

南 華 大 學

資訊管理學系

碩士論文

快速 JPEG 2000 影像尺寸縮小轉換器

Fast downsizing transcoder for JPEG 2000 images



研 究 生：胡光南

指 導 教 授：廖怡欽

中 華 民 國 97 年 6 月 30 日

南 華 大 學

資訊管理學系

碩 士 學 位 論 文

快速 JPEG 2000 影像尺寸縮小轉換器

研究生：胡光南

經考試合格特此證明

口試委員：廖怡欽

羅日順

邱宏彬

指導教授：廖怡欽

系主任(所長)：

口試日期：中華民國 96 年 06 月 27 日

誌 謝

首先誠摯的感謝指導教授廖怡欽博士，在過去的兩年裡，給予我課業上與研究上的細心指導與關懷，老師悉心的教導使我在這兩年中獲益良多。同時也感謝邱宏彬博士，在這兩年裡，不斷給予鼓勵與課外教導，在此一併感謝、感恩兩位老師的指導與鼓勵。

在研究所兩年的日子裡，發生過風風雨雨的大小事，但感謝眾位學長與同學-育鴻學長、明勳、宗亨、宣均等同學的砥礪，讓我能安然渡過低潮期。

最後感謝我的雙親、小阿姨家人、與女友倩如，在我感到灰心時，適時給予我鼓勵，讓我更有信心的完成研究所的學業。

在此將本論文獻給所有關心我的人，希望將這份喜悅與榮耀分享給大家。

快速 JPEG 2000 影像尺寸縮小轉換器

學生：胡光南

指導教授：廖怡欽

南華大學資訊管理系碩士班

摘 要

本論文提出一個快速影像尺寸縮小方法，所提方法在頻率域對 JPEG 2000 影像進行影像尺寸縮小工作。跟傳統在空間域縮小 JPEG2000 影像尺寸的作法相比，本論文所提方法，可有效降低記憶體空間需求以及執行時間。實驗結果顯示，將影像尺寸為 512x512 的輸入影像縮小為原來的 $1/2 \times 1/2$ 至 $1/32 \times 1/32$ 大小時，使用我們所提的快速影像尺寸縮小轉換方法，平均比在空間域進行影像尺寸轉換可節省 53%~78% 的執行時間。

關鍵字：JPEG 2000、轉換器、影像尺寸縮小、頻率域。

Fast downsizing transcoder for JPEG 2000 images

Student : Guang-Nan Hu

Advisor : Dr. Yi-Ching Liaw

Department of Information Management
The M.I.M. Program
Nan-Hua University

Abstract

This paper presents a fast downsizing method for JPEG 2000 images. The proposed method is used to downsize the JPEG 2000 images under frequency domain. Compared with the traditional method, which downsizes the JPEG 2000 images under spatial domain, our proposed method can effectively reduce both required memory space and execution time. Experimental results reveal that the proposed frequency domain downsizing method can reduce the average execution time of the spatial domain downsizing method by 53% to 78% for images of the size 512×512 are downscaled to $1/2 \times 1/2$ to $1/32 \times 1/32$ of the original scale.

Keywords : JPEG 2000 、 transcoder 、 image downsizing 、 frequency domain

目 錄

書名頁	I
著作財產權同意書	II
論文指導教授推薦書	III
論文口試合格證明	IV
誌謝	V
中文摘要	VI
英文摘要	VII
目錄	VIII
圖目錄	IX
表目錄	XI
第一章 緒論	1
第一節 研究背景	1
第二節 研究動機	4
第三節 研究目的	5
第四節 論文架構	5
第二章 研究背景	7
第一節 JPEG 2000 簡介	7
第二節 標準 JPEG 2000 壓縮與解壓縮流程	12
第三節 JPEG 2000 檔案格式	23
第四節 空間域轉換方法	27
第三章 快速 JPEG 2000 影像尺寸縮小轉換方法	34
第一節 簡化 JPEG 2000 壓縮與解壓縮流程	35
第二節 頻率域影像尺寸縮小轉換方法	42
第四章 實驗結果	47
第一節 實驗環境	47
第二節 空間域執行時間	48
第三節 小波層數足夠所需執行時間	51
第四節 小波層數不足所需執行時間	53
第五節 總論	55
第五章 結論與未來展望	57
參考文獻	58

圖 目 錄

圖 1.1:數位影像傳輸架構.....	1
圖 1.2:不同裝置的影像瀏覽畫面.....	2
圖 2.1:影像壓縮至 0.25bpp 後的 Lena 圖.....	11
圖 2.2:標準 JPEG 2000 流程圖.....	13
圖 2.3:Lena 的 R 、 G 、 B 與 Y 、 C_b 、 C_r 色彩平面圖.....	15
圖 2.4:二維順向小波轉換流程圖.....	16
圖 2.5:經三次小波轉換後的 JPEG 2000 影像頻率域資訊.....	17
圖 2.6:EBCOT 編碼流程圖.....	19
圖 2.7:EBCOT 解碼流程圖.....	20
圖 2.8:二維反向小波轉換流程圖.....	21
圖 2.9:編碼串流結構圖.....	24
圖 2.10:空間域影像大小轉換流程圖.....	27
圖 2.11:最近鄰居內插法示意圖.....	28
圖 2.12:使用最近鄰居內插法處理後的影像.....	28
圖 2.13:雙線性內插法示意圖.....	29
圖 2.14:使用雙線性內插法處理後影像.....	30

圖 2.15: 雙立方內插法示意圖.....	31
圖 2.16: 使用雙立方內插法處理後影像.....	32
圖 3.1: 快速 JPEG 2000 影像尺寸縮小轉換方法.....	34
圖 3.2: 經二次小波轉換後產生 7 個子頻帶示意圖.....	37
圖 3.3: 經三次小波轉換後的 JPEG 2000 影像頻率域資訊.....	38
圖 3.4: 三層子頻帶使用不同量化步階值的例子.....	38
圖 3.5: 二層子頻帶使用不同量化步階值.....	39
圖 3.6: 對不同的子頻帶做不同的反量化示意圖.....	40
圖 3.7: 針對層數不足的影像作小波轉換方法示意圖.....	41
圖 3.8: 針對保留子頻率資訊作反量化示意圖.....	43
圖 3.9: 快速小波轉換方法示意圖.....	45
圖 4.1: Lena 測試不同縮小影像尺寸範例.....	56

表 目 錄

表 2.1:JPEG 與 JPEG 2000 壓縮技術比較表.....	10
表 4.1:使用平均法縮小影像尺寸平均所需執行時間.....	49
表 4.2:使用平均法縮小影像尺寸產生的檔案大小.....	49
表 4.3:使用最近鄰居內插法縮小影像尺寸平均所需執行時間.....	49
表 4.4:使用最近鄰居內插法縮小影像尺寸產生檔案大小.....	49
表 4.5:使用平均法產生一層小波轉換 JPEG 2000 影像平均所需執行時間.....	50
表 4.6:使用平均法產生一層小波轉換 JPEG 2000 影像檔案大小.....	50
表 4.7:使用最近鄰居內插法產生一層小波轉換 JPEG 2000 影像平均所需執行時間.....	50
表 4.8:使用最近鄰居內插法產生一層小波轉換 JPEG 2000 影像檔案大小.....	51
表 4.9:JPEG 2000 影像小波層數足夠-對整張影像作反量化平均所需執行時間.....	52
表 4.10:JPEG 2000 影像小波層數足夠-對保留子頻帶做反量化平均所需執行時間.....	52
表 4.11:表 4.9 與表 4.11 產生的檔案大小.....	52

表 4.12:表 4.9 與表 4.10 之轉換執行時間比.....	53
表 4.13 : JPEG 2000 影像小波層數不足-使用一般小波轉換平均所需執行 時間.....	54
表 4.14 : JPEG 2000 影像小波層數不足-使用快速小波轉換平均所需執行 時間.....	54
表 4.15:表 4.12 與 4.13 產生的檔案大小.....	54
表 4.16:表 4.12 與表 4.13 之轉換執行時間比.....	55

第一章 緒論

第一節 研究背景

隨著科技的進步以及數位影像的普及，使用者可簡單的使用手機、個人數位助理(PDA)、筆記型電腦(NB)、或個人電腦(PC)等裝置透過網際網路(Network)連線到伺服器(Server)來取得數位影像資訊。用戶端與伺服器的數位影像傳輸架構如圖 1.1 所示：

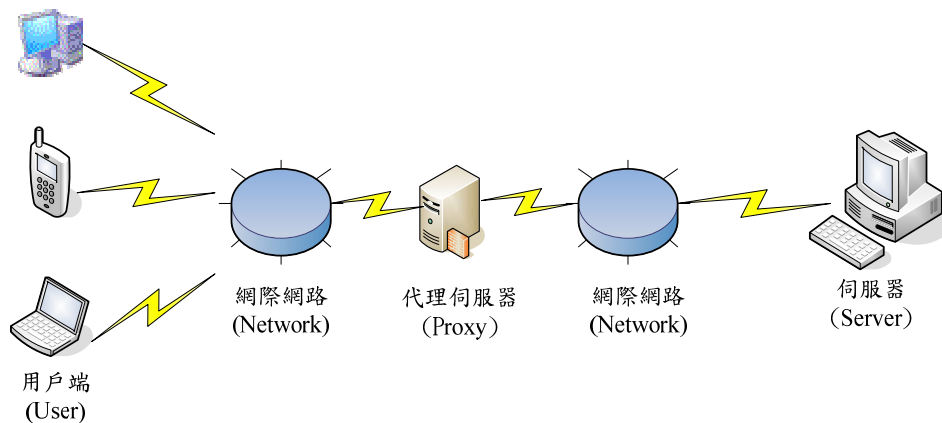


圖 1.1：數位影像傳輸架構

在圖 1.1 所示的數位影像傳輸架構中，用戶端(手機、個人數位助理、或個人電腦等)可透過網際網路連線至伺服器，要求伺服器提供所需的數位影像資訊，伺服器則將用戶所要求的數位影像透過網際網路經由代理主機(Proxy)傳送給用戶端。

由於不同用戶裝置的計算能力、可用網路頻寬、以及顯示能力都不相同，因此提供相同尺寸的影像給不同裝置，並不合適。例如：手機或個人數位助理等裝置通常只提供較小的可用網路頻寬、低階的處理能

力、以及低解析度的顯示介面。若將高解析度的影像直接傳送給低解析度顯示介面裝置時，會導致影像傳送時間久、處理速度慢、或者降低網路整體傳輸速度等問題。因此，一旦手機或 PDA 等這類低階裝置接收伺服器所傳的高解析度影像時，處理起來就顯得特別的緩慢，而且通常無法看到影像全貌。其可能結果如圖 1.2 所示：

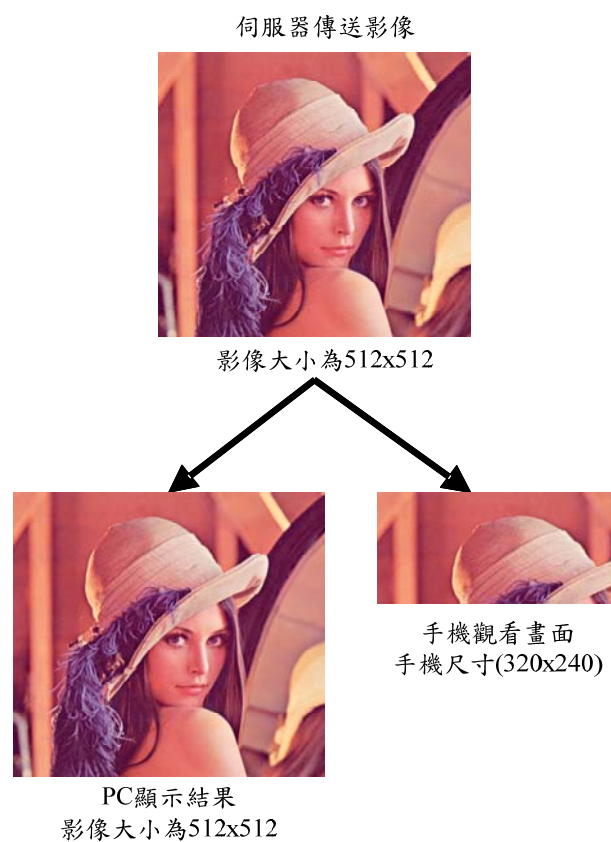


圖 1.2：不同裝置的影像瀏覽畫面

為了解決將高解析度影像傳送給低階裝置的問題，伺服器必須傳送適合用戶裝置大小的影像內容給用戶。要傳送適合用戶裝置的影像大小給用戶，有三種主要的作法：一、儲存多種尺寸的影像；二、使用漸進

式影像傳輸技術；三、使用影像尺寸轉換技術。三種解決方式分別說明如下：

■ 儲存多種尺寸的影像

針對同一張影像儲存多種尺寸，這種作法主要缺點在於佔用儲存空間，以及需維護不同尺寸影像內容的一致性。

■ 使用漸進式影像傳輸技術

作法有以下兩種：

作法一：用戶端向伺服器要求影像，伺服器使用漸進式方式傳送影像內容給用戶，當用戶收到所需要的影像內容後，立即要求伺服器停止傳送。這種作法在用戶端發出停止傳送要求給伺服器至伺服器停止傳送影像內容這段期間，伺服器所傳送的影像內容，將造成網路頻寬的浪費。

作法二：用戶端向伺服器要求所需影像及尺寸後，伺服器針對用戶的要求，分析影像內容並取出適合用戶的影像內容，再將影像內容傳送給用戶。這種作法雖然不會浪費網路頻寬，但會增加伺服器的處理負擔。

■ 使用影像尺寸轉換技術

此技術的作法是當用戶端向伺服器要求所需影像與尺寸後，伺服器

(或代理伺服器)將影像轉換成適合用戶裝置的大小後，將轉換後的影像傳送給用戶並儲存在伺服器的快取區域(Cache)。當下次有用戶要求同樣尺寸的相同影像時，則可直接從快取區域取出轉換後的影像傳送給用戶，不需再進行影像轉換動作。這種作法不會浪費網路頻寬，也不會增加伺服器太多儲存與處理負擔。

由於影像尺寸轉換技術所具有的優點，以及影像大小轉換需求的增加，影像尺寸轉換技術已成為熱門的研究領域之一。

第二節 研究動機

目前已有許多可用的影像與視訊尺寸轉換方法[1-7]。Salazar 等人[1]提出直接在 DCT 頻域下進行影像大小轉換的方法。Patil 等人[2]針對 MPEG-2 格式的視訊提出任意視訊尺寸轉換方法。Shu 等人[3]則利用修改位元率與影像大小等技術來調整傳輸資料量。Park 等人[4]在 DCT 頻域下依所指定位元率來調整影像尺寸與壓縮率。Liagn 等人[5]提出有效的移動向量(motion vectors)估測方法，可加快縮小視訊尺寸時重新計算移動向量的時間，且可維持良好的影像品質。Tan 等人[6]針對 H.264/AVC 格式的視訊使用區域權重向量中位數濾波器(area-weighted vector median filter)來轉換視訊尺寸。Lee 等人[7]為提高視訊品質，提出在空間域下對 MPEG-2 格式的視訊進行任意縮小視訊尺寸轉換方法。

JPEG 2000 [8-10]是一個新的靜態影像壓縮標準，具有失真(lossy)與無失真(lossless)壓縮能力、影像安全性保護、漸進式傳輸以及抗雜訊等優點。目前 JPEG 2000 的應用也越來越廣泛[11]，例如數位相機、掃描器、視訊監視、無線影像傳輸、醫學影像與衛星影像等，而且現在支援 JPEG 2000 的軟體也愈來愈多種，包括有 PhotoShop[12]、PhotoImpact[13]、Image Eye[14]、FastStone Image Viewer[15]以及 Irfan View[16]等。隨著 JPEG 2000 的應用與支援軟體愈來愈多樣化，透過網路傳送 JPEG 2000 影像的需求也將隨之增加。

第三節 研究目的

為了降低 JPEG 2000 壓縮影像大小轉換所需的時間，本論文提出一個新的 JPEG 2000 影像大小轉換方法，可在頻率域(frequency domain)下對 JPEG 2000 影像進行影像尺寸縮小動作，以減少影像大小轉換所需的時間。所提方法針對使用失真壓縮方式所產生的 JPEG 2000 影像，可將影像尺寸縮小成原來的 $(1/2^n \times 1/2^n)$ 大小，其中 $n \geq 1$ 。

第四節 論文架構

本論文其餘內容安排如下，第二章介紹標準 JPEG 2000 壓縮與解壓縮流程以及空間域影像大小轉換方法，第三章說明本論文所提的快速

JPEG 2000 影像尺寸縮小轉換方法，第四章列出頻率域與空間域影像尺寸縮小方法的實驗結果，第五章為結論與未來方向。

第二章 研究背景

本章將詳細介紹 JPEG 2000 壓縮技術以及空間域影像大小轉換方法。

第一節 JPEG 2000 簡介

國際標準組織(International Organization for Standardization, ISO)與國際電工委員會(International Electrotechnical Commission, IEC)在西元 1988 年提出 JPEG(Joint Photographic Experts Group)壓縮標準。起初主要功能用於影像壓縮，之後更廣泛的應用到影像資料交換、掃描器、印表機、數位相機、以及網際網路上[11]。

隨著近年來多媒體的應用急速成長以及網路所提供的有限頻寬，當使用者開始對影像品質與影像功能有更多要求時，JPEG 便已開始顯現出不足之處，例如：

- (1) 影像經由網路傳遞遇到雜訊時，會破壞影像品質。
- (2) 影像壓縮約至 0.25 bpp(bit per pixel)以下時，影像重建品質不佳。
- (3) 無法滿足醫學影像高解析度品質需求。
- (4) 無法支援大於 $2^{16} \times 2^{16}$ 大小的影像。
- (5) 針對圖文混和影像壓縮效能不佳。

為了解決 JPEG 不足之處，ISO 與 IEC 自 1997 年開始籌劃新一代靜態影像的壓縮標準，並於西元 2001 年正式提出 JPEG 2000 影像壓縮標準。到目前為止完整的 JPEG 2000 可分成以下十二個部分[17]：

Part-1：JPEG 2000 Image Coding System-提供基本的編碼器及壓縮位元流(bit-stream)格式。

Part-2：Extensions(*adds more features and sophistication to the core*)-提供除了基本編碼以外的特殊功能。此部份是可選擇的項目，並非所有 JPEG 2000 解碼器都能支援。

Part-3：Motion JPEG 2000-針對連續動態影像而訂的規格，允許包含多張連續影像畫面。

Part-4：Conformance-包含 JPEG 2000 編碼的複雜度與正確性測試等相關規範。

Part-5：Reference software(*Java and C implementations are available*)-提供兩套分別以 Java 與 C 語言實作的 JPEG 2000 參考軟體，並提供開程式碼供參考，其中以 Java 語言開發完成的 JPEG 2000 參考軟體為 JJ 2000，而使用 C 語言開發的參考軟體為 Jasper。

Part-6 : Compound Image file format(*pre-press and fax like applications*)-根據複合文件所做的延伸，主要針對圖形與文字混和型影像的編碼規定，將文字與圖形以最佳狀態進行編碼。

Part-7 : Has been abandoned-原本是關於硬體實現相關規範，但此規範已被終止。

Part-8 : JPSEC(*secure aspects*)-增加 JPEG 2000 本身對加密、數位浮水印(watermark)與權限上的控制。

Part-9 : JPIP(*Interactive protocols and API*)-提昇 JPEG 2000 使用在不同網路與應用上的傳輸效能。

Part-10 : JP3D(*volumetric imaging*)-提供 JPEG 2000 針對 3D 影像上的支援。

Part-11 : JPWL (*wireless applications*)-規範 JPEG 2000 運用在無線應用上可能的限制與條件。

Part-12 : ISO Based Media File Format-增加 JPEG 2000 與其他 ISO 標準(例如：MPEG)的相容性，同時考慮支援未來 MPEG-21 的格式。

JPEG 2000 壓縮技術與傳統 JPEG 壓縮技術之間最大的不同之處，如

表 2.1 所示。

表 2.1：JPEG 與 JPEG 2000 壓縮技術比較表

	JPEG	JPEG 2000
轉換方法	離散餘弦轉換	離散小波轉換
掃描方法	Z 字型掃描	位元平面掃描
編碼方法	霍夫曼編碼	嵌入式區塊編碼

在表 2.1 中，JPEG 2000 採用離散小波轉換(Discrete Wavelet Transform, DWT)為主要的多解析編碼方法，放棄 JPEG 所採用的離散餘弦轉換(Discrete Cosine Transform, DCT)為主的區塊編碼方式。JPEG 2000 會使用離散小波轉換的主要目的在於要將影像的頻率成分抽取出來，分別加以控制與編碼。利用小波轉換的功能，可更容易實現漸進式傳輸的特性，首先傳輸影像的輪廓，然後再慢慢傳輸影像細部資訊，不斷提高影像品質，讓影像從原本朦朧到清晰顯示，不必像目前 JPEG 一樣，由上到下慢慢將影像顯示出來。並且 JPEG 2000 與傳統 JPEG 兩種編碼方法相較之下，JPEG 2000 採用嵌入式區塊編碼(Embedded Block Coding with Optimized Truncation, EBCOT)[18-19]，比起 JPEG 所使用的霍夫曼編碼(Huffman coding)更有效率，且經由 JPEG 2000 壓縮後的影像品質比使用 JPEG 壓縮後的影像品質更加優越。

圖 2.1 所示為 JPEG 與 JPEG 2000 將影像壓縮約至 0.25bpp 時，JPEG 壓縮後的影像明顯出現方塊效應(blocking effect)與蚊子效應(mosquito noise)，但使用 JPEG 2000 壓縮的影像，仍然可以得到不錯的影像品質。



(a)JPEG 壓縮影像



(b)JPEG 2000 壓縮影像

圖 2.1：影像壓縮至 0.25bpp 後的 Lena 圖

JPEG 2000 壓縮技術主要特色整理如下：

- (1) 允許多種影像格式：包括灰階(gray)、全彩(true color)、二階(binary)、multicomponent、與 hypercomponent 影像等。
- (2) 適用多種影像應用：包括自然影像、醫學影像、軍事用途影像、與遙測影像等。
- (3) 優秀低位元率表現：當位元率降低時，JPEG 壓縮後的影像會出現明顯的失真情況，但 JPEG 2000 壓縮的影像能維持不錯的影像品質。
- (4) 漸進式傳輸(Progress Transmission)特性：先傳輸影像的輪廓，然

後傳輸影像細部資訊，不斷提高影像品質，使得影像從朦朧到清晰顯示出來，如此一來能有效節省與充份利用有限的頻寬。

- (5) 增強對位元率錯誤容忍度：有較好的容錯性，可減少解碼失敗的機率。避免造成影像產生連續整條帶狀的畫面出現。
- (6) 支援 ROI(Region of Interesting)編碼：依使用者的喜好，選擇個人有興趣的區域，針對該區域作更細緻的編碼，而非重要的區域可選擇是否該大量壓縮。例如：利用視訊傳送目前演說者正在發表演說的畫面，並以演說者(例如：臉部)為傳送重點，其餘背景影像就可大量壓縮或者選擇不傳送。
- (7) 影像安全性保護：運用浮水印、標記(labeling)、與加密(encryption)讓影像資料保持完整性不易被竄改。
- (8) 更有效率的算術編碼：JPEG 2000 使用 EBCOT 壓縮方法，比現有 JPEG 所使用的 Huffman coding 壓縮更有效率。
- (9) 支援有失真(lossy)與無失真(lossless)壓縮。
- (10) 支援大型影像，最大可至 $2^{32} \times 2^{32}$ 影像大小。

第二節 標準 JPEG 2000 壓縮與解壓縮流程

JPEG 2000 壓縮技術提供兩種壓縮方式，一種為無失真壓縮，另一種

為失真壓縮。本論文所提的影像大小轉換方法，針對使用失真壓縮方式所產生的 JPEG 2000 影像。因此以下僅針對失真壓縮的流程進行說明。

圖 2.2 所示為 JPEG 2000 的壓縮與解壓縮流程：

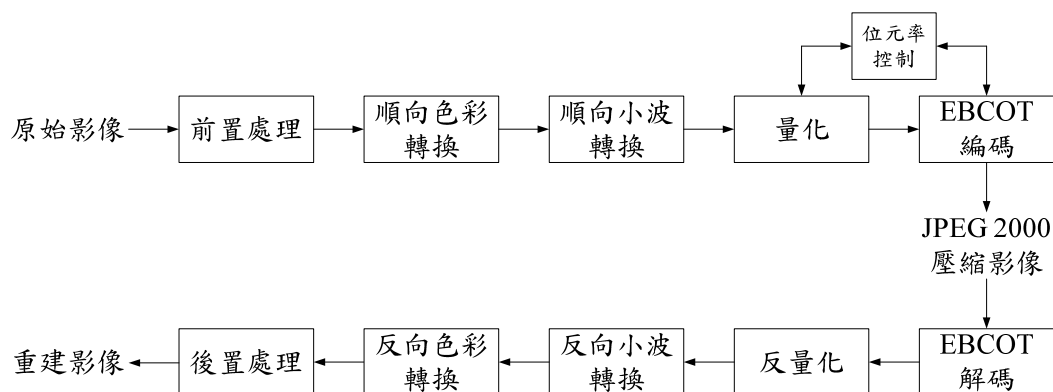


圖 2.2：標準 JPEG 2000 流程圖

壹、 JPEG 2000 壓縮流程

JPEG 2000 壓縮流程如圖 2.2 上半部所示，處理流程簡單說明如下：

原始影像經由前置處理針對影像裡每一個像素作直流準位的位移 (DC-shift)，得到 DC 位移調整後的影像，再進入到色彩轉換(Color transform)將影像的色彩空間由 R 、 G 、 B 轉換成 Y 、 C_B 、 C_R 色彩空間，然後再分別對 Y 、 C_B 、 C_R 三個色彩平面使用離散小波轉換得到頻率域資訊，再利用量化(Quantization)降低頻率域的資訊量，最後再利用 EBCOT 編碼，產生壓縮後的 JPEG 2000 影像。

由以上說明可知，原始影像經由 JPEG 2000 編碼時會經過幾道處理

過程才能夠產生 JPEG 2000 壓縮影像。以下將詳細介紹編碼過程中的每一個處理流程。

■ 前置處理

前置處理(Pre-process)階段最重要的工作是將影像的像素(pixel)值作直流準位的位移(DC-shift)，以利後續的編碼使用。假設某個影像其像素值範圍在 $[0, 2^P-1]$ ，其中 P 表示每個像素點用 P 個位元來表示(也就是位元深度, bit-depth)，則必須將每個像素值範圍減去 2^{P-1} ，使得新的係數值範圍分佈在 $[-2^{P-1}, 2^{P-1}-1]$ 。

■ 順向色彩轉換

在 JPEG 2000 中，提供兩種不同的色彩轉換，分別為不可逆的色彩轉換(Irreversible Color Transform, ICT)以及可逆的色彩轉換(Reversible Color Transform, RCT)。其中不可逆色彩轉換用於失真壓縮，且需配合使用有失真的 9/7 小波轉換，以及使用純量量化器(scalar dead-zone quantization, DZQ)。可逆色彩轉換則供無失真壓縮使用，需配合無失真 5/3 小波轉換，且量化時的索引(index)需為原來的係數值，也就是量化步階值固定為 1。

使用順向不可逆色彩轉換，將影像的 R 、 G 、 B 轉換成 Y 、 C_b 、 C_r 所

採用的公式如公式(1)所示。

$$\begin{aligned} Y &= 0.299 R + 0.587 G + 0.114 B \\ C_b &= -0.16875 R - 0.33126 G + 0.5 B \\ C_r &= 0.5 R - 0.41869 G - 0.08131 B \end{aligned} \quad (1)$$

圖 2.3 所示為 Lena 的 R 、 G 、 B 三個色彩平面與轉換成 Y 、 C_b 、 C_r 三個色彩平面的影像。



圖 2.3：Lena 的 R 、 G 、 B 與 Y 、 C_b 、 C_r 色彩平面圖

(a) R , (b) G , (c) B , (d) Y , (e) C_b , (f) C_r

■ 順向小波轉換

JPEG 2000 裡的小波(wavelet)轉換，分別有實數模式(real mode)與整數模式(integer mode)兩種。實數模式主要採用 9/7 的濾波器(filter)、但儲存資料的精確度有限，所以又稱為不可逆轉換，也就是反向轉換後的值是很接近原先轉換的值，但不會完全相同。另一種整數模式主要採用 5/3 的濾波器、儲存資料的精確度範圍較大，所以是一種可逆的轉換。

當影像進入順向小波轉換時，首先小波轉換的處理過程先針對影像以每一橫列進行一維順向小波處理，當每一列都處理完，將所產生的資訊再以每一直行方法進行一維順向小波處理，最後產生小波頻率資訊。

其處理過程如圖 2.4 所示：

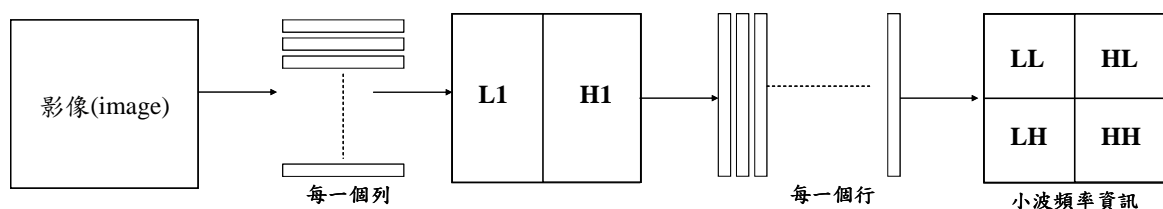


圖 2.4：二維順向小波轉換流程圖

產生後的小波頻率資訊共有 4 塊不同頻率的資訊(圖 2.4 所示)，分別為 LL、HL、LH、與 HH 這 4 種子頻帶(subbands)。LL(低低頻)頻帶屬於低頻影像，且肉眼可接受敏感度高。HL(高低頻)、LH(低高頻)與 HH(高高頻)這些頻帶為屬於高頻影像，肉眼可接受敏感度低。低頻子頻帶(LL)為低解析度空間域影像內容，可再使用小波轉換產生另外 4 個較小的子

頻帶。

圖 2.5 所示為一張影像經過三次小波轉換後的結果。其中 LL0、HL0、LH0 與 HH0 稱為第一層(最內層)的頻率資訊。HL1、LH1 與 HH1 為第二層的頻率資訊，HL2、LH2 與 HH2 為第三層(最外層)的頻率資訊。

LL0	HL0	HL1	HL2
LH0	HH0		
LH1		HH1	
LH2		HH2	

圖 2.5：經三次小波轉換後的 JPEG 2000 影像頻率域資訊

在順向不可逆小波轉換時，所採用的公式如下：

$$\begin{aligned}
 d^{(1)}[n] &= a_0[2n+1] + \alpha(a_0[2n] + a_0[2n+2]), \\
 s^{(1)}[n] &= a_0[2n] + \beta(d^{(1)}[n-1] + d^{(1)}[n]), \\
 d^{(2)}[n] &= d^{(1)}[n] + \gamma(s^{(1)}[n] + s^{(1)}[n+1]), \\
 s^{(2)}[n] &= s^{(1)}[n] + \delta(d^{(2)}[n-1] + d^{(2)}[n]), \\
 d_1[n+1] &= \kappa \times d^{(2)}[n], \\
 a_1[n] &= s^{(2)}[n] / \kappa
 \end{aligned} \tag{2}$$

其中：

$a_0[n]$ ：為原始輸入訊號，其中 $0 < n < N$,

N ：輸入訊號的長度，

$$\begin{aligned}
\alpha &= -1.586134312059924, \\
\beta &= -0.052980118572961, \\
\gamma &= 0.882911075530934, \\
\delta &= 0.443506852043971, \\
\kappa &= 1.230174104914001
\end{aligned}$$

■ 量化

在 JPEG 2000 中，使用到兩種量化器，分別為 Part-1 所使用的純量量化器與 Part-2 所使用的格子碼量化器 (trellis coded quantization)[20-21]。在 Part-1 所使用到的純量量化器如公式(3)所示。其中公式(3)裡 $y_b(n)$ 表示在子頻帶 b 中第 n 個係數的原取樣值，而 $q_b(n)$ 為子頻帶 b 中第 n 個係數量化後的頻率值。

$$q_b(n) = \text{sign}[y_b(n)] \cdot \text{floor} \left[\frac{|y_b(n)|}{\Delta_b} \right] \quad (3)$$

由於每個子頻帶有不同的特性，因此每個子頻帶的量化程度可以不同。子頻帶的量化程度是由該子頻帶的 Δ_b 所決定。 Δ_b 稱為量化步階值，量化步階值的計算方式如公式(4)所示。

$$\Delta_b = 2^{R_b - \varepsilon_b} \left(1 + \frac{\mu_b}{2^{11}} \right) \quad (4)$$

其中每個子頻帶的步階值 (stepsizes) 由動態範圍 R_b 與指數 (exponent) ε_b 和尾數 (mantissa) μ_b [22] 等控制參數來決定。

■ 位元率控制

位元率控制(rate control)過程會依據影像內容對量化步階值的控制參數(ϵ_b, μ_b)進行調整，使得影像壓縮後的位元率符合要求的位元率。

■ EBCOT 編碼

EBCOT 使用兩個階段的編碼器(2-tier Encoder)。第一層(tier-1)編碼器(encoder)主要結合位元編碼(bit-coding)與算術編碼(Arithmetic coding, AC)兩個部份。第二層(tier-2)編碼器將第一層編碼器所編碼後的資料製作成封包(packet)後傳送。

首先將影像所量化後的子頻帶細分成相同大小的編碼區塊(code-block，一般是 32×32 或 64×64 ，最小可到 16×16)，將各個編碼區塊再獨立進行編碼後，產生內文(context)，並利用全文(MQ)編碼器進行算術編碼產生位元串流(bit-stream)；第二層編碼器利用 tier-1 所編碼出的位元串流，進行編碼串流(code-stream)的封裝作業，以產生檔案的封包。圖

2.6 所示為 EBCOT 的編碼流程。

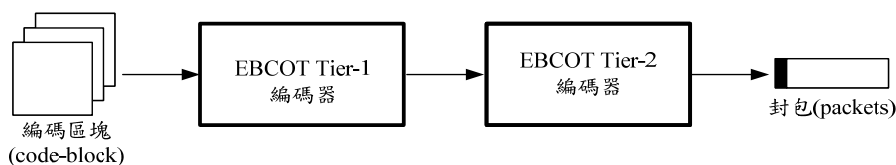


圖 2.6：EBCOT 編碼流程圖

貳、 JPEG 2000 解壓縮流程

JPEG 2000 影像解壓縮流程如圖 2.2 下半部所示，JPEG 2000 壓縮影像首先利用 EBCOT 解碼解出 Y 、 C_b 、 C_r 三個色彩平面被量化的頻域資訊；針對每個色彩平面分別進行反量化(De-quantization)步驟還原影像的頻率資訊；接著進行反向小波轉換(Inverse Wavelet Transform)，產生空間域資訊；有了 Y 、 C_b 、 C_r 三個色彩平面的空間域資訊後，再利用反向色彩轉換將 Y 、 C_b 、 C_r 色彩空間轉換成 R 、 G 、 B 色彩空間；最後再利用後置處理(Post-process)調整每個像素的直流準位，產生重建影像。

JPEG 2000 解壓縮其實就是將 JPEG 2000 壓縮過程反過來作處理，以下將詳細介紹 JPEG 2000 解壓縮流程中的各個處理步驟。

■ EBCOT 解碼

EBCOT 解碼器(decoder)利用 EBCOT 編碼時所產生的封包，將封包所記錄的資料一一解開。首先經由 Tier-2 解碼器解碼出位元串流後，再利用 Tier-1 解碼器將位元串流解碼為編碼區塊。圖 2.7 所示為 EBCOT 的解碼過程。

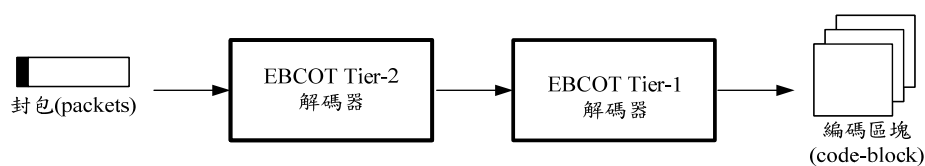


圖 2.7：EBCOT 解碼流程圖

■ 反量化

反量化主要用來將量化後的頻率值還原成原來的頻率值。反量化公式如公式(5)所示。其中 $q_b(n)$ 為在 EBCOT 的量化後的頻率值，其值與在編碼端所量化後的值相同， Δ_b 為量化步階值，其控制參數 (ϵ_b, μ_b) 會記錄在 JPEG 2000 影像檔頭內， $\hat{y}_b(n)$ 代表還原後的像素。

$$\hat{y}_b(n) = q_b(n) \times \Delta_b \quad (5)$$

■ 反向小波轉換

反向小波轉換主要用來將反量化後的頻率值轉換成空間域資訊。反向小波轉換先針對小波頻率資訊以每一直行進行一維反向小波轉換處理，當每一行都處理完，將所產生的資訊再以每一橫列方法進行一維反向小波轉換處理，最後得到空間域影像。反向小波轉換流程如圖 2.8 所示。

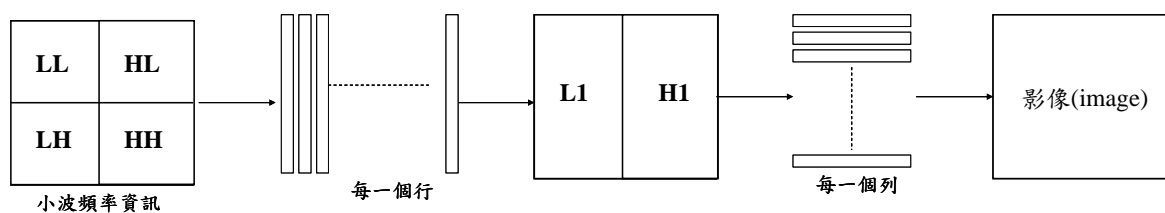


圖 2.8：二維反向小波轉換流程圖

在反向不可逆小波轉換時，所採用的公式如下：

$$\begin{aligned}
 s^{(1)}[n] &= \kappa \times a_1[n], \\
 d^{(1)}[n] &= d_1[n+1] / \kappa, \\
 s^{(2)}[n] &= s^{(1)}[n] + \delta(d^{(1)}[n-1] + d^{(1)}[n]), \\
 d^{(2)}[n] &= d^{(1)}[n] + \gamma(s^{(2)}[n] + s^{(2)}[n+1]), \\
 \hat{a}_0[2n] &= s^{(2)}[n] + \beta(d^{(2)}[n-1] + d^{(2)}[n]), \\
 \hat{a}_0[2n+1] &= d^{(2)}[n] + \alpha(a_0[2n] + a_0[2n+2])
 \end{aligned} \tag{6}$$

其中：

$$\begin{aligned}
 \alpha &= 1.586134312059924, \\
 \beta &= 0.052980118572961, \\
 \gamma &= -0.882911075530934, \\
 \delta &= -0.443506852043971, \\
 \kappa &= 1.230174104914001
 \end{aligned}$$

■ 反向色彩轉換

反向色彩轉換用來將 Y 、 C_b 、 C_r 色彩空間轉換成 R 、 G 、 B 色彩空間。

反向色彩轉換中將 Y 、 C_b 、 C_r 轉換 R 、 G 、 B 色彩空間的公式如下：

$$\begin{aligned}
 R &= Y + 1.402 C_r \\
 G &= Y - 0.34413 C_b - 0.71414 C_r \\
 B &= Y + 1.772 C_b
 \end{aligned} \tag{7}$$

■ 後置處理

後置處理主要對影像進行 DC 位移，將原本落在 $[-2^{P-1}, 2^{P-1}-1]$ 範圍的像素值，調整回 $[0, 2^P-1]$ 範圍。作法是將每個像素值加上 2^{P-1} 。另外需注

意到若使用 JPEG 2000 失真壓縮方式，在後置處理完之後，需加上修剪 (clipping)處理，以防止重建後的影像值超過範圍(需在 0~255 之間)。

第三節 JPEG 2000 檔案格式

上述提到 JPEG 2000 的編碼演算法是將影像以小波轉換分割成許多解析度(resolution)與子頻帶，每個子頻帶有許多編碼區塊，之後針對這些編碼區塊進行 EBCOT 編碼產生封包。

在 JPEG 2000 檔案格式中，封包會以兩種型式的檔頭表示。一個主檔頭(header)及另一個為塊狀部份檔頭(tile-part header)。主檔頭位於編碼串流(code-stream)的開頭，主要記錄原始影像大小、塊狀(tile)大小、編碼區塊長度、色彩轉換型態、小波轉換層數、與量化預設值等資訊。而塊狀部份檔頭則位於每個塊狀部份的開始。會使用到塊狀檔頭的原因：假如主檔頭裡有一個色彩元素，它的編碼方式不同於其它色彩元素，此時塊狀部份檔頭會指出這情形。以下將針對封包裡編碼串流的相關資訊作個詳細說明。圖 2.9 所示為主檔頭與部份檔頭的編碼串流結構[23]，其中每個方塊表示 JPEG 2000 檔頭中一個標記(描述區塊)。

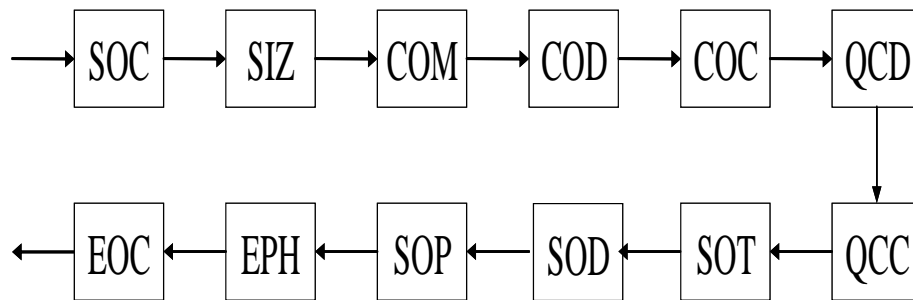


圖 2.9：編碼串流結構圖

圖 2.9 所示的各標記處理流程說明如下：

■ SOC(Start of Code-stream)

SOC 為編碼串流起始的標記，此標記屬於主檔頭。SOC 為編碼串流的第一個標記，而且只有一個。此標記用來表示此檔案為 JPEG 2000 檔案。

■ SIZ(Image and Tile size)

SIZ 為影像與塊狀大小標記，此標記屬於主檔頭，用來記錄原始影像的大小、塊狀寬與高、以及色彩元素等，會接著 SOC 標記段出現，並且只會出現一次。

■ COM(Comment)

COM 是註釋(Comment)標記，此標記屬於主檔頭或塊狀部份檔頭，用來記錄非結構化的註釋資料，可出現在檔頭及塊狀部份檔頭。

■ COD(Coding style Default)

COD 是指色彩轉換及小波轉換等所有資訊的標記，此標記屬於主檔頭。COD 會記錄可逆或不可逆的色彩轉換型態，小波轉換使用 5/3 可逆轉換或者使用 9/7 不可逆轉換方式，以及小波轉換時的層數等資訊。

■ COC(Coding style Component)

COC 為色彩轉換及小波轉換等所有資訊的標記，此標記屬於塊狀部份檔頭資訊。COC 用來記錄某一個特別色彩元素的編碼型態、小波轉換方式、與小波轉換的層數。若此時發現 COD 主檔頭資訊與 COC 塊狀部份檔頭資訊不合時，COC 塊狀部份檔頭資訊會指出不同處並重新記錄。

■ QCD(Quantization Default)

QCD 則是記錄量化預設值的標記，此標記屬於主檔頭，用來記錄某一個色彩元素的量化值，且只會出現一次(QCD 標記主要記錄 Y 影像的量化步階值)。此標記會記錄影像經小波轉換後每個子頻帶的量化步階值控制參數(ϵ_b, μ_b)。

■ QCC(Quantization Component)

QCC 是用來記錄某一個特定色彩元素量化方式的標記，QCC 標記屬於塊狀部份檔頭資訊(QCC 標記主要記錄 C_b 與 C_r 影像

的量化步階值)。QCC 與 COC 做法相同。若發現 QCD 主檔頭資訊與 QCC 塊狀部份檔頭資訊不合時，QCC 塊狀部份檔頭資訊會指出不同處並重新記錄。

■ SOT(Start of Tile-part)

SOT 標記塊狀部份的開始，此標記屬於塊狀部份檔頭資訊。每個編碼串流最少有一個 SOT。

■ SOD(Start of Data)

SOD 用來指出目前塊狀部份裡的資料開始的標記。此標記屬於塊狀部份檔頭資訊。

■ SOP(Start of Packet)

SOP 標記出編碼串流中一個封包開始。此標記是可選擇在主檔頭或者是塊狀部份檔頭。SOP 會記錄產生封包的總數，依總數來傳送。例如：SOP 記錄該封包有 6 塊，則會傳送 6 塊封包。

■ EPH(End of Packet header)

標記封包檔頭的結束，與 SOP 相對應。

■ EOC(End of Code-stream)

指出編碼串流的結束與 SOC 相對應，每一個編碼串流只有一個 EOC。

第四節 空間域轉換方法

空間域影像大小轉換流程如圖 2.10 所示。

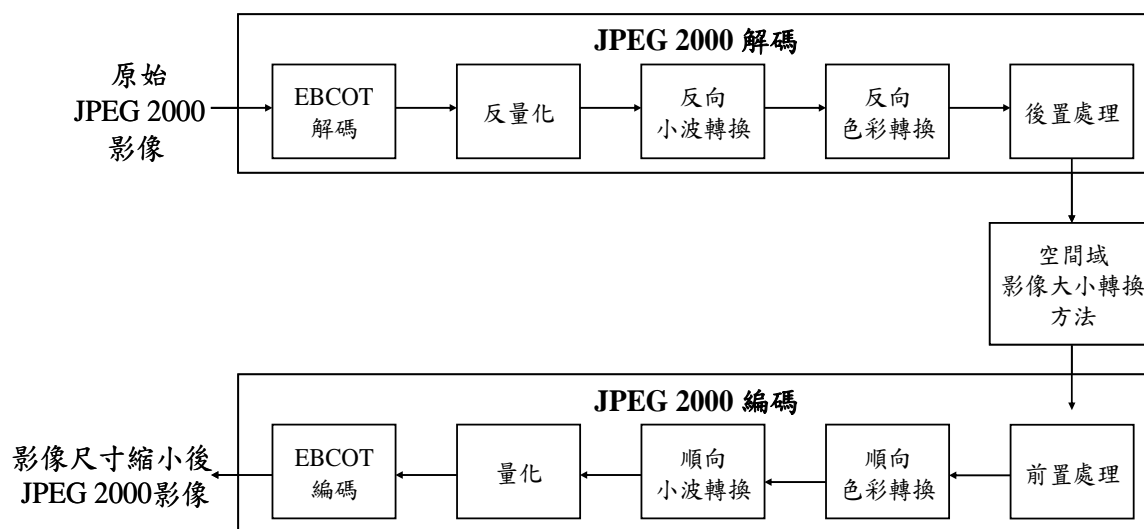


圖 2.10：空間域影像大小轉換流程圖

需要轉換大小的原始 JPEG 2000 影像，首先經過完整的 JPEG 2000 解碼過程，產生空間域的影像資訊，然後在空間域裡針對 JPEG 2000 影像進行大小轉換動作，最後再將調整後的空間域影像進行完整的 JPEG 2000 編碼過程，產生影像大小轉換後的 JPEG 2000 影像。

在空間域中針對影像進行縮放轉換的方法有很多種，以下介紹幾種較常見的方法：最近鄰居內插法[24-25]、雙線性內插法[24-25]、雙立方內插法[26]、與平均法，並且分別對這四種方法說明如下：

■ 最近鄰居內插法(Nearest Neighbor Interpolation)

最近鄰居內插法，又稱替代法，由最靠近該點的像素來決定新產生的像素值。最近鄰居內插法的作法是從原始影像中找出與新影像像素距離最接近的像素值做為新影像的像素值。以圖 2.11 為例，若 P 點為新影像某像素的座標，而 $I(x,y)$ 、 $I(x+1,y)$ 、 $I(x,y+1)$ 、與 $I(x+1,y+1)$ 為原始影像(I)裡 4 個像素的座標，則由於 $I(x,y)$ 的位置距 P 點最近，因此取原始影像 $I(x,y)$ 位置的像素值，做為新影像 P 點的像素值。

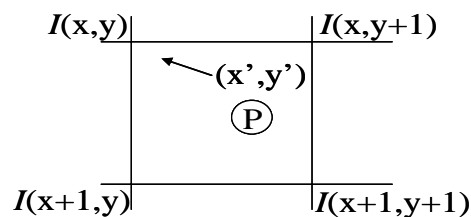
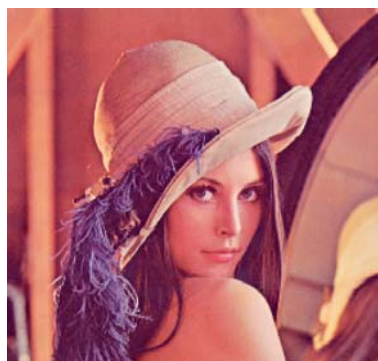


圖 2.11：最近鄰居內插法示意圖

使用最近鄰居內插法是最簡單也是最快的影像尺寸縮放方法，但缺點為若將影像尺寸放大，會容易產生明顯的鋸齒狀(如圖 2.12 所示)。



(a) 經處理後的 Lena 圖



(b) 局部放大

圖 2.12：使用最近鄰居內插法處理後的影像

■ 雙線性內插法(Bilinear Interpolation)

雙線性內插法其作法為利用 4 個鄰近像素值來計算出新的像素值。首先選擇最接近該新影像像素點的 4 個像素，再依照遠近不同，給予個別像素不同的比重(weight)，求出新的像素值。以圖 2.13 為例，取出原始影像(I)中 $I(x,y)$ 、 $I(x+1,y)$ 、 $I(x,y+1)$ 、 $I(x+1,y+1)$ 四個像素座標 P 的鄰近點，然後使用雙線性內插法以連續三次線性內插，算出最後新產生的值。

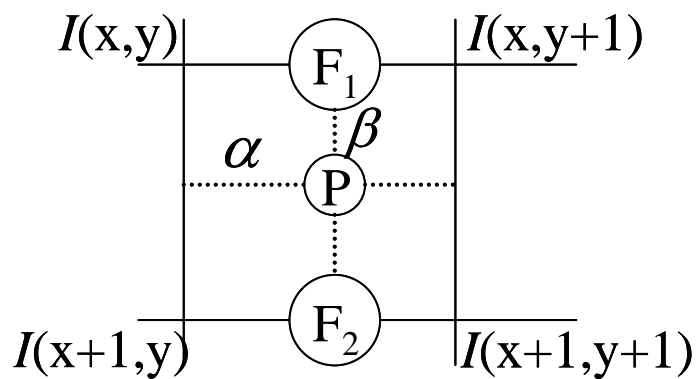


圖 2.13：雙線性內插法示意圖

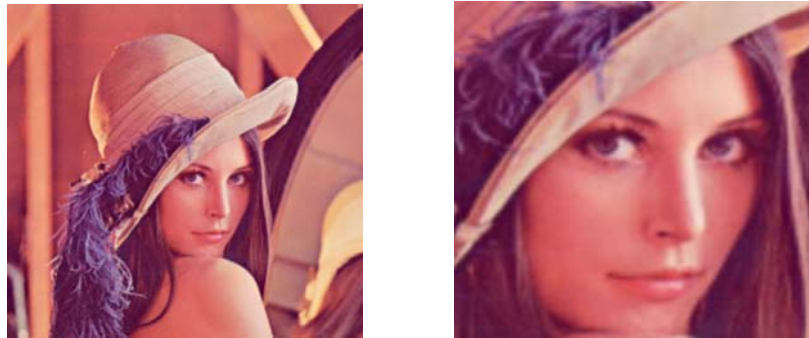
第一次的內插如公式(8)所示，以 $I(x,y)$ 與 $I(x,y+1)$ 兩點算出 F_1 點的像素值；接著第二次內插公式(9)為計算 $I(x+1,y)$ 與 $I(x+1,y+1)$ 兩點算出 F_2 的像素值；最後使用公式(10)將 F_1 與 F_2 兩點再做第三次內插，最後得到內插 P 點的像素值。

$$F_1 = (1 - \alpha) \times I(x, y) + (\alpha \times I(x, y + 1)) \quad (8)$$

$$F_2 = (1 - \alpha) \times I(x + 1, y) + (\alpha \times I(x + 1, y + 1)) \quad (9)$$

$$P = (1 - \beta) \times F_1 + \beta \times (F_2) \quad (10)$$

雙線性內插方法可改善使用最近鄰居內插法時所產生的鋸齒狀現象，但使用雙線性內插方法會造成放大後影像模糊(如圖 2.14 所示)，且需要較多的計算量。



(a) 經處理後的 Lena 圖

(b) 局部放大

圖 2.14：使用雙線性內插法處理後影像

■ 雙立方內插法(Bicubic Interpolation)

雙立方內插法是採用最靠近該點的 16 個像素貢獻程度，以權重平均(Weighted average)的方法，求出最後 P 的像素值。

雙立方內插法與雙線性內插法兩者作法類似，以 4 個像素點為主，分別擴大到 16 個像素。以圖 2.15 為例，P 值被包在原始影像(I)中 $I(x+1,y+1)$ 、 $I(x+1,y+2)$ 、 $I(x+2,y+1)$ 、與 $I(x+2,y+2)$ 這 4 點之內，並且這 4 點分別對外擴張，共可取得 16 個像素，再利用所取出這 16 個像素計算出最後 P 的像素值。

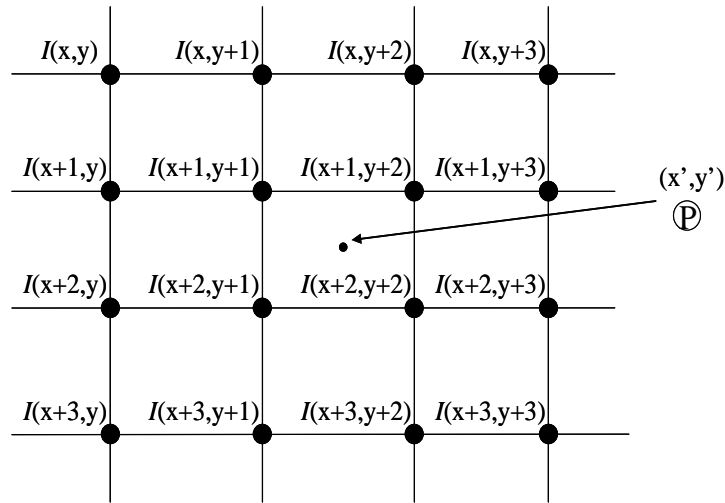


圖 2.15：雙立方內插法示意圖

令 P 點為新影像像素的座標 (x', y') 、 $x = \text{round}(x') - 1$ 與 $y = \text{round}(y') - 1$ ，則包圍 P 的 16 個整數座標點為 $I(x+a, y+b)$ ，其中 $a, b = 0, \dots, 3$ 。在 x 到 y 的方向以一維函數個別定義為公式(11)與(12)。

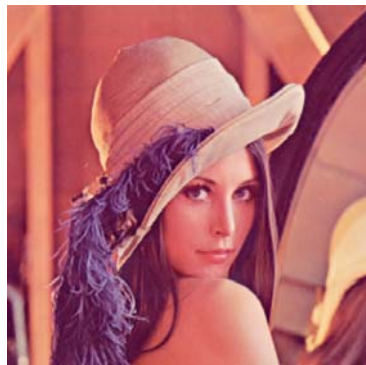
$$H_{x+a} = \begin{cases} 1 - 2|x-x'|^2 + |x-x'|^3 & , 0 \leq |x-x'| < 1 \\ 4 - 8|x-x'| + 5|x-x'|^2 - |x-x'|^3 & , 1 \leq |x-x'| < 2 \\ 0 & , \text{otherwise} \end{cases} \quad (11)$$

$$V_{y+b} = \begin{cases} 1 - 2|y-y'|^2 + |y-y'|^3 & , 0 \leq |y-y'| < 1 \\ 4 - 8|y-y'| + 5|y-y'|^2 - |y-y'|^3 & , 1 \leq |y-y'| < 2 \\ 0 & , \text{otherwise} \end{cases} \quad (12)$$

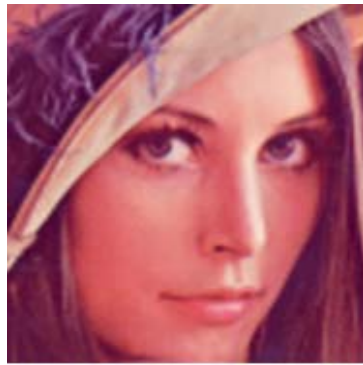
因此使用雙立方內插法公式如下所示：

$$P(x', y') = \sum_{a=0}^3 \sum_{b=0}^3 H_{x+a} V_{y+b} I(x+a, y+b) \quad (13)$$

利用雙立方內插法來做影像尺寸縮放的效果會比使用最近鄰居內插法與雙線性內插法更好，經放大後的影像不容易產生鋸齒狀，並且模糊的程度比使用雙線性內插法還來的輕微。如圖 2.16 所示為使用雙立方內插法對 Lena 影像進行放大後的結果。但使用雙立方內插法缺點會增加許多計算量，使得處理效率比較差。



(a) 經處理後的 Lena 圖



(b) 局部放大

圖 2.16：使用雙立方內插法處理後影像

■ 平均法(Averaging)

平均值法的觀念是要將影像縮小為原尺寸的 $1/s \times 1/t$ 時（其中 $s, t \geq 1$ ），直接將原影像切割成多個不重疊區塊，每個區塊大小為 $s \times t$ ，然後再計算每個區塊的平均值，計算出來的區塊平均值便是新影像的像素值。要將影像 I 縮小為原來的 $(1/s \times 1/t)$ 時，縮小後影像 I' 位於 (i, j) 座標的像素內容 $I'(i, j)$ 計算公式如下：

$$I'(i, j) = \frac{1}{s \times t} \left(\sum_{x=t \times i+1}^{t(i+1)} \sum_{y=s \times j+1}^{s(j+1)} I(x, y) \right) \quad (14)$$



第三章 快速 JPEG 2000 影像尺寸縮小轉換方法

一般在處理 JPEG 2000 影像時，若欲將其影像尺寸縮小，首先需經由 JPEG 2000 解碼器處理，將 JPEG 2000 影像解碼回空間域影像後，在空間域裡將影像縮小至所需尺寸後，再經 JPEG 2000 編碼器將影像作編碼，最後得到尺寸縮小後的壓縮影像。

由於在空間域裡使用影像大小轉換方法來縮小 JPEG 2000 影像，需要大量的計算量、繁雜處理流程、以及使用許多記憶體空間。為了加快縮小影像尺寸的處理速度、減少所需的計算量、以及有效降低記憶體使用量，為此，本論文提出一個快速的 JPEG 2000 影像尺寸縮小轉換方法，所提方法如圖 3.1 所示。

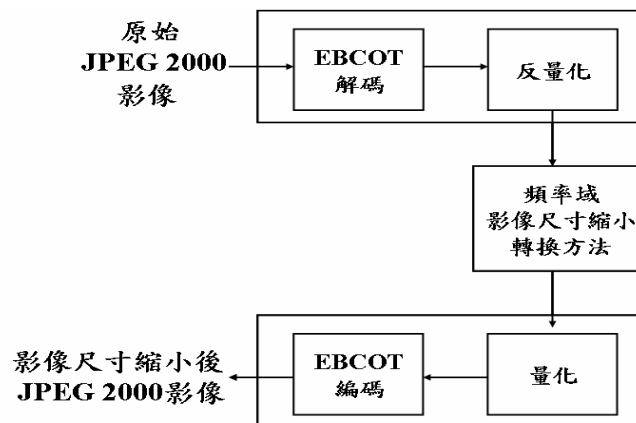


圖 3.1：快速 JPEG 2000 影像尺寸縮小轉換方法

在我們的快速 JPEG 2000 影像尺寸縮小轉換方法中，首先將原始 JPEG 2000 影像經 EBCOT 解碼以及反量化步驟解出影像的頻率域資訊後，再透過頻率域影像尺寸縮小轉換方法，直接在頻率域裡縮小影像尺寸，最後再經量化與 EBCOT 編碼等步驟，將影像尺寸縮小後的影像頻率域資訊編成 JPEG 2000 影像。

本論文所提的 JPEG 2000 影像尺寸縮小轉換方法與空間域影像大小轉換方法相比，所提的方法省掉反向小波轉換、反向色彩轉換、後置處理、前置處理、順向色彩轉換、以及順向小波轉換等六個步驟。由於所提的方法不需將頻率域資訊轉成空間域影像，因此本論文所提的方法除了可更快速的轉換影像大小外，也可省下存放空間域影像內容所需的記憶體空間以及減少所需的計算量。

第一節 簡化 JPEG 2000 壓縮與解壓縮流程

本論文所提快速 JPEG 2000 影像尺寸縮小轉換方法中(如圖 3.1 所示)，我們保留了 EBCOT 解碼、反量化、量化、與 EBCOT 編碼等四個部份，主要原因說明如下：

■ EBCOT 編/解碼

JPEG 2000 編碼後的影像會儲存成封包的格式，但封包並非以子

頻帶為單位儲存，所以要取得各子頻帶的內容，必須先經過 EBCOT 解碼才行。再者本論文的方法有可能需要對子頻帶再進行小波轉換，因此 EBCOT 編/解碼過程不可省略。

■ 量化與反量化

保留量化與反量化步驟的主要原因在於影像經由順向小波轉換後，會產生不同大小的子頻帶頻率資訊，不同子頻帶頻率資訊使用不同的量化步階值進行量化。

子頻帶與量化步階值這兩者有相對應關係，換句話說以具有 7 個子頻帶的 JPEG 2000 影像而言，必須要有 7 個相對應的量化步階值。而子頻帶與量化步階值所產生的數目與小波轉換的層數有關，對於一個經過 m 層小波轉換的影像，所具有的子頻帶數目 $N_{subbands}$ 計算公式如下：

$$N_{subbands} = 3 \times m + 1 \quad (15)$$

圖 3.2 所示為影像經由二次小波轉換後所產生的七個不同的子頻帶。



圖 3.2：經二次小波轉換後產生 7 個子頻帶示意圖

每個子頻帶的量化步階值都是由一組獨立的控制參數(ϵ , μ)決定，該組控制參數必須記錄於 JPEG 2000 檔頭，供解碼端還原量化步階值使用。圖 3.3 所示為一張影像經過三次小波轉換後所產生的頻率域資訊，而圖 3.4 所示為圖 3.3 每個子頻帶所使用的量化步階值的一個例子，其中較大的量化步階值，對資料具有較大的壓縮效果，也會造成較大的資料失真；反之，資料壓縮效果較差，資料失真程度也較小。

LL0	HL0	HL1	HL2
LH0	HH0		
LH1		HH1	
LH2		HH2	

圖 3.3：經三次小波轉換後的 JPEG 2000 影像頻率域資訊

24372	24396	18435	20434
24396	24418		
18435		18501	
20434		20320	

圖 3.4：三層子頻帶使用不同量化步階值的例子

本論文所提的頻率域影像尺寸縮小方法會改變原本影像的小波轉換層數，進而影響到量化步階值與子頻帶的對應關係。如圖 3.4 與 3.5 所示為使用不同小波轉換層數時，每個子頻帶的量化步階值會不同。所以，當影像在進行尺寸縮小前，先使用原本 JPEG 2000 影像的量化步階值對影像進行反量化，還原頻率域資訊，當影像尺寸已調整縮小後，再用新

的量化步階值來量化頻率域資訊，即可解決量化步階值與子頻帶不一致的問題。

22406	18435	20434
18435	18501	
20434		20320

圖 3.5：二層子頻帶使用不同量化步階值

在我們所提的方法中，分別會遇到小波層數足夠與小波層數不足的情況。假設一張 JPEG 2000 影像小波層數為 m 層，欲將影像尺寸縮小為原來的 $(1/2^n \times 1/2^n)$ 大小時，假如 $n < m$ ，則表示 JPEG 2000 影像小波層數足夠。作法是將 EBCOT 解碼後，對整張影像作反量化，接著將不需要的子頻帶丟棄，然後再將沒有被丟棄的子頻帶資訊進行量化與 EBCOT 編碼，即可完成影像尺寸縮小的動作。例如：圖 3.6 所示為具有二層小波轉換的 JPEG 2000 影像。若要將影像縮小成原來的 $1/2 \times 1/2$ 大小時，需先經由 EBCOT 解碼，產生不同的子頻帶資訊，然後對每個子頻帶進行反量化動作，接著將不需要的子頻帶資訊丟棄。而保留的頻率域資訊，最後再經由重新量化與 EBCOT 編碼完後，可得到尺寸縮小後的 JPEG 2000 影像。

由圖 3.6 可知，因為 $n < m$ ，且 m 個外層的中高頻資訊最終將被丟棄，所以在反量化過程中只需針對要保留的頻率域資訊進行反量化，而不必對 m 個外層頻率作反量化動作，以加快影像尺寸縮小轉換的速度。

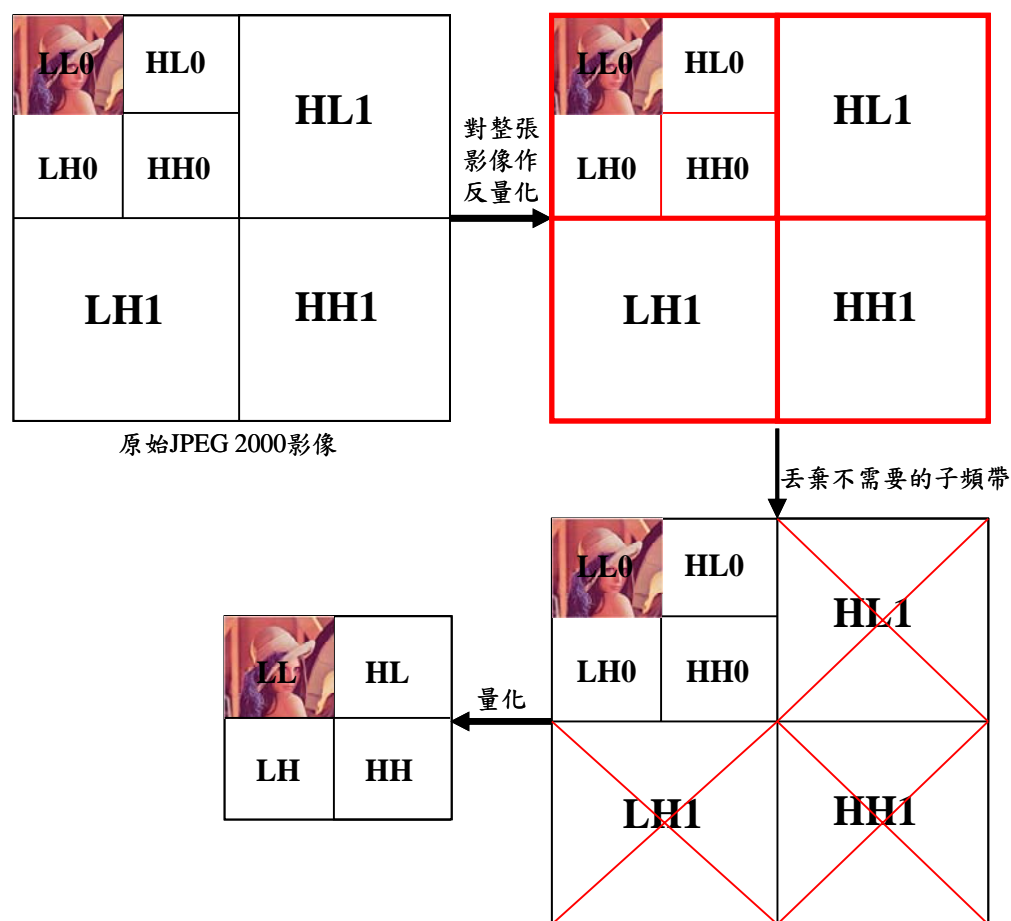


圖 3.6：對不同的子頻帶做不同的反量化示意圖

若 $n \geq m$ 發生，也就是小波層數不足。例如圖 3.7 為一張具有一層小波轉換的 JPEG 2000 影像，且影像大小為 512×512 。要將影像縮小成原來的 $1/4 \times 1/4$ 大小 (128×128) 時，首先經 EBCOT 解碼後，產生不同的子頻帶資訊。針對不同的子頻帶資訊使用反量化，接著進行影像縮小的工作，

將不需要的外頻資訊去除(如圖 3.7 中原始 JPEG 2000 影像 HL0、LH0、與 HH0)。保留的頻率資訊(如圖 3.7 中原始 JPEG 2000 影像 LL0)因小波層數不足(小波層數需為 1 層以上)，因此要對保留的頻率再進行小波轉換。產生出來的小波頻率域尺寸為 256×256 大小超過欲轉換尺寸的 128×128 大小，因此可將外頻的小波頻率資訊去除，保留 LL 子頻帶(大小為 128×128)。此時影像大小雖然已符合所需轉換大小，但 JPEG 2000 規定影像至少要有一層小波轉換，所以必須再做一次小波轉換，得到一張小波轉換層數為 1 層的 JPEG 2000 影像，最後再經量化與 EBCOT 編碼，得到尺寸縮小後的 JPEG 2000 影像。

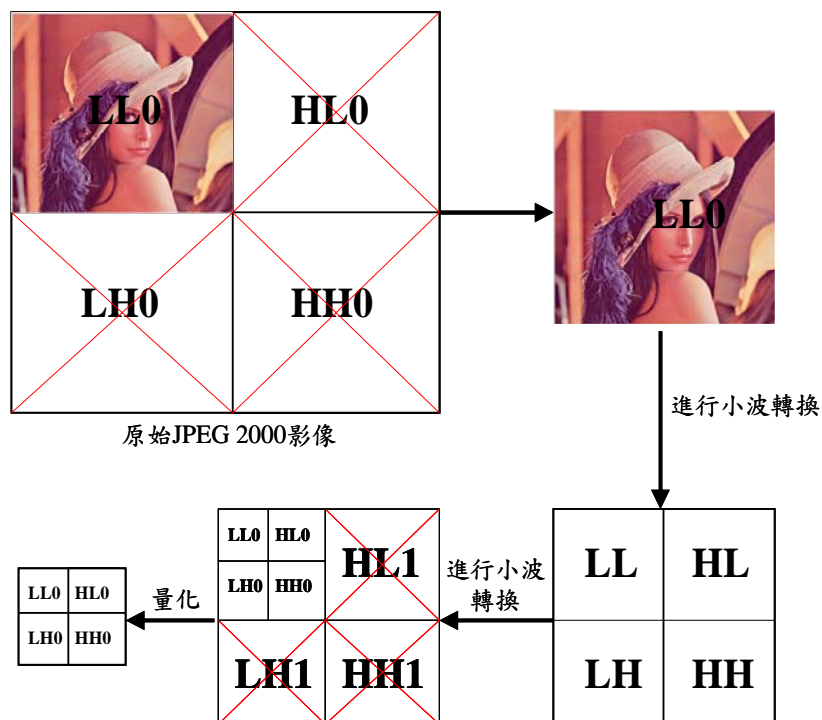


圖 3.7：針對層數不足的影響作小波轉換方法示意圖

由圖 3.7 所示，原始 JPEG 2000 影像裡 HL0、LH0、與 HH0 在影像尺寸轉換過程中終究會被丟棄，因此在反量化過程中，可不需對這三個子頻帶做反量化的動作。

第二節 頻率域影像尺寸縮小轉換方法

圖 3.1 中間的頻率域影像尺寸縮小轉換方法主要工作包括縮小頻率域影像尺寸與修改 JPEG 2000 影像檔頭相關參數等步驟。詳細步驟說明如下：

壹、縮小頻率域影像尺寸

一、小波轉換層數足夠的作法

假設當影像的小波層數為 m 層，欲將影像尺寸縮小為 $(1/2^n \times 1/2^n)$ 大小時，若 $n < m$ ，則表示小波轉換層數足夠。

圖 3.8 所示為小波轉換層數為二層時的頻率域資訊，若要將影像縮小成原來的 $1/2 \times 1/2$ 大小，首先使用 EBCOT 解出頻率域資訊，再對需保留的頻率域資訊作反量化動作，接著將整張影像的尺寸縮小，並且丟棄不需要外頻頻率資訊，最後將所保留的頻率域資訊再重新經過量化與 EBCOT 編碼，即可得到影像尺寸縮小後的 JPEG 2000 影像。

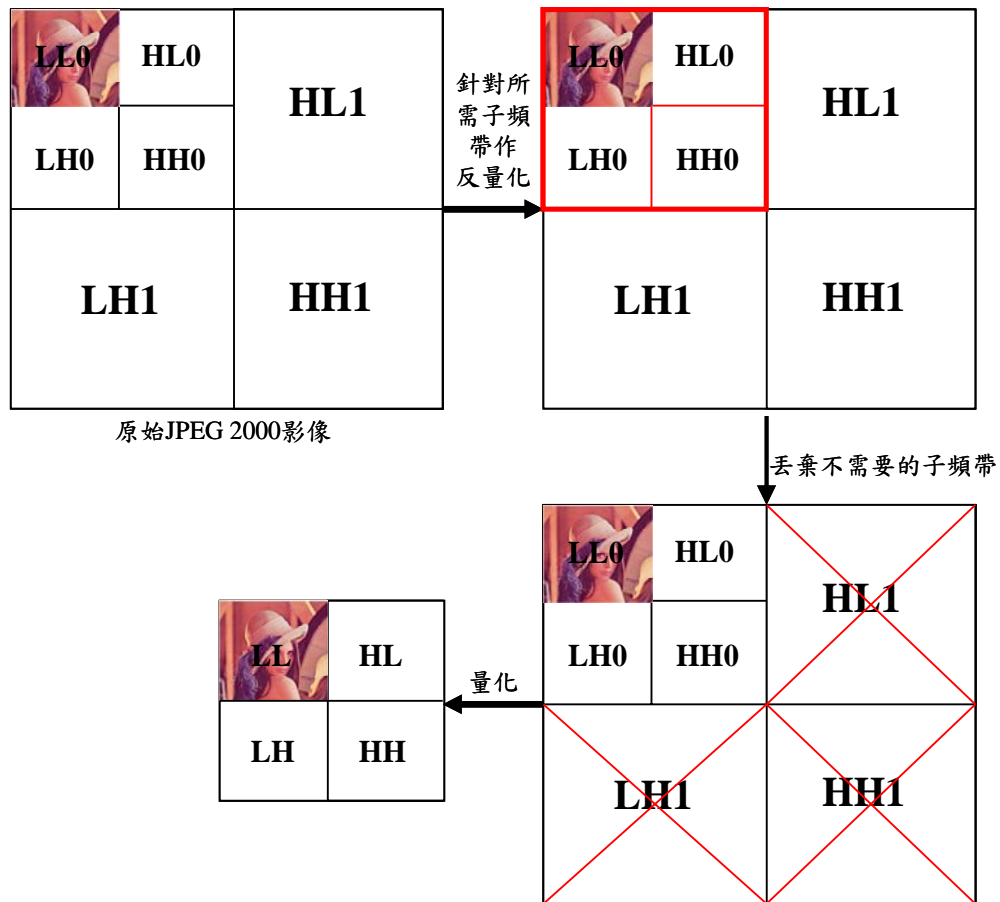


圖 3.8：針對保留子頻率資訊作反量化示意圖

二、小波轉換層數不足的作法

假設當影像的小波層數為 m 層時，欲將影像尺寸縮小為 $(1/2^n \times 1/2^n)$ 大小時，若 $n \geq m$ 發生，就是小波層數不足，則除了丟棄 m 個外層的中高頻資訊外，還需要將原來最內層的低頻資訊，進行 $(n-m)+1$ 次小波轉換，再將所產生的 $(n-m)$ 層的中高頻資訊丟棄才行(如圖 3.9 所示)。由於以上的 $(n-m)$ 次小波轉換後的中高頻資訊最終將被丟棄，因此在進行以上小波轉換時可直接省略許多

計算工作，不必進行完整的小波轉換。為此本論文設計了一個快速小波轉換方法。

使用快速小波轉換方法的流程說明如下，例如：圖 3.9 為一層小波轉換的頻率域資訊，影像大小為 512×512 。要將頻率域資訊縮小成原來的 $1/4 \times 1/4$ 大小(128×128)時，需先丟棄原始 JPEG 2000 影像中的 HL0、LH0、與 HH0 等子頻帶，保留 LL0 子頻帶資訊，針對保留的頻率進行小波轉換，由於此時所產生出來的小波轉換後的中高頻資訊需被丟棄，因此在進行這一層小波轉換時，可使用快速小波轉換。首先針對保留 LL0 子頻帶資訊先進行水平小波轉換產生左右兩邊分別為 L 與 H 的子頻帶，然後再丟棄 H 子頻帶，保留 L 子頻帶。由於 H 子頻帶會被丟棄，因此在做水平小波轉換時，H 部份的計算可以省略不做。同樣的針對保留的 L 子頻帶，只需對上半部進行垂直小波轉換，下半部的子頻帶也可省略不算。

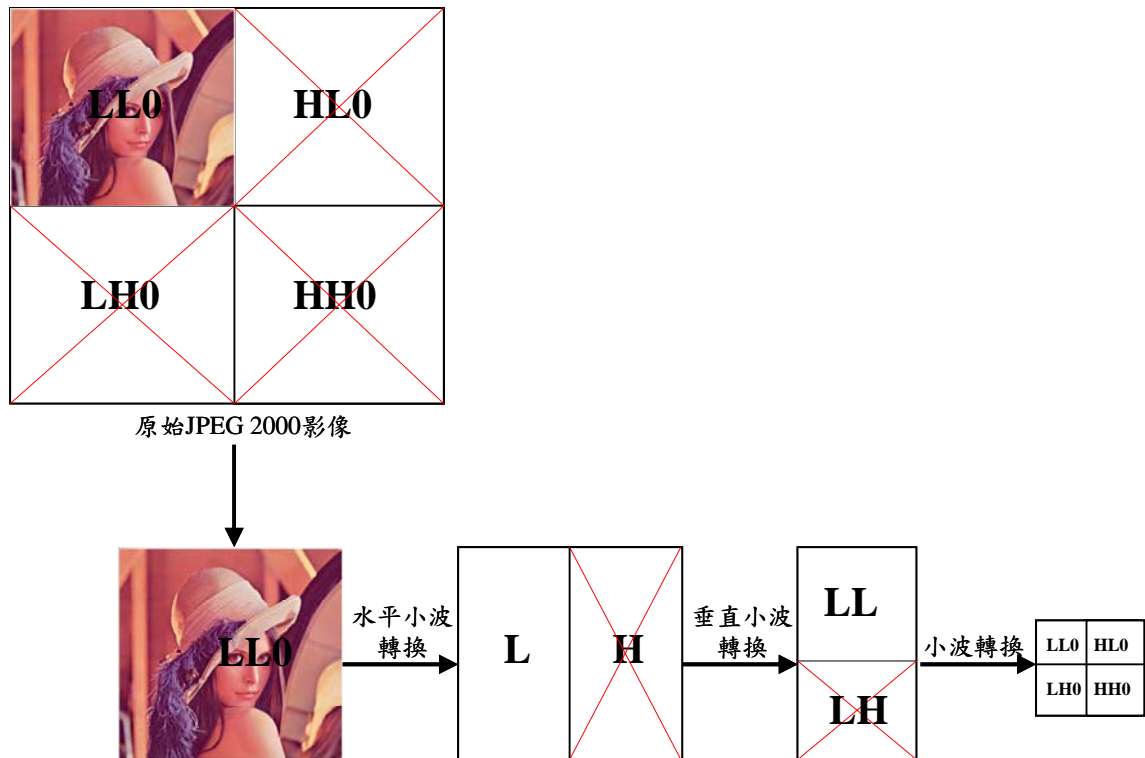


圖 3.9：快速小波轉換方法示意圖

貳、修改 JPEG 2000 影像檔頭相關參數

JPEG 2000 影像檔頭記錄原始影像大小、塊狀(tile)大小、小波層數、與各子頻帶的量化步階值參數(ϵ 與 μ)等資訊。在我們所提的方法中，並沒有將影像解回空間域，而是在頻率域資訊縮小影像尺寸後，直接進行量化與 EBCOT 編碼，產生新的 JPEG 2000 影像。新的 JPEG 2000 影像檔頭資訊無法像空間域轉換方法由 JPEG 2000 壓縮程式設定，而必須自行修改 JPEG 2000 影像檔頭內的相關參數。

必須修改的 JPEG 2000 影像檔頭內的相關參數包含 SIZ、COD、與 QCD 和 QCC 這四 4 標記，這 4 個標記的修改方式說明如下：

- SIZ：SIZ 標記用來記錄整張影像的尺寸，因此若要將影像縮小為原本的 $(1/2^n \times 1/2^n)$ 大小，需修改 SIZ 裡記錄影像大小的 Xsiz、Ysiz、XTsiz、與 YTsiz 等這 4 個參數值的值。
- COD：COD 標記用來記錄影像色彩、小波轉換使用 9/7 不可逆轉換或者使用 5/3 可逆轉換方式、與小波轉換的層數。本論文所提方法裡，會對影像作縮小轉換的動作，但此動作會導致原本 COD 記錄的小波轉換層數與縮小後影像的小波轉換層數不一致，所以解決層數不一致的問題，必需修改原本 COD 標記裡所記錄的小波轉換層數。
- QCD 與 QCC：QCD 與 QCC 兩者標記主要分別記錄 Y 、 C_b 、 C_r 三種色彩平面影像的量化步階值。量化步階值的產生由小波層數所決定。因此我們除了修改小波層數之外，對於不同的子頻帶的量化步階值也需重新調整。

四、實驗結果

第一節 實驗環境

為了評估本論文所提方法與空間域影像尺寸轉換方法的效能差異，本實驗使用 6 張標準影像(Lena, Peppers, Baboon, Tiffany, Parrot, Houses) 作為測試影像，測試影像大小均為 512x512。本實驗修改開放軟體 Jasper 1.900.1[27]實作出空間域影像尺寸縮小轉換程式與頻率域影像尺寸縮小轉換程式。其中，空間域影像尺寸縮小轉換方法的實作方式，使用最簡單的平均法與最近鄰居內插法[24-25]。

實驗方法為先將測試影像轉換成 JPEG 2000 影像，所使用的編碼方式為不可逆轉換，小波轉換層數分別產生 1 層與 7 層，然後再分別使用平均法、最近鄰居內法、以及我們的方法(頻率域影像尺寸轉換方法)將影像尺寸縮小為原來的(1/2x1/2)、(1/4x1/4)、(1/8x1/8)、(1/16x1/16)、與(1/32x1/32)等大小。

為比較各個方法在執行時間上的表現，所有實驗均在同一台個人電腦下執行，所取得的執行時間為執行 10 次後取平均的結果(包括讀、寫檔案的時間)。所使用的電腦配備為 Pentium D 2.8 GHz CPU，內含 1GB 記憶體，作業系統為 Microsoft Windows XP Professional，程式開發環境為 Microsoft Visual C++ 6.0。

第二節 空間域執行時間

使用空間域轉換影像大小的實驗中，我們分別使用平均法與最近鄰居內插法來縮小影像尺寸，並且分別使用 JPEG 2000 影像小波層數為 7 層與 1 層的測試影像來測試縮小不同尺寸所需的執行時間。

表 4.1 到表 4.4 為使用 JPEG 2000 影像小波層數為 7 層時，所產生的實驗數據，表 4.5 到表 4.8 為使用 JPEG 2000 影像小波層數為 1 層時，所產生的實驗數據。表 4.1 與表 4.5 分別為小波層數 7 層與 1 層時，所使用平均法縮小不同尺寸時所需的執行時間，表 4.2 與表 4.6 為使用平均法產生的檔案大小。表 4.3 與表 4.7 分別為小波層數 7 層與 1 層時，使用最近鄰居內插法縮小不同尺寸時所需的時間，以及表 4.4 與表 4.8 為使用最近鄰居內插法產生的檔案大小。

從實驗數據得知，最近鄰居內插法的執行時間與平均法的執行時間差異不大，但使用平均法比使用最近鄰居內插法可得到更大的資料壓縮量。所以在後面的章節，僅使用平均法與本論文所提的頻率域影像尺寸縮小方法作比較。

表 4.1：使用平均法縮小影像尺寸平均所需執行時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	579.5	544.5	510.5	562.0	537.0	531.0	544.1
1/4×1/4	5	422.6	437.0	421.0	437.0	427.1	437.0	430.3
1/8×1/8	4	399.6	399.6	396.3	402.6	397.9	394.8	398.5
1/16×1/16	3	394.7	393.3	394.7	396.2	396.3	393.2	395.2
1/32×1/32	2	391.6	390.0	393.2	390.0	390.0	391.6	391.1

表 4.2：使用平均法縮小影像尺寸產生的檔案大小(單位：KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	70	47	27	46	43	42	46
1/4×1/4	5	21	16	11	17	14	13	15
1/8×1/8	4	6	5	5	6	5	5	5
1/16×1/16	3	2	2	2	2	2	2	2
1/32×1/32	2	1	1	1	1	1	1	1

表 4.3：使用最近鄰居內插法縮小影像尺寸平均所需執行時間

(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	578.0	549.6	507.5	562.0	531.0	531.0	542.9
1/4×1/4	5	433.8	429.0	422.6	432.2	424.2	424.2	427.7
1/8×1/8	4	393.2	398.0	393.2	404.4	393.2	393.2	395.7
1/16×1/16	3	385.5	390.1	385.5	390.0	376.5	379.5	384.5
1/32×1/32	2	376.5	388.5	378.5	388.5	378.5	379.5	381.5

表 4.4：使用最近鄰居內插法縮小影像尺寸產生檔案大小(單位:KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	74	52	32	54	51	52	53
1/4×1/4	5	27	19	13	22	19	19	20
1/8×1/8	4	8	7	5	8	6	6	7
1/16×1/16	3	3	2	2	3	2	2	2
1/32×1/32	2	1	1	1	1	1	1	1

表 4.5：使用平均法產生一層小波轉換 JPEG 2000 影像平均所需執行時間

(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	571.6	538.5	515.0	538.5	521.4	512	534.7
1/4×1/4	1	407.5	406.0	391.6	396.4	390.0	393.2	397.5
1/8×1/8	1	375.0	373.4	371.8	363.8	362.2	362.2	368.1
1/16×1/16	1	359.0	363.8	360.6	359.0	359.0	359.0	360.1
1/32×1/32	1	375.4	363.8	360.6	357.4	354.2	347.8	356.9

表 4.6：使用平均法產生一層小波轉換 JPEG 2000 影像檔案大小

(單位:KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	77	60	41	63	58	58	60
1/4×1/4	1	23	17	13	19	16	16	18
1/8×1/8	1	6	5	5	6	5	5	6
1/16×1/16	1	2	2	2	2	2	2	2
1/32×1/32	1	1	1	1	1	1	1	1

表 4.7：使用最近鄰居內插法產生一層小波轉換 JPEG 2000 影像平均所需

執行時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	578.0	544.5	501.5	547.6	534.0	531.0	538.9
1/4×1/4	1	404.4	402.8	388.5	396.4	391.6	387.0	395.1
1/8×1/8	1	371.8	371.8	365.4	365.4	360.6	360.6	365.9
1/16×1/16	1	355.8	363.8	354.2	357.4	346.2	344.6	353.7
1/32×1/32	1	354.2	354.5	352.6	354.2	347.8	346.2	351.6

表 4.8：使用最近鄰居內插法產生一層小波轉換 JPEG 2000 影像檔案大小

(單位:KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	77	73	53	77	76	76	72
1/4×1/4	1	30	22	17	24	22	22	23
1/8×1/8	1	8	7	6	8	7	7	8
1/16×1/16	1	3	2	2	3	2	2	3
1/32×1/32	1	1	1	1	1	1	1	1

第三節 小波層數足夠所需執行時間

所提的頻率域轉換影像尺寸縮小轉換方法中，我們分別測試對整張影像作反量化與只對保留子頻帶作反量化這兩種方法所需的執行時間，並且使用 JPEG 2000 影像小波層數為 7 層來測試當小波層數足夠時，利用不同影像進行尺寸縮小轉換方式。

表 4.9 與表 4.10 分別為對整張影像作反量化與只對保留子頻帶作反量化兩種方法縮小影像尺寸所需的執行時間，以及表 4.11 為影像尺寸縮小後的 JPEG 2000 影像檔案大小。從實驗數據中可發現，只針對保留子頻帶作反量化比對整張影像作反量化平均執行時間能節省 12% 的執行時間(如表 4.12 所示)，使用只針對保留子頻帶作反量化的方法與平均法相比，所提的方法平均執行時間可節省 73% 的執行時間，且兩者所產生的檔案大小差不多。

由此可見，我們的方法明顯比空間域的方法有更快的處理速度，而且有不錯的資料壓縮量。

表 4.9: JPEG 2000 影像小波層數足夠-對整張影像作反量化平均所需執行時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	223.1	210.6	202.9	224.4	209.0	206.1	212.7
1/4×1/4	5	125.0	128.0	126.5	131.0	128.0	126.5	127.5
1/8×1/8	4	102.6	109.1	109.0	112.2	105.8	109.0	108.0
1/16×1/16	3	99.4	104.2	105.9	102.6	101.0	105.8	103.2
1/32×1/32	2	93.0	96.2	97.8	99.4	96.2	96.4	96.5

表 4.10: JPEG 2000 影像小波層數足夠-對保留子頻帶做反量化平均所需執行時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	215.3	193.5	171.0	210.5	198.2	193.4	196.9
1/4×1/4	5	115.4	117.1	112.2	123.5	117.0	110.7	115.9
1/8×1/8	4	87.0	93.1	93.3	94.6	91.6	93.1	92.1
1/16×1/16	3	85.5	88.5	90.0	93.0	88.5	88.5	89.0
1/32×1/32	2	81.0	85.5	85.5	85.5	84.0	84.0	84.3

表 4.11: 表 4.9 與表 4.10 產生的檔案大小(單位:KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	6	50	41	26	42	43	42	42
1/4×1/4	5	18	16	11	18	15	14	23
1/8×1/8	4	7	5	5	7	5	5	6
1/16×1/16	3	2	2	2	2	2	2	2
1/32×1/32	2	1	1	1	1	1	1	1

表 4.12：表 4.9 與表 4.10 之轉換執行時間比(單位千分之一秒)

縮小尺寸	對整張影像作反量化 平均執行時間	對保留子頻帶做反量化 平均執行時間	使用對保留子頻帶做反 量化可節省執行時間(%)
1/2×1/2	212.7	196.9	7
1/4×1/4	127.5	115.9	9
1/8×1/8	108.0	92.1	15
1/16×1/16	103.2	89.0	14
1/32×1/32	96.5	84.3	13
使用對保留子頻帶做反量化平均可節省執行時間(%)			11.6

第四節 小波層數不足所需執行時間

針對小波層數不足方面，我們使用原本小波轉換與快速小波轉換來比較當小波層數不足時，使用兩者作法平均所需的執行時間。以上實驗將針對作反量化過程中，只對保留頻率域資訊進行反量化的動作。

表 4.13 與表 4.14 分別為使用原本小波轉換與快速小波轉換平均執行的時間、以及表 4.15 為影像尺寸縮小後的 JPEG 2000 影像檔案大小。從表 4.13 與表 4.14 的實驗結果顯示(其中使用一般或快速小波將影像縮小為 1/2×1/2 大小時，因為只作一層的小波轉換，所以速度會相同，因此本論文將不計算)，使用快速小波轉換的方法比使用傳統小波轉換方法，可節省 6% 的執行時間(如表 4.16 所示)。比較表 4.14 與平均法兩種方法的執行時間，使用我們的方法平均所需的執行時間比使用平均法所需的執行時間可節省 63% 的執行時間，並且我們的方法與平均法這兩種方法所產

生的檔案大小是差不多。

表 4.13：JPEG 2000 影像小波層數不足-使用一般小波轉換平均所需執行
時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	285.5	257.0	224.4	262.0	250.0	248.4	254.6
1/4×1/4	1	146.4	148.0	144.8	148.0	146.4	141.6	145.9
1/8×1/8	1	125.0	129.5	128.0	129.5	128.0	125.0	127.5
1/16×1/16	1	117.0	126.5	123.4	125.0	121.8	121.8	122.6
1/32×1/32	1	113.8	123.4	120.2	121.8	120.2	118.7	119.7

表 4.14：JPEG 2000 影像小波層數不足-使用快速小波轉換平均所需執行
時間(單位:千分之一秒)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	288.5	257.0	221.3	262.0	249.9	248.4	254.5
1/4×1/4	1	143.2	143.2	138.5	144.8	140.0	135.5	140.9
1/8×1/8	1	113.8	125.0	118.6	121.8	118.6	115.4	118.9
1/16×1/16	1	112.2	115.4	115.4	112.2	112.2	110.6	113.0
1/32×1/32	1	109.0	113.8	112.2	112.2	110.6	110.6	111.4

表 4.15：表 4.12 與 4.13 產生的檔案大小(單位：KB)

縮小尺寸	層數	Banboo	Houses	Parrot	Peppers	Lena	Tiffany	平均
1/2×1/2	1	77	65	46	72	67	67	66
1/4×1/4	1	24	18	13	19	17	17	18
1/8×1/8	1	6	5	5	7	5	5	6
1/16×1/16	1	2	2	2	2	2	2	2
1/32×1/32	1	1	1	1	1	1	1	1

表 4.16：表 4.12 與表 4.13 之轉換執行時間比(單位千分之一秒)

縮小尺寸	一般小波平均執行時間	快速小波平均執行時間	使用快速小波可節省報 行時間(%)
1/4×1/4	145.9	140.9	3
1/8×1/8	127.5	118.9	7
1/16×1/16	122.6	113.0	8
1/32×1/32	119.7	111.4	7
使用快速小波平均可節省執行時間(%)			6.25

第五節 總結

表 4.1 與表 4.16 分別列出使用空間域轉換的平均法、最近鄰居內插法、與頻率域各種轉換方法所需的平均執行時間，以及空間域轉換方式與頻率域各種轉換方法所產生檔案的平均檔案大小。從表 4.1 到表 4.8 實驗數據觀察到，使用最近鄰居內插法來轉換影像尺寸跟使用平均法來轉換影像尺寸的速度是差不多，但是使用平均法所產生的檔案比使用最近鄰居內插法所產生的檔案比較小。因此對資料壓縮而言，平均法所產生的影像壓縮後的檔案大小比最近鄰居內插法所產生的影像壓縮後的檔案大小來的好。表 4.9 到表 4.16 實驗數據中顯示，當層數足夠時，在頻率域裡針對保留子頻帶作反量化情況下，可節省約 12% 的執行時間。而層數不足時，在頻率域裡針對保留子頻帶作反量化、以及使用快速小波轉換等情況下，可節省約 6% 所需的執行時間

從所有的實驗數據得知，我們的方法比在空間域進行影像尺寸縮小轉換方法更有效率，我們的方法可節省 53%~78%的執行時間，而且可以維持良好的資料壓縮量。

為了說明所實作的方法確實可正確轉換 JPEG 2000 影像，圖 4.1 列出分別使用平均法與最近鄰居內插法、以及頻率域影像尺寸縮小轉換方法，其中頻率域影像尺寸縮小轉換方法為使用 JPEG 2000 影像小波層數足夠-對保留的子頻帶做反量化來進行頻率域轉換。這三種方法所使用的影像皆為測試小波層數為 7 層的 JPEG 2000 影像。並且將原始 Lena 影像縮小成原來的 $(1/2 \times 1/2)$ 大小後的結果，可由圖 4.1 所見，這三種方法皆可正常運作，且產生後的結果也非常相似。



圖 4.1：Lena 測試不同縮小影像尺寸範例：(a)平均法轉換方法產生後的結果(b)最近鄰居內插法產生後的結果,(c)頻率域轉換方法產生後的結果

五、結論與未來展望

在數位影像普及的今日，將數位影像傳送到不同的用戶端裝置變得越來越頻繁與便利。使用者只要簡單經由網路連線，即可接收與觀看數位影像。

數位影像縮小尺寸的技術在數位影像應用中是很重要的一項技術。為了快速將數位影像尺寸縮小，本論文提出一個新的快速影像尺寸縮小方法。所提的方法利用頻率域資訊的影像，將影像尺寸快速縮小，由於在頻率域資訊的作法能降低使用空間域轉換時的處理時間、以及所需的記憶體空間。實驗結果顯示，我們的方法比空間域轉換的方法，平均可節省 53%~78% 處理時間。

本論文所提的方法只針對靜態影像作固定大小的縮小轉換動作，無法對影像作任意大小轉換，對影像作任意大小轉換是一個很好的未來發展方向。另外，本論文所提方法沒有考慮輸出影像的位元率，如何有效整合位元率控制及配合調整頻率域資訊，也是個很好的發展方向。

參考文獻

- [1] Carlos Salazar and Trac D. Tran, "A complexity scalable universal DCT domain image resizing algorithm," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 4, pp. 495-499, April 2007.
- [2] Vasant Patil, Rajeev Kumar and Jayanta Mukherjee, "A fast arbitrary factor video resizing algorithm," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 9, pp. 1164-1171, September 2006.
- [3] Haiyan Shu and Lap-Pui Chau, "The realization of arbitrary downsizing video transcoding," *IEEE Transactions On Circuits and Systems for Video Technology*, vol. 16, no. 4, pp. 540-546, April 2006.
- [4] Young Seo Park and Hyum Wook Park, "Arbitrary-Ratio Image Resizing Using Fast DCT of Composite Length For DCT-Based Transcoder," *IEEE Transaction on Image Processing*, vol. 15, no. 2, pp. 494-500, February 2006.
- [5] YongQing Liang, Lap-Pui Chau and Yap-Peng Tan, "Arbitrary Downsizing Video Transcoding Using Fast Motion Vector Reestimation," *IEEE Signal Processing Letter*, vol. 9, no. 11, pp. 352-355, November 2002.
- [6] Yap-Peng Tan and Haiwei Sun, "Fast Motion Re-Estimation for Arbitrary Downsizing Video Transcoding using H.264/AVC Standard," *IEEE Transactions On Consumer Electronics*, vol. 50, no. 3, pp.887-894, August 2004.
- [7] Yuh-Reuy Lee, Chia-Wen Lin, Sung-Hung Yeh and Yung-Chang Chen, "LOW-COMPLEXITY DCT-DOMAIN VIDEO TRANSCODERS FOR ARBITRARY-SIZE DOWNSCALING," *IEEE 6th Workshop on Multimedia Signal Processing*, pp. 31-34, September 2004, Siena, Italy.

- [8] David Taubman and M. Marcellin, *JPEG 2000: image compression fundamentals, standards and practice*, Kluwer Academic Publishers, 2002.
- [9] M. D. Adams, *The JPEG-2000 still image compression standard, Technical Report*, ISO/IEC JTC1/SC29/WG1 N2412, December 2002.
- [10] Charilaos Christopoulos, Athanassions Skodras, and Touradj Ebrahimi, "The JPEG 2000 still image coding system: an overview," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp.1103-1127, November 2000.
- [11] 吳炳飛/胡益強/瞿忠正/蘇崇彥/林重甫，*JPEG 2000 影像壓縮技術*，全華科技圖書股份有限公司，台北市，2005 年 2 月。
- [12] Photoshop: <http://www.adobe.com/tw/products/photoshop/index.html>,
參照日期: 2008 年 5 月。
- [13] Photoimpact: <http://www.ulead.com.tw/pi/runme.htm>,參照時間: 2008 年
5 月。
- [14] Image eye: <http://www.fmjsoft.com/>, 參照時間: 2008 年 5 月
- [15] FastStone Image Viewer: <http://www.faststone.org/>, 參照時間: 2008 年
5 月。
- [16] Infan View: <http://www.irfanview.com/>, 參照時間: 2008 年 5 月。

- [17] R. Clark, "Taking image compression into the new millennium JPEG2000, a new standard to enrich imaging applications," ISO Bulletin, vol. 34, no. 2, pp. 17-19, Feb. 2003.
- [18] David Taubman, "High performance scalable image compression with EBCOT," IEEE Transactions on Image Processing", vol. 9, no. 7, pp. 1158-1170, July 2000.
- [19] David Taubman, Erik Ordentlich and Ikuro Ueno , "Embedded Block Coding In JPEG 2000," Signal Processing: Image Communication 17 (2002) 49-72.
- [20] 戴顯權，資料壓縮，紳藍出版社，高雄市，2002年10月，第二版二刷。
- [21] 繆紹網，數位影像處理-活用 Matlab，儒林圖書有限公司，台北市，2005年。
- [22] http://en.wikipedia.org/wiki/IEEE_754, 參照時間: 2008年5月。
- [23] The ISO and ITU will Provide Cover Pages, "JPEG 2000 Image Coding System," JPEG 2000 Committee Draft Version 1.0, 9 December 1999.
- [24] Maria Petrou and Panagiota Bosdogianni, "Image Processing The Fundamentals," John Wiley & Sons Ltd, 1999.
- [25] 連國珍，數位影像處理，儒林圖書有限公司，台北市，2004年9月。

[26] 蔡孟達，色彩影像初階校正與縮放技術研究，中原大學資訊工程學系碩士論文，2005。

[27] Jasper 1.900.1：<http://www.ece.uvic.ca/~mdadams/jasper>，參照時間：2007年11月。