

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

Université Mohamed Khider – Biskra
Faculté des Sciences et de la technologie
Département : Génie Electrique.
Ref :



جامعة محمد خيضر بسكرة
كلية العلوم والتكنولوجيا
قسم: الهندسة الكهربائية
المرجع:

Mémoire présenté en vue de l'obtention
Du diplôme de
Magister en Electronique
Option: *Signaux et communications*

**Compression d'images fixes par Ondelettes
géométriques par utilisation des Curvelets et
différents types d'interpolation dans la
quantification scalaire**

Présenté par :

KADRI Oussama

Soutenu publiquement le : 5 Mars 2014

Devant le jury composé de :

Dr. N.DJEDI	Professeur	Président.	Université de Biskra
Dr. Z.E.BAARIR	MCA	Directeur de mémoire	Université de Biskra
Dr. S.SBAA	MCA	Examineur	Université de Biskra
Dr. A.OUAFI	MCA	Examineur	Université de Biskra
Dr. N.TERKI	MCA	Examineur	Université de Biskra

DEDICACES

À mes chers parents, et toute ma famille, sans lesquels, je ne serai pas ici aujourd'hui.

À la mémoire de mon frère Issam et ma sœur Mounia, que Dieu leurs pardonne et leurs accorde sa miséricorde.

À tous ceux qui m'ont soutenu et m'ont encouragé depuis mes premiers pas dans cette vie.

Je leurs dédie ce modeste travail, en espérant qu'il soit à la hauteur de leurs attentes.

Remerciements

Je remercie tout d'abord Dieu le tout puissant de m'avoir donné le courage, la force et la patience d'achever ce modeste travail.

Je tiens à remercier très chaleureusement, Dr. Baarir Zineddine, qui m'a permis de bénéficier de son encadrement.

Je remercie les membres de jury qui ont accepté de juger ce travail et d'y apporter leur caution :

Professeur Djedi Nour Eddine, qui me fait le grand honneur de présider ce jury.

Dr. Sbaa Salim, Dr. Nadjiba TERKI, et Dr. Ouafi Abdelkrim pour avoir accepté de faire partie de ce jury en tant qu'examineurs.

J'adresse mes vifs remerciements également à tous ceux qui ont participé, de près ou de loin, au bon déroulement de cette recherche.

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 et taux de compression de l'image (PSNR et RC).

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

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

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 rate (PSNR and CR).

This method gave encouraging results with interesting perspectives.

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

ملخص:

إن التطور الملحوظ في مجال تقنيات المعلومات وتنوع التطبيقات متعددة الوسائط في السنوات الأخيرة، يستلزم تطوير تقنيات ضغط صور أكثر فعالية لتحسين قدرات إرسال و تخزين المعلومات.

أظهرت الأبحاث مؤخرًا أن المويجات الكلاسيكية غير قادرة على الاستغلال الأمثل للانتظام الهندسي على طول معالم و حواف الأجسام.

نقترح في هذا العمل طريقة ضغط جديدة تعتمد على مويجات هندسية تسمى Curvelets متبوعة بتقنية تكميم سلمي لوغاريتمي، وذلك لتحسين أداء ضغط الصور مقارنة بأداء المويجات الكلاسيكية من حيث جودة الصورة و نسبة الضغط (PSNR et CR).

أعطت الطريقة المقترحة نتائج مشجعة وآفاق بحث أخرى.

كلمات مفتاحية: ضغط الصور، مويجات هندسية، Curvelets، تكميم سلمي لوغاريتمي، PSNR, RC.

Liste des tableaux :

Tableau IV-1: Résultats obtenues pour les images: « House,Goldhill »	62
Tableau IV-2: Résultats obtenues pour les images: « Sardegna ERS-1, Bridge »	62
Tableau IV-3: Résultats obtenues pour les images: « Buildings, Aerial »	63
Tableau IV-4: Résultats obtenues pour les images: « Barbara, MRI(hd01) »	63

Liste des figures :

Figure I-1: Schéma d'un système de compression par transformation.....	5
Figure I-2: Un banc de filtre d'un seul étage, pour l'analyse et reconstruction de signaux....	12
Figure I-3: L'image « House » en un seul niveau de décomposition en ondelettes 2D.....	13
Figure I-4: Détection des contours et singularités ponctuelles par les ondelettes 2D	14
Figure I-5: Illustration de la directivité du transformée en ondelettes séparables 2D.....	15
Figure II-1: Transformée de Ridgelets d'une image filtrée passe-bande	20
Figure II-2: Des Curvelets à différentes orientations et positions.	22
Figure II-3: Pavage du plan fréquentiel par les Curvelets continus.	25
Figure II-4: Présentation d'un Curvelet de point de vue spatial et fréquentiel.....	26
Figure II-5: Pavage de plan fréquentiel de la transformée en Curvelets numérique	28
Figure II-6: La représentation des bords, par ondelettes et Curvelets.....	33
Figure III-1: Exemple d'une quantification scalaire à 8 niveaux.....	37
Figure III-2: (a) Quantificateur Midtread (b) Quantificateur Midrise	39
Figure III-3: Quantificateur uniforme à zone morte.....	41
Figure III-4: Processus de quantification non uniforme par Companding	43
Figure III-5: Exemples des fonctions de compression et d'expansion	44
Figure III-6: Fonctions à loi de puissance pour ($p=0.2$, $p=0.4$, $p=0.6$).	45
Figure III-7: Fonction de compression M-law ($\mu =1, 10, 255$).....	46
Figure III-8: (a)-Etapas de quantification par Companding. (b)-quantification μ -law à 8 niveaux.....	47
Figure IV-1: Pavage pseudo polaire de plan fréquentiel.	55
Figure IV-2: Transformée de Fourier de l'image Lena représentée sur une échelle logarithmique.....	55
Figure IV-3: Etapas de quantification non uniforme par μ -law inversée.	57
Figure IV-4: Caractéristique de décroissance logarithmique de valeurs absolues des coefficients de Curvelets pour les deux images "Goldhill" et "Bridge".	57
Figure IV-5: Illustration d'une quantification par expansion logarithmique.....	58
Figure IV-6: Méthode de compression proposée.	60
Figure IV-7: Performance de la méthode proposée, pour différentes images de test.....	64
Figure IV-8: Compression par Curvelets de l'image "Sardegna-ERS1".....	65
Figure IV-9: Compression par Curvelets de l'image "Aerial"	65
Figure IV-10: Compression par Curvelets de l'image "Buildings"	65
Figure IV-11: Compression de l'image "House"	68
Figure IV-12: Compression de l'image "Aerial"	68
Figure IV-13: Compression de l'image "Sardegna-ERS1".....	69
Figure IV-14: Comparaison pour l'image "House".....	69
Figure IV-15: Comparaison pour l'image "Aerial"	70
Figure IV-16: Comparaison pour l'image "Sardegna-ERS1"	70
Figure IV-17: Résultats de compression pour l'imge "Barbara".....	72
Figure IV-18: Performance de la méthode proposée contre le standard JPEG2000 pour l'image « Goldhill »	73

LISTE D'ABREVIATIONS

L^2	Espace des fonctions à énergie finie.
DCT	Transformée en cosinus discrète.
TOC	Transformée en ondelettes continue.
TOD	Transformée en ondelettes discrète.
FFT	Transformée de Fourier rapide.
FFT2D	Transformée de Fourier rapide à deux dimensions.
IFFT2D	Transformée de Fourier rapide inverse à deux dimensions.
DRT	Transformée de Ridgelets discret.
DCTG1	Transformée en Curvelets discrètes de première génération.
FDCT	Transformée en Curvelets discrètes rapide.
USFFT	Unequally spaced FFT.
JPEG	Joint Photographic Expert Group.
JPEG2000	Standard de compression d'images fixes récent, introduit par JPEG.
EZW	Embedded zero tree wavelets.
SPHIT	Set Partitioning In Hierarchical Trees.
MPEG4	Moving Picture Experts Group 4.
MSE	Erreur quadratique moyenne.
PSNR	Rapport signal sur bruit crête.
SNR	Rapport signal sur bruit.
RC	Rapport de compression.
Bpp	bit par pixel.
PDF	Fonction de distribution de la probabilité.
Q_{MR}	Quantificateur scalaire midtrise
Q_{MT}	Quantificateur scalaire midtread
RLE	Codage de longueur de séquence.
AC	Codeur arithmétique.
LZW	Lempel Zip Welch.
VTC	Volume Texture Compression
ISO	International Organization for Standardization.
GIF	Graphic Interchange Format.
ITU	International Telecommunication Union

SOMMAIRE

INTRODUCTION GENERALE	1
CHAPITRE I: INTRODUCTION A LA COMPRESSION D'IMAGES FIXES	4
I.1. Introduction :	5
I.2. Etapes de compression par transformation	5
I.2.1 Transformation :	6
I.2.2 Quantification :	8
I.2.3 Codage entropique :	8
I.3. Ondelettes classiques et compression d'images fixes :	9
I.3.1 Transformée en ondelettes continues (TOC)	9
I.3.2 Inversion de transformée en ondelettes continues :	10
I.3.3 Discrétisation de transformée en ondelettes :	11
I.3.4 Extension de transformée en ondelettes pour les images :.....	12
I.3.5 Coefficients d'ondelettes en 2D :.....	13
I.3.6 Compression d'images fixes par ondelettes classiques.....	14
I.4. Conclusion	17
CHAPITRE II: TRANSFORMEE EN CURVELETS	18
II.1. Introduction :	19
II.2. Transformée en Curvelets de 1^{ère} génération :	19
II.2.1 Description :	19
II.2.2 Mise en œuvre numérique.....	21
II.3. Transformée en Curvelets de deuxième génération	23
II.3.1 Transformée en Curvelets continue	23
II.3.2 Transformée en Curvelets discrètes	26
II.3.3 Transformée en Curvelets discrètes via Wrapping	30
II.3.4 Isométrie et transformation inverse	31
II.3.5 Propriétés de transformation en Curvelets.....	23
II.4. Coefficients de Curvelets Vs Coefficients d'ondelettes	33
II.5. Conclusion :	34

CHAPITRE III: QUANTIFICATION ET CODAGE	35
III.1. Introduction	36
III.2. Quantification scalaire	36
III.2.1 Quantification scalaire uniforme	38
III.2.2 Quantification uniforme à zone morte	40
III.2.3 Quantification non uniforme	41
III.2.4 Algorithme de Lloyd-Max	41
III.2.5 Quantification non uniforme par compression (companding)	43
III.2.6 Mu-Law Companding	45
III.3. Quantification et Entropie	47
III.3.1 Codage entropique de Huffman	49
III.3.2 Limites du codage de Huffman	50
III.3.3 Codage arithmétique	50
III.4. Conclusion	52
 CHAPITRE IV: RESULTATS ET DISCUSSIONS	 53
IV.1. Introduction	54
IV.2. Méthodologie	54
IV.2.1 Décomposition en Curvelets	54
IV.2.2 Quantification	56
IV.2.3 Codage entropique	59
IV.3. Schéma en blocs d'algorithme de compression proposée	60
IV.4. Présentation des résultats	61
IV.4.1 Taux de compression Vs PSNR	61
IV.4.2 Taux de compression Vs qualité visuelle	64
IV.4.3 Analyse et discussions	66
IV.5. Comparaison avec d'autres méthodes	67
IV.5.1 Comparaison avec les ondelettes classiques	67
IV.5.2 Travaux similaires	71
IV.5.3 Standards de compression actuels : JPEG 2000 (Ondelettes)	72
IV.6. Conclusion	74
 CONCLUSION GENERALE	 75
 BIBLIOGRAPHIE	 77

INTRODUCTION GENERALE

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. Elle est basée généralement sur une phase de transformation qui sert à compacter l'information utile dans un nombre minimum de coefficients non nuls.

La transformation la plus courante est celle par ondelettes classiques qui est à la base des standards de compression les plus récents comme le JPEG2000. Cependant, elle présente des limites dans la représentation des régularités géométriques. Ceci étant dû au fait que les bases de transformée en ondelettes bidimensionnelle sont obtenues par le produit tensoriel de deux ondelettes unidimensionnelles appliquées suivant les lignes et les colonnes de l'image.

De nouvelles méthodes fondées sur l'amélioration des ondelettes classiques, en exploitant la corrélation statistique intra ou inter-échelles entre les coefficients d'ondelettes ont été introduites, telles que le modèle d'arbre de zéros (zero-tree), ce qui a été largement adopté dans diverses applications telles que l'EZW¹ et le SPIHT² [1] [2]. Bien que ces méthodes soient en mesure d'exploiter certaines quantités de la régularité géométrique, l'utilisation de véritables représentations géométriques semble être plus optimale pour un large éventail de classes d'images.

Dans ce mémoire, nous avons fixé comme objectif l'exploitation de potentiel d'une transformée multi-résolutions directionnelle, plus précisément, la transformée en Curvelets, pour améliorer les performances de la compression d'images fixes (PSNR et RC) par rapport aux performances des ondelettes classiques. Le travail effectué est consacré à la phase de quantification qui représente l'étape la plus importante dans l'algorithme de compression d'image avec pertes.

Pour ce faire, nous proposons un système basé sur l'utilisation de la transformée en Curvelets et une nouvelle méthode de quantification qui prend en considération les caractéristiques de ses coefficients.

Ce mémoire est organisé comme suit :

Le premier chapitre présente les concepts généraux de compression d'images fixes, les étapes principales de l'algorithme de compression avec pertes ainsi qu'une brève présentation des ondelettes classiques, qui sont à la base des standards de compression les plus récents, en montrant leurs limites dans la représentation des singularités géométriques.

Nous introduisons dans le deuxième chapitre, les Curvelets, leurs propriétés principales, et les techniques d'implémentation numérique de la transformée en Curvelets, ainsi que ses avantages, qui font des Curvelets un outil intéressant pour améliorer les ondelettes classiques dans la représentation des informations géométriques des images.

¹ Embedded Zerotree Wavelet (EZW) est une technique de codage très efficace et simple, introduite par JM Shapiro, pour la compression d'image, basée sur l'exploitation de l'autosimilarité inhérente aux images.

² Set Partitioning In Hierarchical Trees (SPIHT) présenté par A. Said et WA Pearlman, offre une explication alternative des principes de fonctionnement de l'EZW.

Dans le troisième chapitre, nous étudions les techniques de quantification scalaire les plus utilisés, dans la compression d'images, en démontrant l'effet de cette étape sur la phase de codage entropique.

Le dernier chapitre présente la méthode de compression proposée, basée sur une transformation en Curvelets, associée à une nouvelle technique de quantification non-uniforme, adaptée aux caractéristiques des coefficients de transformée en Curvelets. Les résultats de simulation sur un ensemble d'images de test sont ensuite présentés avec des discussions afin d'évaluer les performances des points de vue taux de compression et de qualité des images reconstruites.

Le mémoire s'achève par une conclusion générale et quelques perspectives

*CHAPITRE I: INTRODUCTION A LA
COMPRESSION D'IMAGES FIXES.*

I.1. INTRODUCTION :

La compression d'images est au cœur d'avancement des technologies de l'information. Elle est utilisée dans la majorité des standards de communications pour réduire le nombre de bits nécessaire pour la représentation des données volumineuses telles que les images ou les séquences vidéo. Dans ce chapitre, nous introduisons, dans un premier temps, les principes de la compression d'images fixes avec pertes, et les différentes étapes du schéma standard de sa construction.

Ensuite, nous présentons les ondelettes classiques qui sont à la base des standards de compression les plus récents, en mettant l'accent sur leurs limites dans la représentation de régularités géométriques, de manière à situer le cadre général de ce mémoire.

I.2. ETAPES DE COMPRESSION PAR TRANSFORMATION

Le schéma d'un système de compression classique est représenté sur la figure I.1. La première étape est la transformation. Elle permet d'exploiter les redondances statistiques de l'image. La deuxième étape est celle de la quantification, c'est l'étape la plus critique puisque c'est l'étape où la perte d'information a lieu. La dernière étape est le codage des données quantifiées. Elle permet d'organiser le train binaire en compressant les données sans altérer l'information.

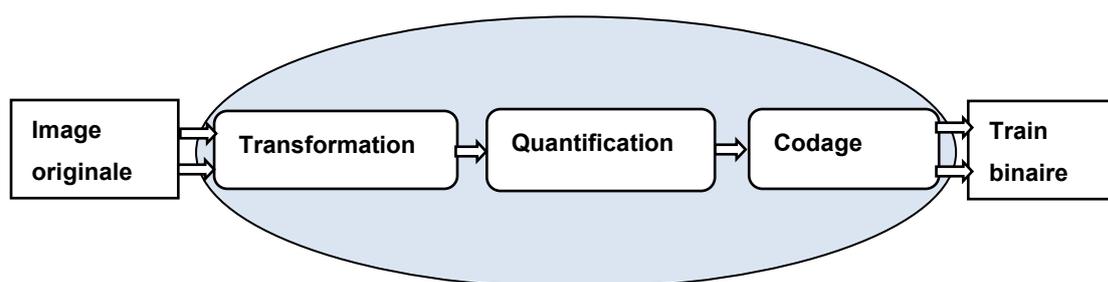


Figure I-1: Schéma d'un système de compression par transformation

1.2.1 TRANSFORMATION :

Elle a comme but la dé-corrélation de l'information. Elle permet de minimiser les redondances statistiques présentes dans l'image, et de concentrer l'information pertinente dans un petit nombre de coefficients significatifs. Ce qui permet d'attribuer plus de bits à ces coefficients lors des phases de quantification et de codage. Cette étape est réversible et sans pertes ou presque.

Les recherches menées dernièrement sur la compression par transformation tendent à montrer que l'adaptation de la transformée aux caractéristiques locales de l'image permet une amélioration notable des performances. En pratique, l'optimisation de la transformée peut être réalisée à deux niveaux :

- Dans le domaine spatial, en adaptant le support de la transformée.
- Dans le domaine transformé, en adaptant les atomes de la base de projection aux caractéristiques du signal que l'on cherche à décrire.

L'adaptation du support de la transformée exploite l'idée d'appliquer la transformation sur des blocs de l'image plutôt que sur l'image en entier. Cette approche peut, en effet, non seulement avoir un intérêt en termes de coût du calcul, mais également en termes de performance débit-distorsion.

Plusieurs techniques peuvent être considérées. La plus simple est le découpage de l'image en blocs de taille fixe, par exemple, de taille 8×8 pixels comme dans le format de compression JPEG.

Parallèlement, de nouvelles transformées ont émergées, tendant vers une meilleure prise en compte des caractéristiques de l'image. La transformée en ondelettes utilisée dans le standard de compression d'images JPEG2000 [3] constitue un exemple typique pour ces transformées. Depuis leur introduction, les ondelettes ont fait l'objet de beaucoup d'attention dans des domaines aussi diversifiés que le dé-bruitage, la compression, le codage, l'imagerie médicale ou satellitaire.

Récemment, il est apparu évident que les transformations usuelles de type séparable, telles que la transformée en ondelettes, semblent former une catégorie restreinte et limitée des possibilités de représentations des images et des signaux multidimensionnels. Cette limitation est dûe, en premier lieu, au fait que de telles représentations ne prennent pas en compte la régularité des structures géométriques dans l'image. Comprendre comment représenter les structures géométriques complexes est devenu donc un enjeu important pour améliorer l'état de l'art du traitement de l'image.

Se basant sur cette observation, des travaux récents montrent qu'il est possible de définir des cadres théoriques de représentations multi-échelles, plus vastes, donnant naissance à de nouvelles transformées directionnelles plus adaptées à l'extraction de structures géométriques lisses et continues, tels que les contours et bords d'objets.

Ces transformées peuvent être classées en deux catégories [4]:

Transformées adaptatives :

Elles cherchent à définir une base dont les fonctions sont choisies au mieux pour s'adapter à une image donnée. Pour cela, une étape préalable d'estimation de la géométrie de l'image est nécessaire avant de procéder à la décomposition. De nombreuses représentations géométriques adaptatives ont été étudiées ces dernières années pour les applications du traitement d'images, comme les bandelettes et les Wedgelets.

Transformées non adaptatives :

Elles utilisent des bases de projection fixes, mais elles cherchent à corriger l'aspect isotrope de transformées en ondelettes classiques. Ces approches ne dépendent pas de l'image à analyser. Elles ont donc le mérite de ne pas nécessiter de surcoût pour spécifier, lors de la synthèse, la configuration utilisée à l'analyse. À titre d'exemple, on peut citer : la transformée en Ridgelets, Curvelets ou Contourlets.

1.2.2 QUANTIFICATION :

C'est la seule étape non réversible du schéma de compression. Dans de nombreuses applications de compression avec perte, nous sommes tenus de représenter le signal source par un petit nombre de mots de code. Généralement, le nombre de valeurs distinctes possibles de signal est beaucoup plus grand que le nombre de mots de code disponibles pour les représenter. Le processus de représenter un grand ensemble (éventuellement infini) de valeurs par un ensemble beaucoup plus petit est appelé quantification.

La quantification peut être vectorielle ou scalaire. La quantification vectorielle permet de quantifier simultanément plusieurs coefficients. Elle est coûteuse en termes de complexité calculatoire et n'est utile que si les données ne sont pas décorréliées. La quantification scalaire est moins complexe et plus efficace si elle est précédée par une transformation, bien adaptée aux caractéristiques de l'image, qui génère des coefficients quasi indépendants. On étudie ce type de quantification avec plus de détails dans le chapitre III.

1.2.3 CODAGE ENTROPIQUE :

Le codage entropique permet la compression des coefficients quantifiés en générant un train binaire de longueur proche de la limite théorique basse défini par l'entropie de Shannon. Plusieurs types de codage entropique sont utilisés pour la compression d'images, nous citons à titre d'exemple :

- Le codage par longueur de séquences RLE utilisé dans le standard JPEG [3].
- Le codage par dictionnaire, comme le codeur LZW utilisé dans le format d'image GIF.
- Le codage par statistique, comme le codage de Huffman utilisé dans le format JPEG, ou le codage arithmétique effectué dans le standard JPEG2000 par le codeur EBCOT [3].

I.3. ONDELETTES CLASSIQUES ET COMPRESSION D'IMAGES FIXES :

La théorie sur laquelle repose la technologie des ondelettes remonte au début des années 1980. Les ondelettes ont largement influencé le domaine de traitement du signal et en particulier l'analyse et la compression d'images. L'une des caractéristiques essentielles des ondelettes est leur grande précision spatiale à petite échelle. Cette propriété est intrinsèquement liée au mode de construction des ondelettes. Dans ces conditions, il est donc naturel que les ondelettes puissent être utilisées pour la caractérisation des propriétés de régularité des fonctions, que ce soit la régularité locale ou la régularité globale.

Cette caractéristique est très utile pour la compression d'image, elle permet de distinguer les zones d'intense activité des régions plates. Nous pouvons être en mesure d'allouer un nombre différent de bits de quantification à ces différentes régions, réalisant ainsi une compression élevée sans sacrifier la qualité visuelle.

Dans ce qui suit, nous présentons brièvement la transformation en ondelettes, et le comportement de ses coefficients envers les régularités et singularités dans les images, ainsi que ses limites qui ont provoqué les chercheurs dans le domaine de traitement d'images à développer une autre génération d'ondelettes, dite les ondelettes géométriques (Ridgelets, Curvelets, ... etc.).

I.3.1 TRANSFORMEE EN ONDELETTES CONTINUES (TOC)

La transformée en ondelettes continues (TOC) permet la représentation des fonctions de L^2 dans une base bien localisée, dans les domaines spatial et fréquentiel, permettant d'exploiter les avantages de l'analyse de Fourier tout en s'affranchissant des inconvénients de son manque de localisation.

Les coefficients de (TOC) sont le résultat du produit scalaire d'un signal original $x(t)$ et des fonctions d'analyse temps-échelle $\psi(a, b)$ qui résulte de la dilatation et la translation d'un même motif élémentaire appelé *ondelette-mère* ψ .

Cette approche, évalue plusieurs fois la même information et conduit à un coût de calcul relativement élevé, typiquement en $M \times N^2$ opérations pour un signal en N points décomposés en M échelles. L'utilisation systématique du FFT pour évaluer ces produits scalaires réduit ce coût à un ordre de $M \cdot N \cdot \log_2(N)$ [5].

La transformée en ondelettes continues peut-être alors définies par [5]:

$$\begin{aligned} L^2(\mathbb{R}) &\xrightarrow{TOC} L^2(\mathbb{R}) \\ x(t) &\xrightarrow{TOC} C(a, b) = \int x(t) \cdot \Psi_{(a,b)}^*(t) \cdot dt = \langle x, \Psi_{(a,b)} \rangle \end{aligned} \quad \text{(EQ : I-1)}$$

Avec : $\Psi_{(a,b)}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)$ Où **a** et **b** sont respectivement les facteurs de dilatation en échelle et translation en temps. $C(a, b)$ est le coefficient d'ondelettes de la fonction $x(t)$.

$\Psi_{(a,b)}$ est la fonction d'ondelette analysante, et $\Psi_{(a,b)}^*$ sa conjuguée complexe.

1.3.2 INVERSION DE TRANSFORMEE EN ONDELETES CONTINUES :

La transformée en ondelettes continues (TOC) est une transformation inversible qui conserve l'énergie du signal si l'ondelette vérifie la condition d'admissibilité suivante [6] :

$$\int_0^{+\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} \cdot d\omega < \infty \quad \text{(EQ : I-2)}$$

Si ψ vérifie la condition ci-dessus, alors l'ondelette est dite admissible. On peut remarquer que la condition d'admissibilité impose nécessairement que l'ondelette soit de moyenne nulle [6] :

$$\int_{-\infty}^{+\infty} \psi(t) \cdot dt = 0 \quad \text{(EQ : I-3)}$$

Supposant $\psi \in L^2(\mathbb{R})$ une ondelette admissible, la fonction originale $f \in L^2(\mathbb{R})$ peut être reconstruite à partir de ses coefficients d'ondelettes en utilisant la relation suivante :

$$f(t) = \int_0^{+\infty} \int_{-\infty}^{+\infty} C_{(a,b)} \cdot \Psi_{(a,b)}(t) \frac{da \cdot db}{a^2} \quad \text{(EQ : I-4)}$$

De plus, le TOC conserve l'énergie du signal. Les coefficients d'ondelettes vérifient la relation suivante :

$$\int_{-\infty}^{+\infty} |f(t)|^2 \cdot dt = \int_0^{+\infty} \int_{-\infty}^{+\infty} |C_{(a,b)}|^2 \cdot \frac{da \cdot db}{a^2} \quad \text{(EQ : I-5)}$$

1.3.3 DISCRETISATION DE TRANSFORMEE EN ONDELETTES :

La transformée en ondelettes continues présentée précédemment est obtenue en prenant le facteur d'échelle a et le pas de translation b dans l'ensemble des nombres réels. Ce type de transformation ne peut être effectué dans la pratique que de façon approximative, et il y a toujours, en fait, une discrétisation du calcul, qui est opérée.

J. Morlet a formulé des bases construites par une discrétisation dyadique de ces paramètres sur le modèle suivant [5] :

$$a = 2^{-j}, \text{ et } b = k \cdot 2^{-j} \text{ avec : } j = 1, 2, \dots, j-1 \text{ et } k = 1, 2, \dots, 2^{-j} - 1.$$

Et l'ensemble des fonctions d'ondelettes analysantes seront donc :

$$\psi_{(a,b)}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) = \psi_{(j,k)}(t) = 2^{j/2} \cdot \psi(2^j t - k) \quad (\text{EQ : I-6})$$

La transformée en ondelettes discrètes (TOD) de la fonction $f(t)$ est donc en fonction de j et k , au lieu de a et b , respectivement, et entraîne un ensemble de coefficients d'ondelettes (détail) :

$$C_{j,k} = \int_{-\infty}^{+\infty} f(t) \cdot \psi_{j,k}(t) \cdot dt \quad (\text{EQ : I-7})$$

Comme les fonctions d'ondelettes, il y a encore un autre ensemble de fonctions appelées fonctions d'échelle $\varphi_{j,k}(t)$ qui donnent par convolution avec $f(t)$, l'ensemble des coefficients d'approximation :

$$A_{j,k} = \int_{-\infty}^{+\infty} f(t) \cdot \varphi_{j,k}(t) \cdot dt \quad (\text{EQ : I-8})$$

Les fonctions d'ondelettes (détail) et d'échelles (approximation) établissent un algorithme de décomposition multi-résolutions. La fonction d'ondelette est orthogonale à la fonction d'échelle à un indice d'échelle particulière j . Ainsi, les informations contenues dans les coefficients d'approximation d'un indice d'échelle j ne sont pas répétées dans les coefficients d'ondelettes. De plus, les coefficients d'ondelettes à un indice d'échelle donnée sont en fonction de la fonction d'échelle de niveau inférieur. Par conséquent, les fonctions d'approximation à un indice d'échelle donnée peuvent être reconstruites en utilisant la fonction d'approximation et les coefficients de détail de l'indice supérieur.

$$f_j(x) = C_{j+1}(x) + A_{j+1}(x) \tag{EQ : I-9}$$

Ainsi, les fonctions d'ondelettes et d'échelle ont la capacité d'exprimer une fonction dans une résolution inférieure. La fonction de décomposition de l'équation précédente peut s'écrire sous la forme :

$$\begin{aligned} f_0(x) &= C_1(x) + A_1(x) = C_2(x) + A_2(x) + A_1(x) \\ &= C_3(x) + A_3(x) + A_2(x) + A_1(x) \end{aligned} \tag{EQ : I-10}$$

La transformée en ondelettes discrètes (TOD) est devenue un outil très polyvalent de traitement de signal, après l'introduction de la représentation multi-résolutions des signaux basée sur la décomposition en ondelettes en 1987.

Stéphane Mallat a mis en avant une certaine catégorie de décompositions en ondelettes, qui peuvent être réalisées numériquement en un temps très court, par « une transformée en ondelettes rapide », constitué d'une cascade de filtres passe-bas et passe-haut suivis par des opérations de sous échantillonnages par un facteur de deux.

Cette approche réduit considérablement la complexité de la construction d'ondelette. Au lieu de choisir une fonction, on choisit l'ensemble discret des coefficients des deux filtres.

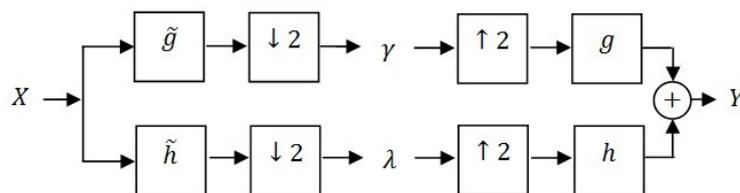


Figure I-2: Un banc de filtre d'un seul étage, pour l'analyse et reconstruction de signaux.

1.3.4 EXTENSION DE TRANSFORMEE EN ONDELETTES POUR LES IMAGES :

Il existe plusieurs façons pour étendre la transformation en ondelettes pour un signal 2D, tel que l'image. Le procédé classique consiste à obtenir la fonction d'échelle 2D $\phi(x,y)$ et la fonction d'ondelette $\psi(x,y)$ par le produit tensoriel des fonctions d'ondelettes 1D $\psi(x), \psi(y)$ et les fonctions d'échelle $\phi(x), \phi(y)$.

La fonction d'ondelette mère 2D correspondante permet d'obtenir les détails dans trois directions, horizontale, verticale et diagonale :

$$\psi^V(x, y) = \phi(x)\psi(y) \quad (\text{EQ : I-11})$$

$$\psi^H(x, y) = \psi(x)\phi(y) \quad (\text{EQ : I-12})$$

$$\psi^D(x, y) = \psi(x)\psi(y) \quad (\text{EQ : I-13})$$

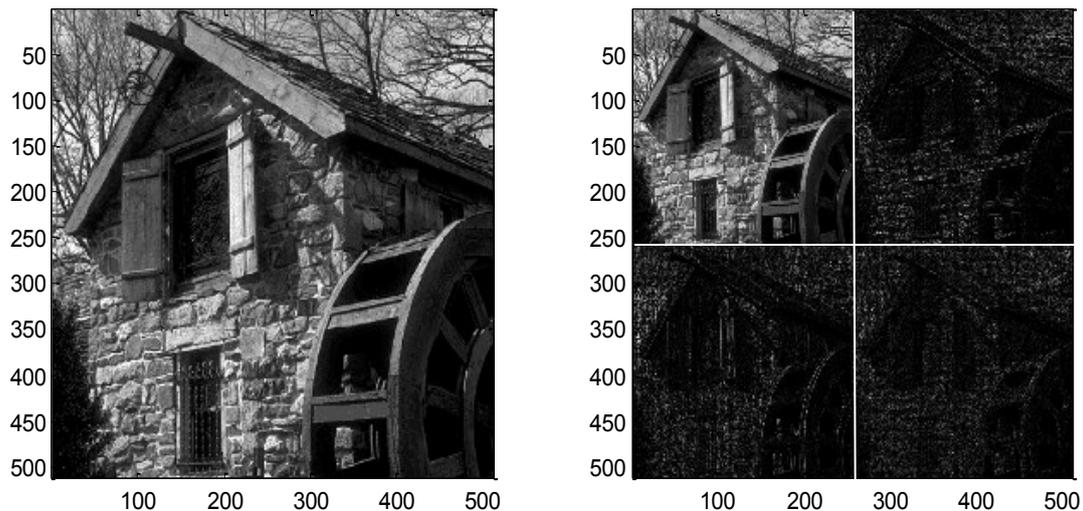


Figure I-3: L'image « House » en un seul niveau de décomposition en ondelettes 2D.

I.3.5 COEFFICIENTS D'ONDELETTES EN 2D :

L'ensemble des coefficients d'échelle et de détails en 2D est de même taille que le nombre de pixels de l'image originale et peut se représenter sous la forme d'une image organisée de façon hiérarchique (voir Figure I-3) dont les pixels les plus blancs représentent les coefficients de grandes amplitudes, alors que les pixels noirs représentent les coefficients d'ondelettes les moins significatifs.

La Figure I-4 donne un exemple de décomposition en ondelettes d'une image à niveaux de gris de taille 512 × 512 pixels. On peut constater qu'aux petites échelles, les coefficients d'ondelettes de grande amplitude se concentrent au voisinage des points où l'intensité des pixels des images change rapidement. Ils correspondent soit aux contours horizontaux, verticaux ou diagonaux de l'image.

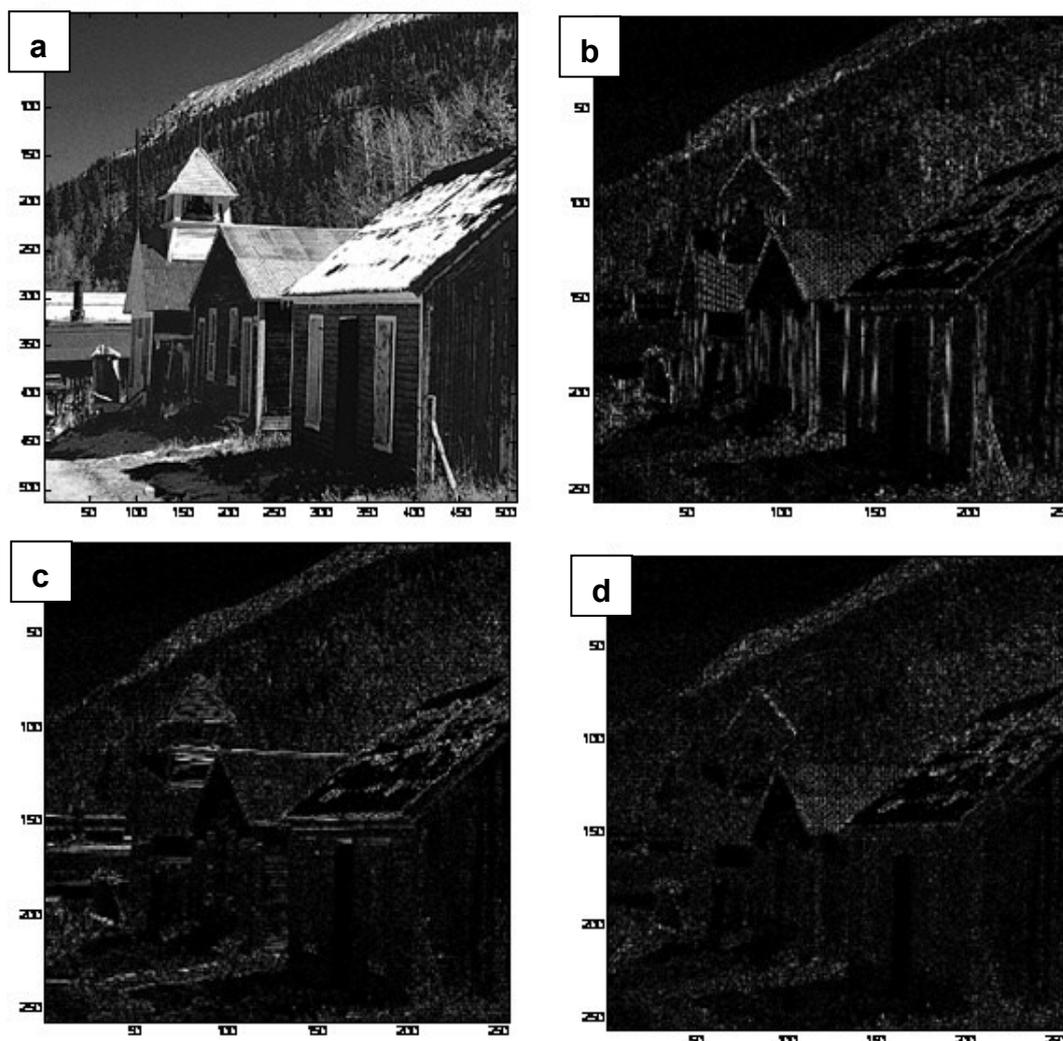


Figure I-4: Détection des contours et singularités ponctuelles par les ondelettes 2D :
(a) Image originale, (b) coefficients détails verticaux, (c) coefficients détails
horizontaux, (d) coefficients détails diagonaux.

1.3.6 COMPRESSION D'IMAGES FIXES PAR ONDELETTES CLASSIQUES

La popularité des ondelettes dans divers domaines scientifiques et techniques est due à son efficacité dans la représentation des singularités ponctuelles dans les signaux unidimensionnels, à cause du support compact des fonctions de bases de décomposition, ce qui implique que si on a un signal 1-D dont l'énergie est concentrée en un point (discontinuité ponctuelle), nous n'aurons que quelques coefficients d'ondelettes significatifs à chaque échelle.

Pour les signaux 2D tels que les images, le problème devient beaucoup plus complexe, à cause de la présence d'autre type de régularités le long des bords et contours des objets.

Les ondelettes séparables 2-D obtenues par le produit tensoriel de deux ondelettes 1D, horizontal et vertical, ne peuvent pas détecter les régularités géométriques, qui ne sont pas le long des directions horizontales, verticales ou diagonales. Ainsi, les coefficients obtenus présentent certaines corrélations inter ou intra-échelles. La Figure I-5 illustre la décomposition d'une forme élémentaire 2-D, suivant une base d'ondelettes séparables.

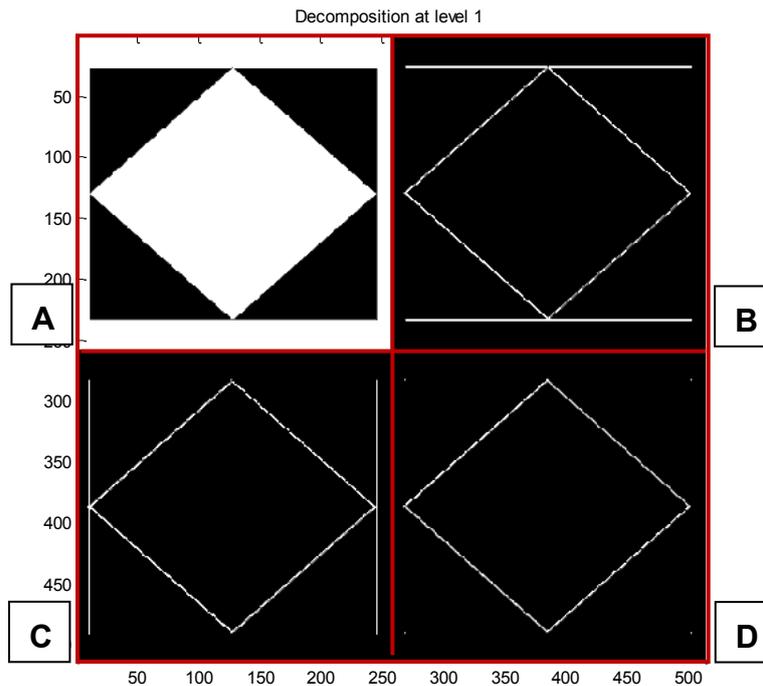


Figure I-5: Illustration de la directivité du transformée en ondelettes séparables 2D (décomposition d'une forme élémentaire)

(A) Image originale, (B) coefficients détails horizontaux, (C) coefficients détails verticaux, (D) coefficients détails diagonaux.

L'efficacité de la représentation est un facteur important dans la transformée en ondelettes 2D. Elle peut se référer à la capacité de capturer les informations importantes à propos d'un objet, avec le moins possible de coefficients non nuls. Ce facteur est de grande utilité dans la compression d'images. De ce fait, un grand nombre de méthodes de compression d'images, parmi les plus récentes et les plus efficaces, prennent appui sur les ondelettes. La plupart de ces méthodes sont basées sur l'exploitation des redondances entre les coefficients obtenus par une transformation en ondelettes.

On peut distinguer les méthodes qui essayent d'exploiter les redondances entre les différentes échelles de décomposition en ondelettes telle que les méthodes de type arbre de zéros (Zero-Tree), et les méthodes qui exploitent les redondances intra-échelle, comme le codage par paquets d'ondelettes ou l'EBCOT utilisé dans le standard JPEG 2000.

Parmi les systèmes de compression basée sur la décomposition en ondelettes, on peut citer:

*a) **Standard MPEG4 :***

Le standard de compression et de manipulation vidéo MPEG4 comprend un volet de codage d'images fixes. Il s'agit de l'algorithme VTC (Visual Texture Coding). L'algorithme est basé sur une transformée en ondelettes discrète dyadique dont les filtres sous-bandes sont les filtres bi-orthogonaux de Daubechies (9,3). La quantification des coefficients est scalaire et les coefficients quantifiés sont codés par une variante de l'algorithme EZW de Shapiro.

*b) **Standard JPEG 2000 :***

La norme JPEG2000 est le dernier standard ISO/ITU-T pour le codage des images fixes. Elle est basée sur une partition de l'image en blocs de codage carrés, chaque bloc est ensuite décomposé en sous bandes d'échelles par le transformée en ondelettes. Sur chacune des sous-bandes, on utilise des critères de débit-distorsion pour déterminer le pas de quantification optimal.

Le standard JPEG 2000 utilise un quantificateur scalaire uniforme à zone morte qui permet de se débarrasser des coefficients moins significatifs en les quantifiant à zéro. Le codage entropique est par la suite effectué sur chaque bloc par un codeur arithmétique adaptatif. Ainsi, les relations statistiques entre les coefficients de chaque bloc d'une sous bande sont exploitées.

I.4. CONCLUSION

La compression d'image par ondelettes est maintenant bien développée pour les algorithmes de compression d'images. On la trouve dans le dernier standard de compression JPEG2000. Ces algorithmes ont prouvé leur capacité à détecter les singularités horizontales, verticales, et diagonales. Cependant, les ondelettes ne sont pas en mesure de bien représenter les régularités géométriques le long des bords des objets. Ceci étant dû au fait que les bases de transformée en ondelettes bidimensionnelle sont obtenues par le produit tensoriel de deux ondelettes unidimensionnelles appliquées suivant les lignes et les colonnes de l'image.

Pour remédier à ce problème, il est recommandé d'utiliser des transformations capables d'exploiter efficacement les régularités géométriques. Le chapitre suivant présente un type de ces transformations nommé les Curvelets.

CHAPITRE II:
TRANSFORMEE EN CURVELETS

II.1. INTRODUCTION :

Tandis que les ondelettes classiques sont à la base des standards de compression les plus récents, elles ne sont pas en mesure d'exploiter complètement les régularités géométriques qui existent le long des contours et bords des objets dans l'image. Par conséquent, l'utilisation d'autres transformées plus adaptées à ces structures, comme la transformée en Curvelets, semble plus appropriée pour la conception d'un schéma de compression plus performant.

Dans ce chapitre, nous décrivons le principe de la transformée en Curvelets et sa mise en œuvre numérique, ainsi que ses principales propriétés, qui leur permettent de représenter les régularités géométriques de façon plus efficace.

II.2. TRANSFORMEE EN CURVELETS DE 1^{ERE} GENERATION :

II.2.1 DESCRIPTION :

En 1999, la transformée en Ridgelets a été proposée par Candès et Donoho [7] [8] où les éléments de base sont des ondelettes géométriques anisotropes. Cette transformée dotée par une sélectivité directionnelle arbitraire fournit une clé pour l'analyse des singularités à hautes dimensions. Malheureusement, la transformation en Ridgelets ne s'avère efficace que dans la représentation des objets avec singularités ou contours rectilignes, qui sont rarement observés dans les applications réelles. Afin d'analyser les singularités curvilignes ou linéaires locales, une idée naturelle est de partitionner l'image, puis d'appliquer la transformation Ridgelets aux sous-images obtenues. Cette transformation basée, sur des blocs Ridgelets, qui a été d'abord proposée par Candès et Donoho en 2000 [9], est nommée Curvelets.

La transformée de Curvelet de première génération [10], [11] ouvre la possibilité d'analyser une image par des blocs de différentes tailles, mais avec une transformation unique.

L'idée est de décomposer l'image en un ensemble de bandes d'ondelettes et d'analyser chaque bande par une Ridgelet local. Aux échelles fines, les courbatures ressemblent à des lignes droites. Le principe est illustré sur la Figure II-1.

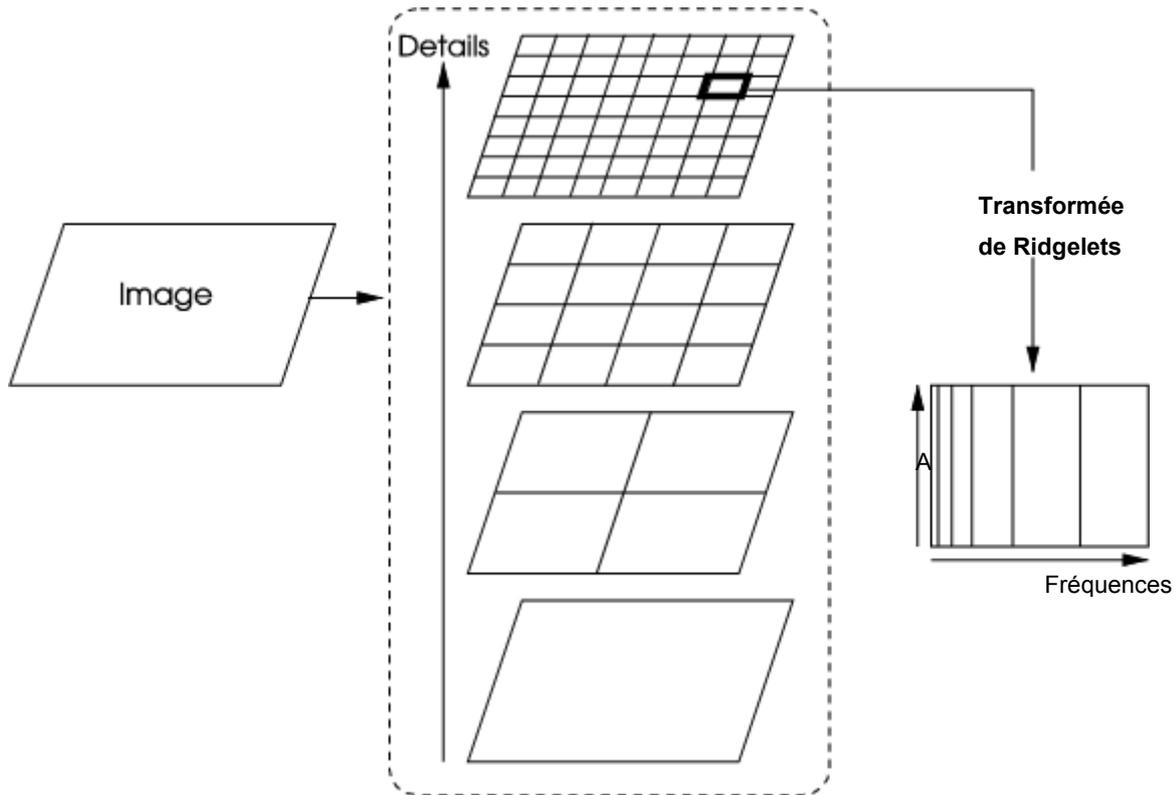


Figure II-1: Transformée de Ridgelets d'une image filtrée passe-bande

La taille de bloc peut être changée à chaque niveau d'échelle. Globalement, les différents niveaux de la pyramide Ridgelets multi-échelles sont utilisés pour représenter les différents sous-groupes d'une sortie de banc de filtres.

En même temps, cette décomposition en sous-bande impose une relation entre la largeur et la longueur des éléments analysants afin qu'ils soient anisotropes et obéissent, approximativement à la loi d'échelle parabolique $Largeur \approx longueur^2$.

La transformée en Curvelets discrètes de première génération (DCTG1), d'une fonction continue $f(t)$, utilise une séquence dyadique d'échelles et un banc de filtres avec la propriété que le filtre passe-bande j est concentrée à proximité des fréquences $[2^{2j}, 2^{j+2}]$ ce qui est:

$$\Delta_j(f) = \Psi_{2j} * f, \text{ avec } \hat{\Psi}_{2j}(\omega) = \hat{\Psi}(2^{-2j}\omega) \quad (\text{EQ : II-1})$$

Dans la théorie des ondelettes, on utilise une décomposition en sous-bandes dyadiques $[2^j, 2^{j+1}]$. En revanche, les sous-bandes utilisées dans le DCTG1 ont la forme non standard $[2^{2j}, 2^{j+2}]$. Cette fonctionnalité non standard du DCTG1 est la caractéristique principale des Curvelets.

La décomposition DCTG1 est la séquence des étapes suivantes :

- Décomposition en sous bande : L'image f est décomposée en sous-bandes.
- Partitionnement lisse : Chaque sous-bande est fenêtrée en « carrés » de tailles dépendantes de l'échelle (longueur de côté $\sim 2^{-j}$).
- L'analyse Ridgelets : Chaque carré est analysé par l'intermédiaire du DRT.

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

II.2.2 MISE EN ŒUVRE NUMERIQUE

Cette approche particulière a été appelée, par la suite, la transformée de Curvelet-99 et une implémentation numérique a été rapportée par D. Donoho et M. Duncan [11] en utilisant la transformée en ondelettes non décimées à trous qui a un facteur de redondance élevé de $16^J + 1$, où J est le nombre total d'échelles dans la décomposition. Cette mise en œuvre a été utilisée dans plusieurs application tel que : le dé-bruitage [12], l'amélioration de contraste [13] et la fusion d'image [14]. L'algorithme décompose une image discrète $f[i_1, i_2]$ de taille $n \times n$ comme une superposition de la forme :

$$f[i_1, i_2] = c_j[i_1, i_2] + \sum_{j=1}^J \mathcal{W}_j[i_1, i_2] \quad (\text{EQ : II-2})$$

Où C_j est la version grossière où la version lisse de l'image originale f et \mathcal{W}_j représente les détails de f au niveau de résolution j . Ainsi l'algorithme génère $J + 1$ tableaux de sous-bandes chacun de taille $N \times N$.

L'algorithme Curvelet-99 [15] de calcul de DCTG1 d'une image $f (N \times N)$, se basant sur la DRT, peut être décrit comme suit :

Paramètres : nombre d'échelles J , taille minimale de bloc B_{\min} .

- 1) Appliquer la transformée en ondelettes à trous sur l'image f et obtenir l'ensemble $\mathcal{W} = \{w_1, \dots, w_J, c_J\}$.
- 2) Initialiser $B_1 = B_{\min}$. Puis, pour chaque échelle $j = 1 \dots, J$:
 - Partitionner la sous-bande w_j , par des blocs de taille B_j , et appliquer le DRT sur chaque bloc, on obtient les coefficients de Curvelets α_j .
 - Si $J \text{ modulo } 2 = 1$, alors $B_{j+1} = 2 B_j$, sinon $B_{j+1} = B_j$.
- 3) L'ensemble des coefficients de Curvelets DCTG1 de l'image f :

$$C = \{\alpha_1, \alpha_2, \dots, \alpha_J, c_J\}.$$

La longueur du côté des blocs B est doublée à chaque sous-bande dyadique. Donc, le maintien de la propriété fondamentale du transformée en Curvelets qui dit que les éléments d'une longueur d'environ $2^{-j/2}$ servent pour l'analyse et la synthèse du $j^{\text{ème}}$ sous-bande $[2^j, 2^{j+1}]$.

L'algorithme Curvelet-99 bénéficie d'une reconstruction exacte et stable parce que, chaque étape d'analyse dans l'algorithme du transformée est elle même inversible. La figure II-2 montre quelques Curvelets à différentes échelles, orientations et positions.

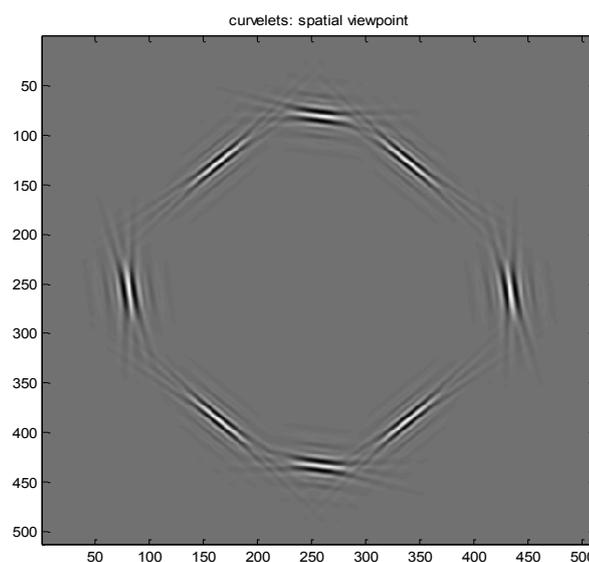


Figure II-2: Des Curvelets à différentes orientations et positions.

II.3. TRANSFORMEE EN CURVELETS DE DEUXIEME GENERATION

Malgré ses propriétés intéressantes, la transformée en Curvelets de première génération présente quelques inconvénients. Sa répartition spatiale utilise des fenêtres superposées pour éviter les effets des blocs. Cela conduit à une augmentation de la redondance et du coût de calcul de l'algorithme DCTG1, ce qui peut être une limitation pour les données volumineuses.

Ultérieurement, une beaucoup plus simple transformation en Curvelets basée sur une technique de pavage du plan fréquentiel a été proposée par E.Candès, L. Demanet et al [16], [17]. Elle est plus rapide et moins redondante que la version de première génération.

Dans cette nouvelle version de la transformée en Curvelets, la transformation Ridgelets a été éliminée, ce qui a réduit les redondances et augmenté la vitesse considérablement. La transformation en Curvelets de deuxième génération est définie à la fois dans le domaine continu et le domaine discret.

II.3.1 TRANSFORMEE EN CURVELETS CONTINUE

La transformée en Curvelets continue, introduite par E. Candès et D. Donoho [10], forme un support compact de $L^2(\mathbb{R}^2)$. Elle est construite par un pavage du domaine fréquentiel par des fenêtres radiales $W(r)$ et des fenêtres angulaires $V(t)$. Ces fenêtres $W(r)$ et $V(t)$ doivent être toutes les deux régulières, non négatives et à valeurs réelles, et leurs supports sont respectivement $r \in (1/2, 2)$ et $t \in [-1, 1]$.

$W(r)$ et $V(t)$ doivent vérifier aussi les conditions d'admissibilité suivantes :

$$\sum_{j=-\infty}^{+\infty} W^2(2^j r) = 1, r \in (3/4, 3/2) \quad \text{(EQ: II-3)}$$

$$\sum_{j=-\infty}^{+\infty} V^2(t - \ell) = 1, t \in (-1/2, 1/2) \quad \text{(EQ: II-4)}$$

Ces fenêtres forment pour chaque échelle des fenêtres fréquentielles sous forme de secteur polaire, avec des largeurs de coté dépendant de l'échelle dans chaque direction. Elles sont définies par :

$$U_j(r, \theta) = 2^{-3j/4} \cdot W(2^{-j} r) \cdot V\left(\frac{2^{|j/2|} \theta}{2\pi}\right) \quad \text{(EQ: II-5)}$$

Où $[j/2]$ est la partie entière de $j/2$.

La forme d'onde $\varphi_j(x)$ définie dans le plan de Fourier par $\varphi_j(\omega) = U_j(\omega)$ est considérée comme une Curvelet « mère », dans le sens où tous les Curvelets à l'échelle 2^{-j} sont obtenues par des rotations et des translations de φ_j .

Ces Curvelets sont contrôlées à l'aide de trois paramètres :

- Le facteur d'échelle $j > 0$.
- La séquence équidistante des angles de rotation $\theta_\ell = 2\pi \cdot 2^{\lfloor j/2 \rfloor} \cdot \ell$ avec $\ell \in \mathbb{N}$ et $0 \leq \theta < 2\pi$ (à noter que l'espacement entre les angles consécutifs est en fonction de l'échelle)
- La séquence de paramètres de translation $k = (k_1, k_2) \in \mathbb{Z}^2$.

Avec ces notations, une Curvelet (comme fonction de $x = (x_1, x_2)$) à l'échelle 2^{-j} , l'orientation θ , et la position $x_k^{(j,\ell)} = R_{\theta_\ell}^{-1}(k_1 \cdot 2^{-j}, k_2 \cdot 2^{-j/2})$ est définie par :

$$\varphi_{j,\ell,k}(x) = \varphi_j \left(R_{\theta_\ell} \left(x - x_k^{(j,\ell)} \right) \right) \quad \text{(EQ : II-6)}$$

Où R_θ est la rotation par θ radians et R_θ^{-1} l'opération inverse (La transposée de R_θ),

$$R_\theta = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix}, \quad R_\theta^{-1} = R_\theta^T = R_{-\theta} \quad \text{(EQ : II-7)}$$

Un coefficient Curvelet est alors, le produit scalaire entre un élément $f \in L^2(\mathbb{R}^2)$ et une Curvelet $\varphi_{j,\ell,k}$.

$$c(j, \ell, k) := \langle f, \varphi_{j,\ell,k} \rangle = \int_{\mathbb{R}^2} f(x) \cdot \overline{\varphi_{j,\ell,k}} \cdot dx \quad \text{(EQ : II-8)}$$

En utilisant le théorème de Plancherel ce produit interne peut être exprimé comme l'intégrale sur le plan de fréquence.

$$c(j, \ell, k) := \frac{1}{(2\pi)^2} \int \hat{f}(\omega) \cdot \overline{\hat{\varphi}_{j,\ell,k}} d\omega = \frac{1}{(2\pi)^2} \int \hat{f}(\omega) \cdot U_j(R_{\theta_\ell} \omega) \cdot e^{i \langle x_k^{(j,\ell)}, \omega \rangle} d\omega \quad \text{(EQ : II-9)}$$

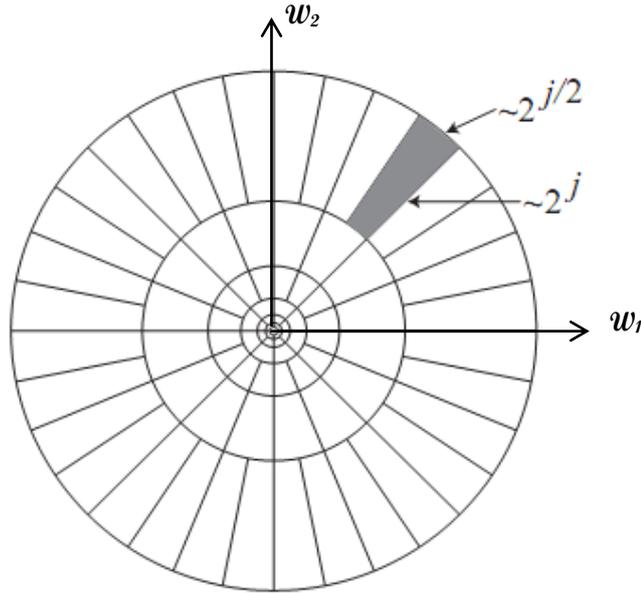


Figure II-3: Pavage du plan fréquentiel par les Curvelets continus.

Comme dans la théorie des ondelettes, on a également des éléments de l'échelle grossière. La fenêtre passe-bas W_0 vérifie la relation suivante :

$$|W_0(r)|^2 + \sum_{j \geq 0} |W_0(2^{-j}r)|^2 = 1 \quad \text{(EQ : II-10)}$$

Et pour $k_1, k_2 \in \mathbb{Z}$, les Curvelets d'échelle grossière sont définies par :

$$\varphi_{j_0, k}(x) = \varphi_{j_0}(x - 2^{-j_0}k) \hat{\varphi}_{j_0}(\omega) = 2^{-j_0} W_0(2^{-j_0}|\omega|) \quad \text{(EQ : II-11)}$$

Par conséquent, les Curvelets d'échelle grossière sont non directionnelles. La transformation de Curvelets « intégrale » est constituée des éléments directionnels $\varphi_{j, \ell, k}$ aux échelles $j \geq j_0$, et des ondelettes isotropes à l'échelle grossière père $\varphi_{j_0, k}$.

La figure II-4 présente un exemple d'une Curvelet des points de vue spatial et fréquentiel.

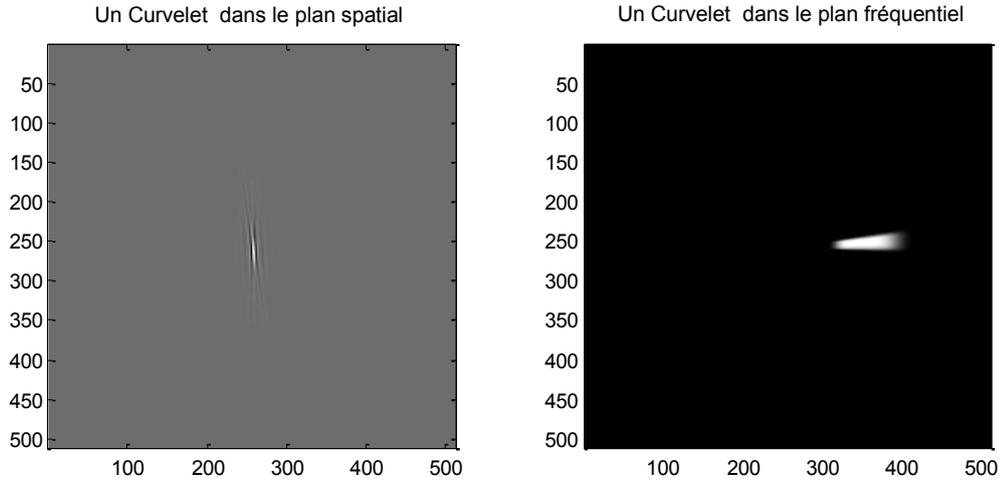


Figure II-4: Présentation d'un Curvelet de point de vue spatial et fréquentiel.

Comme nous travaillons sur des images numériques, nous devons nous placer dans le plan de la transformée discrète. Nous effectuons maintenant une présentation de la transformée en Curvelets dans le plan discret.

II.3.2 TRANSFORMEE EN CURVELETS DISCRETES

Pour réaliser les propriétés intéressantes de la transformation continue, et l'appliquer à un large éventail de problèmes, il faudrait une transformation en Curvelets discrètes, rapide et précise, convenable pour les données numériques. Deux implémentations distinctes de la transformée en Curvelets ont été introduites par E. Candès, L. Demanet et al [17]. Ces implémentations sont fidèles à la description mathématique décrite dans la section précédente.

Les coefficients de la transformée en Curvelets discrètes 2D $C^D(j, k, l)$ sont obtenus à partir d'une image numérique $f[t_1, t_2] / 0 \leq t_1, t_2 \leq n$, par l'équivalent numérique de l'équation (II-8). On obtient alors une collection de coefficients :

$$C^D(j, k, l) = \sum_{0 \leq t_1, t_2 \leq n} f[t_1, t_2] \cdot \varphi_{(j,k,l)}^D(t_1, t_2) \quad \text{(EQ : II-12)}$$

Où, chaque $\varphi_{(j,k,l)}^D$ est une forme d'onde de Curvelet numérique (L'exposant D est synonyme de « Discret »).

Comme dans la plupart des calculs scientifiques, ces formes d'ondes numériques ne sont pas construites, elles sont implicitement définies par les algorithmes ; formellement, elles sont les lignes de la matrice représentant la transformation linéaire et sont également connues comme « Riesz Representers ». Nous nous contentons de présenter ces formes d'onde, parce qu'elles offrent un moyen utile d'expliquer la relation avec la transformation continue. Les deux transformations numériques partagent une architecture commune qu'on introduit, en premier, avant de présenter la version Wrapping utilisée dans ce travail.

Description :

Dans la définition de la transformée en Curvelets continue [9], la fenêtre U_j extrait les fréquences proches de la couronne dyadique donnée par :

$$\{2^j \leq r \leq 2^{j+1}\} \text{ et près de l'angle } \{-\pi \cdot 2^{-\frac{j}{2}} \leq \theta \leq \pi \cdot 2^{-\frac{j}{2}-1}\}.$$

Cependant, Les Couronnes et les rotations ne sont pas spécialement adaptées au plan cartésien.

Pour remédier à ce problème, D. Donoho et E. Candès proposent de remplacer ces concepts par des équivalents cartésiens basés sur des carrés concentriques qui sont ensuite découpés en secteurs. Par exemple, l'analogue cartésien de la famille $W_j(\omega) = W(2^{-j}\omega)$ serait la fenêtre

$$\tilde{W}_j(\omega) = \sqrt{\Phi_{j+1}^2 - \Phi_j^2}, \quad j \geq 0 \tag{EQ : II-13}$$

Où Φ est définie comme le produit des fenêtres passe-bas unidimensionnelles données par :

$$\Phi_j(\omega_1, \omega_2) = \phi(2^{-j}\omega_1)\phi(2^{-j}\omega_2) \tag{EQ : II-14}$$

La fonction ϕ est positive et inférieure à 1, elle est égale à 1 sur $[-1/2, 1/2]$, et nulle en dehors de $[-2, 2]$. L'ensemble de ces fenêtres vérifie que :

$$\Phi_0(\omega) + \sum_{j \geq 0} \tilde{W}_j^2(\omega) = 1 \tag{EQ : II-15}$$

Nous venons de voir comment séparer les échelles dans l'environnement cartésien et maintenant on examine la localisation angulaire.

Supposons que V , vérifiant la condition d'admissibilité de l'équation (II-4) et donnée par : $V_j(\omega) = V(2^{\lfloor j/2 \rfloor} \omega_2 / \omega_1)$

Nous pouvons alors utiliser \tilde{W}_j et V_j pour définir les fenêtres « cartésiennes » :

$$\tilde{U}_j(\omega) := \tilde{W}_j(\omega).V_j(\omega) \quad \text{(EQ : II-16)}$$

Les fenêtres \tilde{U}_j isolent les fréquences proches du secteur :

$\{(\omega_1, \omega_2) : 2^j \leq \omega_1 \leq 2^{j+1}, -2^{-j/2} \leq \omega_2 / \omega_1 \leq 2^{-j/2}\}$, qui est un équivalent cartésien de la fenêtre « polaire » présentée précédemment.

Cette nouvelle architecture remplace les angles par un ensemble de pistes équidistantes :

$$\tan \theta_l := l.2^{-j/2}, \text{ avec } l = \left\{ -2^{-\frac{j}{2}}, \dots, 2^{-\frac{j}{2}} - 1 \right\} \quad \text{(EQ : II-17)}$$

L'ensemble des fenêtres est donc défini par :

$$\tilde{U}_{j,l}(\omega) := W_j(\omega).V_j(S_{\theta_l}\omega) \quad \text{(EQ : II-18)}$$

où S_θ est la matrice $S_\theta := \begin{pmatrix} 1 & 0 \\ -\tan \theta & 1 \end{pmatrix}$

Une fois les \tilde{U}_j terminées par symétrie autour de l'origine et la rotation de $\pm \pi/2$ radians, elles constituent l'analogue cartésien de la famille précédente $U_j(R_{\theta_l}\omega)$. La famille \tilde{U}_j implique un pavage concentrique dont la géométrie est représentée par la figure II-5.

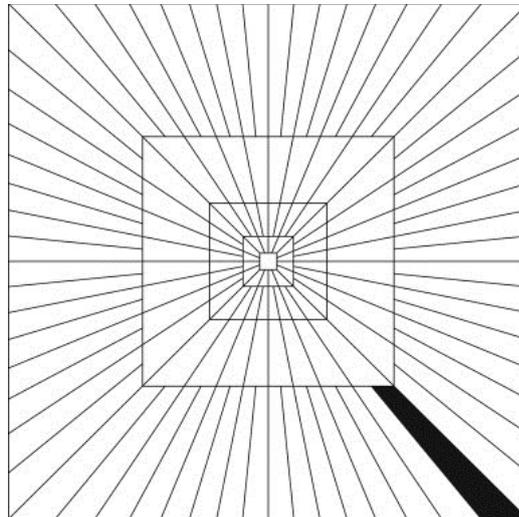


Figure II-5: Pavage de plan fréquentiel de la transformée en Curvelets numérique

Par construction les fenêtres $U_j(S_{\theta_l}\omega)$ sont définies en fonction de fenêtres angulaires par l'équation suivante:

$$U_j(S_{\theta_l}\omega) = V(2^{\frac{j}{2}}(\omega_2/\omega_1 - \ell)) \quad (\text{EQ : II-19})$$

et pour chaque $\omega = (\omega_1, \omega_2)$ avec $\omega_1 > 0$ l'équation (II-3) donne :

$$\sum_{l=-\infty}^{+\infty} |V_j(S_{\theta_l}\omega)|^2 = 1. \quad (\text{EQ : II-20})$$

En raison de la contrainte de support sur la fonction V , la somme ci-dessus est limitée à des angles d'intérêt $-1 \leq \tan \theta < 1$ respectant la formule suivante :

$$\sum_{\text{toutes les angles}} |V_j(S_{\theta}\omega)|^2 = 1 \quad (\text{EQ : II-21})$$

pour $\omega_2/\omega_1 \in [-1 + 2^{-\lceil j/2 \rceil}, 1 - 2^{-\lceil j/2 \rceil}]$

Ainsi, il résulte de (Eq10) que :

$$\sum_{\text{échelles } j} \sum_{\text{angles } \ell} |\tilde{U}_{j,\ell}(\omega)|^2 = 1 \quad (\text{EQ : II-22})$$

Pour évaluer cette formule avec des données discrètes, Il est nécessaire d'évaluer la transformée de Fourier inverse sur la grille irrégulière engendrée par les Curvelets numériques, dans laquelle l'algorithme FFT classique n'est pas valide.

Deux méthodes différentes ont été proposées par E. Candès, L. Demanet et al

- La transformée en Curvelets via l'algorithme Un-equispaced FFTs (USFFT).
- La transformée en Curvelets via l'algorithme Wrapping.

Les deux algorithmes donnent les mêmes résultats, mais l'algorithme via Wrapping est à la fois plus intuitif et plus efficace en temps de calcul [17]. Dans la suite, nous nous concentrons uniquement sur la mise en œuvre de FDCT par la version Wrapping.

II.3.3 TRANSFORMEE EN CURVELETS DISCRETES VIA WRAPPING

La version Wrapping fait un choix différent et plus simple de la grille spatiale pour la translation des Curvelets à chaque échelle et angle. Elle effectue le fenêtrage directement à la bonne localisation fréquentielle, puis utilise la propriété de périodicité du transformée de Fourier discrète afin d'obtenir les différents coefficients de la transformée en Curvelets.

Une grille régulière rectangulaire est utilisée, et les Curvelets « cartésiennes » sont définies par:

$$c(j, \ell, k) = \int \hat{f}(\omega) U_j(S_{\theta_\ell}^{-1} \omega) e^{i\langle b, \omega \rangle} d\omega \quad (\text{EQ : II-23})$$

avec $b \approx (k_1 2^{-j}, k_2 2^{-\frac{j}{2}})$ prenant des valeurs sur une grille rectangulaire.

La difficulté de cette approche est que dans le plan fréquentiel, les fenêtres $\tilde{U}_{j,\ell}[n_1, n_2]$ ne sont pas dans un rectangle de taille $\sim 2^j \times 2^{\frac{j}{2}}$ alignée avec les axes, dans lequel l'IFFT 2D peut être appliqué pour calculer les coefficients de Curvelets. Après discrétisation, l'intégrale sur ω devient une somme sur n_1, n_2 , qui s'étend au-delà des limites autorisées par l'IFFT 2D. La ressemblance de l'équation (II-14) avec une FFT 2D inverse standard n'est à cet égard que formelle.

Pour comprendre pourquoi le respect des tailles des rectangles est une préoccupation, nous rappelons que les $\tilde{U}_{j,\ell}$ sont pris en charge dans la région trapèzique $P_{j,\ell} = S_{\theta_\ell} P_j$.

L'idée proposée par E. Candès, L. Demanet et al, pour obtenir les coefficients de Curvelets à partir de cette fenêtre trapèzique, est de périodiser les échantillons de fréquences et ensuite extraire les données fréquentielles requises [17].

L'architecture de l'algorithme de FDCT via Wrapping est comme suit :

- 1) Appliquer la FFT 2D et obtenir des échantillons de Fourier $f[n_1, n_2]$ avec : $-\frac{n}{2} \leq n_1, n_2 < \frac{n}{2}$.
- 2) Pour chaque échelle j et angle ℓ , de former le produit :

$$\tilde{U}_{j,\ell}[n_1, n_2] \cdot \hat{f}[n_1, n_2]$$

- 3) Wrapping : périodiser le résultat dans un rectangle centré à l'origine et obtenir : $\tilde{f}_{j,\ell}[n_1, n_2] = W(\tilde{U}_{j,\ell}\hat{f})[n_1, n_2]$.Où la plage pour n_1, n_2 est maintenant de $0 \leq n_1 < L_{1,j}$, $0 \leq n_2 < L_{2,j}$ (pour $\theta \in [-\pi / 4, \pi / 4]$).
- 4) Appliquer le FFT 2D inverse à chaque $\tilde{f}_{j,\ell}$, pour obtenir les coefficients discrets $C^D(j, k, \ell)$.

L'opération de Wrapping, s'effectue de la manière suivante :

- 1) Sélection d'un parallélogramme de largeur L_2 et hauteur L_1 à recaler.
- 2) Répétition périodique de ce parallélogramme autour de l'origine.
- 3) Sélection d'un rectangle de largeur L_2 et hauteur L_1 centré sur l'origine.
La transformée de Fourier 2D inverse s'effectue alors sur la fenêtre de dimension $L_1 \times L_2$.

II.3.4 ISOMETRIE ET TRANSFORMATION INVERSE

Dans la pratique, les coefficients Curvelets sont normalisés comme suit,

$$C^{D,N}(j, k, \ell) = \frac{n}{\sqrt{L_{1,j} \cdot L_{2,j}}} \cdot C^D(j, k, \ell) \quad (\text{EQ : II-24})$$

Où $L_{1,j}, L_{2,j}$ sont les longueurs des côtés de parallélogramme $P_{j,\ell}$.Equipé de cette normalisation, la relation de Plancherel est exprimée comme suit :

$$\sum_{t_1, t_2} |f[t_1, t_2]|^2 = |C^{D,N}(j, k, \ell)|^2 \quad (\text{EQ : II-25})$$

Ceci est fondé sur le fait que chaque étape de la transformée est isométrique:

- La transformée discrète de Fourier, normalisée par $f[t_1, t_2] \rightarrow \frac{1}{n} \hat{f}[t_1, t_2]$ est une isométrie (et unitaire).
- La décomposition en différentes sous-bandes d'échelles et angles,
- $\hat{f}[n_1, n_2] = \{\tilde{U}_{j,\ell}[n_1, n_2] \cdot \hat{f}[n_1, n_2]\}_{j,\ell}$ est une isométrie parce que les fenêtres $\tilde{U}_{j,\ell}$ sont construites pour vérifier : $\sum_{j=0}^J \tilde{U}_{j,\ell}(\omega)^2 = 1$.
- La transformation par Wrapping n'est qu'une ré-indexation des échantillons de fréquences, ce qui permet de préserver les normes ℓ^2 .
- La transformée de Fourier locale inverse est une isométrie lorsqu'elle est correctement normalisée par $\frac{1}{\sqrt{L_{1,j} \cdot L_{2,j}}}$.

Grâce à cette propriété d'isométrie, la transformée en Curvelets inverse est tout simplement calculée comme l'adjoint de la transformée. Généralement, les adjoints peuvent être calculés par « l'inversement » de toutes les étapes de la transformée directe. La transformée en Curvelets inverse est calculée de la manière suivante:

1. Pour chaque paire échelle/ angle (j, ℓ) , effectuer une 2-D FFT (normalisée) de chaque vecteur $C^{D,N}(j, k, \ell)$ et obtenir $W(\tilde{U}_{j,\ell}\hat{f})[n_1, n_2]$.
2. Pour chaque paire échelle/angle (j, ℓ) , multiplier la matrice $W(\tilde{U}_{j,\ell}\hat{f})[n_1, n_2]$ par les Curvelets repliés correspondantes $W(\tilde{U}_{j,\ell})[n_1, n_2]$, ce qui donne $W(|\tilde{U}_{j,\ell}|^2\hat{f})[n_1, n_2]$.
3. Déplier chaque tableau $W(|\tilde{U}_{j,\ell}|^2\hat{f})[n_1, n_2]$ sur la grille de fréquence et ajoutez-les tous ensemble. Ceci permet de récupérer $\hat{f}[n_1, n_2]$.
4. Enfin, prendre un 2-D FFT inverse pour obtenir $f[t_1, t_2]$.

Dans l'approche Wrapping, la transformée directe et la transformée inverse sont calculés en $O(n^2 \log n)$ opérations, et nécessitent $O(n^2)$ opération de stockage [17].

II.3.5 PROPRIETES DE TRANSFORMATION EN CURVELETS

La transformée en Curvelets possède différentes propriétés intéressantes, nous allons présenter quelques-unes, à savoir :

a) **Support compact :**

La trame de fonctions Curvelets forme un support compact de $L^2(\mathbb{R}^2)$. Cela signifie que chaque fonction $f(x_1, x_2) \in L^2(\mathbb{R}^2)$ peut être représentée par une série de Curvelets, la formule de reconstruction est donnée par :

$$f = \sum_{j,k,l} \langle f, \varphi_{j,k,l} \rangle \varphi_{j,k,l} \quad (\text{EQ : II-26})$$

Avec l'égalité dans un sens L^2 , et une relation de Plancherel :

$$\sum_{j,k,l} |\langle f, \varphi_{j,k,l} \rangle|^2 = \|f\|_{L^2(\mathbb{R}^2)}^2, \quad \forall f \in L^2(\mathbb{R}^2) \quad (\text{EQ : II-27})$$

b) Localisation fréquentiel et spatial :

Les Curvelets sont bien localisées dans les domaines temporel et fréquentiel. En raison de leur forme, ils possèdent une grande sensibilité directionnelle, leurs longueurs et largeurs effectives obéissent à la relation d'échelle anisotrope.

$$\text{longueur} \approx 2^{-j/2}, \text{largeur} \approx 2^{-j} \Rightarrow \text{longueur} \approx \text{largeur}^2 \quad (\text{EQ : II-28})$$

c) Moments directionnels :

Les Curvelets possèdent un nombre infini de moments directionnels. Cette propriété implique que si le support essentiel du Curvelet $\varphi_{j,k,l}$ se trouve dans une partie régulière de f , alors le coefficient de Curvelet correspondant $C_{j,k,l}$ sera de faible amplitude, tandis que, si le support essentiel de $\varphi_{j,k,l}$ est aligné avec un bord de f , alors le coefficient $C_{j,k,l}$ sera de grande amplitude. Ces propriétés de Curvelets sont essentielles pour leur capacité de détecter les régularités géométriques le long des bords des objets.

II.4. COEFFICIENTS DE CURVELETS VS. COEFFICIENTS D'ONDELETTES

Dans le voisinage d'une singularité curviligne ayant une tangente dans une orientation θ , les coefficients de grandes amplitudes sont créés par les Curvelets et les ondelettes de l'orientation $\alpha = \theta$, qui ont leurs moments nuls oscillants dans la direction $\theta + \pi/2$. A cause de leur support allongé dans la direction du bord θ , moins de Curvelets sont nécessaires pour la représentation du bord curviligne par rapport aux ondelettes, qui se caractérisent par une forme carrée, comme illustré sur la figure suivante :

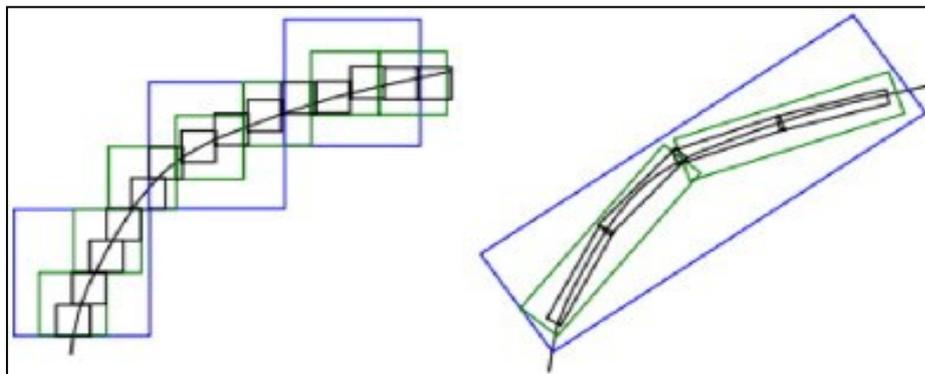


Figure II-6: La représentation des bords, par ondelettes et Curvelets.

Si l'orientation des Curvelets α dévie de θ , les coefficients de Curvelets se décroissent rapidement en raison de la localisation fréquentielle fine de Curvelets. Ce qui donne une haute (meilleure) sélectivité directionnelle aux Curvelets.

Même si les ondelettes oscillent quand $\alpha = \theta + \pi/2$, ils ont une sélectivité directionnelle inférieure par rapport aux Curvelets, et les amplitudes des coefficients d'ondelettes décroissent lentement quand l'angle s'éloigne de θ . En effet, le support fréquentiel des ondelettes est beaucoup plus large et leur support spatial est presque isotrope. En conséquence, un bord produit des coefficients d'ondelettes de grande amplitude dans plusieurs directions. Ainsi, les bords et les structures d'image avec une certaine régularité directionnelle créent moins de coefficients de Curvelets de grandes amplitudes que de coefficients d'ondelettes.

II.5. CONCLUSION :

La nécessité d'avoir des représentations parcimonieuses capables de représenter efficacement les régularités géométriques présentes dans l'image a motivé les chercheurs pour développer une nouvelle famille d'ondelettes, dite les ondelettes géométriques.

Les Curvelets proposées par E. Candès et D. Donoho, constituent une nouvelle famille de trames d'ondelettes géométriques. Elles sont conçues pour représenter de façon parcimonieuse les régularités curvilignes.

Dans ce chapitre, nous avons présenté les Curvelets et la raison derrière leur supériorité dans la représentation des régularités géométriques par rapport aux ondelettes classiques, ainsi que leurs principales propriétés qui font des Curvelets un outil efficace dans plusieurs applications.

CHAPITRE III:
QUANTIFICATION ET CODAGE.

III.1. INTRODUCTION :

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 $X \in \mathbb{R}$, une quantification scalaire estime les valeurs du signal indépendamment les uns des autres de sorte que chaque valeur quantifiée est obtenue par opération séparée. Lorsque les valeurs du signal X sont très interdépendantes, une quantification vectorielle, qui quantifie les valeurs de X ensemble, peut nettement améliorer la performance de la quantification scalaire. Cependant, la mise en œuvre d'un quantificateur vectoriel est d'une plus grande complexité. De plus, si une étape de structuration de la redondance précède la quantification, les coefficients résultants seront par définition moins dépendants les uns des autres, et le gain apporté par le quantificateur vectoriel ne vaudra pas son coût de calcul.

De ce fait, La quantification scalaire est donc la plus souvent utilisée dans la compression des images par transformation en pratique. Dans ce chapitre nous présenterons les différentes méthodes de quantification scalaire, et sa relation avec la phase du codage dans un système de compression d'images.

III.2. QUANTIFICATION SCALAIRE :

Dans la pratique, une quantification est composée de deux blocs : un bloc codeur et un bloc décodeur. Le codeur divise la plage de valeurs du signal à quantifier X en un certain nombre d'intervalles $[b_{i-1}; b_i]$. Chaque intervalle est ensuite représenté par un indice distinct I_i . Le décodeur représente l'ensemble des valeurs de signal entré X qui se situent dans un intervalle donné par le niveau de reconstruction y_i représentant l'intervalle.

Notons les limites de décision par $\{b_i\}_{i=0}^M$ et les niveaux de reconstruction par $\{y_i\}_{i=1}^M$, et l'opération de quantification par $Q(.)$.

Alors : $Q(x) = y_i$ si $b_{i-1} < x \leq b_i$.

Cette opération est irréversible parce que les différentes valeurs de coefficients qui peuvent appartenir au même intervalle seront codées par un seul indice. Connaître l'indice nous informe seulement sur l'intervalle auquel appartient la valeur d'échantillon. Il ne nous permet pas de savoir quelle est la valeur réelle de l'échantillon, parmi les nombreuses valeurs dans l'intervalle.

Pour chaque indice I_i généré par le codeur, le décodeur génère une valeur de reconstruction y_i qui, dans un certain sens, représente le mieux les valeurs de l'intervalle. Il existe différentes méthodes pour la construction des limites de décision $\{b_i\}_{i=0}^M$ et les niveaux de reconstruction $\{y_i\}_{i=1}^M$ suivant l'information que nous avons sur la répartition du signal à quantifier X . Dans l'exemple de la figure III-1, nous utilisons simplement le milieu de l'intervalle comme valeur représentative générée par le décodeur.

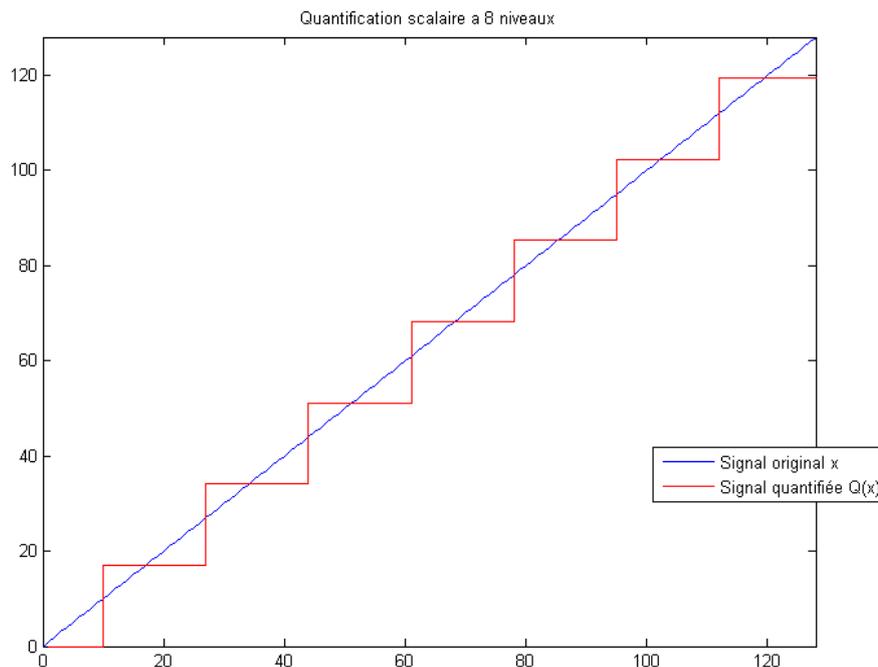


Figure III-1: Exemple d'une quantification scalaire à 8 niveaux.

III.2.1 QUANTIFICATION SCALAIRE UNIFORME :

La Quantification uniforme est la technique de quantification scalaire la plus simple, elle consiste à diviser la dynamique du signal original en un ensemble d'intervalles identiques $B^i = [b_i; b_{i+1}]$, et des niveaux de reconstruction y_i espacés régulièrement.

La largeur fixe des intervalles $\Delta = b_{i+1} - b_i$ est souvent appelée un pas de quantification. Lorsque le pas de quantification est plus grand, nous obtenons des valeurs beaucoup plus redondantes dans le signal de sortie, et par conséquent une capacité de compression plus importante, mais contre une précision plus faible lors de la reconstitution à cause de la distorsion introduite. En revanche, on peut obtenir moins de distorsions de quantification (meilleure précision) en utilisant un pas de quantification plus petite, mais ça sera au détriment de la compression.

Considérons un signal à temps discret $X(k)$ prenant ses valeurs dans l'intervalle $[-A, +A]$. Définir un quantificateur scalaire avec une résolution de n bits par échantillon consiste à réaliser trois opérations :

1. une partition de l'intervalle $[-A, +A]$ en $L = 2^n$ intervalles distincts $\{B^1 \dots B^L\}$ de longueur Δ .
2. une numérotation des intervalles $\{I^1 \dots I^L\}$.
3. la sélection d'un représentant par intervalle, l'ensemble de ces représentants composant un dictionnaire (code book)³ $Y = \{y_1 \dots y_L\}$.

Nous citons deux types des quantificateurs en fonction de principe de reconstitution. Le premier est appelé « Midrise » et sera désigné par Q_{MR} . Le second est appelé « Midtread » et sera désigné par Q_{MT} . Cette terminologie est basée sur ce qui se passe au voisinage du zéro, et utilise l'analogie de représenter la fonction IN-Out de quantificateur sous forme d'un escalier. Le quantificateur « Midtread » a un niveau de reconstruction y_0 nul, cependant, le quantificateur « Midrise » a un seuil de classification du niveau de quantification b_0 nul.

³ Dans le cas scalaire, on parle habituellement de niveaux de quantification, pas de quantification, ou seuil de décision.

Ces deux types de quantificateurs sont définis par les équations suivantes :

$$y = Q_{MR}(x) = \text{sign}(x) \cdot \Delta \cdot \left(\left\lfloor \frac{|x|}{\Delta} + \frac{1}{2} \right\rfloor \right) \quad (\text{EQ : III-1})$$

$$y = Q_{MT}(x) = \text{sign}(x) \cdot \Delta \cdot \left\lfloor \frac{|x|}{\Delta} + \frac{1}{2} \right\rfloor \quad (\text{EQ : III-2})$$

Où $\lfloor \cdot \rfloor$ = arrondissement vers Zéro

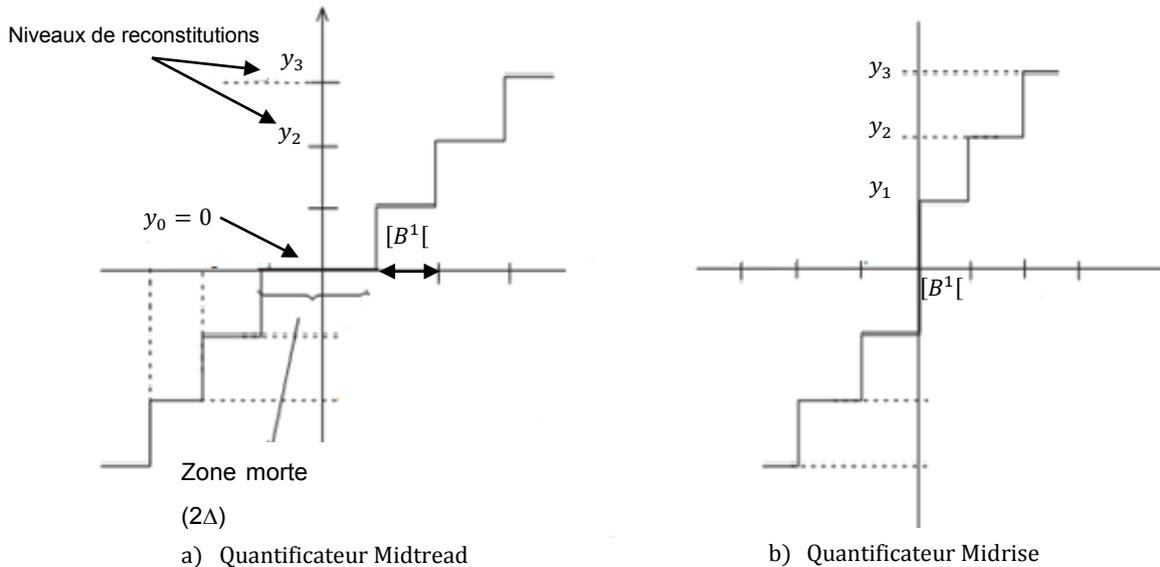


Figure III-2: (a) Quantificateur Midtread (b) Quantificateur Midrise

Les deux types de quantificateurs sont décrits graphiquement sur la figure III-2 dans le cas de six niveaux de reconstitution.

D'après ces figures, on peut noter la non-symétrie du Q_{MT} par rapport aux deux axes x et y . On note aussi que les deux quantificateurs ont un niveau de quantification irrégulier près de l'origine. En examinant les graphes **a** et **b** on peut encore noter que $Q_{MR}(0) = \frac{\Delta}{2}$ et que $Q_{MT}(0) = 0$. En d'autres termes, le niveau zéro est représenté par $\frac{\Delta}{2}$ dans le cas d'un Q_{MR} tandis qu'il est codé parfaitement par un Q_{MT} .

Le choix d'utiliser un Q_{MR} ou un Q_{MT} est basée principalement sur l'importance du niveau zéro dans l'application utilisant le quantificateur. Si la probabilité d'une amplitude nulle est forte dans le signal original, alors un Q_{MT} est le meilleur choix. C'est le cas dans le traitement des signaux audio où les faibles amplitudes sont hautement probables et perceptuellement importantes. Il en va autrement en vidéo puisqu'il est recommandé dans cette industrie de ne pas générer d'images ayant des amplitudes extrêmes. Cette contrainte donne une marge de manœuvre aux modules de traitement qui suivent la génération d'images. En effet, en gardant les amplitudes dans un domaine restreint, les effets de débordement sont évités. Plus particulièrement, il n'est pas recommandé de générer des amplitudes nulles de luminances. Les quantificateurs de type Q_{MR} sont donc favorisés dans le domaine de la vidéo.

III.2.2 QUANTIFICATION UNIFORME A ZONE MORTE

La quantification scalaire à zone morte (Dead-zone en anglais) est une quantification scalaire où l'intervalle autour de zéro est plus large. La zone morte notée T qualifie cet intervalle qui permet à l'ensemble des valeurs considérées comme petites d'être quantifié en une seule et même valeur égale à zéro.

Au sens strict, ce type de quantification est non uniforme ($T \neq \Delta$). Toutefois, si tous les autres pas de quantification sont égaux, on appelle cette quantification, uniforme à zone morte.

Ce type de quantification est très répandu en compression d'image. Elle peut être vue comme l'association d'une quantification uniforme et une opération de seuillage par rapport à une valeur Q . Cette quantification est utilisée dans les schémas de compression par Curvelets proposés par M.S. Joshi, R.R. Manthalkar, et Y.V. Joshi [18], et par K. Gupta et S. Silakari [19].

La Figure III-3 présente une illustration d'un quantificateur scalaire uniforme et un quantificateur scalaire uniforme à zone morte.

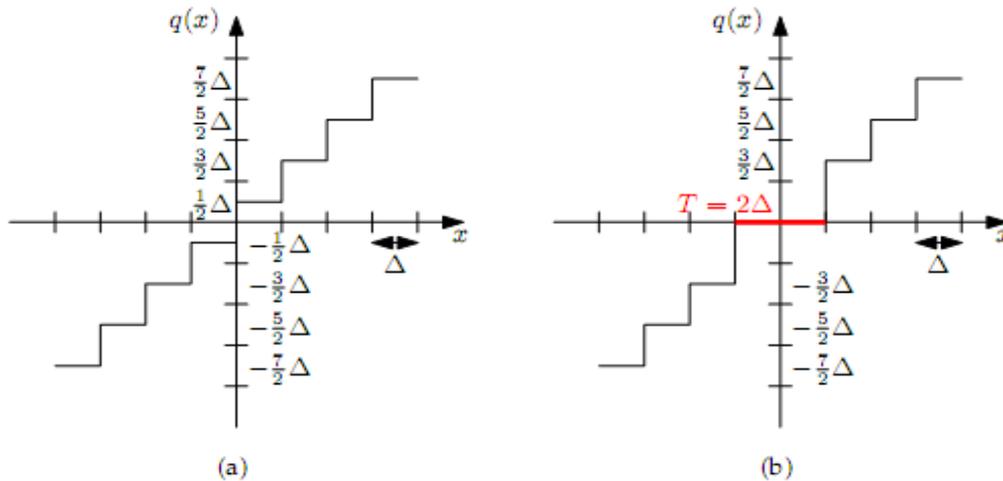


Figure III-3 : Quantificateur uniforme à zone morte

Un quantificateur scalaire uniforme de pas de quantification Δ est représenté graphiquement sur la figure III-3 (a), tandis que la Figure III-3 (b) montre un quantificateur scalaire uniforme à zone morte de largeur $T = 2\Delta$.

III.2.3 QUANTIFICATION 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.

III.2.4 ALGORITHME DE LLOYD-MAX :

Une approche directe pour trouver le meilleur quantificateur non uniforme, si nous avons un modèle de probabilité pour la source $f_x(x)$, est de trouver les bornes des intervalles $\{b_i\}_{i=0}^M$ et les niveaux de reconstitution $\{y_i\}$ qui minimisent la distorsion introduite lors du processus de quantification. (La différence entre les entrées du quantificateur x et la sortie $y = Q(x)$, est appelé l'erreur de quantification).

Lorsqu'on choisit, comme une mesure de distorsion, l'erreur quadratique moyenne est donnée par :

$$\sigma_q^2 = \int_{-\infty}^{\infty} (x - Q(x))^2 f_x(x) dx = \sum_{i=1}^M \int_{b_{i-1}}^{b_i} (x - y_i)^2 f_x(x) dx \quad (\text{EQ : III-3})$$

Le meilleur quantificateur peut être trouvé en supposant le dérivé de l'équation (EQ : III-3) par rapport à y_i égal à zéro, et résoudre l'équation pour y_i , on obtient

$$y_i = \frac{\int_{b_{i-1}}^{b_i} x f_x(x) dx}{\int_{b_{i-1}}^{b_i} f_x(x) dx} \quad (\text{EQ : III-4})$$

La valeur représentative pour chaque intervalle de quantification est le centre de gravité de la masse de probabilités dans cet intervalle. En prenant la dérivée par rapport au b_i , et le considère égal à zéro, on obtient l'expression suivante pour b_i :

$$b_i = \frac{y_{i+1} + y_i}{2} \quad (\text{EQ : III-5})$$

La limite de décision est tout simplement le point milieu entre les deux niveaux de reconstruction voisins. La résolution de ces deux équations nous donne les valeurs des niveaux de reconstruction et les limites de décision qui minimisent l'erreur quadratique moyenne de quantification.

Malheureusement, pour résoudre pour y_i , nous avons besoin des valeurs de b_i et b_{i-1} , et pour résoudre pour b_i , nous avons besoin des valeurs de y_i et y_{i+1} .

Lloyd Stuart P a décrit une approche itérative pour trouver l'ensemble des limites de décision et des valeurs quantifiées pour une quantification non uniforme optimale [20]. Cette approche nommée souvent l'algorithme « Max-Lloyd » implique généralement un grand nombre d'itérations, d'où peut être un calcul intensif, surtout lorsqu'il s'agit d'un grand nombre M . L'exigence de stockage de ces limites de décision et des niveaux de reconstitution peut aussi devenir excessive, en particulier pour le décodeur. Une solution qui permet de surmonter ces difficultés est le modèle Companding.

III.2.5 QUANTIFICATION NON UNIFORME PAR COMPRESSION

(COMPANDING) :

Pour un signal de source non uniformément répartie, on peut 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 peut être utilisé.

Pour le décodage, les indices de quantification reçus sont d'abord dé-quantifiés pour reconstituer les valeurs quantifiées de manière uniforme. Ensuite, la fonction inverse $F^{-1}(x)$ est appliquée afin de produire les valeurs finales quantifiées.

Ce processus est illustré dans le schéma de Figure III-4.

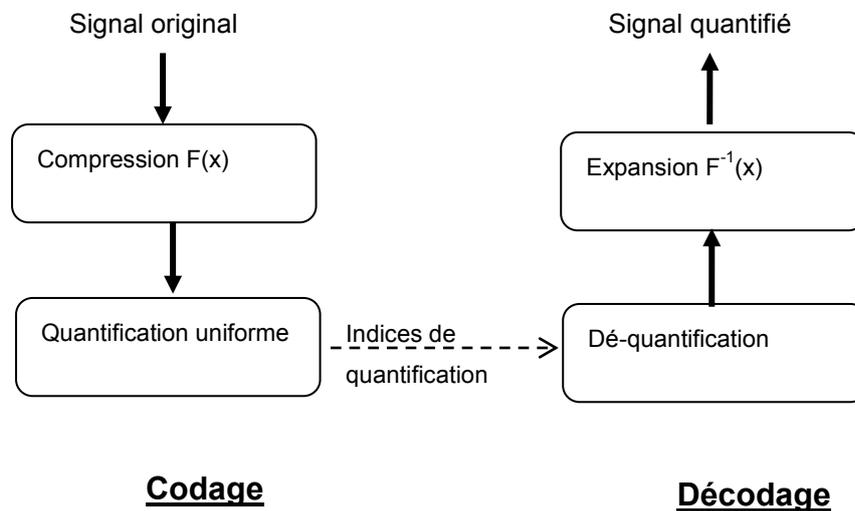


Figure III-4: Processus de quantification non uniforme par Companding

La fonction non linéaire $F(x)$ est appelée une fonction de compression, car elle a généralement une forme analogue à celle représentée sur la Figure III-5 qui amplifie les faibles amplitudes de signal source et comprime les échantillons de grandes amplitudes.

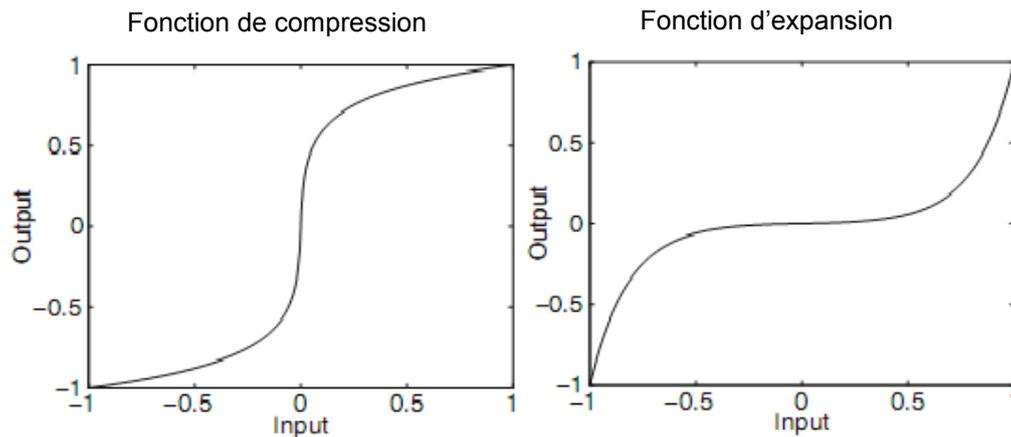


Figure III-5 : Exemples des fonctions de compression et d'expansion

La fonction inverse F^{-1} est appelée une fonction d'expansion (expand en Anglais) du fait que l'inverse de la compression est l'expansion. Après cette opération de compression-expansion, abrégée par « companding », les bornes de décision efficaces seront non uniformes, lorsqu'elles sont vues 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 possèdent 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 du signal converti similaire à une distribution uniforme.

Deux types de fonctions communes sont utilisés dans la conception des quantificateurs non uniforme par Companding: la fonction à loi de puissance (A-law) et la fonction logarithmique. Ces techniques réduisent le nombre de bits nécessaire à coder le signal et maintiennent un rapport signal à bruit acceptable.

Généralement, La fonction à loi de puissance de la forme suivante est utilisée :

$$c_{\text{power}}(|x|) = |x|^p \quad (\text{EQ : III-6})$$

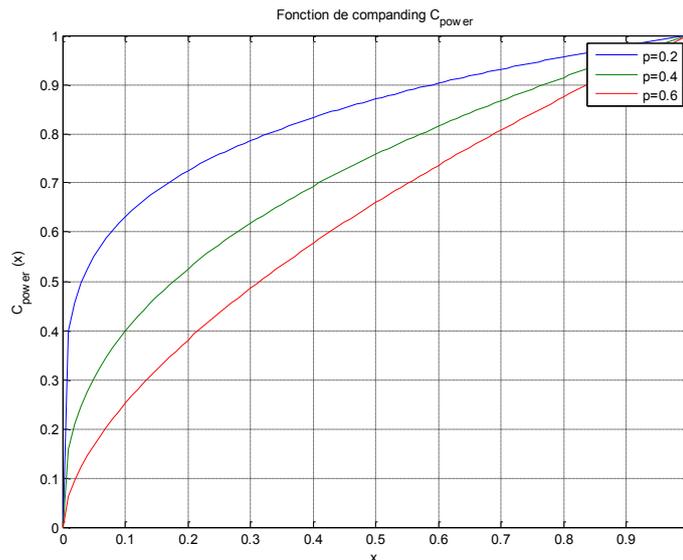


Figure III-6: Fonctions à loi de puissance pour ($p=0.2$, $p=0.4$, $p=0.6$).

Alors qu'une fonction de forme $\log(x)$ est utilisée dans le companding logarithmique, les fonctions souvent utilisées sont les fonctions A-law⁴ et μ -law, cette dernière sera plus détaillée puisqu'elle sera introduite dans la méthode de quantification proposée ultérieurement.

III.2.6 MU-LAW COMPANDING

μ -law companding 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 laquelle les signaux sonores de hautes amplitudes ne nécessitent pas la même résolution que celles de faibles amplitudes.

L'oreille humaine est plus sensible au bruit de quantification dans les petits signaux que dans les signaux forts. « μ -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.

⁴La **loi A** (en anglais *A-Law*) est un système de quantification logarithmique d'un signal audio, utilisé principalement à des fins de compression pour les applications traitant la voix humaine. Elle est standardisée sous la référence ITU-T G.711. Utilisée principalement en Europe, elle est semblable à la loi Mu utilisée aux États-Unis et au Japon.

La fonction $F(x)$ de compression μ -law est définie par l'équation (III-7), où μ est le coefficient de compression ($\mu = 255$ aux États-Unis et le Japon) et x le vecteur de données à quantifier. [21]

$$F(x) = \frac{\text{sign}(x) \times \ln(1 + \mu|x|)}{\ln(1 + \mu)} \quad 0 \leq |x| \leq 1 \quad (\text{EQ : III-7})$$

La Figure III-7 présente cette fonction pour les trois valeurs de $\mu = 1, 10,$ et 255 . Plus la valeur de μ est grande plus les valeurs de faibles amplitudes sont amplifiées.

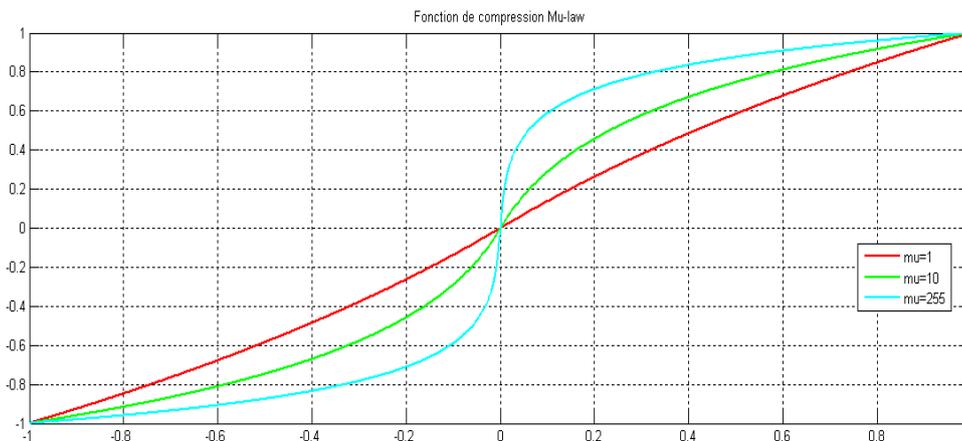


Figure III-7: Fonction de compression M-law ($\mu = 1, 10, 255$)

Le décodage des données codées n'est ensuite qu'une question d'inverser les étapes de l'encodage. Les étapes de quantification par compression μ -law peuvent être représentées par le schéma de la Figure III-8.

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

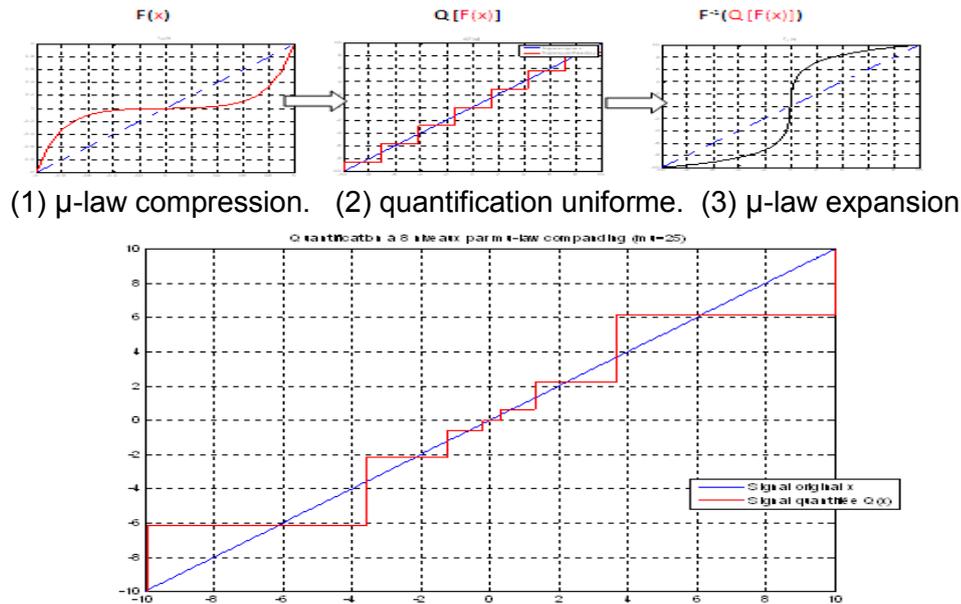


Figure III-8: (a)-Etapes de quantification par Companding. (b)-quantification μ -law à 8 niveaux. ($\mu=25$)

III.3. QUANTIFICATION ET ENTROPIE :

Après l'étape de quantification, plusieurs des valeurs de coefficients sont éliminées ou remplacées par d'autres valeurs convenables. Cependant, ces valeurs sont toujours représentées avec des codes $\{I\}$ de longueur fixe (généralement 8 ou 16 bits), alors le nombre de coefficients conservé à la sortie de quantificateur indique directement la taille de données. Si le nombre de coefficients obtenus à la sortie de quantificateur est M , la longueur du code du signal source X en bits sera tout simplement le produit du nombre de coefficients M par la longueur du mot du code en bits :

$$L_c(X) = M \cdot \ell(\text{bits}) \quad (\text{EQ : III-8})$$

Toutefois, certains coefficients y_i sont plus redondants que les autres, donc il sera plus efficace de les coder avec moins de bits, notons la probabilité d'apparition d'un coefficient y_i par $\text{prob}(y_i)$, la longueur de code $L_c(X)$ du signal est donnée par :

$$L_c(X) = \sum_{i=1}^M \ell_i \cdot \text{prob}(y_i) \quad (\text{EQ : III-9})$$

Où ℓ_i est la longueur de code de l'événement y_i .

Pour un signal X caractérisé par la fonction de distribution $f_x(x)$. Les probabilités $\text{prob}(y_i)$ dépend des limites de décision des intervalles de quantification $[b_i]$, la probabilité que y_i se produise est donnée par :

$$\text{prob}(y_i) = \int_{b_{i-1}}^{b_i} f_x(x) \cdot dx \quad (\text{EQ : III-10})$$

Par conséquent, la taille $L_c(X)$ est en fonction des limites de décision et elle est donnée par l'expression :

$$L_c(X) = \sum_{i=1}^M \ell_i \cdot \int_{b_{i-1}}^{b_i} f_x(x) \cdot dx \quad (\text{EQ : III-11})$$

À partir de cette expression, on constate que les partitions que nous sélectionnons et les codes binaires représentant ces partitions sont les paramètres qui déterminent la taille finale de données $L_c(X)$. Le problème de trouver les partitions optimales, les codes et les niveaux de représentation sont tous liés.

Le but de codage entropique est de minimiser le nombre de bits moyen utilisé pour représenter les réalisations y_i , ce qui revient à approcher le plus possible la borne inférieure de débit définie par le théorème de Shannon qui est égal à l'entropie de la source Y .

$$H(Y) = - \sum_{i=1}^M \text{prob}(y_i) \cdot \log_2(\text{prob}(y_i)) \quad (\text{EQ : III-12})$$

Le codage de Huffman et le codage arithmétique sont les deux codeurs les plus utilisés dans la compression d'image. Dans ce qui suit, nous présentons brièvement ces deux codeurs ainsi que leurs principes de fonctionnement.

III.3.1 CODAGE ENTROPIQUE DE HUFFMAN

D.A. Huffman a inventé en 1952, un algorithme de codage entropique utilisé pour la compression sans perte de données [22].

Le principe est d'utiliser un plus petit nombre de bits pour coder les données qui se produisent plus fréquemment [22]. La longueur moyenne d'un code de Huffman dépend de la fréquence statistique avec laquelle la source produit chaque symbole de l'alphabet.

Un dictionnaire du code Huffman respecte les deux restrictions de base suivantes :

- Il n'ya pas deux symboles se composant de chiffres de codage identiques.
- Les codes de symboles seront réalisés de telle manière qu'aucune indication supplémentaire n'est nécessaire pour le décodage.

Ce qui implique qu'aucun message ne soit codé par un code similaire, bit à bit, à une première partie d'un code de message de plus grande longueur.

Tout ce que le récepteur a besoin de savoir est l'ensemble de codes. Le procédé de codage Huffman est basé sur la construction de ce qu'on appelle un arbre binaire. Le chemin à partir de la racine de cet arbre vers un événement particulier déterminera le groupe de codes que nous associons à cet événement.

Pour la construction de l'arbre, trier l'Alphabet dans l'ordre décroissant, $S = (s_1, \dots, s_n)$ en fonction de la distribution de probabilités associée $P = (p_1, \dots, p_n)$. chaque symbole S_i représente la racine d'un sous-arbre.

La méthode pour effectuer le codage de Huffman consiste à répéter les étapes suivantes jusqu'à ce qu'il ne reste qu'un seul symbole composite en S :

Étape 1 : Relier les deux symboles S_i et S_j de l'alphabet qui ont les plus faibles probabilités p_i et p_j .

Étape 2 : Enlever S_i et S_j de l'alphabet et ajouter un nouveau symbole combiné et (S_i, S_j) avec une probabilité $(P_i + P_j)$. Le nouveau symbole représente la racine du sous-arbre. Maintenant, l'alphabet contient un symbole moins qu'avant.

Étape 3 : Insérez le nouveau symbole (S_i, S_j) à une position plus élevée possible, afin de garder l'ordre décroissant de l'alphabet.

III.3.2 LIMITES DU CODAGE DE HUFFMAN :

Le problème avec ce système réside dans le fait que les codes de Huffman devraient avoir une longueur en bits naturelle ($L \in \mathbb{N}$). Par exemple, si la probabilité d'un symbole est $1/3$, le nombre optimal de bits pour coder ce symbole est d'environ 1,6, mais le codage Huffman affecte soit un ou deux bits pour le code, ce qui est non optimal dans les deux cas. Ce codage non optimal devient un problème perceptible lorsque la probabilité d'un symbole devient très élevée. Pour une probabilité de 90 %, la taille du code optimale serait de 0,15 bit. Tandis que, le système de codage de Huffman attribue un code d'un bit pour le symbole, soit un code six fois plus long que nécessaire.

III.3.3 CODAGE ARITHMETIQUE :

Contrairement à l'algorithme de Huffman qui associe aux symboles des motifs binaires dont la taille dépend de leur distribution. Le codeur arithmétique traite le vecteur de données dans son ensemble, en lui associant un unique nombre décimal rationnel varie entre 0 et 1. Ce nombre code est construit par subdivisions récursives d'intervalles. Un intervalle est subdivisé pour chaque nouveau symbole qui appartient à la séquence. On obtient, à la fin, un sous-intervalle de l'intervalle $[0, 1[$. Tel que tout nombre réel appartenant à cet intervalle représente la séquence à coder. Ce nombre compris entre 0 et 1, possède d'autant moins de chiffres après la virgule que la redondance du vecteur de données. Ces chiffres décimaux dépendent non seulement des symboles du vecteur dans l'ordre où ils apparaissent, mais aussi de leur distribution statistique [23].

Soit $S = \{s_1, \dots, s_N\}$ une source pouvant délivrer N symboles associée aux probabilités $P\{s_k\} = p_k$ pour tout entier k de $[1, N]$.

Pour coder une séquence $s_M = \{s_{\alpha_1}, s_{\alpha_2}, \dots, s_{\alpha_M}\}$ de M symboles, on utilise l'algorithme suivant (s_{α_k} est le k ème symbole de la séquence émise par la source) :

Étape 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_{α_1} parmi tous les symboles s_K de la source). La taille de cet intervalle est donc définie par: $taille = H_c - L_c = 1$.

Étape 2 : 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 :

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

Étape 3 : On choisit le sous-intervalle correspondant au prochain symbole s_{α_k} qui apparaît dans la séquence. On redéfinit alors l'intervalle initial.

$$[L_c, H_c]: \begin{cases} L_c = L_c + taille \times L_{s_k} \\ H_c = L_c + taille \times H_{s_k} \end{cases}$$

Étape 4 : 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 sources.

Similaire au codage de Huffman, le codage arithmétique utilise la probabilité d'apparition des symboles du signal source et cherche à optimiser la longueur du code correspondant. Cependant, l'AC est souvent plus optimal, et rapproche la limite théorique spécifiée par l'entropie de Shannon, malgré les légères inexactitudes causées parfois lors de la division d'intervalles. Par contre, le codage de Huffman produit toujours des erreurs d'arrondissement, car la longueur de son code est limitée à des multiples d'un bit. Cet écart par rapport à l'optimum théorique est beaucoup plus élevé par rapport aux inexactitudes du codage arithmétique. Le codage arithmétique est donc toujours plus efficace ou identique au codage de Huffman.

De ce fait, le Codage arithmétique a été largement utilisé comme un algorithme de compression efficace dans les nouvelles normes telles JPIG2, JPEG2000 [24] et H.264/AVC. L'association de Codage arithmétique à un RLE pour la compression d'image sans perte dans [25] a porté une amélioration nette et considérable.

III.4. CONCLUSION :

La quantification est la phase la plus importante dans le 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 vectorielle, si elle est précédée par une transformation qui génère des coefficients fortement décorrélés. Elle peut être plus performante en adaptant le dictionnaire de quantification aux caractéristiques du signal à traiter, ainsi on obtient des coefficients statistiquement plus redondants.

Deux codeurs entropiques sont souvent utilisés dans la compression d'image, le codage de Huffman et le codage arithmétique. Pourtant, ce dernier semble plus compliqué et plus lent, il fera toujours un meilleur ou aussi bon travail que le codage de Huffman en terme de taux de compression.

CHAPITRE IV:
RESULTATS ET DISCUSSIONS

IV.1. INTRODUCTION :

La compression avec pertes, par opposition à la compression sans pertes, se permet d'éliminer quelques informations pour avoir le meilleur taux de compression 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, on va proposer une méthode de compression d'images fixes avec pertes, utilisant une transformation en ondelettes géométriques, afin d'exploiter les régularités directionnelles qui représentent souvent la partie la plus importante dans plusieurs applications.

Cette méthode s'appuie sur la transformée en Curvelets et la quantification scalaire logarithmique décrites dans les chapitres précédents.

IV.2. METHODOLOGIE :

Une compression de données efficace peut se résumer en trois étapes principales, la transformation, la quantification et le codage.

Nous décrivons dans ce qui suit chacune des trois principales étapes de la méthode de compression proposée.

IV.2.1 DECOMPOSITION EN CURVELETS :

Les deux versions du transformée en Curvelets rapide, FDCT via Wrapping FDCT via USFFT sont implémentées par E. Candès, L. Demanet, et Lexing Ying dans la boîte à outils MATLAB « CurveLab 2.0.3 » [26]. Les résultats présentés dans ce document peuvent être achevés par les deux implémentations. Toutefois, en raison de son efficacité computationnelle, et pour éviter les artefacts d'interpolation de la version USFFT, la mise en œuvre par la version Wrapping est utilisée dans ce travail.

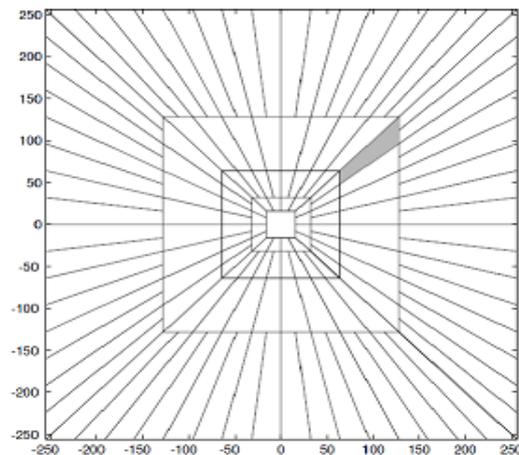


Figure IV-1: Pavage pseudo polaire de plan fréquentiel.

Le choix de nombre d'échelles J dans l'algorithme du transformée en Curvelets permet de définir la taille du niveau le plus grossier ($j = 1$) du pavage pseudo polaire du plan fréquentiel par les Curvelets.

Les valeurs de coefficients de ce niveau représentent les zones continues, et relativement lisses dans le plan spatial de l'image. Naturellement, ces zones ne sont pas directionnelles et l'utilisation des Curvelets dans ces régions n'est pas nécessaire, et pourrait même conduire à une dégradation des performances. La figure IV-2 présente l'image Lena et la valeur absolue de sa transformée de Fourier représentée sur une échelle logarithmique.

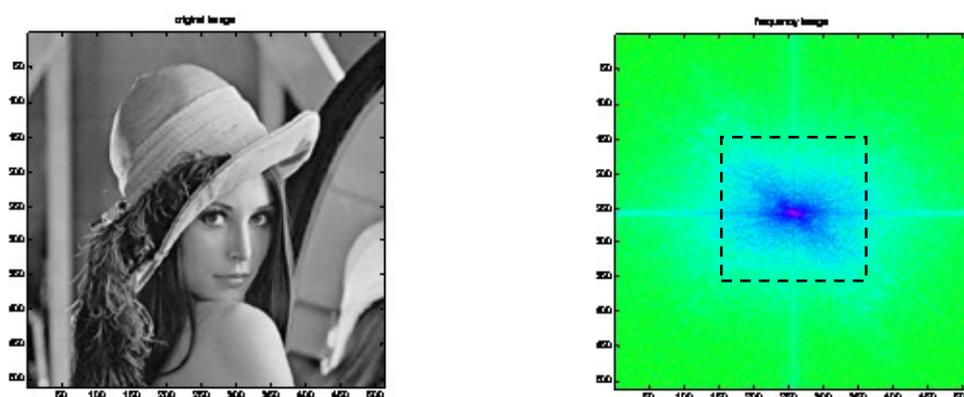


Figure IV-2: Transformée de Fourier de l'image Lena représentée sur une échelle logarithmique.

Dans ce travail nous utilisons pour la mise en œuvre du transformée en Curvelets, la valeur par défaut proposée par E. Candès, L. Damenet et al [17], cette valeur est dépendante de la taille de l'image d'entrée et donnée par :

$$J = \lceil \log_2(\min(N_1, N_2)) - 3 \rceil$$

Où $\lceil * \rceil$ est le plus petit entier supérieur ou égal à $[*]$. N_1 et N_2 sont successivement la longueur et la largeur de l'image en pixels.

Le nombre d'orientations N_θ est proportionnel à l'échelle à laquelle ils appartiennent, il s'accroît par l'augmentation d'échelle, de l'échelle grossière à la plus fine, il est égal à 2^{2j} Où $j = 1, 2, \dots, J$. De sorte qu'on obtienne plus de directivité en hautes fréquences où se trouve généralement l'information directionnelle dans l'image, qui correspond aux régularités géométriques le long des bords des objets.

Pour obtenir les coefficients de niveau externe le plus fin, l'algorithme offre le choix entre l'utilisation d'une ondelette non directionnelle ou de Curvelets. Pour une meilleure directivité, nous avons choisi dans ce travail, l'utilisation des Curvelets directionnelles malgré leur coût de calcul plus élevé.

IV.2.2 QUANTIFICATION :

Se basant sur le fait que la transformée en Curvelets est une décomposition parcimonieuse des signaux, ce qui signifie que les coefficients de grandes amplitudes représentent les informations les plus pertinentes dans l'image d'entrée (les structures géométriques dans notre cas). Nous proposons d'utiliser une quantification non uniforme par compression logarithmique, qui quantifie les coefficients de hautes amplitudes avec plus de précision afin de favoriser, lors de cette étape, les coefficients qui portent les informations directionnelles de l'image.

Le quantificateur peut être construit en inversant la technique M-Law. L'inversement dans ce cas signifie le changement d'ordre entre le bloc de compression et celui d'expansion du schéma de quantification par Companding introduit dans le chapitre précédent.

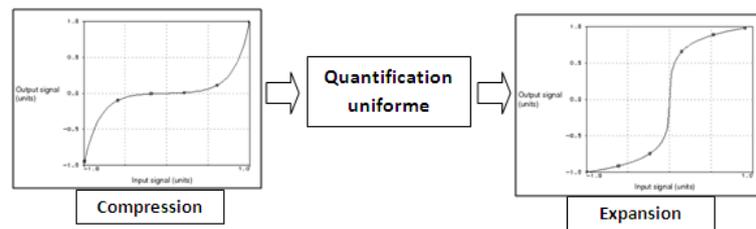


Schéma en blocs de Quantificateur μ -Law

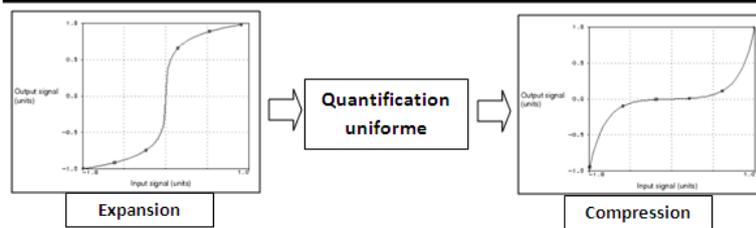


Schéma en blocs de Quantificateur proposée

Figure IV-3: Etapes de quantification non uniforme par μ -law inversée.

La fonction d'expansion F_{exp} est donnée par :

$$\begin{cases} F_{exp} = \text{sign}(x) \left(\frac{1}{\mu}\right) ((1 + \mu)^{|x|} - 1) \\ -1 \leq x \leq 1 \end{cases} \quad (\text{EQ : IV-1})$$

(où x est la valeur originale, et μ le facteur d'expansion.)

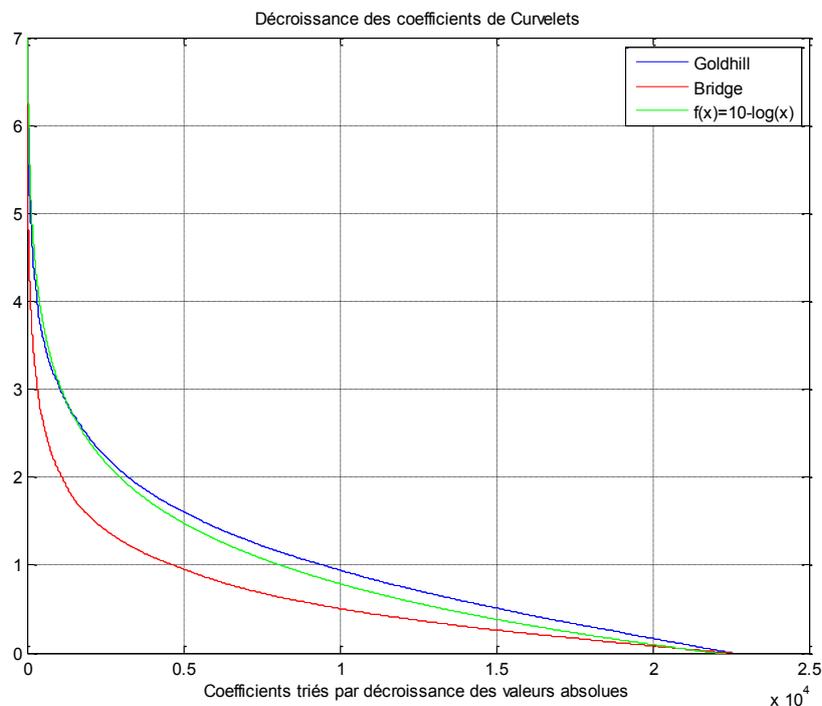


Figure IV-4: Caractéristique de décroissance logarithmique de valeurs absolues des coefficients de Curvelets pour les deux images "Goldhill" et "Bridge".

Cette fonction est bien adaptée à la décroissance rapide des amplitudes de coefficient de Curvelets (voir Figure IV-5). Elle sert à créer des intervalles de quantification avec des largeurs inversement proportionnelles à la distance de l'origine. On obtient donc une quantification avec plus grande précision, pour les coefficients de hautes amplitudes, tandis que les coefficients de faibles amplitudes sont quantifiés avec moins de niveaux de quantification. Un facteur d'expansion $\mu = 0$ correspond à l'utilisation d'une quantification uniforme.

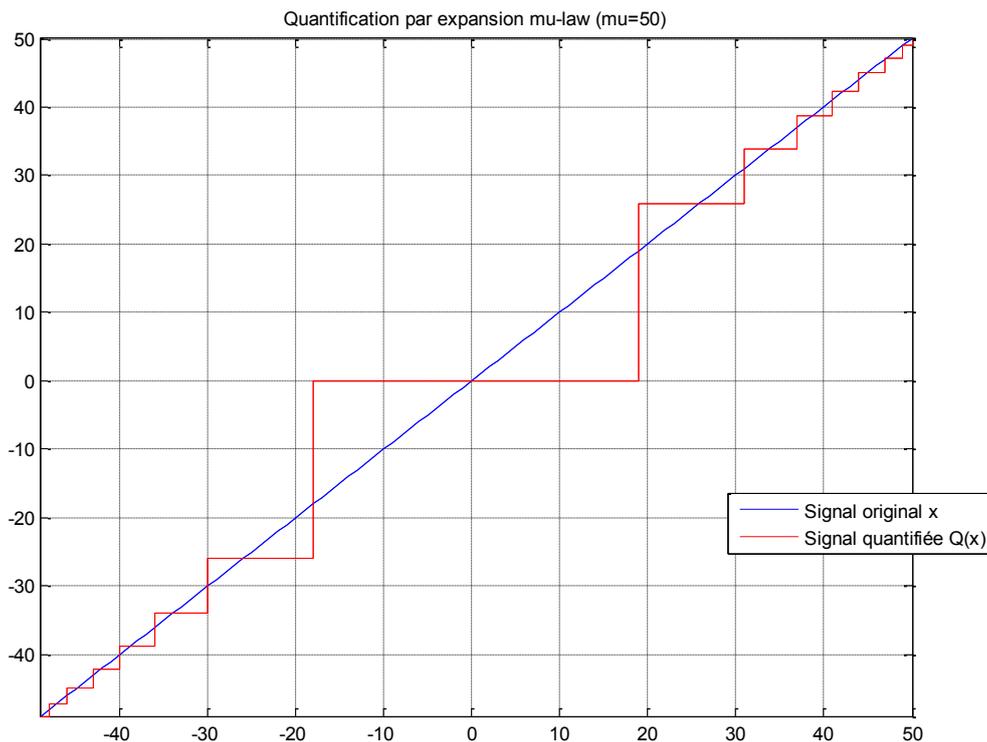


Figure IV-5: Illustration d'une quantification par expansion logarithmique

Le quantificateur uniforme utilisé dans le schéma de la Figure IV-4 est un quantificateur Midtread, à cause de l'importance de niveau de reconstitution $y_0=0$, qui sert à éliminer les valeurs proches de zéro. L'opération de quantification autour de l'origine est équivalent à un seuillage de coefficients par un seuil égal à la borne supérieure de l'intervalle B_0 ($T=b_0$).

La dé-quantification sera simplement l'inversement étape par étape du schéma de la Figure IV-4.

- Dé quantification par le décodeur uniforme Midtread.
- L'application de fonction de compression F_{comp} sur l'ensemble des valeurs dé-quantifiées.

IV.2.3 CODAGE ENTROPIQUE :

Après l'étape de quantification, on obtient un ensemble d'indices, caractérisé par une forte redondance statistique causée par les larges intervalles de quantification utilisés avec les coefficients de faible amplitude. L'intervalle de quantification autour de zéro est le plus large, ce qui implique l'élimination d'un grand nombre des coefficients.

Le codage entropique permet de réduire la taille de données obtenues sans introduire des distorsions en exploitant cette redondance statistique.

À cause de la forte asymétrie de distribution des indices à coder, le choix du codage arithmétique semble plus adéquat pour le schéma de compression proposé.

IV.3. SCHEMA EN BLOCS D'ALGORITHME DE COMPRESSION PROPOSEE :

L'algorithme de compression proposé sera donc décrit par la figure suivante :

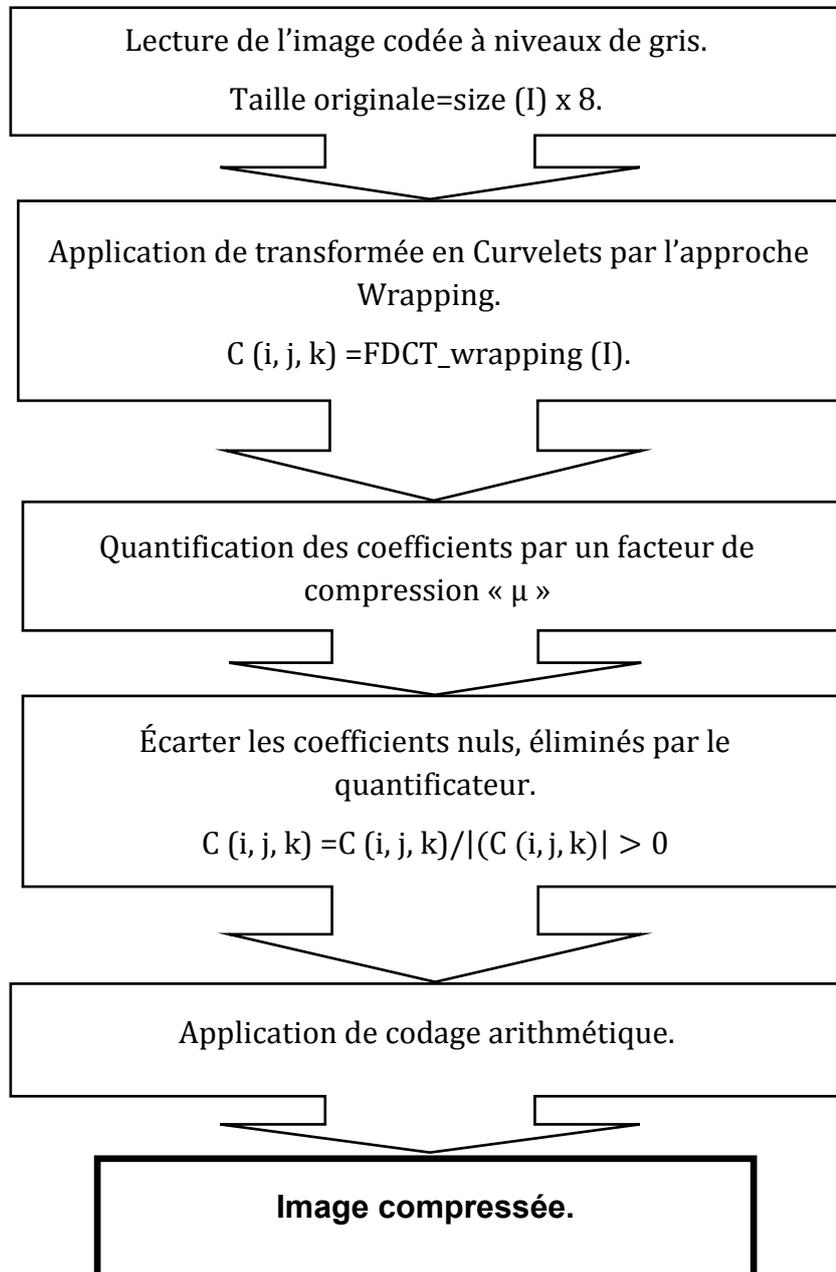


Figure IV-6: Méthode de compression proposée.

La reconstruction de l'image sera ensuite obtenue par l'application de l'opération inverse de chaque étape de méthode présentée ci-dessus.

IV.4. PRESENTATION DES RESULTATS

Afin de tester l'algorithme proposée, on l'applique sur un ensemble d'images présentant des régularités géométriques (textures, bâtiments, télédétection...) avec différents degrés d'expansion en variant le facteur μ . Les images utilisées sont des images à niveaux de gris et de taille 512×512 pixels [27], [28].

L'évaluation de la qualité d'une image après compression est une étape indispensable. L'appréciation visuelle est subjective, mais elle est toujours présente dans l'évaluation [29]. Le calcul du rapport signal bruit (PSNR) et le nombre de bits utilisés pour la reconstruction de l'image à partir de l'image originale sont les mesures objectives dans notre étude.

Le PSNR est défini comme suit :

$$PSNR = 10 \log_{10} \left(\frac{Pic^2}{MSE} \right) \quad (\text{EQ : IV-2})$$

Avec :

$$MSE = \frac{1}{M \times N} \sum_{m=1}^M \sum_{n=1}^N (I(m, n) - \hat{I}(m, n))^2 \quad (\text{EQ : IV-3})$$

Avec Pic est le maximum possible de la valeur du pixel d'image. Comme les pixels des images sont codés sur 8 bits, alors $Pic = 2^8 - 1 = 255$. L'unité du PSNR est le décibel (dB), plus le PSNR est important plus les images sont semblables.

Les résultats sont ensuite comparés avec ceux obtenues par d'autres algorithmes basées sur les transformées en Curvelets et ondelettes, ainsi que le standard JPEG2000.

IV.4.1 TAUX DE COMPRESSION Vs PSNR :

Les images de test sont compressées et décompressées par l'algorithme proposé. Pour chaque image, 9 valeurs de facteur d'expansion (μ) sont sélectionnées : [5, 14, 22, 30, 60, 90, 120, 150, 180]. Ses valeurs sont choisies pour évaluer la performance de la méthode proposée pour différent bit-rates. Les tableaux IV-1, IV-2 et IV-3 présentent les PSNR et bit-rates obtenues pour chacune des images de test.

Tableau IV-1: Résultats obtenus pour les images "House", "Goldhill"

Image	House								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	2.45	1.10	0.72	0.53	0.25	0.16	0.12	0.09	0.09
PSNR (dB)	32.27	28.28	26.64	25.54	23.14	21.91	21.09	20.51	20.10
Image	Goldhill								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	1.22	0.55	0.37	0.28	0.14	0.10	0.08	0.06	0.05
PSNR (dB)	34.75	31.80	30.75	29.75	27.63	26.91	26.25	25.68	25.24

Tableau IV-2: Résultats obtenus pour les images: "Sardegna ERS-1", "Bridge"

Image	Sardegna ERS-1								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	5.41	2.01	1.13	0.73	0.27	0.15	0.11	0.08	0.07
PSNR (dB)	33.61	26.58	23.32	21.46	18.84	18.02	17.59	17.29	17.07
Image	Bridge								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	0.80	0.39	0.27	0.21	0.11	0.08	0.06	0.05	0.04
PSNR (dB)	37.66	35.15	33.96	33.09	31.12	30.01	29.18	28.51	28.08

Tableau IV-3: Résultats obtenues pour les images: "Buildings", "Aerial"

Image	Buildings								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	1.82	0.83	0.55	0.42	0.21	0.14	0.10	0.08	0.07
PSNR (dB)	31.03	27.37	25.80	24.78	22.60	21.44	20.69	20.13	19.66
Image	Aerial								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	2.38	1.10	0.73	0.54	0.26	0.17	0.12	0.10	0.08
PSNR (dB)	34.02	30.11	28.36	27.16	14.82	23.69	22.89	22.32	21.87

Tableau IV-4: Résultats obtenus pour les images: "Barbara", "MRI (hd01)"

Image	Barbara								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	1.48	0.72	0.50	0.39	0.21	0.14	0.11	0.09	0.08
PSNR (dB)	35.83	32.51	30.99	29.99	27.71	26.37	25.47	24.80	24.31
Image	MRI(Hd01)								
Facteur d'expansion (Mu)	5	14	22	30	60	90	120	150	180
Bitrate (bpp)	0.95	0.45	0.31	0.23	0.12	0.08	0.06	0.05	0.04
PSNR (dB)	33.94	30.02	28.35	27.23	24.94	23.66	22.82	22.22	21.70

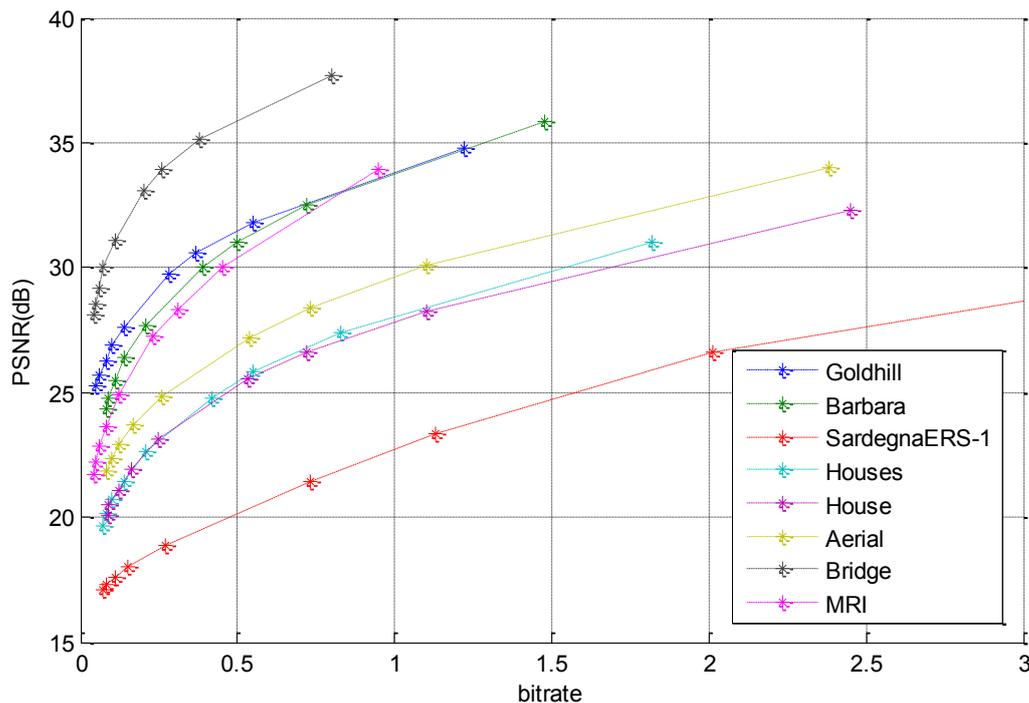


Figure IV-7: Performance de la méthode proposée, pour différentes images de test.

IV.4.2 TAUX DE COMPRESSION VS QUALITE VISUELLE

Pour les forts taux de compression, l'évaluation de la performance de compression, en utilisant le PSNR comme mesure objective de qualité d'image obtenue, ne donne pas une information sur le type de dégradation introduite. Une évaluation perceptuelle subjective est nécessaire.

Pour l'évaluation de la qualité visuelle des images compressées par la méthode proposée, les trois images de test : Sardegna-ERS 1, Aerial et Buildings sont compressés à moyens et forts taux de compression, les images décompressées sont ensuite présentées dans les figures IV-8, IV-9, IV-10.

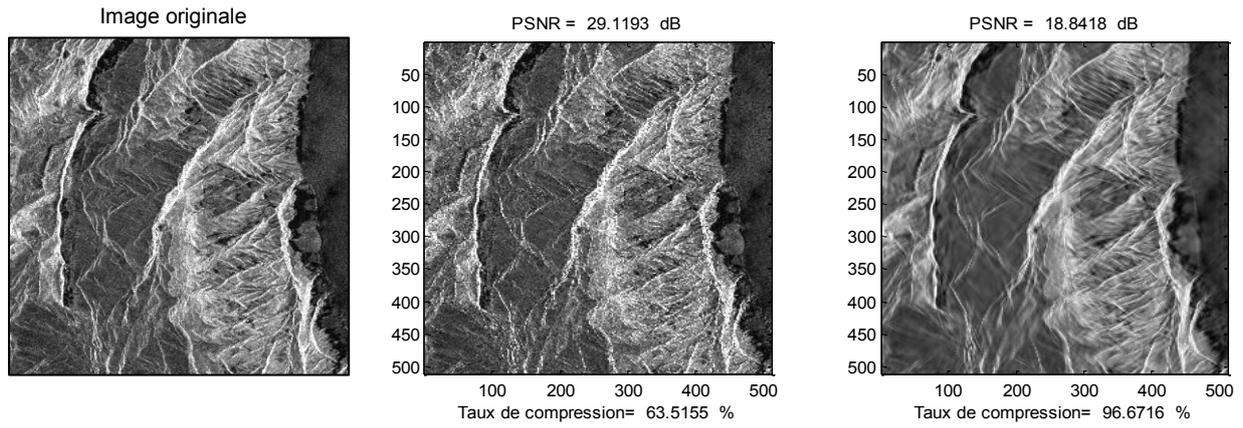


Figure IV-8 : Compression par Curvelets de l'image "Sardegna"

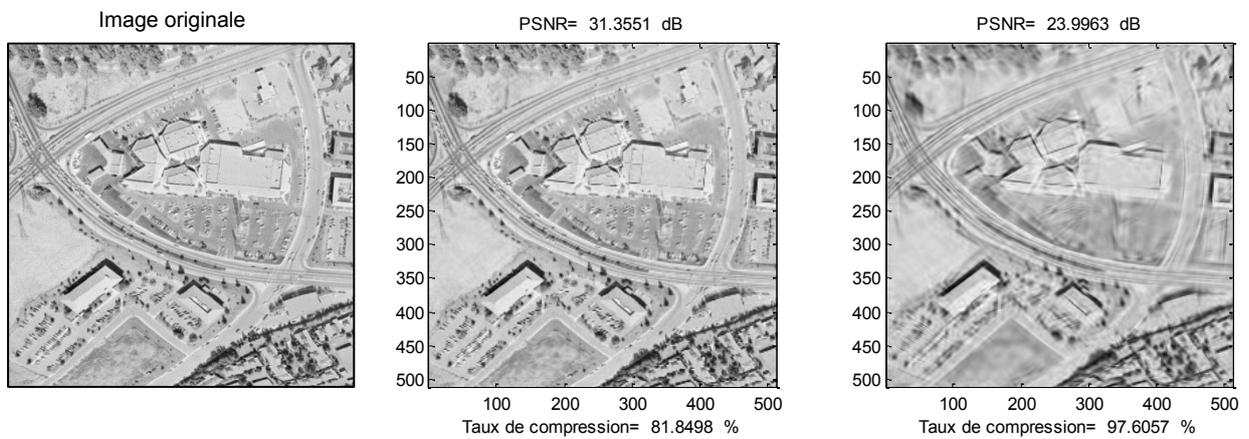


Figure IV-9 : Compression par Curvelets de l'image "Aerial".

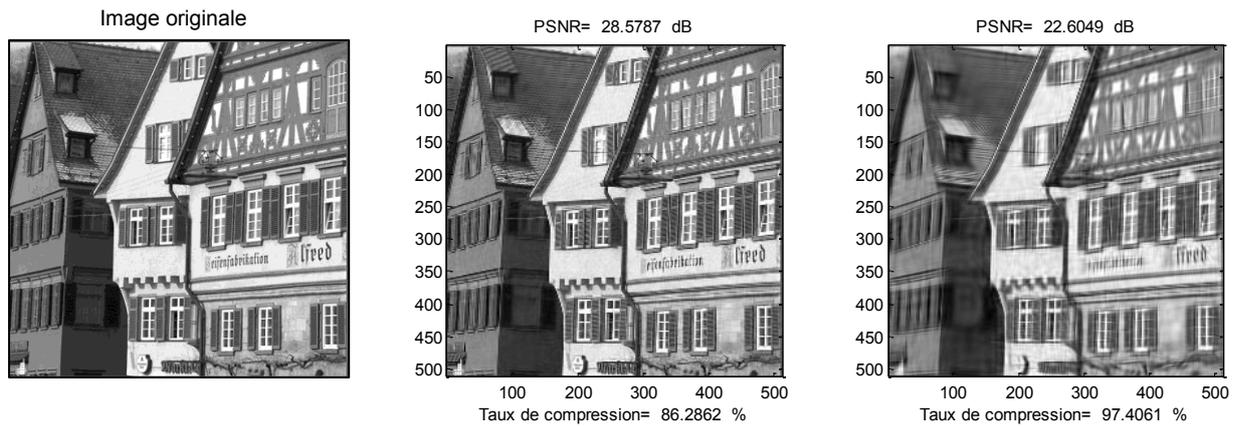


Figure IV-10: Compression par Curvelets de l'image "Buildings"

IV.4.3 ANALYSE ET DISCUSSIONS :

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

Les résultats obtenus pour les trois images sont montrés sous forme de tableaux : IV-1, IV-2, IV-3 et IV-4.

Pour la plupart des images les résultats des PSNR obtenues sont dans l'intervalle [25,30] pour des bit-rates variant entre 0.5 et 1,5 bit par pixel, ce qui est équivalent à des taux de compression entre 81, et 93 %.

La Figure IV-7 représente les résultats obtenues dans le même graphe ce qui nous permet de remarquer que le meilleur rapport PSNR bit-rate est celui de l'image « bridge » tandis que le moindre est celui de l'image « Sardegna-ERS 1 ».

Cela peut être expliqué par la domination des structures directionnelles dans l'image « bridge ». La forme élongée des Curvelets et leur bonne directivité permettent de représenter ces régularités par des coefficients Curvelets de grandes amplitudes. Tandis que, les zones homogènes et lisses sont représentées par des coefficients de faibles amplitudes.

La quantification logarithmique utilisée permet de favoriser les grandes amplitudes lors de la phase de codage, et la plupart des formes géométriques peuvent donc être conservées, contrairement, aux coefficients de faibles amplitudes qui sont éliminés.

Pour les autres images, nous pouvons constater que les rapports PSNR -bit rates obtenus dépendent des caractéristiques de régularité des structures présentes dans chaque image. Pour les images naturelles (Barbra, Goldhill, MRI), les singularités ponctuelles et les zones régulières sont plus dominantes que les structures directionnelles. C'est pourquoi les résultats obtenus sont meilleurs que ceux des images des bâtiments (House, Buildings). Ces dernières, se caractérisent par des régularités géométriques rectilignes le long des bords des objets, et en conséquence, la génération d'un nombre plus important de coefficients de Curvelets non nuls.

La perception visuelle des images de figures IV-8, IV-9, IV-10, nous permet de constater que les régularités géométriques le long des bords des objets sont les dernières informations à perdre dans la compression. En conséquence, la qualité d'image reste acceptable, même pour des taux de compression élevés comme le cas de l'image de la figure IV-8. Bien que les résultats de PSNR présentés sur la figure IV-7 montrent que les pires résultats sont ceux obtenus pour l'image Sardegna-Ers1, la perception visuelle montre que la majorité de distorsions est au niveau des singularités ponctuelles et pas dans les régularités curvilignes qui contiennent l'information la plus importante. La compression peut être vue dans ce cas comme une opération de dé-bruitage malgré le faible taux de PSNR obtenu.

Même si les valeurs sont mieux pour l'image Aerial, les distorsions introduites semblent plus dommageables, à cause de l'importance des petits objets éliminés lors de la compression. La même observation peut être constatée dans l'image de la Figure IV-10, 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.

IV.5. COMPARAISON AVEC D'AUTRES METHODES :

IV.5.1 COMPARAISON AVEC LES ONDELETES CLASSIQUES :

Dans ce qui suit, Nous comparons les résultats de compression pour différents types d'images tels que : bâtiment (House), télédétection (Aerial, Sardegna_ERS1), avec ceux de la compression par ondelettes classiques. La compression par ondelettes classiques est effectuée à l'aide de boîte d'outils MATLAB, « Wavelets toolbox ». Elle est basée sur une décomposition en ondelettes utilisant la fonction 'bior4.4', suivie par une quantification scalaire et un seuillage global de coefficients. Enfin, les valeurs obtenues sont codées par un codage entropique de Huffman.

Les résultats de PSNR et taux de compression obtenus sont présentés sous forme de courbes sur les figures IV-11, IV-12, IV-13.

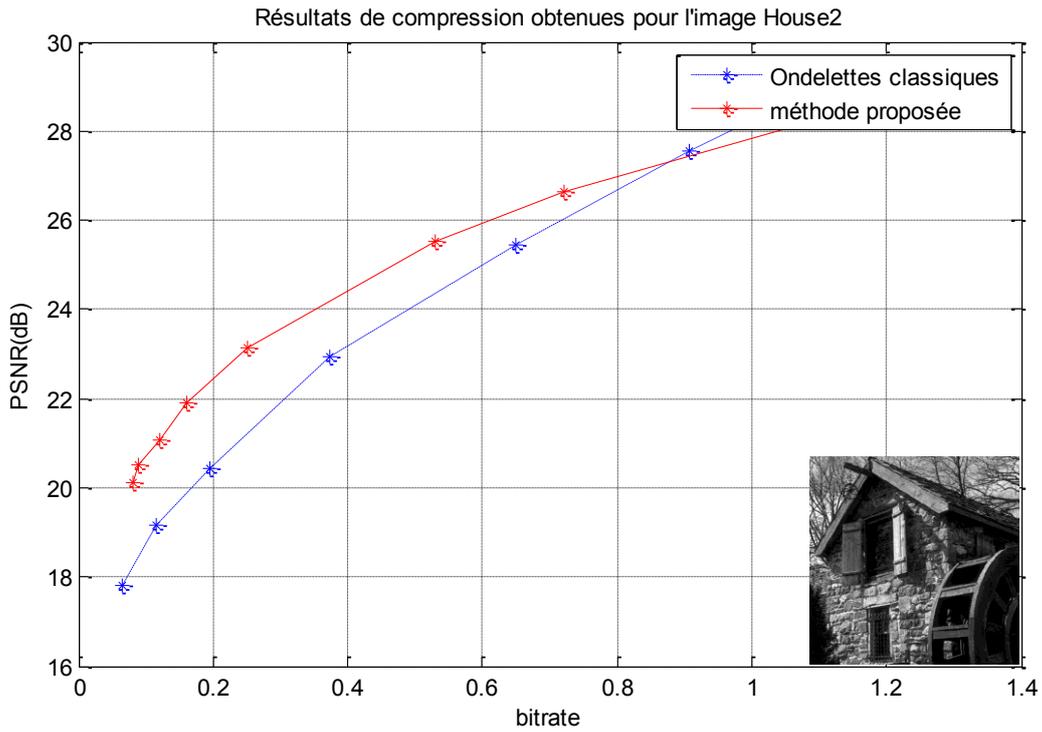


Figure IV-11: Compression de l'image "House"

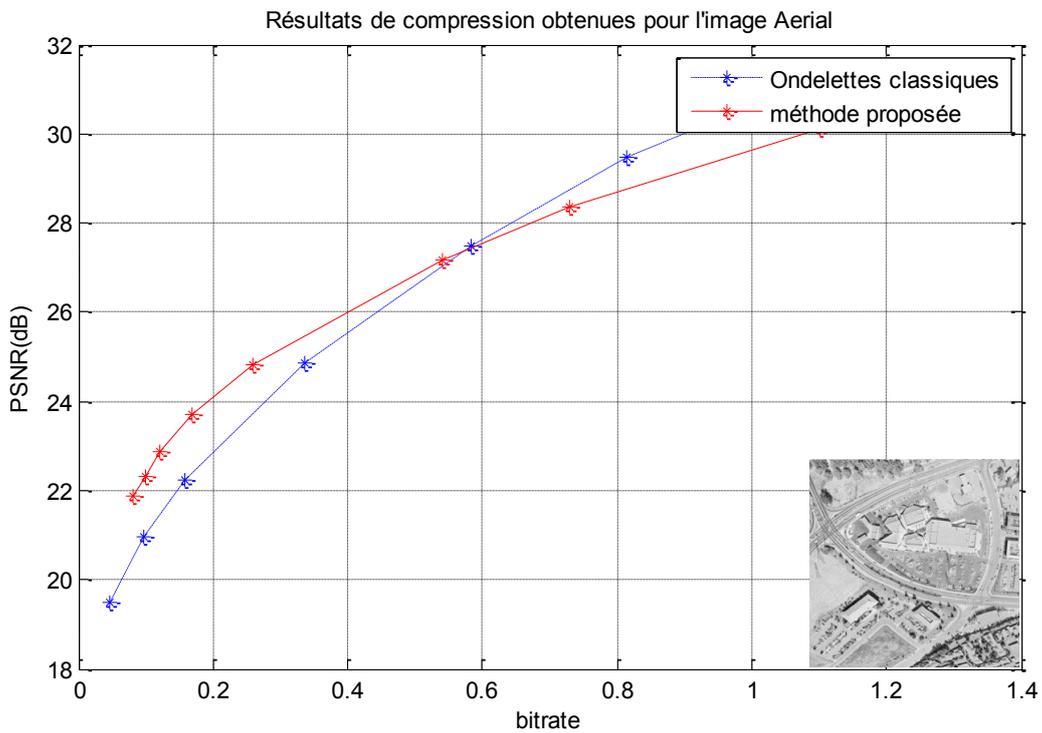


Figure IV-8: Compression de l'image "Aerial"

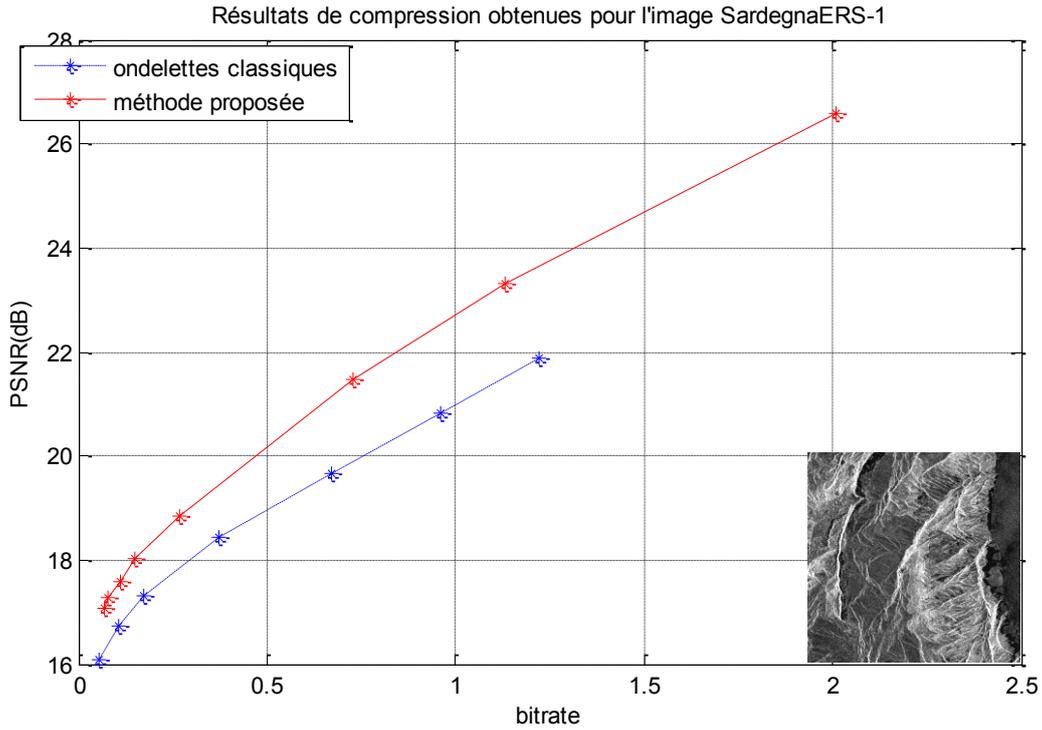


Figure IV-13: Compression de l'image "Sardegna-ERS1"

Pour l'évaluation de la qualité visuelle et la détermination du type de dégradation engendré par chaque méthode nous présentons quelques images compressées à des forts taux de compression (Figures IV-14, IV-15, et IV-16).

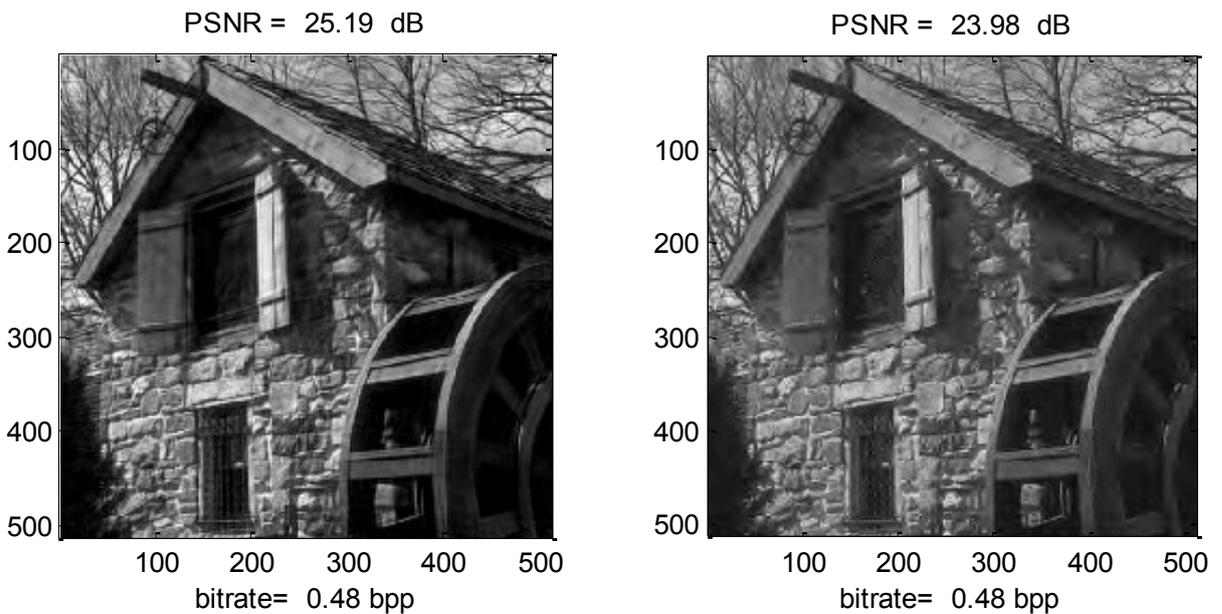


Figure IV-9: Comparaison pour l'image "House"

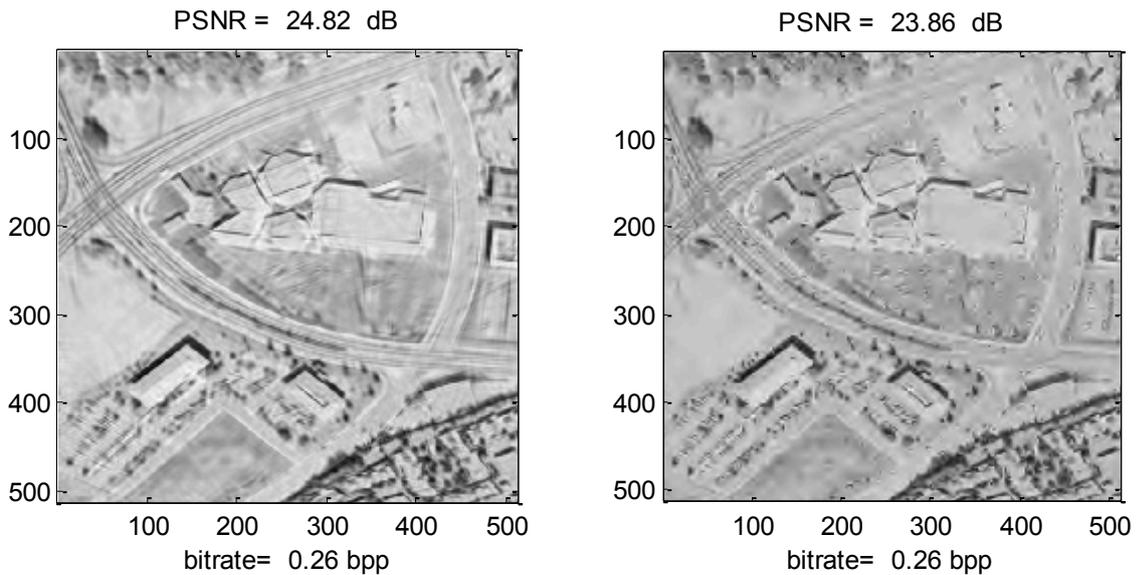


Figure IV-10: Comparaison pour l'image "Aerial"

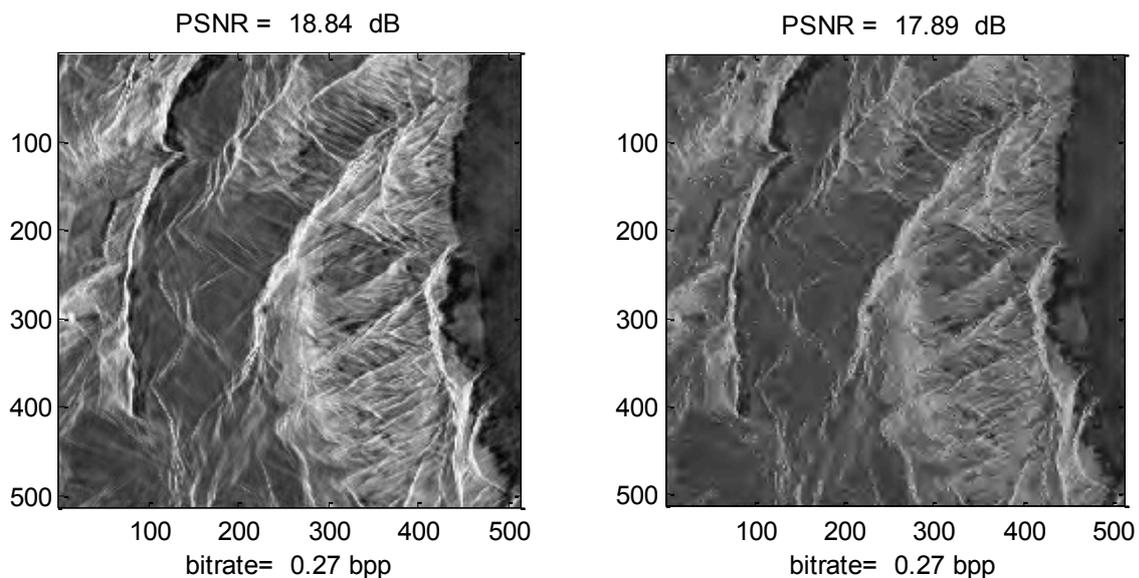


Figure IV-11: Comparaison pour l'image "Sardegna-ERS1"

Les figures présentées ci-dessus montrent que la méthode proposée donne généralement des meilleurs résultats en terme de rapport PSNR /RC par rapport à la compression par ondelettes classiques pour les forts taux de compression (94%, 96,75%, et 96,6% pour les images House, Ariel, et Sardegna successivement). Cette supériorité est plus claire pour les images dominées par des structures géométriques, comme le cas de l'image « Sardegna-ERS1 ».

La visualisation des images compressées nous permet de constater que les dégradations introduites sont principalement au niveau des zones lisses et homogènes, ainsi que dans les singularités ponctuelles. En conséquence, la méthode proposée permet de conserver l'information géométrique directionnelle, même pour les forts taux de compression.

L'interprétation de ce fait est que contrairement aux ondelettes classiques, les Curvelets sont des fonctions d'analyse multidirectionnelles se caractérisant par un support élongé dont la longueur est égale au carré de sa largeur. Elles sont donc, plus adaptées à la détection et représentation des singularités curvilignes, tandis que les ondelettes sont mieux dans la représentation des singularités ponctuelles.

La méthode proposée utilise un quantificateur scalaire non uniforme qui favorise les coefficients les plus significatifs, ce qui permet de conserver les régularités géométriques même pour les hauts taux de compression.

IV.5.2 TRAVAUX SIMILAIRES :

Dans ce qui suit nous comparons notre méthode de compression avec la méthode de compression par Curvelets proposée par G. Reddy et al [30]. 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 et les bit-rates obtenus, pour l'image « Barbara », par les deux méthodes, sont présentés sur la figure IV-17.

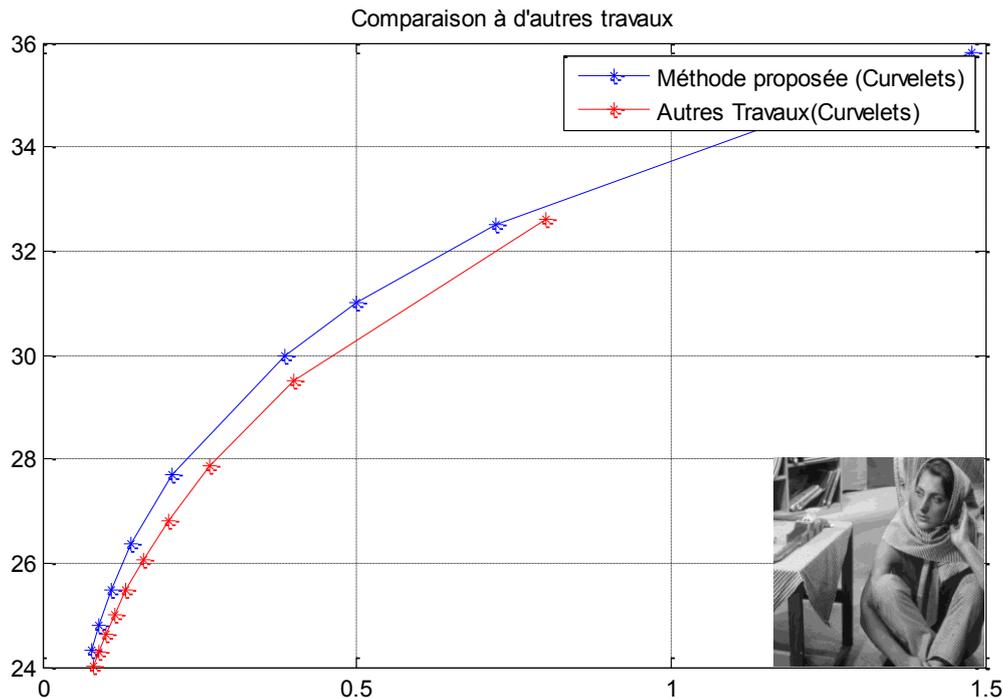


Figure IV-12: Résultats de compression pour l'image "Barbara"

En général, les résultats de PSNR obtenus par notre méthode proposée sont meilleurs pour les différents taux de compression. La performance de notre méthode est due à l'utilisation d'un quantificateur adapté à la parcimonie de coefficients de Curvelets. La distribution de niveaux de quantification suivant la caractéristique de décroissance des coefficients de Curvelets a permis de quantifier les coefficients les plus significatifs avec plus de précision, ce qui se traduit par la conservation des structures importantes dans l'image même pour les forts taux de compression.

IV.5.3 STANDARDS DE COMPRESSION ACTUELS :JPEG 2000 (ONDELETES).

La Figure IV-18 compare les résultats de PSNR et les bit-rates obtenus par notre méthode avec les résultats de compression de standard JPEG 2000 de l'image « Goldhill » apparus dans des travaux antérieurs effectués dans le laboratoire LESIA de l'université de Biskra [3].

Nous remarquons la supériorité du standard JPEG2000 pour les bit-rates supérieurs à 0.56 bit par pixel, tandis que les résultats de PSNR obtenus par la méthode proposée sont meilleurs pour les forts taux de compression.

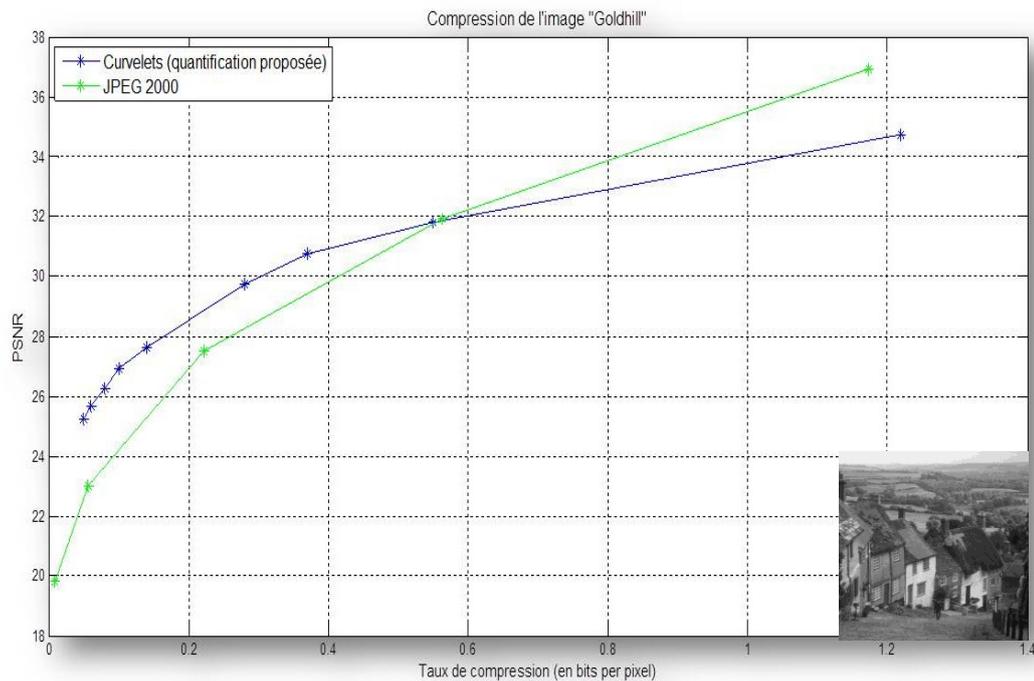


Figure IV-13: Performance de la méthode proposée contre le JPEG2000 pour l'image « Goldhill »

Ces résultats sont dus au fait que les ondelettes classiques, qui sont à la base de standard JPEG2000, sont par construction plus adaptées à la représentation des zones homogènes et lisses qui représentent la plus grande partie de l'image. Cependant, elles ne sont pas capables de détecter les régularités géométriques présentes dans l'image et elles les traitent comme des singularités ponctuelles ce qui se traduit par la génération d'un grand nombre de coefficients non nuls. Par contre les Curvelets représentent ces régularités plus efficacement grâce à la forme élongée et la directivité des Curvelets. En plus de l'intensité des singularités, les amplitudes de coefficients de Curvelets dépendent aussi de la longueur de la régularité géométrique, ainsi on obtient moins de coefficients de hautes amplitudes, mais ils sont plus significatifs.

IV.6. 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.

Malgré que, l'algorithme de compression proposé semble moins performant par rapport aux algorithmes basés sur les ondelettes classiques, pour les faibles taux de compression (inférieur à 94%), La compression à des forts taux de compression montre la supériorité de la méthode proposée pour la plupart des images testées. Elle permet de favoriser les informations géométriques, de sorte que, les structures curvilignes le long des contours et bords des objets dans l'image sont les dernières informations à perdre, même pour les forts taux de compression. La performance de la méthode proposée dépend donc de la présence et l'importance de ces régularités dans l'image à compresser.

CONCLUSION GENERALE

Nous nous sommes intéressés dans ce travail à la compression d'images fixes par ondelettes géométriques et plus précisément par Curvelets, pour améliorer les performances de la compression (PSNR et RC) par rapport aux performances des ondelettes classiques.

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 soient les dernières informations à perdre même pour les forts taux de compression. Cette technique permet au même temps d'améliorer l'entropie des coefficients obtenus et d'écarter les coefficients les moins significatifs.

Les performances de la méthode de compression proposée sont généralement satisfaisantes. Pour la plupart des images de test, le rapport de dégradation PSNR est compris entre 20 et 32 dB pour des RC compris entre 0.08 et 0.72 bpp (entre 91% à 99%) selon la nature des images traitées.

Bien que cette méthode de compression semble moins performante dans la compression des images naturelles par rapport aux ondelettes classiques pour les RC supérieurs à 0.8 bpp (inférieurs à 90%), la compression à des forts RC (supérieurs à 93%), montre la fiabilité de la méthode proposée par rapport à la compression par ondelettes classiques pour la plupart des images, avec une amélioration de l'ordre de 1 à 2 dB pour le PSNR.

La perception Visuelle montre que Les performances de la méthode proposée dépendent de la présence et de l'importance des régularités dans l'image à compresser. Ainsi, la qualité des images reconstruites est meilleure pour les images denses en structures géométriques (Bridge, Buildings, Sardegna-Ers1), surtout pour des taux de compression supérieurs à 97%.

De plus, les résultats obtenus par la méthode proposée sont meilleurs par rapport à ceux obtenus par l'algorithme de compression basée sur la transformation en Curvelets proposé récemment.

À partir de ces observations et des résultats obtenus dans ce travail, quelques perspectives intéressantes peuvent être envisagées pour approfondir ce travail de recherche ; nous citons notamment :

- L'étude de l'influence des nombres d'échelles et d'orientations de l'algorithme proposé.
- L'étude approfondie de la phase de codage statistique pour déterminer le meilleur codeur possible.
- L'extrapolation de la méthode proposée à la compression des images couleurs et les séquences vidéo.

BIBLIOGRAPHIE

1. **M.Shapiro, Jerome.** Embedded Image Coding Using Zero Trees of Wavelet Coefficients. *IEEE Transactions on Signal Processing*. December 1993, Vol. 41, pp 3445-3462.
2. **Amir Said, William A.Pearlman.** A New ,Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees. *IEEE Transactions on circuits and systems for video technology*. June 1996, Vol. 6, pp 243-249.
3. **S.Medouakh, Z-E. Baarir.** Study of the Standard JPEG2000 in Image Compression. *International Journal of Computer Applications*. March 2011, Vol. 18, pp 27-33.
4. **Seriaf, Chahira.** Extraction automatique de points d'intèret à base de la transformée en Contourlets non souséchantillonnée pour le recalage des images. *Thèse de Doctorat*. s.l. : Universite Mentouri Constantine, 2009.
5. **Ouafi, Abdelkrim.** Compression d'images fixes biomédicales par les transformées en ondelettes associées aux algorithmes de quantification vectorielle et de codage entropique. *Mémoire de Magister*. s.l. : Université Mohamed Khider de Biskra, 2001.
6. **K.P. Soman, K.I.Ramachandran, N.G.Resmi.** *Insight into Wavelets from theory to practice*. s.l. : PHI Learning, 2010. 81-203-2650-4.
7. **L.Donoho, David.** *Fast Ridgelet Transforms in dimation 2*. Stanford : Departement of statistics ,Stanford University, October 1997. Rapport technique .no514.
8. **Emmanuel, J.Candes and David, L.Donoho.** Ridgelets: a key to higher-dimensional intermittency? *Philosophical Transactions of the Royal Society. A: Mathematical, Physical and Engineering sciences*. 1999, Vol. 357, pp 1-15.
9. **E, Candès and D.L, Donoho.** Curvelets: A Surprisingly Effective Nonadaptive Representation For Objects with Edges. July 1999.
10. **David L. Donoho, Emmanuel J. Candès.** New Tight Frames of Curvelets and Optimal Representations of Objects with C2 singularities. *Communications on Pure and Applied Mathematics*. 2003, Vol. 57, pp 219-266.
11. **David L, Donoho and Mark, R.Duncan.** *Digital Curvelet Transform: Strategy, Implementation and Experiments*. Stanford : Department of Statistics Stanford University, November, 1999. Rapport technique.
12. **Jean-Luc Starck, Emmanuel J. Candès,David L. Donoho.** The Curvelet Transform for Image Denoising. *IEEE Transactions on Image Processing*. June 2002, Vol. 11, pp 670-684.
13. **Jean-Luc Starck, Fionn Murtagh, Emmanuel J. Candès, et David L. Donoho.** Gray and Color Image Contrast Enhancement by the Curvelet Transform. *IEEE Transactions On Image Processing*. June 2003, Vol. 12, pp 706-716.
14. **M. Choi, R.Y. Kim, et M.G. Kim.** Curvelet transform for image fusion. *International Society for Photogrammetry and Remote Sensing (ISPRS)*. July 2004, N° 931. pp 59–64.
15. **Jean-Luc Starck, Fionn Murtagh, Jalal Fadili.** Sparse image and signal processing : wavelets, curvelets, morphological diversity. Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore : Cambridge University Press, 2010. pp 100-104. 978-0-521-11913-9.

16. **Emmanuel, J.Candès and Laurent, Demanet.** The Curvelet Representation of Wave Propagators is Optimally Sparse. *Communications on Pure and Applied Mathematics*. 2005, Vol. 58, no. 11 pp 1472-1528.
17. **Emmanuel Candes, Laurent Demanet, David Donoho, et Lexing Ying.** *Fast Discrete Curvelet Transforms*. California : California Institute of Technology, July 2005, revised March 2006. Rapport technique.
18. **M.S. Joshi, R.R. Manthalkar, and Y.V. Joshi.** Image Compression Using Curvelet, Ridgelet and Wavelet Transform, A Comparative Study. *ICGST International Journal on Graphics, Vision and Image Processing (GVIP)*. October 2008, Vol. 8, pp 25-34.
19. **Kamlesh Gupta, Sanjay Silakari.** Performance Analysis Of Image Compression Using Curvelets Transform. *Journal of Signal and Image Processing*. July 2011, Vol. 2, pp 13-18.
20. **Lloyd, Stuart .P.** least Squares Quantization in PCM. *IEEE Transactions on information theory*. March 1982, Vols. IT-28, pp 129-137.
21. **Pu, Ida Mengyi.** Fundamental Data Compression. Oxford : Elsevier's Science and Technology, 2006. pp 180-181. ISBN-10:0-7506-6310-3.
22. **David A, Huffman.** *A Method for the Construction of Minimum-Redundancy Codes*. s.l. : Proceedings of the I.R.E., September 1952. pp 1098-1101.
23. **Said, Amir.** *Introduction to Arithmetic Coding - Theory and Practice*. Palo Alto -USA- : Imaging Systems Laboratory -HP Laboratories -, April 21, 2004. Rapport technique. HPL-2004-76.
24. **Asadollah Shahbahrami, Ramin Bahrapour, Mobin Sabbaghi Rostami, Mostafa Ayoubi Mobarhan.** Evaluation of Huffman and Arithmetic Algorithms for Multimedia Compression Standards. *International Journal of Computer Science, Engineering and Applications*. N°4, August 2011, Vol. 1, pp 34-47.
25. **Med Karim Abdmouleh, Atef Masmoudi, et Med Salim Bouhlel.** A New Method Which Combines Arithmetic Coding with RLE for Lossless Image Compression. *Journal of Software Engineering and Applications*. January 2012, Vol. 5, pp 41-44.
26. **Emmanuel Candès, Laurent Demanet , Lexing Ying.** *CurveLab Toolbox*. Pasadena : California Institute of Technology, 2005-2007. Version 2.0.3.
27. Dataset of standard 512X512 grayscale test images. *Departement Ciencias de Ciencias de la Computación e Inteligencia Artificial de l'Université de Grenade*. [Online] [Cited: Octobre 7, 2012.] <http://decsai.ugr.es/cvg/CG/base.htm>.
28. **Wei Xu, and Ian Cumming.** A Region-Growing Algorithm for InSAR Phase Unwrapping. *IEEE Transactions on geosience and remote sensing*. JANUARY 1999, Vol. 37, 1.
29. **Foued Derraz, Mohamed Beladgham, M'hamed Khelif.** *Mesure Objective de la Qualité d'Image Médicale Dérivée de l'Index de Similarité Structurale*. Tlemcen-ALGERIE : Laboratoire de Génie Biomédicale. Rapport technique.
30. **G.Jagadeeswar Reddy, T.Jayachandraprasad, M.N.Giriprasad, M. Madhavi Latha et T. Satya Savithri.** Curvelets with new quantizer for image compression. *Global journal of computer science and technology*. 12, April 2012, Vol. 12, pp 64-69.