



HAL
open science

Compression d'images complexes avec pertes: application à l'imagerie Radar

Cédric Valade

► **To cite this version:**

Cédric Valade. Compression d'images complexes avec pertes: application à l'imagerie Radar. domain_other. Télécom ParisTech, 2006. English. NNT: . pastel-00002064

HAL Id: pastel-00002064

<https://pastel.hal.science/pastel-00002064>

Submitted on 17 Jan 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**THESE DE DOCTORAT
DE L'ECOLE NATIONALE SUPERIEURE DES
TELECOMMUNICATIONS**

présenté par
Monsieur **Cédric VALADE**

pour obtenir le grade de
**DOCTEUR DE L'ECOLE NATIONALE SUPERIEURE DES
TELECOMMUNICATIONS**

Domaine :
Traitement d'Images et des Signaux

Sujet de la thèse :
**Compression d'images complexes avec pertes :
application à l'imagerie Radar**

Thèse soutenue le 8 décembre 2006 devant le jury composé de :

Rapporteur :	M. René Garello
Rapporteur :	M. Marc Antonini
Examineur :	M. Henri Maitre
Examineur :	M. Emmanuel Trouvé
Directeur de thèse :	M. Jean-Marie Nicolas
Co-directeur de thèse :	M. Jacques Blanc-Talon

ECOLE NATIONALE SUPERIEURE DES TELECOMMUNICATIONS
46, Rue Barrault, 75634 PARIS Cedex 13(FRANCE)



Remerciements

Cette thèse a été financée sur une bourse DGA-CNRS.

Je tiens tout d'abord à remercier mon directeur de thèse, Jean-Marie Nicolas et mon co-directeur de thèse, Jacques Blanc-Talon qui ont tous deux accepté d'encadrer cette thèse et de m'avoir fait profiter de leur expérience et leur savoir-faire.

Je tiens à remercier chaleureusement Marc Antonini, de l'Université de Nice Sophia AntiPolis, René Garelo de l'ENST Bretagne qui ont accepté de consacrer du temps à rapporter ce document. Mes remerciements vont également à Henri Maitre de l'ENST Paris et Emmanuel Trouvé de l'Université de Savoie, pour avoir accepté de participer à mon jury de thèse.

Mes remerciements s'adressent également à tout le département GIP du Centre Technique d'Arcueil qui m'a accueilli tout au long de ces années. Une mention particulière à Jérôme Gilles pour ses multiples conseils. Je n'oublie pas non plus le laboratoire TSI de l'ENST et l'équipe Radar.

Durant ces trois années, j'ai pu enseigner dans le Département Traitement du Signal et Informatique et Réseaux de l'ESIEE. Je tiens à remercier tous les enseignants qui m'ont proposé de travailler avec eux et pour la confiance qu'ils m'ont témoigné : Nadia Madaoui, Jean-François Bercher, Ferial Virolleau et Laurent Perroton.

Je pense également à Fabrice Humblot pour ses conseils avisés. Un grand merci à toute ma famille, ma belle-famille et à mes amis Virgine, Karmène, Carine, Cédric, Arnaud, Lionel, Bruno qui m'ont toujours soutenus et encouragés. Enfin, je tiens à exprimer toute ma reconnaissance à ma tendre My Linh pour son soutien précieux.

Table des matières

Remerciements	3
Liste des acronymes	9
Introduction	11
1 Imagerie Radar RSO : généralités	15
1.1 Principe du radar à synthèse d'ouverture	16
1.1.1 Longueurs d'onde	17
1.1.2 Résolution en distance avant synthèse de l'image	17
1.1.3 Résolution en azimuth avant synthèse de l'image	18
1.2 Synthèse de l'image	19
1.2.1 Synthèse en distance : compression d'impulsion	19
1.2.2 Synthèse en azimuth	20
1.2.3 Migration et dépointage	22
1.3 Les données issues du radar à synthèse d'ouverture	25
1.3.1 Données brutes	25
1.3.2 Les images SLC (Single Look Complex)	25
1.3.3 Les données multivues	26
1.4 Chatoiement et modélisation statistique	27
1.4.1 Chatoiement pleinement développé	27
1.4.2 Modèles non-gaussiens	29
1.5 Interférométrie	30
1.5.1 Information de Phase	31
1.5.2 Construction de l'interférogramme	31
1.6 Conclusion	34
2 Compression d'images, état de l'art et application aux images RSO	35
2.1 Introduction à la compression et critères d'évaluation	36
2.1.1 Taux de compression	36
2.1.2 Distorsion et critère qualité	37
2.1.3 Complexité	37
2.1.4 Robustesse	38
2.2 Système de compression avec pertes	38
2.3 Compression sans perte et codage entropique	39
2.3.1 Entropie	39
2.3.2 Condition du préfixe et Inégalité de Kraft	40
2.3.3 Algorithme de Huffman	42
2.3.4 Codage Arithmétique	43

2.3.5	Algorithme LZW	46
2.3.6	Codage RLE (Run Length Encoding)	46
2.3.7	Conclusion	46
2.4	Quantification	47
2.4.1	Quantification scalaire	47
2.4.2	Quantification scalaire à contrainte entropique	51
2.4.3	Quantification scalaire progressive	52
2.4.4	Allocation de bits	54
2.4.5	Quantification vectorielle	56
2.5	Codage par Transformée	62
2.5.1	Transformée orthogonale	62
2.5.2	Transformée en sous-bandes et transformée en ondelettes	65
2.6	Conclusion	81
3	Vers une méthodologie de compression adaptée à la représentation des données complexes	83
3.1	Représentation dans le plan complexe et mesure de distorsion	84
3.1.1	Représentation cartésienne et polaire	84
3.1.2	Distorsion quadratique	85
3.2	Représentation des données SLC et méthodologies d’approches pour la compression	87
3.2.1	Approche 1 : Représentation cartésienne avant et après transformée	87
3.2.2	Approche 2 : Représentation cartésienne avant transformée et polaire après transformée	90
3.2.3	Approche 3 : représentation polaire avant transformée et compression indépendante module/phase	92
3.2.4	Approche 4 : Représentation polaire avant transformée et compression de la phase dépendante du module	94
3.2.5	Conclusion	96
3.3	Loi de Fisher pour les données SLC et caractérisation avant/après DWT .	97
3.3.1	Caractérisation statistique de l’image d’amplitude	97
3.3.2	Modèle de Fisher appliqué après DWT de l’image SLC	108
3.4	Conclusion	111
4	Compression d’images RSO complexes basée sur la transformée en ondelettes	113
4.1	Compression de l’image SLC par la méthode TCE-IQC	114
4.1.1	Généralités	114
4.1.2	Algorithme TCE	116
4.1.3	Application	122
4.2	Quantification polaire des coefficients d’ondelettes : <i>Wavelet ECUPQ</i> . . .	124
4.2.1	Quantification Polaire	124
4.2.2	Quantification polaire à contrainte entropique des coefficients d’ondelettes	130
4.2.3	Application	135
4.3	Compression d’images SLC par la méthode TCEAC-CSPQ	138
4.3.1	Quantification Scalaire de la Phase conditionnellement à l’Amplitude	139
4.3.2	Application	146

4.4	Algorithme TCEAC-IPQ	149
4.5	Conclusion	150
5	Compression d'images RSO complexes basée sur la transformée en paquets d'ondelettes	151
5.1	Les paquets d'ondelettes	151
5.2	Choix de la meilleure base	154
5.2.1	Fonctions de coût	154
5.2.2	Allocation dynamique de débit et recherche de la meilleure base . .	155
5.3	Paquets d'ondelettes et algorithme WP-ECUPQ	158
5.3.1	Principe	158
5.3.2	Application	160
5.4	Paquets d'ondelettes et algorithme TCE	162
5.4.1	Algorithme : WP-TCE	162
5.4.2	Algorithme WPTCE-IQC appliqué à la compression d'images RSO	165
5.4.3	Application à l'image d'amplitude et aux images test standard . . .	168
5.5	Conclusion	169
6	Résultats et étude comparative	171
6.1	Introduction	171
6.2	Critères de qualité	172
6.2.1	Critères de qualité basés sur l'erreur quadratique moyenne	172
6.2.2	Corrélation Complexe spatiale	174
6.2.3	Histogrammes	175
6.2.4	Critères de qualité RSO	175
6.2.5	Critères de qualité sur les interférogrammes	176
6.3	Compression de la scène "Istres"	178
6.3.1	Critères d'erreur quadratique	179
6.3.2	Analyse des résultats	182
6.3.3	Carte de corrélation complexe	183
6.3.4	Histogramme de l'image d'amplitude et de l'image de partie réelle .	186
6.3.5	Spectre en distance et en azimut	189
6.3.6	Analyse des résultats	191
6.4	Compression de la scène "Baux de Provence"	192
6.4.1	Interférogrammes reconstruits après décompression	192
6.4.2	Considérations liées à la complexité	203
6.5	Conclusion	204
	Conclusion	205
	Annexes	209
.1	Extraction de la courbure convexe	210
.2	Coefficient de corrélation complexe moyen/coefficient de variation : scène extraite de la ville d'Istres	211
.3	Coefficient de corrélation complexe moyen/coefficient de variation : scène extraite des Baux de Provence	212
.4	Carte de corrélation complexe : scène extraite des Baux de Provence . . .	213
	Bibliographie	215

Liste des acronymes

- CPQ** : Conventionnal Polar Quantization
- DCT** : Discrete Cosine Transform
- DRI** : Détection-Reconnaissance-Identification
- DWT** : Discrete Wavelet Transform
- EQM** : Erreur Quadratique Moyenne
- ECCPQ** : Entropy Constrained Conventionnal Polar Quantization
- ECRQ** : Entropy Constrained Rectangular Quantization
- ECUPQ** : Entropy Constrained Unrestricted Polar Quantization
- ECVQ** : Entropy Constrained Vector Quantization
- FFT** : Fast Fourier Transform
- GLA** : Generalized Lloyd Algorithm
- PRF** : Pulse Repetition Frequency
- PSFQ** : Progressive Space Frequency Quantization
- RSB** : Rapport Signal sur Bruit
- RSO** : Radar à Synthèse d'Ouverture
- PWT** : Packet Wavelet Transform
- RQ** : Rectangular Quantization
- SLC** : Single Look Complex
- SFQ** : Space Frequency Quantization
- SPIHT-IQC** : Set Partioning In Hierarchical Trees and In phase and Quadrature Coding
- TCE** : Tarp-filter-based system and Classification of coefficients to achieve Embedding
- TCEAC-IPQ** : Tarp-filter-based system and Classification of coefficients to achieve Embedding applied to Amplitude Coding and Independant Phase Quantization
- TCEAC-CSPQ** : Tarp-filter-based system and Classification of coefficients to achieve Embedding applied to Amplitude Coding and Conditional Scalar Phase Quantization
- TCE-IQC** : Tarp-filter-based system and Classification of coefficients to achieve Embedding and In phase and Quadrature Coding
- TKL** : Transformée de Karhunen-Loeve
- UPQ** : Unrestricted Polar Quantization
- WECUPQ** : Wavelet Entropy Constrained Unrestricted Polar Quantization
- WP-ECUPQ** : Wavelet Packet Entropy Constrained Unrestricted Polar Quantization
- WPTCE-IQC** : Wavelet Packet Tarp-filter-based system and Classification of coefficients to achieve Embedding and In phase and Quadrature Coding

Introduction

Le Radar à Synthèse d'Ouverture (RSO) est un système d'imagerie active aux applications multiples de télédétection (surveillance de territoire, études des mouvements de terrain, océanographie, glaciologie, détection de cibles...). Installés à bord d'aéronefs ou de satellites, les systèmes actuels permettent d'imager des zones à haute résolution. Les volumes de données générés sont très importants et les capacités limitées des canaux de transmission et/ou de stockage forment un sous-ensemble critique des systèmes d'observation. De ce fait, la solution classiquement retenue est de recourir à la compression des données.

Le système radar cohérent génère des données de forte dynamique où les pixels issus de la formation de l'image sont des données complexes au sens mathématique du terme, c'est-à-dire constitués d'une valeur d'amplitude et de phase. Historiquement, les applications thématiques portaient principalement sur l'exploitation de l'image d'amplitude qui fournit l'essentiel de l'information visuelle. Néanmoins, le développement récent de techniques d'interférométrie et de polarimétrie, s'appuie sur l'exploitation de l'information de phase des images RSO.

L'acquisition des "données brutes" permet la formation d'une image RSO à partir du signal rétro-diffusé vers l'antenne du capteur. Le processus de formation d'une image nécessite un traitement long et complexe des données. C'est la raison pour laquelle, la synthèse de l'image n'est pas réalisée en temps réel mais seulement après transmission des données brutes du capteur vers la station au sol. Compte-tenu des volumes élevés de données mis en jeu, de nombreuses études ont été consacrées à la compression des données brutes de manière à réduire le débit de transmission. Citons notamment le célèbre algorithme BAQ (*Bloc Adaptive Quantization*) utilisé sur la sonde Magellan.

Désormais, les progrès technologiques en terme de puissance de calcul et d'algorithmie permettent d'envisager la synthèse de l'image à bord du capteur. La conception d'un algorithme de compression sur les images synthétisées permettra alors un dimensionnement des canaux de transmission adapté au débit requis pour la transmission des données compressées. Bien qu'il soit possible de compresser séparément les images d'amplitude et de phase, la prise en compte du caractère complexe de l'image ouvre de nouvelles perspectives peu explorées jusqu'alors.

Il existe deux grandes classes de méthodes de compression/décompression (CoDec) : les méthodes de compression sans perte qui permettent de reconstruire l'image sans dégradation mais avec un faible taux de compression, se distinguent des méthodes de compression avec pertes qui introduisent une dégradation irréversible sur les pixels de l'image. Les méthodes de compression avec pertes autorisent cependant des taux de compression bien plus élevés.

Dans le cadre de cette thèse, nous nous intéresserons aux méthodes de compression avec pertes pour lesquelles la problématique principale consiste à trouver le meilleur compromis entre le taux de compression et la dégradation introduite sur l'image décompressée. Dans le cadre des images RSO, l'évaluation de la qualité reste un problème ouvert car dépendant de l'application et des scénarii d'utilisation de l'image. En outre, le rendu psycho-visuel ne suffit pas à apporter une réponse pertinente et satisfaisante pour l'ensemble des applications visées.

Les propriétés des images RSO sont fondamentalement différentes de celles des images optiques car affectées par le "chatoiement" résultant du processus de formation d'imagerie cohérente. Ce phénomène donne un aspect poivre et sel à l'image et est à l'origine d'une composante de bruit multiplicatif qui dégrade fortement la lisibilité de l'image. La faible corrélation spatiale des pixels rend les techniques de compression répandues dans le domaine optique beaucoup moins efficaces. L'objectif de cette étude consiste à proposer un système de compression avec pertes adapté aux images RSO et prenant en compte le caractère complexe des pixels. Nous disposons à cet effet d'un jeu d'images acquises par le capteur RAMSES de l'ONERA en bande X de résolution métrique à sub-métrique.

L'étude présentée dans ce mémoire s'articule autour de 6 chapitres :

- Nous consacrons en particulier le **chapitre 1** à la description des principes de l'imagerie RSO. La caractérisation de ces images nous permet d'aborder la problématique de la compression avec une palette d'approches spécifiques. Nous évoquons les principes de la formation de l'image RSO avant de nous intéresser plus spécifiquement aux modèles statistiques de caractérisation des images. Nous abordons par exemple les principes de l'interférométrie radar, utilisée dans le cadre de cette thèse comme un outil d'évaluation de la qualité de reconstruction de l'image de phase après décompression.
- Le **chapitre 2** est dédié à l'introduction des principales méthodes de compression d'images fixes présentées dans la littérature avec un état de l'art ponctuel sur les méthodes de compression appliquées à l'imagerie RSO. Nous y développons en dernière partie les principes de la transformée en ondelettes qui sera la clé de voûte des techniques de compression que nous proposons.
- Le **chapitre 3** présente d'une manière générale les méthodes de compression proposées dans le cadre de cette étude. Nous suggérons différents schémas de compression dépendants de la représentation du signal dans le plan complexe. Nous nous focalisons essentiellement sur des techniques basées sur la transformée en ondelettes discrètes et un ensemble de modèles statistiques spécifiques à la caractérisation des

images RSO.

- Le **chapitre 4** s'inscrit directement dans la continuité du chapitre 3. En outre, nous décrirons trois algorithmes portant sur la quantification polaire des coefficients d'ondelettes ainsi que sur la quantification de l'image de phase conditionnellement aux pixels de l'image d'amplitude. Nous y étudions également l'adaptation d'une technique de quantification progressive à la compression d'images complexes.
- Dans le **chapitre 5**, nous proposons d'étendre les techniques développées au chapitre 4 à la transformée en paquets d'ondelettes. A cet effet, deux algorithmes correspondant à deux approches de quantification différentes sont présentés. Nous verrons que cette transformée adaptative permet d'obtenir les meilleurs résultats en terme de qualité de reconstruction de l'image.
- Enfin, le **chapitre 6** est consacré à l'évaluation des algorithmes présentés dans ce manuscrit. Nous proposons en particulier un panel de critères objectifs indépendamment de toute application thématique. La dernière partie de ce chapitre traite plus spécifiquement de l'évaluation de la qualité des images décompressées sur la construction d'interférogrammes.

Chapitre 1

Imagerie Radar RSO : généralités

Ce chapitre est consacré à la présentation de concepts liés à l'imagerie Radar à Synthèse d'Ouverture (RSO). Il s'agit de prendre en compte les caractéristiques de ce type d'images afin de s'orienter vers des méthodes de compression spécifiques. La connaissance des propriétés particulières des images RSO nous permettra également de proposer un ensemble de critères objectifs nécessaires pour évaluer la qualité de reconstruction des images après décompression.

La formation des images RSO est un processus de traitement du signal traitant des données issues de phénomènes physiques liés à la propagation d'ondes électromagnétiques. Nous présenterons ainsi dans un premier temps le principe de la formation d'images RSO avant de décrire certaines propriétés caractéristiques comme le chatoiement et les modèles statistiques associés. Enfin, nous aborderons succinctement les principes de l'interférométrie que nous utiliserons dans le cadre de ce manuscrit comme un outil permettant d'évaluer la qualité de compression des images de nature complexe.

1.1 Principe du radar à synthèse d'ouverture

Le principe du radar (Radio Detection And Ranging) consiste à utiliser des ondes électromagnétiques émises par une antenne dans une direction donnée et à capter en retour l'onde réfléchi par l'obstacle. La vitesse de propagation des ondes électromagnétiques pouvant être raisonnablement considérée comme une constante dans le milieu de propagation, toute mesure de temps de vol se traduit par une mesure en distance. Lorsque l'onde est rétrodiffusée par un objet, la différence entre le temps d'émission et de réception de l'onde fournira alors une information sur la position de cet objet.

Le radar est un système actif qui envoie lui-même ses propres faisceaux d'ondes électromagnétiques et dispose donc de la possibilité de réaliser des acquisitions de jour comme de nuit. Ce système est constitué d'une antenne positionnée sur une plateforme aéroportée ou satellitaire suivant une trajectoire rectiligne. L'antenne à visée latérale émet des trains d'impulsions électromagnétiques suivant un certain angle d'incidence par rapport au sol. Les échos ainsi renvoyés à l'antenne par la surface terrestre permettront la formation de l'image.

Le système imageur a pour but de cartographier avec la meilleure résolution possible la position des rétrodiffuseurs. Le train d'onde réfléchi et enregistré par le capteur à partir de l'instant t_0 et durant une durée ΔT permet de définir la fauchée, c'est à dire la largeur de l'image. La zone au sol imagée est alors définie par une distance relative au capteur comprise entre le *Near Range* $NR = ct_0$ et le *Far Range* $FR = c(t_0 + \Delta T)$ où c est la vitesse de la lumière dans le vide. Entre deux émissions d'impulsions, le capteur se déplace d'une distance $\frac{v}{PRF}$ où PRF (Pulse Repetition Frequency) correspond à la fréquence d'émission des impulsions et v à la vitesse du capteur. Comme l'illustre la figure 1.1, le déplacement du capteur dans la direction azimutale permet une acquisition ligne à ligne de la scène à imager.

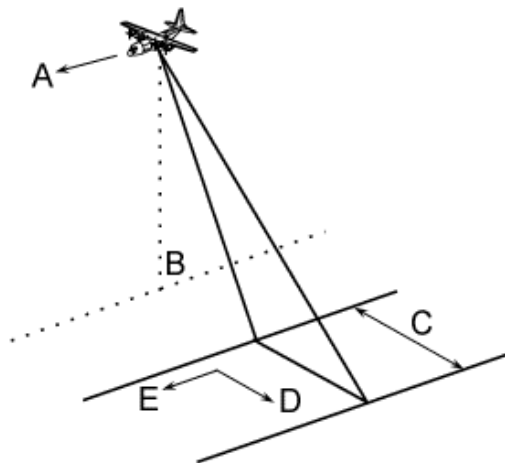


FIG. 1.1 – Principe du radar imageur à visée latérale suivant la direction de vol (A). La projection perpendiculaire au sol forme le nadir (B). La zone imagée est illuminée par le faisceau suivant la direction en distance (D)(perpendiculaire à la trajectoire de vol) et en azimut (E)(parallèle à la trajectoire de vol). La fauchée (C) définit la largeur de l'image.

1.1.1 Longueurs d'onde

Les formes d'ondes propagées sont centrées autour d'une fréquence porteuse comprise entre 300MHz et 100GHz correspondant à des longueurs d'ondes centimétriques. L'atmosphère pouvant être considérée comme pratiquement transparente pour ces longueurs d'onde, les acquisitions sont indépendantes des conditions météorologiques (pluie, nuages) contrairement aux capteurs optiques et proche infra-rouge. Les bandes de fréquences couramment utilisées sont présentées dans la table 1.1. Les grandes longueurs d'onde permettent une meilleure pénétration de la canopée mais les plus faibles longueurs sont plus sensibles à la rugosité. Dans le cadre de cette thèse, nous disposons plus spécifiquement de données acquises en bande X par le capteur RAMSES de l'ONERA.

Bande	Fréquence(GHz)	Longueur d'onde(cm)
P	0.225-0.390	133-76.9
L	0.390-1.550	76.9-19.3
S	1.550-4.20	19.3-7.1
C	4.200-5.75	7.1-5.2
X	5.750-10.90	5.2-2.7
K_u	10.90-22.00	2.7-1.36
K_a	22.00-36.0	1.36-0.83

TAB. 1.1 – Bandes de fréquences et longueurs d'ondes utilisées par les systèmes imageurs RSO.

1.1.2 Résolution en distance avant synthèse de l'image

Le résultat de l'acquisition étant une image, le capteur se caractérise par 2 paramètres physiques très importants que sont les résolutions en distance et en azimut, c'est à dire la capacité du capteur à séparer deux cibles voisines dans ces deux directions.

La résolution en distance dépend uniquement de la durée de l'impulsion de l'onde τ . Pour que deux cibles ponctuelles C_1, C_2 soient distinguées, il faut que l'écart entre la réception des deux échos associés à C_1 et C_2 soit supérieur à la durée de l'impulsion. Cette condition s'écrit sous la forme $t_2 - t_1 \geq \tau$, où t_i est l'instant de réception du signal rétrodiffusé par la cible C_i . La traduction en distance de cette condition conduit à la définition de la résolution en distance dans le cas limite de l'égalité :

$$\delta_d = \frac{c\tau}{2}$$

La résolution obtenue au sol dépend de l'angle d'incidence θ du train d'onde suivant la formule trigonométrique $\delta_{d_{sol}} = \frac{\delta_d}{\sin\theta}$. Soit :

$$\delta_{d_{sol}} = \frac{c\tau}{2\sin\theta}$$

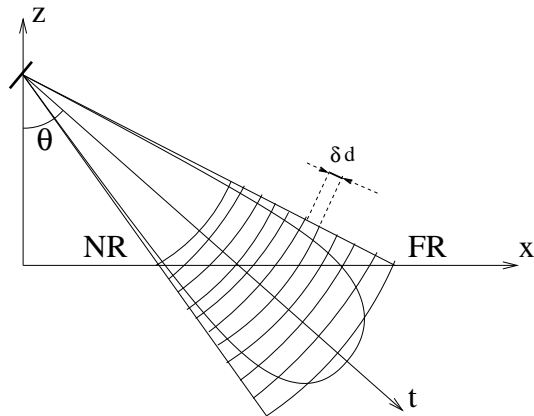


FIG. 1.2 – Emission de l'onde par l'antenne suivant un angle d'incidence θ . Tout point appartenant à la même case "distance" de largeur $\delta_{d_{sol}} = \frac{c\tau}{2\sin\theta}$ sera confondu avec le même échantillon temporel. ©enst.

1.1.3 Résolution en azimuth avant synthèse de l'image

La résolution en azimuth δ_a dépend de la distance radiale R et de l'ouverture du diagramme d'antenne. La résolution est calculée pour un angle d'ouverture du diagramme d'antenne à 3 dB ou plus précisément à 3.92 dB pour avoir un coefficient multiplicateur entier.

$$\delta_a = \theta_{3.92 \text{ dB}} R = \frac{\lambda}{L} R \quad (1.1)$$

où L est la longueur de l'antenne comme illustré sur la figure 1.3.

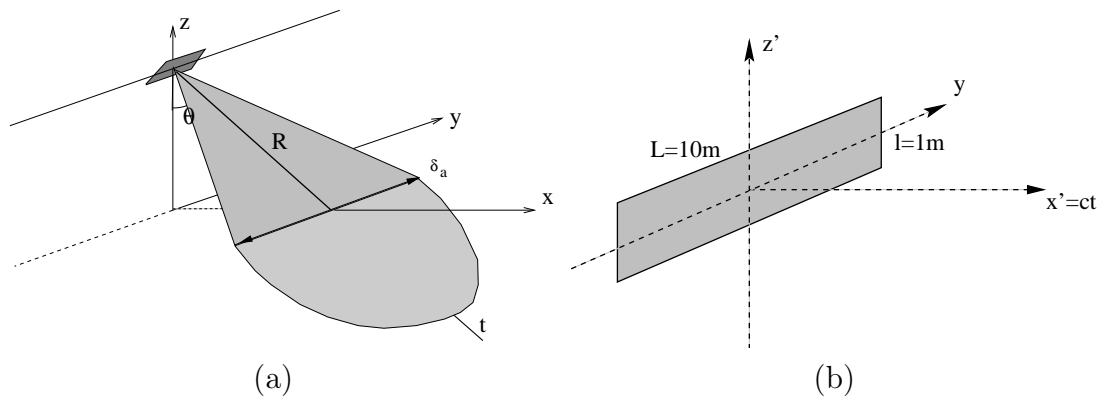


FIG. 1.3 – (a) Résolution azimuthale sans synthèse d'ouverture obtenue à partir du diagramme d'antenne du capteur radar. (b) Antenne plane du capteur AMI d'ERS. ©ENST.

Les résolutions en distance et en azimuth seront ainsi conditionnées par une durée brève d'émission du train d'ondes et une grande antenne.

EXEMPLE D'APPLICATION NUMÉRIQUE :

Dans le cas du satellite Radarsat, on dispose des paramètres suivants :

- $\lambda = 5.66 \text{ cm}$ (Bande C)
- $\theta = 49$ (mode S7)
- $\tau = 20 \mu\text{s}$
- Antenne : $15\text{m} \times 1.5\text{m}$

Dans le cas satellitaire, on choisit $R \simeq 1000\text{km}$. La résolution en distance au sol est alors donnée par $\delta_{d_{\text{sol}}} = 3.9\text{km}$ et en azimut par $\delta_a = 3.77\text{km}$. Ces résolutions sont donc en pratique insuffisantes car limitées techniquement par la durée de l'impulsion et une taille réduite des antennes embarquées. La résolution en distance peut être améliorée par la technique de compression d'impulsion à partir de l'émission d'une forme d'onde particulière appelée "chirp". Quant à la synthèse d'ouverture, elle permet d'améliorer la résolution azimutale en exploitant le déplacement du capteur pour simuler une antenne "virtuelle" de plus grande longueur.

1.2 Synthèse de l'image

1.2.1 Synthèse en distance : compression d'impulsion

Le signal émis par le radar est un "chirp" modulé linéairement en fréquence autour d'une fréquence porteuse f_0 . Son expression est donnée par :

$$f(t) = A_0 e^{j2\pi t(f_0 + \frac{K}{2}t)}, \quad \forall t \in [-\frac{\tau}{2}, \frac{\tau}{2}] \quad (1.2)$$

où K est le taux de modulation défini à partir de la bande passante B_{chirp} du "chirp" et de la durée de l'impulsion τ .

$$K = \frac{B_{\text{chirp}}}{\tau}$$

Après interaction avec le sol, une partie de l'énergie de l'onde est rétrodiffusée vers le capteur avec un coefficient σ et un retard t_c dû au temps de trajet aller-retour de l'onde du radar à la cible.

$$s(t) = \sigma.f(t - t_c)$$

σ est un paramètre dépendant essentiellement de la nature et la réflectivité du terrain (rugosité), du signal émis (fréquence, polarisation, orientation), ainsi qu'aux pertes dues à la propagation du signal et au gain émission-réception de l'antenne.

Pour retrouver la réponse de la cible, le signal reçu est traité par filtrage adapté de réponse impulsionnelle $h(t) = f^*(-t)$:

$$s_{t_c}(t) = \int_{-\infty}^{+\infty} f^*(u - t)s(u)du \quad (1.3)$$

Après intégration, le signal détecté d'une cible ponctuelle située à une distance $c.t_c$ du capteur a pour expression approchée :

$$s_{t_c}(t) \simeq \sigma A_0^2 e^{j2\pi f_0(t-t_c)} \tau \text{sinc}(\pi K \tau (t - t_c)) \quad t \in [t_c - \tau, t_c + \tau] \quad (1.4)$$

Le signal obtenu est de nature **complexe** avec un module mis en forme par un terme en sinus cardinal et un terme de phase dépendant de la fréquence porteuse du chirp et du retard t_c .

La résolution en distance dépend directement du sinus cardinal de l'expression (1.4). L'atténuation du lobe principal à 3.92 dB est obtenue pour un écart temporel $\delta t = t - t_c = \frac{1}{K\tau}$. Traduit en distance, cet écart temporel permet d'obtenir la résolution en distance après compression d'impulsion :

$$\delta_d = \frac{c\delta t}{2} = \frac{c}{2} \frac{1}{K\tau} = \frac{c}{2} \frac{1}{B_{chirp}}$$

La résolution radiale dépend donc directement de la bande passante du chirp et non plus uniquement de la durée de l'impulsion τ . Elle sera d'autant meilleure que la bande passante du chirp sera élevée.

APPLICATION NUMÉRIQUE :

Le système Radarsat dispose d'une émission de 3 types de chirp de bandes passantes $B_{chirp} = \{11.6, 17.3, 30\}MHz$. A l'aide de l'expression précédente et l'équation (1.1), les résolutions en distance possibles sont alors respectivement $\delta_{d_{sol}} = \{17, 11.5, 6.6\}m$ pour un angle d'incidence $\theta = 49^\circ$ (mode S7).

1.2.2 Synthèse en azimuth

Le radar émet un train d'onde tout en suivant une trajectoire rectiligne. En tenant compte du diagramme du lobe d'antenne, un même point sera donc illuminé plusieurs fois. On obtient ainsi une série de données pour chaque point de la scène. En combinant la variation d'amplitude et de phase de ces retours, le traitement de synthèse d'ouverture permet d'obtenir des images des zones observées comme si on utilisait une antenne large à très grande résolution.

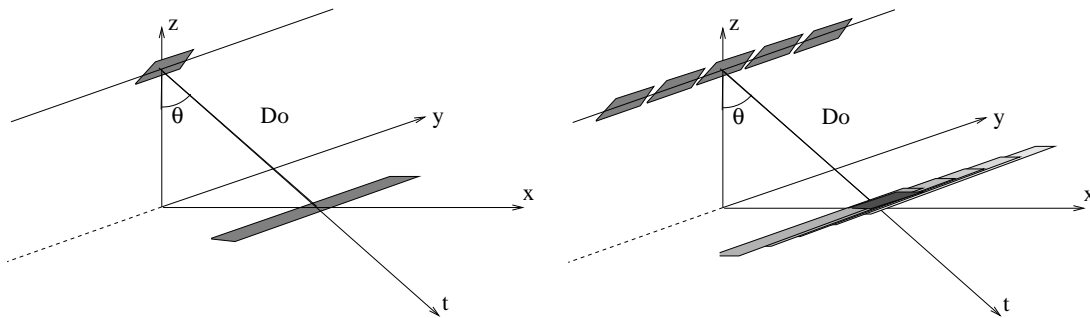


FIG. 1.4 – A gauche : empreinte au sol traduisant la résolution en distance et en azimuth du capteur avant synthèse azimuthale. A droite : en combinant les échos reçus à partir de plusieurs acquisitions, la synthèse d'ouverture permet d'obtenir une résolution azimuthale proche de la dimension de l'antenne réelle. ©ENST.

En considérant le déplacement du porteur dans l'axe azimutal Oy du plan (Oxy) et sous la condition d'une PRF convenablement choisie, un même point sera illuminé plusieurs fois car le lobe de l'antenne est large. La distance du porteur à la cible s'exprime sous la forme $D_c = \frac{ct_c}{2}$. On peut désormais considérer que cette distance $D_c = D_c(y)$ varie au cours du déplacement du porteur.

On introduit le paramètre D_0 correspondant à la distance la plus courte du capteur à la cible, c'est à dire celle obtenue lorsque le radar vise perpendiculairement à sa trajectoire (voir figure 1.4). La distance D_c s'écrit alors :

$$D_c(y) = \sqrt{D_0^2 + y^2} \simeq D_0 + \frac{y^2}{2D_0} \quad (1.5)$$

En reprenant l'expression (1.4) pour une cible située à la distance $D_c = \frac{ct_c}{2}$, on peut réécrire :

$$s_{D_c}(t) \sim e^{j2\pi f_0(t - \frac{2D_c}{c})} \tau \text{sinc}(\pi K \tau (t - \frac{2D_c}{c})) \quad (1.6)$$

Le terme $\tau \text{sinc}(\pi K \tau (t - \frac{2D_c}{c}))$ atteint son maximum pour $t = t_c = \frac{2D_c}{c}$ et ne varie pratiquement pas. En revanche, le terme $e^{j2\pi f_0(t - \frac{2D_c}{c})}$ induit une variation de la phase au cours du temps et qui dépendra de la distance D_c du capteur à sa cible.

En remplaçant $D_c = D_c(y)$ dans l'équation (1.6) et en négligeant l'effet du sinus cardinal, le terme prépondérant relatif au signal reçu est de la forme :

$$s_{D_c}(t) \sim e^{j2\pi f_0(t - \frac{2D_c}{c})} = e^{j2\pi f_0(t - \frac{2D_0}{c})} e^{j2\pi \frac{f_0 y^2}{c D_0}} \quad (1.7)$$

En considérant la vitesse de déplacement azimutale v du porteur, la position y s'écrit $y = vt$. La phase du second terme $e^{j2\pi \frac{f_0 y^2}{c D_0}}$ s'exprime alors comme une fonction quadratique du temps :

$$\phi = 2\pi \frac{f_0 v^2}{c D_0} t^2 \quad (1.8)$$

La fréquence instantanée $f = \frac{1}{2\pi} \frac{d\phi}{dt} = \frac{2v^2}{\lambda D_0} t$ est donc une fonction linéaire du temps. Le déplacement du radar crée de manière naturelle un "chirp" qui peut être traité par filtrage adapté comme dans le cas de la compression d'impulsion en distance.

Par analogie avec le chirp temporel donné par la formule (1.2), on définit la constante du chirp en azimut $K_a = \frac{2}{\lambda D_0}$. A partir de l'expression de la résolution en azimut (formule 1.1), on introduit alors la valeur $Y = \frac{\lambda}{L} D_0$ comme étant la longueur de l'antenne adaptée à une largeur du lobe d'antenne à 3.92 dB d'atténuation. En reportant cette valeur dans le sinus cardinal issu de l'antenne synthétique obtenue par filtrage adapté en azimut, on en déduit la résolution azimutale :

$$\delta_a = \frac{1}{K_a Y} = \frac{L}{2}$$

Etrangement, plus l'antenne est petite, meilleure sera la résolution en azimut. En effet, une antenne plus petite dispose d'un lobe plus large. Tant qu'elle se trouvera dans le faisceau du radar, une cible ponctuelle fournira donc une série d'échos. L'antenne ainsi synthétisée par filtrage adapté est plus longue et par conséquent de meilleure résolution.

APPLICATION NUMÉRIQUE :

Pour le système Radarsat, l'application numérique directe du résultat obtenu précédemment nous permet de déduire la résolution azimutale après synthèse d'ouverture. En prenant $L = 15m$, on obtient $\delta_a = 7.5m$.

1.2.3 Migration et dépointage

Migration

La migration illustre le fait qu'avant synthèse d'ouverture, une cible se trouve sur une courbe parabolique et non sur une ligne droite. Comme le montre la figure 1.5, ces décalages en distance créent parfois des changements de case distance au niveau des pixels et doivent être corrigés avant synthèse.

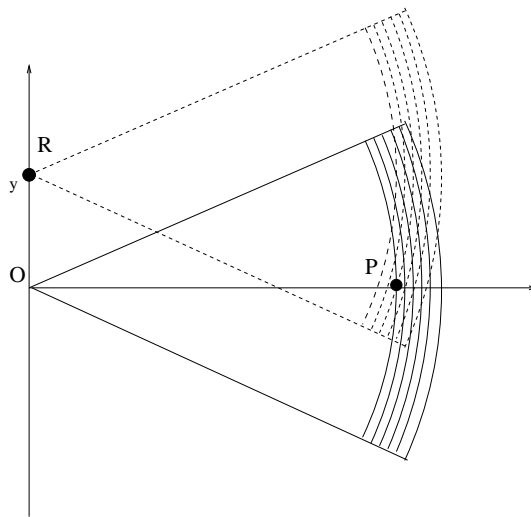


FIG. 1.5 – Illustration de la migration. En fonction de la position du capteur, le point P n'appartient pas à la même case distance par rapport au référentiel du CPA (point 0 tel que la distance OP soit minimale).©ENST.

Le problème de migration est délicat car il interdit l'application directe d'un filtrage adapté dans la direction azimutale. Plusieurs travaux ont ainsi consisté à transposer le filtrage adapté dans le domaine de Fourier de manière à pouvoir ramener le filtrage azimutal à un simple produit. Citons notamment, le changement de variable de Stolt, ou l'algorithme $\Omega - K$.

Dépointage et Doppler Centroid

En reprenant le calcul correspondant à la synthèse d'ouverture, nous avons supposé que l'antenne visait perpendiculairement à la trajectoire du porteur. En pratique, cette hypothèse n'est pas toujours vérifiée, en particulier dans le cas aéroporté à cause des variations de trajectoire de l'avion. Ce dépointage caractérisé par un angle ψ traduit le fait que le lobe d'antenne pointe sur une cible d'abscisse $y_c = D_0 \sin \psi$ comme l'illustre la figure 1.6.

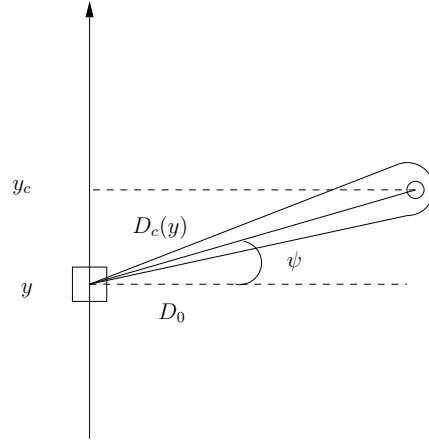


FIG. 1.6 – Dépointage de l'antenne suivant un angle ψ . Le changement de distance entre le capteur et la cible implique que la fréquence instantanée du signal reçu n'est plus à Doppler nul.

En tenant compte du dépointage de l'antenne, l'équation (1.5) est remplacée par :

$$D_c(y) = \sqrt{D_0^2 + (y - y_c)^2} \simeq D_0 + \frac{(y - y_c)^2}{2D_0} \quad (1.9)$$

L'expression (1.7) du signal reçu se réécrit donc :

$$s_{D_c}(t) \sim e^{j2\pi f_0(t - \frac{2D_c}{c})} = e^{j2\pi f_0(t - \frac{2D_0}{c})} e^{j2\pi \frac{f_0(y - y_c)^2}{cD_0}} \quad (1.10)$$

Le point central de référence dans le repère (Oxy) n'est donc plus à Doppler nul (c'est à dire que la fréquence instantanée du chirp en azimuth n'est pas nulle pour la position $y = 0$). Cette fréquence appelée "Doppler centroïd" est donnée par :

$$f_{DC} = -\frac{2v y_c}{\lambda D_0} = -\frac{2v}{\lambda \sin \psi} \quad (1.11)$$

Si la valeur exacte du dépointage n'est pas connue, la reconstruction s'effectuera sans ambiguïté si $f_{DC} \leq \frac{PRF}{2}$, sous peine de créer des cibles fantômes. D'autre part, le "Doppler centroïd" implique une translation du spectre dans le sens azimuthal comme le montre la figure 1.8. En ce qui concerne l'application à la compression, nous verrons au chapitre 3 qu'il est indispensable de recentrer le spectre si possible autour de la fréquence nulle de façon à s'adapter aux transformées multi-résolutions de type ondelettes.

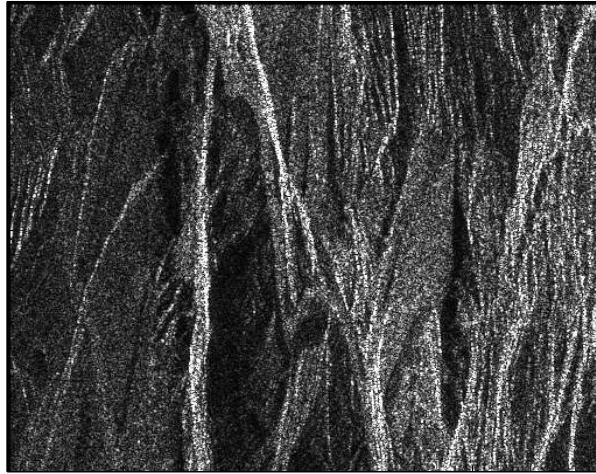


FIG. 1.7 – Amplitude d'une image RSO issue du capteur ERS après synthèse d'ouverture ©ESA.

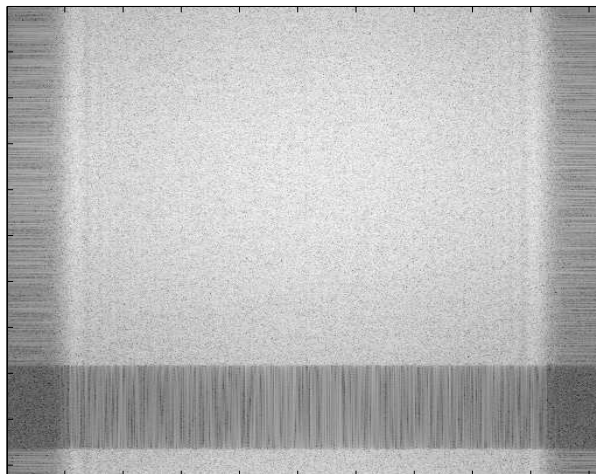


FIG. 1.8 – Module (en dB) de la transformée de Fourier de l'image complexe : le spectre est décentré dû à un dépointage de l'antenne par rapport à la trajectoire du porteur (la fréquence nulle est au centre de l'image).

1.3 Les données issues du radar à synthèse d'ouverture

1.3.1 Données brutes

Ce signal de nature complexe et bi-dimensionnelle est issu directement de l'échantillonnage des échos reçus en distance et en azimut par le capteur après démodulation. Afin de respecter le théorème d'échantillonnage de Shannon, il convient de choisir en distance une fréquence d'échantillonnage supérieure à la bande passante du chirp d'impulsion $f_e \geq \frac{1}{K\tau}$. A titre d'exemple, le système satellitaire ERS-1 dispose d'une fréquence d'échantillonnage $f_e = 18.96 MHz$ pour un chirp de bande passante $B_{chirp} = 15.55 MHz$. En azimut, cette condition s'applique sur la PRF : $PRF \geq B_{doppler}$ avec $B_{doppler} = 2\frac{v}{L}$ définie comme étant la bande passante du chirp créé par le déplacement du porteur. Soit pour ERS-1 une PRF variant de 1640 à 1720 Hz.

Les données sont ensuite quantifiées pour être transmises à une station de réception au sol en vue du traitement de synthèse. Par exemple, dans le cas d'ERS-1, 10 bits par échantillon complexe sont alloués (5 bits pour la partie réelle et 5 pour la partie imaginaire), soit un débit de 100 Mbit/s. De nombreuses méthodes de compression ont été appliquées sur ce type de données. Citons notamment les travaux de D.Lebedeff [50] sur la quantification vectorielle algébrique des données brutes. Cependant dans le cadre de cette thèse, nous nous intéresserons plus spécifiquement aux données complexes après synthèse RSO.

1.3.2 Les images SLC (Single Look Complex)

Il s'agit de l'image complexe obtenue après le traitement de synthèse d'ouverture des données brutes. L'image obtenue est orientée suivant la direction du porteur, c'est-à-dire parallèle à la trace de ce dernier. Le caractère complexe des données permet d'envisager des techniques d'interférométrie utilisant l'information portée par la phase. L'image d'amplitude est notamment utilisée dans un contexte de détection-reconnaissance-identification (DRI) bien que la phase soit également une information utile sur la signature des points ou zones d'intérêt.

La dimension d'un pixel dépend en azimut de la PRF et de la vitesse du porteur. En distance, la dimension du pixel sur un sol plat dépend de l'angle d'incidence θ de l'onde et de la fréquence d'échantillonnage en distance. On obtient ainsi :

$$\begin{aligned}\Delta_a &= v/PRF \\ \Delta_{d_{sol\ plat}} &= \frac{c}{2F_e \sin \theta}\end{aligned}$$

A titre d'exemple, les données ERS sont codées par valeur entière sur 16 bits par partie réelle et imaginaire, soit 32 bits par pixel complexe. Le pixel est de dimension rectangulaire, fortement allongé en distance (5x25m).

Les techniques et les progrès technologiques en terme de puissance de calcul permettent désormais d'envisager une synthèse à bord même du porteur. C'est donc sur ce type de données que s'inscrivent les travaux de cette thèse. Nous travaillerons plus particulièrement sur la compression d'images SLC en bande X issues du capteur RAMSES de

l'ONERA. La figure 1.9 montre une image SLC issue de ce capteur suivant sa représentation amplitude/phase.

Les pixels de ce type d'images sont codés à 64 bits par pixel complexe (32 pour la partie réelle et 32 pour la partie imaginaire) en valeur de type "float". Par exemple, une image de taille 3600x3000 représente un volume de données de 82.4Mo.

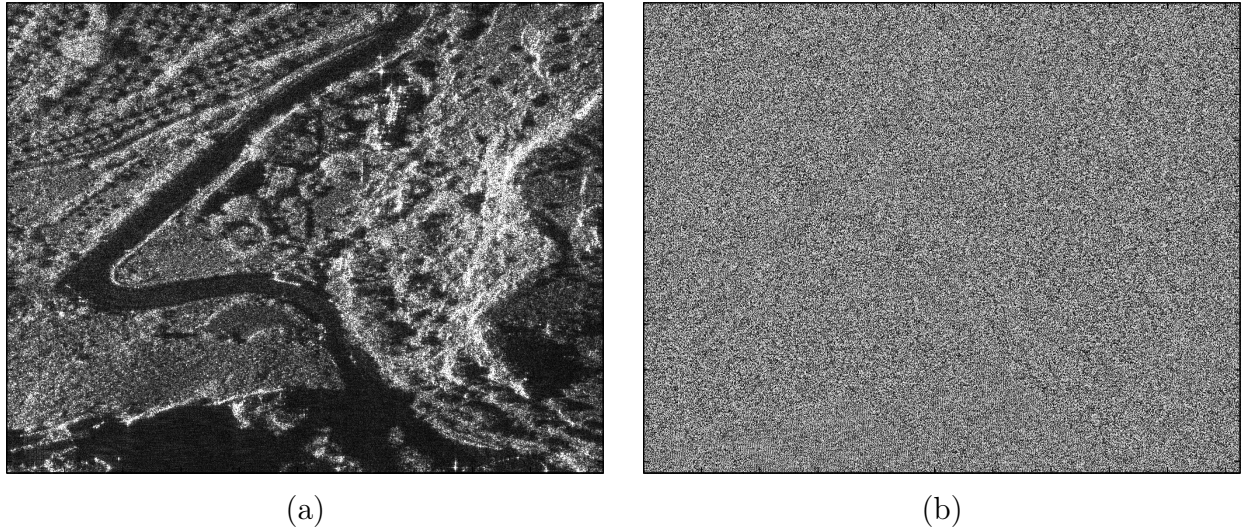


FIG. 1.9 – (a) : Image d'amplitude du capteur RAMSES de l'ONERA (zone extraite des Baux de Provence 1024x1024). (b) : Image de Phase. ©ONERA.

1.3.3 Les données multivues

Les images SLC sont affectées par le phénomène de chatoiement (présenté à la section 1.4) qui altère la lisibilité de l'image et son interprétation. Les images multivues (*multilook* en anglais) ont subi un traitement de façon à réduire l'effet du chatoiement et à ramener le pixel carré à la géométrie du sol. Deux étapes majeures sont nécessaires :

- un filtrage azimutal permettant une réduction du chatoiement et permettant un sous-échantillonnage.
- une étape de suréchantillonnage en distance, prenant en compte la variation le long de la fauchée de l'angle d'incidence.

Les images multivues sont en général construites à partir des images d'intensité (amplitude au carré) : l'image en intensité "L-vues" est obtenue en moyennant L images en intensité 1-vue décorréelées. Ces L images s'obtiennent en découpant l'ouverture synthétique en L sous-ouvertures. Le moyennage effectué entre plusieurs sous-images lors du premier traitement réduit la résolution de l'image, ce qui permet également de réduire le volume de données nécessaire à l'encodage de l'image. On perd cependant le caractère complexe de l'image puisque l'image obtenue est sous forme d'amplitude PRI (*Precision Image*).

Dans le cadre de cette thèse, nous souhaitons cependant conserver l'information de phase et donc le caractère complexe de l'image. En conséquence, nous ne nous intéresserons pas à la compression d'images PRI multivues.

1.4 Chatoiement et modélisation statistique

1.4.1 Chatoiement pleinement développé

Les images RSO représentent la réponse du terrain à l'onde hyperfréquence émise par le radar. La radiométrie du pixel est donc une fonction de l'interaction onde-matière et dépend de deux ensembles de paramètres :

- les paramètres propres du radar (longueur d'onde, polarisation, loi d'antenne..)
- les paramètres liés à la nature du sol (réflectance, humidité, rugosité par rapport à la longueur d'onde, inclinaison du sol..) et à la nature des cibles.

Ainsi, même si les caractéristiques physiques de rétrodiffusion sont stationnaires (région dite "homogène"), les valeurs des pixels sont extrêmement dispersées, ce qui donne aux images un aspect granuleux. Ce phénomène est appelé "chatoiement" (ou *speckle* en anglais) dont les caractéristiques particulières le font qualifier de bruit multiplicatif. Le chatoiement des images monovues (SLC) rend très difficile la lisibilité de l'image, la détection d'objets et également la compression, en raison de la très faible corrélation des pixels.

Le modèle le plus couramment utilisé pour caractériser le chatoiement provient du formalisme de Goodman [39]. On considère que le sol est constitué de petits diffuseurs ponctuels dont le nombre est important et la localisation aléatoire à l'intérieur de la cellule de résolution (zone du sol dont la rétrodiffusion contribue au signal mesuré en un pixel). La réponse d'une cellule de résolution est la somme de N réponses de cibles ponctuelles $s_i = \rho_i e^{j\phi_i}$:

$$S = \sum_{i=1}^N \rho_i e^{j\phi_i} = a e^{j\phi} = i + jq \quad (1.12)$$

où i et q représentent respectivement la partie réelle (i pour *in phase*) et imaginaire (q pour *in quadrature*) du signal complexe démodulé (fréquence porteuse à $f_0 = 5.3GHz$ pour ERS par exemple).

Les hypothèses de Goodman caractérisant le chatoiement dit **pleinement développé** sont les suivantes :

- ρ_i et ϕ_i sont des variables aléatoires indépendantes entre elles et indépendantes des autres diffuseurs.
- La surface est considérée comme rugueuse par rapport à la longueur d'onde λ et la taille de la cellule de résolution grande devant λ . Il en résulte que les phases ϕ_i sont indépendantes entre elles et suivent une loi uniforme sur $[0, 2\pi]$.
- N est grand.
- Les variables aléatoires ρ_i sont identiquement distribuées avec des moments d'ordre 1 et 2 respectivement égaux à $E(\rho)$ et $E(\rho^2) = \sigma^2$.

En utilisant le théorème central limite, on peut en déduire que les variables aléatoires $S : (i, q)$ sont distribuées selon une loi gaussienne. En outre, i et q sont décorréliées ($E(iq) = 0$) et donc indépendantes. Le chatoiement est en conséquence un processus gaussien circulaire. Il est alors aisé de montrer que la densité de probabilité de l'image d'intensité $I = i^2 + q^2$ suit une loi exponentielle.

$$p(I) = \frac{1}{\mu_I} e^{-\frac{I}{\mu_I}} \quad (1.13)$$

avec $\mu_I = R$, $R = 2\sigma^2$ étant le coefficient de rétrodiffusion.

On en déduit la loi d'amplitude suivant la distribution de Rayleigh (ou loi de Rayleigh-Nagakami de paramètre $L = 1$ d'après J.M. Nicolas [69]).

$$p(A) = \frac{2}{\mu_A} \left(\frac{A}{\mu_A} \right) e^{-\left(\frac{A}{\mu_A}\right)^2} \quad (1.14)$$

avec $\mu_A^2 = \mu_I$.

Le chatoisement perturbant notamment la lisibilité des images, il est possible de construire des images "multivues" dont la résolution plus faible est compensée par un lissage du chatoisement. Le moyennage multivue d'une image en intensité vérifiant une loi exponentielle implique une distribution en intensité de l'image "L-vues" caractérisée par une loi Gamma généralisée (suivant les notations de Nicolas [69]) et dont la densité de probabilité (d.d.p) est donnée par :

$$p_L(I) = \frac{1}{\Gamma(L)} \frac{L}{\mu_I} \left(\frac{LI}{\mu_I} \right)^{L-1} e^{-\frac{LI}{\mu_I}} \quad (1.15)$$

où Γ est la fonction Gamma. Le paramètre L , qui est ici par construction le nombre de vues, peut être vu comme un facteur de forme.

La distribution d'amplitude $A = \sqrt{I}$ est donnée par la loi de Rayleigh-Nagakami :

$$p_L(A) = \frac{2}{\Gamma(L)} \frac{\sqrt{L}}{\mu_A} \left(\frac{\sqrt{L}A}{\mu_A} \right)^{2L-1} e^{-\left(\frac{\sqrt{L}A}{\mu_A}\right)^2} \quad (1.16)$$

Pour $L = 1$, on retrouve en particulier l'expression de la d.d.p d'une loi de Rayleigh. La figure 1.10 illustre ces familles de distributions.

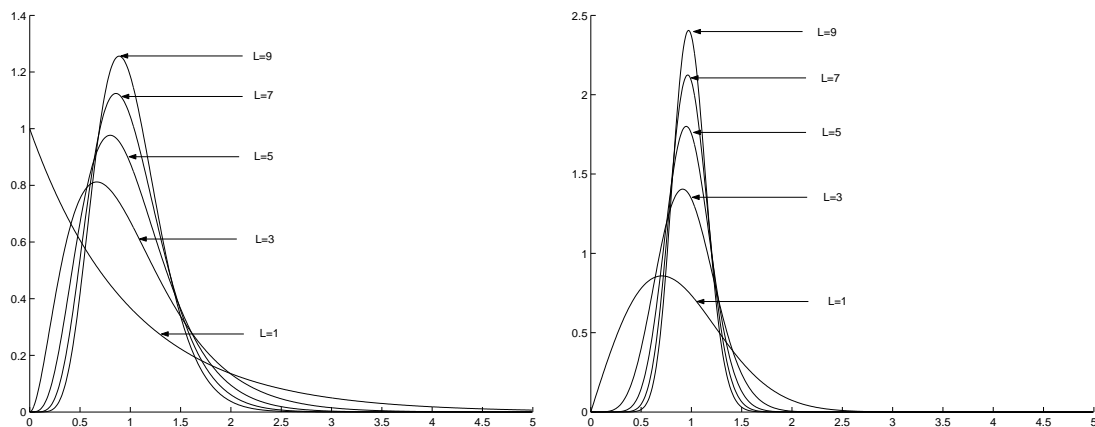


FIG. 1.10 – A gauche : Distribution en intensité du chatoisement pour une image "L-vues" suivant la loi Gamma de paramètre $\mu_I = 1$ et $L = 1, 3, 5, 7, 9$. A droite : Distribution en amplitude du chatoisement pour une image "L-vues" suivant la loi de Rayleigh-Nagakami de paramètre $\mu_A = 1$ et $L = 1, 3, 5, 7, 9$.

D'après le modèle gaussien circulaire considéré ci-dessus, la phase ϕ a donc une distribution uniforme sur $[0 : 2\pi]$:

$$p(\phi) = \frac{1}{2\pi}, \phi \in [0, 2\pi] \quad (1.17)$$

Le modèle de Goodman convient bien aux milieux naturels pour les images de faible résolution. A haute résolution, C.Tison [99] a montré que les propriétés du signal changent avec notamment des textures qui tendent à disparaître, des échos forts qui dominent. Le formalisme de Goodman n'est ainsi pas forcément le meilleur modèle qui soit adapté à la caractérisation statistique du chatoiement.

1.4.2 Modèles non-gaussiens

De nombreux modèles ont été proposés, basés sur des formulations des phénomènes physiques comme par exemple :

- la loi K [27] [71] souvent adaptée pour représenter les surfaces hétérogènes mais peu adaptée aux surfaces très hétérogènes.
- la loi Beta [1] ou autres distributions U, B, W issues du système de Pearson [41], bien adaptées par exemple pour les réponses de surface océanique.

D'autres modèles se fondent plus sur des approches mathématiques, issues d'un choix empirique :

- les lois Log-normale [115] adaptées aux surfaces très hétérogènes ou urbaines.
- la loi de Weibull [72] pouvant être vue comme de Rayleigh-Nagakami "généralisée".

A partir de ses travaux, C. Tison [99] précise que ces modèles ne sont pas toujours adaptés car ne modélisant bien souvent qu'une partie de la scène. Les pixels d'image RSO à haute résolution en milieu urbain présentent de fortes probabilités d'avoir une forte intensité due à la présence de forts réflecteurs. Les histogrammes présentent alors une "queue lourde" à faible décroissance¹ mal représentée par les modèles mentionnés ci-dessus.

Le modèle statistique issu des phénomènes de rétrodiffusion complexes utilisés dans le cadre de cette thèse correspond à un compromis de tous ces modèles, suivant un critère mathématique et empirique, indépendamment de toute considération physique. Nous avons ainsi choisi la distribution de Fisher à 3 paramètres (autrement appelée loi Beta de seconde espèce ou loi Pearson de type VI) et dont l'expression est donnée par :

$$p_I(x) = \frac{\Gamma(L+M)}{\Gamma(L)\Gamma(M)} \frac{L}{M\mu} \frac{\left(\frac{Lx}{M\mu}\right)^{L-1}}{\left(1 + \frac{Lx}{M\mu}\right)^{L+M}} \quad (1.18)$$

Cette densité de probabilité est facilement exploitable par le formalisme des statistiques de seconde espèce dont nous donnerons une courte présentation au chapitre 3. La loi de Fisher à 3 paramètres (une grandeur proche de la valeur moyenne μ et deux autres paramètres de forme L et M) dispose de propriétés de modélisation fort intéressantes et permet d'approcher toutes les lois mentionnées précédemment et de caractériser un grand ensemble de textures. On montre ainsi, que pour M tendant vers $+\infty$, la loi Fisher tend vers la loi Gamma dite "à tête lourde". A l'opposé, pour L tendant vers l'infini, on montre

¹Ces lois ne possèdent pas de moments à partir d'un certain ordre.

que la loi Fisher tend vers la loi Gamma Inverse "à queue lourde".

Ce modèle de distribution de Fisher a été notamment validé sur des données réelles au cours des travaux de C. Tison [99]. Grâce à sa capacité à modéliser les distributions à tête lourde ou queue lourde, la distribution de Fisher permet de caractériser aussi bien les surfaces homogènes, hétérogènes (milieu urbain par exemple), naturelles ou non naturelles. Nous ferons ultérieurement appel à cette loi afin de construire une stratégie de compression adaptée aux statistiques du signal. Quelques exemples de distribution de Fisher sont illustrés par la figure (1.11).

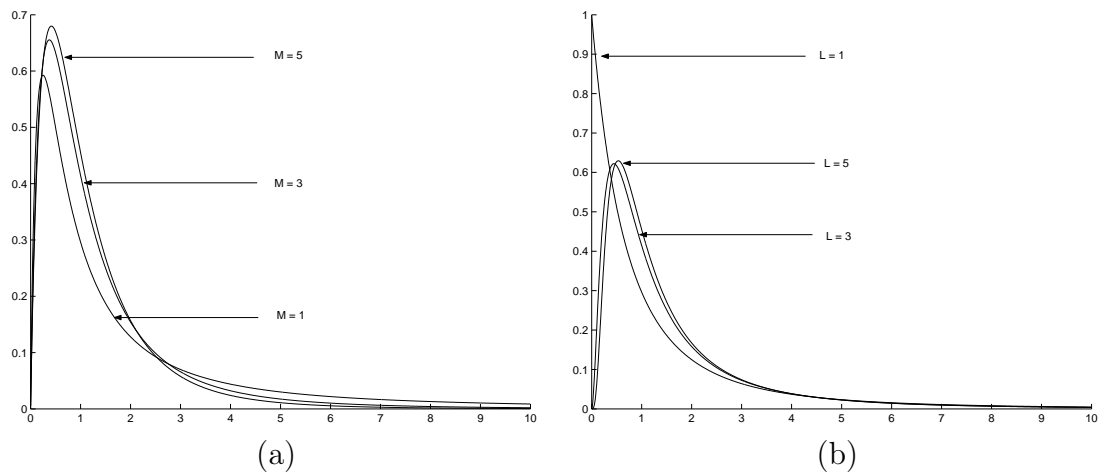


FIG. 1.11 – (a) : Distribution de Fisher pour $\mu = 1, L = 2$ fixé et $M=1,3,5$. (b) : Distribution de Fisher pour $\mu = 1, M = 2$ fixé et $L=1,3,5$.

1.5 Interférométrie

Dans le cadre de cette thèse, la compression avec pertes d'images RSO SLC nous contraint à encoder conjointement l'information d'amplitude et de phase portée par les images. L'interférométrie radar est une technique relativement récente largement utilisée par la communauté radar et qui utilise en particulier l'information de phase comme un moyen de mesure sensible aux variations de distance. Cette technique propose ainsi de coupler deux images de la même zone avec des angles d'incidence légèrement différents. L'analyse fine de la différence de phase entre deux images RSO SLC permettra alors de restituer le relief du terrain ou de mesurer les mouvements du sol. Les images peuvent être acquises simultanément en mono-statique (cas d'une plateforme contenant deux capteurs avec un seul émetteur), ou bi-statique (plateforme survolant la scène à deux dates distinctes ou d'une plateforme avec deux émetteurs).

Nous nous limiterons à exposer brièvement les principes de l'interférométrie radar car les traitements associés à cette technique dépassent le cadre de cette thèse. Pour plus de précisions concernant la théorie approfondie et l'application de l'interférométrie radar, nous renvoyons le lecteur à [99], [100], [85].

Dans un contexte de compression avec pertes, l'information de phase de chaque image RSO sera irréversiblement dégradée. En considérant l'image de différence de phase et en

comparant l'interférogramme issu du couple d'images originales avec l'interférogramme issu du couple d'images après compression-décompression, il est alors possible d'obtenir une mesure de la qualité de reconstruction. Une étude plus détaillée sur l'introduction de cette mesure d'"écart interférométrique" sera faite au chapitre 7.

1.5.1 Information de Phase

Un pixel d'une image de phase est constitué de deux informations :

$$\phi = \phi_{propre} + \phi_D \quad (1.19)$$

- une phase propre ϕ_{propre} issue de l'interaction entre l'onde et le sol. Ce terme déterministe dépend de la nature physique du sol et de l'incidence de prise de vue.
- une phase géométrique ϕ_D qui dépend du trajet aller-retour de l'onde : $\phi_D = \frac{4\pi}{\lambda} |\vec{D}|$ où $|\vec{D}|$ représente la distance du capteur au centre du pixel.

De manière à pouvoir exploiter le terme de phase géométrique, il est nécessaire d'éliminer le terme de phase propre. Pour cela, il suffit de calculer la différence de phase entre deux images acquises dans des conditions quasi-identiques. En faisant l'hypothèse que la nature du terrain n'a pas changé entre deux acquisitions, le terme de phase propre s'élimine. En choisissant un angle d'incidence légèrement différent pour le second capteur de manière à obtenir une différence de phase géométrique non nulle, la phase interférométrique a donc pour expression :

$$\begin{aligned} \Delta\phi &= \phi_{pixel_1} - \phi_{pixel_2} = (\phi_{propre} + \phi_{D_1}) - (\phi_{propre} + \phi_{D_2}) \\ &= \frac{2k\pi}{\lambda} (|\vec{D}_1| - |\vec{D}_2|) \end{aligned} \quad (1.20)$$

où k est égal à 1 dans le cas mono-statique et à 2 dans le cas bi-statique.

La phase interférométrique n'est cependant connue qu'à un multiple de λ près. Ceci implique des "sauts" de phase dans une représentation modulo 2π qui laissera apparaître des franges. L'image ainsi formée constitue l'interférogramme. Il s'agit ensuite de remonter à la différence de marche à partir de l'interférogramme en déroulant les franges. Nous ne traiterons pas du déroulement des franges par la suite et nous nous limiterons dans le cadre de cette étude à la formation du produit interférométrique.

1.5.2 Construction de l'interférogramme

La formation d'un interférogramme nécessite plusieurs étapes :

1. le recalage local : il s'agit de superposer un pixel de l'image 1 avec celui de l'image 2 de façon à ce que les réflecteurs élémentaires, ayant contribué à la formation de l'image 1, soient identiques à ceux de l'image 2. Par différence de phase, on s'assure alors de l'élimination du terme de phase propre.
2. le recalage subpixelaire en tous points de la scène : une transformation globale doit être trouvée en étendant par exemple les recalages locaux. Diverses méthodes [100] basées sur des techniques de corrélation, utilisation de fréquences locales, calcul des valeurs de paramètres d'une fonction de coût globale, sont possibles.
3. la construction de l'interférogramme et de l'image de cohérence.

L'interférogramme est constitué de l'ensemble des pixels $\Delta\phi(x, y)$ en tout point de l'image. En pratique, on calcule le produit hermitien des 2 images complexes z_1 et z_2 :

$$d(x, y)e^{j\Delta\phi(x, y)} = \frac{z_1(x, y)z_2^*(x, y)}{\sqrt{|z_1(x, y)|^2 |z_2(x, y)|^2}} \quad (1.21)$$

Les images RSO sont affectées par le phénomène de chatoiement qui perturbe l'estimation de la différence de phase $\Delta\phi(x, y)$ et de la cohérence $d(x, y)$. De manière à limiter l'effet du chatoiement, le produit hermitien est moyenné sur une fenêtre de L pixels voisins suivant la formule :

$$de^{j\Delta\phi} = \frac{\sum_{i=1}^L z_{1i} z_{2i}^*}{\sqrt{\sum_{i=1}^L |z_{1i}|^2 \sum_{i=1}^L |z_{2i}|^2}}, \quad d \in [0, 1] \quad (1.22)$$

d est alors appelée cohérence empirique et $\Delta\phi$ la différence de phase empirique. La cohérence reflète la capacité des parcelles de la scène imagée à fournir des phases comparables. Une cohérence faible peut être synonyme d'un changement d'angle d'incidence trop important, de changement dans la nature du sol, ou de zones de très faible rétrodiffusion (ombres par exemple) avec une domination du bruit thermique.

Les distributions de la cohérence empirique et différence de phase empirique s'expriment en fonction du paramètre L et de la cohérence réelle D [85] :

$$\begin{aligned} p(d|D, L) &= 2(L-1)(1-D^2)^L d(1-d^2)^{L-2} {}_2F_1(L, L; 1; d^2 D^2) \\ p(\Delta\phi|D, L) &= \frac{(1-D^2)^L}{2\pi} \frac{1}{2L+1} {}_2F_1\left(2, 2L; L + \frac{3}{2}; \frac{1+D \cos(\Delta\phi)}{2}\right) \end{aligned}$$

La figure 1.12 montre les distributions de la cohérence et de la phase d'un interférogramme complexe monovue ($L = 1$) pour diverses valeurs de la cohérence vraie D . On remarque ainsi que plus la cohérence D est élevée, plus l'écart-type de la densité de probabilité de la phase empirique est faible, ce qui est en accord avec l'idée que la cohérence demeure un critère de qualité et de stabilité sur la mesure de la phase interférométrique.

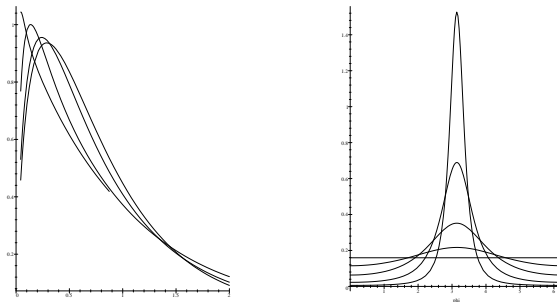


FIG. 1.12 – Distribution de la cohérence de l'interférogramme complexe monovue (à gauche) et de la différence de phase (à droite) pour $D = 0, 0.2, 0.5, 0.8, 0.95$. On remarque que pour $D = 0$, c'est à dire en l'absence de cohérence, la phase est équidistribuée sur $[0, 2\pi]$. ©ENST.

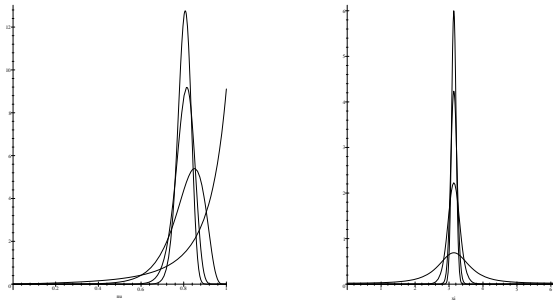


FIG. 1.13 – Distribution de la cohérence de l’interférogramme complexe multi-vues (à gauche) et de la différence de phase (à droite) pour $D = 0.8$ et pour $L = 1, 9, 32, 64$. ©ENST.

La figure 1.13 illustre les densités de probabilité de la phase empirique pour plusieurs valeurs de L et en fonction de la cohérence réelle. Plus le paramètre L de moyennage est élevé plus la densité de probabilité de la cohérence empirique affiche un écart-type faible et centré sur la valeur réelle de la cohérence.

D’après sa construction, l’interférogramme dispose d’informations à l’origine d’application de mesure d’élévation, de mesure de mouvement de terrains :

1. information sur l’élévation : le relief crée des déformations entre les deux images si les orbites ne sont pas identiques. Ces déformations impliquent un déphasage entre les deux images proportionnel à l’élévation du terrain permettant ainsi d’estimer une possible différence d’altitude en un point considéré.
2. information sur le mouvement de terrain : l’utilisation de plusieurs capteurs permet la mise en oeuvre de l’interférométrie différentielle. Entre deux acquisitions non simultanées, un glissement de terrain ou un tremblement de terre a pu déplacer des points de l’image. La variation de la distance par rapport au radar entraîne donc une différence de phase proportionnelle au déplacement radial suivant l’équation (1.20). L’interférogramme traduit donc avec une précision centimétrique, les mouvements de terrain. Les franges observées sont alors les courbes d’*iso-déplacement*.
3. information de cohérence : un changement dans la nature physique ou géométrique du sol (surface liquide, croissance de végétation, variation d’incidence) modifie la phase propre des pixels (hypothèse interférométrique non vérifiée). Dans ce cas, la phase du pixel varie d’une acquisition à l’autre donnant à l’interférogramme un aspect localement bruité. La cohérence étant le reflet de la stabilité de l’information de phase, il est possible de détecter les changements survenus sur le sol.

En conclusion, bien que la compression d’image RSO SLC avec pertes impacte de manière indirecte les méthodes d’exploitation des données interférométriques, nous nous limiterons ici à la construction et à l’évaluation de qualité d’interférogrammes. L’interférogramme est considéré ici comme un outil de base nécessaire aux applications présentées ci-dessus.

1.6 Conclusion

Afin de proposer une méthodologie de compression des images RSO SLC, il demeure indispensable de caractériser et de modéliser au mieux ce type d'images. La présence du chatoiement introduit une forte décorrélation des pixels rendant les méthodes classiques adaptées aux images optiques beaucoup moins efficaces. L'étude du chatoiement et de la modélisation statistique des scènes RSO permettent dans un premier temps d'envisager deux orientations possibles :

- la conservation du chatoiement comme une information utile avec une modélisation statisque des données adaptée à l'algorithme de compression.
- la réduction du chatoiement avant compression de l'image afin d'améliorer la qualité visuelle des images. Cela a également pour effet de réduire l'entropie de l'image (voir chapitre 2) et permet donc des taux de compression plus élevés.

Nous avons privilégié la première approche consistant à conserver le chatoiement avant compression. Les chapitres 4 et 5 de ce manuscrit sont ainsi dédiés à la compression d'images sans pré-traitement.

D'autre part, comme nous le verrons dans le chapitre 2, la compression d'une image avec pertes fait souvent intervenir une projection de l'image dans un espace transformé (transformée en ondelettes par exemple) présentant des propriétés intéressantes pour la compression. La modélisation statistique des images RSO SLC dans l'espace transformé devient alors une étape nécessaire dans l'élaboration d'un algorithme de compression efficace. C'est pour cette raison que nous introduisons le formalisme des statistiques de seconde espèce et de l'estimation des paramètres de la distribution de Fisher dans le chapitre 3, dédié à la méthodologie et aux différentes approches envisagées concernant les systèmes de compression des images RSO SLC.

L'exploitation de l'information de phase trouve notamment son utilité dans les techniques de traitement interférométriques permettant de remonter sur des grandeurs liées au relief. Nous avons ainsi brièvement présenté les principes de cette approche et introduit l'interférogramme comme étant l'outil de travail de base pour les applications interférométriques. Tout en mesurant l'impact de la compression avec pertes sur les images traitées individuellement, il peut s'avérer également intéressant de caractériser la dégradation introduite sur les données complexes après formation du couple interférométrique issu des images décompressées. Cette approche sera étudiée plus en détails au chapitre 6 de ce manuscrit, lors de l'évaluation des performances des algorithmes de compression.

Chapitre 2

Compression d'images, état de l'art et application aux images RSO

Ce second chapitre se concentre sur les outils utilisés en compression d'image. Le codage de source ou compression des données a pour objectif de fournir une représentation efficace des données tout en préservant l'information essentielle. Nous envisageons dans ce chapitre une présentation des principaux concepts de base et techniques associés à la compression d'image en général. Toutes ces méthodes ne sont pas forcément dédiées spécifiquement à la compression d'images RSO. Il n'en demeure pas moins que ces outils de compression forment le socle de base sur lequel nous pourrions nous appuyer pour développer une méthode spécifique. Nous évoquerons notamment quelques algorithmes de compression appliqués aux images RSO récemment publiés dans la littérature.

2.1 Introduction à la compression et critères d'évaluation

La compression consiste à réduire le volume de données nécessaire à la description de l'image. L'exploitation de la redondance spatiale et des faiblesses du système psychovisuel ont permis de développer des méthodes où la quantité de données nécessaires pour représenter l'image est réduite. Il existe deux types bien distincts de compression : la compression sans perte et avec pertes.

Dans le cas d'une compression sans perte, l'image reconstruite après décompression est identique à l'image originale. Aucune perturbation n'est introduite sur les valeurs des pixels. Ainsi, tout traitement effectué fournira les mêmes résultats, que l'image soit comprimée ou non. Au contraire, la compression avec pertes consiste à introduire une perturbation irréversible mais qui autorise une compression bien plus importante que celle obtenue par des méthodes sans perte.

Il existe une grande variété de techniques de compression d'images. Afin de pouvoir comparer ces méthodes, différents critères sont nécessaires. Quatre critères globaux permettent d'évaluer les performances d'un système de compression :

- Le taux de compression mesure la réduction du volume de données par rapport à la taille du fichier initial.
- La distorsion, dans le cadre de méthode de compression avec pertes, permet de mesurer la qualité de l'image restituée par rapport à l'image originale.
- La complexité de l'algorithme et sa facilité de mise en oeuvre.
- La robustesse du code due aux erreurs de transmission introduites par le canal de transmission.

2.1.1 Taux de compression

Ce critère est d'importance puisqu'il est directement dépendant de la technique de compression utilisée. Certaines méthodes proposent des taux de compression fixes (entré par exemple par l'utilisateur) ou non. On parlera alors de codage progressif où il est possible d'encoder mais surtout de décoder une image avec divers taux de compression. Afin de définir le taux de compression à proprement dit, nous introduisons au préalable la notion de débit comme étant le nombre moyen de bits par pixels nécessaires à la description de l'image.

$$\text{debit} = \frac{\text{Taille du fichier (bits)}}{\text{nombre de pixels}} \quad \text{en bits par pixel (bpp)} \quad (2.1)$$

Une image numérique est composée d'un ensemble de pixels présentant un nombre fixe d'états (niveaux de gris par exemple). A chacun de ces éléments est attribué un symbole ou code binaire. Pour les données RSO issues du capteur RAMSES, chaque pixel est composé de 2 réels, ce qui requiert 64 bits par pixel.

Le taux de compression η est défini comme le rapport entre le nombre de bits par pixels de l'image initiale et l'image comprimée (rapport des débits).

$$\eta = \frac{\text{debit de l'image avant compression}}{\text{debit de l'image après compression}} \quad (2.2)$$

Pour les méthodes de codage sans perte, le taux de compression est de l'ordre de 2 à 3 sur des images naturelles. Concernant les images RSO, le taux obtenu est encore plus faible à cause d'une plus forte décorrélacion des pixels. En revanche, les méthodes de compression avec pertes permettent d'obtenir des taux de compression plus élevés.

2.1.2 Distorsion et critère qualité

Pour les méthodes de compression avec pertes, un taux de compression n'a de sens que s'il est mis en correspondance avec la dégradation introduite sur l'image reconstruite. La qualité est une notion très difficile à définir. Pour un photo-interprète, une bonne qualité est synonyme d'images où les défauts sont imperceptibles à l'oeil. Sur une image RSO, des critères comme la restitution des points brillants, des textures, des contours sont évalués au moyen de tests subjectifs et spécifiques. Pour les images de télédétection, le problème est d'autant plus délicat que la notion de qualité dépend avant tout de l'information utile et de l'application envisagée.

Néanmoins, un critère largement utilisé en traitement d'images est la distorsion qui permet de mesurer une différence entre l'image originale et reconstruite. L'Erreur Quadratique Moyenne (EQM) définie à partir de la norme L^2 est le critère le plus courant. Soit I l'image originale de dimension $N \times M$ et \hat{I} l'image reconstruite après décompression, l'EQM est définie par :

$$EQM = \frac{1}{M N} \sum_{i=1}^M \sum_{j=1}^N [I(i, j) - \hat{I}(i, j)]^2 \quad (2.3)$$

Plus explicite, on préférera utiliser le rapport Signal sur Bruit RSB (ou SNR pour *Signal to Noise Ratio*) :

$$RSB = 10 \log_{10} \frac{\sum_{i=1}^M \sum_{j=1}^N I(i, j)^2}{\sum_{i=1}^M \sum_{j=1}^N [I(i, j) - \hat{I}(i, j)]^2} \quad (2.4)$$

Suivant ce critère, la qualité ainsi mesurée sera d'autant meilleure que le SNR sera grand et donc que la distorsion sera petite. Si l'image reconstruite après décompression est identique à l'image originale, le RSB est infini.

Dans le cadre de cette thèse, nous proposerons au chapitre 6 de ce manuscrit un ensemble de critères de mesure spécifiques à la caractérisation de la qualité des images RSO complexes reconstituées.

2.1.3 Complexité

La complexité d'un algorithme ou d'une méthode de compression est difficile à évaluer. Elle recouvre le volume du dispositif, sa rapidité, l'existence de composants intégrés, la charge de calcul. La complexité fait également intervenir les contraintes spécifiques à l'application, les performances des processeurs qui évoluent et le coût économique. Il est important de noter que la charge de calcul n'est pas forcément identique entre le codeur et le décodeur. On parlera alors de complexité asymétrique.

2.1.4 Robustesse

La robustesse caractérise la sensibilité de l'algorithme de compression-décompression à des altérations du code comprimé : une erreur binaire sur la trame doit être la moins perturbatrice possible. On ajoute alors un codage supplémentaire (codage de canal) qui permet de sécuriser et réduire le taux d'erreur binaire lors de la transmission des données. Notons que le rajout de cette redondance va bien entendu à l'encontre de la compression. Cette étape n'est cependant pas prise en compte dans notre étude.

2.2 Système de compression avec pertes

D'une manière générale, un système de compression avec pertes se décompose en trois modules que sont la transformation de l'image originale, la quantification des données de l'image transformée et le codage entropique des données quantifiées. La figure 2.1 illustre le principe général d'un système de compression avec pertes.

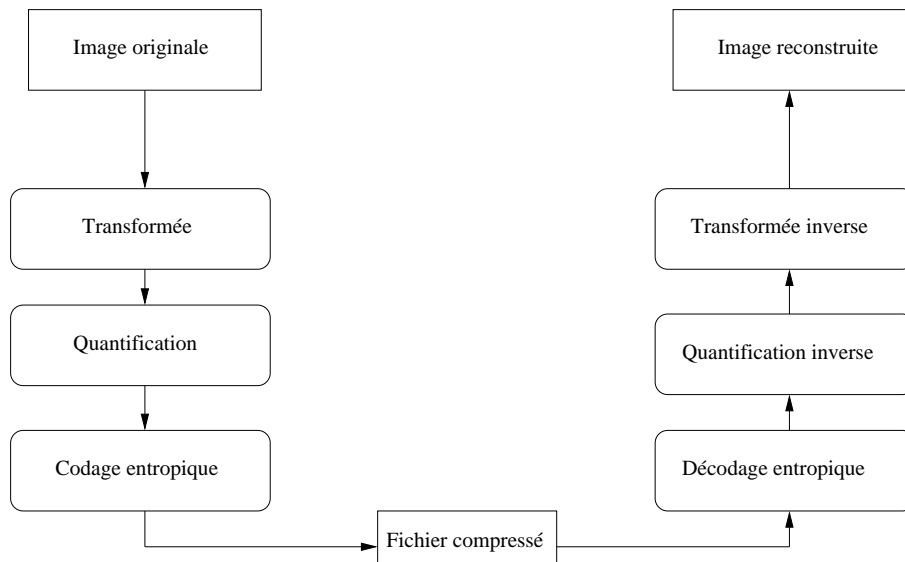


FIG. 2.1 – Principe général des algorithmes de compression/décompression d'image. À gauche : la chaîne de compression classique : transformée de l'image, quantification et codage entropique. À droite : la chaîne de décompression effectue les traitements dans l'ordre inverse.

La transformée a pour objectif de projeter les données originales dans un espace plus propice à la compression. En d'autres termes, il s'agit de décorrélérer le signal de manière à minimiser l'information redondante. Les pertes irréversibles sont introduites lors de la phase de quantification qui consiste à "réduire" l'espace de représentation des données. Le codage entropique (ou codage sans perte) permet d'encoder les données issues de la quantification suivant un code binaire à longueur variable dépendant de la "quantité d'information" portée par le message.

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

2.3 Compression sans perte et codage entropique

La théorie de l'information, développée par Shannon dans les années 1940-1950, fournit des idées fondamentales pour la conception d'un grand nombre de techniques de codage de source. L'idée est que des messages numériques peuvent être comprimés en associant les mots de code les plus courts aux messages les plus probables. Le taux de compression maximum que l'on peut atteindre sans avoir de perte peut être déterminé à partir d'une description statistique des messages.

Cette stratégie conduira à un codage à débit variable et à des techniques de codage sans perte (ou codage entropique), c'est à dire tel que les données reconstituées soient identiques aux données sources.

2.3.1 Entropie

Pour fixer les idées, on considère une source discrète X à valeurs dans un alphabet fini $A = \{a_1, a_2, \dots, a_M\}$. On suppose connue (ou estimée) la loi de probabilité des symboles de l'alphabet :

$$P : \{p(a_1), p(a_2), p(a_i), \dots, p(a_M)\} \quad (2.5)$$

où $p(a_i)$ est la probabilité d'occurrence du symbole a_i tel que $\sum_{i=1}^M p(a_i) = 1$.

Une source est dite sans mémoire si les symboles émis par la source X correspondent à des variables aléatoires indépendantes et identiquement distribuées. En d'autres termes, une suite de symboles $\{a_{t_1}, a_{t_2}, \dots, a_{t_n}\}$ correspondant à des symboles émis aux instants t_1, \dots, t_n suit une loi :

$$p(a_{t_1}, a_{t_2}, \dots, a_{t_n}) = p(a_{t_1})p(a_{t_2}) \dots p(a_{t_n})$$

Dans ce cas, l'information propre associée au symbole a_i est définie par :

$$I(a_i) = -\log_2 p(a_i) \text{ en bits.} \quad (2.6)$$

Cette définition implique que l'information transportée est d'autant plus importante lorsqu'un symbole peu probable est généré. L'entropie d'ordre zéro d'une source discrète sans mémoire X traduit la quantité d'information moyenne. Elle est définie par l'expression :

$$H(X) = -\sum_{i=1}^M p(a_i) \log_2 p(a_i) \quad (2.7)$$

La notion d'information est ainsi reliée à la notion d'incertitude par l'intermédiaire de la distribution de probabilité de la source. L'incertitude la plus grande concerne une source dont les symboles suivent une loi P uniforme, c'est à dire que les symboles a_i sont équiprobables. L'entropie y est donc maximale :

$$\forall i \in 1, 2, \dots, M \quad p(a_i) = \frac{1}{M} \Rightarrow H(X) = \log_2(M) \quad (2.8)$$

L'entropie sera d'autant plus faible que la loi sera "étroite". Une source déterministe (n'apportant pas d'incertitude) aura une entropie nulle et n'apportera donc pas en soit

d'information.

L'entropie d'ordre zéro d'une image donne donc la limite minimale théorique du nombre moyen de bits par symbole nécessaire pour coder sans ambiguïté une source de variables aléatoires indépendantes.

Si l'on considère deux sources discrètes X et Y , prenant leur valeur respectivement dans $A = \{a_1, \dots, a_M\}$ et $B = \{b_1, \dots, b_N\}$, leur probabilité conjointe s'écrit :

$$p_{XY}(a_i, b_j) = \text{Prob}\{X = a_i, Y = b_j\} \quad (2.9)$$

L'entropie du premier ordre ou conjointe des deux sources X et Y s'exprime en fonction de l'information conjointe $I(a_i, b_j) = -\log_2 p_{XY}(a_i, b_j)$ des deux événements :

$$H(X, Y) = - \sum_{i=1}^M \sum_{j=1}^N p_{XY}(a_i, b_j) \cdot \log_2 p_{XY}(a_i, b_j) \quad (2.10)$$

De même, il est possible de définir l'entropie conditionnelle de la source Y sachant X :

$$H(Y/X) = - \sum_{i=1}^M \sum_{j=1}^N p_{Y/X}(b_j/a_i) \cdot \log_2 p_{Y/X}(b_j/a_i) \quad (2.11)$$

En combinant les expressions précédentes en fonction des probabilités conditionnelles et conjointes, on obtient alors la relation suivante :

$$H(X, Y) = H(X) + H(Y/X) \leq H(X) + H(Y) \quad (2.12)$$

L'égalité est réalisée si X et Y sont des variables indépendantes. Par conséquent, si les sources ne sont pas indépendantes alors on en déduit facilement que : $H(Y/X) \leq H(Y)$. Ceci montre en particulier que la "connaissance réduit l'incertitude" et qu'il est donc plus intéressant d'utiliser l'information apportée conjointement par les sources plutôt que de les considérer séparément.

Dans le contexte de compression d'images RSO, les pixels sont affectés par le phénomène de chatoiement qui a tendance à introduire une certaine incertitude et indépendance entre les pixels. L'entropie y est donc plus élevée par rapport à des images naturelles, ce qui explique en partie qu'elles soient plus difficiles à compresser.

2.3.2 Condition du préfixe et Inégalité de Kraft

Un codeur entropique a pour objectif de coder la source X sans perte. Le codeur fait correspondre à chaque symbole d'entrée a_i (réalisation de la variable aléatoire X à valeurs possibles dans l'alphabet A), un mot de code binaire $c(a_i)$ de longueur $l(a_i)$ (mesurée en bits). Le décodeur reconstruit les symboles de source à partir de la séquence binaire des mots de codes. Le but du codage est de garder un nombre de bits transmis aussi faible que possible c'est à dire que l'on cherche à minimiser la longueur moyenne \tilde{l} du code (le nombre moyen de bits codés par symbole de source) dont l'expression est donné par :

$$\tilde{l} = \sum_{i=1}^M p(a_i) \cdot l(a_i) \quad (2.13)$$

Pour que le codage n'introduise pas de perte, il faut que le décodage ait lieu sans ambiguïté. En d'autres termes, une séquence codée donnée doit pouvoir être interprétée

de façon unique comme une succession (concaténation) de mots de code déterminés. Un code permettant un tel décodage est qualifié d'uniquement décodable (u.d).

Pour satisfaire cette condition sans introduire de séparateurs entre les mots de code, le code doit satisfaire la condition dite du préfixe. Cette dernière stipule qu'aucun mot de code ne doit être le préfixe d'un autre mot de code.

Exemples :

$\{0, 10, 101, 0101\}$ n'est pas un code à préfixe.

$\{0, 10, 110, 111\}$ est un code à préfixe.

La plus importante propriété des codes uniquement décodables (codes à préfixes) est donnée par l'inégalité de Kraft.

Inégalité de Kraft

Une condition nécessaire et suffisante pour qu'un code binaire correspondant à la source d'alphabet $A = \{a_1, a_2, \dots, a_M\}$ et constitué de mots de code de longueur $\{l(a_1), l(a_2), \dots, l(a_M)\}$ soit uniquement décodable est que :

$$\sum_{i=1}^M \frac{1}{2^{l(a_i)}} \leq 1 \quad (2.14)$$

Cette inégalité fournit la base théorique pour calculer des bornes inférieures et supérieures à la longueur moyenne d'un code uniquement décodable. Ce résultat permet également de montrer qu'une source X d'entropie $H(X)$ peut être codée sans ambiguïté de décodage, à l'aide d'un code C de longueur moyenne \tilde{l} satisfaisant l'inégalité suivante :

$$H(X) \leq \tilde{l} \leq H(X) + 1 \quad (2.15)$$

Il est donc possible de trouver un code à un bit près de l'entropie. Nous verrons qu'on ne peut pas améliorer cette distance pour des codes "scalaires". En revanche, on peut diminuer cette imprécision en appliquant les résultats précédents à la source "étendue", c'est-à-dire en formant des vecteurs aléatoires \mathcal{X}_N groupant N variables aléatoires.

$$\mathcal{X}_N = \{X_1, \dots, X_N\}$$

En généralisant l'expression (2.11), l'entropie conjointe de la source est donnée par :

$$H(\mathcal{X}_N) = - \sum_{a^N} p_{\mathcal{X}_N}(a^N) \log_2(p_{\mathcal{X}_N}(a^N)) \quad (2.16)$$

où a^N représente un N -uplet de valeurs prises dans l'alphabet A .

Il y a donc M^N possibilités pour la valeur de a^N . Ces probabilités sont généralement difficiles à établir sauf dans le cas où les sources X_i sont indépendantes et identiquement distribuées (i.i.d), auquel cas on a :

$$p_{\mathcal{X}_N}(a^N) = \prod_{a_i \in a^N} p_X(a_i) \Rightarrow H(\mathcal{X}_N) = N.H(X) \quad (2.17)$$

Si on associe un mot de code à chaque groupe de N symboles a^N , alors d'après l'inégalité (2.15), la longueur moyenne du code vérifie :

$$H(\mathcal{X}_N) \leq \tilde{l}_N \leq H(\mathcal{X}_N) + 1 \quad (2.18)$$

avec $\tilde{l}_N = \sum_{a^N} p_{\mathcal{X}_N}(a^N) \cdot l(a^N)$.

Le nombre de bits moyen par symbole $\tilde{l} = \frac{\tilde{l}_N}{N}$ vérifie alors :

$$\frac{H(\mathcal{X}_N)}{N} \leq \tilde{l} \leq \frac{H(\mathcal{X}_N) + 1}{N} \quad (2.19)$$

Soit pour une source i.i.d :

$$H(X) \leq \tilde{l} \leq H(X) \frac{1}{N} \quad (2.20)$$

Dans tous les cas, ce résultat nous indique implicitement qu'il n'est pas possible de trouver un codeur qui permette de compresser la source X, sans perte avec un débit \tilde{l} inférieur à l'entropie. En particulier, le théorème de Shannon démontre de manière théorique, qu'il est possible de construire un code uniquement décodable de sorte que la longueur moyenne des mots de code s'approche de l'entropie de la source $H(X)$. La prise en compte du codage vectoriel de symboles garantit donc l'existence d'un code dont la longueur moyenne dispose d'une borne maximale permettant de se rapprocher encore plus de l'entropie de la source.

2.3.3 Algorithme de Huffman

L'algorithme de codage de Huffman permet de construire un code à longueur variable uniquement décodable dont la longueur moyenne s'approche de l'entropie de la source. Le principe illustré par la figure 2.2 est le suivant :

1. Les symboles sont classés par ordre de probabilité décroissante. On assigne alors aux deux symboles de probabilité les plus faibles les bits 0 et 1.
2. Les deux symboles sont combinés en un symbole fictif dont la probabilité est la somme des probabilités des symboles élémentaires. On attribue arbitrairement un nouveau bit 0 et 1 à chaque branche.
3. La procédure 1) et 2) est réitérée jusqu'à ce que la liste ne comporte plus que deux éléments auxquels on affecte les bits 0 et 1.

Le code pour chaque symbole initial est alors obtenu en partant de la racine de l'arbre jusqu'aux feuilles.

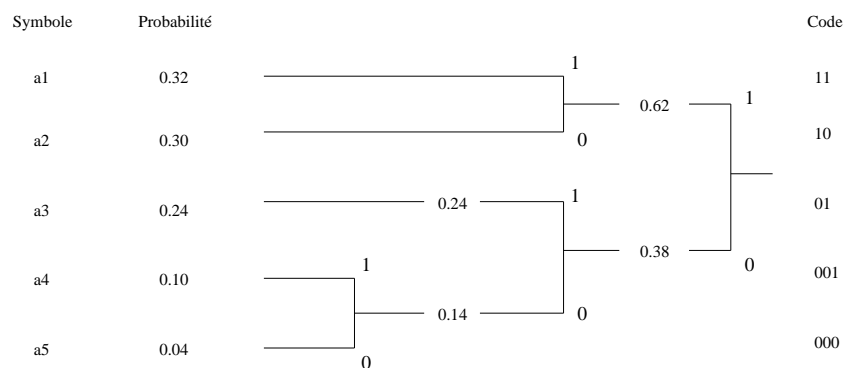


FIG. 2.2 – Principe du codage d'Huffman

Dans cet exemple, l'entropie d'ordre zéro de la source est de $H(X) = 2.06$ bits par symbole. A titre de comparaison, la longueur moyenne du code obtenu après codage de Huffman est donnée par :

$$\tilde{l} = 2p(a_1) + 2p(a_2) + 2p(a_3) + 3p(a_4) + 3p(a_5) = 2.14 \text{ bits/symbole}$$

Contrairement au codage de Shannon-Fano qui part de la racine et opère par divisions successives, le code de Huffman commence par les feuilles de l'arbre en construisant les mots de code par fusions successives des événements les moins probables. Le décodage nécessite de connaître la table de correspondance symboles-code. Une solution consiste à utiliser une table précalculée universelle ou adaptée à une application comme c'est le cas pour la norme JPEG [75].

Le code de Huffman est couramment utilisé et peu complexe. Cependant, il n'est pas optimal car il ne s'approche que de la partie entière de l'information portée par chaque symbole (l'optimalité étant alors obtenue pour un jeu de probabilité du type $p(a_i) = 2^{-l}$). On peut, dans le cas général, s'approcher davantage de l'entropie en employant le code de Huffman appliqué à une source étendue, c'est à dire en regroupant les symboles par bloc. Cependant, la complexité de mise en oeuvre croit très rapidement.

2.3.4 Codage Arithmétique

Le codage arithmétique [109] a connu un succès grandissant depuis les 20 dernières années. A l'aide d'opérations arithmétiques simples, cet algorithme permet de construire un seul mot de code à partir d'une longue séquence arbitraire de symboles. Le décodage est progressif, c'est-à-dire qu'il n'est pas nécessaire de lire le mot de code en entier pour pouvoir décoder une partie du message. On évite ainsi de garder en mémoire une collection gigantesque de mots de code.

Le codage arithmétique permet de se rapprocher de la limite de Shannon à partir de la connaissance des probabilités des symboles $P : \{p(a_1), p(a_2), \dots, p(a_M)\}$ de l'alphabet $A = \{a_1, a_2, \dots, a_M\}$. La construction du code étant incrémentale, ce code est également adapté aux sources de Markov (avec mémoire). Les probabilités conditionnelles peuvent en effet, être estimées de manière adaptative à partir des données précédemment encodées. Le principe du codage arithmétique est le suivant :

- Il traite un message composé de N symboles comme un mot unique (au sens de la source étendue de Huffman).
- Le principe consiste à remplacer ce mot unique par un nombre réel compris dans l'intervalle $[0, 1[$ en précision finie. Ce nombre réel est défini dans l'algorithme par un intervalle compris dans $[0, 1[$, que l'on appellera intervalle courant I_c et dont la longueur est égale à la probabilité de la suite des symboles du message.

Algorithme d'encodage

1. Conditions initiales :

- Chaque symbole $a_i \forall i \in \{1, 2, \dots, M\}$ se voit attribuer un intervalle dans $[0, 1[$ de longueur égale à sa probabilité. Par exemple, pour un alphabet à $M = 2$ symboles avec $A = \{a_1, a_2\}$ et $P : \{p(a_1) = 7/8, p(a_2) = 1/8\}$, on définit la longueur de l'intervalle associé $l(a_1) = p(a_1) = 7/8$ et $l(a_2) = p(a_2) = 1/8$.
- On définit les limites de chaque intervalle par des bornes inférieures et supérieures notées respectivement B_{inf} et B_{sup} . Dans notre exemple, $B_{inf}(a_1) = 0$, $B_{sup}(a_1) = 7/8$ et $B_{inf}(a_2) = 7/8$, $B_{sup}(a_2) = 1$.
- On initialise la longueur de l'intervalle courant $l(I_c) = 1$ avec $B_{inf}(I_c) = 0$ et $B_{sup}(I_c) = 1$.

2. Tant qu'il reste des symboles dans le message à coder :

- on considère le nouveau symbole émis $a_i^{(n)}$. On associe à chaque symbole émis, un sous intervalle dont la longueur est proportionnelle à sa probabilité d'occurrence. A partir de l'intervalle courant, le nouvel intervalle est calculé de la manière suivante :
 - $B_{inf}(I_c) = B_{inf}(I_c) + B_{inf}(a_i^{(n)}) * l(I_c)$
 - $B_{sup}(I_c) = B_{inf}(I_c) + B_{sup}(a_i^{(n)}) * l(I_c)$
 - $l(I_c) = B_{sup}(I_c) - B_{inf}(I_c)$

3. On code l'intervalle courant I_c par un mot binaire de longueur $\lceil -\log_2(l(I_c)) \rceil$ bits.

D'un point de vue implémentation, au lieu de subdiviser l'intervalle courant en un nouvel intervalle plus restreint, le sous-intervalle correspondant au symbole à coder est "dilaté" à l'intervalle $[0, 1[$. La difficulté de cet algorithme réside principalement dans la gestion de la précision numérique croissante du code, limité en pratique par la précision finie des machines. Pour plus de détails concernant cet aspect de l'algorithme, nous recommandons le lecteur de se référer à la publication de Witten et al [109].

Algorithme de décodage

- On traduit le code binaire reçu en nombre compris appartenant à l'intervalle $[0, 1[$ appelé nombre courant NbC associé au n-ième symbole.
- Tant qu'il reste des symboles à décoder :
 1. On recherche un symbole $a_i^{(n)}$ tel que $NbC \in [B_{inf}(a_i^{(n)}), B_{sup}(a_i^{(n)})]$
 2. On extrait le symbole $a_i^{(n)}$.
 3. Mise à jour du nombre courant dans l'intervalle $[0, 1[: NbC = \frac{NbC - B_{inf}(a_i^{(n)})}{l(a_i^{(n)})}$

Exemple

En considérant la source X à 2 symboles avec $P : \{p(a_1) = 7/8, p(a_2) = 1/8\}$ et d'entropie $H(X) = 0.546$ bits/symbole, la table 2.3.4 illustre le principe du codage arithmétique pour une séquence donnée. Suivant la procédure décrite précédemment, l'encodage d'une séquence de symboles associés à la loi P se déroule de la manière suivante :

Symboles	$B_{inf}(I_c)$	$B_{sup}(I_c)$	$l(I_c)$	Longueur du code
a_1	$0.0000.10^0$	$8.7500.10^{-1}$	$8.7500.10^{-1}$	1
a_1	$0.0000.10^0$	$7.6562.10^{-1}$	$7.6562.10^{-1}$	1
a_1	$0.0000.10^0$	$6.6992.10^{-1}$	$6.6992.10^{-1}$	1
a_1	$0.0000.10^0$	$5.8618.10^{-1}$	$5.8618.10^{-1}$	1
a_2	$5.1290.10^{-1}$	$5.8618.10^{-1}$	$7.3272.10^{-2}$	4
a_1	$5.1290.10^{-1}$	$5.7702.10^{-1}$	$6.4113.10^{-2}$	4
a_1	$5.1290.10^{-1}$	$5.6900.10^{-1}$	$5.6099.10^{-2}$	5
a_2	$5.6199.10^{-1}$	$5.6900.10^{-1}$	$7.0124.10^{-3}$	8
a_1	$5.6199.10^{-1}$	$5.6813.10^{-1}$	$6.1358.10^{-3}$	8
a_1	$5.6199.10^{-1}$	$5.6736.10^{-1}$	$5.3688.10^{-3}$	8
a_1	$5.6199.10^{-1}$	$5.6669.10^{-1}$	$4.6977.10^{-3}$	8
a_1	$5.6199.10^{-1}$	$5.6610.10^{-1}$	$4.1105.10^{-3}$	8
a_1	$5.6199.10^{-1}$	$5.6559.10^{-1}$	$3.5967.10^{-3}$	9
a_1	$5.6199.10^{-1}$	$5.6514.10^{-1}$	$3.1471.10^{-3}$	9
a_1	$5.6199.10^{-1}$	$5.6474.10^{-1}$	$2.7537.10^{-3}$	9
a_1	$5.6199.10^{-1}$	$5.6440.10^{-1}$	$2.4090.10^{-3}$	9

TAB. 2.1 – Exemple d'un encodage arithmétique d'une séquence de symboles

La séquence émise de 16 symboles nécessite donc un mot de code de 9 bits, soit un débit de 0.5625 bits par symbole ce qui est proche de l'entropie de 0.546 bits par symbole.

En supposant que l'on s'arrête après le sixième symbole, le code émis sur 4 bits vaut $\{1001\}$ (0.5625). Le décodage se déroule de la manière suivante :

- le nombre courant NbC lu à partir du code $\{1001\}$ est 0.5625
- $NbC = 0.5625 < B_{sup}(a_1)$ donc le premier symbole est a_1
- $NbC = (0.5625 - 0)/0.875 = 0.6429 < B_{sup}(a_1)$
donc le symbole suivant est a_1
- $NbC = (0.6429 - 0)/0.875 = 0.7347 < B_{sup}(a_1)$
donc le symbole suivant est a_1
- $NbC = (0.7347 - 0)/0.875 = 0.8397 < B_{sup}(a_1)$
donc le symbole suivant est a_1
- $NbC = (0.8397 - 0)/0.875 = 0.9596 > B_{sup}(a_1)$
donc le symbole suivant est a_2
- $NbC = (0.9596 - 0.875)/0.875 = 0.6768 < B_{sup}(a_1)$
donc le symbole suivant est a_1

Nous retrouvons donc bien les six premiers symboles $\{a_1, a_1, a_1, a_1, a_2, a_1\}$ de la séquence avant compression.

Notons qu'il existe une version adaptative de cet algorithme où la table de probabilité est mise en fonction de la fréquence empirique d'occurrence des symboles et ne nécessite pas forcément la connaissance a priori des tables de probabilité. En raison de sa simplicité et de ses performances, cet algorithme est présent dans les techniques de compression récentes notamment JPEG2000 [93].

2.3.5 Algorithme LZW

D'autres systèmes de codage de source sans perte ont été proposés pour prendre en compte les dépendances temporelles (d'un symbole à l'autre) de la source avec mémoire. Ces systèmes de codage permettent de coder une source quelconque sans connaître a priori ses statistiques mais sont plus complexes à mettre en oeuvre.

L'un des plus connus est l'algorithme LZW proposé par Ziv, Lempel et Welch [117] [111]. Le principe est de coder des chaînes de caractères, c'est-à-dire de rassembler une suite de caractères en une chaîne et d'y associer un mot de code binaire. Il s'agit de chercher à découper les données en des chaînes les plus longues possibles. Ces dernières sont stockées dans une table de traduction, identifiées par leur adresse dans la table. Le mot de code transmis est alors simplement l'adresse (mot de code de longueur constante). L'algorithme d'encodage a l'inconvénient d'être très gourmand en temps de calcul à l'inverse du décodage.

2.3.6 Codage RLE (Run Length Encoding)

Lorsque l'alphabet est restreint, le codage RLE permet d'encoder de manière efficace les longues suites de symboles répétitives. Toute suite de bits ou de caractères identiques est remplacée par un couple (nombre d'occurrences/bit ou caractère répété).

2.3.7 Conclusion

Les quelques méthodes de codage sans perte présentées sont loin d'être exhaustives. Les techniques présentées ci-dessus ont l'avantage de comprimer les données sans distor-

sion. En contrepartie, les taux de compression obtenus ne sont pas importants car limités par l'entropie de la source beaucoup trop élevée. Ce taux est d'ailleurs difficilement maîtrisable dans le sens où la distribution et l'entropie des symboles relatifs à la description d'une image sont différents d'une image à l'autre. Néanmoins si les caractéristiques statistiques de l'image sont considérées comme stables, les distributions varieront peu et il sera d'autant plus facile de modéliser et d'associer des modèles de probabilité adaptés à la source.

Appliqué à des techniques de compression avec pertes, le codage entropique n'est pas appliqué directement sur les pixels de l'image. D'une manière générale, il s'agit d'encoder sans perte l'index des symboles issus de la quantification des données. Nous consacrerons ainsi la prochaine section à décrire les principes de la quantification.

2.4 Quantification

L'étape de quantification apparaît dans les méthodes de compression avec pertes en réduisant le nombre de paramètres associés à la description du signal. Lorsqu'il y a réduction de la dimension de l'espace des paramètres, on parle de quantification et celle-ci s'accompagne d'une dégradation irréversible du signal. On associe ainsi à l'ensemble des paramètres un ensemble fini de valeurs constituant un dictionnaire de représentants (*codebook* en anglais). L'ensemble des représentants du dictionnaire est répertorié par un index.

A la suite de la quantification, l'index, correspondant à chaque représentant du dictionnaire, peut être encodé par un codage à longueur fixe, soit un codage entropique à longueur variable de manière à améliorer la réduction du débit. La quantification peut être opérée indépendamment échantillon par échantillon (quantification scalaire), ou effectuée par blocs d'échantillons (quantification vectorielle).

2.4.1 Quantification scalaire

La quantification scalaire est l'application qui permet d'associer à un scalaire x d'un ensemble U (\mathbb{R} par exemple) un autre scalaire y_i d'un ensemble $Y = \{y_i, i = 1, \dots, L\}$ correspondant au dictionnaire. L'application est ainsi définie de la manière suivante :

$$\begin{aligned} U &\longmapsto Y = \{y_1, y_2, \dots, y_i, \dots, y_L\} \subset \mathbb{R} \\ x &\longmapsto y_i = Q(x) \end{aligned}$$

Les valeurs y_i sont appelées valeurs de quantification ou représentants. Dans la pratique, le nombre de représentants L est fini et peut être une puissance de 2 tel que $L = 2^b$. Dans ce cas, on peut lors du codage, spécifier une valeur quantifiée y_i par son index i sur b bits. Si l'on utilise un codage entropique, on associera aux L valeurs d'index i des chaînes binaires (mot de code) à longueur variable. On obtiendra alors une longueur moyenne des mots tendant vers l'entropie de l'index.

On associe à un quantificateur de L représentants y_i une partition des valeurs x en cellules ou intervalles U_i tel que :

$$\begin{aligned} x \in U_i &\Rightarrow Q(x) = y_i \\ \cup U_i &= U \\ U_i \cap U_j &= \emptyset \quad \forall i \neq j \end{aligned}$$

Chaque cellule U_i est de la forme $]x_{i-1}, x_i]$ avec $y_i \in U_i$. On définit alors le pas de quantification par la taille de chaque cellule $\Delta_i = x_i - x_{i-1}$. Si les valeurs d'entrée sont bornées, x_0 et x_L représentent les bornes de variations de x . Une cellule qui n'est pas bornée est appelée cellule de saturation ou surcharge.

La construction d'un quantificateur se résume par la question suivante : comment définir le nombre de régions, leur partition (c'est à dire les niveaux de décisions x_i), et les représentants y_i ? Une réponse possible consiste à définir un critère de qualité comme la distorsion. La quantification est alors optimale si le quantificateur est conçu de sorte que la mesure de distorsion soit minimale.

Mesure de performance d'un quantificateur

Quand une valeur d'entrée x est quantifiée par $Q(x) = y_i$, on introduit une erreur de quantification $e = x - Q(x)$. Cette grandeur est bien sûr aléatoire puisque x l'est. Pour mesurer la performance du quantificateur, on évalue généralement la distorsion moyenne comme la distance quadratique moyenne :

$$D = E[(X - Q(X))^2] = \sum_{i=1}^L \int_{U_i} (x - y_i)^2 f_X(x) dx \quad (2.21)$$

où $f_X(x)$ est la distribution de probabilité de la source X .

L'utilisation de ce critère est simple et motivé par son sens immédiat de puissance moyenne d'un signal d'erreur bien qu'il ne corresponde pas à une mesure subjective de qualité. On distingue deux types d'erreur de quantification :

- l'erreur de granulation qui est l'erreur pour une entrée x appartenant à l'une des cellules bornées $U_2 \dots U_{L-1}$ et qui est inférieure ou égale au demi-pas de quantification de la cellule correspondante.
- l'erreur de saturation lorsque $x < x_0$ ou $x > x_L$.

En principe, l'erreur de granulation est relativement faible en comparaison à l'erreur de saturation mais intervient pour presque toutes les valeurs d'entrées. L'erreur de saturation est plus rare mais peut avoir une très grande amplitude.

Quantification uniforme

La quantification uniforme est la plus répandue et la plus simple. Il s'agit d'un quantificateur régulier pour lequel les niveaux de décisions x_i sont également espacés et les représentants y_i se situent au milieu des intervalles U_i . En d'autres termes :

$$\begin{cases} x_i - x_{i-1} = \Delta, \quad \forall i = 1 \dots L \\ y_i = \frac{x_{i-1} + x_i}{2}, \quad \forall i = 1 \dots L \end{cases}$$

Lorsqu'une source X , à valeurs uniformément réparties dans l'intervalle $[x_0, x_L]$, est quantifiée par une quantification scalaire uniforme, la distorsion introduite au sens de l'erreur quadratique moyenne est donnée par :

$$D = E[(X - Q(X))^2] = \frac{\Delta^2}{12} \quad (2.22)$$

Lorsque la source est uniformément distribuée, le quantificateur uniforme est optimal et permet d'obtenir le minimum de l'erreur quadratique moyenne. En revanche, lorsque la distribution du signal n'est pas uniforme, le quantificateur optimal n'est pas uniforme. En adaptant ainsi les seuils de quantification aux caractéristiques statistiques de la source, il est possible de réduire notablement la distorsion.

Quantification non-uniforme

On suppose que l'on cherche à quantifier de façon optimale sur L niveaux une réalisation d'un processus aléatoire stationnaire de loi $f_X(x)$ connue (ou estimable). Intuitivement, la longueur des intervalles $[x_{i-1}, x_i]$ sera d'autant plus petite que $f_X(x)$ sera importante sur ces intervalles.

Ce problème n'admet pas de solution simple. Si l'on connaît les représentants y_i , on peut trouver la meilleure partition $U_i = [x_{i-1}, x_i]$. A l'inverse, si l'on se donne la partition, on peut en déduire les représentants. Il existe alors deux conditions d'optimalité :

1. Condition du plus proche voisin :

Etant donné le dictionnaire $Y = y_1, \dots, y_L$, la partition U_i est donnée par l'ensemble des x_i tels que :

$$x_i = \frac{y_i + y_{i+1}}{2}$$

En d'autres termes, la frontière se situe au milieu des représentants.

Ceci correspond à la *règle du plus proche voisin*

$$U_i = \{x / (x - y_i)^2 \leq (x - y_j)^2 \quad \forall j \in \{1, \dots, L\}\}$$

2. Condition du centroïde :

Etant donné une partition $U_i : [x_{i-1}, x_i] \quad \forall i \in \{1, \dots, L\}$, on recherche les représentants y_i qui minimisent la distorsion quadratique moyenne D définie par la formule (2.21). On montre alors que la solution optimale est donnée par :

$$y_i = \frac{\int_{x_{i-1}}^{x_i} x f_X(x) dx}{\int_{x_{i-1}}^{x_i} f_X(x) dx} \Leftrightarrow y_i = \int_{-\infty}^{+\infty} x f_{X/U_i}(x) dx$$

En d'autres termes :

$$y_i = E(X / X \in U_i) \quad (2.23)$$

Le représentant y_i est donc le *centroïde* de la cellule U_i , c'est à dire la valeur moyenne de x dans l'intervalle $[x_{i-1}, x_i]$.

Algorithme de Lloyd-Max

Dans la pratique, on ne connaît généralement pas la densité de probabilité de la source $f_X(x)$. Pour construire le quantificateur, on utilise une base d'apprentissage composée d'un grand nombre N ($N \gg L$) d'échantillons représentatifs de la source que l'on veut quantifier. L'algorithme de Lloyd-Max est un algorithme itératif qui vérifie alternativement les deux conditions d'optimalité.

1. On initialise le dictionnaire $Y = \{y_1, \dots, y_L\}$ de manière aléatoire (ou uniforme).
2. Connaissant le dictionnaire, on classe tous les échantillons $x(n)$ suivant leur cellule d'appartenance. En appliquant la règle du plus proche voisin (Condition du plus proche voisin), on détermine ainsi implicitement les ensembles $x(n) \in U_i$ tel que :

$$Q(x(n)) = y_i \Rightarrow x(n) \in U_i$$

On calcule la distorsion D associée à cette classification.

3. A partir de ces classes, on calcule de nouveaux représentants y_i qui sont les moyennes des échantillons appartenant à une classe (Condition du Centroïde).
4. On reprend les étapes 2) et 3) tant que la distorsion ne décroît plus de manière significative ou qu'un nombre d'itérations maximum n'a pas encore été atteint.

Performances d'un quantificateur scalaire optimal en haute résolution

En se plaçant sous l'hypothèse haute-résolution ¹, on peut démontrer que la distorsion quadratique moyenne de quantification (quantificateur de type Lloyd-Max) d'une source i.i.d de moyenne nulle est donnée par [30] :

$$D(R) = \frac{1}{3} \left[\int_{-\infty}^{+\infty} f_X^{1/3}(x) dx \right] \cdot 2^{-2R} \quad (2.24)$$

où $R = \log_2(L)$ correspond au débit et est en général un nombre entier de bits par symbole. La taille du dictionnaire L est donc choisie comme une puissance de deux.

Soit $\sigma^2 = \int_{-\infty}^{+\infty} x^2 f_X(x) dx$, le moment centré d'ordre 2 associé à la densité de probabilité de la source. La table 2.2 récapitule l'expression $D(R)$ pour des sources de d.d.p Uniforme, Laplacienne et Gaussienne.

Avec l'augmentation du débit R , on en déduit facilement que la distorsion décroît de 6.02dB par bit supplémentaire. Shannon a montré que la fonction de débit-distorsion $D(R)$ est bornée par une limite théoriquement atteignable. En particulier pour un processus gaussien i.i.d de variance σ^2 , la limite théorique est donnée par :

$$D(R) = \sigma^2 \cdot 2^{-2R} \quad (2.25)$$

Pour R fixé, il existe donc un écart d'un facteur $\frac{\sqrt{3\pi}}{2}$ soit 4.35dB entre le quantificateur de Lloyd-Max optimal et la borne de Shannon.

¹c'est à dire un nombre de cellules de quantification suffisamment important pour considérer la densité de probabilité $f_X(x)$ comme une constante à l'intérieur de la cellule de quantification

d.d.p	D(R)
Uniforme	$D(R) = \frac{1}{12}\sigma^2 \cdot 2^{-2R}$
Laplacienne	$D(R) = \frac{1}{9}\sigma^2 \cdot 2^{-2R}$
Gaussienne	$D(R) = \frac{\sqrt{3}\pi}{2}\sigma^2 \cdot 2^{-2R}$

TAB. 2.2 – Expression de la fonction débit-distorsion d’un quantificateur scalaire optimal associé à trois d.d.p.

2.4.2 Quantification scalaire à contrainte entropique

Le quantificateur scalaire de Lloyd-Max a pour objectif de minimiser l’erreur quadratique moyenne sous une contrainte liée à la résolution du quantificateur fixée à L cellules, soit un débit correspondant de $R = \log_2(L)$ bits/symbole. Le code permettant d’encoder les indices est alors à longueur fixe.

En considérant un codage entropique à longueur variable des indices de quantification, le débit théoriquement accessible en sortie du codeur est proche de l’entropie des représentants du dictionnaire de quantification Y . Le codage entropique offre donc un intérêt si :

$$R \simeq H(Y) \leq \log_2(L) \quad (2.26)$$

Dans le cas où $H(Y) < \log_2(L)$, l’algorithme de Lloyd-Max n’est donc pas optimal. Pour un budget R fixé, le quantificateur optimal consiste à minimiser la distorsion quadratique moyenne $E[(X - Y)^2]$ de l’équation (2.21) sous la contrainte entropique :

$$H(Y) = - \sum_{i=1}^L p_i \log_2 p_i \leq b \quad (2.27)$$

avec

$$p_i = p(y_i = Q(x)) = \int_{U_i} f_X(x) dx \quad (2.28)$$

A l’aide de la technique des multiplicateurs de Lagrange, ce problème se résoud par une minimisation sans contrainte de la fonction $J(\lambda)$ telle que :

$$J(\lambda) = E[(X - Y)^2] + \lambda \cdot (H(Y) - R) \quad (2.29)$$

On montre alors que sous l’hypothèse haut-débit (R important) le quantificateur scalaire optimal à contrainte entropique est uniforme [36]. En supposant un codage entropique tel que $R \simeq H(Y)$, le pas de quantification du quantificateur optimal est donné par :

$$\Delta = 2^{h(X)-R} \quad (2.30)$$

La fonction de débit-distorsion est donnée par :

$$D(R) \simeq \frac{1}{12} 2^{2h(X)-2R} \quad (2.31)$$

où

$$h(X) = E[\log_2 f_X(X)] = - \int f_X(x) \log_2 f_X(x) dx \quad (2.32)$$

est l'entropie différentielle de la source ². En particulier, l'entropie différentielle peut être négative et est maximale pour une source de d.d.p gaussienne avec $h(X) = 0.5 \log_2(2\pi e\sigma^2)$.

A l'aide de la formule (2.31), on en déduit l'expression de la fonction débit-distorsion pour une variable aléatoire gaussienne i.i.d de variance σ^2 :

$$D(R) = \frac{\pi e}{2} \sigma^2 \cdot 2^{-2R}$$

Comparé à la borne de Shannon (formule 2.25), le quantificateur scalaire optimal à contrainte entropique affiche un écart d'un facteur $\frac{\pi e}{2}$ soit $1.53dB$. Par rapport au quantificateur scalaire optimal, le gain apporté par le quantificateur avec contrainte entropique est de $(\sqrt{3}\pi/2)/(\pi e/2) \approx 1.91$ soit $2.81dB$.

Quantificateur scalaire à zone morte

Sous l'hypothèse faible-débit, le quantificateur scalaire à contrainte entropique uniforme n'est cependant plus optimal. En supposant la source centrée, il est possible d'améliorer légèrement la courbe débit-distorsion $D(R)$ en "élargissant" l'intervalle de quantification autour de 0 appelé zone morte (*deadzone* en anglais). La définition des intervalles de quantification :

$$U_i = \begin{cases} [-(1-\delta)\Delta, (1-\delta)\Delta[& i = 0 \\ [(i-\delta)\Delta, (i+1-\delta)\Delta[& i > 0 \\ [(i-\delta)\Delta, (i-1+\delta)\Delta[& i < 0 \end{cases}$$

où δ détermine la largeur de l'intervalle de quantification U_0 .

En choisissant $\delta = 1/2$, on retrouve un quantificateur uniforme. Un autre cas intéressant consiste à prendre $\delta = 0$ en référence à la figure 2.3. Ce type de quantificateur permet notamment une quantification progressive par plan de bits.

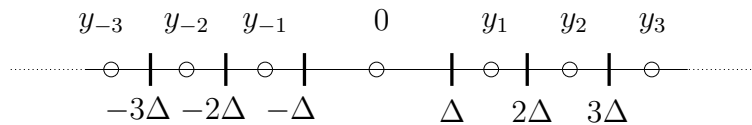


FIG. 2.3 – Quantificateur scalaire uniforme à zone morte ($\delta = 0$).

2.4.3 Quantification scalaire progressive

Certains systèmes de compression permettent de reconstruire de plus en plus fidèlement les données à mesure que la chaîne binaire est décodée. Il s'agit de fournir une approximation grossière des données reconstruites après avoir décodé seulement une petite partie du fichier compressé. En augmentant la lecture et le décodage de la chaîne

²extension de la définition de l'entropie pour les variables aléatoires continues.

binaire, la qualité des données reconstruites s'améliore progressivement jusqu'à ce que tout le fichier ait été décodé. La quantification associée à ce type de codeur est appelée en anglais *Embedded Quantization*.

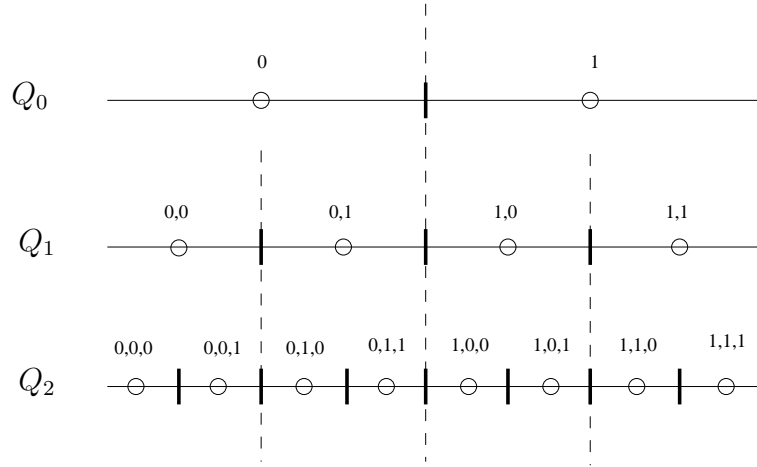


FIG. 2.4 – Quantificateur scalaire progressif Q_0, Q_1, Q_2 à 1,2 et 3 bits/symbole

Dans le principe, ce type de quantification consiste à imbriquer les intervalles de quantification de plusieurs quantificateur scalaires $\{Q_0, \dots, Q_k, \dots, Q_{K-1}\}$ comme le montre la figure 2.4. Dans le cas binaire, le passage d'un niveau k au niveau de quantification suivant $k + 1$ s'effectue en comparant la valeur de la donnée à quantifier avec un seuil T_k situé dans l'intervalle de quantification précédent. Les indices de quantification une fois concaténés peuvent être interprétés comme étant la représentation binaire d'entiers naturels.

Codage par plan de bits

En supposant, $y = Q(x)$ et un quantificateur uniforme à zone morte de paramètre $\delta = 0$, l'indice de quantification est donné par :

$$i = \text{sign}(x) \left\lfloor \frac{|x|}{\Delta} \right\rfloor \quad (2.33)$$

En reprenant la représentation binaire sur K bits, i peut se décomposer sous la forme :

$$i = Q_{K-1}(x) = s, b_0, b_1, b_2, \dots, b_k, \dots, b_{K-1} \quad (2.34)$$

où s correspond au signe de i et b_k le k -ième bit associé à la représentation binaire de i .

Tronquer la séquence binaire de l'équation (2.34) en éliminant les p derniers bits, nous permet d'introduire l'index $i^{(p)}$ tel que :

$$i^{(p)} = s, b_0, b_1, \dots, b_{K-1-p} \quad (2.35)$$

On peut alors vérifier que l'index $i^{(p)}$ est issu d'un quantificateur scalaire progressif à $K - p$ étages. Chaque quantificateur est uniforme à zone morte de paramètre $\delta = 0$ et de pas de quantification $2^{(p)}\Delta$. On obtient ainsi :

$$i^{(p)} = Q_{K-1-p}(x) \quad (2.36)$$

Chaque étage de quantification k consiste donc à déterminer la décomposition binaire de l'index i . Le premier étage Q_0 est le plus important car il permet de déterminer le bit de poids fort b_0 (en anglais *Most Significant Bit*). Les bits issus des étages de quantification b_1, \dots, b_{K-1-p} suivant permettent d'améliorer progressivement la qualité de reconstruction (en anglais *refinement bits*).

Les plans de bits sont introduits en représentant les images sous forme d'un ensemble d'images binaires correspondant chacun à la décomposition sous forme binaire des valeurs de pixels. Généralement, le plan de bits le plus représentatif (c'est à dire l'ensemble des bits de poids forts des pixels) est transmis en priorité avec le signe correspondant. Les plans de bits suivants sont ensuite transmis de façon à "raffiner" la quantification.

Ce type de représentation s'avère très utile pour les images de télédétection (en particulier RSO) à cause de leur grande taille. Les premiers plans de bits permettent de visualiser grossièrement le contenu de l'image pour un faible débit. L'utilisateur peut alors choisir de poursuivre la lecture des plans de bits sur une partie de l'image de manière à accroître progressivement la qualité de la zone décodée.

2.4.4 Allocation de bits

On se place dans le cadre où l'on dispose d'un nombre de bits à répartir sur un ensemble de sources (sous-bandes d'une image par exemple) et l'on recherche la meilleure répartition du débit au sens du minimum de la distorsion globale.

Supposons que l'on dispose d'un ensemble de sources X_1, X_2, \dots, X_k de moyenne nulle et de variance $\sigma_i^2 = E(X_i^2)$ $i = 1 \dots k$. Si l'on considère la densité de probabilité f_{X_i} connue, on peut concevoir un quantificateur optimal pour chaque source X_i pour n'importe quel nombre de niveaux de quantifications L_i . Sans codage entropique, il est souvent souhaitable d'avoir $L_i = 2^{b_i}$ pour pouvoir coder X_i sur $b_i = \log_2(L_i)$ bits.

Soit $D(b_i)$ la distorsion (au sens de l'erreur quadratique moyenne) obtenue en quantifiant de façon optimale la source X_i sur b_i bits. La distorsion totale s'écrit :

$$D = D(b) = \sum_{i=1}^k D(b_i) \quad b = (b_1, \dots, b_k) \quad (2.37)$$

Le problème d'allocation peut se formuler de la manière suivante : trouver les b_i $i = 1 \dots k$ qui minimisent $D(b)$ sous la contrainte $\sum_{i=1}^k b_i \leq B$ où B correspond au nombre de bits disponibles (budget).

Il n'est pas évident de trouver un algorithme exact qui puisse résoudre ce problème dans le cas général. Une recherche exhaustive sur les combinaisons possibles b pourrait être envisagée (avec b_i entiers et satisfaisant $\sum_{i=1}^k b_i \leq B$) mais le nombre de calculs rendrait cette méthode inefficace. En revanche, en faisant l'hypothèse de haute résolution, on peut obtenir une solution exacte et optimale (mais avec b_i non forcément entiers).

Dans ce cas, les performances d'un quantificateur scalaire optimal à haute résolution permettent d'approcher la distorsion suivant une expression de la forme $D(b_i) \simeq h_i \sigma_i^2 2^{-2b_i}$ avec h_i une constante dépendant de la densité de probabilité $f_{X_i}(x)$ de la variable normalisée $\frac{X_i}{\sigma_i}$.

Encore une fois, la technique des multiplieurs de Lagrange permet de trouver l'ensemble des b_i tel que $D = \sum_i h_i \sigma_i^2 2^{-2b_i}$ soit minimale sous la contrainte $\sum_i b_i = B$. Le problème se reformule de la manière suivante :

$$b_i = \arg \min_{\{b_i\}} J(\lambda) = \sum_{i=1}^k D(b_i) + \lambda \left(\sum_{i=1}^k b_i - B \right) \quad i = 1..k$$

En dérivant l'expression précédente par rapport à b_i , on obtient la relation :

$$\frac{dD(b_i)}{db_i} = -\lambda \quad i = 1..k$$

L'optimum est obtenu lorsque la dérivée des fonctions $D(b_i)$ associées à un quantificateur est la même pour tous les quantificateurs. La pente optimale λ (en anglais *distortion slope*) doit être déterminée de manière à satisfaire la contrainte $\sum_i b_i = B$ (en anglais *equal-slope method*).

L'allocation optimale est alors donnée par :

$$b_i = \bar{b} + \frac{1}{2} \log_2 \frac{\sigma_i^2}{\rho^2} + \frac{1}{2} \log_2 \frac{h_i}{H} \quad (2.38)$$

avec $\bar{b} = B/k$, le nombre moyen de bits par source et ρ^2 la moyenne géométrique des σ_i^2 et H la moyenne géométrique des coefficients h_i :

$$\rho^2 = \prod_i^k (\sigma_i^2)^{1/k}$$

$$H = \prod_i^k (h_i)^{1/k}$$

L'expression (2.38) montre ainsi qu'il faut allouer plus de bits aux sources de plus grande variance.

La distorsion totale vaut alors :

$$D = kH\rho^2 2^{-2\bar{b}}$$

Chaque quantificateur apporte la même contribution à la distorsion globale D . Cette contribution s'exprime par :

$$D(b_i) = H\rho^2 2^{-2\bar{b}}$$

Ce résultat est équivalent à celui qu'on obtiendrait pour un ensemble de k sources de même variance ρ^2 , de même distribution de constante H , quantifiées chacune sur \bar{b} bits.

2.4.5 Quantification vectorielle

Le principe est de grouper plusieurs échantillons dans un vecteur et de quantifier l'ensemble. Cette technique permet ainsi de prendre directement en compte la corrélation contenue dans le signal. Soit un vecteur d'entrée $x = (u_1, u_2, \dots, u_N)$ de dimension N appartenant à l'ensemble $U = \{x \in \mathbb{R}^N\}$.

Ce vecteur sera représenté par un vecteur $y_i = \{v_1, v_2, \dots, v_N\}$ appartenant à un autre ensemble $Y = \{y_i \in \mathbb{R}^N, i = 1 \dots L\}$.

L'ensemble des vecteurs de reconstruction $Y = \{y_1, \dots, y_L\} \subset \mathbb{R}^N$ constitue le dictionnaire des représentants.

On appelle quantificateur vectoriel de dimension N et de taille L une application Q :

$$\begin{aligned} U &\rightarrow Y = \{y_1, y_2, \dots, y_L\} \subset \mathbb{R}^N \\ x &\rightarrow Q(x) = y_i \end{aligned} \quad (2.39)$$

L'espace \mathbb{R}^N est alors partitionné en L classes :

$$U_i = \{x / Q(x) = y_i\} \quad i = 1, 2, \dots, L \quad (2.40)$$

Le quantificateur vectoriel prend en entrée le vecteur $x \in \mathbb{R}^N$ comprenant par exemple un groupe de pixels. Ce vecteur est quantifié et encodé par un index binaire i associé au vecteur de reconstruction $y_i \in \mathbb{R}^N$. Tout comme la quantification scalaire, les index i peuvent être transmis par un code à longueur fixe. Le débit binaire en bit par échantillon (c'est à dire une composante du vecteur) est donné par :

$$b = \frac{1}{N} \lceil \log_2 L \rceil \quad (2.41)$$

L'image compressée étant représentée par les indices i , l'opération de décodage est donc complètement décrite une fois que le dictionnaire Y a été spécifié. L'expression du débit (formule 2.41) montre que contrairement à la quantification scalaire qui n'autorise que des débits d'au moins 1 bit par échantillon, la quantification vectorielle permet d'obtenir des débits inférieurs, donc fractionnaires et sans codage entropique.

Soit $d(x, y_j) \geq 0$ une mesure de distorsion entre le vecteur d'entrée x , réalisation d'un vecteur aléatoire X de distribution conjointe $f_X(x)$ et un vecteur de reproduction $y_i, i = 1 \dots L$. La distorsion moyenne est donnée par :

$$\begin{aligned} D &= E[d(X, Q(X))] = \int_{\mathbb{R}^N} d(x, Q(x)) f_X(x) dx \\ &= \sum_{i=1}^L \int_{U_i} d(x, y_i) f_X(x) dx \end{aligned}$$

Le quantificateur vectoriel optimal sélectionne le vecteur y_i associé à la région U_i tel que $d(x, y_i) \leq d(x, y_j) \forall j = 1 \dots L$. C'est la règle du plus proche voisin. La partition U_i ainsi obtenue est appelée cellule de Voronoï.

La théorie de l'information assure que la quantification vectorielle offre des performances meilleures en terme de compromis débit-distorsion que la quantification scalaire.

Considérant ainsi les vecteurs dans un espace de dimension N , les vecteurs de reproduction peuvent être adaptés en s'affranchissant du partitionnement rectangulaire inhérent aux quantificateurs scalaires. En particulier, on démontre que pour N tendant vers l'infini, la fonction de débit-distorsion quadratique d'un quantificateur vectoriel optimal s'approche de la limite théorique de la fonction débit-distorsion de Shannon.

En revanche contrairement au cas scalaire, il est presque impossible de disposer d'un modèle analytique de la distorsion, ne connaissant pas en général la probabilité conjointe de la source. C'est la raison pour laquelle la construction du dictionnaire est généralement abordée à partir d'une séquence d'apprentissage représentative des données susceptibles d'être quantifiées par la suite. Par ailleurs, cela rend la construction du dictionnaire et/ou l'encodage beaucoup plus complexe que dans le cas scalaire, la charge de calcul augmentant sensiblement avec la dimension.

L'algorithme de Lloyd généralisé

L'algorithme GLA (*Generalized Lloyd Algorithm*) a été largement étudié et généralisé à l'espace de dimension N l'algorithme de Lloyd présenté à la section 1.4.1. Cet algorithme est semblable à l'algorithme de classification des K-moyennes [55](ou *K-means*). Sont appliquées de manière itérative :

1. La condition du plus proche voisin : étant donné un dictionnaire $Y^{(t)}$, il s'agit de classer chaque vecteur d'entrée d'une suite $\{x_n\}$ autour du vecteur y_i , choisi comme étant le plus proche au sens de l'erreur quadratique moyenne.

$$d(x_n, y_i) \leq d(x_n, y_j) \quad \forall j = 1 \dots L \quad (2.42)$$

avec $d(x_n, y_i) = (x_n - y_i)^T (x_n - y_i)$.

2. La condition du centroïde appliquée sur les cellules de Voronoï U_i pour déterminer les représentants du nouveau dictionnaire $Y^{(t+1)}$. Pour l'erreur quadratique moyenne, les centroïdes sont déterminés en calculant le vecteur moyen des vecteurs d'entrée appartenant à la partition U_i .

L'algorithme est réitéré avec le nouveau dictionnaire jusqu'à ce que la distorsion moyenne converge vers un minimum et ne varie plus de manière significative. Considérant que M_i vecteurs de la suite de vecteur d'entrée $\{x_n\}$ appartiennent à la cellule de Voronoï U_i , l'erreur quadratique est calculée suivant la formule :

$$D = \sum_{i=1}^L \sum_{k=1}^{M_i} (x_k - y_i)^T (x_k - y_i) \quad x_k \in U_i \quad (2.43)$$

Il est cependant important de noter que l'évolution du dictionnaire va dépendre du dictionnaire initial. Si le dictionnaire est mal choisi initialement, l'algorithme convergera vers un minimum local.

L'algorithme LBG (Linde, Buzo Gray) [53] est une variante de l'algorithme GLA qui permet de :

- de résoudre le problème du choix du dictionnaire initial.
- de construire des dictionnaires de taille 2^k qui se prêteront à une quantification vectorielle arborescente.
- d'accélérer la construction du dictionnaire mais aussi la quantification (si l'on utilise la structure arborescente des dictionnaires successifs).

L'algorithme est itératif :

1. Le dictionnaire initial est formé d'un seul vecteur $y_1^{(0)}$ comme étant le centroïde de toute la base.
2. On éclate le vecteur en deux par une perturbation aléatoire vectorielle de faible énergie $y_1^{(1)} = y_1^{(0)} + \epsilon$ et $y_2^{(1)} = y_1^{(0)} - \epsilon$.
3. On applique l'algorithme GLA sur ce nouveau dictionnaire.
4. Retour à 2) puis 3) jusqu'à ce que la taille du dictionnaire $L = 2^k$ soit atteinte.

L'algorithme LBG converge vers un minimum local mais dont on peut penser qu'il est peu éloigné du minimum global avec une construction proche du dictionnaire optimal à partir d'une base d'apprentissage.

Quantificateurs vectoriels structurés

Bien qu'offrant des performances supérieures à la quantification scalaire, l'algorithme LBG ou autres techniques basées sur les méthodes d'apprentissage, sont gourmandes en ressources calculatoires. La répartition irrégulière des vecteurs de reproduction n'offrent pas de structure simple exploitable par le quantificateur. Ces techniques de quantification nécessitent par conséquent une recherche exhaustive du mot de code le plus proche dans le dictionnaire. La taille du dictionnaire augmentant exponentiellement avec le débit et la dimension ($L = 2^{Nb}$), la complexité de calcul devient prohibitive sans compter la mémoire nécessaire au stockage du dictionnaire.

Certaines méthodes de quantification structurées ou avec contraintes permettent alors d'accélérer l'encodage sans pour autant toujours avoir la garantie d'optimalité sur le minimum de distorsion. Citons notamment :

"Tree-structured Vector Quantization"

Le principe consiste à sélectionner le mot de code par une séquence de décisions binaires basées sur le minimum de distorsion obtenues en comparant le vecteur d'entrée x aux vecteurs de reproduction disponibles à chaque noeud. Le mot de code est alors une séquence binaire représentant la séquence de décisions du noeud racine jusqu'au noeud terminal. L'arbre peut être "équilibré" avec des indices de même longueur b correspondant à un code à longueur fixe, ou au contraire "déséquilibré" avec des codes à longueur variable.

En comparaison d'une méthode à recherche exhaustive, la complexité de recherche d'un arbre "équilibré" est linéaire en fonction du débit au prix d'un coût en capacité de mémoire doublée. En ce qui concerne un arbre "déséquilibré", la complexité reste linéaire en fonction du débit moyen mais avec toujours en contrepartie un coût de mémoire beaucoup plus important [80].

"Multistage Vector Quantization"

L'idée est de réduire la complexité de recherche en utilisant des dictionnaires de petite taille. Cette technique divise ainsi l'étape d'encodage en cascasant des quantificateurs indépendants [45] [4]. La première étape consiste ainsi à quantifier "grossièrement" le vecteur x d'entrée à partir d'un dictionnaire à faible nombre de représentants. L'étape suivante opère une quantification sur le vecteur d'erreur (aussi appelé résidu) entre le vecteur original et le vecteur de reproduction sélectionné à la sortie du premier quantificateur. Les étapes suivantes permettent d'obtenir une quantification de plus en plus fine en requantifiant les résidus.

Application aux images RSO :

Dans le cadre de la compression d'images d'amplitude RSO, M. Venkatraman [106] a développé une méthode de compression où l'image est d'abord segmentée en objets à l'aide d'un algorithme de détection automatique de cibles. L'information de fond a ensuite été encodée à l'aide d'une quantification vectorielle multi-étages. Celle-ci permet alors de restituer les textures avec un débit variable dépendant du nombre d'étages dans le processus de quantification des vecteurs d'erreurs résiduels. La quantification itérative des vecteurs d'erreurs résiduels est réalisée sur des blocs de taille différente à chaque itération. A la manière d'un quad-tree cette technique permet de s'adapter localement au contenu du signal (textures, points brillants) et de préserver la structure de "fond" de l'image. Le nombre d'étages de quantification est alors déterminé en fonction de la variation relative de la distorsion lors du passage d'un nombre d'étages N à $N + 1$.

Les dictionnaires sont construits à partir de l'algorithme de Lloyd généralisé à contrainte entropique appliqué à une séquence d'apprentissage. Les résultats montrent des performances en terme d'erreur quadratique moyenne équivalente aux résultats obtenus par des techniques de transformée en ondelettes.

"Trellis coded vector quantization"

Il s'agit de la technique analogue à la modulation codée introduite par Ungerboeck [101] pour le codage canal (contrôle d'erreur) mais adaptée à la problématique du codage source. La quantification codée en trellis peut être vue comme le fait de disposer d'un super-dictionnaire partitionné en un ensemble de sous-dictionnaires. L'encodeur dispose alors d'un ensemble d'états et de transitions autorisées et non autorisées entre ces états. Chacune de ces transitions correspond à l'un des sous-dictionnaires. L'encodeur transmet ainsi dans un premier temps au décodeur, l'index correspondant à la transition d'état choisie. L'index du mot de code appartenant au sous-dictionnaire sélectionné complète le mot de code [58] [33].

Application aux images RSO :

L'étude menée par Eckstein [20] a montré que la quantification en trellis des coefficients d'ondelettes d'images d'amplitude (voir section suivante) a donné de bien meilleurs résultats que des méthodes basées sur la norme standard JPEG en terme de qualité de speckle, de remontée des lobes secondaires autour des cibles et de lissage introduit par la compression.

Owens [73] a également appliqué cette technique sur les données brutes considérées comme une source de nature gaussienne. En terme de rapport signal sur bruit, cette méthode donne de meilleurs résultats que l'algorithme BAQ [47] (*Bloc Adaptive Quantization*) basé sur une quantification scalaire de type Lloyd-Max.

"Lattice vector quantization (LVQ)"

Les dictionnaires optimaux ne présentent pas en général de structure simple, ce qui rend la quantification vectorielle gourmande en temps de calcul. A l'opposé, la quantification vectorielle par lattices, consiste à prendre un dictionnaire constitué de points régulièrement répartis dans l'espace. On définit ainsi un réseau régulier de points (ou *lattice* en anglais) $\Lambda_n \subset \mathbb{R}^n$ composé de l'ensemble des combinaisons linéaires d'un ensemble de vecteurs linéairement indépendants et qui engendrent l'espace. En dimension 1, le réseau de points correspond à un quantificateur uniforme. En dimension n , l'arrangement régulier des points dans le plan est donné par :

$$L = \{y \in \mathbb{R}^n : y = A.m \ \forall m = (m_1, m_2, \dots, m_n) \in \mathbb{Z}^n\}$$

où A est une matrice $n \times n$ non singulière appelée matrice génératrice et m un vecteur d'entiers.

Convay et Sloane [2] [14] ont déterminé les meilleures lattices pour plusieurs dimensions associées à des algorithmes rapides de codage et décodage exploitant la structure régulière du réseau. Il n'est alors plus nécessaire d'effectuer de recherche exhaustive du meilleur représentant.

Quelques réseaux importants sont les *roots lattices* A_n ($n \geq 1$), D_n ($n \geq 2$) et E_n ($n = 6, 7, 8$), "The Barnes-Wall lattice" Λ_{16} en dimension 16 et "The leech lattice" Λ_{24} en dimension 24. Ces réseaux correspondent au meilleur empilement des sphères dans un espace de dimension de n , c'est à dire au réseau le plus dense. Par exemple, pour $n = 2$, le réseau optimal A_2 est donné par un arrangement hexagonal régulier des points dans le plan.

L'aspect critique de cette quantification concerne la troncature du réseau de points (englobant ainsi un nombre fini de points pour un débit donné) et le facteur d'échelle appliqué au réseau de points (de manière à s'adapter à la nature statistique des données d'entrée). La région tronquée est la région de support, c'est à dire un sous-ensemble du réseau qui sera effectivement encodé. Il convient donc de trouver le compromis entre distorsion granulaire (obtenue à l'intérieur du réseau tronqué) et de saturation (distorsion obtenue à l'extérieur du réseau). Le facteur d'échelle appliqué ou de manière équivalente à la source doit être adapté de manière à minimiser la distorsion globale.

Application aux images RSO :

Ce type de quantification a été en particulier utilisée pour la compression de données brutes. Citons notamment les travaux de D. Lebedeff [50] qui utilise une quantification vectorielle algébrique sur les parties réelle et imaginaire avant synthèse RSO. La source est modélisée par un vecteur circulaire gaussien. L'image est alors découpée en un ensemble de blocs et projetés sur une lattice en fonction de la variance du bloc.

Quantificateurs à contrainte entropique

La quantification vectorielle s'évertue à minimiser la distorsion moyenne sous la contrainte du nombre L de mots de code. Si aucun codage entropique n'est envisagé, le débit fixe du codeur est obtenu par la formule (2.41). Cependant, en considérant un codage entropique sur les index des vecteurs de reproduction, il paraît plus sensé de minimiser la distorsion moyenne sous la contrainte du *débit après codage entropique* que la contrainte du nombre de mot de codes. Citons par exemple, la modification de l'algorithme GLA par Chou et al [11] en introduisant la contrainte de codage entropique. La mesure d'erreur quadratique est modifiée en rajoutant un multiplicateur de Lagrange associé à la quantité d'information qu'apporte le représentant y_i de la cellule de Voronoï (c'est à dire le nombre de bits nécessaires pour transmettre le mot de code). La condition du plus proche voisin permettant de classer un vecteur d'entrée x_n comme appartenant à la cellule de Voronoï U_i est alors donnée par l'expression :

$$d(x_n, y_i) \leq d(x_n, y_j) \quad \forall j = 1 \dots L \quad (2.44)$$

avec $d(x_n, y_i) = (x_n - y_i)^T(x_n - y_i) - \lambda \cdot \log_2[\text{Prob}(y_i \in U_i)]$.

La quantification à contrainte entropique améliore le compromis débit-distorsion quand la quantification est suivie par un codage entropique des index de quantification. A partir des résultats obtenus sur la quantification scalaire à contrainte entropique, la généralisation vectorielle de ces résultats sous l'hypothèse de haute-résolution, suggère que l'optimum est obtenu par une généralisation du quantificateur uniforme au cas multidimensionnel [35] (en d'autres termes un réseau régulier de points).

2.5 Codage par Transformée

La plupart des algorithmes de compression avec pertes opèrent une transformation de l'image qui projète les données dans un espace plus propice à la quantification et au codage entropique. L'idée est qu'en effectuant une transformée linéaire sur l'image originale, on obtient un ensemble de coefficients dans l'espace transformé dont les composantes sont moins corrélées entre elles, plus compactes (c'est à dire que l'énergie du signal est concentrée sur un faible nombre de composantes au lieu d'être uniformément répartie). Les données transformées sont ensuite quantifiées et les index des représentants encodés entropiquement.

L'intérêt du codage par transformée réside entre autres dans le fait que la quantification des coefficients entraîne une perte d'information "répartie" sur les données initiales. Cela permet de se référer plus facilement aux outils perceptifs humains et donc à l'évaluation subjective de la qualité de reconstruction de l'image.

2.5.1 Transformée orthogonale

Supposons que l'on ait un bloc de N échantillons (vecteur) d'un processus aléatoire stationnaire $X = \{x_0, x_2, \dots, x_{N-1}\}^T$. D'après l'hypothèse de stationnarité, ces échantillons ont la même variance et peuvent présenter une corrélation importante. Cette corrélation entraîne une certaine redondance qui est conservée dans les échantillons quantifiés.

On introduit alors le vecteur $Y = \{y_0, y_2, \dots, y_{N-1}\}^T$ obtenu par transformée orthogonale linéaire \mathbf{T} de sorte de que $Y = \mathbf{T}X$. On quantifiera alors les composantes du vecteur transformé afin d'obtenir le vecteur $\hat{Y} = Q(Y)$. Au décodage, le vecteur \hat{X} sera reconstruit par transformation inverse du vecteur quantifié $\hat{X} = \mathbf{T}^{-1}\hat{Y}$.

La mesure de distorsion pour un codage par transformée est alors donnée par :

$$D = \sum_{i=0}^{N-1} E[(X_i - \hat{X}_i)^2] = E[(X - \hat{X})^T(X - \hat{X})]$$

Puisque $\hat{X} = \mathbf{T}^{-1}\hat{Y}$, l'erreur de quantification $X - \hat{X}$ dépend à la fois de l'erreur de quantification $Y - \hat{Y}$ et de la transformée \mathbf{T}^{-1} qui risque d'accentuer les erreurs (si le déterminant est supérieur à 1). On impose donc à la transformation \mathbf{T} , la condition d'orthogonalité $\mathbf{T}^{-1} = \mathbf{T}^T$ pour que la norme soit préservée par la transformation.

On obtient les propriétés suivantes :

- Norme conservée : $E[X^T X] = E[Y^T T T^T Y] = E[Y^T Y]$
- Distorsion conservée : $E[(X - \hat{X})^T(X - \hat{X})] = E[(Y - \hat{Y})^T(Y - \hat{Y})]$
- Matrice de covariance : $R_Y = E[YY^T] = TE[XX^T]T^T = TR_X T^T$

La transformation qui permet d'obtenir un ensemble de coefficients non-corrélés et qui minimise la distorsion est la transformation de Karhunen-Loeve (TKL). Cette transformation est formée par les vecteurs propres de la matrice de covariance R_X du signal X et est donc dépendante du signal traité.

D'autre part, il n'existe pas d'algorithme rapide pour le calcul de la TKL qui nécessite par ailleurs l'évaluation de la matrice d'auto-corrélation pour des signaux qui ne sont pas

stationnaires (en particulier pour les images RSO) et pour lesquels il faut actualiser régulièrement la matrice de covariance et donc la matrice de transformation. On recherchera le plus souvent une transformation "sous-optimale" du point de vue de la décorrélation et/ou de la concentration des coefficients.

Les transformées de Fourier discrètes (TFD) et en cosinus discret (DCT) sont souvent mise en oeuvre en pratique car il existe des implémentations par des algorithmes rapides (FFT : Fast Fourier Transform) qui ont l'avantage d'être indépendantes du signal d'entrée. Les coefficients sont décorréllés (la TFD permet d'approximer la TKL lorsque le support sur lequel elle est effectuée tend vers l'infini). L'énergie des coefficients est bien localisée en fréquence mais pas en temps (ou dans le domaine spatial pour les images).

D'autre part, la transformation est séparable : une DCT en deux dimensions se calcule comme deux DCT mono-dimensionnelles successives. Dans le cas 1D, le vecteur $Y = \{y_0 \dots y_{N-1}\}^T$ issu de la DCT appliquée au vecteur $X = \{x_0 \dots x_{N-1}\}^T$ s'exprime par :

$$y_0 = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x_k \quad \text{et} \quad y_p = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} x_k \cos\left(p\left(k + \frac{1}{2}\right)\frac{\pi}{N}\right) \quad p = 1 \dots N - 1$$

L'hypothèse de stationnarité du signal oblige à effectuer ces transformations non pas sur toute l'image mais sur des petits blocs de taille réduite non recouvrants (une DCT sur des blocs 8x8 est utilisée pour la norme du standard JPEG [75]). La conséquence de ce codage par bloc est que la quantification des coefficients issus des blocs transformés par DCT introduisent la présence d'effets de blocs gênants pour des quantifications à très bas-débit.

Le fait de traiter l'image en blocs indépendants est une limitation des méthodes de transformée par bloc. En effet, les échantillons des blocs "voisins" ne sont pas forcément décorréllés avec les autres blocs, bien au contraire. On peut donc s'attendre à ce que les coefficients transformés issus de blocs adjacents présentent une certaine corrélation.

Application aux images RSO :

L'algorithme de compression JPEG est basé sur une quantification scalaire en blocs de découpage 8x8 de l'image et présuppose une certaine stationnarité locale du signal. Outre l'effet du chatoiement qui induit un contenu en hautes fréquences bien plus important que pour les images naturelles, l'hypothèse de stationnarité est difficile à retenir pour les images RSO.

Plusieurs auteurs ont ainsi testé des méthodes de compression basées sur l'application de l'algorithme JPEG [31], [86], [17]. Les critères de qualité essentiellement retenus n'étant alors que l'erreur quadratique moyenne ou le rapport signal sur bruit (RSB), ces méthodes offrent des performances inférieures aux techniques basées sur la transformée en ondelettes (abordée à la section suivante) ou certaines quantifications vectorielles.

L'effet de bloc crée des structures artificielles particulières avec des zones fortement lissées dues à la quantification des coefficients DCT. Kuperman [46] montre ainsi une augmentation du taux de fausses alarmes d'algorithmes de détection de cibles-fond.

Concernant la compression d'images RSO complexe, P. Eichel et al [21] propose d'utiliser la transformée de Fourier Discrète sur l'image SLC. La forme particulière du module de l'image dans le domaine de Fourier permet d'identifier certaines zones où les coefficients ont une valeur négligeable ou proche de zéro par rapport à la bande centrale. L'approche

proposée par Eichel consiste alors à remplacer les coefficients de faible module ³ par des zéros (*zero padding*) et à quantifier par un quantificateur de type Lloyd-Max le reste des coefficients complexes considérés comme une source gaussienne circulaire i.i.d dans le plan complexe.

Il convient cependant de rester prudent face à ce type de traitement. En effet, le "zero-padding" introduit une distorsion non négligeable. Bien qu'en ne conservant intacts (i.e sans quantification) que 40% des coefficients, le rapport signal sur bruit ne dépassera pas 20-25 dB après zero-padding. La figure 2.5 montre ainsi le principe de cette méthode sur les coefficients de Fourier d'une image issue du capteur RAMSES de l'ONERA.

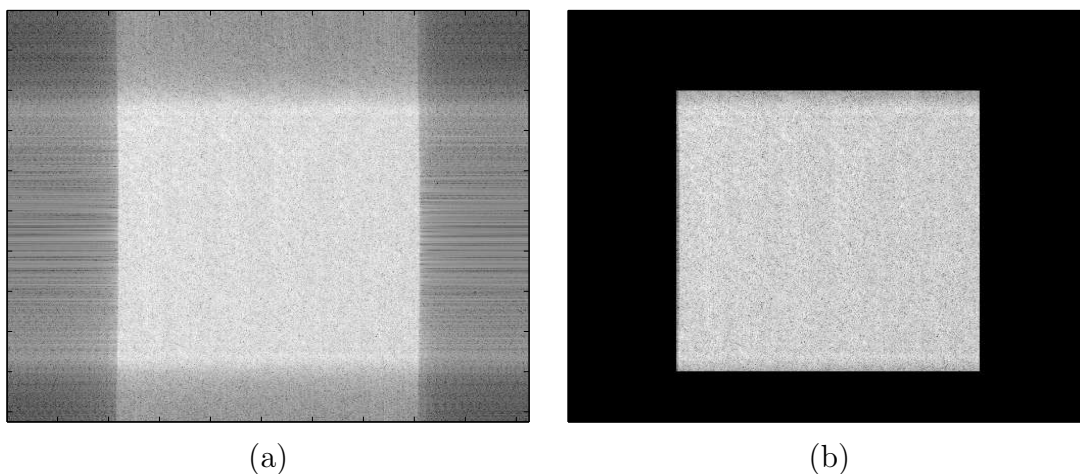


FIG. 2.5 – (a) : module des coefficients de Fourier d'une image SLC issue du capteur RAMSES (échelle logarithmique). (b) : les coefficients de bordure (hautes fréquences) ont été mis à zéro (*zero padding*). Sans quantification des coefficients complexes de la bande centrale, le rapport signal sur bruit est de 21.9 dB pour un taux de compression de $\eta = 2.5 : 1$.

³Cette information n'a pas de sens physique et correspond à du bruit.

2.5.2 Transformée en sous-bandes et transformée en ondelettes

Les transformations du type DCT ou FFT réalisées par blocs proviennent du désir de conserver une information de "localisation" qui semble mieux adaptée aux signaux non-stationnaires. C'est pour prendre en compte cet aspect local que la transformée par ondelettes a été introduite et a connu un succès grandissant pour les applications liées à la compression. Cette transformée réalise une analyse multi-résolution et permet une représentation adaptée aux signaux non-stationnaires en considérant que le signal peut être approché localement par des polynômes. Les travaux de Mallat ont permis de donner un écho particulier à la théorie mathématique de l'analyse par ondelettes qui trouve son équivalent dans l'approche multi-résolution par bancs de filtres.

Cette transformée sera largement utilisée au cours des méthodes proposées dans le cadre de nos travaux. C'est la raison pour laquelle, nous introduirons de manière plus détaillée les principes associés à ce type de transformée à travers la théorie des bancs de filtres.

Bancs de filtres à reconstruction parfaite

On considère la décomposition suivante où apparaît un banc de filtres à deux sous-bandes et à reconstruction parfaite.

H_0 et H_1 sont respectivement des filtres passe-bas et passe-haut permettant de donner une représentation dans deux domaines fréquentiels suivis d'une étape de décimation par un facteur 2. A partir du signal d'entrée $a^{(0)}$, l'étape d'analyse donne lieu à l'apparition de coefficients $a^{(1)}$ et $d^{(1)}$. En considérant la transformée en z définie par $H(z) = \sum_k h[k].z^{-k}$ du filtre de réponse impulsionnelle $\{h[k]\}_{k \in \mathbb{Z}}$, il est alors possible d'établir la relation entre les sorties du filtre d'analyse après décimation :

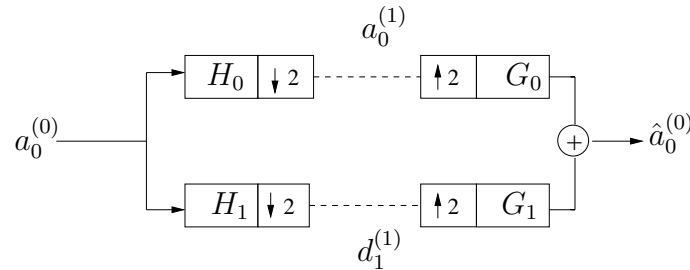


FIG. 2.6 – Banc de filtres d'analyse et de synthèse à 2 canaux et à reconstruction parfaite. H_0 et H_1 correspondent respectivement aux filtres passe-bas et passe-haut d'analyse suivis d'une décimation par 2 (on ne garde qu'un échantillon sur 2 en sortie). G_0 et G_1 correspondent respectivement aux filtres passe-bas et passe-haut de synthèse précédés d'une insertion d'un zéro entre 2 échantillons).

$$a^{(1)}(z^2) = \frac{1}{2}[H_0(z)a^{(0)}(z) + H_0(-z)a^{(0)}(-z)]$$

$$d^{(1)}(z^2) = \frac{1}{2}[H_1(z)a^{(0)}(z) + H_1(-z)a^{(0)}(-z)]$$

La sortie du banc de filtres s'exprime en fonction de l'entrée par :

$$\hat{a}^{(0)}(z^2) = \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)]a^{(0)}(z) + \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)]a^{(0)}(-z)$$

Le second terme correspond au terme de repliement de spectre (aliasing) qu'il convient d'éliminer. Cette condition se traduit par :

$$\frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)] = 0$$

Afin d'obtenir une reconstruction parfaite, il est nécessaire d'avoir :

$$\frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)] = z^{-k} \quad (2.45)$$

si l'on tolère un retard pur z^{-k} .

Pour annuler les recouvrements, une solution des filtres miroirs en quadrature [16] consiste à choisir :

$$\begin{aligned} H_0(z) &= G_0(z) = f(z) \\ H_1(z) &= -G_1(z) = f(-z) \end{aligned}$$

En remplaçant les expressions des filtres ci-dessus dans l'équation (2.45), on obtient alors :

$$H_0(z)^2 - H_0(-z)^2 = 2.z^{-k}$$

Il reste alors un degré de liberté dans le choix de H_0 . Malheureusement, il n'existe pas d'implémentation sous forme de filtre à réponse impulsionnelle finie (FIR) qui satisfasse cette condition (mis à part le cas trivial du filtre de Haar avec $h_0[0] = h_0[1] = 1/\sqrt{2}$).

Une autre solution permettant d'obtenir un système de banc de filtres à reconstruction parfaite est donnée par :

$$\begin{aligned} H_0(z) &= G_0(z^{-1}) = f(z) \\ H_1(z) &= G_1(z^{-1}) = zf(-z^{-1}) \end{aligned}$$

De manière équivalente la réponse impulsionnelle satisfait :

$$\begin{aligned} h_0[k] &= g_0[-k] \\ h_1[k] &= g_1[-k] = (-1)^{k+1}h_0[-(k+1)] \end{aligned} \quad (2.46)$$

En remplaçant ces expressions dans l'équation (2.45), le terme d'aliasing est également dans ce cas annulé. Il reste alors la relation suivante traduisant la reconstruction parfaite sans retard de groupe :

$$H_0(z)H_0(z^{-1}) + H_0(-z)H_0(-z^{-1}) = 2$$

Dans le domaine temporel, la relation précédente s'exprime en fonction des coefficients issus de la réponse impulsionnelle du filtre $h_0[k]$:

$$\sum_k h_0[k]h_0[k-n](1 + (-1)^n) = 2\delta[n] \quad (2.47)$$

Cette relation illustre la condition d'orthonormalité entre la séquence $h_0[k]$ et la même séquence translatée par des multiples de 2 : $h_0[k-2n]$.

$$\sum_k h_0[k]h_0[k-2n] = 0 \quad n \neq 0$$

Analyse multirésolution

A partir du banc de filtres de la figure 2.6, il est possible de cascader le même schéma de décomposition sur les sorties du filtre passe-bas : $a^{(1)}, a^{(2)}, \dots$. On introduit ainsi respectivement a^j et d^j avec $j \geq 1$ comme les coefficients d'approximation $a^{(j)}$ et de détails $d_1^{(j)}$ entre les niveaux $j - 1$ et j .

Les coefficients d'approximation successifs sont des versions lissées et sous-échantillonnées de a_0 . A chaque niveau de décomposition, le signal est caractérisé par la somme de l'approximation et des détails aux résolutions précédentes.

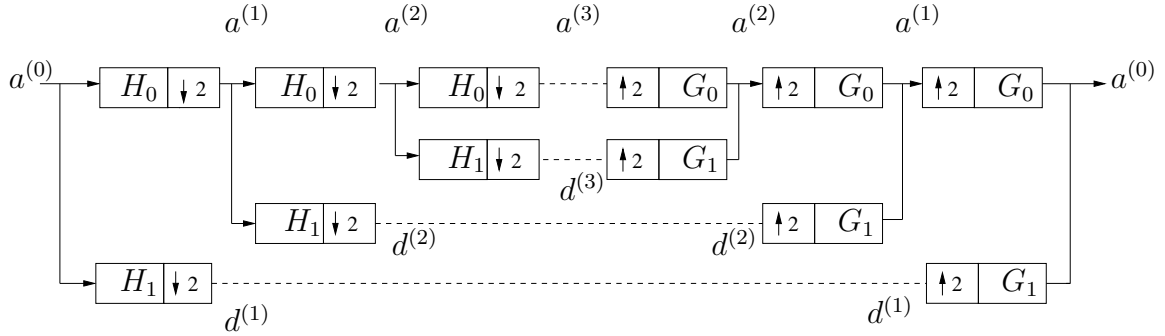


FIG. 2.7 – Analyse et synthèse multirésolution implémentée par banc de filtres

On définit ainsi l'analyse multirésolution de $L^2(\mathbb{R})$ comme un ensemble de sous-espaces encapsulés V_j permettant d'écrire une fonction f de $L^2(\mathbb{R})$ comme une limite d'approximations successives. La résolution augmente quand l'index j décroît.

$$\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_2 \subset \dots \quad (2.48)$$

Ces sous-espaces satisfont aux propriétés suivantes :

1. $\bigcap_{j \in \mathbb{Z}} V_j = \{0\}$ et $\bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R})$
2. $f(t) \in V_0 \Leftrightarrow f(2^{-j}t) \in V_j$. Une version dilatée du signal dans l'espace V_0 par un facteur 2^j engendre un signal appartenant à l'espace de plus faible résolution V_j .
3. $x(t) \in V_0 \Leftrightarrow x(t - n) \in V_0 \quad n \in \mathbb{Z}$. En d'autres termes, la résolution d'un signal n'est pas affectée par translation.
4. Il existe une base orthonormée $\{\varphi_n\}_{n \in \mathbb{Z}}$ tel que $\varphi_n(t) = \varphi(t - n) \in V_0$. La fonction $\varphi(t)$ est appelée fonction d'échelle.

Une famille de sous-espaces satisfaisant à ces conditions permet d'obtenir une analyse multi-échelles. Le paramètre j est assimilé à un paramètre d'échelle puisqu'il décroît à mesure que la résolution du signal augmente.

Fonction d'échelle

Les propriétés 3) et 4) permettent ainsi de définir un ensemble de fonctions $\{\varphi_n^j\}_{n \in \mathbb{Z}}$ formant une base orthonormée de V_j à partir d'une simple et unique fonction d'échelle $\varphi(t)$ et de ses translatées entières.

$$\varphi_n^j(t) = 2^{-j/2} \varphi_n(2^{-j}t - n)$$

La propriété d'orthonormalité qui s'exprime sous la forme :

$$\langle \varphi_n^j, \varphi_p^j \rangle = \int \varphi_n^j(t) \varphi_p^j(t) dt = \delta_{np} [0] \quad \forall n, p \in \mathbb{Z} \quad (2.49)$$

Puisque $V_0 \subset V_{-1}$, tous les espaces V_j sont engendrés par une fonction φ de $L^2(\mathbb{R})$. En conséquence, $\varphi(t)$ s'exprime comme une combinaison linéaire des fonctions $\varphi_n^{-1}(t) = \sqrt{2}\varphi(2t - n)$. On définit ainsi l'équation de dilatation permettant de relier la fonction φ d'un espace de résolution à un autre.

$$\varphi(t) = \sqrt{2} \sum_n g_0[n] \varphi(2t - n) \quad (2.50)$$

où $\{g_0\} \in \mathbb{R}$.

Il est important de remarquer que nous ne disposons pas d'expression explicite de la fonction φ . En revanche, il existe un lien entre l'analyse multirésolution par transformée en ondelettes et la théorie des bancs de filtres. Une première analogie apparaît sur la condition d'orthonormalité des fonctions d'échelle. A partir des expressions (2.49) et (2.50), on montre ainsi facilement que :

$$\langle \varphi_0^0, \varphi_n^0 \rangle = \delta[n] \Rightarrow \sum_k g_0[k] g_0[k - 2n] = \delta[n]$$

On en déduit que la séquence $\{g_0[n]\}$ translaturée d'un facteur multiple de 2 est orthogonale à elle-même. On retrouve ainsi la condition de l'équation (2.47) caractérisant une propriété du filtre passe-bas d'un banc de filtres à reconstruction parfaite.

Fonction d'ondelettes

Le passage d'un sous-espace V_j à un sous-espace de résolution inférieure V_{j-1} permet de définir l'espace orthogonal complémentaire W_j à V_j dans V_{j-1} :

$$W_j \oplus V_j = V_{j-1} \quad \text{et} \quad W_j \perp V_j$$

L'espace complémentaire W_j contient la différence d'information entre une approximation de f à la résolution j et une approximation de f à la résolution $j - 1$. L'analyse par fonction d'échelle $\varphi_n^j(t) \in V_j$ étant implicitement reliée au filtre passe-bas de la séquence g_0 dans le banc de filtre multirésolution de la figure (2.7), on peut alors définir la séquence du filtre passe-haut complémentaire $\{g_1[k]\}$. D'après l'équation (2.46), on peut fixer $g_1[k] = (-1)^{k+1} g_0[-(k + 1)]$.

De manière analogue à la fonction d'échelle, on introduit la fonction d'ondelettes ψ définie par l'équation de dilatation :

$$\psi(t) = \sqrt{2} \sum_n g_1[n] \varphi(2t - n) \quad (2.51)$$

Il est alors aisé de montrer que la fonction $\psi(t)$ satisfait aux propriétés suivantes :

1. $\langle \psi_0, \psi_n \rangle = \sum_k g_1[k]g_1[k-2n] = \delta[n]$. En d'autres termes, $\psi(t)$ et ses versions translatées d'un nombre entier sont orthonormées.
2. $\langle \psi_n, \varphi_p \rangle = \sum_k g_1[k-2n]g_0[k-2p] = 0$. On peut voir que la fonction $\psi(t)$ et ses translatées par nombre entier sont orthogonales au sous-espace V_0 . En prenant en compte le fait que $\psi(t)$ et ses translatées sont des combinaisons linéaires des fonctions de bases $\varphi_n^{-1} \in V_{-1}$, on peut en déduire naturellement que la fonction $\psi(t)$ appartient au complément de V_0 dans V_{-1} , c'est à dire que $\psi(t) \in W_0$.
3. Les propriétés d'orthogonalité mutuelle entre les fonctions $\{\psi_n\}_{n \in \mathbb{Z}}$ de W_0 et $\{\varphi_n\}_{n \in \mathbb{Z}}$ de V_0 permettent d'engendrer l'espace V_{-1} . En d'autre termes, pour $x \in V_{-1}$:

$$\begin{aligned} x &= \sum_{n \in \mathbb{Z}} a^{(-1)}[n] \varphi_n^{-1} \\ &= \sum_{n \in \mathbb{Z}} a^{(0)}[n] \varphi_n^0 + \sum_{n \in \mathbb{Z}} d^{(0)}[n] \psi_n^0 \end{aligned} \tag{2.52}$$

En conclusion, $\{\psi_n\}_{n \in \mathbb{Z}}$ est une base orthonormale qui engendre l'espace W_0 . Cette propriété s'étend également aux versions dilatées de $\psi(t)$ de sorte que $\{\psi_n^j\}_{n \in \mathbb{Z}}$ soit une base orthonormale de l'espace W_j complémentaire à V_j .

Si l'on considère l'analyse multirésolution dans laquelle la cascade de filtres s'étend à l'infini, le signal est entièrement caractérisé par les coefficients de détails $d^{(j)}$, c'est-à-dire par les projections de f sur W_j . Ainsi, $\{\psi_n^j\}_{n,j \in \mathbb{Z}}$ forme une base orthonormée d'ondelettes de $L^2(\mathbb{R})$ avec la propriété :

$$\bigoplus W_j = L^2(\mathbb{R})$$

Les fonctions d'échelle et d'ondelettes sont décrites implicitement par les coefficients de l'équation de dilatation. Dans le cadre de l'analyse multirésolution, il existe un parallèle entre l'analyse par ondelettes et celle par banc de filtres à reconstruction parfaite. Ceci implique qu'il n'est pas possible de choisir une forme d'onde quelconque pour en faire une fonction d'ondelettes. La reconstruction parfaite implique des conditions et impose de ce fait une contrainte sur les coefficients des filtres G_0 et G_1 .

Transformée en ondelettes discrètes (DWT)

L'analyse multirésolution et les transformées en sous-bandes forment le coeur de l'analyse par transformée en ondelettes.

- On identifie alors la séquence $\{g_0[n]\}$ de l'équation de dilatation (2.50) associée à la fonction d'échelle comme les coefficients du filtre de synthèse (passe-bas) d'un banc de filtres à 2 canaux.
- Le filtre de synthèse passe-haut est associé à la séquence $\{g_1[n]\}$ de l'équation de dilatation (2.51).

Les travaux de Mallat [56] ont permis de montrer que le banc de filtres à reconstruction parfaite constitue le support de l'implémentation de la transformée en ondelettes discrètes. En supposant que le signal d'entrée $x(t)$ est caractérisé dans l'espace de résolution V_0 , il s'exprime comme une combinaison linéaire des fonctions de la base orthonormée $\{\varphi_n\}_{n \in \mathbb{Z}}$.

$$x^{(0)}(t) = \sum_{n \in \mathbb{Z}} a_0^{(0)}[n] \varphi(t - n) \quad (2.53)$$

avec $a^{(0)}[n] = \langle x^{(0)}, \varphi_n^{(0)} \rangle$.

La séquence de coefficients $a^{(0)}[n]$ peut se redécomposer en 2 séquences $a^{(1)}[n]$ et $d^{(1)}[n]$ issues de la sortie du banc de filtres d'analyse et formant respectivement les coefficients de deux sous-bandes basses et hautes fréquences.

$$\begin{aligned} a^{(1)}[n] &= \sum_{k \in \mathbb{Z}} h_0[k - 2n] a^{(0)}[k] \\ d^{(1)}[n] &= \sum_{k \in \mathbb{Z}} h_1[k - 2n] a^{(0)}[k] \end{aligned} \quad (2.54)$$

La synthèse qui permet de reconstruire $a^{(0)}[n]$ à partir des coefficients des sous-bandes $a^{(1)}[n]$ et $d^{(1)}[n]$ s'écrit après interpolation et convolution :

$$a^{(0)}[n] = \sum_{k \in \mathbb{Z}} a^{(1)}[k] g_0[n - 2k] + \sum_{l \in \mathbb{Z}} d^{(1)}[j] g_1[n - 2j] \quad (2.55)$$

En remplaçant cette expression dans l'équation (2.53), on obtient :

$$\begin{aligned} x^{(0)}(t) &= \sum_{n \in \mathbb{Z}} \left(\sum_{k \in \mathbb{Z}} a^{(1)}[k] g_0[n - 2k] + \sum_{j \in \mathbb{Z}} d^{(1)}[j] g_1[n - 2j] \right) \varphi(t - n) \\ &= \sum_{k \in \mathbb{Z}} a^{(1)}[k] \left(\sum_{n \in \mathbb{Z}} g_0[n] \varphi((t - 2k) - n) \right) + \sum_{j \in \mathbb{Z}} d^{(1)}[j] \left(\sum_{n \in \mathbb{Z}} g_1[n] \psi((t - 2j) - n) \right) \\ &= \underbrace{\sum_{k \in \mathbb{Z}} a^{(1)}[k] \varphi_n^{(1)}(t)}_{x^{(1)}(t) \in V_1} + \underbrace{\sum_{j \in \mathbb{Z}} d^{(1)}[j] \psi_n^{(1)}(t)}_{w^{(1)}(t) \in W_1} \end{aligned}$$

avec $a^{(1)}[n] = \langle x^{(0)}, \varphi_n^{(1)} \rangle$ et $d^{(1)}[n] = \langle x^{(0)}, \psi_n^{(1)} \rangle$.

La transformée en sous-bandes permet de décomposer $x^{(0)}(t)$ en une somme de deux signaux $x^{(1)}(t) \in V_1$ et $w^{(1)}(t) = x^{(0)}(t) - x^{(1)}(t) \in W_1$ dans les deux espaces de résolution inférieure. En appliquant ainsi récursivement la même décomposition par banc de filtres au niveau des coefficients d'approximation $a_0^{(1)}[n]$, $a_0^{(2)}[n]$.., on réalise une analyse multi-résolution sur L niveaux. Le signal $x^{(0)} \in V_0$ est représenté par son approximation dans l'espace V_L et par la somme des coefficients d'ondelettes manquants pour la reconstruction parfaite : $V_0 = V_L \oplus_{j=1}^L W_j$.

$$x^{(0)} = \sum_{n \in \mathbb{Z}} a^{(L)}[n] \varphi_n^{(L)} + \sum_{l=1}^L \sum_{n \in \mathbb{Z}} d^{(l)} \psi_n^{(l)}$$

$$x^{(0)} = \sum_{n \in \mathbb{Z}} \langle x^{(0)}, \varphi_n^{(L)} \rangle \varphi_n^{(L)} + \sum_{l=1}^L \sum_{n \in \mathbb{Z}} \langle x^{(0)}, \psi_n^{(l)} \rangle \psi_n^{(l)}$$

En généralisant les équations (2.54) et (2.55), on obtient les équations de filtrage inhérents au banc de filtres d'analyse et de synthèse de la figure 2.7 :

Analyse : passage de l'échelle $l - 1$ à l

$$\begin{cases} a^{(l)}[n] = \sum_{k \in \mathbb{Z}} h_0[k - 2n] a^{(l-1)}[k] \\ d^{(l)}[n] = \sum_{k \in \mathbb{Z}} h_1[k - 2n] a^{(l-1)}[k] \\ a^{(0)}[n] = x^{(0)} \end{cases}$$

Synthèse : passage de l'échelle l à $l - 1$

$$a^{(l-1)}[n] = \sum_{k \in \mathbb{Z}} a^{(1)}[k] g_0[n - 2k] + \sum_{j \in \mathbb{Z}} d^{(1)}[j] g_1[n - 2j]$$

avec $g_1[k] = (-1)^k h_0[1 - k]$ et $h_1[k] = (-1)^k g_0[1 - k]$.

En résumé, les équations de dilatation (2.50) et (2.51) inhérentes à l'analyse multi-résolution permettent respectivement d'identifier les filtres passe-bas et passe haut de synthèse d'un banc de filtres à 2 canaux. La transformée discrète en ondelettes (DWT : Discrete Wavelet Transform) peut être vue comme une transformée implémentée sous forme d'une cascade de banc de filtres où les échantillons obtenus en sortie des filtres passe-haut correspondent aux coefficients d'ondelettes. Les coefficients d'approximation $a^{(L)}[n]$ de faible résolution issus du filtrage passe-bas apparaissent uniquement au dernier niveau de décomposition.

Bi-orthogonalité

Jusqu'à présent l'analyse multi-résolution impliquait l'introduction d'une base d'ondelettes orthonormée. La transformée en ondelettes permettant d'obtenir les coefficients issus de la projection du signal sur cette base est associée à un banc de filtres à structure arborescente. Malheureusement la propriété d'orthogonalité des sous-espaces V_0 et W_0 introduit des contraintes sur la construction des ondelettes.

Ainsi, il n'est pas possible d'implémenter une transformée orthogonale par décomposition en sous-bandes et dont les filtres de synthèse sont symétriques et de support fini (hormis le cas particulier du filtre de Haar, à 2 coefficients avec $g_0[0] = g_0[1] = \frac{1}{\sqrt{2}}$). La symétrie des filtres est une propriété intéressante pour les applications de compression car elle implique un filtrage à phase linéaire. Or les filtres à phase non-linéaire (donc non symétriques) introduisent une distorsion visuelle sur les contours des images [51].

Dans le cas orthonormal et d'après l'équation (2.46), les filtres d'analyse $h_0[n]$ et $h_1[n]$ correspondent aux filtres $g_0[n]$ et $g_1[n]$ "retournés" temporellement. L'analyse par banc de filtres assure la projection du signal d'entrée $x^{(0)}$ sur les espaces de résolution V_j et W_j par le produit scalaire :

$$a_0^{(j)}[n] = \langle x^{(0)}, \varphi_n^{(j)} \rangle \text{ et } d_1^{(j)}[n] = \langle x^{(0)}, \psi_n^{(j)} \rangle$$

Dans le cas non-orthogonal, on s'affranchit de la contrainte d'orthogonalité en introduisant les fonctions duales $\tilde{\varphi}$ et $\tilde{\psi}$ des fonctions d'échelle et d'ondelettes φ et ψ permettant d'assurer la reconstruction parfaite. Le produit scalaire s'exprime alors :

$$a_0^{(j)}[n] = \langle x^{(0)}, \tilde{\varphi}_n^{(j)} \rangle \text{ et } d_1^{(j)}[n] = \langle x^{(0)}, \tilde{\psi}_n^{(j)} \rangle$$

Sous certaines conditions, l'ensemble des fonctions $\{\tilde{\psi}_n^j\}_{n \in \mathbb{Z}}$ constitue alors une base d'ondelette duale pour $L^2(\mathbb{R})$. La condition de biorthogonalité est alors donnée par :

$$\langle \psi_n^{(j)}, \tilde{\psi}_m^{(k)} \rangle = \delta[n - m][j - k], \quad \forall n, m, j, k \in \mathbb{Z}$$

L'ensemble des fonctions $\{\tilde{\varphi}_n^j\}_{n \in \mathbb{Z}}$ et $\{\tilde{\psi}_n^j\}_{n \in \mathbb{Z}}$ doit former respectivement les bases des sous-espaces duaux \tilde{V}_j et \tilde{W}_j tel que \tilde{W}_j soit le complément non-orthogonal de \tilde{V}_j dans \tilde{V}_{j-1} . L'analogie avec les bancs de filtres conduit alors à ce que les filtres d'analyse et de synthèse vérifient la relation (2.46).

La fonction d'analyse-synthèse sur L niveau de résolution permet de représenter $x^{(0)} \in V_0$ comme la somme de son approximation dans l'espace V_L et des détails nécessaires à la reconstruction parfaite :

$$x^{(0)} = \sum_{n \in \mathbb{Z}} \langle x^{(0)}, \varphi_n^L \rangle \tilde{\varphi}_n^L + \sum_{l=1}^L \sum_{n \in \mathbb{Z}} \langle x^{(0)}, \psi_n^l \rangle \tilde{\psi}_n^l$$

Quelques propriétés des fonctions d'ondelettes

L'analyse multirésolution nous a donc amené à la transformée en ondelettes associée à une transformée en sous-bande arborescente caractérisée par l'introduction de filtres d'analyse et de synthèse. Cette analyse dépend implicitement du choix des fonctions d'ondelettes et d'échelle et de certaines propriétés souhaitables afin de rendre la transformée efficace pour des applications liées à la compression.

Support :

La propriété de support compact des fonctions d'échelle et d'ondelettes garantit une localisation spatiale précise dans l'analyse des signaux. Les filtres d'analyse H et synthèse G associés sont alors des filtres à réponse impulsionnelle finie (FIR) facilement implémentables d'un point de vue algorithmique.

Orthogonalité, Bi-orthogonalité et Symétrie :

Les fonctions d'échelle et d'ondelettes symétriques ou antisymétriques sont importantes parce qu'elles permettent de construire des bases d'ondelettes régulières sur un intervalle, et non plus sur tout l'axe réel. Cependant, mis à part le cas particulier de l'ondelette de Haar, la symétrie des filtres permettant d'obtenir une phase linéaire est incompatible avec l'orthonormalité.

Le fait qu'il n'existe pas d'ondelettes orthogonales qui vérifient à la fois les propriétés de support compact et de symétrie, a amené Cohen, Daubechies, Feauveau [12] à introduire une famille de transformées bi-orthogonales impliquant des filtres à phase linéaire

qui n'introduisent pas de distorsion de phase à la reconstruction.

Moments nuls :

Une ondelette a $k + 1$ moments nuls si et seulement si, sa fonction d'échelle restitue les polynômes de degré inférieur ou égal à k . Mathématiquement, cette propriété se traduit par :

$$\int t^j \psi(t) dt = 0 \quad \forall j = 0 \dots k$$

En d'autres termes tous les polynômes $p(t) = \sum_{0 \leq j \leq k} a_j t^j$ ont des coefficients d'ondelettes ou de détails nuls. Cette propriété est intéressante du point de vue de la compression car plus la fonction d'ondelettes aura de moments nuls, meilleure sera la capacité d'approximation de la transformée.

Régularité :

La régularité d'une fonction d'ondelettes est identique à celle de la fonction d'échelle correspondante. Elle est liée à un certain degré de continuité et de différentiabilité des fonctions d'ondelettes et d'échelle. Notons tout de même les deux résultats essentiels suivants :

- Il n'existe pas d'ondelette orthogonale indéfiniment dérivable à support compact.
- Pour de nombreuses classes d'ondelettes orthogonales, l'augmentation de la régularité passe par une augmentation du nombre de moments.

Quelques ondelettes :

L'orthonormalité étant incompatible avec les filtres à phase linéaire, nous restreignons nos exemples de construction de fonction d'ondelettes au cas bi-orthogonal.

Etant donné les filtres de synthèse d'un banc de filtres à reconstruction parfaite, les transformées de Fourier des fonctions d'échelle et d'ondelettes notées respectivement $\phi(\omega)$, $\Psi(\omega)$ vérifient les équations suivantes obtenues à partir de la transformée de Fourier des équations de dilatation (2.50) et (2.51) :

$$\phi(\omega) = \alpha \prod_{j=1}^{\infty} G_0\left(\frac{\omega}{2^j}\right) \tag{2.56}$$

$$\psi(\omega) = \alpha e^{-j\frac{\omega}{2}} G_1\left(\frac{\omega}{2}\right) \prod_{j=1}^{\infty} G_0\left(\frac{\omega}{2^j}\right) \tag{2.57}$$

où $G_0(\omega) = \sum_{n \in \mathbb{Z}} g_0[n] e^{-jn\omega}$ et $G_1(\omega) = \sum_{n \in \mathbb{Z}} g_1[n] e^{-jn\omega}$ sont les transformées de Fourier discrètes des séquences $g_0[n]$ et $g_1[n]$. α est une constante de normalisation dépendante du facteur de normalisation des fonctions d'échelle et d'ondelettes.

Les séquences $m_0[n] = \frac{1}{\sqrt{2}} g_0[n]$ et $\tilde{m}_0[n] = \frac{1}{\sqrt{2}} h_0[-n]$ sont soumises à la contrainte de bi-orthogonalité permettant la reconstruction parfaite :

$$M_0(z) \tilde{M}_0(z^{-1}) + M_0(-z) \tilde{M}_0(-z^{-1}) = 1$$

où de manière équivalente dans le domaine de Fourier :

$$M_0(\omega)\tilde{M}_0^*(\omega) + M_0(\omega + \pi)\tilde{M}_0^*(\omega + \pi) = 1 \quad (2.58)$$

La contrainte du nombre de moments N de la fonction d'ondelettes est directement reliée dans le domaine de Fourier au nombre de zéros à $\omega = \pi$ pour le polynôme trigonométrique $G_0(\omega)$ (ou de manière équivalente $G_0(z)$ doit avoir autant de zéro à $z = 1$ que possible). Il s'en suit que l'on peut écrire $M_0(\omega)$ et $\tilde{M}_0(\omega)$ sous la forme :

$$M_0(\omega) = \left(\frac{1 + e^{-j\omega}}{2}\right)^N r(\omega) \quad \text{et} \quad \tilde{M}_0(\omega) = \left(\frac{1 + e^{-j\omega}}{2}\right)^N \tilde{r}(\omega)$$

Le fait de choisir $m_0[n]$ et $\tilde{m}_0[n]$ comme des filtres à phase linéaire symétrique autour de $n = 0$ implique que le transformée de Fourier est une fonction polynômiale en $\cos\omega$. En notant $N = 2L$ et $\tilde{N} = 2\tilde{L}$, on peut alors réécrire l'expression des filtres sous la forme :

$$M_0(\omega) = \left(\frac{1 + e^{-j\omega}}{2}\right)^{2L} p_0(\omega) \quad \text{et} \quad \tilde{M}_0(\omega) = \left(\frac{1 + e^{-j\omega}}{2}\right)^{2\tilde{L}} \tilde{p}_0(\omega)$$

La condition de biorthogonalité de l'équation (2.58) s'écrit alors :

$$\left(\cos^2\frac{\omega}{2}\right)^{L+\tilde{L}} p_0(\cos\omega)\tilde{p}_0(\cos\omega) + \left(\cos^2\frac{\omega + \pi}{2}\right)^{L+\tilde{L}} p_0(\cos(\omega + \pi))\tilde{p}_0(\cos(\omega + \pi)) = 1 \quad (2.59)$$

La solution de cette équation a été trouvée par Daubechies [18] en utilisant le théorème de Bezout et en posant $p_0(x)\tilde{p}_0(x) = Q(\frac{1}{2}(1-x))$. On a alors :

$$Q(x) = \sum_{n=0}^{L+\tilde{L}-1} \binom{L+\tilde{L}-1}{n} x^n + x^M R\left(\frac{1}{2}-x\right)$$

où $R(x)$ est un polynôme de degré impair.

Pour des filtres à phase linéaire de longueur minimale, il suffit de choisir $R(x) = 0$ ce qui amène à la construction de filtres biorthogonaux. Différentes familles d'ondelettes sont ainsi obtenues en choisissant les paramètres L et \tilde{L} (le nombre de moments nuls que l'on souhaite obtenir pour la fonction d'ondelettes) et la façon dont on répartit les facteurs du polynôme $P(x) = p_0(x)\tilde{p}_0(x)$ entre $p_0(x)$ et $\tilde{p}_0(x)$.

En prenant par exemple $L = \tilde{L} = 1$ de telle sorte que $M_0(\omega)$ et $\tilde{M}_0(\omega)$ aient chacun 2 zéros à $\omega = \pi$, on obtient la transformée "Spline 5/3" car la fonction d'échelle correspondante est une B-spline. Les filtres passe-bas et passe-haut d'analyse ont 5 et 3 coefficients ce qui en fait une transformée populaire notamment recommandée par la norme JPEG2000 en raison de sa simplicité en terme d'implémentation et la possibilité d'obtenir des coefficients "entiers" (propriété intéressante pour le codage sans perte) sous la forme d'implémentation "Lifting" [42]. Les fonctions d'échelle d'analyse et de synthèse $\varphi(t)$ et $\tilde{\varphi}(t)$ ainsi que les ondelettes $\psi(t)$ et $\tilde{\psi}(t)$ sont illustrées sur la figure 2.8.

Antonini et al [5] ont proposé d'introduire un nombre de zéros équivalent entre $M_0(\omega)$ et $\tilde{M}_0(\omega)$ avec une répartition la plus équilibrée possible des facteurs de $P(x)$ entre $p_0(x)$ et $\tilde{p}_0(x)$. En choisissant $L = \tilde{L} = 2$ de telle sorte qu'il ait $N = \tilde{N} = 4$ zéros à π , on obtient

en particulier la transformée bi-orthogonale 9/7 en référence respectivement au nombre de coefficients du filtre passe-bas et passe-haut d'analyse.

Les fonctions d'échelle d'analyse et de synthèse $\varphi(t)$ et $\tilde{\varphi}(t)$ ainsi que les ondelettes $\psi(t)$ et $\tilde{\psi}(t)$ sont tracées sur la figure 2.9. Les fonctions d'analyse et de synthèse sont ici beaucoup plus proches les unes des autres que dans le cas de la transformée 5/3, ce qui tend à montrer que cette transformée est approximativement orthogonale avec d'autre part une fonction d'ondelettes très peu oscillante. Pour des forts taux de compression l'oscillation de la fonction d'ondelettes est à l'origine d'une impression d'ondulation lors de la restitution de seuils francs, ce que prévient la transformée 9/7. La transformée biorthogonale 9/7 connaît ainsi un large succès pour de nombreuses applications dédiées à la compression.

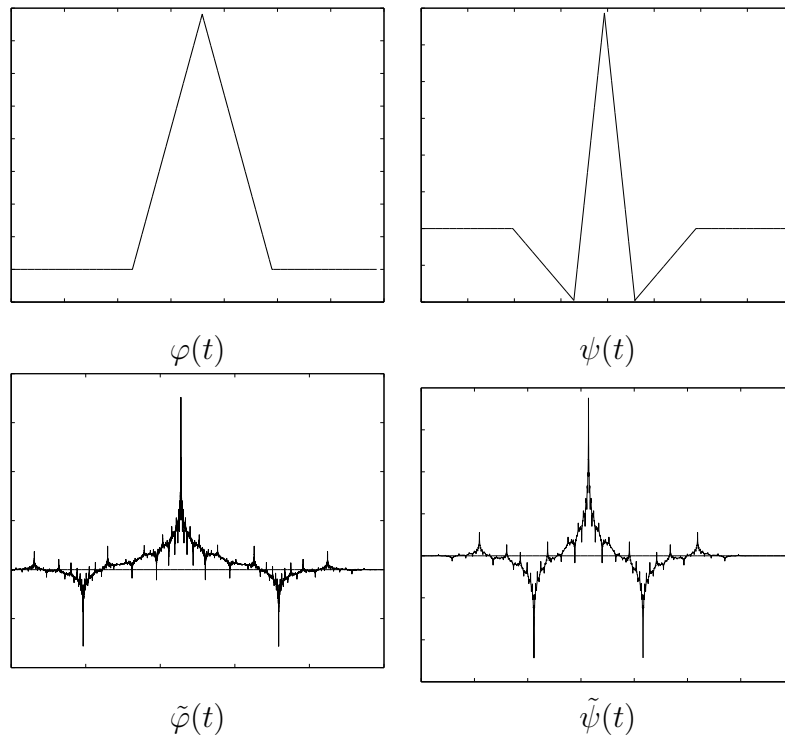


FIG. 2.8 – Fonctions d'échelles et d'ondelettes pour la transformée biorthogonale 5/3.

k	$h_0[k]/\sqrt{2}$	$\tilde{h}_0[k]/\sqrt{2}$
0	0.75	0.50
⁺ - 1	0.25	0.25
⁺ - 2	-0.125	0

TAB. 2.3 – Coefficients des filtres d'ondelettes pour la transformée biorthogonale spline 5/3.

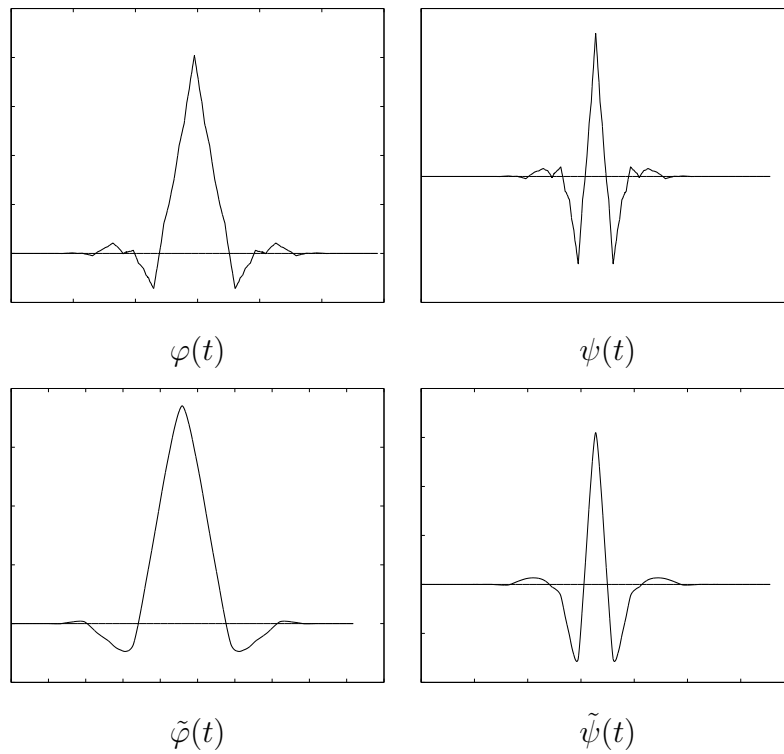


FIG. 2.9 – Fonctions d'échelle et d'ondelettes pour la transformée biorthogonale 9/7.

k	$h_0[k]/\sqrt{2}$	$\tilde{h}_0[k]/\sqrt{2}$
0	0.602949	0.557543
⁺ - 1	0.266864	0.295635
⁺ - 2	0.078223	0.028771
⁺ - 3	0.016864	0.045635
⁺ - 4	0.026748	0

TAB. 2.4 – Coefficients des filtres d'ondelettes pour la transformée biorthogonale spline 9/7.

Application aux images RSO :

Dans le cadre de traitement de signaux de nature complexe, les ondelettes de Daubechies orthogonales, symétriques et à valeurs complexes ont été introduites par Lawton [49] et détaillées par Lina [52]. Les coefficients du filtre $h_0[k]$ de cette famille d'ondelettes SDW (pour *Symmetric Daubechies Wavelets*) sont donnés par la table 2.5 en fonction de la longueur $2J + 2$ du filtre.

Ives [43] a tenté de déterminer s'il existait un réel avantage à utiliser des ondelettes à valeurs complexes pour la compression d'images SLC. Ses travaux ont consisté à étudier les différences de qualité de l'algorithme de quantification par arbre de zéros EZW [82] appliqué séquentiellement sur les parties réelle et imaginaire à partir de familles d'ondelettes symétriques complexes de Daubechies à 6 et 10 coefficients et d'ondelettes orthogonales de Zang et al [114]. Il apparaît alors que l'introduction d'ondelettes de nature complexe n'apporte aucun gain particulier par rapport aux ondelettes de Daubechies à valeurs réelles et à 4,6,10 coefficients.

k	$J = 2$	$J = 4$	$J = 6$
+ 1/0	0.662912 + 0.171163i	0.643003 + 0.182852i	0.633885 + 0.179835i
+ 2/ - 1	0.110485 - 0.085581i	0.151379 - 0.094223i	0.171512 - 0.094452i
+ 3/ - 2	-0.066291 - 0.085581i	-0.080639 - 0.117947i	-0.086478 - 0.130859i
+ 4/ - 3	0	-0.017128 + 0.008728i	-0.030746 + 0.014044i
+ 5/ - 4	0	0.010492 + 0.020590i	0.017651 + 0.037601i
+ 6/ - 5	0	0	0.003238 - 0.001300i
+ 7/ - 6	0	0	-0.001956 - 0.004869i

TAB. 2.5 – Coefficients des filtres d'ondelettes symétriques SDW à valeurs complexes.

DWT en 2 dimensions

Jusqu'à présent nous avons considéré la décomposition d'un signal mono-dimensionnel par DWT en deux sous-bandes par application d'un banc de filtres à 2 canaux. De manière similaire aux transformées en bloc du type DCT, la DWT peut être étendue au cas bi-dimensionnel en appliquant les filtres de manière séparable sur les dimensions de l'image.

Plus spécifiquement, le filtrage mono-dimensionnel est appliqué sur chaque ligne de l'image. Les coefficients finaux sont obtenus en réappliquant la transformée sur les colonnes. Il en résulte une décomposition en 4 sous-bandes classiquement dénotées LL,LH,HL,HH. Ces sous-bandes sont implicitement associées à une analyse par une fonction d'échelle à deux dimensions et trois fonctions d'ondelettes :

- $\varphi_{LL}(x, y) = \varphi(x)\varphi(y)$
- $\psi_{HL}(x, y) = \varphi(x)\psi(y)$
- $\psi_{LH}(x, y) = \psi(x)\varphi(y)$
- $\psi_{HH}(x, y) = \psi(x)\psi(y)$

La sous-bande "LL" est une version filtrée passe-bas et sous résolue de l'image originale. La sous-bande "HL" apporte le contenu haute-fréquence dans la direction horizontale et basse-fréquence dans la direction verticale. En d'autres termes, les lignes et contours

horizontaux y sont principalement mis en valeur. A l'inverse, la sous-bande "LH" répond plus fortement aux contours et lignes verticales alors que la sous-bande "HH" contient principalement les structures diagonales.

Les figures 2.10 et 2.11 illustrent respectivement la décomposition séparable en deux dimensions par banc de filtres et la représentation des sous-bandes sous la forme classique pyramidale dans l'espace fréquentiel.

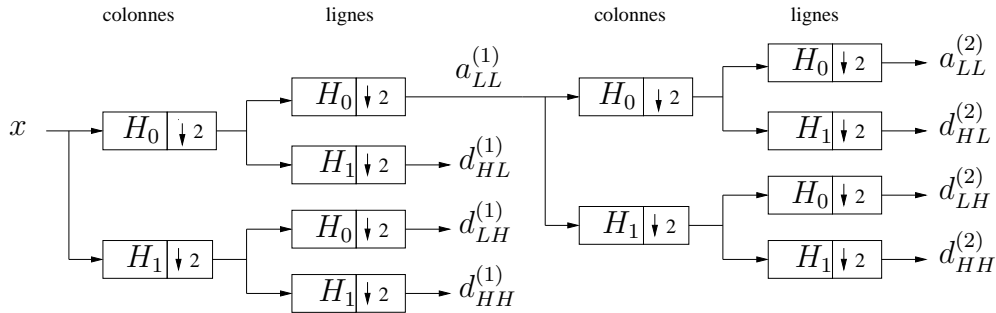


FIG. 2.10 – Implémentation de la DWT (analyse) en 2D par banc de filtres sur L=2 niveaux de décomposition.

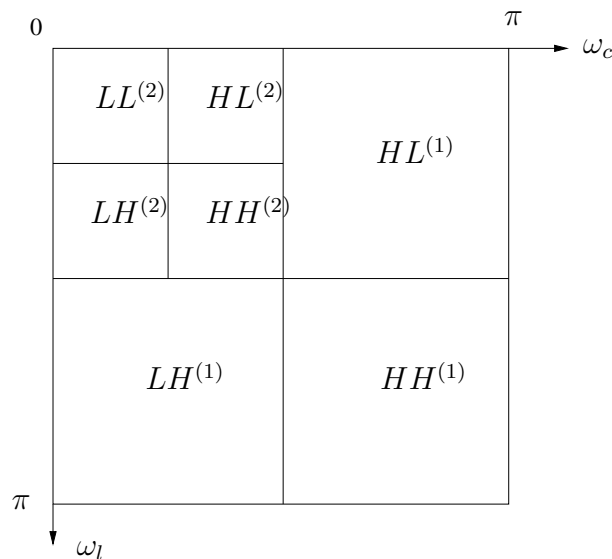


FIG. 2.11 – Représentation des sous-bandes dans le plan fréquentiel après DWT bidimensionnelle sur 2 niveaux

Quelques algorithmes de compression basés sur la transformée en ondelettes

Dans son principe l'image est représentée en termes de coefficients d'ondelettes dans l'espace transformé sur L niveaux de décomposition (donc $3L+1$ sous-bandes). La structure approximative de l'image a une représentation très compacte qui autorise alors plusieurs stratégies de quantification des coefficients. Appliquée à la compression d'images RSO, il est important de noter que ces méthodes ont été principalement appliquées sur l'image d'amplitude.

Quantification scalaire des coefficients d'ondelettes

Cette méthode a été proposée par J. Mvogo [65] pour la compression d'image RSO d'amplitude. La transformée en ondelettes est appliquée sur l'image d'amplitude. Le choix de la base d'analyse est alors déterminée par la fonction qui minimise l'entropie des coefficients d'ondelettes. La quantification des coefficients réalisée scalairement dépendra de quantificateurs scalaires uniformes, à zone morte autour de 0 ou multicouches [91]. Le codage des indices de quantification est assuré par un codeur arithmétique de type adaptatif à partir d'un modèle statistique des coefficients d'ondelettes basé sur la loi gaussienne généralisée. Une procédure d'allocation dynamique permet alors de déterminer le meilleur compromis débit-distorsion à répartir sur les sous-bandes des coefficients d'ondelettes.

Quantification Vectorielle

La quantification vectorielle algébrique des coefficients d'ondelettes a été étudiée notamment par Blain et Fisher [8], P. Mathieu M. Barlaud et al [59] [7]. La modélisation des coefficients d'ondelettes par une distribution laplacienne les amène à considérer une projection des coefficients sur des surfaces d'énergie constante du réseau de lattices appelées "Pyramides" et définie à partir de la norme L^1 .

Appliquée aux images RSO, G. Mercier [61] propose quant à lui une méthode de quantification vectorielle en arborescence. La construction du dictionnaire est basé sur une carte auto-organisatrice de Kohonen de vecteurs arborescents.

Quantification par Arbres de zéros : EZW et SPIHT

L'algorithme EZW (Embedded Zerotrees on Wavelets coefficients) a été introduit par Shapiro [82]. Le principe est d'utiliser la structure hiérarchique de la transformée en ondelettes et consiste à exploiter les redondances entre différentes sous-bandes associées à la capture d'une information de même orientation à des niveaux d'échelles adjacents (par exemple les sous-bandes LH_3, LH_2, LH_1).

Il est ainsi possible de définir des relations parents-enfants entre les coefficients de même orientation et issus de sous-bandes adjacentes. Les sous-bandes doublant de résolution à mesure que l'échelle diminue, un coefficient de la sous-bande LH_l disposera de 4 coefficients "descendants" à la même localisation spatiale dans la sous-bande LH_{l-1} . Ces relations sont également définies pour les sous-bandes d'orientation HL, HH et pour les niveaux de décomposition $l = 2 \dots L$.

L'algorithme de EZW tire fortement profit de la structure pyramidale de la décomposition en ondelettes afin de réaliser une compression efficace de l'image. La compression provient principalement de deux processus :

- Le premier consiste en l'encodage itératif des coefficients dont les valeurs sont fixées par rapport au seuil lors de l'itération considérée. Si la valeur absolue du coefficient est supérieure ou égale à ce seuil, le coefficient est dit "significatif" (*significant*), et "non-significatif" (*insignificant*) dans le cas contraire.
- Le second processus de compression réside dans l'utilisation d'arbres de zéros. Lors qu'une racine d'arbre est identifiée, tous les coefficients descendants ne sont plus encodés.

En divisant par deux le seuil à chaque itération, il existe une équivalence entre les comparaisons des valeurs de coefficients par rapport à un seuil et une quantification scalaire avec zone morte. Ces comparaisons donnent alors lieu à un codage progressif des coefficients par plan de bits. Les indices représentant les valeurs de comparaisons sont ensuite encodés entropiquement à l'aide d'un codeur arithmétique.

L'algorithme populaire SPIHT (Set Partitioning In Hierarchical Trees) de Said et Pearlman [81] s'inspire et améliore la stratégie de Shapiro en offrant plus de degrés de liberté pour définir l'arborescence en introduisant deux ensembles de coefficients non significatifs. La figure 2.12 illustre la dépendance parents-descendants entre les sous-bandes de l'image pour l'algorithme SPIHT. Dans cet arbre, chaque noeud consiste en un groupe de 2×2 pixels et chaque pixel dans le noeud possède 4 descendants excepté au plus haut niveau de décomposition, où un pixel en un noeud indiqué par "*" n'a pas de descendants.

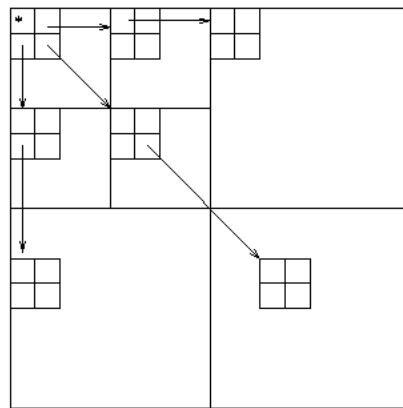


FIG. 2.12 – Relation parents-enfants permettant de définir l'arborescence de coefficients non-significatifs dans l'algorithme SPIHT.

L'algorithme SFQ proposé par Xiong [118] propose une approche hybride consistant en deux étapes :

- Quantification par arbre de zéros de coefficients qui sont groupés dans la structure d'arborescence parents-enfants de l'algorithme EZW.
- Quantification scalaire des coefficients survivants à la quantification par arbre de zéros.

Le but de l'algorithme SFQ est alors d'optimiser conjointement ces deux modes de quantification au sens du meilleur compromis débit-distorsion. Cet algorithme amène ainsi à chercher le meilleur compromis entre un grand sous-ensemble de coefficients quantifiés avec une faible précision et un plus petit sous-ensemble de coefficients quantifiés avec une meilleure précision. Le meilleur pas de quantification est alors celui qui minimise le compromis débit-distorsion $J = D + \lambda R$ dont le paramètre optimal λ est recherché par un algorithme d'allocation optimale de débit de Shoham et Gersho [83].

Application aux images RSO

Gleich [37] a étudié les performances des algorithmes EZW, SPIHT et SFQ sur les images RSO d'amplitude. En outre, il propose l'algorithme PSFQ introduit comme une

extension de l'algorithme SFQ au codage par plan de bits. Ce codage progressif permet ainsi d'adapter spatialement le débit de la quantification scalaire, ce qui permet notamment de représenter les points brillants avec une très bonne qualité radiométrique. En utilisant l'ondelette biothogonale 9/7 comme support pour la DWT, les auteurs ont démontré l'intérêt de la quantification par arbres de zéros pour les images RSO d'amplitude en termes de rapport signal sur bruit global sur toute l'image. La méthode SPIHT présente tout de même des performances légèrement inférieures à la méthode PSFQ en terme de rapport sur bruit mais offre un meilleur compromis d'un point de vue complexité de calcul.

2.6 Conclusion

Nous avons vu au cours de ce chapitre, les principales techniques mises en oeuvre pour la compression d'images. De par la nature même du signal, les images RSO sont cependant fondamentalement différentes des images naturelles :

- L'amélioration des technologies liées au capteur qui permet d'imager des zones avec une résolution métrique ou sub-métrique implique un volume de données considérable, d'autant plus que les pixels sont codés sur une dynamique importante de 64 bits par pixel complexe (32 bits pour la partie réelle et 32 pour la partie imaginaire pour le capteur RAMSES de l'ONERA).
- L'entropie des images RSO est plus importante que celle des images naturelles. Ceci est dû en particulier aux effets de chatoiement donnant un aspect texturé et granuleux à l'image avec une faible corrélation inter-pixellique.
- Contrairement aux images naturelles dont l'énergie est essentiellement concentrée dans les basses fréquences, les images RSO ont des composantes hautes fréquences plus énergétiques qu'il convient de traiter avec précaution.

Les techniques classiques de compression ne donnent pas des résultats aussi performants une fois appliquées aux images RSO. La transformée en cosinus discret (DCT) introduit à fort taux de compression des artéfacts d'effets de blocs dues à la forte dynamique des blocs capturant beaucoup d'informations hautes fréquences.

Avec une quantification vectorielle structurée (trellis, multi-étage), la qualité des images reconstruite est supérieure à celle obtenue par DCT avec une bonne préservation des textures et des points brillants due au codage par blocs à taille variable.

La transformée en ondelettes offre néanmoins des avantages certains en terme de concentration de l'énergie, de décorrélation tout en s'adaptant à la non-stationnarité des signaux.

Nous proposons ainsi un axe de recherche portant sur des algorithmes de compression d'images RSO SLC basées sur la transformée en ondelettes en se focalisant plus spécifiquement sur la nature complexe du signal.

La représentation sous forme d'image d'amplitude est nécessaire pour la photo-interprétation et autres applications liées à un contexte de détection-reconnaissance-identification. Pour des applications interférométriques, l'information de phase conserve toute son importance. A priori, il semble naturel de penser que la compression appliquée aux données de module et de phase plutôt que sur partie réelle et imaginaire, permette de mieux contrôler la qualité de restitution de l'information de phase et de module.

Il se pose ainsi la problématique de la représentation et de la compression d'une source à valeur complexe, considérée comme un vecteur à 2 dimensions. A cet effet, le chapitre suivant est consacré à l'élaboration de plusieurs techniques de compression dépendantes de la projection de l'image SLC dans le plan complexe.

Chapitre 3

Vers une méthodologie de compression adaptée à la représentation des données complexes

Ce chapitre a pour objectif de tirer partie des chapitres 1 et 2 afin de proposer une méthodologie de compression basée sur la transformée en ondelettes et spécifiquement dédiée aux données de natures complexes des images RSO SLC. Ainsi, nous procéderons à l'étude de problèmes concernant :

- Le choix des techniques de compression. En référence à l'état de l'art du chapitre 2, nous proposerons quatre méthodes de compression basées sur différentes possibilités de représentation des données de nature complexe dans l'espace de l'image originale mais également dans l'espace transformé.
- La caractérisation des statistiques des coefficients d'ondelettes à valeurs complexes après DWT sur l'image RSO SLC. Une connaissance précise permettra d'aboutir à une modélisation statistique fidèle au comportement des coefficients d'ondelettes. Nous pourrons en déduire une méthode de quantification et d'allocation des débits adaptées au modèle de représentations des données. Nous aborderons ainsi plus en détails la distribution de Fisher déjà introduite lors du chapitre 1.

3.1 Représentation dans le plan complexe et mesure de distorsion

3.1.1 Représentation cartésienne et polaire

Soit $z(m, n) = i(m, n) + j q(m, n) \in \mathbb{C}$ un pixel complexe de l'image SLC Z de taille $M \times N$ où $\{i, q\}$ correspondent respectivement à la partie réelle et imaginaire (i pour *In Phase* et q pour *In Quadrature*). La représentation **cartésienne** de l'image dans le plan complexe s'exprime sous la forme $Z = I + jQ$.

On introduit également la représentation **polaire** où $z(m, n) = a(m, n)e^{j\theta(m, n)}$ et $\{a, \theta\}$ sont respectivement associés au module (amplitude) et à la phase. La représentation polaire de l'image dans le plan complexe s'exprime alors sous la forme $Z = Ae^{j\Theta}$.

Le module et la phase sont reliés aux parties réelle et imaginaire par les relations :

$$\begin{aligned}i &= a \cos \theta \in \mathbb{R} \\q &= a \sin \theta \in \mathbb{R} \\a &= \sqrt{i^2 + q^2} \in [0, +\infty] \\ \theta &= \arctan\left(\frac{q}{i}\right) \in [0, 2\pi]\end{aligned}$$

On notera de manière analogue $\hat{z}(m, n) = \tilde{i}(m, n) + j \tilde{q}(m, n) = \hat{a}(m, n)e^{j\hat{\theta}(m, n)} \in \mathbb{C}$ le pixel de l'image \hat{Z} reconstruit après décompression.

La compression avec pertes introduit une dégradation irréversible. De manière à pouvoir évaluer une méthode de compression, il est donc nécessaire de comparer la qualité de l'image décompressée avec l'image originale. Définir un critère de qualité reste très difficile car il dépend de l'application. Une bonne qualité peut être synonyme de défauts imperceptibles à l'oeil nu, de préservation des contours, des structures géométriques... Les critères à mettre en oeuvre sont alors spécifiques à l'utilisation ultérieure des images. Néanmoins, un critère largement utilisé est la distorsion qui est une mesure de différence entre les deux images. De l'avis de la communauté radariste, ce critère global ne permet pas à lui tout seul d'évaluer la qualité d'un système de compression d'images RSO. Nous proposerons ainsi au chapitre 6, un ensemble de critères plus spécifiques à la caractérisation de la qualité d'images RSO SLC. Cependant, le critère de distorsion est une mesure pour laquelle il existe un certain nombre de résultats connus d'un point de vue théorie de l'information. Les techniques de quantification s'appuient en général sur la minimisation de la distorsion afin d'optimiser le système de compression. Citons en particulier la distorsion quadratique basée sur la norme L^2 à partir de laquelle on dérive la mesure d'"Erreur Quadratique Moyenne" (EQM).

3.1.2 Distorsion quadratique

Soit le vecteur aléatoire complexe à 2 dimensions $Z = \{I, Q\} = \{A, \Theta\}$ suivant les notations introduites précédemment et \hat{Z} le vecteur quantifié. La distorsion quadratique s'écrit :

$$\begin{aligned}
 D &= E_Z(|Z - \hat{Z}|^2) \\
 &= E_{I,Q}(|I - \tilde{I}|^2 + |Q - \tilde{Q}|^2) \\
 &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} [(i - \tilde{i})^2 + (q - \tilde{q})^2] p_{I,Q}(i, q) di dq, \\
 &= \int_0^{2\pi} \int_0^{+\infty} [(a \cos \theta - \hat{a} \cos \hat{\theta})^2 + (a \sin \theta - \hat{a} \sin \hat{\theta})^2] p_{\Theta,A}(\theta, a) da d\theta \\
 &= \int_0^{2\pi} \int_0^{+\infty} [a^2 - 2a\hat{a} \cos(\theta - \hat{\theta}) + \hat{a}^2] p_{\Theta,A}(\theta, a) da d\theta \\
 &= \int_0^{+\infty} (a - \hat{a})^2 p_A(a) da + \int_0^{+\infty} \int_0^{2\pi} 2a\hat{a}[1 - \cos(\theta - \hat{\theta})] p_{\Theta,A}(\theta, a) da d\theta \\
 D &= E_A[(A - \hat{A})^2] + 2E_{A,\Theta}[A\hat{A}(1 - \cos(\theta - \hat{\theta}))]
 \end{aligned} \tag{3.1}$$

Si l'on suppose une distribution circulaire du vecteur Z telles que les variables aléatoires A et Θ soient indépendantes, la distribution conjointe s'exprime en fonction des distributions marginales suivant la relation :

$$p_{\Theta,A}(a, \theta) = p_A(a)p_{\Theta}(\theta) \tag{3.2}$$

avec $p_{\Theta}(\theta) = \frac{1}{2\pi}$, $\theta \in [0, 2\pi]$

A partir de cette hypothèse, la distorsion quadratique s'écrit alors :

$$\begin{aligned}
 D &= \int_0^{+\infty} (a - \hat{a})^2 p_A(a) da + \int_0^{+\infty} 2a\hat{a} p_A(a) da \int_0^{2\pi} [1 - \cos(\theta - \hat{\theta})] p_{\Theta}(\theta) d\theta, \\
 &= E_A[(A - \hat{A})^2] + 2E_{\Theta}[1 - \cos(\theta - \hat{\theta})].E_A[A\hat{A}], \\
 D &= D_A + D_{\Theta/A}
 \end{aligned} \tag{3.3}$$

Le passage en coordonnées polaires permet donc de séparer deux termes distincts :

- la distorsion quadratique associée au module : $D_A = E_A[(A - \hat{A})^2]$.
- un terme de distorsion associé à la phase et pondéré par un terme quadratique d'amplitude : $D_{\Theta/A} = 2E_{\Theta}[1 - \cos(\theta - \hat{\theta})].E_A[A\hat{A}]$. Nous introduisons la notation $D_{\Theta/A}$ dans le sens où la distorsion de phase est conditionnée par le terme produit sur l'amplitude (bien que d'un point de vue statistique, la phase et l'amplitude soient considérées comme des variables indépendantes).

En ne considérant que le terme $2E_{\Theta}[1 - \cos(\theta - \hat{\theta})]$ et un développement limité au 1er ordre sur une analyse haute-résolution ($\Theta \sim \hat{\Theta}$), on retrouve la distorsion quadratique sur

la phase :

$$2E_{\Theta}[1 - \cos(\theta - \hat{\theta})] \sim E_{\Theta}[(\theta - \hat{\theta})^2] = D_{\Theta}$$

D'après les caractéristiques statistiques du chatolement étudiées au chapitre 1, on considère que les parties réelles et imaginaire de l'image SLC forment un vecteur gaussien circulaire où l'amplitude est indépendante de la phase. La distorsion dans le plan complexe peut donc s'exprimer de deux manières différentes suivant que l'on considère l'image SLC par le couple partie réelle/imaginaire où par le couple amplitude/phase :

$$D = D_I + D_Q = D_A + D_{\Theta/A} \quad (3.4)$$

avec $D_I = E_I[|I - \tilde{I}|^2]$ et $D_Q = E_Q[|Q - \tilde{Q}|^2]$.

L'estimation empirique de la distorsion sur les données aboutit à la définition de l'erreur quadratique moyenne. Dans le plan complexe, elle s'exprime par :

$$\begin{aligned} EQM &= \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |z(m, n) - \hat{z}(m, n)|^2 \\ &= \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N [[i(m, n) - \tilde{i}(m, n)]^2 + [q(m, n) - \tilde{q}(m, n)]^2] \\ EQM &= \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N (a(m, n) - \hat{a}(m, n))^2 \\ &\quad + \frac{2}{M * N} \sum_{m=1}^M \sum_{n=1}^N a(m, n) \hat{a}(m, n) [1 - \cos(\theta(m, n) - \hat{\theta}(m, n))] \quad (3.5) \end{aligned}$$

d'après l'équation (3.3).

Une mesure plus explicite est le rapport signal sur bruit (RSB) donné par l'expression suivante :

$$RSB = 10 \log_{10} \left[\frac{\sum_{m=1}^M \sum_{n=1}^N |z(m, n)|^2}{\sum_{m=1}^M \sum_{n=1}^N |z(m, n) - \hat{z}(m, n)|^2} \right] \quad (3.6)$$

$$(3.7)$$

3.2 Représentation des données SLC et méthodologies d'approches pour la compression

Le chapitre 2 nous a permis d'avoir une approche globale sur diverses techniques de compression. Dans le cadre de ce manuscrit, la compression fait intervenir 3 étapes essentielles que sont :

- La représentation du signal dans un espace propice à la compression des données (transformée).
- La quantification des données issues de la transformée.
- Le codage entropique des symboles représentatifs des coefficients quantifiés.

L'utilisation des ondelettes comme modèle de représentation des données détectées en imagerie RSO a permis de donner des résultats très intéressants de par les propriétés d'approximation liées à leur support compact, leur régularité, leurs moments nuls, leur symétrie. Dans un contexte de compression de données de nature complexe, il convient de se poser la question suivante : sous quelle forme représenter les données avant et après transformée ?

La stratégie de quantification constitue également un point clé sur l'efficacité de la compression puisque c'est à ce niveau que seront introduites les pertes sur les données. Cette quantification doit donc être adaptée aux caractéristiques des données, c'est à dire en particulier à leur représentation dans le domaine complexe. Quelle stratégie adopter ?

Le cas du codeur entropique est plus simple à régler. En effet, le codeur entropique utilisé sera le codeur arithmétique car il permet rapidement de converger vers l'entropie des symboles à encoder pour peu que l'estimation des probabilités dépendante d'un contexte (prise en compte de l'information apportée par les pixels voisins par exemple) et associées aux index soit fiable. Le codeur arithmétique contextuel est notamment utilisé dans les codecs classiques que sont JPEG2000 [93], SPIHT [81] et autres nombreux algorithmes de compression.

3.2.1 Approche 1 : Représentation cartésienne avant et après transformée

Il s'agit de considérer l'image RSO complexe Z comme formée d'une partie réelle et imaginaire $Z = I + jQ$. Il est alors naturellement possible de définir une transformée en ondelettes sur les pixels complexes $z = i + jq \in \mathbb{C}$. Le signal étant complexe, on se peut naturellement se poser la question suivante : y aurait-il un avantage particulier à utiliser une transformée en ondelettes complexes pour analyser des données de nature complexe ?

Pour répondre à cette question, nous avons vu (§ 2.5.2) que Ives et al ont étudié les différences de qualité de l'algorithme de quantification par arbre de zéros EZW [82] appliqué séquentiellement sur partie réelle et imaginaire [43]. La comparaison portait sur l'utilisation d'une famille d'ondelettes symétriques complexes de Daubechies à 6 et 10 coefficients et d'ondelettes orthogonales de Zang et al [114]. La conclusion de cette étude montre que la transformée en ondelettes à valeurs complexe n'apporte pas de gain en terme de qualité de reconstruction pour des taux de compression de 12 :1 à 4 :1. On notera par

ailleurs que l'utilisation d'ondelettes complexes augmente sensiblement la complexité de calcul de la transformée à cause des termes croisés. De par les propriétés de linéarité de la transformée en ondelettes et si les coefficients des filtres $\{g_0[n]\} \in \mathbb{R}$ et $\{g_1[n]\} \in \mathbb{R}$ respectivement représentatifs des fonctions d'échelle et ondelettes sont à valeurs réelles, on a de manière évidente :

$$DWT(I + jQ) \iff DWT(I) + j.DWT(Q) = I_{DWT} + j.Q_{DWT}$$

où I_{DWT} et Q_{DWT} correspondent à l'ensemble des coefficients d'ondelettes issus respectivement de la DWT de la partie réelle et imaginaire de l'image originale.

Nous obtenons donc deux images transformées. La première méthode de compression que nous proposons concerne la quantification progressive des coefficients des images de partie réelle et imaginaire suivant une approche similaire de celle proposée par Ives [43]. Les différences majeures entre les approches de quantification des coefficients d'ondelettes par plan de bits reposent sur la capacité à prédire ou à estimer au mieux le comportement local des coefficients. Citons par exemple les algorithmes par arbres de zéros qui exploitent la corrélation des coefficients inter-échelles (du type EZW [82] de Shapiro ou SPIHT de Said et Pearlman [81]). En fonction du comportement des coefficients voisins, d'autres méthodes consistent à introduire un codage contextuel des coefficients d'ondelettes (ECECOW de X. Wu [110] ou EBCOT de D. Taubman [92]).

Ordentlich et al [22] ont introduit une classification des coefficients basée sur le principe qu'il faut transmettre en priorité les coefficients offrant le meilleur compromis débit-distorsion (c'est à dire les coefficients qui apportent la plus forte décroissance de distorsion moyennant un coût de codage le plus faible possible). En fonction des plans de bits considérés, l'algorithme TCE (*Tarp-filter-based system and Classification of coefficients to achieve Embedding*) de C. Tian and S. S. Hemami [98] s'inspire de cette approche en introduisant une classification des coefficients d'ondelettes suivant leur "potentiel" à devenir significatif par rapport au plan de bit considéré.

Avec l'algorithme SPIHT, nous avons sélectionné l'algorithme TCE pour son efficacité comparable à JPEG2000 [93] et sa faible complexité. Une description plus détaillée de l'algorithme TCE sera faite au chapitre 4.

Indépendamment de toute stratégie de quantification, la transformée en ondelettes a pour particularité de concentrer l'énergie des coefficients dans les sous-bandes basses fréquences. Il est donc important que le spectre de l'image SLC à compresser soit centré autour de la fréquence nulle $f = 0$. Or, nous avons vu qu'au chapitre 1 un dépointage de l'antenne du capteur entraîne un décalage fréquentiel du spectre de l'image complexe dans le sens azimutal ("Doppler Centroid"). Il convient donc de recentrer le spectre de ces images en estimant le décalage fréquentiel sur le spectre de l'image obtenue après transformée de Fourier Discrète (DFT). Une fois le spectre recentré, la transformée de Fourier Discrète Inverse (IDFT) permettra ensuite de revenir dans l'espace image. Le recalage fréquentiel du "Doppler Centroid" sera bien entendu réappliqué après décompression de l'image.

La figure 3.1 illustre ainsi le synoptique de la chaîne de traitement proposée et que l'on notera **TCE-IQC** ou **SPIHT-IQC** (IQC pour *In Phase and In Quadrature Coding*) en fonction de l'algorithme de quantification choisi.

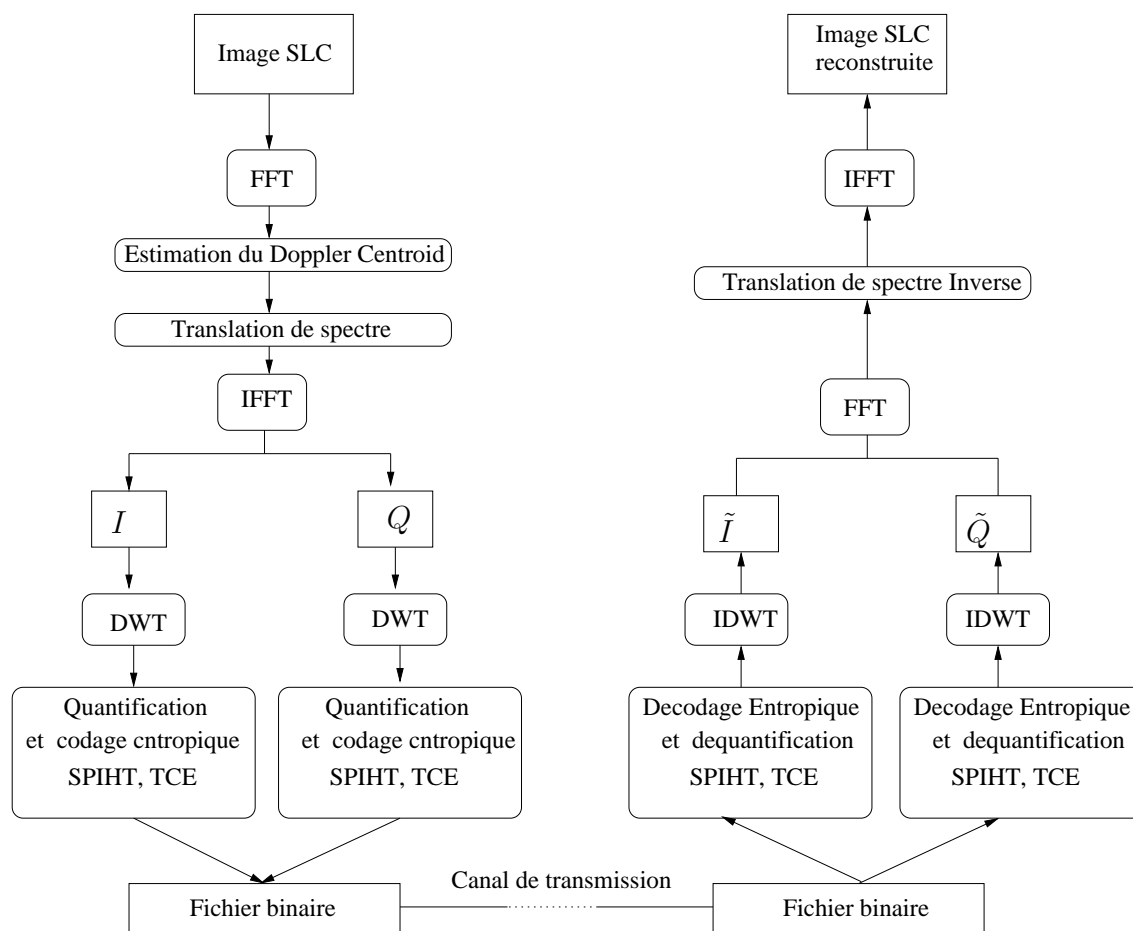


FIG. 3.1 – TCE-IQC : principe de la compression d'une image SLC basée sur transformée en ondelettes de la partie réelle et imaginaire. Les coefficients issus des images de partie réelle et imaginaire sont codés de manière indépendante par une méthode basée sur les arbres de zéros (SPIHT [81]) ou par classification des coefficients (TCE [98]).

3.2.2 Approche 2 : Représentation cartésienne avant transformée et polaire après transformée

Comme nous l'avons vu précédemment, la transformée en ondelettes d'une image complexe appliquée séquentiellement sur la partie réelle et imaginaire permet une représentation des coefficients dans le plan complexe. Chaque coefficient est ainsi obtenu au départ sous sa représentation cartésienne $c = x + j.y$, $\{x, y\} \in \mathbb{R}^2$. Représenté sous sa forme polaire, un coefficient d'ondelette est noté suivant l'expression :

$$c = a.e^{j\theta} \quad a \in [0; +\infty] \quad \theta \in [0; 2\pi]$$

où a et θ correspondent respectivement au module et à la phase du coefficient c .

La représentation polaire des coefficients d'ondelettes permet d'envisager une quantification polaire adaptée à ce type de représentation. Citons notamment la technique UPQ (Unrestricted Polar Quantization) [108] où les données complexes sont quantifiées suivant leur module et leur phase. La particularité de cette méthode (qui peut être vue comme une quantification vectorielle de dimension 2) est d'attribuer un pas de quantification aux données de phase différent suivant le niveau d'amplitude correspondant. Si l'on se fixe une contrainte de débit, la quantification polaire optimale consiste à trouver le meilleur compromis débit-distorsion au sens du minimum de l'erreur quadratique moyenne. En introduisant une contrainte entropique sur les index de quantification, nous proposons d'utiliser une Quantification Polaire Non-restreinte à Contrainte Entropique (ECUPQ : Entropy Constrained Unrestricted Polar Quantization [103]) appliquée aux sous-bandes de l'image transformée.

Comme nous le verrons au chapitre 4, la construction d'un quantificateur polaire optimal nécessite de connaître la distribution du module et de la phase des coefficients d'ondelettes complexes. En conséquence, il conviendra dans un premier temps de chercher à caractériser ces densités à partir de modèles de distribution adaptés aux statistiques des coefficients. Cette étape sera abordée à la section 3 de ce chapitre.

La figure 3.2 illustre ainsi le principe de cet algorithme que l'on notera **WECUPQ** (Wavelet and Entropy Constrained Unrestricted Polar Quantization).

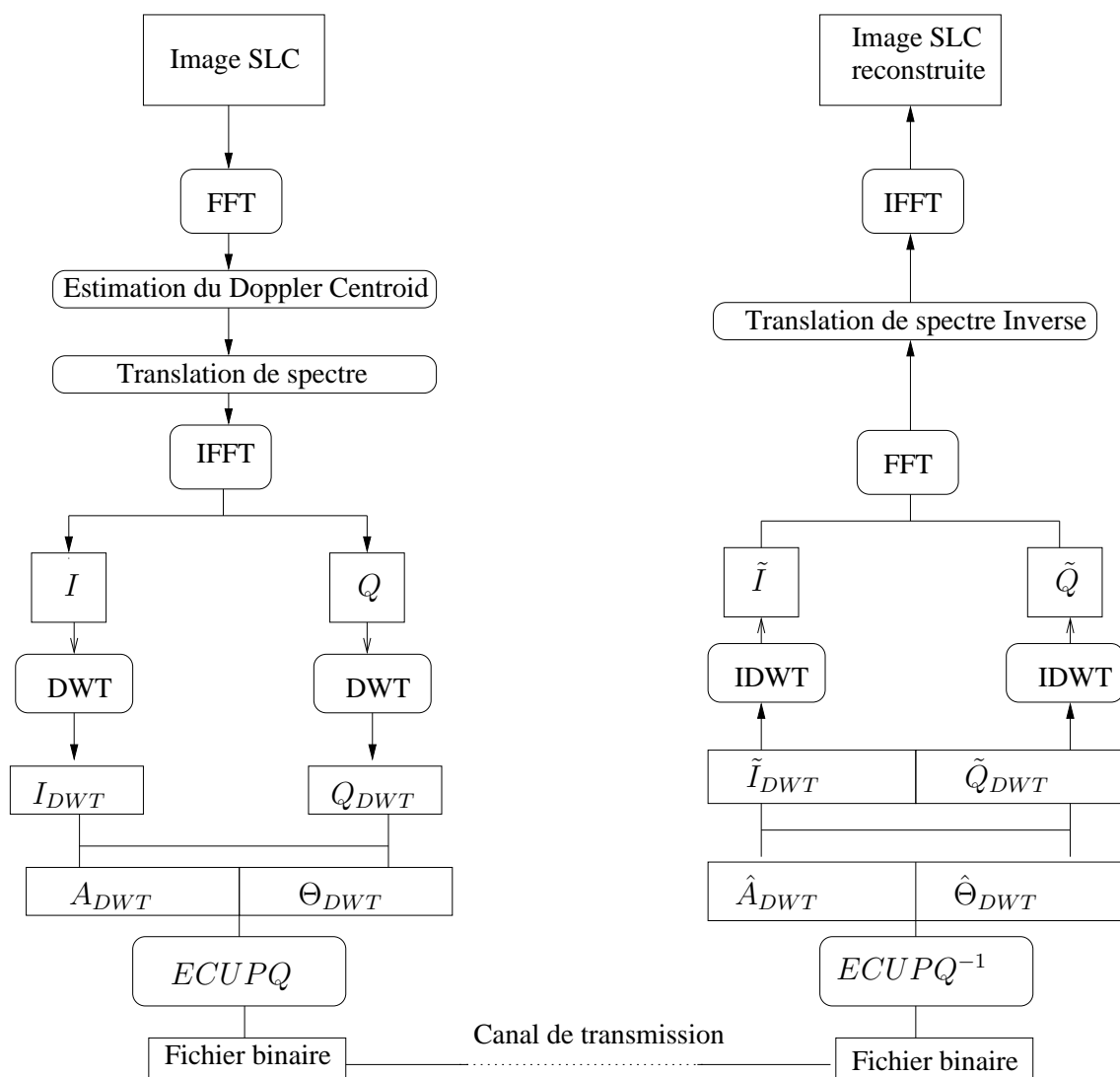


FIG. 3.2 – Méthode WECUPQ : principe de la compression d'une image SLC basée sur DWT à partir de la représentation module/phase des coefficients d'ondelette et quantification polaire par la méthode ECUPQ.

3.2.3 Approche 3 : représentation polaire avant transformée et compression indépendante module/phase

La troisième approche consiste à considérer l'image RSO SLC sous sa représentation polaire $Z = Ae^{j\theta}$. Les données d'amplitude ont la particularité d'avoir un spectre beaucoup plus concentré en basses fréquences avec une décroissance en $1/f$ [38]. Des algorithmes classiques du type SPIHT peuvent ainsi être appliqués sur l'image d'amplitude après transformée en ondelettes. A titre de comparaison par rapport aux méthodes de quantification par arbres de zéros, nous avons choisi l'algorithme TCE de Tian et Hemami [98]. Il est utile de remarquer que la compression des données d'amplitude ne nécessite pas une translation du spectre de l'image SLC, le spectre de l'image d'amplitude étant centré autour de la fréquence nulle.

La question relative à la représentation amplitude/phase de l'image SLC dépend plus spécifiquement de la compression de l'image de phase. Celle-ci étant considérée comme une source fortement décorrélée sans structure géométrique exploitable par une analyse multi-résolution, nous proposons en particulier une quantification vectorielle à l'aide de l'algorithme GLA décrit au chapitre 2. L'image d'amplitude étant comprimée séparément de l'image de phase, cette méthode ne prend pas compte la nature complexe de l'image. Il paraît donc illusoire d'espérer obtenir une qualité globale de l'image SLC qui soit meilleure que les approches précédentes. Néanmoins, cette approche offre l'intérêt de pouvoir contrôler séparément la qualité de reconstruction de l'image de phase et de l'image d'amplitude. Le synoptique de la figure 3.3 illustre cette approche où images d'amplitude et de phase sont compressées indépendamment l'une de l'autre. Nous ferons référence à cette technique par l'acronyme **TCEAC-IPQ** (TCE Amplitude Coding and Independant Phase Quantification).

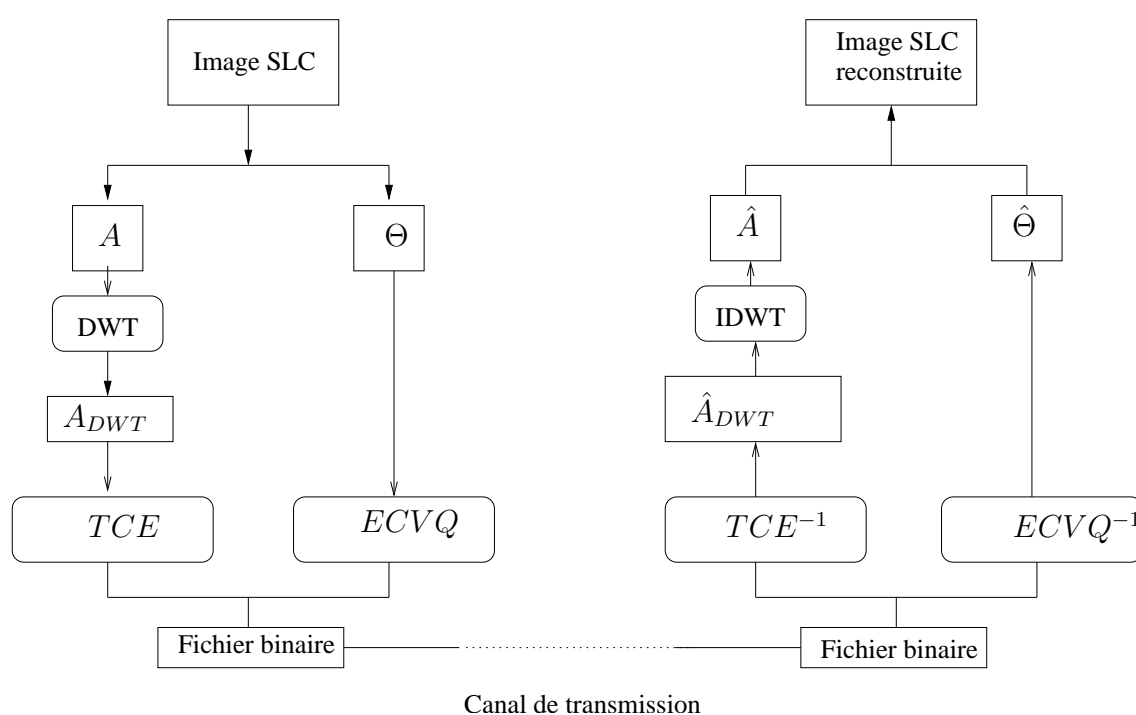


FIG. 3.3 – TCEAC-IPQ : Principe de la compression d'une image SLC à partir d'une quantification indépendante du module et de la phase. L'image d'amplitude est encodée par un algorithme de quantification des coefficients d'ondelettes (SPIHT ou TCE). La phase est quantifiée vectoriellement.

3.2.4 Approche 4 : Représentation polaire avant transformée et compression de la phase dépendante du module

Si l'on souhaite se placer dans le cadre d'une minimisation de l'erreur quadratique moyenne du pixel à valeur complexe, la formule (3.3) montre que la distorsion $D_{\Theta/A}$ associée à la phase est pondérée par un terme dépendant du niveau d'amplitude. Ce terme est d'autant plus important que le pixel est de forte énergie.

La minimisation de ce terme implique que la phase d'un pixel de forte amplitude (point brillant par exemple) doit être quantifiée avec une dégradation plus faible que la phase d'un pixel de faible amplitude (zone d'ombre par exemple) de manière à "compenser" l'effet du terme de pondération. Ce choix peut paraître arbitraire mais il est cohérent avec les quantificateurs polaires non restreints où la phase des pixels d'amplitude élevée est quantifiée avec une distorsion réduite.

Dans le cas de figure qui nous intéresse, l'image d'amplitude peut être comprimée par des algorithmes spécifiques. De manière analogue à la méthode TCEAC-IPQ proposée précédemment, nous utiliserons l'algorithme TCE pour compresser l'image d'amplitude.

Concernant la phase de l'image, nous proposons une méthode dénommée CSPQ (*Conditional Scalar Phase Quantization*) où les données de phase sont quantifiées scalairement et **conditionnellement** au niveau d'amplitude du pixel, de manière à minimiser la fonction de distorsion associée au terme $D_{\Theta/A}$. Cette méthode diffère de l'approche précédente où la quantification de la phase est réalisée indépendamment de toute considération sur le niveau d'amplitude du pixel.

Un argument supplémentaire permet de justifier les fondements de cette approche : la construction d'un interférogramme multivue (formule 1.22) consiste à moyenner le produit hermitien sur une fenêtre de pixels. Dans le repère complexe, ceci correspond par conséquent à réaliser une marche aléatoire par une sommation des vecteurs complexes. Intuitivement, il est naturel de penser que plus le module d'un vecteur particulier de cette somme sera élevé, plus ce dernier influera sur la direction globale du vecteur somme. En d'autres termes, la phase approximative du vecteur somme sera fortement influencée par la phase des vecteurs de forte énergie. Ce raisonnement est illustré par la figure 3.4. Il paraît donc naturel de privilégier une distorsion plus faible concernant la quantification de la phase des pixels de forte amplitude.

La phase étant quantifiée conditionnellement à l'amplitude du pixel, il est évident que le quantificateur et le déquantificateur de phase doivent avoir pour argument une valeur d'amplitude **identique**. Le décodeur n'ayant accès qu'à la version quantifiée de l'image d'amplitude, il convient donc de baser la quantification de la phase une fois que l'image d'amplitude compressée a été **reconstruite**. Le chapitre 4 est en partie consacré à la construction et à l'optimisation d'un tel quantificateur.

La figure 3.5 illustre le principe de ce codeur que l'on nommera **TCEAC-CSPQ** (*TCE Amplitude Coding and Conditional Scalar Phase Quantization*).

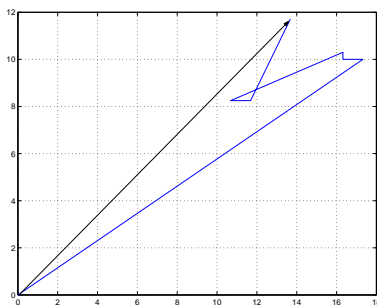


FIG. 3.4 – Illustration de l'importance de la phase en fonction du niveau d'amplitude des vecteurs formant une marche aléatoire. Le trait bleu représente un ensemble de vecteurs dans le plan complexe. Le vecteur noir représente la somme des vecteurs élémentaires. La phase du vecteur somme (i.e l'argument) est beaucoup plus dépendante de la phase des vecteurs de forte amplitude que de la phase des vecteurs de faible amplitude.

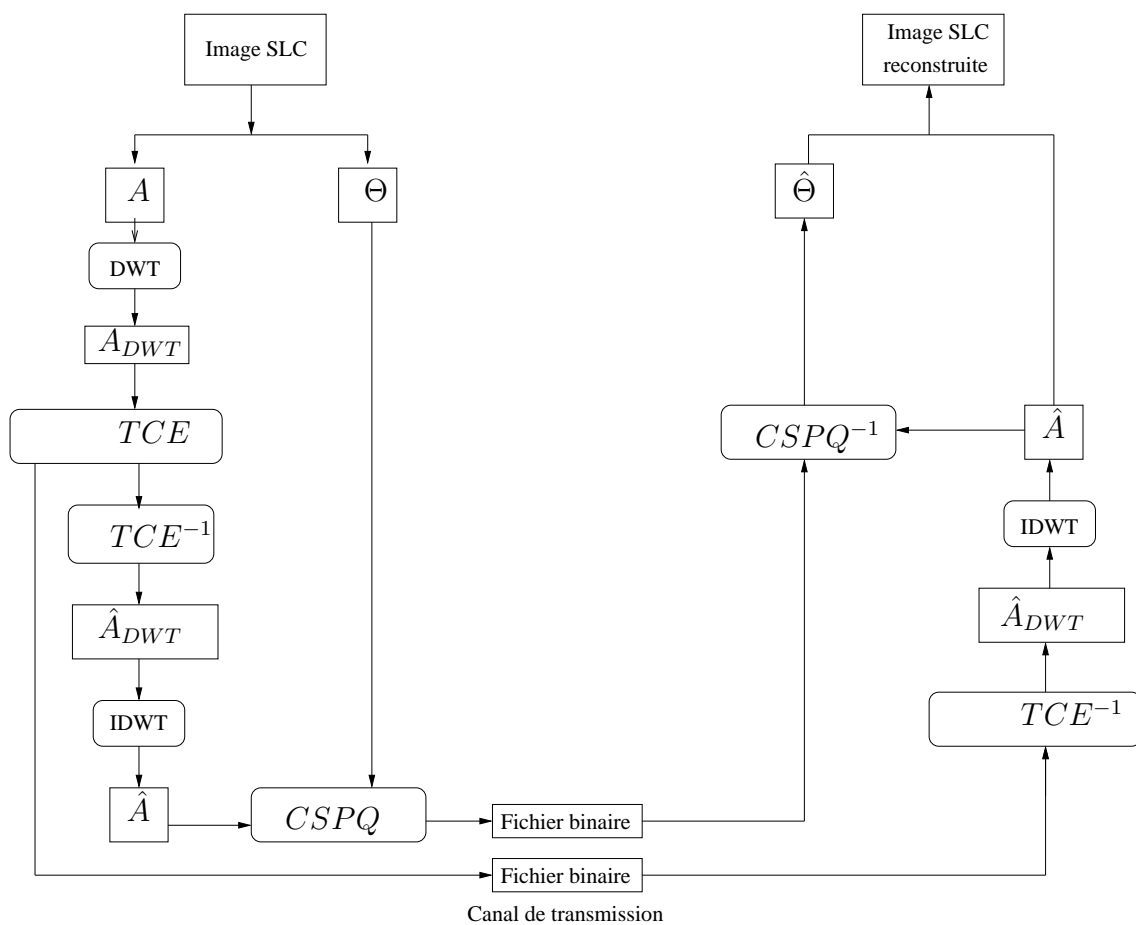


FIG. 3.5 – TCEAC-CSPQ : Principe de compression de l'image d'amplitude indépendamment de la phase par l'algorithme TCE. L'image d'amplitude reconstruite permet de quantifier scalairement la phase en fonction du module du pixel reconstruit (CSPQ).

3.2.5 Conclusion

Nous avons proposé 4 méthodes de compression d'image basée sur la transformée en ondelettes et la représentation des données à adopter pour l'algorithme de quantification. Afin de pouvoir construire les quantificateurs adaptés aux diverses sources à quantifier, il est nécessaire par la suite de connaître avec précisions leur densité de probabilité. Nous nous attacherons en particulier à caractériser dans la section suivante :

- La d.d.p. des données d'amplitude de l'image SLC, nécessaire à la construction du quantificateur de phase conditionnellement à l'amplitude de l'algorithme TCEAC-CSPQ.
- La d.d.p. du module et de la phase des coefficients d'ondelettes intervenant dans l'algorithme WECUPQ.

3.3 Loi de Fisher pour les données SLC et caractérisation avant/après DWT

Nous avons introduit au chapitre 1, une présentation succincte des modèles utilisés pour caractériser le chatoiement pleinement et non pleinement développé de l'image d'intensité (amplitude au carré). La distribution de Fisher ne présente pas de lien direct avec la physique de propagation des ondes mais demeure un outil mathématique qui modélise de manière adéquate plusieurs zones d'images RSO détectées (intensité, amplitude).

La souplesse de cette distribution permet ainsi d'approcher une large palette de distributions comme la loi Gamma, Gamma Inverse, Log-Normale ou encore loi K classiquement utilisées comme modèles associés à l'amplitude ou l'intensité de l'image.

En complément des distributions de type Gaussienne Généralisée ou Laplacienne classiquement retenues pour modéliser la distribution des coefficients d'ondelettes à valeurs dans \mathbb{R} [57], nous verrons également que la distribution de Fisher permet de caractériser la distribution du module des coefficients d'ondelette à valeurs dans \mathbb{C} .

L'intérêt de cette approche est de pouvoir définir un modèle fiable de distribution qui sera ensuite utilisé pour optimiser la quantification des coefficients suivant la minimisation de la distorsion quadratique .

L'estimation des paramètres de la loi de Fisher est cependant délicate voire impossible par les méthodes statistiques classiques (maximum de vraisemblance, méthodes des moments). La solution proposée par J.M. Nicolas consiste à utiliser les statistiques de seconde espèce définies à partir de la transformée de Mellin [69].

3.3.1 Caractérisation statistique de l'image d'amplitude

Loi de chatoiement et transformée de Mellin

Le modèle gaussien circulaire du chatoiement abordé au chapitre 1 suppose que la zone observée est homogène tant à l'échelle du pixel qu'à l'échelle de la fenêtre d'analyse. Nous avons vu alors que l'image en intensité suivait une loi Gamma généralisée dont l'expression est donnée par ($\mu = 1$ une fois normalisé) :

$$S(x) = \frac{L}{\Gamma(L)} (Lx)^{L-1} e^{-Lx} \quad L \geq 0 \quad (3.8)$$

Si la zone observée n'est pas homogène, il convient de prendre en compte la texture de la scène sous-jacente. Connaissant la loi $G(x|\mu)$ correspondant à la zone homogène (x étant la variable aléatoire d'amplitude ou intensité), et la loi de texture sous-jacente $P(\mu)$ de paramètre μ , on peut écrire la loi suivie par les pixels de l'image à partir de la formule de Bayes :

$$Q(x) = \int_0^{+\infty} G(x|\mu) P(\mu) d\mu \quad (3.9)$$

La loi du chatolement peut alors s'écrire $G(x|\mu) = \frac{1}{\mu}S(\frac{x}{\mu})$ avec S correspondant à la loi de probabilité normalisée ($\mu = 1$). On retrouve alors la caractérisation du bruit de type multiplicatif. La distribution $Q(x)$ s'écrit alors :

$$Q(x) = \int_0^{+\infty} S(\frac{x}{\mu})P(\mu)\frac{d\mu}{\mu} = (S\hat{*}P) \quad (3.10)$$

où $\hat{*}$ représente la convolution de Mellin ou convolution "multiplicative". L'opérateur de transformée de Mellin noté \mathcal{M} a pour propriété :

$$\mathcal{M}(Q) = \mathcal{M}(S\hat{*}P) = \mathcal{M}(S).\mathcal{M}(P) \quad (3.11)$$

A l'instar de la première fonction caractéristique de première espèce définie par la transformée de Fourier de la fonction de densité de probabilité, la première fonction caractéristique de seconde espèce est définie par la transformée de Mellin \mathcal{M} . Il s'agit d'une fonction définie sur \mathbb{R}^+ . Son expression est donnée par [76] :

$$\Phi_x(s) = \int_0^{+\infty} u^{s-1}p_x(u)du \quad (3.12)$$

où s est un nombre complexe appelé variable de Mellin. Pour $s = 1$, on retrouve l'expression $\int_0^{+\infty} p_x(u)du = 1$.

Les lois utilisées en imagerie RSO permettant de modéliser le chatolement et la texture sous-jacente ont des transformées de Mellin déjà calculées et tabulées. On peut supposer que la texture sous-jacente suit une loi Gamma de paramètre μ et M telle que :

$$P(x) = \frac{1}{\Gamma(M)} \frac{M}{\mu} \left(\frac{Mx}{\mu}\right)^{M-1} e^{-\frac{Mx}{\mu}} \quad (3.13)$$

A partir du modèle de chatolement distribué suivant une loi Gamma telle que $S \sim \mathcal{G}[1, L]$, on montre alors que l'image RSO en intensité suit une loi K largement utilisée en imagerie RSO et d'expression :

$$\mathcal{K}[\mu, L, M](x) = \mathcal{G}[1, L](x)\hat{*}\mathcal{G}[\mu, M](x) = \frac{1}{\Gamma(L)\Gamma(M)} \frac{2LM}{\mu} \left(\frac{LMx}{\mu}\right)^{\frac{L+M}{2}-1} K_{M-L} \left[2 \left(\frac{LMx}{\mu}\right)^{\frac{1}{2}} \right] \quad (3.14)$$

où K est la fonction de Bessel modifiée de seconde espèce.

On peut également utiliser une loi de scène sous-jacente dite à "queue lourde" comme la loi Gamma Inverse \mathcal{IG} dont la d.d.p est donnée par :

$$P(x) = \frac{1}{\Gamma(M)} \frac{1}{M\mu} \left(\frac{Mx}{\mu}\right)^{M+1} e^{-\frac{Mx}{\mu}}, \quad M \geq 0 \quad (3.15)$$

En conservant le modèle de chatolement suivant la loi Gamma, on en déduit par convolution de Mellin (formule 3.10) la d.d.p de la loi de Fisher à 3 paramètres :

$$\mathcal{F}[\mu, L, M](x) = \mathcal{G}[1, L](x) \hat{*} \mathcal{IG}[\mu, M](x) = \frac{\Gamma(L+M)}{\Gamma(L)\Gamma(M)} \frac{L}{M\mu} \frac{\left(\frac{Lx}{M\mu}\right)^{L-1}}{\left(1 + \frac{Lx}{M\mu}\right)^{L+M}} \quad (3.16)$$

Comme nous l'avons mentionné dans le chapitre 1, cette loi issue d'une "combinaison" d'une loi Gamma à "tête lourde" et d'une loi Gamma Inverse à "queue lourde" semble bien adaptée expérimentalement au tissu urbain mais également à la caractérisation du chatoisement en zone homogène. Il est cependant difficile d'estimer expérimentalement les paramètres d'une telle loi à partir des statistiques classiques de première espèce. Il convient donc de poursuivre l'analyse basée sur la transformée de Mellin à partir de laquelle il est possible de définir les statistiques de seconde espèce (ou log-statistiques).

Statistiques de seconde espèce

Pour une variable aléatoire x à valeur dans \mathbb{R} et de densité de probabilité $p_x(x)$, la première fonction caractéristique est par définition et si celle-ci existe, sa transformée de Fourier :

$$F_x(v) = \int_{-\infty}^{+\infty} e^{jvu} p_x(u) du$$

A partir des propriétés bien connues de la transformée de Fourier, on en déduit une définition équivalente des moments classiques d'ordre n :

$$m_n = \int_{-\infty}^{+\infty} u^n p_x(u) du = (-j)^n \left. \frac{d^n F_x(v)}{dv^n} \right|_{v=0} \quad (3.17)$$

En appliquant une démarche identique pour les variables à valeurs dans \mathbb{R}^+ , on définit les moments de seconde espèce (ou log-moments) [23] à l'aide de la fonction caractéristique de seconde espèce donnée par la transformée de Mellin $\Phi_x(s)$ de la densité de probabilité (formule 3.12) :

$$\tilde{m}_n = \int_0^{+\infty} (\log x)^n p(x) dx = \left. \frac{d^n \Phi_x(s)}{d s^n} \right|_{s=1} \quad (3.18)$$

On définit également la seconde fonction caractéristique de seconde espèce :

$$\psi_x(s) = \log(\Phi_x(s)) \quad (3.19)$$

A l'aide des propriétés de la transformée de Mellin, la seconde fonction caractéristique de seconde espèce permet d'obtenir les cumulants de seconde espèce ou log-cumulants :

$$\tilde{\kappa}_r = \left. \frac{d^r \psi(s)}{d s^r} \right|_{s=1} \quad (3.20)$$

Les relations entre log-moments et log-cumulants sont similaires à celles obtenues avec les moments et cumulants de première espèce [26] :

$$\begin{aligned} \tilde{\kappa}_1 &= \tilde{m}_1 \\ \tilde{\kappa}_2 &= \tilde{m}_2 - \tilde{m}_1^2 \\ \tilde{\kappa}_3 &= \tilde{m}_3 - 3\tilde{m}_1\tilde{m}_2 + 2\tilde{m}_1^3 \end{aligned}$$

3. Vers une méthodologie de compression adaptée à la représentation des données complexes

En appliquant la définition 3.20, le tableau 3.1 donne les 3 premiers log-cumulants concernant les distributions de type Gamma $\mathcal{G}[\mu, L]$, Gamma Inverse $\mathcal{IG}[\mu, M]$ et Fisher $\mathcal{F}[\mu, L, M]$.

Loi	Paramètres	$\tilde{\kappa}_1$	$\tilde{\kappa}_2$	$\tilde{\kappa}_3$
\mathcal{G}	$\mu, L \quad L \geq 0$	$\log(\mu) + \Psi(L) - \log(L)$	$\Psi(1, L)$	$\Psi(2, L)$
\mathcal{IG}	$\mu, M \quad M \geq 0$	$\log(\mu) - \Psi(M) + \log(M)$	$\Psi(1, M)$	$-\Psi(2, L)$
\mathcal{F}	μ, L, M	$\log(\mu) + (\Psi(L) - \log(L)) - (\Psi(M) - \log(M))$	$\Psi(1, L) + \Psi(1, M)$	$\Psi(2, L) - \Psi(2, M)$

TAB. 3.1 – Expression des 3 premiers log-cumulants correspondant aux loi Gamma, Gamma Inverse et Fisher. Ψ est la fonction Digamma [29] et $\Psi(n, L)$ la fonction n -Polygamma [29].

L'estimation empirique des 3 premiers log-cumulants se déduit des 3 équations précédentes. En considérant N échantillons x_i , les estimateurs ont pour expression :

$$\begin{aligned}
 \hat{\kappa}_1 &= \frac{1}{N} \sum_{i=1}^N \log x_i \\
 \hat{\kappa}_2 &= \frac{1}{N} \sum_{i=1}^N (\log x_i)^2 - \left(\frac{1}{N} \sum_{i=1}^N \log x_i \right)^2 \\
 \hat{\kappa}_3 &= \frac{1}{N} \sum_{i=1}^N (\log x_i)^3 - \frac{1}{N^2} \left(\sum_{i=1}^N \log x_i \right) \left(\sum_{i=1}^N (\log x_i)^2 \right) + 2 \frac{1}{N^3} \left(\sum_{i=1}^N \log x_i \right)^3
 \end{aligned} \tag{3.21}$$

Diagramme log-cumulants et caractérisation de la loi de Fisher

Il existe un système de classification des densités de probabilité du système de Pearson obtenu à partir du diagramme $(\beta_1 - \beta_2)$ où $\beta_1 = \frac{m_3^2}{m_2^3}$ correspond au biais (*skewness* en anglais) et $\beta_2 = \frac{m_4}{m_2^2}$ à l'aplatissement (*curtosis* en anglais). Ces expressions sont dérivées des moments centrés classiques m_n d'ordre n .

Malheureusement pour certaines distributions comme la loi de Fisher à 3 paramètres, les moments classiques ne sont pas définis à partir de l'ordre $M + 1$ et conduisent à des expressions ayant une certaine "lourdeur mathématique", ce qui ne les rend pas facilement exploitables.

J.M. Nicolas [69] a ainsi proposé une représentation suivant le diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$ comme le montre la figure 3.6. Ce diagramme permet de caractériser une distribution en fonction de l'ensemble décrit par les log-cumulants associées à la loi de probabilité recherchée.

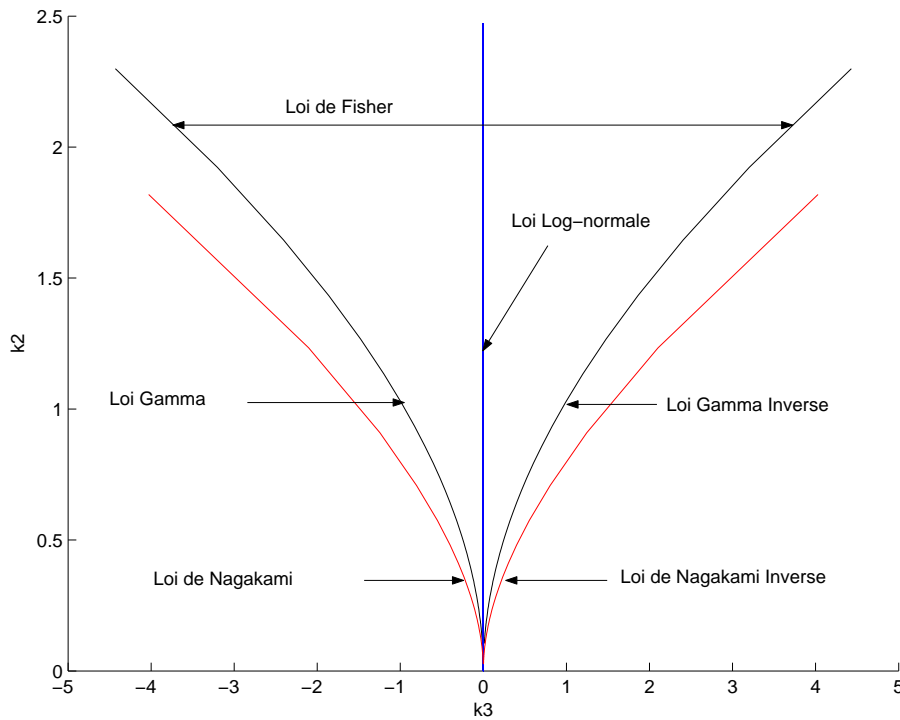


FIG. 3.6 – Diagramme Log-cumulants $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$. A partir des ensembles décrits par les log-cumulants, il est possible de caractériser les distributions classiquement utilisées en imagerie RSO.

Les log-cumulants caractéristiques des lois Log-Normale, K, Gamma, Gamma Inverse, Rayleigh-Nagakami ne couvrent qu'une partie du diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$. En revanche, l'ensemble des log-cumulants de la loi de Fisher à 3 paramètres couvre un espace plus large dont les limites correspondent aux log-cumulants de lois Gamma et Gamma Inverse. Ceci conforte le potentiel de cette densité de probabilité et sa capacité à se conformer empiriquement à plusieurs modèles de distributions classiques.

Estimation des paramètres

La prise en compte d'une loi pour la texture conduit à des distributions pouvant avoir 3 paramètres comme c'est le cas pour la loi de Fisher. L'estimation des paramètres peut se faire de plusieurs manières mais de nombreux problèmes surviennent quant aux différentes méthodes classiquement utilisées.

- La méthode du maximum de vraisemblance est souvent inapplicable en particulier pour les lois K et Fisher [69] car les expressions n'existent pas.
- Les moments de la loi de Fisher $\mathcal{F}[\mu, L, M]$ d'ordre $M + 1$ ne sont pas définis, ce qui rend difficile l'utilisation de la méthode des moments.

En revanche, la méthode des log-moments semble pouvoir s'utiliser dans de nombreux cas et en particulier pour la loi de Fisher. Le tableau 3.1 montre ainsi clairement que les log-cumulants d'ordre 2 et 3 ne dépendent que du paramètre de forme L ou M des lois Gamma ou Gamma Inverse. Cette dépendance est néanmoins implicite. En revanche, la monotonie des fonctions polygamma Ψ permet de résoudre numériquement des équations car il n'existe pas d'expression analytique [26].

La méthode des Log-moments appliquée au modèle de Fisher $\mathcal{F}[\mu, L, M]$ nécessite ainsi la résolution numérique d'un système de deux équations à deux inconnues (\hat{L} et \hat{M}) à partir des log-cumulants $\{\hat{\kappa}_2, \hat{\kappa}_3\}$ empiriquement estimés sur une région de l'image. D'après les équations 3.21 et les expressions du tableau 3.3.1, le système à résoudre est donné par :

$$\begin{cases} \hat{\kappa}_2 = \Psi(1, \hat{L}) + \Psi(1, \hat{M}) \\ \hat{\kappa}_3 = \Psi(2, \hat{L}) - \Psi(2, \hat{M}) \end{cases}$$

La méthode numérique proposée par J.M. Nicolas [69] est la suivante :

1. Initialiser les valeurs L_o et M_o :

$$\Psi(1, \hat{L}_o) = \frac{\hat{\kappa}_2}{2} \quad \Psi(1, \hat{M}_o) = \frac{\hat{\kappa}_3}{2}$$

2. Choisir $\hat{L} < \hat{M}$

3. Itérer jusqu'à convergence :

- calculer $\Psi(2, L_i) - \Psi(2, M_i)$
- incrémenter M_i selon le gradient de $\hat{\kappa}_3$: on obtient M_{i+1} .
- calculer L_{i+1} tel que $\Psi(1, L_{i+1}) + \Psi(1, M_{i+1}) = \hat{\kappa}_2$.

La convergence de cet algorithme est très rapide. Connaissant les valeurs estimées \hat{L} et \hat{M} , le log-cumulant $\hat{\kappa}_1$ permettra de déterminer le paramètre $\hat{\mu}$ par résolution de l'équation :

$$\hat{\kappa}_1 = \log(\hat{\mu}) + (\Psi(\hat{L}) - \log(\hat{L})) - (\Psi(\hat{M}) - \log(\hat{M}))$$

Distribution de Fisher Généralisée

Concernant la loi de Fisher, le diagramme Log-cumulants ($\tilde{\kappa}_2 - \tilde{\kappa}_3$) montre que la limite de caractérisation est donnée par l'ensemble des log-cumulants $\hat{\tilde{\kappa}}_2$ et $\hat{\tilde{\kappa}}_3$ associés aux distributions Gamma et Gamma Inverse. Ainsi, il n'est pas possible d'approcher la distribution d'une variable distribuée suivant une loi de Rayleigh-Nagakami à l'aide d'une distribution de Fisher. En effet, comme le montre la figure 3.6, les log-cumulants d'ordre 2 et 3 associés à la loi de Rayleigh-Nagakami se trouvent en dehors de la zone couverte par les log-cumulants de la loi de Fisher.

Nous cherchons donc dans un premier temps à modifier ou à généraliser la distribution de Fisher de manière à "élargir" l'espace de caractérisation de cette loi. Soit u une variable aléatoire distribuée suivant la loi de Fisher $u \sim \mathcal{F}[\mu, L, M]$. Il est tout à fait possible de définir une variable v correspondant à une puissance quelconque de u et telle que $v = g(u) = u^\alpha$. Cette transformation peut être vue comme une "généralisation" de la loi de Fisher suivant l'approche de McDonald [60].

Par changement de variable, on exprime ainsi facilement le passage de la d.d.p $p(u)$ à $p_\alpha(v)$:

$$p_\alpha(u) = \frac{d g^{-1}(u)}{d u} p(g^{-1}(u)) = \frac{1}{\alpha} u^{(\frac{1}{\alpha}-1)} \cdot p(u^{\frac{1}{\alpha}}) \quad (3.22)$$

En remplaçant $p(u)$ par l'expression de la d.d.p de la loi Fisher (formule 3.16) dans l'équation (3.22), on obtient la loi de Fisher Généralisée à 4 paramètres $\mathcal{FG}[\mu, L, M, \alpha]$:

$$p_\alpha(x) = \mathcal{FG}[\mu, L, M, \alpha](x) = \frac{1}{\alpha} x^{(\frac{1}{\alpha}-1)} \cdot \frac{\Gamma(L+M)}{\Gamma(L)\Gamma(M)} \frac{L}{M\mu} \frac{(\frac{Lx^{\frac{1}{\alpha}}}{M\mu})^{L-1}}{(1 + \frac{Lx^{\frac{1}{\alpha}}}{M\mu})^{L+M}} \quad (3.23)$$

A priori, aucune expression générale simple ne peut être obtenue avec les fonctions caractéristiques de première espèce. Cependant, à partir des propriétés suivantes de la transformée de Mellin, il est possible d'exprimer les relations entre les premières fonctions caractéristiques de seconde espèce :

$$\begin{aligned} \mathcal{M}[x^a p(x)](s) &= \Phi(s+a) \\ \mathcal{M}[p(x^a)](s) &= \frac{1}{a} \Phi\left(\frac{s}{a}\right) \end{aligned}$$

où $\Phi(s) = \mathcal{M}[p(x)](s)$ est la transformée de Mellin de la fonction $p(x)$

A partir des deux propriétés précédentes, on en déduit alors la relation suivante :

$$\Phi_\alpha(s) = \Phi(\alpha s + 1 - \alpha) \quad (3.24)$$

avec $\Phi_\alpha(s) = \mathcal{M}[p_\alpha(x)](s)$ et $\Phi(s) = \mathcal{M}[p(x)](s)$.

L'expression de la seconde fonction caractéristique de seconde espèce est obtenue par dérivation logarithmique de l'expression (3.24). Nous pouvons alors en déduire la relation entre les log-cumulants $\tilde{\kappa}_{p_\alpha, n}$ d'ordre n associés à la distribution de Fisher Généralisée et les log-cumulants de la loi de Fisher $\tilde{\kappa}_{p, n}$:

$$\tilde{\kappa}_{p_\alpha, n} = \alpha^n \cdot \tilde{\kappa}_{p, n} \quad (3.25)$$

A partir de l'expression des log-cumulants associée à la loi de Fisher (cf formules du tableau 3.3.1), on en déduit :

$$\begin{aligned}
 \tilde{\kappa}_{p\alpha,1} &= \alpha \cdot [\log(\mu) + (\Psi(L) - \log(L)) - (\Psi(M) - \log(M))] \\
 \tilde{\kappa}_{p\alpha,2} &= \alpha^2 \cdot [\Psi(1, L) + \Psi(1, M)] \\
 \tilde{\kappa}_{p\alpha,3} &= \alpha^3 \cdot [\Psi(2, L) - \Psi(2, M)]
 \end{aligned}
 \tag{3.26}$$

où Ψ est la fonction Digamma [29] et $\Psi(n, L)$ la fonction n -Polygamma [29].

L'ensemble des log-cumulants de la loi de Fisher à 4 paramètres couvre un espace dépendant de la valeur du paramètre α et dont les limites correspondent aux caustiques obtenues pour $M \rightarrow +\infty$ et $L \rightarrow +\infty$. La figure 3.7 représente ainsi l'espace de caractérisation de cette loi dans le diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$. Cette généralisation de la loi Fisher permet ainsi d'accroître la capacité à approcher empiriquement plusieurs modèles de distributions classiques, et en particulier les lois d'amplitude (en choisissant $\alpha = 0.5$).

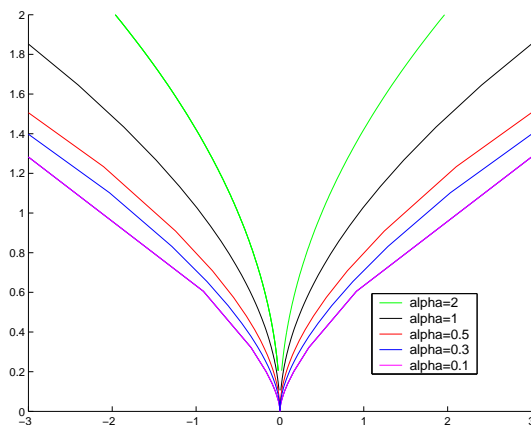


FIG. 3.7 – Ensemble des log-cumulants de la loi de Fisher Généralisée dans le diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$. Le domaine de caractérisation s'élargit en fonction de la décroissance du paramètre α .

La loi Fisher Généralisée permet ainsi de retrouver en particulier :

- La loi de Fisher pour $\alpha = 1$:
 $\mathcal{F}[\mu, L, M] \equiv \mathcal{FG}[\mu, L, M, \alpha = 1]$.
- La loi Gamma pour $\alpha = 1$ et $M \rightarrow +\infty$:
 $\mathcal{G}[\mu, L] \equiv \mathcal{FG}[\mu, L, M \rightarrow +\infty, \alpha = 1]$.
- La loi Gamma Inverse pour $\alpha = 1$ et $L \rightarrow +\infty$:
 $\mathcal{IG}[\mu, M] \equiv \mathcal{FG}[\mu, L \rightarrow +\infty, M, \alpha = 1]$.
- La loi de Rayleigh-Nagakami pour $\alpha = 0.5$ et $M \rightarrow +\infty$:
 $\mathcal{RN}[\mu, L] \equiv \mathcal{FG}[\mu, L, M \rightarrow +\infty, \alpha = 0.5]$.

- La loi de Rayleigh-Nagakami Inverse pour $\alpha = 0.5$ et $L \rightarrow +\infty$:
 $\mathcal{RN}\mathcal{I}[\mu, M] \equiv \mathcal{FG}[\mu, L \rightarrow +\infty, M, \alpha = 0.5]$.

Estimation des paramètres

En observant le diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$, on peut remarquer que le système 3.26 admet une solution tant que les log-cumulants estimés empiriquement sur les données appartiennent à l'espace de caractérisation. Nous utilisons ainsi la méthode suivante :

1. Fixer une valeur de α telle que $|\hat{\kappa}_3| < \tilde{\kappa}_{p\alpha,3}$ et $\hat{\kappa}_2 > \tilde{\kappa}_{p\alpha,2}$. Cette condition stipule simplement qu'il faut choisir α tel que les cumulants empiriques d'ordre 2 et 3 appartiennent à la zone de caractérisation de la loi Fisher Généralisée.
2. Trouver μ, L, M vérifiant le système (3.26). La résolution de ce système se fait de manière analogue à la méthode décrite précédemment pour les paramètres de la loi de Fisher.

Modèles de loi de Fisher et Fisher Généralisée sur les données d'amplitude

Bien qu'une étude plus approfondie soit abordée par C.Tison [99] quant à la validation du modèle de Fisher sur divers types de textures extraites d'images RSO monovues détectées, nous apporterons au lecteur ici quelques illustrations sur la modélisation des données d'amplitude (ou intensité) par une distribution de Fisher ou Fisher Généralisée et dont les paramètres ont été estimés par la méthode des Log-moments.

Les figures 3.8, 3.9, 3.10 représentent les histogrammes mesurés sur les zones analysées ainsi que les fonctions de densité de probabilité des lois de Fisher et Rayleigh-Nagakami estimées suivant la méthode des Log-cumulants. On observe ainsi que le modèle de Fisher est proche de la loi réelle des données d'amplitude.

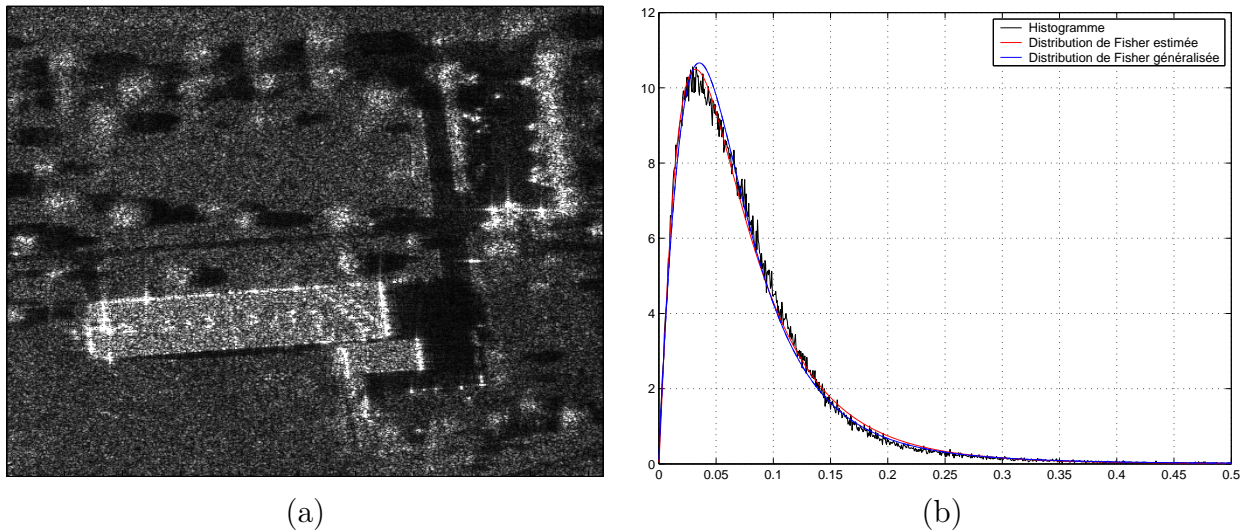


FIG. 3.8 – (a) Image d'amplitude extraite de la ville d'Istres et acquise par le capteur RAMSES (512x512). ©ONERA. (b) Histogramme de l'image et modélisation par une distribution de Fisher (en rouge). Paramètres : $\mu = 0.067, L = 2.200, M = 7.492$. La courbe bleue représente l'estimation par la distribution de Fisher Généralisée. Paramètres : $\mu = 0.010, L = 1.1344, M = 2.075, \alpha = 0.6$.

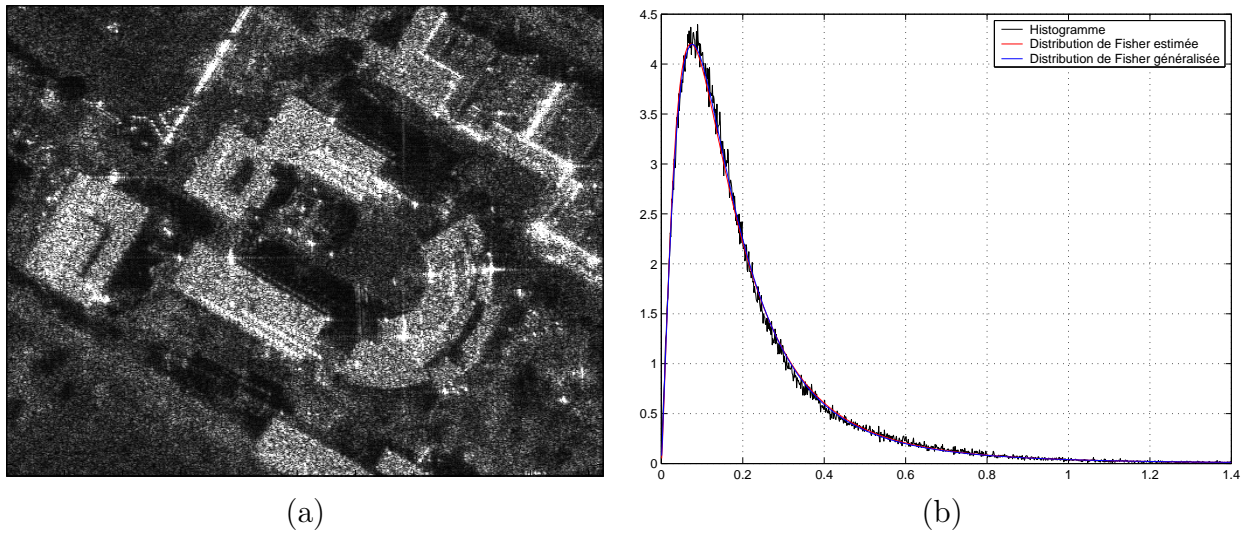


FIG. 3.9 – (a) Image d’amplitude extraite de la ville de Hyères et acquise par le capteur RAMSES (512x512). ©ONERA. (b) Histogramme de l’image et modélisation par une distribution de Fisher. Paramètres : $\mu = 0.158, L = 2.394, M = 3.702$. La courbe bleue représente l’estimation par la distribution de Fisher Généralisée. Paramètres : $\mu = 0.024, L = 0.9070, M = 1.115, \alpha = 0.5$.

La figure 3.11 projette sur le diagramme $(\tilde{\kappa}_2 - \tilde{\kappa}_3)$ l’ensemble des Log-cumulants estimés empiriquement sur les images 3.8, 3.9 et 3.10. Les zones analysées correspondent à des zones de végétation ou des zones urbaines contenant des bâtiments et plusieurs forts réflecteurs. On peut remarquer que les log-cumulants estimés sur la dernière image n’appartiennent pas au domaine de caractérisation par la loi Fisher. En introduisant le paramètre $\alpha = 0.5$ de la loi de Fisher Généralisée, il est alors possible de caractériser la distribution des pixels de cette image.

En résumé, la loi de Fisher modélise fidèlement la distribution des images RSO d’amplitude (ou intensité). L’approche des statistiques de seconde espèce a permis de fournir les expressions simples et facilement exploitables des Log-cumulants associées aux lois à valeurs dans \mathbb{R}^+ . La méthode des Log-moments dérivées des statistiques de seconde espèce donne ainsi la possibilité d’avoir une estimation fiable des paramètres de la loi de Fisher. Dans le cadre de la compression d’image RSO complexe, nous avons proposé une quantification de l’information de phase directement dépendante de l’amplitude du pixel correspondant (méthode TCEAC-CSPQ). Nous utiliserons en particulier le modèle de Fisher afin d’optimiser la construction du quantificateur scalaire de la phase conditionnellement à l’amplitude. L’étude détaillée de cet algorithme sera présentée au chapitre 4 de ce manuscrit.

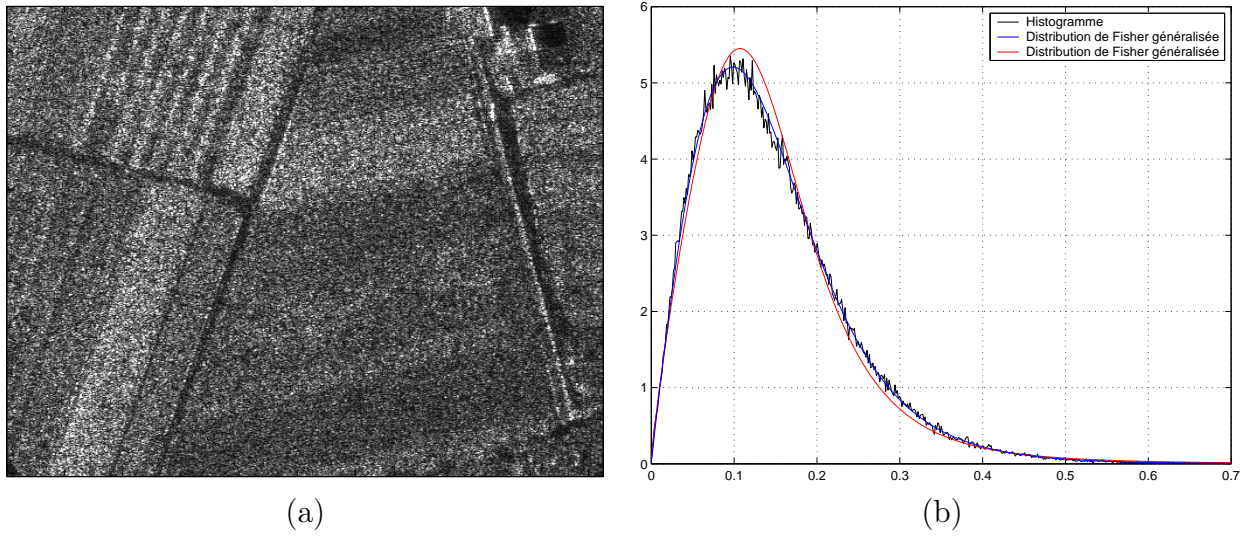


FIG. 3.10 – (a) Image d’amplitude d’une zone agricole issue du capteur RAMSES (512x512). ©ONERA. (b) Histogramme de l’image et modélisation par une distribution de Fisher Généralisée. Paramètres : $\mu = 0.022$, $L = 0.978$, $M = 3.862$, $\alpha = 0.5$ (courbe en bleu). Paramètres : $\mu = 0.016$, $L = 0.547$, $M = 1.1759$, $\alpha = 0.3$ (courbe en rouge).

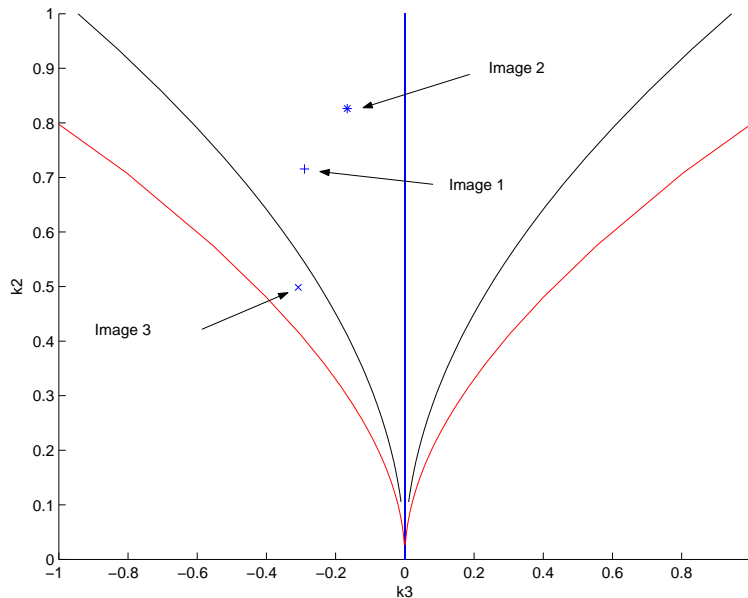


FIG. 3.11 – Log-cumulants estimés sur les images des figures 3.83.9 et 3.10. La caractérisation de l’amplitude par la loi de Fisher est possible pour les images d’Istres et Hyères. En revanche, pour la dernière image, il convient d’élargir le domaine de caractérisation par l’intermédiaire de la loi de Fisher Généralisée.

3.3.2 Modèle de Fisher appliqué après DWT de l'image SLC

La méthode de compression WECUPQ proposée dans le cadre de ce chapitre consiste à réaliser une transformée en ondelettes sur les parties réelle et imaginaire de l'image SLC. Les coefficients ainsi obtenus sont de nature complexe, ce qui implique qu'il est possible de les représenter sous forme polaire avec un module et un argument.

Afin de construire un quantificateur polaire adapté aux sous-bandes de la DWT, il est important dans un premier temps de fournir un modèle de distribution fiable sur le module et la phase des coefficients d'ondelettes. La figure 3.12 montre ainsi le module des coefficients d'ondelettes de l'image d'Istres suivant la représentation pyramidale.

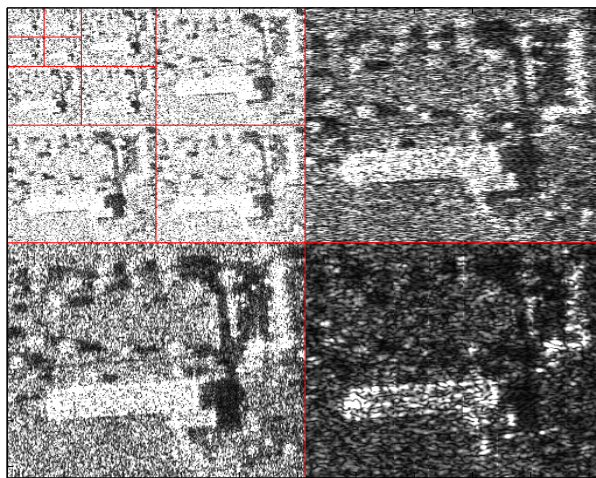


FIG. 3.12 – Ensemble des sous-bandes représentant le module des coefficients d'ondelettes issus d'une DWT sur 4 niveaux d'échelles de l'image SLC d'Istres ©ONERA. (Ondelette bi-orthogonale 9/7 [5]).

Afin de caractériser les distributions du module des sous-bandes, nous suggérons d'utiliser les propriétés de modélisation de loi de Fisher. Les figures 3.13 et 3.14 montrent en particulier que le modèle de distribution de Fisher semble caractériser fidèlement la distribution du module des coefficients des sous-bandes issues d'une transformée par ondelettes biorthogonales 9/7. On retrouve aussi le fait que la phase des coefficients d'ondelettes est uniformément distribuée entre $[0 : 2\pi]$.

La figure 3.15 montre en particulier la distribution des coefficients issus d'une transformée orthogonale et symétrique par une ondelette de Daubechies à valeurs complexes SDW4 (voir chapitre 2). Une fois encore, le modèle de Fisher semble être bien adapté à la caractérisation statistique du module des coefficients. On remarquera que la phase est toujours uniformément répartie sur $[0, 2\pi]$. Conformément aux conclusions énoncées par Ives [43], l'utilisation d'ondelettes de nature complexe sur les données SLC ne semble pas apporter de différence notable, du moins en terme de comportement statistique.

D'autre part, une transformée en ondelettes homomorphique de l'image d'amplitude permet de considérer le bruit multiplicatif associé au chatoiement comme un bruit additif. Quelques méthodes de réduction de chatoiement basées sur un seuillage des coefficients ondelettes s'appuient sur ce modèle [3] [116] [70]. C. Valade et J.M Nicolas ont ainsi montré que la loi de Fisher correspondait à la caractérisation des coefficients d'ondelettes associés aux sous-bandes de l'image RSO d'amplitude [104]. Citons également les travaux récents

de L. Guillemot et al [40] qui modélisent de façon efficace la distribution du module des coefficients d'ondelettes à l'aide d'une mixture de lois Gamma.

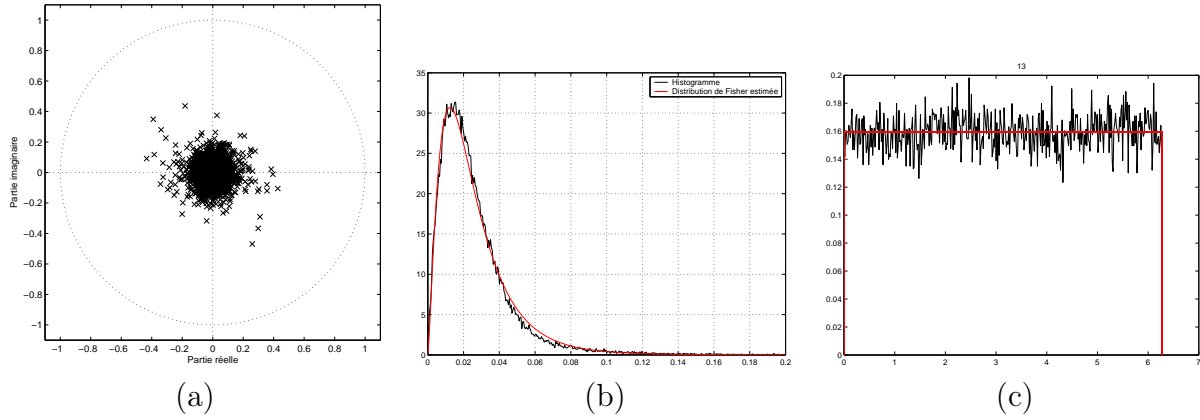


FIG. 3.13 – (a) Distribution dans le plan complexe des coefficients d'ondelettes appartenant à la sous-bande HH_1 (image SLC extraite de la ville d'Istres). (b) Histogramme du module des coefficients d'ondelettes et estimation de la distribution par la loi de Fisher. Paramètres : $\mu = 0.023$, $L = 2.444$, $M = 9.407$. (c) Histogramme de la phase des coefficients d'ondelettes. Transformée biorthogonale 9/7.

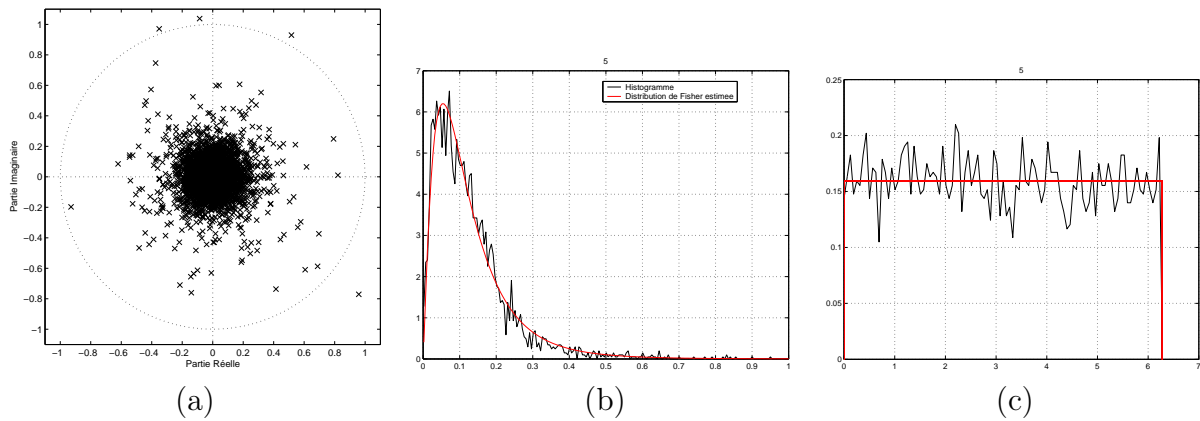


FIG. 3.14 – (a) Distribution dans le plan complexe des coefficients d'ondelettes appartenant à la sous-bande HL_3 (image SLC extraite de la ville d'Istres). (b) Histogramme du module des coefficients d'ondelettes et estimation de la distribution par la loi de Fisher. Paramètres : $\mu = 0.112$, $L = 2.225$, $M = 7.077$. (c) Histogramme de la phase des coefficients d'ondelettes. Transformée biorthogonale 9/7.

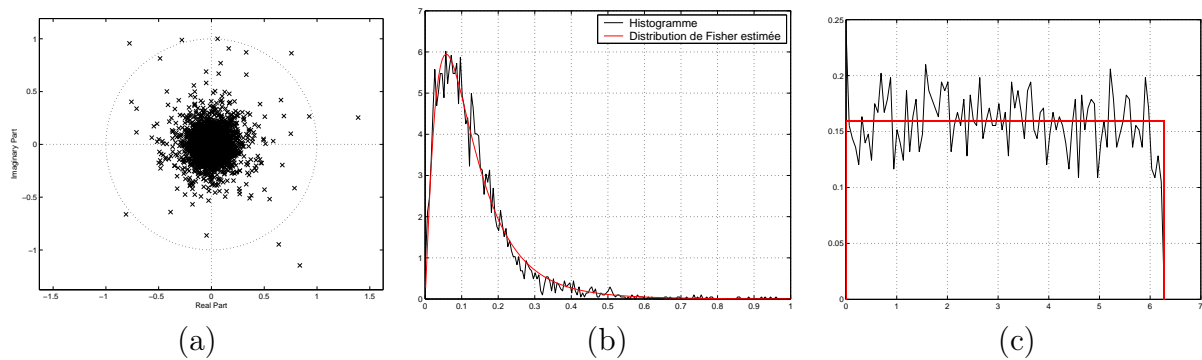


FIG. 3.15 – (a) Distribution dans le plan complexe des coefficients d’ondelettes appartenant à la sous-bande HL_3 (image SLC d’Istres). (b) Histogramme du module des coefficients d’ondelettes et estimation de la distribution par la loi de Fisher. Paramètres : $\mu = 0.118$, $L = 2.225$, $M = 7.865$. (c) Histogramme de la phase des coefficients d’ondelettes. Transformée orthogonale SDW4 (ondelette complexe).

3.4 Conclusion

Ce chapitre nous a permis de faire la synthèse entre les propriétés des images complexes RSO et les méthodes de compression que l'on souhaite mettre en oeuvre. La compression d'images complexes à partir de techniques basées sur la transformée en ondelettes requiert une représentation adéquate des données dans l'espace image et/ou dans l'espace transformé. Nous avons ainsi proposé quatre approches différentes basées sur une représentation cartésienne ou polaire des données dans l'espace "image" ou "transformé". Nous proposons entre autres, d'étudier les performances des algorithmes SPIHT de Said et Pearlman et TCE de C. Tian et S.S Hemami sur la compression d'images RSO SLC. Les algorithmes TCEAC-CPQ et WECUPQ s'appuient respectivement sur une connaissance des distributions associées aux pixels de l'image d'amplitude et aux coefficients d'ondelettes complexes. A cet effet, la dernière partie de ce chapitre fut consacrée à l'étude des propriétés de caractérisation des lois de Fisher et Fisher Généralisée à travers l'utilisation des statistiques de seconde espèce.

A partir de la démarche observée au cours de ce chapitre, nous consacrons le chapitre suivant à la description détaillée des méthodes de compression proposées.

3. Vers une méthodologie de compression adaptée à la représentation des données complexes

Chapitre 4

Compression d'images RSO complexes basée sur la transformée en ondelettes

Ce chapitre s'inscrit directement dans la continuité du chapitre 3. Nous y avons proposé plusieurs algorithmes dépendants de la représentation des données des images RSO SLC dans le plan de complexe. Ces techniques ont en commun de faire appel à la transformée en ondelettes discrètes. Au cours de ce chapitre nous étudierons plus précisément :

- La quantification des coefficients d'ondelettes appliquée sur la partie réelle et imaginaire de l'image SLC transformée : TCE-IQC (*TCE and In Phase and Quadrature Coding*).
- La quantification polaire à contrainte entropique des coefficients d'ondelettes complexes : WECUPQ (*Wavelet Entropy Constrained Unrestricted Polar Quantization*).
- La compression de l'image d'amplitude par l'algorithme TCE de C. Tian et S. Hernani [98] couplée à la quantification scalaire de l'image de phase conditionnellement à l'amplitude : TCEAC-CSPQ (*TCE Amplitude Coding and Conditionnal Scalar Phase Quantization*).
- La compression de l'image d'amplitude par l'algorithme TCE associé à la quantification vectorielle de l'image de phase : TCEAC-IPQ (*TCE Amplitude Coding and Independant Phase Quantization*).

4.1 Compression de l'image SLC par la méthode TCE-IQC

Cette section est consacrée à la description détaillée de l'algorithme de quantification de coefficients d'ondelettes TCE de C. Tian et S. Hemami [98]. Dans le cadre de la compression de données d'images SLC par la méthode TCE-IQC présentée au chapitre 3, l'algorithme TCE sera appliqué respectivement sur la partie réelle et imaginaire de l'image transformée.

4.1.1 Généralités

De nombreuses méthodes initialement développées pour des images naturelles ont été appliquées à la compression d'images RSO. Citons notamment les travaux de J. Mvogo [65], G. Mercier [61] ou D. Gleich et al [37]. Ces schémas de compression diffèrent principalement sur la stratégie de quantification des coefficients d'ondelettes (arbres de zéros, quantification scalaire, vectorielle, contextuelle ..). Nous avons choisi dans le cadre de cette thèse de nous intéresser à une méthode de compression basée sur la transformée en ondelettes et dérivée de l'algorithme TCE de Chao Tian and Sheila S. Hemami [98]. Cet algorithme dispose ainsi des avantages suivants :

- Un codage progressif *fractionnaire* des coefficients d'ondelettes par plan de bits.
- Une faible complexité de calcul obtenue grâce à une implémentation efficace d'un filtre récursif réalisant une estimation fiable de la probabilité des symboles sur l'état "significatif" ou non des coefficients d'ondelettes.
- Les performances obtenues en termes de PSNR et d'erreur quadratique moyenne sont proches de l'état de l'art, notamment de celles obtenues par l'algorithme JPEG2000 pour les images optiques.

Codage progressif par plan de bits

Nous avons brièvement décrit cette méthode au chapitre 2. Un codage progressif génère un train binaire qui peut être tronqué à n'importe quel point tout en étant toujours décodable. Soit l'ensemble des coefficients d'ondelettes $\{c[x]\}$ situés à $x = [x_1, x_2] \in \mathcal{Z}^2$.

On introduit $b_n[x]$ comme l'état "significatif" (*significant*) ou "non significatif" (*unsignificant*) d'un coefficient associé à un seuil t :

$$b_{n,x} = \begin{cases} 1 & \text{si } |c[x]| > t \\ 0 & \text{sinon} \end{cases}$$

Pour correspondre à la représentation binaire, le seuil t est choisi comme une puissance de 2. L'ensemble des valeurs $\{b_n\}$ forme une image binaire du n-ième plan de bits. Par divisions dyadiques successives du seuil t , on introduit ainsi le plan de bits $\{b_{n-1}\}$ et ainsi de suite. La décomposition binaire de $c[x]$ est alors donnée par $b_{n,x}, b_{n-1,x}, \dots, b_{n-i,x}$. Si le bit de poids fort appartient au n-ième plan de bit ($b_{n,x} = 1$) alors les bits suivants $b_{n-1,x}, \dots, b_{n-i,x}$ correspondent aux bits de *raffinage*.

A partir du moment où un coefficient $c(x)$ devient significatif au plan de bits n , l'information complémentaire à transmettre est le signe $s_{n,x}$:

$$s_{n,x} = \text{sign}(c[x]) \quad \text{si } b_{n,x} = 1$$

La table 4.1 illustre la quantification par plan de bits pour une valeur particulière d'un coefficient ainsi que la reconstruction progressive à mesure que les états $b_{n,x}$ sont "lus" par le décodeur :

Plan de bit	Seuil t	Coefficient	Symbole émis	Valeur décodée
n	256	212,55	$b_{n,x} = 0$	0
n-1	128	212,55	$b_{n-1,x} = 1 \quad s_{n-1,x} = 1$	$(+)1,5 * t = (+)192$
n-2	64	84,55	$b_{n-2,x} = 1$	$192 + t/2 = 224$
n-3	32	20,55	$b_{n-3,x} = 0$	$224 - t/2 = 208$
n-4	16	20,55	$b_{n-4,x} = 1$	$208 + t/2 = 216$
n-5	8	4,55	$b_{n-5,x} = 0$	$216 - t/2 = 212$
n-6	4	4,55	$b_{n-6,x} = 1$	$212 + t/2 = 214$
n-7	2	0,55	$b_{n-7,x} = 0$	$212 - t/2 = 213$
n-8	1	0,55	$b_{n-8,x} = 0$	$213 - t/2 = 212,5$

TAB. 4.1 – Exemple d'une quantification par plan de bits

Comme le montre la figure 4.1, les plans de bits sont généralement transmis en parcourant les coefficients de chaque sous-bande suivant un balayage de type *Raster Scan*.

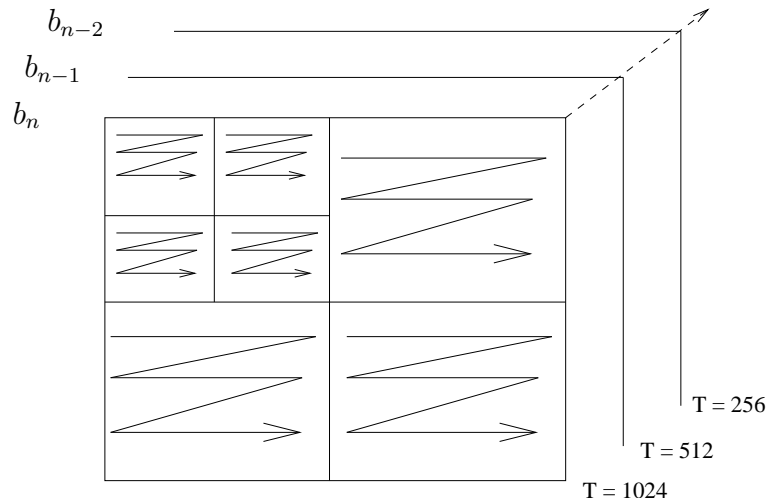


FIG. 4.1 – Quantification des sous-bandes par plan de bits et balayage *Raster Scan*

L'information de quantification est donc portée par la séquence $s_{n,x}, b_{n,x}, b_{n-1,x}, \dots, b_{n-i,x}$ si le coefficient devient significatif au n -ième plan de bits (i.e $b_{n,x} = 1$).

Pour chaque coefficient $c[x]$, on peut donc considérer l'état $b_{n,x}$ comme une variable aléatoire discrète à valeurs dans $\{0, 1\}$ associée à une probabilité $p = \text{Prob}(b_{n,x} = 1)$. Le codeur progressif TCE utilise cette probabilité pour piloter un codeur entropique de type arithmétique. Le calcul d'entropie d'ordre 0 permet d'estimer le coût de codage de ces symboles $\Delta R_{n,x}$.

$$\begin{aligned}\Delta R_{n,x} &= -\log_2(p) \quad \text{si } b_{n,x} = 1 \\ \Delta R_{n,x} &= -\log_2(1-p) \quad \text{si } b_{n,x} = 0\end{aligned}$$

L'encodage d'un nouveau symbole $b_{n,x}$ associé à la représentation binaire de $c[x]$ implique une réduction de la distorsion notée $\Delta D_{n,x}$. Selon Ordenlich et al [22], un codeur progressif dit *fractionnaire* peut être optimisé en transmettant d'abord les symboles offrant le meilleur compromis débit-distorsion, c'est-à-dire les symboles avec la pente $\lambda = \frac{\Delta D_{n,x}}{\Delta R_{n,x}}$ la plus élevée. En d'autres termes, il s'agit de transmettre en priorité les symboles $b_{n,x}$ associés à la plus forte décroissance de distorsion moyennant un coût de codage le plus faible possible.

D'une manière générale, les probabilités concernant les symboles de signe $s_{n,x}$ sont proches de 0.5. On ne peut donc pas s'attendre à un gain significatif de codage entropique pour ce type de symboles bien que Wu [110] mentionne la possibilité d'exploiter le contexte (voisinage) du symbole à encoder afin de mieux prédire le comportement de la carte binaire des signes. Il en est de même pour les symboles de raffinement.

En revanche, les symboles associés à l'état significatif ou non d'un coefficient d'ondelettes b_n ont un comportement bien plus prévisible. La forte concentration des coefficients d'ondelettes autour de la valeur nulle implique que seul un faible nombre de coefficients sera significatif sur les premiers plans de bits (i.e $Prob(b_{n,x} = 1) \ll Prob(b_{n,x} = 0)$ pour n élevé). D'un point de vue entropique, on peut donc espérer un gain substantiel au codage.

Une solution simple consiste alors à encoder les symboles associés au caractère significatif d'un coefficient avant les bits de raffinement. C. Tian et S. Hernani ont ainsi proposé l'algorithme TCE suivant cette approche.

4.1.2 Algorithme TCE

Ordenlich et al [22] ont introduit une classification des coefficients basée sur l'information obtenue du codage des plans de bits précédents. Le principe de cette classification consiste à regrouper des coefficients au comportement similaire. Cela permet de mieux prédire et estimer la probabilité qu'un coefficient soit significatif par rapport au seuil t .

En considérant le seuil courant t correspondant au plan de bits n , l'algorithme TCE s'inspire de cette approche en introduisant 3 états des coefficients :

1. Les coefficients de type NZN (en anglais *Non Zero Neighbours*). Il s'agit des coefficients non-significatifs ayant au moins un voisin significatif aux plans de bits précédents. Soit N_x l'ensemble des 8 voisins du pixel x (8 connexités). La condition d'appartenance à la classe NZN s'écrit donc :

$$c[x] \in NZN \Leftrightarrow \begin{cases} b_{n+i,x} = 0 & i \geq 1 \\ \forall y \in N_x, \exists z \in N_x \text{ tel que } b_{n+i,z} = 1 & i \geq 1 \end{cases}$$

On suppose que ces coefficients étiquetés NZN ont une probabilité non négligeable d'être significatifs à ce plan de bits puisque l'un de leur voisin l'a au moins été à l'un des plans de bits précédents.

L'encodage des symboles $b_{n,x}$ binaire est réalisé par un codeur arithmétique adaptatif piloté par une table de probabilité avec $p = Prob(b_{n,x} = 1)$. L'estimation de cette probabilité dépend de la fréquence empirique des symboles.

2. Les coefficients non-significatifs et qui n'ont pas encore de voisins significatifs par rapport aux plans de bits précédents (en anglais *RC : Run coefficients*). Il s'agit principalement de régions à faible dynamique (absence de contours par exemple).

$$c[x] \in Run\ Coefficients \Leftrightarrow \begin{cases} b_{n+i,x} = 0 & i \geq 1 \\ \forall y \in N_x, b_{n+i,y} = 0 & i \geq 1 \end{cases}$$

L'estimation de la probabilité $p = Prob(b_{n,x} = 1)$ correspond à la méthode d'estimation non paramétrique basée sur les fenêtres de Parzen [28]. Dans le cadre 1-D, cela revient à utiliser un filtre récursif d'ordre 1. La technique a été généralisée au cadre bi-dimensionnel par P. Simard et al [84] à l'aide d'un filtre récursif 2-D "TARP" construit à partir de 3 filtres mono-dimensionnels. L'avantage de cette méthode réside dans sa capacité à estimer de manière efficace la probabilité $p = Prob(b_{n,x} = 1)$ sur les régions à faible dynamique (zones d'ombres par exemple). L'encodage des symboles $b_{n,x}$ est réalisé par un codeur arithmétique non-adaptatif.

3. Les coefficients qui sont devenus significatifs à l'un des plans de bits précédents. Il s'agit par conséquent des coefficients en phase de raffinement (*Refinement Coefficients*).

$$c[x] \in Refinement\ Coefficients \Leftrightarrow b_{n+i,x} = 1 \quad i \geq 1$$

La probabilité $p = Prob(b_{n,x} = 1)$ permettant de piloter un codeur arithmétique adaptatif est mise à jour suivant la fréquence empirique des symboles de raffinement. En pratique cette probabilité reste proche de la valeur $p = 0.5$.

La figure 4.2 illustre cette classification des coefficients d'ondelettes en fonction du plan de bits considéré.

Cette classification permet d'affiner et d'améliorer l'estimation des probabilités $p = Prob(b_{n,x} = 1)$ en regroupant les coefficients par comportement a priori similaire. L'encodage progressif des coefficients par plan de bits fractionnaire se fait donc suivant trois passes correspondant respectivement à chacune des classes des coefficients considérées.

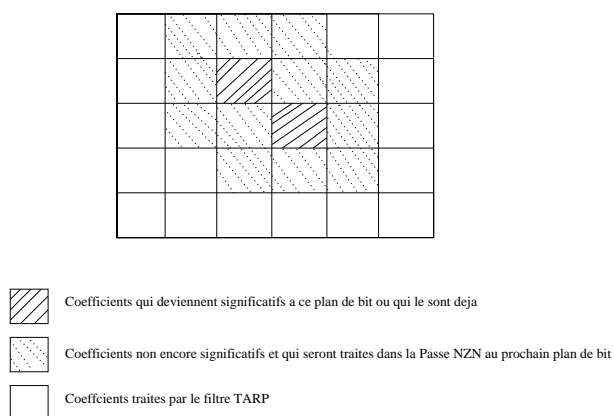


FIG. 4.2 – Exemple de classification des coefficients d'ondelettes par l'algorithme TCE pour un plan de bit donné.

Codage des *Run Coefficients* et filtre TARP

Le codeur basé sur une estimation de la probabilité associée à l'état significatif d'un coefficient par filtrage TARP a été introduit par P. Simard et al [84]. Cette technique a été réutilisée dans l'algorithme TCE concernant la classe des *Run Coefficients*.

Etant donné un plan de bits n et un seuil $t > 0$, nous connaissons la localisation spatiale des coefficients significatifs aux plan de bits précédents $n + i$, $i \geq 1$. D'autre part, un certain nombre de coefficients classés NZN par rapport au plan de bits courant n sont devenus significatifs par rapport à t .

Le principe consiste à utiliser les informations $b_{n,x}$ des coefficients **précédemment traités** afin d'estimer la probabilité qu'un coefficient $c[x]$ localisé en x soit significatif.

Dans le cas mono-dimensionnel, supposons que l'ensemble des coefficients précédemment visités forment une séquence de Bernouilli dont la probabilité $p[x] = \text{prob}(b_{n,x} = 1)$ varie lentement. En introduisant un filtre récursif, on peut estimer la probabilité que le coefficient $c[x + 1]$ soit significatif :

$$p[x + 1] = \alpha.p[x] + (1 - \alpha).b_{n,x} \quad (4.1)$$

où α est un paramètre permettant de s'adapter à la vitesse d'évolution de la probabilité.

L'équation précédente équivaut à filtrer l'ensemble des $b_{n,x}$ avec un filtre à réponse impulsionnelle infinie donnée par :

$$\begin{cases} \phi[x] = \alpha^x.(1 - \alpha) & x \geq 0 \\ 0 & x < 0 \end{cases}$$

En conséquence, $p[x]$ est estimé suivant une somme de fenêtres de Parzen mises en forme par $\phi[x]$. P. Simard et al ont généralisé cette technique d'estimation non paramétrique dans le cadre bi-dimensionnel.

Soit $x \equiv (m, n) \in [0, M - 1] \times [0, N - 1]$ la localisation du pixel dans l'image. L'estimation de la carte de probabilité des *Run Coefficients* est réalisée suivant la formule :

$$p[x] = \sum_{y \in \mathcal{C}_x} \phi(x - y).b_{n,y} \quad (4.2)$$

où \mathcal{C}_x correspond à l'ensemble des pixels précédemment visités par rapport à la position x . Le décodeur fonctionne de la même manière que le codeur mais ne dispose que d'informations causales. En d'autres termes, le décodeur n'a accès qu'aux valeurs $b_{n,x}$ associées aux coefficients $c[x]$ précédemment visités. En prenant en considération le parcours des coefficients suivant un balayage en *Raster Scan*, la figure 4.3 illustre la configuration spatiale de l'ensemble \mathcal{C}_x . La figure 4.4 représente la réponse impulsionnelle du noyau de convolution $\phi[m, n]$ en 2D.

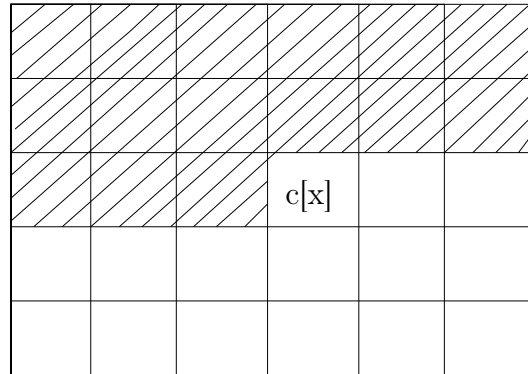


FIG. 4.3 – Traitement causal des coefficients : l'estimation de la probabilité $p[x] = \text{prob}(b_{n,x} = 1)$ dépend des probabilités estimées sur les coefficients précédemment visités suivant le balayage *Raster Scan*.

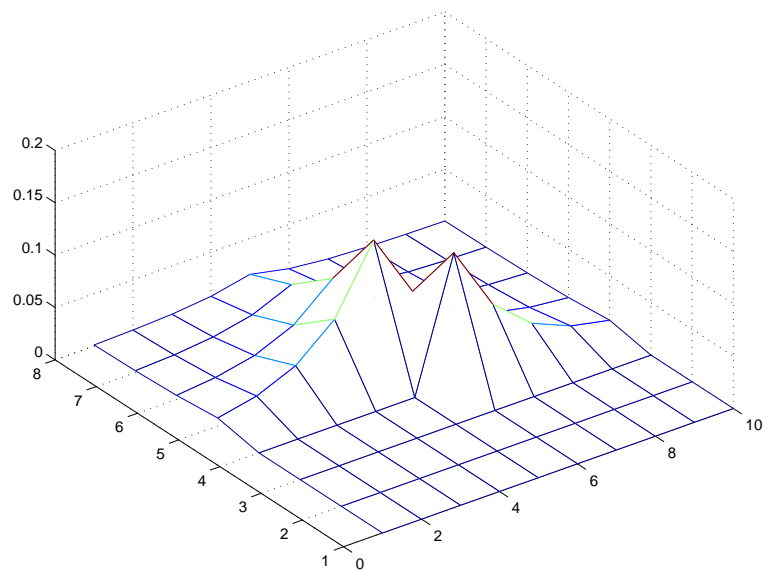


FIG. 4.4 – Noyau de convolution du filtre TARP pour $\alpha = 0.6$

En pratique, cette convolution en 2D est implémentée à l'aide de 3 filtres monodimensionnels appelé filtres Tarp. Le premier filtre balaye les coefficients de gauche à droite, le second de droite à gauche une fois que chaque ligne a été traitée. Le troisième filtre traite les coefficients verticalement. Les équations suivantes correspondent à l'implémentation du filtre :

$$p[x] \equiv p[m, n] = \alpha \left(p_1[m, n-1] + p_2[m-1, n] \right) \quad (4.3)$$

avec

$$\begin{aligned} p_1[m, n] &= \alpha p_1[m, n-1] + \frac{(1-\alpha)^2}{2\alpha} b_{n,c[m,n]} \\ p_2[m, n] &= p_1[m, n] + \alpha p_2[m-1, n] \end{aligned}$$

Le dernier filtre met à jour la table de probabilité $p_2[m, n]$ par un balayage de droite à gauche une fois que la ligne m a été traitée :

$$p_2[m, n] = p_2[m, n] + \alpha p_3[m, n+1]$$

avec

$$p_3[m, n] = \alpha p_3[m, n+1] + \frac{(1-\alpha)^2}{2\alpha} b_{n,c[m,n]}$$

Les étapes de filtrage 1-D permettent ainsi un gain efficace en terme de complexité de calcul et de mémoire. On peut noter que ce filtre a également été adapté au codage de coefficients d'ondelettes dans un cadre 3-D pour la compression d'images hyperspectrales [107].

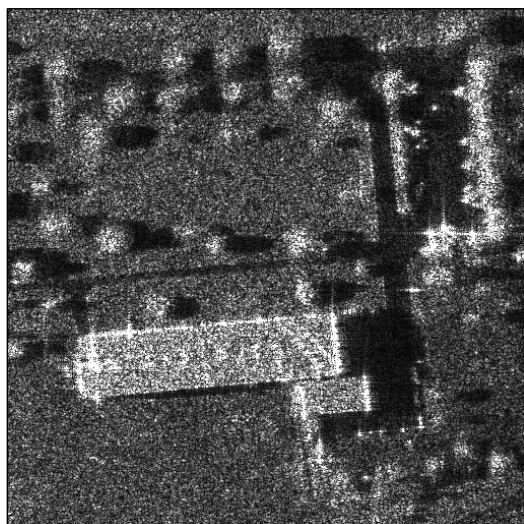
En résumé, la réponse impulsionnelle du filtre Tarp est excitée par les états $b_{n,x}$ $x \in \mathcal{C}_x$ relatifs au caractère significatif des coefficients précédemment traités. La sortie du filtre fournit une estimation de la probabilité $p[x] = \text{prob}(b_{n,x} = 1)$. Cette probabilité sera ensuite utilisée pour encoder le symbole $b_{n,x}$ à l'aide d'un codeur arithmétique non adaptatif. Appliquée à la compression d'images RSO, cette méthode se révèle efficace sur les zones de faible activité (les zones d'ombre, les routes et cours d'eau) où la probabilité $p[x]$ varie faiblement. Pour plus de précisions sur la compression d'images par filtrage TARP, nous suggérons au lecteur de se référer à [84].

4.1.3 Application

Suivant l'approche TCE-IQC que nous proposons, la partie réelle et imaginaire de l'image SLC sont transformées par DWT sur 4 niveaux d'échelle par l'ondelette bi-orthogonale 9/7 [5] puis encodées séparément par l'algorithme TCE. Bien que décorré-ées, les caractéristiques statistiques locales des images de partie réelle et imaginaire sont proches : en terme d'erreur quadratique moyenne, on observe expérimentalement un profil d'évolution de la courbe débit-distorsion similaire entre les deux images. C'est la raison pour laquelle, on alloue pour chaque image un débit identique bien qu'il soit possible d'envisager une allocation de débit optimale entre les sous-bandes issues de la décomposition des partie réelle et imaginaire de l'image. Par exemple, pour un débit global de $R_Z = 2$ bits par pixel complexe (bppc) pour l'image $Z = I + jQ$, les images I et Q sont compressées avec un débit de $R_I = R_Q = 1$ bit par pixel.

Les figures 4.5 illustrent l'image d'amplitude extraite de la ville d'Istres reconstruite à partir des images de partie réelle et imaginaire décompressée à divers débits. L'image a été acquise par le capteur RAMSES de l'ONERA en bande X. Pour des raisons de visualisation, seule l'image d'amplitude reconstruite est représentée car l'image de phase conserve la structure de bruit blanc. Nous renvoyons cependant le lecteur à l'étude comparative du chapitre 6 pour une évaluation plus détaillée de la qualité de l'image dans le domaine complexe.

On observe visuellement une différence notable avec l'image originale pour les faibles débits (1 bit et 2 bits par pixel complexe), notamment en ce qui concerne les zones homogènes de l'image caractérisée par le chatolement pleinement développé (verdure). En effet, les plans de bits transmis en priorité correspondent aux coefficients d'ondelettes de forte amplitude associés aux zones de forte dynamique (points brillants, fort réflecteurs). Le chatolement pleinement développé apparaît dans l'image comme une zone de faible dynamique et dont les coefficients d'ondelettes associés ne seront massivement significatifs qu'à partir de quelques plans de bits.



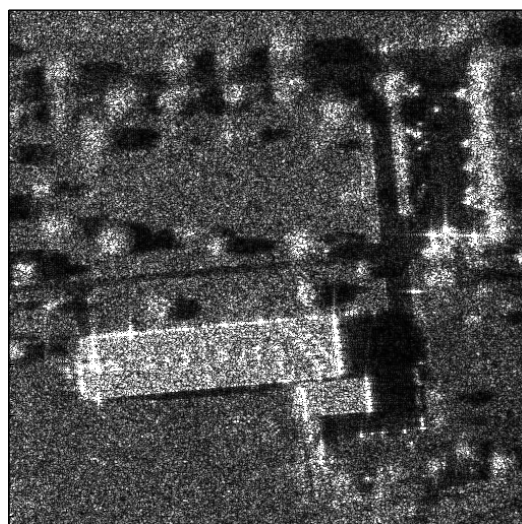
(a) Image RSO SLC originale (amplitude) extraite de la ville d'Istres. ©ONERA.



(b) Taux $\eta = 64 : 1$ (1bppc)
 $RSB_A = 9.25dB$.



(c) Taux $\eta = 32 : 1$ (2bppc),
 $RSB_A = 12.81dB$.



(d) Taux $\eta = 16 : 1$ (4bppc),
 $RSB_A = 18.53dB$.

FIG. 4.5 – Exemple de compression d'image SLC par l'algorithme TCE-IQC. Seule l'amplitude de l'image est représentée.

4.2 Quantification polaire des coefficients d'ondelettes : *Wavelet ECUPQ*

Cet algorithme consiste à effectuer une transformée en ondelettes sur la partie réelle et imaginaire de l'image RSO SLC. Les coefficients de chaque sous-bande b sont complexes et représentés suivant leur module A_b et leur phase Θ_b . Ce type de représentation se prête naturellement à une méthode de quantification polaire et plus précisément à une quantification polaire à contrainte entropique si l'on souhaite introduire un codage entropique des indices de quantification.

4.2.1 Quantification Polaire

La quantification polaire a été étudiée par W.A Pearlman [74] pour un vecteur complexe en terme d'amplitude et de phase. Cette quantification s'applique en règle générale sur des sources dont la partie réelle X et imaginaire Y sont considérées comme des variables aléatoires gaussiennes i.i.d. Ce type de quantification est notamment très utilisé pour la quantification de signaux audio et en particulier de la composante sinusoïdale. Les sinusoïdes sont alors représentées par leur fréquence associée à l'information d'amplitude et de phase. Il existe plusieurs types de quantification polaire :

- La quantification polaire conventionnelle (en anglais : CPQ pour *Conventional Polar Quantization*) consiste à utiliser un quantificateur scalaire uniforme pour la phase et un quantificateur scalaire uniforme pour l'amplitude.
- La quantification polaire non-restreinte (en anglais : UPQ pour *Unrestricted Polar Quantization*), l'amplitude et la phase sont quantifiées par l'intermédiaire de quantificateurs scalaires uniformes. La quantification de l'information de phase est cependant dépendante de l'amplitude.
- La quantification "cartésienne" (en anglais RQ pour *Rectangular Quantization*) consiste simplement à quantifier scalairement la partie réelle et imaginaire. L'intérêt de ces quantificateurs réside entre autres, dans leur simplicité de mise en oeuvre et leur faible complexité. A noter que dans l'espace à dimension 2, une quantification vectorielle par réseau hexagonal A_2 serait plus adaptée qu'une partition rectangulaire.

La figure 4.6 illustre la partition de l'espace complexe en fonction du quantificateur choisi.

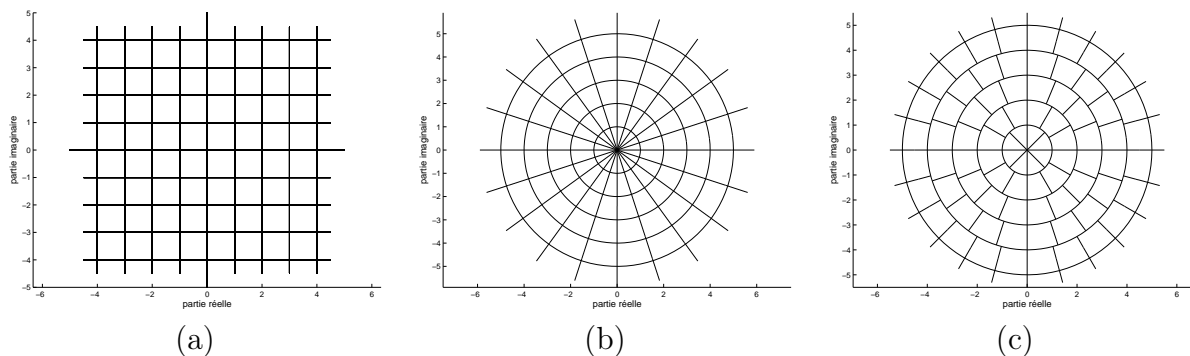


FIG. 4.6 – Illustration de la partition de l'espace complexe. (a) quantification rectangulaire (RQ). (b) quantification polaire conventionnelle (CPQ). (c) quantification polaire non-restreinte (UPQ).

Pour des applications de compression, la contrainte de débit peut se formuler de deux manières : le quantificateur peut être défini en terme de résolution c'est-à-dire qu'un nombre fixe de cellules N de quantification entraînera un codage des symboles à longueur fixe avec un débit $R = \log_2(N)$. Une quantification à contrainte entropique implique l'utilisation d'un code entropique à longueur variable dont le débit convergera vers l'entropie H de la source. Pour une distorsion fixée et à la lumière des résultats énoncés au chapitre 2, une quantification à contrainte entropique permet d'obtenir un débit moyen plus faible. En conséquence, nous nous placerons dans le cadre d'une quantification polaire à contrainte entropique.

Il a été montré par S. G Wilson que pour des sources gaussiennes circulaires, l'erreur quadratique moyenne de la quantification polaire non-restreinte UPQ est inférieure à celle de la quantification polaire conventionnelle CPQ [108]. Les performances de la méthode UPQ sont ainsi bien meilleures que la quantification CPQ et très proches de la quantification rectangulaire RQ si la résolution du codeur est fixe et qu'un codage entropique est appliqué sur les index de quantification.

Les premiers travaux sur la quantification polaire UPQ [108], [88], [68] ne portaient que sur l'optimisation du quantificateur pour un nombre fixe de cellules de quantification et sous l'hypothèse de haute résolution. Plus formellement, une densité de cellules suffisamment élevée permet de considérer que la densité de probabilité des variables aléatoires à quantifier est constante à l'intérieur de la cellule de quantification. Le codage entropique appliqué sur les indices de quantification permet ensuite de réduire le débit moyen [108].

L'approche proposée par R. Vafin et W. Bastiaan [103] se distingue par le fait que la conception du quantificateur UPQ est directement optimisée en fonction de la contrainte entropique. Nous présentons ici les résultats théoriques de la méthode ECUPQ (en anglais *Entropy Constrained Unrestricted Polar Quantization*) à partir desquels nous contruirons notre algorithme.

Quantification Polaire Non-restreinte à Contrainte Entropique.

Soit $Z = X + jY$ un vecteur aléatoire de distribution conjointe circulaire symétrique et $\hat{Z} = \hat{X} + j\hat{Y}$, le vecteur quantifié correspondant. Les couples (A, Θ) et $(\hat{A}, \hat{\Theta})$ forment respectivement le vecteur original et quantifié en représentation polaire avec :

$$\begin{aligned} A &= \sqrt{X^2 + Y^2} \in [0, +\infty[\\ \theta &= \arg(X + jY) \in [0, 2\pi] \end{aligned}$$

La construction du quantificateur ECUPQ [103] se base sur la minimisation de l'erreur quadratique moyenne d'expression (cf equation 3.1) :

$$\begin{aligned} D &= E_Z(|Z - \hat{Z}|^2) \\ &= E_{A,\Theta}(|Ae^{j\Theta} - \hat{A}e^{j\hat{\Theta}}|^2) \\ &= \int_0^{+\infty} (a - \hat{a})^2 p_A(a) da + \int_0^{2\pi} \int_0^{+\infty} 2a\hat{a}(1 - \cos(\theta - \hat{\theta})) p_{\Theta,A}(\theta, a) dad\theta \end{aligned} \tag{4.4}$$

La contrainte de débit H est liée à l'entropie conjointe des indices de quantification du module et de la phase $\{I_A, I_\Theta\}$:

$$H = H(I_A, I_\Theta) = H(I_A) + H(I_\Theta/I_A)$$

où $H(I_A)$ est l'entropie des indices de quantification du module. $H(I_\Theta/I_A)$ correspond à l'entropie conditionnelle des indices de quantification de la phase connaissant l'indice de quantification de l'amplitude.

La connaissance apportée par les indices de quantification d'amplitude permet de réduire l'"incertitude" sur l'information des indices de quantification de la phase. En terme d'entropie, la théorie de l'information nous garantit la relation suivante :

$$H(I_\Theta/I_A) \leq H(I_\Theta)$$

Sous l'hypothèse de haute résolution, il est alors possible d'exprimer la distorsion D en fonction de la densité de probabilité conjointe de la source $f_{A,\Theta}(a, \theta)$ et de la densité de points de quantification du module $g_A(r)$ et de la phase $g_\Theta(\theta, a)$ d'expression :

$$\begin{aligned} g_A(a) &= 1/\Delta_a \\ g_\Theta(a) &= 1/\Delta_{a,\theta} \end{aligned}$$

où Δ_a correspond au pas de quantification du module et $\Delta_{a,\theta}$ au pas de quantification de la phase sachant le niveau d'amplitude du pixel à quantifier.

La distorsion moyenne est alors donnée par [103] :

$$D \approx \int_0^{+\infty} \int_0^{2\pi} f_{A,\Theta}(a, \theta) \left(\frac{g_A^{-2}(a)}{12} + a^2 \frac{g_\Theta^{-2}(\theta, a)}{12} \right) \quad (4.5)$$

L'hypothèse de haute résolution permet également de réécrire la contrainte de débit :

$$\begin{aligned} H(I_A) &\approx h(A) + \int_0^{+\infty} f_A(a) \log_2 g_A(a) da \\ H(I_A/I_\Theta) &\approx h(\Theta) + \int_0^{+\infty} \int_0^{2\pi} f_{A,\Theta}(a, \theta) \log_2 g_\Theta(\theta, a) dad\theta \end{aligned}$$

où $f_A(a)$ correspond à la fonction de densité de probabilité marginale de la variable aléatoire d'amplitude. $h(A)$ et $h(\Theta)$ sont les entropies différentielles de l'amplitude et de la phase définies par :

$$h(A) = - \int_0^{+\infty} f_A(a) \log_2 f_A(a) da \quad (4.6)$$

$$h(\Theta) = - \int_0^{2\pi} f_\Theta(\theta) \log_2 f_\Theta(\theta) d\theta \quad (4.7)$$

où $f_\Theta(\theta)$ est la fonction de densité de probabilité marginale de la variable de phase. En combinant les équations (4.6) et (4.7), la contrainte entropique s'écrit :

$$\begin{aligned} H &= h(A) + h(\Theta) + \int_0^{+\infty} f_A(a) \log_2 g_A(a) da \\ &+ \int_0^{+\infty} \int_0^{2\pi} f_{A,\Theta}(a, \theta) \log_2 g_\Theta(\theta, a) dad\theta \end{aligned} \quad (4.8)$$

Optimisation du quantificateur ECUPQ

L'objectif est de trouver les densités de points de quantification $g_A(a)$ et $g_\Theta(\theta, a)$ minimisant l'erreur quadratique moyenne D sous la contrainte entropique H . Ce problème de minimisation sous contrainte peut être résolu en posant le lagrangien λ . Le critère à minimiser est donné par :

$$J(\lambda) = D + \lambda.H \quad (4.9)$$

où les expressions D et H sont respectivement remplacées par les équations (4.5) et (4.8). Le problème consiste alors à trouver les densités optimales $g_R^*(r)$ et $g_\Theta^*(\theta, r)$ telles que :

$$\begin{aligned} g_A^*(a) &= \operatorname{arg}_{\{g_A(a)\}} \min J(\lambda) \\ g_\Theta^*(\theta, a) &= \operatorname{arg}_{\{g_\Theta(\theta, a)\}} \min J(\lambda) \end{aligned}$$

Les résultats obtenus supposent l'indépendance entre la variable d'amplitude et de phase, ce qui implique : $f_{A,\Theta}(a, \theta) = f_A(a) \cdot f_\Theta(\theta)$. En dérivant $J(\lambda)$ par rapport par $g_A(a)$ et $g_\Theta(\theta, a)$ et en tenant compte de la contrainte afin de déterminer la valeur optimale du multiplicateur de Lagrange, on obtient [103] :

$$\begin{aligned} g_A^*(a) &= g_A = 2^{(H-h(A)-h(\Theta)-b(A))/2} \\ g_\Theta^*(\theta, a) &= g_\Theta(a) = a \cdot 2^{(H-h(A)-h(\Theta)-b(A))/2} \end{aligned} \quad (4.10)$$

où $b(A) = \int f_A(a) \log_2(a) da$.

Le quantificateur optimal en amplitude est uniforme et les niveaux de quantification de la phase dépendent **linéairement** de l'amplitude conformément à la figure 4.6(c). En remplaçant les expressions (4.10) dans (4.5), on en déduit la fonction de débit-distorsion du quantificateur ECUPQ :

$$\begin{aligned} D_{ECUPQ}(H) &= \frac{1}{6} 2^{-(H-h(A)-h(\Theta)-b(A))} \\ &= \frac{1}{6} \cdot h \cdot 2^{-H} \end{aligned} \quad (4.11)$$

avec $h = 2^{h(A)+h(\Theta)+b(A)}$.

Quelques éléments de comparaison

A titre de comparaison, le quantificateur polaire conventionnel (ECCPQ : *Entropy Constrained Conventional Polar Quantization*) dépend des densités de points de quantification $g_A(a)$ et $g_\Theta(\theta)$. La distorsion est minimisée sous la contrainte entropique :

$$H = H(I_A, I_\Theta) = H(I_A) + H(I_\Theta)$$

On obtient alors les densités optimales [103] :

$$\begin{aligned} g_A^* &= \sigma^{(-1/2)}(A) 2^{(H-h(A)-h(\Theta))/2} \\ g_\Theta &= \sigma^{(1/2)}(A) 2^{(H-h(A)-h(\Theta))/2} \end{aligned}$$

où $\sigma^2(A) = \int a^2 f_A(a) da$ est le moment d'ordre 2 de la variable A . Dans ce cas, les quantificateurs optimaux sont uniformes pour l'amplitude et la phase, conformément à la figure 4.6(b).

La distorsion est donnée par :

$$D_{ECCPQ} = \frac{1}{6} \sigma(A) 2^{(H-h(A)-h(\Theta))} \quad (4.12)$$

Le quantificateur rectangulaire haute-résolution (ECRQ : *Entropy Constrained Rectangular Quantization*) est construit à partir des densités de points de quantification $g_X(x)$ et $g_Y(y)$. La minimisation de l'erreur quadratique moyenne dépend de la contrainte entropique :

$$H = H(I_x, I_y) = H(I_x) + H(I_y)$$

où $\{I_x, I_y\}$ correspondent respectivement aux indices de quantification de la partie réelle et imaginaire du pixel complexe.

Les densités optimales sont données par :

$$g_X^* = g_Y^* = 2^{(H-h(X)-h(Y))/2} \quad (4.13)$$

correspondant à la construction d'un quantificateur uniforme pour la partie réelle et imaginaire. La fonction de débit-distorsion associée s'exprime par :

$$D_{ECRQ} = \frac{1}{6} 2^{-(H-h(X)-h(Y))} \quad (4.14)$$

Le quantificateur vectoriel à 2 dimensions sous contrainte entropique (ECVQ) est construit suivant la minimisation de la densité de points de quantification $g_{X,Y}(x, y)$ sous contrainte entropique :

$$H = H(I_{xy}) \quad (4.15)$$

où I_{xy} correspondent aux indices de quantification d'un vecteur dans le plan complexe. La fonction de débit-distorsion est donnée par :

$$D_{ECVQ} = \frac{5}{18\sqrt{3}} 2^{-(H-h(X,Y))}$$

où $h(X, Y)$ est l'entropie différentielle de la vecteur aléatoire (X, Y) .

La figure 4.7 illustre les différentes courbes de débit-distorsion théoriques correspondant aux quantificateurs ECUPQ, ECRQ et ECCPQ et ECVQ. Les variables X et Y prises pour cet exemple sont i.i.d de loi Normale centrée $\mathcal{N}(0, \sigma)$.

L'entropie différentielle est donnée par $h(X) = h(Y) = 0.5 \log_2(2\pi e\sigma^2)$.

Le module R suit alors une loi de Rayleigh dont l'entropie différentielle $h(A) = 0.5 \log_2(0.5e^{2+\gamma}\sigma^2)$ où γ est la constante d'Euler. La phase est indépendante du module et suit une loi de d.d.p uniforme entre $[0, 2\pi]$. L'entropie différentielle de la phase est : $h(\Theta) = \log_2(2\pi)$. De plus, on a $b(A) = 0.5 \log_2(2e^{-\gamma}\sigma^2)$ et $\sigma^2(A) = 2\sigma^2$.

On observe que la quantification par ECUPQ est supérieure en terme de compromis débit-distorsion par rapport à la méthode ECCPQ. En revanche, les performances à haute-résolution des méthodes ECUPQ et ECRQ sont identiques (les courbes se superposent). D'après la figure 4.6, on remarquera à ce propos que la forme des cellules de quantification de la méthode ECUPQ tend à se rapprocher d'une forme carrée. Cette topologie est associée au quantificateur ECRQ.

En outre, il a été montré que les performances entre les quantificateurs UPQ et RQ optimisés par rapport à la résolution (c'est à dire à un nombre N fixe de cellules) ont des performances identiques si un codage entropique est appliqué sur les indices de quantification [108].

La méthode ECVQ donne naturellement le meilleur résultat en comparaison des méthodes scalaires ECUPQ et ECRQ. Néanmoins, les performances sont très proches avec un ratio $\frac{D_{ECUPQ}}{D_{ECVQ}} \approx 1.039$.

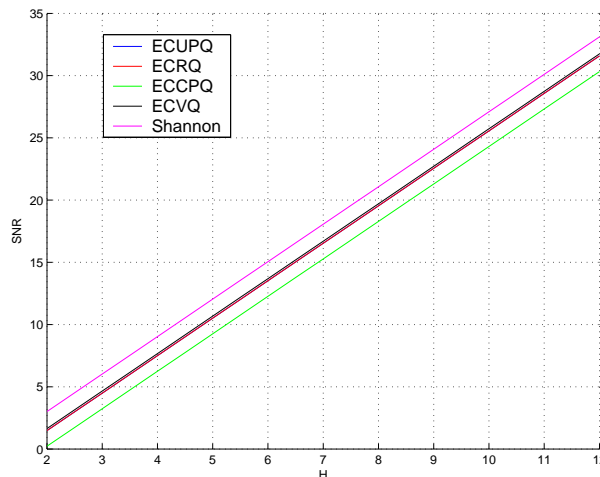


FIG. 4.7 – Performance des quantificateurs polaires suivant l'erreur quadratique moyenne (Rapport Signal sur Bruit) en fonction du débit H (contrainte entropique) pour une variable aléatoire gaussienne circulaire avec $\sigma^2 = 1$.

4.2.2 Quantification polaire à contrainte entropique des coefficients d'ondelettes

Dans le cadre de la méthode WECUPQ, les coefficients d'ondelettes sont à valeurs complexes puisque la transformée en ondelettes est directement réalisée sur l'image SLC. En se plaçant dans le contexte d'une quantification polaire à contrainte entropique, la construction du quantificateur optimal au sens du minimum de l'EQM dépend naturellement de la densité de probabilité des coefficients d'ondelettes (voir équation (4.11)). D'après les résultats obtenus au chapitre 3 (section 3), nous disposons d'un modèle de distribution associé au module et à la phase des coefficients d'ondelettes pour chaque sous-bande. Par conséquent, nous considérerons les coefficients d'ondelettes d'une sous-bande $b \in [0, B - 1]$ comme un ensemble de variables aléatoires i.i.d $C_b = A_b e^{j\Theta_b}$ dont :

- Le module A_b est distribué suivant une loi de Fisher $f_{A_b}(a) \sim \mathcal{F}[\mu_b, L_b, M_b](a)$. Les paramètres μ_b, L_b, M_b de la loi de Fisher sont estimés sur le module des coefficients par la méthode des log-moments suivant l'approche préconisée par J.M Nicolas et développée au chapitre 3.
- La phase Θ_b est indépendante du module et uniformément distribuée sur $[0, 2\pi]$.

Chaque sous-bande étant quantifiée de manière indépendante, il convient de déterminer la répartition du budget binaire global H entre les sous-bandes, de manière à minimiser la distorsion globale notée D_{WECUPQ} . En supposant un codage entropique des indices de quantification, on obtient un débit R_b qui converge vers l'entropie H_b . Déterminer la bonne répartition du budget H entre toutes les sous-bandes revient donc à résoudre un problème d'allocation dynamique qui optimise les débits pour les symboles d'amplitude et de phase

Chaque sous-bande disposant d'un nombre N_b de coefficients d'ondelettes, le nombre de bits total nécessaire pour encoder une sous-bande est de $N_b H_b$ bits avec :

$$\sum_{b=0}^{B-1} N_b = N \quad (4.16)$$

où N correspond au nombre total de coefficients d'ondelettes.

La contrainte de débit s'exprime alors :

$$\sum_{b=0}^{B-1} N_b H_b \leq NH \quad (4.17)$$

En d'autres termes :

$$\sum_{b=0}^{B-1} n_b H_b \leq H \quad (4.18)$$

avec $n_b = \frac{N_b}{N}$.

La distorsion globale moyenne s'exprime en fonction des distorsions moyennes $D_{ECUPQ}(H_b)$ introduites par la quantification de chaque sous-bande b :

$$D_{WECUPQ} = \sum_{b=0}^{L-1} g_b n_b D_{ECUPQ}(H_b) \quad (4.19)$$

où g_b est un facteur de gain correctif lié à la norme L^2 des vecteurs associés aux filtres de synthèse de la DWT [102] [94]. Si les filtres sont orthonormés alors $g_b = 1 \forall b \in [0 : B - 1]$. Ce terme intervient donc pour les transformées biorthogonales.

Le problème se pose de la manière suivante : on recherche l'ensemble $\{H_b\}$ tel que la distorsion globale soit minimale sous la contrainte d'un budget fixé H :

$$H_b^* = \arg \min_{\{H_b\}} \sum_{b=0}^{L-1} g_b n_b D_{ECUPQ}(H_b) \quad (4.20)$$

sous la contrainte posée par l'équation (4.18).

Allocation optimale de débit

A partir des conditions d'optimalité de Kuhn Tucker, le problème (4.20) se résout simplement par une minimisation sans contrainte à partir du lagrangien. L'optimum doit être obtenu pour $H = \sum_{b=0}^{B-1} n_b H_b$:

$$J(\lambda) = \sum_{b=0}^{B-1} g_b n_b D_{ECUPQ}(H_b) + \lambda \left(\sum_{b=0}^{B-1} n_b H_b - H \right) \quad (4.21)$$

Ce qui est équivalent à trouver l'ensemble des $\{H_b^*\} \ b = 0 \dots B - 1$ tel que :

$$H_b^* = \arg \min_{\{H_b\}} J(\lambda) \quad (4.22)$$

Connaissant les modèles de distribution du module et la phase des coefficients, l'expression (4.21) est donc connue. Le Lagrangien est donné en fonction de H_b par :

$$J(\lambda) = \frac{1}{6} \sum_{b=0}^{B-1} g_b n_b h_b 2^{-H_b} + \lambda \left(\sum_{b=0}^{B-1} n_b H_b - H \right) \quad (4.23)$$

En dérivant cette expression par rapport à H_b , on obtient la condition d'optimalité suivante :

$$\frac{\partial J(\lambda)}{\partial H_b} = 0 \quad (4.24)$$

En remplaçant l'expression 4.23 dans l'équation (4.24), on obtient :

$$-\log(2) g_b h_b 2^{-H_b^*} + \lambda = 0 \Rightarrow H_b^* = \log_2 \left(\frac{g_b h_b \log(2)}{\lambda} \right) \quad b = 0 \dots B - 1 \quad (4.25)$$

Il suffit alors de remplacer H_b de l'équation précédente dans la contrainte (4.18) et en appliquant l'égalité. L'allocation optimale de débit est alors donnée par :

$$H_b^* = H + \log_2 \left(\frac{h_b}{\prod_{b=0}^{B-1} h_b^{n_b}} \right) + \log_2 \left(\frac{g_b}{\prod_{b=0}^{B-1} g_b^{n_b}} \right) \quad b = 0 \dots B - 1 \quad (4.26)$$

En substituant l'expression de H_b^* de l'équation (4.26) dans la formule (4.11), l'erreur quadratique moyenne est donnée par :

$$\begin{aligned} D_{WECUPQ} &= \frac{1}{6} \sum_{b=0}^{B-1} n_b h_b g_b 2^{-H_b^*} \\ &= \frac{1}{6} \left(\prod_{b=0}^{B-1} h_b^{n_b} \right) \left(\prod_{b=0}^{B-1} g_b^{n_b} \right) 2^{-H} \end{aligned} \quad (4.27)$$

En particulier pour une transformée en ondelettes orthonormée, on a $g_b = 1$, $b = 0 \dots B - 1$. L'erreur quadratique moyenne s'exprime par :

$$D_{WECUPQ} = \frac{1}{6} \left(\prod_{b=0}^{B-1} h_b^{n_b} \right) 2^{-H} \quad (4.28)$$

Construction du quantificateur

La contrainte de débit optimal $\{H_b^*\}$ $b = 0 \dots B - 1$ étant fixée pour chaque sous-bande de coefficients, la construction du quantificateur s'obtient à partir des densités de points de quantification $g_{A_b}(a)$ et $g_{\Theta_b}(a)$. En combinant l'équation (4.26) avec les expressions (4.10), on obtient dans le cas orthonormé :

$$\begin{aligned} g_{A_b}(a) &= g_A = \frac{2^{\frac{H}{2}}}{\prod_{b=0}^{B-1} h_b^{n_b/2}} \\ g_{\Theta_b}(a) &= g_{\Theta}(a) = a \frac{2^{\frac{H}{2}}}{\prod_{b=0}^{B-1} h_b^{n_b/2}} \quad b = 0 \dots B - 1 \end{aligned} \quad (4.29)$$

Les quantificateurs scalaires associés au module et à la phase ont un pas de quantification qui s'exprime alors sous la forme :

$$\begin{aligned} \Delta_{A_b}(a) &= \Delta_A = 2^{-\frac{H}{2}} \prod_{b=0}^{B-1} h_b^{n_b/2} \\ \Delta_{\Theta_b}(a) &= \Delta_{\Theta}(a) = a^{-1} 2^{-\frac{H}{2}} \prod_{b=0}^{B-1} h_b^{n_b/2} \quad b = 0 \dots B - 1 \end{aligned} \quad (4.30)$$

Le résultat ci-dessus montre que les quantificateurs sont identiques quelles que soient les sous-bandes considérées (résultat asymptotique obtenu à partir de l'hypothèse haute résolution concernant le quantificateur polaire). Le même quantificateur sera donc appliqué à tous les coefficients. En revanche, la probabilité associée aux indices de quantification dépendra directement de la densité de probabilité de la sous-bande considérée. Le codeur arithmétique disposera donc d'une table de probabilité différente d'une sous-bande à une

autre.

Soit $\{I_{A_b}\}$ l'alphabet des indices de quantification du module pour une sous-bande b . Le cardinal $\Omega_{I_{A_b}}$ de cet alphabet est choisi en fonction du rayon de la partition, de manière à ce que la distorsion de saturation du module soit négligeable devant la distorsion granulaire.

La probabilité d'occurrence d'un indice q de l'intervalle de quantification $\mathcal{I}_q = [q \cdot \Delta_{A_b}, (q+1)\Delta_{A_b}[$ s'exprime par :

$$p_{I_{A_b}}(q) = \int_{q \cdot \Delta_{A_b}}^{(q+1) \cdot \Delta_{A_b}} f_{A_b}(a) da \quad q = 0 \dots \Omega_{I_{A_b}} - 1 \quad (4.31)$$

L'alphabet des indices de quantification de la phase sachant le niveau d'amplitude est noté $\{I_{\Theta_b}|I_{A_b}\}$. Le cardinal de cet alphabet dépend naturellement de l'amplitude du coefficient quantifié et donc de l'indice de quantification q . Soit $a_q = E(A_b|A_b \in \mathcal{I}_q)$, le centroïde d'un intervalle de quantification \mathcal{I}_q . Nous introduisons le cardinal $\Omega_{I_{\Theta_b}|I_{A_b}}(q)$ tel que :

$$\Omega_{I_{\Theta_b}|I_{A_b}}(q) = \lceil \frac{2\pi}{\Delta_{\Theta_b}(a_q)} \rceil \quad q = 0 \dots \Omega_{I_{A_b}} - 1$$

La phase des coefficients étant uniformément répartie sur $[0, 2\pi]$ chaque indice de quantification de phase est encodé entropiquement en fonction du rayon d'amplitude $q \cdot \Delta_{A_b}$ correspondant. La probabilité des indices n de quantification sachant q est donnée par :

$$p_{I_{\Theta_b}|I_{A_b}}(n|q) = \frac{1}{\Omega_{I_{\Theta_b}|I_{A_b}}(q)} \quad n = 0 \dots \Omega_{I_{\Theta_b}|I_{A_b}}(q) - 1 \quad \text{et} \quad q = 0 \dots \Omega_{I_{A_b}} - 1 \quad (4.32)$$

Le débit global H_{WECUPQ} théoriquement atteignable après codage entropique des indices de quantification est alors donné par :

$$\begin{aligned} H_{WECUPQ} &= \sum_{b=0}^{B-1} H(I_{A_b}, I_{\Theta_b}) \\ &= \sum_{b=0}^{B-1} H(I_{A_b}) + H(I_{\Theta_b}|I_{A_b}) \\ &= \sum_{b=0}^{B-1} \left(\sum_{q=0}^{\Omega_{I_{A_b}}-1} p_{I_{A_b}}(q) \log_2 [p_{I_{A_b}}(q)] \right. \\ &\quad \left. + \sum_{q=0}^{\Omega_{I_{A_b}}-1} \sum_{n=0}^{\Omega_{I_{\Theta_b}|I_{A_b}}(q)-1} p_{I_{\Theta_b}|I_{A_b}}(n|q) \log_2 [p_{I_{\Theta_b}|I_{A_b}}(n|q)] \right) \end{aligned} \quad (4.33)$$

Limitations du quantificateur polaire liées à l'hypothèse haut-débit

La construction du quantificateur se base sur la fonction débit-distorsion de l'équation (4.11) dérivée à partir l'hypothèse haut-débit. En pratique, il peut arriver que les débits alloués $R_b \sim H_b^*$ aux sous-bandes b soient trop faibles ($H_b \leq 3$ bits par pixel complexe) pour conserver le modèle débit-distorsion développé. Par conséquent, la quantification uniforme du module ne peut être considérée comme optimale pour les très bas débits. Cela entraîne un décalage entre la contrainte entropique globale H et le débit obtenu après codage entropique R_{WECUPQ} .

Un meilleur modèle de fonction de débit-distorsion pour les compressions très bas-débit permettrait de résoudre ce problème. Malheureusement, il reste très difficile d'établir un tel modèle sans hypothèse simplificatrice. Une piste proposée par Parisot et Antonini [10] concerne la quantification scalaire à contrainte entropique de coefficients d'ondelettes modélisés par une distribution gaussienne généralisée. La procédure d'allocation dynamique consiste à optimiser la largeur de l'intervalle central I_0 d'un quantificateur à zone morte au sens du minimum de l'erreur quadratique moyenne. Les auteurs montrent en particulier qu'un quantificateur à zone morte de paramètre $\delta = 0$ (voir chapitre 2, paragraphe 2.4.2) semble optimal pour la quantification d'une source laplacienne à bas débit.

La qualité de reconstruction de la phase étant un objectif prioritaire, les débits retenus en pratique sont suffisamment élevés pour se placer dans l'hypothèse haut-débit et donc haute-résolution.

Algorithme WECUPQ

En résumé, l'algorithme de compression WECUPQ se décompose de la manière suivante :

1. Recentrer le spectre Doppler de l'image SLC autour de la fréquence nulle.
2. Effectuer une transformée en ondelettes de la partie réelle et imaginaire de l'image sur L niveaux d'échelle. Le nombre de sous-bandes est $B = 3L + 1$.
3. Représenter les coefficients d'ondelettes $C_b = A_b e^{j\Theta_b}$ de chaque sous-bande b sous forme polaire.
4. Estimer pour chaque sous-bande les paramètres μ_b, L_b, M_b permettant de modéliser le module A_b des coefficients comme une source sans mémoire distribuée selon la loi de Fisher $f_{R_b}(a) \sim \mathcal{F}[\mu_b, L_b, M_b](a)$ ou Fisher généralisée.
5. Fixer une contrainte entropique H permettant de définir le débit binaire après codage entropique des indices de quantification.
6. Construire le quantificateur polaire optimal correspondant à l'aide des expressions (4.30).
7. Quantifier scalairement le module et la phase de chaque coefficient.
8. Encoder entropiquement les indices de quantification du module q et de la phase $n|q$ à l'aide d'un codage arithmétique. Les tables de probabilité pour le module et la phase sont respectivement calculées à partir des équations (4.31) et (4.32).

4.2.3 Application

Les figures 4.8 illustrent l'image d'amplitude extraite de la ville de Hyères et reconstruite après compression par l'algorithme WECUPQ. L'image a été acquise par le capteur RAMSES de l'ONERA en bande X et représente une zone urbaine composée d'un ensemble de bâtiments.

On observe visuellement une différence notable avec l'image originale pour les faibles débits notamment à 2 bits par pixel complexe. Les structures des bâtiments renvoient au capteur un signal plus énergétique et semblent être mieux reconstruites que les zones à faible réflectivité. On retrouve ici les limites de l'algorithme de quantification ECUPQ développé à partir de l'hypothèse haut-débit. En effet, pour de trop faibles débits ($H < 4$ bits par pixel complexe) le modèle proposé ne correspond pas à l'étude théorique du quantificateur ECUPQ. En revanche, sous l'hypothèse haut-débit ($H > 4$ bits par pixel complexe) les performances de l'algorithme en terme d'EQM convergent vers les courbes théoriques calculées à partir de l'expression (4.27). La figure 4.9 illustre ainsi les courbes débit-distorsion obtenues après compression de l'image extraite de la ville d'Hyères.

On remarquera en particulier que l'utilisation de la transformée en ondelettes à valeurs complexes SDW4 (*Symmetric Daubechies Wavelets*) à 10 coefficients ne semble pas apporter de gain particulier en terme de réduction d'erreur quadratique par rapport à la transformée bi-orthogonale 9/7.

La figure 4.10 illustre la répartition des débits de quantification entre :

- Le débit $H(I_A)$ associé à l'encodage entropique des indices de quantification du module des coefficients d'ondelettes.
- Le débit $H(I_\Theta/I_A)$ associé à l'encodage entropique des indices de quantification de la phase des coefficients d'ondelettes.

A partir de débits suffisamment élevés permettant de se placer sous l'hypothèse haute-résolution, on remarquera que la quantification de la phase des coefficients d'ondelettes requiert environ 0.8 bit par pixel complexe supplémentaire par rapport au débit réservé à la quantification du module.



(a) Image RSO SLC originale (amplitude) extraite de la ville d'Hyeres. ©ONERA.



(b) Taux $\eta = 32 : 1$ (2bppc), $RSB_A = 9.25dB$.



(c) Taux $\eta = 21.3 : 1$ (3bppc), $RSB_A = 12.62dB$.



(d) Taux $\eta = 16 : 1$ (4bppc), $RSB_A = 18.51dB$.

FIG. 4.8 – Exemple de compression d'image SLC par l'algorithme WECUPQ après transformée bi-orthogonale 9/7 de l'image complexe. Seule l'amplitude de l'image est représentée. L'image de phase conserve une structure de bruit blanc uniformément répartie entre $[0 : 2\pi]$.

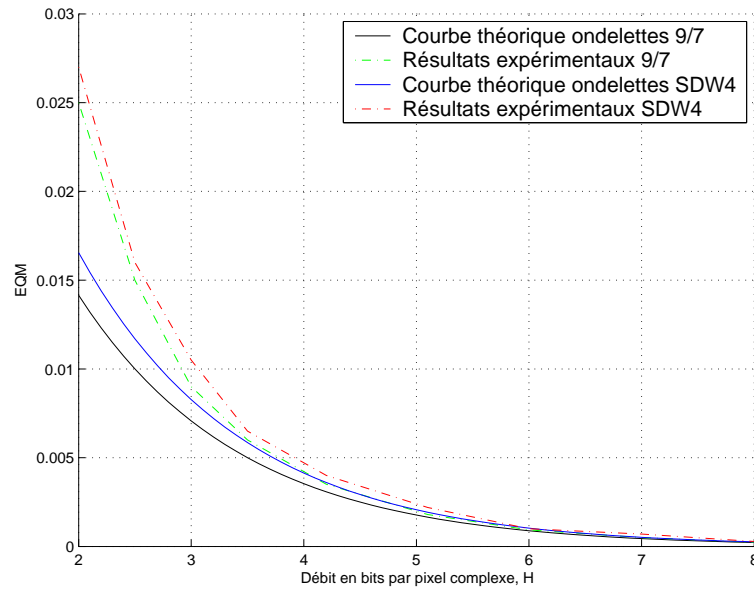


FIG. 4.9 – Evolution de l'EQM en fonction du débit après compression WECUPQ réalisée sur l'image extraite de la ville de Hyères. Les courbes en trait plein représentent les performances théoriques dérivées de l'équation (4.27) en fonction de la transformée en ondelettes choisie : 9/7 [5] ou SDW4 [52].

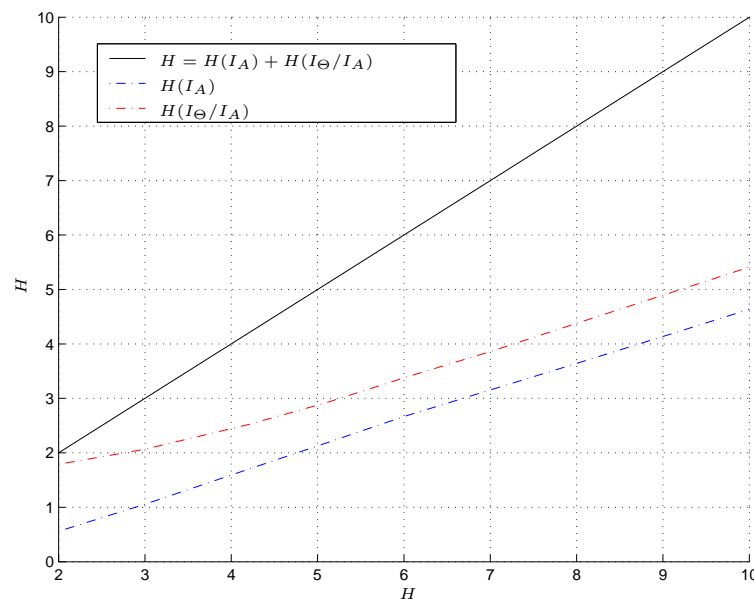


FIG. 4.10 – Répartition du débit entre l'amplitude et la phase pour l'algorithme WECUPQ.

4.3 Compression d'images SLC par la méthode TCEAC-CSPQ

Les méthodes TCE-IQC et WECUPQ ne permettent pas d'avoir un contrôle direct sur la qualité de reconstruction de la phase/amplitude car l'image a été compressée suivant la transformée en ondelettes des parties réelle et imaginaire de l'image. A l'opposé, l'approche TCEAC-CSPQ introduite au chapitre 3 (section 2) s'appuie sur une représentation amplitude/phase de l'image SLC. On peut donc espérer contrôler directement la qualité de l'image d'amplitude et de phase en les comprimant.

Si l'on considère un modèle statistique gaussien i.i.d pour les pixels formant la partie réelle I et imaginaire Q de l'image SLC, la phase Θ est donc distribuée uniformément sur $[0 : 2\pi]$ (voir chapitre 1). Un codage par transformée sensé exploiter les corrélations et la structure géométrique de l'image de phase serait donc inapproprié. D'un point de vue entropique, le gain de codage est nul et un code à longueur fixe peut suffire à représenter chaque pixel de phase sur un nombre équivalent de bits (par exemple 8 bits/pixel de phase soit 256 valeurs de phase disponibles pour une quantification scalaire uniforme). L'approche retenue consiste à accorder une précision plus ou moins importante à la quantification de la valeur de phase en fonction de la valeur d'amplitude associée au pixel en question. Par exemple, un pixel de forte amplitude (point brillant, cible...) pourrait être quantifié avec plus de précisions qu'un pixel de faible amplitude d'une zone d'ombre. Ce choix est cohérent avec les quantificateurs polaires de type UPQ où la phase des pixels d'amplitude élevée est quantifiée avec une distorsion réduite (voir figure 4.6(c)).

En se plaçant dans un cadre de travail visant à minimiser l'erreur quadratique moyenne, on rappelle que son expression dans le domaine complexe s'écrit sous la forme :

$$D = D_A + D_{\Theta/A} = E_A(|A - \hat{A}|^2) + 2E_{\Theta,A}[A\hat{A}(1 - \cos(\theta - \hat{\theta}))]$$

Suivant l'approche TCEAC-CSPQ, nous proposons de compresser l'amplitude de l'image SLC par une méthode dédiée. Nous choisirons en particulier l'algorithme TCE introduit au début de ce chapitre. Pour un débit R_A donné, la distorsion introduite par l'algorithme TCE sur l'image d'amplitude intervient naturellement dans le terme d'erreur quadratique moyenne D_A . La compression de l'information de phase consiste donc à minimiser le terme distorsion $D_{\Theta/A}$ sous contrainte entropique d'un débit noté $R_{\Theta/A}$.

Il est évident que cette approche est sous-optimale puisque l'optimum au sens du critère de l'EQM consiste à minimiser l'ensemble de la fonction de distorsion globale $D = D_A + D_{\Theta/A}$ et non pas les termes séparés. Cependant, nous rappelons que la construction du quantificateur polaire ECUPQ optimal contraint à une différence constante entre les débits alloués à la quantification du module et de la phase (voir figure 4.10). Ceci implique qu'il n'est donc pas possible de contrôler séparément le débit et la distorsion introduite sur le module et la phase.

L'intérêt de cette méthode réside donc dans le fait que l'on peut jouer indépendamment sur le débit R_A dédié au codage de l'image d'amplitude et pour $R_{\Theta/A}$ l'image de phase. Pour un débit global fixé $R = R_A + R_{\Theta/A}$, il sera alors possible d'attribuer plus de débit au codage de l'image de phase au dépend de celui de l'image d'amplitude et vice versa.

4.3.1 Quantification Scalaire de la Phase conditionnellement à l'Amplitude

Les hypothèses permettant de construire le quantificateur de phase conditionnellement à l'amplitude sont les suivantes :

- La distribution de probabilité associée à l'image d'amplitude A est modélisée par une loi de Fisher $p_A(a) \sim \mathcal{F}[\mu, L, M](a)$ (voir chapitres 1 et 3).
- On suppose que le débit R_A est suffisamment élevé de manière à conserver un comportement statistique proche entre l'image d'amplitude originale et l'image décompressée. La distribution de l'image d'amplitude reconstruite \hat{A} après décompression peut être modélisée par une loi de Fisher $p_{\hat{A}}(a) \sim \mathcal{F}[\hat{\mu}, \hat{L}, \hat{M}](a)$.
- Les valeurs des pixels de l'image d'amplitude reconstruite $\hat{a}(n, m)$ sont accessibles à la fois au niveau du codeur et du décodeur.
- La distribution des pixels de phase est uniforme entre $[0, 2\pi]$.
- L'amplitude et la phase sont indépendantes. Par conséquent, $p_{A,\Theta}(a, \theta) = p_A(a) \cdot p_{\Theta}(\theta)$.

L'objectif est de quantifier la phase conditionnellement à l'amplitude sans avoir à consacrer un débit supplémentaire portant sur l'information d'amplitude. C'est la raison pour laquelle la quantification et déquantification de chaque pixel $\theta(n, m)$ dépendra de la connaissance de l'amplitude du pixel reconstruit $\hat{a}(n, m)$. En effet, seule cette information est à la fois accessible au niveau du codeur **et** du décodeur.

De manière à pouvoir associer un quantificateur uniforme de la phase à un niveau d'amplitude $\hat{a}(n, m)$ du pixel reconstruit, nous proposons d'introduire artificiellement dans le plan complexe, une partition circulaire de l'amplitude $\{r_i\}$, $0 \leq i \leq L$ tel que $0 < r_1 < r_2 < r_3 < r_4 < \dots < r_L < r_{L+1} \leq r_{max}$. Pour chaque pixel, on pourra donc classer une valeur d'amplitude reconstruite \hat{a} comme appartenant à un anneau \mathcal{A}_i délimité par 2 cercles de rayon r_i et r_{i+1} .

$$\hat{a} \in \mathcal{A}_i \Leftrightarrow \hat{a} \in [r_i, r_{i+1}[\quad \forall i = 0..L$$

L'ensemble des niveaux d'amplitude représentatifs de cette classification uniforme seront les a_i tel que :

$$a_i = \frac{r_i + r_{i+1}}{2} \quad (4.34)$$

Pour chaque anneau \mathcal{A}_i , le quantificateur de phase consiste à former une partition uniforme sur P_i niveaux, justifié par le fait que la distribution de la phase est considérée comme uniforme quel que soit le niveau d'amplitude. La figure 4.11 illustre la structure de ce quantificateur scalaire de la phase dont la résolution est conditionnée par le module du pixel reconstruit.

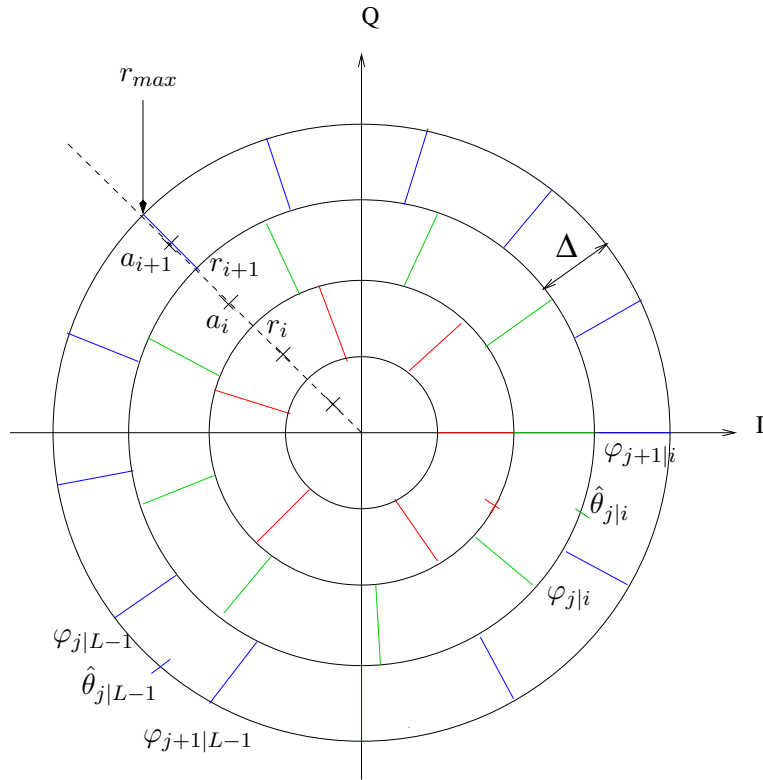


FIG. 4.11 – Principe de la quantification de phase conditionnellement à un niveau d'amplitude dans le plan complexe. Chaque pixel d'amplitude reconstruit par inversion de l'algorithme TCE est classé dans un anneau. La phase correspondant à ce pixel sera quantifiée par un quantificateur scalaire uniforme dont la résolution dépend de l'anneau considéré.

En fonction de l'anneau \mathcal{A}_i , le j -ème niveau de partition de phase s'écrit :

$$\varphi_{j|i} = \frac{(j-1)2\pi}{P_i} \quad j = 1 \dots P_i$$

La quantificateur Q_i de la phase θ s'implémente sous la forme :

$$\begin{aligned} \hat{\theta}_{j|i} &= Q_i(\theta \mid \hat{a} \in \mathcal{A}_i) \\ &= \frac{\varphi_{j|i} + \varphi_{j+1|i}}{2} \\ &= \frac{(2j-1)\pi}{P_i} \end{aligned}$$

Chaque indice de phase est ensuite codé par un codeur arithmétique contextuel et dont le contexte i est déterminé par l'amplitude du pixel reconstruit.

En complément du synoptique 3.5 introduit lors du chapitre 3, la chaîne de compression globale de l'algorithme TCEAC-CSPQ est donnée par la figure 4.12

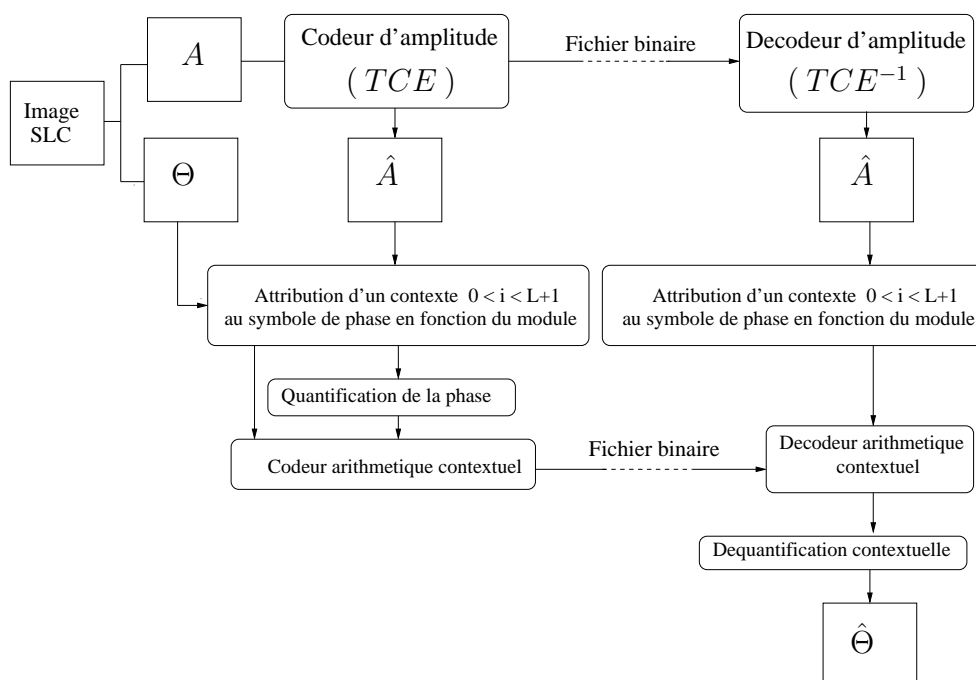


FIG. 4.12 – Chaîne de compression de l'algorithme TCEAC-CSPQ.

Contrainte entropique

Chaque anneau \mathcal{A}_i dispose d'un dictionnaire de P_i représentants de valeurs de phase. La probabilité associée à un index de quantification $j|i$ est uniforme et donnée par :

$$Prob(\hat{\theta}_{j|i} = Q_i(\theta | \hat{a} \in \mathcal{A}_i)) = \frac{1}{P_i} \quad \forall j = 1 \dots P_i$$

En appliquant un codage entropique sur les indices de quantification $j|i$ par un codeur contextuel dépendant du niveau d'amplitude i considéré, l'entropie associée au codage des symboles de phase s'écrit donc :

$$H(I_{\Theta|\hat{A}}) = \sum_{i=0}^{L-1} p_i \log_2(P_i) \quad \text{avec} \quad p_i = \int_{r_i}^{r_{i+1}} p_{\hat{A}}(a) \cdot da \quad (4.35)$$

où p_i représente la probabilité que le module \hat{a} du pixel quantifié appartienne à l'anneau \mathcal{A}_i . $I_{\Theta|\hat{A}}$ correspond à l'ensemble des indices de quantification des niveaux de phase sachant la valeur reconstruite du module.

Distorsion

La distorsion $D_{\Theta|A}$ est l'association de la distorsion granulaire D_g et de la distorsion de saturation D_o . En faisant l'hypothèse $\hat{a} \simeq a_i$, la distorsion granulaire est donnée par :

$$\begin{aligned} D_g &= 2E[A\hat{A}(1 - \cos(\Theta - \hat{\Theta}))] \\ &\simeq \sum_{i=0}^{L-1} \sum_{j=1}^{P_i} \int_{\varphi_{j|i}}^{\varphi_{j+1|i}} \int_{r_i}^{r_{i+1}} 2aa_i [1 - \cos(\theta - \hat{\theta}_{j|i})] p_{\Theta}(\theta) p_A(a) da d\theta \end{aligned} \quad (4.36)$$

On exprime la distorsion de saturation par :

$$D_o \simeq \sum_{j=1}^{P_L} \int_{\varphi_{j|L-1}}^{\varphi_{j+1|L-1}} \int_{r_{max}}^{+\infty} [2aa_L (1 - \cos(\theta - \hat{\theta}_{j|L-1}))] p_{\Theta}(\theta) p_A(a) da d\theta \quad (4.37)$$

En pratique, on choisit r_{max} suffisamment élevé pour négliger la distorsion de saturation de sorte que $D_{\Theta|A} \simeq D_g$.

En intégrant par rapport à θ , l'équation (4.36) se simplifie de la manière suivante :

$$\begin{aligned} D_{\Theta|A} &\simeq \sum_{i=0}^{L-1} \sum_{j=1}^{P_i} \int_{r_i}^{r_{i+1}} 2aa_i \int_{\varphi_{j|i}}^{\varphi_{j+1|i}} [1 - \cos(\theta - \hat{\theta}_{j|i})] \cdot p_{\Theta}(\theta) p_A(a) da d\theta \\ &\simeq \frac{1}{2\pi} \cdot \sum_{i=0}^{L-1} \int_{r_i}^{r_{i+1}} 2aa_i 2P_i \left[\frac{\pi}{P_i} - \sin\left(\frac{\pi}{P_i}\right) \right] p_A(a) da \\ &\simeq \sum_{i=0}^{L-1} \int_{r_i}^{r_{i+1}} 2aa_i \left[1 - \text{sinc}\left(\frac{\pi}{P_i}\right) \right] p_A(a) da \\ &\simeq \sum_{i=0}^{L-1} \frac{a_i \pi^2}{3P_i^2} \int_{r_i}^{r_{i+1}} a p_A(a) da \end{aligned} \quad (4.38)$$

en utilisant un développement limité de la fonction $\text{sinc}(x) = \frac{\sin(x)}{x} = 1 - 0.16605x^2 + \epsilon(x)$.

Optimisation du quantificateur

On cherche à déterminer l'ensemble des niveaux de partition P_i de sorte que la distortion $D_{\Theta|A}$ de l'équation (4.38) soit minimale sous la contrainte :

$$H(I_{\Theta|\hat{A}}) \leq R_{\Theta|\hat{A}}$$

A partir des conditions d'optimalité de Kuhn Tucker, le problème se résoud simplement par une minimisation sous contrainte. On introduit le Lagrangien :

$$\begin{aligned} J(\lambda) &= D_{\Theta|A} + \lambda(H(I_{\Theta|\hat{A}}) - R_{\Theta|\hat{A}}) \\ &= \sum_{i=0}^{L-1} \frac{a_i \pi^2}{3P_i^2} \int_{r_i}^{r_{i+1}} ap_A(a) da + \lambda \left(\sum_{i=0}^{L-1} p_i \log_2(P_i) - R_{\Theta|\hat{A}} \right) \end{aligned} \quad (4.39)$$

L'optimisation du quantificateur de phase consiste à trouver l'ensemble des $\{P_i^*\}$ $i = 0 \dots L - 1$ tel que :

$$P_i^* = \arg \min_{\{P_i\}} J(\lambda) \quad (4.40)$$

En dérivant J par rapport à P_i . En posant $\frac{\partial J}{\partial P_i} = 0$, on obtient :

$$\frac{\partial J}{\partial P_i} = -\frac{2}{3} \frac{\pi^2}{(P_i)^3} a_i \int_{r_i}^{r_{i+1}} ap_A(a) da + \lambda \frac{p_i}{\log(2)P_i} = 0 \quad (4.41)$$

On en déduit :

$$P_i^* = \frac{\alpha}{\lambda} \left[\frac{a_i \int_{r_i}^{r_{i+1}} ap_A(a) da}{\int_{r_i}^{r_{i+1}} p_{\hat{A}}(a) da} \right]^{0.5} \quad (4.42)$$

avec $\alpha = \sqrt{\frac{2\pi^2 \log 2}{3}}$.

Pour obtenir l'expression complète de P_i^* , il suffit de calculer λ telle que la contrainte $H_{\Theta|\hat{A}} = R_{\Theta|\hat{A}}$ soit satisfaite :

$$\sum_{i=0}^{L-1} p_i \log_2(P_i) = R_{\Theta|\hat{A}} \quad (4.43)$$

En remplaçant l'expression de l'équation (4.42) dans l'équation précédente, on obtient :

$$P_i^* = \sqrt{\frac{a_i \int_{r_i}^{r_{i+1}} ap_A(a) da}{p_i}} 2 \left[R_{\Theta|\hat{A}} - \frac{1}{2} \sum_{i=0}^{L-1} p_i \log_2 \left(a_i \int_{r_i}^{r_{i+1}} ap_A(r) dr \right) + \frac{1}{2} \sum_{i=0}^{L-1} p_i \log_2(p_i) \right] \quad (4.44)$$

Analyse Asymptotique

Nous avons vu que le paramètre L définit de manière artificielle un ensemble d'anneaux de largeur $\Delta = \frac{r_{max}}{L} = r_{i+1} - r_i$ et permet de repérer le module du pixel de l'image d'amplitude déquantifiée. On peut supposer que plus le nombre d'anneaux sera élevé plus la classification du niveau d'amplitude sera précise et meilleure sera la quantification de la phase conditionnée à l'amplitude. On se propose de calculer l'expression P_i^* de la distorsion de phase D_θ avec l'hypothèse $L \rightarrow \infty$:

$$L \rightarrow \infty \Rightarrow \Delta \rightarrow 0$$

La densité de probabilité associée au module \hat{A} d'un pixel reconstruit peut être considérée comme constante à l'intérieur d'un anneau \mathcal{A}_i . On en déduit quelques simplifications :

$$p_i = \int_{r_i}^{r_{i+1}} p_{\hat{A}}(a) da \simeq p_{\hat{A}}(a_i) \cdot \Delta \quad (4.45)$$

$$\int_{r_i}^{r_{i+1}} a p_A(a) da \simeq a_i p_A(a_i) \Delta \quad (4.46)$$

Ce qui permet les simplifications suivantes :

$$\begin{aligned} \sum_{i=0}^{L-1} p_i \log_2(p_i) &\simeq \sum_{i=0}^{L-1} p_{\hat{A}}(a_i) \Delta \log_2 [p_{\hat{A}}(a_i) \Delta] \\ &\simeq \int_0^{r_{max}} p_{\hat{A}}(a) \log_2 [p_{\hat{A}}(a)] da + \log_2 \Delta \\ &\simeq -h(\hat{A}) + \log_2 \Delta \end{aligned} \quad (4.47)$$

où $h(\hat{A})$ est l'entropie différentielle associée à l'amplitude de l'image reconstruite.

Suivant le même calcul d'approximation :

$$\begin{aligned} \sum_{i=0}^{L-1} p_i \log_2 \left(a_i \int_{r_i}^{r_{i+1}} a \cdot p_A(a) da \right) &\simeq \sum_{i=0}^{L-1} p_{\hat{A}}(a_i) \Delta \log_2 (a^2 p_A(a_i) \Delta) \\ &\simeq \int_0^{r_{max}} p_{\hat{A}}(a) \log_2 (a^2 p_A(a)) da + \log_2 \Delta \end{aligned} \quad (4.48)$$

En remplaçant les approximations précédentes dans l'équation (4.44), on obtient alors :

$$P_i^* = a_i \frac{p_A(a_i)}{p_{\hat{A}}(a_i)} \cdot 2 \left[R_{\Theta|\hat{A}} - \frac{1}{2} S(A, \hat{A}) - \frac{1}{2} h(\hat{A}) \right] \quad i = 1 \dots L \quad (4.49)$$

avec $S(A, \hat{A}) = \int_0^{r_{max}} p_{\hat{A}}(a) \log_2 [a^2 \cdot p_A(a)] da$.

On retrouve un résultat proche de celui obtenu pour le quantificateur polaire ECUPQ. En effet, la résolution du quantificateur scalaire de phase dépend linéairement du niveau d'amplitude a_i .

En résumé, nous disposons de la partition de phase optimale $\{P_i^*\}$ $0 \leq i \leq L - 1$ minimisant la distorsion $D_{\Theta|A}$. En remplaçant l'expression de P_i^* dans l'équation (4.38), on en déduit l'expression de la distorsion du quantificateur de phase optimal :

$$D_{\Theta|A} \simeq \frac{\pi^2}{3} 2^{-2R_{\Theta|\hat{A}}} [-2R_{\Theta|\hat{A}} + S(A, \hat{A}) + h(\hat{A})] \quad (4.50)$$

A partir de quelle valeur de L peut-on considérer la partition d'amplitude comme suffisamment résolue pour se placer dans les conditions asymptotiques? En pratique, on retrouve les résultats précédents pour $L \geq 15$ comme le montre par exemple la figure 4.13. Pour différents débits $R_{\Theta|\hat{A}}$, on observe alors la convergence de la distorsion $D_{\Theta|A}(L)$ donnée par l'expression (4.38) vers la limite asymptotique (formule 4.50).

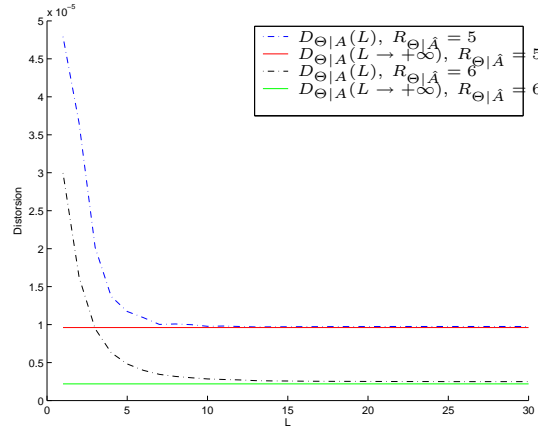


FIG. 4.13 – Evolution de la distorsion en fonction du paramètre L pour une source circulaire dont le module est distribué suivant une loi de Fisher Généralisée. Paramètres : $\mu = 0.010$, $L = 1.1344$, $M = 2.075$, $\alpha = 0.6$.

Supposons que l'image d'amplitude A ait été encodée avec un débit R_A suffisamment important de sorte que les caractéristiques statistiques de l'image d'amplitude reconstruite \hat{A} soient peu différentes de celles de l'image d'origine. On peut alors approcher la distribution du module de l'image originale avec la distribution du module de l'image reconstruite :

$$p_{\hat{A}}(a) \simeq p_A(a) \quad \forall a \in \mathbb{R}^+ \quad (4.51)$$

On en déduit l'expression simplifiée de P_i^* :

$$P_i^* \simeq a_i 2^{-R_{\Theta|\hat{A}} - b(\hat{A})} \quad i = 0 \dots L - 1 \quad (4.52)$$

où $b(\hat{A}) = \int_0^{r_{max}} \log_2(a) p_{\hat{A}}(a) da$.

La distorsion s'exprime alors sous la forme :

$$\begin{aligned} D_{\Theta|A} &\simeq \frac{\pi^2}{3} 2^{-2(R_{\Theta|\hat{A}} - b(\hat{A}))} \\ &\simeq \frac{1}{12} 2^{-2(R_{\Theta|\hat{A}} - b(\hat{A}) - h(\Theta))} \end{aligned} \quad (4.53)$$

où $h(\Theta) = \log_2(2\pi)$ est l'entropie différentielle associée à l'image de phase.

Algorithme TCEAC-CSPQ

En résumé, l'algorithme de compression TCE-CSPQ se décompose de la manière suivante :

1. Compresser l'image d'amplitude A à l'aide de l'algorithme TCE.
2. Reconstruire l'image d'amplitude \hat{A} par IDWT sur les coefficients quantifiés(transformée en ondelettes discrètes inverse).
3. Estimer sur l'image les paramètres $\mu_A, L_A, M_A, \alpha_A$ permettant de modéliser le module A des coefficients comme suivant une distribution de Fisher Généralisée $p_A(r) \sim \mathcal{FG}[\mu_A, L_A, M_A, \alpha_A](a)$.
4. Estimer sur une fenêtre de l'image les paramètres $\mu_{\hat{A}}, L_{\hat{A}}, M_{\hat{A}}, \alpha_{\hat{A}}$ permettant de modéliser le module de l'image reconstruite \hat{A} comme une source sans mémoire distribuée selon la loi de Fisher Généralisée $p_{\hat{A}}(a) \sim \mathcal{FG}[\mu_{\hat{A}}, L_{\hat{A}}, M_{\hat{A}}, \alpha_{\hat{A}}](a)$.
5. Fixer une contrainte entropique en fonction du débit souhaité $H_{\Theta|\hat{A}} \simeq R_{\Theta|\hat{A}}$
6. Construire la partition de phase optimale $\{[P_i^*]\}$, $i = 0 \dots L$ en fonction du contexte d'amplitude i et de l'équation (4.49).
7. Quantifier la phase du pixel $\hat{\theta}_{j|i} = Q_i(\theta | \hat{a} \in \mathcal{A}_i)$ en fonction du niveau d'amplitude du pixel reconstruit $\hat{a}(m, n)$.
8. Encoder entropiquement les indices de quantification $\{j|i\}$ par un codeur arithmétique contextuel pour chaque niveau i d'amplitude. La table de probabilité associée au symbole de phase $\hat{\theta}_{j|i}$ est donnée par l'équation (4.35).

4.3.2 Application

Les figures 4.14 illustrent la compression de l'image d'amplitude extraite de la ville d'Istres. L'image de phase est compressée en fonction de l'image d'amplitude reconstruite. La figure 4.15 montre l'évolution de la fonction de distorsion $D_{\Theta|A}$ en fonction du débit $R_{\Theta|A}$ pour la quantification de l'image de phase. L'image d'amplitude a été préalablement compressée par l'algorithme TCE.

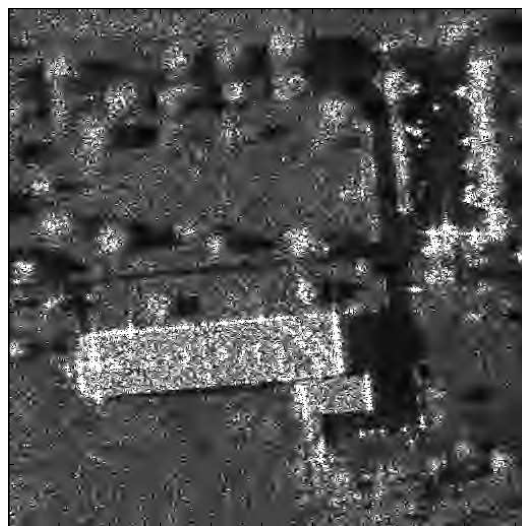
On remarquera que pour les faibles débits ($R_{\Theta|A} < 3bpp$), la distorsion empirique $EQM_{\Theta|A}$ ne correspond plus au modèle estimé $D_{\Theta|A}$ (équation (4.53)). Ceci est dû au fait que le nombre trop faible de cellules de quantification de la phase ne permet plus de se placer dans l'hypothèse haute résolution.

En résumé, le codeur TCEAC-CSPQ a pour intérêt de pouvoir contrôler séparément la distorsion introduite sur la phase de celle introduite sur l'image d'amplitude. Cette dernière est encodée indépendamment de l'image de phase par l'algorithme TCE. Concernant l'image de phase, on recherche à minimiser la mesure de distorsion qui se distingue de l'erreur quadratique moyenne classique par l'adjonction d'un terme de pondération quadratique du module. Cette fonction quadratique pénalise d'autant plus la distorsion que le module associé au symbole est élevé.

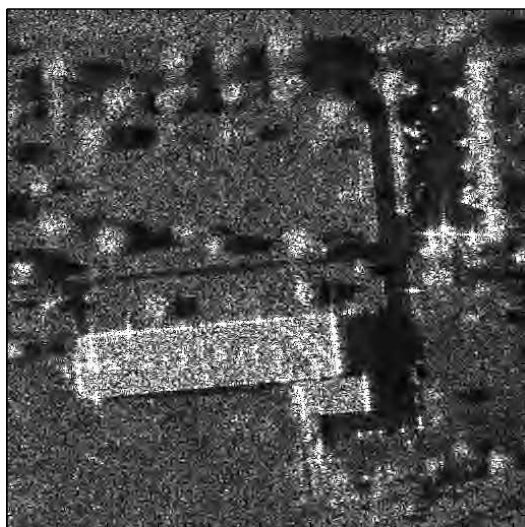
Nous avons construit un quantificateur qui repère dans un premier temps le module du pixel complexe et le relie à un contexte d'amplitude i . Chaque contexte d'amplitude est ensuite associé à un quantificateur de phase scalaire uniforme dont l'alphabet prend ses valeurs dans $[0 : 2\pi]$. Le paramètre à déterminer est donc le cardinal P_i de ces quantificateurs Q_i , $1 \leq i \leq L$ minimisant la distorsion associée à la phase. Les résultats obtenus montrent que ce quantificateur est à bien des égards cohérent avec l'approche ECUPQ puisque la résolution des quantificateurs de phase augmente linéairement avec l'amplitude.



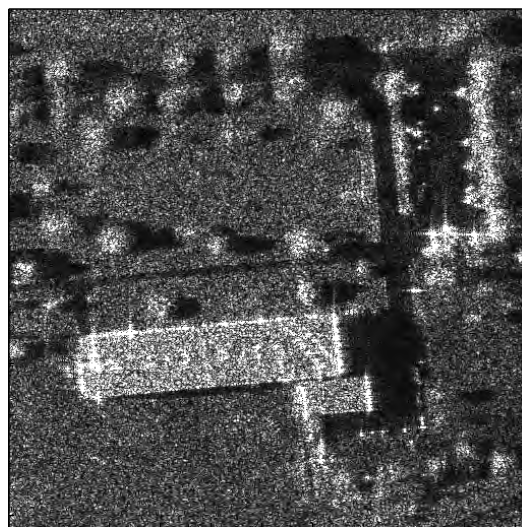
(a) Image RSO SLC originale (amplitude) extraite de la ville d'Istres. ©ONERA.



(b) $R_A = 0.25bpp$, $EQM_A = 1.374.e - 04$, $RSB_A = 9.76dB$.



(c) $R_A = 0.5bpp$, $EQM_A = 9.81.e - 04$, $RSB_A = 11.22dB$.



(d) $R_A = 1.5bpp$, $EQM_A = 2.9.e - 04$, $RSB_A = 16.41.dB$.

FIG. 4.14 – compression d'image d'amplitude extraite de la ville d'Istres par l'algorithme TCE. Transformée : bi-orthogonale 9/7.

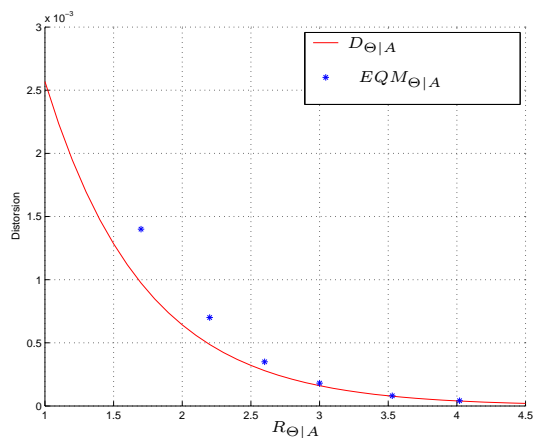


FIG. 4.15 – Evolution de la distorsion de phase $D_{\Theta|A}$ en fonction du débit $R_{\Theta|A}$ alloué à la quantification de l'image conditionnellement à l'image d'amplitude extraite de la ville d'Istres.

4.4 Algorithme TCEAC-IPQ

Cette méthode se différencie de l'approche précédente dans le sens où chaque pixel de l'image de phase est quantifié en fonction du pixel de l'image d'amplitude correspondant. Concernant la compression de l'image d'amplitude, l'approche est similaire à la technique précédente. Nous proposons en particulier d'utiliser l'algorithme TCE basé sur la transformée en ondelettes.

L'image de phase est quantifiée vectoriellement avec une contrainte entropique dépendante du débit R_Θ . L'image est découpée en blocs x_n de taille 2x2 de dimension 4. La construction du dictionnaire en dimension 4 est réalisée en utilisant l'algorithme GLA (voir chapitre 2) à partir d'une base d'apprentissage comprenant une série d'images de phase extraite du capteur RAMSES de l'ONERA. La phase étant considérée comme uniformément répartie sur $[0 : 2\pi]$ une quantification vectorielle algébrique comprenant un réseau régulier de points aurait pu être envisagée.

Cependant, quelle que soit la technique de quantification vectorielle de la phase choisie, il est légitime de penser que celle-ci restera sous-optimale au sens de l'erreur quadratique moyenne calculée pour l'ensemble de l'image SLC complexe (amplitude et phase). En effet, la quantification vectorielle de la phase est optimisée suivant le minimum de l'erreur quadratique moyenne $D_\Theta = (E[(\Theta - \hat{\Theta})^2])$.¹ sans tenir compte de la nature complexe de l'image SLC. D'un point de vue conceptuel et en considérant la quantification vectorielle uniforme à une dimension (donc une quantification scalaire), le schéma de compression de l'image complexe TCEAC-IPQ s'inspire de la technique de quantification polaire CPQ (*Conventional Polar Quantization*) illustrée par la figure 4.6. En effet, la quantification scalaire du module de l'image est alors remplacée par l'algorithme TCE. Le quantificateur scalaire uniforme de la phase est remplacé par un quantificateur vectoriel de manière à profiter du gain sur la distorsion D_Θ apportée par l'augmentation de la dimension.

¹il conviendra de prendre en compte les phénomènes de périodicité, l'erreur de phase étant comprise entre $[-\pi : \pi]$.

4.5 Conclusion

Ce chapitre a permis de détailler les diverses approches de compression d'images SLC dépendant de leur représentation dans le domaine complexe. Ces algorithmes tirent profit de la transformée en ondelettes discrètes qu'elle soit appliquée directement sur l'image d'amplitude ou sur les parties réelle et imaginaire de l'image. Nous avons en particulier :

- appliqué avec succès l'algorithme TCE de C. Tian et S.S Hemami sur les parties réelle et imaginaire de l'image SLC (méthode TCE-IQC).
- appliqué une technique de quantification polaire à contrainte entropique sur les coefficients d'ondelettes à valeurs complexes obtenus à partir d'une transformée en ondelettes réalisée sur les parties réelle et imaginaire de l'image SLC. Nous avons mis en oeuvre une procédure d'allocation dynamique de débit à partir de la fonction débit-distorsion théorique du quantificateur ECUPQ (méthode WECUPQ).
- appliqué l'algorithme TCE à la compression de l'image d'amplitude. Concernant la compression de l'image de phase, nous avons construit un quantificateur scalaire de l'image de phase conditionné par le niveau du module des pixels de l'image d'amplitude (méthode TCEAC-CSPQ). Il est alors possible de contrôler séparément le débit et la distorsion associés respectivement à la compression des images d'amplitude et de phase.
- construit un quantificateur vectoriel de type GLA sur l'image de phase qui est alors compressée indépendamment de l'image d'amplitude (méthode TCEAC-IPQ).

Nous consacrerons le chapitre 6 à une étude comparative et plus détaillée des performances de ces algorithmes. Néanmoins, les résultats préliminaires montrent que l'utilisation d'ondelettes de nature complexe de la famille SDW ne semble pas apporter de gain particulier par rapport à la transformée biorthogonale 9/7 sur l'algorithme WECUPQ. Ces résultats corroborent les conclusions apportées par l'étude de Ives [43].

D'autre part, les images RSO SLC affectées par le phénomène de chatoiement disposent d'un contenu fréquentiel riche en hautes fréquences. Or les transformations en ondelettes effectuent une partition dyadique de l'axe des échelles qui tend à privilégier une représentation du signal par les sous-bandes basses-fréquences. Peut-on considérer la décomposition en ondelettes discrètes comme optimale vis-à-vis de la compression d'images RSO SLC? Cette remarque nous amène à considérer une décomposition spatio-fréquentielle plus générale comme la transformée en paquets d'ondelettes.

Les paquets d'ondelettes ont été introduits par Coifman [13] et réalisent une décomposition suivant la meilleure base du plan fréquentiel à partir d'un critère fixé. En retenant un critère basé sur le compromis débit-distorsion, G. Mercier [61] a appliqué l'algorithme de Ramchandran et Vetterli [77] aux images RSO d'amplitude ERS-1. L'algorithme en question considèrerait alors la décomposition en ondelettes dyadiques comme optimale.

Cependant, compte tenu de la structure fréquentielle de l'image complexe, nous proposons d'appliquer la décomposition en paquets d'ondelettes sur les parties réelle et imaginaire de l'image SLC. Le chapitre suivant aborde la généralisation des algorithmes TCE-IQC et WECUPQ à la transformée en paquets d'ondelettes de l'image RSO SLC.

Chapitre 5

Compression d'images RSO complexes basée sur la transformée en paquets d'ondelettes

Les ondelettes fournissent un outil puissant pour la compression des images. Les images RSO sont affectées par le phénomène de chatoiement qui introduit des composantes hautes fréquences très énergétiques. Ces images deviennent ainsi beaucoup plus difficiles à compresser que les images optiques. Dans ce chapitre, nous nous intéresserons à une méthode de décomposition de l'image adaptative par paquets d'ondelettes.

Les paquets d'ondelettes ont été introduits par Coifman [13] et permettent de s'affranchir du schéma de re-décomposition dyadique de la sous-bande basse-fréquence caractéristique de la DWT. Il s'agit en fait de construire une librairie de fonctions avec différentes propriétés de localisation espace-fréquence. La décomposition n'est pas unique et peut se faire en choisissant une base de fonctions adaptée au contenu de l'image. Le problème classique associé au codage d'image par paquets d'ondelettes consiste à prendre en compte toutes les bases de paquets d'ondelettes possibles et à choisir celles qui permet d'obtenir les meilleures performances au sens d'un critère qui reste à définir.

Ce chapitre se consacre à l'extension des méthodes de quantification présentées aux chapitres 3 et 4 appliquées sur les coefficients issus d'une décomposition en paquets d'ondelettes. Nous présenterons brièvement dans un premier temps, le principe de la transformée en paquets d'ondelettes (en anglais *Wavelet Packet Transform*) afin de se focaliser plus spécifiquement sur la sélection de la meilleure base au sens d'un critère à définir.

Nous avons ainsi mis en oeuvre deux algorithmes de compression proposés comme les extensions respectives des algorithmes WECUPQ et TCE-IQC :

- La quantification polaire à contrainte entropique des coefficients de paquets d'ondelettes complexes : **WP-ECUPQ** (*Wavelet Packets and Entropy Constrained Unrestricted Polar Quantization*)
- L'extension de l'algorithme TCE à la transformée par paquets d'ondelettes appliqué sur les parties réelles et imaginaire de l'image RSO SLC : **WPTCE-IQC** (*Wavelet Packets TCE and In phase and Quadrature coding*).

5.1 Les paquets d'ondelettes

Cette section s'appuie sur les travaux de Coifman [13] et Mvogo [66] qui exposent une étude plus détaillée sur la théorie des paquets d'ondelettes. Nous nous contenterons ici de

rappeler les principaux résultats.

Soient $\{h_0[n]\}$ et $\{h_1[n]\}$ deux séquences associées à un filtre passe-bas et passe-haut et $\{g_0[n]\}$ et $\{g_1[n]\}$ les filtres en quadrature conjugués tels que :

$$\begin{cases} h_1[n] = (-1)^n g_0^*[1-n] \\ g_1[n] = (-1)^n h_0^*[1-n] \end{cases}$$

On définit alors les fonctions suivantes :

$$\begin{cases} W_{2n}(t) = \sqrt{2} \sum_k h_0[k] W_n[2t-k] \\ W_{2n+1}(t) = \sqrt{2} \sum_k h_1[k] W_n[2t-k] \end{cases}$$

Les fonctions $W_0(t)$ et $W_1(t)$ peuvent être respectivement identifiées aux fonctions d'échelle et d'ondelettes $\varphi(t)$ et $\psi(t)$ définies au chapitre 2.

On démontre alors que les fonctions $W_n(t-k)$, pour $k \in \mathbb{Z}$ et $n \geq 0$ forment une base orthonormale de $L^2(\mathbb{R})$ appelée paquets d'ondelettes d'échelle fixe (en anglais *fixed scale wavelets packets*). La base de paquets d'ondelettes de $L^2(\mathbb{R})$ est une base sélectionnée parmi les fonctions $W_{nk}^j = 2^{j/2} W_n(2^j t - k)$.

Soit x un signal discret $x = \{x[k]\}$, $k = 0 \dots K-1$. Les coefficients de paquets d'ondelettes sont définis à partir de l'équation d'analyse :

$$\begin{aligned} w_{2n}^j[k] &= \sum_l h_0[l-2k] w_n^{j-1}[l] = F_0\{w_n^{j-1}\}[k] \\ w_{2n+1}^j[k] &= \sum_l h_1[l-2k] w_n^{j-1}[l] = F_1\{w_n^{j-1}\}[k] \\ w_0^0[k] &= x[k] \end{aligned} \tag{5.1}$$

où

- j est l'indice d'échelle. Le signal x est échantillonné à l'échelle $j = 0$.
- k est l'indice de localisation spatiale.
- n est l'indice fréquentiel correspondant à n oscillations à la fréquence $n2^{-j}$.

Les coefficients au niveau d'échelle j sont obtenus en appliquant de manière récursive les opérateurs F_0 , F_1 sur les coefficients issus du niveau d'échelle précédent $j-1$. La transformée en ondelettes classique est obtenue en itérant l'opérateur F_0 sur les coefficients basse-fréquence $w_{2n}^j[k]$, $n \in \mathbb{N}$. La transformée en paquets d'ondelettes s'autorise le fait de redécomposer les coefficients $w_{2n+1}^j[k]$ issus du filtrage passe haut par les opérateurs F_1 et F_2 .

La transformée en paquets d'ondelettes peut se représenter par un arbre de décomposition dont la racine et les feuilles correspondent respectivement à l'image de départ et aux sous-bandes de l'analyse multi-résolution. En deux dimensions, un noeud père possède 4 noeuds enfants. Cette relation entre noeuds parents-enfants illustrée par les figures 5.1 et 5.2 correspond simplement à une échelle supplémentaire d'analyse par banc de filtres. Le choix d'un sous-arbre est associé à une base particulière de paquets d'ondelettes.

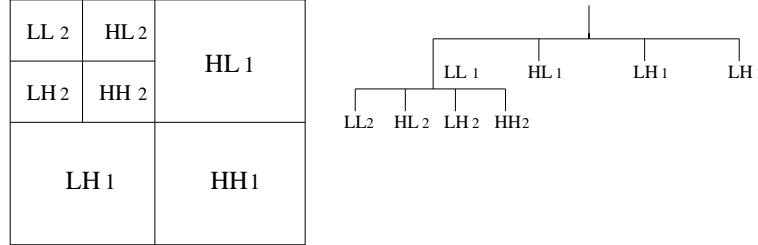


FIG. 5.1 – Correspondance entre le plan analyse fréquentiel par DWT et l'arborescence de décomposition.

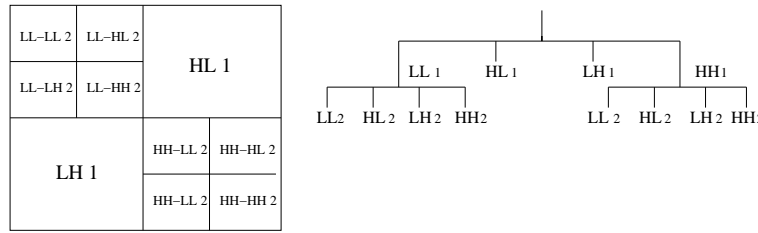


FIG. 5.2 – Correspondance entre le plan analyse fréquentiel pour une base particulière de paquets d'ondelettes et l'arborescence de décomposition.

En fonction du paramètre d'échelle j , on introduit $\{b_x, b_y\} \subset \mathbb{N} \times \mathbb{N}$ comme étant les coordonnées associées à la représentation des sous bandes dans le plan fréquentiel 2-D :

$$(b_x, b_y) \in [0 : 2^j - 1] \times [0 : 2^j - 1] \quad (5.2)$$

A chaque sous-bande de coordonnées (b_x, b_y) correspond un noeud $n \in \mathbb{N}$ de l'arbre de décomposition représentatif de la base de paquets d'ondelettes utilisée. Une relation possible permettant de relier n aux coordonnées (b_x, b_y) est donnée par [9] :

$$n(j, b_x, b_y) = 4^j + b_x \cdot 2^j + b_y - 1 \quad (5.3)$$

La figure 5.3 illustre l'arborescence complète issue de la décomposition récursive des sous-bandes basses et hautes fréquences jusqu'à une profondeur de l'arbre égale à $J = 2$.

Soit $n(j, b_x, b_y)$ l'indice d'un noeud de l'arbre correspondant à la sous-bande de coordonnées (b_x, b_y) à l'échelle j . Ce noeud dispose de 4 noeuds fils indexés de la manière suivante :

$$\begin{aligned}
 n_0 &= n(j+1, 2 \cdot b_x, 2 \cdot b_y) \\
 n_1 &= n(j+1, 2 \cdot b_x, 2 \cdot b_y + 1) \\
 n_2 &= n(j+1, 2 \cdot b_x + 1, 2 \cdot b_y) \\
 n_3 &= n(j+1, 2 \cdot b_x + 1, 2 \cdot b_y + 1)
 \end{aligned} \quad (5.4)$$

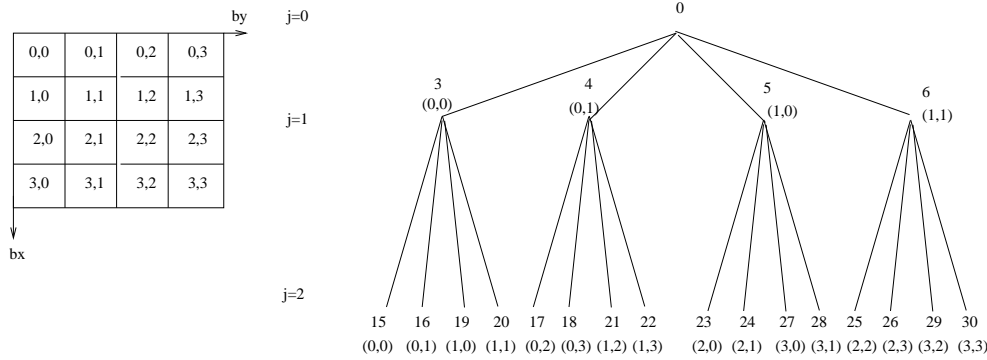


FIG. 5.3 – Correspondance entre la représentation de la décomposition complète et l'indexation des noeuds de l'arbre de profondeur $J = 2$.

En résumé, la souplesse de la transformée en paquet d'ondelettes permet de fournir un ensemble de représentations adaptées au contenu spatio-fréquentiel de l'image. Il s'agit donc de sélectionner avant tout la librairie de fonctions qui offre les meilleures performances suivant un critère à définir. Cette librairie correspond à une arborescence particulière, elle-même représentative d'un découpage particulier du plan fréquentiel à deux dimensions.

5.2 Choix de la meilleure base

La construction des paquets d'ondelettes amène à un degré de liberté extrêmement élevé quant au choix de la meilleure base associée à l'analyse du signal. Il convient donc de définir un critère permettant d'extraire la meilleure base de paquets d'ondelettes, la recherche exhaustive des combinaisons possibles devenant en pratique impossible à mettre en oeuvre.

5.2.1 Fonctions de coût

Les propriétés intéressantes d'une fonction de coût objective concernent sa séparabilité. Dans un contexte de recherche de meilleure base, on se restreint au choix d'un ensemble de fonctions additives.

Coifman et Wickehauser [13] ont proposé un algorithme rapide de sélection de base de complexité $N \log_2(N)$ où N correspond au nombre de pixels de l'image. Pour une sous-bande $\mathbf{c}_n = \{c[j]\}$ indexée par n , la fonction de coût utilisée est basée sur une définition de l'entropie au sens de Shannon-Weaver :

$$h(\mathbf{c}_n) = - \sum_j \frac{|c[j]|^2}{\|\mathbf{c}_n\|^2} \log \left(\frac{|c[j]|^2}{\|\mathbf{c}_n\|^2} \right) \quad (5.5)$$

Dans un contexte de compression d'images RSO d'amplitude, Z. Zeng et I. G. Cumming [116] ont appliqué l'algorithme SPIHT à partir de sous-bandes issues d'une décomposition en paquets d'ondelettes. La fonction de coût permettant la sélection de la meilleure base s'appuie sur l'énergie des coefficients d'une sous-bande \mathbf{c}_n :

$$e(\mathbf{c}_n) = \sum_j \|c[j]\|^2 \quad (5.6)$$

$$(5.7)$$

En pratique, l'inconvénient de ces critères réside dans le fait qu'il ne prennent pas en compte le compromis débit-distorsion. En effet, la fonction h de la formule (5.5) n'a pas de lien direct avec l'entropie d'ordre zéro et ne correspond donc pas au débit nécessaire pour encoder l'ensemble des coefficients d'une sous-bande $\mathbf{c}_n = \{c[j]\}$.

Le compromis débit-distorsion étant dépendant du quantificateur utilisé et donc de la distorsion introduite, Ramchandran et Vetterli [77] [78] [79] [113] ont proposé une fonction de coût basée sur la minisation de l'erreur quadratique moyenne sous contrainte d'un débit fixé.

Etant donné un ensemble de quantificateurs $\{Q\}$, la fonction de débit $R_Q(\mathbf{c}_n)$ peut être estimée par l'entropie d'ordre zéro sur les indices de quantification à transmettre. La distorsion $D_Q(\mathbf{c}_n)$ est donnée par l'erreur quadratique. La sélection de la meilleure base se fait suivant le critère $J_{\lambda,Q}$ dépendant d'un paramètre lagrangien λ et d'expression :

$$J_{\lambda,Q}(\mathbf{c}_n) = D_Q(\mathbf{c}_n) + \lambda R_Q(\mathbf{c}_n) \quad (5.8)$$

Dans le cadre d'un codage indépendant des sous-bandes \mathbf{c}_n issues de la décomposition en paquets d'ondelettes orthonormées, le critère de coût global met en jeu un ensemble de fonctions lagrangiennes ayant la propriété d'être séparables, additives et positives.

$$J_{\lambda,Q} = \sum_n J_{\lambda,Q}(\mathbf{c}_n) \quad (5.9)$$

En l'absence de critères de qualité définis à partir d'une application post-compression, nous avons choisi une sélection de meilleure base rattachée à la minimisation de la fonction $J_{\lambda,Q}$. Ce critère est optimal dans le sens où la distorsion ne pourra être réduite sans augmentation du débit ou vice-versa. Le choix d'une base suivant ce critère se traduit ainsi par un problème d'allocation dynamique de débit attribués à chaque sous-bande [77].

5.2.2 Allocation dynamique de débit et recherche de la meilleure base

Nous introduisons ici quelques notations :

- T : Arborescence complète de la décomposition en paquets d'ondelettes (en anglais *full wavelet packet tree*) illustrée par la figure 5.3.
- $S \subseteq T$: Arborescence tronquée par rapport à T c'est à dire n'importe quel sous-arbre de décomposition admissible.
- $S_n \subseteq S$: Sous-arbre de S et dont le noeud n constitue la racine. Par défaut, le noeud racine est indexé la valeur $n = 0$ suivant la formule 5.3. On a donc $S = S_0$.
- \mathcal{B} une séquence binaire permettant de décrire le sous-arbre S .
- $P \subset S$: Ensemble des noeuds terminaux du sous-arbre S . Ces noeuds correspondant à l'ensemble des sous-bandes qui seront effectivement encodées après sélection de la meilleure base.
- $D_Q(S)$, $R_Q(S)$: distorsion et débit globaux associés au codage du sous arbre S .
- Q_n : Quantificateur associé au codage des coefficients de la sous-bande \mathbf{c}_n .

Les sous-bandes étant quantifiées indépendamment, le problème de recherche de meilleure base consiste à trouver le sous-arbre S^* avec $P^* \subset S^*$ tel que :

$$D_Q(S^*) = \min_{\{Q_n\}} \sum_n D_{Q_n}(\mathbf{c}_n) \quad n \in P^* \quad (5.10)$$

sous la contrainte de débit qui s'exprime par :

$$R_Q(S^*) = \sum_n R_{Q_n}(\mathbf{c}_n) = R_{Budget} \quad \text{et} \quad n \in P^* \quad (5.11)$$

où R_{Budget} est le débit auquel on souhaite compresser l'image.

En fixant le paramètre lagrangien λ en référence à l'équation (5.8), le problème d'optimisation revient à trouver la meilleure base S_λ^* telle que :

$$S_\lambda^* = \arg \min_{\{S\}} \left[\min_{\{Q_n\}} \sum_{n \in P} J_{\lambda, Q_n}(\mathbf{c}_n) \right] \quad (5.12)$$

Recherche de la meilleure base

On décompose l'image de départ suivant l'arborescence complète T sur J niveaux d'échelles, de manière à pouvoir prendre en compte le calcul de la fonction de coût associée à toutes les sous-bandes. La solution du problème (5.12) correspond à 3 niveaux d'optimisation :

1. Pour chaque noeud $n \in T$ et λ fixé, il s'agit de trouver le quantificateur optimal Q_n^* minimisant la fonction de coût $J_{\lambda, Q_n}(\mathbf{c}_n)$. C'est essentiellement sur ce point que diffèrent les méthodes WP-ECUPQ et WP-TCE qui seront présentées à la section suivante.
2. Pour λ fixé, il faut tronquer l'arborescence complète de l'arbre telle que la fonction de coût de l'arbre soit minimale.

De manière à simplifier les notations, nous omettons l'indice Q_n^* associé au quantificateur optimal déterminé lors de la phase 1) de l'algorithme. Soit $J_\lambda(\mathbf{c}_p)$ la fonction de coût d'un noeud père d'indice $p = n(j, b_x, b_y)$. On définit par $J_\lambda^{child\ nodes}$, la fonction de coût associée au codage des 4 noeuds fils :

$$J_\lambda^{child\ nodes} = \sum_{k=0}^{k=1} \sum_{l=0}^{l=1} J_\lambda[\mathbf{c}_{n(j+1, 2.b_x+k, 2.b_y+l)}] \quad (5.13)$$

En partant de l'avant-dernier niveau de décomposition $j = J - 1$ et en remontant récursivement vers la racine $j = 0$, on compare la fonction de coût de chaque noeud père avec celle des noeuds fils de manière à minimiser le coût de l'arbre.

Si la fonction de coût des noeuds enfants est strictement inférieure à celle du noeud père, il existe un intérêt à re-décomposer la sous-bande \mathbf{c}_p . Dans ce cas, on conserve les noeuds fils. Dans le cas contraire, les noeuds fils sont supprimés de l'arbre.

$$\begin{array}{ll}
Si & J_\lambda(\mathbf{c}_p) \leq J_\lambda^{child\ nodes} \\
& J_\lambda(\mathbf{c}_p) = J_\lambda(\mathbf{c}_p) \\
& \mathcal{B}_{S_\lambda^*}(p) = 0 \quad (merge) \\
\\
Sinon & \\
& J_\lambda(\mathbf{c}_p) = J_\lambda^{child\ nodes} \\
& \mathcal{B}_{S_\lambda^*}(p) = 1 \quad (split)
\end{array}$$

L'ensemble de ces décisions binaires forment une séquence $\mathcal{B}_{S_\lambda^*}$ permettant de décrire le sous-arbre S_λ^* optimal.

- La dernière optimisation consiste à reprendre les optimisations 1) et 2) jusqu'à ce que le paramètre λ^* optimal corresponde à la contrainte de débit (cf équation (5.11)). A cet effet, l'algorithme de la bisection [77] permet de progressivement isoler un intervalle tel que $\lambda^* \in [\lambda_{min}, \lambda_{max}]$.

Un exemple est donné par la figure 5.4 où l'arborescence de décomposition évolue en fonction de la courbe débit-distorsion globale $\{D_Q(S_\lambda^*), R_Q(S_\lambda^*)\}$. Les différentes bases dépendent du choix du paramètre de qualité λ qui influe sur la sélection de la base S^* jusqu'à ce que la contrainte de débit soit satisfaite.

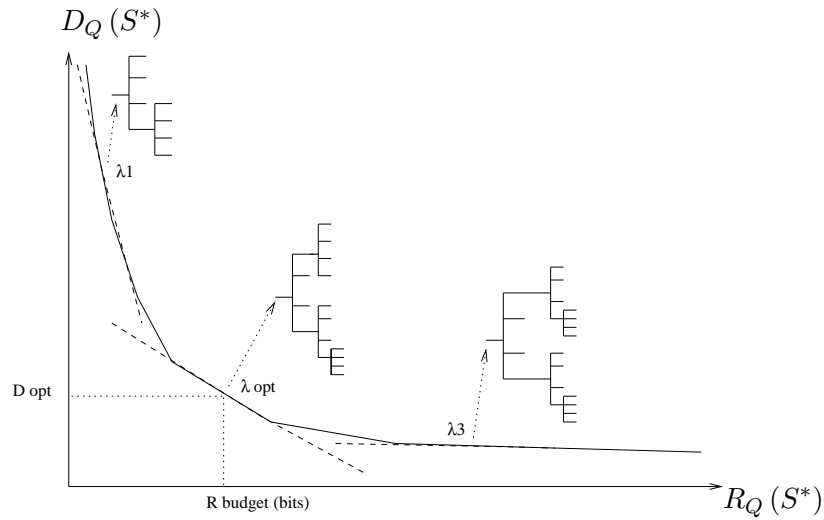


FIG. 5.4 – Courbe débit-distorsion et influence du paramètre λ optimal sur la sélection de la meilleure base S^* en fonction de la contrainte de débit R_{Budget} .

5.3 Paquets d'ondelettes et algorithme WP-ECUPQ

5.3.1 Principe

Les images RSO ont la particularité d'être affectée par le phénomène du chatolement qui introduit de fortes variations locales du signal et un aspect particulièrement texturé. Contrairement à la transformée en ondelettes classique, les paquets d'ondelettes offrent un ensemble important de bases de représentations qui permettent de s'adapter localement au contenu spatio-fréquentiel de l'image.

Dans le cadre de ce chapitre, nous proposons l'algorithme WP-ECUPQ présenté ici comme une extension aux paquets d'ondelettes de l'algorithme WECUPQ détaillé au chapitre 4. Cet algorithme consiste à remplacer la transformée en ondelettes classique par une transformée en paquets d'ondelettes sur la partie réelle I et imaginaire Q de l'image RSO SLC. Les coefficients de chaque sous-bande \mathbf{c}_n sont complexes et représentés suivant leur module A_n et leur phase Θ_n .

L'algorithme comporte alors 5 étapes :

1. La décomposition en paquets d'ondelettes suivant l'arborescence complète T des parties réelle et imaginaire de l'image SLC.
2. La sélection de la meilleure base S_λ^* en fonction de la contrainte de débit binaire R_{Budget} .
3. Le codage entropique de la séquence binaire $\mathcal{B}_{S_\lambda^*}$ permettant au décodeur de retrouver la base de paquets d'ondelettes à partir de laquelle l'image originale a été compressée.
4. La quantification polaire des coefficients de paquets d'ondelettes complexes suivant la méthode ECUPQ.
5. Le codage entropique des indices de quantification.

La figure 5.5 illustre la chaîne de compression dans son ensemble.

Etape 1 : Décomposition en paquets d'ondelettes de l'image SLC complexe

De manière analogue à la méthode WECUPQ, nous appliquons cette fois-ci, la transformée en paquets d'ondelettes sur les parties réelle et imaginaire de l'image SLC. Nous utiliserons en particulier la transformée bi-orthogonale 9/7 dont les coefficients de filtres sont à valeurs réelles.

Dans le cadre d'une transformée en paquets d'ondelettes définie par l'opérateur WPT (*Wavelet Packet Transform*), la propriété de linéarité de la transformée permet d'écrire :

$$WPT(I + jQ) \iff WPT(I) + j.WPT(Q) = I_{WPT} + j.Q_{WPT}$$

où I_{WPT} et Q_{WPT} correspondent à l'ensemble des coefficients issus respectivement de la transformée en paquets d'ondelettes réalisée sur la partie réelle et imaginaire de l'image originale.

Pour chaque sous-bande \mathbf{c}_n issue de la décomposition complète, on considère les coefficients comme les réalisations d'un ensemble de variables i.i.d représentée par la variable aléatoire : $C_n = A_n \cdot e^{j\Theta_n}$.

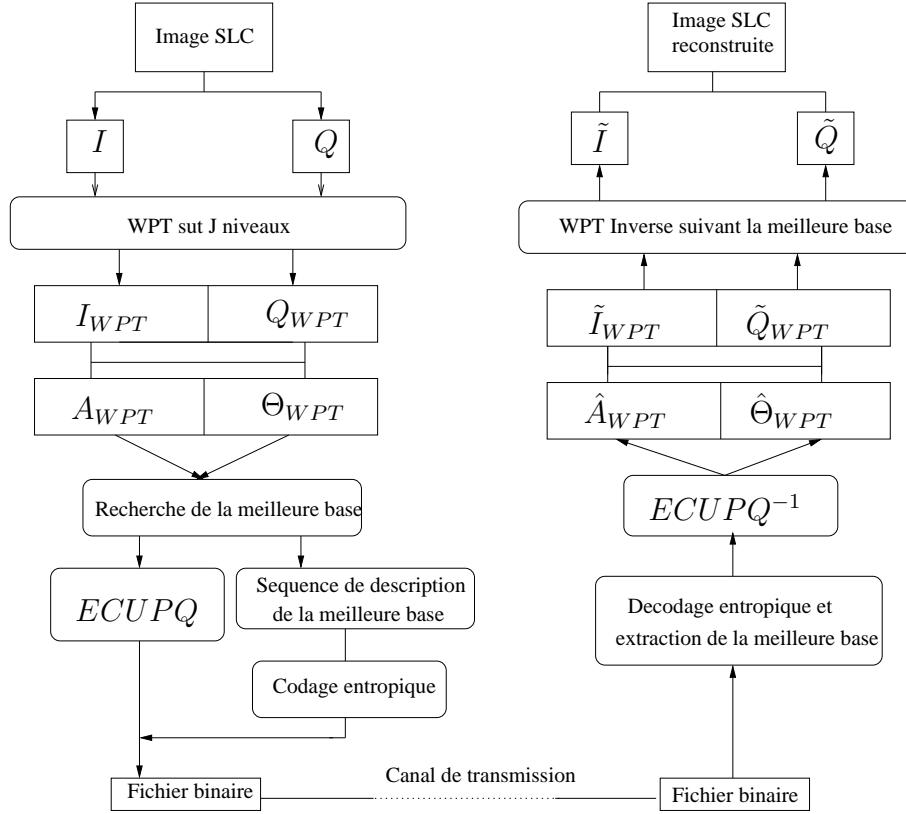


FIG. 5.5 – Compression d’une image SLC basée sur une transformée en paquets d’ondelettes (WPT) et quantification ECUPQ des coefficients à partir de leur représentation polaire

En reprenant les modèles statistiques introduits au chapitre 3, le module A_n peut être caractérisé par une loi de Fisher de d.d.p $f_{A_n}(a) \sim \mathcal{F}[\mu_n, L_n, M_n](a)$ dont les paramètres sont estimés par la méthode des log-cumulants. La phase Θ_n est indépendante de l’amplitude et uniformément distribuée sur $[0 : 2\pi]$.

Etape 2 : Recherche de la meilleure base

A partir de la méthode de quantification polaire à contrainte entropique (ECUPQ) décrite au chapitre 4, il est possible de prédire le comportement débit-distorsion associé au codage de chaque sous-bande \mathbf{c}_n . Le critère $J_{\lambda, Q_n}(\mathbf{c}_n)$ s’exprime à partir de l’expression de la fonction débit-distorsion (formule 4.11) et du Lagrangien λ . On obtient ainsi :

$$\begin{aligned} J_{\lambda, Q_n}(\mathbf{c}_n) &= D(\mathbf{c}_n) + \lambda \cdot R(\mathbf{c}_n) \\ &= \frac{1}{6} 2^{-[R(\mathbf{c}_n) - h(A_n) - h(\Theta_n) - b(A_n)]} + \lambda R(\mathbf{c}_n) \end{aligned} \quad (5.14)$$

où $h(A_n)$, $h(\Theta_n)$ sont respectivement les entropies différentielles du module et la phase. On rappelle que $b(A_n) = \int f_{A_n}(a) \log_2(a) da$.

A partir de ce modèle, nous pouvons en déduire le quantificateur Q_n^* optimal minimisant la fonction de coût J_{λ, Q_n} . Il s’agit en fait de trouver le débit optimal $R(\mathbf{c}_n)$ nécessaire à la construction du quantificateur ECUPQ (voir équations 4.12 du chapitre 4) et tel que :

$$R(\mathbf{c}_n)^* : \arg_{R(\mathbf{c}_n)} \min J_{\lambda, Q_n}(\mathbf{c}_n) \quad \forall n \in T$$

Connaissant l'expression de la fonction de coût pour chaque noeud de l'arbre T , la recherche de la meilleure est réalisée suivant l'algorithme présenté en section 1 de ce chapitre jusqu'à ce que la contrainte de débit soit satisfaite.

Etape 3 : codage de la séquence de meilleure base

La séquence binaire représentant la meilleure base de décomposition est encodée entropiquement par un codeur arithmétique adaptatif dont la table de probabilité est mise à jour suivant la fréquence empirique des symboles. En pratique, le débit associé à la description de la meilleure base est négligeable devant le débit consacré à la quantification des coefficients.

Etape 4 et 5

Une fois la meilleure base déterminée, les étapes 4 et 5 de quantification et codage entropique sont identiques à celles présentées au chapitre 4 de ce manuscrit pour l'algorithme WECUPQ.

5.3.2 Application

Nous présentons ici quelques résultats du codeur WP-ECUPQ appliqué sur une zone extraite de la ville de Hyères. Les figures 5.6 représentent la meilleure base dans le plan fréquentiel en fonction du débit auquel on souhaite compresser l'image. Pour des débits supérieurs à 4 bits par pixel complexe (bppc), la transformée en paquets d'ondelettes appliquée à l'image SLC semble apporter un gain réel par rapport à la DWT. A débit identique, on observe un gain d'environ 3dB sur l'erreur quadratique moyenne par rapport à la méthode WECUPQ. La meilleure base associée correspond à une analyse à plus grande échelle des sous-bandes hautes-fréquences.

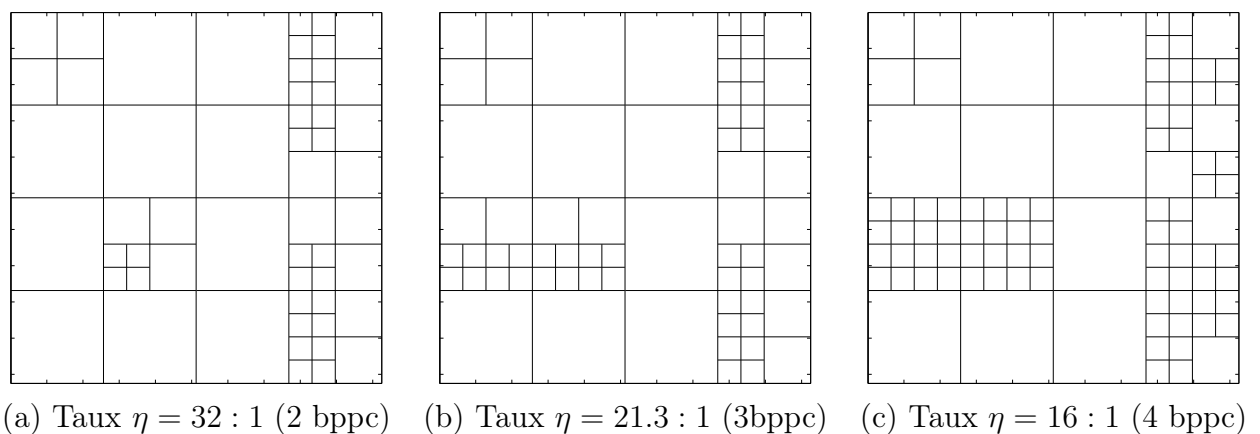


FIG. 5.6 – Algorithme WP-ECUPQ : représentation dans le plan fréquentiel de la transformée en paquets d'ondelettes après recherche de la meilleure base pour différents débits.



(a) Image RSO SLC originale (amplitude) extraite de la ville d'Hyeres. ©ONERA.



(b) Taux $\eta = 32 : 1$ (2 bppc), $RSB_A = 10.09dB$.



(c) Taux $\eta = 21.3 : 1$ (3 bppc), $RSB_A = 15.27dB$.



(d) Taux $\eta = 16 : 1$ (4 bppc), $RSB_A = 21.12dB$.

FIG. 5.7 – Exemple de compression d'image SLC par l'algorithme WP-ECUPQ après transformée bi-orthogonale 9/7 de l'image complexe. Seule l'amplitude de l'image est représentée.

5.4 Paquets d'ondelettes et algorithme TCE

Nous avons vu au chapitre 4 de ce manuscrit, l'application de l'algorithme TCE de Chao Tian and Sheila S. Hemami [98] à la compression des parties réelle et imaginaire de l'image complexe (TCE-IQC). Nous suggérons dans le cadre de cette étude d'étendre cette méthode de compression à la transformée en paquets d'ondelettes (WP-TCE).

5.4.1 Algorithme : WP-TCE

La figure 5.8 illustre la chaîne d'encodage de l'algorithme WP-TCE :

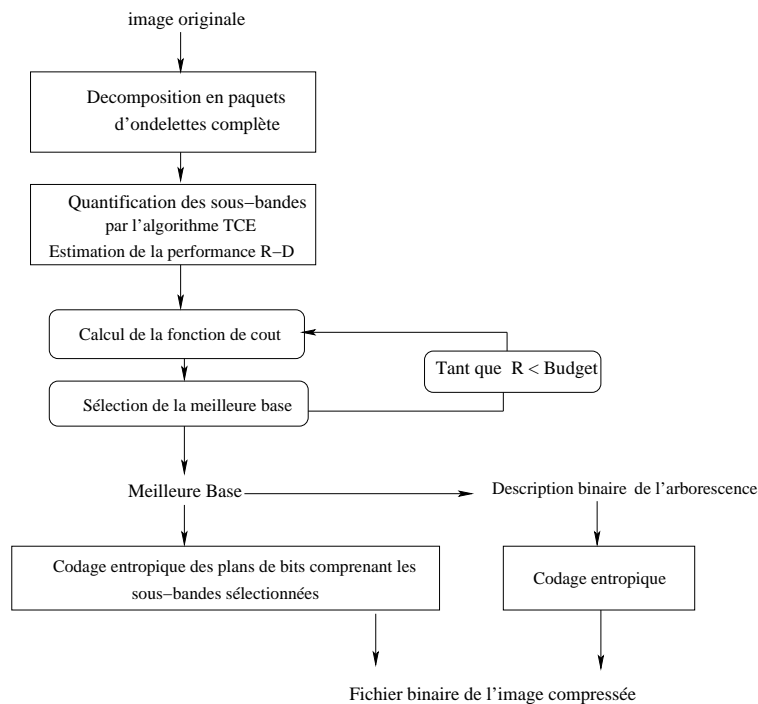


FIG. 5.8 – Schéma général de l'algorithme WP-TCE basée sur la transformée en ondelettes.

Estimation de la courbe débit-distorsion

Il s'agit d'appliquer l'algorithme de recherche de meilleure base de Ramchandran et Vetterli suivant le critère Lagrangien défini dans l'équation (5.8). De manière à pouvoir calculer la fonction de coût $J_{\lambda, Q_n}(\mathbf{c}_n)$ associée à chaque noeud n de l'arbre complet T , il est nécessaire de connaître l'évolution de la distorsion $D_{Q_n}(\mathbf{c}_n)$ en fonction du débit $R_{Q_n}(\mathbf{c}_n)$. Il s'agit en réalité d'une estimation puisqu'il est inutile d'envisager un codage entropique des symboles de quantification sur toutes les sous-bandes \mathbf{c}_n , $n \in T$. En effet, seuls les coefficients appartenant aux sous bandes sélectionnées par la recherche de meilleure base seront transmis.

En prenant en compte la stratégie de quantification progressive par plan de bits de l'algorithme TCE, nous pouvons déterminer l'évolution de la courbe débit-distorsion empirique au fur et à mesure de la quantification d'une sous-bande. Soit $\mathbf{c}_n[j]$ un coefficient de paquets d'ondelettes appartenant à une sous-bande \mathbf{c}_n . Suivant les notations introduites au chapitre 4, la séquence binaire associée au coefficient $\hat{\mathbf{c}}_n^p[j]$ quantifié sur p plan de bits est donnée par :

$$\hat{\mathbf{c}}_n^p[j] \equiv \{s_{q,\mathbf{c}_n[j]}, b_{q,\mathbf{c}_n[j]}, b_{q-1,\mathbf{c}_n[j]}, b_{q-2,\mathbf{c}_n[j]}, \dots, b_{q-p+1,\mathbf{c}_n[j]}\} \quad (5.15)$$

où :

- $s_{q,\mathbf{c}_n[j]} = \text{sign}(\mathbf{c}_n[j])$ correspond au signe du coefficient.
- $b_{q,\mathbf{c}_n[j]} = 1$ correspond à l'état où le coefficient devient significatif par rapport au plan de bits q .
- $b_{q-p,\mathbf{c}_n[j]} \in \{0, 1\}$ correspond aux bits de raffinement du coefficient $\mathbf{c}_n[j]$.

La transmission d'un symbole binaire de la séquence $\hat{\mathbf{c}}_n^p[j]$ permet de réduire progressivement la distorsion d'un coefficient et donc la distorsion associée à la sous-bande \mathbf{c}_n . On introduit ainsi l'ensemble des indices $z \in \mathcal{Z}_n$ représentatifs des points de la courbe débit-distorsion $\{R_{Q_n}^{(z)}(\mathbf{c}_n), D_{Q_n}^{(z)}(\mathbf{c}_n)\}$ relative à la quantification progressive d'une sous-bande \mathbf{c}_n .

Pour chaque plan de bits et chaque coefficient appartenant à la sous-bande \mathbf{c}_n , le codage d'un nouveau symbole entraîne donc une incrémentation de l'indice z . A partir de l'expression (5.15), nous rajoutons l'indice z permettant de faire le lien entre la séquence binaire issue de la quantification par plan de bits et la position du point de coordonnées $(R_{Q_n}^{(z)}(\mathbf{c}_n), D_{Q_n}^{(z)}(\mathbf{c}_n))$ sur la courbe débit-distorsion.

$$\hat{\mathbf{c}}_n^{p(z)}[j] \equiv \{s_{q,\mathbf{c}_n[j]}^{(z_1)}, b_{q,\mathbf{c}_n[j]}^{(z_2)}, b_{q-1,\mathbf{c}_n[j]}^{(z_3)}, b_{q-2,\mathbf{c}_n[j]}^{(z_4)}, \dots, b_{q-p+1,\mathbf{c}_n[j]}^{(z)}\} \quad (5.16)$$

Au point z , l'estimation de la distorsion pour la sous-bande n est alors donnée par l'erreur quadratique :

$$D_{Q_n}^{(z)}(\mathbf{c}_n) = \sum_j \left(\mathbf{c}_n[j] - \hat{\mathbf{c}}_n^{p(z)}[j] \right)^2 \quad (5.17)$$

A chaque symbole de quantification par plan de bits est associé une probabilité permettant de "piloter" le codeur entropique arithmétique. Soit $p[z]$ la probabilité au codage d'un symbole associée au point z telle que $p[z] = \text{Prob}\left(b_{i,\mathbf{c}_n[j]}^{(z)} = 1\right)$. Sans avoir expressément recours à la séquence binaire obtenue en sortie du codeur arithmétique, on estime le débit d'encodage progressif d'une sous-bande par l'entropie d'ordre zéro :

$$\begin{aligned} R_{Q_n}^{(z)}(\mathbf{c}_n) &= R_{Q_n}^{(z-1)}(\mathbf{c}_n) - \log_2(p[z]) \quad \text{si } b_{i,\mathbf{c}_n[j]}^{(z)} = 1 \quad i \in \mathbb{N} \\ R_{Q_n}^{(z)}(\mathbf{c}_n) &= R_{Q_n}^{(z-1)}(\mathbf{c}_n) - \log_2(1 - p[z]) \quad \text{si } b_{i,\mathbf{c}_n[j]}^{(z)} = 0 \quad i \in \mathbb{N} \\ \text{avec } R_{Q_n}^{(0)}(\mathbf{c}_n) &= 0 \end{aligned} \quad (5.18)$$

La figure 5.9 représente l'estimation de la courbe $\{R_{Q_n}^{(z)}(\mathbf{c}_n), D_{Q_n}^{(z)}(\mathbf{c}_n)\}$ pour une sous-bande après quantification par l'algorithme TCE.

Fonction de coût et point de troncature

Nous recherchons la meilleure base de décomposition minimisant le critère Lagrangien $J_{\lambda,Q_n}(\mathbf{c}_n)$. Dans le cadre d'un codage progressif par l'algorithme TCE, il s'agit donc de trouver le point de troncature optimal $z_{\lambda,n}^*$ tel que :

$$z_{\lambda,n}^* : \arg \min_{\{z\}} J_{\lambda,Q_n}(\mathbf{c}_n)^{(z)} = D_{Q_n}^{(z)}(\mathbf{c}_n) + \lambda \cdot R_{Q_n}^{(z)}(\mathbf{c}_n) \quad \forall n \in T \quad (5.19)$$

Ce problème d'allocation optimale peut être facilement résolu si l'on suppose une convexité de la courbe formée par les points $\{R_{Q_n}^{(z)}(\mathbf{c}_n), D_{Q_n}^{(z)}(\mathbf{c}_n)\}$. La fonction $J_{\lambda, Q_n}^{(z)}(\mathbf{c}_n)$ possède alors un minimum global correspondant au point $z_{\lambda, n}^*$ que l'on recherche.

En pratique, la courbe de débit-distorsion n'est pas tout à fait convexe comme illustrée sur la figure 5.9. En effet, suivant l'ordre d'encodage des symboles de quantification rien ne garantit une convexité parfaite de la courbe construite à partir des équations (5.17) et (5.18). D'autre part, l'ensemble de points de troncutures $z \in \mathcal{Z}_n$ est important (plusieurs milliers de points par sous-bande). Sachant que la recherche de minimisation (5.19) doit être itérée jusqu'à ce que le paramètre λ optimal soit trouvé, la réduction de l'ensemble de recherche \mathcal{Z}_n devient inévitable si l'on souhaite une convergence rapide de l'algorithme de recherche de meilleure base.

Nous avons retenu la solution consistant à sélectionner un sous-ensemble des points de troncuture $\mathcal{H}_n \subseteq \mathcal{Z}_n$ formant une courbe linéaire par morceau et convexe (l'ensemble \mathcal{Z}_n étant discret).

Cette approche est similaire à celle présentée par D. Taubman avec l'algorithme PCRD-Opt (Post compression Rate Distortion optimization) [95] présent dans le standard JPEG2000 et l'algorithme EBCOT [92].

Pour de plus de précisions concernant l'algorithme d'extraction de la courbure convexe, nous suggérons au lecteur de se référer à l'ouvrage de D. Taubman [96] ou à la partie Annexes de ce manuscrit.

La figure 5.9 montre l'ensemble des points \mathcal{H}_n retenus, permettant de dresser un "profil" convexe de la courbe débit-distorsion.

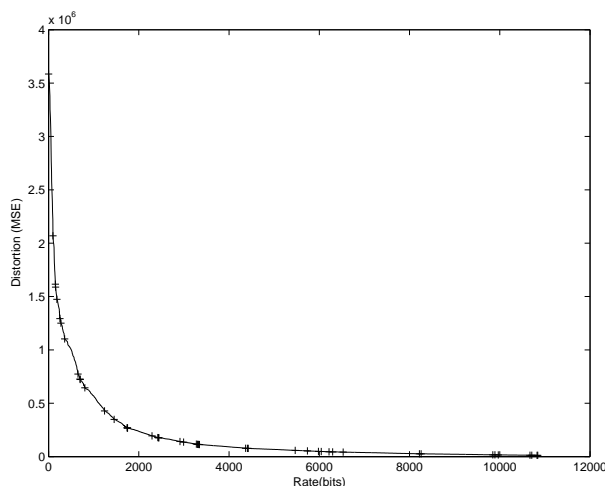


FIG. 5.9 – Ensemble des points z formant la courbe débit-distorsion estimée pour une sous-bande particulière. Les points (+) forment un ensemble \mathcal{H}_n convexe par morceau.

La figure 5.10 illustre la fonction de coût $J_{\lambda, Q_n}(\mathbf{c}_n)$ pour une valeur de λ fixée. L'espace de recherche étant restreint à l'ensemble de points \mathcal{H}_n , la procédure d'extraction du minimum de la fonction de coût est rapide et converge vers un minimum global.

En résumé, la quantification progressive par plan de bits de l'algorithme TCE appliqué à toutes les sous-bandes permet d'établir une estimation sur la courbe débit-distorsion.

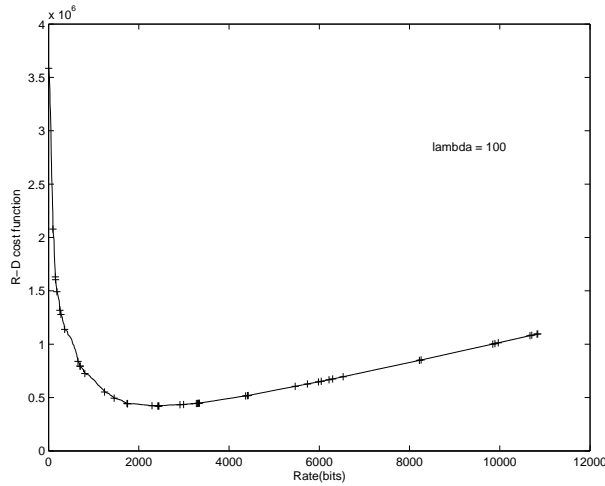


FIG. 5.10 – Calcul de la fonction de coût $J_{\lambda, Q_n}(\mathbf{c}_n)$ et extraction du minimum à partir de l'ensemble de points $z \in \mathcal{H}_n$.

Pour chaque sous-bande \mathbf{c}_n , $n \in T$, l'extraction de la courbure convexe permet de calculer avec une complexité de calcul raisonnable le minimum de la fonction de coût $J_{\lambda, Q_n}(\mathbf{c}_n)$.

La recherche de la meilleure base est ensuite effectuée par les étapes d'optimisation 2) et 3) de l'algorithme présenté au début de ce chapitre. En considérant les sous-bandes sélectionnées par l'algorithme de recherche de meilleure base, on transmet au codeur arithmétique les symboles binaires représentatifs de la quantification par plan de bits des coefficients.

5.4.2 Algorithme WPTCE-IQC appliqué à la compression d'images RSO

L'approche WPTCE-IQC consiste à appliquer l'algorithme WP-TCE à la compression des parties réelle et imaginaire de l'image RSO SLC. Cette méthode est simplement une généralisation à la transformée en paquets d'ondelettes de l'algorithme TCE-IQC présenté au chapitre 4. La figure 5.11 illustre la représentation de la transformée en paquets d'ondelettes dans le plan fréquentiel après recherche de la meilleure base sur la scène extraite de la ville d'Istres (image du capteur RAMSES acquise en bande X). La recherche de meilleure base a été réalisée en utilisant la fonction d'ondelette biorthogonale 9/7 sur 4 niveaux d'échelles.

On remarquera que pour un débit fixé, il existe une similarité entre les meilleures bases de paquets d'ondelettes associées à la compression de la partie réelle et imaginaire. Pour des débits supérieurs à 2 bits par pixel complexe, les mesures de rapport signal sur bruit réalisées sur les images de la figure 5.12 montrent que la méthode WPTCE-IQC apporte un gain substantiel d'environ 3dB par rapport à la méthode TCE-IQC basée sur la DWT. La table 5.1 illustre le temps de calcul nécessaire à l'encodage et au décodage de la partie réelle de l'image.

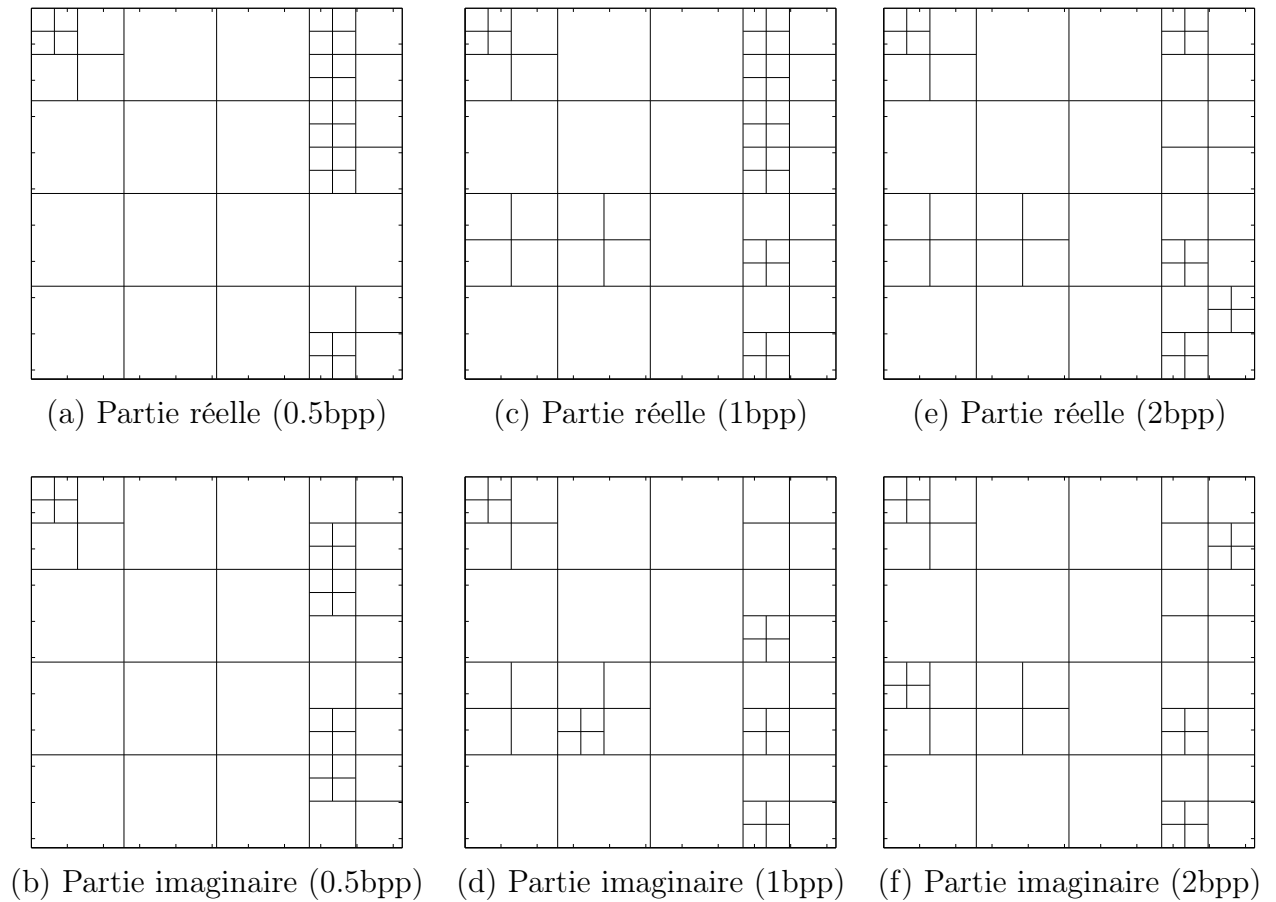
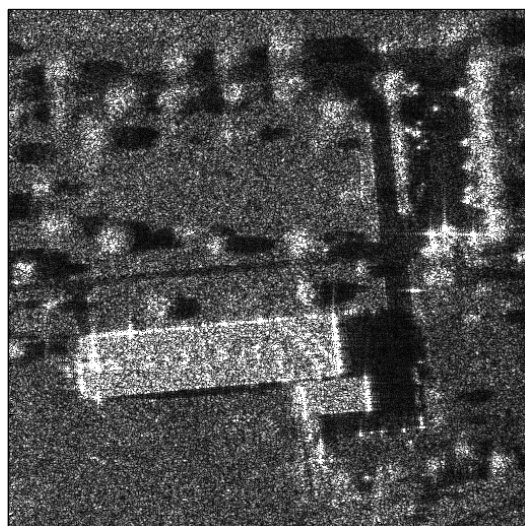


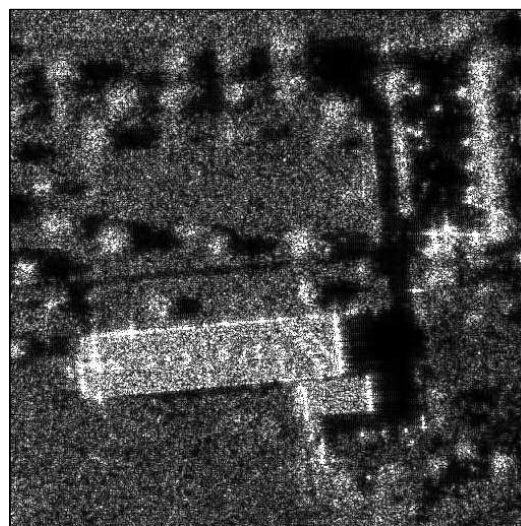
FIG. 5.11 – Algorithme WPTCE-IQC : représentation dans le plan fréquentiel de la transformée en paquets d'ondelettes après recherche de la meilleure base pour différents débits. (a) et (b) Taux $\eta = 64 : 1$ (1bppc) / (c) et (d) Taux $\eta = 32 : 1$ (2bppc) / (e) et (f) Taux $\eta = 16 : 1$ (4bppc)

Débit (bpp)	0.5bpp	1bpp	2bpp
Encodage (ms)	3114	3410	3710
Decodage (ms)	1005	1030	1106

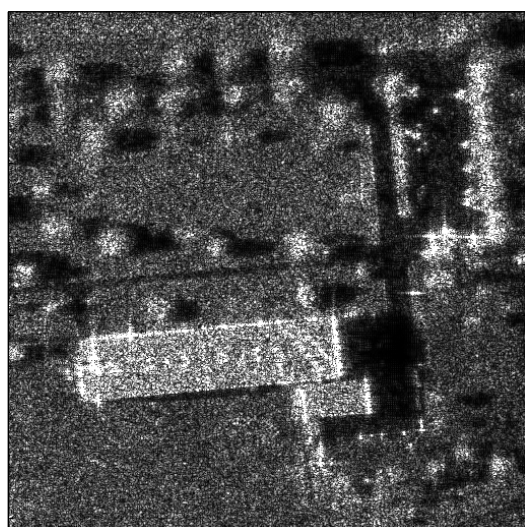
TAB. 5.1 – Temps correspondant à l'encodage et au décodage de la partie réelle de l'image extraite de la ville d'Istres (512x512) par l'algorithme WP-TCE. Processeur : Pentium III 1.8 GHz.



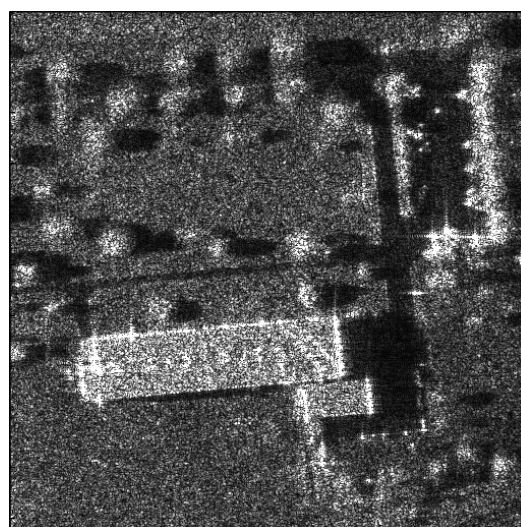
(a) Image RSO SLC originale (amplitude) extraite de la ville d'Istres. ©ONERA.



(b) Taux $\eta = 64 : 1$ (1bppc), $RSB_A = 10.23dB$.



(c) Taux $\eta = 32 : 1$ (2bppc), $RSB_A = 14.81dB$.



(d) Taux $\eta = 16 : 1$ (4bppc), $RSB_A = 22.57dB$.

FIG. 5.12 – Exemple de compression d'image SLC par l'algorithme WPTCE-IQC après transformée bi-orthogonale 9/7 des parties réelles et imaginaires de l'image RSO SLC extraite de la ville d'Istres (512x512). Seule l'amplitude de l'image est représentée.

5.4.3 Application à l'image d'amplitude et aux images test standard

Les algorithmes TCEAC-CSPQ et TCEAC-IPQ présentés au chapitre 4 s'appuient sur la compression de l'image d'amplitude par l'algorithme TCE. En considérant cette fois-ci la transformée en paquets d'ondelettes, nous avons appliqué l'algorithme WP-TCE à la compression des images d'amplitude mono-vue de type RAMSES. La recherche de la meilleure base convergeait systématiquement vers la transformée en ondelettes discrètes. Nous en déduisons que la transformée en paquets d'ondelettes n'apporte pas de gain particulier d'un point de vue compromis débit-distorsion pour la compression des images d'amplitudes. Ces résultats corroborent les conclusions de G. Mercier [62] qui a appliqué la transformée en paquet d'ondelettes sur les images de télédétection (ERS-1 et SPOT). En revanche, l'algorithme WP-TCE reste performant sur les images test standards classiques. Afin de pouvoir établir une base de comparaison par rapport aux résultats publiés dans la littérature, nous avons testé l'algorithme WP-TCE sur des images de tests classiques. Les tables 5.4.3 et 5.4.3 illustrent les performances des algorithmes de compression en terme de PSNR (*Peak Signal to Noise Ratio*) sur les images de *Barbara* et *Goldhill* [34]. Pour les faibles débits, le gain apporté par l'algorithme WP-TCE sur les images naturelles est d'environ 0.3dB par rapport à l'algorithme FAWP de F. G. Meyer [25] et de 1dB par rapport à JPEG2000. Ces résultats sont extraits d'une étude de C. Valade et al [105].

Codec	Filtre	0.25 bpp	0.5 bpp	1 bpp
SPIHT [81]	7/9	27.58	31.39	36.41
SFQ [118]	7/9	28.29	32.15	37.03
TCE [98]	7/9	27.88	31.82	36.76
JPEG2000 [93]	7/9	28.36	32.26	37.15
WP-SFQ [112]	7/9	29.25	33.12	37.69
SRWP [6]	7/9	27.87	31.53	36.41
FAWP [25]	7/9	29.12	32.82	37.24
WP-TCE	7/9	29.41	33.19	37.57

TAB. 5.2 – Comparaison des performances de l'algorithme WP-TCE en terme de PSNR(dB) pour l'image de Barbara.

Codec	Filtre	0.25 bpp	0.5 bpp	1 bpp
SPIHT [81]	7/9	30.42	32.96	36.36
TCE [98]	7/9	30.64	33.23	36.64
JPEG2000 [93]	7/9	30.51	33.21	36.53
SRWP [6]	7/9	29.65	32.33	35.77
FAWP [25]	7/9	30.39	32.94	36.25
WP-TCE	7/9	30.71	33.25	36.71

TAB. 5.3 – Comparaison des performances de l'algorithme WP-TCE en terme de PSNR(dB) pour l'image de Goldhill.

5.5 Conclusion

Nous avons consacré ce chapitre à l'étude des paquets d'ondelettes pour la compression d'images RSO SLC. A cet effet, nous avons adapté les algorithmes WECUPQ et TCE-IQC du chapitre 4 à la transformée en paquets d'ondelettes. La recherche de la meilleure base de décomposition a été réalisée suivant un critère de minimisation du compromis-débit distorsion. Nous avons ainsi adapté l'algorithme de recherche de Ramchandran et Vetterli suivant deux approches correspondant à la quantification polaire à contrainte entropique des coefficients complexes (WP-ECUPQ) et à la quantification par plan de bits des parties réelle et imaginaire (WPTCE-IQC). En terme de compromis débit-distorsion, les résultats préliminaires montrent qu'il existe un intérêt certain à utiliser la transformée en paquets d'ondelettes sur les composantes cartésiennes de l'image complexe.

Les caractéristiques fréquentielles de l'image d'amplitude étant sensiblement différentes de celles des images de partie réelle et imaginaire, l'algorithme de recherche de meilleure base de paquets d'ondelettes considère alors la transformée en ondelettes discrètes comme optimale pour la compression de l'image d'amplitude.

A la lumière de l'ensemble des méthodes de compression d'images RSO SLC proposées dans le cadre de ce manuscrit, nous consacrons le chapitre 6 à une étude comparative des performances de ces algorithmes.

Chapitre 6

Résultats et étude comparative

6.1 Introduction

Ce chapitre a pour objectif de comparer les méthodes de compression abordées dans le cadre de ce manuscrit. De nombreux critères permettent d'estimer les dégradations engendrées par un algorithme de compression et dépendent en général de l'application envisagée (classification, détection de contours, reconstruction de bâtiments, fusion optique-radar...). A titre d'exemple, G. Mercier évalue la qualité d'images RSO comprimées par un algorithme de type fractale sur une séquence multi-temporelle. Il utilise en particulier un critère basé sur les performances d'un algorithme de segmentation en région homogène (SEM) [61]. J. Mvogo a proposé d'évaluer la qualité des images RSO d'amplitude décompressées à partir d'un algorithme de segmentation multi-échelles basé sur les paquets d'ondelettes [67].

Dans cette étude, nous n'avons pas retenu d'application particulière associée à l'exploitation des données de nature complexe. En revanche, dans le cadre d'applications basées sur l'interférométrie radar, nous nous intéresserons en particulier à caractériser la distorsion introduite sur les interférogrammes reconstruits à partir d'un couple images décompressées.

Les critères d'évaluation seront comparés dans ce chapitre sur 2 scènes RSO acquises par le capteur RAMSES de l'ONERA en bande X et de résolution submétrique :

- une image SLC de taille 512x512 extraite d'une zone de la ville d'Istres.
- un couple d'images SLC de taille 1024x1024 extraites de la zone des Baux de Provence à partir desquelles nous construirons un interférogramme.

L'études de la qualité des images décompressées portera en particulier sur les méthodes proposées dans ce document et nous en rappelons ici le principe général :

- TCE-IQC : compression de la partie réelle et imaginaire de l'image SLC basée sur l'algorithme TCE de C. Tian et S.S Hemami [98].
- SPIHT-IQC : compression de la partie réelle et imaginaire de l'image SLC basée sur l'algorithme SPIHT de Said et Pearlman [81].

- WECUPQ : quantification polaire à contrainte entropique appliquée aux coefficients issus de la transformée en ondelettes de l'image SLC.
- TCEAC-CSPQ : compression de l'image d'amplitude par l'algorithme TCE et quantification scalaire de la phase conditionnellement à l'amplitude.
- TCEAC-IPQ : compression de l'image d'amplitude par l'algorithme TCE et quantification vectorielle de l'image de phase.
- WPTCE-IQC : compression de la partie réelle et imaginaire de l'image SLC basée sur l'extension de l'algorithme TCE à la transformée en paquets d'ondelettes.
- WP-ECUPQ : quantification polaire à contrainte entropique appliquée aux coefficients issus de la transformée en paquets ondelettes de l'image SLC.

6.2 Critères de qualité

6.2.1 Critères de qualité basés sur l'erreur quadratique moyenne

Nous introduisons l'ensemble des critères permettant de mesurer la qualité des images RSO SLC à partir de l'erreur quadratique moyenne. Bien qu'introduites au chapitre 3, nous rappelons ici, les notations utilisées pour décrire l'image RSO originale et décompressée.

Soit $z(m, n) = i(m, n) + j q(m, n) \in \mathbb{C}$ un pixel complexe de l'image SLC Z de taille $M \times N$ où $\{i, q\}$ correspondent respectivement à la partie réelle et imaginaire. Sous forme polaire, le pixel $z(m, n) = a(m, n)e^{j\theta(m, n)}$ où $\{a, \theta\}$ est décrit par son module (amplitude) et sa phase.

On notera de manière analogue $\hat{z}(m, n) = \tilde{i}(m, n) + j \tilde{q}(m, n) = \hat{a}(m, n)e^{j\hat{\theta}(m, n)} \in \mathbb{C}$ le pixel de l'image \hat{Z} reconstruit après décompression. Les critères étudiés sont les suivants :

Erreur quadratique moyenne sur les données complexes

$$EQM_Z = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |z(m, n) - \hat{z}(m, n)|^2 \quad (6.1)$$

Rapport signal sur bruit sur les données complexes

$$RSB_Z = 10 \log_{10} \left[\frac{\sum_{m=1}^M \sum_{n=1}^N |z(m, n)|^2}{\sum_{m=1}^M \sum_{n=1}^N |z(m, n) - \hat{z}(m, n)|^2} \right] \quad (6.2)$$

Erreur quadratique moyenne sur l'image d'amplitude

$$EQM_A = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |a(m, n) - \hat{a}(m, n)|^2 \quad (6.3)$$

Rapport signal sur bruit sur l'image d'amplitude

$$RSB_A = 10 \log_{10} \left[\frac{\sum_{m=1}^M \sum_{n=1}^N a(m, n)^2}{\sum_{m=1}^M \sum_{n=1}^N |a(m, n) - \hat{a}(m, n)|^2} \right] \quad (6.4)$$

Erreur quadratique moyenne sur l'image de partie réelle et imaginaire

$$EQM_I = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |i(m, n) - \tilde{i}(m, n)|^2 \quad (6.5)$$

$$EQM_Q = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |q(m, n) - \tilde{q}(m, n)|^2 \quad (6.6)$$

Rapport signal sur bruit sur l'image de partie réelle et imaginaire

$$RSB_I = 10 \log_{10} \left[\frac{\sum_{m=1}^M \sum_{n=1}^N |i(m, n)|^2}{\sum_{m=1}^M \sum_{n=1}^N |i(m, n) - \tilde{i}(m, n)|^2} \right] \quad (6.7)$$

$$RSB_Q = 10 \log_{10} \left[\frac{\sum_{m=1}^M \sum_{n=1}^N |q(m, n)|^2}{\sum_{m=1}^M \sum_{n=1}^N |q(m, n) - \tilde{q}(m, n)|^2} \right] \quad (6.8)$$

Distorsion de phase

Nous envisageons deux expressions : La première s'exprime par l'erreur quadratique moyenne si l'on ne considère que les données de phase indépendamment du module associé :

$$EQM_{\Theta} = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |\theta(m, n) - \hat{\theta}(m, n)|^2 \quad (6.9)$$

$$RSB_{\Theta} = 10 \log_{10} \left[\frac{\sigma_{\Theta}^2}{\sum_{m=1}^M \sum_{n=1}^N |\theta(m, n) - \hat{\theta}(m, n)|^2} \right] \quad (6.10)$$

avec $\sigma_\theta^2 = E(\Theta^2) - E(\Theta)^2 = \pi^2/3$ pour une variable uniformément répartie entre $[0; 2\pi]$.

La seconde expression de distorsion associée au terme $D_{\Theta/A}$ trouve sa justification dans le calcul (3.3). La mesure portant sur la distorsion de phase est pondérée par un le produit entre le module de l'image originale et le module de l'image reconstruite :

$$EQM_{\Theta/A} = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N \left[2a(m, n) \hat{a}(m, n) [1 - \cos(\theta(m, n) - \hat{\theta}(m, n))] \right] \quad (6.11)$$

Il est d'ailleurs important de noter la relation suivante qui se déduit directement des expressions précédentes :

$$EQM_Z = EQM_I + EQM_Q = EQM_A + EQM_{\Theta/A} \quad (6.12)$$

La réécriture du terme EQM_Z permet de caractériser la distorsion sur l'image complexe suivant des termes d'erreur quadratique EQM_I et EQM_Q liés à la représentation de l'image sous forme de partie réelle et imaginaire. Cette relation s'exprime de manière équivalente par les termes EQM_A et $EQM_{\Theta/A}$ associés à la représentation de l'image sous forme d'amplitude et de phase. Ainsi, l'expression d'erreur quadratique moyenne de phase EQM_Θ ¹ (équation (6.9)) ne trouve pas de justification directement liée à une caractérisation de la distorsion dans le plan complexe.

6.2.2 Corrélation Complexe spatiale

De l'avis de nombreux spécialistes, le critère d'erreur quadratique moyenne classiquement utilisé dans la littérature ne suffit pas à évaluer les performances de système de compression. Les critères globaux comme l'erreur quadratique moyenne ont l'inconvénient de ne pas rendre compte des dégradations locales. Des pertes d'informations portant sur une zone particulière de l'image (des routes ou des rivières par exemple) n'apparaîtront pas sur les critères globaux. Un critère plus local proposé par Ives et al. [44] [24] permet ainsi de dresser une carte de corrélation complexe (en fait le terme approprié serait "cohérence") en calculant un coefficient de corrélation local $c(x, y)$ à partir d'une fenêtre de L pixels voisins. Le calcul de ce coefficient est donné par :

$$c(m, n) = \frac{\sum_i \sum_j z(i, j) \hat{z}^*(i, j)}{\sqrt{\sum_i \sum_j |z(i, j)|^2 \sum_i \sum_j |\hat{z}(i, j)|^2}}, \quad c(m, n) \in [0, 1] \quad (6.13)$$

où (i, j) représentent les indices des pixels voisins localisés en (m, n) . Une cohérence associée ($|c(x, y)|$) égale à 1 est synonyme d'une reconstruction parfaite.

La définition (6.13) présuppose en outre que la moyenne des parties réelle et imaginaire du voisinage est nulle. Cette hypothèse, utilisée entre autre pour caractériser le chatolement pleinement développé, se vérifie en pratique pour des images RSO (cf chapitre 1, section 3) et pour un voisinage de pixels suffisant.

¹L'erreur de phase étant comprise entre $[-\pi; \pi]$, il faudra se prévaloir des phénomènes de périodicité dans le calcul de la différence de phase

On pourra remarquer que l'expression du coefficient de corrélation complexe est identique à l'expression du calcul d'interférogramme, du moins dans sa forme mathématique. Les arguments ne sont cependant pas les mêmes. En effet, ici z et \hat{z}^* correspondent aux pixels des images originales et reconstruites après décompression. Dans le cas d'un interférogramme, il s'agit plus spécifiquement de deux images SLC différentes acquises dans des conditions permettant une exploitation de l'information portée par la différence de phase.

La cohérence de la carte de corrélation complexe correspond ici plus spécifiquement à une métrique de qualité dans le sens où elle fournit une évaluation visuelle sur la "distance locale" entre les pixels complexes de l'image originale et décompressée.

Coefficient de corrélation complexe moyen

Il s'agit d'un critère permettant de définir une "distance" globale entre l'image originale et l'image décompressée. Le coefficient est donné simplement par un moyennage de la cohérence (formule (6.13)).

$$\rho = \frac{1}{M * N} \sum_{m=1}^M \sum_{n=1}^N |c(m, n)| \quad (6.14)$$

6.2.3 Histogrammes

La dégradation des statistiques après décompression peut avoir un impact significatif sur les traitements effectués sur l'image après décompression. La conservation des hypothèses et modèles statistiques demeure donc une priorité. Nous étudierons donc en particulier :

- Les histogrammes sur le module de l'image originale et après reconstruction.
- Les histogrammes des parties réelle et imaginaire de l'image originale et après reconstruction (en pratique seuls les histogrammes de la partie réelle seront présentés, la partie imaginaire suivant la même loi).

6.2.4 Critères de qualité RSO

Coefficient de variation

Le coefficient de variation déterminé empiriquement sur les zones homogènes de l'image détectée et permet de fournir une mesure du chatoiement dans l'analyse de bruit multiplicatif. Son expression est donnée par :

$$\gamma = \frac{\sigma}{\mu} \quad (6.15)$$

où μ et σ correspondent à la moyenne et à l'écart-type sur la zone homogène de l'image. Pour une image SLC ce paramètre est proche de la valeur 1.

Comparaison des spectres en distance et en azimut

Il s'agit ici de comparer les spectres des images originales et après reconstruction. Le spectre est calculé à partir du module au carré de la transformée de Fourier discrète réalisée sur les données complexes. Pour une meilleure visualisation en distance et en

azimut, les spectres seront moyennés respectivement suivant les directions azimutales et en distance.

6.2.5 Critères de qualité sur les interférogrammes

Comme nous l'avons vu au chapitre 1, l'exploitation des données de phase des images SLC par interférométrie n'a de sens qu'en "différentiel". Les interférogrammes disposant d'information à l'origine d'application de mesure d'élévation, de mesure de mouvement de terrains, nous proposons un critère d'évaluation permettant de juger de la qualité de l'interférogramme obtenu après décompression d'un couple d'image SLC. Soient :

- $\{z_1, z_2\}$ un couple images SLC acquises dans des conditions permettant de construire un interférogramme I suivant la formule (1.22) et dont nous rappelons ici l'expression :

$$I = \frac{\sum_{i=1}^L z_{1i} z_{2i}^*}{\sqrt{\sum_{i=1}^L |z_{1i}|^2 \sum_{i=1}^L |z_{2i}|^2}} \quad (6.16)$$

- $\{\hat{z}_1, \hat{z}_2\}$ un couple images SLC reconstruites après décompression. On notera \hat{I} l'interférogramme reconstruit.

Il est important de préciser que nous nous plaçons dans le cadre où les images SLC du couple interférométrique ont été compressées de manière **indépendante**. En effet, la réalisation d'un système de compression conjoint d'un couple interférométrique sort du cadre de cette thèse et dépend des conditions d'acquisition des images et du capteur (mono-statique, bi-statique, synthèse RSO à bord ou pas ...). Il n'en demeure pas moins que l'exploitation de la forte corrélation des images issues d'un couple interférométrique permettrait sans aucun doute une compression plus efficace.

En résumé, l'objectif de cette démarche consiste simplement à évaluer les performances des algorithmes de compression en analysant la distorsion introduite sur la différence de phase d'un interférogramme reconstruit à partir d'un couple d'images SLC comprimées indépendamment l'un de l'autre.

Corrélation complexe des interférogrammes

Nous proposons d'appliquer sur les interférogrammes, le calcul d'une carte de corrélation complexe. En d'autres termes, il s'agit de mettre en oeuvre un critère de caractérisation local de la qualité de l'interférogramme reconstruit en réalisant une carte de corrélation complexe des interférogrammes (ou par abus de langage de réaliser l' "interférogramme de l'interférogramme"). En considérant une fenêtre de moyennage autour du pixel (x, y) , le coefficient de corrélation de l'interférogramme est donné par :

$$d(x, y) = \frac{\sum_{i=1}^L I(i, j) \hat{I}^*(i, j)}{\sqrt{\sum_{i=1}^L |I(i, j)|^2 \sum_{i=1}^L |\hat{I}(i, j)|^2}}, \quad d(x, y) \in [0, 1] \quad (6.17)$$

où I et \hat{I} correspondent respectivement à l'interférogramme original et reconstruit à partir des images décompressées.

La cohérence $|d|$ sera d'autant plus proche de 1 que la qualité de reconstruction de l'interférogramme issue des images décompressées sera bonne. Cette mesure fournira donc un indicateur fiable pour déterminer les taux de compression à partir desquels les interférogrammes obtenus après reconstruction demeurent exploitables.

6.3 Compression de la scène "Istres"

L'image d'Istres de taille 512x512 a été compressée avec des taux de $\eta = 32 : 1$, $\eta = 16 : 1$, $\eta = 10.6 : 1$ correspondant respectivement à des débits de 2, 4 et 6 bits par pixel complexe. La scène originale est représentée par des valeurs de type "float" à 32 bits pour la partie réelle et 32 bits pour la partie imaginaire (soit 64 bits par pixel complexe). La transformée en ondelettes et paquets d'ondelettes est réalisée à partir de la transformée biorthogonale 9/7 [5] implémentée sous forme de structure "Lifting" [89] [90] [42]. Cette technique de factorisation implémente la transformée bi-orthogonale sous forme d'une suite finie d'étages de filtres de prédiction (*Prediction step*) et mise à jour (*Update step*). En termes de complexité de calcul, cette méthode a l'avantage d'être plus efficace que l'implémentation sous forme de banc de filtres.

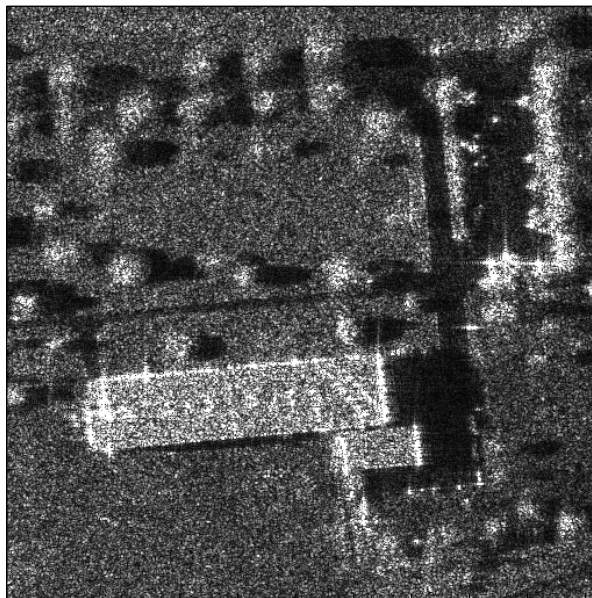


FIG. 6.1 – Image d'amplitude extraite d'une zone de la ville d'Istres(512x512).©ONERA

6.3.1 Critères d'erreur quadratique

Codec	$EQM_I + j.EQM_Q$ ($\times 10^{-4}$)	EQM_A ($\times 10^{-4}$)	$EQM_{A/\Theta}$ ($\times 10^{-4}$)	EQM_{Θ}	EQM_Z ($\times 10^{-4}$)
TCE-IQC	7.35 + j 7.35	6.80	7.90	0.8896	14.70
SPIHT-IQC	8.12 + j 8.11	7.39	8.84	0.9125	16.23
WECUPQ	21.04 + j 21.09	23.25	18.87	0.8565	42.13
TCEAC-CSPQ	11.47 + j 10.11	9.81	11.76	0.5944	21.58
TCEAC-IPQ	24.57 + j 24.36	9.81	39.12	0.3693	48.94
WPTCE-IQC	4.65 + j 4.62	4.291	4.98	0.6126	9.27
WP-ECUPQ	12.48 + j 12.45	11.90	13.03	0.7245	24.93

TAB. 6.1 – Performances des algorithmes de compression obtenues en terme d'erreur quadratique moyenne à 2 bits par pixel complexe.

Codec	$RSB_I + j.RSB_Q$ (dB)	RSB_A (dB)	RSB_{Θ} (dB)	RSB_Z (dB)
TCE-IQC	9.58 + j 9.35	12.81	11.69	9.47
SPIHT-IQC	9.15 + j 8.92	12.45	11.58	9.03
WECUPQ	5.01 + j 4.77	7.47	11.86	4.89
TCEAC-CSPQ	7.65 + j 7.96	11.22	13.44	7.80
TCEAC-IPQ	4.34 + j 4.14	11.22	15.75	4.24
WPTCE-IQC	11.56 + j 11.36	14.81	13.31	11.47
WP-ECUPQ	7.28 + j 7.06	10.38	12.59	7.17

TAB. 6.2 – Performances des algorithmes de compression obtenues en terme de Rapport Signal sur Bruit à 2 bits par pixel complexe.

Codec	$EQM_I + j.EQM_Q$ ($\times 10^{-4}$)	EQM_A ($\times 10^{-4}$)	$EQM_{A/\Theta}$ ($\times 10^{-4}$)	EQM_{Θ}	EQM_Z ($\times 10^{-4}$)
TCE-IQC	1.95 + j 1.94	1.82	2.07	0.3358	3.90
SPIHT-IQC	2.15 + j 2.16	2.02	2.29	0.3591	4.31
WECUPQ	1.98 + j 1.97	1.90	2.05	0.2651	3.96
TCEAC-CSPQ	3.41 + j 3.05	2.97	3.49	0.5038	6.46
TCEAC-IPQ	6.91 + j 6.72	2.97	10.67	0.0882	13.63
WPTCE-IQC	0.71 + j 0.71	0.72	0.70	0.1185	1.42
WP-ECUPQ	1.10 + j 1.10	1.07	1.13	0.1797	2.21

TAB. 6.3 – Performances des algorithmes de compression obtenues en terme d'erreur quadratique moyenne à 4 bits par pixel complexe.

Codec	$RSB_I + j.RSB_Q$ (dB)	RSB_A (dB)	RSB_{Θ} (dB)	RSB_Z (dB)
TCE-IQC	15.34 + j 15.11	18.53	15.92	15.23
SPIHT-IQC	14.91 + j 14.66	18.08	15.63	14.79
WECUPQ	15.27 + j 15.05	18.34	16.95	15.16
TCEAC-CSPQ	12.91 + j 13.16	16.41	14.17	13.03
TCEAC-IPQ	9.85 + j 9.73	16.41	21.73	9.79
WPTCE-IQC	19.71 + j 19.49	22.57	20.45	19.60
WP-ECUPQ	17.81 + j 17.58	20.83	18.64	17.70

TAB. 6.4 – Performances des algorithmes de compression obtenues en terme de Rapport Signal sur Bruit à 4 bits par pixel complexe.

Codec	$EQM_I + j.EQM_Q$ ($\times 10^{-4}$)	EQM_A ($\times 10^{-4}$)	$EQM_{A/\Theta}$ ($\times 10^{-4}$)	EQM_{Θ}	EQM_Z ($\times 10^{-4}$)
TCE-IQC	0.50 + j 0.51	0.49	0.52	0.1101	1.01
SPIHT-IQC	0.55 + j 0.55	0.53	0.57	0.1179	1.10
WECUPQ	0.52 + j 0.52	0.51	0.53	0.0995	1.04
TCEAC-CSPQ	0.83 + j 0.83	0.81	0.85	0.0764	1.66
TCEAC-IPQ	1.85 + j 1.99	0.81	3.04	0.0220	3.85
WPTCE-IQC	0.15 + j 0.15	0.15	0.15	0.0341	0.31
WP-ECUPQ	0.20 + j 0.20	0.20	0.20	0.0458	0.40

TAB. 6.5 – Performances des algorithmes de compression obtenues en terme d'erreur quadratique moyenne à 6 bits par pixel complexe.

Codec	$RSB_I + j.RSB_Q$ (dB)	RSB_A (dB)	RSB_{Θ} (dB)	RSB_Z (dB)
TCE-IQC	21.20 + j 20.96	24.23	20.77	21.08
SPIHT-IQC	20.79 + j 20.56	23.86	20.47	20.6830
WECUPQ	21.03 + j 20.81	24.02	21.21	20.92
TCEAC-CSPQ	19.06 + j 18.84	22.06	22.35	18.95
TCEAC-IPQ	15.56 + j 15.01	22.06	27.73	15.29
WPTCE-IQC	26.27 + j 26.07	29.18	25.86	26.17
WP-ECUPQ	25.17 + j 24.92	28.09	24.58	25.05

TAB. 6.6 – Performances des algorithmes de compression obtenues en terme de Rapport Signal sur Bruit à 6 bits par pixel complexe.

6.3.2 Analyse des résultats

- Quel que soit le débit, l'algorithme WPTCE-IQC permet d'obtenir l'EQM la plus faible sur la partie réelle et imaginaire de l'image. Il en est de même concernant l'EQM mesurée sur l'image d'amplitude et sur l'image SLC dans son ensemble. La transformée en paquets d'ondelettes appliquée à la compression de la partie réelle et imaginaire de l'image offre donc un réel intérêt de ce point de vue. Par rapport à la TCE-IQC basée sur la transformée en ondelette le gain en terme de RSB varie de 2dB pour les bas-débits (2 bppc) jusqu'à 4 à 5dB pour des débits plus élevés.
- Les performances de l'algorithme WP-ECUPQ sont relativement proches de celle de la méthode WPTCE-IQC. L'écart tend à se réduire entre les méthodes à mesure que le débit augmente. En effet, l'algorithme TCE utilise une quantification par plan de bits équivalente à un quantificateur scalaire à zone morte de paramètre $\delta = 0$ (voir chapitre 2). Parisot et al ont ainsi montré que ce quantificateur tend à être optimal pour les sources à distribution Laplacienne à bas débit [10]. En revanche pour des débits élevés, le quantificateur scalaire uniforme ($\delta = 0.5$) tend à être optimal [87]. C'est justement à partir de cette hypothèse que le quantificateur polaire à contrainte entropique de la méthode WP-ECUPQ a été construit. Il est donc naturel que cette méthode offre de meilleures performances pour des débits supérieurs.
- L'application de l'algorithme SPIHT de Said et Pearlman [81] à la compression de la partie réelle et imaginaire de l'image offre des performances légèrement inférieures aux algorithmes TCE-IQC et WECUPQ. Il semble ainsi moins efficace de quantifier les coefficients d'ondelettes par des techniques de quantification du type *zero-tree*.
- L'algorithme TCEAC-IPQ montre que la quantification de la phase indépendamment du module ne permet pas d'obtenir un rapport signal sur bruit satisfaisant. En effet, cette méthode privilégie la minimisation des termes d'erreur quadratique EQM_A et EQM_Θ respectivement associés à l'amplitude et à la phase sans prendre en compte la distorsion globale du vecteur à 2 dimensions. Nous obtenons ainsi le minimum du terme d'erreur quadratique de phase EQM_Θ . En revanche, le terme $EQM_{\Theta/A}$ (qui entre en compte dans l'expression EQM_Z) est maximal. Cette méthode permet néanmoins de contrôler séparément le débit associé à la compression de l'image d'amplitude de celui alloué à la quantification de l'image de phase.
- L'algorithme TCEAC-CSPQ offre un compromis intéressant. Compte tenu de la nature déstructurée et fortement décorrélée de l'image de phase, seule l'image d'amplitude est compressée à partir d'une analyse multi-résolution. On obtient ainsi des performances en terme de rapport signal sur bruit global inférieures aux méthodes basées sur la transformée en ondelettes de la partie réelle et imaginaire de l'image. Cependant à la différence de la méthode TCEAC-IPQ, nous avons optimisé un quantificateur minimisant le terme de distorsion $EQM_{\Theta/A}$. Avec un gain supérieur à 3dB sur le terme d'erreur quadratique EQM_Z , cette méthode permet également de contrôler séparément les débits alloués à la compression de l'image d'amplitude et de la phase.

6.3.3 Carte de corrélation complexe

Cette section est consacrée à l'étude du critère local de corrélation complexe qui permet de donner une caractérisation "visuelle" de la qualité de l'image reconstruite tout en prenant en compte l'aspect complexe de l'image et donc l'information de phase. Les cartes de cohérence ont été réalisées à partir d'une fenêtre de calcul de 5x5 pixels. Afin de faciliter l'interprétation et d'établir une base de comparaison, la carte de cohérence est tracée pour les valeurs comprises entre 0.95 et 1 (saturation à 0.95 pour les valeurs inférieures à ce seuil). L'illustration des résultats est donnée par l'ensemble des figures 6.2 et 6.3.

L'analyse des résultats montre qu'une fois encore l'algorithme WPTCE-IQC permet d'obtenir la meilleure qualité d'image après décompression. La compression à faible débit montre en particulier une cohérence plus élevée sur les structures fortement réfléchives de l'image comme le toit des bâtiments, les coins réflecteurs. De manière similaire à une compression par régions d'intérêt, ces structures seront donc reconstruites en priorité avec une meilleure qualité, au détriment des zones homogènes de végétation caractérisées par le chatoiement pleinement développé. On notera également une faible cohérence de reconstruction sur les zones d'ombre de l'image.

Pour des débits suffisamment élevés, la méthode WP-ECUPQ se différencie essentiellement de la méthode WPTCE-IQC sur la restitution des zones d'ombres et du chatoiement. La carte de cohérence tend alors à s'"homogénéiser" à mesure que le débit augmente.

On remarquera le faible contraste de la carte de cohérence associée à la méthode TCEAC-IPQ. La carte de cohérence est ici plus "homogène" ce qui se traduit par le fait que la qualité de reconstruction de l'image est uniforme sur l'ensemble des pixels et ce, quel que soit le taux de compression considéré. La qualité de reconstruction du toit du bâtiment est ainsi très proche de celle de la végétation. Ceci s'explique par le fait que la distorsion associée à l'image de phase compressée indépendamment de l'image d'amplitude est équitablement répartie sur tous pixels.

En comparant avec la méthode TCEAC-CSPQ, la construction du quantificateur scalaire de la phase conditionnellement à l'amplitude montre bien que la reconstruction de la phase est d'autant meilleure que l'énergie du pixel associé est importante.

Les tables 7, 8, 9 fournies en Annexes permettent de rendre compte du critère de corrélation moyen calculé sur l'image de cohérence. On retrouve ainsi le fait que les techniques basées sur les paquets d'ondelettes WPTCE-IQC permettent d'obtenir la meilleure qualité dans le sens où le critère s'approche de la valeur 1.

Le coefficient de variation mesuré sur une partie homogène de l'image (végétation rase) traduit le fait que les caractéristiques du chatoiement pleinement développé semblent bien conservées pour les débits de 4 et 6 bits par pixel complexe. En revanche, pour un débit global de 2 bits par pixel complexe, l'apparition d'un bruit de fond altère le chatoiement.

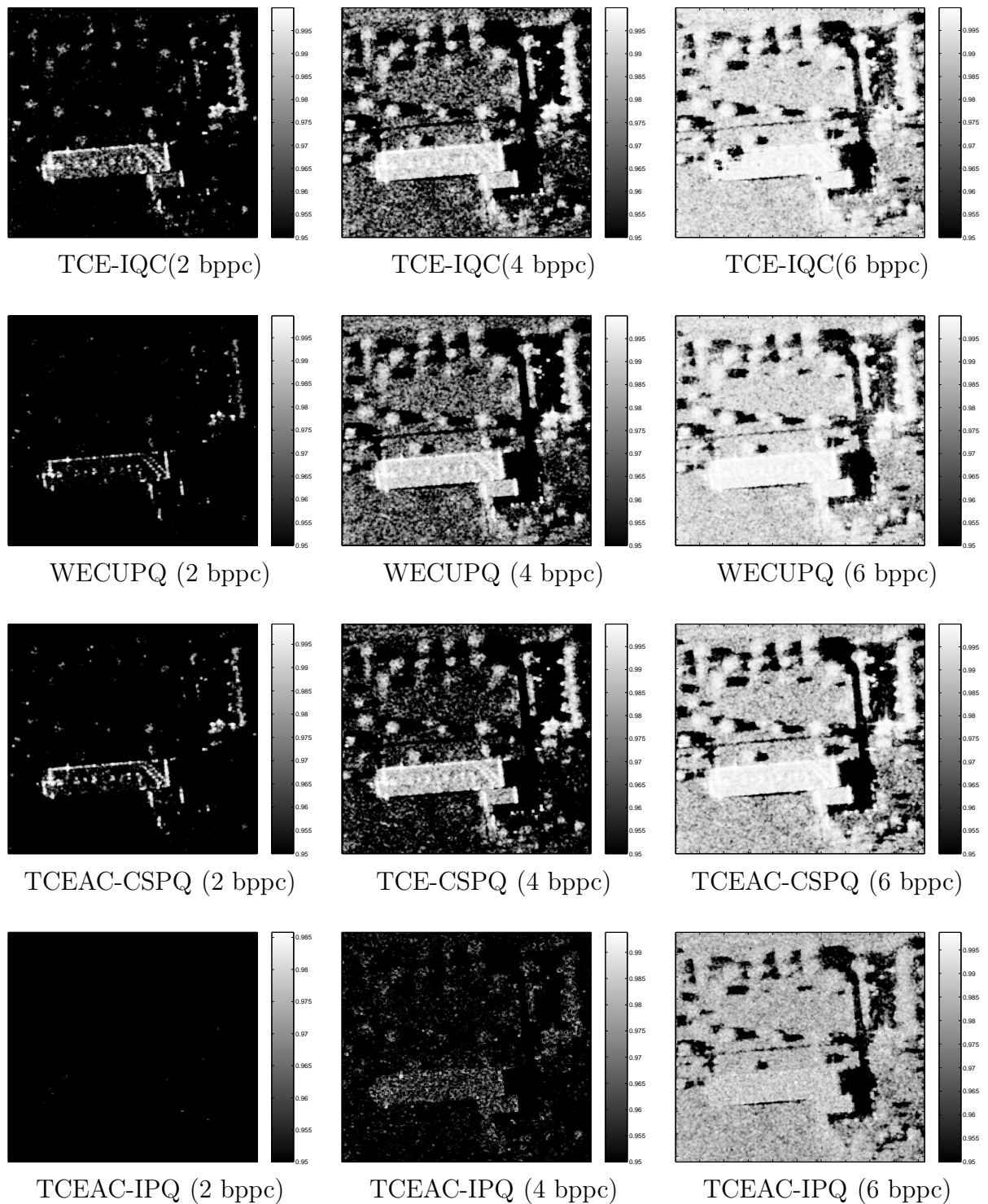


FIG. 6.2 – Carte de corrélation complexe calculée entre l'image SLC originale et l'image décompressée.

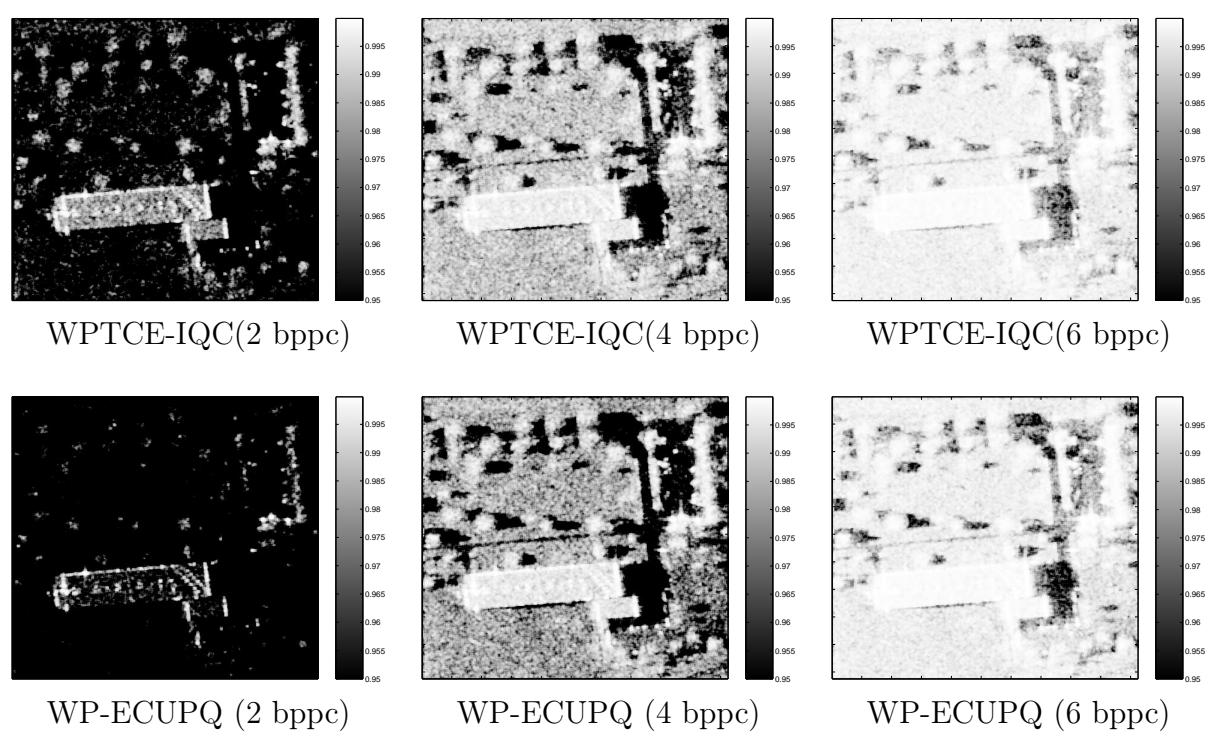


FIG. 6.3 – Carte de corrélation complexe calculée entre l'image SLC originale et l'image décompressée.

6.3.4 Histogramme de l'image d'amplitude et de l'image de partie réelle

Cette partie se consacre à la comparaison des histogrammes de l'image en fonction du taux de compression et de l'algorithme utilisé. Les figures 6.4 et 6.5 illustrent respectivement les histogrammes de l'image d'amplitude et de la partie réelle de l'image SLC.

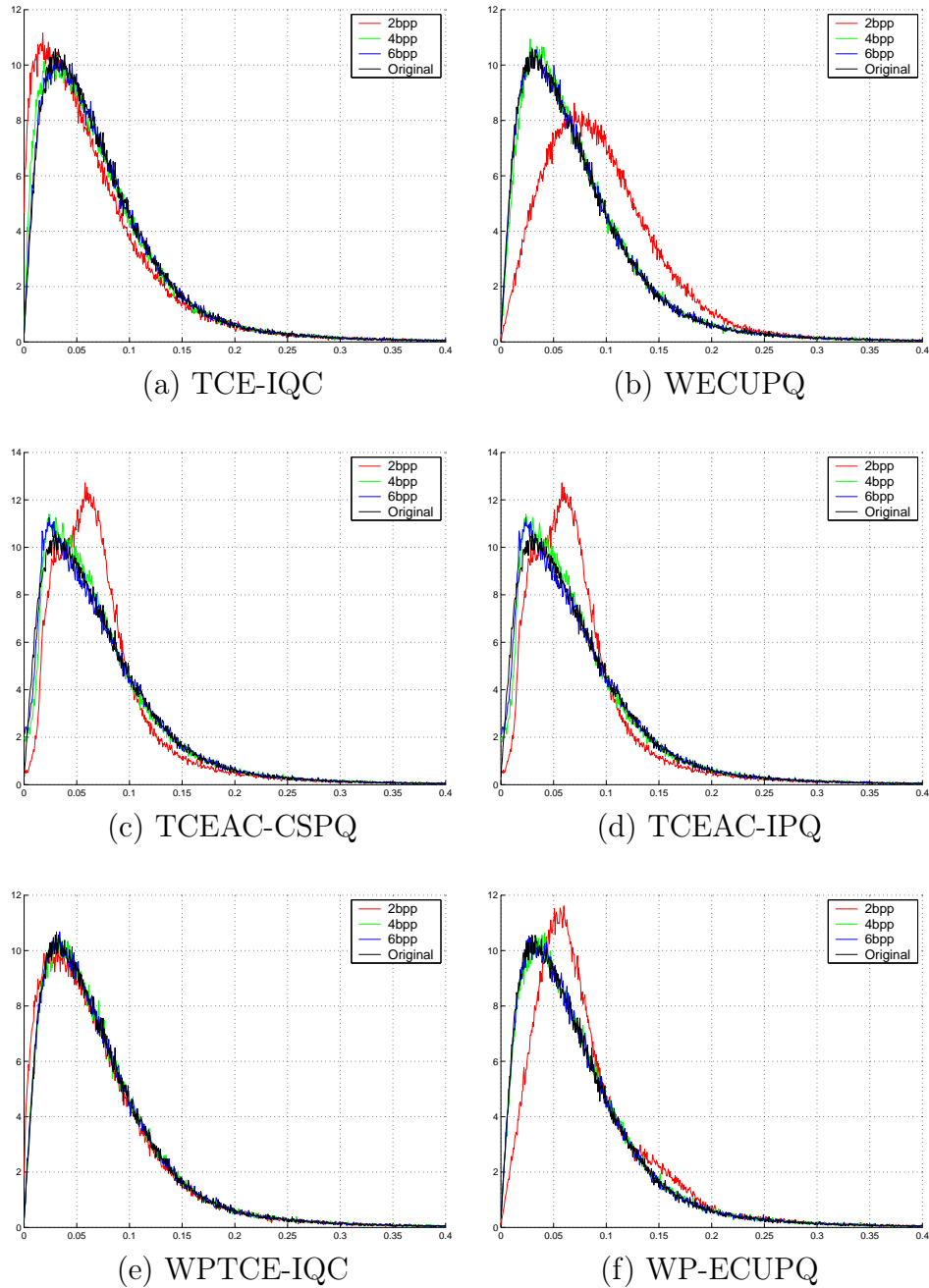


FIG. 6.4 – Histogrammes d'amplitude avant/après compression pour des débits de 2, 4 et 6 bits par pixel complexe.

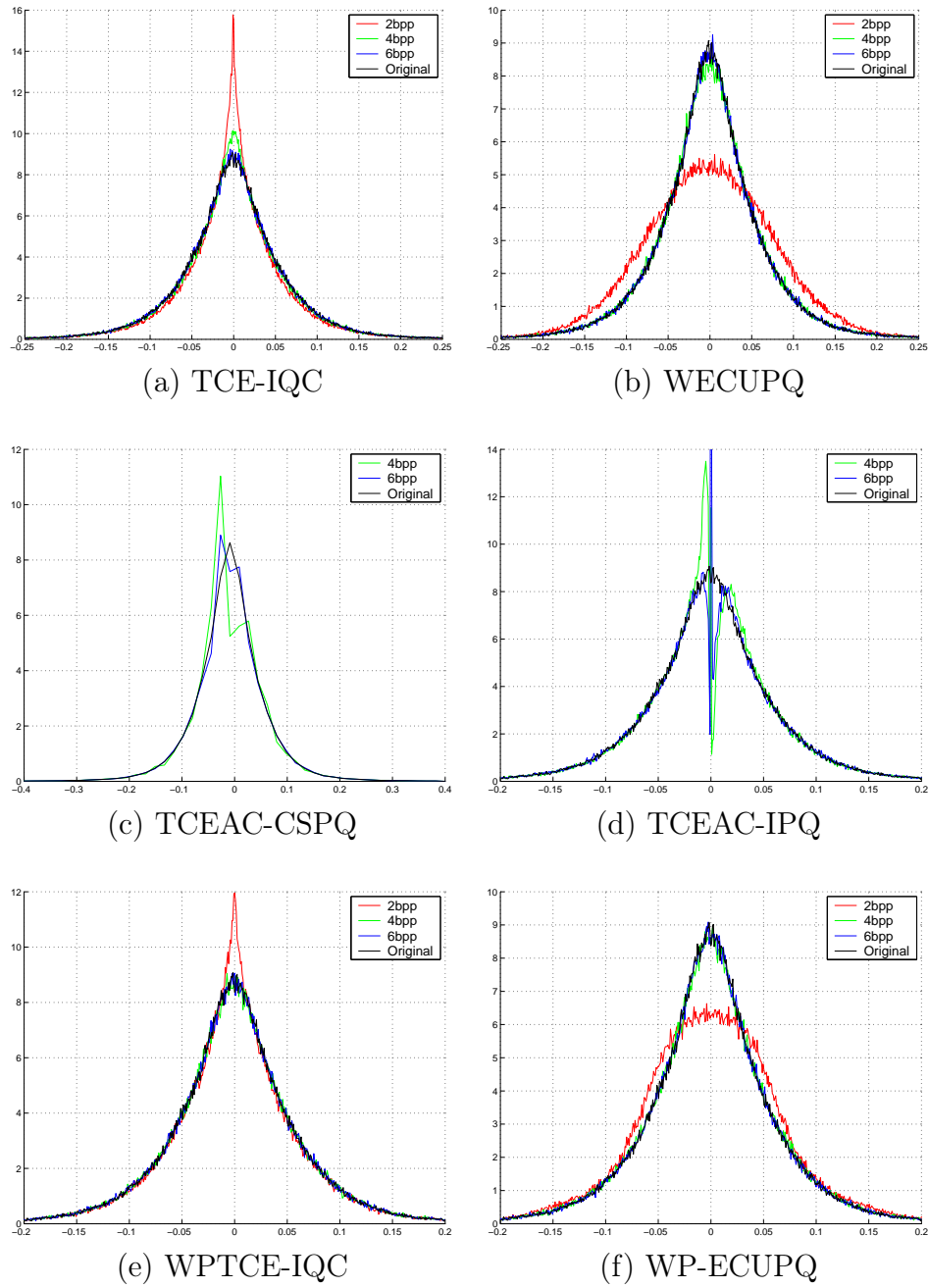


FIG. 6.5 – Histogrammes de la partie réelle avant/après compression pour des débits de 2, 4 et 6 bits par pixel complexe.

Analyse des résultats

- L'ensemble de figures 6.4 montre clairement que pour les débits de 4 et 6 bits par pixel complexe, les histogrammes de l'image d'amplitude sont similaires quelle que soit la technique de compression utilisée. Notons à ce propos, que les méthodes TCEAC-CSPQ et TCEAC-IPQ autorisent le fait de compresser l'image d'amplitude à différents débits indépendamment de ceux alloués à l'image de phase. Pour un débit global, il est possible d'améliorer la qualité de l'image d'amplitude (en allouant plus de débit) au détriment de celui de l'image de phase.
- Alors que la queue de l'histogramme semble bien conservée pour un débit de 2 bits par pixel complexe, on remarquera cependant que seule la méthode WPTCE-IQC semble conserver le mode et la forme générale de l'histogramme (figure 6.4(e)).
- Concernant les histogrammes de la partie réelle, les figures 6.5 montrent des comportements très différents de l'histogramme de l'image reconstruite en fonction de l'algorithme choisi. Pour les débits de 4 et 6 bppc, les histogrammes associés aux techniques basées sur la compression des images de partie réelle et imaginaires semblent bien conserver la distribution des pixels de l'image. En revanche, pour un débit à 2 bppc l'algorithme de quantification TCE modifie sensiblement l'histogramme des pixels avec une forte concentration de valeurs autour de 0 (figures 6.5 (a) et(e)). Cette observation s'explique par le fait qu'à faible débit, un quantificateur par plan de bits ne peut prendre en compte qu'un nombre restreint de plan de bits (typiquement 4 ou 5). En raison de la structure même du quantificateur à zone morte équivalent, la majorité des coefficients d'ondelettes seront ainsi quantifiés par la valeur nulle. Après inversion de la transformée, il est donc naturel qu'un nombre important de pixels reconstruits aient une valeur proche de zéro.
- Les artefacts observés sur les histogrammes (figures 6.5 (c) et(d)) mettent en lumière un défaut lié à la compression bas-débit de l'image d'amplitude. En effet, à fort taux de compression, une faible proportion de pixels de faible amplitude peuvent être reconstruits avec une valeur négative proche de 0. Or, une image d'amplitude avec des pixels à valeur négatives n'a pas de sens. Le fait de ramener l'ensemble de ces pixels à la valeur nulle (ou la valeur absolue) crée les pics visibles sur les histogrammes de la partie réelle.

6.3.5 Spectre en distance et en azimuth

Les figures 6.6 et 6.7 illustrent respectivement le spectre en distance et en azimuth de l'image SLC reconstruite après décompression en fonction de l'algorithme de compression et du débit d'encodage des données.

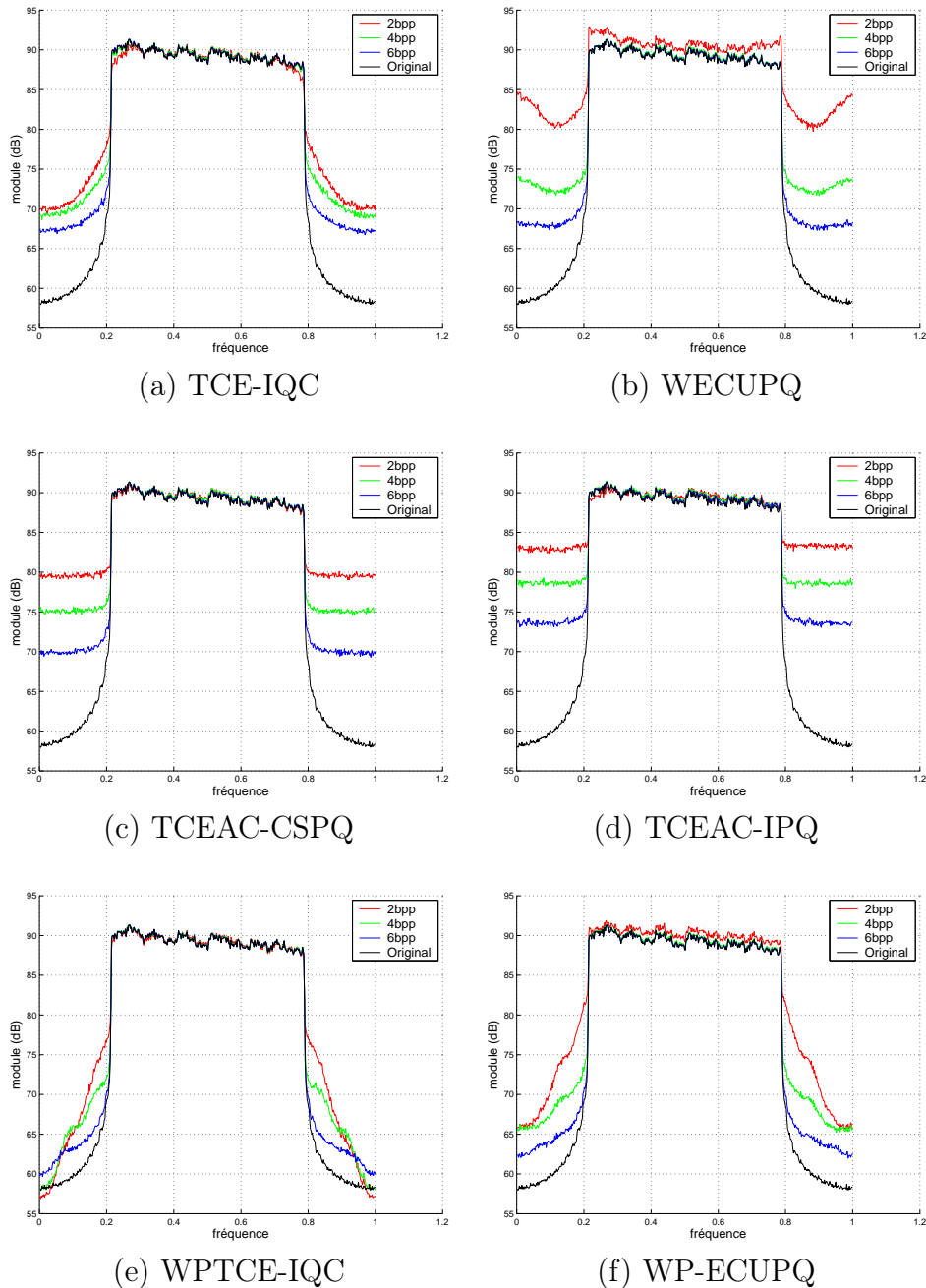


FIG. 6.6 – Spectres en distance de l'image extraite de la ville d'Istres (échelle logarithmique) en fonction de l'algorithme de compression et du débit.

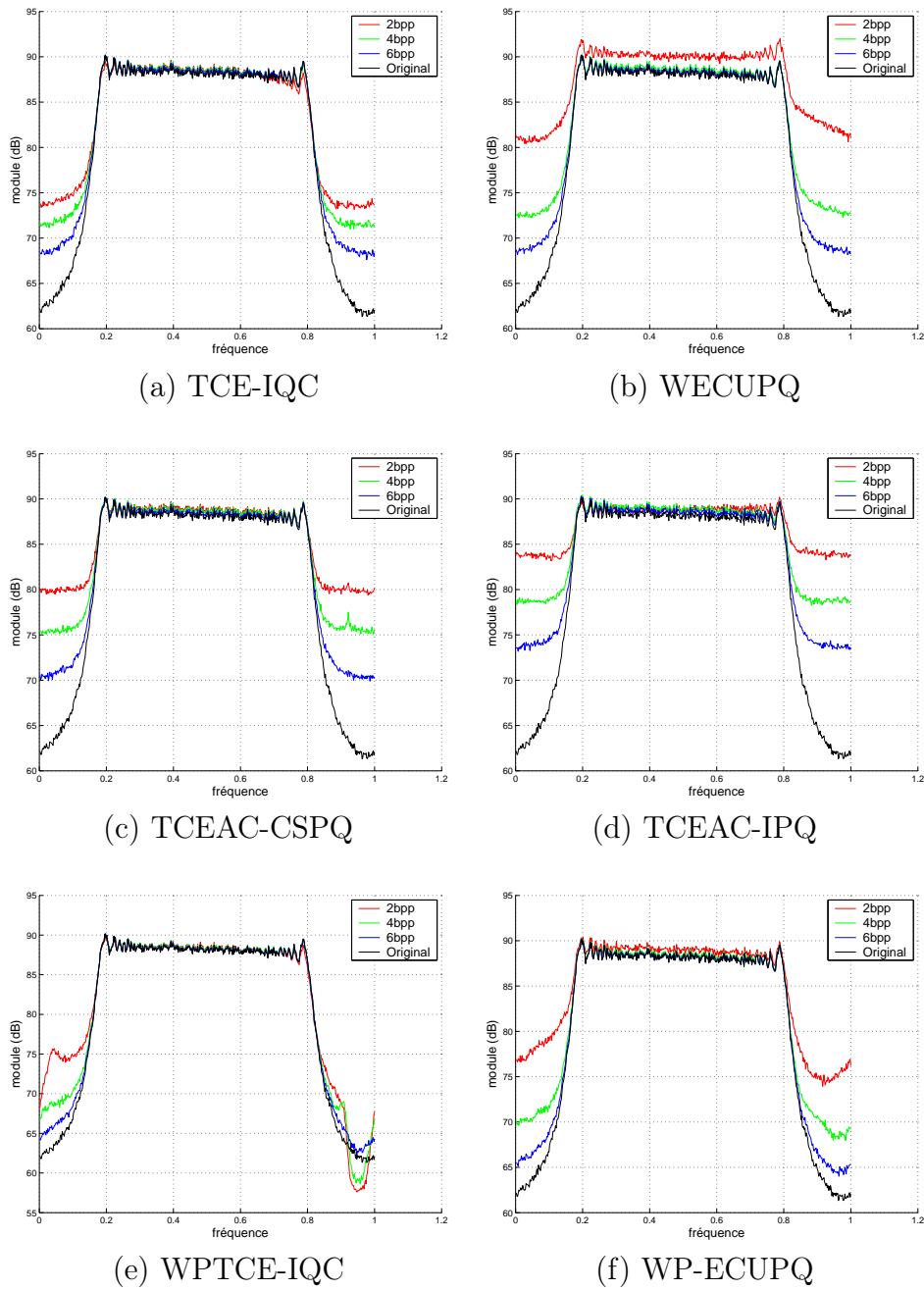


FIG. 6.7 – Spectres en azimut de l’image extraite de la ville d’Istres (échelle logarithmique) en fonction de l’algorithme de compression et du débit.

6.3.6 Analyse des résultats

Les figures 6.6 et 6.7 montrent en particulier que la forme de la bande centrale du spectre "plein" de l'image SLC en azimuth et en distance est conservée. La différence de qualité s'observe essentiellement sur la bande "morte" à faible énergie (due à un sur-échantillonnage des données RSO). On remarque en particulier que les algorithmes par paquets d'ondelettes s'affranchissent du partitionnement fréquentiel dyadique de la transformée en ondelettes discrètes et permettent de reconstituer, avec une meilleure qualité, cette partie du spectre.

6.4 Compression de la scène "Baux de Provence"

Nous apportons ici un éclairage différent sur l'évaluation de la qualité de l'image reconstruite. La scène étudiée est extraite d'un couple interférométrique (Z_1, Z_2) de la zone de Baux de Provence. L'objectif de cette approche consiste à évaluer les performances des algorithmes de compression en comparant l'interférogramme original avec celui obtenu après décompression des images Z_1 et Z_2 . La figure 6.8 montre l'image d'amplitude Z_1 à partir de laquelle nous avons construit l'interférogramme représenté par l'image de cohérence 6.9(a) et l'image de différence de phase 6.9(b).

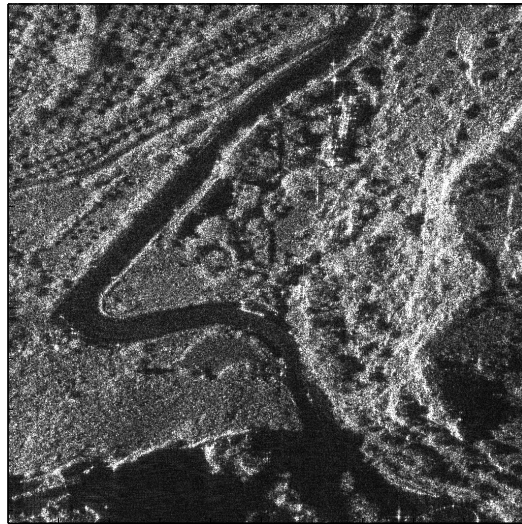


FIG. 6.8 – Image d'amplitude extraite de la scène des Baux de Provence (image Z_1 du couple interférométrique). ©ONERA

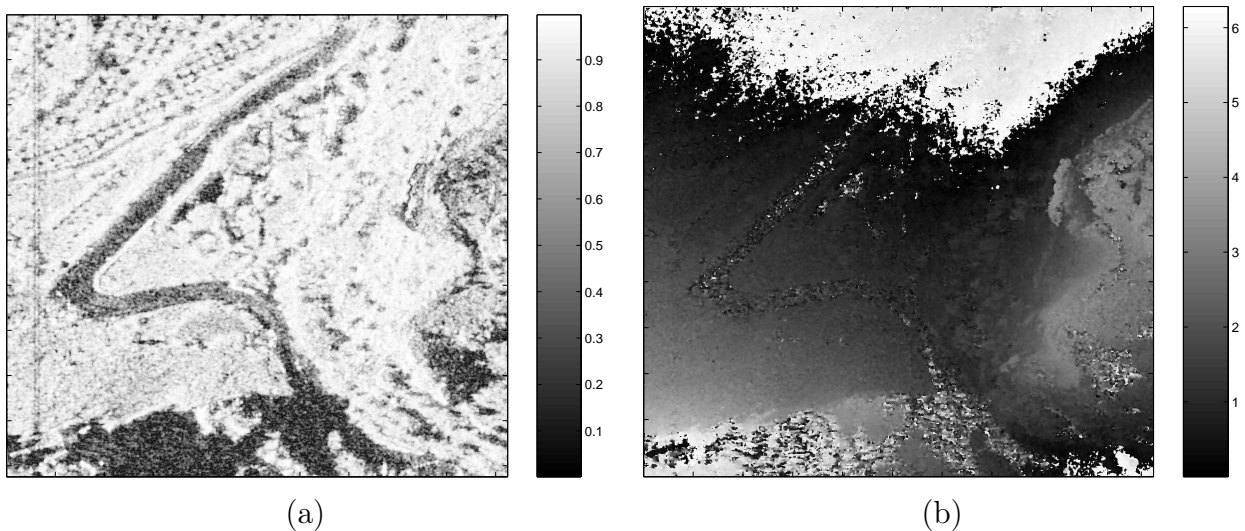


FIG. 6.9 – Cohérence et différence de phase de l'interférogramme original (fenêtre 5×5) de la zone extraite des Baux de Provence.

6.4.1 Interférogrammes reconstruits après décompression

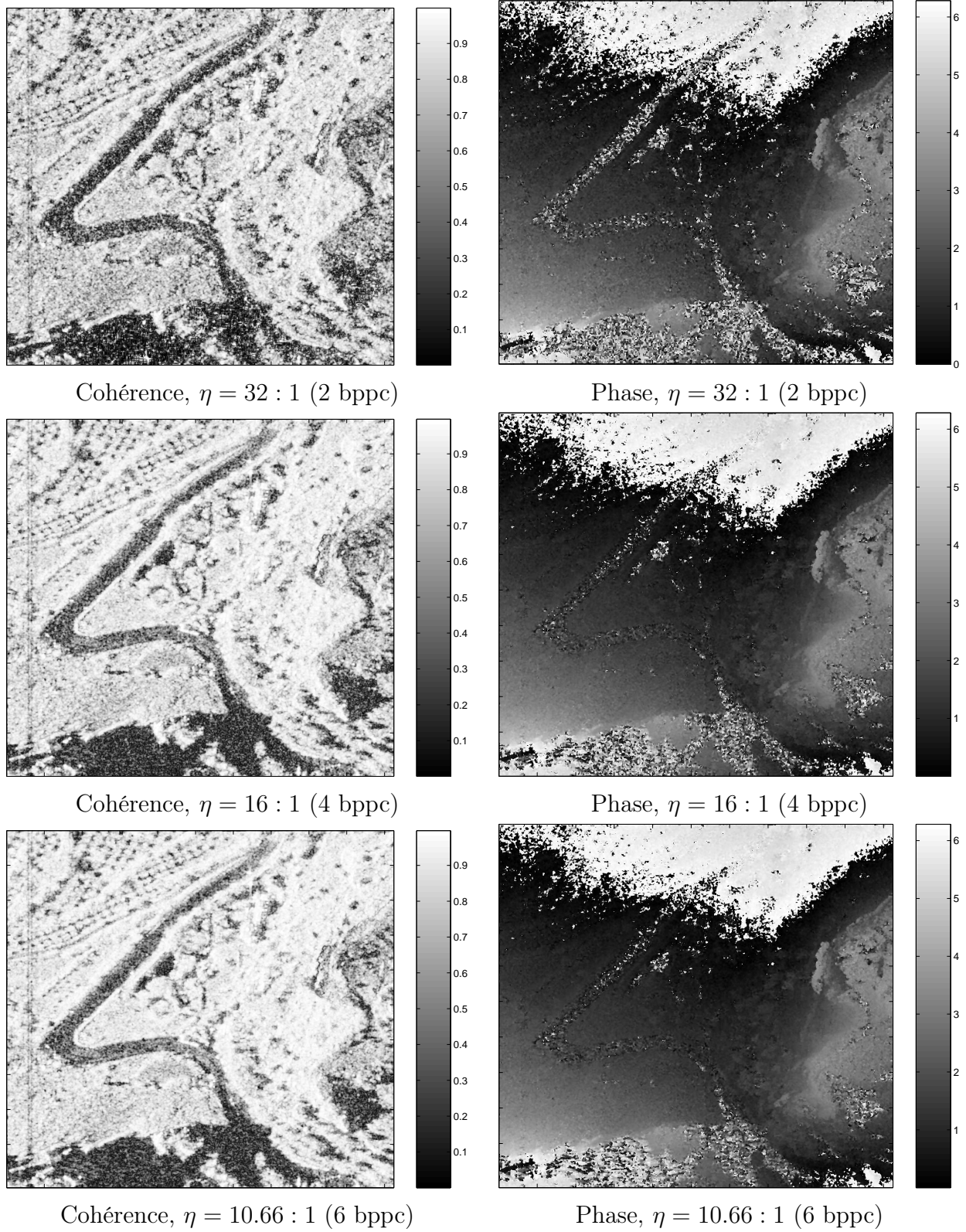


FIG. 6.10 – Algorithme **TCE-IQC** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique.

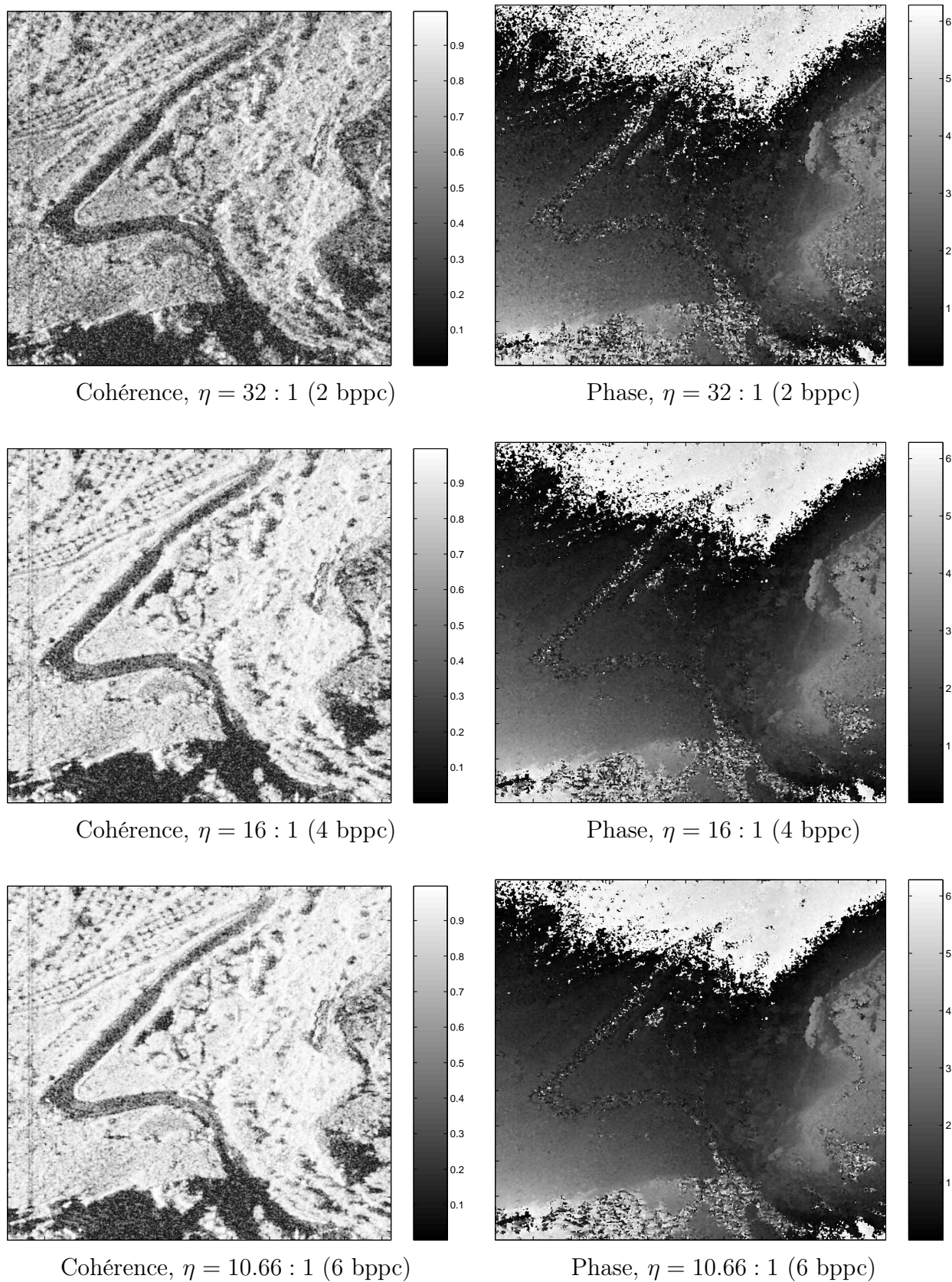


FIG. 6.11 – Algorithme **WECUPQ** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique.

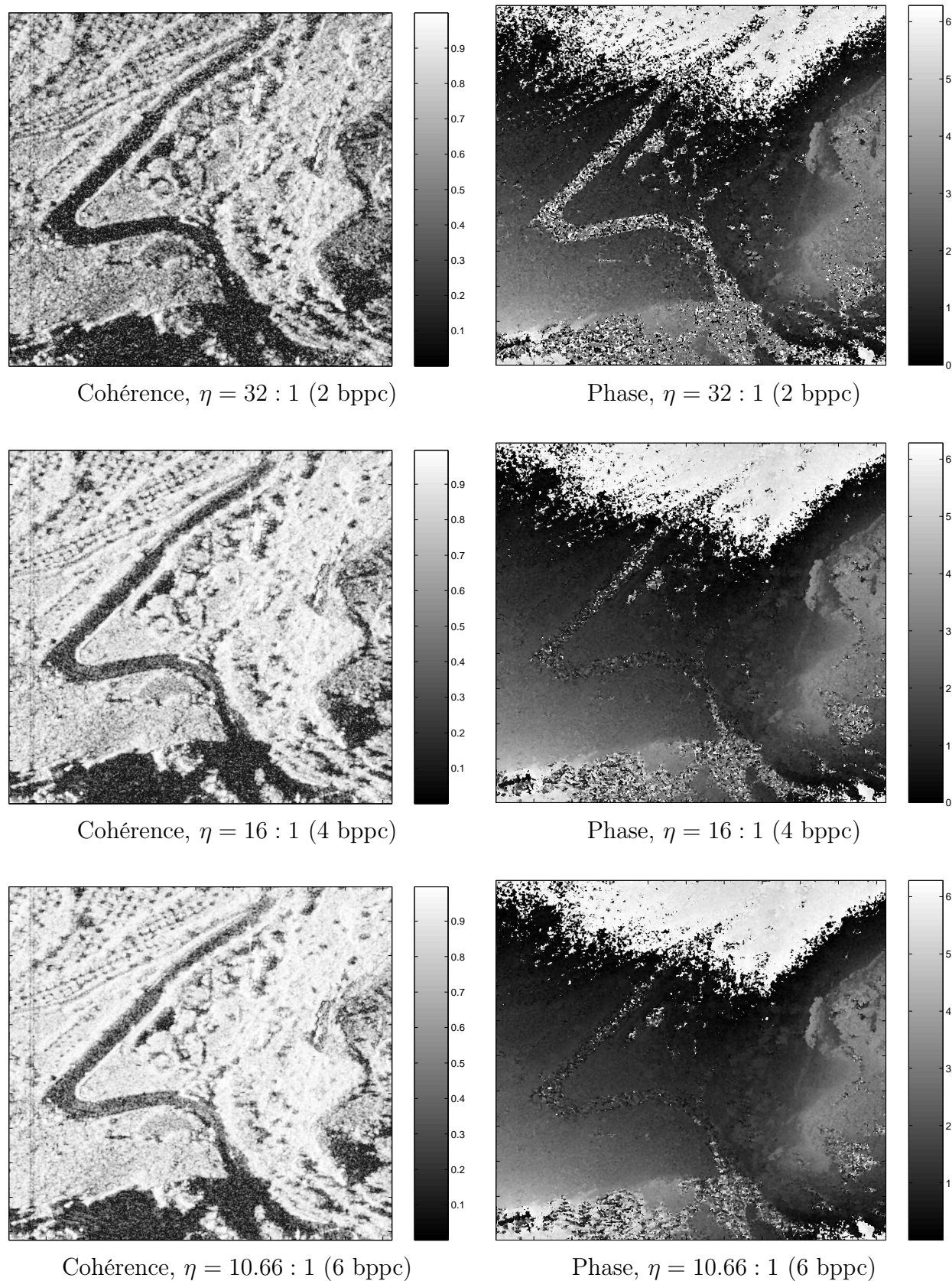


FIG. 6.12 – Algorithme **TCEAC-CSPQ** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique.

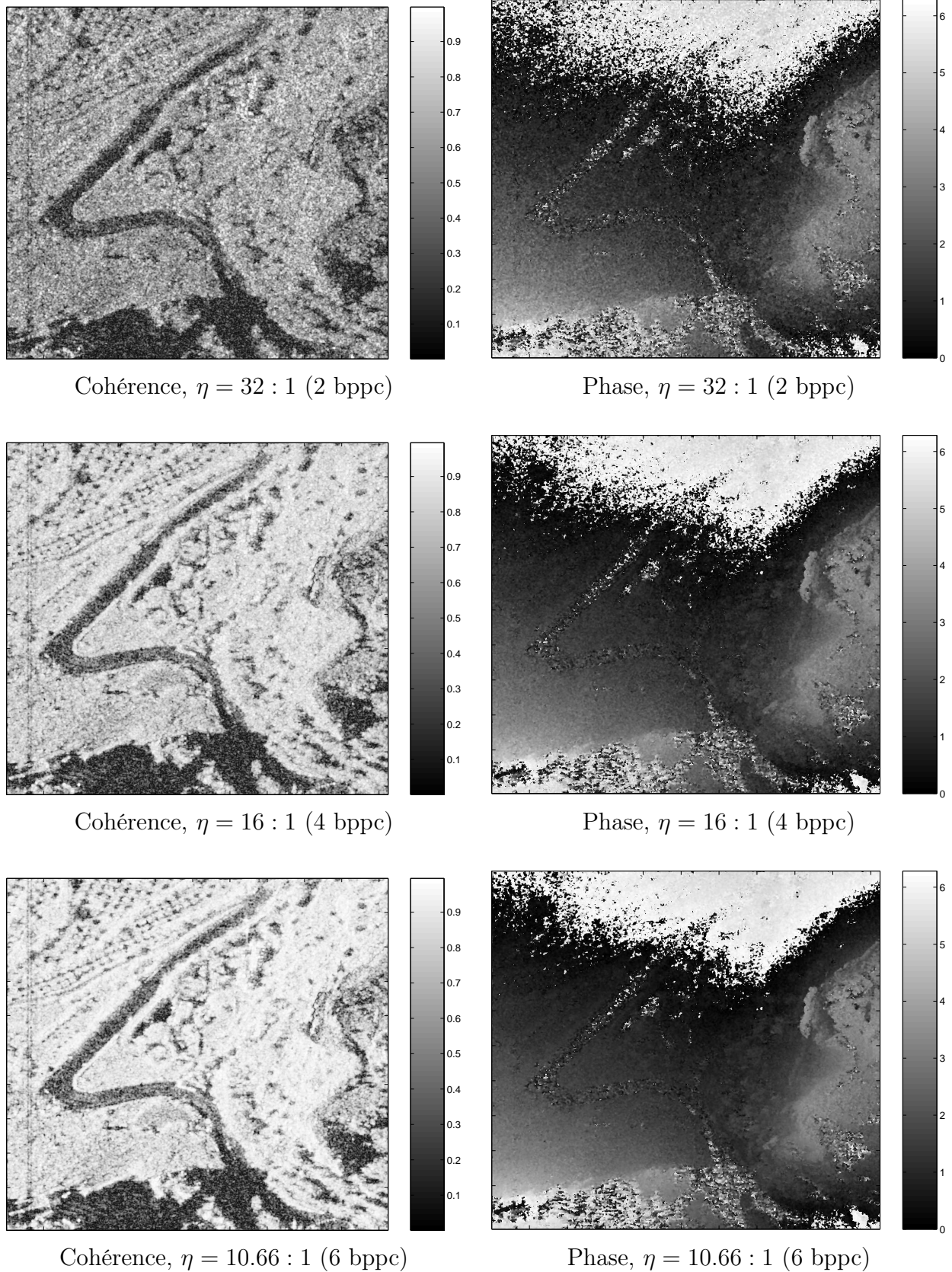


FIG. 6.13 – Algorithme **TCEAC-IPQ** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique.

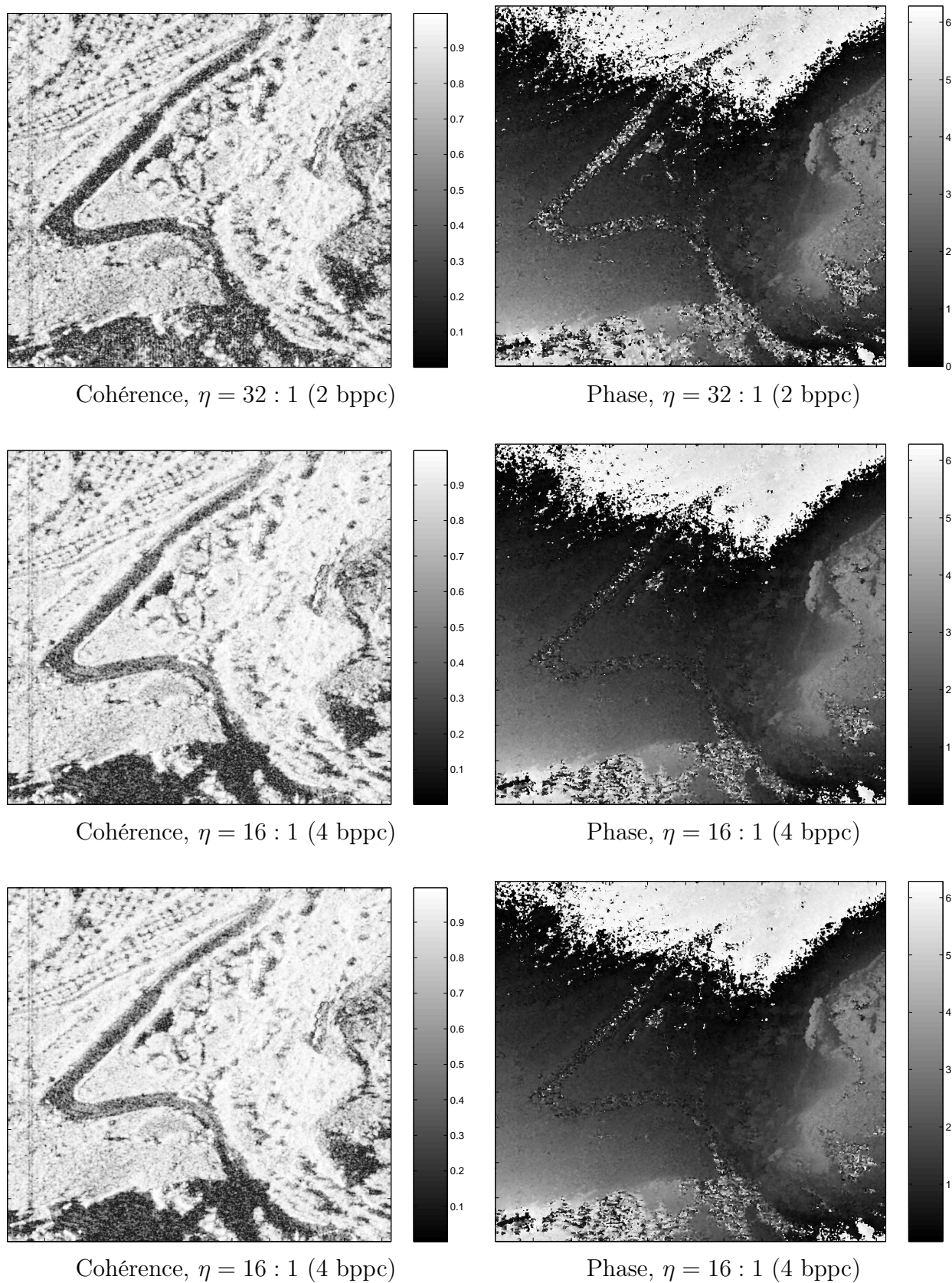


FIG. 6.14 – Algorithme **WPTCE-IQC** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique.

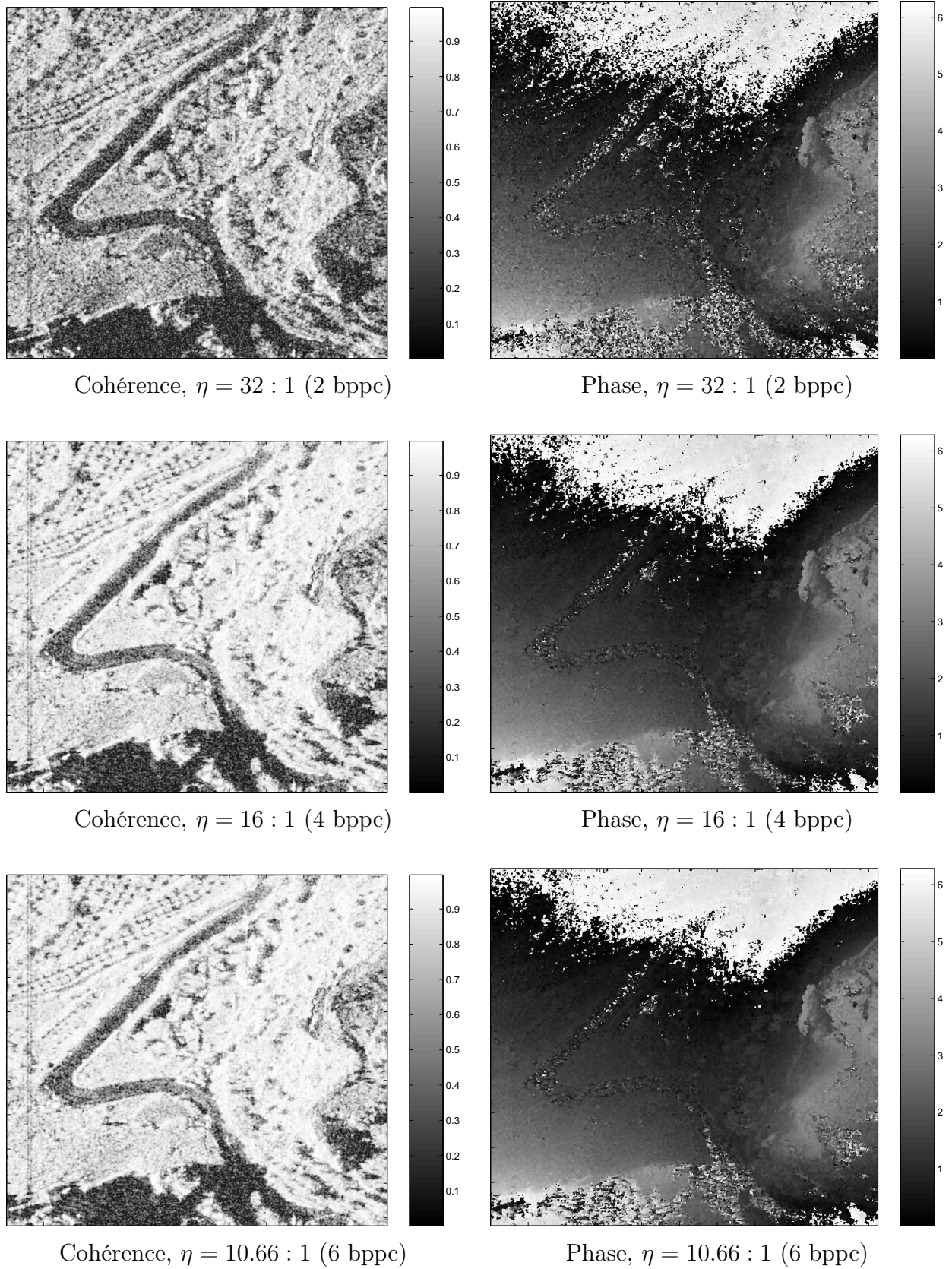


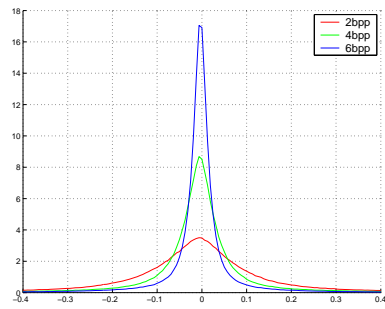
FIG. 6.15 – Algorithme **WP-ECUPQ** : Cohérence et différence de phase de l'interférogramme reconstruit après décompression du couple interférométrique

Visuellement la qualité des interférogrammes est proche pour des débits supérieurs à 4 bits par pixel complexe. L'effet de moyennage dû à la prise en compte de la fenêtre de calcul tend à "lisser" l'interférogramme et donc à réduire la différence de distorsion entre les techniques de compression. Les franges sont correctement restituées et la cohérence fidèle à l'interférogramme original.

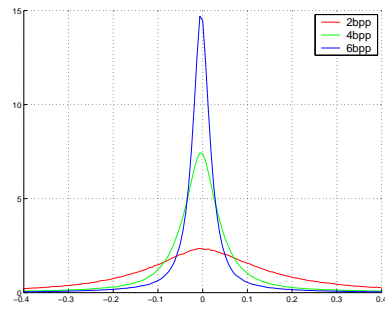
Néanmoins, en considérant un codage faible débit et la méthode de compression TCEAC-IPQ, on remarquera une dégradation de la cohérence répartie sur toute l'image. D'une manière générale, une compression à 2 bits par pixel dégrade la qualité visuelle de l'interférogramme reconstruit. On note ainsi l'apparition d'un "bruit" de phase au niveau des franges et des zones de faible cohérence. La figure 6.16 illustre ainsi l'histogramme de la distorsion entre la phase de l'interférogramme original et décompressé. On remarquera qu'une fois encore, les méthodes WPTCE-IQC et WECUPQ permettent d'obtenir l'erreur de phase la plus concentrée autour de la valeur nulle.

A partir des figures 19 et 20 fournies en Annexes de ce manuscrit, la carte de corrélation complexe de la scène de Baux en Provence est similaire à l'image de cohérence de l'interférogramme original (figure 6.9(a)). Cette observation tend à montrer que les techniques de compression étudiées tendent à privilégier une meilleure qualité de reconstruction sur les zones les plus cohérentes de l'image (excepté la technique TCEAC-IPQ).

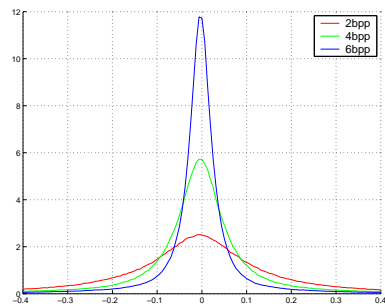
Afin d'établir une mesure de distance de l'interférogramme complexe, l'ensemble des figures 6.17 et 6.18 représentent la carte de cohérence calculée entre l'interférogramme original et reconstruit suivant l'équation (6.17). On prend ainsi en compte la caractérisation locale de la qualité de reconstruction des interférogrammes. Une fois encore, la carte de cohérence montre clairement que les méthodes WPTCE-IQC et WP-ECUPQ permettent de reconstruire avec la meilleure qualité, les interférogrammes issus des images SLC décompressées.



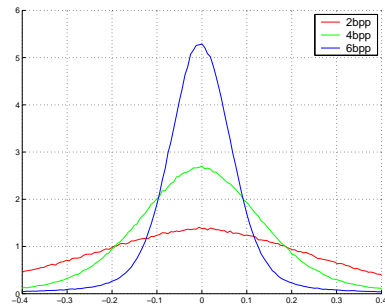
TCE-IQC : Erreur de phase



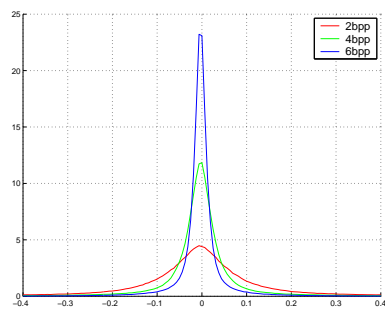
WECUPQ : Erreur de phase



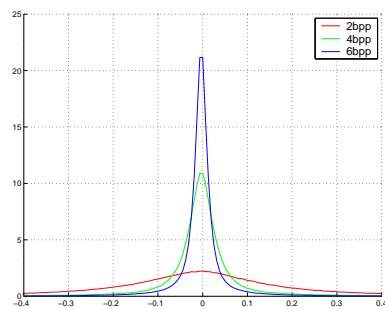
TCEAC-CSPQ : Erreur de phase



TCEAC-IPQ : Erreur de phase



WPTCE-IQC : Erreur de phase



WP-ECUPQ : Erreur de phase

FIG. 6.16 – Histogrammes de l'erreur de phase entre l'interférogramme original et décomposé calculé à partir du couple d'images extraites des Baux de Provence.

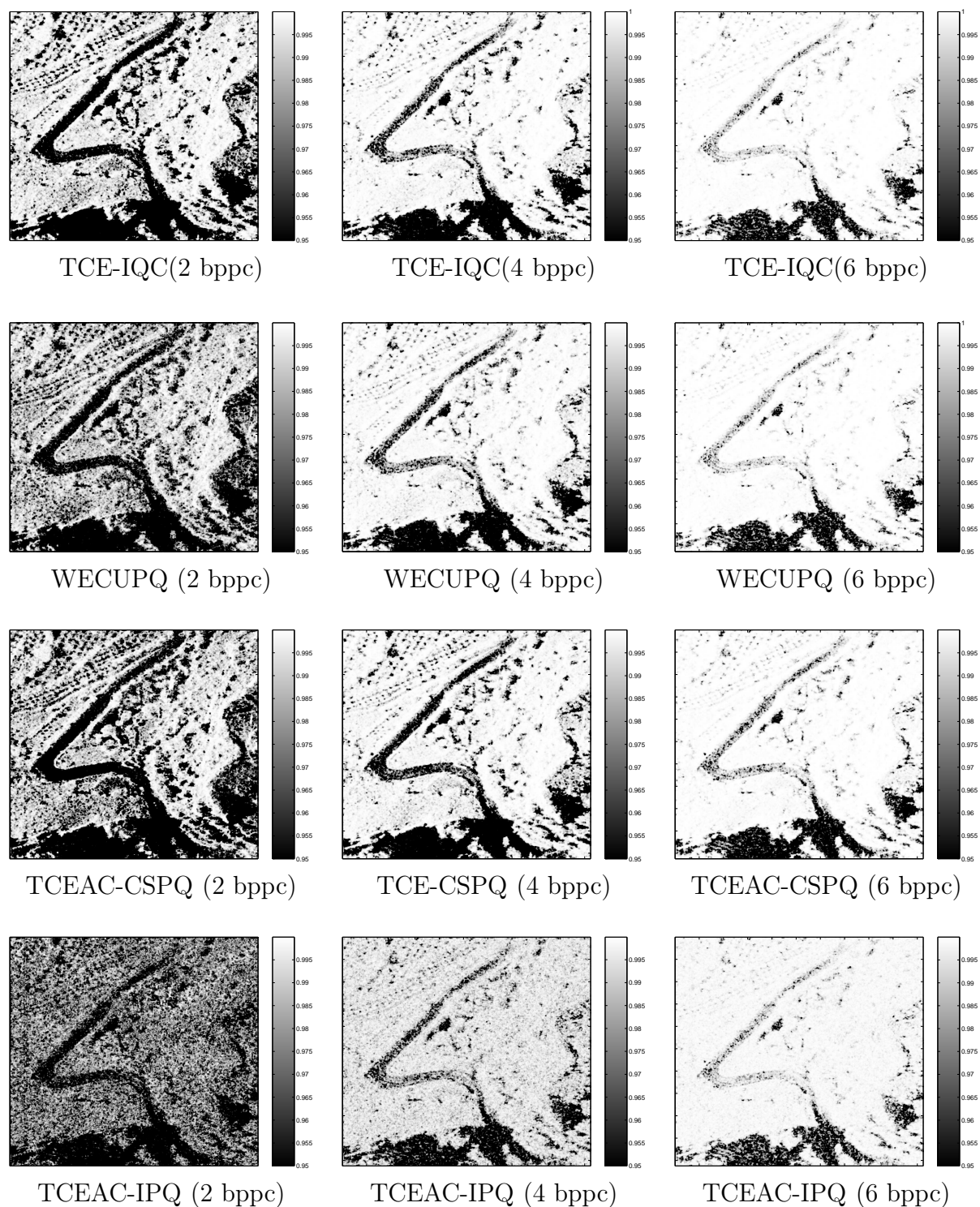


FIG. 6.17 – Cohérence issue de la corrélation complexe entre interférogramme original et reconstruit.

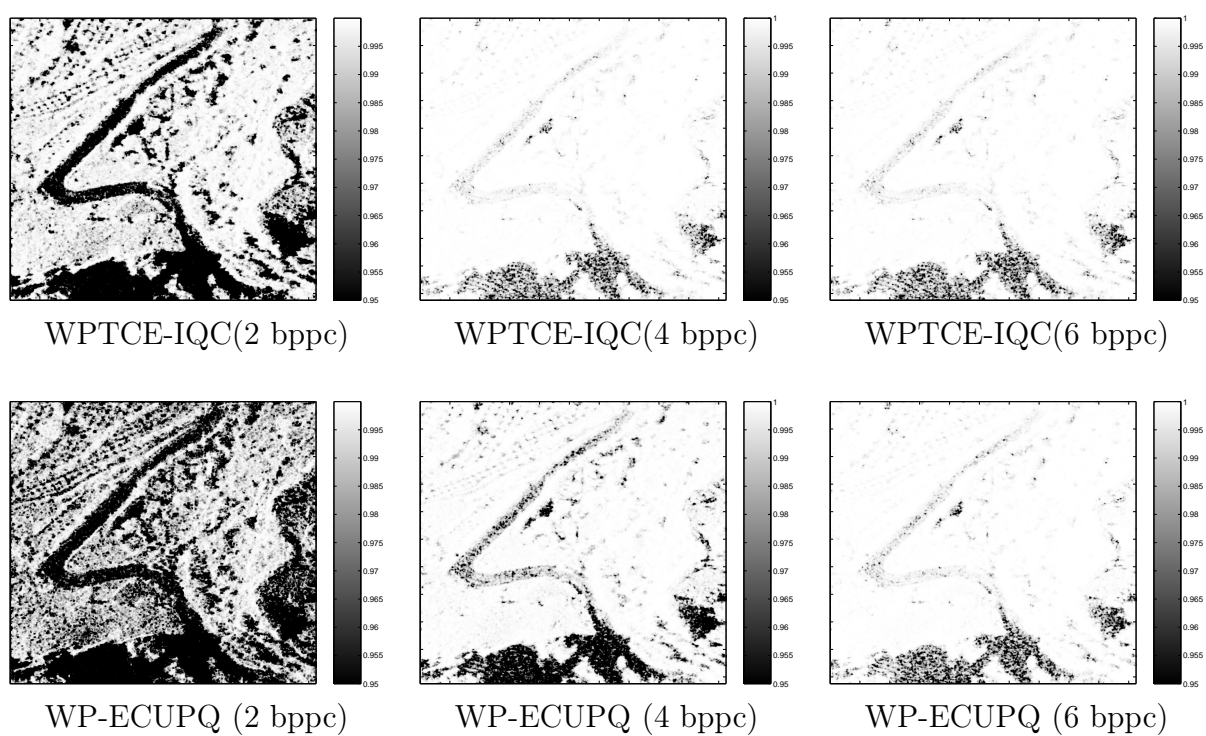


FIG. 6.18 – Cohérence issue de la corrélation complexe entre interférogramme original et reconstruit.

6.4.2 Considérations liées à la complexité

- La technique WECUPQ implémente la quantification polaire des coefficients à partir d'un simple quantificateur scalaire sur l'amplitude et la phase des coefficients d'ondelettes. Mis à part l'estimation des paramètres statistiques, cette technique offre des perspectives intéressantes d'un point de vue complexité de calcul.
- L'algorithme TCE-IQC s'appuie sur une quantification par plan de bits plus sophistiquée. Néanmoins, l'implémentation efficace du filtre Tarp entraîne une charge de calcul qui reste comparable à celle de l'algorithme WECUPQ. On conserve par ailleurs la possibilité d'augmenter progressivement la qualité de reconstruction de l'image.
- Contrairement aux algorithmes WECUPQ et TCE-IQC, l'approche TCEAC-CSPQ ne fait appel qu'à une seule transformée en ondelettes appliquée sur l'image d'amplitude. Moyennant l'estimation des paramètres statistiques associés à l'image d'amplitude, la quantification scalaire de la phase conditionnellement à l'amplitude est évidemment moins coûteuse en temps de calcul. Cette technique offre donc un compromis intéressant entre qualité de reconstruction et complexité de calcul.
- La quantification vectorielle de type GLA associée à la méthode TCEAC-IPQ requiert la construction et le stockage en mémoire du dictionnaire de représentants. La charge de calcul liée à la recherche exhaustive du meilleur représentant est accrue par la taille du dictionnaire et la dimension considérée. Une alternative intéressante consiste à utiliser la quantification vectorielle algébrique qui ne requiert ni recherche exhaustive, ni stockage du dictionnaire. En revanche, l'indexage n'est pas immédiat. Quelques solutions à ce sujet ont été apportées par Conway et Sloane [15], Lamblin [48] ou encore Moureaux et al [54] [64] [32]. Néanmoins, d'un point de vue conceptuel, la quantification de l'image de phase indépendamment du module offre les performances les plus faibles en terme d'erreur quadratique mesurée sur l'image complexe et sur l'ensemble des critères considérés.
- A partir des critères retenus pour cette étude, les méthodes WPTCE-IQC et WPECUPQ offrent indéniablement les meilleures performances en terme de qualité de reconstruction de l'image complexe. En revanche, la recherche de la meilleure base de paquets d'ondelettes réalisée lors de la phase d'encodage entraîne une complexité de mise en oeuvre plus importante.

6.5 Conclusion

Ce chapitre a été consacré à l'évaluation de la qualité associée à la compression des images RSO SLC. L'appréciation de la qualité est une notion complexe à définir et qui, au final dépend essentiellement de l'application visée. A partir d'un ensemble de critères objectifs, nous avons utilisé dans un premier temps un ensemble de mesures classiques associée à l'erreur quadratique moyenne. En complément, la carte de corrélation complexe a permis de fournir une évaluation locale de la qualité de l'image SLC décompressée. Enfin, à partir des images d'un couple interférométrique, nous avons pu caractériser la qualité de reconstruction des données issues de l'interférogramme reconstruit.

Les méthodes WPTCE-IQC et WP-ECUPQ basées sur une représentation en paquets d'ondelettes de la partie réelle et imaginaire de l'image SLC semblent apporter les meilleures performances tous critères confondus. L'hypothèse haut-débit à partir de laquelle nous avons élaboré les algorithmes WECUPQ et WP-ECUPQ semble néanmoins fournir la limitation de cette approche.

Notons qu'à partir d'un codage de 4 bits par pixel complexe, les différences visuelles entre les images ne sont pas très significatives. En revanche, à bas débit, les principales différences se situent au niveau de la restitution du chatoiement qui est plus fortement affecté par le bruit de quantification que les structures géométriques et fortement reflectives de l'image.

L'utilisation des techniques par paquets d'ondelettes entraîne une complexité de mise en oeuvre plus importante qui peut s'avérer néanmoins rédhitoire pour les applications temps-réel embarqué. La technique TCEAC-CSPQ offre en ce sens la plus faible charge de calcul, au prix d'une dégradation plus importante de la qualité l'image SLC décompressée.

Conclusion générale

Le radar à synthèse d'ouverture est un système d'observation à l'origine de nombreuses applications de télédétection qui génère un volume de données très important. La nature complexe des images RSO SLC au sens mathématique du terme et la forte dynamique des valeurs de pixels, accroît de manière significative le débit nécessaire à la transmission des données. La solution retenue consiste alors à comprimer les données moyennant une dégradation irréversible sur celles-ci.

La conservation de l'information de phase en tant que données utiles pour les applications interférométriques et polarimétriques, nous amène à prendre en compte la compression de l'image synthétisée suivant sa représentation complexe. Le chapitre 1 fut consacré à décrire les caractéristiques des images RSO. L'état de l'art des principales méthodes de compression du chapitre 2, nous a conduit à utiliser les techniques basées sur la transformée en ondelettes. Equivalente à une analyse multi-résolution de l'image, elle offre une représentation efficace pour la compression d'images RSO.

A partir d'une transformée en ondelettes appliquée séquentiellement sur la partie réelle et imaginaire de l'image complexe, nous avons ainsi proposé deux approches de quantification :

- TCE-IQC : une quantification progressive de la partie réelle et imaginaire des coefficients d'ondelettes. Nous y avons proposé une adaptation de l'algorithme TCE de C. Tian et S. S. Hemami basé sur classification des coefficients d'ondelettes suivant une approche de codage fractionnaire par plan de bits.

- WECUPQ : une quantification polaire à contrainte entropique des coefficients d'ondelettes complexes, représentés sous leur forme amplitude/phase. Cette méthode s'appuie sur une modélisation de la distribution du module des coefficients par une loi de Fisher. A l'aide du formalisme des statistiques de seconde espèce développé par J. M. Nicolas, nous avons introduit le modèle de loi de Fisher Généralisée qui permet de caractériser un ensemble plus large de distributions d'amplitude. En tenant compte de ce modèle, nous avons mis en oeuvre une procédure d'allocation dynamique de débits entre les sous-bandes de coefficients.

En se basant sur la transformée en paquets d'ondelettes, nous avons ainsi proposé les algorithmes WP-ECUPQ et WPTCE-IQC qui intègrent un algorithme de recherche de

meilleure base suivant la minimisation d'un critère de coût représentatif du meilleur compromis débit-distorsion. Il s'agit alors de trouver les primitives de représentation qui s'adaptent au mieux au contenu spatio-fréquentiel de l'image.

Le second axe de recherche nous a conduit à considérer l'image SLC comme étant constituée d'une composante d'amplitude et de phase. Bien qu'il soit possible de compresser séparément les images d'amplitude et de phase à transmettre, la prise en compte du caractère complexe de l'image nous a amené à développer une quantification de l'image de phase en fonction de l'image d'amplitude. Nous avons ainsi proposé les algorithmes :

- TCEAC-IPQ : compression de l'image d'amplitude par l'algorithme TCE et compression indépendante de l'image de phase par quantification vectorielle réalisée à partir de l'algorithme classique de Lloyd généralisé (GLA).
- TCEAC-CSPQ : compression de l'image d'amplitude par l'algorithme TCE et quantification scalaire de la phase conditionnellement à l'amplitude. Nous avons optimisé ce quantificateur à partir d'une fonction de distorsion à la fois dépendante de la phase et du niveau d'amplitude du pixel associé.

L'évaluation des performances d'un algorithme dépend essentiellement de l'application thématique post-compression. En l'absence de scénarii, nous avons utilisé un ensemble de critères objectifs basés sur les mesures classiques de distorsion. En complément, un critère comme la carte de corrélation complexe nous a semblé bien adapté à la caractérisation locale des erreurs de reconstruction. A ce titre, nous avons évalué la qualité des images décompressées à partir d'une thématique liée à l'interférométrie radar.

Les faibles performances de l'approche TCEAC-IPQ montrent qu'il est sous-optimal de compresser séparément l'image d'amplitude de l'image de phase. On préférera ainsi la méthode TCEAC-CSPQ qui permet d'attribuer une distorsion de phase plus faible aux composantes les plus énergétiques, tout en gardant la possibilité de contrôler séparément les taux de compression associés à chaque composante de l'image SLC.

L'étude comparative menée au chapitre 6 montre clairement une supériorité des algorithmes par paquets d'ondelettes WPTCE-IQC et WP-ECUPQ. Au prix d'une complexité de calcul plus importante, ces techniques offrent une représentation de l'image permettant d'obtenir une quantification plus efficace des composantes hautes-fréquences. Les méthodes WP-ECUPQ et WECUPQ sont cependant limitées par l'hypothèse faite sur la haute résolution du quantificateur. Afin d'améliorer ces méthodes, il serait donc envisageable d'établir un modèle de débit-distorsion basse-résolution qui permettrait de s'affranchir du partitionnement uniforme des quantificateurs polaires.

Dans le cadre de ce manuscrit, nous avons considéré le chatoiement comme une information utile qu'il est souhaitable de conserver. Notons à ce propos que d'autres approches proposées par G. Mercier [63] ou Z. Zeng et al [116] consistent à appliquer une technique de réduction du chatoiement avant compression de l'image. Il existe de multiples techniques de réduction de chatoiement parmi lesquelles les méthodes de seuillage des coefficients d'ondelettes suivant l'approche de Donoho [19]. De manière à ramener le bruit multiplicatif en bruit additif, nous avons mentionné au chapitre 3 l'utilisation d'une transformée en ondelettes homomorphique à partir de laquelle nous avons caractérisé la distribution des coefficients d'ondelettes suivant une loi de Fisher [104]. Il est alors tout à fait possible de construire un filtre MAP (Maximum A Posteriori) permettant de débruiter les coeffi-

cients d'ondelettes suivant un formalisme Bayésien. En lieu et place du modèle a priori de distribution alpha-stables proposé par A. Achim [3], nous pouvons envisager de construire une fonction de seuillage des coefficients d'ondelettes basée sur une distribution de Fisher puis de comprimer l'image à partir des coefficients débruités.

A travers le panel d'algorithmes proposés au cours de cette étude, nous espérons avoir contribué à apporter des réponses spécifiques à la compression d'images RSO SLC. Les algorithmes proposés dans ce manuscrit se focalisent en particulier sur la minimisation du compromis débit-distorsion calculé sur l'ensemble de l'image complexe. Cette remarque ouvre par conséquent une perspective sur l'adéquation entre la compression d'image RSO SLC optimisée suivant ce critère et l'application thématique sous-jacente. Les scénarii imaginables sont évidemment multiples et correspondent à l'exploitation d'une information spécifique de l'image RSO SLC. Il conviendrait donc d'orienter dès le départ la technique de compression de manière à préserver la pertinence de ces informations.

Annexes

.1 Extraction de la courbure convexe

Soit $\mathcal{Z} : \{(R(z), D(z)) \mid z \in \mathbb{N}\}$ un ensemble de points formant une courbe caractéristique de débit-distorsion. L'ensemble \mathcal{Z} étant discret, on peut définir un nombre discret de "pentes" reliant les points $(R(z), D(z))$. Soit $\lambda(z)$ la pente définie par :

$$\begin{aligned}\lambda(0) &\triangleq +\infty \\ \lambda(z) &\triangleq \min_{z' < z} \frac{D(z') - D(z)}{R(z) - R(z')} \quad z > 0\end{aligned}$$

Afin d'extraire l'ensemble des points \mathcal{H} appartenant à la courbure convexe par morceau, D. Taubman [95] montre la condition nécessaire et suffisante d'appartenance $z \in \mathcal{H}$ est donnée par :

$$\lambda(z) > 0 \text{ et } \lambda(z) > \max_{t > z} \frac{D(z) - D(t)}{R(t) - R(z)}$$

Pour plus de précisions, nous renvoyons le lecteur à [97] sur la preuve de ce théorème.

Autrement dit, pour un point $z \in \mathcal{Z}$ donné, celui-ci appartient à l'ensemble \mathcal{H} si toutes les pentes calculées avec les points $t > z$ sont strictement inférieures à $\lambda(z)$. Cette condition nécessaire et suffisante garantit ainsi une décroissance monotone du paramètre $\lambda(z)$.

En référence à [96], nous rappelons ici la procédure d'extraction des points $z \in \mathcal{H}$:

Fixer $\lambda(0) = \infty$, $\mathcal{H} \leftarrow \{0\}$, $h^{last} \leftarrow 0$

Pour $z = 1, 2 \dots Z$

$\Delta D \leftarrow D(h^{last}) - D(z)$ et $\Delta R \leftarrow R(z) - R(h^{last})$

Si $\Delta D > 0$

Tant que $\Delta D \geq \lambda(h^{last}) \Delta R$

$\mathcal{H} \leftarrow \mathcal{H} \setminus \{h^{last}\}$

(exclusion du dernier élément de la liste)

$h^{last} \leftarrow \max\{\mathcal{H}\}$

(extraire le dernier élément de la liste)

$\Delta D \leftarrow D(h^{last}) - D(z)$

$\Delta R \leftarrow R(z) - R(h^{last})$

Fin Tant que

$h^{last} \leftarrow z + 1$

$\mathcal{H} \leftarrow \mathcal{H} \cup \{h^{last}\}$

$\lambda(h^{last}) \leftarrow \frac{\Delta D}{\Delta R}$

Fin Si

Fin Pour

.2 Coefficient de corrélation complexe moyen/coefficient de variation : scène extraite de la ville d'Istres

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.8033	1.1473
SPIHT-IQC	0.7977	1.1478
WECUPQ	0.7844	0.9783
TCEAC-CSPQ	0.7688	0.6876
TCEAC-IPQ	0.7820	0.6876
WPTCE-IQC	0.8669	1.0572
WPECUPQ	0.8177	1.1236

TAB. 7 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 2 bits par pixel complexe ($\gamma_{original} = 1.0245$).

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.9354	1.0534
SPIHT-IQC	0.9298	1.0520
WECUPQ	0.9483	1.0435
TCEAC-CSPQ	0.8969	1.0084
TCEAC-IPQ	0.9262	1.0084
WPTCE-IQC	0.9802	1.0309
WPECUPQ	0.9680	1.0277

TAB. 8 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 4 bits par pixel complexe ($\gamma_{original} = 1.0245$).

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.9817	1.0285
SPIHT-IQC	0.9805	1.0267
WECUPQ	0.9839	1.0302
TCEAC-CSPQ	0.9717	1.0433
TCEAC-IPQ	0.9756	1.0433
WPTCE-IQC	0.9954	1.0241
WP-ECUPQ	0.9936	1.0270

TAB. 9 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 6 bits par pixel complexe ($\gamma_{original} = 1.0245$).

.3 Coefficient de corrélation complexe moyen/coefficient de variation : scène extraite des Baux de Provence

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.7955	1.1501
WECUPQ	0.7909	0.9616
TCEAC-CSPQ	0.7838	0.6963
TCEAC-IPQ	0.7866	0.6963
WPTCE-IQC	0.8672	1.0417
WPECUPQ	0.8262	1.1598

TAB. 10 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 2 bits par pixel complexe ($\gamma_{original} = 1.0364$).

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.9379	1.0813
WECUPQ	0.9475	1.0744
TCEAC-CSPQ	0.9068	1.0226
TCEAC-IPQ	0.9237	1.0226
WPTCE-IQC	0.9790	1.0493
WPECUPQ	0.9749	1.0378

TAB. 11 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 4 bits par pixel complexe. ($\gamma_{original} = 1.0364$)

Codec	$\rho \in [0 : 1]$	$\gamma = \sigma/\mu$
TCE-IQC	0.9823	1.0397
WECUPQ	0.9842	1.0565
TCEAC-CSPQ	0.9692	1.0606
TCEAC-IPQ	0.9744	1.0606
WPTCE-IQC	0.9949	1.0404
WP-ECUPQ	0.9934	1.0393

TAB. 12 – Performances des algorithmes de compression obtenus concernant la mesure du coefficient de corrélation complexe moyen ρ et du coefficient de variation γ à 6 bits par pixel complexe. ($\gamma_{original} = 1.0364$)

.4 Carte de corrélation complexe : scène extraite des Baux de Provence

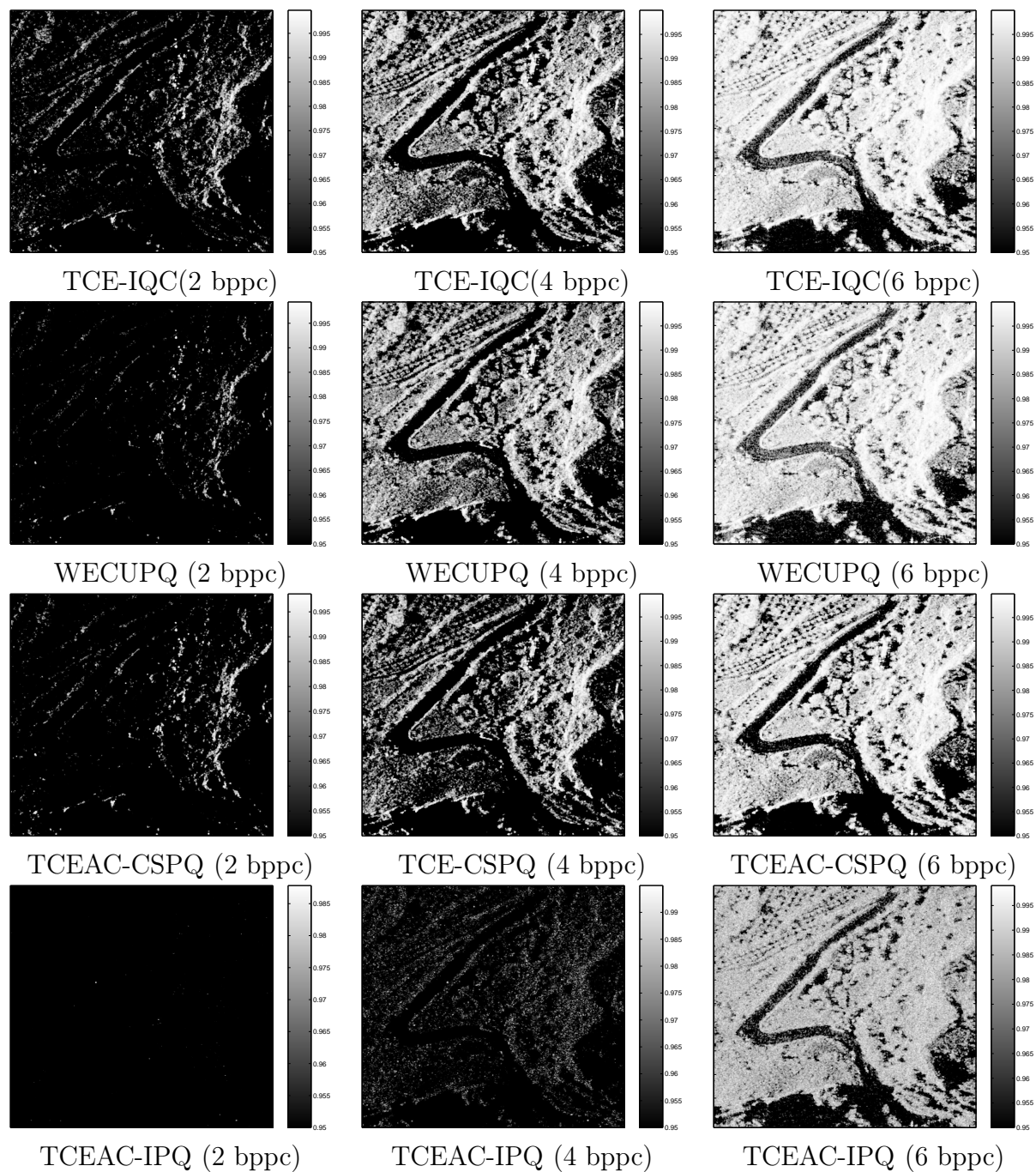


FIG. 19 – Carte de corrélation complexe calculée entre l'image SLC originale et l'image décompressée.

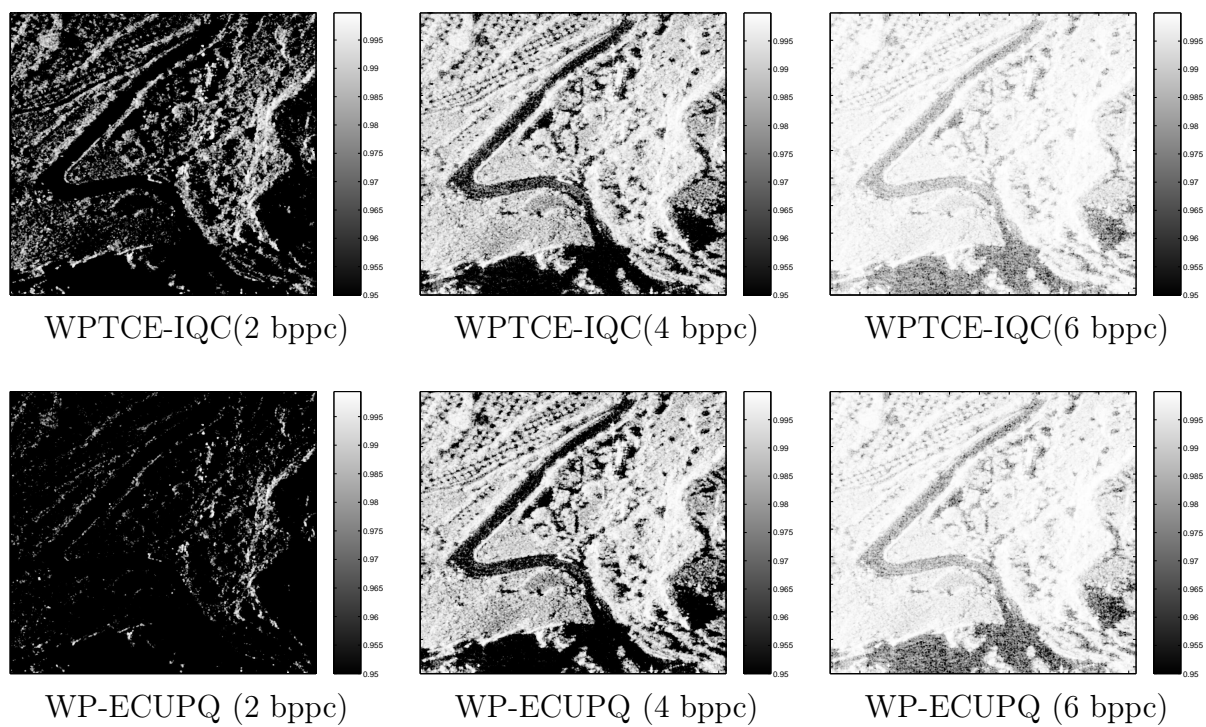


FIG. 20 – Carte de corrélation complexe calculée entre l'image SLC originale et l'image décompressée.

Bibliographie

- [1] H. Laur et E. Nezry A. Lopès. Statistical distribution and texture in multilook and complex sar images. IGARSS'90, pages 20–24, 1990.
- [2] J. H Conway and N. J. A. Sloane. Sphere Packings, Lattice and Groups. New York : Springer Verlag, 1988.
- [3] A. Achim, P. Tsakalides, and A. Bezerianos. SAR image denoising via bayesian wavelet shrinkage based on heavy tailed modelling. IEEE Trans on geosciences and remote sensing, 41(8), August 2003.
- [4] Y. S. Ho and A. Gersho. Variable rate multi-stage vector quantization for image coding. In Proceedings ICASSP, pages 1156–1159, April 1988.
- [5] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. IEEE Trans. Image Processing, 1,2 :713–718, April 1992.
- [6] A. M. Awan, N. M. Rajpoot, and S. A. Husain. Stack-run adaptive wavelet image compression.
- [7] M. Barlaud, Patrick Solé, Thierry Gaidon, Marc Antonini, and Pierre Mathieu. Pyramidal Lattice Vector Quantization for Multiscale Image Coding. IEEE Trans on Image Processing, 3(4) :367–380, jul.
- [8] Blain and E. Fisher. A comparison of vector quantization techniques in transform and subband coding of imagery. IEEE Trans on Signal Processing, 3, February 1995.
- [9] J.B. Buckheit and Donoho D.L. Wavelab and reproducible research. Rapport technique, Department of Statistics, Stanford University, 1995.
- [10] C. Parisot and M. Antonini and M. Barlaud. Optimal nearly uniform scalar quantizer design for wavelet coding. Proceedings of SPIE Visual Communications and Image Processing Conference, January 2002.
- [11] P. A. Chou, T. Lookabaugh, and R. M. Gray. Entropy constrained vector quantization. IEEE Trans Acous Speech Signal Process, 37 :31–42, January 1989.
- [12] A. Cohen, I. Daubechies, and J.C Feauveau. Biorthogonal bases of compactly supported wavelets. Comm Pure and Appl Math, 45(5) :458–560, 1992.
- [13] R.R Coifman and M.V Wickerhauser. Entropy-based algorithms for best basis selection. IEEE Trans Inform Theory, 38 :713–718, March 1992.
- [14] J. H Conway and N. J. A. Sloane. Fast quantizing and decoding algorithms for lattice quantizers. IEEE Trans Inform Theory, 28(2) :227–232, March 1982.
- [15] J. H Conway and N. J. A. Sloane. A fast encoding method for lattices codes and quantizers. IEEE Trans Inform Theory, 29(6) :820–824, March 1983.

- [16] A. Croisier, D. Esteban, and C. Galand. Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques. *Proc Int Conf Acoust Speech and Sig Proc*, pages 443–446, aug 1976.
- [17] M. Datcu, G. Schwarz, K. Schmidt, and C. Reck. Quality evaluation of compressed optical and sar images : Jpeg vs wavelets. *International Geoscience and remote Sensing Symposium (IGARSS)*, 3 :1687–1689, July 1995.
- [18] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm Pure and Appl Math*, 41 :909–996, nov 1998.
- [19] D. L. Donoho. Denoising by soft-thresholding. *IEEE Trans. on Inform Theory*, 41 :613–627, may 1995.
- [20] B.A Eckstein and R. Peter. Assessing the performance effects of data compression for sar imagery. *Applied Imagery Pattern Recognition Workshop*, pages 102–108, 2000.
- [21] P. Eichel and Robert W. Ives. Compression of complex-valued SAR images. *IEEE Trans on Image Proc*, 8(10) :84–85, October 1999.
- [22] E.Ordentlich, M. Weinberger, and G.Seroussi. A low complexity modeling approach for embedded coding of wavelet coefficients. *IEEE Data Compression Conference*, March 1998.
- [23] J. Nicolas et A. Maruani. Modélisation du chatoiement des images de radar à synthèse d’ouverture. *Rapport technique ENST99D007*, 2002.
- [24] C. Jakowatz et al. *Spotlight-mode synthetic aperture radar : A signal processing approach*. Kluwer Academic Publishers, 1996.
- [25] F Meyer et all. Fast adaptive wavelet packet image compression. *IEEE Trans on Image Processing*, 9 :792–800, May 2000.
- [26] J. Nicolas et F. Tupin. Gamma mixture modeled with "second kind statistics" : Application to SAR image processing. In *IEEE*, éditeur, *IGARSS’02*, pages 2489–2491, 2002.
- [27] E.Jakeman et N.Pusey. A model for non rayleigh sea echos. *IEEE Transaction on Ant. and Prop.*, 24(6) :806–814, 1976.
- [28] R. O. Duda et P. E. Hart. *Pattern classification and scene analysis*. Wiley and Sons, 5 :520–533, 1973.
- [29] Gradshteyn et Ryzhik. *Table of integrals, series and products*. A. Jeffrey and D. Zwillinger, sixth edition, 2000.
- [30] P.F Panter et W. Dite. Quantization distortion in pulse count modulation with nonuniform spacing of levels. *Proc IRE*, pages 44–48, January 1951.
- [31] S. Emek F.A Sakarya. Sar image compression. Dans *Conference Record of the Thirtieth Asilomar Conference*, pages 858–862, November 1996.
- [32] T.R Fischer. A pyramid vector quantizer. *IEEE transactions on information theory*, 32(4) :568–583, 1986.
- [33] T. R. Fisher, M. W. Marcellin, and M. Wang. Trellis coded vector quantization. *IEEE Trans Inform Theory*, 37 :1551–1566, November 1991.
- [34] J. E. Fowler. Qccpack : An open-source software library for quantization, compression and coding. In *App. Digital Image Proc. XXIII*, A. G Tescher, Ed, San Diego, *Proc SPIE 4115*, 40 :294–301, August 2000.

- [35] A. Gersho. Assymptotically optimal block quantization. *IEEE Trans on Inform Theory*, 25(4), July 1979.
- [36] H. gish et J.N Pierce. Assymptoticaly efficient quantizing. *IEEE Trans. Inf. Theory*, 14 :676–683, September 1968.
- [37] D. Gleich, P. Planinsic, B. Gergic, and Z. Cucez. Progressive space frequency quantization for SAR data compression. *IEEE Trans on Geosciences and Remote Sensing*, 40 :3–10, January 2002.
- [38] D. Gleish, P. Planinsic, B. Gergic, and Z. Cucej. Progressive space frequency quantization for SAR data compression. *IEEE Trans on Geoscience and Remote Sensing*, 10 :3–10, January 2002.
- [39] J.W Goodman. Some fundamental properties of speckle. *J. Opt. Soc. Am*, 66 :1145–1150, 1976.
- [40] L. Guillemot, Y.Gandeaou, S. Moussaoui, and J.M Moureaux. An analytical gamma mixture based rate-distortion model for lattice vector quantization. *Eusipco*, september 2006.
- [41] Y. Delignon et A. Marzouki H. Quelle. Unsupervised bayesian segmentation of sar images using the Pearson system distributions. *IGARSS'93*, 3 :1538–1540, 1993.
- [42] I.Daubechies and W.Sweldens. Factoring wavelet transforms into lifting steps, November 1997.
- [43] R. W. Ives, P. Eichel, and N. Magrota. Wavelet compression of complex SAR imagery using complex and real valued wavelets : a comparative study. *Conference Record of the Third-Second Asilomar Conference*, 2 :1294–1298, November 1998.
- [44] Robert W. Ives, P. Eichel, and N. Magrota. A new SAR image compression quality metric. *Circuits and Systems, 42nd Midwest Symposium*, 2 :1143–1146, 1999.
- [45] B. H. Juang and A. H. Gray. Multiple stage vector quantization for speech coding. In *Proceedings ICASSP*, 1 :597–600, April 1982.
- [46] G.G Kuperman and T.D Penrod. Evaluation of compressed synthetic aperture radar imagery. *Aerospace and Electronics Conference*, pages 319–326, May 1994.
- [47] R. Kwok. Block adaptive quantization of magellan sar data. *IEEE Trans. on Gesc. and Rem. Sens*, 27(4) :375–383, July 1989.
- [48] C. Lamblin. Quantification Vectorielle Algébrique Sphérique par le réseau de barnes-wall : application au codage de la parole. *Thèse de Doctorat*, Sherbrooke, Canada, March 1988.
- [49] W. Lawton. Application of complex valued wavelets transform to subband decomposition. *IEEE Trans. on Signal Processing*, 41(12), December 1993.
- [50] D. Lebedeff. Etude de la quantification vectorielle des données brutes issues d'un radar à synthèse d'ouverture. *These de Doctorat Unversite Nice-Sophia Antipolis*, December 1995.
- [51] J. Lim. *Two-Dimentionnal Signal and Image Processing*. Prentice Hall, 1990.
- [52] J. M Lina and M. Mayraud. Complex daubechies wavelets. *Applied and Computational Harmonic Analysis*, 2(3) :219–229, April 1995.
- [53] A. Y. Linde and R.M Gray. An algorithm for vector quantizer design. *IEEE Trans on communications*, 28(1) :84–85, January.

- [54] P. Loyer, J.M. Moureaux, and M. Antonini. Lattice codebook enumeration for generalized gaussian source. *IEEE Transactions on Information Theory*, 49(2) :521–528, février 2003.
- [55] J. MacQueen. in proceedings of the fifth berkeley symposium on math stat and prob. pages 281–296.
- [56] S. Mallat. *A Wavelet Tour of Signal Processing*. New York : Academic, 1998.
- [57] S. G. Mallat. A theory for multiresolution signal decomposition : The wavelet representation. *IEEE Trans. PAMI*, 11(7) :674–693, 1989.
- [58] M. W. Marcellin and T. R. Fisher. Trellis coded quantization of memoryless and gauss-markov sources. *IEEE Trans Inform Theory*, 38 :82–93, November 1990.
- [59] P. Mathieu, M. Barlaud, and M. Antonini. Compression d’Image par transformée en ondelette et quantification vectorielle. *Traitement du Signal*, 7 :101–115, juin 1990.
- [60] J. B. McDonald. Some generalized functions for the size distribution of income. *Econometrica*, 52(7) :647–667, mai 1984.
- [61] G. Mercier. Etude de modèles de compression de données de télédétection multi-source dans le cadre d’une utilisation opérationnelle. Thèse de Doctorat Université de Rennes I, 1999.
- [62] G. Mercier. Etude de modèles de compression de données de télédétection multi-source dans le cadre d’une utilisation opérationnelle. Thèse de Doctorat Université de Rennes I, pages 41–42, 1999.
- [63] Gregoire Mercier. Despeckle-based sar image compression.
- [64] J.M. Moureaux, P. Loyer, and M. Antonini. Efficient indexing method for lattice quantization applications. Dans *IEEE International Conference on Image Processing (ICIP)*, pages 447–450, 16-19 septembre 1996.
- [65] J. Mvogo. Compression des images de radar à synthèse d’ouverture dans le cadre de leur utilisation dans les systèmes d’information géographique. Thèse de Doctorat Université de Marne la Vallée, 2001.
- [66] J. Mvogo. Compression des images de radar à synthèse d’ouverture dans le cadre de leur utilisation dans les systèmes d’information géographique. Thèse de Doctorat Université de Marne la Vallée, pages 128–131, 2001.
- [67] J. Mvogo, VP. Onana, and E. Tonyé. Coastline detection in SAR images using wavelets packets, multiscale segmentation and Markov Random field regularization. *Proceedings of the SPIE*, 4173 :122–129, 2000.
- [68] D. L. Neuhoff. Polar quantization revisited. *Proc. IEEE Int. Symp. Info. Theory*, 11 :60, March 1997.
- [69] J.M Nicolas. Introduction aux statistiques de deuxième espèce : application aux loi d’images RSO. Rapport technique ENST2002D001, 2002.
- [70] J. E. Odegard, H. Guo, M. Lang, C. S. Burrus, R. O. Wells, Jr., L. M. Novak, and M. Hiatt. Wavelet based SAR speckle reduction and image compression. Dans *SPIE Proc. on Algorithms for Synthetic Aperture Radar Imagery II at AeroSense ’95*, volume 2487, pages 259–271, Orlando, FL, 1995.
- [71] C. Olivier. A model for non-rayleigh scattering statistics. *Optica Acta*, 31(6) :701–722, 1976.

- [72] C. Olivier. Optimum texture estimators for sar clutter. *J. Phys D. : Appl. Phys.*, 26 :1824–1835, 1993.
- [73] James W. Owens, Michael W. Marcellin, B. R. Hunt, and Marvin Kleine. Compression of synthetic aperture radar video phase history data using trellis coded quantization techniques, 1998.
- [74] W. A. Pearlman. Polar quantization of complex gaussian random variable. *IEEE Trans Communications*, 27 :892–899, June 1979.
- [75] W. Pennebacker and J.L Michell. *JPEG Still Image Compression Standard*. Van Nostrand Rheinhold, 1993.
- [76] Poularikas. *Handbook of formulas and tables for signal processing*. CRC Press, 1999.
- [77] K. Ramchandran and M. Vetterli. Best wavelet packet bases in a rate-distortion sense. *IEEE Trans on Image Processing*, 9 :160–175, April 1993.
- [78] K. Ramchandran, M. Vetterli, and C.Herley. Wavelets, subband coding, and best bases. *Proceedings of the IEEE*, 84(4) :541–560, April 1996.
- [79] K. Ramchandran, Z. Xiong, K. Asai, and M. Vetterli. Adaptive transforms for image coding using spatially varying wavelet packets. *IEEE Trans on Image processing*, 5(6) :1197–1204, July 1996.
- [80] E. A. Riskin and R.M Gray. A greedy tree growing algorithm for the design of variable rate vector quantizers. *IEEE Trans on Signal Proc*, 39 :2500–2507, November 1991.
- [81] A. Said and W.A Pearlman. A new fast and efficient image coder based on set partitioning in hierarchical trees. *IEEE Trans. Circuits and Systems for Video Technology*, pages 243–250, June 1996.
- [82] J.M Shapiro. An embedded wavelet hierarchical image coder. Dans *Proc IEEE ICASSP*, pages 657–660, San Fransisco, March 1992.
- [83] Y. Shoham and A. Gersho. Efficient bit allocation for an arbitrary set of quantizers. *IEEE Transactions on Acoustics,Speech and Signal Processing*, 36 :1445–1453, September 1988.
- [84] P. Simard, D. Steikrauss, and H. Malvar. On line adaption in image coding with a 2-d tarp filter. *Proc. IEEE Data Comp. Conf.*, pages 23–32, April 2002.
- [85] H. Maitre (sous la direction de). *Traitement des images RSO*. Hermès Science, 2001.
- [86] G.C Staples, S. Rossignol, and W. Stevens. Data compression effects on SAR image interpretation. *International Geoscience and remote Sensing Symposium (IGARSS)*, 3 :1678–1680, July 1995.
- [87] G. J. Sullivan. Efficient scalar quantization of exponential and laplacian random variables. *IEEE Trans on Information Theory*, 42 :1365–1374, 1996.
- [88] P. F. Swaszek and T. W. Wu. Asymptotic performance of unrestricted polar quantizers. *IEEE Trans Inf Theory*, 11 :330–333, March 1986.
- [89] W. Sweldens. The lifting scheme : A new philosophy in biorthogonal wavelets constructions. *SPIE*, pages 68–79, 1995.
- [90] W. Sweldens. The lifting scheme : a custom-design construction of biorthogonal wavelets. *Appl Computer Harmonic*, 3(2) :186–200, 1996.

- [91] D. Taubman and A. Zakhor. Multirate 3-d subband coding of video. *IEEE Trans on Image Processing*, 3, sep 1994.
- [92] David Taubman. High performance scalable image compression with ebcot. *IEEE Trans on Image Processing*, 9 :1138–1170, July 2000.
- [93] David S. Taubman and Michael W. Marcellin. *JPEG2000 , Image compression, fundamentals, standards and practice*. Kluwer Academic Publishers, 2002.
- [94] David S. Taubman and Michael W. Marcellin. *Jpeg2000 , image compression, fundamentals, standards and practice*. Kluwer Academic Publishers, pages 192–193, 2002.
- [95] David S. Taubman and Michael W. Marcellin. *Jpeg2000 , image compression, fundamentals, standards and practice*. Kluwer Academic Publishers, pages 327–348, 2002.
- [96] David S. Taubman and Michael W. Marcellin. *Jpeg2000 , image compression, fundamentals, standards and practice*. Kluwer Academic Publishers, page 346, 2002.
- [97] David S. Taubman and Michael W. Marcellin. *Jpeg2000 , image compression, fundamentals, standards and practice*. Kluwer Academic Publishers, page 342, 2002.
- [98] Chao Tian and Sheila S. Hemami. An embedded image coding system based on tarp filter with classification. *ICASSP ,Montreal*, 2004.
- [99] C. Tison. Interférométrie RSO à haute résolution en milieu urbain : application au calcul de mns urbain. *Thèse de Doctorat, ENST Paris*, 2004.
- [100] E. Trouvé. Imagerie interférentielle en radar à ouverture synthétique. *Thèse de Doctorat, ENST 96E014*, 1996.
- [101] G. Ungerboeck. Trellis-coded modulation with redundant signal sets, parts i and ii. *IEEE Communications Magazine*, 25 :5–21, February 1987.
- [102] B. Usevitch. Optimal bit allocation for biorthogonal wavelet coding. *IEEE Data Compression Conference*, 1996.
- [103] R. Vafin and W. Bastiaan. Entropy-constrained polar quantization and its application to audio coding. *IEEE Trans on Speech and Audio Processing*, 13(2) :221–232, March 2005.
- [104] C. Valade and J.M Nicolas. Homomorphic Wavelet Transform and new subband statistics models for SAR image compression. *IGARSS 2004, Anchorage*, 2004.
- [105] C. Valade, J.M Nicolas, and J. Blanc Talon. Best Wavelet Packets Basis Search using Tarp filtering and classification. *soumis à IEEE Trans on Image Proc*, 2006.
- [106] M. Venkatraman, H. Kwon, and N.M Nasrabadi. Object based SAR image compression using vector quantization. *Aerospace and Electronics Conference*, 36 :1036–1046, October 2000.
- [107] Y. Wang, J T.Rucker, and J. E. Fowler. Three-dimensionnal tarp coding for the compression of hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 2 :136–140, April 2004.
- [108] S. G. Wilson. Magnitude/phase quantization of independant gaussian variates. *IEEE Trans Communications*, 11 :1924–1929, November 1980.
- [109] I H. Witten, R.N Neal, and J.G Cleary. Arithmetic coding for data compression. *Communication of the ACM*, 30 :520–540, June 1987.

- [110] X. Wu. High order modelling and embedded conditional entropy coding of wavelet coefficients for image compression. 31st Asilomar Conf Signals, Systems, Computers, pages 1378–1382, November 1997.
- [111] A. Wyner and J. Ziv. The sliding-window lempel-ziv algorithm is asymptotically optimal. Proceedings of the IEEE, 82(6) :872–877, 1994.
- [112] Z. Xiong, K. Ramchandran, and M. T.Orchard. Wavelet packet image coding using space frequency quantization. IEEE Trans on Acoustics, speech and Signal processing, 7(6) :892–898, June 1998.
- [113] Z. Xiong, K. Ramchandran, M. Vetterli, and K. Asai. Wavelets packets-based image coding using joint space-frequency quantization. In ICIP, pages 324–328, July 1994.
- [114] M. Desai X.Zhang and Y. Peng. Orthogonal complex filter bank and wavelets : Some properties and design. to appear in IEEE Trans on Signal Processing, 1998.
- [115] R. Garello et A. Hillion Y. Delignon. Statistical modelling of ocean sar images. IEEE Proc. Radar Sonar and Navigation, 114(6) :348–354, 1997.
- [116] Z. Zeng and I. G. Cumming. Sar image data compression using a tree-structured wavelet transform. IEEE Trans on Geosciences and Remote Sensing, 40 :3–10, March 2002.
- [117] J. Ziv and A. Lempel. A universal algorithm for sequential data compression. IEEE Trans on Information Theory, 23(3) :337–343, May 1977.
- [118] Z.Xiong, K. Ramchandran, and M.T Orchard. Space frequency quantization for wavelet image coding. IEEE Trans on Image Processing, 6 :677–693, 1997.