



HAL
open science

Modélisation des images par patchs pour leur restauration et leur interprétation. Applications à l'imagerie SAR.

Sonia Tabti

► **To cite this version:**

Sonia Tabti. Modélisation des images par patchs pour leur restauration et leur interprétation. Applications à l'imagerie SAR.. Traitement du signal et de l'image [eess.SP]. Télécom ParisTech, 2016. Français. NNT: . tel-01585676

HAL Id: tel-01585676

<https://theses.hal.science/tel-01585676>

Submitted on 11 Sep 2017

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

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



EDITE - ED 130

Doctorat ParisTech

THÈSE

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Images »

présentée et soutenue publiquement par

Sonia TABTI

le 23 Juin 2016

Modélisation des images par patches pour leur restauration et leur interprétation Applications à l'imagerie SAR

Directeur de thèse : **Florence TUPIN**

Co-encadrement de la thèse : **Loïc DENIS et Charles-Alban DELEDALLE**

Jury

M. Abdourahmane ATTO, Polytech Annecy-Chambéry

Mme. Marie CHABERT, INP - ENSEEIHT Toulouse

M. Pierre CHAINAIS, École Centrale de Lille

M. Charles-Alban DELEDALLE, Université de Bordeaux 1

M. Loïc DENIS, Université de Saint-Étienne

Mme. Agnès DESOLNEUX, ENS Cachan

M. Philippe RÉFRÉGIER, École Centrale de Marseille

Mme. Florence TUPIN, Télécom ParisTech

Examineur

Examineur

Rapporteur

Encadrant

Co-directeur

Rapporteur

Président

Directeur

TELECOM ParisTech

école de l'Institut Mines-Télécom - membre de ParisTech

46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

Remerciements

Tout d'abord, je tiens à remercier mes directeurs de thèse. Charles, tu es le meilleur grand frère de thèse dont on puisse rêver. Tu m'as toujours aidée et soutenue depuis mon stage de Master jusqu'à la fin de ma thèse. J'ai également beaucoup apprécié travailler avec toi durant mes séjours à Bordeaux qui m'ont toujours motivée. Loïc, je te remercie d'avoir partagé avec moi tes nombreuses idées toujours très créatives et qui m'ont bien fait avancer dans mes travaux. Enfin Florence, j'admire le recul et l'expérience avec lesquels tu m'as guidée et ta disponibilité pendant ces quatre dernières années. Un grand merci à vous !

Ensuite, je souhaiterais remercier les membres de mon jury, Pierre Chainais et Agnès Desloneux, pour avoir accepté d'être mes rapporteurs, Abdourrahame Atto, Marie Chabert et Philippe Réfrégier pour avoir accepté de les examiner. Vos questions et vos remarques m'ont beaucoup apporté.

Merci au CNRS et à la DGA d'avoir financé mes travaux de thèse à Télécom ParisTech, lieu très stimulant en terme de recherche. J'aimerais remercier les permanents avec qui j'ai pu discuter, en particulier Joseph Salmon, Pooran Memari, Jean-Marie Nicolas, Henri Maître, Isabelle Bloch, Guiseppa Valenzise et Andrés Almansa pour leur gentillesse et leur humanité.

Merci également à l'Université Paris Descartes de m'avoir accueillie en temps qu'ATER pour ma quatrième année de thèse, ce qui m'a permis de revenir en ce lieu cher à mon cœur où j'ai fait la plus grande partie de mes études. J'aimerais remercier l'équipe pédagogique du laboratoire du MAP 5 avec qui j'ai collaboré et qui a permis que j'apprécie autant cette expérience. En particulier, j'aimerais remercier la directrice de l'UFR Maths-Info Christine Graffigne et la directrice du MAP 5, Fabienne Comte, de m'avoir encouragée à accepter cette opportunité et qui m'a toujours bien conseillée même pendant mes études. Merci également aux permanents avec qui j'ai eu beaucoup de plaisir à discuter (désolée si j'en oublie), Annie Raoult, Florent Benaych-Georges, Rachid Lounes, Eric Luçon, Camille Male, Lionel Moisan, Julie Delon et Bruno Galerne. Un remerciement spécial aux permanents qui m'ont beaucoup motivée à l'époque de mes études : Pierre Calka et Hermine Biermé, qui ne sont plus à Descartes aujourd'hui, Nicole Rigal et Georges Koepfler, sans votre soutien je ne pense pas que je serais allée

aussi loin !

J'ai eu beaucoup de chance pendant ma thèse à Télécom ParisTech car j'ai pu connaître deux générations de doctorants. En effet, l'année où j'ai commencé ma thèse peu de doctorants avaient été recrutés à Télécom même et la plupart des doctorants commençaient leur troisième année de thèse. Ce fut un plaisir de profiter de leur expérience ainsi que celle des post-doctorants et de partager pas mal de fous rires et de débats. En particulier (encore désolée pour ceux que j'ai pu oublier), merci à Flora D., pour nos joyeux papotages, Xin pour ta gentillesse, Tartav', pour ton appétit gargantuesque et tes blagues qui ont déteint sur moi, Thierry, notamment pour tes tartes au citron meringuées :D, Alasdair pour ton optimisme permanent, Romane pour le thé ;), Cecilia, Baptiste, Yann T., Émilie G., David G., JC, Loïca (merci pour tes mails d'encouragements et bon courage pour ta soutenance!) et bien sur Edoardo pour nos fous rires à chaque fois que j'essayais de toucher ton ordinateur !

Pour la deuxième génération de doctorants télécomiens, mes pensées vont vers Flora W., ta fantaisie, ta gentillesse sont une inspiration pour moi, Guillaume Ter., j'admire notamment ta force de travail et ta capacité à exceller dans tous les sports (!!), Yifan, j'espère que dans ton coeur tu te souviendras toujours de la "Yifan crew", Sylvain, ce fut une joie de partager le bureau avec toi, merci de de m'avoir écoutée et soutenue, Paul, juste un mot : gogole :D, Antoine, j'espère qu'on collaborera ensemble à l'avenir car c'est toujours agréable de discuter avec toi, tu fais partie de ces gens avec qui il est facile pour moi de travailler et rigoler en même temps, Clément, Weying, Hélène U., avec toi je suis sûre que la tradition de la Friday beer est entre de bonnes mains et que tu la maintiendras, Christian, Gang, Clara, Tim, Ana-Maria, Malik, Gilles, Hélène L., Adrien, Gianni, Yann L., Alessio. Bien sur, que seraient mes remerciements s'ils ne parlaient pas du meilleur stagiaire de tous les temps : Carlos, tu es la gentillesse incarnée, merci de répandre l'amour et les gâteaux autour de toi, ne change rien et je te souhaite une belle thèse à Brest.

J'ai eu la chance également d'avoir des supers collègues à Descartes pendant mon ATER. Je connaissais la plupart d'entre vous grâce à Arthur et j'étais ravie de devenir votre collègue à mon tour. Je dois préciser également que j'ai retrouvé quelques collègues de l'ancienne génération télécomienne et que c'était très agréable de faire un mini Télécom-revival à nous quatre : Thierry, Alasdair, Edoardo, merci pour nos fous rires sur la terrasse et votre soutien sans failles ! Un Big up à mes collègues de bureau, qui m'ont toujours donné le sourire même les jours où je ne l'avais pas forcément : Anne-Sophie, Pierre (Anaïs je te cite ici si ça te dérange pas :D), Rémy, Cambyse, Mélina, Oriël et Vincent ! C'était avec joie aussi que j'ai pu apprécier papoter avec mes collègues des autres bureaux comme : Jean, Alkéos (vive les déj. au CROUS!), Charlotte D., Charlotte L., Maud, Loïc (mon jumeau de date de soutenance!), Gwenaëlle, Christelle, Andréa et

Julie F. .

Un grand merci à tous mes amis de m'avoir épaulée toutes ces années et pour certains d'entre eux tenté de comprendre le monde de la recherche à travers mes récits. Merci à Yasmine : depuis l'école primaire, et malgré la distance, on ne s'est jamais quittées, je suis très heureuse de t'avoir comme amie. Merci à Souad et Alex, Darina et Thomas et Hana pour m'avoir toujours donné du pep's et pris soin de moi. Merci à Ghalia et Simo, notamment pour nos séances de boulot ensemble chez toi Ghalia pendant nos études ! Je n'aurais pas aussi bien réussi ni été aussi bien ces années là sans toi qui me comprend à demi mot, sache que tu me manques beaucoup ! Vijika, tu sais ce que je ressens pour toi, je te remercierai jamais assez d'être toujours présente pour moi... Stéphanie et Xuyen, depuis la fac, assises ensemble à la bibliothèque jusqu'à maintenant quand on se retrouve pour dîner, on s'est toujours motivées, confiées et faite rire. Merci à Fatima, Vicent Vong, Émilie K. et Romain, Camille C., Laure, Stéphane et Omblin, Jehanne et Élie, Nicolas B., Sébastien, Marjorie et Cléa : que ce soit à discuter et rigoler ensemble, ou à travers des textos d'encouragements, vous m'avez mis du baume au coeur et c'est grâce à ces moments que je trouvais la force d'avancer, j'ai beaucoup de reconnaissance envers vous pour cela.

Enfin, je souhaiterais exprimer toute ma gratitude envers mes deux grandes soeurs et leurs petites familles qui ont toujours veillé sur moi et conseillée, merci à Aïda, Raphaël, Gabriel mon neveu chéri, Linda, Fares et l'adorable Adam, le petit dernier dans la famille ! Que dire à mes parents qui ont tout donné pour moi et m'ont poussé à toujours aller plus loin dans mes études ? Encore une fois merci à mon père, ma mère, Yamina, Dahbia, Aïni, Salah et Belaïd et j'en passe car la liste est longue ! Et que dire à ma moitié grâce à qui mon cœur s'emplit de joie quand j'ouvre la porte de la maison, même si parfois ce n'est pas l'état dans lequel je suis avant de l'ouvrir ? Merci Arthur d'être là pour moi et de veiller à mon bonheur. Mes derniers mots vont vers Ouardia, mes grands parents et Vouvouche : je pense fort à vous.

Table des matières

1	Introduction	13
1.1	Contexte	13
1.2	Objectifs et contributions	14
1.2.1	Décomposition parcimonieuses de patches	14
1.2.2	Modélisation de la distribution des patches par un mélange de Gaussiennes (GMM)	15
1.3	Organisation du manuscrit	16
2	Modélisations statistiques des images	17
2.1	Modélisations de signaux	17
2.1.1	Systèmes d'acquisition	17
2.1.2	Modèles de bruit	21
2.2	Bref état de l'art des méthodes de débruitage	23
2.3	Modélisation locale de l'espace des patches	26
2.3.1	Premiers modèles basés sur la redondance des patches d'une image	26
2.3.2	Modélisation locale de l'espace des patches avec des modèles Gaussiens	28
2.4	Modélisation globale de l'espace des patches	31
2.4.1	Des champs de Markov aux champs d'experts	33
2.4.2	Modèles parcimonieux basés sur les patches	36
2.4.3	Modélisation globale de l'espace des patches avec les GMM	40
2.5	Conclusion	48
3	Propriétés d'invariance des modèles d'images	51
3.1	Motivation et définition des invariances considérées	51
3.2	Approches invariantes par translation	54
3.2.1	État de l'art des approches invariantes par translation	54
3.2.2	Un a priori invariant par translation	58
3.3	Approches invariantes aux transformations radiométriques affines	61
3.3.1	État de l'art des approches invariantes aux transformations radiométriques affines	61
3.3.2	Modéliser une approche par GMM avec invariance affine	62
3.3.3	Apprentissage d'un GMM aligné (aGMM)	65

3.3.4	Débruitage par l'aGMM	66
3.3.5	L'aEPLL : EPLL + aGMM	67
3.3.6	Apport de l'alignement	69
3.4	Approches 1-sparse invariantes affines et par translation	73
3.4.1	Étude du degré de parcimonie optimal sous fort bruit	73
3.4.2	Un critère de similarité entre patches invariant par translation et transformation radiométrique affine	74
3.4.3	Un GLR-test invariant affine	78
3.4.4	Cartes d'erreurs d'approximation d'une image par un atome avec invariances	79
3.5	Conclusion	81
4	Applications à l'imagerie SAR	83
4.1	Restauration d'images SAR	83
4.1.1	Plusieurs attaches aux données	84
4.1.2	Plusieurs familles de méthodes	86
4.2	Apprentissage d'un dictionnaire 1-sparse avec invariances	90
4.2.1	Procédure d'apprentissage	91
4.2.2	Limites et perspectives	93
4.3	L'EPLL invariant par translation et adapté aux images SAR	96
4.3.1	Présentation de la méthode	96
4.3.2	Résultats	99
4.4	Apprentissage d'un GMM radar	100
4.4.1	Protocole d'apprentissage	100
4.4.2	Évaluation de l'apport de ce GMM	102
4.5	Une méthode de classification basée sur les GMM	103
4.5.1	Quelques méthodes de classification d'images SAR	105
4.5.2	Principe de la méthode proposée	107
4.6	Conclusion	110
5	Conclusions et perspectives	113
5.1	Approches parcimonieuses	113
5.2	Approches basées sur les GMM	114
5.3	Futures applications aux images SAR	114
A	Base d'apprentissage de l'aGMM	119
B	Base d'images de l'algorithme de classification	121

Notations

De façon générale, les caractères gras représentent des vecteurs ou des matrices tandis que les caractères non-gras représentent des scalaires.

Acronymes

- aGMM : GMM aligné
- aGMM-D : débruitage basé sur l'aGMM, aGMM Denoising en anglais
- DCT : Discrete Cosine Transform
- EM : algorithme Espérance-Maximisation, Expectation-Maximization en anglais
- EPLL : Expected-Patch-Log-Likelihood
- EPLL-FT : EPLL avec une attache aux données Fisher-Tippett
- GLR : Generalized Likelihood Ratio
- GM : Gaussian Model
- GFM : Gaussian Factor Models
- GMM : Gaussian Mixture Models
- K-SVD : algorithme K-Singular Value Decomposition
- MRF : Markov Random Fields
- MAP : Maximum A Posteriori
- MSE : Mean Square Error, en français : EQM pour écart quadratique moyen
- OMP : Orthogonal Matching Pursuit
- PCA : Principal Component Analysis
- SI-EPLL : Shift-Invariant EPLL
- $\text{SSD}_{\text{map}}^{\text{AI+SI}}$: Cartes d'erreurs d'approximation d'une image par un atome où SSD abrège Sum of Square Differences, AI : Affine-Invariant et SI : Shift-Invariant

De l'image au pixel

- \mathbf{y} : image
- w : largeur d'un patch carré
- \mathbf{y}_j : j -ème patch extrait de \mathbf{y}
- \mathbf{Y} : matrice dont chaque colonne est un patch de l'image \mathbf{y}
- $\mathbf{R}_j \mathbf{y}$: Opérateur d'extraction du patch centré sur le j -ème pixel de l'image \mathbf{y} tel que $\mathbf{x}_j = \mathbf{R}_j \mathbf{y}$
- y_j : j -ème pixel de l'image \mathbf{y}

Généralités

- \bar{y} : moyenne de \mathbf{y}
- $\bar{\mathbf{y}}$: \mathbf{y} centrée
- \hat{y} : estimation de y
- \mathbf{I} : matrice identité
- $\mathbb{1}$: vecteur dont toutes les entrées valent 1
- $\mathcal{F}[\mathbf{y}]$: transformée de Fourier (TF) de \mathbf{y}
- $\mathcal{F}[\mathbf{y}]^*$: conjuguée de la TF de \mathbf{y} (TF inverse de \mathbf{y})
- $*$: opérateur de convolution
- \mathbf{B}^t : transposée de la matrice/vecteur \mathbf{B}
- $\det(\mathbf{B})$: déterminant d'une matrice \mathbf{B}

Dictionnaires de patches

- \mathbf{D} : dictionnaire avec des atomes sous forme de colonnes
- \mathbf{A} : opérateur d'analyse
- \mathbf{d} : un atome du dictionnaire
- \mathbf{d}_k : k -ème atome d'un dictionnaire
- \mathbf{u}_i : vecteur colonne dont chaque entrée donne le poids d'un atome dans la décomposition d'un patch
- \mathbf{U} : matrice dont chaque colonne est un vecteur \mathbf{u}_i

GMM

- \mathbf{C}_k : la matrice de covariance associée à la k -ème composante d'un GMM
- $\boldsymbol{\mu}_k$: le vecteur moyen associé à la k -ème composante d'un GMM
- ω_k : poids associé à la k -ème composante d'un GMM

Translation et transformation affine

- α, β : paramètres scalaires de transformation affine du type : $\alpha \mathbf{y}_j + \beta \mathbb{1}$
- \mathbf{T}_p : opérateur de translation 2D entière de paramètre $\mathbf{p} = (h, v)$
- ι : image représentant le support d'un patch de taille $w \times w$

Débruitage Sauf mention explicite :

- \mathbf{x} : image sans bruit
- \mathbf{y} : image bruitée
- PSNR : mesure de performance de reconstruction d'une image débruitée par rapport à l'image sans bruit définie par :
 $10 \log_{10}(d^2/\text{EQM})$, où d est la dynamique de l'image.
- SSIM : Structural SIMilarity est une mesure de similarité entre deux images. Elle se calcule sur plusieurs fenêtres de taille $N \times N$ tirées des deux images avec la formule suivante : $\text{SSIM}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{(2\mu_{\mathbf{x}}\mu_{\hat{\mathbf{x}}}+c_1)(2\sigma_{\mathbf{x}}\sigma_{\hat{\mathbf{x}}}+c_2)(2\text{cov}_{\mathbf{x}\hat{\mathbf{x}}}+c_3)}{(\mu_{\mathbf{x}}^2+\mu_{\hat{\mathbf{x}}}^2+c_1)(\sigma_{\mathbf{x}}^2+\sigma_{\hat{\mathbf{x}}}^2+c_2)(\sigma_{\mathbf{x}}\sigma_{\hat{\mathbf{x}}}+c_3)}$ où $\mu_{\mathbf{a}}$ est la moyenne de \mathbf{a} ,

$\sigma_{\mathbf{a}}$ est la variance de \mathbf{a} , $cov_{\mathbf{x}\hat{\mathbf{x}}}$ est la covariance de \mathbf{x} et $\hat{\mathbf{x}}$ et les c_i des constantes stabilisant la division lorsque le dénominateur est faible.

Paramètres de distributions

- σ^2 : variance d'un bruit gaussien
- L : nombre de vues (paramètre du bruit gamma ou Nakagami-Rayleigh)

Chapitre 1

Introduction

1.1 Contexte

Le début des années 2000 a connu l'essor des approches par patches en traitement d'images. On parle de patches pour désigner des imagerie de petite taille, 8×8 pixels par exemple. Il existe plusieurs manières d'utiliser des patches dans une méthode de restauration d'image. Parmi les méthodes les plus célèbres, citons les moyennes non-locales ou NL-Means en anglais [Buades *et al.* (2005)]. Cette méthode tire profit du fait que les motifs observés dans les images sont répétitifs, ce qui est désigné dans la littérature par la propriété d'autosimilarité des images. Le principe des NL-Means est de débruiter le pixel central d'un patch d'intérêt en calculant la moyenne des pixels centraux de patches qui lui sont similaires. Bien que cette méthode présente des résultats très satisfaisants, des résultats significativement meilleurs ont pu être obtenus via des modèles par patches plus sophistiqués comme il sera détaillé dans le manuscrit.

Une des ces autres approche est proposée dans [Aharon *et al.* (2006)] où l'objectif est de représenter les patches de façon parcimonieuse avec un dictionnaire d'atomes. Un dictionnaire est une collection de patches (atomes) prédéfinie ou apprise, représentant des contours avec diverses orientations, des cibles, des textures et possiblement des structures rares dans l'image. L'approximation d'un patch tiré d'une image est alors donnée par une combinaison linéaire de quelques atomes. Cette combinaison linéaire doit être parcimonieuse, ou *sparse* en anglais, c'est à dire que peu de coefficients de cette combinaison doivent être non-nuls ou encore que peu d'atomes interviennent dans la représentation du patch.

Ces deux approches, l'une s'appuyant sur la similarité des patches et l'autre sur leur représentation par un dictionnaire ont connu un grand succès en traitement d'images et ont donné naissance à de nombreuses variantes. Dans ce manuscrit de thèse, nous présenterons diverses modélisations statistiques de la distribution des patches et nous proposerons de nouveaux modèles basés sur des approches par dictionnaires entre autres.

Conjointement à l'essor des approches de traitement d'images basées sur les patches, plusieurs satellites RADAR (RAdio Detection And Ranging) à Synthèse d'Ouverture, abrégé RSO ou SAR en anglais pour Synthetic Aperture Radar, ont été mis en orbite ces dernières années. Nous disposons donc de plus en plus de données de ce type à traiter. Ce sont des images synthétisées à partir d'échos d'ondes électromagnétiques rétro-diffusées par la scène imagée. Elles ont l'avantage de ne pas dépendre de l'éclairement solaire, contrairement aux capteurs optiques, et peuvent donc être acquises de jour comme de nuit et en présence ou non d'une couverture nuageuse. Ce système d'acquisition entraîne un phénomène de chatoiement, ou speckle en anglais, qui est modélisé par un bruit multiplicatif. Il est par conséquent nécessaire d'élaborer des approches tenant compte des statistiques particulières de ces images.

Compte tenu des masses importantes de données à modéliser, nous nous sommes intéressés dans cette thèse à l'introduction de propriétés d'invariances dans les approches par dictionnaires. Nous considérerons d'une part l'invariance par translation et d'autre part, l'invariance aux transformations radiométriques affines. Ces propriétés sont attrayantes car les dictionnaires de patches sont souvent redondants et ils peuvent contenir la même information d'un atome à l'autre à une translation ou à une transformation affine près. Construire de tels modèles permettra à long terme de pouvoir apprendre des dictionnaires plus petits représentant au mieux la distribution des patches du type de données dont ils ont été extraits.

1.2 Objectifs et contributions

Dans ce contexte, les objectifs de cette thèse sont d'une part de développer des modèles adaptés au traitement des images SAR et d'autre part d'introduire des propriétés d'invariance dans ces modèles. Nous avons développé deux types de modèles : premièrement des modèles basés sur des approches par dictionnaires de patches, et deuxièmement, des modèles basés sur des GMM (Gaussian Mixture Models). Nous résumons ci-dessous les contributions qui seront présentées dans ce manuscrit.

1.2.1 Décomposition parcimonieuses de patches

Dans le contexte des dictionnaires de patches, nous avons tout d'abord développé un critère de similarité invariant par translation et transformation affine entre un patch bruité et un atome sans bruit basé sur le calcul d'une corrélation normalisée. Une application de ce critère au débruitage a été réalisée. Par ailleurs, un test de rapport de vraisemblance généralisé évaluant la similarité d'un patch bruité et d'un patch sans bruit avec invariance par transformation affine a été proposé dans [Deledalle *et al.* (2013)].

Nous avons étendu ce critère au calcul de carte d'erreur d'approximation d'une image bruitée par un patch sans bruit avec invariance par transformation affine et par translation.

Du point de vue applicatif, une méthode d'apprentissage de dictionnaires dédiée aux images SAR muni des deux invariances considérées a été développée à partir des cartes d'erreurs d'approximation mentionnées plus haut [Tabti *et al.* (2014a)]. L'invariance par translation étant obtenue grâce à l'aptitude des atomes du dictionnaire à paver (et résumer) l'image de référence sur laquelle il est appris.

1.2.2 Modélisation de la distribution des patches par un mélange de Gaussiennes (GMM)

Les modèles de mélanges de Gaussiennes présentent des performances état de l'art dans diverses applications de traitement d'images comme le débruitage ou encore l'inpainting. Le principe des approches par patches basées sur des GMM est de représenter la distribution d'un patch par la somme pondérée de modèles Gaussiens caractérisés par un vecteur moyen et une matrice de covariance riche en informations.

Plusieurs travaux ont été menés dans cette thèse avec ce type de modèle. D'un point de vue méthodologique, nous avons apporté les contributions suivantes :

- Nous avons formulé un a priori basé sur des GMM en introduisant une propriété d'invariance par translation. Nous avons utilisé le même principe de pavage de l'image qu'avec les approches par dictionnaires mentionnées plus haut.
- De plus, nous avons proposé une façon de modéliser statistiquement l'invariance radiométrique avec un GMM et élaboré une méthode d'apprentissage et une méthode de débruitage avec une version simplifiée de GMM muni de cette invariance.

Du point de vue des applications aux images SAR,

- nous avons comparé les performances d'une méthode de restauration d'images SAR utilisant un GMM appris sur des images naturelles ou un GMM appris sur le logarithme de l'intensité des images SAR. Nous avons constaté que les résultats sont proches. Les patches d'une images SAR transformée logarithmiquement auraient donc une distribution proche des patches tirés d'images naturelles quand ils sont modélisés par des GMM.
- Nous avons également adapté l'algorithme EPLL basé sur des GMM [Zoran et Weiss (2011)] aux images SAR et introduit une propriété d'invariance par translation dans cette méthode grâce à l'a priori décrit ci-dessus [Tabti *et al.* (2014b)]. Les résultats obtenus sont du même ordre que les méthodes état de l'art.
- Finalement, nous avons développé une procédure supervisée de classification d'images SAR basée sur des GMM [Tabti *et al.* (2015)]. Ses résultats sont satisfaisants et elle est rapide et simple à mettre en œuvre.

1.3 Organisation du manuscrit

Quatre chapitres vont suivre cette introduction. Le **chapitre 2** est un chapitre bibliographique. Après avoir souligné l'importance de la modélisation du système d'acquisition d'une image et présenté divers modèles de bruit, nous décrivons plusieurs manières de modéliser la distribution des patches tirés d'images naturelles. Nous décrivons également comment utiliser ces modèles dans des applications au débruitage de bruit Gaussien.

Le **chapitre 3** traite de la modélisation de la distribution des patches avec des propriétés d'invariances par translation et transformation affine. D'une part, nous décrivons un état de l'art des méthodes invariantes par translation et invariantes par transformations radiométriques affines respectivement. D'autre part nous décrivons nos contributions dans l'élaboration de méthodes munies d'invariance par translation et/ou d'invariance aux transformations radiométriques affines.

Le **chapitre 4** présente les applications de nos travaux de thèse aux images SAR. Une première partie expose un état de l'art des méthodes de restauration des images SAR. Les parties suivantes traitent de nos contributions sur les modèles parcimonieux et les modèles basés sur les GMM. Après un état de l'art des méthodes de classification d'images SAR, nous présenterons un algorithme de classification s'appuyant sur les GMM. Enfin, le **chapitre 5** récapitule les conclusions de nos travaux et les travaux à venir.

Chapitre 2

Modélisations statistiques des images

La modélisation statistique d'un signal dépend d'une part de la manière dont il est acquis et d'autre part généralement d'un a priori choisi pour le modéliser. Afin d'illustrer l'influence d'un système d'acquisition, nous en décrirons plusieurs en section 2.1.1, notamment la photographie numérique et plusieurs modes d'imagerie en télédétection. Nous présenterons en particulier l'imagerie radar sur laquelle la plupart des applications de ce travail de thèse vont se concentrer.

Un des facteurs qui peut rendre le traitement et l'interprétation des images plus difficiles est le bruit lié au système d'acquisition. Il existe de nombreux types de bruit modélisés par différentes lois de probabilités. Nous décrirons les types les plus répandus avec le bruit de Poisson, le bruit blanc Gaussien et le speckle en section 2.1.2.

Enfin, après un bref historique des méthodes de débruitage adaptées au bruit Gaussien, nous décrirons différents a priori. Nous distinguerons les modèles ayant pour objectif de traiter une image en modélisant l'espace global des patches (cf. sec. 2.4), puis d'autres modèles ayant pour objectif de traiter une image en modélisant localement l'espace des patches (cf. sec. 2.3).

2.1 Modélisations de signaux

2.1.1 Systèmes d'acquisition

Un signal peut se présenter sous diverses formes. Par exemple, les électrocardiogrammes (ECG) sont des signaux en une dimension (1D) et les données images qu'on peut voir comme un tableau de pixels sont en 2D. Elles peuvent être constituées d'un seul canal lorsqu'il s'agit d'une image en niveaux de gris ou de trois canaux RGB (Red : rouge, Green : vert, Blue : bleu) lorsqu'il s'agit d'une image couleur ou de plusieurs canaux pour les images multispectrales et hyperspectrales. Il existe aussi des signaux en 3D comme par exemple : des maillages de polygones. Des exemples de ces signaux sont illustrés en figure 2.1.



FIGURE 2.1 – *Exemples de signaux. De gauche à droite : exemple de signal 1D avec un ECG, une image couleur et un signal 3D : un maillage.*

Dans cette thèse, nous nous concentrerons sur les images mono-canal et plus particulièrement dans cette section, nous mettrons en évidence la diversité des systèmes d’acquisition des images dont on doit tenir compte afin de les traiter convenablement. Pour illustrer nos propos, nous décrirons brièvement le principe de la photographie numérique, puis quelques types d’imagerie en télédétection. On parlera des images issues de satellites optiques, d’images infra-rouge (IR) et d’images SAR pour Synthetic Aperture Radar ou RSO, en français, pour Radar à Synthèse d’Ouverture.

Le capteur le plus répandu en photographie numérique est le CCD, pour Charged-Coupled Devices. Il est divisé en plusieurs cellules qui réagissent à la lumière (aux photons) en accumulant une charge électrique donnant après lecture une valeur de luminance au pixel. Ces étapes vont être sources d’un bruit thermique généralement modélisé par un bruit Gaussien et d’un bruit de comptage de photons (typiquement, un bruit de Poisson) qui seront décrits en section 2.1.2.

Les applications de cette thèse étant tournées vers la télédétection, nous commençons par mentionner deux exemples en télédétection optique avec les images satellitaires optiques, dont un exemple est présenté en figure 2.2, et les images IR. Les images issues de ces satellites sont formées en exploitant soit les données de réflectance de l’éclairement solaire (visible et proche IR), soit le rayonnement propre des sols (IR thermique), soit un mélange de ces deux propriétés (MIR : moyen infrarouge). Il existe plusieurs types de capteurs de ces deux types d’imageries dont certains suivant le même principe que le capteur CCD d’un appareil photo numérique.

Les capteurs en imagerie SAR, contrairement aux capteurs optiques, sont dit actifs, ce qui signifie qu’ils ont leur propre source d’émission. Ils peuvent ainsi réaliser des mesures jour et nuit et à travers un couvert nuageux. En effet, une image SAR est obtenue en émettant des ondes électromagnétiques puis en enregistrant l’écho rétro-diffusé, elle ne dépend donc pas de l’éclairement solaire. Les pixels d’une telle image sont des nombres complexes dont le module et l’argument représentent l’amplitude et la phase de l’onde rétro-diffusée respectivement. Le carré de l’amplitude fournit une information liée à la réflectivité de la surface, il s’agit de l’intensité.



FIGURE 2.2 – De gauche à droite, deux images satellitaires optiques en couleurs naturelles avant et après des inondations à Ayutthaya en Thaïlande en 2011. Elles ont été acquises par le satellite *Earth Observing-1* (EO-1) *Advanced Land Imager* (ALI) de la NASA.

Un pixel correspond à une zone au sol qu'on appelle la cellule de résolution. Sa dimension dépend des paramètres du satellite. La radiométrie d'un pixel correspond donc à la réponse au rayonnement électromagnétique du radar dans cette cellule, qui est fonction des paramètres du radar en lui-même et de la nature du sol (humidité, rugosité, inclinaison... etc) [Tupin *et al.* (2014)].

Les images radar sont le résultat de la sommation cohérente d'ondes rétro-diffusées (on parle d'imagerie cohérente). Ceci implique qu'au sein d'une cellule de résolution, les ondes renvoyées par des diffuseurs interfèrent et provoquent des fluctuations importantes de l'intensité mesurée, ce qui produit un phénomène de chatoiement, ou speckle en anglais, comme on le constate sur la figure 2.3.

D'après le modèle de Goodman [Goodman (1976)], dans une zone homogène, l'intensité suit une loi exponentielle et l'amplitude suit une loi de Rayleigh. Il est possible d'atténuer le speckle en créant des images SAR dites multi-vues. Pour cela, on dégrade la résolution d'une image radar en calculant pour chaque pixel la moyenne de ses L pixels voisins dont les valeurs sont considérées comme indépendantes et identiquement distribuées. Le nombre L est appelé nombre de vues. On peut aussi calculer la moyenne des voisins temporels lorsqu'on dispose d'images multitemporelles, c'est à dire d'une pile d'images de la même zone à différentes dates. Nous décrirons les distributions suivies par les données en intensité et en amplitude des images multi-vues dans la section suivante.

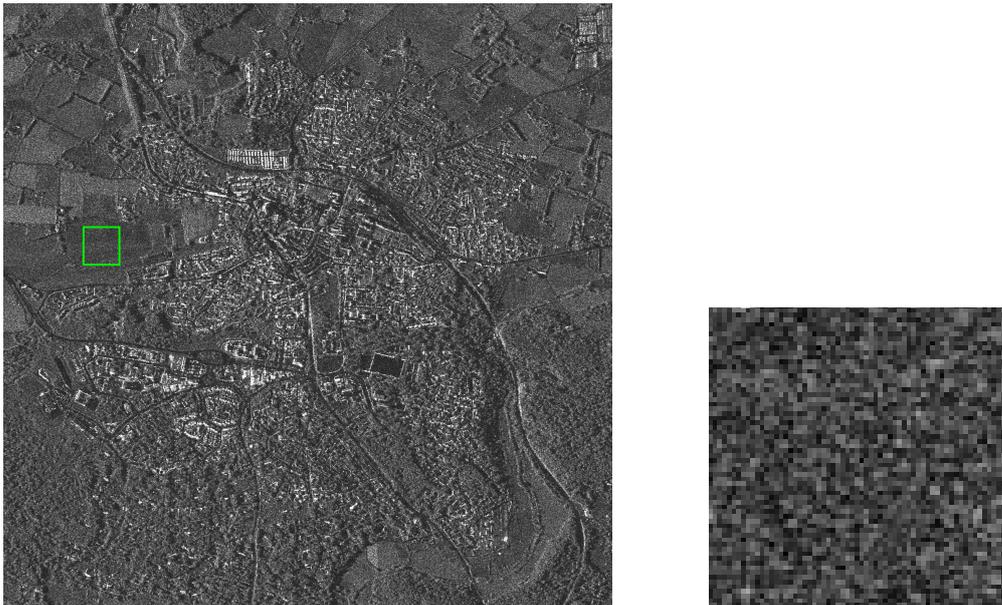


FIGURE 2.3 – De gauche à droite : l'image SAR du satellite TerraSAR-X de la ville de Lambesc et un zoom de la zone dans le carré vert où on observe du speckle. On observe que beaucoup de zones sont sombres et qu'il y a quelques structures et points très brillants. On le constate à travers la figure 2.4.

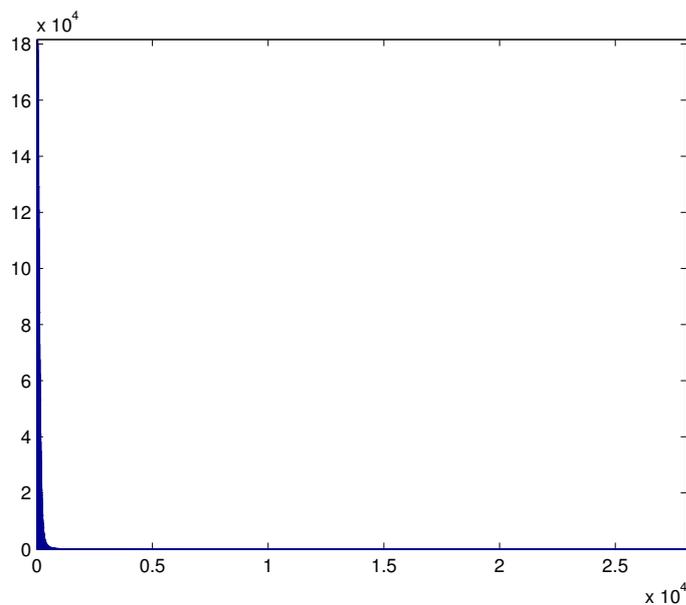


FIGURE 2.4 – Histogramme de l'image SAR de Lambesc. On observe que la dynamique des images SAR est grande car elles comptent beaucoup de zones sombres et quelques structures très brillantes.

2.1.2 Modèles de bruit

Le paragraphe 2.1.1 a mis en évidence la diversité des systèmes d'acquisition des images. Ces systèmes doivent être pris en compte si on souhaite traiter les images convenablement. En particulier, l'étude du système d'acquisition permet dans la plupart des cas de déterminer le type de bruit qui affecte l'image. Ceci permet ensuite de construire ensuite des traitements efficaces qu'il s'agisse de débruitage, de classification, de segmentation.

Nous modélisons une image bruitée, notée \mathbf{y} , par une réalisation d'un champ de N variables aléatoires. Nous notons \mathbf{x} l'image sans bruit et \mathbf{b} le bruit. Nous décrirons différents modèles de \mathbf{b}_i où les \mathbf{b}_i sont supposés indépendants. En particulier, nous citerons le bruit de Poisson, le bruit Gaussien et le speckle. Le bruit de Poisson et le bruit Gaussien sont souvent rencontrés en photographie numérique et avec certains capteurs d'images IR. Le speckle est observé sur des images médicales ultrasonores et sur des images SAR. Il existe d'autres types de bruit rencontrés en imagerie tels que le bruit Ricien, Chi-2 (χ^2) et impulsionnel qu'on ne décrira pas dans ce manuscrit.

Le bruit additif Gaussien Le bruit Gaussien est très souvent rencontré sous sa forme additive, ce qui permet de modéliser l'image bruitée ainsi :

$$\mathbf{y} = \mathbf{x} + \mathbf{b} \quad (2.1)$$

où \mathbf{b} représente le bruit et \mathbf{y} et \mathbf{x} sont des images à valeurs réelles.

On dit qu'il s'agit d'un bruit blanc lorsque la réalisation du bruit en un pixel est indépendante de toutes les autres dans l'image. La distribution Gaussienne de ce bruit permet d'écrire la distribution conditionnelle suivante :

$$p_{\mathcal{N}}(y_j|x_j) = \left(\frac{1}{\sqrt{2\pi}\sigma} \right) \exp \left(\frac{-(x_j - y_j)^2}{2\sigma^2} \right) \quad (2.2)$$

où σ est l'écart-type du bruit ; x_j et y_j , $j \in \{1, \dots, N\}$, dénotent le j -ème pixel des images \mathbf{x} et \mathbf{y} respectivement et $p_{\mathcal{N}}(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^N p_{\mathcal{N}}(y_j|x_j)$.

La figure 2.5 donne un exemple d'une image sans bruit et de deux versions bruitées avec un bruit Gaussien de différents écarts-types permettant d'observer les dégradations provoquées par le bruit.

Le bruit de Poisson Le bruit de Poisson est un bruit de comptage de photons dans des dispositifs optiques. Un exemple de simulation numérique d'un bruit de Poisson est présenté dans la figure 2.5. La distribution de ce bruit conduit à la distribution conditionnelle



FIGURE 2.5 – De gauche à droite : exemple de l'image sans bruit de Lena, deux versions d'elle bruitées avec un bruit Gaussien : l'une avec un écart-type $\sigma = 20$, l'autre avec un écart-type $\sigma = 40$ et une version bruitée avec un bruit de Poisson.

suivante :

$$p_{\mathcal{P}}(y_j|x_j) = \frac{x_j^{y_j} \exp(-x_j)}{y_j!} \quad (2.3)$$

où $p_{\mathcal{P}}(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^N p_{\mathcal{P}}(y_j|x_j)$ avec \mathbf{y} à valeurs entières naturelles et \mathbf{x} à valeurs réelles positives.

Notons que pour certains types d'images le bruit peut être modélisé par la somme d'un bruit Gaussien et de Poisson ou d'un mélange entre bruit Gaussien et bruit impulsif [Delon et Desolneux (2013)].

Le speckle L'amplitude des images radar souffre du phénomène du speckle. Celui-ci est modélisé comme un bruit de nature multiplicative et dépendant du signal, d'où la modélisation de l'image radar par : $y_j = b_j x_j$ où \mathbf{y}, \mathbf{x} et $\mathbf{b} \in \mathbb{R}_+^N$. En amplitude, le speckle suivant une loi de Nakagami-Rayleigh, on obtient la distribution conditionnelle des données suivantes :

$$p_{\mathcal{NR}}(y_j|x_j) = \frac{2L^L}{\Gamma(L)x_j^{2L}} y_j^{2L-1} \exp\left(-L \frac{y_j^2}{x_j^2}\right) \quad (2.4)$$

où L correspond au nombre de vues et Γ est la fonction gamma. Cette distribution est à asymétrie. En notant Y_j une variable aléatoire de loi Nakagami-Rayleigh, son espérance et sa variance sont : $\mathbb{E}(Y_j) = \frac{\Gamma(L+0.5)}{\sqrt{L\Gamma(L)}} x_j$, $\text{Var}(Y_j) = \left(1 - \frac{\Gamma(L+0.5)}{\sqrt{L\Gamma(L)}}\right) x_j^2$ respectivement.

En intensité le speckle suit une loi gamma qui correspond à la loi du carré d'une variable aléatoire suivant une loi de Nakagami-Rayleigh :

$$p_{\mathcal{G}}(y_j|x_j) = \frac{L^L y_j^{L-1}}{x_j^L \Gamma(L)} \exp\left(-L \frac{y_j}{x_j}\right) \quad (2.5)$$

C'est aussi une distribution asymétrique et en notant Y_j une variable aléatoire de loi gamma associée à y_j , son espérance et sa variance sont : $\mathbb{E}(Y_j) = x_j$ et $\text{Var}(Y_j) = \mathbb{E}(Y_j)^2/L = x_j^2/L$. La figure 2.6 illustre cette distribution pour différentes valeurs de

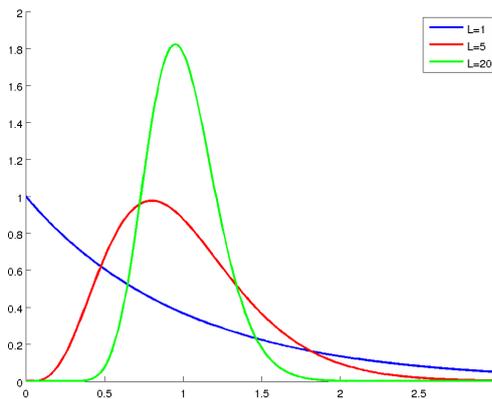


FIGURE 2.6 – Densité de probabilité de la loi gamma pour différentes valeurs de L

L . Plus L est grand plus on se rapproche d'une distribution Gaussienne de moyenne x_j et d'écart-type x_j/\sqrt{L} et meilleur est le rapport signal à bruit.

2.2 Bref état de l'art des méthodes de débruitage

Les modèles présentés dans cette thèse sont dans la majorité des cas appliqués au débruitage d'images. C'est souvent une étape nécessaire avant de pouvoir réaliser des opérations de traitement d'images de plus haut niveau. Dans cette section, nous présentons un bref historique de méthodes de débruitage adaptées au bruit Gaussien. L'image idéale qu'on souhaite retrouver est \mathbf{x} , sa version bruitée est \mathbf{y} et l'estimation de l'image idéale est notée $\hat{\mathbf{x}}$.

Filtrage linéaire La façon la plus intuitive de débruiter un pixel d'une image est de le remplacer par une moyenne de ses pixels voisins contenus dans une fenêtre glissante carrée, ce qui va atténuer le bruit. C'est le principe du filtre moyen ou boxcar filter. On peut choisir à la place d'une fenêtre glissante un noyau Gaussien circulaire et symétrique afin de calculer une moyenne pondérée des pixels voisins du pixel d'intérêt :

$$\hat{x}_i = \frac{\sum_j \omega(i, j) y_j}{\sum_j \omega(i, j)}, \text{ où } : \omega(i, j) = \exp\left(-\frac{\|p_j - p_i\|^2}{2h^2}\right) \quad (2.6)$$

où \hat{x}_i est la valeur du i -ème pixel de l'image débruitée \mathbf{x} , y_j est la valeur du j -ème pixel de l'image bruitée \mathbf{y} , p_i les coordonnées du pixel i et h l'écart-type du noyau Gaussien.

Une autre formulation de ce filtre est obtenue en considérant que l'image débruitée est

solution de l'équation de la chaleur :

$$\begin{cases} \frac{\partial x(u,t)}{\partial t} = \operatorname{div}(\nabla x(u,t)) \\ x(u,0) = y(u) \end{cases} \quad (2.7)$$

En effet, on peut résoudre numériquement cette équation avec un schéma itératif aux différences finies. Analytiquement, en calculant la transformée de Fourier de $x(u, \cdot)$, on aboutit à la solution correspondant au filtre avec un noyau Gaussien de l'équation (2.6).

Filtrage préservant les contours Le filtrage par noyau Gaussien présenté ci-dessus suppose que l'image sans bruit est sans discontinuités. En effet, la solution de l'équation de la chaleur est lisse et régulière. Cette hypothèse n'est pas vérifiée au niveau d'un contour ou d'une zone texturée. Dans ce contexte, une bonne alternative au filtre moyen est le filtre de Yaroslavsky [Yaroslavsky (1985)] qui sélectionne les pixels voisins ayant une intensité proche du pixel d'intérêt avant de calculer leur moyenne. Une autre alternative à l'approche basée sur l'équation de la chaleur est d'utiliser la diffusion anisotrope [Perona et Malik (1990)] :

$$\begin{cases} \frac{\partial x(u,t)}{\partial t} = \operatorname{div}(g(|\nabla x(u,t)|)\nabla x(u,t)) \\ x(u,0) = y(u) \end{cases} \quad (2.8)$$

où g est une fonction décroissante. On peut choisir par exemple, $g(u) = \exp(-u^2/h^2)$ avec le paramètre h contrôlant l'isotropie : plus h est grand plus la diffusion est isotrope.

Le principe de cette approche est que plus le gradient est faible, plus la zone est homogène et on autorise une diffusion plus isotrope, a contrario, plus le gradient est élevé, plus il est probable d'être au niveau d'un contour et la diffusion devient nulle dans la direction orthogonale au contour. La variation totale, ou Total Variation (TV) en anglais [Rudin et al. (1992)] est un a priori qui permet également de lisser les zones homogènes en préservant les contours d'une image. Cette approche sera décrite plus en détails en section 2.4.1.

Filtrage préservant les textures Pour préserver les détails d'une image et ses textures lors de son débruitage, il est possible d'utiliser des approches par dictionnaires. Ces derniers sont constitués d'éléments dits atomes pouvant représenter des cibles, des contours, des zones homogènes et des textures. Le principe est d'approcher l'image par une combinaison linéaire parcimonieuse d'atomes :

$$\min_{\mathbf{u}} \|\mathbf{x} - \mathbf{D}\mathbf{u}\|_2^2 \text{ tel que } \phi(\mathbf{u}) \text{ soit parcimonieux} \quad (2.9)$$

où \mathbf{x} est l'image à approcher de taille $N \times 1$, \mathbf{D} est le dictionnaire de taille $N \times K$ avec K le nombre d'atomes qu'il contient, \mathbf{u} est un vecteur creux de taille $K \times 1$ représentant

les coefficients parcimonieux de cette combinaison, ϕ est un terme de régularisation induisant la parcimonie.

Ces atomes peuvent former une base orthonormale, ex : une base de cosinus discrets, Discrete Cosine Transform (DCT), en anglais ou une base d'ondelettes. Chaque coefficient u_k s'exprime alors comme la projection de \mathbf{x} sur chaque atome tel que le décrit l'équation suivante :

$$u_k = \langle \mathbf{x}, \mathbf{d}_k \rangle \quad (2.10)$$

où u_k est le k -ème élément de \mathbf{u} , \mathbf{d}_k est le k -ème atome de \mathbf{D} .

Ces bases sont très utiles dans des applications de compression (standards : JPEG et JPEG2000) et également en débruitage. En effet, les coefficients les plus faibles sont considérés comme du bruit. En conséquence, seuiller les coefficients faibles supprime le bruit. On peut réaliser un seuillage dur (Hard-Tresholding, HT) des coefficients [Mallat (2008)] lorsque le terme de l'équation (2.9) pour induire la parcimonie est $\phi(\mathbf{u}) = \|\mathbf{u}\|_0$ (nombre d'éléments non nuls dans \mathbf{u}) :

$$\eta_{\text{HT}}(u_k) = \begin{cases} u_k & \text{si } |u_k| > \gamma \\ 0 & \text{sinon} \end{cases} \quad (2.11)$$

ou un seuillage doux (Soft-Tresholding, ST) [Donoho et Johnstone (1994)] lorsque $\phi(\mathbf{u}) = \|\mathbf{u}\|_1$:

$$\eta_{\text{ST}}(u_k) = \text{sign}(u_k) \max(0, |u_k| - \gamma) \quad (2.12)$$

Les ondelettes ne s'adaptant pas à tous les types de structures dans les images, d'autres bases s'en inspirant ont été développées par la suite, par exemple : les curvelets [Starck *et al.* (2002)] et les bandelets [Le Pennec et Mallat (2005)] qui sont adaptées à des contours arrondis. Néanmoins, une méthode de débruitage parmi les plus célèbres basées sur les ondelettes est celle de [Portilla *et al.* (2003)] où on modélise la dépendance entre les coefficients d'ondelettes par un modèle de mélange de Gaussiennes. Nous discuterons dans la section 2.4.2 de modèles parcimonieux par patches (petites imagettes) où le dictionnaire n'est pas prédéfini.

Ce sont d'ailleurs les méthodes dites non-locales utilisant des patches qui ont détrôné les approches basées sur les ondelettes notamment avec l'algorithme NL-Means [Buades *et al.* (2005)] où le pixel central de chaque patch de l'image bruitée est traité en calculant une moyenne pondérée des pixels centraux des patches qui lui sont similaires. Nous décrirons ces approches en section 2.3.1.

D'autres méthodes par patches sont très répandues actuellement et modélisent la distribution des patches par des modèles Gaussiens [Zoran et Weiss (2011)], [Yu *et al.*

(2012)], [Lebrun *et al.* (2013)]. Elles présentent parmi les meilleures performances en débruitage et feront l'objet des paragraphes 2.3.2 et 2.4.3.

Nous avons donc vu des méthodes qui débruitent l'image d'intérêt pixel par pixel (ex : filtre moyen), des méthodes qui débruitent l'image entière (ex : avec les ondelettes) et des méthodes toujours très utilisées actuellement basées sur des patches. Dans la suite de ce chapitre, nous décrirons divers modèles basés sur des patches. Plus précisément, nous décrirons des a priori utilisés notamment dans des méthodes de débruitage tenant compte soit de la distribution locale de l'espace des patches soit de la distribution de l'espace global des patches. Ceci permettra de distinguer les méthodes construisant un a priori grâce aux patches de l'image d'intérêt (modélisation locale) ou les méthodes construisant un a priori avec un nombre assez grand de patches pour qu'on considère qu'il s'agisse d'une modélisation globale de l'espace des patches.

2.3 Modélisation locale de l'espace des patches

Cette section décrit des méthodes de débruitage modélisant localement l'espace des patches. En section 2.3.1, nous traiterons les méthodes non-locales qui ont constitué les premières méthodes de débruitage basées sur des patches au début des années 2000. En section 2.3.2, nous traiterons les méthodes modélisant localement la distribution des patches avec des modèles Gaussiens.

2.3.1 Premiers modèles basés sur la redondance des patches d'une image

Les images naturelles vérifient généralement une propriété dite "d'autosimilarité" dans la littérature. En effet, si on décompose une image en plusieurs patches, nombreux sont ceux qui sont très similaires comme il est illustré en figure 2.7. La méthode de synthèse de texture proposée dans [Efros *et al.* (1999)] figure parmi les premiers travaux à exploiter cette propriété.

Cela a grandement influencé les méthodes de débruitage de l'époque et a notamment permis l'essor des méthodes non-locales qui ont surpassé les méthodes basées sur les ondelettes comme [Portilla *et al.* (2003)]. Par exemple, l'algorithme NL-Means, pour Non-Local Means (moyennes non-locales en français), a été proposé par plusieurs équipes : [Buades *et al.* (2005)], [Awate *et al.* (2005)], avec quelques différences mais c'est celui de [Buades *et al.* (2005)] qui a été le plus popularisé. À l'opposé du filtre moyen qui suppose que les voisins d'un pixel sont ceux qui ont le plus de chance de lui ressembler, ce qui n'est vrai que dans une région homogène, l'algorithme NL-means recherche des patches similaires au patch d'intérêt dans une fenêtre de recherche (de taille 40×40 par exemple), ce qui est moins restrictif que le voisinage de petite taille du filtre moyen et qui tire profit

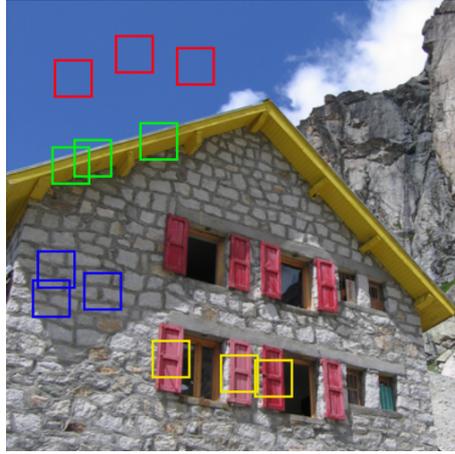


FIGURE 2.7 – Illustration de l'autosimilarité dans les images naturelles. En effet, de nombreux patchs extraits à différents endroits de l'image se ressemblent.

de l'autosimilarité des images. Ensuite, le pixel central du patch d'intérêt est débruité en calculant la moyenne des pixels centraux des patchs similaires. Un pixel y_i est débruité par les moyennes non-locales comme suit :

$$\hat{x}_i = \frac{1}{Z} \sum_{j=1}^{N_S} \omega(i, j) y_j \quad (2.13)$$

où \hat{x}_i est l'estimation sans bruit du pixel central du patch d'intérêt, y_j est un pixel bruité, $Z = \sum_{j=1}^{N_S} \omega(i, j)$ est une constante de normalisation et N_S est le nombre de pixels dans la fenêtre de recherche. Les poids ω de cette moyenne sont définis comme suit :

$$\omega(i, j) = \exp\left(-\frac{1}{h^2} \|\mathbf{y}_i - \mathbf{y}_j\|_2^2\right) \quad (2.14)$$

où \mathbf{y}_i est un patch dont le pixel central est i et h^2 un paramètre de régularisation.

Un autre exemple de méthode de débruitage non-local donnant toujours à l'heure actuelle parmi les meilleures performances est BM3D pour Block-Matching 3D [Dabov *et al.* (2007)]. Comme pour les NL-means, son principe réside dans la recherche de patchs similaires. Cependant, dans cette approche, on ne débruitera pas uniquement les pixels centraux mais les patchs entiers. Dans une première étape, on réalise un clustering des patchs de l'image d'intérêt, puis on applique une transformée DCT 3D à chaque pile de patchs appartenant à un même cluster. Les coefficients de la DCT de chaque pile sont ensuite seuillés avant d'appliquer la transformée inverse, ce qui permet de débruiter tous les patchs du cluster, selon le principe du filtrage collaboratif. Dans une seconde étape, on refait la même chose que dans la première étape mais en réalisant le clustering sur l'image débruitée issue de la première étape. Cet algorithme est proche de l'algorithme Non-Local Bayes [Lebrun *et al.* (2013)] qu'on présentera en section 2.3.2. Tous les deux ont la même architecture à cela près que la façon de filtrer les clusters de patchs similaires



FIGURE 2.8 – De gauche à droite : une image sans bruit, bruitée avec $\sigma = 25$ et débruitée par NL-Means [Buades et al. (2011)] où on observe le phénomène des patchs rares à travers un halo de bruit au niveau des contours très contrastés (en haut de la cheminée par exemple).

n'est pas la même. En effet, dans l'algorithme NL-Bayes on réalise le filtrage à l'aide de modèles Gaussiens construit à partir de patchs similaires.

Malgré leurs qualités, ces méthodes présentent quelques défauts. Par exemple les NL-means souffrent du phénomène des patchs rares. Il s'agit d'un résidu de bruit qui apparaît souvent au niveau de contours très contrastés où il est difficile de trouver des patchs similaires comme il est montré en figure 2.8. Certains travaux tentent de résoudre ce problème notamment en adaptant la forme des patchs [Deledalle et al. (2012b)] et en combinant le modèle non-local à la variation totale [Louchet et Moisan (2011)], [Sutour et al. (2014)]. Une autre possibilité est d'utiliser une approche par dictionnaire de patchs où ce dernier pourrait contenir des structures rares dans l'image.

2.3.2 Modélisation locale de l'espace des patchs avec des modèles Gaussiens

Les approches modélisant localement la distribution des patchs par des modèles Gaussiens sont très répandues dans des applications au débruitage ces dernières années. Par exemple, le principe des approches proposées par [Deledalle et al. (2011)] et [Zhang et al. (2010)] réside dans le calcul des ACP (Analyses en Composantes Principales) de groupes de patchs bruités. Dans [Deledalle et al. (2011)], les patchs sont extraits d'une même région, tandis que dans [Zhang et al. (2010)] on regroupe des patchs similaires (voir illustration 2.9). Ensuite, un seuillage est appliqué aux coefficients dans l'approche de [Deledalle et al. (2011)] tandis que des [Zhang et al. (2010)] on utilise un filtre de Wiener afin de supprimer le bruit. Contrairement à une approche plus naïve où on calculerait une ACP sur tous les patchs de l'image et où on supprimerait les axes de faible variance, l'intérêt de cette approche est de supposer que le bruit se répand uniformément sur tous les axes. Ainsi, utiliser un seuillage ou un filtre de Wiener permet de mieux conserver des structures rares dans l'image [Deledalle et al. (2011)]. L'équation suivante décrit le

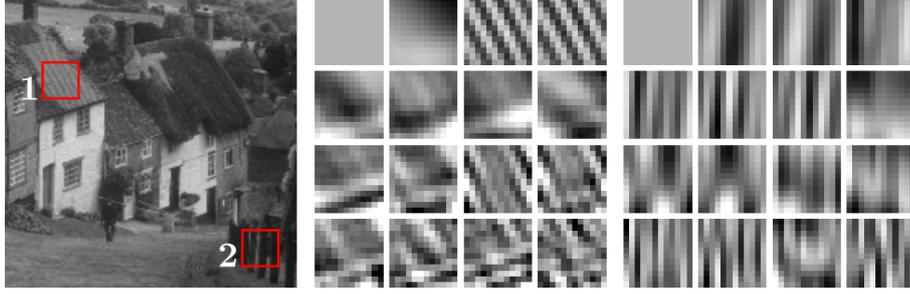


FIGURE 2.9 – De gauche à droite : une image dont on extrait des patches de deux zones, les vecteurs propres associés aux plus fortes valeurs propres de la zone 1, les vecteurs propres associés aux plus fortes valeurs propres de la zone 2. On observe que ces axes sont bien similaires aux structures observées dans les zones où les patches ont été extraits et qu'il est donc pertinent de réaliser des ACP sur des patches similaires [Deledalle et al. (2011)].

débruitage d'un patch \mathbf{y}_i par ce type de procédé :

$$\hat{\mathbf{x}}_i = \bar{\mathbf{y}} + \sum_{k=1}^K \eta(\langle \mathbf{y}_i - \bar{\mathbf{y}}, \mathbf{f}_k \rangle) \mathbf{f}_k \quad (2.15)$$

où $\hat{\mathbf{x}}_i$ est l'estimation sans bruit de \mathbf{y}_i , $\bar{\mathbf{y}}$ est la moyenne des patches \mathbf{y}_i appartenant à un même cluster de patches similaires, \mathbf{f}_k est le k -ème vecteur propre de la matrice de covariance empirique associée à l'ensemble de patches similaires, K est le nombre de vecteurs propres correspondant à la taille d'un patch mis sous forme de colonne et η est l'opération de seuillage ou de filtrage de Wiener.

Notons que dans [Zhang et al. (2010)], les auteurs ont choisi de débruiter de manière pixellique contrairement à [Deledalle et al. (2011)] où les patches débruités sont agrégés.

Les méthodes proposées dans [Lebrun et al. (2013)] correspondant à l'algorithme NL-Bayes (Non-Local Bayes) et dans [Yu et al. (2012)] correspondant au PLE (Piecewise Linear Estimator) sont très proches. Elles se décomposent en deux étapes : une première estimation de l'image sans bruit et une seconde étape améliorant cette estimation.

Dans le cas du PLE, la première estimation est obtenue grâce à des modèles Gaussiens appris sur des images synthétiques de contours de différentes orientations.

Dans le cas de NL-Bayes, la première estimation est obtenue à partir de l'image bruitée. Pour ce faire, on commence par créer K clusters de patches en regroupant les patches similaires à un patch d'intérêt. Ensuite, les auteurs proposent de détecter si le patch d'intérêt est homogène afin de le traiter séparément, ce qui améliore le rapport signal sur bruit, en particulier, dans le cas de forts niveaux de bruit.

On évalue si le patch d'intérêt est homogène en calculant la variance de l'ensemble des patches similaires à ce patch. Si cette dernière est inférieure à une quantité proportionnelle à la variance du bruit, alors le patch est considéré homogène et son estimation est une

moyenne des patchs qui lui sont similaires. Dans le cas contraire, on crée un modèle Gaussien à partir du groupe de patchs similaires à ce patch en calculant une matrice de covariance et une moyenne empiriques et on estime la version sans bruit du patch de la même façon que dans la seconde étape (cf. eq. (2.18)).

C'est dans les secondes étapes respectives du PLE et de l'algorithme NL-Bayes que ces deux approches diffèrent vraiment. Elles ont des architectures comparables à celle de l'algorithme EM. Dans l'étape E, chaque patch est débruité et associé à un modèle parmi K modèles Gaussiens. Dans l'étape M, chaque modèle est mis à jour en calculant les matrices de covariance empiriques à partir des patchs associés à chaque modèle $k \in \{1, \dots, K\}$. Nous expliquons en détails l'étape E pour les deux méthodes dans la suite.

Le problème de débruitage d'un patch s'écrit comme suit avec l'algorithme NL-Bayes :

$$\hat{\mathbf{x}}_i = \underset{\mathbf{x}_i}{\operatorname{argmin}} \|\mathbf{x}_i - \mathbf{y}_i\|_2^2 + \sigma^2 (\mathbf{x}_i - \boldsymbol{\mu})^t \mathbf{C}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) + \sigma^2 \log(\det(\mathbf{C})) \quad (2.16)$$

où σ^2 est la variance du bruit, $\{\boldsymbol{\mu}, \mathbf{C}\}$ sont respectivement le vecteur moyen et la matrice de covariance du modèle Gaussien associé au i -ème patch.

Le problème de débruitage s'écrit comme suit pour débruiter un patch avec le PLE :

$$\{\hat{\mathbf{x}}_i, \hat{k}\} = \underset{\mathbf{x}_i, k}{\operatorname{argmin}} \|\mathbf{x}_i - \mathbf{y}_i\|_2^2 + \sigma^2 \mathbf{x}_i^t \mathbf{C}_k^{-1} \mathbf{x}_i + \sigma^2 \log(\det(\mathbf{C}_k)) \quad (2.17)$$

Le problème (2.17) est résolu de façon itérative (3 itérations) en trouvant \hat{k} puis $\hat{\mathbf{x}}_i$, ce qui diffère par rapport à l'algorithme NL-Bayes. En effet, cela implique que dans le PLE, on cherche le meilleur modèle Gaussien parmi ceux appris à l'itération précédente tandis que l'algorithme NL-Bayes utilise directement le modèle Gaussien appris sur les patchs similaires au patch d'intérêt.

Les problèmes (2.16) et (2.17) sont résolus en choisissant \mathbf{x}_i qui annule le gradient, ce qui donne :

$$\hat{\mathbf{x}}_i = \boldsymbol{\mu}_k + \mathbf{C}_k (\mathbf{C}_k + \sigma^2 \mathbf{I})^{-1} (\mathbf{y}_i - \boldsymbol{\mu}_k) \quad (2.18)$$

Précisons que pour tous k , les moyennes $\boldsymbol{\mu}_k$ sont nulles dans le cas du PLE.

La résolution en k dans le problème (2.17) s'obtient par recherche exhaustive sur $\hat{k} \in \{1, \dots, K\}$ su problème suivant :

$$\hat{k} = \underset{k}{\operatorname{argmin}} \mathbf{x}_i^t \mathbf{C}_k^{-1} \mathbf{x}_i + \log(\det(\mathbf{C}_k)) \quad (2.19)$$



FIGURE 2.10 – De gauche à droite : image bruitée (8% de bruit), NL-Bayes adapté [Guillemot et al. (2014)] et NL-Bayes classique. On observe qu’une partie des briques est totalement lissée avec la version classique de l’algorithme, tandis que la version utilisant un GMM appris sur une base de patches et non sur l’image bruitée n’a pas ce défaut.

L’algorithme de débruitage de bruit Gaussien NL-Bayes est parmi les plus performants actuellement. Un résultat est présenté en figure 2.11 sur l’image de Barbara. Néanmoins, des zones d’images présentant des textures peu contrastées, une fois bruitées sont considérées à tort comme homogènes et sont trop lissées par l’algorithme. Les travaux de [Guillemot et al. (2014)] proposent une structure d’apprentissage rapide de GMM sur de larges bases de patches tirés d’images naturelles et adaptent l’algorithme NL-Bayes afin de réaliser un débruitage utilisant ces modèles. Cette adaptation avec une modélisation globale de l’espace des patches donne des résultats performants, même dans les zones où l’algorithme NL-Bayes est moins efficace comme l’illustre la figure 2.10. C’est le type de modélisation que nous décrivons dans la section suivante.

2.4 Modélisation globale de l’espace des patches

Dans cette section, nous décrivons quelques familles de méthodes représentant la distribution de l’espace global des patches appliquées au débruitage. Cela signifie que pour un type d’image, ces méthodes vont traiter une image en particulier grâce à une modélisation de la variété de l’espace global des patches de ce type d’imagerie. On peut citer en exemple de ces méthodes une approche qui dans un premier temps apprendrait des descripteurs sur une très large base de patches et qui les utiliserait dans un second temps afin de débruiter une image.

Nous commençons par aborder les approches par champs de Markov, sec. 2.4.1, puis par GMM (Gaussian Mixture Models) en section 2.4.3 qui s’expriment dans un cadre Bayésien décrit par l’équation suivante :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmax}} \ p(\mathbf{y}|\mathbf{x}) \ p(\mathbf{x}) \quad (2.20)$$

$$\Leftrightarrow \hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \ -\log p(\mathbf{y}|\mathbf{x}) - \log p(\mathbf{x}) \quad (2.21)$$

où l’attache aux données $-\log p(\mathbf{y}|\mathbf{x})$ s’écrit : $\|\mathbf{y} - \mathbf{x}\|_2^2/2\sigma^2 + C$, où C est une constante, dans le cas Gaussien et l’a priori $p(\mathbf{x})$ sera décrit pour chaque approche.



FIGURE 2.11 – Résultats de débruitage avec l'algorithme NL-Bayes sur Barbara. On observe des résidus de bruit sur le résultat de la première étape du débruitage qui disparaissent après la seconde étape, ce qui illustre l'intérêt de cette architecture de l'algorithme.

2.4.1 Des champs de Markov aux champs d'experts

Supposons qu'une image soit la réalisation d'un champ aléatoire sur un graphe. Ses pixels forment les sites du graphe auxquels on peut associer une variable aléatoire. Un voisinage d'un site est un ensemble de sites étant connectés. On peut définir aussi des cliques, dont l'ensemble est noté \mathcal{C} , qui correspondent à un singleton ou des ensembles de sites tous voisins les uns des autres. Ce champ est dit de Markov si et seulement si la probabilité conditionnelle locale en un site n'est fonction que de la configuration du voisinage du site considéré.

Le principe des approches par champs de Markov appliquées au traitement d'images est généralement de chercher l'image de paramètres $\hat{\mathbf{x}}$ qui maximise la distribution a posteriori comme représenté dans l'équation (2.20). On montre par le théorème d'Hammersley-Clifford que, pour un champ Markovien, l'a priori p s'écrit comme une distribution de Gibbs :

$$p(\mathbf{x}) = \frac{1}{Z} \exp \left(- \sum_{c \in \mathcal{C}} V_c(\mathbf{x}_{(c)}) \right) \quad (2.22)$$

où Z est une constante de normalisation et $V_c(\mathbf{x}_{(c)})$ est le potentiel d'une clique $\mathbf{x}_{(c)}$.

De nombreuses méthodes considèrent des cliques formées par des paires de pixels voisins (cliques d'ordre deux). On peut trouver le minimum global de ce type d'énergie par coupures de graphes (graph-cuts) pour certaines classes de fonctions comme il est montré dans [Ishikawa (2003)].

L'a priori dit de variation totale, Total Variation (TV) en anglais, ou encore modèle de ROF (pour Rudin-Osher-Fatemi) [Rudin *et al.* (1992)], peut être vu comme une approche considérant des cliques d'ordre deux et se résume par l'équation suivante :

$$\hat{\mathbf{x}} = \operatorname{argmin}_{\mathbf{x}} \frac{1}{\sigma^2} \|\mathbf{y} - \mathbf{x}\|_2^2 + \lambda \sum_i |\nabla_i \mathbf{x}| \quad (2.23)$$

où λ est un paramètre contrôlant la force du terme de régularisation de sorte que plus λ est grand plus l'image en sortie \mathbf{x} tend vers une image constante. En effet, le terme $\sum_i |\nabla_i \mathbf{x}|$ s'obtient en réalisant des différences finies sur un grille orientée horizontalement et verticalement et pousse à approcher le signal d'entrée par un signal constant par morceaux. Ceci revient à une parcimonie (grâce à la norme ℓ_1) sur les différences calculées et donc à une approche parcimonieuse en analyse. Ce type d'approche est introduit en section 2.4.2.1.

En résumé, l'a priori TV a donc pour avantage de préserver les contours mais il atténue des détails dans les images et entraîne un phénomène de staircasing, comme on peut l'observer sur la figure 2.12. De plus, il atténue également les textures, et ceci est illustré dans des exemples de décompositions cartoon-textures proposés dans [Atto et Mercier



FIGURE 2.12 – De gauche à droite : l'image sans bruit, bruitée ($\sigma = 20$) et l'image débruitée avec la régularisation TV où on observe l'effet "marche d'escalier", ou *staircasing effect*, par exemple au niveau du nez.

(2015)]. Certains travaux visent à corriger ces défauts, par exemple : l'a priori TGV (Total Generalized Variation) [Bredies *et al.* (2010)] et [Louchet et Moisan (2014)].

On peut également utiliser dans les approches par MRF des cliques de plus grande taille, ce qui constitue finalement des patches. Dans ce contexte, nous décrivons dans la suite de ce paragraphe une méthode de référence dite des champs d'experts (FoE pour Fields of Experts en anglais) [Roth et Black (2009)] qui a notamment été appliquée au débruitage et à l'inpainting.

Les auteurs modélisent un a priori sur une image avec un ensemble de filtres linéaires et un potentiel. En effet, en supposant qu'un voisinage dans le graphe représentant l'image soit sous la forme d'un carré de taille $w \times w$ et qu'il soit centré sur le site i , en définissant la configuration d'une clique maximale que l'on note \mathbf{x}_i , l'a priori proposé s'écrit comme suit :

$$p(\mathbf{x}) = \frac{1}{Z(\Theta)} \exp \left(\sum_{i=1}^N \sum_{k=1}^K \log \phi(\mathbf{J}_k^t \mathbf{x}_i; \gamma_k) \right) = \frac{1}{Z(\Theta)} \prod_{i=1}^N \prod_{k=1}^K \phi(\mathbf{J}_k^t \mathbf{x}_i; \gamma_k) \quad (2.24)$$

où $V_c(\mathbf{x}_i) = -\sum_{k=1}^K \log \phi(\mathbf{J}_k^t \mathbf{x}_i; \gamma_k)$ par rapport à la formulation (2.22). Chaque \mathbf{J}_k est un filtre linéaire appelé expert et γ_k est un paramètre associé au k -ème expert. $\Theta = \{\mathbf{J}_k, \gamma_k | k = 1, \dots, K\}$ représente les paramètres définissant l'ensemble des experts et $Z(\Theta)$ est une fonction de normalisation.

Il est intéressant de pointer, comme il est précisé dans [Chen *et al.* (2014)], que les champs d'experts peuvent être vus comme un modèle parcimonieux en analyse (cf. sec. 2.4.2.1) sur des patches étendu à l'image entière pour tenir compte des patches qui se superposent au moment de les recombinaison après les avoir restaurés.

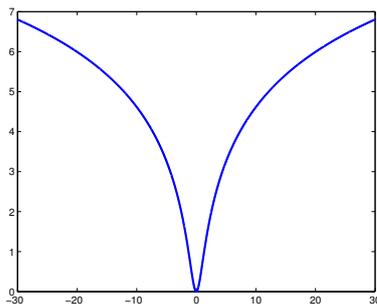


FIGURE 2.13 – Tracé de la fonction définie par : $x \mapsto \log(1 + x^2)$, à mettre en relation avec l'équation (2.26) pour $\gamma_k = 1$.

Souvent la distribution utilisée pour modéliser les experts est la loi t-student :

$$\phi(\mathbf{J}_k^t \mathbf{x}_i; \gamma_k) = \left(1 + \frac{1}{2}(\mathbf{J}_k^t \mathbf{x}_i)^2\right)^{-\gamma_k} \quad (2.25)$$

Ainsi par rapport à la formulation (2.21) :

$$-\log p(\mathbf{x}) = \sum_{i=1}^N \sum_{k=1}^K \gamma_k \log(1 + (\mathbf{J}_k^t \mathbf{x}_i)^2) + \log(Z(\Theta)) \quad (2.26)$$

avec $\log(Z(\Theta))$ une constante ne dépendant pas de \mathbf{x} dont on ne tient pas compte lors de l'optimisation.

Le choix de ce type de distribution pour modéliser des images naturelles est justifié dans plusieurs articles, notamment dans [Weiss et Freeman (2007)] où est décrite une expérience simple qui permet de remarquer que les log-histogrammes de dérivées à différentes échelles d'une image naturelle sont représentés par une distribution proche du logarithme d'une loi t-student illustrée en figure 2.13. De plus, ces distributions pénalisant peu les valeurs fortes favorisent la parcimonie.

Les paramètres $\gamma_k \geq 0$ et les filtres linéaires \mathbf{J}_k sont appris hors-ligne sur une base d'images naturelles en maximisant la vraisemblance de ce produit d'experts. Pour cela, les auteurs utilisent des méthodes de Monte-Carlo gourmandes en temps de calcul. La figure 2.14 présente des filtres appris par les FoE présentant des contours et des textures à différentes échelles et orientations.

Dans leur application au débruitage, les auteurs utilisent une descente de gradient pour résoudre le problème de maximisation de la probabilité a posteriori (cf. éq.(2.20)). Les performances de cette méthode sont très satisfaisantes mais surpassées par certaines méthodes par patches plus récentes.

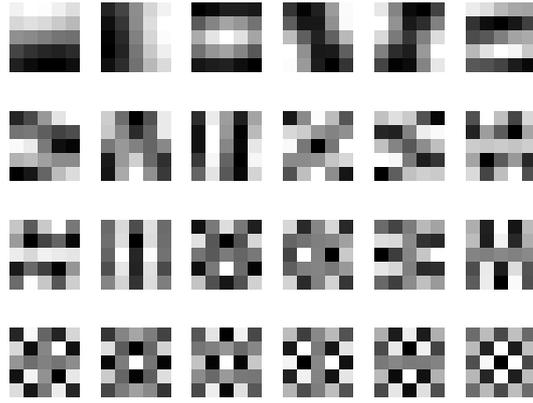


FIGURE 2.14 – Exemple de filtres de taille 5×5 appris dans [Roth et Black (2009)].

2.4.2 Modèles parcimonieux basés sur les patches

2.4.2.1 Principe : Synthèse VS analyse

Le principe de la parcimonie basée sur les patches avec un point de vue de type synthèse réside dans la représentation de patches par une combinaison linéaire d'éléments d'une base dont les coefficients sont pour la plupart nuls, donc parcimonieux ou creux. Pour cela, on peut utiliser des bases d'ondelettes ou de cosinus discret (DCT pour Discrete Cosine Transform) comme nous l'avons mentionné au paragraphe 2.2. Cependant, on peut également utiliser des dictionnaires qui sont constitués d'un ensemble de patches nommés atomes. L'équation suivante formule le problème d'approximation d'un patch par la combinaison linéaire parcimonieuse d'atomes, qui est le problème dit du sparse-coding :

$$\mathbf{u}_i^* = \underset{\mathbf{u}_i}{\operatorname{argmin}} \|\mathbf{D}\mathbf{u}_i - \mathbf{y}_i\|_2^2 + \lambda\phi(\mathbf{u}_i), \quad \mathbf{x}_i = \mathbf{D}\mathbf{u}_i^* \quad (2.27)$$

où \mathbf{D} est un dictionnaire de taille $w^2 \times K$, avec w^2 la taille d'un patch en forme de colonne et K le nombre d'atomes. Le vecteur de coefficients \mathbf{u}_i est de taille $K \times 1$ et chacune de ses entrées donne le poids de chaque atome de \mathbf{D} pour représenter le patch \mathbf{x}_i .

La fonction ϕ est un terme de régularisation induisant la parcimonie et il en existe de nombreuses. Par exemple, la pseudo-norme ℓ_0 , où $\phi(\mathbf{u}_i) = \|\mathbf{u}_i\|_0$, définit le degré de parcimonie du vecteur \mathbf{u}_i , c'est-à-dire le nombre d'éléments non nuls qu'il contient. Cette fonction est discontinue, non convexe et rend le problème (2.27) NP-difficile quand \mathbf{D} n'est pas orthogonale. Il peut être résolu de façon sous-optimale par des méthodes gloutonnes (greedy en anglais) comme le Matching Pursuit (MP) [Mallat et Zhang (1993)] ou l'Orthogonal Matching Pursuit (OMP) [Pati et al. (1993)]. Le principe du MP est de chercher itérativement l'atome du dictionnaire le plus corrélé aux résidus courants qui correspondent à la différence entre le patch et son approximation. La norme ℓ_1 est la relaxation convexe de la pseudo-norme ℓ_0 , ce qui rend le problème d'optimisation à résoudre plus simple. Plusieurs algorithmes ont été proposés à cet effet tel que le Basis-Pursuit [Chen et al. (1998)].

Les dictionnaires adaptatifs sont souvent appris sur une base de patches tirés d'images naturelles ou à partir de l'image à traiter. Le nombre d'atomes d'un dictionnaire est le plus souvent plus grand que w^2 , la largeur d'un patch au carré (dictionnaire redondant ou overcomplete en Anglais). Ce nombre est fixé par avance mais il existe des méthodes d'apprentissage de dictionnaires avec une taille adaptative, comme par exemple [Dang et Chainais (2015)]. Les approches basées sur l'apprentissage de dictionnaires sont efficaces dans différentes applications comme la compression, la classification ou encore le débruitage. Leur avantage est qu'elles s'adaptent aux données traitées contrairement aux dictionnaires paramétriques tels que les bases d'ondelettes.

Un autre point de vue pour introduire la parcimonie est celui de l'analyse, on parle alors de co-parcimonie qu'on modélise à travers le problème suivant :

$$\operatorname{argmin}_{\mathbf{x}_i} \|\mathbf{x}_i - \mathbf{y}_i\|_2^2 + \lambda \phi(\mathbf{A}\mathbf{x}_i) \quad (2.28)$$

où \mathbf{A} est un opérateur d'analyse, \mathbf{y}_i est l'approximation de \mathbf{x}_i et ϕ est un terme de régularisation induisant la parcimonie.

L'apprentissage d'opérateurs d'analyse a été moins développé que l'apprentissage de dictionnaire avec un point de vue de type synthèse. Cependant, il connaît un regain d'intérêt ces dernières années, ex : [Rubinstein *et al.* (2013)], [Chen *et al.* (2014)]. La principale différence entre une approche de type synthèse et une approche de type analyse est que la première a pour objectif de calculer \mathbf{u}_i parcimonieux et \mathbf{D} synthétisant l'image d'intérêt, tandis que la seconde veille à ce que l'approximation de l'image d'intérêt soit parcimonieuse à travers \mathbf{A} . Les deux approches coïncident si le dictionnaire \mathbf{D} en synthèse est inversible (les problèmes (2.27) et (2.28) sont alors équivalents). Notons que l'a priori TV et l'a priori FoE (Fields of Experts) sont parcimonieux en analyse comme il a été précisé au paragraphe 2.4.1.

2.4.2.2 Quelques méthodes d'apprentissage de dictionnaires

Ce paragraphe décrit quelques méthodes de référence basées sur des modèles parcimonieux introduits en section 2.4.2.1. Plus précisément, nous décrirons des méthodes d'apprentissage de dictionnaires par patches. À la fin des années 90, [Olshausen *et al.* (1996)] proposent une méthode d'apprentissage en deux étapes : le codage parcimonieux des patches de l'image d'intérêt dans un dictionnaire et la mise à jour de ce dictionnaire avec les coefficients calculés à l'étape précédente.

$$\{\widehat{\mathbf{U}}, \widehat{\mathbf{D}}\} = \operatorname{argmin}_{\mathbf{D}, \mathbf{U}} \|\mathbf{D}\mathbf{U} - \mathbf{X}\|_F^2 + \lambda \sum_i \log(1 + \mathbf{u}_i^2) \quad (2.29)$$

où \mathbf{X} de taille $w^2 \times N$ est une matrice constituée de patches extraits d'une ou plusieurs images mis sous forme de colonnes \mathbf{x}_i , \mathbf{D} de taille $w^2 \times K$ est un dictionnaire dont les K atomes sont sous forme de colonnes, \mathbf{U} de taille $K \times N$ est la matrice dont les colonnes notées \mathbf{u}_i représentent le poids de chaque atome dans la représentation parcimonieuse d'un patch d'indice i , $\|\mathbf{DU} - \mathbf{X}\|_F^2 = \sum_i \|\mathbf{D}\mathbf{u}_i - \mathbf{x}_i\|_2^2$ et λ un paramètre de régularisation.

Le terme induisant la parcimonie choisi par les auteurs est $\phi(x) = \log(1 + x^2)$ qui est différentiable, ce qui permet d'obtenir une formule analytique de la mise à jour des \mathbf{u}_i en mettant son gradient à zéro. La mise à jour du dictionnaire se fait par descente de gradient. Les dictionnaires obtenus représentant de façon efficace les données de la base d'apprentissage, ce type d'approche connaît un grand succès. Les atomes appris sur des images naturelles avec cette méthode sont proches d'atomes de Gabor connus pour bien représenter les images naturelles également.

Une autre méthode de ce type est également répandue, il s'agit de K-SVD pour K-Singular Value Decomposition [Aharon *et al.* (2006)]. On y applique itérativement les mêmes étapes que [Olshausen *et al.* (1996)] en utilisant la pseudo-norme ℓ_0 , notée $\|\cdot\|_0$, pour terme induisant la parcimonie, ce qui rend le problème (2.30) plus délicat à résoudre :

$$\{\widehat{\mathbf{U}}, \widehat{\mathbf{D}}\} = \underset{\mathbf{U}, \mathbf{D}}{\operatorname{argmin}} \|\mathbf{DU} - \mathbf{X}\|_F^2 + \lambda \sum_i \|\mathbf{u}_i\|_0 \quad (2.30)$$

Dans la suite, nous expliquons les deux étapes de l'apprentissage d'un dictionnaire par l'algorithme K-SVD qui sont répétées itérativement. Dans la première étape, on calcule les coefficients $\widehat{\mathbf{U}}$ par une OMP [Pati *et al.* (1993)]. La deuxième étape consiste à mettre à jour le dictionnaire atome par atome. Ainsi, pour chaque patch d'indice i et pour l'atome $\hat{\mathbf{d}}_l$ (l -ème colonne de \mathbf{D}) en cours de mise à jour, on détermine l'ensemble des indices des patches utilisant $\hat{\mathbf{d}}_l$ défini par :

$$\Gamma_l = \{i, \hat{\mathbf{u}}_i(l) \neq 0\}$$

Puis, on calcule l'erreur commise si cet atome n'était pas pris en compte dans la représentation des patch d'indices appartenant à Γ_l comme le décrit l'équation suivante :

$$\mathbf{e}_i^l = \mathbf{R}_i \mathbf{x} - (\widehat{\mathbf{D}} \hat{\mathbf{u}}_i - \hat{\mathbf{d}}_l \hat{\mathbf{u}}_i(l)) \text{ où } : i \in \Gamma_l \quad (2.31)$$

où \mathbf{e}_i^l est un vecteur de taille $w^2 \times 1$ et \mathbf{R}_i est l'opérateur d'extraction d'un patch centré sur le pixel i .

En calculant \mathbf{e}_i^l pour tout i , on obtient une matrice d'erreur notée \mathbf{E}_l de taille $w^2 \times \operatorname{Card}(\Gamma_l)$. Soit \mathbf{u}_i^l le vecteur de taille $1 \times \operatorname{Card}(\Gamma_l)$ regroupant les valeurs des coefficients $\mathbf{u}_i(l)$ où $i \in \Gamma_l$. Mettre à jour $\hat{\mathbf{d}}_l$ et $\hat{\mathbf{u}}_i^l$ revient finalement à minimiser :

$$\{\hat{\mathbf{d}}_l, \hat{\mathbf{u}}_i^l\} = \underset{\mathbf{d}_l, \hat{\mathbf{u}}_i^l}{\operatorname{argmin}} \|\mathbf{E}_l - \mathbf{d}_l \hat{\mathbf{u}}_i^l\|_2^2 \quad (2.32)$$

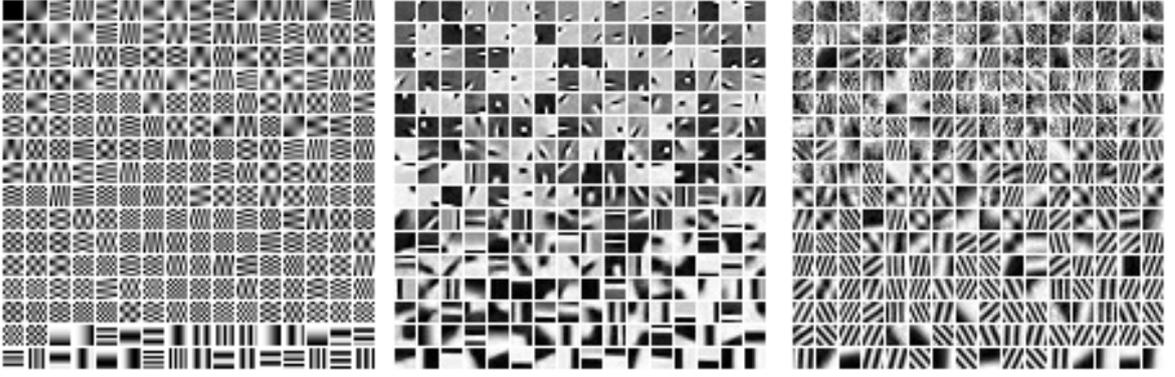


FIGURE 2.15 – De gauche à droite : un dictionnaire DCT, un dictionnaire appris sur des images naturelles avec K -SVD : tous deux sont utilisés pour initialiser le dictionnaire appris par K -SVD appliqué au débruitage. Enfin, un dictionnaire appris lors du débruitage de Barbara (cf. fig 2.16). On observe que les atomes appris ressemblent bien aux rayures présentes dans l'image dans différentes directions.

Ce qui revient à réaliser une approximation de rang 1 de la matrice \mathbf{E}_l , et par le théorème d'Eckart-Young, on y parvient en calculant sa SVD : $\mathbf{E}_l = \mathbf{H}\mathbf{S}\mathbf{G}^t$, où \mathbf{H} et \mathbf{G} sont des matrices orthogonales et \mathbf{S} nulle sauf sa diagonale comportant les valeurs singulières dans l'ordre décroissant. On remplace ensuite \mathbf{d}_l par la première colonne de \mathbf{H} et $\hat{\mathbf{u}}_i^l$ par la première colonne de \mathbf{G} multipliée par $\mathbf{S}(1,1)$.

L'algorithme K -SVD est appliqué à plusieurs problèmes tels que la compression et la classification mais nous présentons dans la suite son application au débruitage de bruit Gaussien décrite en détails dans [Elad et Aharon (2006)] et [Lebrun et Leclaire (2012)] où la formulation du problème d'optimisation associé devient :

$$\min_{\mathbf{x}, \mathbf{D}, \mathbf{U}} \eta \|\mathbf{x} - \mathbf{y}\|_2^2 + \sum_i \lambda \|\mathbf{u}_i\|_0 + \|\mathbf{D}\mathbf{u}_i - \mathbf{R}_i\mathbf{x}\|_2^2 \quad (2.33)$$

où \mathbf{y} est la version bruitée de l'image \mathbf{x} , $\mathbf{R}_i\mathbf{x}$ le i -ème patch extrait de \mathbf{x} et η une constante de pénalisation.

Les étapes de résolution du problème (2.33) se résument à travers les points suivants :

- Initialisation : $\mathbf{x} = \mathbf{y}$, \mathbf{D} : un dictionnaire DCT par exemple (cf. fig. 2.15).
- Répéter J fois :
 - Étape de Sparse Coding : $\forall i$, utiliser une OMP pour approcher la solution de :

$$\min_{\mathbf{u}_i} \|\mathbf{u}_i\|_0 \text{ tel que } \|\mathbf{R}_i\mathbf{x} - \mathbf{D}\mathbf{u}_i\|_2^2 \leq C\sigma^2 \quad (2.34)$$

où C est une constante et σ^2 la variance du bruit.

- Étape de mise à jour du dictionnaire : mettre à jour chaque colonne l de \mathbf{D} comme suit :

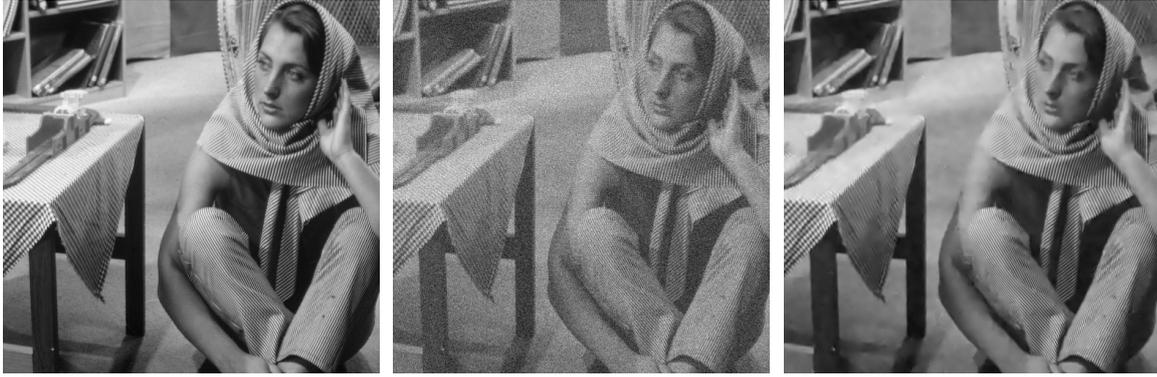


FIGURE 2.16 – Exemple de débruitage avec un dictionnaire K-SVD. De gauche à droite : l'image originale de Barbara, l'image bruitée par un bruit Gaussien d'écart-type $\sigma = 25$, l'image débruitée, PSNR : 29.56 dB.

- Trouver l'ensemble Γ_l des indices des patches qui utilisent cet atome.
- Pour chaque élément de Γ_l , calculer l'erreur en équation (2.31).
- Appliquer une SVD à la matrice des erreurs \mathbf{E}_l et mettre à jour d_l et les éléments de \mathbf{U} concernés.
- Poser la solution analytique du problème (2.33) notée $\hat{\mathbf{x}}$:

$$\hat{\mathbf{x}} = (\eta \mathbf{I} + \sum_i \mathbf{R}_i^t \mathbf{R}_i)^{-1} (\eta \mathbf{y} + \sum_i \mathbf{R}_i^t \mathbf{D} \mathbf{u}_i) \quad (2.35)$$

- Recombiner les patches débruités pour obtenir l'image reconstruite.

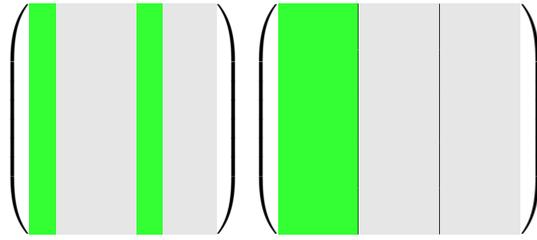
Cette méthode de débruitage est efficace et les dictionnaires appris s'adaptent aux données comme on peut l'observer sur l'exemple de débruitage en figure 2.16 et le dictionnaire appris en figure 2.15.

Plusieurs travaux développent l'algorithme K-SVD et s'en inspirent toujours actuellement, par exemple [Sulam *et al.* (2014)]. Cependant, en particulier dans des conditions de fort bruit, compte tenu du grand nombre de combinaisons possibles pour décomposer un patch comme l'illustre la figure 2.17, il est possible que \mathbf{u} ait un support différent d'une application d'OMP à une autre, ce qui nuit à la stabilité de ce modèle. Utiliser la parcimonie par blocs est une bonne alternative à ce défaut [Yu *et al.* (2012)].

2.4.3 Modélisation globale de l'espace des patches avec les GMM

2.4.3.1 Les modèles de mélange de Gaussiennes (GMM)

Les modèles de mélange de Gaussiennes sont des outils efficaces pour modéliser la distribution des patches tirés d'images naturelles [Weiss et Freeman (2007)] vu leur flexibilité et la richesse des informations qu'ils peuvent encoder. La distribution d'un patch \mathbf{x}_i sous forme de colonne de taille $w^2 \times 1$ est représentée par un GMM à travers une somme



Nombre de choix possibles :

$$\binom{K_a}{k} > K_b$$

FIGURE 2.17 – On illustre à travers ce schéma qu’avec la parcimonie par blocs, on a un nombre de choix possibles plus faible pour représenter un patch. À gauche est représenté un dictionnaire avec ses K_a atomes mis sous forme de colonnes. On utilise un degré de parcimonie égal à k pour décomposer un patch, ce qui donne k parmi K_a choix possibles de représentations. À droite est représenté un dictionnaire de K_b bases. On peut supposer que chaque base est représentée par les vecteurs propres de matrices de covariance apprises sur des clusters de patchs similaires mis sous forme de colonnes. On a donc K_b choix possibles pour sélectionner les atomes qui représentent un patch. Cette sélection est donc plus stable.

de K composantes Gaussiennes qui le constituent. Ces dernières sont caractérisées par un vecteur moyen $\boldsymbol{\mu}_k$, une matrice de covariance \mathbf{C}_k et un poids de mélange ω_k tel que $\sum_{k=1}^K \omega_k = 1$:

$$p(\mathbf{x}_i) = \sum_{k=1}^K \omega_k \frac{1}{(2\pi)^{w^2/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)\right) \quad (2.36)$$

On ne peut obtenir une expression analytique de l’estimateur du maximum de vraisemblance associé à la log-vraisemblance de cette distribution. Cependant, il est proposé notamment dans [Zoran et Weiss (2011)] de calculer pour chaque composante un poids de mélange conditionnel puis de sélectionner la composante gaussienne présentant le poids conditionnel maximal. Ceci revient à réaliser l’approximation suivante de l’équation (2.36) :

$$p(\mathbf{x}_i) \simeq \max_k \omega_k \frac{1}{(2\pi)^{w^2/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)\right) \quad (2.37)$$

L’algorithme EM : Le plus souvent, on apprend un GMM sur une large base de patchs avec l’algorithme Expectation-Maximization (EM) [Dempster *et al.* (1977)]. Cet

algorithme est une procédure itérative en deux étapes. La première est l'étape E, elle consiste à trouver le modèle qui explique le mieux chaque patch. La seconde est l'étape M où on met à jour le poids, la moyenne empirique et la matrice de covariance empirique de chaque modèle grâce à l'étape précédente. Plus formellement, cette procédure a pour objectif de déterminer l'ensemble des paramètres $\Theta = \{\omega_k, \theta_k\}_{k=\{1, \dots, K\}}$, où $\theta_k = \{\mu_k, C_k\}$, maximisant la log-vraisemblance :

$$L(\mathbf{x}, \Theta) = \sum_{i=1}^N \log \left(\sum_{k=1}^K \omega_k f(\mathbf{x}_i, \theta_k) \right) \quad (2.38)$$

$$\text{où } f(\mathbf{x}_i, \theta_k) = \frac{1}{(2\pi)^{w^2/2} \det(C_k)^{1/2}} \exp \left(-\frac{1}{2} (\mathbf{x}_i - \mu_k)^t C_k^{-1} (\mathbf{x}_i - \mu_k) \right)$$

En introduisant une variable latente \mathbf{z}_i qui est un vecteur de taille $K \times 1$ défini comme suit :

$$z_{ik} = \begin{cases} 1 & \text{si } \mathbf{x}_i \in k\text{-ème classe} \\ 0 & \text{sinon} \end{cases} \quad (2.39)$$

où z_{ik} est la valeur de la k -ème coordonnée de \mathbf{z}_i , et en écrivant la densité jointe de $(\mathbf{x}_i, \mathbf{z}_i)$:

$$p(\mathbf{x}_i, \mathbf{z}_i) = \prod_{k=1}^K \omega_k^{z_{ik}} f(\mathbf{x}_i, \theta_k)^{z_{ik}} \quad (2.40)$$

la log-vraisemblance complète des données s'écrit alors :

$$L((\mathbf{x}, \mathbf{z}), \Theta) = \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log(\omega_k f(\mathbf{x}_i, \theta_k)) \quad (2.41)$$

Soit Q l'espérance conditionnelle de la log-vraisemblance complétée :

$$Q(\Theta, \Theta^{(t)}) = \mathbb{E}_{\mathbf{z}} \left(L((\mathbf{x}, \mathbf{z}), \Theta) | \mathbf{x}, \Theta^{(t)} \right) = \sum_{i=1}^N \sum_{k=1}^K \mathbb{E}_{\mathbf{z}}(z_{ik} | \mathbf{x}, \Theta^{(t)}) \log(\omega_k f(\mathbf{x}_i, \theta_k)) \quad (2.42)$$

où $\Theta^{(t)}$ est l'ensemble des paramètres à l'itération t .

En théorie, la suite $\Theta^{(t+1)} = \underset{\Theta}{\operatorname{argmax}} Q(\Theta, \Theta^{(t)})$ fait tendre $L(\mathbf{x}, \Theta^{(t+1)})$ vers un maximum qui peut être local ou global en choisissant bien la valeur initiale $\Theta^{(0)}$ sinon il est possible qu'elle soit bloquée en un point selle [Krishnan et McLachlan (1997)].

En posant $\gamma_{ik} = \mathbb{E}_{\mathbf{z}}(z_{ik} | \mathbf{x}, \Theta^{(t)})$, l'étape E revient à calculer la probabilité a posteriori pour que \mathbf{x}_i appartienne à la classe k sachant $\Theta^{(t)}$:

$$\gamma_{ik} = \frac{\omega_k^{(t)} f(\mathbf{x}_i, \theta_k^{(t)})}{\sum_{p=1}^K \omega_p^{(t)} f(\mathbf{x}_i, \theta_p^{(t)})} \quad (2.43)$$

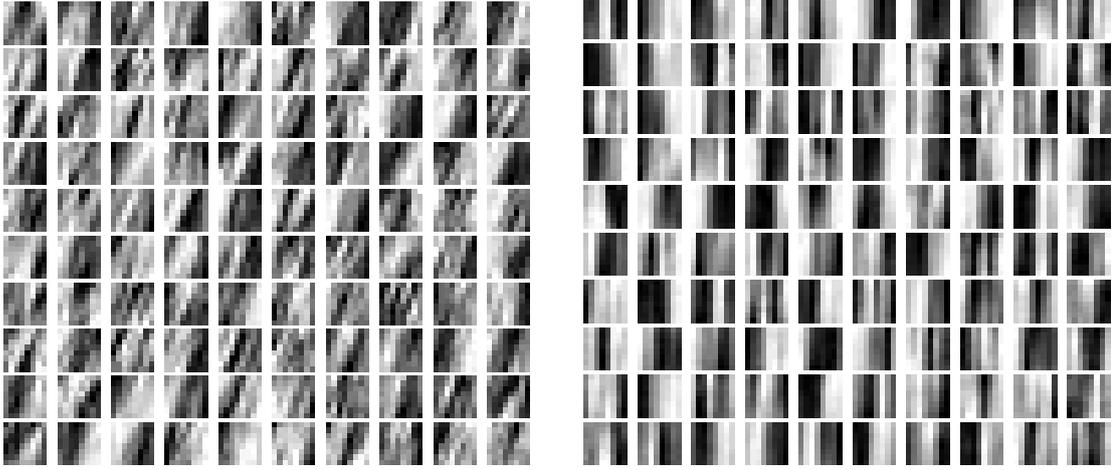


FIGURE 2.18 – Cent tirages de patches avec deux modèles Gaussiens différents, l'un représentant une texture rayée (à gauche) et l'autre un contour vertical (à droite).

Dans l'étape M, on détermine :

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmax}} Q(\Theta, \Theta^{(t)}) \quad (2.44)$$

où :

$$\hat{\omega}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik}, \quad \hat{\boldsymbol{\mu}}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik} \mathbf{x}_i, \quad \hat{\mathbf{C}}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^t \quad (2.45)$$

où Z_k est une constante de normalisation.

La richesse des structures encodées par des matrices de covariance : On peut constater la richesse des informations encodées dans une matrice de covariance \mathbf{C}_k en observant des patches tirés selon la loi $\mathcal{N}(\boldsymbol{\mu}_k, \mathbf{C}_k)$. Dans la figure 2.18, on observe des patches de largeur $w = 8$ générés par un modèle Gaussien représentant bien une texture rayée. Pour obtenir le tirage d'un patch sous forme de colonne noté \mathbf{p} , on calcule la décomposition de Cholesky de la matrice de covariance : $\mathbf{C}_k = \mathbf{S}^t \mathbf{S}$, puis on tire un vecteur $\mathbf{u} \sim \mathcal{N}(0_{\mathbb{R}^{w^2}}, \mathbf{I})$, ainsi : $\mathbf{p} = \boldsymbol{\mu}_k + \mathbf{S}^t \mathbf{u}$. On vérifie que la matrice de covariance de \mathbf{p} notée \mathbf{C}_p est bien la même que celle du modèle considéré : $\mathbf{C}_p = \mathbf{S}^t \mathbf{S} = \mathbf{C}_k$. De même pour la moyenne : $\boldsymbol{\mu}_p = \boldsymbol{\mu}_k$ car \mathbf{u} est de moyenne nulle.

Pour illustrer le fait que les GMM représentent bien les structures observées dans une image, nous présentons ci-dessous un outil que nous utiliserons fréquemment dans cette thèse qui est la carte de modèles Gaussiens. Un exemple est présenté en figure 2.19. Chaque couleur est associée à un modèle Gaussien qui représente le mieux un patch parmi une collection de modèles. Les couleurs proches dans la barre de couleurs représentent des modèles générant des structures proches. On précise que la barre de couleur est circulaire, c'est à dire que les deux couleurs extrêmes de la barre représentent



FIGURE 2.19 – De gauche à droite : l’image de Barbara, la carte des modèles Gaussiens utilisés après avoir traité sa version bruitée avec l’EPLL [Zoran et Weiss (2011)] ($\sigma = 20$) qui utilise un GMM de 200 composantes. On observe par exemple qu’au niveau des rayures ayant la même orientation, ce sont bien les mêmes modèles Gaussiens qui sont utilisés.

des structures proches. Pour obtenir ce classement des modèles Gaussiens, nous calculons la divergence de Kullback-Leibler entre les modèles pris deux à deux. Ensuite, on utilise un algorithme résolvant approximativement le problème du voyageur de commerce (décrit notamment dans [Applegate et al. (2011)]) afin d’ordonner les classes en une séquence dont la somme des divergences de proche en proche est faible. Ordonner ainsi les modèles permet d’obtenir une carte la plus lisse possible. Cet outil se révèle utile pour visualiser quel modèle Gaussien correspond le mieux à chaque patch dans l’image.

Les Gaussian Factor Models D’autres travaux, comme [Wang (2013)], modélisent la distribution des patches avec ce qu’on appelle des GFM pour Gaussian Factor Models qui, au lieu d’utiliser des matrices de covariance de rang plein, utilisent des matrices de rang faible, appelées facteurs, afin de représenter les patches. On peut comprendre intuitivement l’utilité de ce principe en remarquant la décroissance des valeurs propres des matrices de covariance d’un GMM (cf. fig. 2.20).

Plus formellement, un patch sous forme de colonne de taille $w^2 \times 1$ est modélisé par un facteur $\mathbf{V} \in \mathbb{R}^{w^2 \times q}$ de rang q comme suit :

$$\mathbf{x}_i = \mathbf{V}\mathbf{t} + \boldsymbol{\mu} \quad (2.46)$$

où $\mathbf{t} \sim \mathcal{N}(0_{w^2 \times q}, \mathbf{I})$ et $\boldsymbol{\mu}$ est le vecteur moyen associé au facteur \mathbf{V} . La matrice de covariance de \mathbf{x}_i devient $\mathbf{V}\mathbf{V}^t$. Contrairement à une matrice de covariance, un facteur n’est pas unique.

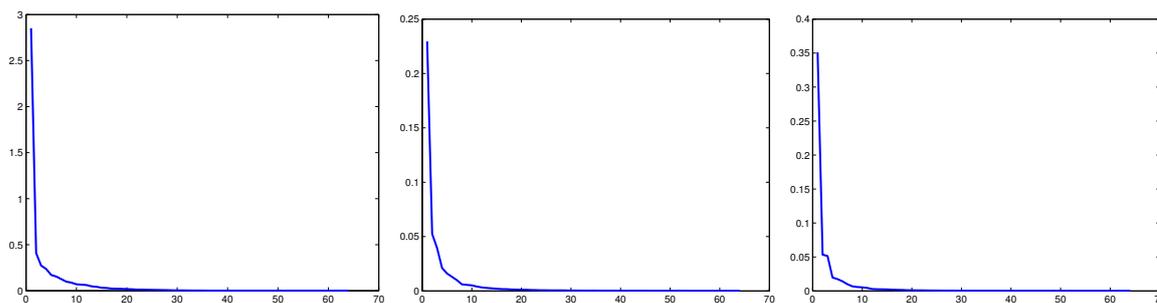


FIGURE 2.20 – *Décroissance des valeurs propres de trois matrices de covariance de taille 64×64 extraites du GMM appris dans [Zoran et Weiss (2011)].*

L'utilisation des GFM a plusieurs intérêts. Tout d'abord, ils permettent d'éviter le problème de modèles Gaussiens dégénérés (quand trop peu de patches d'une base d'apprentissage sont associés à un modèle Gaussien par exemple). Ensuite, du fait que les facteurs soient de rang faible, de nombreux calculs où on les manipule deviennent plus rapides (ex : pseudo-inversion de matrices). Une méthode d'apprentissage de GFM par un algorithme EM adapté est décrite dans l'annexe 5 du document [Tipping et Bishop (1999)]. Nous envisageons d'utiliser ce type de modèles dans nos travaux futurs.

2.4.3.2 L'algorithme EPLL

Comme nous l'avons précisé au paragraphe 2.4.2, la parcimonie par blocs évite d'avoir un trop grand nombre de choix possibles dans la représentation d'un patch comparé à la parcimonie classique. On illustre cette affirmation à travers la figure 2.17. Avec une approche de débruitage basée sur un GMM, restaurer un patch revient à choisir un modèle Gaussien (un bloc) pour le représenter au mieux. C'est ce qui est appliqué dans l'algorithme EPLL, pour Expected Patch Log Likelihood [Zoran et Weiss (2011)], utilisé dans plusieurs problèmes inverses dont le débruitage. L'idée principale des auteurs est que l'image restaurée doit être reconstruite de sorte qu'elle soit proche de l'image bruitée et que chaque patch estimé soit suffisamment "probable" pour cette reconstruction en respectant un a priori basé sur un GMM. Dans l'implémentation de Zoran et al., ce GMM est constitué de 200 composantes et appris au préalable sur une large base de 10^6 patches extraits d'images naturelles à la dynamique entre 0 et 1 avec un algorithme EM (voir sec. 2.4.3.1). Ces patches étant centrés et les vecteurs moyens de GMM appris sur de tels patches étant faibles, les auteurs ont choisi de mettre les moyennes à zéro. Nous reviendrons sur ce point en section 3.3.3.

L'EPLL est très proche du Piecewise Linear Estimator [Yu *et al.* (2012)] et de l'algorithme Non-Local Bayes [Lebrun *et al.* (2013)] présentés en section 2.3.2. Les principales différences avec l'EPLL sont :

- Dans l'EPLL, on utilise un GMM appris sur des patches tirés d'images naturelles

avec un algorithme EM et non des modèles Gaussiens appris sur des clusters de patches similaires dans l'image bruitée.

- Les modèles Gaussiens sont moins nombreux (environ 18) avec le PLE et NL-Bayes, tandis que dans l'EPLL, 200 modèles sont appris. Peu d'entre eux sont utilisés lors du débruitage d'une image. Néanmoins, ces modèles riches en information pourraient contenir des structures rares présentes dans certains types d'image.
- Les modèles Gaussiens appris dans [Lebrun *et al.* (2013)] ne sont pas de moyennes nulles.

Dans l'EPLL, le problème d'optimisation à résoudre afin de débruiter une image \mathbf{y} mise sous forme d'un vecteur s'écrit :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 - \sum_{i=1}^N \log p(\mathbf{R}_i \mathbf{x}) \quad (2.47)$$

où \mathbf{x} est l'estimation sans bruit de l'image mise sous forme de vecteur, \mathbf{R}_i est l'opérateur d'extraction du i -ème patch vectorisé et centré parmi les N patches qui se superposent dans l'image et λ est la constante réglant l'écart entre \mathbf{x} et \mathbf{y} . L'a priori p est celui défini dans l'équation (2.37) que nous rappelons ci-dessous en l'appliquant à $\mathbf{R}_i \mathbf{x}$ et avec les moyennes $\boldsymbol{\mu}_k$ nulles :

$$p(\mathbf{R}_i \mathbf{x}) \simeq \max_k \omega_k \frac{1}{(2\pi)^{w^2/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{R}_i \mathbf{x})^t \mathbf{C}_k^{-1}(\mathbf{R}_i \mathbf{x})\right) \quad (2.48)$$

Le problème (2.47), n'étant pas convexe en \mathbf{x} et en k en même temps, peut être résolu par une méthode itérative dite de half-quadratic splitting. En effet, en introduisant une variable auxiliaire \mathbf{z}_i qu'on souhaite très proche du patch \mathbf{x}_i , on peut résoudre le problème (2.47) de façon alternée :

$$\underset{\mathbf{x}, \mathbf{z}}{\operatorname{argmin}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \sum_i \frac{\delta}{2} (\|\mathbf{R}_i \mathbf{x} - \mathbf{z}_i\|^2) - \log p(\mathbf{z}_i) \quad (2.49)$$

où δ règle l'écart entre $\mathbf{R}_i \mathbf{x}$ et les variables auxiliaires \mathbf{z}_i .

Pour une valeur de δ fixée, on alterne donc l'optimisation en \mathbf{x} et en \mathbf{z}_i . Pour trouver le modèle Gaussien qui maximise la vraisemblance (cf. eq. (??)) de \mathbf{z}_i , on calcule sa log-vraisemblance Gaussienne par rapport à chacune des composantes du GMM, puis on sélectionne celle qui présente la vraisemblance maximale dont l'indice est noté k^* . Reste à résoudre le problème suivant :

$$\hat{\mathbf{z}}_i = \underset{\mathbf{z}_i}{\operatorname{argmin}} \frac{\delta}{2} \sum_{i=1}^N \|\mathbf{R}_i \mathbf{x} - \mathbf{z}_i\|^2 - \log \det(\mathbf{C}_{k^*}) + \mathbf{z}_i^t \mathbf{C}_{k^*}^{-1} \mathbf{z}_i \quad (2.50)$$

en mettant à zéro le gradient en \mathbf{z}_i . On aboutit alors à l'équation suivante :

$$\hat{\mathbf{z}}_i = (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} (\mathbf{C}_{k^*} \mathbf{R}_i \mathbf{x}) \quad (2.51)$$

où \mathbf{C}_{k^*} est la matrice de covariance associée au modèle gaussien à la plus forte vraisemblance, σ^2 est la variance du bruit et \mathbf{I} la matrice identité.

Pour l'optimisation en \mathbf{x} , en calculant le gradient en \mathbf{x} puis en le mettant à zéro on obtient :

$$\hat{\mathbf{x}} = \left(\lambda \mathbf{I} + \delta \sum_j \mathbf{R}_j^T \mathbf{R}_j \right)^{-1} \left(\lambda \mathbf{y} + \delta \sum_j \mathbf{R}_j^T \hat{\mathbf{z}}_j \right) \quad (2.52)$$

L'expression (2.52) revient à réaliser une combinaison linéaire de l'image bruitée \mathbf{y} et des patches \mathbf{z}_i se projetant sur un pixel donné. L'algorithme complet est résumé dans le pseudo-code 1.

Algorithm 1 EPLL

Require: N_{pa} : nombre de patches

Initialisation : $\mathbf{x}_i = \mathbf{y}_i, \forall i \in \{1, \dots, N_{\text{pa}}\}$

for $\delta = [1, 4, 8]/\sigma^2$ **do**

for $i \in \{1, \dots, N_{\text{pa}}\}$ **do**

$k^* = \operatorname{argmin}_k -2 \log(\omega_k) + \log \det(\mathbf{C}_k) + \mathbf{z}_i^t \mathbf{C}_k^{-1} \mathbf{z}_i$

$\mathbf{z}_i = (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} \mathbf{C}_{k^*} \mathbf{x}_i$

end for

 Mettre à jour \mathbf{x} avec l'équation (2.52)

end for

En théorie, en itérant cet algorithme, il converge quand $\delta \rightarrow +\infty$. En pratique, on obtient une solution raisonnable en augmentant la valeur de δ au fil des itérations. Le plus souvent 3 ou 4 itérations sont suffisantes mais on peut en ajouter une en cas de fort bruit. Le réglage du paramètre δ reste un problème ouvert, cependant, après étude expérimentale, la stratégie proposée par les auteurs de poser : $\delta \in \{1, 4, 8, 16\}/\sigma^2$.

Les performances de l'EPLL sont du même ordre que les méthodes état de l'art en débruitage (cf. fig. 2.21). Plusieurs contributions de cette thèse sont des extensions ou généralisations de cet algorithme, notamment l'introduction d'une propriété d'invariance par translation (cf. sec. 3.2.2) et une extension au débruitage de bruit multiplicatif (cf. sec. 4.3).

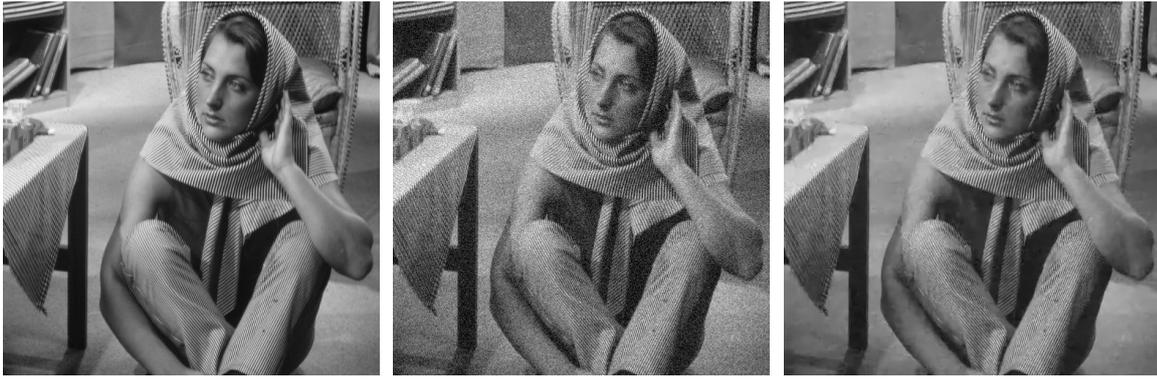


FIGURE 2.21 – De gauche à droite : l’image originale, bruitée ($\sigma = 20$), débruitée avec l’EPLL (PSNR = 29.78 dB).

2.5 Conclusion

Dans ce chapitre, nous avons décrit les grandes familles de méthodes de débruitage développées pour le bruit Gaussien, en particulier les modèles utilisant des patches. La plupart sont illustrées dans le schéma 2.22. Nous avons présenté des méthodes utilisant une modélisation locale de l’espace des patches et d’autres méthodes considérant l’espace des patches dans sa globalité. Parmi les méthodes utilisant une modélisation locale de l’espace des patches, l’algorithme BM3D [Dabov *et al.* (2007)] présente des performances très satisfaisantes grâce au principe du filtrage collaboratif. L’algorithme NL-Bayes [Lebrun *et al.* (2013)] utilise aussi ce principe et présente de bonnes performances également. Il constitue des modèles Gaussiens à partir de clusters de patches similaires dans l’image bruitée. Dans [Guillemot *et al.* (2014)], on constate qu’apprendre un GMM sur une base de patches extraits d’images naturelles améliore les résultats de NL-Bayes. Ainsi, compte tenu du fait qu’une méthode basée sur des modèles Gaussiens peut être améliorée par une modélisation globale de l’espace des patches, nous avons étudié l’algorithme EPLL [Zoran et Weiss (2011)], où un GMM est appris sur une large base de patches tirés d’images naturelles. Nous avons également présenté des méthodes modélisant l’espace des patches parcimonieusement, notamment avec des dictionnaires appris sur des patches. Nous décrivons dans le chapitre 3 différentes méthodes introduisant une propriété d’invariance par translation et d’invariance aux transformations radiométriques affines dans des approches par dictionnaires de patches et par GMM. Nous verrons ensuite des applications de ces modèles à l’imagerie radar dans le chapitre 4.

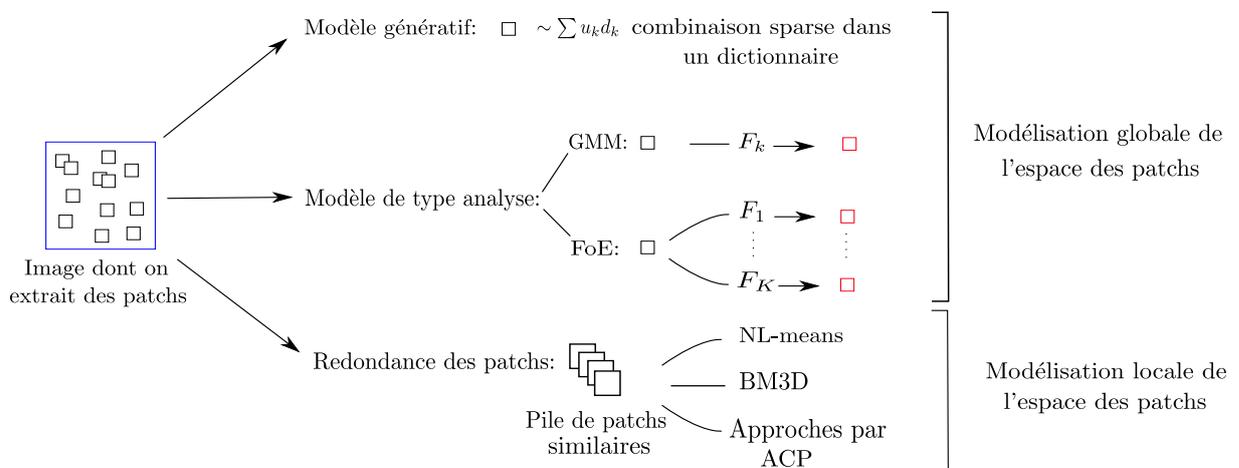


FIGURE 2.22 – Schéma récapitulant les principaux modèles de restauration par patches décrits dans ce chapitre. Avec une approche de type synthèse, on peut notamment représenter un patch par une combinaison linéaire parcimonieuse d'atomes \mathbf{d}_k avec un poids u_k . Deux exemples d'approches de type analyse sont illustrés. Pour les GMM, on sélectionne un modèle k représentant au mieux le patch d'intérêt puis on lui applique l'opérateur de blanchiment \mathbf{F}_k où ce dernier correspond à la décomposition de Cholesky de l'inverse de la matrice de covariance \mathbf{C}_k . Pour les Fields of Experts (FoE), un patch doit être restauré sous contrainte d'être co-sparse dans $\{\mathbf{F}_k\}_{k \in \{1, \dots, K\}}$ appris sur une base d'apprentissage. Enfin, nous citons quelques méthodes basées sur la similarité des patches d'une image.

Chapitre 3

Propriétés d'invariance des modèles d'images

Dans ce chapitre, nous traitons de la modélisation d'images avec des propriétés d'invariance. Nous nous focalisons sur l'invariance par translation et l'invariance aux transformations radiométriques affines. Nous définissons ces invariances et motivons le besoin de ce type de modèles en section 3.1. Ensuite, nous décrivons les modèles existants munis de ces invariances aux paragraphes 3.2.1 pour l'invariance par translation et 3.3.1 pour l'invariance aux transformations radiométriques affines. Dans nos contributions, nous proposons des modèles basés sur des dictionnaires de patches et des modèles basés sur des mélanges de Gaussiennes (GMM). En section 3.2.2, nous présentons comment introduire l'invariance par translation dans un a priori basé sur des mélanges de Gaussiennes. Cette méthode peut être utilisée avec d'autres a priori. Dans la section 3.3.3, nous décrivons un modèle simplifié de mélanges de Gaussiennes invariant radiométriquement. Nous montrons qu'il présente de meilleures performances qu'un GMM classique dans une application au débruitage. Enfin, nous présentons des approches parcimonieuses sur dictionnaire munies des deux invariances considérées en section 3.4.

3.1 Motivation et définition des invariances considérées

Compte tenu de la quantité massive de données récoltées quotidiennement, qu'il s'agisse d'images naturelles (photos sur internet, bases de données d'entreprises) ou d'images satellitaires, développer des modèles munis de propriétés d'invariance peut avoir plusieurs intérêts. Tout d'abord, en ce qui concerne les approches par dictionnaires décrites au paragraphe 2.4.2, la plupart apprennent des dictionnaires redondants où un même motif doit être représenté par des atomes différentes à une translation près ou à une transformation radiométrique affine près. Supposons qu'on dispose d'approches qui n'encoderaient qu'une version d'un motif dans un atome de plus grande taille que les patches de l'image. Supposons également connus les paramètres de translation optimaux

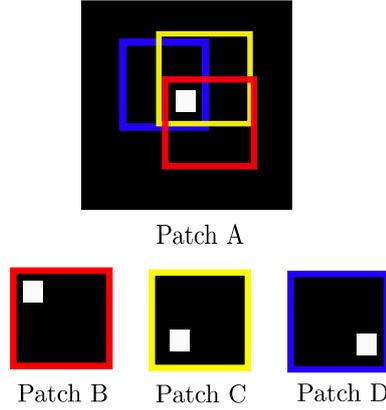


FIGURE 3.1 – *Le carré A est l'image d'intérêt. Les carrés B, C, D représentent les atomes d'un dictionnaire appris sur cette image qui contiennent donc la même information à une translation près. L'invariance par translation a pour but d'éviter cette redondance explicite et de rendre ainsi les dictionnaires plus compacts.*

pour mettre en correspondance l'atome et tous les patches contenant une version translaturée du motif, comme il est illustré en figure 3.1. Alors, ce type d'approche produirait des dictionnaires de taille réduite et ainsi une manière plus compacte de représenter les données. De plus, ce type de méthodes éviterait le cas où un dictionnaire ne contiendrait pas la translation optimale d'un motif pour le représenter. Le même raisonnement est valable pour l'invariance aux transformations radiométriques affines.

D'un point de vue applicatif, en imagerie satellitaire, les zones urbaines contiennent beaucoup de structures répétitives pour lesquelles des approches invariantes par translation seraient très utiles. Par ailleurs, une seule image peut présenter une radiométrie variable d'une zone à l'autre comme on peut le voir en figure 3.2 et l'invariance aux transformations radiométriques affines rendrait son traitement plus aisé.

Dans cette thèse, nous proposons d'étudier l'invariance à une transformation notée \mathbf{T} et de paramètre $\boldsymbol{\theta}$ comme l'égalité suivante, où p est un a priori exprimé sur une image ou des patches :

$$\forall \boldsymbol{\theta}, \quad \mathbf{J}_{\mathbf{T}_{\boldsymbol{\theta}}}(\mathbf{x}) p(\mathbf{T}_{\boldsymbol{\theta}}(\mathbf{x})) = p(\mathbf{x}) \quad (3.1)$$

où la distribution de \mathbf{x} est la même que la distribution de la transformée de \mathbf{x} et $\mathbf{J}_{\mathbf{T}_{\boldsymbol{\theta}}}$ désigne le jacobien de $\mathbf{T}_{\boldsymbol{\theta}}$. Plusieurs types de transformations sont possibles. Nous nous intéresserons à l'invariance par translation et l'invariance aux transformations radiométriques affines.

La translation Nous définissons la translation 2D entière $\mathbf{T}_{\mathbf{p}}$ de paramètre $\mathbf{p} = (h, v)$ à travers un opérateur linéaire rectangulaire appliqué à une image ou un patch et renvoyant



FIGURE 3.2 – Image SAR d’amplitude obtenue par le système ONERA Sethi dans le cadre du projet TropiSAR présentant une radiométrie variable entre le début de la fauchée (à gauche) et la fin (à droite) en raison du lobe d’antenne.

une image ou un patch dont le contenu est décalé :

$$[\mathbf{T}_{\mathbf{p}}\mathbf{x}]_k = \mathbf{x}_{k'} \quad (3.2)$$

où \mathbf{x} est de taille $w \times w$, $\mathbf{p} = (h, v)$ est un vecteur de translation entière, (i_k, j_k) est l’indice 2D au sein du patch du k -ème pixel, $(i_{k'}, j_{k'})$ l’indice 2D au sein du patch du k' -ème pixel avec k' tel que $i_{k'} = i_k + h$ modulo w et $j_{k'} = j_k + v$ modulo w .

Les transformations radiométriques affines Une transformation affine d’un patch est définie par :

$$\mathcal{T}(\mathbf{x}_j) = \alpha\mathbf{x}_j + \beta\mathbf{1} \quad (3.3)$$

où \mathbf{x}_j est le j -ème patch de l’image \mathbf{x} , α et β sont les paramètres scalaires de cette transformation affine et $\mathbf{1}$ un vecteur dont toutes les entrées sont égales à 1. La figure 3.3 permet de voir l’effet produit par une transformation affine sur une image. Un modèle muni de la propriété d’invariance aux transformations radiométriques affines permet de traiter de la même manière un patch \mathbf{x}_i et sa version transformée \mathbf{x}_j en calculant par exemple les paramètres α et β optimaux tels que $\mathbf{x}_i = \alpha\mathbf{x}_j + \beta\mathbf{1}$.

Dans la suite de ce chapitre, nous décrivons nos contributions : un a priori invariant par translation en section 3.2 et un modèle invariant par inversion de contraste en section 3.3. Ils sont tous deux basés sur des GMM et appliqués séparément au débruitage d’images. Ils seront fusionnés dans des travaux futurs. En section 3.4, nous proposons également deux contributions où l’on retrouve les deux types d’invariance considérés dans cette



FIGURE 3.3 – Effet produit par un changement de contraste affine. De gauche à droite, l'image originale du cameraman (dynamique : 0-255) notée \mathbf{x} et les images transformées : $1.5 \times \mathbf{x}$, $0.5 \times \mathbf{x}$. Pour afficher les images transformées, ces dernières sont converties en *wint8* (*unsigned 8-bits integer*), c'à-d que les valeurs hors de l'intervalle $[0, 255]$ sont tronquées.

thèse. L'une des contributions est basée sur la maximisation d'une corrélation normalisée et l'autre sur le calcul d'une carte d'erreur d'approximation d'une image par un patch. Ces deux contributions seront appliquées au débruitage. La méthode de calcul de carte d'erreur sera également appliquée à l'apprentissage de dictionnaires dans le chapitre 4.

3.2 Approches invariantes par translation

Il existe plusieurs façons d'obtenir des modèles invariants par translation. On décrit entre autres plusieurs méthodes d'apprentissage de dictionnaires existantes munies de cette propriété en section 3.2.1. Le plus souvent le problème revient à maximiser une corrélation entre l'atome en cours d'apprentissage et les patches de la base d'apprentissage. Deux choix sont possibles pour éviter les problèmes de bords dus à la translation : soit on impose que le patch soit de plus grande taille que l'atome ou le contraire. Le premier choix est illustré en figure 3.4. Dans nos contributions, nous proposons d'introduire l'invariance par translation dans la manière de recombinaison des patches restaurés (en section 3.2.2). D'autres modèles invariants par translation mais aussi invariants par transformations radiométriques affines seront décrits en section 3.4.

3.2.1 État de l'art des approches invariantes par translation

Souvent, les méthodes d'apprentissage de dictionnaires invariants par translation suivent le même schéma que les méthodes d'apprentissage de dictionnaires classiques avec deux principales étapes : le codage parcimonieux (ou *sparse-coding* en anglais) et la mise à jour des atomes du dictionnaire. La procédure [Olshausen *et al.* (1996)] décrite au paragraphe 2.4.2 est rendue invariante par translation dans [Hashimoto *et Kurata* (2000)]. Le dictionnaire est appris de la même façon qu'avec la méthode

$$\max_{\mathbf{p}} \left\langle \begin{array}{c} \blacksquare \\ \text{patch} \end{array}, T_{\mathbf{p}} \begin{array}{c} \blacksquare \\ \text{atome} \end{array} \right\rangle = \max_{\mathbf{p}} \begin{array}{c} \blacksquare \\ \text{patch} \end{array} \star \begin{array}{c} \blacksquare \\ \text{atome} \\ + \text{zéro-padding} \end{array} \longrightarrow \begin{array}{c} \blacksquare \\ \text{patch} \\ \text{découpé} \end{array}$$

FIGURE 3.4 – Illustration de l’application de l’opérateur de translation \mathbf{T} où l’opération \star désigne une corrélation. Le patch est de plus grande taille que l’atome pour éviter les effets de bords. Calculer le paramètre optimal de translation 2D entière, noté \mathbf{p} , revient à localiser la corrélation maximale entre le patch et l’atome zéro-paddé. Grâce à ce paramètre de translation, on est en mesure de découper la portion du patch (de même taille que l’atome) qui correspond le mieux à l’atome.

originale sauf que les atomes sont d’une taille plus petite que les patches de l’image afin d’éviter des effets de bords. Toutes les translations possibles de ces atomes sont considérées, ce qui engendre une grande complexité. Les auteurs ont élaboré une technique afin d’améliorer les temps de calculs de cet algorithme et parviennent à obtenir des atomes représentant mieux les données considérées qu’avec la procédure d’Olshausen et al.

L’algorithme K-SVD [Aharon *et al.* (2006)] décrit au paragraphe 2.4.2 a également été rendu invariant par translation dans plusieurs travaux tels que [Mailhé *et al.* (2008)] pour les signaux 1D et [Thiagarajan *et al.* (2008)] pour les images. L’étape de sparse-coding est mise en œuvre dans [Thiagarajan *et al.* (2008)] à l’aide d’un algorithme de Matching-Pursuit rendu invariant par translation décrit dans l’algorithme 2. Il permet de trouver $\forall k$, le coefficient u_k associé à l’atome \mathbf{d}_k et le paramètre de translation entière 2D \mathbf{p} afin de représenter un patch \mathbf{x}_j de la façon suivante : $\mathbf{x}_j \simeq \sum_{k=1}^K u_k \mathbf{T}_{\mathbf{p}} \mathbf{d}_k$.

Algorithm 2 Matching-Pursuit invariant par translation

Require: \mathbf{r}_0 : résidus, \mathbf{x}_j : patch d’entrée et $\forall k, \mathbf{p}, u_k^{(0)} = 0$

for $i = 1$ à N_{it} **do**

$(k^{(i)}, \mathbf{p}^{(i)}) = \underset{k, \mathbf{p}}{\operatorname{argmax}} \langle \mathbf{r}^{(i-1)}, \mathbf{T}_{\mathbf{p}} \mathbf{d}_k \rangle$ // Recherche de l’atome et de la translation maximisant la corrélation

$\gamma^{(i)} = \langle \mathbf{r}^{(i-1)}, \mathbf{T}_{\mathbf{p}^{(i)}} \mathbf{d}_{k^{(i)}} \rangle / \|\mathbf{T}_{\mathbf{p}^{(i)}} \mathbf{d}_{k^{(i)}}\|$ // Calcul du coefficient d’approximation

$u_k^{(i)} = u_k^{(i-1)} + \gamma^{(i)}$

$\forall (k, \mathbf{p}) \neq (k^{(i)}, \mathbf{p}^{(i)}), u_k^{(i)} = u_k^{(i-1)}$

$\mathbf{r}^{(i)} = \mathbf{r}^{(i-1)} - \gamma^{(i)} \mathbf{T}_{\mathbf{p}^{(i)}} \mathbf{d}_{k^{(i)}}$ // Mise à jour des résidus

end for

Quant à la mise à jour d’un atome \mathbf{d}_k le plus corrélé avec les patches $\mathbf{x}_n, n \in \{1, \dots, N_k\}$ avec N_k le nombre de patches représentés par le k -ème atome, [Thiagarajan *et al.* (2008)]

proposent de la réaliser en résolvant le problème suivant en utilisant l'adjoint de $\mathbf{T}_{\mathbf{p}}$:

$$\mathbf{d}_k = \operatorname{argmax}_{\|\mathbf{d}\|_2=1} \sum_{n=1}^{N_k} |\langle \mathbf{x}_n, \mathbf{T}_{\mathbf{p}_n} \mathbf{d} \rangle|^2 = \operatorname{argmax}_{\|\mathbf{d}\|_2=1} \sum_{n=1}^{N_k} |\langle \mathbf{T}_{-\mathbf{p}_n} \mathbf{x}_n, \mathbf{d} \rangle|^2 \quad (3.4)$$

où \mathbf{p}_n est le paramètre de translation de l'atome assurant la meilleure corrélation avec le patch \mathbf{x}_n .

Pour la minimisation de ce problème, on définit la matrice \mathbf{X} dont les colonnes sont les patches translattés $\mathbf{T}_{-\mathbf{p}_n} \mathbf{x}_n$. Le problème d'optimisation est ensuite reformulé comme suit :

$$\mathbf{d}_k = \operatorname{argmax}_{\|\mathbf{d}\|_2=1} \mathbf{d}^t \mathbf{X} \mathbf{X}^t \mathbf{d} \quad (3.5)$$

ce qui permet de le résoudre par une approximation de rang 1 de \mathbf{X} . Pour cela, on calcule une décomposition en valeurs singulière (SVD) de la matrice $\mathbf{X} = \mathbf{H} \mathbf{S} \mathbf{V}^{-1}$ et $\mathbf{d}_k = \mathbf{S}(1,1) \mathbf{h}_1$ où $\mathbf{S}(1,1)$ est la plus grande valeur singulière et \mathbf{h}_1 le vecteur associé à $\mathbf{S}(1,1)$. D'après [Thiagarajan *et al.* (2008)], les résultats de l'approche proposée en débruitage sont proches de l'algorithme K-SVD classique. Le problème majeur de cette approche est que la translation utilisée est circulaire, ce qui tend à apprendre des discontinuités dans le dictionnaire qui n'existent pas dans les patches de la base d'apprentissage dues à la périodisation créée lors de la translation. Notons que leur formulation resterait applicable avec des atomes de plus grande taille que les patches.

L'algorithme MoTIF (Matching of Time Invariant Features) [Jost *et al.* (2006)] construit un dictionnaire invariant par translation de façon incrémentale. Il se démarque des autres approches en assurant que chaque nouvel atome soit décorrélié des atomes précédemment calculés (dénominateur dans l'équation (3.6)) tout en étant bien corrélé à la base de patches sur laquelle il est appris (numérateur dans l'équation (3.6)) tel que l'exprime l'équation suivante :

$$\mathbf{d}_k = \operatorname{argmax}_{\|\mathbf{d}\|_2=1} \frac{\sum_{n=1}^N \max_{\mathbf{p}_n} c_{n,\mathbf{p}_n}^2 \langle \mathbf{x}_n, \mathbf{T}_{\mathbf{p}_n} \mathbf{d} \rangle^2}{\sum_{l=0}^{k-1} \sum_{\mathbf{p}} \langle \mathbf{d}_l, \mathbf{T}_{\mathbf{p}} \mathbf{d} \rangle^2} \quad (3.6)$$

où N est le nombre de patches, \mathbf{d}_k l'atome en cours d'apprentissage, \mathbf{x}_n le n -ième élément de la base d'apprentissage qui est d'une taille plus grande que l'atome afin d'éviter les problèmes de bords et $\mathbf{T}_{\mathbf{p}_n}$ l'opérateur de translation entière 2D de paramètre \mathbf{p}_n . Enfin, $c_{n,\mathbf{p}_n} = \frac{1}{\|\tilde{\mathbf{x}}_{n,-\mathbf{p}_n}\|}$ est un scalaire où $\tilde{\mathbf{x}}_{n,-\mathbf{p}_n}$ désigne l'adjoint de $\mathbf{T}_{\mathbf{p}_n} \mathbf{x}_n$ qui correspond à la translattée de \mathbf{x}_n par le vecteur $-\mathbf{p}_n$ restreinte au support de \mathbf{d}_k (voir figure 3.4 pour illustration).

Le problème (3.6) se résout itérativement de manière alternée : premièrement on calcule le paramètre de translation 2D entière optimal \mathbf{p}_n . Pour cela, on localise la corrélation maximale dans l'équation (3.7) en utilisant la transformée de Fourier :

$$\langle \mathbf{x}_n, \mathbf{T}_{\mathbf{p}_n} \mathbf{d} \rangle = \sum_{i,j} \mathbf{x}(i-h, j-v) \mathbf{d}_e(i,j) = \mathcal{F}^{-1} [\mathcal{F}[\mathbf{x}] \cdot \mathcal{F}[\mathbf{d}_e]^*](h, v) \quad (3.7)$$

où \mathbf{d}_e est l'image de même taille que \mathbf{x} contenant uniquement des zéros sauf sur une fenêtre de taille $(w \times w)$ où elle est égale à \mathbf{d} à partir de la position en haut à gauche dans la sous-fenêtre de coordonnées dans l'image $i-h$ en abscisse et $j-v$ en ordonnée. Pour la mise à jour de l'atome \mathbf{d}_k , à \mathbf{p}_n fixé, cela revient à résoudre un problème de décomposition en valeurs propres généralisées.

En pratique, l'algorithme MoTIF apprend des atomes ressemblant à des détecteurs de contours.. Selon les auteurs, la forme des atomes est plus due à la contrainte de décorrélation entre les éléments du dictionnaire qu'à la contrainte de corrélation à la base d'apprentissage. Cependant, expérimentalement, ces atomes parviennent à représenter parcimonieusement le type de données sur lequel ils sont appris.

Une façon intéressante d'apprendre un dictionnaire invariant par translation est de considérer les atomes comme des bancs de filtres convolutifs tel qu'il est proposé notamment dans [Morup *et al.* (2008)], [Kavukcuoglu *et al.* (2010)], [Tartavel (2015)]. Dans ces approches, on ne décompose pas l'image en patches et on modélise cette dernière comme une combinaison linéaire parcimonieuse de tous les translatés possibles des atomes du dictionnaire. La formulation d'un tel problème peut s'écrire de la façon suivante :

$$\operatorname{argmin}_{\mathbf{D}, \mathbf{u}} \left\| \mathbf{x} - \sum_{k=1}^K \mathbf{d}_k * \mathbf{u}_k \right\|_2^2 + \|\mathbf{u}\|_1 \quad (3.8)$$

où \mathbf{d}_k est un filtre, $\mathbf{D} = \{\mathbf{d}_k\}$ l'ensemble des filtres, \mathbf{x} une image, $*$ l'opérateur de convolution discrète, et \mathbf{u}_k la carte des coefficients parcimonieux associée au k -ème atome dont la parcimonie est induite par la norme $\|\mathbf{u}\|_1 = \sum_{k,i,j} |\mathbf{u}_k(i,j)|$. Ces approches utilisent souvent des descentes alternées (block-coordinate) afin de résoudre le problème d'optimisation associé. Les dictionnaires appris avec ce type d'approche parviennent à apprendre des structures plus complexes que des atomes de Gabor (donc plus complexes que [Jost *et al.* (2006)] et [Olshausen *et al.* (1996)]).

Une autre méthode invariante par translation consiste à apprendre un épitome, ou une collection d'épitomes. Introduit par [Jojic *et al.* (2003)] et repris notamment dans [Benoît *et al.* (2011)], il s'agit d'une imagerie de largeur et longueur jusqu'à trois ou quatre fois plus petites que celles de l'image d'intérêt, représentant un résumé des couleurs, des

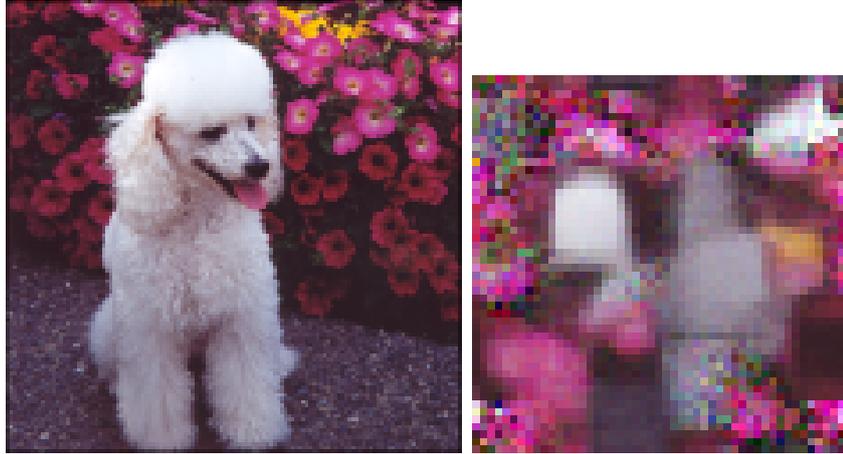


FIGURE 3.5 – À gauche l'image d'intérêt de taille 128×128 , à droite l'épitome appris par [Jojic et al. (2003)] de taille 50×50 à des fins de débruitage.

structures et des textures contenues dans une image. Un exemple est présenté en figure 3.5.

Les méthodes basées sur des filtres convolutifs et sur les épitomes sont d'un grand intérêt que ce soit du point de vue des modèles proposés, qu'au point de vue de l'implémentation. En effet, dans le modèle de filtres convolutifs, on ne découpe pas l'image en patches, ce qui évite l'apprentissage de plusieurs atomes redondants. Quant à l'épitome, il introduit par essence l'invariance par translation étant donné qu'il n'est pas un ensemble d'atomes mais une imagerie. Nous verrons dans la section suivante comment nous proposons d'introduire la propriété d'invariance par translation dans un a priori exprimé sur des patches et basé sur un GMM qui est un outil efficace pour modéliser la distribution des patches.

3.2.2 Un a priori invariant par translation

Dans ce paragraphe, nous proposons de rendre une approche invariante par translation d'une manière autre qu'à travers l'apprentissage de dictionnaire. La méthode que nous proposons intervient lors de l'agrégation de patches restaurés. Elle est décrite dans [Tabti et al. (2014b)].

Son principe est le même que celui des cartes d'erreurs d'approximations avec invariances que nous décrivons au paragraphe 3.4.4, qui réside dans la sélection pour chaque pixel du patch le représentant le mieux parmi tous ceux qui le contiennent (cf. illustration 3.6). Cette sélection revient à réaliser un pavage de l'image qui induit l'invariance par translation, ce qui se rapproche du principe des procédures d'apprentissage de dictionnaires se basant sur l'équation (3.8).

Ce principe est assez général pour être utilisé dans des méthodes de débruitage existantes où un a priori non-invariant par translation exprimé sur les patches s'écrirait :

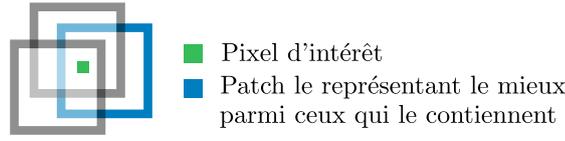


FIGURE 3.6 – Illustration de la façon dont est introduite l'invariance par translation.

$p(\mathbf{x}) = \prod_i p(\mathbf{x}_i)$, alors que l'a priori invariant par translation proposé est :

$$p(\mathbf{x}) = \prod_i \max_{j \in \mathcal{N}(i)} p(\mathbf{x}_j) \quad (3.9)$$

où $\mathcal{N}(i)$ est le voisinage du pixel i .

Donnons un exemple avec l'EPLL [Zoran et Weiss (2011)] décrit en section 2.4.3 et récapitulé dans l'algorithme 1. On rappelle tout d'abord la formulation classique du problème d'optimisation associé :

$$\hat{\mathbf{x}} = \min_{\mathbf{x}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \sum_i \min_{\mathbf{z}_i} \frac{\delta}{2} (\|\mathbf{x}_i - \mathbf{z}_i\|^2) - \log p(\mathbf{z}_i) \quad (3.10)$$

où $\hat{\mathbf{x}}$ est l'estimation de l'image \mathbf{x} , \mathbf{y} est l'image bruitée, \mathbf{x}_i est le i -ème patch tiré de \mathbf{x} , \mathbf{z}_i est une variable auxiliaire qu'on souhaite la plus proche possible de \mathbf{x}_i pour les besoins de la résolution du problème d'optimisation et p est un a priori de type GMM (voir l'équation (2.48)). Avec l'a priori invariant par translation proposé, cette formulation est remplacée par :

$$\hat{\mathbf{x}} = \min_{\mathbf{x}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \sum_i \min_{j \in \mathcal{N}(i), \mathbf{z}_i} \left\{ \frac{\delta}{2} \|\mathbf{x}_j - \mathbf{z}_i\|^2 - \log p(\mathbf{z}_i) \right\} \quad (3.11)$$

On illustre les améliorations apportées par l'introduction de l'invariance par translation dans une expérience sur une image synthétique d'un carré.

Pour cela, un GMM d'une dizaine de composantes est appris sur plusieurs translations de contours horizontaux et verticaux afin de représenter les côtés du carré. Cinq composantes représentent les translations verticales et les cinq autres les translations horizontales.

Un second GMM est appris avec seulement 2 composantes, l'une encodant une transition verticale et l'autre une horizontale.

Pour apprendre ces deux GMM, on génère des centaines de patches avec des transitions (verticales ou horizontales) à des positions aléatoires et on calcule la matrice de covariance empirique qui leur est associée. Les poids des composantes des GMM sont équirépartis.

La figure 3.7 montre des vecteurs propres de plusieurs composantes de ces deux GMM et montre qu'on apprend bien les structures verticales et horizontales désirées.

On utilise par la suite ces GMM pour réaliser un débruitage sur bruit gaussien d'écart-type $\sigma = 20$ de l'image synthétique avec l'EPLL et le SI-EPLL (pour Shift-Invariant EPLL). Les paramètres utilisés sont les mêmes que ceux décrits en section 1. Les résultats sont présentés sur la figure 3.8.

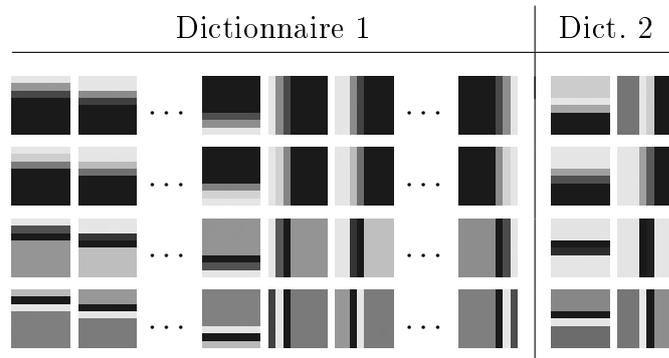


FIGURE 3.7 – Chaque colonne représente les 4 premiers vecteurs propres des matrices de covariance des deux dictionnaires. Le premier dictionnaire modélise les contours horizontaux et verticaux (avec une précision de ± 1 pixel) pour toutes les translations possibles du patch avec un mélange de 10 gaussiennes de moyennes nulles. Le second dictionnaire modélise seulement une position de ces contours avec un mélange de deux gaussiennes de moyennes nulles.

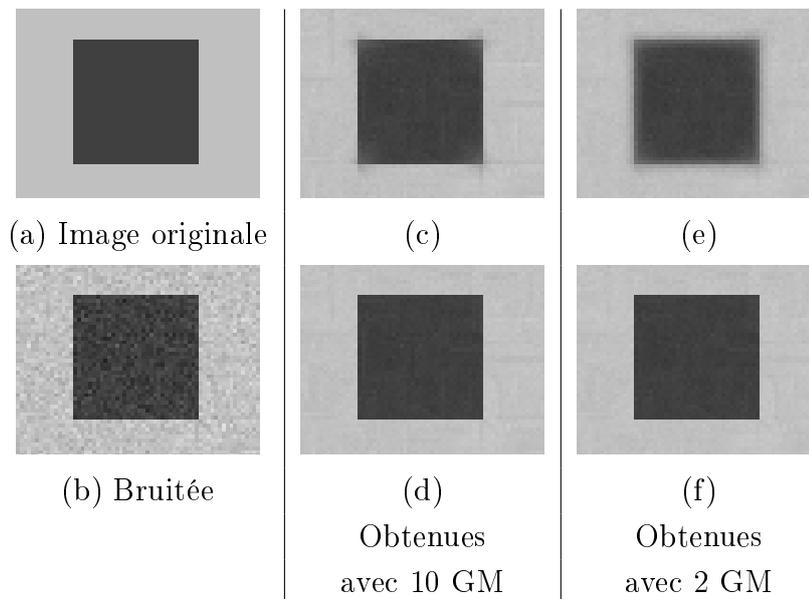


FIGURE 3.8 – Les figures (c-e) représentent les images débruitées sans invariance par translation et les figures (d-f), les images débruitées avec invariance par translation.

On observe sur la figure 3.8 que la méthode invariante par translation produit de bons résultats avec les deux GMM tandis que l'EPLL classique ne parvient pas à d'aussi bonnes performances. Sur le GMM à dix composantes où les coins du carré ne sont pas encodés, l'EPLL classique ne parvient pas à les restaurer. Sur le GMM à deux composantes, lorsque l'invariance par translation n'est pas appliquée, toutes les positions des transitions horizontales et verticales ne sont pas représentées, le carré est alors flou.

Remarquons que la performance du SI-EPLL est aussi bonne avec deux qu'avec dix composantes, ce qui laisse supposer qu'il est possible d'apprendre des GMM de plus petites tailles comme perspective. Nous appliquerons cette méthode d'agrégation de patches avec invariance par translation aux images radar en section 4.3.

3.3 Approches invariantes aux transformations radiométriques affines

Après l'étude de l'état de l'art des méthodes invariantes aux transformations radiométriques affines en section 3.3.1, nous proposons de modéliser la distribution des patches avec un GMM invariant affine en section 3.3.2. Ensuite, en section 3.3.3, nous décrivons un algorithme d'apprentissage d'une version simplifiée d'un GMM invariant aux transformations radiométriques affines dit GMM aligné. Nous présentons une application au débruitage utilisant le GMM aligné aux paragraphes 3.3.4 et 3.3.5 et évaluerons ses performances au paragraphe 3.3.6. En montrant que cette méthode produit des résultats satisfaisants, nous montrons qu'un GMM traitant un patch et son transformé radiométrique de la même façon permettrait à terme de mieux modéliser la distribution des patches et d'apprendre des GMM de plus petite taille.

3.3.1 État de l'art des approches invariantes aux transformations radiométriques affines

Comme il est mentionné dans [Alexander *et al.* (2008)], les images naturelles en plus d'être auto-similaires, présentent aussi une auto-similarité affine locale, ce qui signifie qu'un patch \mathbf{x}_1 extrait d'une image peut être proche, au sens de la distance euclidienne par exemple, d'un patch \mathbf{x}_2 à une transformation affine près : $\mathbf{x}_1 \simeq \alpha \mathbf{x}_2 + \beta \mathbf{1}$ où α et β sont les paramètres scalaires de cette transformation affine. Pourtant peu de travaux sont dédiés à l'élaboration de méthodes invariantes aux transformations radiométriques affines. Citons par exemple une extension de l'algorithme NL-Means avec une propriété d'invariance au changement de contraste proposée dans [Postec (2012)]. L'auteur observe que corriger le contraste de pixels similaires améliore légèrement les performances en débruitage au niveau des zones homogènes. Néanmoins, des méthodes et des critères existants sont intrinsèquement munis de cette invariance comme une corrélation normalisée (cf. sec.3.4.2)

et les approches par dictionnaires à condition qu'ils contiennent l'atome constant. On le démontre ci-dessous avec les équations (3.12) à (3.14) en prouvant que le patch \mathbf{x}_j et sa version transformée : $\alpha\mathbf{x}_j + \beta\mathbf{1}$ sont représentés par un même dictionnaire : $\begin{pmatrix} \mathbf{D} & \mathbf{1} \end{pmatrix} = \{\mathbf{d}_1, \dots, \mathbf{d}_{K-1}, \mathbf{1}\}$ où les atomes \mathbf{d}_k sont sous forme de colonnes et $\mathbf{1}$ est l'atome constant.

$$\operatorname{argmin}_{\mathbf{D}, \mathbf{u}, v} \left\| \mathbf{x}_j - \begin{pmatrix} \mathbf{D} & \mathbf{1} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ v \end{pmatrix} \right\|_2^2 \quad \text{t.q. } \forall k \|\mathbf{d}_k\| = 1 \quad (3.12)$$

$$= \operatorname{argmin}_{\mathbf{D}, \mathbf{u}, v} \left\| \alpha\mathbf{x}_j + \beta\mathbf{1} - \beta\mathbf{1} - \begin{pmatrix} \mathbf{D} & \mathbf{1} \end{pmatrix} \begin{pmatrix} \alpha\mathbf{u} \\ \alpha v \end{pmatrix} \right\|_2^2 \quad \text{t.q. } \forall k \|\mathbf{d}_k\| = 1 \quad (3.13)$$

$$= \operatorname{argmin}_{\mathbf{D}, \mathbf{u}, v} \left\| \alpha\mathbf{x}_j + \beta\mathbf{1} - \begin{pmatrix} \mathbf{D} & \mathbf{1} \end{pmatrix} \begin{pmatrix} \alpha\mathbf{u} \\ \alpha v + \beta \end{pmatrix} \right\|_2^2 \quad \text{t.q. } \forall k \|\mathbf{d}_k\| = 1 \quad (3.14)$$

où $\begin{pmatrix} \mathbf{u} \\ v \end{pmatrix}$ sont les coefficients parcimonieux avec \mathbf{u} les coefficients associés aux $K - 1$ premiers atomes et v le coefficient associé à l'atome constant $\mathbf{1}$.

On constate effectivement que les patches \mathbf{x}_j et $\alpha\mathbf{x}_j + \beta\mathbf{1}$ sont représentés par le même dictionnaire $\begin{pmatrix} \mathbf{D} & \mathbf{1} \end{pmatrix}$.

Les approches par GMM présentant des résultats performants dans plusieurs applications de traitement d'images, nous proposons dans la suite une méthode pour apprendre un GMM pourvu de cette invariance compte tenu du fait que les approches basées sur des GMM classiques ne sont pas intrinsèquement invariantes par transformation affine.

3.3.2 Modéliser une approche par GMM avec invariance affine

La modélisation de la distribution des patches par des modèles Gaussiens dans diverses applications de traitement d'images et en particulier de débruitage a prouvé son efficacité dans de nombreux travaux comme [Zoran et Weiss (2011)], [Yu *et al.* (2012)], [Lebrun *et al.* (2013)]. Nous décrivons dans la suite nos contributions qui mèneront à l'apprentissage d'un GMM invariant aux transformations radiométriques affines. En effet, l'expérience présentée en figure 3.9 montre que l'EPLL est sensible aux transformations radiométriques affines car ce ne sont pas les mêmes modèles qui représentent les patches de l'image d'un changement de contraste à l'autre. Apprendre un GMM qui traiterait un patch ou son transformé affine de la même façon, dans une application au débruitage par exemple, permettrait à terme d'apprendre des GMM contenant moins de composantes et donc d'accélérer l'apprentissage et le débruitage.

Dans la suite de cette section nous expliquons l'approche que nous proposons pour modéliser statistiquement un GMM avec invariance radiométrique. Pour cela, on décompose

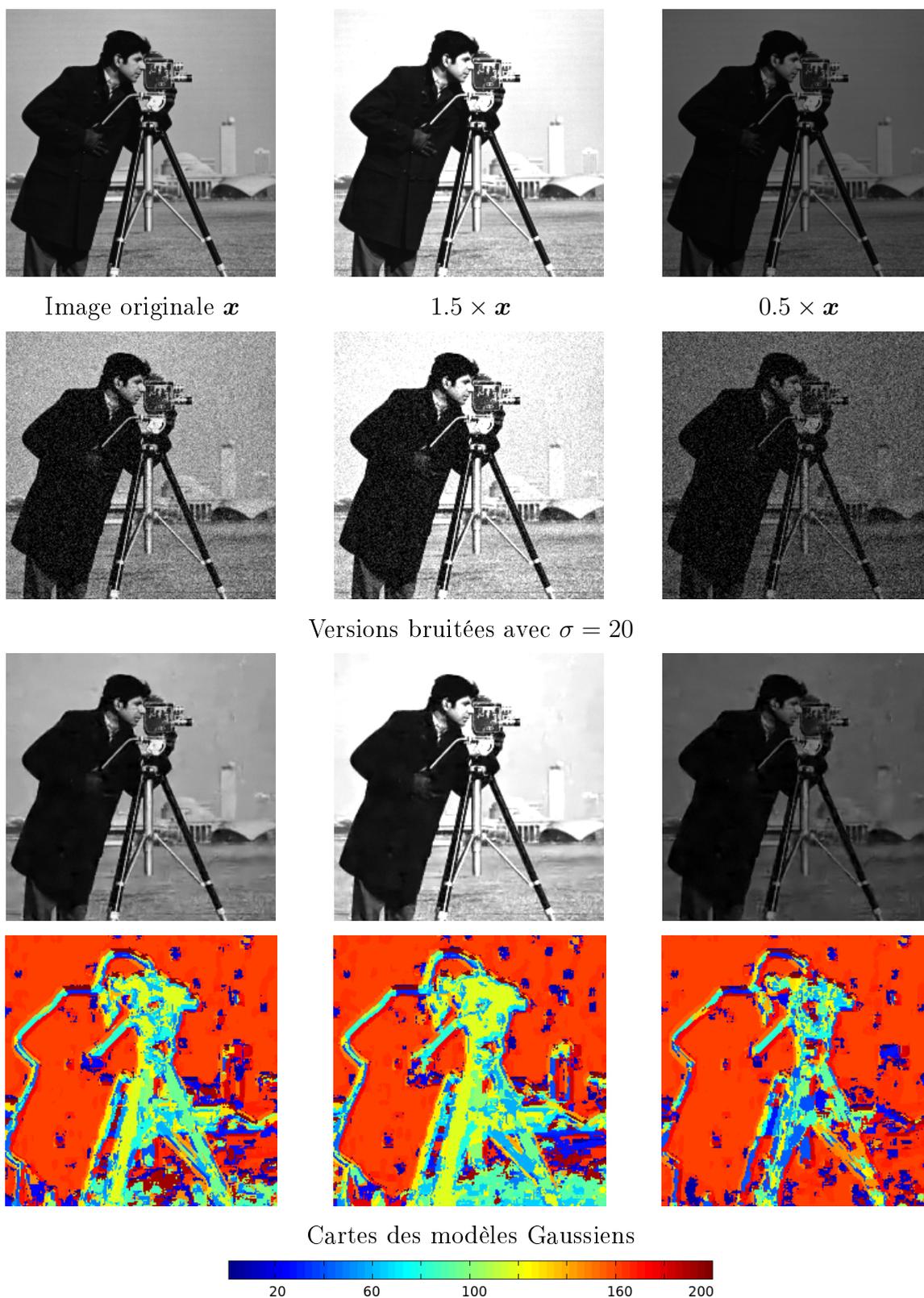


FIGURE 3.9 – Effet du changement de contraste sur la carte des modèles Gaussiens utilisés lors d'un débruitage avec l'EPLL. On note que les modèles Gaussiens utilisés ne sont pas les mêmes dans plusieurs zones de l'image, en particulier au niveau du pied de la caméra.

un patch tiré d'une image sans bruit de cette façon : $\mathbf{x} = \alpha\boldsymbol{\eta}$, où $\boldsymbol{\eta}$ suit une distribution correspondant à un mélange de Gaussiennes dont les paramètres sont représentés par $\boldsymbol{\theta}$:

$$p(\boldsymbol{\eta}|\boldsymbol{\theta}) = \sum_{k=1}^K \omega_k \frac{1}{(2\pi)^{d/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{\eta} - \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1}(\boldsymbol{\eta} - \boldsymbol{\mu}_k)\right) \quad (3.15)$$

où $\boldsymbol{\theta} = \{\omega_k, \mathbf{C}_k, \boldsymbol{\mu}_k\}$ et \mathbf{x} est sous forme d'une colonne de taille $d \times 1$.

La distribution de \mathbf{x} s'écrit alors :

$$p(\mathbf{x}|\alpha, \boldsymbol{\theta}) = \sum_{k=1}^K \omega_k \frac{1}{(2\pi)^{d/2} |\alpha|^d \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}\left(\frac{\mathbf{x}}{\alpha} - \boldsymbol{\mu}_k\right)^t \mathbf{C}_k^{-1}\left(\frac{\mathbf{x}}{\alpha} - \boldsymbol{\mu}_k\right)\right) \quad (3.16)$$

$$= \sum_{k=1}^K \omega_k \frac{1}{(2\pi)^{d/2} \det(\alpha^2 \mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \alpha\boldsymbol{\mu}_k)^t (\alpha^2 \mathbf{C}_k)^{-1}(\mathbf{x} - \alpha\boldsymbol{\mu}_k)\right) \quad (3.17)$$

Remarquons que pour simplifier le modèle, nous modélisons des patches centrés. Nous n'avons donc pas représenté un patch de cette manière : $\mathbf{x} = \alpha\boldsymbol{\eta} + \beta\mathbf{1}$. Dans le cas d'un patch constant, cela rend la distribution dégénérée et la matrice de covariance non inversible. Lors de l'apprentissage, ce problème est résolu en ajoutant à la diagonale de la matrice de covariance avec une constante de faible valeur.

Considérons à présent une observation bruitée du patch \mathbf{x} notée \mathbf{y} . On a dans le cas du bruit blanc Gaussien :

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{d/2} \sigma^d} \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{x}\|^2\right) \quad (3.18)$$

La distribution de \mathbf{y} s'écrit :

$$p(\mathbf{y}|\alpha, \boldsymbol{\theta}) = \sum_{k=1}^K \omega_k \frac{1}{(2\pi)^{d/2} |\alpha^2 \mathbf{C}_k + \sigma^2 \mathbf{I}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \alpha\boldsymbol{\mu}_k)^t (\alpha^2 \mathbf{C}_k + \sigma^2 \mathbf{I})^{-1}(\mathbf{y} - \alpha\boldsymbol{\mu}_k)\right) \quad (3.19)$$

Dans ce manuscrit nous traiterons d'une version simplifiée de GMM invariant radio-métriquement en apprenant un GMM qu'on appellera aligned-GMM (aGMM) ou GMM aligné en français. Cette approche permet de traiter un patch ou sa version au contraste inversé avec les mêmes modèles Gaussiens. Le paramètre de transformation affine α ne prend donc que les valeurs 1 ou -1 . L'apprentissage de l'aGMM est décrit en section 3.3.3, tandis que la procédure de débruitage associée est décrite en section 3.3.4. Enfin, le paragraphe 3.3.6 décrit les études expérimentales menées afin d'évaluer les performances de l'aGMM.

3.3.3 Apprentissage d'un GMM aligné (aGMM)

Dans l'EPLL [Zoran et Weiss (2011)], les moyennes du GMM appris sont mises à zéro. En plus d'avoir appris ce GMM sur une base de patchs centrés, ce qui entraîne que les moyennes de tels modèles Gaussiens sont faibles, ce choix se justifie du fait qu'un patch \mathbf{x}_i et son inverse de contraste $-\mathbf{x}_i$ sont équiprobables comme le montre ce calcul simplifié comparant l'opposé de la log-vraisemblance d'une Gaussienne de moyenne nulle entre \mathbf{x}_i et $-\mathbf{x}_i$ en équation (3.20) :

$$\mathbf{x}_i^t \mathbf{C}_k \mathbf{x}_i = (-1)^2 \mathbf{x}_i^t \mathbf{C}_k^{-1} \mathbf{x}_i = (-\mathbf{x}_i^t) \mathbf{C}_k^{-1} (-\mathbf{x}_i) \quad (3.20)$$

Dans cette sous-section, nous décrivons la procédure d'apprentissage d'un GMM de moyennes non nulles dit aligné. Nous définissons l'alignement par l'inversion du contraste de certains patchs afin que tous les patchs représentant une même structure à une inversion de contraste près soient proches d'un même vecteur moyen $\boldsymbol{\mu}$. Avec la représentation proposée en section 3.3.2 où : $\mathbf{x} = \alpha \boldsymbol{\eta}$, les valeurs possibles de α sont 1 et -1 .

À terme, notre objectif est que l'alignement, allié au fait d'apprendre un GMM de moyennes non nulles, permette de mieux modéliser la distribution des patchs et nous indique qu'apprendre un GMM invariant radiométrique serait fructueux à l'avenir.

Dans la section 2.4.3.1, nous avons décrit l'algorithme EM (Expectation-Maximization) [Dempster *et al.* (1977)] utilisé pour apprendre un GMM sur une base de patchs $\{\mathbf{x}_i\}_{i \in \{1, \dots, N\}}$. Nous rappelons que cet algorithme est itératif et a pour objectif de déterminer :

$$\hat{\Theta}^{(t+1)} = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^N \sum_{k=1}^K \gamma_{ik} \log(\omega_k f(\mathbf{x}_i, \boldsymbol{\theta}_k)) \quad (3.21)$$

où $\hat{\Theta}^{(t)} = \{\omega_k^{(t)}, \boldsymbol{\theta}_k^{(t)}\}$ est l'ensemble des paramètres des modèles Gaussiens à l'itération t avec $\boldsymbol{\theta}_k = \{\boldsymbol{\mu}_k, \mathbf{C}_k\}$ et où :

$$\gamma_{ik} = \frac{\omega_k^{(t)} f(\mathbf{x}_i, \boldsymbol{\theta}_k^{(t)})}{\sum_{k=1}^K \omega_k^{(t)} f(\mathbf{x}_i, \boldsymbol{\theta}_k^{(t)})}$$

défini précédemment en équation (2.43), représente la probabilité a posteriori pour que \mathbf{x}_i appartienne au k -ème modèle Gaussien sachant $\Theta^{(t)}$.

La différence avec un EM classique réside dans la définition de la fonction f . Soit f_α la fonction f adaptée à l'alignement :

$$f_\alpha(\mathbf{x}_i, \boldsymbol{\theta}_k) = \frac{1}{2\pi^{d^2/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \alpha \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i - \alpha \boldsymbol{\mu}_k)\right) \quad (3.22)$$

qui correspond à la modélisation présentée en équation (3.17) mais avec $\alpha = \{-1, 1\}$.

En plus d'estimer à chaque itération les paramètres Θ , avec l'alignement, on doit estimer également le paramètre $\alpha = \{-1, 1\}$ pour chaque patch de la base d'apprentissage et chaque composante du GMM. Le problème (3.21) adapté à l'alignement devient :

$$\hat{\Theta}^{(t+1)} = \underset{\Theta}{\operatorname{argmax}} \sum_{i=1}^N \sum_{k=1}^K \max\{\gamma_{ik,\alpha} \log(\omega_k f_1(\mathbf{x}_i, \boldsymbol{\theta}_k)), \gamma_{ik,\alpha} \log(\omega_k f_{-1}(\mathbf{x}_i, \boldsymbol{\theta}_k))\} \quad (3.23)$$

$$\text{où } \gamma_{ik,\alpha} = \frac{\omega_k^{(t)} f_{\alpha}(\mathbf{x}_i, \boldsymbol{\theta}_k^{(t)})}{\sum_{k=1}^K \omega_k^{(t)} f_{\alpha}(\mathbf{x}_i, \boldsymbol{\theta}_k^{(t)})}.$$

Soit $\hat{\alpha}_{ik}$ le paramètre d'inversion de contraste optimal associé au patch \mathbf{x}_i pour le k -ème modèle Gaussien, les paramètres de ces modèles Gaussiens s'écrivent alors :

$$\hat{\omega}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik, \hat{\alpha}_{ik}}, \quad \hat{\boldsymbol{\mu}}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik, \hat{\alpha}_{ik}} \frac{\mathbf{x}_i}{\hat{\alpha}_{ik}}, \quad \hat{\mathbf{C}}_k = \frac{1}{Z_k} \sum_{i=1}^N \gamma_{ik, \hat{\alpha}_{ik}} (\mathbf{x}_i - \hat{\alpha}_{ik} \boldsymbol{\mu}_k)(\mathbf{x}_i - \hat{\alpha}_{ik} \boldsymbol{\mu}_k)^t \quad (3.24)$$

où $Z_k = \sum_{i=1}^N \gamma_{ik, \hat{\alpha}_{ik}}$ est une constante de normalisation.

Nous évaluerons les performances d'un GMM aligné dans une application au débruitage en section 3.3.6. Avant cela, nous présentons dans la section suivante comment adapter une approche de débruitage par GMM à l'alignement.

3.3.4 Débruitage par l'aGMM

Une fois qu'un GMM aligné est appris en utilisant l'algorithme EM présenté en section 3.3.3, on peut l'utiliser pour réaliser une tâche de débruitage qu'on abrègera par aGMM-D, pour aGMM Denoising. Le problème du débruitage d'un patch extrait d'une image et centré s'écrit comme suit :

$$\hat{\mathbf{x}}_i = \underset{\mathbf{x}_i}{\operatorname{argmax}} p(\mathbf{x}_i | \alpha, \mathbf{y}_i, \boldsymbol{\theta}) = \underset{\mathbf{x}_i}{\operatorname{argmax}} p(\mathbf{y}_i | \mathbf{x}_i) p(\mathbf{x}_i | \alpha, \boldsymbol{\theta}) \quad (3.25)$$

En passant à l'opposé de la log-vraisemblance, dans le cas d'un bruit Gaussien et en utilisant l'a priori du GMM aligné de l'équation (3.17) on obtient :

$$\hat{\mathbf{x}}_i = \underset{\mathbf{x}_i}{\operatorname{argmin}} \left(\frac{1}{\sigma^2} \|\mathbf{y}_i - \mathbf{x}_i\|_2^2 + \min_k [-2 \log(\omega_k) + \log \det(\mathbf{C}_k) \right. \quad (3.26)$$

$$\left. + \min_{\alpha_i \in \{-1, 1\}} (\mathbf{x}_i - \alpha_i \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i - \alpha_i \boldsymbol{\mu}_k) \right)$$

$$= \underset{\mathbf{x}_i}{\operatorname{argmin}} \left(\frac{1}{\sigma^2} \|\mathbf{y}_i - \mathbf{x}_i\|_2^2 + \min_k [-2 \log(\omega_k) + \log \det(\mathbf{C}_k) \right. \quad (3.27)$$

$$\left. + \min\{(\mathbf{x}_i - \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k), (\mathbf{x}_i + \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{x}_i + \boldsymbol{\mu}_k)\} \right)$$

On commence par déterminer le modèle Gaussien et le paramètre d'inversion de contraste optimaux :

$$(k^*, \hat{\alpha}_i) = \underset{k, \alpha_i \in \{1, -1\}}{\operatorname{argmin}} \left(-\log(\omega_k) + \log \det(\mathbf{C}_k) + (\mathbf{x}_i - \alpha_i \boldsymbol{\mu}_k)^t \mathbf{C}_k (\mathbf{x}_i - \alpha_i \boldsymbol{\mu}_k) \right) \quad (3.28)$$

Le problème devient alors :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \left(\frac{1}{\sigma^2} \|\mathbf{y} - \mathbf{x}\|_2^2 + \sum_{i=1}^N (\mathbf{x}_i - \hat{\alpha}_i \boldsymbol{\mu}_{k^*})^t \mathbf{C}_{k^*}^{-1} (\mathbf{x}_i - \hat{\alpha}_i \boldsymbol{\mu}_{k^*}) \right) \quad (3.29)$$

où $\hat{\alpha}_i \in \{-1, 1\}$ est l'estimation du paramètre d'inversion de contraste associé au patch \mathbf{x}_i et k^* le modèle Gaussien lui correspondant le mieux. En mettant le gradient en \mathbf{x}_i de cette expression à zéro pour réaliser cette minimisation, on obtient :

$$\begin{aligned} \frac{1}{\sigma^2} (\mathbf{x}_i - \mathbf{y}_i) + \mathbf{C}_{k^*}^{-1} (\mathbf{x}_i - \hat{\alpha}_i \boldsymbol{\mu}_{k^*}) &= 0 \\ \Leftrightarrow \mathbf{x}_i &= \left(\frac{1}{\sigma^2} \mathbf{I} + \mathbf{C}_{k^*}^{-1} \right)^{-1} \left(\frac{1}{\sigma^2} \mathbf{y}_i + \mathbf{C}_{k^*}^{-1} \hat{\alpha}_i \boldsymbol{\mu}_{k^*} \right) \\ \Leftrightarrow \mathbf{x}_i &= (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} (\mathbf{C}_{k^*} \mathbf{y}_i + \sigma^2 \hat{\alpha}_i \boldsymbol{\mu}_{k^*}) \end{aligned} \quad (3.30)$$

En débruitant tous les patches extraits d'une image bruitée de cette manière, on agrège ces derniers en les moyennant. Remarquons que l'EPLL est un cas particulier de cette approche avec un GMM de moyenne nulle. On peut donc utiliser la même architecture que l'EPLL pour débruiter une image avec une approche par GMM aligné. Nous décrivons cet algorithme dans la section suivante.

3.3.5 L'aEPLL : EPLL + aGMM

Nous adaptons dans cette section l'EPLL afin d'utiliser un GMM aligné. Rappelons le problème original de l'EPLL :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \min_{\mathbf{z}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 - \sum_{i=1}^N \log p(\mathbf{z}_i) + \frac{\delta}{2} \sum_{i=1}^N \|\mathbf{S}_i \mathbf{R}_i \mathbf{x} - \mathbf{z}_i\|^2 \quad (3.31)$$

où \mathbf{z}_i est une variable auxillaire proche de $\mathbf{S}_i \mathbf{R}_i \mathbf{x}$, avec \mathbf{R}_i l'opérateur d'extraction du i -ème patch d'une image et \mathbf{S}_i l'opérateur qui retire la moyenne du i -ème patch d'une image. On résout ce problème de façon itérative avec une valeur croissante de δ proportionnelle à $1/\sigma^2$. Pour cela, on alterne l'optimisation en \mathbf{x} et en \mathbf{z} .

Le problème en \mathbf{z} s'écrit :

$$\hat{\mathbf{z}}_i = \underset{\mathbf{z}_i}{\operatorname{argmin}} \frac{\delta}{2} \sum_{i=1}^N \|\mathbf{S}_i \mathbf{R}_i \mathbf{x}^{(t)} - \mathbf{z}_i\|^2 - \log p(\mathbf{z}_i) \quad (3.32)$$

où l'a priori aGMM s'écrit comme suit :

$$-\log p(\mathbf{z}_i) = \min_k -2\log(\omega_k) + \log \det(\mathbf{C}_k) + \min_{\alpha_i \in \{-1,1\}} (\mathbf{z}_i - \alpha_i \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{z}_i - \alpha_i \boldsymbol{\mu}_k) \quad (3.33)$$

$$= \min_k -2\log(\omega_k) + \log \det(\mathbf{C}_k) + \min\{(\mathbf{z}_i - \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{z}_i - \boldsymbol{\mu}_k), (\mathbf{z}_i + \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{z}_i + \boldsymbol{\mu}_k)\}$$

Le problème (3.32) est résolu en suivant les étapes décrites dans la section 3.3.4, en particulier en l'équation (3.30). On obtient alors :

$$\begin{aligned} \hat{\mathbf{z}}_i &= (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} (\hat{\alpha}_i \mathbf{C}_{k^*} \bar{\mathbf{x}}_i + \sigma^2 \hat{\alpha}_i \boldsymbol{\mu}_{k^*}) \\ &= \hat{\alpha}_i (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} (\mathbf{C}_{k^*} \bar{\mathbf{x}}_i + \sigma^2 \boldsymbol{\mu}_{k^*}) \end{aligned} \quad (3.34)$$

où k^* et $\hat{\alpha}_i$ sont l'indice du modèle Gaussien et le paramètre d'inversion de contraste optimaux du patch d'intérêt respectivement et $\bar{\mathbf{x}}_i$ est le patch \mathbf{x}_i centré.

En introduisant les opérateurs \mathbf{R} qui extrait tous les patches de l'image et \mathbf{S} qui centre tous les patches, le problème en \mathbf{x} s'écrit :

$$\begin{aligned} \hat{\mathbf{x}} &= \underset{\mathbf{x}}{\operatorname{argmin}} \frac{\lambda}{2} \|\mathbf{x} - \mathbf{y}\|^2 + \frac{\delta}{2} \sum_{i=1}^N \|\mathbf{S}_i \mathbf{R}_i \mathbf{x} - \hat{\mathbf{z}}_i\|^2 \\ &= (\lambda \mathbf{I} + \delta \mathbf{R}^t \mathbf{S} \mathbf{R})^{-1} (\lambda \mathbf{y} + \delta \mathbf{R}^t \mathbf{S} \hat{\mathbf{z}}) \end{aligned} \quad (3.35)$$

où l'inverse de $(\mathbf{I} + \sigma^2 \delta \mathbf{R}^t \mathbf{S} \mathbf{R})$ peut être calculé par gradient conjugué [Hestenes et Stiefel (1952)]. Notons que \mathbf{S} est un projecteur orthogonal et : $\mathbf{S}^2 = \mathbf{S}^t = \mathbf{S}$. L'algorithme complet est résumé dans le pseudo-code 3.

Algorithm 3 aEPLL

Require: N_{pa} : nombre de patches

Initialisation : $\mathbf{x}_i = \mathbf{y}_i, \forall i \in \{1, \dots, N_{\text{pa}}\}$

for $\delta = [1, 4, 8]/\sigma^2$ **do**

for $i \in \{1, \dots, N_{\text{pa}}\}$ **do**

$$(k^*, \hat{\alpha}) = \underset{k, \alpha}{\operatorname{argmin}} -2\log(\omega_k) + \log \det(\mathbf{C}_k) + (\mathbf{z}_i - \alpha \boldsymbol{\mu}_k)^t \mathbf{C}_k^{-1} (\mathbf{z}_i - \alpha \boldsymbol{\mu}_k)$$

$$\mathbf{z}_i = \hat{\alpha}_i (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} (\mathbf{C}_{k^*} \bar{\mathbf{x}}_i + \sigma^2 \boldsymbol{\mu}_{k^*})$$

end for

 Mettre à jour \mathbf{x} avec l'équation (3.35)

end for

Nous verrons dans la section suivante des résultats de débruitages avec l'aEPLL et des comparaisons par rapport à l'EPLL classique.

3.3.6 Apport de l’alignement

Dans cette section nous évaluons les performances du GMM aligné. Tout d’abord, nous expliquons son protocole d’apprentissage. Ensuite, nous évaluons ses performances dans une tâche de débruitage et le comparons à un GMM sans alignement.

Protocole d’apprentissage L’algorithme d’apprentissage du GMM aligné est présenté en section 3.3.3. Nous avons appris des aGMM et des GMM classiques de différentes tailles sur une même base de patches tirés d’images naturelles que le lecteur peut consulter en annexe A. Les GMM non-alignés que nous avons appris sont de vecteurs moyens non nuls contrairement au GMM appris dans [Zoran et Weiss (2011)]. La figure 3.10 présente des patches générés à partir de modèles Gaussiens alignés (aGM) et leurs moyennes respectives.

Voici quelques détails sur l’implémentation de l’algorithme EM et les paramètres utilisés :

- Les différentes tailles de GMM sont : 1, 2, 4, 8, 16, 32, 64, 128, 200.
- Le nombre de patches dans la base d’apprentissage est de plus de $2 \cdot 10^6$.
- Le nombre d’itérations de l’algorithme EM est égal à 200.
- L’initialisation de l’algorithme EM est inspirée de celle de la procédure dite de K-Means++ [Arthur et Vassilvitskii (2007)] adaptée à l’alignement.

Comparaisons des performances en débruitage Dans ce paragraphe, nous évaluons l’apport de l’alignement dans une tâche de débruitage. Pour cela, nous comparons les performances de l’EPLL et de l’aEPLL en termes de PSNR et de SSIM sur les images de Lena et Barbara avec des GMM de différentes tailles. Les tableaux 3.1, 3.2 résument ces comparaisons sur l’image de Barbara avec des niveaux de bruit $\sigma = 20$ et 30 respectivement. On observe que le PSNR et le SSIM sont croissants en fonction de la taille des GMM. On observe également, quelque soit le niveau de bruit, que les performances de l’aEPLL et de l’EPLL sont proches pour des dictionnaires de petites tailles mais l’aEPLL devient meilleur quand la taille des GMM est grande. On le constate visuellement sur une portion de l’image de Barbara en figure 3.13. Les conclusions obtenues sur l’image de Lena étaient similaires. Dans les figures 3.11, 3.12, on présente des résultats de débruitages avec l’EPLL et l’aEPLL utilisant des GMM de 200 composantes avec $\sigma = 20$ et 30 respectivement sur Lena et Barbara. On remarque que l’écart entre l’EPLL et l’aEPLL est plus important quand l’image est très texturée comme celle de Barbara.

Ces résultats satisfaisants nous ont encouragé à apprendre récemment un GMM avec invariance radiométrique (où α est donc réel). En section 5.2, nous présenterons des résultats préliminaires obtenus avec une version de l’EPLL utilisant un GMM invariant radiométrique qui présente des performances meilleures que l’EPLL classique et l’aEPLL sur les images testées (du type de Lena) et permet d’apprendre des GMM de taille deux fois plus petite qu’un GMM classique.

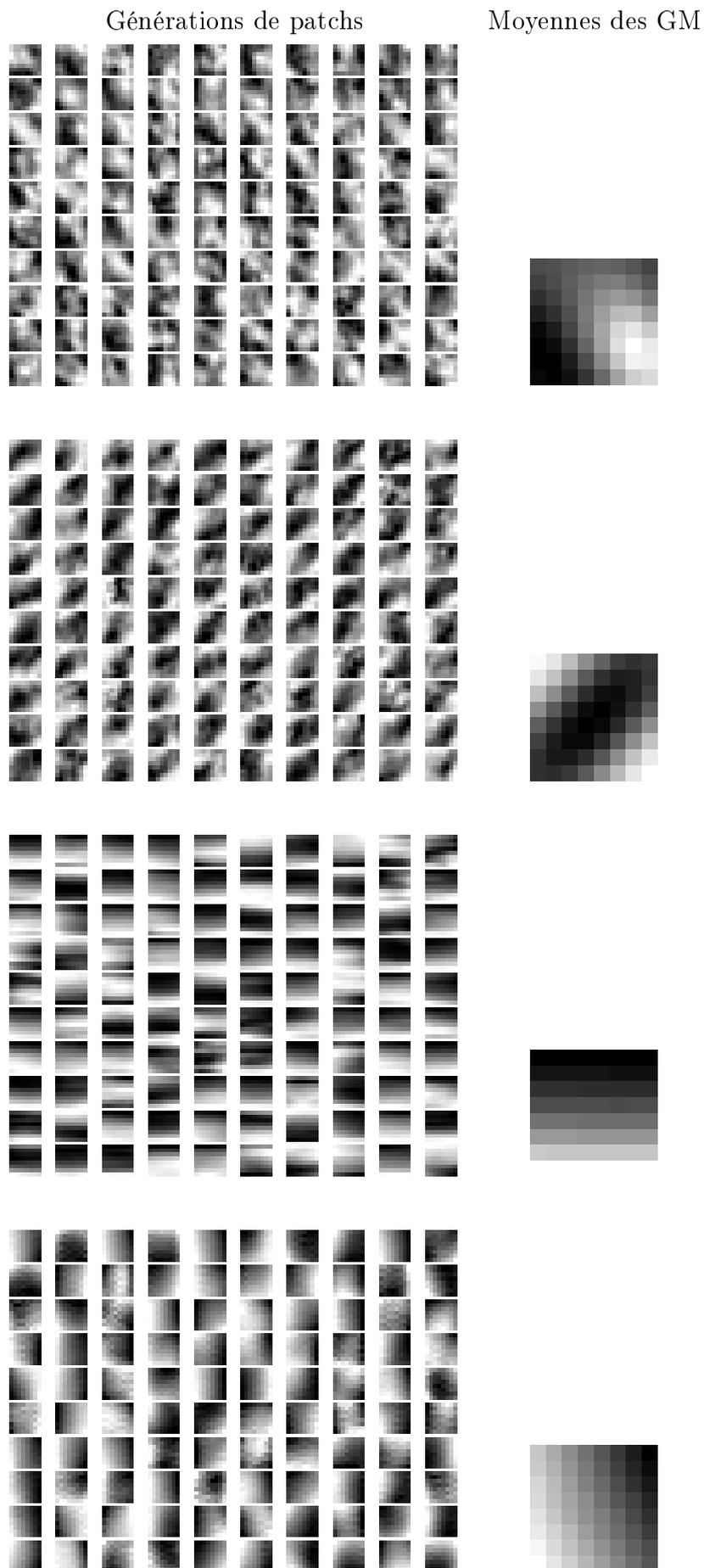


FIGURE 3.10 – Générations de patches et patch moyen de différents aGM

Taille des GMM		1	2	4	8	16	32	64	128	200
Barbara, $\sigma = 20$	sans	25.53	27.80	28.30	28.85	29.64	29.90	30.34	30.67	30.68
	avec	25.52	27.78	28.28	28.98	29.44	29.81	30.46	30.93	31.04
Barbara, $\sigma = 30$	sans	23.57	25.60	25.958	26.48	27.49	27.62	28.06	28.36	28.34
	avec	23.56	25.59	25.953	26.58	27.10	27.60	28.19	28.61	28.80

TABLE 3.1 – Tableau de PSNR comparant des débruitages par EPLL avec et sans alignement avec des GMM de différentes tailles sur l'image de Barbara.

Taille des GMM		1	2	4	8	16	32	64	128	200
Barbara, $\sigma = 20$	sans	0.797	0.893	0.907	0.914	0.924	0.929	0.935	0.93874	0.93872
	avec	0.796	0.837	0.908	0.918	0.923	0.931	0.939	0.945	0.946
Barbara, $\sigma = 30$	sans	0.7005	0.833	0.850	0.860	0.881	0.884	0.892	0.895	0.896
	avec	0.7001	0.834	0.852	0.863	0.874	0.887	0.897	0.906	0.911

TABLE 3.2 – Tableau de SSIM comparant des débruitages par EPLL avec et sans alignement avec des GMM de différentes tailles sur l'image de Barbara.

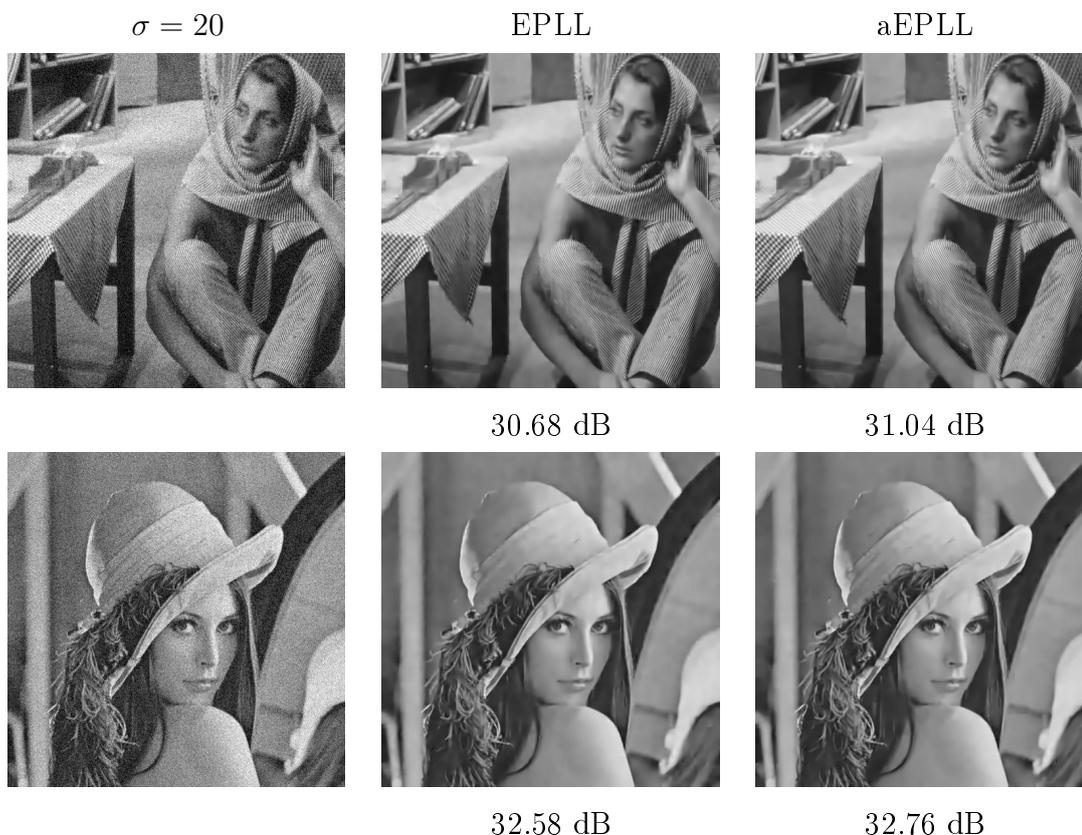


FIGURE 3.11 – Sur chaque ligne, de gauche à droite : image bruitée ($\sigma = 20$) et résultats de débruitages avec 200 GM par l'EPLL et l'aEPLL. En terme de PSNR, avec cette taille de GMM, l'aEPLL présente autour de 0.2 dB supplémentaires par rapport à l'EPLL.

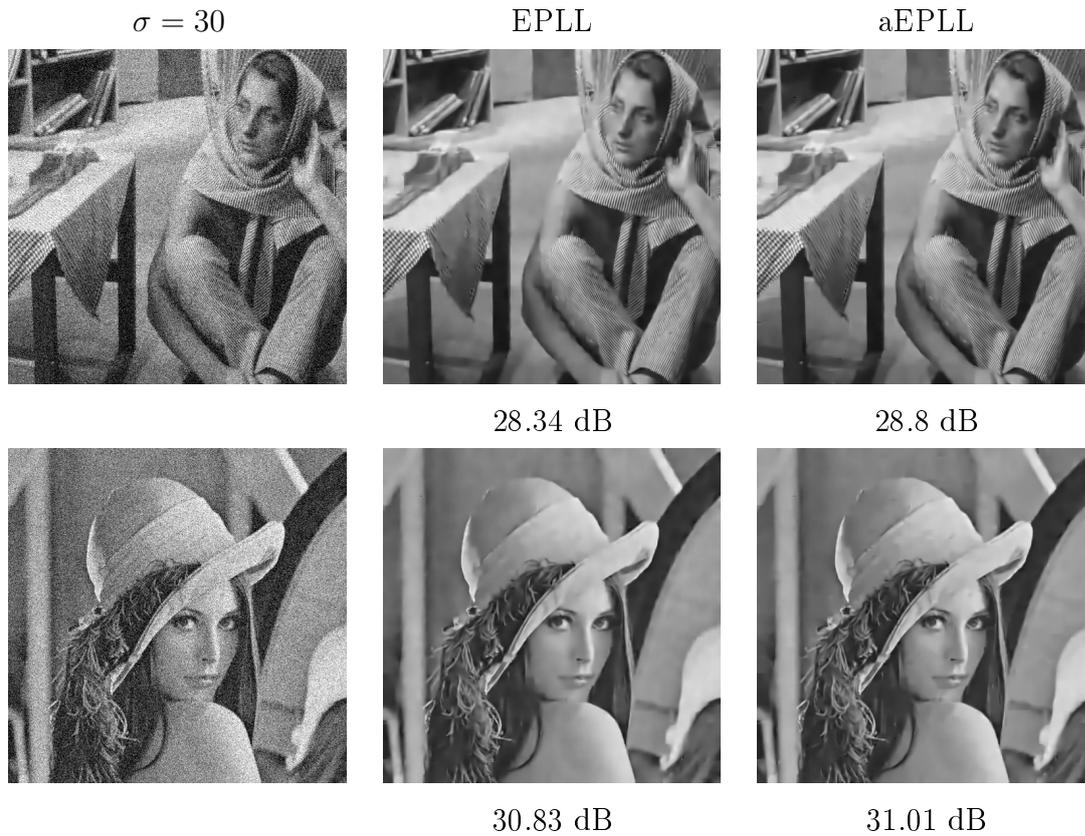


FIGURE 3.12 – Sur chaque ligne, de gauche à droite : image bruitée ($\sigma = 30$) et résultats de débruitages avec 200GM par l'EPLL et l'aEPLL. L'aEPLL présente une amélioration plus notable par rapport à l'EPLL sur l'image de Barbara qui est plus texturée que l'image de Lena.

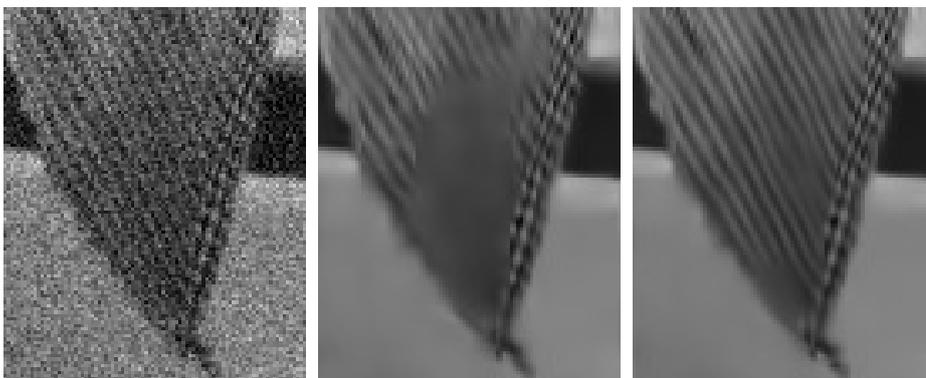


FIGURE 3.13 – De gauche à droite, zoom sur : l'image bruitée de Barbara avec $\sigma = 20$, les débruitages de Barbara avec 200 GM par l'EPLL et l'aEPLL. On observe qu'avec l'EPLL une partie de la texture de la nappe a été totalement lissée tandis qu'avec l'aEPLL elle est bien reconstruite.

3.4 Approches 1-sparse invariantes affines et par translation

Cette section détaille nos contributions au niveau de critères basés sur des modèles parcimonieux munis d'invariances par translation et transformation affine. Les deux approches proposées sont 1-sparse et nous justifions ce point en paragraphe 3.4.1. À terme, nous utiliserons ces méthodes pour apprendre des dictionnaires de patches. La première approche proposée est d'utiliser un dictionnaire avec des atomes de taille plus grande que les patches d'intérêt. La seconde, étend un GLR-test invariant par transformation affine proposé par [Deledalle *et al.* (2013)] et y introduit l'invariance par translation en pavant l'image avec des atomes qui la représentent au mieux.

3.4.1 Étude du degré de parcimonie optimal sous fort bruit

La section 2.4.2 présente le principe des représentations parcimonieuses et quelques méthodes d'apprentissage de dictionnaires. Nous nous proposons ici un protocole expérimental afin d'estimer le degré de parcimonie (voir la description de l'équation (2.27)) optimal dans un contexte de débruitage sous un fort bruit tel qu'on peut l'observer dans des images radar par exemple. Dans [Chainais (2012)], une extension de l'algorithme K-means pour apprendre un dictionnaire 1-sparse dédié aux images radar est proposée. L'hypothèse de 1-sparse-coding est présentée comme une première étape vers l'apprentissage de dictionnaires sur des données avec un bruit non gaussien. Nous proposons ici une justification complémentaire fondée sur une étude expérimentale.

On commence par apprendre un dictionnaire contenant 256 atomes avec l'algorithme K-SVD sur le logarithme de l'intensité d'une image radar 100-vues (donc quasiment sans bruit) afin que K-SVD soit adapté à la forte dynamique des images radar. Ensuite, de nombreux patches sont extraits de plusieurs versions de cette image bruitée par un bruit gamma 1-vue (cf. section 2.1.2 pour la définition du bruit gamma) et débruités en étant décomposés parcimonieusement par une OMP avec différents degrés de parcimonie. Cela nous permet d'évaluer quel degré présentera la plus faible erreur d'approximation, ici un EQM (Ecart Quadratique Moyen) par rapport aux patches 100-vues.

Les distributions des EQM des estimations obtenues avec différents degrés de parcimonie sont présentées en figure 3.14. On observe que la parcimonie égale à 1 produit en moyenne le plus faible EQM. En effet, avec un fort niveau de bruit, un grand degré de parcimonie donne un grand nombre de degrés de liberté et conduit à de plus fortes variances.

Vu les résultats de cette étude, on peut supposer qu'une approche 1-sparse avec des propriétés d'invariance qui compenseraient le biais dû à ce faible degré de parcimonie serait

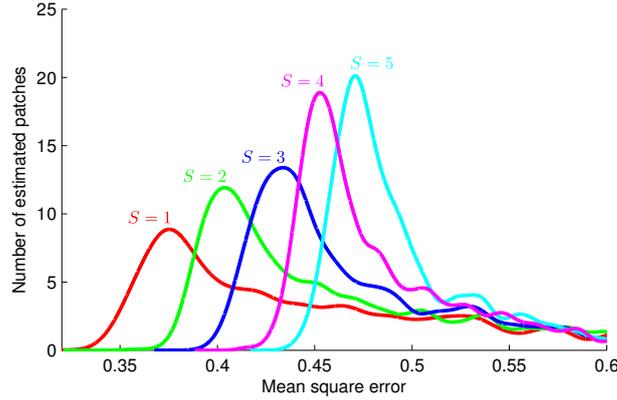


FIGURE 3.14 – Distribution des EQM en fonction du degré de parcimonie S . L'EQM augmente avec le nombre d'atomes utilisés dans la combinaison linéaire des patches hautement bruités.

efficace pour des applications au débruitage par exemple. Nous verrons si ces suppositions sont vérifiées au paragraphe 4.2.2 du prochain chapitre.

3.4.2 Un critère de similarité entre patches invariant par translation et transformation radiométrique affine

Nous proposons dans ce paragraphe un critère invariant par translation et transformation affine comparant un patch à un atome de plus grande taille tiré d'un dictionnaire déjà construit. Nous utiliserons ce critère dans une application au débruitage. L'approche proposée consiste à sélectionner la portion d'atome qui représente le mieux le patch considéré. Pour cela, on calcule le paramètre de translation 2D entière optimal en localisant le maximum d'une corrélation normalisée entre le patch et l'atome. Ainsi, on parvient à découper l'atome pour qu'il corresponde au mieux au patch. Cette procédure est proche de l'algorithme MoTIF [Jost *et al.* (2006)] à cela près que les atomes sont plus grands que les patches et non le contraire. Le problème associé s'écrit comme suit pour un patch \mathbf{x}_j , qu'on note $\eta(\mathbf{x}_j)$ une fois centré et normé :

$$\operatorname{argmin}_{\mathbf{d}_i, \mathbf{p}_j, (\alpha, \beta)} \|\alpha \mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i + \beta \mathbf{1} - \eta(\mathbf{x}_j)\|_2^2 \quad (3.36)$$

où \mathbf{d}_i est un atome centré et normé, \mathbf{T} est l'opérateur de translation 2D entière, \mathbf{p}_j le paramètre de translation 2D, α, β les paramètres de la transformation affine, $\mathbf{1}$ le vecteur dont toutes les entrées sont égales à 1 et N le nombre de patches.

Dans un premier temps, on résout le problème en \mathbf{d}_i et en \mathbf{p}_j qui concerne l'invariance par translation :

$$\operatorname{argmin}_{\mathbf{d}_i, \mathbf{p}_j} \|\mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i - \eta(\mathbf{x}_j)\|_2^2 \quad (3.37)$$

qui une fois développé s'écrit :

$$\operatorname{argmax}_{\mathbf{d}_i, \mathbf{p}_j} \frac{\langle \mathbf{T}_{-\mathbf{p}_j} \eta(\mathbf{x}_j), \mathbf{d}_i \rangle}{\sqrt{\|\mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i\|_2^2 - (w \overline{\mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i})^2}} \quad (3.38)$$

où un patch de l'image est de taille $w \times w$ et $\overline{\mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i}$ est la moyenne de $\mathbf{T}_{\mathbf{p}_j} \mathbf{d}_i$.

Ce problème revient à maximiser une corrélation normalisée qui se calcule rapidement par FFT (Fast Fourier Transform). Une fois qu'on a la solution de ce problème on peut découper le meilleur atome représentant le patch d'intérêt grâce au paramètre de translation optimal et noter cet atome découpé : $\tilde{\mathbf{d}}_i$.

Ensuite, on résout pour chaque patch le problème en (α, β) qui apporte l'invariance aux transformations radiométriques affines :

$$\min_{(\alpha, \beta)} \|\alpha \tilde{\mathbf{d}}_i + \beta \mathbf{1} - \mathbf{x}_j\|_2^2 \quad (3.39)$$

Ce problème étant une régression linéaire, on obtient les expressions suivantes pour α et β :

$$\alpha = \frac{\langle \mathbf{x}_j - \bar{x}_j \mathbf{1}, \tilde{\mathbf{d}}_i - \tilde{d}_i \mathbf{1} \rangle}{\|\tilde{\mathbf{d}}_i - \tilde{d}_i \mathbf{1}\|_2^2} \quad \beta = \bar{x}_j - \alpha \tilde{d}_i \quad (3.40)$$

où \tilde{d}_i est la moyenne de $\tilde{\mathbf{d}}_i$ et \bar{x}_j la moyenne de \mathbf{x}_j .

Application au débruitage : Nous avons appliqué ce critère à une tâche de débruitage. Pour cela, on utilise un dictionnaire prédéfini (fixe) présenté en figure 3.15. On l'obtient en générant une image à l'aide d'un modèle de feuilles mortes transparentes [Galerie et Gousseau (2012)] avec diverses formes géométriques (lignes, carrés et ronds) à des positions aléatoires, puis en extrayant des patches qui constituent les atomes. L'algorithme de débruitage se résume selon les étapes suivantes :

- On extrait les patches bruités de l'image.
- Pour chaque patch, on résout le problème (3.36) et on remplace le patch bruité par l'atome ainsi découpé et dont la radiométrie a été adaptée grâce au couple (α, β) obtenu en (3.40).
- On calcule une moyenne pondérée des patches restaurés pour obtenir l'image débruitée.

On présente un exemple de résultat sur Lena dans la figure 3.16 et sur des images radar dans les figures 3.17 et 3.18. Pour adapter l'algorithme présenté aux images radar, nous avons simplement appliqué une transformation logarithmique à l'image bruitée. De plus, le dictionnaire radar contient des atomes supplémentaires comparé à l'autre dictionnaire avec quelques structures typiques telles que des cibles ou des lignes brillantes (en figure 3.15). On constate que le débruitage présente des performances en dessous

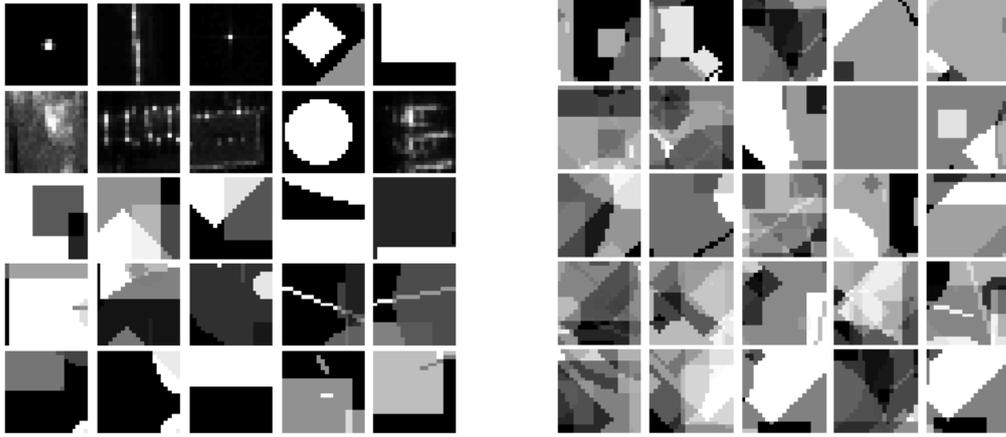


FIGURE 3.15 – À gauche : un exemple de dictionnaire utilisé pour traiter une image radar avec notre approche. Certains atomes sont extraits d’images radar (ex : cibles, lignes brillantes parallèles, unique ligne brillante). À droite : un dictionnaire issu uniquement du modèle de feuilles mortes transparentes de [Galerie et Gousseau (2012)]. Chaque dictionnaire contient 25 atomes de taille 21×21 .



FIGURE 3.16 – De gauche à droite : l’image originale, avec du bruit gaussien à écart-type = 20, débruitée avec l’approche proposée, PSNR : 23.65 dB et débruitée avec une OMP 1-sparse, PSNR : 18.42 dB.

de celles des méthodes actuelles. Cependant, comparé à une OMP 1-sparse (OMP en imposant un degré de parcimonie égal à 1), notre approche 1-sparse invariante présente un meilleur résultat visuel et en terme de PSNR.

Comme nous utilisons une corrélation normalisée avec des propriétés d’invariance, et qu’il a été montré dans [Deledalle *et al.* (2013)] que le GLR-test invariant aux transformations radiométriques est plus robuste au bruit qu’une corrélation normalisée, nous proposons d’améliorer et développer les idées du critère proposé dans le paragraphe 3.4.4. Avant cela nous décrivons dans la section 3.4.3 ce GLR-test invariant aux transformations radiométriques affines.

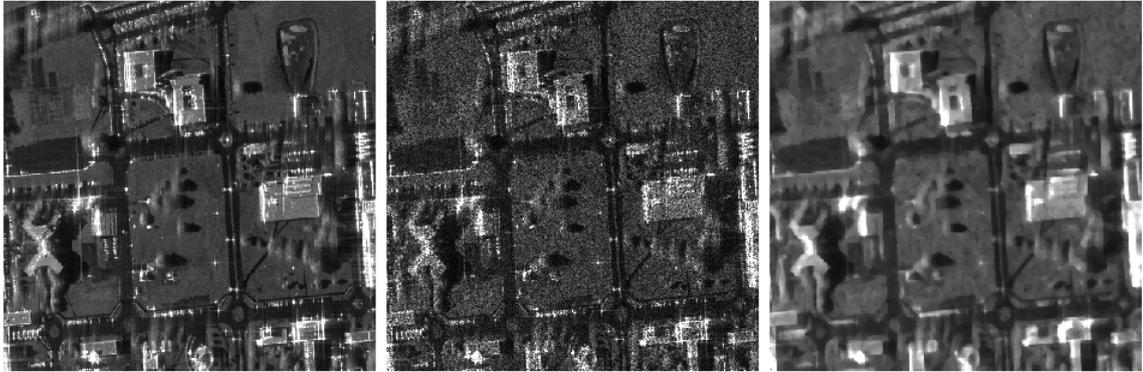


FIGURE 3.17 – De gauche à droite : l'image 100-vues, avec un bruit gamma 1-vue synthétique et le résultat de la méthode proposée.

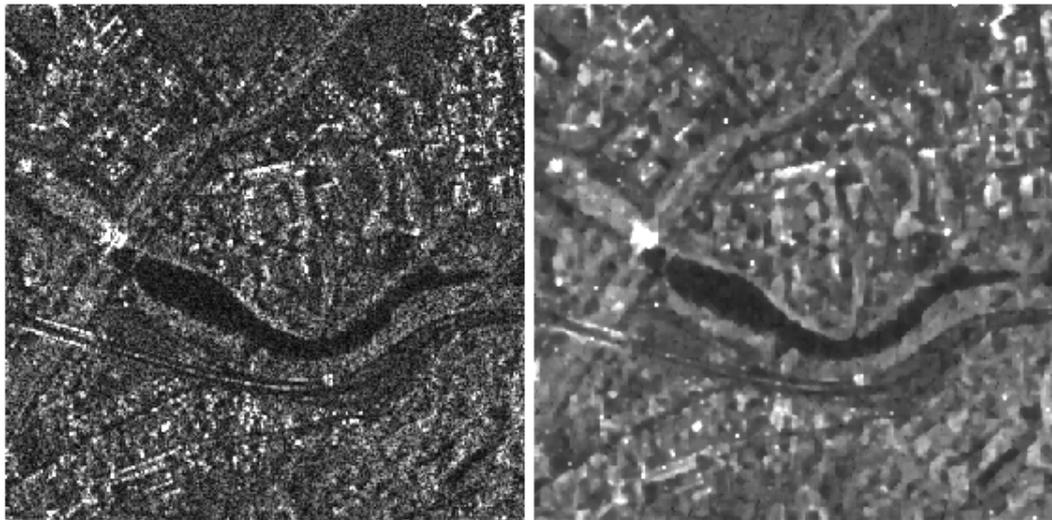


FIGURE 3.18 – De gauche à droite : l'image 1-vue de Saint-Gervais du satellite TerraSAR-X (mode stripmap $1m \times 2$) et débruitée avec notre approche.

3.4.3 Un GLR-test invariant affine

Dans [Deledalle *et al.* (2013)], un GLR-test (pour Generalized Likelihood Ratio test ou test de rapport de vraisemblance généralisé en français) comparant des patches bruités avec différents types de bruit à des atomes d'un dictionnaire à une transformation affine près est proposé. Pour un patch \mathbf{y}_j observé, \mathbf{x}_j sa version non bruitée et un atome \mathbf{d} d'un dictionnaire, on définit les hypothèses du GLR-test ainsi :

$$\begin{aligned} \mathcal{H}_0 : \exists \rho \quad \mathbf{x} &= \mathcal{T}_\rho(\mathbf{d}) && \text{(hypothèse nulle),} \\ \mathcal{H}_1 : \forall \rho \quad \mathbf{x} &\neq \mathcal{T}_\rho(\mathbf{d}) && \text{(hypothèse alternative).} \end{aligned} \quad (3.41)$$

où \mathcal{T}_ρ est une transformation affine appliquée à l'atome \mathbf{d} : $\mathcal{T}_\rho(\mathbf{d}) = \alpha\mathbf{d} + \beta\mathbf{1}$ avec le couple de paramètres scalaires $\rho = (\alpha, \beta)$ et $\mathbf{1}$ le vecteur dont toutes les entrées valent 1.

Le rapport est défini par l'expression :

$$\text{GLR}(\mathbf{y}_j, \mathbf{d}) = \frac{\sup_{\alpha, \beta} p(\mathbf{y}_j | \mathbf{x}_j = \mathcal{T}_\rho(\mathbf{d}), \mathcal{H}_0)}{\sup_{\mathbf{t}} p(\mathbf{y}_j | \mathbf{x}_j = \mathbf{t}, \mathcal{H}_1)} = \frac{p(\mathbf{y}_j | \mathbf{x}_j = \mathcal{T}_{\hat{\rho}}(\mathbf{d}))}{p(\mathbf{y}_j | \mathbf{x}_j = \hat{\mathbf{t}})} \quad (3.42)$$

où $\hat{\rho} = (\hat{\alpha}, \hat{\beta})$ and $\hat{\mathbf{t}}$ sont les estimateurs du maximum de vraisemblance des inconnus $\rho = (\alpha, \beta)$ et \mathbf{x}_j respectivement.

Dans le cadre d'un bruit gaussien de variance σ^2 , en calculant l'estimateur du maximum de vraisemblance de \mathbf{x}_j pour $\hat{\mathbf{t}} = \mathbf{y}_j$ on obtient :

$$-\log \text{GLR}(\mathbf{y}_j, \mathbf{d}) = \|\mathbf{y}_j - \hat{\alpha}\mathbf{d} - \hat{\beta}\mathbf{1}\|_2^2 / 2\sigma^2 \quad (3.43)$$

Par la suite, en injectant $\hat{\alpha}$ et $\hat{\beta}$ qui représentent les coefficients d'une régression linéaire (voir éq.(3.40)) on obtient :

$$-\log \text{GLR}(\mathbf{y}_j, \mathbf{d}) = \begin{cases} (1 - \mathcal{C}(\mathbf{y}_j, \mathbf{d})^2) \frac{\|\mathbf{y}_j - \bar{y}_j\mathbf{1}\|^2}{2\sigma^2} & \text{si } \mathbf{d} \neq \bar{d}\mathbf{1} \text{ et } \mathbf{y}_j \neq \bar{y}_j\mathbf{1} \\ \frac{\|\mathbf{y}_j - \bar{y}_j\mathbf{1}\|^2}{2\sigma^2} & \text{sinon} \end{cases} \quad (3.44)$$

$$\text{avec } \mathcal{C}(\mathbf{y}_j, \mathbf{d}) = \left| \frac{\sum_{k=1}^{w^2} (y_{k,j} - \bar{y}_j)(d_k - \bar{d})}{\sqrt{\sum_{k=1}^{w^2} (y_{k,j} - \bar{y}_j)^2 \sum_{k=1}^{w^2} (d_k - \bar{d})^2}} \right| \text{ la corrélation normalisée,}$$

où \bar{d} et \bar{y}_j sont les moyennes empiriques de \mathbf{d} et \mathbf{y}_j respectivement, w^2 est le nombre d'éléments d'un patch, $y_{k,j}$ (resp. d_k) est la k -ème valeur du patch \mathbf{y}_j (resp. \mathbf{d}).

Dans le cas où $\mathbf{d} = \bar{d}\mathbf{1}$, étant donné que \mathbf{d} est colinéaire à $\mathbf{1}$, alors $-\log \text{GLR}$ est l'infimum de $\gamma \mapsto \|\mathbf{y}_j - \gamma\mathbf{1}\|^2 / (2\sigma^2)$ atteint en : $\gamma = \bar{y}_j$. Quant au cas où $\mathbf{y}_j = \bar{y}_j\mathbf{1}$, le $-\log \text{GLR}$ est l'infimum de $(\alpha, \beta) \mapsto \|(\bar{y}_j - \beta)\mathbf{1} - \alpha\mathbf{y}_j\|^2 / (2\sigma^2)$ qui est atteint en $\alpha = 0$

et $\beta = \bar{y}_j$ et est égal à $0 = \frac{\|\mathbf{y}_j - \bar{y}_j \mathbf{1}\|^2}{2\sigma^2}$.

L'affirmation selon laquelle l'atome optimal est celui maximisant la corrélation est au cœur de beaucoup d'approches gloutonnes tel que le Matching Pursuit. Cependant, la corrélation seule ne suffit pas vu que l'erreur d'approximation dépend également de $\|\mathbf{y}_j - \bar{y}_j \mathbf{1}\|^2/2\sigma^2$, qui est une quantité dépendante du rapport signal sur bruit de l'observation \mathbf{y}_j . Ainsi, la corrélation indique quels atomes expliquent bien un patch donné, tandis que (3.44) définit quels patches sont bien expliqués par un atome donné.

3.4.4 Cartes d'erreurs d'approximation d'une image par un atome avec invariances

Dans cette section, nous décrivons une méthode de calcul de carte d'erreur d'approximation d'une image bruitée par un atome. Cette méthode de calcul possède les deux invariances que nous considérons : à la translation et aux transformations radiométriques affines. Il s'agit d'un produit intermédiaire que nous utiliserons pour élaborer un algorithme d'apprentissage de dictionnaire, de débruitage d'images et de mise en correspondance d'images dans de plus grandes images. Ces applications seront dédiées aux images radar et seront décrites en section 4.2.

Notre méthode de calcul utilise le GLR-test [Deledalle *et al.* (2013)] présenté au paragraphe 3.4.3 décrivant un critère qui évalue à quel point un patch bruité \mathbf{y}_j correspond à un atome \mathbf{d} à une transformation affine près. Pour mesurer la performance globale d'un atome \mathbf{d} , on a besoin de le comparer rapidement à tous les patches \mathbf{y}_j pouvant être extraits de l'image \mathbf{y} . En réécrivant l'équation (3.44) comme suit :

$$-\log \text{GLR}(\mathbf{y}_j, \mathbf{d}) = \|\mathbf{y}_j - \bar{y}_j \mathbf{1}\|_2^2 - \frac{\langle \mathbf{y}_j - \bar{y}_j \mathbf{1}, \mathbf{d} - \bar{d} \mathbf{1} \rangle^2}{\|\mathbf{d} - \bar{d} \mathbf{1}\|_2^2} \text{ si } \mathbf{d} \neq \bar{d} \mathbf{1} \text{ et } \mathbf{y}_j \neq \bar{y}_j \mathbf{1} \quad (3.45)$$

et si on veut passer du patch \mathbf{y}_j à l'image entière \mathbf{y} , on doit considérer qu'on convole l'atome à toute l'image pour obtenir une carte du GLR de la même taille que l'image \mathbf{y} .

$$\text{SSD}_{\text{map}}^{\text{AI}}(\mathbf{d}, \mathbf{y}) = \frac{1}{2\sigma^2} \times \begin{cases} \mathbf{y}^2 * \boldsymbol{\iota} - w^2(\mathbf{y} * \boldsymbol{\iota}/w^2)^2 - \frac{(\mathbf{y} * (\mathbf{d} - \bar{d} \mathbf{1}))^2}{\|\mathbf{d} - \bar{d} \mathbf{1}\|^2} & \text{si } \mathbf{d} \neq \bar{d} \mathbf{1} \\ \mathbf{y}^2 * \boldsymbol{\iota} - w^2(\mathbf{y} * \boldsymbol{\iota}/w^2)^2 & \text{sinon} \end{cases} \quad (3.46)$$

où $\boldsymbol{\iota}$ est une image représentant le support d'un patch de taille $w \times w$.

Ainsi, (3.46) est une carte associant à chaque position de pixel l'erreur d'approximation présentée en équation (3.44) entre l'atome \mathbf{d} et l'image \mathbf{y} avec le critère $\text{SSD}_{\text{map}}^{\text{AI}}$, pour Sum of Square Differences with Affine Invariance. Cette idée est proche des filtres convolutifs comme [Kavukcuoglu *et al.* (2010)] où l'on calcule des cartes de poids

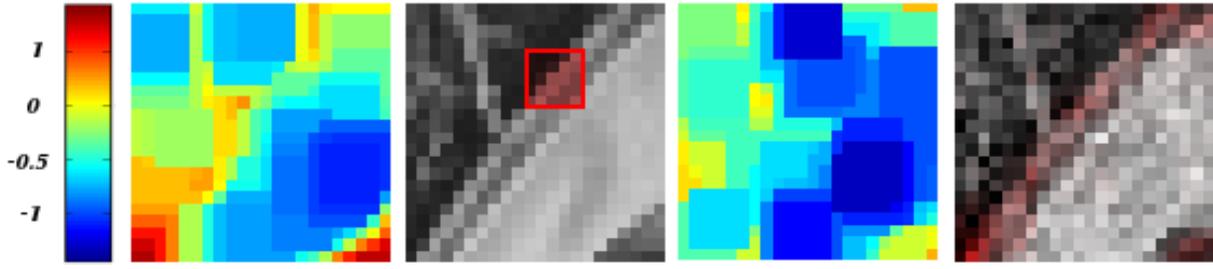


FIGURE 3.19 – De gauche à droite : la carte $\text{SSD}_{\text{map}}^{\text{AI+SI}}$ calculée sur un atome constant (plus c'est bleu plus les valeurs du critère à minimiser sont faibles) ; l'atome choisi (coloré en rouge) sur l'image d'intérêt ; la carte obtenue en ajoutant cet atome ; en rouge : les parties de l'image bruitée qui ont été mieux expliquées par l'atome choisi.

parcimonieux et non des cartes d'erreurs d'approximation.

Au delà de l'invariance aux transformations radiométriques affines, nous souhaitons introduire la propriété d'invariance par translation. Nous procédons pour cela de la même manière qu'avec l'a priori invariant par translation présenté en section 3.2.2. Au lieu d'approcher tous les patchs de l'image bruitée avec un dictionnaire (i.e. chaque pixel de l'image a plusieurs représentants en fonction des patchs auxquels il appartient), on impose que quel que soit le patch qui le contienne, chaque pixel soit bien représenté au moins une fois. On se place donc dans une optique de pavage d'une image plutôt que d'approximation de chaque patch extrait. L'erreur d'approximation sélectionnée en un pixel est donc la plus faible parmi les patchs contenant le pixel en question. La carte donnant pour chaque pixel l'erreur d'approximation optimale entre un atome et tous les patchs peut être obtenue par une érosion au sens de la morphologie mathématique de la carte donnée par l'équation (3.46) :

$$\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{d}, \mathbf{y}) = \text{erode}(\text{SSD}_{\text{map}}^{\text{AI}}(\mathbf{d}, \mathbf{y}), \boldsymbol{\nu}) \quad (3.47)$$

où SI signifie Shift-Invariance pour invariance par translation et $\text{erode}(\cdot, \boldsymbol{\nu})$ est l'opérateur d'érosion avec l'élément structurant $\boldsymbol{\nu}$.

La figure 3.19 illustre les propriétés d'invariances de notre procédure. En effet, dans cet exemple jouet, nous commençons par calculer une carte $\text{SSD}_{\text{map}}^{\text{AI+SI}}$ initiale sur une image bruitée par un bruit additif gaussien avec un atome constant. Ensuite, nous avons sélectionné manuellement un atome dans l'image sans bruit (ici, une portion de contour) et mis à jour la carte en remplaçant les valeurs de la carte calculée avec l'atome constant qui étaient supérieures aux valeurs de la carte calculée avec l'atome sans bruit. On observe qu'il y a bien un gain tout le long du contour ce qui montre l'efficacité du critère $\text{SSD}_{\text{map}}^{\text{AI+SI}}$.

Ce critère s'adapte facilement aux images radar à condition de leur appliquer une transformation logarithmique. Dans la section 4.2, nous décrivons diverses applications utilisant

ces cartes d'erreurs et les étudierons plus en détails.

3.5 Conclusion

Nous avons présenté dans ce chapitre quelques modèles munis d'invariance aux transformations radiométriques affines. Peu de modèles de ce type sont proposés dans la littérature mais certains modèles existants sont intrinsèquement munis de cette propriété (par exemple : certaines approches par dictionnaires). Nous avons également présenté des méthodes existantes d'apprentissage d'épitomes et de dictionnaires de patches avec une propriété d'invariance par translation. Ci-dessous, un récapitulatif des familles de méthodes que nous avons abordées :

Maximisation d'un critère basé sur une corrélation normalisée :

- Algorithmes d'apprentissage de dictionnaires à l'architecture classique : codage parcimonieux + mise à jour du dictionnaire.
- Calcul explicite du paramètre de translation maximisant le critère.
- Exemple : MoTIF [Jost *et al.* (2006)], où l'atome est de plus petite taille que les patches de la base d'apprentissage.

Filtres convolutifs

- Calcul non explicite de paramètre de translation.
- L'image n'est pas décomposée en patches.
- Principe : minimiser l'écart entre l'image à représenter et la somme de convolutions entre les atomes du dictionnaire et des cartes de poids parcimonieux associées.
- Exemple de méthode d'apprentissage de filtres convolutifs : [Kavukcuoglu *et al.* (2010)].

Épitomes

- Calcul non explicite de paramètre de translation.
- Principe : au lieu de construire un dictionnaire composé d'atomes par une méthode d'apprentissage, on construit une imagerie qui résume les structures observées dans les patches extraits d'une base d'images.
- Exemple de méthode d'apprentissage d'épitome : [Benoît *et al.* (2011)].

Nous avons proposé plusieurs contributions dans l'élaboration de critères de similarité entre patches et de méthodes d'apprentissage de dictionnaires avec les invariances considérées. En résumé, nous avons développé :

- un critère de similarité entre un atome et un patch basé sur la maximisation d'une corrélation normalisée où :
 - L'atome est de plus grande taille que le patch pour éviter les problèmes de bords.
 - L'atome est extrait d'un dictionnaire fixe existant.

- Ce critère est invariant par translation et par transformation radiométrique affine.
- Il a été appliqué au débruitage.
- Une méthode de calcul de carte d'erreur d'approximation d'une image par un atome :
 - Cette méthode de calcul est invariante par translation et transformation radiométrique affine.
 - On proposera en section 4.2 un algorithme d'apprentissage de dictionnaire 1-sparse basé sur ces cartes d'erreurs et notamment une application au débruitage d'images SAR.
- Une nouvelle méthode de recombinaison de patches débruités :
 - Elle a été utilisée avec un algorithme de débruitage utilisant un a priori basé sur des GMM mais peut être utilisée avec d'autres a priori exprimés sur des patches.
 - Cette méthode est invariante par translation.
 - Nous la valoriserons en section 4.3 dans une application à la restauration d'images SAR.
- Le GMM aligné (aGMM) :
 - Nous avons proposé une manière de modéliser statistiquement la distribution des patches par des GMM avec une propriété d'invariance radiométrique affine.
 - Nous avons développé un algorithme d'apprentissage de GMM aligné, qui est une version simplifiée d'un GMM invariant radiométrique affine où un patch et son inverse de contraste sont proches d'un même vecteur moyen au signe près.
 - Nous avons également développé un algorithme de débruitage utilisant l'aGMM. Ses performances sont meilleures qu'avec un GMM classique.
 - Nous discuterons des perspectives de cette approche et montrerons des résultats préliminaires de débruitage produits avec un GMM invariant radiométrique en section 5.2. Dans de futurs travaux, on fusionnera l'a priori invariant par translation que nous avons développé au GMM invariant affine, ce qui produira une méthode de débruitage performante munie des deux invariances considérées dans cette thèse.

Chapitre 4

Applications à l'imagerie SAR

Ce chapitre présente les différentes applications de ce travail de thèse aux images SAR. Le speckle qu'on observe sur ces images étant modélisé par un bruit multiplicatif, nous commençons par présenter un état de l'art des familles de méthodes de débruitage d'images SAR en section 4.1.

Nous présentons ensuite nos contributions dans l'apprentissage de dictionnaires dédiés aux images SAR. En section 4.2, nous décrivons un algorithme d'apprentissage de dictionnaire 1-sparse utilisant les cartes d'erreurs d'approximation d'une image par un atome avec invariances par translation et transformation affine présentées en section 3.4.4. Cet algorithme a fait l'objet de la publication [Tabti *et al.* (2014a)].

Ayant constaté la richesse des dictionnaires de modèles Gaussiens, nous adaptons en section 4.3 l'algorithme de débruitage EPLL [Zoran et Weiss (2011)] (décrit en section 2.4.3) à la restauration d'images SAR [Tabti *et al.* (2014b)]. De plus, nous introduisons dans cette adaptation une propriété d'invariance par translation en utilisant l'a priori décrit en section 3.2.2.

En section 4.4, nous proposons d'évaluer l'apport de l'apprentissage d'un GMM sur des données SAR comparé à un GMM appris sur des images naturelles dans une tâche de débruitage.

Enfin, après avoir décrit quelques méthodes de classification d'images satellitaires en section 4.5.1, nous proposons une méthode de classification d'images radar montrant le potentiel des GMM pour représenter les structures des images SAR en section 4.5.2, [Tabti *et al.* (2015)].

4.1 Restauration d'images SAR

En section 2.1, nous avons décrit le principe des images radar et expliqué que le phénomène de chatoiement, ou speckle, est modélisé par une loi gamma lorsque l'image SAR est en intensité ou une loi Nakagami-Rayleigh lorsque l'image est en amplitude. De la même façon que pour les images naturelles, pour de nombreux traitements comme la détection de cibles et la classification, il est souvent nécessaire de restaurer l'image

SAR au préalable, c'est à dire de supprimer le speckle en perdant le moins d'information possible. Nous discuterons en section 4.1.1 de plusieurs attaches aux données possibles à utiliser afin de restaurer une image SAR. Ensuite, nous décrirons en section 4.1.2 plusieurs familles de méthodes de réduction de speckle usuelles.

4.1.1 Plusieurs attaches aux données

Considérons le problème de restauration d'une image en rappelant l'équation (2.21) :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \underbrace{-\log p(\mathbf{y}|\mathbf{x})}_{\text{attache aux données}} \underbrace{-\log p(\mathbf{x})}_{\text{a priori}}$$

où \mathbf{y} est la version bruitée de \mathbf{x} et $\hat{\mathbf{x}}$ est l'estimation de \mathbf{x} .

L'attache aux données utilisée pour restaurer une image SAR varie notamment en fonction du type de données considérées. Comme nous l'avons précisé en section 2.1, d'après le modèle de Goodman, si l'image est en amplitude, elle est modélisée par une distribution de Nakagami-Rayleigh dont on rappelle l'équation (2.4) :

$$p_{\mathcal{NR}}(y_j|x_j) = \frac{2L^L}{\Gamma(L)x_j^{2L}} y_j^{2L-1} \exp\left(-L\frac{y_j^2}{x_j^2}\right)$$

où L correspond au nombre de vues et Γ est la fonction gamma.

Si l'image est en intensité, elle est modélisée par une loi gamma dont on rappelle l'équation (2.5) :

$$p_{\mathcal{G}}(y_j|x_j) = \frac{L^L y_j^{L-1}}{x_j^L \Gamma(L)} \exp\left(-L\frac{y_j}{x_j}\right)$$

D'autres distributions peuvent être considérées comme la loi de Fisher ou encore la loi \mathcal{K} [Nicolas (2006)] mais elles comportent alors de nombreux paramètres à estimer, ce qui rend le problème de restauration difficile.

Une manière de supprimer le bruit multiplicatif est de le rendre additif dans le but d'utiliser des méthodes développées pour du bruit additif. La méthode la plus simple est d'appliquer une transformation logarithmique à l'image bruitée par un bruit multiplicatif, le log transformant la multiplication en addition. On parle d'une approche homomorphique. Ensuite, après application de la méthode dédiée au bruit additif (souvent considéré comme Gaussien), on inverse la transformation logarithmique en appliquant la fonction exponentielle. Mais en réalité, la distribution du logarithme d'une loi gamma n'est pas Gaussienne mais de Fisher-Tippett :

$$p_{\mathcal{FT}}(\tilde{y}_i|\tilde{x}_i) = \frac{L^L}{\Gamma(L)} e^{L(\tilde{y}_i - \tilde{x}_i)} e^{-Le^{\tilde{y}_i - \tilde{x}_i}} \quad (4.1)$$

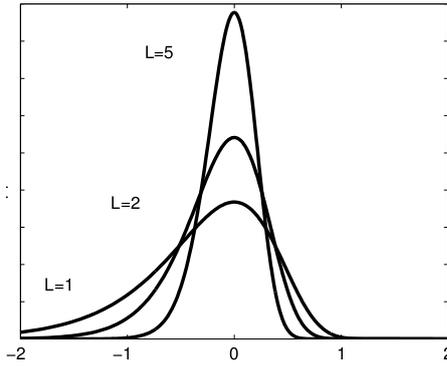


FIGURE 4.1 – *Distribution de Fisher-Tippett avec différents nombres de vues L . Plus L est petit, plus la courbe est dissymétrique.*

où \tilde{y}_i et \tilde{x}_i sont les i -èmes pixels des images bruitée $\tilde{\mathbf{y}}$ et sans bruit $\tilde{\mathbf{x}}$ respectivement qui sont les versions transformées logarithmiquement de \mathbf{y} et \mathbf{x} respectivement.

Sous hypothèse d'indépendance, l'opposé de la log-vraisemblance associé est :

$$-\log p(\tilde{\mathbf{y}}|\tilde{\mathbf{x}}) = L \sum_{i=1}^N \tilde{x}_i + e^{\tilde{y}_i - \tilde{x}_i} + C \quad (4.2)$$

où $C = -L \log(L) + \log(\Gamma(L)) - L\tilde{y}_i$ est une constante vis à vis de $\tilde{\mathbf{x}}$.

La distribution de Fisher-Tippett est illustrée en figure 4.1. On observe qu'elle est asymétrique, ce qui explique un défaut récurrent des méthodes homomorphiques où des pixels sombres isolés apparaissent. La variance de la distribution de Fisher-Tippett est :

$$\text{Var}[\tilde{Y}] = \psi(1, L) \quad (4.3)$$

où Y est une variable aléatoire de loi gamma, $\tilde{Y} = \log(Y)$ est une variable aléatoire de loi Fisher-Tippett et $\psi(m, L)$ est la fonction polygamma d'ordre m et de degré L , définie par la dérivée logarithmique $m + 1$ -ème de la fonction gamma. Cette variance étant indépendante du signal, on constate qu'appliquer le logarithme stabilise bien la variance du bruit gamma.

La moyenne de \tilde{Y} (mentionnée notamment dans [Xie *et al.* (2002b)]) est la suivante :

$$\mathbb{E}[\tilde{Y}|x] = \log(x) + \psi(L) - \log(L) \quad (4.4)$$

La formule (4.4) permet de prendre conscience que si l'on souhaite appliquer une transformation logarithmique à une image avec du bruit gamma et utiliser une méthode de débruitage développée pour du bruit Gaussien, l'estimateur de l'espérance est biaisé. Ce biais est illustré en figure 4.2. Il peut être corrigé avec la formule suivante (notamment décrite dans [Xie *et al.* (2002a)]) après avoir appliqué la fonction exponentielle à l'image

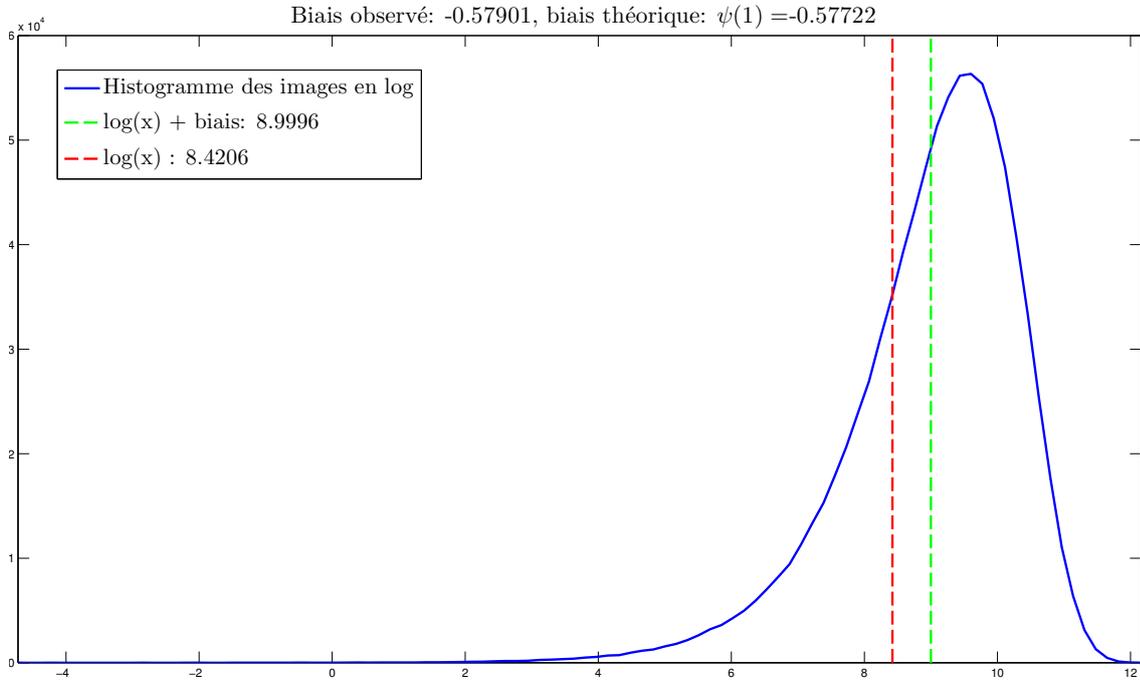
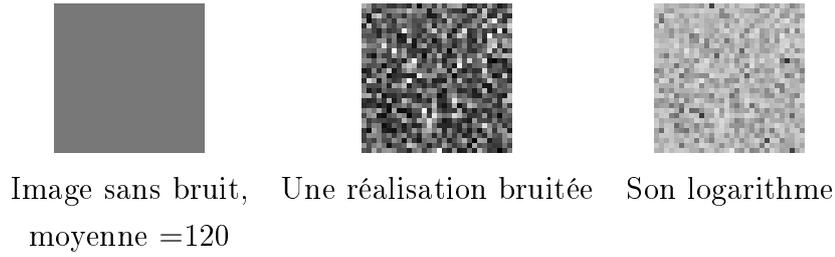


FIGURE 4.2 – Avec 1000 réalisations d’images de taille 30×30 bruitées par une loi Nakagami-Rayleigh de paramètre $L = 1$ transformées logarithmiquement, on calcule un histogramme. On observe que la distribution obtenue est bien similaire à une distribution de Fisher-Tippett. De plus, en calculant la moyenne théorique des données bruitées transformées logarithmiquement avec la formule (4.4) et en la comparant avec la moyenne observée on retrouve bien le biais annoncé.

restaurée :

$$\hat{\mathbf{x}}_d = \frac{L}{\exp(\psi(L))} \hat{\mathbf{x}}_b \quad (4.5)$$

où $\hat{\mathbf{x}}_d$ est l’image restaurée débiaisée, $\hat{\mathbf{x}}_b$ est l’image restaurée biaisée et ψ est la fonction polygamma.

4.1.2 Plusieurs familles de méthodes

Dans cette section, nous décrivons brièvement quelques grandes familles de méthodes de réduction de speckle.

Filtrage adaptatif Le filtre proposé par [Lee (1981)] est un cas particulier du filtre décrit dans [Kuan *et al.* (1987)] dont le principe réside dans la minimisation de l'écart quadratique moyen entre l'image observée et l'estimation de l'image restaurée sous contraintes de linéarité et d'absence de biais :

$$\hat{x}_i = \bar{y}_i + k(y_i - \bar{y}_i) \quad (4.6)$$

où x_i et y_i sont les valeurs de \mathbf{x} l'image sans bruit et \mathbf{y} l'image bruitée au pixel i respectivement et \bar{y}_i la moyenne des pixels dans une fenêtre autour du i -ème pixel de l'image \mathbf{x} . Le paramètre $k \in [0, 1]$ est calculé localement sur une fenêtre glissante et dépend de la variance locale. Il en résulte que si le pixel d'intérêt est dans une zone homogène, il est remplacé par la moyenne de ses pixels voisins, et si le pixel est dans une zone très texturée (de forte variance par rapport à celle du speckle), la valeur bruitée est conservée.

Régularisation par variation totale Dans le but de préserver les contours très présents dans les images satellitaires, beaucoup d'approches traitent le problème de restauration d'images SAR non texturées avec une régularisation par variation totale, ou Total Variation (TV) en Anglais, décrite en section 2.4.1. La plupart de ces méthodes utilisent une attache aux données Fisher-Tippett (4.2). Les différences entre ces approches se situent au niveau de l'a priori TV et au niveau de la façon dont le problème d'optimisation (plus ou moins difficile) est résolu. En effet, [Aubert et Aujol (2008)] utilisent l'a priori TV sur l'image non transformée logarithmiquement :

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \sum_{i=1}^N \tilde{x}_i + e^{\tilde{y}_i - \tilde{x}_i} + \frac{\lambda}{L} |\nabla_i \mathbf{x}| \quad (4.7)$$

Ce qui donne un problème d'optimisation non convexe. Les auteurs proposent de trouver un minimum local en utilisant l'équation d'Euler-Lagrange et une descente de gradient. Un résultat de cette approche est présenté en figure 4.3.

Dans [Shi et Osher (2008)] et [Bioucas-Dias et Figueiredo (2010)] les auteurs utilisent l'a priori TV sur l'image transformée logarithmiquement :

$$\hat{\tilde{\mathbf{x}}} = \underset{\tilde{\mathbf{x}}}{\operatorname{argmin}} \sum_{i=1}^N \tilde{x}_i + e^{\tilde{y}_i - \tilde{x}_i} + \frac{\lambda}{L} |\nabla_i \tilde{\mathbf{x}}|, \quad \hat{\mathbf{x}} = e^{\hat{\tilde{\mathbf{x}}}} \quad (4.8)$$

Ce qui donne un problème d'optimisation convexe. Dans [Shi et Osher (2008)], les auteurs utilisent l'équation d'Euler-Lagrange et une descente de gradient comme dans [Aubert et Aujol (2008)]. Dans [Bioucas-Dias et Figueiredo (2010)], les auteurs reformulent (4.8) comme un problème d'optimisation contraint qu'ils résolvent avec l'algorithme ADMM (Alternating Directions Method of Multipliers).

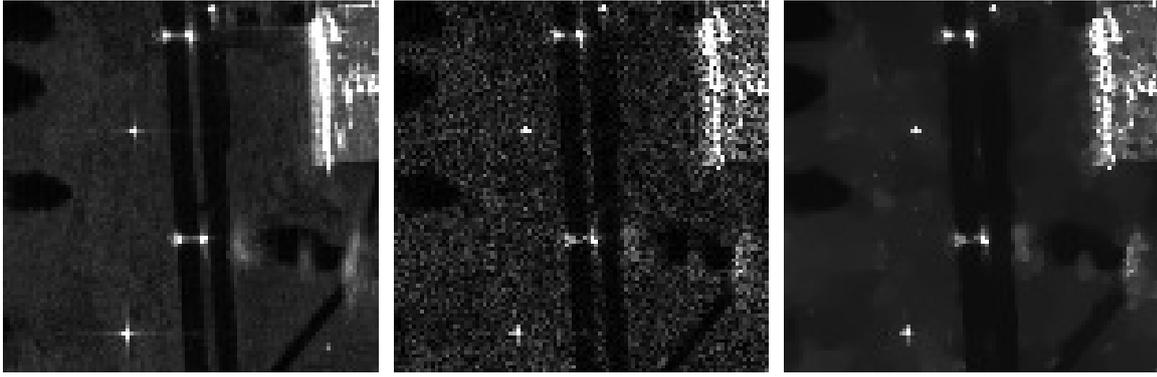


FIGURE 4.3 – De gauche à droite : une image SAR 100-vues, sa version bruitée 1-vue, sa version restaurée avec l’approche proposée dans [Aubert et Aujol (2008)]. On observe qu’il y a beaucoup de bruit résiduel, en particulier au niveau de la construction en haut à droite de l’image.

Une autre attache aux données est proposée dans [Steidl et Teuber (2010)] avec la I-divergence de Csiszàr, ou divergence de Kullback-Leibler (KL) généralisée :

$$-\log p(\tilde{\mathbf{y}}|\tilde{\mathbf{x}}) = \sum_{i=1}^N -e^{\tilde{y}_i \tilde{x}_i} + e^{\tilde{x}_i} \quad (4.9)$$

Habituellement réservée au bruit de Poisson, Steidl et al. montrent qu’avec un a priori de variation totale sur l’image en intensité, le problème (4.8) a le même minimiseur que le problème de restauration suivant :

$$e^{\hat{\tilde{\mathbf{x}}}} = \operatorname{argmin}_{e^{\tilde{\mathbf{x}}}} \sum_{i=1}^N -e^{\tilde{y}_i \tilde{x}_i} + e^{\tilde{x}_i} + \lambda |\nabla_i e^{\tilde{\mathbf{x}}}| \quad (4.10)$$

Les approches basées sur les champs de Markov décrites en section 2.4.1 sont souvent utilisées pour restaurer des images SAR avec un a priori TV. Cependant, c’est un problème difficile à résoudre de façon exacte par optimisation combinatoire à cause des problèmes de mémoire rencontrés en traitant des images de grande taille. Dans [Denis et al. (2009)], les auteurs utilisent l’opposé de la log-vraisemblance d’une distribution de Nakagami-Rayleigh comme attache aux données (non convexe). Pour résoudre ce problème, ils proposent une méthode d’optimisation non-exacte décomposée en plusieurs sous-problèmes qui eux ont une solution exacte sur un graphe à deux niveaux. Dans [Denis et al. (2010)], un a priori TV- ℓ_0 est proposé où les zones homogènes et les contours d’une image SAR sont restaurés grâce à la variation totale et les cibles brillantes sont représentées parcimonieusement par la pseudo-norme ℓ_0 . Ce travail a été étendu à des images SAR multi-temporelles dans [Lobry et al. (2015)].

Modèles basés sur la redondance des patches Les méthodes non-locales ont grandement influencé le débruitage des images naturelles et il en fut de même pour les images radar. En effet, elles permettent de préserver les contours et les textures d’une image

contrairement aux approches basées sur la variation totale qui préservent les contours mais atténuent les textures et les cibles brillantes. L'algorithme des NL-means décrit en section 2.3.1, a été adapté au bruit multiplicatif dans [Deledalle *et al.* (2009)]. L'algorithme BM3D (également décrit en section 2.3.1) a aussi été étendu au bruit multiplicatif dans [Makitalo *et al.* (2010)] et [Parrilli *et al.* (2012)]. Dans [Makitalo *et al.* (2010)], une transformation logarithmique puis l'algorithme BM3D classique sont appliqués à l'image. Dans [Deledalle *et al.* (2009)] et [Parrilli *et al.* (2012)], la modification essentielle par rapport aux approches développées pour le bruit Gaussien se situe au niveau du critère de similarité entre les patchs qui est adapté au speckle. Dans [Deledalle *et al.* (2009)], on calcule un estimateur du maximum de vraisemblance pondéré :

$$\hat{\mathbf{x}}_i = \underset{\mathbf{x}_i}{\operatorname{argmax}} \omega(i, j) \log p(\mathbf{y}_j | \mathbf{x}_i) \quad (4.11)$$

où $\omega(i, j)$ mesure la similarité entre les patchs autour de la position i . Ces poids se décomposent en deux termes : un terme de vraisemblance et un terme d'a priori obtenu grâce à une version de l'image préfiltrée.

Cet algorithme a été amélioré dans [Deledalle *et al.* (2015)]. Les résultats de ces deux adaptations sont très satisfaisants mais peuvent présenter un phénomène de patchs rares au niveau de contours très contrastés comme pour les images naturelles.

Modèles parcimonieux De nombreuses approches par dictionnaires ont été développées pour les images SAR. En effet, les patchs d'une image SAR sont parcimonieux dans une base telle que les cosinus discrets (DCT), comme l'illustre la figure 4.4. Parmi les méthodes basées sur les ondelettes, on retrouve par exemple [Guo *et al.* (1994)] où les auteurs appliquent un seuillage des coefficients d'ondelettes sur des images SAR transformées logarithmiquement. D'autres méthodes vont plus loin en modélisant la distribution des coefficients d'ondelettes des images SAR une fois transformées logarithmiquement. Par exemple, dans [Achim *et al.* (2003)], les coefficients sont modélisés par une famille de distributions alpha-stables (une famille de distributions à queues lourdes).

D'autres approches sont basées sur des dictionnaires de patchs. L'algorithme K-SVD, décrit en section 2.4.2, a été adapté aux images SAR dans [Foucher (2008)] qui applique une transformation logarithmique à l'image puis utilise K-SVD en veillant à ne pas introduire de biais. Une autre approche est proposée par [Chainais (2012)] afin d'apprendre un dictionnaire 1-sparse sur des patchs bruités avec un bruit non-Gaussien. Il s'agit en fait d'une application de l'algorithme des K-medoids (dont une version est décrite dans [Park et Jun (2009)]) qui permet d'être plus robuste aux outliers (données aberrantes) que l'algorithme K-Means en considérant des clusters ou medoids à la place des centroïdes. Au lieu d'utiliser un écart quadratique comme critère de similarité, l'auteur utilise un rapport de vraisemblance généralisé (Generalized Likelihood Ratio,

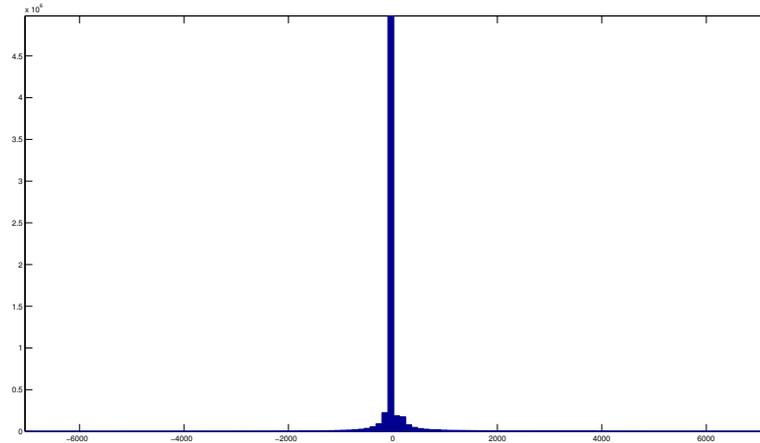


FIGURE 4.4 – Histogramme des coefficients de la décomposition parcimonieuse de patches 8×8 tirés d'une image SAR 1-vue transformée logarithmiquement par une OMP. Le critère d'arrêt est basé sur l'écart entre les données et la décomposition. Les valeurs sont concentrées autour de 0. Ainsi, tout comme les images naturelles, les images SAR sont parcimonieuses dans certaines bases. Il s'agit ici d'un dictionnaire DCT comportant 256 atomes.

GLR, en Anglais) avec une loi gamma tel qu'il est décrit dans [Deledalle *et al.* (2012a)]. Dans la section 4.2, nous présenterons une autre méthode d'apprentissage de dictionnaire 1-sparse dédiée aux images SAR qui est munie de propriétés d'invariances à la translation et aux transformations radiométriques affines.

Dans la suite de ce chapitre, nous présenterons une contribution de ce travail de thèse dans l'apprentissage d'un dictionnaire 1-sparse dédié aux images SAR. Nous présenterons également nos contributions dans des méthodes basées sur les GMM ayant constaté leur efficacité sur les images naturelles allant de l'apprentissage de GMM à la classification d'images SAR en passant par leur restauration.

4.2 Apprentissage d'un dictionnaire 1-sparse avec invariances

Dans le chapitre 3, nous avons motivé l'intérêt d'introduire des propriétés d'invariance dans des méthodes d'apprentissage de dictionnaires afin de rendre leurs atomes moins redondants notamment. Nous avons également décrit une méthode permettant de calculer une carte d'erreur d'approximation d'une image bruitée par un patch sans bruit avec des propriétés d'invariance par translation et transformation affine en section 3.4.4. Nous allons décrire dans cette section une application du calcul de ces cartes d'erreur dans une procédure d'apprentissage de dictionnaire 1-sparse sur des images radar [Tabti *et al.* (2014a)]. Nous avons justifié l'usage de la 1-parcimonie en section 3.4.1 où nous avons



FIGURE 4.5 – À gauche : l'image 100-vues de référence \mathbf{y}^* (Copyright CNES) d'où les patches candidats sont extraits. À droite : l'image 1-vue \mathbf{y} (Copyright CNES) à résumer par le dictionnaire.

montré qu'il s'agit du degré de parcimonie optimal en terme d'erreur quadratique moyenne dans les conditions de fort bruit de l'imagerie radar.

4.2.1 Procédure d'apprentissage

La méthode d'apprentissage de dictionnaire proposée utilise le critère présenté en section 3.4.4. L'objectif de ce dictionnaire est de fournir un résumé compact d'une image radar bruitée \mathbf{y} . Les atomes de ce dictionnaire sont extraits d'une image 100 vues (donc quasiment sans bruit) de référence notée \mathbf{y}^* . Les images \mathbf{y} et \mathbf{y}^* sont toutes deux transformées logarithmiquement afin de réduire leur dynamique et de rendre additif le bruit multiplicatif sur \mathbf{y} .

Dans un premier temps, T patches sont aléatoirement extraits de \mathbf{y}^* . Ils définissent un ensemble de candidats pour faire partie du dictionnaire final \mathbf{D} . Le nombre total d'éléments du dictionnaire, noté K est défini par l'utilisateur. Le dictionnaire initial est constitué des K premiers éléments de cet ensemble de candidats noté \mathcal{C} .

Rappelons l'équation (3.46) qui décrit comment est calculée la carte d'erreur d'approximation d'une image bruitée par un patch sans bruit avec invariance affine :

$$\text{SSD}_{\text{map}}^{\text{AI}}(\mathbf{d}, \mathbf{y}) = \frac{1}{2\sigma^2} \times \begin{cases} \mathbf{y}^2 * \boldsymbol{\iota} - w^2(\mathbf{y} * \boldsymbol{\iota}/w^2)^2 - \frac{(\mathbf{y} * (\mathbf{d} - \bar{d}\mathbf{1}))^2}{\|\mathbf{d} - \bar{d}\mathbf{1}\|^2} & \text{si } \mathbf{d} \neq \bar{d}\mathbf{1} \\ \mathbf{y}^2 * \boldsymbol{\iota} - w^2(\mathbf{y} * \boldsymbol{\iota}/w^2)^2 & \text{sinon} \end{cases}$$

où $\boldsymbol{\iota}$ est une image binaire représentant le support d'un patch de taille $w \times w$.

Rappelons maintenant l'équation (3.47) qui décrit comment est calculée la carte d'erreur d'approximation d'une image bruitée par un patch sans bruit avec invariance affine et invariance par translation :

$$\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{d}, \mathbf{y}) = \text{erode}(\text{SSD}_{\text{map}}^{\text{AI}}(\mathbf{d}, \mathbf{y}), \boldsymbol{\iota})$$

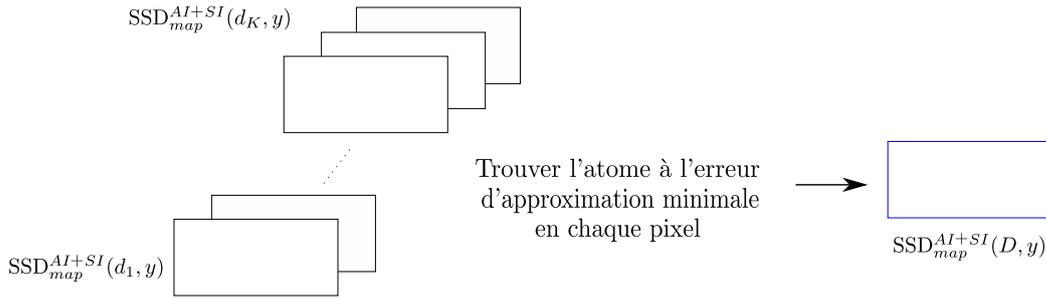


FIGURE 4.6 – Illustration du calcul de la carte d'erreur d'approximation d'une image par un dictionnaire en équation (4.12). On calcule les cartes d'erreur d'approximation de \mathbf{y} pour tous les atomes puis en chaque pixel on détermine quel atome présente l'erreur minimale.

où SI signifie Shift-Invariance pour invariance par translation et $erode(\cdot, \iota)$ est l'opérateur d'érosion avec l'élément structurant ι .

L'approximation d'une image bruitée par les atomes d'un dictionnaire donné peut être calculée efficacement par l'équation (3.46) car chaque convolution peut être calculée séparément dans le domaine de Fourier. La complexité du calcul de $SSD_{map}^{AI}(\mathbf{d}, \mathbf{y})$ vaut en conséquence $O(N \log N)$, où N est la taille de l'image, alors qu'appliquer (3.44) en extrayant tous les patches donne une complexité de $O(Nw^2)$. On définit ainsi la carte d'erreur d'approximation de l'image \mathbf{y} par le dictionnaire \mathbf{D} pour chaque pixel i par :

$$SSD_{map}^{AI+SI}(\mathbf{D}, \mathbf{y})_i = \min_{k_i} SSD_{map}^{AI+SI}(\mathbf{d}_{k_i}, \mathbf{y})_i \quad (4.12)$$

où k_i est l'indice du k -ème atome associé au pixel i . On illustre ce calcul dans la figure 4.6.

Le dictionnaire est mis à jour en considérant tous les patches candidats dans \mathcal{C} à la suite et en remplaçant les atomes du dictionnaire à chaque fois que l'erreur d'approximation moyenne est réduite. La décision de remplacer \mathbf{d}_k par un candidat \mathbf{c}_t issu de l'ensemble \mathcal{C} se fait en calculant une carte de gain local qui peut être positif ou négatif :

$$SSD_{map}^{AI+SI}(\mathbf{D}, \mathbf{y})_i - SSD_{map}^{AI+SI}(\mathbf{D} \cup \{\mathbf{c}_t\} \setminus \{\mathbf{d}_k\}, \mathbf{y})_i \quad (4.13)$$

où $\setminus \{\mathbf{d}_k\}$ signifie : privé de l'atome \mathbf{d}_k . Le patch candidat \mathbf{c}_t devient un atome du dictionnaire courant s'il existe un atome \mathbf{d}_k pour lequel l'erreur d'approximation moyenne (moyenne des gains en chaque pixels) est réduite s'il est remplacé par \mathbf{c}_t . Ce même procédé est répété pour \mathbf{c}_{t+1} et ainsi de suite.

Le calcul de gain est optimisé en tenant compte uniquement des deux atomes présentant les plus faibles erreurs d'approximations. On note leurs indices : k_i^{1st}, k_i^{2nd} pour

chaque pixel i de l'image \mathbf{y} . Le calcul du gain défini en équation (4.13) nécessite seulement le calcul d'une carte $\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{c}_t, \mathbf{y})$ avec l'équation (3.47). Plusieurs cas peuvent se présenter :

- L'atome \mathbf{d}_k n'est pas l'atome optimum au i -ème pixel (i.e., $k \neq k_i^{\text{1st}}$). Dans ce cas, soit \mathbf{c}_t améliore l'approximation du pixel i ou ne l'affecte pas. Le gain au pixel i est donc :

$$\max \left(\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{D}, \mathbf{y})_i - \text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{c}_t, \mathbf{y})_i, 0 \right),$$

- L'atome $\mathbf{d}_{k_i^{\text{1st}}}$ va être remplacé et dans ce cas deux sous-cas sont possibles :
 - Si le candidat \mathbf{c}_t présente une meilleure erreur d'approximation que le second meilleur atome $\mathbf{d}_{k_i^{\text{2nd}}}$, alors le gain au pixel i est la différence :

$$\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{D}, \mathbf{y})_i - \text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{c}_t, \mathbf{y})_i,$$

- Sinon, le candidat \mathbf{c}_t est pire que $\mathbf{d}_{k_i^{\text{2nd}}}$, ce qui donne le gain suivant au pixel i :

$$\text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{D}, \mathbf{y})_i - \text{SSD}_{\text{map}}^{\text{AI+SI}}(\mathbf{d}_{k_i^{\text{2nd}}}, \mathbf{y})_i.$$

Un exemple de dictionnaire produit par notre approche est présenté en figure 4.7. Un autre dictionnaire appris avec K-SVD sur la même image radar 1-vue (cf. fig. 4.5) est également présenté. Ils sont tous deux composés de 64 atomes de taille 8×8 . On observe que les atomes du dictionnaire appris avec l'approche proposée résument bien les structures présentes dans l'image radar (les lignes brillantes, les contours horizontaux et verticaux par exemple). La courbe de décroissance d'énergie au fur et à mesure que le dictionnaire est appris l'illustre bien également. Le dictionnaire appris avec K-SVD représente bien les contours mais contient trop d'atomes homogènes ce qui rend l'information qu'il contient trop redondante. Cependant, les deux dictionnaires échouent à encoder des textures.

En figure 4.8, nous présentons une application utilisant les atomes du dictionnaire appris et les mettant en correspondance dans une image. Ces atomes sont ordonnés de façon à ce que deux atomes voisins soient bien corrélés. Pour réaliser ce classement, on doit maximiser la corrélation entre chaque paire d'atomes, ce qui peut se faire en résolvant un problème de voyageur de commerce [Applegate *et al.* (2011)]. Afin de faciliter l'interprétation, nous imposons que le paramètre α de transformation affine soit positif. De plus, on impose également que $\alpha \geq 0.5$ pour que les patches mis en correspondance dans l'image aient un contraste assez proche de l'atome. On observe que les atomes semblent correctement mis en correspondance avec des patches de l'image.

4.2.2 Limites et perspectives

Nous avons montré par étude expérimentale en section 3.4.1, qu'en cas de fort bruit, le degré de parcimonie optimal dans une tâche de débruitage au sens d'un écart quadratique

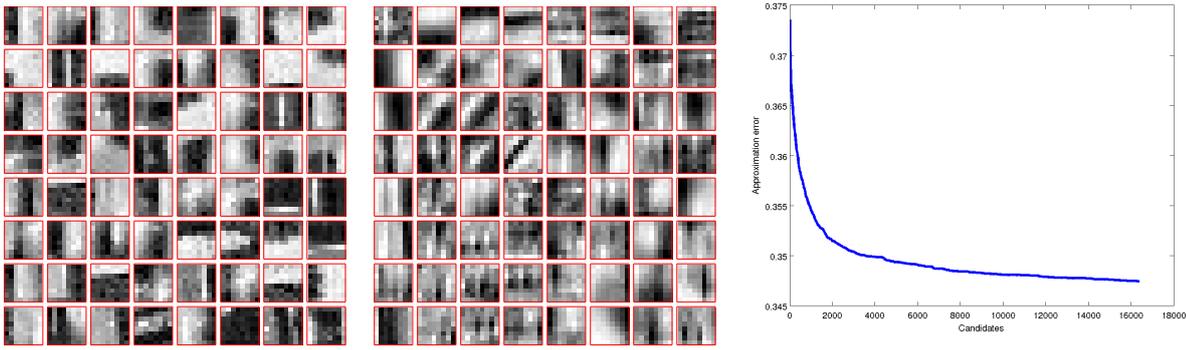


FIGURE 4.7 – De gauche à droite : dictionnaire appris avec notre approche et le dictionnaire appris avec l’algorithme K -SVD. Tous deux sont composés de 64 atomes de taille 8×8 et appris sur les images de la figure 4.5. La courbe représente la décroissance d’énergie observée au fur et à mesure qu’on inclut de nouveaux atomes à notre dictionnaire (celui de gauche).

moyen est égal à un. Nous avons ainsi développé un algorithme d’apprentissage de dictionnaire 1-sparse muni de propriétés d’invariance par translation et transformation affine. Le dictionnaire obtenu semble bien résumer les structures de l’image sur laquelle il est appris. Néanmoins, il est difficile de représenter une ligne brillante ou une cible et de la texture en même temps avec des patches car ces structures apparaissent dans des contextes trop diversifiés dans les images SAR. Pour y parvenir il faudrait des dictionnaires de très grande taille, ce qui nous éloigne de l’objectif d’apprendre des dictionnaires plus compacts. Une autre approche serait d’apprendre des dictionnaires de modèles paramétriques de lignes, de points brillants, de contours ou de végétation plutôt que des dictionnaires de patches. C’est ce que nous apportent les approches par GMM comme nous le verrons dans la suite de ce chapitre.

De plus, le biais (donc la perte de détails par rapport à l’image idéale) d’une méthode de débruitage utilisant un dictionnaire 1-sparse reste important comme on l’observe dans les figures 4.10, 3.16, 3.17. Nous espérons que ce biais serait suffisamment réduit par les informations apportées par des propriétés d’invariances mais les résultats de débruitage obtenus, bien que meilleurs qu’une approche 1-sparse classique, ne sont pas à la hauteur des méthodes les plus performantes actuellement. Apprendre un épitome avec une méthode similaire à [Benoît *et al.* (2011)] et l’utiliser dans une approche de débruitage k -sparse invariante aux transformations radiométriques affines avec k grand dans les zones texturées et k petit dans les zones homogènes, comme il est illustré en figure 4.11, serait une bonne alternative. Une manière simple de décider du degré de parcimonie k serait d’utiliser un coefficient de variation, défini entre autres dans [D’Alvise *et al.* (1999)]. L’invariance affine serait obtenue en calculant les paramètres α et β optimaux par régression linéaire. Dans la figure 4.9, nous présentons un résultat d’apprentissage d’épitome sur le logarithme d’une image SAR avec l’approche proposée par [Jojic *et al.* (2003)]. On

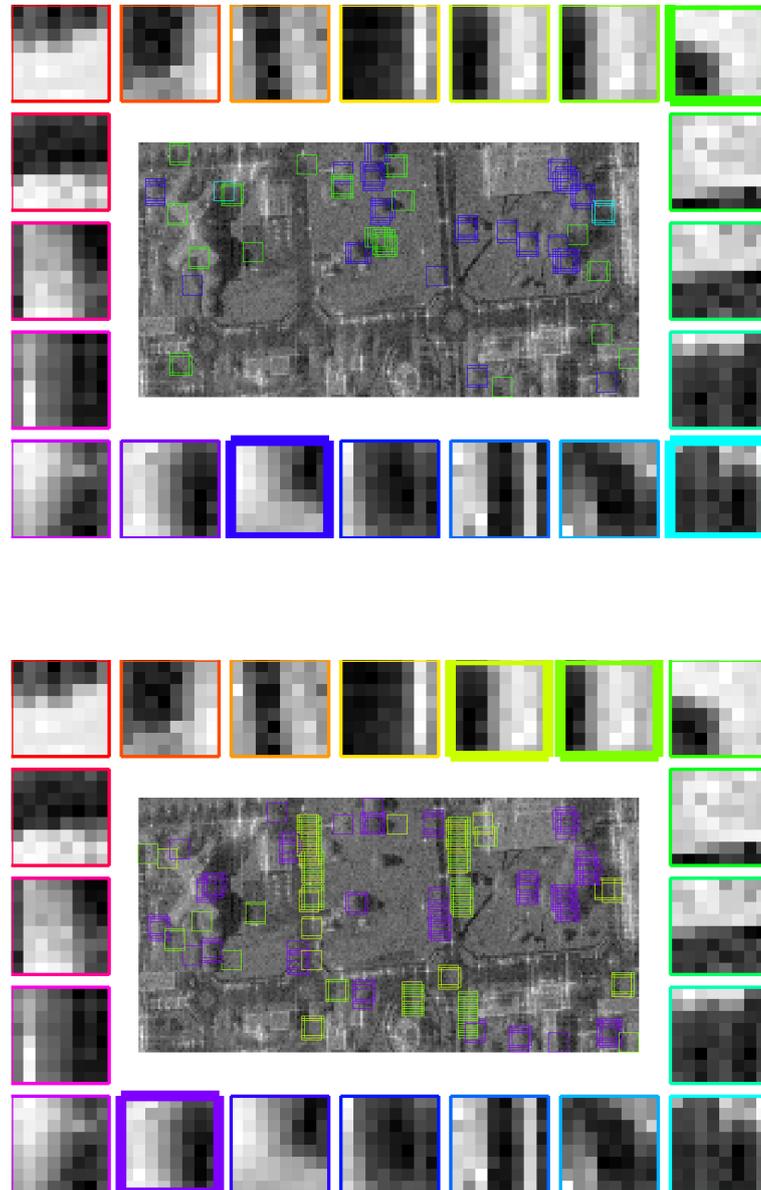


FIGURE 4.8 – *Illustration de l'utilisation du dictionnaire appris pour la mise en correspondance d'atomes dans une image bruitée. En haut : on met en correspondance les atomes représentant des coins arrondis, en bas : les atomes représentant des lignes verticales.*

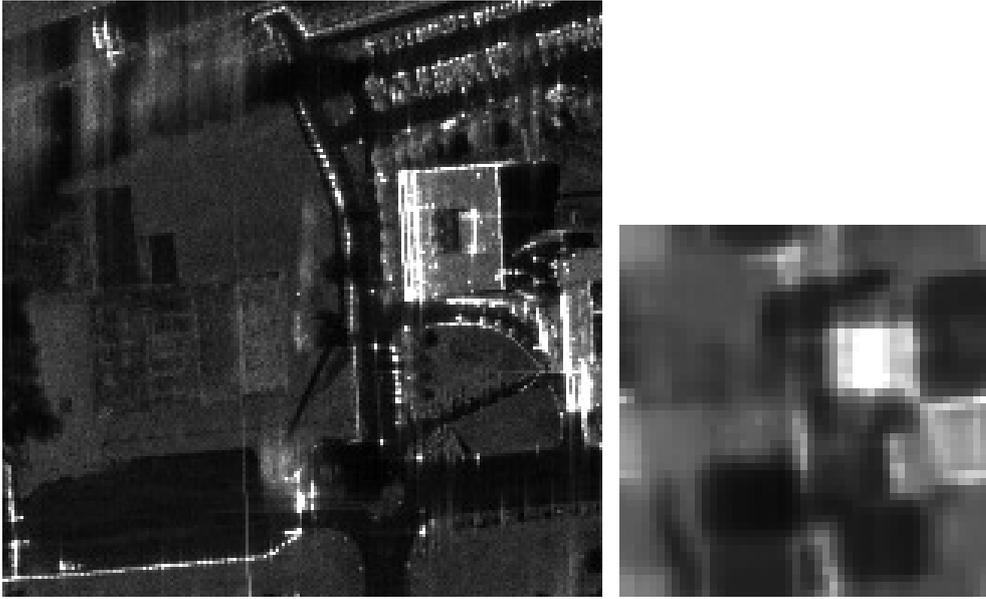


FIGURE 4.9 – Apprentissage d'un épitome de taille 50×50 avec l'approche de [Jojic et al. (2003)] (à droite) sur le logarithme d'une image SAR 100-vues de taille 200×200 (à gauche). On observe que les structures dans l'épitome sont floues et qu'il y a peu de textures. Apprendre un épitome sur des données SAR serait donc une perspective de recherche à exploiter.

observe que les structures sont floues, qu'il n'y a pas de textures ni de points brillants. Une application directe d'une méthode existante sur le logarithme d'une image SAR n'est donc pas suffisante pour apprendre un épitome adapté à ce type d'images.

4.3 L'EPLL invariant par translation et adapté aux images SAR

4.3.1 Présentation de la méthode

Cette section décrit l'adaptation de l'EPLL [Zoran et Weiss (2011)], présenté dans la section 2.4.3, à la restauration d'images SAR avec une propriété d'invariance par translation [Tabti et al. (2014b)]. Dans la section 4.1.1, nous avons décrit une façon d'adapter une méthode de débruitage de bruit Gaussien à la réduction de speckle en transformant logarithmiquement l'image d'intérêt, le bruit étant alors modélisé par une loi de Fisher-Tippett. Ainsi, on rappelle l'équation (4.2) de l'attache aux données qui n'est plus un écart quadratique, comme lorsque le bruit est Gaussien mais qui s'écrit comme suit :

$$-\log p(\mathbf{y}|\mathbf{x}) = \lambda \sum_{i=1}^N (e^{y_i - x_i} + x_i - y_i)$$



FIGURE 4.10 – De gauche à droite : image SAR 100-vues (quasiment sans bruit), image bruitée 1-vue, image débruitée, avec la même approche que celle décrite en section 3.4.2, en sélectionnant pour chaque patch l'atome qui lui correspond le mieux au sens du $\text{SSD}_{\text{map}}^{\text{AI+SI}}$ avec le dictionnaire appris par l'approche proposée en figure 4.7.

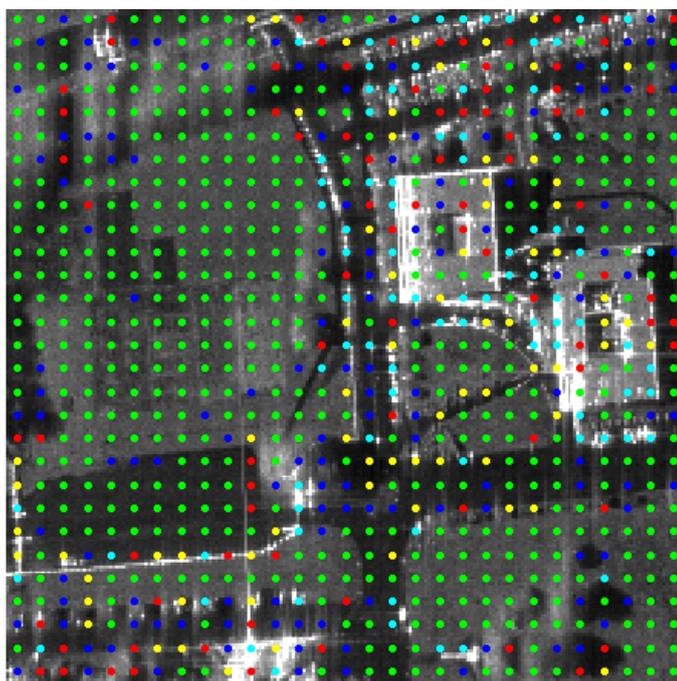


FIGURE 4.11 – Carte du degré de parcimonie optimale k au sens de l'écart quadratique moyen (EQM). Chaque point est associé à un patch et sa couleur représente le degré de parcimonie optimale $k = 1, 2, 3, 4, 5$. Cette carte est obtenue après réduction de speckle d'une centaine de réalisations bruitées de patches d'une image 100 vues avec une OMP k -sparse $k \in \{1, \dots, 5\}$ en utilisant un dictionnaire K -SVD appris sur des données radar. Il s'agit du même protocole que l'expérience décrite en section 3.4.1.

où x_i est le i -ème pixel de \mathbf{x} .

Nous avons également introduit en section 3.2.2 un a priori invariant par translation. En utilisant l'attache aux données Fisher-Tippett et cet a priori, le problème de restauration d'une image radar avec l'EPLL devient :

$$\min_{\mathbf{x}} \lambda \sum_{i=1}^N (e^{y_i - x_i} + x_i - y_i) + \min_{j \in \mathcal{N}(i), \mathbf{z}_i} \left\{ \frac{\delta}{2} \|\mathbf{R}_j \mathbf{x} - \mathbf{z}_j\|^2 - \log p(\mathbf{z}_j) \right\} \quad (4.14)$$

où \mathbf{R}_j est l'opérateur d'extraction du j -ème patch d'une image et p s'écrit :

$$p(\mathbf{z}_i) \simeq \max_k \omega_k \frac{1}{(2\pi)^{w^2/2} \det(\mathbf{C}_k)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{z}_i)^t \mathbf{C}_k^{-1}(\mathbf{z}_i)\right) \quad (4.15)$$

Ce problème peut être résolu en alternant la minimisation en \mathbf{x} à \mathbf{z}_i fixé et en \mathbf{z}_i et j à \mathbf{x} fixé. La résolution de problème en \mathbf{z}_i est la même qu'avec l'EPLL classique et est rappelée dans l'algorithme . La résolution du problème en \mathbf{x} peut être réalisée efficacement par un schéma numérique itératif. Remarquons que la structure particulière des matrices \mathbf{R}_j permet de réécrire la somme de cette façon :

$$\sum_i \|\mathbf{R}_{j_i^*} \mathbf{x} - \mathbf{z}_i\|^2 = \sum_i c_i (x_i - \bar{z}_i)^2 \quad (4.16)$$

où j_i^* est l'indice du patch couvrant le pixel i convenant le mieux à l'a priori, $\bar{\mathbf{z}} = \text{diag}(\mathbf{c})^{-1} \sum_i \mathbf{R}_{j_i^*}^t \mathbf{z}_i$ est la reprojction uniforme des patches \mathbf{z}_i dans le domaine de l'image, autrement dit, l'image obtenue en moyennant les valeurs des patches se projetant en chaque pixel, et c_i est le nombre de patches $\mathbf{R}_{j_i^*} \mathbf{x}$ se projetant sur le pixel i .

Ainsi, résoudre le problème (4.14) revient à minimiser en \mathbf{x} la fonction séparable suivante :

$$\sum_{i=1}^N \left[\lambda (e^{y_i - x_i} + x_i - y_i) + \frac{\delta}{2} c_i (x_i - \bar{z}_i)^2 \right] \quad (4.17)$$

Contrairement au cas de l'attache aux données quadratique, l'équation (4.17) n'a pas de solution analytique. En effet les auteurs de [Bioucas-Dias et Figueiredo (2010)] ont mentionné que cette solution peut s'exprimer avec la fonction Lambert W, définie dans [Corless *et al.* (1996)], dont l'évaluation est gourmande en temps de calcul. Au lieu de cela, puisque (4.17) est strictement convexe et différentiable en \mathbf{x} , on peut résoudre le problème par une méthode de Newton accomplissant à l'itération $t + 1$ la mise à jour suivante pour chaque pixel i :

$$x_i^{(t+1)} = x_i^{(t)} - \frac{\lambda(1 - e^{y_i - x_i^{(t)}}) + \delta c_i (x_i^{(t)} - \bar{z}_i)}{\lambda e^{y_i - x_i^{(t)}} + \delta c_i} \quad (4.18)$$

Puisque ce problème est séparable, plusieurs itérations peuvent être réalisées avec un coût négligeable comparé à l'optimisation du problème MAP pour tous les \mathbf{z}_i . Dans notre cas, quelques itérations sont suffisantes pour atteindre une bonne solution. La procédure résultante est récapitulée dans l'algorithme 4.

Algorithm 4 SAR-SI-EPLL

Require: N_{pa} : nombre de patches, N_{pi} : nombre de pixels, N_{it} : nombre d'itérations,

Initialisation : $\mathbf{x}_i = \mathbf{y}_i, \forall i \in \{1, \dots, N_{\text{pa}}\}$

for $\delta = [1, 4, 8]/\psi(1, L)$ (cf. eq. (4.3)) **do**

for $i \in \{1, \dots, N_{\text{pa}}\}$ **do**

$$k^* = \underset{k}{\operatorname{argmin}} -2 \log(\omega_k) + \log(\det(\mathbf{C}_k)) + (\mathbf{R}_i \mathbf{x})^t \mathbf{C}_k^{-1} (\mathbf{R}_i \mathbf{x})$$

$$\mathbf{z}_i = (\mathbf{C}_{k^*} + \sigma^2 \mathbf{I})^{-1} \mathbf{C}_{k^*} \mathbf{y}_i$$

$$j_i^* = \underset{j \in \mathcal{N}(i)}{\operatorname{argmin}} \|\mathbf{R}_j \mathbf{x} - \mathbf{z}_j\|^2 - \log p(\mathbf{z}_j)$$

end for

$$\bar{\mathbf{z}} = \operatorname{diag}(\mathbf{c})^{-1} \sum_{i=1}^{N_{\text{pa}}} \mathbf{R}_{j_i^*}^t \mathbf{z}_i$$

for $t = 0$ to $N_{\text{it}} - 1$ **do**

for $p \in \{1, \dots, N_{\text{pi}}\}$ **do**

 Mettre à jour $x_p^{(t+1)}$ avec l'équation (4.18)

end for

end for

end for

4.3.2 Résultats

Nous présentons dans les figures 4.12 et 4.13 quelques résultats illustrant l'intérêt de l'invariance par translation et d'une attache aux données adaptée au bruit multiplicatif pour la restauration d'images SAR haute résolution. Ces résultats sont obtenus à partir d'une image radar 2-vues en utilisant l'a priori GMM appris dans [Zoran et Weiss (2011)]. Dans la figure 4.12, on compare l'EPLL appliqué à l'image log-transformée à notre méthode avec et sans invariance par translation. Ces résultats sont satisfaisants et préservent les cibles brillantes, ce qui est nécessaire dans ce type d'imagerie. On observe que notre méthode ne présente pas le défaut des pixels foncés isolés rencontré avec des méthodes par transformation logarithmique (zones entourées en rouge). On voit également que les contours sont mieux restaurés avec invariance tandis qu'ils sont plus flous sans (zones entourées en orange).

La figure 4.13 compare notre approche à quelques méthodes de l'état de l'art telles que [Deledalle *et al.* (2015)] et [Makitalo *et al.* (2010)]. Les PSNR des différentes approches sont très proches et le résultat de notre approche est très proche visuellement de celui de

[Deledalle *et al.* (2015)]. Néanmoins on peut observer visuellement un gain par rapport à [Makitalo *et al.* (2010)].

4.4 Apprentissage d'un GMM radar

Dans cette section, nous apprenons un GMM sur des images SAR afin de comparer les performances en terme de réduction de speckle avec l'EPLL en utilisant le GMM appris dans [Zoran et Weiss (2011)] ou celui appris sur des données SAR. Cette expérience permettra d'évaluer dans une tâche de débruitage si des modèles Gaussiens appris sur des images naturelles représentent aussi bien les images SAR que des modèles appris sur des images SAR transformées logarithmiquement. Nous commençons par décrire notre protocole d'apprentissage en section 4.4.1 puis nous évaluons l'apport de ce dernier en section 4.4.2.

4.4.1 Protocole d'apprentissage

Nous décrivons dans ce paragraphe le protocole d'apprentissage d'un GMM sur des images SAR avec un algorithme EM (Expectation-Maximization) [Dempster *et al.* (1977)].

Voici quelques détails sur l'implémentation :

- Notre base d'apprentissage est constituée de patches extraits de trois images SAR 100-vues (donc quasiment sans bruit) de taille 409×409 transformées logarithmiquement.
- En vue de réaliser une comparaison avec le GMM appris dans l'EPLL, le GMM appris est de vecteurs moyens nuls. Les patches utilisés sont donc centrés au préalable.
- Nous n'avons pas appris un GMM de 200 modèles Gaussiens mais de 98 modèles car nous avons observé, en débruitant plusieurs images SAR, qu'il s'agissait des seuls utilisés.
- On initialise l'algorithme EM avec ces 98 modèles issus du GMM appris dans l'EPLL.
- Le nombre d'itérations de l'algorithme EM est égal à 100

Nous avons calculé l'évolution de la divergence de Kullback-Leibler (KL) entre les Gaussiennes de l'initialisation et celles apprises à chaque itération. Il s'agit de la figure 4.14(a). Beaucoup de Gaussiennes (dès les premières itérations) deviennent très différentes de celles apprises par les auteurs de l'EPLL. On observe la même chose sur l'évolution des poids sur la figure 4.14(b). On note que peu d'itérations suffisent avant que les Gaussiennes n'arrêtent d'évoluer. Par contre, ce ne sont pas forcément les Gaussiennes dont les poids augmentent qui auront des divergences de KL qui vont augmenter. En effet, une Gaussienne à poids faible peut tout de même avoir de l'importance si elle encode des structures rares.

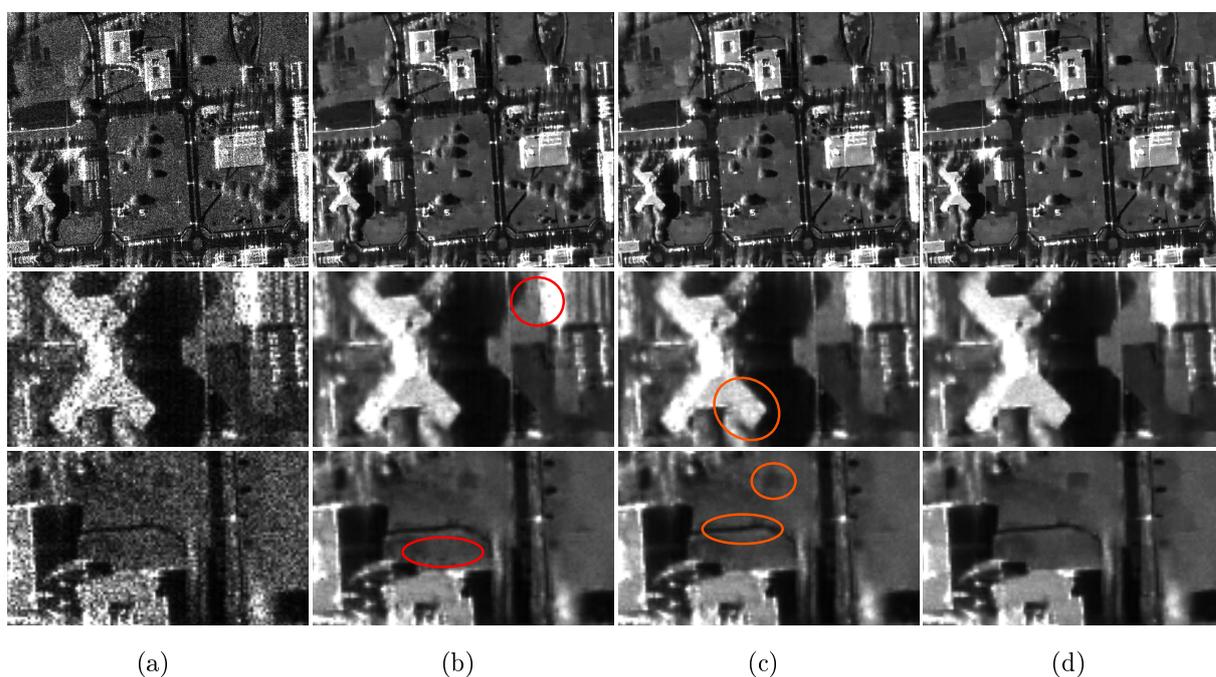


FIGURE 4.12 – (a) Image radar 2 vues de Toulouse ©ONERA ©CNES (Les deux images en bas sont des zooms de celles du dessus). (b) Résultat de l'EPLL appliqué après une transformation logarithmique, avec une étape de post-débiaising. Les ellipses rouges désignent des pixels foncés isolés correspondant à un défaut souvent rencontré avec les méthodes de débruitage développées pour du bruit Gaussien. (c-d) Notre résultat avec une attache aux données Fisher-Tippett sur les données transformées logarithmiquement sans et avec invariance respectivement. Les ellipses orange désignent des zones où l'on remarque le plus l'absence de l'invariance par translation.

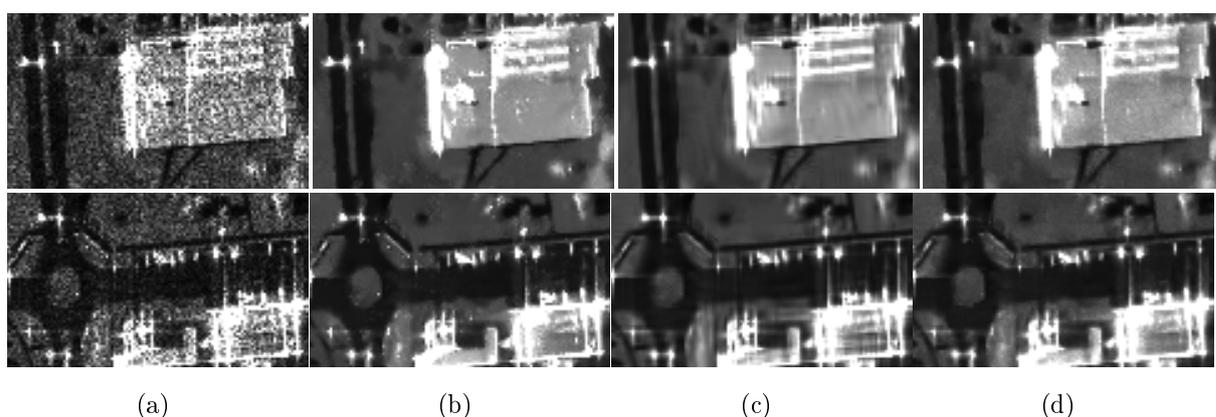


FIGURE 4.13 – (a) Deux zooms de la même image radar que dans la Figure 4.12. Comparaison de résultats de restauration entre : (b) le filtre NL-SAR *Deledalle et al. (2015)* (c) le filtre BM3D sur les données transformées logarithmiquement *Makitalo et al. (2010)*, et (d) notre approche.

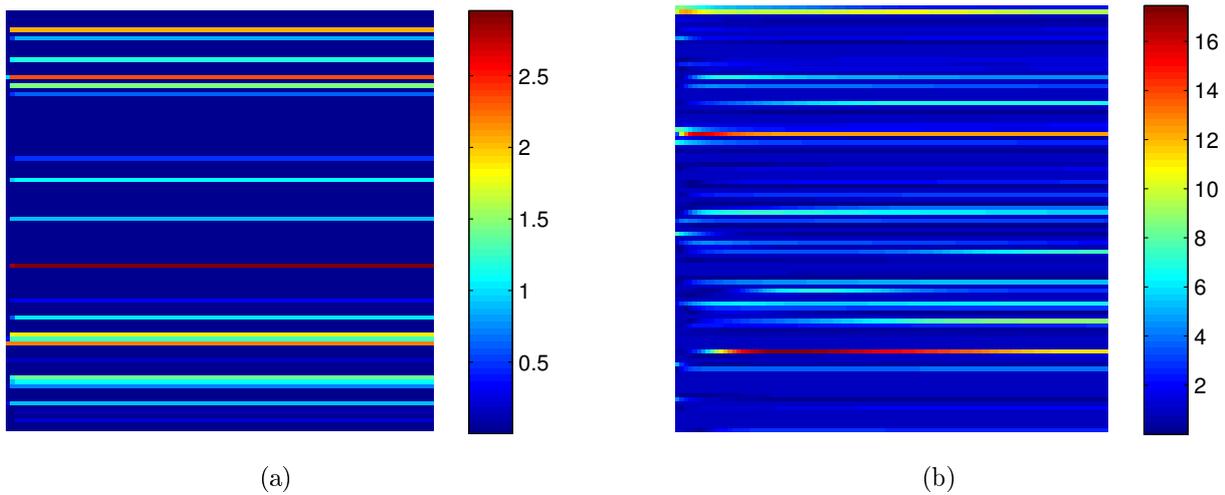


FIGURE 4.14 – Dans les deux figures, chaque colonne représente une itération de l'EM et chaque ligne un modèle gaussien. À gauche : évolution de la divergence de KL à chaque itération de l'EM entre les gaussiennes de Zoran et al. et les gaussiennes courantes. À droite : évolution de la différence relative entre les poids des modèles gaussiens de Zoran et al. et les poids courants à chaque itération.

4.4.2 Évaluation de l'apport de ce GMM

Nous montrons en figure 4.15 des patches générés par un modèle Gaussien à forte divergence de KL. Ce modèle est intéressant car la structure générée est une cible brillante (un sinus cardinal) qui est souvent observée et difficile à reconstituer dans les images SAR. Malheureusement, en pratique, dans une application au débruitage, nous constatons qu'elle n'est pas utilisée pour représenter les points brillants. En effet, en figure 4.18, on observe que l'algorithme préfère utiliser plusieurs Gaussiennes pour les encoder.

Dans la suite, nous réalisons une comparaison des performances en réduction de speckle par l'EPLL-FT (l'EPLL avec une attache aux données Fisher-Tippett et sans invariance par translation, voir section 4.3) entre le GMM appris sur des images SAR, qu'on appellera SAR-GMM, et le GMM appris dans [Zoran et Weiss (2011)]. Les résultats obtenus en figures 4.16 et 4.17 montrent qu'avec le SAR-GMM, le PSNR est légèrement meilleur. De plus, le léger grain observé dans les zones homogènes de l'image 100 vues est retrouvé alors qu'il était complètement lissé avec le GMM appris sur des images naturelles.

Compte tenu du peu d'images 100-vues dont nous disposons, nous avons mené des tests sur une base d'apprentissage élargie. Pour cela, nous avons utilisé des images SAR dont nous avons atténué le bruit puis appliqué des transformations simples, par exemple des symétries. Pour atténuer le bruit nous avons calculé une moyenne sur une pile d'images SAR acquises à différentes dates. Cependant, les conclusions obtenues restent les mêmes :

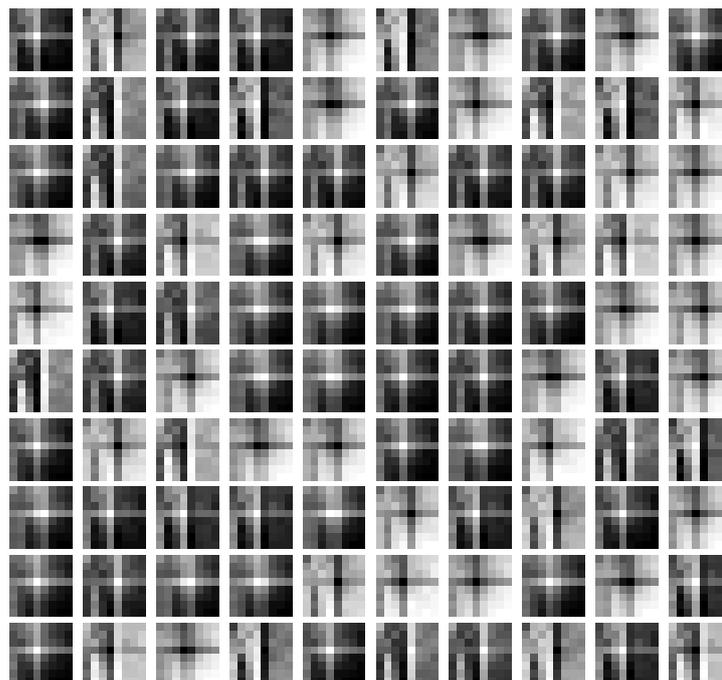


FIGURE 4.15 – Patches générés par un des modèles Gaussiens appris sur des images radar 100 vues qui présente une forte divergence de KL. Il s’agit d’une signature typique d’une cible ponctuelle. De tels patches ne sont pas/peu présents dans les images optiques naturelles.

apprendre un GMM sur des images SAR améliore les résultats de débruitages en comparaison avec ceux obtenus avec un GMM appris sur des images naturelles mais l’écart observé n’est pas significatif.

4.5 Une méthode de classification basée sur les GMM

Dans cette section, nous présentons une contribution de nos travaux de thèse à la classification d’images SAR en utilisant des GMM [Tabti *et al.* (2015)]. La classification d’image est l’association de chaque pixel de l’image d’intérêt à une classe ou label, à ne pas confondre avec la segmentation d’images qui consiste à la partitionner en plusieurs régions [Galland *et al.* (2004)], [Ben Ayed *et al.* (2005)]. La classification est un domaine important en télédétection au vu de ses nombreuses applications, par exemple l’évaluation des conséquences d’une catastrophe naturelle ou encore l’observation de la déforestation. Nous présentons dans la section 4.5.1 quelques méthodes de classification d’images SAR en nous concentrant sur celles qui utilisent des mélanges de distributions qui sont plus proches de la méthode que nous proposons. La section 4.5.2 décrit notre procédure de classification supervisée divisée en une étape d’apprentissage et une étape de test.

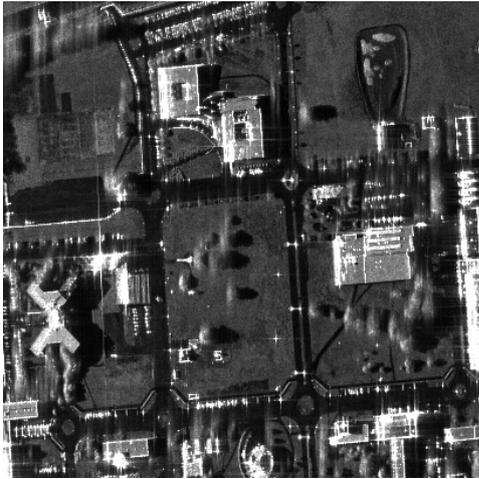


Image SAR 100-vues

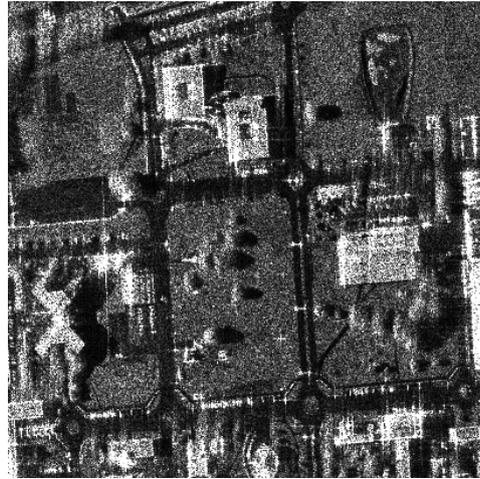


Image bruitée 1-vue



EPLL-FT PSNR, = 42.97dB



EPLL-FT + SAR GMM, PSNR = 43.38dB

FIGURE 4.16 – Comparaison des résultats de l'EPLL-FT entre un GMM classique et un GMM appris sur des données SAR. On observe que le GMM appris sur des données SAR donne de meilleures performances en terme de PSNR. Néanmoins les deux résultats sont satisfaisants visuellement.

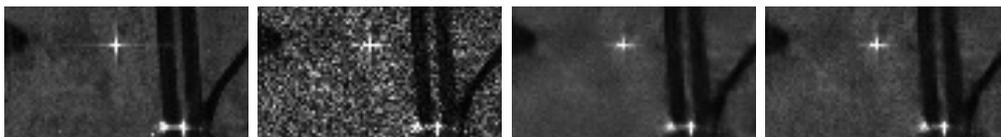


FIGURE 4.17 – Zooms sur des protions des images de la figure 4.16. De gauche à droite : l'image 100-vues, 1-vue, débruitée par l'EPLL-FT avec le GMM appris dans [Zoran et Weiss (2011)] et débruitée avec l'EPLL-FT et le GMM appris sur des données SAR où on observe le même léger grain que dans l'image 100-vues.

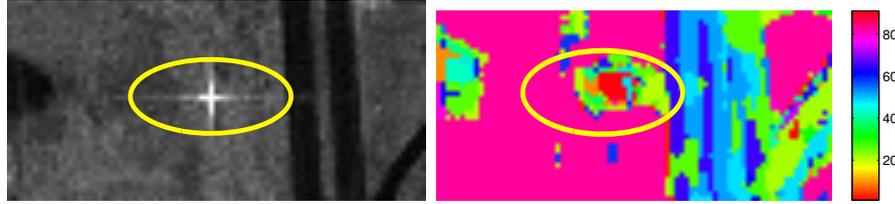


FIGURE 4.18 – À droite : carte des modèles Gaussiens (définies notamment en figure 2.19) utilisés au niveau du point brillant sur l'image de gauche entouré en jaune. On observe qu'un point brillant n'est pas représenté par un unique modèle Gaussien mais plusieurs.

4.5.1 Quelques méthodes de classification d'images SAR

En télédétection, la classification d'images est un domaine actif. En particulier, la classification d'une image SAR en amplitude est un problème difficile à cause du speckle qui la caractérise et nécessite souvent un pré-filtrage. Pour éviter cela, plusieurs méthodes existantes fusionnent les informations de plusieurs canaux d'images polarimétriques [Lardeux *et al.* (2009)] ou d'une image en amplitude avec une image optique (ou autre capteur) [Voisin *et al.* (2014)] ou encore utilisent des images multi-temporelles.

Les algorithmes de classification d'images SAR visent à modéliser l'amplitude et la texture observées dans l'image. Ils s'inspirent de différents domaines de recherche. Certains se basent sur des techniques d'apprentissage statistique et apprennent des descripteurs, que nous décrivons dans la suite de cette section, avec par exemple des forêts aléatoires [Yang *et al.* (2009)] ou des Support Vectors Machines (SVM) [Lardeux *et al.* (2009)].

La distribution standard modélisant l'amplitude du bruit d'une image SAR est celle de Nakagami-Rayleigh comme nous l'avons mentionné en section 2.1.2 et c'est le choix effectué dans [Kayabol et Zerubia (2013)]. Dans [Tison *et al.* (2004)], la distribution de Fisher est utilisée pour mieux représenter les zones urbaines. Dans [Voisin *et al.* (2010)], un dictionnaire de familles paramétriques classiques en imagerie SAR est proposé, il inclut les distributions log-normal, Weibull, Nakagami-Rayleigh et gamma Généralisée. Dans [Frery *et al.* (1997)], une distribution notée \mathcal{G}_A^0 est proposée. La loi \mathcal{K} est un cas particulier de ce type de distribution qui est capable de représenter des zones très hétérogènes. Des cartes des paramètres de cette distribution constituent des descripteurs utiles en classification d'images SAR, comme il a été proposé notamment dans [Mejail *et al.* (2003)].

Parmi les descripteurs utilisés afin de modéliser la texture des images SAR, citons le semivariogram qui décrit les propriétés spatiales de l'image et qui est calculé avec une fenêtre glissante. Si i et j sont deux positions de pixels adjacents séparés d'un pas h , il

est défini par :

$$S(h) = \sum_{(i,j) \in \mathcal{N}_h} \frac{|x_i - x_j|^2}{\mathcal{N}_h} \quad (4.19)$$

où x_i est l'intensité du i -ème pixel de l'image \mathbf{x} et \mathcal{N}_h est l'ensemble de paires d'observations.

Une matrice de co-occurrence de niveaux de gris, ou GLCM pour Grey Level Co-occurrence Matrix en anglais, est un descripteur de texture. Il est défini par une matrice $Z \times Z$, où Z est le nombre de niveaux de gris dans l'image, dont chaque pixel représente l'estimation de la probabilité qu'un pixel de valeur i soit voisin pour un certain pas h d'un pixel de valeur j . Les voisins considérés sont en pratique ceux de la 4-connexité. L'adjacence des pixels peut être définie dans n'importe quelle direction : gauche, droite, horizontale, verticale, diagonale. La texture est extraite de la matrice en calculant une variance avec une fenêtre glissante de taille 5×5 par exemple. Ce descripteur est plus adapté aux zones urbaines que les semivariogrammes d'après [Voisin *et al.* (2010)] et fusionné avec un descripteur basé sur des filtres de Gabor, il présente de bons résultats de classification dans [Ismail *et al.* (2014)].

La texture peut également être décrite par un modèle Auto-Régressif (AR) afin d'introduire l'information contextuelle dans le processus de classification. On l'écrit pour une classe k :

$$x_n = \sum_{n' \in \mathcal{V}(n)} \alpha_{k,n'} x_{n'} + t_{k,n} \quad (4.20)$$

où $\mathcal{V}(n)$ sont les indices des pixels voisins du n -ème pixel d'intensité x_n , $\alpha_{k,n'}$ sont les coefficients d'auto-régression, $t_{k,n}$ sont des variables aléatoires t -distribuées représentant les erreurs de régression.

Ce modèle est utilisé notamment dans [Kayabol et Zerubia (2013)] où l'on observe une amélioration des performances de classification sur des images contenant plusieurs textures.

Pour éviter un manque de cohérence spatiale sur la carte de classification, des modèles de régularisation sont souvent nécessaires. Les modèles markoviens sont le plus souvent utilisés dans ce but, comme par exemple dans [Tison *et al.* (2004)], ainsi que des modèles markoviens hiérarchiques, par exemple dans [Yong *et al.* (2006)].

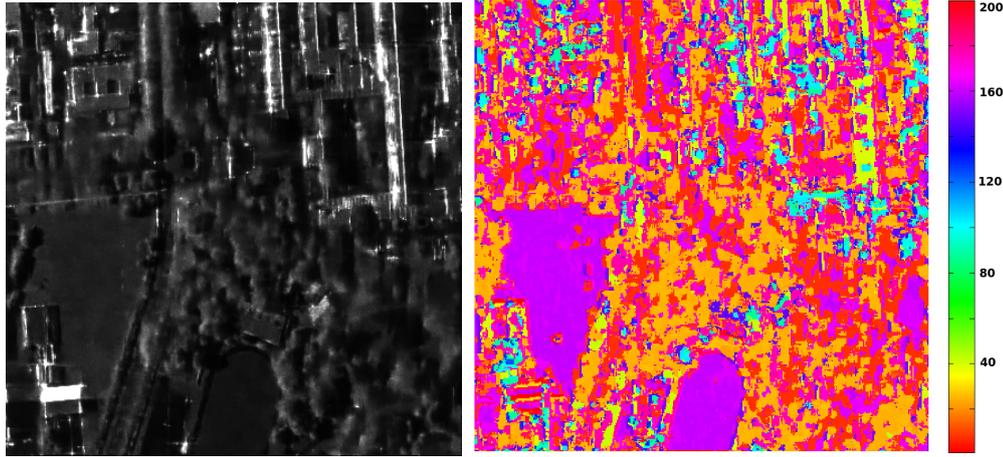


FIGURE 4.19 – De gauche à droite : une image SAR débruitée par le SAR-EPLL et la carte des modèles Gaussiens associée. On observe que ces modèles représentent bien les structures observées dans une image SAR, par exemple, les mêmes modèles Gaussiens sont utilisés pour représenter les zones homogènes.

4.5.2 Principe de la méthode proposée

Nous proposons d'utiliser les GMM dans une application à la classification. En effet, les GMM capturent une information de texture locale et constituent donc de bons candidats en tant que descripteurs comme il est illustré en figure 4.19. La procédure de classification supervisée se divise en deux étapes. La première étape consiste à apprendre deux descripteurs hors ligne. Le premier est basé sur les GMM et le second sur la radiométrie des images. Pour cela, on doit disposer d'une base d'images SAR et des cartes de modèles Gaussiens (définies en section 2.4.3.1). On utilise le GMM de moyenne nulle appris dans [Zoran et Weiss (2011)] et bien qu'il soit appris sur des images naturelles, comme nous l'avons montré en section 4.4, il est aussi performant pour restaurer des images SAR qu'un GMM appris sur des images SAR. On doit disposer également d'une labellisation manuelle des images de la base d'apprentissage. Ces éléments permettent de calculer les fréquences de chaque modèle Gaussien du GMM pour chaque label. On procède de la même manière pour la radiométrie, on calcule pour chaque label la fréquence de chaque radiométrie sur les images de la base quantifiées. Dans la deuxième étape on classe chaque pixel dans un cadre Bayésien basé sur le modèle Gaussien et la radiométrie lui correspondant ainsi que son voisinage.

Apprentissage des descripteurs pour la classification Pour réaliser la phase d'apprentissage de l'algorithme de classification, on commence par restaurer un ensemble de huit images d'amplitude avec une résolution d'un mètre du satellite TerraSAR-X en utilisant l'EPLL-FT (l'EPLL avec une attache aux données Fisher-Tippett et sans invariance par translation) décrit en section 4.3. Les tailles de ces images sont entre 1000×1000 et 2048×2048 pixels et elles sont labellisées manuellement en $N_{\mathcal{L}} = 3$

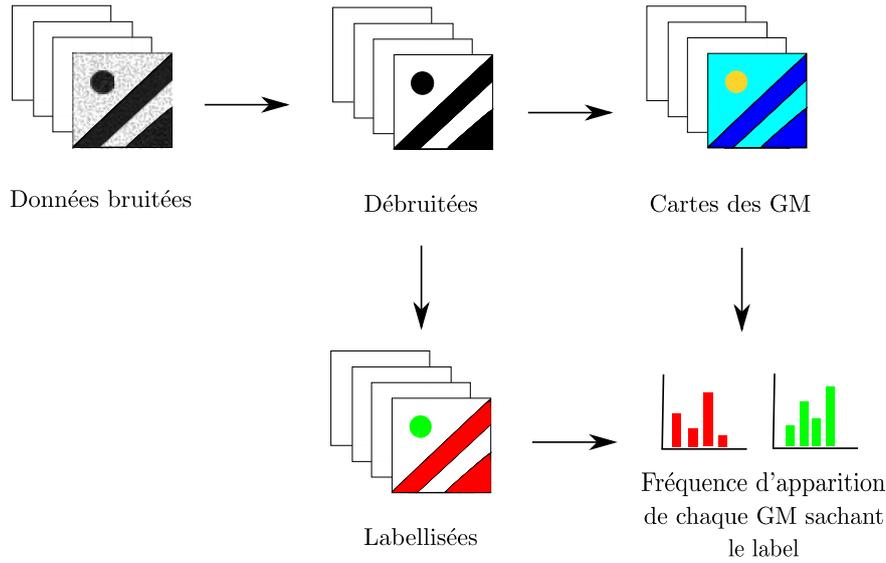


FIGURE 4.20 – Illustration de la phase d'apprentissage.

classes : zone urbaine, végétation haute, zone homogène (végétation basse et eau). Ces images et leurs labellisations manuelles peuvent être consultées en annexe B.

Ensuite la carte des modèles Gaussiens de chacune de ces images introduite en section 2.4.3.1 est utilisée pour calculer pour chaque label \mathcal{L} la fréquence d'apparition $p(\mathbf{d}_k|\mathcal{L})$ de chaque modèle Gaussien \mathbf{d}_k défini par une matrice de covariance et un poids : $\mathbf{d}_k = \{\mathbf{C}_k, \omega_k\}_{k \in \{1, \dots, K\}}$. La figure 4.21 montre ces fréquences pour chaque modèle Gaussien et pour chaque label.

Puisque le GMM a été appris sur des patchs centrés, il est nécessaire de considérer la radiométrie des pixels comme second descripteur. Les images de la base d'apprentissage sont donc quantifiées en $Q = 10$ niveaux de gris et le même procédé que pour les modèles Gaussiens est appliqué pour apprendre la fréquence $p(\nu_q|\mathcal{L})$ de chaque niveau de radiométrie ν_q , $q \in \{1, \dots, Q\}$, pour chaque label. La figure 4.22 montre les fréquences des niveaux de gris pour pour chaque label.

L'assignation des labels Comme dans la phase d'apprentissage, chaque pixel i de l'image d'intérêt est représenté par son modèle Gaussien correspondant \mathbf{d}^i (obtenu avec notre adaptation de l'EPLL) et son niveau de radiométrie quantifiée ν^i . Étant données les fréquences apprises au préalable de chaque modèle Gaussien et de chaque niveau de gris conditionnellement aux labels et en supposant que la radiométrie d'un patch est indépendante de la structure capturée par son atome correspondant, une log-vraisemblance pour chaque label peut être assignée à chaque pixel i comme suit :

$$\log p(\mathbf{d}^i = \mathbf{d}_k, \nu^i = \nu_q|\mathcal{L}) = \log p(\mathbf{d}_k|\mathcal{L}) + \log p(\nu_q|\mathcal{L}) \quad (4.21)$$

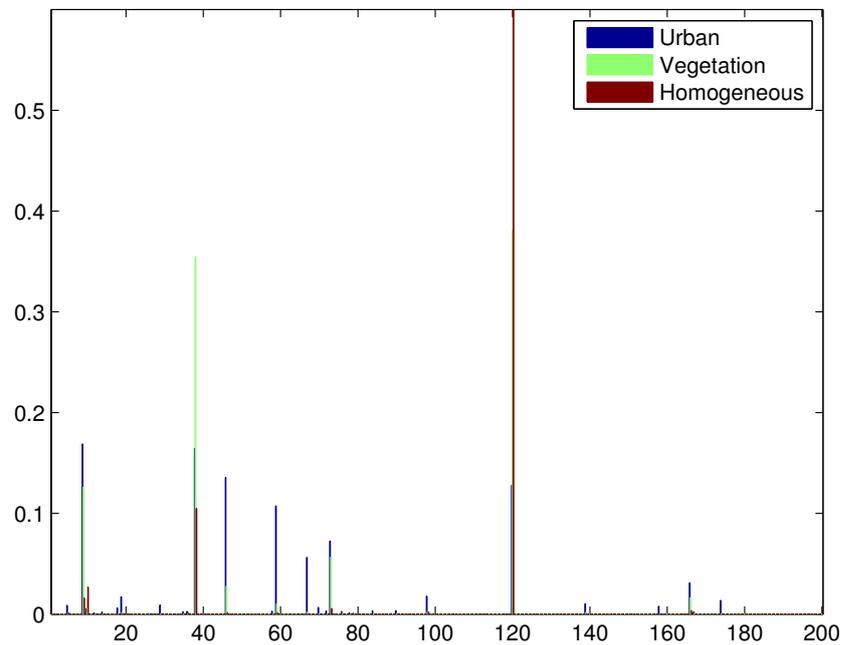


FIGURE 4.21 – *Fréquence d'apparition de chacune des 200 Gaussiennes dans le GMM pour chaque label. On constate que pour chaque label les modèles Gaussiens répondent différemment ce qui permettra de distinguer les classes lors du processus de classification.*

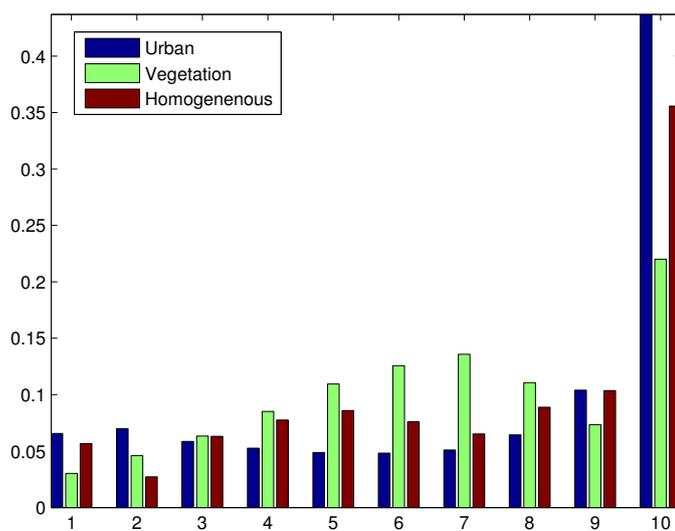


FIGURE 4.22 – *Fréquence d'apparition de chacun des 10 niveaux de gris pour chaque label. On constate que pour chaque label la plupart des niveaux de gris répondent différemment.*

Cette quantité représente la probabilité d'un pixel i d'appartenir à une classe parmi les labels \mathcal{L} .

Pour éviter un manque de cohérence spatiale sur la carte de classification, on suppose également que la classe d'un pixel dépend de la classe affectée à ses pixels voisins. En se plaçant dans un cadre Bayésien conjointement à une régularisation de Potts, cela mène à affecter le label \mathcal{L}^i à chaque pixel i maximisant la probabilité a posteriori tel que l'expression suivante est minimisée :

$$-\sum_i \log p(\mathbf{d}^i, \nu^i | \mathcal{L}^i) + \lambda \sum_{i \sim j} \delta(\mathcal{L}^i, \mathcal{L}^j) \quad (4.22)$$

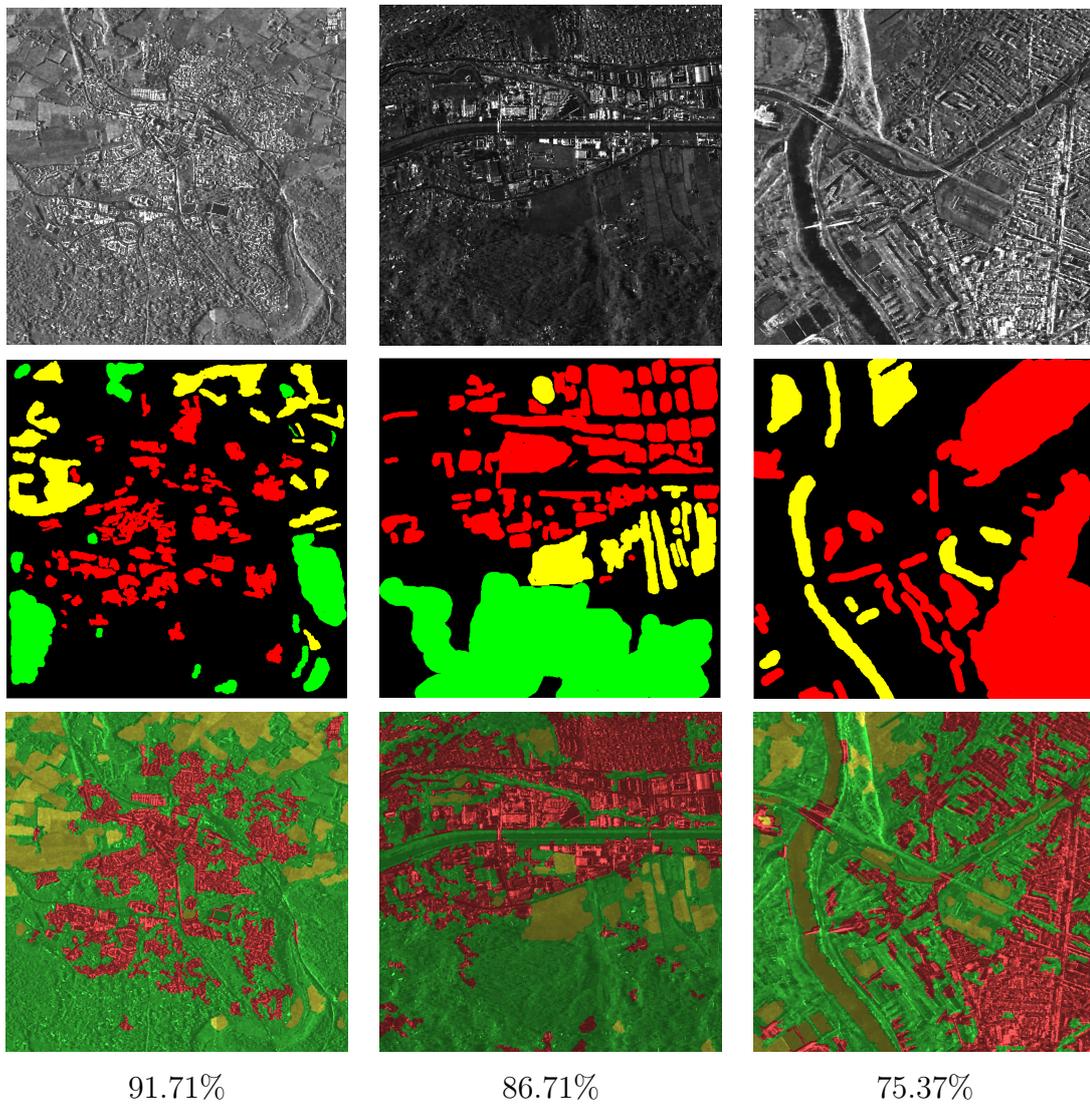
où $\lambda > 0$, $i \sim j$ correspondent aux indices des pixels voisins en système de huit-connexité et $\delta(\mathcal{L}^i, \mathcal{L}^j) = 1$ si $\mathcal{L}^i = \mathcal{L}^j$ et 0 sinon.

Des solutions satisfaisantes de tels problèmes d'optimisation discrète peuvent être obtenues itérativement par coupure de graphes (graph-cuts) avec la stratégie $\alpha - \beta$ swap décrite dans [Boykov *et al.* (2001)].

Résultats Des résultats de classification sur trois images d'amplitude TerraSAR-X sont présentés en figure 4.23. Ils montrent que les descripteurs basés sur un GMM réussissent à représenter la texture et l'amplitude d'une image radar. Ces résultats préliminaires sont prometteurs sur les images testées et présentent des pourcentages de pixels correctement classifiés au dessus de 75% par rapport à une vérité terrain labellisée manuellement. Le pourcentage de 75% est plus faible que les autres car il comporte une zone urbaine avec peu de structures brillantes alors que dans notre base d'apprentissage, les zones urbaines en comportaient toujours beaucoup. Dans nos travaux futurs, nous envisageons donc d'élargir notre base d'apprentissage et d'utiliser une technique d'apprentissage plus sophistiquée comme les SVM par exemple.

4.6 Conclusion

Nous avons abordé dans ce chapitre diverses applications aux images SAR des travaux de thèses présentés dans les chapitres précédents. Dans le cadre des approches par dictionnaires, nous avons utilisé les cartes d'erreurs d'approximation avec invariances par translation et transformation affine, introduites en section 3.4.4, pour élaborer un algorithme d'apprentissage de dictionnaire 1-sparse. Le problème d'optimisation associé à cet algorithme n'étant pas facile de premier abord, une stratégie efficace a été élaborée afin de remplacer au fur et à mesure les éléments du dictionnaire courant par des candidats qui diminuent l'erreur d'approximation moyenne de l'image. Les dictionnaires étant appris



Légende : **Zone urbaine**, **Végétation haute**, **Végétation basse et eau**, Non labellisé

FIGURE 4.23 – De la première à la dernière ligne : images d’amplitude TerraSAR-X débruitées par l’EPLL-FT, la vérité terrain, les résultats de classification et les pourcentages de pixels correctement classés par rapport aux pixels labellisés dans la vérité terrain.

sur une image de référence, on observe qu'ils résument bien les structures présentes dans cette image. Une application à la mise en correspondance de patches sans bruit dans une image SAR a été présentée.

Par rapport aux approches basées sur les GMM, plus efficaces en débruitage que des approches 1-sparse, nous avons montré que les performances en débruitage avec un GMM appris sur des images SAR étaient légèrement meilleures mais du même ordre que celles avec un GMM appris sur des images naturelles.

Nous avons également étendu l'algorithme EPLL [Zoran et Weiss (2011)] développé pour le bruit Gaussien au speckle tout en introduisant une propriété d'invariance par translation en utilisant l'a priori présenté en section 3.2.2. Les performances de cet algorithme de réduction de speckle sont du même ordre que les algorithmes de l'état de l'art. On observe que l'invariance par translation améliore visuellement les résultats, en particulier au niveau de structures linéaires souvent présentes en milieu urbain.

Enfin, malgré la difficulté du problème de classification d'une image SAR en amplitude sans utiliser des informations supplémentaires d'une image satellitaire optique par exemple, nous avons proposé une méthode supervisée dont les descripteurs sont basés sur les GMM et qui présentent des résultats satisfaisants. Les descripteurs étant appris hors ligne, cette méthode est simple et rapide à mettre en œuvre une fois que la carte des modèles Gaussiens de l'image à traiter est calculée.

Chapitre 5

Conclusions et perspectives

Dans ce chapitre, nous résumons les contributions de cette thèse et discutons des travaux futurs envisagés. Nos contributions se situent à deux niveaux. Premièrement, nous avons développé de nouveaux modèles munis de propriétés d'invariances pour représenter la distribution des patches. Ces modèles sont basés sur l'apprentissage de dictionnaires que nous traitons en section 5.1, ou de GMM, que nous traitons en section 5.2. Deuxièmement, nous avons adapté ces modèles au traitement des images SAR. Les applications proposées sont la réduction de speckle et la classification et les perspectives envisagées sont décrites en section 5.3.

5.1 Approches parcimonieuses

Suite à une étude expérimentale, nous avons montré que sous des conditions de fort bruit, si on souhaite minimiser l'écart quadratique moyen entre des données sans bruit et débruitées avec une approche parcimonieuse, il semblerait que le degré de parcimonie optimal soit égal à 1. Nous avons donc commencé par élaborer un critère de similarité invariant par translation et par transformation radiométrique affine entre un patch et un atome de plus grande taille tiré d'un dictionnaire prédéfini. Ce critère est basé sur la maximisation d'une corrélation normalisée.

Ensuite, motivés par le GLR-test invariant aux transformations radiométriques affines proposé dans [Deledalle *et al.* (2013)], nous avons proposé une méthode de calcul de carte d'erreur d'approximation d'une image par un atome avec invariance par translation et aux transformations radiométriques affines. Nous avons utilisé ces cartes pour élaborer un algorithme d'apprentissage de dictionnaire 1-sparse [Tabti *et al.* (2014a)] et de débruitage d'images SAR. Cependant, bien que le dictionnaire soit proche d'être sémantique, les résultats de débruitage obtenus ne sont pas à la hauteur des performances des méthodes actuelles. En effet, les atomes appris ne parviennent pas à représenter de la texture. Pour cela il faudrait un dictionnaire de très grande taille, ce que nous souhaitons éviter. Par ailleurs, l'algorithme que nous avons proposé afin de résoudre le problème d'optimisation

associé à l'apprentissage du dictionnaire 1-sparse est assez général pour être réutilisé dans d'autres problèmes. Par exemple, il pourrait être utilisé afin d'initialiser des méthodes d'apprentissage de GMM.

5.2 Approches basées sur les GMM

Ayant constaté l'efficacité des méthodes de débruitage modélisant la distribution des patchs par des modèles de mélange de Gaussiennes, nous avons étudié ces modèles en profondeur pour les exploiter dans plusieurs applications. Par exemple, nous avons proposé un a priori exprimé sur des patchs et basé sur des GMM muni d'une propriété d'invariance par translation. Utilisé dans l'EPLL [Zoran et Weiss (2011)], il améliore visuellement la qualité du débruitage, en particulier des structures linéaires dans une image.

Nous avons également proposé une manière de modéliser un GMM muni d'invariance aux transformations radiométriques affines. Ensuite, nous avons développé une procédure d'apprentissage et de débruitage d'une version simplifiée d'un GMM muni de cette invariance : le GMM aligné (aGMM). Les performances en débruitage de l'aGMM étant encourageantes, nous avons récemment développé un GMM invariant radiométriquement qui présente des performances en débruitage encore meilleures que l'aGMM. Les premiers résultats sont présentés en figure 5.2. De plus, il permet d'atteindre l'un des principaux objectifs du développement de modèles avec des propriétés d'invariances. En effet, en consultant le tableau 5.1, le PSNR atteint avec un GMM classique composé de 200 modèles est atteint avec un GMM invariant composé d'une centaine de modèles. On peut donc apprendre des GMM de taille deux fois plus petite qu'un GMM classique. Nous envisageons plusieurs perspectives :

- Tout d'abord, nous allons fusionner l'EPLL avec un a priori invariant par translation à l'EPLL invariant aux transformations radiométriques affines (iEPLL).
- Une fois qu'on aura appris un GMM avec invariance radiométrique affine sur des images couleurs, nous envisageons d'évaluer son apport dans une application au débruitage d'images à grande gamme dynamique, ou HDR (High Dynamic Range) en anglais. Un exemple de ce type d'image est présenté en figure 5.1. Un GMM classique appris sur des images naturelles ne peut être utilisé pour débruiter une image HDR à cause de sa gamme dynamique trop grande.

5.3 Futures applications aux images SAR

Plusieurs perspectives concernant les applications aux images SAR sont à étudier. Nous avons vu en section 4.2.2 le résultat d'une application directe de l'apprentissage d'un épitome avec une méthode de référence sur une image SAR. Améliorer le résultat obtenu grâce à une méthode d'apprentissage d'épitomes adaptée aux images SAR



FIGURE 5.1 – Exemple d'image HDR. Ce type d'image ne peut être affiché que sur des écrans adaptés. Pour les visualiser on applique une opération dite de tone-mapping (mise en correspondance de la palette de couleur d'une image HDR avec une palette de couleur adaptée aux écrans classiques).

Taille des GMM		1	2	4	8	16	32	64	128	200
Barbara, $\sigma = 20$	EPLL	25.53	27.80	28.30	28.85	29.64	29.90	30.34	30.67	30.68
	aEPLL	25.52	27.78	28.28	28.98	29.44	29.81	30.46	30.93	31.04
	iEPLL	27.22	27.42	28.23	28.69	30.02	30.33	30.78	31.11	31.2
Barbara $\sigma = 30$	EPLL	23.57	25.60	25.958	26.48	27.49	27.62	28.06	28.36	28.34
	aEPLL	23.56	25.59	25.953	26.58	27.10	27.60	28.19	28.61	28.80
	iEPLL	24.78	24.90	25.37	26.01	27.39	27.81	28.42	28.74	28.92

TABLE 5.1 – Tableau de PSNR comparant des débruitages de l'image de Barbara avec différents niveaux de bruit par l'EPLL classique, avec alignement (aEPLL) et avec invariance aux transformations radiométriques affines (iEPLL). Les GMM sont appris sur une même base de patches (avec et sans invariances) et sont de différentes tailles.



FIGURE 5.2 – De gauche à droite et de haut en bas : image bruitée ($\sigma = 20$), images débruitées avec l'EPLL classique, l'aEPLL et l'iEPLL. La taille des GMM utilisée est de 200 modèles.

serait une piste de recherche intéressante, ces derniers étant invariants par translation et capables d'encoder les structures et les textures des patches extraits d'images naturelles.

Nous avons adapté l'algorithme EPLL [Zoran et Weiss (2011)] basé sur un GMM appris hors ligne sur une base de patches au traitement des images SAR et introduit une propriété d'invariance par translation à l'aide de l'a priori mentionné plus haut dans [Tabti *et al.* (2014b)]. Les performances de cette méthode sont équivalentes aux méthodes état de l'art telle que celle décrite dans [Deledalle *et al.* (2015)]. Plusieurs perspectives sont envisagées pour cette approche :

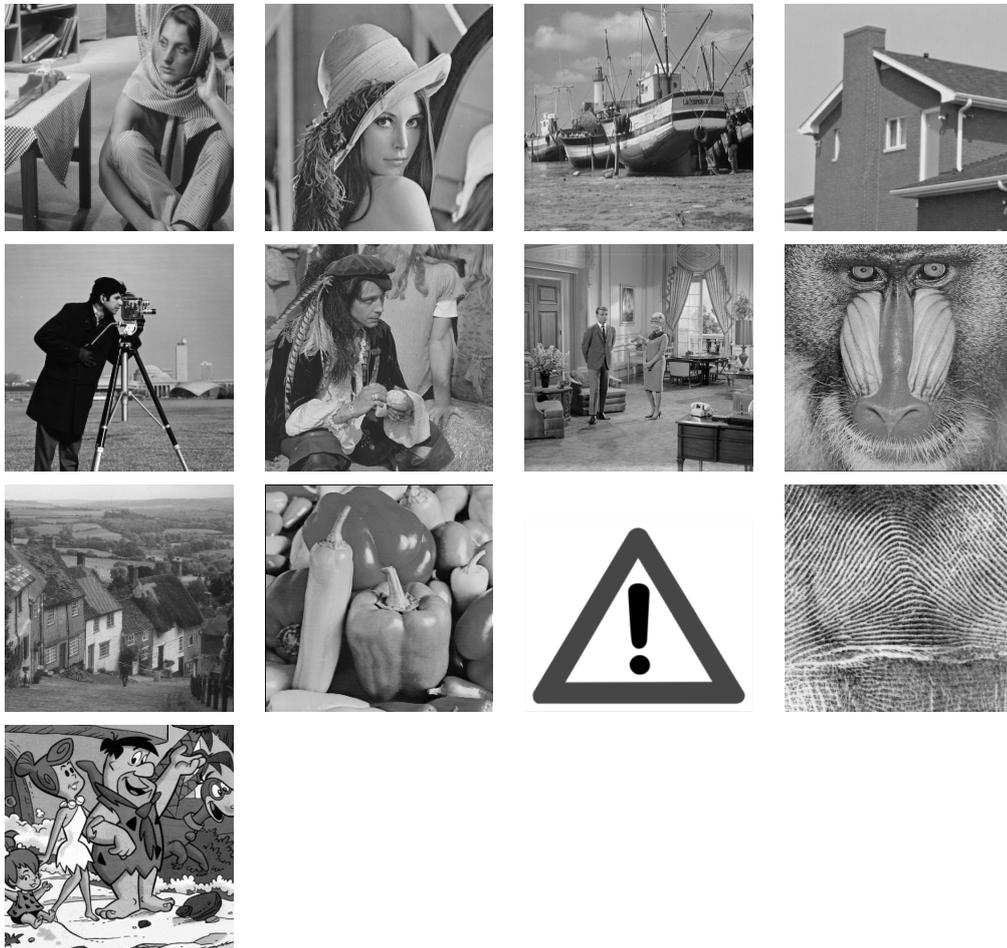
- Par exemple, on pourrait étudier le potentiel d'un filtrage d'image SAR guidé par des données optiques.
- En ce qui concerne l'apprentissage d'un GMM invariant aux transformations radiométriques affines sur des images SAR, les premiers résultats de réduction de speckle obtenus avec l'iEPLL muni d'une attache aux données Fisher-Tippett sont du même ordre que les méthodes état de l'art. Une autre perspective de ce travail est de développer l'iEPLL sur des images SAR multi-canal.

Une application à la classification supervisée des images SAR avec un descripteur basé sur des GMM a été proposée dans [Tabti *et al.* (2015)]. Il s'agit d'une méthode simple et rapide fournissant des résultats encourageants. Des améliorations peuvent lui être apportées telle que l'utilisation de méthodes d'apprentissage statistique plus sophistiquées. On pourrait également étudier la complémentarité du descripteur basé sur des GMM avec d'autres descripteurs proposés dans la littérature. Deux autres axes de recherche sont d'étudier la possibilité d'une approche multi-résolution et de la fusion d'information apportées par différents types de capteurs. Enfin, il serait également intéressant de comparer des cartes de modèles Gaussiens d'images SAR d'une même zone acquises à différentes dates afin d'étudier le potentiel des GMM dans une application à la détection de changements [Dellinger *et al.* (2014)], [Prendes *et al.* (2015)].

Annexe A

Base d'apprentissage de l'aGMM

Images de la base d'apprentissage du GMM aligné présenté en section 3.3.3.

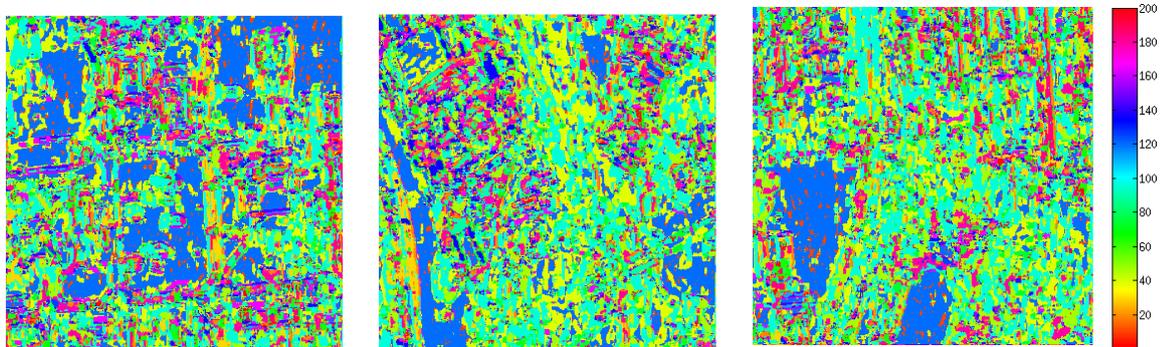
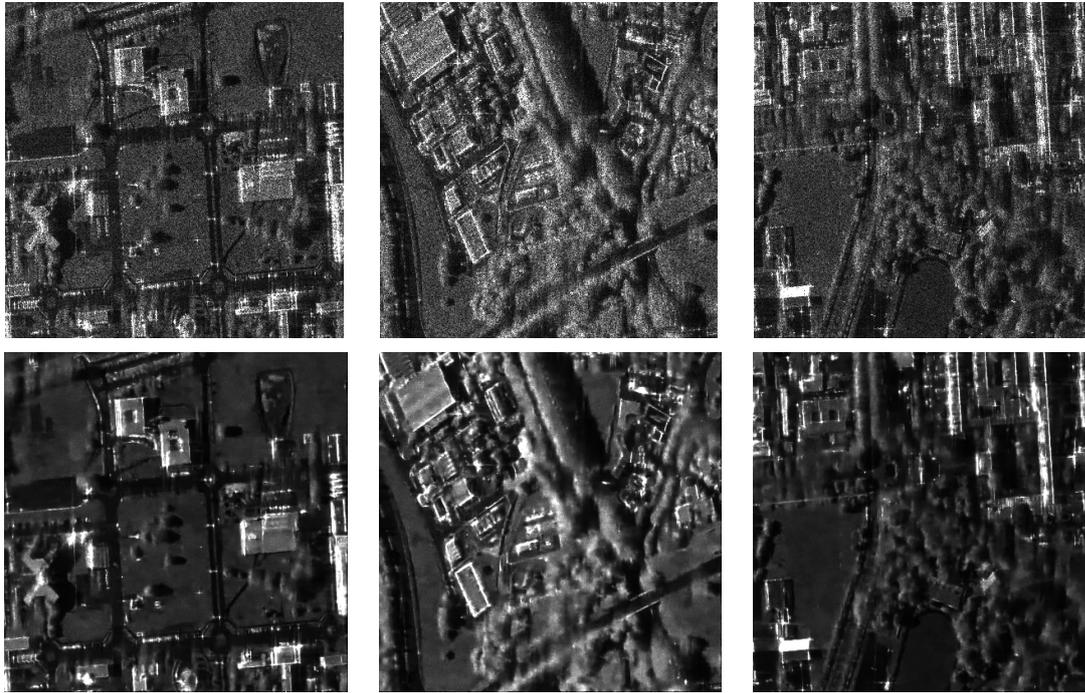


Annexe B

Base d'images de l'algorithme de classification

Nous présentons dans cette annexe dans chacune des pages suivante, en première ligne, les images SAR 1-vue, en deuxième ligne, les images débruitées par l'EPLL-FT et en troisième ligne, les cartes de modèles Gaussiens associés. Ces images constituent la base d'apprentissage et de test de la procédure de classification présentée en section [4.5.2](#).

Images de Toulouse : taille 409×409 . Images acquises par l'ONERA (aéroportées $1m \times 1m$).



Images de Marseille : taille : 2001×2001 . Images TerraSAR-X, résolution : $1m \times 1m$.
L'image de gauche fait partie de la base de tests.

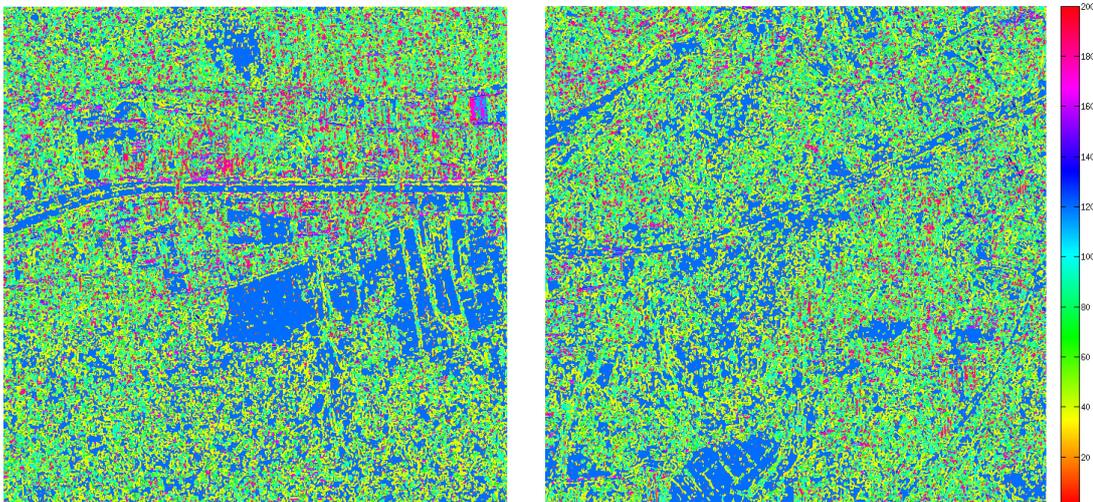
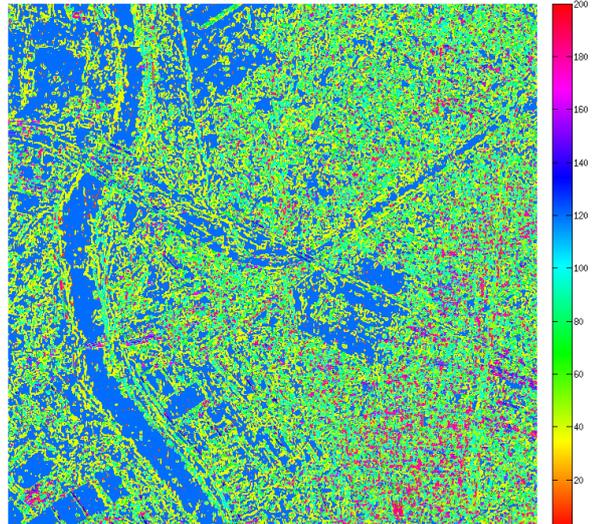
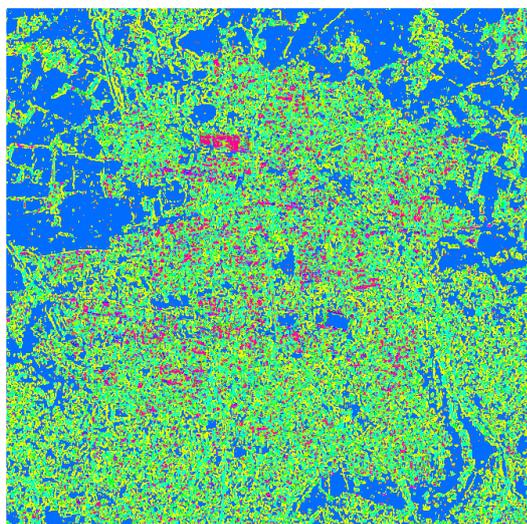
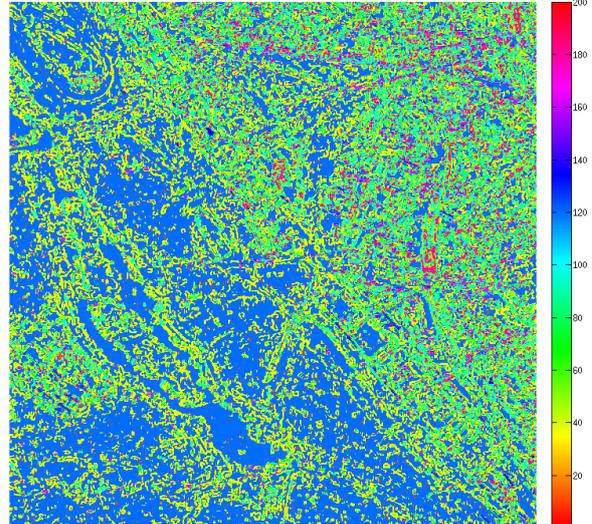
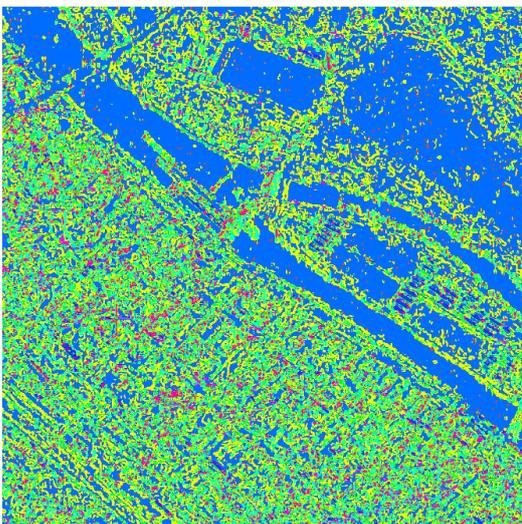


Image de tests, Lambesc (à gauche) et Toulouse (à droite) : tailles : 2048×2048 et 1872×1872 resp. Images TerraSAR-X, résolution : $1m \times 1m$.



Images de Paris : tailles : 2001×2001 et 1800×1800 . Images TerraSAR-X en mode spotlight $1m \times 1m$ du DLR (projets : MTH0232, LAN 2708 et LAN1746).



Bibliographie

- ACHIM, A., TSAKALIDES, P. et BEZERIANOS, A. (2003). SAR image denoising via bayesian wavelet shrinkage based on heavy-tailed modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 41(8):1773–1784. [89](#)
- AHARON, M., ELAD, M. et BRUCKSTEIN, A. (2006). The K-SVD : An algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54(11):4311–4322. [13](#), [38](#), [55](#)
- ALEXANDER, S. K., VRSCAY, E. R. et TSURUMI, S. (2008). A simple, general model for the affine self-similarity of images. *In Image Analysis and Recognition*, pages 192–203. Springer. [61](#)
- APPLEGATE, D. L., BIXBY, R. E., CHVATAL, V. et COOK, W. J. (2011). *The traveling salesman problem : a computational study*. Princeton university press. [44](#), [93](#)
- ARTHUR, D. et VASSILVITSKII, S. (2007). K-Means++ : The advantages of careful seeding. *In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics. [69](#)
- ATTO, A. M. et MERCIER, G. (2015). High order structural image decomposition by using non-linear and non-convex regularizing objectives. *Computer Vision and Image Understanding*, pages http–dx. [33](#)
- AUBERT, G. et AUJOL, J.-F. (2008). A variational approach to removing multiplicative noise. *SIAM Journal on Applied Mathematics*, 68(4):925–946. [87](#), [88](#)
- AWATE, S. P. et WHITAKER, R. T. (2005). Higher-order image statistics for unsupervised, information-theoretic, adaptive, image filtering. *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 44–51. IEEE. [26](#)
- BEN AYED, I., MITICHE, A. et BELHADJ, Z. (2005). Multiregion level-set partitioning of synthetic aperture radar images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):793–800. [103](#)

- BENOÎT, L., MAIRAL, J., BACH, F. et PONCE, J. (2011). Sparse image representation with epitomes. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2913–2920. [57](#), [81](#), [94](#)
- BIUCAS-DIAS, J. et FIGUEIREDO, M. A. (2010). Multiplicative noise removal using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*. [87](#), [98](#)
- BOYKOV, Y., VEKSLER, O. et ZABIH, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239. [110](#)
- BREDIES, K., KUNISCH, K. et POCK, T. (2010). Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526. [34](#)
- BUADES, A., COLL, B. et MOREL, J.-M. (2005). A non-local algorithm for image denoising. *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 60–65 vol. 2. [13](#), [25](#), [26](#)
- BUADES, A., COLL, B. et MOREL, J.-M. (2011). Non-Local Means Denoising. *Image Processing On Line*, 1. [28](#)
- CHAINAIS, P. (2012). Towards dictionary learning from images with non gaussian noise. *In IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE. [73](#), [89](#)
- CHEN, S. S., DONOHO, D. L. et SAUNDERS, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61. [36](#)
- CHEN, Y., RANFTL, R. et POCK, T. (2014). Insights into analysis operator learning : From patch-based sparse models to higher order MRFs. *IEEE Transactions on Image Processing*, 23(3):1060–1072. [34](#), [37](#)
- CORLESS, R., GONNET, G., HARE, D., JEFFREY, D. et KNUTH, D. (1996). On the Lambert W function. *Advances in Computational Mathematics*, 5:329–359. [98](#)
- DABOV, K., FOI, A., KATKOVNIK, V. et EGIAZARIAN, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095. [27](#), [48](#)
- D’ALVISE, R., MOLONEY, C. et RAMPONI, G. (1999). Statistical methods for the evaluation of the coefficient of variation in SAR images. *In IEEE International Geoscience and Remote Sensing Symposium, IGARSS*, volume 2, pages 1253–1255 vol.2. [94](#)
- DANG, H.-P. et CHAINAIS, P. (2015). Une approche bayésienne non paramétrique pour l’apprentissage d’un dictionnaire de taille adaptative. *In GRETSI*. [37](#)

- DELEDALLE, C.-A., DENIS, L. et TUPIN, F. (2009). Iterative weighted maximum likelihood denoising with probabilistic patch-based weights. *IEEE Transactions on Image Processing*, 18(12):2661–2672. [89](#)
- DELEDALLE, C.-A., DENIS, L. et TUPIN, F. (2012a). How to compare noisy patches? patch similarity beyond gaussian noise. *International journal of computer vision*, 99(1): 86–102. [90](#)
- DELEDALLE, C.-A., DENIS, L. et TUPIN, F. (2013). Template matching with noisy patches : A contrast-invariant GLR test. *In Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE. [14](#), [73](#), [76](#), [78](#), [79](#), [113](#)
- DELEDALLE, C.-A., DENIS, L., TUPIN, F., REIGBER, A. et JÄGER, M. (2015). NL-SAR : A unified non-local framework for resolution-preserving (Pol)(In) SAR denoising. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4):2021–2038. [89](#), [99](#), [100](#), [101](#), [117](#)
- DELEDALLE, C.-A., DUVAL, V. et SALMON, J. (2012b). Non-local methods with shape-adaptive patches (NLM-SAP). *Journal of Mathematical Imaging and Vision*, 43(2):103–120. [28](#)
- DELEDALLE, C.-A., SALMON, J. et DALALYAN, A. S. (2011). Image denoising with patch based PCA : local versus global. *In BMVC*, pages 1–10. [28](#), [29](#)
- DELLINGER, F., DELON, J., GOUSSEAU, Y., MICHEL, J. et TUPIN, F. (2014). Change detection for high resolution satellite images, based on SIFT descriptors and an a contrario approach. *In IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1281–1284. [117](#)
- DELON, J. et DESOLNEUX, A. (2013). A patch-based approach for removing impulse or mixed gaussian-impulse noise. *SIAM Journal on Imaging Sciences*, 6(2):1140–1174. [22](#)
- DEMPSTER, A. P., LAIRD, N. M. et RUBIN, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal statistical society, series B*, 39(1):1–38. [41](#), [65](#), [100](#)
- DENIS, L., TUPIN, F., DARBON, J. et SIGELLE, M. (2009). SAR image regularization with fast approximate discrete minimization. *IEEE Transactions on Image Processing*, 18(7):1588–1600. [88](#)
- DENIS, L., TUPIN, F. et RONDEAU, X. (2010). Exact discrete minimization for TV+ ℓ_0 image decomposition models. *In 17th IEEE International Conference on Image Processing (ICIP)*, pages 2525–2528. [88](#)
- DONOHOO, D. L. et JOHNSTONE, J. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455. [25](#)

- EFROS, A., LEUNG, T. K. *et al.* (1999). Texture synthesis by non-parametric sampling. *In The Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1033–1038. IEEE. 26
- ELAD, M. *et* AHARON, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745. 39
- FOUCHER, S. (2008). SAR image filtering via learned dictionaries and sparse representations. *In IEEE International Geoscience and Remote Sensing Symposium, IGARSS*, volume 1, pages I–229. IEEE. 89
- FRERY, A. C., MULLER, H. J., YANASSE, C. C. F. *et* SANT’ANNA, S. J. S. (1997). A model for extremely heterogeneous clutter. *IEEE Transactions on Geoscience and Remote Sensing*, 35(3):648–659. 105
- GALERNE, B. *et* GOUSSEAU, Y. (2012). The transparent dead leaves model. *Advances in Applied Probability*, 44(1):1–20. 75, 76
- GALLAND, F., RÉFRÉGIER, P. *et* GERMAIN, O. (2004). Synthetic aperture radar oil spill segmentation by stochastic complexity minimization. *IEEE Geoscience and Remote Sensing Letters*, 1(4):295–299. 103
- GOODMAN, J. W. (1976). Some fundamental properties of speckle. *Journal of optical society of America*. 19
- GUILLEMOT, T., ALMANSA, A. *et* BOUBEKEUR, T. (2014). Covariance trees for 2D and 3D processing. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 556–563. 31, 48
- GUO, H., ODEGARD, J., LANG, M., GOPINATH, R., SELESNICK, I. *et* BURRUS, C. (1994). Wavelet based speckle reduction with application to SAR based ATD/R. *In IEEE International Conference on Image Processing*, volume 1, pages 75–79 vol.1. 89
- HASHIMOTO, W. *et* KURATA, K. (2000). Properties of basis functions generated by shift invariant sparse representations of natural images. *Biological Cybernetics*, 83(2):111–118. 54
- HESTENES, M. R. *et* STIEFEL, E. (1952). *Methods of conjugate gradients for solving linear systems*, volume 49. NBS. 68
- ISHIKAWA, H. (2003). Exact optimization for Markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336. 33

- ISMAIL, A. S., GAO, X. et DENG, C. (2014). SAR image classification based on texture feature fusion. *In IEEE China Summit International Conference on Signal and Information Processing (ChinaSIP)*, pages 153–156. 106
- JOJIC, N., FREY, B. et KANNAN, A. (2003). Epitomic analysis of appearance and shape. *In Ninth IEEE International Conference on Computer Vision Proceedings (ICCV)*, pages 34–41. IEEE. 57, 58, 94, 96
- JOST, P., VANDERGHEYNST, P., LESAGE, S. et GRIBONVAL, R. (2006). MoTIF : an efficient algorithm for learning translation invariant dictionaries. *In IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*. 56, 57, 74, 81
- KAVUKCUOGLU, K., SERMANET, P., BOUREAU, Y.-L., GREGOR, K., MATHIEU, M. et LECUN, Y. (2010). Learning convolutional feature hierarchies for visual recognition. *In Advances in neural information processing systems*, pages 1090–1098. 57, 79, 81
- KAYABOL, K. et ZERUBIA, J. (2013). Unsupervised amplitude and texture classification of SAR images with multinomial latent model. *IEEE Transactions on Image Processing*, 22(2):561–572. 105, 106
- KRISHNAN, T. et MCLACHLAN, G. (1997). *The EM algorithm and extensions*, volume 1. 42
- KUAN, D. T., SAWCHUK, A. A., STRAND, T. C. et CHAVEL, P. (1987). Adaptive restoration of images with speckle. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):373–383. 87
- LARDEUX, C., FRISON, P.-L., TISON, C., SOUYRIS, J.-C., STOLL, B., FRUNEAU, B. et RUDANT, J.-P. (2009). Support vector machine for multifrequency SAR polarimetric data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 47(12): 4143–4152. 105
- LE PENNEC, E. et MALLAT, S. (2005). Sparse geometric image representations with bandelets. *IEEE Transactions on Image Processing*, 14(4):423–438. 25
- LEBRUN, M., BUADES, A. et MOREL, J.-M. (2013). A nonlocal Bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688. 26, 27, 29, 45, 46, 48, 62
- LEBRUN, M. et LECLAIRE, A. (2012). An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm. *Image Processing On Line*, 2:96–133. 39
- LEE, J.-S. (1981). Refined filtering of image noise using local statistics. *Computer graphics and image processing*, 15(4):380–389. 87

- LOBRY, S., DENIS, L. et TUPIN, F. (2015). Sparse + smooth decomposition models for multi-temporal SAR images. *In 8th International Workshop on the Analysis of Multitemporal Remote Sensing Images (Multi-Temp)*, pages 1–4. 88
- LOUCHET, C. et MOISAN, L. (2011). Total variation as a local filter. *SIAM Journal on Imaging Sciences*, 4(2):651–694. 28
- LOUCHET, C. et MOISAN, L. (2014). Total Variation denoising using iterated conditional expectation. *In Proceedings of the 22nd European Signal Processing Conference (EUSIPCO), 2014*, pages 1592–1596. IEEE. 34
- MAILHÉ, B., LESAGE, S., GRIBONVAL, R., BIMBOT, F., VANDERGHEYNST, P. *et al.* (2008). Shift-invariant dictionary learning for sparse representations : extending K-SVD. *In 16th European Signal Processing Conference (EUSIPCO'08)*. 55
- MAKITALO, M., FOI, A., FEVRALEV, D. et LUKIN, V. (2010). Denoising of single-look SAR images based on variance stabilization and non-local filters. *In Mathematical Methods in Electromagnetic Theory (MMET), 2010 International Conference on*, pages 1–4. IEEE. 89, 99, 100, 101
- MALLAT, S. (2008). *A Wavelet Tour of Signal Processing, Third Edition : The Sparse Way*. Academic Press, 3rd édition. 25
- MALLAT, S. G. et ZHANG, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415. 36
- MEJAIL, M. E., JACOBO-BERLLES, J. C., FRERY, A. C. et BUSTOS, O. H. (2003). Classification of SAR images using a general and tractable multiplicative model. *International Journal of Remote Sensing*, 24(18):3565–3582. 105
- MORUP, M., SCHMIDT, M. N. et HANSEN, L. K. (2008). Shift invariant sparse coding of image and music data. *Submitted to Journal of Machine Learning Research*. 57
- NICOLAS, J.-M. (2006). Application de la transformée de Mellin : étude des lois statistiques de l'imagerie cohérente. Rapport technique, Télécom ParisTech. 84
- OLSHAUSEN, B. A. *et al.* (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609. 37, 38, 54, 57
- PARK, H.-S. et JUN, C.-H. (2009). A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, 36(2):3336–3341. 89
- PARRILLI, S., PODERICO, M., ANGELINO, C. et VERDOLIVA, L. (2012). A non-local SAR image denoising algorithm based on LLMMSE wavelet shrinkage. *IEEE Transactions on Geoscience and Remote Sensing*, 50(2):606–616. 89

- PATI, Y., REZAIIFAR, R. et KRISHNAPRASAD, P. (1993). Orthogonal matching pursuit : recursive function approximation with applications to wavelet decomposition. *In Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pages 40–44 vol.1. [36](#), [38](#)
- PERONA, P. et MALIK, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639. [24](#)
- PORTILLA, J., STRELA, V., WAINWRIGHT, M. J. et SIMONCELLI, E. P. (2003). Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351. [25](#), [26](#)
- POSTEC, S. (2012). *Quelques remarques en débruitage des images liées à des propriétés de similarité, de régularité et de parcimonie*. Thèse de doctorat, Université de Bretagne-Sud. [61](#)
- PRENDES, J., CHABERT, M., PASCAL, F., GIROS, A. et TOURNERET, J.-Y. (2015). A new multivariate statistical model for change detection in images acquired by homogeneous and heterogeneous sensors. *IEEE Transactions on Image Processing*, 24(3):799–812. [117](#)
- ROTH, S. et BLACK, M. (2009). Fields of experts. *International Journal of Computer Vision*, 82(2):205–229. [34](#), [36](#)
- RUBINSTEIN, R., PELEG, T. et ELAD, M. (2013). Analysis K-SVD : A dictionary-learning algorithm for the analysis sparse model. *IEEE Transactions on Signal Processing*, 61(3):661–677. [37](#)
- RUDIN, L. I., OSHER, S. et FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D : Nonlinear Phenomena*, 60(1):259–268. [24](#), [33](#)
- SHI, J. et OSHER, S. (2008). A nonlinear inverse scale space method for a convex multiplicative noise model. *SIAM Journal on Imaging Sciences*, 1(3):294–321. [87](#)
- STARCK, J.-L., CANDÈS, E. J. et DONOHO, D. L. (2002). The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684. [25](#)
- STEIDL, G. et TEUBER, T. (2010). Removing multiplicative noise by Douglas-Rachford splitting methods. *Journal of Mathematical Imaging and Vision*, 36(2):168–184. [88](#)
- SULAM, J., OPHIR, B. et ELAD, M. (2014). Image denoising through multi-scale learnt dictionaries. *In IEEE International Conference on Image Processing (ICIP)*, pages 808–812. [40](#)

- SUTOUR, C., DELEDALLE, C.-A. et AUJOL, J.-F. (2014). Adaptive regularization of the NL-Means : Application to image and video denoising. *IEEE Transactions on Image Processing*, 23(8):3506–3521. [28](#)
- TABTI, S., DELEDALLE, C. A., DENIS, L. et TUPIN, F. (2014a). Building invariance properties for dictionaries of SAR image patches. *In IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4918–4921. [15](#), [83](#), [90](#), [113](#)
- TABTI, S., DELEDALLE, C. A., DENIS, L. et TUPIN, F. (2014b). Modeling the distribution of patches with shift-invariance : Application to SAR image restoration. *In IEEE International Conference on Image Processing (ICIP)*, pages 96–100. [15](#), [58](#), [83](#), [96](#), [117](#)
- TABTI, S., DELEDALLE, C.-A., DENIS, L. et TUPIN, F. (2015). Patch-based SAR image classification : The potential of modeling the statistical distribution of patches with Gaussian mixtures. *In IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 2374–2377. [15](#), [83](#), [103](#), [117](#)
- TARTAVEL, G. (2015). *Modèles variationnels pour les textures : applications à la synthèse et à la restauration*. Thèse de doctorat, Télécom ParisTech. [57](#)
- THIAGARAJAN, J., RAMAMURTHY, K. et SPANIAS, A. (2008). Shift-invariant sparse representation of images using learned dictionaries. *In IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pages 145–150. [55](#), [56](#)
- TIPPING, M. E. et BISHOP, C. M. (1999). Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482. [45](#)
- TISON, C., NICOLAS, J.-M., TUPIN, F. et MAÎTRE, H. (2004). A new statistical model for Markovian classification of urban areas in high-resolution SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 42(10):2046–2057. [105](#), [106](#)
- TUPIN, F., INGLADA, J. et NICOLAS, J.-M. (2014). *Remote Sensing Imagery*. John Wiley & Sons. [19](#)
- VOISIN, A., KRYLOV, V. A., MOSER, G., SERPICO, S. B. et ZERUBIA, J. (2014). Supervised classification of multisensor and multiresolution remote sensing images with a hierarchical copula-based approach. *IEEE Transactions on Geoscience and remote sensing*, 52(6):3346–3358. [105](#)
- VOISIN, A., MOSER, G., KRYLOV, V., SERPICO, S. et ZERUBIA, J. (2010). Classification of very high resolution SAR images of urban areas by dictionary-based mixture models, copulas and Markov random fields using textural features. *In LORENZO BRUZZONE*, éditeur : *SPIE Remote Sensing*, volume 7830, Toulouse, France. SPIE. [105](#), [106](#)

- WANG, Y.-Q. (2013). The Implementation of SURE Guided Piecewise Linear Image Denoising. *Image Processing On Line*, 3:43–67. [44](#)
- WEISS, Y. et FREEMAN, W. (2007). What makes a good model of natural images? *In IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE. [35](#), [40](#)
- XIE, H., PIERCE, L. et ULABY, F. (2002a). SAR speckle reduction using wavelet denoising and Markov random field modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 40(10):2196–2212. [85](#)
- XIE, H., PIERCE, L. et ULABY, F. (2002b). Statistical properties of logarithmically transformed speckle. *IEEE Trans. on Geoscience and Remote Sensing*, 40(3):721–727. [85](#)
- YANG, W., ZOU, T., DAI, D. et SHUAI, Y. (2009). Supervised land-cover classification of TerraSAR-X imagery over urban areas using extremely randomized clustering forests. *In Urban Remote Sensing Event, 2009 Joint*, pages 1–6. [105](#)
- YAROSLAVSKY, L.-P. (1985). Digital picture processing. An introduction. *Springer Series in Information Sciences, Springer-Verlag, Berlin, Volume 9*. [24](#)
- YONG, Y., HONG, S. et CHU, H. (2006). Supervised SAR image MPM segmentation based on region-based hierarchical model. *Geoscience and Remote Sensing Letters, IEEE*, 3(4):517–521. [106](#)
- YU, G., SAPIRO, G. et MALLAT, S. (2012). Solving inverse problems with piecewise linear estimators : From Gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21:2481–2499. [25](#), [29](#), [40](#), [45](#), [62](#)
- ZHANG, L., DONG, W., ZHANG, D. et SHI, G. (2010). Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, 43(4): 1531–1549. [28](#), [29](#)
- ZORAN, D. et WEISS, Y. (2011). From learning models of natural image patches to whole image restoration. *In IEEE International Conference on Computer Vision (ICCV)*, pages 479–486. [15](#), [25](#), [41](#), [44](#), [45](#), [48](#), [59](#), [62](#), [65](#), [69](#), [83](#), [96](#), [99](#), [100](#), [102](#), [104](#), [107](#), [112](#), [114](#), [117](#)

Modélisation des images par patches pour leur restauration et leur interprétation Applications à l'imagerie SAR

Sonia TABTI

RESUME : Le traitement des images satellitaire est un enjeu important dans notre société où l'on a besoin de surveiller la planète pour évaluer la déforestation ou encore réaliser le bilan d'une catastrophe naturelle. Les images radar sont utiles pour cela car elles sont obtenues par un système actif et donc indépendamment de l'éclairement solaire et du couvert nuageux. Ce système d'acquisition entraîne le phénomène de speckle modélisé par un bruit multiplicatif. C'est donc un véritable défi d'adapter à ces données les outils de traitements classiques souvent développés pour du bruit additif.

Nous nous sommes intéressés dans cette thèse à des approches utilisant une décomposition de l'image en patches (petites imageries d'une dizaine de pixels de côté). Le premier type d'approche est basé sur l'apprentissage de dictionnaires. Un dictionnaire est un ensemble de patches, appelés atomes, représentant les structures de base d'une image, par exemple des contours et des textures. Chaque patch d'une image peut être représenté par une combinaison linéaire de peu d'atomes du dictionnaire et une fois tous les patches recombinaés, ils constituent une approximation de l'image d'intérêt. Nous nous sommes également intéressés à un second type d'approche modélisant la distribution des patches des images par des modèles de mélange de gaussiennes ou Gaussian Mixture Models (GMM) en anglais.

La problématique au coeur de cette thèse porte sur la modélisation de la distribution des patches avec des propriétés d'invariance. Pour cela, nous avons proposé plusieurs manières d'introduire l'invariance par translation et l'invariance aux transformations radiométriques affines dans l'apprentissage de dictionnaires et dans l'apprentissage de GMM. Nous avons également appliqué les méthodes proposées au débruitage d'images naturelles et à la réduction de speckle dans les images radar. La méthode de restauration proposée basée sur les GMM présente des performances du même ordre que les méthodes état de l'art. Enfin, une procédure de classification supervisée d'images radar basée sur les GMM a été élaborée et montre leur potentiel à bien représenter la distribution des patches des images radar.

MOTS-CLEFS : images radar, réduction de speckle, débruitage, classification, approches par patches, apprentissage de dictionnaires, modèles de mélange de gaussiennes, propriétés d'invariance.

ABSTRACT : Satellite image processing is an important issue in our society because of the need to monitor the planet, for example, to evaluate the effects of a natural disaster. SAR (Synthetic Aperture Radar) images are useful for this purpose because they can be acquired day and night and through clouds since they are the result of a backscattered echo from earth's surface. This acquisition system produces the speckle phenomenon modeled by a multiplicative noise which makes it difficult to adapt traditional image processing methods often developed to reduce additive noise.

In this thesis, we focused on approaches based on patches which are small windows extracted from images. The first type of approach used is dictionary-based. A dictionary is a set of patches named atoms representing the basic textures and structures observed in images such as edges or targets. Each patch of an image can be approximated by a linear combination of few atoms. Once the approximated patches are combined, we obtain an approximation of the image. The second type of approach used is based on Gaussian Mixture Models (GMM) that model the distribution of patches efficiently.

The main problem addressed in this thesis is how to model the distribution of the patches with invariance properties? To this end, we propose several methods in order to learn dictionaries and GMMs with shift-invariance and affine radiometric transform invariance properties. We applied these methods in denoising and despeckling tasks and the GMM based method provided similar results as state of the art methods. Finally, we proposed a supervised SAR image classification algorithm based on GMM which shows their potential in modeling the distribution of patches extracted from SAR images.

KEY-WORDS : radar images, speckle reduction, denoising, classification, patch-based approaches, dictionary learning, gaussian mixture models, invariance properties.

