以灰階影像模數函式為基礎之區塊差異值隱藏技術

呂慈純
朝陽科技大學資訊管理系
E-Mail：tclu@cyut.edu.tw

張鈞名
朝陽科技大學資訊管理系
E-Mail：s9714643@cyut.edu.tw

摘要

本論文提出以一個以模數函式為基礎的資訊隱藏技術，我們以灰階影像作為媒介，將機密訊息藏於其中。所提方法首先將影像切割成數個不重疊區塊，區塊大小為2×2，接著找出每個區塊中最小像素值作為基準點，以基準點與區塊內鄰近像素做差異值計算，得到一個新的差異值區塊，利用模數函式將機密訊息藏入至差異區塊中，最後將藏入訊息後之差異值與基準點相加，以求得偽裝像素。本方法每個區塊最多可藏入六個位元，由實驗可知，所提方法可以有效提升資訊藏入量。

關鍵字：資訊隱藏、區塊切割、模數函式

Abstract

This paper proposes an information hiding scheme based on modulus function. We use grayscale image as a cover medium to embed the secret information. The proposed method divides the image into several 2×2 non-overlapping blocks and uses the smallest pixel of each block as a comparison point. Then, we compute the differences between the comparison point and the other pixels of the block. The secret message is concealed in the differences by a modulus function. Next, the scheme adds the comparison point and the differences up to get the stego pixels. The experimental results shows that the proposed method can increase embed capacity.

Keywords: Information hiding, block segmentation, modulus function.

1. 前言

隨著網路科技的進步[5]，使人類與人類之間的距離更為接近，多媒體電子資料的傳遞也越來越便利，但其中所衍生的安全問題是我們需要關注的議題，例如，數位影音媒體的拷貝、國防或商業機密遭駭客竊取等，為了防範盜版、拷貝與資料安全，學者們提出資訊隱藏概念[3]，以確保訊息的安全，其技術在傳送方傳遞資訊之前，會先經由特定演算法將機密訊息藏入一般的數位媒體中產生偽裝媒體。偽裝媒體在經由網路傳遞時，若被駭客竊取也沒關係，因為無法察覺偽裝媒體中有任何異常，進而能避免攻擊，使資訊順利傳送到接收方手中。為了不讓駭客有所察覺，偽裝媒體與隱蔽媒體之間的差異度必須要非常小，以影像媒體來說，若偽裝影像的高峰影像信號雜訊比(Peak Signal of Noise Ration, PSNR)[1]高於 30dB 以上，以人類視覺感官系統上是無法察覺有任何差異的，因此能安全的傳遞到接收方手中。

目前已有相當多的資訊隱藏技術被發展出來，如同位位元取代法[2]、2006 年 Zhang 學者的模數函式藏匿法[6]等。同位位元取代法，會將隱蔽影像切割成數個不重疊的區塊，利用同位位元的特性在每個區塊內藏入一個機密位元，其資訊嵌入量較少，但是相對的影像品質較好。一般而言，資訊嵌入量與影像品質呈現反向關係，即嵌入量越高，影像品質會越差。而學者們探討的就是如何在兩者中取得一個平衡點，使得資訊嵌入量能夠越多，偽裝影像品質也能維持在一定水準。

2006 年 Zhang 等學者利用模數函式提出一

個資訊隱藏方法，該方法可在 n 個像素中藏入 $\lfloor \log_2(2n+1) \rfloor$ 個機密訊息，以 $n=4$ 為例，該方法可在 4 個像素中藏入 $3=\lfloor \log_2(2\times 4+1) \rfloor$ 個機密訊息，資訊負載量為 3/4 bpp(Bit Per Pixel)。Zhang 等學者所提方法產生的偽裝影像品質非常好，但是其最大藏入量僅為 1 bpp，即當 $n=2$ 的情況下，可藏入 2 個位元。為了提昇該方法的嵌入量，本論文提出一個以模數函式為基礎之差異值資訊隱藏方法，用以改善 Zhang 等學者的資訊負載量。

以下我們針對相關文獻進行探討，如模數函式藏匿法[6]。第三章節為所提方法，詳細介紹嵌入流程與取出流程，第四章節為實驗結果，證明本方法在資訊嵌入量與影像品質都有不錯的效果，最後第五章為本論文之結論。

2. 相關文獻探討

本章節將介紹相關的資料隱藏技術，如 Zhang 學者[6]的模數函式藏匿法。

2.1 模數函式藏匿法

2006 年 Zhang 學者提出模數函式藏匿法[6]，該方法與 LSB 匹配法[4]相似，皆是利用一個模數函式將機密訊息嵌入隱蔽影像像素中。Zhang 等學者以 n 個像素為一組進行嵌入，並利用模數函式 F 判斷是否需要修改像素，F 函式定義為

$$F=\left[\sum_{i=1}^{n}(p_i\times i)\right] \bmod (2n+1), \qquad (1)$$

其中 p_i 為第 i 個像素值。n 個像素可藏入的資訊位元數為

$$\lfloor \log_2(2n+1) \rfloor \text{個。} \qquad (2)$$

將二進制機密訊息，以 $\lfloor \log_2(2n+1) \rfloor$ 個位元為單位，轉成 $(2n+1)$ 進制的機密符號 m，接著判斷機密符號 m 是否等於 F 值，若 F 值與 m 相等，則不需要修改像素值；反之，若機密符號 m 與 F 值不相等，就必須修改 n 個像素中的其中一個。Zhang 利用公式(3)求得修改像素位置 s，公式如下：

$$s=(m-F) \bmod (2n+1)。 \qquad (3)$$

若 s 大於 n，則第 $(2n+1)-s$ 個像素值減 1；反之，若 s 小於等於 n，則第 s 個像素值加 1，如此即可得到偽裝像素值 p_i'。

機密訊息取出階段，只需將偽裝像素值以 n 個像素為一組，並將 p_i' 代入模數函式 F 中求得 F 值，求出的 F 值即為 $(2n+1)$ 進制的機密符號 m，將之轉換成二進制，即可得到機密訊息。

Zhang 學者提出的方法利用像素值減 1 或加 1 的進行藏入，因此，如果遇到像素值為 255 或 0 時，可能會有溢位的情況發生。例如，當像素值為 0，而因為嵌入資訊需減 1 時，就會造成下溢的情況。此時，就不減 1，改成加 1 以產生新的像素值；反之，假設像素值為 255，當因嵌入資訊要加 1 時，就會造成上溢，此時，就不加 1，改成減 1 以產生新像素值。新的像素值取代原始像素，再代入嵌入流程中，以取得偽裝像素。若還有溢位情形則此步驟會再重覆執行，直到藏入結果不會有溢位情況時結束。

以下我們舉個例子說明嵌入流程，若以每 4 個像素為一組進行藏入的動作，$n=4$，代入公式 (2) 可得 $\lfloor \log_2(2n+1) \rfloor=\lfloor \log_2(2\times 4+1) \rfloor=3$，得知每 4 個像素可藏入 3 個位元，假設二位元機密訊息為 011100011101，以每 3 個為一組轉換成 $(2n+1)=(2\times 4+1)=9$ 進制得到機密符號 3435。以圖 1(a)為例，首先取前 4 個像素值 5、11、10 和 12，並代入模數函式 F，即公式(1)中，求得 $F=(5\times 1+11\times 2+10\times 3+12\times 4) \bmod (2\times 4+1)=6$，因此第一個機密符號 $m=3$ 不等於 F 值，因此代入公式(3)求得修改像素位置，$s=(3-6) \bmod (2\times 4+1)=6$。由於 $s>n$，必須將第 $(2\times 4+1)-6$ 個像素值減 1，也就是將第三個像素值做減 1 的動作，如圖 1(b)前 4 個像素所示。接著，將皆下來的 4 個像素值 20、31、41 和 51，代入模數函式 F 中，求得

$F=(20\times1+31\times2+41\times3+51\times4)\bmod(2\times4+1)=4$，由於 F 值等於第二個機密符號 $m=4$，因此像素值不做變動。接下來的 4 個像素值 1、0、0 和 1 代入模數函式 F 得到 $F=(1\times1+0\times2+0\times3+1\times4)\bmod(2\times4+1)=5$，由於 F 值不等於 $m=3$，所以代入公式(3)求得修改像素位置 $s=(3-5)\bmod(2\times4+1)=7$。因此 s 大於 n，故將第 $(2\times4+1)-6=2$ 個位置之像素值減 1，但是第二個像素值減 1 會變成負 1，這樣就會造成溢位，所以就不減 1，改成加 1，得到新像素值為 1、1、0、1，再重新代入嵌入演算法中得到 1、1、0、0。以此類推，即可得到最後的偽裝影像圖 1(b)。

取出階段只要將像素以 n 個為一組代入模數函式中，即可取出機密訊息，以圖 1(b)為例，將第一組像素 5、11、9、12 代入模數函式 F 中，求得 $F=(5\times1+11\times2+9\times3+12\times4)\bmod(2\times4+1)=3$，取出機密符號 $m=3$，第二組像素 20、31、41、51 代入模數函式 F 求得 $F=(20\times1+31\times2+41\times3+51\times4)\bmod(2\times4+1)=4$，取出機密符號 $m=4$。

以此類推，取得機密符號 3435，將每個機密符號轉換成二進制位元，即可得到機密訊息 011100011101，完成取出步驟。

5	11	10	12
20	31	41	51
1	0	0	1
0	0	0	0

(a) 隱蔽影像

5	11	9	12
20	31	41	51
1	1	0	0
1	0	0	1

(b) 偽裝影像

圖 1. 藏入方法示意圖

由於 Zhang 等學者的所提方法藏入量有限，因此，本論文將提出一個可提昇藏入量的改進方法。

3. 提出方法

本章節我們介紹所提出方法之機密資訊嵌入流程與機密資訊取出流程，首先我們將一張大小為 $M\times N$ 的影像當作隱蔽影像，並將隱蔽影像切割成大小為 2×2 的數個不重疊區塊，其中 B_j 為第 j 個不重疊區塊，接著找出區塊中最小的像素值做為 x_0，其他像素值以 Z 字型掃瞄得到四個像素分別為 $\{x_0, x_1, x_2, x_3\}$，如圖 2(a)所示。以最小像素值 x_0 為基準點，計算與鄰近像素值之差異值 d_i，再將差異值 d_i 與機密訊息利用模數函式概念計算訊息修改量，以將機密訊息藏入差異值 d_i 中，求到偽裝差異值 d_i'，最後將求出的偽裝差異值 d_i' 加上最小像素值 x_0，再按原來像素的順序即得到偽裝區塊，令 B_j' 為已藏入機密訊息的偽裝區塊，如圖 2(b)所示。詳細藏入流程如下。

x_0	x_1
x_2	x_3

(a)隱蔽影像的區塊 B_j

x_0'	x_1'
x_2'	x_3'

(b)偽裝像素的區塊 B_j'

圖 2 藏入方法示意圖

3.1 嵌入流程

步驟 1. 將隱蔽影像切割成數個大小為 2×2 的區塊，對已切割好的隱蔽區塊 B_j 進行掃瞄，找出最小像素值為 x_0，若有兩個以上相同的最小像素值，會取最先掃瞄到的相同像素值為最小值 x_0，其他以 Z 字型方式掃瞄，得到 $B_j=\{x_0, x_1, x_2, x_3\}$。

步驟 2. 接著計算最小像素值 x_0 與其他鄰近像素值 x_i 之差異值 d_i。其公式如下：

$$d_i=|x_i-x_0|, \quad (4)$$

其中 $i=1,2,3$。

步驟 3. 將差異值 d_i 代入模數函式計算出 F

值。其公式如下：

$$F=\left(\sum_{i=1}^{3} d_i \times i\right) \bmod 2^n, \quad (5)$$

其中 n 為機密資訊藏入位元數。

步驟 4. 將二位元機密訊息以 n 個位元為一組，轉成十進制的機密符號得到 m，接著判斷機密符號 m 與模數函式值 F 是否相同，若為相同這個區塊像素值就不作變動，執行下一個區塊的藏入；反之，若不同則繼續步驟 5。

步驟 5. 機密符號 m 與模數函式值 F 代入公式(6)求出修改量 s 值。

$$s=(m-F) \bmod 2^n。 \quad (6)$$

步驟 6. 將 s 值除六取得商值 q_6 與餘數值 r_6。這個動作主要是為了讓 s 修改量更平均的分配到各個差異值中，有效降低失真以提升影像品質。公式如下：

$$\begin{cases} q_6=\left\lfloor \frac{s}{6} \right\rfloor, \\ r_6=s \bmod 6。 \end{cases} \quad (7)$$

步驟 7. 接著將商值 q_6 與餘數值 r_6 代入公式中，分別求出訊息修改量 q_1、q_2、q_3，公式如下：

$$\begin{cases} q_3=\left\lfloor \frac{r_6}{3} \right\rfloor, \\ r_3=r_6 \bmod 3, \\ q_2=\left\lfloor \frac{r_3}{2} \right\rfloor, \\ r_2=r_3 \bmod 2, \\ q_1=r_2。 \end{cases} \quad (8)$$

步驟 8. 接著將差異值 d_i 個別加上商值 q_6，再依序加上訊息修改量 q_1、q_2、q_3 得到偽裝差異值 d_i'，公式如下：

$$d_i'=d_i+q_i+q_6, \quad (9)$$

其中 i 為 1 到 3。

步驟 9. 將偽裝差異值 d_i' 加上最小像素值 x_0，得到偽裝像素區塊 $\hat{B}_j$。公式如下：

$$\hat{B}_{ji}=x_0+d_i'。 \quad (10)$$

步驟 10. 將 $\hat{B}_j$ 像素值按原本順序排列即可求得偽裝區塊 B_j'。

以下舉一個例子說明，將一張隱蔽影像切割成數個大小為 2×2 的區塊，其中一個區塊如圖 3(a)所示，根據步驟 1 找出最小像素值 x_0，圖 3(a)中最小值為 7，將排序後得到圖 3(b)，接著利用公式(4)計算出最小像素值 x_0 與其他鄰近像素值 x_i 之差異值 d_i，如圖 3(c)。假設我們藏入位元數為 6 個位元數，$n=6$，接著將差異值 d_i 利用公式(5)計算出 F 值，$F=(1\times 33+2\times 14+3\times 3) \bmod 2^6=6$。假設機密訊息為100000將其轉換成十進制機密符號得到 $m=32$，接著判斷機密符號 m 是否與 F 值相等，若不相同則繼續步驟 5.，我們將機密符號 m 與模數函式值 F 代入公式(6)求出 s 值，$s=(32-6) \bmod 2^6=26$。將 s 值代入公式(7)取得商值 $q_6=\left\lfloor \frac{26}{6} \right\rfloor=4$ 與餘數值 $r_6=26 \bmod 6=2$。接著將平均值 q_6 與餘數值 r_6 代入公式(8)中求出訊息修改量 $q_3=\left\lfloor \frac{r_6}{3} \right\rfloor=\left\lfloor \frac{2}{3} \right\rfloor=0$，$r_3=r_6 \bmod 3=2 \bmod 3=2$，$q_2=\left\lfloor \frac{r_3}{2} \right\rfloor=\left\lfloor \frac{2}{2} \right\rfloor=1$，$r_2=r_3 \bmod 2=2 \bmod 2=0$，$q_1=0$。接著將差異值 d_i 個別加上商值 $q_6=4$ 和訊息修改量 q_1、q_2、q_3 得到偽裝差異值 d_i'，如圖 3(e)。最後利用公式(9)將偽裝差異值 d_i' 加上最小值 $x_0=7$，即可得到區塊 $\hat{B}_j$，如圖 3(f)，再按原來的順序排列，即可得到偽裝區塊 B_j'，如圖 3(g)。

40	7
21	10

(a) 隱蔽區塊

7	40
21	10

(b)找出最小值 x_0 的區塊 B_j

x_0	33
14	3

(c) 差異值 d_i

x_0	0
1	0

(d) 訊息修改量 q_i

x_0	37
19	7

(e) 偽裝差異值 d_i'

7	44
26	14

(f) 區塊 $\hat{B}_j$

44	7
26	14

(g) 偽裝區塊 B_j'

圖 3. 所提方法藏入實例

3.2 取出流程

當接收方收到偽裝影像，得知切割區塊大小與藏入位元數即可進行解密的步驟。利用像素值差異與模數函式的概念可將機密訊息取出。取出步驟如下：

步驟 1. 將偽裝影像切割成數個大小為 2×2 的偽裝區塊，從偽裝區塊中找出最小像素值 x_0'，再進行 Z 字形掃瞄，得到區塊 B_j'。

步驟 2. 計算最小像素值 x_0' 與其他鄰近偽裝像素值 x_i' 之差異值 d_i'。公式如下：

$$d_i' = \left| x_i' - x_0' \right|,$$

其中 $i = 1,2,3$。 (11)

步驟 3. 將差異值 d_i' 代入模數函式計算出 F' 值。其公式如下：

$$F' = (\sum_{i=1}^{3} d_i' \times i) \bmod 2^n,$$

其中 n 為機密資訊藏入位元數。 (12)

步驟 4. 求出的 F' 值即為十進制的機密符號，再根據藏入的位元數 n，轉換成 n 個位元的二進制機密訊息。

以下舉例說明取出詳細流程，當接收方收到偽裝影像，根據步驟 1 將影像切割成數個大小為 2×2 的區塊，如圖 4(a)所示，找出最小像素值為 x_0'，以這個例子我們找到像素值 7 為最小值，再以 Z 字形掃瞄得到區塊 B_j'，如圖 4(b)。接著計算最小像素值與其他鄰近偽裝像素值之差異值 d_i'，如圖 4(c)。接收方得知每個區塊會嵌入 6 個位元數，將差異值 d_i' 利用模數函式計算出 F' 值，$F' = (1\times 37 + 2\times 19 + 3\times 7) \bmod 2^6 = 32$，再將 F' 轉換成 6 位元二進制的機密訊息，即可得到機密訊息 100000，完成取出步驟。

44	7
26	14

(a) 偽裝區塊

7	44
26	14

(b) 區塊 B_j'

x_0'	37
19	7

(c) 差異值 d_i'

圖 4.取出流程實例

4. 實驗結果

本論文以七張大小為 512×512 的灰階影像做為實驗影像，分別為 Pepper、Baboon、Boat、Babala、F16、Goldhill 和 Lena，機密訊息為隨機產生的二進制字串。本論文使用影像信號雜訊比(Peak Signal of Noise Ration, PSNR)做為判斷偽裝影像品質之基準，其公式如下：

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right),$$

其中 MSE 為均方差(Mean Squared Error, MSE)，為原始影像與偽裝影像的差異值，公式如下：

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M\times N-1} (p_i - p_i')^2,$$

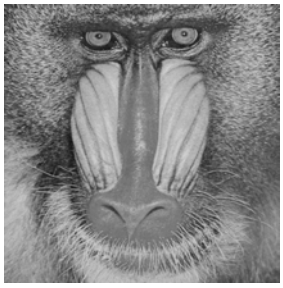
其中 $M \times N$ 為像素值總個數，p_i 為第 i 個原始像素值，p_i' 為第 i 個偽裝像素值。

本實驗針對這七張隱蔽影像進行機密訊息的嵌入動作，將隱蔽影像切割成數個大小為 2×2 的區塊，每個區塊皆可嵌入機密訊息，區塊嵌入位元數為 2、3、4、5、6 個位元，實驗結果如表 1 至表 5 所示。

表 1 顯示，當每個區塊藏入 2 位元時，影像品質 PSNR 平均可達到 55dB 以上，不管是平滑影像或複雜影像嵌入效果都相當好。

本論文所提方法最高可達每個區塊嵌入 6 個位元，圖 5 為每個區塊嵌入 6 位元後的偽裝影像，由此實驗顯示，當達到最高藏量 6 位元時，影像品質 PSNR 還可達到 33dB。一般而言，影像訊號雜訊比 PSNR 值達到 30dB 以上後，人類肉眼可以說是無法察覺出任何異樣的。

(a)Pepper
(PSNR=33.71)



(b)Baboon
(PSNR=33.72)

(c)Boats
(PSNR=33.72)

(d) F16
(PNSR=33.71)

(e)Goldhill
(PNSR=33.74)

(f)Lena
(PNSR=33.70)

圖 5. 偽裝影像(6 位元)

表 1 每個區塊藏 2 位元

測試影像 (512x512)	藏入量 (0.5bbp)	PSNR
Lena	131,072	55.4167
Baboon	131,072	55.3816
Babala	131,072	55.4029
Boats	131,072	55.4019
F16	131,072	55.4105
Goldhill	131,072	55.3911
Peppers	131,072	55.4036

表 2 每個區塊藏 3 位元

測試影像 (512x512)	藏入量 (0.75bbp)	PSNR
Lena	196,608	51.1611
Baboon	196,608	51.1335
Babala	196,608	51.1668
Boats	196,608	51.1312
F16	196,608	51.1249
Goldhill	196,608	51.1395
Peppers	196,608	51.1501

表 3 每個區塊藏 4 位元

測試影像 (512×512)	藏入量 (1bbp)	PSNR
Lena	262,144	45.7930
Baboon	262,144	45.8268
Babala	262,144	45.8173
Boats	262,144	45.8188
F16	262,144	45.8264
Goldhill	262,144	45.8125
Peppers	262,144	45.8286

表 4 每個區塊藏 5 位元

測試影像 (512×512)	藏入量 (1.25bbp)	PSNR
Lena	327,680	39.7704
Baboon	327,680	39.7710
Babala	327,680	39.7905
Boats	327,680	39.7667
F16	327,680	39.7473
Goldhill	327,680	39.7464
Peppers	327,680	39.7717

表 5 每個區塊藏 6 位元

測試影像 (512×512)	藏入量 (1.5bbp)	PSNR
Lena	393,216	33.7038
Baboon	393,216	33.7195
Babala	393,216	33.6991
Boats	393,216	33.7154
F16	393,216	33.7062
Goldhill	393,216	33.7361
Peppers	393,216	33.7129

接著，我們比較本論文所提方法與 Zhang 學者之模數函式藏匿法之最大藏入量，我們以 bpp 代表每個像素中平均嵌入的位元數，其公式如下：

$$bpp = \frac{\text{總資訊藏入量}}{\text{影像大小}} \tag{16}$$

所提方法最大藏入量為 393,216 個位元數，bpp 為 1.5bpp，而 Zhang 學者最大藏入量為 262,144，很明顯地所提方法比 Zhang 學者多出了 131,072 個位元，資訊負載量比 Zhang 學者多出了 0.5bpp。

5. 結論

本論文以模數函式藏匿法為基礎提出一種能達到高藏量的隱藏技術，將一張隱蔽影像切割成數個大小為 2×2 的區塊，針對這些區塊個別進行藏入。實驗採用七張複雜與平滑之隱蔽影像進來藏匿的步驟，由實驗結果顯示在達到最大藏量時，影像訊號雜訊比 PSNR 值還能達到 33dB 以上，在人類視覺感官上保持不錯的視覺效果。

參考文獻

[1] 呂慈純、陸哲明、張真誠，*多媒體安全技術*，***全華圖書股份有限公司***，2007。

[2] Chang, C. C., Hsiao, J. Y. and Chan, C. S., “Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy,” ***Pattern Recognition***, Vol. 36, pp. 1583-1595, 2003.

[3] Fabien, A. P., Anderson, R. J. and Kuhn, M. G., “Information hiding - a survey,” ***Proceeding of the IEEE Special Issue on Protection of Multimedia Content***, Vol. 87, No. 7, pp. 1062-1078, 1999.

[4] Jarno, M., “LSB matching revisited,” ***IEEE Signal Processing Letters***, Vol. 13, No. 5, pp. 285-287, 2006.

[5] Pettey, C., and Stevens, H., “Gartner says 17 countries to surpass 60 percent broadband penetration into the home by 2012,” ***Gartner***

on 2008 Press Releases, 2008.

[6] Zhang, X. and Wang, S. Z., "Efficient steganographic embedding by exploiting modification direction," ***IEEE Communications Letters***, Vol. 10, No. 11, pp. 781-783, 2006.