無需原始數位影像
之小波轉換浮水印技術

李南逸　　王靖鈞
南台科技大學資訊管理系　　南台科技大學資訊管理系
教授　　學生
nylee@mail.stut.edu.tw　　M9490301@webmail.stut.edu.tw

摘要

在本文中，所受保護的數位影像是利用轉換技術將空間域的資訊轉換至頻率域之中，並利用低頻帶產生一串私密字串在浮水印嵌入時使用，而在浮水印嵌入、萃取時利用低頻帶的係數產生出新的中頻帶增強嵌入的強度及協助萃取浮水印時頻帶值的完整性，並利用影像相關的關聯性產生一把私密金鑰進行浮水印驗證。最後，從多項實驗的結果驗證本文所提出的演算法擁有足夠強韌性通過各項攻擊, 例如模糊、銳化以及 JPEG 失真壓縮等等。

關鍵詞：離散小波轉換、數位浮水印、版權保護、密碼學。

1. 前言

網際網路近年的快速成長，使得數位檔案得以大量輕易的快速傳送。然而，如何保護數位影像資料的版權已變成現今一個重要的問題。於是「數位浮水印」的版權保護技術被應用來保護數位資料的版權安全，此項技術可以將浮水印資料隱藏至受保護的數位資料之中，因此可以定義出原創者的所有權。而一般來說，數位浮水印的技術可滿足以下幾項之條件：

a. **強韌性:**嵌入後的浮水印資訊可以由受保護的數位影像中萃取出來，並且在受保護的數位資料被任何的蓄意修改破壞後，不會受很大的影響。

b. **不可視性:**浮水印嵌入至受保護的數位影像後，數位影像本身品質不會透過人眼直接看出改變。

c. **安全性:**除了數位影像的擁有者之外，任何人無法輕易的將嵌入後的浮水印資訊取出或是移除掉。

d. **盲目性:**在未知原始數位影像的情況之下，版權擁有者仍然可以將所嵌入的數位浮水印成功的萃取出。

e. **完整性:**數位影像無論受到多嚴重的影像攻擊，萃取出的數位浮水印影像仍可以完整地用肉眼辨識出來。

浮水印技術可以分成兩種典型：空間域的方法以及頻率域的轉換技術。空間域的技術雖然簡單且無需複雜的轉換，但是此種技術之安全性較低。首先在 1994 年，Schyndel, Tikel 和 Osborne [10] 提出了使用空間域的浮水印技術，這個方法主要目的是將浮水印資訊嵌入至影像每一個像素中的後四個位元之中。但是這種方法對於影像壓縮攻擊(JPEG 壓縮)沒有足夠的強韌性。相對來說，將影像由空間域轉換至頻率域的浮水印技術比空間域的技術較強，且可以抵抗較多的影像破壞攻擊。而在頻率轉換的浮水印技術如離散餘弦轉換(DCT) [6][7][12][13]和離散小波轉換(DWT) [8][9][14]的技術都是將浮水印嵌入至頻率域之中，而且浮水印資訊將被分散嵌入至受保護的圖形之內。因此攻擊者將很難地

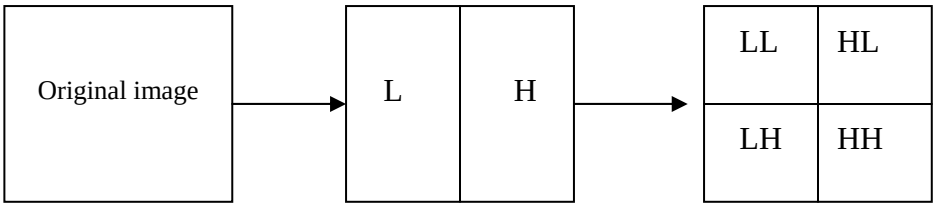


圖 1. 一階 DWT 示意

去察覺出嵌入的資訊並修改。

Adrian 和 Ioannis [2] 提出 DCT 轉換為基礎的浮水印技術將數位浮水印嵌入至 DCT 所轉換的頻率域之中。在這個方法裡首先將受保護的影像分割分許多 8x8 的區塊，透過轉換後的中頻帶取出給予浮水印嵌入的係數值。但是以 DCT 為基礎的數位浮水印技術由於 JPEG 的壓縮技術以及大部份的影像破壞攻擊有可能會影響到中頻帶的數值，於是 Shapiro [5] 在 1993 年以及 Stollnitz 等人[3] 在 1995 年時提出了使用以 DWT 為基礎的浮水印技術以改善這些問題。

如圖 1 所示為基本的 DWT 一階轉換過程步驟。首先對空間域之圖形執行水平運算，將影像分別過濾出低(L)、高(H)頻帶。在完成水平運算之後再分別對 L 及 H 執行垂直運算，最後即可完成一個初步的 DWT 演算過程，而分別得到 LL、LH、HL 及 HH 四個頻帶。在完成 DWT 的步驟後可以發現到大部份的重要係數會集中在 LL 這個頻帶之中，因此集中在 LL 頻帶中的係數值在影像受到各種影像破壞攻擊時不會輕易的被影響而變化太大。

1.2. 相關研究

在 2002 年，Joo 等人 [11] 提出了一個強韌型浮水印技術，此技術是將一個 MxN 大小的數位影像利用 n 階 DWT 轉換後，先濾出 LL_n 、LH_n 、HL_n 以及 HH_n 四個頻帶，再針對 LL_n 執行 n+1 階的 DWT 轉換得到 LL_{n+1} 、LH_{n+1}、HL_{n+1} 以及 HH_{n+1}，並保留 LL_{n+1} 內的係數值後將其餘頻帶內的係數值(LH_{n+1} 、HL_{n+1} 以及 HH_{n+1})調整為 0，在完成上述步驟後再執行一次反向 DWT 得到一個新的 LL'_n。在這個方法中所用的浮水印嵌入法則是使用如圖 2 所示，利用 $|LL_n - LL'_n|$ 運算產生出一組差異的索引值 idx(i)，利用 idx(i)的索引值以及讀取浮水印執行嵌入運算 $LL_3 = LL_3' \pm k \times w(idx(i))$ ，式子中 k 是嵌入浮水印時的強度係數，w 則是浮水印訊號。在這個方法進行嵌入時會一直讀取浮水印訊號直到讀取完畢前一直重覆嵌入的步驟而形成一種回合制的嵌入過程。

[11] 所提出的浮水印嵌入技術是依據浮水印資訊的長度進行回合制的嵌入。由於這種方式效能上尚未達到最佳，因此，在 2005 年時 Lou 等人[4]改進了[11]的做法，在嵌入浮水印時相同利用 $|LL_n - LL'_n|$ 產生位置差異性索引值 idx(i) 嵌入浮水印，並保留了 idx(i) 成為一把私密金鑰，在浮水印取出時按照 idx(i) 的資訊直接針對索引值位置做萃取的動作，而免除了重覆嵌入動作。因此，[4]的方法相較於[11]浮水印技術可得到較好的效能。2005 年，Reddy 等人[1]則是使用了浮水印的重要係數計算出所嵌入頻帶的百分比。在[1]中，利用 IxJ 的而受保護數位影像及 MxN 大小的數位浮水印，分別對要嵌入浮水印資訊的數位影像及原始浮水印，進行 n 階 DWT 和一階的 DWT，並先從進行一階 DWT 的浮水印取得重要係數 $w_1^0(i,j)$ ，緊接著計算所需嵌入的 DWT 係數的百分比及門檻值 T。

而在浮水印取出的步驟則是需要用到原始的數位影像及浮水印。而分別對需要受檢驗的數位影像、原始影像執行 n 階以及原始浮水印影像執行一階的 DWT 轉換。接著透過原始影像及受驗影像轉換後頻帶之間的差異性找

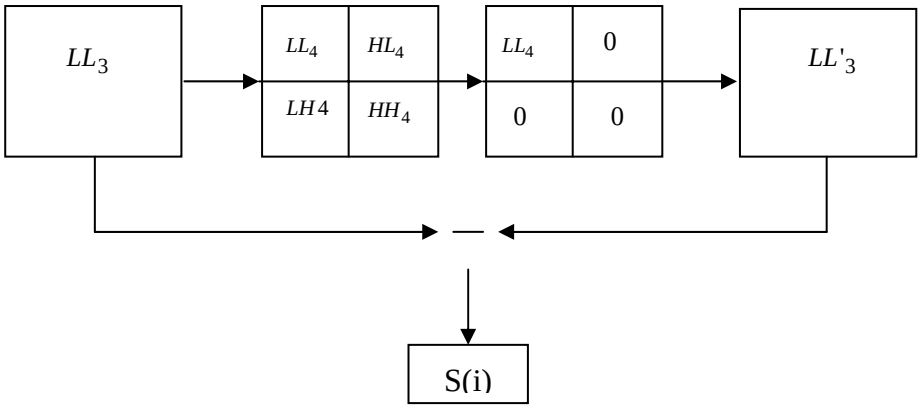


圖 2.透過[11]概念產生私密字串流程

出浮水印的嵌入點並取出初始的浮水印資訊 $x'(m,n)$，但是為了取得一個更完整的浮水印影像，[1] 在含有浮水印的嵌入點鄰近周圍運用原始浮水印的重要係數 $w_1^0(i,j)$ 計算出損害率進而取得更完整的浮水印 $\hat{x}(m,n)$ 。

本文主要目的在加強 [1] 的浮水印演算法，在我們的作法中並不需要原始浮水印的重要係數，並且結合利用 [11] 的作法產生一串秘密字串在浮水印嵌入時使用，並利用原始影像及浮水印影像邊界之間的關聯性取而進行動態的 DWT 轉換，接著在浮水印嵌入時將秘密字串與浮水印訊號進行互斥或(XOR)的邏輯運算，以及利用 LL 頻帶的係數產生新的 LH' 頻帶藏入演算後的浮水印資訊。在完成嵌入過程後產生一把私密金鑰以方便日後驗證使用。

2. 本論文之方法

在本文所提出的方法中，我們先將要受保護的影像做 θ 階 DWT 轉換並針對 LL_θ 頻帶運用[11]的概念產生新的 LL'_θ 頻帶，並利用 $|LL_\theta - LL'_\theta|$ 得到私密字串 S(i)，門檻值 T 則是透過將 LL_θ 頻帶所有係數總和後再取得平均值並乘上一個亂數 n 而產生。在嵌入浮水印的階段，我們先將原來的浮水印使用亂數值打散。在已知的情形之下，LL_θ 頻帶之間的係數在被任何影像破壞後不易被嚴重地影響到。因此，在將打散後的浮水印資訊嵌入至 LH 頻帶之前，我們利用 LL_θ 係數產生一個新的 LH' 頻帶，取代原本的 LH 頻帶來供給浮水印嵌入用。

2.1 浮水印嵌入演算

我們所提出的方法是建立在[1]以及[11]兩種技術概念的精神上，在本演算法中所得到的門檻值 T 透過 LL_θ 平均值乘上一個亂數值 n 所產生。而全新的 LH' 頻帶則是經由 LL_θ 內的係數關聯所建立而成(基本嵌入流程如圖 3 所示)。透過 [11] 中的概念，一組秘密字串 S(i)以及浮水印強度係數為 α 和 β。其所有的嵌入演算過程如下列步驟所示：

Step1: 將一個要嵌入浮水印資訊 MxN 大小的數位影像 A 與 IxJ 大小的浮水印 B 進行如式子(1)的大小關聯比例求出近似 2^θ 的值，並取 θ 進行如圖 3 所示的 θ 階 DWT 轉換，並利用式子(1)計算出門檻值 T。

$$2^\theta \cong (A_M / B_I), \qquad (1)$$

$$T = n \cdot Avg(LL_3), \qquad (2)$$

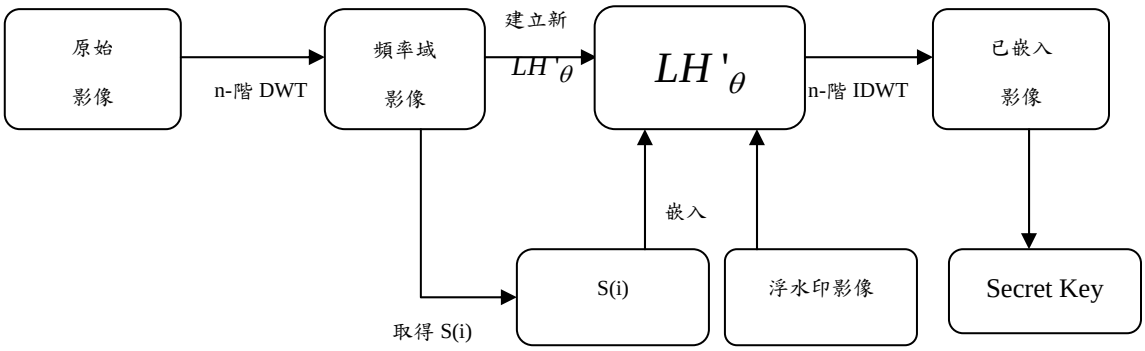


圖 3. 浮水印嵌入流程

在式子(1)中，n 是一個隨機產生的亂數，並且 $Avg(LL_\theta)$ 為在 LL_θ 中的係數平均數。

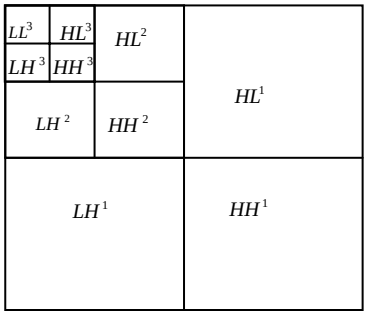


圖 4. θ 階 DWT

Step2: 針對 LL_θ 頻帶作一階 DWT 轉換，保留下 $LL_{\theta+1}$ 的係數值後，去除其餘頻帶($LH_{\theta+1}$、$HL_{\theta+1}$和 $HH_{\theta+1}$)內所有係數值。接著執行如圖 2 所示的流程後再執行一階的反 DWT 得到一個新的頻帶 LL'_θ，並透過計算$|LL_\theta - LL'_\theta|$得到一連串的秘密字串 S(i)。最後透過 LL_θ 內的係數運用式子(3)的演算得到新的頻帶 LH'_θ。

$$LH'_\theta = LH_\theta + \left\{ \left[LL_\theta(i,j) - LL_\theta(i+1) \right] \bmod 2 \right\}, \quad (3)$$

在式子(2)之中，LH'_θ 是用來給予浮水印嵌入的新頻帶，θ 則是代表著的 LH 頻帶階層數。

Step3: 利用先前計算出來的門檻值 T 找出在 LH_θ' 頻帶內的並找出 $T \le LH_\theta'(M,N) < LH_\theta'(M,N)_{\max}$ 重要係數，並透過式子(4)的演算將浮水印嵌入至 LH_θ' 頻帶內。

$$\hat{LH}_\theta = LH'_\theta + \left[(\alpha \cdot w(i,j)) \oplus (\beta \cdot s(i)) \right], \quad (4)$$

在式子(4)之中 α 和 β 是給予浮水印 w(i,j)以及秘密字串 S(i)的嵌入強度值。在嵌入的浮水印訊號前，先以亂數打散之後藏入 LH_θ' 頻帶之中，產生含有已嵌入浮水印資訊的頻帶 $\hat{LH}_\theta$。

Step4: 完成以上嵌入浮水印訊號的步驟後，將頻率域的數值進行反 DWT 轉換回空間域，並得到一張已嵌入浮水印的圖形，最後將 θ 及保留高階的 LH' 係數值和秘密字串 S(i)產生一把可以取出浮水印的私密金鑰 $S_k = (\theta \| LH' \| S(i))$。

2.2 浮水印萃取演算

在本研究中，無需原始影像的協助可從受保護的影像中萃取出藏入的浮水印，而是以私密金鑰將浮水印取出，其相關取出的步驟如下：

Step1: 輸入嵌入時所產生之私密金鑰 S_k 及嵌入浮水印資訊的影像進行 θ 階的 DWT 轉換。

Step2: 重覆式子(3)的做法，對已嵌入浮水印之影像產生一個新的含有浮水印資訊 $\hat{LH}_\theta$ 頻帶。

(a) **(b)** **(c)** **(d)**

圖 5. (a) 原始影像 (b) 原始浮水印 (c) 已嵌入浮水印影像 (d) 萃取後浮水印影像

Step3: 嵌入的浮水印訊息依據式子(5)演算法取出。

$$w'(i,j)=\left(\frac{\hat{LH}_\theta - LH_\theta'}{\alpha}\right)\oplus\left(\beta\cdot S(i)\right), \quad (5)$$

在式子中 $\hat{LH}_\theta$ 為含有浮水印訊號影像的頻帶，而 LH'_θ 及 S(i)為金鑰中所保留之資訊。

Step4:利用亂數種子將步驟 3 所萃取出來的初始浮水印復原成初始的浮水印影像。

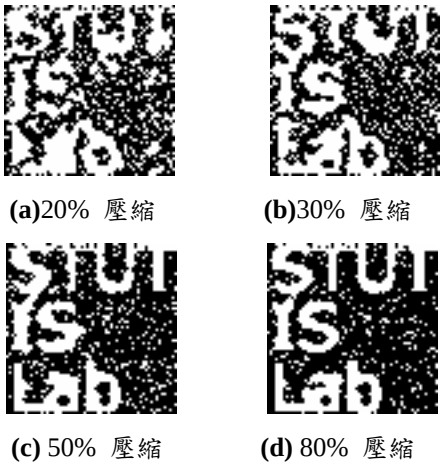
Step5:透過以上步驟所復原的是一個初始的浮水印影像 $w'(i,j)$。為了要能夠取得一個更完整的浮水印，將初始的浮水印分割成 3x3 的大小區塊依式子(6)進行雜訊消除。

$$\hat{w}(i,j)_{\max}=\sum_{m=0}^{x}\sum_{n=0}^{y}\left(\frac{\sum_{i=m}^{3}\sum_{j=n}^{3}w'(i,j)}{\frac{m}{i}x\frac{n}{j}}\right) \quad (6)$$

在式子中，x 和 y 是取出後浮水印的大小，i 及 j 為每一個區塊大小。我們計算出每一個區塊中的中間值取代最大的雜訊值以達到消除雜訊的效果。所有相關的實驗結果將會在下節呈現。

3. 相關實驗結果

在本文的相關實驗是透過多項影像破壞攻擊測試而取得的結果。所測試的影像大小是使用 512x512 大小的 8-bit 灰階影像以及 64x64 大小的浮水印黑白影像，因為 $512:64=8\cong 2^3$，因此在本實驗中進行三階 DWT 轉換。影像破壞則是利用影像處理軟體(Photoshop CS) 進行不同狀況下的模擬破壞，相關的原始影像及浮水印分別如圖 5 中的(a)及(b)所示，並且利用如式子(7)、(8)之峰值噪比值(PSNR)以及相似度(NC)計算嵌入後之影像及萃取出之浮水印影像的品質。

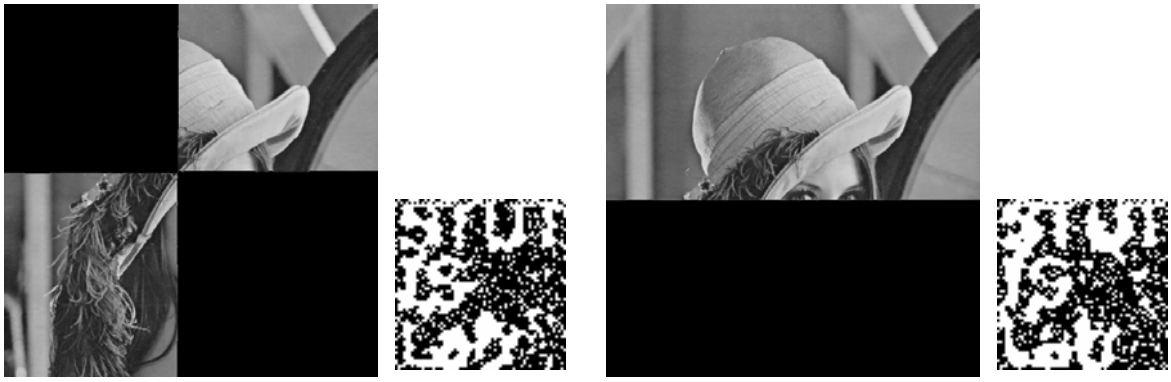


(a)20% 壓縮 **(b)**30% 壓縮

(c) 50% 壓縮 **(d)** 80% 壓縮

圖 6. 不同 JPEG 壓縮下浮水印萃取之結果

$$PSNR=10\log_{10}\frac{255^2}{MSE}dB$$
$$MSE=\frac{1}{m\cdot n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left(X_{(i,j)}-X'_{(i,j)}\right), \quad (7)$$



(a) **(b)** **(c)** **(d)**

圖 7. 不同的 50%剪裁萃取結果

$$NC = \frac{\sum_{x=0}^{x-1}\sum_{y=0}^{y-1} w(x,y)w'(x,y)}{\sum_{x=0}^{x-1}\sum_{y=0}^{y-1} w^2(x,y)}, \qquad (8)$$

在本實驗中分別按照頻帶大小不同的比例設定不同的強度值α_1 及 α_2，其值分別設定為 0.045 和 0.095 隨浮水印影像嵌入至頻帶 LH_3 及 LH_2 之中。而給予秘密字串 S(i)的係數 β 則為 0.9。在這之中使用了不同的強度值是為了平衡嵌入後影像的結果。而在嵌入後的影像以及最出後的浮水印影像的 PSNR 值分別為 47.5 及 24.36db 值，取出結果如圖 5(c)及(d)嵌入後以肉眼不易辨識出浮水印嵌入點，且取出之浮水印影像也可輕易地直接用肉眼辨識出來。接後演算法強度則是分別以 JPEG 壓縮、剪裁、尺寸大小調整、模糊以及邊緣銳化破壞後取出測試，相關結果將在以下來分別討論。

3.1. JPEG 壓縮

JPEG 影像壓縮法是較常使用來對影像進行壓縮的方式，當影像透過 JPEG 壓縮後，在中頻及高頻帶的係數容易被破壞掉，而留下的則是影像邊緣的係數。而在 JPEG 壓縮測試時，我們分別對含有浮水印的影像進行了 80%,50%, 30% 及 20%的品質壓縮後再取出浮水印，相關的結果如圖 6 (a) 、(b) 、(c) 及 (d)所示，在經過這些品質的壓縮只剩下 20%的品質後，仍然可以用肉眼辨識出浮水印的內容，由表 1 中顯示在以上的壓縮破壞後仍有很高的相似度。

表 1. 不同 JPEG 壓縮 NC 值

壓縮比	20%	30%	50%	80%
NC 值	0.877	0.945	0.965	0.975

然而，我們所提出的方法跟[1]、[11]及[4]做比較，在方法[11]將浮水印嵌入在 LL_3 的頻帶之中，而[4]則將方法[11]改良後將浮水印嵌入在空間域之中，並且需要記錄先前所嵌入的位置索引來做取出的動作，因此不會受到很大的破壞影響。而在[1]的方法之中則是將浮水印嵌入在 LH_θ 頻帶之中。但是在取出浮水印時需要利用到原始影像及原始浮水印的重要係數。並在取出後的浮水印需要利用原始浮水印的重要係數做損壞計算，因此可以取得一個較為完整的浮水印圖形。而在我們的方法中是不需要原始浮水印的重要係數，並利用 $|LL_\theta - LL'_\theta|$ 產生一連串秘密字串 S(i)，以及利用低頻帶係數重新建立出近似原始的中頻帶,並利用簡單的降低雜訊的演算相同可以取出可以肉眼辨識的浮水印結果出來。

3.2 剪裁

在圖 7 (a)、(b)所示，我們試著對嵌入

後的影像做不同效果的 50%剪裁。在我們的方法中,浮水印是被分散地嵌入在 LH 頻帶之中，因此剪裁若超過 50%比例的圖形會變得毫無意義且取出之浮水印將很難以肉眼辨認。

3.3 尺寸調整

在尺寸調整攻擊裡,分別做 200%、50%的尺寸大小調整以及擴散、角度轉置等四種破壞。

(a) **(b)**

圖 8. (a) 放大 200% (b) 縮小 50%

在圖 8 (a) 及 (b) 分別是將含有浮水印的影像做 200%及 50%尺寸大小調整後萃取浮水印資訊的結果，在經過此二種尺寸大小破壞後所取出的浮水印影像仍可以清楚地用肉眼辨識出來。而在圖 9 (b) 和 (d) 所示,則為對含有浮水印的影像做影像對外擴散 100%的程度以及角度 3° 轉置後所萃取出浮水印的結果，在所萃取的結果可以明顯的看出浮水印影像仍可以輕易的辨識出來。因此，在這個浮水印的演算法有足夠的強韌性來抵抗以上四種影像尺寸破壞的攻擊。

3.4 Filtering

在這個影像破壞測試中，我們將含有浮水印的影像做如圖 10(a)和(c)的 1.5 強度的高斯模糊以及半徑 80、250%區塊強度的邊緣銳利化破壞，在這兩種破壞後萃取出來的浮水印影像如圖 10 (b) 和 (d) ，所萃取出來的浮水印影像仍然可以用肉眼直接辨識出來，表示本方法可以有效抵抗以上兩種破壞攻擊。

表 2. 浮水印取出方法比較

方法 / 萃取條件	[1]	[4]	[11]	Our Scheme
原始圖	●		●	
原始浮水印	●			
嵌入點索引記錄		●		
回合嵌入			●	
嵌入頻率域	●		●	●
金鑰取出		●		●
金鑰長度變化大				●
浮水印大小不固定				●

3.5. 相關方法比較

表 2 顯示我們的方法與[11]、[14]以及[1]所做的比較，在我們的方法中只需要一把隨著原圖與浮水印關聯而隨機變動長度所產生的私密金鑰來做浮水印萃取的動作，而方法[11]將浮水印嵌入在 LL_3 的頻帶之中,[4]則是改良[11]將浮水印嵌入在空間域之中，並且需要記錄先前所嵌入的位置索引來做取出的動作。在方法[11]中，浮水印嵌入及取出時效能較其它方法為差。而方法[4]雖然較[11]的效能高,但是需要記錄先前將浮水印嵌入浮水印嵌入位置的索引值所產生一把私密金鑰。最後方法[1]裡，浮水印的嵌入及取出都需要用到原圖及原始浮水印的重要係數。而在我們的演算法中，只需要私密金鑰即可以達到取出的效

果，因此我們提供了一種較為簡易且可以達到相同近似的浮水印技術。

4. 結論

在本文所提出的浮水印技術步驟是非常的簡潔有效，且其金鑰長度是隨著 DWT 轉換的次數及原始影像和浮水印影像之間的關聯，而有隨機不固定的特性。並且在實驗中證明了可以有效抵抗許多影像破壞的攻擊，相較其它的浮水印技術是較有效且有更好的特色。

誌謝

感謝國家科學委員會對於本研究的支持,相關編號為 NSC95-2221-E-218-030 及 TWISC@NCKU NSC94-3114-P-006-1-Y。

參考文獻

[1] A. Adhipathi Reddy, & B. N. Chatterji (2005). A new wavelet based logo-watermarking scheme. *Pattern Recognition Letters*, 26, 1019-1027.

[2] A. G. Bor. , & I. Pitas (1996). IMAGE WATERMARKING USING DCT DOMAINCONSTRAINTS. *Image Processing, International Conference*, 3, 231-234.

[3] E. J. Stollnitz, A. D. DeRose, & D. H. Salesin,.(1995). Wavelets for computer graphics: a primer. *Computer Graphics and Applications, IEEE*, 15, 76-84, 1995.

[4] J. L. Liu, D. C. Lou, M. C. Chang, & H. K. Tso (2005). A robust watermarking scheme using self-reference image. *Computer Standards & Interfaces*, 28, 356-367.

[5] J. M. Sapiro, (1993). Embedded image coding using zero-trees of wavelet coefficients.*Signal Processing IEEE Transactions*, 41, 3445-3462.

[6] JR. Hernandez, Amado, & M. F. Perez-Gonzalez, (2000). DCT-domain watermarking techniques for still images: detector performance analysis and a new structure. *Image Processing, IEEE Transaction*, 9, 55-68.

[7] M. Eyadat, & S. Vasikarla (2005). Performance evaluation of an incorporated DCT block-based watermarking algorithm with human visual system model. *Pattern Recognition Letters*, 26, 1405-1411.

[8] M. J. Tsai (2004). Filter Bank Selection for theOwnership Verification of Wavelet Based Digital Image Watermarking. *Image Processing, International Conference*, 5, 3415-3418.

[9] M. J. Tsai, & H. Y. Hung (2002). Wavelet Transform Based Digital Watermarking for Image Authentication. *Electrical and Computer Engineering, IEEE CCECE*, 2, 879-884.

[10] R. G.van Schyndel, A.Z.Tirkel, & C.F.Osborne (1994). A DIGITAL WATERMARK. I*mage Processing, Proceedings, International Conference*, 2, 86-90.

[11] S. Joo, Y. Suh, J. Shin, H. Kikuchi, & S. J. Cho (2002). A new robust watermark embedding into wavelet DC components. *ETRI Journal*, 24, 401-404.

[12] S. Pereira, S. Voloshynoskiy, & T. Pun

(2001). Optimal transform domain watermark embedding via linearprogramming. *Signal Processing*, 81, 1251-1260.

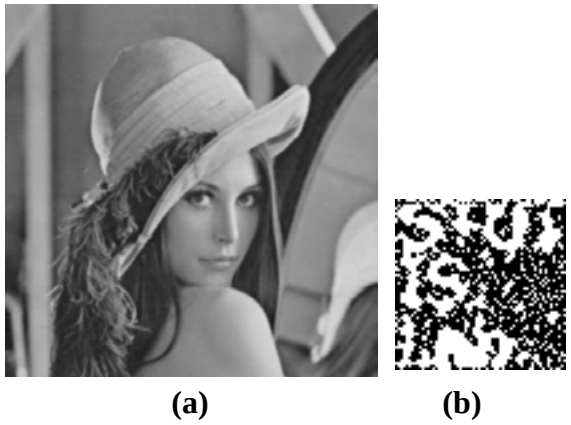
[13] S. Suthaharan, S. W. Kim, H. K. Lee, & S. Sathananthan (2000). Perceptually tuned robustwatermarking scheme for digital images. *Pattern Recognition Letters*, 21, 145-149.

[14] W. Dietl, P. Meerwald, & A. Uhl (2003). Protection of wavelet-based watermarking systems using filter parameterization. *Signal Processing*, 83, 2095-1226.

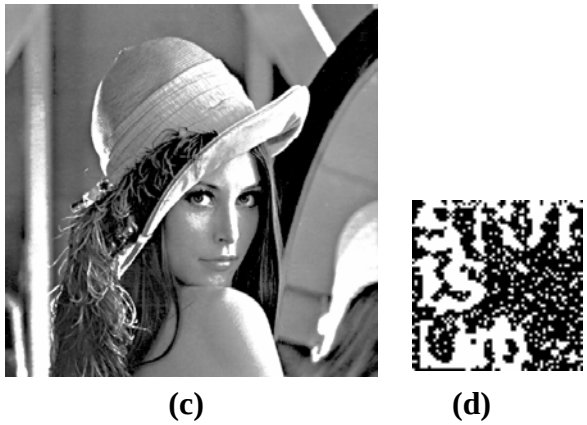
(a) **(b)**

(c) **(d)**

圖 9. (a) 向外擴散 (b) 向外擴散萃取之浮水印
(c) 轉置 3° (d) 轉置萃取之浮水印



(a) **(b)**



(c) **(d)**

圖 10.(a) 模糊 (b) 模糊萃取之浮水印
(c) 邊緣銳化 (d) 邊緣銳化萃取之浮水印