#### REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE UNIVERSITE EL HADJ LAKHDAR BATNA



Faculté de Technologie Département d'Electronique

*Mémoire* Présenté en vue de l'obtention du diplôme de Magister en Electronique

**OPTION**: Traitement du signal

Présenté par :

Mr. KHELALEF Aziz

## Thème

# Débruitage des images fixes par les techniques de l'intelligence artificielle

Soutenu le : / / 2012 Devant le jury :

MELAAB Djamel	M.C. A	Univ. Batna	Président
<b>BENOUDJIT</b> Nabil	Prof.	Univ. Batna	Rapporteur
BENZID Rédha	<b>M. C. A</b>	Univ. Batna	Examinateur
<b>BENABDELKADER Souad</b>	<b>M. C. A</b>	Univ. Batna	Examinateur
KACHA Abdellah	<b>M. C. A</b>	Univ. Jijel	Examinateur

Année Universitaire 2011/2012

Je Dédie ce travail :

A mon père et ma mère.

A ma famille.

A tous ceux qui m'aiment et ceux que j'aime

Nous rendons grâce à Dieu qui nous a donné l'aide, la patience et le courage pour accomplir ce travail.

Je tiens à adresser mes plus vifs remerciements aux:

- Pr. Benoudjit Nabil pour m'avoir encadré et pour les recommandations qu'il m'a prodiguées et qui m'ont été d'un grand apport;
- Dr. MELAAB Djamel qui m'a honoré par sa présence en qualité de président de jury;
- Membres de jury: Dr. BENZID Rédha, Dr. BENABDELKADER Souad et Dr. KACHA Abdellah pour avoir accepté de juger mon travail.

Je remercie aussi tous ceux qui ont contribué à l'élaboration de ce travail de près ou de loin et qui méritent d'y trouver leurs noms.

> L'auteur Khelalef Aziz

Som	ma	ire
00111		

Introduction générale	1
Chapitre I : Dégradation et Débruitage des Images	
I.1.Introduction	.4
I.2. Dégradation d'images	.4
I.2.1. Le bruit blanc gaussien	5
I.2.2. Le bruit poivre et sel	5
I.2.3. Le bruit multiplicatif	5
I.3. Débruitage d'images	5
I.4. Critère d'analyse pour le débruitage d'images	6
I.4.1. Critère subjectif (aspect Visuel)	6
I.4.2. Critère objectif	6
A. L'erreur quadratique moyenne « MSE »	6
B. Le rapport signal sur bruit en pic « PSNR »	.7
I.5. Les méthodes standards de débruitage	7
I.5.1. Filtrage spatial	.7
A. Filtre moyenneur	8
B. Filtre médian	8
I.5.2. Filtrage fréquentiel	8
A. Filtre Gaussien	9
B. Filtre de Wiener	9
I.6. Plateforme informatique	.10
I.7. Résultats expérimentales	.10
I.8. Résultats de débruitage des images contaminées par un bruit gaussien	.11
I.9. Représentation des résultats	.12
I.10.Discussion des résultats	.13
I.11. Conclusion	.15

## Chapitre II : Méthodes Récentes pour le Débruitage des Images

II.1. Introduction	16
II.2. Débruitage d'image contaminée par un bruit blanc gaussien	16
II.3. Estimation de la variance du bruit	17
II.4. Le seuillage (thresholding)	18
II.5. Seuillage de D.L Donoho	19
II.5.1. Seuillage doux « Soft thresholding »	19
II.5.2. Seuillage dur « Hard thresholding »	20
II.6. Seuillage "VisuSrhrink"	20
II.7. Seuillage dépendant-niveau « Level Shrink »	21
II.8. Seuillage " BayesShrink"	22
II.9. Seuillage "NormalShrink"	22
II.10. Résultats de simulation	23
II.10.1. Résultats de débruitage	23
II.10.2. Echantillons des images débruitées	24
II.10.3. Représentation des résultats	26
II.10.4. Discussion des résultats	27
II.11. Conclusion	28

## Chapitre III: Débruitage par TNN (Thresholding Neural Network)

III.1. Introduction	30
III.2. Structure du TNN (thresholding neural network)	30
III.3. Débruitage des images par TNN	32
III.3.1. Méthode proposée par Zhang [6] :« Thresholding neural network for	or adaptive
noise reduction »	32
III.3.2. Méthode Proposée dans [7] (New TNN) :« Image denoising in t	he wavelet
domain using a new adaptive thresholding function »	33
III.4. Résultats de simulation	
III.4.1. Résultats de débruitage	
III.5. Courbes d'apprentissage	
III.6. Echantillons d'images débruitées	40
III.7. Représentation des résultats	42

III.8. I	Discussion4	3
III.9. <b>(</b>	Conclusion4	3

## Chapitre IV : Méthode Proposée pour le Débruitage des Images Fixes

IV.1. Introduction	44
IV.2. Structure générale de la technique proposée	44
IV.3. Configuration de la méthode	47
IV.4. Résultats de débruitage	48
IV.5. Courbes d'optimisation	49
IV.6. Echantillons d'image débruitée	51
IV.7. Représentation des résultats	54
IV.8. Discussion des résultats obtenues par la méthode proposée	55
IV.9. Etat récapitulatif des résultats	56
IV.10. Echantillons des images débruitées	57
IV.11. Discussion des résultats	59
IV.12. Conclusion	61

Conclusion générale
---------------------

## Annexe 1: Transformée en Ondelettes

1.1 Introduction	64
1.2. Les ondelettes	64
1.3. La transformée en ondelettes continue	65
1.4. La transformée en ondelettes discrète (T.O.D)	66
1.5. Algorithme de MALLAT	67
1.6. Architecture Pyramidale	68
1.7. Schéma de décomposition des images en ondelettes	69
1.8. Exemple de décomposition	70
1.9. Exemples d'ondelettes	70
1.10. Conclusion	71

## Annexe 2: Algorithmes Génétiques

2.1. Introduction	72
2.2. Formulation du problème d'optimisation	72
2.3. Terminologie, définitions et notations employées	73
2.3.1. Terminologie et définitions	73
2.3.2. Notations	74
2.4. Structure de l'Algorithme Génétique	75
2.5. Initialisation de la population	77
2.6. Codage et décodage des paramètres	77
2.7. Contraintes de pénalité	77
2.8. Sélection des parents	77
2.9. Méthodes de sélection	78
2.10. Opérateurs de recombinaison génétiques	78
2.10.1. Croisement	78
2.10.1.1. Croisement à un site	79
2.10.1.2. Croisement multi-sites	80
2.10.1.3. Croisement uniforme	80
2.10.2. Mutation	81
2.11. Sélection des individus d'une nouvelle génération	82
2.12. Critères de convergence d'un algorithme génétique	83
2.13. Algorithmes génétiques à codage réel	84
2.13.1. Croisement	84
2.13.1.1. Croisement discret	84
2.13.1.2. Croisement continu	85
2.13.2. Mutation	85
2.13.2.1. Mutation uniforme (ou mutation globale)	85
2.13.2.2. Mutation non uniforme (ou mutation locale)	85
2.14. Conclusion	86

## Annexe 3: SDIF (Simulateur de Débruitage des Images Fixes)

Références	
3.4 Conclusion	89
3.3. Mode d'utilisation	
3.2. Interface du Simulateur	87
3.1. Introduction	

## Liste Des Figures

### Introduction Générale

<b>Figure 1</b> . Exemple de debluitage d'innage
--

## Chapitre I

Figure I.1 : Exemple d'un voisinage 3x3 des pixels	8
Figure I.2: (a) Image bruitée par un bruit poivre et sel à 5% (PSNR= 18.5170),	
(b) Image débruitée avec un filtre médian 3 x 3 (PSNR= 34.4944)	10
Figure I.3: Représentation des résultats de débruitage en PSNR en fonction de	
la variance de bruit sur les images test par différents filtres	12
Figure I.4: Image « Lena » débruitée par un filtre moyenneur avec différentes	
tailles du masque	13
Figure I.5: Image « Lena » débruitée par un filtre médian avec différentes tailles	
du masque	13
Figure I.6: Image « Lena » débruitée par différents filtres	14

## Chapitre II

Figure II.1: (a) Décomposition en ondelettes d'une image, (b) Histogramme de
la sous bande HH <sub>1</sub> (coefficients diagonales du premier niveau de décomposition)17
Figure II.2 : Les trois étapes du seuillage des coefficients en ondelettes
Figure II.3 : Seuillage doux
Figure II.4: Seuillage dur
<b>Figure II.5:</b> Résultats du débruitage sur l'image Lena ( $\sigma$ =20) avec différentes
méthodes de débruitage24
<b>Figure II.6</b> : Résultats du débruitage sur l'image Cameraman ( $\sigma$ =20) avec
différentes méthodes de débruitage25
Figure II.7: Représentation graphique des résultats en PSNR sur les images de
références pour différentes variances de bruit par différentes techniques de débruitage26
Figure II.8 : Phénomène de Gibbs sur une image débruitée par NormalShrink et
BayseShrink

## Chapitre III

## Chapitre IV

Figure IV.1 : Structure générale de la méthode proposée
Figure IV.2 : Principe de l'algorithme « Cycle Spinning »
Figure IV.3 : Fonction de seuillage proposée dans [7]47
Figure IV.4 : Courbe d'optimisation par AG sur l'image Lena avec une variance
de 10
Figure IV.5 : Courbe d'optimisation par AG sur l'image Barbara avec une variance
de 10
Figure IV.6: Images Lena débruitées par différentes techniques avec une variance
σ=20
Figure IV.7: Images débruitées par différentes techniques pour l'image Boat avec
une variance $\sigma=30$
Figure IV.8: Images débruitées par différentes techniques pour l'image cameraman
avec une variance $\sigma=30$
Figure IV.9: Représentation du PSNR des images débruitées en fonction de la variance

de bruit par différentes techniques de débruitage	54
Figure IV.10: Résultats de débruitage de l'image Lena par toutes les techniques	
étudiées (σ=20)	57
Figure IV.11: Images débruitées par toutes les méthodes discutées sur l'image Boat	
avec une variance ( $\sigma$ =30).	59

#### Annexe 1

Figure 1.1 : Représentation Temps échelle pour ondelettes	66
Figure 1.2 : La décomposition en approximation et détail	67
Figure 1.3 : Algorithme de MALLAT bi-dimentionnel	68
Figure 1.4 : Décomposition pyramidale de profondeur 3 d'un signal bidimensionnel	69
Figure 1.5 : La décomposition multi niveaux par ondelettes	69
Figure 1.6 : Exemple de décomposition par ondelettes en 4 niveaux.	70
Figure 1.7 : Exemples d'ondelettes	71

#### Annexe 2

Figure 2.1: Définition de l'environnement, de l'individu et la population.	74
Figure 2.2: un individu caractérisé par un chromosome codé en binaire	74
Figure 2.3 : Structure de l'algorithme génétique.	76
Figure 2.4 : principe de croisement à un site.	79
Figure 2.5 : principe de croisement multi-sites	80
Figure 2.6 : Principe du croisement uniforme	81
Figure 2.7 : principe de la mutation.	81
Figure 2.8 : Convergence de l'algorithme génétique.	83
Figure 2.9 : Chromosome d'un individu dans les algorithmes génétiques à codage réel	84
Figure 2.10 : principe du croisement discret.	85

#### Annexe 3

Figure 3.1 : Interface graphique du SDIF.	ique du SDIF87	
Figure 3.2 : Sélection d'image.	88	
Figure 3.3 : Sélection de la variance du bruit	88	

Figure 3. 4: Fenêtre des performances.		
Figure 3.5: Fenêtre d'affichage et d'enregistrement	89	
Figure 3.6 : Onglette des outils.	89	

#### Liste Des Tableaux

## Chapitre I

<b>Tableau I.1:</b> Résultats de débruitage en PSNR par plusieurs filtres sur différents	
images test	.11

## Chapitre II

<b>Tableau II.1:</b> Résultats de débruitage (PSNR) par plusieurs techniques pour différentes	
variances de bruit.	23

## Chapitre III

<b>Tableau III.1:</b> Résultats du PSNR pour différentes images débruitées par plusieurs	
techniques	37
<b>Tableau III.2:</b> Seuils optimaux pour Barbara ( $\sigma$ =10) par la méthode Zhang [6]	38
<b>Tableau III.3:</b> Seuils optimaux pour Barbara ( $\sigma$ =10) par la méthode [7]	39

## Chapitre IV

<b>Tableau IV.1 :</b> Résultats du PSNR de différentes images débruitées par plusieurs	
techniques	48
<b>Tableau IV.2:</b> Seuils optimaux trouvés sur l'image Lena ( $\sigma$ =10) avec la méthode	
proposée	50
<b>Tableau IV.3:</b> Seuils optimaux trouvés sur l'image Barbara ( $\sigma$ =10) avec la méthode	
proposée	50
Tableau IV.4 : Résultats de débruitage (PSNR [dB]) sur différentes images de tests	
par toutes les méthodes de débruitage étudiées	56

## Liste des abréviations

TNN: Thresholding Neural Network (réseaux de neurone de seuillage).

MSE: Mean Squar Error (erreur quadratique moyenne).

**PSNR:** Peak Signal to Noise Ratio (rapport signal sur bruit en pic).

GLF: Gaussian Lowpass Filter (filtre Gaussien passe bas).

TOD: Transformée en ondelettes discrètes.

**Db:** Onde mère de Daubechies.

GA : Genetic algorithm (algorithme génétique).

# Introduction Générale

#### **Introduction Générale**

L'image recueillie en sortie de tout capteur d'image subit une dégradation engendrée par la chaîne d'acquisition. Celle-ci est représentée par la fonction de transfert du système. On peut distinguer les causes dites déterministes engendrant une perte d'information provoquée par une transformation déterministe comme un filtrage (e.g. lissage par la défocalisation d'une optique) ou une déformation (e.g. mouvement apparent d'une caméra), ou seule une portion de l'image est acquise [1]. Les autres causes sont dites stochastiques puisque les observations elles-mêmes sont des mesures physiques soumises à des fluctuations aléatoires dont les sources peuvent être le bruit du capteur, les fluctuations de la source lumineuse, ...etc [2].

D'une manière générale, le signal déterministe de l'image observée est contaminé par des fluctuations stochastiques que l'on qualifie généralement de bruit. Ce dernier peut être soit additif, soit multiplicatif. Les traitements de restauration sont souvent indispensables pour améliorer la qualité des images observées [2] [3]. Notre principal problème, que nous allons traiter dans ce travail, consistera à récupérer une image de bonne qualité, proche de l'image originale, à partir d'une image bruitée de mauvaise qualité. La figure 1 présente un exemple de débruitage d'image.

Dans la littérature du traitement d'images, différentes méthodes de débruitage ont été proposées et développées. Tout d'abord, des méthodes de filtrage spatial ont été proposées. Celles-ci consistent à réduire le bruit dans les zones qui ne présentent pas d'objets intéressants et à accentuer la perception des structures d'intérêt. Ces techniques de filtrage utilisent un filtre passe-bas pour supprimer les hautes fréquences, ce qui a pour inconvénient d'atténuer les contours de l'image [2]. Pour parer à ces problèmes, de nouvelles techniques, plus performantes, ont vu le jour; citons notamment les approches basées sur les transformées multi-échelles, notamment la transformée en ondelettes. Récemment, ces dernières ont montré leur puissance dans le cadre de l'estimation statistique [2]. Par le biais de ces transformées parcimonieuses, l'énergie du signal utile est concentrée sur un faible nombre de coefficients, ce qui offre ainsi un cadre naturel non linéaire pour estimer ce signal. En effet, il suffit de seuiller les coefficients de l'image observée et d'inverser la transformée pour obtenir une estimée du signal (c-à-d l'image) utile [2] [4].



Image bruitéeImage débruitéeFigure 1 : Exemple de débruitage d'image.

Avec les améliorations et les progrès remarquable dans le domaine de débruitage, récemment les techniques de l'intelligence artificielle ont ouvert d'autres portes vers l'optimisation et la perfection des résultats de l'image restaurée.

En 2001, Zhang a proposé une nouvelle structure basée sur les réseaux de neurones artificiels nommée les TNN (Thresholding Neural Network) [5] [6] dont le but était de rechercher le seuil optimal d'un vecteur monodimensionnel qui contient les coefficients en ondelettes de l'image bruitée afin de l'utiliser dans l'opération de seuillage et ceci dans le but de reconstituer l'image débruitée. En 2009, une nouvelle technique basée sur les TNN a été proposée [7], dans ce travail, l'auteur a proposé l'utilisation d'un TNN pour chaque sous bande de détails de la transformée en ondelettes, c'est-à-dire que chaque seuil est optimisé indépendamment des autres ce qui a engendré des résultats de débruitage très performants.

Inspirés par ces deux travaux, nous avons proposé une nouvelle technique basée sur les algorithmes génétiques et l'algorithme cycle spinning pour le débruitage des images fixes [8]. La première phase de notre méthode est l'optimisation des seuils en optimisant la fonction coût proposée en utilisant les algorithmes génétiques. La deuxième étape consiste à seuiller les coefficients de détails d'ondelettes par l'algorithme cycle spinning. Notre méthode présente plusieurs avantages par rapport aux autres techniques basées sur les TNN, ainsi elle donne des meilleures performances objectives et subjectives.

Dans le cadre ce mémoire, nous nous intéresserons principalement au problème de débruitage des images fixes en utilisant les techniques de l'intelligence artificielle dans le domaine de la transformée multi-échelles.

Le présent manuscrit est organisé en quatre chapitres. Dans le premier chapitre, nous allons donner un aperçu général sur le domaine du débruitage, en commençant par les concepts attachés au bruit, en suite nous présentons les techniques usuelles utilisées dans le but de restaurer les images contaminées par le bruit blanc gaussien (filtres). Dans le deuxième chapitre, les méthodes récentes de débruitage des images fixes dans le domaine de la transformée en ondelettes seront mentionnées. Dans le troisième chapitre, nous allons implémenter deux techniques basées sur les réseaux de neurones artificiels en l'occurrence les TNN (Thresholding Neural Network). Une méthode de débruitage des images fixes en utilisant les algorithmes génétiques pour optimiser les seuils de chaque sous bande est proposée dans le quatrième chapitre.

Nous notons aussi, que dans chaque chapitre nous allons présenter dans le cadre de ce travail une étude comparative des résultats de simulation obtenus par notre logiciel de simulation SDIF (Simulateur de débruitage des images fixes) réalisé sous MATLAB version 7.7. La description de toutes ses principales fonctionnalités est présentée dans l'annexe 3.

## Chapitre I

## Dégradation et Débruitage des Images

- 1. Introduction
- 2. Dégradation d'images
- 3. Débruitage d'images
- 4. Critère d'analyse pour le débruitage d'images
- 5. Les méthodes standards de débruitage
- 6. Plateforme informatique
- 7. Résultats expérimentaux
- 8. Résultats de débruitage des images contaminées par un bruit gaussien
- 9. Représentation des résultats
- 10. Discussion des résultats
- **11.** Conclusion

#### I.1.Introduction

Les traitements de restauration sont souvent indispensables pour améliorer la qualité des images observées. Notre principal problème, que nous allons traiter, consistera à récupérer une image de bonne qualité, proche de l'image originale, à partir d'une image bruitée de mauvaise qualité.

Le présent chapitre a pour objet de présenter le vocabulaire et les concepts attachés au domaine du débruitage d'images, tout en donnant un aperçu sur quelques types de bruits dégradant les images, avec une brève description de quelques techniques classiques (standard) en l'occurrence les filtres, utilisées pour le débruitage des images. Nous terminons ce chapitre par la présentation et la discussion des résultats de simulation obtenus.

#### I.2. Dégradation d'images

Dans la pratique, une image peut être dégradée par plusieurs types et formes de bruit, la source principale des bruits dans les images numériques survenues pendant l'acquisition (numérisation) et la transmission.

Les performances des capteurs d'imagerie est affectée par une variété de facteurs, tel que les conditions environnementale durant l'acquisition, et aussi par les capteurs d'image eux-mêmes. Par exemple, dans l'acquisition des images avec une caméra, les niveaux de lumière et la température du capteur sont des facteurs principaux sur la quantité de bruit dans l'image capturée [9].

Les images sont aussi corrompues lors de la transmission principalement dû à l'interférence dans le canal utilisé pour la transmission, à titre d'exemple, une image transmise en utilisant un réseau sans fil peut-être endommagée en raison de l'éclairage ou d'autres perturbations atmosphériques [9].

Donc, le bruit est un effet indésirable dans l'image, d'où la nécessité de le réduire ou le supprimer sans entraîner un changement significatif du détail et des caractéristiques de l'image [9].

Ainsi, il est important de connaître la nature du bruit contaminant l'image. En fait, le type de bruit le plus connu et le plus difficile à réduire, est le bruit blanc gaussien.

#### I.2.1. Bruit blanc gaussien

Le bruit blanc gaussien apparaît dans l'image comme une variation aléatoire autour des valeurs d'intensité. Il est modélisé par l'équation suivante:

$$f_{x}(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-m)^{2}}{2\sigma^{2}}},$$
 (1.1)

où  $f_x(x)$  est la densité de probabilité gaussienne (unidimensionnelle), m et  $\sigma^2$  représentent respectivement la moyenne et la variance. On suppose que le bruit est de moyenne nulle et de variance  $\sigma^2$ , donc la densité de probabilité devient [10]:

$$f_x(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\left(\frac{x^2}{2\sigma^2}\right)}.$$
 (1.2)

#### I.2.2. Bruit poivre et sel

Ce type de bruit est exprimé par le pourcentage du nombre total de pixels dans l'image, dont la valeur est 0 ou 255. Le bruit généré est distribué uniformément dans toute l'image [10].

#### I.2.3. Bruit multiplicatif

Le bruit multiplicatif ou impulsionnel est un phénomène commun dans tous les systèmes d'imagerie cohérents tels que le laser, l'acoustique, les ultrasons, et les images SAR (Synthetic Aperture Radar). Le bruit impulsionnel est souvent indésirable du fait qu'il dégrade l'image [10].

En général, le bruit impulsionnel est modélisé comme un bruit multiplicatif selon l'expression suivante:

$$y(k1, k2) = x(k1, k2) \cdot \varepsilon(k1, k2)$$
, (1,3)

où y est l'image bruitée, k1 et k2 sont les dimensions de l'image, x est l'image originale et  $\varepsilon$  c'est le bruit impulsionnel caractérisé par une distribution, avec une moyenne unité et de variance  $\sigma_{\varepsilon}^{2}$  [10].

#### I.3. Débruitage d'images

Le débruitage d'images a pour objectif principal d'atténuer, voire de supprimer dans les cas les plus favorables, les dégradations que subie une image. Le plus souvent, les dégradations considérées sont celles qui apparaissent au moment de l'acquisition de l'image, c'est-à-dire au moment de sa création. On pourrait y ajouter toutes les dégradations liées à la transmission, au changement de format ou au stockage de l'image. Le débruitage d'images essaye donc de recréer, à partir de l'image dont on dispose, l'image telle que l'on devrait l'avoir en situation idéale. Ce qui est visé, c'est la fidélité de l'image obtenue à la scène qu'elle représente [11].

Le besoin pour la suppression du bruit sans dégrader significativement les contours et autres composants de hautes fréquences de l'image, a donc motivé le développement des techniques efficaces qui lissent le bruit et conservent les contours. Dans la section suivante, quelques types de méthode standard de débruitage d'images qui sont appliquées dans le domaine spatial et fréquentiel, seront décrites.

#### I.4. Critère d'analyse pour le débruitage d'images

La pertinence des méthodes de débruitage dépend de deux critères, subjectif et objectif permettant d'évaluer l'efficacité de ces méthodes. Le critère subjectif représente l'aspect visuel.

#### I.4.1. Critère subjectif (aspect Visuel)

L'œil humain est un outil essentiel pour apprécier la qualité d'une image. Il va permettre à l'utilisateur d'identifier le contenu des images, la netteté de celles-ci, la présence d'artefacts et la qualité des contours. Il est donc capital que les méthodes de débruitage prennent en compte le système optique humain. Cependant, cette évaluation ne peut être que subjective puisqu'il n'existe aucune mesure correcte pouvant traduire fidèlement la perception de l'œil humain [9].

Sans pour autant négliger le critère subjectif, il est préférable d'introduire des critères objectifs. Les plus célèbres critères utilisés dans la littérature sont l'erreur quadratique moyenne (MSE) et le rapport signal sur bruit (PSNR).

#### I.4.2. Critère objectif

#### A. Erreur quadratique moyenne (MSE)

Bien qu'elle ne corresponde pas toujours avec la perception humaine, la MSE est souvent considéré comme une bonne mesure de la fidélité d'une évaluation d'image. Ce critère est défini comme suit [9] :

$$MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (X(i,j) - \hat{X}(i,j))^2, \qquad (1,4)$$

avec X: Image originale.

- $\widehat{X}$ : Image débruitée.
- *M* : Nombre de lignes de l'image.
- **N** : Nombre de colonnes de l'image.
- (*i*, *j*) : positionnement des pixels.

#### **B.** Rapport signal sur bruit en pic (PSNR)

Une autre mesure de la qualité de l'image approchée est le rapport signal sur bruit en pic (PSNR) qui est inversement proportionnel à la MSE, son unité est le décibel (dB) et est défini par la relation suivante [9]:

$$PSNR = 10 \log_{10} \left[ \frac{255^2}{MSE} \right] \quad dB,$$
 (1.5)

où **255** est la valeur maximale d'un pixel pour une image codée par 8 bits/pixel en niveaux de gris.

Ces deux critères (mesures) en plus de la qualité visuelle sont utilisés dans toutes les parties de simulations de ce mémoire comme mesures déterminant la qualité de l'image bruité et débruité.

#### I.5. Méthodes standards de débruitage

En générale les méthodes standards (classiques) de débruitage sont basées sur l'utilisation des filtres, on distingue alors [2]:

#### **I.5.1.** Filtrage spatial

L'évolution des techniques pour le débruitage d'images a commencé avec le développement des filtres spatiaux qui sont adaptés aux statistiques locales dans les petites sous régions centrées à un pixel qui doit être filtré.

Le filtrage spatial utilise donc l'information portée par le pixel, mais aussi par l'information contenue dans le voisinage de ce pixel. La notion de voisinage d'un pixel est assez générale et désigne l'ensemble des pixels qui sont en relation avec le pixel traité. Donc, il est d'usage de choisir un voisinage carré et symétrique autour du pixel considéré. Ces voisinages sont donc assimilables à des tableaux à deux dimensions (matrices) de taille impaire [11].

$$\begin{pmatrix} p_{x-1,y-1} & p_{x-1,y} & p_{x-1,y+1} \\ p_{x,y-1} & P_{x,y} & p_{x,y+1} \\ p_{x+1,y-1} & p_{x+1,y} & p_{x+1,y+1} \end{pmatrix}$$

Figure I.1 Exemple d'un voisinage 3x3 des pixels.

Nous distinguons dans cette catégorie les filtres suivants :

#### A. Filtre moyenneur

Appelé également *mean filtering, averaging* ou *Box filtering*. Son principe est très simple: un pixel est remplacé par la moyenne de lui-même et de ses voisins. C'est dans la définition du voisinage que les filtres vont différer. Le filtre moyenneur est un filtre passe bas permettant ainsi d'éliminer les hautes fréquences, correspondantes au bruit. Son inconvénient est qu'il élimine également les hautes fréquences correspondantes aux détails de l'image: il rend ainsi l'image moins bruitée mais plus floue [12].

#### **B.** Filtre médian

Le filtre médian (*median filter*) est un filtre non linéaire. Son principe consiste à remplacer un pixel par la médiane de ses voisins. Ainsi, même si plusieurs pixels voisins sont bruités, on peut corriger le pixel courant. Ce filtre induit cependant un lissage puisque même des pixels corrects peuvent être modifiés. De plus, ce filtrage est plus coûteux en temps de calcul car nécessite d'effectuer un tri des voisins pour chaque pixel. Plus le voisinage considéré est grand, plus l'algorithme sera coûteux [12].

#### I.5.2. Filtrage fréquentiel

L'utilisation de la représentation fréquentielle de l'image permet de réaliser ce type de traitement. L'approche la plus simple consiste à calculer, par transformée de Fourier, la décomposition en fréquence de l'image à traiter. On peut ensuite agir, d'une façon directe, sur les fréquences de l'image et cela de façon sélective.

Dans ce qui suit, nous allons traiter deux types de filtres qui utilisent cette technique [11].

#### A. Filtre Gaussien

Le filtre Gaussien passe bas (Gaussian Lowpass Filter) présente une alternative plus réaliste au filtre passe bas idéal qui met à zéro toutes les composantes dont les fréquences sont l'extérieur d'une gamme de basses fréquences spécifiées. Les effets du GLF sur le spectre d'une image sont semblables à ceux d'un filtre passe bas idéal dans le sens que les composantes à basses fréquences peuvent passer pendant que les composantes à hautes fréquences sont bloquées. La principale différence est que la troncature des composantes à hautes fréquences est graduelle et pas tranchante (raide), comme c'est le cas pour le filtre passe bas. Le GLF est représenté par la réponse de la fréquence suivante [9]:

$$H_G(w) = Ae^{-2\pi^2 \sigma^2 |w|^2}$$
, (1.6)

où,  $H_G(w)$  est la réponse fréquentielle, w est la pulsation,  $\sigma$  est la variance (ouverture du filtre) et A est l'amplitude.

#### **B.** Filtre de Wiener

Il est commun, quand on a affaire à des problèmes mal conditionnés, d'utiliser des techniques de régularisation. La solution s'obtient alors en employant un critère dont la solution optimale approxime la solution recherchée.

Dans le cas du filtre de Wiener, on recherche un filtre W(u,v) tel que l'espérance mathématique de l'erreur quadratique moyenne entre l'image idéale inconnue et l'image reconstituée soit minimale: ainsi on ne risque pas de faire diverger la solution [11] [13]. Le critère choisi est de nature probabiliste:

$$\min(E\left[\left(\left(f(x)-\hat{f}(x)\right)^2\right]\right), \qquad (1.7)$$

où f(x) est l'image originale idéale (sans bruit) et  $\hat{f}(x)$  est l'image reconstituée. La solution à cette expression dans le domaine fréquentiel est donnée par :

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)^2| + \frac{S_n(u,v)}{s(u,v)}} , \qquad (1.8)$$

où  $H^*(u, v)$ : *Est* la conjuguée de la fonction de dégradation H(u, v).

 $|H(\boldsymbol{u},\boldsymbol{v})|^2 = H^*(\boldsymbol{u},\boldsymbol{v}) H(\boldsymbol{u},\boldsymbol{v}).$ 

 $S_n(u, v)$ : Spectre de puissance du bruit n.

S(u, v) : Spectre de puissance de l'image idéale.

Le rapport  $S_n/S$  représente le rapport signal sur bruit. D'après l'équation de W(u,v), on peut constater que si  $S_n/S$  tend vers zéro alors le filtre de Wiener tend vers le filtre inverse[14].

Le problème est que les contours de l'image sont "écrasés" par le filtre de Wiener, car le critère utilisé est un critère objectif (quadratique). L'idéal serait évidemment d'utiliser un critère pseudo-visuel.

#### I.6. Plateforme informatique :

Tout au long de ce mémoire, les calcules numérique sont obtenus sur un ordinateur à base de Pentium Dual 2.16 GHz de fréquence et une RAM de 4Go.

L'implémentation des codes a été faite sur MATLAB 7.7, Il convient de noter que les programmes informatiques mis en œuvre ne sont pas forcément optimisés. Pour cela le temps d'exécution ne sera pas pris en considération dans ce travail.

#### I.7. Résultats expérimentaux

Puisque le débruitage d'un bruit de type poivre et sel ne présente pas de difficultés comme l'illustre la figure I.2 et à cause que le bruit impulsionnel est un cas spécial du bruit gaussien. L'étude dans ce mémoire a été consacrée au débruitage des images contaminées par le bruit blanc gaussien.



**Figure I.2:** (a) Image bruitée par un bruit poivre et sel à 5% (PSNR= 18.5170), (b) Image débruitée avec un filtre médian 3 x 3 (PSNR= 34.4944).

#### I.8. Résultats de débruitage des images contaminées par un bruit gaussien

Le tableau 1.1 montre les résultats de débruitage en PSNR obtenus en utilisant différents filtres classiques sur cinq images de test souvent utilisées dans la littérature.

Image	σ	Image	Filtre	Filtre	Filtre	Filtre de
_		bruitée	moyenneur*	médian <sup>*</sup>	gaussien**	Wiener2 <sup>*</sup>
					-	
Lena	10	28.1493	30.9183	32.1060	32.5122	33.5508
(512x512)	15	24.6101	29.8748	30.1022	30.6802	31.2013
	20	22.1175	28.7766	28.7622	29.3960	29.9611
	25	20.1735	27.6761	27.9377	28.4684	28.8824
	30	18.5857	26.8796	27.1408	27.7696	27.7802
Barbara	10	28.1102	24.7861	24.7615	29.7872	29.7587
(512x512)	15	24.6276	24.5226	24.2531	27.3903	28.1547
	20	22.0965	24.1632	23.7475	26.0503	26.6780
	25	20.1702	23.7573	23.1781	25.0240	25.6049
	30	18.6069	23.2811	22.5685	24.2420	24.8527
Boat	10	28.1420	28.9518	30.1650	31.6378	32.1973
(512x512)	15	24.6050	28.2768	28.7217	29.6225	30.2249
	20	22.0965	27.4787	27.3328	28.2908	28.4101
	25	20.1748	26.6096	26.0933	27.3025	27.3896
	30	18.6028	25.8082	25.4446	26.5617	26.5679
Cameraman	10	28.1449	24.9180	26.5184	29.8628	30.6020
(256x256)	15	24.6078	24.6449	25.7859	27.4725	29.0929
	20	22.1009	24.3129	25.0146	26.1206	27.6928
	25	20.1036	23.8527	24.2417	25.0120	26.2552
	30	18.5796	23.3844	23.3960	24.2368	25.3809

\* résultats obtenus par la meilleure taille du masque des filtres.

\* \* résultats obtenus par les meilleures valeurs de variance et taille du masque du filtre gaussien.

Tableau I.1: Résultats de débruitage en PSNR par plusieurs filtres sur différents images test.

#### I.9. Représentation des résultats

La figure 1.3 illustre bien le critère PSNR (dB) en fonction de la variance de bruit pour chaque type de filtre et pour chaque image de test.



**Figure I.3:** Représentation des résultats de débruitage en PSNR en fonction de la variance de bruit sur les images test par différents filtres.

#### I.10.Discussion des résultats :

Les figures I.4 et I.5 montrent respectivement les résultats de débruitage par les filtres spatiaux moyenneur et médian, avec différentes tailles du masque sur l'image Lena.







3x3 (PSNR=27.6804)



Image bruitée (PSNR=20.1793)



5x5 (PSNR=27.3084)





7x7 (PSNR=25.9722) 9x9 (PSNR=24.8689) Figure I.4: Image « Lena » débruitée par un filtre moyenneur avec différentes tailles du masque.



Image originale



Image bruitée (PSNR=20.1793)



3x3 (PSNR= 26.8177)



5x5 (PSNR= 27.9014)



7x7 (PSNR= 27.2091)9x9 (PSNR= 26.3460)Figure I.5: Image « Lena » débruitée par un filtre médian avec différentes tailles du masque.

Les figures (I.4 et I.5) montrent que les performances des filtres moyenneur et médian dépendent principalement de la taille du masque utilisé, un petit masque ne permet pas la réduction du bruit, par contre l'utilisation d'un grand masque génère un effet de flou dans l'image, c'est-à-dire une image très lisse. Les résultats obtenus montrent que les meilleurs résultats pour « Lena » avec  $\sigma$ =25 sont obtenus par un filtre médian avec un masque de 5x5 et avec un filtre moyenneur par un masque de 3x3.

Les résultats (tableau I.1) et (figure I.6) montrent que le filtre gaussien donne de meilleurs résultats que les filtres moyenneur et médian parce qu'il est bien adapté au filtrage du bruit gaussien. L'inconvénient de ce filtre qu'il dépend de deux critères, la largeur de la gaussienne (variance) et le masque utilisé.

Généralement les résultats sont flous parce que le filtre gaussien supprime également les hautes fréquences correspondantes aux détails de l'image.



Image originale



Image bruitée



Moyenneur (PSNR= 27.6804)



Médian (PSNR= 27.9014)



Gaussien (PSNR= 28.5367)

Wiener2 (PSNR= 28.8054)

Figure I.6: Image « Lena » débruitée par différents filtres.

Le filtre de Wiener permet un lissage très grand ou voisinage des contours et un lissage faible dans les régions qui ne contiennent pas de détails, en plus, des artefacts sont visibles à proximité des contours comme le montre la figure I.6.

La comparaison entre les différents filtres étudiés, montre que le filtre de Wiener est le meilleur filtre quelque soit dans les valeurs du PSNR (voir Tableau I.1) ainsi que la qualité de l'image débruitée (voir figure I.6) pour toutes les images de test et pour n'importe qu'elle variance de bruit.

#### I.11. Conclusion

Un aspect essentiel de traitement d'images concerne l'analyse et la compréhension de l'image dans le but d'en extraire des informations utiles. C'est le domaine de débruitage, objet de notre travail. Des notions attachées à la dégradation des images, et au débruitage en particulier ont été abordées dans ce chapitre d'une manière succincte. En effet, ce chapitre à permis de donner un aperçu sur quelques méthodes classiques de débruitage. Une étude comparative a été menée pour montrer les différences qui peuvent exister entre ces méthodes.

Le chapitre prochain traitera d'une façon particulière les méthodes récentes pour le débruitage des images fixes, à savoir les techniques basées sur la transformée en ondelettes.

## Chapitre II

## Méthodes Récentes pour le Débruitage des Images

- 1. Introduction
- 2. Débruitage d'image contaminée par un bruit blanc gaussien
- 3. Estimation de la variance du bruit
- 4. Le seuillage
- 5. Seuillage de D. L. Donoho
- 6. Seuillage "VisuShrink"
- 7. Seuillage dépendant-niveau « Level Shrink »
- 8. Seuillage " BayesShrink"
- 9. Seuillage "NormalShrink"
- **10. Résultats de simulations**
- 11. Conclusion

#### **II.1. Introduction**

Dans le chapitre précédent, quelques méthodes standard de débruitage appliquées dans le domaine spatial aussi bien que dans celui du domaine fréquentiel de l'image bruitée, ont été décrites. Ces méthodes présentent des limitations soit pour la réduction de bruit gaussien soit pour la qualité de l'image restaurée qui est généralement floue [9].

En effet, il y a eu un progrès considérable dans le développement des nouvelles méthodes pour le débruitage d'images qui s'exécutent considérablement mieux que les méthodes standards, décrites dans le chapitre précédent. Parmi ces méthodes, on trouve celles qui sont basées sur la transformée en ondelettes [plus de détailles voir Annexe 1] et plus particulièrement sur l'utilisation du seuillage [9]. Ces techniques sont basées sur le seuillage dans le sens que chaque coefficient d'ondelettes de l'image est comparé à un seuil donné; si le coefficient est plus petit que le seuil, alors il est mis à zéro, autrement il est inchangé ou son amplitude est réduite légèrement. Plus de détails concernant ces techniques seront donné dans la partie qui suit.

Dans le présent chapitre, nous présentons le principe de base de débruitage par seuillage, une description des techniques récentes de restauration d'image et nous terminerons par une étude comparative des résultats de simulation.

#### II.2. Débruitage d'image contaminée par un bruit blanc gaussien

Dans le cas du bruit gaussien additif, le problème consiste à déterminer une fonction réelle notée f, à partir des mesures g connues telles que :

$$g = f + \varepsilon$$
, (2.1)  
avec  $\varepsilon \sim N(0, \sigma^2)$ 

Avec une transformée discrète en ondelettes [Annexe 1], le bruit blanc gaussien se décompose en une série de coefficients aléatoires normaux centrés et décorrélés [14]. En conséquences, la transformée en ondelettes discrète des mesures définies dans l'équation (2.1) peut s'écrire :

$$W_g = W_f + W_{\varepsilon}$$
, (2.2)

ou, W symbolise la transformée en ondelettes discrète.

Afin de reconstituer la fonction f à partir des mesures g, tous les coefficients de la décomposition  $W_g$  sont rétrécis par un seuillage dépendant de la contribution du bruit. Alors dans ce qui suit nous allons décrire plusieurs types de seuillage [14].

#### II.3. Estimation de la variance du bruit

Pour réduire le bruit dans une image, il est nécessaire d'avoir une estimation de la variance  $\sigma^2$  du bruit. Pour l'estimer à partir des données *g*, il faut supprimer l'influence de *f*.

En conséquence, l'écart-type est estimé dans une bande où le signal informatif est considéré inexistant ou quasi-inexistant, notamment dans les plus hautes fréquences. Dans la littérature beaucoup d'auteurs utilisent un estimateur robuste à partir de la valeur médiane des coefficients en ondelettes [Annexe 1] de la sous bande de détail diagonale du premier niveau de décomposition [9].

$$\widehat{\sigma}^2 = \left[\frac{median(|Y_{i,j}|)}{0.6745}\right]^2, Y_{i,j} \in subband HH_1, \qquad (2.3)$$



avec  $Y_{i,j}$ : Représente les coefficients de détails diagonal du premier niveau de décomposition.

**Figure II.1:** (a) Décomposition en ondelettes d'une image, (b) Histogramme de la sous bande HH<sub>1</sub> (coefficients diagonales du premier niveau de décomposition).

#### II.4. Le seuillage (thresholding)

Le seuillage est une technique utilisée pour le débruitage des signaux et des images. La transformation en ondelettes discrète (T.O.D) [Annexe 1] utilise deux types de filtrage, le filtrage passe bas et le filtrage passe haut. Quand nous décomposons un signal à l'aide de la transformation en ondelettes nous obtenons une suite de coefficients qui corrèlent avec les sous bandes de hautes fréquences [15]. Ces sous bandes de hautes fréquences sont les détails des données. Si ces détails sont assez petits ils peuvent être négligés sans pour autant affecter les données de base.

En plus, ces détails sont souvent associés directement au bruit et en remplaçant ces coefficients par zéro on diminue les composantes de bruit [16]. Dans un but d'optimisation on pourra représenter le spectre de l'image afin de pouvoir déterminer les composantes les plus importantes de ce dernier et donc celles que l'on doit garder. Ceci est la base du seuillage que nous pouvons schématiser comme l'indique la figure II.2 [17].



Figure II.2 : Les trois étapes du seuillage des coefficients en ondelettes

D'où :

x: Image sans bruit (tell qu'elle est obtenue dans les conditions optimales).

W : Bruit blanc gaussien.

y : Image bruitée.

T.O.D : transformée discrète en ondelettes.

Y : Coefficients bruités.

 $T(Y, \lambda)$ : fonction de seuillage avec un seuil  $\lambda$ .

- $\hat{X}$ :Coefficients débruités (seuillés).
- $\hat{x}$  : image débruitée.
#### II.5. Seuillage de D.L Donoho

Donoho a proposé deux types de seuillage des coefficients d'ondelettes dénotés par seuillage doux (Soft thresholding) et seuillage dur (Hard thresholding) [9] [18].

#### II.5.1. Seuillage doux « Soft thresholding »

Dans le cas de seuillage doux (voir figure II.3), on met toujours à zéro les coefficients inférieurs à un seuil  $\lambda$ . Par contre, pour ceux supérieurs à  $\lambda$ , on atténue l'amplitude des coefficients par la valeur du seuil afin de s'assurer d'avoir enlevé l'effet du bruit même pour les grands coefficients [18] [9].

$$\theta(x) = \begin{cases} 0 & si |x| < \lambda \\ (x - \lambda)sign(x) & si |x| \ge \lambda \end{cases}$$
 (2.4)

où  $\theta(x)$  est la fonction de seuillage, x sont les coefficients de détails de la transformée en ondelettes et  $\lambda$  représente le seuil.



Figure II.3 : Seuillage doux.

Le coefficient seuillé sera donc plus petit que le coefficient du signal (figure II.3). Ce type de seuillage garantit que le signal obtenu sera toujours plus régulier que le signal de départ. L'inconvénient de ce type de seuillage est que la possibilité d'erreur due au choix du seuil est plus grande qu'avec le seuillage dur [9].

#### II.5.2. Seuillage dur « Hard thresholding »

Le seuillage dur est celui qui est le plus « intuitif » (voir figure II.4). On se fixe un seuil par exemple  $\lambda > 0$ . On ne conserve que les coefficients d'ondelettes supérieurs à  $\lambda$  et on met à zéro les autres [9] [18]:

La fonction seuillage dur est définie par l'équation suivante :

$$\theta(x) = \begin{cases} 0 & si |x| < \lambda \\ x & si |x| \ge \lambda \end{cases}, \quad (2.5)$$

où,  $\theta(x)$  est la fonction de seuillage, x sont les coefficients de détails de la transformée en ondelettes et  $\lambda$  représente le seuil.



Figure II.4: Seuillage dur.

#### II.6. Seuillage "VisuShrink"

La méthode VisuShrink introduite par Donoho et Johnstone en 1994 [16]. Cette méthode propose un seuil universel déterminé à partir de l'énergie estimée du bruit. Cette méthode utilise l'hypothèse d'un bruit blanc gaussien superposé au signal.

Le seuil proposé est égal à :

$$\lambda = \sigma \sqrt{2 \log (M)} , \qquad (2.6)$$

où M désigne le nombre de points du signal (taille de l'image) et  $\sigma$  la variance du bruit.

Ce seuil est performant si le signal informatif n'est pas très riche en détails, c'est à dire si ses coefficients sont rares.

Autrement dit, le seuillage risque d'être trop fort pour un signal quelconque. Un autre problème posé par le seuillage universel est la valeur unique du seuil pour toutes les bandes de fréquence de la décomposition en ondelettes.

#### II.7. Seuillage dépendant-niveau « Level Shrink »

Comme est mentionné dans la section précédente, le seuillage soft ou hard adopte le seuil universel à être utilisé uniformément partout dans l'arbre de la décomposition d'ondelettes de l'image bruitée. Intuitivement, dû à la haute variabilité des coefficients d'ondelettes à travers les différents niveaux de la décomposition, il serait plus raisonnable, et peut-être plus efficace, d'utiliser un seuil adapté à chaque niveau de décomposition [9].

Il a été prouvé qu'une telle adaptation dans la sélection du seuil a tendance à améliorer la performance du seuillage en ondelettes parce qu'il s'adapte aux statistiques locales supplémentaires de l'image, tel que le lissage ou les contours. Ces observations sont logiques avec la nature de processus adaptatifs qui caractérisent les statistiques locales du signal.

Dans cette section, une telle technique du seuillage en ondelettes dépendante du niveau de décomposition sera étudiée [9].

L'algorithme du seuillage dépendant du niveau, appelé « LevelShrink », propose l'usage d'un seuil différent pour chaque niveau de l'arbre d'ondelettes.

Le plan du seuillage niveau-dépendant, consiste à adapter le seuil au *j*ème niveau de décomposition de l'arbre d'ondelettes comme suit [9]:

$$\lambda_{j} = \sqrt{2 \log (M)} * 2^{-(J-j)/2} = \lambda_{univ} * 2^{-(J-j)/2} , \qquad (2.7)$$

pour J = 1, 2, ..., J. J est le niveau maximal de décomposition en ondelettes,  $\lambda_{univ}$  est le seuil universel et M désigne le nombre de points du signal (image).

#### II.8. Seuillage " BayesShrink"

Le BayesShrink [19] utilise une structure mathématique Bayesian des images pour tirer des seuils spécifiques pour chaque sous-bande. La formule de calcul du seuil est donnée par :

$$T_B = \frac{\widehat{\sigma}_n^2}{\widehat{\sigma}_x} , \qquad (2.8)$$

où  $\hat{\sigma}_n^2$  est la variance estimée du bruit et  $\hat{\sigma}_x$  est l'écart-type estimé de signal.

$$\widehat{\sigma}_x = \sqrt{max(\widehat{\sigma}_y^2 - \widehat{\sigma}_n^2, \mathbf{0})} \quad , \qquad (2.9)$$

où  $\hat{\sigma}_{y}^{2}$  est la variance estimée du signal (image) observé qui s'écrit :

$$\hat{\sigma}_{y}^{2} = \frac{1}{M^{2}} \sum_{i,j=1}^{M} Y_{ij}^{2} , \qquad (2.10)$$

où M désigne la taille de la sous bande et  $Y_{i,j}$  les coefficients de la sous bande en considération.

#### II.9. Seuillage "NormalShrink"

Un autre travail de qualité cité dans la littérature, propose une méthode d'évaluation adaptative du seuil pour le débruitage d'image par ondelette basé sur la distribution Gaussienne généralisée. La méthode proposée, appelée NormalShrink [20], est adaptative parce que le paramètre  $\beta$ , exigé pour évaluer le seuil, dépend des données des sous- bandes. Le seuil est calculé alors par :

$$T_N = \frac{\beta \hat{\sigma}_n^2}{\hat{\sigma}_x} \,, \tag{2.11}$$

où  $\beta$  est le paramètre d'échelle qui dépend de la taille de la sous-bande et du niveau de décompositions. Il s'écrit:

$$\beta = \sqrt{log\left(\frac{L_k}{J}\right)} , \qquad (2.12)$$

avec k = 1, 2, ..., J.  $L_k$  est la longueur de la sous-bande à la  $k^{ime}$  échelle,  $\hat{\sigma}_n^2$  est la variance estimée du bruit,  $\hat{\sigma}_x$  est l'écart-type estimé du signal, et J c'est le nombre maximal des décompositions.

#### II.10. Résultats de simulation :

Lors de l'implémentation des techniques de débruitage motionnées précédemment, les meilleures propriétés et les configurations originales on été utilisées (exactement comme dans les articles publier dans la littérature et cités dans la partie bibliographique de ce mémoire).

Pour les techniques VisuShrink et LevelShrink (soft et hard), nous allons utiliser 4 niveaux de décomposition avec l'onde mère Db8 (Daubechies).

Pour les techniques BayseShrink et NormalShrink : nous allons utiliser 4 niveaux de décomposition en ondelette avec l'onde mère Db8 et l'application d'une fonction de seuillage doux (soft thresholding).

#### II.10.1. Résultats de débruitage :

Le tableau II.1 montre les résultats de débruitage (PSNR) par les différentes méthodes récentes basées sur la transformée en ondelettes sur les images de références et avec différentes variances de bruits blanc gaussien.

Image	σ	Image	VisuShrink	VisuShrink	LevelShr	LevelShr	Wiener2	NormalShri	BayseShrink
_		Bruitée	Soft	Hard	ink (soft)	ink	(3x3)	nk	
				(hard)					
Lena	10	28.1538	28.0664	30.4323	30.2706	32.0832	33.5508	33.5473	33.4404
(512x512)	15	24.6113	26.6738	28.6763	28.9771	30.4730	31.1125	31.6357	31.6148
	20	22.0965	25.7490	27.4602	28.1072	29.2816	28.9920	30.2804	30.3188
	25	20.1931	25.0418	26.5795	27.3521	28.3786	27.2295	29.1985	29.3731
	30	18.5878	24.6298	25.9068	26.8220	27.6339	25.7290	28.3901	28.6018
Barbara	10	28.1291	24.9862	27.6043	26.1164	28.6369	29.7587	31.8453	31.5230
(512x512)	15	24.6076	23.5463	25.3221	24.6523	26.2557	28.1547	29.3953	29.3361
	20	22.1232	22.8624	24.0434	23.9361	24.9022	26.6780	27.6362	27.7740
	25	20.1652	22.4062	23.3113	23.4854	24.0861	25.3770	26.3962	26.5559
	30	18.5792	22.1640	22.8309	23.2321	23.6433	24.2290	25.4184	25.6210
Boat	10	28.1278	26.3593	28.6612	28.2589	30.1073	32.1973	32.3771	32.4387
(512x512)	15	24.6364	25.0501	26.9204	26.9657	28.4050	30.2249	30.2594	30.2710
	20	22.1247	24.1546	25.8080	26.1533	27.2973	28.4101	28.7873	28.8722
	25	20.1715	23.5088	24.9659	25.5209	26.4911	26.7963	27.7174	27.8366
	30	18.5729	23.0671	24.3340	25.0079	25.8712	25.3963	26.9061	27.0236
Cameram	10	28.1449	24.6287	27.3462	26.0796	28.3064	30.5864	31.1806	31.1318
an	15	24.6078	23.2068	25.4601	24.8398	26.5568	29.0860	28.7392	28.6142
(256x256)	20	22.1009	22.3331	24.0969	23.9338	25.3723	27.5811	27.1981	27.0871
	25	20.1036	21.6377	23.2452	23.2879	24.6793	26.2126	26.0078	25.8961
	30	18.5796	21.0822	22.5278	22.8476	23.9186	24.9617	25.1655	25.0992

**Tableau II.1:** Résultats de débruitage (PSNR) par plusieurs techniques pour différentes variances de bruit.

#### II.10.2. Echantillons des images débruitées :

Les résultats du débruitage sur les images Lena et Cameraman avec les différentes méthodes étudiées précédemment sont présentées sur les figure II.5 et II.6.



Image originale



VisuShrink (soft)



LevelShrink (soft)



Image bruitée



VisuShrink (hard)



LevelShrinl (hard)

NormalShrink



Wiener2 (filtre)



BayseShrink

Figure II.5: Résultats du débruitage sur l'image Lena ( $\sigma$ =20) avec différentes méthodes de débruitage.



Image originale



Image bruitée



VisuShrink (soft)



VisuShrink (hard)



LevelShrink (soft)



LevelShrink (hard)



Wiener2



NormalShrink



BayseShrink

**Figure II.6**: Résultats du débruitage sur l'image Cameraman (σ=20) avec différentes méthodes de débruitage.

#### II.10.3. Représentation des résultats

La figure II.7 représente graphiquement les résultats obtenus (valeurs du PSNR (dB))

par les différentes méthodes mentionnées précédemment sur les images de références.



**Figure II.7**: Représentation graphique des résultats en PSNR sur les images de références pour différentes variances de bruit par différentes techniques de débruitage.

#### II.10.4. Discussion des résultats

L'observation du tableau II.1 et de la figure II.7 montre l'efficacité des méthodes de débruitage récentes basées sur la T.O.D en comparaison avec la méthode de Wiener (meilleure technique dans la famille des filtres) et surtout lorsque la variance du bruit est élevée cela peut être interprété par le fait que les méthodes de débruitage récentes utilisent les propriétés statistiques de l'image bruitée dans plusieurs niveaux de résolution (meilleure représentation de l'image par la transformée en ondelettes).

Une comparaison entre les deux techniques VisuShrink et LevelShrink (Tableau II.1) (figure II.7) permet de distinguer l'efficacité de la technique LevelShrink, cela peut être interprété par le fait que dans le VisuShrink un seul seuil (seuil universal) est utilisé pour tous les niveaux de l'arbre de décomposition, par conséquence l'image reconstituée est généralement très floue (voir figure II.6), par contre pour la technique levelShrink, chaque niveau de décomposition à son propre seuil (LevelShrink utilise des seuils plus large pour des niveaux les plus fins et des seuil plus petits pour les niveaux les plus grands), donc une image reconstituée moins floue que le visuShrink (voir figure II.5), parce qu'en réalité les propriétés statistiques varient d'un niveau à un autre et d'une sous bande à une autre, donc on peut conclure que ces résultats peuvent être améliorés si le seuil est adapté à chaque sous bande.

L'examen aussi des résultats (figure II.5 et II.6) montre que l'utilisation d'une fonction de seuillage doux donne des résultats lisses et de petites valeurs de PSNR, par contre l'utilisation de la fonction de seuillage dur donne des meilleurs PSNR mais à cause de la discontinuité de la fonction de seuillage les résultats sont instables.

Il est à noté que l'utilisation des fonctions de seuillage proposée par D.L Donoho permettent l'apparition du phénomène de Gibbs « pseudo-Gibbs artifacts » [9] comme le montre la figure II.8.



Figure II.8 : Phénomène de Gibbs sur une image débruitée par NormalShrink et BayseShrink.

Les résultats de simulation montrent la supériorité des deux techniques NormalShrink et BayseShrink (meilleure qualité d'image reconstruite, meilleure conservation des contours et des meilleurs PSNR) (voir figures II.5, II.6, II.7), cela peut être interprété par l'utilisation des seuils adaptés pour chaque sous bande.

La technique bayseShrink donne des meilleurs résultats à cause de la relation qui existe entre le seuil utilisé et le rapport signal sur bruit (SNR). L'utilisation de ce seuil permet un petit seuillage dans les régions des contours et un grand seuillage dans les autres régions, ce qui permet une meilleure conservation de l'information et une meilleure qualité visuelle de l'image reconstituée.

#### **II.11. Conclusion**

Dans ce chapitre nous avons présenté plusieurs techniques de débruitage d'images corrompues par du bruit blanc gaussien. En effet, ces techniques trouvent leur succès dans les stratégies d'organisation et de représentation des cœfficients d'ondelettes par le biais de l'analyse multi-résolution. Cette dernière est d'une grande utilité pour les méthodes de débruitage par ondelettes consistant à modifier les coefficients de détails afin de restaurer une image utile à partir d'observations corrompues par un bruit additif.

Au cours de cette étude quatre techniques de débruitage ont été implémenté, notées VisuShrink (soft et hard), LevelShrink (soft et hard), NormalShrink et BayseShrink.

Les résultats obtenues a partir de l'implémentation des quatre techniques, nous permettent de conclure que l'efficacité d'une technique de débruitage basée sur la transformé en ondelettes dépend fortement du choix du seuil pour chaque sous bande de coefficients d'ondelettes. Dans ce contexte, dans le prochain chapitre, nous allons utiliser les méthodes d'optimisation en l'occurrence les réseaux de neurones pour l'obtention des seuils optimaux.

## Chapitre III

# Débruitage par TNN "Thresholding Neural Network"

#### **1. Introduction**

- 2. Structure du TNN (thresholding neural network)
- 3. Débruitage des images par TNN
- 4. Résultats de simulation
- 5. Courbes d'apprentissage
- 6. Echantillons d'images débruitées
- 7. Représentation des résultats
- 8. Discussions
- 9. Conclusion

#### **III.1. Introduction**

Les réseaux de neurones formels sont devenus en quelques années des outils précieux dans des domaines très divers de l'industrie et des services [21]. L'intérêt des réseaux de neurones, dans le domaine des Sciences de l'Ingénieur, ne doit rien à la métaphore biologique : il est uniquement dû aux propriétés mathématiques spécifiques de ces réseaux.

Comme nous l'avons montré dans le chapitre précédent, l'efficacité de débruitage par seuillage dépend de la fonction de seuillage utilisée ainsi que le bon choix du seuil. Visant cet objectif, Xp Zhang a proposé la structure du TNN (thresholding neural network) pour le débruitage des images fixe dans le domaine de la transformée.

Le TNN existe en deux modes, le mode supervisé et le mode non supervisé. Dans ce travail nous allons s'intéresser qu'au mode (apprentissage) supervisé.

L'objectif de ce chapitre est de donner la structure générale du TNN et de montrer l'implémentation des deux algorithmes de débruitage basés sur ce dernier.

#### III.2. Structure du TNN (Thresholding Neural Network)

Le TNN a été conçu pour le seuillage dans le domaine de la transformée pour la réduction du bruit, la structure de TNN est représentée sur la figure III.1:



Figure III.1: Structure du TNN

L'entrée du TNN est les échantillons du signal corrompu par le bruit,  $y_i = x_i + n_i$ ,  $i = 0 \dots, N - 1$ , avec x est le signal original (désiré) et n est le bruit additif. La transformée dans le TNN peut être n'importe quelle transformée linéaire et orthogonale. La sélection de la

transformée à utiliser doit être choisie de telle sorte qu'elle permette la concentration de l'énergie du signal par-rapport au bruit par le seuillage. L'énergie du signal est donc conservée et le bruit est supprimé. La fonction du seuillage n(x,t) est utilisée comme fonction d'activation pour le réseau de neurones. La transformée inverse permet la reconstitution du signal original (image de référence) à partir des coefficients  $\hat{v}_i$  [5].

Dans ce nouveau type des réseaux de neurones (TNN), la fonction d'activation est remplacée par la fonction de seuillage, l'apprentissage se fait sur les seuils et non pas sur les poids comme dans les réseaux de neurones classique tel que le perceptron. En d'autres termes, les poids sont fixés à une valeur de 1 et la fonction d'activation aura une structure variable comme elle peut subir un apprentissage [7] [6].

On considère le vecteur  $\mathbf{x} = [x_0, x_1, ..., x_{N-1}]^T$  corrompu par le bruit additif,  $x_i = v_i + n_i$ , i = 0 ..., N - 1, le but du TNN est de minimiser l'erreur quadratique moyenne (MSE) [7],

$$J_{mse} = \frac{1}{2} E \| \widehat{V} - V \|^2 = \frac{1}{2N} \sum_{i=0}^{N-1} (\widehat{v}_i - v_i)^2 , \qquad (3.1)$$

 $\widehat{V} = [\widehat{v}_0, \widehat{v}_2, ..., \widehat{v}_{N-1}]$ : Les coefficients à la sortie de la fonction du seuillage.

 $V = [v_0, v_2, ..., v_{N-1}]$ : Les coefficients du signal original (sans bruit).

Dans le TNN, le seuil (threshold) t est adaptivement ajusté pour la fonction de seuillage non linéaire afin de minimiser l'erreur (MSE). Comme l'image originale n'est pas connue (elle représente le but du débruitage), les coefficients  $v_i$  ne peuvent être utilisés pour l'estimation du MSE. Pour cela, une solution pratique est adoptée et c'est l'utilisation de deux signaux y et y' obtenus du même signal x avec deux bruit non corrélées n et n'. y' est utilisé alors comme référence. La solution adoptée est raisonnable parce qu'on peut avoir plusieurs versions du signal bruité dans plusieurs applications.

Zhang a prouvé que l'utilisation d'un deuxième signal bruité comme référence pour le calcul du MSE (apprentissage) conduit à un même seuil optimal qu'avec l'image originale [5] [6].

#### Apprentissage du TNN

L'apprentissage du TNN peut être en mode supervisé ou non supervisé, dans ce travail nous allons nous baser sur l'apprentissage supervisé.

L'apprentissage du TNN revient à l'estimation du seuil, c'est-à-dire à déterminer le seuil optimal qui permet de minimiser l'erreur MSE. Pour cela, Zhang a proposé l'utilisation de la technique de descente de gradient stochastique illustré par la figure III. 2 [5] [6]:



Figure III.2 : Apprentissage stochastique du TNN.

La règle d'apprentissage pour estimer le seuil est :

$$t(i+1) = t(i) + \Delta t(i)$$
, (3.2)

où t(i + 1) est le nouveau seuil et t(i) est l'ancien seuil.

Le seuil à la position *i* est ajusté par :

$$\Delta t(i) = -\alpha(i) \cdot \frac{\partial \hat{v}_i}{\partial t} \cdot \varepsilon(i) , \qquad (3.3)$$

 $i = 1, \dots, N$  et  $\varepsilon(i) = \hat{v}_i - \dot{v}_i$ .

 $\dot{v}_i$ : Coefficients du signal de référence [5].

Il est à noter que la transformée orthogonale peut avoir plusieurs canaux (sous bandes dans le cas de la TOD [Annexe 1]). Zhang a mentionné que chaque canal peut avoir son propre seuil et que chaque seuil sera ajusté séparément [6] [22].

#### III.3. Débruitage des images par TNN

## III.3.1. Méthode proposée par Zhang [6] :« Thresholding neural network for adaptive noise reduction »

Zhang dans [6] a utilisé le TNN pour le débruitage des images en proposant l'utilisation de la transformée en ondelettes comme transformée linéaire orthogonale du TNN.

Ce choix a été élaboré à cause des propriétés intéressantes de la TOD, c'est à dire la concentration de l'énergie du signal face au bruit.

#### Fonction d'activation (fonction de seuillage):

Il est connu que les réseaux de neurones doivent avoir un très bon algorithme d'apprentissage, généralement c'est la technique de la descente de gradient où la fonction d'activation peut admettre des dérivées supérieures. La fonction de seuillage douce (soft) peut avoir des dérivées supérieures quand à la fonction de seuillage dure (hard) ne peut en avoir (à cause des discontinuités). Zhang a proposé l'utilisation d'une nouvelle fonction de seuillage douce [5] [6] représentée sur la figure III.3 :

$$\eta_{\lambda}(x,t) = x + \frac{1}{2} \left( \sqrt{(x-t)^2 + \lambda} - \sqrt{(x+t)^2 + \lambda} \right), \qquad (3.4)$$

où *t* est le seuil, et  $\lambda$  est un paramètre à fixer. Cette fonction a des dérivées d'ordre supérieures pour  $\lambda > 0$ , pour  $\lambda = 0$  c'est une fonction de seuillage douce (soft).



Figure III.3 : Les fonctions de seuillage proposées par Zhang [5] [6].

Zhang a proposé l'utilisation de trois niveaux de décomposition en ondelette avec l'onde mère Db8 et avec  $\lambda$ =0.01 dans la fonction de seuillage.

Dans cette technique, nous avons utilisé un TNN pour chaque sous bande, c'est-à-dire que chaque sous bande aura un seuil indépendant des autres sous bandes.

## III.3.2. Méthode Proposée dans [7] (New TNN) :« Image denoising in the wavelet domain using a new adaptive thresholding function »

Dans la méthode proposée par Zhang dans ses articles [5] et [6], un seul TNN est utilisé, dont l'entrée est un vecteur 1D qui contient les coefficients en ondelettes des trois niveaux de décomposition.

Après l'opération d'apprentissage, le seuil est déterminé, et la reconstruction de l'image sera obtenue par la transformée inverse en ondelettes.



Figure III.4 : Nouveau type de TNN proposé dans [7].

Dans cette technique, une nouvelle structure de débruitage par TNN est proposée par l'auteur (voir Figure III.4). En effet, chaque sous bande utilise son propre TNN, c'est-à-dire que chaque seuil dépend seulement de la sous bande qui le concerne (subband-adaptive).

#### a) Fonction d'activation (fonction de seuillage)

L'auteur Mehdi et al. [7] ont proposé une nouvelle fonction de seuillage présentée dans la figure III.5 :

$$\eta(x, thr) = \begin{cases} x - 0.5 \frac{thr^2}{x} & |x| > thr \\ 0.5 \frac{x^3}{thr^2} & |x| \le thr \end{cases}$$
(3.5)

avec , thr est le seuil à déterminer, et x est les coefficients de détais de chaque sous bande.



Figure III.5: Les fonctions de seuillage proposées dans [7].

#### b) Algorithme d'apprentissage:

L'auteur a proposé l'utilisation d'un algorithme LMS (least mean squares). A chaque itération, le seuil est ajusté par la technique de la descente de gradient de l'erreur MSE. La valeur du seuil à l'étape j pour la sous bande p est calculée par la règle d'apprentissage suivante:

$$thr_{p}(j+1) = thr_{p}(j) + \Delta thr_{p}(j), \qquad (3.6)$$

avec

$$\Delta thr_p(j) = -\propto \frac{\partial J_{MSE}}{\partial thr}\Big|_{thr=thr_p(j)} , \qquad (3.7)$$

où  $\alpha$  est le coefficient d'apprentissage,  $J_{MSE}$  est le risque MSE (erreur quadratique moyenne).

#### Apprentissage supervisé :

Dans ce mode d'apprentissage, une autre image bruitée de référence est utilisé:

$$\Delta thr_{p}(j) = -\alpha \frac{\partial J_{MSE}}{\partial thr}\Big|_{thr=thr_{p}(j)}, \qquad (3.8)$$
$$= -\alpha \sum_{i=0}^{M-1} \varepsilon_{i} \frac{\partial \widehat{v}_{i}}{\partial thr}\Big|_{thr=thr_{p}(j)}, \qquad (3.9)$$

où M est la longueur de la sous bande p et  $\varepsilon_i$  est l'erreur de seuillage entre le coefficient seuillé et le coefficient bruité (référence).

$$\varepsilon_i = \widehat{v}_i - v_i$$

#### **III.4.** Résultats de simulation :

Les paramètres utilisés lors de l'implémentation des deux algorithmes sont:

- Pour la méthode de Zhang, nous avons utilisé 3 niveaux de décompositions avec l'onde mère Db8 et λ=0.01, le coefficient d'apprentissage est fixé à α=10e-5
- Pour le nouveau TNN, nous avons utilisé 4 niveaux de décompositions avec l'onde mère Db8 et α=10e-6 avec <sup>Δthr(i)</sup>/<sub>thr(i)</sub> < 1e - 6.
   </li>
- Dans les deux méthodes, nous allons travailler en mode supervisé. L'image de référence c'est une image bruitée de la même variance que l'image en cours de débruitage avec un bruit non corrélé.

### III.4.1. Résultats de débruitage :

Le tableau III.1 montre une comparaison entre les résultats obtenus (PSNR) par les méthodes classiques et les méthodes basées sur les réseaux de neurones (TNN).

Image	σ	Bruitée	Wiener2	BayseShrink	Méthode de	New TNN
					Zhang [6]	[7]
	10	28.1538	33.5508	33.4404	33.6192	33.7452
	15	24.6113	31.1125	31.6148	31.6885	31.8606
Lena	20	22.0965	28.9920	30.3188	30.4010	30.5670
(512x512)	25	20.1931	27.2295	29.3731	29.3783	29.5121
	30	18.5878	25.7290	28.6018	28.5555	28.7208
	10	28.1291	29.7587	31.5230	31.6769	32.0576
	15	24.6076	28.1547	29.3361	29.3314	29.6381
Barbara	20	22.1232	26.6780	27.7740	27.7191	27.9456
(512x512)	25	20.1652	25.3770	26.5559	26.4981	26.6340
	30	18.5792	24.2290	25.6210	25.6206	25.7363
	10	28.1278	32.1973	32.4387	32.4754	32.6597
	15	24.6364	30.2249	30.2710	30.2947	30.4663
Boat	20	22.1247	28.4101	28.8722	28.8994	28.9925
(512x512)	25	20.1715	26.7963	27.8366	27.8711	27.9429
	30	18.5729	25.3963	27.0236	26.9940	27.1660
	10	28.1449	30.5864	31.1318	31.2715	31.4361
	15	24.6078	29.0860	28.6142	28.8458	29.0092
Cameraman	20	22.1009	27.5811	27.0871	27.1598	27.3252
(256x256)	25	20.1036	26.2126	25.8961	25.9618	26.0821
	30	18.5796	24.9617	25.0992	25.0788	25.2519

Tableau III.1: Résultats du PSNR pour différentes images débruitées par plusieurs techniques.

#### III.5. Courbes d'apprentissage :

Les figures III.6 et III.7 représentent les courbes d'apprentissage de chaque sous bande pour  $\sigma$ =10 par les méthodes proposées en [7] et [6] sur l'image Barbara:



Figure III.6: Courbes d'apprentissage pour chaque sous bande de l'image Barbara pour une variance  $\sigma$ =10 par la méthode de Zhang [6] (apprentissage par descente de gradient stochastique).

Les seuils optimaux obtenus pour l'image Barbara ( $\sigma$ =10) sont présentés par le tableau III.2:

Horizontal	Vertical	Diagonal	Niveau de	
4.6535	4.6535	5.6245	3	
8.6032	5.9453	8.5471	2	
17.5031	7.5648	17.0401	1	

**Tableau III.2:** Seuils optimaux pour Barbara ( $\sigma$ =10) par la méthode Zhang [6].



**Figure III.7:** Courbes d'apprentissage pour chaque sous bande de l'image Barbara pour une variance  $\sigma$ =10 par la méthode [7] (apprentissage par descente de gradient).

Les seuils optimaux obtenus pour l'image Barbara ( $\sigma$ =10) sont données par le tableau III.3:

Horizontal	Vertical	Diagonal	Niveau de
			décomposition
7.8249	12.8598	9.6703	4
13.9949	12.9709	18.0154	3
21.3119	17.1819	20.8575	2
37.5836	20.5807	35.4213	1

**Tableau III.3:** Seuils optimaux pour Barbara ( $\sigma$ =10) par la méthode [7].

#### III.6. Echantillons d'images débruitées:

Les résultats de débruitage des images Lena et Cameraman par les différentes méthodes citées dans les sections précédentes sont représentés par les figure III.8 et III.9.



Image originale

Image bruitée



Wiener2

BayseShrink



Zhang [6]





**Figure III.8**: Images débruitées par différentes techniques pour l'image Lena avec une variance  $\sigma$ =20.



Image Originale



Image bruitée



Wiener2





New TNN [7]

**Figure III.9**: Images débruitées par différentes techniques pour l'image Cameraman avec une variance  $\sigma$ =20.

#### III.7. Représentation des résultats

La figure III.10 représente graphiquement les résultats obtenus (valeurs du PSNR) par

les différentes méthodes mentionnées précédemment sur les images de références.



Figure III.10: Représentation du PSNR des images débruitées en fonction de la variance de bruit, par différentes techniques de débruitage sur les images test de références.

#### **III.8.** Discussion

Les figures (III.6 et III.7), montrent que l'apprentissage par l'algorithme de descente de gradient dans [7] et la descente de gradient stochastique dans [6] a permet la minimisation de l'erreur quadratique moyenne (MSE), c'est à dire d'obtenir les seuils (thresholds) optimaux (tableau III.2 et III.3) pour chaque sous bande. Nous avons montré que l'utilisation d'une autre version d'image bruitée avec un bruit non corrélé permet l'obtention des seuils optimaux.

Les tableaux (III.2 et III.3) montrent les seuils optimaux obtenus par l'algorithme d'apprentissage, chaque seuil est utilisé dans la fonction de seuillage pour l'obtention des coefficients en ondelettes des images débruitées, la transformée inverse permette la reconstitution de l'image débruitée.

La figure (III.10) montre la mise en évidence de l'efficacité des deux méthodes basées sur les TNN pour la réduction du bruit blanc gaussien. Il est remarquable que la sélection du seuil par le TNN a permet l'obtention des résultats optimaux de seuillage par la fonction d'activation (seuillage) utilisée.

La comparaison des résultats du PSNR sur la figure (III.10), montrent la supériorité des deux techniques basées sur le TNN et les techniques usuelles (BayseShrink et normalShrink) sur les techniques classiques (Filtre de Wiener).

Comme nous l'avons mentionné précédemment, pour discuter l'efficacité d'une technique de débruitage, il faux utiliser les critères subjectifs et objectifs. Pour cela, les figures (III.8 et III.9) représentent les résultats de débruitage de quelques images par les différentes techniques, il est clair dans ces conditions que les méthodes basées sur les TNN en l'occurrence proposées en [7] et [6] donnent un meilleur aspect visuel.

#### **III.9.** Conclusion

Dans ce chapitre, nous avons présenté la structure générale du TNN ainsi que deux techniques de débruitage basées sur ce dernier (TNN).

Les résultats de simulation ont été présentés et une étude comparative a été faite.

Les résultats de simulation ont montré l'efficacité des TNN pour le débruitage des images fixes.

Dans le chapitre suivant, nous allons proposer une nouvelle méthode basée sur les algorithmes génétiques (optimisation des seuils) pour le débruitage des images fixes.

### Chapitre VI

# Méthode Proposée pour le Débruitage des Images Fixes

- 1. Introduction
- 2. Structure générale de la technique proposée
- 3. Configuration de la technique
- 4. Résultats de débruitage
- 5. Courbes d'optimisation
- 6. Echantillons d'image débruitée
- 7. Représentation des résultats
- 8. Discussions sur la méthode proposée
- 9. Etat récapitulatif des résultats
- 10. Echantillons des images débruitées
- 11. Discussion des résultats
- 12. Conclusion

#### **IV.1. Introduction**

Depuis toujours, l'homme tente d'améliorer sa vie au quotidien ; chaque jour, il résout sans parfois s'en rendre compte des problèmes d'optimisation : il planifie ses occupations, s'efforce de trouver les prix les plus bas en cas d'achat, cherche à minimiser le temps de conduite et la consommation en carburant lors d'un long trajet en voiture, etc..., enfin, il attache souvent de l'importance au confort avant tout. Ce dernier point montre là, toute la problématique de l'optimisation car cet objectif est une notion très subjective et floue.

Dans le Chapitre précédent, nous avons présentés et discutés l'efficacité des méthodes de l'intelligence artificielle en l'occurrence les TNN, pour le débruitage des images fixes corrompues par un bruit blanc gaussien.

Dans le présent chapitre, nous allons proposer une nouvelle méthode de débruitage basée sur les techniques d'optimisations (optimisation des seuils) en l'occurrence les algorithmes génétiques [8]. Au début du chapitre nous donnerons la structure générale de notre méthode. En suite nous présenterons quelques résultats de simulation, et enfin nous terminerons par la discussion des résultats obtenus.

#### IV.2. Structure générale de la technique proposée

La figure suivante (figure IV.1) illustre la structure générale de la méthode proposée [8]:



Figure IV.1 : Structure générale de la méthode proposée.

La méthode proposée est basée sur le seuillage dans le domaine de la transformée en ondelettes, dont le choix du seuil est une étape primordiale, pour cela nous allons proposés l'utilisation des algorithmes génétiques pour l'optimisation des seuils. L'estimation des seuils optimaux Thr<sub>PN</sub> se fait par l'algorithme génétique en minimisant la fonction coût représentée par l'erreur  $J_{mmse}$  donnée par :

$$MSE_{p} = \frac{1}{2N} \sum_{i=1}^{N} (\widehat{V}_{i} - V_{i})^{2} , \qquad (4.1)$$

avec p = 1, ..., P.

$$J_{mmse} = \frac{1}{P} \sum_{j=1}^{P} MSE_j , \qquad (4.2)$$

ou :

*p* : Sous bands.

**P** : Nombre maximal des sous bandes.

*N* : Longueur de la sous bande.

 $MSE_{p}$ : Erreur quadratique moyenne pour la sous bande p

 $J_{mmse}$ : La moyenne des erreurs quadratiques moyenne des sous bandes

 $\hat{V}_i$ : Les coefficients après opération de seuillage.

 $V_i$ : Les coefficients de référence.

De l'équation (4.1), le recours au besoin d'une autre image de référence pour le calcul des erreurs quadratiques moyennes des sous bandes est indispensable, cela est identiques au TNN proposé par Zhang dans [5] et [6]. Zhang a prouvé que l'utilisation d'une deuxième image bruitée comme référence pour le calcul du MSE conduit au même seuil optimal qu'avec l'image originale.

Pour cela une solution pratique est adoptée, l'utilisation de deux images y et y' obtenue de la même image x avec n et n' deux bruits non corrélés. L'image y' est utilisée comme image de référence.

La solution adoptée est raisonnable du faite que dans plusieurs applications on peut avoir plusieurs versions du signal bruité.

Après l'obtention des seuils optimaux par les algorithmes génétiques pour chaque sous bandes, nous proposons l'utilisation de l'algorithme Cycle spinning (voire Figure IV.1) proposée par Coifman et Donoho [23] pour remédier au problème de l'effet de Gibbs et aussi pour améliorer la qualité de l'image débruitée. Le principe de cet algorithme est (voir figure IV.2):

Décaler l'image, décomposer la version décalée de l'image par transformée en ondelettes, appliquer l'opération de seuillage, enfin reconstruire puis redécaler l'image.

Faire ces étapes pour chaque décalages, et faire la moyenne de plusieurs résultats obtenus, ceci permet de réduire d'une manière efficace l'effet de Gibbs généré dans

l'opération de seuillage et plus faiblement affecté par le phénomène de Gibbs que celle obtenue par le seuillage utilisant la transformée en ondelettes orthogonales traditionnelles.



Figure IV.2 : Principe de l'algorithme « Cycle Spinning »

Où :

- y : Image bruitée.
- $y^k$  : Image bruitée décalée.
- W: transformée en ondelettes.
- $W^{-1}$ : transformée en ondelettes inverses.
- *T* : opération de seuillage.
- $\hat{y}^k$ : Image débruitée décalée.
- $\hat{y}$ : Image débruitée.

#### Fonction de seuillage

La fonction de seuillage utilisée par la méthode proposée est la fonction d'activation proposée dans [7], est donnée par :

$$\eta(x, thr) = \begin{cases} x - 0.5 \frac{thr^2}{x} & |x| > thr \\ 0.5 \frac{x^3}{thr^2} & |x| \le thr \end{cases}$$
(4.3)

Nous avons choisi l'utilisation de cette fonction de seuillage, à cause de ses caractéristiques remarquables (voir figure IV.3). Cette fonction prend les caractéristiques de la fonction de seuillage doux (meilleur aspect visuel) et aussi de la fonction de seuillage dur (meilleurs résultats du PSNR).



Figure IV.3 : Fonction de seuillage proposée dans [7].

#### IV.3. Configuration de la méthode

L'algorithme génétique utilisé est à codage réel, donc le bloc d'algorithme génétique est constitué d'un seul chromosome, ce dernier est composé de 12 gènes, chaque gène représente le seuil d'une sous bande (pour plus de détails voir annexe 2).

Pour la simulation de notre méthode, nous avons utilisé le Toolbox « Optimisation » de Matlab version 7.7.

Pour cela nous allons conserver la configuration par défaut avec les changements suivants :

La fonction d'adaptation c'est l'équation (4.2).

Nombre de variables: 12.

Dans la population initiale, tous les gènes = 10.

La fonction de tolérance: 1e-9.

### IV.4. Résultats de débruitage

Le tableau IV.1 donne une comparaison des résultats en PSNR obtenus par la méthode proposée [8] et les meilleures méthodes discutées dans les parties antérieures de ce mémoire.

Image	σ	Bruitée	Wiener2	BayseShrink	Méthode	New TNN	Méthode
					de Zhang	[7]	proposée
					[6]		[8]
	10	28.1538	33.5508	33.4404	33.6192	33.7452	34.5502
	15	24.6113	31.1125	31.6148	31.6885	31.8606	32.4853
Lena	20	22.0965	28.9920	30.3188	30.4010	30.5670	31.0686
(512x512)	25	20.1931	27.2295	29.3731	29.3783	29.5121	30.0327
	30	18.5878	25.7290	28.6018	28.5555	28.7208	29.1635
	10	28.1291	29.7587	31.5230	31.6769	32.0576	33.0529
	15	24.6076	28.1547	29.3361	29.3314	29.6381	30.4931
Barbara	20	22.1232	26.6780	27.7740	27.7191	27.9456	28.7403
(512x512)	25	20.1652	25.3770	26.5559	26.4981	26.6340	27.4033
	30	18.5792	24.2290	25.6210	25.6206	25.7363	26.3441
	10	28.1278	32.1973	32.4387	32.4754	32.6597	33.5443
	15	24.6364	30.2249	30.2710	30.2947	30.4663	31.2868
Boat	20	22.1247	28.4101	28.8722	28.8994	28.9925	29.8028
(512x512)	25	20.1715	26.7963	27.8366	27.8711	27.9429	28.5920
	30	18.5729	25.3963	27.0236	26.9940	27.1660	27.7792
	10	28.1449	30.5864	31.1318	31.2715	31.4361	32.3003
	15	24.6078	29.0860	28.6142	28.8458	29.0092	29.7855
Cameraman	20	22.1009	27.5811	27.0871	27.1598	27.3252	28.0848
(256x256)	25	20.1036	26.2126	25.8961	25.9618	26.0821	26.8893
	30	18.5796	24.9617	25.0992	25.0788	25.2519	25.8999

Tableau IV.1 : Résultats du PSNR de différentes images débruitées par plusieurs techniques.

#### **IV.5.** Courbes d'optimisation

Les figures IV.4 et IV.5 illustrent les courbes d'optimisation des seuils par les algorithmes génétiques.

La courbe bleue dans les deux figures (IV.4 et IV.5) représente les valeurs moyennes de la fonction coût pour chaque génération (Mean fitness), et la courbe noire illustre les valeurs minimales (meilleurs valeurs obtenues pour chaque génération) de la fonction coût (Best fitness).

A partir des figures IV.4 et IV.5, il est clair que l'algorithme génétique a minimisé la fonction coût proposée, ce qui a permis l'optimisation des seuils (Voir Tableaux IV.2 et IV.3).



Best: 172.2376 Mean: 172.2507



Les seuils optimaux trouvés pour l'image Lena avec la variance de bruit  $\sigma$ =10 sont donnés dans le tableau IV.2.

Horizontal	Vertical	Diagonal	Niveau de décomposition
7.1619	13.0998	12.3783	4
18.8501	16.8553	19.5399	3
24.6894	21.5789	25.2529	2
39.4552	31.0881	42.5950	1

**Tableau IV.2:** Seuils optimaux trouvés sur l'image Lena ( $\sigma$ =10) avec la méthode proposée.



Figure IV.5 : Courbe d'optimisation par AG sur l'image Barbara avec une variance de 10.

Horizontal	Vertical	Diagonal	Niveau de				
			décomposition				
8.6819	9.2747	8.9882	4				
14.9378	13.3498	16.9121	3				
22.8708	17.3457	20.5794	2				
35.5338	20.2221	35.8288	1				

Les seuils optimaux trouvés pour l'image Barbara avec la variance de bruit  $\sigma$ =10 sont présentés dans le tableau IV.3.

**Tableau IV.3:** Seuils optimaux trouvés sur l'image Barbara ( $\sigma$ =10) avec la méthode

proposée.

#### IV.6. Echantillons d'image débruitée

Pour une comparaison subjective, les figures IV.6, IV.7 et IV.8 montrent les résultats de débruitage sur les trois images de test Lena, Boat et Cameraman obtenus par les différentes méthodes étudiées.



Image originale



Image bruitée



Wiener2





BayseShrink



New TNN [7]



Méthode proposée [8]

**Figure IV.6**: Images Lena débruitées par différentes techniques avec une variance  $\sigma=20$ .



Image originale



Image bruitée



Wiener2



BayseShrink



New TNN [7]



Méthode proposée [8]

**Figure IV.7:** Images débruitées par différentes techniques pour l'image Boat avec une variance  $\sigma=30$ .


Image originale



Image bruitée



Wiener2



BayseShrink



New TNN [7]



Méthode proposée [8]

Figure IV.8: Images débruitées par différentes techniques pour l'image cameraman avec une variance  $\sigma$ =30.

# IV.7. Représentation des résultats

La figure IV.9 montre une comparaison graphique entre la méthode proposée est les meilleures méthodes rappelées dans les sections précédentes de ce mémoire.



Figure IV.9: Représentation du PSNR des images débruitées en fonction de la variance de bruit par différentes techniques de débruitage

# IV.8. Discussions des résultats obtenus par la méthode proposée

Le tableau IV.1, montre clairement que la méthode proposée [8] est performante pour l'ensemble des images test du point de vue de la réduction du bruit, des valeurs élevées du PSNR (voir figure IV.9), et aussi de la non génération d'artefacts sur les images reconstituées (voir figures (IV.6, IV.7 et IV.8)). Cela peut être interprété par l'utilisation des seuils optimaux et indépendants pour chaque sous bande, et encore par l'utilisation de l'algorithme Cycle Spinning [23] pour la réduction des artefacts générés par l'effet de Gibbs.

A partir des résultats obtenus dans les chapitres précédents, nous pouvons conclure que le recours aux algorithmes génétiques pour optimiser les seuils de chaque sous bandes est très bénéfiques pour le débruitage des images fixes.

# IV.9. Etat récapitulatif des résultats

Le tableau IV.4 montre un état récapitulatif des résultats de simulations en PSNR obtenus à travers ce travail. La méthode proposée est comparée aux dix méthodes les plus connues dans le domaine de débruitage des images fixes.

Image	σ	Bruitée	Filtre	Filtre	Filtre	Wiener2	VisuSh	VisuSh	LevelSh	LevelSh	NormalSh	BayseShri	Zhang [6]	New TNN	Proposée
			moy <sup>*</sup>	médian <sup>*</sup>	gauss**		Soft	Hard	soft	hard	rink	nk		[7]	[8]
	10	28.1538	30.9183	32.1060	32.5122	33.5508	28.0664	30.4323	30.2706	32.0832	33.5473	33.4404	33.6192	33.7452	34.5502
Lena (512x512)	15	24.6113	29.8748	30.1022	30.6802	31.1125	26.6738	28.6763	28.9771	30.4730	31.6357	31.6148	31.6885	31.8606	32.4853
	20	22.0965	28.7766	28.7622	29.3960	28.9920	25.7490	27.4602	28.1072	29.2816	30.2804	30.3188	30.4010	30.5670	31.0686
	25	20.1931	27.6761	27.9377	28.4684	27.2295	25.0418	26.5795	27.3521	28.3786	29.1985	29.3731	29.3783	29.5121	30.0327
	30	18.5878	26.8796	27.1408	27.7696	25.7290	24.6298	25.9068	26.8220	27.6339	28.3901	28.6018	28.5555	28.7208	29.1635
	10	28.1291	24.7861	24.7615	29.7872	29.7587	24.9862	27.6043	26.1164	28.6369	31.8453	31.5230	31.6769	32.0576	33.0529
	15	24.6076	24.5226	24.2531	27.3903	28.1547	23.5463	25.3221	24.6523	26.2557	29.3953	29.3361	29.3314	29.6381	30.4931
Barbara (512x512)	20	22.1232	24.1632	23.7475	26.0503	26.6780	22.8624	24.0434	23.9361	24.9022	27.6362	27.7740	27.7191	27.9456	28.7403
	25	20.1652	23.7573	23.1781	25.0240	25.3770	22.4062	23.3113	23.4854	24.0861	26.3962	26.5559	26.4981	26.6340	27.4033
	30	18.5792	23.2811	22.5685	24.2420	24.2290	22.1640	22.8309	23.2321	23.6433	25.4184	25.6210	25.6206	25.7363	26.3441
Boat (512x512)	10	28.1278	28.9518	30.1650	31.6378	32.1973	26.3593	28.6612	28.2589	30.1073	32.3771	32.4387	32.4754	32.6597	33.5443
	15	24.6364	28.2768	28.7217	29.6225	30.2249	25.0501	26.9204	26.9657	28.4050	30.2594	30.2710	30.2947	30.4663	31.2868
	20	22.1247	27.4787	27.3328	28.2908	28.4101	24.1546	25.8080	26.1533	27.2973	28.7873	28.8722	28.8994	28.9925	29.8028
	25	20.1715	26.6096	26.0933	27.3025	26.7963	23.5088	24.9659	25.5209	26.4911	27.7174	27.8366	27.8711	27.9429	28.5920
	30	18.5729	25.8082	25.4446	26.5617	25.3963	23.0671	24.3340	25.0079	25.8712	26.9061	27.0236	26.9940	27.1660	27.7792
Cameraman (256x256)	10	28.1449	24.9180	26.5184	29.8628	30.5864	24.6287	27.3462	26.0796	28.3064	31.1806	31.1318	31.2715	31.4361	32.3003
	15	24.6078	24.6449	25.7859	27.4725	29.0860	23.2068	25.4601	24.8398	26.5568	28.7392	28.6142	28.8458	29.0092	29.7855
	20	22.1009	24.3129	25.0146	26.1206	27.5811	22.3331	24.0969	23.9338	25.3723	27.1981	27.0871	27.1598	27.3252	28.0848
	25	20.1036	23.8527	24.2417	25.0120	26.2126	21.6377	23.2452	23.2879	24.6793	26.0078	25.8961	25.9618	26.0821	26.8893
	30	18.5796	23.3844	23.3960	24.2368	24.9617	21.0822	22.5278	22.8476	23.9186	25.1655	25.0992	25.0788	25.2519	25.8999

\* résultats obtenus par la meilleure taille de masque des filtres.

\* \* résultats obtenus par les meilleures valeurs de variance et taille du masque du filtre gaussien.

Tableau IV.4 : Résultats de débruitage (PSNR [dB]) sur différentes images de tests par toutes les méthodes de débruitage étudiées.

# IV.10. Echantillons des images débruitées

Les figures IV.10 et IV.11 montrent les résultats obtenus par toutes les méthodes discutées dans ce mémoire.





**Figure IV.10:** Résultats de débruitage de l'image Lena par toutes les techniques étudiées (σ=20).

De gauche à droit : Image originale, image bruitée, filtre moyenneur, filtre médian, filtre Gaussien, filtre de wiener, VisuShrink (soft), VisuShrink (hard), LevelShrink (soft), LevelShrinl (hard), NormalShrink, BayseShrink, zhang, new TNN [7], méthode proposée [8].





**Figure IV.11:** Images débruitées par toutes les méthodes discutées sur l'image Boat avec une variance ( $\sigma$ =30).

De gauche à droite : Image originale, image bruité, Filtre moyenneur, Filtre médian, Filtre gaussien, Filtre de Wiener2, VisuShrink (soft), VisuShrink (hard), LevelShrink (soft), LevelShrink (hard), NormalShrink, BayseShrink, New TNN [7], Méthode proposée [8].

#### IV.11. Discussion des résultats

Afin de mener une étude comparative complète, nous avons utilisé les méthodes classiques de débruitage donnée par les filtres. L'examen des tableaux présentés dans ce chapitre montre l'efficacité des méthodes basées sur la T.O.D par comparaison avec la méthode classique pour toutes les tailles d'images et aussi lorsque la variance du bruit est élevée.

Le tableau IV.4 et les figure IV.10 et IV.11 montrent que le filtre de Wiener donne les meilleures performances objectif et subjectif que les autres filtres étudiés, cela peut être interprété par l'utilisation des méthodes adaptatives lors du calcule du filtre de Wiener.

Les colonnes 6 et 7 du tableau (IV.4) représentent la technique classique du seuillage en ondelettes (VisuShrink). Le seuillage consiste à annuler l'ensemble des coefficients en ondelettes en dessous d'un seuil. Comme nous l'avons mentionné auparavant. Ce procédé existe sous deux formes: le seuillage dur consiste à laisser inchangé les coefficients en ondelettes au dessous du seuil, par contre le seuillage doux les modifie en diminuant leur amplitudes. Les images obtenues après reconstitution contiennent généralement une information non homogène et de hauts niveaux du bruit. La reconstruction en ondelettes génère des artefacts sur des telles images (voire figure IV.10 et IV.11).

D'après la lecture du tableau IV.4, il ressort que les résultats obtenus par la méthode « LevelShrink » sont meilleurs à ceux obtenus par les méthodes de seuillage classiques en terme du PSNR et qualité visuelle.

Ces performances peuvent être interprétées par le fait que la méthode « LevelShrink » utilise un seuil adapté pour chaque niveau de décomposition à la différence du seuillage classique qui emploie un seuil commun (universel) pour tous les niveaux.

En ce qui concerne les méthodes, BayseShrink et NormalShrink, il est évident à partir des résultats représentés dans le tableau (IV.4) qui sont plus performants en terme du gain en PSNR, et de qualité visuelle, comparativement aux méthodes déjà interprétées « VisuShrink, Level Shrink », cela peut être expliqué par le concept d'adaptation du seuil utilisé pour chaque niveau et chaque sous bande de décomposition.

Le tableau IV.4, montre aussi que les deux techniques basées sur les TNN (Thresholding Neural Network) en l'occurrence, Zhang [6] et new TNN [7], donnent les meilleures performances (meilleure valeur du PSNR et qualité visuelle (voir figures IV.10 et IV.11)).

Ceci peut être interprété par l'utilisation des seuils optimaux obtenus pour chaque sous bande ainsi que la fonction de seuillage (activation) utilisée.

Nous notons aussi, que les deux techniques basées sur les réseaux de neurones ne dépendent pas des caractéristiques statistiques des images et du bruit. Par contre dans les méthodes déjà discutées (VisuShrink, LevelShrink, BayseShrink et NormaleShrink), le calcul des seuils dépend principalement des caractéristiques statistiques de l'image ainsi que du bruit gaussien.

Sur l'ensemble des images de test, la meilleure amélioration en rapport signal sur bruit en pic a été obtenue par la méthode proposée [8] par comparaison aux méthodes décrites ci-dessus.

Il est bien clair, que la méthode proposée est la plus performante pour l'ensemble des images de test, du point de vue de la réduction du bruit, des valeurs élevées du PSNR,

et aussi de la non génération d'artefacts sur les images reconstituées (voir les images des figures (IV.10, IV.11)). En effet l'étude a pu montrer une supériorité relative de la méthode proposée.

Par ailleurs, nous notons un constat que l'on peut tirer des résultats expérimentaux obtenus qu'ils y a plusieurs facteurs affectant les performances des méthodes de débruitage discutées précédemment, à savoir la dimension, le contenu de l'image et la variance du bruit.

Cependant ces facteurs n'influent pas sur les performances de la méthode proposée. En d'autres termes, l'algorithme proposé est toujours performant quelque soit la taille de l'image, son contenu et la variance du bruit contenu dans l'image.

# **IV.12.Conclusion**

Dans ce chapitre, nous avons proposé une nouvelle méthode basée sur les algorithmes génétiques pour le débruitage des images fixes dans le domaine de la transformée. Dans un premier lieu, nous avons donné la structure générale de la méthode, en suite les résultats de simulations obtenus ont été présentés.

Nous avons terminé cette étude par l'élaboration d'un tableau récapitulatif des résultats obtenus en utilisant toutes les méthodes étudié, ainsi une discussion des résultats a été présentée.

Les résultats obtenus ont montrés l'efficacité de la méthode proposée pour la réduction du bruit gaussien sur des images fixes.

61

# Conclusion Générale

#### **Conclusion Générale**

Définir une méthodologie permettant d'extraire, avec précision maximale, l'information présente dans un signal est primordiale dans de nombreuses applications. Dans ce travail, nous avons étudié le problème du débruitage qui a engendré une importante littérature en prétraitement des images. D'où l'idée de restaurer une image de bonne qualité à partir de sa version dégradée. Nous nous sommes focalisés sur la technique de seuillage dans le domaine de la transformée en ondelettes comme solution au problème.

Le travail présenté dans ce mémoire avait pour but le débruitage des images par les techniques de l'intelligence artificielle, dont dix méthodes ont été implémentées et une nouvelle méthode basée sur les algorithmes génétiques a été proposée.

Dans ce mémoire, nous avons donnée un aperçu générale sur la transformée en ondelettes ainsi que sur les algorithmes génétiques (voir annexes 1 et 2).

L'extension du débruitage, tel qu'il a été proposé par DONOHO, reposant sur le seuillage des coefficients d'ondelettes, représente une référence pour les techniques du débruitage avec seuillage qui viennent après.

Dans le but d'étudier les performances des différentes méthodes sélectionnées, plusieurs expériences ont été menées. L'étude comparative établie est basée sur le calcul du rapport signal à bruit en pic (PSNR) et de l'erreur quadratique moyenne (MSE) ainsi que sur la qualité visuelle des images débruitées.

A travers les résultats obtenus, nous avons constaté d'une manière claire la supériorité entière des techniques se basant sur l'intelligence artificielle (réseaux de neurones et algorithmes génétiques) pour la réduction d'un bruit additif, opérant sur l'ensemble des images de test utilisées.

L'étude comparative a montrée la supériorité de la méthode proposée, où nous avons utilisé les algorithmes génétiques pour l'obtention des seuils optimaux et l'application de l'algorithme cycle spinning pour améliorer la qualité de l'image débruitée, pour cela, nous pouvons conclure que nous avons utilisé avec succès les algorithmes génétiques pour l'optimisation des seuils pour chaque sous bande dans le domaine de débruitage des images fixes.

Nous espérons que ce travail sera d'une grande aide et utilité à ceux qui aborderont les sujets relevant du domaine du débruitage d'images fixes dans l'avenir proche.

Enfin, le plus important, c'est que le débruitage d'images ou généralement le traitement d'images reste un domaine très ouvert qui bénéficie de toute amélioration ou idée proposée, et cela nous laisse penser que le travail que nous avons effectué reste ouvert à d'éventuelles améliorations.

# Annexe 1

# Transformée en Ondelettes

# Transformée en Ondelettes

#### **1.1 Introduction**

L'étymologie du mot ondelettes n'est, en soi, pas très compliquée. Les ondelettes utilisées par Jean Morlet avaient, en effet, une allure de petites ondes. D'où cette appellation. Vers 1975, ce géophysicien, qui travaillait à l'époque pour Elf Aquitaine, a créé des petites fonctions mathématiques ayant certaines propriétés intéressantes ; ceci afin de sonder les différentes couches géologiques et ainsi, détecter des nappes de pétroles. J. Morlet est donc un des artisans majeurs des ondelettes. Cependant, d'autres chercheurs travaillant dans des domaines de recherche fort différents, utilisaient à cette époque des outils fort semblables aux petites ondes de Morlet. Seulement, ces méthodes étaient souvent expérimentales et ne possédaient aucun formalisme rigoureux. Ce formalisme émergera durant les années 80 des travaux de scientifiques tels que ceux d'Yves Meyer, de Stéphane Mallat, d'Ingrid Daubechie et de bien d'autres [24].

Les ondelettes sont, avant tout, un puissant outil d'analyse mathématique. La véritable révolution due aux ondelettes s'est opérée dans le très vaste domaine du traitement du signal.

En règle générale, les ondelettes n'ont rien démontré qui n'ait déjà été démontré rigoureusement auparavant. Donc, c'est par les spécialistes du traitement du signal que les ondelettes ont été accueillies comme une mine d'or.

Cette annexe a pour but de rappeler les notions les plus élémentaires sur la théorie en ondelettes ainsi que ses différentes propriétés

## **1.2.** Les ondelettes

Il est bien connu que la transformée de Fourier donnée comme étant une généralisation de la série de Fourier caractérisant les signaux périodiques a l'inconvénient majeure d'ignorer complètement la contribution temporelle exacte d'une fréquence dans un signal.

En pensant à résoudre ce problème, GABOR en 1946 a introduit la transformée de Fourier a court terme. Cette dernière se base sur le fenêtrage (Windowing). Il s'agit de segmenter en tranche de temps fixes le signal à analyser pour appliquer par la suite la TF à chaque tranche. Cette manière d'analyse a mené à un compromis difficile à satisfaire. Ce critère a mené aux constatations suivantes [25] :

➢ Fenêtre du temps longue → Une bonne résolution fréquentielle contre une résolution temporelle pauvre.

➤ Dans le cas inverse, une fenêtre du temps courte → Une résolution fréquentielle pauvre contre une bonne résolution temporelle.

La suite logique qui devrait résoudre ce problème a été l'élaboration de l'outil puissant dénommé par la transformée en ondelettes (wavelets).

Les domaines d'applications qui ne cessent de s'accroître, depuis la télécommunication à la médecine en passant par l'analyse des divers phénomènes physiques tels que : les signaux sismiques, l'océanographie, la prospection géologique,..., démontrent d'une manière concluante la puissance de cet outil [25].

Dans ce qui suit des définitions mathématiques essentielles seront présentées.

#### 1.3. La transformée en ondelettes continue

Les ondelettes (wavelets) sont issues de ce qu'on appelle onde mère (mother wave). On aboutira à représenter n'importe quel signal, par une base d'ondelettes qui ne sont que des versions dilatées et translatées de l'onde mère. La représentation d'un signal continu x(t) par une base d'ondelettes est donnée par [25] :

$$w_x(a,b) = \int_{-\infty}^{+\infty} x(t) \psi_{a,b}^*(t) dt$$
 (a1.1)

Pour le cas continu les paramètres de translation et de dilatation varient de manière continue comme montrée par la relation suivante :

$$\psi_{a,b} = \frac{1}{\sqrt{|a|}} \psi(\frac{t-b}{a}) \quad \text{avec } a \in R, b \in R, a \neq 0 \quad (a1.2)$$

Òи

*a* : facteur d'échelle pour la dilatation de  $\psi$ .

*b* : le coefficient de translation.

La transformée en ondelettes est réversible si l'ondelette mère vérifie la condition d'admissibilité suivante :

$$\int \left|\psi(\omega)\right|^2 \frac{d\omega}{\omega} < +\infty \tag{a1.3}$$

Donc on peut dire que l'ondelette mère est une fonction moyenne nulle :

$$\int \psi(t)dt = 0 \tag{a1.4}$$

C'est-à-dire que l'ondelette mère oscille de part et d'autre de l'axe t, ce qui lui donne dans une certaine mesure l'apparence d'une onde.

Dans les ondelettes continue, le choix de l'échelle *a* dépend de l'analyse voulue, pour *a* petit on effectue l'analyse du signal très localisé en temps, aussi pour *a* grand on réalise une analyse sur un horizon plus large (qui donne une vue d'ensemble). L'analyse des hautes fréquences est réalisée avec une résolution fréquentielle pauvre mais permet une localisation temporelle très précise. Au contraire les basses fréquences sont analysées avec une très forte résolution fréquentielle, mais leurs instants d'occurrence sont imprécis [24]. La figure 1.1 illustre la représentation temps échelle pour les ondelettes



Figure 1.1 : Représentation Temps échelle pour ondelettes

Mais parmi toutes ces propriétés l'ondelette continue présente un inconvénient major qui est la redondance, donc on est amené à diminuer cette redondance, et cela par l'utilisation de l'ondelette discrète [26].

#### 1.4. La transformée en ondelettes discrète (T.O.D)

La transformée discrète en ondelettes est issue de la version continue, à la différence de cette dernière, la TOD utilise un facteur d'échelle et une translation discrétisés. Donc les bases d'ondelettes sont alors définies par les fonctions :

$$\psi_{j,k}(t) = a_0^{-j/2} \psi(a_0^{-j}t - kb_0)$$
(a1.5)

La transformée en ondelettes et la transformée inverse sont définies par :

$$W_{x}(j,k) = \langle x, \psi_{j,k} \rangle = \int_{-\infty}^{+\infty} x(\alpha) \psi_{j,k}^{*} d\alpha \qquad (a1.6)$$

$$x(t) = \sum_{(j,k)\in Z} W_x(j,k) \psi_{j,k}^*$$
(a1.7)

La transformée en ondelettes retourne un ensemble discret de coefficients Wx(j,k). Y. Meyer a mis en évidence une transformée en ondelettes dyadiques en posant  $a_0 = 2$  et  $b_0 = 1$  pour des fonctions particulières  $\psi \in L^2(R)$ .la famille d'ondelettes

$$\psi_{j,k}(t) = 2^{-j/2} \psi(2^{-j}t - k)$$
 (a1.8)

 $(j,k) \in (Z,Z)$  Constitue une base orthonormée.

- La fonction d'ondelettes \u03c6 , qui permet de relever les hautes fréquences est représentée par un filtre numérique passe haut pour définir les détails.
- La fonction d'échelle \u00f8, qui permet de relever les basses fréquences est représentée par un filtre passe bas pour définir les parties lisses ou douces du signal (approximations).

La figure suivante montre la décomposition du signal en approximation et détail [27] :



Figure 1.2 : La décomposition en approximation et détail

#### 1.5. Algorithme de MALLAT

Le schéma montrant la décomposition des signaux bidimensionnels tels que l'image est présentée par la figure 1.3 Cette architecture est adaptée à l'analyse des signaux bidimensionnels et spécialement à l'image. Après une décomposition d'une image (par exemple) de taille NxM et après la décimation, quatre sous images (sub-images) résultent dénotant CA (les coefficients d'approximations), CH (les coefficients de détails horizontaux), CV (les coefficients de détails verticaux) et CD (les coefficients de détails diagonaux). La reconstruction est l'opération similaire mais inversement [25].



**Figure 1.3 :** Algorithme de MALLAT bi-dimentionnel *(a)* Décomposition, *(b)* Reconstruction (T.O.D.I)

Récemment, Les images médicales volumétriques ainsi que la vidéo ont fait l'objet d'une analyse par ondelettes selon l'algorithme de MALLAT extensible au cas du 3D. Dans le cas des images volumétriques un troisième axe a été considéré, tandis que pour la vidéo ce n'est que le temps [25].

# 1.6. Architecture Pyramidale

L'architecture pyramidale (voir figure 1.4) se base sur la décomposition du signal en deux signaux : approximation et détails qui se composent eux même de trois types (horizontal, vertical, diagonal). Le signal approximation est à son tour décomposé et ainsi de suite. La figure suivante illustre cette stratégie [25].



Figure 1.4 : Décomposition pyramidale de profondeur 3 d'un signal bidimensionnel

Ce schéma illustre la décomposition d'une profondeur (depth) de 3 niveaux. x(n) est le signal original de taille NxM. CA3 est le signal basses fréquences de taille N/4xM/4 ainsi que CH3, CV3 et CD3 sont les coefficients de détails (hautes fréquences) au niveau 3. CH2, CV2 et CD2 représentent les coefficients de détails du deuxième niveau de décomposition ainsi que CH1, CV1 et CD1 sont les coefficients de détails du premier niveau de décomposition [25].

# 1.7. Schéma de décomposition des images en ondelettes

La décomposition d'une image par la transformée en ondelettes est illustrée par la figure suivante [27] :



Figure 1.5 : La décomposition multi niveaux par ondelettes

# 1.8. Exemple de décomposition

La décomposition de l'image « Lena » en utilisant l'ondelette mère « Db8 » avec 4 niveaux de décomposition :



Figure 1.6 : Exemple de décomposition par ondelettes en 4 niveaux.

# **1.9. Exemples d'ondelettes**

Il y a une variété de différentes ondelettes pour convenir les besoins de différentes applications [28].

Nous présentons sur la figure 1.8 quelques types d'ondelettes utilisées dans la suite du travail telles que les ondelettes de Daubechies de différents ordres 1, 4, 8 et 12, celle de Coiflet d'ordres 2 et 4. L'axe horizontal représente le nombre d'échantillons, l'axe vertical représente l'amplitude.



Figure 1.7 : Exemples d'ondelettes

# 1.10. Conclusion

Dans cette annexe nous avons présenté un aperçu sur les ondelettes dont la théorie s'est développée dans les années 90. En effet, celle-ci repose sur l'analyse multi-résolution, qui prévoit que l'on peut analyser un signal en le décomposant en un signal d'approximation et un signal de détail, remédiant ainsi aux problèmes rencontrés lors de l'utilisation de la transformée de Fourier classique et de la transformée de Fourier à fenêtre.

Par ailleurs, les ondelettes ont eu un grand succès dans les domaines de traitement de signal et d'images et sont devenues un outil privilégié pour plusieurs applications tels que la compression, le tatouage, la segmentation,...etc. On a alors envisagé à l'exploiter dans le domaine de débruitage d'images fixes monochromes.

# Annexe 2

# Algorithmes Génétiques

# **Algorithmes Génétiques**

# **2.1. Introduction**

Les algorithmes génétiques sont des techniques d'optimisation stochastiques qui tentent d'imiter les processus d'évolution naturels des espèces et de la génétique. Ils agissent sur une population d'individus soumis à une sélection darwinienne : les individus, dits parents, les mieux adaptés à leur environnement survivent et peuvent se reproduisent. Ils sont alors soumis à des mécanismes de recombinaison analogues à ceux de la génétique. Des échanges de gènes entre parents résultent la création de nouveaux individus, dits enfants, qui permettent de tester d'autres configurations de l'espace de recherche [29].

Les algorithmes génétiques diffèrent fondamentalement des autres méthodes dans la recherche de l'optimum [32] :

1. Ils utilisent un codage des paramètres de la fonction à optimiser et non les paramètres euxmêmes.

2. Ils agissent sur un ensemble de configurations (population) et non sur un point unique.

3. Ils n'utilisent que les valeurs de la fonction à optimiser, pas sa dérivée ou une autre connaissance auxiliaire.

4. Ils utilisent des règles de transitions probabilistes (non déterministes) [30].

# 2.2. Formulation du problème d'optimisation

En raison de l'analogie avec la théorie de l'évolution, l'algorithme génétique est naturellement formulé en terme de maximisation [31] :

$$\max_{\mathbf{X}\in\mathbf{I}}\mathbf{F}_{obj}(\mathbf{X}) \tag{a2.1}$$

où I désigne l'espace de recherche et Fobj la fonction objectif à maximiser.

Par ailleurs, la fonction  $F_{obj}$  doit avoir des valeurs positives sur l'ensemble du domaine I (cette condition est imposée par l'utilisation de certains opérateurs génétiques que nous décrirons par la suite). Dans le cas contraire, il convient d'ajouter aux valeurs de  $F_{ojb}$  une constante positive  $F_{min}$  suffisamment grande conformément à l'équivalence de (1) et (2) :

$$\max_{\mathbf{X}\in\mathbf{I}}\mathbf{F}_{obj}(\mathbf{X}) + \mathbf{F}_{\min} \tag{a2.2}$$

Dans de nombreux problèmes, l'objectif est exprimé sous forme de minimisation d'une fonction  $f_{obj}$ :

$$\min_{\mathbf{X} \in \mathbf{I}} \mathbf{f}_{obj}(\mathbf{X}) \tag{a2.3}$$

Le passage d'un problème de minimisation à un problème de maximisation est obtenu par transformation de la fonction  $f_{obj}$ . Il est aisé de montrer par exemple que (3) est équivalent à(4) :

$$\mathbf{F}_{obj}(\mathbf{X}) = \mathbf{F}_{max} - \mathbf{f}_{obj}(\mathbf{X}) \tag{a2.4}$$

Où  $F_{max} \ge \max_{X \in I} f_{obj}(X)$  pour éviter que la fonction  $F_{obj}(x)$  à maximiser prenne des valeurs négatives sur son domaine de définition [29].

Notons que le choix de la fonction F<sub>obj</sub> n'est pas unique. On rencontre également dans la littérature la fonction de transformation suivante [32][33] :

$$F_{obj}(X) = \frac{1}{1 + f_{obj}(X)}$$
 (a2.5)

# 2.3. Terminologie, définitions et notations employées

# 2.3.1. Terminologie et définitions

L'environnement ou espace de recherche définit l'ensemble des configurations possibles des paramètres de la fonction à optimiser [29].

Un individu représente un point de cet espace et une configuration possible des paramètres. La population désigne un groupe d'individus.

On qualifie souvent la fonction à optimiser de fonction d'adaptation. Elle mesure la performance (capacité d'adaptation à l'environnement par similarité avec le processus d'évolution naturel) des individus dans leur environnement [31].

Chaque individu est caractérisé par son chromosome. C'est une chaîne de gènes qui contient les caractéristiques génétiques d'un individu et code les paramètres de sa fonction d'adaptation. Le gène (ou allèle) est la partie élémentaire du chromosome. Il représente un trait de caractère ou une fonction particulière. Si le chromosome est codé en binaire, un gène sera symbolisé par un bit auquel seront associées plusieurs de ces caractéristiques (valeur, position ou poids dans la chaîne,...).

Le génotype d'un individu caractérise la structure du chromosome tandis que le phénotype désigne les valeurs réelles des paramètres réels et de la fonction objectif qui lui sont associées [29].

Les définitions ci-dessus sont illustrées aux figures Fig.2.1 et Fig.2.2.

# 2.3.2. Notations

Dans la suite,

$$\mathbf{P}^{t} = (\mathbf{a}_{1}^{t}, \mathbf{a}_{2}^{t}, \dots, \mathbf{a}_{N}^{t}) \in \mathbf{I}^{N}$$
 (a2.6)

Désignera la population à la génération t, avec N le nombre total d'individus (soit la taille de la population),  $\mathbf{a}_{\mathbf{i}}^{t}$  le i<sub>ème</sub> individu et I l'espace de recherche [31].



Figure 2.1: Définition de l'environnement, de l'individu et la population.



Figure 2.2: un individu caractérisé par un chromosome codé en binaire.

La fonction f:  $I \to \Re$  donne la performance des individus. Nous supposerons dans ce qui suit que les individus sont classés par ordre décroissant de leur performance,  $a_1^t$  étant le meilleur

individu de la population Pt. On définit alors l'application rang :  $I \rightarrow \{1, 2, ..., N\}$  de la manière suivante [29]:

$$\forall i \in \{1,2,\ldots,N\}: \quad rang(a_i^t) = i \qquad (a2.7)$$

# 2.4. Structure de l'Algorithme Génétique

L'organigramme fonctionnel de la figure Fig.2.3 illustre la structure de l'algorithme génétique standard. Nous détaillerons les diverses phases qui le constituent et présenterons les mécanismes associés à chacune d'entre elles dans les sections suivantes [31].



Figure 2.3 : Structure de l'algorithme génétique.

#### 2.5. Initialisation de la population

Les chromosomes des individus de la première génération sont habituellement initialisés de façon uniformément aléatoire, en respectant l'équiprobabilité d'obtenir une valeur égale à 0 ou 1 pour chaque bit.

# 2.6. Codage et décodage des paramètres

Le codage est la présentation des paramètres sous forme des codes. Le décodage est la conversion des codes de paramètres en valeur naturelle.

L'utilisateur de l'algorithme génétique doit choisir le plus petit alphabet qui permette une expression naturelle des paramètres du problème (principe des alphabets minimaux). C'est pourquoi l'alphabet binaire {0,1} est particulièrement bien adapté à la présentation des paramètres. Autrement dit, le codage utilisé sera le codage binaire [29].

# 2.7. Contraintes de pénalité

L'algorithme génétique crée de nouvelles valeurs de paramètres qui se trouvent dans les intervalles spécifiés. Cependant, il ne prend pas en compte les contraintes liant les différents paramètres. C'est pourquoi ces contraintes doivent être introduites après le décodage, avant de calculer la fonction d'adaptation[35]. Les méthodes suivantes peuvent être utilisées [31]:

- La fonction d'adaptation peut être toujours calculée : une fonction de pénalité est associée à la fonction d'adaptation pour chaque violation de contraintes[35]. Cette technique est appelée pénalité extérieure. Si la pénalisation est effectuée de sorte que  $F_{obj} = 0$  pour chaque cas de violation, cette méthode est dite pénalisation radicale [29].
- La fonction d'adaptation ne peut pas être calculée tout le temps : Il faut définir les paramètres prioritaires pour lesquels les valeurs fournies par l'algorithme génétique sont conservées. Les valeurs des autres paramètres sont calculées en prenant en compte les contraintes. Une pénalité est ensuite appliquée à la fonction d'adaptation [29].

# 2.8. Sélection des parents

La sélection des parents est un mécanisme qui fixe à partir d'une génération donnée, les individus qui pourront se reproduire pour former la génération suivante. Lors de cette phase, les individus les plus forts (les plus adaptés à l'environnement) sont généralement dupliqués et forment les parents de la génération en cours, alors que les individus les plus faibles disparaissent sans avoir la possibilité de se reproduire [31].

# 2.9. Méthodes de sélection

Les mécanismes de sélection dans les algorithmes génétiques jouent un rôle très important : d'une part pour diriger les recherches vers les meilleurs individus et d'autre part, pour maintenir la diversité des génotypes dans la population. Ils sont directement liés au compromis entre la vitesse de convergence élevée et une forte probabilité de trouver un optimum global dans le cas d'un problème multimodal. Ils sont notamment caractérisés par la probabilité de sélection  $P_s(a_i^t)$  qu'un individu  $a_i^t$  de la population  $P^t$  soit retenu pour participer à la recombinaison génétique [31].

Les techniques de sélection sont nombreuses et on peut en citer ici quelques-unes :

- Sélection proportionnelle (Proportional Selection ou Roulette Wheel Selection, RWS)
- Sélection proportionnelle à reste stochastique (Stochastic Remainder Selection, SRS)
- Sélection stochastique universelle (Stochastic Universal Selection, SUS)
- Sélection linéaire par rapport au rang (Linear Ranking Selection)
- Sélection uniforme par rapport au rang (Uniform

Pour les trois premières méthodes, la sélection devra être effectuée à la suite d'un réajustement de la fonction d'adaptation. Cette opération n'est pas nécessaire pour les autres.

# 2.10. Opérateurs de recombinaison génétiques

Les opérations de recombinaison génétique permettent de créer de nouveaux individus à partir des parents sélectionnés dans l'étape de sélection. Elles garantissent la diversification des individus dans la population, en assurant le brassage des matériels génétiques au cours des générations [31].

La recombinaison génétique, notée  $R_g$ , modifie la population comme suit :

$$p^{\prime\prime t} = R_g(P^{\prime t}) = (a_1^{\prime\prime t}, a_2^{\prime\prime t}, \dots, a_N^{\prime\prime t})$$
(a2.8)

Les deux opérateurs fondamentaux de la recombinaison génétique sont le croisement et la mutation.

#### 2.10.1. Croisement

Le croisement est un processus aléatoire de probabilité  $p_c$  appliqué séquentiellement à des couples de parents pris au hasard dans la population. Il consiste à échanger une partie du matériel génétique des parents pour former deux nouveaux individus (enfants) [29].

Des parents après croisement peuvent être retirés de la population de reproducteurs (croisement sans replacement) ou bien être gardés pour avoir une nouvelle chance de se reproduire (croisement avec replacement). C'est la première solution qui est généralement adoptée[36].

# 2.10.1.1. Croisement à un site

Considérons deux chaînes binaires de longueur l qui représentent des chromosomes de deux parents reproducteurs. Lorsque le croisement a lieu, un site de croisement est choisi entre les positions 1 et (l - 1) du chromosome. Le mécanisme de croisement consiste alors à échanger les gènes de chaque parent entre le site sélectionné et la position finale l des deux chaînes, comme le montre la figure 2.4 [31].



Figure 2.4 : principe de croisement à un site.

Si  $p_d(H)$  désigne la probabilité de destruction d'un schéma de longueur utile  $\delta(H)$  *et*  $p_c$  la probabilité de croisement, le taux effectif de destruction d'un schéma devient [30] :

$$P_{d,eff}(H) \le P_c \cdot P_d(H) \tag{a2.9}$$

La probabilité de survie est alors :

$$P_{s,eff}(H) \ge 1 - P_c \cdot P_d(H)$$
 (a2.10)

La relation précédente est générale et ne dépend pas du type de croisement. Dans le cas du croisement standard à un site,  $p_d(H)$  est donnée par :

$$P_d(H) = \frac{\delta(H)}{l-1} \tag{a2.11}$$

Où l est la longueur du chromosome et  $p_{s, eff}(H)$  peut être réécrite comme suit :

$$P_{s,eff}(H) \ge l - P_c \cdot \frac{\delta(H)}{l-1}$$
 (a2.12)

L'inéquation (38) montre que  $p_{s, eff}(H)$  diminue lorsque la longueur utile du schéma  $\delta(H)$  augmente. Donc, le croisement à un site préserve essentiellement des schémas de longueur utile courte [29].

#### 2.10.1.2. Croisement multi-sites

Le croisement peut être généralisé à n sites, la permutation des gènes étant effectuée entre chaque site. Si le nombre de sites est impair, on fixe un emplacement supplémentaire correspondant à la dernière position dans la chaîne comme pour le croisement à un site [31]. Le mécanisme du croisement généralisé est illustré à la figure 2.5.

Pour le croisement multi-sites, la probabilité de survie d'un schéma peut être exprimée récursivement par rapport à son ordre o(H), et en fonction de sa longueur utile  $\delta$ (H) [37].



Figure 2.5 : principe de croisement multi-sites.

# 2.10.1.3. Croisement uniforme

Le croisement uniforme [38] est obtenu à partir d'un masque binaire qui possède un nombre de bits égal au nombre de gènes des individus de la population. Ce masque est généralement construit de manière uniformément aléatoire, en respectant l'équiprobabilité d'obtenir une valeur égale à 0 ou 1 pour chaque bit. Toutefois, un croisement uniforme paramétré en fonction d'une probabilité supplémentaire  $p_0$ , liée à l'apparition d'une des deux valeurs (0 et 1) dans chaque bit du masque, a été proposé pour augmenter l'efficacité de l'opérateur [39]. Lorsque  $p_0 = 0.5$ , on retrouve évidemment le croisement uniforme standard [29].

Le premier enfant est créé en prenant les gènes du premier parent lorsque les bits correspondants dans le masque valent 1. Dans le cas contraire, les gènes du deuxième parent sont pris. Le deuxième enfant s'obtient de la même manière mais en utilisant le complément du masque au lieu du masque (voir la figure 2.6).

Bien que le nombre effectif de sites ne soit pas fixe à chaque recombinaison, on estime que le croisement uniforme produit une moyenne de 1/2 sites, l étant toujours la longueur totale du chromosome [29].

La probabilité de survie d'un schéma H soumis au croisement uniforme est donnée par

$$P_s(H) = \frac{1}{2^{o(H)-1}}$$
(a2.13)

Ainsi, les taux de survie et de destruction d'un schéma H ne sont pas liés alors à sa longueur utile  $\delta(H)$ , mais à son ordre o(H). Il devient alors difficile de quantifier l'effet de ce type de croisement [30].

Masque de croisement

0 0 1 1 0 1 0 1 0 1 0



Figure 2.6 : Principe du croisement uniforme.

# 2.10.2. Mutation

La mutation est une altération aléatoire d'un gène d'un individu. L'opérateur de mutation consiste à complémenter la valeur d'un bit du chromosome avec la probabilité p<sup>m</sup> (voir figure 2.7). Le processus est exécuté bit à bit [29].



# Enfant muté



:

La combinaison de la sélection et du croisement est nécessaire pour assurer l'évolution de la population et la convergence de l'algorithme vers l'optimum. Mais il arrive parfois que des informations essentielles contenues dans les gènes des individus disparaissent au cours des générations. Le rôle principal de la mutation est de protéger les individus contre cette perte irréparable de matériel génétique. Prise indépendamment, la mutation constitue un opérateur de recherche secondaire qui favorise l'apparition de génotypes nouveaux [31].

Pour qu'un schéma survive à l'opération de mutation, il faut que toutes ses positions instanciées soient conservées. Comme la probabilité de survie d'un gène quelconque est  $(1 - p_m)$  et que le nombre de positions instanciées dans un schéma H est o(H), la probabilité de survie d'un schéma à la mutation est :

$$P_{s}(H) = (l - P_{m})^{o(H)}$$
(a2.14)

Pour de très faibles taux de mutation, (40) peut être évalué approximativement comme

$$P_s(H) = l - o(H) P_m$$
 (a2.15)

Pour ne pas détériorer les performances de l'algorithme génétique, [32] propose une fréquence de mutation tous les 1000 bits. [40] et[34] conseillent la valeur suivante pour pm :

$$P_m = \frac{1}{l} \tag{a2.16}$$

Où l est la longueur du chromosome.

[41] propose une formule empirique qui exprime le taux optimal de mutation  $p_m$ en fonction de la longueur *l* du chromosome et de la taille *N* de la population [31]:

$$ln(N) + 0,93. ln(P_m) + 0,45. l = 0,56$$
 (a2.17)

et  $p_m$  est déterminée approximativement à partir de (43)

$$P_m = \frac{l}{N\sqrt{l}} \tag{a2.18}$$

Signalons enfin que dans plusieurs méthodes, dites méthodes de mutation adaptatives, la valeur de  $p_m$  sera changée au cours des générations.

# 2.11. Sélection des individus d'une nouvelle génération

A la suite de la combinaison génétique, la population comporte 2N individus (N parents et N enfants). Il faut donc éliminer N individus pour constituer la nouvelle génération. C'est le rôle de la sélection finale  $S_f$ , qui agit sur les populations de parents et d'enfants d'une génération pour créer la nouvelle génération [29]:

## 2.12. Critères de convergence d'un algorithme génétique

Un aspect important des algorithmes génétiques concerne le choix du critère de convergence. Quand peut-on considérer que l'algorithme a convergé vers l'optimum global ?

- Le problème est généralement éludé en imposant un nombre maximal de générations  $t_{max}$ et en arrêtant la recherche lorsque  $t = t_{max}$ . On estime alors que l'algorithme a convergé et que l'individu de plus forte performance dans la dernière génération  $P_{t_{max}}$  correspond à la solution recherchée.
- Une méthode plus rigoureuse consiste à supposer que l'algorithme converge vers l'optimum lorsque l'adaptation d'une partie ou de l'ensemble de la population se rapproche de celle du meilleur individu (voir la figure 2.8). On peut considérer que cet événement se produit à la génération t pour laquelle,

$$1 - \frac{f_{moy}}{f_{max}} \le \varepsilon \tag{a2.19}$$

Où  $\varepsilon$  est la précision requise sur la convergence,  $f_{max} = f(a_1^t)$  est la performance du meilleur individu de la population à la génération t et f<sub>moy</sub> est la moyenne de l'adaptation, calculée sur l'ensemble de population, ou sur une partie correspondant à un pourcentage des représentants les plus performants [31].



Génération

Figure 2.8 : Convergence de l'algorithme génétique.

Nous pouvons aussi supposer que l'algorithme a convergé lorsque le meilleur individu de la population n'évolue plus. Il suffit pour cela de tester la variation de sa performance sur une série de génération [43] :

$$\left|\Delta t.f(a_1^t) - \sum_{i=1}^{\Delta t} f(a_1^{t-i})\right| \cdot \frac{1}{\varepsilon + f(a_1^t)} \le \varepsilon$$
 (a2.20)

Ces critères ne sont pourtant absolument pas fiables dans la mesure où l'algorithme peut converger et se stabiliser autour d'une solution dans l'attente d'une mutation qui le dirigera vers une autre région plus intéressante.

# 2.13. Algorithmes génétiques à codage réel :

Dans l'algorithme génétique standard présenté précédemment, les paramètres de la fonction objectif sont codés, généralement en binaire, et ce sont leurs codes qui sont utilisés au lieu des paramètres eux-mêmes [29].

Nous orientons maintenant nos études sur les algorithmes génétiques à codage réel. Ils sont similaires à l'algorithme génétique standard, tant en ce qui concerne leur structure que leurs mécanismes de sélection. Par contre, ils sont caractérisés par des opérateurs de recombinaison génétique qui utilisent directement les valeurs des paramètres pour évoluer vers l'optimum. Le chromosome des individus comporte n gènes dont chacun est un paramètre réel (voir la figure 2.9) :



Figure 2.9 : Chromosome d'un individu dans les algorithmes génétiques à codage réel 2.13.1. Croisement

Deux types de croisement sont employés par les algorithmes génétiques à codage réel :

# 2.13.1.1. Croisement discret

Ce type de croisement est assez similaire aux méthodes de croisement utilisées dans l'algorithme génétique binaire. Il consiste à permuter les variables de chaque parent entre des sites du chromosome choisis aléatoirement (voir la figure 2.10). Le croisement discret peut être à un site, multi-sites ou uniforme [29].

Cette technique a un taux d'exploitation relativement limité car elle ne fait apparaître aucune nouvelle valeur pour les paramètres.



Figure 2.10 : principe du croisement discret.

# 2.13.1.2. Croisement continu

Ce type de croisement assure l'exploration de l'espace en testant de nouvelles valeurs des paramètres. Il est appliqué paramètre par paramètre en utilisant le mécanisme suivant :

$$\begin{cases} X'_{i} = X_{i} + a_{1}(Y_{i} - X_{i}) \\ Y'_{i} = Y_{i} + a_{2}(X_{i} - Y_{i}) \end{cases}$$
(a2.21)

Où xi, yi sont des paramètres de même rang de deux parents,  $\mathbf{X'}_i$ ,  $\mathbf{Y'}_i$  les paramètres, de rang Correspondant, des deux enfants obtenus. ai et a2 sont deux variables prises au hasard dans l'intervalle [0,1]. Lorsque ai  $\neq$  a2, le croisement est continu asymétrique. Lorsque ai = a2, le croisement est dit continu symétrique. Lorsque ai = a2 = 0.5, le croisement est continu arithmétique. Il conduit alors à un unique enfant représentant une moyenne arithmétique des parents. Dans ce cas, il est nécessaire d'effectuer un nombre de croisement deux fois plus grand afin d'obtenir autant d'enfants que de parents à chaque génération [29].

## 2.13.2. Mutation

La mutation consiste à perturber une variable du problème. Elle est généralement appliquée paramètre par paramètre, avec des probabilités plus élevées que pour le codage binaire standard[36]. Comme pour l'opérateur de croisement, plusieurs modes de mutation sont possibles :

#### 2.13.2.1. Mutation uniforme (ou mutation globale)

Ce type de mutation modifie un paramètre en choisissant une nouvelle valeur de façon uniformément aléatoire dans l'intervalle défini par les contraintes de domaine.

## 2.13.2.2. Mutation non uniforme (ou mutation locale)

La mutation non uniforme, ou mutation locale, consiste à perturber de façon locale (légèrement) la valeur d'un paramètre. Il est par exemple possible d'ajouter ou de retrancher à la variable aléatoire une grandeur  $\Delta x$  dont l'amplitude diminue avec le nombre de générations:

$$\mathbf{X}'_{\mathbf{i}} = \mathbf{X}_{\mathbf{i}} \pm \Delta \mathbf{X}_{\mathbf{i}} \tag{a2.22}$$
$\Delta x$  peut aussi être une variable à densité gaussienne dont l'amplitude est adaptée à chaque génération, comme pour les stratégies d'évolution [44] ou la programmation évolutionniste [42].

Les opérateurs de mutation locale confèrent à l'algorithme génétique à codage réel un mécanisme de recherche local. Contrairement à l'algorithme génétique à codage binaire, la mutation joue cette fois un rôle prépondérant dans l'exploration des nouvelles solutions [29].

#### 2.14. Conclusion

Dans cette annexe, nous avons présenté les bases fondamentales des algorithmes génétiques, sur la base de ces connaissances, nous avons défini la structure de notre travail, mise en œuvre dans le chapitre IV.

# Annexe 3

# SDIF (Simulateur de Débruitage des Images Fixes)

# SDIF (Simulateur de Débruitage des Images Fixes)

#### **3.1. Introduction**

Le but de débruitage est la construction d'une image significative et informative à partir d'une image bruitée. L'absence de l'image originale (qui est le but du débruitage) nous a conduit à réaliser un logiciel qui permet de prendre en considération ce cas.

L'entrée de l'interface de simulation est une image originale non bruitée qui sera dégradée en l'introduisant un bruit dont sa variance est contrôlée par le logiciel. La sortie est une image débruitée par une des méthodes implémentées. La comparaison du résultat obtenu avec l'image originale nous permet de déterminer le PSNR et MSE selon la méthode utilisée.

Dans le cadre de notre projet intitulé le débruitage des images fixes par les techniques de l'intelligence artificielle, nous avons mené une étude comparative entre différentes méthodes de débruitage déjà citées dans la littérature et notre méthode de débruitage. Ces méthodes sont implémentées dans le simulateur SDIF.

Dans le présent annexe, nous allons présentés notre simulateur conçu sous Matlab, comme le montre la figure 3.1. Nous avons implémenté onze méthodes de débruitage, dont la dernière est la méthode proposée dans ce travail



## **3.2. Interface du simulateur**

Figure 3.1 : Interface graphique du SDIF.

Notre interface graphique est conçue d'une manière à simplifier la manipulation, pour cela, nous avons organisé chaque méthode dans la catégorie qui lui convienne. On trouve la

catégorie des filtres, seuillage et intelligence artificielle (TNN et les algorithmes génétiques). Les catégories proposées dans le simulateur SDIF sont:

- Les filtres
- Seuillage
- Intelligence artificielle

#### 3.3. Mode d'utilisation

La première étape d'utilisation consiste à choisir une image de simulation (image dans l'interface), pour cela nous avons introduit les cinq images du référence dans le domaine du traitement d'image en l'occurrence (Lena, Barbara, Cameraman, Boat et Airplane) (figure 3.2).



Figure 3.2 : Sélection d'image.

La deuxième démarche consiste à bruiter l'image par un bruit blanc gaussien avec une variance de (5, 10, 15, 20, 25 ou 30) (figure 3.3).

-В	ruit gaussien	
	Variance	-
	Variance	
	5	
	10	
	15	
	20	har
	25	Tich
	30	

Figure 3.3 : Sélection de la variance du bruit.

Dans l'étape qui suit, il suffit d'appliquer une technique de débruitage en choisissant une catégorie. Pour la catégorie des filtres, il faux tout d'abord choisir la taille du masque (figure 3.1).

Tout au long de la simulation, les deux fenêtres à gauche permettent d'afficher les images désirées.

Les deux fenêtres performance permettent l'affichage le PSNR ainsi que la MSE (figure 3.4).



Figure 3. 4: Fenêtre des performances.

L'onglette afficher l'image (figure 3.5) permet de revoir les images débruitées et bruitées ainsi que leur MSE et PSNR. Elle permet aussi d'enregistré l'image affichée.

Afficher l'image
Image bruitée 👻
Enregistrer

Figure 3.5: Fenêtre d'affichage et d'enregistrement.

L'onglette outils permet les fonctionnalités suivantes : zooming, défilement, initialisation et enfin la fermeture du simulateur (figure 3.6).

Outils	
Zoom	Pan
initialisation	Fermer

Figure 3.6 : Onglette des outils.

## **3.4.** Conclusion

Dans cette annexe, nous avons présenté notre logiciel de simulation de débruitage des images fixes (SDIF), dont nous avons donné la démarches à suivre pour son utilisation ainsi qu'une brève présentation des ses fonctionnalités.

#### **Bibliographie**

- Yacov Hel-Or and Doron Shaked, « A Discriminative Approach for Wavelet Denoising», IEEE transactions on image processing, Vol. 17, N°. 4, PP, 443-457, APRIL 2008.
- [2] Larbi BOUBCHIR « Approches bayésiennes pour le débruitage des images dans le domaine des transformées multi-échelles parcimonieuses orientées et non orientées » thèse de doctorat de l'université de CAEN, 2007.
- [3] Fengxia Yan, Lizhi Cheng, and Silong Peng « A New Interscale and Intrascale OrthonormalWavelet Thresholding for SURE-Based Image Denoising », IEEE signal processing letters, Vol. 15, 2008, PP, 139-142.
- [4] D.Gnanadurai, and V.Sadasivam, « An Efficient Adaptive Thresholding Technique for Wavelet Based Image Denoising », International Journal of Signal Processing Volume 2 Number 2, 114-119.
- [5] X.-P. Zhang, «Space-scale adaptive noise reduction in images based on thresholding neural networks », in: Proceedings of IEEE International Con- ference on Acoustics, Speech, and Signal Processing, 2001, pp. 1889–1892
- [6] X.-P. Zhang, « Thresholding neural network for adaptive noise reduction », IEEE Trans. Neural Networks 12 (3) (2001) 567–584.
- [7] Mehdi Nasri, Hossein Nezamabadi-pour, «Image denoising in the wavelet domain using a new adaptive thresholding function neurocomputing», Neurocomputing 72, 2009, pp. 1012-1025.
- [8] Khelalef.A, Benoudjit.N, «Wavelet Image Denoising based on Genetic Algorithm and Cycle Spinning Thresholding», Submitted to Neurocomputing, ELSEVIER, (January 2012).
- [9] M. Ghazel « Adaptive Fractal and Wavelet Image Denoising », Thèse de doctorat Electrical and Computer Engineering, University of Waterloo, 2004.

- [10] Y. M. Hawwar « Nonlinear image filtering in the wavelet transform domain », Thèse de doctorat, The university of Wisconsin-Milwaukee, 2001.
- [11] S. Bres, J. M. Jolion et F. le bourgeois «Traitement et analyse des images numériques» , LAVOISIER, 2003.
- [12] L. Diane « Cours du traitement d'images » Université de Nice.
- [13] J. M. Vézien « Traitement d'images et vision par machines ».
- [14] Feature K. K. Gupta and R. Gupta, « Feature Adaptive Wavelet Shrinkage for Image Denoising », IEEE-ICSCN, Feb. 2007, India, pp. 81-85.
- [15] D. Donoho « Wavelet shrinkage and w.v.d : a 10 minute tour », Technical report. Standford University, 1992.
- [16] D. Donoho and I. Johnstone. « Ideal spatial adaptation by wavelet shrinkage » Biometrika, vol. 81, N°. 3, pp.425-455, 1994.
- [17] Gang HUA, « noncoherent image denoising», a thesis submitted in partial fulfillment of the requirements for the degree master of science, Houston Texas, 2005.
- [18] Wiem FOURATI , Mohamed Salim BOUHLEL « Techniques de Débruitage d'Images », 5th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications ,March 22-26, 2009
- [19] S. Grace Chang, Bin Yu and M. Vetterli, « Adaptive Wavelet Thresholding for Image Denoising and Compression », IEEE Trans. Image Processing, Vol. 9, pp. 1532-1546, Sept. 2000.
- [20] Lakhwinder KAUR, Savita GUPTA, R.C CHAUHAN, « Image Denoising using Wavelet Thresholding;Proceedings of the Third Indian Conference on Computer Vision, Graphics&Image Processing. 2002.
- [21] G. DREYFUS, «Les réseaux de neurones», Mécanique Industrielle et Matériaux, n°51, septembre 1998.
- [22] S. M E. Sahraeian, F. Marvasti, N. Sadati «wavelet image dénoising based on imroved thresholding neural network and cycle spinnig », ICASSP IEEE, 2007, pp. 585-588

- [23] R.R. Coifman and D.L. Donoho, « Translation invariant denoising », in Wavelets and Statistics, Springer Lecture Notes in Statistics 103, New York, Springer-Verlag, pp. 125-150, 1994.
- [24] D. Jonathan, B. Michael et S. Forthomme « Les Ondelettes »Université Libre de Bruxelles.
- [25] R. BENZID « Ondelettes et Statistiques d'Ordre Supérieur Appliquées aux Signaux Uni et Bidimensionnels » thèse de doctorat, 2005.
- [26] H. Egon, M. Marie, P. Porée « Traitement du signal et automatique », Harmann, Editeurs des Sciences et des Arts (chapitre 9 page 130).
- [27] F. Laouir, N. Kechacha « Conception et réalisation d'un logiciel pour la compression d'images fixes », Mémoire de fin d'études d'ingénieur, Département d'électronique, Université de Jijel 2007
- [28] C. L. Tan, « Still Image Compression Using Wavelet Transform », School of Information Technology and Electrical Engineering. The University of Queensland, Australia, 2001.
- [29] Dung HOANG, « détermination simultanée de la résistivité et de l'épaisseur des tôles métalliques et d'alliages non magnétiques par méthode électromagnétique sans contacte », thèse de doctorat, 2002
- [30] Omessaad HAJJI, « contribution au développement des méthodes d'optimisation stochastiques. Application à la conception des dispositifs électrotechniques », thèse de doctorat, 2003.
- [31] Bruno SARENI, « Méthodes d'optimisation multimodales associés a la modélisation numérique en électromagnétisme », thèse de doctorat, 1999.
- [32] D.E. GOLDBERG, « Genetic Algorithms in Search Optimization & Machine Learning », Addison Wesley, 1989.
- [33] L. SALUDJIAN, « Optimisations en Electrotechnique par Algorithmes génétiques », Thèse de Doctorat, INPG, N° 0073, 1997.

- [34] TH. BÄCK, « Optimal mutation rates in genetic search, S. Forrest (Ed.), Proceedings of the 5th International Conference on Genetic Algorithms and Their Applications », Morgan Kauffmann Publishers, pp. 2 - 9, 1993.
- [35] F. THOLLON, « Conception et optimisation des capteurs à courants de Foucault pour la détection de défauts profonds dans les matériaux amagnétiques », Thèse de Doctorat, Ecole Centrale de Lyon, France, 1995.
- [36] B. SARENI, « Méthodes d'optimisation multimodales associées à la modélisation numérique en électromagnétisme », Thèse de Doctorat, Ecole Centrale de Lyon, France, 1999
- [37] K.A. DE JONG, W.M. SPEARS, « A formal analysis of the role of multi-point crossover in genetic algorithms », Annals of Mathematics and Artificial Intelligence Journal, Vol. 5, No. 1, pp. 1 - 26, 1992
- [38] G. SYSWERDA, « Uniform crossover in genetic algorithms, J.D. Schaffer (Ed.), Proceedings of the 3rd International Conference on Genetic Algorithms and Their Applications », San Mateo, CA, Morgan Kauffmann Publishers, pp. 2 - 8, 1989
- [39] W.M. SPEARS, K.A. DE JONG, « On the virtues of parametrized uniform crossover, R.K. Belew and L.B. Booker (Eds.), Proceedings of the 4th International Conference on Genetic Algorithms and Their Applications », University of California, San Diego, Morgan Kauffmann Publishers, pp. 230 - 236, 1991
- [40] H. MÜHLENBEIN, « How genetic algorithm really work : I. mutation and hillclimbing,
  R. Männer and B. Manderick (Eds.) », Parrallel Problem Solving from Nature 2 PPSN II, Amsterdam, Elsevier, pp. 15 25, 1992
- [41] J.D. SCHAFFER, R.A. CARUANA, L.J. ESHELMAN, R. DAS, « A study of control parameters affecting online performance of genetic algorithms for function optimization », J.D. Schaffer (Ed.), Proceedings of the 3rd International Conference on Genetic Algorithms and Their Applications, San Mateo, CA, Morgan Kauffmann Publishers, pp. 51 - 60, 1989.
- [42] TH. BÄCK, « Evolutionary Algorithms in theory and practice », Oxford University Press, N.Y., 1996

- [43] J.A. VASCONCELOS, R.R. SALDANHA, L. KRÄHENBÜHL, A. NICOLAS, « Genetic algorithm coupled with a deterministic method for optimization in electromagnetics », IEEE Transactions on Magnetics, Vol. 33, No. 2, pp. 1860 - 1863, 1997
- [44] H.P. SCHWEFEL, « Evolution and Optimum Seeking, Sixth- Generation Computer Technology Series », John Wiley & Sons, N.Y., 1995.

#### Résumé

Les systèmes d'imagerie occupent une place de plus en plus importante dans de nombreux domaines de la science et de l'industrie. Ces systèmes mettent en œuvre des capteurs d'images de qualité variable, et ils trouvent leurs applications dans de nombreux domaines : biomédical, astronomie, environnement, contrôle industriel, surveillance, restauration de films anciens, etc. Dans tous ces domaines, l'objectif commun est d'extraire l'information utile des images. Cette extraction est souvent rendue difficile, soit par la dégradation de ces images par des fluctuations stochastiques que l'on qualifie généralement de bruit, soit par une qualité médiocre des capteurs pour réduire les coûts de fabrication, soit par la nature des environnements.

En général un des problèmes essentiels pour l'utilisation des images est le bruit qui s'ajoute entraînant par la suite des dégradations aux images d'origines.

Dans ce travail, nous nous intéresserons principalement au problème de débruitage des images fixes en utilisant les méthodes de l'intelligence artificielle. Nous avons implémenté dix techniques de débruitage en l'occurrence les filtres, les méthodes récentes basées sur le seuillage et deux technique qui utilisent les TNN (Thresholding Neural Network), à la fin du travail avons proposé une nouvelle technique basée sur les algorithmes génétiques et l'algorithme Cycle Spinning.

#### Summary

Imaging systems play an increasingly important step in many fields of science and industry. These systems utilize image sensors of varying quality, and they find applications in many areas: biomedicine, astronomy, environment, industrial control, monitoring, restoration of old films, etc... In all these areas, the common goal is to extract useful information from images. This extraction is often difficult, by the degradation of these images by stochastic fluctuations that are usually referred as noise, or by poor quality sensors to reduce manufacturing costs, either by the nature of environments.

In general one of the key issues for the use of images is the noise that is added in turn results in degradation to the original image.

In this paper, we focus mainly to the problem of denoising of images using the methods of artificial intelligence. We implemented ten denoising techniques in this case filters, recent methods based on thresholding and two that use the TNN (Thresholding Neural Network), at the end of the work we have proposed a new technique based on genetic algorithms and Cycle Spinning algorithm.