# 二維Hamming糾錯碼與DES密碼技術於醫學影像通訊傳輸之應用
# Applications of 2D Hamming Error-Correcting Code and DES Encryption Techniques for Medical Image Communication Transmission

林胤忱[1] 林宗宏[1] 林基源[1] 王俊傑[2] 黃喻翔[1]

[1]國立勤益科技大學 資訊工程系(所)
[2]國立中興大學 電機工程研究所
e-mail:cylin2.monica@msa.hinet.net

## 摘 要

數位醫療影像資料量大，為了滿足通訊傳輸上的可靠度與完整性，本論文旨在提出一個整合二維漢明(Hamming)編碼與 DES(Data Encryption Standard)加密技術於數位醫學影像傳輸之應用。因為，隨著資訊與通訊時代的來臨，當數位醫療影像資料透過開放式通訊網路傳送時，可能會因為通道雜訊或干擾的情況發生而使資料產生錯誤，因此為了能夠讓數位醫療影像資料能夠正確的傳遞和接收，本論文使用二維漢明編碼之通訊編碼技術來解決或改善這些遭受破壞的資料，以提高數位醫療影像資料傳輸之可靠度，且對於醫療影像資料於通訊傳輸上之完整性則使用 DES 資料加密技術來達到鑑別的目標。

關鍵字：二維漢明(Hamming)碼；DES (Data Encryption Standard)；數位醫學影像。

## 1. 前言

隨著資訊與通訊時代的來臨，當數位醫療影像資料透過開放式通訊網路傳送時，可能會因為通道雜訊或干擾的情況發生而使資料產生錯誤，而醫療影像是幫助醫生做臨床上診斷的重要工具，所以影像的內容通常不充許在通道傳輸的過程中受到更動，這時為了達到高可靠的傳輸，錯誤更正碼(Error Correction Code, ECC)已扮演舉足輕重的角色，同時發揮無可取代作用。而我們通常將通訊系統中的錯誤更正碼歸類為通道編碼(Channel Coding)，它不同於資料壓縮的訊源編碼(Source Coding)，其目的在確保資料傳輸的正確性。資料在傳輸或接收的過程中可能因為傳輸媒介的可靠度不佳，或外在因素的干擾而遭到破壞（即資料產生錯誤），而錯誤更正碼的作用即是儘可能還原這些遭受破壞的資料（即將錯誤資料更正）。

在一般數位通訊系統(Digital Communication System)運用中，如何在存有通道雜訊(Noise)或干擾(Interference)的情形下持續可靠且正確的傳遞或接收資料，這些問題都可以藉由錯誤更正碼的運用來解決或改善[1-2]。

錯誤更正碼主要是在編碼時，加入具有結構特性之糾錯資訊(Error Detection and Correction Information)；即額外使用部分頻寬傳輸額外位元，然後在接收端接收訊號時，則使用相對應之錯誤更正解碼，利用糾錯資訊來達成錯誤偵測與更正之目的，進而達到降低位元錯誤率的目標[3]。

另一方面，醫療影像於通訊傳輸上之完整性我們則使用 DES 加密技術來達到鑑別的目標[4]，首先在傳送端的部份會發送出兩張影像，一張是原始的醫療影像，一張則是經由 DES 加密過後的加密影像，當兩張影像皆傳送至接收端後則把經過加密後的影像解密，與原始影像做比對，如果解密後與原始影像作比對，若兩張影像不相同則表示影像已被更動或產生錯誤，這時則要求傳送端重新傳送。

## 2. 二維漢明編碼技術

一般的(7,4)漢明編碼它原理簡單且具備有一位元錯誤偵測與更正的能力。但此技術對位元錯誤的更正能力較低，在本論文中，我們針對錯誤更正技術中的「一維(7, 4)漢明碼」技術做進階的發展成為「二維(7, 4)漢明碼」錯誤更正技術[5]，以提升編碼處理錯誤更正之能力。以下為編碼原理與程序：

***Step1***：假設資料位元有 4 位元則利用 $2^K \geq M+K+1$ 可知 $K \geq 3$。

***Step2***：因同位元至少需 3 個，所以同位元 $K_i$ ($i$=1,2,3.) 放置位置為 $2^{i-1}$ 處：

| 位置 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 同位元 (K) | $K_1$ | $K_2$ | $M_1$ | $K_3$ | $M_2$ | $M_3$ | $M_4$ |

***Step3***：假設原始資料為 Bit0、Bit1，……，Bit15，我們將它排列成二維形式之資料，而此二維資料可看成 4 列 4 行的一維資料。

列→

| 行 ↓ | | | | | |
|---|---|---|---|---|---|
| | Bit0 | Bit1 | Bit2 | Bit3 | R1 |
| | Bit4 | Bit5 | Bit6 | Bit7 | R2 |
| | Bit8 | Bit9 | Bit10 | Bit11 | R3 |
| | Bit12 | Bit13 | Bit14 | Bit15 | R4 |
| | C1 | C2 | C3 | C4 | |

***Step4***：利用 *Step1*、*Step2* 對每一列產生漢明編碼。

| | | | | | | |
|---|---|---|---|---|---|---|
| $K_1$ | $K_2$ | Bit0 | $K_3$ | Bit1 | Bit2 | Bit3 |
| $K_4$ | $K_5$ | Bit4 | $K_6$ | Bit5 | Bit6 | Bit7 |
| $K_7$ | $K_8$ | Bit8 | $K_9$ | Bit9 | Bit10 | Bit11 |
| $K_{10}$ | $K_{11}$ | Bit12 | $K_{12}$ | Bit13 | Bit14 | Bit15 |

***Step5***：用 *Step1*、*Step2* 對 *Step4* 產生的編碼之每一行再進行漢明編碼。

| | c | c | m | c | m | m | m |
|---|---|---|---|---|---|---|---|
| | $K_1$ | $K_2$ | Bit0 | $K_3$ | Bit1 | Bit2 | Bit3 |
| | $K_4$ | $K_5$ | Bit4 | $K_6$ | Bit5 | Bit6 | Bit7 |
| | $K_7$ | $K_8$ | Bit8 | $K_9$ | Bit9 | Bit10 | Bit11 |
| | $K_{10}$ | $K_{11}$ | Bit12 | $K_{12}$ | Bit13 | Bit14 | Bit15 |
| | ↓ 漢明編碼 | ↓ 漢明編碼 | ↓ 漢明編碼 | ↓ 漢明編碼 | ↓ 漢明編碼 | ↓ 漢明編碼 | ↓ 漢明編碼 |
| C | $K_{13}$ | $K_{16}$ | $K_{19}$ | $K_{22}$ | $K_{25}$ | $K_{28}$ | $K_{31}$ |
| C | $K_{14}$ | $K_{17}$ | $K_{20}$ | $K_{23}$ | $K_{26}$ | $K_{29}$ | $K_{32}$ |
| M | $K_1$ | $K_2$ | | $K_3$ | | | |
| C | $K_{15}$ | $K_{18}$ | $K_{21}$ | $K_{24}$ | $K_{27}$ | $K_{30}$ | $K_{33}$ |
| M | $K_4$ | $K_5$ | | $K_6$ | | | |
| M | $K_7$ | $K_8$ | | $K_9$ | | | |
| M | $K_{10}$ | $K_{11}$ | | $K_{12}$ | | | |

***Step6***：剩下之位置為存置原始之二維資料，二維編碼即告完成。

## 3. DES資料加密標準

DES (Data Encryption Standard)是IBM公司在1970年代所發展出來的，且DES隨後被美國國家標準局公佈為資料加密標準的一種區塊加密法(Block Cipher)[6]。DES是在秘密金鑰密碼系統中，最被廣泛運用的演算法，原因是它加解密的速度非常快，接下來本論文將介紹其加密系統的流程[7](圖一)及其重要之元件。

每一個 64 位元大小的明文輸入至 DES 加密系統後，系統會先將明文進行初始排列，打亂資料原來之順序後，再將明文經過 16 回合的運算，最後一回合將運算結果調換，經過終結排列後輸出 64 位元的密文，而在這 16 回合的運算過程中，每一回合的運算動作皆相同。

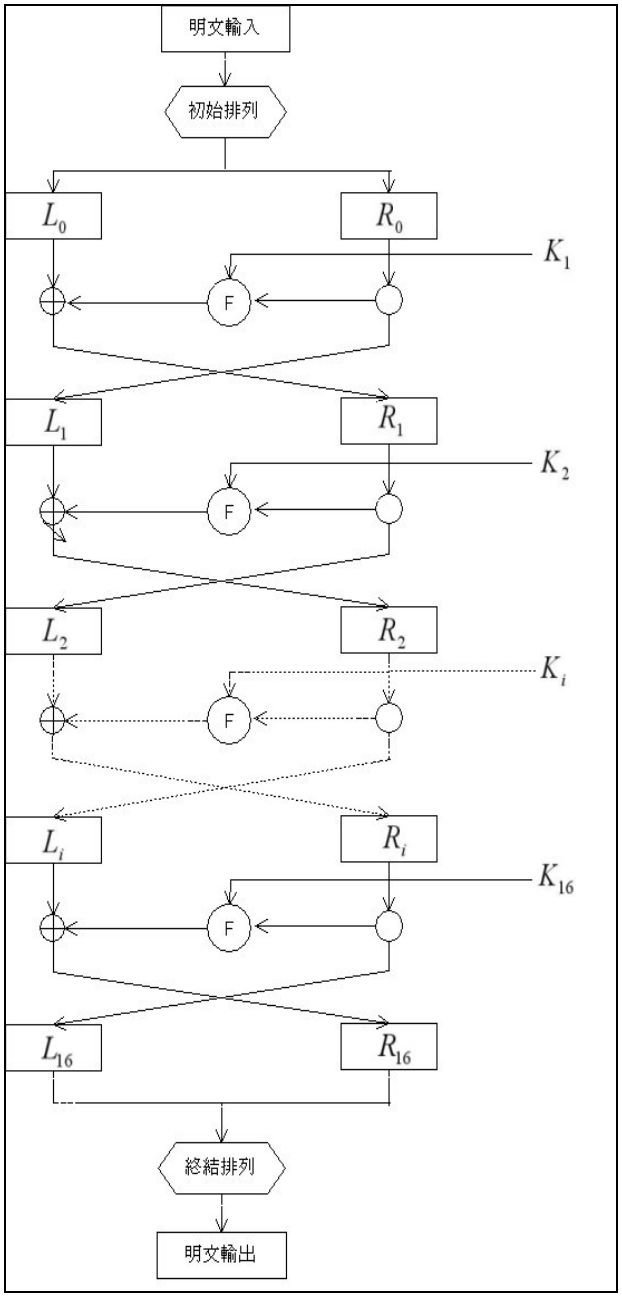
假設目前要進行第 1 回合的運算，首先系統先將此回合輸入的 64 位元分為$(L_0, R_0)$各 32 位元的區段，接下來將$R_0$和子金鑰$K_0$經過 f 函數運算後，再與$L_0$做互斥或(XOR)的運算，其$R_0$運算過後的輸出則為下一回合的$R_1$，而下一回合的$L_1$則為此一回合的$R_0$，此回合輸出的$(L_1, R_1)$即為下一回合的輸入。以上每個回合的處理程序都可以用下列方程式來表示：

$$L_i = R_i - 1 \tag{1}$$

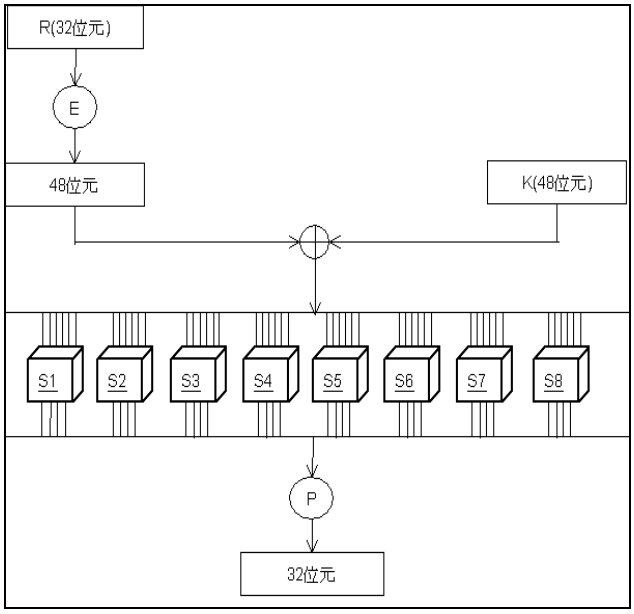
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i) \tag{2}$$

而 F 函數是 DES 加密演算法中最重要的部分(圖二)。每個回合所輸入的 R 為 32 位元，使用的子金鑰$K_i$則為 48 位元。首先 32 位元的密文 R 會先經過擴增排列 E(表一)先擴充為 48 個位元後，再與另一組輸入的子金鑰 K 做互斥或(XOR)運算，運算後的結果再平均分配給 8 個替換盒$S_1$、$S_2 \cdots S_8$分別為六個位元的輸入。替換盒的替換方式是把六位元輸的的最左邊與最右邊的位元取出並以十進位數值來表示列數，而中間

剩下的四個位元同樣以十進位數值來表示行數。就 $S_1$(表二)來說，如果輸入為 110010，則被選取的列就是 10(第二列)，被選取的行則是 1001(第九行)，位於第二列第九行的數值是 10，所以輸出就是 1010。每個替換盒輸出後各為四個位元總共 32 位元，最後經過縮減排列 P(表三)輸出。




圖一 DES加密流程圖




圖二 F(*R*,*K*)之計算流程

表一 擴增排列(E)

| | | | | | |
|---|---|---|---|---|---|
| 32 | 1 | 2 | 3 | 4 | 5 |
| 4 | 5 | 6 | 7 | 8 | 9 |
| 8 | 9 | 10 | 11 | 12 | 13 |
| 12 | 13 | 14 | 15 | 16 | 17 |
| 16 | 17 | 18 | 19 | 20 | 21 |
| 20 | 21 | 22 | 23 | 24 | 25 |
| 24 | 25 | 26 | 27 | 28 | 29 |
| 28 | 29 | 30 | 31 | 32 | 1 |

表二 DES 的 S-Box 定義(以 S1 為例)

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 14 | 4 | 13 | 1 | 2 | 15 | 11 | 8 | 3 | 10 | 6 | 12 | 5 | 9 | 0 | 7 |
| 0 | 15 | 7 | 4 | 14 | 2 | 13 | 1 | 10 | 6 | 12 | 11 | 9 | 5 | 3 | 8 |
| 4 | 1 | 14 | 8 | 13 | 6 | 2 | 11 | 15 | 12 | 9 | 7 | 3 | 10 | 5 | 0 |
| 15 | 12 | 8 | 2 | 4 | 9 | 1 | 7 | 5 | 11 | 3 | 14 | 10 | 0 | 6 | 13 |

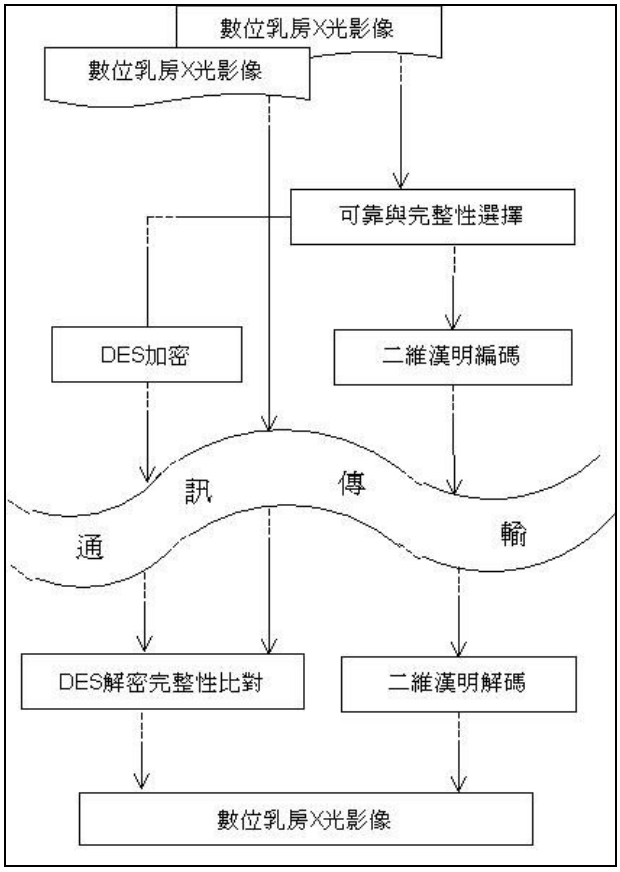
表三 縮減排列(P)

| | | | |
|---|---|---|---|
| 16 | 7 | 20 | 21 |
| 29 | 12 | 28 | 17 |
| 1 | 15 | 23 | 26 |

| | | | |
|---|---|---|---|
| 5 | 18 | 31 | 10 |
| 2 | 8 | 24 | 14 |
| 32 | 27 | 3 | 9 |
| 19 | 13 | 30 | 6 |
| 22 | 11 | 4 | 25 |

## 4. 提出的整合系統

本論文中提出一套整合系統(圖三)在醫學影像的應用上，我們可以依照我們的需求選擇加強傳輸的可靠性亦或是傳輸完整性之鑑別。在加強可靠性的部份，經過漢明編碼後之醫療影像在經過通訊傳輸中所受到的干擾可以在漢明解碼後得到改善，使影像具備較大的正確性。而在完整性鑑別部份，兩張影像傳輸至接收端後會有兩張相同大小的影像，一張是原始的醫療影像，一張則是經由 DES 加密過後的密文影像，這時可以將加密後的影像解密，然後與原始影像做比對，如果比對結果兩張影像不一致，則表示影像通訊傳輸過程中已遭更動或產生錯誤，這時即要求傳送端重新傳送。

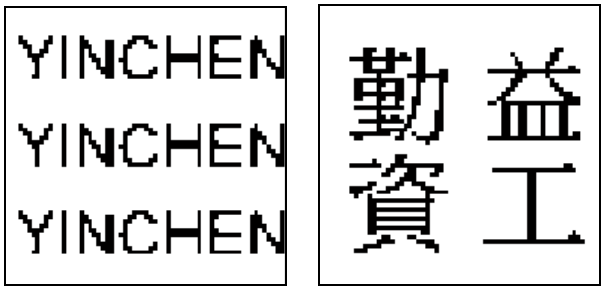



圖三 整合系統架構圖

## 5. 實驗結果

### 5.1 二維漢明編碼糾錯效能評估

測試影像為灰階影像，其影像大小為 64x64 像素，首先將影像經過漢明編碼後，再對編碼後的影像做亂數破壞，並利用漢明解碼來更正破壞後的影像，最後再對還原後的影像做 *PSNR* 評估。影像破壞設定約為 300 個位元數。漢明編碼是使用二維漢明碼，模擬資料位元為 1、2、4、8、16 及 32 的位元數，如測試影像以 1 位元為單位做漢明編碼，需增加 2 個檢查位元，所以 1 位元經過漢明編碼後，位元總長度增加為 3 位元，其餘資料位元數以此類推。

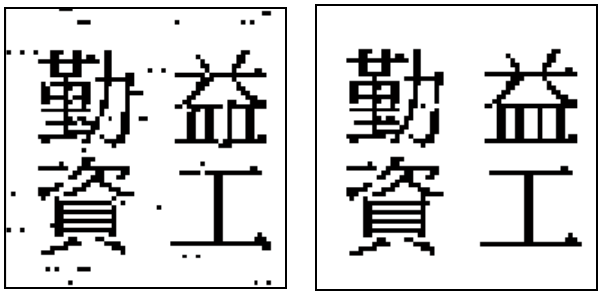



圖四 原始影像

***一維漢明碼與二維漢明碼效能評估***:

資料位元=4，檢查位元=3，漢明位元總長度=7

*Error bits* = 288　　*Error bits* = 288



經一維漢明碼更正後:　　經二維漢明碼更正後:

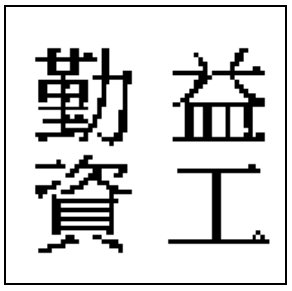
*Error bits* =61　　*Error bits* =2

*PSNR*=18.27 dB　　*PSNR*=33.11 dB

***二維漢明碼效能評估 1***:

資料位元=1，檢查位元=2，漢明位元總長度=3

*Error bits* = 292　　*Error bits* = 294




更正後:

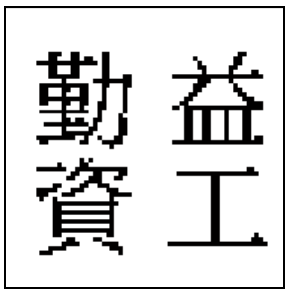
*Error bits* =4 *Error bits* =4

*PSNR*=30.10 dB *PSNR*=30.10 dB

***二維漢明碼效能評估 2***:

資料位元=2，檢查位元=3，漢明位元總長度=5

*Error bits* = 296 *Error bits* = 298




更正後:

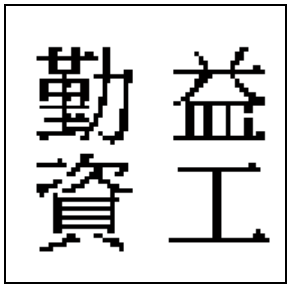
*Error bits* =2 *Error bits* =2

*PSNR*=33.11 dB *PSNR*=33.11 dB

***二維漢明碼效能評估 3***:

資料位元=4，檢查位元=3，漢明位元總長度=7

*Error bits* = 294 *Error bits* = 294




更正後:

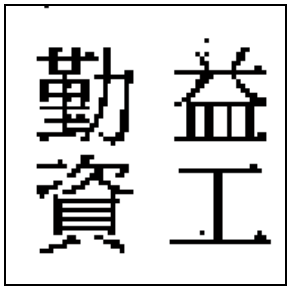
*Error bits* =1 *Error bits* =2

*PSNR*=36.12 dB *PSNR*=33.11 dB

***二維漢明碼效能評估 4***:

資料位元=8，檢查位元=4，漢明位元總長度=12

*Error bits* = 292 *Error bits* = 284




更正後:

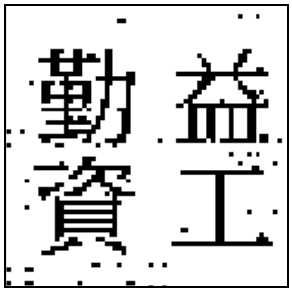
*Error bits* =5 *Error bits* =4

*PSNR*=29.11 dB *PSNR*=30.10 dB

***二維漢明碼效能評估 5***:

資料位元=16，檢查位元=5，漢明位元總長度=21

*Error bits* = 292 *Error bits* = 284




更正後:

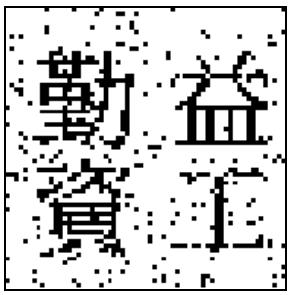
*Error bits* =62 *Error bits* =58

*PSNR*=18.20 dB *PSNR*=18.50 dB

***二維漢明碼效能評估 6***:

資料位元=32，檢查位元=6，漢明位元總長度=38

*Error bits* = 286 *Error bits* = 286



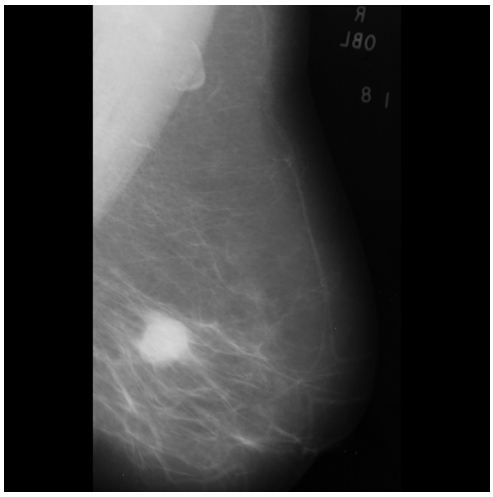

更正後:

*Error bits* =240 *Error bits* =240

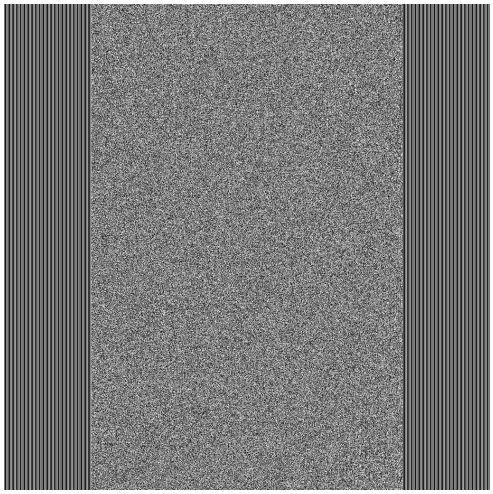
*PSNR*=12.32 dB *PSNR*=12.32 dB

### 5.2 DES 影像加解密實驗結果

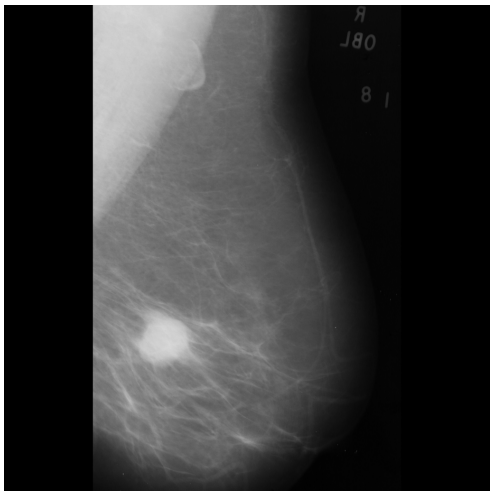
在本論文中，我們使用 MIAS 所提供的影像來做 DES 的加解密。



圖五 原始影像



圖六 DES 加密後影像



圖七 DES 解密後影像

## 6. 結論

醫療影像在通訊傳輸上，傳輸時除了頻寬，傳輸品質也需要維持一定的水準。本文中，我們提出加入額外且具結構特性之二維漢明編碼技術，用以達到傳輸過程中之位元錯誤偵測，並達成位元錯誤更正能力，以做到有效降低位元之錯誤率，以提升通訊傳輸上之可靠度。由模擬結果也呈現了我們所提出的二維漢明碼影像品質有明顯的改善，以編碼資料位元為 4 bits，檢查位元為 3 bits 為例，在錯誤位元數皆為 288 bits 時，二維漢明碼的錯誤位元數較一維漢明碼減少 59 bits，而 *PSNR* 則改善約 15 dB。而在確保影像傳輸過程之完整性方面，我們在實驗結果也成功的把一張醫療影像做加解密的動作，這也將提供於接收端的鑑別比對。

## 參考文獻


[1] 林高洲，錯誤更正碼在寬頻通訊系統之運用與發展(上)，中華民國電子零件認證委員會。

[2] 林高洲，錯誤更正碼在寬頻通訊系統之運用與發展(下)，中華民國電子零件認證委員會。

[3] O. M.Ibarra, A. G. Guierrez and J.C.R. Suarez, “Neural Networks for Error Correction of Hamming Code,” Proceedings on Information Theory and Statistics, pp. 94, 1994.

[4] 郭育郎，“淺談檔案加密原理與實務應用”，網路社會學通訊期刊，Vol. ，2006。

[5] 林基源、王俊傑、張書豪、林胤忱、黃維廉，“二維漢明碼編碼技術於通訊傳輸上之效能評估”，第二屆海峽兩岸科技與人文教育暨產學合作研討會，pp. 644-649, Dec. 2008.

[6] Data Encryption Standard, Federal Information Processing Standard (FIPS), vol. 46, National Bureau of Standards, January 1977.

[7] C.S. Laih, L. Harn, C.C. Chang, Contemporary cryptography and its application, Flag, 2003.