再經由位元圖運算可得亮部平均值 **91.9** 與暗部平均值 **74.7**，最後依方塊截短編碼的重建方式可得中度相關係數之重建還原圖 3.9 所示。

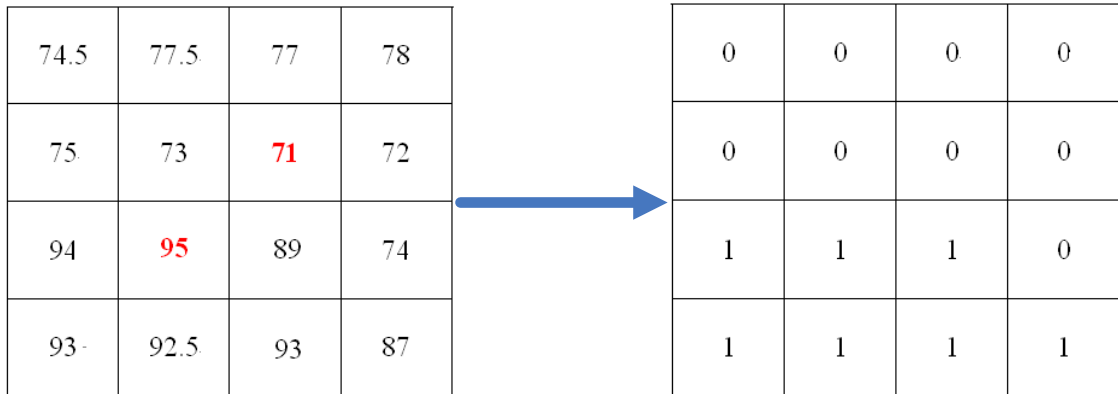


圖 3.8 中度相關之位元圖。

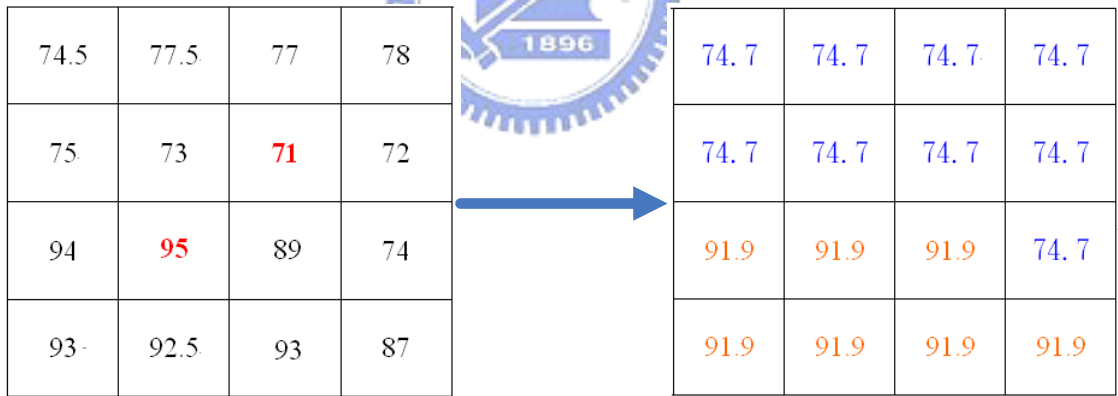


圖 3.9 中度相關係數之重建還原圖。

三. 低度相關

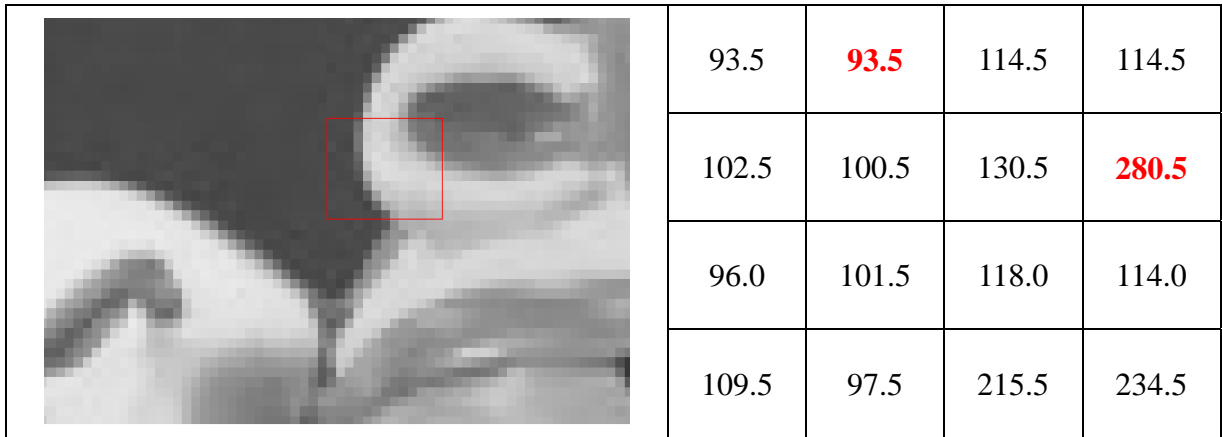


圖 3.10 低度相關之 BTC 係數圖。

由低度相關影像圖 3.10 知，其最大值為 **280.5**、最小值 **93.5** 故其 $(Ma-Mi)$ 差值為 **187**，而集合 $\{A\}=\{280.5\}$ 、集合 $\{B\}=\{93.5、93.5、96、97.5、101.5、102.5、100.5\}$ ，集合 $\{A\}$ 與 $\{B\}$ 個數和 $8(=1+7)$ 小於 **16**，所以是屬於第三類低度相關。經由表 3.2 內的低度相關係數所述之處理方式，將 4×4 係數方塊轉換成 1×16 之係數向量。



3.4 位元圖選取方式

由 3.3 節的表 3.2 知，中度相關的係數皆可分為兩部份。第一部份為亮部及暗部平均值，第二部份為位元圖。如把所有種類的位元圖都傳送到接收端時，是一種龐大且又暫空間的作法。例如 4×4 的方塊而言共有 2^{16} 個。但並不是所有的組合皆會出現，而且太多的位元圖也會導致索引表的所需的代表位元增加。所以如何有效的篩選出適當的位元圖就顯的很重要。本論文中是使用資料探勘中的聯合方式及基因演算法的適應值之輪盤式選擇觀念所衍生出的位元圖選取方式來篩選，而其資料探勘觀念如下這一例子。

例子:

TID.	Items.
100.	A, C, D, E.
200.	B, C, E.
300.	A, B, C, E.
400.	B, E.

如果設定其最低門檻值為 2

符合該條件者則稱為「頻繁樣本」

頻繁的項目: {A}、{B}、{C}、{E}、{A,C}、
{B,C}、{C,E}

由例子中可得知，頻繁樣本的數量多寡與最低門檻值有相關，所以最低門檻的值大小會影響位元圖組的數目，但在資料探勘的方法中，門檻值的決定均無一定的準則，而大致給定的方式分為兩種，第一種為直接給定門檻值 t ，第二種則是依據其樣本數給定一個參數值 z ，再由樣本數除以 z 值來決定門檻值大小。而其兩種方式均由人為所訂，所以為了解決這問題本論文使用了基因演算法的適應值之輪盤式選擇觀念來自動決定門檻值，而何謂適應值之輪盤式選擇(roulette wheel selection)，其定義如下:

輪盤式選擇

在每一代的演化過程中，首先依每個物種(字串)的適應函數值的大小來分割輪盤的位置，適應函數值越大的話，則在輪盤上佔有的面積比例也越大，每個物種在輪盤上佔有的面積比例越大代表被挑選到交配池中的機率越大，然後隨機選取輪盤的一點，其所對應的物種即被選入到交配池中。

故依上述之觀念，來說明位元圖的最低門檻值決定方式。其方法是依照位元圖各組原始出現的機率，在輪盤上分割大小，然後與壓縮圖片中各組位元圖所佔的比例，所分割的輪盤面積大小做比較後，再決定各組位元圖選取的門檻值，而本論文中所謂的位元圖組是為位元圖之集合如位元圖組之示意圖 3.11 所示。

0.	0.	0.	1.
0.	0.	0.	0.
0.	1.	0.	0.
0.	0.	0.	0.

(a)

0.	1.	0.	1.
0.	0.	0.	0.
0.	0.	0.	0.
0.	0.	0.	0.

(b)

0.	1.	0.	0.
0.	1.	0.	1.
1.	0.	0.	0.
0.	1.	1.	0.

(c)

0.	0.	1.	1.
0.	1.	1.	1.
0.	1.	0.	0.
0.	0.	0.	0.

(d)

圖 3.11 位元圖組示意圖。(a) 個數為 2 的位元圖。(b) 個數為 2 的位元圖。(c) 個數為 6 的位元圖。(d) 個數為 6 的位元圖。

圖 3.11(a)、(b)之位元圖，其為總合為 2，而在 4×4 位元圖組中，總合為 2 的組合共有 120 種，然而若序列影像中，中度相關之係數有 3 萬筆，其位元圖總合為 2 的個數又佔了 600 筆。而該類位元圖的最低門檻值則依原本該組的所佔的機率及在圖片中所佔的比重做比較，其流程如圖 3.12，決定好最低門檻值後便依上述例子，挑選位元圖群中大於最低門檻值的位元圖。

由於在取最低門檻值時，可能會發生門檻值設定太大導致一些對影像具有代表性但未符合門檻值的位元圖被遺失，故本論文在其後增加一個成長機制，來使位元圖組可以依照使用者需求來成長位元圖組，其流程圖如圖 3.13 所示。

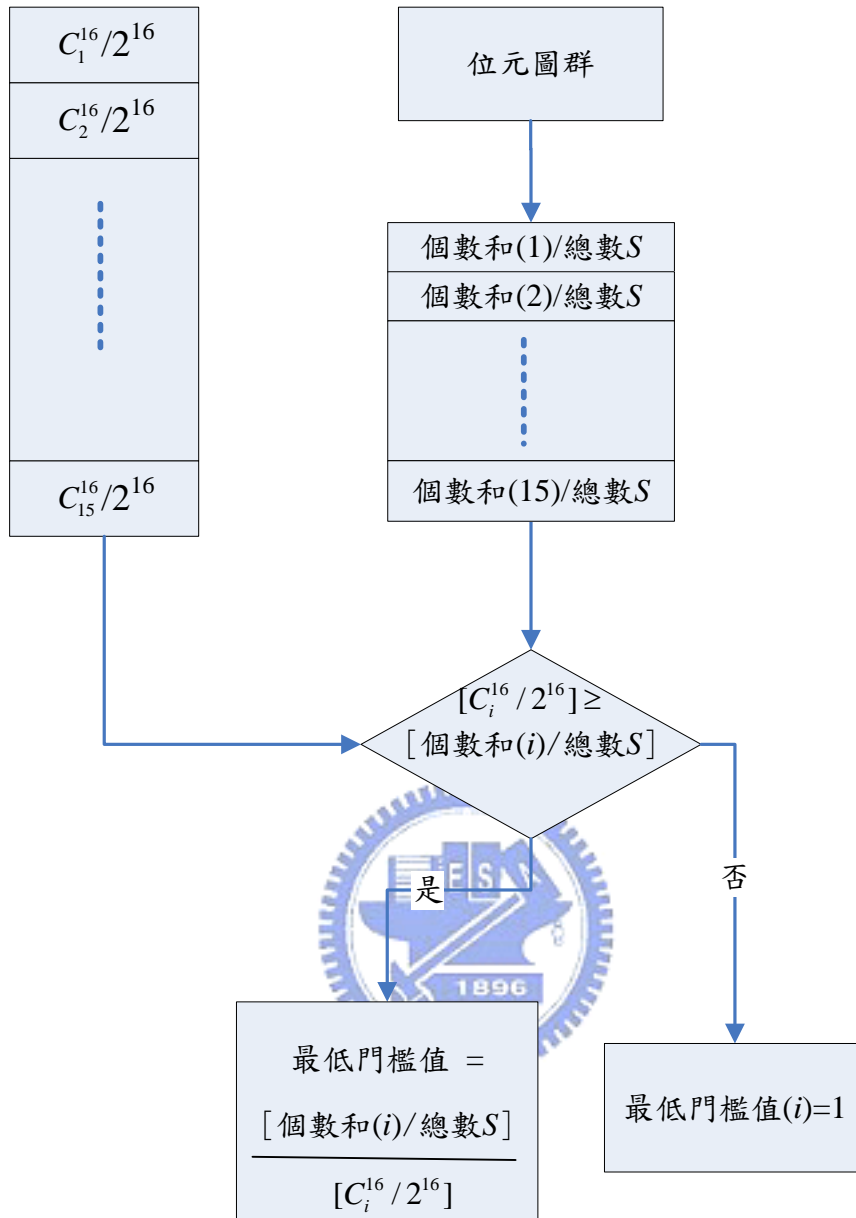


圖 3.12 決定最低門檻值的流程圖。

由其可得以下公式：

$$P_2(k) = \sum_{i=1}^m X_i^k / S, \quad m \in N, \quad k = 1, 2, 3, \dots, 15, \quad (3.1)$$

$$P_1(i) = (C_i^{16} / 2^{16}), \quad i = 1, 2, 3, \dots, 15, \quad (3.2)$$

$$P_1(k) - P_2(k) = \begin{cases} \leq 0, & \text{minimum support}=1, \\ > 0, & \text{minimum support}=\lceil P_2(k) / P_1(k) \rceil, \end{cases} \quad (3.3)$$

$k = 1, 2, 3, \dots, 15.$

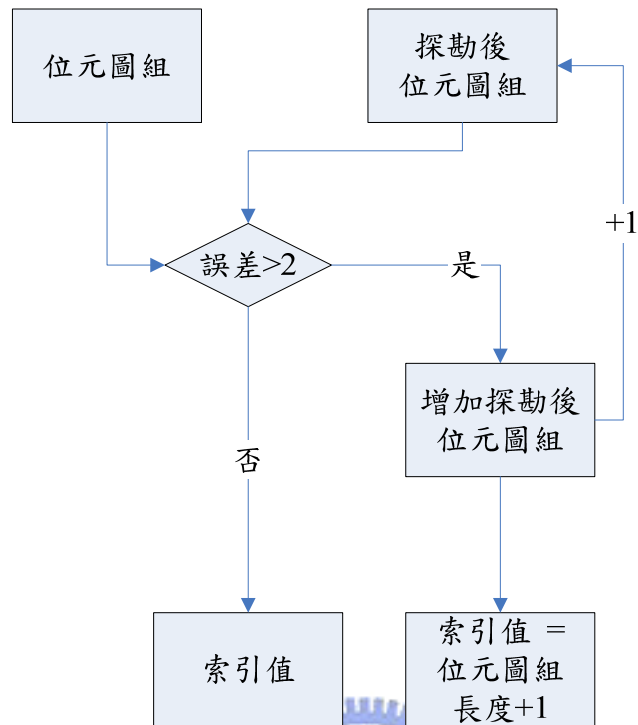


圖 3.13 位元圖組成長機制流程圖。

當該原始位元圖組在資料探勘決策後所得的位元圖組中，無法利用向量量化編碼法的搜尋編碼簿之方式找到與自身最為接近之位元圖來代表時，則會把該位元圖加入到該類探勘後的位元圖組最後一個。並把該索引值直接設定為該類探勘後位元圖組長度+1。透過此機制可以使最低門檻值所產生誤差效應降低，圖 3.14 為其示意圖。

3.5 小波係數重建編碼

在第 3.3 節中所介紹的各類相關，都有一些係數值產生，而這些係數值分別為該方塊之平均值、方塊亮部及暗部平均值或者未經處理之係數方塊原始係數值。在所有的係數值皆經四捨五入(取到整數位)後，再依圖 3.1 所示，各別分類。而我們令 A 類為高度相關及中度相關之平均值集合、B 類為低度相關的 1×4 係數向量集合與 C 類則為低度相關的 1×16 係數向量集合，各類集合如表 3.3 所示。

總和為 T
位元圖組

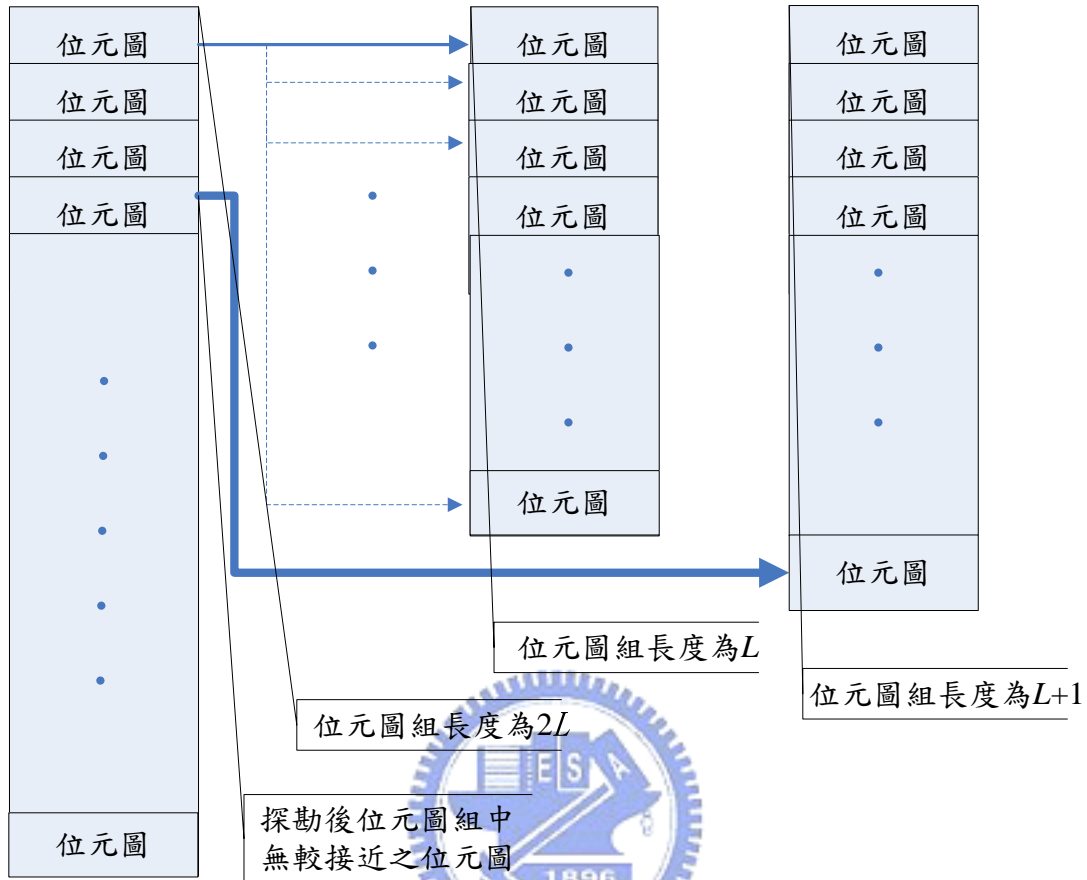


圖 3.14 位元圖組成長機制示意圖。


表 3.3 可知，高、中度相關係數為 2 行 N 列的格式表示，而低度相關係數中又分為 1×4 係數向量集合如表 3.3 中的 B 類格式表示，以及 1×16 係數向量集合如表 3.3 中的 C 類格式表示。這些 A、B 與 C 類表格均以圖 3.15 之流程處理。

在圖 3.10 中， $X_{i,j}^a$ 的 a 代表 A 類表格、 X 代表 A 類表格內的係數， i 和 j 分別為列與行，

$P_{i,j}^a$ 則為與 A 類相同大小的空表格， T^a 門檻值如公式 3.4 所示。

$$T^a = 2^{\lfloor \log(\max(X_{i,j}^a)) \rfloor} \tag{3.4}$$

表 3.3 相關係數分類規格表。

類別	A 類						B 類																																																																																																																																																																				
格式	<table border="1" data-bbox="440 360 766 936"> <tr><td>高</td><td>395</td><td>0</td></tr> <tr><td>中</td><td>40</td><td>159</td></tr> <tr><td>中</td><td>98</td><td>130</td></tr> <tr><td>高</td><td>150</td><td>0</td></tr> <tr><td>中</td><td>80</td><td>95</td></tr> <tr><td>高</td><td>470</td><td>0</td></tr> <tr><td>高</td><td>448</td><td>0</td></tr> <tr><td>中</td><td>256</td><td>311</td></tr> <tr><td>中</td><td>114</td><td>130</td></tr> <tr><td>中</td><td>18</td><td>60</td></tr> <tr><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td></tr> </table>						高	395	0	中	40	159	中	98	130	高	150	0	中	80	95	高	470	0	高	448	0	中	256	311	中	114	130	中	18	60	。	。	。	。	。	。	。	。	。	<table border="1" data-bbox="994 360 1313 936"> <tr><td>-31</td><td>-13</td><td>-5</td><td>-84</td></tr> <tr><td>-18</td><td>190</td><td>186</td><td>37</td></tr> <tr><td>1</td><td>-19</td><td>22</td><td>-5</td></tr> <tr><td>9</td><td>27</td><td>-60</td><td>-1</td></tr> <tr><td>-34</td><td>-95</td><td>8</td><td>3</td></tr> <tr><td>-7</td><td>230</td><td>220</td><td>12</td></tr> <tr><td>19</td><td>-22</td><td>-8</td><td>2</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td></tr> </table>						-31	-13	-5	-84	-18	190	186	37	1	-19	22	-5	9	27	-60	-1	-34	-95	8	3	-7	230	220	12	19	-22	-8	2	。	。	。	。	。	。	。	。	。	。	。	。																																																																																
	高	395	0																																																																																																																																																																								
	中	40	159																																																																																																																																																																								
	中	98	130																																																																																																																																																																								
	高	150	0																																																																																																																																																																								
	中	80	95																																																																																																																																																																								
	高	470	0																																																																																																																																																																								
	高	448	0																																																																																																																																																																								
	中	256	311																																																																																																																																																																								
	中	114	130																																																																																																																																																																								
	中	18	60																																																																																																																																																																								
	。	。	。																																																																																																																																																																								
。	。	。																																																																																																																																																																									
。	。	。																																																																																																																																																																									
-31	-13	-5	-84																																																																																																																																																																								
-18	190	186	37																																																																																																																																																																								
1	-19	22	-5																																																																																																																																																																								
9	27	-60	-1																																																																																																																																																																								
-34	-95	8	3																																																																																																																																																																								
-7	230	220	12																																																																																																																																																																								
19	-22	-8	2																																																																																																																																																																								
。	。	。	。																																																																																																																																																																								
。	。	。	。																																																																																																																																																																								
。	。	。	。																																																																																																																																																																								
																																																																																																																																																																											
<table border="1" data-bbox="371 1070 1401 1653"> <tr><td>0</td><td>-7</td><td>-1</td><td>1</td><td>0</td><td>-34</td><td>-13</td><td>-2</td><td>0</td><td>-14</td><td>-40</td><td>-24</td><td>0</td><td>7</td><td>-5</td><td>-25</td></tr> <tr><td>0</td><td>-11</td><td>0</td><td>79</td><td>0</td><td>-14</td><td>64</td><td>43</td><td>0</td><td>97</td><td>47</td><td>-4</td><td>0</td><td>171</td><td>5</td><td>-3</td></tr> <tr><td>1</td><td>4</td><td>9</td><td>19</td><td>1</td><td>1</td><td>1</td><td>3</td><td>-8</td><td>1</td><td>1</td><td>0</td><td>-34</td><td>-16</td><td>-4</td><td>2</td></tr> <tr><td>-1</td><td>-36</td><td>-31</td><td>-13</td><td>8</td><td>3</td><td>-12</td><td>-36</td><td>8</td><td>4</td><td>-5</td><td>-2</td><td>4</td><td>1</td><td>8</td><td>3</td></tr> <tr><td>-7</td><td>-2</td><td>11</td><td>78</td><td>-4</td><td>7</td><td>87</td><td>28</td><td>11</td><td>91</td><td>35</td><td>-6</td><td>80</td><td>30</td><td>1</td><td>-2</td></tr> <tr><td>39</td><td>-6</td><td>-1</td><td>-6</td><td>-5</td><td>-5</td><td>-9</td><td>-7</td><td>-7</td><td>-4</td><td>-12</td><td>4</td><td>-2</td><td>-7</td><td>-2</td><td>1</td></tr> <tr><td>-1</td><td>5</td><td>-12</td><td>7</td><td>3</td><td>-6</td><td>-6</td><td>17</td><td>0</td><td>-1</td><td>5</td><td>9</td><td>7</td><td>0</td><td>0</td><td>-14</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td></tr> <tr><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td><td>。</td></tr> </table>												0	-7	-1	1	0	-34	-13	-2	0	-14	-40	-24	0	7	-5	-25	0	-11	0	79	0	-14	64	43	0	97	47	-4	0	171	5	-3	1	4	9	19	1	1	1	3	-8	1	1	0	-34	-16	-4	2	-1	-36	-31	-13	8	3	-12	-36	8	4	-5	-2	4	1	8	3	-7	-2	11	78	-4	7	87	28	11	91	35	-6	80	30	1	-2	39	-6	-1	-6	-5	-5	-9	-7	-7	-4	-12	4	-2	-7	-2	1	-1	5	-12	7	3	-6	-6	17	0	-1	5	9	7	0	0	-14	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。
0	-7	-1	1	0	-34	-13	-2	0	-14	-40	-24	0	7	-5	-25																																																																																																																																																												
0	-11	0	79	0	-14	64	43	0	97	47	-4	0	171	5	-3																																																																																																																																																												
1	4	9	19	1	1	1	3	-8	1	1	0	-34	-16	-4	2																																																																																																																																																												
-1	-36	-31	-13	8	3	-12	-36	8	4	-5	-2	4	1	8	3																																																																																																																																																												
-7	-2	11	78	-4	7	87	28	11	91	35	-6	80	30	1	-2																																																																																																																																																												
39	-6	-1	-6	-5	-5	-9	-7	-7	-4	-12	4	-2	-7	-2	1																																																																																																																																																												
-1	5	-12	7	3	-6	-6	17	0	-1	5	9	7	0	0	-14																																																																																																																																																												
。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。																																																																																																																																																												
。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。																																																																																																																																																												
。	。	。	。	。	。	。	。	。	。	。	。	。	。	。	。																																																																																																																																																												

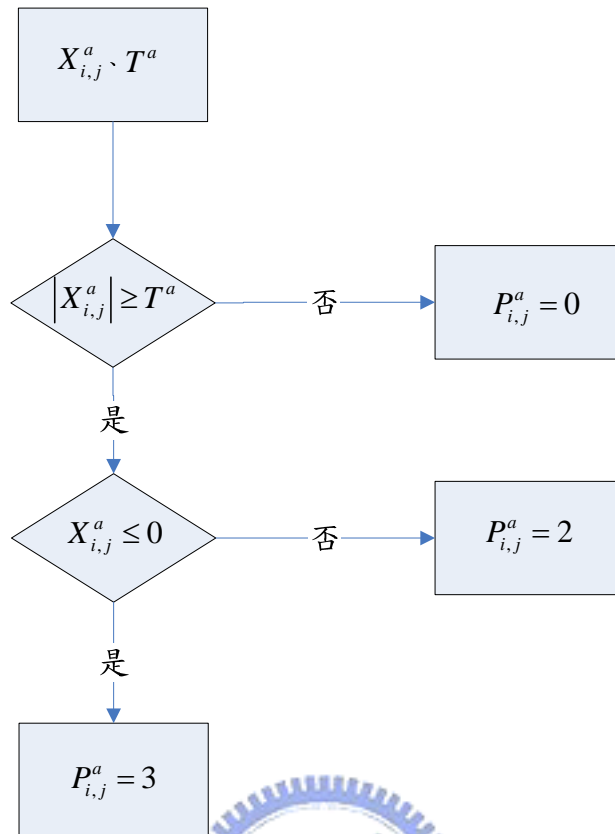


圖 3.15 重要係數重建編碼流程圖。

經過圖 3.15 之流程處理後，可得重要係數位置編碼，而其編碼輸出 DP(1)如式 3.5

$$DP(1)=000200320003220000\dots0002000, \quad (3.5)$$

在本論文中，將 DP(1)改以另外表式方式額外處理，如式 3.6、3.7

$$DP(1_0)=\{3, 2, 2, 4, \dots, 3, 3\}, \quad (3.6)$$

$$DP(1_1)=\{2, 3, 2, 3, 2, 2, \dots, 2\}. \quad (3.7)$$





而其 A、B、C 類表之重建值方式則以第二章所介紹之 EZW 演算法內之重建值方法相同。

3.6 序列影像位元圖組更新

經 3.4 節介紹可知，每張不同影像經 WT-BTC 演算法壓縮後，所產生之位元圖組數皆會不同如表 3.4 所示。

表 3.4 各影像所需位元圖組表。

(單位: 個)

	
<p>位元圖組:993</p>	<p>位元圖組:1441</p>
	
<p>位元圖組:1677</p>	<p>位元圖組:2292</p>

如將表 3.4 內之四張影像作序列傳輸時，所傳送影像之位元圖組數將為四張影像各自位元圖組數和。但由實驗得知，經重覆性計算後，傳送四張影像所需位元圖僅 **4989** 個位元圖，與不經過重覆性計算下，四張影像則需傳輸 **6403** 個位元圖，相差了 **1414** 個位元圖，所以在作體積型壓縮時，其位元圖組之更新是有其必要，因更新後的位元圖組可減少傳送的資料量，所以位元圖更新流程如圖 3.16 所示。

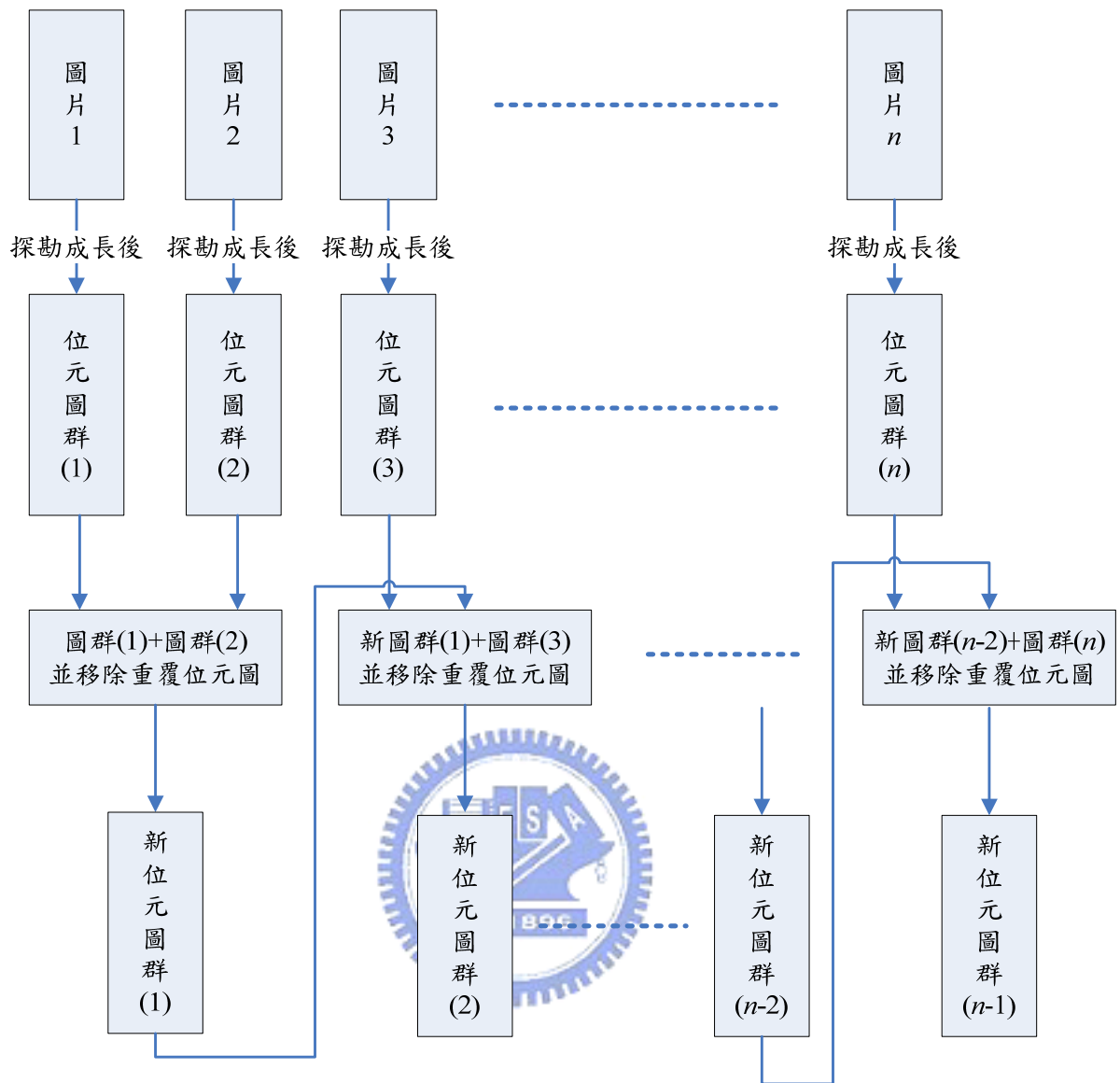


圖 3.16 位元圖組更新流程圖。

第四章

實驗結果與分析

在此章節中要討論本論文在第三章所提出的演算法來驗證，經改良式資料探勘的方法，所挑選出來的位元圖組是有代表性的且不同的小波係數分類對整體的壓縮是有效率的，在 4.1 節中會說明用來測試系統效能所採用的實驗機制，在 4.2 節中會顯示實驗的結果，在 4.3 節中會對實驗結果與問題加以討論。

4.1 實驗機制

實驗的設備以及輸入影像皆會在 4.1.1 節敘述，用來評估系統效能的方法會在 4.1.2 節說明。

4.1.1 實驗設備

本論文中的實驗影像分為五種，第一種的影像是由相機 Canon EOS Kiss Digital N 隨意拍攝取得，解析度為 1024×768 像素，24 位元的彩色影像，分 3 類，如表 4.1，共 60 張；第二種為 Google Earth 上所截取的空照圖 160 張，如圖 4.1(a)、(b)、(c)、(d)，解析度為 352×288 像素，24 位元的彩色影像；第三種為核磁共振的醫學斷層掃描影像 16 張，解析度 352×288 像素，8bpp，如圖 4.2(a)、(b)、(c)、(d)，第四種為 LENA 圖，解析度為 512×512 ，8bpp 影像，如圖 4.3，第五種為 claire-orig 影片，解析度 176×144 像素，視訊取樣大小 24bpp，框架速度 29(框架/秒)，如圖 4.4，以上輸入均在中央處理處為 Pentium(R)4 3.00GHz，記憶體為 1G MB 的個人電腦上搭配 Matlab R2007a.軟體來實現。

表 4.1 實驗影像分類表。

單純畫面	複雜畫面	有主體畫面
		
		



(a)



(b)



(c)

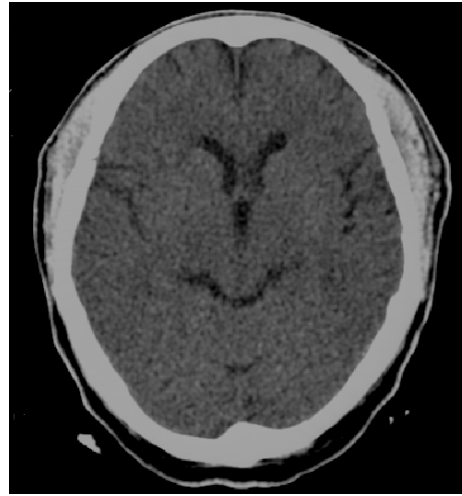


(d)

圖 4.1 Google Earth 空照圖。(a) 山脈。(b) 道路。(c) 有雲霧。(d) 都市。



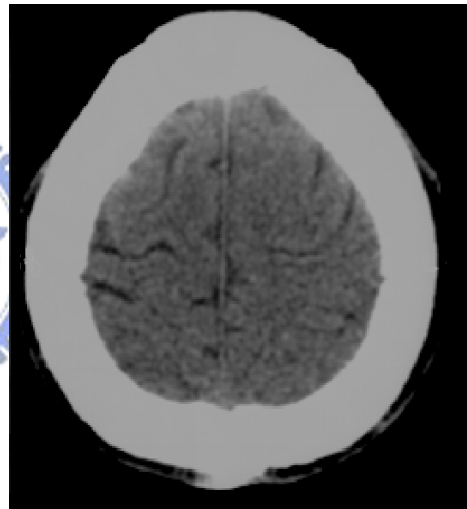
(a)



(b)



(c)



(d)

圖 4.2 核磁共振的醫學斷層掃描影像。(a) 第一層。(b) 第二層。(c) 第四層。(d) 第七層。



圖 4.3 LENA 圖。



圖 4.4 claire-orig 影片圖。

4.1.2 效能評估

首先對改良式的資料探勘與傳統式的資料探勘方法，在選取各組位元圖的最低門檻值之效能評估。其次在對 4.1.1 節所提出的三種不同實驗對象做影像壓縮效能評估。而一般影像壓縮方法對壓縮影像的評估，除了主觀部分是由主觀者的判斷外，客觀部分大多使用 *PSNR* 及位元率等方式來測試，故本論文也使用上述之兩方法來測試，但本論文所提出之演算法，除在單張影像上有其較好效能的壓縮外，其另一特性是在體積型影像上做群體壓縮。因此本論文在效能評估方面會分為兩部分，第一部分為單張影像上與其它相關演算法比較，第二部分為體積型影像上與其它演算法做比較。

4.2 實驗結果

在 4.2.1 節中我們將介紹改良式的資料探勘與傳統式的資料探勘選取方式效能的實驗結果。在 4.2.2 節中，我們將介紹單張影像壓縮時，本論文所提之 WT-BTC 演算法、2D SPIHT[19]演算法與 2D PROGRES[20]演算法之比較。在 4.2.3 節中，我們將介紹壓縮體積型影像時，本論文提出之 WT-BTC 演算法與 3D SPIHT[21]、3D PROGRES[20]兩演算法之比較。

4.2.1 改良式與傳統式資料探勘效能比較

在本節，首先說明論文中的位元圖組成長法與無成長的位元圖組之間差異以及傳統是參數大小所造成的影響。其次會再說明經過基因演算法改良的資料探勘挑選的位元圖組與一般使用給定一的參數值 z 的資料探勘方法所挑選的位元圖組，在影像上所造成的差異，其差異分兩種表示，第一種為給定不同參數值 z 之影像比較，如圖 4.5 及表 4.2 所示。第二種為採用改良式與傳統式之影像比較，如圖 4.7 及表 4.3 所示。

參數值 z ， $z \in N$ 定義如下：

$$S_j \square \{S_j \in N : TD_j / z, j = 1, 2, \dots, 14\} \quad . \quad (4.1)$$

S_j 為各組位元圖最低門檻值、 TD_j 為符合 j 值之位元圖總數、 j 為位元圖和。

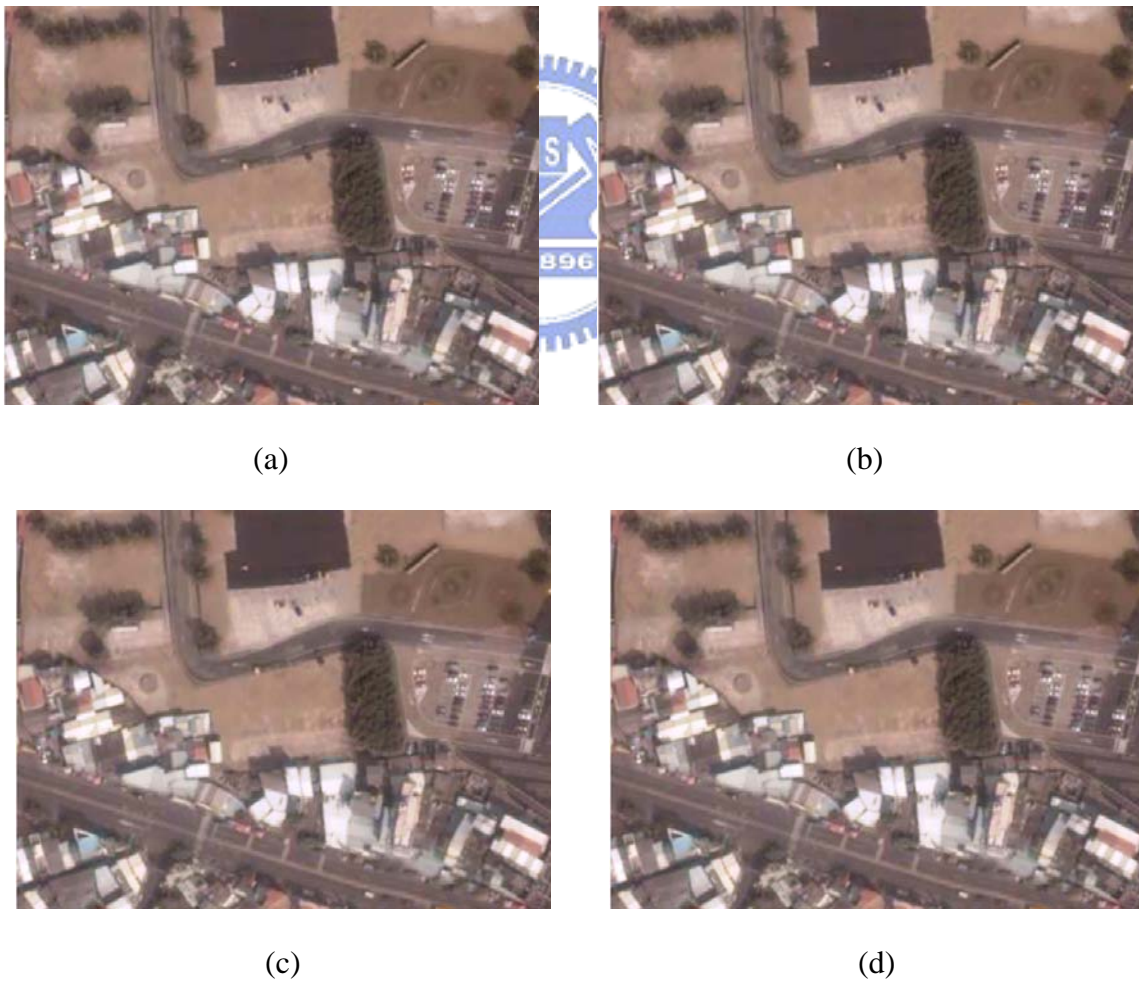


圖 4.5 第 15 張空照影像壓縮還原圖。(a) 參數值 5 無成長壓縮還原圖。(b) 參數值 5 有成長壓縮還原圖。(c) 參數值 7 無成長壓縮還原圖。(d) 參數值 7 有成長壓縮還原圖。

表 4.2 PSNR 與壓縮率比較表。

項 目	PSNR(dB)		C_R	
	無	有	無	有
參數值 5	42.360	42.916	10.538	10.496
參數值 7	42.586	42.986	10.497	10.482

由表 4.2 知，在不同參數值下其 PSNR 會有 0.226dB 的差異，但有使用成長法與無成長之間的差異為 0.556dB 與 0.4dB，不同參數與有無成長法兩者差異相比就明顯差了許多。另外可由參數值與位元圖組之關係看出參數值越大，其壓縮率越低但成長差異也越小，並且可以觀察知，參數越大其所需的位元組越多，如圖 4.6 所示。

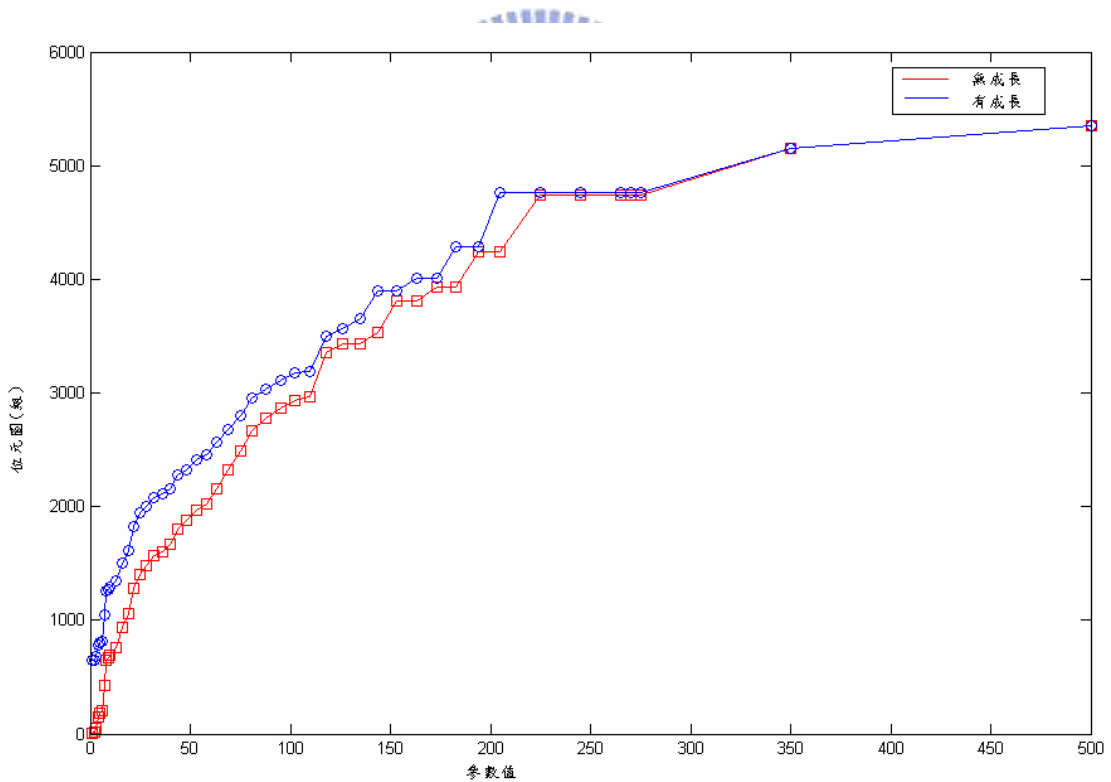


圖 4.6 參數值與位元圖組之關係圖。

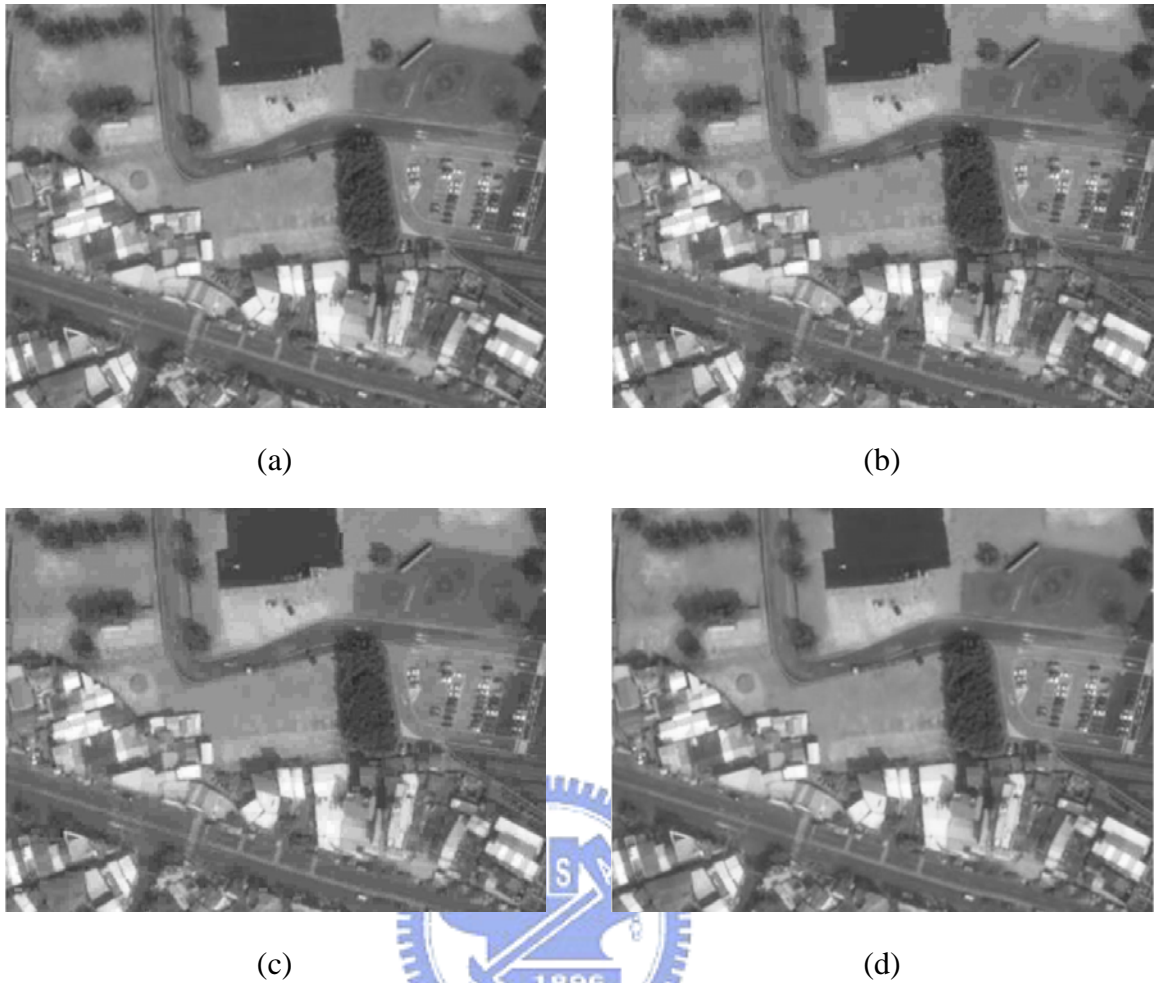


圖 4.7 第 15 張空照影像壓縮還原圖。(a) 原圖。(b) 參數值為 5 的資料探勘法壓縮還原圖。(c) 參數值為 7 的資料探勘法壓縮還原圖。(d) 改良式資料探勘法壓縮還原圖。

表 4.3 傳統式與改良式資料探勘之比較表。

項目	傳統 I(5)	傳統 II(7)	改良式
$PSNR(dB)$	42.916	42.986	45.954
C_R	10.456	10.413	10.531
原位元圖(組)	987	1129	1072
成長位元圖(組)	1191	1314	1074

由表 4.3 知，由傳統的資料探勘所得到的位元圖組，雖然可以依使用者需求而自定參數值大小，但所得的位元圖組並無有充份的取代影像中的常出現的位元圖組，所以導

致需取較多的位元圖組才能達到改良式所達到之效果。

4.2.2 單張影像效能比較

在此節我們對單張影像做效能評估。為達到公平我們選用 2D SPIHT、2D PROGRES 中所使用的 LENA 圖，解析度為 512×512 ，8bpp 影像作比較，其比較結果如圖 4.8 及表 4.4 所示。



圖 4.8 LENA 圖。(a) 原圖。(b) 2D PROGRES 壓縮還原圖。(c) 2D SPIHT 壓縮還原圖。(d) WT-BTC 壓縮還原圖。

表 4.4 單張影像之位元率與 $PSNR$ 比較表。

(單位:dB)

Bit rate	2D PROGRES	2D SPIHT	WT-BTC
0.125	26.0621	25.2954	28.6839
0.25	29.4258	28.8292	32.7212
0.5	32.8104	32.6906	36.7415
1	36.8768	37.0628	40.6211

由表 4.4 知，本論文所提出之演算法在單張影像壓縮時，與 2D PROGRES 演算法在 1bpp 時其差為 3.7443dB 與 2D SPIHT 演算法在 1bpp 時其差值為 3.5583dB，在 0.125bpp 時與 2D SPIHT 差 3.3885dB，如圖 4.9 所示。

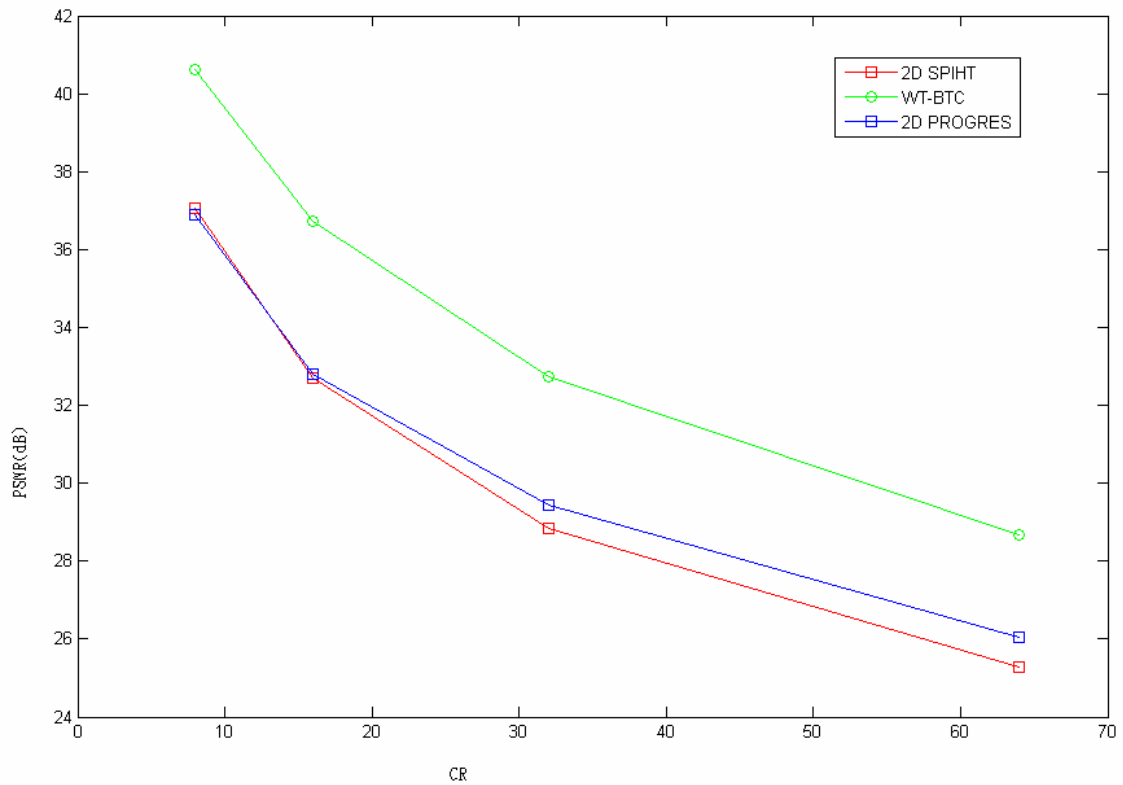


圖 4.9 單張影像壓縮之 $PSNR$ 與 C_R 關係圖。

4.2.3 體積型影像效能比較

在此節中，首先使用空照圖的序列影像 160 張做體積型影像效能評估之輸入，並與 3D SPIHT 及 3D PROGRES 兩演算法作比較，其輸出結果分別以主觀評估之影像如圖 4.10 及客觀評估之峰值信噪比與壓縮率之關係如圖 4.11 及表 4.5 所示。其次再使用核磁

共振的醫學斷層掃描影像 8 張(差異較小之 8 張影像)，當作輸入比較樣本，其輸出結果分別以主觀評估之影像如圖 4.12 及客觀評估之峰值信噪比與壓縮率之關係如圖 4.13 及表 4.6 所示，並於圖 4.14 中說明醫學體積影像的 8 張影像其峰值信噪比。在本節最後會使用 claire-orig 影片當輸入影像，而其輸出結果以影片中所有影像之峰值信噪比作為評估標準，如圖 4.15 所示。

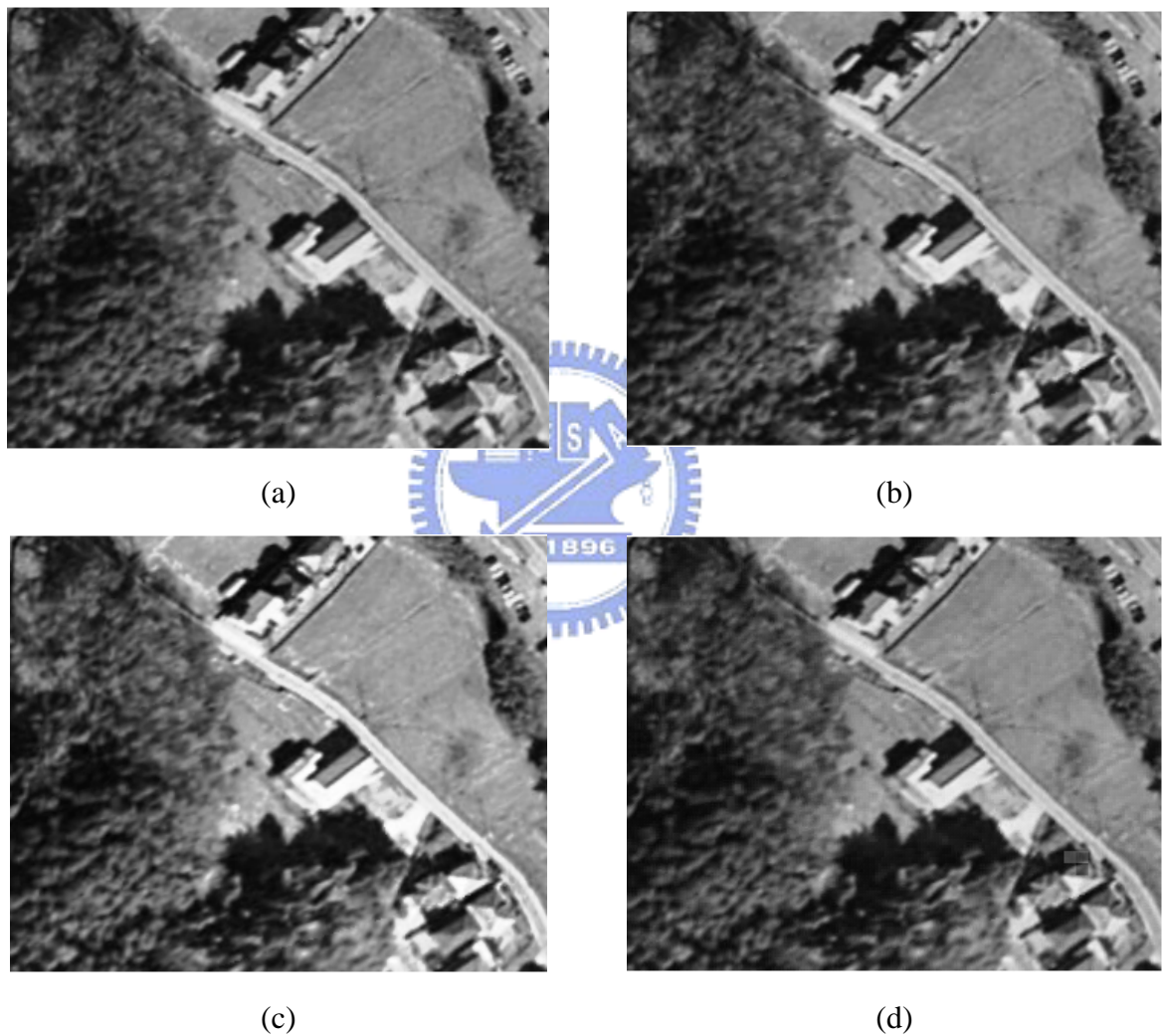


圖 4.10 第 8 張空照影像壓縮還原圖。(a) 原圖。(b) 3D SPIHT 演算法壓縮還原圖。(c) 3D PROGRES 演算法壓縮還原圖。(d) WT-BTC 演算法壓縮還原圖。

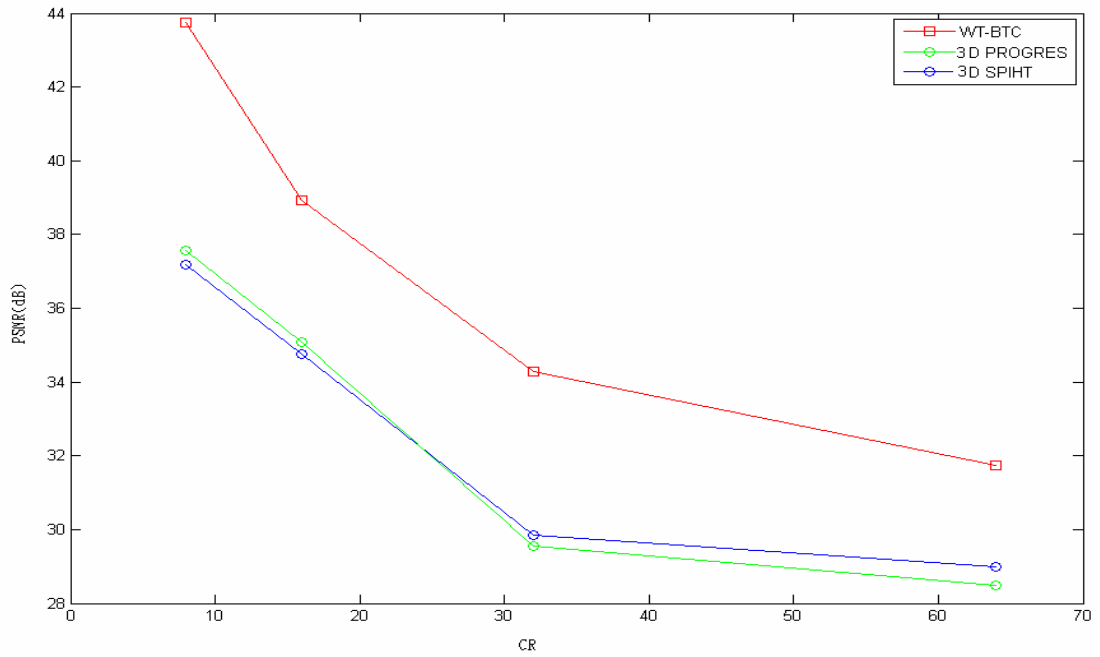


圖 4.11 體積型之空照影像的 $PSNR$ 與 C_R 關係圖。

表 4.5 空照體積型影像之位元率與 $PSNR$ 比較表。(單位: dB)

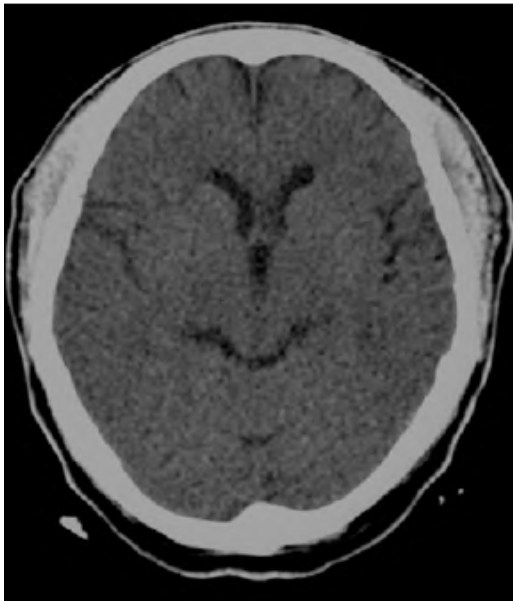
Bit rate	WT-BTC	3D SPIHT	3D PROGRES
0.125	30.667	27.489	27.683
0.25	34.824	29.978	30.010
0.5	38.418	33.576	33.344
1.0	43.754	37.174	37.576



(a)



(b)



(c)



(d)

圖 4.12 第二層醫學斷層掃描圖。(a) 原圖。(b) 3D SPIHT 壓縮還原圖。(c) 3D PROGRES 壓縮還原圖。(d) WT-BTC 壓縮還原圖。

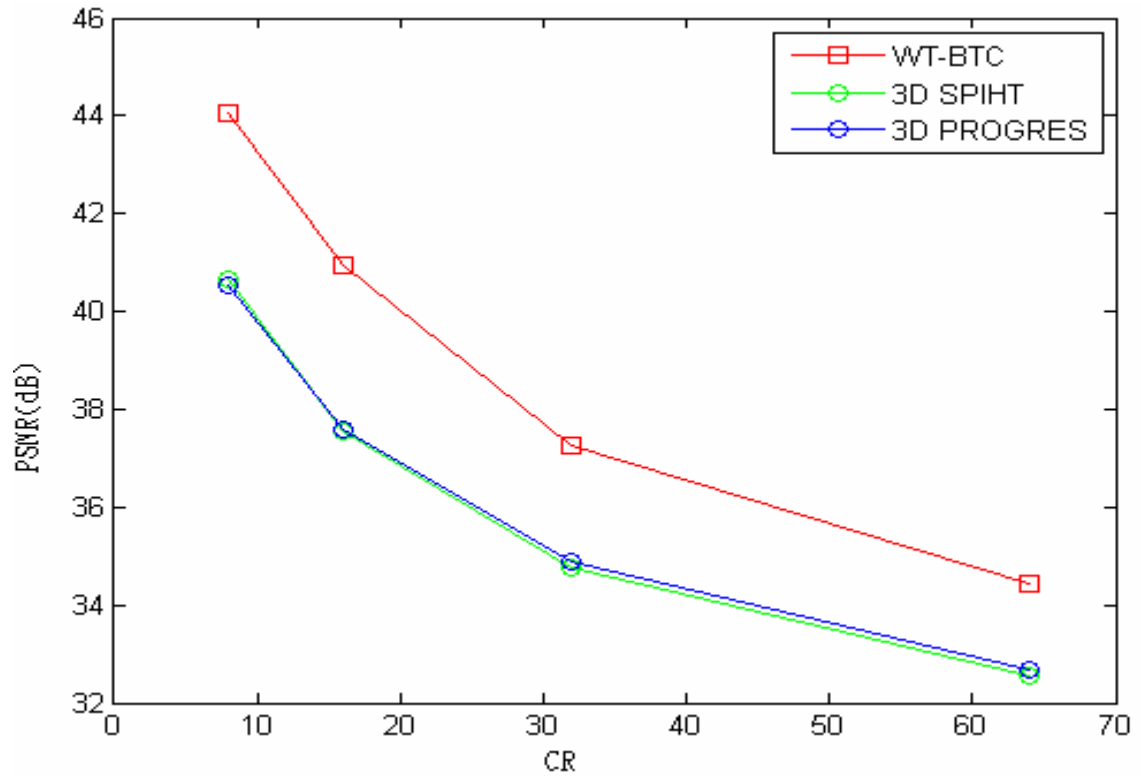


圖 4.13 體積型之斷層掃描影像的 $PSNR$ 與 C_R 關係圖。

表 4.6 醫學體積型影像之位元率與 $PSNR$ 比較表。(單位:dB)

Bit rate	WT-BTC	3D SPIHT	3D PROGRES
0.125	34.422	32.553	32.672
0.25	37.251	34.750	34.886
0.5	40.944	37.556	37.589
1.0	44.048	40.667	40.517

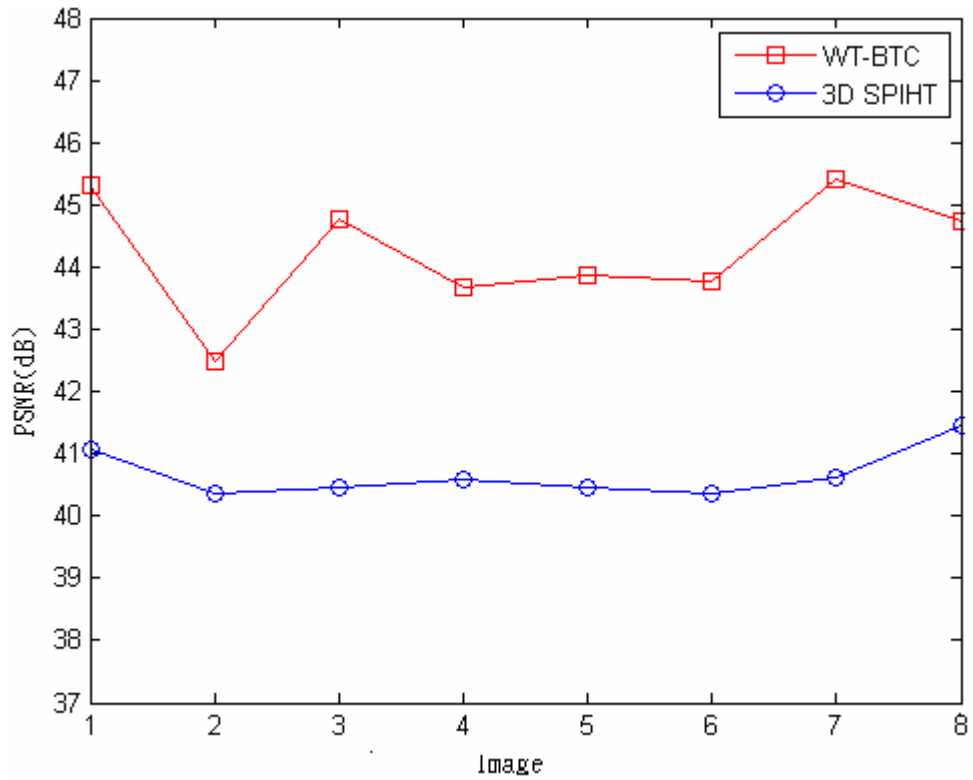


圖 4.14 體積型影像序列區間與 $PSNR$ 關係圖。

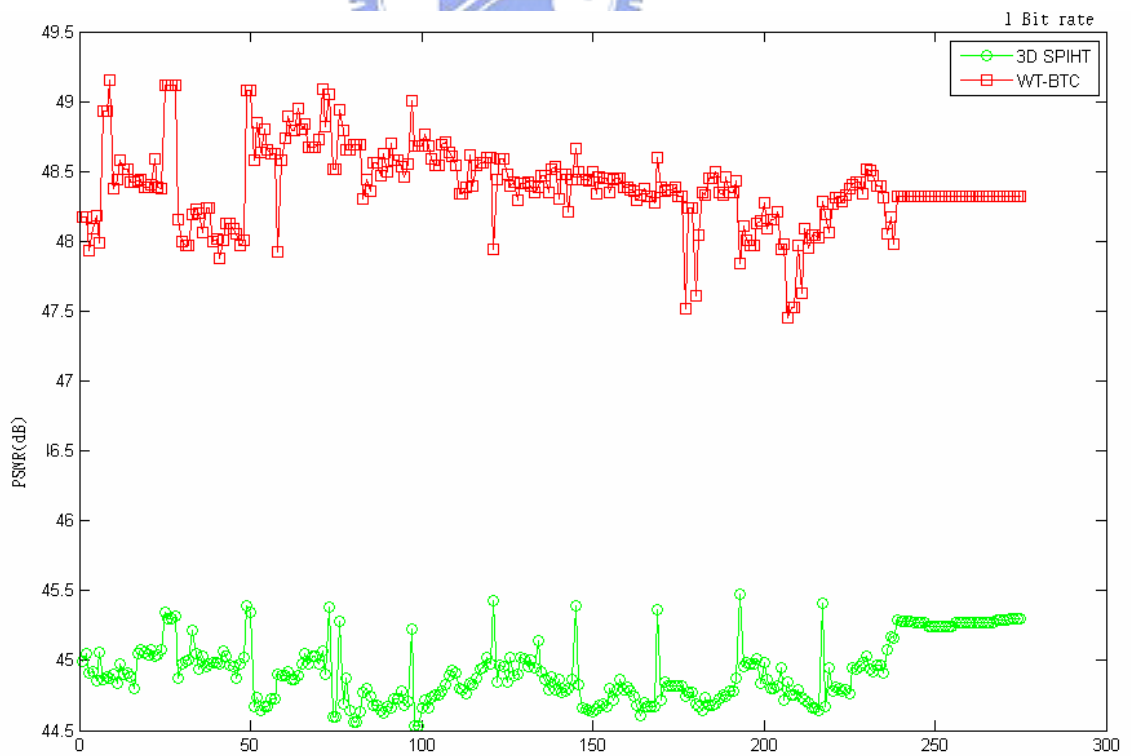


圖 4.15 3D SPIHT 與 WT-BTC 在影片中 $PSNR$ 變化圖。

4.3 實驗分析

由表 4.4 觀察可得知，本論文所提出之 WT-BTC 演算法在作影像壓縮時，在相同位元率下其 *PSNR* 均比其他演算法有明顯的差異，而造成這些差異之處，可由圖 4.16 之小波係數組合開始說明。

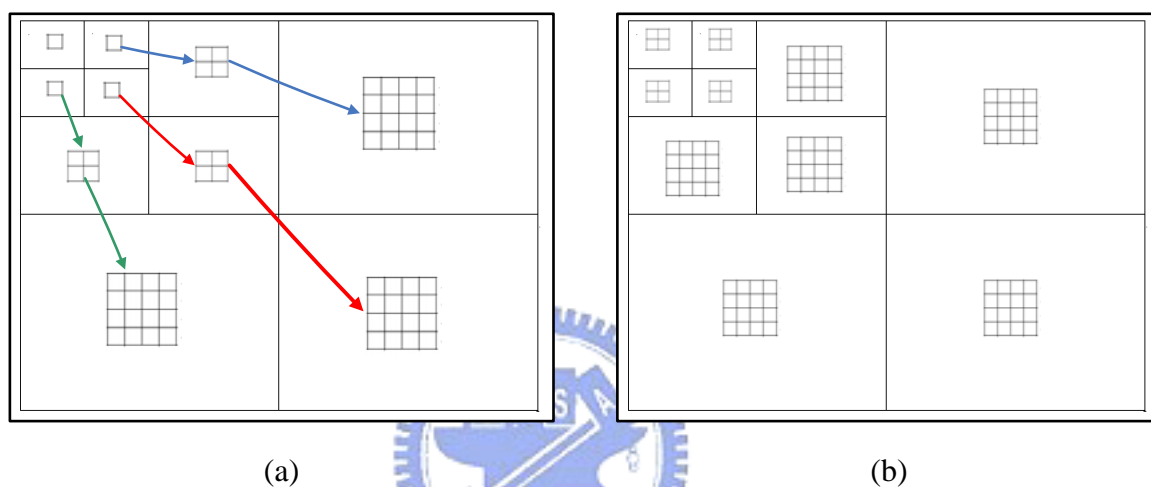


圖 4.16 三階小波係數組合圖。(a) 2D SPIHT 演算法。(b) WT-BTC 演算法。

在目前相關之小波轉換的演算法，皆是使用圖 4.16(a)之係數排列方式，而本論文所提出之演算法如圖 4.16(b)，並使用 3.3.2 節之分類處理方式處理。以 LENA 圖當做壓縮樣本時，經小波轉換後其係數未經重建還原之前的條件下，2D SPIHT 需 262144 個數字代表原圖，而本論文所提出之演算法僅需 173490 個數字就能代替，整整節省了 33% 的空間。然而在類似的係數編碼疊代方法下，就會出現在相同的位元率時，所作的疊代次數也會比其它演算法來的多次，而疊代的次數多寡所造成的影像品質差異將在表 4.7 中表示。

表 4.7 EZW 演算法重建疊代次數與 PSNR 關係表。

疊代次數	7	8	9	10
PSNR(dB)	37.4917	40.6211	42.6188	43.312
所需位元	127546	196440	269997	352764
相差位元	-	68894	73557	82767

由表 4.7 及上述其小波係數組合及分類後所得之效能可知，LENA 圖在相同位元率條件下，本論文所得到之重建係數疊代次數為 10 次而 2D SPIHT 演算法約為 7 次，所以本論文所提出的演算法在相同位元率下，其 PSNR 會比其他演算法高，再經由 4.2 節之實驗證實，確實如此。

其次可再由表 4.5 及表 4.6 觀察可得知，本論文所提出之 WT-BTC 演算法運用在體積型影像時，在相同位元率下，其 PSNR 表現更有明顯的差異，而造成這些差異之處，可表 4.8 內的三種不同類型之影像當輸入樣本所做的實驗可大略得知，當不同類型影像疊在一起，視為一個體積型影像壓縮時，其造成的效果如表 4.9 所示。

表 4.8 不同影像之 PSNR、位元率與位元圖組。

	單純背景			複雜背景			有主體		
代號	A	B	C	D	E	F	G	H	I
PSNR	44.02	43.39	42.24	40.68	39.94	38.59	42.37	41.11	42.43
bpp	0.14	0.19	0.18	0.67	0.76	0.93	0.27	0.40	0.24
位元組	467	806	786	1677	1882	2292	1159	1441	993

表 4.9 位元率與位元圖組之關係表。

	單純背景			複雜背景			有主體		
組合	ABC	ABF	ABH	EFD	EFB	EFH	GHI	GHB	GHE
bpp	0.17	0.42	0.38	0.79	0.54	0.63	0.30	0.29	0.53
位元組	1938	3344	3247	5273	4003	4574	3337	3159	4487
Total									
組合	ABCDEFGHI			bpp	0.42	位元組	9389		

由表 4.9 知，當體積型影像中，如在原複雜背景群中，出現較單純背景之影像時，其傳送整體之位元率會相對下降，從表 4.9 中的 EFD 及 EFB 組合看出，兩組相同是一次傳送三張之序列影像，但因影像本身因素，導致在整體傳送時所造成的效應。另外在表 4.5 及 4.6 中，也間接說明此一關係。因表 4.5 所使用之輸入影像為空照圖影像，空照圖影像是屬於變化性較大之影像，而表 4.6 所使用之影像為核磁共振之醫學影像，其影像變化差異較小，所以在表 4.5 及 4.6 內 1bpp 下，其大致相差為 6.2dB 及 3.5dB。最後在對本論文所提出之 WT-BTC 演算法在壓縮上所需的時間與其他演算法做分析比較，其結果分為兩類，第一類為單張影像在相同位元率下 WT-BTC 演算法與 2D SPIHT 演算法所需時間，結果如表 4.10 所示，第二類為體積型影像相同位元率下 WT-BTC 與 3D SPIHT 所需時間，結果表 4.11 所示。

表 4.10 相同位元率下 WT-BTC 與 2D SPIHT 所需時間表。 (單位:秒)

位元率(bpp)	0.125	0.25	0.5	1
WT-BTC	39.13 s	40.04 s	42.53 s	51.39 s
2D SPIHT	1.90 s	3.03 s	5.14 s	11.59 s

表 4.11 相同位元率下 WT-BTC 與 3D SPIHT 所需時間表 (單位:秒)

位元率(bpp)	0.125	0.25	0.5	1
WT-BTC	87.65 s	95.774 s	109.843 s	112.45 s
3D SPIHT	4 s	4.45 s	4.995 s	6.93 s

由表 4.10 及表 4.11 可觀察知，本論文所提之演算法雖在 *PSNR* 及壓縮率均比其它演算法來的佳，但在時間上確比其它演算來多出 5-20 倍之多，尤其在低位元率下，所需之時間倍數也較多且 *PSNR* 差異也較小，因此本論文所提之演算法適用於傳輸位元率較高之影像壓縮。

第五章

結論與未來展望

本論文提出了一個結合小波轉換及方塊截短編碼的演算法，並利用資料探勘的方式，在許多影像中取出具代表其影像之位元圖，來達到集體壓縮的效果。而在本論文的第四章中，許多不同的輸入影像測試下，也突顯出來使用資料探勘及基因演算法內適應值之輪盤式選擇觀念所到的位元圖群，與之比其它使用方塊截短編碼所篩選之位元圖群，來更具代表性且有所依據。而另外在之前的方塊截短編碼演算法裡，本論文使用影像辨識的分群觀念，將其作改變以讓其更符合所需，其所造成之效能在本論文第四章最後第一節有詳細說明。

在未來的發展上可以分為下列幾項：1.本論文所使用的方法，對於壓縮影像的時間上仍須改進。2.利用本論文內篩選位元圖之方法在其它影像壓縮法，如 EBCOT 等中，來實現讓其壓縮率更高。3.充份利用基因演算法尋找最佳解特性使找到之位元圖群數可得最佳化，使系統之效能更佳。4.結合**分散式系統**對各類相關係數表格做平行處理以加速編碼。5.結合於**3D 影像重建技術與資料探勘中方向與時間之序列處理技術**，完成新 3D 影像重建。

參考文獻

- [1] D. T. Larose, *Discovering Knowledge in Data, An Introduction to Data Mining*. New York: John Wiley & Sons Inc, 2005.
- [2] C. Karr, “Design of an Adaptive Fuzzy Logic Controller Using a Genetic Algorithm,” *Proc. 4th Intl. Conf. Genetic Algorithms*, 1991, pp. 450–457.
- [3] K. S. Tang, K. F. Man, S. Kwong, and Z. F. Liu, “Design and Optimization of IIR Filter Structure Using Hierarchical Genetic Algorithms,” *IEEE Trans. on Industrial Electronics*, vol. 45, no. 3, Jun. 1999, pp. 481-487.
- [4] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Upper Saddle River, NJ: Prentice Hall, 2002.
- [5] S. G. Mallat, “A Theory for Multiresolution Signal Decomposition: the Wavelet Representation,” *IEEE Trans. on Pattern Analysis and Machine Intel.*, vol. 11, no. 7, Jul. 1989, pp. 674-693.
- [6] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image Coding Using Wavelet Transform,” *IEEE Trans. on Image Processing*, vol. 1, no. 2, Apr. 1992, pp. 205-220.
- [7] T. Kurita and N. Otsu, “A Method of Block Truncation Coding for Color Image Compression,” *IEEE Trans. on Cmmunications*, vol. 41, no. 9, , Sep. 1993, pp. 1270-1274.

- [8] Y. Wu and D. C. Coll, "Single Bit-Map Block Truncation Coding of Color Image," *IEEE J. Selected Areas in Communications*, vol. 10, no. 5, Jun. 1992, pp. 952-959.
- [9] J. M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficient," *IEEE Trans. on Signal Processing*, vol. 41, no. 12, Dec. 1993, pp. 3445-3462.
- [10] P. G. Howard and J. S. Vitter, "Arithmetic Coding for Data Compression," *IEEE Proceedings*, vol. 82, no. 6, Jun. 1994, pp. 857-865.
- [11] H. Meng and Z. Wang, "Fast Spatial Combinative Lifting Algorithm of Wavelet Transforms Using the 9/7 Filter for Image Block Compression," *Electronics Letters*, vol. 36, no. 21, Oct. 2000, pp. 1766-1767.
- [12] R. Calderbank, I. Daubechies, W. Sweldens, and B. L. Yeo, "Wavelet Transforms That Map Integers to Integers," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, 1998, pp. 322-369.
- [13] A. Said and W. A. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," *IEEE Trans. on Circuits Systems for Video Technology*, vol. 6, no. 3, Jun. 1996, pp. 243-250.
- [14] D. Taubman, "High Performance Scalable Image Compression with EBCOT," *IEEE Trans. on Image Processing*, vol. 9, no. 7, Jul. 2000, pp. 1158-1170.
- [15] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Kluwer Academic Publishers, Nov. 2001.
- [16] C. J. Lian, K. F. Chen, H. H. Chen, and L. G. Chen, "Analysis and Architecture Design of Block Coding Engine for EBCOT in JPEG 2000," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 3, Mar. 2003, pp. 219-230.

- [17] W. J. Chen and S. C. Tai, "A Genetic Algorithm Approach to Multilevel Absolute Moment Block Truncation Coding," *IEICE Trans. on Fundamentals of Electronics, Communications, and Computer Science*, vol. E82-A, no. 8, Aug. 1999.
- [18] D. A. Huffman, "A Method for the Construction of Minimum Redundancy Codes," *Proc. IRE*, 40, 1952, pp. 1098-1101.
- [19] E. Khan and M. Ghanbari, "Very Low Bit Rate Video Coding Using Virtual SPIHT," *Electronics Letters*, vol. 37, no. 1, Jan. 2001, pp. 40-42.
- [20] Y. Cho and W. A. Pearlman, "Hierarchical Dynamic Range Coding of Wavelet Subbands for Fast and Efficient Image Decompression," *IEEE Trans. on Image Processing*, vol. 16, no. 8, Aug. 2007, pp. 2005-2015.
- [21] B.-J. Kim, Z. Xiong, and W. A. Pearlman, "Low Bit-Rate Scalable Video Coding with 3-D Set Partitioning in Hierarchical Trees (3-D SPIHT)," *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 10, no. 8, Dec. 2000, pp. 1374-1387.