

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE



UNIVERSITE ECHAHID HAMMA LAKHDAR – EL OUED
FACULTE DES SCIENCES ET TECHNOLOGIE
Département D'électronique

Mémoire de fin d'études présenté
Pour l'obtention du diplôme de

Master ACADEMIQUE

Domaine : Sciences et techniques
Filière : Electronique
Spécialité : Télécommunications

Présenté par : ZEGHIB Maymouna
SEROUTI Abla

Intitulé

Compression des images avec curvelet

Soutenu le .. /06/2015

Devant le jury composé de :

M. **HIMA Abdelkader**
M. **MADJOURI Abdelkader**
M. **HETTIRI Messaoud**

MAA.
MAA.
MAA.

Directeur du mémoire
Président
Rapporteur

2014-2015

REMERCIEMENT

Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.

En second lieu, nous tenons à remercier notre encadreur M. HIMA ses précieux conseils et ses aides durant toute la période du travail.

Nous tenons à exprimer nos sincères remerciements à tous les professeurs qui nous ont enseigné et qui par leurs compétences nous ont soutenu dans la poursuite de nos études.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Merci à tous et à toutes.

Résumé :

Le développement remarquable dans le domaine des technologies de l'information et la diversité des applications multimédias dans les dernières années implique le développement des techniques de compression d'image plus efficaces dont l'objectif est d'améliorer la capacité de transmission et le stockage des données. Les travaux de recherche récents ont montré que les ondelettes classiques ne sont pas en mesure d'exploiter de façon optimale les régularités géométriques le long des contours et aux bords des objets.

Nous proposons dans ce travail une nouvelle méthode de compression basée sur des ondelettes géométriques, appelées Curvelets, associée à une technique de quantification scalaire logarithmique, afin d'améliorer les performances de compression, par rapport à celles des ondelettes classiques, du point de vue qualité de compression de l'image .

Cette méthode a donné des résultats encourageants avec des perspectives intéressantes.

Mots-Clés : Compression d'image, ondelettes géométriques, Curvelets, quantification scalaire logarithmique,

Abstract :

The remarkable development in the field of information technology and the diversity of multimedia applications in recent years imply the development of more efficient image compression techniques to improve the data transmission and storage capacity .

The recent researches showed that classical wavelets are not able to exploit optimally the geometric regularities along the contours and edges of objects .

We propose in this work, a new compression method based on geometric wavelets called Curvelets, associated with a logarithmic scalar quantization technique, to improve the compression performances compared to that of classical wavelets, in terms of image quality and compression

This method gave encouraging results with interesting perspectives .

Key-words: Image compression, geometric wavelets, Curvelets, scalar logarithmic quantization,

ملخص:

ان التطور الملحوظ في مجال تقنيات المعلومات و تنوع التطبيقات متعددة الوسائط في السنوات الاخيرة ،يستلزم تطوير تقنيات ضغط الصور اكثر فعالية لتحسين قدرات ارسال و تخزين المعلومات .

اظهرت الابحاث مؤخرا ان الموجات الكلاسيكية غير قادرة على الاستغلال الامثل للانتظام الهندسي على طول معالم و حواف الاجسام .

نقترح في هذا العمل طريقة ضغط جديدة تعتمد على موجات هندسية تسمى كورفلات متبوعة بتقنية تكميم سلمي لوغاريتمي ،و ذلك لتحسين اداء ضغط الصور مقارنة بأداء الموجات الكلاسيكية من حيث جودة الصورة .

اعطت الطريقة المقترحة نتائج مشجعة و افاق بحث اخرى

كلمات مفتاحيه: ضغط الصور،موجات هندسيه ,كورفلات، تكميم سلمي لوغاريتمي.

Sommaire

Introduction générale.....	1
Chapitre 1 : Différents types d'images	
1.1 Introduction.....	2
1.2 Image numérique.....	2
1.3 Caractéristiques d'une image numérique.....	2
1.3.1 Pixel.....	2
1.3.2 Image aux niveaux de Gris.....	3
1.3.3 Image en couleurs.....	3
1.4 Image Matricielle et Image Vectorielle.....	3
1.4.1 Image Matricielle.....	3
1.4.2 Image Vectorielle.....	5
1.5 Conclusion.....	7
Chapitre 2 : Généralités sur la compression	
2.1 Introduction.....	8
2.2 Compression d'images.....	8
2.2.1 Principe de compression.....	8
2.2.2 Performance de la compression.....	9
2.3 Méthodes de compression.....	11
2.3.1 Compression sans pertes.....	11
2.3.2 Compression avec pertes.....	13
2.4 La transformation.....	13
2.4.1 La transformation de Fourier discrète DFT.....	13
2.4.2 Transformée en cosinus discrète.....	14
2.4.3 Transformée en ondelettes.....	15
2.5 Transformée on ondelettes géométrique.....	17
2.5.1 Transformée de Radon.....	17
2.5.2 Transformée de Redgelets.....	18
2.5.3 Transformée de Curvelets.....	19
2.5.4 Transformée de Contourlets.....	20
2.6 Conclusion.....	21
Chapitre 3 : La compression par curvelets et quantification	
3.1 Introduction.....	22
3.2 L'ondelettes vers le curvelets.....	22
3.3 Principe de la transformée en curvelettes.....	23
3.4 Transformée en curvelets discrète.....	23
3.5 Propriétés des curvelets.....	26
3.6 Quantification.....	27
3.6.1 Quantification scalaire.....	27
3.7 Conclusion.....	36
Chapitre 4 : simulation et résultats	
4.1 Introduction.....	37
4.2 Algorithme de compression.....	37
4.3 Présentation des résultats.....	37
4.4 Analyse et Discussions.....	43
4.5 Conclusion.....	43
Conclusion générale.....	44
Bibliographie.....	45

Liste des figures

Figure 2.1 : Schéma fonctionnel de la compression.....	8
Figure 2.2 : Transformée de Radon.....	18
Figure 2.3 : Représentations d'image par Ondelettes (a) et Contourlets(b).....	21
Figure 3.1 : Fonctions d'approximation des ondelettes et des curvelettes.....	22
Figure 3.2 : Principe de la transformée en curvelettes.....	25
Figure 3.3 : Décomposition en curvelettes d'une image en 2 niveaux.....	26
Figure 3.4 : Exemple d'un quantificateur scalaire uniforme avec cinq niveaux $L = 5$	29
Figure 3.5 : Quantificateur midtread , Quantificateur midrize.....	29
Figure 3.6 : Illustration des quantificateur scalaire uniforme (a) et quantificateur scalaire uniforme a zone morte (b, ici $T = 2\Delta$).....	30
Figure 3.7 : Quantificateur Scalaire non uniforme.....	31
Figure 3.8 : Processus de quantification non uniforme par companding.....	32
Figure 3.9 : Exemples des fonctions de compression et d'expansion.....	33
Figure 3.10 : Fonctions A loi de puissance pour ($P=0.2$, $P=0.4$, $P=0.6$).....	34
Figure 3.11 : La fonction de compression MU-LAW ($\mu=1, 10, 255$).....	35
Figure 3.12 : (A)-étapes de quantification par companding, (B)-quantification MU-LAW a 8 niveaux ($MU=25$).....	36
Figure 4.1 : Méthode de compression et décompression proposée.....	37
Figure 4.2 : Evolution du PSNR pour la méthode Mu-law.....	39
Figure 4.3 : Evolution du PSNR pour la méthode lloyd-Max.....	39
Figure 4.6 : résultats obtenues avec Mu-law et sans Mu-law.....	40
Figure 4.7 : résultats obtenues avec lloyd-Max et sans lloyd-Max.....	41
Figure 4.8 : Comparaison entre deux méthodes (Mu-law et lloyd-Max) pour l'image Barbara et Peppers.....	42

Liste des tableaux

Tableaux 4-1 Résultats obtenus de méthode Mu-law (PSNR).....	38
Tableaux 4-2 Résultats obtenus de méthode lloyd-Max (PSNR).....	38

Lexique d'abréviations

AC	Alternating Current
BMP	Bit map picture
bpp	bits par pixel.
CR	Compression Ratio
DCT	Discrete Cosine Transform
DXF	Drawing eXchange Format
DC	Direct Current
DWT	Discrète Wavelet Transform
DQs	la distorsion La Quantification Scalaire
CGM	Computer Graphics Metafile
EPS	Postscript / Encapsulated Postscript
EDIGEO	Échange de données informatisées dans le domaine de l'information géographique
EZW	Embedded Zerotree Wavelet
FFT2D	Transformée de Fourier rapide à deux dimensions
IFFTD	Transformée de Fourier rapide inverse à Une dimension.
GIF	Graphic Interchange Format
HPGL	Hewlett-Packard Graphics Language Hewlett-Packard Graphics La
IGES	Initial Graphics Exchange Specification
JPEG	Joint Photographic Expert Group.
LZW	Lempel Zip Welch.
$L^2(R)$	L'ensemble des carrés sommable
MSE	Erreur quadratique moyenne
PDF	Fonction de distribution de la probabilité.
PSNR	Rapport signal sur bruit crête.
PCX	Picture Exchange Image Bitmap Zsoft
PICT	Macintosh Picture
Q(x)	La fonction de quantification
QS	La Quantification Scalaire
R,V,B	(R) Rouge, (V) Vert, (B) Bleu.
RLE	Codage de longueur de séquence
Rf	la fonction ridgelet
SPIHT	Set Partitioning In Hierarchical Trees.
T_c	Taux de compression.
TOD	Transformation en ondelettes discrètes
TOC	Transformation en ondelettes continue
TIFF	Tagged image (file) format
USFFT	Unequally spaced FFT

Introduction général

Les évolutions récentes des technologies de l'information et systèmes de communication, ainsi que la diversification des applications et fonctionnalités du multimédia dans plusieurs domaines (télémédecine, télédétection... 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 d'information. 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. Parmi les données les plus volumineuses, on compte les 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.

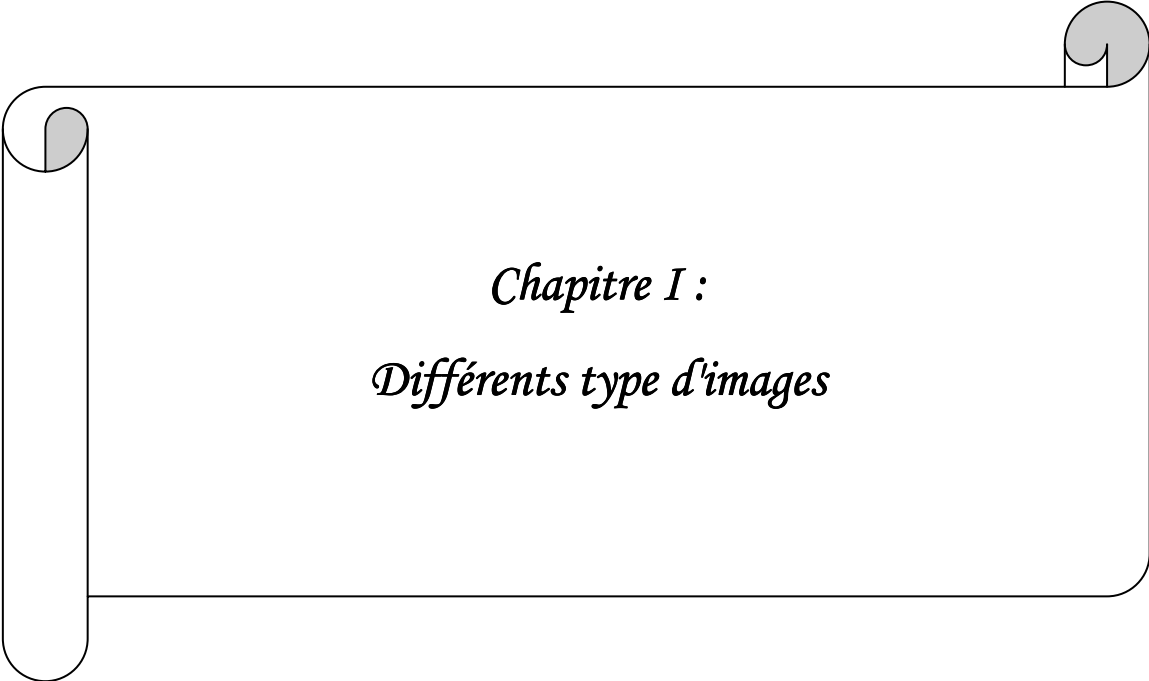
Le transformée en ondelettes discret est la technique de transformation la plus populaire adoptée pour la compression d'image.

Malgré leur popularité, les ondelettes classiques ne sont pas en mesure de détecter les régularités géométriques, et l'utilisation de véritables représentations géométriques semble être plus optimale pour un large éventail de classes d'images.

Ce travail de mémoire s'inscrit principalement dans l'un des aspects de ce domaine : la compression d'images fixes par un nouveau type d'ondelettes géométriques dite les Curvelets.

Le premier chapitre est réservé à des généralités sur les différentes images et leurs définitions et le deuxième chapitre permet de présenter des concepts généraux de compression d'images fixes en plus le principe de compression, ainsi qu'une brève présentation des ondelettes classiques. Le troisième chapitre sur les curvelets, où on expose les propriétés principales et les techniques de son transformé. Dans le quatrième chapitre on présente les techniques de quantification les plus utilisés dans la compression d'images en déduisant une nouvelle technique de quantification basé sur les caractéristiques des coefficients de transformé en curvelets. Le cinquième chapitre est réservé pour l'exposition et la discussion de résultats obtenus par les schémas proposés.

Enfin, une conclusion générale sur les travaux menés au cours de ce mémoire sera exposée, et les différentes perspectives seront soulevées.



Chapitre I :
Différents type d'images

1.1 Introduction :

L'image est une représentation d'une personne ou d'un objet par la peinture, la sculpture le dessin, la photographie, le film,...etc. C'est aussi un ensemble structuré d'informations qui, après affichage sur l'écran, ont une signification pour l'œil humain.

Nous présentons dans ce chapitre la définition et les types de l'image numérique.

1.2 Image Numérique :

L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter [1] .

1.3 Caractéristiques d'une image numérique :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants:

1.3.1 Pixel :

Le pixel est la contraction de l'expression anglaise " picture elements". Le pixel, étant le plus petit point de l'image, C'est une entité calculable qui peut recevoir une structure et une quantification. Si le bit est la plus petite unité d'information que peut traiter un ordinateur, le pixel est le plus petit élément que peuvent manipuler les matériels et logiciels d'affichage ou d'impression.

Dans une image couleur (R.V.B.), un pixel peut être représenté sur trois octets : un octet pour chacune des couleurs : rouge (R), vert (V) et bleu (B) [2].

1.3.2 Images aux Niveaux de Gris :

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaire.

Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise entre 0 et 255. Par convention, la valeur zéro représente le noir (intensité lumineuse null) et la valeur 255 le blanc (intensité lumineuse maximale).

Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la "couleur" de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux

1.3.3 Images en Couleurs :

Elle consiste à utiliser 24 bits pour chaque point de l'image. Huit bits sont employés pour décrire la composante rouge (R), huit bits pour la composante verte (V) et huit bits pour la composante bleu (B). Il est ainsi possible de représenter environ 16,7 millions de couleurs différentes simultanément.

1.4 Images Matricielle et Images Vectorielles :

Les images appartiennent à deux grandes familles : Matricielle et vectorielle. Alors qu'une image vectorielle est décrite à l'aide de courbes et d'équations mathématiques, une image Matricielle est constituée de pixels et se réduit donc à une matrice de points. Si les images vectorielles peuvent être manipulées avec beaucoup de facilité, les modifications de taille.

1.4.1 Image Matricielle:

Les images affichées sur l'écran d'un PC sont des images matricielles, encore appelées bitmap (image bit), chaque image est en réalité une matrice de pixels.

a. Les formats d'images Matricielle :

- Le format BMP origine Microsoft pour Windows 3.x de plus en plus répandu
Contraction de Bitmap. Constitue un format de fichier image, pixel par pixel et sans compression
- Le format GIF (Graphic Interchange Format) : Créé par CompuServe, utilise aussi le codage RVB mais le format GIF n'utilise pas toutes les 16 millions de couleurs. Il prendles 256 couleurs les plus courantes pour réaliser l'image au format GIF. Cela permet unebonne algorithme de compression LZW et un affichage rapide de l'image.
- Le format JPG ou JPEG (Joint Photographique Experts Group): Créé par un consortium industriel, ce format tres utilise sur Internet, permet d'afficher les images en mode 16millions de couleurs et donc c'est le mode sans perte.Le codage d'un pixel (point de l'ffichage) 3x1, byte (RVB) soit : $(0 < R < 255)$, $(0 < V < 255)$, $(0 < B < 255)$. Le format JPG peutaussi être utilisé pour compacter les images, il utilise un algorithme de compression qui réduit énormément la taille des images.
- Le format PCX (Picture Exchange Image Bitmap Zsoft) : A été crée à l'origine par ZSoft pour un programme de dessin (paint bruch). Permet de traiter des images 8 bits et non adapté aux images 16, 24, 32 bit, il utilise un algorithme de compression : RLC
- Le format TIFF (Tagged Image File Format) origine Aldus et Microsoft, pour les images scannées, la compression du format TIFF est effectue sans perte de données il utilise plusieurs algorithmes de compression : RLC, LZW
- Le format TGA (Truevision Targa) origine Truevision, algorithme de compression RLC.
- Le format FAX utilisé pour la transmission de documents (télécopie), codage binaire, application d'un codage RLC puis d'un codage de Huffman

b. Avantages et inconvénients :**Avantages :**

- Le mode de codage des images bitmap (24 bits, codage RBG) les rend adaptées au fonctionnement des principaux périphériques, notamment les contrôleurs d'écran "true colors"
- Adaptable aux images complexes
- Le mode de codage point par point étant relativement universel, une fois cette opération effectuée, le transcodage demande des calculs répétitifs mais relativement simples: la compatibilité est aisée entre les différents formats.
- La structure du fichier est telle que des dégradations minimales des données -mais non pas de l'en tête- ne le rendent pas nécessairement inutilisable.

Inconvénients :

- Elles supportent mal les opérations de redimensionnement, réduction ou agrandissement. Les deux opérations se traduisent par une perte d'information.
- Encombrement important.
- Les images bitmap ont une résolution fixe: aussi la qualité maximale sur un périphérique d'affichage ou d'impression donné rend-elle nécessaire de travailler, dans la majorité des cas, dans la résolution native de ce périphérique.

1.4.2 Image vectorielle:

La description vectorielle d'une image consiste en une description géométrique: le document numérisé prend donc la forme d'une suite de formules mathématiques décrivant les formes élémentaires constituant l'image (carrés, rectangles, ellipses, cercles, courbes, etc.). Chaque forme élémentaire constitue un objet et se voit assigné un certain nombre d'attributs tels que la couleur, la transparence, l'épaisseur du trait, le type de trait (pointillé, etc.)

a. Les formats d'images vectorielles:

- Le format EPS (Postscript / Encapsulated Postscript) : Le Postscript est un langage de description de page conçu pour imprimer des documents indépendamment du périphérique utilisé. Il contient toutes les commandes nécessaires pour dessiner l'image sauvegardée.
- Le format CGM (Computer Graphics Metafile) : Les fichiers graphiques CGM restent un moyen privilégié d'échange de dessins vectoriels entre applications, mais il ne traite pas les images bitmaps.
- Le format PICT (Macintosh Picture) déjà mentionné plus haut
- Le format DXF (Drawing eXchange Format) origine Autocad, standard en CAO; adapté au travail en 2D; connu et très répandu
- Le format HPGL (Hewlett-Packard Graphics Language) origine Hewlett Packard; format répandu pour les périphériques de traçage
- Le format IGES (Initial Graphics Exchange Specification) permet l'échange de données en CAO; environnement des stations de travail
- Le format EDIGEO (Échange de données informatisées dans le domaine de l'information géographique) adapté à la cartographie; normalisé par l'AFNOR

b. Avantages et inconvénients :**Avantages :**

- Adaptable à chaque périphérique (car ajustable, coordonnées relatives, adaptable aux drivers intelligents,...).
- Stockage moins encombrant (la taille ne dépend que de la complexité du document lui-même).
- Adapté aux objets mathématiques et leur extension (voiture, avion,...).

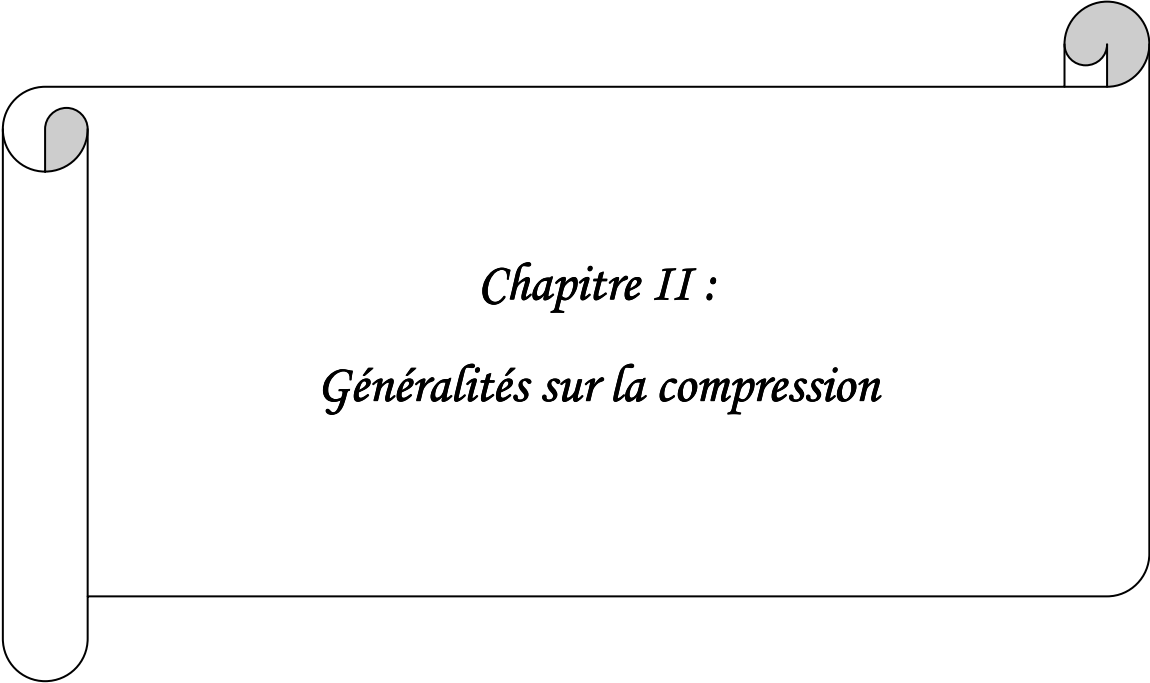
Inconvénients :

- Mal adapté aux images quelconques.
- La difficulté de traitement d'une partie de l'image

1.5 Conclusion :

Dans ce chapitre, on a essayé de faire un récapitulatif sur les notions élémentaires de traitement d'image. Nous avons présenté les notions de l'image numérique et ses différents types formats (Bitmap, jpeg,...etc.) ainsi que les objectifs de la compression et les différentes solutions apportés.

Le chapitre suivant est consacré aux différentes méthodes de compression et transformé



Chapitre II :
Généralités sur la compression

2.1 Introduction :

La compression d'une image numérique permet de réduire le nombre de bits qu'elle occupe. Les deux principes qui interviennent pour atteindre cet objectif sont la réduction de la redondance (n'introduisant pas de perte), et la représentation approximative de l'information contenue dans l'image (introduisant une perte).

2.2 Compression d'images :

2.2.1 Principe de compression :

Le schéma fonctionnel de la compression est présenté dans la figure ci-dessous [3] :

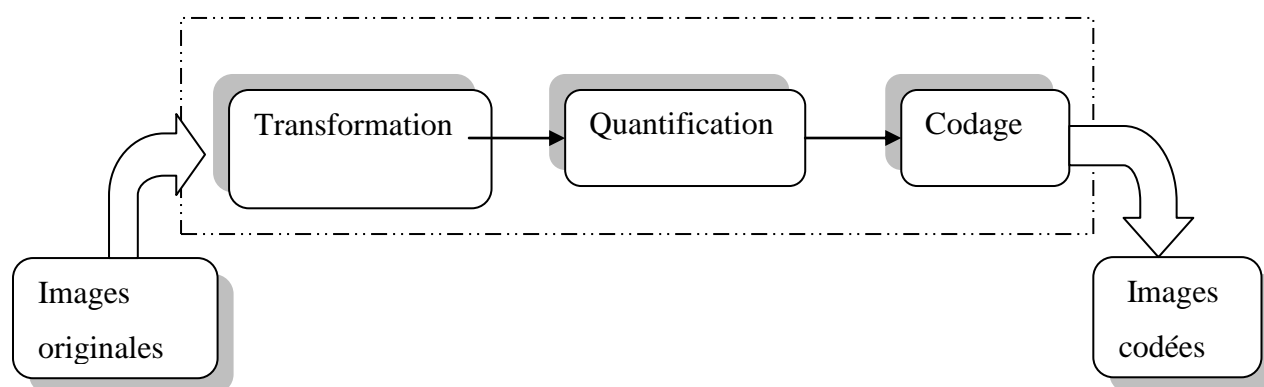


Figure 2.1 : Schéma fonctionnel de la compression

a. Transformation :

La dépendance existante entre chacun des pixels et ses voisins (la luminosité varie très peu d'un pixel à un pixel voisin) traduit une corrélation très forte sur l'image. On essaie donc de tirer partie de cette corrélation, pour réduire le volume d'information en effectuant une opération de transformation des pixels.

La transformation consiste à transformer les pixels initiaux en un ensemble de coefficients plus corrélés, c'est une opération réversible.

b. Quantification :

La quantification des coefficients a pour but de réduire le nombre de bits nécessaires pour leurs représentations. 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, elle peut être scalaire ou vectorielle.

c. Codage :

Les coefficients quantifiés forment un train binaire le plus réduit possible. Cette opération ne provoque aucune distorsion ou perte à l'information contenue dans l'image.

2.2.2 Performance de la compression :**a. Taux de compression :**

Etant donné que l'objectif de la compression est de minimiser la quantité de l'information indispensable pour la représentation d'une image, on définit le paramètre taux de compression CR (Compression Ratio) :

$$CR = \frac{\text{Nombre de bits avant la compression}}{\text{Nombre de bits après la compression}} \quad (2.1)$$

A partir de ce quotient on définit aussi le taux de compression en pourcent par le rapport:

$$T_c = \left(1 - \frac{1}{CR}\right) \times 100 \quad (2.2)$$

On peut aussi définir le taux de bits par le rapport suivant :

$$\text{Taux de bits} = \frac{\text{Nombre d'octets après la compression}}{\text{Nombre d'octets avant la compression}} \times 8 \quad (\text{bpp}) \quad (2.3)$$

Ainsi en augmentant le taux de compression (en diminuant le taux de bits), on gagne en plus l'espace mémoire mais on dégrade la qualité de l'image après sa restitution ce qui risque de perdre totalement l'information. Par ce fait on introduit la notion de la «distorsion».

b. Mesures de distorsion :

La distorsion 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 MSE (Mean Square Error). Cette grandeur est définie par la moyenne des écarts aux carrés entre le pixel (i, j) de l'image originale est celui de l'image reconstruite suivant l'équation suivante :

$$MSE = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} [I(i,j) - \tilde{I}(i,j)]^2}{N \times N} \quad (2.4)$$

Avec :

$N \times N$: Taille de l'image.

$I(i, j)$: Pixel de l'image originale.

$\tilde{I}(i, j)$: Pixel de l'image résultante.

On référence cette erreur par rapport à la dynamique de l'image en décibels. On obtient un rapport signal sur bruit crête pour une image (dans notre cas $R=8$), appelé PSNR (Peak Signal to Noise Ratio) qui se traduit par l'équation suivante :

$$PSNR = 10 \log_{10} \frac{[2^R - 1]^2}{MSE} \quad (2.5)$$

Où: $2^R - 1$ représente la dynamique de l'image.

Lorsque la reconstruction est parfaite, la MSE est nulle et le PSNR tend vers l'infini. Ces mesures de distorsion sont objectives et simples à calculer. Cependant elles ne rendent pas compte de la qualité visuelle engendrée par la compression.

c. Entropie :

L'entropie est une grandeur qui caractérise la quantité d'information que contient une image. Par exemple une image dont tous les pixels ont la même valeur contient très peu d'information car elle est extrêmement redondante, son entropie est faible. Par contre, une image dont tous les pixels ont une valeur aléatoire contient beaucoup d'information, son entropie est forte.

L'entropie d'ordre zéro d'une Image en niveau de gris dont les pixels sont codés sur R bits peut s'exprimer par :

$$H_0 = - \sum_{k=0}^{2^R-1} P(k) \log[P(k)] \quad \text{bpp} \quad (2.6)$$

$p(k)$: probabilité d'apparition de la valeur k .

L'entropie H_0 d'une image originale fournit le débit minimal qu'il est possible d'atteindre par compression pixel par pixel sans dégrader l'image, et par là même un taux de compression sans perte maximal [4].

2.3 Méthodes de compression :

2.3.1 Compression sans pertes :

Les méthodes sans pertes peuvent s'appliquer dans le domaine spatial, ou plus difficilement, dans le domaine des fréquences. En général, nous emploierons le terme symbole au lieu de pixel ou bien coefficient du domaine des fréquences, et la notion de séquence de symboles pour une image ou une partie d'image. Les termes réversible et irréversible seront employés respectivement comme synonymes de sans et avec perte.

a. Méthode par plages (Runlength encoding) :

Le codage RLE est très répandu dans les algorithmes de compression d'images. Ce codage est beaucoup moins efficace que l'autre, il est facile à implémenter, rapide à exécuter et représente une bonne alternative. Le codage RLE travaille sur la réduction de la taille physique des chaînes de caractères répétitives. Ces répétitions nomment en anglais un «run».

Le premier octet représente le nombre de répétition; en pratique cette valeur est comprise le plus souvent entre 1 et 128 ou 256. L'octet suivant représente l'octet à itérer [5].

L'efficacité du RLE dépend du nombre de transitions des niveaux de gris de l'image à coder. Plus le nombre de niveaux de gris est faible, plus le codage est meilleur. Le taux de compression atteint dépend de la complexité de l'image.

b. Codeurs entropiques :

Le but du codage entropique est de s'approcher le plus possible de l'entropie H_0 de la séquence de symboles, en affectant les codes les plus courts possibles aux symboles de probabilité élevée et vice versa. Ces systèmes sont conçus de manière à ce que les codes résultants, bien que de longueur variable, puissent être décodés de façon unique.

b.1 Le codage de HUFFMAN :

Cette méthode repose sur la construction d'un arbre basé sur les probabilités d'apparition des éléments, on représente chaque élément i par une séquence de bits de longueur inversement proportionnelle à la probabilité d'apparition $p(i)$, ce code est optimal pour les codes à nombre de bits entiers [6].

b.2 Codage arithmétique :

Le codage arithmétique est un codage récent utilisant un modèle statistique, tout comme le codeur de Huffman. Contrairement à ce dernier, il produit un code pour une séquence de symboles tout entier, et non pas un code par symbole.

Chaque nouveau symbole lu modifie de façon incrémentale le code de sortie. Ce code de sortie est un nombre à virgule flottante compris entre 0 et 1, dont le nombre de chiffres après la virgule correspond au nombre de symboles. Contrairement à Huffman, il n'est pas obligatoire que chaque code ait un nombre entier de bits [7].

2.3.2 Compression avec pertes :

Tout comme les méthodes sans pertes, les méthodes avec pertes peuvent avoir pour domaine de travail le plan spatial ou le plan des fréquences.

a. Quantification scalaire :

La quantification scalaire consiste à remplacer un nombre très grand de symboles par un nombre de codes. C'est une opération irréversible très largement employée en compression. Par exemple le fait d'arrondir des nombres flottants à l'entier le plus proche est une quantification scalaire.

b. Quantification vectorielle :

La quantification vectorielle découpe la matrice de symboles (pixels ou coefficients du plan des fréquences) en petits blocs. Chaque bloc est codé par un indice dans un dictionnaire de blocs types. Le codage consiste à repérer dans le dictionnaire des mots codes disponible celui qui ressemble le plus au bloc d'entrée. Le critère consiste en général à minimiser l'erreur quadratique. Le décodage consiste simplement à remplacer l'indice par le mot code correspondant dans le dictionnaire. La création du dictionnaire de bloc types est une étape préalable et cruciale.

2.4 La transformation :

2.4.1 La transformation de Fourier discrète DFT :

La transformation de Fourier discrète s'introduit quand il s'agit de calculer la transformée de Fourier d'une fonction à l'aide d'un ordinateur numérique. En effet, un tel opérateur ne peut traiter que des nombres et de plus en quantité par la taille de la mémoire.

Soit un signal discret $x(n)$, la DFT et la DFTI sont données par les formules suivantes [8]:

$$X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{nk}{N}} \quad (2.7)$$

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{j2\pi \frac{nk}{N}} \quad (2.8)$$

2.4.2 Transformée en cosinus discrète:

Cette transformée est introduite en 1974 par Ahmed et al [9]. Elle a fait l'objet de beaucoup d'études et d'applications de la compression des signaux. Elle est en fait une variante de la transformée de Fourier discrète qui transforme un signal réel en un signal réel (la notion de phase n'apparaît pas). Elle prend un ensemble de points d'un domaine spatial et le transforme en une représentation équivalente dans le domaine fréquentiel.

La DCT-1D [9] d'un vecteur x de longueur N donne un autre vecteur X de N coefficients, selon la définition :

$$X(u) = \sqrt{\frac{2}{N}} C(u) \sum_{n=0}^{N-1} x(n) \cos \left[\left(\frac{2n+1}{2N} \right) u \pi \right] \quad (2.9)$$

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{pour } u = 0 \\ 1 & \text{pour } u = 1, 2, \dots, N-1 \end{cases}$$

Le premier coefficient $X(0)$ est appelé composante continue ou composante DC (Direct Current), Il possède la propriété d'être proportionnel à la valeur moyenne du vecteur avant transformation. Le reste des coefficients sont référés comme les coefficients AC (Alternating Current). Sur les coefficients ACs, les énergies sont groupées en basse, moyenne et haute fréquence. Les coefficients sont des nombres réels même si les données d'entrée sont des entiers. Similairement, les coefficients peuvent être positifs ou négatifs même si les données d'entrée sont positives seulement. Pour alléger ces calculs, plusieurs algorithmes de calculs rapides de la DCT ont été proposés afin de diminuer le nombre d'opérations nécessaires en passant par la FFT (Fast Fourier Transform) [10] [11].

La transformée inverse est donnée par :

$$x(n) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} C(u) X(u) \cos \left[\left(\frac{2n+1}{2N} \right) u\pi \right] \quad (2.10)$$

La DCT monodimensionnelle peut être utilisée pour compresser les données monodimensionnelles, telles que les signaux audio. Pour les images, la DCT-2D est utilisée. Elle est appliquée à des portions (blocs) de l'image, généralement, de taille 8×8 ou 16×16 pixels (à cause des calculs effectués sur les matrices). En exploitant la propriété de séparabilité de la DCT, on peut calculer la DCT-2D en appliquant la DCT-1D dans les deux directions, les lignes puis les colonnes. Des algorithmes rapides [12] [13] permettent d'effectuer cette transformation avec un nombre réduit de multiplications. Son expression d'analyse est définie pour une matrice x de taille $M \times N$ par :

$$X(u, v) = \frac{4}{M \times N} C(u) C(v) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m, n) \cos \left[\left(\frac{2m+1}{2M} \right) u\pi \right] \cos \left[\left(\frac{2n+1}{2N} \right) v\pi \right] \quad (2.11)$$

$$\text{Avec : } C(k) = \begin{cases} \frac{1}{\sqrt{2}} & \text{pour } k = 0 \\ 1 & \text{pour } k = 1, 2, \dots, N-1 \end{cases}$$

La DCT a pour effet de concentrer l'information ou l'énergie du signal en très peu de coefficients fréquentiels correspondants aux basses fréquences, et que les autres coefficients de hautes fréquences ont des valeurs très faibles. C'est cette caractéristique qui sera exploitée par la compression.

La transformée en cosinus discrète inverse qui permet de revenir au domaine spatial à partir des coefficients DCT est donnée par l'expression suivante :

$$x(m, n) = \frac{4}{M \times N} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u) C(v) X(u, v) \cos \left[\left(\frac{2m+1}{2M} \right) u\pi \right] \cos \left[\left(\frac{2n+1}{2N} \right) v\pi \right] \quad (2.12)$$

2.4.3 Transformée en ondelettes :

Les ondelettes sont des familles de fonctions déduites d'une même fonction par des opérations de translation dans le temps 'b' et par dilatation en fréquence d'un facteur

d'échelle 'a'. Elles ont trouvé, d'une part la puissance de leur théorie, des applications dans de nombreux domaines aussi variés que les mathématiques (analyse, probabilités, fractales), le traitement du signal et d'images (compression, astronomie, sismique), la physique (mécanique quantique, turbulence). En effet, cet outil permet la représentation de fonctions de $L^2(\mathbb{R})$, dans une base bien localisée en temps et en fréquence, offrant les avantages de l'analyse de Fourier et s'affranchissant des inconvénients du manque de localisation de cette dernière. Les ondelettes sont normalisées suivant l'équation :

$$\psi_{a,b}(t) = \frac{1}{|a|^{1/2}} \psi\left(\frac{t-b}{a}\right) \quad \text{Avec } a \in \mathbb{R}^{+*} \text{ et } b \in \mathbb{R} \quad (2.13)$$

a. Transformation en ondelettes continue (TOC) :

La transformée en ondelettes [14] d'une fonction $f \in L^2(\mathbb{R})$ est une représentation de celle-ci sur une base d'ondelettes choisie. Elle est dite continue car les paramètres de dilatation et de translation prennent des valeurs continues. Elle est définie par l'équation:

$$W(a,b) = TOC(f)(a,b) = \langle \Psi_{a,b}, f \rangle = |a|^{-1/2} \int_{-\infty}^{+\infty} f(x) \overline{\Psi\left(\frac{x-b}{a}\right)} dx \quad (2.14)$$

Où $\overline{\Psi}$: Représente le conjugué complexe de Ψ .

b. Transformation en ondelettes discrètes (TOD) :

La discrétisation de la DWT implique que les opérateurs de dilatation et de translation (a et b) prennent des valeurs discrètes suivant l'expression suivante :

$$(a,b) = (a_0^j, kb_0 a_0^j) \quad \text{Où } (j,k) \in \mathbb{Z}^2 \quad (2.15)$$

Pour former ce qu'on appelle un réseau discret ; avec $a_0 > 1$ (le pas de dilatation) et $b_0 \neq 0$ (le pas de translation).

Ainsi, la nouvelle famille d'ondelettes s'établit par :

$$\psi_{j,k}(t) = a_0^{-j/2} \cdot \psi(a_0^{-j}t - kb_0) \quad (2.16)$$

Et la DWT s'établit comme suit :

$$W(a,b) = \langle \psi_{j,k}, f \rangle = a_0^{-j/2} \int_{-\infty}^{+\infty} \psi(a_0^{-j}t - kb_0) f(t) dt \quad (2.17)$$

En 1985, Meyer [15] découvre qu'en choisissant des opérateurs de dilatation et de translation dyadiques ($a_0 = 2$ et $b_0 = 1$) [15], la réduction de redondance de la représentation en ondelettes est maximale. En plus, les $\psi_{j,k}$ forment une base Hilbertienne de $L^2(\mathbb{R})$. Avec $a = 2^j$ et $b = k2^j = ka$ on obtient.

$$\left\{ \psi_{j,k} \right\}_{j,k \in \mathbb{Z}^2} = \left\{ 2^{-j/2} \psi(2^{-j}t - k) \right\}_{j,k \in \mathbb{Z}^2} \quad (2.18)$$

Un tel choix de (a et b) permet de construire des algorithmes numériquement stables pour la reconstitution de la fonction $f(x)$ grâce à l'expression suivante :

$$f(x) = \sum_{j,k \in \mathbb{Z}} \langle f(x), \psi_{j,k}(x) \rangle \psi_{j,k}(x) \quad (2.19)$$

2.5 Transformé en ondelette géométrique :

2.5.1 Transformée de Radon :

La transformée de Radon [16,17] consiste à projeter l'image sur un certain nombre d'orientations en intégrant l'image le long de la direction orthogonale à la projection (Figure 2.2), puis à réaliser la transformée de Fourier de ces projections. La reconstruction s'obtient en

plaçant, pour chaque orientation de projection choisie, les coefficients de Fourier obtenus le long de cette même orientation, dans le domaine fréquentiel on obtient l'image reconstruite en effectuant ensuite une transformée de Fourier 2D inverse. La reconstruction parfaite pour cette transformée continue s'obtient pour un nombre de projections infini, parcourant l'ensemble des orientations possibles. La transformée de Hough est un cas particulier de transformée de Radon pour une image à valeurs binaire, et s'utilise principalement pour la reconnaissance de formes.

La transformée de Radon est très utilisée en tomographie, où les données capturées correspondent précisément à des projections du contenu de l'objet dont on cherche à obtenir une image. Une reconstruction approximative de l'image recherchée, d'autant plus précise que le nombre de directions de projection est élevé, est obtenue par transformée de Radon inverse. Notons que pour une image discrète carrée dont la taille est un nombre premier, la discrétisation proposée peut s'appliquer pour obtenir une transformée de Radon discrète à reconstruction parfaite peu redondante. Une approximation discrète rapide et inversible de la transformée de Radon.

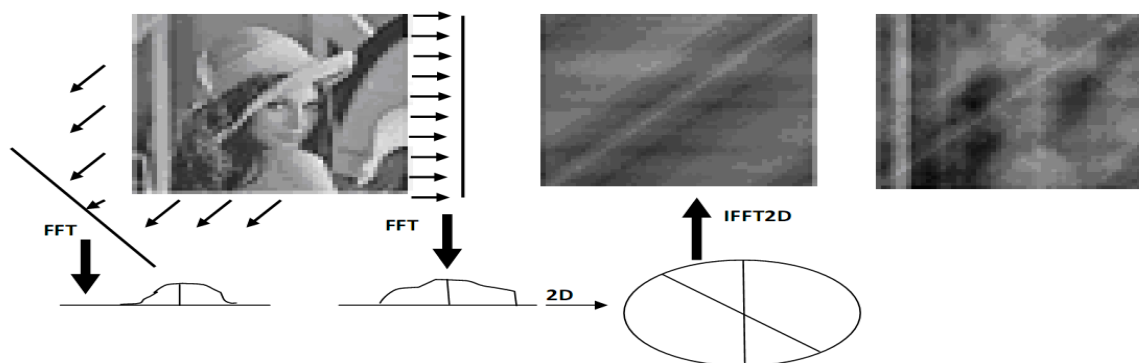


Figure 2.2 : Transformée de Radon

2.5.2 Transformée les Ridgelets:

L'analyse en ridgelets peut se construire comme une analyse en ondelettes dans le domaine de Radon. La transformée de Radon 2D d'un objet f est la collection des intégrales sur une ligne indexées par $(\theta, t) \in [0, 2\pi[\times R$ donnée par:

$$Rf(\theta, t) = \int_{\mathbb{R}^2} f(x_1, x_2) \delta(x_1 \cos \theta + x_2 \sin \theta - t) dx_1 dx_2 \quad (2.20)$$

où δ est la distribution de Dirac. Ensuite, la transformée en ridgelets est précisément l'application d'une transformée en ondelettes 1D sur les lignes de la transformée de Radon, où la variable angulaire θ est constante et t varie. Pour chaque échelle $s > 0$, chaque position $t \in \mathbb{R}$ et chaque angle $\theta \in [0, 2\pi[$ la fonction ridgelet 2D $\psi_{s,t,\theta}$ est définie à partir d'une fonction ondelette 1D ψ comme:

$$\psi_{s,t,\theta}(x_1, x_2) = s^{-1/2} \cdot \psi((x_1 \cos \theta + x_2 \sin \theta - t)/s) \quad (2.21)$$

Une fonction ridgelet est constante le long des lignes $x_1 \cos \theta + x_2 \sin \theta = \text{const}$. Perpendiculairement à ces lignes, on a une fonction ondelette.

Ainsi, une stratégie élémentaire pour calculer la transformée en ridgelets est de calculer tout d'abord la transformée de Radon $Rf(t, \theta)$, puis d'appliquer une transformée en ondelettes unidimensionnelle sur les lignes $Rf(., \theta)$.

2.5.3 Transformée de Curvelet :

La transformée en curvelet [16,18] se dérive des ridgelets multi échelles. Une première décomposition permet d'obtenir une analyse multi échelles en sous-bandes centrées sur les couronnes de fréquences $f \in [0, 1/2^{2s}] / [0, 1/2^{2^{s+2}}]$ ou $s \in n$ où $s \in \mathbb{N}$ représente l'échelle. Notons que cette découpe de l'espace fréquentiel n'est pas classique. Ces sous-bandes sont en suites analysées par une transformée en orthoridgelets sur des blocs de tailles $2^s * 2^s$. Les blocs d'analyse sont alors des éléments rectilignes de taille $2^s * 2^{2s}$. Ces éléments suivent donc une loi de changement d'échelle parabolique $l = w^2$. L'ondelette utilisée pour construire les orthoridgelets est l'ondelette de Meyer tandis que la fonction d'échelle de Le marié est utilisée pour la représentation des basses fréquences. Une construction différente et plus générale, reposant sur la théorie des frames. La frame d'analyse est construite directement à partir d'une fonction mère C bidimensionnelle de haute fréquence selon l'un des axes et de

basse fréquence selon l'autre (typiquement le produit tensoriel d'une fonction d'ondelette et d'une fonction d'échelle). La famille de curvelets $(Cl, n, \theta)_{l \in \mathbb{N}, n \in \mathbb{Z}, \theta \in 2\pi/2^l \mathbb{Z}/2^l \mathbb{Z}}$ correspondante est alors donnée par :

$$C_{l,n,\theta(t)} = 2^{3l/2} C(D_l R_\theta t - n) \quad (2.22)$$

$D_l = \begin{pmatrix} 2^{2l} & 0 \\ 0 & 2^l \end{pmatrix}$ Est la matrice de sous-échantillonnage du changement d'échelle

Notons au passage que $2^{3l/2}$ correspond à la racine carrée de son déterminant), R_θ est la matrice de rotation d'angle θ et n indique la position de la curvelet. Cette transformée a été utilisée avec succès dans le cadre du débruitage.

2.5.4 Transformée les Contourlets :

Inspirés par les curvelets, Do et Vetterli [19] ont développé la transformée en contourlets qui obéit à un nouveau schéma de décomposition multi résolution qui permet de représenter l'image comme un ensemble compacte de niveaux de résolution spatiale et directionnelle (orientées). Contrairement au curvelets, la transformée en contourlets est conçue directement dans le domaine discret. L'esprit de la méthode est cependant similaire à celui des curvelets car il s'agit d'une base fixe avec une grande sélectivité directionnelle. Le but principal de la construction contourlets est de fournir une représentation éparsée des données aussi bien aux résolutions spatiales que fréquentielles. L'appellation «contourlet» vient du fait que cette nouvelle décomposition est un frame composé de segments de contour.

Les ondelettes 2D, résultant du produit tensoriel de deux transformées en ondelettes 1D, fournissent un nombre limité et fixe d'orientations et peuvent capturer uniquement les discontinuités ponctuelles sans prendre en compte la régularité des contours. Les contourlets ont été développées comme remède à cette inefficacité des ondelettes. En effet, la transformée en contourlets possède non seulement les principales propriétés des ondelettes (à savoir, sa multi résolution, sa bonne localisation en espace et en fréquence, ainsi que sa décimation critique) mais offre aussi une grande sélectivité directionnelle. Pour mettre en évidence la

différence entre la transformée en ondelettes et la transformée en contourlets, la Figure 2.3 montre quelques images de base de ces deux transformées.

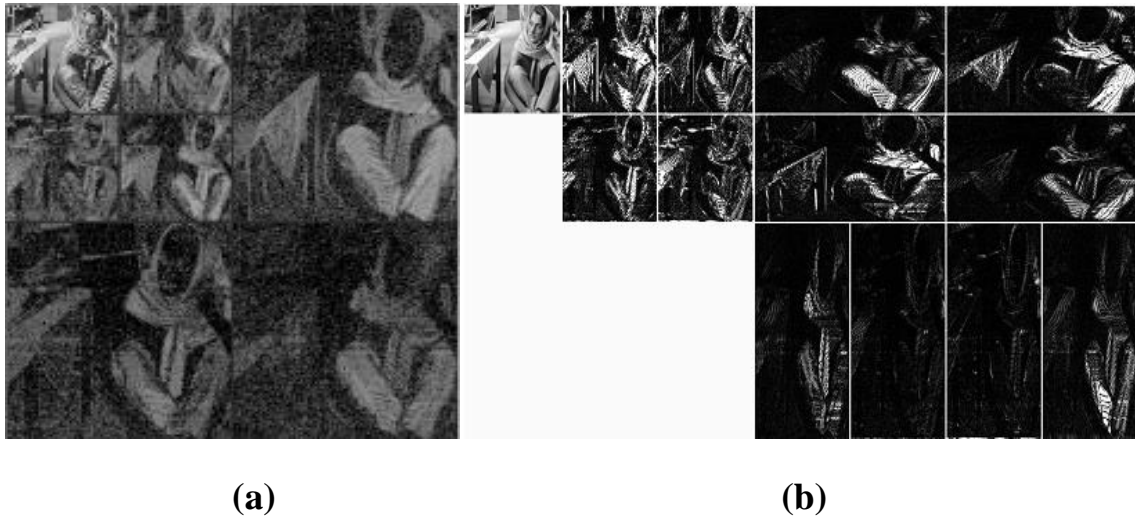


Figure 2.3 : Représentations d'image par Ondelettes (a) et Contourlets (b)

2.6 Conclusion :

Dans ce chapitre, on a déduit que les méthodes de compression sans pertes donnent un taux de compression inférieur à celle donné par les méthodes de compression avec pertes. Par contre, elles sont caractérisées par une rapidité de codage meilleure. Pour améliorer le taux de compression, la qualité de l'image et le temps de calcul, on combine ces deux méthodes de compression.

Pour la compression d'images, les ondelettes apparaissent donc comme un outil extrêmement efficace. Elles permettent en effet d'obtenir simultanément une qualité de compression supérieure aux méthodes précédentes et une compression progressive de l'image.

Le chapitre suivant présente un nouveau membre de cette famille nommé les Curvelets.



Chapitre III :

La compression par curvelets et quantification.

3.1 Introduction:

Les curvelets ont été proposées par E. Candes et D. Donoho, constituent une nouvelle famille de bases d'ondelettes géométriques plus efficaces que les transformées traditionnelles, et qui sont conçues pour représenter de façon parcimonieuse les régularités curvilignes.

3.2 L'ondelette vers le curvelet :

La théorie des ondelettes a eu naissance dans le milieu des années 80 [20]. De nos jours, elle s'utilise dans de nombreux domaines comme l'analyse harmonique, la vision par ordinateur, le traitement du signal, la compression d'images et autres. Son succès est dû à son adaptabilité aux données et à sa facilité d'implémentation. La transformée en ondelettes (TO) est un outil très efficace dans l'analyse et le débruitage de signaux non stationnaires présentant des discontinuités ponctuelles. Dans les images, les discontinuités sont principalement générées par les contours et par conséquent elles sont distribuées le long d'une courbe lisse dans le plan spatial. Dans la dernière décennie, une diversité de nouveaux outils d'analyse multi-échelle basés sur une analyse en ondelettes dans un espace géométrique a été proposée [21]. Ces nouvelles techniques visent à améliorer le pouvoir d'approximation des ondelettes en introduisant une notion de directionnalité ce qui leur permet de mieux tenir compte de la géométrie présente dans l'image.

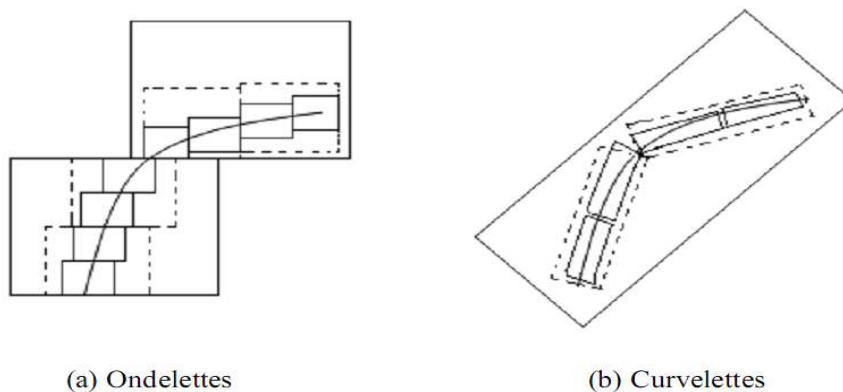


Figure 3.1 Fonctions d'approximation des ondelettes et des curvelettes.

La figure (3.1) montre une comparaison de l'approximation non-linéaire des ondelettes et des curvelettes. On remarque que les ondelettes nécessitent beaucoup de coefficients pour

représenter précisément un tel contour. Cependant les curvelettes peuvent représenter un contour lisse avec moins de coefficients pour la même précision. [18,22, 23].

3.3 Principe de la transformée en curvelettes :

L'idée de la transformée en curvelettes est d'appliquer la décomposition pyramidale en ridgelettes non plus sur l'image elle-même, mais sur chacune des sous-bandes obtenues en sortie de la transformée en ondelettes non décimées de l'image. Pour cela, une analyse multi-résolutions est appliquée à l'image avant de lui appliquer la transformée en ridgelettes localement sur des blocs dyadiques.

3.4 Transformée en curvelets discrète :

La transformée en curvelets [18][23][24] a ouvert la possibilité d'analyser une image avec des tailles de bloc différentes, mais avec une seule transformation. L'idée est de commencer par décomposer l'image en un ensemble de plans d'ondelettes, et ensuite d'analyser chaque bande par une transformée en ridgelets locale. La taille des blocs peut ainsi être changée à chaque échelle. Les différents niveaux de la pyramide en ridgelets multiechelle sont utilisés pour représenter les différentes sous-bandes de la transformée en ondelettes.

Cette décomposition en sousbandes impose une relation entre la longueur et la largeur des éléments, de telle façon qu'ils soient anisotropes et obéissent approximativement à une loi d'échelle parabolique $\text{largeur}^2 \approx \text{longueur}$.

La transformée en curvelets discrète d'une fonction continue $f(x)$ utilise une séquence dyadique d'échelles et un banc de filtres dont la bande passante δ_j est concentrée sur des fréquences $[2^{2j}, 2^{2j+2}]$, c'est à dire:

$$\delta_j(f) = \Psi_{2^j} * f, \hat{\Psi}_{2^j}(v) = \hat{\Psi}(2^{-2j} v) \quad (3.1)$$

Dans la théorie des ondelettes, on utilise habituellement une décomposition en bandes dyadique $[2^j, 2^{j+1}]$. Par opposition, les sous-bandes utilisées dans la transformée en curvelets discrète ont la forme nonstandard $[2^{2j}, 2^{2j+1}]$. Ceci est une particularité nonstandard de la transformée en curvelets qui est importante, car c'est ceci qui fixe la loi d'échelle parabolique.

Les différentes étapes de la décomposition sont les suivantes :

- Décomposition en sous-bandes. L'image f est décomposée en sous-bandes.
- Partitionnement doux. Chaque sous-bande est fenêtrée avec un filtre lisseur à une échelle appropriée (de taille $\sim 2^{-j}$).
- Transformation en ridgelets. Chaque bloc est analysé par une transformée en ridgelets.

Dans cette définition, les deux sous-bandes $[2^{2j}, 2^{2j+1}]$ et $[2^{2j+1}, 2^{2j+2}]$ sont fusionnées avant d'appliquer la transformée en ridgelets.

Implémentation numérique : La transformée en ondelettes isotrope "à trous" est spécialement bien adaptée au besoin de la transformée en curvelets discrète [24].

Cet algorithme décompose une image de la taille $n \times n$ en une superposition de plans :

$$f[i_1, i_2] = a_j[i_1, i_2] + \sum_{j=1}^j d_j[i_1, i_2] \quad (3.2)$$

où a_j est le plan lissé de l'image originale et les d_j représentent les coefficients en ondelettes de f à l'échelle j . Ainsi, l'algorithme fournit $J + 1$ sous-bandes de taille $n \times n$.

Algorithme de la transformée en curvelettes :

L'algorithme de la transformée en curvelets 2D peut être résumé par le pseudo-code suivant :

1. Appliquer la transformation isotrope "à trous" à J échelles.
2. Initialiser $B_1 = B_{\min}$, où B_{\min} est la taille du bloc de la première échelle.
3. Pour $j = 1, \dots, J$ faire
4. Partitionner la sous-bande w_j avec une taille de bloc b_j et appliquer la transformée en ridgelets sur chaque bloc.
5. si j modulo 2 = 1 alors
6. $B_{j+1} = 2B_j$
7. sinon $B_{j+1} = B_j$
8. fin Si
9. fin Pour

La taille des fenêtres de localisation est doublée tous les deux sous-bandes dyadiques, permettant de garantir la propriété fondamentale de la transformation en curvelets, qui est que les éléments de longueur proche de $2^{-j/2}$ sont utilisés pour l'analyse et la synthèse de la j ème sous bande $[2^j, 2^{j+1}]$. On peut également noter que le plan lissé C_j de l'image est laissé intact. La valeur $B_{\min} = 16$ est couramment utilisée dans la plupart des implémentations.

Cette implémentation est redondante. Le facteur de redondance est égal à $16J + 1$, lorsque J échelles sont utilisées. Cet algorithme permet une reconstruction exacte, du fait que chaque étape de la décomposition est elle-même inversible.

Les figures 3.2 et 3.3 présentent le principe de la transformée en curvelettes ainsi qu'un exemple de décomposition en curvelettes d'une image en 3 niveaux.

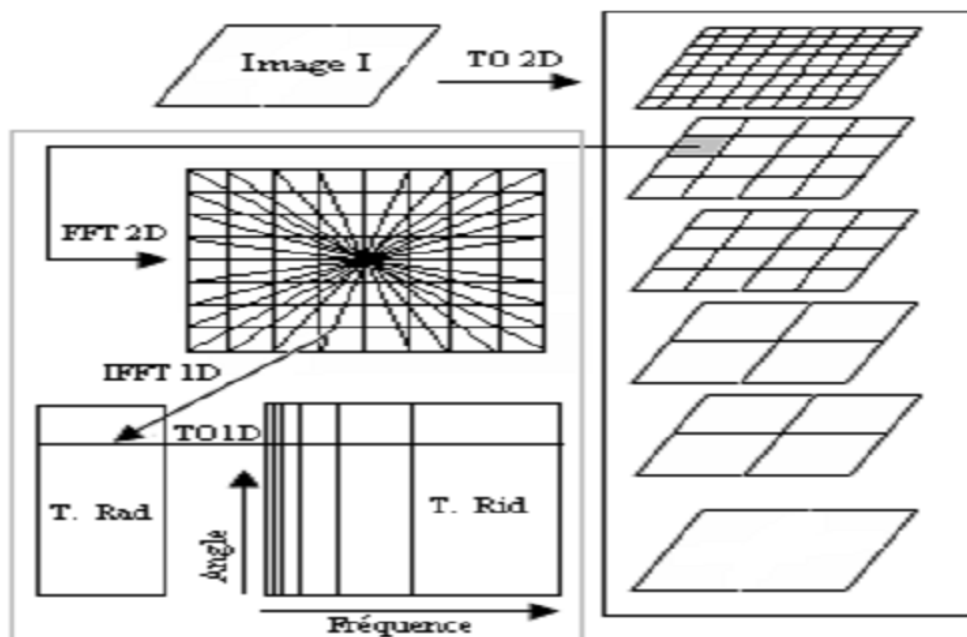


Figure. 3.2. Principe de la transformée en curvelettes.

La reconstruction est obtenue en déroulant l'algorithme dans l'ordre inverse (transformée en ridgelettes inverse, reconstruction de la partition initiale de l'image, filtrage dual pour la reconstruction de l'image à partir des sous-bandes).

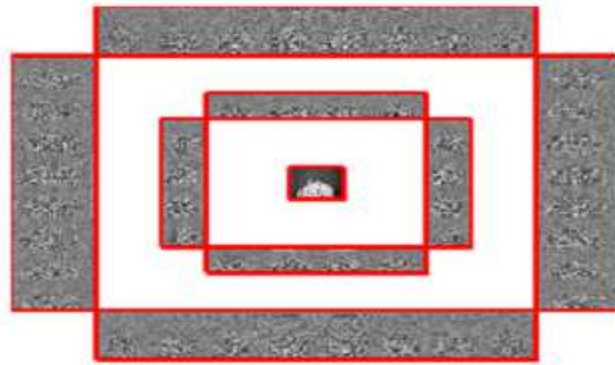


Figure. 3.3. Décomposition en curvelettes d'une image en 2 niveaux.

3.5 Propriétés des Curvelets: [25]

La transformée en Curvelet est une transformée multi échelle multi directionnelle avec des atomes indexés par un paramètre de position l , d'échelle j et de direction k .

Nous citons maintenant quelque propriétés de la transformée en curvelet :

1. **le frame étroit** : dans une base orthonormée, nous pouvons facilement étendre une fonction $f(x_1, x_2) \in L^2(\mathbb{R}^2)$ comme une série de curvelet : nous avons la formule de reconstruction :

$$f = \sum_{j,k,l} \langle f, \varphi_{j,l,k} \rangle \varphi_{j,l,k}$$

En utilisons l'identité de parseval, on peut aussi l'écrire sous forme :

$$\sum_{j,l,k} |\langle f, \varphi_{j,l,k} \rangle|^2 = \|f\|_{L^2(\mathbb{R}^2)}^2, \forall f \in L^2(\mathbb{R}^2)$$

2. **échelle parabolique** : Les curvelets sont liés à la localisation spatiale échelle. Les atomes sont anisotropes répondant à une loi d'échelle paraboliques ou les rectangles D'Heisenberg satisfont à la notion du support échelle qui est différent dans ce cas : obéit à la loi (non dyadique) :

$$\text{Hauteur} = \text{longueur}^2, \text{hauteur} \approx 2^{-j/2}, \text{largeur} \approx 2^{-j}$$

Nous observons ici que plus le support sera long, plus la courbe sera régulière.

3. **comportement oscillatoire** : la curvelet mère $\varphi_j(x_1, x_2)$ est oscillante et de moyenne nulle selon x_1 (en largeur), alors qu'elle se comporte comme un passe bas selon la direction x_2 (en longueur).

4. **moment** : le modèle du curvelet est dit que pour avoir « q » moments nulles il faut :

$$\int_{-\infty}^{\infty} \varphi_j(x_1, x_2) x_1^n dx_1 \text{ pour toutes } 0 \leq n < q \text{ et } \forall x_2$$

3.6 Quantification :

La quantification est la seule étape non réversible du schéma de compression. Son objectif est double : extraire l'information jugée pertinente et faciliter le codage entropique. L'information quantifiée doit alors pouvoir être décrite par un nombre de symboles suffisamment petit pour atteindre de bonnes performances en débit mais aussi suffisamment grand pour ne pas trop détériorer l'information originale.

Pour un signal d'entrée $X \in \mathfrak{R}$, une quantification scalaire approxime chaque coefficient du signal d'entrée indépendamment les uns des autres de sorte que le signal quantifié obtenu est constitué par N opérations séparées. Lorsque les coefficients de X sont très interdépendants, un quantificateur vectoriel qui quantifie ensemble les N coefficients de X peut nettement améliorer la performance d'un quantificateur scalaire. Cependant, un quantificateur vectoriel est d'une plus grande complexité qu'un quantificateur scalaire. Si une étape de structuration de la redondance est utilisée en outre de la quantification, les coefficients qui en résulteront seront par définition moins dépendants les uns des autres, et le gain apporté par le quantificateur vectoriel ne vaudra pas le coût de calcul.

Les quantificateurs scalaires sont donc en pratique plus souvent utilisés. Dans cette sous-section, nous présentons quelques types de quantification scalaire.

3.6.1 Quantification Scalaire:

Puisqu'une quantification scalaire de X^N vers \tilde{X}^N peut être vue comme N opérations séparées, étudier un quantificateur scalaire de dimension N revient à étudier N quantificateurs unidimensionnels. Dans la suite de cette sous-section, on considère donc un quantificateur scalaire q avec $N = 1$ et la variable aléatoire X a réalisations dans X .

Si $X = [a, b]$ (avec éventuellement $a = -\infty$, $b = +\infty$), l'opération de quantification revient à partitionner l'intervalle $[a, b]$ en K intervalles $]\alpha_{k-1}, \alpha_k]$ $1 \leq k \leq K$, puis à associer à chaque valeur x un point de quantification x_k tel que

$$\forall x \in]\alpha_{k-1}, \alpha_k], q(x) = x_k. \quad (3.3)$$

La longueur de chaque intervalle de quantification $]\alpha_{k-1}, \alpha_k]$ est appelée pas de quantification et est notée $\Delta_k = \alpha_k - \alpha_{k-1}$.

a. Quantification scalaire uniforme :

Le quantificateur scalaire uniforme à débit fixe est entièrement déterminé par :

- Les $L+1$ niveaux de décisions : x_0, x_1, \dots, x_L , qui partitionnent en L intervalles égaux l'axe des réels \mathbb{R} et déterminent le pas de quantification.
- Les L valeurs de reproduction : y_1, y_2, \dots, y_L , qui sont les centres de masse de chacun des intervalles de décision.

Le débit (en bits/échantillon) de ce quantificateur est donc donné par $R = \log_2 L$ [bits/échantillon]. Il apparaît deux sortes d'erreurs ou bruits de quantification :

- Le bruit granulaire qui se produit lorsque la valeur d'entrée x se situe dans l'un des intervalles $[x_i, x_{i+1}]$, l'erreur résultante qui est la différence entre x et la valeur quantifiée $Q(x)$ ne peut être supérieure à un demi pas de quantification. Le pas de quantification étant défini par :

$$\Delta = y_{i+1} - y_i, \quad i = 1, 2, \dots, L-1.$$

- Le bruit de surcharge ou de dépassement qui se produit lorsque la valeur d'entrée se situe hors de l'intervalle $[x_0, x_L]$. La valeur de reproduction est alors : y_1 ou y_L , et l'erreur de quantification peut être supérieure à un demi pas de quantification.

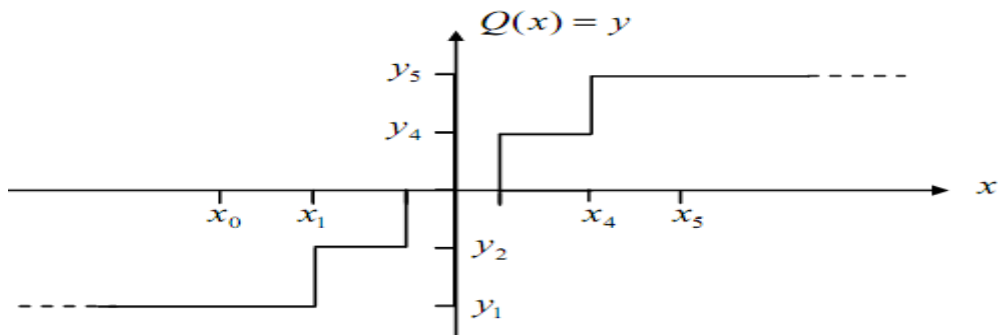


Figure 3.4 : Exemple d'un quantificateur scalaire uniforme avec cinq niveaux $L = 5$

Propriétés :

- ✓ Tout les intervalles sont de même tailles: $\Delta = b_i - b^{i-1}$
- ✓ La reconstruction utilise le point central de l'intervalle de quantification $[b_i, b^{i-1}) \rightarrow (b_i + b^{i-1})/2$
- ✓ Quantificateur midrize: 0 n'est pas un niveau de sortie.
- ✓ Quantificateur midread: 0 est pas un niveau de sortie.

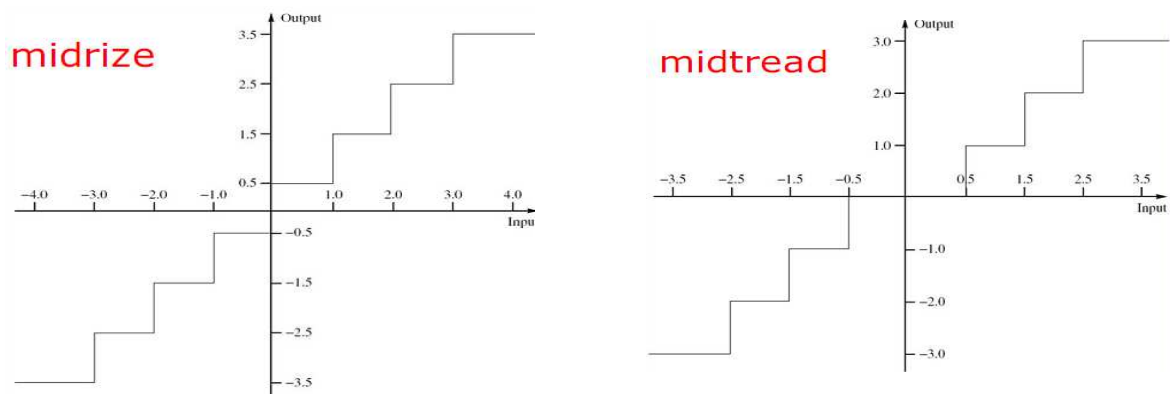


Figure 3.5 : Quantificateur midread , Quantificateur midrize

b. Quantificateur uniforme à zone morte :

Un quantificateur scalaire à zone morte (dead zone en anglais) est un quantificateur scalaire où l'intervalle autour de zéro est plus large. La zone morte (ou dead zone) notée T , qualifie cet intervalle, qui permet à l'ensemble des valeurs considérées comme petites, d'être quantifiées à une seule et même valeur zéro.

Stricto sensu, ce type de quantificateur est non uniforme ($T \neq \Delta$). Toutefois, si tous les autres pas de quantification sont égaux, on qualifie le quantificateur de uniforme à zone morte, et si T reste de grandeur comparable aux autres pas de quantification.

Ce type de quantificateur est très répandu en compression d'image. C'est celui que nous utiliserons dans les schémas de compression que nous proposerons dans la suite de ce manuscrit, en précisant la longueur de la zone morte.

La figure 3.6 présente une illustration des quantificateur scalaire uniforme et quantificateur scalaire uniforme à zone morte. Sur la figure 3.6(a) est représenté graphiquement un quantificateur scalaire uniforme de pas de quantification Δ tandis que la figure 3.6(b) montre un quantificateur scalaire uniforme à zone morte $T = 2\Delta$.

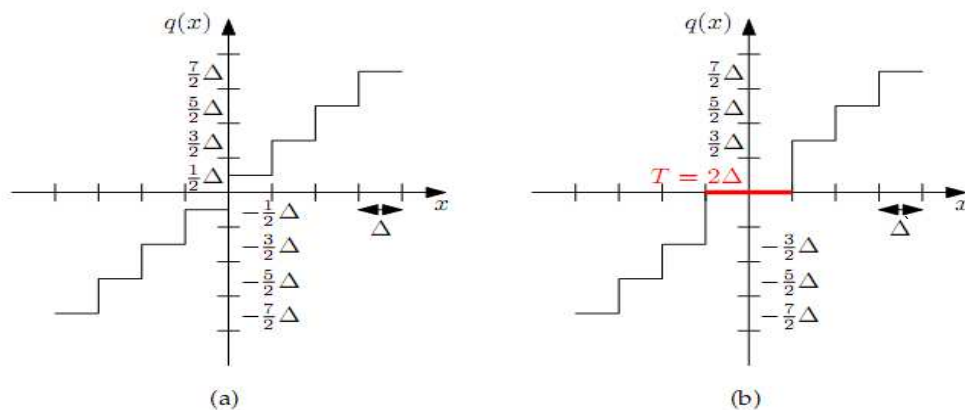


Figure 3.6 : Illustration des quantificateur scalaire uniforme (a) et quantificateur scalaire uniforme à zone morte (b, ici $T = 2\Delta$)

c. Quantification Scalaire non uniforme :

La quantification non uniforme est motivée par le constat qu'un quantificateur uniforme n'est pas optimal que dans le cas où le signal d'entrée a une distribution

uniforme, ce qui n'est pas le cas pour la plupart des signaux, elles suivent des distributions différentes et des fois aléatoires : Les signaux de parole ont une densité de probabilité proche d'une loi de Laplace, tandis que les signaux de musique ont souvent une distribution gaussienne. La distribution des valeurs des pixels d'une image peut beaucoup varier d'une image à l'autre, et elle est rarement uniforme. Ainsi, certaines valeurs de quantification correspondant à des échantillons peu probables sont rarement utilisées, ce qui représente un gaspillage de bits.

c.1 Algorithme de Lloyd-Max :

Les travaux de Lloyd et de Max [26][27] ont permis d'établir les deux conditions nécessaires d'optimalité pour construire un quantificateur adapté à des sources quelconques. Comme nous.

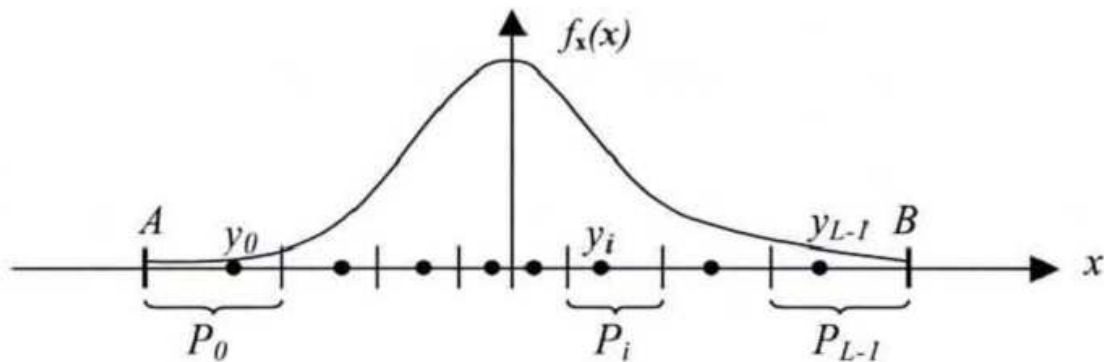


figure 3.7: Quantificateur Scalaire non uniforme

L'avons expliqué pour la QS non uniforme, définir un quantificateur "optimal" consiste à trouver les L meilleures partitions P_i du signal et les meilleurs représentants Y_i associés (les représentants du dictionnaire C) afin de minimiser la distorsion. Cette optimisation conjointe n'admet pas de solution simple. Cependant, il est possible de définir les deux conditions suivantes:

- connaissant le dictionnaire, il est possible de trouver le meilleur partitionnement. En effet, étant donné un dictionnaire $C = \{y_0, \dots, y_{L-1}\}$, il est possible de classer les valeurs de x dans chaque partition P_i telle que la règle 3.4 dite du plus proche voisin soit vérifiée;

$$P_i = \{x / d(x, y_i) \leq d(x, y_j) \quad \forall i \neq j, j \in \{0, \dots, L-1\}\} \tag{3.4}$$

➤ connaissant le partitionnement, il est possible de trouver le meilleur dictionnaire. En effet, étant donné un partitionnement $P = \{P_0, \dots, P_{L-1}\}$, il est possible de trouver le meilleur représentant Y_i pour chaque partition P_i en calculant le centre de gravité ou centroïde de la partition P_i , formule 3.5.

$$y_i = E[x | x \in P_i] = \frac{\int_{x \in P_i} x f_x(x) dx}{\int_{x \in P_i} f_x(x) dx} \tag{3.5}$$

De manière générale, les algorithmes d'optimisation pour la création du dictionnaire sont basés sur ces deux conditions. L'algorithme de Lloyd-Max utilise ces deux conditions afin de construire itérativement un dictionnaire de façon à minimiser la distorsion DQs.

c.2 Quantification non-uniforme par compression (companding) :

Pour un signal de source non-uniformément réparties, on pourrait utiliser une fonction non linéaire $F(x)$ pour le convertir en un autre signal avec une PDF proche d'une distribution uniforme. Ensuite, le quantificateur uniforme simple et efficace pourrait être utilisé.

Après les indices de quantification sont transmis (codés) et reçus ensuite par les décodeurs, ils sont d'abord quantifié inversement pour reconstituer les valeurs quantifiées de manière uniforme, puis la fonction inverse $F^{-1}(x)$ est appliqué afin de produire les valeurs finales quantifiés.

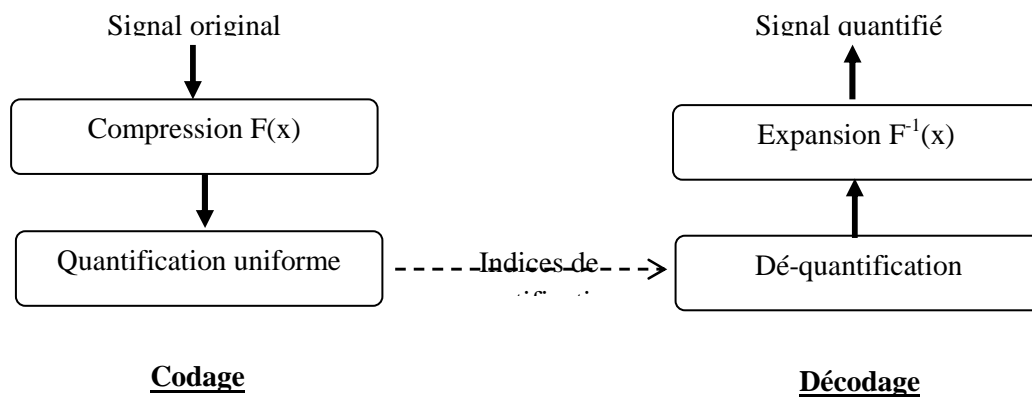


Figure 3.8 : Processus de quantification non uniforme par companding

La fonction non linéaire $F(x)$ est appelé une fonction de compression, car elle a généralement une forme analogue à celle représentée sur la Figure 3.6 qui amplifie les faibles amplitudes de signal source et comprime les échantillons de grandes amplitudes.

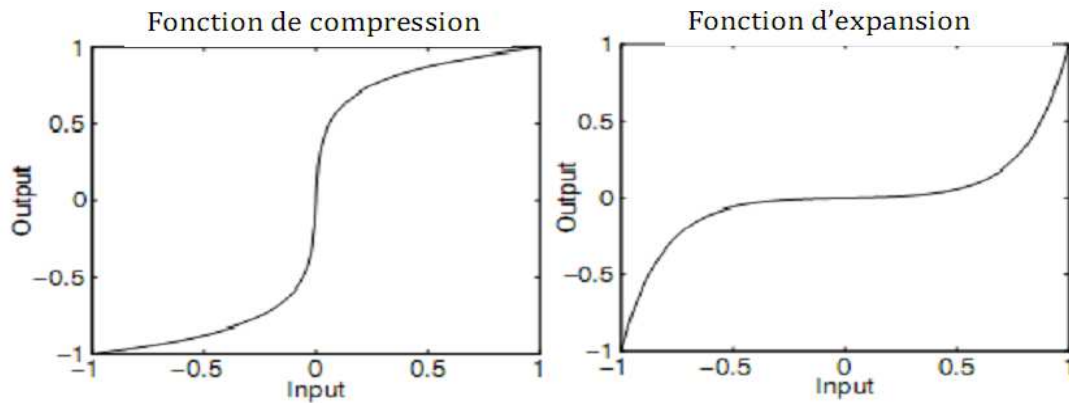


Figure 3.9 : Exemples des fonctions de compression et d'expansion

La fonction inverse est appelé une fonction d'expansion (expand en Anglais), en raison que l'inverse de la compression est l'expansion. Après la compression-expansion, d'où vient le terme "companding", les bornes de décision efficaces sont non uniformes lorsqu'elle sont vue à partir de la sortie de bloc d'expansion, ainsi l'effet global est une quantification non uniforme. Cette forme de compression est généralement utilisée avec des formes de PDF qui possède une densité de probabilité importante pour les petites valeurs absolues d'échantillons et décroît en direction de grandes valeurs absolues d'échantillons, tel que la distribution gaussienne et Laplacienne, afin de rendre le PDF de signal converti similaire à une distribution uniforme.

Deux types des fonctions communes sont utilisés dans le quantificateur non uniforme: la quantification (companding) à loi de puissance et la quantification (companding) logarithmique. Ces techniques réduit le nombre de bits nécessaire a coder le signal et maintient un rapport signal à bruit acceptable.

Pour le companding à loi de puissance, une fonction de la forme $c_{\text{power}}(|x|) = |x|^p$ est utilisée.

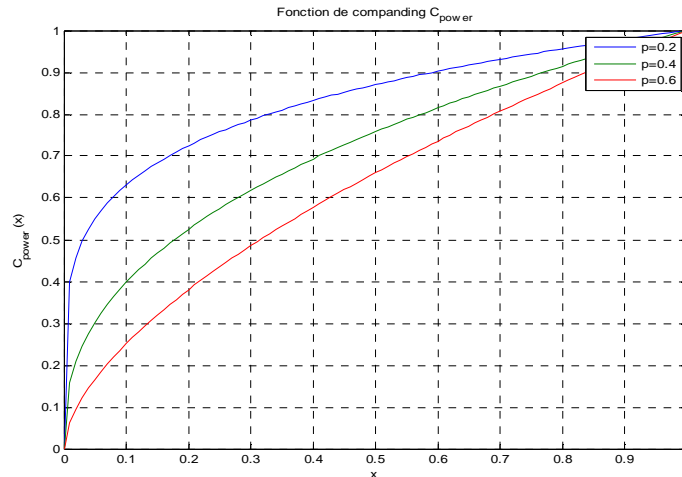


Figure 3.10 : Fonctions A loi de puissance pour (P=0.2, P=0.4, P=0.6)

Alors qu'une fonction de forme $\log(x)$ est utilisée dans le companding logarithmique, les fonctions souvent utilisés sont les fonctions A-law et μ -law, cette dernière sera détaillée en plus puisqu'elle sera introduite dans la méthode de quantification proposée ultérieurement.

c.3 MU-LAW companding :

Mu-law companding (Roddy et Coolen, 1997; Tomasi, 2004) est une forme de quantification par compression logarithmique, elle a été introduite pour la première fois dans l'industrie Téléphonique au Japon et les États-Unis (le standard G.711), pour les données audio elle exploite le fait que le système de perception auditive humaine est un processus logarithmique, dans lequel les signaux sonores de hautes amplitudes ne nécessitent pas la même résolution que ceux de faibles amplitudes.

L'oreille humaine est plus sensible au bruit de quantification dans les petits signaux que les signaux forts. « Mu-law » applique une fonction de quantification logarithmique $F(x)$ pour ajuster la résolution de données en proportion du niveau du signal d'entrée. Les petits signaux sont représentés avec plus de précision que les grands signaux.

La fonction $F(x)$ de compression Mu-law est définie par l'équation (EQ 3.6), où μ est le coefficient de compression ($\mu = 255$ aux États-Unis et au Japon) et x est l'entier normalisé pour être comprimé. [28]

$$F(x) = \frac{\text{sign}(x) \times \ln(1 + \mu|x|)}{\ln(1 + \mu)} \quad 0 \leq |x| \leq 1 \quad (3.6)$$

La Figure 3.9 présente cette fonction pour les trois valeurs de $\mu = 1, 10, \text{ et } 255$. Plus la valeur de μ est grande plus les valeurs de faibles amplitudes sont amplifiées.

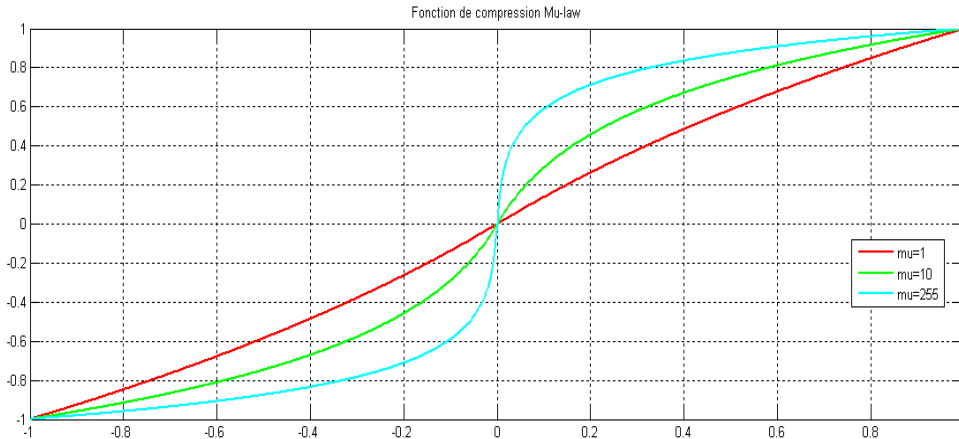


Figure 3.11 : La fonction de compression MU-LAW ($\mu=1, 10, 255$)

Le décodage des données codées n'est ensuite qu'une question d'inverser les étapes de l'encodage.

Les étapes de quantification par compression mu-law peuvent être représentées par le schéma de la figure 3.10.

Il est clair que les grandes valeurs de μ provoquent une quantification grossière pour les grandes amplitudes et plus précis pour les faibles amplitudes.

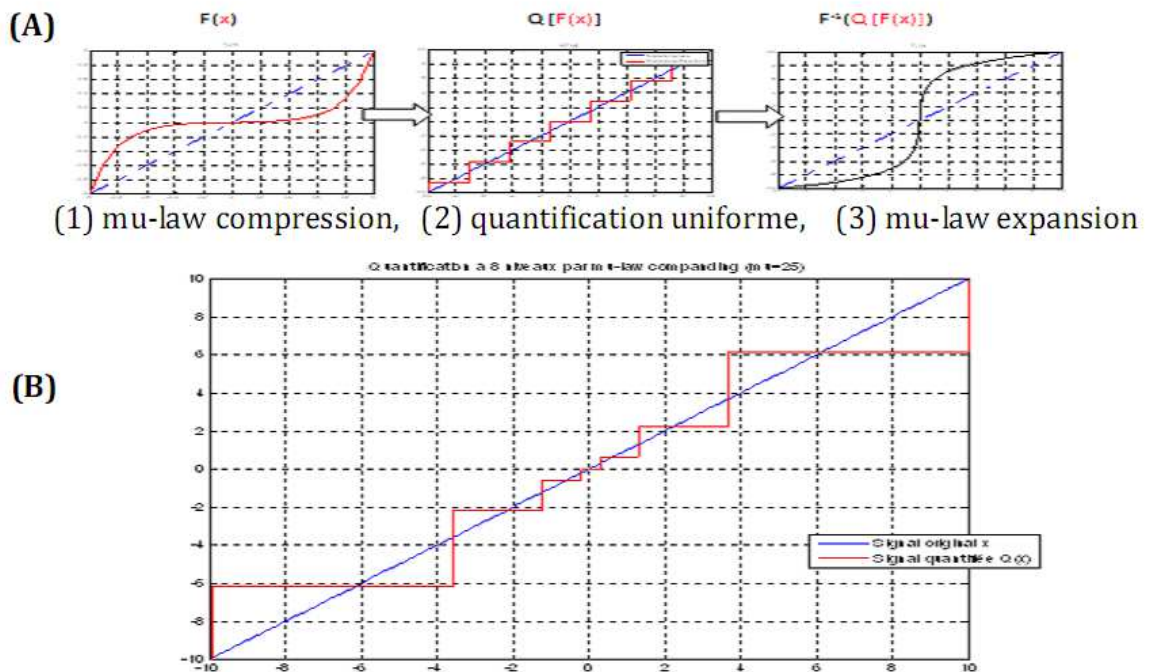
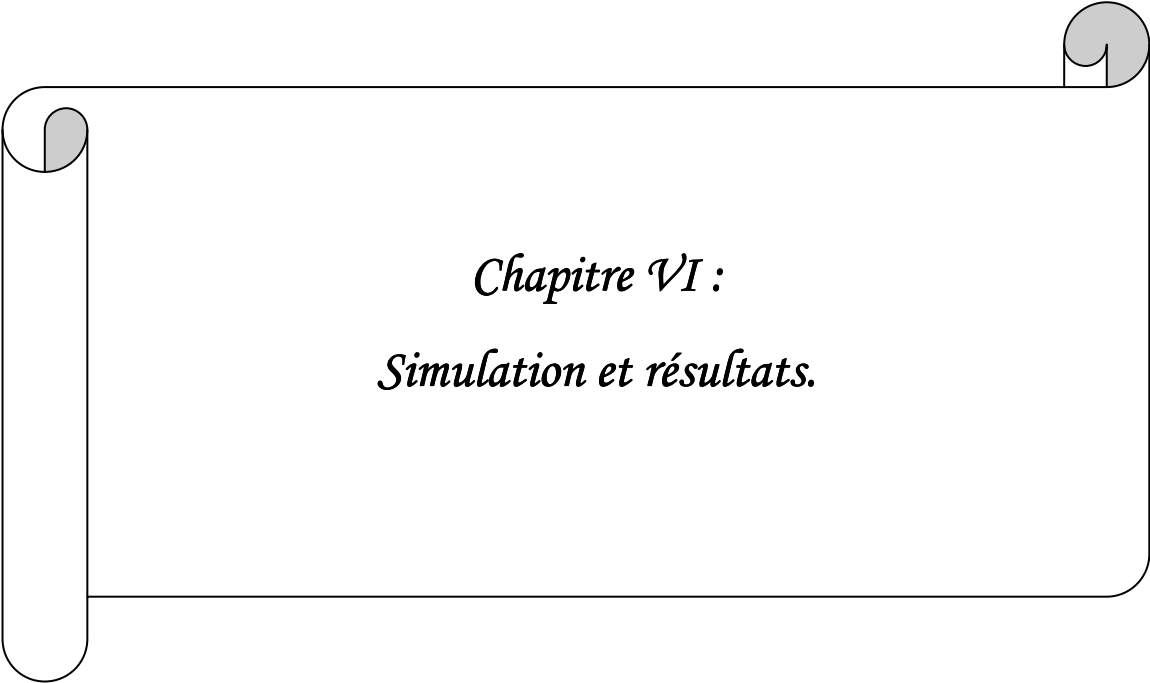


Figure 3.12 : (A)-étapes de quantification par companding, (B)-quantification MU-LAW à 8 niveaux (MU=25)

3.7 Conclusion :

La quantification est plus importante dans schéma de compression avec perte, elle a un double objectif, d'une part elle sert à extraire l'information jugée pertinente de l'ensemble de coefficients, et d'autre part elle les prépare pour la phase de codage entropique.

La quantification scalaire est plus utile que la quantification vectoriel si elle est précédée par une transformation qui génère des coefficients fortement dé-corrélés, elle peut être plus performant en adaptant le dictionnaire de quantification au caractéristiques de signal à traité.



Chapitre VI :
Simulation et résultats.

4.1 Introduction :

La compression avec pertes, par opposition à la compression sans pertes, se permet d'éliminer quelques informations pour avoir le meilleur PSNR possible, tout en gardant un résultat qui soit le plus proche possible des données originales.

L'utilisation d'une transformée capable de détecter les informations essentielles dans une image, et les représenter par un petit nombre de coefficients significatifs permet d'attribuer plus de bits à ces coefficients lors de la phase de quantification et codage. Dans ce chapitre, nous appliquons l'algorithme de transformée en Curvelets décrite au chapitre 3 ainsi que le quantificateur proposé dans le chapitre 4 à la compression d'images.

4.2 Algorithme de compression :

Nous avons développé deux méthodes de compression, une de type de quantification non uniforme par μ -law et l'autre de type Algorithme de lloyd-Max. sera donc décrit par la figure suivante :

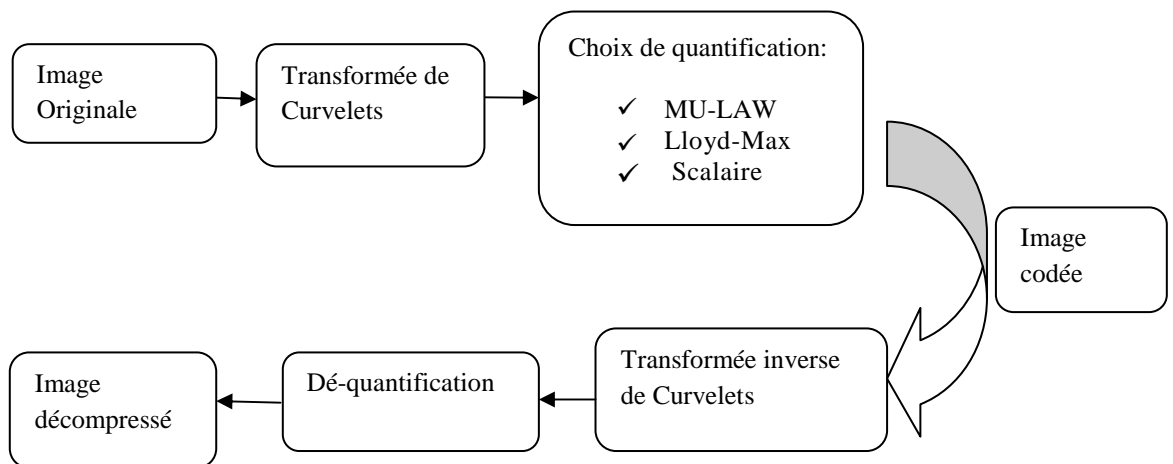


Figure 4.1 : Méthode de compression et décompression proposée

4.3 Présentation des résultats :

Afin de tester l'algorithme proposée, on l'applique sur un trois d'images présentant des régularités géométriques (barbara, girls, peppers) avec différents degrés d'expansion en variant le facteur (μ) et le partitions (p). Les images utilisées sont des images à niveaux de gris et de taille 512×512 pixels.

Le tableau 4-1 présentent les PSNR en fonction de Facteur d'expansion (μ) et obtenues pour chacune des images de test.

Image Facteur d'expansion (μ)	Girls		Barbara		Peppers	
	Quantification scalaire	Quantification Mu-law	Quantification scalaire	Quantification Mu-law	Quantification scalaire	Quantification Mu-law
	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)
1	22.6895	52.9066	22.4789	53.1111	22.2441	52.7819
10	10.8211	40.8213	9.3913	44.4126	9.6076	39.4642
50	-	34.4946	-	34.9667	-	30.5473
150	-	31.2714	-	30.5621	-	27.1473
255	-	30.0919	-	29.0652	-	26.0129

Tableaux 4-1 Résultats obtenus de méthode Mu-law (PSNR)

Le tableau 4-1 comparent les résultats de PSNR obtenues par méthode Mu-law dans trois images girls, barbara et peppers

Les tableaux 4-2 présentent les PSNR en fonction de Facteur de partition (p) pour chacune des images de test.

Image Facteur de partition (p)	Girls		Barbara		Peppers	
	Quantificatio n scalaire	Quantification lloyd-Max	Quantificatio n scalaire	Quantification lloyd-Max	Quantification scalaire	Quantification lloyd-Max
	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)	PSNR(dB)
50	22.3272	44.4515	21.3974	42.8960	21.5997	42.5895
100	10.5922	41.1658	10.5119	38.5419	10.5932	39.6266
500	-	29.2953	-	28.5356	-	28.1150

Tableaux 4-2 Résultats obtenus de méthode lloyd-Max (PSNR)

Les tableaux 4-2 comparent les résultats de PSNR par méthode lloyd-Max dans trois images girls, Barbara et peppers.

La figure 4.2 et 4.3 représente les courbes des caractéristiques obtenues pour les méthodes Mu-law (PSNR/ μ) et lloyd-Max (P SNR/ p)

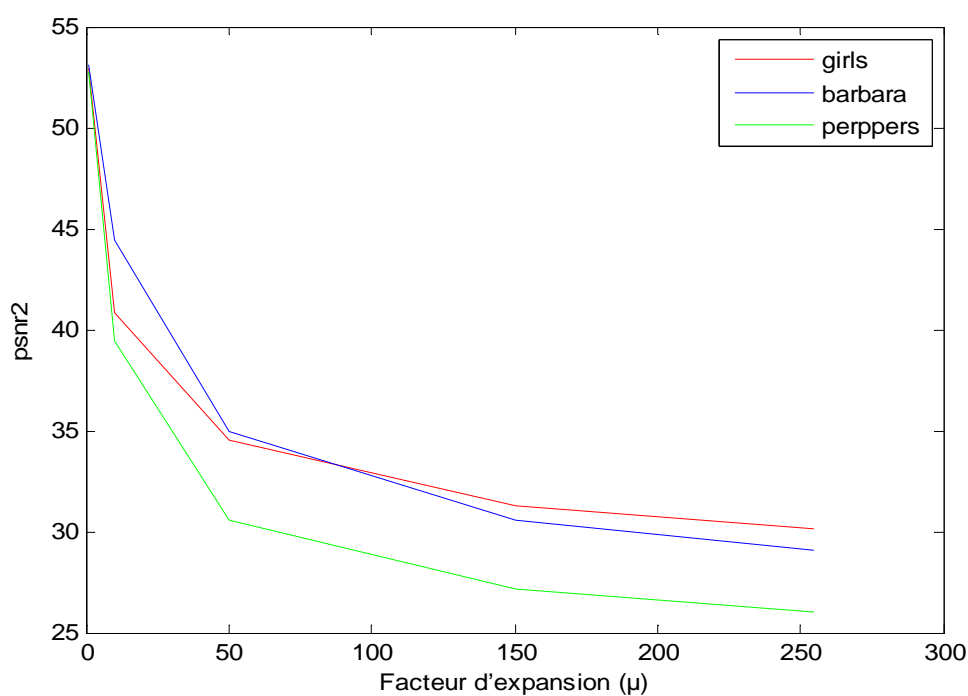


Figure 4.2 : Evolution du PSNR pour la méthode Mu-law

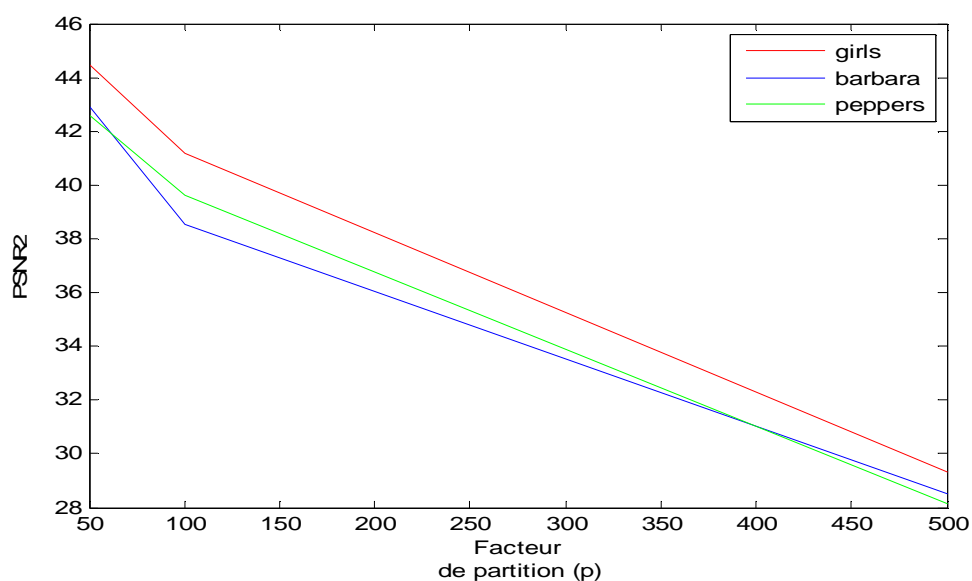


Figure 4.3 : Evolution du PSNR pour la méthode loyd-Max

Pour l'évaluation de la qualité visuelle des images compressées par la méthode proposée, les trois images de test : Barbara, Girls, Peppers sont compressés à moyens et forts PSNR de compression, les images décompressées sont ensuite présentées dans les figures (4.4), (4.5).

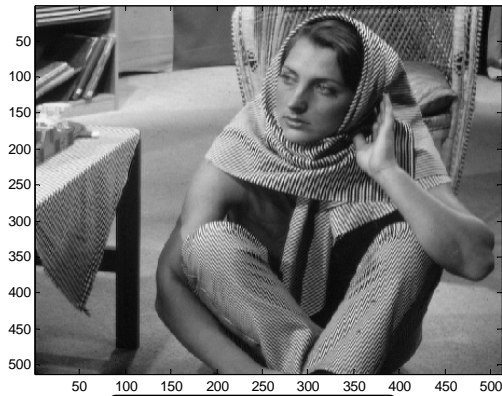


Image originale

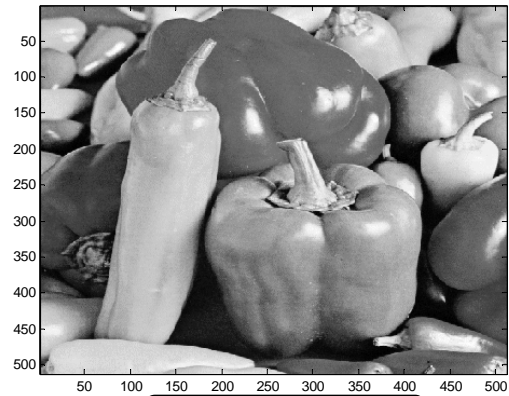


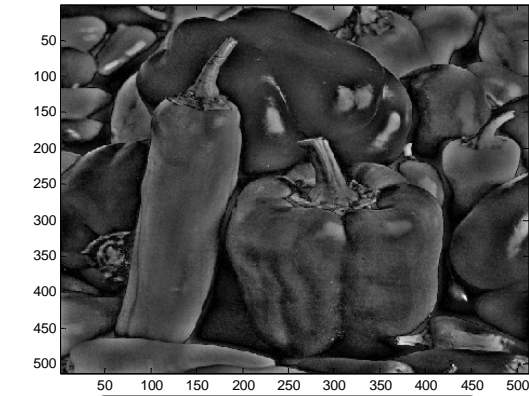
Image originale

$\mu=10$



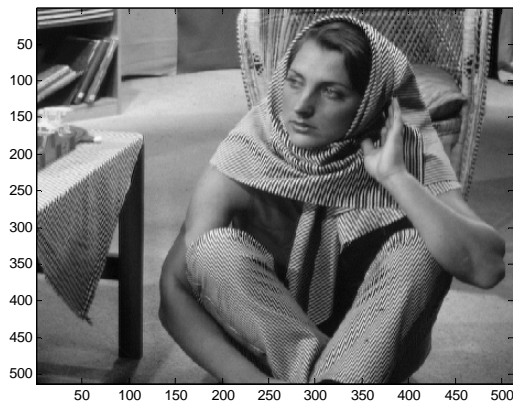
Sans Mu-law PSNR =9.39db

$\mu=10$



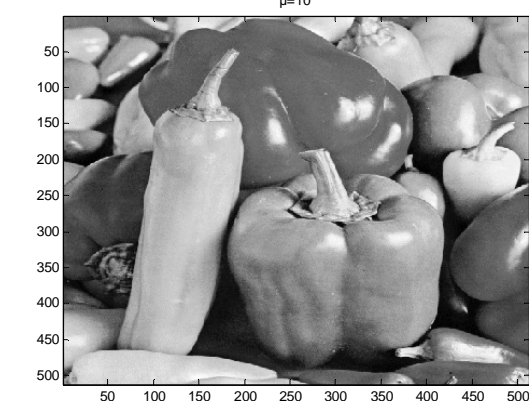
Sans Mu-law PSNR =9.60db

$\mu=10$



Avec Mu-law PSNR =44db

$\mu=10$

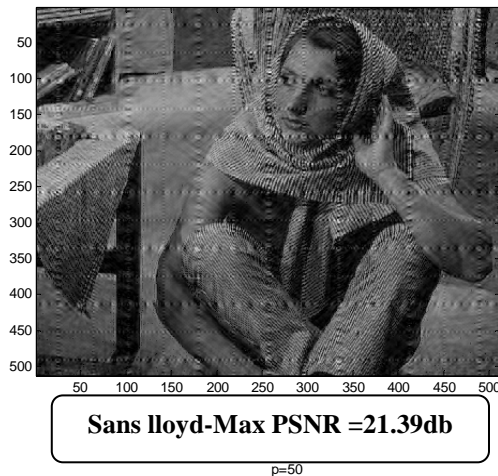


Avec Mu-law PSNR =39.46db

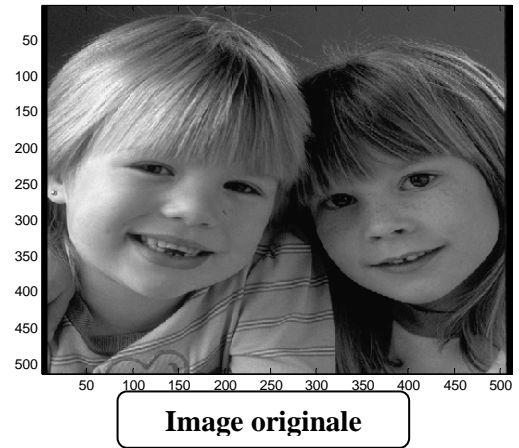
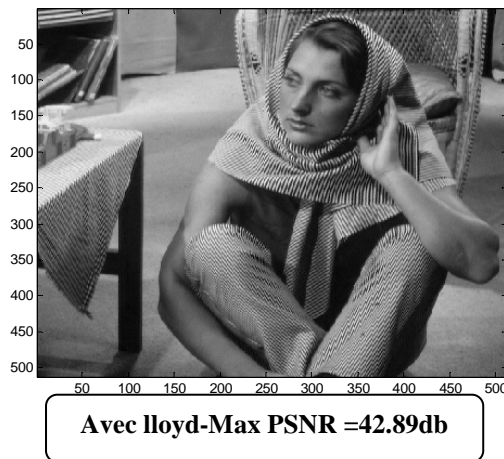
Figure 4.4 : résultats obtenues avec Mu-law et sans Mu-law



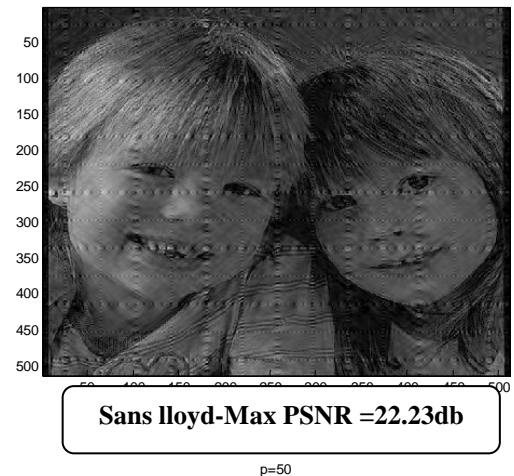
p=50



p=50



p=50



p=50

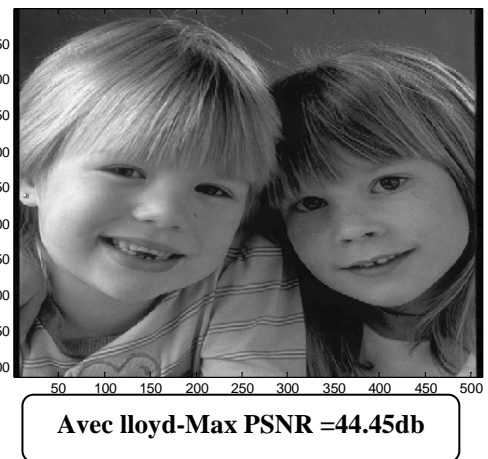


Figure 4.5 : résultats obtenues avec lloyd-Max et sans lloyd-Max

Les figures suivantes illustrent les résultats subjectifs obtenus pour les deux méthodes avec Mu-law et lloyd-Max appliqués sur l'image Barbara et peppers

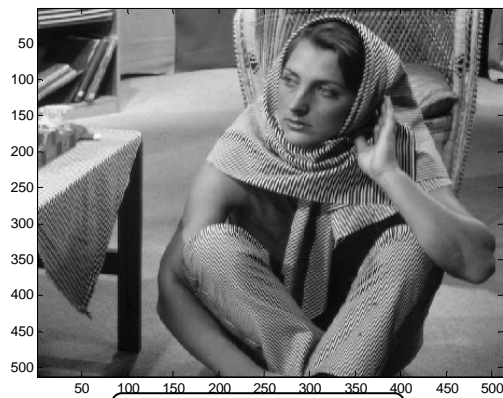


Image originale

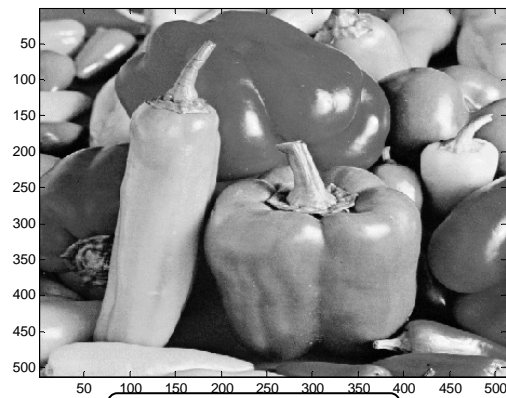
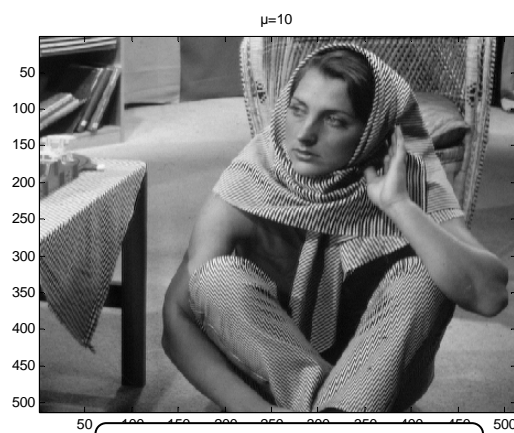
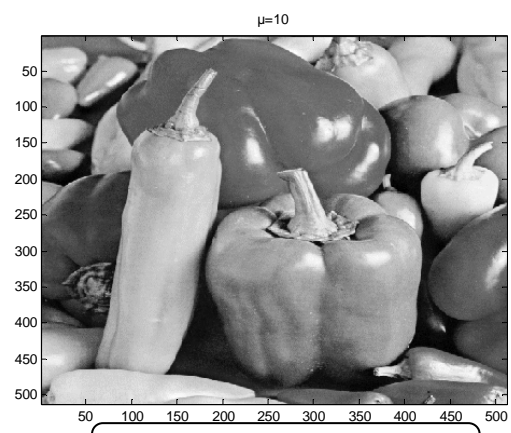


Image originale



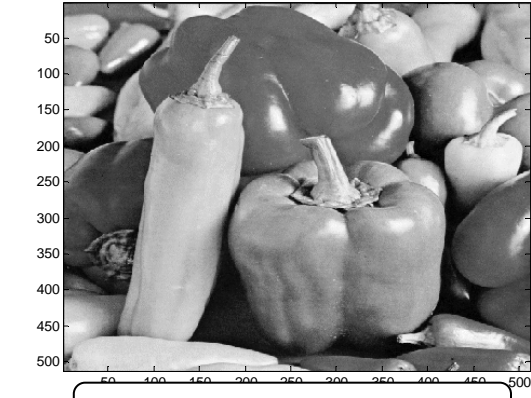
Avec Mu-law PSNR =44db



Avec Mu-law PSNR =39.46db



Avec lloyd-Max PSNR =42.89db



Avec lloyd-Max PSNR =42.58db

Figure 4.6 : Comparaison entre deux méthodes (Mu-law et lloyd-Max) pour l'image Barbara et Peppers

Nous remarquons que l'image avec lloyd-Max a meilleure par rapport avec Mu-law

4.4 Analyse et Discussions :

Nous avons simulé et testé l'algorithme développé sur un ensemble d'images présentant des régularités géométriques pour différentes valeurs de facteur de quantification logarithmique pour chacune.

Les résultats obtenus pour les trois images sont montrés sous forme de tableaux : 4-1, 4-2.

Malgré que les valeurs de PSNR soient mieux pour les image de la méthode Mu-law , les distorsions présentes semble plus dommageable à cause l'importance des petites objets éliminés dans la compression. La même remarque peut être vue dans les images de la méthode lloyd-Max, malgré la forte présence des formes géométriques régulières, l'erreur introduite dans la quantification des coefficients de faibles amplitudes a généré des distorsions importantes sur les formes géométriques

Dans ce qui suit nous comparons un méthode de compression Mu-law avec la méthode de compression lloyd Max. Dans cette dernière, l'image est d'abord dé-corrélée par la transformée en Curvelets discrètes rapide (FDCT). Les coefficients de Curvelets sont ensuite quantifiés par une nouvelle méthode de quantification scalaire uniforme basée sur une optimisation de niveaux de reconstruction. Enfin, les coefficients résultants sont codés par un codeur entropique. Les résultats de PSNR obtenus, pour l'image « Barbara », par les deux méthodes

Les figures présentées ci-dessus montrent que la méthode proposée donne généralement des meilleurs résultats en terme de rapport PSNR par rapport à la compression avec Mu-law et lloyd-Max pour les images girls, Barbara, et peppers .

4.5 Conclusion :

Dans ce chapitre, nous avons proposé un algorithme de compression d'images fixes basée sur une transformation en Curvelets FDCT via Wrapping, et une quantification scalaire non uniforme où le quantificateur est choisi en fonction des caractéristiques de décroissance des coefficients de Curvelets obtenus afin de favoriser les coefficients les plus significatifs. Tandis que la qualité visuelle est mesurée de façon objective par l'indice de PSNR.

Conclusion générale

Dans le présent mémoire nous avons exposé d'une façon générique les étapes de compression d'une image en curvelets tout en tenant en considération les principales contraintes suivantes:

- Le taux de compression assez important pour réduire l'occupation au niveau des canaux de transmission.
- Après l'opération inverse de la compression, il faut que le PSNR soit le plus élevé possible car c'est l'un des paramètres permettant l'estimation de la qualité de l'image compressée.
- Réduction des coûts en temps lors des différents traitements.

Dans ce travail, nous avons étudié différentes méthodes de quantification d'images fixes à niveau de gris.

Les résultats de simulation obtenus dans ce travail ont montré que :

La technique de quantification scalaire non uniforme proposée permet de favoriser les coefficients les plus significatifs, de sorte que les structures géométriques curvilignes le long des contours et bords des objets présents dans l'image. Cette technique permet au même temps d'améliorer l'entropie des coefficients obtenus et d'écarter les coefficients les moins significatifs.

Ce travail peut faire l'objet d'une extension et d'amélioration sur plusieurs niveaux :

- Utilisation d'autres types de quantificateur,
- Amélioration du temps de calcul,
- Optimisation de résultat par l'algorithme SPIHT.

Bibliographies

- [1] **A.d'hardancourt**, "fou du multimédia", Sybex, 1995
- [2] **K. Chakib**, "La Compression des Images Fixes par les Approximations Fractales Basée sur la Triangulation de Delaunay et la quantification Vectorielle", mémoire de fin d'étude, 1999.
- [3] **M. Derridj** «La compression de données».Cours de module: OSI et protocoles, 06 Octobre 1995.
- [4] **J. Guilois** «Compression Données, Compression des images»
- [5] **James S.Walker** «wavelet-based Image Compression » Departement of Mathematics, University of Wisconsin-Eau Claire.
- [6] **Hervé Gutter** «La compression des images numériques».IEEE Centre Associé de Clermont-Ferrand, pages 25-51. Décembre 1995 .
- [7] **M . Lahdir** «Compression d'image en sous-bandes associant une quantification vectorielle et une quantification en Cosinus, application aux images météo».
- [8] **A. Blehedri et A. Madjouri** «Compression d'image fixes par un codeur emboîté via la DCT » juin 2006.
- [9] **N. Ahmed, T. Natarajan, and K. R. Rao**, "Discrete Cosine Transform", IEEE Trans. Computers, vol. C-23, pp. 90-93, January 1974.
- [10] **K. R. Rao and R. Yip**, "Discrete Cosine Transform - Algorithms, Advantages, Applications", Academic Press, 1990.
- [11] **J. Makhoul**, "A Fast Cosine Transform in One and Two Dimensions", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 28, pp. 27-34, February 1980.
- [12] **Y. Arai, T. Agui, and M. Nakajima**, "A fast dct-sq scheme for images", Transactions of the Institute of Electronics and Communication Engineers of Japan. Section E, vol. 71, no11, pp. 1095-1097, 1988.
- [13] **R. Westwater and B. Furht**, "Real-Time Video Compression: Techniques and Algorithms", Kluwer Academic Publishers 1997.
- [14] **S. Mallat**, "A Wavelet Tour of Signal Processing, Academic Press, 1998.
- [15] **Y. Meyer**, "Wavelets and Operators", Cambridge University Press, 1992.
- [16] **V. Chappellie**, "Codage progressif d'images par ondelettes orientées". Thèse de doctorat, Université de Rennes 1. Déc 2005.

- [17] **J. Radon.**, On the determination of functions from their integral values along certain manifolds, in reports _ Saxon Academy of Sciences, Leipzig Math Nat,1917, vol. 69, pp. 262_277.
- [18] **E. J. Candes and D. L. Donoho**, Curvelets _ a surprisingly effective non adaptive representation for objects with edges, pp. 1_10, Vanderbilt University Press, Nashville, TN, 1999.
- [19] **M. N. Do and M. Vetterli**, The contourlet transform : An efficient directional multiresolution image representation, IEEE Transactions on Image Processing, Oct. 2003.
- [20] **S. Mallat**. Une exploration des signaux en ondelettes. Les éditions de l'école Polytechnique, 2000.
- [21] **F. Yu and A.N. Willson, Jr.**, "A flexible hardware-oriented fast algorithm for motion estimation", ICASSP97, pp.268-288, 1997.
- [22] **E. Candès and D. Donoho**. "Continuous curvelet transform, part II: Discretization and frames", Rapport technique, Department of Statistics,Stanford University, 2003.
- [23] **D. Donoho and M. Duncan**. "Digital curvelet transform : strategy, implementation and experiments". Rapport technique, Department of Statistics, Stanford University, 1999.
- [24] **J.-L. Starck, E. Candès, and D. Donoho**. The curvelet transform for image denoising. IEEE Transactions on Image Processing, 11(6) :131, 2003.
- [25] **E.J. Candes, L. Demanet, D.L. Donoho, L. Ying**, "Fast Discrete Curvelet Transforms" Technical Report, Cal Tech, 2006.
- [26] **S.P. Lloyd**, "Least Squares Quantization in PCM", IEEE Transactions on Information Theory, vol. IT-28, p. 129-137, Mars 1982.
- [27] **J. Max**, "Quantizing for minimum distortion II, IEEE Transactions on Information Theory, p.7-12, Mars 1960.
- [28] **HUFFMAN, DAVID A**. A Method for the Construction of Minimum-Redundancy Codes. s.l. : Proceedings of the I.R.E., September 1952.

Résumé :

Le développement remarquable dans le domaine des technologies de l'information et la diversité des applications multimédias dans les dernières années implique le développement des techniques de compression d'image plus efficaces dont l'objectif est d'améliorer la capacité de transmission et le stockage des données. Les travaux de recherche récents ont montré que les ondelettes classiques ne sont pas en mesure d'exploiter de façon optimale les régularités géométriques le long des contours et aux bords des objets. Nous proposons dans ce travail une nouvelle méthode de compression basée sur des ondelettes géométriques, appelées Curvelets, associée à une technique de quantification scalaire logarithmique, afin d'améliorer les performances de compression, par rapport à celles des ondelettes classiques, du point de vue qualité de compression de l'image .

Cette méthode a donné des résultats en encourageants avec des perspectives intéressantes.

Mots-Clés : Compression d'image, ondelettes géométriques, Curvelets,

Abstract :

The remarkable development in the field of information technology and the diversity of multimedia applications in recent years imply the development of more efficient image compression techniques to improve the data transmission and storage capacity .

The recent researches showed that classical wavelets are not able to exploit optimally the geometric regularities along the contours and edges of objects .

We propose in this work, a new compression method based on geometric wavelets called Curvelets, associated with a logarithmic scalar quantization technique, to improve the compression performances compared to that of classical wavelets, in terms of image quality and compression

This method gave encouraging results with interesting perspectives .

Key-words: Image compression, geometric wavelets, Curvelets, scalar logarithmic quantization,

ملخص:

ان التطور الملحوظ في مجال تقنيات المعلومات و تنوع التطبيقات متعددة الوسائط في السنوات الاخيرة ،يستلزم تطوير تقنيات ضغط الصور اكثر فعالية لتحسين قدرات ارسال و تخزين المعلومات .

اظهرت الابحاث مؤخرا ان الموجات الكلاسيكية غير قادرة على الاستغلال الامثل للانتظام الهندسي على طول معالم و حواف الاجسام .

نقترح في هذا العمل طريقة ضغط جديدة تعتمد على موجات هندسية تسمى كورفلات متبوعة بتقنية تكميم سلمية لوغاريتمية ،و ذلك لتحسين اداء ضغط الصور مقارنة بأداء الموجات الكلاسيكية من حيث جودة الصورة .

اعطت الطريقة المقترحة نتائج مشجعة و افاق بحث اخرى

كلمات مفتاحيه: ضغط الصور،موجات هندسيه, كورفلات، تكميم سلمية لوغاريتمية.