

一個基於最佳 LSB 替代和禁忌搜尋法的資訊隱藏技術

A New Image Hiding Technique by Optimal LSB Substitution and Tabu Search

蔡正發

屏東科技大學資訊管理所

cftsai@mail.npust.edu.tw

陳政緯

屏東科技大學資訊管理所

M9556003@mail.npust.edu.tw

摘要

本論文提出一個新的資訊隱藏技術稱為 TS-OLSBS，此方法係混合最佳 LSB 替代方法及禁忌搜尋法。最佳 LSB 替代方法可以增強偽裝影像之品質。此外，禁忌尋法亦是一可得到接近最佳解的極佳啟發式演算法。因此，本論文混合最佳 LSB 替代技術與禁忌搜尋法以進行影像隱藏之處理。根據本研究實驗之系統模擬結果顯示，本論文所提出之 TS-OLSBS 資訊隱藏技術在時間代價與訊號雜訊比之比較，確實較最佳 LSB 替代技術及 GA-OLSBS(基因演算法混合最佳 LSB 替代技術)為佳。

關鍵詞：簡單最不重要位元替代、禁忌搜尋法、基因演算法

Abstract

This paper presents a new image hiding technique using Tabu search and optimal Least Significant Bit (LSB) substitution named TS-OLSBS. An optimal LSB substitution may enhance the quality of stego image. In addition, a Tabu search can obtain near optimal solution. Therefore, this paper hybridizes Tabu search and optimal LSB substitution for image hiding

processing. According to our simulation results, it is observed that the proposed method outperforms LSB substitution and genetic algorithm hybridizes OLSBS (GA-OLSBS) in time cost and peak-signal-noise-ratio(PSNR) comparisons.

Key word : LSB, Tabu Algorithm, Genetic Algorithm

一、簡介

隨著電腦科技和網際網路的快速發展，人們可以利用電腦來處理一些多媒體資訊，例如影像(Image)、語音(Audio)與視訊(Video)，藉由網際網路我們可以輕易的傳輸多媒體資訊，縮短人與人之間溝通的橋樑以及資訊傳遞的時間，帶來生活的便利。但是由於每一個人都可以很容易在網際網路中傳輸資訊、溝通與交流，相對的這些資訊可能很容易地會被有心人士監聽、竄改和擷取，因而在網路上傳遞機密資訊將有風險性的存在，如何保護資訊在傳輸過程中的安全性和秘密性頗為重要，資訊隱藏領域已成為目前熱門的研究領域。

在影像資訊隱藏領域中，掩護機密影像的圖片稱為掩護影像(Host Image)，將機

密影像藏入掩護影像後的圖片稱為偽裝影像(Stego Image)，而偽裝影像的品質將是我們研究資訊隱藏的一個重要依據，倘若品質太小，很容易讓竊取者發現偽裝影像被藏入重要的機密資訊，是以如何將偽裝影像的品質維持到可被接受的範圍，已是研究者戮力追求的目標。在影像資訊隱藏的技術上，主要處理影像之方式可分成頻率域和空間域兩類。頻率域技術係指將原始影像做頻率的轉換，再把機密資訊藏入於頻率係數裡，而空間域技術並不必像前者需做任何的轉換，可直接對原圖做處理，將機密資訊藏匿於掩護影像的像素中或者藏匿於壓縮技術產生的壓縮碼中即可。頻率域技術較具有強韌性，因數位浮水印需較高的強韌性，通常數位浮水印採用頻率域技術(陳孝同，2003)(Lin et al., 2000)，而若需較大的藏入容量，則通常在空間域下做處理(Wang et al., 2001)(C. - C. Chang et al., 2003)。本篇論文主要是著重於空間域相關技術的探討，故不對數位浮水印的技術做深入的討論。

空間域的資訊隱藏技術中有一方法稱為簡單最不重要位元(Simple Least Significant Bit, Simple LSB)替代技術，其概念為將機密影像藏入掩護影像各像素值最不重要的位元中，此方法的理論和實作都非常簡單，但是存在一些問題，當藏入 k 位元(k 代表機密影像藏入掩護影像的位元數)數量愈多時，偽裝影像的品質相對的愈差，竊取者很容易可以察覺此圖已被藏入機密資訊，且由於方法簡單，只要取出每個像素值 k 位元數便可獲取到機密資訊，其安全性相對的也比較低。之後便有學者提出最佳最不重要位元技術以改善偽裝影像品質和安全性(Wang et al., 2001)，將機密影像透過替代矩陣轉換成另一個新的機密影像，若要還回原來的

影像也必須透過替代矩陣，如何尋找最佳的替代矩陣，最簡單的方式是將機密影像代入所有可能的替代矩陣，一一去計算已經藏入新機密資訊的偽裝影像和掩護影像的品質。但是此方法會造成時間複雜度的增加，因此有學者提出利用基因演算法來搜尋近似最佳解的替代矩陣(Wang et al., 2001)，而 Chang 等人認為此方法雖可減少時間複雜度，但所尋找的解並不是最佳解。在我們的方法，所尋找的解相較於基因演算法將更趨近於最佳解，並且能夠減少計算的時間。

本論文第二節將列出一些相關文獻，第三節將詳細描述我們的方法，第四節呈現實驗結果，最後第五節則為論文作結論。

二、文獻探討

在這章節中，我們將相關文獻分兩個部分做介紹，在 2.1 節將介紹最不重要位元替代的相關技術，而 2.2 節將介紹啟發式演算法。

2.1 最不重要位元替代的相關技術

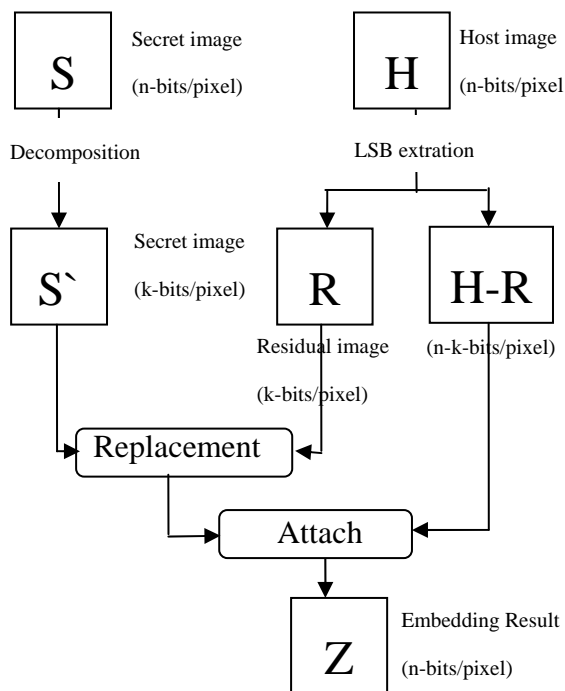
本節將介紹簡單最不重要位元替代技術和最佳最不重要位元替代技術兩種方法。

2.1.1 簡單最不重要位元替代技術

簡單最不重要位元替代技術是最早被提出且最容易實作的資訊隱藏技術，其概念為在空間域下將機密影像藏入到掩護影像最不重要位元中，而此位元的改變對於像素值的影響不大，藉由此特性進而達到資訊隱藏的目的。

假設我們的機密影像為 S ，掩護影像為 H ，此方法主要分為兩個步驟，第一步驟為：機密影像是由 8-bit 像素值所組成，必須先將機密影像分解為每一個像素值的單位為 k -bit，由 k -bit 像素值所組成的機密影像稱為 S' ， S' 影像大小將會相

等於掩護影像 H 的大小。第二步驟為：影像 R 係由掩護影像 H 的每一個像素值萃取最右邊 k -bit 為單位的像素值所組成，將機密影像 S' 每一個像素值依序取代影像 R 相對應的位置，最後將影像 R 的每一個像素值依序與影像 H-R 相對應位置的像素值進行相加，便可得到偽裝影像 Z。圖一呈現出簡單最不重要位元替代技術的藏入流程：



圖一：簡單最不重要位元替代的藏入流程圖。(本論文為自行整理)

取出機密影像的方法也十分簡單，只要依序取出偽裝影像 Z 每一個像素值最右邊 k -bit，再將由 k -bit 像素值所組成的機密影像 S' 轉換成 n -bit 為單位的像素值，即可取得原來的機密影像 S。此方法理論和實作都非常的簡單，但其缺點有以下數點：第一，當 k 愈大時，偽裝影像品質會愈低，使得竊取者很容易可以察覺此偽裝影像已被藏入重要資訊。第二，由於

此方法理論非常簡單，當竊取者發現此影像已被藏入重要資訊，便可很容易的取出，因此便有安全性的問題。後續幾年有學者提出一些方法來改進簡單最不重要位元替代技術。

2.1.2 最佳最不重要位元替代技術

Wang 等人在 2001 年提出最佳最不important位元替代技術(Wang et al., 2001)，此方法主要是改善簡單最不important位元替代技術的方法，以提升偽裝影像的品質和安全性。此技術主要增加了兩個步驟，第一步驟，在取代影像 R 之前，利用 Wang 論文中提到公式(1)一一映射函數或雙射函數(bijective mapping function)將 S' 每一個像素值的位置 x 隨機轉換到新的位置 $f(x)$ ，其中 s 為機密影像的大小，例如：機密影像大小為 $512*512$ ，如以下公式(1)所示。 k_0 和 k_1 為兩把金鑰，其限制式為 k_0 和 k_1 的最大公因數需為 1，當竊取者擷取偽裝影像，若無 k_0 和 k_1 這兩把金鑰，則無法還原原始的機密影像，藉由一一映射函數達到資料安全性，經過轉換後的機密影像稱為 S''。

$$f(x) = (k_0 + k_1 * x) \bmod s \quad (1)$$

第二步驟為：利用替代矩陣將 S'' 像素值替代另外一個數值，進而轉換成新的機密影像，其轉換的目地在於提升偽裝影像的品質。替代矩陣定義為 $A_{n*n} = \{a[i][j] | 0 \leq i, j \leq 2^k - 1\}$ ， n 的大小等於 2^k ，當 k 值為 2 時，矩陣大小為 $4*4$ ， $a[i][j]$ 的數值為 1 或 0，當數值為 1 時，代表在 S'' 影像裡的像素值為 i 值將會被 j 值所取代；當數值為 0 時，則不需要做任何的動作。而此替代矩陣有一限制，即矩陣中的每一行及每一列只能出現一次數值 1，我

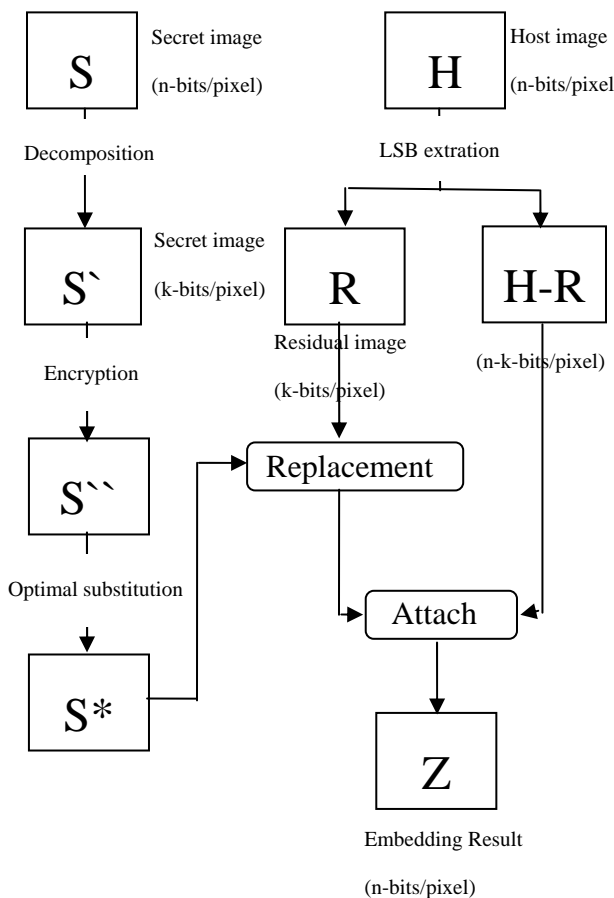
們以一個簡單的例子來說明：

$$S'' = \begin{bmatrix} 01 & 11 \\ 00 & 10 \end{bmatrix}_2 = \begin{bmatrix} 1 & 3 \\ 0 & 2 \end{bmatrix}_{10}$$

$$A_{4*4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$S^* = \begin{bmatrix} 11 & 10 \\ 00 & 01 \end{bmatrix}_2 = \begin{bmatrix} 3 & 2 \\ 0 & 1 \end{bmatrix}_{10}$$

機密影像 S' 經過替代矩陣轉換後的影像稱為 S^* 。圖二呈現最佳最不重要位元替代技術的藏入流程圖：



圖二：最佳最不重要位元替代技術的藏入流程圖。(本論文為自行整理)

此一技術存在一個問題，即：我們如何尋找最佳的替代矩陣呢？替代矩陣總共有 $(2^k)!$ 的可能解，隨著 k 值的增加，替代矩陣的可能解也相對地增加，若需要將全部可能解計算偽裝影像的品質，將會造成時間複雜度的增加。針對此問題，Wang 等人便提出了利用基因演算法來尋找近似最佳替代矩陣，以減少計算時間。

2.1 啟發式演算法

啟發式演算法常使用在解決組合最佳化的問題，已有不錯的成果。常見的啟發式演算法有基因搜尋法演算法、模擬退火法和禁忌搜尋法等。下一節將詳細介紹禁忌搜尋法。

2.2.1 禁忌搜尋法

禁忌搜尋法(Tabu Search)是由 Fred Glover(Glover, 1980)(Glover, 1990) 在 1977 年所提出一種啟發式方法，主要是用來解決組合最佳化的問題。此方法與其它啟發式演算法不同之處在於其具有記憶能力，通常像基因演算法、模擬退火法之類不具有任何記憶的演算法，常常會重複搜尋已經檢視過的可行解或是在同一區域搜尋時間過長，因此禁忌搜尋法利用此特性，使演算法能夠快速跳脫區域解，提升求解效率。

以下針對禁忌搜尋法的組成元件提出說明：

1. 移步

又稱移動路徑，指在所有鄰近解中，搜尋比目前解還要更好的解作為改善的移動路徑，使其向最佳解趨近。

2. 禁忌名單

記錄過去搜尋中每次移步的屬性。而在禁忌名單中的屬性，在未解禁之前，則不能被選取，以避免搜尋的重複。一般而言，當禁忌名單愈大時，陷入區域解的可能性

愈低，但會增加求解的速度；相反地，禁忌名單愈小時，容易陷入區域解。

一般而言，禁忌名單大小沒有一定的限制，通常根據問題特性來加以決定。

3. 免禁準則

用來釋放被列為禁忌名單的移步屬性。

4. 搜尋結束準則

用來停止搜尋的條件，常見的有四種：

(1)達到預設的最大迭代次數為止(2)達到cpu 最大允許處理時間(3) 達到預設目標函數持續一段時間未改善的最大迭代(4)達到預設可以接受的目標數值。

禁忌搜尋法的演算法步驟如下：

步驟一：讀取問題資料，設定參數包含有禁忌名單大小、免禁條件、搜尋結束條件等等，並計算起始解，令計數器 = 0，禁忌名單= Φ 。

步驟二：搜尋目標值最佳的解，此可行解不能被列為禁忌名單或是此解滿足免禁條件。

步驟三：更新禁忌名單和記錄目前搜尋最佳解，並且進行移步。

步驟四：檢查搜尋終止條件是否成立，如果成立的話，輸出最佳解，否則更新計數器後回到步驟二繼續執行。

三、研究方法

在這個章節，我們將詳細介紹本論文所提出之利用禁忌搜尋法來尋找最佳的替代矩陣。以下將分兩個部分進行說明，第一部分主要是說明將機密影像藏入掩護影像的流程，第二部分則是說明如何將機密影像從偽裝影像萃取的流程。

3.1 藏入流程

步驟一：掩護影像 H 分成影像 R 和影像 H-R。影像 R 由掩護影像 H 的每一

個像素值萃取最右邊 k -bit 為單位的像素值所組成。影像 H-R 由掩護影像 H 的每一個像素值扣除最右邊 k -bit 為單位的像素值所組成。

步驟二：機密影像是由 8-bit 像素值所組成，必須先將機密影像分解為每一個像素值的單位為 k -bit (k 指的是機密影像藏入掩護影像的位元數)，由 k -bit 像素值所組成的機密影像稱為 S' ， S' 影像大小將會相等於掩護影像 H 的大小。

步驟三：將 S' 每一個像素值的位置利用一一映射函數或雙射函數 (bijective mapping function) 隨機轉換到新的位置。經過轉換後的機密影像稱為 S'' 。

步驟四：利用替代矩陣將 S'' 的像素值作轉換，經過轉換後的機密影像稱為 S^* 。我們利用禁忌搜尋法找尋最佳化的替代矩陣，首先要尋找最佳的替代矩陣之前，需將替代矩陣轉換成我們所提出方法可使用的矩陣，如同 Wang 把替代矩陣轉換成基因演算法中的染色體一樣。假設替代矩陣為 A_{n*n} ，轉換過後的矩陣 $R_n = \{(r[i]) | 0 \leq i, r[i] \leq 2^k-1\}$ ， n 為 2^k ， $r[i]$ 為記錄 A_{n*n} 中出現 1 的位置，以下將以一個簡單的例子說明：

$$A_{4*4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$R_4 = \begin{bmatrix} 0 \\ 3 \\ 1 \\ 2 \end{bmatrix}$$

從這個例子中，可發現 R_4 裡的值是不會重複的，當機密影像為 i 值將會被 $R[i]$ 所取代，例如 $R[1] = 3$ ，代表在機密影像為 1 的數值將會被 3 取代。接下來將利用矩陣 R 作為使用禁忌搜尋法的初始資料。

以下為利用禁忌搜尋法來尋找最佳的替代矩陣的詳細步驟：

- 1、以隨機的方式產生一條初始解，並設定此解為目前最佳解。
- 2、設定禁忌名單大小和免禁規則，定義禁忌名單為 $Tn \times S = \{t[i][j] | 0 \leq i \leq 2^k - 1, 0 \leq j \leq 2\}$ ， n 的大小為 2^k ， s 的大小為 2，主要是記錄移步屬性。免禁規則為在禁忌名單內的移步屬性將被限制 2^k 的時間。
- 3、利用兩點交換方法搜尋鄰近解，此兩點為移步屬性，須符合評估是否為非列為禁忌名單內或是為滿足免禁條件，如果成立的話，則執行步驟 4，否則，繼續尋找鄰近解。
- 4、進行移步並更新禁忌名單，將在步驟 3 所搜尋的移步屬性加入禁忌名單中，以及找尋禁忌名單中是否有滿足免禁條件，若有符合，則將此移步屬性從禁忌名單中移除。
- 5、判別鄰近解是否優於目前搜尋最佳解，若成立的話，則取代最佳解。鄰近解的求法是依據計算偽裝影像和掩護影像的 PSNR 值而定，PSNR 值愈高代表

品質愈好，但是計算 PSNR 值會造成增加時間複雜度，因此為了減少計算時間，將計算平方誤差總和作為衡量品質的好壞，平方誤差為 $(s(i, j) - h(i, j))^2$ ，其中 s 為機密影像， h 為掩護影像。誤差總和愈低，代表影像品質愈好。

- 6、檢查是否有達到終止條件，若成立，則停止搜尋，並輸出最佳解的 PSNR 值，否則，返回步驟 3 繼續執行。

步驟五：將機密影像 S^* 的像素值，依序取代影像 R 的像素值。

步驟六：影像 $H - R$ 的每一個像素值需加回取代後影像 r 相對應位置的像素值。便可得到偽裝影像 Z 。

步驟七：將替代矩陣將藏入偽裝影像 Z 的 n 個像素值 $(k+1)$ 位元中。

3.2 萃取流程

步驟一：從偽裝影像 Z 的每一個像素值中取出 k 位元，便可得到機密影像 S^* 。

步驟二：從偽裝影像 Z 的 n 個像素值取出 $(k+1)$ 位元，便可得到替代矩陣。

步驟三：機密影像 S^* 利用替代矩陣將像素值作轉換，便可得到機密影像 S' 。

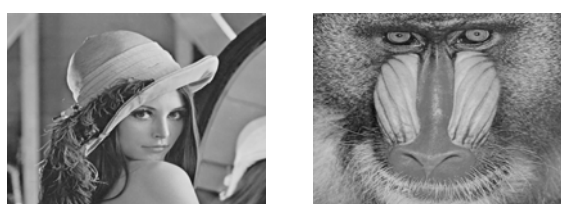
步驟四：利用 k_0 和 k_1 兩把金鑰和一一映射函數，將打散後的機密影像 S' 還原成機密影像 S 。

步驟五：機密影像 S' 是由 k -bit 像素值所組成，需將轉換成每一個像素值由 8-bit 為單位，便可得到最原始的機密影像 S 。

四、實驗結果

本研究使用 C++ 程式語言來實作三種方法，分別為 TS-OLSBS、GA-OSBS (Genetic

algorithm-optimal LSB substitution 和 Optimal LSB substitution。系統配備以個人電腦 CPU 3.00GHz，記憶體為 1GB，作業系統為 Windows XP。實驗測試的圖片以 256 色灰階圖為主，採用圖片大小為 512*512 的 Lena、Baboon、和 Pepper 三張圖片作為掩護影像，如圖三所示，另採用 Tiffany、Jet、和 Boat 三張圖片作為機密影像，如圖四所示，所有機密影像各分別有 256*256、384*256、和 512*256 不同大小的圖片。



(a) Lena (b) Baboon



(c) Pepper

圖三：掩護影像測試圖片。



(a) Tiffany (b) Jet



(c) Boat

圖四：機密影像測試圖片。

影像品質的好壞是以 peak-signal-noise-ratio(PSNR)作為依據，當 PSNR 值愈高時，代表影像品質愈好，反之，PSNR 值愈低時，代表影像品質愈差。

在第二節文獻中曾提到，在機密影像藏入掩護影像之前，利用一一映射函數或雙射函數隨機打散機密影像，在此我們使用和 Wang 相同的數據。

$$f(x) = (1019 + 17 * x) \bmod 262114$$

在基因演算法設定參數的部分，此方法執行時有三個重要的步驟：(1)選擇(2)交配(3)突變，最初的染色體以隨機的方式選擇十條，交配為兩條染色體各自一半的基因作交換，突變率為 0.1，適應函式為平方誤差總和 $(s(i, j) - h(i, j))^2$ ，下世代的選擇則是以適應值最低的十條染色體，迭代次數為 8 次，輸出最好的 PSNR 值。在禁忌演算法設定參數的部分，當 k 為 2 時，禁忌名單大小為 4，免禁規則為迭代 4 次的時間；當 k 為 3 時，禁忌名單大小為 8，免禁規則為迭代 8 次的時間；當 k 為 4 時，禁忌名單大小為 16，免禁規則為迭代 16 次的時間；最佳解選擇為平方誤差總和最低的數值，迭代次數為 50 次，輸出最好的 PSNR 值。

在 Table 1-3 可以看到這三個方法所求得的品質，Optimal LSB substitution 的方法主要是將替代矩陣所有可能解一個一個作比對，因此求出的解便是最佳解，將此當作是衡量的依據。當 $k=2$ 時，GA 和 Tabu 都可以求得最佳解；當 $k=3$ 時，可以看出 GA 所求得解趨近於最佳解，相反的，Tabu 還是能夠求得最佳解；當 $k=4$ 時，由於 Optimal LSB substitution 時間複雜度太高，導致無法求得最佳解，而 Tabu 搜尋到的解比 GA 的方法更加趨近於最佳解。

在 Table 4-6 呈現出這三個方法執行的時間，當 $k=2$ 時，Optimal LSB substitution 的時間複雜度比其他兩個

方法低；當 $k=3$ 時，替代矩陣的數量增加，Optimal LSB substitution 的時間複雜度大幅度的增加；當 $k=4$ 時 Optimal LSB substitution 計算時間太長而導致在表格中無法填入數值。Tabu 與 GA 這兩個方法由於每一次計算迭代的次數是固定的，因此藏入容量多寡並不會造成時間複雜度太大的差異，而我們可以看出 Tabu 比 GA 的方法可有效的減少計算的時間。最後藏入機密資訊的偽裝影像呈現於圖五中。



圖五：利用 TS-OLSBS 和 GA-OLSBS 方法的藏入結果比較，藏入機密影像為 boat，大小為 512×256 。圖(a)、(b) TS-OLSBS 方法的藏入結果，圖(c)、(d) GA-OLSBS 方法的藏入結果，

五、結論

Wang 等人所提出的最佳最不重要位元替代技術在尋找替代矩陣時，需將所有可能性的替代矩陣計算過一次，因此會造成計算時間太長，而利用基因演算法來尋找最佳的替代矩陣，雖可改善的時間複雜度，但所搜尋的解是近似於最佳解，而非最佳解。本論文中，我們提出以禁忌搜尋

法來尋求最佳的替代矩陣，所搜尋的解相較於基因演算法更趨近於最佳解，並且計算時間也相對的比較快，可改善最佳最不重要位元替代技術的缺失。

六、誌謝

感謝行政院國科委員會提供經費補助本計畫之執行，計畫編號：NSC 95-2221-E-020-036

七、參考文獻

- [1] 陳孝同、張真誠、黃國峰，數位影像處理技術，旗標出版社，2003。
- [2] C. C. Chang, J. Y. Hsiao, and C. S.Chen, "Fiding optimal least-significant-bit substitution in image hiding by dynamic programming strategy," Pattern Recognition, Vol. 36, pp. 1583-1595, 2003.
- [3] F. Glover, "Tabu search: part I," ORSA Journal on Computing, Vol. 1, pp. 190-206, 1989.
- [4] F. Glover, "Tabu search: part II," ORSA Journal on Computing, Vol. 2, pp. 4-32, 1990.
- [5] F. Glover, "Tabu search: a tutorial," Interfaces, Vol. 20, pp. 74-94, 1990.
- [6] R. Z. Wang, C. F. Lin, and J. C. Lin, "Image Hiding by Optimal LSB Substitution and Genetic Algorithm," Pattern Recognition, Vol. 34, No. 3, pp. 671-683, 2001.
- [7] S. D. Lin, C.F. Chen, "A robust DCT-based watermarking for copyright protection," IEEE Trans. Consumer Electron. Vol. 3, pp.415-421, 2000.

表 1：掩護影像 Lina 和偽裝影像的 PSNR 值比較

Method	Tiff	Jet	Boat
The secret images are 256*256			
TS-OLSBS	44.60469	44.37816	44.56313
Optimal LSB substitution	44.60469	44.37816	44.56313
GA-OLSBS	44.60469	44.37816	44.56313
The secret images are 384*256			
TS-OLSBS	38.48965	38.30553	38.2475
Optimal LSB substitution	38.48965	38.30553	38.2475
GA-OLSBS	38.48781	38.30484	38.24265
The secret images are 512*256			
TS-OLSBS	32.90754	32.82579	32.57973
Optimal LSB substitution	not available	not available	not available
GA-OLSBS	32.82537	32.74213	32.44549

表 2：掩護影像 Baboon 和偽裝影像的 PSNR 值比較：

Method	Tiff	Jet	Boat
(a)The secret images are 256*256			
TS-OLSBS	44.56698	44.35268	44.55709
Optimal LSB substitution	44.56698	44.35268	44.55709
GA-OLSBS	44.56698	44.35268	44.55709
(b)The secret images are 384*256			
TS-OLSBS	40.54887	40.25349	40.16199
Optimal LSB substitution	40.54887	40.25349	40.16199
GA-OLSBS	40.54384	40.21602	40.1461
(c)The secret images are 512*256			
TS-OLSBS	33.63184	33.53917	33.23065
Optimal LSB substitution	not available	not available	not available
GA-OLSBS	33.53707	33.40052	33.12001

表 3：掩護影像 Pepper 和偽裝影像的 PSNR 值比較：

Method	Tiff	Jet	Boat
(a)The secret images are 256*256			
TS-OLSBS	44.87434	44.63591	44.87637
Optimal LSB substitution	44.87434	44.63591	44.87637
GA-OLSBS	44.87434	44.63591	44.87637
(b)The secret images are 384*256			
TS-OLSBS	39.85097	39.56299	39.46939
Optimal LSB substitution	39.85097	39.56299	39.46939
GA-OLSBS	39.76041	39.54338	39.41386
(c)The secret images are 512*256			
TS-OLSBS	33.49344	33.40632	33.1264
Optimal LSB substitution	not available	not available	not available
GA-OLSBS	33.32578	33.19368	32.96297

表 4：掩護影像為 Lina 的計算時間：

Method	Tiff	Jet	Boat
(a)The secret images are 256*256			
TS-OLSBS	2.8981	2.89558	2.89094
Optimal LSB substitution	1.375	1.359	1.375
GA-OLSBS	9.65722	9.60346	9.64652
(b)The secret images are 384*256			
TS-OLSBS	2.97402	2.97156	2.96622
Optimal LSB substitution	1956.39	1956.703	1957.312
GA-OLSBS	9.71938	9.71808	9.74372
(c)The secret images are 512*256			
TS-OLSBS	2.99966	3.0053	3.0006
Optimal LSB substitution	too huge	too huge	too huge
GA-OLSBS	9.69216	9.69442	9.7075

表 5：掩護影像為 Baboon 的計算時間：

Method	Tiff	Jet	Boat
(a)The secret images are 256*256			
TS-OLSBS	2.92284	2.92028	2.88406
Optimal LSB substitution	1.375	1.375	1.391
GA-OLSBS	9.72184	9.6528	9.94272
(b)The secret images are 384*256			
TS-OLSBS	2.98938	2.98806	2.98564
Optimal LSB substitution	1970.656	1971.266	1970.531
GA-OLSBS	9.76868	9.78278	9.79128
(c)The secret images are 512*256			
TS-OLSBS	3.00816	3.00974	3.07282
Optimal LSB substitution	too huge	too huge	too huge
GA-OLSBS	9.7372	9.72158	9.7454

表 6：掩護影像為 Pepper 的計算時間：

Method	Tiff	Jet	Boat
(a)The secret images are 256*256			
TS-OLSBS	2.891	2.89	2.875
Optimal LSB substitution	1.375	1.36	1.359
GA-OLSBS	9.562	9.578	9.657
(b)The secret images are 384*256			
TS-OLSBS	2.985	2.953	2.953
Optimal LSB substitution	1949.734	1949.578	1950.235
GA-OLSBS	9.703	9.688	9.766
(c)The secret images are 512*256			
TS-OLSBS	3.062	2.984	2.984
Optimal LSB substitution	too huge	too huge	too huge
GA-OLSBS	9.703	9.672	9.672