網路傳輸之視覺偽裝機制研究

王旭正* 陳宏和 林庭宇
*Department of Information Management
Central Police University, Taoyuan, Taiwan 333
*Email: sjwang@mail.cpu.edu.tw

Abstract

網路時代之下，人際間的通訊管道逐漸轉移到成本低廉且使用方便的網際網路架構。網際網路扶搖直上成為當今多數人使用的通訊管道，將網際網路的附加價值發揮的淋漓盡致，另一方面也彰顯出秘密資訊傳送越來越頻繁和重要。如何完成資訊交換安全並且成功傳遞的議題，是很多學者致力研究的方向。藉此，本文提出一種改良的作法，把文字型的秘密訊息以資訊隱藏的作法來處理並加以改良。其作法是先產生一組隨機的二維座標位置(X，Y)，接下來將欲隱藏的文字資料依序轉換成國際標準碼(Unicode)編碼，接著將編碼透過文字秘密分享機制產生 *n* 份子訊息，並且從隨機產生的二維座標位置開始依序藏入 *n* 張 Cover-Image 之第七層位元平面中。隨機的二維座標位置(X，Y)亦經過視覺安全技術處理產生 *n* 張 Shares，並取代 *n* 張 Cover-Image 之第八層位元平面，最後合併所有位元平面層。如此處理過後之影像在其傳輸的過程中，完全是可見的有意義的影像，不但不容易引起不肖企圖者的懷疑，可確保資料機密傳輸安全。相對於先前相關之技術研究，本文同時運用文字/視覺秘密分享機制與資訊隱藏技術，提供了更佳的應用。

Keywords：*Steganography*、*Information Hiding*、*Unicode*、*Cover Image*、*Bit-plane*、*Visual Security*

第一節　簡介

密碼學(Cryptography)曾經是使用者保障私人訊息的利器，運用複雜的數學函式打散原有的資訊使之成為一串串毫無意義的亂碼，傳給特定的人，使得沒有經過授權的人不能夠去了實解原始文件的內容。但是這樣一來，或許有達到資料保護的效果，卻更容易引起不法意圖者的懷疑，而將這一串亂碼全部截取甚至加以破壞。反而使得合法人員或是原本交換訊息的雙方不能夠取得正確的資料，而失去保護數位資料以達秘密交換的原先用意。再者，電子晶片的生產技術不斷突破，電腦的運算能力大步地向上提升，量子電腦(Quantum Computer)已經不再是夢想，有朝一日恐怕再複雜的密碼學加密方式都難不倒電腦，傳統式密碼學的優勢或許無法維持。

有鑑於此，隱藏技術逐漸受到青睞與重視，專家學者熱烈投入隱藏技術的研究[3, 7, 10, 12, 16-19]。相較於密碼學利用加密演算法打散原有資訊內容，目的在於保護資訊；隱藏技術將資訊隱藏在數位媒體之中，在不影響數位媒體品質原則下讓使用者達到秘密通訊/隱藏特殊訊息的目的。如今，隱藏技術的發展已經有相當多數的軟體問世，使資訊多了一個比較有保障的傳遞管道。

若直接使用密碼學加密技術，對數位秘密資料加以處理的方式來達到訊息秘密交換、傳送之目的，會存在如上述亂碼型式傳送易被不法意圖者截取，為了在數位化時代中對這些大量數位化的資料之秘密傳送及交換訊息上有一安全的機制可供參考，本文基於 Steganography 的概念提出一個不同的做法，同時利用文字/視覺式秘密分享機制，對文字與數位影像的秘密資料先加以處理，在耗費少量電腦運算資源的條件下完成資訊隱藏的動作。與文獻資料[7, 16, 18, 19]所提出之方法相較之下，本文所提出之方法不再使用碼簿(Codebook)、索引表(Index Table)、隨機亂數產生器以及密碼學(Cryptography)加密機制等技術，因為上述四項技術一旦要處理大量的加密資料時，將造成電腦運算資源大量的負荷。為了改善這個問題，本文採用視覺安全[13]技術，因為視覺安全技術能同時保有資訊之安全性與少量耗用電腦運算資源兩項優點，將其運用在本文所提出之方法中，使資訊隱藏過程僅需要少量之電腦運算資源，而且仍可確保資訊之安全性，實為整個資訊隱藏過程之關鍵要素。

在 Steganography 技術的部份，本文選擇將訊息隱藏在影像圖 LSB(Least Significant Bit)之資訊隱藏方法，藉此把資訊隱藏於任意選取的掩護影像中，因為改變影像圖 LSB 之後，人類的肉眼並無法辨識如此細微之變化，而達到隱藏安全之效果。如此一來，不但可使傳送過程透明化，使駭客不易對其產生懷疑而加以截取、破壞；也使得處理數位資料能夠多元化，不再只是針對文字型式資料機密來進行處理和傳送。

在本文的章節編排方面，說明如下：第二節中，回顧與資訊隱藏主題有關的幾項研究以及在本文中所採用的視覺安全技術。接著在第三節，提出本文所使用的方法；在第四節以一個實際範例來說明；第五節我們討論相關的研究與比較執行效果；最後於第六節提出我們的結論。

第二節 相關文獻回顧

2.1 Wang & Lu's scheme

Wang & Lu[16]利用加密過的秘密訊息數字索引檔案，轉成二進制碼的型式。並且應用資訊隱藏技術，將其簡單二進制碼的 0 與 1 型式，隱藏入我們所傳輸的 Cover-image 中。如此一來，所有在網路傳輸的文字或影像，都是可見的 Plaintext，即不易遭到駭客的注意與刻意攻擊，因此可以避免不必要的截取和破壞。在[16]中的訊息隱藏機制演算法如下：

Input：
(1)一張具有圖片及其雙語說明文字的資料檔。
(2)欲傳送的秘密中英文訊息。
(3)對稱式的加密金鑰。

Output：
隱藏秘密中英文訊息代碼資料的圖片及其雙語說明文字檔。

執行步驟如下：

Step 1：將欲傳送的秘密訊息之中文字以Big-5碼的型式表示。
Step 2：將傳送圖片之中文說明文字轉成Big-5碼。
Step 3：檢查中文說明文字之 Big-5 碼是否包含秘密中文訊息之 Big-5 碼。
Step 4：檢查英文說明文字是否包含秘密英文訊息之英文字元。
Step 5：根據中文說明文字之 Big-5 碼建立十六進位索引表格(Hexadecimal Code Index Table, HCIT)。
Step 6：根據英文說明文字建立英文字元對應表格(English Character's Position Table, ECPT)。
Step 7：根據 HCIT 比對秘密中文訊息，並隨機選取一數字代表，以建立秘密中文訊息之數字索引檔案(Chinese Index File, CIF)。
Step 8：根據 ECPT 比對秘密英文訊息，並隨機選取一數字代表，以建立秘密英文訊息之數字索引檔案(English Index File, EIF)。
Step 9：將 Step7 與 Step8 所產生之 CIF 與 EIF 分別

進行 IDEA 加密。

Step 10：將 IDEA 加密過的 CIF 與 EIF 轉成二進位碼的形式。

Step 11：將二進位碼隱藏入灰階圖中。

在隱藏訊息的取得方面，說明如下：

接收端收到灰階圖後，首先從灰階圖中取出代表中文訊息之 CIF 與 EIF 的二進位碼，並且進行解密動作；根據中/英文說明文字分別建立 HCIT 以及 ECPT，比對 CIF 與 HCIT，還原中文秘密訊息；比對 EIF 與 ECPT，還原英文秘密訊息。

2. 2 Hou & Tu's scheme

Hou & Tu[7]利用主要利用虛擬隨機函式，將秘密訊息中每一個內碼計算出差值，並建立成差值表。接著將差值表利用秘密分享機制分成子訊息，並且應用資訊隱藏技術，將其簡單二進制碼的 0 與 1 型式，隱藏入我們所傳輸的 Cover-image 中。在[7]中的訊息隱藏機制演算法如下：

Input：

(1)欺敵訊息。

(2)秘密訊息。

(3)*n*張灰階影像。

(4)虛擬隨機重排的seed值。

Output：

(1)*n*張Stego-image

(2)*n*個鍵值$seed_1$~$seed_n$

執行步驟如下：

Step 1：將欺敵訊息和秘密訊息分別轉換成十六進位的Big-5碼，並計算秘密訊息的長度*p*。

Step 2：針對秘密訊息中每一個Big-5碼，計算下列的差值：$diff(i)=P_i-H_{per(i)}$，其中P_i 代表位置索引值為 *i* 的內碼，per為一個虛擬隨機函數，目的是將位置索引*i*對映至另一個位置索引per(*i*)。

Step 3：將所有差值轉換成 5 個位元為單位的二進位值，共為 *5p* 位元的長度。

Step 4：將所有二進位差值依(*k*，*n*)-threshold 秘密分享機制分解成 *n* 份子訊息，其位元長度為 *5mp*。

Step 5：將 seed 值轉換成二進位並且依(*k*，*n*)-threshold 秘密分享機制分解成 *n* 個十進位的值，即為 $seed_i$，*i*=1, 2, …, *n*。

Step 6：分享訊息的長度儲存在 Cover-image 最低位元平面的前 32 個位元，並且將 *n* 份子訊息分別隱藏至 *n* 張 Cover-image 中。

在隱藏訊息內更改之部位說明如下：

接收端須先從*n*張Cover-image中取出*n*個秘密分享訊息並還原成二進位差值表與*n*個seed值，並利用$diff(i)=P_i-H_{per(i)}$函數代入*n*個seed值依序還原Big-5碼，最後得到秘密訊息。

綜觀上述兩個資訊隱藏之方法，我們不難發現兩者不論加密或解密的過程中必須要建立對照表、差值表等，並且在過程中使用到加密技術或虛擬隨機函數，整個過程顯得相當繁雜。由此可見當使用電腦進行此兩種資訊隱藏技術時，得耗費相當大量的電腦運算資源。而本文基於上述兩者方法之缺點，提出一個步驟簡單、節省電腦運算資源之方法並且保有高安全性的方法，與上述兩個方法比較之下明顯地改善了系統安全的便利性與計算資源。

第三節 視覺偽裝機制

比較文獻資料[7,16]所提出的兩個方法，兩者有兩個共通點，其一：將處理過的資訊隱藏到灰階圖中；其二：兩者對於中文秘密訊息的處理方式是採用 Big-5 編碼技術，先將中文秘密訊息轉換成 Big-5，再進行各自的演算法。

文獻[7,16]所提出的方法，[16]最後會將秘密訊息的數字索引檔案加密處理，形成秘文的亂碼型式傳送給接收者，並且伴隨中英文的說明文字；[7]則是需要使用虛擬隨機函數來產生差值，再進行秘密分享機制使的多個使用者同時掌握秘密資訊之一部分，而還原訊息所需要的虛擬隨機函數關鍵 seed 值則以二進位碼之方式隱藏在影像圖中。

使用加密技術以及虛擬隨機函數這兩種方法，在面對處理少量資訊時或許不會造成電腦運算太大的負擔，一旦處理到大量的加密資訊時，將造成電腦運算資源大量的耗費以及操作者必須花費等待時間。因此在本文中我們提出一種改良的作法，本方法使用一組隨機二維座標位置當作加/解密演算法之關鍵點，關鍵的隨機二維座標位置值代表秘密資訊隱藏的起始位置，為了方便使用者操作過程中不須額外記憶任何數字以及增加操作的方便，我們將隨機二維座標位置值利用圖像來呈現，加密過程中則將其利用視覺安全技術分散成 Shares 並取代各 Cover-image 之 LSB。

另外，為了避免將所有秘密資訊集中儲存於單張影像圖中，我們同時採用(*k, n*)-threshold 文字以及視覺秘密分享機制，將秘密訊息以及關鍵二維座標位置值分解成 *n* 份子訊息與 *n* 份 Shares，欲得知秘密訊息，必須結合門檻值 *k* 張以上之子訊息才能將秘密訊息還原，如此一來可以提高安全信任，降低訊息遭到破解的機會。

最後，為了改進 Big-5 編碼所存在的缺點，包括：缺乏許多不常見的字元、僅包含部份繁體中文字體等，我們採用統一碼(Unicode)編碼，增加操作者能使用之繁體中文字體，而且內容亦不再侷限於繁體中文字體，Unicode 已經支援中文的簡繁體、日文、韓文、西歐各種字母、西藏文、阿拉伯文、緬甸文、泰文等全部被收納進來[1]。另外統一碼也把世界上大部分可以被書寫的各種符號列入了，因此擴大了可以使用文字之範圍。

3.1 高容量性文字隱藏機制

本文所提之演算法中的秘密分享機制，針對文獻[7]所提出之秘密分享機制在加以改進，改善演算法之後資訊隱藏可以容納更多訊息。以(*2, 3*)-threshold 為例，首先定義兩個 3 × 3 的矩陣如下所示：

$$S_{white}=\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\end{bmatrix},\quad S_{black}=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}。$$

由這兩個矩陣我們可以得出兩個矩陣集合 C_0 和 C_1：

$$C_0=\left\{\begin{bmatrix}1&0&0\\1&0&0\\1&0&0\end{bmatrix},\begin{bmatrix}0&1&0\\0&1&0\\0&1&0\end{bmatrix},\begin{bmatrix}0&0&1\\0&0&1\\0&0&1\end{bmatrix}\right\}$$

與

$$C_1=$$

$$\left\{\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},\begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix},\begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix},\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix},\begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}\right\},$$

其中 C_0 矩陣集合中，因為每個矩陣中必需有由 1 所組成的行，所以共有 3 個矩陣；C_1 矩陣集合中，因為矩陣裡每一行只能出現一次 1，所以總共出現 $3\times2\times1=3!=6$ 個矩陣。我們可以看出，將矩陣中的任兩列以上經過 Bit-wise OR 運算後得出的 3 維向量中，若以 C_1 內的列向量運算有兩個以上的 1，可代表原機密訊息的位元 1；而以 C_0 內的列向

量運算只有一個 1，可代表原機密訊息的位元為 0。因此，判別 0 與 1 的門檻值可以設為 2，即高於(含)2 個 1 的個數值可代表位元值 1，否則即為位元值 0。另外，因為 S_{white} 與 S_{black} 矩陣中的每一列共有三種可能，分別為[1 0 0]、[0 1 0]、[0 0 1]，若我們個別定義一組位元配對分別代表不同的三列。例如：“00” = [1 0 0]、“01” = [0 1 0]、“10” = [0 0 1]。藉此即可以一個 2 位元值配對來取代 3 個位元的列向量。因此，當矩陣為 3 × 3 時，利用此方法原本需要用 3 個位元儲存的子訊息可以節省下一個位元，利用 2 個位元即可儲存，也就是節省三分之一的容量。

藉此推至所定義的矩陣為 $n \times n$ 時：

$$S_{white}=\begin{bmatrix}1&0&\cdots&0&0\\1&0&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\1&0&\cdots&0&0\\1&0&\cdots&0&0\end{bmatrix}\quad,\quad S_{black}=\begin{bmatrix}1&0&\cdots&0&0\\0&1&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&0&\cdots&1&0\\0&0&\cdots&0&1\end{bmatrix}。$$

由這兩個矩陣我們可以得出兩個矩陣集合 C_0 和 C_1：

C_0 =

$$\left\{\begin{bmatrix}1&0&\cdots&0&0\\1&0&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\1&0&\cdots&0&0\\1&0&\cdots&0&0\end{bmatrix},\begin{bmatrix}0&1&\cdots&0&0\\0&1&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&1&\cdots&0&0\\0&1&\cdots&0&0\end{bmatrix},\cdots\cdots,\begin{bmatrix}0&0&\cdots&0&1\\0&0&\cdots&0&1\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&0&\cdots&0&1\\0&0&\cdots&0&1\end{bmatrix}\right\}$$

與

C_1 =

$$\left\{\begin{bmatrix}1&0&\cdots&0&0\\0&1&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&0&\cdots&1&0\\0&0&\cdots&0&1\end{bmatrix},\begin{bmatrix}1&0&\cdots&0&0\\0&1&\cdots&0&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&0&\cdots&0&1\\0&0&\cdots&1&0\end{bmatrix},\cdots\cdots,\begin{bmatrix}0&0&\cdots&0&1\\0&0&\cdots&1&0\\\vdots&\vdots&&\vdots&\vdots\\\vdots&\vdots&\cdots&\vdots&\vdots\\0&1&\cdots&0&0\\1&0&\cdots&0&0\end{bmatrix}\right\},$$

其中 C_0 矩陣集合中，因為每個矩陣中必需有由 1 所組成的行，所以共有 n 個矩陣；C_1 矩陣集合中，因為矩陣裡每一行只能出現一次 1，所以總共出現 $n \times n\text{-}1 \times \cdots\cdots \times 2 \times 1 = n!$ 個矩陣。我們可以看出，將矩陣中的任兩列以上經過 Bit-wise OR 運算後得出的 n 維向量中，若以 C_1 內的列向量運算有兩個以上的 1，可代表原機密訊息的位元 1；而以 C_0 內的列向量運算只有一個 1，可代表原機密訊息的位元為 0。因此，判別 0 與 1 的門檻值可以設為 2，即高於(含)2 個 1 的個數值可代表位元值 1，否則即為位元值 0。另外，因為 S_{white} 與 S_{black} 矩陣中的每一列共有 n 種可能，分別為[1 0 ..... 0 0]、[0 1 ..... 0 0]、…… 、[0 0 ..... 1 0]、[0 0 ..... 0 1]，若我們個別定義一組位元配對分別代表不同的五列。例如：“00…00” = [1 0 ..... 0 0]、“0 0 … 0 1” = [0 1 ..... 0 0]、…… 、“1 1 … 1 0” = [0 0 ..... 1 0]、“11…11” = [0 0 ..... 0 1]。藉此即可以一個 $\log_2^n$ 位元值配對來取代 n 個位元的列向量。因此，當矩陣為 $n \times n$ 時，原本需要用 n 個位元儲存的子訊息可利用 $\log_2^n$ 個位元即可儲存。換言之，節省 $\frac{n-\log_2^n}{n}$ 的容量。

3.2 視覺偽裝分享機制

本文使用一組隨機二維座標位置當作加/解密演算法之起始座標位置，為了方便使用者操作過程中不須額外記憶任何數字以及增加操作的方便，我們將隨機二維座標位置值利用圖像來呈現，加密過程中將其利用視覺安全技術分散成 Shares 並取代各 Cover-image 之 LSB。我們同時採用 (k, n)-threshold 文字以及視覺秘密分享機制，將秘密訊息以及關鍵二維座標位置值分解成 n 份子訊息與 n 份 Shares，欲得知秘密訊息，必須結合門檻值 k 張以上之子訊息才能將秘密訊息還原，如此一來可以提高安全信任，降低訊息遭到破解的機會。此外，我們採用統一碼(Unicode)編碼，擴大了可以使用文字之範圍。

Input：

(1)欲隱藏的秘密訊息；

(2)n 張 Cover-image；

(3)隨機二維座標位置(X，Y)。

Output：

(1) n 張 Stego-image。

秘密隱藏執行步驟如下：

Step1：將秘密訊息轉換成 Unicode 編碼。

Step2：將十六進位編碼的 Unicode 編碼轉換成二進位碼，並利用第 3.1 節之 (k, n)-threshold 秘密分享機制將二進位碼分解成 n 組子訊息。

Step3：利用 (k, n)-threshold 之視覺安全技術將隨機二維座標位置值(X，Y)以及長度值 L 之影像圖片分解成 n 張隨機二維座標位置 Shares。

Step4：將 Cover-image 的左上角定義為(0，0)，將第 i 組子訊息從第 i 張 Cover-image 的第七位元值層(Bit-plane-7,簡稱 BP-7)的(X，Y)位置開始由上而下、由左而右取代原有數值，$i=1,2,\ldots,n$。

Step5：將 n 張隨機二維座標位置 Shares 依序直接取代 n 張 Cover-image 的 BP-8。

Step6：合併每張 Cover-image，CI_i 內的所有位元平面，$i=1,2,\ldots,n$。

欲還原傳送者在 Stego-image 中所嵌入的秘密訊息，根據 (k, n)-threshold 視覺式秘密分享機制則必須至少集合 k 張 Stego-image。從每一張 Stego-image 取出的 BP-8 即為隨機二維座標位置的 Shares，並且在疊合 k 張 Shares 之後得到一組隨機二維座標位置(X,Y)以及長度值 L。緊接著根據上個步驟所得之隨機二維座標位置值，任取 k 張 Stego-image 的 BP-7 的隨機二維座標位置開始由上而下、由左而右取出二元值資料，其長度等同疊合 k 張 Shares 所得之 L，即為 k 組子訊息。將 k 組子訊息依據文字秘密分享機制逐一還原成秘密訊息之二元值，接下來先將二元值轉換成十六進位碼，最後再把十六進位碼利用 Unicode 還原成文字，接收端即得知原始的秘密訊息。

接收端收到該隱藏機密資料的 Stego-images 後，將可依一定的方法把秘密資料萃取及還原出來。而資料解密的流程如下：

Step1：將收到的 k 張 Stego-images 逐一取出 BP-8，並且將其進行疊合，即得到隨機二維座標位置(X，Y)以及長度值 L。

Step2：Stego-image 的左上角定義為(0，0)，從 k 張 Stego-images 的 BP-7 之(X，Y)位置開始由上而下、由左而右取出長度為 L 的二進位碼，即為 k 組子訊息。

Step3：將 k 組子訊息利用第 3.1 節之 (k, n)-threshold 文字秘密分享機制還原成原始二進位碼訊息，並將其轉換成十六進位碼。

Step4：最後，將十六進位碼轉換成 Unicode 編碼，即可還原秘密訊息。

第四節 範例與實作

若以*(2,3)*-threshold 為範例說明資訊隱藏加解密的過程如下。

4.1 加密範例

Input：

(1)秘密訊息：“ICCL 資訊密碼暨建構實驗室”。

(2)三張 Cover-images，如圖 1 所示。

(3)隨機二維座標位置(X，Y)=(56，27)。

Step1：將秘密訊息“ICCL 資訊密碼暨建構實驗室”轉換成 Unicode 編碼，以十六進制表示如下：
4900 4300 4300 4C00 C78C 0A8A C65B BC78 A866 FA5E CB69 E65B 579A A45B

Step2：將十六進位編碼的 Unicode 編碼轉換成二進位碼並計算其長度，編碼如下所示：

0100100100000000010000110000000001000011000000000100
1100000000001100011110001100000010101000101011000110 0
1011011101111000111100010101000011001101111101001011 11
0110010110110100111100110010110110101011110011010101 0
010001011011

將二進位碼利用第 3.1 節之*(k, n)*-threshold 文字秘密分享機制分解成三組子訊息分別為 PART Ⅰ、PART Ⅱ與 PART Ⅲ：

PART Ⅰ：

00000000000000000000000000000000.....000
00000000000000000000000000000000

PART II:

00010000010000010000000000000000.....100
01000100000100000001000101000101

PART Ⅲ:

00100000100000100000000000000000.....000
10001000001000000010001010001010

Step3：利用*(k, n)*-threshold 視覺式秘密分享技術將隨機二維座標位置(56，27)與二進位碼長度共 448 位元長之影像圖片分解成三張隨機二維座標位置 Shares，如圖 2 所示。

Step4：Cover-image 的左上角定義為(0，0)，將子訊息從 Cover-image 的 BP-7 之(56，27)位置開始由上而下、由左而右取代原有數值。

Step5：將 Shares 取代 Cover-images 的 BP-8，如圖 3、圖 4 與圖 5 所示。

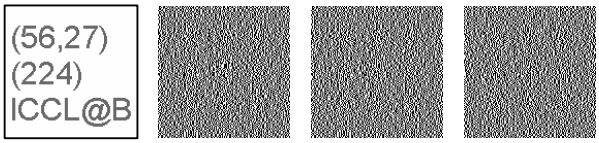
Step6：重複 Step4~Step5，直到所有子訊息與 Share 皆取代 Cover-image 之 BP-7 與 BP-8。

Step7：分別合併內含子訊息與隨機二維座標位置 Share 之 Cover-image，如圖 6 所示。

(a)Lena (b)Airplane (c)Pepper

128×128 像素 灰階圖

圖 1、本實驗所採用的三張 Cover-images



(a)隨機位置碼 (b)Share1 (c)Share2 (d)Share3

圖 2、視覺安全技術

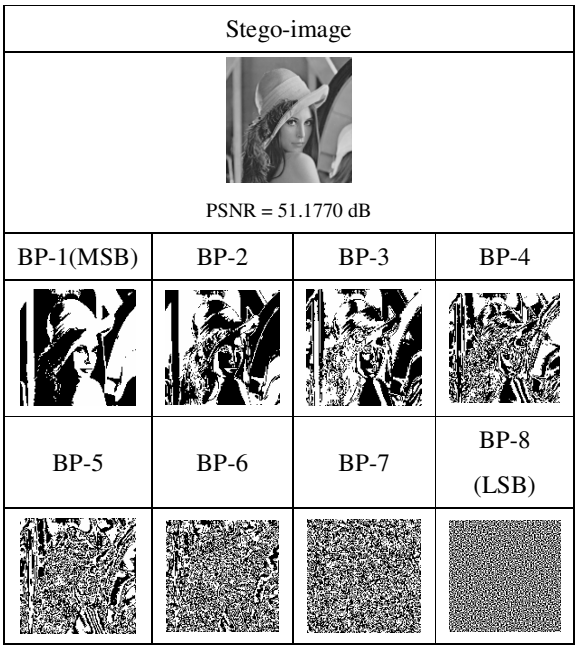


圖 3、第 1 組已取代 BP7 與 BP8 之 Cover-image

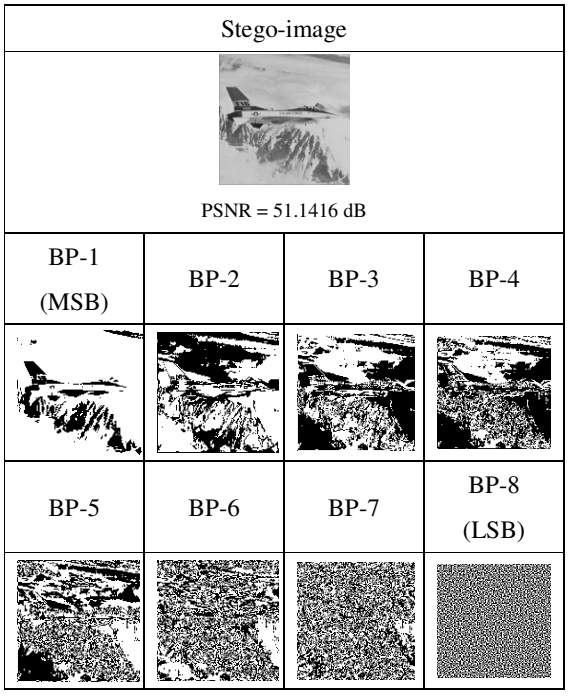


圖 4、第 2 組已取代 BP7 與 BP8 之 Cover-image

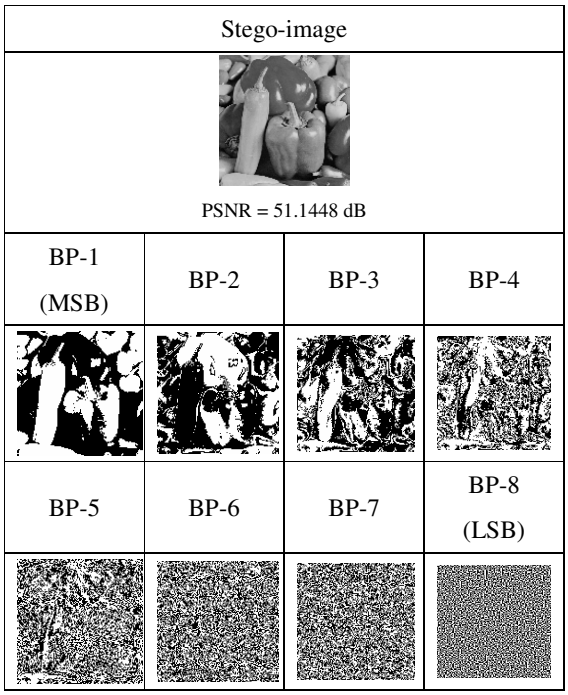


圖 5、第 3 組已取代 BP7 與 BP8 之 Cover-image

(a)Lena (b)Airplane (c)Pepper

128×128 像素灰階圖

PSNR = 50.5174 dB、50.3968 dB 、50.4437 dB

圖 6、Stego-images

4.2 解密範例

以本文 4.1 資訊隱藏加密範例所產生之三張 Stego-images 為例，任意取出其中的兩張 Stego-images，接收端萃取及還原之流程如下：

Step1：從收到的兩張 Stego-images 中取出 BP-8，並且將其進行疊合，即得到隨機二維座標位置(56，27)以及二進位碼長度值 448，如圖 7 所示。

Step2：Stego-image 的左上角定義為(0，0)，從兩張 Stego-images 的 BP-7 之(56，27)位置開始由上而下、由左而右取出長度為 448 的二進位碼，即得到兩組子訊息如下。

00000000000000000000000000000000000.....000000
0000000000000000000000000000000

00010000010000010000000000000000.....100010
00100000100000001000101000101

Step3：將兩組子訊息利用第 3.1 節之

(k, n)-threshold 文字秘密分享機制還原成:

原始二進位碼訊息：

0100100100000000010000110000000001000011000000000100110000000000110001111000110000001010100001010110001100101101110111100011110001010100001100110111111010010111101100101101101001111001100101101101010111100110101010010001011011

並將其轉換成十六進位碼，如下所示：

4900430043004C00C78C0A8AC65BBC78A866FA5ECB69E65B579AA45B

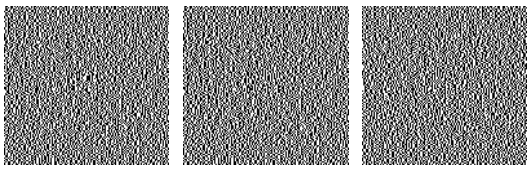
Step4：最後，將十六進位碼轉換成 Unicode 編碼：

4900 4300 4300 4C00 C78C 0A8A C65B BC78 A866 FA5E CB69 E65B 579A A45B

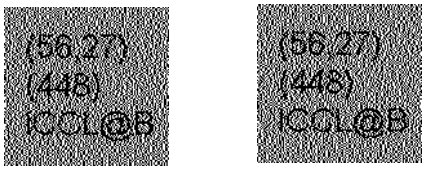
即可還原秘密訊息：

“ICCL 資訊密碼暨建構實驗室”。

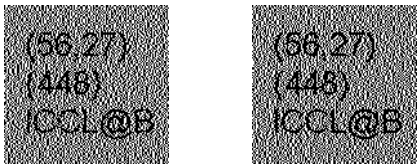
■



(a)Share1 (b)Share2 (c)Share3



(d)Share1+Share2 (e)Share1+Share3



(f)Share2+Share3 (g)Share1+ Share2+ Share3

圖 7、解密範例

第五節 討論與分析

以下我們列出相關比較表來探討本文和相關文獻[7,16]中提出的機制之差異：

[16]使用 IDEA 加密演算法，所以在操作過程中需耗費相當程度之電腦運算資源。必須依序建立 HCIT、ECPT、CIF 與 EIF 表，過程繁雜而使用者較不易操作。此外，[16]無法同時處理中、英文訊息，而且使用 Big-5 轉換內碼造成特殊之中文字體無法顯示。[7]使用虛擬隨機函數，操作過程中相對的較需耗費電腦運算資源。另外，如同上一段所述，採用 Big-5 轉換內碼會有特殊之中文字體無法顯示的問題產生。例如：塋。由上述的比較和分析探討的結果，我們可以知道本文所提出的改良方法，不會產生文獻[7,16]的各項缺點。所以，本文

的方法提供了機密資料傳送上較有效且安全的改良機制。藉此，我們彙整我們所提方法與其他方法比較下所具有的優勢如表 1 所示：

表1：演算法功能比較表

	Wang & Lu's[16]	**Hou & Tu's[7]**	**Our Scheme**
視覺化關鍵值	No	No	**Yes** (視覺安全)
提高壓縮比	No	No	**Yes**
支援秘密分享	No	Yes	**Yes**
節省電腦運算資源	No	No	**Yes**
支援多國語言	No	No(中、英文)	**Yes**

由表 1 中，可看出由於本文所提出的方法採用視覺式密碼分享技術，所以操作者可以方便讀取視覺化關鍵值，相對於[7, 16]而言，提供使用者操作上之便利性。本文所提出的方法與[7]皆採用文字秘密分享技術，可以將秘密子訊息傳送給多個接收端，接收端皆掌握秘密訊息的其中一部份，但個別的接收端無法獨立檢視秘密訊息，如此一來，即使少數秘密子訊息遭他人攔截或竊取，亦無法檢視秘密訊息。更進一步，本文所提之文字秘密分享技術改進了[7]所提之方法，大幅度提高子訊息的壓縮比，亦即增加了隱藏訊息的容量。當 S_{white} 與 S_{black} 為 $n \times n$ 矩陣時，我們可利用 $\log_2^n$ 個位元值即可代表矩陣集合的一個列向量裡的 n 個位元，可以節省 $\frac{n-\log_2^n}{n}$ 的容量。事實上，本文所提出之方法使用視覺式秘密分享技術用以取代加密演算法以及虛擬隨機函數兩項技術，而視覺式秘密分享技術的優點在於無需耗費大量的運算資源即可提供高度的安全性，所以在同樣的安全性之下本文所提出之方法更為有效率。最後，本文所使用的 Unicode 碼成功的解決[16]無法同時處理中英文並存的問題，也突破了[7]只能處理中英文訊息的限制。

第六節 結論

本文的研究是以資訊隱藏為基本的概念，加上利用視覺式秘密分享技術來加強該研究對於資訊隱藏的安全性，避免操作時耗費大量的電腦資源做運算，操作過程中視覺式秘密分享技術幫助隱藏關鍵的隨機二維座標位置值，使用者無須記憶額外的密碼或相關資訊，對使用者而言為方便並且易於操作。此外，本方法達到下列目標：

1. 用視覺式關鍵的隨機二維座標位置值，使用者無須額外記憶。
2. 提高壓縮比，達到 $\frac{n-\log_2^n}{n}$ 。
3. 採用(k, n)-threshold 文字/視覺秘密分享機制，多人同時掌控秘密訊息。
4. 加/解密過程中電腦不需耗費大量資源做運算。
5. 採用 Unicode，可支援多國語言。

綜合上述，我們的方法較之先前的研究明顯提升了進行資訊隱藏之便利性以及在安全傳送上的方便性，且可依然保持原有的優點。並在普通性圖片的偽裝下，使得秘密資料的傳送過程中獲得了保障。也提供了網路安全中，對於機密訊息傳送在兼顧便利與安全的需求下的新方法研究與可行採用的依據。

Acknowledgements: This work was supported in part by National Science Council in R.O.C. under Grant No. NSC 95-2221-E-015-002-MY2

參考文獻

[1] 鄭褚璋譯，Ken Lunde 著，2002，中日韓越資訊處理，台北，美商歐萊禮股份有限公司台灣分公司。

[2] F.L. Bauter, Decrypted Secret: Methods and Maxims of Cryptography, Spring-Verlag, Berlin, pp. 8-9, 1997.

[3] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, " Techniques for Data Hiding," IBM System Journal, Vol.35, No. 3&4, pp. 313-336, 1996.

[4] A. Curiger and B. Stuber, "Specifications for the IDEA Chip," Technical Report No 92/03, ETH Zurich, Institute for Integrate System, 1992.

[5] R.M. Davis, The Data Encryption Standard in Perspective, Computer Security and the Data Encryption Standard, National Bureau of Standards Special Publication, February 1978.

[6] H.J. Highland, "Data Encryption: a Non-mathematical Approach-Part 5," Computers & Security, 14(2), pp. 93-97, 1995.

[7] Y.C. Hou and S.F. Tu, "An Unexpanded Gray-level Visual Cryptography Using Multu-pixel Encoding Method," Journal of Information Management, Vol. 12, Num 2, April, pp. 141-161, 2005

[8] D. Kahn, The Codebreakers, Macmillan, New York, 1967.

[9] C. Kaufman, R. Perlman and M. Speciner, Network Security: Private Communication in a Public World, Prentice Hall Series in Computer Networking and Distributed Systems, Englewood Cliffs, NJ, 1995.

[10] C.H. Lin, and T.C. Lee, "A Confused Document Encrypting Scheme and Its Implementation," Computers & Security, Vol. 17, No. 6, pp. 543-551, 1998.

[11] X. Lai and J. Massey, "A Proposal for a New Block Encryption Standard," Proceedings of Eurocrypt'90, Spring-Verlag, Berlin, pp. 389-404, 1991.

[12] F.J. Neil, and J. Sushil, "Exploring Steganography: Seeing the Unseen," IEEE computer, Vol. 31, No. 2, pp. 26-34, 1998

[13] M. Naor and A. Shamir, "Visual Cryptography," Proceedings in Eruocrypt'94, Lecture Notes in Computer Science, Springer-Verlag, pp. 1-12, 1994.

[14] B. Schneier, Applied Cryptography, Wiley, New York, 1994.

[15] H.Q. Wang and S.Z. Wang, "Cyber Warfare: Steganography v.s Steganalysis," Communications of the ACM, Vol. 47, No. 10, Oct. 2004.

[16] S.J. Wang and C.K. Lu, "A Scheme of Non-sensible Document in Transit with Secret Hiding," Journal of Information Management, Vol. 9, No. 2, pp. 169-182, Jan. 2003.

[17] S.J. Wang, and K.S. Yang, "A Scheme of High Capacity Embedding on Image Data Using Modulo Mechanism," The Second International Workshop on Information Security Applications (WISA), Korea, pp. 299-309, Sept. 2001.

[18] W.H. Yeh, and J.J. Hwang, "Hiding Digital Information Using a Novel System Scheme," Computers & Security, Vol. 20, No. 6, pp. 533-538, 2001.

[19] W.H. Yeh, and J.J. Hwang, "A Scheme of Hiding Secret Chinese Information in Confused Documents," Journal of Information Management, Vol. 7, No. 2, pp. 183-191, 2001.