

國立臺灣科技大學
電機工程系

碩士學位論文

學號:M10307302

自適應點擴散區塊截斷編碼技術
及多色調高效率藍雜訊抖動法

**Self-adaptive Dot-diffused Block Truncation Coding
and Efficient Blue-noise Dithering for Multitoning**

研究 生：鄭榆

指導 教授：郭景明 博士

中華民國 105 年 07 月 22 日



M10307302



碩士學位論文指導教授推薦書

本校 電機工程系 鄭榆(CHENG, YU) 君

所提之論文：

自適應點擴散區塊截斷編碼技術及多色調高效率藍雜訊抖動法

係由本人指導撰述，同意提付審查。

指導教授：郭景明

指導教授 郭景明

105年 8月 2日



碩士學位考試委員審定書



Barcode

M10307302

指導教授：郭景明

本校 電機工程系 鄭榆 君

所提之論文：

自適應點擴散區塊截斷編碼技術及多色調高效率藍雜訊抖動法

經本委員會審定通過，特此證明。

學校考試委員會

委員： 郭景明

鄭景明

繆紹綱

李寧角

王乃空

郭景明

指導教授：

學程主任：

系(學程)主任、所長：

陳俊良

中華民國 105 年 9 月 22 日

中文摘要

本論文主要貢獻有兩個部分：1)自適應點擴散區塊截斷編碼技術與 2)多色調高效率藍雜訊抖動法，內容簡述如下：

區塊截斷編碼是一個應用數十年的著名高效率影像壓縮技術，在本研究中提出一個新自適應點擴散區塊截斷編碼。在設計當中，輸入的原始影像依照影像各區域的複雜度自行判斷並切割成不同大小的區塊。此外，在此方法中使用一自適應參數用來控制量化色階權衡不自然紋理問題，例如區塊效應或是脈衝雜訊。另外，我們也提供一個利用人眼視覺絕對梯度誤差的評估方法，它能夠用來評估影像受到不自然紋理影響的程度。實驗結果可看出，相較於其它過往壓縮技術，本研究所提出的改良技術在四個資料庫及不同的壓縮率下都能夠產生最佳的影像品質。這表示本研究的自適應點擴散區塊截斷編碼，在高解析度影片或圖像場景等應用中都能夠具有最佳的競爭優勢。

數位多色調技術延伸自數位半色調技術，將連續色調的影像轉換成使用三色或者更多色調的量化影像。然而，過往的多色調技術大多容易產生不自然的紋理，像是帶狀效應及線性圖案，而這些問題發生的原因幾乎都是因為不均勻的點分佈，使得量化影像嚴重的降低品質。雖然迭代式多色調技術能夠產生極佳的影像，但也引入了高計算複雜度、帶狀效應及高對比問題。為了能夠改善上述情況，本論文提出多色調高效率藍雜訊抖動法。如實驗結果所呈現，本論文所提出的技術所產生的多色調影像有著傑出的色調相似度。此外，相較於過往其它的多色調技術，本論文提出的方法可達最快的處理速度，並避免了絕大多數的不自然紋理問題。綜合上述說明，本論文所提出的多色調高效率藍雜訊抖動法在實際應用上是最具優勢的候選。

關鍵字：影像壓縮、區塊截斷編碼技術、數位半色調技術、點擴散技術、數位多色調技術、直接二元搜尋法、帶狀效應、藍雜訊

Abstract

In this thesis, two contributions are presented, including self-adaptive dot-diffused block truncation coding (SDBTC) and efficient blue-noise dithering for multitoneing techniques (M-EBND).

Block truncation coding (BTC) is renowned for its high efficiency on image compression applications for decades. In this study, a new approach termed the self-adaptive dot-diffused BTC (SDBTC) is proposed. In this design, the image is adaptively divided into individual blocks of various sizes as per the image textural complexity. In addition, an adaptive parameter selection scheme is utilized to control quantization levels for leveraging artifacts such as blocking effect and impulse noise. In addition, a metric termed the human-visual absolute gradient error (HAGE) is formulated to objectively evaluate the artifacts of interest. Experimental results demonstrate that the proposed method can compress images with the lowest visually perceivable artifacts in contrast to state-of-the-arts across various compression ratios. Moreover, the proposed method is also able to yield the best image quality benchmarked with four datasets. It shows that the proposed method is competitive on the applications of high resolution video and images scenarios.

Multitoneing technique, extended from the halftoning, converts the continuous-tone image into quantized image using more than two tones. However, the former methods easily result in the unnatural textures such as banding effect, and thus seriously degrade image quality. Although iterative multitoneing methods introduce a better result, the high computational complexity is also involved. To improve visual quality and runtime, the efficient blue-noise dithering for multitoneing (M-EBND) is proposed in this paper. As documented in the experimental results, excellent tone-similarity can be achieved. Comparing to the state-of-the-art methods, the proposed method features the shortest runtime, yet avoids the most artifacts. As a result, the proposed M-EBND can be a very

competitive candidate for multitoneing applications.

Keywords: *Image compression, block truncation coding, digital halftoning, dot diffusion, multitoneing dithering, direct binary search, banding effect, blue noise.*

誌謝

學習的路程一路走來，感謝家人與親戚們一直陪伴在我的身邊並支持我，特別是我的父母，沒有他們的無私奉獻給予我不虞匱乏的生活與栽培，不會有今天的我，讓我能夠安心的學習並完成學業，再次感謝我的父母、家人與親戚們。由衷感謝指導教授郭景明博士的提攜，培養我面對問題的態度與解決事情的能力，教導我待人處事的道理，學生都將銘記在心。感謝口試委員林鼎然教授、繆紹綱教授、李宗南教授與王乃堅教授在口試時所點出的問題以及對本論文所提出的寶貴建議與指正，使本論文能夠更臻完善。特別在此感謝劉雲夫學長的指導，在研究所期間提供許多半色調相關領域的知識及指導程式碼與論文寫作的改善，協助學生得以完成這眾多的研究。

在這邊感謝實驗室所有的同仁：感謝蘇懷生學長給予許多專業學術上的寶貴意見，與學長反覆的討論及研究才能夠不斷地有新的想法產生；感謝張嘉裕學長在修改論文時的協助；感謝賴國弘學長交接研究時孜孜不倦的指導；感謝張立穎學長指導實驗室相關事務的應對方式，並協助大家解決英文寫作方面的問題；感謝吳重昇同學程式及其它問題的指導；感謝吳哲逸同學各方面的幫助；感謝薛晉宇同學擔任實驗室英文諮詢顧問；感謝吳逸庭同學帶給實驗室許多歡笑；感謝姚振傑學弟的甜點，真是個好男人；感謝林士傑學弟幫實驗室管帳；感謝李紹恩學弟熱情參與我提議的活動；感謝林育達學弟設備方面的幫助；感謝王祥毅學弟交接我的研究並成為半色調領域的達人；最後，感謝實驗室所有同仁的協助與照顧。

鄭榆 謹致

國立台灣科技大學 電機工程所 系統工程組

中華民國 105 年 07 月 14 日

目錄

中文摘要	I
Abstract	II
誌謝	IV
目錄	V
圖表索引	VIII
第一章 緒論	1
1.1 研究動機與目的	1
1.2 論文架構	6
第二章 數位半色調技術文獻探討	7
2.1 誤差擴散法(Error-diffused, ED)	10
2.2 有序抖動法(Ordered dither, OD)	20
2.3 點擴散法(Dot-diffused, DD)	24
2.4 直接二元搜尋法(Direct binary search, DBS)	36
2.5 雙指標直接二元搜尋法(Dual metric DBS, DMDBS)	41
第三章 區塊截斷編碼技術文獻探討	44
3.1 區塊截斷編碼(Block truncation coding, BTC)	48
3.2 絶對矩量區塊截斷編碼(Absolute moment BTC, AMBTC)	52
3.3 誤差擴散區塊截斷編碼>Error-diffused BTC, EDBTC)	55
3.4 有序抖動區塊截斷編碼(Ordered dither BTC, ODBTC).....	58
3.5 點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC).....	61
3.6 彩色絕對矩量區塊截斷編碼(Color absolute moment BTC, CAMBTC)	65
3.7 單一位元圖區塊截斷編碼(Single bit-map BTC, SBMBTC).....	68
3.8 基於單一位元圖的編碼方案(Coding schemes based on SBM, CS-SBM).....	75
3.9 分群區塊截斷編碼(BTC with k-means quad clustering, IBTC-KQ)	80

3.10 彩色點擴散區塊截斷編碼(Color DDBTC, CDDBTC).....	83
第四章 數位多色調技術文獻探討	86
4.1 多色調誤差擴散法(Multitoning error-diffused, M-ED)	89
4.2 多色調有序抖動法(Multitoning ordered dither, M-OD)	92
4.3 多色調直接二元搜尋法(Multitoning direct binary search, M-DBS)	96
4.4 多色調藍雜訊抖動法(Multitoning blue-noise dithering, M-BND)	99
4.5 直接多位元搜尋法(Direct multi-bit search, DMS).....	110
4.6 色調替代之誤差擴散法(Multitoning tone-replacement ED, M-TRED)	114
第五章 自適應點擴散區塊截斷編碼技術	119
5.1 彩色通道分析	120
5.2 自適應策略(Adaptive strategy)	122
5.3 自適應量化色階(Adaptive quantization level)	124
5.4 週期性圖案(Periodic pattern)	128
5.5 $\gamma_\omega(\varphi)$ 的定義	131
5.6 參數設定	133
5.7 自適應點擴散區塊截斷編碼	135
5.8 實驗結果比較	136
5.8.1 影像品質及壓縮率(Image quality and compression ratio)	138
5.8.2 不自然紋理問題(Artifact problem).....	164
5.8.3 處理速度(Runtime).....	170
5.9 小結	172
第六章 多色調高效率藍雜訊抖動法	173
6.1 墨水濃度(Ink concentration)	174
6.2 篩選法(Screening)	178
6.3 實驗結果比較	181
6.3.1 平均功率頻譜密度(Average power spectrum density)	182

6.3.2 多色調紋理(Multitone textures).....	186
6.3.3 總結比較	209
6.4 小結	211
第七章 結論與未來展望	212
第八章 投稿資料	214
參考文獻	215

圖表索引

圖 1.1 灰階影像來自資料庫 SIPI [1]	2
圖 1.2 將圖 1.1 的影像經 Otsuz 法轉換後的二值化影像	2
圖 1.3 將圖 1.1 的影像經數位半色調處理後的影像	3
圖 1.4 將圖 1.1 的影像經 JPEG 壓縮後的重建影像	4
圖 1.5 將圖 1.1 的影像經 BTC 壓縮後的重建影像	5
圖 2.1 半色調測試用之灰階影像	9
圖 2.2 誤差擴散(ED)之流程圖	10
圖 2.3 誤差擴散法實際運算流程示意圖	11
圖 2.4 誤差擴散法所產生的兩種不自然的紋理圖樣	12
圖 2.5 使用 Floyd-Steinberg 的誤差擴散權重所產生的半色調影像	13
圖 2.6 使用 Jarvis 的誤差擴散權重所產生的半色調影像	14
圖 2.7 使用 Stucki 的誤差擴散權重所產生的半色調影像	15
圖 2.8 使用 Shiau 與 Fan 的誤差擴散權重所產生的半色調影像	16
圖 2.9 四種不同的處理路徑	17
圖 2.10 有序抖動法之流程圖	21
圖 2.11 使用有序抖動矩陣 Classical-4 所產生的半色調影像	22
圖 2.12 使用有序抖動矩陣 Bayer-5 所產生的半色調影像	23
圖 2.13 類別矩陣擴散詳細示意圖	26
圖 2.14 類別矩陣之邊界擴散處理範例示意圖	26
圖 2.15 使用 Knuth 提出的類別矩陣及擴散矩陣所產生的半色調影像	30
圖 2.16 使用 Mese 與 Vaidyanathan 提出的類別矩陣及擴散矩陣所產生的半色調影像	32
圖 2.17 使用 Guo 與 Liu 提出的類別矩陣及擴散矩陣所產生的半色調影像	34
圖 2.18 以各種點擴散法所產生出的不自然週期性紋理	35

圖 2.19 翻點(Toggle)與交換(Swap)處理示意圖	37
圖 2.20 使用 7×7 高斯模糊作為點擴展函數所產生的半色調影像	40
圖 2.21 使用 Näsänen 的人眼視覺系統模型所產生的半色調影像	43
圖 3.1 壓縮測試用之灰階影像	46
圖 3.2 測試用之彩色影像	47
圖 3.3 傳統區塊截斷編碼(BTC)的流程圖	49
圖 3.4 BTC 重建之影像(8×8 區塊大小).....	50
圖 3.5 BTC 重建之影像(16×16 區塊大小)	51
圖 3.6 AMBTC 重建之影像(8×8 區塊大小)	53
圖 3.7 AMBTC 重建之影像(16×16 區塊大小)	54
圖 3.8 EDBTC 使用 Floyd-Steinberg 的誤差擴散權重建之影像(8×8 區塊大小)	56
圖 3.9 EDBTC 使用 Floyd-Steinberg 的誤差擴散權重建之影像(16×16 區塊大小)	57
圖 3.10 ODBTC 使用 Bayer-5 的有序抖動矩陣重建之影像(8×8 區塊大小)	59
圖 3.11 ODBTC 使用 Bayer-5 的有序抖動矩陣重建之影像(16×16 區塊大小)	60
圖 3.12 DDBTC 使用 Guo 及 Liu 所提出之矩陣重建之影像(8×8 區塊大小)	63
圖 3.13 DDBTC 使用 Guo 及 Liu 所提出之矩陣重建之影像(16×16 區塊大小)	64
圖 3.14 CAMBTC 重建之影像(8×8 區塊大小) , $CR = 11.64$	66
圖 3.15 CAMBTC 重建之影像(16×16 區塊大小) , $CR = 16.00$	67
圖 3.16 SBMBTC 依據亮度規則重建之影像(8×8 區塊大小) , $CR = 13.71$	69
圖 3.17 SBMBTC 依據亮度規則重建之影像(16×16 區塊大小) , $CR = 20.21$	70
圖 3.18 SBMBTC 依據權重規則重建之影像(8×8 區塊大小) , $CR = 13.71$	71
圖 3.19 SBMBTC 依據權重規則重建之影像(16×16 區塊大小) , $CR = 20.21$	72
圖 3.20 SBMBTC 依據多數決規則重建之影像(8×8 區塊大小) , $CR = 13.71$	73
圖 3.21 SBMBTC 依據多數決規則重建之影像(16×16 區塊大小) , $CR = 20.21$	74
圖 3.22 第二方案(SS)的流程圖	76
圖 3.23 FS-BMO 重建之影像(8×8 區塊大小).....	78

圖 3.24 FS-BMO 重建之影像(16×16 區塊大小).....	79
圖 3.25 IBTC-KQ 重建之影像(8×8 區塊大小)， $CR = 3.20$	81
圖 3.26 IBTC-KQ 重建之影像(16×16 區塊大小)， $CR = 3.76$	82
圖 3.27 CDDBTC 重建之影像(8×8 區塊大小)， $CR = 13.71$	84
圖 3.28 CDDBTC 重建之影像(16×16 區塊大小)， $CR = 20.21$	85
圖 4.1 用來展示特殊缺陷之測試灰階影像	88
圖 4.2 使用 Floyd-Steinberg 誤差擴散矩陣所產生的三色階多色調影像	90
圖 4.3 使用 Floyd-Steinberg 誤差擴散矩陣所產生的三色階多色調影像	91
圖 4.4 使用 Bayer-5 有序抖動矩陣所產生的三色階多色調影像	93
圖 4.5 使用藍雜訊遮罩有序抖動矩陣所產生的三色階多色調影像	94
圖 4.6 使用藍雜訊遮罩有序抖動矩陣所產生的三色階多色調影像	95
圖 4.7 使用 7×7 高斯模糊作為點擴展函數所產生的三色階多色調影像	97
圖 4.8 使用 7×7 高斯模糊作為點擴展函數所產生的三色階多色調影像	98
圖 4.9 分層式架構示意圖	100
圖 4.10 四色階之多色調影像的堆疊範例	100
圖 4.11 兩種不同的墨水濃度表	102
圖 4.12 對應圖 4.11 的墨水濃度累積曲線	103
圖 4.13 以圖 4.1(a)當作範例之第一層分層影像.....	104
圖 4.14 以圖 4.1(a)當作範例之第二層分層影像.....	105
圖 4.15 使用圖 4.12 (a)的累積墨水濃度曲線所產生的三色階多色調影像.....	107
圖 4.16 使用圖 4.12 (b)的累積墨水濃度曲線所產生的三色階多色調影像	109
圖 4.17 直接多位元搜尋法所產生的三色階多色調影像	112
圖 4.18 直接多位元搜尋法所產生的三色階多色調影像	113
圖 4.19 色調替代之誤差擴散法之流程圖	115
圖 4.20 藍雜訊遮罩之範例	115
圖 4.21 色調替代之誤差擴散法所產生的三色階多色調影像	117

圖 4.22 色調替代之誤差擴散法所產生的三色階多色調影像	118
圖 5.1 自適應點擴散區塊截斷編碼壓縮技術對各通道的流程圖	119
圖 5.2 使用單一位元圖技術(SBM)產生嚴重的區塊效應	121
圖 5.3 自適應點擴散區塊截斷編碼壓縮技術對彩色影像的流程圖	121
圖 5.4 平均標準差(σ_ω)在不同大小區塊的關聯性	123
圖 5.5 經彩色點擴散區塊截斷編碼(CDBTC)壓縮後重建的影像，其中量化色階套用公式(5.3)及公式(5.4)使用不同的 β_ω ，區塊大小 $\omega = 16$	125
圖 5.6 在不同標準差情況下平均的優化 β_ω	126
圖 5.7 DDBTC 技術慣例的鋪裝方法	128
圖 5.8 在此以 3×3 大小的矩陣做為範例表示八種不同的 CM。值得注意的是，任何一種都只是經由簡單的旋轉或對調而產生	129
圖 5.9 安排 CM 的方法中不能出現的案例。(a)相鄰 CM 有完全相同的處理順序，(b)相鄰像素有相同的處理順序(會導致 3×3 大小的擴散矩陣無法正常運作)	129
圖 5.10 在不同標準差情況下平均的人眼視覺峰值信噪比(HPSNR).....	132
圖 5.11 在不同壓縮率(CR)及縮放倍率(δ)下，使用評估方法(a) PSNR 及(b) FSIMc 的平均影像品質。其中，每條線的數值標示的是使用不同縮放倍率(δ)的結果。	134
圖 5.12 所有技術在不同資料庫中，不同壓縮率下的兩種影像評估：人眼視覺峰值信噪比(HPSNR)及針對灰階影像的特徵相似度指數(FSIM)的平均數值	143
圖 5.13 所有技術壓縮灰階影像 1 後的重建影像，其中依照表 5.3 的分類	146
圖 5.14 所有技術壓縮灰階影像 2 後的重建影像，其中依照表 5.3 的分類	149
圖 5.15 所有技術在不同資料庫中，不同壓縮率下的兩種影像評估：峰值信噪比(PSNR)及針對彩色影像的特徵相似度指數(FSIMc)的平均數值	154
圖 5.16 所有技術壓縮後的局部影像	155
圖 5.17 所有技術壓縮彩色影像 1 後的重建影像，其中依照表 5.4 的分類	159
圖 5.18 所有技術壓縮彩色影像 2 後的重建影像，其中依照表 5.4 的分類	163
圖 5.19 各個技術重建影像經一維的自相關計算結果，其中 $D = 39$	165

圖 5.20 128 × 128大小的人工影像，在不同空間頻率下各技術壓縮後的結果	166
圖 5.21 各技術在相同壓縮率下的重建影像其局部放大圖及其對應的梯度平滑影像，括號內為整張影像的 HAGE 數值	168
圖 5.22 各技術在不同壓縮率下以 HAGE 評估其損壞程度	169
圖 5.23 各技術在不同參數設定下的平均處理速度	171
圖 6.1 多色調高效率藍雜訊抖動法的流程圖	173
圖 6.2 在不同純色影像中，三色階多色調技術(a) M-DBS 、(b) DMS 及(c) M-TRED 所統計的墨水濃度比率	176
圖 6.3 512 × 96 的 Ramp 影像經三色階的多色調技術轉換後的量化影像，(a) M-DBS 、(b) DMS 、(c) M-TRED 、(d)-(e) Proposed method 使用不同模擬墨水濃度的結果	177
圖 6.4 在相同頻率下，相同平均灰階但不同對比的兩色調影像	177
圖 6.5 大小為 512 × 96 的 Ramp 影像經各個三色階的多色調技術轉換後的量化影像	187
圖 6.6 大小為 512 × 512 的自然影像經各個多色調技術量化後的結果，從左至右的量化色階數分別為三、四、五色階。	188
圖 6.7 不同的多色調技術產生的測試影像-灰階影像 3 之多色調影像	198
圖 6.8 不同的多色調技術產生的測試影像-灰階影像 4 之多色調影像	208
圖 6.9 不同多色調技術在三、四、五色階時的平均處理速度	210
表 2.1 數位半色調技術類型比較表	8
表 2.2 Floyd-Steinberg 、 Jarvis 、 Stucki 、 Shiau 與 Fan 等提出之四種不同的誤差擴散權重	12
表 2.3 Ostromoukhov [7] 提出的誤差擴散權重	18
表 2.4 Ostromoukhov [7] 提出的灰階值 g 對應不同之誤差擴散權重表	19
表 2.5 文獻[8]所提出的兩種有序抖動矩陣	21
表 2.6 Knuth [9] 提出之類別矩陣與擴散權重	27
表 2.7 Mese 與 Vaidyanathan [10] 提出之類別矩陣與擴散權重	28

表 2.8 Guo 與 Liu [11]提出之類別矩陣與擴散權重	29
表 3.1 Guo 與 Liu [29]提出之類別矩陣與擴散權重	62
表 3.2 不同SR對應的尋找範圍	76
表 3.3 三通道位元圖省略技術可能發生的所有情況	77
表 4.1 文獻[42]在 $p = 0.65$ 下提供的不同色階的最佳化誤差濾波器	116
表 5.1 β_ω 的模擬曲線參數	127
表 5.2 $t_\omega(\varphi)$ 的模擬曲線參數	132
表 5.3 各技術參數(過往技術 a~e 為區塊大小；改良技術為 φ)的設定	136
表 5.4 為了符合四種壓縮率範圍，各技術參數(過往技術 a~e 為區塊大小；改良技術為 φ)的設定，括號為技術在該參數設定下所產生的壓縮率	137
表 5.5 五種不同的客觀影像評估技術評估重建影像的結果(在此將會把最佳及次佳的結果框出；最佳的結果會再以灰底表示).....	138
表 6.1 128×128 大小的影像在特定灰階下，各個多色調技術三色階量化影像及對應的 APSD 影像，其中”n/a”表示沒有任何點輸出	184
表 6.2 128×128 大小的影像在特定灰階下，各個多色調技術三色階量化影像其分層半色調影像及對應的 APSD 影像，其中”n/a”表示沒有任何點輸出	185
表 6.3 總結各個多色調技術在不同層面比較下的結果(各類的最佳解將用圓圈標示)	210

第一章 緒論

1.1 研究動機與目的

資訊獲取的管道雖然依據時代的進步，使得交由網路的方式成為現今開發中以上水平之國家不可或缺的方式，然而由於顯示裝置的多元性，導致各種圖像資訊的顯示將會產生難以預期之誤差存在。針對此類媒體構成，其是由文字及圖像所組成，如上述提及的，現今的列印暨印刷裝置是一個有限色階輸出的硬體裝置，何謂色階有所限制呢？以生活中常見的輸出裝置印表機為例，欲表達 8-bit 灰階影像就需要有 256 種顏料對應其不同灰階強度(Grayscale intensity)進行影像輸出。在這樣的情況下以目前科技雖然足以製造容納 256 種顏料的裝置，但會造成三個主要的問題，第一個問題是會使裝置體積大增以至於無法輕易裝置在任一空間內，第二個問題則是每一個顏料都必需進行品質管理以確保輸出色階一致，第三個則將輸出色階延伸至 24-bit 彩色影像時，會使得上述兩個問題愈顯嚴重，如以此概念設計印表機將會造成無法有效製造與應用在生活中。

因此，文字及圖像的顯示結果只能透過由受色階限制的輸出硬體表現，可想而知輸出結果與原始資料之間會有不可避免的失真存在。以圖 1.1 為轉換對象，圖 1.2 使用圖 1.1 經 Otsuz 法進行二值化之結果，此範例為一個失真明顯的結果。因此，尚需考慮可藉由何種技術來有效的減少因降低表現色階的資料嚴重喪失問題，此亦為多/半色調技術發展的一個動機所在。數位多/半色調技術為印刷與顯示產業的重要技術之一，目前仍持續被廣泛地討論與研究。其中評判多/半色調技術是否為出色的方法，可以從輸出的影像品質進行評估。因此本研究將以提升視覺影像品質為目標進行數位多色調技術的研究與開發。



圖 1.1 灰階影像來自資料庫 SIPI [1]



圖 1.2 將圖 1.1 的影像經 Otsuz 法轉換後的二值化影像

多色調與半色調技術是一個利用 q 個訊號表達 $>q$ 種訊號之技術。圖 1.3 是將圖 1.1 使用兩色階(黑與白)轉換後的數位半色調影像。當仔細地觀察這兩張影像時，雖然能感受其相異之處，但亦可感受到影像之灰階明暗漸層之變化。然而，此類技術的發展有一個基礎的假設，若此基礎假設不成立，則無法達成前述的要求：利用具備低通特性之接收器進行處理過後的影像接收。例如人類視覺系統(Human visual system, HVS)則符合此假設，其可利用人眼天生具有的低通特性將有限的輸出顏色擴展至近於其處理前之顏色，因此能使得數位多/半色調影像能夠使人們觀看時產生多色階或連續色階的感受，如圖 1.3 之影像所呈現之結果。

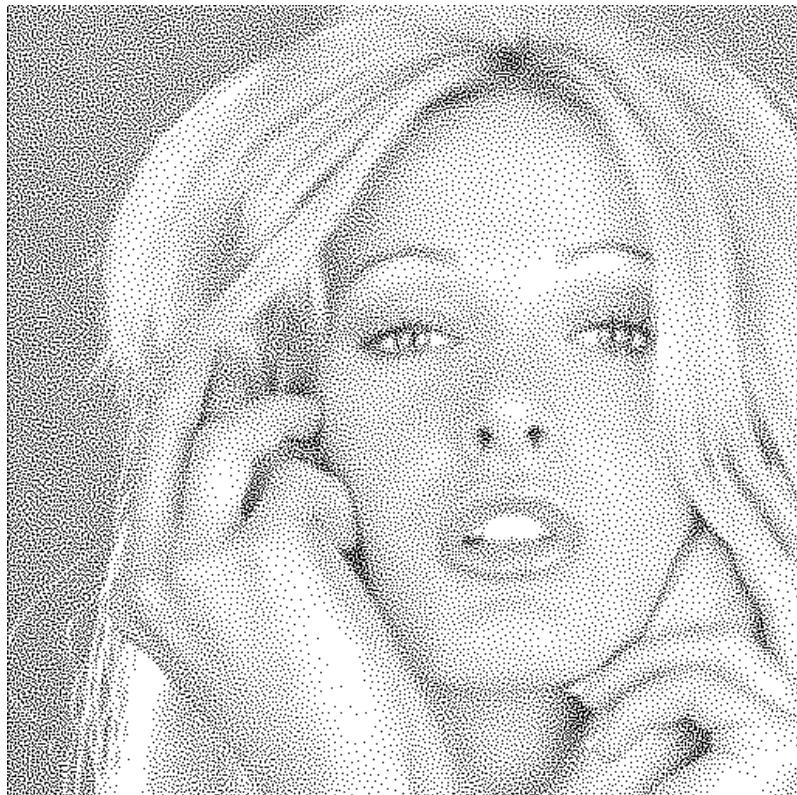


圖 1.3 將圖 1.1 的影像經數位半色調處理後的影像

基於數位半色調技術的蓬勃發展，日常生活常見的資訊傳播媒介如報紙、書籍、海報或傳單等皆具有此技術的影子，甚至亦延伸出了利用該影像特性進行壓縮等之應用。在這個科技發達的世代中，相機的畫素越高所拍的照片越細膩，但照片資料大小

也越變越大，想上傳到網路分享會因為檔案太大所以難以上傳，就算成功放在網頁供人瀏覽也會因影像太大使得使用者網頁開得慢或是跑不出圖片，或是想儲存在自己的電腦中留做紀念也會占據大量記憶體儲存的空間。而影像畫素較高的攝影機可拍攝出清晰之影像，提升對物體的辨識率，有助於犯罪偵防時進行人臉辨識及車牌辨識，但因檔案太大而不利傳輸。對於此，雖然技術的日益精進，目前受到最為廣泛使用的影像壓縮技術仍為 JPEG (Joint photographic experts group)，壓縮結果如圖 1.4。雖然可得到不錯的壓縮品質，但由於其計算過程中考量到非常繁複如：色彩空間轉換、縮減取樣(Downsampling)、離散餘弦變換(Discrete cosine transform)、量化(Quantization)、以及熵編碼技術(Entropy coding)等，造成 JPEG 計算複雜度居高不下，因此於一些特殊應用如具有高速影像壓縮需求的監控系統等等，難以採用此種類型之技術進行處理。



圖 1.4 將圖 1.1 的影像經 JPEG 壓縮後的重建影像

無論採取何種壓縮技術，對最終用戶而言他最為關心的無非幾大要素：清晰

度、存儲量、穩定性還有價格。採用不同的壓縮技術，將很大程度影響以上幾大要素。在未來趨勢上，相機畫素越高，想要拍比較多照片就得要有較大的記憶卡儲存，而好的影像壓縮能有效減少相片的檔案大小也能維持其影像品質。本研究所要探討的區塊截斷編碼(Block truncation coding, BTC)影像壓縮技術相較於 JPEG 等先進影像壓縮技術而言，其具備極低計算複雜度之特性並也能夠有不錯的重建影像，如圖 1.5。如此，當用於具有計算複雜度限制或者具有即時處理需求之應用，例如嵌入式系統或智慧型監控系統，本研究所針對的影像壓縮技術更具備明顯之優勢。



圖 1.5 將圖 1.1 的影像經 BTC 壓縮後的重建影像

1.2 論文架構

本篇論文共分為七個章節，論文架構簡述如下：

第一章：

簡述研究動機與目的。

第二章：

對於目前現有的幾種數位半色調技術進行介紹，並且分析不同技術之優缺點。後續的第二章及第三章是以半色調為基礎做延伸亦或是有相關聯的技術。

第三章：

將介紹幾種著名的區塊截斷編碼技術，部分介紹到的技術將在後續章節用來與本研究提出的改良技術做比較。

第四章：

對於目前各種轉換方式的數位多色調技術進行介紹，部分介紹到的技術將在後續章節用來與本研究提出的改良技術做比較。

第五章：

介紹本論文所提出的自適應點擴散區塊截斷編碼技術，其主要是改善過往區塊截斷編碼對區域只能使用固定大小區塊的問題，並以點擴散區塊截斷編碼為基礎改善影像品質及不自然紋理問題(Artifact problem)。

第六章：

介紹本論文所提出的多色調高效率藍雜訊抖動法，其主要目的是降低多色調藍雜訊抖動法的高計算複雜度，並維持其原本在各個輸出色階完美的藍雜訊點分佈。

第七章：

對本論文提出之技術進行結論，並探討未來可能的延伸研究方向。

第二章 數位半色調技術文獻探討

數位半色調技術(Digital halftone technique)被提出的主要意義在於它主要是二位元(Binary)影像，也就是整張影像只由黑點(Black)及白點(White)所組成。由於人眼視覺系統(Human visual system, HVS)的低通特性，視覺在一定的距離情況下，人眼會覺得半色調影像類似連續色調影像(Continuous tone)。因此，數位半色調技術被廣泛地應用在印刷和電子紙(E-papers)產業，例如：報紙、公車上的廣告…等。有鑑於印刷對於社會的重要性，數位半色調技術不斷地日新月異，由於需求上的不同，因此在技術上已發展出各自不同特色的數位半色調技術。

若將印刷結果放大看，就會發現它是由無數個網點所組成，藉由網點的大小、疏密來表示顏色的深淺及明暗層次，依照此性質可分類成藍雜訊(Blue-noise)及綠雜訊(Green-noise)。藍雜訊使用相同大小的網點並藉由不同的打點頻率來控制印刷的結果，反之綠雜訊則是使用相同的打點頻率並藉由不同大小的網點來控制印刷結果。舉例來說，噴墨印表機(Inkjet printer)就是利用藍雜訊半色調的優勢印出更好的錯覺結果，由於靜電印刷過程(Electrophotography printing process)引發不穩定的打印點，因此雷射印表機(Laser printer)是倚靠綠雜訊半色調去印刷。

另外一種分類的觀點是考量其處理類型，這大致上可以分為以下三類：基於單點像素點處理(Single-pixel-based process)、基於周圍像素點處理(Neighborhood-pixels-based process)、基於迭代式處理(Iteration-based process)。基於單點像素點處理是將灰階影像轉換至半色調影像時，僅考量單一像素點，也就是處理過程中不同的像素點之間並不會互相影響，因此可以有平行處理的特性。此類型有著極高的處理效率，但相對的其轉換影像的品質也較低。至於基於周圍像素點處理，在處理一個像素點時會考量其周圍的其它像素點，此方式改善了基於單點像素點處理的問題，可以產生較高影像品質的半色調結果，但是處理效率上比基於單點像素點處理還差。最後基於迭代式處理則是不斷地改變半色調影像內的點分佈，來提升影像品質，此迭代的過程會依照演算法本身的成本函數(Cost function)以及輸入影像的大小來決定花費時間的快慢，快

則數秒慢則數十分鐘，為這三類之中最耗時間，但卻擁有最自然的轉換結果。總結以上敘述，這三種分類產生的半色調結果其影像品質及處理速率的比較關係如表 2.1 所示。

以下的數位半色調技術依照上述的處理類型能分類為：

- 1) 基於單點像素點處理(Single-pixel-based process)
 - 有序抖動法(Ordered dither, OD)
- 2) 基於鄰近像素點處理(Neighborhood-pixels-based process)
 - 誤差擴散法(Error-diffused, ED)
 - 點擴散法(Dot-diffused, DD)
- 3) 基於迭代式處理(Iteration-based process)
 - 直接二元搜尋法(Direct binary search, DBS)
 - 雙指標直接二元搜尋法(Dual metric DBS, DMDBS)

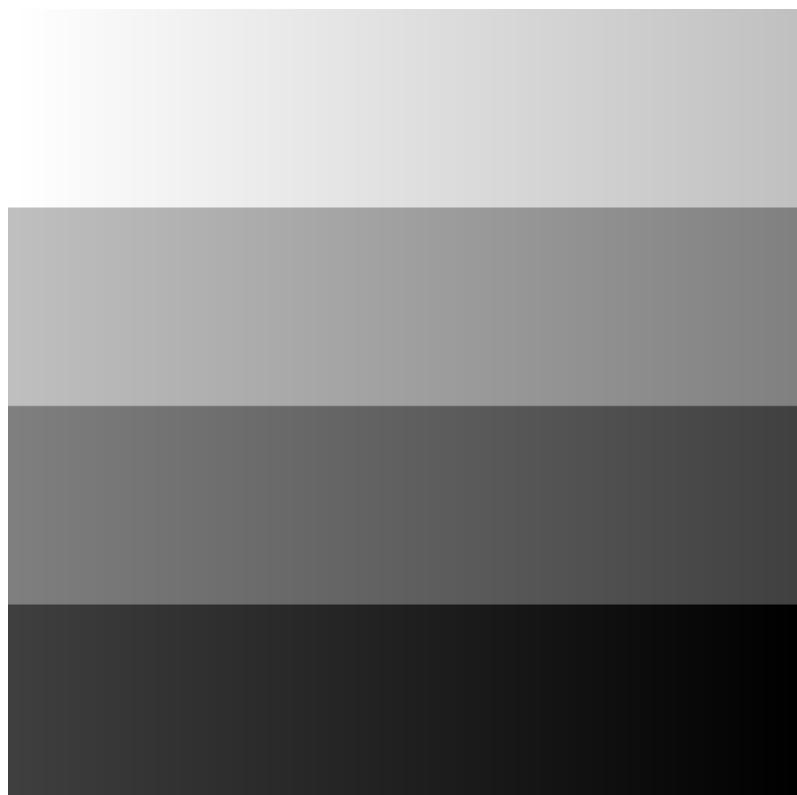
這些技術將在後續章節依照提出的時間做介紹。此外，上述方法並非所有數位半色調技術，這邊所介紹的技術將會在之後的延伸技術(區塊截斷編碼技術、多色調技術)中應用。本論文藉由分析這些不同的半色調技術，以利後續介紹時能夠理解使用這些半色調技術將會帶來的優/劣勢。在後續章節的介紹過程中會以圖 2.1 中兩張 512×512 大小的灰階影像作為實驗的展示結果。

表 2.1 數位半色調技術類型比較表

影像品質： 基於單點像素點處理 < 基於鄰近像素點處理 < 基於迭代式處理
處理效率： 基於單點像素點處理 > 基於鄰近像素點處理 > 基於迭代式處理



(a) 灰階影像 1



(b) 灰階影像 2

圖 2.1 半色調測試用之灰階影像

2.1 誤差擴散法(Error-diffused, ED)

在 1975 年，Floyd 與 Steinberg 提出誤差擴散法[2]，誤差擴散法藉由補償的方式降低取樣誤差的發生，將單一像素點產生的取樣誤差透過周圍的像素點進行補償擴散以降低整個區域的總合取樣誤差，使得此技術有較高的影像品質。誤差擴散的流程如圖 2.2 所示，像素經過誤差擴散補償後透過臨界值(Thresholding)判斷半色調輸出為黑或白，通常臨界值都設為 128，演算法的公式如下所示：

$$c(i,j) = x(i,j) + s(i,j), \text{ where } s(i,j) = \sum_{(p,q) \in R} \sum \frac{e(i+p,j+q) \times w(p,q)}{\text{sum}}, \quad (2.1)$$

$$e(i,j) = c(i,j) - y(i,j), \quad (2.2)$$

$$y(i,j) = \begin{cases} 255, & \text{if } c(i,j) \geq 128 \\ 0, & \text{if } c(i,j) < 128 \end{cases} \quad (2.3)$$

$x(i,j)$ 為原始影像在座標 (i,j) 的灰階值； $c(i,j)$ 為經過誤差擴散補償過後的灰階值； $e(i,j)$ 為輸出與當下補償後灰階的差值； $w(p,q)$ 表示在誤差擴散矩陣的範圍 R 內的誤差擴散權重，擴散權重矩陣的種類如表 2.2 所示。以表 2.2(a)為例，則其長寬的範圍分別為 $p \in [0,1]$ 及 $q \in [-1,1]$ ； $y(i,j)$ 為輸出的半色調影像。在此特別說明，在處理座標位置 $(0,0)$ 時，並沒有受到任何誤差擴散的補償，因此此時的 $c(0,0)$ 等於 $x(0,0)$ 。值得注意的是，由於誤差擴散時可能會超出影像的邊界，因此誤差的權重需要視能擴散的範圍正規化，

$$\text{sum} = \sum_{(p,q) \in R} \begin{cases} w(p,q), & \text{if } (i+p, j+q) \text{ within range} \\ 0, & \text{if } (i+p, j+q) \text{ out of range} \end{cases} \quad (2.4)$$

像素點由左上至右下處理(此移動方式被稱為 Raster scan)並重複以上步驟直到整張影像做完，就可以產生誤差擴散法的半色調影像。

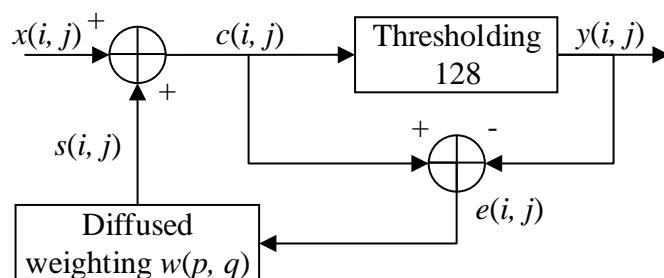


圖 2.2 誤差擴散(ED)之流程圖

圖 2.3 所呈現的是誤差擴散法實際的運算過程，當中使用的是 Floyd-Steinberg 所提出的誤差擴散權重矩陣。假設黑框為目前進行處理的像素點位置並且其影像位置為 $(0,0)$ ，而該位置的灰階值 $x(0,0)$ 為 65。透過公式(2.3)判斷後，因數值小於 128，故輸出 $y(0,0)$ 為 0。接下來計算能夠進行誤差擴散的位置其權重總和如公式(2.4)，由於左下角的權重超出邊界，因此 sum 計算後的數值為 13。藉由公式(2.2)計算可得知誤差 $e(0,0)$ 為 65，並透過公式(2.1)將誤差擴散至周圍。在圖中分別有三個不同的擴散方向，將該位置之誤差值 65 分別乘上三個不同方向的擴散權重，其結果分別為 35、25 與 5，即為各位置需被擴散的誤差值。最後，將上述三個不同方向的誤差值，依照擴散權重所擴散的位置，加回到原始影像數值當中，加完後的結果在此稱為擴散影像 (Diffused image)。依照上述的步驟，由影像像素處理位置從左上至右下方式進行反覆更新擴散影像即可得到完整的半色調影像。

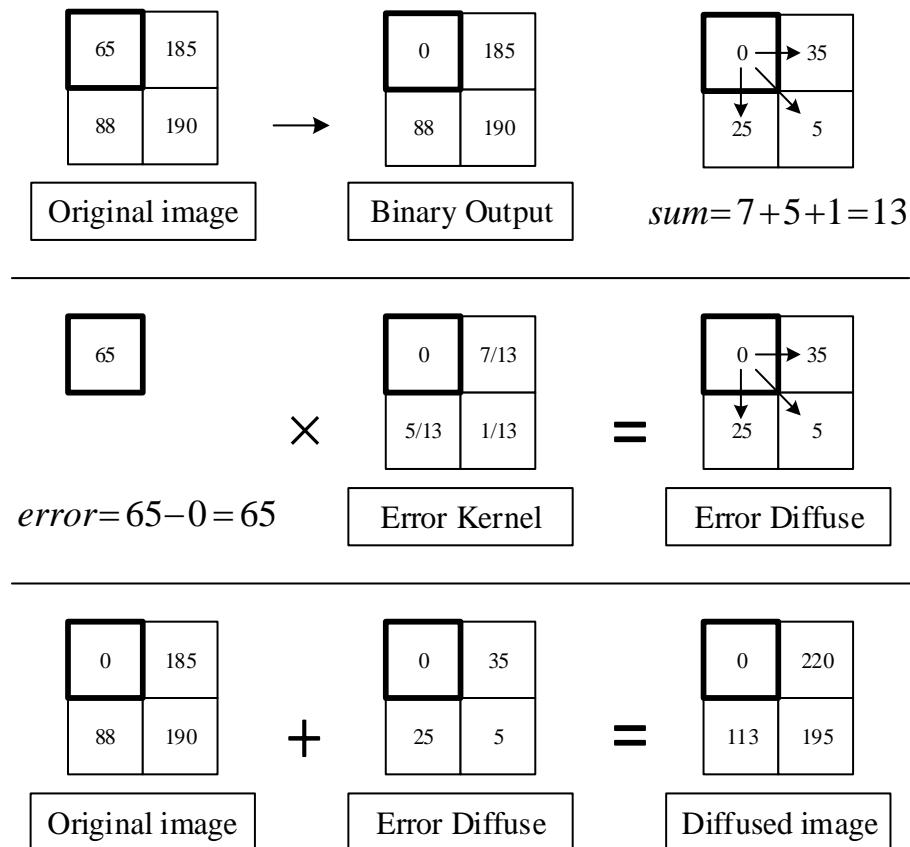


圖 2.3 誤差擴散法實際運算流程示意圖

儘管使用 Floyd 與 Steinberg 的誤差擴散權重，透過誤差擴散的方式能有效的降低原始影像與半色調影像之間的取樣誤差，使數位半色調影像品質提升，但是仍然存在著一些不自然的紋理圖樣，例如：蠕蟲效應(Worm effect)與線性圖樣(Line pattern)等不自然影像，如圖 2.4 所示。該問題大致有三種方法能夠解決。第一，調整權重及擴散範圍。由 Jarvis 與 Steinberg [3]、Stucki [4]、Shiau 與 Fan [5]分別提出了不同數值及範圍的誤差擴散權重，以改善誤差擴散法的影像品質以及一些不自然的紋路，其權重數值如表 2.2 所示，而 Floyd 與 Steinberg 及上述三種誤差擴散權重分別的測試結果如圖 2.5、圖 2.6、圖 2.7 與圖 2.8 所示。



圖 2.4 誤差擴散法所產生的兩種不自然的紋理圖樣

表 2.2 Floyd-Steinberg、Jarvis、Stucki、Shiau 與 Fan 等提出之四種不同的誤差擴散
權重

	(i,j)	7
3	5	1

(a) Floyd-Steinberg [2]

		(i,j)	7	5
3	5	7	5	3
1	3	5	3	1

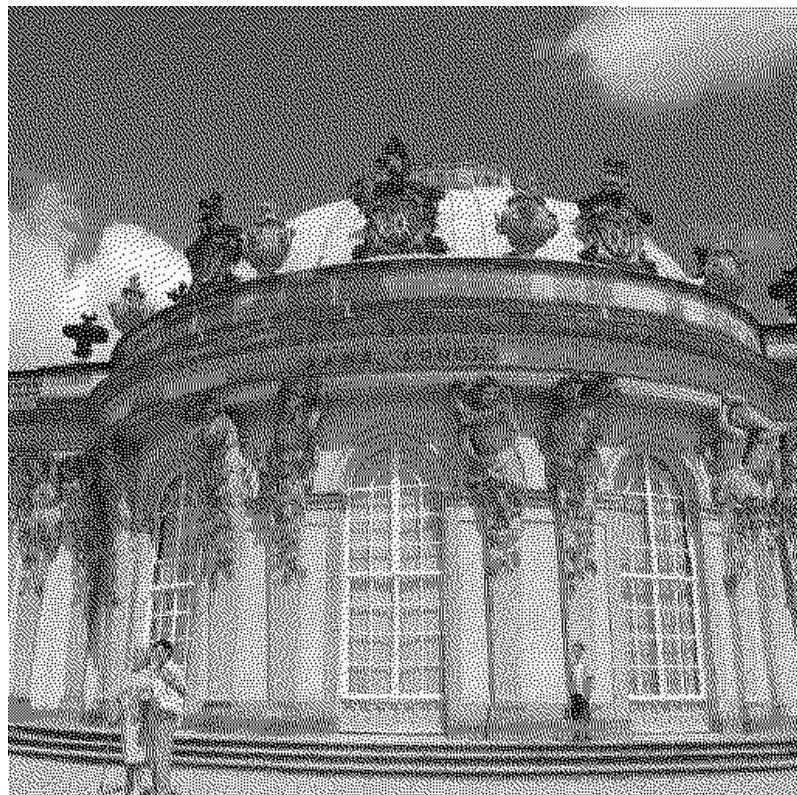
(b) Jarvis [3]

		(i,j)	8	4
2	4	8	4	2
1	2	4	2	1

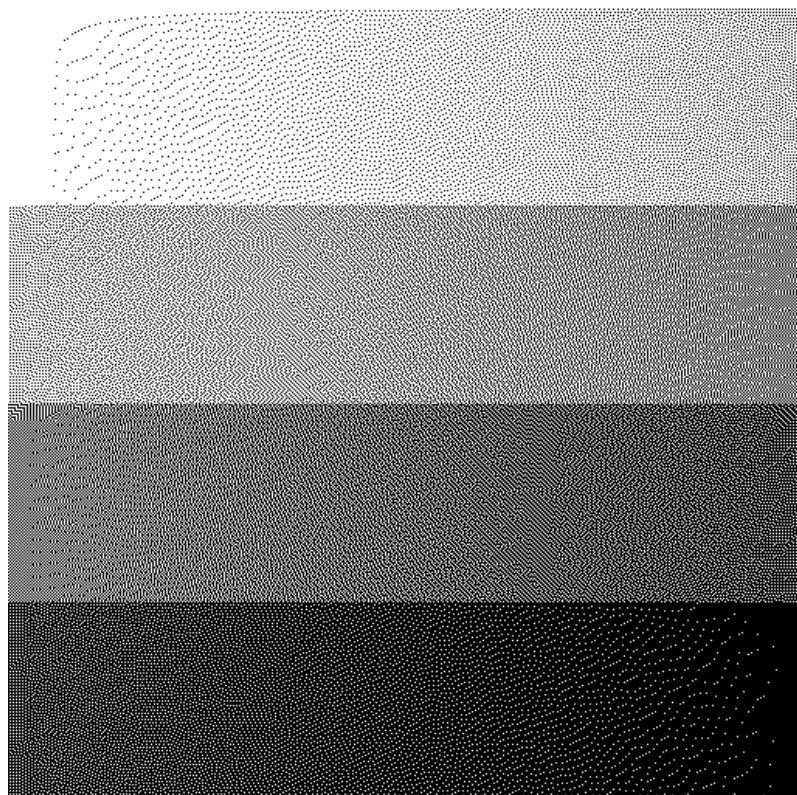
(c) Stucki [4]

			(i,j)	8
1	1	2	4	

(d) Shiau 與 Fan [5]



(a) 灰階影像 1 經誤差擴散法處理後的半色調影像

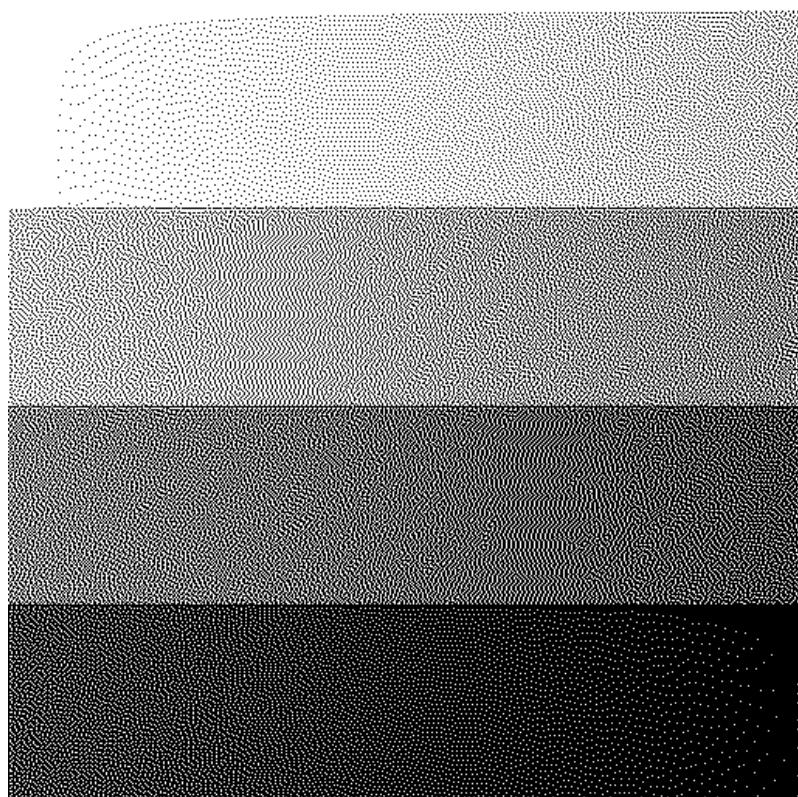


(b) 灰階影像 2 經誤差擴散法處理後的半色調影像

圖 2.5 使用 Floyd-Steinberg 的誤差擴散權重所產生的半色調影像



(a) 灰階影像 1 經誤差擴散法處理後的半色調影像

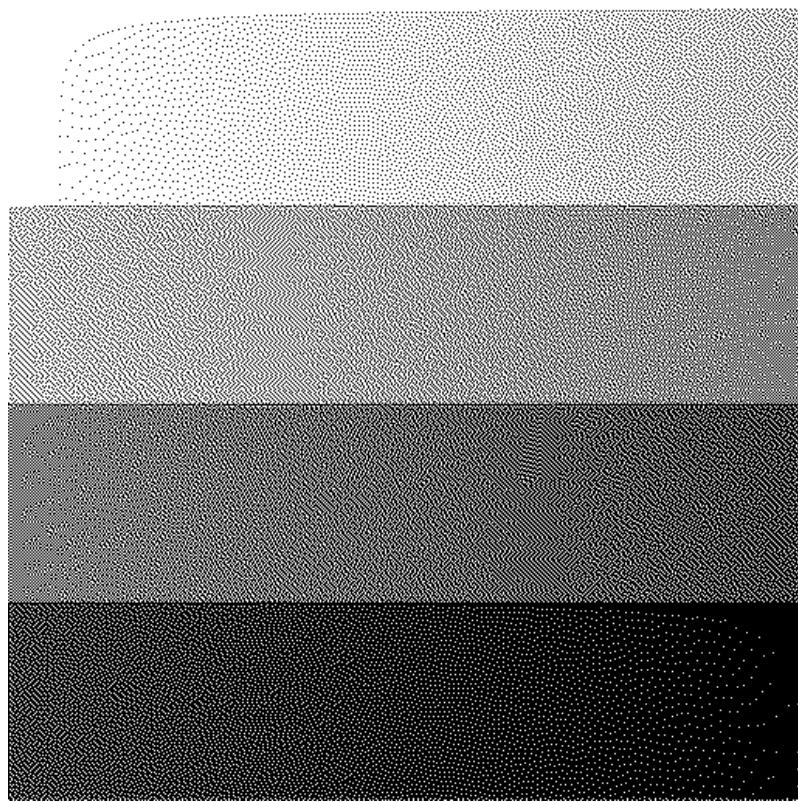


(b) 灰階影像 2 經誤差擴散法處理後的半色調影像

圖 2.6 使用 Jarvis 的誤差擴散權重所產生的半色調影像



(a) 灰階影像 1 經誤差擴散法處理後的半色調影像

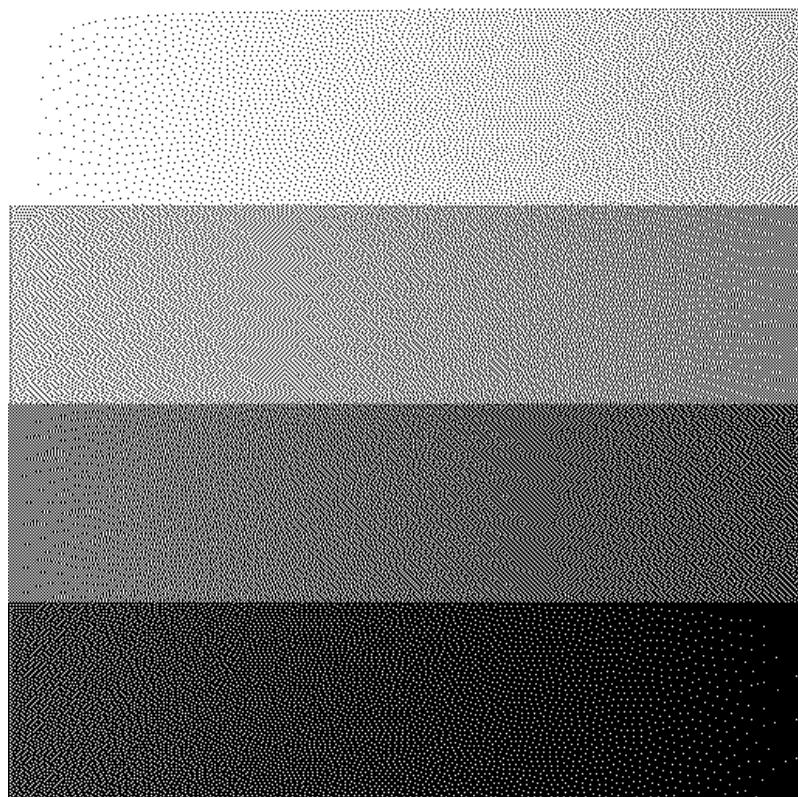


(b) 灰階影像 2 經誤差擴散法處理後的半色調影像

圖 2.7 使用 Stucki 的誤差擴散權重所產生的半色調影像



(a) 灰階影像 1 經誤差擴散法處理後的半色調影像



(b) 灰階影像 2 經誤差擴散法處理後的半色調影像

圖 2.8 使用 Shiau 與 Fan 的誤差擴散權重所產生的半色調影像

第二，修改像素的處理順序。儘管修改誤差擴散權重可以改善這些不自然的紋理圖樣，但仍無法全面性的移除這些不自然的影像。由 Li 與 Allebach 研究[6]中提出藉由修改前述之 Raster scan 處理路徑，以改善先前所述的不自然紋理圖樣，如圖 2.9 所表示。

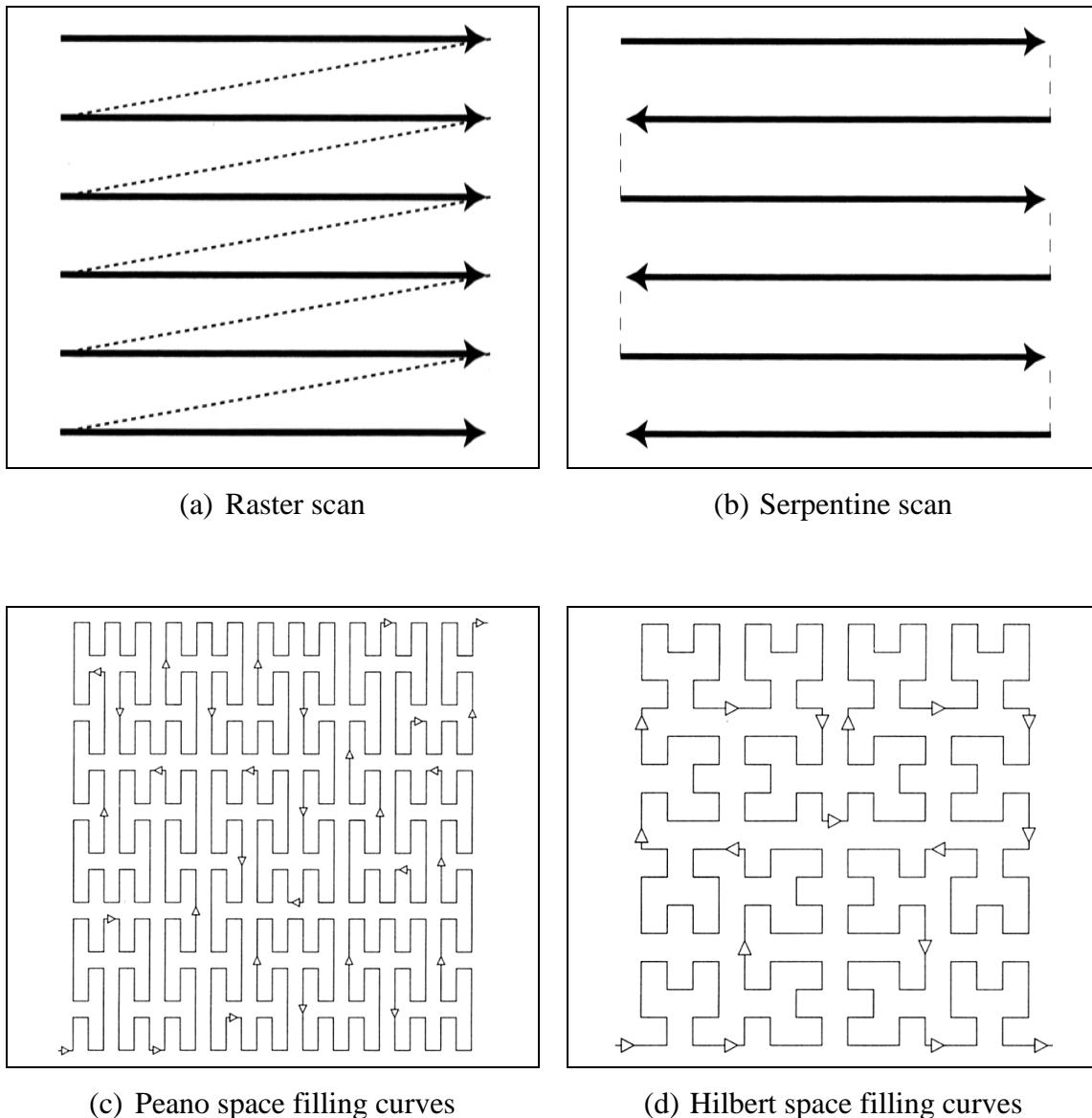


圖 2.9 四種不同的處理路徑

第三，增加更多不同的誤差擴散權重數值。前面所述之誤差擴散法都是基於單一組擴散權重進行數位半色調轉換，由 Ostromoukhov [7]所提出的多組誤差擴散權重之誤差擴散法，此方法的概念是將每一個灰階值 g 分別產生出對應的誤差擴散權重 w_g ，總共會有 256 組誤差擴散權重，而每一個誤差擴散權重則是由最佳化演算法中的經典最小值問題(Classical minimization problem)去分別訓練出不同灰階值所適應的誤差擴散權重，其每一組對應的誤差擴散權重如表 2.3 所示。由此表可得知，Ostromoukhov 假設誤差擴散權重為三個擴散方向，分別為 $w_{g,1}$ 、 $w_{g,2}$ 與 $w_{g,3}$ ，並針對這三個擴散方向找尋最佳解，便可提升影像品質與改善不自然紋理圖樣。此技術選擇以 0、1、2、3、4、10、22、32、64、72、77、85、95、107、127 等灰階值當作訓練樣本，訓練樣本以外的灰階值所對應的擴散權重數值以線性內插法(Linear interpolation)所產生。值得注意的是當灰階值大於 127 後，所有誤差擴散權重都是以鏡射的方式挑選，譬如灰階值 128 的擴散權重等於灰階值 127 的擴散權重，而 129 的擴散權重就等於灰階值 126 的權重，並以此類推。其公式如下：

$$w_g = w_{255-g}, \text{ where } g \in [0,127], \quad (2.5)$$

因此灰階值對應的誤差擴散權重如表 2.4 所示。

表 2.3 Ostromoukhov [7]提出的誤差擴散權重

	(i,j)	$w_{g,1}$
$w_{g,2}$	$w_{g,3}$	

表 2.4 Ostromoukhov [7]提出的灰階值 g 對應不同之誤差擴散權重表

g	$w_{g,1}$	$w_{g,2}$	$w_{g,3}$												
0	13	0	5	32	20	10	19	64	11	10	0	96	5	3	2
1	13	0	5	33	1937	1000	1767	65	158	151	3	97	5	3	2
2	21	0	10	34	977	520	855	66	178	179	7	98	5	3	2
3	7	0	4	35	657	360	551	67	1030	1091	63	99	5	3	2
4	8	0	5	36	71	40	57	68	248	277	21	100	5	3	2
5	47	3	28	37	2005	1160	1539	69	318	375	35	101	5	3	2
6	23	3	13	38	337	200	247	70	458	571	63	102	5	3	2
7	15	3	8	39	2039	1240	1425	71	878	1159	147	103	5	3	2
8	22	6	11	40	257	160	171	72	5	7	1	104	5	3	2
9	43	15	20	41	691	440	437	73	172	181	37	105	5	3	2
10	7	3	3	42	1045	680	627	74	97	76	22	106	5	3	2
11	501	224	211	43	301	200	171	75	72	41	17	107	5	3	2
12	249	116	103	44	177	120	95	76	119	47	29	108	305	176	119
13	165	80	67	45	2141	1480	1083	77	4	1	1	109	155	86	59
14	123	62	49	46	1079	760	513	78	4	1	1	110	105	56	39
15	489	256	191	47	725	520	323	79	4	1	1	111	80	41	29
16	81	44	31	48	137	100	57	80	4	1	1	112	65	32	23
17	483	272	181	49	2209	1640	855	81	4	1	1	113	55	26	19
18	60	35	22	50	53	40	19	82	4	1	1	114	335	152	113
19	53	32	19	51	2243	1720	741	83	4	1	1	115	85	37	28
20	237	148	83	52	565	440	171	84	4	1	1	116	115	48	37
21	471	304	161	53	759	600	209	85	4	1	1	117	35	14	11
22	3	2	1	54	1147	920	285	86	65	18	17	118	355	136	109
23	481	314	185	55	2311	1880	513	87	95	29	26	119	30	11	9
24	354	226	155	56	97	80	19	88	185	62	53	120	365	128	107
25	1389	866	685	57	335	280	57	89	30	11	9	121	185	62	53
26	227	138	125	58	1181	1000	171	90	35	14	11	122	25	8	7
27	267	158	163	59	793	680	95	91	85	37	28	123	95	29	26
28	327	188	220	60	599	520	57	92	55	26	19	124	385	112	103
29	61	34	45	61	2413	2120	171	93	80	41	29	125	65	18	17
30	627	338	505	62	405	360	19	94	155	86	59	126	395	104	101
31	1227	638	1075	63	2447	2200	57	95	5	3	2	127	4	1	1

2.2 有序抖動法(Ordered dither, OD)

此數位半色調技術收錄於 1987 年出版的數位半色調(Digital halftoning)書籍[8]中，最主要的貢獻在於高效率的平行處理能力。此技術藉由一個 $M \times N$ 大小的有序抖動矩陣(Dithering matrix)作為像素點轉換的門檻值，將原始灰階影像轉換成數位半色調影像。在此研究中將以 8×8 大小之抖動矩陣為例介紹，此技術之演算法主要流程如下：

1. 將原始影像以 $M \times N$ 大小分割出數個同樣大小的不重疊區塊。
2. 在每一個不重疊區塊中，以點對點的方式，將原始灰階影像之像素灰階值比對有序抖動矩陣中的數值，若小於則輸出為 0(黑)，反之則輸出為 255(白)，公式如下：

$$y(i, j) = \begin{cases} 255, & \text{if } x(i, j) \geq d(m, n) \times 255 \\ 0, & \text{if } x(i, j) < d(m, n) \times 255 \end{cases}, \text{ where } m = i\%M, n = j\%N, \quad (2.6)$$

$x(i, j)$ 表示在原始灰階影像位置 (i, j) 的灰階值； $d(m, n)$ 表示在有序抖動矩陣映射(Mapping)到灰階影像位置 (i, j) 的數值，由於原本的有序抖動矩陣數值範圍為 0 到 1，因此將數值乘上 255 使其正規化到跟灰階的範圍相同； $y(i, j)$ 則表示藉由有序抖動矩陣所判斷後的半色調影像數值。

3. 重複步驟 2，直到整張影像的每一個不重疊區塊做完為止，即可得到一張完整的半色調影像。

以上演算法之概念可以繪製成相對應之流程圖如圖 2.10 所示。由於每一個像素點都是在各自獨立的區塊內經有序抖動矩陣比較，因此區塊與區塊之間不會互相影響，故可以達到平行處理之效果，像素點之間也可互相獨立處理。

在文獻[8]當中提供兩種不同的有序抖動矩陣：Classical-4 與 Bayer-5，這兩種有序抖動矩陣的係數如表 2.5，其中 Classical-4 矩陣中數值的分佈會使黑白點有聚集的效果，因此稱之為聚集抖動點(Clustered-dot dithering)，為本章節前言所述之綠雜訊(Green-noise)。而 Bayer-5 矩陣中數值的分佈則會使黑白點之間的分佈會被打散，稱之為分散抖動點(Dispersed-dot dithering)，又稱藍雜訊(Blue-noise)。兩種有序抖動矩陣的影像測試結果如圖 2.11、圖 2.12 所示。

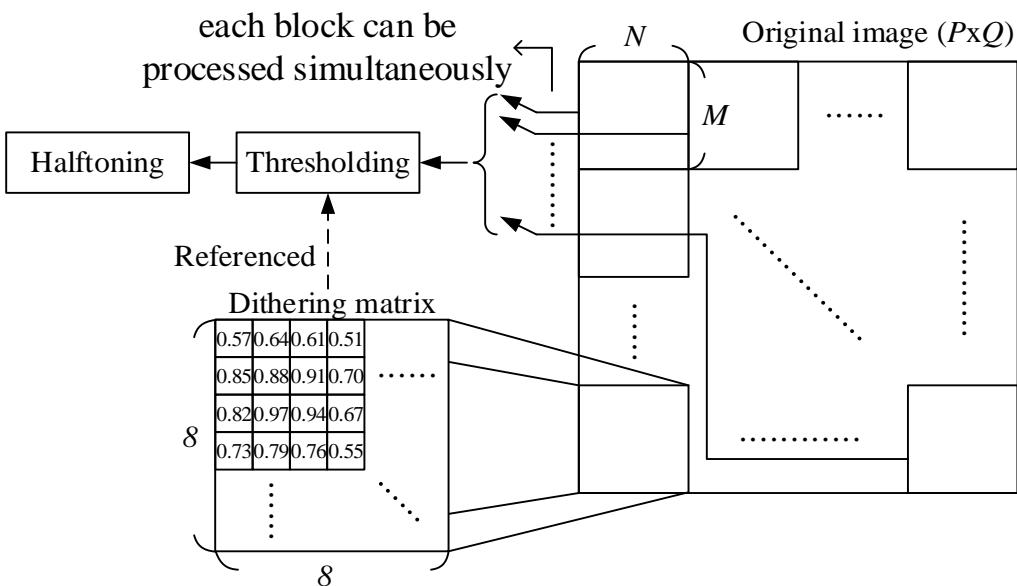


圖 2.10 有序抖動法之流程圖

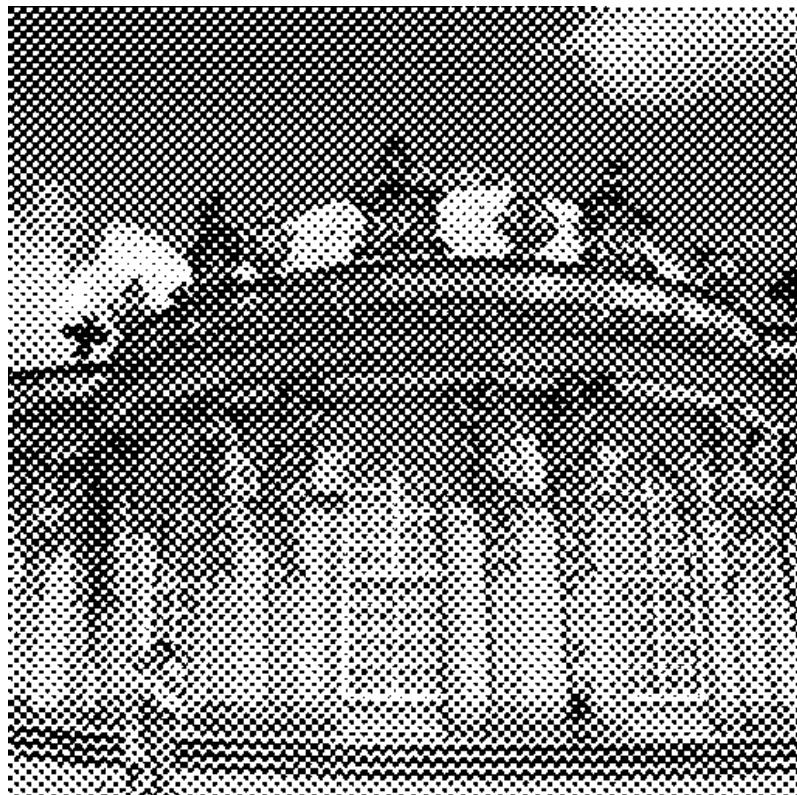
表 2.5 文獻[8]所提出的兩種有序抖動矩陣

0.567	0.635	0.608	0.514	0.424	0.365	0.392	0.486
0.847	0.878	0.910	0.698	0.153	0.122	0.090	0.302
0.820	0.969	0.941	0.667	0.180	0.031	0.059	0.333
0.725	0.788	0.757	0.545	0.275	0.212	0.243	0.455
0.424	0.365	0.392	0.486	0.567	0.635	0.608	0.514
0.153	0.122	0.090	0.302	0.847	0.878	0.910	0.698
0.180	0.031	0.059	0.333	0.820	0.969	0.941	0.667
0.275	0.212	0.243	0.455	0.725	0.788	0.757	0.545

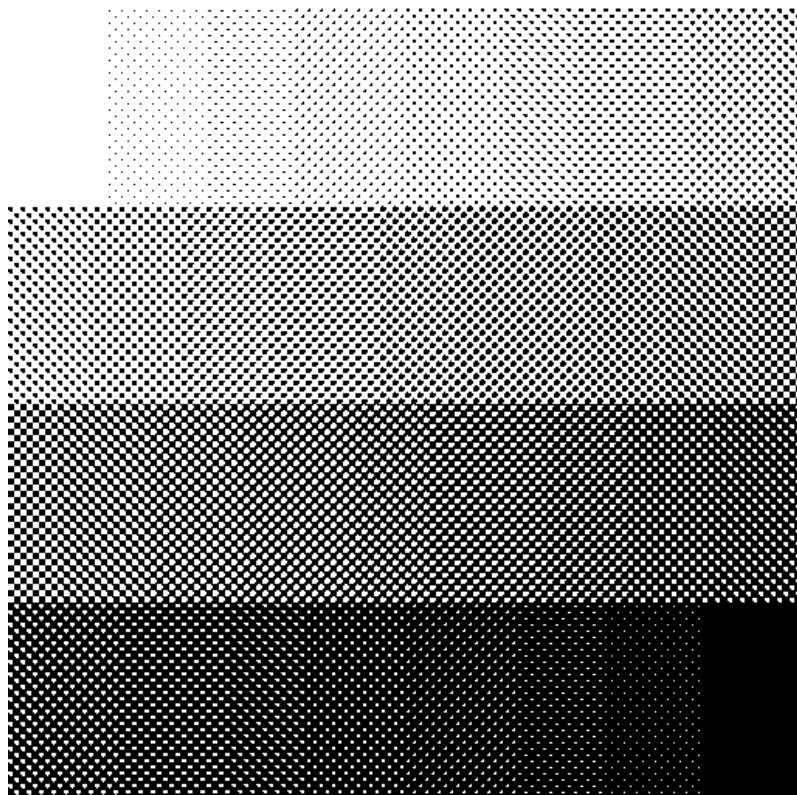
(a) Classical-4

0.513	0.272	0.724	0.483	0.543	0.302	0.694	0.453
0.151	0.755	0.091	0.966	0.181	0.758	0.121	0.936
0.634	0.392	0.574	0.332	0.664	0.423	0.604	0.362
0.060	0.875	0.211	0.815	0.030	0.906	0.241	0.845
0.543	0.302	0.694	0.453	0.513	0.272	0.724	0.483
0.181	0.758	0.121	0.936	0.151	0.755	0.091	0.966
0.664	0.423	0.604	0.362	0.634	0.392	0.574	0.332
0.030	0.906	0.241	0.845	0.060	0.875	0.211	0.815

(b) Bayer-5

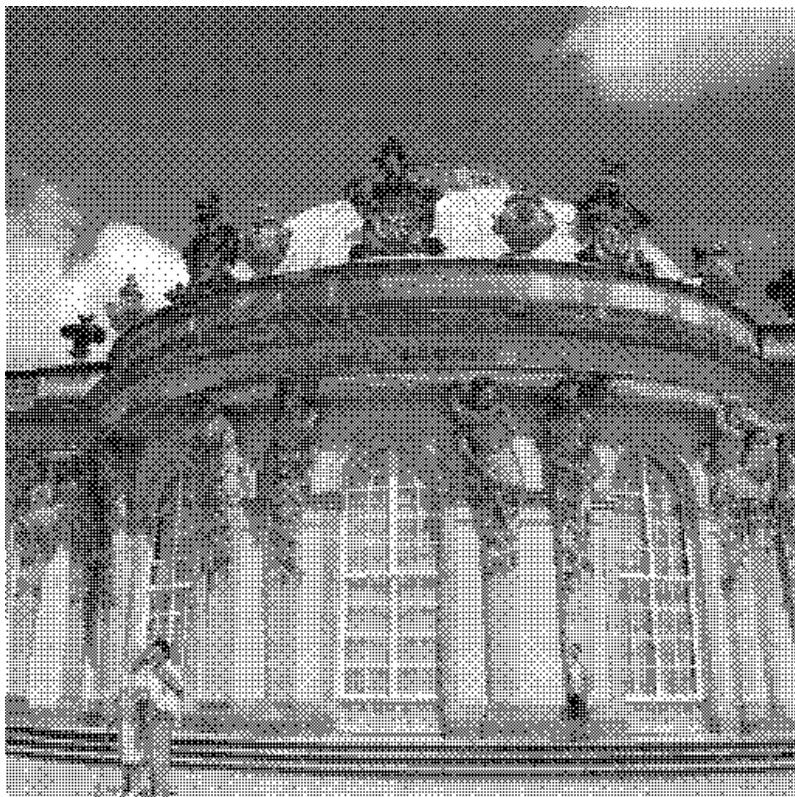


(a) 灰階影像 1 經有序抖動法處理後的半色調影像

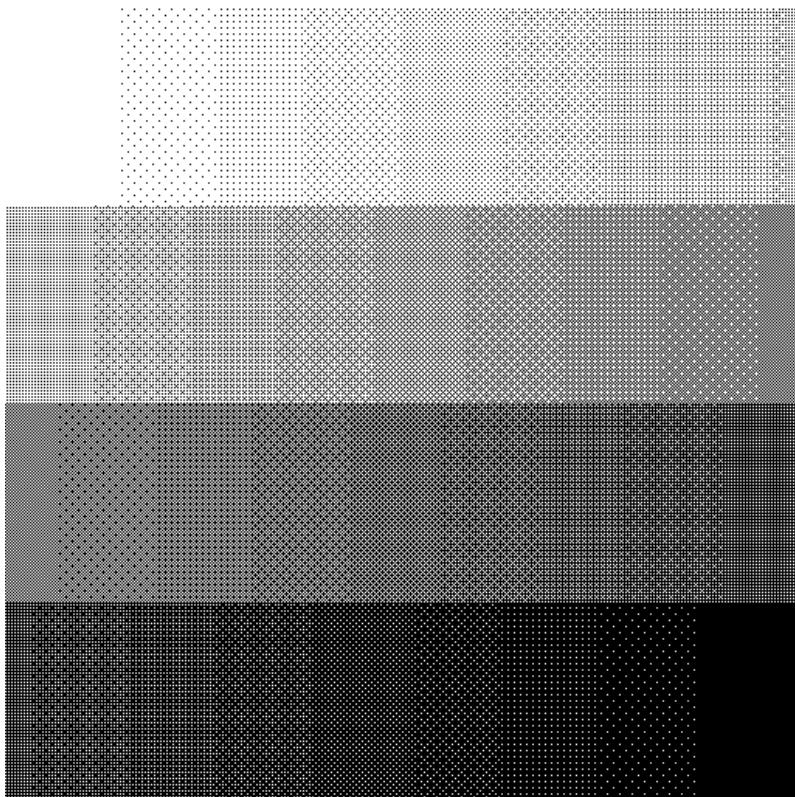


(b) 灰階影像 2 經有序抖動法處理後的半色調影像

圖 2.11 使用有序抖動矩陣 Classical-4 所產生的半色調影像



(a) 灰階影像 1 經有序抖動法處理後的半色調影像



(b) 灰階影像 2 經有序抖動法處理後的半色調影像

圖 2.12 使用有序抖動矩陣 Bayer-5 所產生的半色調影像

2.3 點擴散法(Dot-diffused, DD)

點擴散法為 Knuth [9]在 1987 年所提出的技術，其主要的特色在於結合了誤差擴散法的擴散取樣誤差與有序抖動法的平行處理能力，因此同時具備誤差擴散矩陣與有序抖動矩陣的特性，在此方法中稱為擴散矩陣(Diffused matrix, DM)與類別矩陣(Class matrix, CM)。點擴散法主要流程如下，首先，將原始灰階影像根據類別矩陣大小分割成數個不重疊區塊，此概念類似於有序抖動法中的有序抖動矩陣處理方式。類別矩陣內記錄著在區塊內要被處理像素點位置的先後類別，依照此處理類別，藉由擴散矩陣針對該處理位置進行擴散取樣誤差，此部分與誤差擴散法相同，將輸出後的取樣誤差擴散至周圍像素，重複上述步驟直到處理完所有區塊即可得到一張半色調影像。

接著詳細介紹演算法如下：假設此時的類別矩陣數值為 $o(m, n)$ ，在相同位置下其所對應的原始影像灰階為 $x(i, j)$ ，而 $c(i, j)$ 為在該位置更新後的灰階值。與誤差擴散法一樣，假設處理位置為類別矩陣數值 $o(m, n) = 0$ 時，代表該像素點為區塊內第一個處理點，並且沒有受到任何誤差擴散的補償，因此此時的 $c(i, j) = x(i, j)$ 。接下來將 $c(i, j)$ 以單一門檻值 128 進行比較，若大於等於此門檻值，則該位置之半色調像素點數值 $y(i, j)$ 輸出為 255，反之則輸出為 0。公式如下：

$$c(i, j) = x(i, j) + s(i, j), \text{ where } s(i, j) = \sum_{(p, q) \in R} \sum \frac{e(i+p, j+q) \times w(p, q)}{\text{sum}}, \quad (2.7)$$

$$e(i, j) = c(i, j) - y(i, j), \quad (2.8)$$

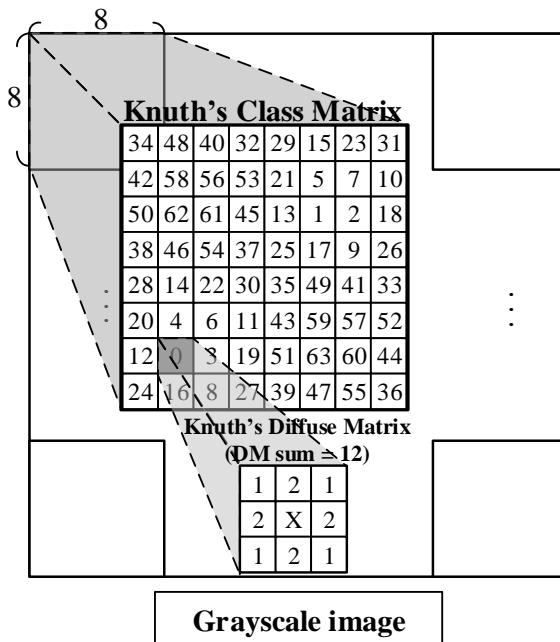
$$y(i, j) = \begin{cases} 255, & \text{if } c(i, j) \geq 128 \\ 0, & \text{if } c(i, j) < 128 \end{cases} \quad (2.9)$$

與誤差擴散法不同，由於擴散矩陣的大小固定為 3×3 ，因此 $p, q \in [-1, 1]$ 。值得注意的是，如同誤差擴散(ED)時要考量可能會超出影像的邊界，此外還要考慮到周圍像素是否已處理完畢，詳細公式如下：

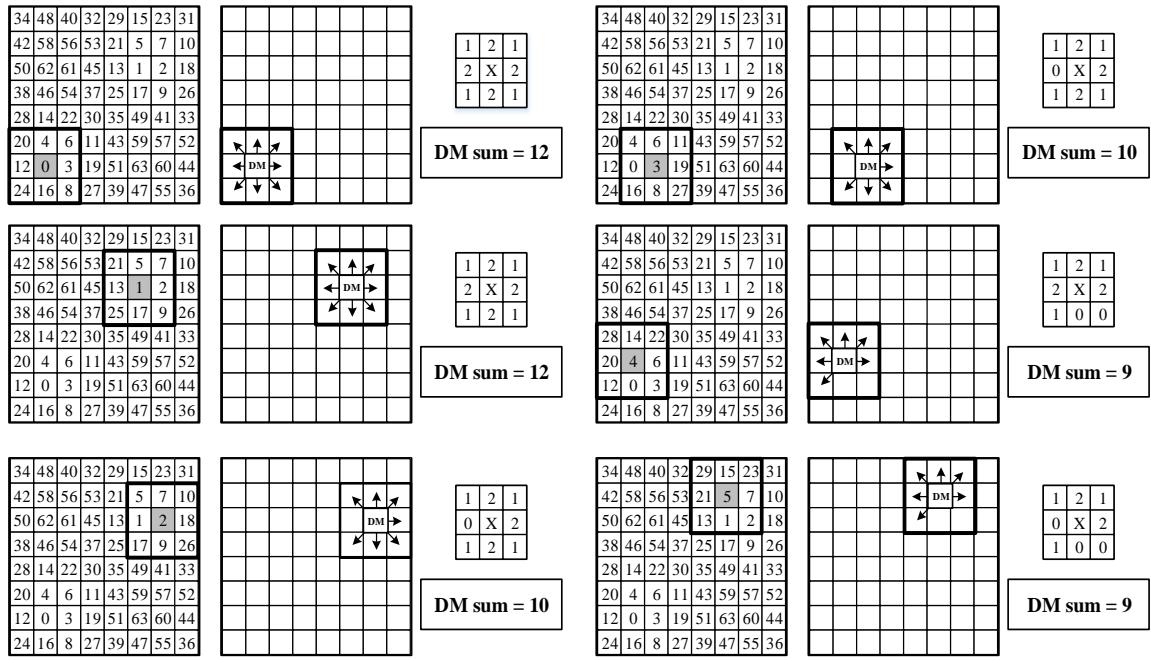
$$\text{sum} = \sum_{(p, q) \in R} \begin{cases} w(p, q), & \text{if } (i + p, j + q) \text{ within range and } o(m + p, n + q) > o(m, n) \\ 0, & \text{if } (i + p, j + q) \text{ out of range or } o(m + p, n + q) \leq o(m, n) \end{cases}, \quad (2.10)$$

當擴散目標的類別矩陣值小於或等於當前像素的類別矩陣值時，這表示擴散目標的像素已經處理過，因此不該將這個像素作為擴散的目標。

如圖 2.13 所表示，以 Knuth [9] 提出的 8×8 類別矩陣以及 3×3 擴散矩陣為例，針對每一個不同的類別數值進行擴散，由於先後類別的問題，所以擴散矩陣只會針對有效擴散位置進行擴散，並且重新計算擴散矩陣權重的總合，正規化使其係數總和為 1。另外，若遇到相鄰的不重疊區塊，邊界仍須判斷有效擴散位置並重新計算擴散矩陣總合，如圖 2.14 所示。由於一個區塊當中的像素點是否需要將誤差擴散至鄰近區塊仍然取決於類別矩陣的先後類別，代表區塊與區塊之間互為獨立運行，因此能夠平行處理，如同有序抖動法之處理方式。



(a) 點擴散法針對大小為 8×8 區塊、 8×8 類別矩陣與 3×3 的擴散矩陣處理之示意圖



(b) 不同類別數值之擴散情況(示意圖以類別矩陣數值 0 至 5 為例，可依序類推)

圖 2.13 類別矩陣擴散詳細示意圖

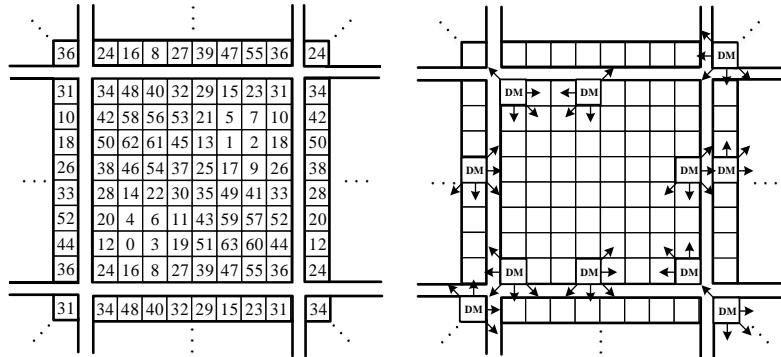


圖 2.14 類別矩陣之邊界擴散處理範例示意圖

在 Knuth [9]提出的文獻中，針對點擴散法的類別矩陣進行最佳化，其中最主要改善的便是類別矩陣當中的 Baron 與 Near-baron，Baron 代表目前處理位置周圍沒有有效擴散位置，而 Near-baron 代表目前處理位置周圍只有一個有效擴散位置。表 2.6 為最佳化後的類別矩陣以及提出的擴散矩陣，而該方法所產生的數位半色調影像圖 2.15 所示。但 Knuth 所提出的方法並沒有考量人類視覺之特性，故此方法產生的數位半色調影像之影像品質不高。

因此，Mese 與 Vaidyanathan [10]便提出了在點擴散法之中加入人類視覺之特性，並針對類別矩陣進行最佳化，表 2.7 為最佳化後 8×8 與 16×16 的類別矩陣以及提出的擴散矩陣，而該方法所產生的數位半色調影像如圖 2.16 所示。在 Mese 與 Vaidyanathan 所提出的方法所衍生的問題，Guo 與 Liu [11]提出將類別矩陣與擴散矩陣加入最佳化流程中，並配合多張自然影像加以訓練，即可訓練出符合自然影像特性的類別矩陣與擴散矩陣，表 2.8 為最佳化後 8×8 與 16×16 的類別矩陣以及對應其區塊大小的擴散矩陣，而該方法所產生的數位半色調影像如圖 2.17 所表示。雖然上述的方法都是針對類別矩陣與擴散矩陣去改良，提升半色調影像的品質，但是當人眼看點擴散法後的數位半色調影像，仍然會發現不自然的週期性紋理圖樣(Periodic pattern)，這是因為兩個不重疊區塊內的結構相似，且處理類別順序與擴散方式完全相同，便會產生相同的半色調影像圖樣，如圖 2.18 所表示。假設以整張影像以灰階 128 當作測試影像，用此三種方法做點擴散法，可以很明顯看出上述的不自然週期性紋理。

表 2.6 Knuth [9]提出之類別矩陣與擴散權重

34	48	40	32	29	15	23	31
42	58	56	53	21	5	7	10
50	62	61	45	13	1	2	18
38	46	54	37	25	17	9	26
28	14	22	30	35	49	41	33
20	4	6	11	43	59	57	52
12	0	3	19	51	63	60	44
24	16	8	27	39	47	55	36

(a) 8×8 類別矩陣

1	2	1
2	(i, j)	2
1	2	1

(b) 擴散矩陣

表 2.7 Mese 與 Vaidyanathan [10] 提出之類別矩陣與擴散權重

47	31	51	24	27	45	5	21
37	63	53	11	22	4	1	33
61	0	57	16	26	29	46	8
20	14	9	62	18	41	38	6
17	13	25	15	55	48	52	58
3	7	2	32	30	34	56	60
28	40	36	39	49	43	35	10
54	23	50	12	42	59	44	19

(a) 8×8 類別矩陣

1	2	1
2	(i, j)	2
1	2	1

(b) 擴散矩陣

207	0	13	17	28	55	18	102	81	97	74	144	149	169	170	172
3	6	23	36	56	50	65	87	145	130	137	158	182	184	195	221
7	14	24	37	67	69	86	5	106	152	150	165	183	192	224	1
15	26	43	53	51	101	115	131	139	136	166	119	208	223	226	4
22	39	52	71	84	103	164	135	157	173	113	190	222	225	227	16
40	85	72	83	104	117	167	133	168	180	200	219	231	228	12	21
47	120	54	105	123	132	146	176	179	202	220	230	245	2	20	41
76	73	127	109	138	134	178	181	206	196	229	244	246	19	42	49
80	99	112	147	142	171	177	203	218	232	243	248	247	33	48	68
108	107	140	143	185	163	204	217	233	242	249	255	44	45	70	79
110	141	88	75	175	205	214	234	241	250	254	38	46	77	116	100
111	148	160	174	201	215	235	240	251	252	253	61	62	93	94	125
151	159	189	199	197	216	236	239	25	31	60	82	92	95	124	114
156	188	191	209	213	237	238	29	32	59	64	91	118	78	128	155
187	194	198	212	9	10	30	35	58	63	90	96	122	129	154	161
193	210	211	8	11	27	34	57	66	89	98	121	126	153	162	186

(c) 16×16 類別矩陣

表 2.8 Guo 與 Liu [11]提出之類別矩陣與擴散權重

22	5	57	8	45	30	36	19
40	58	32	18	1	43	29	38
34	4	62	42	20	16	48	37
28	7	21	56	15	3	49	11
6	23	35	17	55	51	50	44
47	12	39	26	25	27	63	61
14	46	41	31	2	33	60	13
9	24	52	0	53	54	59	10

(a) 8×8 類別矩陣

0.47972	1	0.47972
1	(i,j)	1
0.47972	1	0.47972

(b) 使用於 8×8 類別矩陣之擴散矩陣

0.38459	1	0.38459
1	(i,j)	1
0.38459	1	0.38459

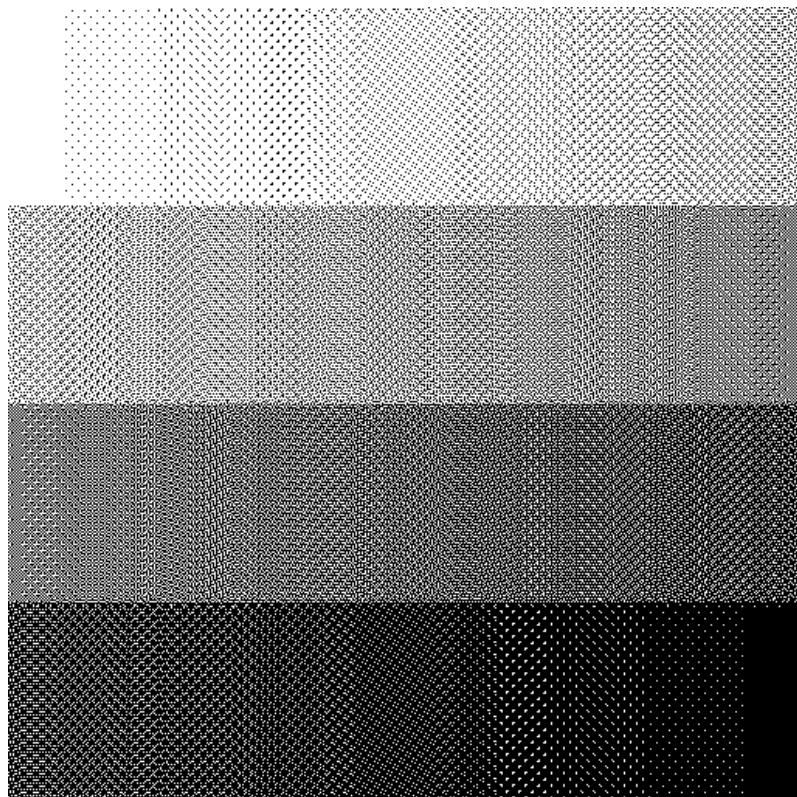
(c) 使用於 16×16 類別矩陣之擴散矩陣

204	0	5	33	51	59	23	118	54	69	40	160	169	110	168	188
3	6	22	36	60	50	74	115	140	82	147	164	171	142	220	214
14	7	42	16	63	52	94	56	133	152	158	177	179	208	222	1
15	26	43	75	79	84	148	81	139	136	166	102	217	219	226	4
17	39	72	92	103	108	150	135	157	193	190	100	223	225	227	13
28	111	99	87	116	131	155	112	183	196	181	224	232	228	12	21
47	120	91	105	125	132	172	180	184	205	175	233	245	8	20	41
76	65	129	137	165	145	178	194	206	170	229	244	246	19	24	49
80	73	106	138	176	182	174	197	218	235	242	249	247	18	48	68
101	107	134	153	185	163	202	173	231	241	248	253	44	88	70	45
123	141	149	61	195	200	221	234	240	243	254	38	46	77	104	109
85	96	156	130	203	215	230	250	251	252	255	53	62	93	86	117
151	167	189	207	201	216	236	239	25	31	34	113	83	95	124	114
144	146	191	209	213	237	238	29	32	55	64	97	126	78	128	159
187	192	198	212	9	10	30	35	58	67	90	71	122	127	154	161
199	210	211	2	11	27	37	57	66	89	98	121	119	143	162	186

(d) 16×16 類別矩陣

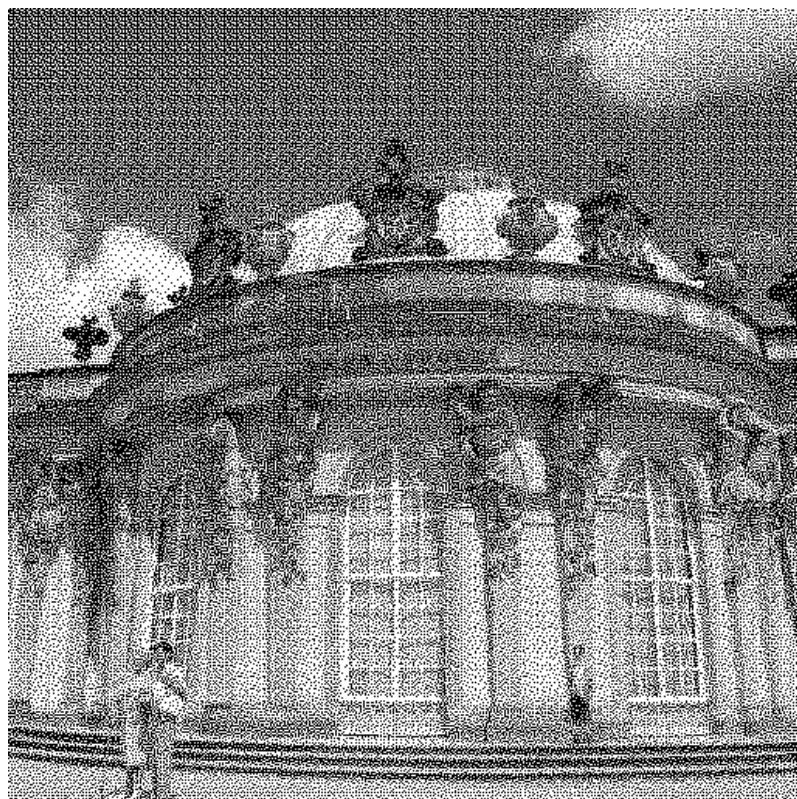


(a) 灰階影像 1 經點擴散法處理後的半色調影像

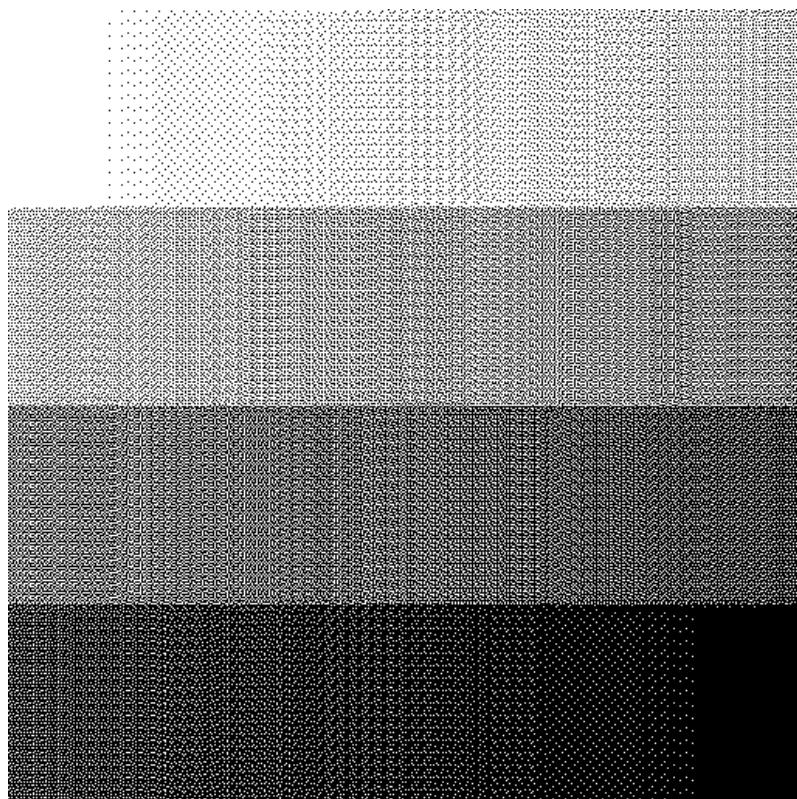


(b) 灰階影像 2 經點擴散法處理後的半色調影像

圖 2.15 使用 Knuth 提出的類別矩陣及擴散矩陣所產生的半色調影像



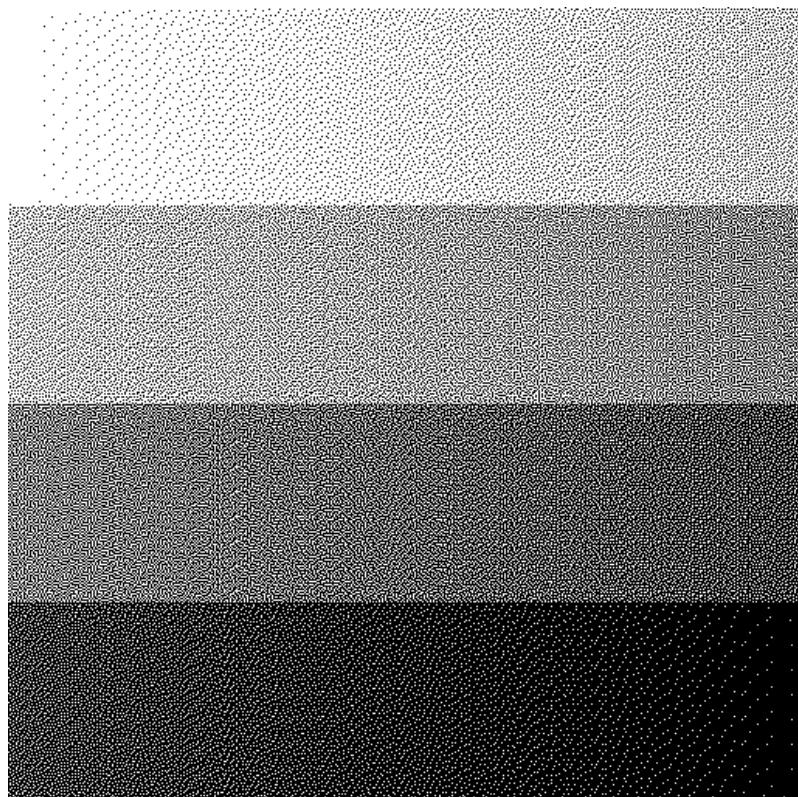
(a) 灰階影像 1 經點擴散法處理後的半色調影像(8×8 類別矩陣)



(b) 灰階影像 2 經點擴散法處理後的半色調影像(8×8 類別矩陣)

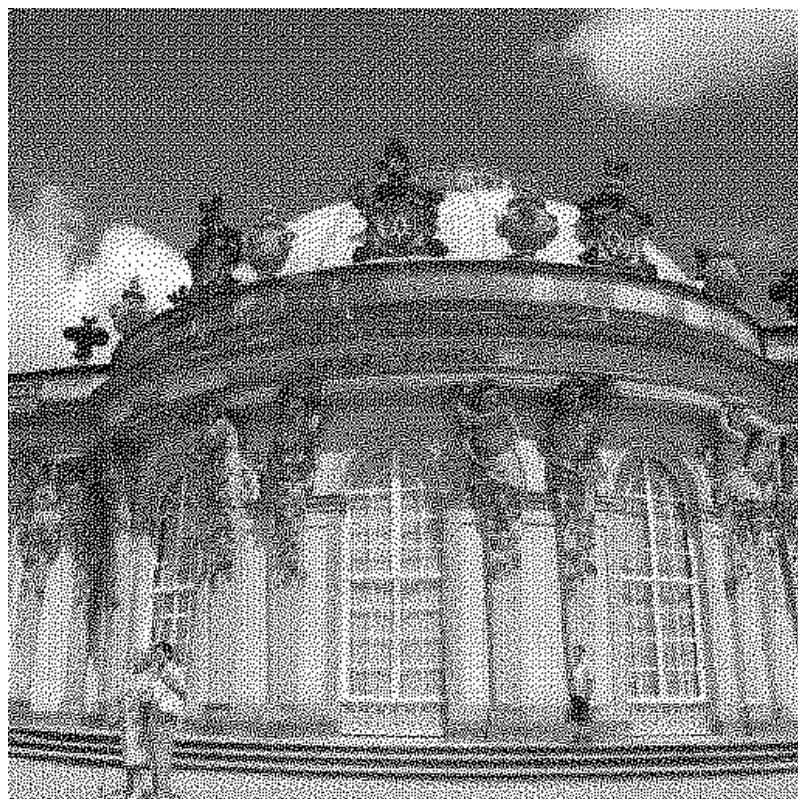


(c) 灰階影像 1 經點擴散法處理後的半色調影像(16×16 類別矩陣)

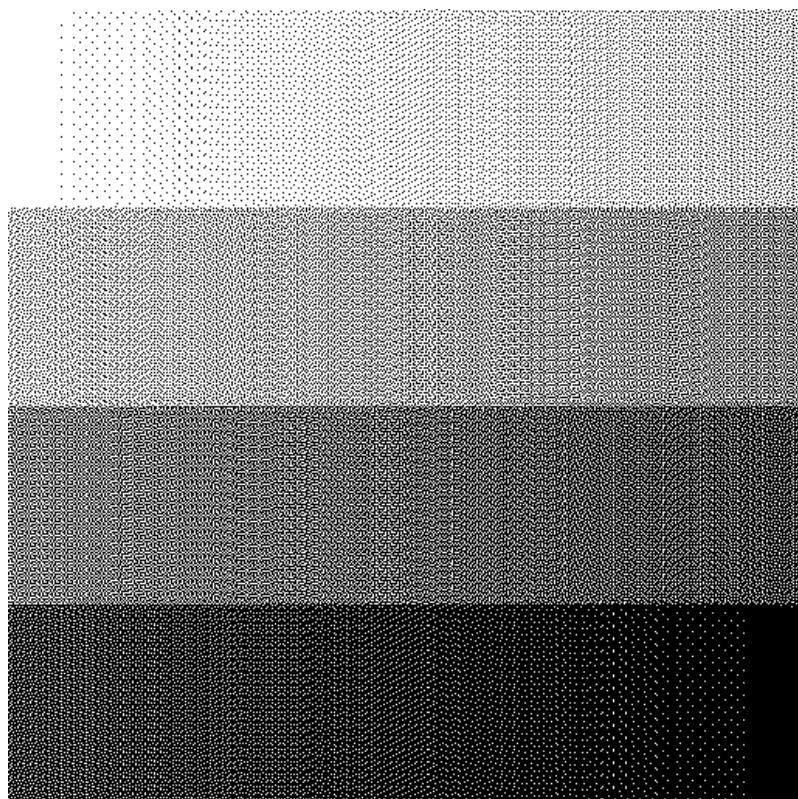


(d) 灰階影像 2 經點擴散法處理後的半色調影像(16×16 類別矩陣)

圖 2.16 使用 Mese 與 Vaidyanathan 提出的類別矩陣及擴散矩陣所產生的半色調影像



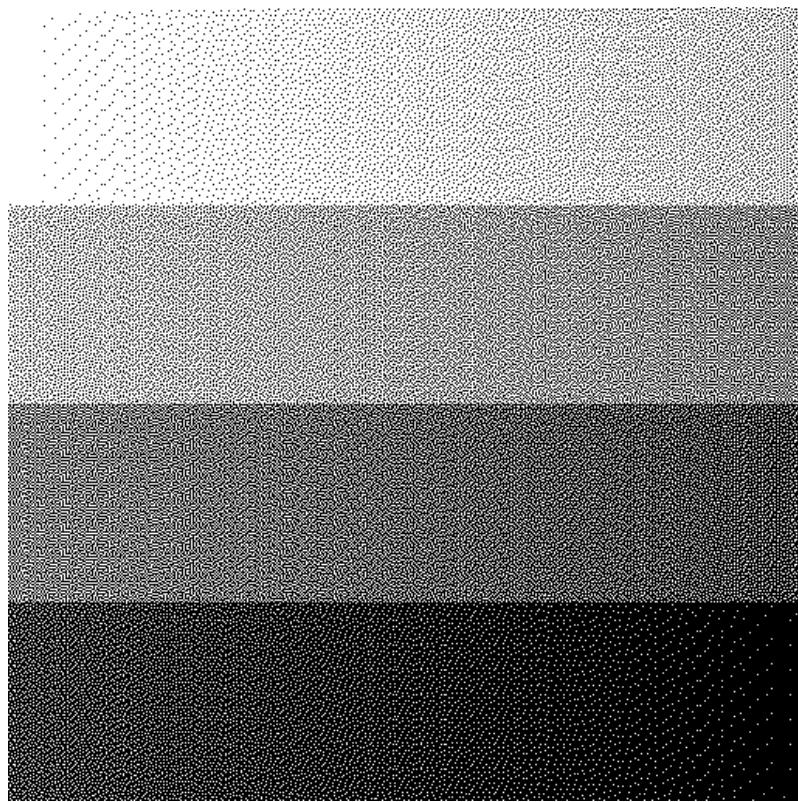
(a) 灰階影像 1 經點擴散法處理後的半色調影像(8×8 類別矩陣)



(b) 灰階影像 2 經點擴散法處理後的半色調影像(8×8 類別矩陣)



(c) 灰階影像 1 經點擴散法處理後的半色調影像(16×16 類別矩陣)

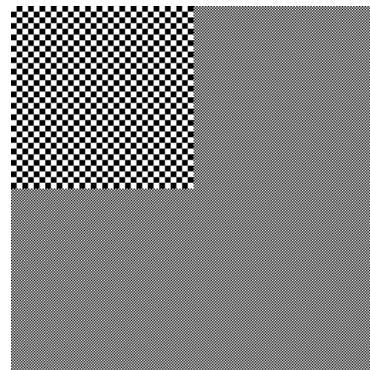


(d) 灰階影像 2 經點擴散法處理後的半色調影像(16×16 類別矩陣)

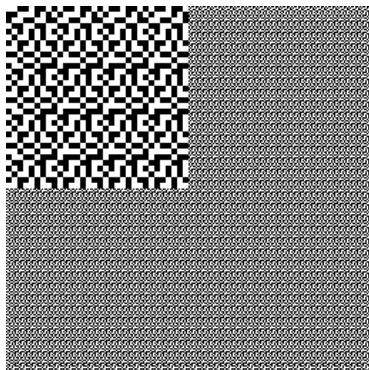
圖 2.17 使用 Guo 與 Liu 提出的類別矩陣及擴散矩陣所產生的半色調影像



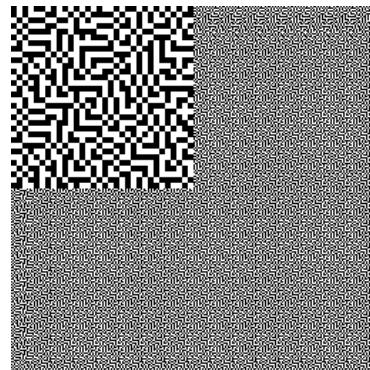
(a) 測試影像(灰階值 128)



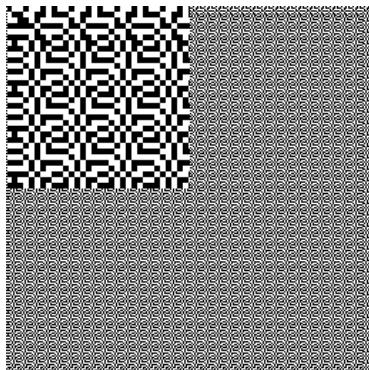
(b) Knuth



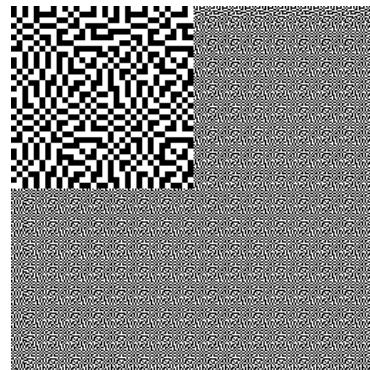
(c) Mese 與 Vaidyanathan (8×8)



(d) Mese 與 Vaidyanathan (16×16)



(e) Guo 與 Liu (8×8)



(f) Guo 與 Liu (16×16)

圖 2.18 以各種點擴散法所產生出的不自然週期性紋理

2.4 直接二元搜尋法(Direct binary search, DBS)

直接二元搜尋法是由 Allebach 與 Analoui [12]在 1992 年所提出，往後亦有許多其相關延伸的研究[13]-[16]。直接二元搜尋法是一種迭代式的最佳化演算法，其主要特色是能夠將一張隨機產生的半色調影像，藉由重複改變點分佈並以成本函數(Cost function)評估目前的點分佈是否為好的效果，直到目前所有可能的點分佈都無法再次帶來更好的效果(即收斂)，即可得到一張近似於原始影像的半色調影像。由於此演算法需要耗費大量的時間進行迭代，因此在實際應用的價值較低，但其半色調影像品質極佳，因此許多半色調研究都以直接二元搜尋法的結果為目標進行比較。

首先，先介紹傳統直接二元搜尋法(DBS)的流程。假設 $g[m, n]$ 、 $b[m, n]$ 分別為連續的原始影像及半色調影像，而此時定義一個函數 $p(x, y)$ 為離散的點擴展函數(Point spreadfunction, PSF)，此函數即是一個模擬人眼視覺系統(Human visual system, HVS)的模型。並利用該函數 $p(x, y)$ 與原始影像 $g[m, n]$ 及半色調影像 $b[m, n]$ 做摺積(convolution)，來模擬人眼視覺上的效果，其公式如下：

$$g(x, y) = \sum_{m,n} g[m, n]p(x - mX, y - nY), \quad (2.11)$$

$$b(x, y) = \sum_{m,n} b[m, n]p(x - mX, y - nY), \quad (2.12)$$

其中，半色調影像 $b[m, n]$ 是以隨機產生二位元半色調的方式做初始化。由於直接二元搜尋法(DBS)的目標是希望能夠讓 $b(x, y)$ 趨近於 $g(x, y)$ ，因此將定義成本函數 E 為目前兩張影像點對點之平方差總和，公式如下：

$$E = \|g(x, y) - b(x, y)\|^2, \quad (2.13)$$

其中，藉由不斷地調整 $b[m, n]$ ，使得 E 會慢慢趨近最小值。調整方式即是以翻點(Toggle)與交換(Swap)兩種方式進行改變，如圖 2.19 表示。翻點代表將該處理位置的像素點進行反向的動作，將像素點從黑變白或從白變黑，而交換代表將該處理位置的像素點與周圍的八個像素點進行交換，總共會有八種可能的點分佈。將上述包含原始點分佈狀態在內的十種不同點分佈組合，選擇一組能使成本函數產生最小值的點分佈當作更新後的點分佈，更新的半色調影像定義為 $b'[m, n]$ ，經摺積後為 $b'(x, y)$ 。將這些可能的嘗試代入公式(2.13)可得到 E' 如下：

$$E' = \|g(x, y) - b'(x, y)\|^2, \quad (2.14)$$

將可能的嘗試結果 E' 中最低的結果與 E 做比較，如果低於就表示這個動作(翻點或交換)具有最低成本的點分佈，因此將此最低的 E' 取代 E 並將其對應的 $b'(x, y)$ 取代 $b(x, y)$ 。依照上述方法不斷地重複進行，直到成本函數不再下降為止。

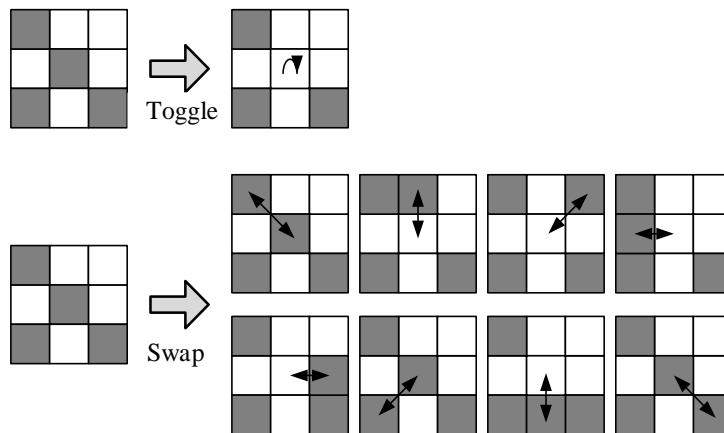


圖 2.19 翻點(Toggle)與交換(Swap)處理示意圖

由於上述的做法非常耗時，尤其是當公式(2.14)要取得 $b'(x, y)$ 時，需要透過公式(2.12)進行摺積。因此 Allebach 對上述公式進行化簡，並在 1997 年提出高效率直接二元搜尋法(Efficient DBS) [14]。主要概念為考慮一個像素點能被多少像素點影響，並產生對應的誤差值影像，其計算對應產生的誤差當作此像素點帶來的誤差。可以想像為一個像素點帶來的影響力，以誤差表示代表希望其值越低越好。接著在往後更動半色調影像時只需要更新目前像素點在誤差影像中的影響區域，而不需冗長的重複計算。公式如下：

$$e[m, n] = g[m, n] - b[m, n], \quad (2.15)$$

$$e(x, y) = \sum_{m,n} e[m, n]p(x - mX, y - nY), \quad (2.16)$$

$$E = \int \int |e(x, y)|^2 dx dy, \quad (2.17)$$

藉由上述公式，以下公式以一維方式推導並簡化，因此成本函數可改寫如下：

$$\begin{aligned}
 E &= \int \sum_m \sum_n e[m]e[n]p(x - Xm)p(x - Xn) dx \\
 &= \sum_m \sum_n e[m]e[n]c_{pp}(n - m) \\
 &= \sum_m e[m]c_{ep}[m]
 \end{aligned} \tag{2.18}$$

其中， c_{pp} 為點擴展函數做自相關(Auto correlation)運算，而 c_{ep} 則是將 c_{pp} 函數與相減影像 $e[n]$ 做摺積，其公式定義如下：

$$c_{pp}(x) = \int p(y)p(y - x) dy, \tag{2.19}$$

$$c_{ep}[m] = \sum_n e[n]c_{pp}(n - m), \tag{2.20}$$

透過上述公式的化簡，當半色調影像在進行翻點與交換時，只需觀察 $e[m]$ 與 $c_{ep}[m]$ 對成本函數的影響。接下來將定義翻點對於 $b[m]$ 、 $e[m]$ 與 $c_{ep}[m]$ 之間的關係，其公式如下：

$$b'[m] = b[m] + a_0\delta[m - m_0], \tag{2.21}$$

$$e'[m] = e[m] + a_0\delta[m - m_0], \text{ where } \delta[m, n] = \begin{cases} 1, & \text{if } [m, n] = [0, 0], \\ 0, & \text{otherwise} \end{cases}, \tag{2.22}$$

$$c'_{ep}[m] = c_{ep}[m] + a_0c_{pp}[m - m_0], \tag{2.23}$$

其中， a_0 定義將在後續說明。而翻點對於成本函數 ΔE_0 之間的關係，該公式如下：

$$\begin{aligned}
 \Delta E_0 &= \int |e'(x)|^2 dx - \int |e(x)|^2 dx \\
 &= \sum_m (e'[m]c_{ep}'[m] - e[m]c_{ep}[m]) \\
 &= \sum_m (e[m] + a_0\delta[m - m_0])(c_{ep}[m] + a_0c_{pp}[m - m_0]) - e[m]c_{ep}[m] \\
 &= \sum_m (e[m]c_{ep}[m] + a_0e[m]c_{pp}[m - m_0]) + (a_0c_{ep}[m_0] + a_0^2c_{pp}[0]) \\
 &\quad - e[m]c_{ep}[m] \\
 &= \sum_m (a_0e[m]c_{pp}[m - m_0] + a_0c_{ep}[m_0] + a_0^2c_{pp}[0]) \\
 &= 2a_0c_{ep}[m_0] + a_0^2c_{pp}[0]
 \end{aligned} \tag{2.24}$$

再來是定義交換，它可以看成是針對點 m_0 與 m_1 分別做翻點，即可達到交換的效果，因此交換對於成本函數 ΔE_1 之間的關係與 ΔE_0 定義相同，而成本函數 ΔE 即為 ΔE_0 與 ΔE_1 的總合，其公式寫成如下：

$$\begin{aligned}
 \Delta E &= \Delta E_0 + \Delta E_1 \\
 &= (2a_0c_{ep}[m_0] + a_0^2c_{pp}[0]) + (2a_1c'_{ep}[m_1] + a_1^2c_{pp}[0]) \\
 &= 2a_0c_{ep}[m_0] + a_0^2c_{pp}[0] + 2a_1(c_{ep}[m_1] + a_0c_{pp}[m_1 - m_0]) + a_1^2c_{pp}[0] \\
 &= 2a_0c_{ep}[m_0] + a_0^2c_{pp}[0] + 2a_1c_{ep}[m_1] + 2a_0a_1c_{pp}[m_1 - m_0] + a_1^2c_{pp}[0] \\
 &= (a_0^2 + a_1^2)c_{pp}[0] + 2a_0c_{ep}[m_0] + 2a_1c_{ep}[m_1] + 2a_0a_1c_{pp}[m_1 - m_0]
 \end{aligned} \tag{2.25}$$

最後，將成本函數 ΔE 以二維形式來描述，則公式可以寫成如下：

$$\Delta E = (a_0^2 + a_1^2)c_{pp}[0,0] + 2a_0c_{ep}[m_0, n_0] + 2a_1c_{ep}[m_1, n_1] + 2a_0a_1c_{pp}[m_1 - m_0, n_1 - n_0], \quad (2.26)$$

a_0 及 a_1 在不同的動作會有不同的數值，翻點時的數值如下：

$$\begin{aligned} a_0 &= \begin{cases} 1, & \text{if } b[m - m_0, n - n_0] = 0 \\ -1, & \text{if } b[m - m_0, n - n_0] = 1 \end{cases}, \\ a_1 &= 0 \end{aligned} \quad (2.27)$$

交換時的數值如下：

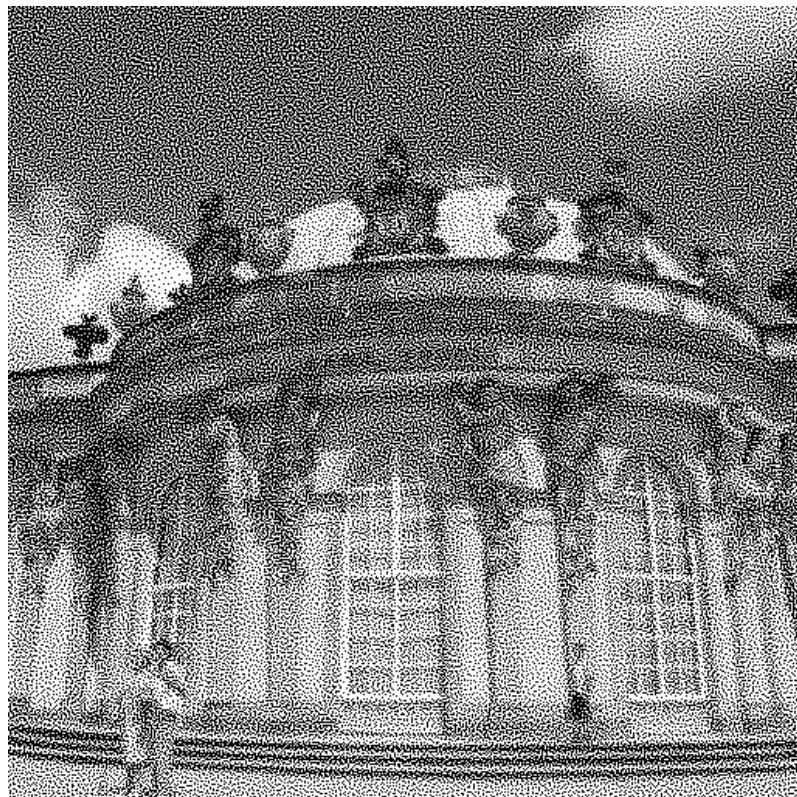
$$\begin{aligned} a_0 &= \begin{cases} 1, & \text{if } b[m - m_0, n - n_0] = 0 \\ -1, & \text{if } b[m - m_0, n - n_0] = 1 \end{cases} \\ a_1 &= \begin{cases} 0, & \text{if } b[m - m_0, n - n_0] = b[m - m_1, n - n_1] \\ -a_0, & \text{if } b[m - m_0, n - n_0] \neq b[m - m_1, n - n_1] \end{cases} \end{aligned} \quad (2.28)$$

其中 m_0 與 n_0 表示 $C_{ep}[m, n]$ 的中心位置， m_1 與 n_1 表示相對應中心點的交換位置，譬如與相對應目前處理像素點的 $[1, -1]$ 交換，因此 $m_1 = 1$ 與 $n_1 = -1$ 。上述成本函數公式 ΔE ，包含了翻點與交換等 10 種不同狀態。其中若 ΔE 有最小值，即是代表該黑白點分佈為最佳狀態，因此即可更新 $b'[m, n]$ 與 $c'_{ep}[m, n]$ ，其更新公式可以寫成如下：

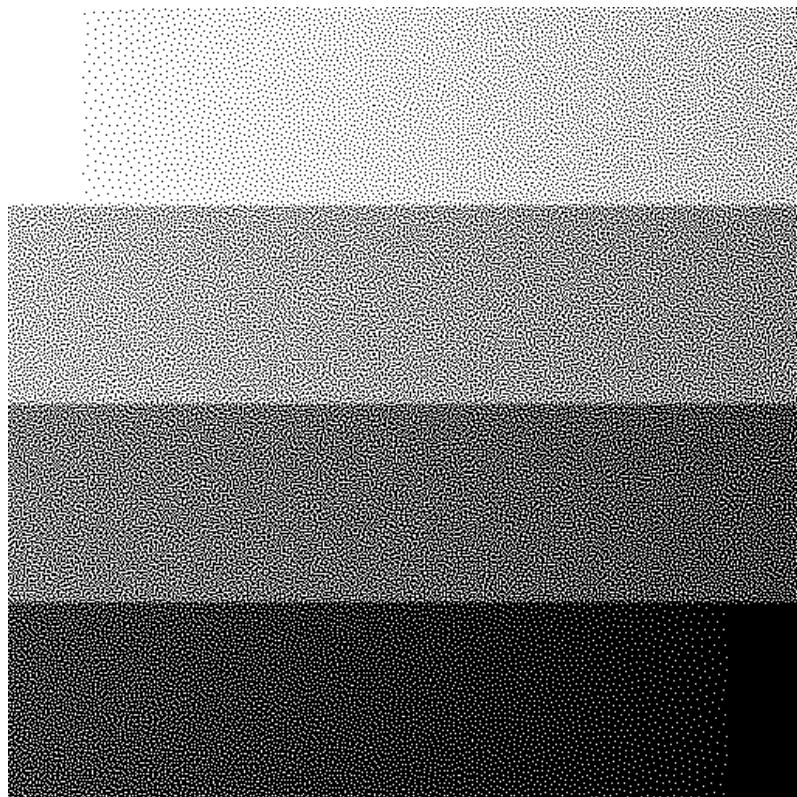
$$b'[m, n] = b[m, n] + a_0\delta[m - m_0, n - n_0] + a_1\delta[m - m_1, n - n_1], \quad (2.29)$$

$$c'_{ep}[m, n] = c_{ep}[m, n] + a_0c_{pp}[m - m_0, n - n_0] + a_1c_{pp}[m - m_1, n - n_1], \quad (2.30)$$

藉由上述公式重複不斷地提升影像品質，直到所有像素點的 ΔE 都不小於 0，即可得到一張直接二元搜尋法的影像。圖 2.20 為直接二元搜尋法產生的半色調結果，觀察後可以發現其影像品質非常優異，其半色調影像除了沒有前面介紹的數種不自然紋理，此外更預備優異的藍雜訊(Blue-noise)點分佈特性。上述優點使得學術研究中常以直接二元搜尋法的半色調結果當作最佳比較對象(Ground truth)。儘管高效率的直接二元搜尋法(EDBS)比起傳統的直接二元搜尋法(DBS)能夠節省處理時間 100 到 1000 倍，但比起其他技術仍然需要較長的時間運算，因此無法實際應用在印刷領域。



(a) 灰階影像 1 經直接二元搜尋法處理後的半色調影像



(b) 灰階影像 2 經直接二元搜尋法處理後的半色調影像

圖 2.20 使用 7×7 高斯模糊作為點擴展函數所產生的半色調影像

2.5 雙指標直接二元搜尋法(Dual metric DBS, DMDBS)

此為 Kim 與 Allebach [16]於 2002 年提出以 DBS 為基礎做改良的技術，名為雙指標直接二元搜尋法。人眼視覺系統(Human visual system, HVS)模型在很多半色調演算法中扮演著重要的角色。因此，此研究在 Näsänen [17]提出的人眼視覺系統模型中挑選最適合直接二元搜尋法(DBS)的模型，但由於直接二元搜尋法使用此模型的運算複雜度太高，所以作者接著以此模型為基礎，提出一類似的高斯模型套用在高效率直接二元搜尋法上，使其結果能夠有著極高的影像品質又能有不錯的處理速度。詳細的公式說明如下：

$$e[m, n] = g[m, n] - b[m, n], \quad (2.31)$$

$$e_i(x, y) = \sum_{m,n} w_i[g[m, n]] e[m, n] p(x - mX, y - nY), \quad (2.32)$$

$$E = \sum_{i=1}^2 \int \int |e_i(x, y)|^2 dx dy, \quad (2.33)$$

公式(2.32)所述的 $w_i[g[m, n]]$ 為一隨著灰階值變動的權重，它表示的是半色調中藍雜訊(Blue-noise)特性理想的點距離，其定義如下：

$$w_1[g] = \begin{cases} \sqrt{1 - (4g - 1)^2}, & \text{if } 0 \leq g < \frac{1}{4} \\ |4g - 2|, & \text{if } \frac{1}{4} \leq g < \frac{3}{4}, \\ \sqrt{1 - (4g - 3)^2}, & \text{if } \frac{3}{4} \leq g < 1 \end{cases} \quad (2.34)$$

$$w_2[g] = 1 - w_1[g]$$

藉由上述公式，可將成本函數化簡如下：

$$E = \sum_{i=1}^2 \sum_{m,n} \sum_{k,l} e_i[m, n] e_i[k, l] c_{p_i p_i}(m - k, n - l), \quad (2.35)$$

在文獻中，自相關(Auto-correlation)函數 $c_{p_i p_i}$ 所使用的點擴展函數為 Näsänen 的人眼視覺系統模型，它近似於一種雙指標的高斯函數，其詳細公式如下：

$$c_{p_i p_i}[m, n] = \frac{180^2}{(\pi D)^2} c_{h_i h_i} \left(\frac{180m}{\pi S}, \frac{180n}{\pi S} \right), \quad (2.36)$$

其 $c_{h_i h_i}(x, y)$ 定義如下：

$$c_{h_i h_i}(x, y) = k_1 \exp \left(-\frac{u^2 + v^2}{2\sigma_1^2} \right) + k_2 \exp \left(-\frac{u^2 + v^2}{2\sigma_2^2} \right), \quad (2.37)$$

在此， S 為尺度參數並設 $S = RD$ ， R 及 D 分別表示解析度(dpi)及視覺距離(inch)， R 及 D 如文獻所述分別設定為 300 dpi 及 9.5 inch；為了取得最佳的半色調影像，雙指標高

斯模型 $c_{p_1p_1}$ 及 $c_{p_2p_2}$ 的參數 $(k_1, k_2, \sigma_1, \sigma_2)$ 如同文獻所定義分別被設為 $(43.2, 38.7, 0.0219, 0.0598)$ 及 $(19.1, 42.7, 0.0330, 0.0569)$ 。接著，藉由公式(2.35)可將每次動作產生的成本函數簡化成以下公式：

$$\begin{aligned}\Delta E = \sum_{i=1}^2 & \{(a_0^2 w_i^2 [g[m_0, n_0]] + a_1^2 w_i^2 [g[m_1, n_1]]) c_{p_ip_i}[0,0] \\ & + 2a_0 w_i [g[m_0, n_0]] c_{ep}[m_0, n_0] + 2a_1 w_i [g[m_1, n_1]] c_{ep}[m_1, n_1], \\ & + 2a_0 w_i [g[m_0, n_0]] a_1 w_i [g[m_1, n_1]] c_{pp}[m_1 - m_0, n_1 - n_0]\}\end{aligned}\quad (2.38)$$

如同直接二元搜尋法， a_0 及 a_1 在不同的動作會有不同的數值，翻點時的數值如下所示：

$$\begin{aligned}a_0 &= \begin{cases} 1, & \text{if } b[m - m_0, n - n_0] = 0 \\ -1, & \text{if } b[m - m_0, n - n_0] = 1 \end{cases}, \\ a_1 &= 0\end{aligned}\quad (2.39)$$

交換時的數值如下：

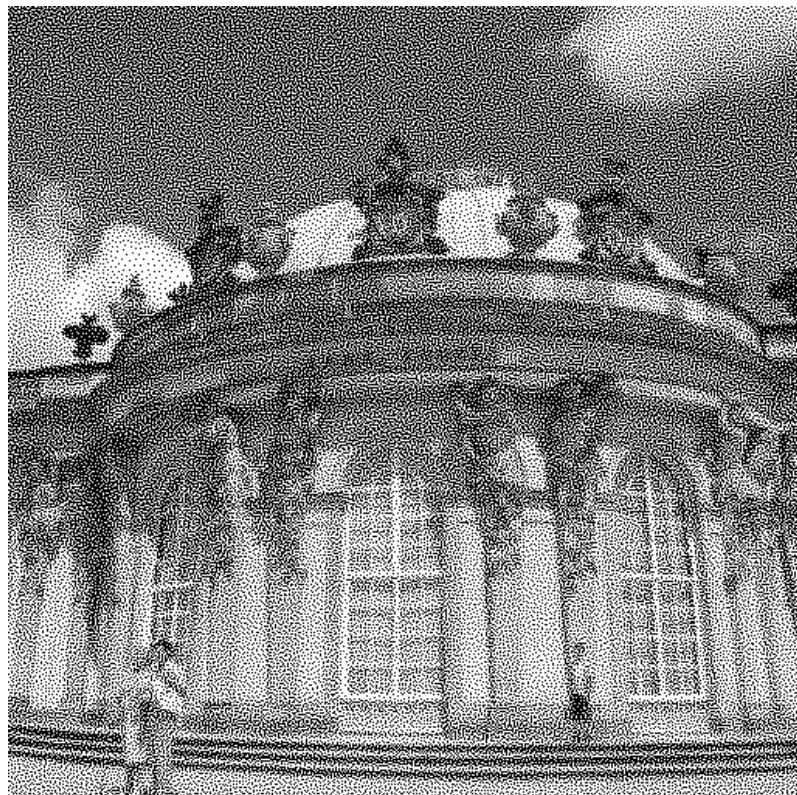
$$\begin{aligned}a_0 &= \begin{cases} 1, & \text{if } b[m - m_0, n - n_0] = 0 \\ -1, & \text{if } b[m - m_0, n - n_0] = 1 \end{cases} \\ a_1 &= \begin{cases} 0, & \text{if } b[m - m_0, n - n_0] = b[m - m_1, n - n_1] \\ -a_0, & \text{if } b[m - m_0, n - n_0] \neq b[m - m_1, n - n_1] \end{cases}\end{aligned}\quad (2.40)$$

半色調結果影像 b 及 c_{ep} 函數可藉由以下公式做更新：

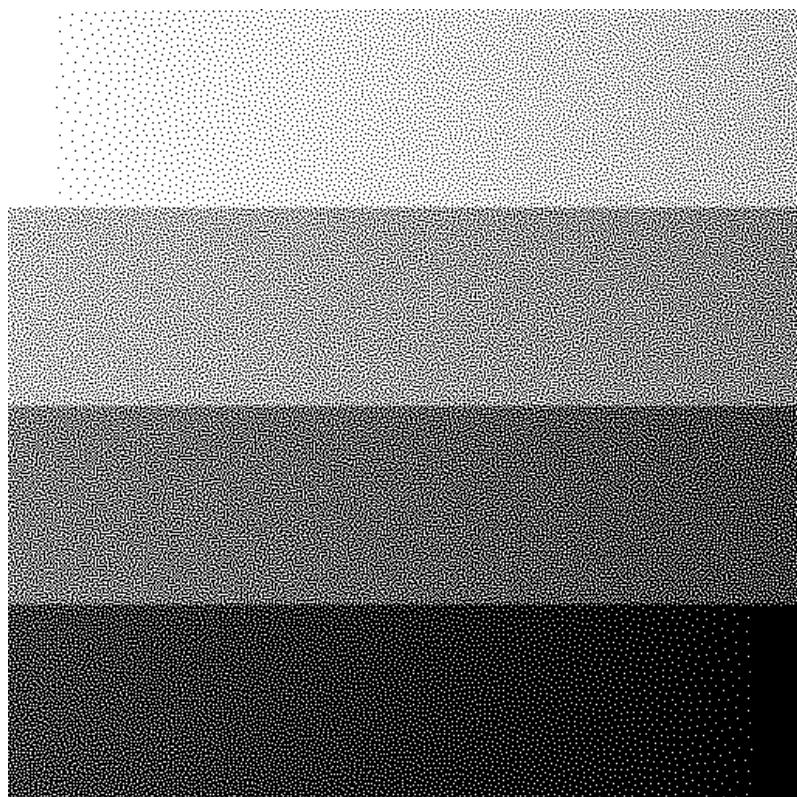
$$b'[m, n] = b[m, n] + a_0 \delta[m - m_0, n - n_0] + a_1 \delta[m - m_1, n - n_1], \quad (2.41)$$

$$c'_{e_ip_i}[m, n] = c_{ep}[m, n] + a_0 w_i [g[m_0, n_0]] c_{pp}[m - m_0, n - n_0] \\ + a_1 w_i [g[m_1, n_1]] c_{pp}[m - m_1, n - n_1], \quad (2.42)$$

藉由上述公式重複不斷地提升影像品質，直到所有像素點的 ΔE 都不小於 0，即可得到雙指標直接二元搜尋法的半色調結果。如圖 2.21 所示為測試影像經過使用 Näsänen 人眼視覺系統模型的雙指標直接二元搜尋法所產生的半色調影像，其半色調影像與直接二元搜尋法一樣具備著優異的藍雜訊分佈特性，也沒有不自然的圖案產生。除此之外，雙指標直接二元搜尋法的結果比起直接二元搜尋法更加平滑自然，但因為要考量兩個高斯函數的關係使其處理速度會比高效率直接二元搜尋法還慢一些。因此，雙指標直接二元搜尋法仍然無法實際應用，只能當作最佳的比較對象(Ground truth)。



(a) 灰階影像 1 經雙指標直接二元搜尋法處理後的半色調影像



(b) 灰階影像 2 經雙指標直接二元搜尋法處理後的半色調影像

圖 2.21 使用 Näsänen 的人眼視覺系統模型所產生的半色調影像

第三章 區塊截斷編碼技術文獻探討

區塊截斷編碼(Block truncation coding, BTC)為 Delp 及 Mitchell [18]所提出，是一種有損的影像壓縮技術。這技術將影像切割成多個不重疊的區塊，各個區塊可獨立處理並且以兩個簡單的資訊：位元圖(Bitmap)及量化色階(Quantization level)來重建影像，因此 BTC 不僅有不錯的壓縮率且處理效率高。由於這些優勢，BTC 被廣泛地應用在隱寫技術(Steganography) [19]、影像檢索(Image retrieval) [20]-[21] 及浮水印技術(Watermark) [22]-[24] 等領域。對於應用程序上需求低計算複雜度的編碼器或者追求高幀速率的應用，例如：高速相機或者設計用於大量影像/影片資料庫的搜尋引擎等，BTC 能夠滿足這些實用上的需求。

為了改善影像視覺上的品質，過往的 BTC 技術至今為止都是高度注重在改良位元圖或者量化色階其中一方。舉例來說，絕對矩量區塊截斷編碼(Absolute moment BTC, AMBTC) [25]為改善量化色階的知名代表，使用一階絕對中心矩量取代傳統 BTC 的量化色階，不僅能夠降低計算複雜度，也降低了重建影像與原始影像訊號上的誤差(藉由 Mean squared error (MSE)評估)。但傳統 BTC 固有的模糊及區塊效應(Blocking effect)等主要缺陷仍然沒有解決。

過往傳統的彩色 BTC 技術是直接在 RGB 或者其他色彩模型上各通道分別執行，但它們最終會得到如同灰階壓縮時的壓縮率。由於彩色影像由三個平面紅(R)、綠(G)、藍(B)所組成，在一張影像中它們之間的相關性非常地高，因此這三通道經過 BTC 處理後容易產生十分相似的位元圖。藉由此特性，可將三通道的位元圖減少為一個，用單一的位元圖去重建彩色影像，此方法稱為單一位元圖技術(Single bit-map, SBM) [26]。雖然這種方法能夠顯著地提升壓縮率，但也因為三通道都使用相同位元圖重建使得區塊效應變得更加嚴重，導致視覺品質會急遽變差。

另一方面，數位半色調技術[8]與 BTC 技術的結合是個新銳的方法用來改善位元圖，即量化色階的分佈。數位半色調是著名的影印技術，透過有限的顏色模擬連續色調圖像。基於半色調技術的 BTC 方法有一個獨特的特性，就是可以藉由視覺偏愛的

抖動點渲染整張影像，明顯地消除傳統 BTC 及 AMBTC 固有的區塊效應，但也因此使結果影像受到脈衝雜訊(Impulse noise)的影響。

以下的區塊截斷編碼技術依照處理類型能分類為：

1) 灰階影像：

- 傳統的區塊截斷編碼(BTC)
- 絕對矩量區塊截斷編碼(AMBTC)
- 有序抖動區塊截斷編碼(Ordered dither BTC, ODBTC)
- 誤差擴散區塊截斷編碼(Error-diffused BTC, EDBTC)
- 點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)

2) 彩色影像：

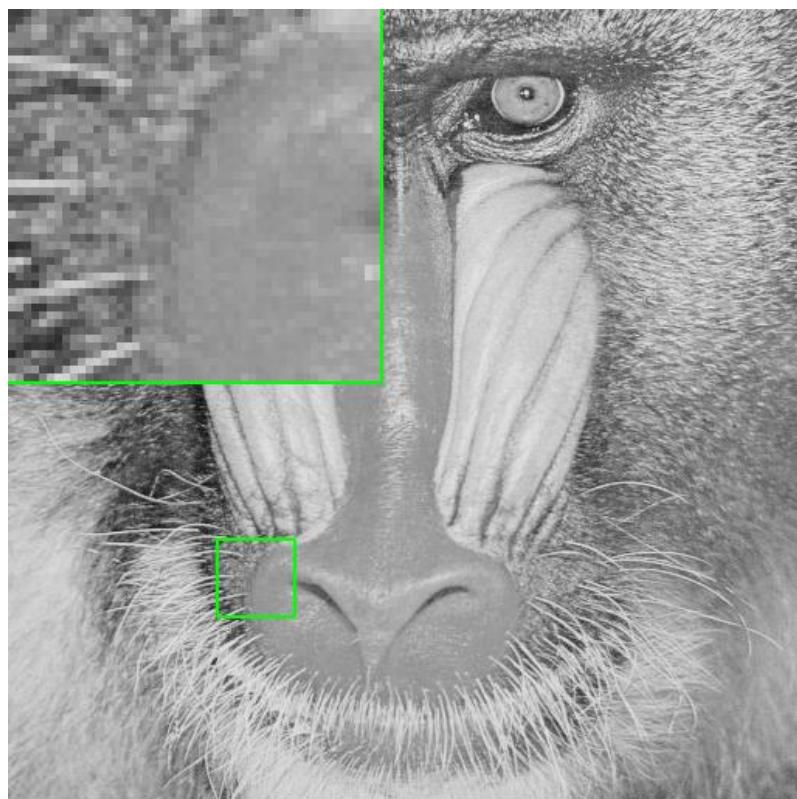
- 彩色絕對矩量區塊截斷編碼(Color AMBTC, CAMBTC)
- 單一位元圖區塊截斷編碼(Single bit-map BTC, SBMBTC)
- 基於單一位元圖的編碼方案(Coding schemes based on SBM, CS-SBM)
- 分群區塊截斷編碼(BTC with k-means quad clustering, IBTC-KQ)
- 彩色點擴散區塊截斷編碼(Color DDBTC, CDDBTC)

這些技術將在後續章節依照提出的時間做介紹。此外，上述方法並非所有區塊截斷編碼技術，這邊所介紹的技術為代表性作品或者將會在之後與本論文所提出的技術做比較。

介紹過程中會以圖 3.1 中兩張灰階影像及圖 3.2 中兩張彩色影像作為實驗的展示結果，這四張影像的大小皆為 512×512 ，其來源為資料庫 SIPI[1]的影像。此外，壓縮技術所使用的區塊大小設定為通用的 8×8 及 16×16 ，且展示實驗結果的同時會將影像放大方便觀看及分析，以便了解各技術的特性及差異。

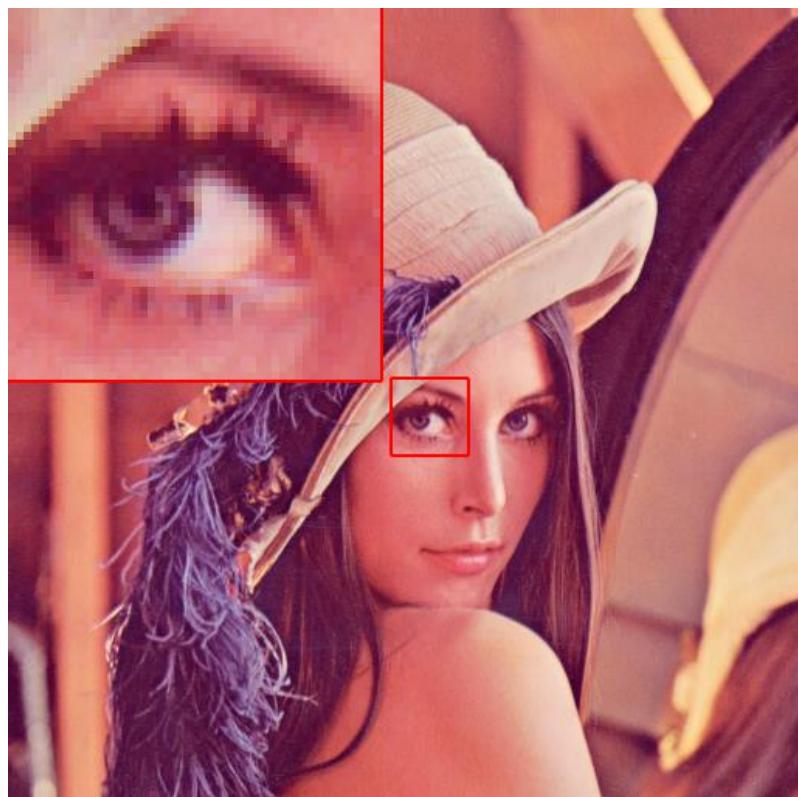


(a) 灰階影像 1

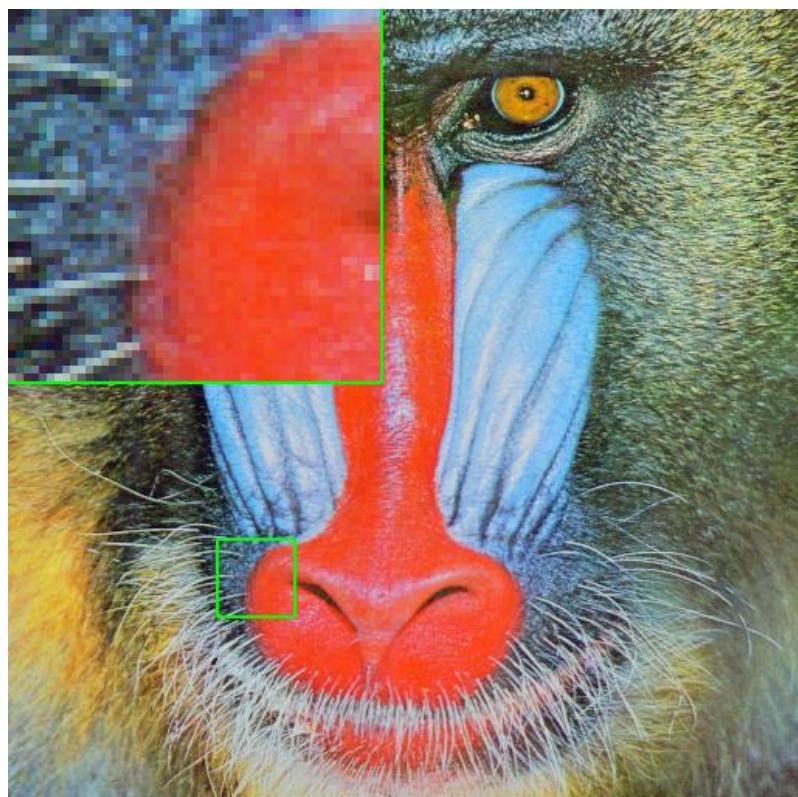


(b) 灰階影像 2

圖 3.1 壓縮測試用之灰階影像



(a) 彩色影像 1



(b) 彩色影像 2

圖 3.2 測試用之彩色影像

3.1 區塊截斷編碼(Block truncation coding, BTC)

在過往許多影像壓縮技術當中，區塊截斷編碼為 Delp 及 Mitchell [18]所提出的一種非常低計算複雜度的編解碼器，且與其他技術相比之下具有不錯的比特率(Bit rate)。概念如前言所述，將一張大小為 $P \times Q$ 的連續色調灰階影像切割成多個大小為 $M \times N$ 的不重疊區塊，每個區塊計算並存取各自的兩個量化色階(Quantization level)及位元圖(Bitmap)就完成壓縮，區塊截斷編碼技術的流程圖如圖 3.3，區塊的處理如下所示：

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.1)$$

$$\bar{x^2} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n)^2, \quad (3.2)$$

$$\sigma = \sqrt{\bar{x^2} - \bar{x}^2}, \quad (3.3)$$

\bar{x} 、 $\bar{x^2}$ 及 σ 分別表示區塊的一階矩量(或稱平均值)、二階矩量及標準差； $x(m, n)$ 表示在區塊內位置 (m, n) 的像素數值。量化色階 a 及 b 可由取得的區塊平均值 \bar{x} 及標準差 σ 去計算，公式如下：

$$a = \bar{x} - \sigma \sqrt{\frac{q}{M \times N - q}}, \quad (3.4)$$

$$b = \bar{x} + \sigma \sqrt{\frac{M \times N - q}{q}}, \quad (3.5)$$

其中， q 表示的是在該區塊中像素數值大於等於平均值 \bar{x} 的個數。值得注意的是，可能會發生分母為 0 的情況只會在 $q = 0$ 時，此情況發生的原因為區塊各個像素的數值都相同，因此當 $q = 0$ 時，量化色階 a 及 b 都設為 \bar{x} 。接著，各區塊將取得的平均值 \bar{x} 作為臨界值(Thresholding)可得出位元圖 h ，定義如下：

$$h(m, n) = \begin{cases} 1, & \text{if } x(m, n) \geq \bar{x} \\ 0, & \text{if } x(m, n) < \bar{x} \end{cases} \quad (3.6)$$

依照上述的演算法直到做完所有區塊就完成區塊截斷編碼對整張影像的壓縮。於解碼端，每個區塊重建的方式是藉由位元圖判斷兩個量化色階的分佈，重建區塊 y 的方法如下：

$$y(m, n) = \begin{cases} b, & \text{if } h(m, n) = 1 \\ a, & \text{if } h(m, n) = 0 \end{cases} \quad (3.7)$$

依照上述的方法直到做完所有區塊就可以重建壓縮後的影像。圖 3.4 及圖 3.5 分別為區塊截斷編碼使用區塊大小 8×8 及 16×16 的結果，雖然區塊越大壓縮率越高，但從實驗結果圖中可以明顯看出越大的區塊也造成越嚴重的區塊效應(Blocking effect)及假邊現象(False contour)。

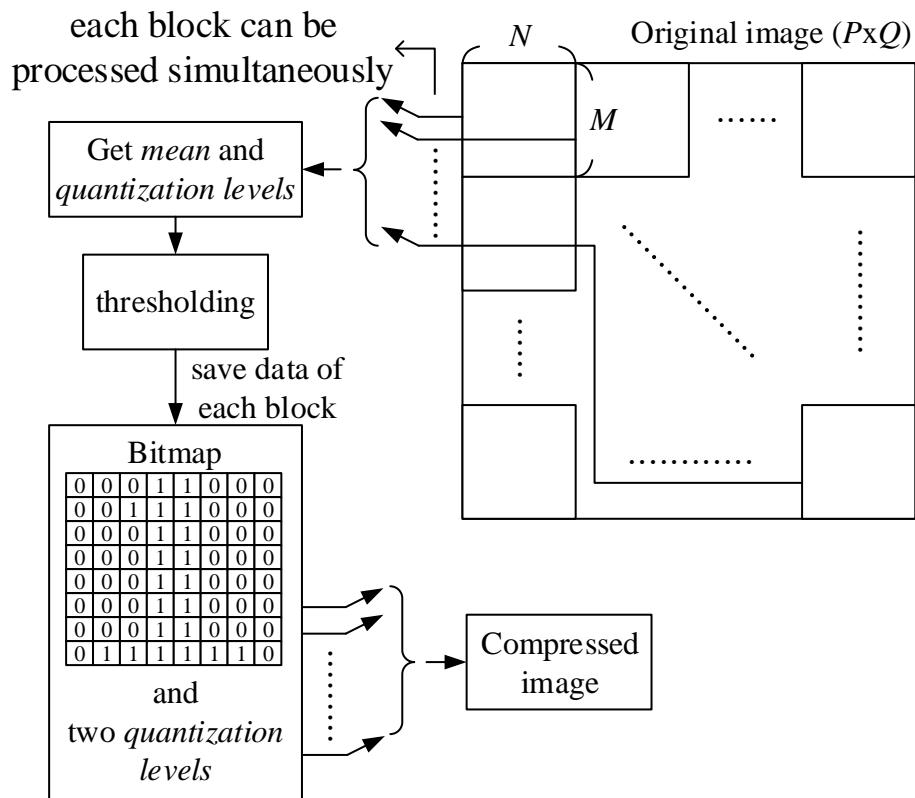
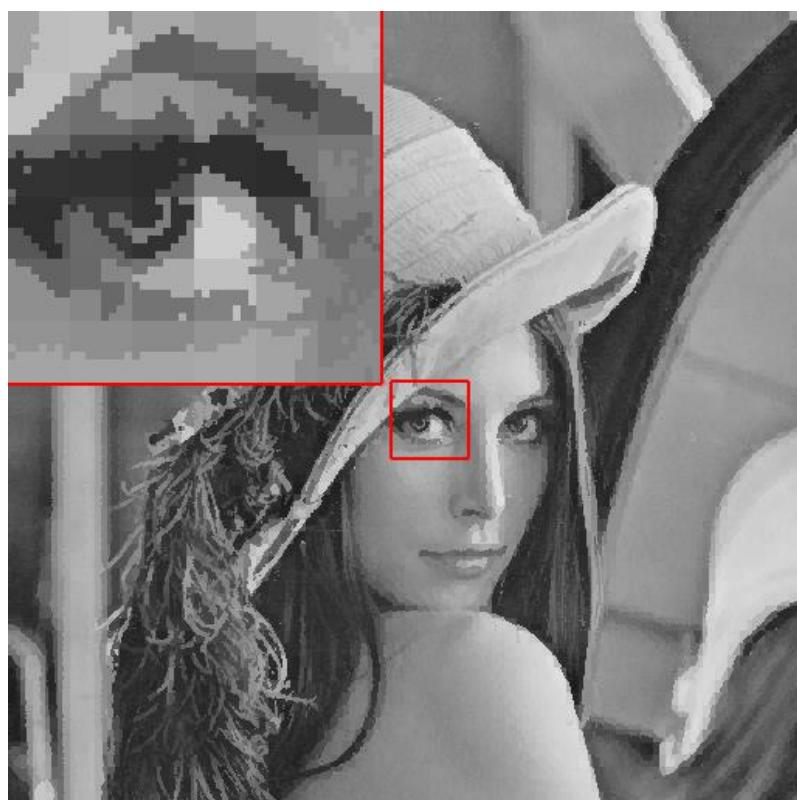
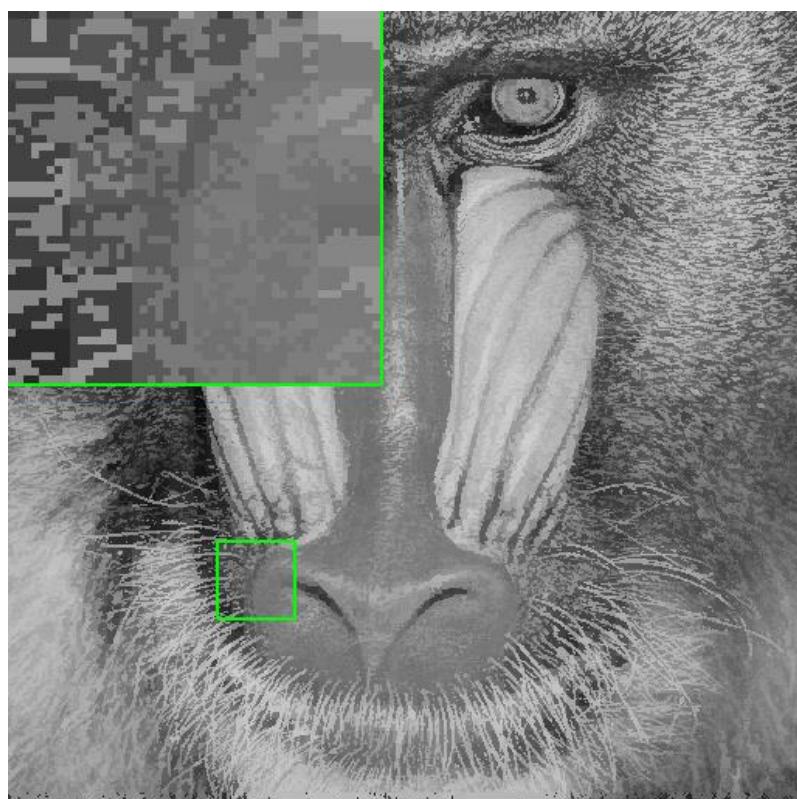


圖 3.3 傳統區塊截斷編碼(BTC)的流程圖

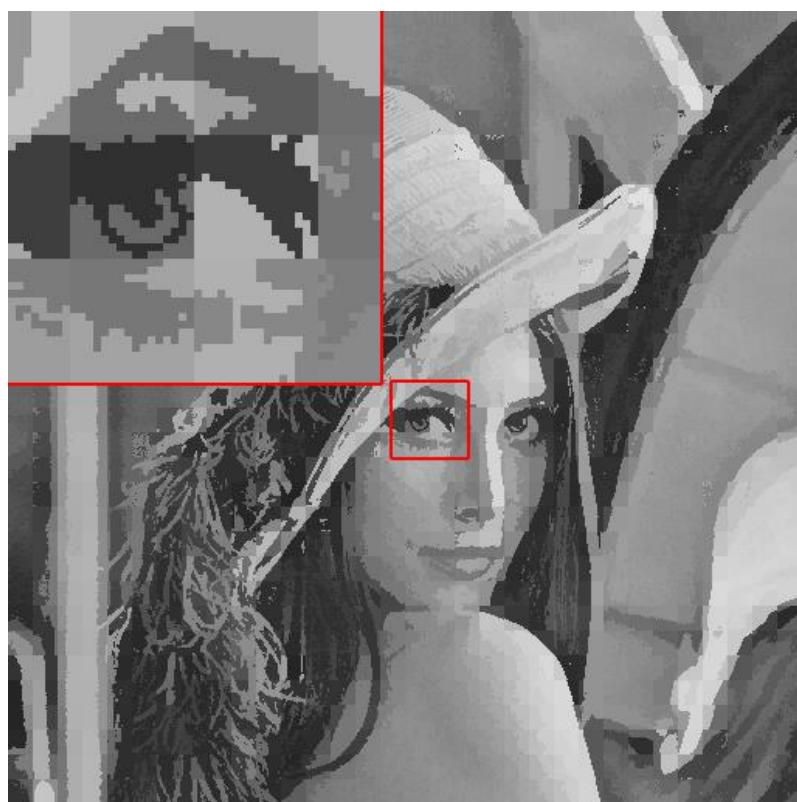


(a) 灰階影像 1 經 BTC 壓縮後重建的影像

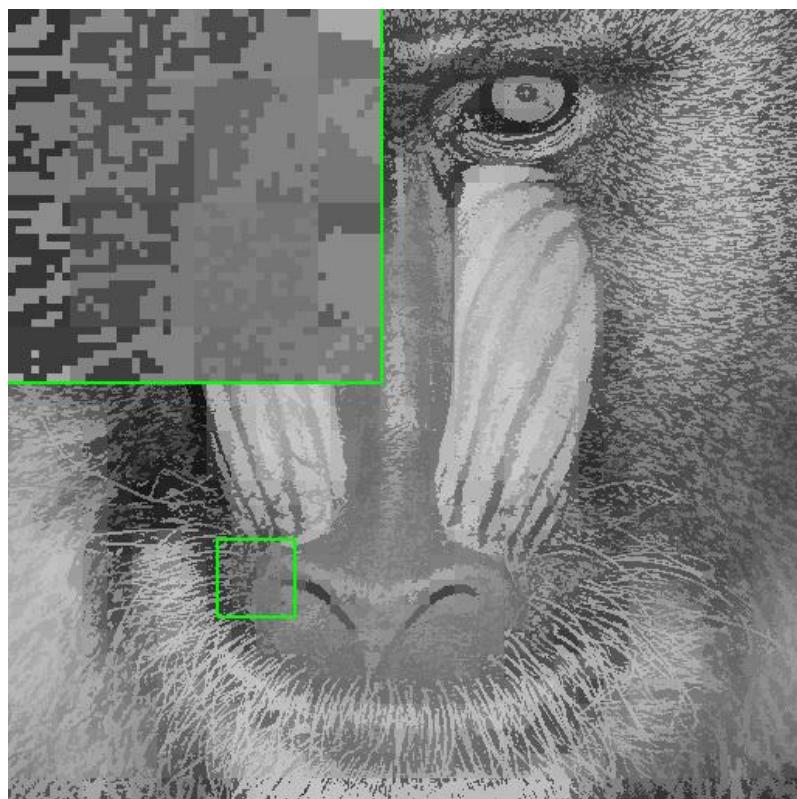


(b) 灰階影像 2 經 BTC 壓縮後重建的影像

圖 3.4 BTC 重建之影像(8×8 區塊大小)



(a) 灰階影像 1 經 BTC 壓縮後重建的影像



(b) 灰階影像 2 經 BTC 壓縮後重建的影像

圖 3.5 BTC 重建之影像(16×16 區塊大小)

3.2 絶對矩量區塊截斷編碼(Absolute moment BTC, AMBTC)

絕對矩量區塊截斷編碼為 Lema 及 Mitchell [25]在 1984 年所提出，目的是要改良傳統區塊截斷編碼技術當中的量化色階。首先，將一張大小為 $P \times Q$ 的連續色調灰階影像切割成多個大小為 $M \times N$ 的不重疊區塊，區塊的處理如下所示：

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.8)$$

\bar{x} 表示區塊的一階矩量(或稱平均值)； $x(m, n)$ 表示在區塊內位置 (m, n) 的像素數值。量化色階 a 及 b 可由取得的區塊平均值 \bar{x} 去計算，公式如下：

$$a = \frac{1}{m-q} \sum_{x(m,n) < \bar{x}} x(m, n), \quad (3.9)$$

$$b = \frac{1}{q} \sum_{x(m,n) \geq \bar{x}} x(m, n), \quad (3.10)$$

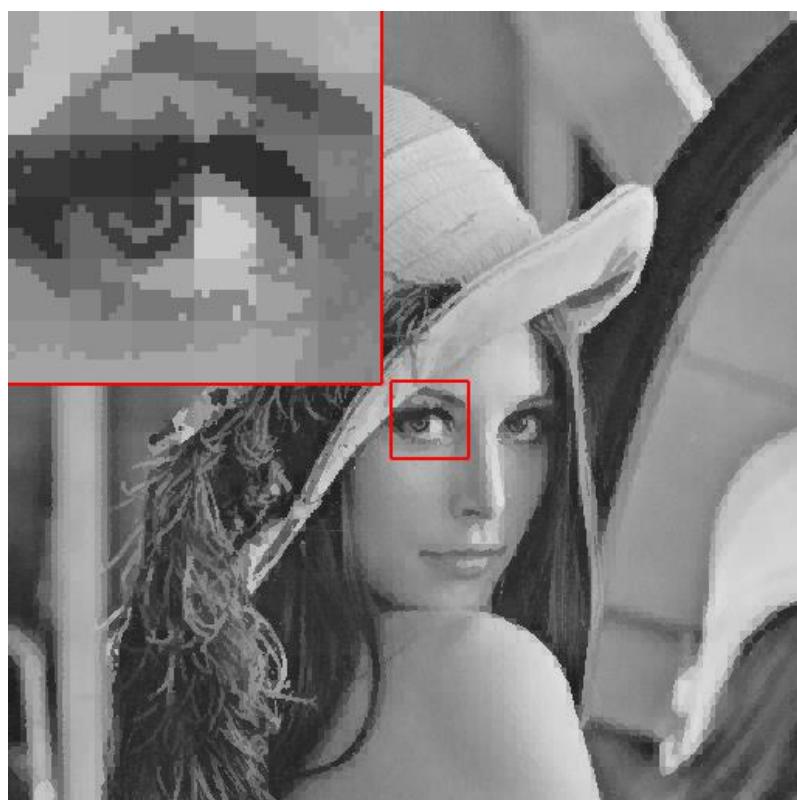
其中， q 表示的是在該區塊中像素數值大於等於平均值 \bar{x} 的個數。值得注意的是，可能會發生分母為 0 的情況只會在 $q = 0$ 時，此情況發生的原因為區塊各個像素的數值都相同，因此當 $q = 0$ 時，量化色階 a 及 b 都設為 \bar{x} 。接著，各區塊將取得的平均值 \bar{x} 作為臨界值(Thresholding)可得出位元圖 h ，定義如下：

$$h(m, n) = \begin{cases} 1, & \text{if } x(m, n) \geq \bar{x} \\ 0, & \text{if } x(m, n) < \bar{x} \end{cases} \quad (3.11)$$

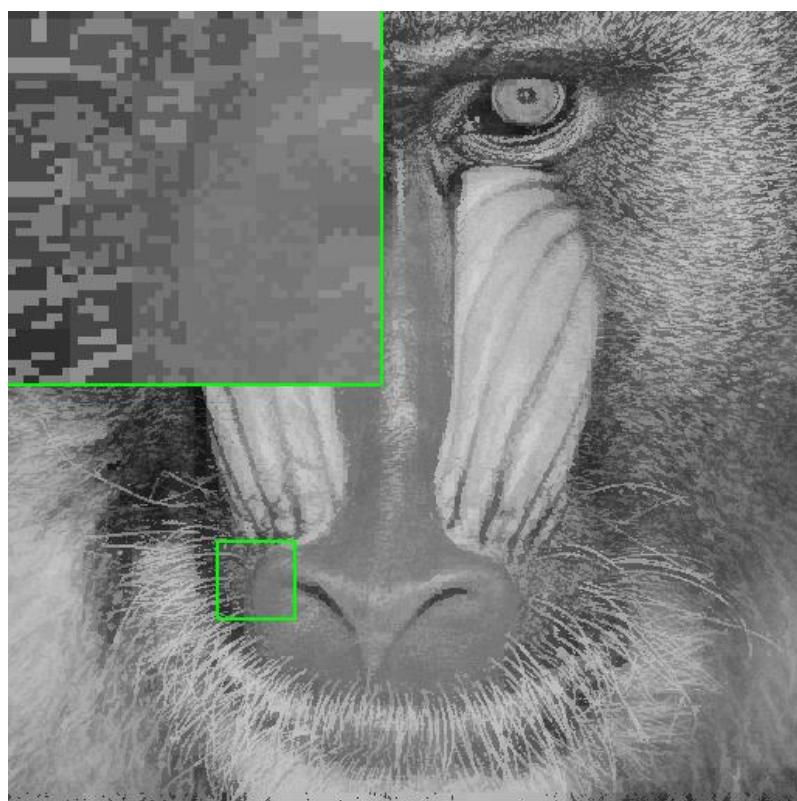
依照上述的演算法直到做完所有區塊就完成區塊截斷編碼對整張影像的壓縮。於解碼端，每個區塊重建的方式是藉由位元圖判斷兩個量化色階的分佈，重建區塊 y 的方法如下：

$$y(m, n) = \begin{cases} b, & \text{if } h(m, n) = 1 \\ a, & \text{if } h(m, n) = 0 \end{cases} \quad (3.12)$$

依照上述的方法直到做完所有區塊就可以重建壓縮後的影像。圖 3.6 及 3.7 分別為絕對矩量區塊截斷編碼使用區塊大小 8×8 及 16×16 的結果，如同傳統的區塊截斷編碼技術，使用越大的區塊會導致越嚴重的區塊效應(Blocking effect)及假邊現象(False contour)。比起傳統的區塊截斷編碼技術，雖然絕對矩量區塊截斷編碼的計算複雜度更低，但也使得結果變得更加模糊。

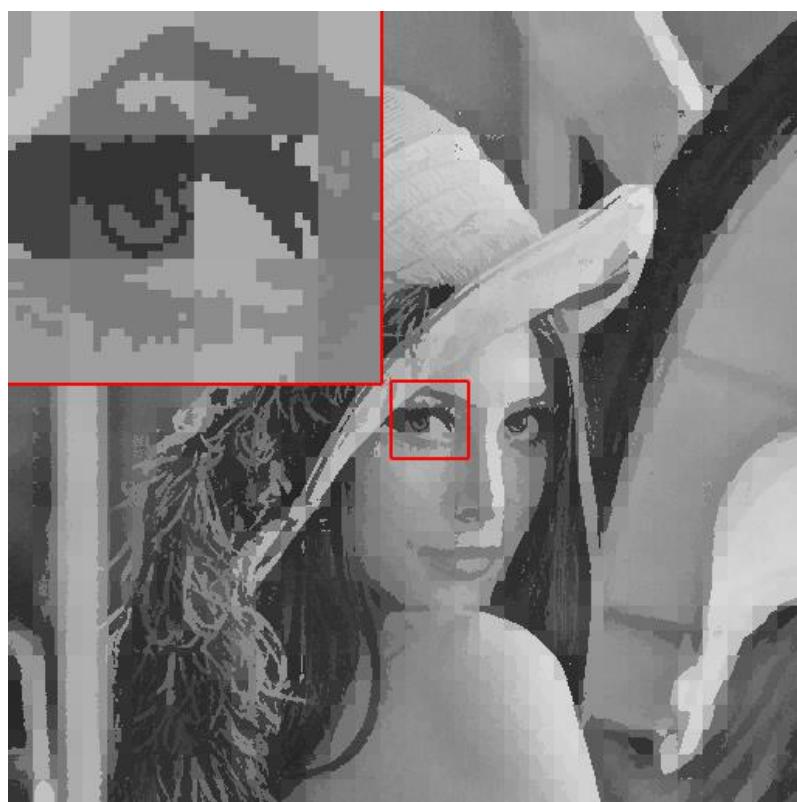


(a) 灰階影像 1 經 AMBTC 壓縮後重建的影像

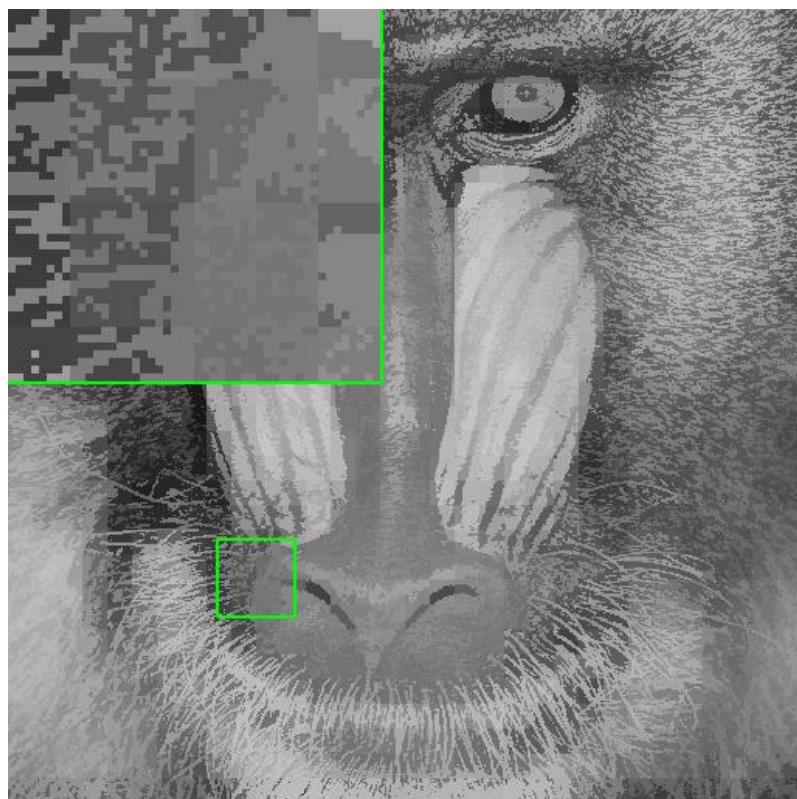


(b) 灰階影像 2 經 AMBTC 壓縮後重建的影像

圖 3.6 AMBTC 重建之影像(8×8 區塊大小)



(a) 灰階影像 1 經 AMBTC 壓縮後重建的影像



(b) 灰階影像 2 經 AMBTC 壓縮後重建的影像

圖 3.7 AMBTC 重建之影像(16×16 區塊大小)

3.3 誤差擴散區塊截斷編碼(Error-diffused BTC, EDBTC)

誤差擴散區塊截斷編碼為 Guo [27]在 2008 年所提出，與前面 2.1 節的誤差擴散法(Error-diffused, ED)結構相似，只是將半色調技術用於各個區塊上，其最大的特色就是透過誤差擴散法提升各像素之間的關聯性並藉此改善影像的品質。其處理步驟如同傳統區塊截斷編碼技術，區塊的處理如下所示：

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.13)$$

$$a = \min(B), \quad (3.14)$$

$$b = \max(B), \quad (3.15)$$

\bar{x} 表示區塊的一階矩量(或稱平均值)； $x(m, n)$ 表示在區塊內位置 (m, n) 的像素數值； $\min(\cdot)$ 、 $\max(\cdot)$ 分別表示區域內的最小值及最大值； B 表示區塊。得到區塊的量化色階後，藉由平均值 \bar{x} 作為臨界值(Thresholding)可得出位元圖 h ，定義如下：

$$c(m, n) = x(m, n) + s(m, n), \text{ where } s(m, n) = \sum_{(p,q) \in R} \sum \frac{e(m+p, n+q) \times w(p, q)}{\text{sum}}, \quad (3.16)$$

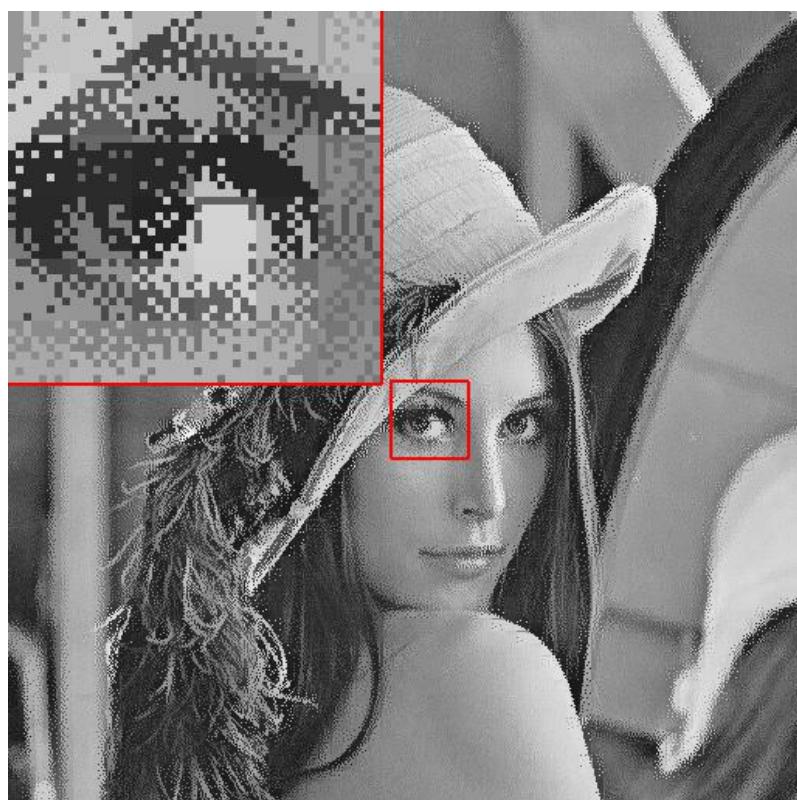
$$e(m, n) = c(m, n) - y(m, n), \quad (3.17)$$

$$y(m, n) = \begin{cases} b, & \text{if } c(m, n) \geq \bar{x} \\ a, & \text{if } c(m, n) < \bar{x} \end{cases} \quad (3.18)$$

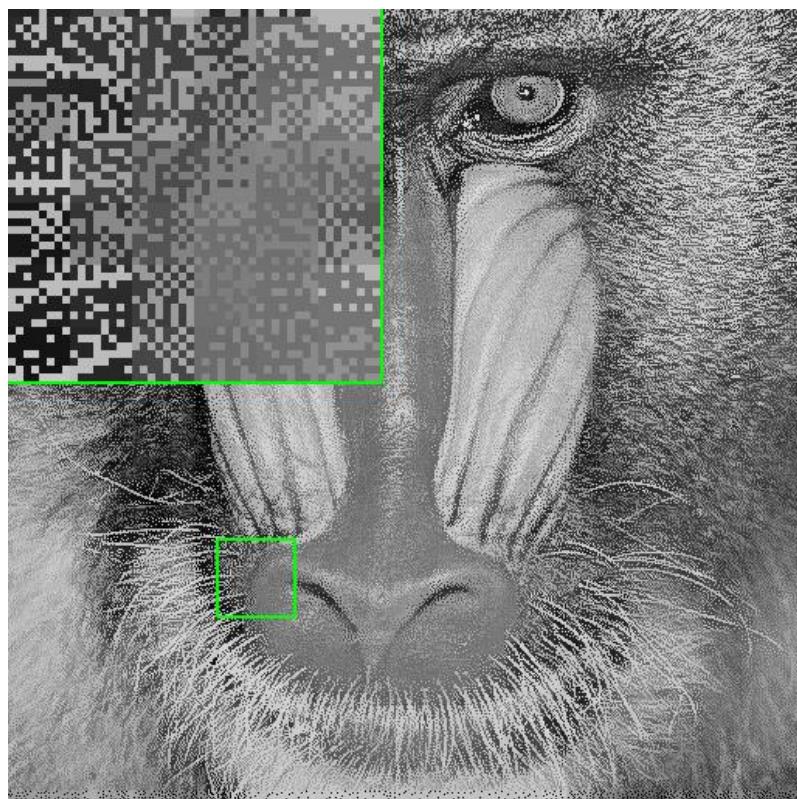
$x(m, n)$ 定義如上； $c(m, n)$ 為經過誤差擴散補償過後的灰階值； $e(m, n)$ 為輸出與當下補償後灰階的差值； $w(p, q)$ 表示在誤差擴散矩陣的範圍 R 內的誤差擴散權重，擴散權重矩陣的種類如表 2.2； $y(m, n)$ 為輸出的壓縮影像。在此特別說明，在處理座標位置 $(0,0)$ 時，並沒有受到任何誤差擴散的補償，因此此時的 $c(0,0)$ 等於 $x(0,0)$ 。值得注意的是，由於誤差擴散時可能會超出影像的邊界，因此誤差的權重需要視能擴散的範圍正規化，

$$\text{sum} = \sum_{(p,q) \in R} \begin{cases} w(p, q), & \text{if } (i + p, j + q) \text{ within range} \\ 0, & \text{if } (i + p, j + q) \text{ out of range} \end{cases} \quad (3.19)$$

各區塊重複上述方法就能得到壓縮影像。雖然此方法能夠將量化色階的誤差補償至周圍像素，降低區塊效應及假邊現象的產生，但犧牲了平行處理的優勢並導致脈衝雜訊(Impulse noise)的產生。使用表 2.2(a)誤差擴散權重的實驗結果如圖 3.8 及圖 3.9。

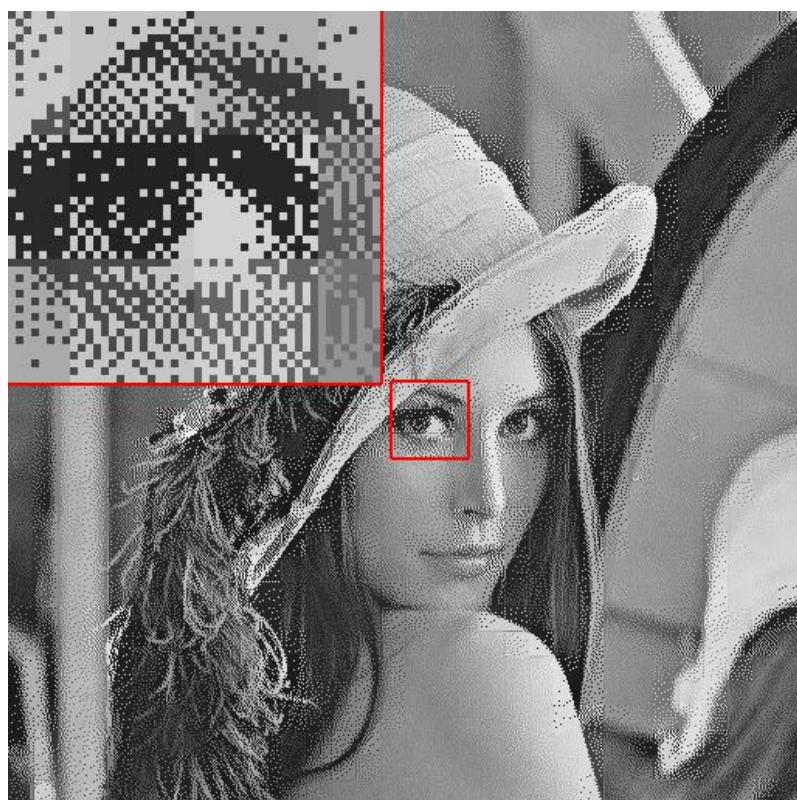


(a) 灰階影像 1 經 EDBTC 壓縮後重建的影像

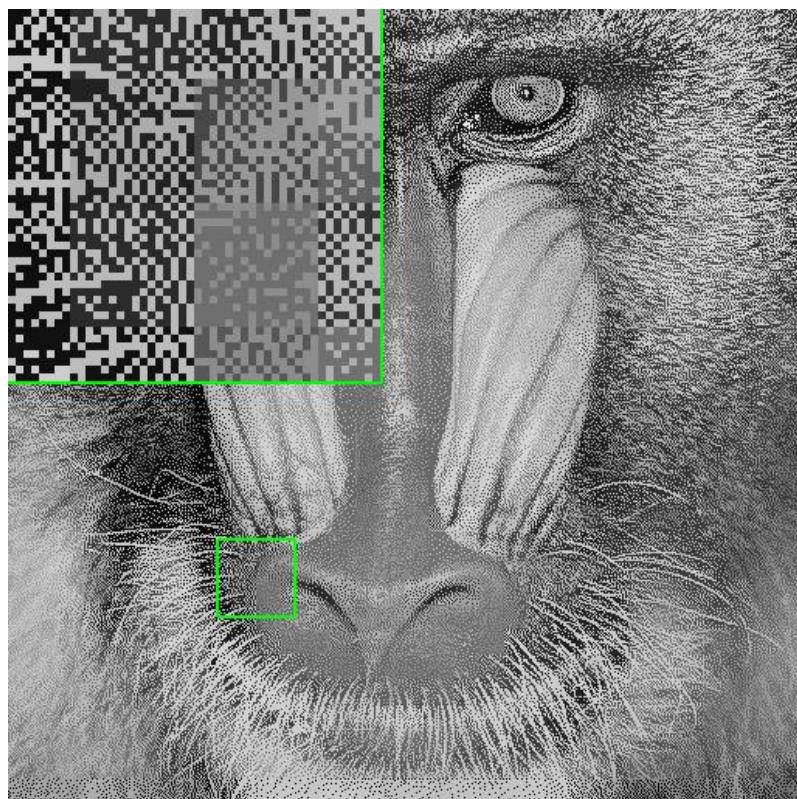


(b) 灰階影像 2 經 EDBTC 壓縮後重建的影像

圖 3.8 EDBTC 使用 Floyd-Steinberg 的誤差擴散權重重建之影像(8 × 8區塊大小)



(a) 灰階影像 1 經 EDBTC 壓縮後重建的影像



(b) 灰階影像 2 經 EDBTC 壓縮後重建的影像

圖 3.9 EDBTC 使用 Floyd-Steinberg 的誤差擴散權重重建之影像(16×16 區塊大小)

3.4 有序抖動區塊截斷編碼(Ordered dither BTC, ODBTC)

有序抖動區塊截斷編碼為 Guo 及 Wu [28]在 2009 年所提出，結合前面 2.2 節所介紹的有序抖動法(Ordered dither, OD)及區塊截斷編碼技術的產物，主要目的是改善誤差擴散區塊截斷編碼(EDBTC)不能平行處理的問題。其區塊首先要取得的參數如同誤差擴散區塊截斷編碼技術，區塊的平均值及量化色階如下所示：

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.20)$$

$$a = \min(B), \quad (3.21)$$

$$b = \max(B), \quad (3.22)$$

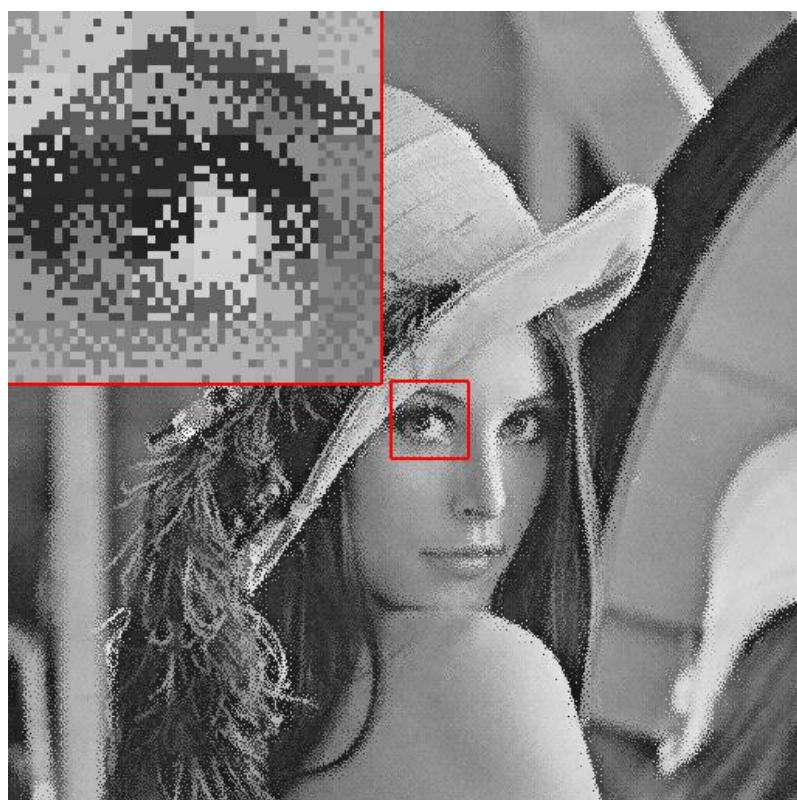
當中的定義如同公式(3.13)-(3.15)。接著在每一個不重疊區塊中，以點對點的方式，將原始灰階影像的像素點比對有序抖動矩陣中的數值。此時需要特別注意，由於區塊截斷編碼是以區塊的方式進行壓縮，代表其參考範圍為該區塊而非整張影像，故有序抖動矩陣數值必須隨著不同的區塊而進行正規化調整(Normalize)，調整方式如下：

$$d'(m, n) = (b - a) \times \frac{(d(m, n) - \min(d))}{\max(d) - \min(d)}, \quad (3.23)$$

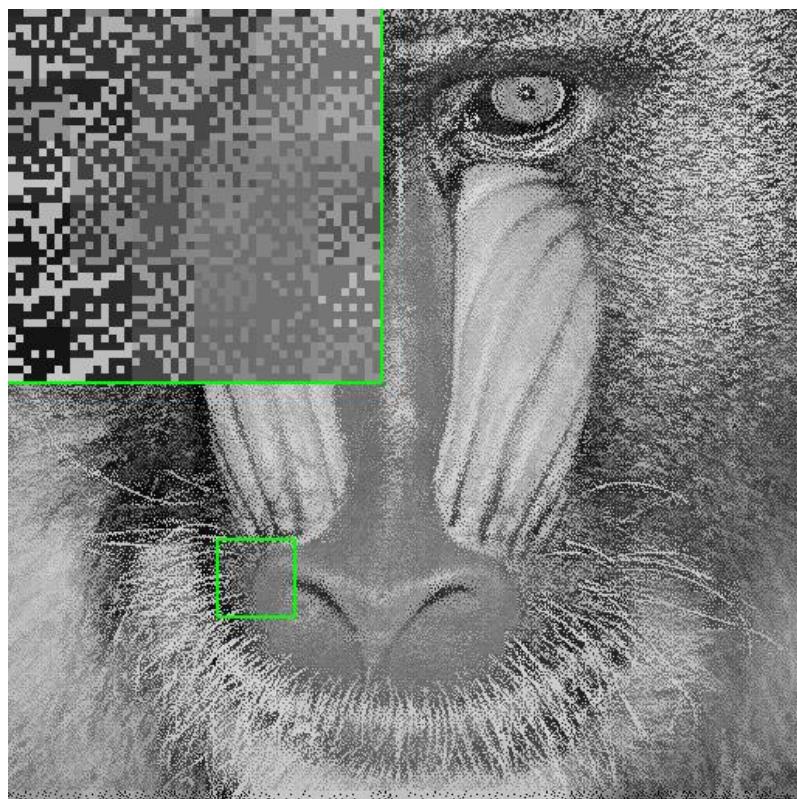
其中， $d(m, n)$ 為原始的有序抖動矩陣，如表 2.5； $\max(d)$ 與 $\min(d)$ 為原始有序抖動矩陣中的最大值與最小值。藉此公式便可針對不同區塊的最大值與最小值，調整計算出相對應的有序抖動矩陣 $d'(m, n)$ 。最後藉由不同區塊的 $d'(m, n)$ 來決定各區塊中的量化色階及臨界值，公式如下：

$$y(m, n) = \begin{cases} b, & \text{if } x(m, n) \geq d'(m, n) + a \\ a, & \text{if } x(m, n) < d'(m, n) + a \end{cases}, \quad (3.24)$$

此時需特別說明，經過正規化後的有序抖動矩陣 $d'(m, n)$ ，其數值範圍為 0 到 $(b - a)$ ，在使用門檻值計算時需再加上 a 使其範圍變成 a 到 b 。重複上述流程，直到完成整張影像為止，即可得到一張有序抖動區塊截斷編碼影像。藉由 Classical-4 的有序抖動矩陣所重建的影像會有色階聚集成塊狀，而使用 Bayer-5 重建的影像其輸出色階擴散成點狀。後者因為雜訊點較小容易被人眼忽略，因此有較好的影像品質，使用 Bayer-5 重建的影像其結果如圖 3.10 及圖 3.11 所示。

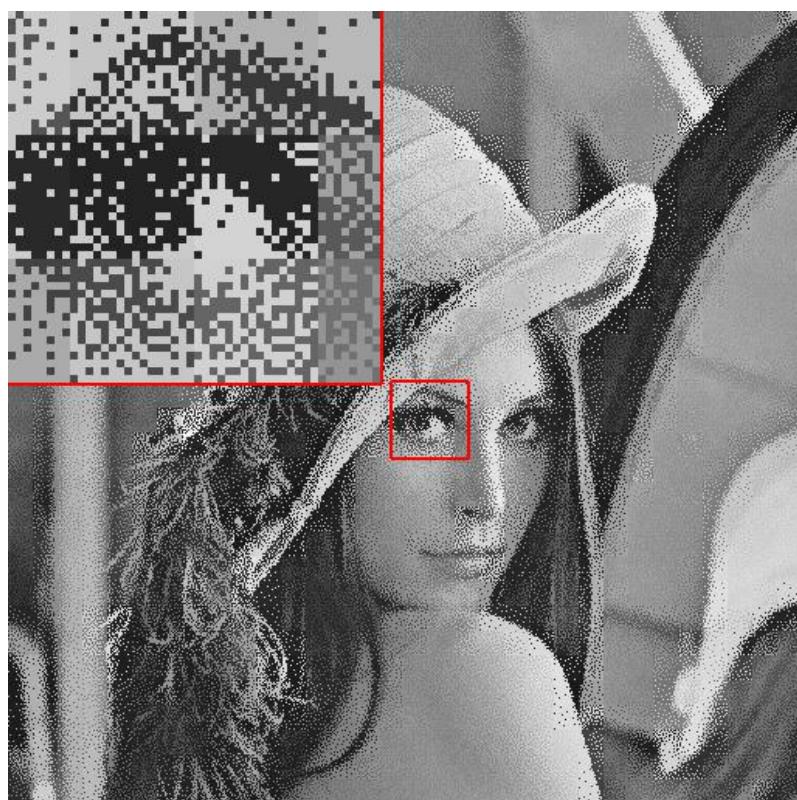


(a) 灰階影像 1 經 ODBTC 壓縮後重建的影像

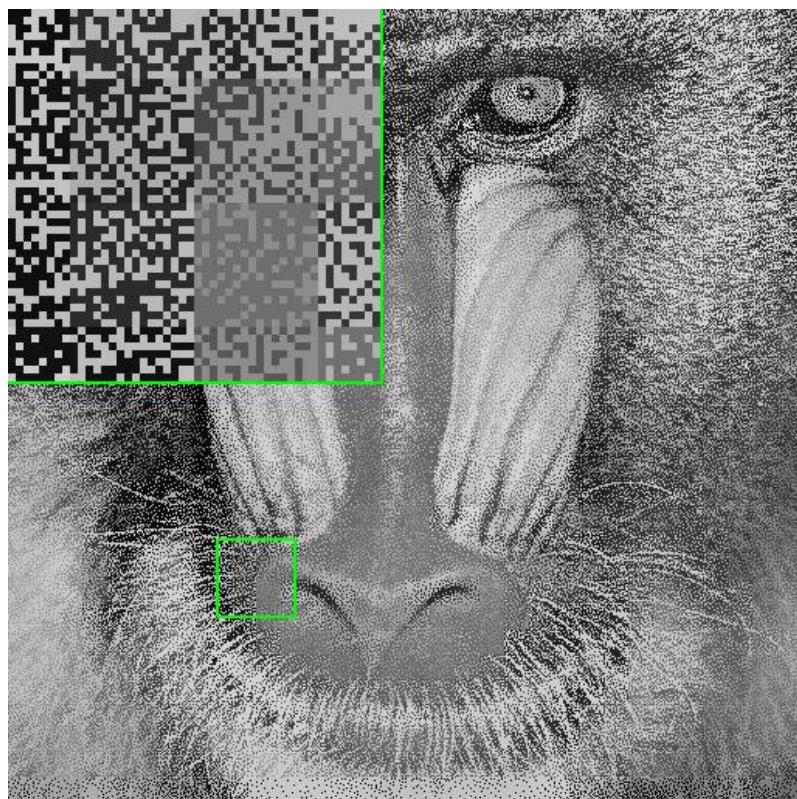


(b) 灰階影像 2 經 ODBTC 壓縮後重建的影像

圖 3.10 ODBTC 使用 Bayer-5 的有序抖動矩陣重建之影像(8×8 區塊大小)



(a) 灰階影像 1 經 ODBTC 壓縮後重建的影像



(b) 灰階影像 2 經 ODBTC 壓縮後重建的影像

圖 3.11 ODBTC 使用 Bayer-5 的有序抖動矩陣重建之影像(16×16 區塊大小)

3.5 點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)

點擴散區塊截斷編碼與 2.3 節的點擴散法(Dot-diffused, DD)概念相似，為 Guo 及 Liu [29]在 2014 年所提出，此技術有著誤差擴散法(ED)優異的影像品質及有序抖動法(OD)平行處理的特性。值得注意的是，此方法將使用到類別矩陣(Class matrix, CM)及擴散矩陣(Diffused matrix, DM)，這兩個矩陣的數值分別表示的是區塊的處理順序以及點擴散的權重。點擴散區塊截斷編碼的處理方式類似誤差擴散區塊截斷編碼技術(EDBTC)，各區塊首先要取得的參數如下：

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.25)$$

$$a = \min(B), \quad (3.26)$$

$$b = \max(B), \quad (3.27)$$

當中的定義如同公式(3.13)-(3.15)。接著，各區塊的像素依照類別矩陣當中的數值 $o(m, n)$ 由小到大去對其所對應到的位置做處理，處理公式如下：

$$c(m, n) = x(m, n) + s(m, n), \text{ where } s(m, n) = \sum_{(p,q) \in R} \sum \frac{e(m+p, n+q) \times w(p, q)}{\text{sum}}, \quad (3.28)$$

$$e(m, n) = c(m, n) - y(m, n), \quad (3.29)$$

$$y(m, n) = \begin{cases} b, & \text{if } c(m, n) \geq \bar{x} \\ a, & \text{if } c(m, n) < \bar{x} \end{cases} \quad (3.30)$$

當中的定義如同公式(3.16)-(3.18)。在此特別說明，當 $o(m, n) = 0$ 時，並無受到任何誤差擴散的補償，因此此時的 $c(m, n) = x(m, n)$ 。由於擴散矩陣的大小固定為 3×3 ，因此 $p, q \in [-1, 1]$ 。值得注意的是，如同誤差擴散區塊截斷編碼(EDBTC)時要考量可能會超出影像的邊界，此外還要考慮到周圍像素是否已處理完畢，詳細公式如下：

$$\text{sum} = \sum_{(p,q) \in R} \begin{cases} w(p, q), & \text{if } (i + p, j + q) \text{ within range and } o(m + p, n + q) > o(m, n) \\ 0, & \text{if } (i + p, j + q) \text{ out of range or } o(m + p, n + q) \leq o(m, n) \end{cases}, \quad (3.31)$$

當擴散目標對應的類別矩陣其數值小於或等於當前像素對應的類別矩陣其數值時，這表示擴散目標的像素已經處理過，因此不該將這個像素作為擴散的目標。目前最佳的類別矩陣及擴散矩陣如表 3.1 所示為文獻[29]所提出，以此所產生的點擴散區塊截斷編碼結果如圖 3.12 及圖 3.13 所示。

表 3.1 Guo 與 Liu [29]提出之類別矩陣與擴散權重

42	47	46	45	16	13	11	2
61	57	53	8	27	22	9	50
63	58	0	15	26	31	40	30
10	4	17	21	3	44	18	6
14	24	25	7	5	48	52	39
20	28	23	32	38	51	54	60
19	33	36	37	49	43	56	55
12	62	29	35	1	59	41	34

(a) 8×8 類別矩陣

0.27163	1	0.27163
1	x	1
0.27163	1	0.27163

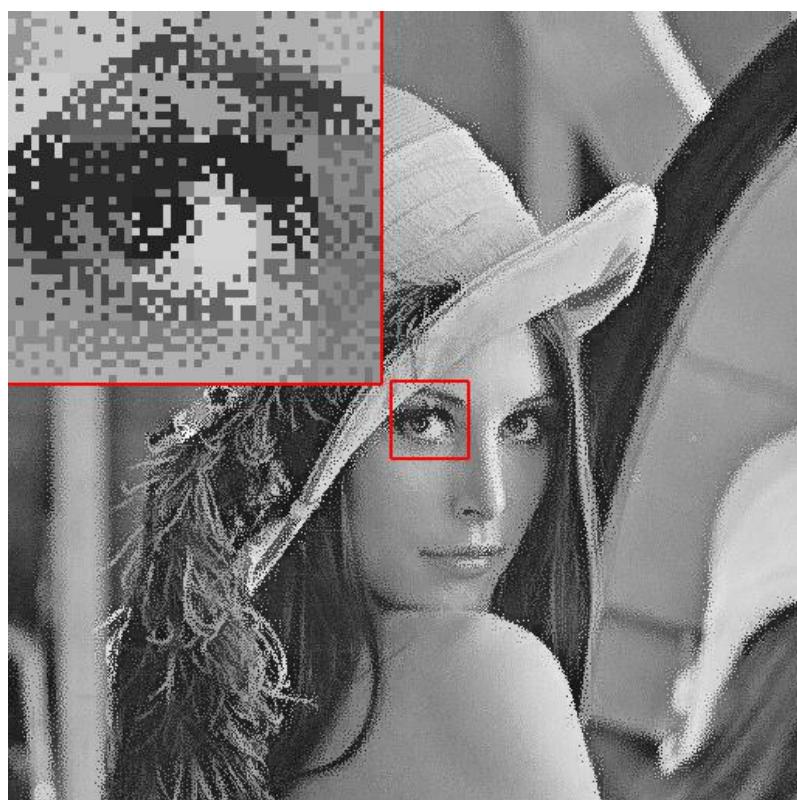
(b) 使用於 8×8 類別矩陣之擴散矩陣

0.305032	1	0.305032
1	x	1
0.305032	1	0.305032

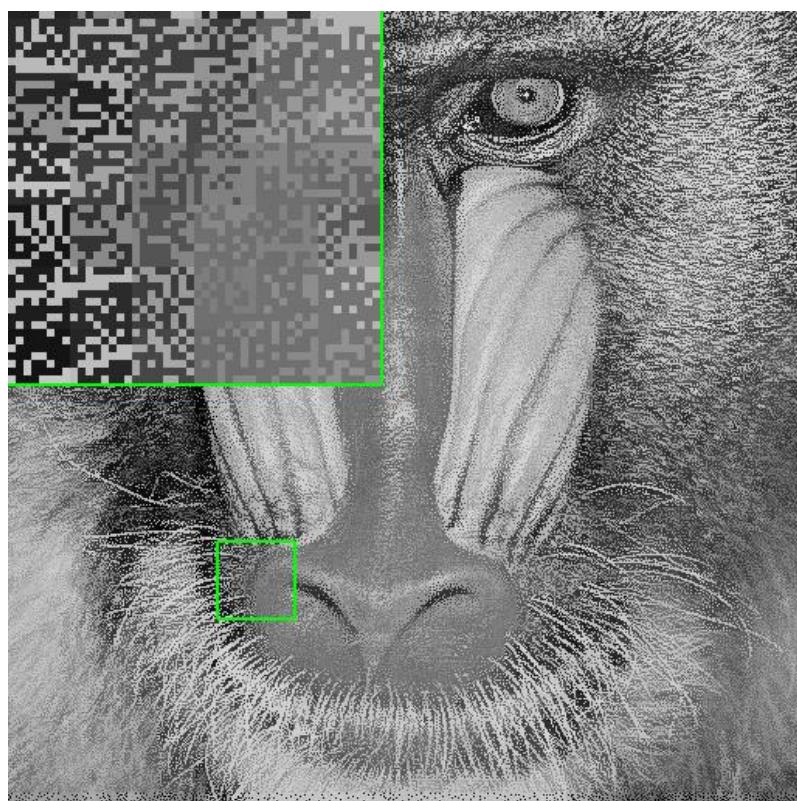
(c) 使用於 16×16 類別矩陣之擴散矩陣

6	7	20	10	53	55	66	87	137	142	143	144	172	122	175	164
3	9	23	50	60	51	65	74	130	145	138	148	179	180	214	221
0	14	24	37	67	79	96	116	39	149	162	198	12	146	224	1
15	26	43	28	71	54	128	112	78	159	177	201	208	223	225	242
22	4	48	32	94	98	80	135	157	173	113	182	222	226	227	16
40	85	72	83	104	117	163	133	168	184	200	219	244	237	183	21
47	120	101	105	123	132	170	176	190	202	220	230	245	235	17	41
76	73	127	109	97	134	178	181	206	196	229	231	246	19	42	49
103	99	131	147	169	171	166	203	218	232	243	248	247	33	52	68
108	107	140	102	185	167	204	217	233	106	249	255	44	45	70	69
110	141	88	75	192	205	195	234	241	250	254	38	46	77	5	100
111	158	160	174	119	215	207	240	251	252	253	61	62	93	84	125
151	136	189	199	197	216	236	239	25	31	56	82	92	95	124	114
156	188	191	209	213	228	238	29	36	59	64	91	118	139	115	155
187	194	165	212	2	13	30	35	58	63	90	86	152	129	154	161
193	210	211	8	11	27	34	57	18	89	81	121	126	153	150	186

(d) 16×16 類別矩陣

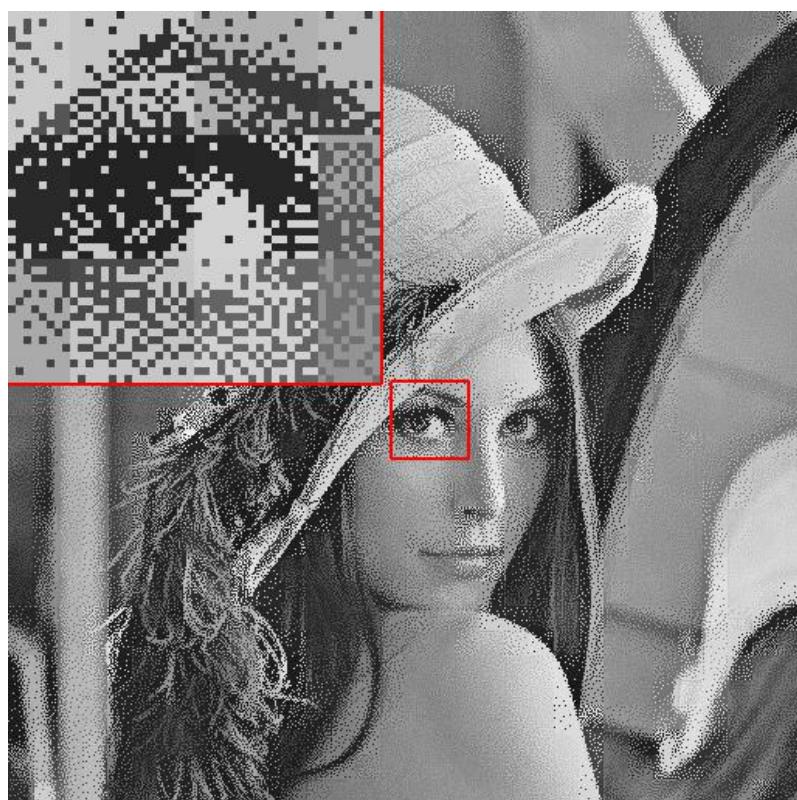


(a) 灰階影像 1 經 DDBTC 壓縮後重建的影像

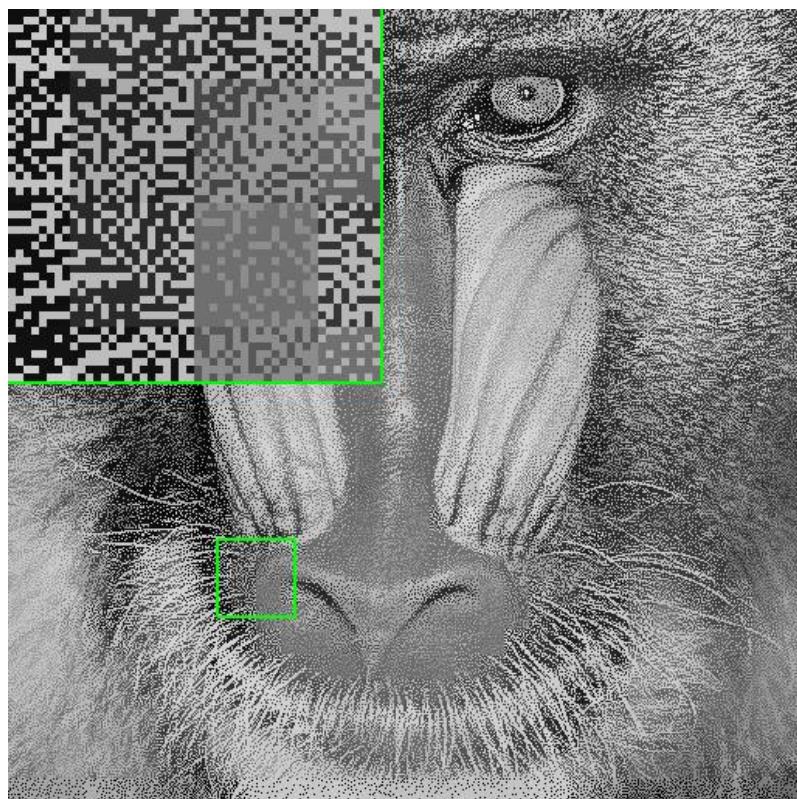


(b) 灰階影像 2 經 DDBTC 壓縮後重建的影像

圖 3.12 DDBTC 使用 Guo 及 Liu 所提出之矩陣重建之影像(8×8 區塊大小)



(a) 灰階影像 1 經 DDBTC 壓縮後重建的影像



(b) 灰階影像 2 經 DDBTC 壓縮後重建的影像

圖 3.13 DDBTC 使用 Guo 及 Liu 所提出之矩陣重建之影像(16×16 區塊大小)

3.6 彩色絕對矩量區塊截斷編碼(Color absolute moment BTC, CAMBTC)

如同前言所介紹，若直接對彩色影像的各通道分別壓縮，則其壓縮率將會如同對灰階影像壓縮時的壓縮率。為了提升壓縮率，文獻[25]將 RGB 色彩模型轉換成 YIQ 色彩模型並對 I、Q 兩通道取樣，接著再使用 3.2 節所介紹的絕對矩量區塊截斷編碼(Absolute moment BTC, AMBTC)分別對取樣後的 YI'Q' 通道做壓縮。其中，YIQ 為電視系統標準(National television standards committee, NTSC)，Y 表示的是提供黑白及彩色電視的亮度信號(Luminance)，又稱亮度(Brightness)；I 表示同相(In-phase)，色彩從橙色到青色；Q 表示正交(Quadrature-phase)，色彩從紫色到黃綠色。轉換公式如下：

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (3.32)$$

其中， $R, G, B \in [0,1]$ ； $Y \in [0,1], I \in [-0.596, 0.596], Q \in [-0.523, 0.523]$ 。將 RGB 色彩模型轉換成 YIQ 色彩模型後，I 及 Q 通道取樣的方式分別是取 2×2 及 4×4 範圍的灰階平均值，若原始影像大小為 $P \times Q$ ，則取樣後的 I' 及 Q' 影像大小分別為 $\frac{P}{2} \times \frac{Q}{2}$ 及 $\frac{P}{4} \times \frac{Q}{4}$ 。接著用絕對矩量區塊截斷編碼分別壓縮 YI'Q' 三通道，就可完成壓縮。

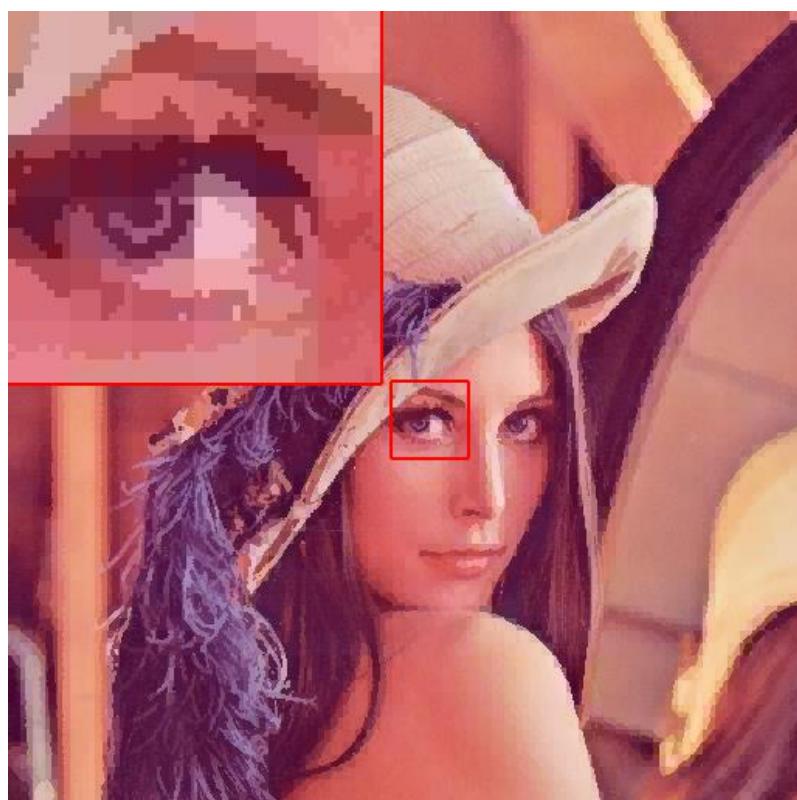
解碼時，由於取樣的關係，I、Q 通道需要經過雙線性內差法(Bilinear interpolation)還原成原始影像大小。然後再經由下方轉換公式從 YIQ 轉換回 RGB：

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.621 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.106 & 1.703 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix}, \quad (3.33)$$

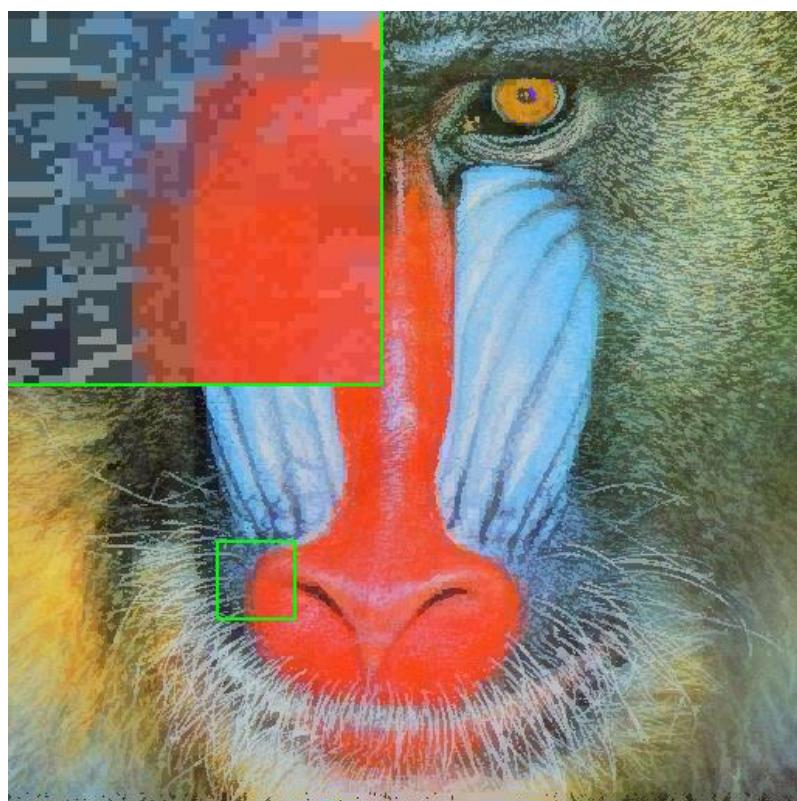
依照上述方法就可得到重建的壓縮影像，圖 3.14 及圖 3.15 分別為彩色絕對矩量區塊截斷編碼使用區塊大小 8×8 及 16×16 的結果，其保留了絕對矩量區塊截斷編碼的優點及缺點，雖然有著極低的計算複雜度，但仍然會產生區塊效應(Blocking effect)及假邊現象(False contour)。不過因為取樣，壓縮率有不錯的提升，壓縮率的公式如下：

$$CR = \frac{3 \times M \times N \times 8}{M \times N + \frac{M}{2} \times \frac{N}{2} + \frac{M}{4} \times \frac{N}{4} + 3 \times 2 \times 8}, \quad (3.34)$$

CR 表示壓縮率(Compression ratio, CR)； $M \times N$ 表示區塊大小。若將區塊大小代入 8×8 及 16×16 ，可得知彩色矩量區塊截斷編碼的壓縮率分別為 11.64 及 16.00。

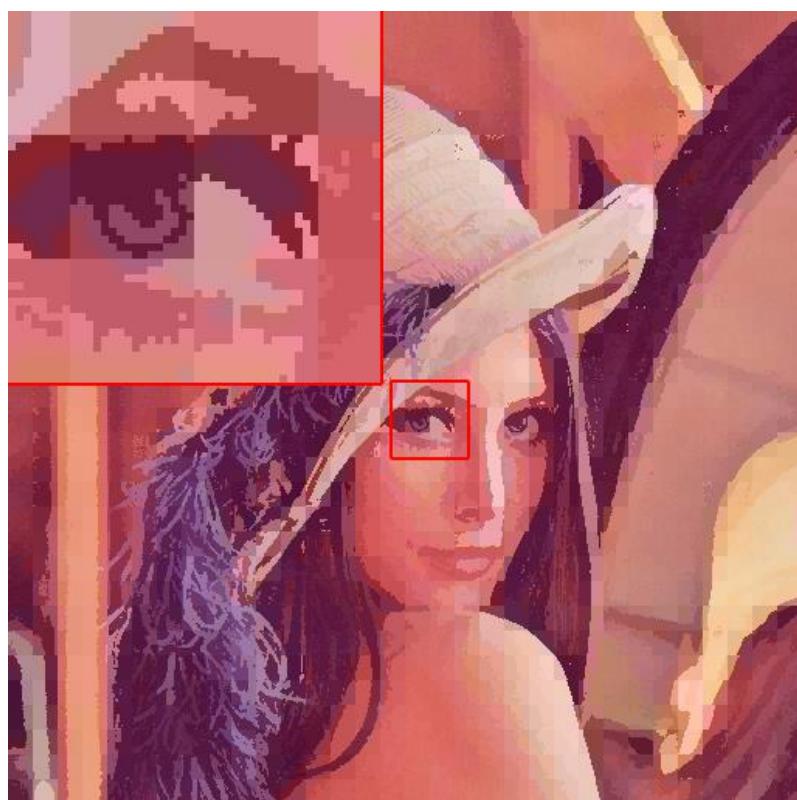


(a) 彩色影像 1 經 CAMBTC 壓縮後重建的影像

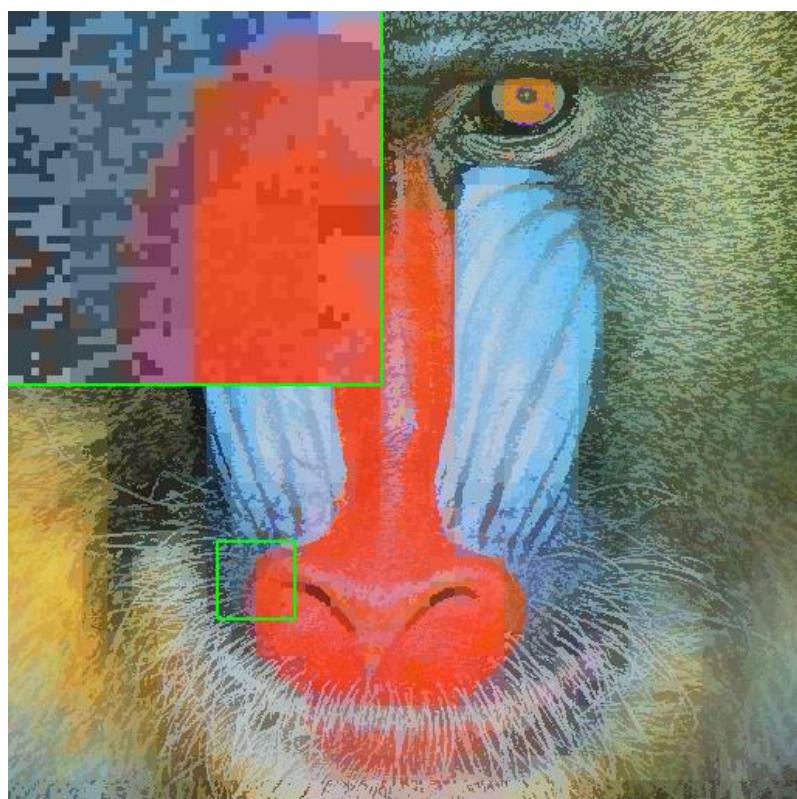


(b) 彩色影像 2 經 CAMBTC 壓縮後重建的影像

圖 3.14 CAMBTC 重建之影像(8×8 區塊大小)， $CR = 11.64$



(a) 彩色影像 1 經 CAMBTC 壓縮後重建的影像



(b) 彩色影像 2 經 CAMBTC 壓縮後重建的影像

圖 3.15 CAMBTC 重建之影像(16×16 區塊大小), $CR = 16.00$

3.7 單一位元圖區塊截斷編碼(Single bit-map BTC, SBMBTC)

如前言所述，色彩模型 RGB 三通道之間的關聯性高，因此三通道可以使用同一張位元圖(Bitmap)重建。藉此特性，Wu 及 Coll [26]在 1992 年提出單一位元圖區塊截斷編碼技術，單一位元圖 h_{SBM} 的產生有兩種方式。首先介紹絕對矩量區塊截斷編碼(AMBTC)，其量化色階的公式(3.9)及(3.10)可以轉換成以下公式：

$$a = \frac{1}{m-q} \sum_{h(m,n)=0} x(m, n), \quad (3.35)$$

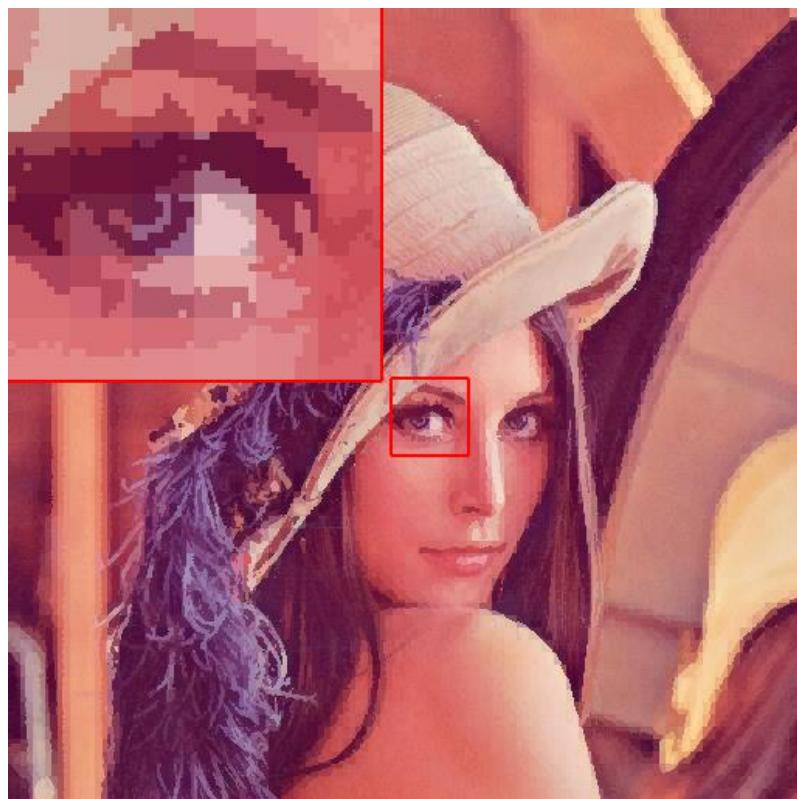
$$b = \frac{1}{q} \sum_{h(m,n)=1} x(m, n), \quad (3.36)$$

原本由平均值判斷的部分可以改成由位元圖 $h(m, n)$ 判斷。接著將說明單一位元產生的第一種方法，首先是將彩色影像轉換成灰階影像，用灰階影像產生的位元圖 h_L 分別在 RGB 三通道中代入上述公式，就可得到三通道的量化色階($a_R, b_R, a_G, b_G, a_B, b_B$)並完成壓縮。彩色影像轉換成灰階影像的公式如下：

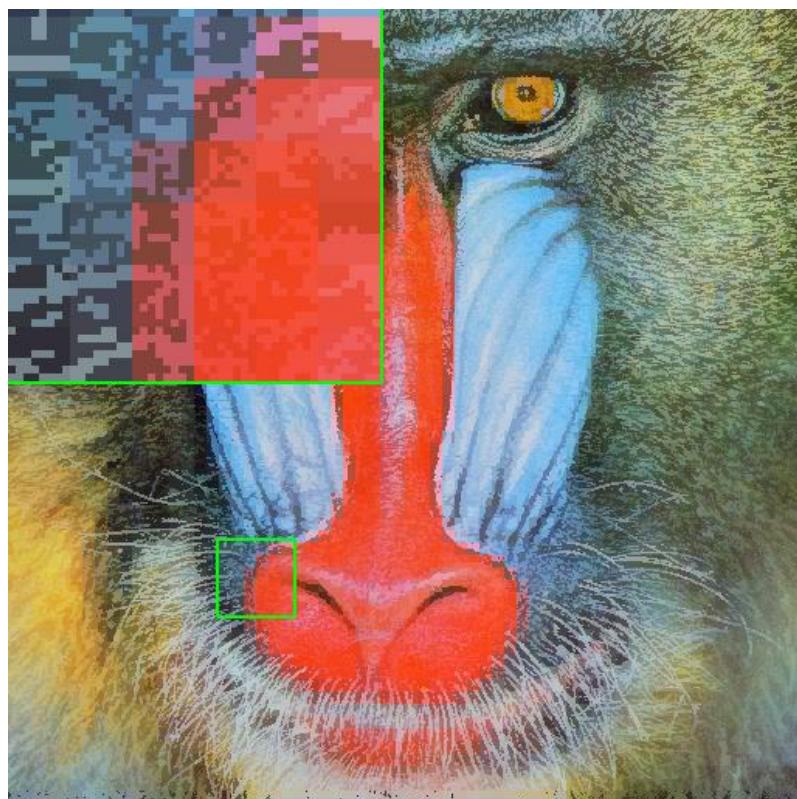
$$Y = W_R \times X_R + W_G \times X_G + W_B \times X_B, \quad (3.37)$$

X_R 、 X_G 及 X_B 為彩色影像三通道的灰階值； W_R 、 W_G 及 W_B 為三通道的權重； Y 為彩色影像轉換後的灰階影像。若為一般依據的亮度規則(Luminance rule)，三通道的權重(W_R, W_G, W_B)則為(0.299, 0.587, 0.114)，但在此研究中作者提出另一種權重規則(Weighted rule)，根據其規則可得到權重(1/3, 1/3, 1/3)，依照論文中的實驗結果可以發現此權重能夠降低重建影像的均方差(Mean square error, MSE)。

另一種方法是先將 RGB 三通道分別藉由絕對矩量區塊截斷編碼壓縮，並取得各通道的量化色階及位元圖(a_R, b_R, h_R)、(a_G, b_G, h_G)及(a_B, b_B, h_B)。最後，產生單一位元圖 h_{SBM} 的方法是依據多數決規則(Majority rule)判斷三通道位元圖 h_R 、 h_G 、 h_B ，當相同位置下有兩個以上的通道其位元圖為”1”，則 $h_{SBM} = 1$ ，反之則 $h_{SBM} = 0$ 。圖 3.16 及圖 3.17 為單一位元圖區塊截斷編碼依據亮度規則 8×8 及 16×16 的結果，圖 3.18 及圖 3.19 為依據權重規則 8×8 及 16×16 的結果，圖 3.20 及圖 3.21 為依據多數決規則 8×8 及 16×16 的結果。雖然依據權重規則能有較低的均方差，但從實際影像中可以發現依據三個不同的規則所產生的結果差異不大。

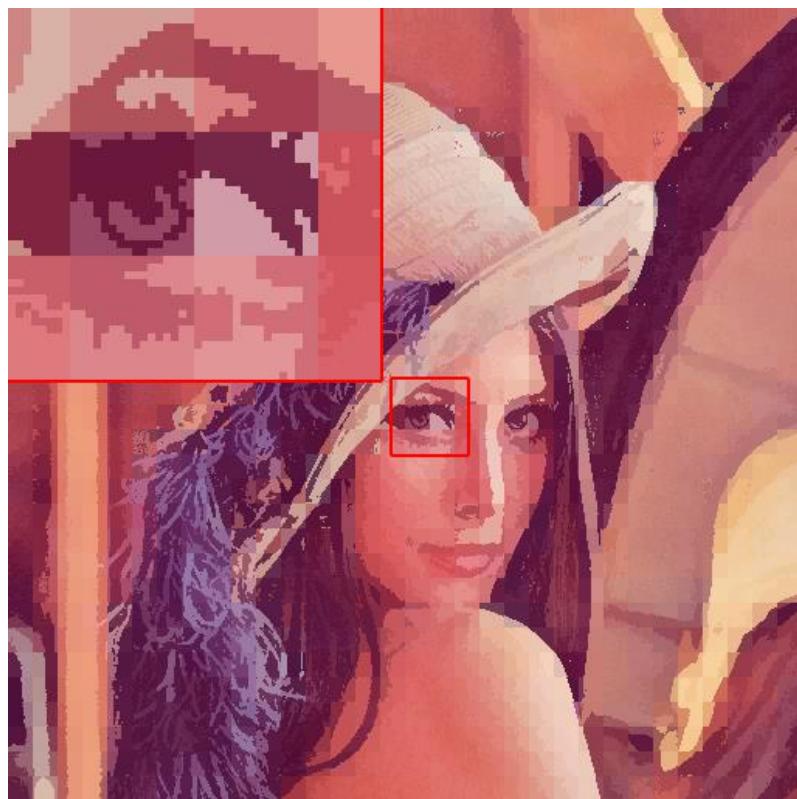


(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像

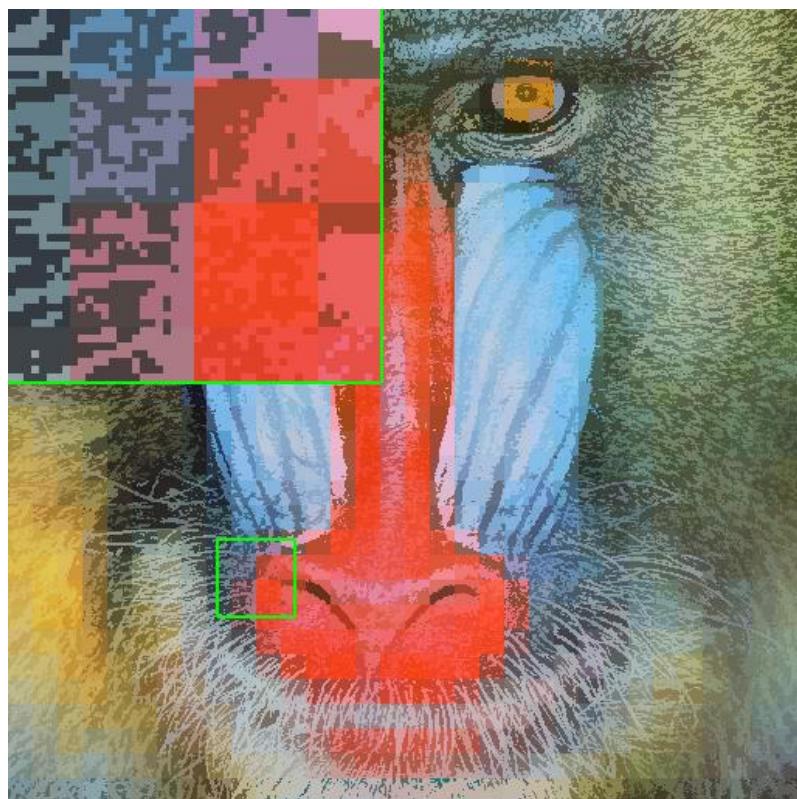


(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.16 SBMBTC 依據亮度規則重建之影像(8×8 區塊大小)， $CR = 13.71$

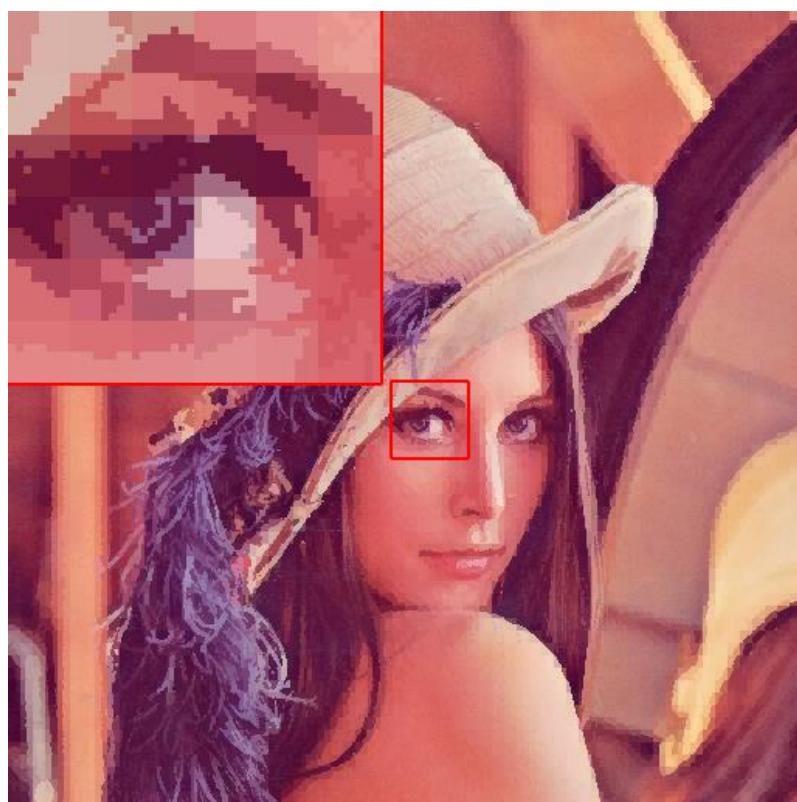


(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像

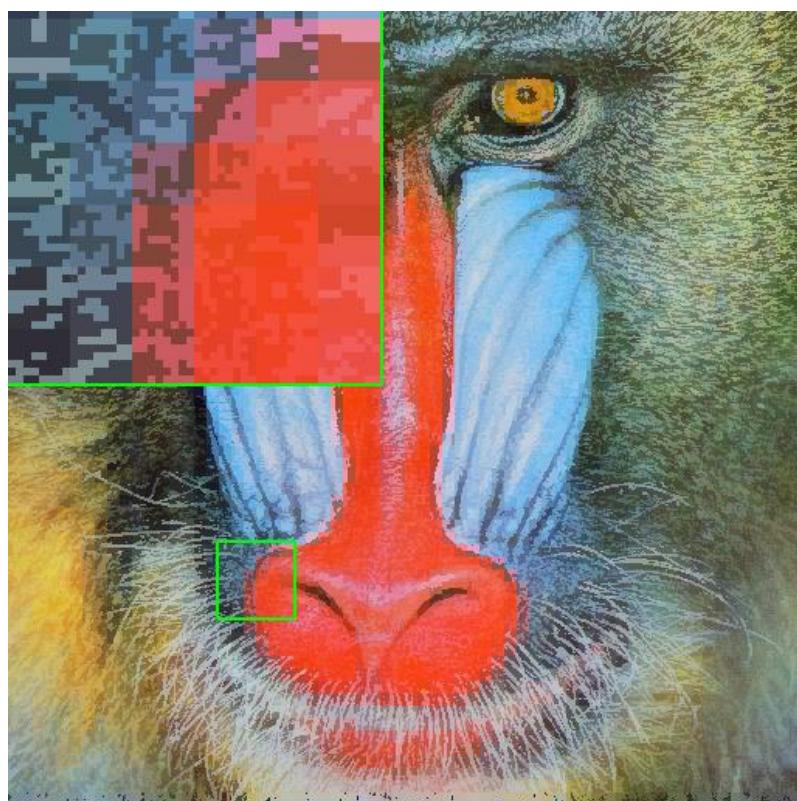


(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.17 SBMBTC 依據亮度規則重建之影像(16×16 區塊大小)， $CR = 20.21$

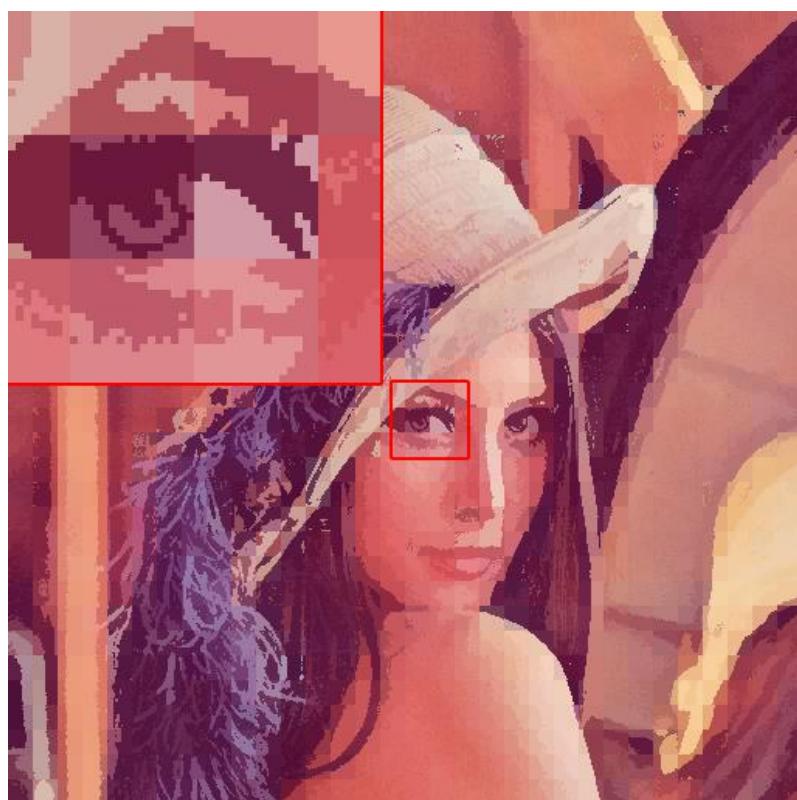


(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像

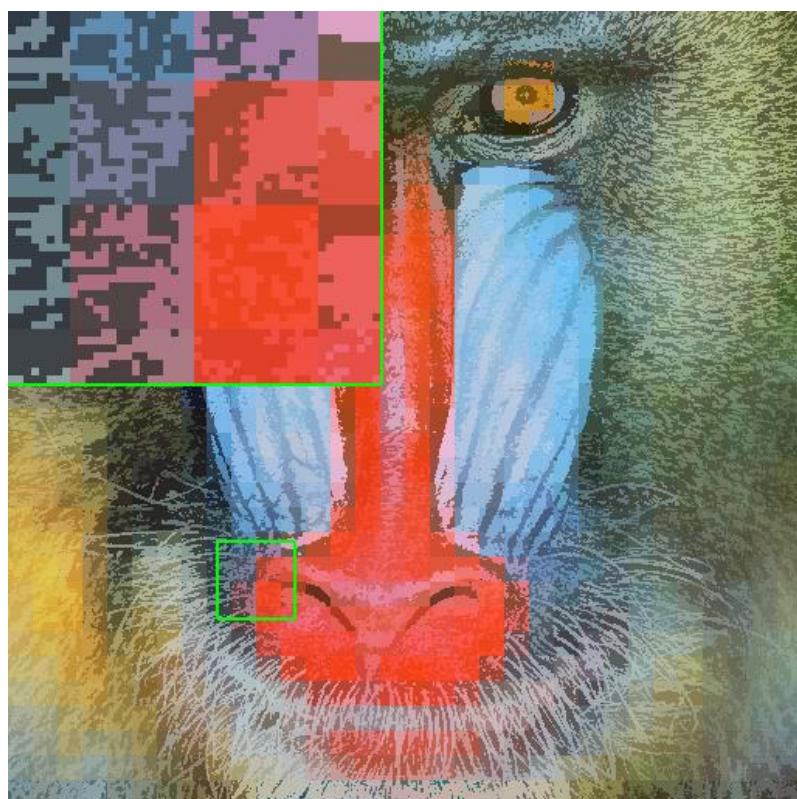


(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.18 SBMBTC 依據權重規則重建之影像(8×8 區塊大小)， $CR = 13.71$

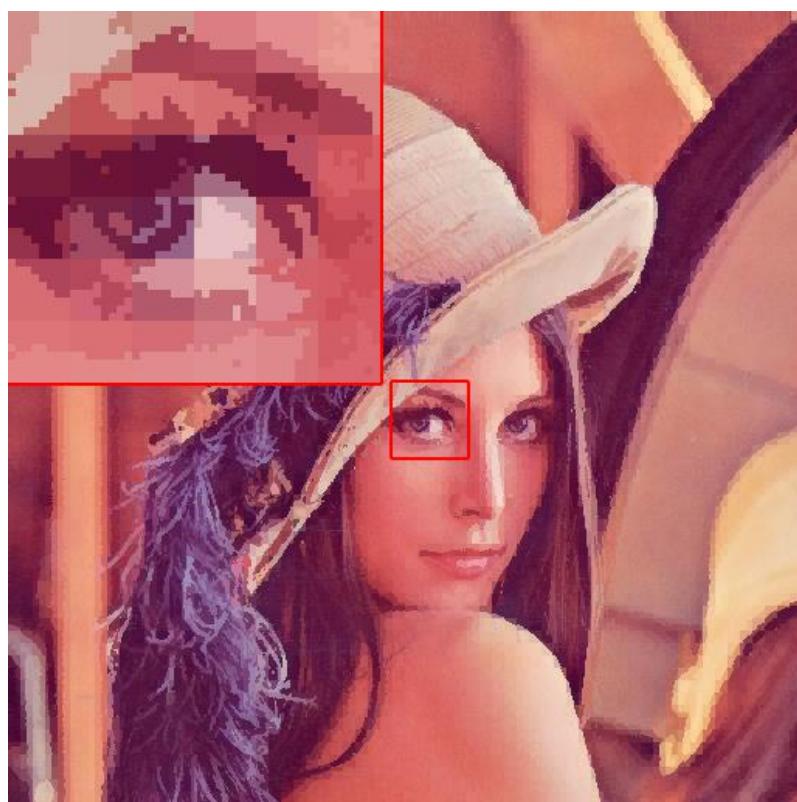


(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像

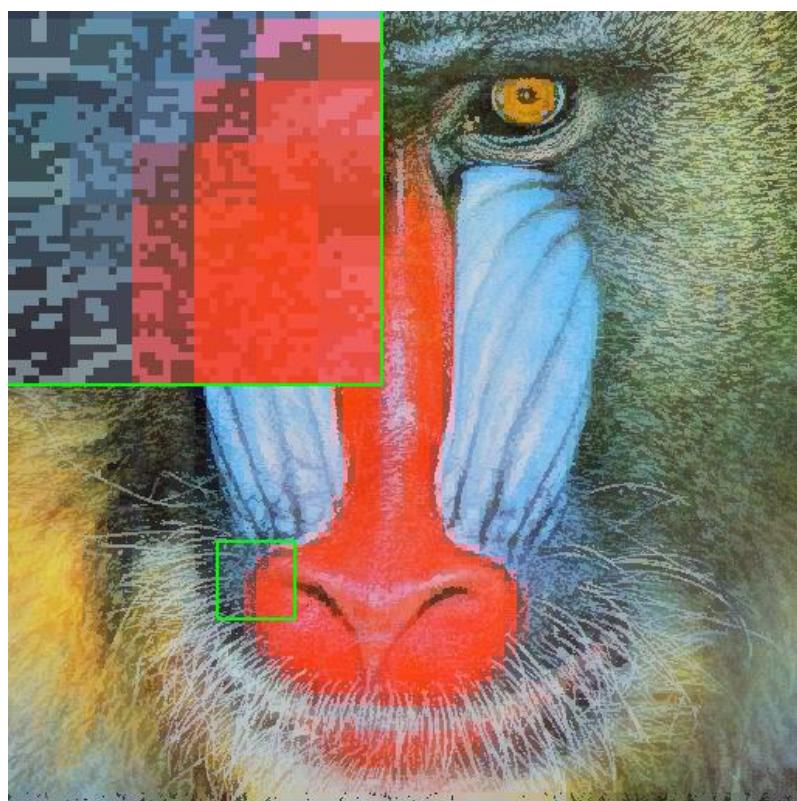


(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.19 SBMBTC 依據權重規則重建之影像(16×16 區塊大小)， $CR = 20.21$

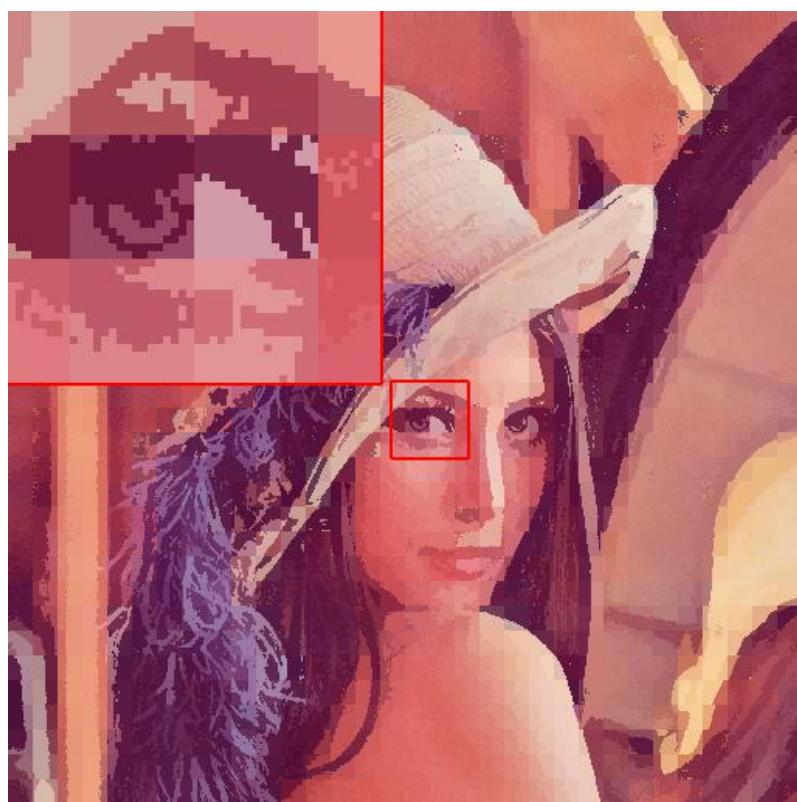


(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像

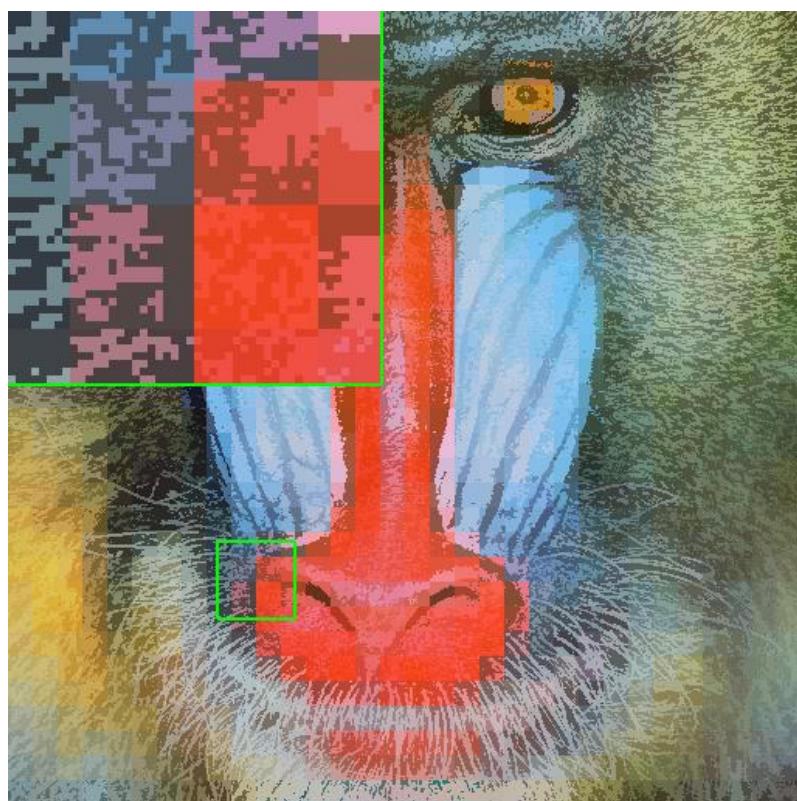


(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.20 SBMBTC 依據多數決規則重建之影像(8×8 區塊大小)， $CR = 13.71$



(a) 彩色影像 1 經 SBMBTC 壓縮後重建的影像



(b) 彩色影像 2 經 SBMBTC 壓縮後重建的影像

圖 3.21 SBMBTC 依據多數決規則重建之影像(16×16 區塊大小)， $CR = 20.21$

3.8 基於單一位元圖的編碼方案(Coding schemes based on SBM, CS-SBM)

雖然單一位元圖技術(SBM)能夠有效地提升壓縮率，但也使得影像品質降低。因此，Hu 及 Chang 等人[30]於 2014 年提出一個基於單一位元圖區塊截斷編碼(SBMBTC)技術的改良方法。文獻中提出兩個部分的改良，第一個方案(First scheme, FS)的目的是改善單一位元圖區塊截斷編碼的影像品質並維持其原有的優異壓縮率。它的處理方法是先將 RGB 三通道分別用絕對矩量區塊截斷編碼壓縮，得到各通道的量化色階 (a_R, b_R) 、 (a_G, b_G) 及 (a_B, b_B) ，再藉由以下公式取得單一位元圖：

$$d_0 = (X_R - a_R)^2 + (X_G - a_G)^2 + (X_B - a_B)^2, \quad (3.38)$$

$$d_1 = (X_R - b_R)^2 + (X_G - b_G)^2 + (X_B - b_B)^2, \quad (3.39)$$

X_R 、 X_G 及 X_B 為彩色影像三通道的灰階值； d_0 及 d_1 為平方的歐幾里得距離(Squared Euclidean distances)。藉由上述公式可以得知是較大還是較小的三通道量化色階與原始三通道灰階較相近，若 $d_0 > d_1$ 則 $h_{SBM} = 1$ ，反之則 $h_{SBM} = 0$ 。由於取得的單一位元圖 h_{SBM} 會與原本各通道產生的 h_R 、 h_G 及 h_B 有所不同，這樣與矩量區塊截斷編碼定義的量化色階會有差異，因此在此研究中將會重新計算量化色階，RGB 三通道用取得的單一位元圖 h_{BM} 代入下方公式就可得到各自重新計算的量化色階，

$$a = \frac{1}{m-q} \sum_{h(m,n)=0} x(m, n), \quad (3.40)$$

$$b = \frac{1}{q} \sum_{h(m,n)=1} x(m, n), \quad (3.41)$$

$h(m, n)$ 表示的是區塊的位元圖； $x(m, n)$ 表示原始區塊的灰階值； q 表示 $h(m, n) = 1$ 的數量。依照上述處理方式處理所有的不重疊區塊後就可完成第一方案(FS)壓縮。

第二個方案(Second scheme, SS)是基於第一方案(FS)處理後再進一步的減少比特率(Bit rate)的方法，其流程圖如圖 3.22 所示。首先介紹相似區塊預測技術(Similar block prediction technique)，當目前處理的區塊不在第一行或第一列時，在當前區塊尋找已編碼完成的周圍區塊判斷是否相似，尋找周圍區塊的數量用 SR 表示，不同尋找的數量會對應不同尋找的範圍，如表 3.2 所示。依照平方的歐幾里得距離計算各區塊與當前區塊的相似度 d ，接著將最小的 d_{min} 拿來判斷是否比設定的門檻值 $THSED$ 還低，若

低於則表示 d_{min} 對應的區塊能取代目前的區塊，反之則表示不夠相似，因此將此區塊用後續介紹方式的壓縮。此外要注意的是，這邊需要多 1 bit 去表示區塊是否由鄰近區塊表示，”0”表示目前區塊是經由一般的壓縮；”1”表示目前區塊是由其他區塊取代，其後續儲存的資料為取代當前區塊的區塊位置，要讀取的二進制資料量為 $\lceil \log_2 SR \rceil$ 。以 $SR = 14$ 為例，則需要讀取 $\lceil \log_2 14 \rceil = \lceil 3.807 \rceil = 4$ bits。此 1 bit 在此稱為區塊類型指標(Block type indicator)。

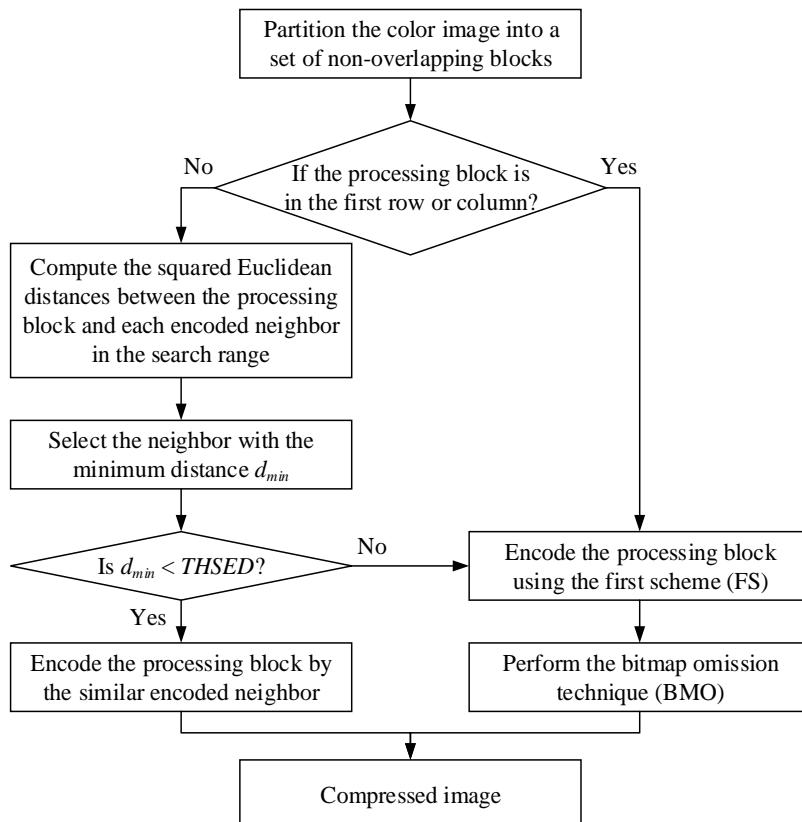


圖 3.22 第二方案(SS)的流程圖

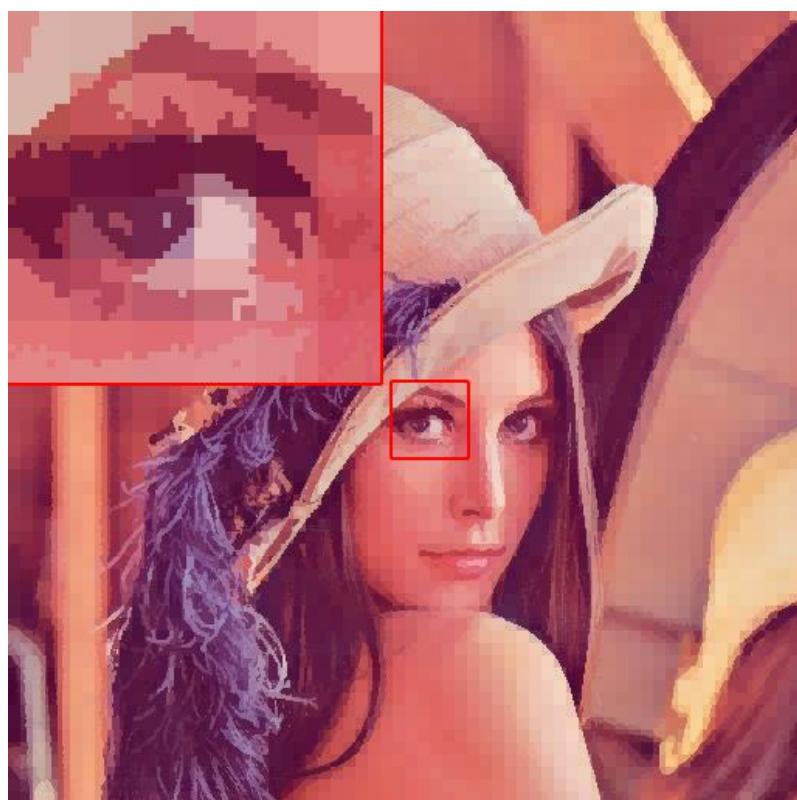
表 3.2 不同 SR 對應的尋找範圍

20	15	16	17	18	21
14	9	7	8	10	19
13	6	3	2	4	11
12	5	1	x		

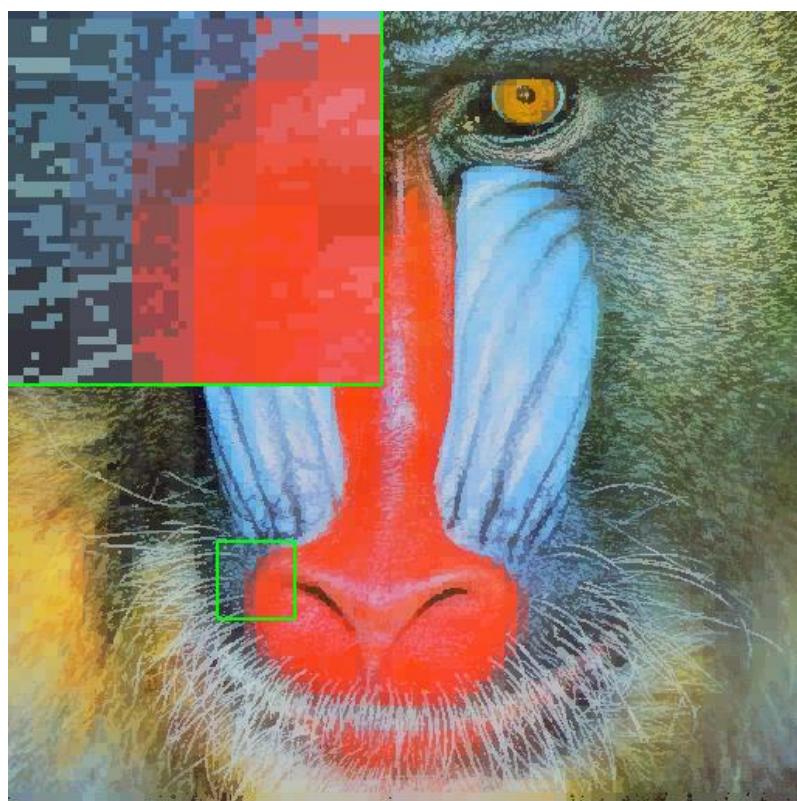
當不夠相似時，則用第一方案位元圖省略技術(FS-BMO)去壓縮此區塊。第一方案如前面所述，而位元圖省略技術(Bit-map omission technique)則是出自文獻[31]，其設一門檻值 $THBO$ 去判斷目前區塊的兩個量化色階是否相似，當量化色階相減的絕對值小於 $THBO$ 則表示量化色階足夠相似，所以只須輸出平均值 m 。此外，當相同區塊三通道的量化色階各自都足夠相似時，因為三通道都只需輸出平均值 (m_R, m_G, m_B) ，所以不需要位元圖去表示量化色階的位置。此外，由於三通道都需要 1 bit 表示區塊是否使用位元圖省略技術，一個區塊總共會需要 3 bits 表示，這 3 bits 稱為組件類型指標(Component type indicator)，所有可能的情況如表 3.3。實驗結果如圖 3.23 及圖 3.24 為第一方案位元圖省略技術 8×8 及 16×16 的結果， $THBO$ 設為 10。在此僅呈現第一方案位元圖省略技術(FS-BMO)，因為雖然第二方案(SS)能夠有更高的壓縮率，但不僅使影像品質降低，也因判斷相似度的運算複雜度極高而使得處理速度大幅下滑，實際應用的價值不高。

表 3.3 三通道位元圖省略技術可能發生的所有情況

組件類型指標	區塊三通道的壓縮資料	資料量
000	(m_R, m_G, m_B)	$3 + 8 \times 3$
001	$(m_R, m_G, a_B, b_B, h_{SBM})$	
010	$(m_R, a_G, b_G, m_B, h_{SBM})$	$3 + 8 \times 2 + (2 \times 8) \times 1 + M \times N$
100	$(a_R, b_R, m_G, m_B, h_{SBM})$	
011	$(m_R, a_G, b_G, a_B, b_B, h_{SBM})$	
110	$(a_R, b_R, a_G, b_G, m_B, h_{SBM})$	$3 + 8 \times 1 + (2 \times 8) \times 2 + M \times N$
101	$(a_R, b_R, m_G, a_B, b_B, h_{SBM})$	
111	$(a_R, b_R, a_G, b_G, a_B, b_B, h_{SBM})$	$3 + (2 \times 8) \times 3 + M \times N$

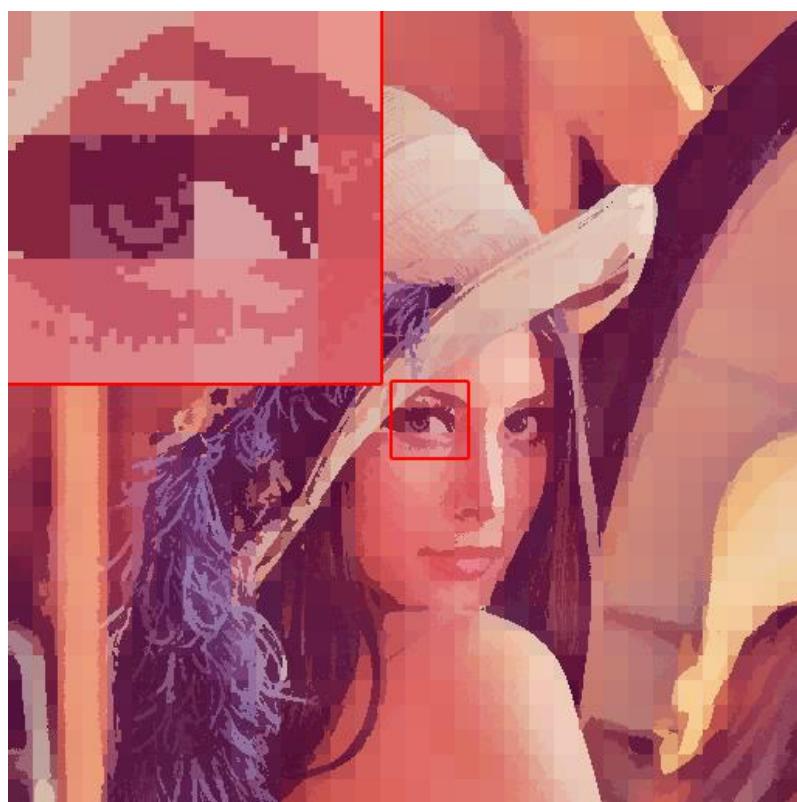


(a) 彩色影像 1 經 FS-BMO 壓縮後重建的影像， $CR = 18.73$

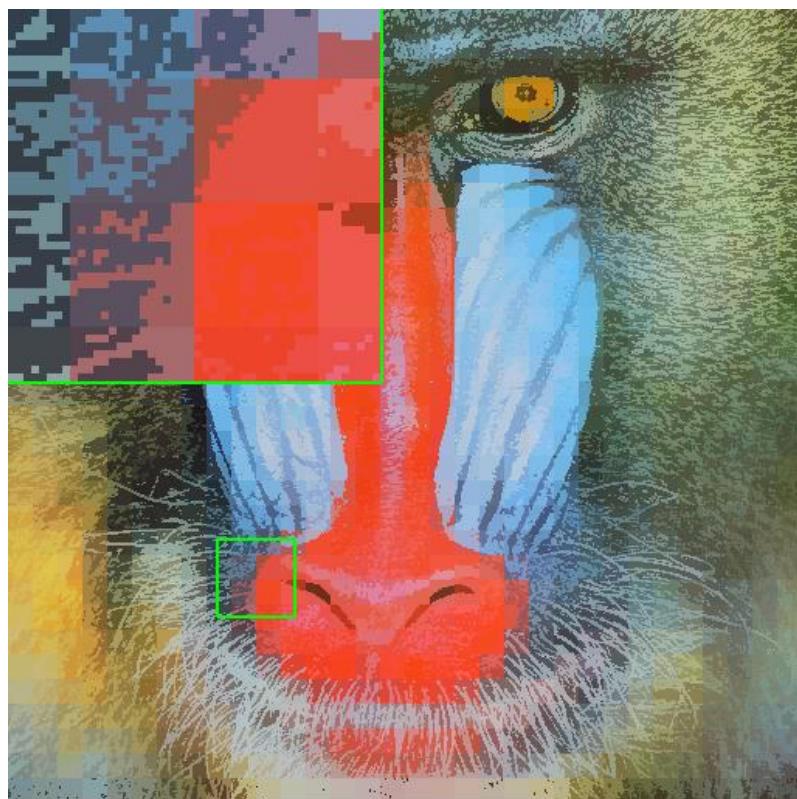


(b) 彩色影像 2 經 FS-BMO 壓縮後重建的影像， $CR = 13.70$

圖 3.23 FS-BMO 重建之影像(8 × 8區塊大小)



(a) 彩色影像 1 經 FS-BMO 壓縮後重建的影像， $CR = 25.81$



(b) 彩色影像 2 經 FS-BMO 壓縮後重建的影像， $CR = 20.14$

圖 3.24 FS-BMO 重建之影像(16 × 16 區塊大小)

3.9 分群區塊截斷編碼(BTC with k-means quad clustering, IBTC-KQ)

分群區塊截斷編碼為 Mathews 及 Nair 等人[32]於 2014 所提出，其目的是改善重建影像的品質。過往大部分的技術都是在 RGB 色彩模型上處理，但 HSV 色彩模型(Hue, Saturation, Value)是最被廣泛使用且較符合人眼視覺感受的彩色模型。因此在此研究中，會先將彩色影像從 RGB 色彩模型轉換成 HSV 色彩模型，接著將各通道的影像切成多個不重疊的區塊，各區塊的處理步驟如下，

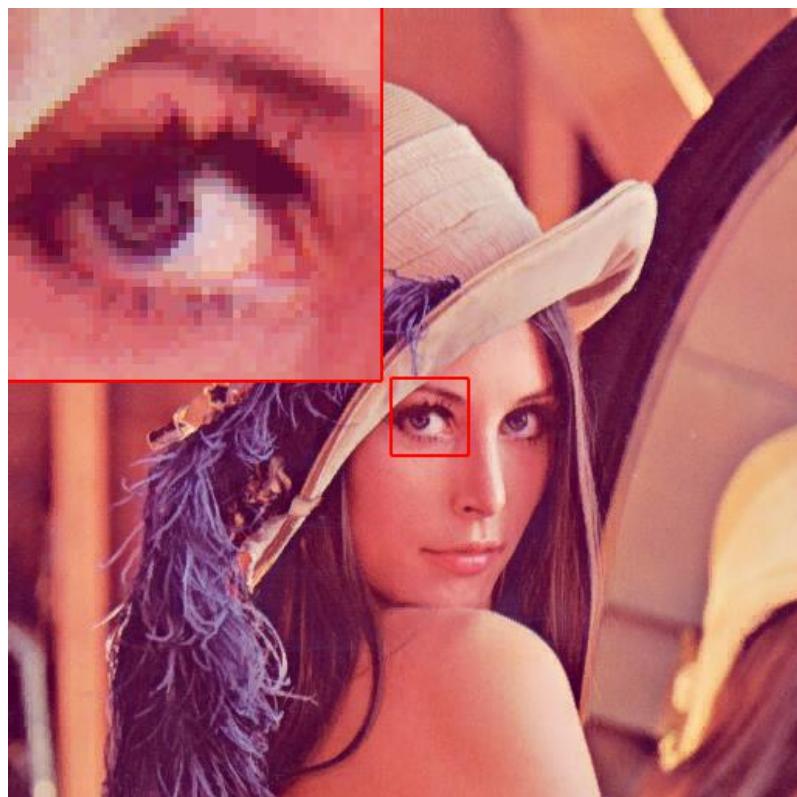
1. 將區塊 B_H 、 B_S 及 B_V 當中的像素值各自用 k-means 演算法分成四個群 (hw_0, hw_1, hw_2, hw_3) 、 (sw_0, sw_1, sw_2, sw_3) 及 (vw_0, vw_1, vw_2, vc_3) ，這樣就能夠將相似的像素聚集到同一個群裡，然後再從各個像素群裡取得各群的中心 (hc_0, hc_1, hc_2, hc_3) 、 (sc_0, sc_1, sc_2, sc_3) 及 (vc_0, vc_1, vc_2, vc_3) 。
2. 計算各通道各群的平均灰階值 $(h\mu_0, h\mu_1, h\mu_2, h\mu_3)$ 、 $(s\mu_0, s\mu_1, s\mu_2, s\mu_3)$ 及 $(v\mu_0, v\mu_1, v\mu_2, v\mu_3)$ 。
3. 依照上述取得的資訊可產生區塊三通道的位元圖 (hb, sb, vb) ，公式如下：

$$hb = \begin{cases} 00, & \text{if } hw_j \in hc_0 \\ 01, & \text{if } hw_j \in hc_1 \\ 10, & \text{if } hw_j \in hc_2 \\ 11, & \text{if } hw_j \in hc_3 \end{cases}, \quad sb = \begin{cases} 00, & \text{if } sw_j \in sc_0 \\ 01, & \text{if } sw_j \in sc_1 \\ 10, & \text{if } sw_j \in sc_2 \\ 11, & \text{if } sw_j \in sc_3 \end{cases}, \quad vb = \begin{cases} 00, & \text{if } vw_j \in vc_0 \\ 01, & \text{if } vw_j \in vc_1 \\ 10, & \text{if } vw_j \in vc_2 \\ 11, & \text{if } vw_j \in vc_3 \end{cases}, \quad (3.42)$$

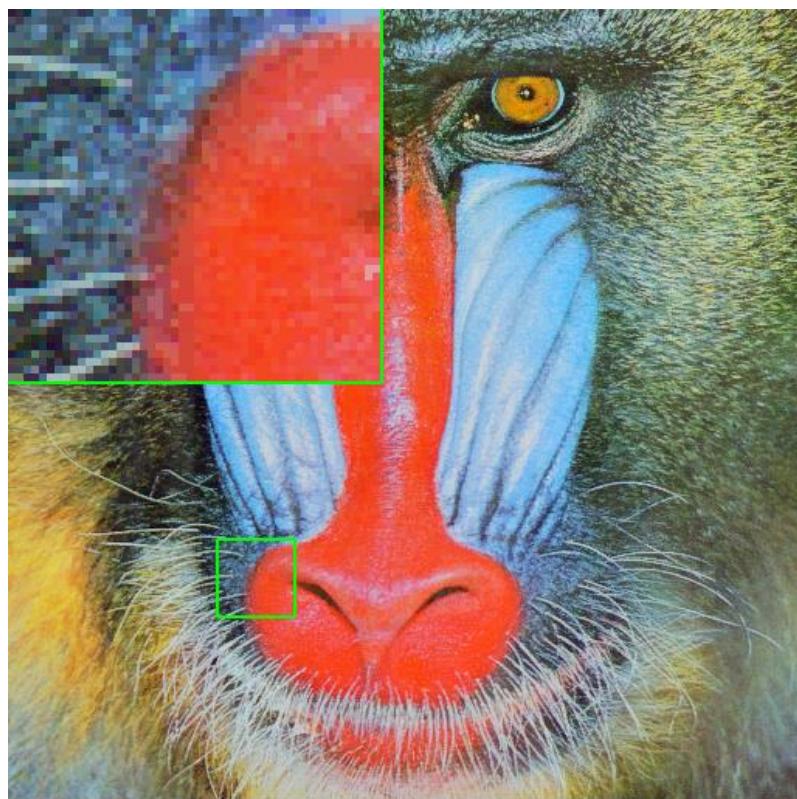
每個位元圖需要 2 bits 表示。

4. 各區塊儲存各群的平均灰階及位元圖 $(\mu_0, \mu_1, \mu_2, \mu_3, b)$ ，重複步驟 1 到 3 直到完成所有區塊。

重建影像則是依照位元圖的值 $(00, 01, 10, 11)$ ，去填入對應的平均灰階值 $(\mu_0, \mu_1, \mu_2, \mu_3)$ 。實驗結果如圖 3.25、圖 3.26，雖然重建的影像沒有任何壓縮的痕跡，但壓縮率卻非常的低，且由於使用 k-means 導致當越多區塊(輸入影像越大或者區塊大小設定的較小)時，處理整張影像的時間會變得越長，有時可能需要耗費數十秒甚至數百秒。

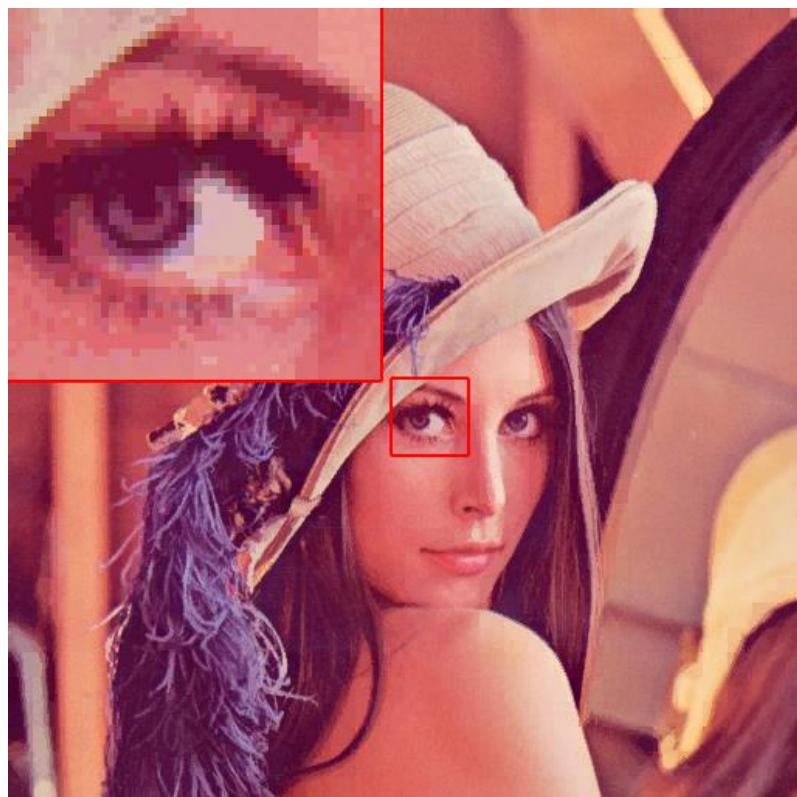


(a) 彩色影像 1 經 IBTC-KQ 壓縮後重建的影像

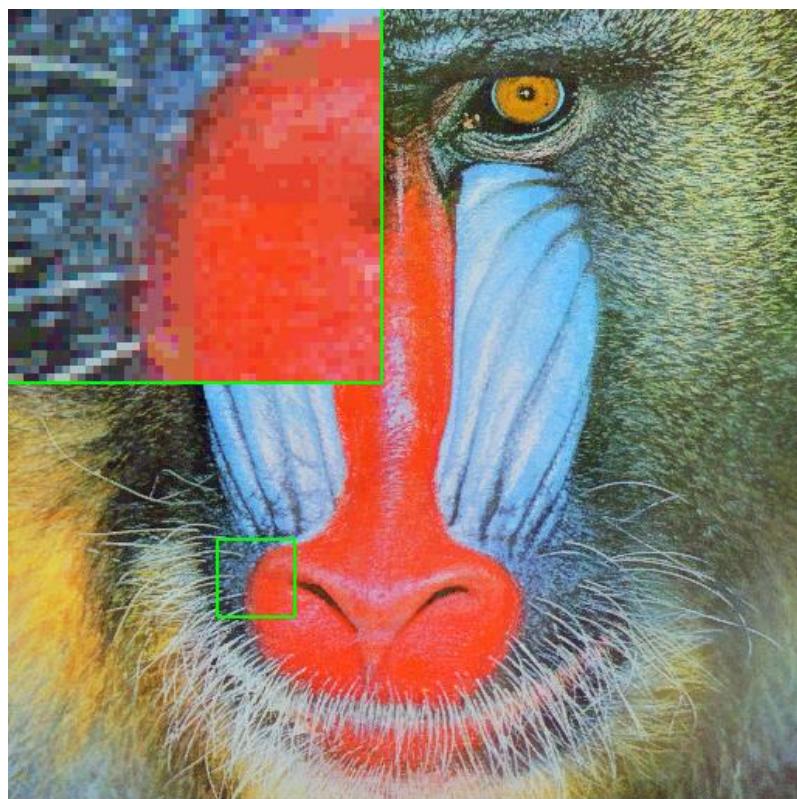


(b) 彩色影像 2 經 IBTC-KQ 壓縮後重建的影像

圖 3.25 IBTC-KQ 重建之影像(8×8 區塊大小)， $CR = 3.20$



(a) 彩色影像 1 經 IBTC-KQ 壓縮後重建的影像



(b) 彩色影像 2 經 IBTC-KQ 壓縮後重建的影像

圖 3.26 IBTC-KQ 重建之影像(16×16 區塊大小)， $CR = 3.76$

3.10 彩色點擴散區塊截斷編碼(Color DDBTC, CDDBTC)

彩色點擴散區塊截斷編碼技術為 Guo 及 Prasetyo [24]於 2015 所提出，此將點擴散區塊截斷編碼(DDBTC)及單一位元圖(SBM)這兩個技術結合，其中，單一位元圖技術將會依照其權重規則。會使用到的處理公式如下，詳細定義如 3.5 節的點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)：

$$a = \min(B), \quad (3.43)$$

$$b = \max(B), \quad (3.44)$$

$$\bar{x} = \frac{1}{M \times N} \sum_{(m,n)} x(m, n), \quad (3.45)$$

$$c(m, n) = x(m, n) + s(m, n), \text{ where } s(m, n) = \sum_{(p,q) \in R} \sum \frac{e(m+p, n+q) \times w(p, q)}{\text{sum}}, \quad (3.46)$$

$$e(m, n) = \begin{cases} c(m, n) - b, & \text{if } c(m, n) \geq \bar{x} \\ c(m, n) - a, & \text{if } c(m, n) < \bar{x} \end{cases} \quad (3.47)$$

$$h(m, n) = \begin{cases} 1, & \text{if } c(m, n) \geq \bar{x} \\ 0, & \text{if } c(m, n) < \bar{x} \end{cases} \quad (3.48)$$

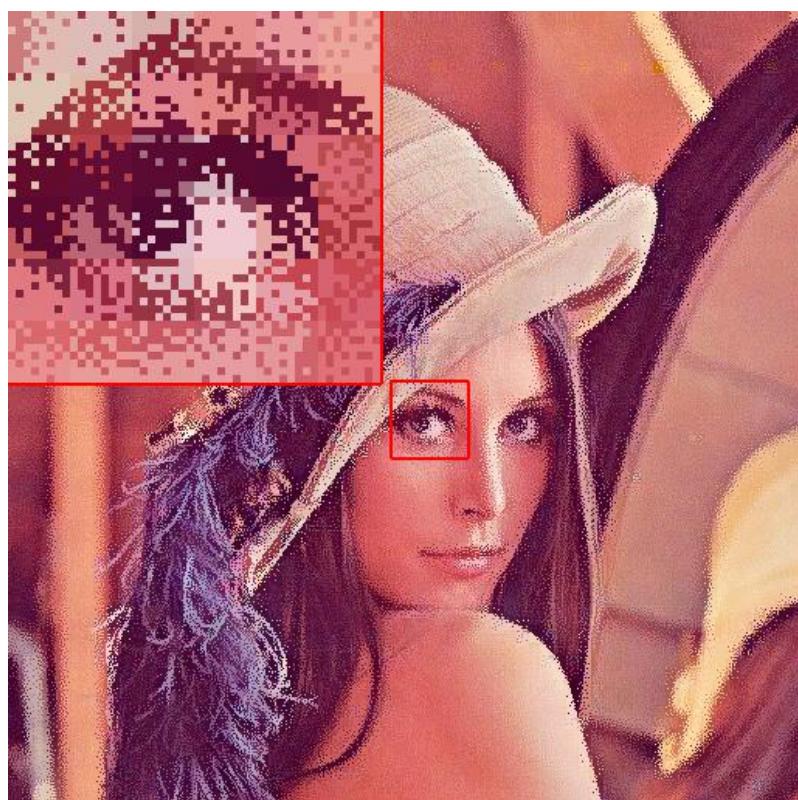
$$\text{sum} = \sum_{(p,q) \in R} \begin{cases} w(p, q), & \text{if } (i + p, j + q) \text{ within range and } o(m + p, n + q) > o(m, n) \\ 0, & \text{if } (i + p, j + q) \text{ out of range or } o(m + p, n + q) \leq o(m, n) \end{cases}, \quad (3.49)$$

首先，將彩色影像從 RGB 色彩模型依據權重規則轉換成灰階影像，接著將彩色影像及灰階影像切成多個不重疊的區塊。彩色影像的區塊代入公式(3.43)-(3.44)可得到三通道的量化色階($a_R, b_R, a_G, b_G, a_B, b_B$)；灰階影像的區塊帶入公式(3.45)-(3.49)可得到單一位元圖 h_L ，取得量化色階及單一位元圖後即完成壓縮。當中會使用到的類別矩陣(CM)及擴散矩陣(DM)如表 3.1 所示。

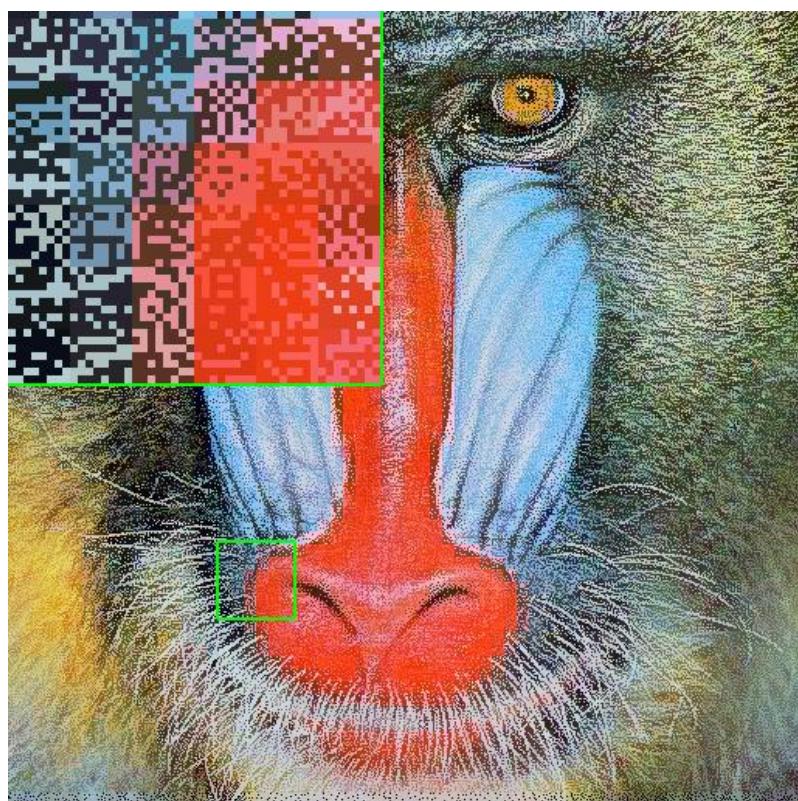
彩色點擴散區塊截斷編碼的重建方式如下：

$$y(m, n) = \begin{cases} b, & \text{if } h(m, n) = 1 \\ a, & \text{if } h(m, n) = 0 \end{cases} \quad (3.50)$$

三通道的區塊使用同一個位元圖代入上述公式即可得到重建的影像， 8×8 及 16×16 的結果如圖 3.27 及圖 3.28。雖然重建影像上明顯有許多脈衝雜訊(Impulse noise)，但影像比過往技術的結果更清晰，並且降低了區塊效應(Blocking effect)的影響。當使用越高解析度的原始影像，以半色調為基礎的半色調技術能有更佳的影像品質，脈衝雜訊會細微到人眼無法察覺，而過往技術仍會受到區塊效應影響人眼視覺感受。

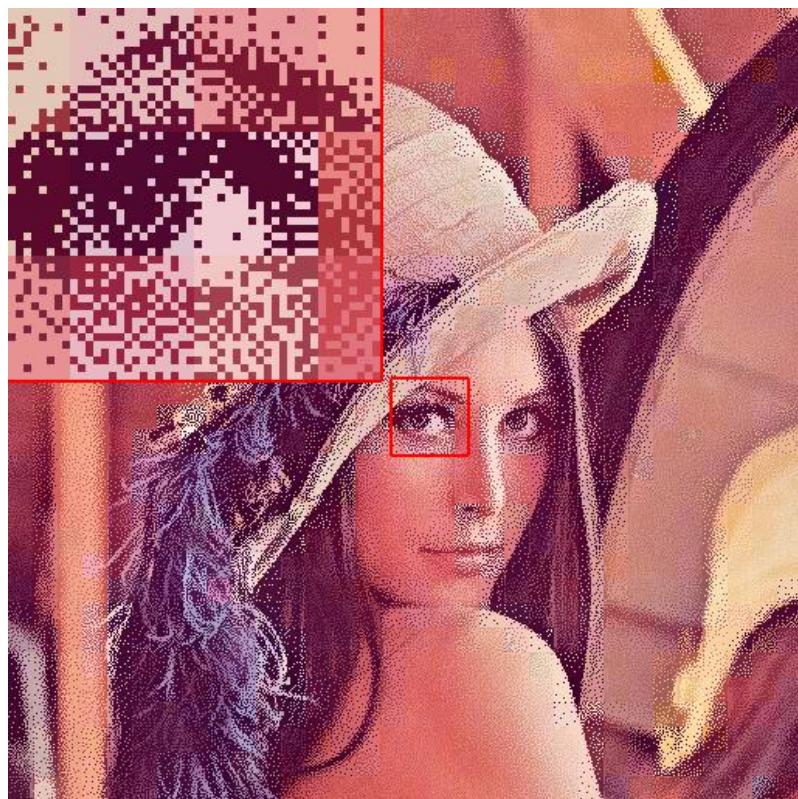


(a) 彩色影像 1 經 CDDBTC 壓縮後重建的影像

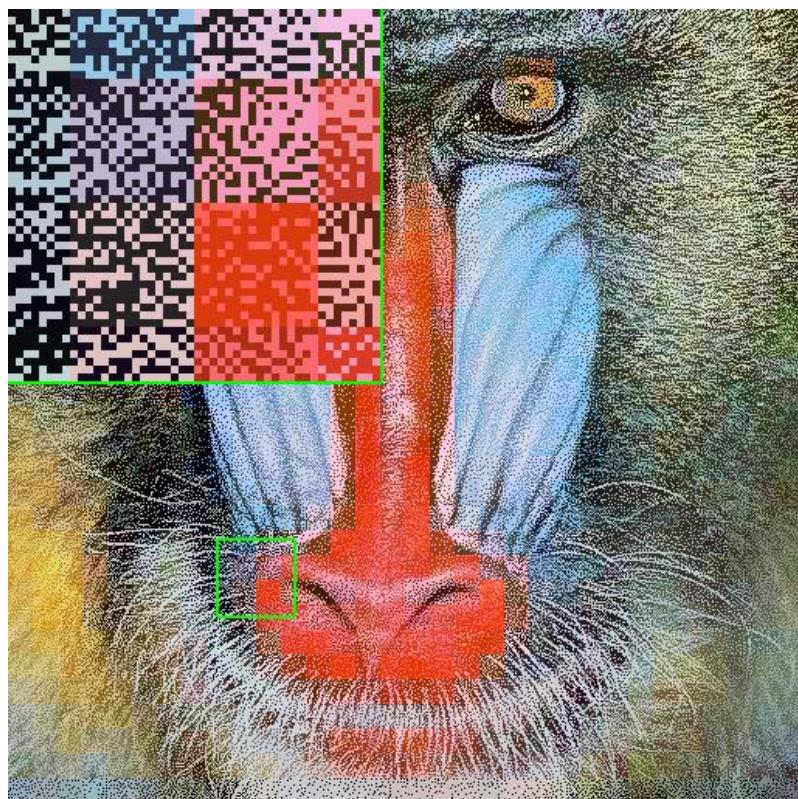


(b) 彩色影像 2 經 CDDBTC 壓縮後重建的影像

圖 3.27 CDDBTC 重建之影像(8×8 區塊大小), $CR = 13.71$



(a) 彩色影像 1 經 CDBTC 壓縮後重建的影像



(b) 彩色影像 2 經 CDBTC 壓縮後重建的影像

圖 3.28 CDBTC 重建之影像(16×16 區塊大小), $CR = 20.21$

第四章 數位多色調技術文獻探討

在此章節中將介紹數種不同的數位多色調技術(Digital multitoneing technique)。多色調技術的前身是第二章所討論的數位半色調技術。半色調技術能夠將影像從灰階轉成由黑白點所組成的影像，在目前的印刷產業中已經是主要使用的技術，但為了能夠持續提升輸出結果的品質，人們開始著手從硬體設備、演算法、墨水材料等進行改良。在現在的情況下，印刷設備只能以半色調技術的方法將黑與白(打點或不打點)表現 8-bit 灰階的輸出結果，這使得輸出結果容易有失真的情況，若是假設硬體設備能夠用更多種不同打印的輸出來表現 8-bit 灰階，則能夠減輕失真的問題並改善影像的品質，因此才衍伸出數位多色調技術。多色調技術能夠以多個色階重建影像，當輸出的色階越多時，輸出的結果也會越接近原始的灰階影像，但這是一個設備成本及輸出品質的權衡問題，當需要越多色階時則硬體設備必須能夠支援，成本相對也比較高。因此在數位多色調技術的研究目標也如同數位半色調技術，期望能夠在相同的條件下，例如輸出色階固定時，能夠有最佳的輸出結果。

如同半色調技術，從性質可分類成藍雜訊(Blue-noise)及綠雜訊(Green-noise)。而從處理類型則可分類為基於單點像素點處理(Single-pixel-based process)、基於周圍像素點處理(Neighborhood-pixels-based process) 及基於迭代式處理(Iteration-based process)。以下的數位多色調技術依照上述的處理類型能分類為：

- 1) 基於單點像素點處理(Single-pixel-based process)
 - 多色調有序抖動法(Multitoneing ordered dither, M-OD)
 - 直接多位元搜尋法(Direct multi-bit search, DMS)
- 2) 基於鄰近像素點處理(Neighborhood-pixels-based process)
 - 多色調誤差擴散法(Multitoneing error-diffused, M-ED)
 - 色調替代之誤差擴散法(Multitoneing tone-replacement ED, M-TRED)
- 3) 基於迭代式處理(Iteration-based process)
 - 多色調直接二元搜尋法(Multitoneing direct binary search, M-DBS)

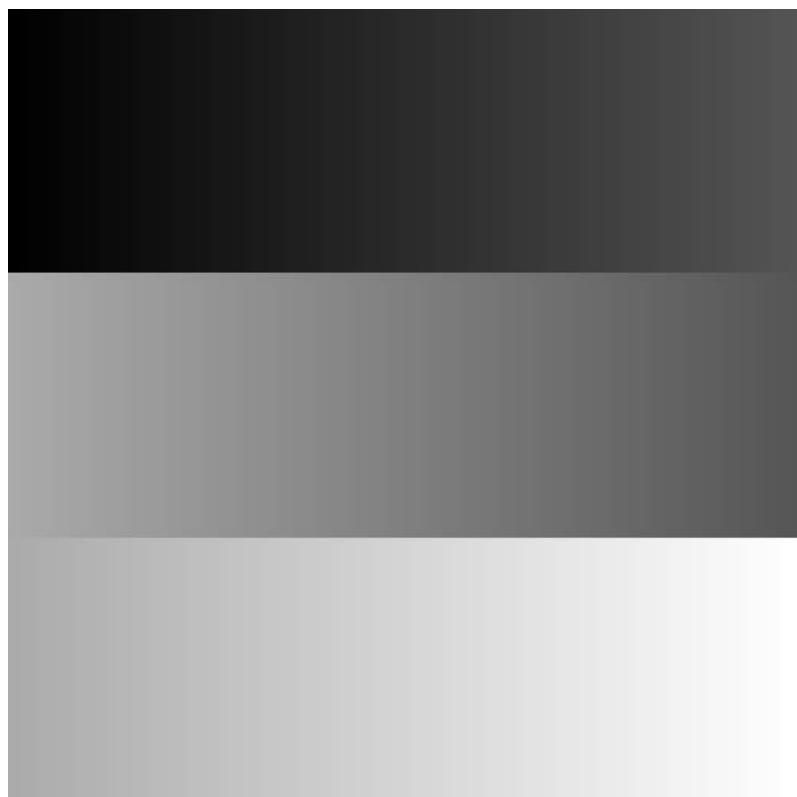
- 多色調藍雜訊抖動法(Multitone blue-noise dithering, M-BND)

這些技術將在後續章節依照提出的時間做介紹。此外，上述方法並非所有數位多色調技術，這邊所介紹的技術為代表性作品或者將會在之後與本論文所提出的技術做比較。

由於多色調可以使用多種色階產生不同的多色調影像，因此 q 色階(q -level)的多色調技術則表示使用 q 個輸出色階(Output level)，而 $\mathbf{Q} = \{g_1, g_2, \dots, g_q\}$ 則表示包含所有輸出色階的集合。在後續章節的介紹過程中會用與半色調相同的測試圖如圖 2.1 中兩張 512×512 大小的灰階影像作為實驗的展示結果，輸出色階 q 若沒有特別說明，一般情況將設定輸出色階為三。此外，由於有些技術可能在圖 2.1 中無法看出其缺陷，在此會以圖 4.1 作為額外的展示。



(a) 灰階影像 3



(b) 灰階影像 4

圖 4.1 用來展示特殊缺陷之測試灰階影像

4.1 多色調誤差擴散法(Multitonning error-diffused, M-ED)

多色調誤差擴散法相關技術可以參考文獻[33]-[36]，在這個小節中將基於這些參考文獻介紹多色調誤差擴散法的基礎演算法，基本上 M-ED 是前述 2.1 節誤差擴散法(Error-diffused, ED)的延伸。假設位置 (i, j) 表示目前處理位置，而每一點像素點都會經過以下方式處理。一個灰階像素點 $x(i, j)$ 輸入時，將會一併考慮周圍像素點擴散過來的取樣誤差 $s(i, j)$ ，並將兩者相加獲得 $c(i, j)$ 。接著 $c(i, j)$ 將與多色階集合 \mathbf{Q} 內的所有色階 g_s 進行比較，並且擁有最小絕對誤差的輸出色階 g' 將會被輸出至 $y(i, j)$ 當作結果。經過輸出 g' 之後，輸出的輸出色階 g' 與原始灰階像素點所產生的取樣誤差 $e(i, j)$ 會以誤差濾波器>Error filter, EF>進行誤差擴散的動作，將取樣誤差分配至周圍鄰近的像素點。重複以上步驟至整張影像的每一個像素點就可以獲得對應的數位多色調影像，而對應的公式如下：

$$c(i, j) = x(i, j) + s(i, j), \text{ where } s(i, j) = \sum_{(p, q) \in R} \sum \frac{e(i+p, j+q) \times w(p, q)}{\text{sum}}, \quad (4.1)$$

$$e(i, j) = c(i, j) - y(i, j), \quad (4.2)$$

$$\text{sum} = \sum_{(p, q) \in R} \begin{cases} w(p, q), & \text{if } (i + p, j + q) \text{ within range} \\ 0, & \text{if } (i + p, j + q) \text{ out of range} \end{cases}, \quad (4.3)$$

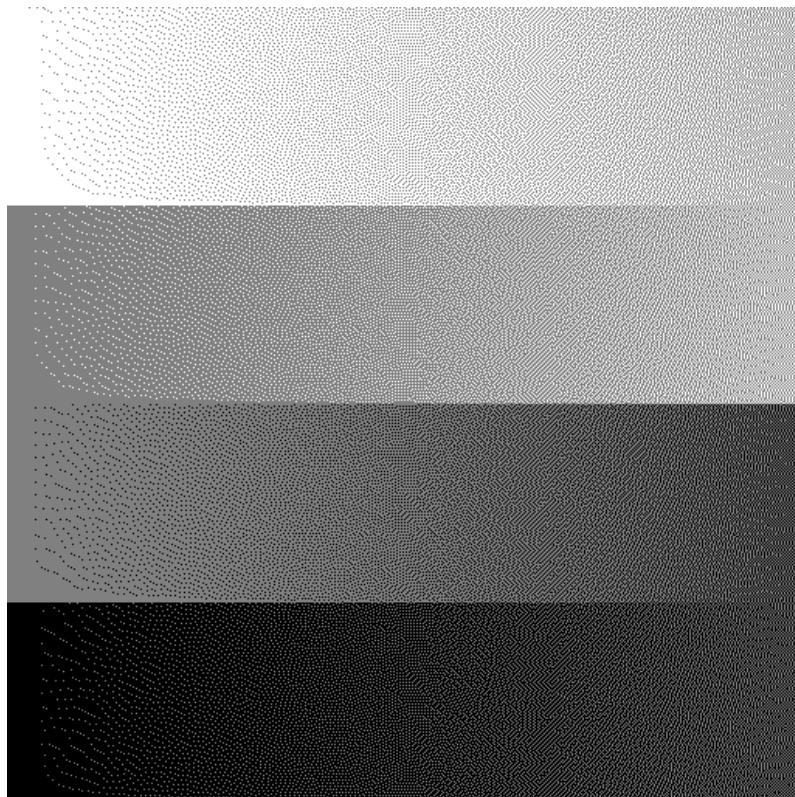
$$y(i, j) = g' = \arg \min_{g_s \in \mathbf{Q}} |c(i, j) - g_s|, \quad (4.4)$$

公式(4.1)-(4.3)的定義如同(2.1)、(2.2)及(2.4)。圖 4.2 則為對應的三色階之數位多色調影像。觀察其結果可得知，一些不自然的紋理圖樣仍出現在多色調影像結果中。

在多色調技術中常常會產生一個不自然的圖樣名為帶狀效應(Banding effect)，由於在圖 4.2 比較無法清楚表達帶狀效應，因此以圖 4.3 解釋其效應，其中以粗實線包圍的區域就是帶狀效應產生的位置。此現象的發生是因為在帶狀區域(Banding area)時，原本的灰階值與輸出色階是非常相似甚至相同，因此在周圍區域都會以某一個輸出色階轉換這個區域。但當周圍區域開始有變化，則會開始出現不同的輸出色階，由於人眼視覺系統對於低頻訊號較於敏感，因此這樣的低頻分佈很容易被觀察出來，並且造成不夠平滑的視覺質感。

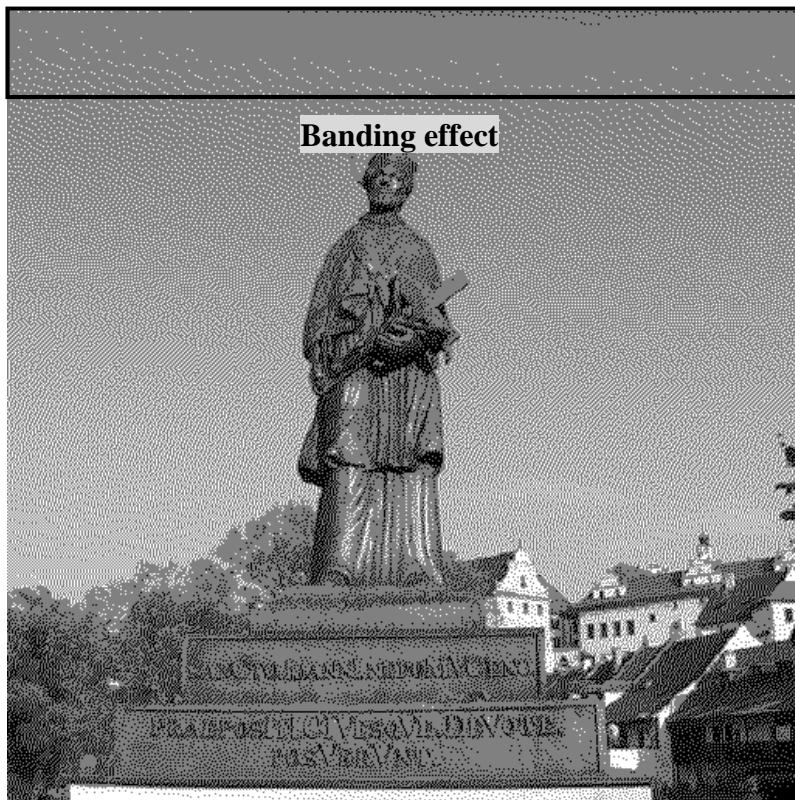


(a) 灰階影像 1 經多色調誤差擴散法處理後的多色調影像

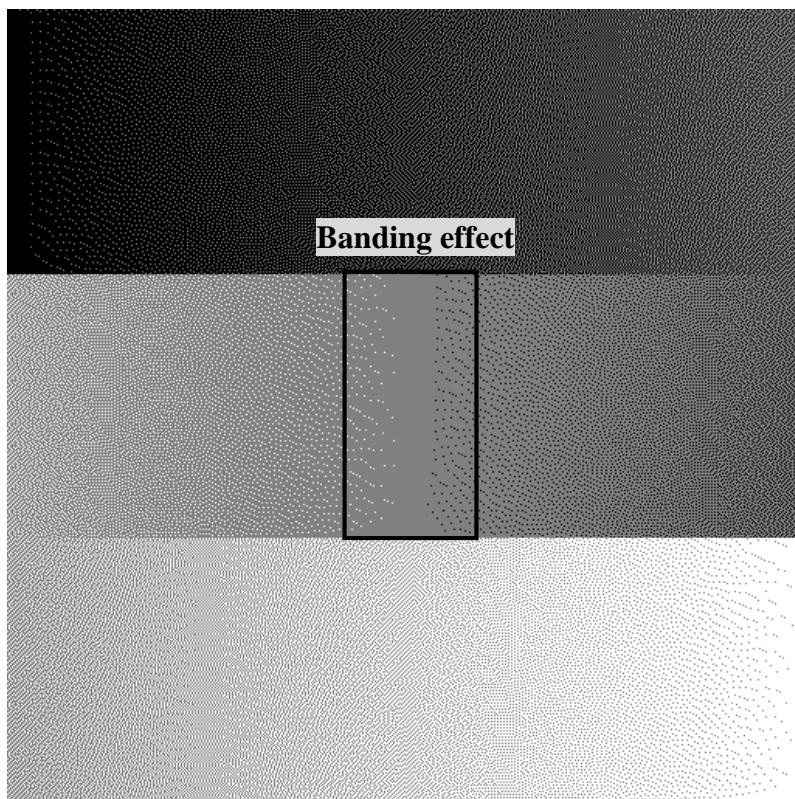


(b) 灰階影像 2 經多色調誤差擴散法處理後的多色調影像

圖 4.2 使用 Floyd-Steinberg 誤差擴散矩陣所產生的三色階多色調影像



(a) 灰階影像 3 經多色調誤差擴散法處理後的多色調影像



(b) 灰階影像 4 經多色調誤差擴散法處理後的多色調影像

圖 4.3 使用 Floyd-Steinberg 誤差擴散矩陣所產生的三色階多色調影像

4.2 多色調有序抖動法(Multitone ordered dither, M-OD)

此數位多色調技術可以從文獻[37]中得知，如同有序抖動法(Ordered dither, OD)，其技術採用一個有序抖動矩陣(Dithering matrix)進行數位多色調影像的轉換。此多色調技術與有序抖動法最大的不同之處為調整(Scaling)有序抖動矩陣。調整有序抖動矩陣是為了在不同輸出色階區間的灰階值能夠進行比較以輸出結果。假設目前色階數為三，因此對應的輸出量化色階應表示為0、0.5、1(以0至1尺度表示)，並採用表2.5中的有序抖動矩陣，要是目前輸入的灰階值為0.25，則有序抖動矩陣內的數值要等比例地調整為0(此稱前輸出色階 $g_{previous}$)至0.5(此稱後輸出色階 g_{next})這個範圍，接著才進行比較的動作。其詳細演算法如下所述，首先將原始灰階影像以 $M \times N$ 不重疊之區塊進行劃分，接著在每一個不重疊的區塊中，依照取得的原始灰階值進行調整有序抖動矩陣，接著依相對位置與調整後的有序抖動矩陣之數值進行比較，如果其結果為小於，則將半色調的結果輸出前輸出色階 $g_{previous}$ ，反之則輸出後輸出色階 g_{next} 。詳細公式如下：

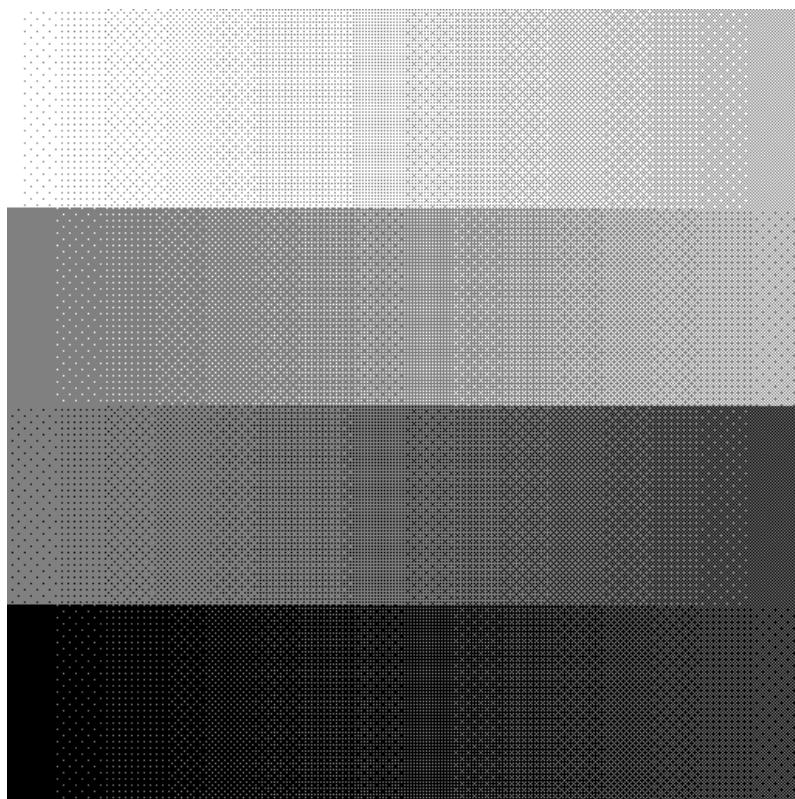
$$y(i,j) = \begin{cases} g_{previous}, & \text{if } x(i,j) \geq dm'(m,n) \\ g_{next}, & \text{otherwise} \end{cases}, \text{ where } m = i \bmod M, n = j \bmod N, \quad (4.5)$$

其中， $x(i,j)$ 表示一灰階值在原始灰階影像之 (i,j) 位置； $y(i,j)$ 表示透過判斷後所獲得的半色調影像； $dm(m,n)$ 表示原有的序抖動矩陣； $dm'(m,n)$ 表示目前映射(Mapping)到輸入影像 $x(i,j)$ 進行調整後的有序抖動矩陣； M 與 N 則分別表示有序抖動矩陣之高與寬。最後，將每一個不重疊之區塊完成前述動作就可以獲得完整多色調之影像。

圖4.4為採用表2.5(b)中的有序抖動矩陣所產生對應的三色階之數位多色調影像。觀察其結果可得知，有序抖動法本身會產生的不自然紋理圖樣依然存在於多色調影像中。多色調有序抖動法為減緩此種效應產生在輸出結果中，採用如同在半色調有序抖動法之延伸技術，藍雜訊遮罩有序抖動矩陣以改善現象，圖4.5、圖4.6為採用藍雜訊遮罩有序抖動矩陣所產生的對應三色階之數位多色調影像。然而，從圖4.6中可發現，此技術仍有帶狀效應的問題。



(a) 灰階影像 1 經多色調有序抖動法處理後的多色調影像

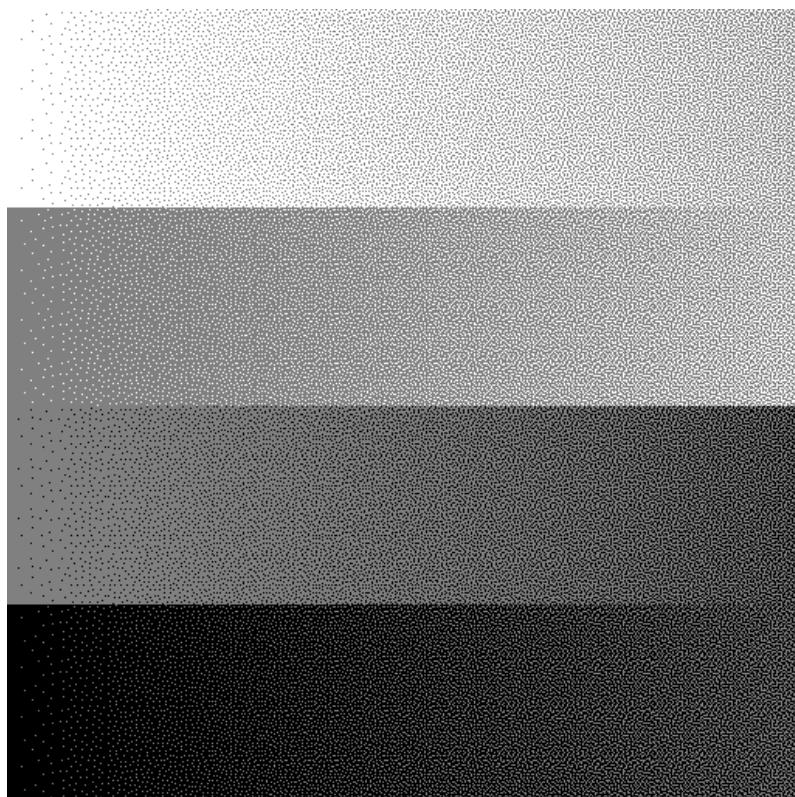


(b) 灰階影像 2 經多色調有序抖動法處理後的多色調影像

圖 4.4 使用 Bayer-5 有序抖動矩陣所產生的三色階多色調影像



(a) 灰階影像 1 經多色調有序抖動法處理後的多色調影像

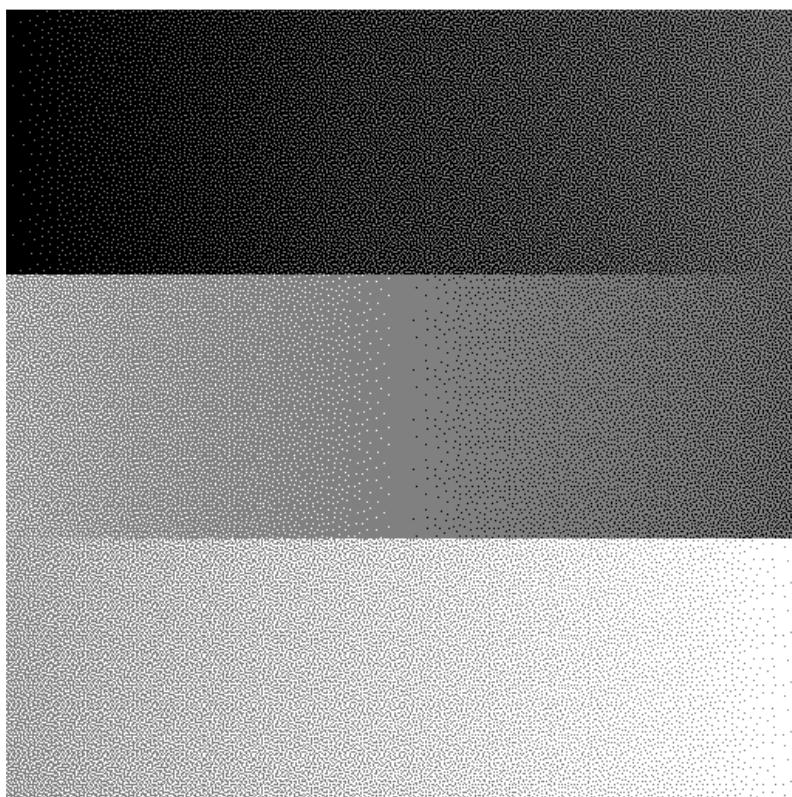


(b) 灰階影像 2 經多色調有序抖動法處理後的多色調影像

圖 4.5 使用藍雜訊遮罩有序抖動矩陣所產生的三色階多色調影像



(a) 灰階影像 3 經多色調有序抖動法處理後的多色調影像



(b) 灰階影像 4 經多色調有序抖動法處理後的多色調影像

圖 4.6 使用藍雜訊遮罩有序抖動矩陣所產生的三色階多色調影像

4.3 多色調直接二元搜尋法(Multitone direct binary search, M-DBS)

多色調直接二元搜尋法為 Lin 與 Allebach 在文獻[38]所提出，其技術為直接二元搜尋法(Direct binary search, DBS)的延伸，想法及概念幾乎相同，其初始公式如下：

$$e[m, n] = g[m, n] - y[m, n], \quad (4.6)$$

$$c_{pp}[m, n] = p(m, n) \otimes p(m, n), \quad (4.7)$$

$$c_{ep}[m, n] = e[m, n] \otimes c_{pp}[m, n], \quad (4.8)$$

$e[m, n]$ 、 $g[m, n]$ 及 $y[m, n]$ 分別表示誤差影像、原始灰階影像及初始多色調影像； $p(m, n)$ 為離散的點擴展函數(Point spread function, PSF)，此函數即是一個模擬人眼視覺系統(Human visual system, HVS)的模型； \otimes 表示摺積。接著，主要公式如下：

$$\Delta E = (a_0^2 + a_1^2)c_{pp}[0,0] + 2a_0c_{ep}[m_0, n_0] + 2a_1c_{ep}[m_1, n_1] + 2a_0a_1c_{pp}[m_1 - m_0, n_1 - n_0], \quad (4.9)$$

與直接二元搜尋法唯一的不同在於 a_0 及 a_1 的數值，翻點時的數值如下：

$$\begin{aligned} a_0 &= g_s - y[m - m_0, n - n_0], \\ a_1 &= 0 \end{aligned}, \quad (4.10)$$

交換時的數值如下：

$$\begin{aligned} a_0 &= y[m - m_1, n - n_1] - y[m - m_0, n - n_0] \\ a_1 &= y[m - m_0, n - n_0] - y[m - m_1, n - n_1] \end{aligned}, \quad (4.11)$$

其中， $g_s \in \mathbf{Q}$ 且輸出色階集合為 $\mathbf{Q} = \{g_1, g_2, \dots, g_q\}$ ，若以三色階為例，則為 $g_1 = 0$ 、 $g_2 = 0.5$ 、 $g_3 = 1$ 。上述 m_0 與 n_0 表示 $C_{ep}[m, n]$ 的中心位置， m_1 與 n_1 表示相對應中心點的交換位置，譬如與相對應目前處理像素點的 $[1, -1]$ 交換，因此 $m_1 = 1$ 與 $n_1 = -1$ 。上述成本函數公式 ΔE ，包含了翻點與交換等不同狀態。其中若 ΔE 的最小值小於 0，則代表該點分佈為最佳狀態，可更新 $y'[m, n]$ 與 $c'_{ep}[m, n]$ ，其更新公式可以寫成如下：

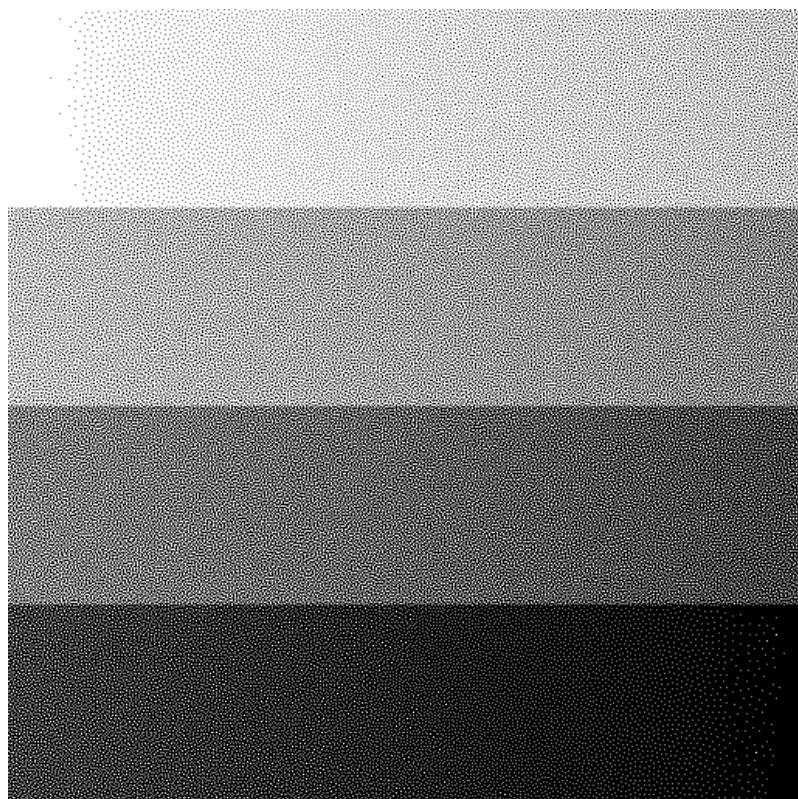
$$y'[m, n] = y[m, n] + a_0\delta[m - m_0, n - n_0] + a_1\delta[m - m_1, n - n_1], \quad (4.12)$$

$$c'_{ep}[m, n] = c_{ep}[m, n] + a_0c_{pp}[m - m_0, n - n_0] + a_1c_{pp}[m - m_1, n - n_1], \quad (4.13)$$

藉由上述公式重複不斷地提升影像品質，直到所有像素點的 ΔE 都不小於 0，即可完成多色調直接二元搜尋法，結果如圖 4.7 及圖 4.8。雖然產生的結果中幾乎沒有任何不自然的情況，但其計算複雜度過高，使直接二元搜尋在學術研究中僅作為最佳比較對象(Ground truth)。



(a) 灰階影像 1 經多色調直接二元搜尋法處理後的多色調影像

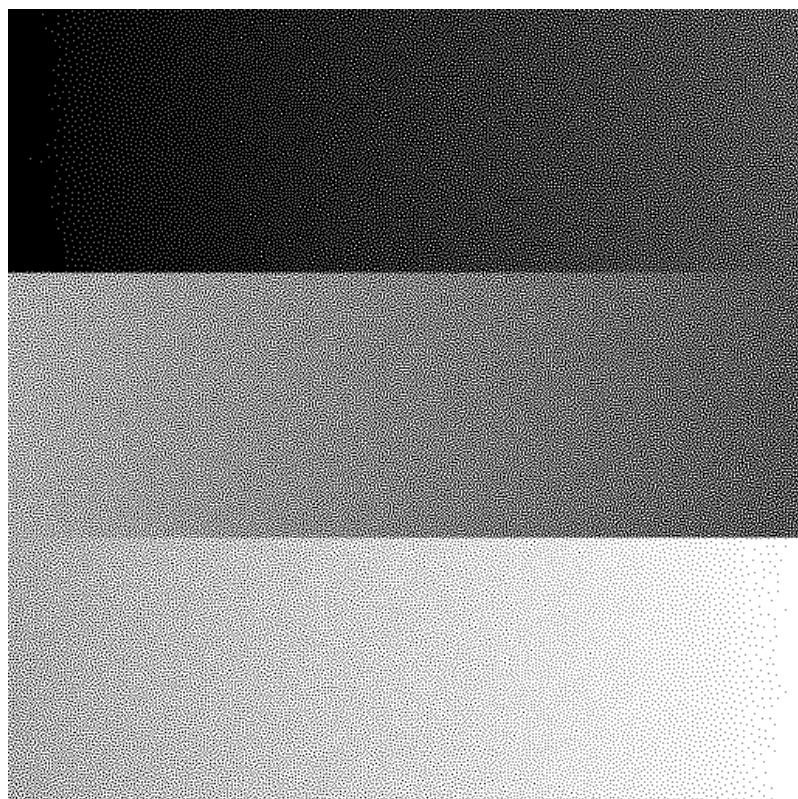


(b) 灰階影像 2 經多色調直接二元搜尋法處理後的多色調影像

圖 4.7 使用 7×7 高斯模糊作為點擴展函數所產生的三色階多色調影像



(a) 灰階影像 3 經多色調直接二元搜尋法處理後的多色調影像



(b) 灰階影像 4 經多色調直接二元搜尋法處理後的多色調影像

圖 4.8 使用 7×7 高斯模糊作為點擴展函數所產生的三色階多色調影像

4.4 多色調藍雜訊抖動法(Multiton blue-noise dithering, M-BND)

此方法為 Rodríguez 等人[39]在 2008 年所提出，其方法的目標與前面介紹的多色調直接二元搜尋法(Multiton direct binary search, M-DBS)相同，產生一個影像品質良好的多色調影像。Rodríguez 等人提出的概念是將一張原始影像依照墨水濃度(Ink concentration)分層為數張影像，接著依照所選定的半色調技術進行各自的轉換，最重要的是每一張分層影像所對應的半色調影像必須考慮半色調堆疊的規則，才能產生相較於多色調直接二元搜尋法更好的藍雜訊分佈。以下將假設墨水堆疊越多時將會越黑(與前述使用灰階值 0 表示黑色和使用 1 表示白色正好相反)。分層的公式如下：

$$S_j[gray] = \sum_{i=j}^q C_i[gray], \text{ where } j = 1, \dots, q, \text{ gray} = 1, \dots, L - 1, \quad (4.14)$$

$$X_i(m, n) = 255 \times (S_{i+1}[X(m, n)]), \text{ where } i = 1, \dots, q - 1 \quad (4.15)$$

L 表示灰階的數量，通常都設為 256； $C_i[gray]$ 表示統計在灰階 $gray$ 時所產生 i 色階的數量； $S_j[gray]$ 為大於等於 j 色階時所有 $C_i[gray]$ 的總和，又稱為累積墨水濃度曲線； q 為量化色階的數量； X_i 為原始影像 X 的子影像。圖 4.9 為分層式架構之示意圖，圖中 X 表示一張原始影像，接著將其分層成 $q - 1$ 張分層影像接著各自進行半色調轉換，並且對應的結果為 H ，接著將這數張半色調影像進行堆疊的動作以產生最後的多色調影像 M 。首先在此介紹半色調轉換時需考量的堆疊規則，

$$H_j(m, n) = 0, \text{ where } j = i + 1, \dots, q - 1, \text{ if } H_i(m, n) \text{ turns from "1" to "0"}, \quad (4.16)$$

$$H_j(m, n) = 1, \text{ where } j = 1, \dots, i - 1, \text{ if } H_i(m, n) \text{ turns from "0" to "1"}, \quad (4.17)$$

當半色調的子影像 $H_i(m, n)$ 因翻點/交換而從”1”變成”0”時， $H_j(m, n)$ 所有分層 $j \in [i + 1, q - 1]$ 在位置 (m, n) 的數值都要變為”0”；反之，當半色調的子影像 $H_i(m, n)$ 因翻點/交換而從”0”變成”1”時， $H_j(m, n)$ 所有分層 $j \in [1, i - 1]$ 在位置 (m, n) 的數值都要變為”1”。圖 4.10 為使用四色階多色調藍雜訊抖動法的範例，其半色調轉換時遵守了堆疊規則。接著介紹半色調轉換後如何堆疊成多色調影像，公式如下：

$$M(m, n) = \sum_{i=1}^{q-1} d_i H_i(m, n), \quad (4.18)$$

$$d_i = g_{i+1} - g_i, \text{ where } g_i = \frac{i-1}{q-1}, \text{ and } i = 1, \dots, q - 1, \quad (4.19)$$

M 為將各個半色調後的子影像堆疊而成的多色調結果； g_i 及 d 分別表示第*i*個量化色階的值及量化色階的間距。以圖 4.10 為例，在 A 點只有半色調子影像 H_1 中有墨水，而在 B 點時半色調子影像 H_1 及 H_2 都有墨水。因此，根據公式(4.18)及(4.19)可以將半色調子影像堆疊成多色調影像 $M(A) = \frac{1}{3}$ 及 $M(B) = \frac{2}{3}$ 。從中也能夠發現，圖 4.10 有遵守公式(4.16)及(4.17)的堆疊規則，在有半色調子影像中有墨水的點，比其更深的半色調子影像在相同的點上必有墨水。

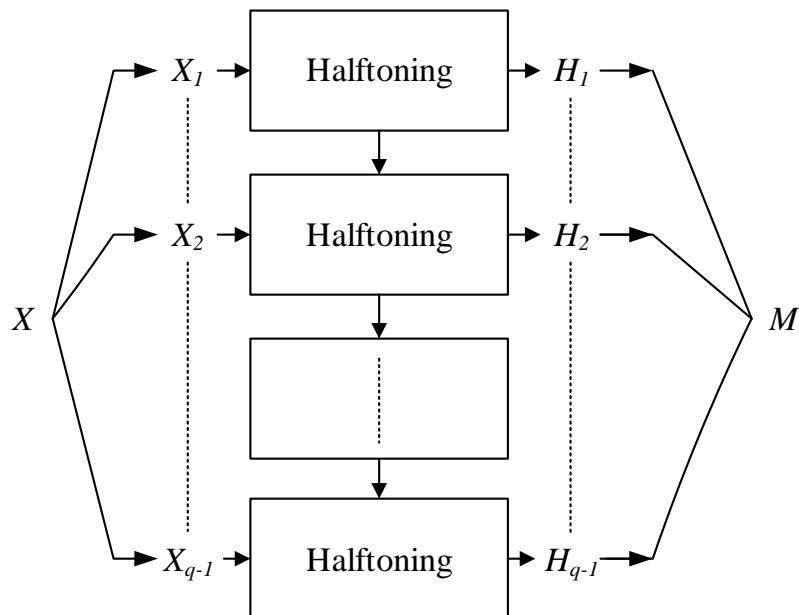


圖 4.9 分層式架構示意圖

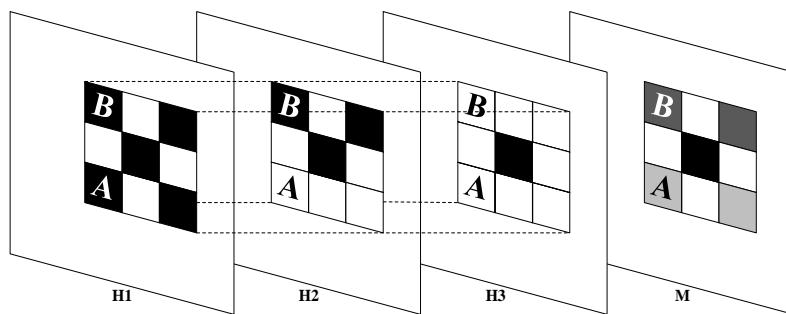
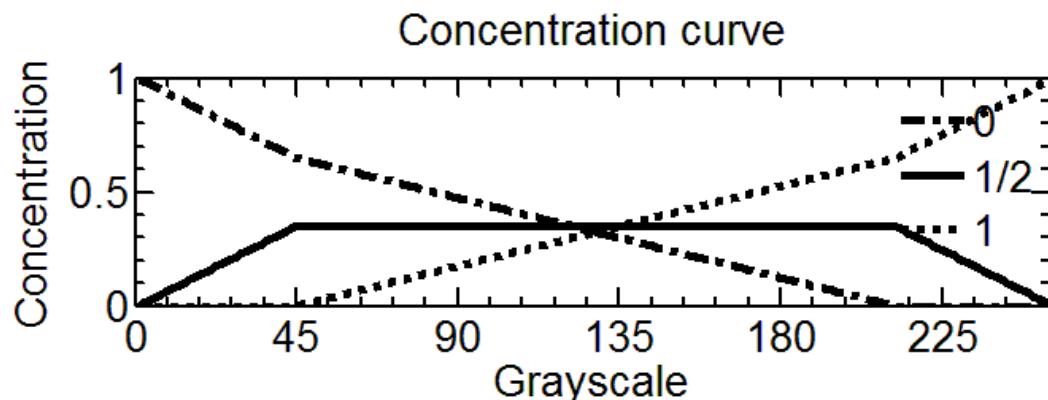
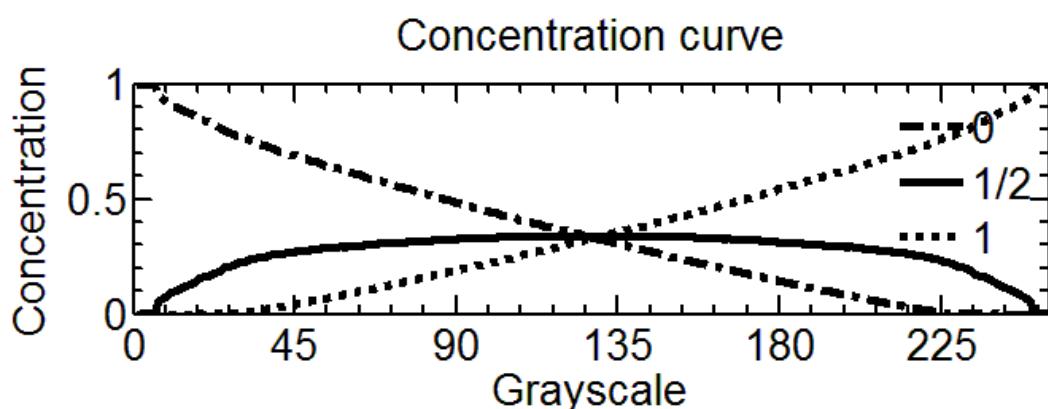


圖 4.10 四色階之多色調影像的堆疊範例

在介紹完分層影像與堆疊的概念後，接著要介紹如何進行影像的分層。圖 4.11 為對應的兩種墨水濃度曲線，圖中的三條曲線表示在不同的灰階值中三種墨水所佔的比例，這裡所說的三種墨水為 0(白色)、0.5(灰色)，1(黑色)。這樣的墨水濃度曲線是可以由使用者自行定義的，因此可以看到在圖 4.11(a)為 Rodríguez 等人的研究[39]中所提供之一種墨水濃度曲線，而圖 4.11(b)則是以統計單一灰階值影像所產生的多色調影像所得到的結果，其採用的是三色階多色調直接二元搜尋法。接著依照公式(4.14)可得到如圖 4.12 中 S2 與 S3 所示之結果。S2 與 S3 就是分層時所使用的依據如公式(4.15)，也可以說為一種對應的函數，當一個灰階值輸入時，將會對應到 S2 與 S3 所輸出比例值，因此這一個灰階值將會乘上比例值並且分配到對應的分層影像。以圖 4.12(a)為範例，當灰階值為 85 時，對應的 S2 與 S3 的輸出比例值為 0.5723 與 0.0948，因此在第一層分層影像所分配的結果為 $255 \times 0.5723 \cong 146$ ，而第二層分層影像所分配的結果為 $255 \times 0.0948 \cong 24$ 。圖 4.13 與圖 4.14 為使用三色階進行分層所產生的兩張對應的分層影像與對應的半色調影像，圖 4.15 為使用圖 4.12(a)的累積墨水濃度曲線所產生的數位多色調影像，其中使用圖 4.15(c)為使用圖 4.13(b)與圖 4.14(b)堆疊後所產生的多色調影像，而圖 4.16 為使用圖 4.12(b)的累積墨水濃度曲線所產生的數位多色調影像。結果中可以看到，此方法產生的多色調影像其藍雜訊分佈特性是比多色調直接二元搜尋法是更好的。且從圖 4.15 及圖 4.16 中可以發現，墨水濃度曲線會明顯地影響最後產生的結果，但在這篇研究中並沒有討論如何取得最佳的墨水濃度。此外，由於是基於直接二元搜尋法，因此處理時間過長造成實用性較低。



(a) Rodríguez 文獻[39]所提供之結果



(b) 從 M-DBS 統計出來的結果

圖 4.11 兩種不同的墨水濃度表

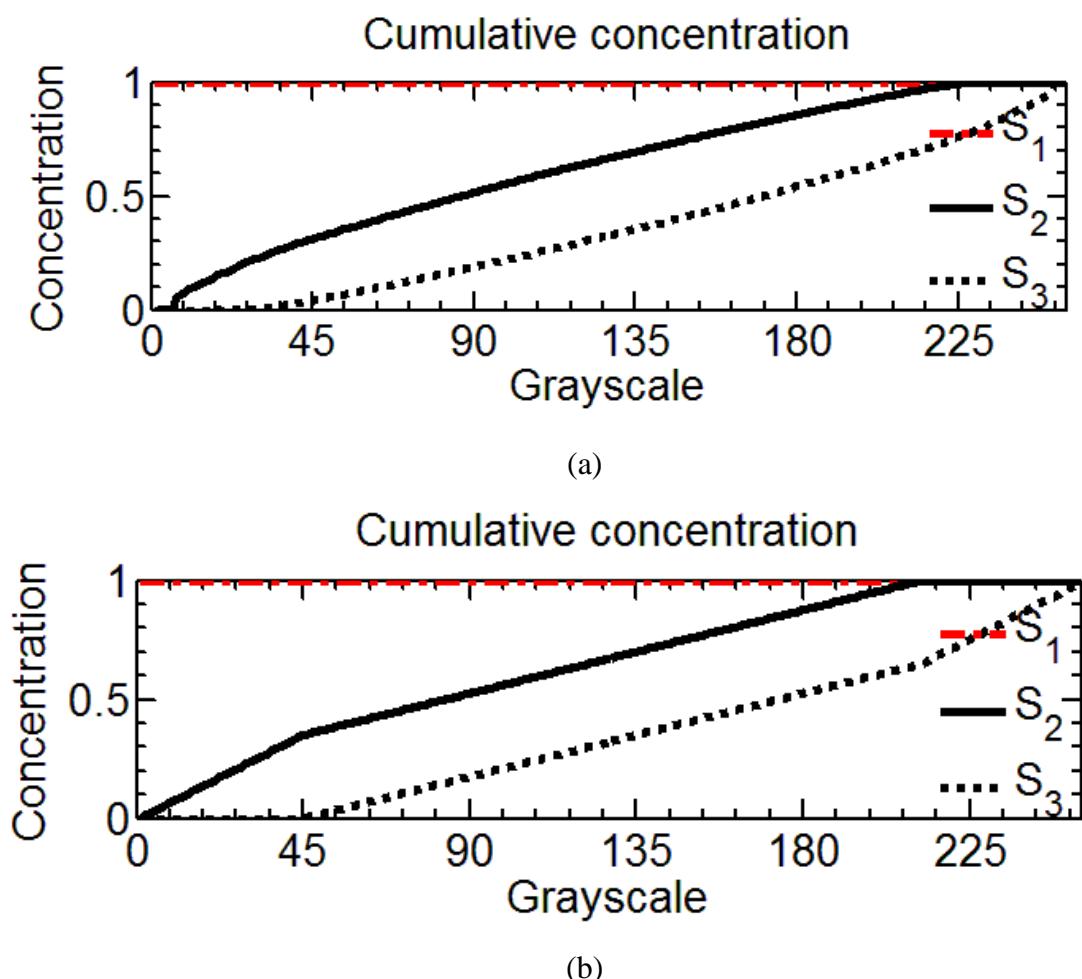


圖 4.12 對應圖 4.11 的墨水濃度累積曲線



(a) 第一層分層影像



(b) 第一層分層影像的半色調影像

圖 4.13 以圖 4.1(a)當作範例之第一層分層影像



(a) 第二層分層影像

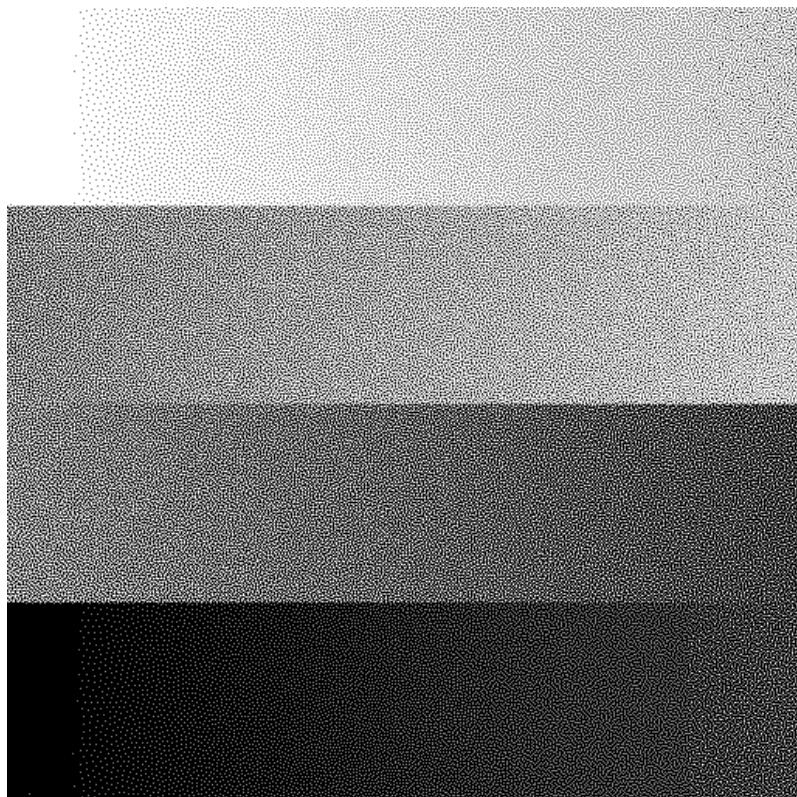


(b) 第二層分層影像的半色調影像

圖 4.14 以圖 4.1(a)當作範例之第二層分層影像



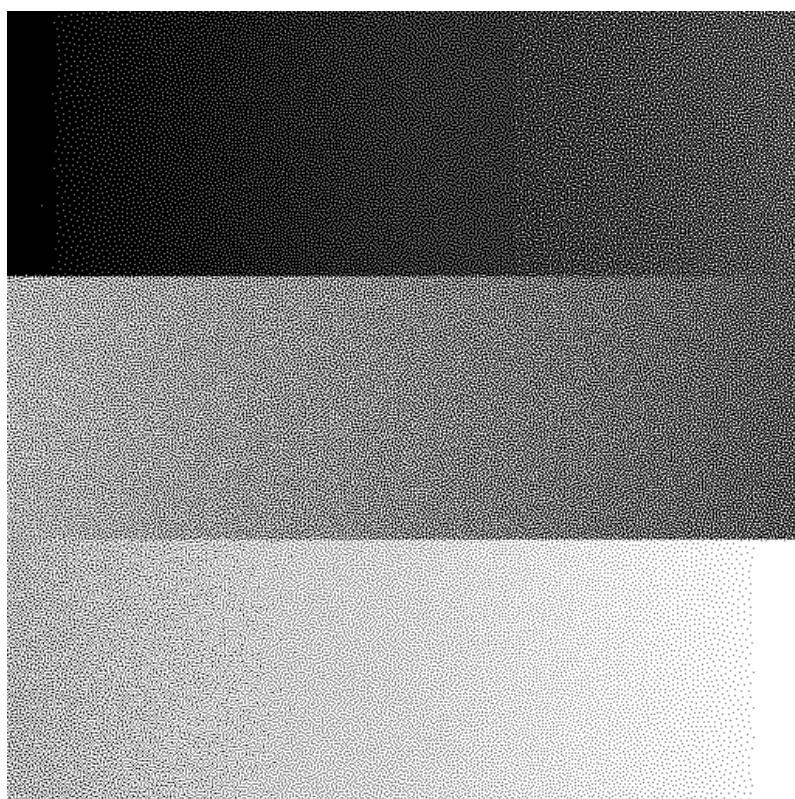
(a) 灰階影像 1 經多色調藍雜訊抖動法處理後的多色調影像



(b) 灰階影像 2 經多色調藍雜訊抖動法處理後的多色調影像



(c) 灰階影像 3 經多色調藍雜訊抖動法處理後的多色調影像

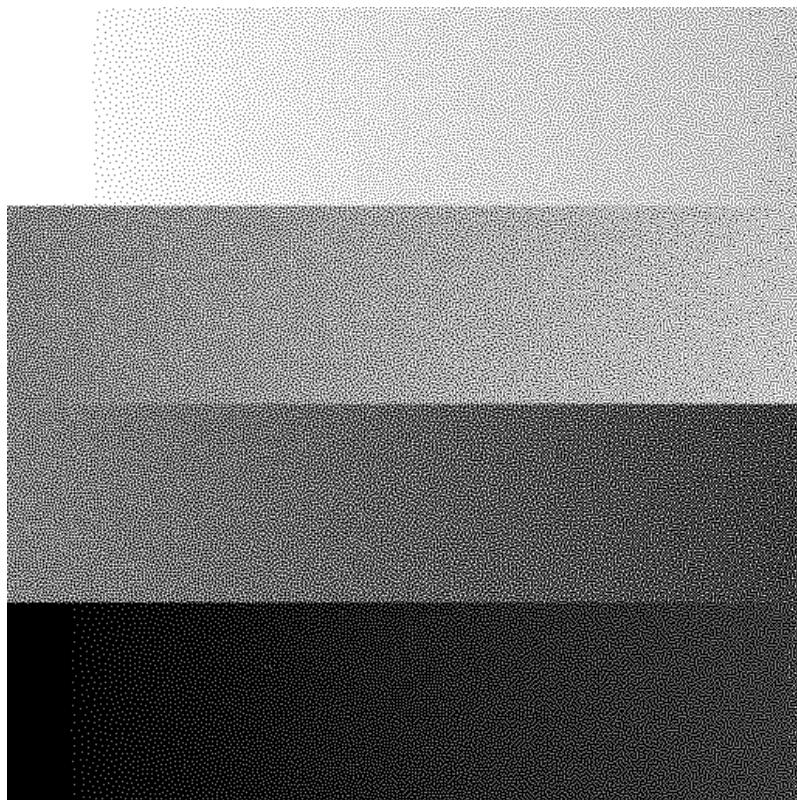


(d) 灰階影像 4 經多色調藍雜訊抖動法處理後的多色調影像

圖 4.15 使用圖 4.12 (a)的累積墨水濃度曲線所產生的三色階多色調影像



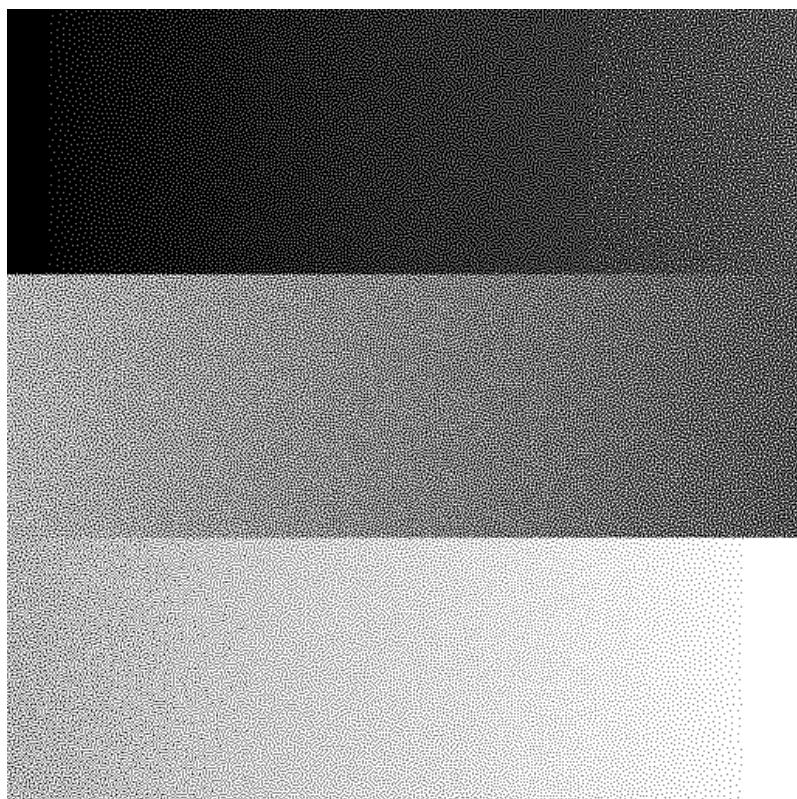
(a) 灰階影像 1 經多色調藍雜訊抖動法處理後的多色調影像



(b) 灰階影像 2 經多色調藍雜訊抖動法處理後的多色調影像



(c) 灰階影像 3 經多色調藍雜訊抖動法處理後的多色調影像



(d) 灰階影像 4 經多色調藍雜訊抖動法處理後的多色調影像

圖 4.16 使用圖 4.12 (b)的累積墨水濃度曲線所產生的三色階多色調影像

4.5 直接多位元搜尋法(Direct multi-bit search, DMS)

直接多位元搜尋法為 Chandu 及 Stanich 等人[40]於 2012 年所提出，為直接二元搜尋法(Direct binary search, DBS)的延伸技術。初始公式如下：

$$e[m, n] = g[m, n] - y[m, n], \quad (4.20)$$

$$c_{pp}[m, n] = k_1 \exp\left(-\frac{u^2+v^2}{2\sigma_1^2}\right) + k_2 \exp\left(-\frac{u^2+v^2}{2\sigma_2^2}\right), \quad (4.21)$$

$$c_{ep}[m, n] = e[m, n] \otimes c_{pp}[m, n], \quad (4.22)$$

$e[m, n]$ 、 $g[m, n]$ 及 $y[m, n]$ 分別表示誤差影像、原始灰階影像及初始多色調影像； \otimes 表示摺積；這裡使用的 c_{pp} 與公式(4.9)不同，其為 Kim 及 Allebach 於文獻[16]所提出的混合高斯模型，參數($k_1, k_2, \sigma_1, \sigma_2$)設為(43.2, 38.7, 0.0219, 0.0598)。接著，主要公式如下：

$$\Delta E = (a_0^2 + a_1^2)c_p[0,0] + 2a_0c_{ep}[m_0, n_0] + 2a_1c_{ep}[m_1, n_1] + 2a_0a_1c_{pp}[m_1 - m_0, n_1 - n_0], \quad (4.23)$$

比較特別的是其 a_0 及 a_1 的數值，翻點時的數值如下：

$$\begin{aligned} a_0 &= g_s - y[m - m_0, n - n_0], \\ a_1 &= 0 \end{aligned}, \quad (4.24)$$

交換時的數值如下：

$$\begin{aligned} a_0 &= y[m - m_1, n - n_1] - y[m - m_0, n - n_0] \\ a_1 &= y[m - m_0, n - n_0] - y[m - m_1, n - n_1] \end{aligned}, \quad (4.25)$$

其中， $g_s \in Q$ 且輸出色階集合為 $Q = \{g_1, g_2, \dots, g_q\}$ 。上述 m_0 與 n_0 表示 $C_{ep}[m, n]$ 的中心位置， m_1 與 n_1 表示相對應中心點的交換位置，譬如與相對應目前處理像素點的 [1, -1] 交換，因此 $m_1 = 1$ 與 $n_1 = -1$ 。上述成本函數公式 ΔE ，包含了翻點與交換等不同狀態。其中若 ΔE 的最小值小於 0，即是代表該點分佈為最佳狀態，因此即可更新 $y'[m, n]$ 與 $c'_{ep}[m, n]$ ，其更新公式可以寫成如下：

$$y'[m, n] = y[m, n] + a_0\delta[m - m_0, n - n_0] + a_1\delta[m - m_1, n - n_1], \quad (4.26)$$

$$c'_{ep}[m, n] = c_{ep}[m, n] + a_0c_{pp}[m - m_0, n - n_0] + a_1c_{pp}[m - m_1, n - n_1], \quad (4.27)$$

藉由上述公式重複不斷地提升影像品質，直到所有像素點的 ΔE 都不小於 0，即可完成直接多位元搜尋法，結果如圖 4.17 及圖 4.18。

其基礎結構與多色調直接二元搜尋法(M-DBS)類似，不同的地方在於其 c_{pp} 及翻點與交換時 a_0 、 a_1 的數值。但這樣的結果會與多色調直接二元搜尋法一樣有著極高的計算複雜度，因此在此論文中引用文獻[41]所提出的篩選法(Screening)。首先，篩選法會需要臨界值矩陣，其紀錄半色調輸出結果開啟(Turn-on)時的灰階，產生的方法如下：

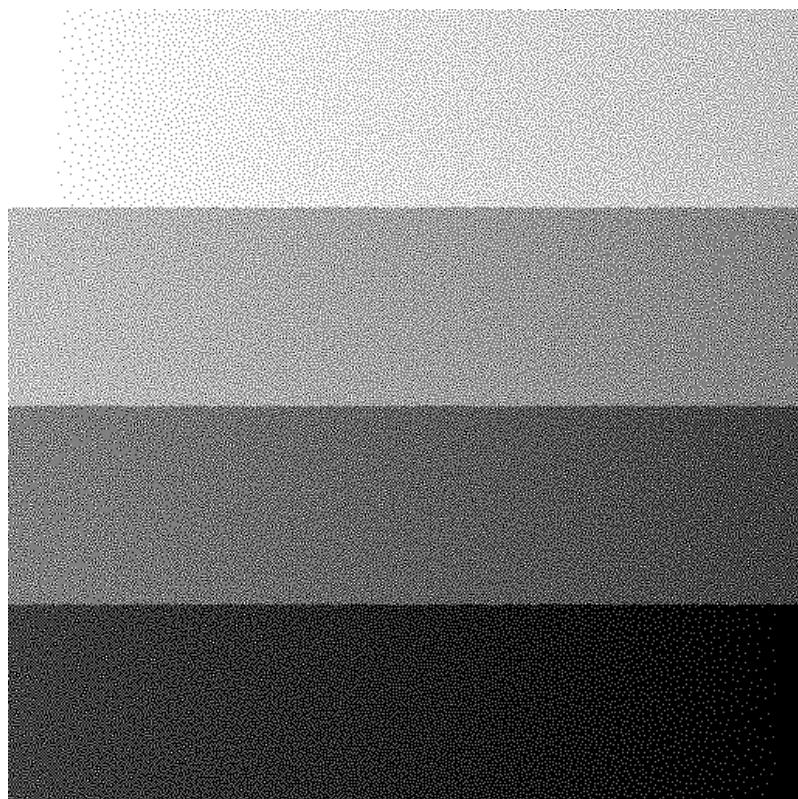
1. 產生 L 張 $U \times V$ 大小的連續調影像作為輸入影像，每張影像為純色灰階 $x_{gray}(u, v) = \frac{gray}{L-1}$ ，一般影像的灰階為 8 bits，因此 L 通常為 256。
2. 使用前面所述的多色調技術並依照堆疊法則(Stacking constraint)產生輸出多色調影像 $y_j(u, v)$ 。堆疊法則為當位置 (u, v) 量化灰階為 j 的多色調輸出結果 $y_j(u, v) = g_s$ ，則相同位置且量化灰階 $i \geq j$ 的多色調結果必為 $y_i(u, v) \geq g_s$ ，相同位置且灰階 $i < j$ 的多色調結果則為 $y_i(u, v) < g_s$ 。
3. 臨界值矩陣 $t_l(u, v)$ 產生的方法是紀錄步驟 2 所產生的多色調其從 g_s 變 g_{s+1} 時的灰階值 $gray \in [0, 255]$ 。當有 q 個量化色階，則會有 $q - 1$ 個臨界值矩陣，因此 $l \in [1, q - 1]$ 。舉例來說，若為三色階 $Q = \{g_1, g_2, g_3\}$ ，則會有兩個臨界值矩陣， $t_1(u, v)$ 紀錄多色調影像 $y(u, v)$ 量化色階 g_1 變 g_2 時的灰階； $t_2(u, v)$ 則紀錄多色調影像 $y(u, v)$ 量化色階 g_2 變 g_3 時的灰階。

最後，篩選法藉由上述產生的臨界值矩陣可以快速地產生多色調影像 y ，

$$y(m, n) = \begin{cases} g_{l+1}, & \text{if } x(m, n) \geq t_l(u, v) \\ g_1, & \text{otherwise} \end{cases}, \text{ where } u = m \bmod U, v = n \bmod V, \quad (4.28)$$



(a) 灰階影像 1 經直接多位元搜尋法處理後的多色調影像

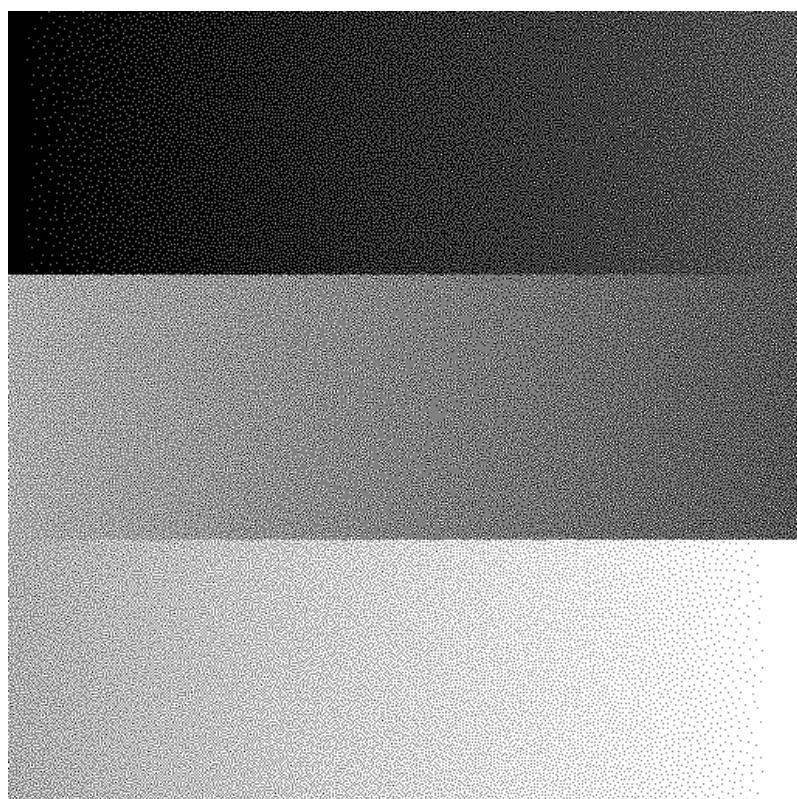


(b) 灰階影像 2 經直接多位元搜尋法處理後的多色調影像

圖 4.17 直接多位元搜尋法所產生的三色階多色調影像



(a) 灰階影像 3 經直接多位元搜尋法處理後的多色調影像



(b) 灰階影像 4 經直接多位元搜尋法處理後的多色調影像

圖 4.18 直接多位元搜尋法所產生的三色階多色調影像

4.6 色調替代之誤差擴散法(Multitone tone-replacement ED, M-TRED)

為了有效地消除帶狀效應，Guo 及 Chang [42]於 2015 年提出色調替代之誤差擴散法，其結構是基於多色調誤差擴散法(Multitone error-diffused, M-ED)，詳細流程如圖 4.19。為了避免誤差擴散法中不自然的紋理圖案，色調替代之誤差擴散法運行的方式是採蛇行掃描(Serpentine scan)的方式如圖 2.9 (b)。根據圖 4.19 的流程圖，其中色調替代之程序(Tone-replacement process)運作流程將分為以下三個步驟介紹：

1. 進入到色調替代之程序(Tone-replacement process)的第一步，將搜尋目前輸入的數值 $c(i, j)$ 與輸出色階集合 \mathbf{Q} 中哪一個輸出色階 g_n 有最小的絕對誤差，而擁有與 $c(i, j)$ 最小絕對誤差的輸出色階將被定義為 g_s ，其定義如下列公式所示，

$$g_s = \arg \min_{g_n \in \mathbf{Q}} |c(i, j) - g_n|, \quad (4.29)$$

其中輸出色階集合 \mathbf{Q} 的定義是與第四章數位多色調技術文獻探討中開頭的定義是相同的，假設目前為 q 色階之多色調，則 $\mathbf{Q} = \{g_1, g_2 \dots g_q\}$ 。

2. 接著進入到圖 4.19 中色調替代開關(Tone-replacement switch)的部分，開關的開起與關閉的條件是使用下列公式表示，

$$\text{switch} = \begin{cases} \text{On}, & bn(i', j') = 1 \wedge \text{RAND}(0,1) \leq p \wedge g_s \notin \{g_1, g_q\}, \\ \text{Off}, & \text{O.W.} \end{cases} \quad (4.30)$$

其中 $bn(i', j')$ 為藍雜訊遮罩圖樣，如圖 4.20 所示； p 為調整色調替代機率，在此研究中設為 0.65。根據此公式，當色調替代開關是處於開啟(on)狀態，則表示要進行色調替代並進入至步驟 3，反之處於關閉(off)狀態則維持原本的多色調誤差擴散法所輸出之結果，也就是將 g_s 輸出至 $y(i, j)$ 並進行誤差擴散，不進入至步驟 3。色調替代開關的開啟條件中 $g' \notin \{g_1, g_q\}$ ，此部分是因為當 g_s 等於 g_1 或 g_q 時，將會缺少上一個輸出色階 g_{s-1} 或下一個輸入色階 g_{s+1} ，色調替代只會用另一個色階進行替代，將無法進行誤差補償，因此將此情況排除。舉例來說當 $g_s = g_1$ ，則進行色階取代的結果將只會有 $g'_s = g_1$ ，當要進行補償時原本是要用 g_{s-1} 進行補償，但是 g_{s-1} 的色階卻是不存在，因此會造成較大的誤差造成降低影像品質。 $bn(i', j')$ 表示藍雜訊遮罩圖樣的大小可能與輸入影像大小不同，因此 $i' = i \bmod M$ 、 $j' =$

$j \bmod N$ ，其中 M 與 N 分別表示輸入影像的高跟寬。

3. 首先產生另一個輸出色階集合 $\mathbf{Q}' = \mathbf{Q} - \{g_s\}$ ，也就是將擁有最小誤差之輸出色階從輸出色階集合中移除。接著在這一個輸出色階集合 \mathbf{Q}' 搜尋與 $c(i, j)$ 擁有最小誤差值之輸出色階，並且定義為 g'_s 。

$$y(i, j) = g'_s = \arg \min_{g_n \in \mathbf{Q}'} |c(i, j) - g_n|, \quad (4.31)$$

此步驟也可以視為是搜尋第二最小誤差值之輸出色階。找到 g'_s 後將其輸出為最後結果 $y(i, j)$ ，並且進行誤差擴散，在此的誤差擴散權重經訓練結果如表 4.1。

以上三個步驟為一個像素點的處理流程，因此只要將整張影像帶入此步驟，並搭配圖 2.9 (b) 之處理路徑將可以產生對應的多色調影像，結果如圖 4.21 及圖 4.22。

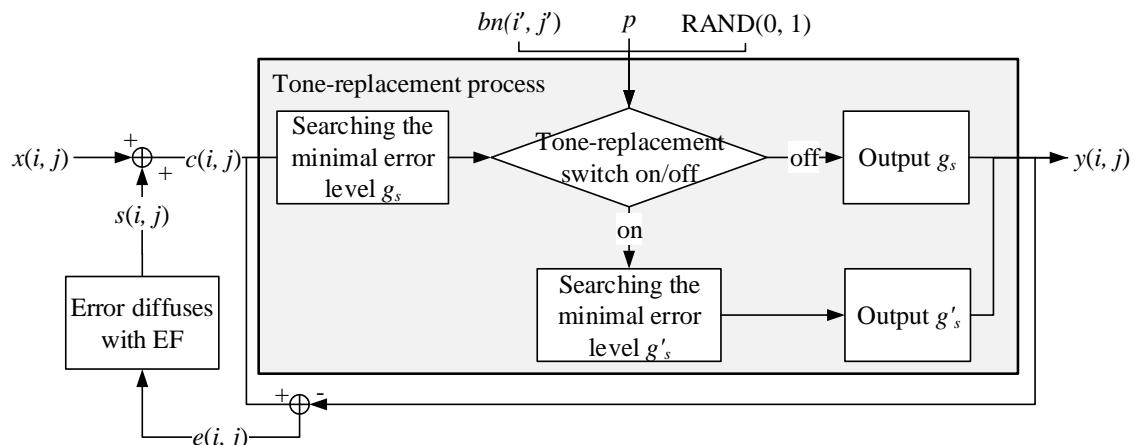


圖 4.19 色調替代之誤差擴散法之流程圖

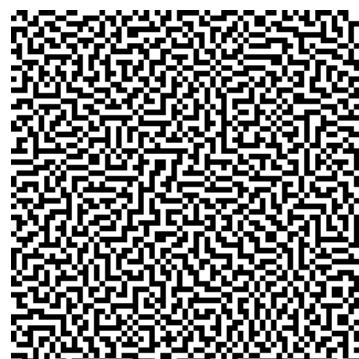


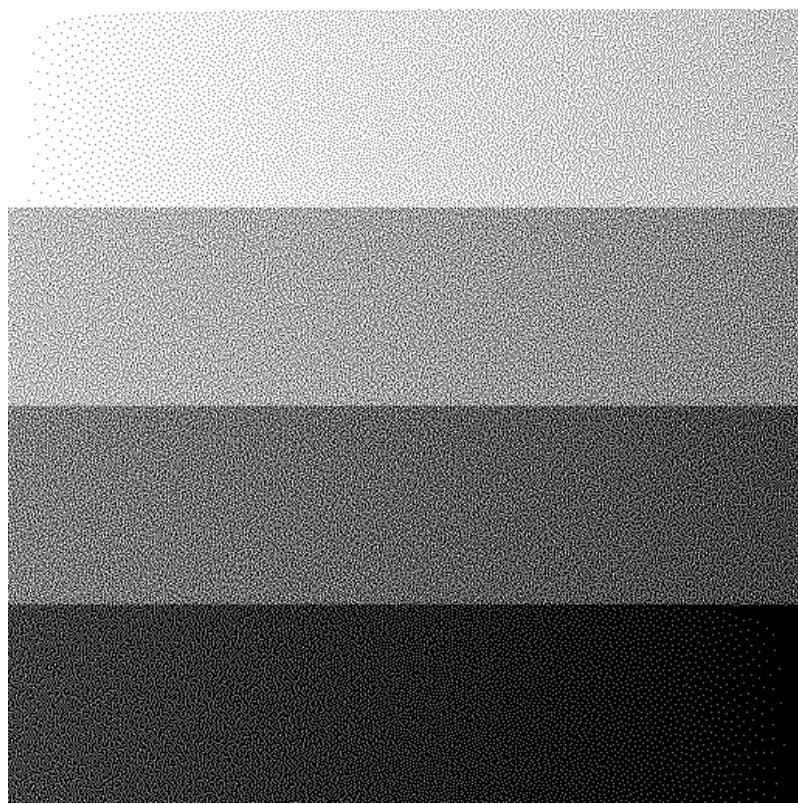
圖 4.20 藍雜訊遮罩之範例

表 4.1 文獻[42]在 $p = 0.65$ 下提供的不同色階的最佳化誤差濾波器

	x	0.4955		x	0.5136
0.2666	0.2379	0	0.2560	0.2304	0
	3-level			4-level	
	x	0.5218		x	0.5227
0.2612	0.2168	0.0002	0.2629	0.2144	0
	5-level			6-level	

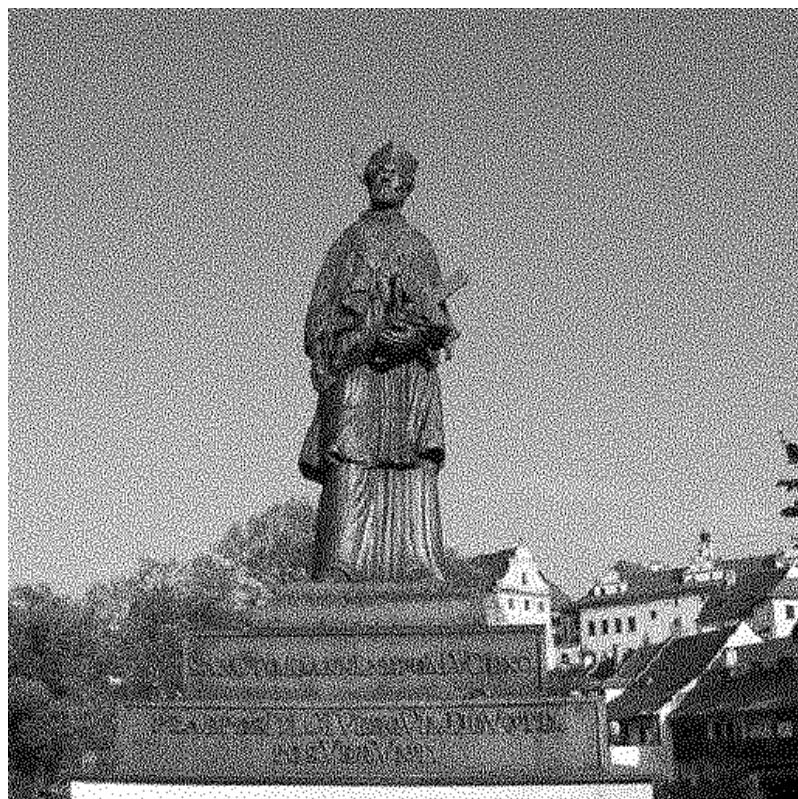


(a) 灰階影像 1 經色調替代之誤差擴散法處理後的多色調影像

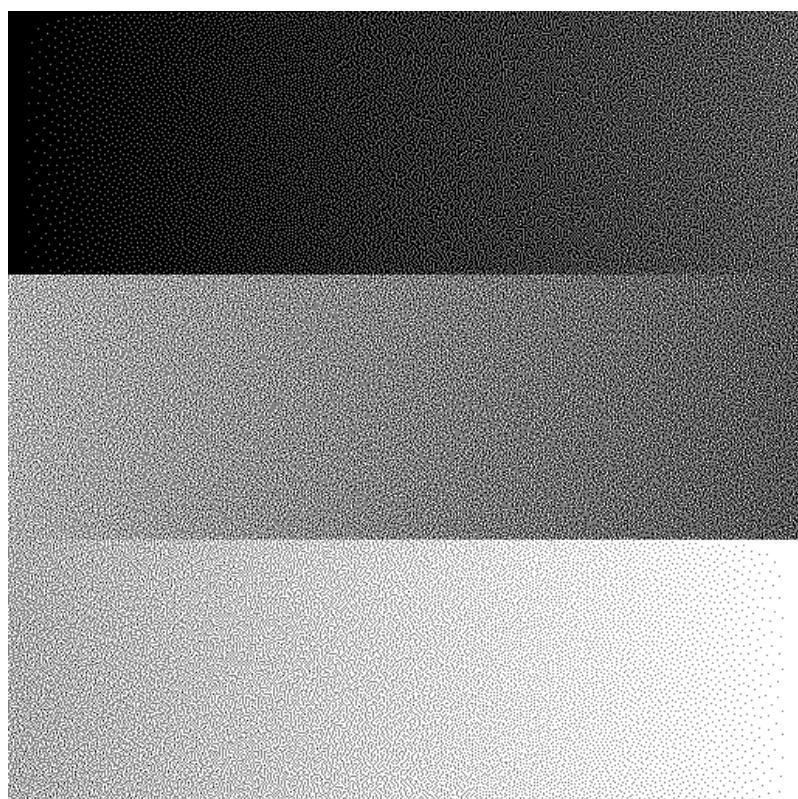


(b) 灰階影像 2 經色調替代之誤差擴散法處理後的多色調影像

圖 4.21 色調替代之誤差擴散法所產生的三色階多色調影像



(c) 灰階影像 3 經色調替代之誤差擴散法處理後的多色調影像



(d) 灰階影像 4 經色調替代之誤差擴散法處理後的多色調影像

圖 4.22 色調替代之誤差擴散法所產生的三色階多色調影像

第五章 自適應點擴散區塊截斷編碼技術

在此章節，一個新的壓縮技術名為自適應點擴散區塊截斷編碼(Self-adaptive dot-diffused block truncation coding, SDBTC)被提出，其目的是為了產生更自然的重建影像以及在不破壞品質的情況下提升壓縮率。為了第一個目標，一個新的演算法改善過往量化色階的問題，平衡了區塊效應(Blocking effect)及脈衝雜訊(Impulse noise)的產生。第二個目標是藉由一個預先訓練好的參數，可以使使用者得到在其期望的影像品質下最高的壓縮率。此外，對於彩色影像壓縮，由於人眼視覺對於彩色通道的敏感度較低，藉由縮減取樣(Downsample)的手法獲得與過往使用單一位元圖技術(SBM)同樣高的壓縮率，並且有著更佳的影像品質。本論文所提出的方法對各通道的處理流程如圖 5.1 所示，是以 3.5 節介紹的點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)為基礎做改良，改善的部分將以灰色表示。在接下來的小節中將介紹本論文提出的改良部分，最後將會統整說明整個系統處理的步驟，並且在實驗結果中將會詳細比較各種方法之效能。

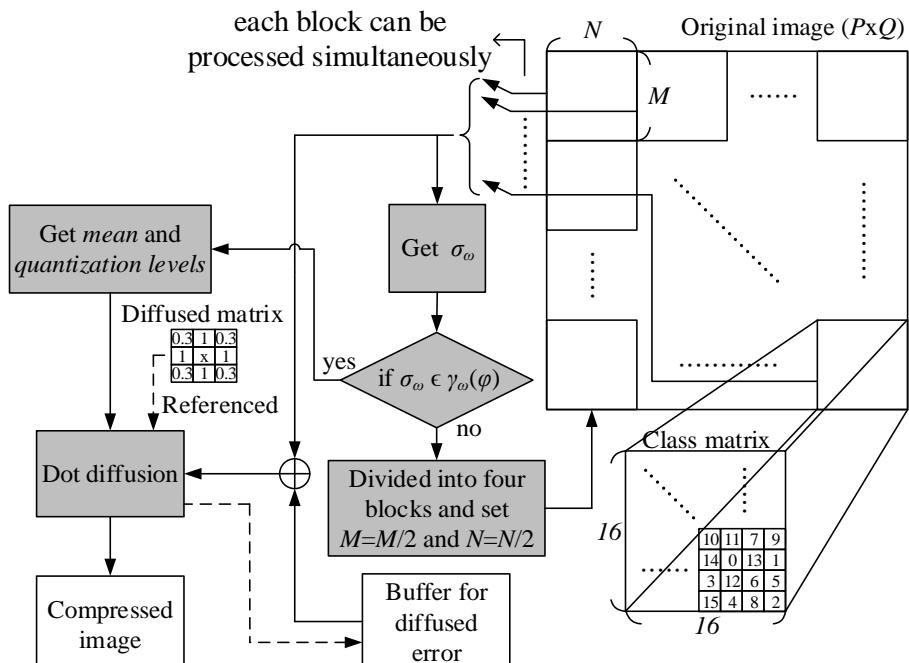


圖 5.1 自適應點擴散區塊截斷編碼壓縮技術對各通道的流程圖

5.1 彩色通道分析

過往彩色區塊截斷編碼技術，對彩色影像的處理大多在 RGB 色彩模型上。R、G 及 B 通道之間的關聯性極高，因此如 3.7 節的單一位元圖區塊截斷編碼(Single bit-map BTC, SBMBTC)，能將三通道的位元圖(Bitmap)合而為一，雖然能夠有極高的壓縮率，但由於三通道使用相同的位元圖重建影像，導致原有的區塊效應(Blocking effect)更加明顯，結果就如圖 5.2 所示。由於人眼對於亮度差異的敏感度高於色彩的變化，將亮度與色彩分成不同的通道表示更能提升壓縮率，因此我們將對 RGB 以外常見的幾個色彩模型分析並判斷。最常被使用且更符合人眼感知的色彩模型是 HSV，其三個分量為色相(Hue, H)、飽和度(Saturation, S)及明度(Value, V)。但區塊截斷編碼技術並不適合用在 HSV 色彩模型上，因為彩色影像的色彩僅被一個通道(色相)控制，這會導致壓縮後的影像嚴重色彩失真。而另一個最接近人眼視覺的色彩模型為 CIE Lab，Lab 色彩模型為顏色對立空間，其 L 通道表示亮度，而 a 及 b 通道表示顏色對立維度。因此，可以藉由調整 L 通道來改善影像的明亮度，或者修改 a、b 通道做色平衡校正，而 RGB 是建模於物理設備的輸出，所以無法達成上述精細的調整。此外，雖然 YCbCr 色彩模型有相同的效用，但 Lab 與人眼的感受更貼近，這表示當相同量的色彩變化會產生相同的人眼感受度。因此我們在改良技術中，對彩色影像壓縮時選擇將影像從 RGB 轉換成 Lab 再做處理。

由於人眼視覺系統(Human visual system, HVS) [43]對於 a、b 分量色彩變化的敏感度較低，因此能將這兩個分量藉由高斯金字塔(Gaussian pyramid)做空間的縮小/放大，又稱縮減取樣(Downsample)/增加取樣(Upsample)。在壓縮時，將 a、b 分量縮小就能得到更高的壓縮率，並在重建影像時將 a、b 分量放大還原成原始影像的大小，接著轉換回 RGB 色彩模型就能夠得到沒有明顯失真的影像。a、b 分量縮小/放大的比例藉由參數 $\delta \in \mathbb{R}$ 控制，可表示為 $\tilde{P} = P/\delta$ 及 $\tilde{Q} = Q/\delta$ ，其中 $P \times Q$ 表示原始影像的大小，此參數詳細的影響將會在後續的參數設定中討論及說明。對於彩色影像的壓縮，完整的自適應點擴散區塊截斷編碼流程圖如圖 5.3 所示，其中各通道的編碼流程如同圖 5.1。



(a) Original image

(b) Former method

圖 5.2 使用單一位元圖技術(SBM)產生嚴重的區塊效應

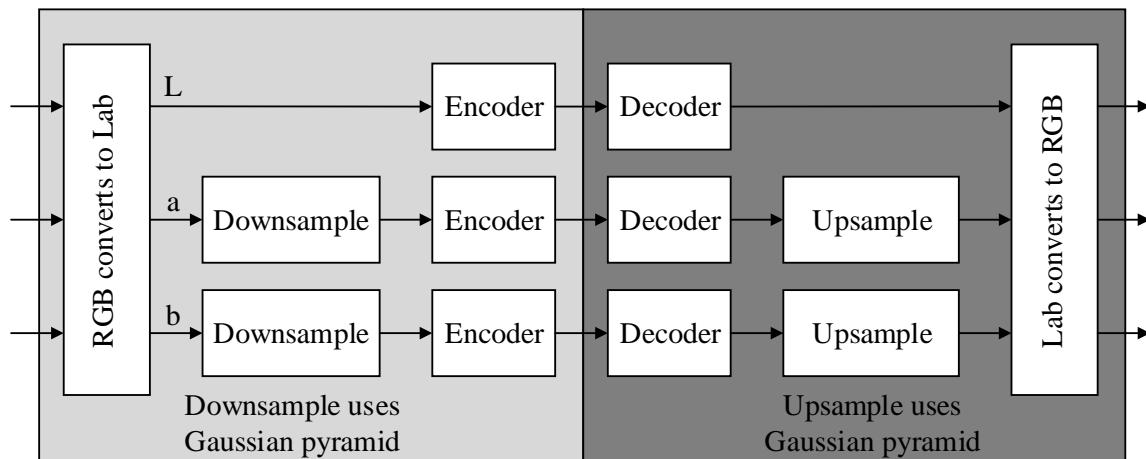


圖 5.3 自適應點擴散區塊截斷編碼壓縮技術對彩色影像的流程圖

5.2 自適應策略(Adaptive strategy)

關於區塊截斷編碼的解碼，典型的方式是每個固定大小的區塊都使用兩個量化色階(Quantization level)並依照位元圖(Bitmap)的分佈重建壓縮影像。儘管以少量的顏色足夠重建一個平滑的區域，但明顯地這不足以完整的描繪一個色彩複雜的區域。顯然地，這是一個介於影像品質及壓縮率(Compression ratio, CR)之間的權衡問題，需要考量是否增加區域的量化色階。為了解決這個問題，我們主要的想法是將色彩複雜的區域更進一步地分割成多個子區域(Sub-region)，以便在每個子區域用有限的量化色階產生最佳的結果。b-ary entropy (H_n)能夠衡量一個給定資料的不確定性，它受制於

$$H_n(p_1, \dots, p_n) \leq \log_b(n), \quad (5.1)$$

其中 p_i 表示第 i 個像素值出現的機率， $i \in [1, n]$ ； b 表示理想不同像素值的數量。這個公式定義了資料不確定性的上限，這表明一子集像素的變化量是正比於 \log 的子集大小。與此同時，自然影像具有一個獨特的性質，就是兩個像素的相似度反比於它們的空間距離。可以從上述這兩種現象歸納出，區塊的大小限制了區塊資訊其頻率的上限。為了評估這個特性，標準差在這邊使用且其公式如下定義

$$\sigma_\omega = \sqrt{\frac{1}{M \times N} \sum_{(m,n)} (x(m, n) - \bar{x})^2}, \quad (5.2)$$

其中 $x(m, n)$ 表示區塊內的像素位置； \bar{x} 為區塊的灰階平均值； ω 是簡化後表示區塊的長度或者寬度。圖 5.4 以公式(5.2)計算多個不同大小區塊的平均標準差，可以發現區塊越小平均標準差越低，其展示出如公式(5.1)的特性。在圖 5.4 的測試當中，使用了包含 1,000 張大小 256×384 的自然影像資料庫 Corel 1,000 [44]。

為了對不同空間複雜度的區域做分類，一個區塊區域分解演算法(Block region decomposition algorithm)在此被提出，其概念如同圖 5.1 中間上方灰色標示的三個步驟流程：

1. 將大小為 $P \times Q$ 的影像切割為多個不重疊的 $\omega \times \omega$ 區塊，初始 ω 設為 16。
2. 若區塊的標準差(Standard deviation, σ_ω)沒有落在設定的 $\gamma_\omega(\varphi)$ 範圍內，則將區塊切割成四個相同 $\frac{\omega}{2} \times \frac{\omega}{2}$ 大小的子區塊。重複上述動作直到不能再切割(區塊最小為

2×2)或者標準差有落在設定的 $\gamma_\omega(\varphi)$ 範圍內。

3. 藉由後續會介紹的改良方法壓縮各個區塊。

其中， $\gamma_\omega(\varphi)$ 指的是使用者期望的品質 φ 在區塊大小為 $\omega \times \omega$ 時所限定的標準差範圍，詳細的說明將會在後續章節中討論。

值得注意的是，從上述步驟中可以發現到，同一個區域可能會很密集的重複計算區域內的標準差 σ_ω ，因而阻礙了壓縮演算法的效率。為了解決這個問題，以積分影像為基礎的計算方法在文獻[45]中被提出，藉由事先運算完成的一階及二階積分影像就可以極低的計算複雜度完成各個區塊標準差的計算。除此之外，後續會使用到的區塊灰階平均值 \bar{x} 也能以一階積分影像去計算。

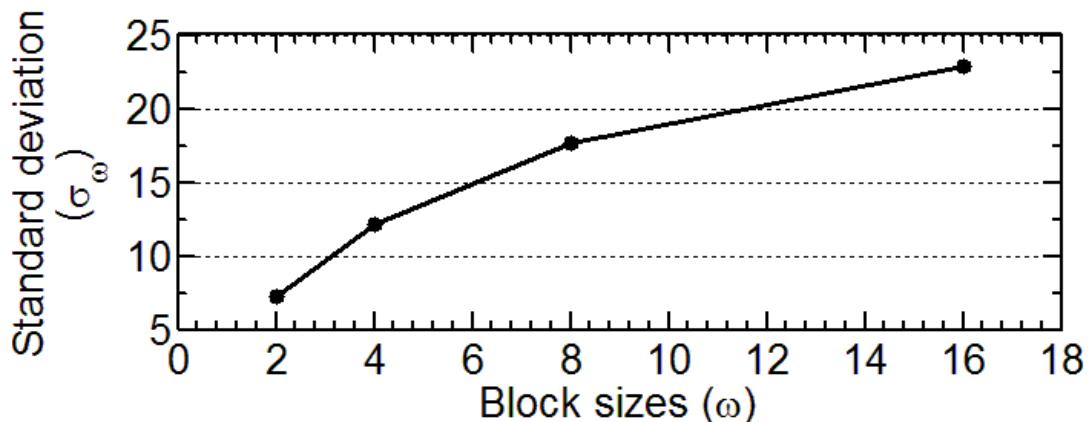


圖 5.4 平均標準差(σ_ω)在不同大小區塊的關聯性

5.3 自適應量化色階(Adaptive quantization level)

如同在文獻[29]中的討論，過往以半色調為基礎的區塊截斷編碼技術，每個區塊都是以該區塊的最大及最小灰階值做為該區塊重建所需的量化色階。雖然用此量化色階重建的影像能夠減緩可視的區塊效應(Blocking effect)及偽輪廓(False contour)等問題的產生，但也因此引入了大量的脈衝雜訊(Impulse noise)。這個問題在區塊灰階複雜度更高的時候會更嚴重。而過往技術如傳統的區塊截斷編碼(BTC) [18]或者絕對矩量區塊截斷編碼(AMBTC) [25]，企圖藉由調整量化色階的方式減少上述脈衝雜訊的產生。然而，雖然保留了區塊的矩量使得脈衝雜訊減緩許多，但也因此造成區塊效應及模糊邊界等問題，犧牲了重建影像的品質。且過往許多對彩色影像壓縮的區塊截斷編碼技術[25]、[26]、[30]都是以絕對矩量區塊截斷編碼為基礎去改良，但原有的區塊效應等問題仍然沒有完全解決。綜合上述討論，極端的量化色階促使重建影像產生脈衝雜訊，但區塊效應能被緩解。反之，當兩個量化色階越相近會導致有明顯的區塊形狀產生，但能夠完美的抑制脈衝雜訊。由此可知，量化色階是影響人眼視覺感知極大的因素之一。為了提供一種能夠滿足量化色階變化的一般形式，平衡區塊效應及脈衝雜訊的量化色階公式如下：

$$a = x_{min} + (\bar{x} - x_{min}) \times \beta_\omega, \quad (5.3)$$

$$b = x_{max} - (x_{max} - \bar{x}) \times \beta_\omega, \quad (5.4)$$

其中， a 及 b 分別表示較低及較高的量化色階； $\beta_\omega \in [0,1]$ 表示在不同區塊大小 $\omega \times \omega$ 用來控制量化色階趨近區塊極端值或者平均值的變數； x_{min} 及 x_{max} 表示區塊的最小及最大灰階值。於此，一個較低的 β_ω 使量化色階變得趨近極端值 x_{min} 及 x_{max} ，且因此使得區塊效應較不容易被察覺，這是由於半色調抖動特性的優勢，在邊界的地方以高頻的點去呈現使得人眼感覺區塊邊界不再那麼明顯。然而，較低的 β_ω 使量化色階之間的差異變大，這造成重建影像上能夠明顯地察覺一個區塊中兩個色階的不同，當在平滑區域時選擇到這種量化色階則會產生脈衝雜訊。為了提供一個主觀的觀察，一個實際案例如圖 5.5 所示，將公式(5.3)及公式(5.4)的量化色階套用在彩色點擴散區塊截斷編碼(CDDBTC)中，並使用不同 β_ω 來觀察其重建後的影像，其中區塊大小設為 $\omega = 16$ 。從

圖中可以發現，當 $\beta_{16} = 0$ 時會在區塊邊界的周圍產生抖動的點減緩區塊效應。相反地，當 $\beta_{16} = 0.5$ 時則能夠避免脈衝雜訊的產生，但同時也引入了容易被感知的區塊效應如圖 5.5(b)。

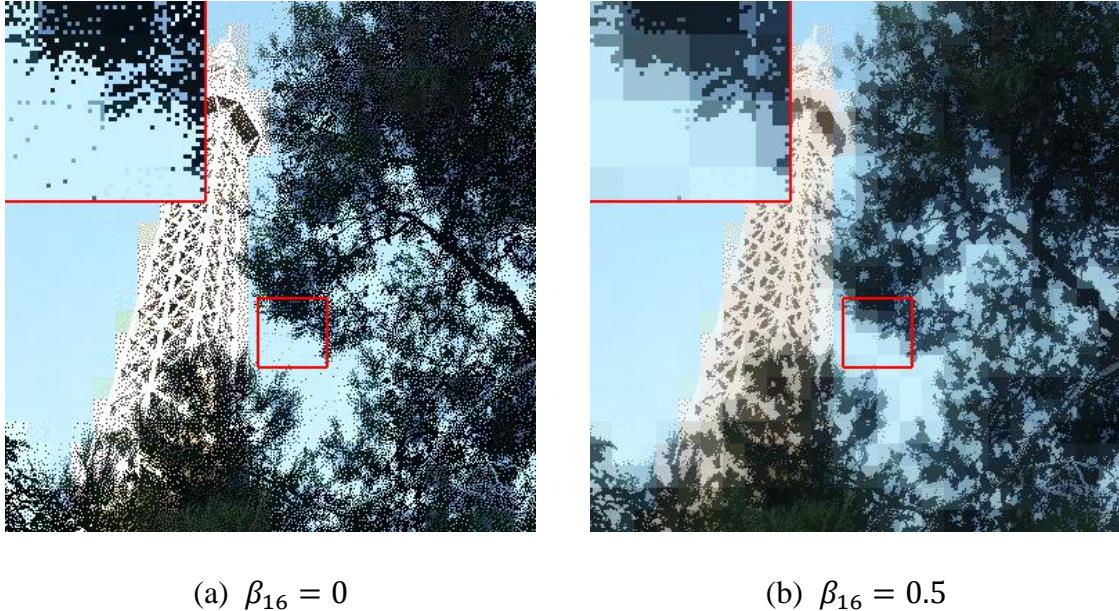


圖 5.5 經彩色點擴散區塊截斷編碼(CDDBTC)壓縮後重建的影像，其中量化色階套用公式(5.3)及公式(5.4)使用不同的 β_ω ，區塊大小 $\omega = 16$

為了產生最佳化的 β_ω ，首先評估影像內容的變動量及變數 β_ω 之間的關係。為了訓練最佳化變數，資料庫 Corel 1,000 [44] 在這被使用，並將當中的影像切割成多個不重疊的區塊。相應的結果如圖 5.6 所示，其中每個黑色的線(又名為實際線, Actual)表示的是在情況 $B(\omega, \sigma_\omega)$ 時平均最佳的 β_ω ， $B(\omega, \sigma_\omega)$ 是表示區塊的大小為 $\omega \times \omega$ 且區塊是特定的標準差 σ_ω 。為了得到在各個情況 $B(\omega, \sigma_\omega)$ 下最佳的 β_ω ，使用暴力破解法透過連續以 10^{-3} 改變 β 從 0 到 1，接著根據點擴散區塊截斷編碼論文[29]所定義的類別矩陣(Class matrix, CM)及擴散矩陣(Diffused matrix, DM)對區塊分別壓縮處理，詳細壓縮流程如 3.5 節所介紹的點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)。當中會使用到不同大小的類別矩陣及其對應的擴散矩陣，這些會依據 Guo 及 Liu 在文獻[29]中所提出的優化演算法去訓練。壓縮完成後，使用人眼視覺峰值信噪比(Human-visual

PSNR, HPSNR) [46]做為成本函數(Cost function)評估重建後的影像，其公式如下：

$$HPSNR = 10 \log_{10} \frac{P \times Q \times 255^2}{\sum_{i=1}^P \sum_{j=1}^Q [\sum_{(m,n)} g(m,n) \times (x(i+m,j+n) - y(i+m,j+n))]^2}, \quad (5.5)$$

其中， $x(i,j)$ 及 $y(i,j)$ 分別表示的是原始影像及對應的壓縮影像其像素值； $g(m,n)$ 表示高斯模型模擬人眼視覺系統(Human visual system, HVS)的權重，在本研究中，高斯模型的標準差固定設定為 1.3。將各區塊由人眼視覺峰值信噪比評估為最佳結果時的 β_ω 統計並平均就可得優化的結果，如圖 5.6 所示，一個普遍的現象就是最佳化的 β_ω 與區塊標準差 σ_ω 成反比，因為具有低標準差的區塊當中灰階值都是相似的，所以使用趨近平均值的量化色階重建平滑區域效果最佳，而反之亦然。由於相同標準差的區塊，其數量並不一定相同，因此會導致如實際線那樣起伏的問題。為此，將實際線分別以線性回歸(Linear regression)的方式模擬出相似的模型曲線，公式如下：

$$\beta_\omega = \sum_{i=0}^6 u_i (\sigma_\omega)^i, \quad (5.6)$$

u_i 為*i*次方的區塊標準差對應的權重，如表 5.1 所示。如圖 5.6 所示以彩色的線(又名為回歸線，Regression)表示模擬後的曲線。在本研究所提出的改良技術中，將用此回歸線的優化 β_ω 來權衡區塊效應及脈衝雜訊。

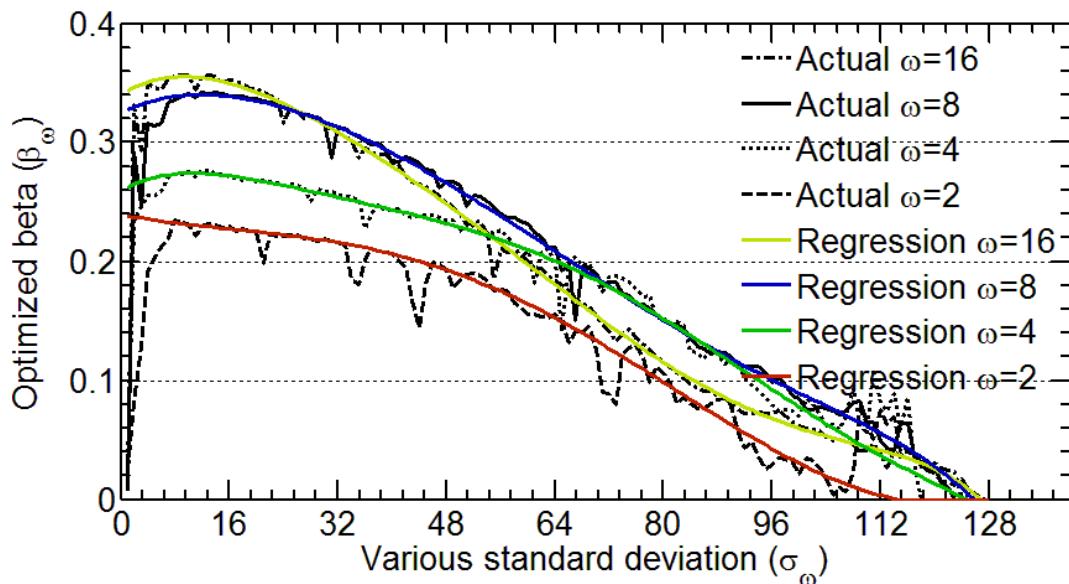


圖 5.6 在不同標準差情況下平均的優化 β_ω

表 5.1 β_ω 的模擬曲線參數

Parameters \ ω	2	4	8	16
u_0	0.23843	0.26201	0.32729	0.34302
u_1	-1.0307×10^{-3}	3.0048×10^{-3}	2.5414×10^{-3}	3.2634×10^{-3}
u_2	2.9603×10^{-5}	-2.3414×10^{-4}	-1.4796×10^{-4}	-2.5452×10^{-4}
u_3	-6.4311×10^{-7}	6.4693×10^{-6}	2.3608×10^{-6}	5.4849×10^{-6}
u_4	-1.1594×10^{-9}	-9.1932×10^{-8}	-2.9332×10^{-8}	-7.4589×10^{-8}
u_5	5.0776×10^{-11}	6.0742×10^{-10}	2.2610×10^{-10}	5.4796×10^{-10}
u_6	-1.5251×10^{-13}	-1.4973×10^{-12}	-7.0371×10^{-13}	-1.5716×10^{-12}

5.4 週期性圖案(Periodic pattern)

位元圖(Bitmap)如同公式(3.6)所定義，控制了兩個量化色階(Quantization level)的分佈位置並會影響到重建區域的結果。然而，當同一個區域的位元圖極度相似時，一問題名為週期性圖案(Periodic pattern) [47]時常會發生。過往的 DDBTC 由於使用同一種類別矩陣(Class matrix, CM)去鋪裝整個影像，其類別鋪裝(Class tiling, CT)的結果如圖 5.7，因此時常會在相似結構的區域產生週期性圖案。為了解決這個問題，藉由文獻[47]所提出的方法，將相鄰區域的每個區塊使用不同的 CM，可以避免產生相似的位元圖並減緩週期性圖案的問題，在此方法中將隨意的從八種 CM 中挑選並產生 CT，八種 CM 的產生方法如圖 5.8，圖中的 UL、UR、LL 及 LR 分別表示的是 CM 的左上、右上、左下及右下角落。在挑選 CM 時，以下兩種情況需要考量：

1. 避免將兩個相同的 CM 放置在相鄰的空間，減少週期性圖案的產生，實際的案例如圖 5.9(a)。
2. 為了使誤差擴散能夠盡量向周圍補償，要避免與相鄰 CM 其邊界的像素有相同的處理順序 $o_{i,j}$ ， $o_{i,j}$ 的定義如同公式(3.31)，實際的案例如同圖 5.9(b)。

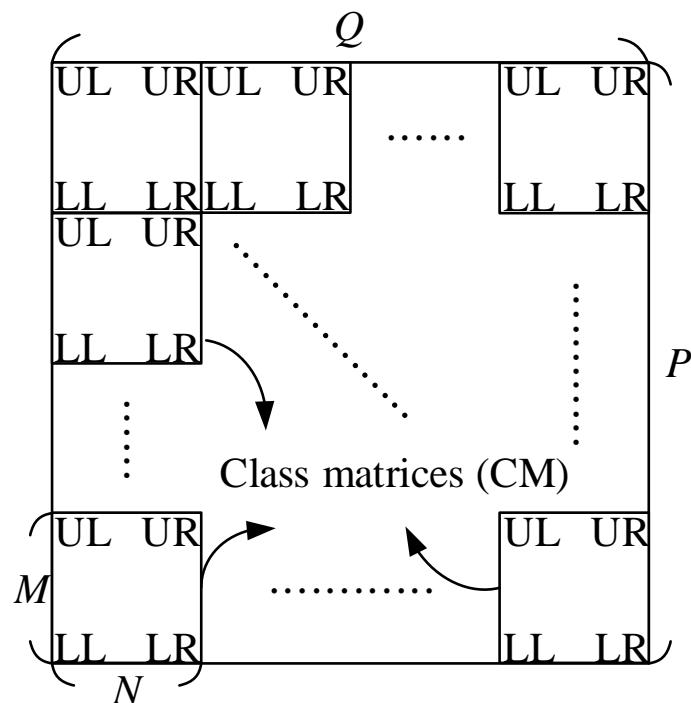


圖 5.7 DDBTC 技術慣例的鋪裝方法

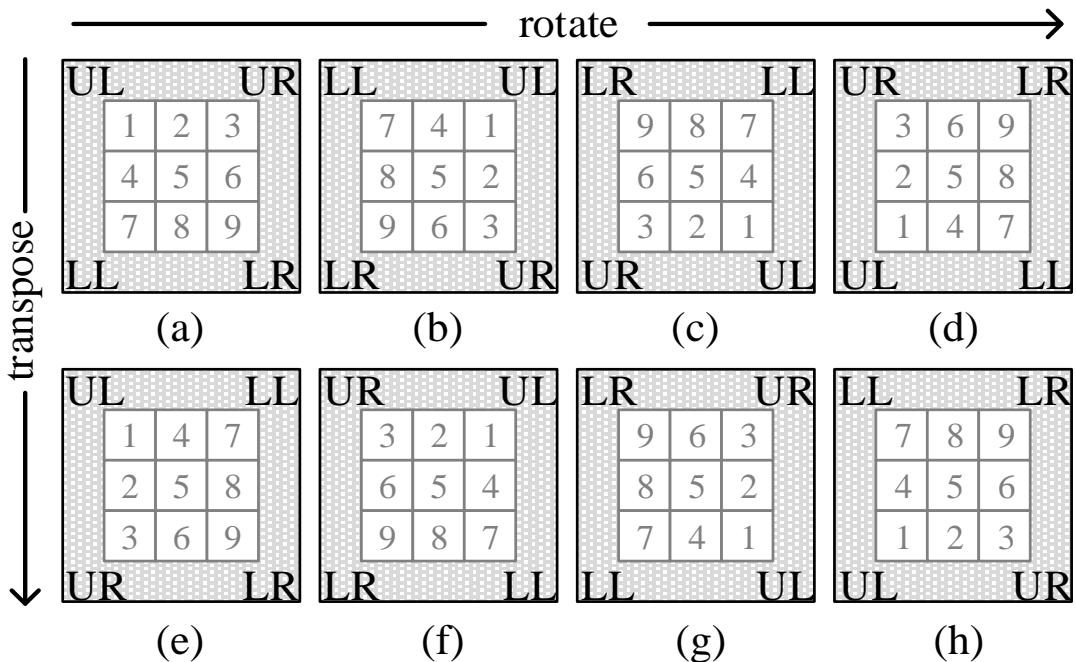


圖 5.8 在此以 3×3 大小的矩陣做為範例表示八種不同的 CM。值得注意的是，任何一種都只是經由簡單的旋轉或對調而產生

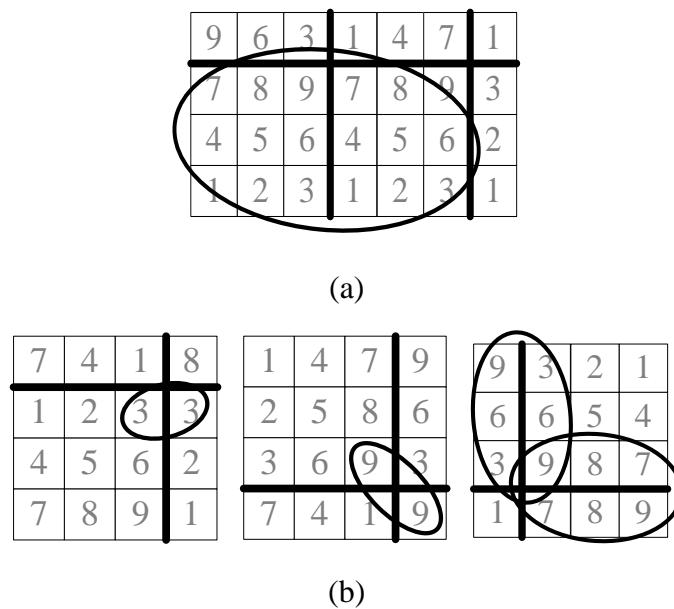


圖 5.9 安排 CM 的方法中不能出現的案例。(a)相鄰 CM 有完全相同的處理順序，(b)相鄰像素有相同的處理順序(會導致 3×3 大小的擴散矩陣無法正常運作)

為了遵守情況 2，候補 CM 的索引 k^* 選擇方法如下：

$$k^* = \operatorname{argmin}_k(s_k), \text{ where} \quad (5.7)$$

$$s_k = \sum_{(i,j)} \sum_{(m,n)} \begin{cases} 1, & \text{if } o(i,j) = o(i+m, j+n), \\ 0, & \text{otherwise} \end{cases}, \quad (5.8)$$

其中 (i,j) 表示候補 CM 中像素的座標； (m,n) 表示 DM 中像素的座標； k 表示圖 5.8 中所有 CM 的索引。值得注意的是，上述情況只適用於相鄰區塊與現在要鋪裝的區塊有相同的大小時，舉例來說，當鄰近的 CM 為 4×4 大小時，只需確認當前 4×4 大小的候補 CM 是否符合上述兩個情況即可。在判斷情況 2 時，各個候補 CM 將依照公式 (5.8) 計算出 s_k ，並根據公式 (5.7) 挑選出最小 s_k 的索引 k^* ，再用 k^* 所對應的 CM 去鋪裝 CT。上述判斷中， k^* 所對應的 CM 必須符合情況 1 才行，因此在選擇候補 CM 時，若不符合情況 1 則不需判斷情況 2，可直接取下一個候補 CM 重新做判斷。

由於彩色影像是由多個通道組合而成，因此使用如圖 5.7 慣例的鋪裝方法產生 CT 極少會使重建影像產生週期性圖案，所以對彩色影像壓縮的步驟中，其 CT 中的 CM 不會經過旋轉。CT 不一定會與原始影像一樣大小，可以藉由映射(Mapping)的方式將 CT 映射到原始影像上，可避免因為要產生較大的 CT 耗費太多的時間。此外，為了降低運算複雜度，在程式中會依照圖 5.8 各個 CM 的處理順序儲存其座標位置，舉例來說，圖 5.8(a) 中 “1”的位置為 $(0,0)$ 、 “9”的位置為 $(2,2)$ ，則程式可依照此座標迅速依照順序處理區塊。在程式中，公式 (3.31) 需要額外的判斷是否會超出邊界，可以使用擴邊的方式將映射後與原始影像大小相同的 CT' 其外圍多加一圈並設為 0，原本每個像素需要判斷上下左右四個方向是否超出邊界，若影像大小為 $P \times Q$ 則需要判斷 $4PQ$ 次，而改用擴邊後則可省略判斷，只需要多處理外圍增加的 $(PQ + 4)$ 個像素，比起原本的判斷大幅地降低了計算複雜度。

5.5 $\gamma_\omega(\varphi)$ 的定義

$\gamma_\omega(\varphi)$ 如同前述，表示的是使用者期望的品質 φ 在區塊大小為 $\omega \times \omega$ 時所限定的標準差範圍。其表示方式為 $\gamma_\omega(\varphi) = [0, t_\omega(\varphi)]$ 。圖 5.10 呈現不同區塊大小在不同的區塊標準差時的品質曲線，當中使用 Corel 1,000 [44]做為統計的資料庫。圖中的實際線表示使用特定的區塊大小壓縮後以人眼視覺峰值信噪比評估的平均結果，壓縮的過程中使用訓練完成的變數 β_ω 如圖 5.6 產生能夠適應區塊情況的量化色階，接著依照訓練好的類別矩陣(CM)依序處理並用擴散矩陣(DM)擴散誤差。從圖 5.10 中可以發現，區塊大小與影像品質成反比，因為在相同範圍下，較小的區塊能夠用較多的量化色階去重建，但也使的壓縮率(CR)較低。為了在各個品質下取得最高的壓縮率，實際線將以以下公式模擬並重建：

$$t_\omega(\varphi) = e^{\frac{\varphi - v_0}{v_1}}, \quad (5.9)$$

v_i 表示在不同區塊大小下的參數，如表 5.2 所示，模擬後的結果如圖 5.10 彩色的線。以 φ 代入公式(5.9)就可取得限定的標準差範圍 $\gamma_\omega(\varphi)$ ，當中使用者期望的品質 φ ，其正比於圖 5.10 垂直軸的影像品質 HPSNR，因此區塊的標準差其對應曲線符合條件 $HPSNR \geq \varphi$ 時都可以使用。舉例來說，當期望的品質 $\varphi = 60$ ，則可以公式(5.9)得到 $t_2(\varphi) = 60.63$ 、 $t_4(\varphi) = 28.59$ 、 $t_8(\varphi) = 12.28$ 、 $t_{16}(\varphi) = 4.63$ 。則各區塊允許的標準差範圍為 $\gamma_2(\varphi) = [0, 60.63]$ 、 $\gamma_4(\varphi) = [0, 28.59]$ 、 $\gamma_8(\varphi) = [0, 12.28]$ 、 $\gamma_{16}(\varphi) = [0, 4.63]$ 。假設目前 16×16 的區塊其標準差大於 4.63，則表示這個區域使用 16×16 的區塊產生的壓縮影像其品質可能會無法符合期望的品質 $\varphi = 60$ 。

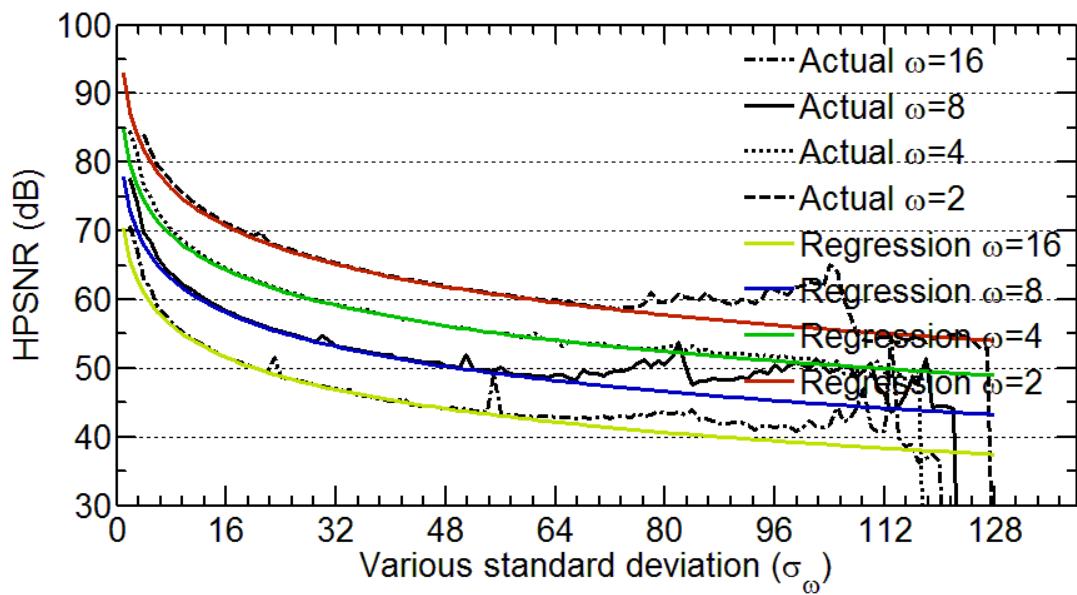


圖 5.10 在不同標準差情況下平均的人眼視覺峰值信噪比(HPSNR)

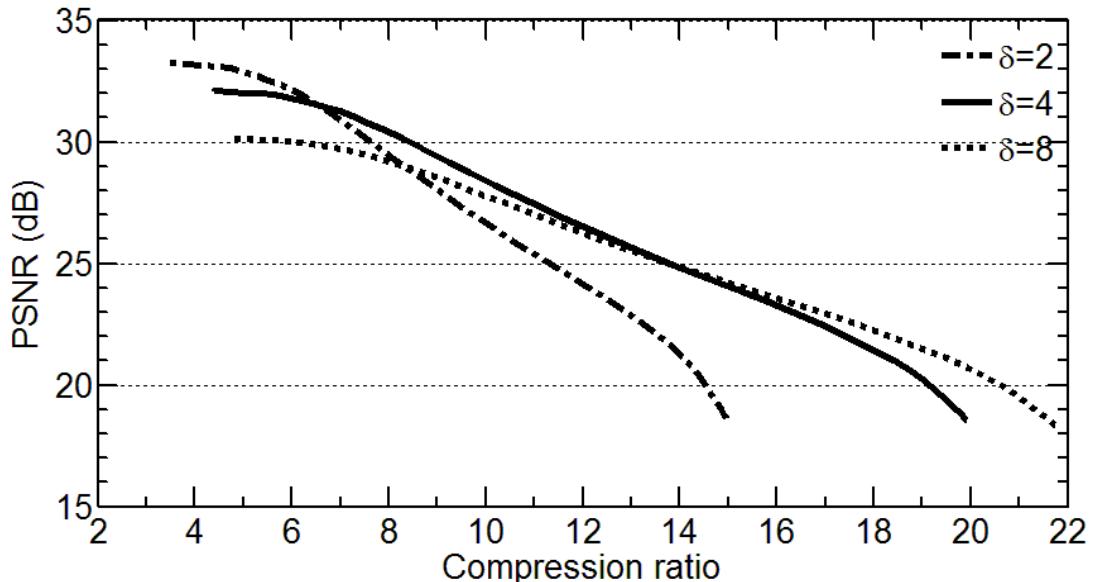
表 5.2 $t_\omega(\varphi)$ 的模擬曲線參數

Parameters \ ω	2	4	8	16
v_0	92.945	84.688	77.924	70.4
v_1	-8.026	-7.363	-7.146	-6.788

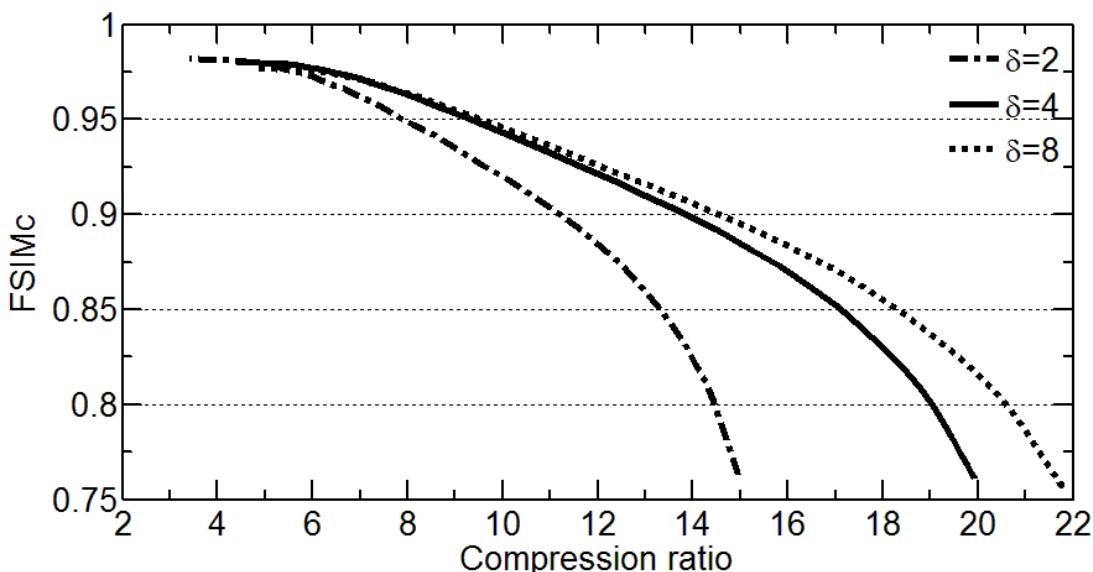
5.6 參數設定

在本研究中，會有兩個參數控制改良技術演算法的行為：1) φ 控制期望的壓縮影像品質，2) δ 控制 a、b 分量的縮放比例，如圖 5.3 的流程，縮小的比例與其對應放大的比例都要為 δ 才能得到與原始影像一樣的大小。為了觀察 δ 對重建影像的影響，在此使用公認做為評估重建影像指標的峰值信噪比(PSNR)。然而，由於它沒有考量到人眼視覺系統(Human visual system, HVS) [43]，在過往文獻[48]-[49]中證實了峰值信噪比在主觀性的保真度評估是較低的。因此，在本研究的比較中也使用了基於人眼視覺系統的評估方法-特徵相似度指數(Feature similarity index, FSIM) [49]。它能夠感知影像突出的細微特徵如相位一致性(Phase congruency)及梯度大小(Gradient magnitude)，使其具有較佳的保真度評估。對於灰階影像及彩色影像的評估分別簡稱為 FSIM 及 FSIMc，其評估數值的範圍為 0 到 1，越高的數值表示被評估的影像與原始影像的相似度越高。由於特徵相似度指數的各項優勢，在後續的實驗結果分析中，我們的決策將較傾向於特徵相似度指數評估的結果，並以此做為客觀的結論。

在此測試中使用資料庫 Corel 1,000 [44]，在不同壓縮率情況下對應的結果如圖 5.11 所示。在 $\delta = 8$ 時根據特徵相似度指數評估，雖然改良技術能夠保持壓縮後的影像品質又能夠有最高的壓縮率，但在峰值信噪比的評估中，明顯地比 $\delta = 4$ 時的品質降低了許多。從這一觀察可以了解到，此情況表示人眼視覺在一適當距離下不會察覺到壓縮影像的失真，如同特徵相似度指數的評估。峰值信噪比的評估是表示在仔細觀察下(較近的距離)的感受，從圖 5.11(a)可以明顯地發現，在 $\delta = 8$ 時的影像會比在 $\delta = 4$ 時劣化的更嚴重。綜合上述討論，我們決定將參數設定在較適當的數值 $\delta = 4$ ，此設定將會在後續的實驗結果中使用。



(a) PSNR



(b) FSIMc

圖 5.11 在不同壓縮率(CR)及縮放倍率(δ)下，使用評估方法(a) PSNR 及(b) FSIMc 的平均影像品質。其中，每條線的數值標示的是使用不同縮放倍率(δ)的結果。

5.7 自適應點擴散區塊截斷編碼

本論文所提出的方法根據 3.5 節介紹的點擴散區塊截斷編碼(Dot-diffused BTC, DDBTC)的概念，最佳化區域適合的區塊大小及量化色階。其詳細的編碼流程如下：

1. 產生給灰階影像壓縮使用的類別鋪裝(CT)，其由 5.4 所介紹的八種類別矩陣(CM)所組成，鋪裝時需遵守 5.4 節所述的兩種情況；給彩色影像壓縮使用的 CT 由固定的 CM 所組成。
2. 若為彩色影像則如圖 5.3 所示將影像從 RGB 轉換成 Lab，並將 a、b 通道的長寬分別縮小 δ 倍，依照 5.6 節分析， δ 設為 4。若為灰階影像則直接跳至步驟 3。
3. 彩色三通道/灰階單通道，將 CT 映射(Mapping)到影像上並依照 5.3 節所述的區塊區域分解演算法(Block region decomposition algorithm)將區域分割成最適合的區塊大小。當中可藉由積分圖得到區塊的平均值 \bar{x} 及標準差 σ_ω 。
4. 各區塊的處理是先取得區塊的平均值 \bar{x} 及標準差 σ_ω ，並依照公式(5.3)-(5.4)取得區塊的量化色階 a 及 b 。
5. 區塊各像素依照 CT 中 CM 的數值從小到大依序處理，處理的公式如下：

$$c(m, n) = x(m, n) + s(m, n), \text{ where } s(m, n) = \sum_{(p, q) \in R} \sum \frac{e(m+p, n+q) \times w(p, q)}{\text{sum}}, \quad (5.10)$$

$$\text{sum} = \sum_{(p, q) \in R} \begin{cases} w(p, q), & \text{if } o(m+p, n+q) > o(m, n) \\ 0, & \text{if } o(m+p, n+q) \leq o(m, n), \end{cases} \quad (5.11)$$

$$e(m, n) = \begin{cases} c(m, n) - b, & \text{if } c(m, n) \geq \bar{x} \\ c(m, n) - a, & \text{if } c(m, n) < \bar{x}, \end{cases} \quad (5.12)$$

$$h(m, n) = \begin{cases} 1, & \text{if } c(m, n) \geq \bar{x} \\ 0, & \text{if } c(m, n) < \bar{x}, \end{cases} \quad (5.13)$$

$x(m, n)$ 為原始影像在各區塊內的灰階值； $c(m, n)$ 為經過誤差擴散補償過後的灰階值； $e(m, n)$ 為輸出與當下補償後灰階的差值； $w(p, q)$ 表示在誤差擴散矩陣的範圍 R 內的誤差擴散權重； $o(m, n)$ 為 CM 中的數值； $h(m, n)$ 為區塊所產生的位元圖。與 DDBTC 不同的是，公式(5.11)依照 5.4 節最後所述的方法能省略邊界的判斷。

6. 各區塊將儲存 2 bits、位元圖 $h(m, n)$ 及量化色階 a 和 b ，2 bits 表示的是區塊的大小。
7. 重複上述步驟 3-6 直到完成所有區塊，就可完成整張影像的壓縮。

5.8 實驗結果比較

改良技術將與過往的區塊截斷編碼技術做比較，對灰階影像為以下五個過往技術：1) 傳統區塊截斷編碼(BTC) [18]，2) 絕對矩量區塊截斷編碼(AMBTC) [25]，3) 誤差擴散區塊截斷編碼(EDBTC) [27]，4) 有序抖動區塊截斷編碼(ODBTC) [28]，5) 點擴散區塊截斷編碼(DDBTC) [29]；對彩色影像則為以下五個過往技術：1) 彩色絕對矩量區塊截斷編碼(CAMBTC) [25]，2) 單一位元圖區塊截斷編碼(SBMBTC) [26]，在此根據亮度規則(Luminance rule)產生的結果做為評估對象，3) 基於單一位元圖的編碼方案(CS-SBM)中的第一方案位元圖省略技術(FS-BMO) [30]，4) 分群區塊截斷編碼(IBTC-KQ) [32]，5) 彩色點擴散區塊截斷編碼(CDDBTC) [24]，以上技術的參數將會設定在最佳的情況。

對灰階影像及彩色影像的壓縮技術其實驗結果都使用資料庫 UCID [50]，共 1,338 張大小為 512×384 的無壓縮彩色影像。在後續的比較中將會針對以下三個方向做評估：1) 影像的品質及其對應的壓縮率，2) 不自然紋理問題(Artifact problem)，3) 處理速度(Runtime)。在灰階影像的測試中，過往技術分別使用 4×4 、 8×8 及 16×16 大小的區塊，要注意的是，點擴散區塊截斷編碼技術無法使用 4×4 大小的區塊，各技術輸入參數詳細的設定及其對應的壓縮率如表 5.3。此外，灰階影像壓縮技術在不自然紋理問題的章節中，將會額外針對週期性圖案(Periodic pattern)做測試。

表 5.3 各技術參數(過往技術 a~e 為區塊大小；改良技術為 φ)的設定

Methods \ Group (CR)	Group 1 (3.8~4.2)	Group 2 (6.2~6.6)	Group 3 (7.3~7.7)
a. BTC [18]	4×4 (4.0)	8×8 (6.4)	16×16 (7.5)
b. AMBTC [25]	4×4 (4.0)	8×8 (6.4)	16×16 (7.5)
c. EDBTC [27]	4×4 (4.0)	8×8 (6.4)	16×16 (7.5)
d. ODBTC [28]	4×4 (4.0)	8×8 (6.4)	16×16 (7.5)
e. DDBTC [29]	n/a	8×8 (6.4)	16×16 (7.5)
f. Proposed SDBTC	60 (4.0)	51(6.4)	30 (7.5)

由於過往彩色區塊截斷編碼技術使用相同大小的區塊並不會產生相同的壓縮率，導致在實驗結果中難以用相同大小的區塊比較。因此在彩色影像壓縮技術其實驗結果的比較中將會以各技術的壓縮率分成四類，如表 5.4。在此表中，”n/a”表示該技術的壓縮率使用任何大小的區塊都無法達到(如：彩色絕對矩量區塊截斷編碼、分群區塊截斷編碼)或者是缺少訓練的資源(如：彩色點擴散區塊截斷編碼)。另一方面，由於第一方案位元圖省略技術會視影像的情況改變壓縮率，因此在使用 8×8 大小的區塊時，壓縮率的範圍可能會落在 Group 3 及 4 之間。

表 5.4 為了符合四種壓縮率範圍，各技術參數(過往技術 a~e 為區塊大小；改良技術為 φ)的設定，括號為技術在該參數設定下所產生的壓縮率

Methods \ Group (CR)	Group 1 (3.6~4.4)	Group 2 (5.4~6.6)	Group 3 (13.7~15.3)	Group 4 (19.8~20.2)
a. CAMBTC [25]	3×3 (3.6)	4×4 (5.6)	12×12 (14.6)	n/a
b. SBMBTC [26]	3×3 (3.8)	4×4 (6.0)	8×8 (13.7)	16×16 (20.2)
c. FS-BMO [30]	3×3 (3.7)	4×4 (5.6)	8×8 (15.3~20.4)	
d. IBTC-KQ [32]	16×16 (3.8)	n/a	n/a	n/a
e. CDDBTC [24]	n/a	4×4 (6.0)	8×8 (13.7)	16×16 (20.2)
f. Proposed SDBTC	90 (4.4)	71 (6.4)	54 (14.9)	30 (19.9)

5.8.1 影像品質及壓縮率(Image quality and compression ratio)

在灰階影像壓縮的比較中，為了對影像品質提供一個較客觀的評價，在此考量五個影像品質評估(Image quality assessment, IQA)指標，其中包括了多尺度結構相似度(Multi-scale structural similarity, MS-SSIM) [51]、訊息內容權重峰值信噪比(Information content weight PSNR, IW-PSNR) [52]、視覺訊息保真度(Visual information fidelity) [53]、人眼視覺峰值信噪比(HPSNR) [46]及針對灰階影像的特徵相似度指數(FSIM) [49]。相對應的結果都整理在表 5.5 中，其中平均排名(Average ranking)表示的是在資料庫UCID [50]中各評估技術下排名的平均值。從表中可以發現到，雖然本研究所提出的技術並不是在每個評估中都保持第一位，但在六個技術中都能夠確保在前三名內，且平均排幾乎都排在第一位(1.8)。

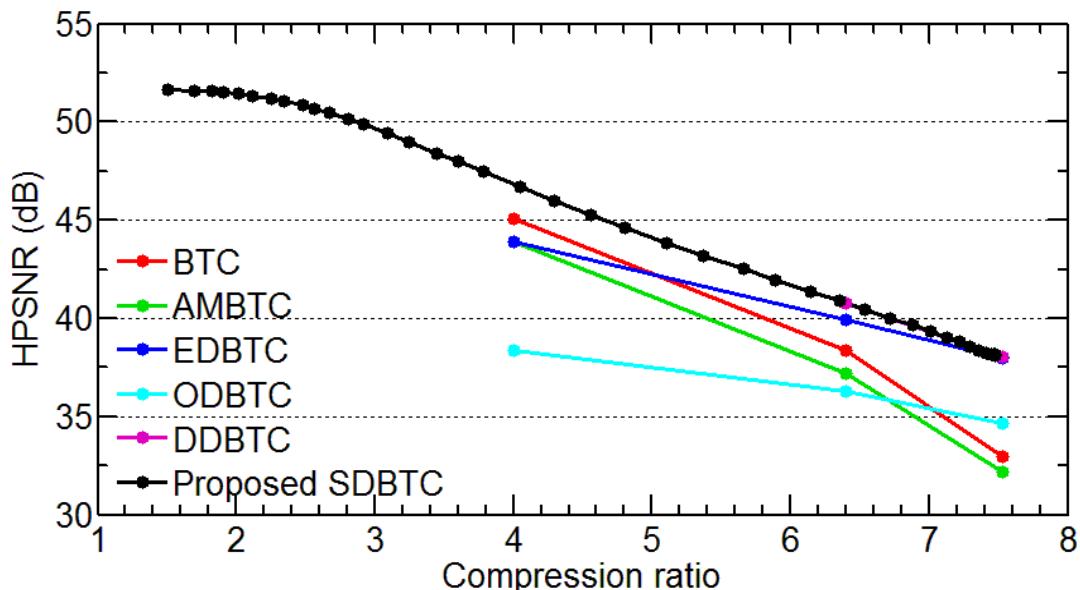
表 5.5 五種不同的客觀影像評估技術評估重建影像的結果(在此將會把最佳及次佳的結果框出；最佳的結果會再以灰底表示)

CR	Methods	Average ranking	MS-SSIM [51]	IW-PSNR [52]	VIF [53]	HPSNR [46]	FSIM [49]
4 ($\omega = 4$)	BTC [18]	1.6	0.992	38.358	0.617	45.064	0.991
	AMBTC [25]	3	0.992	36.997	0.586	43.886	0.990
	EDBTC [27]	3.6	0.977	38.348	0.580	43.924	0.979
	ODBTC [28]	5	0.962	29.971	0.426	38.353	0.969
	DDBTC [29]	n/a	n/a	n/a	n/a	n/a	n/a
	Proposed SDBTC	1.8	0.984	39.233	0.645	46.705	0.988
6.4 ($\omega = 8$)	BTC [18]	3	0.977	31.160	0.434	38.375	0.962
	AMBTC [25]	4	0.975	29.814	0.408	37.184	0.959
	EDBTC [27]	3	0.946	33.717	0.468	39.899	0.948
	ODBTC [28]	6	0.926	28.370	0.357	36.290	0.940
	DDBTC [29]	3.2	0.939	34.462	0.460	40.764	0.944
	Proposed SDBTC	1.8	0.960	32.957	0.468	40.894	0.964
7.53 ($\omega = 16$)	BTC [18]	3.6	0.947	25.312	0.322	32.934	0.898
	AMBTC [25]	4.8	0.944	24.302	0.304	32.162	0.896
	EDBTC [27]	2.6	0.901	30.817	0.397	37.952	0.905
	ODBTC [28]	5.2	0.869	26.981	0.290	34.656	0.885
	DDBTC [29]	3	0.888	32.135	0.376	38.053	0.896
	Proposed SDBTC	1.8	0.938	29.789	0.401	38.138	0.931

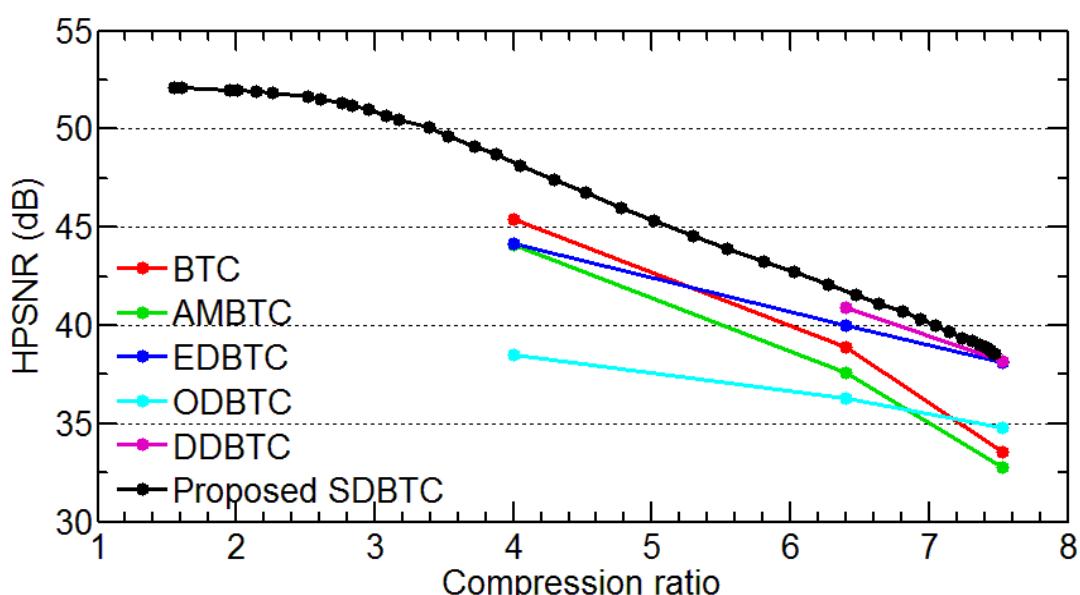
為了更準確的觀察各技術在不同壓縮率下的結果，除了先前提到的 UCID [50]，再額外的增加三個資料庫做為評估基準：1) MSPLAB [54]：1,000 張 512×512 的灰階影像，2) Kodak [55]：24 張大小為 768×512 的無損彩色影像，3) Holidays [56]：1,491 張經 jpg 壓縮後的彩色影像，資料庫中的影像有不同的大小，最小及最大分為 640×480 及 3888×2592 。上述彩色資料庫在測試時都將會轉換成灰階影像。圖 5.12 為所有技術壓縮影像在不同壓縮率(CR)下的人眼視覺峰值信噪比(HPSNR)及針對灰階影像的特徵相似度指數(FSIM)，其中所有數值都是整個資料庫平均後的結果。

從圖 5.12 中可以得知，在壓縮率範圍 $CR = [1.5, 6.4]$ 中，雖然改良的 SDBTC 技術以 FSIM 評估幾乎與 BTC 及 AMBTC 不分上下，但以 HPSNR 評估時改良的 SDBTC 技術幾乎高於其他過往的 BTC 技術；而在壓縮率範圍 $CR = [6.4, 7.5]$ 中，儘管 SDBTC 以 HPSNR 評估時與 EDBTC 及 DDBTC 差異不大，在 FSIM 評估下改良的技術遠高於其他過往的 BTC。綜合上述分析，改良的 SDBTC 在所有壓縮率下幾乎都有不錯的影像品質，並且能夠彈性地調整其壓縮率。

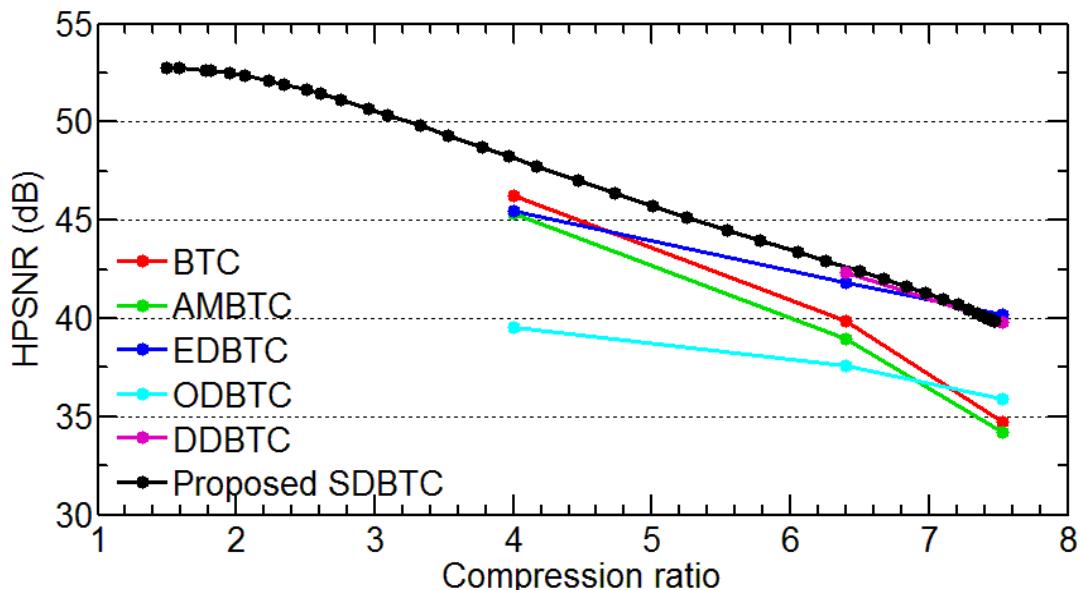
圖 5.13 及圖 5.14 分別展示圖 3.1 的灰階影像 1 及灰階影像 2 經各技術壓縮後的重建影像，括號中表示的是其壓縮率及其對應的 FSIM 數值。從圖中能夠更明確地看出在不同壓縮率下的差異，儘管 BTC 及 AMBTC 看似有不錯的結果，但在圖 5.13 中能發現它們受到明顯的區塊效應(Blocking effect)影響，而在圖 5.14 中另外會察覺到其結果較為模糊。其他以半色調為基礎的 BTC 技術，在圖 5.13 及圖 5.14 中都可以發現它們的結果嚴重受到脈衝雜訊(Impulse noise)影響。改良的技術明顯地平衡了區塊效應及脈衝雜訊並產生較自然的重建影像，保持半色調技術產生較銳利影像的優勢並同時壓抑區塊效應及脈衝雜訊的影響。



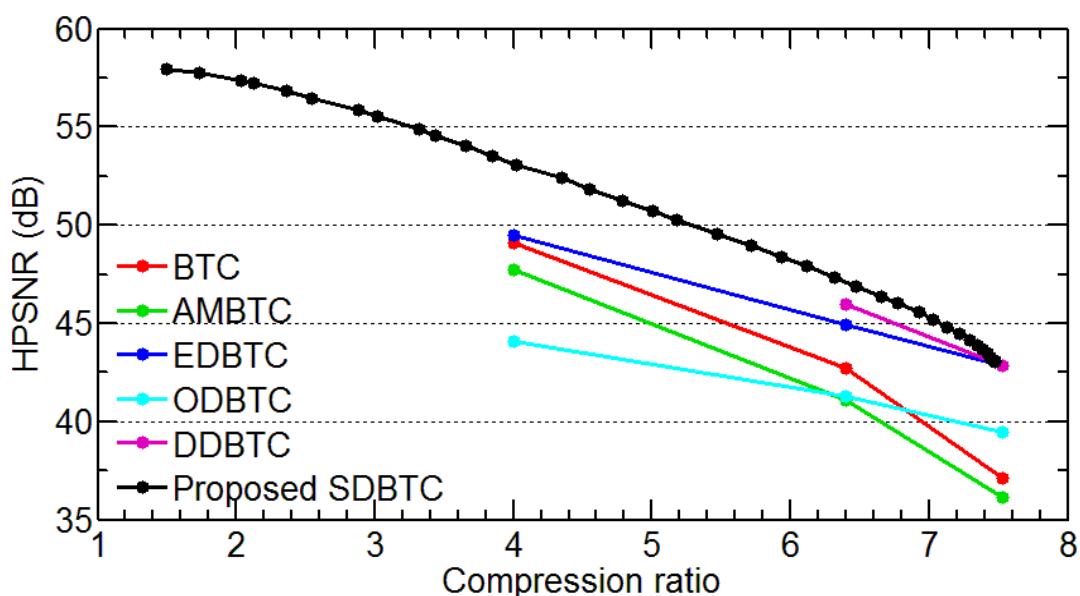
(a) 以 HPSNR 評估資料庫 UCID [50] 經灰階影像壓縮過後的結果



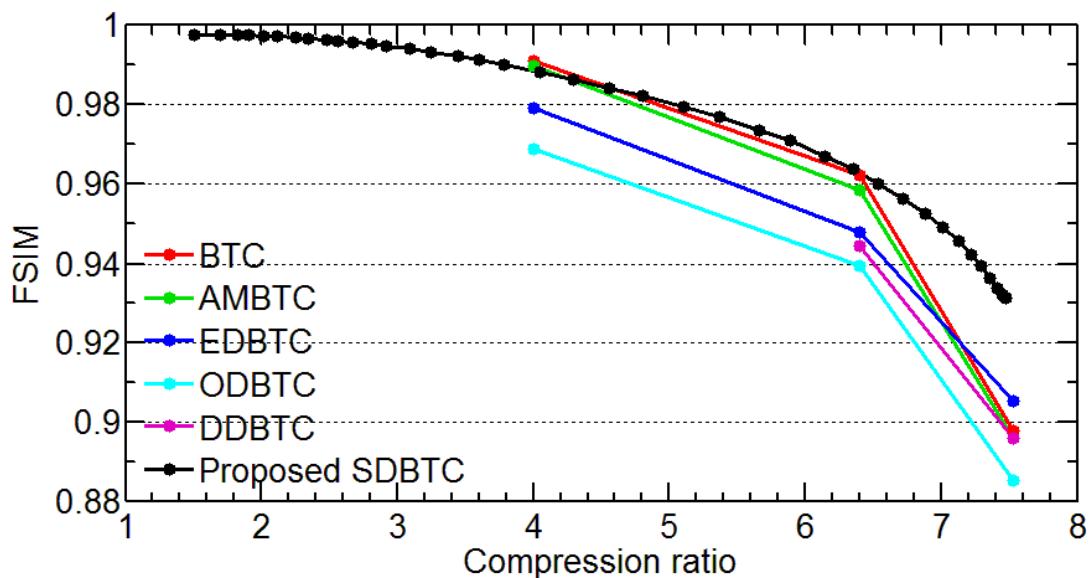
(b) 以 HPSNR 評估資料庫 MSPLAB [54] 經灰階影像壓縮過後的結果



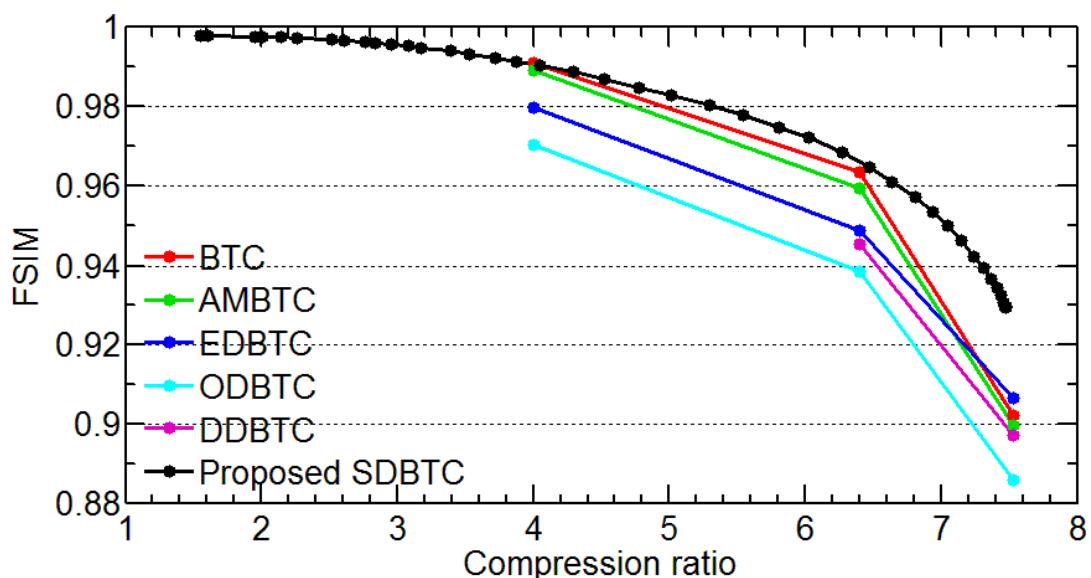
(c) 以 HPSNR 評估資料庫 Kodak [55] 經灰階影像壓縮過後的結果



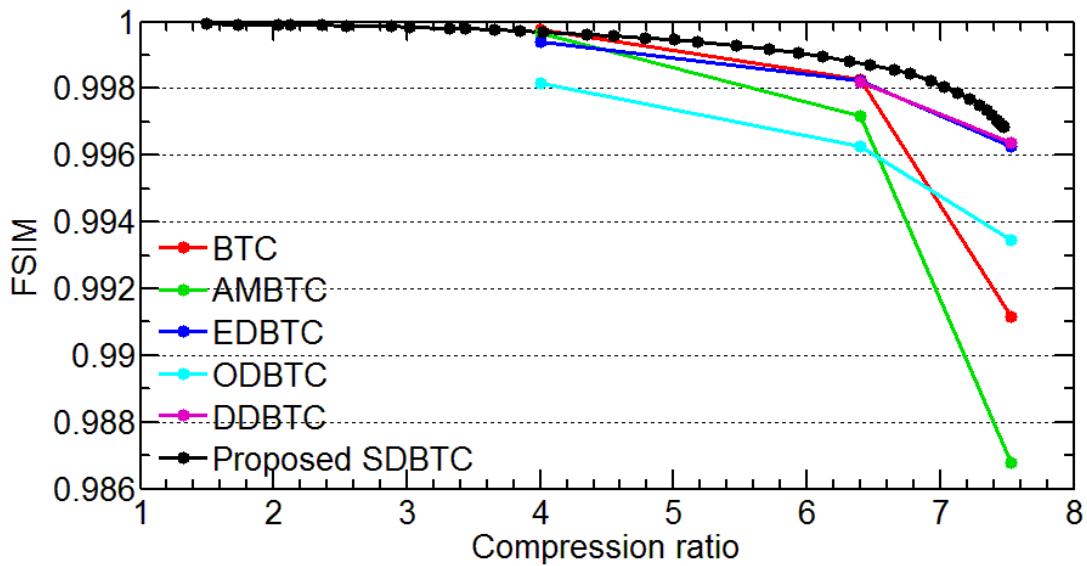
(d) 以 HPSNR 評估資料庫 Holidays [56] 經灰階影像壓縮過後的結果



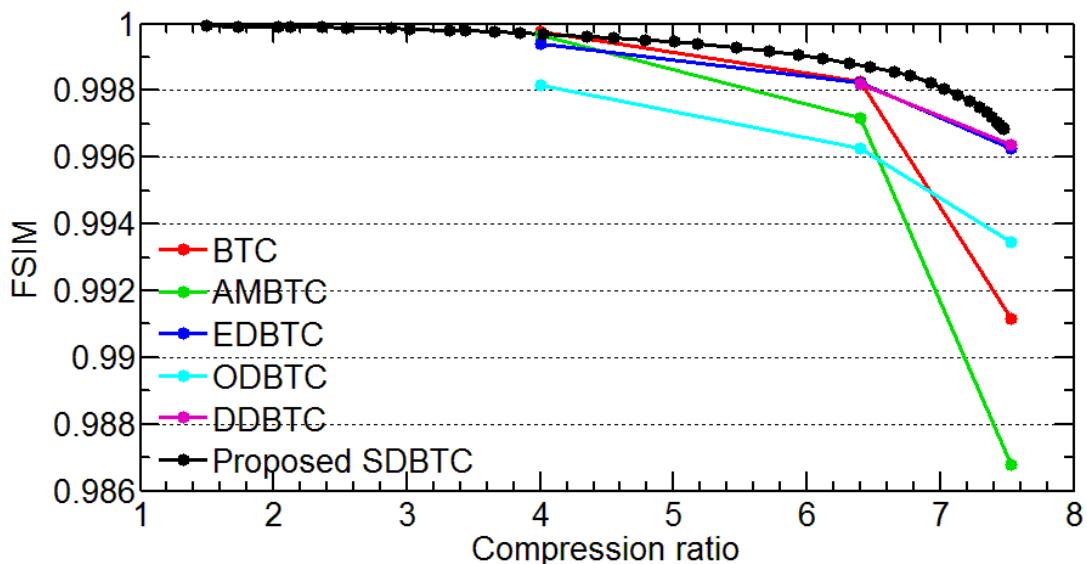
(e) 以 FSIM 評估資料庫 UCID [50] 經灰階影像壓縮過後的結果



(f) 以 FSIM 評估資料庫 MSPLAB [54] 經灰階影像壓縮過後的結果

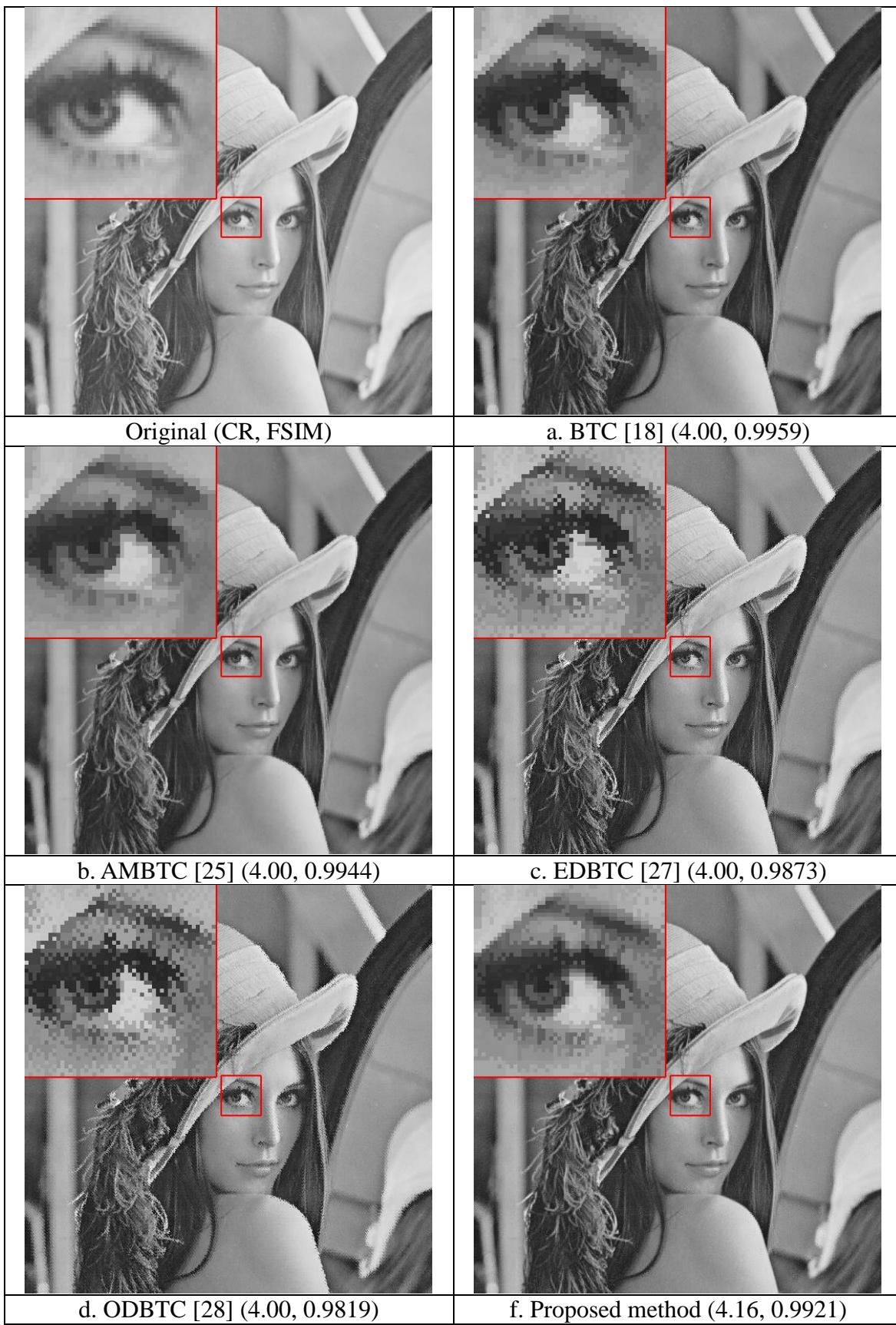


(g) 以 FSIM 評估資料庫 Kodak [55] 經灰階影像壓縮過後的結果

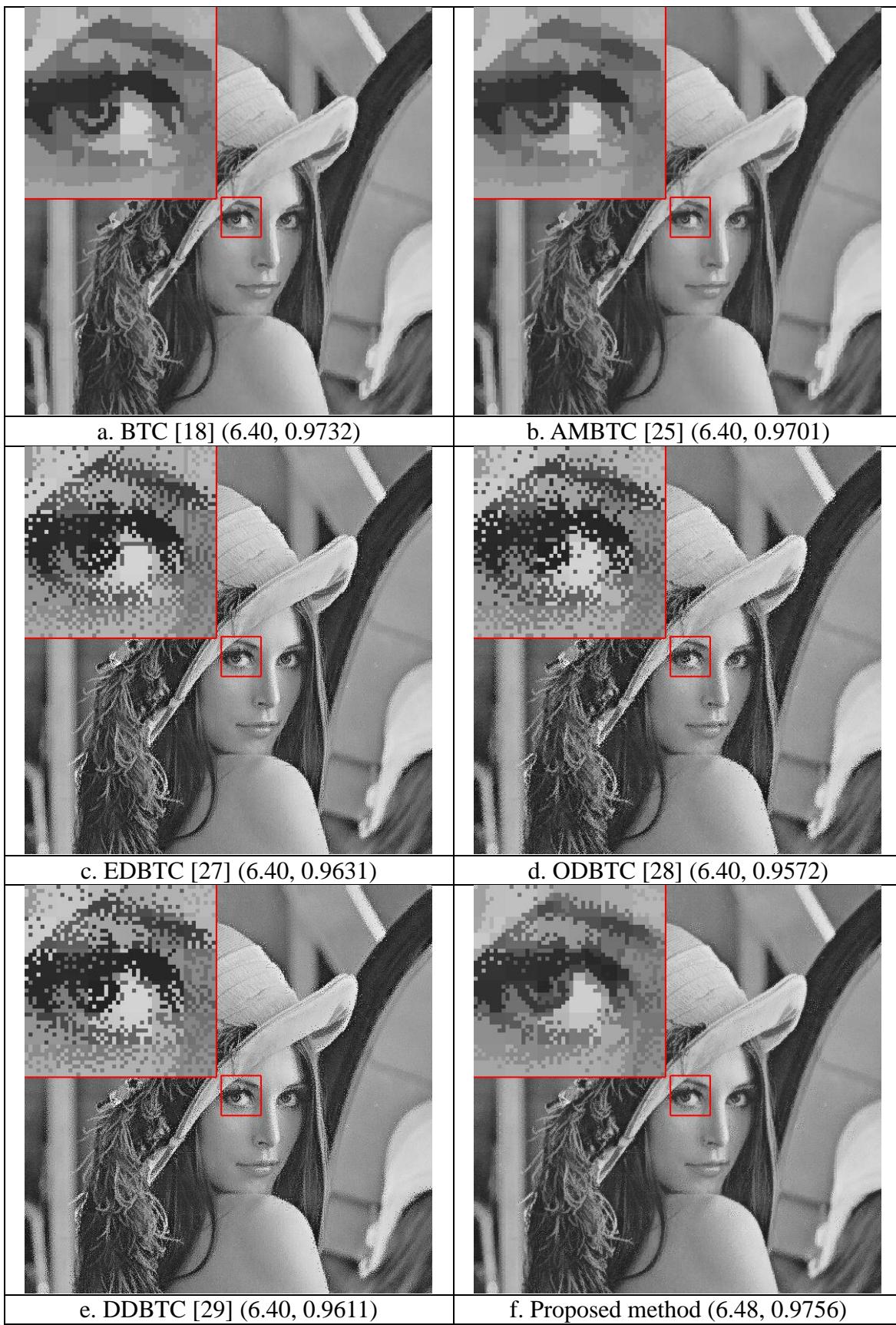


(h) 以 FSIM 評估資料庫 Holidays [56] 經灰階影像壓縮過後的結果

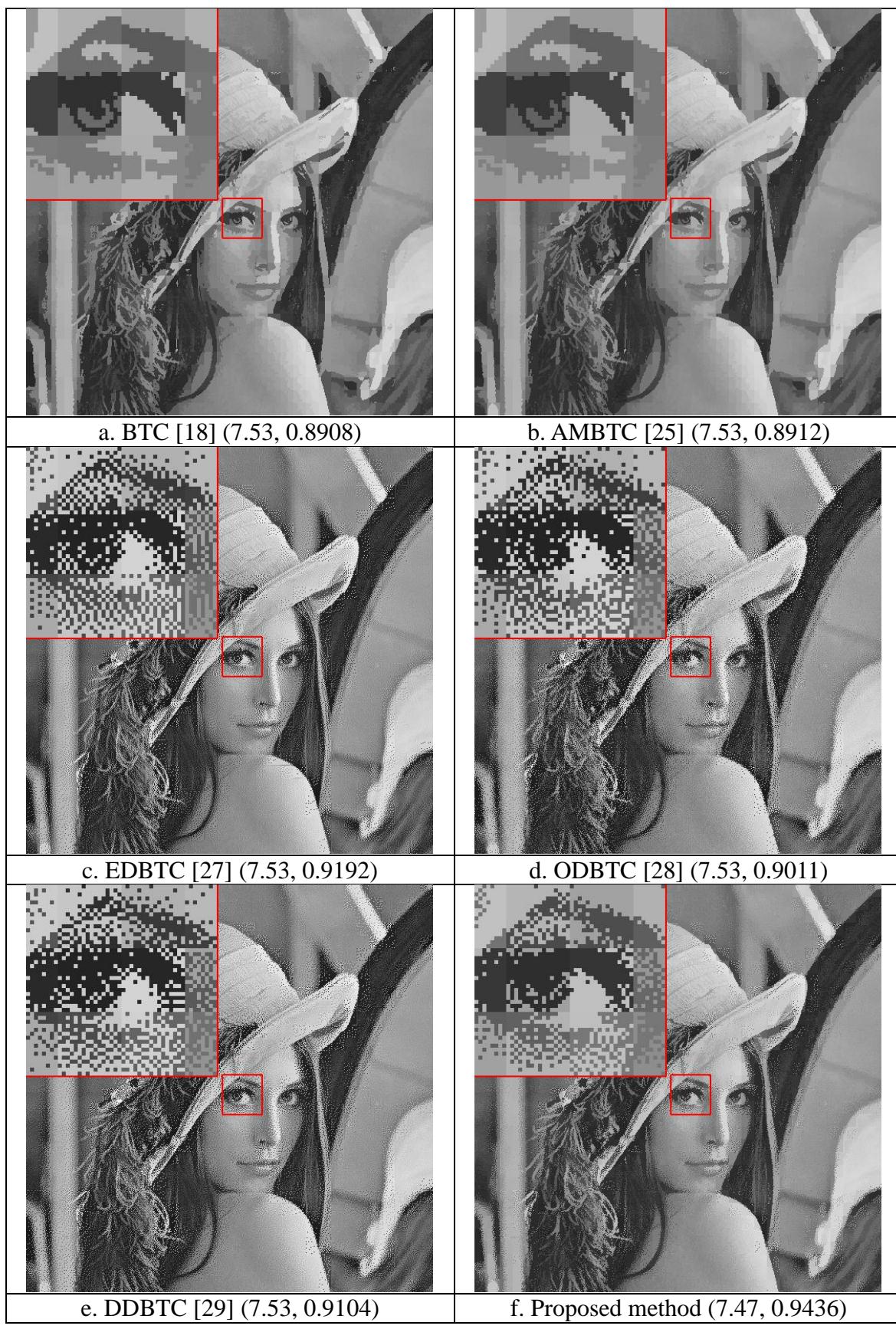
圖 5.12 所有技術在不同資料庫中，不同壓縮率下的兩種影像評估：人眼視覺峰值信噪比(HPSNR)及針對灰階影像的特徵相似度指數(FSIM)的平均數值



(a) Group 1 (3.8~4.2)

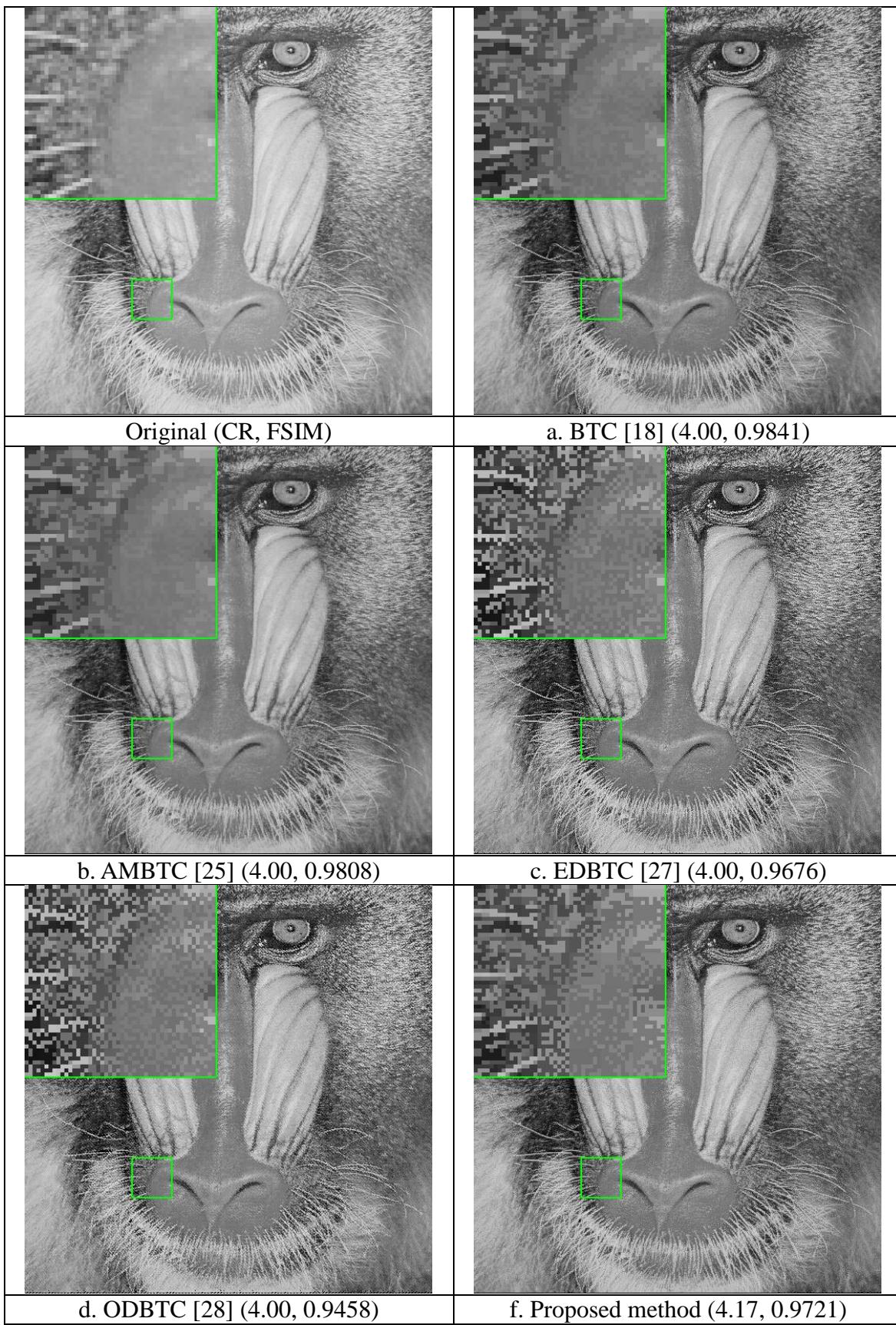


(b) Group 2 (6.2~6.6)

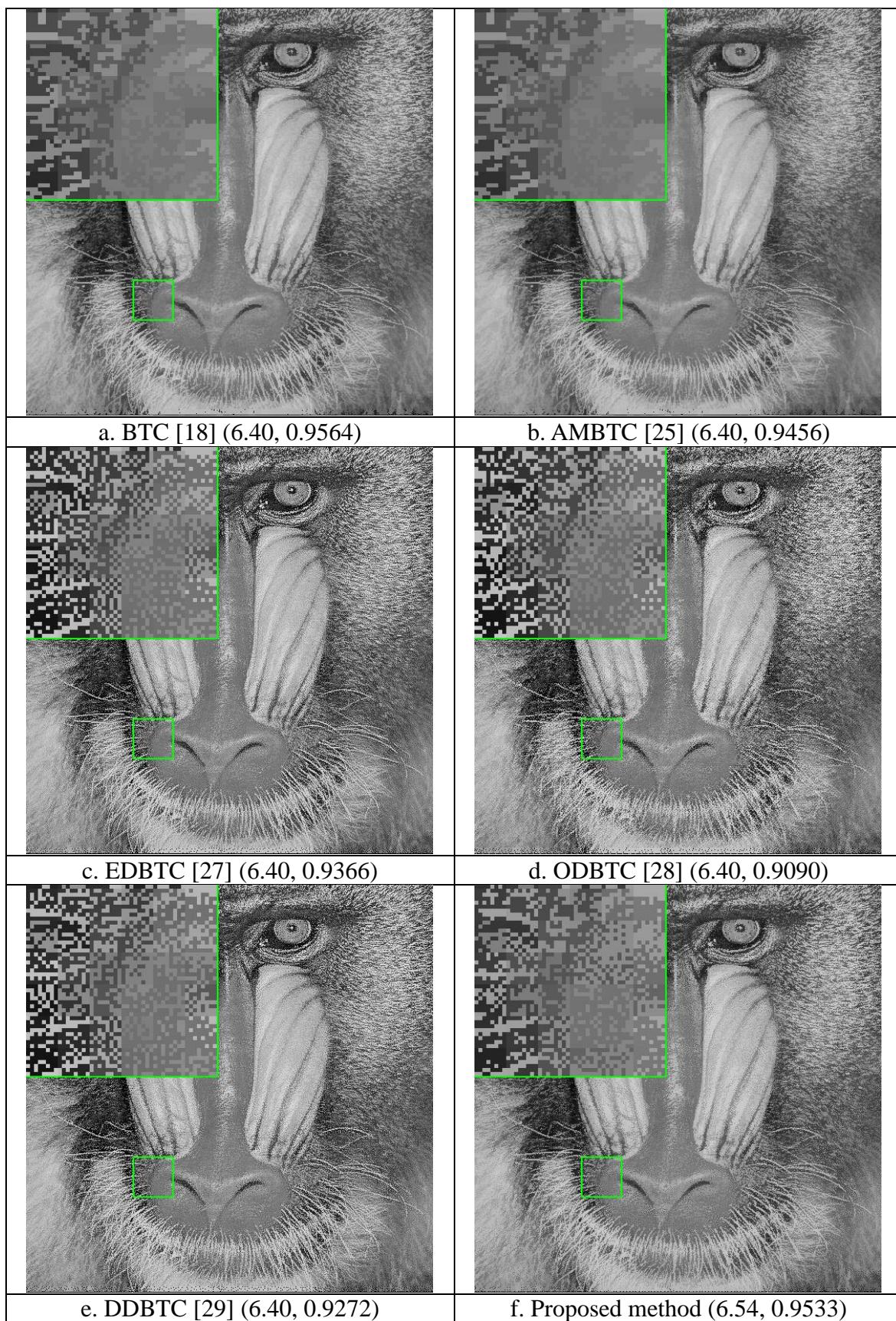


(c) Group 3 (7.3~7.7)

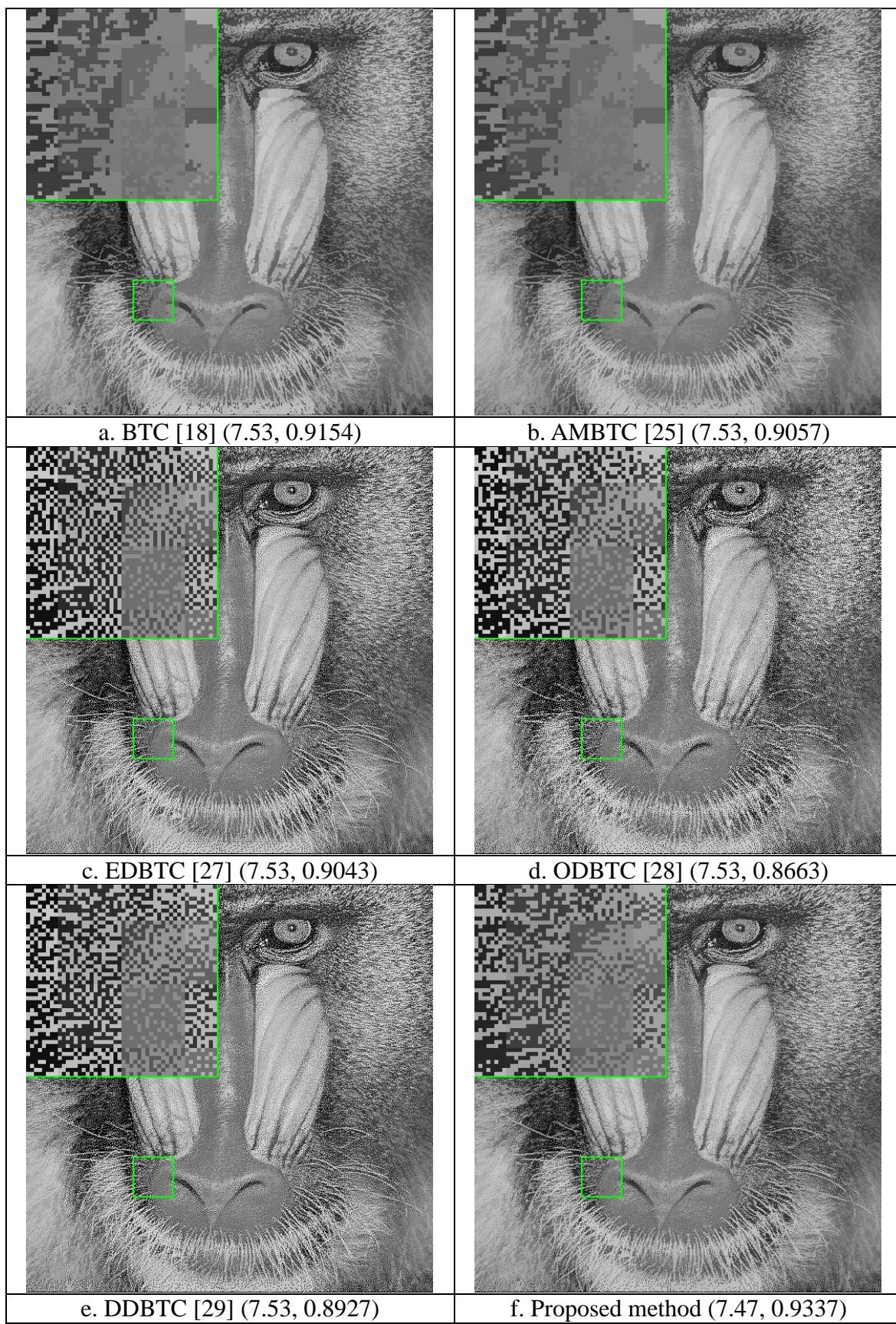
圖 5.13 所有技術壓縮灰階影像 1 後的重建影像，其中依照表 5.3 的分類



(a) Group 1 (3.8~4.2)



(b) Group 2 (6.2~6.6)



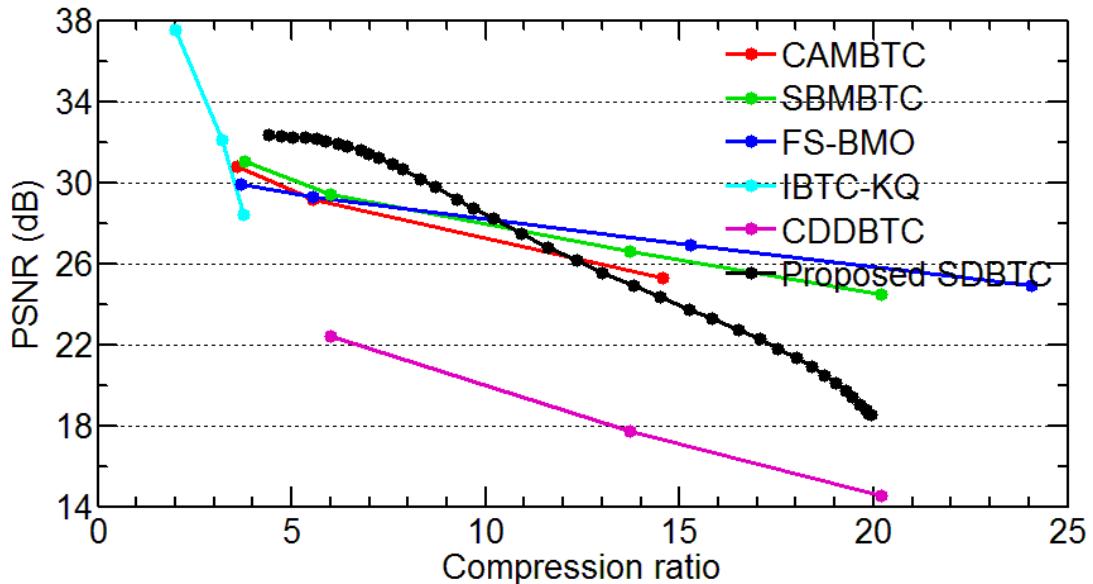
(c) Group 3 (7.3~7.7)

圖 5.14 所有技術壓縮灰階影像 2 後的重建影像，其中依照表 5.3 的分類

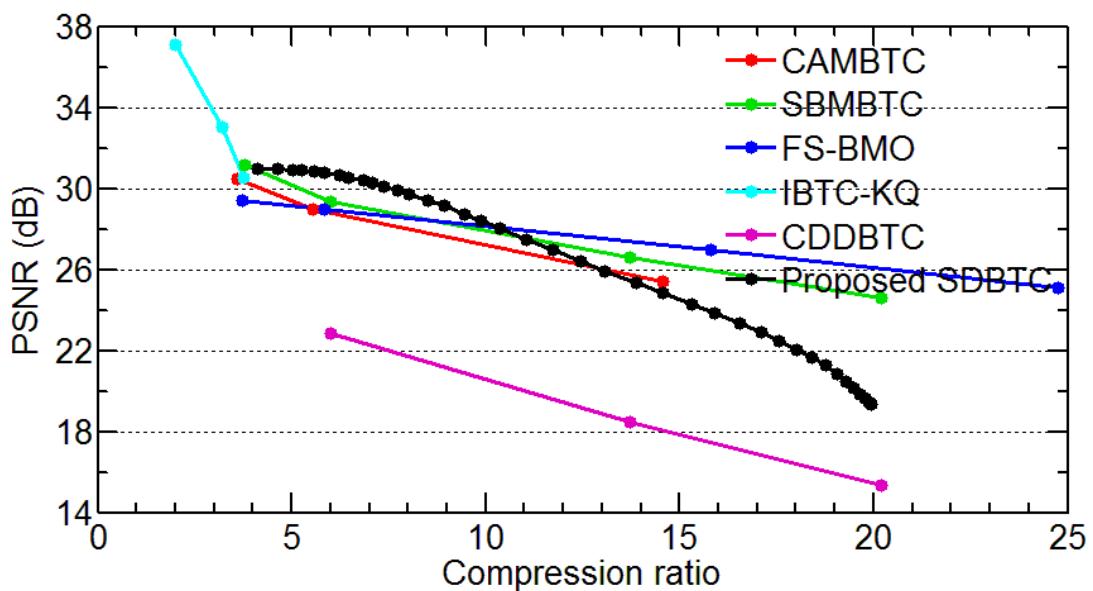
在彩色影像壓縮的比較中，除了訓練用的資料庫 Corel 1,000 [44] 及先前提到的 UCID [50]，再額外的增加兩個資料庫做為評估基準：1) Kodak [55]，2) Holidays [56]。圖 5.15 為所有技術壓縮影像在不同壓縮率(CR)下的峰值信噪比(PSNR)及針對彩色影像的特徵相似度指數(FSIMc) [49]，其中所有數值都是整個資料庫平均後的結果。特別在此說明，資料庫 Holidays 的測試中不考量 IBTC-KQ，因為資料庫的影像較大，且該技術的計算複雜度高，運算完整個資料庫需要耗費數周。

如圖 5.15 所示，雖然 IBTC-KQ 能夠達到最佳的影像品質，但代價是極低的壓縮率。此外，儘管 CAMBTC 有著較高的壓縮率，不過同時也降低了其重建影像的品質。且這兩個技術都有一個固定的缺陷，不論使用多大的區塊它們的壓縮率範圍都會受到限制，這個問題會在圖 5.15 中以”n/a”標記表示。然而，改良的技術並不會發生以上狀況，它能夠提供廣泛的壓縮率範圍從 5 到 20。另外，從圖中能發現到，當在峰值信噪比及特徵相似度指數在壓縮率分別為範圍 CR = [5,10] 及 CR = [5,18] 時，改良的技術與其他過往彩色區塊截斷編碼技術相比具有最佳的影像品質。

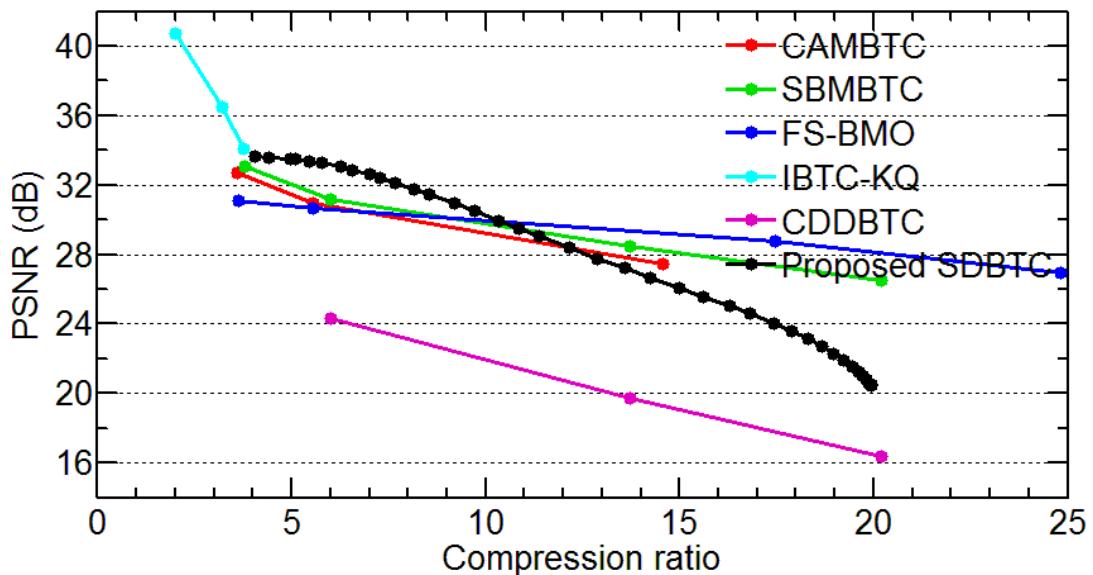
為了能更明確的看出實際在不同壓縮率下的差異，圖 5.16 展示出兩張實際影像壓縮後局部放大的結果，其中圖 5.16(b)表示的是表 5.4 分群的順序。如同在 5.6 節所說明，特徵相似度指數(FSIMc)較符合人眼實際感受，因此圖 5.16 中各技術重建影像有其對應的評估分數。在這些當中，(a) CAMBTC、(b) SBMBTC、(c) FS-BMO 等技術都有著明顯的區塊效應(Blocking effect)，尤其是在圖片”鸚鵡”的時候。而(e) CDDBTC 的結果則是遭受到嚴重的脈衝雜訊(Impulse noise)影響，如圖中分類 Groups 2~4。相較之下，改良技術 SDBTC 在分類 Groups 1~3 中產生最自然的壓縮影像。此外在分類 Group 4 中，SDBTC 能夠平衡區塊效應及脈衝雜訊，它的重建影像比起其他技術更加接近原圖。為了觀察更完整的壓縮結果，依照表 5.4 的分群，圖 5.17 及圖 5.18 分別展示圖 3.2 的彩色影像 1 及彩色影像 2 經各技術壓縮後的重建影像，括號中表示的是其壓縮率及其對應的 FSIMc 數值。



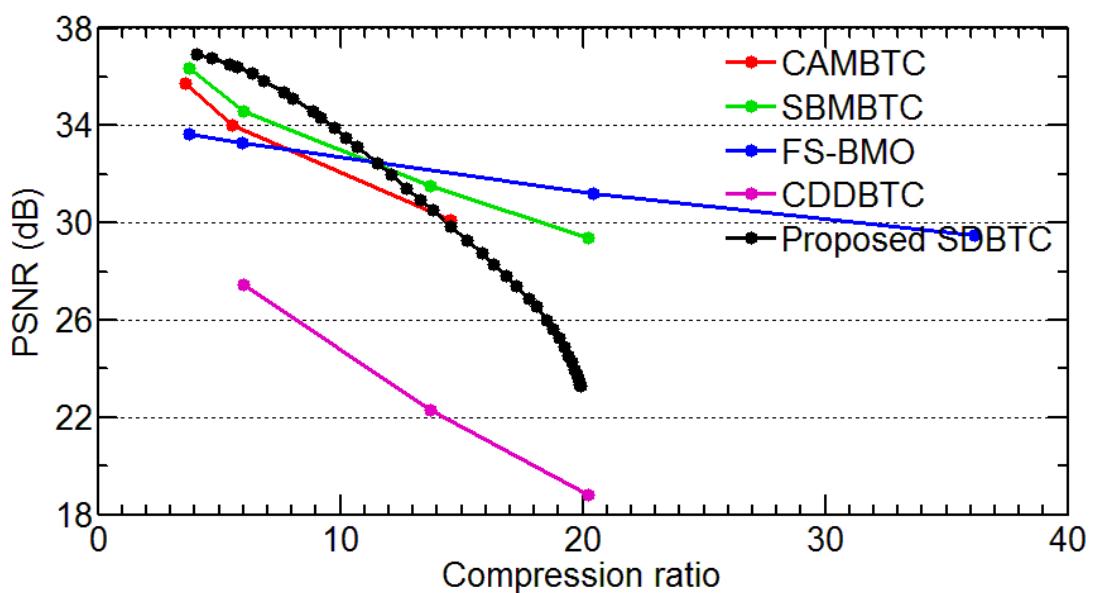
(a) 以 PSNR 評估資料庫 Corel 1,000 [44] 經彩色影像壓縮過後的結果



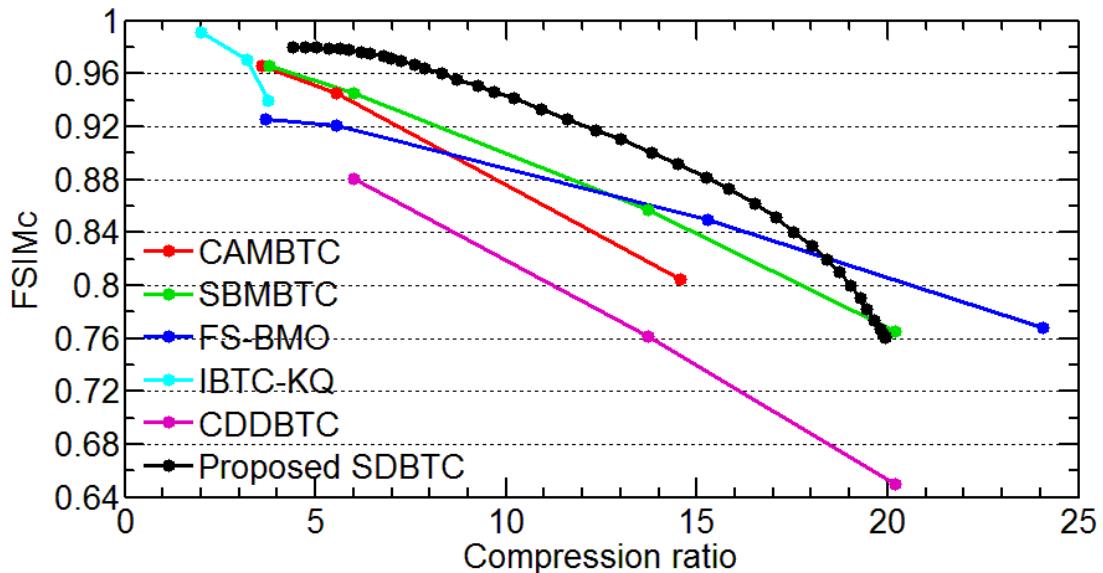
(b) 以 PSNR 評估資料庫 UCID [50] 經彩色影像壓縮過後的結果



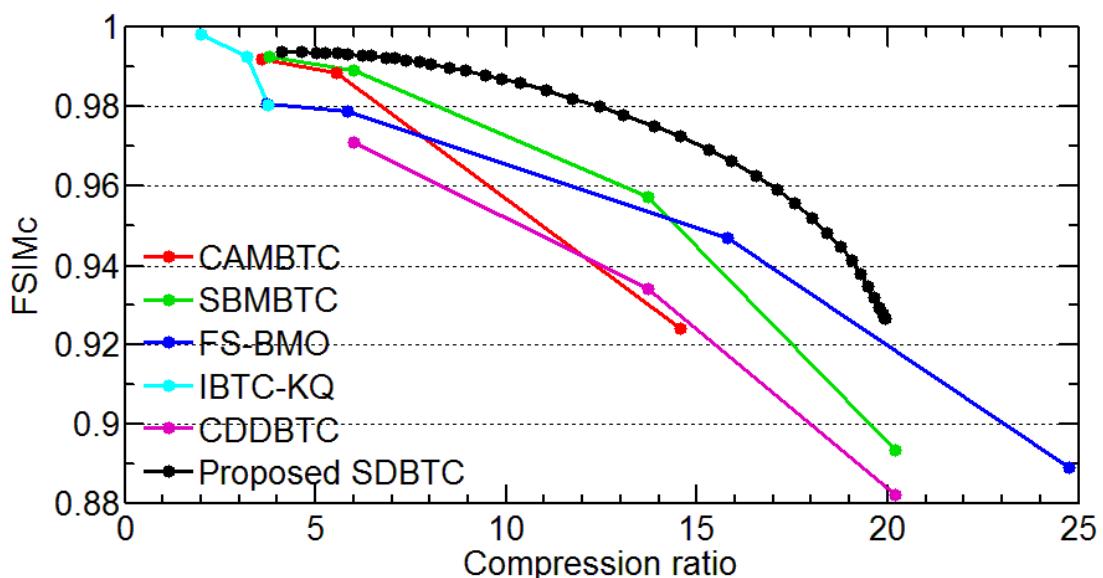
(c) 以 PSNR 評估資料庫 Kodak [55] 經彩色影像壓縮過後的結果



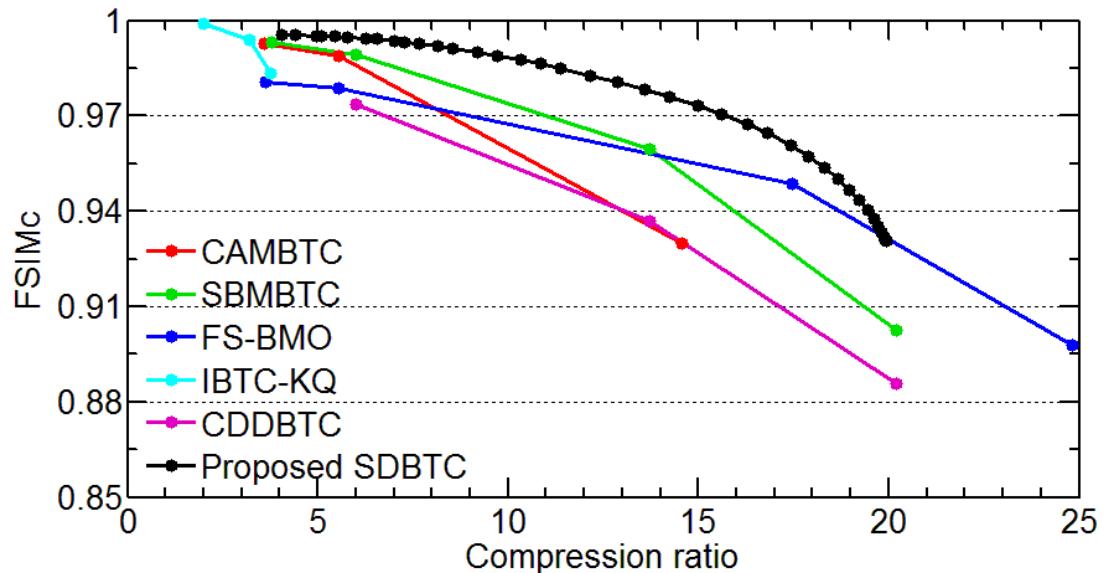
(d) 以 PSNR 評估資料庫 Holidays [56] 經彩色影像壓縮過後的結果



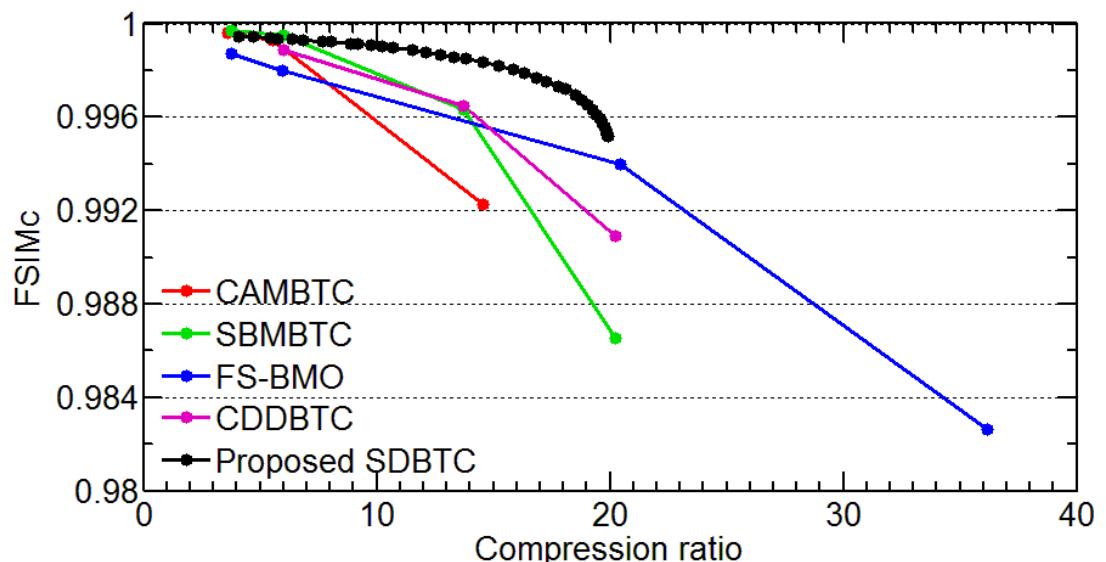
(e) 以 FSIMc 評估資料庫 Corel 1,000 [44] 經彩色影像壓縮過後的結果



(f) 以 FSIMc 評估資料庫 UCID [50] 經彩色影像壓縮過後的結果



(g) 以 FSIMc 評估資料庫 Kodak [55] 經彩色影像壓縮過後的結果

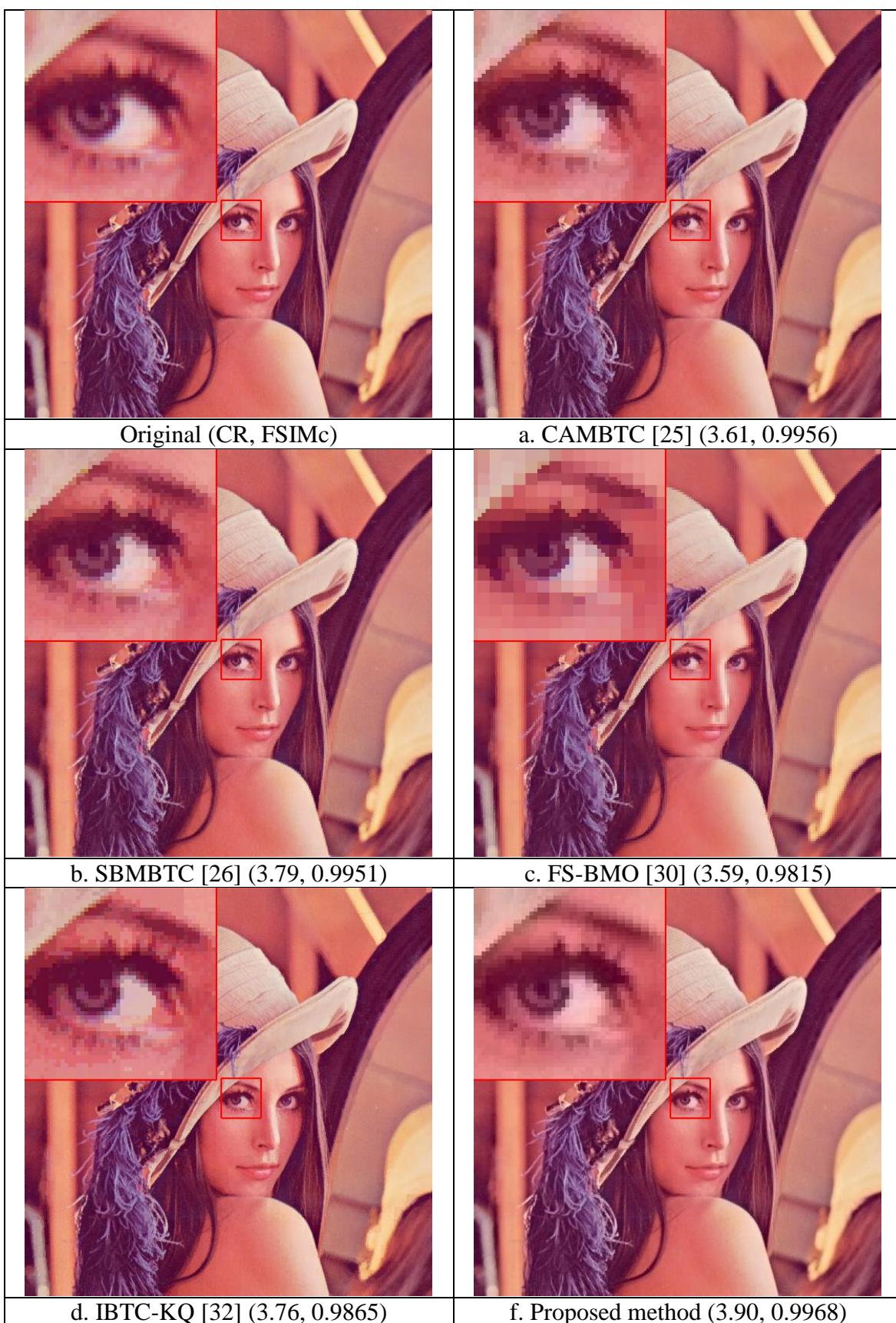


(h) 以 FSIMc 評估資料庫 Holidays [56] 經彩色影像壓縮過後的結果

圖 5.15 所有技術在不同資料庫中，不同壓縮率下的兩種影像評估：峰值信噪比(PSNR)及針對彩色影像的特徵相似度指數(FSIMc)的平均數值



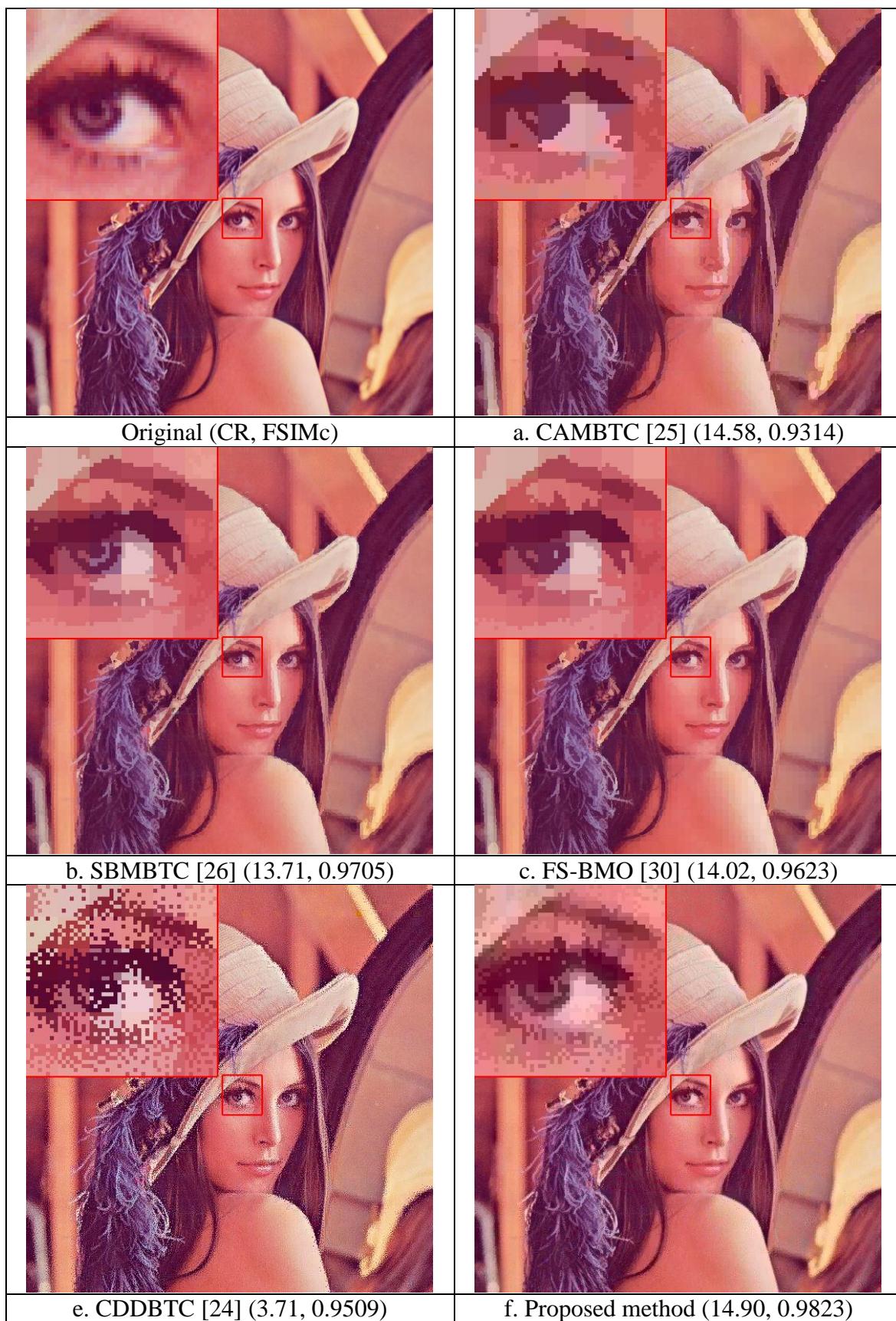
圖 5.16 所有技術壓縮後的局部影像



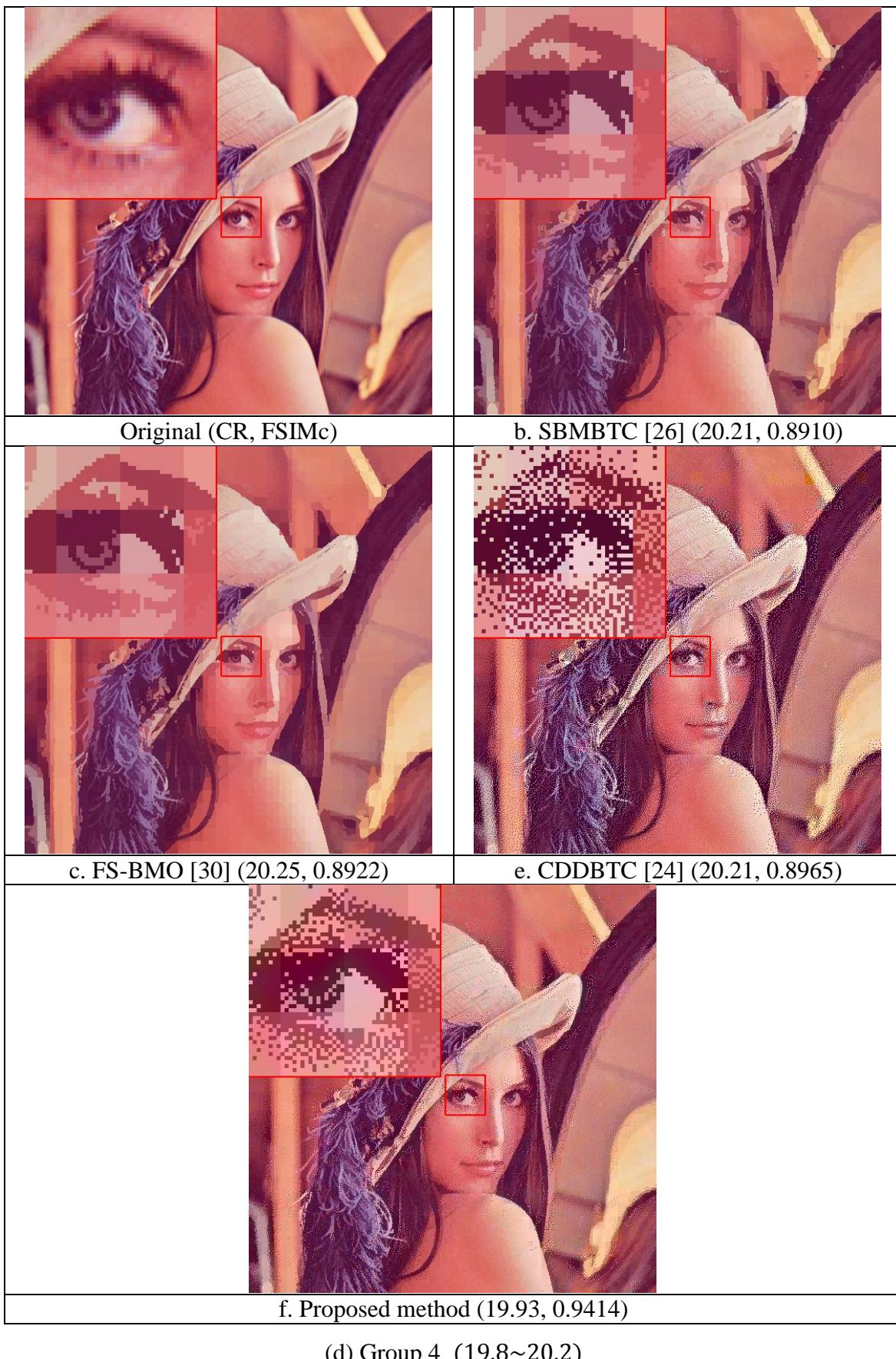
(a) Group 1 (3.6~4.4)



(b) Group 2 (5.4~6.6)



(c) Group 3 (13.7~15.3)

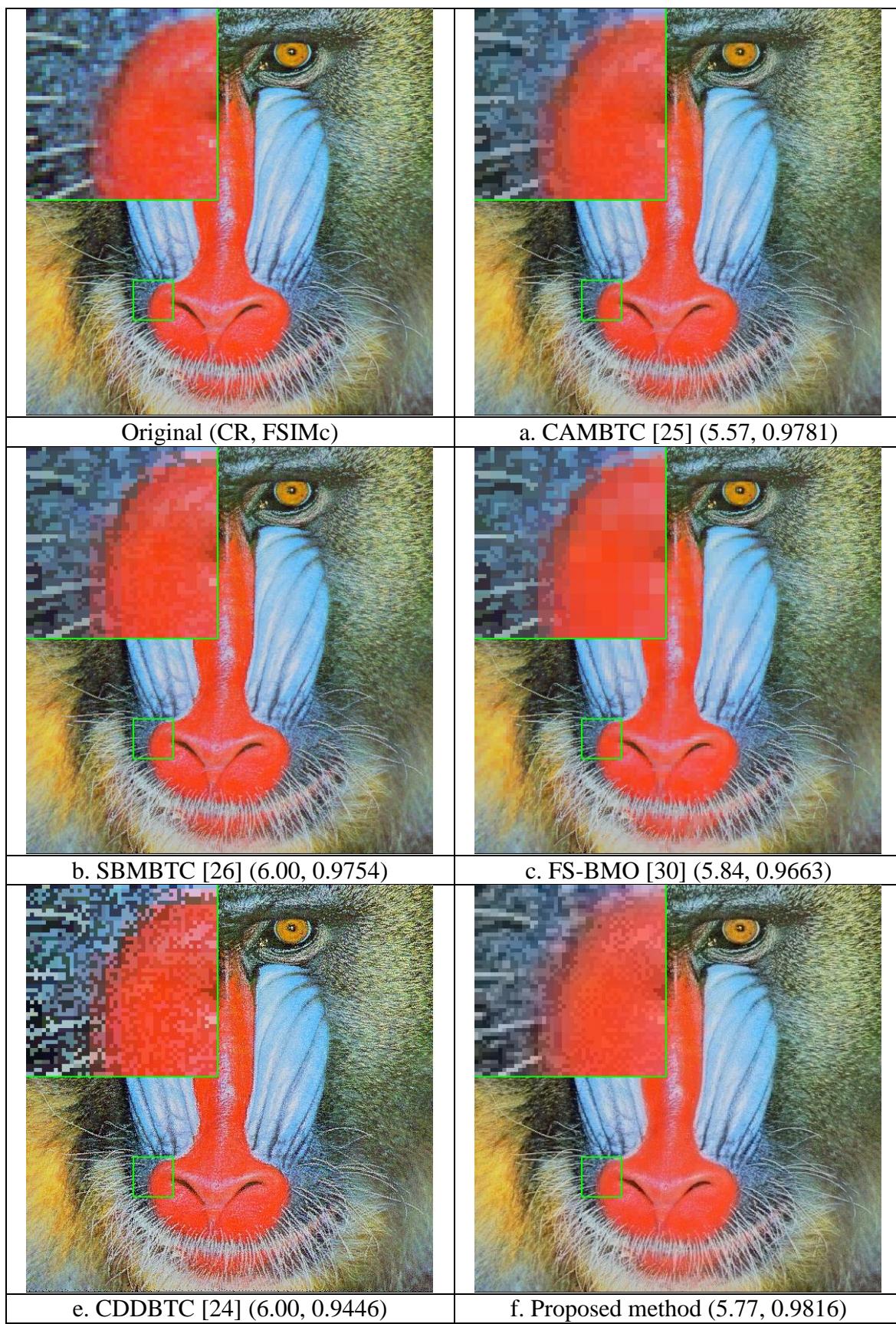


(d) Group 4 (19.8~20.2)

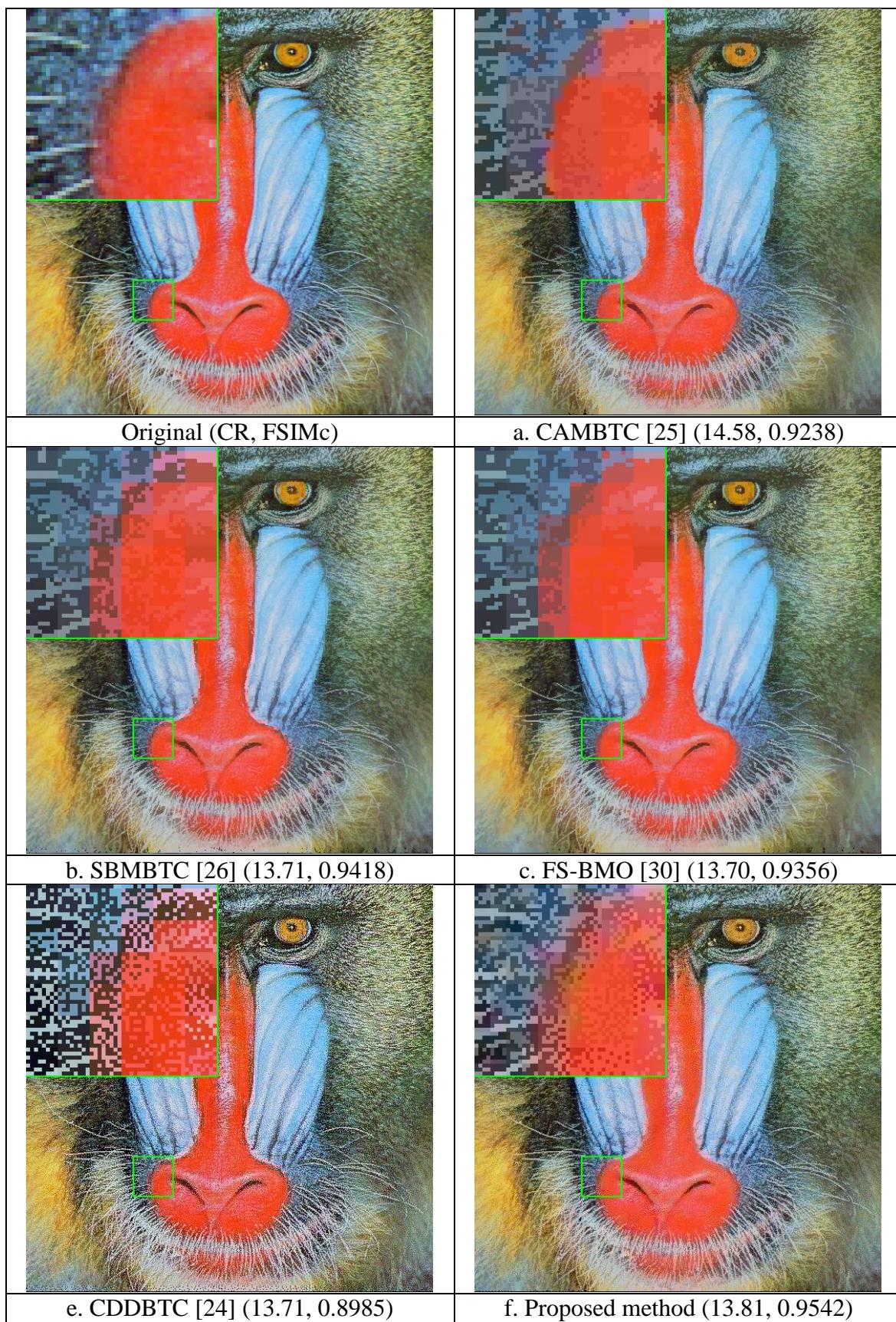
圖 5.17 所有技術壓縮彩色影像 1 後的重建影像，其中依照表 5.4 的分類



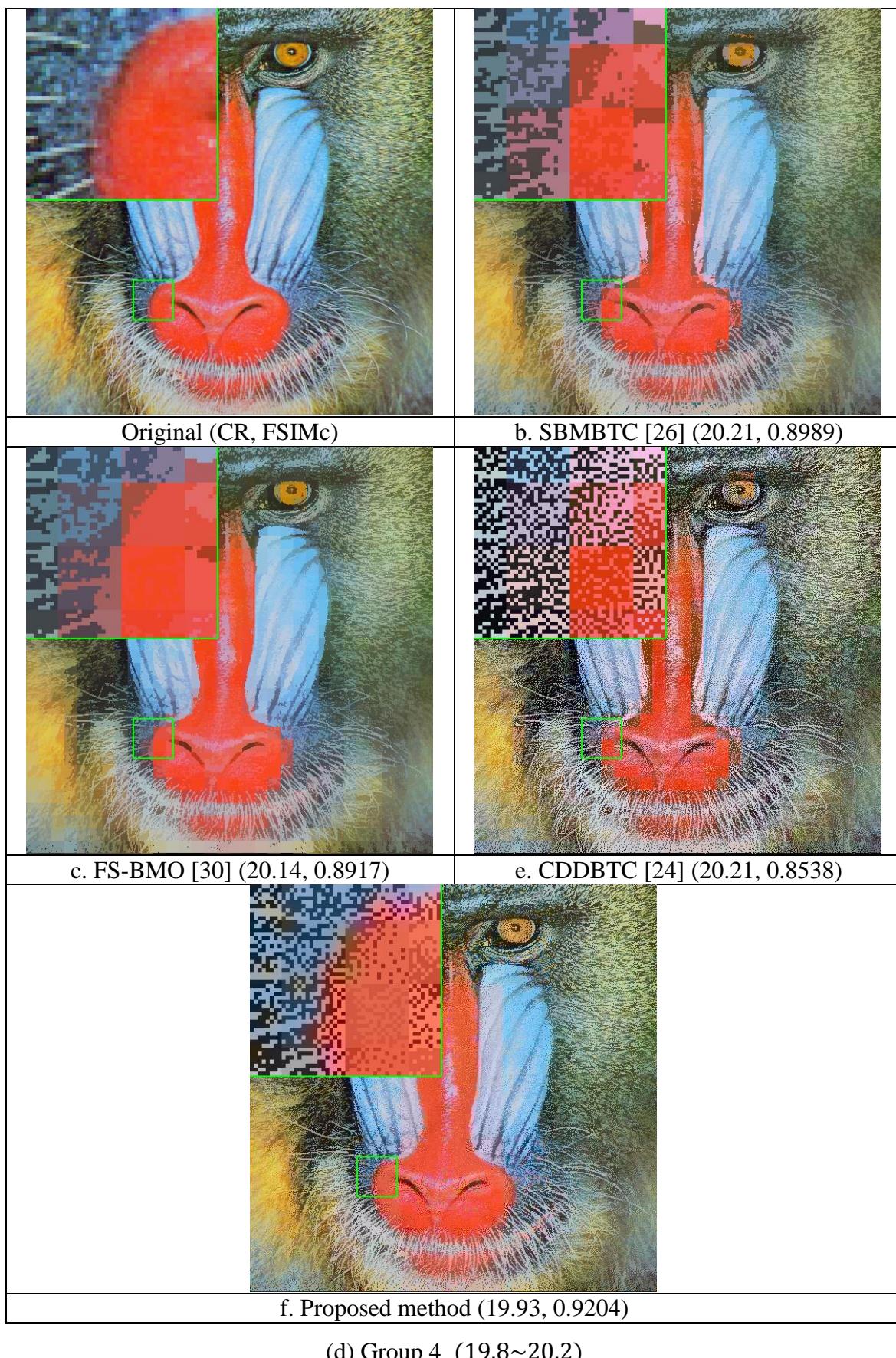
(a) Group 1 (3.6~4.4)



(b) Group 2 (5.4~6.6)



(c) Group 3 (13.7~15.3)



(d) Group 4 (19.8~20.2)

圖 5.18 所有技術壓縮彩色影像 2 後的重建影像，其中依照表 5.4 的分類

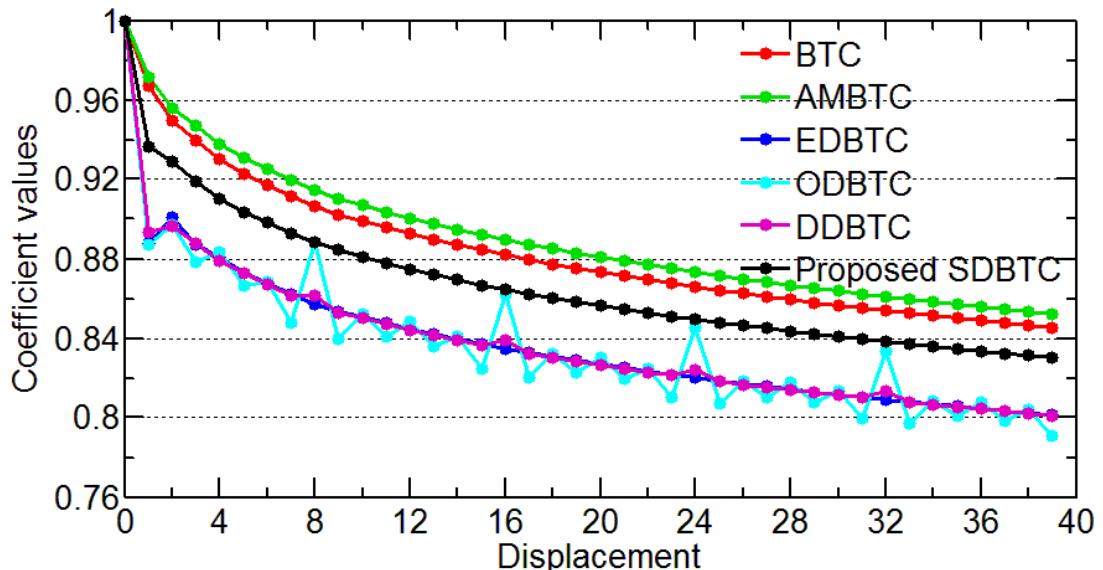
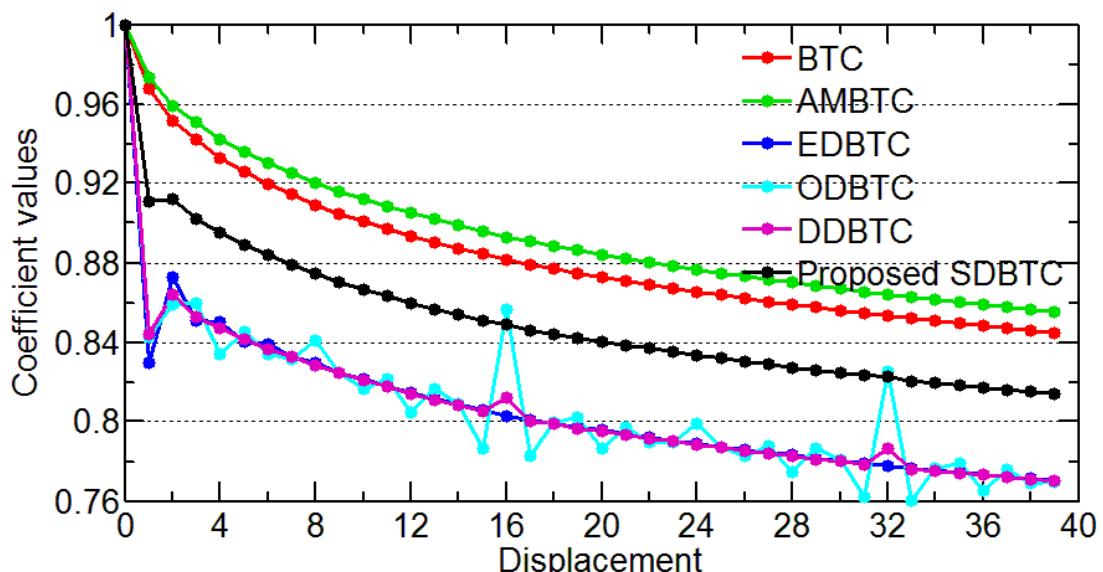
5.8.2 不自然紋理問題(Artifact problem)

一維的自相關(Auto-correlation)公式原本是用來評估半色調技術的週期性問題，為了應用在區塊截斷編碼的週期性評估上，公式修改如下：

$$ac_d = \frac{1}{P \times (Q-D)} \sum_{i=0}^{P-1} \sum_{j=0}^{Q-D-1} 1 - (y(i, j) - y(i, j + d)), \quad (5.14)$$

其中 $y(i, j)$ 表示壓縮後的重建影像； $d = [0, D]$ 表示與當前像素位置 (i, j) 的水平距離； P 及 Q 為原始影像的長寬。自相關呈現的是像素之間的關聯性，相似度越高則自相關數值越高。圖 5.19 為各技術透過公式(5.14)在資料庫 UCID [50] 中計算出來的平均自相關結果，其中設定 $D = 39$ 。如圖所見，ODBTC 隨著區塊的大小產生多個週期性的脈衝，這意味著 ODBTC 重建的影像其區塊之間有著極高相似度的結構。由於 DDBTC 各區塊使用著相同的 CM，因此它在圖 5.19 中同樣地有著週期性的脈衝。以上兩者都是因為在各個區塊都固定使用相同的遮罩產生抖動的點分佈，使得它們在相似結構區域當中的區塊都產生相同的重建結果。相較之下，由於 BTC 及 AMBTC 不使用抖動網點重建影像，因此它們的結果通常是一個平坦的結構，而改良的技術則是藉由不同的 CM 以及不同大小的區塊避免重建影像產生週期性圖案。

藉由兩個不同空間頻率模擬的人工圖案，圖 5.20 展示出各技術更具體的結果。同上面 BTC 及 AMBTC 的說明，雖然它們的結果影像沒有週期性圖案的問題，但區塊間明顯的不連續。圖 5.20(e)-(f)顯示 ODBTC 及 DDBTC 在兩個不同空間頻率下的重建影像，沿著水平方向可明顯地發覺到有極高相似度的結構，這會使人眼錯覺那是原始圖像上的圖案。與其它過往技術相較之下，改良的 SDBTC 技術產生一個更穩定且更均勻分佈的結構，提供了對人眼感受最自然的重建影像。

(a) 壓縮率在 6.4 時(區塊大小為 8×8)(b) 壓縮率在 7.53 時(區塊大小為 16×16)圖 5.19 各個技術重建影像經一維的自相關計算結果，其中 $D = 39$

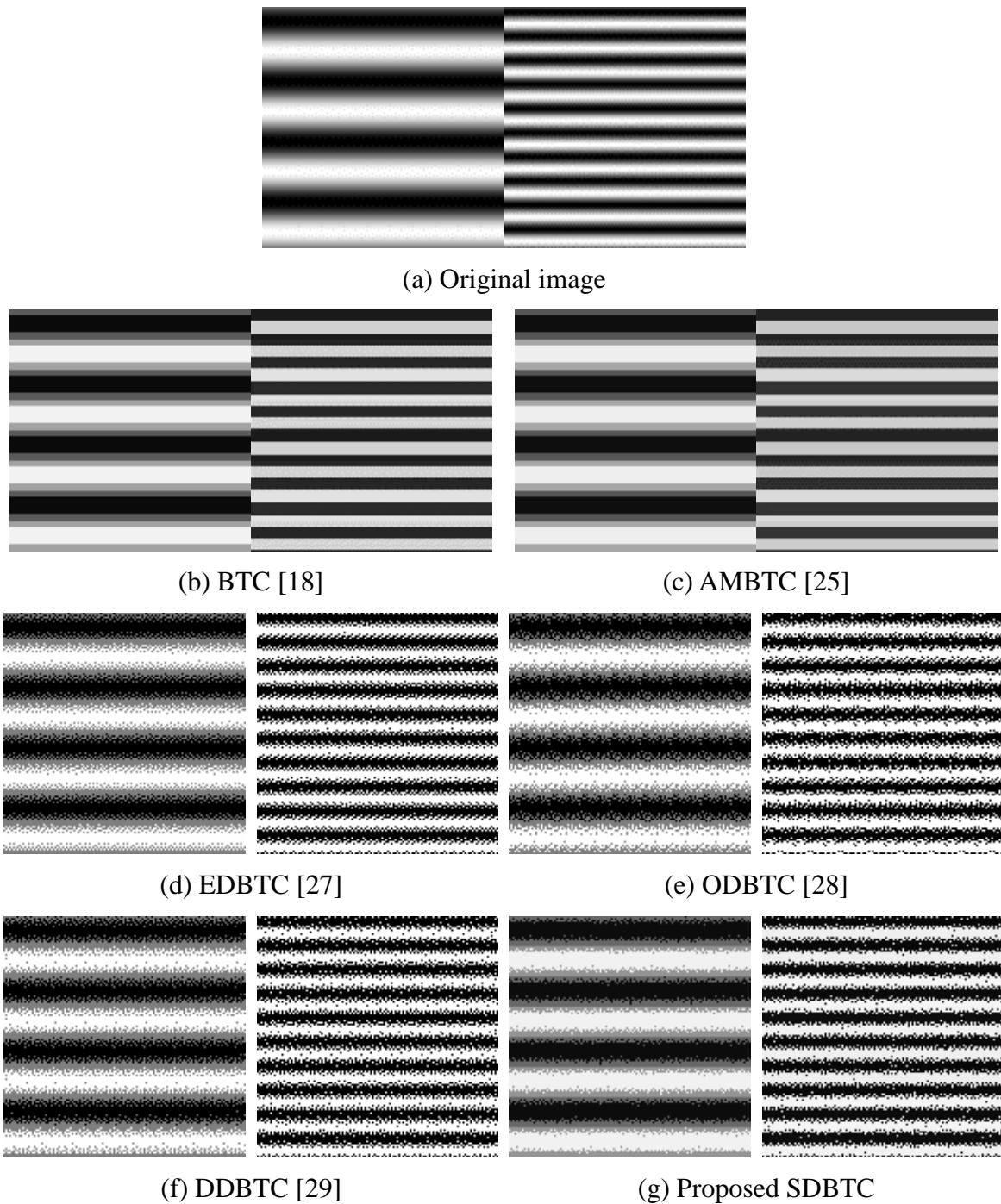


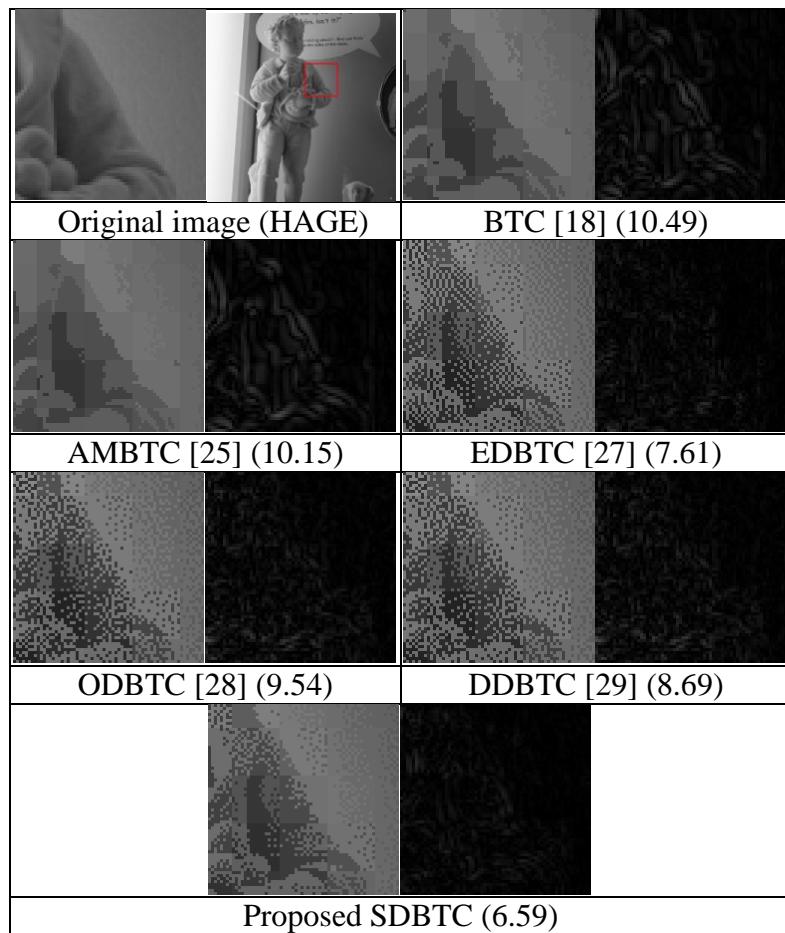
圖 5.20 128 × 128 大小的人工影像，在不同空間頻率下各技術壓縮後的結果

為了驗證改良方法是否平衡了區塊效應及脈衝雜訊的問題，在此提出一個新的評估方法名為人眼視覺絕對梯度誤差(Human-visual absolute gradient error, HAGE)，其主要目的是用來計算可視的不自然紋理，特別是對區塊效應及脈衝雜訊，詳細的公式如下：

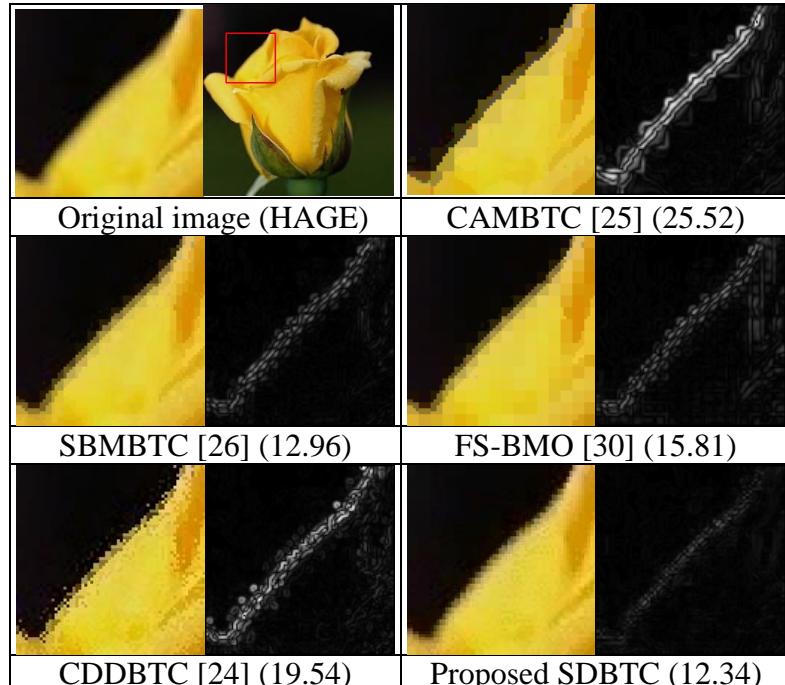
$$HAGE = E(\|S(G(X)) - S(G(Y))\|), \quad (5.15)$$

其中，運算子 $G(\cdot)$ 、 $S(\cdot)$ 及 $E(\cdot)$ 分別表示高斯平滑、梯度檢索及期望值；而運算子 $\|\cdot\|$ 則表示歐幾里德正規化(Euclidean norm)，若為彩色影像則是運算所有 Lab 彩色模型的通道； X 、 Y 分別表示原始及壓縮後的影像。在此，高斯平滑使用的遮罩大小設定為 7×7 ，而梯度檢索使用的技術為索貝爾運算子(Sobel operator)。一般來說，越小的 HAGE 數值表示不自然紋理對人眼視覺的影響越少。

圖 5.21 顯示各技術在相同壓縮率下的重建影像，經歐幾里德正規化 $\|S(G(X)) - S(G(Y))\|$ 後輸出的結果，括號內為整張影像的 HAGE 數值。圖 5.21(a)為灰階壓縮技術在壓縮率 7.5 左右時的結果，如表 5.3 的 Group 3 設定。從圖中可以發現，BTC 及 AMBTC 由於受到偽輪廓及區塊效應的影響，因而產生許多冗餘的輪廓線，使得它們的重建影像與原始影像落差較大。而以半色調為基礎的 BTC 技術則是受到脈衝雜訊影響，使得重建影像變得較為雜亂。相較之下，改良的 SDBTC 維持銳利的影像並避免受到過度脈衝雜訊的影響，從其歐幾里德正規化的輸出圖中可明顯發現 SDBTC 比過往技術的梯度誤差更少。圖 5.21(b)為彩色壓縮技術在壓縮率 6.0 左右時的結果，如表 5.4 的 Group 2 設定。由於 CAMBTC、SBMBTC 及 FS-BMO 都繼承了 AMBTC 的特性，它們的結果都受到明顯區塊效應的影響，而 CDBTC 繼承 DDBTC 的特性，結果影像上有著雜亂的脈衝雜訊。與過往技術相比，SDBTC 對彩色影像壓縮仍然能夠避免掉上述的問題並有著最佳的重建影像。為了更直接的觀察在不同壓縮率下各個技術受到不自然紋理問題影響的程度，圖 5.22 呈現了以資料庫 UCID 測試後平均的 HAGE 數值。可以明顯地發現在各個壓縮率下，改良的 SDBTC 比起其他技術都有著更佳的優勢。

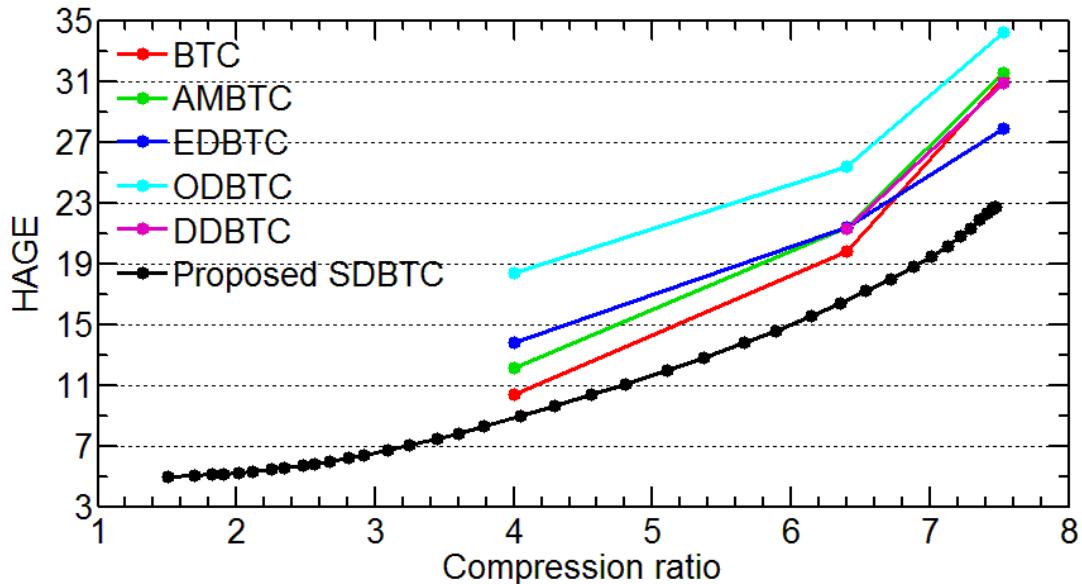


(a) Grayscale image compression

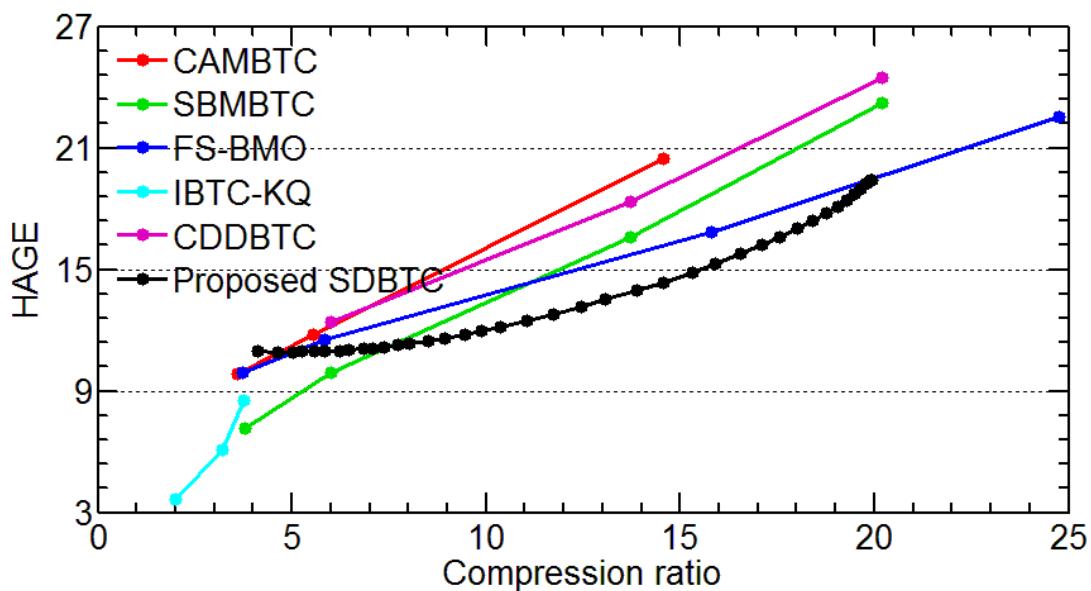


(b) Color image compression

圖 5.21 各技術在相同壓縮率下的重建影像其局部放大圖及其對應的梯度平滑影像，
括號內為整張影像的 HAGE 數值



(a) 以 HAGE 評估資料庫 UCID [50] 經灰階影像壓縮過後的結果



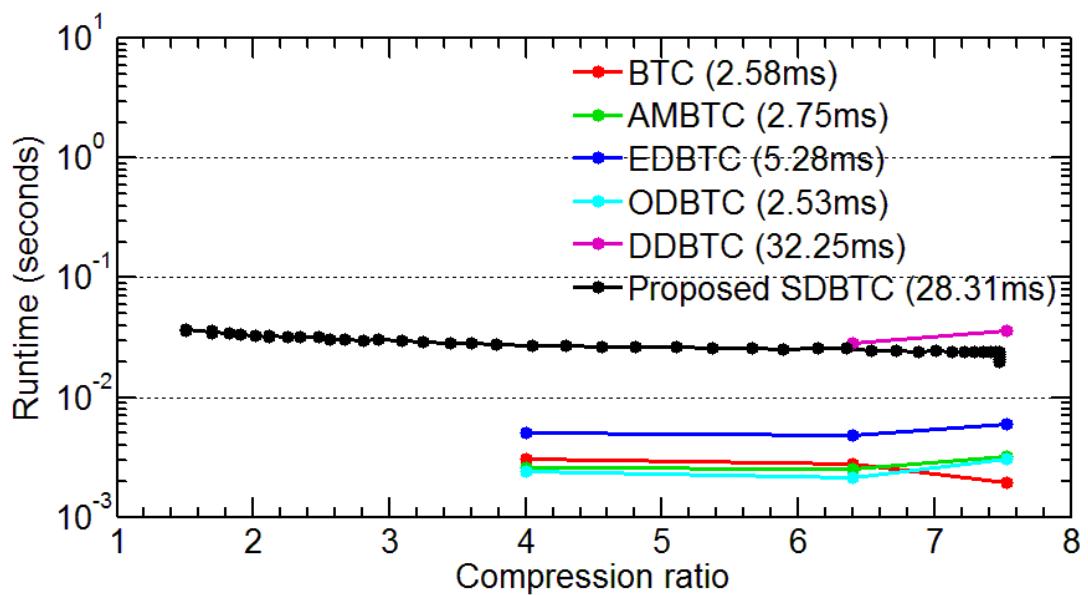
(b) 以 HAGE 評估資料庫 UCID [50] 經彩色影像壓縮過後的結果

圖 5.22 各技術在不同壓縮率下以 HAGE 評估其損壞程度

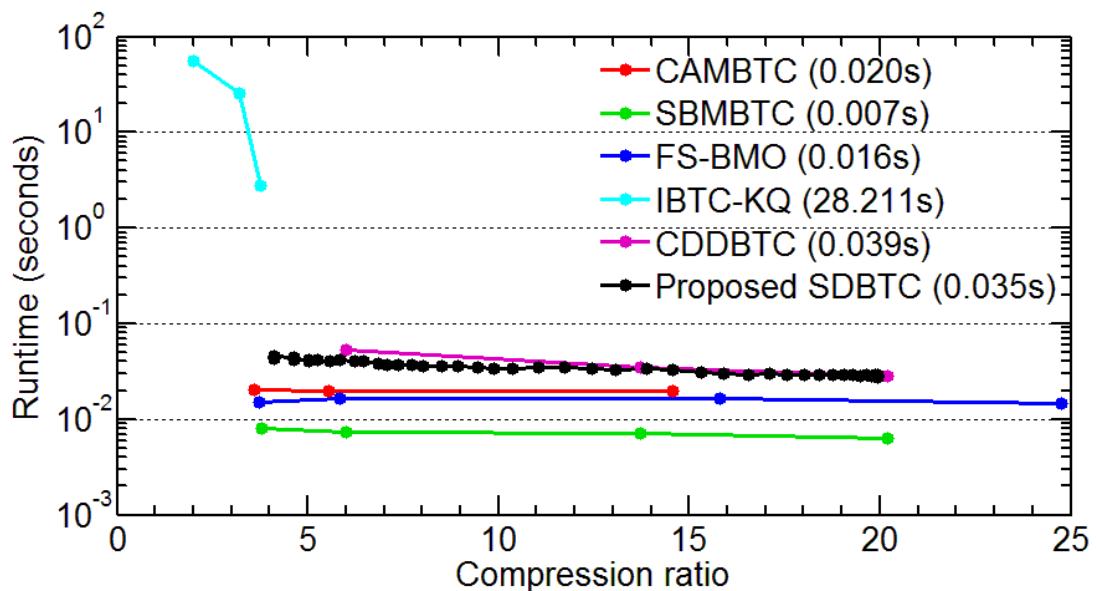
5.8.3 處理速度(Runtime)

圖 5.23(a)及(b)所示分別為灰階影像壓縮技術及彩色影像壓縮技術在不同參數設定下的平均處理速度，測試時僅使用一個核心處理，此外數據是整個資料庫 UCID [50] 的平均結果。在此，所有技術都是使用 Visual Studio C++ 在硬體設備 i7-4790 3.60GHz CPU 及 8GB RAM 上模擬。值得注意的是，由於區塊可平行處理的特性，所有彩色及除了 EDBTC 之外的灰階區塊截斷編碼壓縮技術，都能夠藉由多個核心處理來加速執行。舉例來說，若有無限個核心能夠平行處理各個區塊，改良技術的運算速度能夠比 EDBTC 快上數十倍。由於過往技術用簡單的判斷來壓縮影像，而改良技術的每個像素需要額外的將誤差擴散至周圍像素，因此過往技術比改良的 SDBTC 快了很多。以 EDBTC 來說，其速度上的優勢是由於它擴散的範圍比改良技術的八個擴散像素更少(在圖中顯示的 EDBTC 是使用 Floyd-Steinberg 的誤差擴散矩陣，擴散範圍只有四個像素)。雖然改良技術是基於 DDBTC 技術且運算更為複雜，但由於使用了前面提到的積分影像使得 SDBTC 的處理速度與 DDBTC 差異不大。

如圖 5.23(b)所示，在所有彩色影像壓縮技術中，IBTC-KQ 需要更多的時間運算，因為 k-means 分群演算法的計算複雜度高，且當越多區塊時(使用的區塊小且壓縮率較低)會需要耗費更多的時間。相較之下，其餘的技術如 CAMBTC、SBMBTC 及 FS-BMO 雖然能夠比改良的 SDBTC 有更快的處理速度，但它們壓縮後的結果容易受到嚴重的區塊效應影響，如圖 5.16。CDDBTC 雖然藉由 SBM 技術只計算了一個位元圖並有著較快的處理速度，但改良的 SDBTC 也藉由縮減取樣(Downsample)的方式減少了運算量，所以與 CDDBTC 有著差不多的處理時間。總結上述不同層面的考量，改良的 SDBTC 比起其它低複雜度的區塊截斷編碼技術更具競爭力。



(a) 評估資料庫 UCID [50] 經灰階影像壓縮所需的時間



(b) 評估資料庫 UCID [50] 經彩色影像壓縮所需的時間

圖 5.23 各技術在不同參數設定下的平均處理速度

5.9 小結

一個新的區塊截斷編碼影像壓縮技術名為自適應點擴散區塊截斷編碼(Self-adaptive dot-diffused block truncation coding, SDBTC)被提出。與過往技術相比，它能夠獲得有明顯較佳影像品質的壓縮影像。此外，使用者能夠藉由控制 φ 使 SDBTC 的重建影像符合使用者期望的影像品質。在此技術的演算法設計中，參數都經過優化處理，使之能夠自行平衡區塊效應(Blocking effect)及脈衝雜訊(Impulse noise)。儘管過往的彩色壓縮技術 CCC 及 FS-BMO 藉由單一位元圖技術(SBM)大幅地提升它們的壓縮率，但同時也引入了明顯的區塊效應，而本技術能夠藉由縮減取樣避免上述問題的發生，同時能夠降低運算量並維持不錯的壓縮品質。如同上述研究中實驗結果所顯示，改良的 SDBTC 技術能夠成功地權衡不自然紋理問題(Artifact problem)並提升影像的品質。相較於過往技術，在相同壓縮率下 SDBTC 受到可視不自然紋理的影響最低。除此之外，在影像評估技術 FSIM 的評估之下，SDBTC 在各個壓縮率情況下幾乎都有著最佳的影像品質。因此，本研究所提出的 SDBTC 技術是個極具競爭力的影像壓縮技術，特別是對於其中需要低功率處理裝置或者高解析度的應用。

第六章 多色調高效率藍雜訊抖動法

在此章節，一個新的多色調技術名為多色調高效率藍雜訊抖動法(Efficient blue-noise dithering for multitone, M-EBND)在此研究中被提出，其目的是為了改善過往多色調藍雜訊抖動法處理效率的問題並維持其優異影像品質的優勢。一個演算法名為篩選法(Screening) [57]的技術在此用來簡化繁複的運算，其中，篩選法會用到的臨界值矩陣(Threshold matrix)藉由挑選的半色調方法能夠解決在極端灰階的帶狀效應問題。此外，藉由挑選適當的墨水濃度(Ink concentration)曲線，改善迭代式高對比的問題。本論文所提出的方法之流程圖如圖 6.1 所示，是以 4.4 節介紹的多色調藍雜訊抖動法(Multitone blue-noise dithering, M-BND) [39]為基礎做改良。本技術的流程如下，先將輸入的灰階影像 X 藉由墨水濃度曲線轉換成 $q - 1$ 張分層影像 X_i ，接著藉由篩選法的臨界值矩陣轉將分層影像換成具有點繼承特性的半色調影像 H_i ，統計各個分層半色調影像 H_i 的像素輸出總和，將總和乘上量化色階的間距即可得到一張多色調影像。舉例來說，三色階 $\mathbf{Q} = \{0, 0.5, 1\}$ ，其量化色階間距為 0.5，當一張分層半色調的輸出為 1 時，則相同位置的多色調影像輸出 0.5×1 ；而兩張都有輸出則為 0.5×2 ，如同公式(4.18)-(4.19)。在接下來的小節中將介紹本論文提出的改良部分，並且在實驗結果中將會詳細比較各種方法之效能。

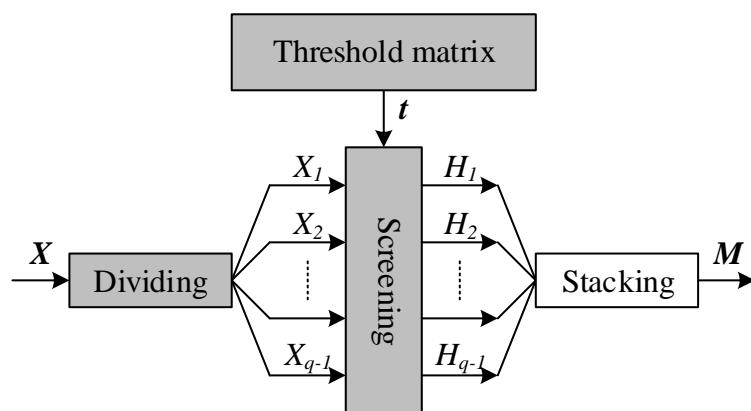


圖 6.1 多色調高效率藍雜訊抖動法的流程圖

6.1 墨水濃度(Iink concentration)

如圖 4.13、圖 4.14 及圖 4.15 (c)所示，多色調藍雜訊抖動法(M-BND)的量化影像高度依賴其半色調的分層影像，每個量化色階的顯示則是受到墨水濃度曲線的影響。然而，在多色調藍雜訊抖動法的文獻[39]中並沒有討論如何取得最佳的墨水濃度曲線。因此，在此章節詳細地討論它對多色調量化影像的影響。

為了分析過往技術，圖 6.2 顯示的是各技術在不同純色影像中統計的墨水濃度比率，而圖 6.3 所示為 512×96 的 Ramp 影像經過多色調轉換後的各技術結果。其中，圖 6.2 及圖 6.3(a)-(c)使用的是三色階的多色調直接二元搜尋法(M-DBS)、直接多位元搜尋法(DMS)及色調替代之誤差擴散法(M-TRED)。從圖中可以發現，在灰階 $g = 40$ 及 $g = 215$ 左右時，M-DBS 的結果與原始 Ramp 影像有明顯的差異。這是因為根據人眼視覺系統(Human visual system, HVS)的研究文獻[43]分析，人類的眼睛觀看具有相同頻率的影像時，對高對比的影像較敏感。舉例來說，兩個相同頻率不同對比的影像如圖 6.4(a)-(b)，兩張影像的平均灰階都為 128。雖然具有相同的平均灰階，但可以明顯的發現到較低對比的影像如圖 6.4(b)較接近灰階 128。因此，從圖 6.2 中可以發現多色調技術 DMS 及 M-TRED 比起 M-DBS 使用更多的中間色調，這表示這兩技術產生的多色調影像對比度較低，人眼的感受會覺得它們的結果與原始影像更為貼近，如圖 6.3(b)-(c)。此外，為了更加地瞭解使用不同數量中間色調對於量化影像的影響，使用指數轉換(Power law transformation)模擬三色階累積的墨水濃度曲線，公式如下：

$$S_2[g] = 1 - \frac{[256-(g+1)]^\gamma}{256}, \text{ where } g = 0, \dots, L-1, \quad (6.1)$$

$$S_3[g] = \frac{(g+1)^\gamma}{256} \mid_{g=0}^{L-1}, \text{ where } g = 0, \dots, L-1, \quad (6.2)$$

其中， $\gamma \in \mathbb{R}$ 主要是用來控制曲線，灰階 $g \in [0, L-1]$ 。如同上述說明，由於 γ 正比於中間色調的數量，越低的 γ 表示會使用越少的中間色調並導致較高的對比，使得人眼覺得結果與原始影像有落差。為了更明確的瞭解，圖 6.3(d)-(e)展示了改良技術使用不同 γ 模擬的累積墨水濃度所產生的結果，其中，由於 $\gamma = 1.6$ 及 $\gamma = 2.4$ 時的墨水濃度趨近於 M-DBS 及 M-TRED 的墨水濃度，因此在圖中展示時使用低於 1.6 及高於 2.4 的

數值觀察更多的變化。雖然圖 6.3(e)具有較低的對比，但結果仍然與原始影像有些差異，由於使用了過多的中間色調，使得中間色調與極端色調的頻率落差太大，導致影像上有類似帶狀效應的問題。

綜合上述討論，使用較多的極端色調會使得多色調結果影像產生高對比的圖案並使視覺品質降低；反之，由於使用過多的中間色調則會導致不自然的狀況發生。簡而言之，這是一個高對比及帶狀效應的權衡問題。然而，由於現存評估技術中沒有能夠完美模擬人眼視覺評估的方法，所以無法以訓練的方式得到最佳的墨水濃度曲線。因此在本論文中，使用從目前最新最佳的 M-TRED 技術所統計出來的墨水濃度做為暫時的最佳解。

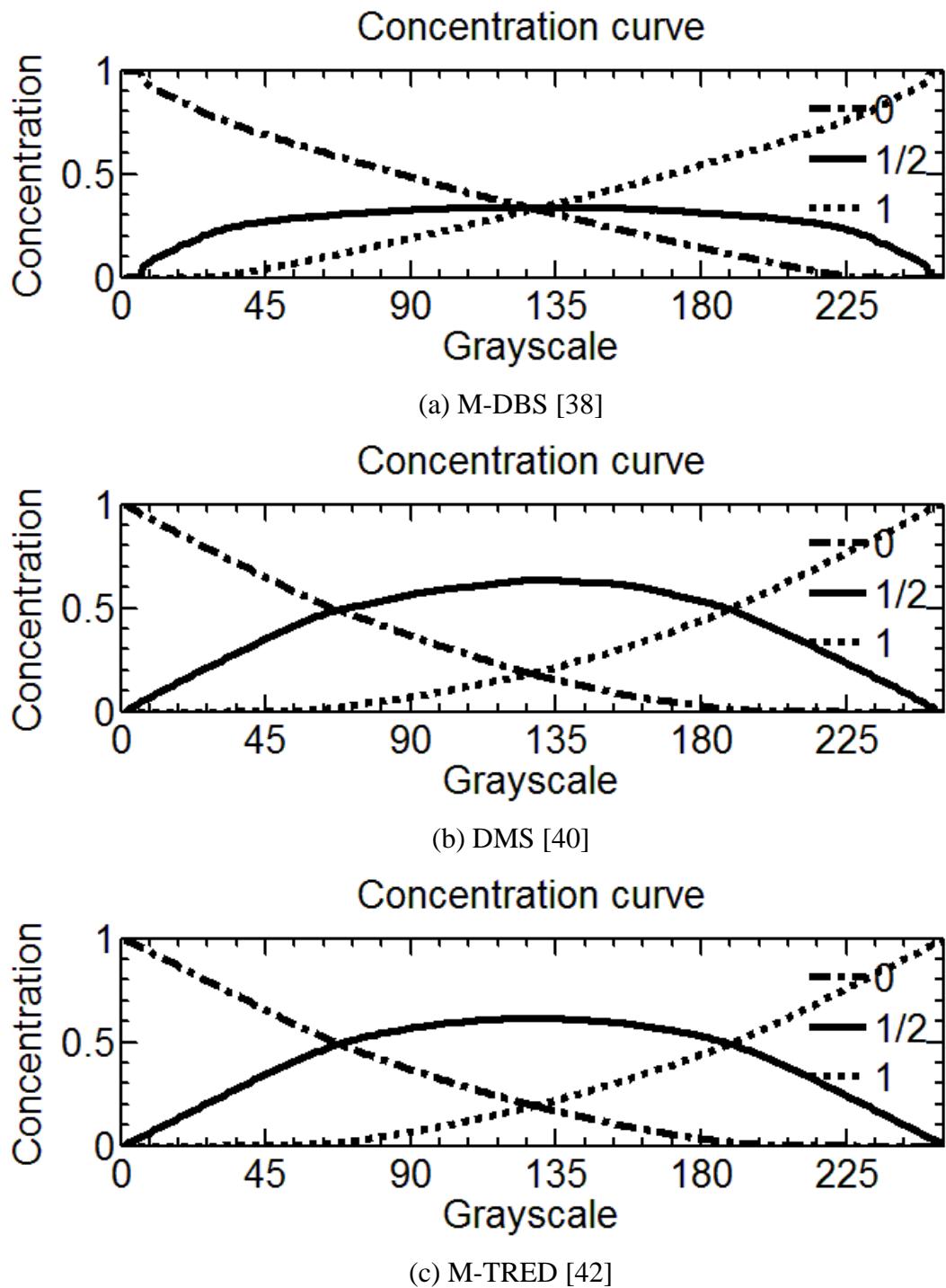


圖 6.2 在不同純色影像中，三色階多色調技術(a) M-DBS、(b) DMS 及(c) M-TRED 所統計的墨水濃度比率

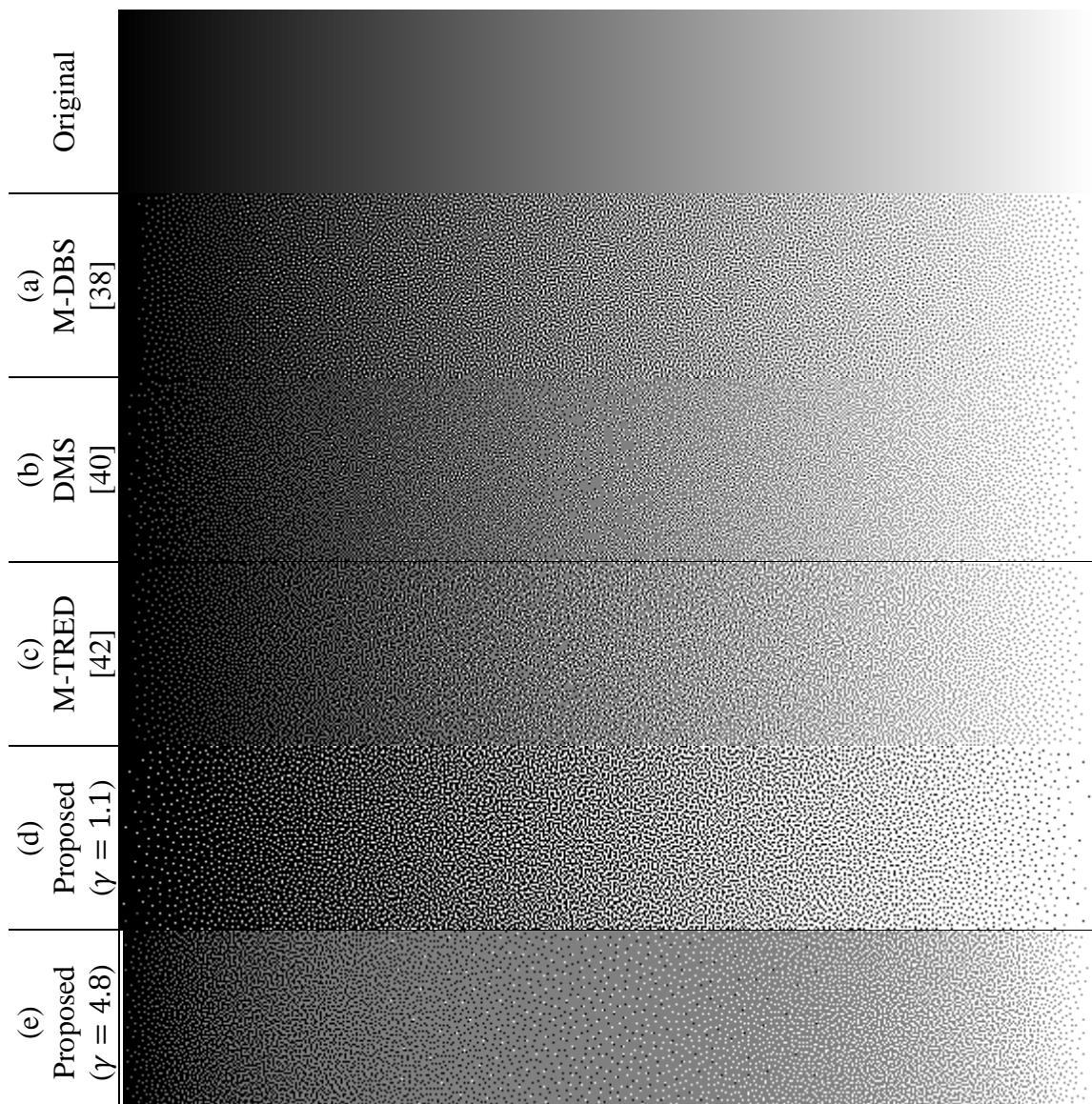


圖 6.3 512×96 的 Ramp 影像經三色階的多色調技術轉換後的量化影像，(a) M-DBS、(b) DMS、(c) M-TRED、(d)-(e) Proposed method 使用不同模擬墨水濃度的結果

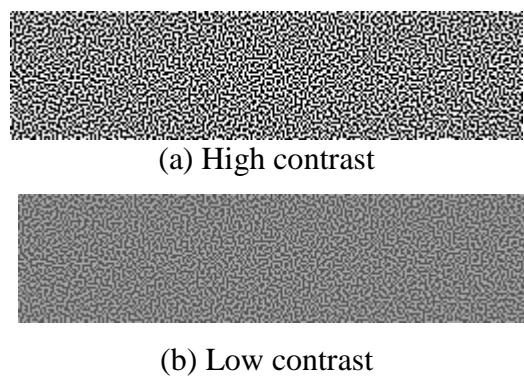


圖 6.4 在相同頻率下，相同平均灰階但不同對比的兩色調影像

6.2 篩選法(Screening)

雖然 M-DBS 延伸自 DBS 技術，但僅有一個輸出色階能夠維持 DBS 的點分佈特性。舉例來說，如圖 6.3(a)，三色階的 M-DBS 多色調結果只有灰色的輸出色階在不同灰階時都能保持均勻的點分佈，而白色及黑色的輸出色階不均勻的分佈導致影像有些微雜訊。相反地，由於 M-BND 的每個子影像都是藉由 DBS 處理，因此 M-BND 的每個輸出色階都有著完美的藍雜訊特性。然而，當 M-BND 的輸出色階越多，就會需要越多的時間考量各個輸出色階的點分佈。由於 4.4 節所述之 M-BND 的堆疊規則 (Stacking constraint) 與篩選法[57]的堆疊規則十分相似，因此可以藉由篩選法替代 M-BND 的堆疊規則並降低其運算量。篩選法不僅能透過臨界值矩陣(Thresholding matrix)簡化演算法，還能夠藉由挑選好的迭代式半色調技術產生臨界值矩陣，藉此改善在極端灰階時沒有點輸出的問題。臨界值矩陣的產生方式如下：

1. 產生一大小為 $M \times N$ 的連續色調影像 $X_g(m, n)$ ，並將所有像素值設為 $\frac{g}{L-1}$ ，其中灰階 $g \in [0, L - 1]$ ，一般灰階影像的每個像素由 8 bits 組成，因此 $L = 256$ 。此外，對應 $X_{initial}(m, n)$ 的半色調 $H_{initial}(m, n)$ 初始用隨機的半色調影像。
2. 選擇迭代式半色調技術，將純色的影像 $X_g(m, n)$ 轉換成對應的半色調影像 $H_g(m, n)$ ，轉換時須根據堆疊規則。(這部分更詳細的流程會在最後說明)
3. 當前一灰階半色調影像 $H_{g-1}(m, n) = 0$ 且目前灰階的半色調影像 $H_g(m, n) = 1$ ，則臨界值矩陣 $T(m, n)$ 紀錄灰階值 g ，其公式可以定義如下

$$T(m, n) = L - \sum_{g=0}^{L-1} H_g(m, n), \quad (6.3)$$

最後，依照下方公式就可藉由臨界值矩陣快速地產生半色調影像

$$H(u, v) = \begin{cases} 1, & \text{if } X(u, v) \geq T(u \bmod M, v \bmod N), \\ 0, & \text{otherwise} \end{cases}, \quad (6.4)$$

其中， $X(u, v)$ 表示大小為 $U \times V$ 的原始影像在座標位置 (u, v) 。當矩陣比原始影像小時，可藉由映射(Mapping)的方式處理，就可得到半色調影像 $H(u, v)$ 。由於臨界值矩陣為離線(Off-line)訓練產生的，當原始影像大小為 $U \times V$ ，則篩選法的計算複雜度為 $O(UV)$ ，比起過往迭代式運算更具實際應用的競爭力。

雖然大多的迭代式(Iterative)半色調/多色調技術能夠產生不錯的半色調/多色調影像，但極端灰階時通常沒辦法有點的輸出，因為模擬人眼視覺系統的模型其範圍不夠大到足以捕捉點的稀疏性，點的稀疏性正比於理想的點距離 $\lambda_{\bar{g}}$ 且當灰階趨近極端時會急遽的成長。在半色調中，符合藍雜訊特性的理想點距離 $\lambda_{\bar{g}}$ 如下定義：

$$\lambda_{\bar{g}} = \begin{cases} 1/\sqrt{\bar{g}}, & \text{if } g \in [0, 1/4] \\ 2, & \text{if } g \in [1/4, 3/4], \\ 1/\sqrt{1-\bar{g}}, & \text{if } g \in [3/4, 1] \end{cases} \quad (6.5)$$

其中， $\bar{g} = \frac{g}{L-1}$, $g \in [0, L-1]$, g 表示可能的灰階值。由於上述問題，在灰階範圍 $1 \leq g \leq 5$ 及 $250 \leq g \leq 254$ 時，三色階多色調迭代式技術 M-DBS 與 M-BND 都沒有點的輸出。雖然藉由使用較多中間色階的墨水濃度能夠改善這個問題，但會導致有不自然的紋理產生。半色調技術 DBS 也會發生相同的問題，因此臨界值矩陣不能使用 DBS 來產生。根據過往文獻[58]，可以使用 2.5 節的雙指標直接二元搜尋法(Dual metric DBS, DMDBS) [16]來除去以上問題。將公式(2.36)定義的自相關函數(Auto-correlation) $c_{p_i p_i}$ 加入一參數控制其尺度參數，公式如下：

$$c_{p_i p_i}^{\gamma}[m, n] = \frac{180^2}{(\pi D)^2} c_{h_i h_i} \left(\frac{180m}{\pi \gamma S}, \frac{180n}{\pi \gamma S} \right), \quad (6.6)$$

$$\text{where } c_{h_i h_i}(u, v) = k_1 \exp \left(-\frac{u^2 + v^2}{2\sigma_1^2} \right) + k_2 \exp \left(-\frac{u^2 + v^2}{2\sigma_2^2} \right), \quad (6.7)$$

在此， γ 用來控制尺度參數的大小，其餘參數定義如同公式(2.36)及(2.37)。DMDBS 在不同大小的尺度參數下會有不同的情況，在較小的尺度參數時可以產生較平滑的分佈，但會造成極端灰階沒有輸出；在較大的尺度參數時雖然會產生一些隨機的不自然紋理，但在極端灰階時能夠有穩定的輸出。藉由 γ 調整尺度參數，依照文獻[58]所討論，在灰階 $4 \leq g \leq 251$ 時 γ 設為 1，其餘灰階則 γ 設為 2，最後就能得到在各種灰階情況下都有完美點分佈且無帶狀效應(Banding effect)問題的半色調結果。

值得注意的是，一般的迭代式在對影像優化時，會對像素點執行翻點(Toggle)及交換(Swap)的動作。然而，若要遵守堆疊規則的話，這兩動作需要視情況判斷是否執行，DMDBS 遵守堆疊規則的簡易編碼(Pseudo code)修改如下

Pseudo code - DMDBS abide by stacking constrain.

初始設定如同前面產生臨界值矩陣時的步驟 1

產生自相關函數 $c_{\tilde{p}\tilde{p}}^Y$ 及誤差函數 $c_{\tilde{p}\tilde{e}}$

For 灰階 g 從 0 到 $L - 1$

If 在前一灰階產生的半色調 H_{g-1} 中，所有像素都為 0，或者 $g = 0$

產生隨機的半色調影像做為當前半色調 H_g 的初始影像

Else

用前一灰階產生的半色調 H_{g-1} 做為當前半色調 H_g 的初始影像

While 當前半色調影像 H_g 中，仍有像素能藉由翻點或交換產生更好的結果

For 當前半色調像素位置 $H_g(m, n)$ ，從 $(0, 0)$ 執行到 $(M - 1, N - 1)$

If H_g 是繼承前一灰階產生的半色調 H_{g-1} 且 $H_g(m, n) = 1$

跳過當前像素，執行下一個像素

假設當前像素 $H_g(m, n)$ 翻點，計算並保存其對應的 Δe

For $H_g(m, n)$ 與周圍像素交換，從 $(m - 1, n - 1)$ 執行到 $(m + 1, n + 1)$

If H_g 是繼承前一灰階產生的半色調 H_{g-1} 且交換的點輸出為 1

跳過當前像素，執行下一個像素

Else

假設當前像素 $H_g(m, n)$ 與周圍像素交換，計算並保存其 Δe

從以上的 Δe 中挑選出最小的 Δe_{min} 並記錄其對應的動作

If $\Delta e_{min} < 0$

藉由 Δe_{min} 及其對應的動作更新當前誤差函數 $c_{\tilde{p}\tilde{e}}$ 及半色調像 H_g

Else

跳過當前像素，執行下一個像素

End

6.3 實驗結果比較

在此章節，四個多色調技術將被用來與改良技術比較：1) 多色調直接二元搜尋法(M-DBS) [38]，2) 多色調藍雜訊抖動法(M-BND) [39]，3) 直接多位元搜尋法(DMS) [40]，4) 色調替代之誤差擴散法(M-TRED) [42]。改良的多色調高效率藍雜訊抖動法(M-EBND)與各技術將會從不同的層面去比較，包括點分佈、不自然紋理問題(Artifact problem)、處理效率等。比較時將會以平均的數據去分析，當中使用資料庫 MSPLAB [54]，其包含 1,000 張自然的灰階影像，影像大小為 512×512 。

所有技術將會設定在其最佳情況下，詳細的設定如下：

- 1) 多色調直接二元搜尋法(M-DBS) [38]：點擴展函數(Point spread function, PSF)使用大小為 7×7 標準圓形的對稱高斯濾波器，其標準差設為 1。
- 2) 多色調藍雜訊抖動法(M-BND) [39]：在 q 色階時，使用從 q 色階多色調直接二元搜尋法依照上述設定產生的結果中統計的墨水濃度曲線。
- 3) 直接多位元搜尋法(DMS) [40]：其自相關函數中的參數 $(k_1, k_2, \sigma_1, \sigma_2)$ 依照該論文設定為 $(43.2, 38.7, 0.02, 0.06)$ 。
- 4) 色調替代之誤差擴散法(M-TRED) [42]：機率參數 p 依照該論文設定為 0.65。
- 5) 本研究提出的多色調高效率藍雜訊抖動法(M-EBND)：在 q 色階時，使用從 q 色階色調替代之誤差擴散法依照上述設定產生的結果中統計的墨水濃度曲線。篩選法要用到的臨界值矩陣，使用改良的 DMDBS 遵守堆疊規則產生。

6.3.1 平均功率頻譜密度(Average power spectrum density)

表 6.1 展示各個多色調技術在特定灰階下，三色階的結果及其對應的改良的平均功率頻譜密度圖(Averaged power spectrum density, APSD) [58]，APSD 的功能是用來觀察能量的分佈。當中要注意的是，M-DBS、M-BND 及 DMS 在極端灰階時沒有輸出點(在表中標記為”n/a”)。觀察表 6.1 中的 APSD 可以發現，M-DBS 及 M-BND 的能量集中且穩定，這表示它們的多色調影像能夠避免掉大部分不自然或不連續的紋理。相較之下，雖然 DMS 在灰階 $g \leq 16$ 能量都有聚集在中央，但當 $g \geq 32$ 時，DMS 的能量卻能夠進一步地分成兩個能量，一個能量為黑色較小的圓圈，另一個是灰色較大的圓圈。這個現象產生的原因是因為 DMS 的量化影像是由兩個差距很大的頻率所構成，所以兩個 APSD 所呈現的圓圈的大小差很多，這表示其多色調影像容易產生帶狀效應(Banding effect)及雜訊問題(Noising effect)。如表 6.1(d)所展示，M-TRED 的能量在每個灰階下雖然聚集但不是均勻的圓形，這代表著它的多色調影像中相同數值的像素點之間的距離是不一致的，這不均勻的問題容易導致多色調影像產生許多不自然的紋理。與上述技術相比，本論文提出的改良技術有著不錯的功率頻譜密度，其能量聚集且穩定呈現圓形，結果僅略差於 M-DBS 及 M-BND。雖然在灰階 $g = 64$ 時有類似 DMS 兩個能量的現象，但由於兩個圓圈差異不大(表示兩個構成影像的頻率相似)，因此改良的 M-EBND 產生的結果僅受到一些雜訊影響。這邊是一個權衡問題，當點的分佈較分散時，多色調影像看起來才夠連續，但較散的分佈看起來會像是雜訊；若點的分佈較密集，則多色調影像會像圖 4.15(b)及(d)那樣因為灰階劇烈的變化而看似不連續。

為了更詳細地觀察不同輸出色階的能量分佈，將 q 色階的多色調影像藉由轉換公式轉換成 $q - 1$ 張半色調子影像，轉換公式如下：

$$D_i(m, n) = \begin{cases} 1, & M(m, n) \geq g_{i+1}, \text{ where } i = 1, \dots, q - 1, \\ 0, & \text{otherwise} \end{cases} \quad (6.8)$$

其中， D_i 表示從多色調影像 M 轉換後的半色調子影像； g_i 為第 i 個量化色階的值。各個三色階多色調技術在特定灰階下，其轉換後的半色調子影像及對應的 APSD 如表 6.2 所示。在表 6.2 中，三色階情況下的半色調子影像分別顯示的是 D_1 (以黑色為基底灰點

與白點的分佈)及 D_2 (以黑色為基底白點的分佈)。雖然 M-DBS 的能量在表 6.1 中非常的穩定，但在表 6.2(a)中只有 D_1 勉強算是均勻的分佈， D_2 在灰階 $g \leq 64$ 時明顯地分佈不均勻。此外，根據表 6.2 所顯示，DMS 及 M-TRED 的子影像 D_1 其 APSD 只有在灰階 $g \leq 32$ 時能量足夠聚集，子影像 D_1 其它灰階及子影像 D_2 所有灰階的 APSD 都不夠穩定，這表示 D_1 及 D_2 的點分佈幾乎都是不均勻的。相較之下，M-BND 及改良的技術 M-EBND 在各個灰階情況下產生的結果其 APSD 都是聚集且穩定的。從表 6.1 及表 6.2 各技術結果影像及對應的 APSD 可以發現，表 6.2 比表 6.1 能夠提供更多可靠的訊息，因為從表 6.2 中可得知各個輸出色階點分佈的情況。

表 6.1 128×128 大小的影像在特定灰階下，各個多色調技術三色階量化影像及對應的 APSD 影像，其中”n/a”表示沒有任何點輸出

	$g = 1$	$g = 2$	$g = 4$	$g = 8$	$g = 16$	$g = 32$	$g = 64$	$g = 128$
(a) M-DBS [38]	n/a	n/a	n/a					
(b) M-BND [39]	n/a	n/a	n/a					
(c) DMS [40]	n/a							
(d) M-TRED [42]								
(e) Proposed method								

表 6.2 128×128 大小的影像在特定灰階下，各個多色調技術三色階量化影像其分層半色調影像及對應的 APSD 影像，其中”n/a”表示沒有任何點輸出

	$g = 32$		$g = 64$		$g = 96$		$g = 128$	
	D_1	D_2	D_1	D_2	D_1	D_2	D_1	D_2
(a) M-DBS [38]								
(b) M-BND [39]	n/a							
(c) DMS [40]	n/a							
(d) M-TRED [42]	n/a							
(e) Proposed method	n/a							

6.3.2 多色調紋理(Multitone textures)

在這個章節中將會探討各個技術量化影像的視覺品質。圖 6.5 所示為 512×96 的 Ramp 影像經過各個三色階的多色調技術轉換後的量化影像。從圖中可以發現，如同 6.1 節的墨水濃度(Iink concentration)所討論，M-DBS 及 M-BND 的量化影像在灰階 $g = 40$ 及 $g = 215$ 左右時比起其他技術具有更嚴重的高對比問題(High contrast issue)。此外，M-DBS 由於高對比及不均勻分佈等問題，導致其結果輕易的受到雜訊(Noising effect)影響。雖然 DMS 及 M-TRED 使用更多的中間色調減輕了高對比的問題，但不均勻的點分佈造成 DMS 在中間灰階時有帶狀效應(Banding effect)，而 M-TRED 則是繼承了 ED 的特性，使其結果有明顯的線性圖案(Line pattern)。相較之下，改良的 M-EBND 保持 M-BND 均勻分佈的特性，並且藉由改善的墨水濃度曲線避免高對比情況發生，因此 M-EBND 有著平滑柔順的結果。

為了分析各個多色調技術的優劣，圖 6.6 呈現的是 512×512 大小的自然影像經過各技術不同色階量化後的多色調影像。如同 6.3.1 小節平均功率頻譜密度(Average power spectrum density)所述，M-DBS、DMS 及 M-TRED 不均勻的分佈，因此造成如圖 6.6(a)、(c) 及 (d) 的不自然結果。十分明顯的是，M-BND 及改良的 M-EBND 擁有最好的影像視覺品質，這證明了均勻的分佈能夠改善多色調影像的品質。為了更詳細地觀察各多色調技術完整的輸出影像，圖 6.7 及圖 6.8 分別以不同的多色調技術將圖 4.1 的測試影像 3、4 轉換為多色調影像的結果，其中包含了三色階至六色階的多色調影像。上述各技術的問題都能在圖中明顯的察覺，儘管色階越多時問題會越不明顯，但將會造成硬體設備上的負擔。

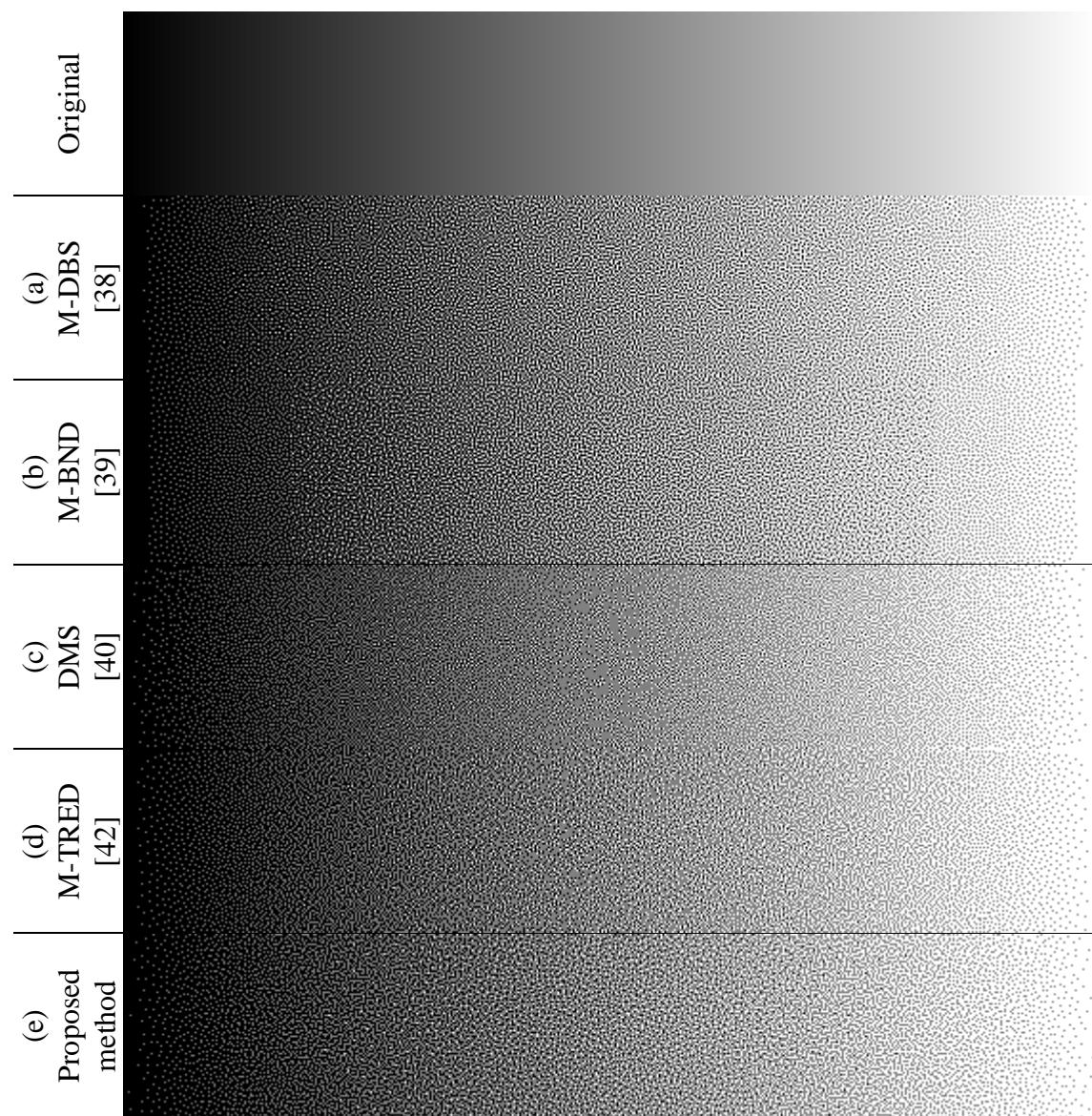


圖 6.5 大小為 512×96 的 Ramp 影像經各個三色階的多色調技術轉換後的量化影像

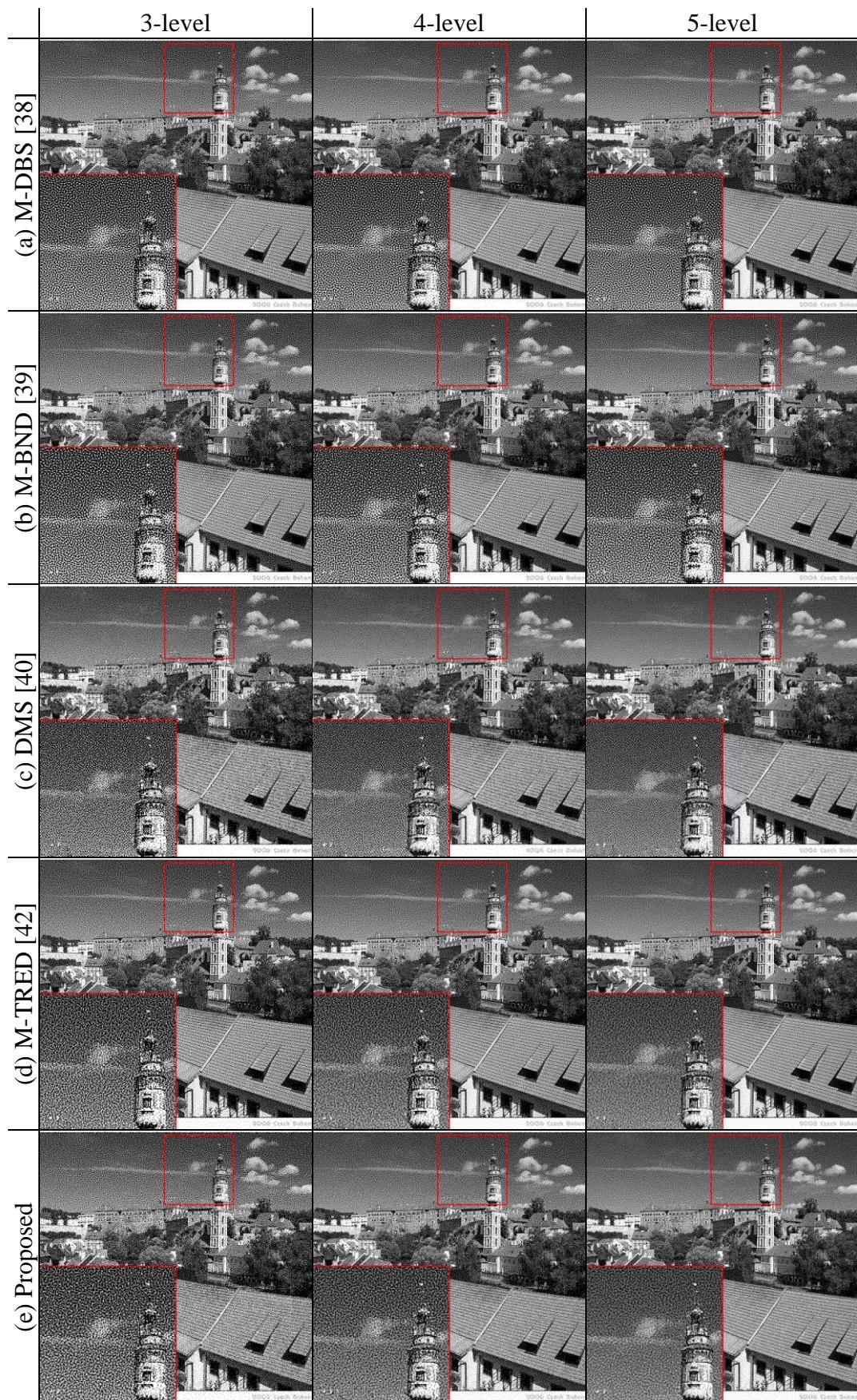


圖 6.6 大小為 512×512 的自然影像經各個多色調技術量化後的結果，從左至右的量化色階數分別為三、四、五色階。



(a.1) 三色階



(a.2) 四色階



(a.3) 五色階



(a.4) 六色階

(a) 多色調直接二元搜尋法(M-DBS) [38]



(b.1) 三色階



(b.2) 四色階



(b.3) 五色階



(b.4) 六色階

(b) 多色調藍雜訊抖動法(M-BND) [39]



(c.1) 三色階



(c.2) 四色階

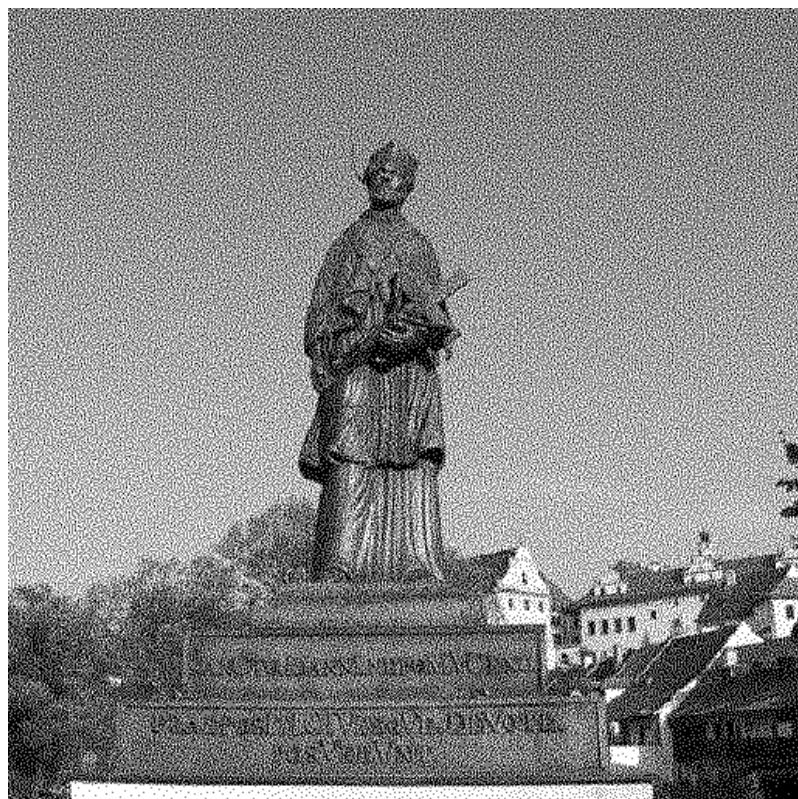


(c.3) 五色階



(c.4) 六色階

(c) 直接多位元搜尋法(DMS) [40]



(d.1) 三色階



(d.2) 四色階



(d.3) 五色階



(d.4) 六色階

(d) 色調替代之誤差擴散法(M-TRED) [42]



(e.1) 三色階



(e.2) 四色階



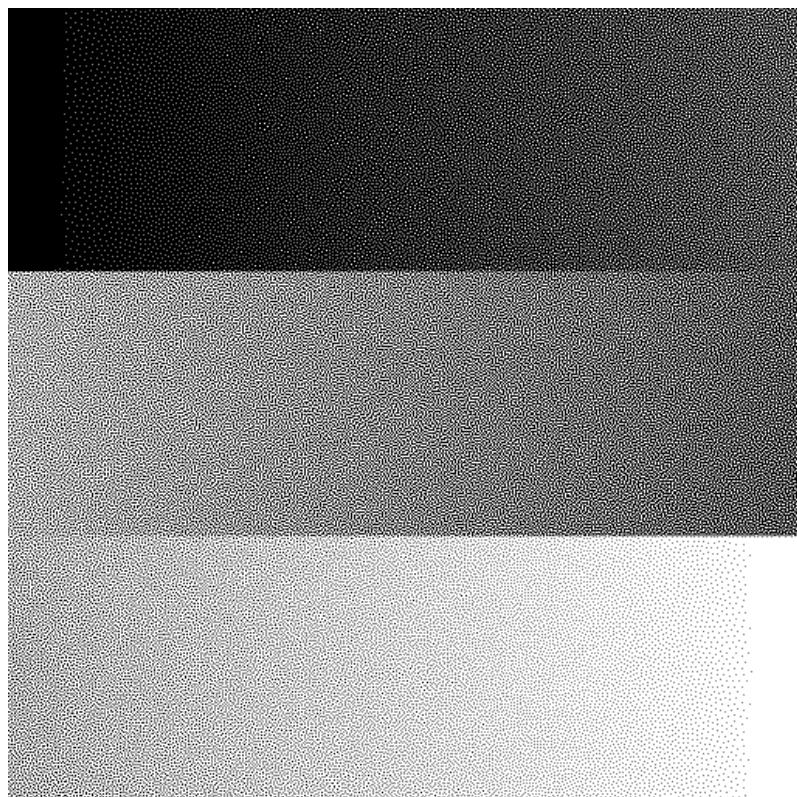
(e.3) 五色階



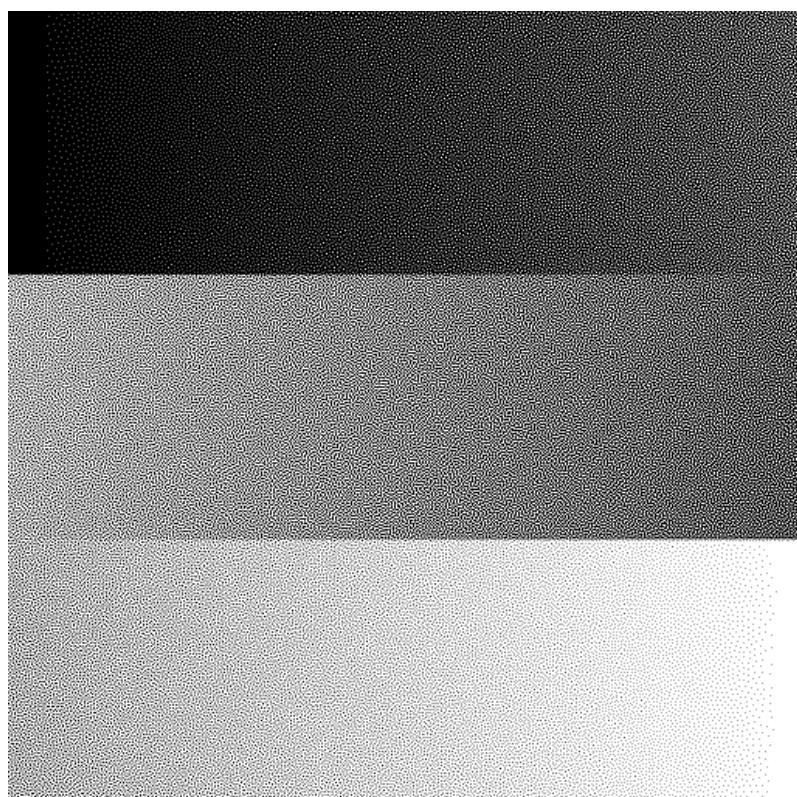
(e.4) 六色階

(e) 改良的多色調高效率藍雜訊抖動法

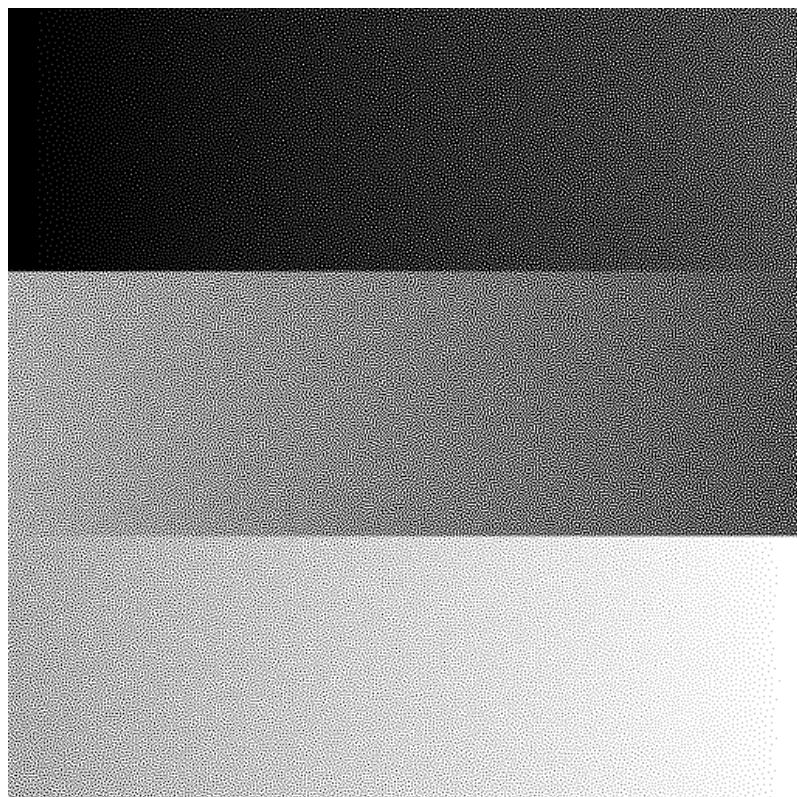
圖 6.7 不同的多色調技術產生的測試影像-灰階影像 3 之多色調影像



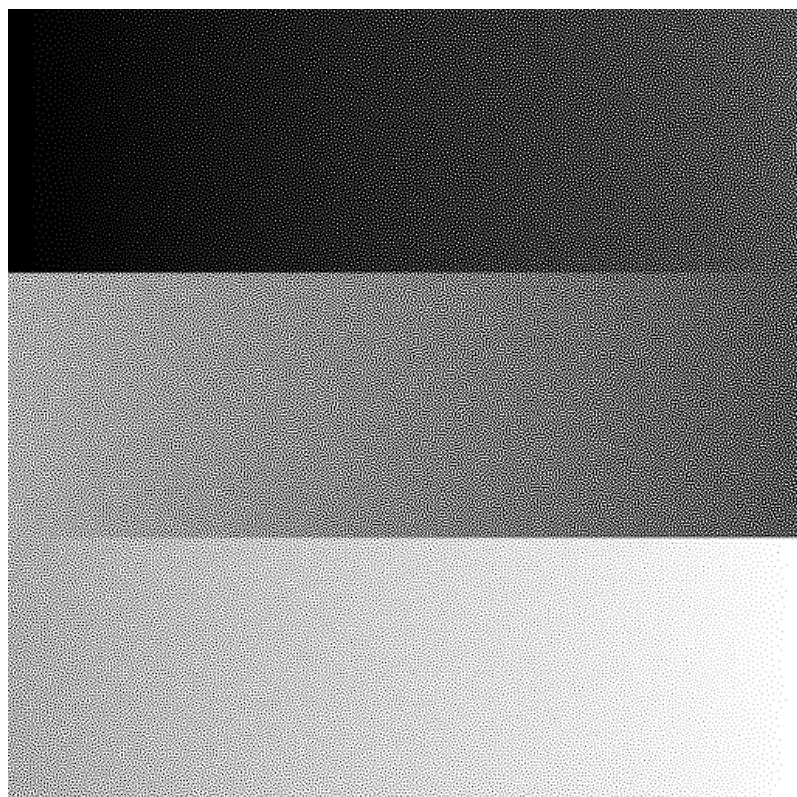
(a.1) 三色階



(a.2) 四色階

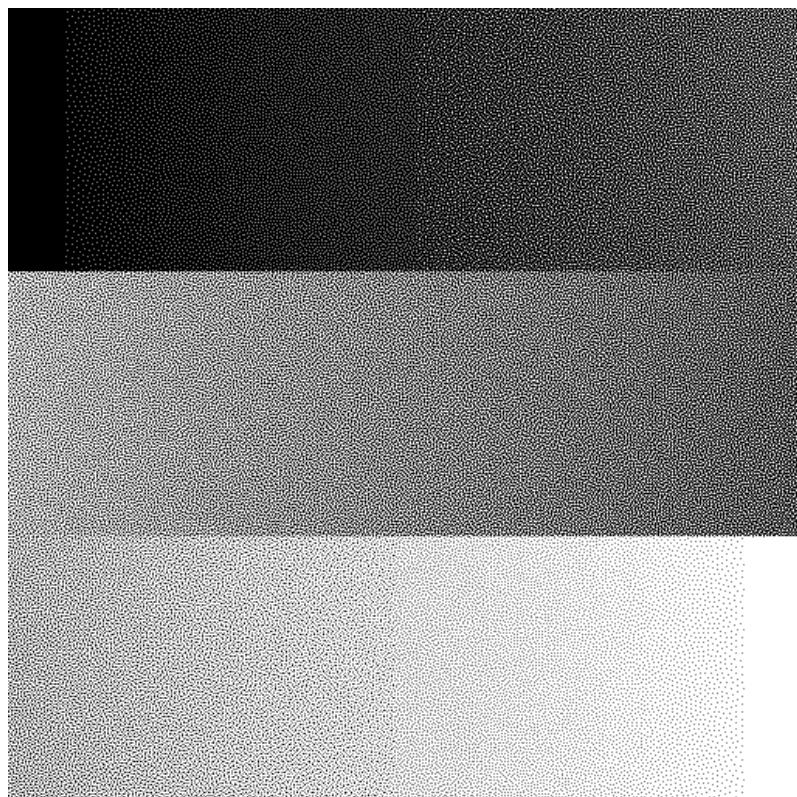


(a.3) 五色階

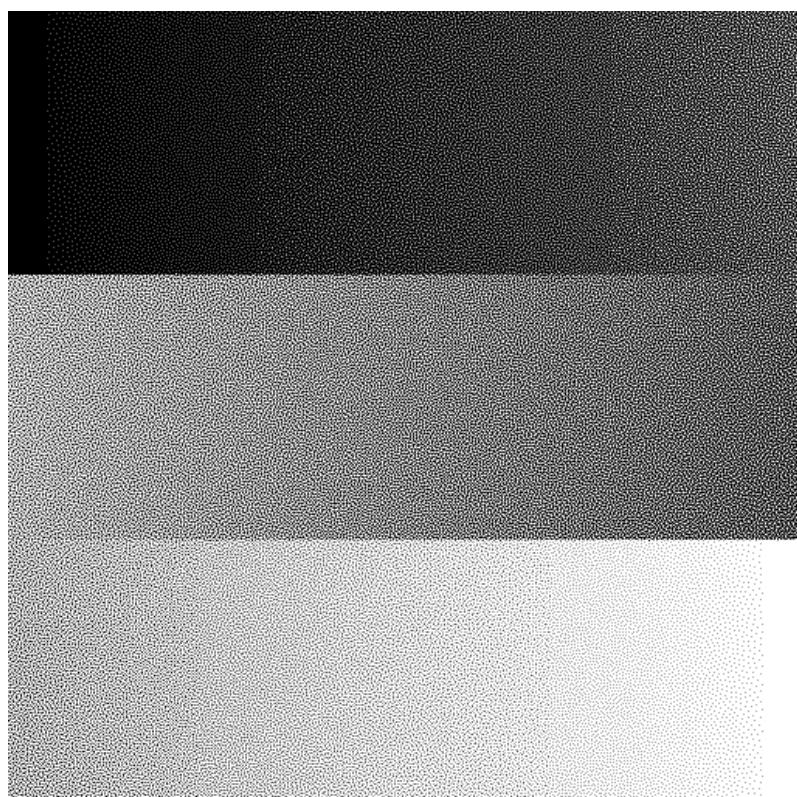


(a.4) 六色階

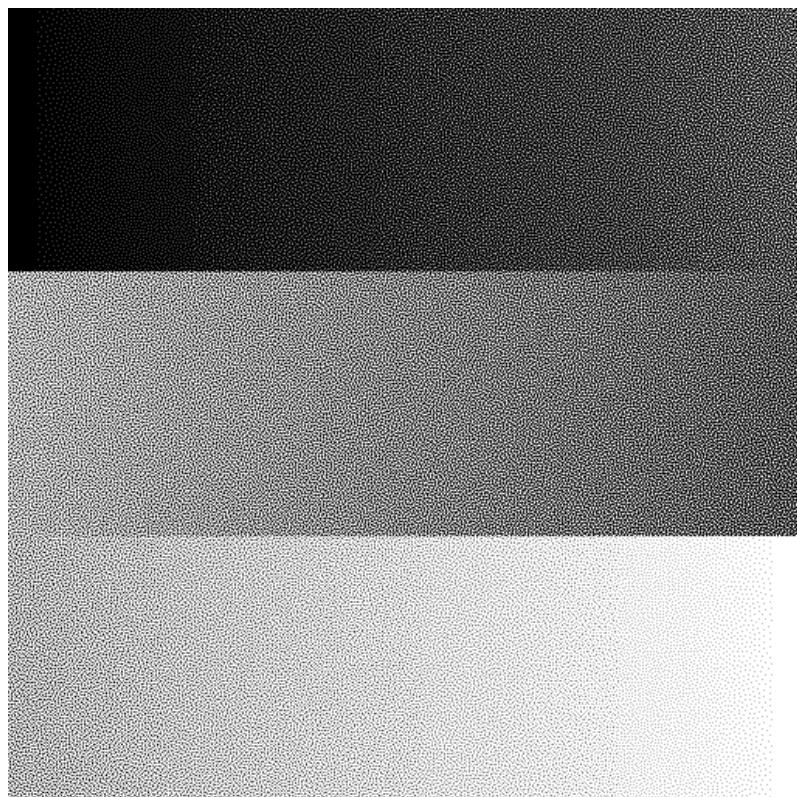
(a) 多色調直接二元搜尋法(M-DBS) [38]



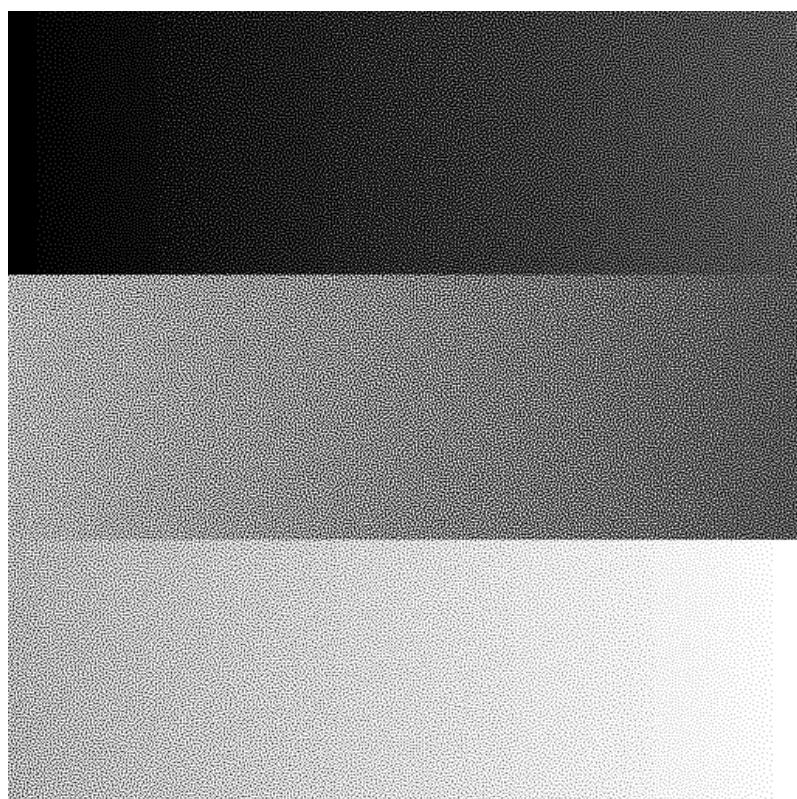
(b.1) 三色階



(b.2) 四色階

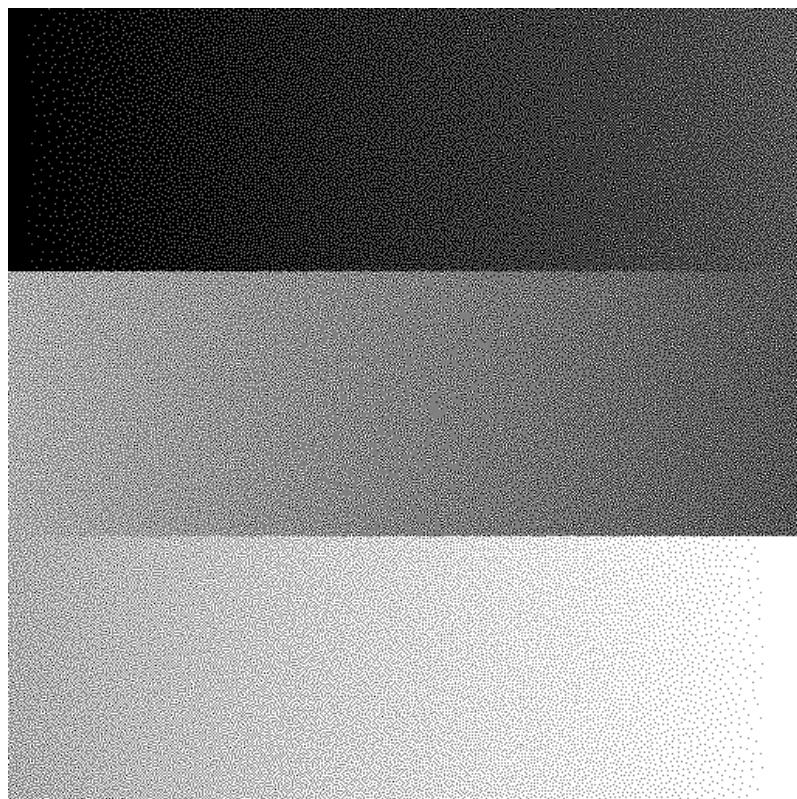


(b.3) 五色階

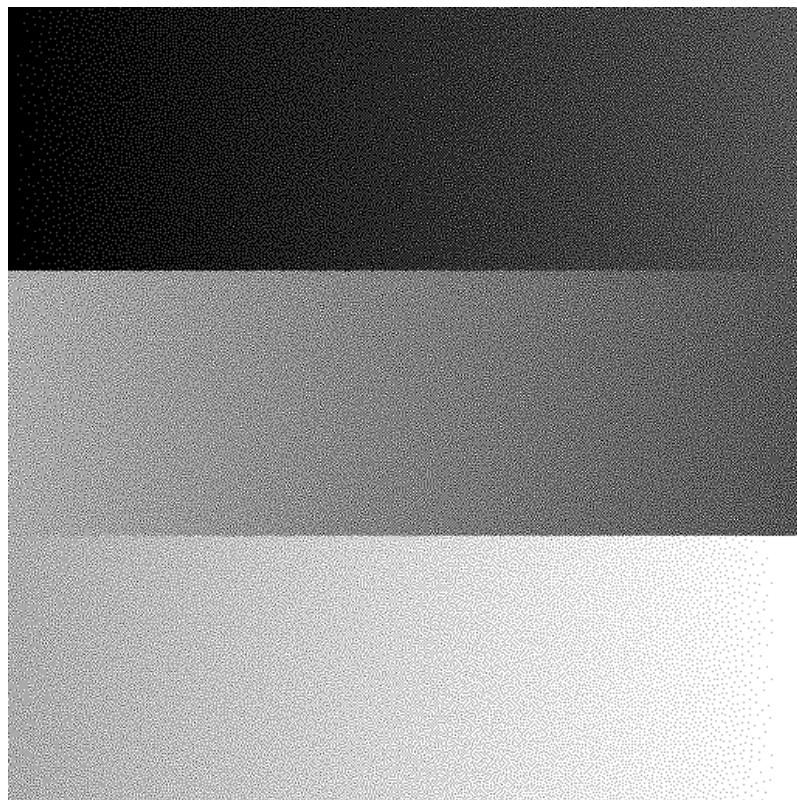


(b.4) 六色階

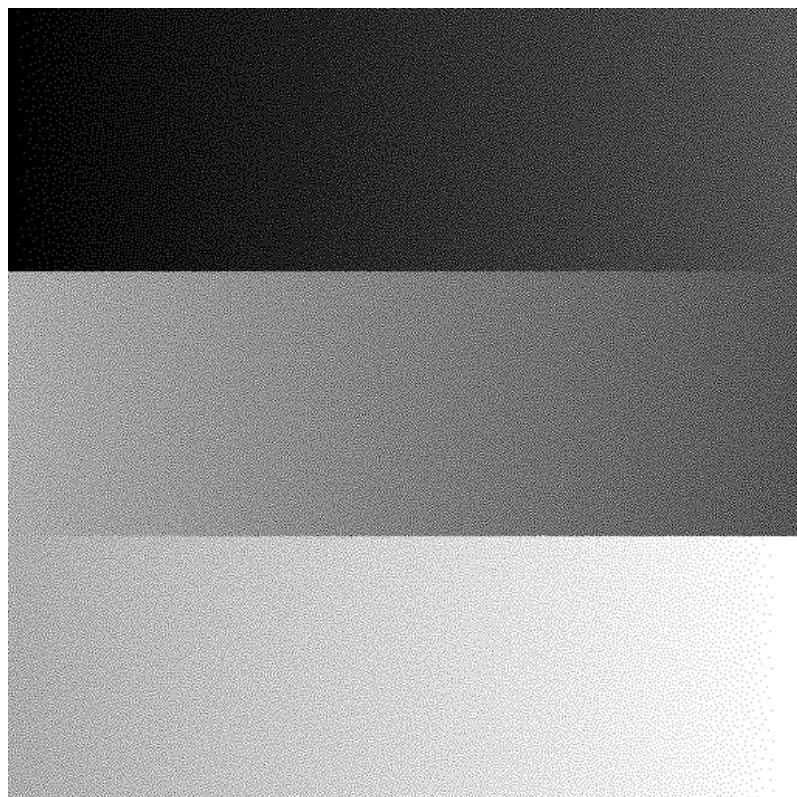
(b) 多色調藍雜訊抖動法(M-BND) [39]



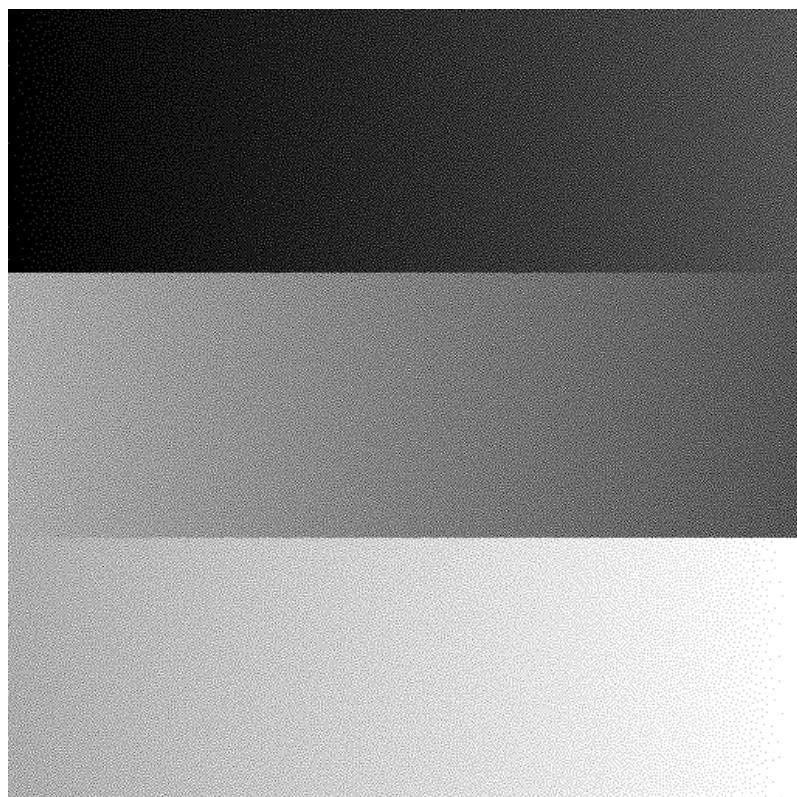
(c.1) 三色階



(c.2) 四色階

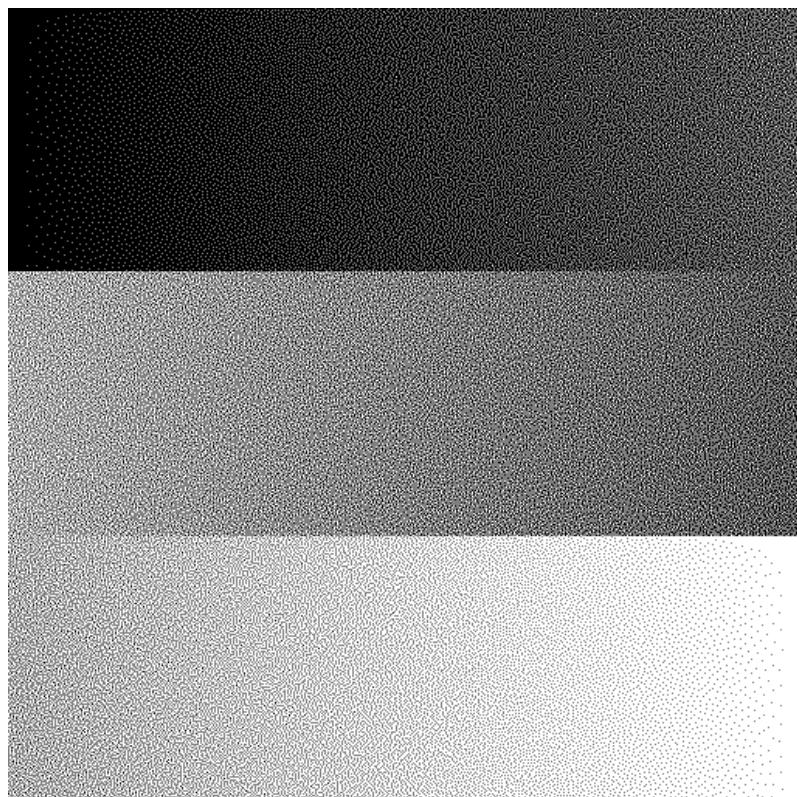


(c.3) 五色階

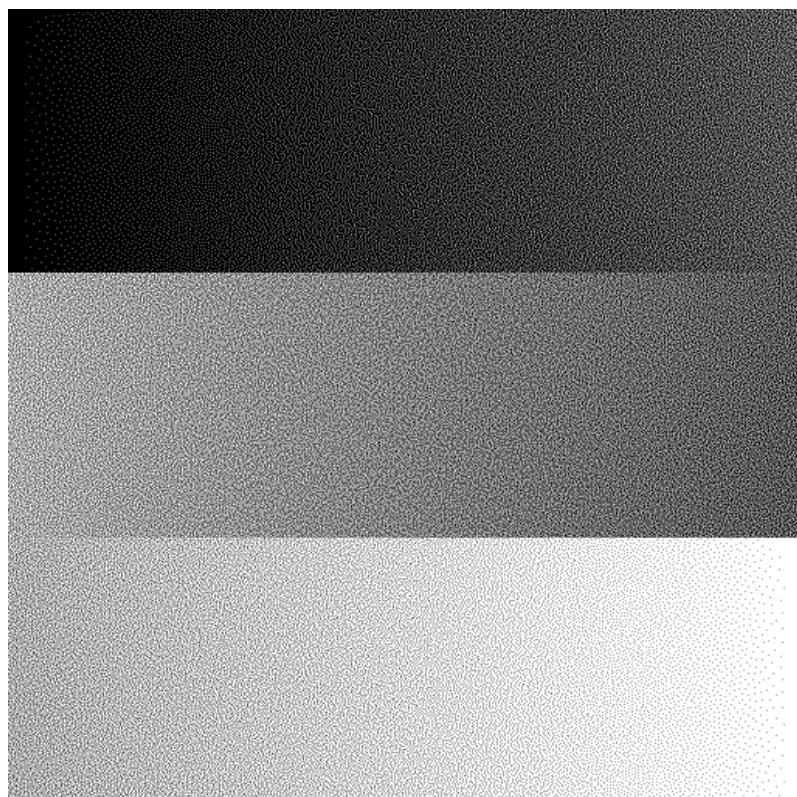


(c.4) 六色階

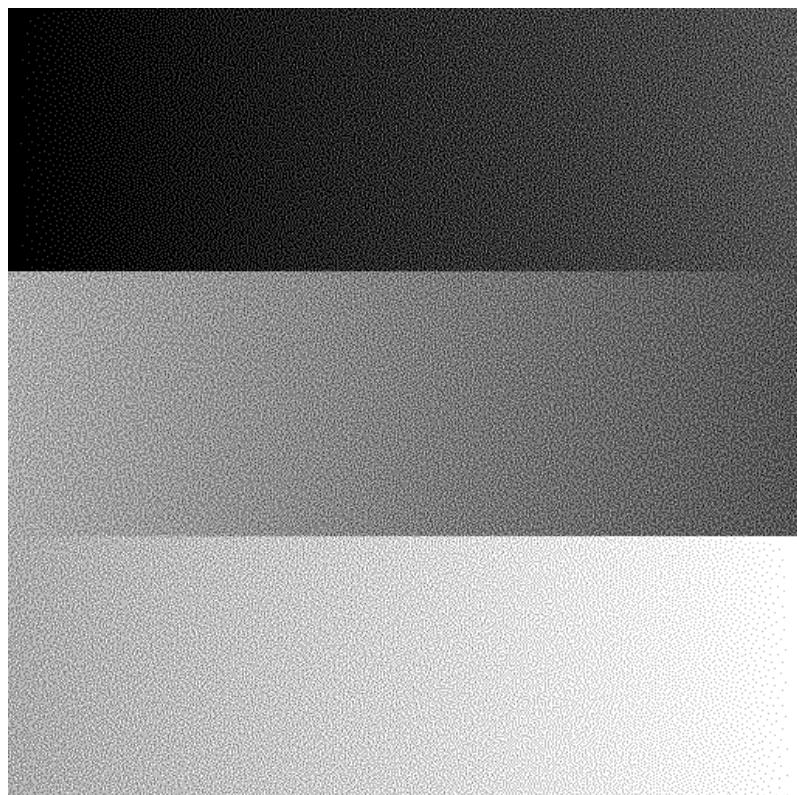
(c) 直接多位元搜尋法(DMS) [40]



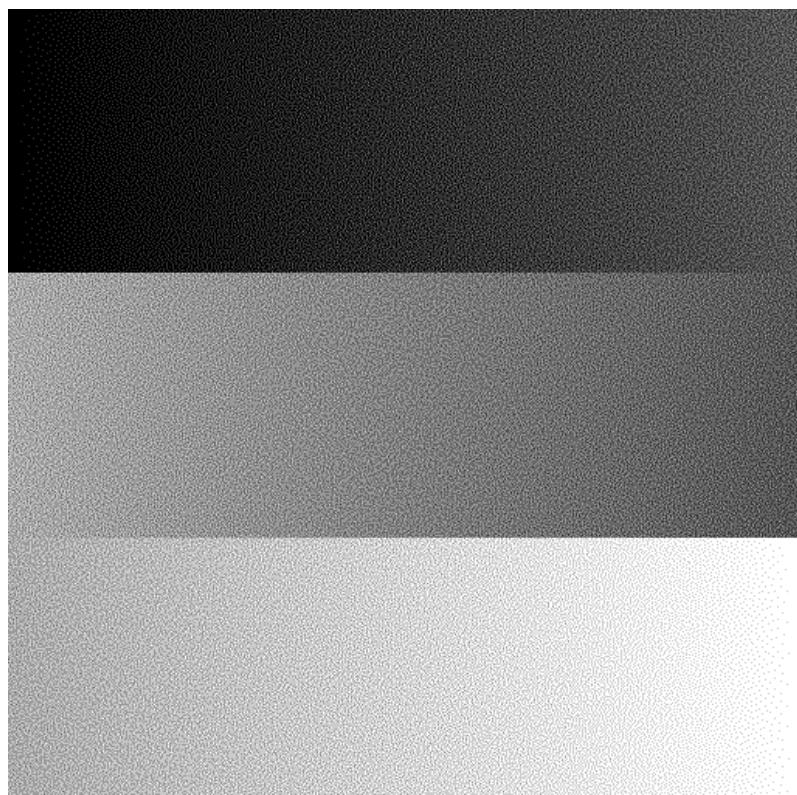
(d.1) 三色階



(d.2) 四色階

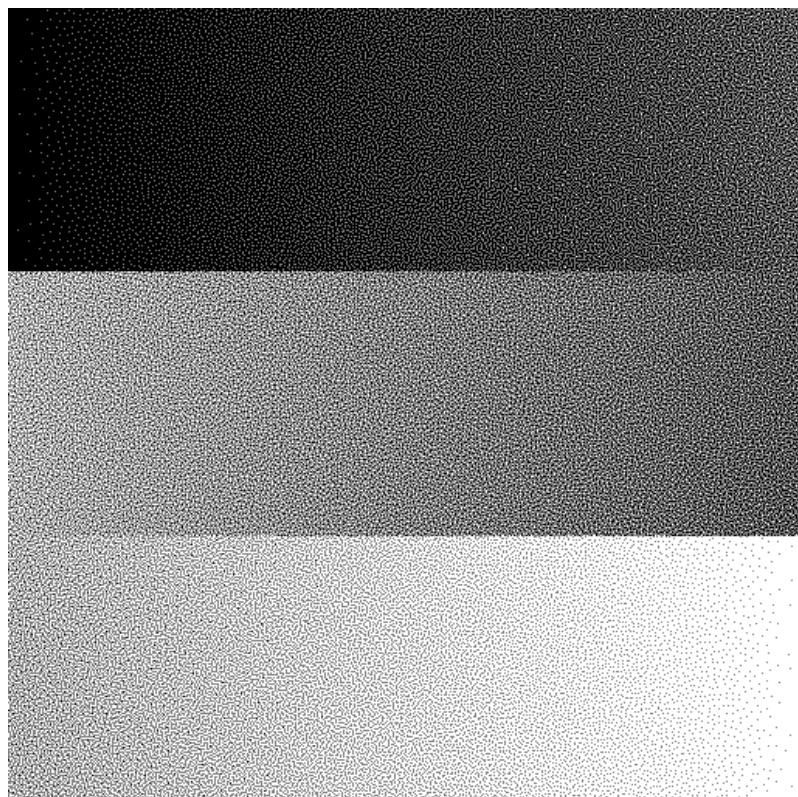


(d.3) 五色階

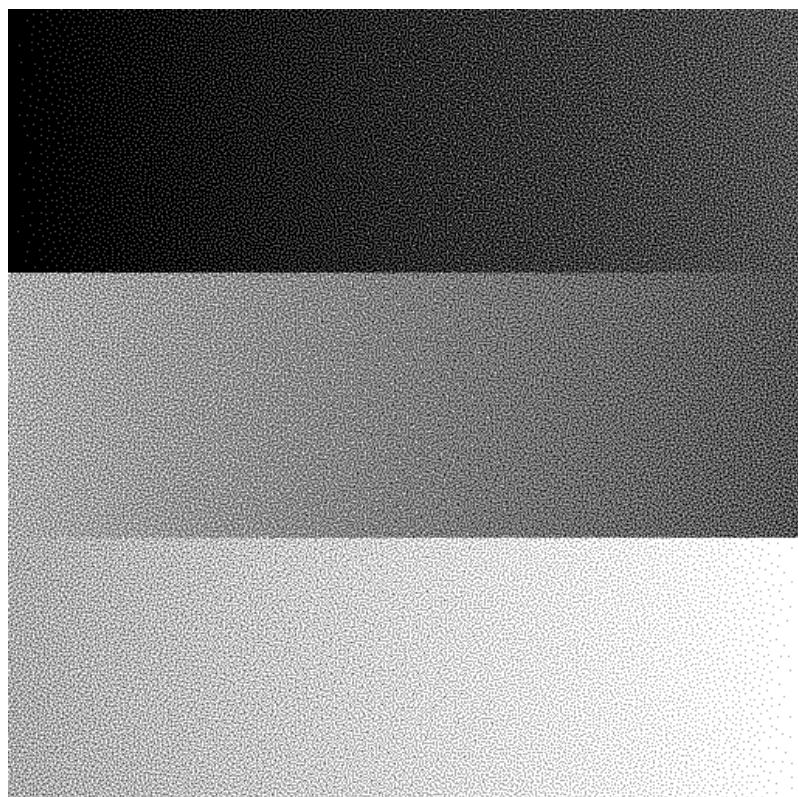


(d.4) 六色階

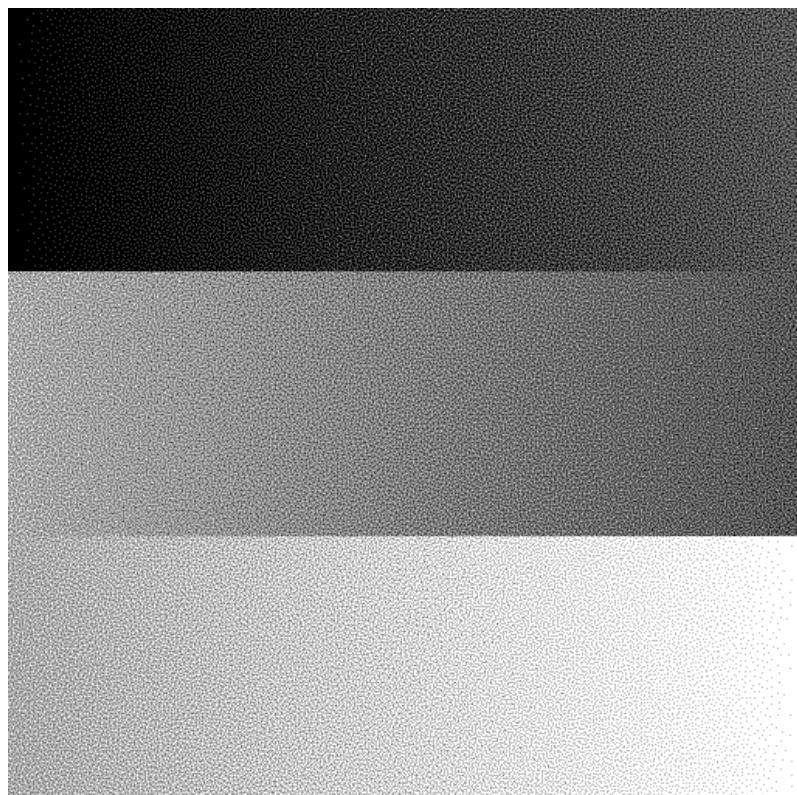
(d) 色調替代之誤差擴散法(M-TRED) [42]



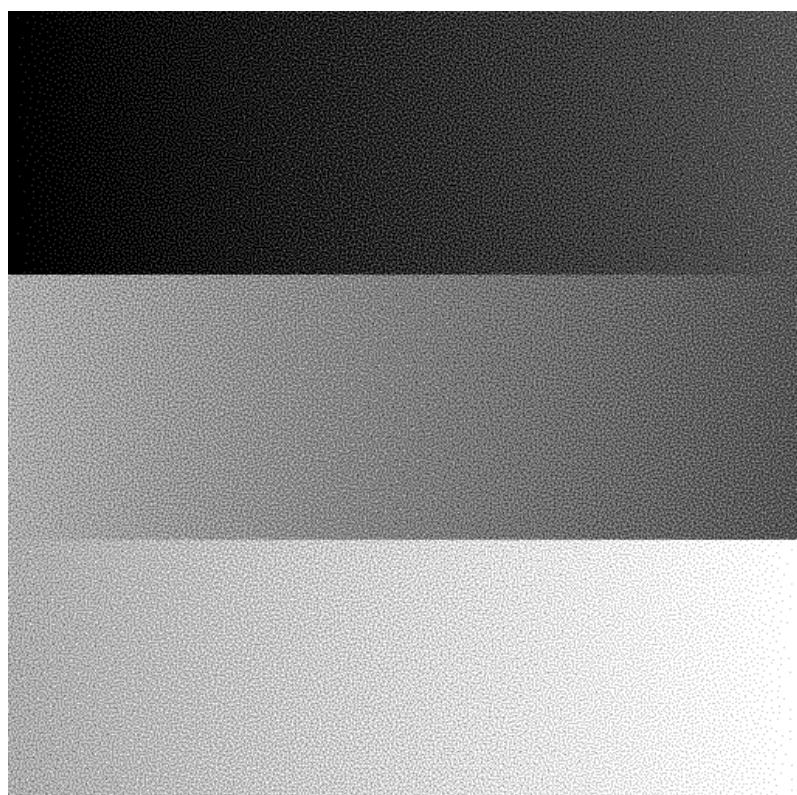
(e.1) 三色階



(e.2) 四色階



(e.3) 五色階



(e.4) 六色階

(e) 改良的多色調高效率藍雜訊抖動法

圖 6.8 不同的多色調技術產生的測試影像-灰階影像 4 之多色調影像

6.3.3 總結比較

圖 6.9 所示為不同多色調技術在不同色階數時的平均處理速度，使用資料庫 MSPLAB [54]所有的影像。在此，所有的多色調技術都以 Visual Studio 2012 C++在硬體設備 Intel Core i7-4790 3.60GHz CPU 及 8GB RAM 上實現。顯而易見的，在圖 6.9 中 M-DBS 及 M-BND 的高計算複雜度使其處理速度遠高於其他多色調技術約 100 倍，因此它們明顯的沒辦法實際地應用在印刷領域。相比之下，鑑於 DMS、M-TRED 及改良技術的計算複雜度，它們更具處理速度上的優勢。然而，因為 M-TRED 是基於半色調 ED 技術的延伸，因此其像素的處理需要將誤差擴散至周圍，這表示每個像素都需要考量周圍像素，使得其處理速度高於 DMS 及改良技術 2 倍多。雖然 DMS 與改良技術都是使用篩選法簡化後的多色調技術，因此有著相似的處理時間，但 DMS 在 q 色階時會需要 $q - 1$ 個臨界值矩陣，而改良的 M-EBND 不論在多少色階的情況下都只需要一個臨界值矩陣，M-EBND 比起 DMS 使用更少的記憶體空間。各技術在不同量化色階時的表現統計在表 6.3 中，各類的最佳解將用圓圈標示。表 6.3 中的處理時間如圖 6.9，不自然紋理問題(Artifact problem)的比較在最後四欄。綜合各個不同層面的比較，改良的 M-EBND 比起其他過往技術更具優勢。

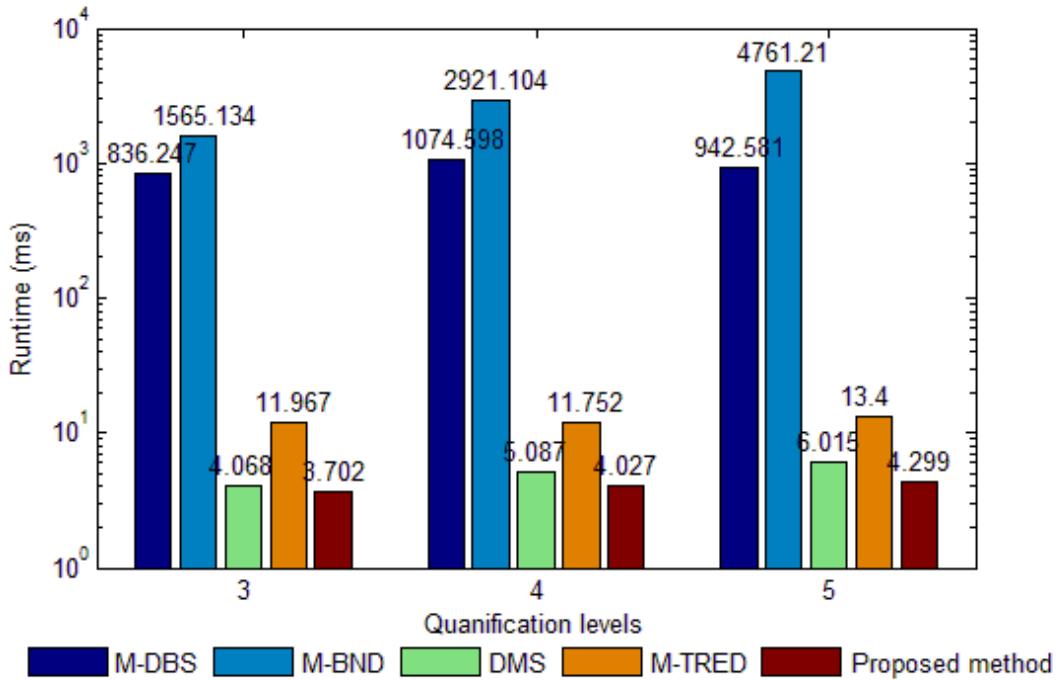


圖 6.9 不同多色調技術在三、四、五色階時的平均處理速度

表 6.3 總結各個多色調技術在不同層面比較下的結果(各類的最佳解將用圓圈標示)

Methods	Level	Coverage of tone range	Runtime (ms)	Uniform distribution	High contrast issue	Banding effect	Line pattern	Noising effect
M-DBS [38]	3	$6 \leq g \leq 249$	836.247	Only D_1	Yes	Slight	No	Slight
	4	$4 \leq g \leq 251$	1074.598	Only D_1	Yes	Slight	No	Slight
	5	$3 \leq g \leq 252$	942.581	Only D_1	Yes	Slight	No	Slight
M-BND [39]	3	$6 \leq g \leq 249$	1565.134	All plane	Yes	Slight	No	No
	4	$4 \leq g \leq 251$	2921.104	All plane	Yes	Slight	No	No
	5	$3 \leq g \leq 252$	4761.21	All plane	Yes	Slight	No	No
DMS [40]	3	$2 \leq g \leq 251$	4.068	No plane	No	Slight	Slight	Slight
	4	$1 \leq g \leq 252$	5.087	No plane	No	Slight	Slight	Slight
	5	$1 \leq g \leq 252$	6.015	No plane	No	Slight	Slight	Slight
M-TRED [42]	3	$1 \leq g \leq 254$	11.967	No plane	No	Slight	Yes	Slight
	4	$1 \leq g \leq 254$	11.752	No plane	No	Slight	Yes	Slight
	5	$1 \leq g \leq 254$	13.4	No plane	No	Slight	Yes	Slight
Proposed method	3	$1 \leq g \leq 254$	3.702	All plane	No	No	No	Slight
	4	$1 \leq g \leq 254$	4.027	All plane	No	No	No	Slight
	5	$1 \leq g \leq 254$	4.299	All plane	No	No	No	Slight

6.4 小結

在本論文中提出一個新的多色調技術，其名為多色調高效率藍雜訊抖動法 (Efficient blue-noise dithering for multitone, M-EBND)。在不同的量化色階中，M-EBND 都能夠產生比其他多色調技術更自然的多色調影像，同時並維持最高的處理效率。本論文提出的技術藉由改善墨水濃度 (Ink concentration) 曲線，消除過往 M-DBS 及 M-BND 易產生高對比問題 (High contrast issue)，並藉由挑選的半色調技術消除極端灰階無法有輸出點而引發的帶狀效應 (Banding effect)。除此之外，改良的 M-EBND 藉由篩選法提供極低的計算複雜度，並保持其各個輸出色階優異的藍雜訊點分佈特性。雖然 DMS 同樣使用篩選法簡化演算法，但改良技術不僅能更快的轉換影像以及使用更少的記憶體空間，還能夠產生比 DMS 更佳的多色調影像。如實驗結果所討論，改良技術幾乎沒有受到任何不自然或不連續紋理影響，在實際的多色調影像中與 M-BND 不分上下。總結來說，在多色調領域中改良技術 M-EBND 比起其他技術更具優勢。

第七章 結論與未來展望

本論文有兩項主要的技術貢獻：1)自適應點擴散區塊截斷編碼(Self-adaptive dot-diffused block truncation coding, SDBTC)。2)多色調高效率藍雜訊抖動法(Efficient blue-noise dithering for multitone, M-EBND)。以上兩個研究皆投稿至國際期刊，詳細投稿資訊將在下個章節中說明。

第一個方法主要是以點擴散區塊截斷編碼技術為基礎架構，針對過往其它區塊截斷編碼技術的缺陷進行改善，如：區域使用固定大小區塊以及區塊量化色階(Quantization level)選擇不佳所導致的脈衝雜訊(Impulse noise)及區塊效應(Blocking effect)等問題。本論文提出區塊區域分解演算法(Block region decomposition algorithm)以及自適應量化色階(Adjustable quantization level)來解決上述問題。除此之外，對於灰階影像壓縮，規律的處理順序容易導致週期性圖案(Periodic pattern)的產生，本技術藉由將類別矩陣(Class matrix)旋轉/對調避免發生相同的問題。而在彩色影像壓縮方面，過往技術為了增加壓縮率大多使用單一位元圖技術(SBM)，使得重建影像遭受到更嚴重的區塊效應影響，在此將Lab通道中a及b通道的影像縮減取樣(Downsampling)就能夠得到與單一位元圖技術相似的壓縮率並能維持其影像的品質。就結論而言，本研究提出的方法在各個壓縮率情況下都有不錯的影像品質，然而，因為需要將每個像素的誤差補償至周圍像素以提升像素之間的相關性，該方法在處理時間(Runtime)方面，比起傳統的區塊截斷編碼(BTC)及絕對矩量區塊截斷編碼(AMBTC)等技術耗時較長。因此如何將程式優化或者以其他方法提升處理效率並保持影像品質，將會是未來研究開發與嘗試的方向。

雖然多色調藍雜訊抖動法(M-BND)優異的多色調結果為其特色，但由於漫長的運算導致它無法應用於實際的市場。因此本論文提出以多色調藍雜訊抖動法為架構，藉由篩選法(Screening)改善處理效率，並改良該技術會用到的墨水濃度曲線(Ink concentration curve)，避免造成高對比問題(High contrast issue)。實驗證明，本方法在處理時間及各個輸出色階的點分佈都有不錯的結果，但由於墨水濃度曲線沒有一個好

的影像評估方法去訓練，使得在本技術中只能以主觀的評估選擇。此外，雖然本技術與色調替代之誤差擴散法(M-TRED)的各個輸出色階有相同數量，而且本研究提出的方法有著均勻的點分佈，但多色調結果卻有較明顯的雜訊問題(Noising effect)。以上缺憾希望在未來研究時能夠解決，以使多色調技術有更長足的發展。

第八章 投稿資料

本論文所提出的自適應點擴散區塊截斷編碼及多色調高效率藍雜訊抖動法已投稿至 IEEE Transcation on Circuits and Systems for Video Technology、IEEE Transcation on Image Processing 及 IEEE International Conference on Image Processing。詳細如下：

可調式點擴散區塊截斷編碼(Adjustable dot-diffused block truncation coding)

- [1] Y. F. Liu, J. M. Guo, and Y. Cheng, “Adjustable dot-diffused block truncation coding,” *IEEE Trans. Circuits and Systems for Video Technology*, Submitted.

自適應點擴散區塊截斷編碼(Self-adaptive dot-diffused block truncation coding, SDBTC)

- [2] Y. F. Liu, J. M. Guo, and Y. Cheng, “Image compression using self-adaptive dot-diffused block truncation coding,” *IEEE Trans. Image Processing*, Submitted.

多色調高效率藍雜訊抖動法(Efficient blue-noise dithering for multitone, M-EBND)

- [3] J. M. Guo and Y. Cheng, “Efficient blue-noise dithering for multitone,” *IEEE Trans. Image Processing*, Submitted.

適應型點擴散區塊截斷編碼(Adaptive dot-diffused block truncation coding)

- [4] Y. F. Liu, J. M. Guo, and Y. Cheng, “Adaptive block truncation coding image compression technique using optimized dot diffusion,” in *Proc. IEEE ICIP*, 2016, Accepted.

參考文獻

- [1] *The USC-SIPI Image Database*. (1981) [Online]. Available:
<http://sipi.usc.edu/database>
- [2] R. W. Floyd and L. Steinberg, “An adaptive algorithm for spatial gray scale,” in *proc. SID 75 Digest. Society for information Display*, pp. 36-37, 1975.
- [3] J. F. Jarvis, C. N. Judice, and W. H. Ninke, “A survey of techniques for the display of continuous-tone pictures on bilevel displays,” *Comp. Graph. Image Proc.*, 5(1), pp. 13-40, 1976.
- [4] P. Stucki, “MECCA-A multiple-error correcting computation algorithm for bilevel image hardcopy reproduction,” Res. Rep. RZ1060, IBM Res. Lab., Zurich, Switzerland, 1981.
- [5] J. N. Shiao and Z. Fan, “A set of easily implementable coefficients in error diffusion with reduced worm artifacts,” *SPIE*, 2658: 222-225, 1996.
- [6] P. Li and J. P. Allebach, “Block interlaced pinwheel error diffusion,” *JEI*, 14(2), Apr-Jun. 2005.
- [7] V. Ostromoukhov, “A simple and efficient error diffusion algorithm,” *Computer Graphics (Proceedings of SIGGRAPH 2001)*, pp. 567-572, 2001.
- [8] R. Ulichney, *Digital halftoning*, Cambridge, MA: MIT Press, 1987.
- [9] D. E. Knuth, “Digital halftones by dot diffusion,” *ACM Trans. Graph.*, 6(4), Oct. 1987.
- [10] M. Mese and P. P. Vaidyanathan, “Optimized halftoning using dot diffusion and methods for inverse halftoning,” *IEEE Trans. Image Processing*, 9(4), pp. 691-709, Apr. 2000.
- [11] J. M. Guo and Y. F. Liu, “Improved dot diffusion by diffused matrix and class matrix co-optimization,” *IEEE Trans. Image Processing*, 18(8), pp. 1804-1816, Aug. 2009.
- [12] M. Analoui and J. P. Allebach, “Model based halftoning using direct binary search,”

- Proc. SPIE*, vol. 1666, pp. 96-108, Aug. 1992.
- [13] J. P. Allebach, “FM screen design using DBS algorithm,” in *Proc. IEEE ICIP*, vol. 1, Lausanne, Switzerland, pp. 549-552, 1996.
- [14] D. J. Lieberman and J. P. Allebach, “Efficient model based halftoning using direct binary search,” in *Proc. IEEE ICIP*, pp. 775-778, 1997.
- [15] D. J. Lieberman and J. P. Allebach, “A dual interpretation for direct binary search and its implications for tone reproduction and texture quality,” *IEEE Trans. Image Processing*, 9(11), pp. 1950-1963, 2000.
- [16] S. H. Kim and J. P. Allebach, “Impact of HVS models on model-based halftoning,” *IEEE Trans. Image Processing*, 11(3), pp. 258-269, 2002.
- [17] R. Näsänen, “Visibility of halftone dot textures,” *IEEE Trans. Syst., Man, Cybern.*, vol. 14, no. 6, pp. 920–924, 1984.
- [18] E. J. Delp and O. R. Mitchell, “Image compression using block truncation coding,” *IEEE Trans. Communications*, vol. 27, no. 9, pp. 1335-1342, Sept. 1979.
- [19] H. Yang and J. Yin, “A secure removable visible watermarking for BTC compressed images,” *Multimedia Tools and Applications*, vol. 74, no. 6, pp. 1725-1739, Sept. 2013.
- [20] C.-C. Lo, Y.-C. Hu, W.-L. Chen, and C.-M. Wu, “Reversible data hiding scheme for BTC-compressed images based on histogram shifting,” *International Journal of Security and its Applications*, vol. 8, no. 2, pp. 301-314, 2014.
- [21] J. M. Guo and Y. F. Liu, “High capacity data hiding for error-diffused block truncation coding,” *IEEE Trans. Image Processing*, vol. 21, no. 12, pp. 4808-4818, Dec. 2012.
- [22] J. M. Guo, H. Prasetyo, and J. H. Chen, “Content-based image retrieval using error diffusion block truncation coding features,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 466-481, Mar. 2015.
- [23] J. M. Guo and H. Prasetyo, “Content-based image retrieval using features extracted from halftoning-based block truncation coding,” *IEEE Trans. Image Processing*, vol.

- 24, no. 3, pp. 1010-1024, Mar. 2015.
- [24] J. M. Guo, H. Prasetyo, and N. J. Wang, “Effective image retrieval system using dot-diffused block truncation coding features,” *IEEE Trans. Multimedia*, DOI: 10.1109/TMM.2015.2449234, 2015.
- [25] M. Lema and O. R. Mitchell, “Absolute moment block truncation coding and its application to color images,” *IEEE Trans. Communications*, vol. 32, no. 10, pp. 1148-1157, Oct. 1984.
- [26] Y. Wu and D. C. Coll. “Single bit-map block truncation coding of color images,” *IEEE J. Sel. Areas Commun*, vol. 10, no.5, pp. 952-959, Jun. 1992.
- [27] J. M. Guo “Improved block truncation coding using modified error diffusion,” *Electronics Letters*, vol. 44, no. 7, pp. 462-464, Mar. 2008.
- [28] J. M. Guo and M. F. Wu, “Improved block truncation coding based on the void-and-cluster dithering approach,” *IEEE Trans. Image Processing*, vol. 18, no. 1, pp. 211-213, Jan. 2009.
- [29] J. M. Guo and Y. F. Liu, “Improved block truncation coding using optimized dot diffusion,” *IEEE Trans. Image Processing*, vol. 23, no. 3, pp. 1269-1275, Mar. 2014.
- [30] Y. C. Hu, I. C. Chang, K. Y. Liu, and C. L. Hung, “Improved color image coding schemes based on single bit map block truncation coding,” *Optical Engineering*, vol. 53, no. 9, pp. 093104-093104, Sept. 2014.
- [31] P. Franti, P. Nevalainen, and T. Kaudoranta, “Compression of digital image by block truncation coding: a survey,” *The Computer Journal*, vol. 37, no. 4, pp. 308–332, Mar. 1994.
- [32] J. Mathews, M. S. Nair, and L. Jo, “A novel color image coding technique using improved BTC with k-means quad clustering,” *Advances in Signal Processing and Intelligent Recognition Systems*, vol. 264, pp. 347-357, 2014.
- [33] B. W. Kolpatzik and C. A. Bouman, “Optimized error diffusion for image display,” *J.*

- Electron. Imaging* 1, pp. 277–292, 1992.
- [34] S. Sugiura and T. Matika, “An improved multilevel error diffusion method,” *J. Imaging Sci. Technol.* 39, pp. 495–501.
- [35] Q. Yu and K. Spaulding, “Combining error diffusion, dithering and over-modulation for smooth multilevel printing,” in *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts V*, R. Eschbach and G. G. Marcu, eds., *Proc. SPIE* 3963, pp. 446–457, 2000.
- [36] F. Faheem, D. L. Lau, and G. R. Arce, “Multilevel halftoning using bilevel quantizers,” in *Proceedings of IS&T's 2001 Processing Images, Image Quality, Capturing Images, Systems (PICS) Conference*, pp. 51–54.
- [37] Q. Yu, K. J. Parker, K. Spaulding, and R. Miller, “Digital multitone with overmodulation for smooth texture transition,” *J. Electron. Imag.*, 8(3), pp. 311–321, Jul. 1999.
- [38] G. Lin and J. P. Allebach, “Multilevel screen design using direct binary search,” in *Proc. SPIE Conf. Ser.*, vol. 4663, pp. 264–277, Dec. 2001.
- [39] J. B. Rodríguez, G. R. Arce, and D. L. Lau, “Blue-noise multitone dithering,” *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1368–1382, Aug. 2008.
- [40] K. Chandu, M. Stanich, C. W. Wu, and B. Trager, "Direct multi-bit search (DMS) screen algorithm," in *Proc. IEEE ICIP*, pp. 817-820, 2012.
- [41] C. Lee and J. P. Allebach, “The Hybrid Screen-Improving the Breed,” *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 435-450, Feb. 2010.
- [42] J. M. Guo, J. Y. Chang, Y. F. Liu, G. H. Lai, and J. D. Lee, “Tone-replacement error diffusion for multitone,” *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4312–4321, Nov. 2015.
- [43] S. Winkler, C. J. van den Branden Lambrecht, and M. Kunt, “Vision and video: models and applications,” *Vision Models and Applications to Image and Video Processing*.

- Springer US, p. 201-229, 2001.
- [44] J. Z. Wang, J. Li, G. Wiederhold, "SIMPLIcity: semantics-sensitive integrated matching for picture libraries," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947-963, Sept. 2001.
- [45] Y. F. Liu, J. M. Guo, B. S. Lai, and J. D. Lee, "High efficient contrast enhancement using parametric approximation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2444-2448, May. 2013.
- [46] J. M. Guo and Y. F. Liu, "Joint compression/watermarking scheme using majority-parity guidance and halftone-based block truncation coding," *IEEE Trans. Image Processing*, vol. 19, no. 8, pp. 2056-2069, Aug. 2010.
- [47] J. M. Guo and Y. F. Liu, "New class tiling design for dot-diffused halftoning," *IEEE Trans. Image Processing*, vol. 22, no. 3, pp. 1199-1208, Mar. 2013.
- [48] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Processing*, vol. 15, no. 11, pp. 3440-3451, Nov. 2006.
- [49] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: a feature similarity index for image quality assessment," *IEEE Trans. Image Processing*, vol. 20, no. 8, pp. 2378-2386, Aug. 2011.
- [50] G. Schaefer and M. Stich, "UCID – an uncompressed color image database," in *Proc. SPIE, Storage and Retrieval Methods and Applications for Multimedia*, vol. 5307, pp. 472-480, Jan. 2014.
- [51] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, vol. 2, pp. 1398-1402, Nov. 2003.
- [52] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Trans. Image Processing*, vol. 20, no. 5, pp. 1185-1198, May. 2011.

-
-
- [53] H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” *IEEE Trans. Image Processing*, vol. 15, no. 2, pp. 430-444, Feb. 2006.
- [54] *NTUST MSP Lab Image Database*. (2014) [Online]. Available:
<http://msp.ee.ntust.edu.tw/public%20file/ImageSet.rar>
- [55] *Kodak lossless true color image suite*. (2004) [Online]. Available:
<http://r0k.us/graphics/kodak>
- [56] H. Jégou, M. Douze, and C. Schmid, “Hamming embedding and weak geometry consistency for large scale image search,” *Proceedings of the 10th European conference on Computer vision*, Oct. 2008.
- [57] C. Lee and J. P. Allebach, “The hybrid screen-improving the breed,” *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 435-450, Feb. 2010.
- [58] Y. F. Liu, and J. M. Guo, “Dot-diffused halftoning with improved homogeneity,” *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4581–4591, Nov. 2015.