Remerciements

Je tiens à exprimer ma profonde gratitude à Monsieur le professeur DJAMEL CHIKOUCHE pour m'avoir diriger et orienter, pour ces encouragements et ces conseils si précieux qui ont été déterminants pour l'élaboration de ce projet.

J'associe aussi mes remerciements à Monsieur le professeur R.E.BEKKA d'avoir accepter la présidence de ce jury.

Mes remerciements s'adressent également aux membres de jury Messieurs : le professeur N.BOUZIT, Docteur N. AMARDJIA, et Docteur S.BOUGUEZEL, d'avoir exprimé leur volonté d'examiner ce travail.

De même, j'exprime ma gratitude et remerciements aux compagnons du parcours Docteur L.ZIET et Monsieur F.RADJAH, pour l'aide précieuse qu'ils m'ont apportée, pour l'appui moral et scientifique qu'ils m'ont accordé.

Je n'oublie pas de remercier Monsieur : A.KOUNFOUEH et T. SELMANE de leur aide et soutien lors de l'élaboration de ce manuscrit.

En fin, à tout le personnel de l'institut spécialisé de formation professionnelle de Sétif, à tous ceux qui m'ont aidé de prés ou de loin , je leur dit merci .

TOUFIK BEKKOUCHE

5. Attaques:

Les attaques tiennent une place très importante dans le cahier des charges d'un processus de tatouage puisqu'elles définissent la robustesse d'un système. Une façon simple de classer les attaques serait de quantifier les dégradations qu'elles font subir à l'image. Ceci permet de vérifier directement que le cahier des charges est respecté. Nous avons vu ci dessus les problèmes posés par la caractérisation de la qualité d'une image. Nous n'avons pas pour le moment de critère assez performant pour classer les attaques de cette façon.

Attaques sur l'implémentation	Attaques sur la détection
cropping	applications affines
filtrage	ajout de bruit
compression	jitter attack
débruitage	passage à l'analogique
moyennage	Stirmark
impasse	Unsign
	mosaïque
	collusion
	surmarquage
	copiage
	fausses alarmes naturelles

Figure 4 : Classification des attaques



IV.1- Introduction

La transformée en ondelettes discrète TWD (de l'anglais : DWT discret wavelet transform), est une technique inventée par Jean Morlet, un ingénieur Français pour la prospection pétrolière. [ ] C'est une technique qui nous permettre aussi le passage du domaine spatial au domaine fréquentiel.

Si la transformée DCT donne des informations sur l'amplitude des fréquences, la transformée par ondelettes DWT donne l'amplitude de la fréquence à un endroit donné, ce qui est plus proche de la perception humain. Dans le cas d'une image de résolution 2^K, la transformée par ondelettes DWT donne quatre sous images de résolution 2^{K-1}.

Dans ce chapitre, nous allons présenter une étude du tatouage des images à base de la transformée en ondelettes DWT, ainsi que les résultats de test de performances de cette technique sur un certain nombre d'images fixes.

Notre étude sera limitée à la transformée en ondelettes de HAAR.

IV.2-Les ondelettes de HAAR

IV.2.1-Procedure de calcul

Pour procéder à cette transformation, on distingue quatre étapes différentes

1) Moyenner les pixels de l'image originale deux a deux suivant l'axe horizontal :

$$H(x)= (X_{n} + X_{n+1}) / 2 \quad \textbf{(IV.1)}$$

On obtient des fréquences deux fois plus basses.

2) Calculer l'erreur entre l'image originale et l'image sous échantillonnée dans le sens horizontal .

$$G(x)= (X_{n} - X_{n+1}) / 2 \quad \textbf{(IV.2)}$$

3) Pour chacune des deux images intermédiaires, moyenner les pixels deux a deux suivant l'axe vertical.

$$H(y)= (Y_{n} + Y_{n+1}) / 2 \quad \textbf{(IV.3)}$$

4) Pour chacune des deux images intermédiaires, calculer l'erreur suivant l'axe vertical.

$$G(y)= (Y_{n} - Y_{n+1}) / 2 \quad \textbf{(IV.4)}$$

L'application de cette procédure de HAAR sur l'image du chat de la figure IV.1 donne la transformée de HAAR à différents niveaux représentée sur les figures IV.2 ,IV.3 , IV.4 .

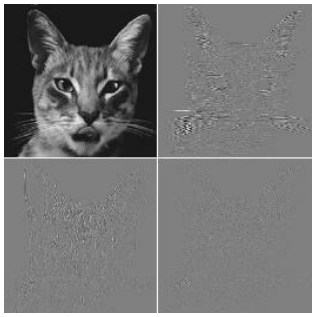


Figure IV.2 : Transformation DWT (niveau 1).

Le résultat de l'application de la procédure de HAAR au niveau1 est une image d'approximation qui a une résolution divisée par deux et trois images de détails qui donnent les hautes fréquences de l'image originale.

Cette transformation est répétée autant de fois que nécessaire pour obtenir le nombre voulu de sous bandes.

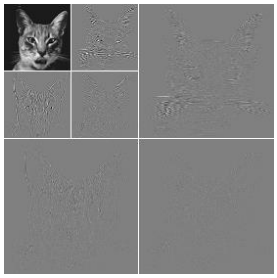


Figure IV.3 : Transformation DWT de HAAR (niveau 2) de l'image chat

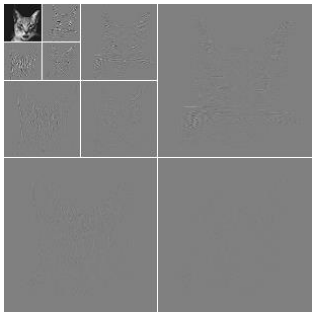


Figure IV.4 : Transformation DWT de HAAR (niveau 3) de l'image chat

Remarque :

Dans le cas de la transformée DCT, les fréquences varient de façon linéaire, tandis que dans le cas de la transformée en ondelettes DWT les fréquences varient de manière logarithmique.

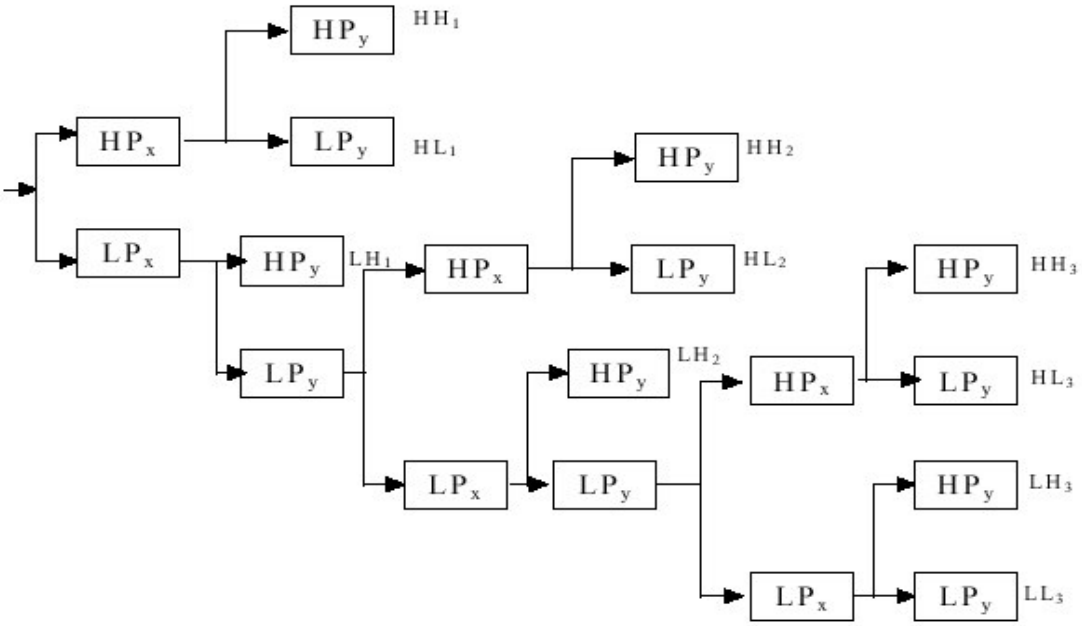


Figure IV.5 : Procédure de décomposition de Haar au niveau 3.

IV.3- Exemple de calcul de la transformée DWT par la méthode de HAAR

Pour bien éclaircir les étapes décrites dans la section précédente pour le calcul de la transformée de HAAR par DWT, prenons

l'exemple de la figure ci-dessous, et appliquant les quatre étapes sur un bloc de pixels 8x8

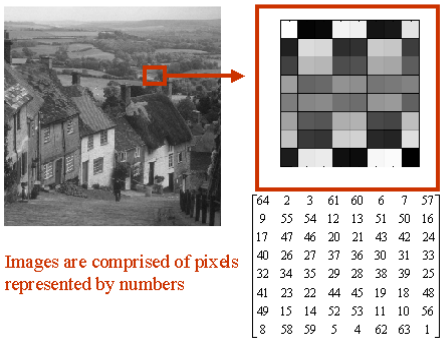


Figure IV.6: Un bloc de pixel tiré d'une image.

Etape1) : calculons les moyennes des pixels du bloc 8x8 de la figure

Nous obtenons :

$$B_1 = \begin{bmatrix} 33 & 32 & 33 & 32 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 33 & 32 & 33 & 32 & \dots \\ 33 & 32 & 33 & 32 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 33 & 32 & 33 & 32 & \dots \end{bmatrix}$$

Etape2) : calculons l'erreur entre le bloc original est le bloc sous échantillonné dans le sens horizontal :les résultats obtenus vont remplacés les points

$$\begin{bmatrix} 33 & 32 & 33 & 32 & 31 & -29 & 27 & -25 \\ 32 & 33 & 32 & 33 & -23 & 21 & -19 & 17 \end{bmatrix}$$

$$B_2 = \begin{matrix} 32 & 33 & 32 & 33 & -15 & 13 & -11 & 09 \\ 33 & 32 & 33 & 32 & 07 & -5 & 3 & -1 \\ 33 & 32 & 33 & 32 & -1 & 03 & -5 & 07 \\ 32 & 33 & 32 & 33 & 09 & -11 & 13 & -15 \\ 32 & 33 & 32 & 33 & 17 & -19 & 21 & -23 \\ 33 & 32 & 33 & 32 & -25 & 27 & -29 & 31 \end{matrix}$$

Etape3) : calculons les moyennes des pixels du bloc 8x8 résultant dans le sens vertical

$$\Big|$$

$$B_3 = \begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \end{bmatrix}$$

Etape4) : calculons l'erreur dans le sens vertical

$$\Big|$$

$$\begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \end{bmatrix}$$

$$B_4 = \begin{matrix} 0.5 & -0.5 & 0.5 & -0.5 & 27 & -25 & 23 & -21 \\ -0.5 & 0.5 & -0.5 & 0.5 & -11 & 09 & -7 & 05 \\ 0.5 & -0.5 & 0.5 & -0.5 & -5 & 07 & -9 & 11 \\ -0.5 & 0.5 & -0.5 & 0.5 & 21 & -22 & 25 & -27 \end{matrix}$$

B_4 : représente la matrice transformée DWT du bloc B_1

Le processus est réversible, on peut calculer la IDWT(la transformée par ondelettes discrète inverse)qui va nous permettre de revenir au domaine spatial une fois notre traitement est terminé.

Dans le sens horizontal :

$X_n = H(x) + G(x)$ …….4.5

$X_{n+1} = H(x) - G(x)$ …….4.6

Dans le sens vertical :

$H(y) = H(y + G(y)$ …….4.7

$G(y) = H(y)G(y)$ ……4.8

3- Algorithme de tatouage utilisant la DWT :

Notre algorithme est articlée sur le principe suivant :

3.1 Phase d'insertion : Soit a insérer une marque désignée par (w), dans une image originale désignée par (P_O).

Soient g : les coefficients par la transformée DWT du niveau (1) de l'image originale (P_O) selon la figure ci dessous





Figure6: image originale (P_O) et sa transformée par DWT

w : sont les coefficients de la marque (P_w) tels qu'ils sont (sans subir de transformation).

L'insertion se fait suivant la formule de tatouage suivante :

$$g^* = g + \alpha \cdot w \quad \ldots\ldots\ldots\ldots 4.9$$

Ou g* : l'image tatouée dans le domaine fréquentiel

α : la force de tatouage

La figure ci-dessous illustre bien cette phase

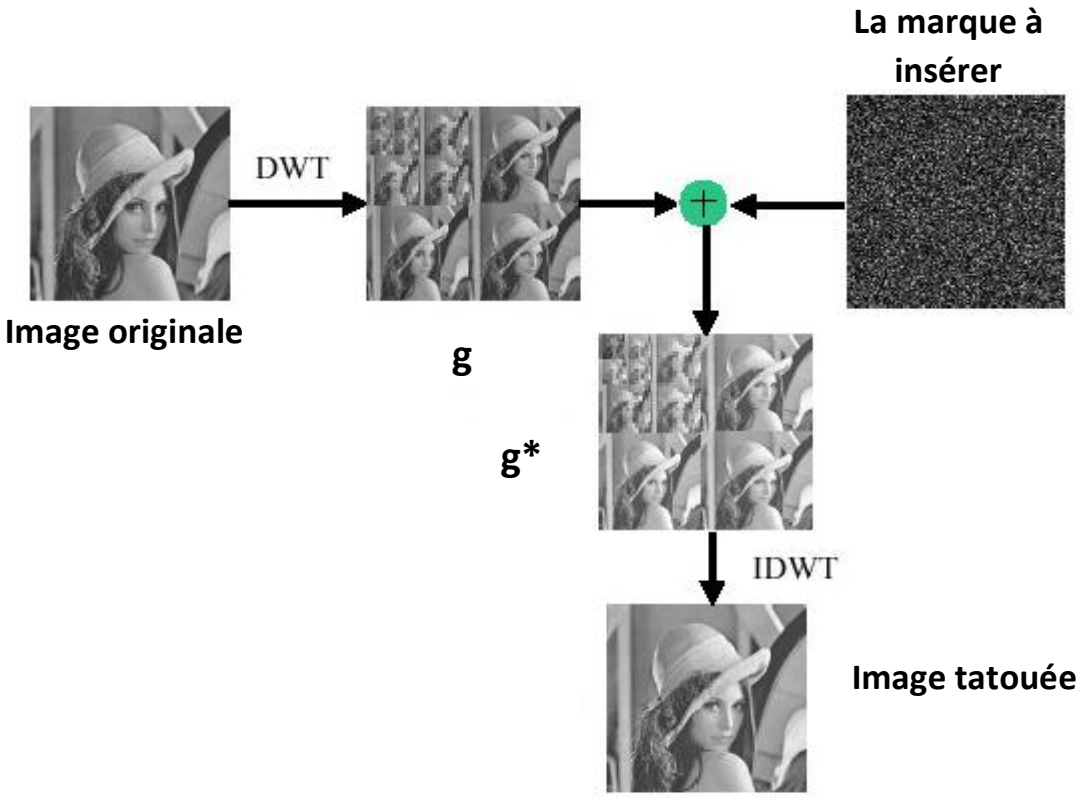


Figure 7 :Insertion de la marque

Pour reconstruire l'image tatouée dans le domaine spatial,on utilise la transformée par ondelettes inverse (IDWT)

$$P_w = IDWT(g^*)$$

3.2 Organigramme d'insertion :

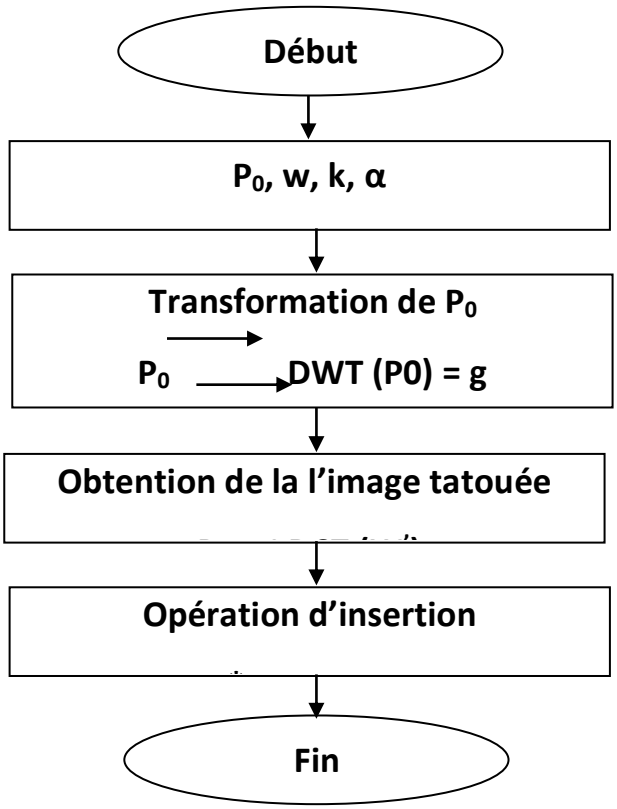


Figure 8 : Organigramme d'insertion de la marque

3.3 Phase d'extraction

Pour l'extraction on dispose des paramètres suivants :

L'image originale (P_o)

L'image tatouée (P_w)

La force de tatouage :α

L'endroit ou l'emplacement de la marque dans l'image originale

L'algorithme d'extraction se fait suivant le schéma suivant :

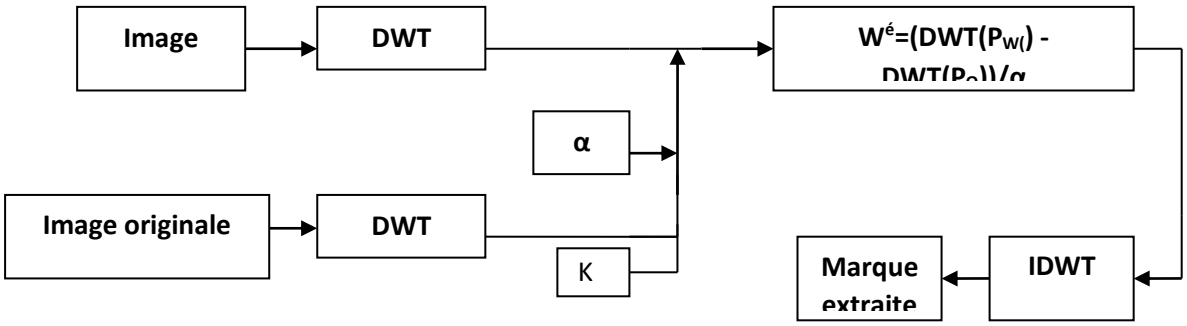


Figure 9 : Schéma d'extraction de la marque par DWT

On applique la DWT a (P_o) et (P_w), ce qui donne g et g*,la marque extraite est déduite a partir de la formule suivante :

$$W' = (g^* - g) / \alpha \quad \ldots\ldots\ldots\ldots\ldots\ldots 4.10$$

Pour avoir la marque reconstruite,on applique la IDWT

w'= IDWT (W')

Remarque : On doit différencier entre w qui est la marque originale, et

w'qui est la marque reconstruite.

3.4 Organigramme d'extraction :

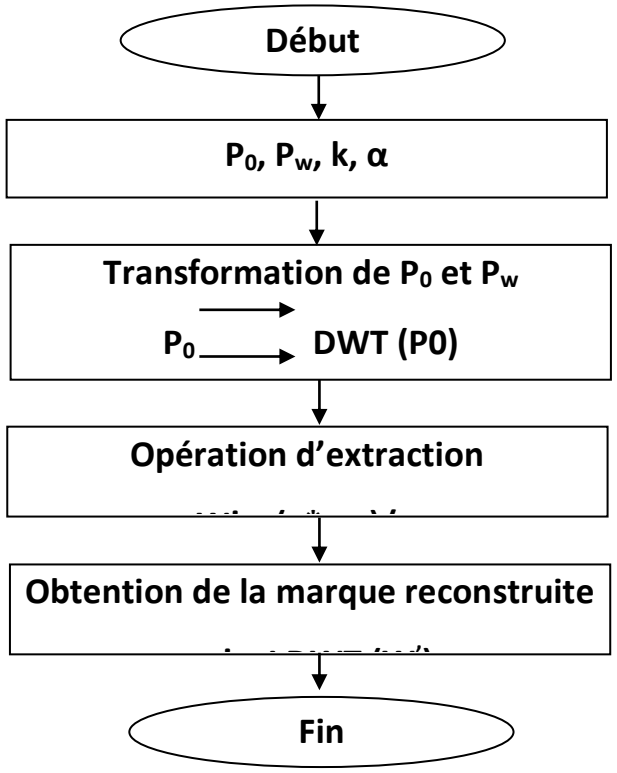


Figure 8 : Organigramme d'extraction de la marque

IV.6- Test de la méthode de tatouage par DWT

La méthode de tatouage par DWT a été implémentée sur Matlab 7.1.

Pour pouvoir faire une comparaison fiable, on doit garder le même environnement (images) utilisé dans le cas de la DCT.

(Po) est l'image originale de (Lena) de taille 256x256 utilisée au format bmp. La marque (w) à insérer est de taille 64x64, utilisée au format bmp, elle porte mon nom et prénom et ma willaya.



Figure IV. 12 : Image originale



Figure IV.13 : Marque à insérer



Figure 11 : Image tatouée pour α=0.05



Figure 12 : Image tatouée compressée pour α=0.05



Figure 12 : marque extraite pour α=0.05



Figure 13 : marque extraite après compression pour α=0.05



Figure 14 : Image tatouée pour α=0.1



Figure 15 : Image tatouée compressée pour α=0.1



Figure 16 : marque extraite pour α=0.1



Figure 17 : marque extraite après compression pour α=0.1



Figure 18 : Image tatouée pour α=0.5



Figure 19 : Image tatouée compressée pour α=0.5



Figure 20 : marque extraite pour α=0.5



Figure21 : marque extraite après compression pour α=0.5

Pour mesurer la robustesse de ce tatouage, on calcule le : $PSNR_1$

Entre l'image originale et la marque originale

Pour α=0. 5

Sans attaque : $PSNR_1$=14.0186dB

Attaque avec compression jpeg $PSNR_1$=13.9667dB

Pour α=0.05

Sans attaque : $PSNR_1$=32.6016dB

Attaque avec compression jpeg $PSNR_1$=30.1239dB

Pour α=0.1

Sans attaque : $PSNR_1$=26.7483dB

Attaque avec compression jpeg $PSNR_1$=31.3747dB

Pour α=0.2

Sans attaque : $PSNR_1$=20.8177dB

Attaque avec compression jpeg $PSNR_1$=20.5053dB

Ainsi que le $PSNR_2$ entre la marque originale et la marque extraite

Pour α=0. 5

Sans attaque : $PSNR_2$=15.2609dB

Attaque avec compression jpeg $PSNR_2$=14.6924dB

Pour α=0.05

Sans attaque : $PSNR_2=\infty$dB

Attaque avec compression jpeg $PSNR_2$=12.5638dB

Pour α=0.1

Sans attaque : $PSNR_2$=57.6080dB

Attaque avec compression jpeg $PSNR_2$=15.0381dB

Pour α=0.2

Sans attaque : $PSNR_2$=31.1844dB

Attaque avec compression jpeg $PSNR_2$=21.8773dB

Commentaires :**1)** on remarque d'après ces résultats que la marque est

Invisible et commence a devenir visible a partir de

α =0.5, ce qui qualifie ce tatouage d'invisible.

2) La plus grande robustesse est obtenu pour α=0.05

PSNR= 32.6016, donc peu faible devant celui obtenu

avec la DCT ce qui qualifie ce tatouage de fragile

3) La DWT est un peu fragile devant une compression

jpeg par rapport a la DCT

Résultat : Le tatouage par la transformée DWT est un tatouage fragile et invisible

5) Comportement du tatouage vis-à-vis les attaques géométriques

5.1) Attaque de rotation :



Figure22 : image tatouée qui a subi une rotation de 15^0 (α=0. 5)

PSNR=8.9950 dB



Figure22-a : marque extaite aprés une rotation de 15^0 (α=0. 5)



Figure23 : image tatouée qui a subi une rotation de 15^0 (α=0.1)



Figure23-a : marque extaite aprés une rotation de 15^0 (α=0. 1)

PSNR=9.5955 dB



Figure24 : image tatouée qui a subi une rotation de 15^0 (α=0.05)

PSNR=9.6277 dB



Figure24-a : marque extaite aprés une rotation de 15^0 (α=0. 05)

Pour (α=0.2) PSNR=9.4759

5.2 Attaque de redimensionnement (resizing)



Figure25 : image tatouée qui a subi un resizing de1.25 (α=0.05)

PSNR=12.2544 dB



Figure25-a : marque extaite aprés un resizing de 1.25 (α=0. 05)



Figure26 : image tatouée qui a subi un resizing de1.25 (α=0.1)

PSNR=12.2038 dB



Figure26-a : marque extraite après un resizing de 1.25 (α=0. 1)



Figure27 : image tatouée qui a subi un resizing de1.25 (α=0.5)

PSNR=11.2924 dB



Figure26-a : marque extraite après un resizing de 1.25 (α=0. 5)

5.3 Attaque de cropping



Figure25 : image tatouée qui a subi un cropping (α=0.5)

Image bmp(107x201)

PSNR=9.9127 dB



Figure26 : image tatouée qui a subi un cropping (α=0.1)

Image bmp(84x194)

PSNR=12.0463dB



Figure27 : image tatouée qui a subi un cropping (α=0.05)

Image bmp(84x217)

PSNR=11.8589dB

5.4 Attaque de filtrage (passe-bas)



Figure28 : image tatouée qui a subi un filtrage (α=0.5)

PSNR=21.8613dB

	α = 0,05	α = 0,1	α = 0,2	α = 0,5
Sans attaques	32.6016	31.3747	20.8177	14.0186
Compression JPEG	30.1239	26.7484	20.5053	13.9667
Redimensionnement	12.2544	12.2038		11.2924
Cropping	11.8589	12.6463		9.9127
Filtrage passe-bas				21.8613
Rotation	9.6277	9.5955	9.4759	8.9950

Tableau recapitulatif du PSNR en dB vis à vis les différentes attaques

D’après le tableau suivant, on remarque que le tatouage par DWT insensible a la compression jpeg et filtrage passe-bas mais il est moyennement sensible pour les attaques géométriques de redimensionnement, de rotation et de cropping en se basant bien sure sur les valeurs du PSNR

6. Conclusion : Dans ce chapitre, on a acquit une expérience simple mais très significative dans le domaine de traitement d’image, elle nous a permis d’améliorer nos connaissances dans le domaine des transformées, et même de développer nos taxonomies d’analyse et de synthèse dans la recherche scientifique



IV.1- Introduction

La transformée en ondelettes discrète TWD (de l'anglais : DWT discrete wavelet transform), est une technique inventée par Jean Morlet, un ingénieur Français pour la prospection pétrolière. [ ] C'est une technique qui nous permet aussi le passage du domaine spatial au domaine fréquentiel.

Si la transformée DCT donne des informations sur l'amplitude des fréquences, la transformée par ondelettes DWT donne l'amplitude de la fréquence à un endroit donné de l'axe du temps, ce qui est plus proche de la perception humaine. Dans le cas d'une image de résolution 2^K, la transformée en ondelettes DWT donne quatre sous images de résolution 2^{K-1}.

Dans ce chapitre, nous allons présenter une étude du tatouage des images à base de la transformée en ondelettes DWT, ainsi que les résultats de test de performances de cette technique sur un certain nombre d'images fixes.

Notre étude sera limitée à la transformée en ondelettes de HAAR.

IV.2-Les notions de la transformée en ondelettes

IV.3-Les ondelettes de HAAR

IV.3.1-Définition de la transformée de Haar

IV.3.2-Procédure de décomposition de la transformée de Haar

IV.3.1-Procedure de calcul

Pour procéder à cette transformation, on distingue quatre étapes différentes

2) Moyenner les pixels de l'image originale deux a deux suivant l'axe horizontal :

$$H(x)= (X_n + X_{n+1}) / 2 \quad \textbf{(IV.1)}$$

On obtient des fréquences deux fois plus basses.

2) Calculer l'erreur entre l'image originale et l'image sous échantillonnée dans le sens horizontal .

$$G(x)= (X_n - X_{n+1}) / 2 \quad \textbf{(IV.2)}$$

3) Pour chacune des deux images intermédiaires, moyenner les pixels deux a deux suivant l'axe vertical.

$$H(y)= (Y_n + Y_{n+1}) / 2 \quad \textbf{(IV.3)}$$

4) Pour chacune des deux images intermédiaires, calculer l'erreur suivant l'axe vertical.

$$G(y)= (Y_n - Y_{n+1}) / 2 \quad \textbf{(IV.4)}$$

L'application de cette procédure de HAAR sur l'image du chat de la figure IV.1 donne la transformée de HAAR à différents niveaux représentée sur les figures IV.2, IV.3, IV.4.

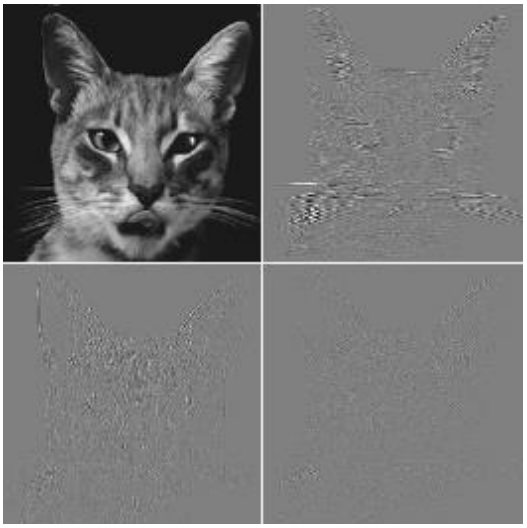


Figure IV.2 : Transformation DWT (niveau 1).

Le résultat de l'application de la procédure de HAAR au niveau1 est une image d'approximation qui a une résolution divisée par deux et trois images de détails qui donnent les hautes fréquences de l'image originale.

Cette transformation est répétée autant de fois que nécessaire pour obtenir le nombre voulu de sous bandes.

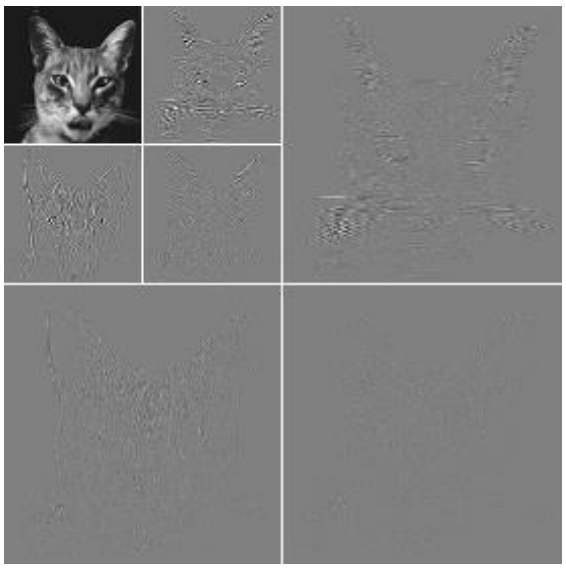


Figure IV.3 : Transformation DWT de HAAR (niveau 2) de l'image chat

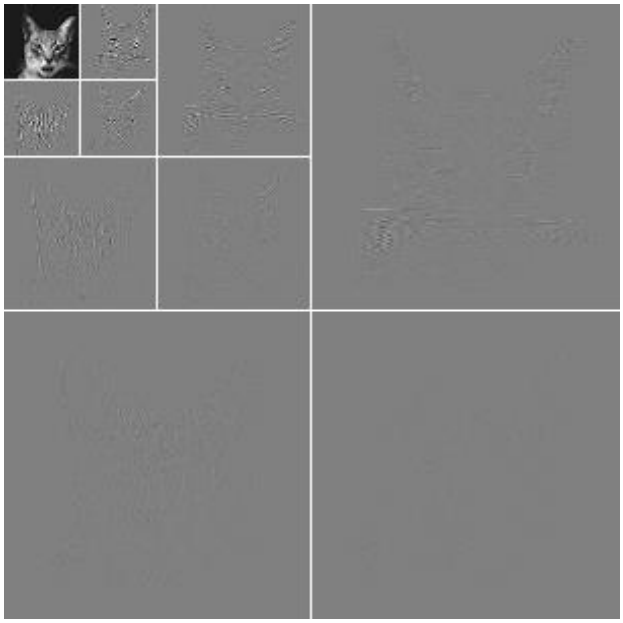


Figure IV.4 : Transformation DWT de HAAR (niveau 3) de l'image chat.

Remarque :

Dans le cas de la transformée DCT, les fréquences varient de façon linéaire, tandis que dans le cas de la transformée en ondelettes DWT les fréquences varient de manière logarithmique.

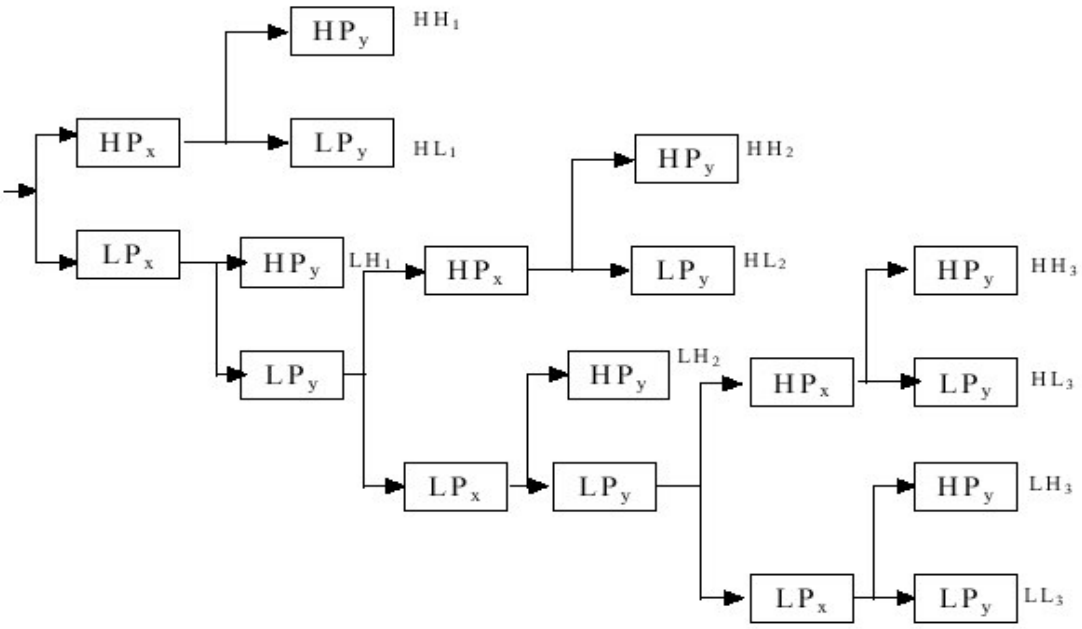


Figure IV.5 : Procédure de décomposition de Haar au niveau 3.

IV.4- Exemple de calcul de la transformée DWT par la méthode de HAAR

Pour bien éclaircir les étapes décrites dans la section précédente pour le calcul de la transformée de HAAR par DWT, prenons l'exemple de la figure ci-dessous, et appliquant les quatre étapes sur un bloc de pixels 8x8

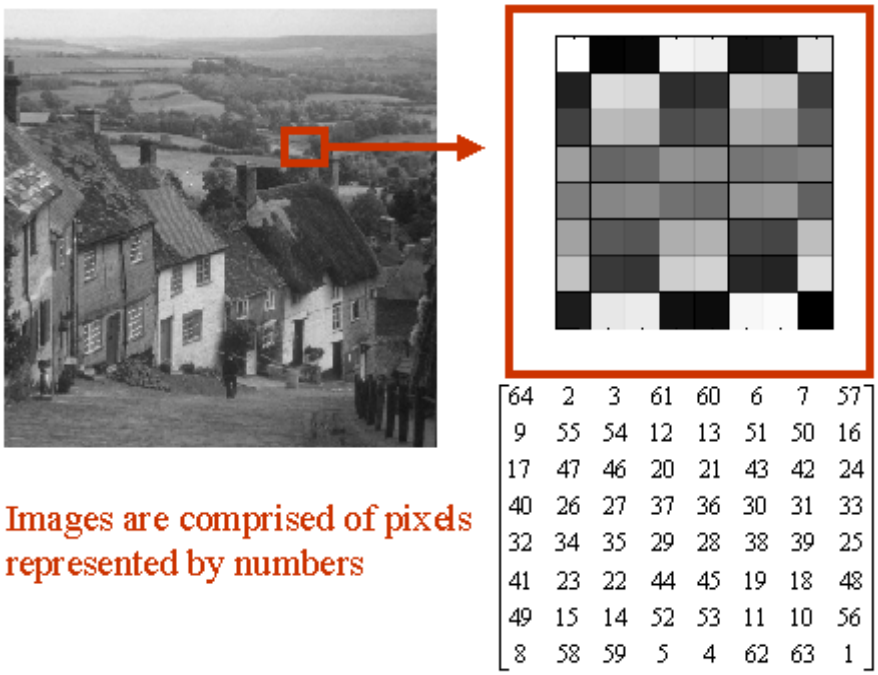


Figure IV.6: Un bloc de pixel tiré d'une image.

Etape1) : calculons les moyennes des pixels du bloc 8x8 de la figure

Nous obtenons :

$$B_1 = \begin{bmatrix} 33 & 32 & 33 & 32 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 33 & 32 & 33 & 32 & \dots \\ 33 & 32 & 33 & 32 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 32 & 33 & 32 & 33 & \dots \\ 33 & 32 & 33 & 32 & \dots \end{bmatrix}$$

Etape2) : calculons l'erreur entre le bloc original est le bloc sous échantillonné dans le sens horizontal :les résultats obtenus vont remplacés les points

$$B_2 = \begin{bmatrix} 33 & 32 & 33 & 32 & 31 & -29 & 27 & -25 \\ 32 & 33 & 32 & 33 & -23 & 21 & -19 & 17 \\ 32 & 33 & 32 & 33 & -15 & 13 & -11 & 09 \\ 33 & 32 & 33 & 32 & 07 & -5 & 3 & -1 \\ 33 & 32 & 33 & 32 & -1 & 03 & -5 & 07 \\ 32 & 33 & 32 & 33 & 09 & -11 & 13 & -15 \\ 32 & 33 & 32 & 33 & 17 & -19 & 21 & -23 \\ 33 & 32 & 33 & 32 & -25 & 27 & -29 & 31 \end{bmatrix}$$

Etape3) : calculons les moyennes des pixels du bloc 8x8 résultant dans le sens vertical

$$B_3 = \begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \end{bmatrix}$$

Etape4) : calculons l'erreur dans le sens vertical

$$B_4 = \begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 0.5 & -0.5 & 0.5 & -0.5 & 27 & -25 & 23 & -21 \\ -0.5 & 0.5 & -0.5 & 0.5 & -11 & 09 & -7 & 05 \\ 0.5 & -0.5 & 0.5 & -0.5 & -5 & 07 & -9 & 11 \\ -0.5 & 0.5 & -0.5 & 0.5 & 21 & -22 & 25 & -27 \end{bmatrix}$$

B_4 : représente la matrice transformée DWT du bloc B_1

Le processus est réversible, on peut calculer la IDWT(la transformée par ondelettes discrète inverse)qui va nous permettre de revenir au domaine spatial une fois notre traitement est terminé.

Dans le sens horizontal :

$$X_n = H(x) + G(x) \qquad \textbf{(IV.5)}$$

$$X_{n+1} = H(x) - G(x) \qquad \textbf{(IV.6)}$$

Dans le sens vertical :

$$H(y) = H(y + G(y) \qquad \textbf{(IV.7)}$$

G(y)= H(y)G(y) **(IV.8)**

IV.5- Algorithme de tatouage utilisant la DWT :

Notre algorithme est articlée sur le principe suivant :

IV.5.1- Phase d'insertion : Soit a insérer une marque désignée par (w), dans une image originale désignée par (P_O).

Soient g : les coefficients par la transformée DWT du niveau (1) de l'image originale (P_O) selon la figure ci-dessous



Figure6: image originale (P_O) et sa transformée par DWT

w : sont les coefficients de la marque (P_w) tels qu'ils sont (sans subir de transformation).

L'insertion se fait suivant la formule de tatouage suivante :

$$g^* = g + \alpha . w \quad \ldots\ldots\ldots\ldots 4.9$$

Ou g* : l'image tatouée dans le domaine fréquentiel

α : la force de tatouage

La figure ci-dessous illustre bien cette phase

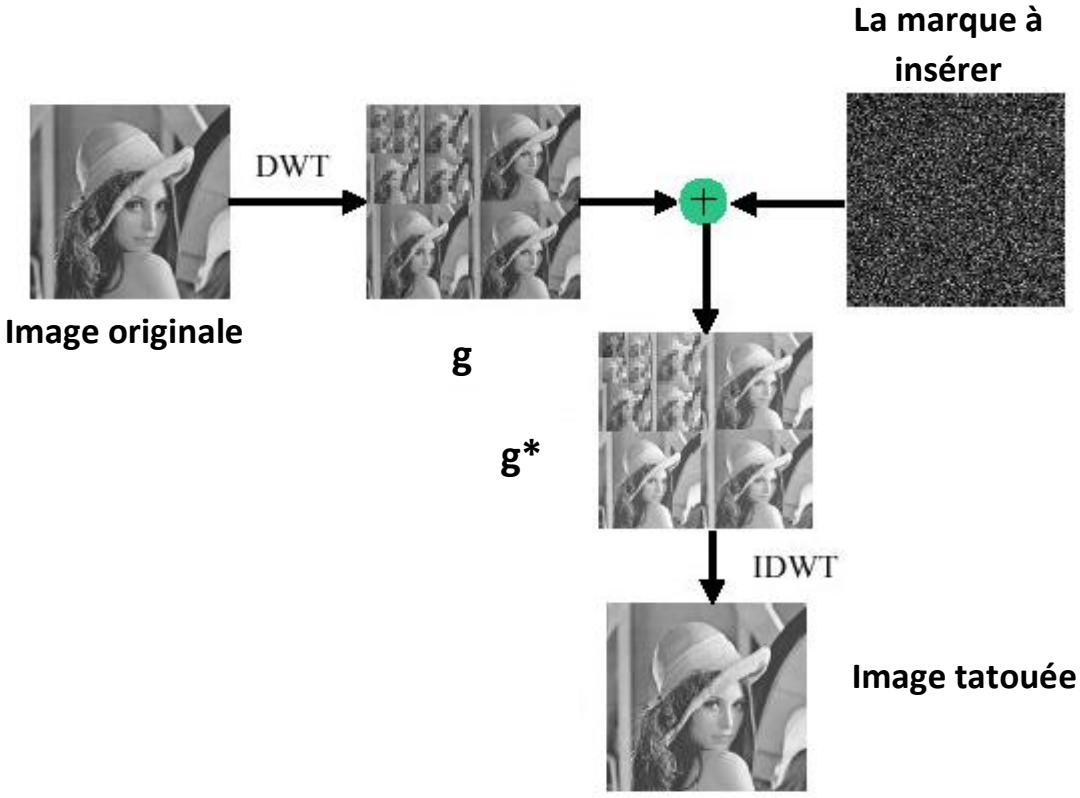


Figure 7 :Insertion de la marque

Pour reconstruire l'image tatouée dans le domaine spatial,on utilise la transformée par ondelettes inverse (IDWT)

$$P_w = IDWT(g^*)$$

IV.5.2- Organigramme d'insertion :

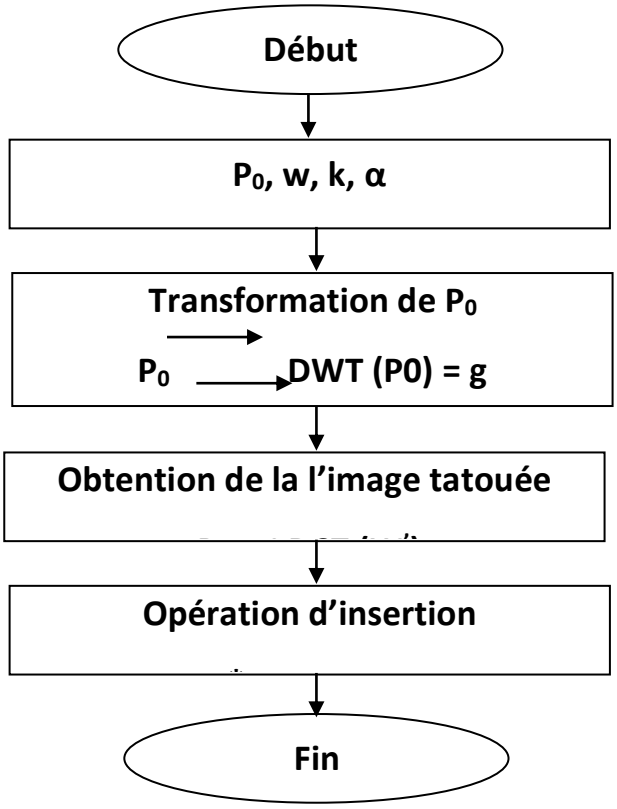


Figure 8 : Organigramme d'insertion de la marque

3.3 Phase d'extraction

Pour l'extraction on dispose des paramètres suivants :

L'image originale (P_o)

L'image tatouée (P_w)

La force de tatouage :α

L'endroit ou l'emplacement de la marque dans l'image originale

L'algorithme d'extraction se fait suivant le schéma suivant :

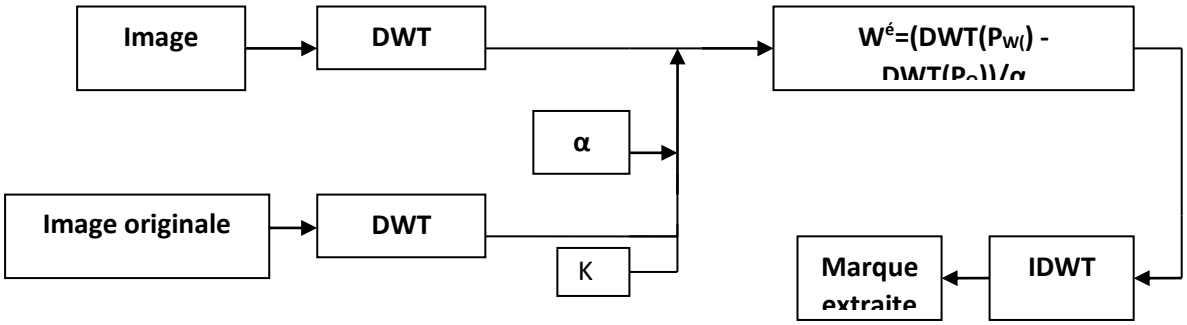


Figure 9 : Schéma d'extraction de la marque par DWT

On applique la DWT a (P_o) et (P_w), ce qui donne g et g*,la marque extraite est déduite a partir de la formule suivante :

$$W' = (g^* - g) / \alpha \quad \ldots\ldots\ldots\ldots\ldots\ldots 4.10$$

Pour avoir la marque reconstruite,on applique la IDWT

$$w' = IDWT(W')$$

Remarque : On doit différencier entre w qui est la marque originale, et

w'qui est la marque reconstruite.

3.4 Organigramme d'extraction :

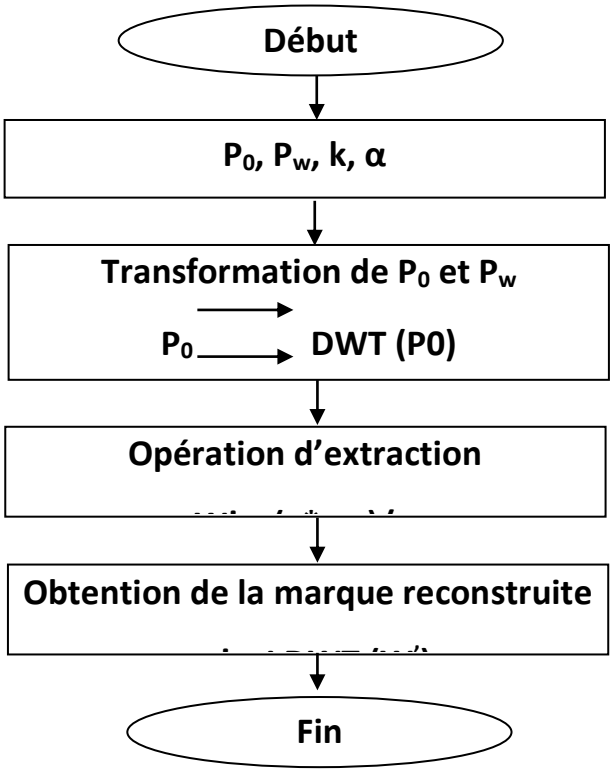


Figure 8 : Organigramme d'extraction de la marque

IV.6- Test de la méthode de tatouage par DWT

La méthode de tatouage par DWT a été implémentée sur Matlab 7.1.

Pour pouvoir faire une comparaison fiable, on doit garder le même environnement (images) utilisé dans le cas de la DCT.

(Po) est l'image originale de (Lena) de taille 256x256 utilisée au format bmp. La marque (w) à insérer est de taille 64x64, utilisée au format bmp, elle porte mon nom et prénom et ma willaya.





Figure IV. 12 : Image originale **Figure IV.13 : Marque à insérer**

IV.6.1- Test de la méthode de tatouage par DWT sans attaques

IV.6.1-Avec α =0.02





Figure IV. 14.1 : Image tatouée avec α =0.02

Figure IV.15.1 : Marque extraite avec α =0.02

$PSNR_1$= 40.9011dB

$PSNR_2$= 77.1236dB

IV.6.1.2-Avec α =0.05





Figure IV.14.2 : Image tatouée

Figure IV.15.2 : Marque extraite

avec α =0.05 **avec α =0.05**

$PSNR_1$= 32.6016dB **$PSNR_2$= 62.1478dB**

IV.6.1.3-Avec α =0.1





Figure IV.14.3 : Image tatouée avec α =0.1

Figure IV.15.3 : Marque extraite avec α =0.1

$PSNR_1$= 26.7483dB **$PSNR_2$= 57.6080 dB**

IV.6.1.4-Avec α =0.2



Figure IV.14.4 : Image tatouée avec α =0.2



Figure IV.15.4 : Marque extraite avec α =0.2

$PSNR_1$= 20.8177dB

$PSNR_2$= 31.1844 dB

IV.6.1.5-Avec α =0.3





Figure IV.14.5 : Image tatouée avec α =0.3

Figure IV.15.5 : Marque extraite avec α =0.3

$PSNR_1$= 17.5927dB

$PSNR_2$= 21.0909 dB

IV.6.1.6-Avec α =0.4





Figure IV.14.6 : Image tatouée avec α =0.4

Figure IV.15.6 : Marque extraite avec α =0.4

$PSNR_1$= 15.3850dB

$PSNR_2$= 17.5735 dB

IV.6.1.7-Avec α =0.5



Figure IV.14.7 : Image tatouée avec α =0.5



Figure IV.15.7 : Marque extraite avec α =0.5

$PSNR_1$= 13.9667dB

$PSNR_2$= 15.2609 dB

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
$PSNR_1$ en dB	409011	32.6016	26.7483	20.8177	17.5927	15.3850	13.9667
$PSNR_2$ en dB	77.1236	62.1478	57.6080	31.1844	21.0909	17.5735	15.2609

Tableau IV.1 : Tableau récapitulatif du $PSNR_1$ et $PSNR_2$ en fonction de α.

IV.6.2-Comportement du tatouage par DWT vis-à-vis les attaques

IV.6.2.A-Attaque par compression JPEG

IV.6.2.A.1-Avec α =0.02



Figure IV.16.1 : Image compressée avec α =0.02

PSNR$_1$= 34.9513dB



Figure IV.17.1 : Marque extraite avec α =0.02

PSNR$_2$= 06.1983 dB

IV.6.2.A.2-Avec α =0.05



Figure IV.16.2 : Image compressée avec α =0.05



Figure IV.17.2 : Marque extraite avec α =0.05

$PSNR_1$= 30.1239dB

$PSNR_2$= 12.5638 dB

IV.6.2.A.3-Avec α =0.1



Figure IV.16.3 : Image compressée avec α =0.1



Figure IV.17.3 : Marque extraite avec α =0.1

$PSNR_1$= 26. 7517dB

$PSNR_2$= 15.0381 dB

IV.6.2.A.4-Avec α =0.2





Figure IV.16.4 : Image compressée avec α =0.2

Figure IV.17.4 : Marque extraite avec α =0.2

$PSNR_1$= 20. 5053dB

$PSNR_2$= 21.8773 dB

IV.6.2.A.5-Avec α =0.3





Figure IV.16.5 : Image compressée avec α =0.3

Figure IV.17.5 : Marque extraite avec α =0.3

$PSNR_1$= 17. 5188dB

$PSNR_2$= 19.8234 dB

IV.6.2.A.6-Avec α =0.4



Figure IV.16.6 : Image compressée avec α =0.4



Figure IV.17.6 : Marque extraite avec α =0.4

$PSNR_1$= 15. 4195dB

$PSNR_2$= 16.9475 dB

IV.6.2.A.7-Avec α =0.5



Figure IV.16.7 : Image compressée avec α =0.5



Figure IV.17.7 : Marque extraite avec α =0.5

$PSNR_1$= 14. 0186dB

$PSNR_2$= 14.6924 dB

IV.6.2.B-Attaque par rotation de 15º

IV.6.2.B.1- Avec α =0.02





Figure IV.18.1 : Image après rotation avec α=0.02

Figure IV.19.1 :Marque extraite avec α=0.02

$PSNR_1$ = 9.6374dB

$PSNR_2$ = 2.1191dB

IV.6.2.B.2- Avec α =0.05





Figure IV.18.2 : Image après rotation avec α=0.05

Figure IV.19.2 :Marque extraite avec α=0.05

$PSNR_1$ = 9.6277dB

$PSNR_2$ = 1.8536dB

IV.6.2.B.3- Avec α =0.1





Figure IV.18.3 : Image après rotation avec α=0.1

Figure IV.19.3 :Marque extraite avec α=0.1

$PSNR_1$ = 9.5955dB

$PSNR_2$ = 1.6811dB

IV.6.2.B.4- Avec α =0.2





Figure IV.18.4 : Image après rotation avec α=0.2

Figure IV.19.4 :Marque extraite avec α=0.2

$PSNR_1$ = 9.4759dB

$PSNR_2$ = 1.5520dB

IV.6.2.B.5- Avec α =0.3





Figure IV.18.5 : Image après rotation avec α=0.3

Figure IV.19.5 :Marque extraite avec α=0.3

$PSNR_1$ = 9.3251dB

$PSNR_2$ = 1.4958dB

IV.6.2.B.6- Avec α =0.4





Figure IV.18.6 : Image après rotation avec α=0.4

Figure IV.19.6 :Marque extraite avec α=0.4

$PSNR_1$ = 9.1480dB

$PSNR_2$ = 1.4610dB

IV.6.2.B.7- Avec α =0.5





Figure IV.18.7 : Image après rotation avec α=0.5

FigureIV.19.7 :Marque extraite avec α=0.5

$PSNR_1$ = 8.9950dB

$PSNR_2$ = 1.4344dB

IV.6.2.C-Attaque par redimensionnement (resizing) de 1.25

IV.6.2.C.1- Avec α =0.02





Figure IV.20.1 : Image après resizing avec α=0.02

Figure IV.21.1 :Marque extraite avec α=0.02

$PSNR_1$ = 12.2695dB

$PSNR_2$ = 1.9596 dB

IV.6.2.C.2- Avec α =0.05





Figure IV.20.2 : Image après resizing extraite

avec α=0.05

$PSNR_1$ = 12.2544dB

Figure IV.21.2 :Marque

avec α=0.05

$PSNR_2$ = 1.7200 dB

IV.6.2.C.3- Avec α =0.1





Figure IV.20.3 : Image après resizing extraite

avec α=0.1

$PSNR_1$ = 12.2038dB

Figure IV.21.3 :Marque

avec α=0.1

$PSNR_2$ = 1.5950 dB

IV.6.2.C.4- Avec α =0.2





Figure IV.20.4 : Image après resizing extraite

Figure IV.21.4 :Marque

avec α=0.2

avec α=0.2

$PSNR_1$ = 12.0051dB

$PSNR_2$ = 1.5101 dB

IV.6.2.C.5- Avec α =0.3





Figure IV.20.5 : Image après resizing extraite **Figure IV.21.5 :Marque**

avec α=0.3 **avec α=0.3**

$PSNR_1$ = 11.7771dB $PSNR_2$ = 1.4716 dB

IV.6.2.C.6- Avec α =0.4





Figure IV.20.6 : Image après resizing extraite **Figure IV.21.6 :Marque**

avec α=0.4 **avec α=0.4**

$PSNR_1$ = 11.5177dB $PSNR_2$ = 1.4439 dB

IV.6.2.C.7- Avec α =0.5



Figure IV.20.7 : Image après resizing avec α=0.5



Figure IV.21.7 :Marque extraite avec α=0.5

$PSNR_1$ = 11.2924dB

$PSNR_2$ = 1.4264 dB

IV.6.2.D-Attaque par filtrage (passe-bas)

IV.6.2.D.1- Avec α =0.02



Figure IV.22.1 : Image après filtrage

avec α=0.02

$PSNR_1$ = 38.5897dB



Figure IV.23.1 :Marque extraite

avec α=0.02

$PSNR_2$ =0.8553 dB

IV.6.2.D.2- Avec α =0.05



Figure IV.22.2 : Image après filtrage

avec α=0.05

$PSNR_1$ = 37.1457dB



Figure IV.23.2 :Marque extraite

avec α=0.05

$PSNR_2$ =4.3954 dB

IV.6.2.D.3- Avec α =0.1





Figure IV.22.3 : Image après filtrage extraite

avec α=0.1

$PSNR_1$ = 34.2322dB

Figure IV.23.3 :Marque

avec α=0.1

$PSNR_2$ =4.3435 dB

IV.6.2.D.4- Avec α =0.2



Figure IV.22.4 : Image après filtrage extraite

avec α=0.2

$PSNR_1$ = 29.5385dB



Figure IV.23.4 :Marque

avec α=0.2

$PSNR_2$ =4.2922 dB

IV.6.2.D.5- Avec α =0.3



Figure IV.22.5: Image après filtrage extraite

avec α=0.3

$PSNR_1$ = 26.2994dB



Figure IV.23.5 :Marque

avec α=0.3

$PSNR_2$ =4.1015 dB

IV.6.2.D.6- Avec α =0.4





Figure IV.22.6: Image après filtrage avec α=0.4

$PSNR_1$ = 23.7848dB

Figure IV.23.6 :Marque extraite avec α=0.4

$PSNR_2$ =3.9176 dB

IV.6.2.D.7- Avec α =0.5





Figure IV.22.7: Image après filtrage avec α=0.5

$PSNR_1$ = 21.8613dB

Figure IV.23.7 :Marque extraite avec α=0.5

$PSNR_2$ =3.6653 dB

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	34.9513	30.1239	26.7517	20.5053	17.5188	15.4195	14.0186
Redimensionnement	12.2695	12.2544	12.2038	12.0051	11.7771	11.5177	11.2924
Cropping							

Filtrage passe-bas	**38.5897**	**37.1457**	**34.2322**	**29.5385**	**26.2994**	**23.7848**	**21.8613**
Rotation	**9.6374**	**9.6277**	**9.5955**	**9.7459**	**9.3251**	**9.1480**	**8.9950**

Tableau IV.2 : Tableau récapitulatif du $PSNR_1$ en dB vis à vis les différentes attaques.

α	**0,02**	**0,05**	**0,1**	**0,2**	**0,3**	**0,4**	**0,5**
Compression JPEG	**6.1983**	**12.5638**	**15.0381**	**21.8773**	**19.8234**	**16.9475**	**14.6924**
Redimensionnement	**1.9596**	**1.7200**	**1.5950**	**1.5101**	**1.4716**	**1.4439**	**1.4264**
Cropping							
Filtrage passe-bas	**0.8553**	**4.3954**	**4.3435**	**4.2922**	**4.1015**	**3.9176**	**3.6653**
Rotation	**2.1191**	**1.8536**	**1.6811**	**1.5520**	**1.4958**	**1.4610**	**1.4344**

Tableau IV.3 : Tableau récapitulatif du $PSNR_2$ en dB vis à vis les différentes attaques.

III.7-Commentaires

1) En se référant au tableau VI.1, nous remarquons que les valeurs du $PSNR_1$ et $PSNR_2$ sont inversement proportionnelles avec la force de tatouage α. Nous remarquons aussi que la marque extraite d'une façon très claire pour les différentes valeurs de α , elle est presque la même que la marque originale,ce qui est vérifié par les valeurs du $PSNR_2$ qui varie entre [77.1236-15.2609], cela d'une part ;d'autre part, nous remarquons que l'image tatouée n'a subi aucune dégradation avec un $PSNR_1$ variant entre [40.9611-13.9667] ;ce qui qualifie ce tatouage de robuste en moyenne.
2) Nous observons aussi que la marque sur l'image tatouée commence à apparaître tard (pour α =0.4) ; et elle est invisible pour les valeurs de α< 0.4, ce qui qualifie ce type de tatouage d'invisible.
3) En se référant au tableau VI.II et VI.III, nous remarquons que ce type de tatouage avec DWT est fragile devant les

attaques géométriques (rotation et resizing), et filtrage passe-bas suite aux dégradations remarquées sur la marque extraite et selon les valeurs du $PSNR_1$ et $PSNR_2$ (trop faibles).En ce qui concerne l'attaque par compression JPEG, ce tatouage par DWT est moyennement robuste en se basant sur l'état de la marque extraite, et les valeurs du $PSNR_1$ et $PSNR_2$(des valeurs modérées).

III.7-Conclusion

Dans ce chapitre, nous avons présenté une étude de performance de la méthode de tatouage des images fixes à base de la transformée en ondelettes de Haar. La technique a été testée sur un certain nombre d'images comme Lena. La robustesse du marquage a été validée à l'aide des attaques de compression JPEG, redimensionnement, rotation et filtrage passe-bas. Les résultats obtenus montrent que le tatouage par DWT est insensible à la compression JPEG, et fragile vis-à-vis les autres attaques. Dans le chapitre suivant, nous allons étudier une technique de tatouage combinant DCT et DWT.

Sommaire

Introduction ……..... 1

Chapitre I : Notions de tatouage numérique

I.1-Introduction…….. 3

I.2-Pourquoi le tatouage numérique ? ……. 3

I.3-Terminologie……..3

I.3.1-Chiffrement …… 3

I.3.2-Stéganographie……..... 3

I.3.3-Tatouage numérique……..... 3

I.4-Schéma de tatouage…… 3

I.4.1-Phase d'insertion…….. 4

I.4.2-Phase de détection…… 4

I.5-Propriétés de la marque…….. 5

I.5.1-Capacité…… 5

I.5.2-Invisibilité…….. 5

I.5.3-Robustesse…… 5

I.6-Les attaques en tatouage numérique…….. 6

I.7-Tatouage unidirectionnel et tatouage bidirectionnel 7

I.7.1-Tatouage unidirectionnel…… 7

I.7.2-Tatouage bidirectionnel…….... 7

I.7.2.1-Notion de pixel…… 7

a- Définition …….. 8

b- Dimension d'un pixel……. 9

c- Occupation mémoire d'un pixel……. 9

I.7.2.2-Image numérique……..... 9

a- Type d'images…… 9

a.1- Images matricielles (ou image bitmap) ……..
9

a.2-Images vectorielles…….. 9

b- Formats d'images…….. 10

b-1-Définition…….. 10

I.8-Marquage des images numériques (classification des principaux algorithmes)…… ….10

I.8.1-Classification selon la manière d'insertion…… 11

I.8.1.1-Schéma additif …… …. 11

I.8.1.2-Schéma substitutif…… … 11

I.8.2-Classification selon le domaine utilisé…… ……. 11

I.8.2.1-Le tatouage dans le domaine spatial…… ….
11

a-La méthode du patchwork…… …… 11

b-Watermarking sur les bits de poids faible…… 12

I.8.2.2-Le tatouage dans le domaine fréquentiel……..…
14

I.9-Autres applications du tatouage…….. 16

I.10-
Conclusion.....
16

Chapitre II : Classification des attaques en tatouage numérique des images fixes.

II.1-Introduction…….. 17

II.2- Classification des attaques en tatouage numérique des images fixes..17

II.2.1- Classification de Hartung…… 17

II.2.2- Classification de Voloshynovskiy…… 17

II.3-Attaques non intentionnelles en tatouage d'images fixes …….. 19

II.4- Attaques intentionnelles en tatouage d'images fixes…….. 20

II.4.1- Attaques intentionnelles en connaissance de l'encodeur et du détecteur……20

II.4.2- Attaques intentionnelles en connaissance de l'encodeur ou du détecteur…….…… 21

II.4.3- Attaques intentionnelles en connaissance ni de l'encodeur ni du détecteur…….. 21

II.4.3.1-Attaque par ambiguïté…… 21

II.4.3.2Attaque par désynchronisation…… 22

II.4.3.1-Attaque de protocole…….. 22

II.5-Les bancs de tests……... 24

II.5.1-Banc de test Stirmark…….. 24

II.5.2-Banc de test Checkmark…….. 24

II.5.3-Banc de test Optimark…….. 25

II.5.4-Banc de test Certimark…….. 25

II.6-Conclusion…….. 26

Chapitre III : Tatouage des images numériques fixes par la transformée en cosinus discrète DCT.

III.1-Introduction…….. 27

III.2-Pourquoi la DCT ?..... 27

III.3-Rappels mathématiques…… 28

III.4-Principe de tatouage d'images fixes à base de la transformée DCT..28

III.5-Algorithme de tatouage utilisant la transformée DCT…….. 31

III.5.1- Algorithme de tatouage proposé utilisant la DCT…….. 32

III.5.1.a-Phase d'insertion de la marque…… 32

III.5.1.b-Phase d'extraction de la marque…….. 33

III.6-Test de la méthode de tatouage par DCT…….. 34

III.6.1- Test de la méthode de tatouage par DCT sans attaques…….. 37

III.6.2-Comportement du tatouage par DCT vis-à-vis les attaques…. 41

III.7-Commentaires…… 55

III.8-Conclusion……. 56

Chapitre IV : Tatouage des images numériques fixes par la transformée en ondelettes.

IV.1-Introduction…….…… 57

IV.2-Les notions de la transformée en ondelettes…….. 57

IV.2.1-Définition…… 57

IV.2.2-Les notions de la décomposition par les filtres en quadratures..... 57

IV.3-Les ondelettes de Haar……..… 58

IV.3.1-Procedure de calcul……..… 58

IV.4- Exemple de calcul de la transformée DWT par la méthode de Haar…….. 61

IV.5-Algorithme de tatouage utilisant la DWT…… 64

IV.5.1- Phase d'insertion de la marque…… 64

IV.5.2- Organigramme d'insertion…….. 66

IV.5.3- Phase d'extraction de la marque…… 66

IV.5.4-Organnigramme d'extraction…….. 67

IV.6-Test de la méthode de tatouage par DWT……..... 68

IV.6.1- Test de la méthode de tatouage par DWT sans attaques……. 68

IV.6.2-Comportement du tatouage par DWT vis-à-vis les attaques….72

IV.7-Commentaires…… 87

IV.8-Conclusion……. 87

Chapitre V : Développement de techniques de tatouage dans le domaine des transformées.

V.1-Introduction…… 88

V.2-Comparaison entre le tatouage par (DCT) et le tatouage par (DWT)..88

V.3-Algorithme de tatouage utilisant la méthode combinée…… 90

V.3.1- Phase d'insertion de la marque…… 90

V.3.2-Organigramme d'insertion de la marque par la méthode combinée…… 91

V.3.3- Phase d'extraction de la marque par la méthode combinée…… 91

V.3.4- Organigramme d'extraction de la marque par la méthode combinée…… 92

V.4-Test de la méthode proposée…….. 92

V.4.1- Test de la méthode proposée sans attaques…….. 93

V.4.2- Comportement de la méthode proposée vis-à-vis les attaques... 97

V.5- Commentaires…… 112
V.6- Conclusion…. 112

Conclusion et perspectives

Bibliographie

Liste des tableaux

Liste des figures

Nomenclature

Introduction

Le watermarking ou tatouage est une technique de protection de l'information contre toute attaque [25]. Cette technique, qui a reçu une attention considérable par un grand nombre de chercheurs ces dernières années, est devenue une nécessité pour la protection des films, images ou base de données sonores contre toute tentative d'attaque, surtout avec l'accroissement des échanges d'informations sur le web[11]. Pour répondre à ce besoin, le tatouage trouve son origine dans le manque de techniques fiables. En effet le principe de tatouage consiste à insérer dans un document numérique une marque invisible contenant un code robuste face a toute attaque susceptible de modifier la donnée tatouée [30]. Plusieurs algorithmes de tatouage qui ont vu le jour ; cherchent à optimiser le compromis robustesse- invisibilité. Toutefois, aucun d'eux ne satisfait un cahier de charge idéal, et en outre chacune de ces méthodes s'inscrit dans de type de document bien précis [38]. On ne connaît, en effet, aucune méthode qui s'adapte à tout type de documents ; ceci contribue à l'expansion de ce domaine de recherche visant la découverte d'une meilleure solution [27]. Tout ça, nous a incité à prendre part dans ce domaine de recherche a travers ce modeste travail, avec l'espoir de contribuer par des idées et des résultats satisfaisants. Dés la première vue, nous pouvons distinguer deux types de tatouages : Le tatouage visible et le tatouage invisible [2]. Le tatouage visible est généralement destiné à insérer un logo (marque) ou une information indiquant la propriété intellectuelle dans un film (vidéo), une image, un ECG ou une prise sonore. Le tatouage invisible sert à insérer une empreinte indiquant la propriété dans l'information utile d'une façon imperceptible. Toute technique de tatouage doit être robuste, simple pour l'insertion et la vérification et imperceptible dans le cas de tatouage invisible [30].

L'objectif de notre travail, consiste à développer des techniques de tatouage d'images numériques fixes, basées sur les transformées célèbres dans le domaine fréquentiel à savoir : la transformée en cosinus discrète DCT et la transformée en ondelettes DWT. Une étude des performances de ces techniques est jugée utile en termes de robustesse, imperceptibilité, et simplicité d'insertion et de vérification. La robustesse aux attaques les plus célèbres (compression, filtrage, transformation géométrique, etc.) sera considérée [13].

A fin d'arriver à cette fin, nous avons rappelé au chapitre I les notions générales du tatouage numérique à savoir les différentes phases pour effectuer un tatouage, l'outil mathématique pour mesurer la robustesse d'un tel tatouage, ensuite nous avons cerné le sujet dans l'image numérique fixe,là où nous avons bien détaillé les constituants et les caractéristiques de cette dernière (notion de pixel, format d'image, etc.…), puis nous avons présenté la classification du tatouage numérique selon la manière d'insertion et selon le domaine utilisé (spatial ou fréquentiel). C'est dans cette dernière classification où nous avons décrit quelques algorithmes [4], [20], [43], [17] surtout dans le domaine fréquentiel comme celui de Cox et al [4], et qui constituent une assise fondamentale pour notre thème.

Dans le chapitre II, nous avons présenté toutes les attaques que peut subir un tatouage d'image. En effet, il existe deux catégories d'attaques : les attaques intentionnelles et les attaques non intentionnelles [14] à travers lesquelles nous pouvons mesurer les performances d'un tel tatouage.

Le chapitre III, est consacré pleinement au tatouage numérique des images fixes dans le domaine fréquentiel par utilisation de la transformée en cosinus discrète DCT. Après avoir donner l'outil mathématique pour effectuer cette transformation, nous avons présenté un simple exemple sur un bloc de 8x8, avant de se pencher vers l'application de cette méthode sur une image fixe.

Une fois maîtrisant le principe, nous avons développé notre propre algorithme de tatouage par DCT qui se base sur l'incrustation de la marque dans les moyennes fréquences afin de ne pas endommager l'image originale. Enfin, nous avons concrétisé cet effort par la partie test, là où nous avons tiré des résultats et des constats.

Au chapitre IV, une deuxième transformation est utilisée, il s'agit de la transformée en ondelettes discrète DWT. Nous avons suivi le même cheminement du chapitre précédent, et nous avons présenté aussi notre propre algorithme de tatouage qui s'articule sur l'incrustation de la marque dans la zone de détail pour ne pas avoir des dégradations sur l'image originale. Tout ça est achevé par la partie test ; là aussi nous avons tiré des résultats et des constats.

Dans le chapitre V, nous avons développé une nouvelle méthode combinée entre les deux transformées DCT et DWT dans le but d'améliorer les performances du tatouage et de trouver des solutions au compromis rencontré aux chapitres III et IV.

Pour atteindre les objectifs de ce travail, Matlab 7.1 a été choisi comme outil de programmation.

Enfin, nous terminons notre manuscrit par une conclusion et des perspectives.

I.1-Introduction

Ce chapitre rappelle les notions générales de tatouage numérique, les caractéristiques des images numériques et les différentes techniques de marquage proposées dans la littérature scientifique.

I.2-Pourquoi le tatouage numérique ?

Certainement notre ère numérique a ouvert une immense capacité de stockage des données ainsi qu'une grande facilité de circulation de ces derniers, mais ceci a engendré aussi une facilité de piratage. Les données peuvent être dupliquées de nombreuses fois et le développement de réseaux rend très difficile la poursuite des fraudeurs, donc la question qui se pose alors : « comment protéger les documents numériques contre le piratage et comment protéger leurs ayant droit ? » [34].

I.3- Terminologie

I.3.1-Chiffrement

C'est une technique cryptographique qui permet de protéger le document pendant sa transmission, mais une fois le document décrypté, il ne présente aucune protection **[11]**.

I.3.2-Stéganographie

« C'est l'art de cacher un message dans un autre ». [11] C'est une autre solution de protection et contrairement a la cryptographie,

cette technique est invisible et permet de protéger le document même lorsque celui-ci est diffusé. Sa faiblesse réside dans le manque de robustesse, en effet, il est facile d'effacer le message inséré par changement systématique du document. De ce fait, la nécessité de recourir à des procédés plus performants de protection du copyright devient un besoin primordial, d'où l'apparition récente de la notion de tatouage numérique [18].

I.3.3-Tatouage numérique [18]

Il ne s'agit pas ici de cacher une marque dans un document, mais de le marquer d'une façon indélébile. Le document peut être de diverses natures (texte, image, son, vidéo). Pour notre cas, on s'intéresse essentiellement au type image.

I.4- Schéma de tatouage [34]

Le schéma de tatouage se découpe en deux étapes fondamentales : L'insertion et la détection de la signature.

I.4.1-Phase d'insertion

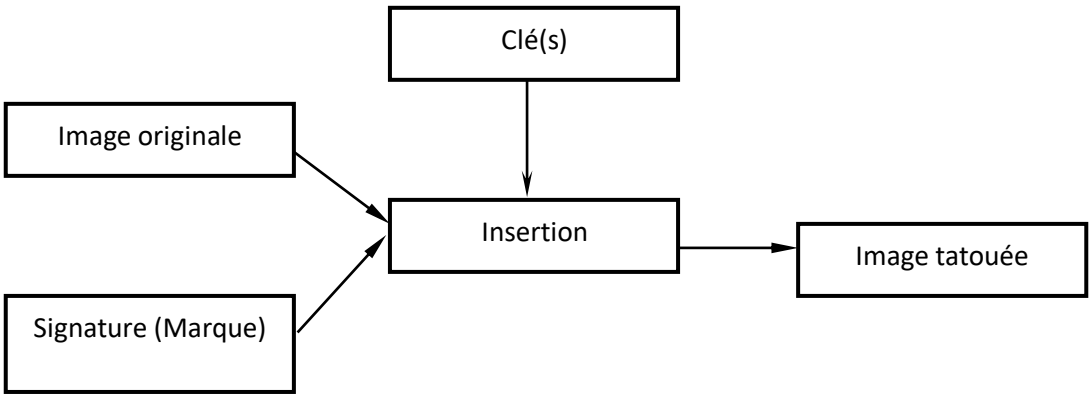


Figure I.1 : Schéma d'insertion.

Cette phase consiste a introduire une marque dans un document afin d'identifier son propriétaire et cette insertion nécessite la possession de l'objet original et de la marque a insérer, un troisième paramètre facultatif (une clé secrète) peut être utilisée en cas de besoin. Cette clé peut servir soit a la mise en forme de la signature, soit pour localiser sur l'objet d'origine l'emplacement de pixels ou il va y avoir l'insertion C.H.Lee et COL utilisent bien dans **[34]**, deux clés secrètes pour satisfaire ces deux points.

I.4.2-Phase de détection

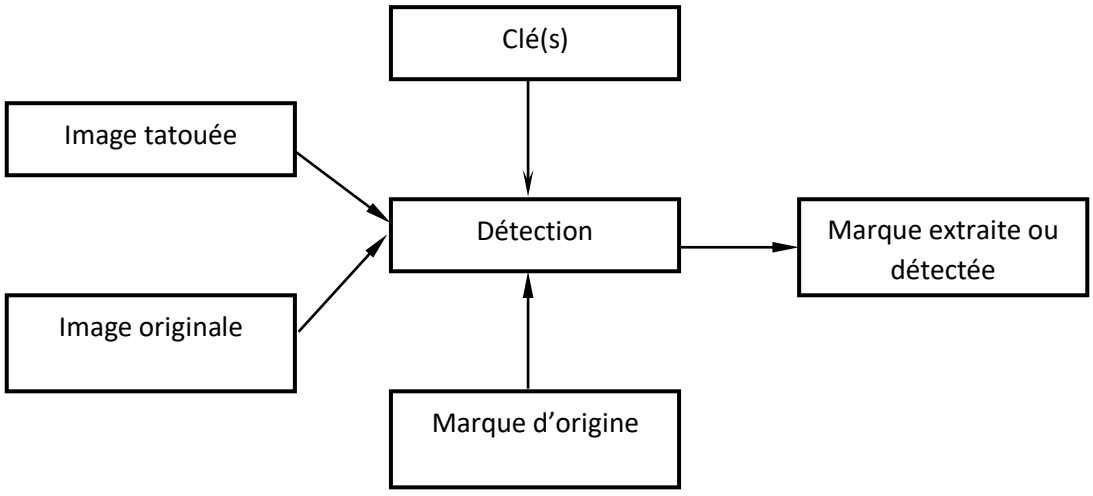


Figure I.2 : Schéma de détection.

Cette phase permet de prouver la présence de la marque, ceci peut être réalisé par extraction complète de la marque insérée **[34]** ou par détection d'un pic supérieur à un certain seuil prédéfini obtenu par une simple inter corrélation entre la marque d'origine et celle extraite **[34].**

Dans le premier cas, on a besoin d'une image tatouée, des clés et éventuellement de l'image d'origine, alors que dans le second on se sert en plus de la marque d'origine. D'une manière générale, lors de l'utilisation de l'image d'origine, on qualifie la détection d'"Aveugle" et de "Non Aveugle" dans le cas contraire.

I.5-Proprietes de la marque [11]

La marque insérée doit satisfaire les trois contraintes suivantes, à savoir :

- Capacité.
- Invisibilité.
- Robustesse aux différentes attaques.

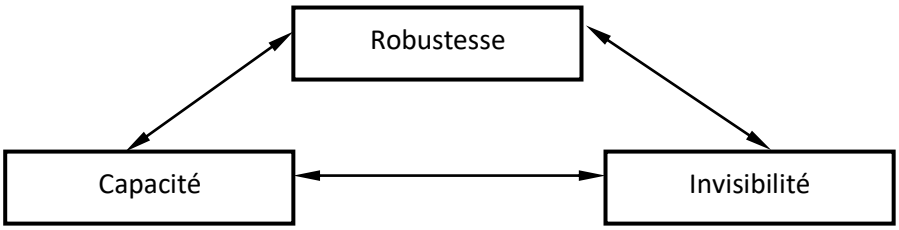


Figure I.3 : Propriétés de la marque.

I.5.1-Capacité

Elle représente la quantité d'information que l'on insère dans la marque.

La tendance actuelle vise à ce que la marque contienne au moins 60 bits d'informations codées en un message composé de 0 et 1. Ceci est important pour rendre la marque assez significative (nom, auteur, entreprise…).

I.5.2-Invisibilité

La marque doit être invisible, en effet une marque visible peut être aisément écrasée par une autre marque. De ce fait garantir l'invisibilité de la signature revient à respecter la qualité visuelle de l'image, aucune dégradation ne doit être perçue sur l'objet d'origine. C'est pour cette raison que plusieurs algorithmes de marquage existants ont tendance à effectuer l'insertion dans les zones d'intérêts les moins sensibles à l'œil humain (contour, zone texturée…).

I.5.3-Robustesse

Etre robuste c'est équivalent à pouvoir retrouver la marque inséré quelles que soient les modifications que peut subir l'objet tatoué. Ces modifications sont dues à plusieurs types de traitements (attaques) dont trois sont essentiels.

Le premier concerne les attaques simples telles que la compression JPEG, le filtrage, l'addition de bruit, l'impression suivie par une numérisation à l'aide d'un scanner. Cette classe ne change pas la géométrie de l'image, alors que la deuxième catégorie la modifie, on y trouve la rotation, le changement d'échelle et les transformations affines. La plus part des systèmes de marquage résistent à ces transformations simples mais ne supportent pas la combinaison de plusieurs entre elles.

La troisième catégorie d'attaque vise a introduire une ambiguïté, ceci peut être réalisé par une attaque de sur marquage dans laquelle on insère d'autres marques pouvant créer un conflit pour décider quelle est la vraie marque d'origine. F.Petitcolas et Kuhn détaillent bien dans **[28]** ces différentes attaques.

Pratiquement pour mesurer la robustesse d'un tel tatouage, on calcule traditionnellement en image fixe le PSNR (Peak Signal to Noise Ratio) [12].

Soit P_o l'image de référence et P_w l'image modifiée, les deux images sont de la même taille : M×N.

Le PSNR s'exprime en décibels :

$$PSNR\ (P_o, P_w) = 10\log_{10}\left(\frac{D^2}{EQM\ (p_o, p_w)}\right) \quad (I.1)$$

Où D est la dynamique du signal sur une image dont chaque pixel

est codé sur 8 bits, les valeurs de ces pixels vont de 0 à 255.

La dynamique D vaut donc 255.L'EQM est l'erreur quadratique moyenne :

$$EQM\ (P_o, P_w) = \frac{1}{M \times N} \sum_{}^{M-1\ N-1} \sum (P_o(y, x) - P_w(y, x))^2 \quad (I.2)$$

Ainsi le PSNR est infini si les deux images sont identiques, plus le PSNR

est élevé, plus les images sont proches, et inversement [6].

I.6- Les attaques en tatouage numérique [38]

Les attaques tiennent une place très importante dans le cahier des charges d'un processus de tatouage puisqu'elles définissent la robustesse d'un système. Une façon simple de classer les attaques serait de quantifier les dégradations qu'elles font subir à l'image. Ceci permet de vérifier directement que le cahier des charges est respecté. Nous avons vu ci dessus les problèmes posés par la caractérisation de la qualité d'une image. Nous n'avons pas pour le moment de critère assez performant pour classer les attaques de cette façon.

Attaques sur l'implémentation	Attaques sur la détection
cropping	applications affines
filtrage	ajout de bruit
compression	jitter attack
débruitage	passage à l'analogique
moyennage	Stirmark
impasse	Unsign
	mosaïque
	collusion
	surmarquage
	copiage
	fausses alarmes naturelles

Tableau I.1 : Classification des attaques.

I.7-Tatouage unidirectionnel et tatouage bidirectionnel [2]

I.7.1- Tatouage unidirectionnel

Il consiste à marquer les signaux représentants un ECG (électrocardiogramme) et les signaux audibles.

I.7.2- Tatouage bidirectionnel

C'est le tatouage de l'image fixe, et du moment que notre projet s'intéresse de l'image fixe, on a jugé utile de donner quelques notions sur l'image numérique.

I.7.2.1-Notion de Pixel [40]

a- Définition

Le pixel, abrégé Px, est l'unité de base d'une image numérique. Son nom provient de la locution anglaise picture element, qui signifie, « élément d'image » ou « point élémentaire ».C'est l'unité minimale adressable par le contrôleur vidéo. C'est par exemple l'unité utilisée pour spécifier les définitions d'affichage (largeur × hauteur):

- La définition du VGA (Video Graphics Array) est de 640 × 480, soit 307 200 pixels.
- La définition du XGA (eXtended Graphics Array) est de 1 024 × 768, soit 786 432 pixels.
- La définition du UXGA (ultra extended graphics array) est de 1 600 × 1 200, soit 1 920 000 pixels.

À chaque pixel est associée une couleur, usuellement décomposée en trois composantes primaires (Rouge Vert Bleu).

b- Dimension d'un pixel [40]

Les pixels sont rectangulaires, approximativement carrés, mais rarement parfaitement carrés. Leur dimension peut être changée en réglant l'écran ou la carte graphique. Habituellement, on indique la taille de l'écran en donnant la longueur de la diagonale, en pouces pour le matériel informatique. Un écran classique a un rapport largeur sur hauteur de 4/3.

Définition de l'écran	Taille de pixel (mm × mm)			
VGA (640 × 480 px)	0,45 × 0,45	0,54 × 0,54	0,60 × 0,60	0,66 × 0,66
XGA (1 024 × 768 px)	0,28 × 0,28	0,34 × 0,34	0,37 × 0,37	0,41 × 0,41
SXGA (1 280 × 1 024 px)	0,225 × 0,211	0,269 × 0,252	0,300 × 0,281	0,331 × 0,311
UXGA (1 600 × 1 200)	0,180 × 0,180	0,215 × 0,215	0,240 × 0,240	0,265 × 0,265

c- Occupation mémoire d'un pixel

Dans le domaine informatique, un pixel est codé sur un ou plusieurs bits, selon la « profondeur », c'est-à-dire du nombre de couleurs affichables :

- noir et blanc : un bit ;
- 16 couleurs (standard VGA) : 4 bits ;
- 256 couleurs : 8 bits (1 octet) ;
- 65 536 couleurs (« milliers de couleurs ») : 16 bits ;
- 16 777 216 couleurs (« 16 millions de couleurs », *true colors*) : 24 bits.

La place mémoire réelle utilisée peut être plus importante. Par exemple, en mode 16 millions de couleurs, le pixel occupe parfois 32 bits (4 octets), l'octet supplémentaire étant inutilisé ou bien utilisé pour des calculs 3D.

I.7.2.2- Image numérique [40]

On désigne sous le terme d'image numérique toute image (dessin, icône, photographie …) acquise, créée, traitée ou stockée sous forme binaire (suite de 0 et de 1):

- Acquise par des dispositifs comme les scanners, les appareils photo ou caméscopes numériques, les cartes d'acquisition vidéo (qui numérisent directement une source comme la télévision).
- Créée directement par des programmes informatiques, via la souris, les tablettes graphiques ou par la modélisation 3D (ce que l'on appelle par abus de langage les « images de synthèse »).

- Traitée grâce à des outils informatiques. Il est facile de la modifier en taille, en couleur, d'ajouter ou supprimer des éléments, d'appliquer des filtres variés, etc.
- Stockée sur un support informatique (disquette, disque dur, CD-ROM, …)

a- Types d'images [40]

On distingue deux types d'images à la composition et au comportement différent : images matricielles et images vectorielles.

a.1- Images matricielles (ou images bitmap)

Elle est composée comme son nom l'indique d'une matrice de points à plusieurs dimensions, chaque dimension représentant une dimension spatiale (hauteur, largeur, profondeur), temporelle (durée) ou autre (par exemple, un niveau de résolution). Dans le cas des images à deux dimensions (le plus courant), les points sont les pixels. D'un point de vue mathématique, on considère l'image comme une fonction de $\mathbb{R} \times \mathbb{R}$ dans $\mathbb{R}$ où le couplet d'entrée est considéré comme une position spatiale, le singleton de sortie comme un codage.

a.2- Images vectorielles

Le principe est de représenter les données de l'image par des formules géométriques qui vont pouvoir être décrites d'un point de vue mathématique. Cela signifie qu'au lieu de mémoriser une mosaïque de points élémentaires, on stocke la succession d'opérations conduisant au tracé. L'avantage de ce type d'image est la possibilité de l'agrandir indéfiniment sans perdre la qualité initiale, ainsi qu'un faible encombrement. L'usage de prédilection de ce type d'images concerne les schémas qu'il est possible de générer avec certains logiciels de DAO (Dessin Assisté par Ordinateur) comme AUTO CAD ou CATIA. Ce type d'images est aussi utilisé pour les animations Flash, utilisées sur Internet pour la création de bannières publicitaires, l'introduction de sites web, voire des sites web complets.

b- Formats d'images [40]

b.1- Définition

Un format d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler.

	Type (matriciel/ vectoriel)	Compression des données	Nombre de couleurs supportées	Affichage progressif	Animation	Transparence
JPEG	matriciel	Oui, réglable (avec perte)	16 millions	Oui	Non	Non
JPEG2000	matriciel	Oui, avec ou sans perte	32 millions	Oui	Oui	Oui
GIF	matriciel	Oui, Sans perte	256 maxi (palette)	Oui	Oui	Oui
PNG	matriciel	Oui, sans perte	Palettisé (256 couleurs ou moins) ou 16 millions	Oui	Non	Oui (couche Alpha)
TIFF	matriciel	Compression ou pas avec ou sans pertes	de monochrome à 16 millions	Non	Non	Oui (couche Alpha)
SVG	vectoriel	compression possible	16 millions	* ne s'applique pas *	Oui	Oui (par nature)

Tableau I.2 : Formats d'images.

I.8-Marquage des images numériques(classification des principaux algorithmes)

La multiplicité des algorithmes de marquage rend leur présentation exhaustive assez difficile. Donc, notre intérêt sera dirigé vers la classification de ces algorithmes [38].

Essentiellement, on peut distinguer deux critères de classification :

- le mode d'insertion de la marque (schéma additif, schéma substitutif) ;
- le domaine utilisé (spatial, fréquentiel, multi résolution).

Pour mieux étudier les algorithmes de tatouage, nous allons traiter séparément ces deux critères.

I.8.1-Classification selon la manière d'insertion [34]

I.8.1.1-Schéma additif

Ce schéma se résume dans l'extraction des cœfficients à modifier de l'objet d'origine puis le tatouage de cet objet par ajout de la marque à ces cœfficients.

L'insertion peut s'effectuer soit directement sur l'image, dans le domaine spatial, soit dans un domaine transformé. De ce fait, adapter la marque à l'objet d'origine est une contrainte essentielle à respecter pour que le signal qu'elle représente ne soit ni trop faible (problème de robustesse) ni trop fort (dégradation du signal original).

I.8.1.2-Schéma substitutif

Dans les modes substitutifs, l'information à insérer est substituée à des caractéristiques de l'image, l'idée de base consiste donc à insérer une signature en modifiant le contenu structurel de l'image. Ainsi, l'étape d'insertion consiste d'une part à détecter les points d'intérêts et d'autre part à insérer des similarités autour de ces points. A la suite, la détection de marque s'effectue par recherche de ces similarités.

I.8.2-Classification selon le domaine utilisé [34]

I.8.2.1-Le tatouage dans le domaine spatial

Les méthodes agissant dans le domaine spatial modifient directement les valeurs des pixels de l'image. Comme aucun traitement initial n'est requis, ces algorithmes sont très rapides et permettent de travailler en temps réel. Ils offrent souvent une bonne résistance aux opérations géométriques. Le schéma de « Patchwork » se situe dans ce domaine.

a-La méthode du Patchwork [38]

Cette méthode, très classique, est introduite par Bender et al. Elle est classée dans les méthodes à schéma semi-produit. Voici son principe :

A l'implémentation, l'algorithme sélectionne pseudo aléatoirement (avec une clé K) N paires de pixels de l'image. Les valeurs des luminances de ces paires de pixels $(a_i , b_i) =[1..N]$ sont alors modifiées selon les formules suivantes :

$$a^*_i = a_i + 1$$

$$b^*_i = b_i + 1$$

(I.3)

On augmente de 1 la valeur d'un certain groupe A de pixels et on diminue d'autant un autre groupe B de même cardinal ; la connaissance de ces deux ensembles étant conditionnée par celle de K. A la détection, la somme S, différence entre les valeurs de ces deux groupes de pixels, est calculée :

$$S^* = \sum_{i=1}^{N} (a_i^* - b_i^*)$$

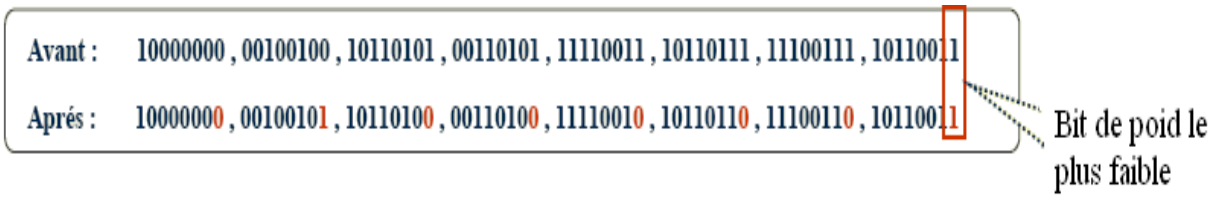
(I.4)

Si l'on connaît la clé définissant les deux ensembles A et B et en supposant que l'image satisfait certaines propriétés statistiques, la valeur attendue de la somme est 2N. Si l'on ne connaît pas cette clé, on ne peut pas retrouver les deux ensembles, on ne peut que générer deux ensembles différents. Si ces ensembles sont générés pseudo aléatoirement (avec une autre clé K0), l'espérance de la somme est alors nulle.

b-watermarking sur les bits de poids faibles [34]

Il s'agit là très certainement de la méthode la plus simple de data hi ding sur une image. Imaginons que l'on travaille sur une image en niveaux de gris (pour les images en couleurs, il suffit de travailler sur la luminance), et que l'on ait seulement 256 niveaux de gris possibles (de 0..255). On considère qu'entre 2 niveaux de gris consécutifs, l'oeil humain ne fait pas la différence (en fait cela dépend des niveaux de gris ou l'on se situe, mais nous verrons cela un peu plus loin).

Ex : Insérer un ‘A’ (en Binaire01000001, en Décimal 65)



Si l'on considère le niveau de gris en binaire, nous sommes en présence de 8 bits (<=> 256). Le changement du dernier de ces bits (le bit de poids faible) ne change le niveau de gris que de 1 unité (soit disant imperceptible à l'oeil). On peut donc raisonnablement considérer que ce dernier bit n'est pas important, donc que l'on peut y mettre ce que l'on veut. Et c'est ce que l'on fait, on prend une image en 2 couleurs (1 bit) et on la place sur le bit de poids faible. Ca y est votre image est dissimulé.

Bon, il faut avouer que cette méthode présente de nombreux problèmes (en tout cas présentée de cette façon). En effet, il est très simple d'enlever ce marquage en mettant par exemple à 0 tous les bits de poids faible. De plus, toutes les "nettoyages" de l'image ou les filtres sont radicaux pour ce marquage. Par exemple la compression JPEG ne lui laisse quasiment aucune chance. De plus le système visuel humain peut être sensible à ces petits changements.

En effet, l'oeil humain réagit de façon différente selon les contrastes. Ainsi, il est très sensible aux contrastes dans les gris de faibles intensités et beaucoup moins dans les teintes proches du blanc. Ainsi certaines méthodes profitent de cela en adaptant le nombre de bits de poids faibles à coder en fonction de la teinte en cours et de la teinte adjacente (tout en se référant à des données physiologiques sur les couleurs).

Malgré ses faiblesses de nombreuses méthodes dérivent de ce principe, bien que ce type de marquage soit de moins en moins à la mode.

De plus ce marquage dépend réellement du format de l'image que l'on veut marquer. En effet, cette méthode ne peut s'appliquer à des formats tels que le GIF (de Compuserve). Ici les couleurs des pixels ne sont pas indépendantes mais codées selon une palette de 256 couleurs. On donne une valeur comprise entre 0 et 255 à chaque pixel et lorsque l'on lit les pixels, on se réfère à la palette pour y mettre la couleur. Cela

signifie que si l'on s'amuse à changer le bit de poids faible des pixels on peut changer complètement de couleurs.

Exemple :

11111111 <=> 255 : et la couleur 255 est définie comme étant un rouge vif.

11111110 <=> 254 : et la couleur 254 est définie comme étant un bleu foncé.

Ainsi, en ne changeant que le dernier bit vous allez complètement détruire votre image....
pour remédier ce problème**,** Il existe plusieurs solutions pour marquer les palettes. La plus basique consiste à n'utiliser que 128 couleurs et à créer une palette ou les couleurs marchent par pairs:

Couleur 00 : Rouge=150, Vert=12, Bleu=59

Couleur 01 : Rouge=152, Vert=10, Bleu=50

Couleur 10 : Rouge=50, Vert=200, Bleu=98

Couleur 11 : Rouge=50, Vert=212, Bleu=95

Ainsi, en ne changeant que le bit de poids faible des pixels, on reste quasiment sur la même couleur. En théorie, on pourrait même mettre deux fois exactement la même couleur à la suite, mais cela serait vraiment trop voyant. On pourrait ainsi mettre 2 bits par pixel en se limitant à une palette de 64 couleurs. Cette technique additive a été proposée par Bender et Col [28].Elle représente une décomposition de l'image en deux ensembles de même taille. Ensuite, chaque pixel est modifié différemment selon son appartenance à un ensemble donné. La détection de la signature s'effectue en calculant la différence entre les moyennes des pixels des deux ensembles.Si cette valeur calculée est supérieure a un certain seuil, alors on peut affirmer la présence de la signature. Cependant, un tel schéma n'offre qu'une protection minimale contre la compression. Un faible taux de compression avec JPEG est capable de détruire la marque. Ce manque de robustesse a poussé

la recherche sur d'autres types de schémas résistants à la compression, d'où l'idée des schémas agissant dans le domaine fréquentiel.

I.8.2.2-Le tatouage dans le domaine fréquentiel [30]

Le tatouage dans le domaine fréquentiel est obtenu après utilisation de l'une des transformées suivante à savoir :

-la DFT (transformée de Fourier discrète)

-la DCT (transformée en cosinus discrète)

-la DWT (transformée en ondelettes discrète).Notre travail sera basé sur la DCT et la DWT.

Nous allons décrire deux algorithmes essentiels, qui sont a la base de nombreux algorithmes de ce type.

***Zhao** et **al. [20]** proposent d'insérer la signature dans les coefficients DCT de l'image. Pour chaque bit a insérer, un bloc 8x8 de la représentation est choisi. Une unique clé secrète détermine les blocs 8x8 choisis. Au sein de chaque bloc, trois coefficients DCT sont choisis parmi les huit cœfficients correspondant à des fréquences relativement basses de l'image. Sur la figureI.4, ces huit coefficients sont coloriés en gris.

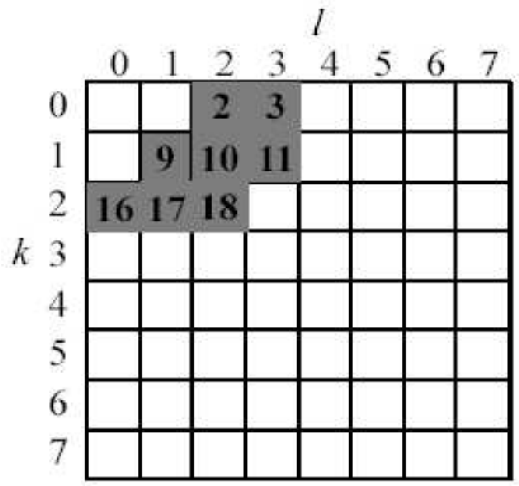


Figure I.4 : Les huit cœfficients qui peuvent être choisi pour enfouir un bit dans le bloc 8x8.

On identifie tout d'abord la relation d'ordre existant entre les valeurs absolues des coefficients, les coefficients le plus élevé sera appelé H (pour high), celui qui est entre les deux sera appelé M (pour medium) et le plus faible sera appelé L (pour low). Ces valeurs sont ensuite modifiées suivant le bit à insérer (0 ou 1).

(k_1,l_1)	(k_2,l_2)	(k_3,l_3)
H	M	L
M	H	L
H	H	L
M	L	H
L	M	H
L	L	H
H	L	M
L	H	M
M	M	M

Figure I.5 : Différentes possibilité de relation d'ordre entre les trois cœfficients.

***Cox** et **al. [4]** proposent une méthode qui utilise aussi la transformée en cosinus discrète pour insérer la marque dans l'image.

Ils appliquent la DCT à toute l'image et insèrent la signature dans les basses fréquences, c'est-à-dire dans les composantes les plus significatives de l'image. Ils modifient les fréquences de plus grande amplitude, exceptée la composante continue, suivant l'une des formules suivantes:

$$f_w = f + \alpha.w. \quad \textbf{(I.5)}$$

$$f_w = f + \alpha.w.f \quad \textbf{(I.6)}$$

$$f_w = f\ .e^{\alpha.w} \quad \textbf{(I.7)}$$

Avec :

f_w : coefficients DCT de l'image marquée

f : coefficients DCT de l'image originale

α : coefficient d'invisibilité,ou la force de tatouage modulable selon l'utilisateur.

W : coefficients DCT de la marque

L'extraction s'effectue en inversant le processus d'insertion et en utilisant l'image originale pour retrouver la marque. La suite w_i extraite est comparée à la suite w par un calcul de corrélation.

I.9-Autres applications du tatouage[21]

Si le tatouage numérique a pour but la protection des copyrights cités auparavant, il trouve aussi son application dans d'autres domaines tels que l'authentification des documents et l'indexation.

En ce qui concerne l'authentification, il s'agit ici de contrôler l'intégrité des documents ; donc elle fait recours à un tatouage fragile où la marque est conçue de manière à se détériorer dés que le document subi une attaque de modification. Un exemple de ce type d'utilisation du tatouage est l'authentification des papiers d'identité.

Maintenant, en ce qui concerne l'indexation, elle consiste à faciliter la recherche dans une base de données.

En conclusion, nous disons que ces applications demandent le moins de robustesse, elles ressemblent donc plutôt à la steganographie.

I.10- Conclusion

Les notions de base du tatouage numérique et d'image numérique ont été rappelées dans ce chapitre. Nous avons également présenté les différentes techniques de tatouage.

Les attaques qui peuvent subir une image tatouée feront l'objet du chapitre suivant.

II.1- Introduction

Les attaques tiennent une place très importante dans le cahier des charges d'un processus de tatouage puisqu'elles définissent la robustesse d'un système. Une façon simple de classer les attaques serait de quantifier les dégradations qu'elles font subir à l'image. Ceci permet de vérifier directement que le cahier des charges est respecté. Nous avons vu ci dessus les problèmes posés par la

caractérisation de la qualité d'une image. Nous n'avons pas pour le moment de critère assez performant pour classer les attaques, néanmoins il existe des propositions qui ont vu le jour depuis 1995.

Les premières attaques sur les schémas de tatouage ont été proposées simultanément par plusieurs chercheurs en 1995 [37].Stone et al. [31], Craver et al. [7], Cox et al. [4] [5].

Chacun d'eux proposa sa propre attaque :des attaques de collusion contre les schémas d'étalement de spectre,des attaques d'inversion,des attaques géométriques ou encore des attaques cryptographiques.

Dans ce qui suit, nous allons présenter tout d'abord deux classifications différentes des attaques proposées par Hartung et al.[13] et Voloshynovskiy et al [39].

II.2- Classification des attaques en tatouage numérique des images fixes [30]

Les deux principales classifications dans la littérature scientifique ont été présentées par Hartung et Voloshynovskiy.

Hartung propose de classer les attaques en quatre groupes :

II.2.1- Classification de Hartung

1. Les simples attaques qui ne sont pas intentionnelles comme les filtrages, les compressions, le bruit du à l'utilisation des données, la quantification ou encore les conversions analogiques –numériques.
2. Les attaques de désynchronisation qui empêchent la détection telles que les transformations géométriques, le fenêtrage, les permutations des pixels, l'insertion ou la suppression de pixels.
3. Les attaques d'ambiguïté qui visent à rendre confuse la détection telles que les attaques IBM [7] ou l'attaque de copie [22].
4. Les attaques de suppression dont le but est dans un premier temps d'estimer la marque pour ensuite la soustraire à la donnée numérique.

II.2.2- Classification de Voloshynovskiy [39]

Voloshynovskiy a amélioré cette classification en ajoutant de nouvelles techniques d'attaque, il a proposé le schéma suivant :

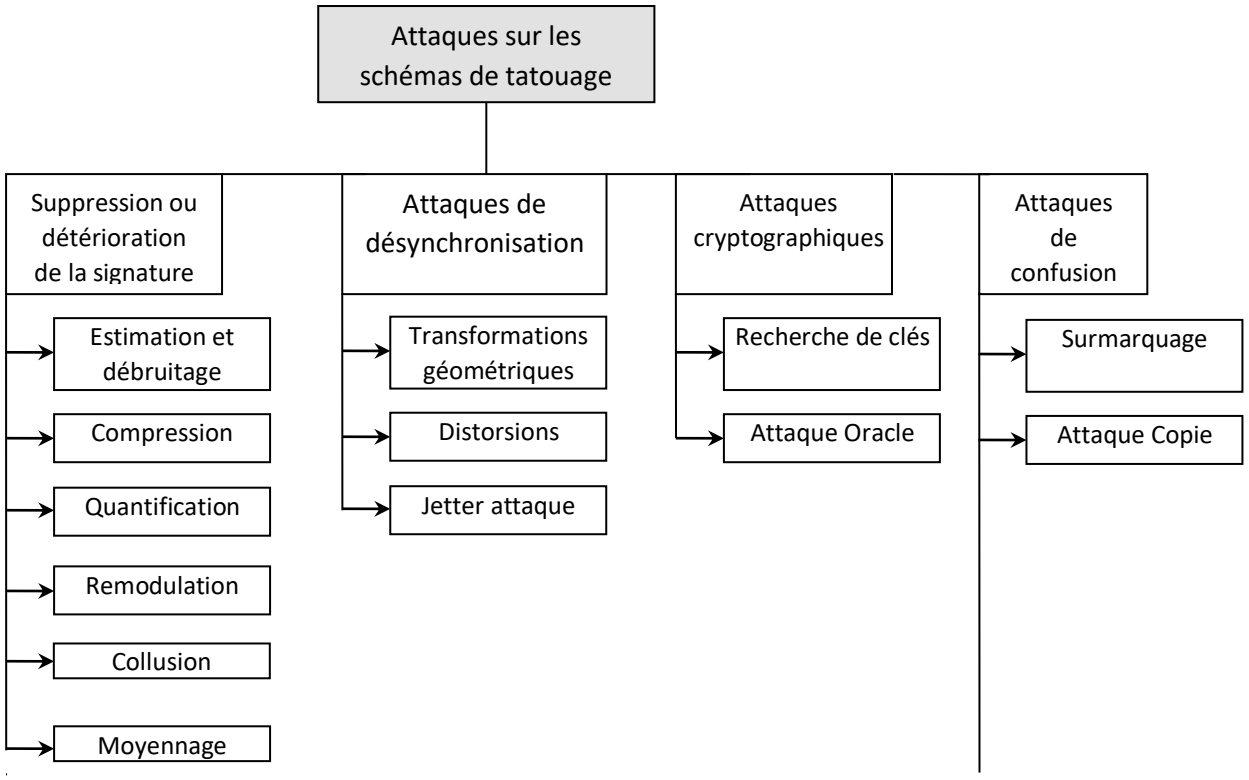


Figure II.1: Classification des attaques sur les schémas de ta touage par Voloshynovskiy.

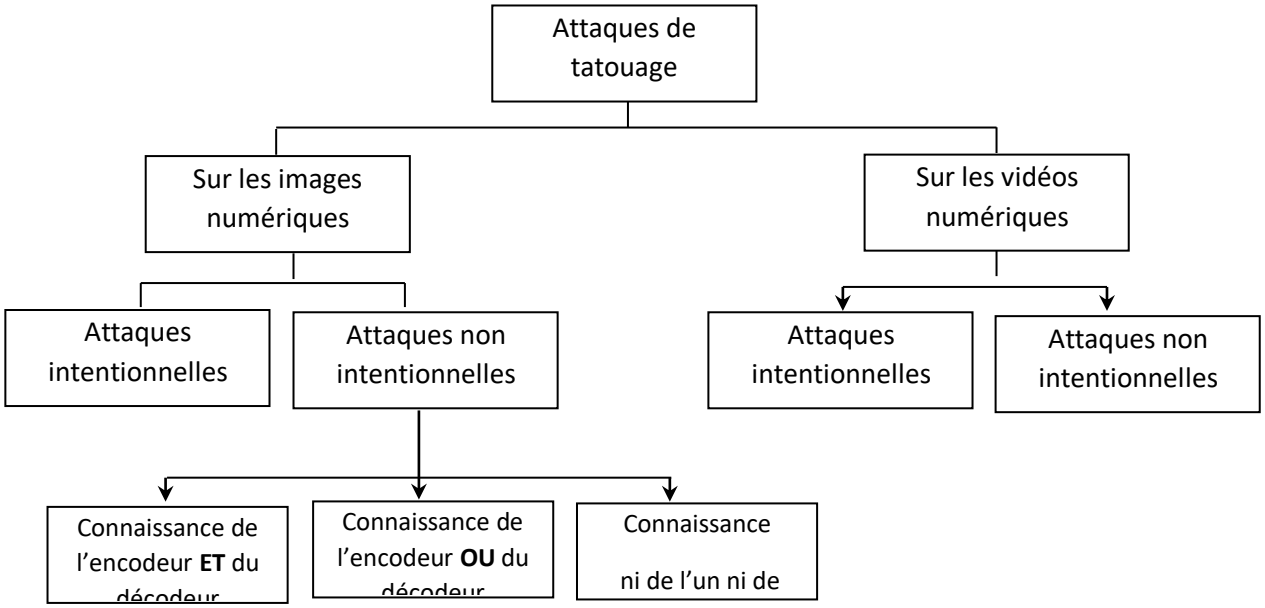


Figure II.2 : Classification des attaques selon la conférence de Seattle.

Reste a rappelé aussi la classification utilisée à Seattle en juillet 2002 lors de la conférence SPIE [37] pour la présentation de l'article :

« A survey on attacks in image and vidéo watermarking », présenté dans le schéma de la figure II.2

Ce schéma divise les attaques en deux catégories : les attaques qui sont intentionnelles et celles qui ne le sont pas.

II.3- Attaques non intentionnelles en tatouage d'images [30]

Dans ce cas, les attaques ne visent pas à supprimer la signature, l'identifier ou l'isoler. Ce sont des attaques consécutives à l'utilisation de la donnée tatouée par l'utilisateur. Elles peuvent provenir des canaux de diffusion qui ajoutent du bruit, ce bruit même s'il peut être estimé, dépend du type de données et des spécifications imposées par le distributeur (ou demandées par l'utilisateur). La taille de l'image tatouée peut être l par exemple varier selon le format de visualisation, la signature doit donc elle-même s'adapter à un agrandissement ou à une réduction de la taille de l'image qui la porte.

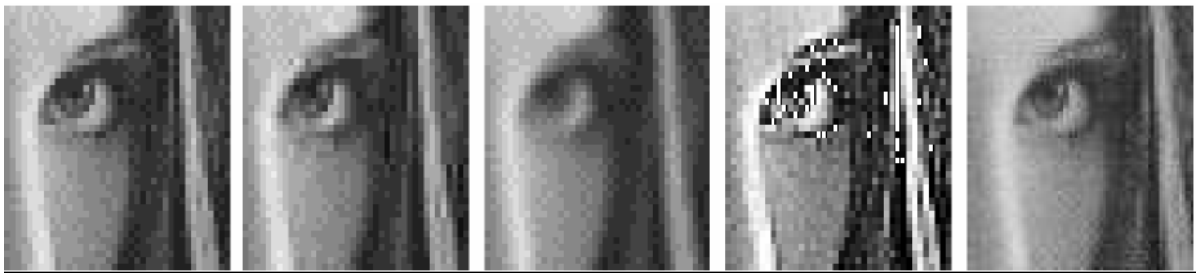
Dans le même temps ,les images doivent souvent être compressées pour s'adapter aux canaux de diffusion ou de visualisation,sur Internet par exemple ,la très grande majorité des images sont proposées dans un format compressé ,JPEG ou GIF , or ces compressions sont susceptibles de dégrader la marque insérée.

Dans le cas d'applications qui nécessitent une authentification du document et son intégrité, les impressions et les scannages peuvent dégrader suffisamment les données pour que la signature devienne indétectable : soit avec les conversions analogiques – numériques que nécessitent l'impression, soit à cause d'une éventuelle désynchronisation spatiale due au scannage.

Enfin les manipulations des données par l'utilisateur peuvent modifier le document et les propriétés de la marque insérée.

Ces attaques non intentionnelles correspondent à la manière dont le produit tatoué est utilisé, par exemple dans le cas de photos numériques, l'utilisateur peut chercher à améliorer la qualité de l'image. La marque devra alors résister à des opérations comme le fenêtrage, le zoom,des modifications de contraste ,de couleur ,de luminosité,les filtrages linéaires (passe bas, passe haut , passe bande) ou non linéaires (médian,morphologique).

Dans la figure II.3 sont représentées les différentes attaques non intentionnelles.



(1) (2) (3) (4) (5)

Figure II.3 : Attaques non intentionnelles.

(1) : Image originale zoomée puis fenêtrée.
(2) : Après une compression JPEG : des blocs de pixels apparaissent de détails disparaissent.
(3) : Après Moyennage : des détails disparaissent.
(4) : Après rehaussement : des hautes fréquences apparaissent.
(5) : Après impression et scannage : des blocs de pixels périodiques apparaissent.

II.4- Attaques intentionnelles en tatouage d'images fixes [30]

Ces attaques visent à supprimer ou à dégrader la marque insérée, l'attaque est dite efficace lorsque la signature est indétectable ou lorsque la décision au sujet de l'authenticité de la signature est ambiguë.

Trois situations vont être traitées séparément :

1. Quand l'attaquant connaît la totalité de l'algorithme de tatouage, à la fois l'encodeur et le décodeur.
2. Quand l'attaquant ne connaît que l'encodeur ou le décodeur.
3. Quand l'attaquant ne connaît rien de l'algorithme de tatouage.

II.4.1- Attaque intentionnelle en connaissance de l'encodeur et du détecteur

Dans ce cas, l'attaquant va chercher à introduire de l'ambiguïté lors de la phase de détection, l'exemple classique est celui présenté par Craver et al. [7].

L'attaque consiste à insérer une autre signature dans l'image déjà tatouée.

La nouvelle signature (w ') est insérée avec la même méthode d'insertion que pour la signature authentique (w).

Si on appelle l'image tatouée $P_w=P_O+w$, l'attaquant va construire la nouvelle image $P_{w'}= P_w+w'$ et dire que l'image originale est en fait P_O+w-w'

Enfin on peut citer l'attaque de Cox et al.[5] , l'attaque nommée oracle

Permet de former une image non tatouée à partir de la connaissance de l'encodeur et du décodeur.

II.4.2- Attaque intentionnelle en connaissance de l'encodeur ou du détecteur

Avec la connaissance de l'encodeur, Stone [31] propose une méthode de collusion qui consiste en un moyennage de différentes versions de l'image tatouée. L'objectif est de former une image sans signature ,des méthodes de collusion avancées ,telles que les méthodes dans le domaine DCT,permettent d'attaquer des schémas d'étalement de spectre avec des taux de réussite 50%,d'autres méthodes utilisent cette technique de moyennage, notamment dans le cas du tatouage vidéo. Voloshinovskiy et al. [39] n'ont pas besoin de connaître la méthode d'encodage mais uniquement des informations statistiques sur le type de signal inséré. L'attaque est décomposée en deux parties : la première vise à estimer la marque insérée et à la soustraire à l'image marquée grâce à un filtrage basé sur une estimation de Maximum à posteriori.

Ensuite, un bruit est ajouté à l'image, il est calculé en fonction des caractéristiques de la marque estimée et en exploitant les propriétés du Système Visuel Humain.

Les auteurs ajoutent du bruit pour brouiller le détecteur : l'idée consiste à changer aléatoirement le signe de la marque dans les zones les moins pénalisantes d'un point de vue visuel (ces zones sont déterminées à partir d'un calcul de masque psycho visuel).

II.4.3 Attaque intentionnelle en connaissance ni de l'encodeur ni du détecteur

Cette situation est la plus fréquente dans la littérature récente,en fait, les données tatouées sont souvent distribuées avec aucune connaissance de la façon dont la marque a été crée et comment elle sera détectée.

Beaucoup de ces attaques cherchent à estimer la marque pour ensuite la supprimer de l'image.

Les travaux de Langelaar et al. [23] concernant seulement les attaques sur les schémas d'étalement de spectre utilisant une marque de bruit blanc. La marque est estimée et soustraite avec un certain facteur d'amplitude. Su et al. [33] estiment la signature à partir d'un filtre de Wiener [15]puis la soustraient après l'avoir multipliée au préalable par un facteur force,ce facteur est déterminé lorsque la corrélation entre la marque estimée et l'image attaquée est minimale ,ensuite les auteurs ajoutent un bruit gaussien stationnaire à l'image modifiée.

II.4.3.1- Attaque par ambiguïté

Pour rendre la décision ambiguë Moulin et al.[24]utilisent les principes de la théorie de l'information pour évaluer l'influence de l'attaque ,ils concluent qu'ajouter un bruit blanc gaussien peut être une attaque optimale quand l'énergie du bruit blanc ajoutée est suffisamment supérieur à l'énergie de la marque ,Su et al [33] étudient la robustesse du schéma de tatouage après différentes attaques.

Une attaque est modélisée par un canal hostile. La dégradation est mesurée par une estimation des moindres carrés. L'attaque proposée consiste à ajouter un bruit gaussien et effectuer des petits filtrages linéaires. En sélectionnant le filtre approprié et le bruit adéquat, les auteurs affirment obtenir une attaque optimale en terme d'imperceptibilité.

II.4.3.2- Attaque par désynchronisation

Les attaques les plus pénalisantes dans cette situation sont celles qui visent à désynchroniser la marque avant la phase de détection. Avec l'attaque « Jitter », Petit colas et al. [28] proposent d'inverser, de supprimer ou de remplacer certaines lignes ou colonnes de l'image numérique. Cette attaque est très efficace face à des schémas d'insertion de type étalement de spectre. Dans le même article, les auteurs présentent l'attaque « Mosaïc » qui consiste à diviser

l'image en différentes parties, grâce à un navigateur Internet par exemple. A cause de la division, la détection ne pourra pas être effectuée sur toute l'image mais seulement sur des parties séparées de l'image. Pour beaucoup d'algorithmes, c'est impossible. Herrigel et al. [14] proposent l'attaque de mires. Elle consiste à supprimer les mires fréquentielles, générées lors du tatouage, pour désynchroniser la marque en détection. Deux phases peuvent être distinguées : la première consiste à estimer la marque et à ensuite la soustraire à l'image. La seconde consiste à supprimer les plus grandes fréquences dans le domaine de Fourier grâce à des filtres moyenneurs locaux. Comme dernier exemple on peut citer l'attaque « random bending » proposée par Voloshinovskiy et al.[39]. Elle consiste à applique des déformations géométriques aléatoires sur l'image tatouée comme le logiciel Stirmark. Des petits seuillages sont effectués sur les zones planes de l'image tatouée pour dégrader la marque dans ces zones.

II.4.3.3- Attaque de protocole

Le dernier groupe d'attaques dans cette situation est celui des attaques appelées attaques de protocole. Leur but est d'introduire une ambiguïté en détection sans pour autant dégrader la signature insérée. La plus connue est proposée par Kutter et al. [22] et se nomme attaque de copie. L'idée consiste à copie la signature contenue dans une image tatouée dans autre image proposée par l'attaquant. Le processus est divisé en trois étapes : la première consiste à estimer la marque, la seconde à adapter cette marque à la « fausse » image authentique et la dernière à ajouter l'estimation remodelée à la « fausse » image. Le résultat d'une attaque de copie est représenté sur la figure 2.16.



(1) (2) (3)

Figure II.4 : Illustration de l'attaque de copie.

(1) : Image tatouée.

(2) : « Fausse » image.

(3) : « Fausse » image tatouée.

La marque estimée a été rehaussée sur l'image (3) pour que celle-ci devienne visible. Dans un cas réel la marque estimée et insérée dans la « fausse » image doit être invisible. En cas de litige, et en possédant l'image 3, l'attaquant peut arguer que lui-même est aussi capable de détecter la signature dans une autre image. Donc le propriétaire ne pourra pas revendiquer la propriété de la donnée tatouée puisque la signature qui est censée l'authentifier peut être détectée dans une autre donnée.

Holliman et al. [16] utilisent le même principe mais n'insèrent que certaines spécificités de la marque détectée au sein d'une autre image. Elles sont cependant suffisantes pour assurer la détection de la signature dans n'importe quelle image proposée par l'attaquant.

Dans les paragraphes 2 et 3 nous avons vu un éventail très complet des attaques sur les schémas de tatouage d'images numériques. On pourra remarquer que bien souvent dés qu'un schéma de tatouage

novateur et performation est proposé à la communauté, il suffit de quelques mois supplémentaires pour voir apparaître une technique d'attaque qui remettra en cause sa robustesse. Le meilleur exemple de cette situation est le challenge SDMI (Secure Digital Music Initiative) : cette initiative a été lancée en 2000 et vise à faire tester la robustesse des meilleurs algorithmes de protection audio (dont le tatouage) par des chercheurs et industriels. Aucun algorithme ne s'est révélé infaillible .

II.5- Les bancs de tests

Nous allons décrire dans cette partie les quatre bancs de test existants en tatouage numérique : Stirmark, Chechmark, Optimark et Certimark.

II.5.1- Banc de tests Stirmark

Le banc de test Stirmark [22], [28] est le premier apparu et reste toujours le plus célèbre, il consiste à combiner neuf attaques de types : rotations, fenêtrages, rehaussement du signal, changements d'échelle, découpages, transformations linéaires, d'autres transformations géométriques et distorsions géométriques.

Pour chaque attaque, le banc de test évalue l'algorithme : un score de 1 est donné si la signature est toujours détectable, un score de 0 dans le cas contraire, une moyenne des neuf résultats donne le résultat final de robustesse de l'algorithme.

Pour assurer une juste comparaison entre les algorithmes, Stirmark stipule que le PSNR de l'image tatouée ne doit pas être supérieur à 38 dB.

La figureII.4 montre les dégradations dues à Stirmark.

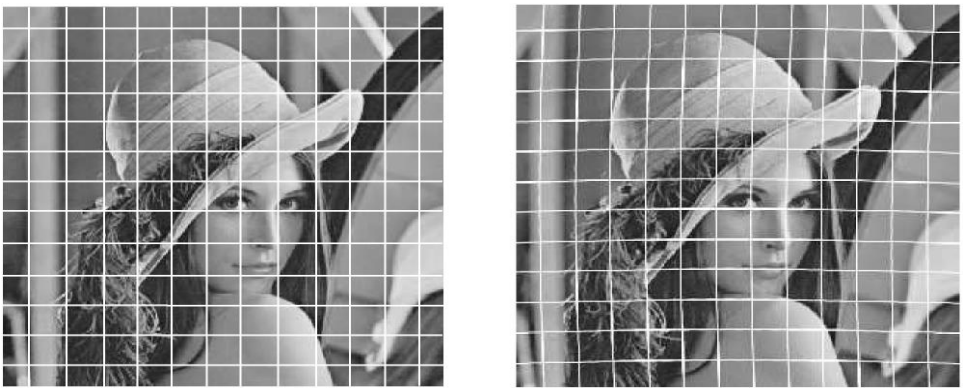


Figure II.5 : Dégradations de l'image après le banc de tests Stirmark.

Ce banc de tests est extrêmement efficace et très peu d'algorithmes de tatouage d'images y sont robustes.

II.5.2- Banc de test de Checkmark

Checkmark contient six catégories d'attaques.

1. Débruitage : filtre de Wiener et seuillages.
2. Débruitage et modulation psycho visuelle.
3. Débruitage et attaque.
4. Attaque de copie.
5. Petites rotations et attaques de mires.
6. Compression par transformée en ondelettes.

Ce banc de test améliore celui proposé par Petit colas et les algorithmes de tatouage ne sont pas testés seulement pour leur robustesse mais aussi pour

leur qualité visuelle.

II.5.3- Banc de tests Optimark

L'objectif de ce test est de donner la meilleure analyse des performances des algorithmes, pas seulement en terme de robustesse mais aussi en terme de capacité d'insertion ou de temps d'exécution.

Le banc de tests mesure les paramètres suivants :

1. Performances de détection/ décodage en utilisant de nombreux tests et différents paramètres (images, clés et messages).

2. Performances de détection pour les détecteurs qui identifient une séquence binaire : probabilité de fausse alarme, taux d'erreur, courbes caractéristiques.

3. Performances de détection pour les détecteurs qui identifient uniquement la présence d'une signature.

4. Temps de calcul pour l'insertion d'une marque et pour sa détection.

5. Capacité d'insertion de l'algorithme.

6. L'algorithme limite pour chaque attaque (les réglages minimums à faire pour être robuste à l'attaque).

II.5.4- Banc de tests Certimark[30]

Avec le développement croissant de nombreux algorithmes de tatouage, la nécessité de des normes internationales devint une priorité pour la communauté du tatouage d'images.

Le projet européen Certimark [32] (certification for watermarking techniques) débuté en 2000 et arrivé à son terme fin 2002 a pour but d'obtenir une structure de banc de tests commune et flexible.

Ce projet a regroupé des partenaires européens académiques et industriels.

Les objectifs atteints [32] étaient les suivants :

1. Désigner développer et publier un banc de tests complet pour les schémas de tatouage.

2. Faire de ce banc de tests un outil de référence aussi bien pour les fournisseurs de technologies que pour les consommateurs.

3. Etablir un processus de certification pour les algorithmes de tatouage.

4. permettre à la technologie européenne d'avoir une position dominante dans le domaine de la protection d'images et de vidéos dans le contexte du commerce électronique sur Internet.

II.6- Conclusion

Dans ce chapitre, nous avons présenté presque toutes les attaques que peut subir un tatouage numérique des images fixes que ce soit intentionnelles ou non intentionnelles et nous avons vu la diversité de ces méthodes. Pour notre cas, nous allons utiliser seulement les attaques géométriques pour mesurer les performances de notre tatouage (robustesse et invisibilité). Le chapitre suivant sera consacré à l'étude du tatouage numérique des images fixes par la transformée DCT.

IV.1- Introduction

La transformée en ondelettes discrète TWD (de l'anglais : DWT discret wavelet transform), est une technique inventée par Jean Morlet, un ingénieur Français pour la prospection pétrolière. [11] C'est une technique qui nous permet aussi le passage du domaine spatial au domaine fréquentiel.

Si la transformée DCT donne des informations sur l'amplitude des fréquences, la transformée par ondelettes DWT donne l'amplitude de la fréquence à un endroit donné de l'axe du temps, ce qui est plus proche de la perception humain. Dans le cas d'une image de résolution 2^K, la transformée par ondelettes DWT donne quatre sous images de résolution 2^{K-1}. [40]

Dans ce chapitre, nous allons présenter une étude du tatouage des images à base de la transformée en ondelettes DWT, ainsi que les résultats de test de performances de cette technique sur un certain nombre d'images fixes.

Notre étude sera limitée à la transformée en ondelettes de HAAR.

IV.2 – Les notions de la transformée en ondelettes

IV.2.1 –Définition [2]

La transformée en ondelettes est une méthode de traitement du signal adaptée aux signaux non périodiques, elle permet une analyse temps- fréquence. Contrairement à l'analyse par Fourier, les algorithmes d'ondelettes ont deux composantes appelées coefficients d'approximations et coefficients de détails. Dans le plus part des cas, lorsque il s'agit d'un traitement d'image, nous utilisons les ondelettes de Haar [25].

IV.2.2 –Les notions de la décomposition par les filtres en quadratures [17]

Le principe de décomposition consiste à prendre une image et la faire convoluées par deux filtres : un filtre passe haut et un filtre passe bas. On obtient donc quatre sous images (voir figure IV.1).

La partie LL, correspondant à la convolution par le filtre passe bas (horizontalement et verticalement), appelée aussi zone des cœfficients d'approximations là ou l'énergie de l'image est concentrée.

Les autres parties sont HL, HH, LH appelées aussi respectivement zone des cœfficients de détail horizontal, de détail diagonal, et de détail vertical. La figure IV.1 détaillera clairement cette décomposition [1].

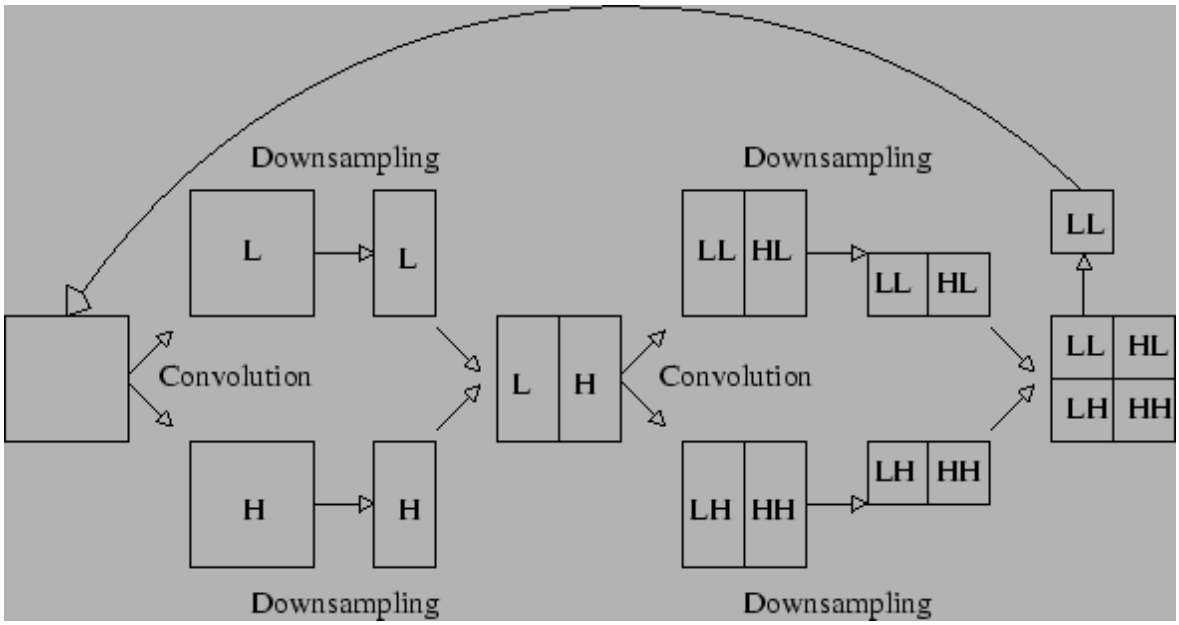


Figure IV.1 : Décomposition de l'image par les filtres en quadrature.

IV.3-Les ondelettes de HAAR [40]

IV.3.1-Procedure de calcul

Pour procéder à cette transformation, on distingue quatre étapes différentes

3) Moyenner les pixels de l'image originale deux a deux suivant l'axe horizontal :

$$H(x)= (X_{n} + X_{n+1}) / 2 \quad \textbf{(IV.1)}$$

On obtient des fréquences deux fois plus basses.

2) Calculer l'erreur entre l'image originale et l'image sous échantillonnée dans le sens horizontal .

$$G(x)= (X_n - X_{n+1}) / 2$$

(IV.2)

4) Pour chacune des deux images intermédiaires, moyenner les pixels deux a deux suivant l'axe vertical.

$$H(y)= (Y_n + Y_{n+1}) / 2$$

(IV.3)

4) Pour chacune des deux images intermédiaires, calculer l'erreur suivant l'axe vertical.

$$G(y)= (Y_n - Y_{n+1}) / 2$$

(IV.4)

L'application de cette procédure de HAAR sur l'image du chat de la figure IV.2 donne la transformée de HAAR à différents niveaux représentée sur les figures IV.3, IV.4, IV.5.



Figure IV.2 : Image du chat.

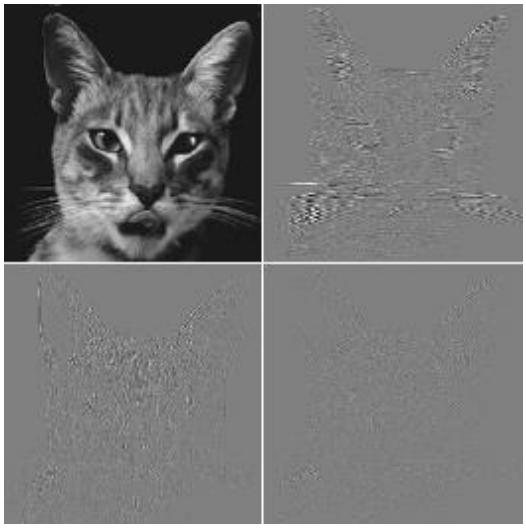


Figure IV.3 : Transformation DWT de HAAR (niveau 1) du chat.

Le résultat de l'application de la procédure de HAAR au niveau1 est une image d'approximation qui a une résolution divisée par deux et trois images de détails qui donnent les hautes fréquences de l'image originale.

Cette transformation est répétée autant de fois que nécessaire pour obtenir le nombre voulu de sous bandes.

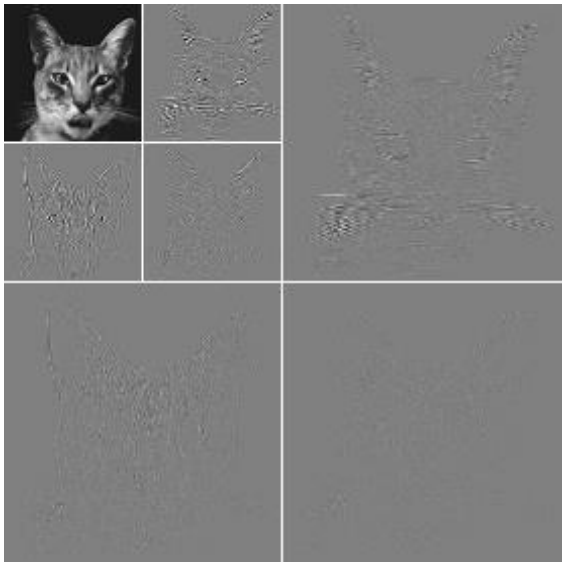


Figure IV.4 : Transformation DWT de HAAR (niveau 2) de l'image du chat.

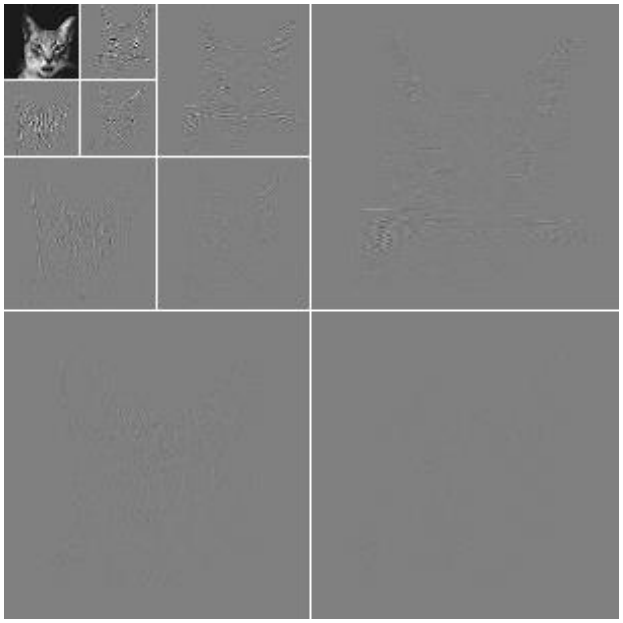
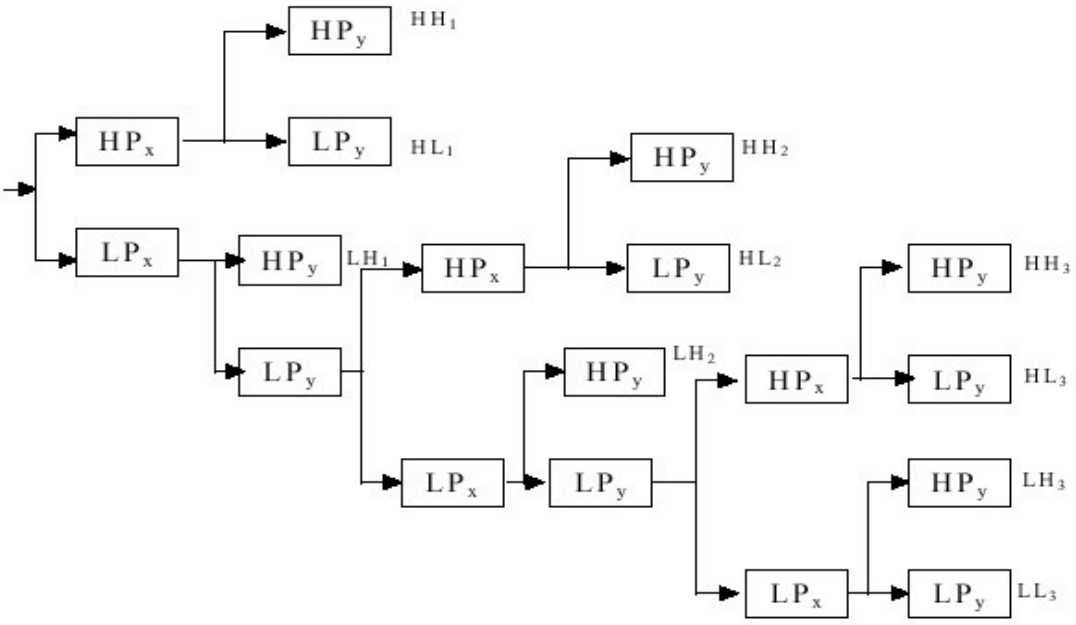


Figure IV.5 : Transformation DWT de HAAR (niveau 3) de l'image du chat.

Remarque [8] :

Dans le cas de la transformée DCT, les fréquences varient de façon linéaire, tandis que dans le cas de la transformée en ondelettes DWT les fréquences varient de manière logarithmique.



P0

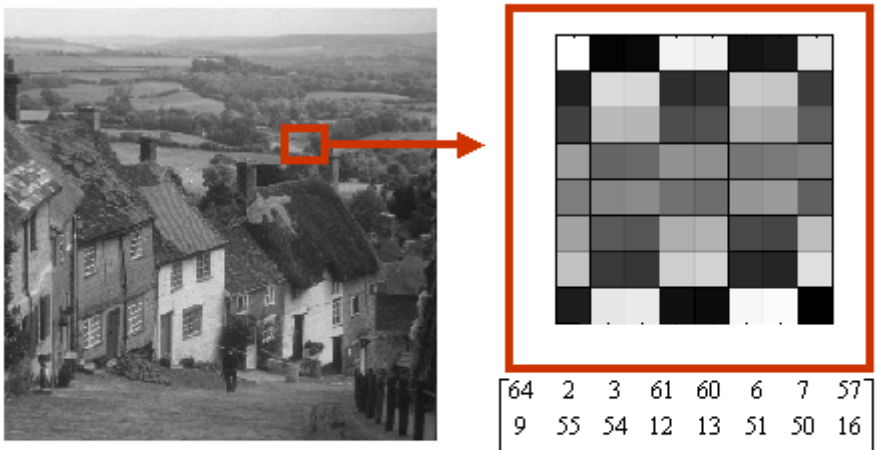
Figure IV.6 : Procédure de décomposition de Haar au niveau 3.

IV.4 Exemple de calcul de la transformée DWT par la méthode de Haar

Pour bien éclaircir les étapes décrites dans la section précédente pour le calcul de la transformée de HAAR par DWT, prenons l'exemple de la figure IV.6, et appliquant les quatre étapes sur un bloc de pixels 8x8.

1. Calculons la moyenne de chaque paire des cœfficients de chaque ligne (n/2moyennes).
2. Calculons la différence de chaque paire des cœfficients utilisés précédemment (n/2 différences).
3. Placer les moyennes dans la première moitié de la matrice B_1
4. Placer la différence dans la seconde moitié de la matrice B_1
5. Répétons la procédure sur la matrice résultante sur les colonnes.

Nous portons les résultats sur les matrices B_1, B_2, B_3, et B_4 comme illustrer dans la page suivante [34].



$$\begin{bmatrix} 64 & 2 & 3 & 61 & 60 & 6 & 7 & 57 \\ 9 & 55 & 54 & 12 & 13 & 51 & 50 & 16 \\ 17 & 47 & 46 & 20 & 21 & 43 & 42 & 24 \\ 40 & 26 & 27 & 37 & 36 & 30 & 31 & 33 \\ 32 & 34 & 35 & 29 & 28 & 38 & 39 & 25 \\ 41 & 23 & 22 & 44 & 45 & 19 & 18 & 48 \\ 49 & 15 & 14 & 52 & 53 & 11 & 10 & 56 \\ 8 & 58 & 59 & 5 & 4 & 62 & 63 & 1 \end{bmatrix}$$

Figure IV.7: Un bloc de pixel tiré d'une image. [34]

Etape1) : Calculons les moyennes des pixels du bloc 8x8 de la figure IV.6.

Nous obtenons :

$$B_1 = \begin{bmatrix} 33 & 32 & 33 & 32 & . & . & . & . \\ 32 & 33 & 32 & 33 & . & . & . & . \\ 32 & 33 & 32 & 33 & . & . & . & . \\ 33 & 32 & 33 & 32 & . & . & . & . \\ 33 & 32 & 33 & 32 & . & . & . & . \\ 32 & 33 & 32 & 33 & . & . & . & . \\ 32 & 33 & 32 & 33 & . & . & . & . \\ 33 & 32 & 33 & 32 & . & . & . & . \end{bmatrix}$$

Etape2 : calculons l'erreur entre le bloc original et le bloc sous échantillonné dans le sens horizontal :les résultats obtenus vont remplacer les points

$$\begin{bmatrix} 33 & 32 & 33 & 32 & 31 & -29 & 27 & -25 \\ 32 & 33 & 32 & 33 & -23 & 21 & -19 & 17 \\ 32 & 33 & 32 & 33 & -15 & 13 & -11 & 09 \\ 33 & 32 & 33 & 32 & 07 & -5 & 3 & -1 \\ 33 & 32 & 33 & 32 & -1 & 03 & -5 & 07 \end{bmatrix}$$

B_2= **32 33 32 33 09 -11 13 -15**

32 33 32 33 17 -19 21 -23

33 32 33 32 -25 27 -29 31

Etape3 : Calculons les moyennes des pixels du bloc 8x8 résultant dans le sens vertical

$$B_3 = \begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \end{bmatrix}$$

Etape4 : Calculons l'erreur dans le sens vertical

$$B_4 = \begin{bmatrix} 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 32.5 & 32.5 & 32.5 & 32.5 & 4 & -4 & 4 & -4 \\ 32.5 & 32.5 & 32.5 & 32.5 & -4 & 4 & -4 & 4 \\ 0.5 & -0.5 & 0.5 & -0.5 & 27 & -25 & 23 & -21 \\ -0.5 & 0.5 & -0.5 & 0.5 & -11 & 09 & -7 & 05 \\ 0.5 & -0.5 & 0.5 & -0.5 & -5 & 07 & -9 & 11 \end{bmatrix}$$

-0.5 0.5 -0.5 0.5 21 -22 25 -27

B_4 : représente la matrice transformée DWT du bloc B_1.

Le processus est réversible, on peut calculer la IDWT(la transformée en ondelettes discrète inverse)qui va nous permettre de revenir au domaine spatial une fois notre traitement est terminé.

Dans le sens horizontal :

$$X_n = H(x) + G(x) \quad \textbf{(IV.5)}$$

$$X_{n+1} = H(x) - G(x) \quad \textbf{(IV.6)}$$

Dans le sens vertical :

$$Y_n = H(y) + G(y) \quad \textbf{(IV.7)}$$

$$Y_{n+1} = H(y) - G(y) \quad \textbf{(IV.8)}$$

IV.5- Algorithme de tatouage utilisant la DWT [17] [41] [3]

L'algorithme développée se compose de deux phases : une phase d'insertion et une phase d'extraction.

IV.5.1- Phase d'insertion

Soit à insérer une marque désignée par (w), dans une image originale désignée par (P_O). Soient g : les coefficients par la transformée DWT au niveau (1) de l'image originale Lena (P_O) de la figure IV.8



a) Image originale Lena. **b) Transformée de Haar de l'image Lena (niveau 1).**

Figure IV.8: Image originale (P_O) et sa transformée DWT (niveau 1).

w : sont les éléments de la matrice marque (P_w) sans subir de transformation de Haar.

L'insertion se fait selon la formule de tatouage suivante :

$$g^* = g + \alpha \cdot w \quad \textbf{(IV.9)}$$

Où g* : est l'image tatouée dans le domaine fréquentiel.

α : la force de tatouage.

La figure IV.8 illustre bien cette phase.

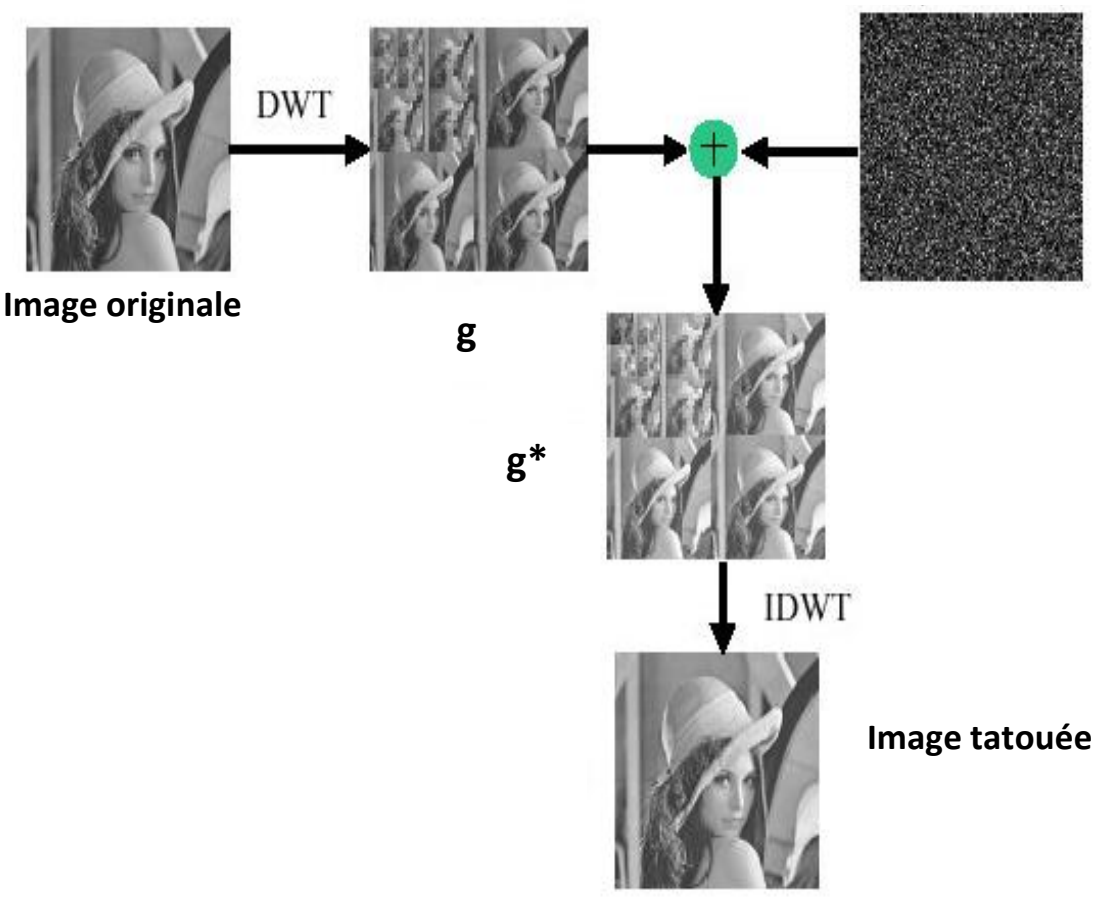


Figure IV.9 : Insertion de la marque.

Pour reconstruire l'image tatouée dans le domaine spatial, nous utilisons la transformée en ondelettes inverse (IDWT).

$$P_w = IDWT(g^*)$$

IV.5.2- Organigramme d'insertion

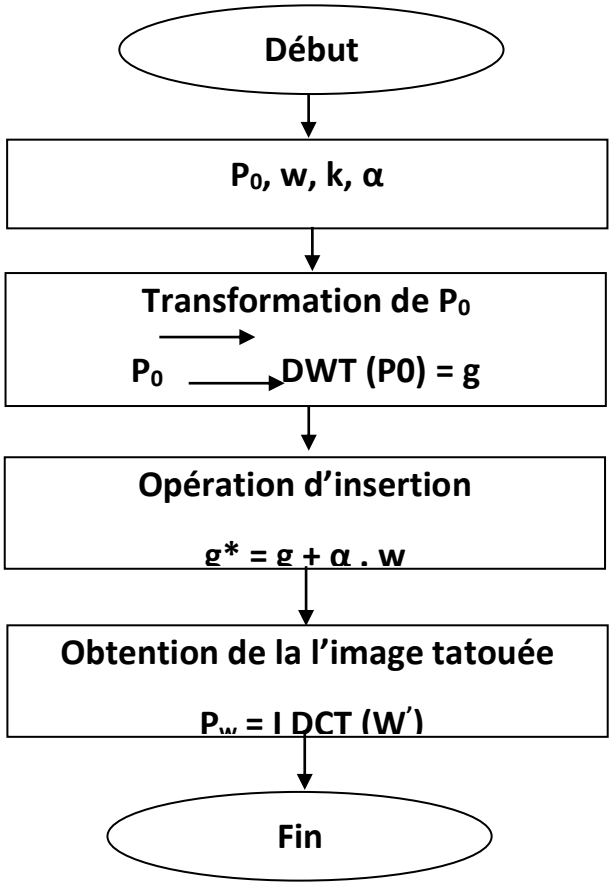


Figure IV.10 : Organigramme d'insertion de la marque.

IV.5.3- Phase d'extraction

Pour l'extraction on dispose des paramètres suivants :

L'image originale (P_o)

L'image tatouée (P_w)

La force de tatouage :α

L'endroit où l'emplacement de la marque dans l'image originale.

L'algorithme d'extraction se fait selon le schéma de la figure IV.10.

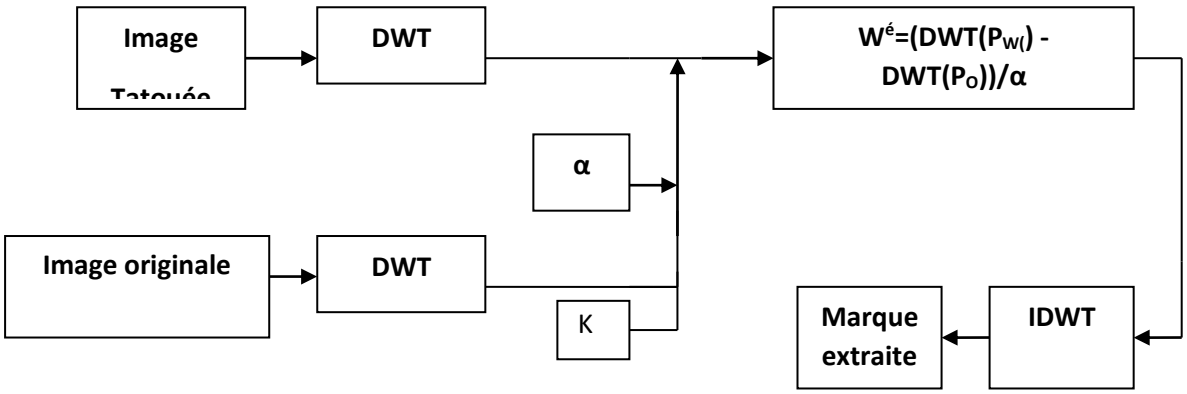


Figure IV.11 : Schéma d'extraction de la marque par DWT.

On applique la DWT à (P_o) et (P_w), ce qui donne g et g*. La marque extraite est déduite a partir de la formule suivante :

$$W' = \frac{g^* - g}{\alpha} \qquad \textbf{(IV.10)}$$

Pour avoir la marque reconstruite,on applique la IDWT

w'= IDWT (W')

Remarque : On doit différencier entre w qui est la marque originale, et

w'qui est la marque reconstruite.

IV.5.4- Organigramme d'extraction

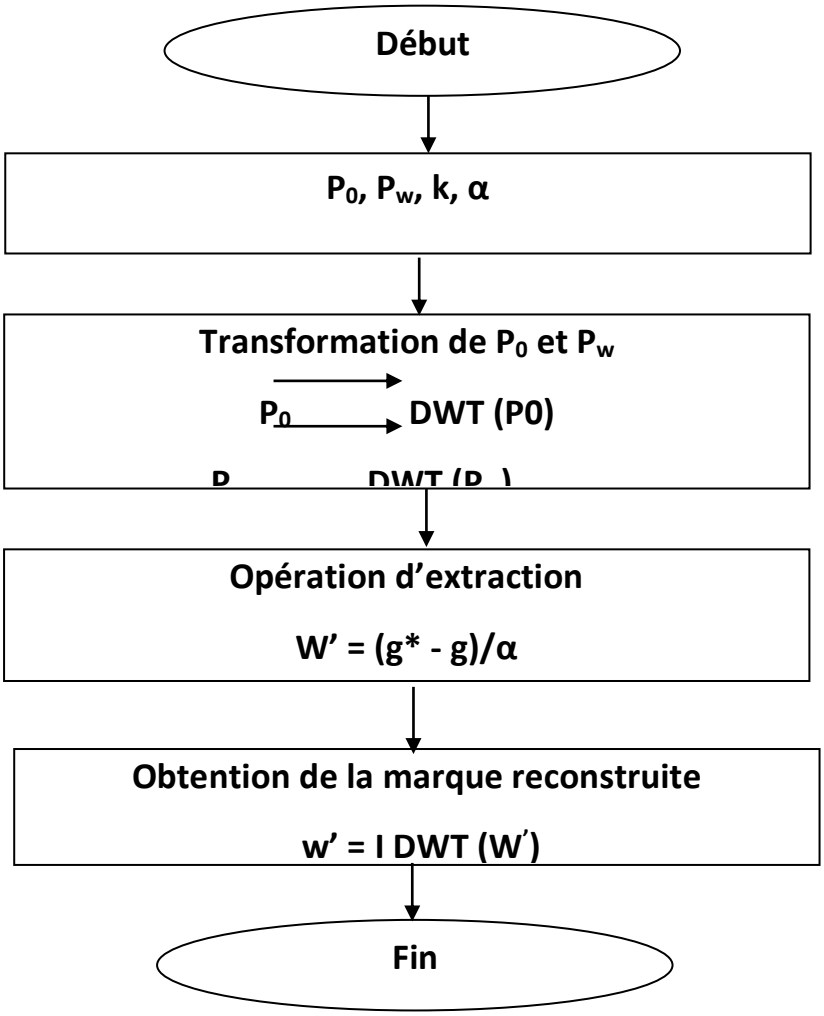

Début
P0, Pw, k, α
Transformation de P0 et Pw
P0
DWT (P0)
Opération d'extraction
W' = (g* - g)/α
Obtention de la marque reconstruite
w' = I DWT (W')
Fin



Figure IV.12 : Organigramme d'extraction de la marque.

IV.6- Test de la méthode de tatouage par DWT

La méthode de tatouage par DWT a été implémentée sur Matlab 7.1.

Pour pouvoir faire une comparaison fiable, on doit garder le même environnement (images) utilisé dans le cas de la DCT.

(Po) est l'image originale de (Lena) de taille 256x256 utilisée au format bmp. La marque (w) à insérer est de taille 64x64, utilisée au format bmp, elle porte mon nom et prénom et ma willaya.





Figure IV. 13.a : Image originale **Figure IV.13.b : Marque à insérer**

IV.6.1- Test de la méthode de tatouage par DWT sans attaques

(Image de Lena)

IV.6.1-Avec α =0.02





Figure IV. 14.1 : Image tatouée avec α =0.02

Figure IV.15.1 : Marque extraite avec α =0.02

$PSNR_1$= 40.9011dB

$PSNR_2$= 77.1236dB

IV.6.1.2-Avec α =0.05





Figure IV.14.2 : Image tatouée avec α =0.05

Figure IV.15.2 : Marque extraite avec α =0.05

$PSNR_1$= 32.6016dB

$PSNR_2$= 62.1478dB

IV.6.1.3-Avec α =0.1





Figure IV.14.3 : Image tatouée avec α =0.1

Figure IV.15.3 : Marque extraite avec α =0.1

$PSNR_1$= 26.7483dB

$PSNR_2$= 57.6080 dB

IV.6.1.4-Avec α =0.2





Figure IV.14.4 : Image tatouée avec α =0.2

Figure IV.15.4 : Marque extraite avec α =0.2

$PSNR_1$= 20.8177dB

$PSNR_2$= 31.1844 dB

IV.6.1.5-Avec α =0.3





Figure IV.14.5 : Image tatouée avec α =0.3

Figure IV.15.5 : Marque extraite avec α =0.3

$PSNR_1$= 17.5927dB

$PSNR_2$= 21.0909 dB

IV.6.1.6-Avec α =0.4



Figure IV.14.6 : Image tatouée avec α =0.4



Figure IV.15.6 : Marque extraite avec α =0.4

$PSNR_1$= 15.3850dB

$PSNR_2$= 17.5735 dB

IV.6.1.7-Avec α =0.5





Figure IV.14.7 : Image tatouée avec α =0.5

Figure IV.15.7 : Marque extraite avec α =0.5

$PSNR_1$= 13.9667dB

$PSNR_2$= 15.2609 dB

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
$PSNR_1$ en dB	409011	32.6016	26.7483	20.8177	17.5927	15.3850	13.9667
$PSNR_2$ en dB	77.1236	62.1478	57.6080	31.1844	21.0909	17.5735	15.2609

Tableau IV.1 : Tableau récapitulatif du $PSNR_1$ et $PSNR_2$ en fonction de α.

IV.6.2-Comportement du tatouage par DWT vis-à-vis les attaques

IV.6.2.A-Attaque par compression JPEG [27]

IV.6.2.A.1-Avec α =0.02





Figure IV.16.1 : Image compressée avec α =0.02

Figure IV.17.1 : Marque extraite avec α =0.02

$PSNR_1$= 34.9513dB

$PSNR_2$= 06.1983 dB

IV.6.2.A.2-Avec α =0.05





Figure IV.16.2 : Image compressée avec α =0.05

Figure IV.17.2 : Marque extraite avec α =0.05

$PSNR_1$= 30.1239dB

$PSNR_2$= 12.5638 dB

IV.6.2.A.3-Avec α =0.1





Figure IV.16.3 : Image compressée avec α =0.1

Figure IV.17.3 : Marque extraite avec α =0.1

$PSNR_1$= 26. 7517dB

$PSNR_2$= 15.0381 dB

IV.6.2.A.4-Avec α =0.2





Figure IV.16.4 : Image compressée avec α =0.2

Figure IV.17.4 : Marque extraite avec α =0.2

$PSNR_1$= 20. 5053dB

$PSNR_2$= 21.8773 dB

IV.6.2.A.5-Avec α =0.3





Figure IV.16.5 : Image compressée avec α =0.3

Figure IV.17.5 : Marque extraite avec α =0.3

$PSNR_1$= 17. 5188dB

$PSNR_2$= 19.8234 dB

IV.6.2.A.6-Avec α =0.4





Figure IV.16.6 : Image compressée avec α =0.4

Figure IV.17.6 : Marque extraite avec α =0.4

$PSNR_1$= 15. 4195dB

$PSNR_2$= 16.9475 dB

IV.6.2.A.7-Avec α =0.5





Figure IV.16.7 : Image compressée avec α =0.5

Figure IV.17.7 : Marque extraite avec α =0.5

$PSNR_1$= 14. 0186dB

$PSNR_2$= 14.6924 dB

IV.6.2.B-Attaque par rotation de 15º

IV.6.2.B.1- Avec α =0.02



Figure IV.18.1 : Image après rotation

avec α=0.02

$PSNR_1$ = 9.6374dB



Figure IV.19.1 :Marque extraite

avec α=0.02

$PSNR_2$ = 2.1191dB

IV.6.2.B.2- Avec α =0.05





Figure IV.18.2 : Image après rotation avec α=0.05

Figure IV.19.2 :Marque extraite avec α=0.05

$PSNR_1$ = 9.6277dB

$PSNR_2$ = 1.8536dB

IV.6.2.B.3- Avec α =0.1





Figure IV.18.3 : Image après rotation **Figure IV.19.3 :Marque extraite**

avec α=0.1 **avec α=0.1**

$PSNR_1$ = 9.5955dB $PSNR_2$ = 1.6811dB

IV.6.2.B.4- Avec α =0.2





Figure IV.18.4 : Image après rotation **Figure IV.19.4 :Marque extraite**

avec α=0.2 **avec α=0.2**

$PSNR_1$ = 9.4759dB $PSNR_2$ = 1.5520dB

IV.6.2.B.5- Avec α =0.3





Figure IV.18.5 : Image après rotation avec α=0.3

Figure IV.19.5 :Marque extraite avec α=0.3

$PSNR_1$ = 9.3251dB

$PSNR_2$ = 1.4958dB

IV.6.2.B.6- Avec α =0.4





Figure IV.18.6 : Image après rotation avec α=0.4

Figure IV.19.6 :Marque extraite avec α=0.4

$PSNR_1$ = 9.1480dB

$PSNR_2$ = 1.4610dB

IV.6.2.B.7- Avec α =0.5





Figure IV.18.7 : Image après rotation avec α=0.5

FigureIV.19.7 :Marque extraite avec α=0.5

$PSNR_1$ = 8.9950dB

$PSNR_2$ = 1.4344dB

IV.6.2.C-Attaque par redimensionnement (resizing) de 1.25

IV.6.2.C.1- Avec α =0.02





Figure IV.20.1 : Image après resizing avec α=0.02

Figure IV.21.1 :Marque extraite avec α=0.02

$PSNR_1$ = 12.2695dB

$PSNR_2$ = 1.9596 dB

IV.6.2.C.2- Avec α =0.05





Figure IV.20.2 : Image après resizing extraite

avec α=0.05

$PSNR_1$ = 12.2544dB

Figure IV.21.2 :Marque

avec α=0.05

$PSNR_2$ = 1.7200 dB

IV.6.2.C.3- Avec α =0.1





Figure IV.20.3 : Image après resizing extraite

avec α=0.1

$PSNR_1$ = 12.2038dB

Figure IV.21.3 :Marque

avec α=0.1

$PSNR_2$ = 1.5950 dB

IV.6.2.C.4- Avec α =0.2



Figure IV.20.4 : Image après resizing avec α=0.2



Figure IV.21.4 :Marque extraite avec α=0.2

$PSNR_1$ = 12.0051dB

$PSNR_2$ = 1.5101 dB

IV.6.2.C.5- Avec α =0.3





Figure IV.20.5 : Image après resizing extraite

avec α=0.3

$PSNR_1$ = 11.7771dB

Figure IV.21.5 :Marque

avec α=0.3

$PSNR_2$ = 1.4716 dB

IV.6.2.C.6- Avec α =0.4





Figure IV.20.6 : Image après resizing extraite

avec α=0.4

$PSNR_1$ = 11.5177dB

Figure IV.21.6 :Marque

avec α=0.4

$PSNR_2$ = 1.4439 dB

IV.6.2.C.7- Avec α =0.5



Figure IV.20.7 : Image après resizing



Figure IV.21.7 :Marque extraite

avec α=0.5

$PSNR_1$ = 11.2924dB

avec α=0.5

$PSNR_2$ = 1.4264 dB

IV.6.2.D-Attaque par filtrage passe-bas (Gaussien, [3 3], sigma=0.5)

IV.6.2.D.1- Avec α =0.02





Figure IV.22.1 : Image après filtrage extraite

avec α=0.02

$PSNR_1$ = 38.5897dB

Figure IV.23.1 :Marque

avec α=0.02

$PSNR_2$ =20.8553 dB

IV.6.2.D.2- Avec α =0.05





Figure IV.22.2 : Image après filtrage avec α=0.05

Figure IV.23.2 :Marque extraite avec α=0.05

$PSNR_1$ = 37.1457dB

$PSNR_2$ =14.3954 dB

IV.6.2.D.3- Avec α =0.1





Figure IV.22.3 : Image après filtrage avec α=0.1

Figure IV.23.3 :Marque extraite avec α=0.1

$PSNR_1$ = 34.2322dB

$PSNR_2$ =22.3435 dB

IV.6.2.D.4- Avec α =0.2





Figure IV.22.4 : Image après filtrage extraite

avec α=0.2

$PSNR_1$ = 29.5385dB

Figure IV.23.4 :Marque

avec α=0.2

$PSNR_2$ =23.2922 dB

IV.6.2.D.5- Avec α =0.3





Figure IV.22.5: Image après filtrage extraite

avec α=0.3

$PSNR_1$ = 26.2994dB

Figure IV.23.5 :Marque

avec α=0.3

$PSNR_2$ =24.1015 dB

IV.6.2.D.6- Avec α =0.4





Figure IV.22.6: Image après filtrage avec α=0.4

Figure IV.23.6 :Marque extraite avec α=0.4

$PSNR_1$ = 23.7848dB

$PSNR_2$ =24.9176 dB

IV.6.2.D.7- Avec α =0.5





Figure IV.22.7: Image après filtrage avec α=0.5

$PSNR_1$ = 21.8613dB

Figure IV.23.7 :Marque extraite avec α=0.5

$PSNR_2$ =25.6653 dB

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	**34.9513**	**30.1239**	**26.7517**	**20.5053**	**17.5188**	**15.4195**	**14.018**
Redimensionnement	**12.2695**	**12.2544**	**12.2038**	**12.0051**	**11.7771**	**11.5177**	**11.292**
Filtrage passe-bas	**38.5897**	**37.1457**	**34.2322**	**29.5385**	**26.2994**	**23.7848**	**21.861**
Rotation	**9.6374**	**9.6277**	**9.5955**	**9.7459**	**9.3251**	**9.1480**	**8.995**

Tableau IV.2 : Tableau récapitulatif du $PSNR_1$ en dB vis à vis les différentes attaques par DWT.

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	**6.1983**	**12.5638**	**15.0381**	**21.8773**	**19.8234**	**16.9475**	**14.692**
Redimensionnement	**1.9596**	**1.7200**	**1.5950**	**1.5101**	**1.4716**	**1.4439**	**1.4264**
Filtrage passe-bas	**14.3954**	**20.8553**	**22.3435**	**22.2922**	**18.1015**	**17.9176**	**15.665**
Rotation	**2.1191**	**1.8536**	**1.6811**	**1.5520**	**1.4958**	**1.4610**	**1.4344**

Tableau IV.3 : Tableau récapitulatif du $PSNR_2$ en dB vis à vis les différentes attaques par DWT.

IV.6.3-Commentaires

1. En se référant au tableau IV.1, nous remarquons que les valeurs du $PSNR_1$ et $PSNR_2$ sont inversement proportionnelles avec la force de tatouage α. Nous remarquons aussi que la marque est extraite d'une façon très claire pour les différentes valeurs de α, elle est presque la même que la marque originale, ce qui est vérifié par les valeurs du $PSNR_2$ qui varie entre [77.1236-15.2609], cela d'une part.D'autre part, nous remarquons que l'image tatouée n'a subi aucune dégradation avec un $PSNR_1$ variant entre [40.9611-13.9667] ; ce qui qualifie ce tatouage de robuste en moyenne.

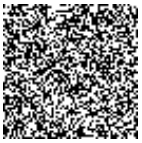
2. Nous observons aussi que la marque sur l'image tatouée commence à apparaître (a partir de $\alpha = 0.4$) ; et elle est invisible pour les valeurs de $\alpha < 0.4$, ce qui qualifie ce type de tatouage d'invisible.

3. En se référant aux tableaux VI.II et VI.III, nous remarquons que ce type de tatouage avec DWT est fragile devant les attaques géométriques (rotation et resizing), suite aux dégradations remarquées sur la marque extraite et selon les valeurs du $PSNR_1$ et $PSNR_2$ (trop faibles).En ce qui concerne l'attaque par compression JPEG, et filtrage passe-bas, ce tatouage par DWT est moyennement robuste en se basant sur l'état de la marque extraite, et les valeurs du $PSNR_1$ et $PSNR_2$(des valeurs modérées).

IV.6.4- Test de la méthode de tatouage par DWT sans attaques

(Image de Hamza)





N

(0,1)

Figure IV.24 : Image originale de Hamza.

Figure IV.25 : Marque à Insérer.









(1) **(2)** **(3)** **(4)**



(5) **(6)** **(7)**

Figure IV.26 : Image tatouée de Hamza.

(1) Avec α=0.02.
(2) Avec α=0.05.
(3) Avec α=0.1.
(4) Avec α=0.2.
(5) Avec α=0.3.
(6) Avec α=0.4.
(7) Avec α=0.5.



(1) (2) (3) (4) (5) (6) (7)

Figure IV.27 : Marque extraite sans attaques.

(1) **Avec** α=0.02.
(2) **Avec** α=0.05.
(3) **Avec** α=0.1.
(4) **Avec** α=0.2.

(5) **Avec** α=0.3.
(6) **Avec** α=0.4.
(7) **Avec** α=0.5.

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
$PSNR_1$ en dB	54.6165	46.3413	40.6272	35.0858	32.0606	30.0195	28.4603
$PSNR_2$ en dB	37.1130	41.1671	25.8495	19.3332	16.3654	14.5862	13.5381

Tableau IV.4 : Tableau récapitulatif du $PSNR_1$ et $PSNR_2$ en fonction de α (Image de Hamza).

IV.6.5-Comportement du tatouage par DWT vis-à-vis des attaques (Image de Hamza)

IV.6.5.A-Attaque par compression JPEG [27]



(1) **(2)** **(3)** **(4)**



(5) **(6)** **(7)**

Figure IV.28 : Image tatouée compressée de Hamza.



(1) (2) (3) (4) (5) (6) (7)

Figure IV.29 : Marque extraite après compression.

(1) Avec α=0.02.
(2) Avec α=0.05.
(3) Avec α=0.1.
(4) Avec α=0.2.
(5) Avec α=0.3.
(6) Avec α=0.4.
(7) Avec α=0.5.

IV.6.5.B-Attaque par rotation de 15º



(1) (2) (3)





(4) (5) (6)



(7)

Figure IV.30 : Image tatouée après rotation de 15°.



(1) (2) (3) (4) (5) (6) (7)

Figure IV.31 : Marque extraite après rotation de 15°.

IV.6.5.C-Attaque par redimensionnement (resizing) de 1.25



(1)



(2)



(3)



(4)



(5)

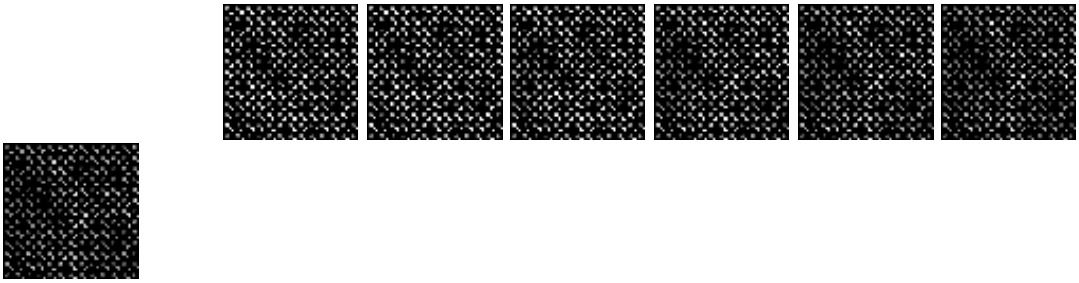


(6)



(7)

Figure IV.32 : Image tatouée après redimensionnement.



(1) (2) (3) (4) (5) (6) (7)

Figure IV.33 : Marque extraite après redimensionnement.

(1) Avec α=0.02.
(2) Avec α=0.05.
(3) Avec α=0.1.
(4) Avec α=0.2.
(5) Avec α=0.3.
(6) Avec α=0.4.
(7) Avec α=0.5.

IV.6.5.D- Attaque par filtrage passe-bas (Gaussien, [3 3], sigma=0.5)



(1) **(2)** **(3)**

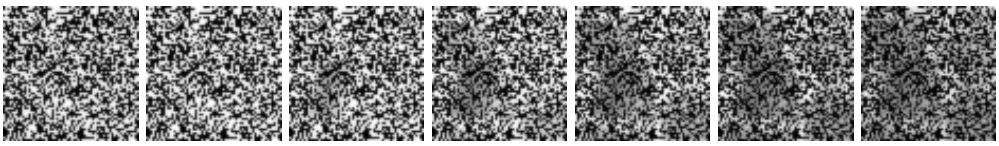


(4) **(5)** **(6)**



(7)

Figure IV.34 : Image tatouée après filtrage passe bas.



(1) **(2)** **(3)** **(4)** **(5)** **(6)** **(7)**

Figure IV.35 : Marque extraite après filtrage passe-bas.

(1) Avec α=0.02.
(2) Avec α=0.05.
(3) Avec α=0.1.
(4) Avec α=0.2.
(5) Avec α=0.3.
(6) Avec α=0.4.
(7) Avec α=0.5.

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	**36.6583**	**36.6570**	**36.2956**	**35.3160**	**33.8198**	**32.5078**	**31.2893**
Redimensionnement	**54.6274**	**46.3496**	**40.6228**	**35.0690**	**32.0198**	**29.9584**	**28.3782**
Filtrage passe-bas	**46.2456**	**45.5878**	**43.8796**	**40.1824**	**36.9838**	**34.4354**	**32.2717**

Rotation	**7.8481**	**7.8476**	**7.8462**	**7.8417**	**7.8347**	**7.8258**	**7.8148**

Tableau IV.5 : Tableau récapitulatif du $PSNR_1$ en dB vis à vis des différentes attaques.

α	**0,02**	**0,05**	**0,1**	**0,2**	**0,3**	**0,4**	**0,5**
Compression JPEG	**13.3604**	**13.1458**	**14.0997**	**14.6352**	**14.9729**	**15.0350**	**15.0023**
Redimensionnement	**3.2415**	**3.2245**	**3.2350**	**3.2728**	**3.3110**	**3.3384**	**3.3566**
Filtrage passe-bas	**14.2434**	**14.4023**	**14.2488**	**13.9490**	**13.6383**	**13.3871**	**13.2050**
Rotation	**3.5126**	**3.2680**	**3.1518**	**3.0675**	**3.0367**	**3.0199**	**3.0122**

Tableau IV.6 : Tableau récapitulatif du $PSNR_2$ en dB vis à vis des différentes attaques.

IV.6.6-Commentaires

L'exemple N° :2 avec l'image de « HAMZA » vient de confirmer les résultats déjà trouvés avec le premier exemple « image de Lena ».

1. La marque devient visible à partir de α =0.4.

2. Ce type de tatouage avec DWT est fragile devant les attaques géométriques (rotation et resizing), suite aux dégradations remarquées sur la marque extraite et selon les valeurs du $PSNR_1$ et $PSNR_2$ (trop faibles).

IV.7-Conclusion

Dans ce chapitre, nous avons présenté une étude de performance de la méthode de tatouage des images fixes à base de la transformée en ondelettes de Haar. La technique a été testée sur un certain nombre d'images comme Lena et Hamza. La robustesse du marquage a été validée à l'aide des attaques de compression JPEG, redimensionnement, rotation et filtrage passe-bas. Les résultats obtenus montrent que le tatouage par DWT est insensible à la compression JPEG, et filtrage passe-bas, tandis qu'il est fragile vis-à-vis des autres attaques. Dans le chapitre suivant, nous allons étudier une technique de tatouage combinant DCT et DWT.

V.1-Introduction

Aux chapitres trois et quatre, nous avons étudié le tatouage numérique par deux transformées différentes à savoir la (DCT) et la (DWT). Nous avons tiré aussi des résultats et des constatations concernant les performances d'un tel tatouage comme la robustesse et l'invisibilité.

Dans ce chapitre, nous avons jugé utile d'effectuer une comparaison entre ces deux approches afin de citer les avantages et les inconvénients de chaque méthode.

V.2-Comparaison entre les transformées (DCT) et (DWT)

D'après les résultats affichés sur les tableaux V.1 et V.2, nous remarquons que les valeurs du $PSNR_1$ et $PSNR_2$ avec la (DCT) sont plus élevées que celles calculées avec la (DWT) quelque soit la valeur de la force de tatouage α .

Tandis que le deuxième facteur qui est l'invisibilité est beaucoup plus remarquable avec la (DWT) qu'avec la (DCT). En conséquence il y a un compromis rencontré qui nous a poussé à poser la question suivante : est ce qu'on peut créer une méthode

combinée (DCT-DWT) qui peut produire un tatouage robuste et invisible ?

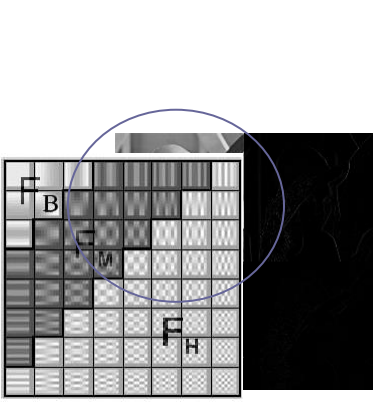
Pour répondre à cette question, nous avons développé l'algorithme décrit en V.3.

V.3- Algorithme de tatouage utilisant la méthode combinée :

Il s'agit ici de prendre une image originale (P_O) et l'a transformé à l'aide de la (DCT) en une matrice g. De même, on prend aussi la marque (w) et l'a transformé à l'aide de la (DWT) en une matrice W.

V.3.1-Phase d'insertion

Dans la phase d'insertion, nous incorporons seulement la zone d'approximation de la transformée de la marque dans les moyennes fréquences de la transformée originale (voir figure V.1).



a) Transformée de l'image Originale par DCT.

b) Transformée de la marque par DWT.

FigureV.1 : Le principe d'insertion de la marque par la méthode combinée.

L'insertion de la marque s'effectue selon l'expression suivante :

$$g^* = g + \alpha . w \quad (V.1)$$

Où g* : est l'image tatouée dans le domaine fréquentiel (domaine DCT), et α : la force de tatouage.

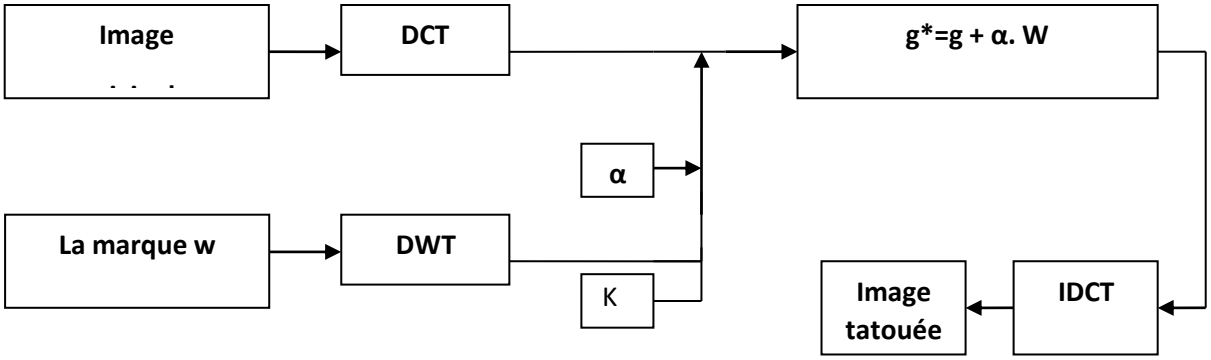
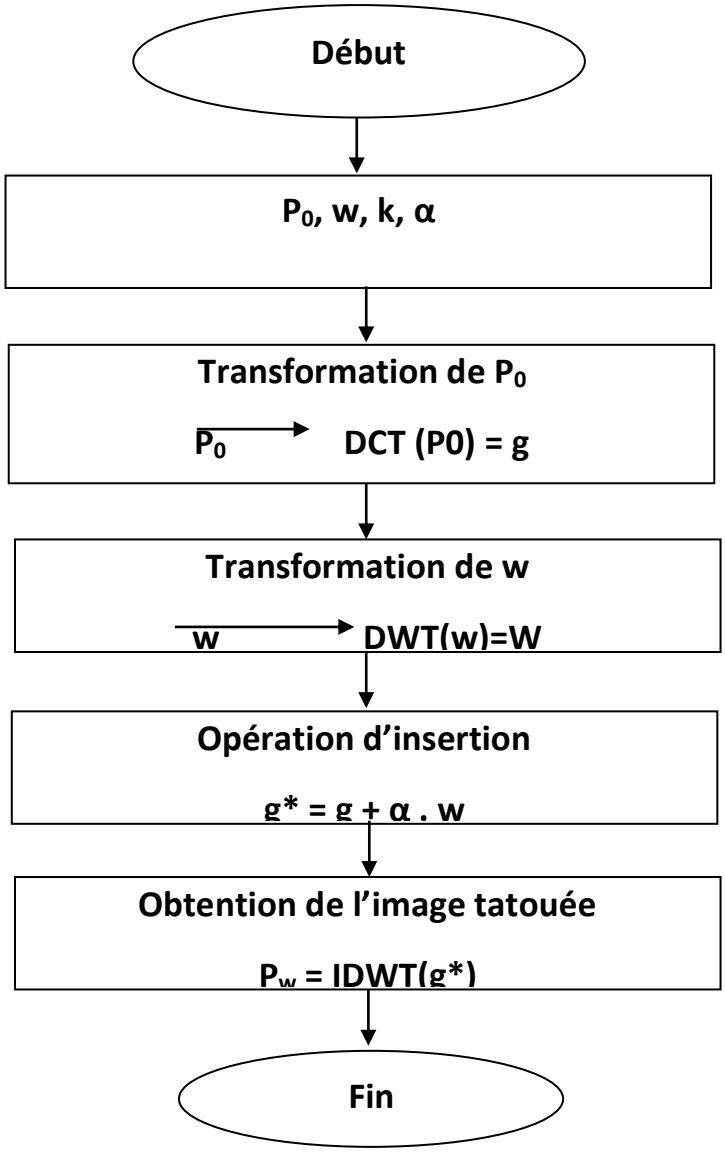


Figure V.2 : Schéma d'insertion.

L'image tatouée (P_o) est obtenue par application de la IDCT.

V.3.2- Organigramme d'insertion de la marque par la méthode combinée


Début
P0, w, k, α
Transformation de P0
P0
DCT (P0) = g
Transformation de w
w
DWT(w)=W
Opération d'insertion
g* = g + α . w
Obtention de l'image tatouée
Pw = IDWT(g*)
Fin

FigureV.3 : Organigramme d'insertion de la marque.

V.3.3 Phase d'extraction par la méthode combinée

L'extraction de la marque se fait selon la formule suivante :

$$W' = (g^* - g) / \alpha \quad \textbf{(V.2)}$$

Pour avoir la marque reconstruite,on applique la IDWT

$$w' = IDWT(W')$$

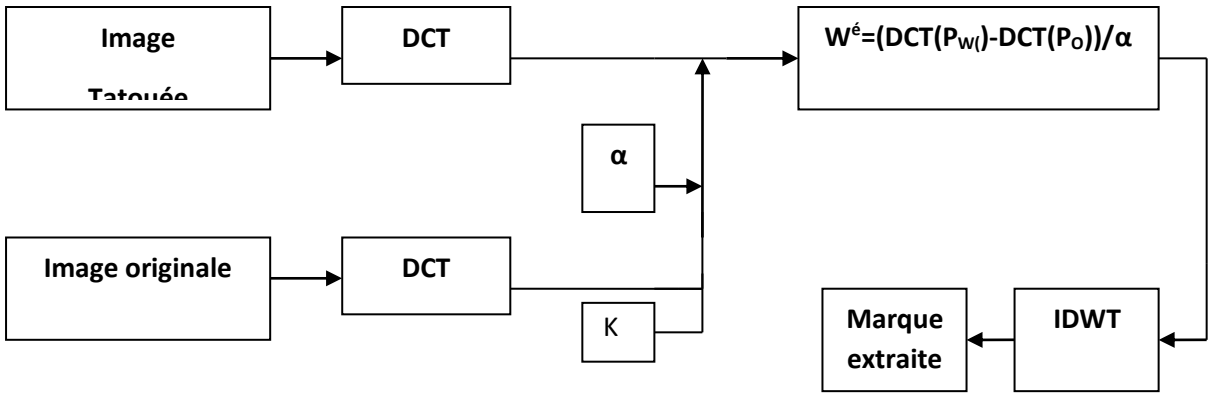


Figure V.4 : Schéma d'extraction de la marque par la méthode combinée.

V.3.4 Organigramme d'extraction par la méthode combinée

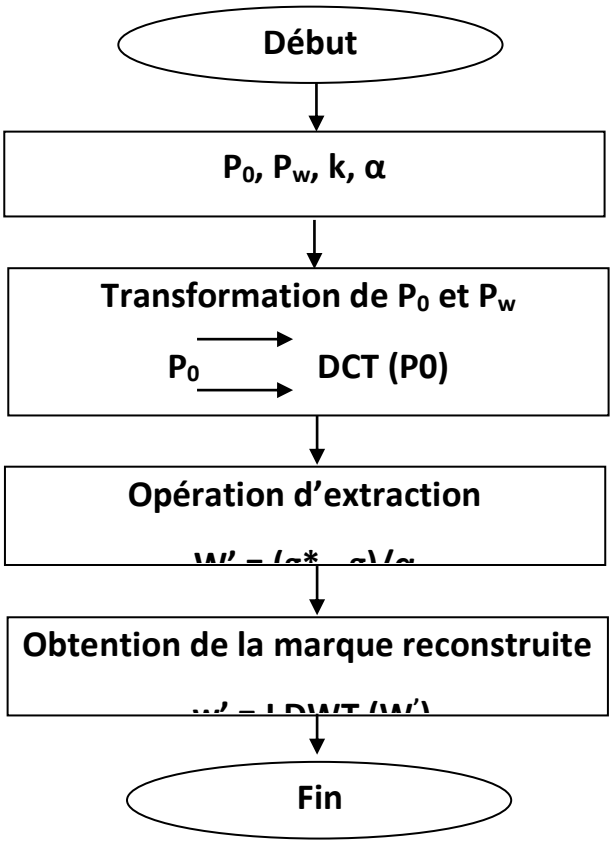


Figure V.5 : Organigramme d'extraction de la marque.

V.4-Test de la méthode proposée

Toujours, nous gardons le même environnement, en conservant les mêmes images déjà utilisées avec la DCT et la DWT. L'image originale de Lena de taille 256x256 utilisée au format bmp. La marque (w) à insérer est de taille 64x64, utilisée au format bmp, elle porte mon nom et prénom et ma willaya.

V.4.1- Test de la méthode combinée sans attaques (Image de Lena)





Figure IV. 6. a : Image originale **Figure IV.6.b : Marque à insérer**

V.4.1.1-Avec α =0.05





Figure IV.7. 1 : Image tatouée avec α =0.05

Figure IV.8. 1 : Marque extraite avec α =0.05

$PSNR_1$= 42.5263dB

$PSNR_2$= 79.2541dB

V.4.1. 2-Avec α =0.02





Figure IV.7. 2 : Image tatouée avec α =0.02

Figure IV.8. 2 : Marque extraite avec α =0.02

$PSNR_1$= 34.8956dB

$PSNR_2$= 69.5874dB

V.4.1. 3-Avec α =0.1





Figure IV.7. 3 : Image tatouée avec α =0.1

Figure IV.8. 3 : Marque extraite avec α =0.1

$PSNR_1$= 29.3215

$PSNR_2$= 68.2589dB

V.4.1. 4-Avec α =0.2





Figure IV.7. 4 : Image tatouée avec α =0.2

Figure IV.8. 4 : Marque extraite avec α =0.2

$PSNR_1$= 23.5628dB

$PSNR_2$= 41.2214dB

V.4.1. 5-Avec α =0.3





Figure IV.7.5: Image tatouée avec α =0.3

Figure IV.8. 5: Marque extraite avec α =0.3

$PSNR_1$= 22.5585dB

$PSNR_2$= 33.6535dB

V.4.1. 6-Avec α =0.4





Figure IV.7.6: Image tatouée avec α =0.4

Figure IV.8. 6: Marque extraite avec α =0.4

$PSNR_1$=18.3325dB

$PSNR_2$= 23.2201dB

V.4.1. 7-Avec α =0.5





Figure IV.7.7: Image tatouée

Figure IV.8. 7: Marque extraite

avec α =0.5 **avec α =0.5**

$PSNR_1$= 16.9266dB **$PSNR_2$= 15.6958dB**

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
$PSNR_1$ en dB	34.8956	42.5263	29.3215	23.5628	22.5585	18.3325	16.9266
$PSNR_2$ en dB	77.1236	79.2541	68.2589	41.2214	33.6535	23.2201	15.6958

Tableau V.2 : Tableau récapitulatif du $PSNR_1$ et $PSNR_2$ en fonction de α.

V.4.2-Comportement de la méthode combinée vis-à-vis les attaques

V.4.2.A- Attaque par compression JPEG

V.4.2.A.1-Avec α =0.02





Figure V.9.1: Image compressée avec α =0.02

Figure V.10. 1: Marque extraite avec α =0.02

$PSNR_1$= 32.5247dB

$PSNR_2$= 7.9222dB

V.4.2.A.2-Avec α =0.05





Figure V.9.2: Image compressée avec α =0.05

Figure V.10. 2: Marque extraite avec α =0.05

$PSNR_1$= 31.1218dB

$PSNR_2$= 13.6523dB

V.4.2.A.3-Avec α =0.1





Figure V.9.3: Image compressée

Figure V.10. 3: Marque extraite

avec α =0.1 **avec α =0.1**

$PSNR_1$= 29.7330dB **$PSNR_2$= 14.8881dB**

V.4.2.A.4-Avec α =0.2





Figure V.9.4: Image compressée **Figure V.10. 4: Marque extraite**

avec α =0.2 **avec α =0.2**

$PSNR_1$= 24.9102dB **$PSNR_2$= 18.5778dB**

V.4.2.A.5-Avec α =0.3





Figure V.9.5: Image compressée avec α =0.3

PSNR$_1$= 19.8567dB

Figure V.10. 5: Marque extraite avec α =0.3

PSNR$_2$= 18.8567dB

V.4.2.A.6-Avec α =0.4





Figure V.9.6: Image compressée avec α =0.4

PSNR$_1$= 16.3179dB

Figure V.10. 6: Marque extraite avec α =0.4

PSNR$_2$= 15.2211dB

V.4.2.A.7-Avec α =0.5





Figure V.9.7: Image compressée

Figure V.10. 7: Marque extraite

avec α =0.5 **avec α =0.5**

$PSNR_1$= 15.0018dB **$PSNR_2$= 13.3218dB**

V.4.2.B- Attaque par rotation de 15º

V.4.2.B.1-Avec α =0.02





Figure V.11.1: Image après rotation **Figure V.12. 1: Marque extraite**

Avec α =0.02 **avec α=0.02**

$PSNR_1$= 9.6211dB **$PSNR_2$= 3.5471dB**

V.4.2.B.2-Avec α =0.05





Figure V.11.2: Image après rotation **Figure V.12. 2: Marque extraite**

Avec α =0.05 **avec α=0.05**

$PSNR_1$= 9.6198dB **$PSNR_2$= 3.3325dB**

V.4.2.B.3-Avec α =0.1





Figure V.11.3: Image après rotation **Figure V.12. 3: Marque extraite**

Avec α =0.1 **avec α=0.1**

$PSNR_1$= 9.5862dB **$PSNR_2$= 3.0018**

V.4.2.B.4-Avec α =0.2





Figure V.11.4: Image après rotation **Figure V.12. 4: Marque extraite**

Avec α =0.2 **avec α=0.2**

$PSNR_1$= 9.6724dB **$PSNR_2$= 2.5847dB**

V.4.2.B.5-Avec α =0.3





Figure V.11.5: Image après rotation **Figure V.12. 5: Marque extraite**

Avec α =0.3 **avec α=0.3**

$PSNR_1$= 9.4259dB **$PSNR_2$= 2.4695dB**

V.4.2.B.6-Avec α =0.4





Figure V.11.6: Image après rotation **Figure V.12. 6: Marque extraite**

Avec α =0.4 **avec α=0.4**

$PSNR_1$= 9.2854dB **$PSNR_2$= 2.3145dB**

V.4.2.B.7-Avec α =0.5





Figure V.11.7: Image après rotation **Figure V.12. 7: Marque extraite**

Avec α =0.5 **avec α=0.5**

$PSNR_1$= 8.8547dB **$PSNR_2$= 2.0369dB**

V.4.2.C-Attaque par redimensionnement (resizing) de 1.25

V.4.2.C.1-Avec α =0.02





Figure V.13.1 : Image après resizing avec α=0.02

Figure V.14.1 :Marque extraite avec α=0.02

$PSNR_1$ = 12.1845dB

$PSNR_2$ = 3.1258 dB

V.4.2.C.2-Avec α =0.05





Figure V.13.2 : Image après resizing extraite

avec α=0.05

$PSNR_1$ = 12.1964dB

Figure V.14.2 :Marque

avec α=0.05

$PSNR_2$ = 3.2157 dB

V.4.2.C.3-Avec α =0.1





Figure V.13.3 : Image après resizing extraite

avec α=0.1

$PSNR_1$ = 12.2261dB

Figure V.14.3 :Marque

avec α=0.1

$PSNR_2$ = 2.5237 dB

V.4.2.C.4-Avec α =0.2





Figure V.13.4 : Image après resizing avec α=0.2

Figure V.14.4 : Marque extraite avec α=0.2

$PSNR_1$ = 12.1285dB

$PSNR_2$ = 2.4156 dB

V.4.2.C.5-Avec α =0.3





Figure V.13.5 : Image après resizing avec α=0.3

Figure V.14.5:Marque extraite avec α=0.3

$PSNR_1$ = 11.9915dB

$PSNR_2$ = 2.3256 dB

V.4.2.C.6-Avec α =0.4





Figure V.13.6 : Image après resizing avec α=0.4

Figure V.14.6:Marque extraite avec α=0.4

$PSNR_1$ = 11.6691dB

$PSNR_2$ = 1.9693 dB

V.4.2.C.7-Avec α =0.5





Figure V.13.7 : Image après resizing avec α=0.5

Figure V.14.7 :Marque extraite avec α=0.5

$PSNR_1$ = 11.2391dB

$PSNR_2$ = 1.5206 dB

V.4.2.D-Attaque par filtrage passe-bas(Gaussien, [3 3], sigma=0.5)

V.4.2.D.1-Avec α =0.02





Figure V.15.1 : Image après filtrage extraite

avec α=0.02

$PSNR_1$ = 36.8635dB

Figure V.15.1 :Marque

avec α=0.02

$PSNR_2$ =18.7193 dB

V.4.2.D.2-Avec α =0.05





Figure V.15.2 : Image après filtrage extraite

avec α=0.05

$PSNR_1$ = 37.6645dB

Figure V.16.2 :Marque

avec α=0.05

$PSNR_2$ = 12.3695 dB

V.4.2.D.3-Avec α =0.1





Figure V.15.3 : Image après filtrage avec α=0.1

Figure V.16.3 :Marque extraite avec α=0.1

$PSNR_1$ = 33.2153dB

$PSNR_2$ = 21.3521 dB

V.4.2.D.4-Avec α =0.2





Figure V.15.4 : Image après filtrage extraite

avec α=0.2

$PSNR_1$ = 27.2548dB

Figure V.16.4 :Marque

avec α=0.2

$PSNR_2$ = 21.2654 dB

V.4.2.D.5-Avec α =0.3





Figure V.15.5 : Image après filtrage extraite

avec α=0.3

$PSNR_1$ = 25.3621dB

Figure V.16.5 :Marque

avec α=0.3

$PSNR_2$ = 15.2147 dB

V.4.2.D.6-Avec α =0.4





Figure V.15.6 : Image après filtrage avec α=0.4

Figure V.16.6 :Marque extraite avec α=0.4

$PSNR_1$ = 22.3654dB

$PSNR_2$ = 16.6547 dB

V.4.2.D.7-Avec α =0.5





Figure V.15.7 : Image après filtrage avec α=0.5

Figure V.16.7 :Marque extraite avec α=0.5

$PSNR_1$ = 20.1869dB

$PSNR_2$ = 13.2589 dB

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	**32.5247**	**31.1218**	**29.7330**	**24.9102**	**19.8567**	**16.3179**	**15.001**
Redimensionnement	**12.1845**	**12.1964**	**12.2261**	**12.1285**	**11.9915**	**11.6691**	**11.239**
Filtrage passe-bas	**36.8635**	**37.6645**	**33.2153**	**27.2548**	**25.3621**	**22.3654**	**20.186**
Rotation	**9.6211**	**9.6198**	**9.5862**	**9.6724**	**9.4259**	**9.2854**	**8.8547**

Tableau V.3 : Tableau récapitulatif du $PSNR_1$ en dB vis à vis les différentes attaques (méthode combinée).

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
Compression JPEG	**7.9222**	**13.6523**	**14.8781**	**18.5778**	**18.8567**	**15.2211**	**13.321**
Redimensionnement	**3.1258**	**3.2157**	**2.5237**	**2.4156**	**2.3256**	**1.9693**	**1.5206**
Filtrage passe-bas	**18.7193**	**12.3695**	**21.3521**	**21.2654**	**15.2147**	**16.6547**	**13.258**
Rotation	**3.5471**	**3.3325**	**3.0018**	**2.5847**	**2.4695**	**2.3145**	**2.0369**

Tableau V.4 : Tableau récapitulatif du $PSNR_2$ en dB vis à vis les différentes attaques (méthode combinée).

V.4.3-Commentaires

1. En se référant au tableau V.2, et en comparaison avec les résultats affichés sur le tableau V.1 , nous remarquons que les valeurs du $PSNR_1$ et $PSNR_2$ sont très proches pour les trois méthodes (DCT,DWT, et méthode combinée quelque soit la force de tatouage α. Nous remarquons aussi que la marque est extraite d'une façon très claire pour les différentes valeurs de α, elle est presque la même que la marque originale, ce qui est vérifié par les valeurs du $PSNR_2$ qui varie entre [77.1236-15.6958], cela d'une part . D'autre part, nous remarquons que l'image tatouée n'a subi aucune dégradation avec un $PSNR_1$ variant entre [34.8956-16.9266] ; ce qui qualifie ce

tatouage de robuste en moyenne, et répond aux attentes que nous avons prévu.

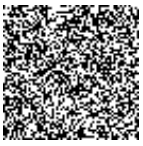
2. Nous observons aussi que la marque sur l'image tatouée commence à apparaître tard (pour α =0.3) ; et elle est invisible pour les valeurs de α< 0.3, ce qui nous a permis de réaliser une nette amélioration et qualifie ce type de tatouage d'invisible.

3. En se référant au tableau V.2 et V.3, nous remarquons que ce type de tatouage avec la méthode combinée est fragile devant les attaques géométriques (rotation et resizing), suite aux dégradations remarquées sur la marque extraite et selon les valeurs du $PSNR_1$ et $PSNR_2$ (trop faibles).En ce qui concerne l'attaque par compression JPEG, et filtrage passe-bas, ce tatouage par méthode combinée est moyennement robuste en se basant sur l'état de la marque extraite, et les valeurs du $PSNR_1$ et $PSNR_2$(des valeurs modérées).

V.4.4 - Test de la méthode combinée sans attaques (Image de Hamza)



Figure V.17 : Image originale de Hamza.



N(0,1)

Figure V.18 : Marque à Insérer.



(1) **(2)** **(3)** **(4)**



(5) **(6)** **(7)**

Figure V.19 : Image tatouée de Hamza.



(1) (2) (3) (4) (5) (6) (7)

Figure V.20 : Marque extraite sans attaques.

(1)Avec α=0.02.

(2)Avec α=0.05.

(3) Avec α=0.1.

(4) Avec α=0.2.

(5) Avec α=0.3.

(6) Avec α=0.4.

(7) Avec α=0.5.

α	0,02	0,05	0,1	0,2	0,3	0,4	0,5
$PSNR_1$ en dB	56.1248	50.2364	43.2658	38.2517	35.2698	32.5214	30.2178
$PSNR_2$ en dB	37.1025	35.2587	28.6598	23.1478	23.9517	19.2358	17.2589

Tableau V.5 : Tableau récapitulatif du $PSNR_1$ et $PSNR_2$ en fonction de α (Image de Hamza).

V.4.5-Comportement du tatouage combinée vis-à-vis des attaques

V.4.5.A-Attaque par compression JPEG [27]



(1) **(2)** **(3)** **(4)**



(5) **(6)** **(7)**

FigureV.21 : Image tatouée compressée de Hamza.



(1) (2) (3) (4) (5) (6) (7)

Figure V.22 : Marque extraite après compression.

(1)Avec α=0.02.

(2)Avec α=0.05.

(3) Avec α=0.1.

(4) Avec α=0.2.

(5) Avec α=0.3.

(6) Avec α=0.4.

7) Avec α=0.5.

V.4.5.B-Attaque par rotation de 15º



(1) (2) (3)



(4) (5) (6)



(7)

Figure V.23 : Image tatouée après rotation de Hamza.



FigureV.24 : Marque extraite après rotation.

(1)Avec α=0.02.

(2)Avec α=0.05.

(3) Avec α=0.1.

(4) Avec α=0.2.

(5) Avec α=0.3.

(6) Avec α=0.4.

(7) Avec α=0.5.

V.4.5.C-Attaque par redimensionnement (resizing) de 1.25



(1)



(2)



(3)



(4)



(5)

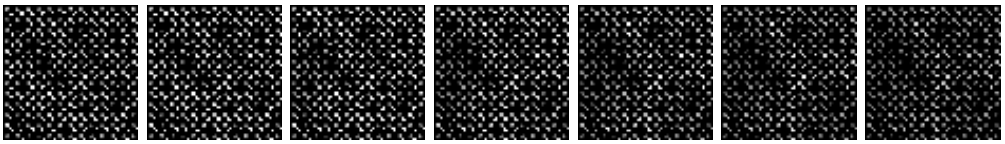


(6)



(7)

Figure V.25 : Image tatouée après redimensionnement de Hamza.



(1) (2) (3) (4) (5) (6) (7)

Figure V.26 : Marque extraite après redimensionnement.

(1)Avec α=0.02.

(2)Avec α=0.05.

(3) Avec α=0.1.

(4) Avec α=0.2.

(5) Avec α=0.3.

(6) Avec α=0.4.

(7) Avec α=0.5.

V.4.5.D-Attaque par filtrage passe-bas (Gaussien, [3 3], sigma=0.5)



(1) (2) (3)



(4) (5) (6)



(7)

Figure V.27 : Image tatouée après filtrage passe-bas de Hamza.



(1) (2) (3) (4) (5) (6) (7)

Figure V.28 : Marque extraite après filtrage passe-bas.

(1)Avec α=0.02.

(2)Avec α=0.05.

(3) Avec α=0.1.

(4) Avec α=0.2.

(5) Avec α=0.3.

(6) Avec α=0.4.

(7) Avec α=0.5.

α	**0,02**	**0,05**	**0,1**	**0,2**	**0,3**	**0,4**	**0,5**
Compression JPEG	**36.6583**	**36.6570**	**36.2956**	**35.3160**	**33.8198**	**32.5078**	**31.2893**
Redimensionnement	**54.6274**	**46.3496**	**40.6228**	**35.0690**	**32.0198**	**29.9584**	**28.3782**
Filtrage passe-bas	**46.2456**	**45.5878**	**43.8796**	**40.1824**	**36.9838**	**34.4354**	**32.2717**
Rotation	**7.8481**	**7.8476**	**7.8462**	**7.8417**	**7.8347**	**7.8258**	**7.8148**

Tableau V.5 : Tableau récapitulatif du $PSNR_1$ en dB vis à vis des différentes attaques.

α	**0,02**	**0,05**	**0,1**	**0,2**	**0,3**	**0,4**	**0,5**
Compression JPEG	**13.3604**	**13.1458**	**14.0997**	**14.6352**	**14.9729**	**15.0350**	**15.0023**
Redimensionnement	**3.2415**	**3.2245**	**3.2350**	**3.2728**	**3.3110**	**3.3384**	**3.3566**
Filtrage passe-bas	**14.2434**	**14.4023**	**14.2488**	**13.9490**	**13.6383**	**13.3871**	**13.2050**
Rotation	**3.5126**	**3.2680**	**3.1518**	**3.0675**	**3.0367**	**3.0199**	**3.0122**

Tableau V.6 : Tableau récapitulatif du $PSNR_2$ en dB vis à vis des différentes attaques.

V.8-Conclusion

Dans ce chapitre, nous avons présenté une étude de performance de la méthode de tatouage des images fixes par une méthode combinée (DCT, DWT). La technique a été testée sur un certain nombre d'images comme Lena et Hamza. La robustesse du marquage a été validée à l'aide des attaques de compression JPEG, redimensionnement, rotation et filtrage passe-bas. Les résultats obtenus montrent que le tatouage avec la méthode combinée a nettement amélioré l'invisibilité de la marque (elle devienne visible avec α=0.3 au lieu de 0.1 dans le cas de la DCT) et en même temps, il a gardé les mêmes performances trouvées par les transformées (DCT, DWT).



Dans ce manuscrit, nous avons présenté trois techniques de tatouage numérique des images fixes dans le domaine fréquentiel: une technique utilisant la transformée en cosinus discrète DCT, une autre basée sur la transformée en ondelettes de Haar, et une méthode combinant DCT et DWT. Ces techniques ont été testées sur deux images fixes Lena, et mon fils Hamza. La marque a été choisie selon deux modèles : une image qui

porte mon nom, prénom, et ma willaya, et une marque générée par un générateur pseudo aléatoire. Les tests sont caractérisés par la simplicité que ce soit dans la phase d'incrustation ou dans la phase d'extraction par utilisation de simples programmes développés sous environnement Matlab7.1.

Les performances des techniques de tatouage citées sont validées à l'aide des attaques de compression JPEG, redimensionnement, rotation et filtrage passe bas. Les résultats obtenus ont montré que le tatouage par DCT est robuste lors du test effectué sans attaques, tandis qu'il est fragile vis-à-vis des attaques géométriques. Enfin, vis-à-vis des attaques de compression et de filtrage passe bas, il est moyennement robuste. En ce qui concerne la visibilité, nous avons constaté l'apparition de la marque à partir de la valeur de la force de tatouage (α=0,1), ce problème est dû probablement à l'endroit d'incrustation.

Les résultats obtenus lors des tests effectuées par la transformée (DWT) de Haar ont montré aussi que ce type de tatouage est robuste dans le cas du test sans attaques, mais moins robuste par rapport a la transformée (DCT). Les résultats de robustesse obtenus de cette méthode vis-à-vis les attaques déjà citées sont très proches à ceux trouvés avec la transformé (DCT). A propos de la visibilité, elle a connu une nette amélioration par rapport au cas de la (DCT). Vu le compromis robustesse – invisibilité rencontré par les transformées (DCT) et (DWT), la méthode combinée a donné de bons résultats, il y a une amélioration dans l'invisibilité (l'apparition de la marque à partir de la valeur de la force de tatouage (α=0,3)), tout en gardant les mêmes performances de robustesse.

Lors des tests effectués, nous avons rencontré quelques contraintes, nous citons par exemple : problème de visibilité de la marque rencontré avec la transformée en DCT, temps de calcul, un peu élevé, observé dans la méthode combinée. Pour cela, nous suggérons en perspectives l'amélioration de l'invisibilité par incrustation de la marque dans des endroits qui n'altèrent pas la qualité des images tatouées, et qui

assurent la bonne extraction de la marque. Pour remédier au problème du temps d'exécution élevé, nous proposons soit le développement d'algorithmes rapides de tatouage, soit l'implémentation des algorithmes de tatouage sur les processeurs DSP, ou bien le développement d'architectures parallèles ou systoliques reliées à ces algorithmes et réalisables sur FPGA.