

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT AUTOMATIQUE

## MEMOIRE DE MAGISTER

en Automatique

Option **Traitement d'Images et  
Reconnaissance de Formes**

Présenté par

**AMROUN Fazia**

Ingénieur UMMTO

## EXTRACTION DE LA COMPOSANTE TEXTUREE D'UNE IMAGE

Mémoire soutenu le : 21/03/2013

devant le jury d'examen composé de :

**AHMED OUAMER Rachid**  
**DIAF Moussa**  
**HAMMOUCHE Kamel**  
**LAHDIR Mourad**

Professeur à l'U.M.M.T.O.  
Professeur à l'U.M.M.T.O.  
M.C.A. à l'U.M.M.T.O.  
M.C.A. à l'U.M.M.T.O.

**Président**  
**Rapporteur**  
**Examineur**  
**Examineur**

## Avant propos

Le travail que nous présentons dans ce mémoire a été effectué au sein de laboratoire « Robotique et vision » du département automatique, Faculté de Génie Electrique et Informatique, Université Mouloud Mammeri, Tizi-Ouzou.

J'exprime mes remerciements et ma gratitude à mon directeur de mémoire, Monsieur **DIAF Moussa**, professeur à l'Université Mouloud Mammeri, pour l'aide et le soutien et les conseils qu'il m'apportés tout au long de ce travail.

Nos vifs remerciements vont aussi à Monsieur **AHMED OUAMER Rachid** professeur à l'Université Mouloud Mammeri, pour avoir fait l'honneur de présider le jury de ce mémoire et l'intérêt qu'il a apporté à ce travail.

Monsieur **HAMMOUCHE Kamel**, maître de conférences classe A à l'Université Mouloud Mammeri a bien voulu participer au jury de ce mémoire, qu'il trouve, ici, l'expression de notre profonde gratitude pour l'intérêt qu'il a apporté à notre travail.

Nous tenons à remercier Monsieur **LAHDIR Mourad**, maître de conférences classe A de l'université Mouloud Mammeri, pour avoir bien voulu faire partie de du jury de ce mémoire.

# Sommaire

## Introduction générale

### Chapitre 1 : Méthodes d'Analyse de texture

1. Introduction.....	3
2. Analyse de texture.....	4
3. Définition de la texture.....	4
3. Types de textures .....	5
4. Méthodes d'Analyse des textures .....	6
4.1 Méthodes structurelles .....	7
4.2 Méthodes statistiques .....	8
4.2.1 Méthodes d'ordre un .....	8
4.2.2 Matrices de cooccurrences.....	9
4.2.3 Longueurs de plage .....	11
4.2.4 L'autocorrelation .....	12
4.3.5 Méthodes fondées sur un modèle .....	13
4.3.5.1 Le modèle autorégressif.....	13
4.3.5.2 Les champs de Markov.....	13
4.3.5.3 Les Champ de Gibbs .....	14
4.3.5.4 Un modèle de segmentation.....	15
4.2 Les méthodes basées sur l'étude des formes .....	16
4.3 Les méthodes fréquentielles .....	17
4.4 Les méthodes fractales.....	18
5. Conclusion .....	18

### Chapitre 2 : Décompositions parcimonieuses morphologiques

#### Introduction

1.Terminologie .....	
2. Introduction à la parcimonie .....	
3. Le principe de la parcimonie .....	
4. Problématique scientifique .....	
5. Analyse harmonique computationnelle.....	
6. Redondance et Diversité morphologique.....	
6.1 Redondance .....	
6.2 Diversité morphologique.....	
7. Dictionnaires .....	
7.1 Des bases orthonormales aux dictionnaires redondants.....	
7.2 Choix du dictionnaire.....	

7.3 Dictionnaires adaptatifs .....	
8. Algorithmes de décomposition parcimonieuse .....	
8.1 Matching Pursuit .....	
8.2 Orthogonal matching pursuit.....	
8.4 Poursuite de base.....	
8.5 Analyse en composantes morphologiques.....	
8.6 Différents types de seuillage .....	
8.6 .1 Seuillage brut ou dur "hard thresholding" .....	
8.6.2 Seuillage doux ou "soft thresholding" .....	
8.6.3. Seuillage doux alterné.....	
Conclusion	

### **Chapitre 3 : Extraction de la composante texturée**

1. Introduction.....	25
2. Séparation de composantes basée sur l'Analyse en Composantes Morphologiques .....	25
2.1. Analyse en Composantes Morphologiques.....	26
2.2 .Extraction de la composante texturée d'une image.....	27
3. Seuillage .....	29
3.1. Seuillage linéaire(SL). .....	29
3.2. Seuillage exponentielle(SE) .....	30
3.4. Seuillage Mean of Max(SMoM) .....	30
4. Choix du dictionnaire .....	31
5 .Transformée en ondlettes .....	31
5.1. Transformée en ondelette non décimée (TOD) .....	32
6. Transformée en cosinus discrète.....	36
6.1. Transformée en cosinus locale.....	37
7. Tests et Résultats .....	37
8. Conclusion .....	44

#### **Conclusion générale**

#### **Bibliographie**

# **Introduction Générale**

La texture joue un rôle très important dans l'identification et l'extraction des informations thématiques contenues dans l'image. L'analyse des textures est un vaste domaine dont le but est de cerner la nature propre d'une texture, soit via des algorithmes de classification, soit via des algorithmes de synthèse visant la création d'une texture, visuellement similaire à la texture originale. Ces dernières années, des travaux du domaine se sont orientés vers la décomposition morphologique d'une image et l'extraction de ses composantes principales.

La décomposition morphologique d'une image en ses constituants, et ses méthodes ont un impact dans beaucoup d'applications comme l'imagerie médicale et biomédicale l'imagerie astronomique, imagerie aérienne etc. Dans ce mémoire, Nous proposons d'aborder ce problème sous l'angle de la séparation d'une image en une superposition linéaire de deux composantes morphologiques ayant des structures caractéristiques, la texture et le contour.

Nous concentrerons sur une modélisation particulière de l'image : le cas du modèle de mélange. Il s'agira alors de modéliser l'image comme un mélange linéaire de signaux élémentaires appelés « sources ». L'analyse de ces données ainsi modélisées conduit à la résolution du problème de séparation. La technique proposée repose sur une décomposition en composantes morphologiques, basée sur la représentation parcimonieuse (éparse, économe ou creuse). L'hypothèse de base est de considérer que pour toute structure de l'image, il existe un dictionnaire de fonctions caractéristiques, à même de reconstituer cette structure particulière. L'image est ainsi décomposée sur un méta-dictionnaire constitué d'un sous-dictionnaire pour représenter la texture et d'un autre sous-dictionnaire pour récupérer la structure géométrique. L'algorithme d'analyse en composantes

morphologiques (MCA) permet de résoudre de façon itérative le problème d'optimisation correspondant à la recherche de ces composantes.

C'est ainsi que dans le cadre de ce mémoire, nous nous sommes fixés comme objectif, l'étude et l'application de l'analyse en composantes morphologiques pour chercher à décomposer l'image en les deux composantes texturée et régulière. Un logiciel a été développé.

Ainsi, dans le premier chapitre de ce mémoire, une présentation de la texture en termes de définition, de méthode et un exposé sur l'état de l'art sur l'analyse et synthèse de texture s'avèrent nécessaires.

Dans le second chapitre, nous présentons les concepts élémentaires sur lesquels est fondé le développement du Chapitre 3. On attachera une importance particulière aux termes de la parcimonie. La diversité morphologique et l'analyse harmonique computationnelle sur laquelle la décomposition est fondé sont présentés. Enfin, on verra comment les algorithmes de type glouton permettent de résoudre le problème de la décomposition d'image.

Dans le troisième chapitre, il s'agit de l'application proprement dite. Elle est dédiée à la décomposition d'image. Les approches utilisées, deux dictionnaires appropriés sont définis pour chaque composante de l'image ainsi un algorithme de décomposition d'image MCA. Les résultats obtenus sont analysés et commentés.

Ce travail est terminé par une conclusion générale et des perspectives en vue de son amélioration.

# **Chapitre 1**

## ***Méthodes d'Analyse de textures***

### **1. Introduction**

Ce chapitre présente l'état de l'art sur les méthodes les plus communément utilisées en analyse de la texture.

L'étude des textures rencontrées dans les images est un thème qui suscite beaucoup d'intérêt dans la communauté du traitement du signal et des images. Il s'agit d'un domaine de recherche encore largement ouvert. Il y a un nombre très important de travaux qui ont été publiés récemment [1] [2]. Il existe une grande diversité dans les approches utilisées pour décrire ou identifier une texture. Certaines méthodes s'appuient sur des modèles stochastiques ou de prédiction linéaire [3]. D'autres approches sont basées sur des calculs statistiques ou des transformations orthogonales [4] [5].

## 2. Analyse de texture

Analyse d'images est un terme générique englobant entre autres :

### a. Traitement d'images :

- De-bruitage, suppression de flou, rehaussement de contraste

### b. localisation, extraction des éléments intéressants : événements exceptionnels (contrôle de qualité) ;

- Tumeur (imagerie médicale : radio, RMN, scanner, etc.).

### c. Détection de bords, segmentation :

- frontières physiques des différentes régions dans l'image
- frontières plus abstraites, non nécessairement visibles.

## 3. Définition de la texture

Bien que la notion de texture soit naturelle pour l'être humain, elle résiste depuis longtemps à toute tentative de définition. On peut s'en approcher en disant qu'une texture est une zone de l'image qui présente certaines caractéristiques d'homogénéité qui la font apparaître comme une zone unique. Nous pouvons aussi la décrire comme étant un ensemble de primitives de taille et de forme variables, présentant une organisation spatiale particulière[6]. La texture est définie aussi comme étant une région d'une image pour laquelle il est possible de définir une fenêtre de dimensions minimales, telle qu'une observation au travers de celle-ci se traduit par une perception (impression) visuelle identique pour toutes les translations possibles de cette fenêtre à l'intérieur de la région considérée.

## 3. Types de textures

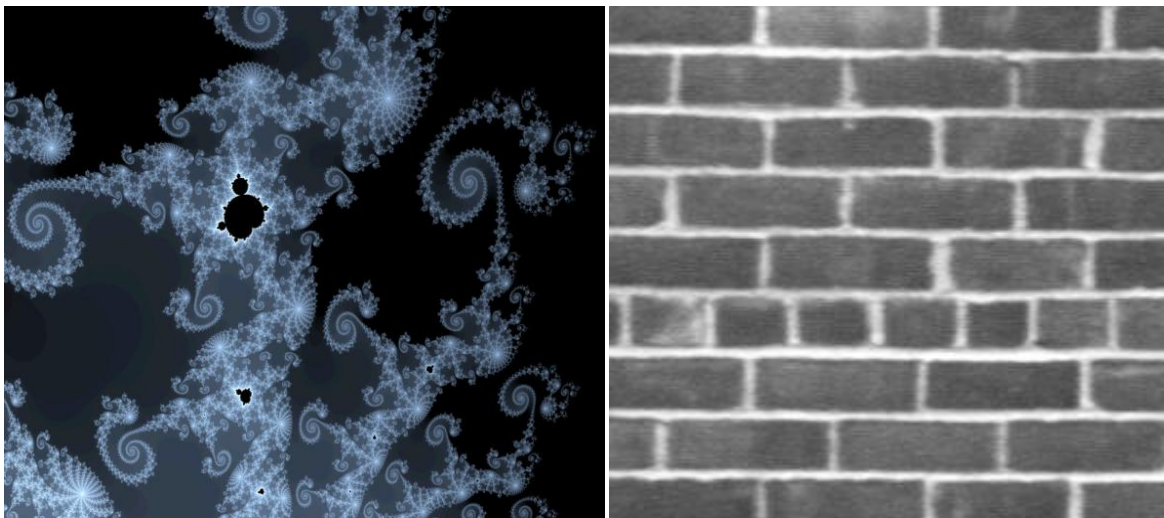
Il existe deux types de textures: les textures périodiques (Fig.1) définies par la répétition d'un motif de base et les textures dites aléatoires qui ont un aspect anarchiques (Fig.2) mais qui apparaissent homogènes vis-à-vis de leur environnement. Brodatz a établi un catalogue de texture à des fins de synthèse d'images texturées .Ce catalogue constitue aujourd'hui une référence en matière de base de test pour les chercheurs travaillant sur la

texture. Il s'agit de 112 textures (sable, herbe, bulles, eau, bois, etc.) ayant chacune seize variantes différentes. Ces images sont classiquement utilisées pour valider les différentes méthodes d'analyse de texture.

➤ *Cas déterministe (périodique)*

La texture peut être définie comme la répartition spatiale *régulière* d'un motif géométrique *constant* (appelé *texton*). On parle dans ce cas de texture déterministe.

Une région texturée est alors constituée par la répétition d'un motif original selon un réseau bidimensionnel caractérisé par une direction et une période particulière. La description du motif élémentaire, les dimensions du réseau et son orientation suffisent alors à décrire complètement la texture. Cette définition ne convient qu'à des textures parfaitement régulières que l'on ne rencontre que rarement dans la réalité. Il est évident que l'image de la trame d'un textile et celle d'un champ de blé ne revêtent pas le même caractère de régularité géométrique dans la disposition des motifs texturaux.



**Fig1. 1:**Exemples de textures déterministes [7]

➤ *Cas stochastique (anarchique)*

A l'inverse du cas précédent, on peut considérer des textures composées de motifs différents appartenant à une population dont seules les propriétés statistiques sont définies et dont la répartition spatiale suit

également une grille irrégulière. La définition de la texture est alors totalement stochastique [8].

Une texture pourra alors être considérée comme une réalisation d'un champ aléatoire bidimensionnel.



**Fig 1.2:** Exemples de textures stochastiques naturelles [9].

#### **4. Méthodes d'Analyse des textures**

L'objectif principal de l'analyse de la texture est la détermination de signatures texturales permettant une meilleure caractérisation de l'image autrement dit c'est la formalisation des descriptifs de la texture par des paramètres mathématiques qui serviraient à l'identifier. L'absence d'une définition formelle et précise de la texture a conduit à l'élaboration d'une multitude de méthodes d'analyse de textures. Le choix d'une méthode de caractérisation de textures est étroitement lié à l'application visée et à la nature de la texture à discriminer. Une multitude de méthodes, de variantes et de combinaisons de méthodes ont déjà été proposées dans la littérature. On les distingue :

##### **4.1 Méthodes structurelles**

On considère que les textures sont constituées de motifs de base se répétant dans des directions de l'espace de façon quasi-régulière. Dans les images étudiées, on considère donc l'existence de primitives. Une description structurale d'une texture implique la recherche des motifs

élémentaires, leur description, puis, la détermination des règles conditionnant leur position. Elles sont donc applicables sur les textures aléatoires.

Plusieurs approches peuvent être regroupées en un ensemble de méthodes heuristiques, dites méthodes structurelles classiques [10] [11], pour lesquelles on cherche à détecter et caractériser des primitives, puis, à trouver des règles de placement, (méthodes « bottom-up ») ou inversement (méthodes « top-down »). Elles s'appuient principalement sur le traitement du signal, la topologie et la géométrie. L'avantage des méthodes «bottom-up» est que l'on peut utiliser les techniques classiques de segmentation, comme la croissance de région, les seuillages, la détection de contours, etc., pour isoler les primitives, considérées comme des ensembles de pixels ayant des propriétés d'homogénéité communes (en niveau de gris, par exemple).

Ces propriétés (luminance, aire, taille, courbure, directionnalité, etc.) permettent la description d'un certain nombre de classes de primitives. Les règles de placement sont exprimées à l'aide de « vecteurs de régularité », de vecteurs de densités de primitives ou à partir des centres de gravité des classes.

D'autres approches, dites approches syntaxiques [12] [13] font appel à la théorie des grammaires qui permet d'engendrer des formes en appliquant un ensemble de règles de placements données à un petit nombre de symboles. Ces symboles peuvent être considérés comme des sous-primitives de base et doivent pouvoir reconstituer toute texture structurale. Mais, le formalisme de ces approches est souvent complexe. De plus, une texture peut être générée ou analysée par plusieurs grammaires. On limite donc leur utilisation à la génération de textures binaires ou à des applications très particulières.

Enfin, il existe une autre catégorie de méthodes structurelles dites approches ensemblistes. Ces approches sont principalement descriptives. Elles s'appuient sur des opérations de morphologie effectuées essentiellement sur des images binaires; il est donc nécessaire d'appliquer des prétraitements (seuillage, détection de contours) sur les images

originales. La description des primitives sera essentiellement géométrique avec des propriétés comme la taille, la forme, [14] [15].

## 4.2 Méthodes statistiques

Les méthodes statistiques étudient les relations entre un pixel et ses voisins et définissent des paramètres discriminants de la texture en se basant sur des outils statistiques. Généralement, elles sont utilisées pour caractériser des structures fines, sans régularité apparente. Plus l'ordre de la statistique est élevé et plus le nombre de pixels mis en jeu est important. Suivant la modalité des images à étudier, la signature la plus discriminante de la texture est à rechercher soit dans des méthodes qui exploitent directement les propriétés statistiques de la texture (matrices de cooccurrences, matrice de longueurs de plages, matrice de voisinage, fonction d'autocorrelation, modèle de Markov, modèle autorégressif, modèles issues de la morphologie mathématique), soit dans des méthodes qui exploitent les propriétés statistiques à partir d'un plan transformé dans lequel on réécrit l'image de texture (densité spectrale, méthode des extremas locaux, méthodes de transformation de Fourier, Karhunen Loeve, etc) voir [16][17][18].

### 4.2.1 Méthodes d'ordre un

L'analyse des régions d'une image par une méthode d'ordre un s'effectue directement à partir des niveaux de gris des pixels de l'image. Chaque pixel de la région sera donc représenté par une valeur commune à la région. Dans les calculs qui suivent, on étudiera l'image dans une fenêtre  $g$ . Cette fenêtre contient  $N$  pixels. Les paramètres d'étude les plus souvent utilisés sont la moyenne, la variance, l'aplatissement, la dissymétrie et le rapport signal-bruit.

- *Moyenne*

La moyenne donne le niveau de gris moyen dans la fenêtre :

$$MOY = \frac{1}{N} \sum_{i,j} g\{i,j\} \quad (1.1)$$

- *Variance*

La variance donne l'écart carre moyen des niveaux de gris par rapport a la moyenne :

$$VAR = \frac{1}{N} \sum_{i,j} (g(i,j) - MOY)^2 \quad (1.2)$$

- *Aplatissement*

$$SKEW = \frac{1}{N} \sum_{i,j} (g(i,j) - MOY)^3 \quad (1.3)$$

- *Dissymétrie*

La Dissymétrie caractérise la forme du sommet de l'histogramme des niveaux de gris de l'image. Plus celle-ci est faible, plus le sommet est arrondi.

$$KURT = \frac{1}{N} \sum_{i,j} (g(i,j) - MOY)^4 \quad (1.4)$$

- *Rapport signal-bruit*

Le rapport signal-bruit rend compte de l'hétérogénéité d'une texture.

$$SNR = \frac{MOY}{\sqrt{VAR}} \quad (1.5)$$

#### 4.2.2 Matrices de cooccurrences

Les matrices de cooccurrences sont une méthode d'ordre deux. En effet, chaque résultat met en valeur les relations entre deux pixels. Les matrices de cooccurrences représentent donc les dépendances spatiales des niveaux de gris.

- *Construction*

Pour une image contenant  $G$  niveaux de gris, la matrice de coocurrence est une matrice  $M$  de taille  $G * G$ . Elle est construite de cette sorte :

1. On fixe une distance  $d$  et une direction  $\theta$ .
2. Initialisation de la matrice  $M(d, \theta)$  a zéro.
3. Pour chaque pixel  $P_0$  , on fait correspondre le pixel  $P_1$  situé a la distance  $d$  de  $P_0$  dans la direction  $\theta$  . Soient alors  $ndg_0$   $ndg_1$  les niveaux de gris dans les cases  $P_0$  et  $P_1$  . On incrémente donc la case  $(ndg_0, ndg_1)$  de la matrice  $M(d, \theta)$

La formulation exacte de  $M(d, \theta)$  est :

$$M(d, \theta) = \frac{\#(x,y),(x',y')|(x,y)R(x',y'),I(x,y)=i,I(x',y')=j}{\#(x,y),(x',y')|(x,y)R(x',y')} \quad (1.7)$$

où la relation  $(x,y)R(x',y')$  est vraie si les pixels  $(x,y)$  et  $(x',y')$  sont distants de  $d$  dans la direction  $\theta$ .

- *Matrice symétrique associée*

A la matrice  $M(d, \theta)$ , on peut associer :

$$S_\theta(d) = \frac{1}{2} [M(d, \theta) + M(d, \theta + 180)] \quad (1.8)$$

Cette matrice est symétrique. Si les textures de l'image sont grossières, les valeurs de la matrice sont concentrées sur la diagonale. Si par contre les textures sont fines, les valeurs de la matrice sont dispersées.

- *Extraction des paramètres*

L'interprétation des matrices de cooccurrences demande d'en extraire des paramètres. Ceux-ci sont nombreux et calculés directement sur les matrices. On compte parmi eux l'énergie, la corrélation, l'inertie, l'entropie.

- *Efficacité de la méthode*

La méthode des matrices de coocurrences, bien que donnant de bons résultats d'analyse est peu adaptée à une étude détaillée de la texture car elle demande beaucoup de calculs. De nombreux articles utilisent les matrices de coocurrences pour identifier, analyser et classer les textures ou bien pour segmenter une image selon ses textures [19][20].

Enfin, on peut noter qu'une alternative aux matrices de coocurrences en plus rapide existe. Ce sont les matrices de caractéristiques statistiques (voir [1]). La première étape de cette méthode est de calculer la différence entre l'image étudiée et sa translatée. On en ressort ensuite directement le

contraste, la covariance et la dissimilarité de l'image ou d'une région de l'image.

### 4.2.3 Longueurs de plage

Les longueurs de plage (ou isosegments) est une méthode d'ordre supérieur. On appelle une plage un ensemble de pixels consécutifs et dans une direction donnée ayant le même niveau de gris. La longueur d'une plage est alors le nombre de pixels pour une plage donnée.

L'objet qui caractérise les textures dans cette méthode est alors un ensemble de matrices  $P_\theta = (p_\theta(i; j))$ . Chaque case  $(i; j)$  de la matrice  $P_\theta$  contient le nombre de plages de longueur  $j$  et de niveau de gris  $i$  dans la direction  $\theta$

Tout comme pour les méthodes vues précédemment, les longueurs de plages peuvent être étudiées sur toute l'image ou bien sur des régions précisées. Il est pour cela nécessaire d'en extraire quelques attributs :

Soient  $L$  le nombre de niveaux de gris dans l'image et  $n_\theta$  la longueur de corde maximale dans la région étudiée et dans la direction  $\theta$ .

- nombre de longueurs de plage :

$$SLP = \sum_{i=0}^{L-1} \sum_{j=1}^{n_\theta} p_\theta(i, j) \quad (1.9)$$

- proportions de petites plages :

$$RF1 = \frac{1}{SLP} \sum_{i=0}^{L-1} \sum_{j=1}^{n_\theta} \frac{p_\theta(i, j)}{j^2} \quad (1.10)$$

- proportions de grandes plages :

$$RF2 = \frac{1}{SLP} \sum_{i=0}^{L-1} \sum_{j=1}^{n_\theta} j^2 p_\theta(i, j) \quad (1.11)$$

- hétérogénéité des niveaux de gris :

$$RF3 = \frac{1}{SLP} \sum_{i=0}^{L-1} \left( \sum_{j=1}^{n_\theta} p_\theta(i, j) \right)^2 \quad (1.12)$$

- hétérogénéité des longueurs de plage :

$$RF4 = \frac{1}{SLP} \sum_{j=1}^{n_\theta} \left( \sum_{i=0}^{L-1} p_\theta(i, j) \right)^2 \quad (1.13)$$

- pourcentage de plages

$$RF5 = \frac{SLP}{K} \quad (1.14)$$

Avec  $K$  le nombre total de pixels dans la région.

Tout comme pour les matrices de cooccurrences, cette méthode permet une bonne caractérisation des textures des régions. Cependant, elle reste relativement lente et est efficace sur des images comportant peu de niveaux de gris. Elle est ainsi peu rencontrée dans les articles récents cependant elle est se retrouve dans des états de l'art ou bien des études comparatives [21] [22].

#### 4.2.4 L'autocorrelation

Une autre méthode simple de description des textures est l'autocorelation. Pour une image  $F$ , au point  $(i, j)$ , elle se calcule par :

$$A(\varepsilon, \eta; i, j) = \frac{\sum_{m=i-W}^{i+W} \sum_{n=j-W}^{j+W} F(m, n) F(m-\varepsilon, n-\eta)}{\sum_{m=i-W}^{i+W} \sum_{n=j-W}^{j+W} (F(m, n))^2} \quad (1.15)$$

Pour caractériser les résultats, il est par exemple possible de calculer les moments seconds :

$$T(i, j) = \sum_{\varepsilon=-T}^T \sum_{\eta=-T}^T \varepsilon^2 \eta^2 A(\varepsilon, \eta; i, j) \quad (1.16)$$

D'une manière générale, la fonction d'autocorelation diminue d'autant plus lentement lorsque  $\varepsilon$  et  $\eta$  grandissent que la texture est grossière. [4] présente cette méthode et [23] la compare à d'autres méthodes dont les coocurrences.

#### 4.3.5 Méthodes fondées sur un modèle

Les méthodes fondées sur un modèle décrivent la texture par un modèle probabiliste. Celle-ci est alors caractérisée par les paramètres de ce modèle [24] [25].

##### 4.3.5.1 Le modèle autorégressif

###### Présentation

Le modèle autorégressif (AR) considère une interaction entre l'intensité de chaque pixel de l'image et la somme pondérée des intensités de ses voisins. En prenant l'image  $I$  comme un champ aléatoire, le modèle AR se définit par :

$$I(x) = \mu + \sum_{r \in \mathcal{D}} \theta(r) I(x+r) + \varepsilon(x) \quad (1.17)$$

Ou  $x$  est un point de l'image,  $D$  définit un voisinage,  $\varepsilon(x)$  est une variable Gaussienne de moyenne nulle,  $\mu$  est le biais (inutilisée pour la segmentation) et les  $\theta(r), r \in D$  sont les paramètres du modèle.

Les différentes textures sont caractérisées par les différentes dépendances de voisinage, elles mêmes représentées par les différents paramètres du modèle. Ces paramètres se retrouvent le plus souvent en utilisant les moindres carrés.

#### 4.3.5.2 Les champs de Markov

Les champs de Markov ne sont pas un outil d'étude de la texture a proprement parler. Ils sont plutôt un outil permettant de segmenter une image. Cependant, beaucoup de méthodes de segmentation d'images texturées sont fondées sur les champs de Markov[26].

##### ➤ Champs de Markov

Ici, chaque pixel est appelé un site et  $S$  est l'ensemble des sites de l'image  $N$  est le nombre de sites de  $S$  (correspond au nombre de pixels de l'image).

Un voisinage  $V$  est défini. Il doit respecter  $s \in V(s)$

$$s \in V(t) \Leftrightarrow t \in V(s) \quad (1.18)$$

Nous notons  $X = \{X_1, \dots, X_N\}$  une famille de variable aléatoires sur  $S$ . chaque variable aléatoire  $X_i$  est a valeurs dans  $\Omega$  (ici les niveaux de gris de l'image), l'espace des états. L'événement  $X_i = x_i$ ,  $x_i \in \Omega$  signifie que la variable aléatoire  $X_i$  prend la valeur  $x_i$ . De même l'événement  $X = x, x \in \Omega^N$  signifie  $(X_1 = x_1, \dots, X_N = x_n)$ . ceci revient a instancier tous les listes, c'est a dire à poser une image dans son support (une matrice).

$(X, \Omega^N, P)$  est appelée alors champs de Markov par rapport au système de voisinage  $V$  si il est vérifié :

$$\begin{cases} \forall x \in \Omega^N, P(X = x) > 0 \\ P(X_S = x_S | X_r = x_r, r \in S_{-\{s\}}) = P(X_S = x_S | X_r = x_r, r \in V(s)) \end{cases} \quad (1.19)$$

La deuxième propriété traduit le fait que l'intensité de chaque pixel n'est liée Qu'à l'intensité de ses voisins est la propriété Markovienne.

### 4.3.5.3 Les Champ de Gibbs

Reprenons les mêmes notations que pour les champs de Markov.  $(X, \Omega^N, P)$ , est un champs de Gibbs induit par la fonction d'énergie  $U$  si  $P(X)$  est de la forme :

$$P(x) = \frac{\exp\{-U(X)\}}{\sum_{X_i \in \Omega^N} \exp\{-U(X)\}} \quad (1.21)$$

Si le champ  $(X, \Omega^N, P)$  est de plus un champ de Markov par rapport au système de voisinage  $V$ , alors sa distribution  $P(X)$  est une distribution de Gibbs de portée bornée. Pour un  $X$  donnée, la distribution s'écrit donc :

$$P(x) = \frac{\exp\{-U(X)\}}{Z} \quad (1.22)$$

Avec  $Z = \sum_{X_i \in \Omega^N} \exp\{-U(X)\}$  une constant de normalisation appelé fonction de partition.  $U(X)$  est la fonction d'énergie elle est définie par :

$$U(X) = \sum_{c \in \mathcal{C}} V_c(X_s, s \in c) \quad (1.23)$$

Ou  $\mathcal{C}$  est l'ensemble des cliques (ici les pixels mutuellement voisins)liées au système de voisinage.  $V_c$  est une fonction de potentiel [27].

### 4.3.5.4 Un modèle de segmentation

Nous allons maintenant étudier comment segmenter l'image a partir d'un champ de Gibbs et en utilisant une méthode Bayésienne. Nous noterons ici  $X$  l'ensemble des classes et  $Y$  l'ensemble des valeurs de niveau de gris de l'image.  $x$  et  $y$  sont alors des instanciations du système.

La première étape consiste à définir un champ de Gibbs  $P$  pour les classes  $x$ . Ensuite, il est nécessaire de définir la loi de  $y$  étant donnée  $x$ ,  $q(y|x)$ .

La probabilité conjointe  $p(x, y)$  s'obtient alors par :

$$p(x, y) = P(x)q(y|x) \quad (1.24)$$

Enfin, l'image est segmentée en utilisant la méthode du maximum de vraisemblance :

$$\hat{x} = \operatorname{argmax}_{x \in \Omega} p(x, y) \quad (1.25)$$

Avec :

$$r(x|y) = \frac{p(x, y)}{u(y)} = \frac{P(x)q(y|x)}{u(y)} \quad (1.26)$$

$$q(y|x) = \prod_{s \in S} f(y_s|x_s) \quad (1.27)$$

Le terme  $P(x)$  dépend de la classe des voisins alors que le terme  $q(y|x)$  dépend du niveau de gris du point étudiée et de sa classe. Souvent, la fonction  $f$  est une Gaussienne centrée autour d'un niveau de gris.

Ce problème est difficilement résolvable par des méthodes d'optimisation classiques. On retrouve donc souvent des méthodes de recuit simulé. Des alternatives à ce modèle existent. On notera par exemple qui utilise les  $k$  plus proches voisins et qui se sert d'histogrammes. La deuxième propriété traduit le fait que l'intensité de chaque pixel n'est liée qu'à l'intensité de ses voisins. C'est la propriété Markovienne.

## 4.2 Les méthodes basées sur l'étude des formes

Se trouvent au croisement de la reconnaissance des formes, de la caractérisation de défauts et de l'analyse macro textural. Les régions texturales de l'image épousent des formes particulières et peuvent être caractérisées par des paramètres dits de formes. Parmi ces méthodes on trouve

### *Moment*

- Moments invariants qui sont souvent utilisés pour l'analyse des textures. Hu introduit les moments en faisant le parallèle avec leur utilisation en physique. En effet, les moments sont utilisés en physique pour décrire la répartition des masses dans un corps. En analyse d'image, on peut envisager la même démarche en associant le niveau de gris d'un point de l'image à la masse élémentaire en un point.
- Les moments de Zernike sont utilisés dans une approche pour la vidéo surveillance, D'autres moments ont été utilisés par les chercheurs, En analyse d'image, et reconnaissance des formes .Les plus intéressants sont les moments orthogonaux qui peuvent être obtenues par le biais des polynômes de Legendre ou de Zernike [28] [29] Leur utilisation en tant qu'attributs discriminants permet de mettre en évidence les données structurelles pertinentes. En plus, ces deux familles sont tout

à fait aptes à représenter ou à reconstruire des objets avec un minimum de redondance de données .

- Les descripteurs de Fourier-Mellin [30] ont été aussi utilisés récemment dans le cadre d'applications de recherche et d'indexation, des expériences récentes ont montré qu'un nombre suffisamment large (mais limité) de descripteurs s'avère suffisant pour reconstruire un objet à niveaux de gris avec une très bonne qualité.

#### *Attributs géométriques*

Les attributs géométriques classiques se calculent sur des régions  $R_i$  homogènes, au sens d'un certain critère, et connexes. Il s'agit, principalement, de l'aire, du périmètre, du diamètre, de la compacité, de la direction principale d'inertie, du nombre de connexité, etc., de chaque région  $R_i$ .

### **4.3 Les méthodes fréquentielles**

Ces méthodes, souvent utilisées en traitement du signal, permettent d'analyser une texture en identifiant les différentes fréquences qui la composent. Parmi elles, on retrouve notamment la transformée de Fourier, les filtres de Gabor ainsi que les ondelettes. De façon générale, la  $\lambda$  transformation linéaire d'une image  $f$  est obtenue en calculant la corrélation de  $f$  et d'une fonction  $\psi_\lambda$

$$\int_{-\infty}^{\infty} f(x, y) \bar{\psi}_\lambda(x, y) dx dy \quad (1.28)$$

avec  $\lambda$  un ensemble de paramètres propre à  $\psi$  et  $\bar{\psi}$  le complexe conjugué de la quantité en argument.

Ces méthodes, préservent à la fois les informations globales et locales. Elles sont bien adaptées aux signaux quasi périodiques, en effet, les textures sont des signaux quasi périodiques qui ont une énergie fréquentielle localisée. Ces méthodes permettent de caractériser la texture à différentes échelles. Analyser une texture de part son spectre fréquentiel nous donne beaucoup d'informations sur celle-ci.

- La transformée de Fourier (TF) permet de passer d'une représentation de l'image dans le domaine spatial à sa représentation dans le domaine fréquentiel. [31]

- Transformée Cosinus Discret (DCT) La Transformée en cosinus discrète ou TCD (de l'anglais : DCT ou Discrète Cosine Transform) est une transformation proche de la transformée de Fourier discrète (DFT). Le noyau de projection est un cosinus et génère donc des coefficients réels, contrairement à la DFT, dont le noyau est une exponentielle complexe et qui génère donc des coefficients complexes. La DCT possède une excellente propriété de "regroupement" de l'énergie : l'information est essentiellement portée par les coefficients basses fréquences.
- La transformée en Ondelettes, contrairement à Fourier, est beaucoup plus précise et riche en informations et en pertinence [31]. Pour faire face aux difficultés d'analyse des textures naturelles, l'outil espace-fréquence et temps-échelle qui semble le plus approprié est la transformée en ondelettes. Celle ci permet de prélever des paramètres locaux dans le domaine spatial, dans le domaine fréquentiel, à des résolutions multiples ainsi que des paramètres inter-échelles. De plus, son mode d'investigation, du global vers le détail, possède une similitude avec le système visuel humain.
- Filtre de gabord [32].

#### **4.4 Les méthodes fractales**

texture Ces méthodes permettent de synthétiser des images très proches de la réalité. En analyse de texture, la dimension fractale, qui est une mesure du degré d'irrégularité d'un objet, décrit une certaine propriété de la texture. Le modèle fractal est basé essentiellement sur l'estimation par des méthodes spatiales de la dimension fractale de la surface représentant les niveaux de gris de l'image [33].

#### **5. Conclusion**

Ce chapitre traitant de l'analyse de texture fait état d'un grand nombre d'approches et de méthodes existantes dans la littérature. Il est témoin d'un besoin croissant de connaissances plus approfondies sur la notion même de texture et sur le mécanisme humain de perception des textures. Au vu de cette synthèse, il apparaît que la profusion des approches et des méthodes

traitant de l'analyse de textures rend difficile le choix d'une méthode particulière pour une application donnée et, plus généralement, le besoin d'une hiérarchisation.

# Chapitre 2

## ***Décompositions parcimonieuses morphologiques***

### **Introduction**

La parcimonie est devenue un concept important en traitement de signal et des images, et son utilisation comme source d'a priori a connu un essor considérable ces dernières années notamment pour des tâches comme la compression, l'estimation, la restauration, la séparation de sources ou encore la détection, etc. Depuis quelques années, les représentations parcimonieuses de signaux et d'images font l'objet de développement important [1], [2] Le développement de ces techniques est issu des travaux de Stéphane Mallat [3] Actuellement la parcimonie connaît une grande utilisation dans plusieurs domaines des mathématiques appliquées, notamment en analyse harmonique appliquée et computationnelle, en mathématiques statistiques ou en théorie de l'apprentissage.

L'intérêt de la parcimonie a sans doute apparue grâce à l'émergence plus récente de l'échantillonnage compressé (compressed sensing) qui renouvelle la vision de Shannon de l'échantillonnage pour l'acquisition des signaux analogiques qui ne sont pas nécessairement à bande limitée [4]. Ainsi la parcimonie s'est imposé sur les protocoles d'échantillonnage. L'échantillonnage compressé a eu des conséquences directes dans plusieurs domaines scientifiques comme la théorie de l'information, les codes

correcteurs, l'imagerie médicale, la géophysique, l'imagerie astronomique [5] [6] [7] [8].

## 1. Terminologie

Nous présentons dans cette partie quelques notions utilisées dans la communauté de parcimonie [34].

### Notions :

**Atome :** Un atome est un signal élémentaire entrant comme élément constituant d'un signal plus complexe par superposition linéaire. exemple les sinusoides, les ondelettes, etc.

**Dictionnaire :** Un dictionnaire  $A$  est une collection d'atomes  $\{\varphi_\gamma\}_{\gamma \in \Gamma}$  indexés par  $\gamma$ , ou  $\Gamma$  est un ensemble dénombrable de cardinalité  $|\Gamma| = T$ . L'interprétation de l'indice  $\gamma$  dépend du dictionnaire : position pour Dirac, fréquence pour Fourier, position-échelle pour les ondelettes, position-durée-fréquence pour les paquets de cosinus etc.

### Analyse et synthèse

soit un signal  $x \in H$  et un dictionnaire  $A = \{\varphi_i, i \in I\}$ , il y a deux opérations : l'analyse et la synthèse. L'analyse est le calcul des produits scalaires avec tous les atomes du dictionnaire, et on s'intéresse aux coefficients  $\{\langle x, A_i \rangle\}$ . La synthèse est l'opération qui consiste à reconstruire  $x$  en superposant les atomes :  $x = \sum_{i \in I} a_i A_i$

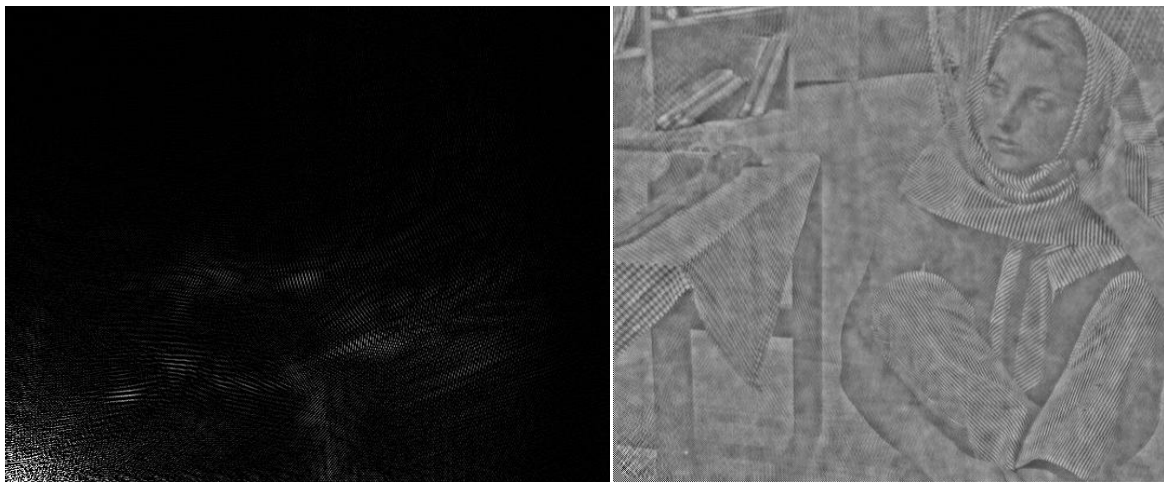
Si le dictionnaire choisi est une base orthonormée, alors l'analyse et la synthèse de  $x$ , il ya unicité de la décomposition. On a alors  $x = \sum_{i \in I} \langle x, A_i \rangle A_i$  . si  $A$  est redondant, c'est-à-dire qu'on dispose plus d'atomes que nécessaire pour représenter le signal, il existe une infinité de représentations possibles. La (fig.3.1) nous montre un exemple de dictionnaire la transformée en cosinus discrets(DCT), les deux opérations l'analyse et de synthèse.

## 2. Introduction à la parcimonie

La parcimonie est un principe consistant à n'utiliser que le minimum de causes élémentaires pour expliquer un phénomène (wikipedia). Dans le dictionnaire français c'est l'économie de petites quantités.

on définit un vecteur parcimonieux si la majorité de ses coefficients sont nuls. En traitement de signal, On dit qu'un signal est parcimonieux dans une base s'il peut être décrit par un faible nombre de coefficients non nuls

dans cette base. On cherche à construire la représentation parcimonieuse d'un signal à partir de fonctions définies dans un espace redondant, la décomposition la plus parcimonieuse. On obtient une décomposition originale d'un signal sur un dictionnaire, on fait intervenir le moins d'éléments possibles. Notons que les représentations parcimonieuses n'offrent qu'une représentation approximative du signal.



**Fig 2.1:** exemple d'un dictionnaire de base de cosinus(DCT), en haut image originale Barbara, en bas a gauche la DCT est l'opérateur d'analyse et en bas a droite la DCT inverse est l'opérateur de synthèse.

### 3. Le principe de la parcimonie

Les techniques classiques de représentation des signaux utilisent en général une description des composantes sur une base sur laquelle la

représentation du signal est unique (base de Fourier, base d'ondelettes orthogonales par exemple). Contrairement aux représentations parcimonieuses (sparse representations) consistent en la décomposition du signal sur un dictionnaire comprenant un nombre d'éléments (ou atomes) très supérieur à la dimension du signal. Cette décomposition va introduire dans la nouvelle représentation du signal un grand nombre de valeurs nulles, on parle de représentation parcimonieuse.

#### **4. Problématique scientifique**

Les problématiques scientifiques que posent ces représentations sont au nombre de trois :

1. La définition du critère de parcimonie est une étape importante de toutes les méthodes utilisant ces représentations.
2. Il est nécessaire de trouver des méthodes de calcul de  $x$ , connaissant le signal  $y$  et le dictionnaire  $D$ .
3. D'autre part, il faut aussi développer des techniques de construction du dictionnaire  $D$  en fonction de l'application à réaliser.

Ces trois aspects ont donné lieu à de nombreuses publications qui sont au coeur de la thématique des représentations parcimonieuses de signaux et d'images [9] [10] [11].

#### **5. Analyse harmonique computationnelle**

C'est une branche récemment avancées[35], permis le développement des ondelettes, des curvelets, des contourlets, des bandlets, des brushlets, des wave-atoms, des paquets d'ondelettes ou de cosinus, et plusieurs représentations fournissant une description parcimonieuse d'images qui a différents contenus géométriques.

#### **6. Redondance et Diversité morphologique**

##### **6.1 Redondance**

Le dictionnaire se compose alors de diverses fonctions, de natures différentes. Cette diversité est à l'origine de la notion de redondance. Lorsqu'on enrichit dictionnaire, l'avantage est que le dictionnaire aura de multiples représentations possibles[36][37] [38][39].

## 6.2 Diversité morphologique

En 2004 Starck *et al.*[40][41] ont introduit le concept de diversité morphologique dans le cadre d'un problème de décomposition d'images en composantes de contour et de texture. L'idée est de constater qu'il existe des ensembles de signaux élémentaires qui représentent parcimonieusement l'une et une seule des deux composantes (contour ou texture).

Le concept de diversité morphologique pousse l'idée de la redondance encore plus loin en supposant que le signal d'intérêt  $x$  peut être écrit comme un mélange fini de  $K$  composantes  $x_k$  qui sont morphologiquement distinctes,

$$x = \sum_{k=0}^K x_k$$

ou  $x_k$  est appelée composante morphologique, qui est parcimonieuse dans un dictionnaire dédié  $A_k$  avec les opérateurs d'analyse et de synthèse rapides associés.

## 7. Dictionnaires

### 7.1 Des bases orthonormales aux dictionnaires redondants

A l'époque les représentations classiques des signaux utilisent une description des composantes sur une base sur laquelle la représentation du signal est unique comme par exemple base de Fourier, base d'ondelettes orthogonales ;

Soit  $y$  un signal réel de taille  $N(y \in \mathbb{R}^N)$  la représentation du signal sur la base  $B$  s'écrit matriciellement :  $y = Bx$   $B$  est une base de  $\mathbb{R}^N$ , les vecteurs  $b_i \in \mathbb{R}^N$  sont des vecteurs orthogonaux,  $\forall i \neq j, b_i \perp b_j$ , de norme unitaire,  $\|b_i\|=1$ . Il y a unicité de  $x$ , décomposition de  $y$  sur la base  $B$ . La transformation est inversible (si  $B$  est inversible) et, dans ce cas, on a directement :  $B^{-1}y = B^{-1}Bx$  donc  $B^{-1}y = x$

Le signal peut se représenter d'une infinité de manières via les colonnes du dictionnaire, les atomes  $\gamma$ . Le signal est décomposé sous la forme d'une combinaison linéaire pondérée de fonctions élémentaires :

$$y = \sum_{\gamma \in \Gamma} x_\gamma a_\gamma$$

Où  $x_\gamma$  sont les coefficients de pondération des atomes  $a_\gamma$  et  $\Gamma$  l'ensemble fini d'indices représentant les fonctions élémentaires choisies.

## 7.2 Choix du dictionnaire

Le dictionnaire redondant global est choisi comme une union de sous-dictionnaires. Les arguments empruntés à l'analyse harmonique computationnelle nous permettent de choisir les sous-dictionnaires. Par exemple, les curvelets pour des images avec des contours curvilignes réguliers, les wave-atoms ou la DCT locale pour les textures localement oscillantes, ou les ondelettes vont être bien adaptées pour des structures isotropes, les ridgelets pour des singularités le long de lignes globales, etc. En fait, en choisissant un sous-dictionnaire plutôt qu'un autre, un a priori est implicitement injecté sur ce que sont les constituants élémentaires du signal-image.

## 7.3 Dictionnaires adaptatifs

Le dictionnaire adaptatif construit à partir des données sources elles-mêmes est utilisé dans des méthodes d'apprentissage. L'avantage est la grande liberté qu'offre l'adaptabilité pour représenter n'importe quel signal.

## 8. Algorithmes de décomposition parcimonieuse :

Dans cette partie, nous décrivons les algorithmes développés pour trouver le jeu de coefficients le plus parcimonieux, tout en respectant le critère qui a été choisi.

### 8.1 Matching Pursuit :

Le Matching Pursuit est le premier algorithme présenté puis étudié dans le cadre de la décomposition parcimonieuse. Le (MP) [42] est un algorithme glouton, connu pour la recherche de la solution optimale. L'algorithme MP permet de trouver une approximation sous-optimale.

Le principe du MP est de chercher dans un dictionnaire l'atome qui se rapproche le plus du signal, de retirer cet élément du signal, puis d'itérer jusqu'à obtenir l'approximation souhaitée.

considérons le problème général de la décomposition d'un signal  $y$  sur un dictionnaire  $A$  constitué d'un ensemble de vecteurs unitaires  $\{a_i\}_{i \in I}$ . Le dictionnaire est supposé redondant.

**Algorithme 1** : Matching Pursuit**ENTRÉES** : Le signal source  $y$ , le dictionnaire  $A$  et le seuil  $\rho$ Initialization :  $\hat{y}^{(0)} = 0, R_y^{(0)} = y$  et  $k = 1$ **Tant que**  $\|R_y^{(k-1)}\|^2 \geq \rho$  faire

- Recherche de l'atome le plus corrélé :  $\gamma_k = \operatorname{argmax}_\gamma |\langle R_y^{(k-1)}, a_\gamma \rangle|$
- Calcul du nouveau coefficient :  $x_{\gamma k} = |\langle R_y^{(k-1)}, a_{\gamma k} \rangle|$
- Mise à jour des données :
  1. De l'estimée :  $\hat{y}^{(k)} = \hat{y}^{(k-1)} + x_{\gamma k} a_{\gamma k}$
  2. Du résidu :  $R_y^{(k)} = y - \hat{y}^{(k)} = R_y^{(k-1)} - x_{\gamma k} a_{\gamma k}$
  3. Du vecteur parcimonieux :  $x[\gamma k] \leftarrow x_{\gamma k}$

**Fin Tant que**

L'intérêt majeur du Matching Pursuit est sa grande simplicité d'implémentation et sa relative rapidité d'exécution. Contrairement à d'autres algorithmes, il ne nécessite aucune inversion matricielle. Cependant, cette simplicité a un inconvénient : il peut falloir un grand nombre d'itérations pour converger vers une solution et aussi dans certains cas un atome déjà sélectionné peut à nouveau l'être.

**8.2 Orthogonal matching pursuit**

L'objectif de l'OMP [43] est de pallier la faille du MP, c'est-à-dire d'empêcher la sélection multiple d'un même atome. L'Orthogonal Matching Pursuit (OMP) se base sur le même principe que le MP, il sélectionne pas à pas les atomes les plus corrélés au signal pour tendre vers une approximation de la solution. La différence est dans la mise à jour des coefficients.

**Algorithme 2** : orthogonal Matching Pursuit**ENTRÉES** : Le signal source  $y$ , le dictionnaire  $A$  et le seuil  $\rho$ Initialization :  $\hat{y}^{(0)} = 0, R_y^{(0)} = y, A_0 = [.]$  et  $k = 1$ **Tant que**  $\|R_y^{(k-1)}\|^2 \geq \rho$  faire

- Recherche de l'atome le plus corrélé :  $\gamma_k = \operatorname{argmax}_\gamma |\langle R_y^{(k-1)}, a_\gamma \rangle|$
- Ajout du nouvel atome au sous-dictionnaire :  $A_k = [A_{k-1} \ a_\gamma]$
- Calcul des coefficients :  $x_k = A_k^+ y$  avec  $A_k^+ = (A_k^T A_k)^{-1} A_k^T$
- Mise à jour des données :
  4. De l'estimée :  $\hat{y}^{(k)} = A_k \cdot x_k$
  5. Du résidu :  $R^k = y - \hat{y}^{(k)}$

**Fin Tant que**

- 
- $A_k$ , la matrice contenant l'ensemble des atomes sélectionnés à l'itération  $k$  :  $A_k = [a_{\gamma_1} \dots a_{\gamma_k}]$
  - $x_k$ , le vecteur contenant uniquement les coefficients non-nuls qui ont été calculés jusqu'au pas courant :  $x_k = [x_{\gamma_1} \dots x_{\gamma_k}]$

La projection du signal sur l'ensemble des atomes sélectionnés jusqu'au pas courant se fait par le calcul de la pseudo-inverse  $A_k^+$ , cette méthodologie empêche la sélection d'un atome qui aurait déjà été sélectionné. Cela a l'avantage de surpasser le Matching Pursuit en termes de nombre d'itérations mais nécessite une inversion matricielle à chaque itération

**8.3 Meilleure base**

L'approche « meilleure base [2] considère le dictionnaire  $D$  comme la réunion de  $B$  bases distinctes, puis à rechercher (exhaustivement ou non) parmi ces bases celle qui offre a décomposition optimale selon le critère de parcimonie retenu.

**8.4 Poursuite de base**

L'approche « Poursuite de base » (Basis pursuit) [6] utilise des techniques de programmation linéaire pour minimiser la norme  $L_1$  de la décomposition. La

complexité de l'algorithme est importante, la solution est en général satisfaisante, sans toutefois atteindre le résultat qui aurait été obtenu avec la norme  $L^0$ . Mais l'utilisation de la norme  $L^0$  est un problème d'optimisation NP-complet.

### 8.5 Analyse en composantes morphologiques

Le désavantage de MP et OMP est de réaliser le processus d'analyse et la synthèse via deux algorithmes distincts. L'Analyse en composantes morphologiques (MCA) propose une alternative à cette décomposition en combinant les deux traitements en un seul au sein d'un unique algorithme. Le MCA repose sur une décomposition en composantes morphologiques, basée sur les représentations parcimonieuses, que nous exposons en détail au chapitre 3. Notons bien que l'algorithme MCA est un algorithme de seuillage itératif basée sur l'algorithme Blok coordinate Relaxation.

### 8.6 Différents types de seuillage

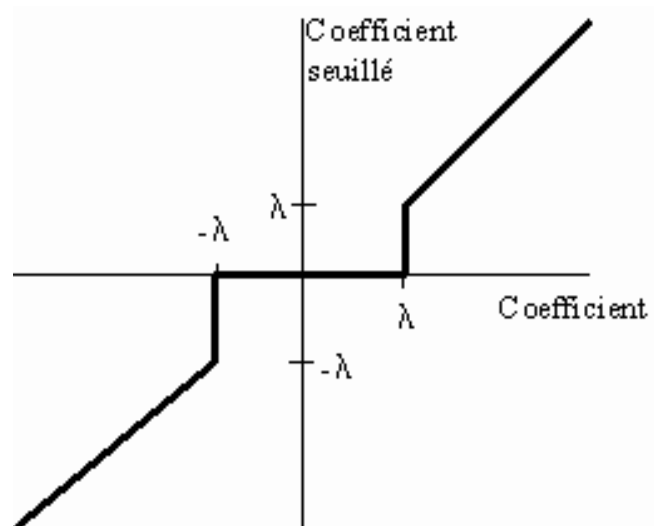
Il existe plusieurs types de seuillages. Nous définissons trois types de seuillage utilisées dans notre application, le seuillage dur, le seuillage doux et le seuillage doux alternée.

#### 8.6 .1 Seuillage brut ou dur "hard thresholding"

Le seuillage dur est celui qui est le plus "intuitif". On se fixe un seuil  $T > 0$ . On ne conserve que les coefficients d'ondelettes supérieurs à  $T$  et on met à zéro les autres, sa fonction mathématique est :

$$\theta(x) = \begin{cases} 0 & \text{si } |x| < T \\ x & \text{si } |x| \geq T \end{cases}$$

Nous aurons donc le seuillage suivant :

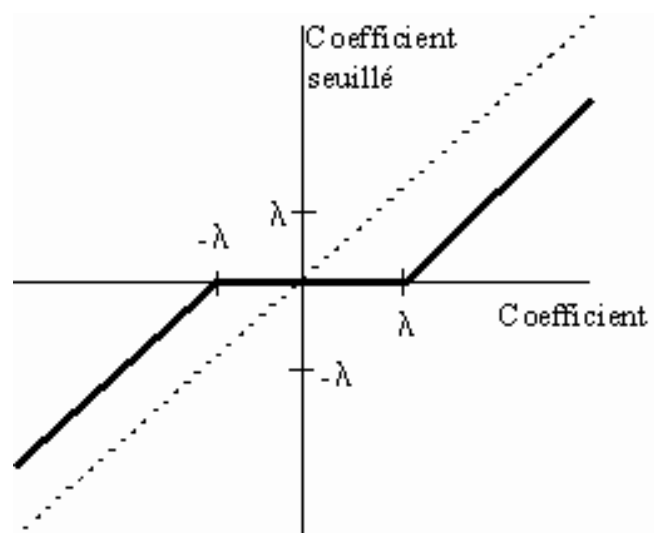


### 8.6.2 Seuillage doux ou "soft thresholding"

Dans le cas du seuillage dur, on met toujours à zéro les coefficients inférieurs à un seuil  $T$ . Par contre, pour ceux supérieurs à  $T$ , on atténue l'amplitude des coefficients par la valeur du seuil afin de s'assurer d'avoir enlever l'effet du bruit même pour les forts coefficients.

$$\theta(x) = \begin{cases} 0 & \text{si } |x| < T \\ x - \text{sign}(x)T & \text{si } |x| \geq T \end{cases}$$

Dans ce cas, sa fonction de seuillage est continue :



Le coefficient seuillé sera donc plus petit que le coefficient du signal. Ce type de seuillage garantit que le signal obtenu sera toujours plus régulier que le signal de départ.

### 8.6.3. Seuillage doux alterné

une méthode appelée aussi Blok coordinate Relaxation (BCR), lorsque le dictionnaire est constituée d'une union de base orthonormées .

#### L'algorithme 3:

**Donnée :** un signal  $x$ .

**Entrées :** dictionnaire  $A^T = [A_1^T \dots \dots \dots A_p^T]$ ,  $C^T = [C_1^T \dots \dots \dots C_p^T]$ .

Initialisation du compteur  $i:=0$  et des coefficients  $C^T = [C_1^T \dots \dots \dots C_p^T]$

#### Faire

1. sélection d'une base  $A_p$
2. soustraire les autres bases au signal :  $x_p := x - \sum_{p' \neq p} C_{p'}^T \cdot A_{p'}$
3. Mise à jours des coefficients de la base  $A_p$  par Seuillage doux :

$$C_p^{(i)} = \eta_\lambda (A_p \cdot x_p)$$

4. Les autres coefficients sont inchangés :

$$C_{p'}^{(i)} = C_{p'}^{(i-1)}, p' \neq p$$

Tant que la convergence n'est pas atteinte.

## Conclusion

Nous avons présenté dans ce chapitre les termes utilisée dans la communauté de parcimonie, le concept de parcimonie, et les représentations parcimonieuses, la redondance et la diversité morphologique ainsi les algorithmes de décomposition parcimonieuse. Les représentations parcimonieuses se sont révélées être une technique particulièrement performante en pratique. La recherche d'approximations parcimonieuses du signal s'est révélée fort utile dans les domaines de la compression, du débruitage, de la séparation de sources ou encore dans le domaine de l'analyse en composantes indépendantes.

# Chapitre 3

## *Extraction de la composante texturée*

### **1. Introduction**

Ce chapitre est consacré à l'application de l'approche adoptée. Nous proposons un modèle pour séparer la texture et le contour dans une image. Nous considérons que la texture et le contour représentent un signal bidimensionnel pour lequel chaque composante a des caractéristiques différentes. L'algorithme d'analyse en composantes morphologiques (MCA) basé sur une représentation parcimonieuse d'une image pour lequel deux dictionnaires appropriés sont définis chacun donnant une représentation parcimonieuse pour une composante du signal et pas pour l'autre.

### **2. Séparation de composantes basée sur l'Analyse en Composantes Morphologiques**

## 2.1 Analyse en Composantes Morphologiques

L'analyse en composantes morphologiques (MCA), permet de résoudre de façon itérative le problème de séparation en décomposant le signal dans un dictionnaire formé d'un enchainement de  $K$  sous dictionnaires à décomposition et reconstruction rapide.

Supposons que l'on observe un mélange linéaire bruité de  $K$  sources

$$x = \sum_{k=1}^K x_k + n \quad (3.1)$$

L'algorithme MCA (Fig 3.1) recherche la représentation la plus parcimonieuse du mélange dans le dictionnaire  $A = [A_1, \dots, A_k]$ , succession de sous dictionnaires  $k$ , ce qui correspond à minimiser l'équation

$$\lambda \sum_{k=0}^K \|\alpha_k\|_p^p + \frac{1}{2} \quad (3.2)$$

Où la norme  $\ell_p, 0 \leq p \leq 1$  force la parcimonie de la solution. Pour obtenir une séparation entre les différents dictionnaires, il ne faut pas garder tout le résidu dans  $A_m$ , car il contient initialement tout le signal. Il convient d'appliquer un seuil élevé, tel qu'uniquement des coefficients réellement associés à  $A_m$  soient gardés. Ceci peut être atteint en initialisant le seuil à  $\max_k \|A_k^T X\|_\infty$ , puis en le faisant descendre suffisamment lentement en bouclant sur les différents dictionnaires. Ainsi, si le seuil décroît suffisamment lentement et si les dictionnaires sont suffisamment décorrélés, à chaque itération, chaque dictionnaire va récupérer un groupe de coefficients associés aux structures les plus fortes du résidu dans ce dictionnaire, qui ne correspondront à aucune structure dans les dictionnaires concurrents.

**Donnée :** Un signal observé  $X$  contaminée par un bruit de variance  $\sigma^2$ .

**entrées:** Un dictionnaire  $A = [A_1, \dots, A_K]$  composé de  $K$  sous-dictionnaires, un nombre d'itérations  $N_{iter}$  et un seuil final  $\tau$ .

**Début**

Initialiser les composantes  $f_k^{(0)} = 0, k = 1, \dots, K$

Initialiser le résidu  $r^{(0)} = X$ .

Initialiser le seuil :  $k^* = \operatorname{argmax}_k \|A_k^T X\|_\infty$ , soit  $\lambda_0 = \max_{k \neq k^*} \|A_k^T X\|_\infty$

**For**  $t = 1$  **to**  $T_{iter}$  **do**

**for**  $k = 1$  **to**  $K$  **do**

Calculer les résidus marginaux  $r_k^{(t)} = r_k^{(t-1)} + f_k^{(t-1)}$

Mettre à jour les coefficients de la  $K^{ieme}$  composante en seuillant

$\alpha_k^{(t)} = \operatorname{thresh}_{\lambda_{t-1}}(A_k^T r_k^{(t)})$

Mettre à jour la  $K^{ieme}$  composante  $f_k^{(t)} = A_k \alpha_k^{(t)}$

Mettre à jour la solution  $f^{(t)} = \sum_{k=1}^K f_k^{(t)}$ .

Mettre à jour le résidu  $r^{(t)} = X - f^{(t)}$

Mettre à jour le seuil  $\lambda_t = \lambda_0 - t(\lambda_0 - \tau\sigma)/N_{iter}$

**Fin**

**Résultat:** Les composantes  $f_k$  de  $f$  dans  $A_k$

Fig.1 Algorithme de MCA

## 2.2 Extraction de la composante texturée d'une image

La séparation contour/texture constitue une étape majeure dans le traitement numérique d'image Depuis le livre d'Y. Meyer [46]. Il s'agit de les séparer en deux couches : la texture d'une part et le contour d'autre part. Extraire la texture est essentiel car la texture véhicule des informations thématiques contenues dans l'image. De nombreuses méthodes ont été proposées dans cette problématique [47][48][49][50] La plus connue étant celle de Vese et Osher [51]

Nous considérons l'image  $I$  comme un signal bidimensionnelle qui est composé d'un mélange de deux signaux de même taille.  $X_t$  contenant la texture et  $X_c$  contenant le contour. Pour résoudre ce problème nous utilisons l'algorithme d'analyse en composante morphologique. Le MCA permet une bonne séparation de caractéristiques contenues dans une image lorsque ces caractéristiques présentent des aspects morphologiques différents. Pour ce faire nous appuyons sur une représentation parcimonieuse définie à partir de deux dictionnaires appropriés uniquement pour chacune des caractéristiques.

- Modèle

Soit  $I \in \mathbb{R}^n$  une image étant une combinaison linéaire [51] [52] [53] de deux composantes, texturée et contour  $x_t$  et  $x_c$  tel que :

$$I = x_t + x_c \quad (3.3)$$

ou  $x_t$  et  $x_c$  représentent deux structures différentes. Supposons qu'il existe deux dictionnaires  $A_1, A_2 \in \mathbb{R}^{n \times K}$  vérifiant les deux conditions :

1) Pour  $i = t, c$

$\min_{x_i} \|x_i\|_1$  sous condition que

$$A_i x_i = x_i \quad (3.4)$$

Conduit à une représentation parcimonieuse  $\hat{x}_i$  de  $x_i$  dans  $A_i$

2) Pour  $i = t, c$

$\min_{x_i} \|x_i\|_1$  sous condition que

$$A_j x_i = x_i \quad (3.6)$$

Conduit à une représentation non parcimonieuse  $\hat{x}_i$  de  $x_i$  dans  $A_i$ .

Dans ce cas, deux dictionnaires  $A_1$  et  $A_2$  sont dit discriminant dans le cadre d'une représentation parcimonieuse pour des contenus de type différents. L'algorithme d'analyse en composantes morphologique permet de résoudre le problème d'optimisation suivant :

$$\min_{x_1, x_2} \|x_t\|_0 + \|x_c\|_0 \quad (3.7)$$

sous condition que

$$A_1 x_1 + A_2 x_2 = I \quad (3.8)$$

Ce qui peut être converti en :

$$\min_{x_1, x_2} (\|x_1\|_1 + \|x_2\|_1 + \lambda \|b - A_1 x_1 - A_2 x_2\|_2) \quad (3.9)$$

La solution de cette équation donne  $\hat{x}_1$  et  $\hat{x}_2$  la représentation parcimonieuse de  $x_t$  et  $x_c$  respectivement dans  $A_1$  et  $A_2$ . Ceci signifie aussi que le signal original  $b$  a été séparé en deux parties  $A_1 \hat{x}_1$  et  $A_2 \hat{x}_2$  qui correspondent à des approximations de  $x_t$  et  $x_c$  respectivement.

**Donnée :** Un image  $I$  contaminée par un bruit de variance  $\sigma^2$ .

**Entrées:** Un dictionnaire  $A = [A_1 A_2]$  composé de 2 sous-dictionnaires, un nombre d'itérations  $N_{iter}$  et un seuil final  $\tau$ .

**debut**

Initialiser les deux composantes  $\hat{x}_c = I$ ,  $\hat{x}_t = 0$

Initialiser le résidu  $r^{(0)} = \hat{x}_c$ .

Initialiser le seuil :  $k^* = argmax_k \|A_k^T I\|_\infty$ , soit  $\lambda_0 = max_{k \neq k^*} \|A_k^T\|_\infty$

**For**  $t = 1$  **to**  $T_{iter}$  **do**

**for**  $k = 1$  **to**  $2$  **do**

Calculer les résidus marginaux  $r_k^{(t)} = r_k^{(t-1)} + \hat{x}_k^{(t-1)}$

Mettre à jour les coefficients de la  $K^{ieme}$  composante en seuillant

$\alpha_k^{(t)} = thresh_{\lambda_{t-1}}(A_k^T r_k^{(t)})$

Mettre à jour la  $K^{ieme}$  composante  $\hat{x}_k^{(t)} = A_k \alpha_k^{(t)}$

Mettre à jour la solution  $\hat{x}^{(t)} = \sum_{k=1}^K \hat{x}_k^{(t)}$ .

Mettre à jour le résidu  $r^{(t)} = I - f^{(t)}$

Mettre à jour le seuil  $\lambda_t = \lambda_0 - t(\lambda_0 - \tau\sigma)/T_{iter}$

**fin**

**Résulte:** Les composantes  $\hat{x}_k$  de  $I$  dans  $A_k$

**Fig.2 :** l'Algorithme MCA appliquée a la décomposition d'image

### 2.3 Seuillage

Plusieurs seuillage existent dans la littérature [54]. Un point commun de ces seuillage est l'étape d'initialisation. La valeur de  $\lambda^1$  est égale au coefficient le plus fort du signal d'origine  $y$  projeté sur l'ensemble des bases du dictionnaire. Il s'écrit :

$$\lambda^{(1)} = \max_i \|T_i y\|_\infty \quad (3.10)$$

Deux comportements du seuil  $\lambda(k)$  sont proposées pour l'algorithme MCA : une décroissance linéaire ou exponentielle.

#### 2.4.1 Seuillage linéaire (SL)

L'approche de seuillage linéaire (SL) est qu'à chaque étape, une constante  $\sigma_{SL}$  définit la décroissance de  $\lambda$  au cours des itérations de l'algorithme. Elle est fixée à l'initialisation de l'algorithme et est retirée à chaque étape à la valeur précédente du seuil. Sa formule est de la manière suivante :

$$\lambda^{(k+1)} = \lambda^{(k)} - \delta_{SL} \text{ avec } \delta_{SL} = \frac{\lambda^{(1)} - \lambda_{min}}{N_{iter}} \quad (3.11)$$

Avec  $N_{iter}$ , le nombre d'itérations de la boucle principale de l'algorithme (fixe par l'utilisateur) et  $\lambda_{min}$  la plus petite valeur de seuil que l'on s'autorise. Généralement,

$$\lambda_{min} = \tau \sigma_\varepsilon \quad (3.12)$$

avec  $\tau = [3, 4]$ , valeurs usuellement utilisées en débruitage d'images avec l'algorithme MCA .

#### 2.4.2 Seuillage exponentielle (SE)

Dans certains cas, la distribution des coefficients se concentre autour de l'origine. Ce phénomène peut se produire pour diverses raisons comme le choix non pertinent des bases conduisant par exemple a une représentation similaire dans les différentes bases .Face a cette situation le Seuillage linéaire (SL) n'est pas optimale. C'est pourquoi, dans les auteurs utilisent également un seuil a décroissance exponentielle (SE), permettant de seuiller de grande coefficients dans les premières itérations de l'algorithme, et de plus petites coefficients sur les dernières itérations.

$$\lambda^{(k+1)} = \lambda^k * \delta_{SE} \quad (3.13)$$

avec :

$$\delta_{SE} = (\lambda^{(1)} - \lambda_{min})^{-\frac{1}{N_{iter} - 1}} \quad (3.14)$$

avec  $N_{iter}$  le nombre d'itérations de la boucle principale de l'algorithme (fixe par l'utilisateur) et  $\lambda_{min}$  le seuil minimum.

#### 2.4.3 Seuillage Mean of Max (MoM)

Afin d'obtenir un comportement adaptatif et pour pallier le problème d'initialisation de  $N_{iter}$  , dans [55] proposent une autre méthode de seuillage, donnée par les équations suivantes :

$$m_1 = \max_i \|A_i r(k)\|_\infty \quad (3.15)$$

$$m_2 = \max_{j, j \neq i_0} \|A_j r(k)\|_\infty \quad (3.16)$$

avec

$$i_0 = \operatorname{argmax}_i \|A_i r(k)\|_\infty \quad (3.17)$$

Le seuil de l'itération suivante sera décidé en effectuant la moyenne entre les deux maxima  $m_1$  et  $m_2$ . Ainsi :

$$\lambda^{(k+1)} = \frac{1}{2} (m_1 + m_2) \quad (3.18)$$

Cette méthode de seuillage a l'avantage de ne pas avoir de paramètre de décroissance à fixer et d'être adaptative (dépendante des caractéristiques du signal observé). Par contre, contrairement aux deux méthodes de seuillage originelles SL et SE, une projection supplémentaire à chaque itération sur les bases du dictionnaire est nécessaire.

## **2.5 Choix du dictionnaire**

Le choix du dictionnaire joue un rôle primordial pour aboutir à une bonne séparation. De nombreux dictionnaires ont été proposés pour capturer de différents types de caractéristiques des images. Les ondelettes [Mal98] constituent un outil important en compression d'images et permet de capturer de façon optimale les images à variations bornées. En effet le dictionnaire doit être choisi en adéquation avec le signal traité.

## **2.6 Transformée en ondelettes**

Il existe de nombreuses méthodes permettant d'effectuer la transformée en ondelettes, chacune se différenciant des autres soit par le type de l'ondelette mère employée, soit par la méthode d'implémentation de la transformée, suivant le type de traitement effectué.

La liste suivante expose quelques variantes de type d'ondelettes qui existe dans la littérature.

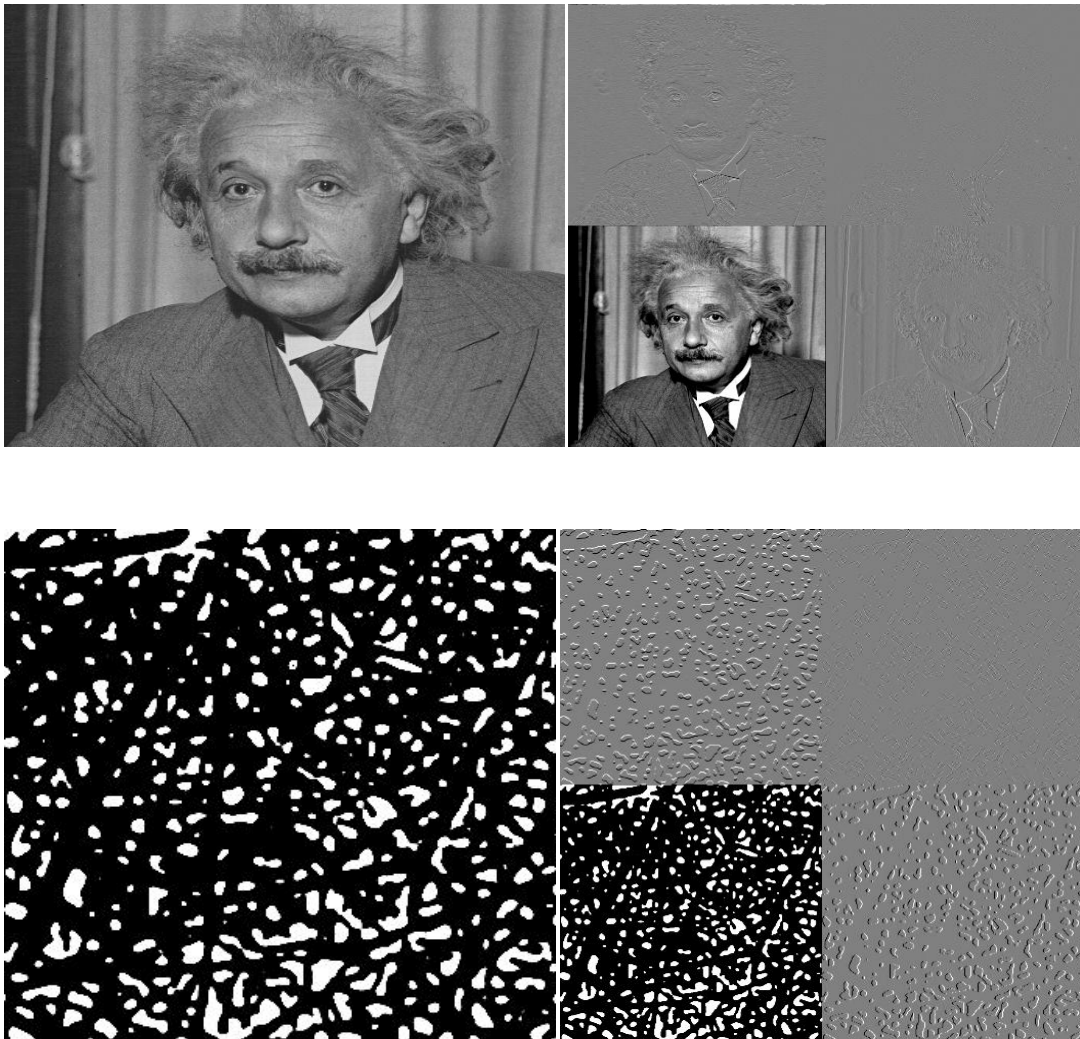
- les ondelettes orthogonales et bi-orthogonales : ce sont les ondelettes classiques. Elles s'implémentent par des filtres miroirs conjugués ou à reconstruction parfaite.
- les paquets d'ondelettes : les paquets d'ondelettes permettent de décomposer le signal sur un grand nombre de bases et de choisir au sens d'un certain critère, celle qui représente au mieux le signal.
- ondelettes directionnelles : elles permettent d'analyser les caractéristiques d'une fonction dans une direction donnée. Les bases de cosinus locaux : elles permettent une analyse fréquentielle précise sur de fenêtres adaptées à la fréquence en conservant les avantages de la transformée discrète.

- les ondelettes multiple: l'idée n'est plus d'utiliser une fonction fixe que l'on va translater et dilater, mais plutôt un nombre fini de fonctions. Cette méthode permet d'obtenir des combinaisons de propriétés utiles qui seraient impossibles avec les ondelettes classiques.
- Le lifting scheme : on abandonne ici complètement l'idée de translation et de dilatation. Cela fournit une grande flexibilité pour le développement d'ondelettes non linéaires.
- Ridgelets : adaptées aux contours de l'image.
- Curvelets [56] : représentation adaptée aux fonctions régulières par morceaux.
- Ondelettes morphologiques : on utilise ici la morphologie mathématique dans le but de développer des ondelettes basées sur des filtres non-linéaires.
- Ondelettes géométriques (bandelettes) : décomposition dans une base d'ondelettes anisotropes adaptée à la géométrie.

Dans notre travail on a utilisée la Transformée en ondelettes pour extraire le composant contour.

### **2.6.1 Transformée en ondelette non décimée (TOD)**

Il s'agit d'une version non décimée de la transformée orthogonale en ondelette (TOO) obtenu en ne tenant pas compte du pas de décimation dans la décomposition. La TOD permet de compenser le manque d'invariance à la translation de la TOO. Contrairement à la TOO, la transformée TOD peut être représentée par une matrice avec plus de colonnes que de lignes. Le facteur de redondance (i.e. le rapport entre le nombre de colonnes sur le nombre de lignes) est  $3J + 1$  où  $J$  est le nombre d'échelles. La TOD est censée de donner une représentation parcimonieuse pour des caractéristiques isotrope et non-parcimonieuse pour des caractéristiques fortement anisotrope. L'algorithme à trous de Shensa [SHE 92] donne un moyen efficace pour implémenter la TOD.



**Fig.3** exemple de décomposition en ondelettes orthogonale(TOD) d'une image. Algorithme de Mallat

### Algorithme a trous

L'algorithme a trous a été développé par [HOL89] et [SHE92] quelques années plus tard qui en a montre le lien avec la multi résolution. Il est similaire a celui de Mallat, mais les convolutions discrètes se font sans décimation. Dans cet algorithme, une seule fonction d'échelle est utilisée. L'approximation de l'image originale, est obtenue par filtrage et l'image de coefficients d'ondelettes est obtenue par soustraction de l'image originale et de l'approximation. Cet algorithme permet d'obtenir à chaque échelle une approximation et une image de détails non-directionnelle.

Le but de cet algorithme est de décomposer la fonction  $f(t)$  en un ensemble de  $J$  fonctions appelées plans d'ondelettes  $\{w_j(k)\}_{j=1 \dots J}$ .

Des fonctions appelées plans lisses  $\{c_j(k)\}_{j=1\dots J}$  correspondent à la fonction initiale  $f(t)$  lissée  $j$  fois. La fonction  $c_j(k)$  correspond à la projection de  $f(t)$  dans l'espace  $V_j$  et  $w_j(k)$  correspond à la projection de  $f(t)$  dans l'espace  $W_j$ . On considère que les données discrètes  $c_0(k)$  (plan lisse à la résolution 0) sont définies comme un produit scalaire à la position  $k$  entre  $f(t)$  et la fonction d'échelle  $\phi(t)$  :

$$c_0(k) = \langle f(t), (t - k) \rangle \quad (3.19)$$

La fonction d'échelle doit satisfaire l'équation de dilatation :

$$\frac{1}{2}\phi\left(\frac{t}{2}\right) = \sum_{l \in \mathbb{Z}} h_l \phi(t - l), \quad (3.20)$$

Où  $h$  est un filtre passe bas discret. Il a la même définition à tous les niveaux de résolution. Ainsi, si  $h$  a  $n$  valeurs significatives à l'échelle  $j$ , il en est de même à l'échelle  $j + 1$ . Le plan lisse  $c_j(k)$  à la résolution  $j$  et la position  $k$  est donnée par :

$$c_j(k) = \langle f(t), \frac{1}{2^j} \phi\left(\frac{t-k}{2^j}\right) \rangle \quad (3.21)$$

L'équation de dilatation permet d'obtenir la récurrence suivante :

$$c_{j+1}(k) = \sum_{l \in \mathbb{Z}} h_l c_j(k + l2^j) \quad (3.22)$$

L'approximation  $c_{j+1}$  est obtenue à partir du plan lisse  $c_j$  par convolution avec le filtre  $h$  et un pas de  $2^j$  entre coefficients créant ainsi des trous dans le filtre d'où la dénomination de l'algorithme.

En pratique,  $2^j - 1$  zéros sont insérés dans le filtre  $h$  à chaque résolution, ce qui permet d'obtenir, par convolution avec le signal de départ, le plan lisse  $c_{j+1}$ .

Nous considérons maintenant une fonction ondelette  $\psi$  (filtre passe haut)

$$\text{telle que : } \frac{1}{2}\psi\left(\frac{t}{2}\right) = \sum_{l \in \mathbb{Z}} g_l \phi(t - l), \quad (3.23)$$

Nous avons :

$$w_j(k) = \langle f(t), \frac{1}{2^j} \psi\left(\frac{t-k}{2^j}\right) \rangle \quad (3.24)$$

Nous obtenons une récurrence :

$$w_{j+1}(k) = \sum_{l \in \mathbb{Z}} g_l w_j(k + l2^j) \quad (3.25)$$

Le plus simple, pour le choix du filtre  $g$ , consiste à effectuer la différence entre approximations successives :

$$\hat{g}(\xi) = 1 - \hat{h}(\xi) \quad (3.26)$$

L'algorithme permet d'obtenir une pyramide de résolution contenant des approximations successives. La différence entre une approximation à l'échelle  $j-1$  et celle d'une échelle immédiatement supérieure donne ce qu'on appelle le plan d'ondelette à l'échelle

$$w_j(k) = c_{j-1}(k) - c_j(k) = \frac{1}{2^{j-1}} \langle f(t), \frac{t-k-}{2^j} \phi(\frac{t-k}{2^j}) \rangle \quad (3.27)$$

Nous pouvons reconstruire le signal d'une façon simplifiée en additionnant les plans d'ondelettes avec la dernière approximation :

$$c_0 = c_p + \sum_{j=1}^p w_j \quad (3.28)$$

L'algorithme à trous est facilement étendu dans le cas bidimensionnel. On considère la fonction  $\phi(t, u)$  telle que :

$$\frac{1}{4} \phi\left(\frac{t}{2}, \frac{u}{2}\right) = \sum_{l, m \in \mathbb{Z}} h_{l, m} \phi(t-l, u-m) \quad (3.29)$$

Les données discrètes  $c_0(l, m)$  sont définies comme le produit scalaire à la position  $l$  et  $m$  entre la fonction image  $f(t, u)$ :

$$c_0(l, m) = \langle f(t, u), \phi(t-l, u-m) \rangle \quad (3.30)$$

Comme dans le cas monodimensionnel, la suite d'approximations successives de l'image est calculée par la récurrence :

$$c_{j+1}(k, k') = \sum_{l, m \in \mathbb{Z}} c_j(k + l2^j, k' + m2^j) \quad (3.31)$$

On choisit une fonction ondelette  $\psi$  engendrée par la fonction d'échelle  $\phi$  soit :

$$\frac{1}{4} \psi\left(\frac{t}{2}, \frac{u}{2}\right) = \sum_{l, m \in \mathbb{Z}} g_{l, m} \phi(t-l, u-m) \quad (3.32)$$

Les plans d'ondelettes sont obtenus par la récurrence suivante :

$$w_{j+1}(k, k') = \sum_{l, m \in \mathbb{Z}} g_{l, m} c_j(k + l2^j, k' + m2^j) \quad (3.33)$$

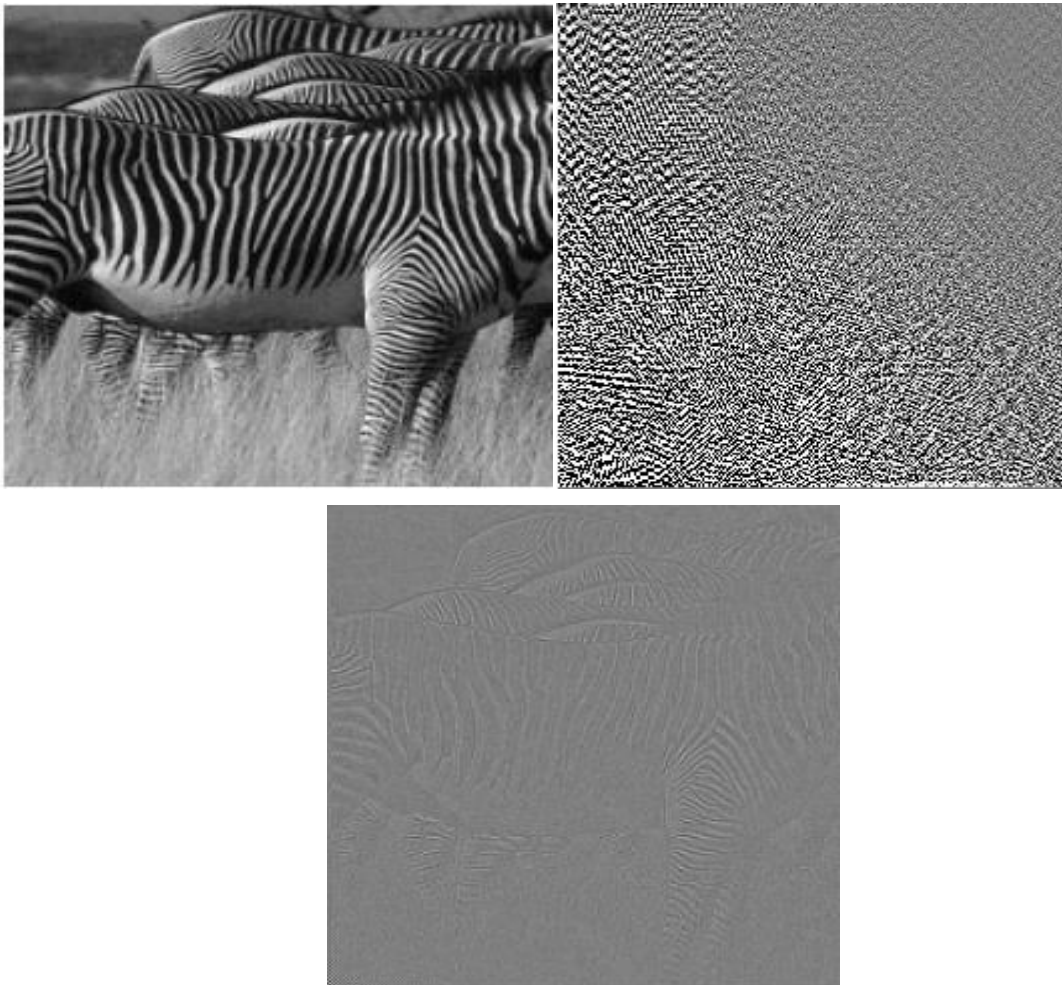
De nombreuses fonctions échelles répondent aux critères décrits ci-dessus. Pour utiliser facilement la transformation en ondelettes en 2D et en 3D, on choisit une fonction échelle à variables séparées :

$$\phi(t, u) = \phi(t)\phi(u) \quad (\text{ou } \phi(t, u, v) = \phi(t)\phi(u)\phi(v)). \quad (3.34)$$

**2.7 Transformée en cosinus discrète :** Cette transformée a été utilisée pour l'extraction de la composante texture.

**✚ DCT globale**

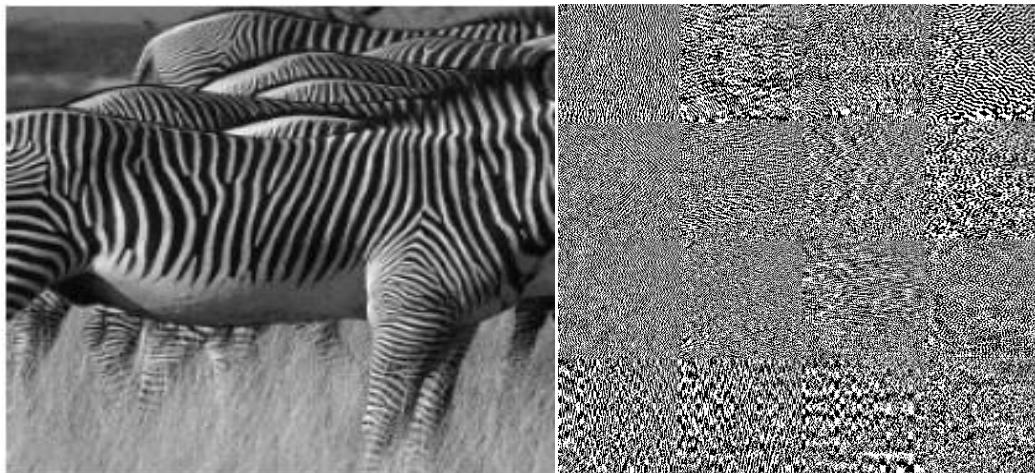
La transformée en cosinus discrète est utilisée dans la compression parce qu'elle permet de concentrer l'information pertinente sur un petit nombre de coefficients.



**Fig.4** exemple de Transformée en cosinus discrète et la reconstruction de l'image originale

### ✚ Transformée en cosinus locale

Transformée en cosinus locale est appliquée à l'image par bloc.



**Fig.5** exemple Variante Transformée en cosinus discrète locale IV

## 5. Tests et Résultats

Nous présentons dans cette partie quelques exemples de résultats expérimentaux obtenus à partir de l'algorithme MCA étudiés. Les tests ont été implémentés sous l'environnement Microsoft Visual Studio 2008 sur un micro ordinateur portable de type Acer, ayant une fréquence de 2 GHZ, une mémoire (RAM) de 2 GO et un disque dur de 64 Go.

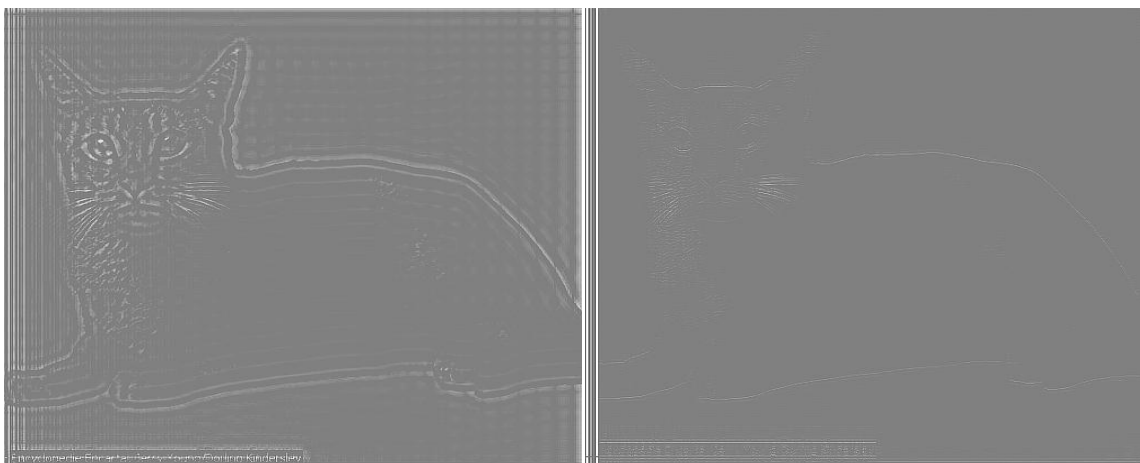
Les résultats de la décomposition obtenue sur chaque image sont donnés aux figures suivantes.

**Tableau 1** : Les paramètres de MCA appliquée à la décomposition d'image

Images	Dictionnaire	Paramètres de MCA
chat 510×414	Contour : UDWT  Texture Local : LDCT	$N_{iter}=100$ $A_1=D_w$ (TOD) $A_2 = D_c$ (transformée en cosinus locaux)
Zèbre 224×224		
léopard490×320		
Tigre 468×312		
Branchy354×335		
Barbara 512×512		



Image originale



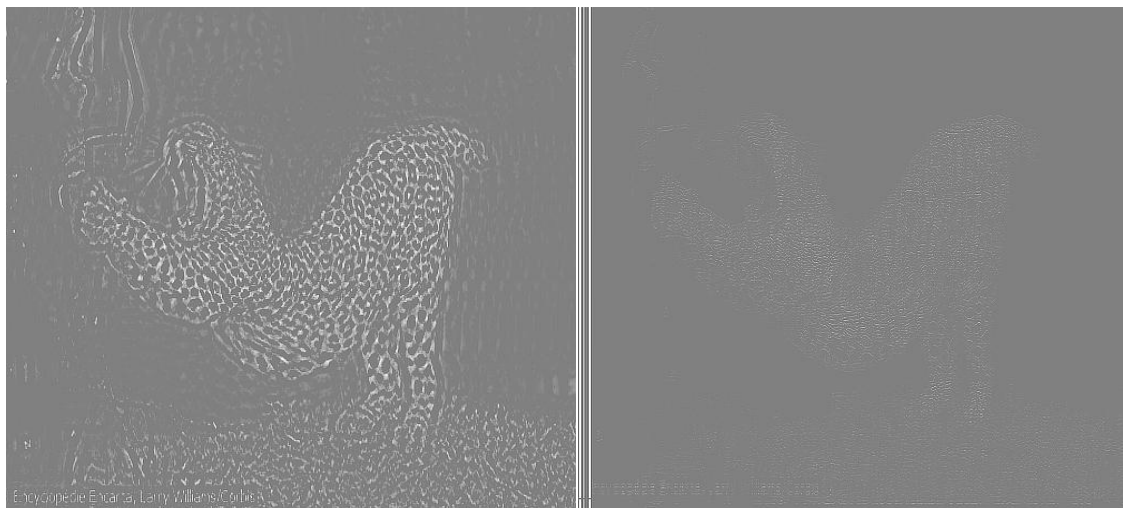
Texture

structure (contour)

**Fig.6** : Composantes texture +structures



Image originale



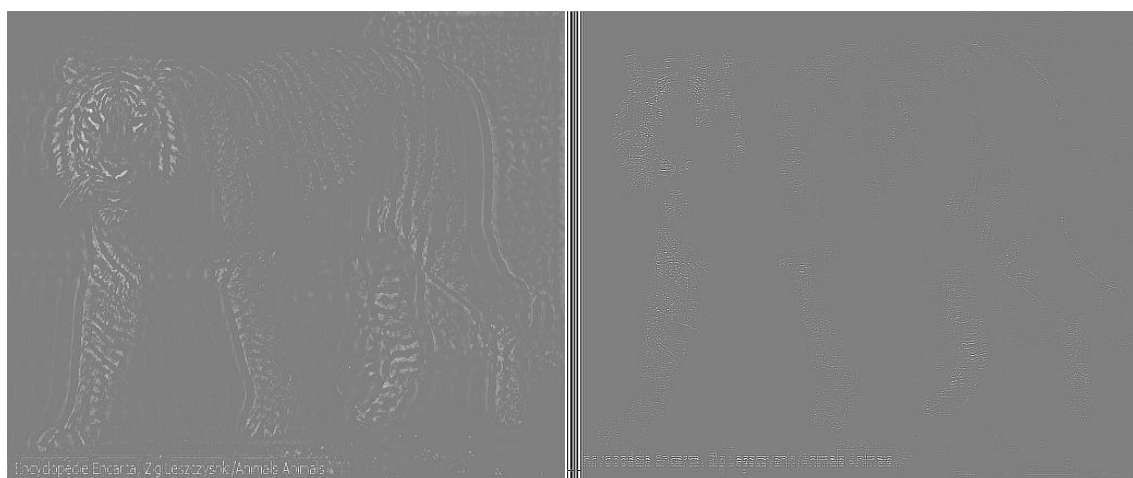
Texture

structure (contour)

**Fig.7:** Composantes texture +structures



Image originale



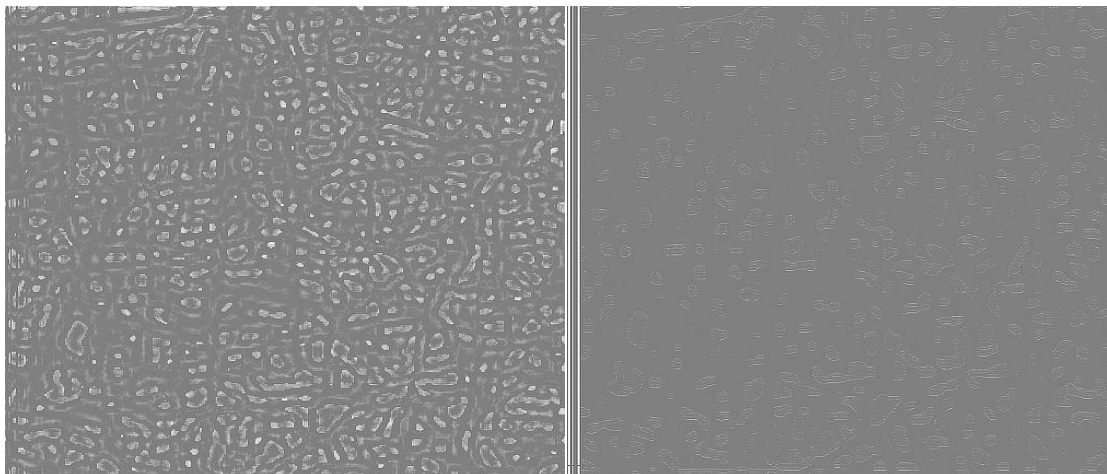
Texture

structure

**Fig.8:** Composantes texture +structures



Image originale



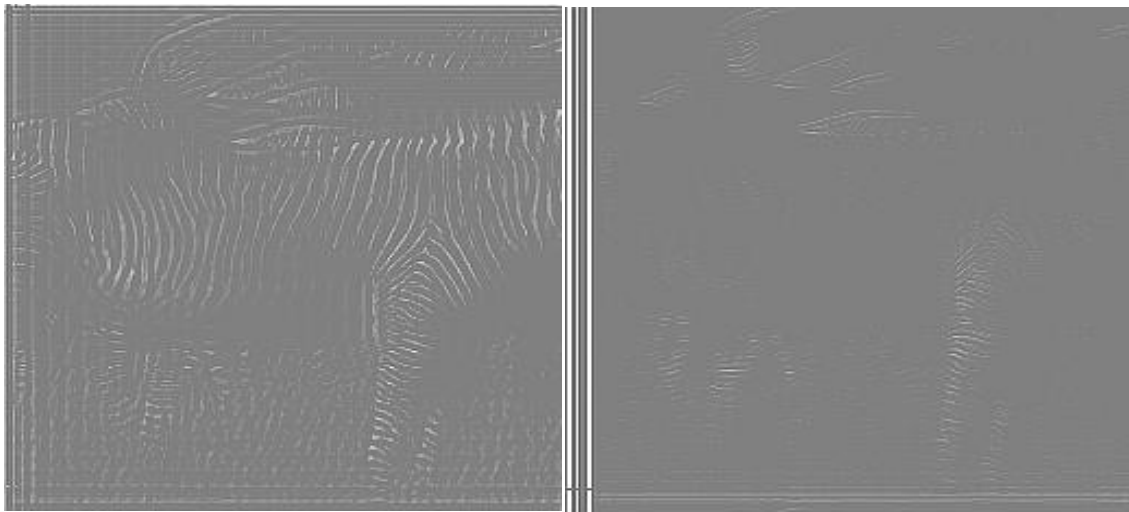
Texture

structure

**Fig.9:** Composantes texture +structures



Image originale



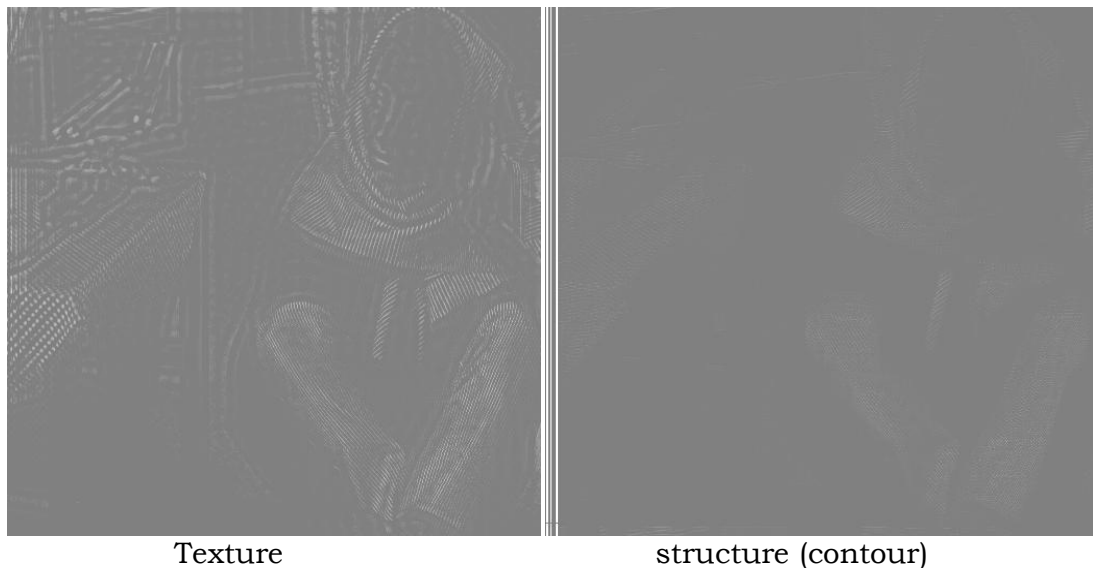
Texture

structure

**Fig.10:** Composantes texture +structures



Image originale



Texture

structure (contour)

**Fig.11** Composantes texture +structures

Les résultats expérimentaux ont montré que les structures extraites sont bonnes pour quelques images et sont médiocres pour d'autres images car l'algorithme à trous est adapté aux images qui ont des petit motifs ou sous forme de trous. Le Choix des paramètres influe aussi sur la qualité des structures extraites. Notre méthode a le pouvoir de séparabilité de contour et de texture en un temps d'exécution très cours (2min).

## **6. Conclusion**

Dans ce chapitre nous avons proposé une méthode qui s'attaque au problème difficile d'extraction de texture dans une image à partir d'une représentation parcimonieuse. L'algorithme d'analyse en composantes morphologiques (MCA) est utilisé pour optimiser le modèle de décomposition parcimonieux. Deux dictionnaires sont utilisés pour capturer les structures texture et contour de l'image. Le dictionnaire de la texture est basé sur la transformée en cosinus locaux. L'autre est basé sur Transformée en ondelettes (à trous). Ensuite le seuillage exponentielle a été appliqué sur les coefficients de la transformée en cosinus locaux et les coefficients de l'algorithme à trous. L'algorithme final permet d'obtenir la décomposition en deux composantes, contours et texture.

On peut proposer d'autres dictionnaires pour la structure géométrique et la texture selon le principe de l'analyse harmonique computationnelle.

## **Conclusion générale**

Les problèmes posés par la décomposition sont multiples et encore loin d'être résolus car ils sont souvent très complexes. Rappelons que l'idée principale derrière MCA est d'employer la diversité morphologique des différents dispositifs contenus dans les données et d'associer chaque morphologie à un dictionnaire. Rappelons aussi que le MCA est basé sur le modèle de la représentation parcimonieuse.

Dans ce mémoire nous avons présenté une nouvelle technique de séparation qui est l'analyse en composantes morphologique. Cette méthode s'attaque au problème difficile d'extraction de texture dans l'image à partir d'une représentation parcimonieuse. Deux dictionnaires sont utilisés pour capturer la structure géométrique et la texture de l'image. L'un est basé sur une transformée d'ondelettes non décimée et l'autre sur la transformée en cosinus locaux. L'algorithme d'analyse en composantes morphologiques est utilisé pour optimiser le modèle de décomposition parcimonieux. D'autres dictionnaires peuvent être proposés pour la texture et la structure géométrique, La nouveauté de cette méthode ci que cette notion contribue à la robustesse et à l'amélioration des performances des méthodes de séparation de sources basées sur la parcimonie, notamment à la méthode de l'analyse en composantes morphologique.

Il serait intéressant, dans le futur, d'utiliser de d'autres nouvelles représentations comme les bandelettes, les curvelettes, les contourellette comme dictionnaire pour la structure géométrique qui permettrait certainement de gagner encore sur la qualité de la composante extraite.

[1]C.Germain, M.KUNT Contribution à la caractérisation multi-échelle de l'anisotropie des images texturées, Université Bordeaux I, thèse de doctorat, Décembre 1997.

[2] S.Dubois R.Péteri M. Ménard, Indexation de Textures Dynamiques à l'aide de Décompositions Multi-échelle, publié dans "RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle), Lyon : France (2012).

[3] C.Germain Contribution à la caractérisation multi- échelle de l'anisotropie des images texturées université bordeauxI 1997 IEEE, Trans. On image processing, vol8N°4, pp.4 906 503, 1999.

[4] L.Paulhac, Outils et méthodes d'analyse d'images 3D texturées : application à la segmentation des images échographiques, université de François Rabelais de tours, thèse, novembre 2009.

[5] C.Bernard, Ondelettes et problèmes mal posés : la mesure du flot optique et l'interpolation irrégulière,these de doctorat,France.

[6] Linda G. Shapiro , George C. Stockman, *Computer Vision*, Upper Saddle River: Prentice-Hall, 2001.

[7] [http://uploads.siteduzero.com/files/65001\\_66000/65110.jpg](http://uploads.siteduzero.com/files/65001_66000/65110.jpg)

[8]B. Galerne, Modèles d'image aléatoires et synthèse de texture Stochastic Image Models and Texture Synthesis, thèse de doctorat, Ecole Normale Supérieure de Cachan, 2010.

[9]<http://images.google.fr/imgres>

[10] G.Nguyen,Extraction de zones d'interets dans une image de textures, Rapport de stage, Laboratoire Informatique, Image et Interaction ,Université de La Rochelle,2009.

[11] Journet N., Analyse d'images de documents anciens : une approche texture, Thèse de doctorat, Université de La Rochelle, 2006.

[12] A. Hillion, Une méthode de classification de textures par extraction linéaire non paramétrique de caractéristiques A non parametric linear feature extraction approach to texture classification, volume 5 - n° 4 – 1988.

[13] J. Landre , F. T ruchetet, Approche exploratoire multiresolution basée sur le contenu d'une base d'images paléontologique.

[14] j-p. goglio, A. chehikian, Mesure d'orientations cristallographiques par traitement d'images de microscope électronique, seizième colloque gretsi , Grenoble 15-19septembre 1997.

[15] T-L. Le, Indexation et recherche de vidéo pour la vidéosurveillance, thèse de doctorat, Université de Unice-Sophia Antipolis, 2009.

[16] S.Boucher, Traitement d'image et morphologie mathématique, Thèse de doctorat, Ecole nationale supérieur de Mines de Paris, 1990.

[17] Derin, H. Elliott, R. Cristi and D. Geman, Bayes smoothing algorithms for segmentation of images modeled by Markov random fields, in Int. Conf. Acoust. Speech Signal Process., ICASSP'84, vol.9, Mars. 1984.

[18] Youness , Saporta. Une méthodologie pour la comparaison de partitions. revue de statistique appliquée, 52:97-120, 2004.

[19] C.C.GotliebandH.E.Kreyszig,Texture descriptors based on co-occurrence matrices.Computer Vision,Graphics ,andImage Processing, 51(1):7086,juliet 1990.

[20] T.iftenea et S. abdelmounaine, Comparaison entre la matrice de cooccurrence et la transformation en ondelettes pour la classification texturale des images hrv (xs) de spot. Télédétection, vol. 4, n° 1, p. 39-49,2004.

[21] A. Akono, E. Tonyé et A. Ndi nyongui, Etude comparative de paramètres de texture d'ordre deux et trois pour une classification d'images rso dans un contexte de multi-résolution hiérarchique, Télédétection, 2006, vol. 6, n° 4, p. 249-261.

[22] T. Randen, , J.H. Husoy, Filtering for Texture Classification: a comparative study. IEEE Transactions PAMI, vol. 21, n° 4, p. 291-310,1999.

[23] G. n. srinivasan, shobha g. statistical texture analysis, proceedings of world academy of science, engineering and technology volume 36 december 2008 issn 2070-3740, 2008.

[24] D. Jeulin. Random texture models for material structures. *Statistics and Computing*, 10:121–132, 2000. doi:10.1023/A:1008942325749,110.

[25] A.Bekkhoucha, Traitement d'images et analyse de textures par les longueurs de plages, these de doctorat, Université Mohammed Premier - Oujda ,septembre 2001.

[26] S. Mavromatis ,Analyse de texture et Visualisation scientifique,these de doctorat ,Université de la Méditerranée,2001.

[27] S. Geman and D. Geman, “Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images”, *IEEE Trans. on Pattern Anal.Machine Intell.*, vol.6, no.6, 1984.

[28] y. s. abu-mostafa, d. psalti, Recognitive aspects of moment invariants, *IEEE transactions on pattern analysis and machine intelligence*, vol. pami-6, no. 6, november 1984.

[29] Oren Boiman, Eli Shechtman and Michal Irani, In Defense of Nearest-Neighbor Based Image Classification, *IEEE Conference on Computer Visionand Pattern Recognition (CVPR)*, juin,2008.

[30] T. arif, Z. shaaban, L. Krekor, S. baba Object classification via geometrical, zernike ,and le gendre moments, *Journal of Theoretical and Applied Information Technology*, University, Amman 11931, Jordan,2009.

[31] L. journaux, J. miteran, F. cointault, marie-france destainclassification de textures par descripteurs généralisés de fourier dans différents contextes de réduction de dimension, *Université de Belgique*.

[32] Arivazhagan, S., L. Ganesan, and S.P. Priyal, Texture classification using Gabor wavelets based rotation invariant features.*Pattern Recognition Letters*, 2006. 27(16): p. 1976-1982.

[33] A.Taleb-Ahmed, P.Dubois,E.Duquenoy,Analysis methods of CT-scan images for the characterization of bone texture: First results”, *Pattern Recognition Letters* 24 (2003) 1971- 1982.

[34] J.M. Fadili. *Une exploration des problèmes inverses par la représentations parcimonieuses et l'optimisation non lisse*, 2010. 123

- [35] S. Mallat. *A Wavelet Tour of Signal Processing, The Sparse Way*. Academic Press, 3rd edition, 2009.
- [36] P. Abrial, Y. Moudden, J.L. Starck, B. Afeyan, J. Bobin, M.J. Fadili, and M.K. Nguyen. Morphological component analysis and inpainting on the sphere : Application in physics and astrophysics. *Journal of Fourier Analysis and Applications*, 13(6) :729–748, 2007.
- [37] J. Bobin, J-L. Starck, M.J. Fadili, and Y. Moudden. Sparsity and morphological diversity in blind source separation. *IEEE Transactions on Image Processing*, 16(11) :2662–2674, 2007.
- [38] J. Bobin, Y. Moudden, J.-L. Starck, and M.J. Fadili. Sparsity and morphological diversity for hyperspectral data analysis. In *IEEE ICIP*, Cairo, 2009.
- [39] J. Bobin. *Diversité morphologique et analyse de données multivaluées*. PhD thesis, Université Paris XI, LAIM CEA-CNRS, 2008.
- [40] J.-L. Starck, M. Elad, and D.L. Donoho. Redundant multiscale transforms and their application for morphological component analysis. *Advances in Imaging and Electron Physics*, 132, 2004.
- [41] J.-L. Starck, M. Elad, and D.L. Donoho. Image decomposition via the combination of sparse representation and a variational approach. *IEEE Transactions on Image Processing*, 14(10), 2005.1570–1582.
- [42] S. Mallat and Z. Zhang. Matching pursuit in a time-frequency dictionary. *IEEE Transactions on Signal Processing*, pages 3397 – 3415, 1993.
- [43] Y.C. Pati, Ramin Rezaifar et P.S. Krishnaprasad : Orthogonal matching pursuit : Recursive function approximation with applications to wavelet decomposition. In *Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems, and Computers*, 1993.
- [44] D.L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7), 2001. 2845–2862.

- [45] S. S. Chen, D.L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1), 1999. 33–61.
- [46] Y. Meyer. Oscillating patterns in image processing and in some nonlinear evolution equations. *The Fifteenth Dean Jacqueline B. Lewis Memorial Lecture*, 2001.
- [47] J.F. Aujol and A. Chambolle. Dual norms and image decomposition models. *International Journal of Computer Vision*, 63(1), 2005. 85–104.[
- [48] J.-F. Aujol and G. Gilboa. Constrained and based solutions for TV-hilbert space image denoising. *Journal of Mathematical Imaging and Vision*, 26(1-2), 2006. 217-237.
- [49] J.F. Aujol, Guy Gilboa, Tony Chan, and Stanley Osher. Structure and texture compression modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1),2006. 111–136.
- [50] J.-F. Aujol, G. Aubert, L. Blanc-Feraud, and A. Chambolle. Image decomposition into a bounded variation component and an oscillating component. *Journal of Mathematical Imaging and Vision*,22, 2005. 71–88.
- [51] L.A. Vese and S. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. *Journal of Scientific Computing*, 19, 2003. 553–577.
- [52] M.J. Fadili, J.-L. Starck, J. Bobin, and Y. Moudden. Image decomposition and separation using sparse representations : an overview. *Proceedings of the IEEE, Special Issue : Applications of Sparse Representation*, 2009. in press.
- [53] A.M. Bruckstein, D.L. Donoho, and M. Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51(1), 2009. 34–81.
- [54 ] S. dubois, décompositions spatio-temporelles pour l'étude des textures dynamiques contribution à l'indexation vidéo, thèse de doctorat, université de la rochelle ,novembre 2010

[55] J. Bobin, J-L. Starck, M.J. Fadili, Y. Moudden, and D.L. Donoho. Morphological component analysis : An adaptive thresholding strategy. *IEEE Transactions on Image Processing*, 16(11) :2675–2681,2007.

[56] E.J. Candes, L. Demanet, D.L. Donoho, and L. Ying. Fast discrete curvelet transforms. *Multiscale Modeling and Simulation*, 5(3), 2006. 861–899.

.