

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur Et de La Recherche Scientifique



Université de Ghardaïa

N° d'ordre :
N° de série :

Faculté des Sciences et Technologies
Département d'automatique et électromécanique

Mémoire présenté en vue de l'obtention du diplôme de

MASTER

Domaine : *Sciences et Technologies*

Filière : Automatique

Spécialité : Automatique et Systèmes

Par :

- ELBAR Mammar
- LAOUAR Mohamed

Thème

**Compression d'Image par Fusion
Technique**

Soutenu publiquement le 26/09/2020

Devant le jury :

DJALLOULI Taher	MAA	Univ. Ghardaïa	Président
ARIF Mohamed	MAA	Univ. Ghardaïa	Examineur
ALLALI Mohamed	MAA	Univ. Ghardaïa	Examineur
LAADJEL Boumediene	MAA	Univ. Ghardaïa	Encadreur

Année universitaire 2019/2020

DEDICAS

À toi ma mère

*Toi qui m'a donnée la tendresse, le courage et La volonté pour
arriver à mon but*

À toi mon père

*Qui a veillé tout au long de ma vie
à m'encourager, à me donner l'aide et à me protéger.
Que dieu les gardes et les protège.*

À ma femme

*Les mots ne suffisent guère pour exprimer l'attachement,
l'amour et l'affection que je porte pour toi.*

*À mes adorables enfants : AYA, ABDERAHMANE,
TAREKZIAD et la petite fleur NIHAL*

À mes cousine « IMANE, BRAHIME ET ELHACHEMI »

*Je ne peux trouver les mots justes et sincères pour vous
exprimer mon affection et mes pensées, vous êtes pour moi des
frères, et des amis sur qui je peux compter.*

Enfin, A tous ceux qui me sont chers et que je n'ai pas cité.

ELBAR Mammar

DEDICAS

Je dédie ce travail

*A ma mère qui m'a soutenu et encouragé durant ces années
d'études.*

*Qu'elle trouve ici le témoignage de ma profonde
reconnaissance.*

*A mes frères, et Ceux qui ont partagé avec moi tous les
moments d'émotion lors de la réalisation de ce travail. Ils
m'ont chaleureusement supporté et encouragé tout au long de
mon parcours.*

*A ma famille, mes proches et à ceux qui me donnent de
l'amour et de la vivacité.*

*A tous mes amis qui m'ont toujours encouragé, et à qui je
souhaite plus de succès.*

*ELBAR, HADOU, HHABET, LHADJ-MOHAMED FAICEL,
et YOUNCE DJKAWA*

À tous ceux que j'aime

I.

LAOUAR Mohamed

REMERCIEMENTS

*En premier lieu, nous tenons à remercier notre DIEU,
notre
Créateur pour nous avoir la force pour accomplir ce
travail.*

*Nous remercions également tous les enseignants du
département d'Automatique et Electromécanique qui ont
contribué, de près ou de loin, à notre étude et pour tous
les enseignements de l'université de Ghardaïa.*

*En particulier notre promoteur M. LAADJEL
Boumediene, pour ces conseils et l'aide qu'il nous a
apporté.*

Respectivement le chef de département.

*Je remercie aussi, tous les membres de jury, pour
l'intérêt qu'ils ont porté à ce travail. Nous leur
exprimons toute mes reconnaissances.*

*Que Toute personne ayant participé de près ou de loin à la
réalisation de ce travail accepte nos grands et sincères*

Remerciements.

Merci!



Table des matières

Résumé	
Abstract	
Introduction générale.....	1
CHAPITRE I : Les Bases du Traitement d'Images	
I.1. INTRODUCTION.....	2
I.2. DEFINITIONS.....	2
I.2.1 Image et pixel.....	2
I.2.2 Dimension	3
I.2.3 Résolution d'une image.....	3
I.2.4 Quelques exemples de résolutions fréquemment utilisées	4
I.2.5 Stocker une image	4
I.3. TYPES D'IMAGES.....	5
I.3.1 Image binaire (noir et blanc).....	5
I.3.2 Image en niveaux de gris	5
I.3.3 Image couleur (RGB).....	6
I.4. FORMATS D'IMAGES	7
I.5. HISTOGRAMME D'UNE IMAGE	8
I.5.1 Définition	8
I.5.2 Dynamique d'une image.....	9
I.5.3 Histogrammes d'une image couleur	9
I.6. TRAITEMENTS D'HISTOGRAMME	10
I.6.1 Effet de la rotation d'une image	10
I.6.2 Sensibilité de l'image aux modifications d'histogrammes	10
I.6.3 Égalisation d'histogramme	11
I.6.4 Inversion d'image	12
I.7. NOTION DES PROCHES VOISINS ET CONTOUR	12
I.7.1 Voisinage d'un pixel	12
I.7.2 Adjacence	13
I.7.3 Chemin	13
I.7.4 Distance	13
I.7.5 Connectivité.....	13
I.8. REGIONS ET FRONTIERES	14
I.8.1 Régions.....	14
I.8.2 Frontières	14
I.8.3 Chaîne de traitement d'images	15
I.9. CONCLUSION	15

CHAPITRE II : Généralités sur les Techniques de la Compression d'Images

II.1.	INTRODUCTION	18
II.2.	DEFINITION.....	18
II.3.	PRINCIPE GENERAL DE LA COMPRESSION DES IMAGES.....	20
II.4.	CLASSIFICATION DES METHODES DE COMPRESSION.....	21
II.4.1	Méthodes de compression sans perte d'informations	21
II.4.2	La méthode RLE.....	21
II.4.3	La méthode de HUFFMAN.....	22
II.4.4	La méthode entropique de Shannon-Fano	24
II.4.5	Codage arithmétique.....	26
II.4.6	Méthodes de Compression avec perte d'informations	29
II.5.	CRITERES DE MESURES DES PERFORMANCES DE COMPRESSION ...	31
II.5.1	Taux de compression	31
II.5.2	Gain de compression	31
II.5.3	Mesures de distorsion	31
II.5.4	Rapport signal à bruit en pic PSNR.....	31
II.5.5	Temps de calcul	32
II.6.	CONCLUSION	32

CHAPITRE III: Compression par Fusion Technique

III.1.	INTRODUCTION	33
III.2.	LA TRANSFORMEE EN ONDELETTES.....	33
III.2.1	Choix d'ondelettes	34
III.2.2	Algorithme transformée des ondelettes à deux dimensions	34
III.3.	LA TRANSFORMEE DE WALSH.....	35
III.3.1	La fonction de Walsh	35
III.3.2	Matrice de Hadamard.....	36
III.3.3	Les transformées de Walsh directe et inverse	36
III.3.4	L'algorithme de WWT (Wavelet Walsh Transform).....	37
III.4.	PROCESSUS DE COMPRESSION ET DECOMPRESSION D'IMAGES.....	39
III.5.	TESTS ET RESULTATS	40
III.5.1	Images de test.....	40
III.5.2	Paramètres d'évaluation :.....	40
III.5.3	Logiciel de l'implémentation	40
III.5.4	Interprétations des tests	45
III.6.	CONCLUSION	45
	Conclusion générale.....	45
	Perspectives	
	Bibliographie	
	

Table des figures

<i>figure I-1 representation des notions image et pixel</i>	3
<i>figure I-2 exemples de resolution d'une image.</i>	4
<i>figure I-3 exemple d'une image binaire</i>	5
<i>figure I-4 exemple d'une image en niveaux de gris.</i>	5
<i>figure I-5 exemple d'une image couleur reelle</i>	6
<i>figure I-6 decomposition d'une image couleur en ses trois composantes.</i>	6
<i>figure I-7 exemples d'histogrammes</i>	8
<i>figure I-8 dynamique d'une image</i>	9
<i>figure I-9 histogramme de l'image couleur traitee</i>	9
<i>figure I-10 histogramme d'une image avant et apres rotation</i>	10
<i>figure I-11 exemple d'un etirement d'histogramme</i>	11
<i>figure I-12 la transformation T</i>	11
<i>figure I-13 exemple d'une inversion d'image et son histogramme</i>	12
<i>figure I-14 representation du voisinage d'un pixel</i>	12
<i>figure I-15 representation les regions et frontieres</i>	14
<i>figure I-16 schema d'un systeme de traitement d'images</i>	15
<i>figure II-1 architecture generale de la compression d'une image</i>	19
<i>figure II-2 schema de compression /decompression classique d'image</i>	20
<i>figure II-3 compression RLE</i>	21
<i>figure II-4 les etapes de la methode HUFFMAN</i>	23
<i>figure II-5 subdivisions de la partition initiale</i>	28
<i>figure II-6 principe general des algorithmes de compression/decompression d'image</i>	29
<i>figure II-7 principe de compression DCT avec perte</i>	30
<i>figure II-8 principe de compression DWT avec perte</i>	30
<i>figure III-1 analyse multi-resolutions en deux dimensions</i>	34
<i>figure III-2 schema de decomposition en ondelette niveau 1</i>	35
<i>figure III-3 architecture de compression hybride</i>	38
<i>figure III-4 distribution balisage Zig-Zag</i>	38
<i>figure III-5 schema general des etapes de la compression et de la decompression</i>	39
<i>figure III-6 represente la reconstruction de l'image medicale (main)</i>	40
<i>figure III-7 represente la reconstruction de l'image biomedicale (œil)</i>	41
<i>figure III-8 represente la reconstruction de l'image photographie (vue de ghardaïa)</i>	42
<i>figure III-9 represente la reconstruction d'une image de reconnaissance des objets (historique)</i>	43
<i>figure iii-10 represente la reconstruction d'une image photographie de profondeur marine (fond marin)</i>	44

Liste des tableaux

TABLE II-1 La phase d'encodage	23
TABLE II-2 Codage de l'arbre.....	24
TABLE II-3 Frequences des symboles	25
TABLE II-4 Probabilites des symboles	25
TABLE II-5 Exemple de codage de Shannon-Fano	26
TABLE III-1 Represente les parametres d'evaluation du test 1	41
TABLE III-2 Represente les parametres d'evaluation du test 2.....	41
TABLE III-3 Represente les parametres d'evaluation du test 3.....	42
TABLE III-4 Represente les parametres d'evaluation du test 4.....	43
TABLE III-5 Represente les parametres d'evaluation du test 5.....	44

Abréviation

WWT	Wavelet Walsh Transform
DCT	Discret Cosine Transform
DWT	Discret Wavelet Transform
DFT	Discret Fourier Transform
Tc	Temps de calcul
Bpp	Bit par pixel
MSE	Mean Square Error
JPEG	Joint Photographic Experts Group
PPP	points par pouce
DPI	dots per inch
Rc	ratio of compression
EZW	Embedded Zerotree Wavelet
RLE	Run Length Encoding
SNR	Signal to Noise Ratio
PSNR	Peak Signal to Noise Ratio
SSIM	Structural Similarity Index
LZ W	Lempel-Ziv-Welch
BMP	Bitmap
GIF	Graphic Interchange Format
TIF	Tagged Image FileFormat
TSV	Teinte, saturation, valeur
RGB	Red , Green, bleu
Aret-Code	Arithmétique codage



ملخص: عمليات ضغط الصور تعمل على تعديل عدد البايتات (bites) الأساسية للدلالة على الصورة، والتي يمكن أن

تحسن من أداء الأنظمة أثناء التخزين والنقل دون المساس بجودة الصورة.

وفقاً لذلك، تم اقتراح في هذا البحث تقنية هجينة جديدة لضغط الصورة تعتمد أساساً على تقنيتان يعملان على مبدأ

التحويل:

تم الجمع بين تقنيه التحويل الموجي المنفصل (DWT) وتحويل Walsh Hadamard (WWT) لضغط الصور

لإضفاء الخصائص الجيدة لهذه التقنية الجديدة. تعمل الطريقة المقترحة على ضغط المصادر وتظهر النتائج المحققة

أنها تعمل بشكل جيد مقارنة بالطرق الحالية. ولاختبار مستوى الضغط، يتم استخدام القياسات الكمية لنسبة ذروة الإشارة

إلى الضوضاء (PSNR) ومتوسط الخطأ التربيعي (MSE) لضمان فعالية النظام المقترح.

الكلمات المفتاحية: تحويل الموجية. تحويل والش هادا ماردا؛ دمج البيانات؛ ضغط الصورة.

Résumé : Les processus de compression d'image modèrent le nombre de bits essentiels pour signifier une image, ce qui peut améliorer les performances des systèmes pendant le stockage et la transmission sans compromettre la qualité de l'image. En conséquence, une nouvelle technique de compression d'image hybride a été suggérée dans cette recherche. deux techniques basées sur la transformation:

la transformée en ondelettes discrète (DWT) et La transformée de Walsh Hadamard (WWT), ont été combinées pour la compression d'image, afin de conférer les bonnes caractéristiques de ces méthodes.

La méthode proposée fonctionne pour la compression de déchéance et les résultats obtenus montrent qu'elle fonctionne bien par rapport aux approches existantes. Pour tester le niveau de compression, les mesures quantitatives du rapport signal / bruit de crête (PSNR) et de l'erreur quadratique moyenne (MSE) sont utilisées pour garantir l'efficacité du système suggéré.

Mots clés: transformée en ondelettes; La transformée de Walsh Hadamard; La fusion des données; Compression d'image.

Abstracts: Image compression processes moderate the number of bits essential to signify an image, which can improve the performance of systems during storage and transmission without compromising image quality. Accordingly, a new hybrid image compression technique has been suggested in this research. two transform based discrete wavelet transform (DWT), and The Walsh Hadamard Transform (WWT) have been combined for image compression to confer the good characteristics of these methods. The proposed method works for forfeiture compression and the attained results show that it works well compared to existing approaches. To test the level of compression, the quantitative measures of the peak signal-to-noise ratio (PSNR) & mean squared error (MSE) are used to ensure the effectiveness of the suggested system.

Keywords: wavelet transform; Walsh Hadamard Transform; Data fusion; Image compression.



**INTRODUCTION
GENERALE**

Introduction générale

Les évolutions récentes des méthodes de l'information et systèmes de communication, ainsi que la diversification des applications fonctionnalités du multimédia dans plusieurs domaines (télémédecine, astronomie, sécurité, télédétection, la vision artificielle, la reconnaissance... etc.) impliquent la manipulation de données très volumineuses. Il est donc indispensable de disposer d'outils performants pour la transmission et le stockage d'énormes quantités des informations. Pour remédier à ces contraintes, les chercheurs ont développé au cours des dernières décennies de nombreuses méthodes de compression de données déduites de la théorie de l'information et faisant appel à de nombreux domaines des mathématiques et de l'informatique.

Cette opération devient plus que nécessaire vu le volume important d'information mis en œuvre lors de l'utilisation des images et vidéos numériques, qui ont de ce fait un besoin particulièrement important d'un traitement adapté à leurs spécificités. Les méthodes de compression de ce type de données peuvent être classées suivant la nécessité de récupération parfaite ou non de l'information originale en deux grandes catégories :


La compression sans pertes d'information ou réversible qui a l'avantage de préserver la qualité de l'image originale, mais avec un taux de compression relativement faible et la compression avec pertes qui regroupe des algorithmes caractérisés par leur taux de compression assez élevé tout en gardant le mieux possible une acceptable qualité de l'image originale. Elle est basée généralement sur une phase de transformation qui sert à compacter l'information utile dans un nombre minimum de coefficients non nuls. [13]

Le premier chapitre présente les concepts du traitement d'images fixes, est réservé à des généralités sur les différentes images et leurs définitions et de la qualité des images reconstruites afin de se familiariser avec les notions et termes dans ce mémoire.

Le deuxième chapitre est un aperçu global sur la compression. Son objectif principal est de cerner les différentes techniques de compression réversibles et irréversibles des images fixes existantes ainsi que les critères utilisés pour l'évaluation des performances d'une méthode, afin de pouvoir choisir la plus appropriée à notre schéma.

Dans le troisième chapitre nous avons proposés une compression hybride des images basées sur la transformée en ondelette couplée avec la transformée de Walsh pour atteindre des taux de compression élevés avec une qualité satisfaisante de l'image reconstruite ; et un temps de calculs minimal lors de processus de codage. Nous terminons notre étude par une série d'expérimentations pour l'algorithme proposé.

Nous terminons ce mémoire par une conclusion générale et les perspectives envisagées concernant notre travail.



CHAPITRE I

LES BASES DU TRAITEMENT

D'IMAGES

I. Les bases du traitement d'images

I.1. Introduction

Le traitement numérique des images est l'ensemble des méthodes qui cherche à rendre cette opération possible, et plus simple. Les débuts du traitement d'images remontent au 18^{ème} siècle avec l'apparition de l'appareil photo puis l'introduction de l'électronique au 19^{ème} siècle. Le traitement d'image trouve son application dans diverses disciplines, telles que l'électronique, l'informatique, la neurophysiologie, les statistiques, etc. [1,2].

Les images numériques sont présentées sous forme de matrices, cependant les opérations, en particulier arithmétique, se font entre pixels, des deux images, situés aux mêmes positions.

De ce fait, nous allons réserver ce chapitre à la présentation des notions générales liées à l'image et son traitement numérique.

I.2. Définitions

I.2.1 Image et pixel

L'image c'est la représentation d'un objet par la peinture, la sculpture, la photographie, etc... C'est aussi un ensemble structuré d'informations qui après affichage sur écran, ont une signification pour l'œil humain. Elle peut être écrite sous forme de fonction $I(x,y)$ ou I est une fonction d'intensité lumineuse ou de couleur aux coordonnées spatiales (x,y) . De cette façon l'image est exploitable par la machine, d'où la nécessité de sa numérisation.

- **L'image numérique** : Il est clair que les images manipulées par l'ordinateur sont numériques (série de bits). L'image numérique est l'image dont la surface est divisée en éléments de taille fixe appelés pixels, ayant comme caractéristique le niveau de gris ou de couleur. La numérisation d'une image est la conversion de celle-ci en une image numérique représentée par une matrice bidimensionnelle de valeurs numériques $f(x, y)$ qui sont les niveaux de gris des coordonnées réelles (x,y) .
- **Pixel** : C'est le plus petit point de l'image. Chaque pixel a une valeur numérique qui représente le niveau de gris ou de couleur selon la nature de l'image.

- **Voisinage d'un pixel :** Le voisinage d'un pixel est l'ensemble des pixels qui lui sont adjacents. Selon le codage de FREEMAN on distingue deux types de voisinages : 4-connexité et 8-connexité.

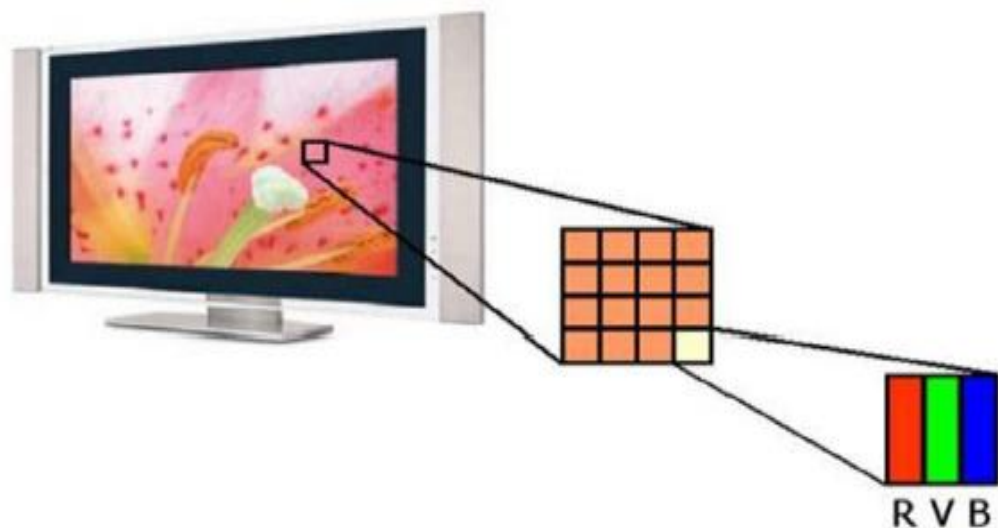
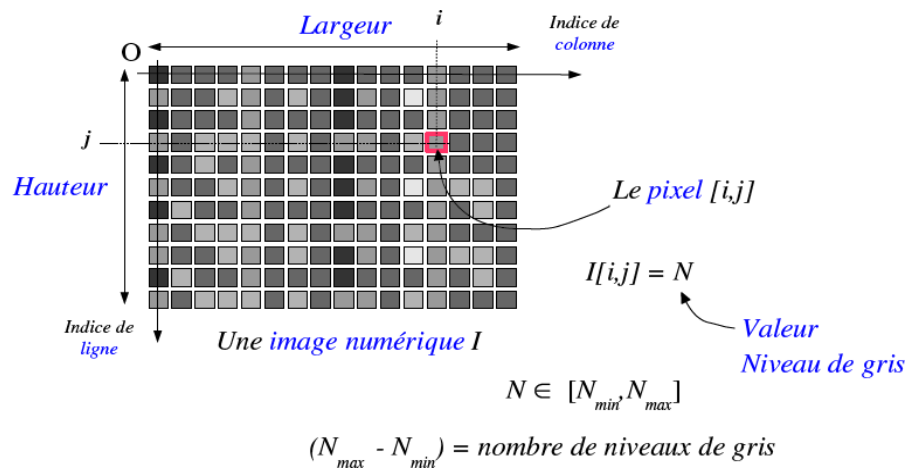


Figure I-1 Représentation des notions image et pixel

I.2.2 Dimension

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image.

I.2.3 Résolution d'une image

La résolution d'une image est le nombre de pixels par pouce qu'elle contient : (1 pouce = 2.54 centimètres). Elle est exprimée en "PPP" (points par pouce) ou DPI (dots per inch). Plus il y a de pixels (ou points) par pouce et plus il y aura d'information dans l'image (plus précise). Par exemple, une résolution de 300dpi signifie que l'image comporte 300

pixels dans sa largeur et 300 pixels dans sa hauteur, elle est donc composée de 90 000 pixels (300x300 ppp). La résolution permet ainsi d'établir le rapport entre la définition en pixels d'une image et la dimension réelle de sa représentation sur un support physique (affichage écran, impression papier...). [2]

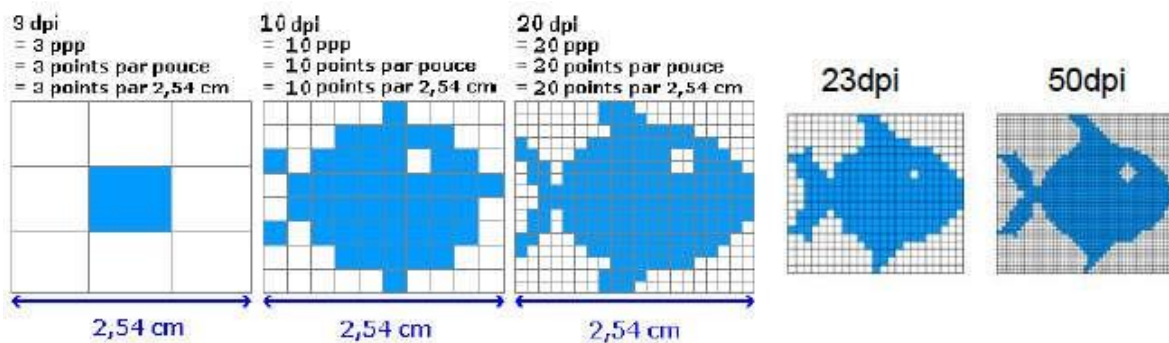


Figure I-2 Exemples de résolution d'une image.

I.2.4 Quelques exemples de résolutions fréquemment utilisées

- Ecrans d'ordinateur : 72 dpi. C'est aussi dans cette résolution que sont les images sur Internet.
- Fax : en générale en 200 dpi.
- Imprimantes grand public : entre 360 dpi et 1400 dpi. Cela permet d'obtenir une qualité tout à fait honorable pour tous les travaux courants (courriers, rapports, etc....)
- Scanners grand public : 300, 600 ou 1200 dpi.
- Matériel d'impression professionnel : aux minimum 4800 dpi (impression de qualité et grandes tailles pour les affiches).

I.2.5 Stocker une image

Stocker de grandes images sur le disque dur d'un ordinateur prend beaucoup de place. Les nombres entiers sont stockés en écriture binaire, c'est-à-dire sous la forme d'une succession de 0 et de 1. Chaque 0 et chaque 1 se stocke sur une unité élémentaire de stockage, appelée bit. Pour obtenir l'écriture binaire d'un pixel ayant comme valeur 179, il faut décomposer cette valeur comme somme de puissances de deux. On obtient ainsi : $179=2^7+2^5+2^4+2^1$,

L'écriture binaire de la valeur 179 du pixel est ainsi (1,0,1,1,0,0,1,1), où chaque 1 et chaque 0 correspond au facteur multiplicatif qui apparaît devant chaque puissance. On peut écrire toute valeur entre 0 et 255 de cette manière, ce qui nécessite d'utilisation de 8 bits. Il y a en effet 256 valeurs possible, et $2^8=256$. Pour stocker l'image complète, on a donc besoin de $n \cdot p \cdot 8$ bits.

Pour l'image montrée à un exemple de $n=240$ et $p=240$, on a ainsi besoin de :

$240 \times 240 \times 8 = 460800$ bits.

On utilise le plus souvent l'octet (8 bits) comme unité, de sorte que cette image nécessite 57.6ko (kilo octets).

I.3. Types d'images

Nous distinguons 3 types d'images :

I.3.1 Image binaire (noir et blanc)

Une image binaire est une image numérique qui a seulement deux valeurs possibles pour chaque pixel . En règle générale, les deux couleurs utilisées pour une image binaire sont en noir et blanc. [2]

1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	1	1	1
1	1	0	1	1	1	1	0	1	1
1	0	1	1	1	1	1	1	0	1
1	0	1	0	1	1	0	1	0	1
1	0	1	1	1	1	1	1	0	1
1	0	1	0	1	1	0	1	0	1
1	0	1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0	1	1
1	1	1	0	0	0	0	1	1	1
1	1	1	1	1	1	1	1	1	1

Figure I-3 Exemple d'une image binaire [12].

I.3.2 Image en niveaux de gris (256 niveaux de gris)

Chaque pixel est un niveau de gris, allant de 0 (noir) à 255 (blanc). Cet intervalle de valeurs signifie que chaque pixel est codé sur huit bits (un octet). 256 niveaux de gris sont généralement suffisants pour la reconnaissance de la plupart des objets d'une scène.



Figure I-4 Exemple d'une image en niveaux de gris [12].

I.3.3 Image couleur (ou RGB)

Chaque pixel possède une couleur décrite par la quantité de rouge (R), vert (G) et bleu (B). Chacune de ces trois composantes est codée sur l'intervalle [0, 255], ce qui donne $255^3 = 16\,777\,216$ couleurs possibles. Il faut donc 24 bits pour coder un pixel.

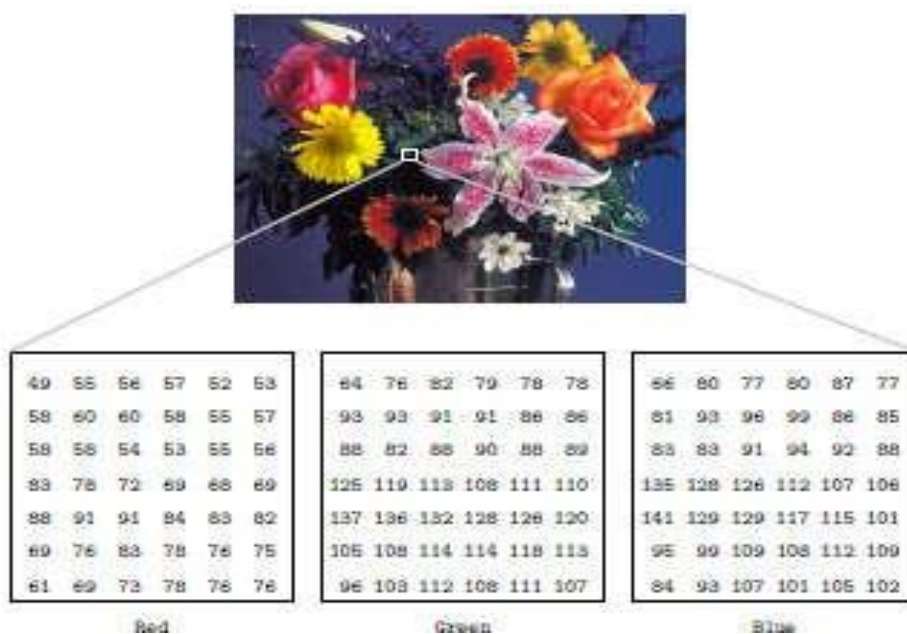


Figure I-5 Exemple d'une image couleur réelle

En utilisant l'image que nous avons pris et que nous allons exploiter par la suite, nous avons décomposé cette image en ses trois composantes : Rouge, Verte, Bleu. Le résultat obtenu est donné sur la figure I-6 ci-dessous.

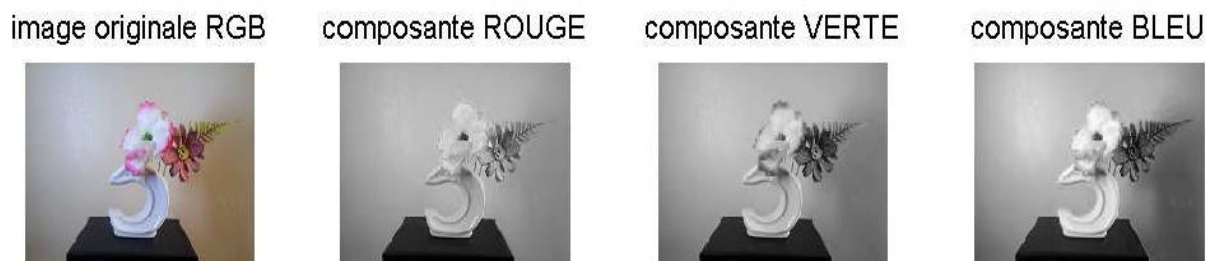


Figure I-7 Décomposition d'une image couleur en ses trois composantes (Rouge, verte, bleu).

I.4. Formats d'images

Les formats les plus utilisés sont :

- **BMP (bitmap)**

Le format BMP est un des premiers formats d'image utilisé sous Windows. Il est un des seuls formats à ne pas utiliser. Cette technologie a pour principal avantage la qualité des images fournies : pas de compression = pas de perte de qualité.

Revers de la médaille : ne pas compresser le fichier donne des fichiers de grande taille, impossible à afficher sur internet pour un utilisateur ayant une connexion bas débit.

Pour exemple, une image 800*600*3 pixels pèseront 1.37Mo. [3]

De plus, les fichiers n'étant pas compressés, deux images ayant la même définition auront toujours la même taille.

- **GIF (Graphic Interchange Format)**

Il s'agit d'un format de compression sans perte. Ce format assure une division environ par 5 de la taille du fichier initial.

- **JPEG (Joint Photo Expert Group)**

Ce format offre des taux de compression inégalés, même si la qualité de l'image s'en ressent au fur et à mesure que vous augmentez la compression. Avec des taux de compression élevés donnant lieu à des fichiers images de petite taille, ce format est devenu le standard des formats d'image sur internet.

- **TIF (Tagged Image FileFormat).**

Ce format est orienté vers les professionnels (imprimeurs, publicitaires...) car il a l'avantage d'être reconnu sur tous types de système d'exploitation : Windows, Mac, Linux, Unix ...

En plus de ces 4 formats, on cite :

- FPX (Flashpix), PCD (PhotoCD),
- PNG (Portable NetworkGraphic),
- PSD (PhotoShopDocument),
- PSP (Paint ShopPro).

I.5. Histogramme d'une image

I.5.1 Définition :

En imagerie numérique, l'histogramme représente la distribution des intensités (ou des couleurs) de l'image. C'est un outil fondamental du traitement d'images, avec de très nombreuses applications. Les histogrammes sont aussi très utilisés en photographie et pour la retouche d'images.

Considérons une image monochrome dans laquelle $f(i, j)$ représente la fonction intensité du pixel de coordonnées (i, j) . L'histogramme est la représentation graphique de la fréquence d'apparition $h(f)$ de chaque niveau f dans l'image.

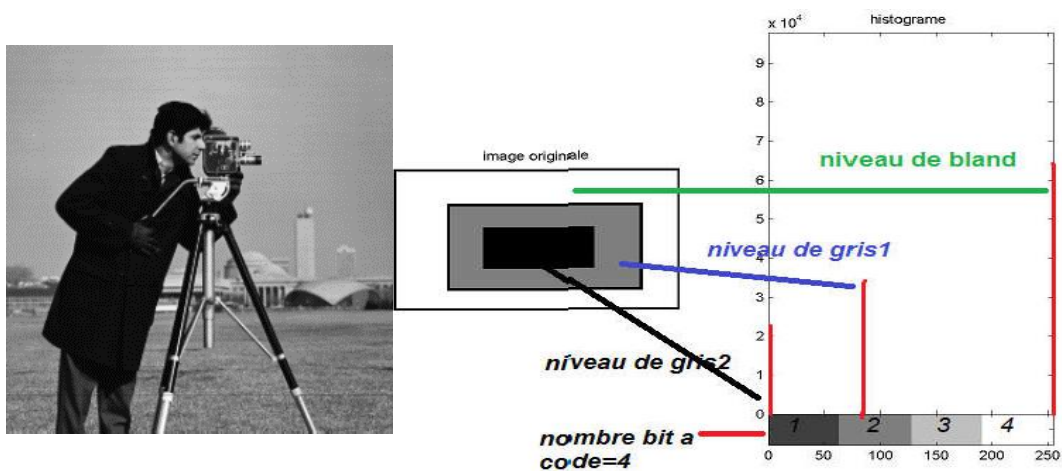


Figure I-8 Exemples d'histogrammes [1]

L'abscisse d'un histogramme représente les niveaux d'intensité allant du plus foncé à gauche au plus clair à droite.

I.5.2 Dynamique d'une image

On appelle dynamique : l'intervalle $[a, b]$ où a et b sont respectivement les niveaux de gris minimal et maximal présents dans l'image. La dynamique maximale est $[0, 255]$.

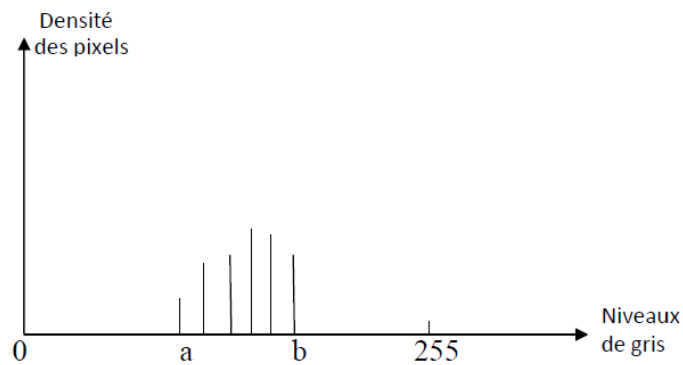


Figure I-9 *Dynamique d'une image*

I.5.3 Histogrammes d'une image couleur

L'histogramme d'une image couleur est constitué de :

- Distribution de la luminance Y ;
- Distributions respectives des composantes Rouge, Verte et Bleu.

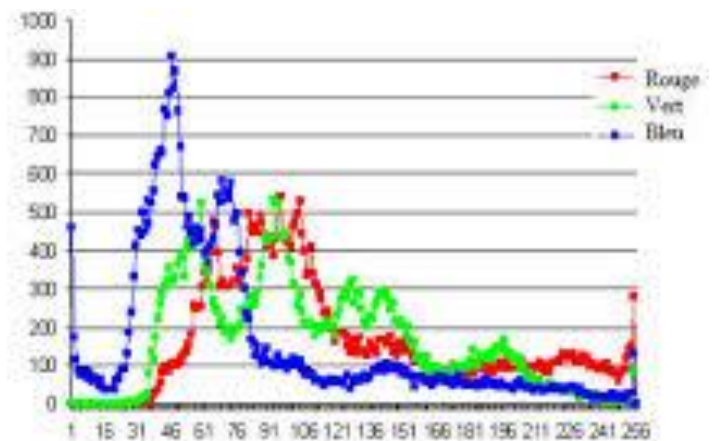


Figure I-10 *Histogramme de l'image couleur traitée*

I.6. Traitements d'histogramme

Le traitement d'histogramme permet d'améliorer la qualité d'une image en intervenant sur la distribution de ses pixels.

I.6.1 Effet de la rotation d'une image

L'image et sa rotation aboutissent au même histogramme.

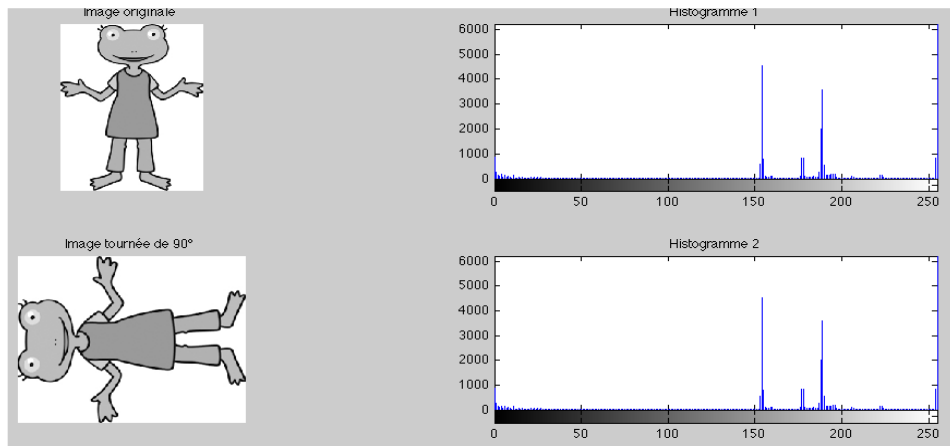


Figure I-11 Histogramme d'une image avant et après rotation

I.6.2 Sensibilité de l'image aux modifications d'histogrammes

Estournet a fait réaliser plusieurs copies d'une même photo sur des supports variés : papiers photographiques de marques, de gradations et de duretés différentes. L'analyse de ces documents montre clairement de grandes différences d'histogrammes, différences bien plus remarquables que celles qui affectent les images elles-mêmes. On en déduit que l'histogramme de l'image non seulement n'est pas significatif mais aussi peut être modifié artificiellement sans que l'aspect extérieur de l'image n'en soit beaucoup affecté, et surtout sans que la signification de l'image ne soit altérée.

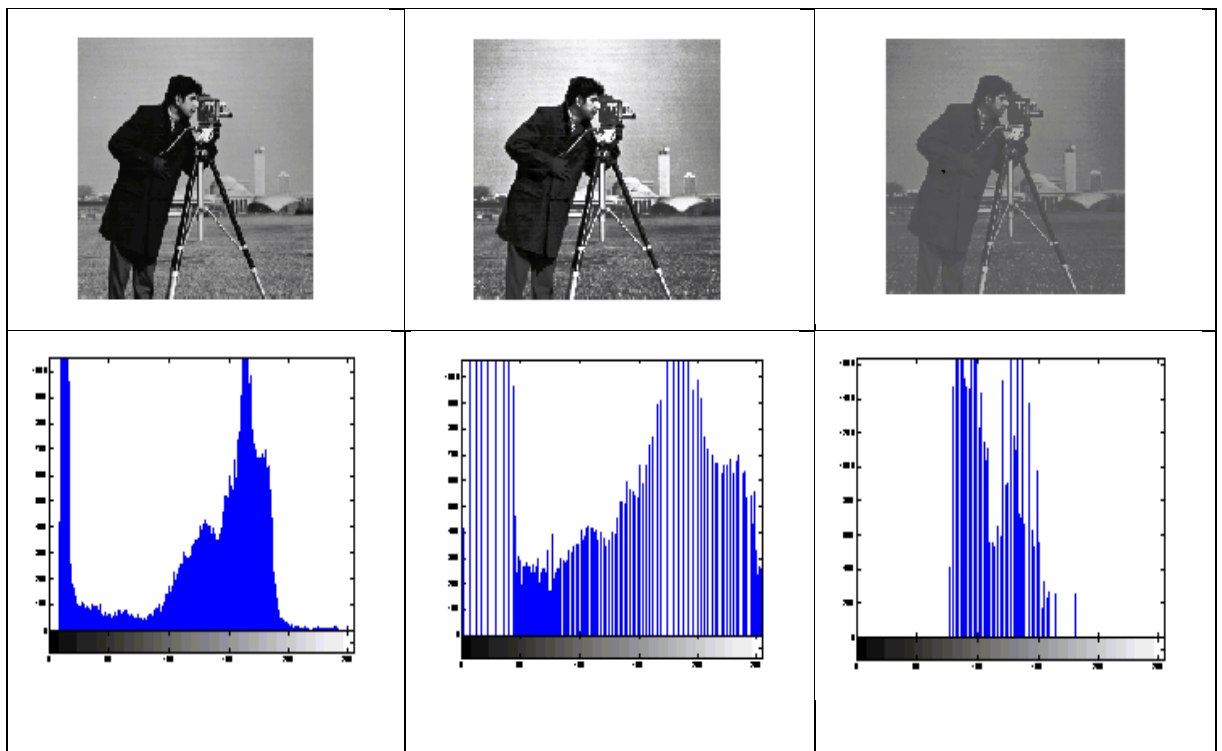


Figure I12- Exemple d'un étirement d'histogramme

I.6.3 Égalisation d'histogramme

L'égalisation d'histogramme est une méthode d'ajustement du contraste d'une image qui utilise l'histogramme. Elle consiste à appliquer une transformation sur chaque pixel de l'image, et donc d'obtenir une nouvelle image à partir d'une opération indépendante sur chacun des pixels. Cette transformation est construite à partir de l'histogramme cumulé de l'image de départ.

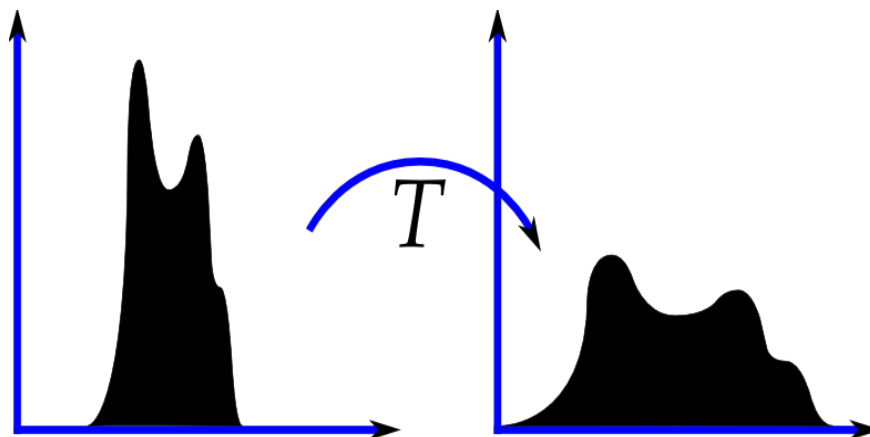


Figure I-13 La transformation T permet d'égaliser l'histogramme et d'améliorer le contraste

I.6.4 Inversion d'image

Il s'agit d'inverser l'intervalle des niveaux de gris (inversion dynamique), en inversant les extrêmes noir et blanc. En effet, parfois, on distingue mieux certains détails en blanc sur fond noir qu'en noir sur fond blanc. [4]

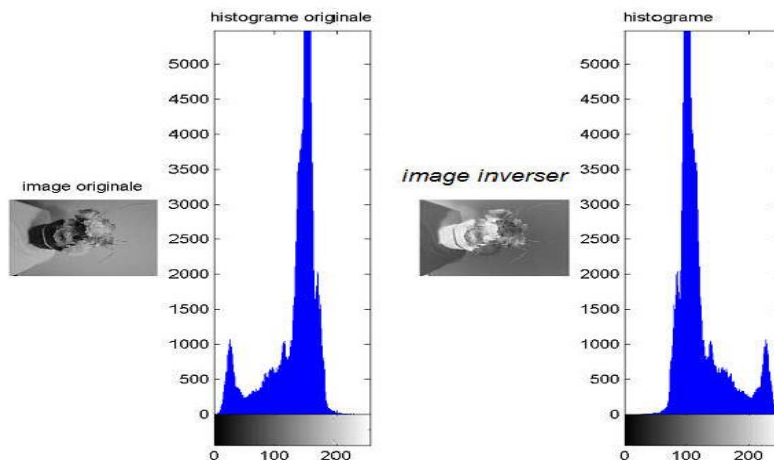


Figure I-14 Exemple d'une inversion d'image et son histogramme

I.7. Notion des proches voisins et contour

I.7.1 Voisinage d'un pixel :

Soit un pixel $e(x, y)$. Ce pixel possède deux voisins horizontaux et deux voisins verticaux de coordonnées :

$$(x+1, y), (x-1, y), (x, y+1) \text{ et } (x, y-1).$$

Cet ensemble est appelé **4 voisins** de e noté par $N_4(e)$, Chaque pixel est à une distance d'un *pixel* par rapport à (x, y) . De même, il a quatre voisins selon les diagonales de coordonnées :

$$(x + 1, y + 1), (x + 1, y-1), (x-1, y+1), (x-1, y-1). [5]$$

Cet ensemble, $(N_D(e))$ de points, plus $N_4(e)$ est appelé **8voisins** de P noté par $N_8(e)$.

Remarque : Pris tous seuls les 4 voisins en diagonale sont notés en N_D -voisins

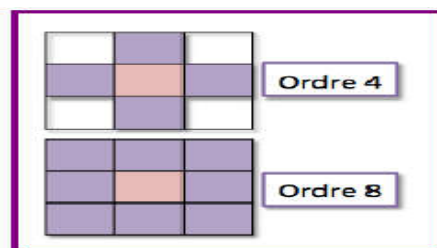


Figure I-15 Représentation du voisinage d'un pixel

I.7.2 Adjacence :

La relation d'adjacence est définie en considérant uniquement des pixels similaires. Dans ce contexte l'intensité du pixel (valeur de niveau de gris) est utilisée pour définir cette relation de similarité.

Différent type d'adjacence

On distingue trois types d'adjacence :

- **4-adjacence**

Deux pixels p et q sont en *4-adjacence* si $q \in V$ et si $q \in N_4(e)$.

- **8-adjacence:**

Deux pixels p et q sont en *8-adjacence* si $q \in V$ et si $q \in N_8(e)$.

- ***m-adjacence***

Deux pixels e et q sont en *m-adjacence* si $q \in V$ et si :

1. $q \in N_4(e)$

Où

2. $q \in N_D(e)$ et \forall pixel $r \in N_4(e) \cap N_4(q)$, $r \in V$.

I.7.3 Chemin

Soit K est le nombre d'adjacences choisies ($K = 4$, $K = 8$, ou $K = 6$).

Un ***K chemin*** du pixel e au pixel q est une suite $x_0, \dots ; x_n$ de pixels tels que : $x_0 = e$ et $x_n = q$ (Où n est un entier naturel). Pour $i = 0, \dots, n-1$, x_i est *K adjacence* à x_{i-1} ; né tant la longueur du chemin.

Un chemin de longueur n comporte donc $n + 1$ pixels et n transitions. Si $x_0 = x_n$ alors on obtient un *chemin fermé*.

I.7.4 Distance :

On définit la *K distance* $K(e, q)$ entre 2 pixels p et q comme étant la longueur minimale d'un *K chemin* de e à q .

I.7.5 Connectivité

La connectivité entre pixels est un concept fondamental dans l'imagerie numérique. En ce sens, qu'il simplifie la définition de beaucoup de concepts, tels que *la région* ou *les frontières* dans une image.

Pour que deux pixels soient connexes, il faut que :

- Qu'ils soient voisins.
- Que leurs niveaux de gris satisfassent à un critère de similarité (plus largement appartenant à un ensemble de valeurs V).

I.8. Régions et frontières

I.8.1 Régions

Soit R un sous ensemble de pixels dans une image, on appelle R une région de l'image I si R est un ensemble *connexe*. [2]

Règles :

- Les relations d'adjacence peuvent être étendues aux régions.
- Deux régions R_1 et R_2 sont adjacentes si $R_1 \cup R_2$ est un ensemble connexe.
- Deux régions non adjacentes sont dites disjointes.
- L'adjacence des régions dépend du type d'adjacence adopté.
- L'ensemble des régions constituent le premier plan de l'image (foregrounds).
- L'ensemble des pixels qui n'appartiennent à aucune région constituent le fond de l'image (background).
- Pratiquement les objets de l'image sont toujours entourés d'un fond.

I.8.2 Frontières

La *frontière* (appelée aussi *bordure* ou *contour*) d'une région R est l'ensemble de pixels de la région R qui ont au moins un de leurs voisins qui n'est pas dans R . [2]

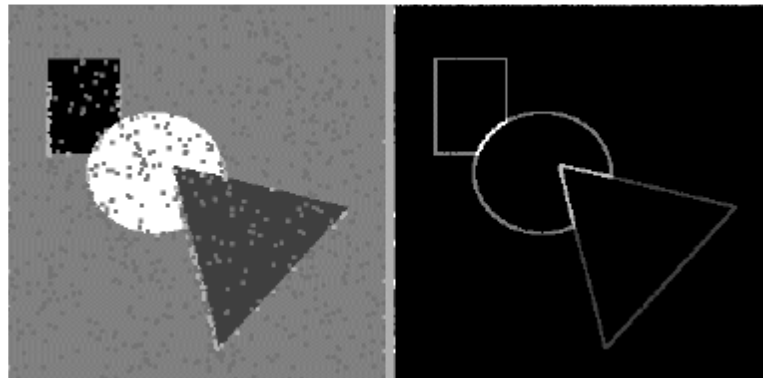


Figure I-16 Représentation les Régions et frontières

Remarques :

- Si la région correspond à toute l'image alors la frontière est constituée par la première et dernière ligne et colonne.
- Normalement quand on parle d'une région, on se réfère à un sous ensemble de l'image. S'il arrive que certains pixels de la frontière sont sur la bordure de l'image, ils sont implicitement inclus dans la frontière de la région.

I.8.3 Chaîne de traitement d'images

Un système de traitement numérique d'images est composé de :

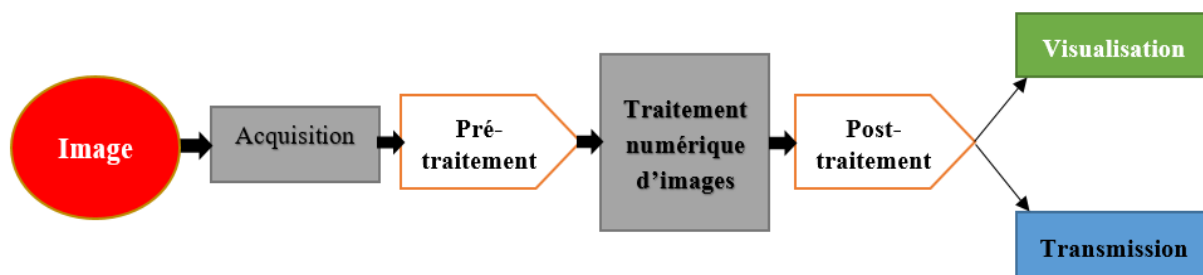


Figure I-17 Schéma d'un système de traitement d'images

- **Acquisition des images** : un grand nombre de difficultés peuvent être résolues au niveau du système de saisie (Choix du type de caméra (couleur ?), Positionnement de la caméra (fixe ?) , Choix de l'arrière-plan (homogène et contrastant ?).
- **Pré-traitement des images** : Enlever le bruit des images : filtres moyenne, médian, ... ; Contraste des images : Égalisation de l'histogramme, Correction de la dynamique de l'image ... ; Image couleur ver niveaux de gris (si nécessaire) : Ou changement d'espace couleur (TSV, Lab.,) ; Modifier la résolution (taille) de l'image.
- **Traitement des images** Étape très difficile ; Chaque algorithme de traitement est très différé ; Le but est de trouver un algorithme + paramétrage unique. Mêmes traitements pour toutes les images ; Un algorithme de traitement seul n'est pas forcément efficace. Combinaison de plusieurs techniques.
- **Post-traitement des résultants** (Après la traitement, besoin de corriger les résultants ; Morphologie mathématique ; Éliminer/corriger les erreurs de traitement...). [5]

I.9. Conclusion

Les techniques de traitement d'image sont des techniques très diverses et le choix de l'une parmi elles, est un choix qui dépend essentiellement de la nature de l'application et des résultats qui peuvent être obtenus par l'application de l'une ou de l'autre. Cependant, chaque ensemble de ces techniques est destiné à une application spécifique.

Dans ce chapitre, nous avons donné un récapitulatif sur les définitions et notions élémentaires liées à l'image numérique représentée par sa forme matricielle. Il s'agissait des notions de pixel, image, formats, notion des proches voisins et contour, régions et frontières. De même, une définition du domaine de traitement d'images est également donnée.



CHAPITRE II :

**GENERALITES SUR LES
TECHNIQUES DE LA
COMPRESSION D'IMAGES**

II. Généralités sur les techniques de la compression d'images

II.1. Introduction

La révolution du numérique impose des échanges de volumes importants d'information ce qui pose des problèmes de stockage et de transfert et de gestion de cette masse de données exponentiellement croissante. D'où l'intérêt de développer des techniques de compression et de décompression de données. Leur but est de changer le format des informations de telle sorte qu'elles occupent moins de volume. Une fois compressées, ces données ne sont plus accessibles en tant que données cohérentes. Pour les récupérer, il suffit de reconstruire l'image par décompression.

Exemple : Transmission par télécopie

Une page à transmettre est constituée de points blancs et noirs. Chaque point est représenté par un élément binaire ("1" si la couleur est noire, "0" si elle est blanche).

Les dimensions de la page sont de **8,5x11** pouces.

Sachant que la résolution est de **200** points par pouce, le nombre d'éléments binaires nécessaires pour représenter une page est donc :

$$(8,5 \times 200) \times (11 \times 200) = 3,74 \text{ Mbits}$$

Si on utilise un modem au débit de **14,4** Kbits/s, le temps nécessaire à la transmission d'une page serait de $3,74 \cdot 10^6 / 14,4 \cdot 10^3 = \mathbf{4 \text{ min } 20 \text{ s}}$.

Grâce aux méthodes de compression, cette durée est réduite à 17s ! [6]

II.2. Définition

La compression est l'opération informatique consistant à transformer une suite de bits A en une suite de bits B plus courte pouvant restituer les mêmes informations, ou des informations voisines, en utilisant un algorithme de décompression. C'est une opération de codage qui raccourcit la taille (de transmission, de stockage) des données au prix d'un travail de compression. Celle-ci est. operation inverse de la decompression.

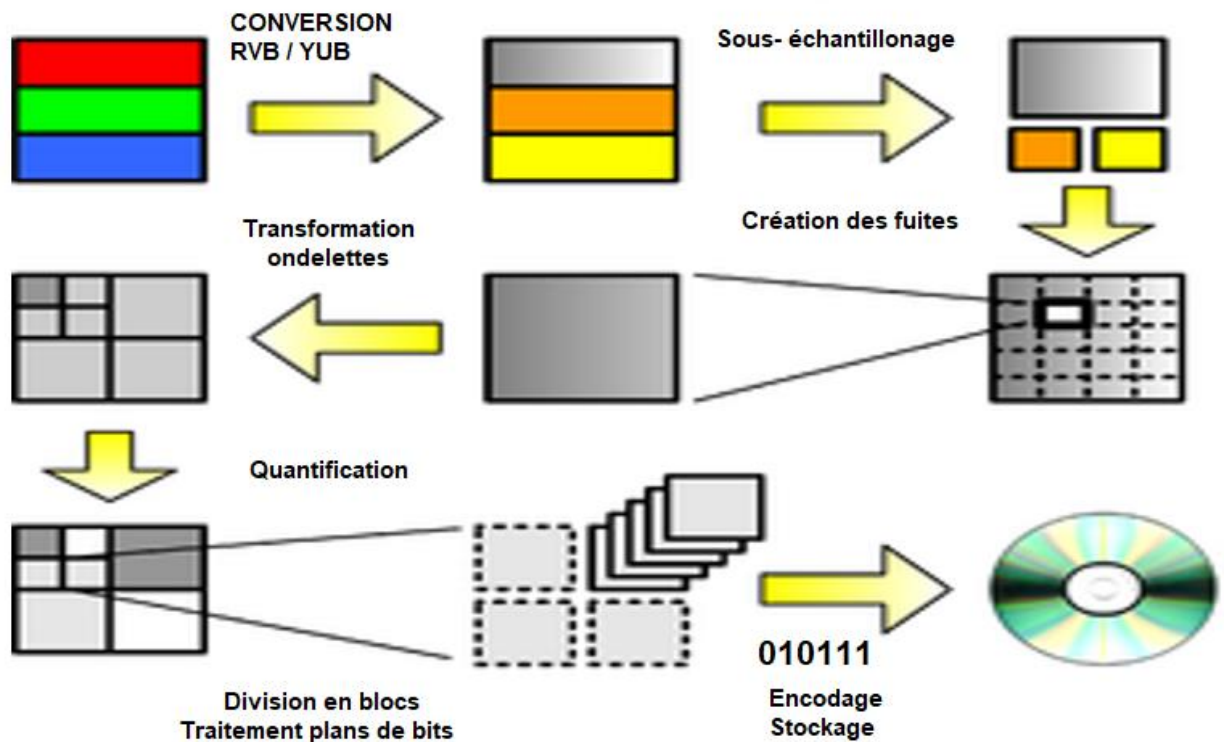


Figure II-1 architecture générale de la compression d'une image

Un algorithme de compression sans perte restitue après décompression une suite de bits strictement identique à l'originale. Les algorithmes de compression sans perte sont utilisés pour les archives, les fichiers exécutables ou les textes.

Pour la compression de données sans pertes, on distingue principalement le codage entropique et le codage algorithmique. Le codage entropique est fondé sur des *a priori* quant à la source. On doit, par exemple, pour le codage de Huffman, transmettre une table de probabilités des symboles de la source. D'autres exemples sont les codages par dictionnaire comme LZ77, LZ78 et LZW. Le codage algorithmique, lui, ne nécessite de transmettre d'autres informations que le résultat du codage et la méthode de compression utilisée.

Avec un algorithme de compression avec perte, la suite de bits obtenue après décompression est plus ou moins voisine de l'original selon la qualité désirée. Les algorithmes de compression avec perte sont utiles pour les images, le son et la vidéo.

II.3. Principe général de la compression des images

D'une manière générale, La compression est également appelée codage source ou réduction de débit, La compression des données, d'une manière générale c'est l'ensemble des méthodes et règles qui permettent de réduire le volume d'une donnée sans perdre les informations essentielles.

D'une manière plus simple, c'est une technique ou on emploie une paire de fonctions, une d'elle a pour but de compresser les données et l'autre de les décompresser.

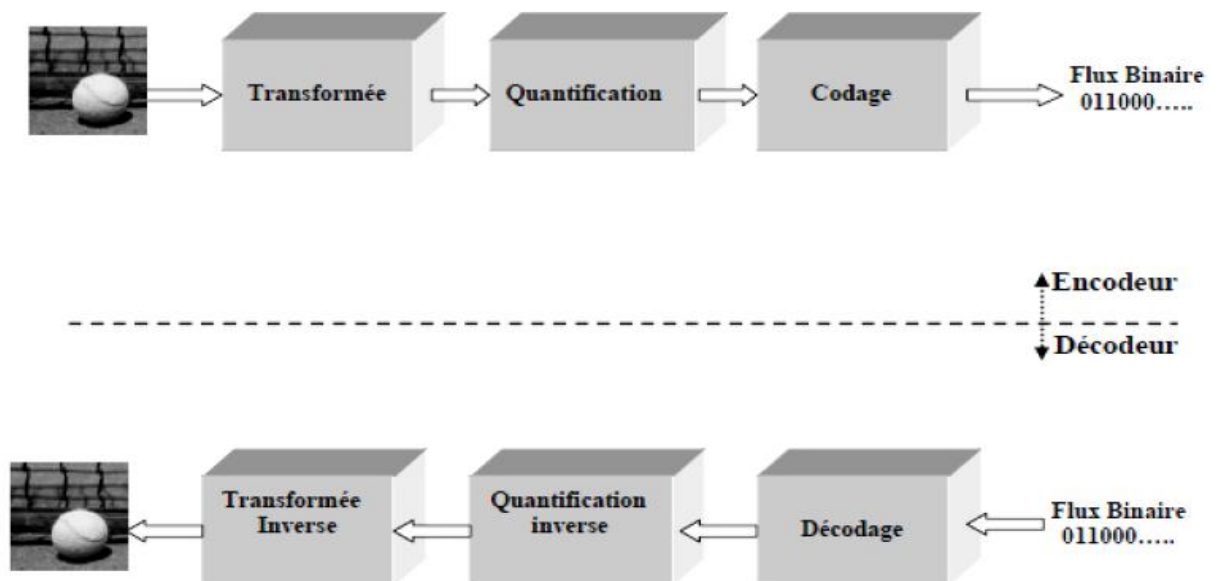


Figure II-2 Schéma de compression /décompression classique d'image [7]

Un système de compression se décompose en trois modules que sont la transformée de l'image originale, la quantification des données de l'image et le codage des données quantifiées.

Première étape : La transformée :

Les pixels composant l'image numérique sont généralement corrélés ceci génère une information redondante qu'il faut exploiter pour la compression afin de diminuer la taille de l'image. En effet, en plus de réorganiser l'information, elle doit représenter les composantes importantes d'un signal avec le moins d'éléments possibles : c'est ce qu'on appelle donner une représentation creuse du signal ou, de manière équivalente, compacter l'énergie.

Deuxième étape : La quantification :

Dans le schéma de compression, l'étape de quantification est celle qui dégrade de manière irréversible l'image. Elle est cependant d'une importance capitale dans la réduction du débit

binaire. Elle représente une étape clé de la compression. Elle approxime chaque valeur d'un signal par un multiple entier d'une quantité q , appelée quantum élémentaire ou pas de quantification.

Troisième étape : Codage :

Le codage entropique est employé à la dernière étape de la chaîne de compression avec pertes. Il consiste à donner la représentation binaire plus compacte de l'entité quantifiée, il existe différents types de codeur utilisés dans la compression d'image notamment, le codeur Hoffman et le codeur arithmétique.

II.4. Classification des méthodes de compression

Les méthodes de compression d'images peuvent être classées en deux grandes classes. Les méthodes de compression sans perte d'informations (non destructives) et les méthodes de compression avec perte d'informations (destructives).

II.4.1 Méthodes de compression sans perte d'informations

Ce sont des méthodes qui permettent de trouver exactement l'image originale. Nous citerons la méthode RLE, la méthode HUFFMAN, la méthode entropique de Shannon-Fano.

II.4.2 La méthode RLE (R.L.E. ou Run Length Encoding)

Cette technique consiste à repérer et à éliminer la redondance des données. Une image contient souvent des surfaces de couleurs identiques, il est donc plus efficace de coder cette couleur et sa répétition que de coder unitairement chaque point. Par exemple la séquence " 4444444 " est remplacée par " 4/7 ". [8]

Ce codage appelé RLE " Run Length Encoding " est intéressant pour des données comportant peu de valeurs différentes et de longues séquences. Par contre il est moins intéressant pour des données textuelles ou de type images photographiques.

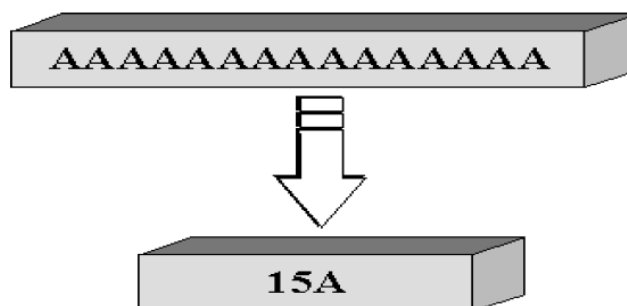


Figure II-3 Compression RLE

Exemple : chaîne de caractère à compresser :

AAAABBBCCCAACCA donnera 4A 3B 3C 2A 2C 1A

Les espaces ne figureront pas dans le fichier compressé, ils ne sont là que pour améliorer la lisibilité.

Ainsi 12 éléments en remplacent 15 dans notre exemple, ce qui nous donne un taux de compression de 20% soit un gain de 3 octets.

II.4.3 La méthode de HUFFMAN

Le codage de Huffman est un algorithme de compression de données sans perte. Le codage de Huffman utilise un code à longueur variable pour représenter un symbole de la source (par exemple un caractère dans un fichier). Le code est déterminé à partir d'une estimation des probabilités d'apparition des symboles de source, un code court étant associé aux symboles de source les plus fréquents. [8]

Un code de Huffman est optimal au sens de la plus courte longueur pour un codage par symbole, et une distribution de probabilité connue. Des méthodes plus complexes réalisant une modélisation probabiliste de la source permettent d'obtenir de meilleurs ratios de compression.

Il a été inventé par David Albert Huffman, et publié en 1952.

Principe de fonctionnement:

- Un code préfixe est un ensemble de mots tel qu'aucun mot de l'ensemble n'est préfixe d'un autre mot de l'ensemble
- Le décodage est alors immédiat
- Un code préfixe sur l'alphabet binaire (0, 1) peut être représenté par un trie qui est un fait un arbre binaire ou tous les nœuds internes ont exactement deux successeurs.

Arbre :

- Les feuilles sont étiquetées avec les caractères originaux, les branches par 0 ou 1 et les chemins depuis la racine jusqu'aux feuilles épellent les codes des caractères originaux.
- L'utilisation d'un code préfixe assure que les codes sont bien représentés par les feuilles.

Encodage :

La phase d'encodage se compose de trois étapes :

- ❖ Comptage des fréquences des caractères
- ❖ Construction du code préfixe
- ❖ Codage du texte

Exemple :

Fenêtre contient cagataagaaa 12 *8 = 96 bits

les fréquences sont	a	c	G	t	FIN
Fréquences	7	1	3	1	1

Table II-1 La phase d'encodage

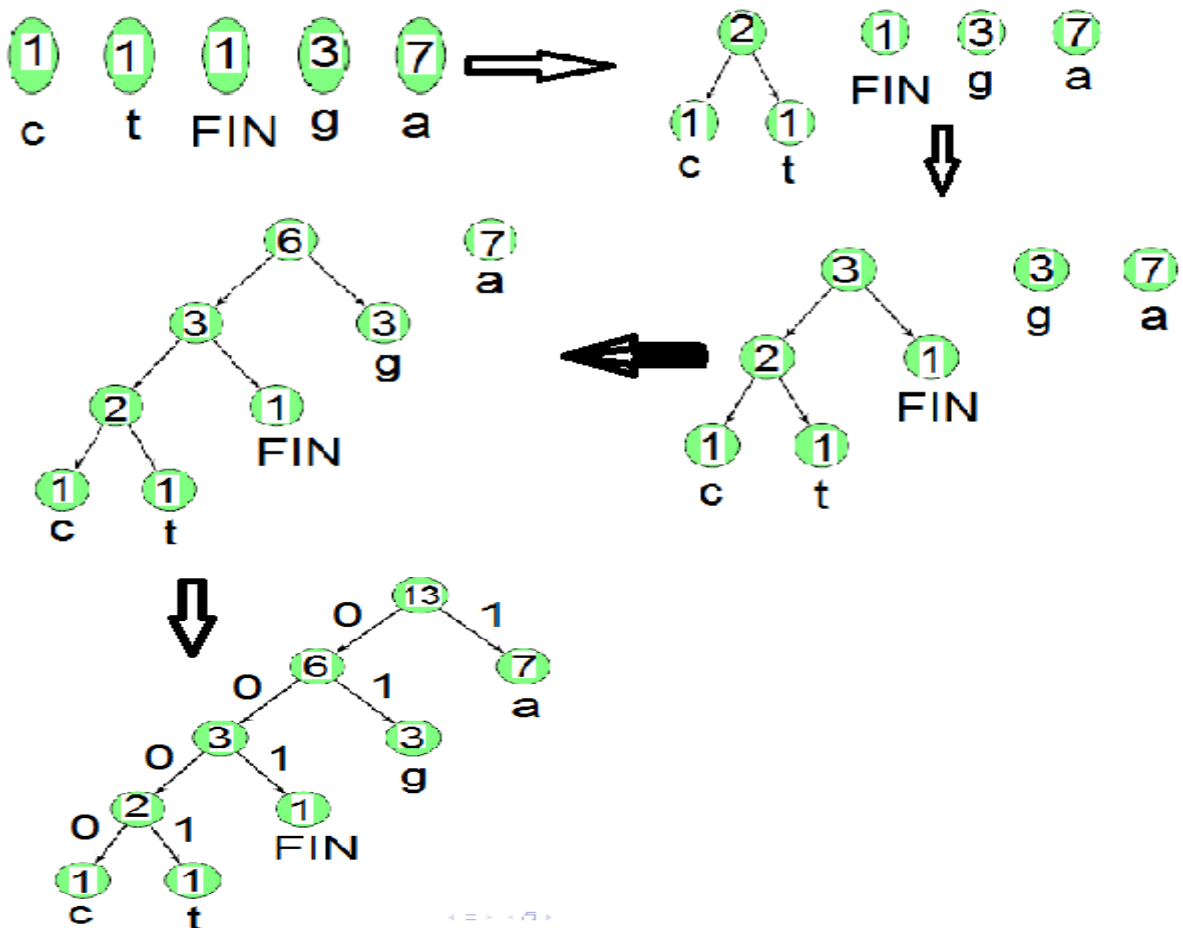


Figure II-4 les étapes de la méthode HUFFMAN

les fréquences sont	a	c	g	T	FIN
code	1	0000	01	0001	001

Table II-2 Codage de l'arbre

Codage de l'arbre :

00001 binaire (c, 9) 1 binaire (t, 9) 1 binaire (FIN, 9) 1 binaire (g, 9) 1 binaire (a, 9)

En ASCII :

00001 001000011 1 001010100 1 100000000 1 001000111 1 001000001

54 bits

Codage du texte :

0000 1 01 1 0001 1 1 01 1 01 1 1 001

24bits

II.4.4 La méthode entropique de Shannon-Fano

Le codage de Shannon est un algorithme de compression de données sans perte élaboré par Robert Fano à partir d'une idée de Claude Shannon. [8]

Il s'agit d'un codage entropique produisant un code préfixe très similaire à un code de Huffman, bien que pas toujours optimal, contrairement à ce dernier.

Les codes sont définis à l'aide d'un algorithme spécifique selon un arbre de Shannon-Fano :

- ✓ Les symboles sont triés et classés en *fonction* de leur fréquence en commençant par le plus fréquent.
- ✓ La liste des symboles est ensuite divisée en deux parties de manière à ce que le total des fréquences de chaque partie soit aussi proche que possible.
- ✓ Le chiffre binaire 0 est affecté à la première partie de la liste, le chiffre 1 à la deuxième partie.
- ✓ Chacune des deux parties fait à son tour l'objet des démarches 2 et 3.

Et ainsi de suite jusqu'à ce que chaque symbole soit devenu une feuille de l'arbre correspondant à un code déterminé.

Exemple : Soit la table des fréquences (ou probabilités) des symboles :

Symbol	Frequency	Effectuons un classement hiérarchique et une première division :				
A	7	D	14	22	0	
B	6	A	7		0	1 ere division
C	5	B	6	16	1	
D	14	C	5		1	
E	4	E	4		1	

Table II-3 fréquences des symboles

D	14	0	0		2 ième division
A	7	0	1		
B	6	1	1		
C	5	1	0	0	3 ième division
E	4	1	0	1	4 ième division

Table II-4 probabilités des symboles

Les codes attachés aux divers symboles sont :

D	0	0	
A	0	1	
B	1	1	
C	1	0	0
E	1	0	1

Table II-5 Exemple de Codage de Shannon-Fano

II.4.5 Codage arithmétique

Le codage arithmétique permet, à partir de la probabilité d'apparition des symboles d'une source de créer un seul mot-code qui soit associé à une séquence de longueur arbitraire de symboles. Ceci diffère du code de Huffman qui attribue des mots-codes de longueurs variables à chaque symbole de la source.

Le code associé à une séquence est un nombre réel de l'intervalle $[0, 1[$. Ce code est construit par subdivisions récursives d'intervalles. Un intervalle est subdivisé pour chaque nouveau symbole qui appartient à la séquence. On obtient, en définitive, un sous-intervalle de l'intervalle $[0, 1[$ tel que tout nombre réel appartenant à cet intervalle représente la séquence à coder. [12]

Procédure de codage arithmétique :

Soit une source $S = \{s_1, s_2, \dots, s_N\}$ avec $p_k = P(s_k)$

$[L_{s_k}, H_{s_k}]$ est l'intervalle « assigné » au symbole s_k avec : $H_{s_k} - L_{s_k} = p_k$

Algorithme de codage :

- **Étape 1:** Initialisation: $L_c = 0$; $H_c = 1$

On initialise un premier intervalle avec deux bornes : la borne inférieure $L_c = 0$ et la borne supérieure $H_c = 1$ (correspondant à la probabilité de choisir un premier symbole s_{a1} parmi tous les symboles s_k de la source).

La taille de cet intervalle est donc défini par : $\text{taille} = H_c - L_c = 1$.

- **Étape 2 :** Déterminer les longueurs des sous intervalles liés aux symboles

Cet intervalle est partitionné en N sous-intervalles $[L_{s_k}, H_{s_k}]$ en fonction des probabilités de chaque symbole s_k de la source. Ce sont les partitions initiales. La longueur ($H_{s_k} - L_{s_k}$) de ces sous intervalles est donnée par : $L_{s_k} - H_{s_k} = p_k$, on a donc :

$$L_{s_k} = L_c + \text{taille} \times \sum_{i=1}^{k-1} p_i \quad \text{et} \quad H_{s_k} = L_c + \text{taille} \times \sum_{i=1}^k p_i$$

- **Étape 3 :** Passer au codage sur le prochain symbole s_k On choisit le sous-intervalle correspondant au prochain symbole s_{ak} qui apparaît dans la séquence. On redéfinit alors l'intervalle initial $[L_c, H_c]$:
$$\left\{ \begin{array}{l} L_c = L_c + \text{taille} \times L_{s_k} \\ H_c = L_c + \text{taille} \times H_{s_k} \end{array} \right.$$
- **Étape 4 :** Déterminer, pour le sous-intervalle considéré, le raffinement en sous intervalles. Cet intervalle est subdivisé à nouveau, selon le même procédé que celui utilisé dans l'étape 2.
- **Étape 5 :** Les étapes 2, 3, et 4 sont répétées jusqu'à obtenir le mot-code représentant la séquence complète des symboles source.

Exemple:

- ✓ On considère la source $S = \{-2, -1, 0, 1, 2\}$ qui définit une série de 5 vecteurs
- ✓ 'Y' est la variable aléatoire qui désigne la valeur d'un vecteur
- ✓ On a les probabilités suivantes :
 - $P\{Y = -2\} = p_1 = 0,1$
 - $P\{Y = -1\} = p_2 = 0,2$
 - $P\{Y = 0\} = p_3 = 0,4$
 - $P\{Y = 1\} = p_4 = 0,2$
 - $P\{Y = 2\} = p_5 = 0,1$
- ✓ L'objectif est de coder la séquence de vecteurs de mouvement (0, -1, 0, 2)

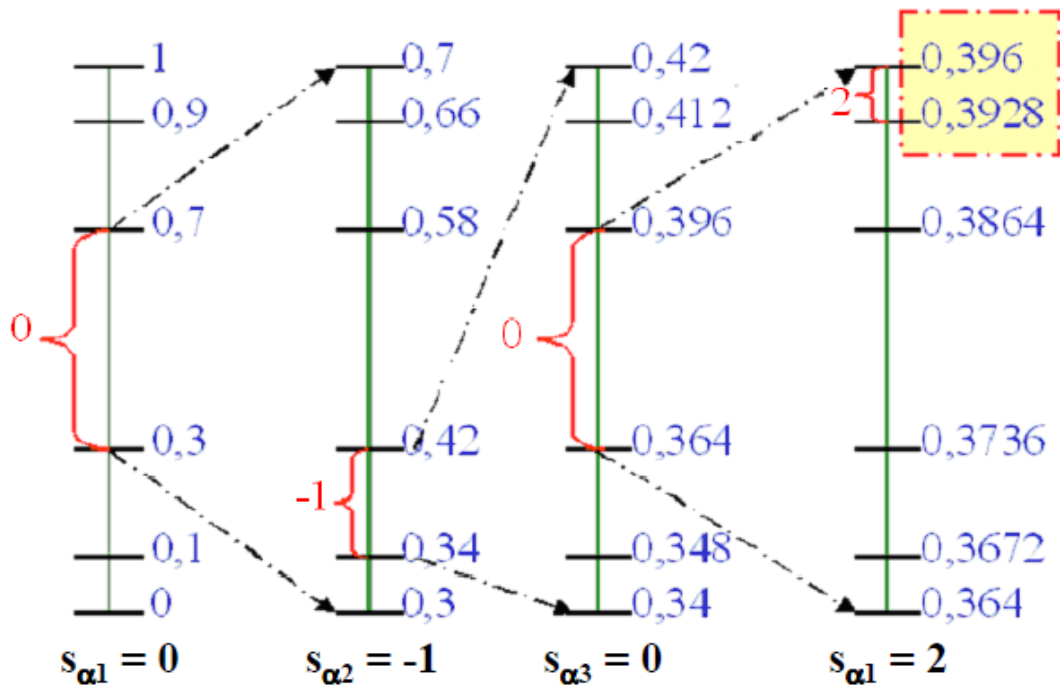


Figure II-5 Subdivisions de la partition initiale

Pour coder la séquence $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\} = \{s_3, s_2, s_3, s_5\} = \{0, -1, 0, 2\}$, on divise l'intervalle $[0, 1]$ en 5 partitions initiales, puis on se place sur le sous-intervalle correspondant au premier vecteur de mouvement de la séquence (de valeur 0). Pour le prochain symbole de la séquence.

Le vecteur de mouvement -1 , on subdivise la partition initiale du premier vecteur '0' en autant d'intervalles qu'il y a de vecteurs possibles. On procède récursivement pour tous les symboles (ici les vecteurs de mouvement) de la séquence. Dès lors, on peut coder la séquence $(0, -1, 0, 2)$ des vecteurs de mouvements verticaux par tout nombre réel appartenant à l'intervalle $[0,3928 ; 0,936]$. Le nombre 0,3945 code cette séquence. Il requiert 8 bits :

$$0,3945 = 0 \cdot 2^{-1} + 2^{-2} + 2^{-3} + 0 \cdot 2^{-4} + 0 \cdot 2^{-5} + 2^{-6} + 0 \cdot 2^{-7} + 2^{-8}$$

Nous utilisons donc $8/5 = 1,6$ bits/symbole.

Contrairement au codage de Huffman, le codage arithmétique permet de coder sur un nombre de bits non nécessairement entier. Cet avantage permet de réaliser une compression plus performante. Cependant le codage arithmétique est plus long que le codage de Huffman car il n'est pas possible de commencer le décodage avant d'avoir obtenu la séquence entière des symboles envoyés par la source.

Les performances du codage arithmétique peuvent être améliorées dès lors que les tables de probabilités ne sont pas fixées (comme c'est le cas dans l'exemple présenté), mais adaptées à la séquence courante et fonction des séquences précédentes. On parle alors de codage arithmétique adaptatif.

II.4.6 Méthodes de Compression avec perte d'informations

Ce type de compression a un rôle tout à fait différent par rapport à celui de sans perte. Le principe de la compression avec perte est d'avoir une permission de supprimer quelques données (qui sont inutiles) de l'information pour avoir une meilleure compression, ce qui signifie qu'après une décompression du compressé on aura un résultat approximatif du fichier original.

D'une manière générale, un système de compression avec pertes se décompose en trois modules que sont la transformation de l'image originale, la quantification des données de l'image transformée et le codage entropique des données quantifiées.

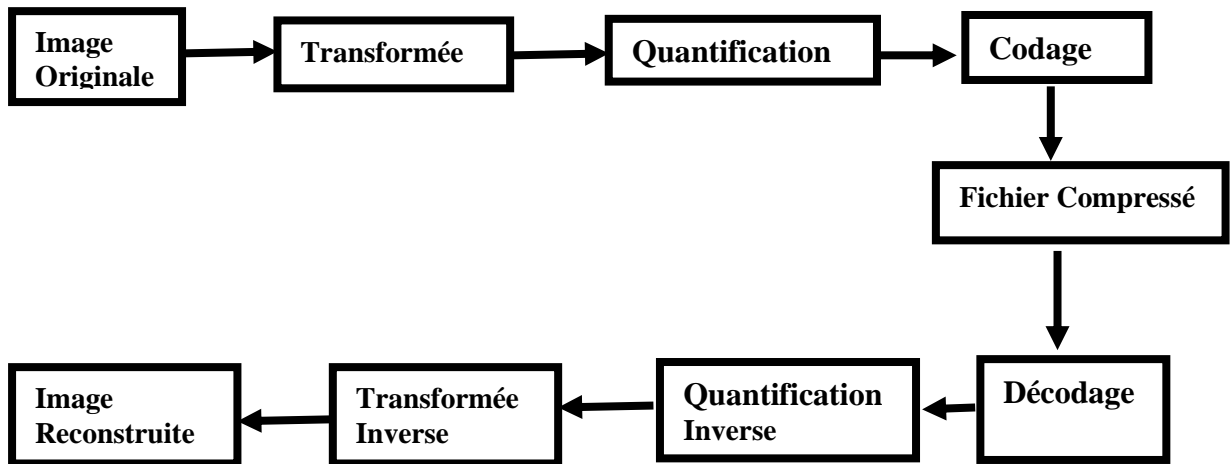


Figure II-6 Principe général des algorithmes de compression/décompression d'image avec perte

A gauche vers la droite : la chaîne de compression classique : transformée de l'image, quantification et codage. A droite vers la gauche : la chaîne de décompression effectue les traitements inverse.

La transformée a pour objectif de projeter les données originales dans un espace plus propice à la compression. En d'autres termes, il s'agit de décarreler le signal de manière à minimiser l'information redondante. Les pertes irréversibles quantification qui consiste à "réduire" l'espace de représentation des données.

Le codage entropique (ou codage sans perte) permet d'encoder les données issues de la quantification suivant un code binaire à longueur variable dépendant de la "quantité d'information" portée par le message.

Les sections suivantes sont consacrées à dresser l'état de l'art non exhaustif de ces trois blocs de traitement dans l'ordre suivant : codage entropique, quantification et codage par transformée.

Les méthodes les plus efficaces sont basées sur de puissants outils tels que :

La transformée discrète en cosinus(DCT), la transformée en ondelettes ou l'approche Fractale.

a Transformation en cosinus discrète DCT

C'est une méthode très utilisée dans tous les domaines d'imagerie y compris le domaine médical. La matrice de transformation DCT est complètement dépendante de l'image, cette transformation est très utilisée pour l'exécution des algorithmes rapide en calcul. [9]

Les méthodes de compression avec perte qui sont basées sur la DCT suivie d'une quantification et d'un codeur entropique, représentées sur la figure II-7.

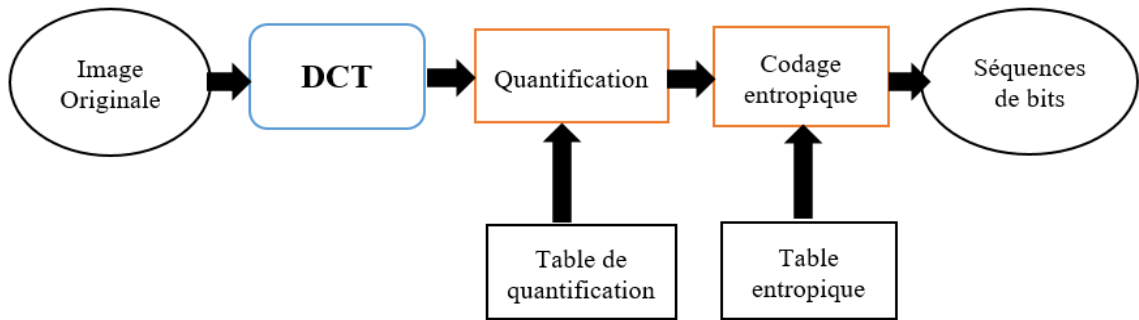


Figure II-8 principe de compression DCT avec perte

b Transformation en ondelettes discrète DWT

La transformation en ondelettes discrète (DWT) bidimensionnelle repose sur la notion d'analyse multi-résolution d'une image. Celle-ci est décomposée en un ensemble de sous bandes représentant l'information portée par l'image source à différent niveau de résolution : l'image d'approximation étant une version réduite et lissée de l'image initiale tandis que les images de détails « horizontaux, verticaux, diagonaux » contiennent uniquement des informations relatives à la texture locale et aux contours des régions de l'image, à une résolution donnée et selon une direction donnée. [10]

Le codage et décodage d'une image au s'effectuent en quatre étapes principales : Les trois étapes classiques en compression d'image (Transformation, quantification, codage) plus une étape de Pré-traitement de l'image qui a pour but de rendre l'opération de codage plus efficace. Ces différentes étapes sont illustrées sur la figure II-9.

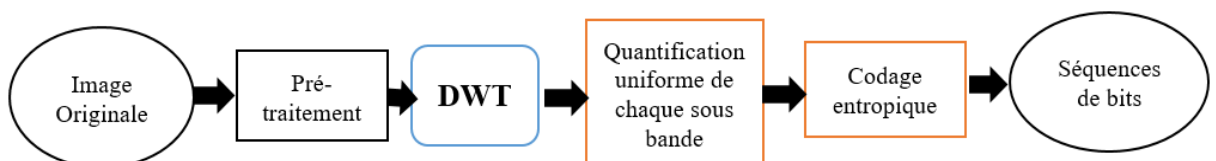


Figure II-10 Principe de compression DWT avec perte

II.5. Critères de mesures des performances de compression

II.5.1 Taux de compression

Une mesure courante pour déterminer le degré de compression obtenu est le taux de compression CR. Il est défini par :

$$CR = \frac{\text{Nombr de bits de fichier originale}}{\text{Nombr de bits de fichier compressé}}$$

La théorie de l'information donne une limite théorique au CR maximal qu'il est possible d'atteindre sans distorsion pour n'importe quelle méthode de compression. Les propriétés statistiques des informations originales jouent un rôle prépondérant dans le résultat obtenu.

II.5.2 Gain de compression:

$$G = 1 - CR = \frac{\text{Taille}_{\text{Initiale}} - \text{Taille}_{\text{Finale}}}{\text{Taille}_{\text{Initiale}}}$$

II.5.3 Mesures de distorsion

La distorsion (D) est l'erreur introduite par l'opération de compression, due au fait qu'éventuellement l'image reconstruite n'est pas exactement identique à l'image originale. La mesure de distorsion utilisée généralement en compression d'image est l'erreur quadratique moyenne (EQM : MSE (Mean Square Error)). Cette grandeur est définie par la moyenne des écarts au carré entre le pixel de l'image originale $I_o(i, j)$, et le pixel (i, j) de l'image reconstruite $I_r(i, j)$. Elle définit par l'équation suivante :

$$EQM = MSE = \frac{1}{M \cdot N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (I_o(i, j) - I_r(i, j))^2$$

Avec :

- ✓ $M \times N$: Taille de l'image
- ✓ $I_o(i, j)$: Un pixel de l'image originale
- ✓ $I_r(i, j)$: Un pixel de l'image reconstruite

II.5.4 Rapport signal à bruit en pic PSNR (Peak Signal to Noise Ratio)

Un rapport signal sur bruit crête pour une image dont le maximum est $2^R - 1$ dénoté PSNR (Pic Signal to Noise Ratio) [25], il se mesure en décibel (dB) :

$$PSNR = 10 \cdot \log_{10} \frac{(2^R - 1)^2}{MSE}$$

Où R représente le nombre de bits désignés pour un pixel.

La valeur typique de PSNR varie entre 20 dB et 40 dB. [11]

II.5.5 Temps de calcul

La contrainte du temps est un facteur essentiel dans l'évaluation des performances de toute méthode de compression, elle revient à calculer le temps pris par la compression et la décompression des images. Cette contrainte est plus au moins imposée selon l'application visée par la compression (transmission ou archivage). En effet, il serait dommage, dans une application de transmission, que le temps gagné par une réduction de la taille des données à transmettre soit inférieur au temps passé à la compression décompression. Cette qualité sera cependant moins cruciale dans des applications visant l'archivage de données.

II.6. Conclusion

La compression des données est appelée à prendre un rôle encore plus important en raison du développement des réseaux et du multimédia. Les méthodes déjà utilisées couramment sont efficaces et modéré (RLE, Huffman, Shannon-Fano). On a présenté au sein du chapitre les techniques de compression des données, une introduction aux concepts de base sur la compression des données et les techniques de compression des données sans/ avec perte et les critères de mesures des performances.



CHAPITRE III

COMPRESSION PAR FUSION

TECHNIQUES

III. Compression par fusion technique

III.1. Introduction

Les méthodes hybrides définies comme étant une combinaison des méthodes traditionnelles précédemment présentées an d'en palier les limites, ces dernières sont actuellement les plus représentées dans la littérature, notamment à cause du fait qu'elles soient jugées comme étant les plus effaces. Selon ce dernier, un système hybride est généralement organisé en deux phases :

- Effectuer de manière indépendante les techniques de compression des items via des méthodes collaboratives ou par le contenu (ou autres).
- Combiner ces ensembles de recommandations via des méthodes d'hybridations telles que des pondérations, commutations, cascade, etc.

Ces derniers temps, le développement de systèmes hybrides pour améliorer la qualité de l'image compressé a acquis une énorme popularité parmi les chercheurs. Certaines études récentes montrent que la méthode hybride par l'approche fondée sur la transformation de Walsh, les ondelettes et les techniques de codage de source d'information conduit à un meilleur taux de compression.

Dans cette mémoire, nous proposons un algorithme de compression hybride d'images médicales basé sur la DWT couplée avec la transformée de Walsh. Les fonctions de Walsh forment un système orthonormé qui peut être appliqué dans de nombreuses situations et sont très intéressantes à la fois théoriques et des points de vue pratiques. La DWT phare de ces dernières années pour la compression des images. Elle réputée pour sa simplicité d'utilisation, ses bonnes propriétés de décorrélation, sa représentation multi résolution, etc.

III.2. La transformée en ondelettes

La transformée en ondelettes est une représentation multi-résolutions/multifréquences. C'est un outil qui découpe les données, les fonctions ou les opérateurs en composantes fréquentielles suivant une résolution adaptée à l'échelle. [13]

Ce principe est illustré en figure *III-1*.

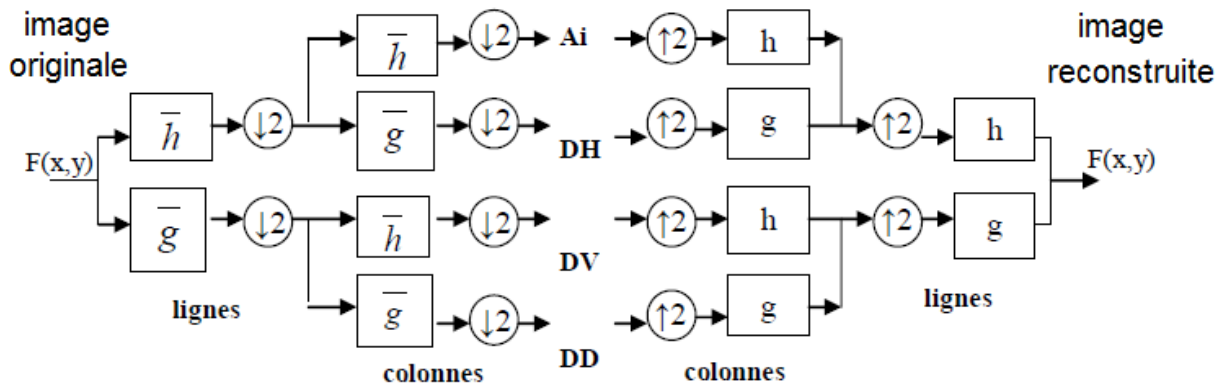


Figure III-2 Analyse multi-résolutions en deux dimensions



Décimation d'une ligne (colonne) sur deux



Interpolation des lignes (colonnes), c'est-à-dire mettre une ligne (colonne) de zéros entre deux.

III.2.1 Choix d'ondelettes

Il n'y a pas une ondelette qui soit meilleure qu'une autre. Tout dépend de l'application utilisée. Dans certains cas, l'ondelette la plus simple (Haar) sera optimale. Pour d'autres applications, ce sera le pire des choix possibles. En pratique, il semblerait que l'élément le plus important soit le nombre de moments nuls. Pour la plupart des applications, il est désirable d'avoir le plus de coefficients d'ondelettes nuls et donc plus de moments nuls implique une meilleure transformation. Cependant, les ondelettes ayant un plus grand nombre de moments ont aussi un support plus grand ce qui signifie que si la fonction ou le signal a des discontinuités brusques, elle sera plus sujette aux phénomènes de Gibbs. [13]

III.2.2 Algorithme transformée des ondelettes à deux dimensions

On distingue 4 étapes différentes pour procéder à la transformation dans le cas très particulier de l'ondelette de Haar :

- ❖ Moyenner les pixels de l'image originale deux à deux suivant l'axe horizontal ; par la formule : $H(x) = \frac{X_i + X_{i+1}}{2}$
- ❖ Calculer l'erreur entre l'image originale et l'image sous-échantillonnées dans le sens horizontal ; par la formule : $G(x) = \frac{X_i - X_{i+1}}{2}$
- ❖ Pour chacune des deux images intermédiaires, moyenner les pixels deux à deux suivant l'axe vertical ; par la formule : $H(y) = \frac{Y_i + Y_{i+1}}{2}$

- ❖ Pour chacune des deux images intermédiaires, calculer l'erreur suivant l'axe vertical ;
par la formule : $G(y) = \frac{Y_i + Y_{i+1}}{2}$

La figure suivante représente un exemple comment ça marche la compression avec DWT. La compression par Ondelette est une technique récente qui donne de très bons résultats, même avec des taux de compression élevés.



Figure III-3 schéma de décomposition en ondelette niveau 1

III.3. La transformée de Walsh

La transformée de Walsh est l'analogue de la transformée de Fourier discrète. Elle opère sur un corps fini à la place des nombres complexes. Cette transformée est utilisée pour éliminer le bruit et préserver la structure de l'image dans une image échantillonnée et dans des applications de codage, compression d'images et reconnaissance de formes. [13]

III.3.1 La fonction de Walsh

Nommée d'après Joseph L. Walsh, sont un ensemble de fonctions qui forment une base hilbertienne de l'espace $L^2([0, 1])$ des fonctions de carré intégrable sur l'intervalle unité.

Ces fonctions prennent uniquement les valeurs -1 et 1 , sur des sous-intervalles définis par les fractions dyadiques. Elles sont utiles en électronique et d'autres applications en ingénierie.

Les fonctions orthogonales de Walsh sont utilisées pour effectuer les transformées de Hadamard, qui sont très similaires aux sinusoïdales orthogonales employées dans le cadre de la transformée de Fourier. Les fonctions de Walsh partagent également des similitudes avec l'ondelette de Haar. Le système de Haar est toutefois préférable dans certaines situations où la localisation est nécessaire (alors que les fonctions de Walsh sont bornées) ou d'autres caractéristiques propres aux ondelettes doivent être respectées.

III.3.2 Matrice de Hadamard

Une matrice de Hadamard est une matrice carrée dont les coefficients sont tous 1 ou -1 et dont les lignes sont toutes orthogonales entre elles. Le nom retenu pour ces matrices rend hommage au mathématicien français Jacques Hadamard. Des exemples de telles matrices avaient été donnés par James Joseph Sylvester.

Pour une matrice H d'ordre n , la propriété d'orthogonalité des colonnes peut également s'écrire sous la forme $H^t \times H = nI_n$

où I_n est la matrice identité d'ordre n et H^t est la matrice transposée de H .

Exemples :

$$H_1 = (1)$$

$$H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$H_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}$$

Pour $k=3,4,\dots$

$$H_{2^k} = \begin{pmatrix} H_{2^{k-1}} & H_{2^{k-1}} \\ H_{2^{k-1}} & -H_{2^{k-1}} \end{pmatrix}$$

La transformée de Hadamard peut être vue comme étant issue d'une transformée de Fourier discrète et s'avère être en fait l'équivalent d'une transformée de Fourier discrète multidimensionnelle d'une taille de $2 \times 2 \times \dots \times 2 \times 2$. Elle décompose un vecteur arbitraire en entrée en une superposition de fonctions de Walsh.

III.3.3 Les transformées de Walsh directe et inverse

Sont alors définies comme suit :

Si $X(i,j)$ est une suite périodique finie (matrice) de $n \times n$ éléments alors la transformée de Walsh donnée par :

$$W_n = H_n \times X_n \times H_n$$

La transformée inverse est réalisée par la même transformation :

$$X_n = H_n^{-1} \times W_n \times H_n^{-1}$$

Tq: H_n la matrice de Hadamard et H_n^{-1} matrice de Hadamard inverse

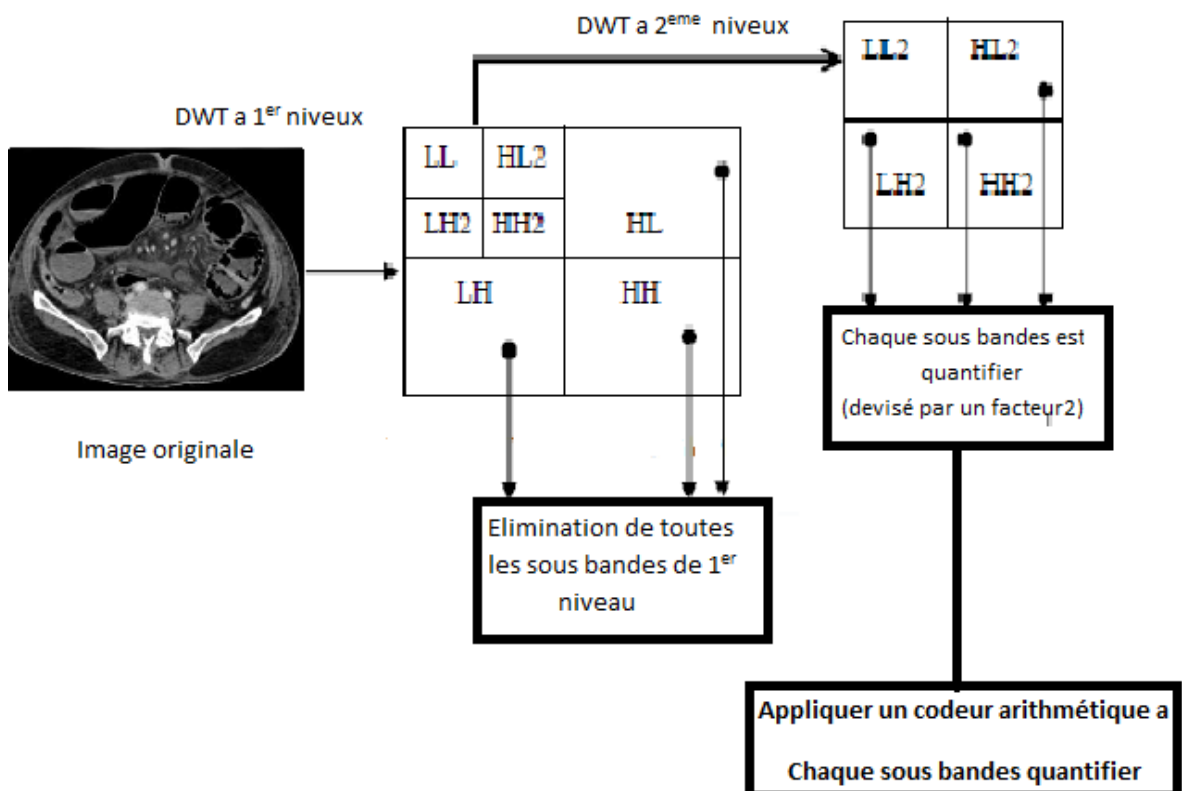
III.3.4 L'algorithme de WWT (Wavelet Walsh Transform)

Ces derniers temps, le développement de systèmes hybrides pour améliorer la qualité de l'image compressé a acquis une énorme popularité parmi les chercheurs. Certaines études récentes montrent que la méthode hybride par la transformation Walsh-Hadamard et les ondelettes classiques conduites à un meilleur taux de compression.

Les méthodes hybrides sont une combinaison de plusieurs techniques de compression (spatiales et/ou transformées) visant à jumeler les avantages et inconvénients de celles-ci, afin d'avoir un taux de compression élevé, une meilleure qualité de restitution de l'image et un temps de calcul minimal lors du processus de codage. [7]

Ce travail présent un algorithme se compose en 4 étapes :

- Appliquer une décomposition par ondelette en deux niveaux.
- Appliquer la transformée de Walsh-Hadamard 2D sur chaque block 8x8 (ou 16 x16...) de là sous bande de la basse fréquence (approximation).
- Diviser tous les valeurs de DC formé par chaque transformées 8x8 blocs
- Compresser chaque sous-bande en utilisant un codage arithmétique



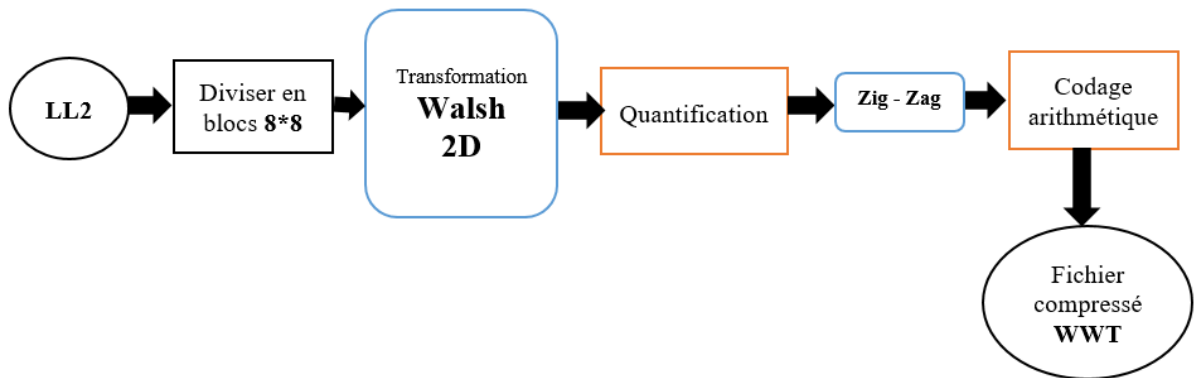


Figure III-4 architecture de Compression hybride (DWT+Walsh+ Aret-Code)

- Nous appliquons la transformée en ondelette avec deux niveaux de décompositions sur notre image. On élimine toutes les sous bandes du premier niveau (HL1, LH1, HH1).
- La dernière étape consiste à appliquer la transformée de Walsh sur chaque bloc de 8x8 de l'approximation LL2, puis de diviser chaque bloc quantifié par un facteur choisi par un utilisateur / programmeur.
- Le codage du reste de la matrice Walsh quantifiée va se faire en parcourant les éléments dans l'ordre imposé par une séquence particulière appelée séquence zigzag. Le codage s'effectue en zigzag comme le montre la figure suivante :

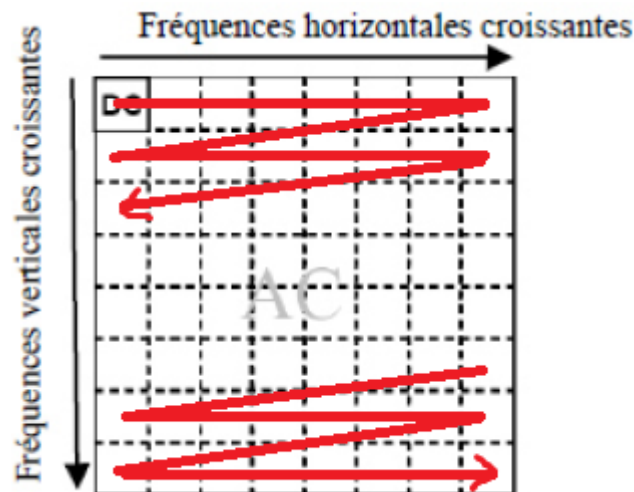


Figure III-5 Distribution Balisage Zig-zag

- Stockés les valeurs dans une matrice des données. Cette dernière est convertie en un vecteur unidimensionnel, et puis le compresser avec le codeur arithmétique.

III.4. Processus de compression et décompression d'images

Par Algorithmes de codage et par Plan de bits

Le processus de compression et de décompression d'images fixes donné dans la Littérature suit les étapes résumées par le schéma synoptique suivant :

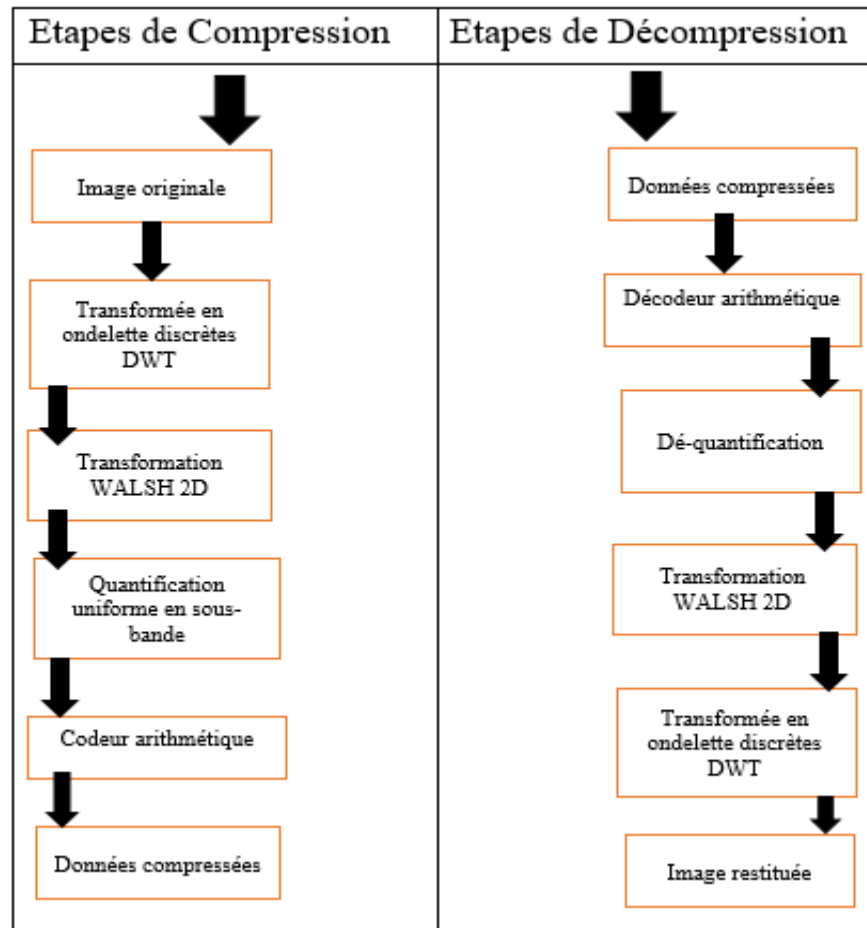


Figure III-6 Schéma Général des étapes de la compression et de la décompression

❖ L'algorithme de compression qu'on a développé sous MATLAB, est basé sur un codage 2-DWT_2-Walsh_ Aret-Code réversible, c'est à dire que l'image reconstruite après compression est un duplicata exact de l'image initiale. Cependant le système mis en œuvre peut être modifié afin de réaliser une compression irréversible, plus performante en termes de taux de compression.

III.5. Tests et Résultats

III.5.1 Images de test

Pour notre application, nous avons utilisé des images testes en niveaux de gris (codées sur 24 bits) de différents types et de différentes tailles.

III.5.2 Paramètres d'évaluation :

Les paramètres les plus utilisés dans le domaine de la compression des images ; témoignés de la qualité des images ainsi que leurs tailles mémoire occupée sont le rapport signal sur bruit crête (PNSR), taux de compression Rc (Bpp : Bits par pixel) /gain de compression, et le temps de compression/décompression.

III.5.3 Logiciel de l'implémentation

Le logiciel Matlab constitue un système interactif et convivial de calcul numérique et de visualisation graphique. Destiné aux ingénieurs, aux techniciens et aux scientifiques, c'est un outil très utilisé, dans les universités comme dans le monde industriel, qui intègre des centaines de fonctions mathématiques et d'analyse numérique (traitement de signal, traitement d'images, visualisations graphiques, etc...).

L'algorithme mentionné a été mis en œuvre sous Matlab 2010a et Matlab 2016a avec un PC Intel(R) Core (TM) i3- CPU ; 2.20GHz PC : 4Go de RAM utilisant.

- **Test 1**

La première teste se fait sur une image médicale :

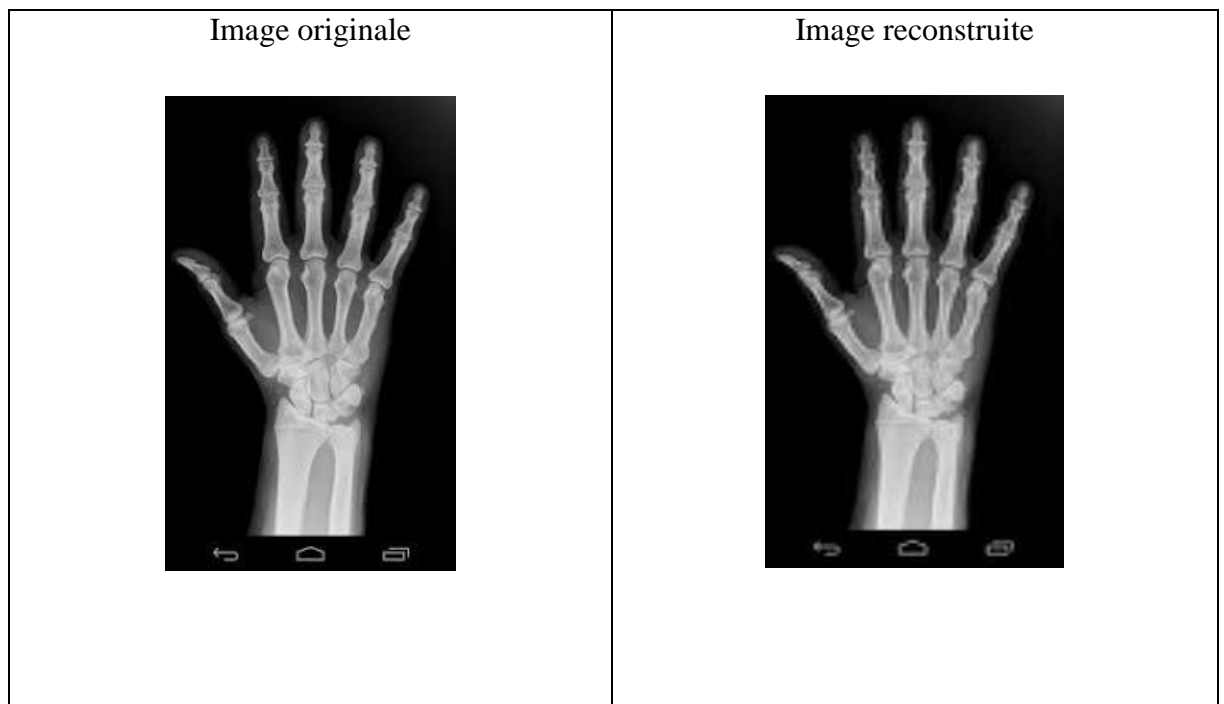


Figure III-7 représente la reconstruction de l'image médicale (main)

Le tableau suivant montre les paramètres d'évaluation d'une image en niveaux de gris.

Temps de compression	1.251418 seconds
Temps de décompression	1.772441 seconds
Taux de compression	5%
Gain de compression	80.01%
PSNR	22.3031dB
Taille de l'image originale	40.86 ko
Taille de l'image reconstruite	4.86 ko

Table III-1 représente les paramètres d'évaluation du test 1

- **Test 2**

La deuxième teste se fait sur une image biomédicale :



Figure III-8 représente la reconstruction de l'image biomédicale (Œil)

Temps de compression	9.329387 seconds
Temps de décompression	13.468330 seconds
Taux de compression	32%
Gain de compression	96,88%
PSNR	30.8517 dB
Taille de l'image originale	768 ko
Taille de l'image reconstruite	20 ko

Table III-2 représente les paramètres d'évaluation du test 2

- **Test 3**

Dans le troisième test on a expérimenté la reconstruction d'une image photographie :



Figure III-9 représente la reconstruction de l'image photographie (vue de Ghardaïa)

Temps de compression	21.003036 seconds
Temps de décompression	25.491921 seconds
Taux de compression	6.88 %
Gain de compression	85,47 %
PSNR	27.1504 dB
Taille de l'image originale	773 ko
Taille de l'image reconstruite	53 ko

Table III-3 représente les paramètres d'évaluation du test 3

- **Test 4**

Le quatrième test est réservé à la reconnaissance des objets :

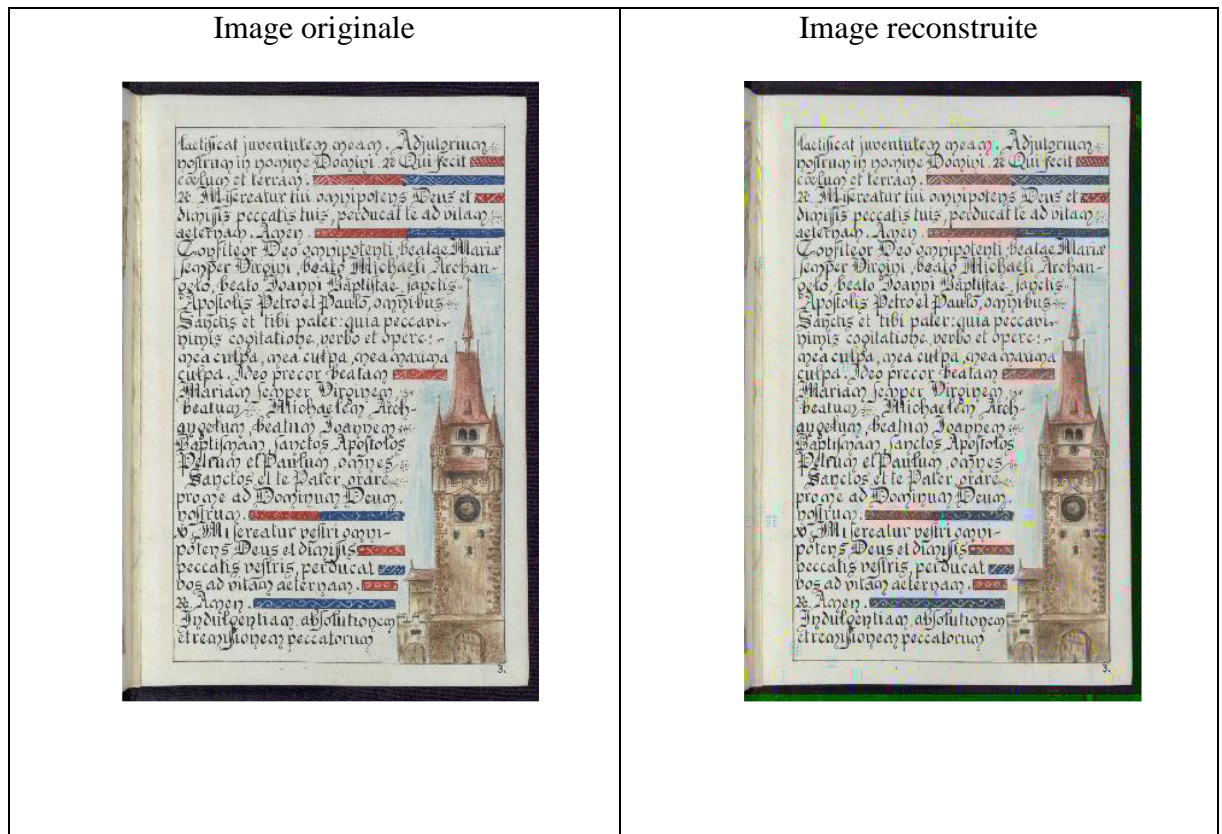


Figure III-10 représente la reconstruction d'une image de reconnaissance des objets (historique)

Temps de compression	218.849722 seconds
Temps de décompression	301.316688 seconds
Taux de compression	23.90 %
Gain de compression	95.81 %
PSNR	27.8284 dB
Taille de l'image originale	516 ko
Taille de l'image reconstruite	27.7 ko

Table III-4 représente les paramètres d'évaluation du test 4

- **Test 5**

La dernière teste se fait sur une image de profondeur marine :

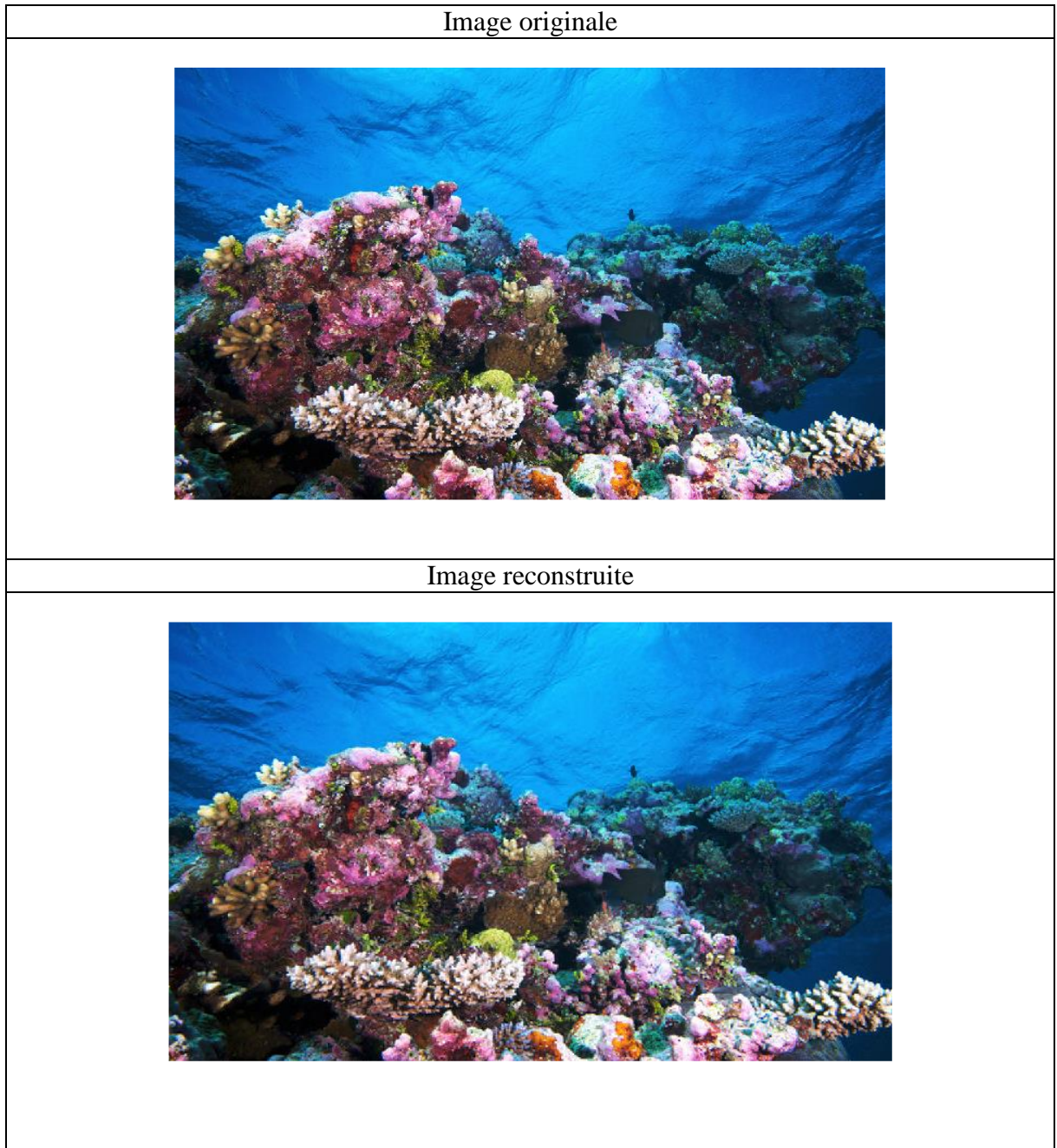


Figure III-11 représente la reconstruction d'une image photographie de profondeur marine (fond marin)

Temps de compression	59.630681 seconds
Temps de décompression	88.577852 seconds
Taux de compression	8.57 %
Gain de compression	88.33 %
PSNR	27.51 dB
Taille de l'image originale	817 ko
Taille de l'image reconstruite	57 ko

Table III-5 représente les paramètres d'évaluation du test 5

Tableau comparatif :

	<i>Teste 1</i>	<i>Teste 2</i>	<i>Teste 3</i>	<i>Teste 4</i>	<i>Teste 5</i>
<i>Image</i>	<i>Médicale</i>	<i>biomédicale</i>	<i>Nature</i>	<i>Reconnaissance des objet</i>	<i>Fond marin</i>
<i>Type image</i>	Niveau de gris	Couleur	Couleur	Couleur	Couleur
<i>Temps de compression (seconds)</i>	1.251418	9.329387	21.003036	218.849722	59.630681
<i>Temps de décompression (seconds)</i>	1.772441	13.468330	25.491921	301.316688	88.577852
<i>Taux de compression (%)</i>	5	32	6.88	23.90	8.57
<i>Gain de compression (%)</i>	80.01	96,88	85,47	95.81	88.33
<i>PSNR (dB)</i>	22.3031	30.8517	27.1504	27.8284	27.51
<i>Taille de l'image originale (ko)</i>	40.86	768	773	516	817
<i>Taille de l'image reconstruite (ko)</i>	4.86	20	53	27.7	57

III.5.4 Interprétations des tests

- ✓ Le résultat est satisfaisant, mais un meilleur compromis entre le taux de compression et la qualité visuelle peut être obtenu en utilisant des méthodes de compression plus sophistiquées, qui impliquent des étapes de seuillage et de quantification plus strictes.
- ✓ Ces résultats ont montré que l'utilisation d'un niveau de décomposition égal à 2 est suffisant pour la compression d'images couleurs, et donne de meilleurs résultats en termes de PSNR et Gain de compression.
- ✓ Compression hybride (DWT +Walsh+ Aret-Code) est très efficace pour la reconnaissance des objets et transmission d'information.
- ✓ Temps de compression/ décompression est lié avec la taille de l'image originale.

III.6. Conclusion

En général, on peut dire que l'approche proposée permet d'atteindre de très bons résultats des points de vue qualité des images reconstruites d'une part et le taux de compression d'autre part. En effet, elle permet de réduire énormément la quantité de données contenues dans les images en niveau de gris et couleur. Tout en conservant leurs propriétés originales. Ces résultats dépendent du type de l'ondelette, de l'algorithme de couple Walsh et DWT et du type de codage entropique utilisés (codeur arithmétique).

A graphic of a rolled-up scroll with a ribbon, containing the text 'CONCLUSION GENERALE'. The scroll is white with a black outline and is set against a light gray shadow. The ribbon is white with a black outline and is tied around the scroll. The text is in a bold, black, serif font.

**CONCLUSION
GENERALE**

Conclusion générale

Les nombreuses recherches et documentations effectuées dans le domaine des images et de leur compression nous montrent l'importance de celle-ci dans la technologie numérique. En effet le monde moderne qui demande de plus en plus d'informations et d'images fait appel aux nouvelles technologies pour permettre de stocker, transmettre et exploiter ce flux gigantesque d'information et d'images. Diverses méthodes de compression ont été proposées afin de répondre aux besoins du moment.

- Techniques destructives dont l'image restituée présente des dégradations mais avec un taux de compression raisonnable.
- Techniques non destructives ne permettant pas une réduction significative du volume.

Les techniques de compression avec perte assurent généralement des taux de compression relativement élevé mais avec un certain degré de dégradation. Les objectifs des fusion (hybridation) techniques des nouvelles techniques de compression est donc de réduire plus et d'altérer le moins possibles. Le but de notre mémoire est d'aboutir à une compression de données d'images tout en gardant la qualité essentielle de l'image originale, c'est-à-dire, avoir un bon compromis entre PSNR et taux de compression. Cette méthode a l'avantage d'une grande précision de reconstruction.

Les différentes simulations sur les images en niveau de gris et couleur, nous ont montré que la compression hybride DWT+Walsh donne des résultats satisfaisants et encourageants en terme de taux de compression qui dépassent 85%, temps de calcul acceptable par rapport de la taille de l'image et une bonne qualité de l'image compressée.

Perspectives

Pour améliorer l'algorithme automatique de l'identification nous proposons comme perspectives de ce travail :

- Combiner les transformée DWT, DCT, et transformer fractal pour augmenter le taux de compression
- Combiner les codeurs de source Shannon-Fano, Lempel Ziv Welch, et Run Length Encoding... pour obtention une meilleure codage d'information
- L'algorithme proposé peut aussi être implémenté sur d'autres espaces d'image couleur.

Bibliographie

- [1] Thierry Pun, "*Cours d'imagerie numérique*", CUI, Université de Genève, 1996-01-01.
- [2] Guz Almouzni, "*traitement numérique des images* ", ELSTI 2008-2009.
- [3] Majid Rabbani et Paul W Jones, *Digital Image Compression Techniques*, Bellingham, Spie Optical Engineering Presse, 1991, 240 p.
- [4] Mathieu Cord, 2016, Base de traitement des images.
- [5] ISIMA Vincent BARRA, traitement image.
- [6] Dr. Mvogo Ngoni Joseph "Principes généraux de codage entropique d'une source" Cours : Compression d'images
- [7] P. Steven. "Contribution a la Compression de Données". Thèse de Doctorat : Informatique. Montréal, 2001.
- [8] Chérif TAOUCHE " Implémentation d'un Environnement Parallèle pour la Compression d'Images à l'aide des Fractales" mémoire de Magister en Informatique, Université Mentouri Constantine 2005
- [9] BELADGHAM Mohammed, " construction d'une technique d'aide au diagnostic en imagerie médicale. Application à la compression d'images". UNIVERSITE ABOU-BEKR BELKAÏD – TLEMCEM
- [10] N.SADOUN, « Compression d'images fixes par une méthode hybride », thèse de magistère en électronique, UMMTO, 2005.
- [11] D. Saloman, "Data compression: the complet reference," London, Fourth Edition, Springer-Verlag London Limited, 2007.
- [12] F. Davoine, " Compression d'Images par Fractales Basée sur la Triangulation de Delaunay", Thèse l'INPG, Institut National Polytechnique de Grenoble, Décembre 1995.
- [13] Chérif TAOUCHE "Implémentation d'un Environnement Parallèle pour la Compression d'Images à l'aide des Fractales" mémoire magister en informatique, Université Mentouri Constantine
-