植基於區塊差值擴張的可回復式資訊偽裝演算法
Reversible Steganography Using Block-based Difference Expansion

黃姵甄
中興大學資訊科學與工程學系
e-mail：s9656047@cs.nchu.edu.tw

王皓正
中興大學資訊科學與工程學系
e-mail：phd9516@cs.nchu.edu.tw

王宗銘
中興大學資訊網路與多媒體研究所/資訊科學與工程學系
e-mail：cmwang@cs.nchu.edu.tw

摘要

本論文提出一個灰階影像可回復資訊偽裝演算法。我們的方法具備可回復式的特性，即在取出秘密資訊後還可完整的重建原來的灰階影像。本方法首先利用區塊化掩護影像，將影像切割成不同的區塊。接著，我們藉由分類，將這些區塊區分到不同的集合。最後，我們利用區塊差值擴張技術嵌入秘密資訊。藉此，我們的方法不但可以提高資訊嵌入量，也同時能減少額外所需記錄的資訊量。藉由上述的特性，我們的演算法可以有效地達到高嵌入量及低失真性的目的，兼具還原掩護影像的特性。實驗結果與現存技術[8][12]比較後顯示：我們的演算法不但大幅提升嵌入量，且所產生之偽裝影像亦具有良好的視覺品質。經量化顯示：平均每一個像素之淨嵌入量可高達 0.66 位元，其 PSNR 值能維持在 35 dB 以上，具有極佳的視覺品質。

關鍵詞：偽裝學、可回復性、區塊差值擴張技術

Abstract

This paper proposes a reversible steganography for gray images. Being reversible, the cover gray image can be recovered from the stego image completely after the secret message being extracted. Our algorithm first partitions the cover image into a number of blocks, each of which represents a sub-image. Then, we classify pixels inside a block with respect to its features, leading to producing a number of candidate blocks that are able to embed secret message. Finally, we exploit the difference expansion technique on candidate blocks to embed secret message. This approach allows us to embed large capacity of secret message, yet lowering the pixel variation due to data embedding. Experimental results show that, on average, our algorithm achieves the capacity for up to 0.66 bits per pixel (bpp). In addition, the peak signal-to-noise ratio (PSNR) for the stego image is above 35 dB. Experimental results verify that our algorithm outperforms than our counterparts [8][12]. We conclude that our algorithm employing the block-based difference expansion is feasible to provide a reversible solution for steganography on gray images.

Keywords: steganography, reversibility, block-based difference expansion

1. 前言

所謂偽裝學(steganography)[9,11]是將秘密資訊隱藏在常見且廣泛被使用的掩護多媒體物件上(cover media)，如影像、文字、語音或視訊等。在掩護多媒體物件嵌入秘密資訊後即成為偽裝物件(stego media)，可使惡意或無意的使用者在取得偽裝物件後無法得知此物件所嵌入的秘密資訊，以達到秘密通訊的目的。

一般來說，影像資訊偽裝技術可以粗略的分成兩類，分別為非可回復式(irreversible)與可回復式(reversible)資訊偽裝技術。非可回復式資訊偽裝技術，在嵌入秘密資訊後，會對掩護影像造成永久性的失真。僅管這些影像遭受輕微的破壞與失真，以人類的感官視覺是察覺不出來的，但對於某些專業領域而言，即使是微小的失真也是不被接受的，如法律、軍事及醫學用途等，都必須維持影像的完整性。所以，

可回復式資訊偽裝技術或稱為無失真性資訊偽裝技術(lossless steganography)即被提出用來確保秘密資訊取出後，仍可回復至原始掩護影像以維持影像的完整性。

近年來，所提出的可回復式資訊偽裝技術主要可分成三類：(1)資訊壓縮技術[4,6]：將原始影像所有像素值的最低有效位元(least significant bit，LSB)取出，利用無失真壓縮技術壓縮所有最低有效位元，建構出額外的空間來嵌入秘密資訊。由於原始像素 LSB 平面位元的分佈較為零散，所以無法得到較佳的壓縮率，因此，此類技術所能達到的可嵌入量是有限的。(2)直方圖技術[5,10]：從影像的特徵中產生直方圖，並從直方圖中，找到一組峰點(peak point)與零點(zero point)，依照所嵌入的秘密資訊“0”或“1”修改峰點所代表的像素值。此類技術的優點在於簡單且有效率。但是，很少影像會有很大量的相同像素值，所以此技術的可嵌入量亦是非常有限的。(3)差值擴張技術[1-3,7-8,12-13]：計算像素之間的差異，產生差值，再將秘密資訊嵌入到擴張後的差值之中。此類技術可達到較高的可嵌入量。然而，此類技術為了達到可回復性，在嵌入資訊後，通常會產生一份位置對映圖(location map)的負載資訊，用來判斷哪些像素是進行過擴張的動作。

在本論文中，主要是改進 Kim 等人所提出的新穎差值擴張轉換技術[8]，提出一個植基於區塊差值擴張的可回復式資訊偽裝演算法，透過本文所提出技術嵌入秘密資訊，可達到可回復式、高嵌入量及低影像失真性等目的。本論文架構如下：第二章相關文獻探討。第三章詳細說明本論文的演算法。第四章是演算法運行的實驗結果，針對嵌入量和影像品質進行分析。第五章則是結論以及未來方向。

2. 相關文獻探討

本章共分成兩節。在 2.1 節介紹 Tian 所提出的差值擴張(difference expansion，DE)技術[12]；接著，在 2.2 節介紹 Kim 等人於 2008 年所提出的新穎差值擴張轉換技術[8]，其方法主要改進[12]所提出的差值擴張技術，利用簡化的位置對映圖(location map)與新的擴張技術，以提高資訊嵌入量並維持低影像失真性。

2.1 Tian 所提方法

Tian 於 2003 年提出一個應用於灰階影像的可回復式資訊隱藏技術[12]。此方法利用所提出的差值擴張技術，將資訊嵌入於擴張後的差值之中。首先，將原始影像中相鄰兩個像素視為一對，每對像素皆不和其它對像素重疊。接著，計算每對像素的差異，得到差值，挑選出不會溢位(overflow)或欠位(underflow)的像素的差值來嵌入一個位元。嵌入的資訊包含位置對映圖(location map)、原始差值的 LSB 及秘密資訊。

假設有兩個像素值 A = 120、B = 115，我們要嵌入的一個位元值 $s \in \{0,1\}$。首先，計算 A 和 B 的整數平均值 l 及差值 d，這裡的 $\lfloor\ \rfloor$ 符號是指小於或等於括弧值的最大整數值。

$$l = \left\lfloor \frac{A+B}{2} \right\rfloor = \left\lfloor \frac{120+115}{2} \right\rfloor = 117,$$
$$d = A - B = 120 - 115 = 5.$$

接著，假設嵌入的一個位元 $s = 0$，計算新的差值 $d' = 2 \times d + s = 2 \times 5 + 0 = 10$，差值($d$)由原先的三位元擴張為四位元($d'$)。再利用新的差值($d'$)及原始整數平均值 l 產生新的像素對 (A', B')，如此即能完成嵌入資訊的動作：

$$A' = l + \left\lfloor \frac{d'+1}{2} \right\rfloor = 117 + \left\lfloor \frac{10+1}{2} \right\rfloor = 117 + 5 = 122,$$
$$B' = l - \left\lfloor \frac{d'}{2} \right\rfloor = 117 - \left\lfloor \frac{10}{2} \right\rfloor = 117 - 5 = 112.$$

當然，從嵌入後的新像素對 (A', B') 中，我們可以擷取出嵌入的秘密資訊及恢復原始像素值 (A, B)。首先，計算 A' 和 B' 的整數平均值 l' 及差值 d'：

$$l' = \left\lfloor \frac{A'+B'}{2} \right\rfloor = \left\lfloor \frac{122+112}{2} \right\rfloor = 117,$$
$$d' = A' - B' = 122 - 112 = 10.$$

接著，將 d' 轉換為二進制：$d' = 10_{10} = 1010_2$，取出 d' 的 LSB 即可得到嵌入的 s 值(s = 0)，原始的差值即為 $d = 101_2$：

$$s = d' - \left\lfloor \frac{d'}{2} \right\rfloor \times 2 = 10 - \left\lfloor \frac{10}{2} \right\rfloor \times 2 = 0,$$
$$d = \left\lfloor \frac{d'}{2} \right\rfloor = \left\lfloor \frac{10}{2} \right\rfloor = 5.$$

所以，利用整數平均數 l' 和還原後的差值 d，即可恢復原始像素對 (A, B)。

$$A = l' + \left\lfloor \frac{d+1}{2} \right\rfloor = 117 + \left\lfloor \frac{5+1}{2} \right\rfloor = 117 + 3 = 120,$$
$$B = l' - \left\lfloor \frac{d}{2} \right\rfloor = 117 - \left\lfloor \frac{5}{2} \right\rfloor = 117 - 2 = 115.$$

然而，差值經過擴張後可能會產生溢位或欠位的問題，所以，擴張後的差值必須在像素範圍[0,255]之內。也就是說，不是所有的像素

對皆可以擴張，為了清楚判斷所有經過差值擴張處理的差值位置，此時，會產生位置對映圖，用來判斷哪些差值有經過擴張的動作。

[12]所提出方法，主要會將差值分成 *EZ*、*EN*、*CN*、*NC* 四個集合，其中：

EZ：包含全部 d = 0 及 d = -1 的可擴張差值。若整數平均值 l 在 s = 0 及 s = 1 的情況下，皆滿足條件(1)，則稱差值 d 為可擴張的(expandable)。

$$\begin{aligned} 0 \le l + (2 \times d + s) \le 255, \\ 0 \le l - (2 \times d + s) \le 255. \end{aligned} \tag{1}$$

EN：包含全部不是 *EZ* 的可擴張差值。[12]使用門檻值 *T*(threshold)再區分為 $|d| \le T$ (稱為 EN_1)及 $|d| > T$ (稱為 EN_2)兩種差值。

CN：包含全部可變的及不是 *EZ*、*EN* 的差值。若整數平均值 l 在 s = 0 及 s = 1 的情況下，皆滿足條件(2)，則稱差值 d 為可變的(changeable)。

$$\begin{aligned} 0 \le l + (2 \times \left\lfloor \frac{d}{2} \right\rfloor + s) \le 255, \\ 0 \le l - (2 \times \left\lfloor \frac{d}{2} \right\rfloor + s) \le 255. \end{aligned} \tag{2}$$

NC：不改變的差值均是此集合。

由上述可以得知，只有 *EZ* 和 EN_1 集合的差值屬於可擴張的，因此，需設定該集合內的每一個差值在位置對映圖中的位元為 1；而為了避免嵌入後會產生太大的影像失真性，EN_2 集合的差值是屬於不可擴張的，因此，對於屬於 EN_2、*CN* 及 *NC* 集合的差值，需設定該集合內的每一個差值在位置對映圖中的位元為 0。所以，位置對映圖的大小為 $EZ \cup EN \cup CN \cup NC$ 的集合個數。也就是說，位置對映圖的大小為原始影像的一半。[12]所提方法需記錄 EN_2 和 *CN* 兩種集合中差值的原始 LSB 到可改變位元集合(changeable set)，因此，可改變位元集合的大小為 $EN_2 \cup CN$ 的集合個數。最後，位置對映圖、可改變位元集合及秘密資訊會被組合成位元串流(bitstream)，經無失真壓縮技術壓縮後嵌入到影像之中。

此方法的額外所需記錄資訊，包含位置對映圖及可改變位元集合。在不壓縮額外資訊的情況下，會產生影像無足夠的空間可以嵌入額外資訊的問題。因此，Kim 等人提出一個新穎的差值擴張轉換技術[8]，將差值分成可擴張(expandable)與不可擴張(inexpandable)兩種集合，不考慮可改變(changeable)的差值，所以，不需要記錄原始差值的 LSB。並且，利用門檻值重新定義集合，有效地減少位置對映圖的大小，並達到高嵌入量的目的。

2.2 Kim 等人所提方法

Kim 等人於 2008 年提出一個應用於灰階影像的可回復式資訊嵌入技術[8]。此方法是基於[12]的差值擴張技術，提出一個新穎的差值擴張轉換(difference expansion transform)技術，透過其方法可以簡化額外所需記錄資訊量的大小，因此可提高可嵌入量，亦維持較低的影像品質失真性。此方法在進行嵌入處理之前，需定義一個門檻值 T，用以避免過大的擴張差值，導致影像品質失真性大幅降低。首先，將原始影像中的像素分對，計算每對像素的差異，得到差值，挑選出不會溢位或欠位的像素的差值來嵌入一個位元。嵌入的資訊包含簡化後的位置對映圖(location map)及秘密資訊。

假設 T 為已知的門檻值，定義 d 為像素對的差值和 l 為像素對的整數平均值。[8]所提出的新穎差值擴張轉換技術如條件(3)和(4)所示，透過上述兩個條件可以將差值分成兩種集合，分別為可擴張集合(expandable set)和不可擴張集合(inexpandable set)。利用條件(4)進行分類，是為了避免嵌入後會產生溢位或欠位的問題。除此之外，更可藉由此條件過濾掉不可擴張的差值可以有效地簡化位置對映圖的大小。

$$|d| \le T. \tag{3}$$

$$T \le l < 255 - T. \tag{4}$$

對於可擴張集合中的差值，利用門檻值 T，將差值 d 進行分類，共分成四種不同的集合，分別為：(1)當 $|d| \le \lfloor T/2 \rfloor$ 時，分類 d 到 M 集合。(2)當 $\lfloor T/2 \rfloor < |d| \le T$ 時，分類 d 到 N_e 集合。(3)當 $T < |d| \le 2T+1$ 時，分類 d 到 $N_{\tilde{e}}$ 集合。(4)當 $|d| > 2T+1$ 時，分類 d 到 U 集合。對於 M 及 N_e 集合的差值，將利用差值擴張技術[12]嵌入一個位元的資訊而對於 $N_{\tilde{e}}$ 及 U 集合的差值，將不進行任何的嵌入處理，維持原始相鄰像素之間的差異。由上述可以發現，M 集合經擴張後，差值所在的範圍會變成 $-T \le d' \le T$。而 N_e 集合經擴張後，差值所在範圍會變成 $T < d' \le 2T+1$ 或 $-2T \le d' < -T$。由此可見，N_e 與 $N_{\tilde{e}}$ 集合的差值會互相混肴，此時，集合內的每一個差值皆需利用一個位元來表示原先是位在哪個集合中，確保接收端能正確判斷原始差值屬於 N_e 或 $N_{\tilde{e}}$ 集合。而 N_e 及 $N_{\tilde{e}}$ 集合內的位元集合即為位置對映圖，其大小為 $|N_e| + |N_{\tilde{e}}|$，在此|·|代表該集合的位元個數。

此方法強調在不壓縮位置對映圖的情況

下，所能達到的嵌入量仍高於[12]所提方法，表示此方法有效地以簡化位置對映圖的技術，增加可嵌入量。

3. 我們所提的演算法

本論文提出一個植基於區塊差值擴張的可回復式資訊偽裝演算法。透過區塊化的方式，可以更進一步精簡位置對映圖的大小，以達到更高的嵌入量。首先，掩護影像會以 $m \times n$ 大小且不相重疊的方式切割成許多區塊，針對每一個區塊，會依照其複雜度的不同，將區塊分類到四種不同的集合，針對四種集合內的區塊分別利用本論文所提出的區塊差值擴張技術進行嵌入處理，透過此種區塊化的嵌入處理方式，可以產生更多的差值，以嵌入更多的資訊，達到提高嵌入量的目的，同時亦可以簡化位置對映圖的大小。

首先，我們將於 3.1 節介紹本論文所提出之區塊差值擴張技術；接著，在 3.2 節詳細說明本演算法流程；最後，在 3.3 節詳細說明本演算法擷取流程。

3.1 區塊差值擴張技術

首先，我們定義每一個區塊內的像素值分別以 $b_0, b_1, b_2, \ldots, b_{k-1}$ 表示，其中 $k = m \times n$。接著，利用公式(5)決定一個像素值 b_g，稱作區塊內的中間值，以此中間值 b_g 為參考像素值，利用公式(6)與區塊內其他像素值計算差異產生 k - 1 個差值，並且差值可依序被定義成{$d_0, d_1, d_2, \ldots, d_{g-1}, d_{g+1}, \ldots, d_{k-1}$}。相較於[8]所提出的技術，透過此種計算方式，可以產生更多的差值，有效地提高嵌入量。

$$b_g = b_{\left\lfloor \frac{k-1}{2} \right\rfloor}. \quad (5)$$

$$d_i = b_i - b_g, \quad 0 \le i \le k-1 \text{ and } i \ne g. \quad (6)$$

接下來，我們可藉由公式(7)得到一個新的差值，此差值則是包含原始差值及所要嵌入的秘密資訊所構成。

$$d_i' = \begin{cases} 2 \times d_i + s, & \text{if } b_i \ge b_g, \\ 2 \times d_i - s, & \text{if } b_i < b_g, \end{cases} \quad (7)$$

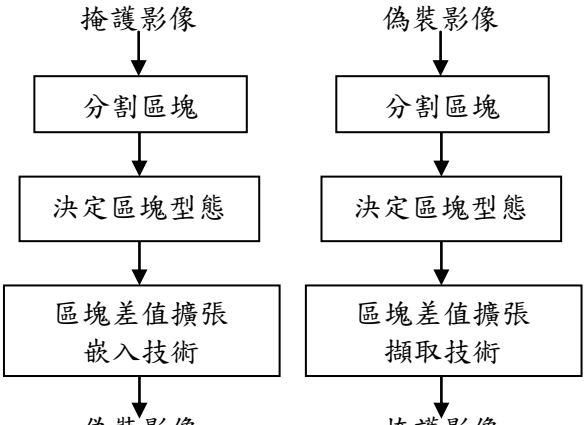
在此，s 代表每一個差值所要嵌入的秘密資訊，且滿足 $s \in \{0,1\}$。

本演算法可維持 b_g 與其他像素之間的關係在嵌入秘密資訊的前後皆不會有所改變；也就是說，在公式(7)中，若 b_i 大於或等於 b_g，則擴張後的差值 $2 \times d_i$ 必須加上秘密資訊 s；反之，若 b_i 小於 b_g，則擴張後的差值 $2 \times d_i$ 必須減去秘密資訊 s。藉由以上的調整，完成秘密資訊嵌入處理之後，每一個區塊的 b_g 與其他像素之間的關係將維持不變。最後，即可利用公式(8)產生新的像素值。

$$b_i' = b_g + d_i', \quad \text{where } 0 \le i \le k-1 \text{ and } i \ne g. \quad (8)$$

3.2 演算法流程

如圖 1 所示，本演算法由嵌入資訊與擷取資訊兩大獨立程序所構成，主要可分為分割區塊、決定區塊型態及區塊差值擴張嵌入技術三個部份。首先，掩護影像會被以 $m \times n$ 大小且不相重疊的方式切割成數個區塊；接著，將以循序掃描的方式依序對每一個區塊進行分類，每一個區塊將會被分成可嵌入與不可嵌入兩大類。除了判別區塊是否能嵌入資訊之外，本演算法還會進一步的將區塊分成四種集合；最後，針對每特定集合內的區塊利用本論文所提出的區塊差值擴張技術進行嵌入處理，透過此技術可以達到更高的嵌入量，同時針對區塊內像素值變動範圍較大的區塊保持不變，以維持較少的影像失真性。以下將會詳細說明本演算法如何利用區塊差值擴張技術將秘密資訊嵌入差值之中，並簡化位置對映圖的大小，以達到提高嵌入量及低影像品質失真性的目的。



(a) 嵌入程序　(b) 擷取程序

圖 1：本演算法嵌入和擷取流程圖

3.2.1 分割區塊

對於一張尺寸為 $h \times w$ 的灰階影像，在嵌入秘密資訊之前，我們必須先決定切割區塊大小，以 $m \times n$ 表示，其中 m 為水平方向的像素個數；n 為垂直方向的像素個數。接著，每一個區塊依照公式(6)可產生 k - 1 個差值，分別為 $d_0, d_1, d_2, \ldots, d_{g-1}, d_{g+1}, \ldots, d_{k-1}$。最後，對於每一個區塊，利用公式(9)找出最大絕對差值 d_{max}

。我們可以根據 d_{max} 將每一個區塊分類到不同的集合並且進行相對應的處理。

$$d_{max} = max\{|d_0|, |d_1|, |d_2|, \ldots, |d_{g-1}|, |d_{g+1}|, \ldots, |d_{k-1}|\}. \quad (9)$$

3.2.2 決定區塊型態

在計算出最大絕對差值 d_{max} 後，可根據事先定義的門檻值 T 將區塊分類到不同集合。假設一個區塊的中間值 b_g 及最大絕對差值 d_{max} 同時滿足公式(10)及(11)，即歸類為可嵌入區塊；反之，即歸類為不可嵌入區塊。

$$d_{max} \le T. \quad (10)$$

$$2T < b_g < 255 - 2T, \quad (11)$$

其中公式(11)可以有效避免資訊嵌入的過程中會有像素值溢位或欠位的情況產生。除此之外，對於不可嵌入區塊可由公式判斷，因此不需要額外的位元幫助判斷是否為可嵌入，如此可以有效地更簡化位置對映圖的大小。

對於所有可嵌入區塊，利用區塊中的 d_{max} 及門檻值 T，將區塊分成四種不同集合，分別為：(1)如果 $d_{max} \le \lfloor T/2 \rfloor$，則分類此區塊屬於 ES_1 集合。(2)如果 $\lfloor T/2 \rfloor < d_{max} \le T$，則分類此區塊屬於 ES_2 集合。(3)如果 $T < d_{max} \le 2T+1$，則分類此區塊屬於 NE 集合。(4)如果 $d_{max} > 2T+1$，則分類此區塊屬於 NS 集合。對於屬於 ES_1 及 ES_2 集合的區塊，將利用公式(7)針對區塊內的 k - 1 個差值各嵌入一個位元的資訊。而對於屬於 NE 及 NS 集合的區塊，將不進行任何的嵌入處理，維持區塊內原始差異。

由上述的分類可以發現，屬於 ES_1 集合的區塊，該區塊內的 k - 1 個差值經擴張後，利用公式(9)找到最大絕對差值 d_{max}，其所在的範圍為 $d_{max} \le T$，可以保證此集合的區塊不會與其他集合的區塊互相混肴。而 ES_2 集合的區塊，該區塊內的 k - 1 個差值經擴張後，利用公式(9)找到 d_{max}，其所在範圍為 $T < d_{max} \le 2T+1$。由此可見，ES_2 與 NE 集合的區塊在嵌入後會互相混肴，此時，需用一個位元記錄該區塊是否有經過擴張並存放於位置對映圖 L 當作額外所需記錄資訊，確保接收端能正確判斷該區塊原是屬於 ES_2 或 NE 集合。對於 ES_2 及 NE 集合中的每一區塊，皆需花費一個位元，所以，位置對映圖的大小為 $L = |ES_2| + |NE|$，在此|·|代表該集合的位元個數。

3.2.3 區塊差值擴張嵌入技術

經由 3.2.2 節決定區塊的型態，針對每一個可嵌入資訊的區塊內的差值進行嵌入處理，嵌入程序步驟，如下所示：

步驟 1：統計 ES_1 集合的區塊數量，以 E 表示 ES_1 集合的最大嵌入量，即 $E = |ES_1| \times (k - 1)$。當 $E \ge L$，代表 ES_1 集合的最大嵌入量大於或等於額外所需記錄資訊量，跳至步驟 2。反之，代表 ES_1 集合的最大嵌入量小於額外所需記錄資訊量，表示所產生的額外資訊量過大，影像無足夠空間可以嵌入額外所需記錄資訊，進而忽略這次的嵌入處理，但以一般自然影像而言，E 通常會大於 L。

步驟 2：先針對屬於 ES_1 集合的區塊進行資訊嵌入的動作。利用公式(7)將資訊嵌入到每一個差值之中，完成資訊嵌入處理後，即可透過公式(8)產生偽裝影像像素值。在此步驟必須先嵌入長度為 L 的位置對映圖，若 $E - L$ 大於 0，代表還有剩餘空間可用來嵌入秘密資訊。

步驟 3：對於屬於 ES_2 集合的區塊進行秘密資訊嵌入處理。利用公式(7)將資訊嵌入到每一個差值之中，完成資訊嵌入處理後，即可透過公式(8)得到偽裝影像像素值。

步驟 4：所有區塊完成嵌入處理動作之後，即產生一張偽裝影像並可將偽裝影像及門檻值 T 送至接收端。

3.3 擷取流程

當接收端收到偽裝影像及門檻值 T，如同圖 1(b)的擷取流程。首先，以 $m \times n$ 大小且不相重疊的方式切割影像成為區塊，接著，透過以下三個步驟擷取出秘密資訊，依不同型態的區塊重建掩護影像。擷取步驟如下：

步驟 1：對於每一個區塊，首先，利用公式(5)找到中間值 b_g；接著，利用公式(6)計算區塊內其他像素值與 b_g 之間的差異，得到 k - 1 個差值；最後，利用公式(9)找出區塊內的最大絕對差值 d_{max}，依此 d_{max} 及門檻值 T 對區塊進行分類，在滿足公式(11)的情況下，當 $d_{max} \le T$，則此區塊屬於 ES_1 集合；反之，當 $T < d_{max} \le 2T+1$，則此區塊屬於 ES_2 或 NE 集合，須依額外記錄資訊(即位置對映圖)來識別。

步驟 2：定義秘密資訊擷取、差值還原及像素值還原公式如下：

$$s = LSB(|d_i'|), \quad (12)$$

其中 $LSB(|d_i'|)$ 表示 $|d_i'|$ 的最低有效位元。

$$d_i = \begin{cases} \lfloor d_i'/2 \rfloor, \text{if } d_i' \ge 0, \\ \lceil d_i'/2 \rceil, \text{if } d_i' < 0, \end{cases} \quad (13)$$

where $0 \le i \le k-1$ and $i \ne g$.

$$b_i = b_g + d_i, \quad \text{where } 0 \le i \le k-1 \text{ and } i \ne g. \qquad (14)$$

對於 ES_1 集合的區塊利用公式(12)取出區塊內 k - 1 個差值所嵌入的 k - 1 位元資訊。此步驟所取出的資訊共有 $|ES_1| \times (k-1)$ 位元，其中包含所嵌入的位置對映圖及秘密資訊。取出嵌入的資訊後，即透過公式(13)得到原始差值，並利用公式(14)還原原始像素值。同時再利用擷取出的位置對映圖得知屬於 ES_2 集合的所有區塊。

步驟 3：利用公式(12)擷取出嵌入在 ES_2 集合中每一個區塊的秘密資訊，並利用公式(13)得到原始差值，最後，再以公式(14)還原原始像素值。

經由上述三個步驟後，即可完成秘密資訊擷取程序並還原回掩護影像。

4. 實驗結果與分析

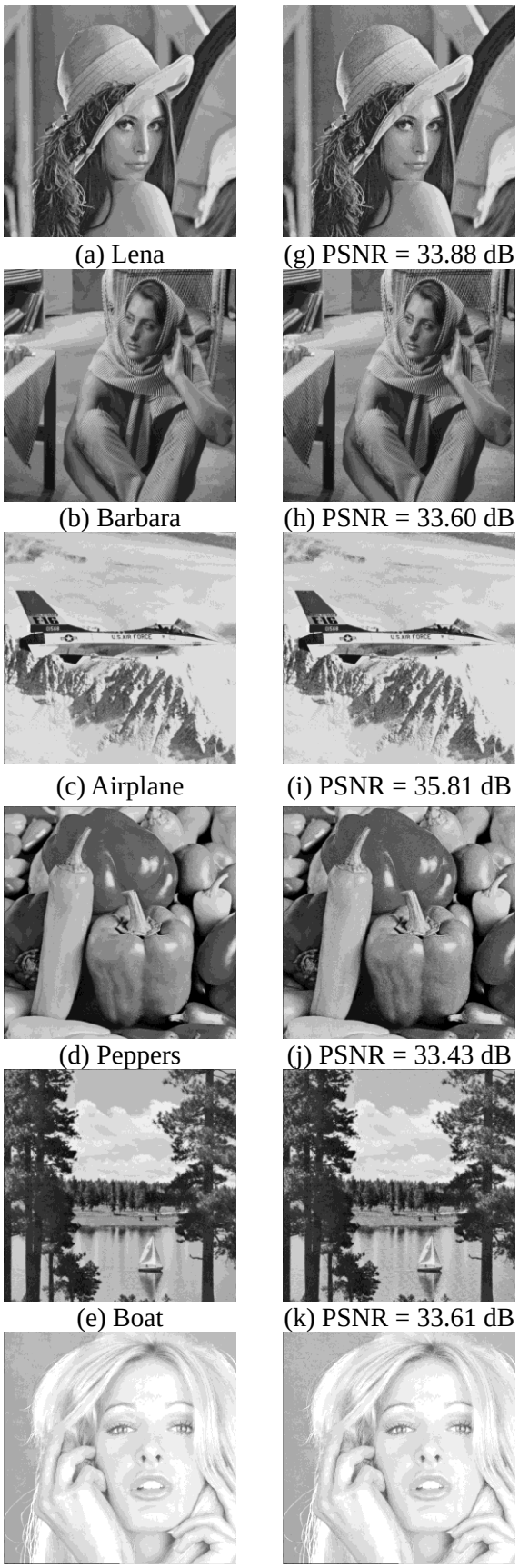
在本論文的實驗中，我們以六張 8 位元的灰階影像當作受測影像，用來評估所提方法的嵌入量與影像品質，每一張掩護影像的大小為 512 × 512，如圖 2(a) - (f)所示。利用範圍{0,1}的亂數作為所要嵌入的秘密資訊，並且利用峰值信噪比(Peak Signal to Noise Ratio，簡稱 PSNR)來評估影像品質失真性。

由實驗結果證明本論文的方法，在門檻值 T = 32 的情況下，所得到的偽裝影像，如圖 2(g) - (l)所示，能維持 PSNR 值在 33 dB 以上，且人眼視覺不易察覺其差異。

由第 3 節中的說明可以得知，在嵌入過程中會產生位置對映圖，即為我們所要記錄的額外資訊，透過本論文所提出之區塊差值擴張技術，可以有效利用區塊化影像及分類區塊的方式簡化位置對映圖的大小。本論文假設可嵌入量為在不壓縮額外資訊的條件下，扣除額外資訊後，所得到的結果為實際的可嵌入量，在此稱作淨嵌入量(pure payload)。所以，定義本論文的淨嵌入量，如公式(15)：

$$P = C - L. \qquad (15)$$

其中，P 為淨嵌入量、C 為掩護影像所能提供的可嵌入量、L 為位置對映圖所需要的位元數。



(a) Lena (g) PSNR = 33.88 dB
(b) Barbara (h) PSNR = 33.60 dB
(c) Airplane (i) PSNR = 35.81 dB
(d) Peppers (j) PSNR = 33.43 dB
(e) Boat (k) PSNR = 33.61 dB
(f) Tiffany (l) PSNR = 38.58 dB

圖 2：本論文的六張掩護影像及偽裝影像

表 1 列出受測影像“Lena”在不同門檻值 *T* 之下的嵌入量、每一個像素的平均位元攜帶量 bpp(bits per pixel)、淨嵌入量及 PSNR 值。藉由表 1 可以得知我們所提出方法的嵌入量在扣除額外所需記錄資訊量(即位置對映圖)後，其淨嵌入量仍可達到 0.38 ~ 0.66 bpp，而維持 PSNR 值在 35 dB 以上。證明偽裝影像具有良好的視覺品質，人眼不易察覺其差異。圖 3 表示，對於所有受測影像，以不同的門檻值 *T* 所得到的淨嵌入量(pure payload)及 PSNR 值。

表 1：受測影像-Lena 的嵌入量及 PSNR 值

門檻值 (*T*)	嵌入量 (bits)	bpp	淨嵌入量 (bits)	bpp	PSNR (dB)
8	119040	0.45	98897	0.38	41.28
12	157904	0.60	141007	0.54	38.10
16	176960	0.68	163835	0.63	36.31
18	181976	0.69	170384	0.65	35.69
20	183600	0.70	173453	0.66	35.18

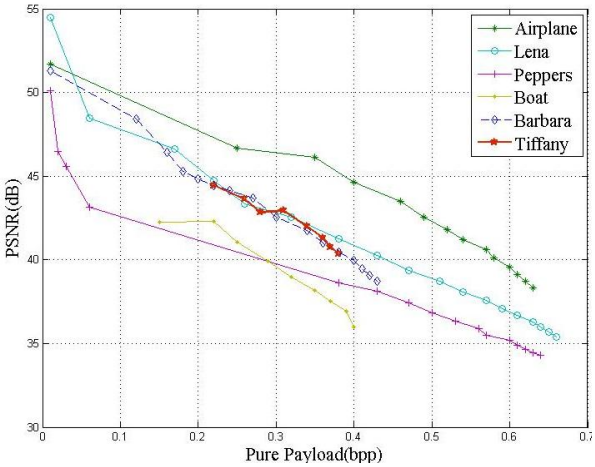


圖 3：不同門檻值下，所有受測影像的淨嵌入量及 PSNR 值

如圖 4 所示，受測影像“Lena”根據不同影像品質所得到的嵌入量。圖中列出本論文與[8]的淨嵌入量，而[12]為扣除壓縮的額外資訊後所得到的嵌入量，所以，[12]的淨嵌入量應會比圖中的更少。將[8]及[12]的方法與本論文進行比較，可證明在淨嵌入量及影像品質上，本論文皆有較好的結果。觀察圖 4 可以得知，本論文方法相較於[8]與[12]所提方法，淨嵌入量可以高達 0.66 bpp 且維持 PSNR 值在 35 dB 以上，所能達到的淨嵌入量高於[8]與[12]所提方法約 0.2 bpp，影像品質亦多約 2 ~ 4 dB。

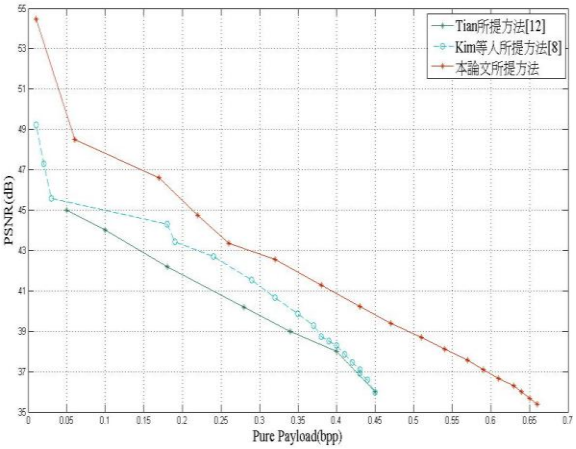
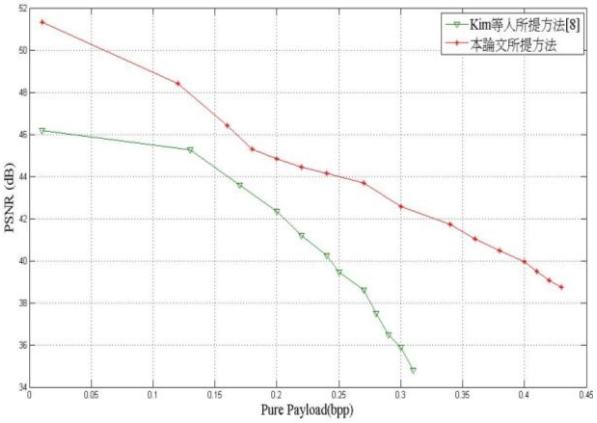
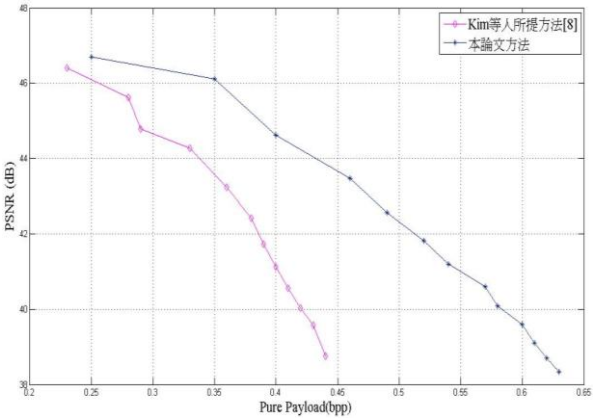


圖 4：嵌入量及 PSNR 值的比較(以 Lena 為主)

本論文方法強調在扣除額外所需記錄資訊後，所得到的淨嵌入量才是實際的嵌入量。由圖 5 可以證明，與[8]所提方法比較，本論文的淨嵌入量及影像品質皆會高於[8]所提方法。以受測影像“Barbara”來說，在 PSNR 為 39 dB 的情況下，本論文的淨嵌入量可達 0.43 bpp，相對地[8]大約只有 0.27 bpp。因此，本論文所提方法，能有效地減少額外所需記錄資訊量的大小，提升淨嵌入量，達到比[8]更高的淨嵌入量，並且維持良好的影像視覺品質。



(a) Barbara



(b) Airplane

圖 5：淨嵌入量及 PSNR 值的比較

5. 結論及未來工作

本論文提出植基於區塊差值擴張的可回復式資訊偽裝技術。我們的演算法主要透過三個步驟來嵌入秘密資訊。首先是進行區塊分割，以 $m \times n$ 大小且不相重疊的方式切割掩護影像成為區塊，找出區塊內的中間值之後，即可產生 $k-1$ 個差值；接著，我們決定區塊型態，從 $k-1$ 個差值中選定一個最大絕對差值，利用此最大絕對差值將區塊分成四種集合；最後，我們嵌入秘密資訊，利用區塊差值擴張技術嵌入秘密資訊到差值之後，即完成嵌入動作。由此三個步驟，可以建構出可回復式、高容量且低影像失真性的偽裝影像。

由實驗結果證明本方法的嵌入量在扣除所需記錄的位置對映圖之後，其淨嵌入量仍可維持高達 0.66 bpp 且影像品質在 35 dB 以上，具有極佳的視覺品質。更進一步地，透過本論文所列出與[8]和[12]所提出方法的實驗結果比較，可以證明本方法可以達到更高的淨嵌入量與影像品質。由於利用區塊差值擴張技術產生差值的概念，使得本方法所要記錄的位置對映圖相對減少很多，所以，可以保證在不壓縮位置對映圖的情況下，淨嵌入量仍可高於[8]與[12]所提出方法的淨嵌入量。

我們提出的可回復式資訊偽裝技術在未來研究的課題將朝向三個方向繼續延伸探討：第一、增加嵌入的資訊量，可與各種壓縮技術結合運用。第二、將影像品質提高。及第三、提高廣泛應用性，應用於其他掩護多媒體物件或各類型影像上。

參考文獻

[1] Alattar, A. M., “Reversible Watermark Using Difference Expansion of Triplets,” ***in Proceedings International Conference on Image Processing***, Vol. 1, pp. 501-504, 2003.

[2] Alattar, A. M., “Reversible Watermark Using Difference Expansion of Quads,” ***in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing***, Vol. 3, pp. 377-380, 2004.

[3] Alattar, A. M., “Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform,” ***IEEE Transactions on Image Processing***, Vol. 13, No. 8, pp. 1147-1156, 2004.

[4] Celik, M. U., Sharma, G., Tekalp, A. M., and Saber, E., “Lossless Generalized-LSB Data Embedding,” ***IEEE Transactions on Image Processing***, Vol. 14, No. 2, pp. 253-266, 2005.

[5] Fallahpour, M. and Sedaaghi, M. H., “High capacity lossless data hiding based on histogram modification,” ***IEICE Electronics Express***, Vol. 4, No. 7, pp. 205-210, 2007.

[6] Fridrich, J., Goljan, M., and Du, R., “Invertible authentication,” ***Proceedings SPIE Security and Watermarking of Multimedia Contents***, Vol. 4314, pp. 197-208, 2001.

[7] Kamstra, L. and Heijmans, H. J. A. M., “Reversible Data Embedding Into Images Using Wavelet Techniques and Sorting,” ***IEEE Transactions on Image Processing***, Vol. 14, No. 12, pp. 2082-2090, 2005.

[8] Kim, H. J., Shi, Y. Q., Nam, J., and Choo, H. G., “A Novel Difference Expansion Transform for Reversible Data Embedding,” ***IEEE Transactions on Information Forensics and Security***, Vol. 3, No. 3, pp. 456-465, 2008.

[9] Moulin, P. and Koetter, R., “Data-Hiding Codes,” ***Proceedings of the IEEE***, Vol. 93, No. 12, pp. 2083-2126, 2005.

[10] Ni, Z., Shi, Y. Q., Ansari, N., and Su, W., “Reversible Data Hiding,” ***IEEE Transactions on Circuits and Systems for Video Technology***, Vol. 16, No. 3, pp. 354-362, 2006.

[11] Petitcolas, F. A. P., Anderson, R. J., and Kuhn, M. G., “Information Hiding – A Survey,” ***Proceedings of the IEEE: Special Issue on Identification and Protection of Multimedia Content***, Vol. 87, pp. 1062-1078, 1999.

[12] Tian, J., “Reversible Data Embedding Using a Difference Expansion,” ***IEEE Transactions on Circuits and Systems for Video Technology***, Vol. 13, No. 8, pp. 890-896, 2003.

[13] Thodi, D. M. and Rodriguez, J. J., “Expansion Embedding Techniques for Reversible Watermarking,” ***IEEE Transactions on Image Processing***, Vol. 16, No. 3, pp. 721-730, 2007.