

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITE MOULOU MAMMARI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE  
DEPARTEMENT D'ELECTRONIQUE

# Mémoire de fin d'études

Présenté en vue de l'obtention  
du diplôme de Master en Électronique

*Option : Réseaux et Télécommunication*

***Thème:***

**L'apport des bandelettes par rapport aux  
ondelettes dans les applications de traitement  
d'image**

**Proposé et dirigé par:**  
Mme : HEMDANI.N

**Etudié et réalisé par:**  
M<sup>elle</sup>: SAAD nora

Année universitaire 2011/2012

# Remerciements

Je souhaite exprimer mes remerciements les plus sincères pour toutes les personnes qui m'ont apportée leur aide et grâce auxquelles j'ai pu élaborer et mener à terme ce mémoire.

Avant tout, je remercie le Bon Dieu de m'avoir donné le courage et la force nécessaire pour terminer ce travail.

Je tiens ensuite à remercier ma promotrice Madame HEMDANI naima née SADOUN, enseignante au département d'électronique à l'université Mouloud Mammeri de Tizi ousou , de m'avoir proposé un thème aussi intéressant qui m'a permis d'acquérir énormément de connaissances dans le domaine du traitement d'image. Je tiens aussi à la remercier pour l'intérêt qu'elle m'a accordé.

Je remercie également le président et les membres de jury qui me font l'honneur de juger mon travail.

Mes remerciements pour tous ceux qui ont contribué, de près ou de loin, à l'élaboration de ce travail.

## *Dédicace*

*Je dédie ce modeste travail à*

- *À mes parents, mes frères, mes sœurs,*
- *À tout mes amies sans exceptions.*
- *À toute la promotion sans exception.*

*NORA*

*SAAD*

---

<b>Introduction générale.....</b>	<b>1</b>
<b>Chapitre I : Généralités sur le traitement d'images</b>	
Préambule.....	3
I.1 Définition de l'image.....	3
I.2 Image numérique .....	4
I.2.1 Qualité d'une image numérique .....	4
I.2.2 Caractéristique d'une image numérique .....	4
I.2.2.1 Dimension .....	4
I.2.2.2 Résolution .....	4
I.2.2.3 Luminance.....	4
I.2.2.4 Contraste .....	5
I.2.2.5 Contour .....	5
I.3 Notion d'histogramme.....	5
I.3.1 Histogramme uni modal .....	5
I.3.2 Histogramme bimodal .....	5
I.3.3 Histogramme multimodal .....	5
❖ Egalisation d'histogramme.....	6
I.4 Bruit .....	7
I.4.1 Causes du bruit .....	7
I.4.2 Distorsions géométriques .....	8
I.4.3 Modélisation du bruit .....	9
✓ Bruit impulsionnel .....	9
I.4.4 Filtrage d'images .....	10
I.4.4.1 les filtres linéaires.....	10
a) Filtre passe bas .....	10
b) Filtre passe haut .....	10
c) Filtre directionnel .....	11

I.4.4.2 Les filtres non linéaires .....	11
a) Filtre d'ordre .....	11
b) Filtre de position.....	11
c) Filtre médian .....	11
I.5 Définition de la texture .....	12
I.5.1 Texture périodique (macro-texture) .....	12
I.5.2 Texture stochastique (micro textures) .....	13
I.5.3 Texture hybride (directionnelle) .....	13
I.5.4 Méthode d'analyse de texture .....	13
❖ Méthode statistique.....	14
❖ Analyse texturale par cooccurrence .....	14
I.6 La Segmentation d'images texturées .....	15
I.6.1 Les méthodes de segmentation de textures.....	16
I. 6.1.1 segmentation par région.....	16
➤ Méthode par séparation (Quad-Tree).....	16
➤ Méthode par fusion.....	16
➤ Méthode par séparation/fusion.....	16
I.6.1.2 Méthodes de détection de contours.....	16
I.7 compression d'une image .....	17
I.7.1 Définition de la compression d'image.....	17
I.7.2 Principe générale de la compression d'image.....	17
I.7.2.1 decorrelation .....	18
I.7.2.2 quantification .....	18
I.7.2.3 Codage entropique.....	18
I.7.3 Principe méthode de compression d'image.....	18
I.7.3.1 Compression fractale .....	19
I.7.3.2 La compression par ondelette.....	20
Discussion .....	21

## Chapitre II : Transformée en ondelettes

Préambule.....	22
II.1 La transformée de FOURIER (TF).....	22
II.2 Transformée de Fourier Fenêtré (STFT).....	23
II.3 La transformée en ondelettes (WT) .....	25
II.3.1 Définition.....	25
II.3.2 Condition d'admissibilité .....	26
II.3.3 La transformée en ondelettes continue (CWT).....	27
II.3.4 La transformée en ondelettes discrète (DWT) .....	27
II.3.5 L'analyse multi-résolution.....	28
II.3.6 Fonction d'échelle et d'ondelette.....	29
II.3.7 L'algorithme de S.MALLAT .....	30
II.3.7.1 Analyse .....	30
II.3.7.2 Synthèse .....	32
II.3.7.3 Les filtres de décomposition/reconstruction.....	33
II.3.8 Ondelettes orthogonales .....	34
II.3.9 Ondelettes bi-orthogonales .....	34
II.3.10 Passage à deux dimensions.....	35
II.3.11 Propriétés des ondelettes .....	39
1) Régularité .....	39
2) Support Compact .....	39
3) Localisation .....	39
4) Orthogonalité .....	39
5) Symétrie.....	39
Conclusion .....	40

## Chapitre III : les bandelettes

Préambule .....	41
III.1 Première génération de bandelettes .....	41
III.1.1 Géométrie.....	41
III.1.1.1 Contours, courbes et flots.....	41
III.1.2 Bandelettes.....	42
III.1.2.1 Déformation de l'espace .....	42
III.1.2.2 Relation flot-courbe.....	43
III.1.2.3 Opérateur de déformations sur les fonctions.....	44
III.1.2.4 définition de base de bandelettes .....	46
❖ Rôle et choix des deux bases.....	47
III.1.2.5 Base de bandelettes.....	48
III.2 Bandelettes deuxième génération .....	49
III.2.1 Transformée en bandelettes sur un petit carré .....	49
III.2.1.1 Paramétrisation de la géométrie.....	50
III.2.1.2 Approximation dans une meilleure base de bandelettes.....	51
III.2.2 Segmentation en quadtree .....	51
III.2.3 Bases et dictionnaire de bandelettes.....	52
III.2.4 Structure d'un dictionnaire de bandelettes .....	54
❖ quantification de la géométrie .....	54
❖ Construction du dictionnaire .....	55
III.2.5 Transformée en bandelettes discrète .....	55
❖ Image discrétisée.....	55
❖ Fonction d'échelles en ondelettes.....	55
❖ bases de bandelettes discrètes.....	56
❖ Transformée en bandelettes discrètes.....	56
III.3 Transformer en ondelettes par groupement .....	56
III.3.1 La géométrie de 16 coefficients d'ondelettes .....	56
III.3.2 Construction de bases de bandelettes par groupements directionnels .....	58
Conclusion .....	59

## Chapitre IV : Application

Préambule .....	60
IV. 1. Application au débruitage .....	60
IV.1.1 Débruitage par seuillage .....	60
IV.1.2 Résultats numériques .....	61
IV.2 Applications dans le domaine de la texture .....	66
IV.2.1. Champ d'association multi-échelles .....	67
IV.2.2. Synthèse par égalisation d'histogrammes .....	67
IV.2.3. Transformée en bandelettes par groupements .....	69
IV.2.4. Algorithme générique de synthèse de texture .....	71
IV.3. Application à la Compression .....	75
IV.3.1 Algorithme de compression en bandelettes.....	75
IV.3.2 Algorithme rapide.....	76
IV.3.3. Choix de la taille des blocs.....	80
IV.3.4. Performance de la transformée en bandelette en compression.....	81
Conclusion .....	84
<b>Conclusion générale.....</b>	<b>85</b>

## **Résumé :**

Dans ce travail, nous nous sommes intéressés à la représentation géométrique des images. Ce travail consiste d'une part, à faire une recherche bibliographique dans le domaine de traitement d'images en générale et d'autre part à présenter une nouvelle technique qui capture la régularité géométrique des images le long des contours.

La transformée en bandelettes représente efficacement les contours des images, grâce aux bases orthogonales utilisées, et les zones de régularité homogène, sous forme d'un résidu décomposé en ondelettes.

Les bandelettes ont montré bien leur supériorité par rapport aux ondelettes, car elles permettent de capturer au mieux les régularités géométriques le long des contours, à savoir dans les applications de: débruitage, où les images restituées avec les bandelettes semblent visuellement plus fines que les images obtenues par ondelettes. L'analyse de texture, où certaines textures synthétisées ne sont pas proches des textures d'origines cela revient aux structures géométriques qui sont relativement longues. La compression, où la transformée en bandelettes respecte mieux les contours très marqués de l'image par rapport à la transformée en ondelettes.

## **Mot clé :**

Texture d'images, débruitage d'images, compression d'images, ondelettes, bandelettes.

### Introduction générale :

La vision de science définissant les bases théoriques et algorithmiques permettent d'extraire et d'analyser automatiquement des informations relevées dans notre environnement à partir d'une ou d'un ensemble d'images ou bien d'une séquence d'images, sur une machine spécialisée. Cette définition laisse imaginer combien ce domaine de recherche est vaste. Il est donc impératif de réfléchir à la manière d'aborder ce problème.

L'analyse de l'image s'inscrit dans une problématique appelée la vision par ordinateurs, la définition d'une méthode générale pour l'analyse d'image est actuellement très difficile vue la diversité des problèmes abordés. De même, il est délicat d'élaborer un système complet de vision par ordinateur. Nous allons alors, nous intéresser plus particulièrement à la représentation géométrique des images.

Un regard naïf sur les images peut laisser penser que la géométrie se réduit aux contours d'objets distincts comme dans le monde de la bande-dessinée. De tels contours existent mais constituent une faible partie des « structures géométriques » des images. Dans une image réelle, les textures ont souvent une géométrie beaucoup plus complexe.

D'un point de vue mathématique, les outils classiques de la géométrie différentielle peuvent décrire les propriétés des contours lorsqu'ils sont bien définis. L'analyse harmonique donne avec la transformée en ondelettes une première réponse flexible pour représenter les composantes régulières et les textures des images. Ainsi les bases orthogonales d'ondelettes ont été adoptées pour le standard de compression d'images JPEG2000. Cependant les bases d'ondelettes sont sous-optimales pour la représentation de structures géométriquement peu régulières. Ceci mène à la représentation géométrique d'images par bandelettes.

Notre travail porte sur l'étude et l'évaluation d'analyse de texture et l'amélioration des méthodes de débruitage et des techniques de compression basées sur la transformée en bandelettes établie dans la littérature.

Pour se faire, le premier chapitre sera consacré aux notions de base utilisées en traitement d'images.

Le principe mathématique de la transformation en ondelettes est proposé en deuxième chapitre.

Dans le troisième chapitre, nous abordons d'abord la première génération de la transformée en bandelettes proposée par Le Pennec dans sa thèse [12]. Ensuite, nous présentons la seconde génération de la transformée en bandelettes dans ses deux versions qui sont dérivées des ondelettes.

Dans le dernier chapitre, nous présentons les résultats de l'application au débruitage dans un frame de bandelettes, une étude d'un algorithme générique dans le but de synthétiser des textures géométriques et quelques performances de l'application de la transformée en bandelette en compression.

Nous terminons notre travail par une conclusion générale et des perspectives.

**Préambule :**

Le traitement d'image est né de l'idée de la nécessité de remplacer l'observateur humain par la machine. L'image ou les signaux provenant des capteurs ont alors été numérisés pour pouvoir être traités par ordinateur.

Le traitement d'image est l'ensemble des méthodes et techniques permettant d'améliorer facilement et efficacement l'aspect visuel d'une image, puis d'en extraire les informations jugées pertinentes.

La segmentation est une étape primordiale dans toute opération d'analyse d'image et la vision par ordinateur. Elle est utilisée avant les étapes d'analyse et de prise de décision dans plusieurs processus de traitement d'image. Son but est l'extraction à partir d'une image numérique, des informations « utiles » qui pourront permettre de localiser et de délimiter les entités présentes dans l'image.

La texture est une entité très importante dans l'analyse et la synthèse d'image. En effet, les principales informations dans l'interprétation du message visuel pour un observateur humain sont les contours et les textures.

La compression d'image fait l'objet de nombreuses études qui portent sur l'amélioration des algorithmes de compression ainsi que la mise au point de nouvelles techniques et standards de compression. Aujourd'hui, on distingue un outil très utilisé en compression qui est les ondelettes.

Dans ce chapitre, nous débutons par des notions fondamentales de traitement d'image, ensuite nous présentons quelques applications utilisées dans ce domaine.

**I.1 Définition de l'image :**

L'image n'est qu'une représentation planaire d'un objet, d'une scène en générale dans l'espace tridimensionnel.

L'image peut être représentée sous forme d'un ensemble de points répartis dans une surface donnée. La valeur de chaque point est la réalisation d'une variable aléatoire liée à l'apparition de la luminance dans un site donné de l'image. Sur l'écran de visualisation, une image numérique se présente comme ensemble de points appelés pixels ayant L lignes et K

colonnes. On dit alors que l'image possède une dimension de  $K \times L$  pixels. Le mot « pixel » vient de l'expression anglaise « Picture Element » qui veut dire élément d'image.

Une image peut être considérée comme un signal bidimensionnel variant dans l'espace suivant les directions horizontales X et verticales Y.

## **I.2 Image numérique :**

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

### **I.2.1 Caractéristique d'une image numérique :**

Le plus petit élément comportant une image appelé Pixel. Alors, une image est représentée sous forme d'un tableau qui contient un ensemble de pixels. L'image est un ensemble structuré d'information caractérisée par plusieurs paramètres :

#### **I.2.1.1 Dimension :**

La représentation de l'image est une matrice dont les éléments sont des valeurs numériques, donc la multiplication du nombre de ligne par le nombre de colonnes nous donnera la taille de l'image en pixels.

#### **I.2.1.2 Résolution :**

La résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour désigner le nombre total de pixels est affichable horizontalement ou verticalement sur un moniteur. Plus le nombre de pixels est grand plus la résolution est meilleure.

#### **I.2.1.3 Luminance :**

C'est le degré de luminance des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité d'une surface par l'aire apparente de cette surface, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet.

La luminance (brillance) d'une image numérique en niveau de gris est définie comme la moyenne des pixels de l'image :

$$L(I) = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \quad (\text{I. 1})$$

Avec : M : nombre de colonnes.

N : nombre de lignes.

$f(x, y)$  : la valeur de niveau de gris du point(x, y).

#### **I.2.1.4 Contraste :**

Le contraste est défini en fonction des luminances entre deux zones d'image, plus précisément entre les régions sombre et les régions claires de cette image. Si L1 et L2 sont les degrés de luminosité respectivement de deux zones voisines A1 et A2 d'une image, le contraste C est défini par le rapport :

$$C = \frac{L1 - L2}{L1 + L2} \quad (\text{I. 2})$$

#### **I.2.1.5 Contour :**

Les contours présentent les frontières entre les objets de l'image ou la limite entre deux pixels dont le niveau de gris représente une différence significative.

### **I.3 Notion d'histogramme :**

L'histogramme des niveaux de gris de l'image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris. L'histogramme permet alors de donner la distribution des niveaux de gris de l'image appelée « dynamique de l'image ».

L'histogramme est un outil privilégié en analyse d'image car il représente un résumé simple, mais souvent suffisant du contenu de l'image. On peut distinguer trois types d'histogramme d'image :

#### **I.3.1 Histogramme uni-modal :**

Ce type d'histogramme n'a qu'un seul pic, il présente soit un objet soit un fond.

#### **I.3.2 Histogramme bi-modal :**

Il est formé de deux modes bien séparés (deux pics séparés par une vallée) et l'on déduit ainsi qu'il existe un objet sur un fond.

#### **I.3.3 Histogramme multi-modal :**

Il est formé de plusieurs modes séparés (plusieurs pics séparés par plusieurs vallées) qui nous renseigne sur la présence de plusieurs objets.

❖ **Egalisation d'histogramme :**

L'égalisation d'histogramme est une transformée de niveau de gris dont le principe est d'équilibrer le mieux possible la distorsion des pixels dans la dynamique (idéalement en cherche à obtenir un histogramme plat).

Pour une image  $X$  en niveau de gris codée sur  $L$  niveaux, on définit  $n_k$  le nombre d'occurrences du niveau  $x_k$ . La probabilité d'occurrence d'un pixel de niveau  $x_k$  dans l'image est :

$$p_x(x_k) = p(x = x_k) = \frac{n_k}{n}, \quad 0 \leq k < L \quad (I.3)$$

Avec :

$n$  : Le nombre total de pixel de l'image.

$p_x$  : L'histogramme normalisé sur  $[0,1]$ .

La transformation  $T$  est obtenue pour chaque pixel de valeur  $x_k$  de l'image d'origine, associe une nouvelle valeur  $s_k$ ,  $s_k = T(x_k)$  est alors définie par :

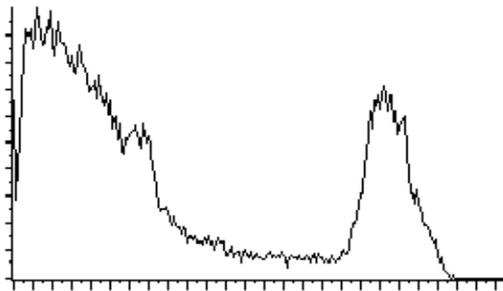
$$T(x_k) = \frac{(L-1)}{n} \sum_n^k n_j \quad (I.4)$$



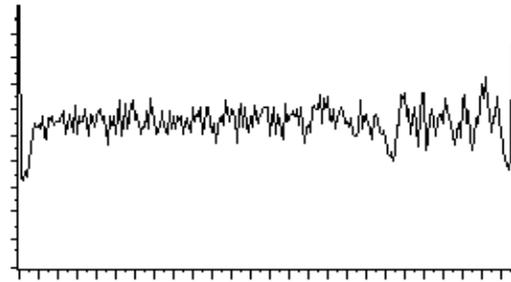
Avant égalisation d'histogramme.



Après égalisation d'histogramme.



Histogramme avant égalisation.



Histogramme après égalisation.

**Fig I.1** : traitement par égalisation d'histogramme.

## I.4 Bruit :

### I.4.1 Causes du bruit :

Un appareil d'acquisition d'images numériques génère toujours un bruit. Une cause principale liée au contexte d'acquisition sont dus à une sur ou sous illumination réduit l'intervalle de couleurs de la scène ainsi que son nombre de couleurs utilisées.

Parmi les bruits liés au contexte d'acquisition, on peut citer la perturbation des capteurs, par exemple des perturbations magnétiques pendant une acquisition d'Imagerie par Résonance Magnétique, ou des perturbations sur antenne de télévision lors de la réception sur carte tuner.

Certains capteurs (ceux des appareils photos) induisent des distorsions géométriques ou d'intensités, dues à la présence d'un certain nombre de dioptries. L'étape d'échantillonnage est aussi source de bruit, notamment de phénomène de Moiré lorsque les conditions de Shannon et Nyquist ne sont pas respectées, ou de bruit poivre et sel lorsque des objets de la scène sont

projetés dans une image de la taille d'un pixel. La quantification apporte un bruit dit de quantification. Enfin, la transmission des images est l'occasion de perturbation à savoir perte de données ou corruption de données.



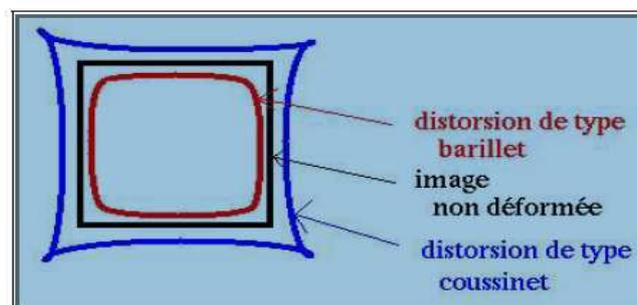
**Fig I.2** exemple d'image bruitée

#### I.4.2 Distorsions géométriques :

Les distorsions géométriques de l'image sont dues à des défauts de l'optique des appareils photos : mauvais alignement des centres optiques des dioptrés, courbure non parfaite des lentilles,....

Nous nous intéressons à une première approximation de ces distorsions : ce sont les distorsions radiales. En pratique, c'est la distorsion géométrique radiale symétrique qui a l'influence la plus importante. Les distorsions géométriques radiales les plus couramment observées sont de deux formes, comme illustré par la figure ci-dessous :

- distorsions en forme de barillet.
- distorsions en forme de coussinet.



**Fig I.3** Les types de distorsions les plus observés : barillet et coussinet

La distorsion est représentée par deux composantes radiale et tangentielle. Si on note  $\rho$  et  $\theta$  les coordonnées polaires d'un point  $(x, y)$  de l'image,  $\delta_\rho(\rho, \theta)$  et  $\delta_\theta(\rho, \theta)$  les composantes radiale

et tangentielle de la distorsion,  $\delta_x(x, y)$  et  $\delta_y(x, y)$ , les composantes selon  $x$  et  $y$  de la distorsion, s'expriment au premier ordre selon :

$$\delta_x(x, y) = \cos(\theta)\delta_\rho(\rho, \theta) - \sin(\theta)\delta_\theta(\rho, \theta) \quad (\text{I.5})$$

$$\delta_y(x, y) = \sin(\theta)\delta_\rho(\rho, \theta) + \cos(\theta)\delta_\theta(\rho, \theta) \quad (\text{I.6})$$

Les distorsions radiales concernent les défauts de courbure des lentilles constituant la caméra. Cette déformation de la courbure est cependant utilisée dans le but d'avoir une luminosité constante sur toute la surface de l'image, afin d'éliminer l'effet de vignetage (luminosité forte au centre, plus faible en périphérie). Une approximation suffisante en pratique pour modéliser est donnée par :

$$\delta_x(x, y) = k_1 x (x^2 + y^2) \quad (\text{I.7})$$

$$\delta_y(x, y) = k_1 y (x^2 + y^2) \quad (\text{I.8})$$

$k_1$  étant une constante dont le signe détermine le sens de distorsion  $k_1 > 0$  correspond aux distorsions en forme de coussinet ;  $k_1 < 0$  aux distorsions en forme de barillet.

### I.4.3 Modélisation du bruit :

Le bruit peut être dépendant des données de l'image (bruit de quantification) ou indépendant (les poussières sur l'objectif). On peut alors modéliser le bruit comme additif ou multiplicatif. Il est bien plus simple de traiter un bruit additif que multiplicatif. Afin d'éliminer le bruit, on peut considérer qu'il concerne des hautes fréquences non présentes dans l'image et qu'il suffit d'effectuer un filtre passe-bas pour améliorer l'image. Malheureusement, cela n'est pas toujours simple et évident.

Nous allons présenter un type de bruit le plus répandu est le bruit impulsionnel (bruit blanc gaussien).

#### ✓ Bruit impulsionnel :

Un bruit impulsionnel est défini par une densité de probabilité :

$$f(a) = C. e^{-k|a|^\alpha} \quad (\text{I.9})$$

Pour  $\alpha = 1$ , il s'agit d'un bruit exponentiel et pour  $\alpha = 2$ , il s'agit d'un bruit gaussien.

Le bruit blanc est un bruit gaussien de moyenne nulle. Sa variance est  $\sigma^2$ . Ce bruit tend à modéliser le bruit d'acquisition. La figure suivante illustre un exemple d'un bruit blanc ajouté à une image.



**Fig I.4** A gauche, l'image originale. A droite, l'image bruitée par un bruit blanc additif de  $\sigma = 16$ .

#### **I.4.4 Filtrage d'images :**

Pour améliorer la qualité visuelle de l'image, on doit éliminer les effets des bruits (parasites) en lui faisant subir un traitement appelé filtrage.

Le filtrage consiste à modifier la distribution fréquentielle des composantes d'une image selon des spécifications données.

Le système utilisé est appelé filtre numérique. Parmi ces systèmes, nous distinguons deux types : les filtres linéaires et non linéaires.

##### **I.4.4.1 les filtres linéaires :**

Le filtrage linéaire consiste à remplacer chaque niveau de gris de l'image par une combinaison linéaire des niveaux de gris des points voisins, les coefficients de cette combinaison sont définis par la réponse impulsionnelle du filtre.

##### **a) filtre passe bas :**

Il élimine les fréquences élevées, c'est-à-dire, il a tendance à éliminer le bruit, ce filtrage réduit aussi les hautes fréquences qui étaient présentes dans l'image, ce qui adoucit les contours, et rend l'image floue. En pratique, il faut faire un compromis entre l'atténuation du bruit et la conservation des détails des contours significatifs.

##### **b) Filtre passe haut :**

Il permet de mettre en évidence les contours des objets en amplifiant les hautes fréquences d'où le terme " le filtre d'accentuation ".

**c) Filtre directionnel :**

Dans certains cas, on cherche à faire apparaître des détails de l'image dans une direction bien déterminée. Pour cela, on utilise des filtres qui opèrent suivant des directions (horizontales, verticales et diagonales).

**I.4.4.2 Les filtres non linéaires :**

Dans ce type de filtre, la nouvelle valeur du pixel n'est plus une combinaison linéaire des pixels voisins, mais une autre forme de calcul excluant toute relation linéaire.

**a) Filtre d'ordre :**

Les filtres d'ordres procèdent en remplaçant les valeurs de chaque pixel par la valeur qui occupe un certain rang, lorsqu'on trie les valeurs observées dans un certain voisinage du pixel. Il existe deux types de filtre d'ordre :

- Filtrage par la médiane : la valeur du point centrale d'une fenêtre de l'image est affectée par celle de ses voisins en classant les valeurs de cette dernière par ordre croissant (en prenant la valeur médiane).
- Filtrage par le plus proche voisin radiométrique : dans ce cas, on attribue au point central la moyenne des K pixels voisins dont les valeurs radiométriques sont plus proches de la sienne.

**b) Filtre de position :**

Ce filtre utilise des voisinages  $5 \times 5$ , le sous-voisinage le plus homogène est sélectionné et son niveau de gris moyen est affecté au point central.

**c) Filtre médian :**

Le pixel et ses voisins forment un ensemble dont on calcule " la médiane", le pixel sera alors remplacé par cette valeur médiane.

*Exemple :*

1	4	5
3	1	6
3	7	8

On obtient après un tri croissant ce qui suit : 1 1 3 3 4 5 6 7 8 : la valeur médiane est égale à 4

➤ *Intérêt du filtre médian :*

- le pixel non représentatif dans le voisinage affectera peu la valeur médiane.
- la valeur médiane choisie étant le niveau de gris d'un des pixels considérés, on ne crée pas alors de nouveaux niveaux de gris dans l'image.



**Fig I.5 :** A gauche, l'image originale. Au milieu, l'image bruitée par un bruit blanc additif de  $\sigma = 16$ , A droite, l'image lissée par un filtre médian 3x3.

## I.5 Définition de la texture :

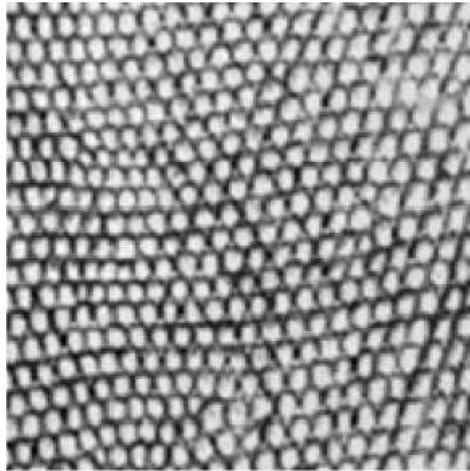
Plusieurs définitions de la notion de texture ont été présentées au cours des vingt dernières années. Néanmoins la définition précise d'une texture est encore sujette à controverse.

Toutefois, on peut réaliser une synthèse de ces définitions en introduisant les deux concepts qui sont à la base de la notion de texture : les statistiques portant sur les pixels et la répartition spatiale de motifs plus ou moins réguliers.

L'agencement spatial et statistique d'une population d'objets (motif) permet ainsi de définir la notion de texture. Cette combinaison peut être exploitée dans le cadre de texture déterministe ou stochastique.

### I.5.1 Texture périodique (macro-texture) :

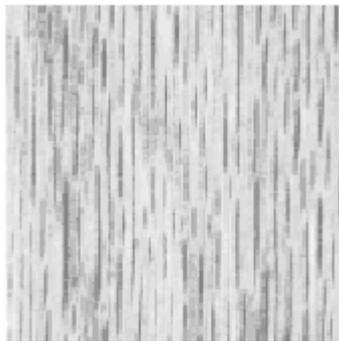
Elle représente un aspect régulier, sous forme de motifs répétitif spatialement placés selon une règle précise (ex : peau de lézard, mur de brique), donc c'est une approche structurelle ou déterministe.



**Fig I.6 :** exemple de texture périodique (peau de lézard)

### **I.5.2 Texture stochastique (micro textures) :**

Qui représente des primitives (microscopiques) distribuées de manière aléatoire (ex : bois, laine tissée, herbe) d'où une approche probabiliste cherchant à caractériser l'aspect anarchique homogène.



**Fig I.7 :** exemple de texture stochastique (bois).

### **I.5.3 Texture hybride (directionnelle) :**

C'est une combinaison des deux type précédent où on peut trouver par exemple la texture compose d'un seul motif de base réparti aléatoirement (ex : sable, canevas).

### **I.5.4 Méthode d'analyse de textures :**

Vu la difficulté d'une définition précise de la texture, plusieurs approches ont été élaborées ces dernière années pour identifier cette opération. On cite quelques méthodes essentiellement utilisées, elles sont de nature :

- structurelles ou déterministes.

- spatio-fréquentielles.
- statistiques.

➤ **Méthodes statistiques :**

Ces méthodes se basent aussi sur les relations entre les pixels et leur voisins en termes de variation des niveaux de gris, les attributs extraits de ces méthodes permettent de caractériser tout type de texture, même les textures fines et sans régularité apparente.

De nombreux descripteurs statistiques sont utilisés dans la discipline de traitement d'image texturée en niveau de gris, on retrouve entre autre [1] :

- les statistiques d'images.
- les histogrammes d'images
- les matrices de cooccurrences

Ces descripteurs peuvent être divisés en plusieurs catégories selon leur ordre. L'ordre du descripteur dépend de type d'interaction spatiale entre les pixels considérés, par exemple :

- pour les histogrammes d'images, on s'intéresse qu'au pixel lui-même, ce descripteur appartient donc à la catégorie des descripteurs statistiques de **premier ordre**.
- les matrices de cooccurrences sont quant-à-elles des descripteurs **d'ordre 2** car leurs calculs considèrent des couples de pixels.

❖ **Analyse texturale par cooccurrence :**

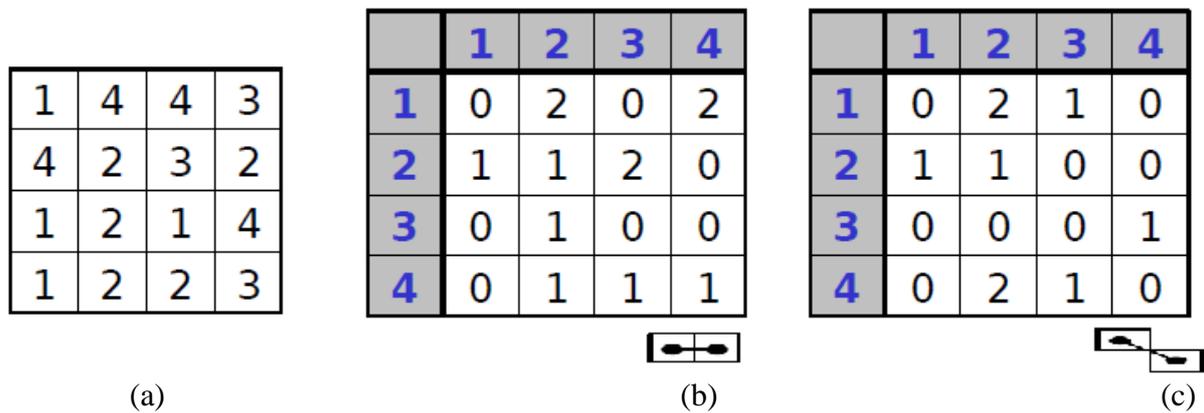
Les matrices de cooccurrences permettent d'estimer des propriétés des images relatives à des statistiques de second ordre. Cette méthode consiste à identifier les **répétitions de niveaux de gris** selon une **distance** et une **direction**.

Pour une translation  $t$ , la matrice de cooccurrence  $MC_t$  d'une région  $R$  est définie pour tout couple de niveau de gris  $(i, j)$  par :

$$MC_t(i, j) = \text{card}\{(s, s + t) \in R^2 \mid I[s] = i, I[s + t] = j\} \quad (I.10)$$

$MC_t(i, j)$  est donc le nombre de couple de site  $(s, s + t)$  de la région considérée, séparés par le vecteur de translation  $t$ , tel que  $s$  a pour niveau de gris  $i$  et  $(s+t)$  a pour niveau de gris  $j$ .

Exemple de calcul final des matrices.



**Fig I.8 :** Exemple de calcul d'une matrice de cooccurrence : (a)Image initiale, (b) Matrice de cooccurrence  $t=(1,0)$ , (c) Matrice de cooccurrence  $t=(1,45)$ .

## I.6 La Segmentation d'images texturées :

La segmentation selon le dictionnaire Larousse est la division d'une unité anatomique en plusieurs éléments. En imagerie, la segmentation d'image est la division (« partitionnement ») d'une image en zones homogènes selon un critère choisi (niveaux de gris, texture, etc....) afin de séparer les divers composants visibles et de les identifier ou bien c'est d'effectuer une partition de cette image en régions, telle que chacune d'elles possède plus ou moins une caractéristique que ne possèdent pas les régions voisines, mathématiquement, nous pouvons la modéliser de la façon suivante :

Soit une image  $I$ , la segmentation consiste à créer une partition de l'image  $I$  en sous ensembles  $R_i$  appelés régions, tels que :

$$\begin{cases} \forall i > R_i \neq \emptyset \\ \forall i, j; i \neq j : R_i \cap R_j = \emptyset \\ I = \cup R_i \end{cases} \quad (\text{I.11})$$

En d'autres termes, les conditions suivantes doivent être vérifiées :

- Chaque pixel doit appartenir à une classe.
- Les régions doivent être homogènes.
- Les régions doivent être deux à deux disjointes.

Il existe une grande variété de méthode de segmentation dans la littérature

### I.6.1 Les méthodes de segmentation de textures

Une fois les caractéristiques de textures déterminées, il reste à réaliser la segmentation de l'image en zones homogènes au sens de ces attributs. Nous allons décrire brièvement les techniques usuellement utilisées à cet effet : segmentation par régions, segmentation par les méthodes de détection de contours et segmentation suivant un modèle markovien.

#### I.6.1.1 Segmentation par régions

Les approches régions recherchent la similarité, elles considèrent une région comme un ensemble de pixels connexes dont les attributs de textures sont proches les uns des autres, cependant la localisation des frontières reste généralement peu précise.

Les techniques de traitement d'images pour la segmentation par régions se décomposent globalement suivant trois approches [2] :

- **Méthode par séparation (Quad-Tree)** : elle consiste à tester un critère d'homogénéité sur une zone large de l'image. Si le critère est respecté, la segmentation est faite. Sinon, on décompose la zone en quatre zones plus petites et on ré applique la méthode sur chaque petite zone, jusqu'à ce que le critère soit validé. Cela correspond à réaliser un arbre des zones segmentées ;
- **Méthode par fusion** : inversement à la méthode précédente, elle consiste à explorer l'image par petites zones et à les faire croître, jusqu'à ce que le critère soit validé ;
- **Méthode par séparation/fusion** : elle combine les deux méthodes précédentes. On applique d'abord la division par Quad-Tree, puis on parcourt l'arbre dans le sens inverse en fusionnant les régions satisfaisant le critère d'homogénéité.

#### I.6.1.2 Méthodes de détection de contours :

L'approche locale de détection de contour consiste à balayer l'image avec une fenêtre d'analyse. Un opérateur est appliqué à chaque pixel de l'image afin de détecter les transitions significatives au niveau de l'attribut choisi.

Notons que les méthodes classiques de détection de contours ne peuvent pas être directement utilisées pour l'image texturée, puisque les pixels dans la texture sont représentés par plusieurs attributs. La détection de contours dans une image texturée consiste à rechercher les discontinuités importantes des attributs de texture. Pour l'intégrer, les informations spatiales, les opérateurs de dérivation sont souvent utilisés [3].

L'approche contour n'aboutit pas directement à une segmentation car les contours détectés ne sont pas toujours connexes. Il existe cependant des techniques permettant

d'obtenir des contours fermés. Dans ce cas, on observe une parfaite dualité entre les contours et les régions.

Comme les opérateurs de dérivation sont très sensibles au bruit, des images bruitées doivent être préalablement lissées. Un grand nombre d'opérateurs gradient ont été proposés. Ils se distinguent entre eux principalement par le choix du filtre de lissage. Le lissage et la dérivation sont en pratique réunis dans un seul filtre. Des opérateurs optimaux ont été proposés par un ensemble de chercheurs tel que Canny et Derriche [3] [4].

## I.7 Compression d'une image :

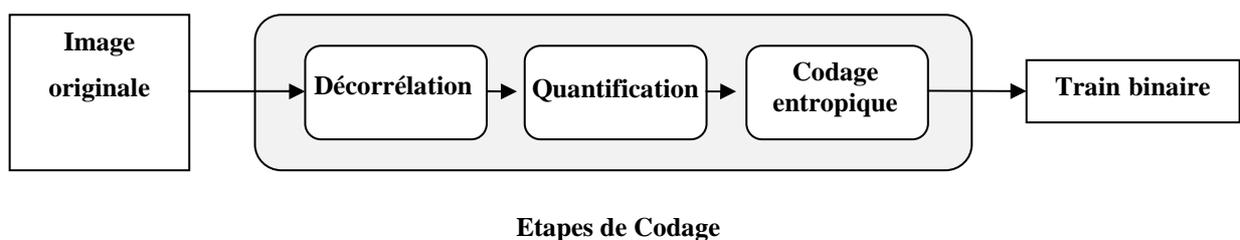
### I.7.1 Définition de la compression d'image :

La **compression d'image** est la réduction du nombre de bits par pixels à stocker ou à transmettre. Cette compression a pour utilité de réduire la redondance des données d'une image afin de pouvoir l'emmagasiner sans occuper beaucoup d'espace ou la transmettre rapidement.

On distingue deux grandes familles d'algorithmes de compression : les méthodes dites sans pertes (réversibles) qui restituent de façon parfaite les images mais avec un faible taux de compression, et les méthodes dites avec pertes (irréversibles) qui permettent d'atteindre des taux de compression élevés mais qui introduisent des distorsions lors de la restitution.

### I.7.2 Principe générale de la compression d'image:[5]

Le schéma d'un système de compression classique est représenté sur la figure 4. La première étape est la décorrélation. Elle permet d'exploiter les redondances statistiques de l'image. Ensuite vient l'étape de quantification. C'est là que la perte d'information a lieu. Enfin, le codage entropique des données quantifiées constitue la dernière étape. Elle permet d'organiser le train binaire en compressant les données sans altérer l'information.



**Fig I.9** Schéma générale d'un système de compression.

**I.7.2.1 décorrélation :**

Elle consiste à transformer les pixels initiaux en un ensemble de coefficient moins corrélés. La dépendance existante entre chacun des pixels et ses voisins (la luminosité varie très peu d'un pixel à un pixel voisin) traduisent une corrélation très forte sur l'image. Donc on essaye de réduire le volume d'information en effectuant une opération de décorrélation des pixels.

Cette étape est réversible sans pertes d'informations. Elle fait intervenir une transformée. Par exemple la DCT (Discrete Cosine Transform) est utilisée dans le très populaire standard de compression d'images JPEG ou la transformée en ondelettes qui est à la base du standard JPEG2000.

**I.7.2.2 quantification :**

La quantification des coefficients obtenus a pour but de réduire le nombre de bits nécessaire pour leur représentation. Elle approxime chaque valeur d'un signal par un multiple entier d'une quantité « Q » appelé « quantum élémentaire » ou « pas de quantification » la quantification peut être scalaire, uniforme ou vectorielle

**I.7.2.3 Codage entropique**

Le codage entropique permet la compression des coefficients quantifiés en générant un train binaire de longueur proche de la limite basse théorique de Shannon. Plusieurs types de codage entropique existent. Le codage par Run-Length est utilisé dans le fax mais aussi dans le standard JPEG. Le codage par dictionnaire comme LZW (Lempel-Ziv-Welch) est utilisé dans le format d'image GIF (Graphics Interchange Format). Le codage par statistique comme le codage de Huffman est aussi utilisé dans le format JPEG. Enfin, un codage arithmétique est effectué dans le standard JPEG2000.

**I.7.3 Principales méthodes de compression d'image : [6]**

Les méthodes de compression d'images dites avec perte constituent la majorité des travaux de recherche, en particulier lors de l'étape de quantification. Nous pouvons distinguer :

1. les méthodes basées sur la Quantification Scalaire (QS) : elles consistent à traiter les valeurs de pixels (ou de coefficients) de manière individuelle. Différent type de Quantification Scalaire existent, et sont encore utilisés, par exemple dans le standard jpeg2000.
2. les méthodes basées sur la Quantification Vectoriel (QV) : elles traitent à la fois un ensemble de pixels ou de coefficient appelés vecteurs. Elles permettent théoriquement d'être toujours plus performante que les méthodes basées sur la Quantification Scalaire.

Dans la littérature, plusieurs propositions de schémas de compression en été développé que ce soit dans le domaine fréquentiel ou spatial, on cite quelques unes :

### **I.7.3.1 Compression fractale :**

Cette technique relativement nouvelle est basée sur une idée très originale; en supposant qu'une image possède des propriétés fractales, reconstruisons-la à partir d'elle-même au travers des transformations mathématiques affines contractantes [13]. Plusieurs phénomènes naturels sont de type fractal. Par exemple, les nuages, les rives d'un cours d'eau, les arbres, ...etc. Une image météorologique peut être considérée comme une figure géométrique extrêmement irrégulière et composée de plusieurs types de nuages. Cette figure, présente les mêmes irrégularités à plusieurs échelles, on dit alors qu'il y a auto similarité entre les blocs ou parties (les nuages) qui constituent l'image à différentes échelles. En conséquence, les images satellitaires peuvent être efficacement simulées et codées par une technique fractale.

Le principe du codage fractal est de partitionner une image en blocs destination (*range bloc*) et blocs source (*domain bloc*). Il s'agit ensuite de rechercher pour chaque bloc source considéré après transformations contractantes (réduction, symétrie et rotation), un bloc destination qui lui ressemble. En effet, le codage fractal d'une image consiste à la représenter par un système de fonctions itérées (*IFS*) [13]. Le code IFS est constitué des coordonnées des blocs source et des paramètres des transformations appliquées. Au décodage, on applique une procédure itérative, contenant le code IFS, sur une image initiale quelconque en remplaçant les blocs source par ceux transformés. Si certaines conditions sont vérifiées, l'image finale est une bonne approximation de l'image originale.

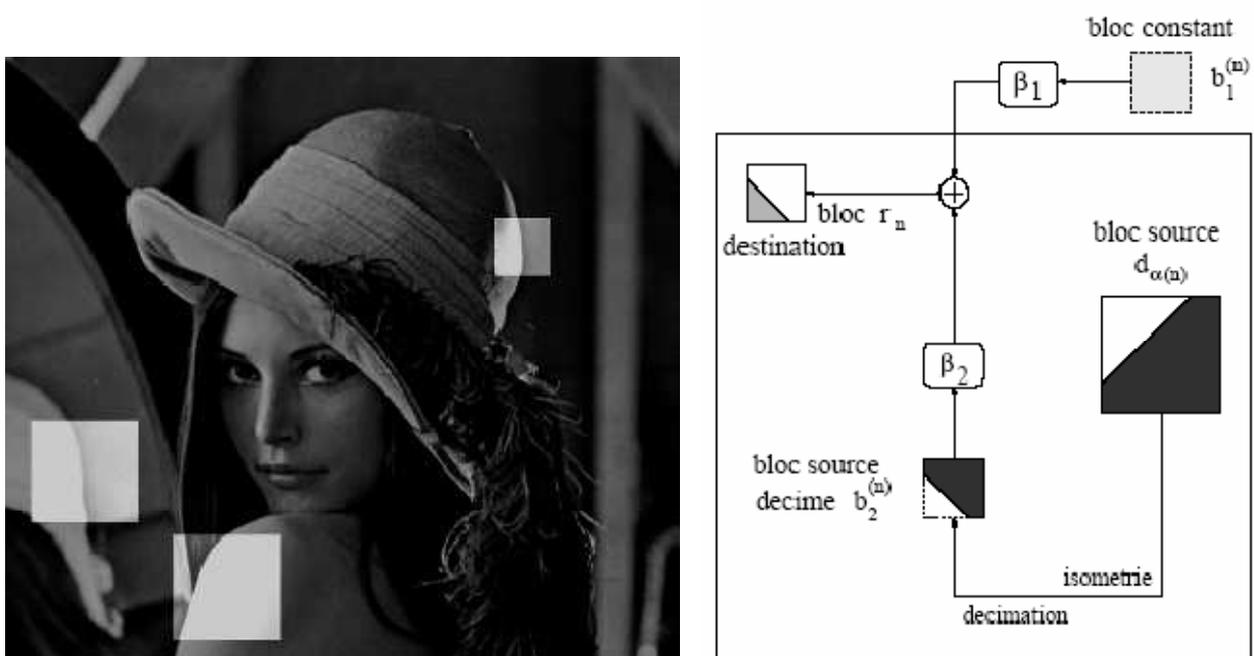
Le codage fractal est une méthode irréversible. L'inconvénient majeur de cette approche est le coût en temps de calcul pendant la procédure de codage.

❖ **Intérêts :**

- Compression indépendante de la taille de l'image ;
- Image restituée ne subit pas d'effets de pixellisation (lors d'un zoom, l'image est recalculée, donc elle pourra devenir floue mais pas de pixellisation).

❖ **Inconvénients:**

- Temps de compression très coûteux par rapport aux autres méthodes de compression ;
- format de fichier non standardisé (donc non reconnu par les navigateurs Internet).



**Fig I.10 :** Principe du codage par fractales

A gauche, similarités locales ; A droite, collage d'un bloc source sur un bloc destination

### I.7.3.2 La compression par ondelettes :

La transformée en ondelettes a été inventée par Jean Morlet, un ingénieur français, c'est une technique consistant à décomposer un signal en sous bande. Autrement dit un signal est décomposé en un ensemble de signaux de résolution inférieure.

A l'heure actuelle, la transformée en ondelettes (en l'anglais Wavelet Transform) est la technologie de compression utilisée dans la norme JPEG2000. Cette norme a un énorme avantage par rapport aux autres normes à savoir, le plus important est le décodage progressif.

Un algorithme typique de codage JPEG2000 se divise essentiellement en cinq modules: transformée couleur, transformée en ondelettes discrète, quantification, codage entropique et allocation de débit.



**Fig I.11 :** Schéma typique d'un codeur JPEG 2000

JPEG 2000 utilise une transformée en ondelettes biorthogonales séparables pour effectuer une décomposition dyadique de l'image entière en sous-bandes de fréquences (cette technique sera détaillée dans le chapitre suivant).

Dans le Jpeg2000, une ondelette de type Daubechies 9/7 est utilisée pour la compression avec perte. Pour la compression sans perte, on utilise une LeGall 5/3. Les coefficients obtenus après décomposition en ondelette de LeGall sont rationnels, cette particularité est nécessaire pour assurer une reconstitution parfaite du signal original. Ces ondelettes n'ont pas été choisies au hasard, elles remplissent plusieurs critères mathématiques qui permettent d'obtenir de bons résultats sur les images. Il faut savoir que, selon le type de signal à analyser, les ondelettes utilisées ne seront pas les mêmes.

### **Discussion :**

Ce chapitre est centré sur tout ce qui concerne le domaine du traitement d'images. Ce dernier est le processus qui prend une image en entrée et produit une autre image, qui est le résultat d'une opération de transformation qui vise à satisfaire la vision humaine. Nous avons abordé la notion de segmentation, de caractérisations de textures, de filtrage ainsi que de compression qui seront les traitements sur lesquels s'articule notre travail.

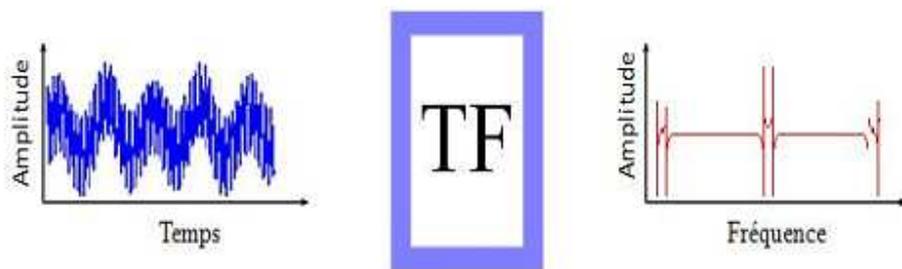
## Préambule :

La transformée de Fourier est la technique la plus utilisée dans le traitement du signal. Elle permet le passage du domaine temporel au domaine fréquentiel. Dans ce passage, l'information temps est perdue pour les signaux non stationnaires, d'où l'introduction de la technique des fenêtres. Grâce à cette technique, il est devenu possible de connaître à quel moment un événement du signal s'est produit mais avec une largeur de la fenêtre réglée à l'avance.

La transformée en ondelettes ou Wavelets est une solution, elle permet d'avoir la taille de la fenêtre variable. Au lieu de représenter un signal comme la somme pondérée de sinusoïdes, nous avons la somme pondérée d'ondes translatées et dilatées dans le temps.

### II.1 La transformée de FOURIER (TF):[7]

La transformée de Fourier est un outil qui sert à mesurer la fréquence d'un signal. Son principe repose sur le fait que toute fonction périodique peut être représentée comme la somme d'une série de sinus et de cosinus dont on fait varier d'une part les amplitudes en les multipliant par des coefficients, et d'autre part les phases en les décalant de manière à ce qu'elles s'additionnent ou se compensent.



**Fig II.1** Transformée de Fourier.

La définition de la TF est donnée par la formule suivante :

$$F(w) = \int_{-\infty}^{+\infty} f(x)e^{-j\omega t} dx \quad (\text{II.1})$$

Cependant, l'analyse de Fourier, de par sa nature, montre assez vite ses limitations : son calcul nécessite la connaissance de toute l'histoire temporelle du signal. De plus, dans une transformée de Fourier, l'information sur le temps est présente (la transformée inverse est

donc possible), mais elle est cachée dans les phases : elle est en pratique impossible à extraire. Il en est donc réduit à étudier un signal soit en fonction du temps, soit en fonction des fréquences qu'il contient, sans possibilité de conjuguer les deux analyses. La définition de la TF inverse est donnée par la formule suivante:

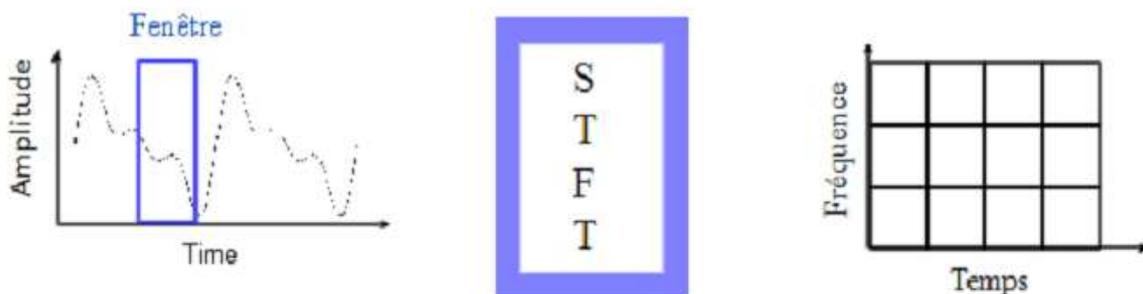
$$f(x) = \int_{-\infty}^{+\infty} F(w)e^{j\omega t} dw \quad (II.2)$$

Tel que :  $w=2\pi$

On peut utiliser la transformée de Fourier pour extraire des informations fréquentielles d'une image, toutefois le principal problème de la transformée de Fourier est son manque de résolution temporelle. Cela signifie simplement que si on est effectivement capable de détecter toutes les fréquences qui apparaissent dans un signal, on est en revanche incapable de déterminer à quel moment elles se produisent dans le signal. Il existe une transformée de Fourier plus « locale » donnant des informations mieux localisées, il s'agit de la transformée de Fourier Fenêtré (STFT : Short Time Fourier Transform).

## II.2 Transformée de Fourier Fenêtré (STFT) : [7]

Pour pallier le manque d'information sur le temps dans la transformée de Fourier, une nouvelle méthode d'analyse est donc introduite : elle utilise une « Fenêtre glissante ». Cette méthode, pouvant être adaptée aux signaux non-stationnaires, est très proche de l'analyse spectrale: on définit une fenêtre qui sera utilisée comme masque sur le signal, et dans laquelle on considère que le signal est localement stationnaire, puis on décale cette fenêtre le long du signal afin de l'analyser entièrement.



**Fig II.2** Transformée de Fourier fenêtrée.

La transformée de Fourier fenêtrée remplace la sinusoïde de la transformée de Fourier par le produit d'une sinusoïde et d'une fenêtre localisée en temps.

La définition de la STFT est donnée par la formule suivante :

$$Gf(\mu, \tau) = \int_{-\infty}^{+\infty} f(t)G(t, \tau)e^{-2\pi\mu jt} dt \quad (II.3)$$

Avec :

$f(t)$  : étant le signal lui-même.

$G(t)$  : est la fonction fenêtre.

$\mu$  : représente la fréquence.

$\tau$  : représente le paramètre de translation.

Comme l'indique l'équation, la STFT du signal n'est autre que la TF du signal multiplié par la fonction fenêtre. Pour chaque valeur de  $f$  et de  $\tau$ , on calcule un nouveau coefficient de la STFT.

Ainsi, l'étude d'un signal avec la STFT permet d'obtenir à la fois une information sur le temps et sur la fréquence, mais la résolution d'analyse est fixée par le choix de la taille de l'enveloppe :

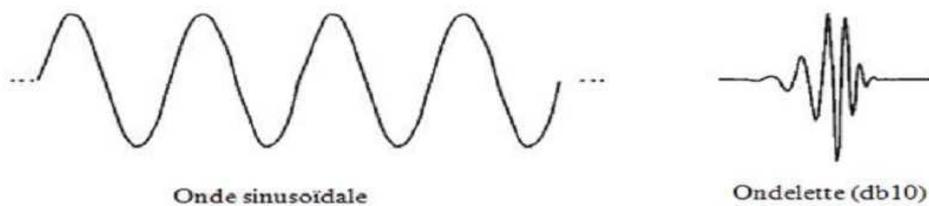
- Si la fenêtre est trop petite, les basses fréquences n'y seront pas contenues ;
- Si la fenêtre est trop grande, l'information sur les hautes fréquences est noyée dans l'information concernant la totalité de l'intervalle contenu dans la fenêtre.

Donc la taille fixe de la fenêtre est un gros inconvénient. L'outil idéal serait une fenêtre qui s'adapte aux variations de fréquence dans le signal à analyser. Cet outil existe, il s'agit de la récente analyse en ondelettes.

## II.3 La transformée en ondelettes (WT) :

### II.3.1 Définition :

Une ondelette est une forme d'onde qui possède une valeur moyenne nulle et une durée limitée. On dit que les ondelettes sont des ondes localisées. En regardant les figures des ondelettes et des ondes sinusoïdales, on voit intuitivement que des signaux avec les changements pointus pourraient mieux être analysés avec une ondelette irrégulière qu'avec une sinusoïde douce.



**Fig II.3** Différence entre une onde sinusoïdale et une ondelette

La transformée en ondelettes est un outil mathématique qui décompose un signal en fréquences en conservant une localisation spatiale. Le signal de départ est projeté sur un ensemble de fonctions de base qui varient en fréquence et en espace. Ces fonctions de base s'adaptent aux fréquences du signal à analyser. Cette transformation permet donc d'avoir une localisation en temps et en fréquence du signal analysé.



**Fig II.4** Transformée en ondelettes.

Deux opérations importantes appliquées à la fonction d'ondelette nous permettent de générer une infinité de variations de celle-ci.

- **La translation** qui correspond au déplacement de l'ondelette le long de l'axe des abscisses.
- **la dilatation** (aussi dite compression en temps ou espace).

Cette notion est illustrée par la forme générale d'une ondelette, appelée "ondelette mère" :

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi \left[ \frac{t-b}{a} \right]; (a,b) \in \mathbb{R}^2, a \neq 0 \quad (II.4)$$

$\frac{1}{\sqrt{a}}$  : facteur de normalisation peut être aussi pris à 1 ou  $\frac{1}{a}$ .

$a$  : facteur de dilatation « Echelle ».

$b$  : facteur de translation.

### II.3.2 Condition d'admissibilité :[8]

La condition nécessaire pour que  $\Psi$  puisse être utilisée comme une ondelette analysante est qu'elle vérifie une condition d'admissibilité :

$$\int_{-\infty}^{+\infty} (|TF(\Psi(w))|^2 / |w|) dw < \infty \quad (II.5)$$

L'équation peut s'écrire d'une façon équivalente sous la forme

$$\int_{-\infty}^{+\infty} \frac{|\hat{\Psi}(w)|^2}{|w|} dw < \infty$$

Cette condition permet d'analyser le signal, puis de le reconstruire sans perte d'information. La condition d'admissibilité implique en outre que la transformée de Fourier de l'ondelette (pour  $w = 0$ ) doit être nulle. Soit,

$$\int_{-\infty}^{+\infty} \Psi(t) dt = 0 \quad (II.6)$$

montre que  $\Psi(t)$  doit être à moyenne nulle.  $\Psi(t)$  est donc une fonction à largeur temporelle finie (fenêtre temporelle) possédant un caractère oscillatoire. On est donc bien en présence d'une **petite onde** appelée ondelette.

### II.3.3 La transformée en ondelettes continue (CWT):

La transformée en ondelettes continue utilise des translations et des dilatations de la fonction ondelette mère  $\Psi \in L^2(\mathbb{R})$  durant toute l'intervalle du temps de manière continue.

La transformée en ondelettes continue est définie par :

$$W(a, b) = \langle f, \Psi_{a,b} \rangle \quad (\text{II.7})$$

que l'on peut également noter :

$$W(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t) \Psi\left(\frac{t-b}{a}\right) dt ; (a, b) \in \mathbb{R}^2, a \neq 0$$

$$W(a, b) = \int_{-\infty}^{+\infty} f(t) \Psi_{a,b}(t) dt ; \quad \text{avec } \Psi_{a,b}(t) = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right)$$

Cette transformation est en théorie infiniment redondante puisque l'ondelette est traduite de manière continue. Cependant, il existe des méthodes pour diminuer cette redondance et une de ces méthodes consiste en l'emploi de la transformée en *ondelettes discrètes*.

### II.3.4 La transformée en ondelettes discrète (DWT) :

Nous avons vu que la transformée en ondelettes continue est infiniment redondante. La transformée en ondelettes discrète permet de diminuer fortement cette redondance sans l'annuler entièrement, elle translate et dilate l'ondelette selon des valeurs discrètes, On doit discrétiser les valeurs de  $a$  et  $b$ .

Daubechies a proposé de choisir le paramètre  $a$  comme une puissance entière et positive d'un paramètre de dilatation fixé  $a_0$  supérieur à 1,  $a = a_0^m$  avec  $m \in \mathbb{Z}$ .

Le choix du paramètre  $b$  est lié à celui de  $a$  par  $\mathbf{b} = \mathbf{nb_0 a_0^m}$  où  $(b_0 > 0) \in \mathbb{Z}$ , l'expression de l'ondelette devient alors :

$$\Psi_{m,n}(t) = a_0^{-\frac{m}{2}} \Psi(a_0^{-m}t - nb_0)$$

$m$  et  $n$  représentent les indices d'échelle et de temps (dilatation et translation).

La transformation en ondelettes discrète est donnée par :

$$W(a_0^m, nb_0) = C_{m,n} = a_0^{-\frac{m}{2}} \int_{-\infty}^{+\infty} f(t) \Psi_{m,n}(t) dt \quad (\text{II.8})$$

### II.3.5 L'analyse multi-résolution :

La multirésolution analyse le signal à haute fréquences, pour en prélever les détails, ensuite elle analyse le signal à une résolution deux fois moins fine, et elle réitère l'opération en grossissant son échelle d'un facteur de deux, jusqu'à obtenir une description complète du signal, elle est définie comme une suite décroissante de sous espaces emboîtés  $\dots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \dots$  qui satisfait les propriétés suivantes [8] :

- 1- Emboîtement des espaces :  $V_j$

$$V_{j+1} \subset V_j, \text{ Si } f(t) \in V_{j+1} \text{ alors } f(t) \in V_j$$

L'espace  $V_{j+1}$  contient des signaux plus *grossiers* que dans l'espace  $V_j$

- 2- Densité de l'espace engendré

$$\lim_{j \rightarrow -\infty} V_j = \overline{\bigcup_{j \in \mathbb{Z}} V_j} = L^2(\mathbb{R})$$

La projection de  $f(t)$  dans  $V_j$  converge vers  $f(t)$  quand  $j \rightarrow -\infty$

- 3- Racine de l'analyse nulle

$$\lim_{j \rightarrow +\infty} V_j = \bigcap_{j \in \mathbb{Z}} V_j = \{0\}$$

La projection de  $f(t)$  converge vers la fonction nulle quand  $j \rightarrow +\infty$

4- Invariance par dilatation

$$\forall j \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f\left(\frac{t}{2}\right) \in V_{j+1}$$

5- Invariance par translation

$$\forall k \in \mathbb{Z} \text{ Si } f(t) \in V_0 \Leftrightarrow f(t - k) \in V_0$$

Si  $f(t)$  est dans  $V_j$ , ses translatées sont dans  $V_j$

6- Existence d'une base

Il existe une fonction  $\phi(t)$  telle que la famille des fonctions  $\{\phi(t - k), k \in \mathbb{Z}\}$  forme une base orthonormée dans  $V_0$ .

### II.3.6 Fonction d'échelle et d'ondelette

La fonction d'échelle permet d'engendrer une base orthonormée de  $V_j$  par dilatation et translation :

$$\phi_{j,k} = 2^{-j/2} \phi(2^{-j}t - k); \text{ Avec } j, k \in \mathbb{Z} \quad (\text{II.9})$$

Les  $\phi_{j,k}$  constituent les fonctions de base de l'espace  $V_j$ . On peut projeter n'importe quelle fonction  $f$  de  $L^2(\mathbb{R})$  sur la base créée.

On pose :

$$A(j, k) = \langle f, \phi_{j,k} \rangle = \int_{-\infty}^{\infty} f(t) \cdot \phi_{j,k} dt \quad (\text{II.10})$$

L'approximation à la résolution  $j$  définie par la suite discrète des nombres  $A(j, k)$ .

L'espace des détails  $W_j$  est l'espace tel que

$$V_{j-1} = V_j \oplus W_j$$

contenant la différence d'information entre une approximation à la résolution  $j$  et une approximation à la résolution  $j - 1$ , le passage de l'espace  $V_{j-1}$  vers l'espace grossier  $V_j$  s'accompagne d'une perte d'information. Pour quantifier ces informations perdues, on a besoin d'une base de fonctions supplémentaire. Ces fonctions supplémentaires génèrent un espace  $W_j$  qui est le complémentaire de  $V_j$  dans  $V_{j-1}$ . Toutes les fonctions de  $V_{j-1}$  peuvent être décomposées sur l'union des deux bases  $V_j$  et  $W_j$ .

Les fonctions ondelettes, par dilatation et translation, engendrent une base orthonormée des  $W_j$ .

$$\psi_{j,k} = 2^{-j/2} \psi(2^{-j}t - k); j, k \in \mathbb{Z} \quad (\text{II.11})$$

Les  $\psi_{j,k}$  constituent les fonctions de base de l'espace  $W_j$

Les coefficients de détail sont définis par :

$$D(j, k) = \langle f, \psi_{j,k} \rangle = \int_{-\infty}^{\infty} f(t) \cdot \psi_{j,k} dt \quad (\text{II.12})$$

Les équations à deux échelles pour la fonction d'échelle et l'ondelette nous permettent d'écrire :

$$\phi(t) = \sqrt{2} \sum_k h[k] \cdot \phi(2t - k); \text{ avec } h[k] = \sqrt{2} \int \phi(t) \cdot \phi(2t - k) dt \quad (\text{II.13})$$

$$\psi(t) = \sqrt{2} \sum_k g[k] \cdot \phi(2t - k); \text{ avec } g[k] = \sqrt{2} \int \psi(t) \cdot \phi(2t - k) dt \quad (\text{II.14})$$

### II.3.7 L'algorithme de S.MALLAT :

L'algorithme de S.MALLAT est très utilisé pour la décomposition/reconstruction sur une base orthonormée d'ondelette. Il est appliqué sur les ondelettes issues d'une analyse multi-résolution qui a comme but de calculer les coefficients de détails  $D^k_j$  et les coefficients d'approximations, à partir des différentes sous-images d'approximations.

#### II.3.7.1 Analyse :

Nous avons vu que l'analyse multirésolution, nous permettait de décomposer un signal en coefficients d'approximations et de détails, ceci à partir d'une fonction d'échelle  $\Phi(t)$  et d'une ondelette  $\psi(t)$ . En fait chacune de ces fonctions est estimée au

moyen des séquences  $h(n)$  et  $g(n) = (-1)^n h(1-n)$  et il suffit de connaître les valeurs de la suite  $h(n)$  pour effectuer la décomposition en multirésolution. Une fonction de  $f \in V_{j-1}$  peut être exprimée comme la somme de la projection orthogonale de  $f$  dans  $V_j$  et de la projection orthogonale de  $f$  dans  $W_j$ . On aura alors à décomposer la fonction  $f$  ayant une échelle donnée, en une somme de coefficients d'échelle et de coefficients de détails à l'échelle grossière suivante.

$$f_{j-1}(t) = \sum_{k \in \mathbb{Z}} a_{j-1,k} \phi_{j-1,k}(t) = \sum_{k \in \mathbb{Z}} a_{j,k} \phi_{j,k}(t) + \sum_{k \in \mathbb{Z}} d_{j,k} \psi_{j,k}(t) \quad (\text{II. 15})$$

La première somme peut être vue comme une approximation de  $f$  à une échelle grossière, la seconde comme les détails enlevés de  $f$  pour voir l'approximation. Les coefficients  $a(j, k)$  et  $d(j, k)$  sont calculés comme suit :

$$\begin{aligned} a(j, k) &= \langle f, \phi_{j,k} \rangle = \int_{-\infty}^{\infty} f(t) \cdot \phi_{j,k}(t) dt \\ &= \int f(t) 2^{-j/2} \phi(2^{-j}t - k) dt \\ &= \int f(t) 2^{-j/2} [\sqrt{2} \sum_n h[n] \cdot \phi(2(2^{-j}t - k) - n)] dt \\ &= \sum_n h(n) \int f(t) 2^{\frac{j-1}{2}} \phi(2^{-(j-1)}t - (2k + n)) dt \\ &= \sum_n h(n - 2k) a(j-1, n) \end{aligned}$$

On voit bien que les coefficients d'approximation à la résolution  $j$ , s'obtiennent donc à partir des coefficients d'approximation à la résolution  $j-1$  d'une façon récursive selon :

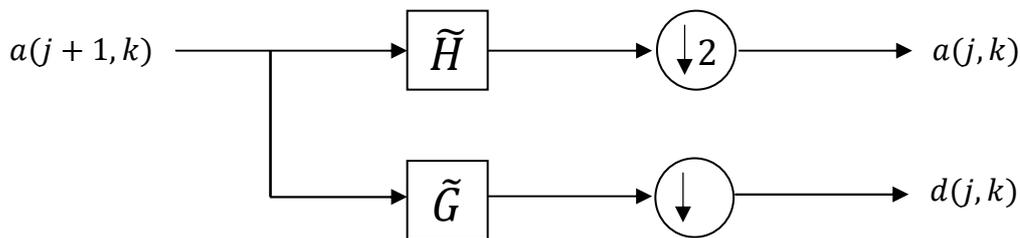
$$a(j, k) = \sum_n h(n - 2k) a(j-1, n) \quad (\text{II. 16})$$

De même pour les coefficients de détails on aura :

$$\begin{aligned} d(j, k) &= \langle f, \psi_{j,k} \rangle = \int_{-\infty}^{\infty} f(t) \cdot \psi_{j,k}(t) dt \\ d(j, k) &= \sum_n g(n - 2k) a(j-1, n) \quad (\text{II. 17}) \end{aligned}$$

Ils se calculent également à partir des coefficients d'approximation à l'échelle  $j - 1$  mais en utilisant le filtre  $g(n)$ .

Ces deux dernières relations montrent comment faire le passage d'une approximation d'un niveau  $j$  à un niveau  $j + 1$ . Leur calcul peut être interprété comme un produit de convolution entre  $\tilde{h}[k]$  et  $a(j + 1, k)$  (respectivement  $\tilde{g}[k]$  et  $a(j + 1, k)$  évalué pour un indice sur deux ; ou encore comme un filtrage de la séquence  $a(j + 1, k)$  par le filtre de réponse impulsionnelle  $\tilde{h}[k]$  (respectivement  $\tilde{g}[k]$ ) suivi par un sous échantillonnage de rapport 2. Ceci est illustré par le schéma suivant :



**Fig II.5** Algorithme d'analyse de Mallat

### II.3.7.2 Synthèse :

Nous avons vu au paragraphe précédent que l'approximation  $f_{(j-1)}(t)$  du signal  $f(t)$  à la résolution  $j - 1$  comme étant la somme de l'approximation de la projection de  $f(t)$  dans  $V_j$  à et de la projection de  $f(t)$  dans  $W_j$

$$\begin{aligned} f_{(j-1)}(t) &= \sum_{k \in \mathbb{Z}} a_{j-1,k} \phi_{j-1,k}(t) \\ &= \sum_{k \in \mathbb{Z}} a_{j,k} \phi_{j,k}(t) + \sum_{k \in \mathbb{Z}} d_{j,k} \psi_{j,k}(t) \end{aligned}$$

En utilisant le produit scalaire de cette équation avec  $\phi_{j-1,k}(t)$ , on obtient

$$a_{j-1,k} = \sum_{k \in \mathbb{Z}} a_{j,k} \langle \phi_{j,k}(t), \phi_{j-1,k}(t) \rangle + \sum_{k \in \mathbb{Z}} d_{j,k} \langle \psi_{j,k}(t), \phi_{j-1,k}(t) \rangle \quad (\text{II.18})$$

Le produit scalaire de l'équation précédente peut s'écrire

$$\langle \phi_{j,k}(t), \phi_{j-1,k}(t) \rangle = \int 2^{-j/2} \phi(2^{-j}t - k) 2^{-\frac{j-1}{2}} \phi(2^{-(j-1)}t - k)$$

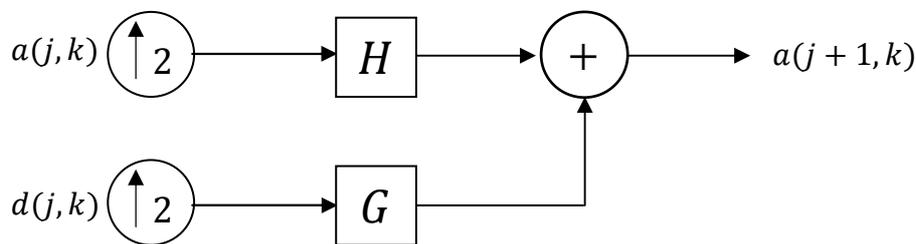
par changement de variable  $u = 2^{-j}t - k$

$$\begin{aligned}
 \langle \phi_{j,k}(t), \phi_{j-1,k}(t) \rangle &= \sqrt{2} \int \phi(u) \cdot \phi(2u + 2l - k) dt \\
 &= \sqrt{2} \int \phi(u) \cdot \phi(2u - (k - 2l)) dt \\
 &= h[k - 2l]
 \end{aligned}$$

Finalement la formule de reconstruction aura comme expression :

$$a_{j-1,k} = \sum_{l \in \mathbb{Z}} h[k - 2l] a_{j,l} + \sum_{l \in \mathbb{Z}} g[k - 2l] d_{j,l} \quad (\text{II.19})$$

La figure II.6 illustre l'algorithme de reconstruction



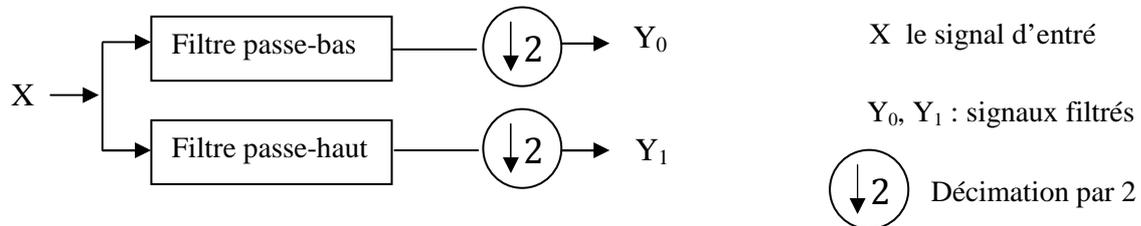
**Fig II.6** Algorithme de reconstruction de Mallat

### II.3.7.3 Les filtres de décomposition/reconstruction:

L'analyse s'effectue selon la résolution choisie. Les fonctions de transferts des filtres utilisés doivent avoir une distorsion d'amplitude la plus réduite possible. Cela permet de ne pas défavoriser d'une part, certaines fréquences par rapport à d'autres et d'autre part, les filtres à phase linéaire de ceux dont la phase ne l'est pas. Cette dernière caractéristique est d'autant plus importante que le signal que nous voulons analyser est un signal bidimensionnel représentant une image et que la perception visuelle chez l'homme est sensible aux fluctuations irrégulières de la phase.

De ce fait, les Filtres Miroirs en Quadrature (QMF) à réponse impulsionnelle infinie (RII) ont été retenus dans un premier temps pour avoir une reconstruction exacte, mais l'inconvénient résidait dans un temps de calcul important. La limitation des filtres RII pour obtenir des filtres RIF (réponse impulsionnelle finie) permettait de réduire les temps mais provoquait à leur tour des dégradations pendant la reconstruction. C'est pour résoudre ces

problèmes que les Filtres Conjugés en Quadrature (QCF) ont été développés. En effet, ces filtres présentent l'avantage d'être à reconstruction exacte.



**Fig II.7** Banc de filtres

### II.3.8 Ondelettes orthogonales

Soit  $V$  un espace vectoriel de dimension  $n$  et  $\{\phi_j / j = 1, \dots, n\}$  une base de cette espace. Cette base est qualifiée de Orthogonale si est seulement si

$$\langle \phi_j, \phi_i \rangle = 0 \forall i \neq j ; i, j = 1, \dots, n$$

Normée si est seulement si

$$\langle \phi_j, \phi_j \rangle = 1 \forall j = 1, \dots, n$$

### II.3.9 Ondelettes bi-orthogonales

Deux familles de fonctions duales sont considérées, l'une sera utilisée pour l'analyse et l'autre pour la reconstruction ou synthèse. Ces familles sont orthogonales entre elles mais ne sont pas orthogonales en elles-mêmes. Nous avons besoin de deux familles duales de sous-espaces emboîtés définies par les deux familles de fonctions d'échelle, chacune formant une base non orthogonale. Les propriétés à respecter sont l'indépendance linéaire et l'existence et l'unicité de la décomposition. Si on note  $V_j$  et  $\tilde{V}_j$  les deux familles de sous-espaces, on a :

$$V_j \subset V_{j-1} \text{ et } \tilde{V}_j \subset \tilde{V}_{j-1}$$

Les sous-espaces complémentaires sont définis différemment car ils sont complémentaires au sous-espace dual :

$$W_j \perp \tilde{V}_j \text{ et } \tilde{W}_j \perp V_j$$

Dans le cas bi-orthogonal, l'analyse est effectuée de la même manière que dans le cas orthogonale :

$$A(j, k) = \sum_n h(n - 2k) A(j - 1, n)$$

$$D(j, k) = \sum_n g(n - 2k) A(j - 1, n)$$

Par contre, la reconstruction est assurée par les filtres conjugués, soit :

$$A(j - 1, k) = \sum_{l \in \mathbb{Z}} \tilde{h}[n - 2k] A_{j,l} + \sum_{l \in \mathbb{Z}} \tilde{g}[n - 2k] D_{j,l} \quad (\text{II.20})$$

La relation imposée entre les filtres étant :

$$h(n) = (-1)^n h(1 - n) \text{ et } g(n) = (-1)^n \tilde{h}(1 - n)$$

### II.3.10 Passage à deux dimensions :

La transformée en ondelettes 2D découle directement de la transformée en ondelette 1D. La méthode la plus simple pour obtenir les fonctions d'ondelettes et d'échelle en deux dimensions est de multiplier deux fonctions à une dimension [6]. On obtient la fonction d'échelle 2D par multiplication de deux fonctions d'échelle 1D  $\phi(x, y) = \phi(x)\phi(y)$ . L'obtention des fonctions d'ondelettes est réalisée par multiplication de deux fonctions d'ondelettes 1D ou d'une fonction d'ondelettes et d'une fonction d'échelles. Ainsi une transformée en ondelette 2D possède trois fonctions d'ondelettes  $\psi(x, y)^H = \phi(x)\psi(y)$ , cette première fonction d'ondelettes permet de récupérer les détails horizontaux;  $\psi(x, y)^V = \psi(x)\phi(y)$ , pour les détails verticaux;  $\psi(x, y)^D = \psi(x)\psi(y)$ , pour les détails diagonaux.

La TOD 2D peut être implémentée en utilisant un banc de filtres, illustrée par les figures ci-dessous.



$\tilde{h}[n]$  Filtre passe-bas ;  $\tilde{g}[n]$  Filtre passe-haut, en décomposition

$h[n]$  Filtre passe-bas ;  $g[n]$  Filtre passe-bas, en reconstruction

$$\tilde{h}[n] = h[-n] \text{ et } \tilde{g}[n] = g[-n]$$

$h[L - 1 - n] = (-1)^n g[n]$  L est la longueur de filtre (en nombre de points).

Afin d'obtenir la décomposition désirée, il faut procéder à un balayage des lignes alterné avec un balayage des colonnes. En effet, le balayage des lignes conduit à décomposer l'image verticalement en des coefficients d'approximations et de détails. Après chaque balayage des lignes, le balayage des colonnes permet de décomposer l'image horizontalement en deux parties approximation et détails, on obtient alors bien la décomposition désirée.

L'observation de la figure II.10, permet d'illustrer les transformations subies par l'image Lena lors de l'application de TOD 2D.

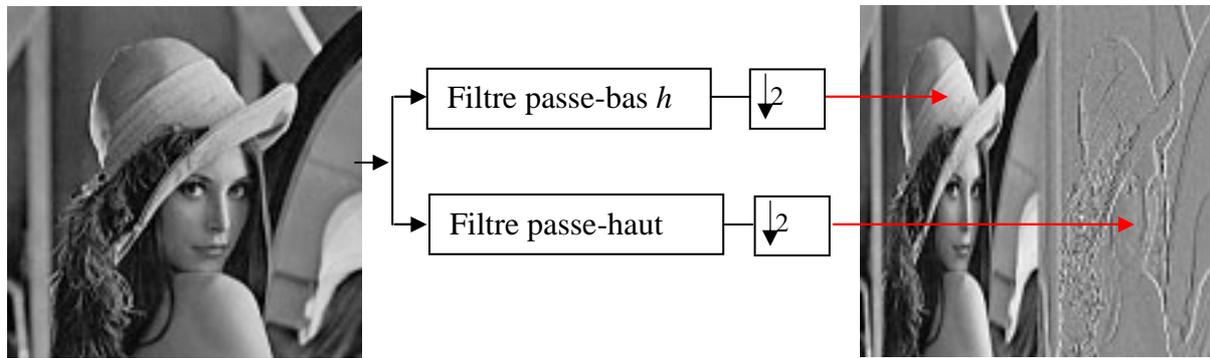
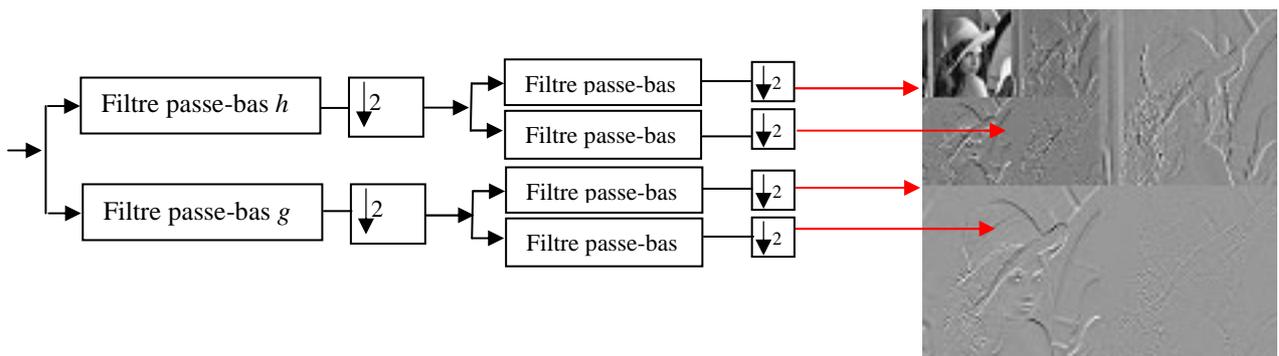
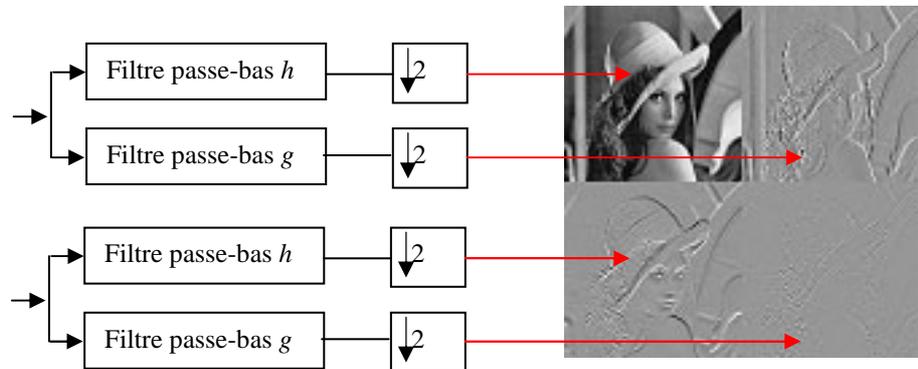


Image originale 'Lena'



**Fig II.10** Transformée en Ondelettes d'une image

### II.3.11 Propriétés des ondelettes :

La théorie des ondelettes permet d'associer aux fonctions d'échelle et d'ondelette un certain nombre de propriétés intéressantes pour l'analyse et la synthèse des signaux. Les principales propriétés de la fonction d'ondelettes sont[8] :

#### 1) Régularité :

Une ondelette est dite régulière d'ordre ' $n$ ' si elle est ' $n$ ' fois continument dérivable, et plusieurs passage à zéro ( $n$  nombre de moments nuls). Plus une ondelette est régulière, plus son spectre décroît rapidement à l'infinie. Si on prend le cas d'une ondelette peu régulière, la décroissance spectrale sera lente par conséquent, introduise des fréquences élevées dans le signal temporel ce qui est cohérent avec une certaine irrégularité. La régularité d'ondelette dépend du nombre de moments nuls.

#### 2) Support Compact:

Si la fonction d'échelle  $\Phi$  et l'ondelette  $\Psi$  sont à support compact, les filtres  $h$  et  $g$  sont à réponse impulsionnelle finie (FIR), et sont propices à une implantation informatique. Dans le cas contraire, une décroissance rapide des filtres permet tout de même de les approcher raisonnablement par des filtres FIR.

#### 3) Localisation :

La qualité de localisation en temps et en fréquence peut se mesurer par la borne d'incertitude du principe d'Heisenberg, qui assure une meilleure localisation. Lorsque cette incertitude est atteinte, une mauvaise localisation induit un étalement de l'énergie du signal autour d'un instant moyen et d'une fréquence moyenne pour une échelle donnée.

#### 4) Orthogonalité :

L'orthogonalité permet de minimiser la redondance, autorisant un codage efficace grâce aux faible nombre de coefficient. L'orthogonalité simplifie la reconstruction qui reste néanmoins possible même lorsque cette propriété est vérifiée.

#### 5) Symétrie :

Si les fonctions d'échelle et d'ondelette sont (anti-)symétriques, les filtres sont alors à phase linéaire. On observe une distorsion de la phase à la reconstruction lorsque cette propriété est absente. En fait, il n'existe pas d'ondelettes orthogonales qui soient à la fois symétriques et à

support compact. C'est pourquoi, les ondelettes biorthogonales ont été introduites et sont abondamment utilisées.

### **Conclusion :**

Nous avons présenté dans ce chapitre quelques notions importantes sur la théorie des ondelettes, et le principe de la multirésolution de manière la plus synthétique possible. En premier lieu nous avons exposé le principe de base de la théorie, puis nous avons établi le lien entre la transformation en ondelettes et l'analyse en multirésolution. Nous avons introduit la technique de bancs de filtres pour exposer la transformation en ondelettes rapide.

Bien évidemment, les applications de la transformée en ondelettes sont très vastes : elle est appliquée dans la compression, le débruitage, la détection de contours, la reconnaissance de visage, la détection de mouvements et d'une manière générale dans la plupart des applications reposant sur l'analyse d'images.

## Préambule :

Les ondelettes orthogonales ou bi-orthogonales permettent une représentation efficace des zones de régularités homogènes des images, que ce soit les singularités ponctuelles, les textures homogènes ou les zones uniformément régulières. Cependant, elles ne capturent pas les régularités géométriques des images le long des contours. Toute régularité implique une forme de redondance que l'on souhaite pouvoir exploiter. La géométrie apparaît dans ce travail comme une source de régularité plus qu'une source d'information.

L'exploitation des redondances est une des idées centrales de la construction des représentations creuses, c'est à dire des représentations permettant d'obtenir de meilleures approximations avec peu de coefficients. L'exploitation de nouvelles formes de régularité, les régularités géométriques, doit donc permettre une amélioration par rapport aux représentations existantes déjà.

Nous proposons ici la construction d'une représentation adaptée à l'exploitation de ces régularités géométriques. Cette représentation, dite représentation en bandelettes.

### III.1 Première génération de bandelettes :

La première génération de la transformée en bandelettes a été proposée par Le Pennec [11]. L'image est d'abord divisée en régions contenant chacune un flot géométrique parallèle. Ces régions sont régulières le long des lignes de flot. Autour d'un contour, le flot est parallèle aux tangentes de la courbe du contour.

#### III.1.1 Géométrie

Le lien entre les directions de régularité maximale et les contours est fort, puisque la tangente à un contour est une direction de régularité maximale. La notion de direction de régularité est cependant plus générale et plus facile à évaluer.

##### III.1.1.1 Contours, courbes et flots

Les techniques basées sur la vision recherchent les contours dans les images et les considèrent comme une source principale d'information. La géométrie de l'image est représentée par ses contours, c'est-à-dire par des courbes. Chaque courbe  $C$  peut être paramétrée par sa paramétrisation normale, la paramétrisation par l'abscisse curviligne  $s$  est :

$$C = \{c(s) = (c_1(s), c_2(s)), s \in \mathbb{R}, c_1: \mathbb{R} \mapsto [0,1] \text{ et } c_2: \mathbb{R} \mapsto [0,1]\} \quad (\text{III. 1})$$

Ou par une autre paramétrisation. Au-delà de l'information de position, la courbe donne une information de direction, sa tangente indique la direction qui ne la traverse pas et qui est donc une direction de régularité maximale.

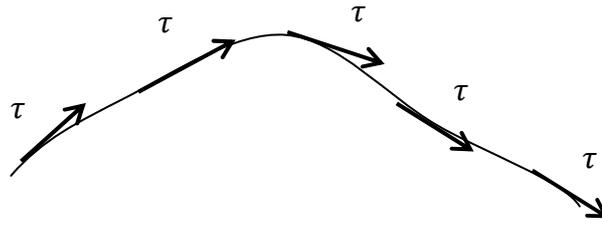


Fig III.1 : Contour  $C$  avec son flot  $\tau$  défini par sa tangente.

On parle de flot  $\tau$  le long du contour pour désigner le vecteur unitaire de cette direction de régularité maximale le long du contour

$$\tau(c(s)) = c'(s) = \begin{pmatrix} c'_1(s) \\ c'_2(s) \end{pmatrix} \quad (\text{III.2})$$

On peut étendre ce flot au voisinage du contour grâce à cette notion de régularité maximale ou il est beaucoup mieux défini que la position. En effet, pour tout contour la direction de régularité peut être mesurée tandis que la position ne peut être déterminée que dans le cas des contours visibles.

### III.1.2 Bandelettes

Les bandelettes sont obtenues à partir d'une déformation locale  $w$  de l'espace permettant d'aligner la direction de régularité avec une direction fixe et de se ramener à une base séparable.

#### III.1.2.1 Déformation de l'espace

On voudrait trouver une transformation  $w$  du plan telle que l'action de son gradient  $\nabla w$  sur la direction  $\tau(x)$  au point  $x$  donne une direction constante  $\tau_0$

$$(\nabla w)(\tau(x)) = \tau_0 \quad (\text{III.3})$$

La courbe  $C$  peut dans ce cas être paramétrée par  $c$  de sorte que

$$C = \{ (x_1, c(x_1)) \} \quad (\text{III.4})$$

et la transformation  $w$  est alors

$$\begin{aligned} w : \mathbb{R}^2 &\rightarrow \mathbb{R}^2 \\ (x_1, x_2) &\mapsto (x_1, x_2 - c(x_1)) \end{aligned} \quad (\text{III.5})$$

### III.1.2.2 Relation flot-courbe

Il existe une correspondance entre les flots constants le long de la direction verticale qui ne sont pas verticaux et les courbes dont la tangente n'est jamais verticale.

En effet, si l'on reprend la courbe  $C$  introduite précédemment et paramétrée par l'équation (III.4)

Le flot  $\tau$  associé à cette courbe est donné par :

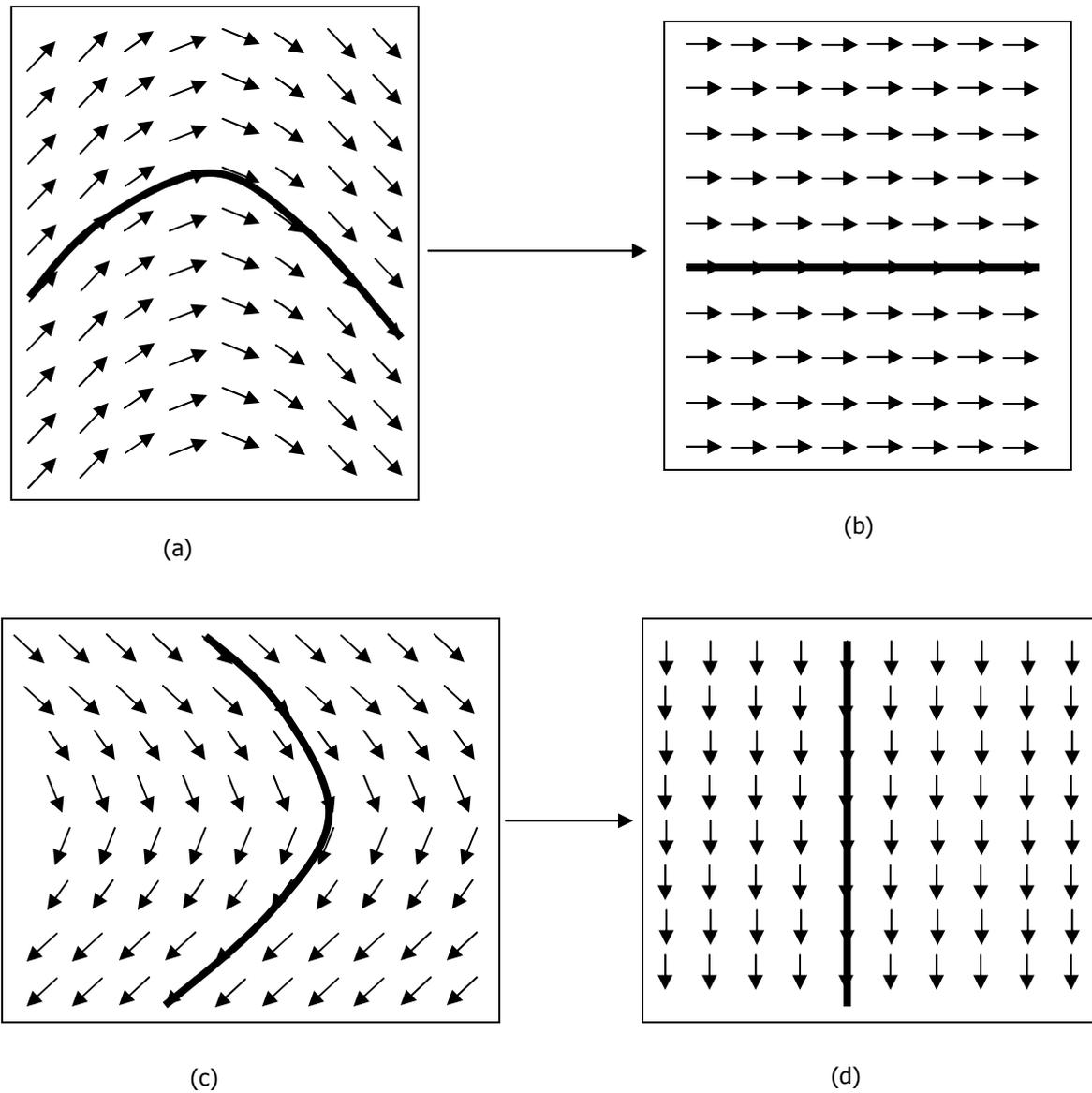
$$\tau(x) = \frac{1}{\sqrt{1+|c'(x_1)|^2}} \begin{pmatrix} 1 \\ c'(x_1) \end{pmatrix} \quad (\text{III.6})$$

Réciproquement pour tout flot  $\tau = \begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix}$  constant le long de la verticale qui n'est jamais vertical, on peut associer des courbes horizontales : les courbes obtenues en intégrant le flot. Ces courbes sont données par :

$$C = \{ (x_1, c(x_1)) \} \quad (\text{III.7})$$

avec :

$$c(x_1) = \int_{x_0}^{x_1} \frac{\tau_2(u)}{\tau_1(u)} du + c \quad (\text{III.8})$$



**Fig III.2** : Déformation horizontale et déformation verticale : **(a)** une courbe horizontale et son flot, **(b)** la courbe rectifiée et son flot strictement horizontal, **(c)** une courbe verticale et son flot, **(d)** la courbe rectifiée et son flot strictement vertical.

### III.1.2.3 Opérateur de déformations sur les fonctions

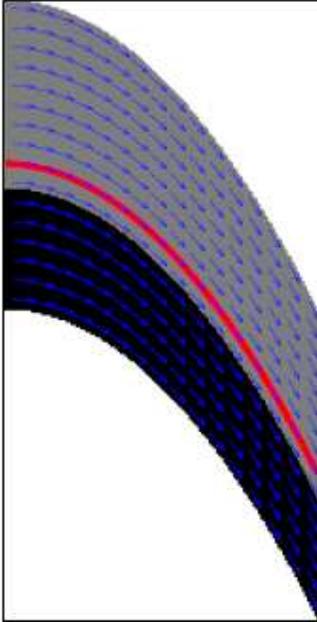
Pour le cas des flots constants selon la direction verticale qui ne sont jamais verticaux, on définit un opérateur de déformation  $W$  qui agit sur les fonctions à partir de la transformation  $w$  donnée en (III.5) de sorte que

$$Wf(w(x)) = f(x) \quad (III.9)$$

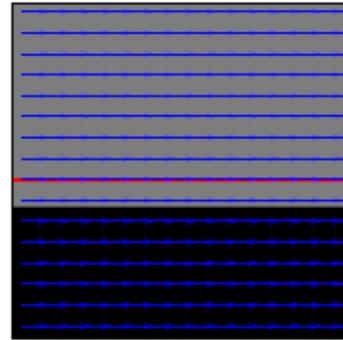
soit

$$Wf(x) = f(w^{-1}(x)) \quad (III.10)$$

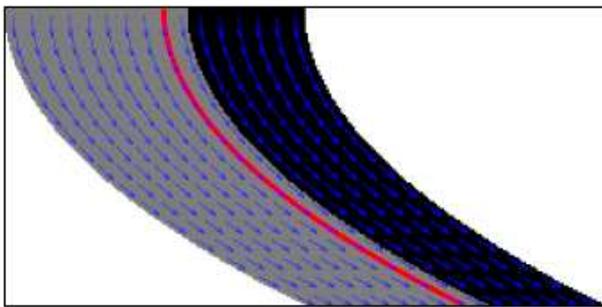
$$Wf(x) = f(x_1, x_2 + c(x_1)) \quad (III.11)$$



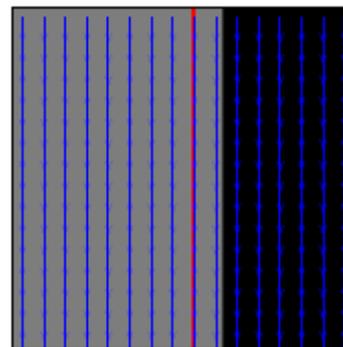
(a)



(b)



(c)



(d)

**Fig III.3** : Déformation des images : (a) une image ayant un flot horizontale, (b) son image par l'opérateur  $W$ , (c) une image ayant un flot vertical, (d) son image par l'opérateur  $W$ .

### III.1.2.4 définition de base de bandelettes :[11]

Maintenant on veut introduire des bases utilisant la régularité géométrique. Pour cela, on se ramène à la fonction  $Wf$  obtenue par l'équation (III.11) qui est régulière selon la direction horizontale et potentiellement irrégulière selon la direction verticale. On décompose alors celle-ci dans une base séparable  $B$

$$B = \{ g_{m_1}^1(x_1) g_{m_2}^2(x_2) \}, m_1 \in \mathbb{Z} \text{ et } m_2 \in \mathbb{Z} \quad (\text{III.12})$$

où les familles  $g_{m_1}^1, m_1 \in \mathbb{Z}$  et  $g_{m_2}^2, m_2 \in \mathbb{Z}$  sont deux bases monodimensionnelles. Ceci permet de traiter séparément les deux directions.

On calcule dans cette décomposition des produits scalaires

$$\langle Wf, g_{m_1}^1 \otimes g_{m_2}^2 \rangle \quad (\text{III.13})$$

On se ramène à des produits scalaires avec  $f$  à l'aide de la transposée de l'opérateur  $W$

$$\langle Wf, g_{m_1}^1 \otimes g_{m_2}^2 \rangle = \langle f, W^* g_{m_1}^1 \otimes g_{m_2}^2 \rangle \quad (\text{III.14})$$

Avec

$$W^* g_{m_1}^1 \otimes g_{m_2}^2(x_1, x_2) = g_{m_1}^1(x_1) g_{m_2}^2(x_2 - c(x_1)) \quad (\text{III.15})$$

Comme on le vérifie avec le calcul suivant :

$$\langle Wf, g_{m_1}^1(x_1) g_{m_2}^2(x_2) \rangle = \int_x Wf(x) g_{m_1}^1(x_1) g_{m_2}^2(x_2) dx \quad (\text{III.16})$$

Soit en insérant l'expression de  $Wf$

$$\int_x f(x_1, x_2 + c(x_1)) g_{m_1}^1(x_1) g_{m_2}^2(x_2) dx \quad (\text{III.17})$$

Puis par un changement de variable en  $x_2$

$$\int_x f(x_1, x_2) g_{m_1}^1(x_1) g_{m_2}^2(x_2 - c(x_1)) dx \quad (\text{III.18})$$

$$\langle Wf, g_{m_1}^1(x_1) g_{m_2}^2(x_2) \rangle = \langle f, g_{m_1}^1(x_1) g_{m_2}^2(x_2 - c(x_1)) \rangle \quad (\text{III.19})$$

Les produits scalaires

$$\langle f, W^* g_{m_1}^1 \otimes g_{m_2}^2 \rangle \quad (\text{III.20})$$

Correspondent à la décomposition de  $f$  dans une base orthonormée, celle définie par

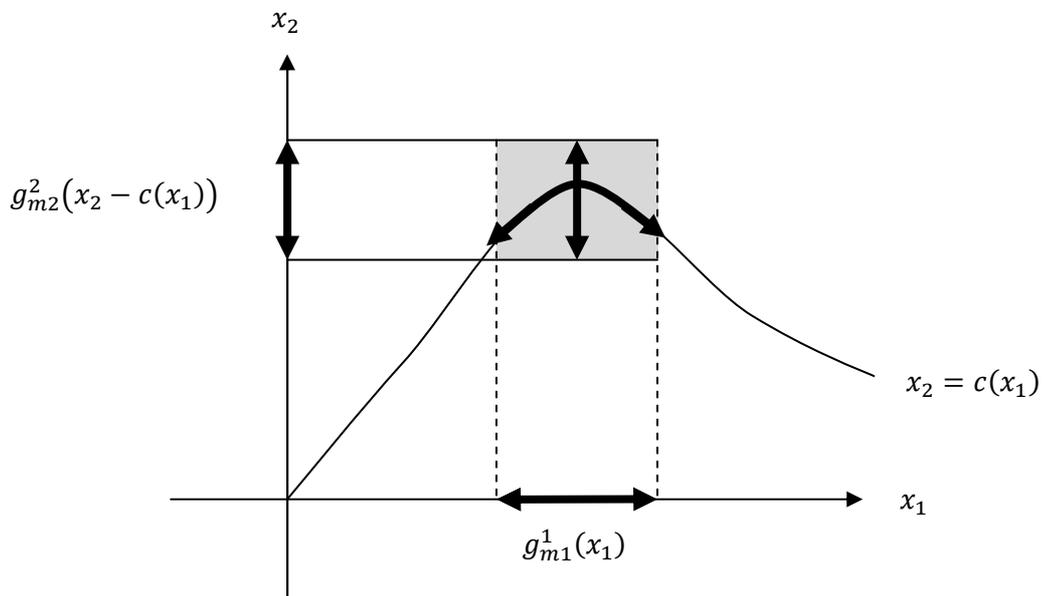
$$B_c = \{ W * g_{m_1}^1 \otimes g_{m_2}^2 \}, \quad m_1 \in \mathbb{Z} \text{ et } m_2 \in \mathbb{Z} \quad (\text{III.21})$$

$$B_c = \{ g_{m_1}^1(x_1) g_{m_2}^2(x_2 - c(x_1)) \}, \quad m_1 \in \mathbb{Z} \text{ et } m_2 \in \mathbb{Z} \quad (\text{III.22})$$

### ❖ Rôle et choix des deux bases

Le choix des bases  $\{ g_{m_1}^1 \}$  et  $\{ g_{m_2}^2 \}$  est crucial pour l'efficacité des bandelettes. Leurs rôles sont très différents bien que finalement on choisisse d'utiliser des ondelettes monodimensionnelles dans les deux cas.

La base  $\{ g_{m_1}^1 \}$  est alignée avec la direction de régularité et doit donc permettre d'exploiter ceci. La base  $\{ g_{m_2}^2 \}$  est alignée avec une direction dans laquelle la fonction peut présenter des singularités, elle doit donc permettre de représenter efficacement celles-ci.



**FigIII.4** : Une bandelette au voisinage d'un contour paramétré par  $x_2 = c(x_1)$

La bandelette  $g_{m_1}^1(x_1) g_{m_2}^2(x_2 - c(x_1))$  a un support (grise) qui suit le contour, l'analyse selon  $x_1$  suit le contour tandis que celle selon  $x_2$  le traverse.

La base  $\{ g_{m_1}^1 \}$  est utilisée pour capturer une régularité uniforme, La base  $\{ g_{m_2}^2 \}$  pour sa part, doit permettre de reproduire efficacement les singularités. les bandelettes sont issues d'un produit de deux ondelettes monodimensionnelles à support compact. Ces produits tensoriels correspondent aux ondelettes anisotropes. Si  $C$  est une courbe horizontale paramétrée par  $c$ ,  $w$  la transformation et  $W$  l'opérateur associés, la base de bandelettes  $B$  de l'espace tout entier est définie par le transport par  $W^*$ , l'opérateur transposé de l'opérateur  $W$ , d'une base d'ondelettes anisotropes

$$\mathbf{B}_c = \{W^*(\psi_{j_1,k_1} \otimes \psi_{j_2,k_2})\}_{(j_1,k_1) \in \mathbb{Z}^2, (j_2,k_2) \in \mathbb{Z}^2} \quad (\text{III. 23})$$

$$\mathbf{B}_c = \{\psi_{j_1,k_1}(x_1)\psi_{j_2,k_2}(x_2 - c(x_1))\}_{(j_1,k_1) \in \mathbb{Z}^2, (j_2,k_2) \in \mathbb{Z}^2} \quad (\text{III. 24})$$

### III.1.2.5 Bases de bandelettes [9]

Au lieu de représenter la géométrie des images par des contours, qui sont bien souvent mal définis, celle-ci est définie par des flots géométriques. Les vecteurs d'un flot géométrique indiquent des directions où l'image a localement des variations régulières. Les bases orthonormées de bandelettes sont construites en divisant l'image en régions où ces flots restent parallèles.

Les bandelettes sont construites à partir d'ondelettes bidimensionnelles déformées le long du flot géométrique. Dans une région  $\Omega$ , le flot géométrique est un champ de vecteur  $\vec{\tau}(x_1, x_2)$  qui donne une direction dans laquelle la fonction  $f$  est régulière dans un voisinage.

Afin d'exploiter la régularité, les ondelettes bidimensionnelles sont déformées pour suivre le flot. L'image déformée  $wf(x_1, x_2) = f(x_1, x_2 + c(x_1))$  est régulière le long des lignes horizontales. On utilise alors pour la région déformée

$$\Omega' = W\Omega = \{(x_1, x_2) : (x_1, x_2 + c(x_1)) \in \Omega\} \quad (\text{III. 25})$$

La base d'ondelettes bidimensionnelles de  $L^2(\Omega')$

$$\left\{ \begin{array}{l} \phi_{j,m_1}(x_1)\psi_{j,m_2}(x_2) \\ \psi_{j,m_1}(x_1)\phi_{j,m_2}(x_2) \\ \psi_{j,m_1}(x_1)\psi_{j,m_2}(x_2) \end{array} \right\} j, m_1, m_2 \in I_{\Omega'} \quad (\text{III. 26})$$

L'opérateur de déformation  $W$  étant orthogonal, l'application de son inverse à ces ondelettes donne une base orthonormée de  $L^2(\Omega)$  que l'on appelle base d'ondelettes déformées

$$\left\{ \begin{array}{l} \phi_{j,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1)) \\ \psi_{j,m_1}(x_1)\phi_{j,m_2}(x_2 - c(x_1)) \\ \psi_{j,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1)) \end{array} \right\} j, m_1, m_2 \in I_{\Omega'} \quad (\text{III. 27})$$

Afin d'exploiter la régularité de la fonction  $f$  selon le flot, on remplace les ondelettes déformées de la forme

$$\{\phi_{j,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1))\}_{m_1, m_2} \quad (\text{III. 28})$$

par la famille de fonctions engendrant le même espace

$$\{\psi_{l,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1))\}_{l > j, m_1, m_2} \quad (\text{III. 29})$$

Cette opération est appelée bandéletisation et on vérifie qu'elle s'implémente par une simple transformée en ondelettes monodimensionnelles discrètes.

Les fonction

$$\psi_{l,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1)) \quad (III.30)$$

sont appelées bandelettes du fait de leur support allongés selon les lignes de flots. La base orthonormée de bandelettes du domaine  $\Omega$  est donc définie par

$$\left\{ \begin{array}{l} \psi_{l,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1)) \\ \psi_{j,m_1}(x_1)\psi_{l,m_2}(x_2 - c(x_1)) \\ \psi_{j,m_1}(x_1)\psi_{j,m_2}(x_2 - c(x_1)) \end{array} \right\} \quad j, l > j, m_1, m_2 \quad (III.31)$$

## III.2 Bandelettes deuxième génération :

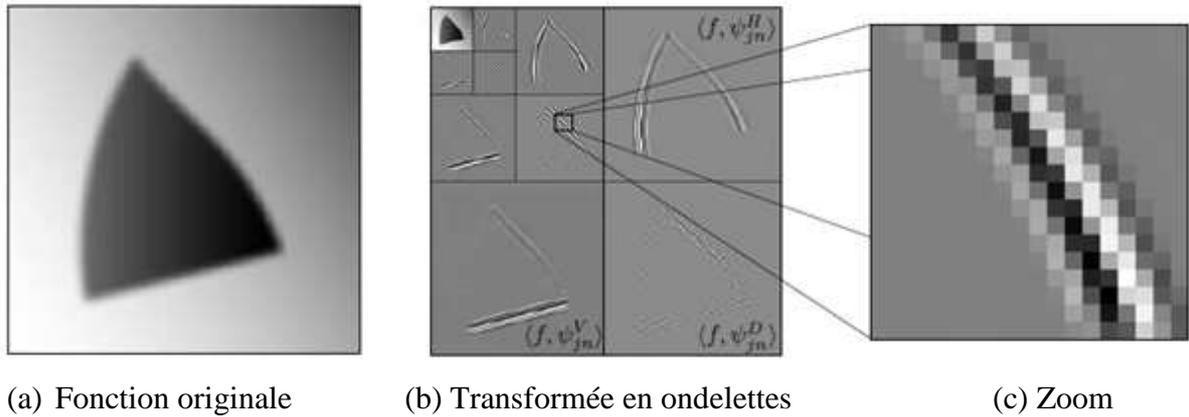
La construction des bandelettes a été raffinée par Peyré [10] pour obtenir des bases orthonormées adaptées aux fonctions géométriquement régulières. Ces bases de bandelettes de seconde génération sont définies à partir d'une représentation en ondelettes en ajoutant une étape d'une transformation géométrique sur les coefficients en ondelettes eux-mêmes. Cette transformée en bandelettes s'appuie sur une segmentation par quadtree des différentes bandes de fréquences de la transformée en ondelettes.

Les bandelettes sont obtenues par un changement de bases orthogonales orientes par cette géométrie sur les ondelettes correspondantes à chacun des carrés de la segmentation.

### III.2.1 Transformée en bandelettes sur un petit carré :

Soit  $f$  une fonction avec une régularité géométrique  $C^\alpha$ ,  $2^j$  une échelle fixée de la transformée en ondelettes, et  $k$  une direction donnée.

Une transformée en bandelettes est implémentée à l'aide d'un ensemble d'opérateurs orthogonaux locaux opérant sur les coefficients en ondelettes de  $f$ . Plus précisément, chaque opérateur agit sur un sous ensemble de coefficients  $f_s = \{f_j^k[n]\}_{(2^j[n]) \in S}$  ou  $S = S_1 \times S_2 \in [0,1]^2$  est un carré



**Fig III.5** : une image avec la régularité géométrique et ses coefficients en ondelettes

La figure III.5 montre un zoom sur les coefficients en ondelettes d'une fonction géométrique, près d'une courbe de singularité. Les opérateurs orthogonaux qui implémentent la nouvelle technique de bandeletisation sont capables d'exploiter la redondance directionnelle qui existe dans cet ensemble de coefficients.

### III.2.1.1 Paramétrisation de la géométrie

On construit une base de bandelettes orthogonales discrètes  $\mathcal{B}(S, \tilde{\gamma}_S)$  de  $l^2_j(S)$ , qui est paramétrée par une géométrie approchée  $\tilde{\gamma}_S$ . Pour d'écrire cette géométrie, il faut tout d'abord un paramètre  $\Omega$  qui code le type de la géométrie, qui peut être :

- une géométrie horizontale si  $\Omega = H$ . Dans ce cas, nous utilisons une courbe paramétrée par  $(x, \tilde{\gamma}^\Omega(x)) \in S$ . Ce type de géométrie est utile si un contour unique  $\gamma_i$  traverse  $S$  tout en restant relativement horizontal.
- une géométrie verticale si  $\Omega = V$ . Dans ce cas, nous utilisons une courbe paramétrée par  $(\tilde{\gamma}^\Omega(y), y) \in S$ .
- aucune géométrie n'est utilisée si  $\Omega = \Xi$ . Ceci est utile si le carré  $S$  ne contient aucune courbe de singularité, ou bien s'il contient un coin ou contours concourant.

Il est important de noter que chaque valeur de  $\tilde{\gamma}^\Omega$  donne naissance à une base de bandelettes, et que le calcul d'un jeu de paramétrés adaptés à une fonction donnée fait partie d'une stratégie globale de meilleure base décrite à la section III.2.1.2.

Lorsque  $\Omega = \Xi$ , nous gardons les coefficients en ondelettes  $f_S$  de la fonction d'origine, ce qui revient à définir  $\mathcal{B}(S, \tilde{\gamma})$  comme étant la base de Dirac discret de  $l^2_j(S)$ .

Lorsque  $\Omega \neq \Xi$ , nous utilisons une géométrie polynômiale quantifiée  $\tilde{\gamma}^\Omega$  pour définir notre base. Nous ne considérons qu'un nombre fini de géométries potentielles. La paramétrisation exacte de la géométrie est expliquée à la sous-section III.2.5.

**III.2.1.2 Approximation dans une meilleure base de bandelettes**

Pour comparer les différentes bases de bandelettes pour l'approximation d'une fonction  $f$  donnée, nous utilisons comme précédemment, l'entropie qui est aussi une fonctionnelle pénalisée. Soit  $\mathcal{B}(\Gamma_j^k) \stackrel{\text{def}}{=} \{\Psi_\mu\}_\mu$  une base de bandelettes pour une échelle  $2^j$  et orientation  $k$ . On définit, pour une fonction  $f \in l^2$

$$\varepsilon(f_j^k, \mathcal{B}(\Gamma_j^k), T) \stackrel{\text{def}}{=} \sum_{|\langle \Psi_\mu, X_j^k \rangle| < T} |\langle \Psi_\mu, f_j^k \rangle|^2 + T^2 M \tag{III.32}$$

Où

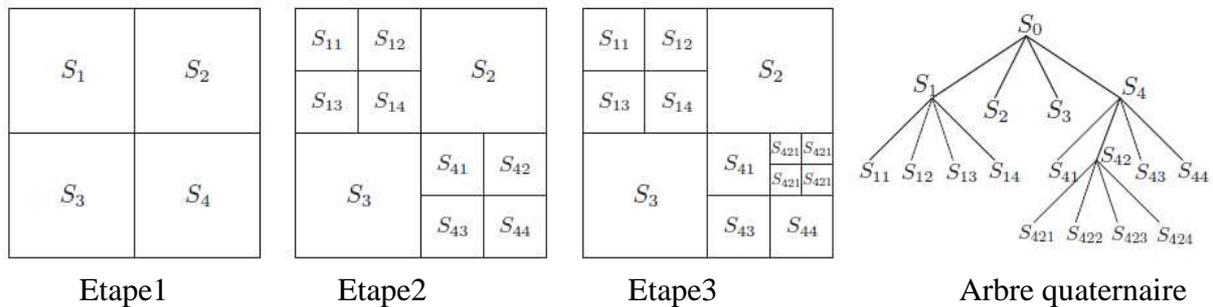
$$M \stackrel{\text{def}}{=} \text{Card} \{ \mu \mid |\langle \widetilde{\Psi}_\mu, f \rangle| \geq T \} + M_G + M_S \tag{III.33}$$

Où  $M_S+M_G$  est le nombre de paramètres nécessaires pour décrire la segmentation en quadtree et les géométries approchées qui composent les paramètres  $\Gamma_j^k$ .

La meilleure base de bandelettes minimisant cette entropie sous forme lagrangienne est alors capable de représenter de façon efficace les coefficients de  $f$  à l'échelle  $2^j$  et orientation  $k$ . L'algorithme effectif calculant cette meilleure base est décrit à la sous-section III.2.5.

**III.2.2 Segmentation en quadtree :**

Une segmentation en quadtree de  $[0, 1]^2$  est obtenue par subdivisions successives du carré initial en quatre carrés de tailles égales. La figure III.6 montre les étapes de subdivision menant à la construction d'un quadtree, ainsi que l'arbre quaternaire permettant de représenter cette subdivision.



**Fig III.6 :** Construction d'un quadtree par subdivisions successives

Ceci mène à une segmentation de  $[0,1]^2$  en des carrés de tailles dyadiques. On cherche une segmentation  $Q_j^k$  qui encapsule les courbes de singularité par un ensemble de carrés de taille  $b$  et qui remplissent les zones restantes par des grands carrés.

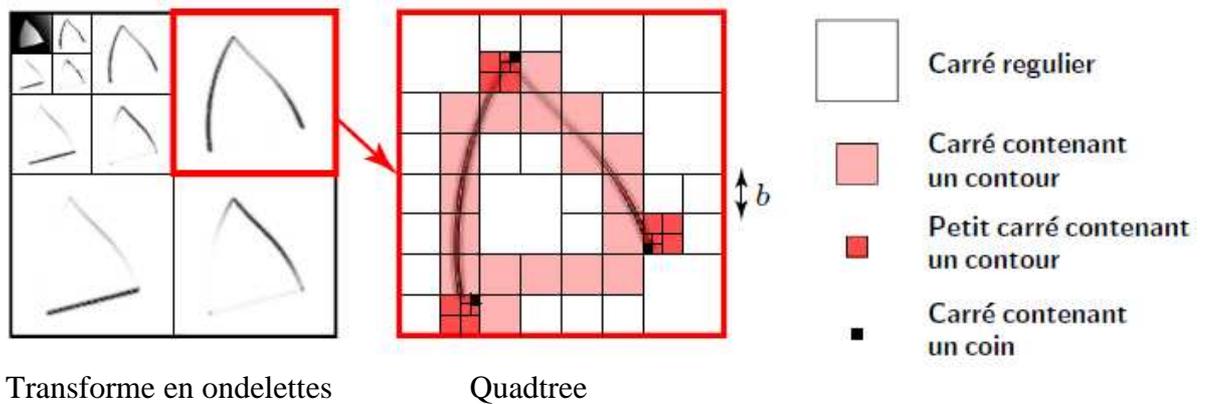
### III.2.3 Bases et dictionnaire de bandelettes

On construit une base discrète de bandelettes orthogonales  $\mathcal{B}(\Gamma_j^k)$  de  $l_j^2([0, 1]^2)$ , pour chaque échelle  $2^j$  et orientation  $k$ .

Les paramètres géométriques  $\Gamma_j^k \stackrel{\text{def}}{=} (Q_j^k, \{\tilde{\gamma}_S\})$  sont composés de :

- Une segmentation en quadtree  $Q_j^k$  dans  $[0, 1]^2$ . Un quadtree est obtenu en subdivisant de façon récursive le carré  $[0, 1]^2$  en quatre sous-carrés de taille égale. La figure III.7, montre un exemple de quadtree adaptée à une image donnée.
- Pour chaque carrée  $S \in Q_j^k$  correspond à une géométrie approchée  $\tilde{\gamma}_S$ . Une base de bandelettes  $\mathcal{B}(\Gamma_j^k)$  de l'espace entier de coefficients  $l_j^2([0,1]^2)$  est définie par :

$$\mathcal{B}(\Gamma_j^k) = \bigcup_{S \in Q_j^k} \mathcal{B}(S, \tilde{\gamma}_S) \quad (\text{III. 34})$$



**FigIII.7** : un exemple de quadtree adapté.

Un vecteur de bandelettes  $\Psi_\mu \in \mathcal{B}(\Gamma_j^k)$  est ainsi spécifié par  $\mu = (j, k, S, \tilde{\gamma}_S, l, i)$  où

- $2^j$  est l'échelle de la transformée en ondelettes 2D et  $k \in \{V, H, D\}$  est l'orientation.
- $S \in Q_j^k$  est un carré dyadique de largeur  $L = 2^j$ .
- $\tilde{\gamma}_S$  est une géométrie approchée.

- $l \in \{-2j \log_2 L, \dots, 0\}$  et  $i \in \{1, \dots, 2^{-l}\}$  sont l'échelle et l'index d'un vecteur  $\Psi_{ls}$  dans la base de bandelettes orthogonales  $\mathcal{B}(S, \tilde{\gamma}_s)$ .

Nous avons ainsi défini

$$\forall x_n \in Q_j^k, \Psi_\mu[n] = \Psi_{li}[n] \quad (\text{III. 35})$$

$$\text{où } \mathcal{B}(S, \tilde{\gamma}_s) = \{\Psi_{li}\}_{l,i} \quad (\text{III. 36})$$

Ces vecteurs discrets  $\Psi_\mu \in l^2([0,1]^2)$  peuvent être transformés en fonction  $\tilde{\Psi}_\mu \in L^2([0,1]^2)$  en définissant

$$\tilde{\Psi}_\mu = \Psi_\mu[n] \psi_{jn}^k \quad (\text{III. 37})$$

La figure III.8, montre plusieurs fonctions bandelettes. Ces fonctions sont aussi régulières que les fonctions d'ondelettes 2D utilisées. Bien que chaque quadtree segmente l'espace des coefficients en carrés disjoints, la reconstruction après une décomposition en bandelettes seuillée ne souffre pas d'effets de blocs. Ceci est dû au fait que la reconstruction par blocs des coefficients d'ondelettes est filtrée par la transformée en ondelettes inverse.

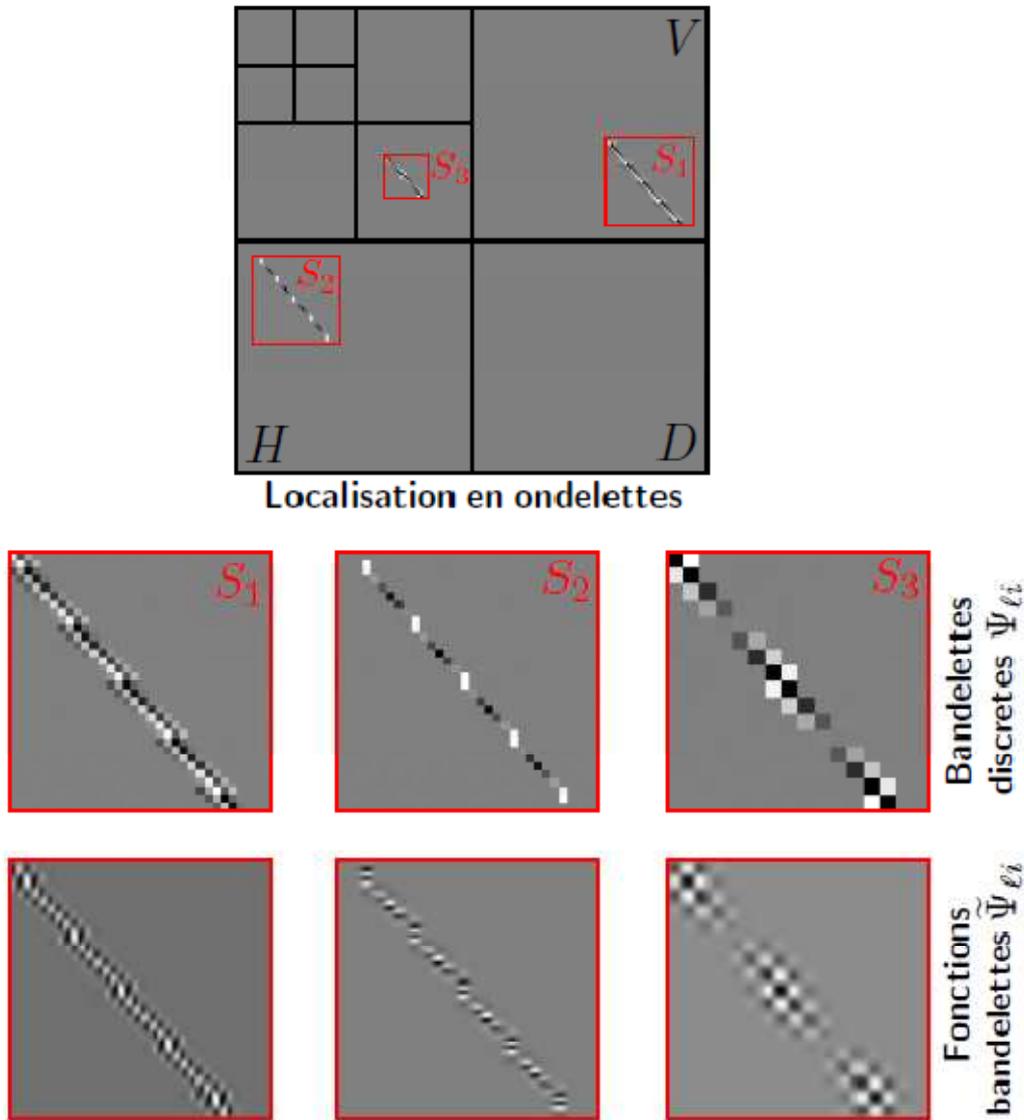


Fig III.8 : schéma des bases de bandelettes pour plusieurs échelles  $2^j$

III.2.4 Structure d'un dictionnaire de bandelettes :

❖ quantification de la géométrie :

Les coefficients en bandelettes inférieur au seuil  $T$  vont être rejetés pour l'approximation. Pour réduire le nombre de bases de bandelettes potentielles, on ne considère qu'un petit nombre de géométries quantifiées à l'intérieur de chaque carré  $S$  du quadtree. De façon plus précise, à partir d'une paramétrisation  $\tilde{\gamma}$  on définit une géométrie quantifiée  $\tilde{\gamma}_{T^2}$  avec une précision de  $T^2$ .

$$\tilde{\gamma}_{T^2} = \sum_{i=0}^{p-1} Q_{T^2}(\langle \gamma, \theta_i \rangle) \theta_i \tag{III.38}$$

Où  $\{ \theta_i \}$  est la base de polynômes de Legendre. Nous avons utilisé le quantificateur

$$Q_\tau(x) = q\tau \quad \text{Si} \quad \left(q - \frac{1}{2}\right)\tau \leq x \leq \left(q + \frac{1}{2}\right)\tau \quad (\text{III.39})$$

Pour un pas de quantification  $\tau = T^2$ , on obtient un nombre fini de géométries  $\mathcal{G}_{T^2}$

❖ **Construction du dictionnaire :**

Pour une direction fixée  $k$  et une échelle  $2^j$ , le dictionnaire  $\mathcal{D}_j^k$  de bandelettes discrètes est composé de bases  $\mathcal{B}(\Gamma_j^k)$  obtenues à l'aide de toutes les segmentations possibles en quadtree  $Q_j^k$  et toutes les géométries approchées  $\tilde{\gamma}_s$  à l'intérieur des carrés de la segmentation. En approximant les géométries  $\tilde{\gamma}_s$  à appartenir à  $\mathcal{G}_{T^2}(s)$  on définit un dictionnaire final  $\mathcal{D}_{j,T^2}^k$  de bases de bandelettes discrètes

$$\mathcal{D}_{j,T^2}^k = \{\mathcal{B}(\Gamma_j^k) \in \mathcal{D}_j^k \setminus \Gamma_j^k = (Q_j^k, \{\tilde{\gamma}_s\}) \quad \text{et} \quad \forall s \in Q_j^k, \tilde{\gamma}_s \in \mathcal{G}_{T^2}(s)\} \quad (\text{III.40})$$

L'union de toutes ces bases  $\{\mathcal{B}(\Gamma_j^k)\}_{k,j \leq 0}$  forme un dictionnaire de base de  $l^2([0,1]^2)$ .

### III.2.5 Transformée en bandelettes discrète :

❖ **Image discrétisée**

Une image discrétisée  $\bar{f}$  de  $N * N$  pixels est obtenue en projetant une fonction  $f \in L^2([0,1]^2)$  sur un ensemble de fonctions d'échelles  $\{\Phi_{jn}\}_n$  à une résolution  $2^J = N^{-1}$

$$\forall n \in \{0, \dots, N-1\}^2, \bar{f}[n] = \langle f, \Phi_{jn} \rangle$$

Où

$$\Phi_{jn}(t) = 2^{-J} \Phi(2^{-J}t - n) \quad (\text{III.41})$$

❖ **Fonction d'échelles en ondelettes**

Les espaces intermédiaires  $V_j \subset L^2([0,1]^2)$  peuvent se décomposer comme des sommes d'espace de détails en ondelettes

$$V_j = \bigoplus_{k,p < j} W_j^k \quad \text{et} \quad V_j = \text{vect} \{\varphi_{jn} \mid n \in \{0, \dots, 2^{-j} - 1\}^2\}$$

Les fonctions d'ondelettes et les fonctions d'échelles peuvent s'écrire à l'aide d'une cascade de filtrage passe-bas  $\{g^k[n]\}_n$  et passe-haut  $\{h^k[n]\}_n$ .

### ❖ bases de bandelettes discrètes

Nous avons décrit par l'équation (III.29) la manière dont on transforme des vecteurs de bandelettes discrets  $\psi_\mu \mathcal{B}(\Gamma_j^k)$  de  $l_j^2$  en un ensemble de fonctions  $\tilde{\psi}_\mu$ . Pour  $j > J$ , on peut aussi transformer ces vecteurs en des vecteurs de l'espace de pixels de l'image  $l_j^2$  comme suit

$$\bar{\mathcal{B}}(\Gamma_j^k) = \{\bar{\psi}_\mu\}_\mu \quad \text{Où} \quad \bar{\psi}_\mu = \sum_{(2^j n) \in [0,1]^2} \Psi_\mu[n] \bar{\psi}_{jn}^k \quad (\text{III.42})$$

Une base discrète de bandelettes est définie par

$$\bar{\mathcal{B}}(\Gamma) = \bigcup_{k,j>J} \bar{\mathcal{B}}(\Gamma_j^k) \quad \text{Où} \quad \Gamma = \bigcup_{k,j>J} \Gamma_j^k \quad (\text{III.43})$$

L'ensemble de toutes les bases possibles de bandelettes discrètes  $\bar{\mathcal{B}}(\Gamma)$  pour tous les ensembles de bases  $\bar{\mathcal{B}}(\Gamma_j^k) \in \mathcal{D}_{j,T^2}^k, j > J$  forme le dictionnaire de bandelettes discrètes  $\bar{\mathcal{D}}_{T^2}$ .

### ❖ Transformée en bandelettes discrètes

La transformée en bandelettes discrètes d'une image  $\bar{f}$  correspond à la projection de cette image sur les vecteurs d'une base  $\bar{\mathcal{B}}(\Gamma)$ . Cette transformée est calculée à l'aide tout d'abord d'une transformée en ondelettes discrète rapide de l'image  $\bar{f} \in l_j^2$  puis d'une transformée en bandelettes discrète de l'ensemble de coefficients d'ondelettes  $\bar{f}_j^k$  pour chaque échelle  $2^j > 2^J$  et orientation  $k$ .

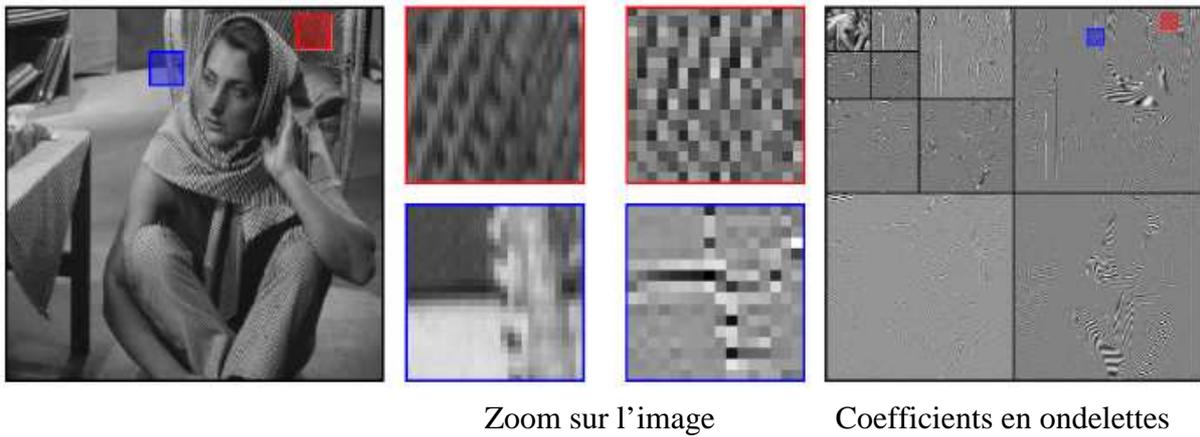
## III.3 Transformée en ondelettes par groupement :

### III.3.1 La géométrie de 16 coefficients d'ondelettes :

La géométrie des images naturelles est complexe. La topologie de l'ensemble des courbes est souvent compliquée avec de très nombreuses jonctions en T, croisements et enchevêtrements.

Dans une image, c'est rare de pouvoir exploiter une régularité géométrique sur une longueur de plus d'une dizaine de pixels. A l'échelle la plus fine de la transformée en ondelettes, ceci se traduit par des corrélations sur environ 4 coefficients d'ondelettes. Etant donné le coût du codage d'une géométrie.

Ces constatations ainsi que la pratique empirique de la transformée en bandelettes, nous ont amenés à ne pas utiliser de quadtree, en se restreignant à des carrés S de côté fixe, c'est à dire des paquets de 4\*4 coefficients d'ondelettes.



**Fig III.9** : Configurations complexes de coefficients d'ondelettes.

Après la transformée en ondelettes, les sous-bandes sont segmentées en blocs de 4\*4 coefficients, le fait de se restreindre à des groupes de telle taille comporte plusieurs avantages :

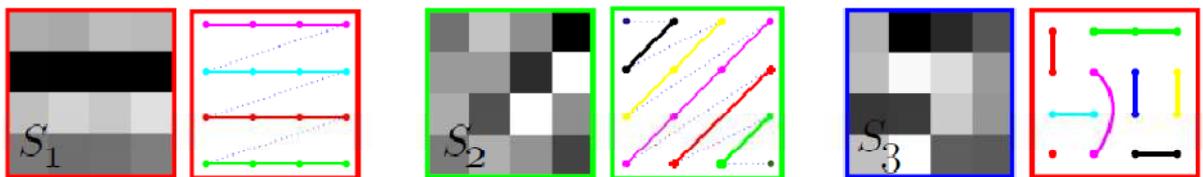
- Ils contiennent les corrélations typiques que l'on trouve dans les images naturelles.
- Le fait de ne pas utiliser de quadrees assouplie la procédure de codage.
- En se concentrant sur seulement 16 coefficients, on peut mieux comprendre la nature des redondances des coefficients d'ondelettes. On va ainsi construire une transformée en bandelettes plus adaptée.



(a) Image d'origine

(b) Transformée en ondelettes

(c) Zoom sur une orientation



(d) Groupement horizontal

(e) Groupement diagonal

(f) Groupement non géométrique

**Fig III.10** : Zoom sur des carrés de côté 4\*4 d'une image transformée par les ondelettes.

La figure III.10, illustre trois paquets typiques de 16 coefficients d'ondelettes. A chaque cas, on a effectué un regroupement adapté des coefficients.

- Le carré  $S_1$  correspond à une régularité purement horizontale. On a donc groupé les coefficients suivant les 4 lignes.
- Le carré  $S_2$  correspond à une régularité diagonale. On a donc groupé les coefficients selon les 7 lignes diagonales.
- Le carré  $S_3$  correspond à une géométrie chaotique (prés d'une jonction). Les groupements correspondant sont donc complexes.

Les groupements ont été effectués de façon manuelle pour garantir que les coefficients en ondelettes sont quasi-constants dans chaque groupe.

Plutôt que d'utiliser une transformée en bandelettes orthogonales classique, qui transforme tous les coefficients d'une seule fois, une meilleure stratégie consiste à transformer chaque groupe de coefficients de manière indépendante.

### III.3.2 Construction de bases de bandelettes par groupements directionnels :

Les coefficients de bandelettes étant les coefficients obtenus par projection des coefficients d'ondelettes de chaque bloc dans une base polynomiale, la reconstruction des coefficients d'ondelettes à partir des coefficients de bandelettes quantifiés est une approximation polynomiale des coefficients d'ondelettes originaux. De plus, cette approximation polynomiale est directionnelle.

Douze directions discrètes sont définies sur une grille de 4\*4 pixels dans le tableau de la figure III.11. Pour chacune de ces directions, les pixels voisins dans la direction considérée sont liés et définissent un groupe. Les douze configurations de groupement suivant ces directions et telles que définies par Peyré dans [10] sont représentées sur la figure III.2. Si l'on suit strictement les directions définies dans le schéma de cette figure, certains pixels de la grille 4\*4 se retrouvent isolés. Dans [10], il a été choisi de regrouper les pixels isolés suivant la direction horizontale ou verticale la plus proche de la direction des autres groupes comme le montre la figure suivante.

Configuration	#1	#2	#3	#4	#5	#6
Angle (rad)	$\text{atan}(1/3)$	$\text{atan}(1/2)$	$\pi/4$	$\text{atan}(2)$	$\text{atan}(2)$	$\pi/2$
Angle (degré)	18.4	26.6	45	63.4	71.6	90

Configuration	#7	#8	#9	#10	#11	#12
Angle (rad)	$-\text{atan}(3)$	$-\text{atan}(2)$	$-\pi/4$	$-\text{atan}(1/2)$	$-\text{atan}(1/3)$	0
Angle (degré)	-71.6	-36.4	-45	-26.6	-18.4	0

**Fig III.11** : Angles correspondant aux différentes directions des groupes

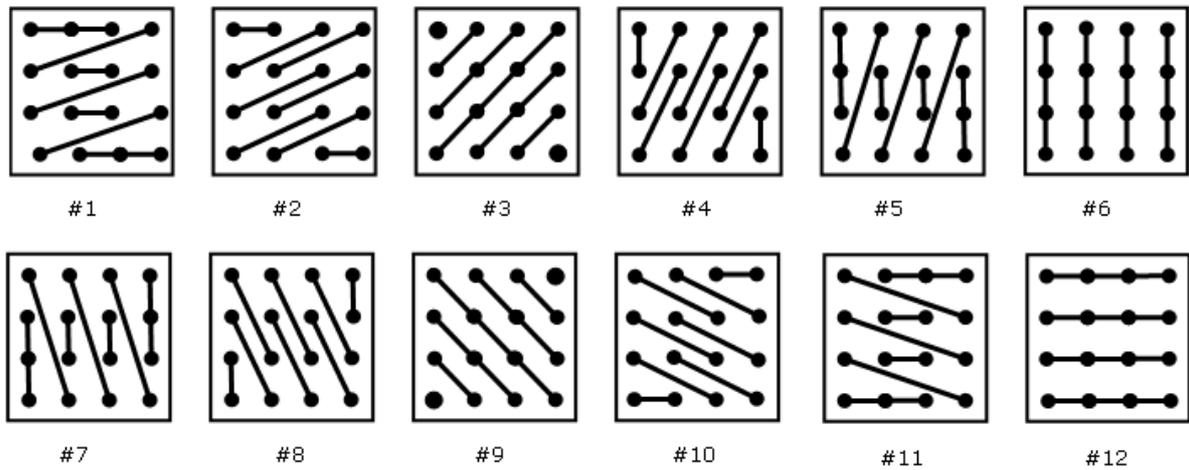


Fig III.12 : Douze configuration de groupes directionnels.

## Conclusion

Nous avons présenté dans ce chapitre la première génération de la transformée en bandelettes. Ces bandelettes obtenues à partir de déformations d'ondelettes anisotropes permettant via une rectification géométrique d'exploiter la régularité de la fonction selon sa direction de régularité maximale. Ensuite, nous avons élaboré la seconde génération de bandelettes développée par Peyré[10] qui ne suit pas le même raisonnement. Cette nouvelle transformée en bandelettes est construite au-dessus d'une transformée en ondelettes 2D. L'objectif de ces bandelettes est d'exploiter les régularités géométriques le long des contours dans les sous-bandes de la transformée en ondelettes. Enfin, nous avons détaillé la transformée en bandelettes par groupement qui consiste à segmenter les sous-bandes de la transformée en ondelettes en blocs de  $4 \times 4$  coefficients, ce qui permet d'assouplir l'algorithme de calcul et facilite la transmission dans le cas de la compression.

## Préambule

L'information contenue dans une image n'est pas régulière, cela est dû aux différentes courbes géométriques qui ne suivent pas un modèle mathématique. Cependant, le traitement de ces images nécessite des modèles plus adaptatives pour applications plus affinées.

Dans ce chapitre, nous présentons l'apport de l'application des bandelettes par rapport aux ondelettes dans les applications en traitement d'images trouvée dans la littérature ; à savoir : le débruitage, l'analyse de textures et la compression d'images.

### IV. 1. Application au débruitage :

#### IV.1.1 Débruitage par seuillage :[11]

Les estimateurs par seuillage introduit par Donoho et Johnstone forment une famille d'estimateurs simples et performants pour le débruitage d'un bruit blanc. Ils sont définis à partir d'une base orthonormée  $\{g_m\}_{m \in \mathbb{Z}}$  par un opérateur diagonal dans cette base. On détermine la décomposition dans la base des données bruitées  $\check{f} = f + W$ , où  $W$  est un bruit blanc de variance  $\sigma^2$ .

$$\check{f} = \sum_m \langle \check{f}, g_m \rangle g_m \quad IV.1$$

$$\sum_m (\langle f, g_m \rangle + \langle W, g_m \rangle) g_m \quad IV.2$$

$$\check{f} = \sum_m \langle f, g_m \rangle g_m + \sum_m \langle W, g_m \rangle g_m \quad IV.3$$

Avec :

$f$  : est une fonction.

$g_m$  : base orthonormée.

Et on estime  $f$  par  $\check{f}$  obtenu en modifiant les coefficients par des opérateurs de seuillage  $\rho(T)$

$$\check{f} = \sum_m \rho_T(\langle \check{f}, g_m \rangle), g_m \quad IV.4$$

L'opérateur de seuillage peut-être dur

$$\rho_T(x) = \begin{cases} x & \text{si } |x| > T_j \\ 0 & \text{si } |x| \leq T_j \end{cases} \quad IV.5$$

Ou doux

$$\rho_T(x) = \begin{cases} x - T & \text{si } x \geq T \\ x + T & \text{si } x \leq -T \\ 0 & \text{si } |x| \leq T \end{cases} \quad IV.6$$

Le seuil  $T$  doit être choisi proportionnel à l'écart type du bruit blanc pour obtenir des résultats acceptables. En pratique, on choisit souvent  $T = 3\sigma$ .

### IV.1.2 Résultats numériques :

Les figures concluant dans cette partie présente les résultats de l'application d'un algorithme de débruitage par seuillage dur dans un frame de bandelettes invariant par translation [11]. Ces résultats sont comparés avec un algorithme similaire dans un frame d'ondelettes isotropes invariant par translation. Ceci permet en effet de visualiser l'effet de l'introduction de la géométrie.

Les expériences ont été réalisées sur l'image Lena, avec deux niveaux de bruits,  $\sigma = 25$  et  $\sigma = 40$ . Pour chacune de ces expériences, on présente l'image originale, l'image bruitée et les deux estimations, celle obtenue à l'aide des bandelettes et celle obtenue à l'aide des ondelettes. On effectue alors un agrandissement sur une région intéressante pour souligner les différences entre les deux méthodes.

De manière générale, le débruitage en bandelettes permet d'obtenir un gain en PSNR par rapport au débruitage en ondelettes. Bien que ce gain soit faible, les images obtenues avec les bandelettes semblent visuellement plus fines que les images obtenues avec les ondelettes. La différence entre les deux reconstructions est localisée aux voisinages des contours et ceux-ci, bien que correspondant à une faible surface et donc modifiant peu le PSNR, ont une grande importance dans la perception des images. La diminution des effets oscillants aux voisinages des contours obtenue grâce à l'introduction de la géométrie qui augmente bien la qualité visuelle des estimations.



(a)



(b)

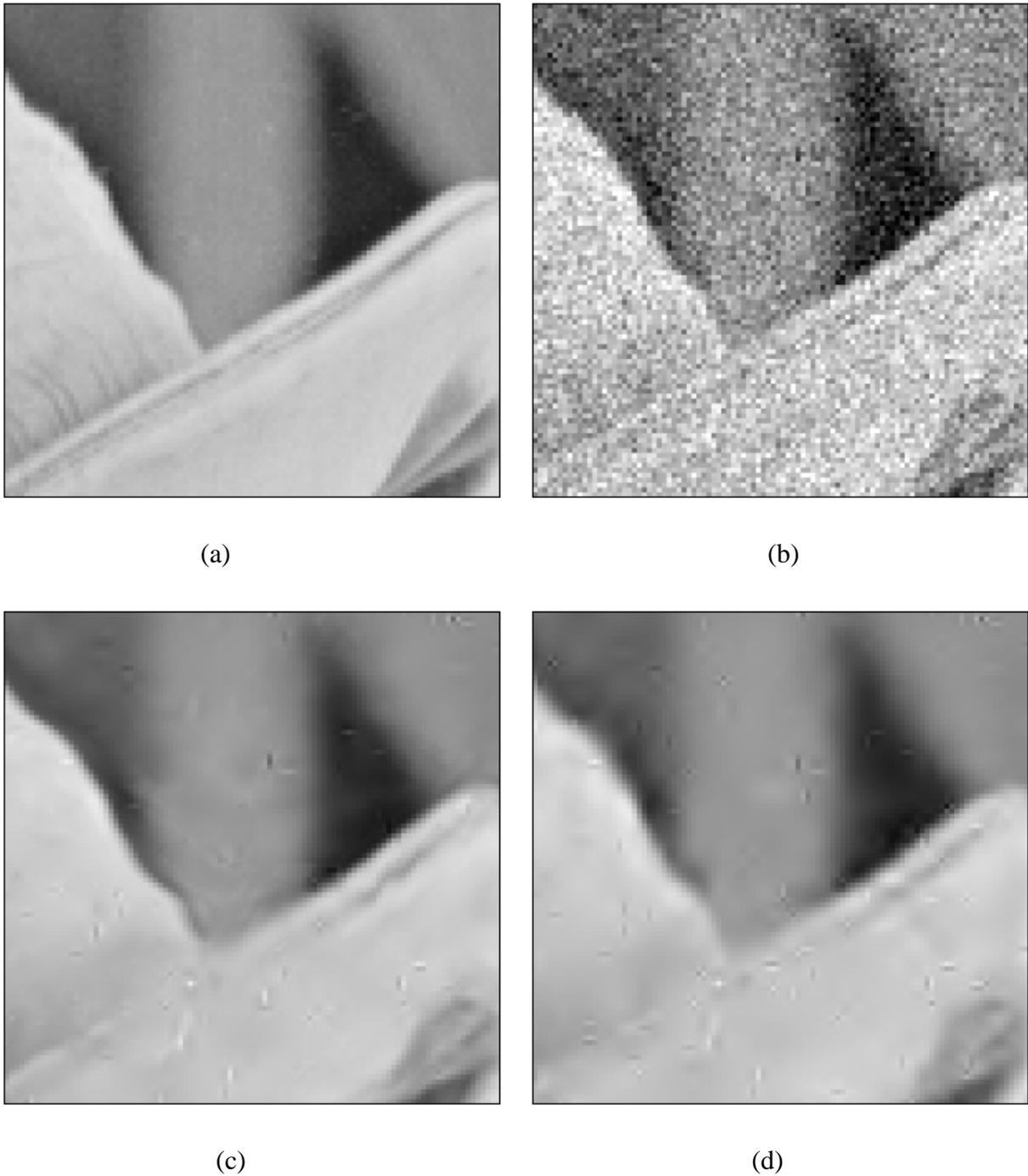


(c)



(d)

**Fig IV.1** : Débruitage de Lena avec  $\sigma=25$  : (a) image originale, (b) image bruitée (PSNR = 20.2 db), (c) image débruitée avec les bandelettes (PSNR = 30.2 db), (d) image débruitée avec les ondelettes (PSNR = 29.7 db).



**Fig IV.2 :** Débruitage de Lena avec  $\sigma = 25$  (agrandissement) : (a) agrandissement de l'image originale, (b) agrandissement de l'image bruitée (PSNR = 20.2 db), (c) agrandissement de l'image débruitée avec les bandelettes (PSNR = 30.2 db), (d) agrandissement de l'image débruitée avec les ondelettes (PSNR = 29.7 db).



(a)



(b)



(c)



(d)

**Fig IV.3** : Débruitage de Lena avec  $\sigma = 40$  : (a) image originale, (b) image bruitée (PSNR = 16.1 db), (c) image débruitée avec les bandelettes (PSNR = 27.0 db), (d) image débruitée avec les ondelettes (PSNR = 26.2 db).



(a)



(b)



(c)



(d)

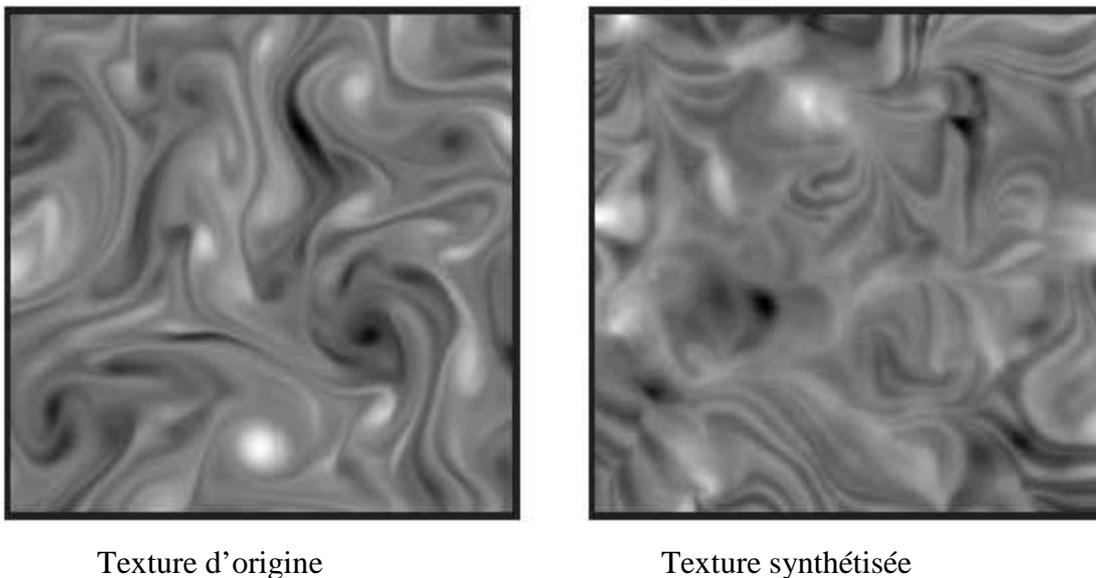
**Fig IV.4:** Débruitage de Lena avec  $\sigma = 40$  (agrandissement) : (a) agrandissement de l'image originale, (b) agrandissement de l'image bruitée (PSNR = 16.1 db), (c) agrandissement de l'image débruitée avec les bandelettes (PSNR = 27.0 db), (d) agrandissement de l'image débruitée avec les ondelettes (PSNR = 26.2 db).

## IV.2 Applications dans le domaine de la texture :

Dans la littérature, Peyre [10] a utilisé une nouvelle classe de transformation en bandelettes développée par Stéphane Mallat au sein de la société Let It Wave et qui a été déposée dans un brevet . Cette transformée utilise un champ d'association géométrique multi-échelles qui est calculé à l'aide d'un critère d'appariement de voisinages. Ce champ multi-échelles permet d'exploiter des corrélations à des distances de plus en plus longues.

Pour l'analyse de textures, on utilise une instance particulière de cette transformée qui permet d'avoir une quasi-invariance par translation et une grande liberté dans le choix des directions d'appariement.

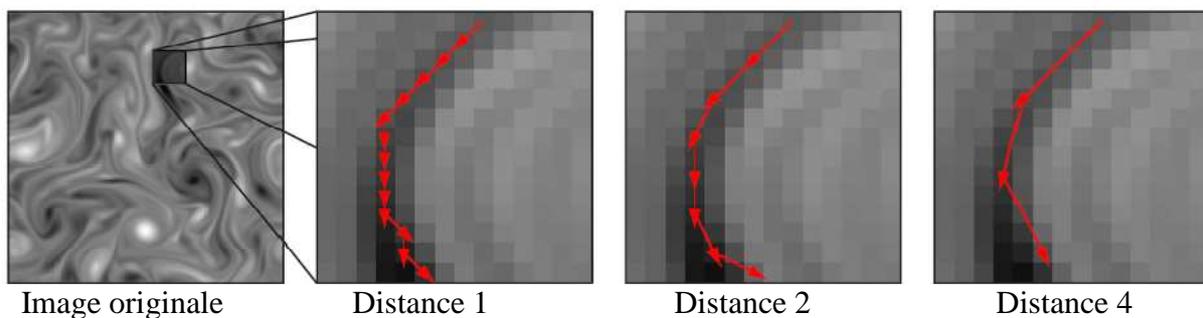
L'efficacité de cette transformée validée à l'aide d'une application à la synthèse de textures qui consiste à créer une nouvelle texture ressemblant à un exemple donné. La classe des méthodes retenue modélise les textures intéressantes à l'aide des histogrammes de coefficients transformés. L'enjeu est alors de modéliser avec des histogrammes relativement creux à la fois les coefficients en bandelettes et la géométrie qui paramétrise cette transformée. Ainsi, une synthèse réussie nécessite une réelle modélisation de la géométrie de la texture. À la figure IV.5, on peut voir un exemple de texture géométrique synthétisée à l'aide de cette méthode.



**Fig IV.5:** Synthèse de texture avec synthèse de la géométrie et des coefficients de bandelettes.

### IV.2.1. Champ d'association multi-échelles

La transformée en bandelettes par groupement est un algorithme qui a été développé par Stéphane Mallat. On utilise un cas particulier de cet algorithme qui est adapté au problème d'analyse de texture. L'idée principale est d'exploiter la redondance géométrique qui s'exerce sur de longues distances en définissant un champ d'association multi-échelles. Ce champ est utilisé pour réaliser des groupements entre des couples de points de plus en plus distants dans l'image. La figure IV.6, montre une partie d'un tel champ le long d'une singularité géométrique turbulente. Les vecteurs représentent quelques exemples de couples de points que l'on apparie le long de la géométrie. Ce champ se construit petit à petit depuis les courtes distances (échelle fine) vers les distances plus longues (échelles grossières).



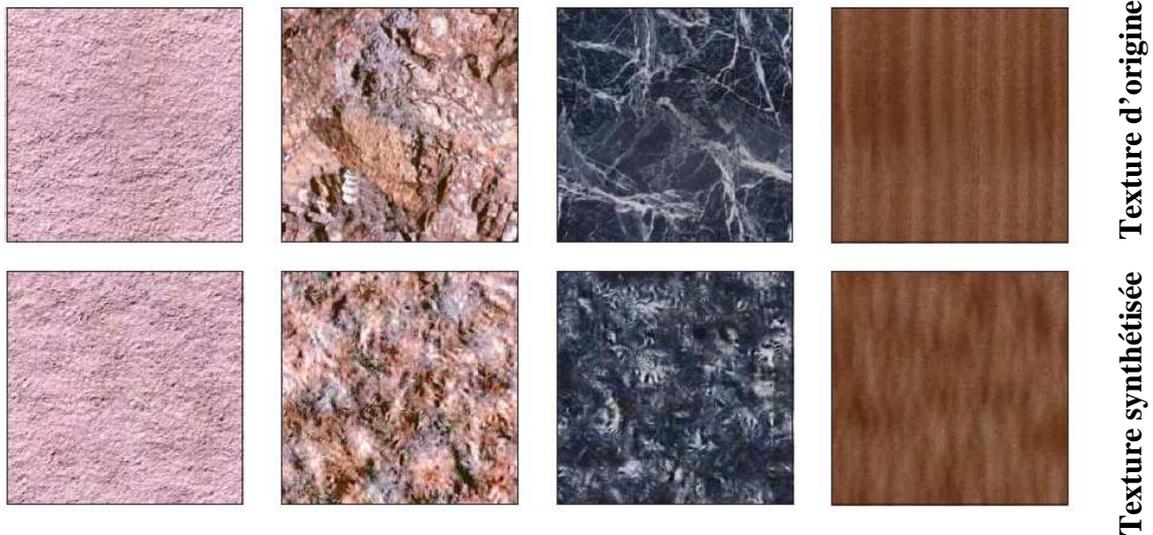
**Fig IV.6:** Illustration du principe d'un champ d'association multi-échelles.

On peut rapprocher cette construction d'un procédé de quantification de la géométrie. Ainsi plus la géométrie doit relier des points éloignés, plus elle doit être quantifiée de façon précise. Cette méthode permet d'assurer un suivi de la singularité sur une longue distance.

### IV.2.2. Synthèse par égalisation d'histogrammes :

Heeger et Bergen utilisent l'égalisation des histogrammes des coefficients d'ondelettes pour la synthèse non paramétrique. Cette méthode donne de bons résultats pour les textures homogènes et fractales. L'idée est d'utiliser une réalisation d'un processus aléatoire dont les coefficients multi-échelles sont indépendants et suivent une loi fixée à chaque échelle, qui est estimée à partir de la texture d'origine.

La figure IV.7 montre des exemples de succès de cette méthode mais aussi les problèmes qui ont lieu lorsque l'on s'intéresse à des images avec des structures géométriques.



**Fig IV.7 :** Exemples de synthèse de texture par égalisation d'histogrammes. Textures fractales (gauches) et textures avec des structures géométriques (droite).

- ❖ pour réaliser une texture synthétisée  $B$ , on peut utiliser la procédure d'égalisation d'histogrammes présentée dans l'algorithme 1.

**Algorithme 1 : la synthèse de texture par égalisation d'histogrammes**

Fonction  $B = \text{synthèse\_égalité}(A, B)$ .

Entrée : texture  $A$ .

Sortie : texture synthétisée  $B$ .

Initialisation :  $B = \text{bruit\_blanc}()$ .

Calcul de la transformée de l'image d'origine :

$[A_0, A_1, \dots, A_m] = \text{transforme}(A)$ .

Répéter jusqu'à convergence :

Égalisation spatiale :  $B = \text{égalité}(A, B)$ .

Passage en domaine transformé :  $[B_0, B_1, \dots, B_m] = \text{transforme}(B)$ .

Pour chaque  $j = 0, \dots, m$  :  $B_j = \text{égalité}(A_j, B_j)$ .

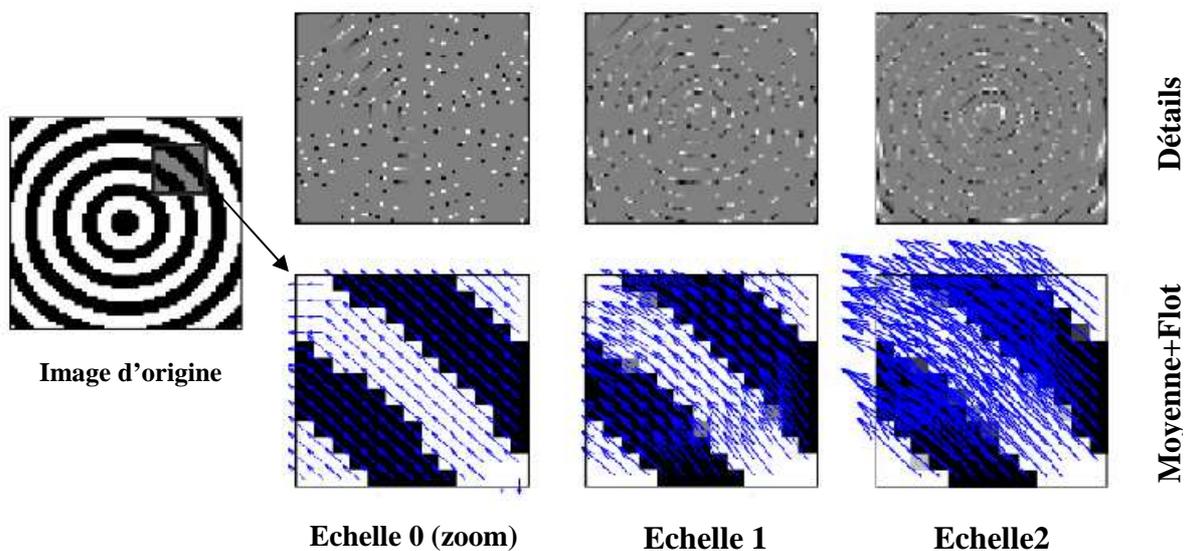
Passage en domaine spacial :  $B = \text{transforme\_inverse}(B_0, \dots, B_m)$ .

**N.B** : Les fonctions *transforme* et *transforme\_inverse* sont inverses l'une de l'autre et calculent une représentation multirésolution de l'image. La fonction *egalisation* réalise l'égalisation d'histogrammes.

- ❖ *Cet algorithme sert à calculer la transformée de l'image d'origine qui est calculée sur une représentation multirésolution  $(A_0, \dots, A_{j-1})$  de la texture d'origine A, tel que B c'est la fonction d'égalisation d'histogramme.*

### IV.2.3. Transformée en bandelettes par groupements :

La figure IV.8 montre les coefficients de détails, ainsi que le champ d'association (affiché sur l'image de moyenne temporaire A à l'échelle  $\ell$ ). On voit que le champ est de plus en plus long et qu'il y a très peu de coefficients de détails non nuls.



**Fig IV.8** : Champ d'association géométrique multi-échelles détecté par l'algorithme2, coefficients de moyenne et de détail.

**Algorithme 2 : la transformée en bandelettes par groupements.**

Fonction  $[D_0, \dots, D_L, F_0, \dots, F_{L-1}, W] = \text{transfo\_bandelettes}(M)$ .

ou Fonction  $[D_0, \dots, D_L, F_0, \dots, F_{L-1}, W] = \text{transfo\_bandelettes}(M, \check{F}_0, \dots, \check{F}_{L-1})$ .

Entrée : image  $M$ ,

Sortie : coefficients de détails  $D_0, \dots, D_{L-1}$ ,

moyenne basse fréquence  $D_L$ , poids  $W$ ,

champ d'association multi-échelles  $F_0, \dots, F_{L-1}$ , avec  $F_\ell [x] \in \mathbb{R}^2$ .

Initialisation :  $A = M$ .

Répéter pour chaque échelle  $\ell = 0, \dots, L-1$  :

Répéter pour chaque pixel  $x_k$  :

Si  $\ell > 0$ , calcul du centre de la zone de recherche

$$Z_k = x_k + F_{\ell-1} [x_k] + F_{\ell-1} [x_k + F_{\ell-1} [x_k]],$$

(sinon on pose  $Z_k = x_k$ ).

Si le champ  $\check{F}_\ell$  n'est pas fourni, recherche du meilleur appariement :

pour chaque pixel  $y \in Z_k + V$ , calculer

$$E(y) = \sum_{s \in V} |A[y + s] - A[x_k + s]|^2$$

retenir  $y_k = \text{argmin}(E(y))$  et poser  $F_\ell [x_k] = y_k - x_k$

(sinon on pose :  $F_\ell [x_k] = \check{F}_\ell [x_k]$  et  $y_k = x_k + F_\ell [x_k]$ )

Transformée de Haar : poser

$$D_\ell [x_k] = (A[x_k] - A[y_k]) \sqrt{W[x_k]W[y_k]} / \sqrt{W[x_k] + W[y_k]}.$$

$$A[y_k] = (W[x_k]A[x_k] + W[y_k]A[y_k]) / (W[x_k] + W[y_k]).$$

$$W[y_k] = W[x_k] + W[y_k].$$

Enregistrer les basses fréquences  $D_L = A$ .

**N.B :** La fonction a deux prototypes suivant que l'utilisateur fournit un champ pré-calculé  $\check{F}_0, \dots, \check{F}_{L-1}$ , ou bien que l'algorithme doit calculer lui-même le champ  $F_0, \dots, F_\ell$  par recherche de meilleurs appariements.

- ❖ Cet algorithme renvoie des coefficients de détails  $\{D_\ell\}_{\ell=0}^{L-1}$  ainsi qu'une image basse fréquence  $D_L$ . L'algorithme peut être employé dans une version simplifiée où il ne calcule par les champs  $F_\ell$  ( $F_\ell$  : champs d'association) ceux-ci étant supposés donnés par l'utilisateur par l'intermédiaire des variables  $\check{F}_\ell$ , ( $\check{F}_\ell$  : champs d'association synthétisé).

### Algorithme3 : la transformée en bandelettes par groupements inverse.

Fonction  $M = \text{transfo\_bandelettes\_inverse}(D_0, \dots, D_L, F_0, \dots, F_{L-1}, W)$ .

Entrée : coefficients de bandelettes  $D_0, \dots, D_L$ , matrice de poids  $W$

champ d'association multi-échelles  $F_0, \dots, F_{L-1}$ , avec  $F_\ell[x] \in \mathbb{R}^2$ .

Sortie : image  $M$ .

Initialisation :  $M = D_L$  (basse fréquences).

Répéter pour chaque échelle  $\ell = L-1, L-2, \dots, 1$  :

Répéter pour chaque pixel  $x_k$  :

Pixel couplé  $y_k = x_k + F_\ell[x_k]$  transformée de Haar inverse :

$$M[y_k] = M[x_k] - D[x_k] \sqrt{W[y_k] - W[x_k]} / \sqrt{W[y_k]W[x_k]}$$

#### IV.2.4. Algorithme générique de synthèse de texture :

L'algorithme générique de synthèse de textures à l'aide de la transformée en bandelettes utilise l'égalisation d'histogrammes des coefficients en bandelettes par groupements calculés sur une représentation multirésolution  $A_0, \dots, A_{j-1}$  de la texture d'origine  $A$ .

**Algorithme4: Algorithme générique de synthèse de texture**

Fonction  $B = \text{synthese\_bandelettes}(A)$ .

Entrée : texture originale  $A$  de taille  $n \times n$ .

Sortie : texture synthétisée  $B$ .

Initialisation :  $B = \text{bruit\_blanc}(n)$ .

Calcul du champ d'association à synthétiser

$$[\check{F}_1, \dots, \check{F}_{L-1}] = \text{calcul\_champ}(A).$$

Égalisation des histogrammes en espace

$$A = \text{egalisation}(A, B).$$

Calcul de la transformée multirésolution de la texture d'origine

$$[A_0, \dots, A_m] = \text{transforme}(A).$$

Répéter jusqu'à convergence :

Calcul de la transformée multirésolution de la texture synthétisée

$$[B_0, \dots, B_m] = \text{transforme}(B).$$

Répéter pour chaque échelle  $j = 0, \dots, m$  :

Égalisation des histogrammes multirésolution

$$A_j = \text{egalisation}(A_j, B_j).$$

Calcul du champ d'association et des coefficients de bandelettes à l'échelle  $j$

$$[F_0, \dots, F_{L-1}, D_0, \dots, D_L] = \text{transfo\_bandelettes}(A_j).$$

Calcul des coefficients de bandelettes de l'image à synthétiser à l'échelle  $j$

$$[\check{D}_0, \dots, \check{D}_L] = \text{transfo\_bandelettes}(B_j, \check{F}_0, \dots, \check{F}_{L-1})$$

Répéter pour chaque échelle  $\ell = 0, \dots, L-1$  :

Égalisation des histogrammes en bandelettes

$$\check{D}_\ell = \text{egalisation}(\check{D}_\ell, D_\ell).$$

Calcul de la transformée en bandelettes inverse de la texture synthétisée

$$B_j = \text{transfo\_bandelettes\_inverse}(\check{D}_0, \dots, \check{D}_L, \check{F}_0, \dots, \check{F}_{L-1}).$$

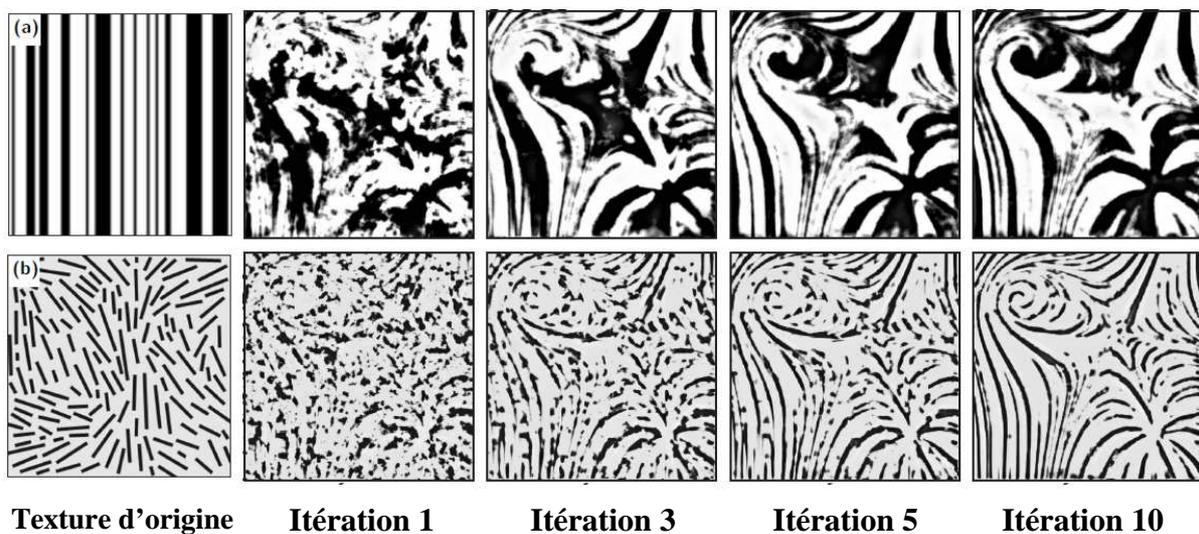
Calcul de la transformée en multirésolution inverse de la texture synthétisée

$$B = \text{transforme\_inverse}(B_0, \dots, B_m):$$

**N.B :** La fonction *egalisation* fait l'objet du l'algorithme 1 et la fonction *transfo\_bandelettes* est décrite dans l'algorithme2. La fonction *transforme* réalise une transformée multirésolution

(transformée en ondelettes). La fonction *calcul\_champ* détermine un champ d'association à partir de la texture d'origine ou à partir de données fournies par l'utilisateur.

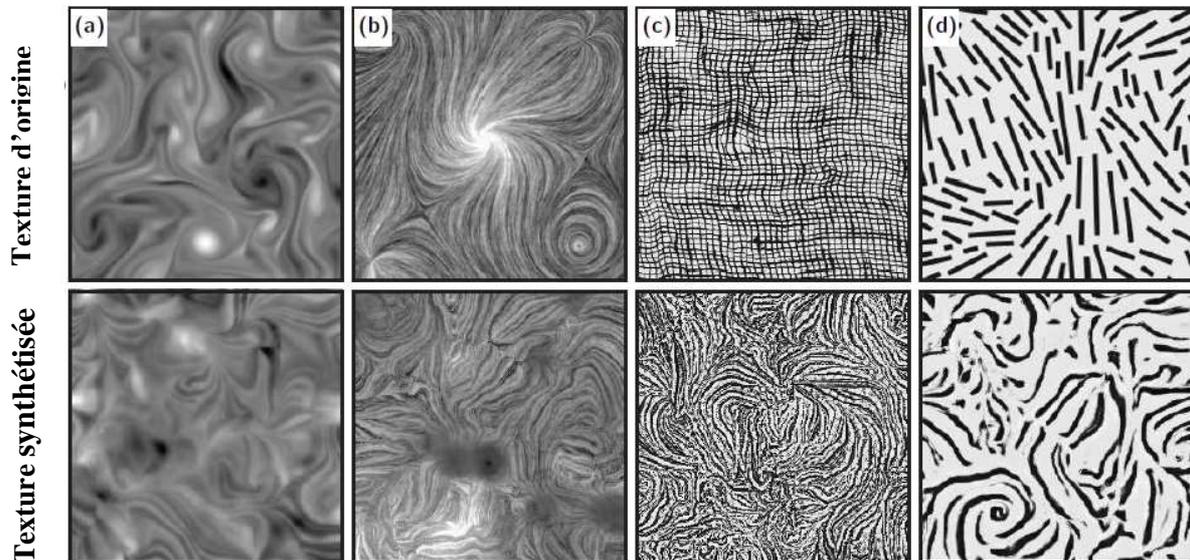
- ❖ Cet algorithme sert à synthétiser une texture discrétisée  $B$  de  $n \times n$  pixels à partir d'une texture originale  $A$  de même taille, pour cela on utilise la procédure d'égalisation d'histogrammes présentée dans l'algorithme1, il faut calculer un champ d'association  $F_0, \dots, F_{L-1}$  pour la texture d'origine  $A$  et un autre champ  $\check{F}_0, \dots, \check{F}_{L-1}$  pour la texture  $B$  en cours de synthèse. Le champ d'origine est calculé à l'aide de l'*algorithme2* (transformée en bandelette), puis le calcul de la transformée en bandelettes inverse de la texture synthétisée.
- ❖ L'*algorithme 4* est itératif et il est nécessaire d'effectuer une dizaine d'itérations pour obtenir une synthèse de qualité. La figure IV.9 montre l'évolution de la synthèse.



**Fig IV.9 :** Progression de la synthèse de textures.

- ❖ On peut générer de nouveaux paramètres angulaires  $\tilde{\omega}_\ell$  en utilisant une procédure d'égalisation d'histogrammes similaire à celle décrite par l'algorithme1.

Une fois ces paramètres synthésés, on peut générer un champ d'association  $\{\tilde{F}_\ell\}_\ell$ . Ceci constitue une nouvelle implémentation de la fonction  $\text{calcul\_champ}(A)$  du l'algorithme4.



**Fig.IV.10** : Synthèse de texture avec synthèse du champ d'association et des coefficients de bandelettes.

La figure IV.10 montre des résultats de synthèse de textures turbulentes. On observe que les textures synthésées possèdent des structures géométriques relativement longues.

- En (a) et (b), les structures géométriques sont un peu plus courtes que celles d'origine et la géométrie est globalement moins régulière. Ceci provient de la modélisation des champs d'association par des coefficients d'ondelettes de différences angulaires qui n'est pas assez efficace.
- En (c) la géométrie est totalement différente de la géométrie initiale (composée d'un enchevêtrement de courbes exclusivement horizontales et verticales). Ceci est dû au fait que notre transformée n'est pas capable de prendre en compte la juxtaposition de deux orientations (la géométrie doit être localement parallèle).

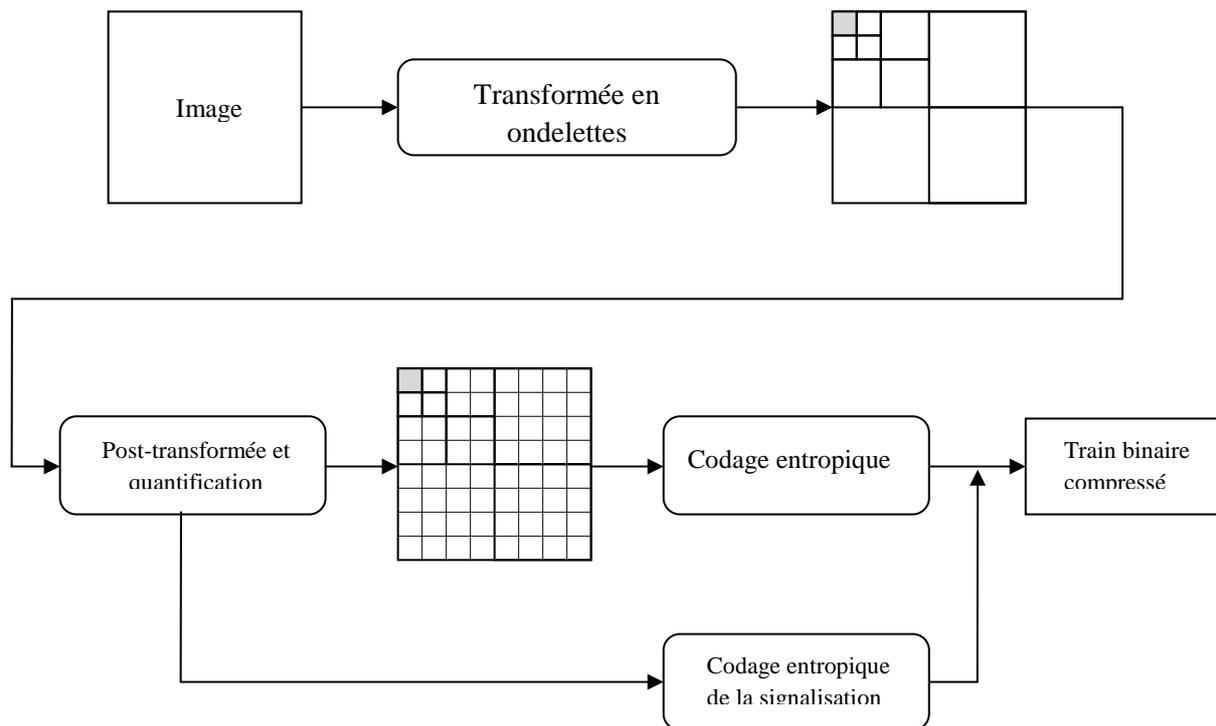
- En (d), on observe une géométrie courbe (ou rectiligne sur des courtes distances), différente de la géométrie initiale. Les champs d'association sont en effet synthétisés indépendamment des coefficients de bandelettes. Ceci est problématique pour les textures possédant des zones sans géométrie.

### IV.3. Application à la Compression :

Les systèmes de compression les plus récents JPEG2000, utilisent une transformée en ondelettes séparables. Ils sont adaptés à la description de zones homogènes et des contours de l'image dans des directions particulières, elles ne représentent pas tous les contours. De nouvelles transformées géométriques [12] ont également été proposées. Les éléments de base de ces transformées sont directionnels et anisotropes et donc mieux adaptés à la représentation des contours, qui est la transformée en bandelettes.

#### IV.3.1 Algorithme de compression en bandelettes

La figure IV.11 illustre le schéma de la compression en bandelettes. Dans un premier temps, on effectue la transformée en ondelette de l'image. Les sous bandes de la transformée en ondelette sont ensuite segmentées en bloc  $4 \times 4$  coefficients. Les blocs ainsi obtenus sont ensuite transformés dans les bases bandelettes et quantifiés pour obtenir la transformée en bandelettes de l'image et la signalisation de la transformée utilisée sur chaque bloc. Enfin, on code entropiquement les coefficients de bandelettes quantifiés et la signalisation pour obtenir une image compressée.



**Fig IV.11** : Schéma de compression par transformée en bandelettes par bloc.

### IV.3.2 Algorithme rapide

Nous décrivons dans cette section l'algorithme de la transformée en bandelettes rapide et direct. Ceci inclut à la fois le calcul de la meilleure base de bandelettes  $\mathcal{B}^*(\Gamma)$  ainsi que le calcul de la décomposition de  $\bar{f}$  dans cette base. L'algorithme inverse en découle immédiatement, puisqu'il ne nécessite pas la recherche de la meilleure base et qu'il consiste à inverser les transformées en ondelettes ainsi que la transformée de Haar sur les carrés des quadrees.

#### (1) Paramètres de l'algorithme

L'utilisateur fournit une image discrétisée  $\bar{f}$  de taille de  $N * N$  pixels, et un seuil  $T$  qui contrôle l'application. On note  $2^J = N^{-1}$ .

#### (2) Transformée en ondelettes 2D

On calcule la transformée en ondelettes discrète 2D de l'image d'origine  $\bar{f}$ . Ceci produit un ensemble de vecteurs de coefficients  $\{\bar{f}_j^k\}_{k,j>J}$ .

Les vecteurs  $\bar{f}_j^k$  pour chaque échelle  $2^j$  et orientation  $k \in \{V, H, D\}$  peuvent être stockés dans une unique image ayant la même taille que l'image originale  $\bar{f}$ . Les étapes suivantes (3)-(7) implémentent la bandéletisation, qui est répétée pour chaque échelle et orientation.

**(3) Sélection de chaque carré dyadique**

Un carré dyadique est un carré obtenu en subdivisant de façon récursive le carré  $[0,1]^2$  en quatre carres de même taille. Pour chaque carré  $S$  on regroupe les coefficients en ondelettes  $\{\bar{f}_j^k[n]\}_{2^j n \in S}$  dans un vecteur noté  $f_S$ . Les étapes suivantes (4)-(7) sont répétées pour chaque carré dyadique  $S$  à chaque échelle  $2^j$  et orientation  $k$  d'une transformée en ondelettes.

**(4) Quantification de la géométrie**

Nous devons maintenant trouver la meilleure géométrie approchée quantifiée  $\tilde{\gamma}_S$  à l'intérieur d'un carré  $S$ . Nous devons donc tester toutes les bases de bandelettes  $\mathcal{B}(S, \tilde{\gamma}_m^\Omega)$  de  $l_j^2(S)$  pour les géométries

$$\tilde{\gamma}_m^\Omega = \sum_{i=0}^{p-1} (m_i T^2) \theta_i \quad \text{Où} \quad |m_i| \leq A_\theta T^{-2} \quad \text{et} \quad \Omega \in \{V, H\}$$

Pour chaque ensemble possible de coefficients  $m = \{m_i\}_i$  et orientation  $\Omega$ . Nous ajoutons aussi à ces géométries le symbole spécial  $\tilde{\gamma} = \Xi$  dont la transformée correspondante est l'identité. Une géométrie horizontale  $\tilde{\gamma}_m^H$  (respectivement une géométrie verticale  $\tilde{\gamma}_m^V$ ) paramétrise une courbe approchée horizontale  $(x, \tilde{\gamma}_m^H(x))$  (respectivement une courbe approchée verticale  $(\tilde{\gamma}_m^V(y), y)$ ).

L'étape suivante (5) est répétée pour chaque géométrie potentielle  $\tilde{\gamma}_m^\Omega$ .

**(5) Calcul de la transformée de Alpert**

Pour une géométrie donnée  $\tilde{\gamma} = \tilde{\gamma}_m^\Omega$ , on calcule l'ensemble des produits scalaires

$$A_{\tilde{\gamma}}(f_S)[l, i] \stackrel{\text{def}}{=} \langle f_S, \Psi_{li} \rangle \quad \text{Où} \quad \mathcal{B}(S, \tilde{\gamma}_S) = \{\Psi_{li}\}_{l,i}$$

Cette transformée peut être calculée avec l'algorithme de transformée de Alpert discrète, comme expliqué dans [10]. Pour le cas particulier  $\tilde{\gamma} = \Xi$ , la transformation  $A_\Xi$  est l'identité.

**(6) Sélection de la meilleure géométrie**

Pour un seuil donné  $T$ , nous devons choisir la meilleure géométrie  $\tilde{\gamma} = \tilde{\gamma}_m^\Omega$  qui minimise le Lagrangien.

$$\mathcal{L}(\tilde{f}_S, \mathcal{B}(S, \tilde{\gamma}), T) = \sum_{(l,i) \in J_T} |A_{\tilde{\gamma}}(\tilde{f}_S)[l, i]|^2 + (\text{Card}(J_T) + M_G) T^2 \tag{III.44}$$

Où

$$J_T = \{(l, i) \mid |A_{\tilde{\gamma}}(\tilde{f}_S)[l, i]| > T\}$$

Et

$$\begin{cases} M_G = 1 & \text{si } \tilde{\gamma} = \Xi \\ M_G = p + 1 & \text{sinon} \end{cases}$$

On peut ainsi définir la géométrie sélectionnée pour le carré  $S$  par

$$\tilde{\gamma}_S = \operatorname{argmin}_{\tilde{\gamma}_m^\Omega} \mathcal{L}(\tilde{f}_S, \mathcal{B}(S, \tilde{\gamma}), T) \quad (\text{III.45})$$

### (7) Construction du quadtree

Une fois que l'on a calculé les coefficients transformés  $A_{\tilde{\gamma}_S}(\tilde{f}_S)$  pour chaque carré dyadique, on doit choisir la meilleure segmentation en carré. Ceci revient à conserver uniquement un sous-ensemble de carrés dyadiques de façon à obtenir une segmentation en quadtree. Une telle segmentation peut s'obtenir à partir de la segmentation complète en petits carrés, suivi d'une phase de regroupement par quatre des carrés. Grâce à l'additivité du Lagrangien et à la structure hiérarchique du quadtree, la minimisation de  $\mathcal{L}$  s'effectue à l'aide d'un algorithme rapide parcourant l'arbre de bas en haut. Cet algorithme a été introduit par Donoho[14]. A l'étape précédente (6), nous avons enregistré, pour chaque carré dyadique  $S$  la valeur  $\mathcal{L}(S) = \mathcal{L}(\tilde{f}_S, \mathcal{B}(S, \tilde{\gamma}), T)$  du Lagrangien restreint à  $S$ , de même que la meilleure géométrie quantifiée  $\tilde{\gamma}_S$ . Alors, pour chaque échelle  $2^j$  et orientation  $k$ , on calcule la structure de quadtree à l'aide de l'algorithme suivant

#### (a) Initialisation du quadtree

Chaque carré  $S$ , de taille  $b = 2^j$  est une feuille. Enregistrer les géométries optimales  $\tilde{\gamma}_S$  et initialiser  $\mathcal{L}_0$ , le Lagrangien cumulatif du sous-arbre à  $\mathcal{L}_0(S) = \mathcal{L}(S)$

#### (b) Commencer avec des carrés $S$ de taille $2 * 2^j$

#### (c) Pour chaque carré $S$ , on note $(S_1, S_2, S_3, S_4)$ ses quatre sous-carrés et

$$\mathcal{L}'(S) = \mathcal{L}_0(S_1) + \mathcal{L}_0(S_2) + \mathcal{L}_0(S_3) + \mathcal{L}_0(S_4) + T^2 \quad (\text{III.46})$$

Le Lagrangien du sous-arbre (le terme additionnel  $T^2$  est du au coût d'un coefficient pour la subdivision). Les sous-carrés doivent être rassemblés si  $\mathcal{L}(S) < \mathcal{L}'(S)$ . Si c'est le cas, déclarer  $S$  comme une feuille, enregistrer la géométrie optimale  $\tilde{\gamma}_S$  et mettre à jour

$$\mathcal{L}_0(S) = \min(\mathcal{L}(S), \mathcal{L}'(S)) \quad (\text{III.47})$$

Tant que  $b \geq 1$ , faire  $b \leftarrow 2 * b$  et répéter l'étape précédente (b).

L'étape Post-transformée et quantification est représentée par le schéma de la figure IV.12[5]

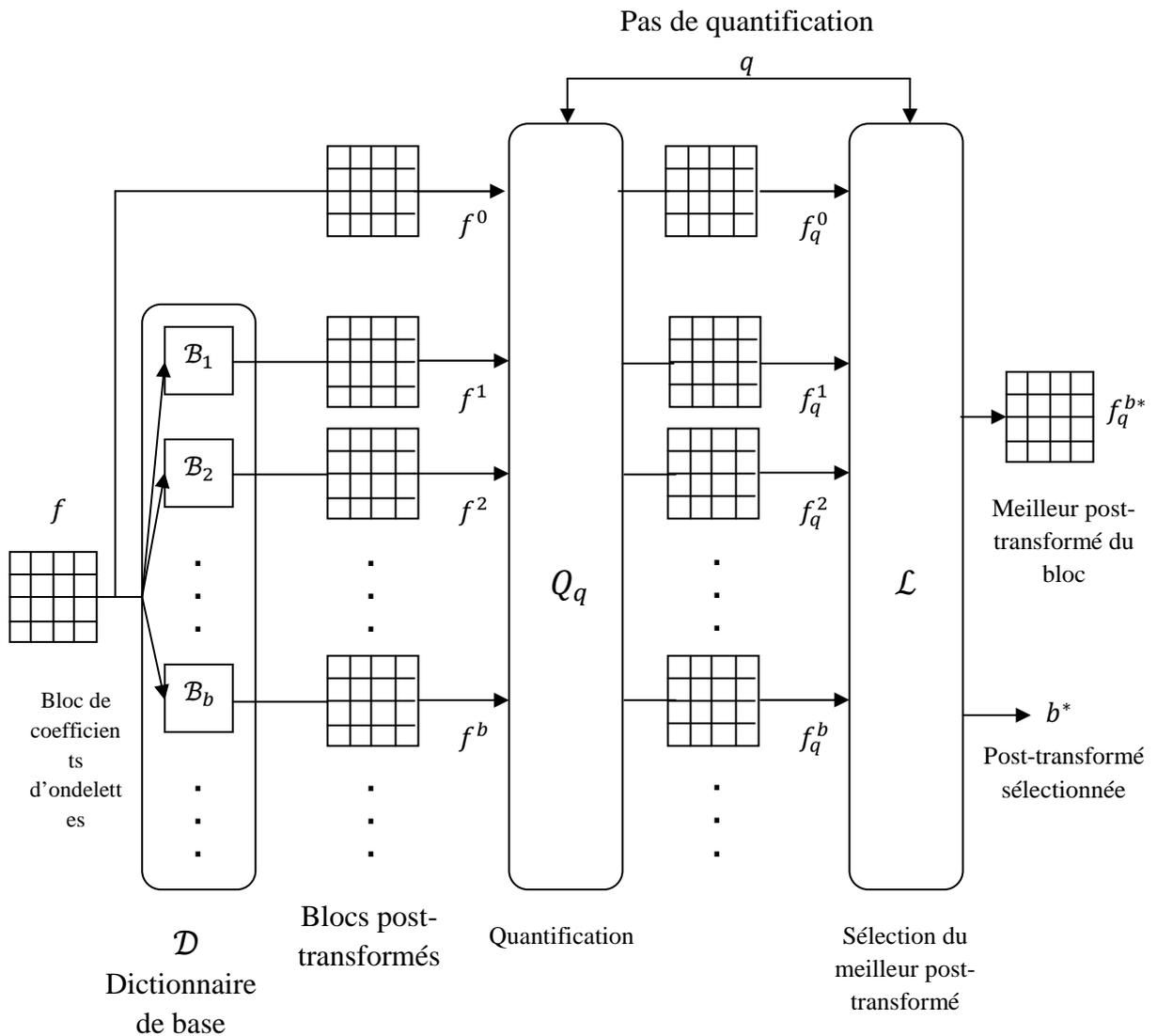


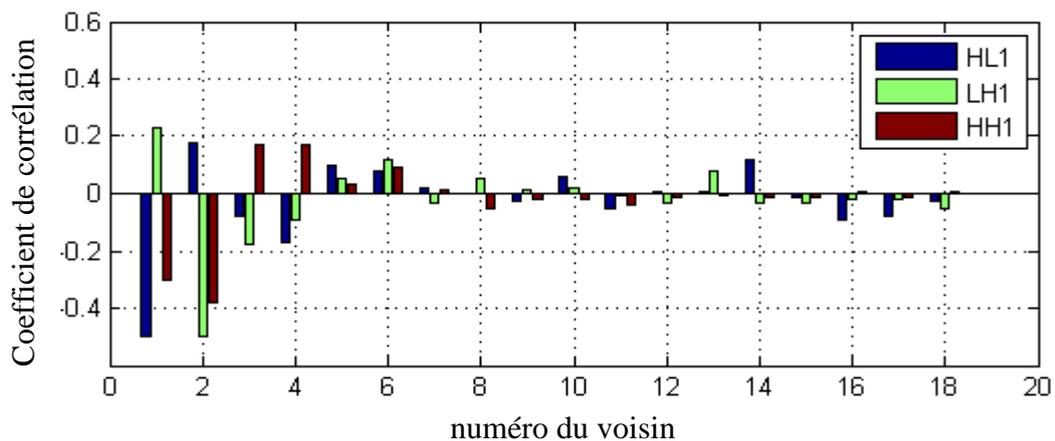
Fig IV.12: Processus de transformée en bandelettes d'un bloc  $f$  de coefficients d'ondelettes.

**IV.3.3. Choix de la taille des blocs [5]**

Dans cette section, on s'intéresse à la taille de blocs permettant d'obtenir les meilleurs résultats en compression pour la transformée en bandelettes. Pour choisir une taille de blocs, nous commençons par étudier les dépendances entre coefficients d'ondelettes voisins. En effet, le but de la transformée est de réduire ces dépendances.

		16	14	17		
	11	8	6	9	12	
15	7	3	2	4	10	18
13	5	1	C			

**Fig IV.13:** Définition d'un voisinage intra-bande dans le domaine

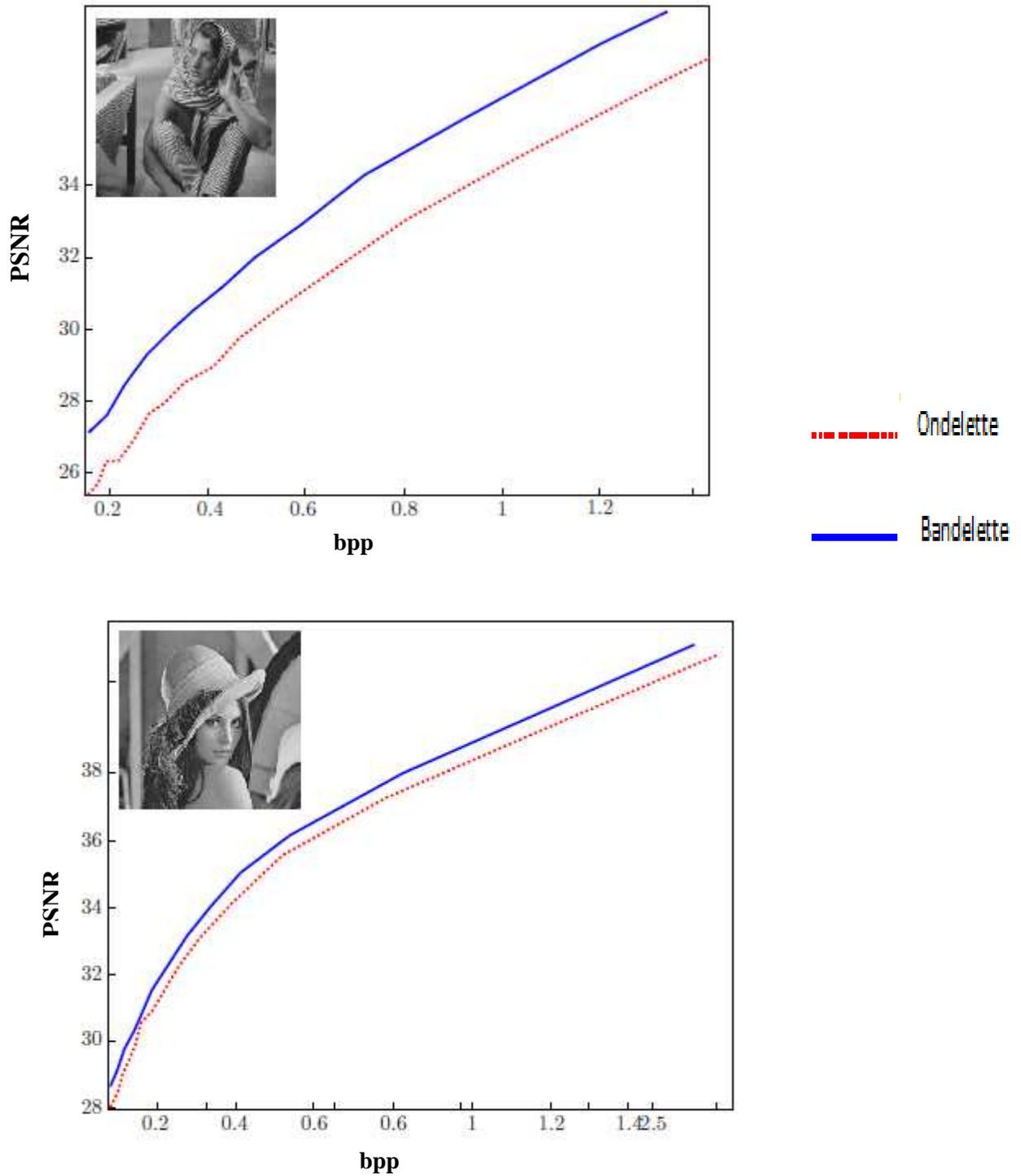


**Fig IV.16 :** Coefficients de corrélation entre un coefficient d'ondelettes C et chacun de ses dix-huit plus proches voisins intra-bande.

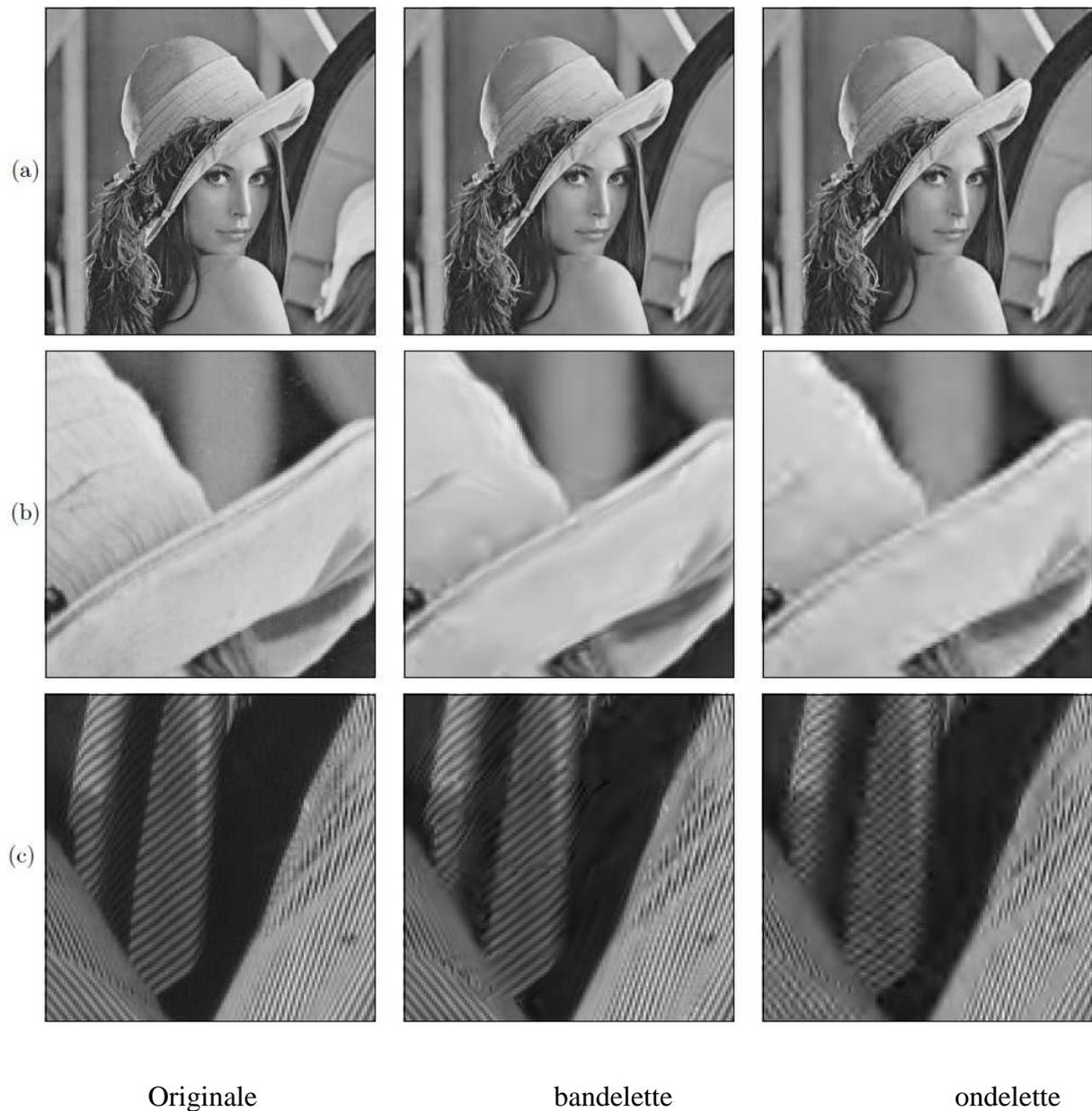
#### IV.3.4. Performance de la transformée en bandelette en compression

Les performances d'une base en compression dépendent de la capacité d'approximation de celle-ci. Pour une classe d'image régulière par morceaux, la transformée en ondelettes est effectuée sur trois niveaux de décomposition, ce nombre de niveaux de décomposition permet un bon compromis entre les performances en compression et la complexité calculatoire, la transformée en bandelettes est effectuée en suivant les étapes de l'algorithme précédemment. les images de l'ensemble de test sont compressées successivement en utilisant divers pas de quantification  $q$ . Pour plus de lisibilité, les performances obtenues avec la transformée en bandelettes sont tracées relativement aux performances obtenues avec la transformée en ondelettes.[10]

La figure IV.17, présente la courbe de performance pour les images de Lena et Barbara. Le codage en bandelettes est supérieur au codage en ondelettes en améliorant le PSNR.



**Fig. IV.17 :** Courbes de distorsion du codeur en bandelettes pour (gain en PSNR  $\gg 0.6\text{dB}$ ) et Barbara (gain en PSNR  $\gg 2\text{dB}$ ).



**Fig IV.18:** Résultat de compression : (a) Lena, (b) détail de Lena et (c) détail de Barbara. L'image de Lena est comprimée à 0,22 bits par pixel en bandelettes et en ondelettes et les PSNR respectifs sont de 33,04 db et 32,55 db. Pour Barbara, l'image est comprimée à 0,39 bits par pixels et les PSNR respectifs sont de 31,22 db et 28,50 db.

**Conclusion :**

Nous avons étudié et présenté les résultats de l'application d'un algorithme de débruitage par seuillage dur dans un frame de bandelettes invariant par translation. Ces résultats sont comparés avec un algorithme similaire dans un frame d'ondelettes isotropes invariant par translation. Et nous avons étudié un algorithme générique dans le but de synthétiser des textures géométriques, en utilisant la transformée en bandelettes par groupements. Et enfin, nous avons étudié et comparé les performances de la transformée en bandelettes par blocs. Nous avons vu que cette dernière permet d'améliorer les résultats en compression par rapport à la transformée en ondelettes.

Les bandelettes ont de bien meilleures performances d'approximation non-linéaire que les ondelettes.

### **Conclusion :**

Dans ce travail, nous nous sommes intéressés à la représentation géométrique des images. Ce travail consiste d'une part, à faire une recherche bibliographique dans le domaine de traitement d'images en générale et d'autre part à présenter une nouvelle technique qui capture la régularité géométrique des images le long des contours.

La transformée en bandelettes représente efficacement les contours des images, grâce aux bases orthogonales utilisées, et les zones de régularité homogène, sous forme d'un résidu décomposé en ondelettes.

Nous avons présenté la première génération de la transformée en bandelettes proposée par Le Pennec [11]. Ces bandelettes sont obtenues à partir de déformations d'ondelettes anisotropes permettant via une rectification géométrique de capturer à la fois la régularité le long des contours et la régularité de zones. La seconde génération de la transformée en bandelettes est présentée par Peyré [10] qui s'appuie sur une segmentation en quadtree des différentes bandes de fréquences de la transformée en ondelettes.

Nous avons présenté les résultats des applications aux: débruitage, l'analyse de textures et la compression d'images, donnés dans la littérature.

- le débruitage par bandelettes a permis d'obtenir un gain en PSNR par rapport au débruitage en ondelettes. Bien que ce gain soit faible, mais les images obtenues avec les bandelettes semblent visuellement plus fines que les images obtenues avec les ondelettes.
- Les structures géométriques, obtenues dans l'analyse de la texture sont proches de celles d'origines et la géométrie globalement moins régulière.
- Nous avons terminé par une comparaison visuelle entre les résultats issus des deux transformées ; la transformée en bandelettes respecte mieux les contours très marqués de l'image.

Vu les résultats médiocres de l'application des bandelettes dans l'analyse de textures, plusieurs améliorations ont été apportées et présentées dans les travaux de : Le pennec [9] [11] et Peyré&Mallat publiés dans la littérature.

Une généralisation vers une 3D des bandelettes a été aussi développée par Le-pennec&Mallat déposé dans un brevet.

Nous nous espérons que nous avons atteint l'objectif du thème proposé, ainsi que ce mémoire servira pour les promotions à venir comme état de l'art sur l'application des bandelettes dans le domaine du traitement d'images.

# *Bibliographie*

- [1] **Rabia Nacim**, "Revue des méthodes de segmentation d'images texturées : cas des images couleurs", thèse de magister, département d'électronique, UMMTO 2011,
- [2] **Marion A**, "Introduction aux techniques de traitement d'image", Eyrolles, Paris, 1987
- [3] **Canny.J.F**, "A computational approach to edge detection", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol 8 p.769-768, 1986.
- [4] **Deriche.R**, "Optimal edge detection using recursive filters", In Proc. International Conference on Computer Vision (ICCV'87), London, United Kingdom, p.8-12, 1987
- [5] **X. Delaunay**. "Compression d'images satellite par post-transformées dans le domaine ondelettes", Thèse de doctorat, Université de Toulouse, novembre 2008.
- [6] **Ouahioune Mohand**, "compression d'images hyperspectrales par la transformée en ondelette 3D", thèse de magister, département d'électronique, UMMTO 2011,
- [7] **Jean-Paul Guillement**, "Analyse de Fourier - Ondelettes", cours, Département de Mathématiques, université de Nantes, 2011.
- [8] **Oukali Salim**, "compression d'image par bandelettes : application à des images de télédétection", thèse de magistère, département d'électronique, UMMTO 2011,
- [9] **Erwan Le Pennec, Stéphane Mallat**, "Bandelettes et représentation géométrique des images", École Polytechnique, France. 2002.
- [10] **G. Peyré**. "Géométrie multi-échelle pour les images et les textures". Thèse de doctorat, École Polytechnique, 2005.
- [11] **E. Le Pennec**. "Bandelettes et représentation géométrique des images". Thèse de doctorat, École Polytechnique, décembre 2002.
- [12] **Xavier Delaunay, Marie Chabert, Géraldine Morin, Carole Thiebaut**, "Post-transformée dans le domaine ondelettes appliquée à la compression d'images satellite". CNRS, Troyes, France, septembre 2007.
- [13] **Y.Fisfer**, "fractal image compression theory and application", spurge-verlag, new yourk, 1994.
- [14] **D. Donoho**. Wedgelets : "Nearly-minimax estimation of edges". Ann. Statist, 27 :353–382, 1999.

**Annexe A :****Les mesures des performances :**

- **Le taux de compression :**

Le taux de compression est le rapport entre le nombre de bits utilisé par l'image originale et celui utilise par l'image comprimée.

Rapport de compression :

$$Rc = \frac{\text{nombre de bits de l'image originale}}{\text{nombre de bits de l'image décompressée}}$$

Taux de compression Tc :

$$Tc = \left(1 - \frac{1}{Rc}\right) \times 100$$

- **L'entropie :**

C'est la mesure de qualité d'information contenue dans une image. Une image dont tous les pixels ont la même valeur, contient très peu d'information donc son entropie est très faible, par contre, une image dont tous les pixels ont une valeur aléatoire, contient beaucoup d'information, son entropie est élevée.

$$H(p) = - \sum_{k=0}^m \{P(k) \log P(k)\}$$

Avec :

k : les niveaux de gris que peut revêtir un point particulier d'une image.

p(k) : la probabilité d'apparition du niveau du gris.

- **Mesure de distorsion :**

La distorsion est l'erreur quadratique moyenne introduite lors de compression d'une image (MSE : Mean Square Error) :

$$MSE = \frac{1}{MN} \sum_{i=0}^m \sum_{j=0}^n [I(i,j) - I'(i,j)]^2$$

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

$I'(i, j)$  : pixel de l'image reconstruite.

m: nombre de ligne de la matrice image.

n: nombre de colonne de la matrice image.

- **La qualité de reconstitution de l'image :**

le rapport signal sur bruit crête (PSNR : Peak Signal to Noise Ratio) : est la mesure de performance d'un codeur en termes de qualité visuelle de l'image reconstruite.

$$PSNR = 10 \log_{10} \left[ \frac{(255)^2}{MSE} \right] (\text{dB})$$

Avec : 255 représente les niveaux de gris max pour une image variant de 0 à 255.

## JPEG 2000 :

### Une norme en sept parties

La première remarque que nous pouvons faire à propos de cette norme est qu'elle ne définit que l'algorithme de décodage et le format des données compressées (*codestream*). Ceci laisse une assez grande liberté dans le choix du système de codage et ouvre la porte à la compétition pour l'élaboration de codeurs optimaux, pour autant que ceux-ci produisent des *codestreams* conformes aux spécifications de la norme. Cette dernière correspond en réalité à sept parties relativement indépendantes:

#### Partie 1

Elle définit la technologie minimale de l'algorithme de décodage ainsi que le format du *codestream* devant être compris par tous les produits se réclamant compatibles avec la norme. En outre, cette partie introduit le format de fichier JP2 qui permet d'associer au *codestream* des informations additionnelles sur l'image. Enfin, elle a pour but de couvrir une grande majorité des applications touchant au domaine de l'imagerie numérique (archivage, diffusion sur Internet...).

#### Partie 2

Celle-ci définit des extensions à la partie précédente afin d'améliorer les performances ou augmenter le nombre de fonctionnalités. Elle est surtout dédiée à des applications plus spécifiques telles que le codage de données hyper-spectrales (ex: images satellite).

#### Partie 3

Elle correspond à *Motion JPEG2000*. Elle définit un format de codage de séquences audiovisuelles et d'animations en tant que succession d'images indépendantes compressées par un système compatible avec JPEG2000 Partie1.

#### Partie 4

Elle définit les règles de compatibilité avec la norme JPEG2000.

#### Partie 5

Elle correspond au logiciel de référence de la Partie 1. Il y a en tout deux implémentations. La première, appelée JJ2000, est en Java et est élaborée dans le cadre d'un partenariat entre l'EPFL, Canon Research France et Ericsson (<http://jj2000.epfl.ch>), alors que la seconde, appelée JasPer, est écrite en C (<http://www.imagepower.com>).

**Partie 6**

Elle spécifie un format de fichier pour les images *compound* (i.e. contenant à la fois du texte, des photographies, des graphiques...).

**Partie 7**

Elle définit un support minimal de compatibilité pour des appareils à puissance limitée (ex: appareils photo numériques...).

Il est à noter qu'à ce jour, seule la Partie 1 a atteint le stade de norme internationale. C'est d'ailleurs essentiellement cette partie que nous détaillons dans le reste de cet article. Le lecteur aura déjà remarqué qu'une de deux implémentations de référence, JJ2000, est développée à l'EPFL, plus précisément au Laboratoire de Traitement des Signaux (LTS). En effet, le LTS participe activement à la définition de cette norme et a contribué, entre autres, au développement des technologies telles que le codage par régions d'intérêt. D'autres projets du LTS, en rapport direct avec JPEG2000, ont été NexImage (<http://ltswww.epfl.ch/~neximage/>) et Eurostill (<http://eurostill.epfl.ch/>). Le premier a vu comme résultat le développement du premier décodeur JPEG2000 public. Il était alors basé sur une version expérimentale de l'algorithme de JPEG2000 mais inspira, par la suite, la création de JJ2000.

**Les principales fonctionnalités de JPEG2000**

L'énorme avantage de JPEG2000 par rapport aux autres normes est qu'il définit, déjà dans sa Partie 1, une pléiade d'outils pouvant servir aux applications de tous les jours.

Parmi ceux-ci, le plus important est très certainement le décodage progressif. Pour cela, les *codestreams* JPEG2000 sont, de par leur nature, multi-résolutionnels. En effet, comme nous le verrons par la suite, l'image est compressée résolution par résolution, sans redondance, ce qui permet ensuite la transmission ou le décodage direct de la résolution la plus adaptée au système d'affichage. De plus, il est possible d'ordonner l'information dans le *codestream* de manière croissante en qualité, grâce au concept de *layers* expliqué un peu plus loin, ou encore de grouper l'information par rapport à la position qu'elle occupe dans une résolution de l'image.

De plus, ces trois types de progression peuvent être combinés et former, par exemple, un *codestream* où le décodage direct se fait par résolution et où, dans chaque résolution,

l'information apparaît de manière progressive en qualité. Enfin, il est à noter que la structure des *codestreams* JPEG2000 facilite l'accès aléatoire aux données. Ceci offre la possibilité de réordonner l'information au moment de la transmission et d'obtenir ainsi n'importe quel type de progression à partir d'un seul et unique *codestream* contenant un nombre de résolutions et de qualités suffisamment grand.

Une des caractéristiques nouvelles de JPEG2000 est la possibilité de définir des régions d'intérêt. Ces dernières permettent une affectation non uniforme du budget de qualité dans une image. Il est ainsi possible d'atteindre des taux de compression très élevés tout en gardant une très bonne précision dans les régions les plus importantes aux yeux de l'utilisateur.

Les autres fonctionnalités les plus marquantes de cette norme sont, d'une part, la possibilité de compresser sans perte ou avec pertes en utilisant le même algorithme ou encore l'intégration d'un mécanisme de détection d'erreurs très efficace pour les transmissions en environnements *bruités*. D'autre part, comme évoqué précédemment, elle permet l'accès aléatoire rapide aux données d'une image compressée, ce qui permet de ne décoder uniquement qu'une certaine zone d'une image lorsque celle-ci est très grande. Elle permet encore la plupart des manipulations géométriques élémentaires (*flip*, permutation paysage/portrait...) directement dans le domaine compressé et sans aucune perte en qualité. Enfin, une autre caractéristique de JPEG2000 est que le même algorithme de codage s'applique à une grande variété de types d'images (couleurs, niveaux de gris, multi-composantes...).

Pour finir, notons qu'un certain nombre d'autres fonctionnalités sont apportées par le format de fichier présenté ci-après.

### **Format de fichier**

Comme il a été mentionné auparavant, JPEG2000 Partie1 spécifie un format de fichier optionnel, appelé JP2. Celui-ci encapsule le *codestream* JPEG2000 et peut contenir d'autres informations importantes sur l'image. Le format de ce fichier est basé sur le concept de *boîtes*, où chaque boîte est une séquence contiguë de données, contenant des informations de type et longueur. Quelques boîtes - les *superboîtes* - peuvent en contenir d'autres, ce qui donne une structure hiérarchique. Les boîtes de base fournissent l'identification du type de fichier, un moyen de détection d'erreurs de transmission (e-mail 7 bits, transferts ftp ASCII, et similaires), la taille de l'image, le nombre de composantes, et leur gammes de valeurs, aussi

bien que la résolution d'acquisition et d'affichage par défaut. JP2 fournit aussi deux méthodes pour identifier l'espace de couleurs de façon précise. La méthode énumérée (par nom) impose à toutes les applications de pouvoir interpréter chacun des espaces possibles, ce qui rend son implémentation difficile dans le cas où il y en aurait un nombre important. Pour cette raison, cette méthode a été limitée à deux espaces: sRGB et niveaux de gris non-linéaires. Pour les autres espaces de couleurs, JP2 supporte une forme restreinte des profils ICC (*International Color Consortium*), qui autorisent une courbe de non-linéarité ainsi qu'une matrice de transformation 3x3.

En plus d'un espace couleur précis, JP2 permet de spécifier une palette de couleurs, qui peut contenir jusqu'à 1024 entrées, ainsi que l'information de transparence (plan alpha). Ce format de fichier permet aussi d'inclure des informations spécifiques à certaines applications avec des boîtes XML et UUID (*Universally Unique Identifier*), qui peuvent être interprétées par ces applications, et ignorées par d'autres, sans conséquences. Finalement, il y a aussi une boîte qui permet d'inclure des informations sur la propriété intellectuelle.

### **Quel est l'algorithme utilisé ?**

Comme évoqué précédemment, seul l'algorithme de décodage est décrit dans cette norme. Cependant, pour faciliter l'introduction de la terminologie et celle des techniques mises en œuvre, nous présentons ici un algorithme de codage typique compatible avec JPEG2000 Partie 1. Enfin, pour éviter d'alourdir les explications, nous n'aborderons pas ici certaines fonctionnalités avancées telles que les régions d'intérêt, le décodage progressif par position ou les outils de détection et correction d'erreurs, pour lesquels il est conseillé de se reporter aux références en fin d'article.

La figure 1 présente le schéma bloc d'un codeur JPEG2000. Comme on peut le constater, un algorithme typique de codage JPEG2000 se divise essentiellement en 5 modules: transformée couleur, transformée en ondelettes discrète, quantification, codage entropique et allocation de débit. Chacune de ces opérations est brièvement abordée dans la suite, mais notons tout d'abord que l'image de départ peut être divisée en *tiles* (pour des raisons d'usage de mémoire) qui sont ensuite codées de manière totalement indépendants mais réunis finalement dans un seul et même *codestream* (au moment de l'allocation de débit).



fig.1 - Schéma typique d'un codeur JPEG 2000

### ➤ Transformée couleur

Cette transformée est optionnelle mais peut s'avérer intéressante dans le cas où l'image (ou la *tile*) à coder appartient à l'espace de couleurs R, G et B. Elle permet alors d'obtenir une représentation de l'image dans un espace luminance/chrominance plus adaptée à la compression des données. Cette transformée est soit réversible et alors les données restent entières (ce qui autorise un éventuel codage sans perte), soit elle est irréversible et les données sont alors représentées en virgule flottante (ce qui implique déjà des pertes lors du codage). Il est à noter que l'image à coder peut être dans un tout autre espace couleur que ceux cités précédemment. Celui-ci doit alors être signalé grâce au format de fichier JP2.

### ➤ Transformée en ondelettes discrète

JPEG 2000 utilise une transformée séparable pour effectuer une décomposition dyadique de l'image entière (en réalité, chaque composante de chaque *tile*) en sous-bandes de fréquences (voir figure 2).

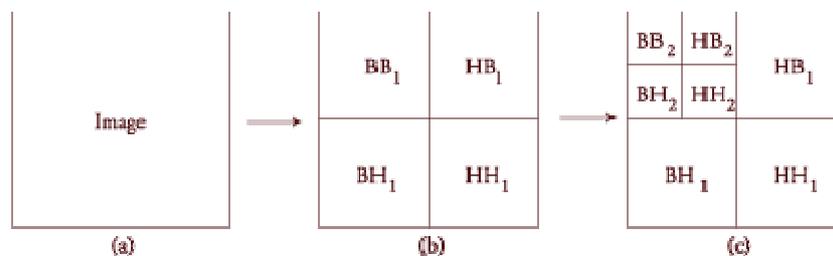


fig.2 - Division dyadique d'une image (a) en sous-bandes de fréquences après un (b) et deux (c) niveaux de décomposition.

Ainsi les lignes et les colonnes sont successivement décomposées suivant l'algorithme récursif de Mallat, ce qui donne autant de sous-bandes BB (basses fréquences horizontales et verticales), BH (basses fréquences horizontales et hautes fréquences verticales), HB (hautes fréquences horizontales et basses fréquences verticales) et HH (hautes fréquences horizontales et verticales).

Les filtres autorisés sont soit la paire (9,7) irréversible de Daubechies ou la paire (5,3) réversible de Le Gall. La première définit un filtre passe-bas à 9 coefficients et un filtre passe-haut à 7 coefficients, tous deux à coefficients irrationnels. D'un autre côté, la paire (5,3) de filtres passe-haut et passe-bas est à coefficients rationnels. En règle générale, la première permet des taux de compression plus élevés pour une qualité donnée, mais seule la seconde est utilisable pour compresser sans perte.

### ➤ Quantification

Lors du codage avec pertes, la précision sur les coefficients d'ondelettes est réduite par quantification scalaire uniforme. Cette étape permet, d'une part, de mettre à zéro les coefficients de faibles amplitudes (influant peu la qualité de l'image) mais fixe aussi le taux de compression minimal de l'image (i.e lorsque l'information quantifiée est intégralement incluse dans le *codestream*). La fonction de quantification est présentée à la figure 3 où est le pas de quantification de la sous-bande (chaque sous-bande peut avoir un pas de quantification différent),  $w$  est le coefficient d'ondelettes à quantifier et  $Q(w)$  l'index de quantification obtenu.

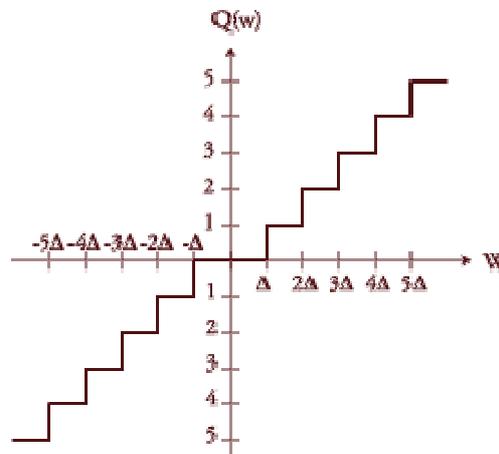


figure 3 - Fonction de quantification dans JPEG 2000

Dans le cas d'un codage sans perte, le pas de quantification est égal à 1, ce qui signifie qu'aucune quantification n'est appliquée.

### ➤ Codage entropique

JPEG2000 code les index de quantification obtenus précédemment grâce à un codeur arithmétique adaptatif avec contexte. Les coefficients quantifiés sont préalablement groupés,

dans chaque sous-bande, en blocs rectangulaires (*code-blocks*) typiquement de taille 64x64 ou 32x32. Puis chaque *code-block* est codé plan de bits par plan de bits, en commençant par les bits de poids le plus fort. En réalité, dans chaque plan de bits, les bits sont d'abord séparés en trois groupes, en fonction de leur voisinage, puis codés en trois passes (*coding passes*) successives.

Il est important de constater ici que le *bit stream* obtenu après un tel type de codage peut être tronqué à la fin de chacune des *coding passes* (il y a donc trois points de troncature possible par plan de bits) qui correspondent alors à des qualités progressives de chaque *code-block*. Cette propriété est la base de l'algorithme EBCOT et est largement exploitée par tous les types d'allocations de débits tel que celui présenté ci-dessous.

### ➤ Allocation de débit

Cette fonction peut varier fortement d'un algorithme de codage à un autre suivant les performances et fonctionnalités escomptées pour le codeur. Néanmoins tous les algorithmes d'allocation de débit ont pour but commun la création de paquets de données tels qu'ils sont définis dans la norme.

Chaque paquet correspond à un certain *layer* (généralement associé au concept de qualité) d'un niveau de résolution d'une composante de l'image. Il est constitué d'une en-tête identifiant son contenu et permettant un accès aléatoire rapide dans le *codestream*, ainsi que de données compressées obtenues par concaténation d'un certain nombre de *coding passes* de *code-blocks* d'un même niveau de résolution. Il est à noter que, pour obtenir des taux de compression élevés, les dernières *coding passes* d'un *code-block* sont souvent *sautées*. Ce dernier cas revient plus ou moins à changer le pas de quantification, et donc à diminuer la précision des coefficients dont les bits de poids les plus faibles ont été évincés.

Enfin chaque paquet est ajouté au *codestream* à la suite d'une en-tête (regroupant tous les paramètres de codage) et suivant un ordre d'inclusion dépendant du type de progression désiré (ex: par résolution, par *layer*). Rappelons tout de même qu'il est toujours possible de modifier la progression au moment de la transmission, à partir d'un même *codestream* stocké côté serveur. Ainsi, suivant l'ordre dans lequel le décodeur reçoit les paquets, il est capable de reconstruire une image progressivement par résolutions ou par *layers*.

## Annexe c :

## Coefficients des filtres d'ondelettes orthogonales :

Désignation	n	$\sqrt{2} * h(n)$
Haar	0	+0.707106781187
	1	+ 0.707106781187
Daubechies 04	0	+ 0.4829629131445341
	1	+0.8365163037378079
	2	+ 0.2241438680420134
	3	-0.1294095225512604
Daubechies 06	0	+ 0.3326705529500825
	1	+ 0.8068915093110924
	2	+0.4598775021184914
	3	-0.1350110200102546
	4	-0.0854412738820267
	5	+ 0.0352262918857095
Daubechies 08	0	+ 0.2303778133088964
	1	+ 0.7148465705529154
	2	+ 0.6308807679398587
	3	-0.0278937694168599
	4	-0.1870348117190931
	5	+ 0.0308413818355607
	6	+ 0.0328830116668852
	7	-0.0105974017850690
Daubechies 10	0	
	1	+0,1601023979741929
	2	+0,6038292697971895
	3	+0,7243085284377726
	4	+0,1384281459013203
	5	-0,2422948870663823
	6	-0,0322448695846381
	7	+0,0775714938400459
	8	-0,0062414902127983
	9	-0,0125807319990820
	10	+0,0033357252854738



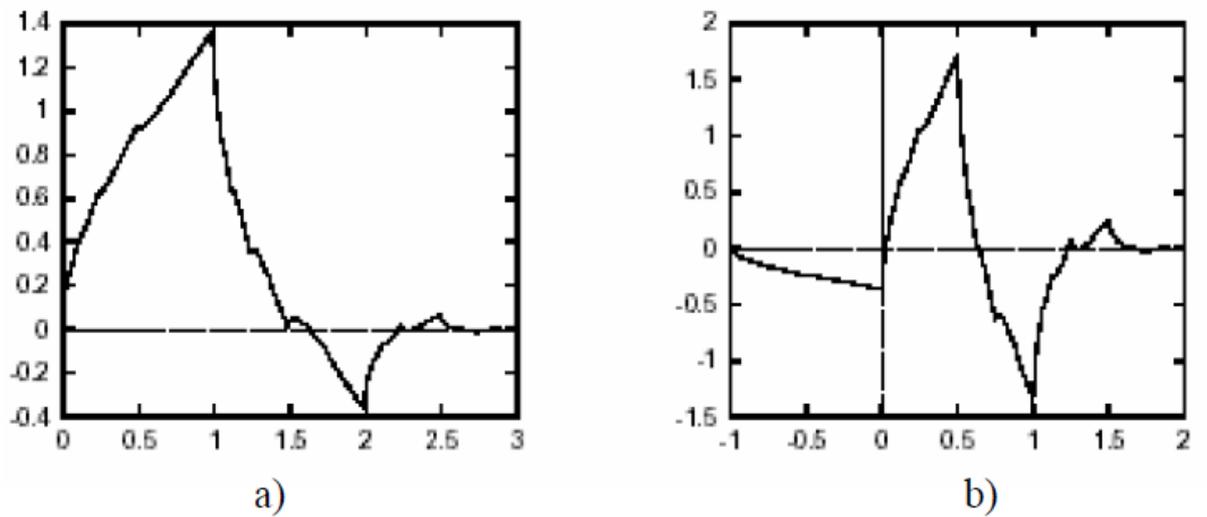
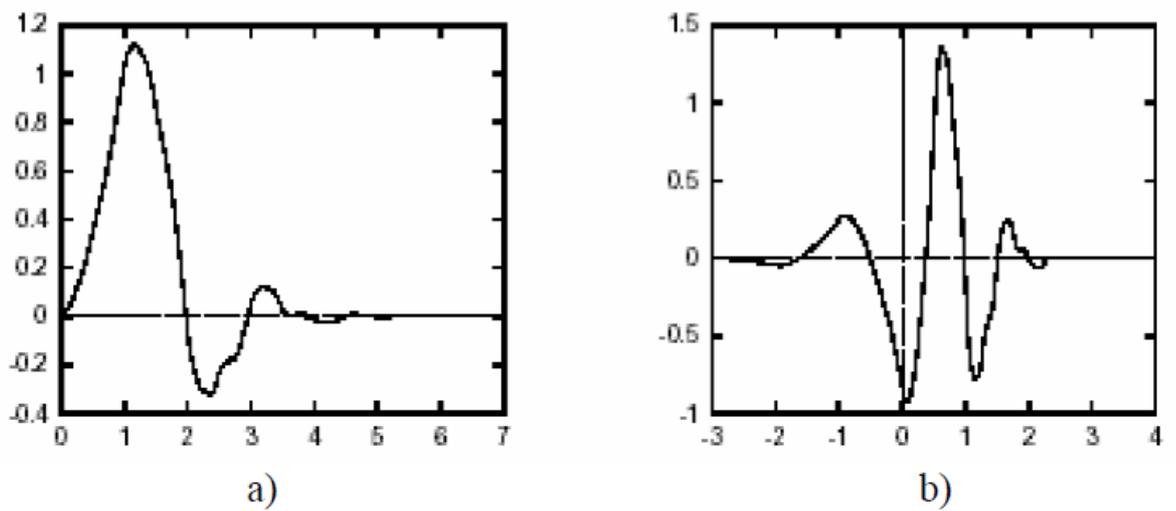
Daubechies 12	0	+0,1115407433591095
	1	+0,4946238003984533
	2	+0,7511339080210959
	3	+0,3152503517091982
	4	-0,2262646939654400
	5	-0,1297668675672625
	6	+0,0975016055873225
	7	+0,0275228655303053
	8	+0,0315820393174862
	9	+0,0005538422011614
	10	+0,0047772575109455
	11	-0,0010773010853085
<b>Désignation</b>	<b>n</b>	$\sqrt{2} * h(n)$
Daubechies 14	0	+0,0778520540850037
	1	+0,3965393194818912
	2	+0,7291320908461957
	3	+0,4697822874051889
	4	-0,1439060039285212
	5	-0,2240361849938412
	6	+0,0713092192668272
	7	+0,0806126091510774
	8	-0,0380299369350104
	9	-0,0165745416306655
	10	+0,0125509985560086
	11	+0,0004295779729214
	12	-0,0018016407040473
	13	+0,0003537137999715
Daubechies 16	0	+0,0544158422401072
	1	+0,3128715909143166
	2	+0,6756307362973495
	3	+0,5833346836342139
	4	-0,0158291052563853
	5	-0,2840133429615824
	6	+0,0004274845739124
	7	+0,1287474266204863
	8	-0,0173693010018090
	9	-0,0440882539307971
	10	+0,0139810279174001
	11	+0,0087460940471063
	12	-0,0048703529934520
	13	-0,0003917403733770
	14	+0,0006734494064506
	15	-0,0001174767841248
Daubechies 18	0	+0,0380779473638778
	1	+0,2438347746125858
	2	+0,6048231236900955
	3	+0,6572880780512736
	4	+0,1331973858249883

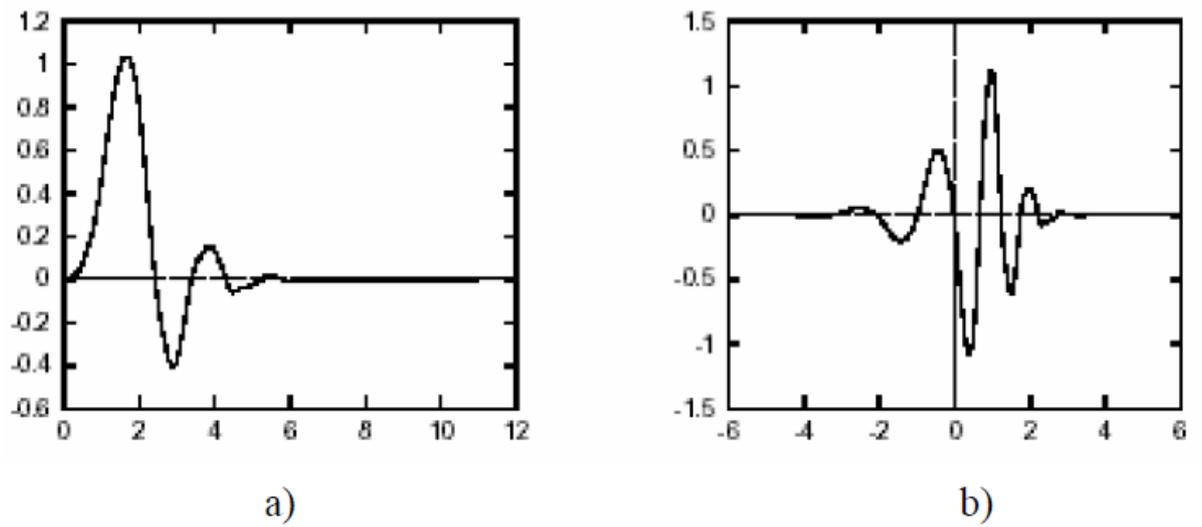
	5	-0,2932737832791663
	6	-0,0968407832229492
	7	+0,1485407493381256
	8	+0,0307256814793385
	9	-0,0676328200613279
	10	+0,0002509471148340
	11	+0,0273616621236798
	12	-0,0047232047377518
	13	-0,0042815036824635
	14	+0,0018476468830563
	15	+0,0002303857635232
	16	-0,0002519631989427
	17	+0,0000393473203163
	18	
<b>Désignation</b>	<b>n</b>	$\sqrt{2} * h(n)$
Daubechies 20	0	+0,0266700579005473
	1	+0,1881768000776347
	2	+0,5272011889315757
	3	+0,6884590394534363
	4	+0,2811723436605715
	5	-0,2498464243271598
	6	-0,1959462743772861
	7	+0,1273693403357541
	8	+0,0930573646035547
	9	-0,0713941471663501
	10	-0,0294575366281399
	11	+0,0332126710593612
	12	+0,0036065535669870
	13	-0,0107331754833007
	14	+0,0013953517470688
	15	+0,0019924052951925
	16	-0,0006858566949564
	17	-0,0001164668551285
	18	+0,0000935886703202
19	-0,0000132642028945	

ondelette bi orthogonales :

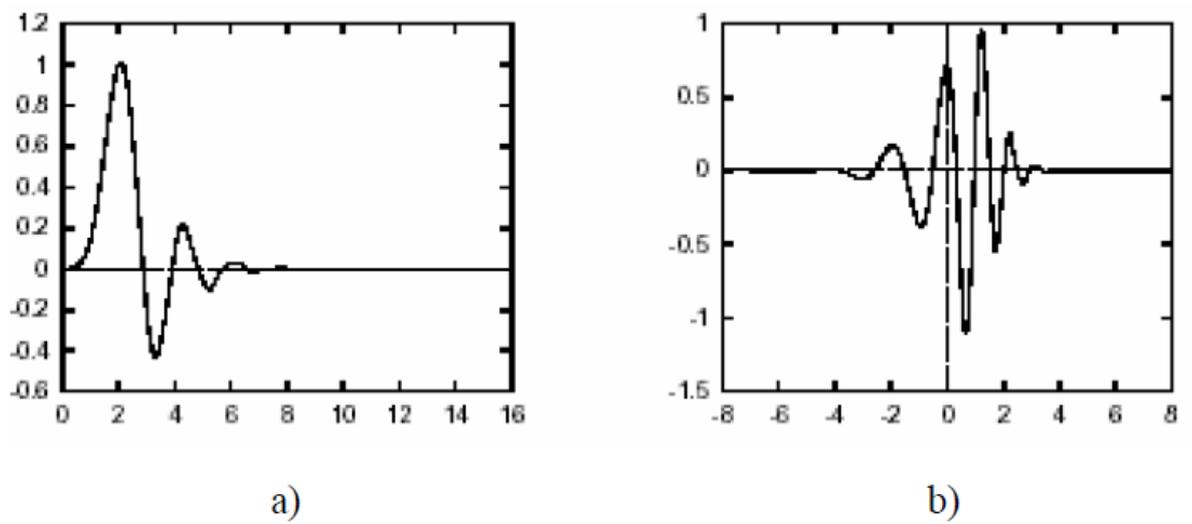
Désignation	$n$	$h(n)$	$\tilde{h}(n)$
5/3	-2	-0.125	
	-1	0.25	0.25
	0	0.75	0.5
	1	0.25	0.25
	2	-0.125	
5/7	-3		-0.010714285714
	-2	-0.05	-0.053571428571
	-1	0.25	0.260714285714
	0	0.60	0.607142857142
	1	0.25	0.260714285714
	2	-0.05	-0.053571428571
9/7	3		-0.010714285714
	-4	0.026748757411	
	-3	-0.016864118443	-0.045635881557
	-2	-0.078223266529	-0.028771763114
	-1	0.266864118443	0.295635881557
	0	0.602949018236	0.557543526229
	1	0.266864118443	0.295635881557
	2	-0.078223266529	-0.028771763114
	3	-0.016864118443	-0.045635881557
4	0.026748757411		

## Les fonctions d'échelle et d'ondelettes Daubechies

**Fig 1:** Fonctions d'échelle (a) et ondelette (b) de Daubechies pour  $N = 2$ **Fig 2:** Fonctions d'échelle (a) et ondelette (b) de Daubechies pour  $N = 4$



**Fig 3:** Fonction d'échelle (a) et ondelette (b) de Daubechies pour  $N = 6$



**Fig 4 :** Fonctions d'échelle (a) et ondelette (b) de Daubechies pour  $N = 8$