

# Partition d'images par minimisation de la complexité stochastique et grille active : application à la segmentation d'images de radar à ouverture synthétique

## THÈSE

présentée et soutenue publiquement le 1er décembre 2004

pour l'obtention du

Doctorat de l'Université d'Aix-Marseille III  
(spécialité Traitement des Images)

par

Frédéric GALLAND

### Composition du jury

*Rapporteurs :* Jean-Marc Chassery (LIS, Grenoble)  
Mario A. T. Figueiredo (IT, Lisbonne)

*Examinateurs :* Patrick Bouthemy (INRIA, Rennes)  
François Le Chevalier (Thales TSA, Élancourt)  
Henri Maître (ENST, Paris)  
Philippe Réfrégier (Directeur de thèse)  
Nicolas Bertaux (Co-encadrant)



Les radars à ouverture synthétique (SAR), opérationnels de jour comme de nuit et par tout type de temps, constituent un système d'imagerie satellitaire très intéressant pour l'observation de la terre. Cependant, la quantité de données fournies par ces satellites est d'une telle ampleur que leur interprétation par un opérateur humain n'est pas envisageable. A ce titre, le développement de techniques de segmentation automatique d'images, qui consiste à partitionner une image en régions d'intérêt, constitue un enjeu majeur. Malheureusement, du fait de l'éclairage cohérent utilisé, les images radars obtenues sont fortement dégradées par un phénomène de speckle qui limite considérablement leur interprétation automatique.

Au cours de cette thèse, nous présentons une technique de partition d'image en régions homogènes, reposant sur la minimisation de la complexité stochastique de l'image à l'aide d'une grille active polygonale, c'est-à-dire un ensemble de noeuds reliés par des segments afin de définir les différentes régions de l'image. L'algorithme obtenu est :

- adapté à la segmentation d'images SAR grâce à une modélisation probabiliste du phénomène de speckle,
- rapide (moins d'une seconde pour une image de  $256 \times 256$  pixels sur un PC cadencé à 2.8 GHz),
- et qui repose sur l'optimisation d'un critère sans paramètre à régler de la part de l'utilisateur.

Cette approche permet ainsi l'estimation automatique du nombre de régions présentes dans l'image, de leur topologie ainsi que de la position et du nombre de noeuds définissant les contours.

Après avoir détaillé quelques applications en imagerie SAR, notamment la segmentation de parcelles agricoles ou la détection de nappes de pétrole, nous proposons quelques généralisations de notre technique, tout d'abord à d'autres bruits que le speckle, puis à l'imagerie vectorielle et enfin à la segmentation d'image à l'aide de contours actifs multi-régions.

Finalement, nous présentons une méthode originale qui permet la partition d'images sans nécessiter la connaissance *a priori* de la famille de loi de probabilité des niveaux de gris des pixels dans chacune des régions de l'image.

**Mots clés :** radar à ouverture synthétique (SAR), segmentation, principe de minimisation de la complexité stochastique (MDL), grille et contour actif statistiques.

Synthetic Aperture Radar (SAR) instruments, which can operate at any time of day and in any weather condition, are widely used for remote sensing and earth observatory. However, the amount of data delivered by satellites is so huge that their analysis cannot be performed by human operators. The development of automatic image segmentation techniques - which consists on partitioning the image into regions of interest - thus appears as a crucial issue. Unfortunately, due to coherent imaging, such radar images are corrupted by a speckle phenomenon which complicates their analysis in an automatic way.

In this work, we present a technique for partitioning an image into homogeneous regions. It is based on the minimisation of the stochastic complexity of the image thanks to a polygonal active grid, *i.e.* a set of nodes linked by segments to define the different regions of the image. The obtained algorithm is :

- adapted to SAR image segmentation thanks to a probabilistic modelisation of the speckle phenomenon,
- fast (less than one second on a  $256 \times 256$  pixel image with a 2.8 GHz PC),
- and based on the optimization of a criterion without parameter to be tuned by the user.

This approach allows one to automatically estimate the number of regions in the image, their topology and the position and the number of nodes used to define the contour.

After having detailed some applications to SAR imagery, notably the segmentation of agricultural areas or the detection of oil spills, we propose some generalizations of this technique, first to other noise phenomenon than the speckle one, then to vectorial images and finally to image segmentation with multi-region active contours.

We will then present an original approach for image partitioning without *a priori* knowledge of the probability law of the pixel grey levels in the regions of the image.

**Key words :** Synthetic Aperture Radar, segmentation, Minimum Description Length principle, statistical active grids and contours.

# Remerciements

Je tiens en premier lieu à remercier Jean-Marc Chassery et Mario A. T. Figueiredo pour le temps qu'ils ont consacré à rapporter ce manuscrit et pour l'intérêt qu'ils ont manifesté à l'égard de ces travaux. Mes remerciements vont également aux autres membres du jury, Patrick Bouthemy et François Le Chevalier et tout particulièrement à Henri Maître qui m'a fait l'honneur de présider ce jury.

Je tiens aussi tout particulièrement à remercier Philippe Réfrégier qui a dirigé cette thèse et qui a su me communiquer tout au long de ces trois années son enthousiasme et sa motivation. Il a su allier avec justesse les différents ingrédients pour que cette thèse soit passionnante, tout en gardant à l'esprit la question de mon insertion future dans le monde de la recherche. Je retiendrai également sa très grande disponibilité ainsi que son soutien moral constant.

Merci également à Nicolas Bertaux, pour ses conseils avisés et son aide précieuse. Beaucoup de problèmes informatiques que j'ai rencontrés ont été résolus grâce à lui.

Que François Goudail trouve également ici mes plus sincères remerciements. Ses conseils, notamment lors de l'écriture d'articles ou de ce manuscrit, se sont toujours avérés très pertinents.

Je n'oublierai pas non plus Muriel Roche, avec laquelle j'ai eu la chance de co-encadrer mon premier stagiaire. Son aide, ses conseils et sa gentillesse m'ont été du plus grand secours.

Je voudrais encore souligner la très grande disponibilité dont ont fait preuve tous les membres de l'équipe PhyTI et je leur exprime une nouvelle fois toute ma gratitude.

Mes remerciements vont également à mes compagnons thésards. Merci à Pascal Martin et Nicolas Roux pour les nombreuses discussions que nous avons pu avoir, mais également pour la bonne humeur qu'ils ont apportée. Le lecteur n'oubliera pas non plus que le manuscrit qu'il tient a été imprimé grâce à Pascal qui a eu l'amabilité de se charger de toutes les tracasseries administratives que je ne pouvais pas gérer depuis Paris : je lui en suis infiniment reconnaissant.

Je tiens également à remercier Guillaume Delyon et Jérôme Morio, qui vont continuer à travailler avec mes programmes. Je leur souhaite à tous deux de le faire avec le même plaisir que moi.

Je ne terminerai pas sans remercier tous les autres thésards de l'institut Fresnel, qui ont tous contribué à rendre cette période très agréable.

Et puis bien sûr, merci à Cécile, mais ça, c'est une autre histoire...



# Table des matières

## Remerciements

## Introduction

### Chapitre 1

#### Etat de l'art

1.1	Imagerie radar . . . . .	6
1.1.1	Historique et applications . . . . .	6
1.1.2	Modélisation probabiliste des images . . . . .	7
1.2	Segmentation d'images SAR . . . . .	9
1.2.1	Etat de l'art en segmentation d'images SAR . . . . .	9
1.2.2	Segmentation par contours actifs . . . . .	11
1.2.3	Partition en régions homogènes et segmentation . . . . .	14
1.3	Complexité stochastique . . . . .	15
1.3.1	Principe de la minimisation de la complexité stochastique . . . . .	16
1.3.2	Segmentation et minimisation de la complexité stochastique . . . . .	19
1.3.3	Analyse du principe de la minimisation de la complexité stochastique au vu de ces quelques travaux . . . . .	20
1.4	Conclusion . . . . .	20

### Chapitre 2

#### Partition en régions homogènes par minimisation de la complexité stochastique

2.1	Complexité stochastique . . . . .	24
2.1.1	Modèle d'image . . . . .	24
2.1.2	Détermination de la complexité stochastique . . . . .	25
2.2	Algorithme d'optimisation proposé . . . . .	30
2.2.1	Fusion de régions . . . . .	30
2.2.2	Déplacement des nœuds . . . . .	31
2.2.3	Suppression des nœuds . . . . .	31

2.2.4	Procédure globale d'optimisation . . . . .	32
2.2.5	Algorithme rapide . . . . .	35
2.3	Exemples de partition en régions homogènes . . . . .	36
2.3.1	Images synthétiques . . . . .	36
2.3.2	Sélection automatique de la grille initiale . . . . .	37
2.4	Conclusion . . . . .	40

### Chapitre 3

#### Applications à l'imagerie SAR

3.1	Segmentation de parcelles agricoles . . . . .	42
3.2	Estimation de l'ordre du speckle . . . . .	47
3.2.1	Influence du choix de l'ordre . . . . .	47
3.2.2	Partition en régions homogènes sans connaissance <i>a priori</i> de l'ordre du speckle . . . . .	48
3.3	Segmentation de nappes de pétrole . . . . .	53
3.3.1	Etape 1 : partition en régions homogènes . . . . .	53
3.3.2	Etape 2 : classification par seuillage . . . . .	54
3.3.3	Amélioration de la robustesse et discussion . . . . .	58
3.4	Conclusion . . . . .	60

### Chapitre 4

#### Généralisations du modèle

4.1	Généralisation à d'autres lois de probabilités . . . . .	64
4.1.1	Détermination de la complexité stochastique . . . . .	64
4.1.2	Cas des images vidéo . . . . .	68
4.1.3	Généralisation à des modèles d'images plus complexes . . . . .	70
4.1.4	Utilisation de la partition en régions homogènes comme pré-segmentation dans des images vidéo . . . . .	73
4.2	Images vectorielles . . . . .	77
4.2.1	Détermination de la complexité stochastique . . . . .	77
4.2.2	Images de type multi-dates . . . . .	78
4.2.3	Images dont les composantes possèdent des ddp de familles différentes .	84
4.2.4	Images SAR polarimétriques . . . . .	86
4.2.5	Conclusion . . . . .	88
4.3	Segmentation par contours actifs multi-régions . . . . .	91
4.3.1	Complexité stochastique . . . . .	91
4.3.2	Algorithme de segmentation par contours actifs multi-régions . . . . .	92
4.3.3	Résultats de segmentation . . . . .	93

---

4.3.4	Introduction de connaissances <i>a priori</i> sur les paramètres de la cible . . . . .	100
4.3.5	Segmentation multi-lois . . . . .	101
4.4	Conclusion . . . . .	104

## Chapitre 5

### Partition d'images à $Q$ niveaux

5.1	Détermination de la complexité stochastique . . . . .	108
5.2	Algorithme rapide . . . . .	110
5.3	Résultats expérimentaux . . . . .	110
5.3.1	Images pré-classifiées réelles . . . . .	111
5.3.2	Quantification des niveaux de gris . . . . .	115
5.3.3	Images couleurs et polarimétriques . . . . .	117
5.3.4	Contrainte de la topologie par contours actifs . . . . .	118
5.4	Conclusion . . . . .	119

## Conclusion et perspectives

## Annexe A

### Calculs et algorithmes annexes

A.1	Détermination du terme de codage de la grille $\Delta_G(w)$ . . . . .	128
A.2	Détermination de la log-vraisemblance généralisée d'une loi gamma d'ordre $L$ .	131
A.3	Algorithme de minimisation de la complexité stochastique . . . . .	132
A.3.1	Fusion de régions . . . . .	132
A.3.2	Déplacement des nœuds . . . . .	132
A.3.3	Suppression des nœuds . . . . .	133
A.3.4	Temps de calcul . . . . .	134
A.4	Définition non ambiguë de l'intérieur des régions par translation de la grille .	137
A.5	Algorithme rapide : présentation, généralisation et mise en œuvre . . . . .	140
A.5.1	Rappels . . . . .	140
A.5.2	Généralisation aux régions non simplement connexes . . . . .	140
A.5.3	Mise en œuvre . . . . .	142
A.5.4	Gestion de la topologie . . . . .	143
A.6	Cas où la moyenne n'est plus constante pas morceaux . . . . .	145
A.6.1	ddp gaussienne et espérance mathématique distribuée selon une fonction linéaire . . . . .	145
A.6.2	ddp gaussienne et espérance mathématique distribuée selon une fonction quadratique . . . . .	146

A.6.3 ddp gamma et espérance mathématique distribuée selon une fonction linéaire . . . . .	147
A.7 Contour actif et transmission des paramètres de la cible . . . . .	149
A.8 Estimation de la ddp d'une image à $Q$ niveaux . . . . .	150

**Annexe B**

**Acronymes**

**Annexe C**

**Liste des travaux**

**Bibliographie**

**155**

# Introduction

L'observation de la terre à l'aide de systèmes d'imagerie satellitaires ou aéroportés intervient dans des domaines aussi divers que la cartographie, la météorologie, l'océanographie ou même l'archéologie. Les images obtenues sont délivrées par des capteurs embarqués mesurant le rayonnement électromagnétique provenant de la surface étudiée. Ces capteurs peuvent être soit *passifs* soit *actifs*.

Les systèmes d'imagerie optiques passifs, tels que les satellites SPOT ou Landsat, mesurent directement le rayonnement réfléchi par la surface terrestre, ce rayonnement pouvant être visible ou infrarouge. Bien que l'utilisation de tels imageurs soit très intéressante, ils ont cependant quelques inconvénients. Le premier est lié à leur dépendance vis à vis de l'éclairage solaire. Il est ainsi impossible de faire des acquisitions nocturnes, ce qui peut être problématique notamment pour l'observation des régions polaires plongées dans l'obscurité une grande partie de l'année. De plus, en cas de couverture nuageuse ou de brume, il est impossible d'obtenir une image du sol : si cela ne pose pas de réels problèmes pour des observations dans le sud de la France, cela peut être plus gênant lorsqu'elles ont lieu au dessus de l'Ecosse, voire même problématique dans les régions tropicales où la couverture nuageuse quasi-permanente interdit toute observation.

L'utilisation de systèmes actifs radar permet de résoudre en partie ces problèmes. De tels capteurs actifs émettent un rayonnement micro-onde (1-10 GHz) et enregistrent sa rétro-diffusion par la surface terrestre. En disposant de leur propre source d'illumination, ils deviennent indépendants de l'illumination solaire et peuvent donc opérer de jour comme de nuit. De plus, les micro-ondes sont insensibles à la teneur en eau de l'atmosphère : la qualité des images acquises est indépendante des conditions météorologiques lors de l'observation. Soulignons également que si les micro-ondes traversent aisément les nuages, elles peuvent aussi pénétrer partiellement la végétation, permettant de sonder le sol sous un couvert végétal. Parmi les imageurs radar, les radars à ouverture synthétique (ou SAR, acronyme de Synthetic Aperture Radar en langue anglaise), tels que les satellites ERS, RadarSat ou ENVISAT, sont particulièrement intéressants dans la mesure où ils offrent une bonne résolution spatiale.

Le développement de ces différentes techniques d'observations complémentaires (et non concurrentes, dans la mesure où elles ne donnent pas accès aux mêmes informations) permet maintenant d'appréhender notre planète dans son ensemble. Cela peut ainsi permettre d'analyser divers phénomènes à une échelle planétaire, et non plus localement, ce qui peut s'avérer capital par exemple pour l'étude des courants marins, la prévision des évolutions climatiques ou la détection d'événements particuliers. Parmi ces événements particuliers, nous pourrions citer le dégazage d'un pétrolier ou l'apparition d'algues toxiques dans l'océan. Malheureusement, même s'ils pourraient être repérés sur des images satellitaires (radar ou optique) par un opérateur humain, la quantité de données fournies par ces satellites est d'une telle ampleur qu'il est impossible d'analyser toutes ces images. Il est donc impératif de pouvoir traiter ces données de façon automatique.

Avec le développement des outils informatiques, le domaine du traitement des images a connu un essor considérable. En général l'analyse d'une image se fait en plusieurs étapes. On distingue ainsi des traitements de *bas niveau* et de *haut niveau*. Les premiers travaillent directement sur les intensités des pixels, sans faire le lien avec l'entité qu'ils représentent, alors que les seconds manipulent des entités symboliques représentatives du contenu de l'image. La segmentation d'image est un traitement de bas niveau et consiste à réaliser une partition en différentes régions correspondant aux régions d'intérêt de l'image ou possédant certaines propriétés d'homogénéité (intensité moyenne, texture). Cette tâche intervient dans de nombreuses chaînes de traitements pour faciliter et améliorer des traitements ultérieurs, comme le filtrage, la classification ou la reconnaissance de formes.

Cependant, l'imagerie radar utilisant un rayonnement cohérent, les images obtenues ont un aspect fortement bruité. Ainsi, l'image d'une surface de réflectivité constante n'aura pas une intensité constante mais qui fluctuera fortement d'un pixel à l'autre. Ce phénomène, appelé couramment *speckle* (*chatoiement* en langue française), complique énormément l'interprétation des images. Il en résulte qu'un grand nombre de méthodes utilisées pour segmenter des images issues de capteurs optiques passifs ne sont pas adaptées à la segmentation d'images SAR, fortement dégradées par le speckle. Il est donc nécessaire de développer des méthodes spécifiques adaptées à ce type d'imagerie.

La segmentation d'images SAR a ainsi suscité de nombreux travaux et de multiples approches intéressantes ont été proposées, notamment des approches basées sur des détecteurs de bords, des champs de Markov ou des équations aux dérivées partielles. Cependant, ces méthodes contiennent des paramètres empiriques, c'est-à-dire dont la valeur doit être ajustée par l'expérience, ce qui peut s'avérer problématique dans l'optique d'une segmentation automatique des images.

## Objectifs et plan de thèse

Le but que nous nous sommes fixés est d'essayer de définir un module sans paramètre, qui repose sur l'optimisation d'un critère dont le sens est clairement défini en termes de théorie de l'information. La technique obtenue aura alors pour principaux avantages de reposer sur une mesure quantitative associée au résultat de la partition, et offrira une grande reproductibilité. L'optimisation du critère sera effectuée au travers d'un modèle d'image précis dont la topologie pourra être quelconque, mais également contrainte. A notre connaissance, il n'existe pas à ce jour un tel algorithme, même si celui proposé par Zhu et Yuille [100] s'en approche dans le cas des images optiques, sans pour autant avoir atteint ce but dans la mesure où différents paramètres devant être réglés par l'utilisateur subsistent dans le critère optimisé.

En résumé, l'objectif de cette thèse est d'obtenir un algorithme de segmentation :

- adapté aux images radar à ouverture synthétique (SAR),
- qui repose sur l'optimisation d'un critère sans paramètre dont la valeur doit être ajustée par l'utilisateur,
- rapide, c'est-à-dire de l'ordre de quelques secondes sur un ordinateur PC standard pour une image de  $1000 \times 1000$  pixels.

Ce manuscrit est composé de cinq chapitres.

Après cette brève introduction, le chapitre 1 présente les concepts centraux utilisés au cours de cette thèse. Il débute par un bref historique de l'imagerie SAR et surtout sur sa modélisation probabiliste. Ensuite, nous dressons un état de l'art en segmentation d'images, en mettant l'accent sur les méthodes par contours actifs et adaptées à l'imagerie SAR, avant de

---

proposer une définition probabiliste rigoureuse de la notion de partition en régions homogènes, par opposition à la notion de segmentation qui reste un terme général dont la signification peut varier en fonction de l'application visée. Nous terminons ce chapitre par la présentation du principe de minimisation de la complexité stochastique et de ses applications à la segmentation d'images.

Le chapitre 2 présente la méthode de partition en régions homogènes, adaptée aux images SAR modélisées par du speckle pleinement développé, et qui repose sur la minimisation de la complexité stochastique à l'aide d'une grille active polygonale. L'algorithme obtenu, rapide et sans paramètre à régler de la part de l'utilisateur dans le critère à optimiser, est alors utilisé pour partitionner quelques images synthétiques.

Le chapitre 3 traite ensuite de différentes applications de cette méthode sur des images SAR réelles. Dans un premier temps, nous nous sommes intéressés à la segmentation d'images SAR de parcelles agricoles, lorsque l'ordre du speckle présent dans les images est connu. Nous présentons ensuite une méthode d'estimation automatique de l'ordre du speckle lorsqu'il est inconnu. Enfin, nous proposons une technique de segmentation de nappes de pétroles basée sur l'utilisation conjointe de l'algorithme de partition présenté dans le chapitre précédent et d'un algorithme de classification par seuillage, la valeur du seuil étant déterminée automatiquement.

Le chapitre 4 propose une généralisation du modèle de partition du chapitre 2. Cette généralisation peut être effectuée selon deux voies. La première consiste à prendre en compte d'autres modèles d'images : cela nous conduit dans un premier temps à prendre en compte d'autres lois de probabilité que celle du bruit de speckle, puis à étudier le cas des images vectorielles, notamment les images SAR multi-dates et SAR polarimétriques. La seconde voie de généralisation possible est d'utiliser une autre modélisation paramétrique du contour : nous proposons ainsi un algorithme de segmentation par contours actifs multi-régions.

Enfin le chapitre 5 présente un algorithme de partition d'images pré-classifiées sur  $Q$  niveaux. Cette méthode est d'abord appliquée sur des images réelles pré-classifiées. Nous proposons ensuite une méthode systématique permettant la partition d'images issues de divers systèmes d'imagerie, sans connaissance *a priori* de la loi de probabilité associée aux niveaux de gris de l'image.

Nous rappelons alors en conclusion les principaux résultats de cette thèse avant de dégager diverses perspectives de ces travaux.

Par ailleurs, 3 annexes complètent ce manuscrit. L'annexe A détaille différents aspects (calculatoires ou relatifs à la mise en œuvre algorithmique) utilisés dans ce manuscrit mais non nécessaires à sa compréhension en première lecture. Quant aux annexes B et C, elles contiennent respectivement les acronymes utilisés dans le manuscrit et la liste des travaux publiés au cours de cette thèse.



# Chapitre 1

## Etat de l'art

### Sommaire

---

<b>1.1</b>	<b>Imagerie radar</b>	<b>6</b>
1.1.1	Historique et applications	6
1.1.2	Modélisation probabiliste des images	7
<b>1.2</b>	<b>Segmentation d'images SAR</b>	<b>9</b>
1.2.1	Etat de l'art en segmentation d'images SAR	9
1.2.2	Segmentation par contours actifs	11
1.2.3	Partition en régions homogènes et segmentation	14
<b>1.3</b>	<b>Complexité stochastique</b>	<b>15</b>
1.3.1	Principe de la minimisation de la complexité stochastique	16
1.3.2	Segmentation et minimisation de la complexité stochastique	19
1.3.3	Analyse du principe de la minimisation de la complexité stochastique au vu de ces quelques travaux	20
<b>1.4</b>	<b>Conclusion</b>	<b>20</b>

---

Dans ce chapitre, nous allons tout d'abord présenter, un bref historique de l'imagerie radar, avant de rappeler un modèle simple du phénomène de speckle qui intervient dans ce type d'images et complique considérablement leur interprétation. La deuxième partie sera relative à la segmentation des images SAR, et notamment à la segmentation par contours actifs statistiques, puis nous proposerons une définition de la partition en régions homogènes. Enfin, nous présenterons en troisième partie le principe de minimisation de la complexité stochastique et ses applications à la segmentation d'images.

## 1.1 Imagerie radar

### 1.1.1 Historique et applications

En 1887, Heinrich Hertz découvre les ondes électromagnétiques et montre qu'elles possèdent les mêmes propriétés que la lumière. Si le principe du radar est énoncé dès 1911 par Hugo Gernsback, les premières applications n'apparaissent que dans les années 1930 : elles concernent la détection d'avions et d'icebergs. C'est surtout pendant la seconde guerre mondiale que la technique prend un essor remarquable. Le radar y est largement exploité et s'avère un atout décisif pour les pays alliés lors de la bataille d'Angleterre. Après la guerre, le développement de systèmes militaires et civils se poursuit et les premiers radars imageurs aéroportés voient le jour. Ceux-ci sont utilisés en visée latérale pour exploiter la résolution en distance dans la direction normale à la trajectoire du porteur. Cependant, la résolution azimutale (parallèle à la trajectoire) reste très faible. Au début des années 1950, le principe de la synthèse d'ouverture, proposé par Wiley, permet d'améliorer considérablement la résolution azimutale. Les applications aéroportées dans le domaine militaire suivent immédiatement. En Europe, les premières études ont été menées en France en 1960 et une expérimentation en vol a eu lieu en 1964. La télédétection civile spatiale à l'aide de SAR débute alors en 1978 avec le lancement du satellite américain Seasat pour des applications maritimes et continentales. Suivent deux missions où des capteurs SAR sont embarqués sur la navette spatiale américaine : SIR-A et SIR-B. Dans les années 1990, pas moins de cinq SAR sont satellisés (tableau 1.1). Parmi eux, SIR-C est le premier SAR spatial à délivrer une information multi-spectrale (3 longueurs d'onde : 1.5 cm, 5 cm et 23 cm) et polarimétrique. Actuellement, seuls les satellites ERS-2, Radarsat et ENVISAT sont toujours opérationnels.

Capteur	Pays	Lancement
Seasat	Etats-Unis	1978
SIR-A	Etats-Unis	1981
SIR-B	Etats-Unis	1984
Almaz	Russie	1991
ERS-1	Europe	1991
JERS-1	Japon	1992
SIR-C/X-SAR	Etats-Unis / Allemagne / Italie	1994
ERS-2	Europe	1995
Radarsat	Canada	1995
ENVISAT	Europe	2002

TABLE 1.1 – Historique des capteurs SAR spatiaux civils pour l'observation terrestre.

Les champs d'application des capteurs SAR spatiaux sont d'ores et déjà nombreux et variés [23, 69]. Sans prétendre à l'exhaustivité, on peut donner quelques exemples d'applications déjà opérationnelles.

- *Ressources agricoles et forestières.* Les capteurs SAR permettent la surveillance et le suivi de la croissance des cultures [91]. Comme le signal radar rétro-diffusé dépend en particulier de l'âge des plants ou de l'humidité du sol, l'image SAR fournit ici des informations complémentaires aux données optiques. De même, l'utilisation de données SAR pour le suivi des défrichements dans les zones tropicales est très intéressante [71] puisque cette étude est quasi-impossible avec des capteurs optiques à cause de la couverture nuageuse.
- *Restitution du relief.* Dans ce domaine, l'imagerie radar bénéficie d'un atout majeur par rapport à l'optique : l'image obtenue est complexe et la phase contient une information sur la distance entre le capteur et le terrain. Il est donc possible de faire interférer un couple d'images complexes pour en déduire le relief [64, 93].
- *Etude de la mer.* Le SAR est devenu un outil performant pour analyser l'état de surface des océans et remonter à des paramètres météorologiques ou à la bathymétrie. Dans les régions polaires, l'analyse de la réflectivité radar de la glace permet le routage des navires à travers la banquise en formation.
- *Détection de cibles.* L'imagerie radar permet de détecter ou de localiser des cibles militaires, éventuellement camouflées ou partiellement enfouies sous un couvert végétal [72, 60].
- *Cartographie.* Cette application, encore en développement, concerne notamment la détection de réseaux routiers ou des zones urbaines [94, 95].

### 1.1.2 Modélisation probabiliste des images

Dans cette partie, nous allons présenter le modèle le plus simple qui permet une description du phénomène de speckle intervenant en imagerie SAR. Le principe de l'acquisition d'une image SAR ne sera pas évoqué ici, mais pour une description détaillée, on pourra se reporter à la littérature abondante sur le sujet [5, 55, 73].

#### Distribution de l'intensité

La valeur complexe  $Z$ , attribuée à un pixel de l'image SAR, représente l'enregistrement du champ électromagnétique rétro-diffusé par la cellule de résolution correspondante sur le terrain. Dans cette cellule se trouvent un nombre fini  $N$  de réflecteurs élémentaires dont les contributions  $a_k e^{j\phi_k}$  s'additionnent de façon cohérente :

$$Z = |Z|e^{j\phi} = \sum_{k=1}^N a_k e^{j\phi_k} \quad (1.1)$$

La valeur de  $Z$  résulte donc de l'interférence des ondes rétro-diffusées par chaque réflecteur élémentaire et dépend donc fortement de la répartition des réflecteurs :  $|Z|$  peut être élevé si ceux-ci sont en phase ou faible dans le cas contraire. Ceci explique pourquoi on observe une grande variabilité d'intensité entre les pixels correspondant à la même zone homogène au sol. Ce phénomène, qui se retrouve dans tous les systèmes d'imagerie cohérente (sonar, échographie, ...), porte le nom de *speckle*.

Goodman a proposé un modèle connu sous le nom de *speckle pleinement développé* qui permet de caractériser la distribution de  $Z$ , considérée comme une variable aléatoire [41]. Les hypothèses de ce modèle sont les suivantes :

- Le nombre  $N$  de diffuseurs élémentaires dans la cellule est grand. Cette hypothèse est raisonnable tant que la taille de la cellule de résolution est grande devant la longueur d'onde  $\lambda$ .
- L'amplitude  $a_k$  et la phase  $\phi_k$  pour chaque réflecteur sont indépendantes l'une de l'autre.
- L'amplitude  $a_k$  et la phase  $\phi_k$  pour un réflecteur sont indépendantes de l'amplitude et de la phase des autres réflecteurs. On fait donc abstraction d'éventuelles interactions entre les réflecteurs.
- Les amplitudes  $a_k$  sont distribuées avec les deux premiers moments finis  $\langle a \rangle$  et  $\langle a^2 \rangle$ .
- Les phases  $\phi_k$  sont uniformément distribuées sur  $[-\pi, \pi]$ . Cette condition implique que la surface soit rugueuse par rapport à  $\lambda$ .

Dans le cadre de ce modèle et pour une zone de réflectivité constante, l'intensité  $I$  reçue en un pixel de l'image suit une loi exponentielle :

$$P_I(I) = \begin{cases} \frac{1}{\mu} e^{-\frac{I}{\mu}} & \text{si } I \geq 0 \\ 0 & \text{sinon} \end{cases} \quad (1.2)$$

où  $\mu$  est la réflectivité radar du sol de la cellule considérée. Cette réflectivité est proportionnelle au coefficient de rétro-diffusion  $\sigma^0$ , grandeur sans dimension caractéristique du type de terrain considéré. Pour la plupart des applications de traitement d'images SAR,  $\sigma^0$  (et donc  $\mu$ ) est la grandeur physique que l'on cherche à mesurer.

La moyenne et la variance de cette loi exponentielle s'écrivent :

$$\langle I \rangle = \mu \quad (1.3)$$

$$Var(I) = \mu^2 \quad (1.4)$$

La réflectivité radar s'identifie donc à l'espérance de l'intensité reçue par le capteur. Soulignons de plus que nous avons l'égalité  $\sqrt{Var(I)} = \mu$ , ce qui signifie que l'écart-type de l'intensité augmente avec la moyenne : les zones claires d'une image ont ainsi une variabilité d'intensité plus importante que les zones sombres. En d'autres termes, le speckle est un bruit multiplicatif. Pour une réflectivité radar  $\mu$ , l'intensité du signal enregistré est en effet :

$$I = \mu \cdot b \quad (1.5)$$

où  $b$  est un bruit de densité de probabilité (ddp) exponentielle (équation 1.2) de moyenne unité ( $\langle b \rangle = 1$ ) qui vient perturber la mesure.

Sur une image SAR d'intensité, le contraste  $C$  (permettant de caractériser la difficulté de traitement d'une image) entre deux zones d'une image, de réflectivités respectives  $\mu_1$  et  $\mu_2$ , est défini comme le rapport des réflectivités radars :

$$C = \frac{\mu_1}{\mu_2} \quad (1.6)$$

## Images multi-vues

Nous avons vu que sur une image d'intensité, le speckle vient dégrader la mesure de la réflectivité  $\mu$ . Pour réduire cet effet, on peut utiliser une technique multi-vues qui consiste à sommer de façon incohérente plusieurs échantillons indépendants issus d'une zone de réflectivité homogène afin de réduire la variance du bruit. Cette sommation peut se faire *dans le temps* si l'on dispose de  $L$  prises de vues de la même zone à des dates différentes ou *dans*

l'espace en moyennant la zone par bloc de  $L$  pixels<sup>1</sup>. Dans le cas d'un moyennage spatial, l'amélioration de la résolution *radiométrique*, c'est-à-dire de la réflectivité radar  $\mu$ , se fait au détriment de la résolution spatiale qui est dégradée d'un facteur  $L$ . L'intensité multi-vues résultant du moyennage de  $L$  échantillons s'écrit :

$$\bar{I} = \frac{1}{L} \sum_{k=1}^L I_k \quad (1.7)$$

Si le speckle est pleinement développé,  $\bar{I}$  est distribuée selon une loi Gamma de moyenne  $\mu$  et d'ordre  $L$  :

$$P_{\bar{I}}(\bar{I}) = \begin{cases} \frac{L^L}{\mu^L \Gamma(L)} \left(\frac{\bar{I}}{\mu}\right)^{L-1} e^{-L\frac{\bar{I}}{\mu}} & \text{si } \bar{I} \geq 0 \\ 0 & \text{sinon} \end{cases} \quad (1.8)$$

La moyenne et la variance s'écrivent :

$$\langle \bar{I} \rangle = \mu \quad (1.9)$$

$$Var(\bar{I}) = \frac{\mu^2}{L} \quad (1.10)$$

La variance du speckle a donc été réduite d'un facteur  $L$ . Notons que si les échantillons  $I_k$  ( $k \in \{1, 2, \dots, L\}$ ) sont corrélés,  $\bar{I}$  suit une loi de Goodman [11, 41]. Cette loi est bien représentée par une loi gamma avec un ordre  $L_{eq} < L$ , appelé *nombre équivalent de vues indépendantes*. La réduction du speckle est alors moindre.

Cette modélisation simple du phénomène de speckle nous sera particulièrement utile dans les chapitres suivants, dans la mesure où elle nous permettra de définir un algorithme adapté aux images SAR, tout en conservant une relative simplicité et de faibles temps de calculs. Avant de poursuivre, soulignons cependant que les hypothèses du speckle pleinement développé ne sont pas toujours satisfaites, notamment en haute résolution où le nombre de diffuseurs par cellule devient faible, ou bien en présence de zones texturées qui introduisent un phénomène de mélange de lois [45, 46, 47].

## 1.2 Segmentation d'images SAR

Nous allons maintenant établir dans cette partie un bref état de l'art de la segmentation d'images SAR, avant de présenter les approches par contours actifs, et notamment les contours actifs statistiques polygonaux, qui peuvent présenter différents intérêts dans certaines situations. Afin de préciser davantage le cadre d'étude de cette thèse, nous serons alors conduits à définir la notion de partition d'images en régions homogènes de façon rigoureuse, contrairement à la notion de segmentation d'images qui reste assez floue et dépend grandement de l'application visée.

### 1.2.1 Etat de l'art en segmentation d'images SAR

Comme nous venons de le voir, une image SAR est perturbée par un bruit de speckle : une zone de réflectivité constante correspond sur l'image à un ensemble de pixels dont les niveaux

1. En pratique, ce traitement est effectué directement sur le signal brut [10].

de gris fluctuent fortement. Il en résulte que les méthodes utilisées pour segmenter des images issues de capteurs optiques passifs ne sont que rarement adaptées à ce type d'images fortement bruitées.

Une première possibilité consiste à filtrer l'image afin de réduire le speckle, avant de la segmenter avec des techniques classiques [57]. Le principal inconvénient de cette approche est lié à l'étape de filtrage qui dégrade la résolution spatiale en lissant les contours. Il semble donc plus judicieux de ne pas chercher à se ramener à des méthodes de segmentation classiques, mais plutôt de développer de nouveaux outils prenant en compte la nature bruitée de l'image.

A ce titre, des méthodes de segmentation par détection de bords ont été développées [29]. Ces méthodes consistent à détecter des transitions entre régions uniformes, plutôt que d'essayer directement d'identifier ces régions. Les algorithmes obtenus nécessitent généralement deux étapes : la première étape consiste à construire une carte de *puissance de contour* de l'image, avant de chercher à extraire, dans une seconde étape, des contours de préférence fermés et squelettisés. L'obtention de cette carte de puissance de contour est habituellement obtenue en utilisant un filtre détecteur de bords. Lorsque les images SAR sont modélisées par un bruit multiplicatif de densité de probabilité Gamma, comme décrit dans la partie précédente, il est possible de définir un tel filtre ayant un *taux de fausse alarme* constant. Bovik [8, 9] et Touzi [92] ont ainsi défini des filtres calculant le rapport normalisé des moyennes entre les deux régions adjacentes d'une fenêtre glissante, qui est translatée sur toute l'image analysée. Dans le contexte de la théorie de la *décision statistique*, Oliver *et al.* [70] ont déterminé un filtre optimal, basé sur le *test d'hypothèse par rapport de vraisemblance* [2, 4]. Ces différentes approches, intéressantes par leur relative simplicité, apparaissent dans de nombreux algorithmes de segmentation d'images SAR [27, 28]. Cependant, il a été montré [36, 28] que ces détecteurs de bords introduisent un biais et une augmentation de la variance d'estimation de la position du contour lorsque la fenêtre utilisée n'a pas la même orientation que le bord<sup>2</sup>. Il a d'ailleurs été prouvé qu'une technique efficace [36, 37, 38], basée sur des contours actifs statistiques, permettait d'affiner la position des bords.

D'autres méthodes, basées sur la croissance, la fusion ou la division de régions ont également été proposées [32, 97, 19]. La précision spatiale de la segmentation obtenue est alors limitée par la taille caractéristique des régions à l'initialisation ou par la validité des tests de fusion quand les régions sont trop petites.

Contrairement aux méthodes précédentes, les méthodes globales consistent généralement à optimiser un critère dépendant de toute l'image. Ces approches sont initialisées à partir d'un modèle donné qu'elles laissent évoluer afin d'optimiser le critère considéré.

Si beaucoup de techniques ont été proposées pour la segmentation globale des images SAR, depuis les travaux de Geman et Geman [31] les méthodes statistiques, qui permettent de prendre en compte un modèle de bruit, ont été très utilisées. Les techniques bayésiennes basées sur les champs de Markov [52, 77, 20, 63, 85] présentent des propriétés intéressantes. Leur principal avantage est la prise en compte efficace d'informations relatives à la formation de l'image (speckle, fonction d'appareil du capteur, ...) dans un cadre statistique, mais aussi la possibilité d'utiliser des méthodes de régularisation efficaces (permettant de régulariser les contours, de limiter le nombre de régions, ...). De telles régularisations sont nécessaires dans la mesure où la segmentation est un problème inverse généralement *mal posé* du point de vue mathématique [90]. Néanmoins, les modèles par champs de Markov introduisent des pa-

---

2. Ce qui est une situation classique en pratique, dans la mesure où les bords peuvent avoir une orientation arbitraire dans l'image.

ramètres ne pouvant être facilement déterminés de façon automatique et peuvent conduire à des problèmes d'optimisation difficiles.

Récemment, les méthodes variationnelles ont suscité un grand intérêt dans le domaine de la segmentation d'image [50, 67, 18, 61, 13, 12, 99, 3, 39, 65], notamment les *contours actifs*. En raison des résultats intéressants obtenus sur des images bruitées (notamment SAR) et de la proximité de ces techniques avec celle que nous avons développée, nous allons présenter un bref historique de la segmentation par contours actifs.

### 1.2.2 Segmentation par contours actifs

Un contour actif est une courbe du plan qui, à partir d'un état initial donné, a la propriété de se déformer, en accord avec l'optimisation d'un critère, pour converger vers un état final définissant directement une segmentation de l'image (fig. 1.1). Si le contour est fermé, l'image est segmentée en deux classes correspondant à l'intérieur et à l'extérieur du contour. L'intérêt majeur d'une telle approche est que les propriétés du contour recherché (courbe fermée, degré de régularité, ...) sont contrôlées *a priori*. On évite donc une étape de post-traitement des contours détectés qui consiste à fermer, amincir et lisser les contours obtenus lors d'une première étape, à l'aide par exemple d'outils issus de la morphologie mathématique, comme l'extraction de bords par ligne de partage des eaux [96, 7].

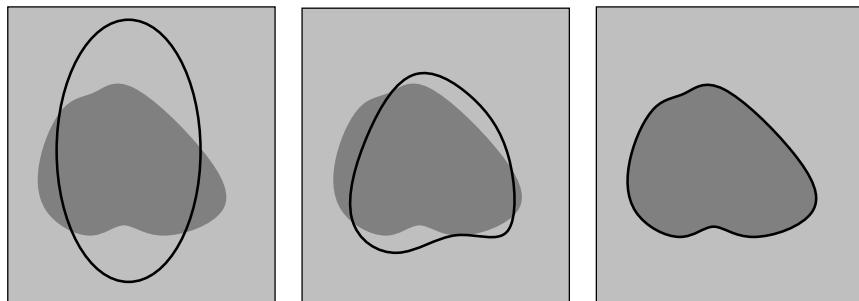


FIGURE 1.1 – *Principe de la segmentation par contour actif. De gauche à droite : initialisation du contour, convergence et résultat final.*

Nous allons voir que ce principe général englobe de nombreuses méthodes qui diffèrent par :

- la définition du critère à optimiser
- le modèle de contour adopté
- la stratégie d'optimisation du critère

#### Le modèle classique

Le premier modèle de contour actif a été introduit en 1988 par Kass, Witkin et Terzopoulos [51]. Les auteurs parlent de *snake* en référence aux ondulations subies par le contour lors de sa convergence. Il se présente sous la forme d'une courbe dans le plan  $C = [x(t), y(t)]^\dagger$ , continue, fermée ou non, paramétrée par son indice curviligne  $t$ .

Le critère à optimiser est une somme pondérée de trois termes qui traduisent des forces de natures différentes et parfois contradictoires s'exerçant sur le snake.

$$E(C) = E_{interne}(C) + \lambda_1 E_{externe}(C) + \lambda_2 E_{interactive}(C) \quad (1.11)$$

On recherche alors l'état d'équilibre du snake, correspondant à un minimum de l'énergie totale définie par (1.11).

L'énergie externe traduit les forces exercées par l'image sur le contour actif : c'est ce terme qui traduit l'attache aux données. Kass *et al.* ont proposé d'utiliser :

$$E_{externe}(C) = - \int_{t_0}^{t_1} |\nabla I(x(t), y(t))|^2 \, dt \quad (1.12)$$

où  $t_0$  et  $t_1$  sont les abscisses curvilignes, éventuellement confondues, des extrémités de la courbe.  $|\nabla I|^2$  représente l'image de puissance de contours obtenue après application d'un détecteur de bord sur l'image  $I(x, y)$ . Pour minimiser cette énergie, le contour actif a tendance à recouvrir les lignes de fortes intensités dans l'image  $\nabla I$ . Cette méthode relève donc de l'approche contour puisque le snake agit comme un extracteur de bord sur l'image de puissance de contours.

L'énergie interne permet quant à elle, de contrôler les propriétés du snake et régularise la solution au sens des problèmes mal posés [90]. Le snake est considéré comme un matériau déformable, caractérisé par ses facultés d'allongement et de courbure. Comme en mécanique, on définit alors sa rigidité locale  $\alpha(t)$  et son élasticité locale  $\beta(t)$  pour écrire l'énergie sous la forme :

$$E_{interne}(C) = \int_{t_0}^{t_1} \left( \alpha(t) \left[ \left( \frac{\partial x}{\partial t} \right)^2 + \left( \frac{\partial y}{\partial t} \right)^2 \right] + \beta(t) \left[ \left( \frac{\partial^2 x}{\partial t^2} \right)^2 + \left( \frac{\partial^2 y}{\partial t^2} \right)^2 \right] \right) \, dt \quad (1.13)$$

L'énergie interactive est présente lorsque la segmentation est semi-automatique avec une aide manuelle de l'utilisateur. Il lui est alors possible d'introduire de manière interactive des informations de haut niveau telles que la présence de frontières préalablement détectées ou de points de passage obligé, par l'intermédiaire de potentiels simulant des forces d'attraction-répulsion sur le snake.

## Stratégie d'optimisation

En adoptant une approche de type modèle stationnaire, la minimisation de la fonctionnelle (1.11) se ramène à la résolution d'une équation aux dérivées partielles (équation d'Euler-Lagrange). La résolution numérique passe par une phase de discrétisation du problème et on est finalement amené à résoudre un système d'équations linéaires. Comme la matrice du système est singulière, il ne peut être résolu directement [6] et il est nécessaire de mettre en œuvre un algorithme de résolution itératif pour approcher la solution.

## Améliorations du modèle classique

L'introduction de cette approche a suscité un vif intérêt dans la communauté des traiteurs de l'image et de nombreuses études ont permis des améliorations remarquables du modèle initial.

Afin de diminuer la sensibilité à l'initialisation, une force de pression a été proposée pour "gonfler" le snake comme un ballon [18]. Cette "force ballon" permet d'améliorer la convergence du snake vers son objectif mais rend moins facile l'obtention du point d'équilibre.

Plus récemment, des modèles basés sur la théorie de l'évolution de courbes planes et des équations aux dérivées partielles géométriques sont apparus. L'arrêt de la convergence ne se fait plus en cherchant un point d'équilibre mais lorsqu'une certaine "fonction d'arrêt" s'annule. Les contours actifs *géodésiques* reposent sur ce principe et permettent la segmentation d'objets avec une topologie complexe [12].

### Contour actif et approche région

Lorsque le contour actif est fermé, il est possible de l'utiliser en adoptant une approche par *régions*, par opposition aux approches par *contours* précédemment étudiées [17]. Il suffit pour cela de considérer l'intérieur et l'extérieur du contour. Ainsi, Ronfard a développé un modèle où le contour actif est déformé sous l'action de forces, normales au contour dont les intensités sont liées à une mesure de similitude entre régions [80]. La similitude entre deux régions  $R_1$  et  $R_2$  est évaluée à l'aide de la *distance de Ward*  $D(R_1, R_2)$ , qui dans le cas étudié se résume à :

$$D[R_1, R_2] = \hat{\sigma}_{R_1 \cup R_2}^2 - \hat{\sigma}_{R_1}^2 - \hat{\sigma}_{R_2}^2 \quad (1.14)$$

où  $\hat{\sigma}_R^2$  représente la variance empirique des niveaux de gris dans la région  $R$ . L'intensité de la force s'écrit alors :

$$D[R_{in}, \delta R] - D[R_{out}, \delta R] \quad (1.15)$$

où  $R_{in}$  et  $R_{out}$  représentent les régions intérieure et extérieure au contour et  $\delta R$  est une petite région dans le voisinage du point d'application de la force. Selon le signe de l'intensité, la force peut donc dilater ou rétracter le contour actif. La taille de  $\delta R$ , le nombre de points sur le contour et l'amplitude du déplacement de ces points sont des paramètres clés dans cette méthode.

### Contours actifs statistiques

Parmi les méthodes de contours actifs basées sur l'approche par régions, certaines ont été développées dans un cadre statistique avec le but de définir des solutions optimales pour un modèle d'image donné et permettant ainsi à la fois une estimation et une régularisation efficace des contours. Si certaines de ces méthodes ont été développées dans le cadre de la segmentation d'un objet unique dans une scène [87, 68, 53, 88, 34, 26, 48, 62], d'autres permettent la segmentation d'une image en plusieurs régions [100, 38].

Dans le cadre de la segmentation d'images SAR, l'approche introduite par Germain *et al.* dans [34] s'est avérée particulièrement intéressante. Il a notamment été démontré que cette méthode, basée sur des contours actifs statistiques polygonaux et sur la modélisation simple du speckle détaillée dans la partie précédente, permettait une bonne précision dans la localisation des contours [37]. Parallèlement, une accélération de cet algorithme a été proposée par Chenaud *et al.* [16], permettant la segmentation d'images bruitées (notamment SAR) en quelques centaines de millisecondes. Ruch *et al.* [82] ont également montré, mais nous reviendrons sur ce point dans la partie 1.3, qu'il était possible d'estimer automatiquement le nombre de noeuds du contour polygonal, permettant ainsi une régularisation efficace du contour sans pour autant nécessiter de réglage de paramètres de pondération comme ce serait le cas en introduisant un terme d'énergie interne (équation 1.13). Soulignons enfin que cette méthode a été ensuite généralisée à une approche multi-régions [38] et utilisée pour la correction du biais observé lors de la détection de bords sur des images SAR de parcelles agricoles. Cependant cette approche nécessite une initialisation avec une pré-segmentation ayant permis de détecter le nombre de régions présentes dans l'image et leur position approximative. De plus, l'accélération proposée

dans [16], n'avait pas été généralisée à cette approche multi-régions dans le cas où les régions ne sont pas simplement connexes (par exemple lorsqu'elles comportent des trous).

A la vue de ces derniers résultats, la segmentation d'images SAR à l'aide de contours actifs statistiques polygonaux semble prometteuse. Cependant, dans toutes les études relatives à cette méthode qui viennent d'être présentées, le nombre de régions dans l'image est toujours supposé connu. Pour lever cet *a priori* très fort, nous devons nous pencher sur le problème de l'estimation du nombre de régions présentes dans l'image.

Mais avant d'introduire le principe de minimisation de la complexité stochastique, outil pouvant s'avérer très intéressant pour l'estimation d'ordre de modèles, nous allons nous pencher plus en détail sur la notion de partition en régions homogènes qui sera l'objectif à atteindre des algorithmes que nous développerons.

### 1.2.3 Partition en régions homogènes et segmentation

L'objectif de la segmentation d'images consiste le plus souvent à partitionner une image en régions d'intérêt afin de pouvoir les analyser ultérieurement. La segmentation d'une image sera ainsi jugée satisfaisante si toutes ses régions d'intérêt ont été partitionnées avec une bonne précision. Evidemment la notion de *région d'intérêt* est difficile à définir, d'autant plus qu'elle dépend fortement de l'application visée et fait en réalité référence le plus souvent à des concepts de haut niveau (c'est-à-dire symboliques).

Dans le cadre de techniques de bas niveau, il est plus simple de définir la notion de partition en régions homogènes. C'est l'objectif que l'on s'est fixé pour les algorithmes que nous avons développés.

La définition standard de l'homogénéité repose sur la modélisation statistique des images. A ce titre, une image  $s(x, y)$  est considérée comme la réalisation d'un champ aléatoire  $S_\lambda(x, y)$ , qui dépend de l'évènement aléatoire  $\lambda$ . Notons  $s_1(x, y), s_2(x, y), \dots, s_P(x, y)$  différentes réalisations de ce champ aléatoire. Si le nombre de réalisations  $P$  est grand, il devient possible d'estimer la probabilité  $P_{S,x,y}(s)$  d'observer la valeur  $s$  au pixel de coordonnées  $(x, y)$ . Ceci nous amène alors à la définition de l'homogénéité au sens strict. Un champ aléatoire  $S_\lambda(x, y)$  est homogène *au sens strict* à l'ordre 1 si  $P_{S,x,y}(s)$  est indépendante des coordonnées  $(x, y)$ . Il est ainsi possible de définir une région homogène comme l'ensemble des pixels de coordonnées  $(x, y)$  ayant la même loi de probabilité  $P_{S,x,y}(s)$ .

L'estimation de  $P_{S,x,y}(s)$  à partir d'un échantillon étant une tâche difficile, il est parfois suffisant de s'intéresser aux moments d'ordre 1 et 2, c'est-à-dire à  $\langle S_\lambda(x, y) \rangle$  et  $\langle S_\lambda(x, y) S_\lambda(x + \eta, y + \mu) \rangle$ , où nous avons noté  $\langle \cdot \rangle$  la moyenne sur l'ensemble des réalisations de l'évènement aléatoire  $\lambda$ . Une région sera alors homogène *au sens large* si l'ensemble de ses pixels sont tels que  $\langle S_\lambda(x, y) \rangle$  et  $\langle S_\lambda(x, y) S_\lambda(x + \eta, y + \mu) \rangle$  ne dépendent pas de  $x$  et  $y$ .

Dans notre cas, nous ne disposons que d'une seule réalisation du champ aléatoire  $S_\lambda(x, y)$ , à savoir l'image  $s(x, y)$ . Il n'est donc pas possible d'estimer  $P_{S,x,y}(s)$ , ni même  $\langle S_\lambda(x, y) \rangle$  ou  $\langle S_\lambda(x, y) S_\lambda(x + \eta, y + \mu) \rangle$  en chaque pixel. Nous nous proposons donc de choisir une définition de la partition en régions homogènes plus pragmatique.

Pour cela, considérons une image  $s(x, y)$ , générée à partir d'un patchwork de  $R$  régions homogènes, notées  $\Omega_1, \Omega_2, \dots, \Omega_R$ . Ce patchwork est défini par la partition  $w(x, y)$  (fig. 1.2.a) telle que  $w(x, y) = r$  si et seulement si  $(x, y) \in \Omega_r$ . A chacune de ces régions  $\Omega_r$ , nous associons une probabilité  $P_{S,\Omega_r}(s)$ . L'image bruitée  $s(x, y)$  est alors obtenue en effectuant en chaque pixel

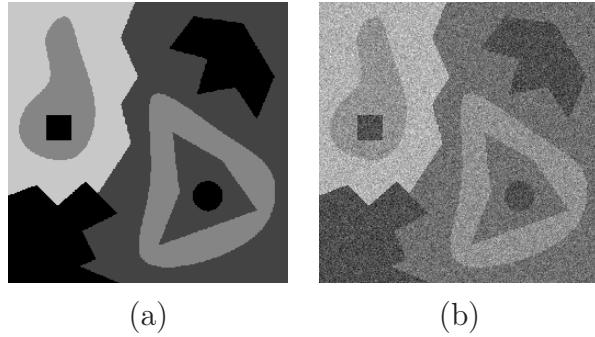


FIGURE 1.2 – Synthèse d'une image constituée d'un patchwork de régions homogènes. (a) Partition  $w(x, y)$ . (b) Image bruitée associée.

de coordonnées  $(x, y)$ , un tirage aléatoire selon la loi de probabilité qui correspond à la région à laquelle appartient le pixel (fig. 1.2.b). En d'autres termes, nous générerons  $R$  images  $a_r(x, y)$ , où chaque image  $a_r$  est la réalisation d'un champ aléatoire de probabilité  $P_{S, \Omega_r}(s)$ . L'image bruitée  $s$  est alors définie de la sorte :

$$s(x, y) = \sum_{r=1}^R a_r(x, y) \cdot \delta [w(x, y), r] \quad (1.16)$$

où  $\delta(a, b)$  est le symbole de Kronecker ( $\delta(a, b) = 1$  si  $a = b$ , et 0 sinon).

La définition de la partition en régions homogènes découle directement de cette procédure. Connaissant une image bruitée  $s$ , le but de la partition en régions homogènes est de retrouver la fonction de partition  $w(x, y)$ .

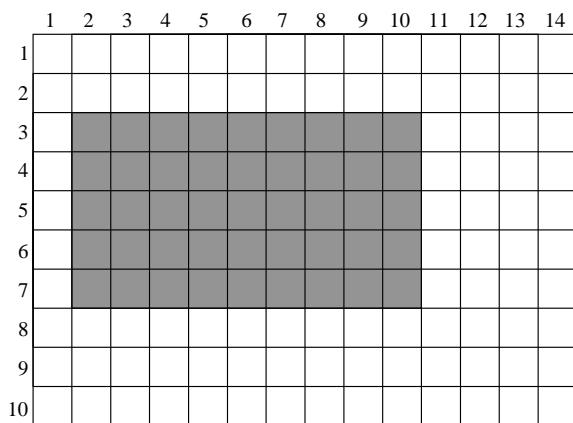
Même si nous avons choisi de nous restreindre à la partition en régions homogènes, soulignons que cette restriction offre les avantages suivants :

- elle s'applique assez bien à la modélisation des images SAR de parcelles agricoles,
- elle permet de travailler dans le cadre d'une modélisation statistique rigoureuse du problème de la partition,
- elle aboutit à des algorithmes simples et rapides.

Ceci explique d'ailleurs que beaucoup de méthodes de segmentation d'images soient basées sur cette notion de partition en régions homogènes, la notion d'homogénéité n'étant d'ailleurs pas nécessairement définie en termes probabilistes [56, 40, 80, 14, 38, 74, 100].

### 1.3 Complexité stochastique

Le principe de la Minimisation de la Complexité Stochastique (MCS), est un principe très général introduit par Rissanen en 1978 [78]. Basé sur des concepts issus à la fois de la théorie de l'information et de l'estimation statistique, ce principe permet notamment de résoudre des problèmes d'estimation lorsque l'ordre du modèle est inconnu, problème qui se pose notamment dans le cas de la partition en régions homogènes telle que nous l'avons définie précédemment. Ce principe offre l'avantage d'être intuitivement assez naturel. De plus, il a été rapidement appliqué dans différents domaines, notamment la segmentation d'images, et a donné des résultats intéressants.



**Codage 1 :** (liste des niveaux de gris)

```
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1
1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0
1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

**Codage 2 :** (description) *rectangle noir sur fond blanc ayant un coin en (3,2) et en (10,7)*

FIGURE 1.3 – Exemples de codage d'une image binaire de taille  $14 \times 10$  pixels.

Dans cette partie, nous allons donc présenter rapidement les concepts sous-jacents à ce principe de la MCS, avant d'analyser plus en détail ses applications dans le domaine de la segmentation d'images.

### 1.3.1 Principe de la minimisation de la complexité stochastique

Avant d'entrer dans le vif du sujet, commençons par un petit exemple. Sur la fig. 1.3, nous pouvons voir une image binaire contenant  $14 \times 10$  pixels. Supposons que nous voulions la transmettre par téléphone à notre interlocuteur. Pour cela, il nous suffit de la lui décrire, ou en d'autres termes, il suffit de coder l'image et de transmettre ce code. Nous pouvons ainsi proposer deux codes. Le premier, très basique, consiste à donner la liste des niveaux de gris des pixels dans l'ordre lexicographique, avec pour convention 0 = noir et 1 = blanc. Le deuxième codage repose quant à lui sur une description *intelligente* et concise de l'image, consistant simplement à dire que l'image est constituée d'un rectangle noir sur un fond blanc.

En comparant ces deux codages, nous constatons immédiatement les deux faits suivants :

- la lecture du premier codage sera longue et fastidieuse, alors que la lecture du deuxième est beaucoup moins complexe.
- le deuxième codage repose sur une analyse de la structure de l'image, contrairement au premier.

Nous constatons ainsi que pour cette image, une analyse de sa structure permet de la transmettre de façon beaucoup plus concise que si nous l'avions transmise basiquement comme avec le codage 1. La sagesse populaire a d'ailleurs exprimé cela à travers ce petit proverbe : *ce qui se conçoit bien, s'énonce [simplement]*.

Partant de ce constat, le principe de la MCS consiste à énoncer la réciproque de ce proverbe qui pourrait s'écrire : *ce qui est énoncé [simplement], a été bien compris*.

Toujours à propos de l'image de la fig. 1.3, cette réciproque consiste à dire que, le codage 2 étant moins complexe que le codage 1, il repose sur une meilleure compréhension de l'image. Toute la difficulté est maintenant de définir ce que nous entendons par le terme *moins complexe*, c'est pourquoi nous allons nous pencher sur les notions qui sont à la base de ce principe, à savoir les notions de quantité d'information et de complexité.

## Complexité de Kolmogorov

Analysons dans un premier temps la notion de complexité introduite par Kolmogorov dans les années soixante. La complexité de Kolmogorov  $C(\chi)$  d'une suite de nombre  $\chi = x_1, x_2, \dots, x_n$  de longueur  $n$  est égale à la longueur du plus petit programme (mesurée en bits) qui permet de l'engendrer. La complexité de Kolmogorov est alors maximale quand il n'existe pas d'algorithme plus court qu'une simple énumération. Dans ce cas, si la longueur de la suite  $\chi$  est  $n$ , nous avons  $C(\chi) = n + Cte$  où  $Cte$  est une valeur indépendante de  $n$ .

Cette définition semble particulièrement bien adaptée pour décrire l'information contenue dans une image. Plus une image sera difficile à décrire, plus sa complexité de Kolmogorov sera grande, ce qui pourrait par exemple permettre de mesurer la complexité de la structure d'une image. Cependant, cette notion se heurte à de sérieuses limitations.

La première de ces limitations est directement liée au calcul de cette complexité [21]. En effet, dans la mesure où une suite  $\chi$  de longueur  $n$  a une complexité d'au plus  $C(\chi) = n + Cte$ , nous pourrions penser qu'afin de trouver le plus court programme, il suffit d'essayer tous ceux comportant moins de  $n + Cte$  bits et d'analyser leur résultat. Le plus petit programme ayant engendré la suite  $\chi$  permettrait alors d'en déduire la complexité de Kolmogorov de cette suite. Cependant, parmi les programmes testés, un certain nombre ne s'arrêteront jamais et il n'est pas possible de savoir *a priori* lesquels. Ce n'est donc pas parce qu'au bout d'une durée donnée aucun programme n'aura fourni un résultat correspondant à cette image que les programmes qui continuent à calculer ne le fourniront pas plus tard. Cela signifie en particulier qu'il n'est pas possible de déterminer la complexité de Kolmogorov d'une suite en un temps fini, ce qui limite donc grandement son utilisation.

Rissanen [79] souligne également une deuxième limitation de l'utilité de la complexité de Kolmogorov dans le cas qui nous intéresse. En effet, notre but est de déterminer le meilleur modèle permettant de décrire la suite  $\chi$ , autrement dit la structure sous-jacente à cette suite. Cependant, une description des propriétés de cette suite n'est pas nécessairement facile à extraire du programme le plus court. Il semble plus efficace *a priori* d'analyser la complexité de l'image au travers du modèle dans lequel ces propriétés sont faciles à identifier.

## Information de Shannon

Une façon de résoudre le problème de non-calculabilité de la complexité de Kolmogorov avait déjà été proposée par Shannon depuis 1948 [84]. Son approche consiste à ne pas considérer chaque réalisation indépendamment du processus aléatoire dont elle est issue. L'idée de Shannon pour définir la notion de quantité d'information consiste à considérer qu'une réalisation apporte d'autant plus d'information qu'elle était improbable. La quantité d'information  $Q(\chi)$  associée à une suite  $\chi$  est alors directement reliée à sa probabilité d'apparaître  $P(\chi)$  :

$$Q(\chi) = -\log_2 P(\chi) \quad (1.17)$$

où  $\log_2$  est le logarithme en base 2.

Même si l'approche de Shannon est radicalement différente de celle de Kolmogorov - puisque l'ambition de Kolmogorov était de caractériser une suite indépendamment de sa loi de probabilité alors que Shannon caractérise chaque suite en fonction de l'ensemble des réalisations possibles au regard de leur probabilité - il est possible de montrer [21] que la valeur moyenne de la complexité de Kolmogorov d'une série de réalisations de suites  $\chi$  issues d'une certaine

loi de probabilité, est en fait égale, quand le nombre de réalisations est grand, à l'espérance mathématique de leur quantité d'information (c'est-à-dire l'entropie de Shannon [84]).

Cela signifie en particulier que la quantité d'information de Shannon peut être considérée comme une approximation du nombre de bits nécessaires pour coder une suite, lorsque la loi de probabilité l'ayant engendrée est connue. Le code associé, ou code entropique, n'est en effet optimal (c'est-à-dire correspond à un nombre minimal de bits) qu'en moyenne sur toutes les réalisations de suites.

## Complexité stochastique

Si la notion de quantité d'information de Shannon a permis d'obtenir une approximation du nombre de bits nécessaires au codage d'une suite, cette approche reste cependant beaucoup moins générale que la complexité de Kolmogorov. En effet, la quantité d'information de Shannon suppose que la loi de probabilité ayant permis de générer la suite considérée est connue. Cela signifie en particulier que si l'on désire transmettre une suite  $\chi$  avec une approche de type Kolmogorov, il suffit de transmettre un nombre de bits égal à la complexité de Kolmogorov de cette suite. En revanche, avec une approche de type Shannon, la transmission d'un nombre de bits égal à la quantité d'information de Shannon de cette suite ne sera pas suffisante pour reconstruire cette suite, dans la mesure où le récepteur doit connaître la loi de probabilité  $P(\chi)$  - c'est-à-dire en fait le modèle - pour pouvoir décoder le message et reconstruire la suite.

C'est notamment dans le but de pallier à ce problème que Rissanen a introduit la notion de complexité stochastique [79]. Cette notion consiste à substituer à la complexité de Kolmogorov le nombre de bits qu'il faudrait pour coder la suite avec un code entropique auquel il faut ajouter le nombre de bits nécessaires pour décrire le modèle probabiliste permettant de déterminer ce code.

La complexité stochastique - appelée ainsi par opposition à la complexité *algorithmique* de Kolmogorov - permet alors de définir une mesure de la complexité intégrant un terme relatif au modèle sous-jacent aux données.

## Principe de la minimisation de la complexité stochastique

Rissanen proposa dès les années 1970 un principe basé sur la minimisation de la complexité, et dénommé principe de la longueur de description minimale (ou MDL en langue anglaise, pour Minimum Description Length). Soulignons que même si dès le début, l'expression de cette "longueur de description" est analogue à la complexité stochastique dans la plupart des exemples qu'il traite, ce n'est qu'en 1989 qu'il introduit explicitement cette notion [79], dans un ouvrage de synthèse de ses principaux travaux sur ce thème. Dans ce manuscrit, nous avons choisi de nous référer à cet ouvrage, c'est pourquoi nous utilisons la notion de complexité stochastique. De même, nous parlerons non pas du principe du MDL, mais du principe de Minimisation de la Complexité Stochastique (MCS). En effet, le terme MDL reste assez ambigu dans la mesure où la longueur de description correspond à la complexité de Kolmogorov d'un point de vue exact, alors que la notion de complexité stochastique fait bien référence à l'aspect stochastique du problème : ce n'est donc qu'en moyennant sur différentes réalisations que ce code est optimal.

Ce principe propose ainsi un critère permettant de choisir le meilleur modèle parmi un jeu de modèles, et ceci sans connaissance *a priori* du véritable modèle sous-jacent. Si ce modèle s'avère par ailleurs être aussi le meilleur en l'analysant à l'aide d'autres critères (qui

eux peuvent être basés sur des connaissances *a priori* ou qui peuvent être beaucoup plus subjectifs), nous pourrons alors dire que le principe de la MCS est pertinent.

### 1.3.2 Segmentation et minimisation de la complexité stochastique

En 1989, Leclerc proposa une première application du principe de minimisation de la complexité stochastique à la segmentation d'images [56]. Si une image  $s(x, y)$  ( $(x, y) \in [1, N_x] \times [1, N_y]$ ) contenant  $N = N_x \times N_y$  pixels est composée d'un patchwork de  $R$  régions, notées  $\Omega_r$  ( $r \in \{1, 2, \dots, R\}$ ) de moyennes respectives  $m_r$  et perturbée par un bruit blanc gaussien d'écart-type  $\sigma$  constant sur l'image, la complexité stochastique de l'image proposée par Leclerc s'écrit :

$$a \sum_{r=1}^R \sum_{(x,y) \in \Omega_r} \left( \frac{s(x, y) - m_r}{\sigma} \right)^2 + b \sum_{r=1}^R C_r \quad (1.18)$$

où  $C_r$  est la longueur du contour (en pixel) de la région  $\Omega_r$  et où  $a = \frac{1}{2 \log 2}$ . En revanche la valeur de  $b$  n'est pas donnée explicitement dans cet article. Soulignons aussi que Leclerc propose une généralisation de cet algorithme au cas où  $\sigma$  n'est plus constant sur l'image mais varie d'une région à l'autre et où la moyenne  $m_r$  n'est plus constante sur chaque région  $\Omega_r$  mais peut varier selon une fonction polynomiale dépendant de la position  $(x, y)$  et s'écrit donc  $m_r(x, y)$ . Les principales différences avec notre approche sont liées d'une part au type d'images étudiées (optiques pour Leclerc, SAR dans cette thèse), mais surtout à la description du contour qui est faite pixel par pixel dans le cas de Leclerc alors que nous utiliserons une description paramétrique du contour à l'aide de polygones.

En 1994, Kanungo *et al.* [49] - puis plus tard Lee [59] (en 1998) - ont utilisé une approche basée sur la minimisation de la complexité stochastique et sur des fusions de régions. Le critère utilisé dans les deux cas ne contient aucun paramètre à régler de la part de l'utilisateur. Si l'article de Kanungo *et al.* est relatif aux images multi-bandes, avec des moyennes dans chaque région pouvant être distribuées selon des fonctions polynomiales et si l'article de Lee concerne la segmentation d'images en présence de bruit corrélé, ces deux articles ne traitent cependant que du cas de bruits gaussiens, et ne sont donc pas adaptés à la segmentation d'images SAR. De plus, leurs algorithmes étant basés sur des fusions de régions, ils doivent être initialisés à l'aide d'une pré-segmentation satisfaisante.

En 1996, Zhu et Yuille [100] proposèrent un algorithme de *compétition de régions*, combinant dilatation de régions, fusion de régions et contours actifs, et adapté à la segmentation d'images optiques. Leur approche propose une interprétation du principe de la MCS dans un cadre bayésien. En raison de cette interprétation bayésienne, le critère à minimiser, bien que dérivé du principe de la MCS, contient plusieurs paramètres de pondérations réglables en fonction de l'image considérée, leur permettant ainsi de segmenter des images vidéos complexes.

Figueiredo *et al.* ont alors montré [26] dans le cadre de contours actifs que le principe de la MCS pouvait être utile pour obtenir un algorithme sans paramètre à régler et adapté à la segmentation d'un objet unique en présence non seulement d'un bruit gaussien mais aussi d'un bruit de Rayleigh. Plus précisément, le principe de la MCS sert à estimer l'ordre du modèle utilisé pour décrire le contour, c'est-à-dire le nombre de points de contrôle des B-splines du contour. Ces résultats ont ensuite été généralisés au cas de contours polygonaux par Ruch *et al.* [82]. La complexité stochastique déterminée précédemment dans [26] leur a

alors permis d'estimer le nombre de nœuds du contour polygonal, même en présence d'objets fortement non convexes, à l'aide d'un algorithme simple et rapide. Soulignons une fois encore que ces deux approches, contrairement aux précédentes, ont été développées dans le cadre de la segmentation d'un seul objet simplement connexe.

### 1.3.3 Analyse du principe de la minimisation de la complexité stochastique au vu de ces quelques travaux

Nous pouvons dégager plusieurs aspects intéressants dans le principe de la MCS. Pour cela, définissons une classe de modèles  $\mathcal{M} = \{\mathcal{M}_k, k \in [1, K]\}$ , où chaque modèle  $\mathcal{M}_k$  est défini par un vecteur de paramètre  $\theta^k$  dont la taille peut varier avec  $k$ . Soit  $\chi$  l'échantillon que nous voulons étudier. La complexité stochastique associée au modèle  $\mathcal{M}_k$  est la longueur de code  $\Delta(\chi, \mathcal{M}_k)$  nécessaire pour décrire l'échantillon. Cette longueur de code se décompose en deux termes : la longueur de code  $\Delta(\chi|\theta^k)$  nécessaire pour décrire les données connaissant les paramètres du modèle et la longueur de code  $\Delta(\theta^k)$  nécessaire pour décrire les paramètres du modèle. Nous avons ainsi :

$$\Delta(\chi, \mathcal{M}_k) = \Delta(\chi|\theta^k) + \Delta(\theta^k) \quad (1.19)$$

Le premier terme peut être interprété comme un terme d'attache aux données. Quant au second terme, relatif au codage du modèle, il joue le rôle de terme de régularisation.

Le premier intérêt du principe de la MCS est donc de fournir un terme de régularisation différent de ceux plus classiquement utilisés [90]. Vu sous cet angle, ce principe permet simplement de proposer un choix pour le terme de régularisation dans une approche bayésienne. Les articles de Leclerc [56] et surtout de Zhu et Yuille [100] mettent essentiellement l'accent sur ce point.

Une autre propriété intéressante est qu'il n'est pas nécessaire d'introduire de paramètres de pondérations entre le terme d'attache aux données  $\Delta(\chi|\theta^k)$  et le terme de régularisation  $\Delta(\theta^k)$ , dans la mesure où ils s'expriment tous deux dans la même unité, à savoir le *bit*. La complexité stochastique constitue ainsi un critère à minimiser ne contenant aucun paramètre à régler de la part de l'utilisateur. Les premiers à avoir souligné ce point sont, à notre connaissance, Kanungo *et al.* [49], même si bien sûr cette propriété est aussi mise en avant dans [59, 26, 82].

Le dernier point que nous soulignerons est relatif à l'estimation de l'ordre du modèle. Dans [56, 49, 100, 59], la minimisation de la complexité stochastique permet en effet d'estimer le nombre de régions présentes dans l'image, alors que dans [26, 82] elle conduit à une estimation de l'ordre du modèle paramétrique utilisé pour décrire le contour d'un objet, permettant ainsi la régularisation efficace du contour des objets dans des images bruitées.

Dans cette thèse, nous allons présenter une méthode de partition d'images basée sur la minimisation de la complexité stochastique combinant ces différents aspects. La méthode obtenue permettra alors d'estimer non seulement le nombre de régions contenues dans l'image mais également l'ordre du modèle paramétrique utilisé pour décrire les contours des régions.

## 1.4 Conclusion

Dans ce chapitre, nous avons dans un premier temps présenté un modèle simple du phénomène de speckle présent en imagerie SAR. Nous avons ensuite vu que cette modélisation avait été utilisée dans le cadre de la segmentation par contours actifs statistiques polygonaux

et avait ainsi conduit à des algorithmes simples et rapides adaptés aux images SAR. Il est alors apparu nécessaire de définir la notion de partition en régions homogènes, définition plus précise que celle de segmentation, afin de pouvoir proposer, dans un cadre plus rigoureux, la généralisation de ces résultats lorsque le nombre de régions dans l'image n'est plus supposé connu *a priori*. Enfin, nous avons présenté le principe de la minimisation de la complexité stochastique, qui sera à la base de la méthode de partition proposée dans cette thèse et nous permettra notamment d'estimer le nombre de régions dans l'image, ainsi que le nombre de nœuds du modèle polygonal utilisé pour décrire les contours de ces différentes régions.

Dans le chapitre suivant, nous allons présenter la mise en application de ces principes afin d'obtenir une méthode de partition d'images en régions homogènes.



## Chapitre 2

# Partition en régions homogènes par minimisation de la complexité stochastique

### Sommaire

---

<b>2.1</b>	<b>Complexité stochastique</b>	<b>24</b>
2.1.1	Modèle d'image	24
2.1.2	Détermination de la complexité stochastique	25
<b>2.2</b>	<b>Algorithme d'optimisation proposé</b>	<b>30</b>
2.2.1	Fusion de régions	30
2.2.2	Déplacement des nœuds	31
2.2.3	Suppression des nœuds	31
2.2.4	Procédure globale d'optimisation	32
2.2.5	Algorithme rapide	35
<b>2.3</b>	<b>Exemples de partition en régions homogènes</b>	<b>36</b>
2.3.1	Images synthétiques	36
2.3.2	Sélection automatique de la grille initiale	37
<b>2.4</b>	<b>Conclusion</b>	<b>40</b>

---

La méthode de partition en régions homogènes que nous allons présenter dans ce chapitre repose sur différents concepts introduits précédemment, notamment le principe de minimisation de la complexité stochastique (MCS), utilisé dans le cadre des contours actifs statistiques, ou plus précisément d'une grille active statistique. Cette technique, adaptée aux images perturbées par un bruit de speckle, repose sur la minimisation de la complexité stochastique de l'image. Nous allons dans un premier temps nous attacher à déterminer la complexité stochastique associée à une partition arbitraire (voir partie 2.1). Nous présenterons ensuite dans la partie 2.2, l'algorithme retenu pour minimiser cette complexité stochastique. Nous terminerons alors ce chapitre (partie 2.3) par une série de résultats de partition en régions homogènes sur quelques images synthétiques. Les résultats sur des images SAR réelles seront présentés dans les chapitres suivants.

## 2.1 Complexité stochastique

Dans cette partie, nous allons nous attacher à déterminer la complexité stochastique d'une image lorsqu'elle est décrite à l'aide d'une partition donnée.

### 2.1.1 Modèle d'image

Avant de calculer la complexité stochastique associée à une partition, il est nécessaire de définir le modèle statistique des images considérées.

Nous cherchons à partitionner des images homogènes par morceaux. En d'autres termes, nous considérons que les images que nous étudions correspondent à un patchwork de régions homogènes c'est-à-dire dont les niveaux de gris des pixels, pour chaque région, sont décrits par des variables aléatoires associées à la même loi de probabilité. Nous supposerons de plus que les niveaux de gris des différents pixels de chaque région peuvent être correctement décrits par des variables aléatoires indépendantes. Ces hypothèses peuvent apparaître très restrictives *a priori*. En effet, bien qu'elles correspondent à un certain nombre de situations réalistes et en particulier à celle qui nous motive au premier plan, c'est-à-dire la segmentation d'images SAR (et notamment les images de parcelles agricoles), on comprend aisément qu'un tel modèle d'image puisse être inadapté pour décrire un grand nombre de situations. Il présente cependant deux intérêts principaux, ce qui explique qu'il ait été utilisé à maintes reprises [40, 80, 100, 14, 38, 74]. Il permet tout d'abord de bien poser le problème de la partition d'images en régions homogènes. Le deuxième avantage, qui est certainement le plus important, est la simplicité de la technique de partition à laquelle il conduit.

Nous proposons donc d'évaluer ce modèle à la lumière des résultats auxquels il permet d'aboutir. La définition de ses limites sera alors analysée *a posteriori*. Le principe qui nous guide est donc de considérer un modèle simple qui permet d'aboutir à un algorithme simple et rapide et de chercher à en définir le domaine d'application le plus prometteur. Notre motivation principale sera la segmentation d'images radar, pour lesquelles ce modèle nous semble être un bon compromis entre simplicité et capacité à modéliser l'essentiel des caractéristiques que l'on cherche à décrire. De plus, le résultat de la partition en régions homogènes devra être considéré comme une étape dans un processus de traitement d'images plus complexe, comme l'analyse d'images ou la reconnaissance de formes.

Soit une image

$$\mathbf{s} = \{s(x, y) | (x, y) \in N_x \times N_y\} \quad (2.1)$$

contenant  $N = N_x \times N_y$  pixels et où  $s(x, y)$  représente le niveau de gris du pixel de coordonnée  $(x, y)$ . Nous considérons que cette image est composée de  $R$  régions  $\Omega_r$  ( $r \in \{1, 2, \dots, R\}$ ). Chacune des régions  $\Omega_r$  est supposée homogène, ce qui signifie que tous les pixels d'une même région  $\Omega_r$  sont distribués selon la même densité de probabilité (ddp)  $P_r$ . Nous supposerons de plus que les ddp  $P_r$  de toutes les régions  $\Omega_r$  appartiennent à la même famille, et qu'elles ne se différencient donc que par leurs paramètres. Ainsi, les niveaux de gris des pixels seront supposés par exemple tous distribués selon une loi gaussienne, quelle que soit la région considérée, mais la moyenne  $m_r$  et l'écart-type  $\sigma_r$  des lois gaussiennes pourront varier en fonction de la région considérée. La ddp dans la région  $\Omega_r$  peut alors être écrite sous la forme  $P_r = P_{\theta_r}$ , où  $\theta_r$  est le vecteur de paramètres de la ddp dans la régions  $\Omega_r$  (pour des lois gaussiennes, on aurait donc  $\theta_r = (m_r, \sigma_r)$ ).

La partition de l'image peut être décrite à l'aide de la fonction de partition  $w$ , qui prend ses valeurs dans  $\{1, 2, \dots, R\}$ , de la façon suivante :

$$w(x, y) = r \Leftrightarrow (x, y) \in \Omega_r \quad (2.2)$$

et nous pouvons écrire :

$$s(x, y) = \sum_{r=1}^R a_r(x, y) \delta[w(x, y), r] \quad (2.3)$$

où les valeurs  $a_r(x, y)$  sont des variables aléatoires distribuées selon la ddp  $P_{\theta_r}$  de vecteur de paramètres  $\theta_r$ . Dans l'équation 2.3,  $\delta(i, j)$  représente le symbole de Kronecker

$$\delta(i, j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \quad (2.4)$$

En d'autres termes, dans l'équation 2.3, nous avons

$$\delta[w(x, y), r] = \begin{cases} 1 & \text{si } (x, y) \in \Omega_r \\ 0 & \text{sinon} \end{cases} \quad (2.5)$$

Partitionner une image en régions homogènes consistera à trouver la fonction de partition  $w$  la plus adaptée à l'image analysée. Il convient bien entendu de préciser cette notion. La méthode que nous proposons reposera sur la minimisation de la complexité stochastique. Nous allons tout d'abord déterminer cette complexité stochastique pour une image décrite par une partition  $w$  arbitraire.

### 2.1.2 Détermination de la complexité stochastique

Rappelons que la complexité stochastique est une approximation de la plus petite longueur de code nécessaire pour décrire une image et se mesure usuellement en *bits*. Pour des raisons de simplicité dans les calculs, nous choisirons dans la suite d'exprimer cette complexité en *nats*, où  $1 \text{ nat} = \log 2 \text{ bits}$  (c'est-à-dire  $1 \text{ nat} \approx 0.693 \text{ bits}$ ). Ce choix permet en effet d'utiliser le logarithme népérien ( $\log$ ) à la place du logarithme en base 2 ( $\log_2$ ). La complexité stochastique introduite par Rissanen [79] consiste à coder d'une part les données avec le code optimal au sens de la théorie de la communication de Shannon [84], dénommé code entropique dans la suite, et d'autre part les paramètres du modèle ayant servi à l'obtention de ce code entropique. Dans notre cas, puisque nous considérons une image constituée d'un patchwork

de régions homogènes, il conviendrait de coder les niveaux de gris de chaque région homogène avec leur code entropique. Cependant, ce code entropique ayant été déterminé grâce à la connaissance des ddp  $P_{\theta_r}$  dans chaque région, un utilisateur ne pourra espérer le décoder sans la connaissance de ces ddp. Et il en est de même si l'on ne connaît pas la partition de l'image en régions homogènes. Il convient donc de fournir également, en plus du code entropique, les ddp dans chacune des régions ainsi que la partition de l'image. Dans cette section, notre objectif est de déterminer la complexité stochastique d'une image associée à une partition donnée  $w$ .

Comme nous venons de l'évoquer, cette complexité stochastique se décompose en deux termes, un premier représentant la longueur de code associée au modèle de l'image et un deuxième relatif à la longueur de code associée au codage des données connaissant le modèle. Dans notre cas, le modèle est constitué de deux parties : la partition  $w$  ainsi que les ddp  $P_{\theta_r}$  dans chacune des  $R$  régions  $\Omega_r$  définies par la partition  $w$ . Puisque chacune de ces ddp ne se différencie que par son vecteur de paramètres  $\theta_r$ , le codage des ddp se ramène simplement au codage des  $\theta_r$ . Dans la suite, nous noterons  $\boldsymbol{\theta} = (\theta_r)_{r \in \{1, 2, \dots, R\}}$ , l'ensemble de ces  $R$  vecteurs de paramètres.

Ainsi, si l'on note  $\Delta(\mathbf{s}, w, \boldsymbol{\theta})$  la complexité stochastique nécessaire pour décrire l'image  $\mathbf{s}$  associée à la partition  $w$  et aux  $R$  ddp définies par  $\boldsymbol{\theta}$ , nous pouvons écrire :

$$\Delta(\mathbf{s}, w, \boldsymbol{\theta}) = \Delta_G(w) + \Delta_P(\boldsymbol{\theta}|w) + \Delta_L(\mathbf{s}|\boldsymbol{\theta}, w) \quad (2.6)$$

où  $\Delta_G(w)$  et  $\Delta_P(\boldsymbol{\theta}|w)$  sont les deux termes de codage du modèle associés à la partition  $w$  et où  $\Delta_L(\mathbf{s}|\boldsymbol{\theta}, w)$  est le terme de codage de l'image  $\mathbf{s}$  connaissant le modèle. Le premier terme de codage du modèle,  $\Delta_G(w)$ , est égal au nombre de nats nécessaires au codage de la partition  $w$  et le deuxième,  $\Delta_P(\boldsymbol{\theta}|w)$ , est le nombre de nats nécessaires au codage des vecteurs de paramètres  $\theta_r$  dans chacune des  $R$  régions définies par la partition  $w$ .

Détaillons maintenant chacun des trois termes de l'équation 2.6.

### Terme de codage de la partition : $\Delta_G(w)$

En nous basant sur différents travaux présentés dans le chapitre précédent (notamment [16, 38, 82]), nous avons choisi de décrire la partition  $w$  à l'aide d'une grille polygonale. Une telle grille est constituée d'un ensemble de noeuds reliés entre eux par des segments, de façon à délimiter différentes régions. Nous noterons  $k$  le nombre de noeuds de la grille et  $p$  son nombre de segments.  $R$  est donc le nombre de régions délimitées par cette grille  $w$ . La fig. 2.1 présente un exemple d'une telle grille.

Soulignons de plus que nous avons utilisé l'approche développée dans [38] afin de définir sans ambiguïté une partition de l'image à l'aide d'une grille polygonale. En effet, si les noeuds sont situés sur les sites occupés par les pixels, il peut être délicat de définir à quelle région appartiennent les pixels qui sont situés sur les arrêtes ou les sommets de la grille polygonale. Une solution pragmatique et rigoureuse a été proposée dans [38]. Elle consiste à appliquer à la grille une translation de  $(1/2, 1/4)$  de pixel. De cette façon, la grille ne passe plus par les sites des pixels de l'image. Elle définit alors bien une partition de l'image sans ambiguïté. Nous discutons plus en détail cet aspect dans l'annexe A.4.

Une fois choisi le modèle paramétrique de la partition  $w$ , il faut déterminer une approximation de la plus petite longueur de code nécessaire pour décrire une partition  $w$  quelconque. Nous proposons en annexe A.1, une estimation de cette longueur. Le terme de codage de la

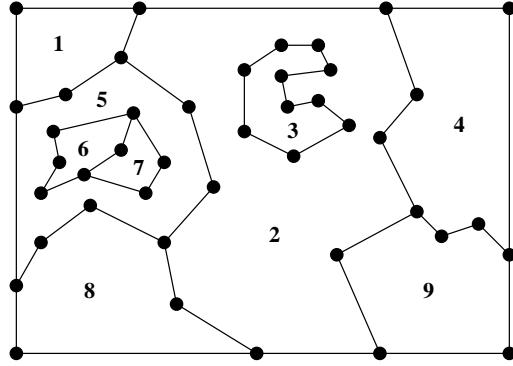


FIGURE 2.1 – *Exemple de grille polygonale.* Cette grille délimite  $R = 9$  régions et contient  $k = 43$  nœuds et  $p = 49$  segments.

grille s'écrit alors :

$$\Delta_G(w) = n (\log N + \log p) + \log p + p (2 + \log (2\hat{m}_x) + \log (2\hat{m}_y)) \quad (2.7)$$

Dans cette équation, nous avons utilisé les notations suivantes :

- $n$  est le nombre minimal de sous-graphes eulériens nécessaires pour décrire la grille<sup>3</sup>,
- $p$  est le nombre de segments dans la grille,
- $\hat{m}_x$  (respectivement  $\hat{m}_y$ ) est la longueur moyenne (mesurée en pixels) des projections horizontales (resp. verticales) des segments de la grille.

**Terme de codage des paramètres :  $\Delta_P(\boldsymbol{\theta}|w)$**

Notons  $\alpha$  la dimension du vecteur de paramètres  $\theta_r$ . Par exemple,  $\alpha = 1$  pour une loi gamma dont l'ordre est supposé connu (le seul paramètre de cette loi est alors la moyenne) et  $\alpha = 2$  pour une gaussienne (les deux paramètres sont la moyenne et la variance). Pour une région  $\Omega_r$  donnée, nous devons donc coder  $\alpha$  paramètres scalaires. Ces paramètres étant réels, il faudrait un nombre de bits infini pour les coder sans approximation. Cependant, ces  $\alpha$  paramètres ne sont connus qu'à travers l'échantillon constitué par les  $N_r$  niveaux de gris des pixels de la région  $\Omega_r$ . Cela signifie en particulier que l'on ne travaille pas avec la valeur vraie du vecteur  $\theta_r$ , mais avec une estimée  $\hat{\theta}_r$ <sup>4</sup>. Chacun de ces vecteurs  $\hat{\theta}_r$  étant estimé sur un échantillon de  $N_r$  pixels (où  $N_r$  est le nombre de pixels dans la région  $\Omega_r$ ), Rissanen propose [79] d'approcher la longueur de code associée au vecteur  $\hat{\theta}_r$  par  $\alpha \log \sqrt{N_r}$ .

La longueur de code associée au codage des paramètres pour toutes les régions s'écrit donc :

$$\Delta_P(\hat{\boldsymbol{\theta}}|w) = \sum_{r=1}^R \frac{\alpha}{2} \log N_r \quad (2.8)$$

où nous avons noté  $\hat{\boldsymbol{\theta}} = (\hat{\theta}_r)_{r \in \{1, 2, \dots, R\}}$ .

3. C'est-à-dire le nombre minimal de fois où il faut lever le crayon pour parcourir la grille sans repasser deux fois par le même segment. Nous proposons en annexe A.1 une expression très simple permettant de calculer cette grandeur (équation A.1).

4. L'estimée  $\hat{\theta}_r$  peut être calculée de diverses façons (au sens du Maximum de Vraisemblance, au sens des moments, de façon empirique, etc). Cependant, cela ne change pas les conclusions de cette partie. Le choix de l'estimateur sera abordé dans la partie suivante.

### Terme de codage des niveaux de gris : $\Delta_L(\mathbf{s}|\boldsymbol{\theta}, w)$

Une fois que le modèle est connu, c'est-à-dire à la fois la partition  $w$  et l'ensemble des  $R$  vecteurs de paramètres  $\boldsymbol{\theta} = (\theta_r)_{r \in \{1, 2, \dots, R\}}$ , nous devons déterminer la longueur du code  $\Delta_L(\mathbf{s}|\boldsymbol{\theta}, w)$  nécessaire pour décrire les niveaux de gris des pixels de l'image avec un codage entropique dans chacune des régions  $\Omega_r$ .

Calculons dans un premier temps la longueur de code  $\Delta_r(\theta_r)$  nécessaire pour coder les niveaux de gris des pixels de la région  $\Omega_r$ , connaissant le vecteur de paramètres  $\theta_r$ . Comme nous l'avons vu dans la partie 1.3.1, la quantité d'information nécessaire pour coder le niveau de gris  $s(x, y)$  d'un pixel  $(x, y)$  situé dans cette région est égale à  $-\log [P_{\theta_r}(s(x, y))]$  nats [84]. La quantité d'information nécessaire pour coder l'ensemble des niveaux de gris situés dans la région  $\Omega_r$  s'écrit donc

$$\Delta_r(\theta_r) = \sum_{(x,y) \in \Omega_r} -\log [P_{\theta_r}(s(x, y))] \quad (2.9)$$

c'est-à-dire l'opposée de la log-vraisemblance  $\mathcal{L}_e[\Omega_r | \theta_r]$  (en base  $e$ ) dans la région  $\Omega_r$ .

Pour les  $R$  régions nous obtenons ainsi

$$\Delta_L(\mathbf{s}|\boldsymbol{\theta}, w) = \sum_{r=1}^R \Delta_r(\theta_r) = \sum_{r=1}^R (-\mathcal{L}_e[\Omega_r | \theta_r]) \quad (2.10)$$

la log-vraisemblance ayant l'expression suivante :

$$\mathcal{L}_e[\Omega_r | \theta_r] = \sum_{(x,y) \in \Omega_r} \log [P_{\theta_r}(s(x, y))] \quad (2.11)$$

Remarquons que dans cette dernière équation,  $\mathcal{L}_e[\Omega_r | \theta_r]$  dépend du vecteur de paramètres  $\theta_r$  de la ddp. Dans la plupart des applications,  $\theta_r$  est inconnu et doit donc être considéré comme un paramètre de nuisance. Nous devons donc l'estimer. D'après [16], nous avons choisi de l'estimer au sens du Maximum de Vraisemblance (MV). En effet, l'utilisation du MV se justifie bien dans notre cas :

- Dans tout ce manuscrit, nous nous sommes intéressés uniquement à des ddp appartenant à la famille exponentielle (par exemple gaussienne, Gamma, Poisson, Rayleigh, Bernoulli). Il est alors connu [24, 30] que s'ils sont non biaisés, les estimateurs du MV sont à variance minimale.
- De plus, il est possible de voir dans (2.7) et (2.8), que les expressions de  $\Delta_G$  et  $\Delta_P$  ne dépendent pas de  $\theta_r$  : seul le terme  $\Delta_L$  en dépend. La valeur de  $\theta_r$  qui maximise la log-vraisemblance minimise donc la complexité stochastique. Dans notre cas, l'estimateur de  $\theta_r$  au sens du MV est donc rigoureusement celui qui minimise la complexité stochastique<sup>5</sup>.

Notons  $\widehat{\theta}_r$  l'estimée de  $\theta_r$  au sens du MV. En remplaçant  $\theta_r$  par  $\widehat{\theta}_r$ , la log-vraisemblance devient  $\mathcal{L}_e[\Omega_r | \widehat{\theta}_r]$  et on parle alors de log-vraisemblance généralisée (*profile log-likelihood* en anglais). L'équation 2.10 devient ainsi :

$$\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w) = \sum_{r=1}^R \left( -\mathcal{L}_e \left[ \Omega_r \middle| \widehat{\theta}_r \right] \right) \quad (2.12)$$

5. Soulignons que ceci n'est pas vrai dans le cas général [54].

où comme précédemment,  $\widehat{\boldsymbol{\theta}} = (\widehat{\theta}_r)_{r \in \{1, 2, \dots, R\}}$ .

Notre principale motivation étant l'étude d'images SAR, il est nécessaire de déterminer la log-vraisemblance généralisée associée à ce type d'images. Comme nous l'avons vu dans le chapitre 1.1, l'utilisation de ddp gamma d'ordre  $L$  permet de prendre en compte le phénomène de speckle caractéristique de ce type d'imagerie à l'aide d'un modèle simple [41]. Cet ordre  $L$  sera considéré comme étant un paramètre connu *a priori*, égal au nombre de vues équivalentes de l'image. La ddp de cette loi s'écrit :

$$P_{\theta_r}(x) = \frac{L^L}{\theta_r^L \Gamma(L)} x^{L-1} e^{-\frac{L}{\theta_r} x} \quad (2.13)$$

où le paramètre scalaire  $\theta_r$  de la loi est égal à l'espérance mathématique de la ddp dans la région  $\Omega_r$ .

Il est facile de montrer dans ce cas (voir annexe A.2), que le terme de codage des niveaux de gris s'écrit simplement :

$$\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w) = L \sum_{r=1}^R N_r \log \widehat{\theta}_r + K(\mathbf{s}, L) \quad (2.14)$$

où  $\widehat{\theta}_r$  est la moyenne empirique des niveaux de gris dans la région  $\Omega_r$  :

$$\widehat{\theta}_r = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} s(x, y) \quad (2.15)$$

et où  $K(\mathbf{s}, L)$  est une constante indépendante de la partition (mais dépendante de l'image  $\mathbf{s}$  et de l'ordre  $L$  de la loi gamma) qu'il est donc inutile de prendre en compte pour l'instant.

## Complexité stochastique

En combinant les trois termes  $\Delta_G(w)$ ,  $\Delta_P(\widehat{\boldsymbol{\theta}}|w)$  et  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  issus des équations (2.7), (2.8) et (2.12), nous pouvons maintenant écrire la complexité stochastique  $\Delta(\mathbf{s}, w, \widehat{\boldsymbol{\theta}})$  associée à une partition  $w$  donnée :

$$\begin{aligned} \Delta(\mathbf{s}, w, \widehat{\boldsymbol{\theta}}) &= \Delta_G(w) + \Delta_P(\widehat{\boldsymbol{\theta}}|w) + \Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w) \\ &= n(\log N + \log p) + \log p + p(2 + \log(2\widehat{m}_x) + \log(2\widehat{m}_y)) \\ &\quad + \sum_{r=1}^R \left( \frac{\alpha}{2} \log N_r - \mathcal{L}_e \left[ \Omega_r \middle| \widehat{\theta}_r \right] \right) \end{aligned} \quad (2.16)$$

Dans cette équation, plusieurs points doivent être soulignés. Tout d'abord, remarquons que nous n'avons pas calculé  $\Delta(\mathbf{s}, w, \boldsymbol{\theta})$  comme noté dans (2.6), mais  $\Delta(\mathbf{s}, w, \widehat{\boldsymbol{\theta}})$ , obtenue en remplaçant les paramètres des ddp par leurs estimés au sens du MV. Ainsi, dans (2.16), une fois fixée la grille  $w$ , tous les paramètres sont connus, puisque soit ils dépendent uniquement de  $w$  (comme  $p$ ,  $n$ ,  $\widehat{m}_x$ ,  $\widehat{m}_y$  et  $N_r$ ), soit ils sont estimés directement à partir de l'image  $\mathbf{s}$  (comme  $\widehat{\theta}_r$ ) pour chaque région  $\Omega_r$  définie par  $w$ . Cette équation ne contient donc aucun paramètre à régler de la part de l'utilisateur. En particulier, elle ne contient pas de facteurs d'échelle entre les différents termes de la complexité stochastique  $\Delta_G(w)$ ,  $\Delta_P(\widehat{\boldsymbol{\theta}}|w)$  et  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$ .

Afin de simplifier les notations, puisque pour une image  $\mathbf{s}$  donnée, la complexité stochastique ne dépend plus que de  $w$ , nous noterons dans la suite uniquement la dépendance par rapport à  $w$ , c'est-à-dire que nous écrirons

$$\Delta(w) = \Delta_G(w) + \Delta_P(w) + \Delta_L(w) \quad (2.17)$$

ou même plus simplement encore  $\Delta = \Delta_G + \Delta_P + \Delta_L$

La meilleure partition de l'image que nous recherchons est celle qui permet de minimiser la complexité stochastique. Notre but est donc de trouver la partition  $w^{MCS}$  qui minimise la complexité stochastique  $\Delta(w)$  :

$$w^{MCS} = \operatorname{argmin}_w \Delta(w) \quad (2.18)$$

## 2.2 Algorithme d'optimisation proposé

Le problème d'optimisation auquel nous sommes confrontés pour déterminer  $w$  est complexe. En effet  $w$  dépend du nombre de régions  $R$  et de leur topologie ainsi que du nombre de nœuds  $k$  et de leurs positions dans l'image. Nous nous proposons d'utiliser une approche simple consistant à estimer alternativement le nombre et la topologie des régions (à l'aide d'un processus de fusion de régions), le nombre de nœuds  $k$  (à l'aide d'un processus de suppression des nœuds) et la position des nœuds (à l'aide d'un processus de déplacement des nœuds). L'algorithme obtenu sera un algorithme de simplification : en effet, le nombre de régions et de nœuds ne pourra que diminuer, dans la mesure où ils sont estimés par des processus de fusion de régions et de suppression de nœuds. Dans la pratique, nous initialiserons notre algorithme avec une grille initiale fine, c'est-à-dire constituée d'un grand nombre de régions et de nœuds.

Nous allons maintenant présenter chacune de ces trois phases d'optimisation. Pour plus de détails, on pourra se référer à l'annexe A.3.

### 2.2.1 Fusion de régions

Le principe qui sera mis en œuvre est simple : deux régions voisines<sup>6</sup>  $\Omega_A$  et  $\Omega_B$  seront fusionnées si leur fusion<sup>7</sup> engendre une diminution de la complexité stochastique  $\Delta(w)$ .

Soit  $w$  la partition avant fusion de  $\Omega_A$  et  $\Omega_B$  et  $w'$  celle après fusion. Notons de plus  $\widehat{\theta}_{AB}$  l'estimée (au sens du MV) du vecteur de paramètres de la région  $\Omega_{AB} = \Omega_A \cup \Omega_B$ .  $\Omega_A$  et  $\Omega_B$  seront fusionnées si  $\Delta(w') - \Delta(w) < 0$ , c'est-à-dire, à l'aide de (2.16), si

$$\mathcal{L}_e \left[ \Omega_A | \widehat{\theta}_A \right] + \mathcal{L}_e \left[ \Omega_B | \widehat{\theta}_B \right] - \mathcal{L}_e \left[ \Omega_{AB} | \widehat{\theta}_{AB} \right] < S(w, w') \quad (2.19)$$

où  $S(w, w') = \Delta_G(w) + \Delta_P(w) - \Delta_G(w') - \Delta_P(w')$ .

Nous obtenons ainsi un test similaire à un rapport de vraisemblance généralisé [76] (ou GLRT pour *Generalized Likelihood Ratio Test* en langue anglaise), avec un seuil  $S$  déterminé à l'aide de la complexité stochastique.

Afin de permettre une convergence graduelle, nous combinerons dans la suite deux approches de fusion : une avec le seuil  $S(w, w') = \Delta_G(w) + \Delta_P(w) - \Delta_G(w') - \Delta_P(w')$  déduit de la complexité stochastique, et une autre à l'aide d'un test du type GLRT avec un seuil empirique  $\tilde{S}$  fixé *a priori* par l'utilisateur.

En effet, imposer dès le début une étape de fusion de régions avec un seuil  $S(w, w')$  basé sur la complexité stochastique peut s'avérer dangereux. La grille associée à la partition initiale  $w^0$  contient beaucoup plus de nœuds et de régions que la partition qui minimise la complexité stochastique, notée  $w^{MCS}$ , et à laquelle nous voulons aboutir. Cela signifie donc en particulier

6. Deux régions sont voisines si elles ont au moins un segment en commun.

7. La fusion de deux régions voisines consiste à supprimer dans la grille tous les segments communs aux deux régions.

que les seuils  $S(w^0, w^{0'})$  calculés à partir de la grille initiale  $w^0$  vont être très différents de leur valeur finale  $S(w^{MCS}, w^{MCS'})$ <sup>8</sup>. Ainsi, utiliser dès le début une fusion basée sur la complexité stochastique peut conduire à la fusion irréversible d'un trop grand nombre de régions. Nous utiliserons donc l'algorithme de fusion GLRT avec un seuil empirique  $\tilde{S}$  lors de la toute première étape, afin de réduire graduellement le nombre de régions.

Dans la suite, nous parlerons de fusion GLRT lorsque nous nous référerons à l'approche qui vient d'être décrite et qui repose sur l'utilisation d'un seuil empirique  $\tilde{S}$ , et de fusion MCS dans le cas où le seuil  $S(w, w')$  est déterminé directement à l'aide de la complexité stochastique. Soulignons d'autre part que dans tout ce manuscrit, la fusion GLRT sera utilisée uniquement à l'initialisation. De plus, comme le paramètre  $\tilde{S}$  n'est pas critique, nous avons pu faire en sorte d'utiliser toujours la même valeur de seuil, à savoir  $\tilde{S} = 3$ . Après cette première étape de fusion GLRT, toutes les étapes de fusion utilisées ultérieurement seront des fusions MCS.

Pour plus de précision sur l'algorithme, notamment sur l'ordre dans lesquelles sont fusionnées les régions, on pourra se reporter à l'annexe A.3.

### 2.2.2 Déplacement des nœuds

La grille obtenue après fusion - aussi bien d'ailleurs que la grille initiale - peut contenir des régions inhomogènes. En d'autres termes, une région de la grille peut contenir des parties appartenant à différentes régions homogènes de l'image. Au lieu de mettre en place un processus de division de régions, qui conduirait à l'apparition de très petites régions et donc à des tests statistiques peu précis, nous avons choisi de déplacer les nœuds afin que la grille puisse définir des régions homogènes. Ce processus de déplacement consiste à déplacer les nœuds de la grille afin de minimiser la complexité stochastique  $\Delta(w)$ . Le principal avantage de cette méthode est de permettre de travailler à chaque étape de l'algorithme avec des régions contenant un nombre de pixels suffisamment grand.

Ce processus, analogue à celui mis en œuvre pour les techniques de contours actifs statistiques polygonaux [16, 82], consiste simplement à considérer successivement chaque nœud et à lui affecter un certain déplacement. Ce déplacement est alors accepté s'il a entraîné une diminution de la complexité stochastique, sinon le nœud est replacé à sa position initiale (pour plus de détails, voir annexe A.3).

### 2.2.3 Suppression des nœuds

Comme la grille possède de multiples paramètres qui doivent être estimés (essentiellement les coordonnées des nœuds), le partitionnement en régions homogènes est un problème inverse mal posé. Lorsque l'on cherche à minimiser la complexité stochastique  $\Delta(w)$  avec un nombre de nœuds très important, les contours sont très chahutés. Une approche classique consiste à régulariser la solution en ajoutant au critère un terme de pénalisation de type Tikhonov [90]. Cela peut être par exemple une énergie de pénalisation élastique analogue à celle utilisée dans des approches variationnelles [18, 98, 99]. Cependant, ces techniques de régularisation présentent de nombreux inconvénients dans notre contexte. Tout d'abord, ce terme de régularisation introduit des paramètres qui ne peuvent être estimés facilement. De plus, les étapes de déplacement des nœuds et de fusion de régions reposent sur la minimisation de la complexité stochastique  $\Delta(w)$ . L'ajout d'un terme de régularisation n'empêcherait pas la grille de

8. En effet, lorsque la grille  $w$  contient notamment un grand nombre de segments,  $S(w, w')$  sera assez élevé, favorisant la fusion d'un grand nombre de régions afin de simplifier la grille.

contenir un grand nombre de noeuds et conduirait alors à une surestimation de la complexité de la grille  $\Delta_G$  et donc à une partition en régions homogènes erronée.

Nous proposons donc de mettre en œuvre un processus de suppression des noeuds inutiles, analogue à celui introduit dans [82] pour le Contour Actif Statistique Polygonal. Ce processus, très simple d'un point de vue algorithmique, consiste simplement à parcourir chaque noeud de multiplicité deux (c'est-à-dire relié à seulement deux autres noeuds) et à le supprimer si cela entraîne une diminution de la complexité stochastique  $\Delta(w)$ .

Là encore, plus de détails sont donnés en annexe A.3, notamment concernant l'ordre de suppression des noeuds.

#### 2.2.4 Procédure globale d'optimisation

Nous allons maintenant présenter la procédure d'optimisation en trois phases (fusion - déplacement - suppression) qui a été choisie. Cependant, avant d'appliquer ce processus, il est nécessaire de définir une grille initiale.

##### Grille initiale

Dans la mesure où le nombre de régions et de noeuds pourra seulement décroître au cours de la convergence, il convient de choisir une grille initiale comportant plus de régions et de noeuds que n'en contient la partition souhaitée.

Dans tous nos travaux - sauf mention du contraire - nous avons utilisé deux types de grilles régulières dont la taille peut varier : soit une grille de type *rectangulaire*, soit de type *mur de briques* (fig. 2.2).

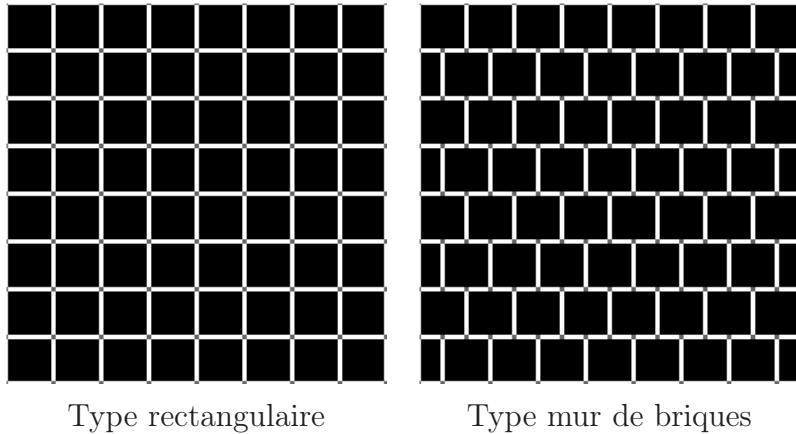


FIGURE 2.2 – Les deux types de grilles initiales utilisées dans la suite et qui délimitent toutes deux des régions de  $32 \times 32$  pixels. Soulignons que la grille en mur de briques contient deux fois plus de noeuds que celle de type rectangulaire.

Ces deux types de grilles permettent ainsi de définir des petites régions carrées qui ont toutes la même taille (à part sur les bords de l'image), et qui contiennent  $c \times c$  pixels où  $c$  est la dimension de chaque petite région carrée. Par soucis de simplification, nous parlerons de grille initiale de taille  $c$ , sous-entendant qu'il s'agit d'une grille régulière délimitant des régions de  $c \times c$  pixels.

Bien entendu, d'autres choix de grille initiale - peut-être plus adaptés - auraient pu être considérés. Nous montrerons ainsi dans le chapitre 4.3, qu'un choix plus adapté à la forme de l'objet recherché dans l'image peut conduire à une technique de type contours actifs multi-régions plus robuste que celle discutée ici puisqu'elle repose sur l'introduction de connaissances *a priori* plus importantes. Cependant, notre but étant tout d'abord de démontrer la robustesse de cet algorithme de partitionnement, ce choix d'une grille initiale générique contribue à cette démonstration. De plus, nous montrerons dans la partie 2.3.2 qu'il est possible de choisir le type et la taille de la grille initiale automatiquement au sens de la MCS.

Une fois choisie une grille initiale, l'algorithme de partitionnement en régions homogènes consiste alors à exécuter successivement les 3 phases d'optimisation précédemment décrites.

### Convergence et critère d'arrêt

Partant d'une grille initiale fine, nous allons appliquer successivement une phase de fusion de régions (de type GLRT avec un seuil  $\tilde{S} = 3$  comme nous l'avons vu dans la partie 2.2.1), suivie d'une phase de déplacement des nœuds puis d'une phase de suppression des nœuds inutiles. Afin de garantir la convergence, nous ré-appliquons alors ce cycle composé des trois phases, fusion de régions (mais de type MCS maintenant), déplacement des nœuds et suppression des nœuds inutiles.

Comme chacune de ces phases (à part la première fusion GLRT) repose sur la minimisation de la complexité stochastique, il est inutile d'introduire des critères d'arrêt *ad hoc* pour chacune des phases de convergence. En effet, chaque étape de fusion de régions MCS ou de suppression des nœuds s'arrête automatiquement dès lors que plus aucune fusion ou suppression n'entraîne de diminution de la complexité stochastique. Lorsque les fusions sont mises en œuvre en utilisant une approche GLRT, la situation est analogue, mais cette fois, c'est le seuil  $\tilde{S}$  qui fixe le nombre de régions qui seront fusionnées.

En revanche, le cas du déplacement des nœuds est légèrement différent. En effet, il n'est pas possible d'explorer en un temps raisonnable tous les déplacements possibles. Nous avons donc choisi de ne considérer pour chaque nœud que 8 déplacements parmi tous ceux possibles. Ces 8 nouvelles positions de nœud testées sont situées sur un carré de demi-côté  $a$  centré sur l'ancienne position du nœud (fig. 2.3). L'étape de déplacement des nœuds s'arrête néanmoins automatiquement dès lors que plus aucun déplacement de nœuds, parmi ceux testés, ne permet de faire diminuer la complexité stochastique. Soulignons qu'afin d'obtenir une bonne précision dans la localisation des contours, il est nécessaire de faire décroître cette valeur  $a$  (appelée amplitude des déplacements dans la suite) au cours de la convergence. La stratégie utilisée pour cette décroissance est évoquée en annexe A.3.

Pour plus de détails sur ces procédures d'optimisation, notamment en ce qui concerne les temps de calcul, on pourra se référer à l'annexe A.3.

Nous remarquons que la minimisation de la complexité stochastique repose sur une technique d'optimisation qui comporte un certain nombre de paramètres. Ces paramètres sont la taille et le type de la grille initiale, le seuil de fusion GLRT  $\tilde{S}$ , l'amplitude  $a$  des déplacements de nœuds et l'ordre dans lesquelles sont utilisées chacune des trois phases de fusion, déplacement ou suppression. Bien que la technique de minimisation contienne des paramètres *ad hoc*, comme d'ailleurs la plupart des algorithmes d'optimisation, la partition est obtenue en minimisant un critère sans paramètre. De plus, si certains de ces paramètres d'optimisation

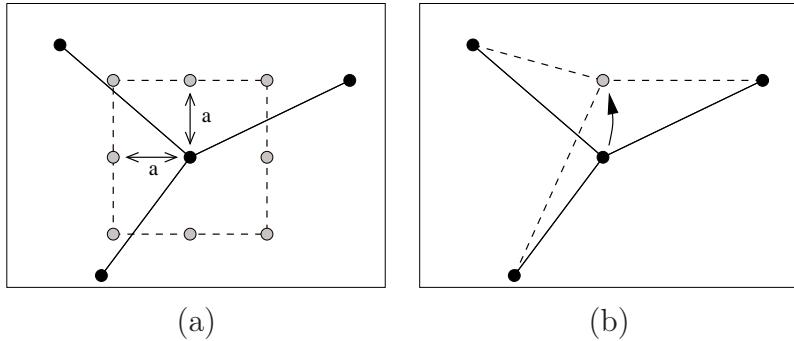


FIGURE 2.3 – Déplacement d'un nœud avec une amplitude  $a$ . (a) Les 8 déplacements de nœuds testés sont positionnés sur un carré de côté  $2a$  - (b) Un exemple de déplacement.

peuvent être fixés une fois pour toute, comme c'est le cas par exemple pour  $\tilde{S}$  (dont la valeur est toujours fixée à 3), pour l'amplitude  $a$  des déplacements (qui peuvent être fixés directement par l'algorithme en fonction de la longueur des segments déplacés - voir annexe A.3) ainsi que pour l'ordre d'utilisation des différentes phases (qui sera toujours pris dans l'ordre fusion, déplacement, suppression), il est également possible de les estimer automatiquement à l'aide de la complexité stochastique. Ce sera le cas par exemple pour la taille de la grille initiale (partie 2.3.2).

### Détail de l'algorithme sur un exemple

Nous allons maintenant présenter en détail l'algorithme de partition en régions homogènes à l'aide de l'exemple de l'image SAR de la fig. 2.4.a. Il s'agit d'une image mono-vue de taille  $105 \times 522$  pixels d'une zone agricole près de Bourges. Comme nous l'avons vu dans le chapitre 1.1, un modèle simple du bruit de speckle lié à l'éclairage cohérent du radar consiste à représenter les ddp des niveaux de gris par des lois gamma d'ordre  $L = 1$  dans le cas des images mono-vues [41]. Nous utiliserons donc cette ddp pour déterminer la longueur du terme de codage des niveaux de gris (équation 2.13).

La grille initiale considérée est une grille rectangulaire de taille 6 (fig. 2.4.b). Le nombre de régions définies par cette grille initiale étant très élevé (1566), une première phase de fusion de régions GLRT (toujours avec un seuil  $\tilde{S} = 3$ ) est donc tout d'abord mise en œuvre (fig. 2.4.c). Une phase de déplacement des nœuds permet ensuite d'estimer la position des nœuds (fig. 2.4.d). Le nombre de nœuds de la grille est ensuite réduit à l'aide de la phase de suppression des nœuds (fig. 2.4.e).

Le nombre de régions et de nœuds étant alors beaucoup plus proche de leur valeur optimale (c'est-à-dire beaucoup plus proche du nombre de régions et de nœuds contenus dans la grille  $w^{MCS}$ ), une étape de fusion de régions de type MCS peut maintenant être utilisée sans risquer la fusion - irréversible - d'un trop grand nombre de régions (fig. 2.4.f). Afin de garantir la convergence, il est souvent nécessaire d'utiliser plusieurs cycles d'optimisation, composés d'une phase de fusion MCS, une phase de déplacement des nœuds et une phase de suppression des nœuds inutiles. Après la phase de fusion MCS qui vient d'être effectuée, une nouvelle phase de déplacement des nœuds (fig. 2.4.g) puis de suppression des nœuds (fig. 2.4.h) sont donc mises en œuvre. La fig. 2.4.h constitue alors le résultat de la partition en régions homogènes.

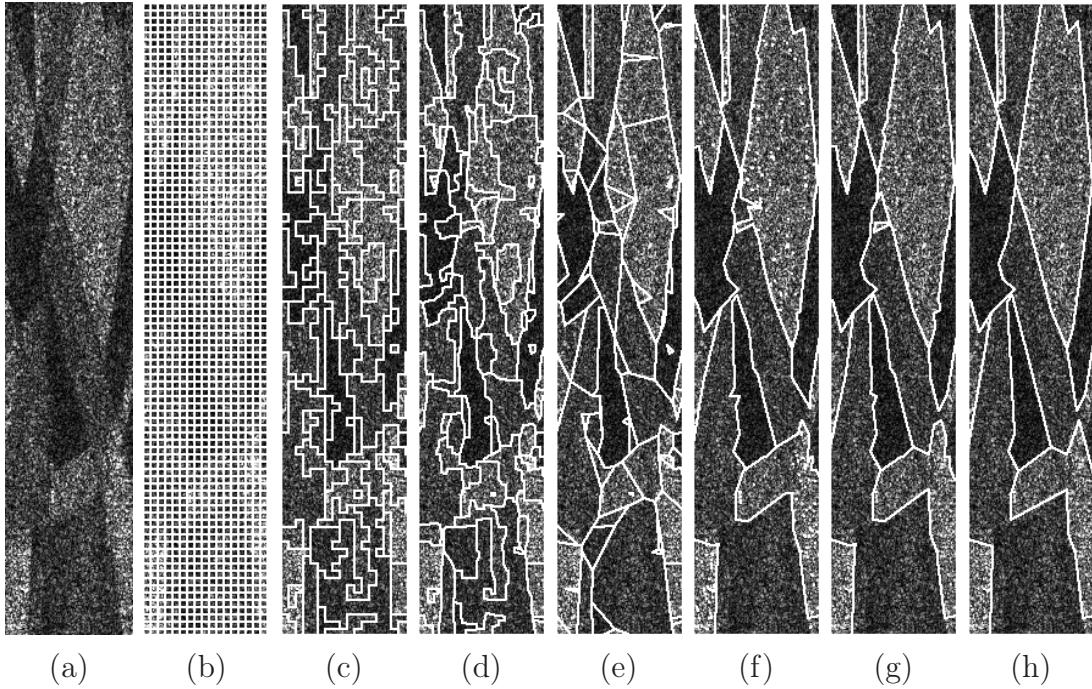


FIGURE 2.4 – Détail des différentes étapes de la partition en régions homogènes effectuée sur une image SAR mono-vue de zones agricoles près de Bourges (France) acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA). Voir texte pour plus d'explication. (a) Image SAR ( $105 \times 522$  pixels), (b) grille initiale (rectangulaire de taille 6), (c) partition obtenue à l'issue de la phase de fusion de régions GLRT ( $\tilde{S} = 3$ ), (d) de déplacement des nœuds, (e) de suppression de nœuds, (f) de fusion de régions MCS, (g) de déplacement des nœuds, (h) et d'une dernière phase de suppression de nœuds.

## 2.2.5 Algorithme rapide

Le calcul de la log-vraisemblance généralisée  $\mathcal{L}_e[\Omega_r | \hat{\theta}_r]$  dans chacune des régions  $\Omega_r$  nécessite le calcul du nombre de pixels  $N_r$  dans la région et l'estimation du vecteur de paramètres  $\theta_r$  de la ddp au sens du MV. Quand cette ddp appartient à la famille exponentielle, comme c'est le cas pour la ddp gamma utilisée dans ce chapitre et comme ce sera aussi le cas pour toutes les ddp considérées dans ce manuscrit, elle possède alors un certain nombre de propriétés :

- L'estimateur de son vecteur de paramètres  $\theta_r$  au sens du Maximum de Vraisemblance est à variance minimale [24, 30] (s'il est non biaisé).
- Cet estimateur ne dépend que de statistiques exhaustives, c'est-à-dire qu'il ne dépend de l'échantillon  $\chi_r$  qu'à travers des fonctions  $T_k(\chi_r)$  de cet échantillon.
- Ces statistiques exhaustives s'écrivent sous la forme  $T_k(\chi_r) = \sum_{x \in \chi_r} t_k(x)$ .

Comme dans notre cas  $\chi_r = \{s(x, y) | (x, y) \in \Omega_r\}$ , les statistiques exhaustives s'écrivent sous la forme :

$$T_k(\Omega_r) = \sum_{(x,y) \in \Omega_r} t_k(s(x, y)) \quad (2.20)$$

A titre d'exemple, nous pouvons remarquer que les deux statistiques  $T_k$  suivantes

$$T_k(\Omega_r) = \sum_{(x,y) \in \Omega_r} [s(x, y)]^k, \quad k \in \{0, 1\} \quad (2.21)$$

suffisent pour déterminer la log-vraisemblance généralisée d'une loi gamma d'ordre  $L$  dans chacune des régions  $\Omega_r$ . En effet,  $N_r = T_0(\Omega_r)$  et l'estimée de la moyenne  $\widehat{\theta}_r$  fournie dans (2.15) s'écrit  $\widehat{\theta}_r = T_1(\Omega_r)/T_0(\Omega_r)$ . En rajoutant le cas  $k = 2$ , les trois statistiques  $T_0$ ,  $T_1$  et  $T_2$  permettent par exemple de calculer les log-vraisemblances généralisées pour les lois gaussiennes.

Ces statistiques  $T_k$  doivent être recalculées à chaque modification de la grille, puisqu'elles interviennent dans le calcul des log-vraisemblances généralisées. Leur calcul nécessite une sommation sur toute la surface de la région considérée, ce qui est très coûteux en temps de calcul.

Afin d'accélérer l'algorithme, il est possible, en utilisant une approche analogue à celle développée dans [16] et adaptée à une grille active dans [38], de remplacer ces sommes 2D sur une région  $\Omega_r$  par des sommes sur son contour. Notons cependant que dans [38], cette approche n'avait été développée que pour une partition constituée de régions simplement connexes, c'est-à-dire des régions ne comportant aucun trou et constituées d'une seule partie. Nous détaillons dans l'annexe A.5 sa généralisation aux régions quelconques.

En utilisant cette technique d'accélération, les calculs de  $\widehat{\theta}_r$  et de  $N_r$  sont alors très rapides, sans pour autant avoir nécessité d'approximation. Il devient alors possible d'effectuer tous les calculs avec les vraies valeurs de  $\widehat{\theta}_r$  dans le critère optimisé, ce qui signifie que  $\widehat{\theta}_r$  est mis à jour à chaque modification de la grille. Pour plus de détails, à la fois sur le principe de l'algorithme rapide et sur sa mise en œuvre informatique, on pourra se référer à l'annexe A.5.

## 2.3 Exemples de partition en régions homogènes

Nous proposons maintenant d'étudier des exemples de partition d'images synthétiques en régions homogènes.

### 2.3.1 Images synthétiques

Présentons tout d'abord quelques partitions en régions homogènes sur des images perturbées par un bruit gamma. Dans tous ces exemples, nous avons pris l'ordre  $L$  du bruit gamma égal à 1, permettant ainsi de modéliser une image SAR mono-vue. Nous aurions évidemment pu prendre des valeurs de l'ordre  $L$  supérieures à 1, ce qui correspondrait alors à des images SAR multi-vues. Cependant, soulignons que plus l'ordre du bruit gamma est élevé, plus l'analyse de l'image se simplifie, ce qui justifie d'ailleurs l'utilisation d'images SAR multi-vues. Le cas  $L = 1$  est donc de ce point de vue le plus défavorable<sup>9</sup>.

Sur les fig. 2.5 et 2.6, nous avons affiché les différentes étapes de la partition en régions homogènes. L'image de la fig. 2.5 a été générée en s'inspirant de l'image SAR réelle d'une zone agricole en Ukraine qui sera étudiée dans le chapitre suivant (fig. 3.1). Quant à l'image de la fig. 2.6, elle illustre la généralisation de l'algorithme rapide aux cas des images non simplement connexes. Nous montrons enfin un dernier résultat de partition d'une image synthétique avec un bruit gamma d'ordre  $L = 1$  sur la fig. 2.7.

---

9. Évidemment, nous pourrions choisir pour  $L$  des valeurs inférieures à 1 (et donc non entières), mais dans la mesure où de telles valeurs n'interviennent pas dans le type d'images que nous avons traité, nous n'étudierons pas ce cas.

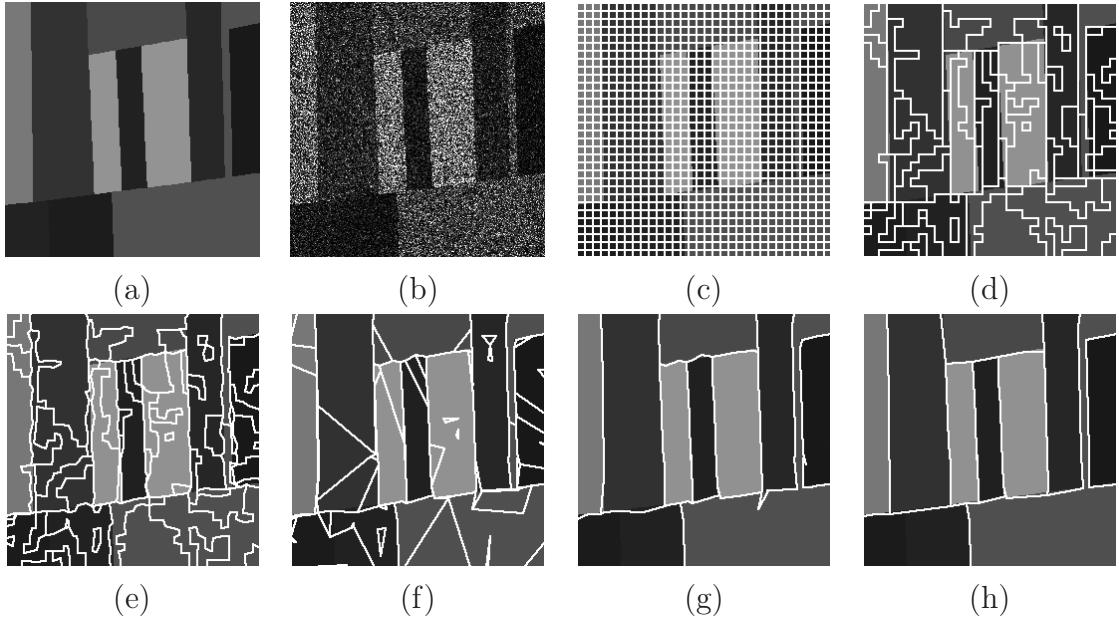


FIGURE 2.5 – Partition en régions homogènes d’une image synthétique ( $256 \times 256$  pixels) bruitée par un bruit gamma d’ordre  $L = 1$ . Cette image est analogue à une image SAR réelle étudiée dans le chapitre suivant (voir fig. 3.1). (a)-(b) image synthétique sans, puis avec, bruit - (c) grille initiale (rectangulaire de taille 8) - (d) partition obtenue après fusion de régions (GLRT, seuil  $\tilde{S} = 3$ ) - (e) après déplacement des nœuds - (f) après suppression des nœuds inutiles - (g) après une nouvelle fusion de régions (MCS) - (h) Partition obtenue après un dernier déplacement des nœuds et une dernière suppression des nœuds inutiles. Soulignons que les résultats sont superposés à l’image non bruitée, mais qu’ils ont été obtenus sur l’image bruitée (b). Temps de calcul : moins d’une seconde.

Pour chacune de ces images, le temps de calcul obtenu sur un PC standard sous Linux (Mandrake 8.2) équipé d’un processeur Xéon à 2.8 GHz, est de moins d’une seconde. Pour plus de détails sur ces temps de calculs, on pourra se référer à l’annexe A.3.

### 2.3.2 Sélection automatique de la grille initiale

Comme nous l’avons évoqué précédemment, bien que la partition en régions homogènes repose sur la minimisation d’un critère sans paramètre, l’algorithme de minimisation comporte quant à lui un certain nombre de paramètres. Un des plus importants est le choix de la grille initiale : sur les quelques exemples qui viennent d’être présentés, différents types de grilles, rectangulaire et en mur de briques, et de différentes tailles ont été utilisés.

Nous allons montrer dans cette section qu’il est possible d’estimer la taille de la grille initiale. Pour cela, partitionnons l’image de la fig. 2.5 avec des grilles initiales rectangulaires de différentes tailles. Nous montrons sur la fig. 2.8 les partitions obtenues avec des grilles initiales dont la taille varie entre 5 et 8 pixels. Nous avons également reporté sous chaque résultat, la valeur de la complexité stochastique  $\Delta(w)$  obtenue pour ces différents cas. Puisque le principe adopté consiste à minimiser la complexité stochastique, nous en déduisons que la partition obtenue avec une grille initiale rectangulaire de taille 5 est la meilleure des quatre (fig. 2.8.b). Ce résultat correspond d’ailleurs à ce que nous aurions choisi visuellement. Nous

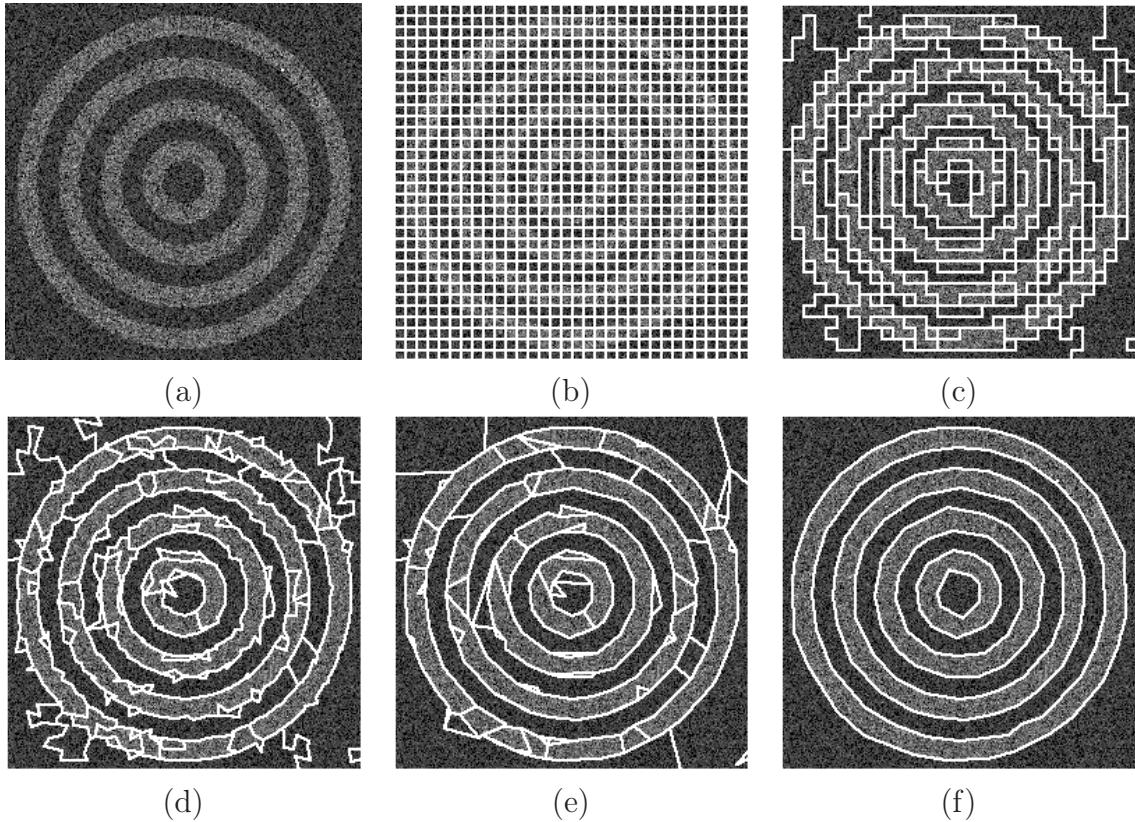


FIGURE 2.6 – Partition en régions homogènes d’une image synthétique de  $256 \times 256$  pixels constituée de régions non simplement connexes et perturbée par un bruit gamma d’ordre  $L = 1$ . (a) image - (b) grille initiale (rectangulaire de taille 8) - (c) partition obtenue après fusion de régions (GLRT - seuil  $\tilde{S} = 3$ ) - (d) après déplacement des nœuds - (e) après suppression des nœuds inutiles - (f) après une nouvelle fusion de régions (MCS). Temps de calcul : moins d’une seconde. Le contraste entre les différentes régions, c’est-à-dire le rapport des moyennes est égal à 4. Les niveaux de gris de l’image ont été modifiés pour une meilleure visualisation.

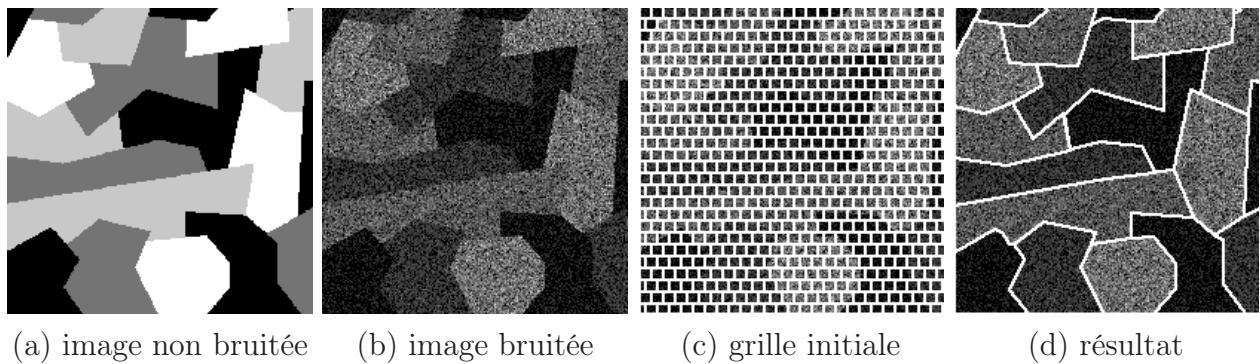


FIGURE 2.7 – Partition en régions homogènes d’une image synthétique ( $256 \times 256$  pixels) bruitée par un bruit gamma d’ordre  $L = 1$ . Grille initiale : mur de brique de taille 5. Temps de calcul : moins d’une seconde. Soulignons que les niveaux de gris ont été modifiés pour une meilleure visualisation.

pouvons opérer de la même manière avec des grilles initiales en mur de briques dont la taille varie toujours entre 5 et 8 pixels. La partition ayant la plus faible complexité stochastique est alors obtenue avec une grille en mur de briques de taille 6 (voir fig. 2.9.a).

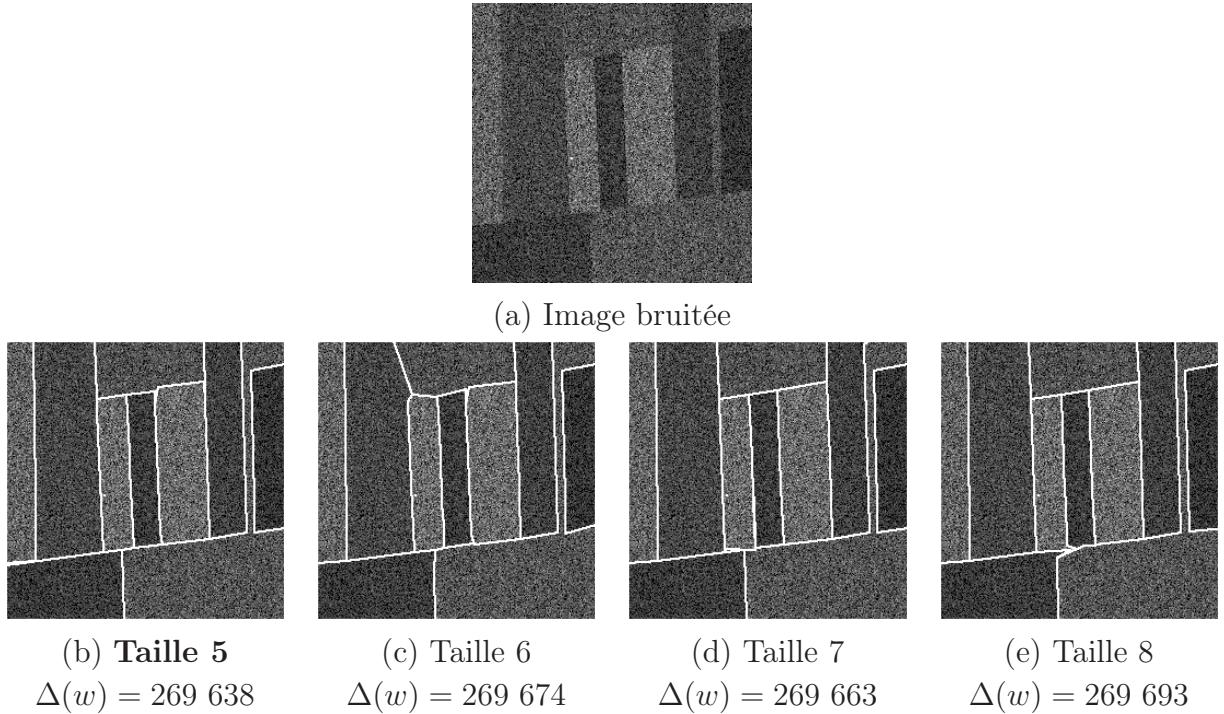


FIGURE 2.8 – Partitions d'une image synthétique obtenues à partir de grilles initiales toutes de type rectangulaire mais de tailles différentes. (a) Même image synthétique que sur la fig. 2.5. (b-e) Partition et valeur de la complexité stochastique associées à différentes tailles de grilles initiales. La meilleure grille rectangulaire initiale, au vu de la complexité stochastique, est donc une grille de taille 5 pixels.

Il est également facile de déterminer parmi ces deux types de grilles initiales, lequel est le meilleur du point de vue de la complexité stochastique. D'après les valeurs  $\Delta(w)$  fournies sur la fig. 2.9, nous en déduisons que la meilleure partition est obtenue avec une grille rectangulaire de taille 5 (voir fig. 2.9.b). Ainsi, nous avons pu sélectionner facilement la meilleure grille initiale parmi 2 types de grilles et 4 tailles possibles, c'est-à-dire en effectuant au total 8 partitions distinctes, ce qui correspond à un temps de calcul total inférieur à 8 secondes.

Les grilles initiales en mur de briques se déformant beaucoup plus facilement que les grilles initiales rectangulaires, nous aurons donc tendance, dans la pratique, à utiliser des grilles initiales de type mur de briques.

Dans ce chapitre, nous nous sommes servis de la complexité stochastique comme d'un critère de qualité, afin de choisir la meilleure partition parmi plusieurs. Nous aurions pu le faire avec un critère du type *Nombre de Pixels Mal Classés*, mais contrairement à la complexité stochastique, l'obtention du nombre de pixels mal classés nécessite la connaissance d'une vérité terrain.

Toute la question est maintenant de savoir si la complexité stochastique constitue un bon critère de qualité, ce que nous allons essayer de démontrer dans le chapitre suivant à travers divers exemples d'images SAR réelles.

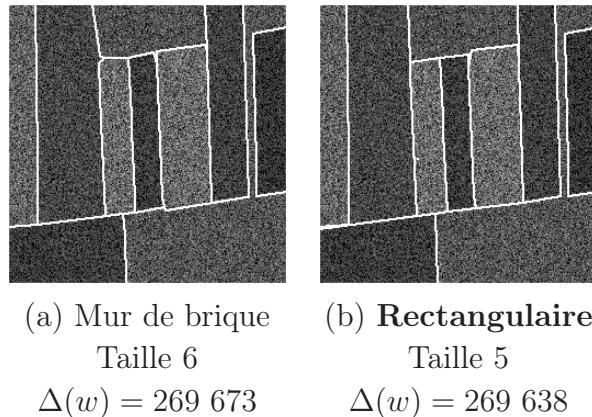


FIGURE 2.9 – Partitions d’une image synthétique obtenues à partir de grilles initiales de types différents. (a) Meilleure partition obtenue lorsque l’on impose comme grille initiale une grille en mur de briques dont seule la taille peut varier. (b) Idem, mais en imposant cette fois-ci une grille rectangulaire. Après examen des complexités stochastique associées, la meilleure grille initiale est donc une grille rectangulaire de taille 5.

## 2.4 Conclusion

Dans ce chapitre, nous avons présenté une méthode de partition d’images par grille active qui repose sur la minimisation de la complexité stochastique. Le critère obtenu ne contient aucun paramètre à régler de la part de l’utilisateur et a été déterminé à partir d’une modélisation du phénomène de speckle à l’aide de lois gamma. L’utilisation de ce modèle simple a permis de mettre en place un algorithme rapide de minimisation de la complexité stochastique, basé sur des fusions de régions, des déplacements et des suppressions de noeuds.

Après avoir montré quelques résultats de partitions intéressants sur des images synthétiques, nous devons maintenant aborder le problème de la partition d’images SAR réelles, afin de justifier l’intérêt pratique de la méthode proposée.

# Chapitre 3

## Applications à l'imagerie SAR

### Sommaire

---

<b>3.1</b>	<b>Segmentation de parcelles agricoles</b>	<b>42</b>
<b>3.2</b>	<b>Estimation de l'ordre du speckle</b>	<b>47</b>
3.2.1	Influence du choix de l'ordre	47
3.2.2	Partition en régions homogènes sans connaissance <i>a priori</i> de l'ordre du speckle	48
<b>3.3</b>	<b>Segmentation de nappes de pétrole</b>	<b>53</b>
3.3.1	Etape 1 : partition en régions homogènes	53
3.3.2	Etape 2 : classification par seuillage	54
3.3.3	Amélioration de la robustesse et discussion	58
<b>3.4</b>	<b>Conclusion</b>	<b>60</b>

---

Dans ce chapitre, nous allons discuter quelques exemples d'application à l'imagerie SAR de l'algorithme de partition en régions homogènes qui vient d'être présenté. Nous allons ainsi nous intéresser dans un premier temps à la segmentation de parcelles agricoles. Nous analyserons dans une deuxième partie la possibilité d'estimer automatiquement l'ordre du bruit de speckle présent dans ces images. Nous terminerons ce chapitre en étudiant la segmentation de nappes de pétrole causées par le naufrage du Prestige au large des côtes espagnoles en novembre 2002.

### 3.1 Segmentation de parcelles agricoles

Intéressons-nous tout d'abord au cas des images SAR de parcelles agricoles. Ces images sont constituées de zones de réflectivité constante perturbées par un bruit de speckle. Elles semblent donc être en bonne adéquation avec le modèle d'image que nous avons considéré.

Nous présentons sur les fig. 3.1, 3.2 et 3.3 les différentes étapes de la partition de telles images SAR de parcelles agricoles. Nous montrons également sur la fig. 3.4 le résultat de la partition en régions homogènes d'une image SAR ainsi que l'image moyenne associée, c'est-à-dire l'image dans laquelle les niveaux de gris dans chaque région sont remplacés par la valeur moyenne estimée.

Enfin, la fig. 3.5 présente la partition d'une image SAR plus complexe qui comporte à la fois des zones agricoles, des lacs et des rivières (c'est-à-dire des régions filamentaires).

Nous observons sur ces quelques résultats que l'algorithme de partition fournit des résultats intéressants : la partition en régions homogènes correspond à la segmentation de l'image attendue par l'utilisateur, c'est-à-dire qu'elle délimite correctement les différentes parcelles.

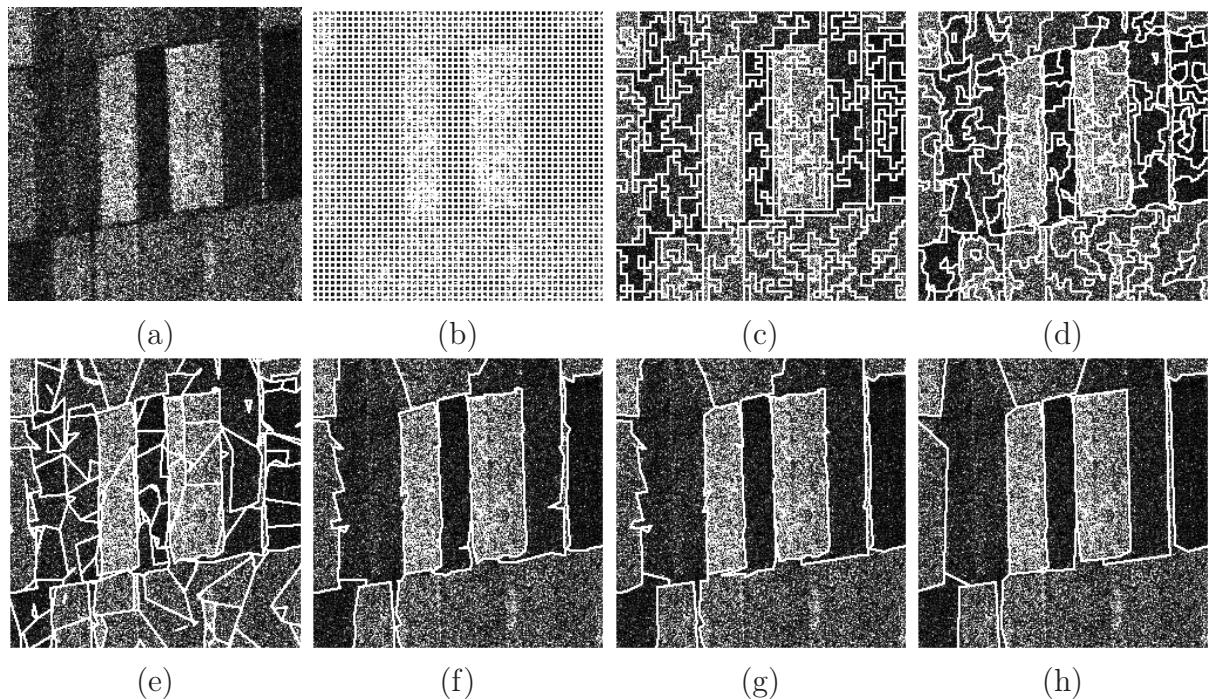


FIGURE 3.1 – Partition d'une image SAR mono-vue ( $250 \times 250$  pixels) d'une zone agricole en Ukraine acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA). Ordre de la loi gamma :  $L = 1$ . (a) image SAR - (b) grille initiale (rectangulaire de taille 5) - (c) partition obtenue après fusion de régions (GLRT avec seuil  $\tilde{S} = 3$ ) - (d) après déplacement des nœuds - (e) après suppression des nœuds - (f) après fusion de régions (MCS) - (g) après déplacement des nœuds - (h) résultat de partition après une dernière suppression des nœuds. Temps de calcul : moins d'une seconde.

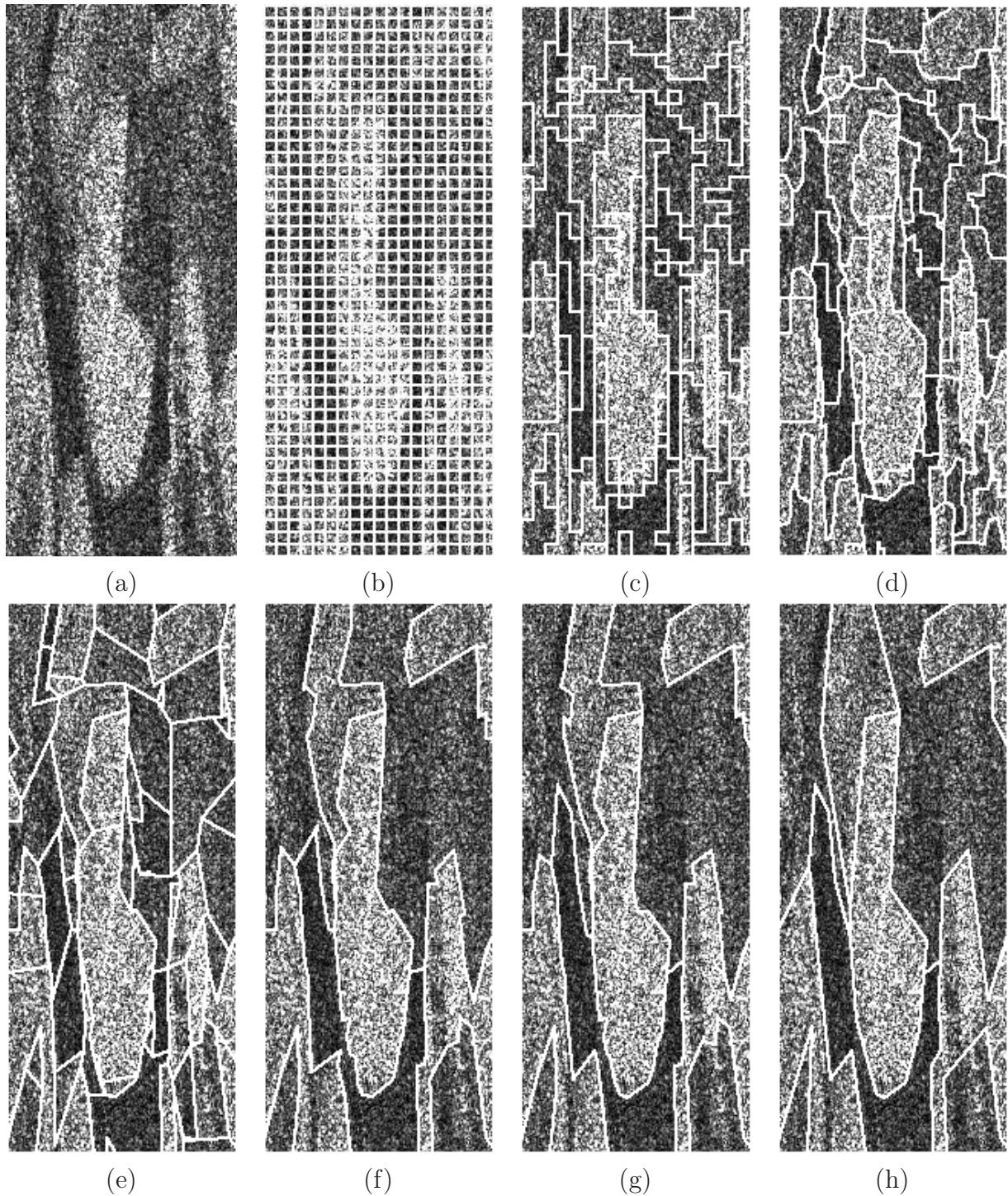


FIGURE 3.2 – Partition d'une image SAR mono-vue ( $150 \times 350$  pixels) d'une zone agricole près de Bourges (France) acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA). Ordre de la loi gamma :  $L = 1$ . (a) image SAR - (b) grille initiale (rectangulaire de taille 8) - (c) partition obtenue après fusion de régions (GLRT avec seuil  $\tilde{S} = 3$ ) - (d) après déplacement des nœuds - (e) après suppression des nœuds - (f) après fusion de régions (MCS) - (g) après déplacement des nœuds - (h) résultat après une dernière suppression des nœuds. Temps de calcul : 700ms.

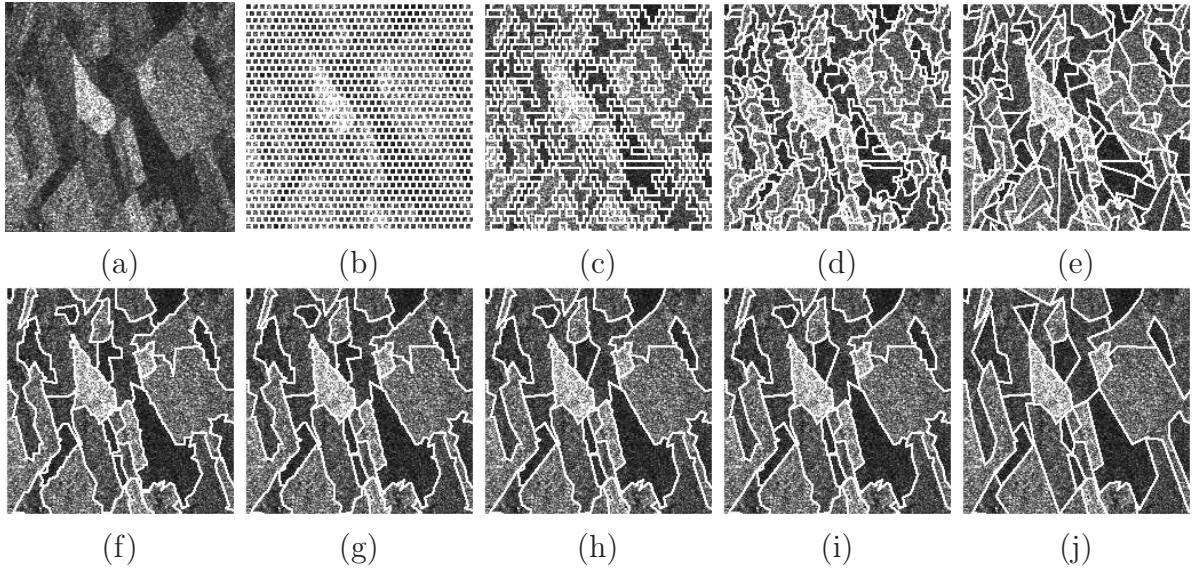


FIGURE 3.3 – Partition d'une image SAR ( $201 \times 201$  pixels) d'une zone agricole près de Bourges (France) acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA). Ordre de la loi gamma :  $L = 2$ . (a) Image SAR. (b) Grille initiale (mur de briques de taille 6), (c) partition obtenue après fusion de régions (GLRT,  $\tilde{S} = 3$ ), (d) après déplacement des nœuds, (e) après suppression des nœuds, (f) après fusion de régions (MCS), (g) après déplacement des nœuds, (h) après suppression des nœuds, (i) après fusion de régions (MCS), (j) résultat de partition après une dernière phase de déplacement puis de suppression des nœuds. Temps de calcul : 1.2 secondes. Les niveaux de gris ont été rehaussés pour une meilleure visualisation.

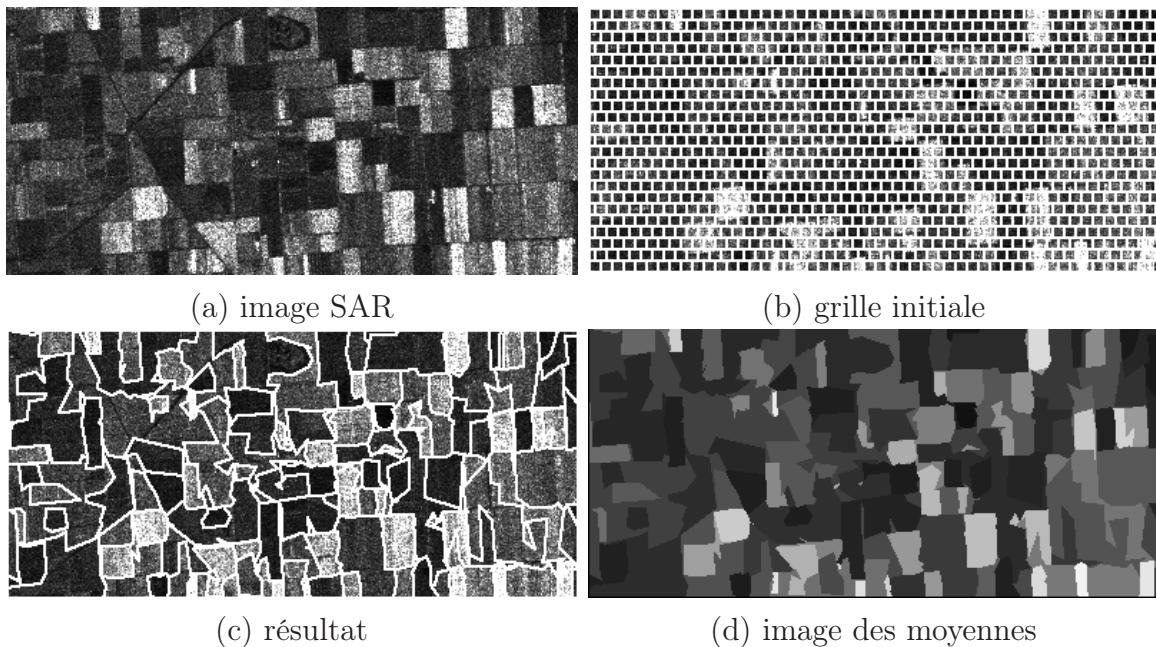


FIGURE 3.4 – Partition d'une image SAR ( $400 \times 187$  pixels) d'une zone agricole en Ukraine acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA). Grille initiale : mur de briques de taille 8. Ordre de la loi gamma :  $L = 7$ . Temps de calcul : 2.4 secondes. Les niveaux de gris ont été rehaussés pour une meilleure visualisation. Sur la fig. (d), les niveaux de gris des pixels ont été remplacés par les valeurs moyennes des régions.

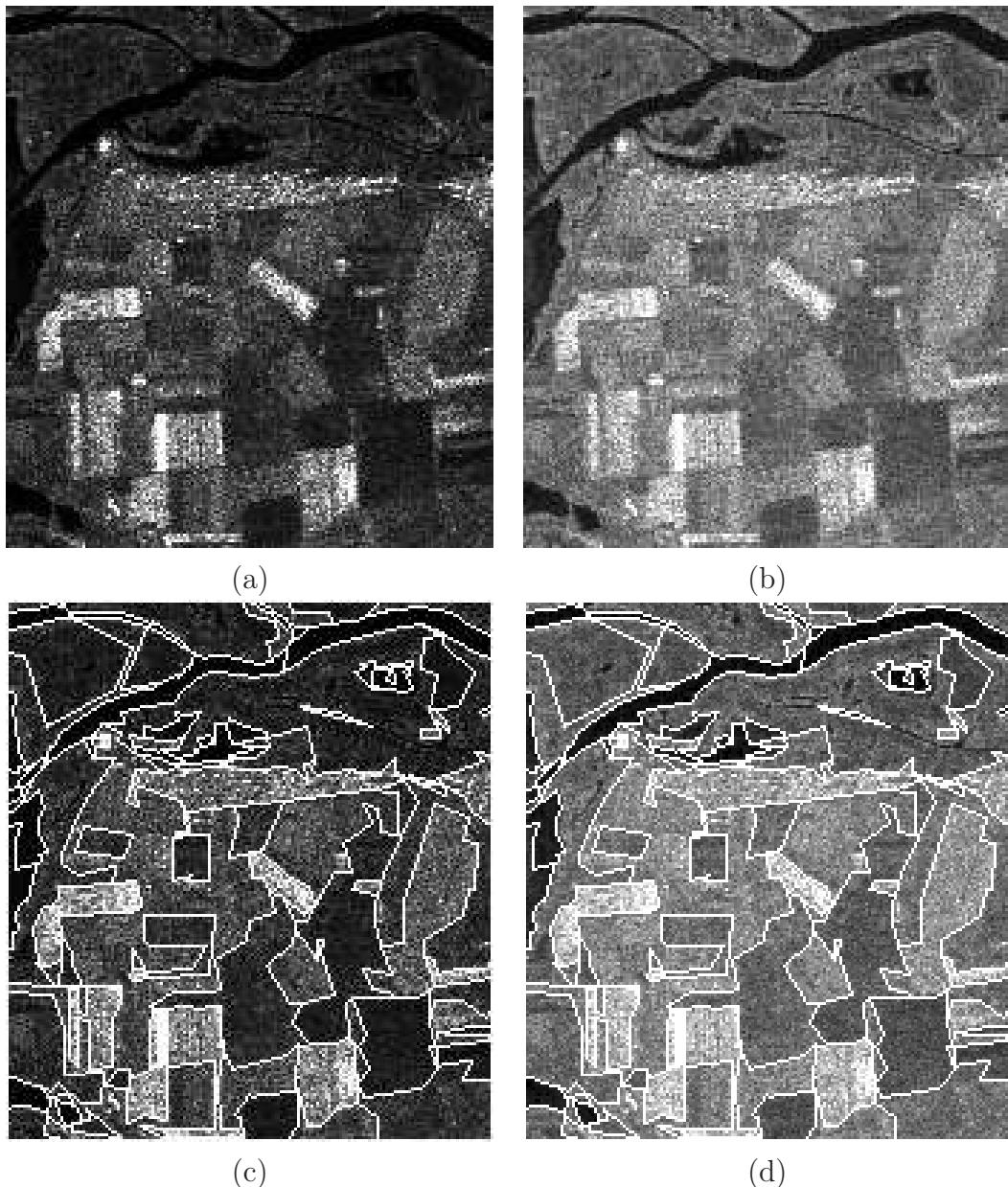


FIGURE 3.5 – Partition d'une image SAR mono-vue ( $171 \times 190$  pixels) acquise par le satellite ERS-1 (fournie par le CNES et distribuée par l'ESA) et composée de zones agricoles, de lacs et de rivières. Ordre de la loi gamma :  $L = 1$ . Ligne 1 : image SAR originale (a) et après modification des niveaux de gris pour une meilleure visualisation (b). Ligne 2 : résultat de la partition affichée sur les images de la première ligne. Afin de pouvoir détecter les rivières très fines, une grille initiale en mur de briques de taille 3 a été utilisée.

## 3.2 Estimation de l'ordre du speckle

Dans tous les exemples de la partie précédente, nous avons supposé que l'ordre  $L$  du speckle était connu. Nous allons tout d'abord étudier l'influence d'un choix d'ordre erroné sur la partition obtenue. Nous montrerons ensuite qu'il est possible simultanément de partitionner l'image en régions homogènes et d'estimer cet ordre.

### 3.2.1 Influence du choix de l'ordre

Afin d'analyser l'influence du choix de l'ordre  $L$  de la loi gamma, nous avons générée une image synthétique ( $128 \times 128$  pixels) perturbée par un bruit de speckle d'ordre 3 (voir fig. 3.6.a). Nous avons alors partitionné cette image en régions homogènes en utilisant la complexité stochastique adaptée à un bruit gamma d'ordre  $L$ , et ceci pour 3 valeurs différentes de  $L$  :  $L \in \{1, 3, 10\}$ . Les résultats sont présentés sur la fig. 3.6.

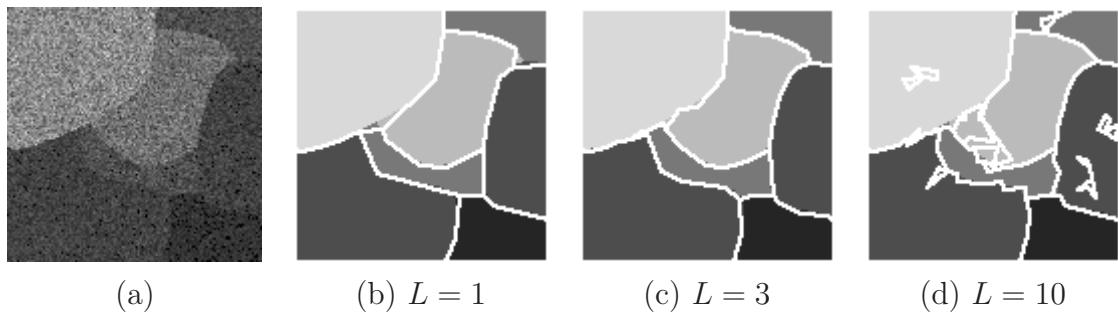


FIGURE 3.6 – *Influence de l'ordre  $L$  de la loi gamma : partition de l'image (a) dont les niveaux de gris sont perturbés par un bruit gamma d'ordre 3. Résultats obtenus lorsque l'ordre  $L$  intervenant dans la log-vraisemblance est fixé à 1 (b), 3 (c) et 10 (d). Les résultats sont affichés sur l'image non bruitée.*

Nous pouvons remarquer sur la fig. 3.6.d que lorsque l'on impose  $L = 10$  dans le calcul de la complexité stochastique (c'est-à-dire une valeur de  $L$  supérieure à l'ordre du speckle dans l'image), le nombre de noeuds et de régions de la grille est surestimé. En revanche, pour  $L = 1$  (c'est-à-dire une valeur inférieure à l'ordre du speckle de l'image) cela entraîne la suppression d'un trop grand nombre de noeuds (voir fig. 3.6.b). Nous pouvons ainsi constater que le meilleur résultat est obtenu lorsque l'ordre  $L$  utilisé dans la complexité stochastique est égal à l'ordre du speckle dans l'image, c'est-à-dire  $L = 3$  (fig. 3.6.c).

Ces résultats s'interprètent facilement en analysant la log-vraisemblance généralisée d'une ddp gamma (équation 2.14). En effet, l'ordre  $L$  apparaît comme une constante multiplicative devant le terme de log-vraisemblance et permet donc de pondérer l'influence du terme d'attache aux données  $\Delta_L$  par rapport aux deux autres termes de régularisation de la grille que sont  $\Delta_G$  et  $\Delta_P$ . Ainsi, la régularisation de la grille est plus importante pour les faibles valeurs de  $L$  que pour les fortes<sup>10</sup>.

La connaissance de l'ordre du speckle présent dans l'image est donc primordiale pour pouvoir appliquer de manière efficace l'algorithme de partition en régions homogènes.

10. Ceci est d'ailleurs en accord avec le sens physique puisqu'un phénomène de speckle d'ordre faible sera très bruité, ce qui conduit à avoir une faible confiance dans les données et nécessite une régularisation importante.

### 3.2.2 Partition en régions homogènes sans connaissance *a priori* de l'ordre du speckle

L'ordre de la loi gamma utilisée pour modéliser le speckle peut ne pas être connu. Nous allons montrer dans cette partie qu'il est possible de l'estimer à l'aide de la complexité stochastique. Dans ce cas, l'ordre  $L$  du speckle est considéré comme un paramètre de nuisance, identique pour toutes les régions de l'image, le paramètre d'intérêt étant toujours la partition  $w$ .

Une première approche envisageable consiste à estimer l'ordre  $L$  au sens du MV, comme cela est fait pour la moyenne. En effet, bien qu'aucune expression rigoureuse de l'estimée de  $L$  au sens du MV ne puisse être obtenue, il est possible de considérer des approximations de cette estimée ou d'utiliser une approche itérative.

Une autre solution possible consiste à déterminer la complexité stochastique pour différentes valeurs de  $L$  et de choisir celle qui la minimise. Nous nous proposons d'étudier cette méthode.

La technique d'optimisation de la grille étant exclusivement une procédure de simplification<sup>11</sup>, nous proposons dans un premier temps de partitionner l'image avec un ordre  $L$  élevé (typiquement 10 dans les exemples suivants), ce qui conduit à une partition ayant été peu régularisée et contenant donc un grand nombre de régions et de noeuds. Une fois obtenue la partition pour cet ordre  $L$ , la grille associée peut alors être utilisée comme grille initiale pour une nouvelle partition en régions homogènes, mais avec cette fois-ci un ordre  $L - \delta L$  plus faible ( $\delta L$  sera pris égal à 1 dans la suite). Il suffit alors de réitérer ce processus en diminuant l'ordre  $L$  jusqu'à une valeur limite, typiquement 1. L'ordre optimal  $\hat{L}$  au sens de la complexité stochastique est celui qui minimise ce critère. Le résultat de la partition en régions homogènes est alors la partition associée à cet ordre  $\hat{L}$ . Il est cependant important de remarquer que pour comparer les complexités stochastiques à des ordres différents, il est nécessaire de prendre en compte le terme  $K(\mathbf{s}, L)$  dans l'équation 2.14. Ce terme additif s'écrit (équation A.10 de l'annexe A.2) :

$$K(\mathbf{s}, L) = N(L \log L - \log \Gamma(L) - L) + (L - 1) \sum_{Image} \log(s(x, y)) \quad (3.1)$$

Cet algorithme est présenté plus en détail dans le tableau 3.1 et est illustré sur la fig. 3.7.

Nous présentons ensuite deux résultats de partition en régions homogènes, le premier sur une image synthétique de type parcellaire agricole perturbée par un bruit gamma d'ordre  $L = 2$  (fig. 3.8) et le deuxième sur une image SAR réelle de parcelles agricoles en Ukraine (fig. 3.9).

Ces exemples montrent donc que l'algorithme présenté ici permet simultanément d'estimer l'ordre du speckle efficacement et de fournir une bonne partition de l'image, sans pour autant trop augmenter le temps de calcul. En effet, typiquement, ce temps de calcul n'est multiplié que par un facteur 2 ou 3 par rapport à l'algorithme de partition supposant l'ordre connu.

---

11. En effet, le nombre de noeuds et de régions ne peut que diminuer tout au long de la convergence.

- 
- ◊ Générer une grille initiale
  - ◊  $L = L_{max}$
  - ◊ Tant que  $L \geq L_{min}$  :
    - Partitionner l'image en régions homogènes
      - à partir de la grille initiale
      - avec une complexité stochastique adaptée à une loi gamma d'ordre  $L$
    - Prendre la grille obtenue comme nouvelle grille initiale
    - $L \leftarrow L - \delta L$
  - ◊ Le minimum de la complexité stochastique fournit simultanément :
    - une estimation de l'ordre  $\hat{L}$
    - la partition en régions homogènes associée
- 

TABLE 3.1 – *Algorithme de partition en régions homogènes avec estimation simultanée de l'ordre du speckle. Dans tous les exemples présentés dans cette partie, nous avons pris  $L_{max} = 10$ ,  $L_{min} = 1$  et  $\delta L = 1$ .*

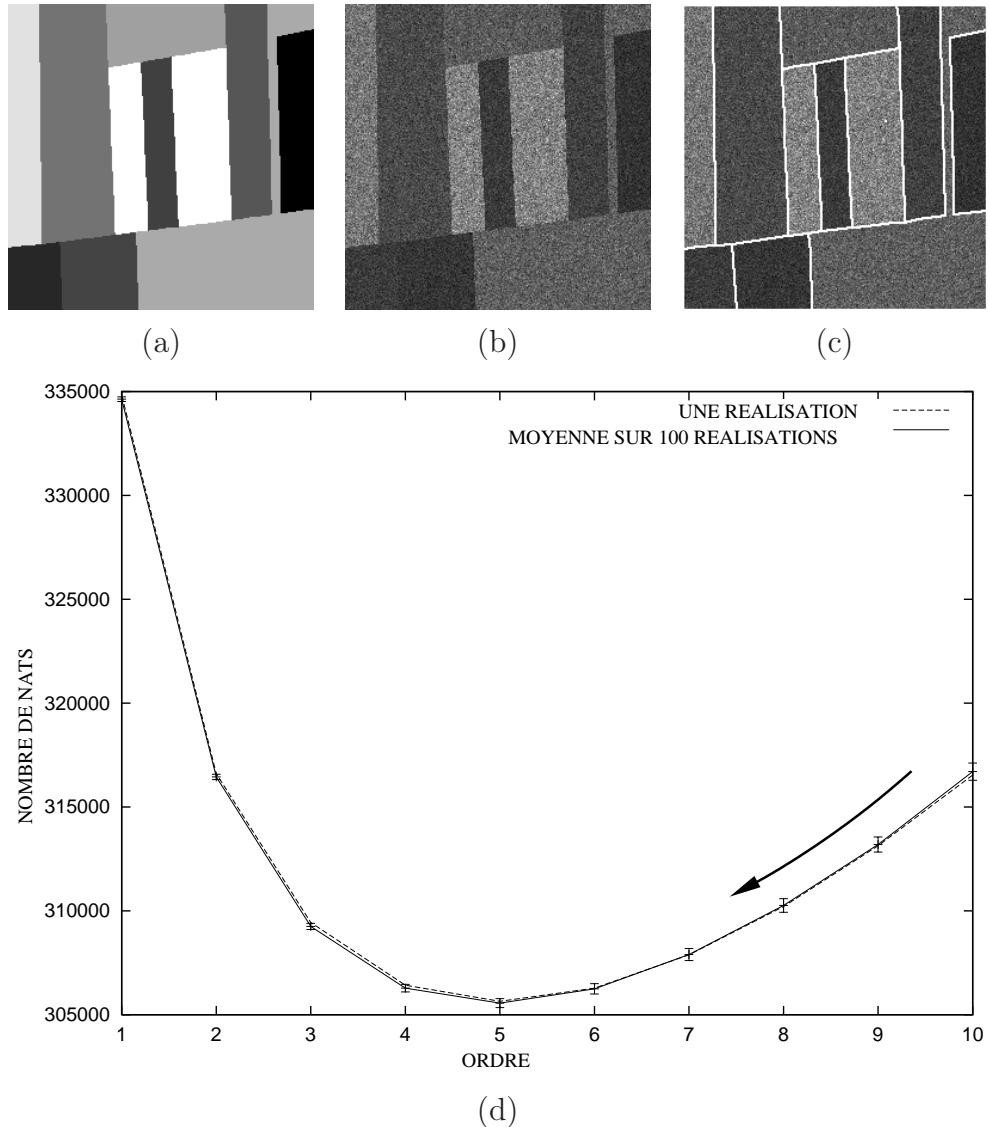


FIGURE 3.7 – *Estimation de l'ordre du speckle : évolution de la complexité stochastique.* On génère à partir de l'image synthétique de  $256 \times 256$  pixels (a) une image perturbée par un bruit de speckle d'ordre 5 (b). L'algorithme utilisé est décrit dans le tableau 3.1 et consiste à partitionner l'image avec un ordre fixé successivement à  $L = 10, L = 9, \dots, L = 1$ , en utilisant à chaque fois la partition obtenue précédemment comme grille initiale. La courbe (d) montre l'évolution de la complexité stochastique lorsque  $L$  diminue (trait plein : évolution pour une réalisation - trait pointillé : moyenne sur 100 réalisations). Nous obtenons bien un minimum pour  $L = 5$ , qui correspond à la valeur exacte de l'ordre. La partition correspondante est affichée sur la figure (c).

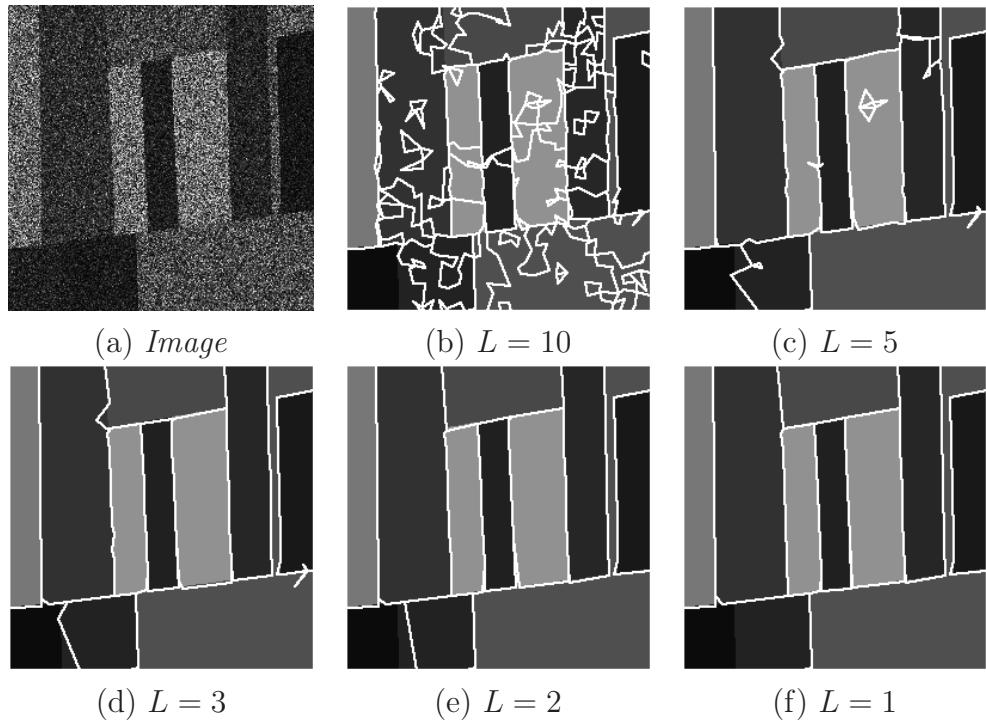


FIGURE 3.8 – *Estimation de l'ordre du speckle* : résultat de la partition. La partition est effectuée sur la même image synthétique que sur la fig. 3.7.a mais perturbée cette fois-ci par un bruit de speckle d'ordre 2 (a). L'algorithme utilisé pour partitionner l'image est décrit dans le tableau 3.1, sachant que l'ordre décroît de  $L = 10$  à  $L = 1$  par pas de 1. Les fig. (b)-(c)-(d)-(e)-(f) montrent les résultats obtenus pour différents ordres. Le minimum de la complexité stochastique est atteint pour  $L = 2$ , correspondant à la fois à l'ordre vrai du speckle mais aussi à la meilleure partition puisque les 2 petites régions sur le coin gauche présentant un très faible contraste (1.2) sont fusionnées lorsque  $L$  est fixé à 1. Grille initiale : rectangulaire de taille 8.

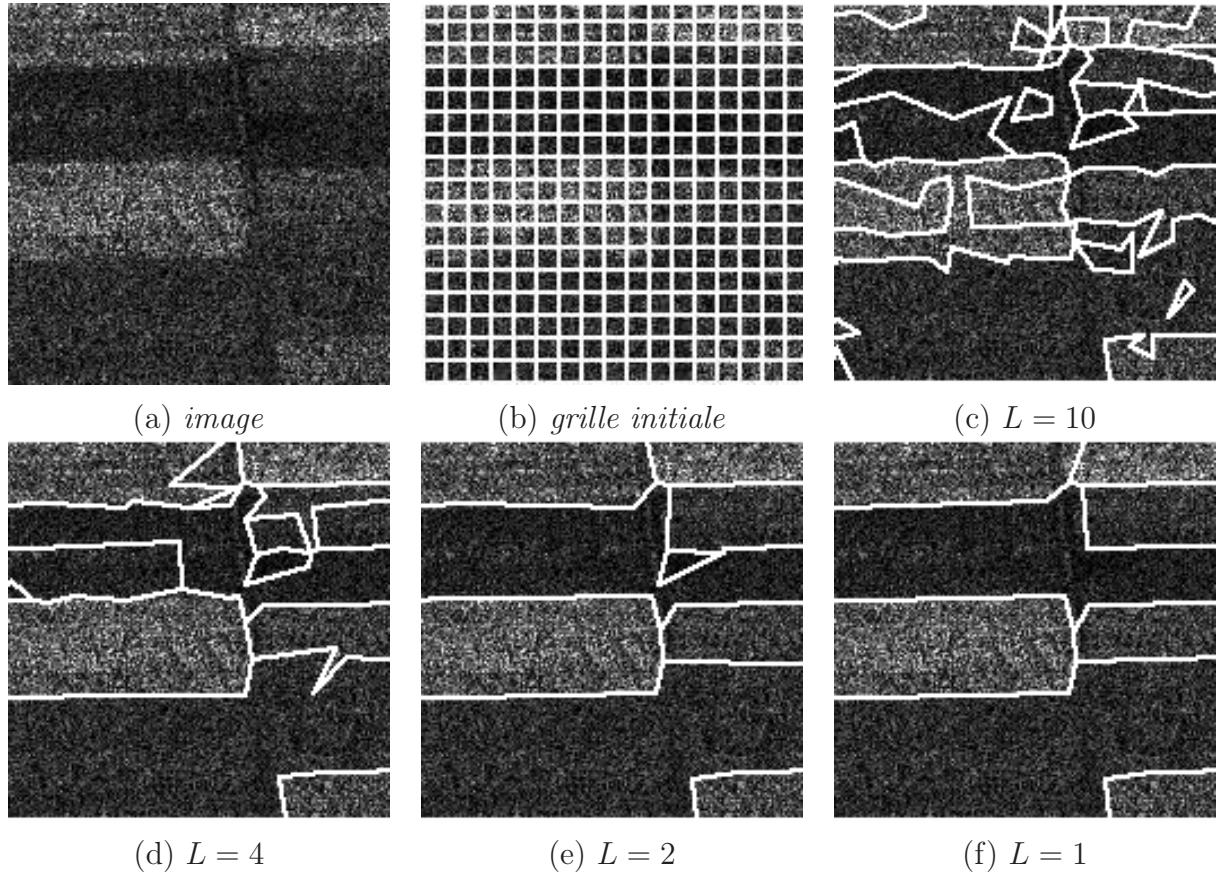


FIGURE 3.9 – Partition d'un extrait ( $170 \times 170$  pixels) d'une image SAR mono-vue (a) d'une zone agricole d'Ukraine obtenue par le satellite ERS-1 (image distribuée par l'ESA et fournie par le CNES). La grille initiale est montrée en (b). En utilisant l'algorithme décrit dans le tableau 3.1, nous obtenons un ordre estimé  $\hat{L}$  égal à 1, ce qui était prévisible, l'image étant mono-vue. Les fig. (b,c,d,e,f) montrent les partitions associées à différents ordres obtenus au cours de l'algorithme, la meilleure étant bien sûr celle obtenue en fixant  $L = 1$ , c'est-à-dire (f).

### 3.3 Segmentation de nappes de pétrole

Les marées noires causées par le naufrage de pétroliers (comme l'Exxon Valdez, l'Erica ou le Prestige), bien que spectaculaires et très médiatisées, sont responsables de seulement 5% des rejets d'hydrocarbures dans les océans : 95% proviennent de dégazages illégaux. Afin de pouvoir observer de tels événements par tout type de temps, le potentiel offert par les systèmes d'imagerie SAR satellitaires a été démontré [1, 75]. La détection automatique de nappes de pétrole dans des images SAR a donc suscité un grand intérêt [86, 22], notamment en ce qui concerne la classification des zones sombres susceptibles de constituer de telles nappes. Cependant, la plupart de ces approches reposent sur une pré-segmentation supervisée - voire manuelle - de ces zones d'intérêt. Le développement d'algorithmes de segmentation rapides et automatiques est donc nécessaire.

Dans cette partie, nous allons nous intéresser à la segmentation des zones sombres dans des images SAR. Notre but est donc simplement de fournir une segmentation des nappes de pétrole potentielles contenues dans l'image, leur identification - qui permettra de conclure s'il s'agit bien de nappes de pétrole - se faisant à l'aide d'autres techniques que nous n'étudierons pas ici. L'étape de segmentation est donc cruciale pour la réalisation d'un système complet de détection et d'analyse de nappes de pétrole requérant une intervention humaine minimale.

La méthode de segmentation des zones sombres dans une image SAR que nous proposons dans cette partie se divise en deux étapes : une première qui repose sur une partition en régions homogènes et une seconde qui consiste en une classification basée sur une méthode de seuillage avec estimation automatique des seuils à l'aide de la complexité stochastique. Cette méthode, fruit d'une collaboration avec la société Starlab (Barcelone), sera illustrée sur des images SAR (fournies par l'ESA) du satellite ASAR-ENVISAT (Wide Swath Mode) de la marée noire causée par le naufrage du Prestige à proximité des côtes de Galice (Espagne) en novembre 2002. Pour information, nous présentons sur la fig. 3.10 l'image SAR complète d'où sont extraites les différentes images utilisées.

#### 3.3.1 Etape 1 : partition en régions homogènes

Cette première étape peut être vue comme une étape de pré-segmentation qui consiste à partitionner l'image SAR en régions homogènes. Nous présentons sur la fig. 3.11.a une image SAR ( $598 \times 875$  pixels) de la marée noire causée par le Prestige. Pour cette image, l'ordre de la loi Gamma a été estimé en utilisant l'algorithme décrit dans la partie précédente. Nous avons obtenu un ordre de  $L = 50$ , comme c'est le cas d'ailleurs pour les 2 autres images SAR de marées noires étudiées dans cette partie. La fig. 3.11.b montre la partition en régions homogènes obtenue à partir d'une grille initiale en mur de briques de taille 8 pixels (c'est-à-dire plus de 8 000 régions et 16 000 nœuds). Le temps de calcul est de 6.7 secondes sur un PC standard sous Linux (Mandrake 8.2) avec un processeur Xeon cadencé à 2.8 GHz.

Nous pouvons voir sur cette figure que la partition en régions homogènes n'a pas permis d'aboutir à une segmentation satisfaisante, dans la mesure où les nappes de pétrole potentielles sont morcelées en plusieurs petites régions. Ce résultat montre simplement que les zones sombres, mais aussi les zones de mer, ne constituent pas des régions ayant des réflectivités homogènes. Afin de regrouper les différentes régions homogènes appartenant à la même classe (pétrole ou mer), nous proposons d'utiliser une technique simple qui repose sur un seuillage ainsi que sur la minimisation de la complexité stochastique.

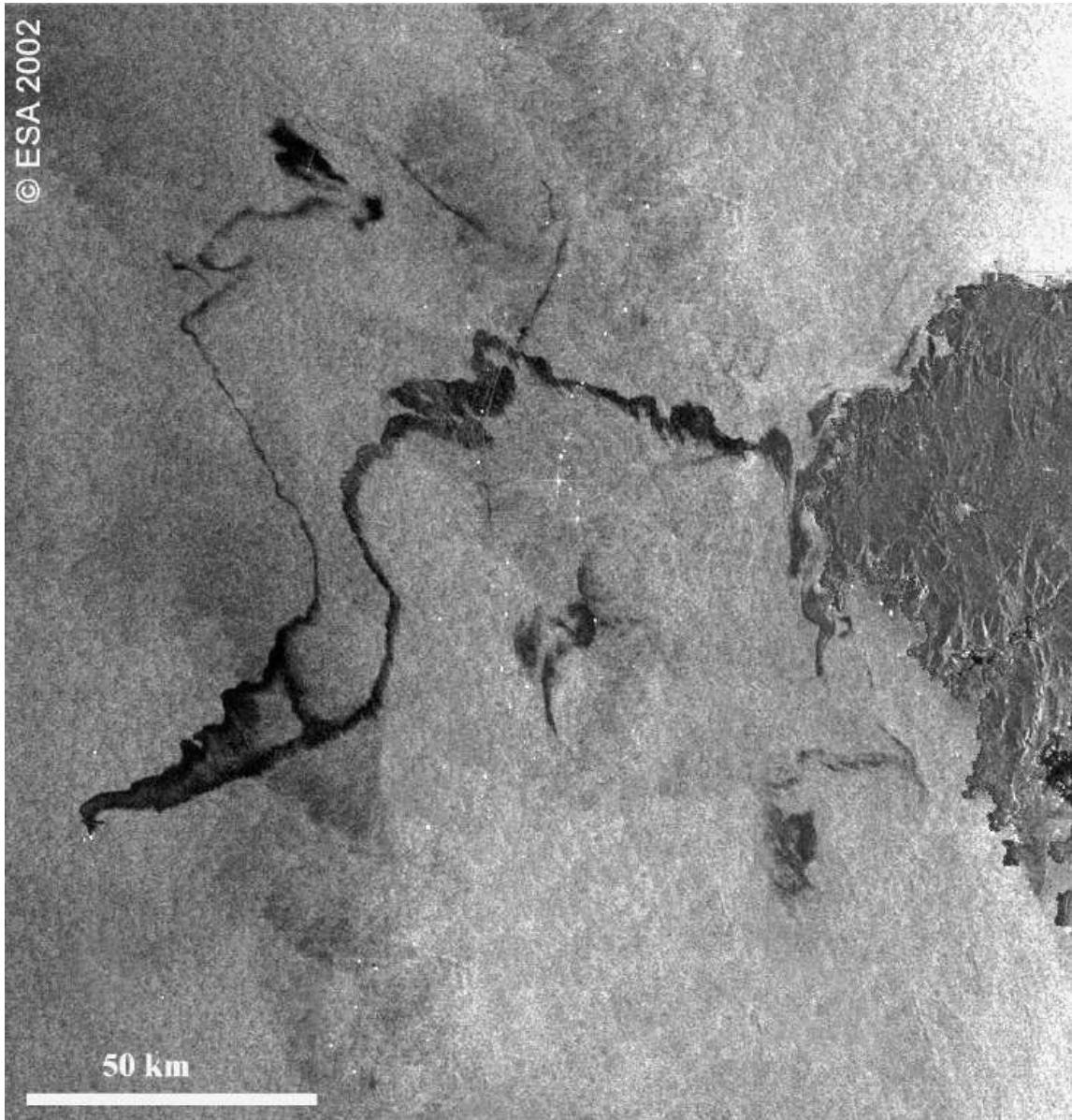


FIGURE 3.10 – *Image SAR de la marée noire causée par le Prestige à proximité des côtes de Galice (Espagne) en novembre 2002, d'où seront extraites les différentes images utilisées dans cette partie (image fournie par l'ESA).*

### 3.3.2 Etape 2 : classification par seuillage

Comme les contours ont été bien localisés lors de l'étape de partition en régions homogènes [38], l'étape de classification que nous proposons maintenant ne nécessite qu'une étape de fusion de régions. La technique de seuillage proposée consiste simplement à fusionner les différentes régions  $\Omega_r$  - non nécessairement adjacentes - en comparant leur niveau de gris moyen  $m_r$  à un seuil  $T$  donné. L'image est ainsi divisée en 2 classes,  $C_1(T) = \{\Omega_r | m_r < T\}$  et  $C_2(T) = \{\Omega_r | m_r \geq T\}$ .

La valeur de ce seuil  $T$  peut être automatiquement estimée en minimisant la complexité stochastique  $\Delta$ . Pour cela, nous devons déterminer la valeur de la complexité stochastique associée à une partition en deux classes  $C_1(T)$  et  $C_2(T)$ . Remarquons que cette valeur se calcule

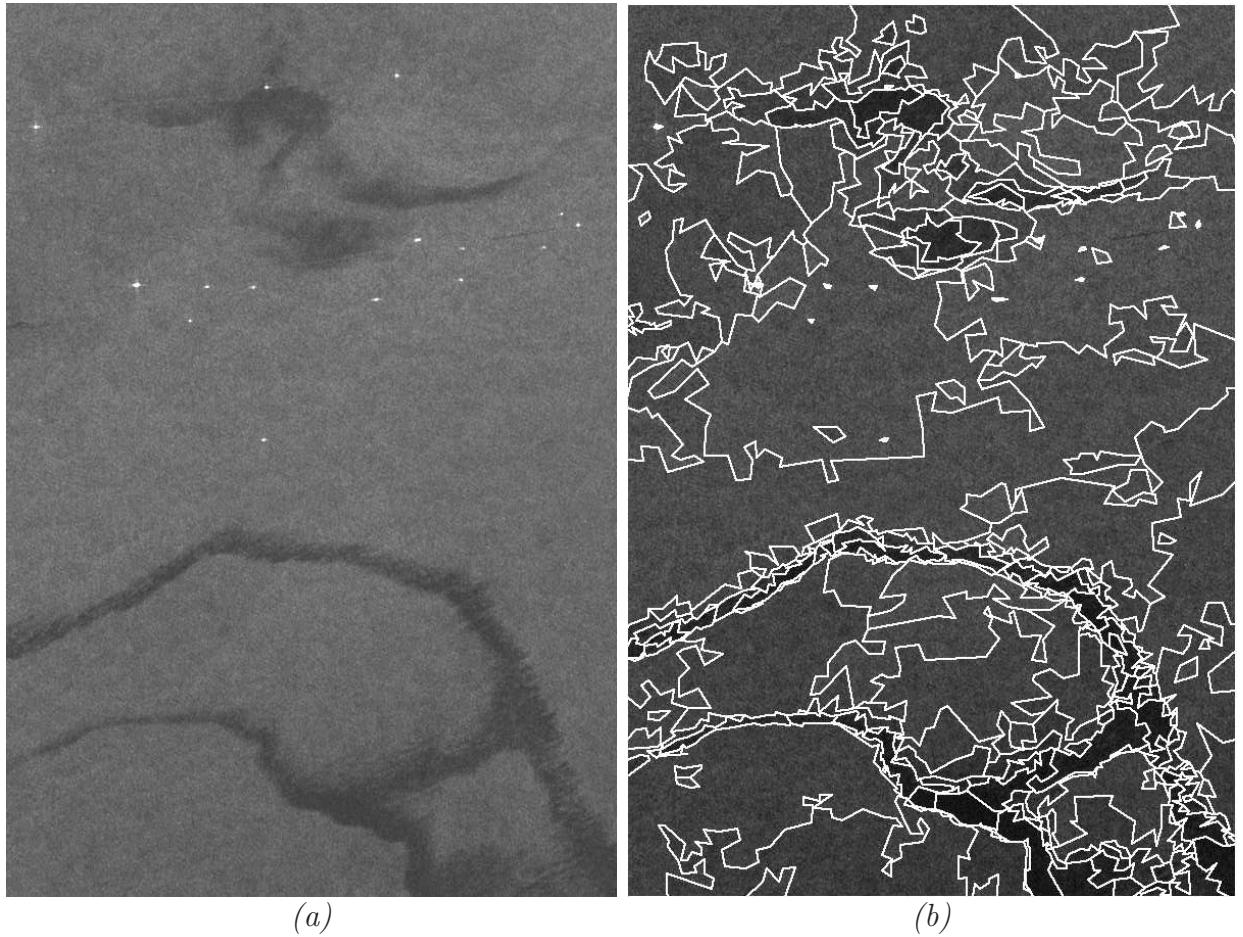


FIGURE 3.11 – *Etape 1 : (a) image SAR ( $598 \times 875$  pixels) obtenue par le satellite ASAR-ENVISAT (fournie par l'ESA) de la marée noire causée par le Prestige le long des côtes de Galice (Espagne). (b) Résultat de la partition en régions homogènes (temps de calcul : 6.7 secondes). Grille initiale : mur de briques de taille 8.*

très facilement puisque ces deux classes sont obtenue en appliquant une fusion de régions<sup>12</sup> à la grille issue de la première étape de partition en régions homogènes. Cette complexité stochastique, qui dépend de la valeur du seuil  $T$ , sera notée  $\Delta(T)$ . Nous avons ainsi tracé sur la fig. 3.12.a les variations de  $\Delta(T)$  obtenues pour différentes valeurs du seuil  $T$ . Le seuil optimal  $\tilde{T}$  est alors défini comme étant la valeur qui minimise le critère  $\Delta(T)$  :

$$\tilde{T} = \operatorname{argmin}_T \Delta(T) \quad (3.2)$$

On remarque sur la fig. 3.12.a que  $\tilde{T} = 331$ , ce qui conduit à la partition en deux classes représentée sur la fig. 3.12.b.

Nous présentons sur la fig. 3.13 un autre résultat de segmentation de nappes de pétrole sur une image SAR. Soulignons que sur cet exemple, nous fournissons uniquement le temps de calcul associé à la première étape de partition en régions homogènes, dans la mesure où l'étape d'estimation de la valeur du seuil n'a pas encore été optimisée en temps de calcul. Nous mon-

12. Le critère utilisé pour cette fusion consiste simplement à regrouper toutes les régions appartenant à la même classe.

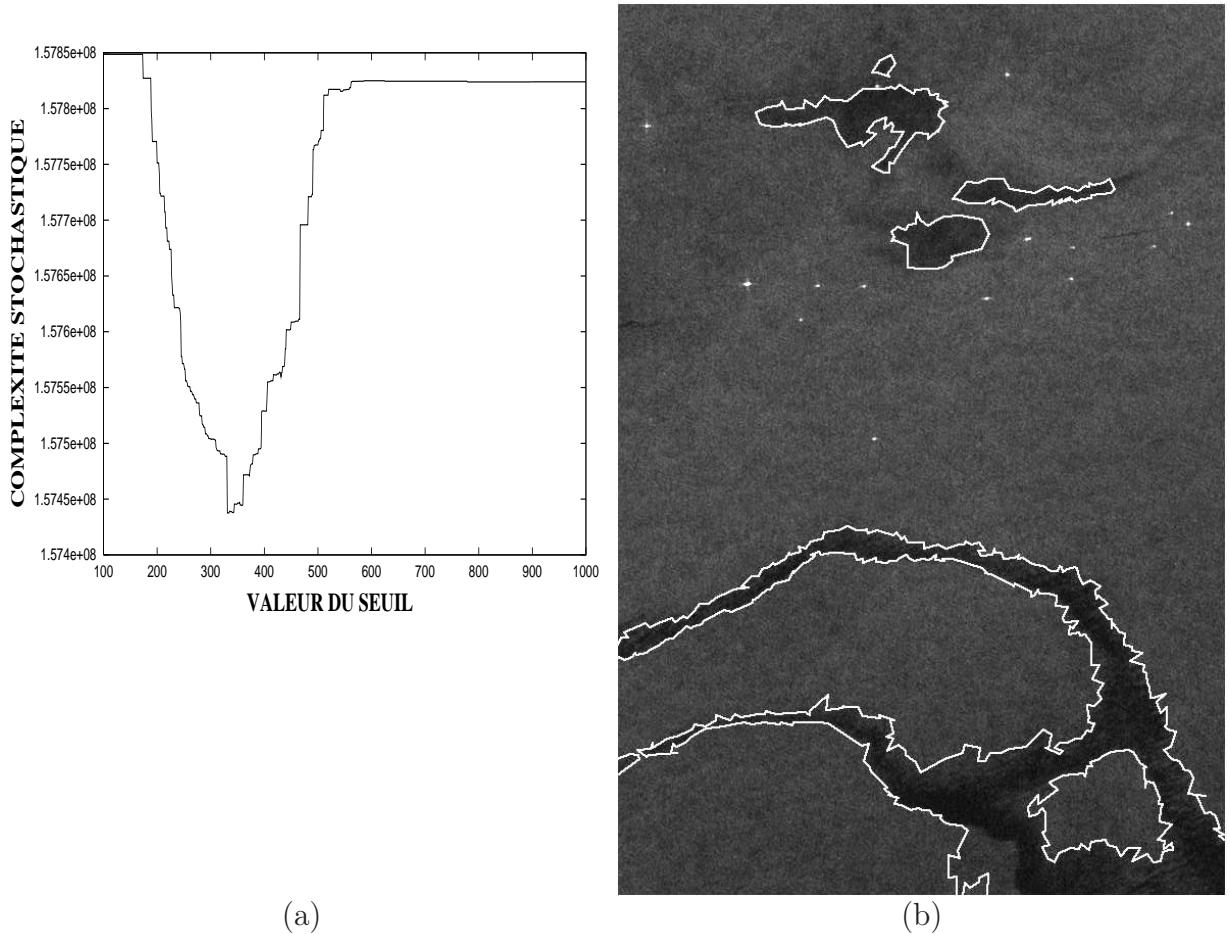


FIGURE 3.12 – Etape 2 : (a) Variations de la complexité stochastique  $\Delta(T)$  en fonction du seuil  $T$ . Le minimum est atteint pour  $\tilde{T} = 331$ . (b) Grille associée à cette partition en 2 classes.

trons également sur la fig. 3.14, les histogrammes associés à chacune des deux classes obtenues après segmentation pour les deux images SAR qui viennent d'être présentées. Nous constatons sur cette figure que pour ces deux images, les histogrammes des deux classes sont largement superposés, ce qui montre qu'un simple seuillage basé sur l'histogramme global des images, sans partition préalable, aurait été inefficace.

Considérons maintenant l'image de la fig. 3.15.a, qui n'est plus constituée de deux mais de trois classes : pétrole, mer et côtes. Afin d'obtenir une segmentation en trois classes  $C_1$ ,  $C_2$  et  $C_3$ , le seuillage précédemment décrit doit être généralisé à deux seuils  $T_1$  et  $T_2$ . Après avoir partitionné l'image en régions homogènes, le premier seuil  $T_1$  est déterminé en utilisant le seuillage qui vient d'être présenté. Nous obtenons  $\tilde{T}_1 = 675$ .

Considérons maintenant  $\Delta(T_1, T_2)$  la complexité stochastique associée à 2 seuils  $T_1$  et  $T_2$ , c'est-à-dire aux 3 classes  $C_1(T_1, T_2) = \{\Omega_r | m_r < t_1\}$ ,  $C_2(T_1, T_2) = \{\Omega_r | t_1 \leq m_r < t_2\}$  et  $C_3(T_1, T_2) = \{\Omega_r | m_r \geq t_2\}$  où  $t_1 = \min(T_1, T_2)$  et  $t_2 = \max(T_1, T_2)$ . Comme dans le cas d'un seuil unique, nous définissons le seuil  $\tilde{T}_2$  optimal au sens de la MCS de la façon suivante :

$$\tilde{T}_2 = \operatorname{argmin}_{T_2} \Delta(T_1 = \tilde{T}_1, T_2) \quad (3.3)$$

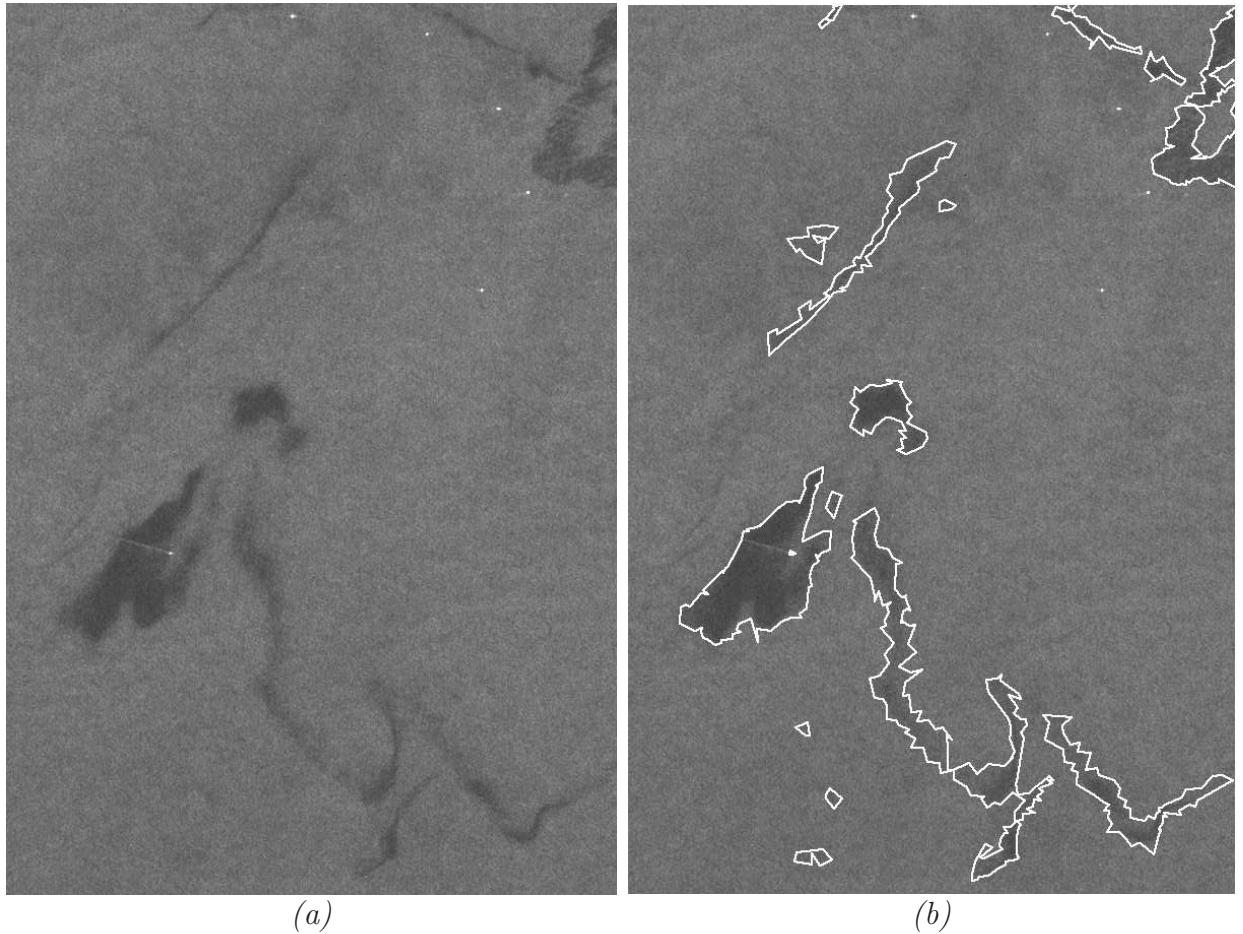


FIGURE 3.13 – Segmentation de nappes de pétrole : (a) image ASAR-ENVISAT ( $601 \times 878$  pixels) de la marée noire causée par le Prestige (image fournie par l'ESA). (b) Segmentation des nappes de pétrole contenues dans l'image à l'aide de l'algorithme de segmentation en 2 étapes décrit dans cette partie (temps de calcul associé à la première étape de partition en régions homogènes : 6.4 secondes).

qui conduit dans notre cas à  $\tilde{T}_2 = 416$ <sup>13</sup>. Le résultat de segmentation obtenu avec une première étape de partition en régions homogènes suivie de la technique de seuillage qui vient d'être décrite ( $\tilde{T}_1 = 675$  et  $\tilde{T}_2 = 416$ ) est présenté sur la fig. 3.15.b.

En conclusion, les fig. 3.12.b, fig. 3.13.b et fig. 3.15.b montrent l'intérêt de cet algorithme en 2 étapes qui permet de segmenter des zones sombres susceptibles d'être des nappes de pétrole sans nécessiter un réglage manuel des seuils. Seul le nombre de seuils à estimer doit être fourni, ce qui correspond simplement au nombre de classes souhaité. Ce nombre de classes est la seule connaissance *a priori* nécessaire à l'algorithme. Par exemple, le nombre de classes dans l'image de la fig. 3.15.a peut *a priori* être fixé à 3, les 3 classes étant : une classe de faible réflectivité incluant nappes de pétrole et surfaces de mer à l'abri du vent, une classe de

13. Nous avons vérifié que nous obtenions un résultat équivalent lorsque  $T_1$  et  $T_2$  sont estimés simultanément, c'est-à-dire lorsque l'on considère  $(\tilde{T}_1, \tilde{T}_2) = \text{argmin}_{T_1, T_2} \Delta(T_1, T_2)$ . Bien sûr, l'estimation simultanée de tous les seuils conduit à des temps de calculs beaucoup plus importants, c'est pourquoi nous avons choisi d'estimer les différents seuils les uns après les autres.

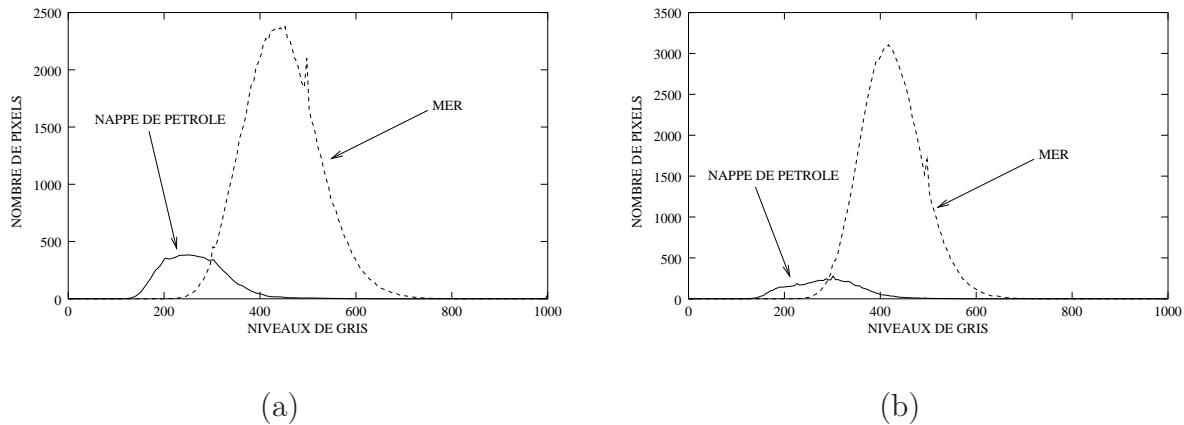


FIGURE 3.14 – Histogrammes de chacune des deux classes des fig. 3.11.a et fig. 3.13.a, déduits (a) de la segmentation obtenue sur la fig. 3.12.b, (b) de celle obtenu sur la fig. 3.13.b.

réflectivité moyenne constituant la plus grande partie de la surface de la mer et une classe de forte réflectivité correspondant à la côte et aux bateaux.

### 3.3.3 Amélioration de la robustesse et discussion

L'image de la fig. 3.15.a étant fortement inhomogène - notamment la mer - une étape supplémentaire de déplacement et de suppression de noeuds peut être appliquée en vue d'améliorer la robustesse de la technique de segmentation proposée.

Afin de comprendre la principale motivation pour introduire cette étape supplémentaire, analysons la petite région située en bas à droite de la fig. 3.15.b. Cette petite région ayant une moyenne inférieure à 416, elle est donc affectée à la classe "pétrole", alors que la région qui l'entoure appartient à la classe "mer". Cependant, ses frontières ne correspondent à aucune frontière réelle : l'affectation de cette région à la classe "pétrole" semble donc peu pertinente.

Une solution simple pour résoudre ce problème consiste à estimer à nouveau la position des contours, c'est-à-dire à utiliser à nouveau une phase de déplacement des noeuds puis de suppression des noeuds inutiles. Cette étape de déplacement et de suppression de noeuds ne modifiera pas les contours correspondants aux frontières vraies de l'image, alors que les autres contours, et notamment ceux de la petite région située en bas à droite de la fig. 3.15.b, seront fortement modifiés. Les moyennes des niveaux de gris dans chaque région pouvant alors avoir fortement varié, il est nécessaire de revoir l'affectation de chaque région à une classe. Ainsi, la segmentation finale (fig. 3.16.a) est obtenue en seuillant cette nouvelle partition avec les 2 seuils  $\tilde{T}_1 = 675$  et  $\tilde{T}_2 = 416$  déterminés précédemment. Nous présentons également l'histogramme de chacune des trois classes associées sur la fig. 3.16.b.

Cette technique a également été appliquée sur les deux autres images étudiées précédemment et nous pouvons constater que dans ces cas aussi, elle permet de supprimer les petites régions aberrantes tout en préservant la qualité des contours des régions d'intérêt (voir fig. 3.17). Soulignons enfin que cette étape étant appliquée sur une grille contenant peu de noeuds, elle est très rapide (moins d'une seconde pour les trois exemples présentés), tout en ayant permis d'éliminer un grand nombre de petites régions.

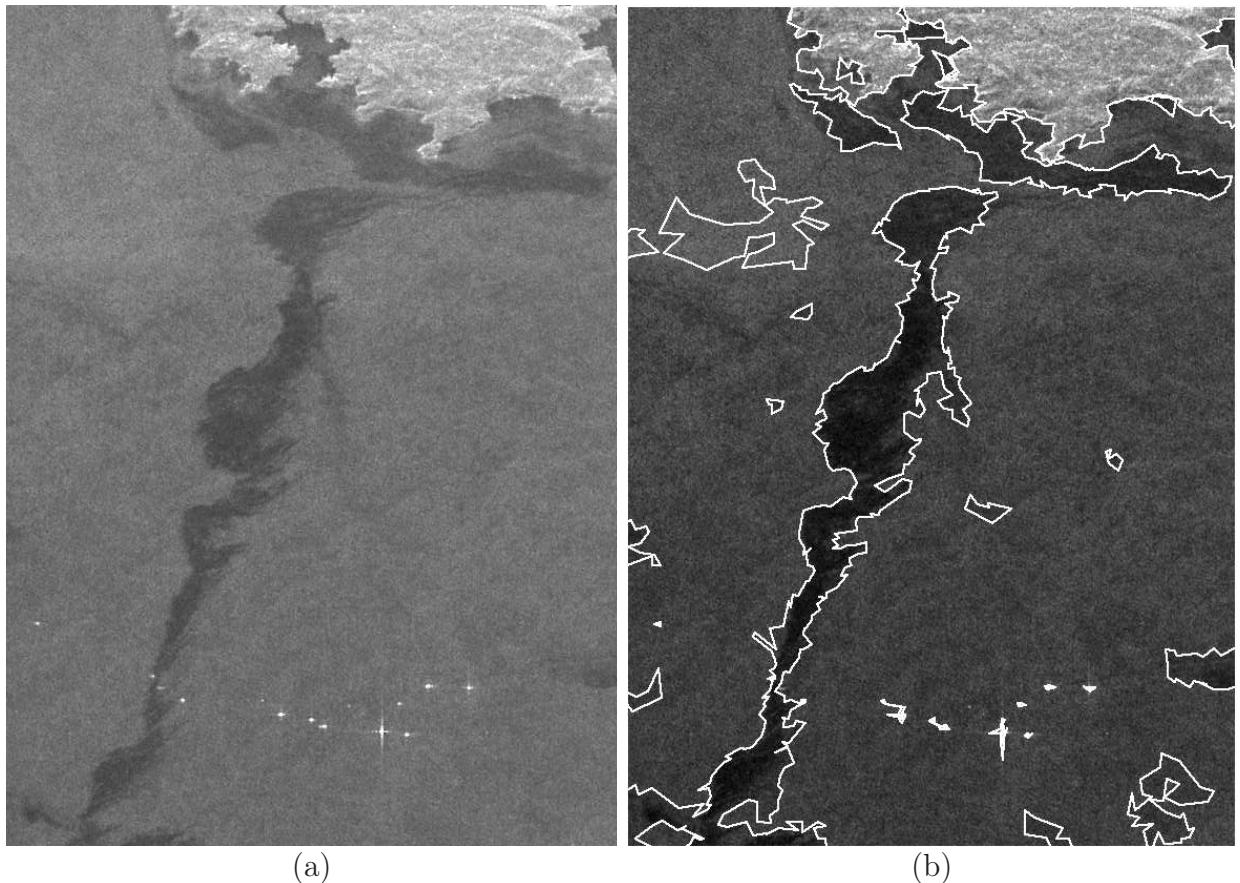


FIGURE 3.15 – Segmentation de nappes de pétrole : (a) images ASAR-ENVISAT ( $581 \times 801$  pixels) de la marée noire causée par le Prestige (fournie par l'ESA). (b) Grille associée à la partition en trois classes obtenue en appliquant les deux seuils estimés  $\tilde{T}_1 = 675$  et  $\tilde{T}_2 = 416$  sur la valeur moyenne des niveaux de gris de chaque région obtenue après la première étape de partition en régions homogènes (temps de calcul associé à la première étape de partition en régions homogènes : 6.4 secondes).

Revenons maintenant au résultat de segmentation obtenu sur le fig. 3.16.a. Nous remarquons que les 10 zones de faible réflectivité, qui correspondent à des nappes de pétrole potentielles, ont été segmentées. Cette segmentation pourra alors permettre à un algorithme d'identification d'extraire les caractéristiques pertinentes de chaque zone (telles que taille, forme, distance à la côte, intensité moyenne, texture, ...) afin de les classer de façon supervisée ou non supervisée. L'étude de ces caractéristiques se situe au-delà des objectifs que nous nous étions fixés, c'est-à-dire la détection et la segmentation des nappes de pétrole potentielles. Cependant, à titre d'exemple de caractéristique intéressante pouvant être extraite, nous présentons sur la fig. 3.16.c, l'histogramme des niveaux de gris calculés sur chacune des 10 nappes potentielles.

Afin d'illustrer l'intérêt de la première étape de partition en régions homogènes par rapport à une simple technique de seuillage, nous montrons sur la fig. 3.18.a le résultat de la partition en 3 classes obtenue en appliquant les 2 seuillages précédents ( $\tilde{T}_1 = 675$  et  $\tilde{T}_2 = 416$ ) directement

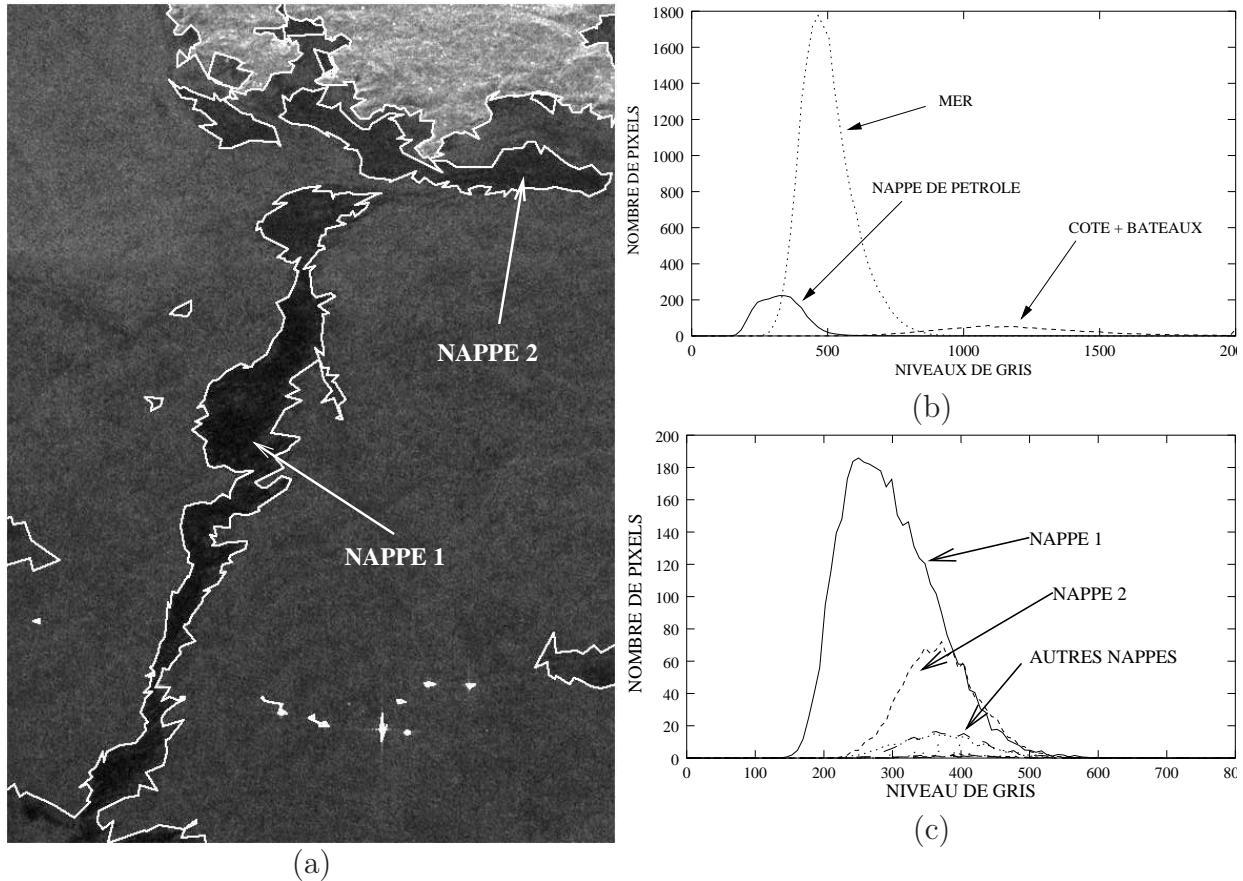


FIGURE 3.16 – (a) : Segmentation de la fig. 3.15.a obtenue en appliquant sur la grille de la fig. 3.15.b, une étape additionnelle destinée à accroître la robustesse. (b) Histogramme des 3 classes détectées en (a). (c) Histogramme des 10 nappes de pétrole détectées en (a).

sur l'image initiale de la fig. 3.15.a. Dans le but de comparer ce résultat à celui obtenu avec la technique de segmentation proposée, nous avons affiché sur la fig. 3.18.b le résultat de segmentation associé (déduit de la fig. 3.16.a) en utilisant les mêmes conventions de couleur pour chaque classe. Il apparaît clairement que le seuillage effectué sur la partition en régions homogènes, plutôt que sur les niveaux de gris de chacun de pixels pris individuellement, permet de réduire très fortement le nombre de pixels mal classés.

### 3.4 Conclusion

Les résultats obtenus dans ce chapitre nous permettent de souligner un point important. En effet, dès le début de ce manuscrit, nous avons choisi de nous restreindre à la partition en régions homogènes plutôt que de poser le problème en terme beaucoup plus général de segmentation d'images. Cette restriction nous a permis d'aboutir à une méthode de partition simple, rapide et bien adaptée à la segmentation d'images SAR de zones agricoles, comme nous avons pu le voir dans la première partie de ce chapitre.

Cependant, même dans le cas où les régions ne sont plus homogènes, comme par exemple dans le cadre de la segmentation de nappes de pétrole, nous avons mis en évidence que la partition en régions homogènes pouvait être utilisée avantageusement comme une étape de

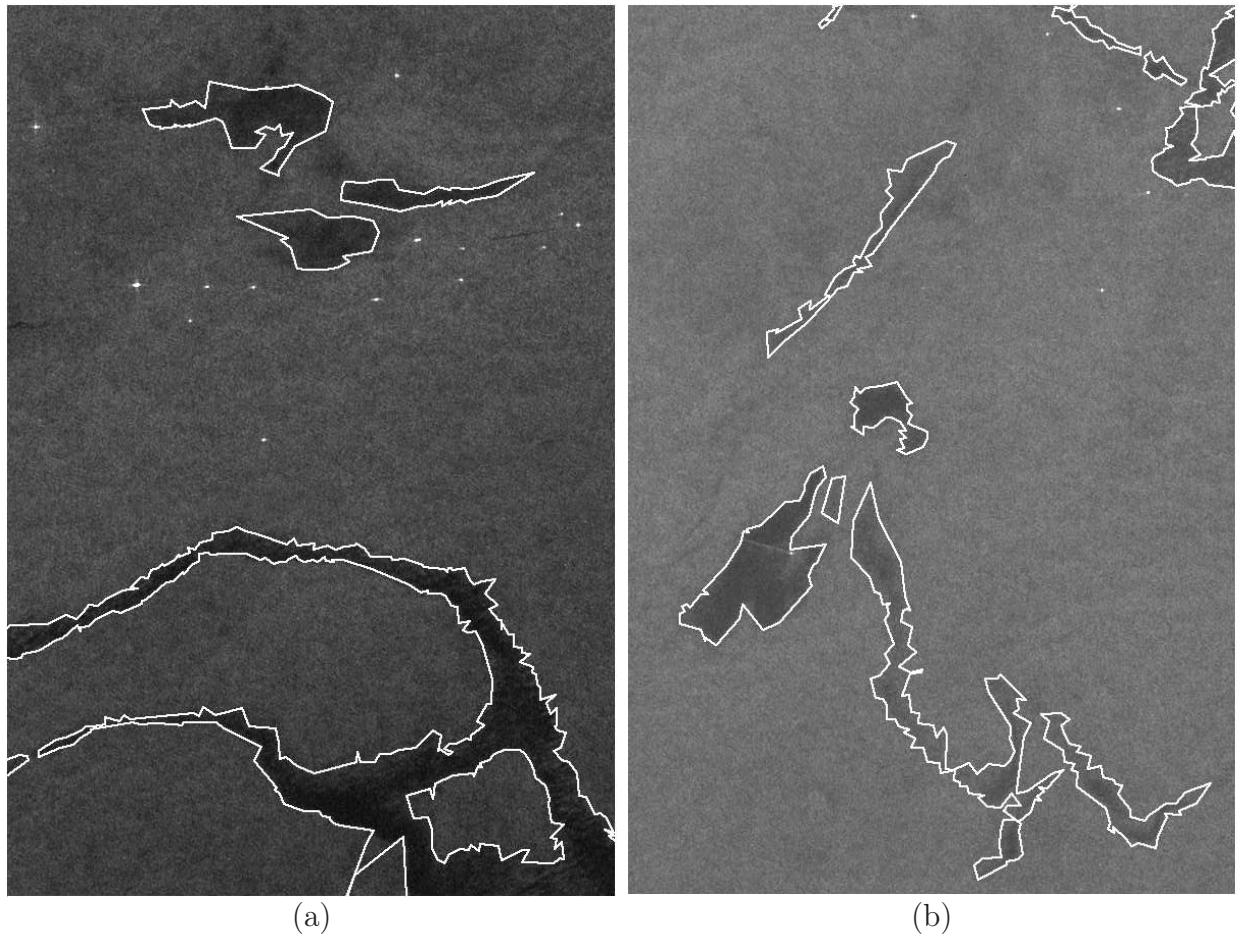


FIGURE 3.17 – Utilisation d'une étape additionnelle destinée à accroître la robustesse : (a) segmentation obtenue à partir de la grille de la fig. 3.12.b - (b) idem mais à partir de la grille de la fig. 3.13.b.

pré-segmentation rapide et permettant une bonne localisation des contours.

Nous avons mis en évidence dans le chapitre précédent que la complexité stochastique pouvait être utilisée pour estimer divers paramètres de l'algorithme d'optimisation, notamment la taille de la grille initiale. Nous avons montré ici qu'elle permet également l'estimation de paramètres du modèle d'image, comme l'ordre de la loi gamma. D'autre part, la segmentation de nappes de pétrole nous a amenés à utiliser la complexité stochastique en tant que critère de qualité, nous permettant de choisir automatiquement la valeur des seuils qui interviennent dans une étape de post-traitement par seuillages.

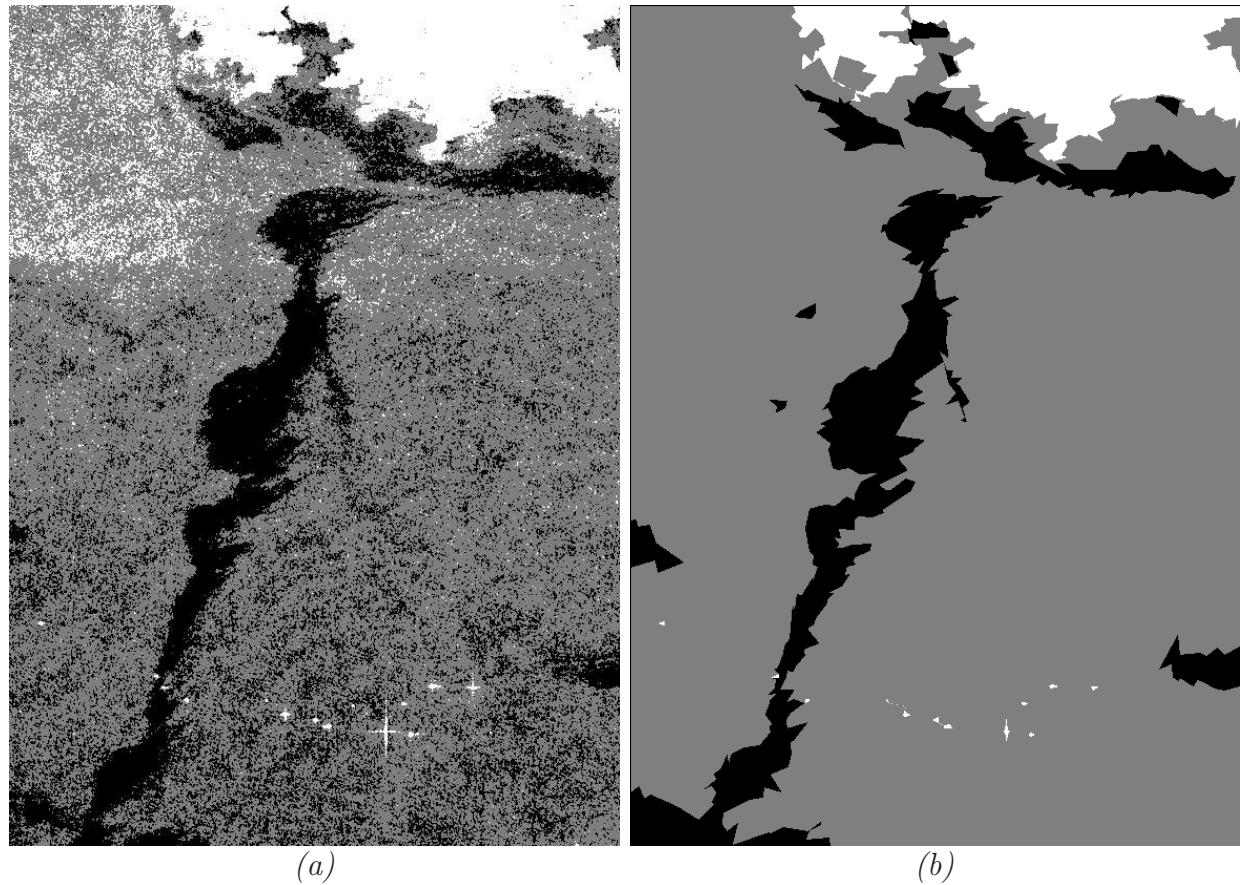


FIGURE 3.18 – Illustration de l'intérêt de la première étape de partition en régions homogènes : partition en 3 classes obtenue (a) en appliquant directement 2 seuillages ( $\tilde{T}_1 = 675$  et  $\tilde{T}_2 = 416$ ) sur les niveaux de gris de chaque pixel (et non sur la moyenne des régions), et (b) en utilisant la technique de segmentation proposée. Chacune des trois classes est représentée par un niveau de gris différent.

# Chapitre 4

## Généralisations du modèle

### Sommaire

---

<b>4.1 Généralisation à d'autres lois de probabilités . . . . .</b>	<b>64</b>
4.1.1 Détermination de la complexité stochastique . . . . .	64
4.1.2 Cas des images vidéo . . . . .	68
4.1.3 Généralisation à des modèles d'images plus complexes . . . . .	70
4.1.4 Utilisation de la partition en régions homogènes comme pré-segmentation dans des images vidéo . . . . .	73
<b>4.2 Images vectorielles . . . . .</b>	<b>77</b>
4.2.1 Détermination de la complexité stochastique . . . . .	77
4.2.2 Images de type multi-dates . . . . .	78
4.2.3 Images dont les composantes possèdent des ddp de familles différentes	84
4.2.4 Images SAR polarimétriques . . . . .	86
4.2.5 Conclusion . . . . .	88
<b>4.3 Segmentation par contours actifs multi-régions . . . . .</b>	<b>91</b>
4.3.1 Complexité stochastique . . . . .	91
4.3.2 Algorithme de segmentation par contours actifs multi-régions . . . . .	92
4.3.3 Résultats de segmentation . . . . .	93
4.3.4 Introduction de connaissances <i>a priori</i> sur les paramètres de la cible	100
4.3.5 Segmentation multi-lois . . . . .	101
<b>4.4 Conclusion . . . . .</b>	<b>104</b>

---

Dans le chapitre précédent, nous avons présenté quelques exemples d'applications en imagerie SAR de l'algorithme de partition en régions homogènes. Au cours de ce chapitre, nous allons proposer quelques généralisations du modèle utilisé jusqu'à présent. Rappelons pour cela que le modèle défini dans le chapitre 2 se divise en deux parties : la première liée à la définition de la partition  $w$  et la deuxième liée au modèle statistique de la distribution des niveaux de gris. Les généralisations proposées dans les parties 4.1 et 4.2 toucheront ainsi à la partie statistique de la modélisation, alors que la généralisation introduite dans la partie 4.3 portera sur le modèle de grille utilisé.

Plus précisément, nous allons généraliser le modèle à d'autres lois de probabilités dans la partie 4.1, puis nous traiterons dans la partie 4.2 du cas des images à plusieurs composantes (c'est-à-dire vectorielles), avant de présenter dans la partie 4.3 comment à l'aide de la technique développée, il est possible d'obtenir un algorithme de segmentation par contours actifs multi-régions.

## 4.1 Généralisation à d'autres lois de probabilités

Dans les chapitres précédents, nous n'avons travaillé que sur des images SAR, réelles ou synthétiques, que nous avons modélisées à l'aide de ddp gamma. Nous allons maintenant généraliser l'expression de la complexité stochastique à d'autres lois de probabilité.

### 4.1.1 Détermination de la complexité stochastique

Comme nous l'avons exposé dans le chapitre 2, les ddp appartenant à la famille exponentielle présentent un certain nombre d'avantages :

- les estimées de leurs paramètres au sens du MV sont à variance minimum [24, 30] (si l'estimateur est non biaisé),
- ces estimateurs dépendent uniquement de statistiques exhaustives qui peuvent être calculées à l'aide de la technique rapide proposée dans la partie 2.2.5,
- un certain nombre de bruits présents en imagerie se modélisent bien à l'aide de lois appartenant à la famille exponentielle.

Nous donnons dans le tableau 4.1 quelques exemples de ddp<sup>14</sup> appartenant à la famille exponentielle que nous utiliserons par la suite.

Comme nous l'avons vu dans le chapitre 2, seuls les termes  $\Delta_P(\widehat{\boldsymbol{\theta}}|w)$  et  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  dépendent du choix de la ddp. Il est en effet nécessaire de connaître la taille  $\alpha$  du vecteur de paramètres de la ddp pour le calcul de  $\Delta_P(\widehat{\boldsymbol{\theta}}|w)$  et de déterminer sa log-vraisemblance généralisée pour le calcul de  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$ . En effectuant le même type de calcul que dans le cas d'une ddp gamma d'ordre  $L$  (voir annexe A.2), nous pouvons obtenir la log-vraisemblance généralisée de toutes les lois présentées dans le tableau 4.1 et donc l'expression de  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$ <sup>15</sup>.

Nous précisons ainsi dans le tableau 4.2 la valeur du paramètre  $\alpha$ , ainsi que l'expression de  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  et de l'estimée  $\widehat{\theta}_r$  (au sens du MV) du vecteur de paramètres  $\theta_r$  dans la région  $\Omega_r$  pour différentes ddp de la famille exponentielle.

En combinant les expressions fournies dans ce tableau et les équations 2.7 et 2.8 précisant les expressions de  $\Delta_G(w)$  et de  $\Delta_P(w)$ , nous obtenons l'expression de la complexité stochas-

14. Dans ce manuscrit, nous employons le terme densité de probabilité (ddp) aussi bien pour des lois continues que pour des lois discrètes.

15. Car, rappelons-le,  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  est égal à l'opposé de la log-vraisemblance généralisée de l'image.

ddp	$\mathcal{D}$	Paramètres	ddp
Gaussienne	$\mathbb{R}$	$(m, \sigma)$	$\frac{1}{\sqrt{2\pi}\sigma} e^{\frac{1}{2}(\frac{x-m}{\sigma})^2}$
Exponentielle	$\mathbb{R}^+$	$m$	$\frac{1}{m} e^{-\frac{x}{m}}$
Gamma d'ordre $L$	$\mathbb{R}^+$	$m$	$\frac{L^L}{m^L \Gamma(L)} x^{L-1} e^{-\frac{L}{m}x}$
Rayleigh	$\mathbb{R}^+$	$\beta$	$\frac{x}{\beta^2} e^{\frac{-x^2}{2\beta^2}}$
Poisson	$\mathbb{N}$	$m$	$\frac{m^x}{x!} e^{-m}$
Géométrique	$\mathbb{N}$	$p$	$p(1-p)^x$
Bernoulli	$\{0, 1\}$	$p$	$p^x(1-p)^{(1-x)}$

TABLE 4.1 – Quelques exemples de ddp. Nous avons noté  $\mathcal{D}$  l'ensemble de définition des ddp. Notons qu'une loi exponentielle est en fait une loi gamma d'ordre  $L = 1$ .

ddp	$\alpha$	$\Delta_L(\mathbf{s} \widehat{\boldsymbol{\theta}}, w)$	$\widehat{\theta}_r$
Gaussienne	2	$\sum_{r=1}^R \frac{1}{2} N_r \log \left( \widehat{\theta}_r^{(2)} \right) + K_1$	$\widehat{\theta}_r^{(1)} = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)$ $\widehat{\theta}_r^{(2)} = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)^2 - \left( \widehat{\theta}_r^{(1)} \right)^2$
Exponentielle	1	$\sum_{r=1}^R N_r \log \left( \widehat{\theta}_r \right)$	$\widehat{\theta}_r = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)$
Gamma d'ordre $L$	1	$L \sum_{r=1}^R N_r \log \left( \widehat{\theta}_r \right) + K_2$	$\widehat{\theta}_r = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)$
Rayleigh	1	$\sum_{r=1}^R N_r \log \left( \widehat{\theta}_r \right) + K_3$	$\widehat{\theta}_r = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)^2$
Poisson	1	$-\sum_{r=1}^R N_r \widehat{\theta}_r \log \left( \widehat{\theta}_r \right) + K_4$	$\widehat{\theta}_r = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)$
Géométrique	1	$-\sum_{r=1}^R N_r \left( \log \widehat{\theta}_r + \frac{1-\widehat{\theta}_r}{\widehat{\theta}_r} \log \left( 1 - \widehat{\theta}_r \right) \right)$	$\widehat{\theta}_r = 1 / \left( 1 + \frac{1}{N_r} \sum_{\Omega_r} s(x, y) \right)$
Bernoulli	1	$-\sum_{r=1}^R N_r \left\{ \widehat{\theta}_r \log \widehat{\theta}_r + (1 - \widehat{\theta}_r) \log(1 - \widehat{\theta}_r) \right\}$	$\widehat{\theta}_r = \frac{1}{N_r} \sum_{\Omega_r} s(x, y)$

TABLE 4.2 – Dimension  $\alpha$  du vecteur de paramètres, longueur du codage des niveaux de gris  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  et estimée  $\widehat{\theta}_r$  au sens du Maximum de Vraisemblance du vecteur de paramètres pour les différentes ddp du tableau 4.1. Les termes additionnels (notés  $K_1$ ,  $K_2$ ,  $K_3$  et  $K_4$ ) intervenant dans  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$  et indépendants de la partition ne sont pas précisés.

tique associée à chacune des ddp du tableau 4.1. Nous sommes donc en mesure d'appliquer l'algorithme de partition en régions homogènes à d'autres types d'images.

A titre d'exemple, montrons quelques résultats de partitions obtenues sur des images synthétiques. Nous présentons ainsi sur les fig. 4.1, 4.2 et 4.3 la partition d'images synthétiques perturbées respectivement par un bruit de Poisson, un bruit gaussien et un bruit de Bernoulli.

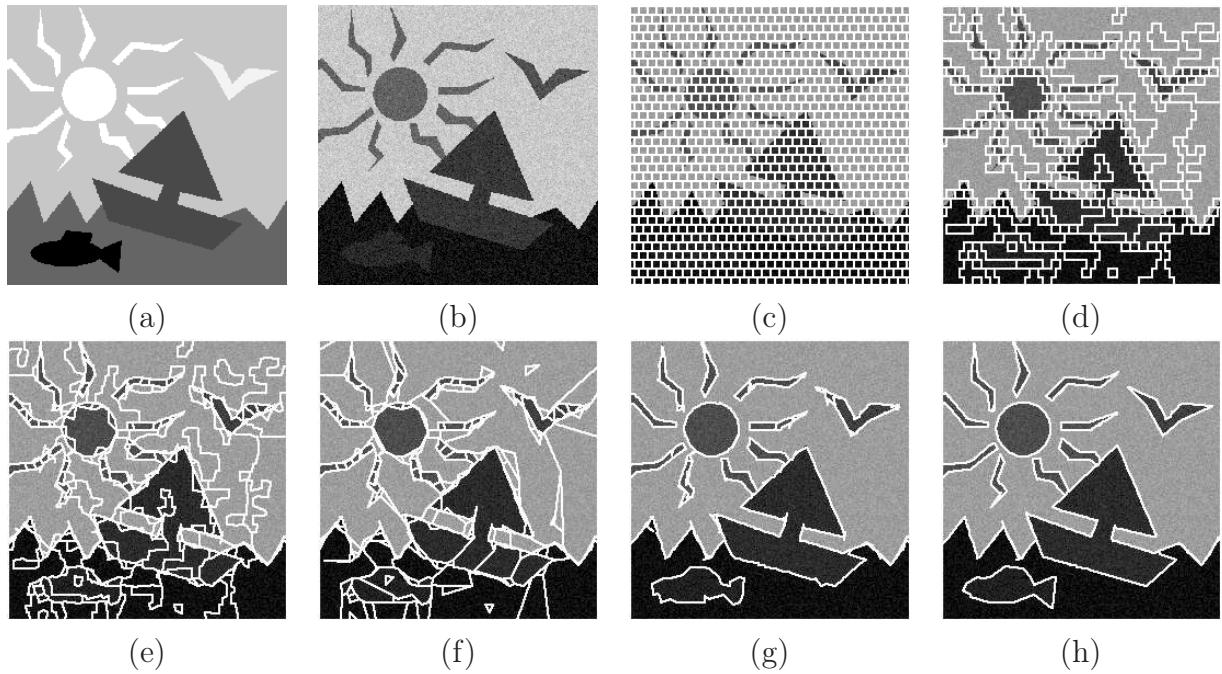


FIGURE 4.1 – Partition d'une image synthétique ( $256 \times 256$  pixels) perturbée par un bruit de Poisson. (a) image non bruitée - (b) image perturbée par un bruit de Poisson - (c) grille initiale (mur de briques de taille 8) - (d) partition obtenue après fusion de régions GLRT ( $\tilde{S} = 3$ ) - (e) après déplacement des nœuds - (f) après suppression des nœuds inutiles - (g) après fusion de régions (MCS) - (h) partition obtenue après un autre cycle de déplacement, suppression et fusion. Temps de calcul : moins de 2 secondes.

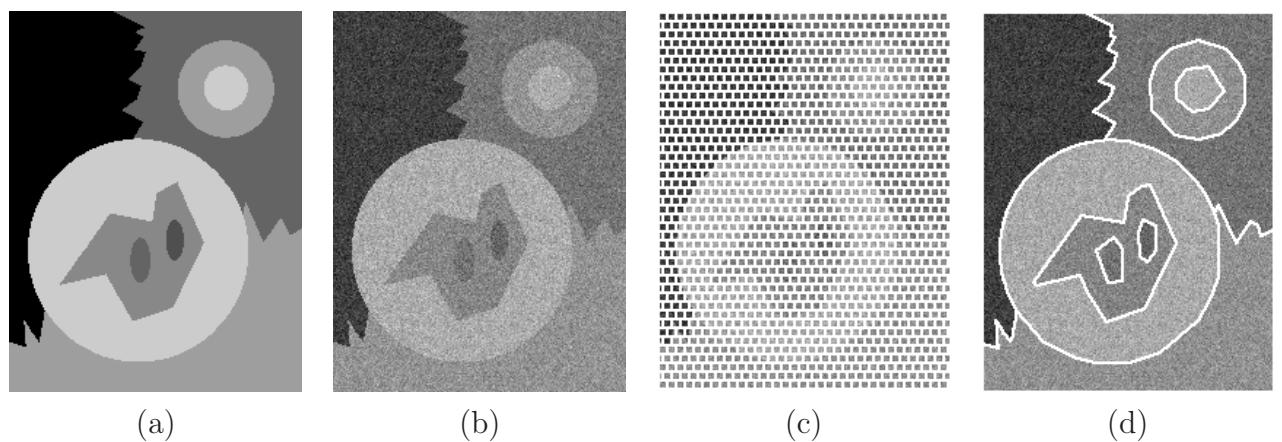


FIGURE 4.2 – Partition d'une image synthétique ( $201 \times 261$  pixels) perturbée par un bruit gaussien d'écart type  $\sigma = 40$ . (a) image non bruitée - (b) image bruitée - (c) grille initiale (mur de briques de taille 6) - (d) résultat de partition. Temps de calcul : moins de 2 secondes.

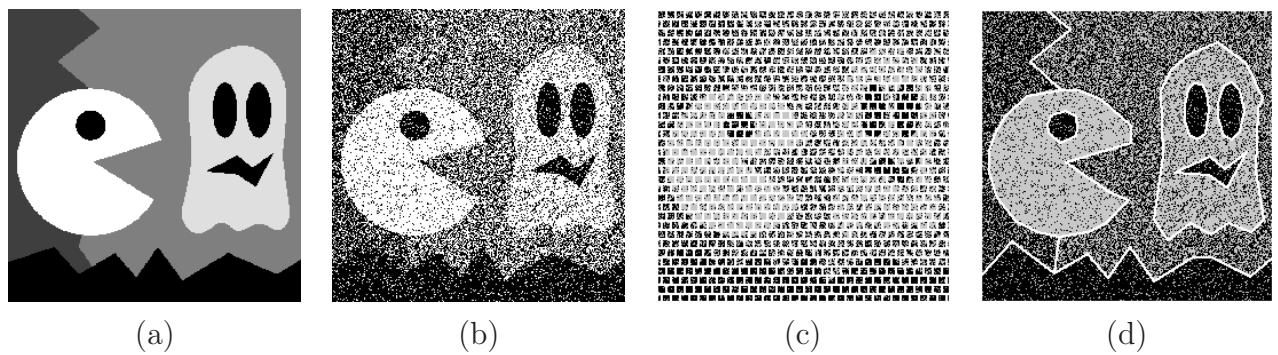


FIGURE 4.3 – Partition d'une image synthétique ( $256 \times 256$  pixels) perturbée par un bruit de Bernoulli. (a) image non bruitée - (b) image bruitée - (c) grille initiale (mur de briques de taille 8) - (d) résultat de partition. Temps de calcul : moins de 1 seconde.

### 4.1.2 Cas des images vidéo

Nous allons maintenant nous intéresser au cas des images vidéo. En effet, même si l'étude de ce type d'imagerie ne constitue pas l'objet principal de cette thèse, plutôt consacrée à l'imagerie SAR, son importance est telle que nous ne pouvions pas le passer sous silence. Nous pouvons ainsi voir sur les fig. 4.4, 4.5 et 4.6, trois exemples de partition en régions homogènes sur des images vidéo. Pour ces trois exemples, nous avons choisi un critère gaussien.

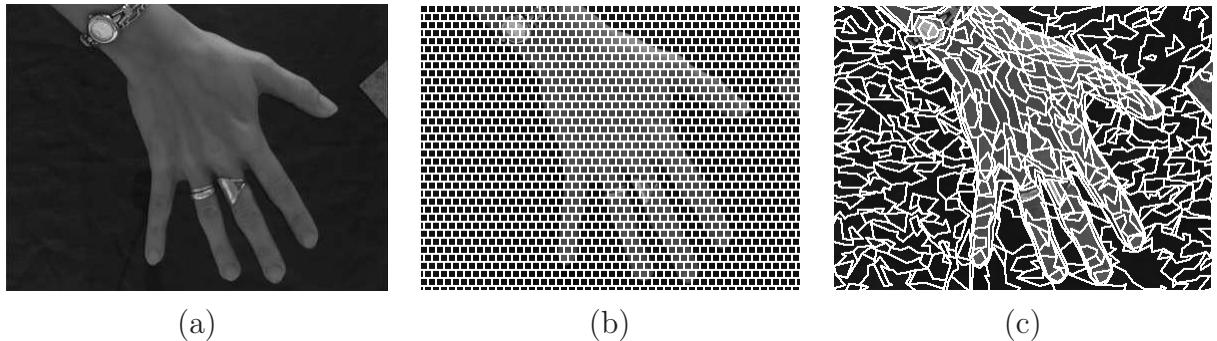


FIGURE 4.4 – Partition d'une image vidéo ( $376 \times 288$  pixels) en régions homogènes à l'aide d'un critère gaussien. (a) image - (b) grille initiale (mur de briques de taille 8) - (c) résultat.

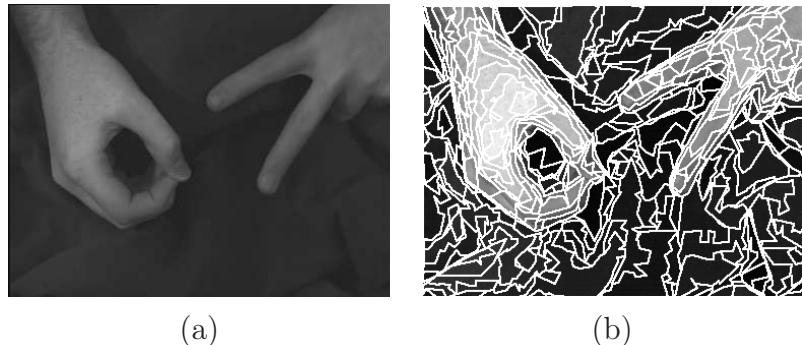


FIGURE 4.5 – Partition d'une image vidéo ( $376 \times 288$  pixels) en régions homogènes à l'aide d'un critère gaussien. (a) image - (b) résultat. Grille initiale : mur de briques de taille 8.

Comme nous pouvons le voir sur ces différents résultats, chacune de ces partitions comporte beaucoup plus de régions qu'on pourrait le souhaiter pour une segmentation qui correspondrait à des objets distincts. En effet, on pourrait souhaiter par exemple que la main dans l'image de la fig. 4.4 soit segmentée en une seule région. Cependant, la main ne constitue pas une région homogène dans la mesure où les variations de la moyenne des niveaux de gris sont très nettement supérieures à la variance du bruit présent dans l'image. Ainsi, comme dans le cas des images SAR de nappes de pétroles (voir partie 3.3), nous voyons que la partition en régions homogènes comporte beaucoup plus de régions que la segmentation correspondant aux objets présents dans l'image, du fait de l'inhomogénéité des régions.

La généralisation à de nouvelles ddp et donc l'application de notre technique de partition en régions homogènes à d'autres types d'imagerie, notamment vidéo, nous ramène au problème, déjà rencontré pour les nappes de pétroles, de la segmentation d'images constituées de

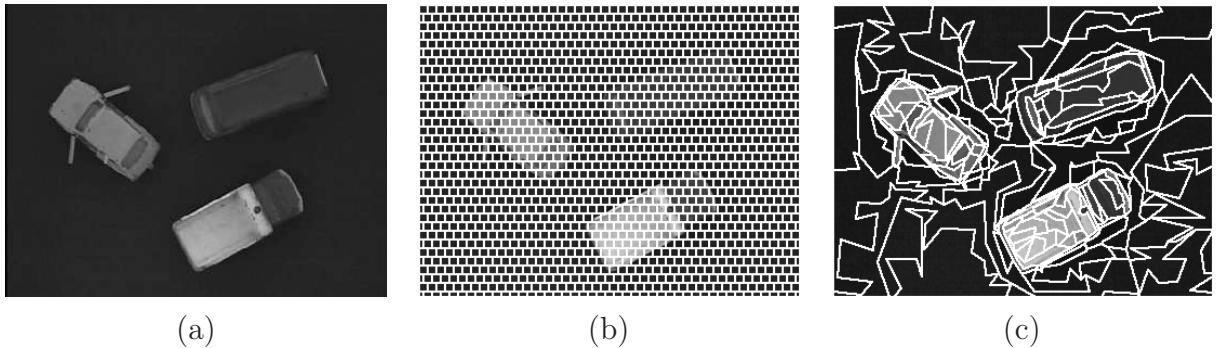


FIGURE 4.6 – Partition d'une image vidéo ( $376 \times 288$  pixels) en régions homogènes à l'aide d'un critère gaussien. (a) image - (b) grille initiale (mur de briques de taille 8) - (c) résultat.

régions inhomogènes.

D'après ces quelques exemples, deux concepts bien distincts se dégagent : la partition en régions homogènes et la segmentation en objets significatifs, cette dernière nécessitant des concepts de haut niveau. Ainsi, dans le cas d'images vidéo, si notre méthode de partition fournit bien des régions homogènes, ces dernières ne constituent pas une segmentation en objets significatifs.

Une première solution consiste à se servir de cette partition en régions homogènes comme étape de pré-segmentation. La segmentation en objets significatifs sera alors obtenue en effectuant divers post-traitements intégrant des concepts de plus haut niveaux. Par exemple, il est possible d'utiliser, comme dans le cas de nappes de pétrole (partie 3.3), une étape de classification basée sur des seuillages. La partie 4.1.4 présentera ainsi quelques résultats obtenus sur des images vidéo en utilisant de tels post-traitements.

Une autre solution consiste à chercher des représentations pour lesquelles les objets significatifs correspondent à des régions homogènes. A titre d'exemple, il est possible de travailler, non plus sur l'image d'intensité mais sur les composantes colorimétriques U et V dans l'espace de couleur YUV (voir partie 4.3.3) ou sur la composante de teinte H dans l'espace de couleur YHS [83] (voir partie 5.3.3 et 5.3.4), mais aussi sur une image d'angle de polarisation (voir partie 5.3.3). Il est également possible d'appliquer des prétraitements destinés à faire ressortir des caractéristiques propres à un objet, comme ce sera le cas dans la partie 5.3.1 où nous travaillerons sur des images pré-classifiées. Dans tous ces exemples, l'utilisation de notre méthode de partition en régions homogènes est intéressante car ces nouvelles représentations de l'image sont bruitées et les objets présents sont beaucoup plus homogènes.

Enfin, il est possible d'envisager un troisième type de solutions, consistant toujours à injecter des informations de haut niveau, mais directement dans l'algorithme de partition, et non plus après (comme dans le cas de post-traitements) ou avant (comme dans le cas de prétraitements ou de changements de représentation destinés à rendre les objets homogènes). Nous proposerons ainsi en partie 4.3, un algorithme de segmentation par contours actifs multi-régions, c'est-à-dire la généralisation d'une grille active au cas où le nombre d'objets dans l'image et leur topologie sont connus *a priori*.

Soulignons enfin que dans les images vidéo, les régions étant très inhomogènes et non bruitées, des techniques de détection de bords peuvent éventuellement être plus adaptées.

Mais avant de conclure cette partie sur les images vidéo par quelques exemples obtenus à

l'aide de post-traitements (partie 4.1.4), nous allons montrer que notre méthode peut se généraliser au cas où chaque région n'est plus caractérisée seulement par une moyenne constante, mais par une variation spatiale plus complexe du niveau de gris moyen. Nous allons pour cela utiliser une approche polynomiale, initialement proposée par Leclerc [56] dans le cadre de la segmentation par MCS.

### 4.1.3 Généralisation à des modèles d'images plus complexes

Dans [56], Leclerc propose un algorithme de segmentation d'images, et notamment vidéo, prenant en compte un modèle d'image plus général que celui que nous avons choisi. Il ne suppose plus que la moyenne est constante dans la région mais qu'elle est décrite par une fonction polynomiale. Nous proposons de montrer dans cette section que le modèle d'image présenté dans la partie 2.1.1 peut être généralisé dans le cas de paramètres statistiques non constants par morceaux.

#### Détermination de la complexité stochastique

Plaçons-nous dans un premier temps dans le cas d'une ddp gaussienne. Jusqu'à présent, nous avons supposé que le vecteur de paramètres  $\theta_r$  dans chaque région  $\Omega_r$  était constant sur toute cette région. Dans le cas gaussien, le vecteur de paramètre  $\theta_r$  est composé de deux paramètres, l'espérance mathématique  $m_r$  et l'écart-type  $\sigma_r$  dans la région  $\Omega_r$ .

Supposons que  $\sigma_r$  est toujours constant dans la région mais que l'espérance mathématique  $m_r$  varie selon une fonction polynomiale. L'espérance mathématique dépend alors de la position  $(x, y)$  du pixel considéré et sera notée  $m_r(x, y)$ . Leclerc conjecture dans [56], qu'une fonction polynomiale de degré 2 est suffisante dans le cas des images vidéo, pour que la partition corresponde à une segmentation en objets significatifs. Nous avons donc calculé en annexe A.6 la complexité stochastique dans le cas où la moyenne varie selon une fonction linéaire d'équation  $m_r(x, y) = a_r x + b_r y + c_r$  (voir annexe A.6.1) ou une fonction quadratique d'équation  $m_r(x, y) = a_r x^2 + b_r y^2 + c_r x y + d_r x + e_r y + f_r$  (voir annexe A.6.2).

Il est intéressant de remarquer que dans cette complexité stochastique, la log-vraisemblance généralisée possède la même expression que dans le cas d'une ddp gaussienne avec moyenne constante dans les régions (voir tableau 4.2). Les seules différences apparaissent dans les estimations des paramètres statistiques au sens du MV<sup>16</sup>, qui se compliquent au fur et à mesure que le degré de la fonction polynomiale  $m_r$  augmente.

Le cas où la moyenne est constante par morceau ayant été assez simple à généraliser pour une ddp gaussienne, il est tentant de vouloir le faire pour d'autres ddp, notamment une ddp gamma. Cependant, comme nous le montrons dans l'annexe A.6.3, cette généralisation n'est pas possible dans le cas d'une ddp gamma : en effet, même lorsque l'espérance mathématique  $m_r(x, y)$  suit une fonction linéaire d'équation  $m_r(x, y) = a_r x + b_r y + c_r$ , il n'est plus possible d'utiliser l'algorithme rapide présenté en partie 2.2.5 pour remplacer les sommes 2D par des sommes 1D dans le calcul des estimés au sens du MV de  $a_r$ ,  $b_r$  et  $c_r$ . Il n'est donc pas judicieux dans notre cas d'utiliser ce type de généralisation pour une ddp gamma si nous souhaitons conserver des algorithmes rapides et nous limiterons donc notre étude au cas de ddp gaussiennes.

---

16. Dans le cas d'une fonction linéaire, ces paramètres sont  $a_r$ ,  $b_r$ ,  $c_r$  et  $\sigma_r$ , alors que dans le cas d'une fonction quadratique ce sont  $a_r$ ,  $b_r$ ,  $c_r$ ,  $d_r$ ,  $e_r$ ,  $f_r$  et  $\sigma_r$ .

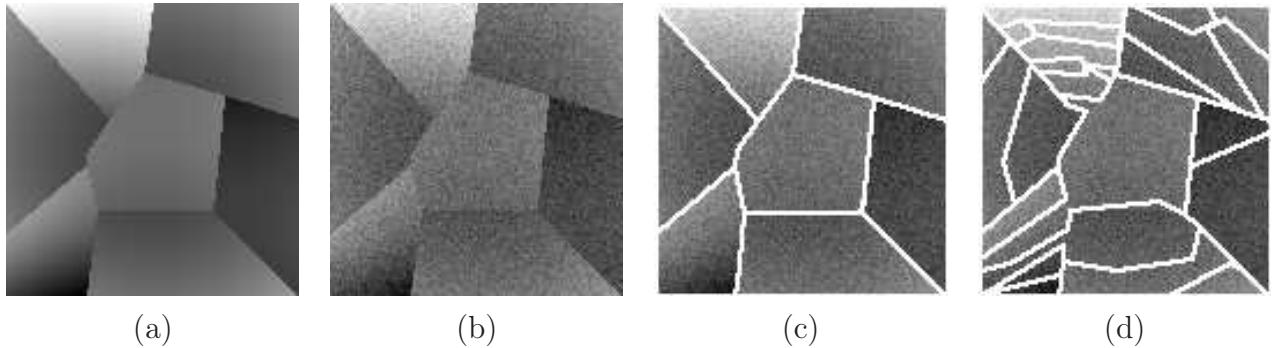


FIGURE 4.7 – Partition d'une image synthétique perturbée par un bruit gaussien dont l'espérance mathématique suit une surface quadratique différente dans chaque région. (a) image non bruitée ( $127 \times 127$  pixels) - (b) image bruitée - (c) partition avec un modèle où la moyenne suit une fonction quadratique - (d) partition avec le modèle de partition en régions homogènes.

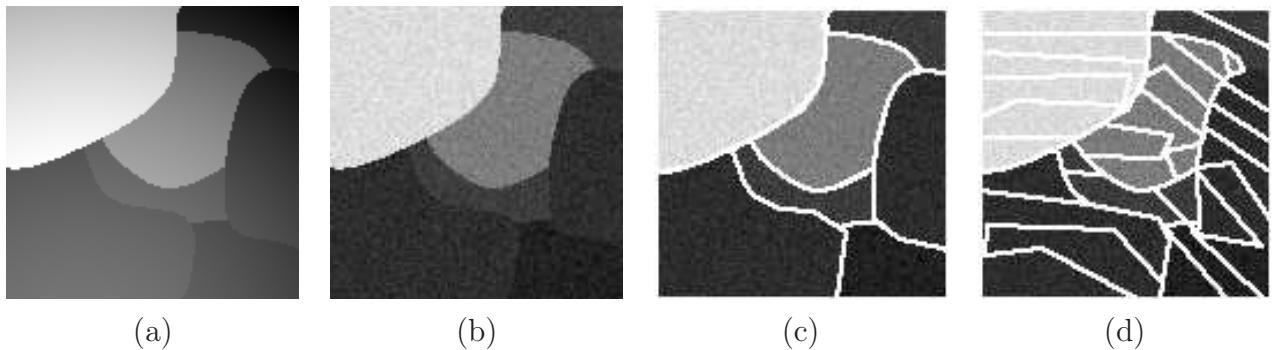


FIGURE 4.8 – Partition d'une image synthétique perturbée par un bruit gaussien, dont l'espérance mathématique suit une surface quadratique. Cette image a été obtenue en ajoutant une fonction quadratique (censée représenter l'influence d'un éclairage non uniforme) à une image de moyenne constante par morceaux. (a) image non bruitée ( $127 \times 127$  pixels) - (b) image bruitée - (c) partition avec un modèle où la moyenne suit une fonction quadratique - (d) partition avec le modèle de partition en régions homogènes.

Nous proposons de présenter quelques résultats obtenus sur des images synthétiques. Pour cela, nous avons synthétisé sur la fig. 4.7 une image composée de 7 régions, chacune ayant sa moyenne distribuée selon une fonction quadratique différente. Nous avons ensuite généré sur la fig. 4.8 une deuxième image, obtenue en ajoutant à une image de moyenne constante par morceaux, une fonction quadratique censée représenter l'influence d'un éclairage non uniforme.

Nous pouvons constater sur ces deux exemples que le modèle adapté, c'est-à-dire avec des moyennes distribuées selon des fonctions quadratiques, permet d'obtenir une partition qui correspond aux régions présentes dans l'image.

Ces résultats nous permettent de remarquer que la partition en régions homogènes est robuste. En effet, les objets ne correspondant pas à des régions homogènes, ils sont naturellement partitionnés en plusieurs régions. En revanche, bien que le modèle d'image d'un patchwork de régions homogènes ne soit pas adapté aux images que nous venons d'analyser, nous pouvons constater que les frontières entre objets sont globalement correctement localisées. La partition avec le modèle adapté conduit cependant bien à des résultats de meilleure qualité.

## Résultats sur des images réelles

Examinons le cas d'une image optique réelle. Nous présentons sur la fig. 4.9 les résultats de partition avec un critère gaussien en utilisant les trois modèles d'image qui viennent d'être présentés, à savoir une moyenne distribuée selon une constante, selon une fonction linéaire ou selon une fonction quadratique dans chaque région. Nous constatons que la complexification du modèle conduit toujours à une partition qui comporte un nombre de régions trop élevé pour constituer une segmentation qui corresponde aux objets présents dans l'image. Pour cette image, les niveaux de gris de chaque objet sont distribués selon une ddp dont l'espérance mathématique suit une surface complexe que ni un modèle constant par morceaux ni même un modèle quadratique n'approchent avec une précision suffisante, ceci étant d'autant plus vrai que le niveau de bruit dans cette image est très faible. En revanche, lorsque nous regardons les images reconstruites à partir de la grille en remplaçant les niveaux de gris dans chaque région  $\Omega_r$  par la moyenne  $m_r(x, y)$  dans cette région (voir ligne 3 de la fig. 4.9), nous constatons que les images obtenues avec un modèle plus complexe permettent une bien meilleure reconstruction.

Si l'utilisation d'un modèle plus complexe qu'un simple patchwork de régions homogènes peut s'avérer intéressante dans certains cas, comme nous l'avons montré sur les images synthétiques des fig. 4.7 et 4.8, nous voyons que dans le cas d'images vidéo, cette complexification n'entraîne pas de grandes modifications sur le résultat de la partition obtenue (mais par contre les temps de calcul ont été augmentés d'un facteur 6).

Le fait que Leclerc obtienne directement un algorithme de segmentation en régions d'intérêt en travaillant sur des images d'intensité (et non en travaillant sur d'autres représentations ou en utilisant des post-traitements) soulève des questions. A cela, nous pouvons avancer deux hypothèses.

La première est sans doute liée à la méthodologie développée dans son article [56], qui se retrouve d'ailleurs dans l'article de Zhu et Yuille [100], dans lequel le principe du MDL lui permet essentiellement de trouver un critère adapté à la segmentation d'images. L'absence de constantes de pondération entre les termes intervenant dans le critère n'est pas revendiquée dans cet article, ni d'ailleurs dans celui de Zhu et Yuille [100], ce qui explique que certains coefficients de pondération ne soient pas fixés dans le critère. A titre de comparaison, nous avons partitionné l'image de la fig. 4.10.a (la même que sur la fig. 4.4.a), non plus à l'aide de la complexité stochastique, mais en minimisant le critère suivant :

$$\Delta' = \lambda_1 \Delta_G + \lambda_2 \Delta_P + \lambda_3 \Delta_L \quad (4.1)$$

obtenu en ajoutant des constantes de pondération devant chacun des termes de la complexité stochastique. Le résultat est présenté sur la fig. 4.10.b et montre que le réglage de ces termes de pondération peut permettre d'obtenir une partition correspondant non plus à des régions homogènes mais aux objets présents dans l'image. Soulignons cependant que même si la segmentation obtenue avec cette méthode est intéressante, son obtention a nécessité le réglage manuel de paramètres qui ne seront pas nécessairement adaptés à d'autres images.

La seconde hypothèse pour expliquer les résultats obtenus par Leclerc [56], mais aussi par Zhu et Yuille [100] et Kanungo *et al* [49] vient du fait qu'ils utilisent une description du contour pixel à pixel, alors que dans notre cas, nous utilisons une description paramétrique du contour à l'aide de polygones. Le nombre de bits nécessaires pour décrire un contour avec une approximation polygonale étant beaucoup plus faible que pour le décrire pixel à pixel, le

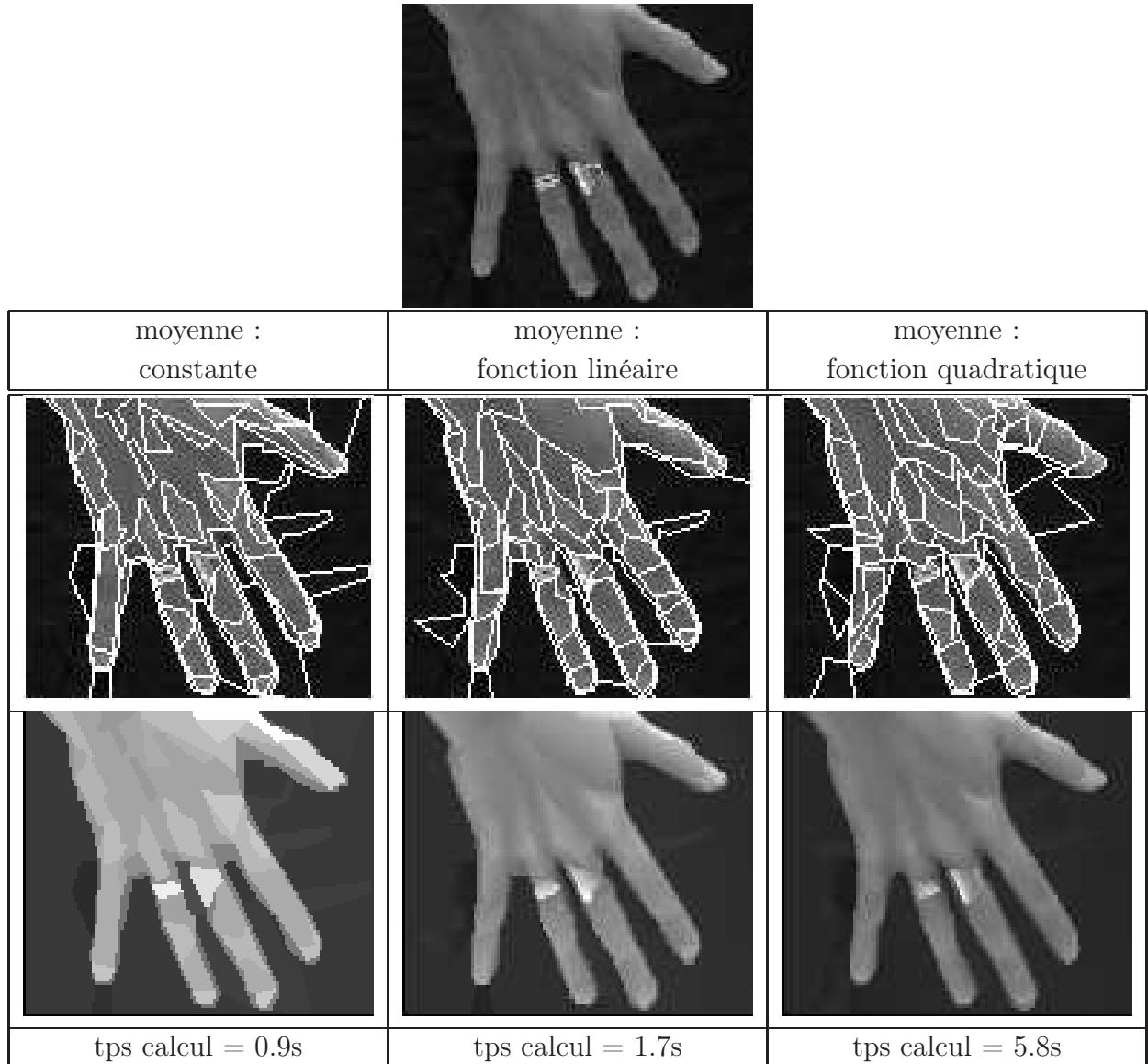


FIGURE 4.9 – Partition d'une image vidéo en utilisant un critère gaussien, avec la moyenne dans chaque région distribuée selon une constante (colonne 1), une fonction linéaire (colonne 2) ou une fonction quadratique (colonne 3). Ligne 1 : image ( $122 \times 106$  pixels) - Ligne 2 : résultat de la segmentation - Ligne 3 : moyenne dans chaque région - Ligne 4 : temps de calcul.

terme de codage du contour est donc sans doute surestimé dans [56, 100, 49] par rapport à notre approche, ce qui conduit à une sur-fusion. Si cette sur-fusion n'est pas nécessairement problématique dans le cas de l'imagerie vidéo, elle peut poser des problèmes dans d'autres types d'imagerie, notamment l'imagerie SAR.

#### 4.1.4 Utilisation de la partition en régions homogènes comme pré-segmentation dans des images vidéo

Dans cette partie, nous allons présenter quelques exemples d'utilisation de l'algorithme de partition en régions homogènes comme étape de pré-segmentation sur des images vidéo.

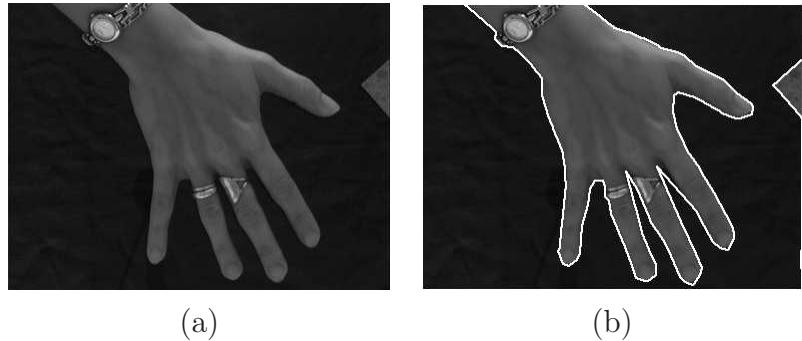


FIGURE 4.10 – Segmentation obtenue en pondérant les termes de la complexité stochastique. Le critère à minimiser s'écrit ainsi  $\Delta' = \lambda_1 \Delta_G + \lambda_2 \Delta_P + \lambda_3 \Delta_L$ . Dans cet exemple, nous avons pris  $\lambda_1 = 1$ ,  $\lambda_2 = 500$  et  $\lambda_3 = 1$ . Le terme  $\Delta_L$  est déterminé en utilisant un critère gaussien avec la moyenne dans chaque région distribuée selon une fonction quadratique. (a) Image à segmenter ( $376 \times 288$  pixels), (b) Résultat de la segmentation.

A titre d'exemple, nous avons repris sur la fig. 4.11.a, l'image de la fig. 4.6. Notre but est maintenant de segmenter les trois véhicules présents sur cette image (fig. 4.11.a), à partir de la partition en régions homogènes (fig. 4.11.b) obtenue précédemment. Afin d'obtenir une segmentation comportant un nombre de régions correspondant au nombre d'objets présents dans l'image, nous avons alors utilisé une nouvelle étape de fusion de régions mais en imposant cette fois le nombre de régions finales. Les fusions sont donc effectuées, en commençant bien sûr par celles entraînant la plus faible augmentation de la complexité stochastique, jusqu'à ce qu'il ne reste plus que quatre régions (fig. 4.11.c). Le résultat de la segmentation est alors obtenu en effectuant une nouvelle étape de déplacement puis de suppression des nœuds, afin d'affiner la position des contours (fig. 4.11.d).

Nous obtenons ainsi un résultat satisfaisant, et ceci grâce à l'ajout d'une information supplémentaire, à savoir le nombre de régions dans l'image.

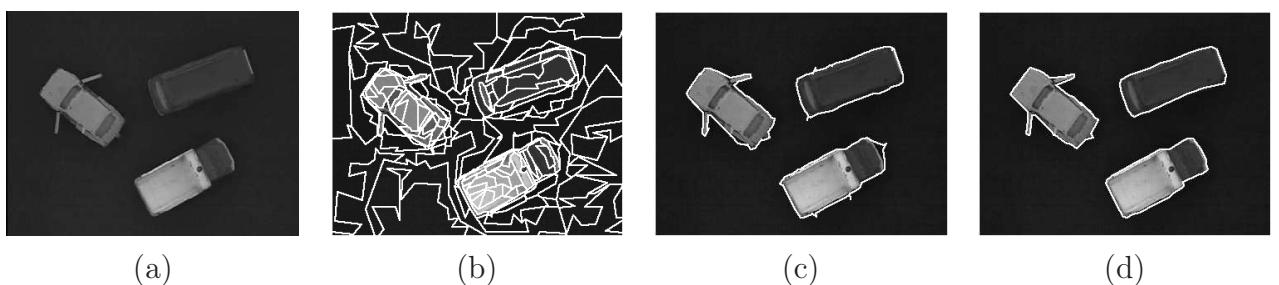


FIGURE 4.11 – Segmentation d'une image vidéo contenant trois véhicules, lorsque le nombre de régions dans l'image est connu. (a) image vidéo ( $376 \times 288$  pixels) - (b) partition en régions homogènes obtenue précédemment (voir fig. 4.6) - (c) fusion de régions en fixant le nombre de régions à 4 (les 3 véhicules et le fond) - (d) segmentation finale après une nouvelle étape de déplacement puis de suppression des nœuds.

Il serait également possible d'utiliser après cette pré-segmentation une étape de post-traitement basée sur d'autres critères et adaptée à l'application considérée, comme par exemple un algorithme de classification. Cependant, l'étude de tels post-traitements dépassant le cadre

des objectifs que nous nous sommes fixés, nous nous sommes contentés d'utiliser un simple seuillage sur les niveaux de gris moyens (avec estimation automatique de la valeur du seuil), comme dans le cas de la segmentation de nappes de pétrole (partie 3.3). Nous présentons ainsi sur les fig. 4.12 et 4.13 les résultats obtenus à partir des partitions en régions homogènes déterminées précédemment sur les fig. 4.4 et 4.5.

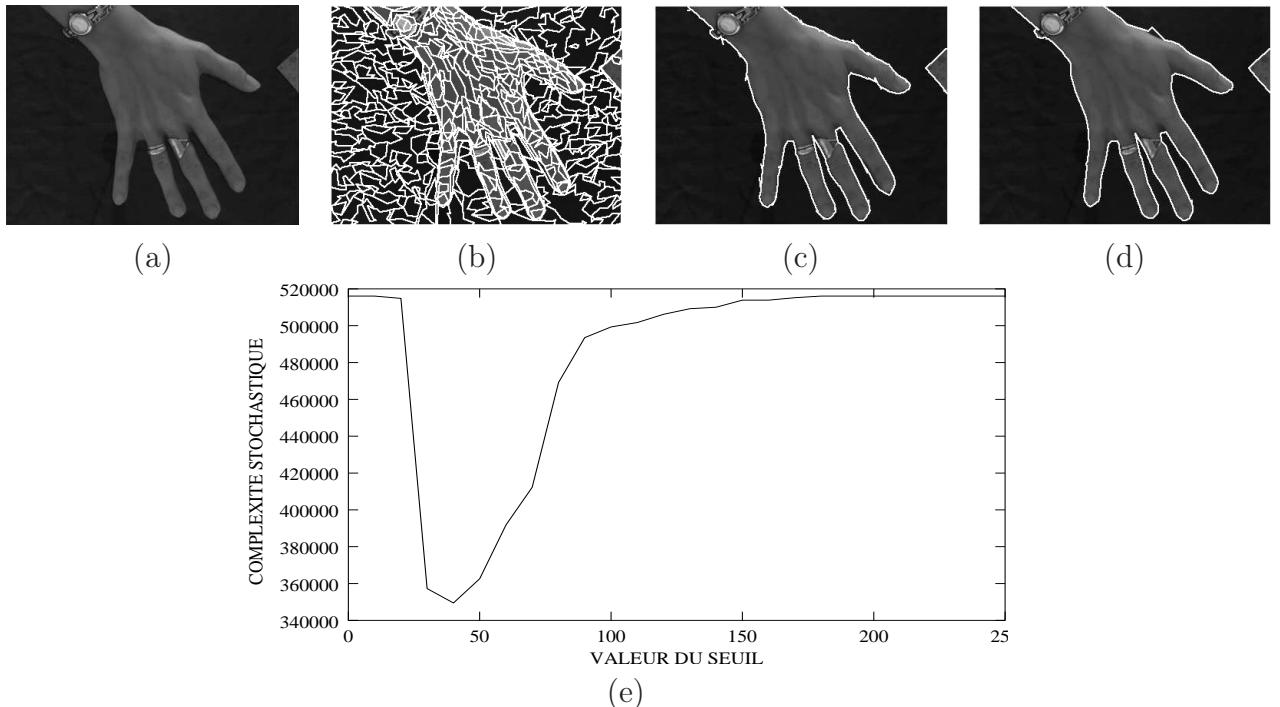


FIGURE 4.12 – Segmentation d'une image vidéo en deux classes à l'aide de l'algorithme de segmentation par seuillage automatique décrit en partie 3.3. (a) image vidéo ( $376 \times 288$  pixels) - (b) étape 1 : partition en régions homogènes obtenue précédemment (voir fig. 4.4) - (c) étape 2 : segmentation en 2 classes associée au minimum de la courbe (e) - (d) étape 3 : segmentation après une nouvelle convergence de la grille (amélioration de la robustesse). (e) évolution de la complexité stochastique en fonction de la valeur du seuil.

Ces quelques exemples montrent que même dans le cas d'images très inhomogènes et non bruitées, la partition en régions homogènes peut être utilisée comme pré-segmentation et conduire, moyennant quelques post-traitements ou la donnée de quelques informations supplémentaires, à une segmentation intéressante en régions qui correspondent à des objets significatifs.

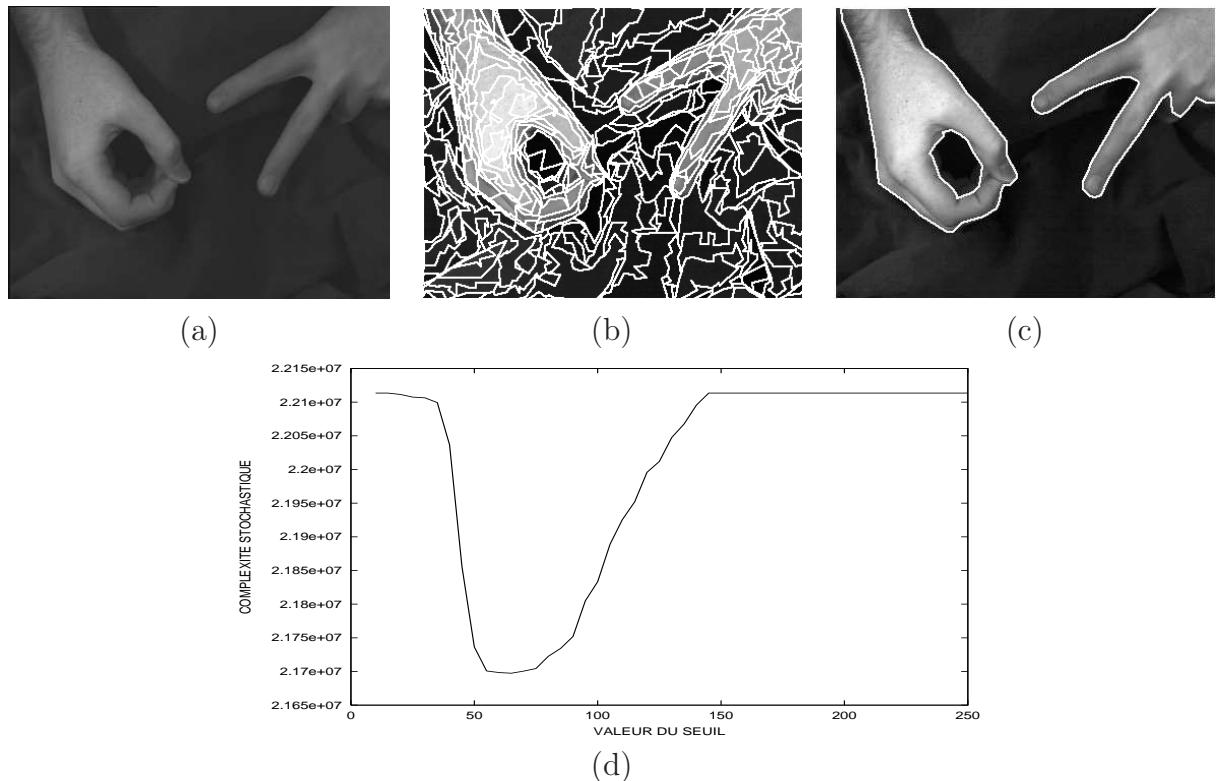


FIGURE 4.13 – Segmentation d'une image vidéo en deux classes à l'aide de l'algorithme de segmentation par seuillage automatique décrit en partie 3.3. (a) image vidéo ( $376 \times 288$  pixels) - (b) partition en régions homogènes (étape 1) obtenue précédemment (voir fig. 4.5) - (c) segmentation en 2 classes obtenue après seuillage de la partition (b) puis nouvelle convergence de la grille - (d) évolution de la complexité stochastique associée au seuil. Le résultat de la segmentation (c) est obtenu avec le seuil  $S = 65$  correspondant au minimum de la courbe (d).

## 4.2 Images vectorielles

Il existe de nombreux modes d'imagerie qui ne fournissent pas une simple valeur scalaire en chaque pixel. On peut bien entendu citer l'imagerie couleur, qui associe à chaque pixel l'intensité mesurée dans différentes bandes spectrales. Il peut également être intéressant d'associer à chaque pixel de l'image l'intensité mesurée à différents instants. Dans tous ces cas, on peut associer à chaque pixel un ensemble de valeurs que l'on peut représenter sous la forme d'un vecteur. On parlera alors d'images vectorielles. La généralisation de la technique de partition en régions homogènes afin de pouvoir traiter des images vectorielles est donc intéressante.

Dans cette partie, nous allons ainsi nous intéresser au cas des images vectorielles, afin notamment d'étudier les deux cas particuliers évoqués plus haut, à savoir les images multidates et multi-bandes.

### 4.2.1 Détermination de la complexité stochastique

Afin de généraliser la technique de partition aux images vectorielles, il est nécessaire de déterminer la complexité stochastique associée à de telles images. Une image vectorielle  $\mathbf{s}$  comportant  $N_x \times N_y$  pixels est constituée de  $M$  composantes, c'est-à-dire  $M$  images scalaires  $s_i$  ( $i \in \{1, 2, \dots, M\}$ ) :

$$\mathbf{s} = \{\mathbf{s}(x, y) = (s_1(x, y), s_2(x, y), \dots, s_M(x, y)) \mid (x, y) \in [1, N_x] \times [1, N_y]\} \quad (4.2)$$

Dans chaque région  $\Omega_r$ , les vecteurs de niveaux de gris des pixels  $\mathbf{s}(x, y)$ ,  $(x, y) \in \Omega_r$ , sont supposés distribués, comme dans le cas scalaire, selon une ddp  $P_{\theta_r}$  de vecteur paramètre  $\theta_r$ .

#### Cas général

La partition  $w$  étant définie à partir d'une grille polygonale unique<sup>17</sup>, nous obtenons la même complexité stochastique que dans le cas scalaire (équation 2.16). La différence entre le cas scalaire et vectoriel se situe dans l'expression de la log-vraisemblance.

Généralement, la détermination de la log-vraisemblance dans le cas d'une image vectorielle n'est pas triviale et doit être effectuée pour chaque ddp particulière. Par exemple, le cas de l'imagerie SAR polarimétrique (composée de  $M = 4$  composantes scalaires) sera étudié dans la partie 4.2.4. Cependant, dans le cas où les  $M$  composantes sont indépendantes, il est possible de ramener le calcul de la log-vraisemblance à  $M$  calculs de log-vraisemblance scalaire. Analysons maintenant ce cas particulier.

#### Composantes indépendantes

Supposons donc que les différentes composantes sont indépendantes. La ddp dans une région  $\Omega_r$  donnée s'écrit alors :

$$P_{\theta_r}(\mathbf{s}(x, y)) = \prod_{i=1}^M P_{\theta_r^i}(s_i(x, y)) \quad (4.3)$$

où  $P_{\theta_r^i}$  est la ddp des niveaux de gris  $s_i(x, y)$  de la composante  $i$ ,  $(x, y) \in \Omega_r$ , et où  $\theta_r^i$  est son vecteur de paramètre. La log-vraisemblance généralisée devient alors simplement :

$$\mathcal{L}_e \left[ \Omega_r | \widehat{\theta}_r \right] = \sum_{i=1}^M \mathcal{L}_e \left[ \Omega_r^i | \widehat{\theta}_r^i \right] \quad (4.4)$$

17. La partition  $w$  est donc commune à toutes les composantes.

où  $\Omega_r^i = \{s_i(x, y), (x, y) \in \Omega_r\}$  et où  $\mathcal{L}_e[\Omega_r^i | \hat{\theta}_r^i]$  est la log-vraisemblance généralisée de la région  $\Omega_r$  calculée sur la composante  $i$ . Dans cette expression nous avons remplacé les paramètres  $\theta_r^i$  des ddp par leurs estimées  $\hat{\theta}_r^i$  au sens du MV, de la même façon que précédemment.

De même, la taille  $\alpha$  du vecteur de paramètre  $\theta_r$  de la ddp  $P_{\theta_r}$  s'obtient facilement à partir de la taille  $\alpha_i$  du paramètre  $\theta_r^i$  de chacune des composantes :

$$\alpha = \sum_{i=1}^M \alpha_i, \quad (4.5)$$

En introduisant (4.4) et (4.5) dans l'équation 2.16 donnant l'expression de la complexité stochastique dans le cas général, nous obtenons l'équation suivante, adaptée aux images vectorielles dont les différentes composantes sont indépendantes :

$$\begin{aligned} \Delta = & n(\log N + \log p) + \log p + p(2 + \log(2\widehat{m}_x) + \log(2\widehat{m}_y)) \\ & + \sum_{r=1}^R \left[ \left( \sum_{i=1}^M \frac{\alpha_i}{2} \right) \log N_r - \sum_{i=1}^M \mathcal{L}_e[\Omega_r^i | \hat{\theta}_r^i] \right] \end{aligned} \quad (4.6)$$

Dans cette équation, pour chaque composante  $i$ ,  $\alpha_i$  et  $\mathcal{L}_e[\Omega_r^i | \hat{\theta}_r^i]$  sont définies à partir de ddp scalaires et leurs expressions sont fournies dans le tableau 4.2 du chapitre précédent.

Dans les deux parties suivantes, deux exemples d'images vectorielles avec des composantes indépendantes vont être étudiés. Dans le premier cas, les ddp appartiennent à la même famille quelle que soit la composante considérée (ce seront par exemple toutes des ddp de Poisson). En revanche, les paramètres de ces ddp dans une région  $\Omega_r$  pourront être différents d'une composante à l'autre, c'est-à-dire que l'on pourra avoir  $\theta_r^i \neq \theta_r^j$  si  $i \neq j$ . C'est notamment le cas des images multi-dates, multi-spectrales ou plus généralement des images constituées de plusieurs prises de vues. Dans la suite, nous illustrerons ce cas essentiellement sur des images SAR multi-dates.

Nous présenterons ensuite, très succinctement, un second cas où les ddp appartiendront à des familles différentes pour chacune des composantes. De telles images peuvent être obtenues en utilisant différents capteurs pour observer la même scène.

### 4.2.2 Images de type multi-dates

Une image multi-dates est obtenue en observant la même scène à différentes dates. Par exemple, si un capteur prend la photo d'un même paysage à chaque saison, l'image vectorielle associée sera composée de 4 images scalaires, une pour chaque saison. Entre ces 4 composantes, la végétation aura changée : les réflectivités ne seront plus nécessairement identiques sur toutes les composantes. Ainsi, dans chaque région de l'image, toutes les composantes seront distribuées selon des ddp appartenant à la même famille (par exemple gaussienne) puisqu'elles ont été acquises par le même capteur, mais qui pourront avoir des paramètres différents selon la composante considérée.

Dans un premier temps, nous allons présenter quelques résultats sur des images synthétiques.

#### Images synthétiques

Dans cette partie, nous allons comparer les résultats obtenus sur des images de type multi-dates en utilisant :

**cas 1** : soit l'image vectorielle,

**cas 2** : soit une seule des  $M$  composantes,

**cas 3** : ou soit la somme des  $M$  composantes de l'image vectorielle.

Bien sûr, lorsque nous travaillerons sur une seule composante, comme c'est le cas avec les approches (cas 2) et (cas 3), nous utiliserons la complexité stochastique habituelle définie au chapitre 2 par l'équation 2.16, alors que nous utiliserons l'équation 4.6 lorsque nous travaillerons sur l'image vectorielle avec l'approche (cas 1).

L'image multi-dates de la fig. 4.14 est constituée de 6 composantes qui ont chacune été générées à partir d'une image synthétique de type zone agricole et composée de  $256 \times 256$  pixels et de 14 régions. Plus précisément, chacune des 6 composantes a été obtenue en tirant aléatoirement la réflectivité  $\rho$  parmi 4 valeurs :  $\rho \in \{1, 2, 3, 4\}$ . Les 6 images de réflectivités obtenues sont ainsi présentées sur la ligne 1. Ces 6 images sont ensuite bruitées par un bruit gamma d'ordre  $L = 1$ , conduisant aux 6 composantes de la ligne 2. Nous présentons sur la troisième ligne les partitions obtenues en utilisant chacune des 6 composantes de la ligne 2 séparément. Sur la ligne 4 (à gauche), nous avons représenté la somme des 6 composantes bruitées et la partition obtenue à partir de cette image somme<sup>18</sup>. La partition obtenue en utilisant une approche vectorielle sur les 6 composantes simultanément est présentée sur la quatrième ligne au centre.

Nous allons maintenant étudier une deuxième image synthétique constituée de 3 composantes perturbées par un bruit de Bernoulli (fig. 4.15.a, 4.15.b et 4.15.c). La partition vraie est affichée sur la fig. 4.15.d. Soulignons de plus que les moyennes dans chacune des régions ont été choisies de telle sorte que leur somme soit constante sur toute l'image : la somme des trois dates ne contient donc aucune information utile pour son partitionnement en régions homogènes. La fig. 4.15.f présente le partition obtenu en utilisant une approche vectorielle sur les 3 composantes simultanément, alors que les résultats avec une approche scalaire sur chacune des 3 composantes sont donnés sur les fig. 4.15.g, 4.15.h et 4.15.i.

Dans ces deux exemples, seule l'approche vectorielle a permis de partitionner correctement toutes les régions.

## Images SAR multi-dates

Illustrons maintenant cette approche sur des images SAR multi-dates. Les résultats sont présentés sur les fig. 4.16 et 4.17 et montrent l'intérêt d'utiliser une approche vectorielle sur ce type d'images. Cela permet en effet de combiner simultanément les avantages de chacune des dates, sans pour autant détériorer le contraste entre régions par effets de moyennage, comme c'est le cas avec une approche scalaire sur la somme des composantes.

## Etude du cas sous-optimal

Dans tous les exemples présentés précédemment, les vecteurs de paramètres étaient différents sur chaque composante. L'approche vectorielle était donc adaptée à ce type d'images. Dans cette partie, nous nous proposons de nous placer dans le cas où l'approche basée sur la somme est maintenant optimale au sens du MV, c'est-à-dire lorsque les niveaux de gris dans chaque région ont la même moyenne quelle que soit la composante. Nous pourrons ainsi déterminer dans quelle mesure le fait de considérer que les paramètres de chaque canal sont

18. Notons que l'image somme est obtenue en sommant les 6 dates de la ligne 2. Ces 6 dates étant perturbées par un bruit gamma d'ordre  $L = 1$ , la somme de ces 6 dates présente donc un bruit gamma d'ordre  $L = 6$ .

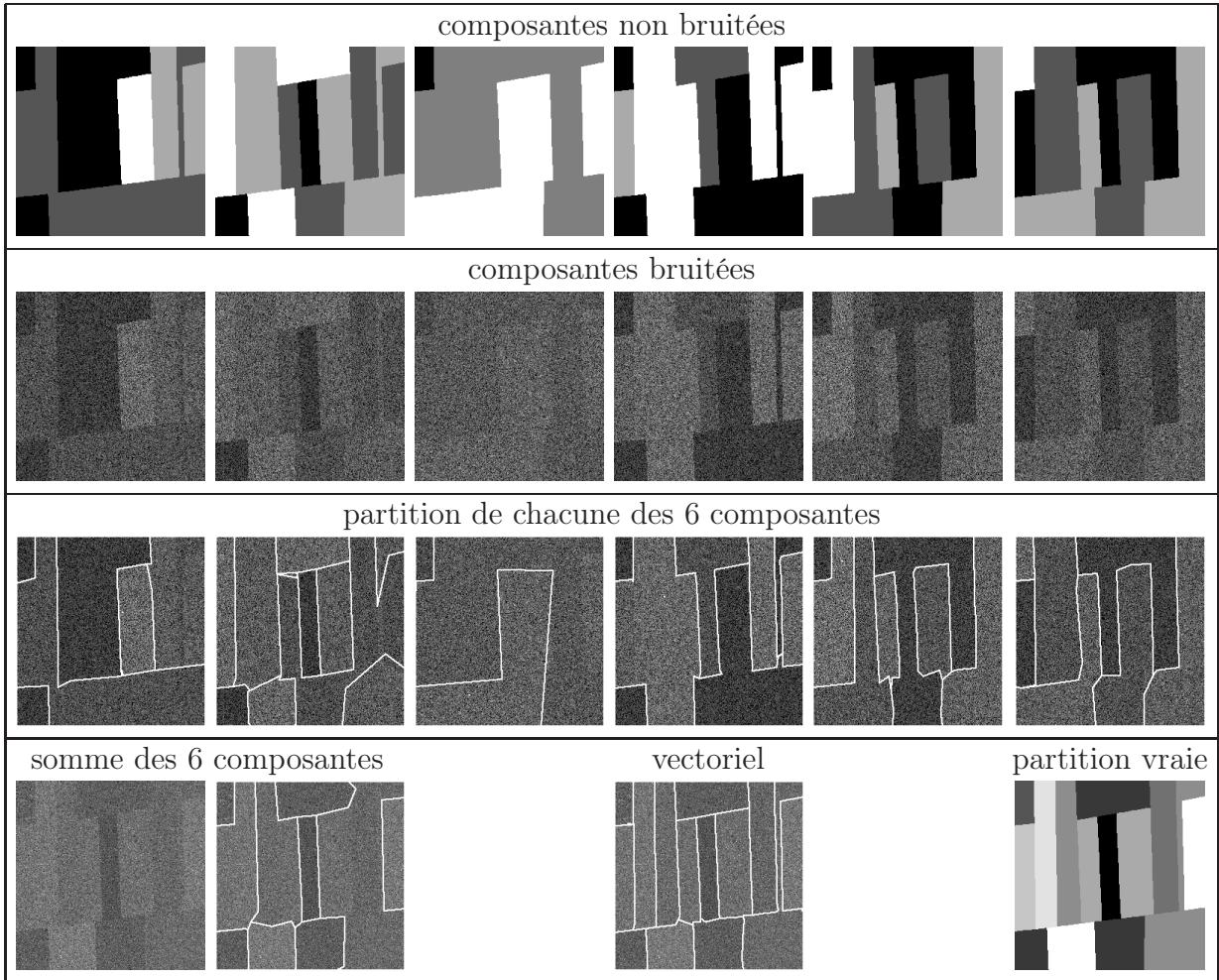


FIGURE 4.14 – Partition d'une image synthétique à 6 composantes ( $256 \times 256$  pixels) de type parcellaire agricole et perturbée par un bruit de speckle d'ordre  $L = 1$  : comparaisons des approches scalaires et vectorielles (voir texte pour plus de détails). Ligne 1 : 6 composantes non bruitées - Ligne 2 : 6 composantes bruitées - Ligne 3 : Partition de chacune des 6 composantes avec une approche scalaire - Ligne 4 (gauche) : Somme des 6 composantes de la ligne 2 et partition de la somme des 6 composantes avec une approche scalaire - Ligne 4 (centre) : Partition obtenue avec une approche vectorielle sur les 6 composantes de la ligne 2. Ligne 4 (droite) : Partition vraie utilisée pour générer les 6 images de la ligne 1. Les niveaux de gris des images ont été modifiés pour une meilleure visualisation. De plus, les résultats de la dernière ligne sont affichés sur la somme des 6 composantes, alors que les résultats de la ligne 3 sont affichés sur les images de la ligne 2. Grilles initiales : murs de briques de taille 6.

différents, alors qu'ils sont en réalité identiques, dégrade les performances.

Dans le premier cas, nous supposerons que les vecteurs de paramètres  $\theta_r^i$ ,  $i \in \{1, 2, \dots, M\}$ , sur une région  $\Omega_r$  donnée sont égaux pour les différentes composantes :

$$\theta_r^i = \theta_r \quad \forall i \in \{1, 2, \dots, M\} \quad (4.7)$$

Les ddp de chaque composante sont ainsi définies par un vecteur de paramètre  $\theta_r$  unique, commun à toutes les ddp. Il est possible de montrer que dans le cas de ddp gamma d'ordre  $L$ ,

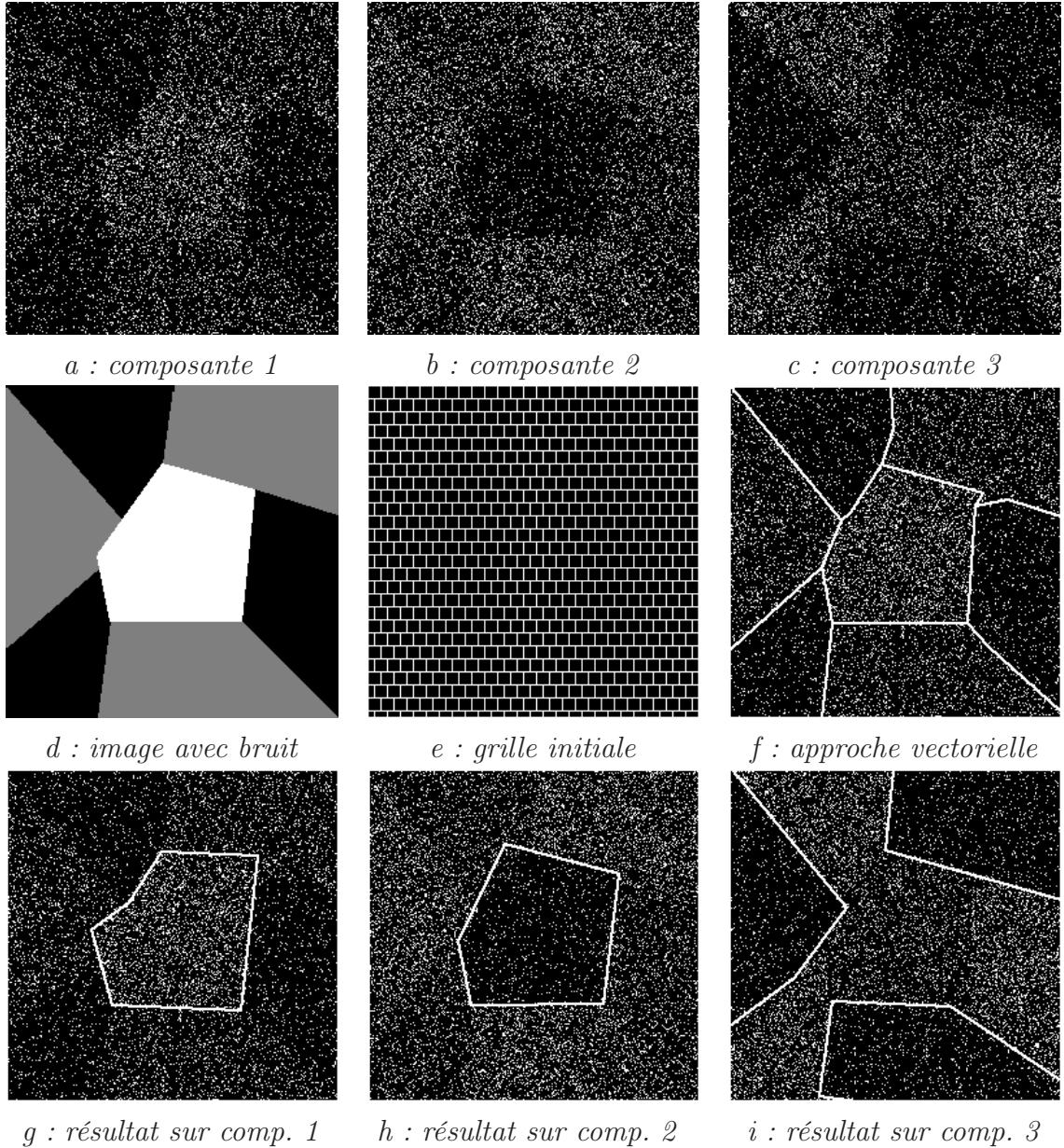


FIGURE 4.15 – Partition d’une image synthétique de 3 composantes ( $256 \times 256$  pixels) perturbées par un bruit de Bernoulli : comparaison des approches scalaires et vectorielles (voir texte pour plus d’explications). Les résultats de partition (g),(h),(i) ont été superposés respectivement sur (a),(b),(c), et le résultat (f) a été superposé sur (a).

l’estimée du paramètre  $\theta_r$  au sens du MV dans une région  $\Omega_r$  s’écrit [35] :

$$\hat{\theta}_r = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} \left[ \sum_{i=1}^M s_i(x, y) \right] \quad (4.8)$$

La log-vraisemblance généralisée de cette région  $\Omega_r$  devient alors :

$$\mathcal{L}_e [\Omega_r | \hat{\theta}_r] = M \times L N_r \log \left( \hat{\theta}_r \right) + K \quad (4.9)$$

où  $K$  est une constante indépendante de la partition. Nous voyons donc que la solution obtenue revient à sommer les  $M$  composantes de l’image et à partitionner l’image somme obtenue

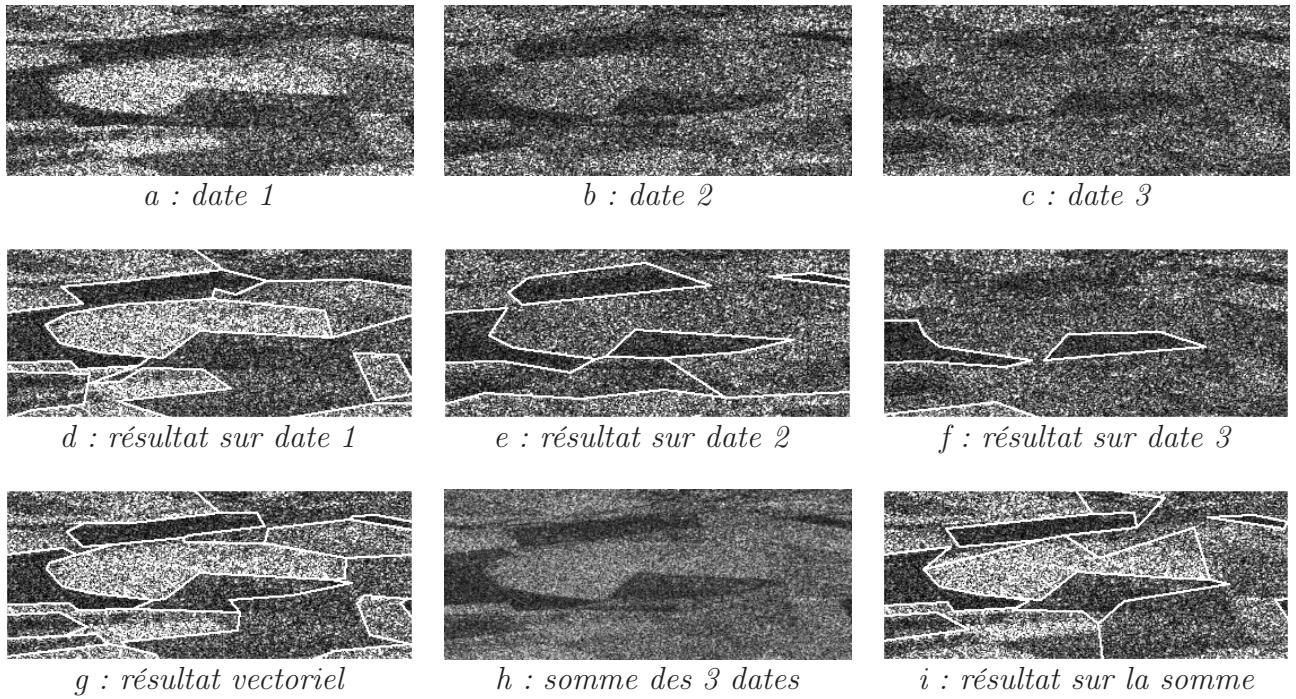


FIGURE 4.16 – Partition d'une image SAR 3-dates ( $351 \times 151$  pixels) obtenue par le satellite ERS-1 (distribuée par l'ESA et fournie par le CNES). Les 3 dates (a), (b), (c) sont des images SAR mono-vue de parcelles agricoles près de Bourges (France). La partition de chacune de ces 3 dates avec une approche scalaire est présentée en (d), (e), (f). Le résultat obtenu avec une approche vectorielle sur les 3 dates est reporté en (g). Finalement, la partition de la somme de ces 3 dates (voir h) avec une approche scalaire est présentée en (i). Les résultats (g) et (i) ont été superposés sur la première date.

$$\tilde{s}(x, y) = \sum_{i=1}^M s_i(x, y) \text{ avec un critère gamma d'ordre } M \times L.$$

Dans le deuxième cas, nous supposerons au contraire que  $\theta_r^i$  et  $\theta_r^j$  sont totalement indépendants, c'est-à-dire que la connaissance de  $\theta_r^i$  n'apporte aucune information sur  $\theta_r^j$  ( $i \neq j$ ). Dans le calcul de la log-vraisemblance généralisée,  $\theta_r^i$  et  $\theta_r^j$  seront donc estimés au sens du MV de manière indépendante. Nous pouvons donc utiliser l'expression de la complexité stochastique donnée dans l'équation 4.6.

Considérons une image synthétique à  $M = 4$  composantes ( $256 \times 256$  pixels) composée de 10 régions (fig. 4.18.a) : les régions sombres dont les niveaux de gris sont égaux à 1 et les régions claires dont les niveaux sont égaux à  $c$ . Cette image est alors bruitée (fig. 4.18.b) avec un bruit gamma d'ordre  $L = 1$  pour chaque composante :  $c$  représente ainsi le contraste entre chaque régions voisines<sup>19</sup>.

Analysons les performances obtenues avec chacune des trois approches suivantes : partition sur l'image vectorielle, sur une composante ou sur la somme des composantes, lorsque le contraste  $c$  varie ( $c$  étant bien sûr le même sur chaque composante). Le critère retenu pour cette analyse est le nombre de régions détectées. Ce nombre est égal à 10 quand toutes les régions ont

19. Rappelons qu'en présence de ddp gamma d'ordre  $L$ , le contraste entre deux régions se définit comme le rapport des moyennes.

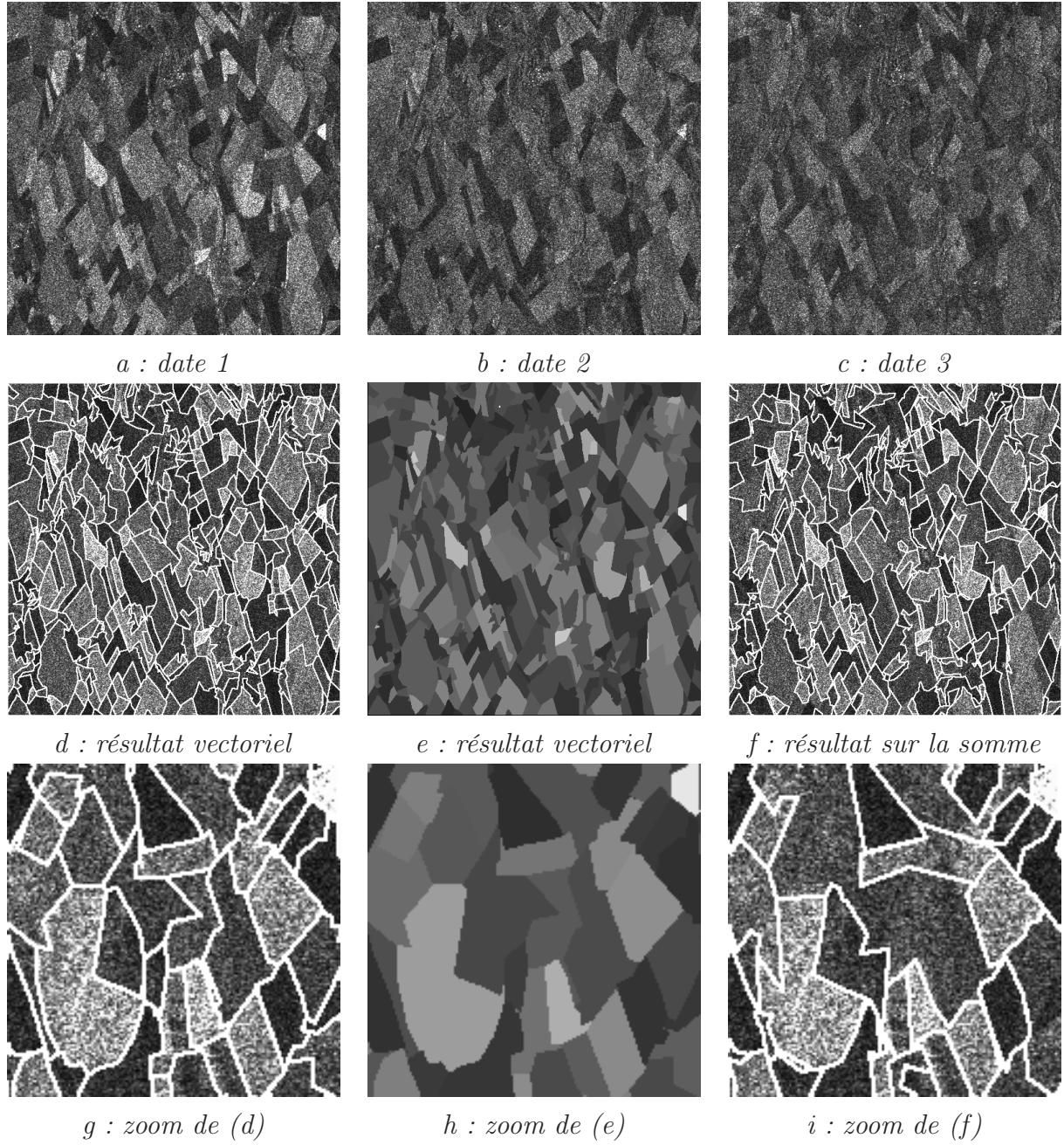


FIGURE 4.17 – Partition d'une image SAR 3-dates ( $512 \times 512$  pixels) de parcelles agricoles près de Bourges (France) obtenue par le satellite ERS-1 (distribuée par l'ESA et fournie par le CNES). (a),(b),(c) : images des 3 dates. (d),(e) : partition vectorielle des 3 dates (sur la fig.(e), les niveaux de gris de chaque région ont été remplacés par leur valeur moyenne sur la première date). (f) : partition scalaire appliquée sur la somme des 3 dates. Sur les fig.(d),(f), la grille est affichée sur la première date. (g),(h),(i) : zoom de (d),(e),(f).

été détectées et est égal à 1 quand aucune région n'a été détectée, l'image étant alors composée d'une seule et même région. Nous présentons sur la fig. 4.18.c les résultats obtenus moyennés sur 20 réalisations. Nous pouvons constater que lorsque l'approche scalaire sur la somme est optimale au sens du MV, comme c'est le cas ici, les performances de l'approche vectorielle restent comparables. Plus précisément, la différence entre ces 2 approches n'apparaît qu'aux

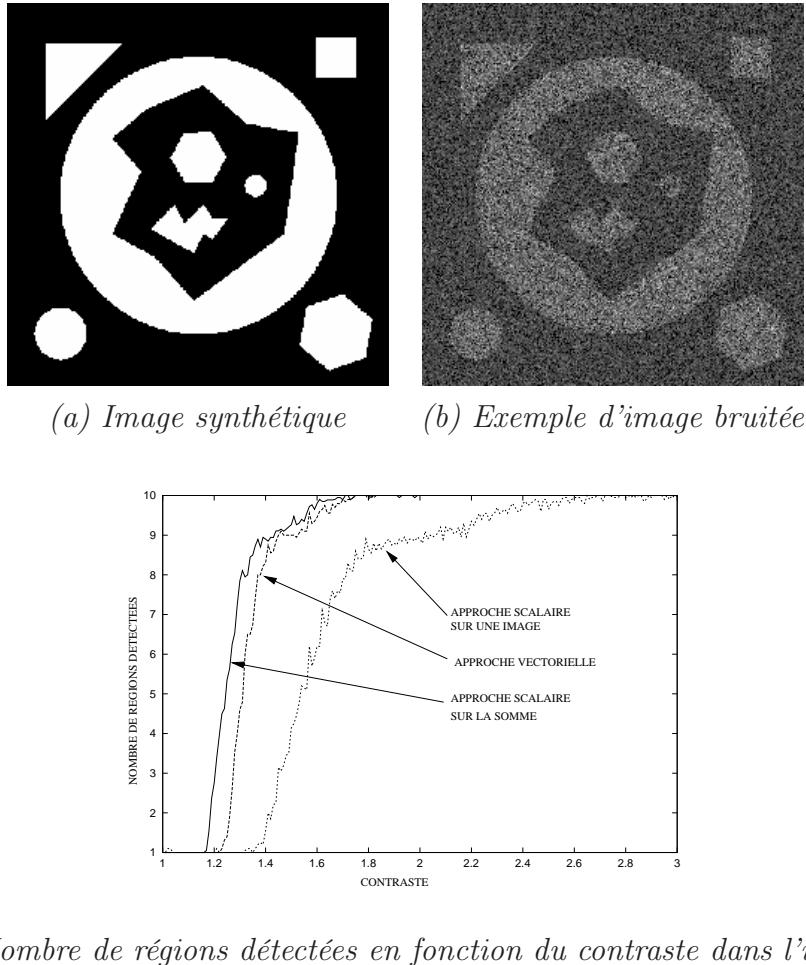


FIGURE 4.18 – Comparaison des performances de l’approche scalaire sur la somme, vectorielle et scalaire sur une seule composante, lorsque l’approche scalaire sur la somme est optimale. Voir texte pour plus d’explications. Le contraste sur (b) a été modifié pour une meilleure visualisation.

très faibles contrastes ( $c < 1.8$ ), c'est-à-dire lorsque les partitions obtenues sont de mauvaise qualité, même avec l'approche scalaire sur la somme.

En conclusion, même en cas d'optimalité de l'approche scalaire sur la somme, l'approche vectorielle fournit des résultats intéressants, alors que les performances de l'approche scalaire sur la somme se dégradent rapidement dès que les paramètres des ddp diffèrent entre chaque composante. Ainsi, lorsque l'on ne peut être sûr que les paramètres des ddp sont identiques entre chaque composante, l'approche vectorielle, plus robuste, est donc préférable [33].

#### 4.2.3 Images dont les composantes possèdent des ddp de familles différentes

Considérons maintenant différentes images de la même scène dont les composantes sont perturbées par des bruits de lois de probabilité différentes.

Nous allons considérer une image synthétique (fig. 4.19.a) dont la première composante est perturbée par un bruit gamma et la deuxième par un bruit de Poisson. Afin de simplifier l'analyse, les réflectivités (c'est-à-dire les moyennes) sont prises identiques pour les deux

composantes. Les deux composantes de l'image ne diffèrent donc que par leur famille de ddp (gamma pour la composante 1, Poisson pour la composante 2) et non par la moyenne des niveaux de gris.

Les deux composantes, perturbées respectivement par un bruit gamma d'ordre  $L = 1$  et par un bruit de Poisson sont représentées sur les fig. 4.19.b et fig. 4.19.c. La partition obtenue avec une approche scalaire sur la fig. 4.19.b (respectivement 4.19.c) en utilisant un critère gamma d'ordre  $L = 1$  (respectivement Poisson) est présentée sur la fig. 4.19.d (respectivement 4.19.e). Nous pouvons constater que si de bons résultats sont obtenus sur la partie inférieure de l'image de la fig. 4.19.d, ceux obtenus sur la partie supérieure ne sont pas satisfaisants, alors que nous obtenons des conclusions inverses sur la fig. 4.19.e. Ce résultat illustre le fait que le contraste entre deux régions dépend non seulement de leur moyenne respective mais aussi de la famille de la ddp. L'utilisation d'une approche vectorielle avec une ddp gamma d'ordre  $L = 1$  pour la première composante et une ddp de Poisson pour la deuxième permet alors d'exploiter de manière efficace l'information contenue dans les fig. 4.19.b et 4.19.c, comme on peut le voir sur la fig. 4.19.f.

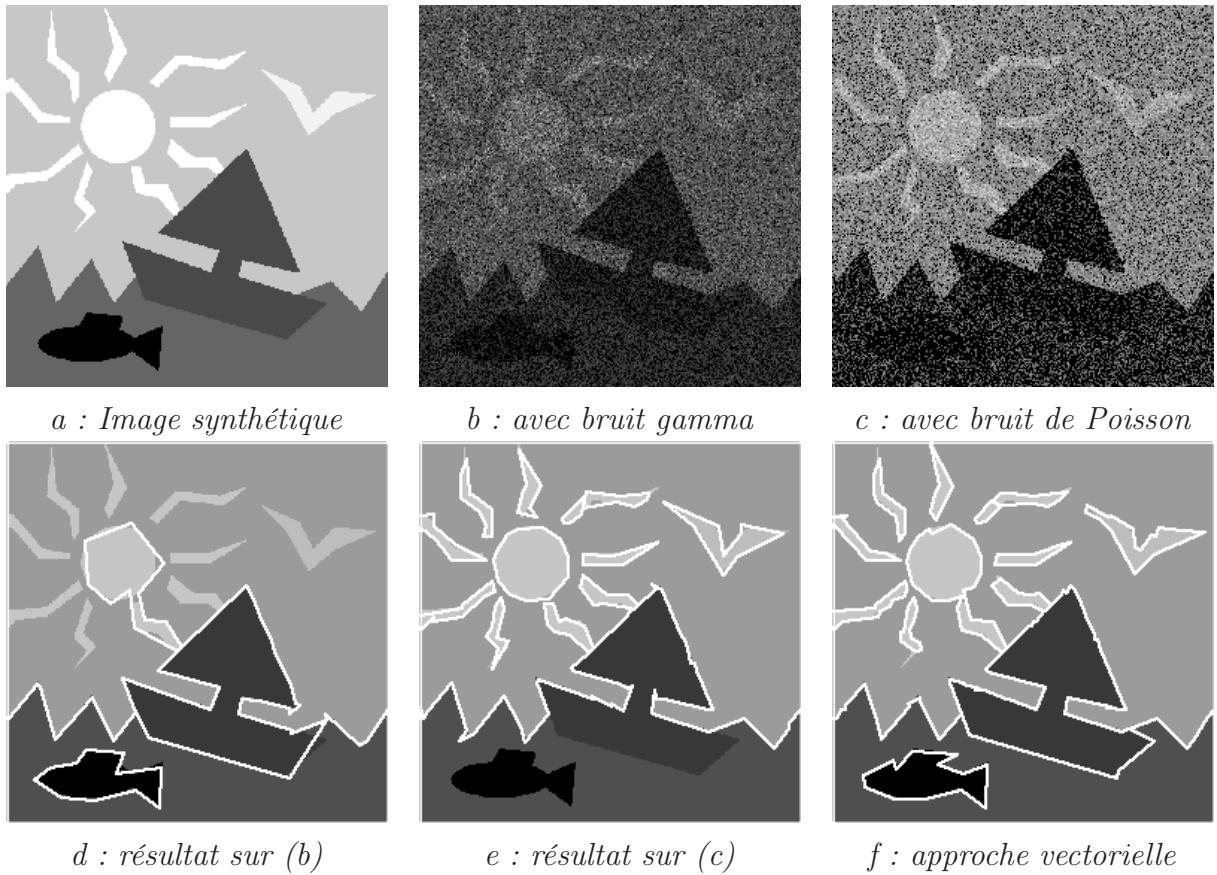


FIGURE 4.19 – Partition d'une image synthétique vectorielle ( $256 \times 256$  pixels) (a) et perturbée par un bruit gamma d'ordre  $L = 1$  sur sa première composante (b) et un bruit de Poisson sur sa seconde composante (c). La partition de l'image (b) (respectivement c) avec une approche scalaire est présentée en (d) (respectivement e) alors que celle obtenue avec une approche vectorielle sur (b) et (c) simultanément apparaît en (f). Les images bruitées ont été modifiées pour une meilleure visualisation. De plus les résultats (d), (e) et (f) ont été affichés sur l'image non bruitée (a).

#### 4.2.4 Images SAR polarimétriques

Jusqu'à présent, nous nous sommes intéressés uniquement à des images vectorielles dont les composantes sont décorrélées, c'est pourquoi, dans cette partie, nous allons étudier le cas des images SAR polarimétriques qui constituent un bon exemple d'images vectorielles de composantes corrélées.

L'état de polarisation des ondes électromagnétiques contient des informations importantes. L'imagerie polarimétrique active consiste à illuminer une scène et à former une image de l'état de polarisation de l'onde rétro-diffusée. Même si les images d'état de polarisation que nous allons présenter sont toutes de type SAR, c'est-à-dire obtenues en utilisant une illumination micro-ondes, le formalisme que nous allons développer peut s'appliquer à des images optiques obtenues par illumination à l'aide de lasers. Que l'illumination soit de type laser ou radar, l'utilisation d'une source cohérente génère un phénomène de speckle sur l'image obtenue. Comme dans le cas de l'imagerie SAR classique (non polarimétrique), la présence de ce bruit de speckle complique le traitement de ces images et l'utilisation d'algorithmes adaptés est nécessaire [58].

Nous allons maintenant présenter un modèle statistique des images polarimétriques cohérentes, afin de pouvoir calculer leur complexité stochastique.

#### Modèle statistique des images polarimétriques cohérentes

Le champ électrique  $\mathbf{E} = (E_1, E_2)^T$  d'une radiation électromagnétique cohérente est classiquement représenté par un vecteur aléatoire 2D gaussien circulaire [42] dont la ddp est la suivante :

$$P_{\Gamma}(\mathbf{E}) = \frac{1}{\pi^2 \det \Gamma} \exp [-\mathbf{E}^\dagger \Gamma^{-1} \mathbf{E}] \quad (4.10)$$

où le symbole  $\dagger$  correspond au complexe conjugué et où  $\Gamma$  est la matrice de covariance :

$$\Gamma = \begin{bmatrix} \langle |E_1|^2 \rangle & \langle E_1 E_2^* \rangle \\ \langle E_2 E_1^* \rangle & \langle |E_2|^2 \rangle \end{bmatrix} = \begin{bmatrix} \mu_1 & \delta \\ \delta^* & \mu_2 \end{bmatrix} \quad (4.11)$$

Le symbole  $\langle . \rangle$  représente la moyenne d'ensemble. La matrice de covariance  $\Gamma$  est communément appelée *matrice de cohérence* dans le domaine optique. Elle définit l'état de polarisation de l'onde [42]. Cette matrice étant hermitienne, elle est entièrement définie par 4 paramètres scalaires réels : les intensités moyennes  $\mu_1$  et  $\mu_2$  ainsi que les parties réelle et imaginaire du coefficient de corrélation complexe  $\delta$ .

A partir de ce modèle d'image, nous allons déterminer la complexité stochastique d'une image polarimétrique active, afin d'appliquer l'algorithme de partition en régions homogènes décrit dans le chapitre 2.

#### Détermination de la complexité stochastique

L'objectif est toujours de partitionner l'image en régions homogènes, c'est-à-dire en régions où les paramètres de la ddp du champ électrique  $\mathbf{E}$  (équation 4.10) sont identiques.

Le terme de codage de la partition  $\Delta_G$  reste identique à celui fourni dans l'équation 2.7 du chapitre 2. Quant au terme de codage des paramètres de la ddp,  $\Delta_P$ , il est lui aussi toujours fourni par l'équation 2.8 : il suffit simplement de déterminer la dimension  $\alpha$  du vecteur de

paramètre de la ddp du champ électrique  $\mathbf{E}$ . Le seul paramètre de cette ddp étant la matrice de cohérence  $\Gamma$ , elle-même constituée de 4 paramètres scalaires réels (équation 4.11), nous avons donc  $\alpha = 4$ .

Il reste maintenant à déterminer le terme de codage des niveaux de gris connaissant le modèle,  $\Delta_L$ , c'est-à-dire l'opposé de la log-vraisemblance généralisée. Notons  $\Gamma_r$  la matrice de cohérence dans la région  $\Omega_r$  et calculons la log-vraisemblance du champ électrique  $\mathbf{E}$  dans cette région :

$$\mathcal{L}_e[\Omega_r, \Gamma_r] = -N_r \log \pi^2 - N_r \log[\det \Gamma_r] - \sum_{(x,y) \in \Omega_r} \mathbf{E}(x,y)^\dagger \Gamma_r^{-1} \mathbf{E}(x,y) \quad (4.12)$$

La matrice de cohérence  $\Gamma_r$  étant la plupart du temps inconnue, nous devons l'estimer. Un calcul classique, bien que non trivial, montre que l'estimée au sens du Maximum de Vraisemblance (MV) de la matrice de cohérence est en fait égale à la matrice de covariance empirique [66] :

$$\widehat{\Gamma}_r = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} \mathbf{E}(x,y) \mathbf{E}(x,y)^\dagger \quad (4.13)$$

En reportant cette estimée  $\widehat{\Gamma}_r$  dans la log-vraisemblance de l'équation 4.12, nous obtenons la log-vraisemblance généralisée suivante :

$$\mathcal{L}_e(\Omega_r, \widehat{\Gamma}_r) = -N_r \log[\det \widehat{\Gamma}_r] - 2N_r(1 + \log \pi) \quad (4.14)$$

Le terme de codage des niveaux de gris s'écrit donc :

$$\begin{aligned} \Delta_L &= \sum_{r=1}^R -\mathcal{L}_e(\Omega_r, \widehat{\Gamma}_r) \\ &= \sum_{r=1}^R N_r \log[\det \widehat{\Gamma}_r] + 2N(1 + \log \pi) \end{aligned} \quad (4.15)$$

Dans cette équation, le second terme  $2N(1 + \log \pi)$  est indépendant de la partition.

De plus, il est facile de montrer que la ddp du champ  $\mathbf{E}$  appartient à la famille exponentielle et que la statistique  $\sum_{(x,y) \in \Omega_r} \mathbf{E}(x,y) \mathbf{E}(x,y)^\dagger$  intervenant dans le calcul de  $\widehat{\Gamma}_r$  (équation 4.13) est bien une statistique exhaustive. Cette statistique exhaustive, écrite sous forme matricielle, se décompose en 4 statistiques exhaustives scalaires :

$$\begin{aligned} T_1(\Omega_r) &= \sum_{(x,y) \in \Omega_r} |E_1(x,y)|^2 \\ T_2(\Omega_r) &= \sum_{(x,y) \in \Omega_r} |E_2(x,y)|^2 \\ T_3(\Omega_r) &= \sum_{(x,y) \in \Omega_r} \operatorname{Re}[E_1(x,y) E_2(x,y)^*] \\ T_4(\Omega_r) &= \sum_{(x,y) \in \Omega_r} \operatorname{Im}[E_1(x,y) E_2(x,y)^*] \end{aligned} \quad (4.16)$$

où l'opérateur  $*$  est l'opérateur de conjugaison. La matrice de cohérence estimée  $\widehat{\Gamma}_r$  s'écrit alors simplement à l'aide de ces 4 statistiques :

$$\widehat{\Gamma}_r = \frac{1}{N_r} \begin{bmatrix} T_1(\Omega_r) & T_3(\Omega_r) + \mathbf{i} T_4(\Omega_r) \\ T_3(\Omega_r) - \mathbf{i} T_4(\Omega_r) & T_2(\Omega_r) \end{bmatrix} \quad (4.17)$$

Afin de pouvoir appliquer l'algorithme rapide présenté en partie 2.2.5, il suffit donc de travailler sur les 4 images quadratiques suivantes : deux images d'intensité  $|E_1(x,y)|^2$  et  $|E_2(x,y)|^2$ , et deux termes croisés  $\operatorname{Re}[E_1(x,y) E_2(x,y)^*]$  et  $\operatorname{Im}[E_1(x,y) E_2(x,y)^*]$ .

Nous allons donc présenter maintenant quelques résultats de partition obtenus d'abord sur une image polarimétrique synthétique puis une image SAR polarimétrique réelle.

## Résultats

Dans un premier temps, nous avons générée une image synthétique (fig. 4.20) que nous avons partitionnée en régions homogènes en utilisant deux approches : la première où l'on utilise le critère adapté au cas polarimétrique (équation 4.15) et qui prend donc en compte l'information contenue dans les 4 composantes polarimétriques (fig. 4.20, ligne 1), et une deuxième approche où seule l'intensité de l'image polarimétrique, c'est-à-dire  $|E_1|^2 + |E_2|^2$ , est utilisée. Notons que dans cette deuxième approche, nous devons partitionner une image active d'intensité : nous avons donc utilisé un critère adapté à un bruit gamma d'ordre  $L = 1$ .

Nous présentons enfin sur la fig. 4.21 la partition d'une image SAR polarimétrique réelle de la baie de San Francisco.

Comme nous avons pu le voir sur ces quelques exemples, l'utilisation de capteurs SAR polarimétriques permet de révéler des caractéristiques auxquelles les imageurs SAR classiques ne donnent pas accès, à condition bien sûr d'utiliser un modèle adapté à ce type d'images vectorielles dont les composantes sont corrélées.

### 4.2.5 Conclusion

Au cours de cette partie, nous avons montré que la généralisation du modèle d'images scalaires exposé dans le chapitre 2 à un modèle vectoriel se faisait naturellement, que les composantes soient indépendantes entre elles ou non. De plus, il est clairement apparu que l'utilisation d'une approche vectorielle permettait de tirer parti de l'information contenue dans chacune des composantes, contrairement aux approches scalaires. Soulignons également, dans le cas de l'imagerie SAR multi-dates, la robustesse de l'approche vectorielle par rapport à l'approche consistant à travailler directement sur la somme des différentes composantes, c'est-à-dire une image scalaire. En effet, même en cas d'optimalité de cette dernière, l'approche vectorielle reste comparable en termes de performance, alors que la qualité de la partition obtenue avec une approche scalaire sur la somme se dégrade rapidement dès lors que les réflectivités ne sont plus identiques sur toutes les composantes.

La partie suivante, consacrée aux contours actifs, nous permettra de fournir quelques autres exemples d'utilisation d'images vectorielles, notamment des images optiques en couleurs.

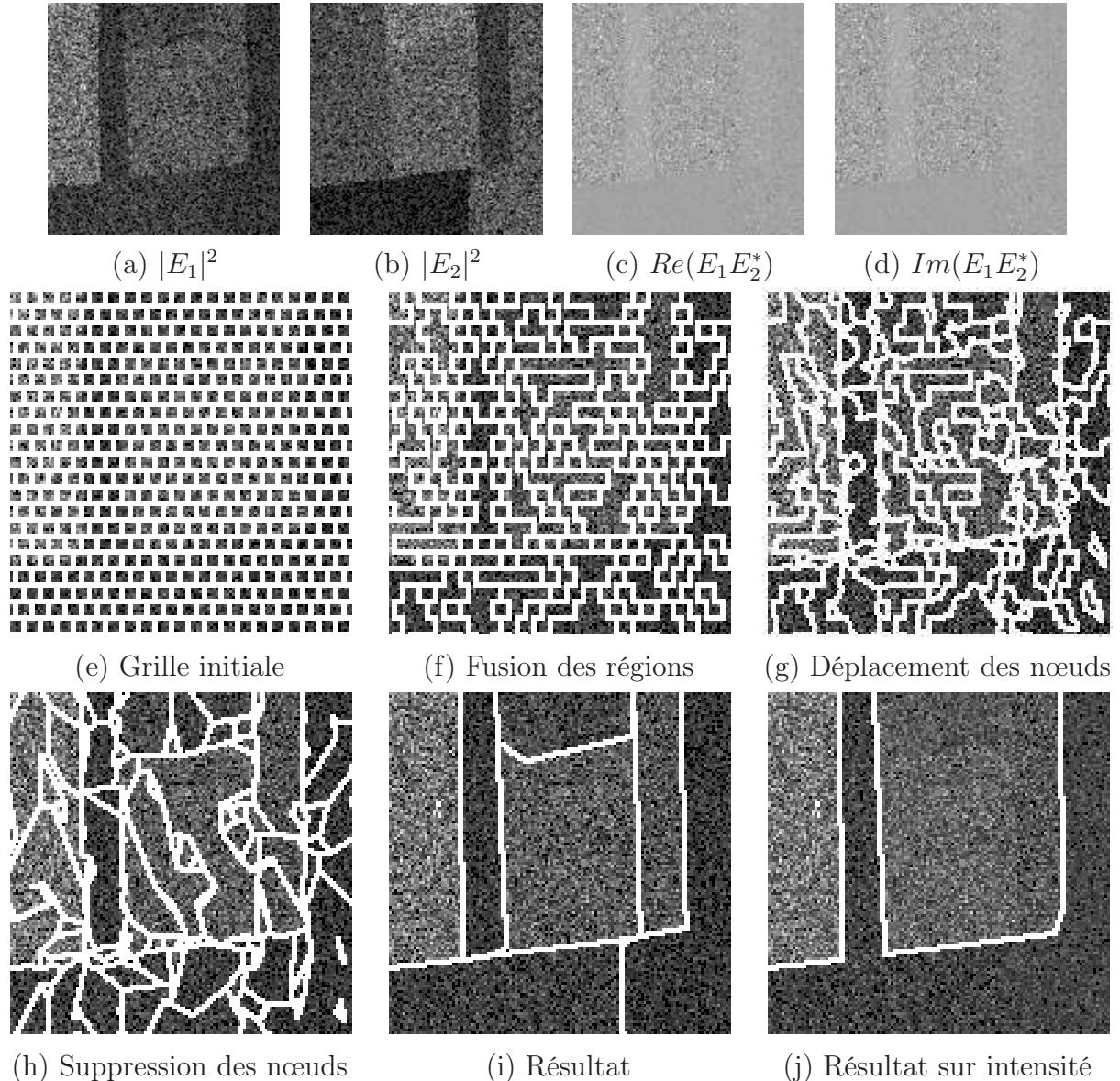


FIGURE 4.20 – Partition d'une image polarimétrique active synthétique ( $256 \times 256$  pixels). (a,b,c,d) image synthétique. (e,f,g,h,i) différentes étapes de la partition de l'image en utilisant la complexité stochastique adaptée aux images polarimétriques actives : (e) grille initiale - (f) partition après fusion de régions (GLRT,  $\tilde{S} = 3$ ) - (g) après déplacement des nœuds - (h) après suppression des nœuds inutiles - (i) résultat après avoir effectué un autre cycle de fusion (MCS), déplacement, suppression. (j) Résultat obtenu uniquement sur l'image d'intensité (c'est-à-dire  $|E_1|^2 + |E_2|^2$ ) en utilisant un critère gamma d'ordre  $L = 1$ . Tous les résultats sont affichés sur l'image d'intensité  $|E_1|^2 + |E_2|^2$ .

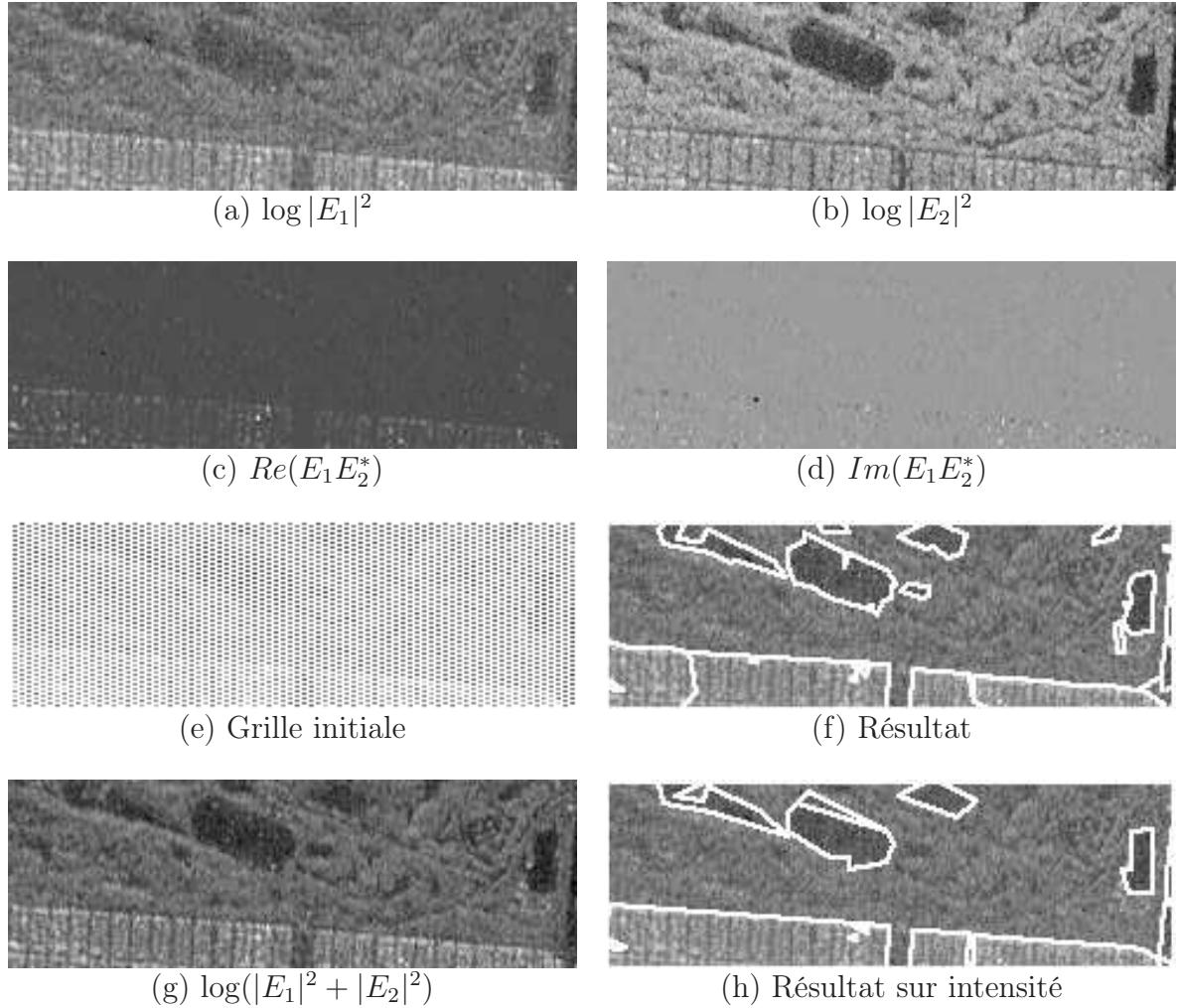


FIGURE 4.21 – Partition d'un extrait ( $80 \times 242$  pixels) d'une image SAR polarimétrique réelle de la baie de San Francisco, acquise par le capteur AIRSAR de la NASA/JPL. (a,b,c,d) Image SAR. (e) Grille initiale - (f) Résultat de la partition en régions homogènes de l'image polarimétrique - (g) Image d'intensité  $|E_1|^2 + |E_2|^2$ . (h) Résultat obtenu uniquement sur l'image d'intensité (g) en utilisant un critère gamma d'ordre  $L = 1$ . Nous avons pris le logarithme des images (a), (b) et (g) afin de permettre une meilleure visualisation. De plus, tous les résultats sont affichés sur l'image (g).

## 4.3 Segmentation par contours actifs multi-régions

Dans tous les cas analysés précédemment, nous avons supposé que le nombre de régions et leur topologie étaient inconnus. Il existe cependant de nombreuses applications, telles que la poursuite de cibles ou la reconnaissance de formes, pour lesquelles ces informations sont connues. L'objectif est alors uniquement de retrouver la forme de ces régions. Dans ce cas, l'utilisation de contours actifs polygonaux multi-régions peut s'avérer intéressante.

Tout d'abord, cela permet d'introduire les informations *a priori* intéressantes que sont le nombre de régions et leurs relations topologiques. Comme nous l'avons déjà souligné à plusieurs reprises, la partition en régions homogènes ne correspond en général pas à la segmentation des objets présents dans l'image. Cette notion d'objet correspond le plus souvent à une information de haut niveau. Les contours actifs multi-régions permettent d'introduire des informations *a priori* correspondant à des caractéristiques de haut niveau. Le second avantage est pragmatique. Il correspond à la réduction du temps de calcul par rapport à l'utilisation d'une grille initiale complexe comme celles utilisées jusqu'à présent.

Dans cette partie, nous allons expliquer comment l'algorithme de partition par grille active peut être utilisé, avec très peu de modifications, comme algorithme de segmentation par contour actif multi-régions.

### 4.3.1 Complexité stochastique

Un contour actif polygonal multi-régions est simplement une grille active polygonale particulière dont le nombre de régions et la topologie sont connus et fixés.

Nous différencierons dans la suite deux types de contours actifs : les contours actifs monorégions, composés d'une seule région cible (c'est-à-dire de deux régions au total, la cible et le fond) comme c'est le cas par exemple dans les travaux de [16, 26, 82], et les contours actifs multi-régions, qui sont eux composés de plusieurs régions cibles et peuvent donc avoir des topologies complexes [38].

La convergence du contour actif repose toujours sur la minimisation de la complexité stochastique. Il est donc possible d'utiliser l'expression déterminée pour une grille active dans le chapitre 2 (équation 2.16). Cependant, cette expression peut se simplifier, ce qui permet d'aboutir à une complexité stochastique proche de celle considérée dans les travaux de Ruch *et al.* [82].

Dans [82], la complexité stochastique est ainsi constituée de seulement deux termes :

$$\Delta(\mathbf{s}, w, \hat{\boldsymbol{\theta}}) = \Delta_G(w) + \Delta_L(\mathbf{s}|\hat{\boldsymbol{\theta}}, w) \quad (4.18)$$

le terme de codage des paramètres n'étant pas pris en compte. L'omission de ce terme se justifie puisque le nombre de régions est fixe tout au long de la convergence et le terme  $\Delta_P(\hat{\boldsymbol{\theta}}|w)$  varie alors de façon négligeable par rapport aux deux autres termes.

De plus, dans [82] l'image est supposée constituée de seulement deux régions. Il suffit donc pour décrire le contour de donner la liste de tous les noeuds dans le sens trigonométrique. Ruch *et al.* ont ainsi proposé un terme de codage du contour égal à :

$$\Delta_G(w) = k \log N \quad (4.19)$$

où  $k$  est le nombre de noeuds sur le contour et où  $\log N$  est le nombre de nats nécessaires pour coder la position d'un noeud. L'utilisation d'un terme en  $\log N$  suppose que pour un noeud

donné, toutes les positions sont équiprobables, indépendamment de la position des nœuds auxquels il est relié. Quant au terme de codage des niveaux de gris, il reste bien évidemment toujours égal à l'opposé de la log-vraisemblance.

Dans la suite, nous utiliserons systématiquement la complexité stochastique fournie par l'équation 4.18. Le terme de codage de la grille en  $k \log N$  fournie par l'équation 4.19 sera utilisé dans le cas de contours actifs mono-région. En revanche, dans le cas multi-régions, il est nécessaire de pouvoir décrire plusieurs contours et le terme  $\Delta_G(w)$  plus général fourni dans l'équation 2.7 est donc plus approprié.

En résumé, nous utiliserons dans cette partie une complexité stochastique formée de seulement deux termes. Le premier,  $\Delta_G(w)$ , sera défini soit par l'équation 4.19 soit par l'équation 2.7, selon que le contour actif est mono-région ou multi-régions. Quant au second,  $\Delta_L(\mathbf{s}|\widehat{\boldsymbol{\theta}}, w)$ , il demeure égal à l'opposé de la log-vraisemblance, ce qui signifie en particulier que toutes les généralisations présentées précédemment peuvent aussi s'appliquer au cas des contours actifs.

### 4.3.2 Algorithme de segmentation par contours actifs multi-régions

La segmentation par contours actifs multi-régions est effectuée en utilisant la procédure de minimisation de la complexité stochastique décrite dans le cas d'un contour actif polygonal mono-région [82].

Cette procédure consiste à initialiser l'algorithme avec un contour comportant peu de nœuds (typiquement, le contour initial est un rectangle composé de 4 nœuds) puis à alterner des phases de déplacement des nœuds et des phases d'ajout de nœuds, jusqu'à ce que la distance entre deux nœuds consécutifs soit inférieure à une certaine distance  $d$  (généralement,  $d = 4$  pixels). Ce n'est qu'à partir de ce contour dont le nombre de nœuds a été surestimé, que la procédure de suppression des nœuds est mise en œuvre. L'objectif de cette approche, complexification puis simplification du contour, est de permettre au contour actif de ne pas rester piégé dans un minimum local lors de la convergence [82].

Pour généraliser cette approche au cas de contours actifs multi-régions, il suffit simplement de considérer une grille initiale dont on a imposé le nombre de régions et la topologie en fonction de l'objet que l'on cherche à segmenter. Ainsi, l'algorithme détaillé dans cette partie est adapté aussi bien au cas d'un contour actif mono-région que multi-régions.

Si la phase de déplacement des nœuds est rigoureusement identique à celle décrite précédemment et repose toujours sur la minimisation de la complexité stochastique, en revanche, une phase d'ajout de nœuds doit être mise en œuvre puisqu'elle n'était pas nécessaire dans l'algorithme de partition par grille active. Cette phase d'ajout de nœuds consiste en fait simplement à ajouter des nœuds sur chaque segment, sans se préoccuper de la variation de la complexité stochastique que cela induit : l'estimation du nombre de nœuds ne sera effectuée que dans un second temps.

Lors de l'alternance des phases d'ajout de nœuds, la complexité du contour augmente. Ce n'est qu'au fil des ajouts de nœuds que tous les détails de la cible vont pouvoir apparaître. L'utilisation de cette approche de type *multi-résolution*, permet de diminuer très fortement l'influence du contour initial sur le résultat de la segmentation [34, 16]. La seconde étape du processus d'optimisation permet de régulariser le contour en supprimant tous les nœuds inutiles [82]. Cela est réalisé en minimisant la complexité stochastique afin d'estimer le nombre de nœuds du contour.

A titre d'exemple, nous présentons sur la fig. 4.22 différentes étapes de la segmentation d'un avion par contour actif polygonal mono-région.

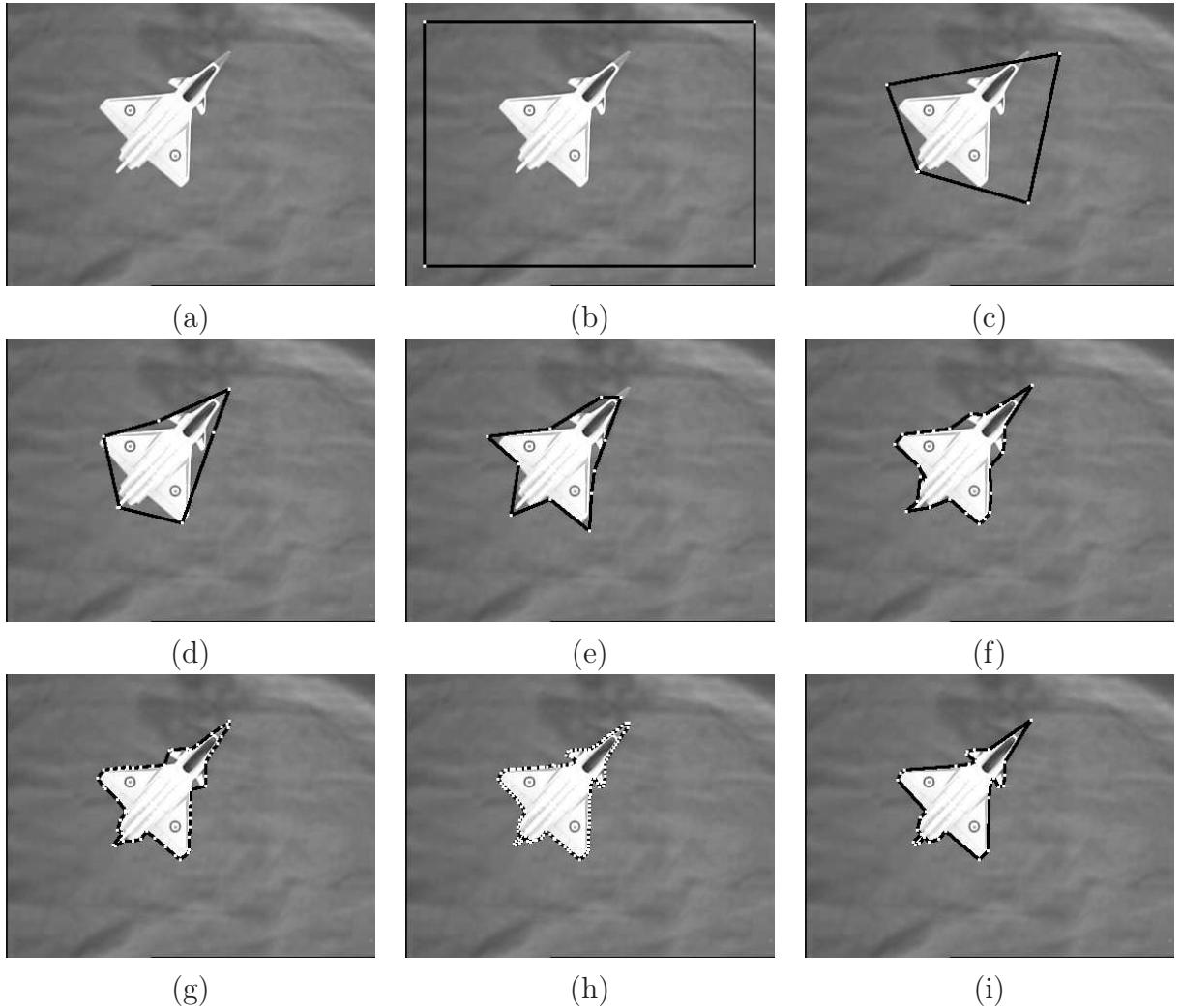


FIGURE 4.22 – Segmentation d'un modèle réduit d'avion par contour actif mono-région avec la technique développée par Ruch et al. [82] : (a) Image ( $376 \times 288$  pixels) - (b) Contour initial - (c) Après déplacement des nœuds - (d,e,f,g,h) Après ajout de nœuds puis déplacement des nœuds - (i) Après suppression des nœuds inutiles. Sur ces images, le contour est affiché en noir et les nœuds en blanc.

### 4.3.3 Résultats de segmentation

Nous présentons dans cette section quelques résultats de segmentation par contours actifs multi-régions sur différents types d'images scalaires ainsi que sur des images vectorielles couleurs et SAR polarimétriques.

#### Contours actifs multi-régions

Nous considérons tout d'abord des exemples de segmentation par contours actifs multi-régions d'images scalaires. Nous présentons la segmentation d'une brosse métallique et de son

ombre acquise à l'aide d'une caméra CCD sur la fig. 4.23, celle d'un doigt bruité par un bruit de Poisson sur la fig. 4.24 ainsi que celle d'une paire de ciseaux sur la fig. 4.25. Ces résultats illustrent clairement l'intérêt de cet algorithme de segmentation multi-régions.

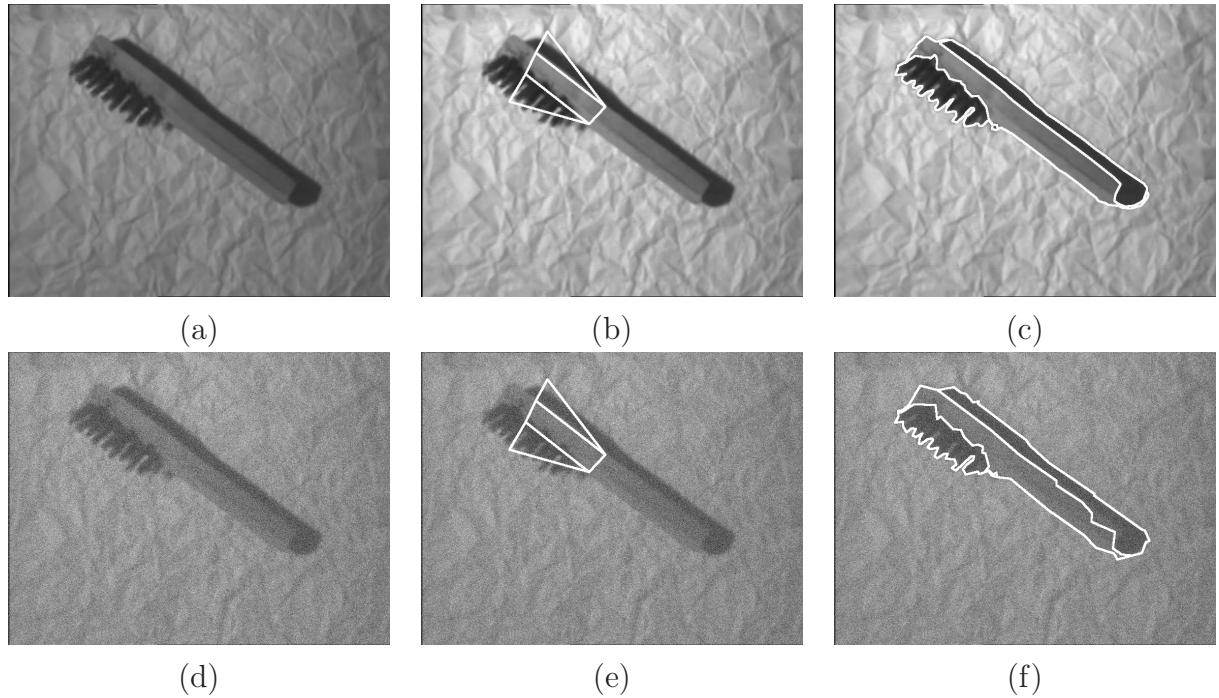


FIGURE 4.23 – Segmentation d'une brosse métallique et de son ombre par contour actif multi-régions : Ligne 1 (image réelle) : (a) Brosse métallique et son ombre acquise à l'aide d'une caméra CCD - (b) Contour initial - (c) Résultat de la segmentation. Ligne 2 (idem mais après bruitage) : (d) Image (a) bruitée par un bruit gaussien ( $\sigma = 30$ ) - (e) Contour initial - (f) Résultat de la segmentation. Ces 2 images ont une taille de  $376 \times 288$  pixels et ont été segmentée en moins de 500ms en utilisant un critère adapté à un bruit gaussien.

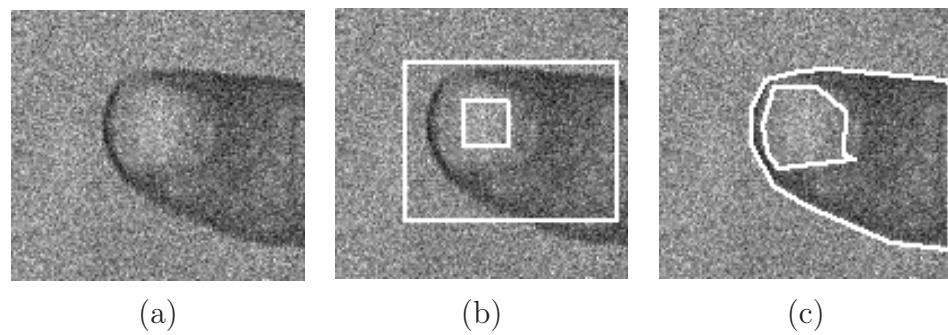


FIGURE 4.24 – Segmentation par contour actif multi-régions d'un doigt bruité par un bruit de Poisson : (a) Image acquise à l'aide d'une caméra CCD puis bruitée ( $120 \times 130$  pixels) - (b) Contour initial - (c) Résultat de la segmentation. Cette image a été segmentée en 70ms en utilisant un critère adapté à un bruit de Poisson.

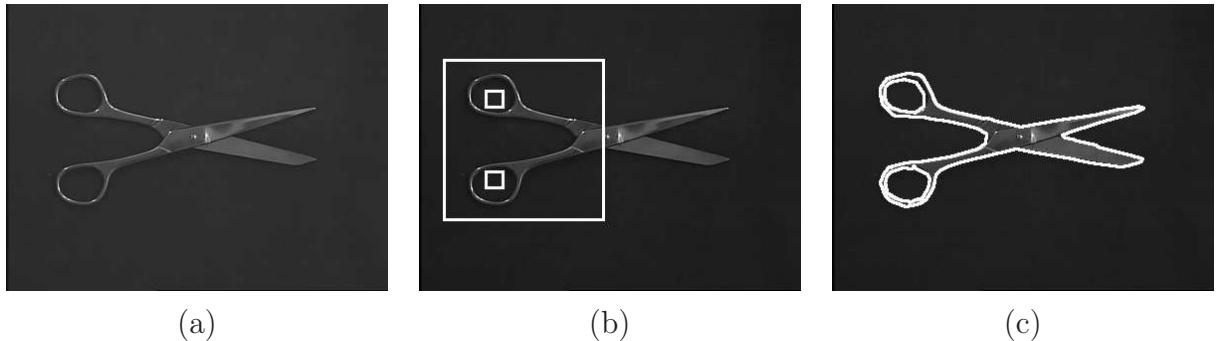


FIGURE 4.25 – Segmentation d'une paire de ciseaux par contour actif multi-régions : (a) Image ( $384 \times 288$  pixels) - (b) Contour initial - (c) Résultat de la segmentation. Cette image a été segmentée en 400ms en utilisant un critère adapté à un bruit gaussien.

### Segmentation d'images vectorielles

Nous allons maintenant utiliser cet algorithme sur des images vectorielles. Nous avons donc combiné l'algorithme de contour actif présenté ci-dessus avec l'expression de la complexité stochastique adaptée aux divers types d'images vectorielles fournies dans la partie 4.2 de ce chapitre.

Nous présentons sur la fig. 4.26, la segmentation d'une l'image SAR polarimétrique d'une petite île de la baie de San Francisco, à l'aide d'un contour actif mono-région et en utilisant le critère adapté aux images polarimétriques qui a été développé dans la partie 4.2.4. Ce premier exemple illustre la facilité d'utilisation des résultats établis dans le cadre de la partition en régions homogènes pour la segmentation par contour actif statistique polygonal.

Nous proposons également d'illustrer l'approche proposée dans le cas de la segmentation d'images couleurs optiques. Ces images sont classiquement composées de 3 canaux, un canal rouge (R), un canal vert (V) et un canal bleu (B) (on parle d'espace de couleur RVB). Il peut donc s'avérer intéressant de segmenter de telles images en prenant en compte simultanément les 3 canaux. Nous avons supposé dans cette section que les différents canaux étaient indépendants : bien que cette hypothèse ne soit pas forcément satisfaite, elle offre l'avantage de la simplicité. Les images couleurs utilisées dans la suite ayant été acquises à l'aide d'une caméra vidéo, les ddp des niveaux de gris dans chaque composante seront supposées gaussiennes.

Sur la fig. 4.27, nous présentons les résultats de la segmentation d'un hélicoptère à l'aide d'un contour actif mono-région en prenant en compte, soit les 3 canaux de l'espace RVB (fig. 4.27.c), soit seulement un des canaux R, V ou B (fig. 4.27.d,e,f) ou soit l'image scalaire en niveaux de gris (fig. 4.27.b).

Nous présentons ensuite les résultats sur l'image couleur de la fig. 4.28.a, tout d'abord en ne prenant en compte que l'image en niveau de gris (fig. 4.28.b), puis les 3 composantes couleur RVB (fig. 4.28.c). Nous remarquons que les résultats ne sont pas satisfaisants : la présence d'ombres perturbe la segmentation de l'objet. Analysons l'influence d'un changement d'espace de couleurs [83], afin de passer de l'espace RVB à l'espace YUV. Ce changement est obtenu en effectuant une transformation linéaire. Dans ce nouvel espace, la composante Y est la luminance et les composantes U et V sont des composantes colorimétriques dans lesquelles l'influence des ombres est beaucoup moins marquée. Quand on effectue une segmentation sur les composantes YUV, on obtient un résultat nettement plus satisfaisant, comme cela est

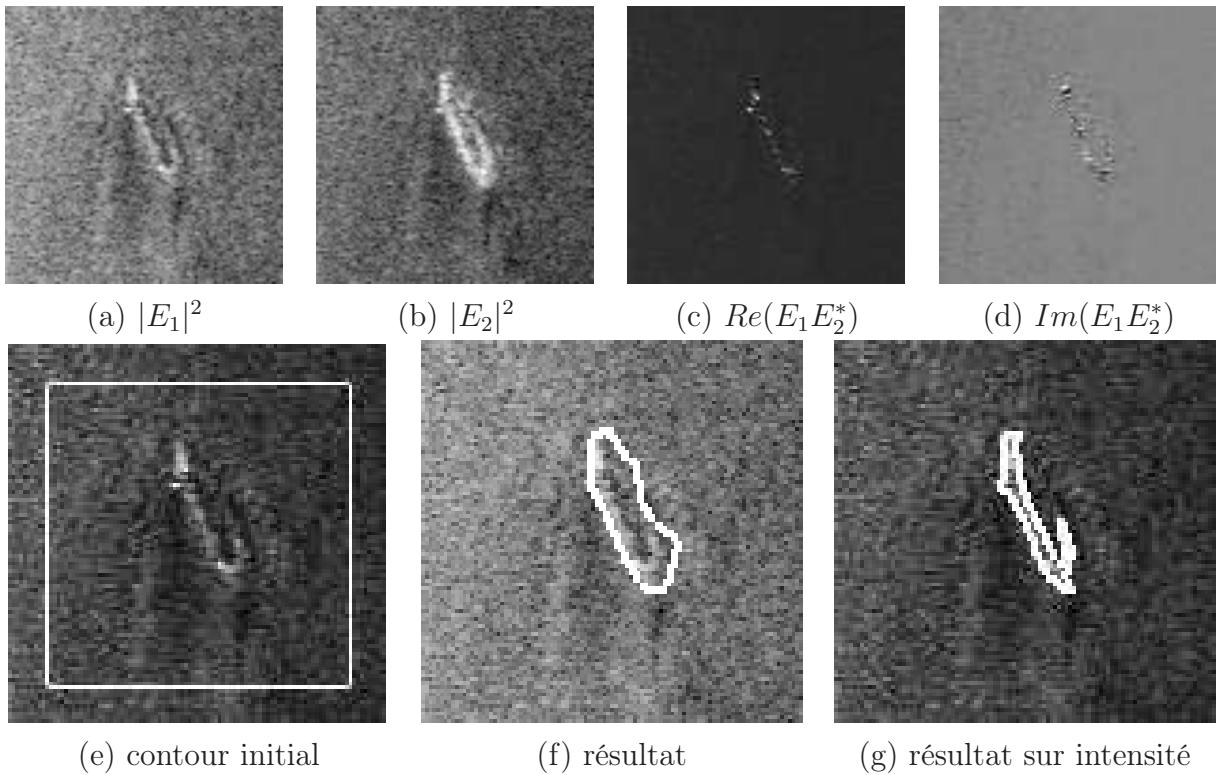


FIGURE 4.26 – Segmentation par contour actif d'une image SAR polarimétrique de la baie de San Francisco (image AIRSAR NASA/JPL). (a,b,c,d) Image polarimétrique - (e) Contour initial - (f) Segmentation de l'image polarimétrique - (g) Segmentation de l'image d'intensité  $|E_1|^2 + |E_2|^2$ .

illustré sur la fig. 4.28.d.

Analysons un troisième exemple (voir fig. 4.29) pour lequel les segmentations obtenues aussi bien en niveaux de gris (fig. 4.29.b) que dans les espaces RVB (fig. 4.29.d) ou YUV (fig. 4.29.e) ne sont pas satisfaisantes en raison de la présence d'ombres très importantes. Comme les composantes U et V de l'espace YUV sont assez peu sensibles aux phénomènes d'ombrage, la plupart de l'information liée à l'éclairement est concentrée dans la composante Y. Il est donc intéressant d'analyser les résultats de segmentation lorsque seules les composantes U et V de l'espace YUV sont prises en compte. La segmentation obtenue sur cette image à deux composantes est alors satisfaisante, comme on peut le voir sur la fig. 4.29.f.

Ces trois exemples illustrent l'intérêt du choix d'un espace de couleur adapté à l'application visée. Ainsi, comme nous l'avions souligné dans la partie 4.1.2 relative à la partition d'images vidéo, le choix d'autres espaces de représentation des images dans lesquels les objets d'intérêt correspondent à des régions plus homogènes peut s'avérer intéressant. A ce propos, nous montrerons d'ailleurs dans le chapitre suivant que l'utilisation de la composante de teinte H de l'espace colorimétrique YHS est prometteuse.

Le dernier exemple présenté dans cette partie concerne la poursuite de la forme d'un objet dans une séquence d'images. Nous pouvons voir sur la ligne 1 de la fig. 4.30 une séquence d'images couleurs au cours de laquelle un modèle réduit de camion vide sa benne et se déplace. Le résultat de la segmentation sur cette séquence d'images à l'aide d'un contour actif multi-régions (ici, la cible est composée de 2 régions : la région rouge et la région jaune) est affiché

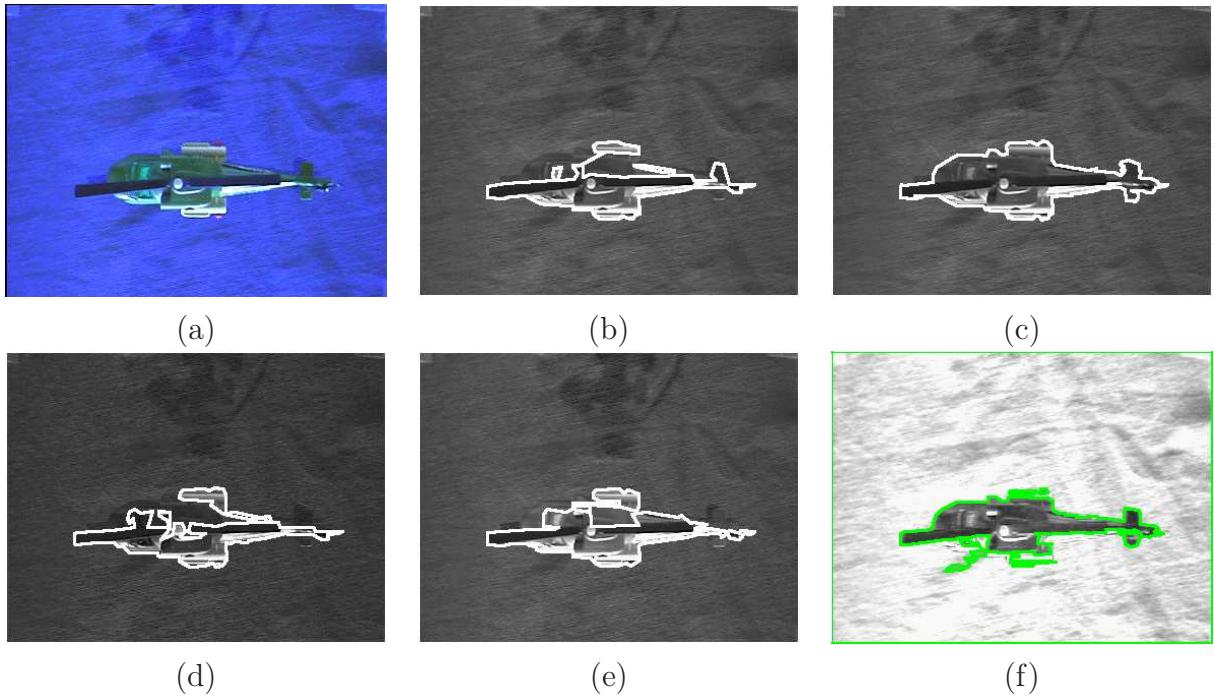


FIGURE 4.27 – Segmentation d’une image couleur ( $376 \times 288$  pixels) obtenue par caméra CCD. (a) image couleur - (b) segmentation de l’image en niveaux de gris - (c) segmentation de l’image couleur dans l’espace RVB - (d) segmentation sur la composante rouge (R) - (e) segmentation sur la composante verte (V) - (f) segmentation sur la composante bleue (B).

sur la troisième ligne de la fig. 4.30. A chaque étape, le contour initial utilisé, affiché sur la deuxième ligne, est composé de 6 noeuds et délimite deux régions cibles. Il est obtenu, sauf sur la première image, à partir du résultat de segmentation de l’étape précédente sur lequel on a appliqué une phase de suppression des noeuds qui est poursuivie jusqu’à ce qu’il ne reste plus que 6 noeuds sur le contour.

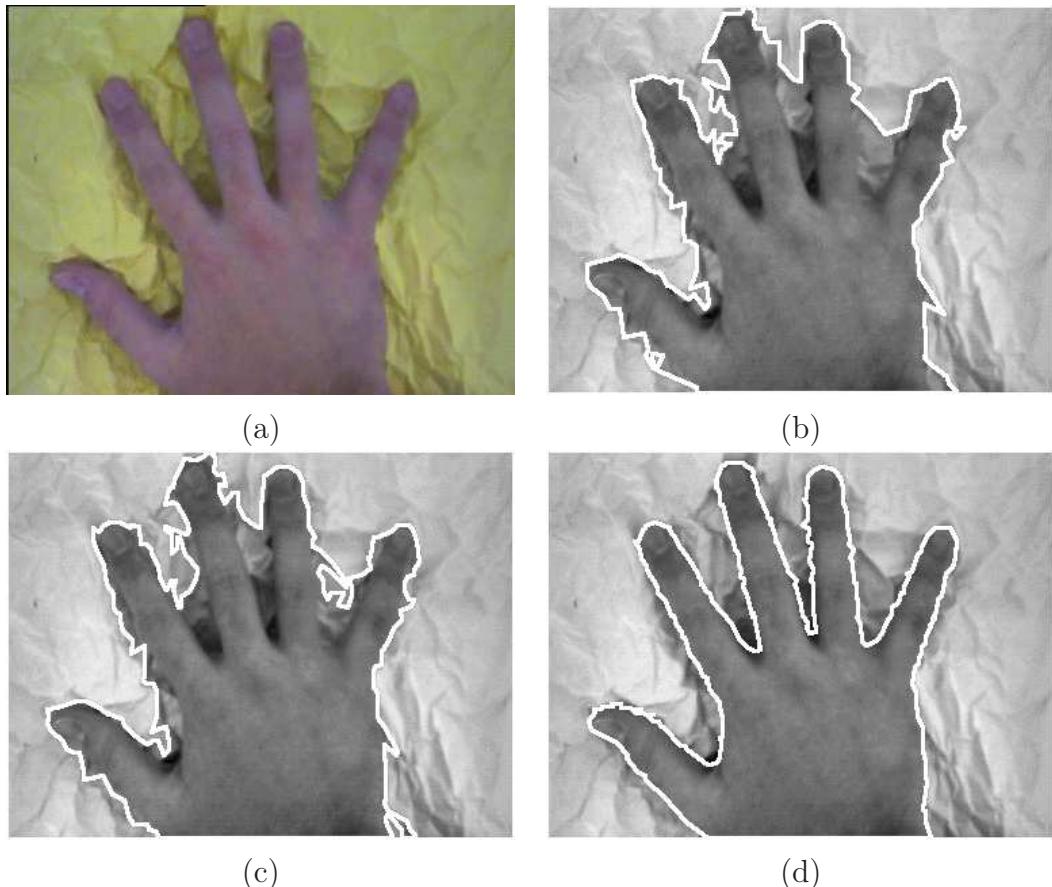


FIGURE 4.28 – Segmentation d'une image couleur ( $376 \times 288$  pixels) obtenue par caméra CCD. (a) image couleur - (b) segmentation de l'image en niveaux de gris - (c) segmentation de l'image couleur dans l'espace RVB - (d) segmentation de l'image couleur dans l'espace YUV. Les résultats de la segmentation sont affichés sur l'image en niveaux de gris.

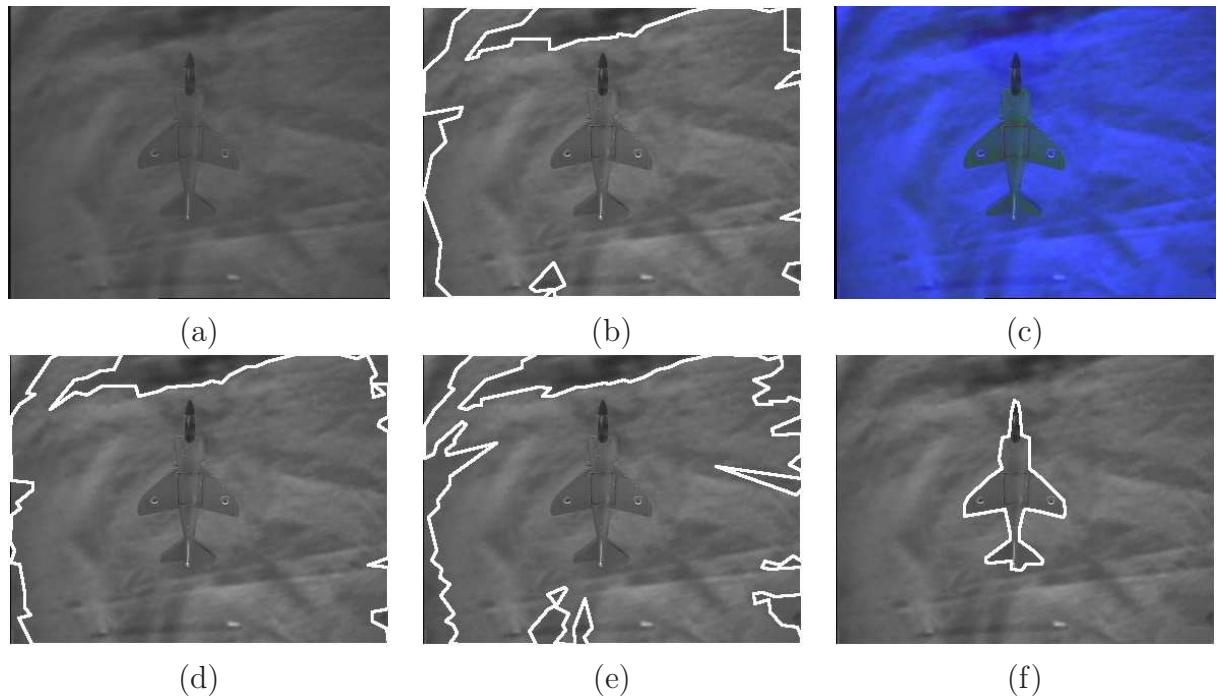


FIGURE 4.29 – Segmentation d’une image couleur ( $376 \times 288$  pixels) obtenue par caméra CCD. (a) image en niveaux de gris - (b) segmentation de l’image en niveaux de gris - (c) image couleur - (d) segmentation de l’image couleur dans l’espace RVB - (e) segmentation de l’image couleur dans l’espace YUV - (f) segmentation de l’image couleur en prenant seulement en compte les composantes  $U$  et  $V$  de l’espace YUV. Les résultats de la segmentation sont affichés sur l’image en niveaux de gris.

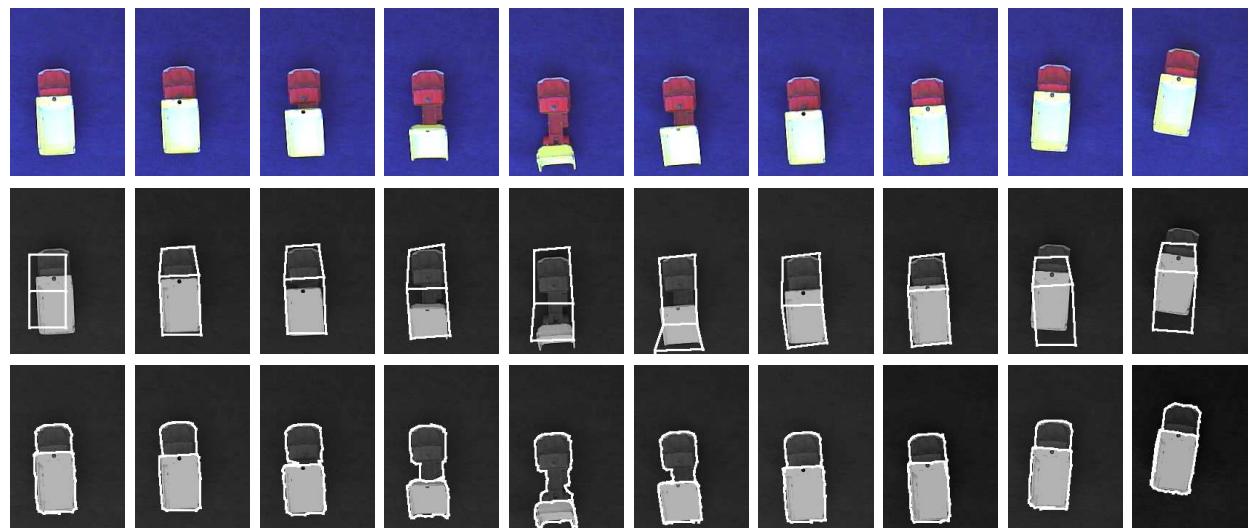


FIGURE 4.30 – Segmentation et poursuite d’un modèle réduit de camion vidant sa benne. Ligne 1 : séquence d’images. Ligne 2 : contours initiaux déterminés à partir de la segmentation obtenue à l’étape précédente. Ligne 3 : résultats de la segmentation (affichés sur l’image en niveaux de gris). Chaque image contient  $384 \times 288$  pixels et est segmentée dans l’espace RVB en environ 200ms.

#### 4.3.4 Introduction de connaissances *a priori* sur les paramètres de la cible

Pour certaines applications comme la poursuite d'objets, il se peut que l'on dispose d'informations *a priori* sur la cible, par exemple sa topologie comme nous avons pu le voir précédemment, mais aussi sa couleur. Nous allons voir ici que lorsque l'on connaît les paramètres statistiques de la cible, il est possible de les injecter dans le modèle, afin d'augmenter la robustesse de l'algorithme de segmentation.

Plaçons-nous dans le cas d'un contour mono-région, c'est-à-dire d'une image composée de deux régions, la cible  $\Omega_a$  et le fond  $\Omega_b$ . Nous considérerons de plus le cas d'images vidéo que nous modéliserons à l'aide de ddp gaussiennes. Les niveaux de gris seront ainsi distribués selon une ddp gaussienne de moyenne  $m_a$  et d'écart-type  $\sigma_a$  pour la cible, de moyenne  $m_b$  et d'écart-type  $\sigma_b$  pour le fond. Nous supposerons de plus que les paramètres de la cible,  $m_a$  et  $\sigma_a$  sont connus : seuls les paramètres du fond  $m_b$  et  $\sigma_b$  seront supposés inconnus et devront donc être estimés (toujours au sens du MV).

Nous devons donc calculer le terme de codage des niveaux de gris  $\Delta_L(\mathbf{s}|w, (m_a, \sigma_a, \hat{m}_b, \hat{\sigma}_b))$ , où l'on a noté  $\hat{m}_b$  et  $\hat{\sigma}_b$  les estimées au sens du MV des paramètres du fond  $m_b$  et  $\sigma_b$  (les valeurs de  $\hat{m}_b$  et  $\hat{\sigma}_b$  dans le cas d'une ddp gaussienne sont indiquées dans le tableau 4.2). Bien évidemment, les paramètres de la cible  $m_a$  et  $\sigma_a$  n'ont pas à être estimés puisqu'ils sont supposés connus.

Nous pouvons montrer facilement (voir annexe A.7), que le terme de codage des niveaux de gris s'écrit :

$$\begin{aligned} \Delta_L(\mathbf{s}|w, (m_a, \sigma_a, \hat{m}_b, \hat{\sigma}_b)) &= -\mathcal{L}_e[\Omega_a|m_a, \sigma_a] - \mathcal{L}_e[\Omega_b|\hat{m}_b, \hat{\sigma}_b] \\ &= \frac{1}{\sigma_a^2} \left( \sum_{(x,y) \in \Omega_a} s(x,y)^2 - 2m_a \sum_{(x,y) \in \Omega_a} s(x,y) + N_a m_a^2 \right) \\ &\quad + N_a \log \sigma_a + \frac{1}{2} N_b \log (\hat{\sigma}_b)^2 + \frac{N_b}{2} + \frac{1}{2} N \log 2\pi \end{aligned} \quad (4.20)$$

Ayant déterminé la log-vraisemblance généralisée, nous pouvons maintenant utiliser l'algorithme de segmentation par contour actif afin de réaliser la poursuite d'objet sur des images couleurs. Les différents canaux seront toujours supposés indépendants, ce qui signifie que  $\Delta_L(w)$  est simplement la somme des termes de codage des niveaux de gris calculés sur chaque canal à l'aide de (4.20).

Les résultats de la fig. 4.31 illustrent l'apport que permet la connaissance des paramètres de la cible. Sur la fig. 4.31.b, l'image très simple de la fig. 4.31.a contenant un stylo vert a été segmentée. La valeur de la moyenne et de la variance de ce stylo dans chacune des composantes couleur peut ainsi être déterminée facilement. Une nouvelle image, contenant ce même stylo vert parmi trois autres stylos est ensuite considérée sur la fig. 4.31.c. Nous présentons les résultats obtenus avec deux stratégies différentes : soit en ne considérant aucune information *a priori* sur les paramètres de la cible (fig. 4.31.d), soit en supposant connus les paramètres de moyennes et de variances du stylo vert (fig. 4.31.e) c'est-à-dire en considérant alors les valeurs déterminées précédemment. Nous voyons clairement sur la fig. 4.31.e que la donnée d'information *a priori* sur la cible permet de segmenter le bon stylo.

Nous avons ensuite utilisé cette approche sur une séquence de poursuite de cible (fig. 4.32). Soulignons qu'afin d'obtenir un algorithme robuste, la position du contour initial est estimée automatiquement à chaque étape, à l'aide d'une procédure simple qui consiste à choisir parmi différentes positions possibles, celle qui minimise la complexité stochastique de l'image (cal-

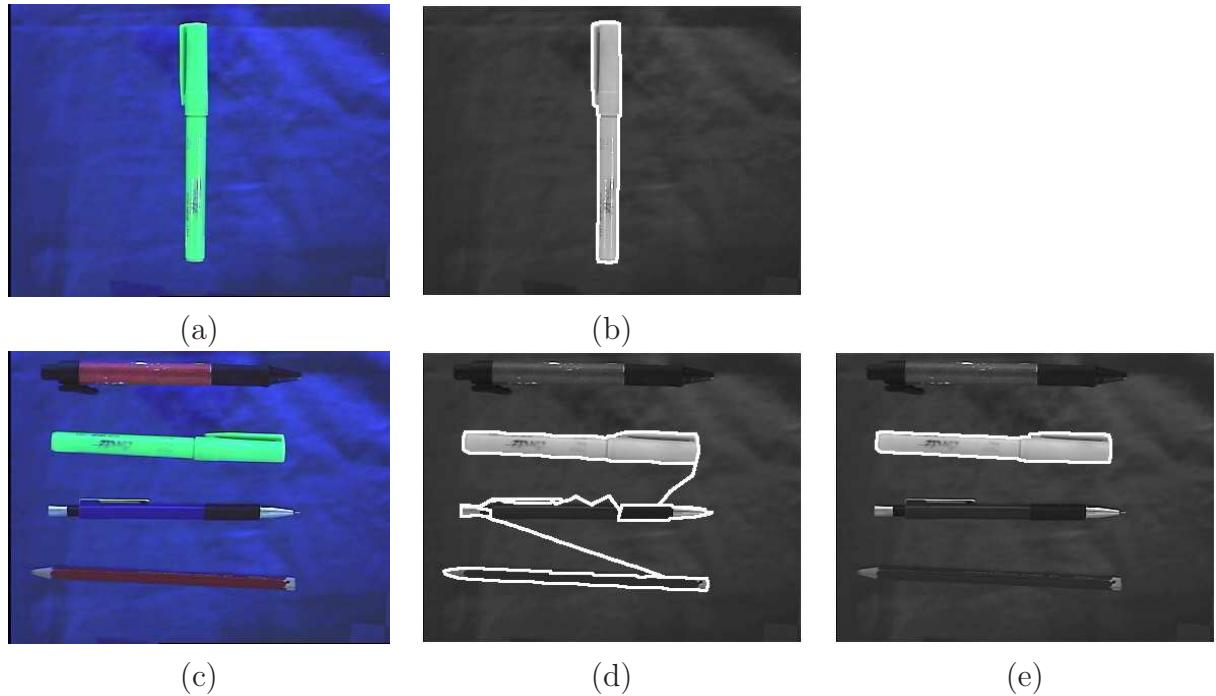


FIGURE 4.31 – Segmentation par contour actif d’un stylo dans une image couleur ( $376 \times 288$  pixels) lorsque l’on connaît les paramètres de sa ddp. (a) image couleur contenant seulement l’objet d’intérêt - (b) segmentation de l’image permettant d’extraire moyenne et variance de la cible dans chaque composante couleur - (c) image couleur contenant 4 stylos (dont l’objet d’intérêt) - (d) segmentation de l’image (c) sans prise en compte des paramètres de l’objet d’intérêt - (e) segmentation de l’image (c) en injectant moyennes et variances de l’objet segmenté en (b). Les différentes segmentations ont été effectuées dans l’espace de couleur YUV. Les résultats de segmentation sont affichés sur l’image en niveaux de gris.

culée à l’aide de l’équation 4.20).

### 4.3.5 Segmentation multi-lois

Comme nous l’avons vu précédemment, le principal avantage de l’utilisation d’un contour actif par rapport à une grille active est de pouvoir injecter dès le début des informations *a priori* dans le modèle, principalement le nombre et la topologie des objets recherchés. Cependant, d’autres informations peuvent être utilisées comme par exemple les paramètres statistiques de la cible (voir partie 4.3.4).

Dans cette section, nous allons montrer qu’il est possible d’injecter une information *a priori* sur les familles des ddp du bruit présent dans chaque région. En effet, jusqu’à présent, nous avons supposé que la famille des ddp du bruit présent dans les images était connue et identique dans toutes les régions. Pour certaines applications particulières, cette dernière hypothèse peut être une limitation. Par exemple en imagerie active (radar ou sonar), les régions éclairées sont dégradées par un phénomène de speckle, que l’on modélisera simplement par une loi gamma comme nous l’avons vu précédemment, alors que les zones d’ombres ne sont perturbées que par un bruit dû à l’électronique, qui est mieux modélisé par une ddp gaussienne. C’est également le cas en présence de textures qui peuvent modifier la famille des ddp dans les différentes régions.

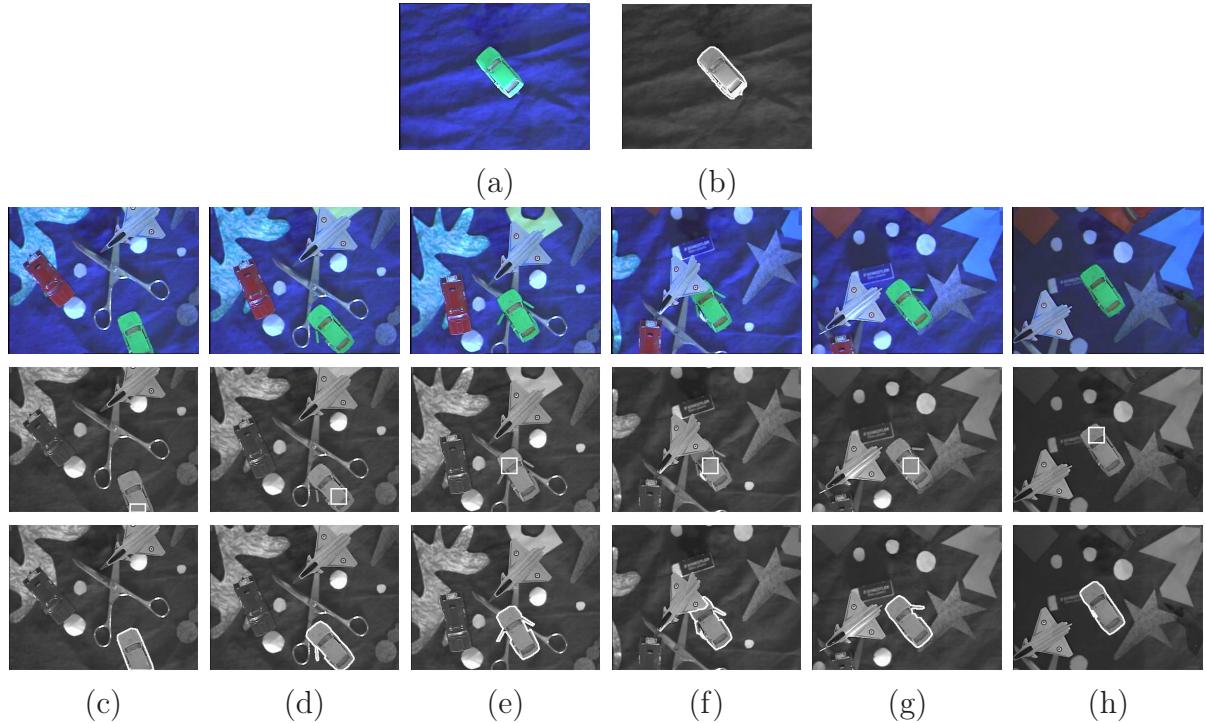


FIGURE 4.32 – Poursuite d'une cible dans une série d'images couleurs ( $376 \times 288$  pixels) lorsque les paramètres de la cible sont connus. (a) Image contenant uniquement la cible. (b) Segmentation de l'image (a) afin d'extraire moyenne et variance de la cible sur chaque composante couleur. (c-h) Poursuite de la cible en utilisant les informations moyennes et variances de la cible obtenues en (b) : ligne 1 : séquence d'images, ligne 2 : contours initiaux, ligne 3 : résultats de la segmentation. Les différentes segmentations ont été effectuées dans l'espace de couleur YUV.

Nous allons ainsi nous intéresser à la segmentation d'objets par contours actifs multi-régions lorsque la famille de ddp dans chaque région peut être différente. Nous montrerons comment généraliser la technique que nous avons développée et nous verrons qu'il est nécessaire de prendre en compte de nouvelles statistiques qui n'intervenaient pas jusqu'à présent dans le calcul des log-vraisemblances (tableau 4.2).

Notons  $P^{l_r}$  la famille de ddp de la région  $\Omega_r$ , de sorte que les niveaux de gris des pixels de cette région sont distribués selon la ddp  $P_{\theta_r}^{l_r}$  de vecteur de paramètres  $\theta_r$ . Afin de pouvoir calculer la complexité stochastique associée à ce nouveau modèle, il suffit de déterminer le terme de codage des niveaux de gris  $\Delta_r$  dans chacune des régions  $\Omega_r$ , en fonction de la famille de sa ddp  $P^{l_r}$ . Cependant, les informations fournies dans le tableau 4.2 ne sont pas suffisantes car elles ne sont définies qu'à des termes additifs près, indépendants de la partition mais dépendants de la famille de la ddp.

A titre d'exemple, dans le cas d'une ddp de Poisson de paramètre  $\theta_r$ ,  $\Delta_r$  s'écrit :

$$\Delta_r(\Omega_r | \widehat{\theta}_r) = -N_r \widehat{\theta}_r \log \widehat{\theta}_r + \sum_{(x,y) \in \Omega_r} (s(x,y) + \log [s(x,y)!]) \quad (4.21)$$

où  $\widehat{\theta}_r$  est la moyenne empirique calculée sur la région  $\Omega_r$  et  $n! = n(n-1) \times (n-2) \dots 2 \times 1$ . Lorsque les niveaux de gris de toutes les régions sont distribués selon la même famille de ddp,

à savoir poissonnienne, nous obtenons après sommation sur les  $R$  régions :

$$\begin{aligned}\Delta_L(\mathbf{s}|w, \hat{\boldsymbol{\theta}}) &= -\sum_{r=1}^R N_r \hat{\theta}_r \log \hat{\theta}_r + \sum_{r=1}^R \left( \sum_{(x,y) \in \Omega_r} (s(x,y) + \log [s(x,y)!]) \right) \\ &= -\sum_{r=1}^R N_r \hat{\theta}_r \log \hat{\theta}_r + \sum_{image} (s(x,y) + \log [s(x,y)!])\end{aligned}\quad (4.22)$$

Nous observons alors que le deuxième terme est simplement une sommation sur toute l'image de  $s(x,y) + \log [s(x,y)!]$ , c'est-à-dire un terme additif indépendant de la partition et qui peut donc être omis dans le tableau 4.2. Ainsi, lorsque toutes les ddp appartiennent à la même famille, il est inutile de prendre en compte le deuxième terme intervenant dans l'expression de  $\Delta_r$  fournie par (4.21).

Lorsque les familles de ddp peuvent différer d'une région à l'autre, cette simplification n'est plus possible. Il faut alors considérer tous les termes de (4.21), ce qui signifie en particulier qu'il est nécessaire de calculer - en plus des statistiques exhaustives qui étaient déjà utilisées - d'autres statistiques  $T_k(\chi_r)$  dépendant de l'échantillon  $\chi_r$  (avec  $\chi_r = \{s(x,y)|(x,y) \in \Omega_r\}$ ). Par exemple, pour le cas d'une ddp de Poisson, nous voyons d'après (4.21) qu'il est nécessaire de calculer, outre la statistique exhaustive  $T_1(\chi_r) = \sum_{(x,y) \in \Omega_r} s(x,y)$  intervenant dans le calcul de la moyenne  $\hat{\theta}_r$ , la statistique  $T_2(\chi_r) = \sum_{(x,y) \in \Omega_r} \log [s(x,y)!]$ .

Nous présentons dans le tableau 4.3 les expressions de  $\Delta_r(\Omega_r|\hat{\theta}_r)$  obtenues pour différentes familles de ddp et qui doivent être utilisées lorsque la famille de ddp peut varier d'une région à l'autre.

ddp	$\Delta_r(\Omega_r \hat{\theta}_r)$	$\hat{\theta}_r$	$T(\chi_r)$
Gaussienne	$N_r \left( \log \sqrt{2\pi} + \frac{1}{2} \log \hat{\theta}_r^2 + \frac{1}{2} \right)$	$\hat{\theta}_r^1 = \frac{1}{N_r} T_1(\chi_r)$ $\hat{\theta}_r^2 = \frac{1}{N_r} T_2(\chi_r) - (\hat{\theta}_r^1)^2$	$T_1(\chi_r) = \sum_{\Omega_r} s(x,y)$ $T_2(\chi_r) = \sum_{\Omega_r} s^2(x,y)$
Gamma d'ordre $L$	$N_r \left( \log \Gamma(L) + L(\log \hat{\theta}_r + 1 - \log L) \right) - (L-1)T_2(\chi_r)$	$\hat{\theta}_r = \frac{1}{N_r} T_1(\chi_r)$	$T_1(\chi_r) = \sum_{\Omega_r} s(x,y)$ $T_2(\chi_r) = \sum_{\Omega_r} \log s(x,y)$
Poisson	$T_1(\chi_r) + T_2(\chi_r) - N_r \hat{\theta}_r \log \hat{\theta}_r$	$\hat{\theta}_r = \frac{1}{N_r} T_1(\chi_r)$	$T_1(\chi_r) = \sum_{\Omega_r} s(x,y)$ $T_2(\chi_r) = \sum_{\Omega_r} \log (s(x,y)!)$
Géométrique	$N_r \left( (\hat{\theta}_r + 1) \log(\hat{\theta}_r + 1) - \hat{\theta}_r \log \hat{\theta}_r \right)$	$\hat{\theta}_r = \frac{1}{N_r} T_1(\chi_r)$	$T_1(\chi_r) = \sum_{\Omega_r} s(x,y)$

TABLE 4.3 – Expression du terme de codage des niveaux de gris  $\Delta_r(\Omega_r|\hat{\theta}_r)$  dans la région  $\Omega_r$  pour différentes ddp. Nous précisons aussi la valeur de l'estimée  $\hat{\theta}_r$  au sens du MV du vecteur de paramètres ainsi que l'expression des statistiques  $T_k(\chi_r)$ . Nous avons de plus utilisé la notation  $x! = x \times (x-1) \times \dots \times 2 \times 1$ .

Afin d'illustrer cette approche, nous avons simulé sur la fig. 4.33 une image de type SAR contenant une cible et son ombre. Dans cet exemple, la cible et le fond sont distribués selon des ddp gamma d'ordre  $L = 1$  alors que l'ombre de la cible est distribuée selon une ddp gaussienne. Nous présentons les segmentations obtenues à l'aide d'un contour actif multi-régions tout d'abord lorsque l'on impose la bonne famille de ddp dans chaque région (fig. 4.33.d), puis lorsque la famille de ddp est supposée la même sur toute l'image (fig. 4.33.e et f).

Sur la fig. 4.34, nous avons effectué le même type de simulation avec une image synthétique pouvant représenter une image obtenue à faible flux de photons. L'objet est constitué de plusieurs régions perturbées par des bruits de ddp de Poisson ou géométriques. Une ddp géométrique correspond à des textures ayant des réflectivités fluctuant selon une loi exponentielle et qui sont illuminées à un faible flux de photons, modélisé par un bruit de Poisson.

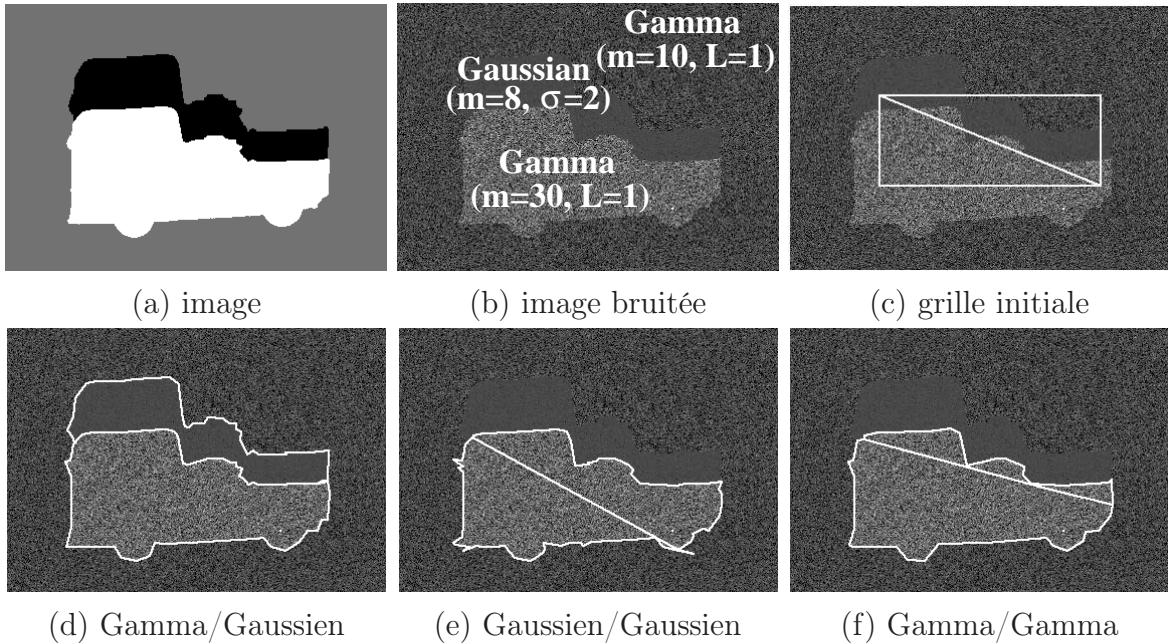


FIGURE 4.33 – Segmentation d’une image synthétique ( $380 \times 266$  pixels) perturbée par un bruit gamma d’ordre  $L = 1$  pour la voiture et le fond et par un bruit gaussien pour l’ombre de la voiture. (d) Résultat de segmentation avec les vraies familles de ddp (c’est-à-dire gaussienne pour l’ombre et gamma d’ordre  $L = 1$  pour les deux autres régions). (e) Résultat de la segmentation en choisissant uniquement des ddp gaussiennes. (f) idem mais avec uniquement des ddp gamma d’ordre  $L = 1$ . Les images ont été réhaussées pour une meilleure visualisation.

Ces deux exemples montrent clairement que la prise en compte de la variabilité de la famille des ddp permet d’améliorer nettement les résultats de segmentation.

## 4.4 Conclusion

Au cours de ce chapitre, nous avons montré que la méthode de partition proposée au chapitre 2 pouvait très simplement être généralisée dans diverses directions.

En premier lieu, le modèle d’image a été généralisé à d’autres lois de probabilité que des lois gamma. Nous avons de plus montré que dans le cas de lois gaussiennes, l’espérance mathématique pouvait être supposée suivre une fonction linéaire ou même quadratique et non plus seulement constante sur la région.

Dans la deuxième partie, nous nous sommes penchés sur la généralisation à des images vectorielles, notamment SAR multi-dates et SAR polarimétriques. Nous avons ainsi pu montrer l’intérêt de l’utilisation d’une approche vectorielle par rapport à des approches travaillant uniquement sur une seule composante ou même sur la somme de toutes les composantes de l’image.

Enfin, dans une dernière partie, nous nous sommes intéressés à la segmentation d’images par contour actif multi-régions, c’est-à-dire à l’aide d’une grille active dont la topologie et le nombre de régions sont connus et peuvent donc être imposés dès l’initialisation. Soulignons aussi que cette partie a été l’occasion d’utiliser simultanément les différentes généralisations proposées dans ce chapitre. Nous avons ainsi segmenté des images distribuées selon des lois de probabilité non gamma, des images SAR polarimétriques mais également des images vidéo

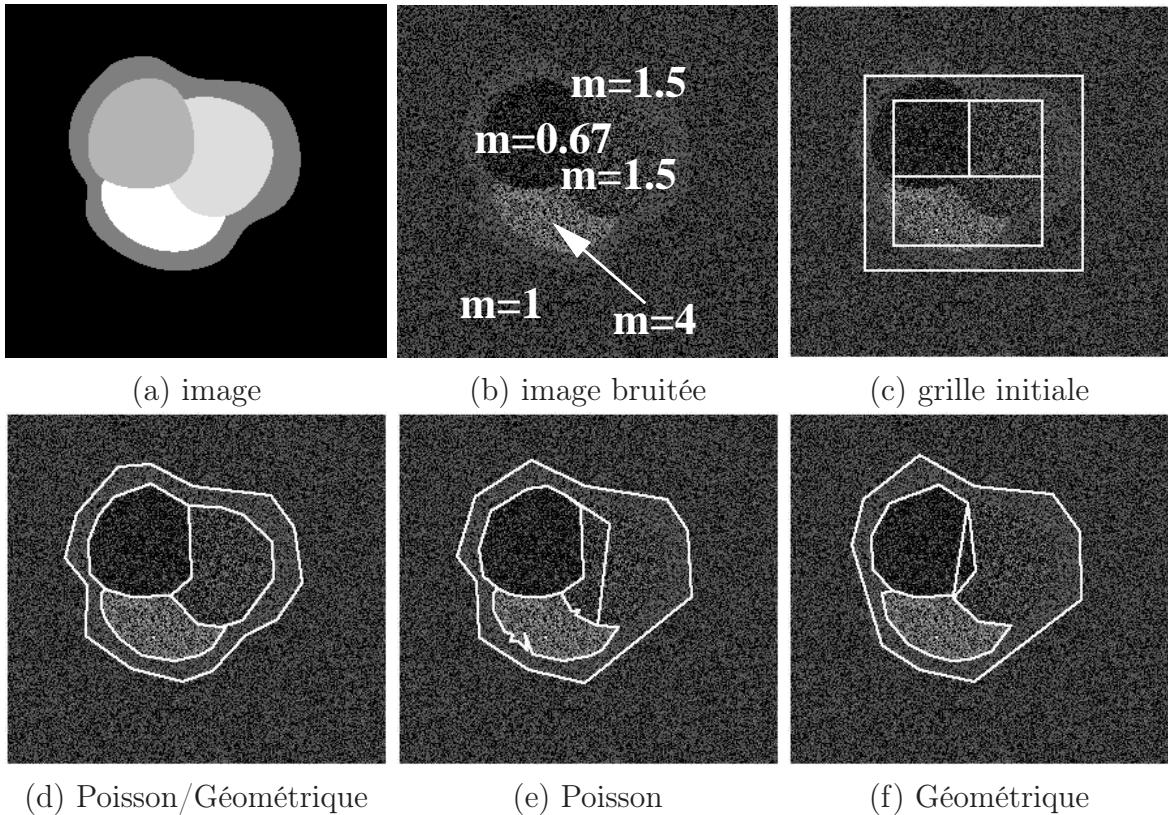


FIGURE 4.34 – Segmentation d’une image synthétique ( $293 \times 271$  pixels) perturbée par un bruit géométrique pour les trois régions centrales et un bruit de Poisson pour les deux autres régions. Les valeurs moyennes dans chaque région sont affichées en (b). (d) Résultat de segmentation avec les vraies familles de ddp (c’est-à-dire géométrique pour les trois régions centrales et Poisson pour les deux autres). (e) Résultat de segmentation en choisissant uniquement des ddp de Poisson. (f) idem mais avec uniquement des ddp géométriques. Les images ont été rehaussées pour une meilleure visualisation.

couleurs.

Nous conclurons ce chapitre en évoquant le cas de l’imagerie vidéo. En effet, il s’agit d’un bon exemple d’images très peu bruitées et dans lesquelles les objets d’intérêt ne constituent pas des régions homogènes. Dans ce cas, même si la partition en régions homogènes de telles images est satisfaisante, elle ne constitue pas une segmentation en régions d’intérêt.

Cependant, nous avons montré au cours de ce chapitre que l’injection d’informations de plus haut niveau permettait d’obtenir des segmentations intéressantes. Nous avons ainsi utilisé les informations *a priori* suivantes, séparément ou combinées entre elles :

- le nombre de régions contenues dans l’image,
- la topologie des régions,
- l’espace de couleur permettant de rendre les différents objets d’intérêt homogènes,
- les paramètres de la ddp de l’objet d’intérêt,
- la famille de ddp de chacune des régions de l’image.

La prise en compte de ces différentes informations, lors d’un pré-traitement, d’un post-traitement ou directement en les injectant dans le modèle, nous a alors permis de segmenter les objets d’intérêt dans différentes images vidéo.



# Chapitre 5

## Partition d'images à $Q$ niveaux

### Sommaire

---

5.1	Détermination de la complexité stochastique . . . . .	108
5.2	Algorithme rapide . . . . .	110
5.3	Résultats expérimentaux . . . . .	110
5.3.1	Images pré-classifiées réelles . . . . .	111
5.3.2	Quantification des niveaux de gris . . . . .	115
5.3.3	Images couleurs et polarimétriques . . . . .	117
5.3.4	Contrainte de la topologie par contours actifs . . . . .	118
5.4	Conclusion . . . . .	119

---

Jusqu'à présent, nous avons supposé que les niveaux de gris pouvaient être distribués selon quelques ddp appartenant à la famille exponentielle (Gaussienne, Gamma, Poisson, Bernoulli, etc). S'il est possible d'étendre facilement ces approches à d'autres ddp de la famille exponentielle, il reste cependant des cas où la ddp n'appartient pas à la famille exponentielle, comme par exemple pour une loi de Cauchy. Plus important encore : dans certains cas, la forme paramétrique de la ddp peut ne pas être connue *a priori* et l'approche utilisée jusqu'à présent ne peut alors plus être appliquée. C'est notamment le cas des images pré-classifiées en  $Q$  classes, où la valeur de chaque pixel n'est pas nécessairement une grandeur numérique. Ainsi, dans le cas où le modèle de la ddp des niveaux de gris n'est pas dans la famille exponentielle ou n'est pas disponible, l'approche utilisée jusqu'ici présente de fortes limitations.

Dans cette partie, nous proposons de généraliser cette approche au cas où les pixels prennent leurs valeurs dans un ensemble fini de taille  $Q$ . Nous montrerons qu'il est possible d'obtenir la partition de telles images sans pour autant imposer de modèle paramétrique pour la distribution des niveaux de gris, à condition de quantifier de manière adaptée son histogramme.

Nous verrons ensuite que même dans le cas où les niveaux de gris des pixels ne sont plus définis sur un ensemble discret, il est également possible d'utiliser cette approche pour partitionner une image sans connaissance *a priori* de la ddp des niveaux de gris. Nous illustrerons alors cette technique sur différentes applications, notamment la régularisation d'images pré-classifiées et la partition d'images couleurs, de polarisation, sonar et SAR.

## 5.1 Détermination de la complexité stochastique

Considérons le cas des images à  $Q$  niveaux, c'est-à-dire que la valeur de chaque pixel appartient à un nombre fini  $Q$  d'éléments, noté  $\Lambda_Q$  dans la suite.

La complexité stochastique  $\Delta$  de l'image s'écrit ici encore comme la somme de trois termes : un terme de codage des niveaux de gris  $\Delta_L$  connaissant la partition  $w$  et les ddp dans chacune des régions, un terme de codage  $\Delta_P$  des ddp dans chacune des régions définies par  $w$  et enfin un terme de codage de la partition  $w$ .

Si l'on suppose la partition  $w$  connue ainsi que les ddp  $P_r$  dans chacune des régions  $\Omega_r$  ( $r \in \{1, 2, \dots, R\}$ ) définies par  $w$ , le nombre de *nats* nécessaires pour coder les niveaux de gris des pixels s'écrit :

$$\Delta_L = \sum_{r=1}^R \Delta_r \quad (5.1)$$

où  $\Delta_r$  est égal à l'opposé de la log-vraisemblance dans la région  $\Omega_r$  :

$$\begin{aligned} \Delta_r &= -\mathcal{L}_e [\Omega_r | P_r] \\ &= -\sum_{(x,y) \in \Omega_r} \log P_r [s(x, y)] \end{aligned} \quad (5.2)$$

Notons  $\Lambda_Q = \{\lambda_q\}_{q=\{1,2,\dots,Q\}}$  l'ensemble des  $Q$  éléments de  $\Lambda_Q$  qui ne sont pas nécessairement ordonnés (c'est-à-dire pour lesquels  $\lambda_q \leq \lambda_{q'}$  n'a pas nécessairement de sens). Soit  $p_r(q)$  la probabilité qu'un pixel  $(x, y)$  de la région  $\Omega_r$  prenne la valeur  $\lambda_q$ . Nous avons donc  $\sum_{q=1}^Q p_r(q) = 1$  et  $\forall q \in \{1, 2, \dots, Q\}$

$$P_r [s(x, y) = \lambda_q] = p_r (q) . \quad (5.3)$$

Comme  $P_r$  est supposée inconnue, nous pouvons considérer son estimée au sens du MV. En effet, la ddp de l'équation (5.3) peut être considérée comme une ddp paramétrique de paramètre

$[p_r(1), p_r(2), \dots, p_r(Q)]$  de dimension  $Q$  (avec bien sûr  $p_r(1) + p_r(2) + \dots + p_r(Q) = 1$ ). Ce modèle général appartient à la famille exponentielle et il est bien connu que dans ce cas l'estimateur du MV est à variance minimum s'il est non biaisé [24, 30]. Les pixels étant supposés indépendants, la log-vraisemblance  $\mathcal{L}_e[\Omega_r | P_r]$  dans la région  $\Omega_r$  s'écrit :

$$\mathcal{L}_e[\Omega_r | P_r] = \sum_{(x,y) \in \Omega_r} \sum_{q=1}^Q \delta[s(x,y), \lambda_q] \log p_r(q) \quad (5.4)$$

où  $\delta(a, b)$  est le symbole de Kronecker (qui vaut 1 si  $a = b$  et 0 sinon). Notons que pour obtenir ce résultat nous devons écrire la ddp  $P_r$  de l'équation (5.3) sous la forme :

$$P_r[s(x,y)] = \prod_{q=1}^Q [p_r(q)]^{\delta(s(x,y), \lambda_q)} . \quad (5.5)$$

L'estimée  $\hat{p}_r(q)$  au sens du MV de  $p_r(q)$  s'obtient simplement (voir annexe A.8)

$$\hat{p}_r(q) = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} \delta[s(x,y), \lambda_q] = \frac{N_r(q)}{N_r} \quad (5.6)$$

où  $N_r$  est le nombre de pixels dans la région  $\Omega_r$  et  $N_r(q)$  le nombre de pixels dans la région  $\Omega_r$  pour lesquels  $s(x,y) = \lambda_q$ . Comme  $\hat{p}_r(q)$  est non biaisé, on en déduit que cet estimateur est à variance minimum. En remplaçant  $p_r(q)$  par son estimée au sens du MV  $\hat{p}_r(q)$  dans l'équation 5.4, on en déduit le nombre de *nats* nécessaires pour coder les niveaux de gris des pixels :

$$\Delta_L = - \sum_{r=1}^R \sum_{q=1}^Q N_r(q) \log \frac{N_r(q)}{N_r} . \quad (5.7)$$

Le codage de la ddp  $\{\hat{p}_r(q)\}_{q=1, \dots, Q}$  dans chacune des  $R$  régions  $\Omega_r$  conduit simplement à coder  $Q-1$  paramètres scalaires<sup>20</sup> dans chaque région. Comme la ddp est estimée dans chaque région sur un ensemble de  $N_r$  pixels, la longueur de code nécessaire pour coder  $\{\hat{p}_r(q)\}_{q=1, \dots, Q-1}$  peut être approchée par  $(Q-1) \log \sqrt{N_r}$  [79]. La longueur du code des ddp dans chacune des régions  $\Omega_r$  s'écrit donc :

$$\Delta_P = \sum_{r=1}^R \frac{Q-1}{2} \log N_r . \quad (5.8)$$

La partition  $w$  étant toujours définie à l'aide d'une grille polygonale, le nombre de *nats* nécessaires pour coder cette partition est donnée par (2.7).

La complexité stochastique d'une image à  $Q$  niveaux associée à n'importe quelle grille  $w$  est ainsi facile à déterminer. La recherche de la grille qui minimise cette complexité stochastique est obtenue comme précédemment, c'est-à-dire en alternant des phases de fusion de régions, de déplacement des nœuds et de suppression des nœuds inutiles.

20. En effet, le paramètre numéro  $Q$  n'a pas besoin d'être fourni, sachant que  $p_r(Q) = 1 - p_r(1) - p_r(2) - \dots - p_r(Q-1)$

## 5.2 Algorithme rapide

Comme nous l'avons vu dans le chapitre 2, l'estimation après chaque modification de la grille des différents paramètres des ddp est très coûteuse en temps de calcul, à moins de remplacer toutes les sommes sur des surfaces par des sommes sur les contours. Nous allons montrer que dans le cas présent, cette accélération est encore applicable.

Ainsi, dans le cas que nous étudions, les tâches les plus coûteuses en temps de calcul sont les estimations des statistiques  $N_r(q)$  nécessaires pour le calcul de (5.7). Afin de pouvoir appliquer l'algorithme rapide proposé dans la section 2.2.5, nous allons considérer une image vectorielle constituée des  $Q$  composantes à valeurs binaires  $s_q$  définies par :

$$s_q(x, y) = \delta(s(x, y), \lambda_q). \quad (5.9)$$

Nous avons alors simplement pour chaque région  $\Omega_r$

$$N_r(q) = \sum_{(x,y) \in \Omega_r} s_q(x, y) \quad (5.10)$$

ce qui montre clairement que la technique proposée en 2.2.5 peut être mise en œuvre, à condition de travailler sur l'image vectorielle  $[s_1(x, y), s_2(x, y), \dots, s_Q(x, y)]$ .

A titre d'exemple, nous présentons des résultats obtenus sur des images synthétiques. La première image de la fig. 5.1 (ligne 1) contient  $256 \times 256$  pixels prenant leur valeur parmi  $Q = 30$  valeurs possibles. Cette image est composée de 9 régions homogènes  $\Omega_r$  ayant chacune différentes ddp  $p_r(q)$  ( $q \in \{1, 2, \dots, 30\}$ ). Le résultat de la partition de la fig. 5.1.b (obtenu à partir de la grille initiale de la fig. 5.1.c) est présenté sur la fig. 5.1.d (où le résultat de la partition a été superposé à la partition vraie de l'image). Nous présentons des résultats similaires sur la fig. 5.1 (ligne 2) avec une image de  $256 \times 256$  pixels pouvant prendre  $Q = 4$  valeurs possibles et composée de 21 régions homogènes.

Soulignons de plus que grâce à l'utilisation de l'algorithme rapide qui vient d'être présenté, il est alors possible de partitionner l'image de la fig. 5.1.a en moins de 10 secondes et celle de la fig. 5.1.e en moins d'une seconde sur un PC standard sous Linux 2.4 (gcc 2.96) équipé d'un processeur Xeon cadencé à 2.8 GHz. La différence de temps de calcul entre ces deux exemples est essentiellement due au nombre de statistiques qui doivent être calculées, puisque  $Q = 30$  dans le premier cas alors que  $Q = 4$  pour le second.

## 5.3 Résultats expérimentaux

La méthode qui vient d'être présentée est adaptée aux images sur  $Q$  niveaux. De telles images résultent habituellement de pré-classifications appliquées sur des images dont les niveaux de gris peuvent varier continûment. Cette pré-classification correspond à la définition d'une application allant de l'ensemble des niveaux de gris possibles  $\mathcal{D}$  vers un ensemble discret  $\Lambda_Q$  composé de  $Q$  valeurs. L'étude de cette étape de pré-classification se situe bien au-delà des objectifs du présent manuscrit et dans la suite nous ne considérerons donc que des cas particuliers simples. Notre but sera simplement d'illustrer le type de résultats qu'il est possible d'obtenir avec la méthode de partition en régions homogènes proposée sur des images réelles à  $Q$  niveaux.

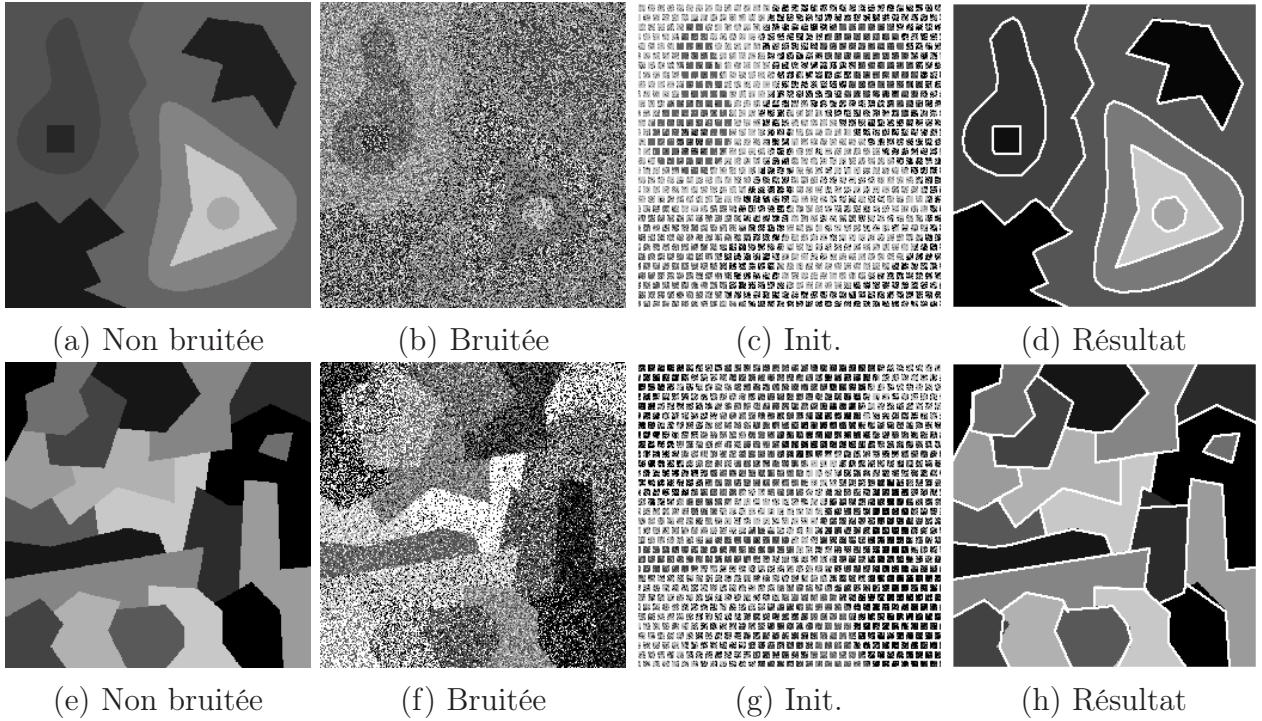


FIGURE 5.1 – *Images synthétiques contenant  $R$  régions homogènes et  $256 \times 256$  pixels prenant leurs valeurs parmi  $Q$  possibles. Ligne 1 :  $Q = 30$  et  $R = 9$ . Ligne 2 :  $Q = 4$  et  $R = 21$ . Grilles initiales (c et g) : murs de briques de taille 8. Les résultats de la partition (d et h) ont été superposés à l'image non bruitée, bien qu'ils aient évidemment été obtenus sur l'image bruitée. Soulignons que les images non bruitées (a et e) ne sont pas représentatives du contraste sur les images bruitées (b et f).*

### 5.3.1 Images pré-classifiées réelles

Nous présentons dans cette section la partition en régions homogènes d'images réelles pré-classifiées. Sur la fig. 5.2, nous avons partitionné en régions homogènes une image (composée de  $801 \times 581$  pixels) ayant été préalablement pré-classifiée en 3 classes. Cette pré-classification (fig. 5.2.b) a été obtenue en appliquant 2 seuillages sur une image SAR ENVISAT (fournie par l'ESA) de la marée noire causée par le Prestige et déjà étudiée précédemment sur la fig. 3.15.a. La partition en régions homogènes est présentée sur la fig. 5.2.c. Afin d'obtenir une partition ne contenant que trois classes, les régions obtenues sont labellisées par leur classe la plus représentée (fig. 5.2.d). Par exemple, si nous avons  $p_r(1) = 0.6$ ,  $p_r(2) = 0.3$  et  $p_r(3) = 0.1$  dans une région  $\Omega_r$ , cette région portera le label 1, la classe 1 étant la classe de plus forte probabilité.

Sur la fig. 5.3, nous présentons des résultats analogues sur une image optique aérienne ISTAR de  $512 \times 512$  pixels (fournie par le GDR ISIS). La pré-classification est obtenue à l'aide de 2 seuillages appliqués sur le module carré du Laplacien de l'image, afin d'obtenir une partition de l'image en 3 classes : zones agricoles, urbaines et forestières. Soulignons que dans la section suivante, nous présenterons une autre partition de cette image en zones homogènes mais qui ne reposera plus sur une pré-classification (voir fig. 5.9).

Enfin nous illustrons cette méthode sur la fig. 5.4, pour la détection de changements sur une image SAR 3-dates ( $187 \times 400$  pixels). La pré-classification considérée est très simple. Soit  $s_i(x, y)$  le niveau de gris du pixel  $(x, y)$  à la date  $i$  ( $i \in \{1, 2, 3\}$ ) et  $D_{i,j}(x, y) =$

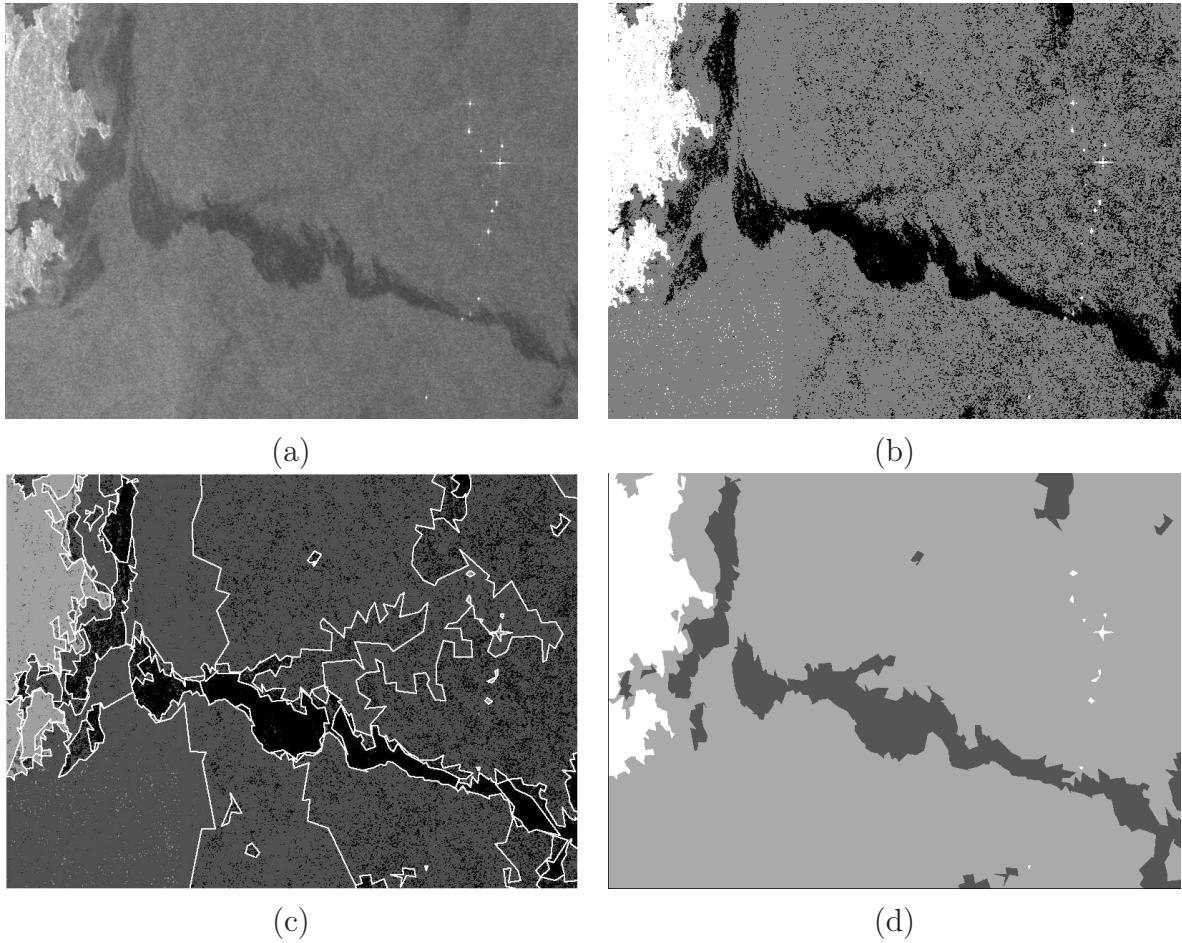


FIGURE 5.2 – Partition d'une image SAR pré-classifiée : (a) Image SAR ENVISAT (Wide Swath Mode) de la marée noire causée par le Prestige le long des côtes de Galice (Espagne) en novembre 2002 (fournie par l'ESA) et qui a déjà été étudiée précédemment sur la fig. 3.15.a. (b) Image pré-classifiée en 3 classes à l'aide de 2 seuillages. (c) Résultat de la partition en régions homogènes avec la méthode proposée. (d) Partition en 3 classes obtenue en remplaçant les niveaux de gris des pixels par le numéro de la classe la plus représentée dans chacune des régions. Grille initiale : mur de briques de taille 8.

$|\log s_i(x, y) - \log s_j(x, y)|$  avec  $i < j$ . Le seuillage des différentes images  $D_{i,j}$  conduit à 3 images binaires :  $\tilde{D}_{1,2}$ ,  $\tilde{D}_{1,3}$  et  $\tilde{D}_{2,3}$ . En additionnant ces 3 images binaires, nous obtenons l'image de la fig. 5.4.d, composée de  $Q = 4$  niveaux de gris différents. Cette image pré-classifiée permet ainsi de détecter les pixels ayant été modifiés entre les 3 dates (les pixels blancs), entre seulement 2 dates (les pixels gris) ou au contraire ceux qui n'ont pas variés (les pixels sombres). Le résultat de la partition de cette image à 4 niveaux est présenté sur la fig. 5.4.e.

Ces divers résultats illustrent les capacités de la méthode proposée, qui a notamment permis de régulariser efficacement ces différentes images pré-classifiées. Bien sûr, pour toutes ces images, de meilleures pré-classifications auraient pu être obtenues, mais rappelons que notre but est simplement de montrer l'intérêt de cette technique de partition en régions homogènes. Soulignons de plus que pour tous les exemples présentés, le processus d'optimisation est identique.

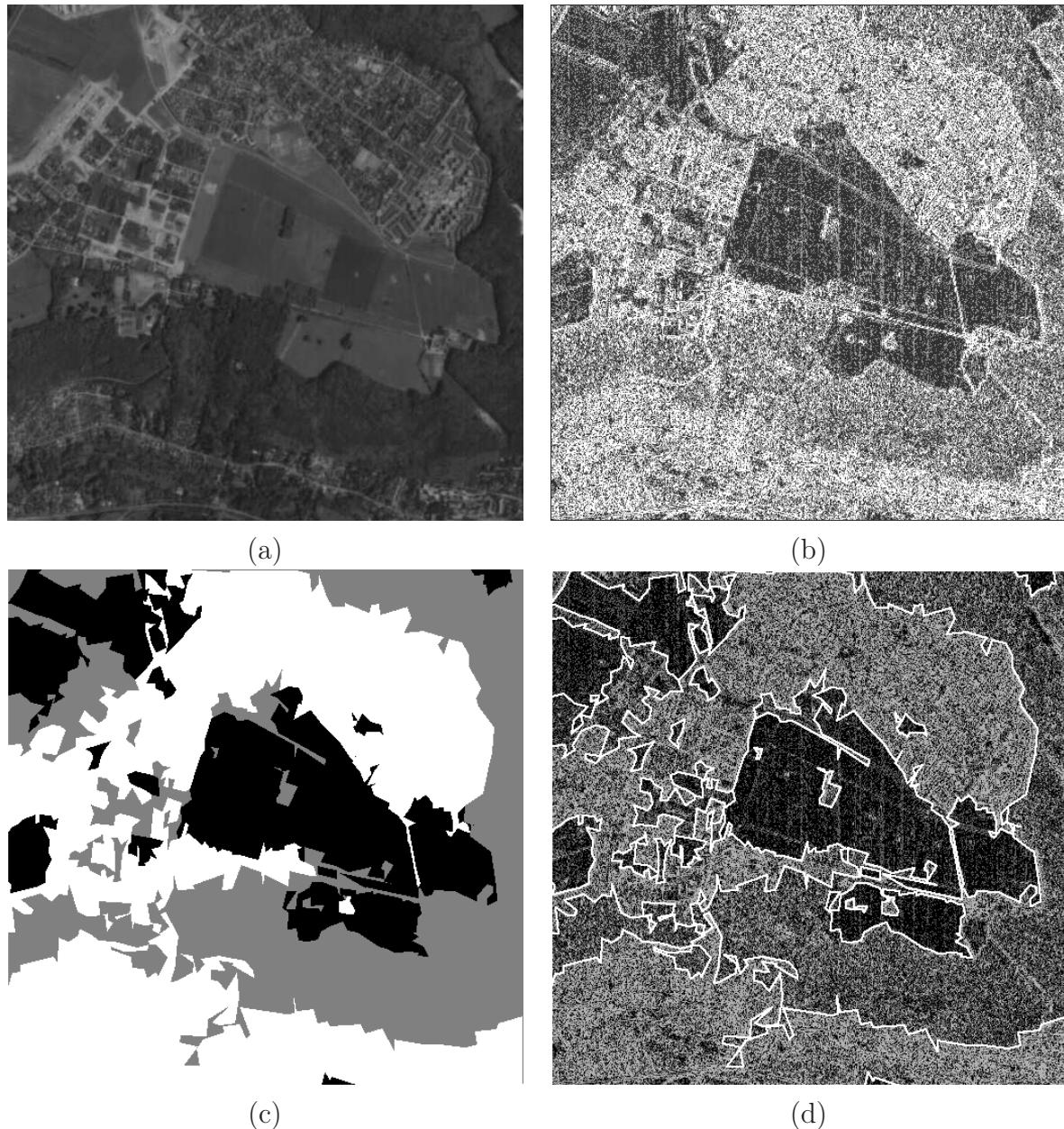


FIGURE 5.3 – Partition d'une image optique aérienne ISTAR préalablement pré-classifiée. (a) Image ISTAR contenant  $512 \times 512$  pixels (fournie par le GDR-ISIS). (b) Image pré-classifiée en 3 classes (zones agricoles, urbaines et forestières). (c) Partition de l'image pré-classifiée après avoir remplacé les niveaux de gris des pixels par le numéro de la classe majoritaire dans chaque région. (d) Grille associée à la partition (c) et affichée sur l'image (b). Grille initiale : mur de briques de taille 8.

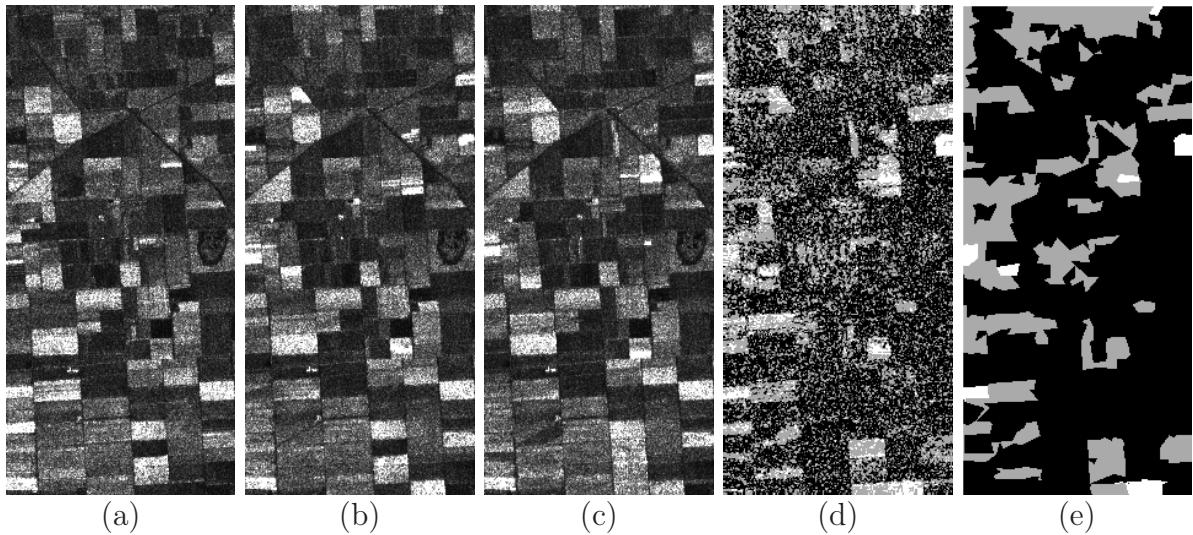


FIGURE 5.4 – Partition d'une image SAR multi-date pré-classifiée. (a,b,c) Images SAR ( $187 \times 400$  pixels) d'une zone agricole près de Bourges (France) prise à 3 dates différentes. Ces images ont été obtenues par le satellite ERS-1 (fournies par le CNES et distribuées par ESA). (d) Image pré-classifiée en 4 classes afin de détecter d'éventuels changements entre les 3 dates : cela permet de séparer les pixels dont l'intensité a subi une modification entre les 3 dates (les pixels blancs) ou entre 2 dates seulement (les pixels gris) des pixels ayant conservé la même intensité au cours des 3 prises de vue (les pixels noirs). (e) Résultat de la partition en régions homogènes de l'image pré-classifiée après avoir remplacé les niveaux de gris des pixels par le numéro de la classe la plus représentée dans la région. Grille initiale : mur de briques de taille 8.

### 5.3.2 Quantification des niveaux de gris

Nous proposons de discuter dans cette section des résultats obtenus en utilisant simplement une quantification des niveaux de gris de l'image. Lorsque l'ensemble  $\mathcal{D}$  des valeurs que peuvent prendre les pixels d'une image est borné et correspond à l'intervalle  $[a, b]$ , la quantification de cette image sur  $Q$  niveaux peut être effectuée simplement en associant la valeur entière  $q$  au niveau de gris  $x$  si  $x \in [a + (q - 1)h, a + qh[$  et  $q = Q$  si  $x = b$ , où l'on a utilisé  $h = (b - a)/Q$ . Lorsque  $\mathcal{D}$  n'est plus borné, il suffit d'appliquer aux niveaux de gris  $x$  une transformation non linéaire  $y = f(x)$  de façon à ce que l'ensemble des valeurs que peut prendre  $y$  soit borné. Il est alors possible d'appliquer la procédure de quantification précédemment décrite. L'égalisation d'histogramme est un exemple d'une telle transformation non linéaire. Si  $h(x)$  est l'histogramme de l'image considérée, l'égalisation d'histogramme correspond à la transformation  $y(x) = \int_{-\infty}^x h(\xi)d\xi$ . Dans les sections suivantes, nous allons illustrer ces deux techniques de quantification sur divers exemples.

Nous allons en premier lieu nous intéresser à des images synthétiques. Dans cet objectif, nous avons utilisé dans la fig. 5.5 la même image synthétique que pour la fig. 5.1.a. Sur la première ligne (respectivement, seconde ligne) de la fig. 5.5, les niveaux de gris dans chaque région sont distribués selon une ddp gamma d'ordre  $L = 1$  (respectivement, selon une ddp de Poisson) avec différentes moyennes. Il est donc possible d'utiliser les log-vraisemblances fournies dans le tableau 4.2 du chapitre 4 afin de partitionner chaque image tour à tour avec un critère optimal pour une ddp gamma d'ordre  $L = 1$  (colonne b) ou pour une ddp de Poisson (colonne c). Nous voyons alors clairement sur la fig. 5.5 (colonne b) que l'utilisation d'un critère adapté à une loi gamma conduit à une bonne partition de l'image de la première ligne (perturbée par un bruit gamma) mais aboutit à de mauvais résultats sur l'image de la deuxième ligne (perturbée par un bruit de Poisson). Nous observons évidemment les conclusions inverses lorsque l'on utilise un critère adapté à une loi de Poisson (voir colonne c). Nous présentons enfin sur la colonne (d) de la fig. 5.5 les résultats obtenus en utilisant la méthode proposée dans cette partie et qui ne nécessite pas de connaissance *a priori* de la famille des ddp des niveaux de gris. Pour obtenir les résultats de la dernière colonne, nous avons quantifié l'image sur  $Q = 10$  niveaux de gris après avoir appliqué une égalisation d'histogramme. Nous pouvons constater que la technique proposée conduit à des résultats satisfaisants, dans la mesure où ils sont comparables à ceux obtenus en utilisant un critère adapté au bruit.

Un des principaux avantages de la technique proposée dans ce chapitre est sa capacité à gérer des régions dont les niveaux de gris sont distribués selon différentes ddp, n'appartenant d'ailleurs pas nécessairement à la famille exponentielle. Afin d'illustrer cette propriété, nous avons synthétisé, à partir de l'image non bruitée de la fig. 5.6.a, une image dont les pixels sont distribués selon différentes ddp ayant même moyenne et même variance (voir fig. 5.6.b). La partition obtenue est présentée sur la fig. 5.6.c en utilisant une égalisation d'histogramme suivie d'une quantification sur  $Q = 10$  niveaux. Bien évidemment, comme tous les critères présentés dans le tableau 4.2 du chapitre 4 reposent sur des estimations de moyennes ou de variances, aucun ne permet de détecter une quelconque région. Nous montrons enfin sur la ligne 2 de la fig. 5.6, le résultat obtenu sur une image dont les niveaux de gris sont distribués selon des ddp gaussiennes, gamma mais aussi des ddp de Cauchy, loi dont la moyenne et la variance ne sont pas définies.

Pour mieux comprendre ces résultats, il suffit de constater que la quantification d'une

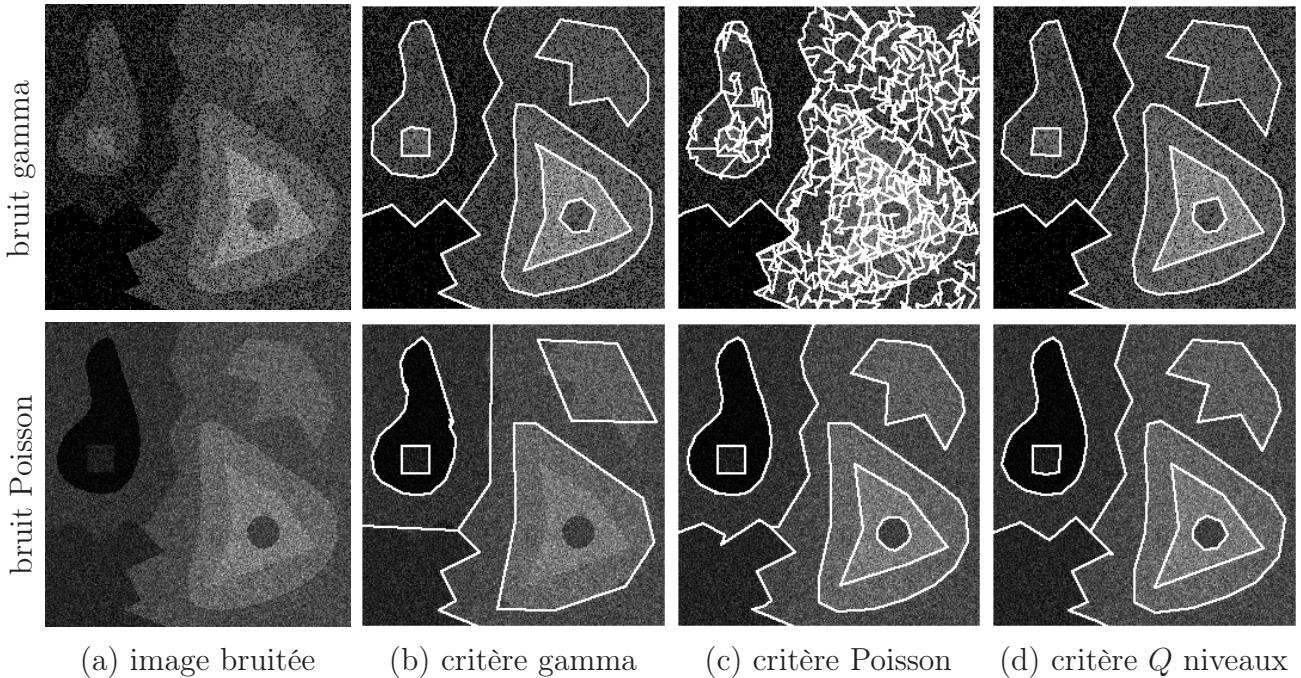


FIGURE 5.5 – Comparaison des partitions obtenues avec un critère adapté à des images comportant  $Q$  niveaux et un critère adapté à des fluctuations de niveaux de gris décrits par des lois gamma ou de Poisson. L'image est perturbée par un bruit gamma d'ordre  $L = 1$  (ligne 1) et un bruit de Poisson (ligne 2). Ces images sont ensuite partitionnées en utilisant 3 critères différents : un critère gamma (b), un critère Poisson (c) et un critère adapté à des images possédant  $Q = 10$  niveaux lorsque les niveaux de gris de l'image ont été quantifiés après égalisation d'histogramme (d). Pour toutes ces expériences, nous avons utilisé une grille initiale en mur de briques de taille 8 pixels.

image sur  $Q$  niveaux revient en fait à approcher son histogramme à l'aide de seulement  $Q$  quantiles. L'égalisation d'histogramme intervenant avant l'étape de quantification permet simplement d'obtenir une image avec un histogramme quasiment plat. A l'issu de ces deux étapes, nous obtenons donc une approximation de l'histogramme de l'image à l'aide de  $Q$  quantiles contenant tous environ le même nombre de pixels. Le principal avantage de cette technique d'approximation d'histogramme est sa très grande simplicité, permettant à la méthode de partition qui en découle de pouvoir toujours utiliser l'algorithme rapide présenté en partie 2.2.5 lors de l'estimation des paramètres de l'histogramme dans chaque région.

Nous présentons sur la fig. 5.7 la partition en régions homogènes de deux images SAR de zones agricoles près de Bourges (France) fournies par le satellite ERS-1. Cette technique de partition a aussi été appliquée sur des images de cristaux photoniques auto-assemblés [25] obtenues par microscopie électronique à balayage (fig. 5.8) puis sur la même image optique aérienne ISTAR que la fig. 5.3 (fig. 5.9) mais sans avoir effectué de pré-classification. Soulignons que pour toutes les images présentées dans cette section, la même procédure a été utilisée, c'est-à-dire que les niveaux de gris ont été quantifiés sur  $Q = 10$  valeurs après avoir effectué une égalisation d'histogramme.

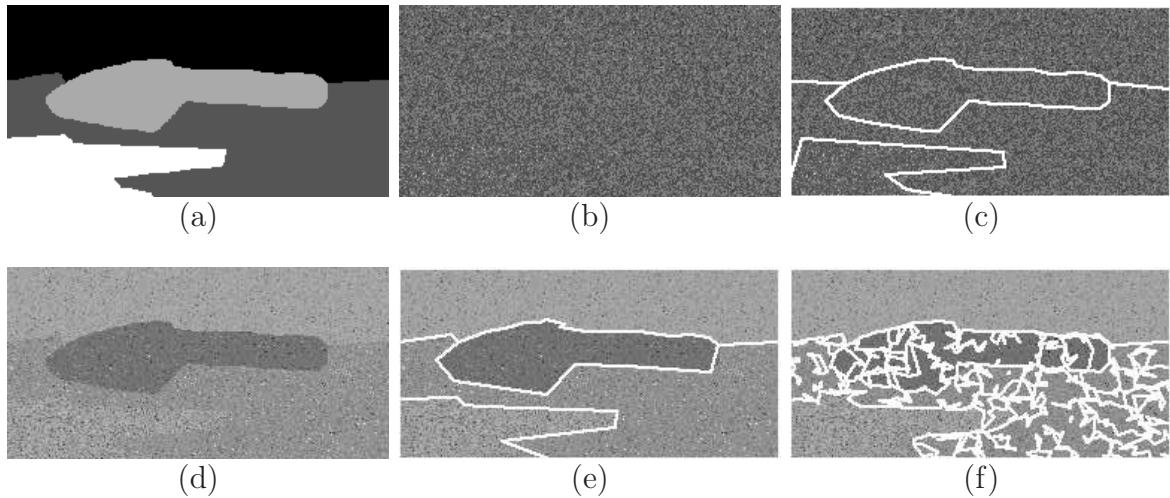


FIGURE 5.6 – Partition d’images perturbées par des ddp non standards. (a) Image non bruitée ( $256 \times 128$  pixels). (b) Image (a) bruitée : les 4 régions ont été perturbées par une ddp (de haut en bas) Gaussienne, bivaluée, uniforme et gamma d’ordre  $L = 1$ , mais toutes possèdent la même moyenne ( $m = 1.5$ ) et le même écart-type ( $\sigma = 0.29$ ). (c) Partition de (b) en utilisant un critère avec  $Q = 10$  niveaux après avoir effectué une égalisation d’histogramme. (d) Image (a) bruitée : les 4 régions ont été perturbées par une ddp (de haut en bas) Gaussienne, Cauchy, Cauchy et gamma d’ordre  $L = 1$ . (e) Partition de (d) en utilisant un critère avec  $Q = 10$  niveaux après avoir effectué une égalisation d’histogramme. (f) Partition de (d) en utilisant un critère gaussien. Pour toutes ces expériences, nous avons utilisé une grille initiale en mur de briques de taille 6 pixels.

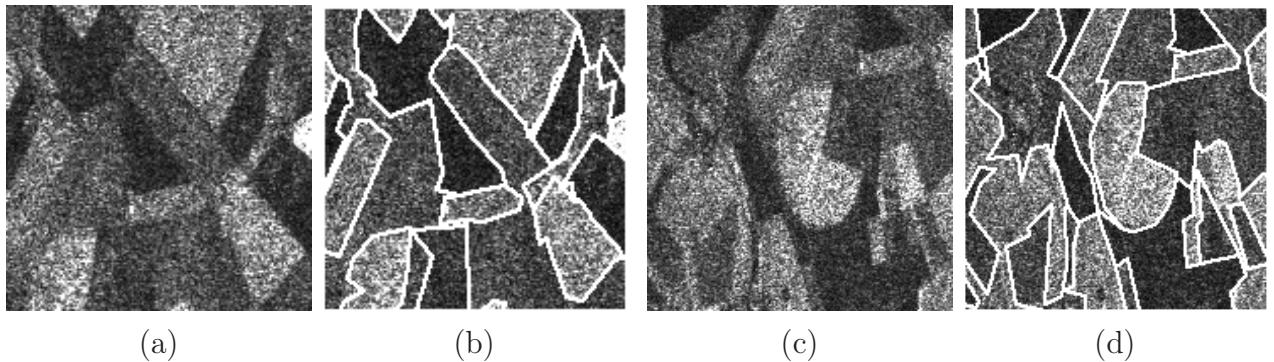


FIGURE 5.7 – Partition d’images SAR de zones agricoles près de Bourges (France) obtenues par le satellite ERS-1 (images fournies par le CNES et distribuées par l’ESA). (a,c) Images SAR ( $151 \times 151$  et  $171 \times 171$  pixels) - (b,d) Partition de (a) et (c) après avoir effectué une égalisation d’histogramme puis quantifié l’image sur  $Q = 10$  niveaux. Grilles initiales : murs de briques de taille 8.

### 5.3.3 Images couleurs et polarimétriques

La méthode qui vient d’être utilisée peut être aisément généralisée au cas des images couleurs. Pour cela, afin d’être moins sensible aux effets de fluctuations d’intensité dues à l’éclairement et à la présence d’ombres, nous nous plaçons dans l’espace de couleur YHS

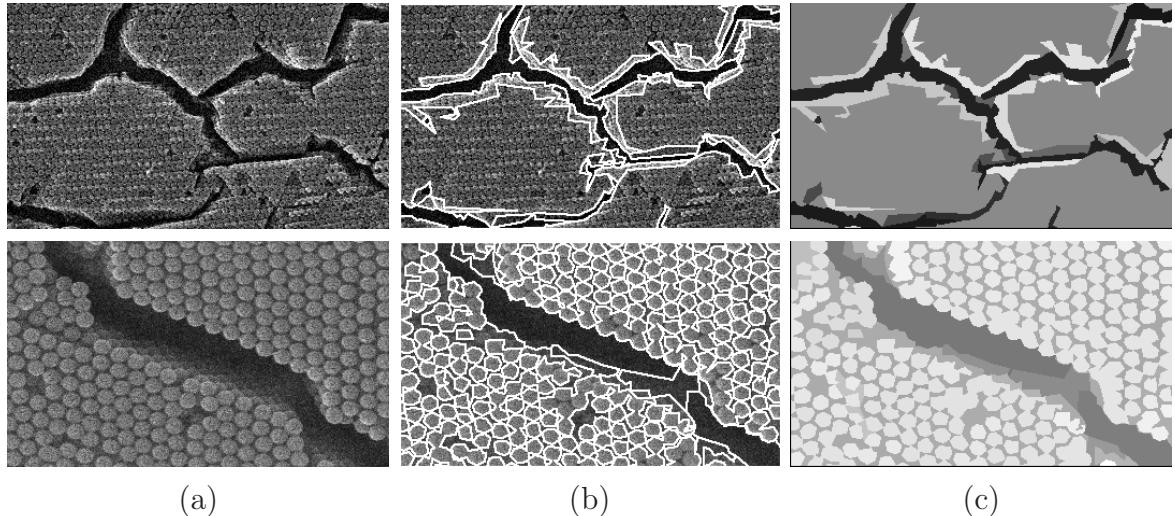


FIGURE 5.8 – Partition d'une image de cristaux photoniques auto-assemblés obtenus par microscopie électronique à balayage (images fournies par P. Ferrand de l'université de Wuppertal). (a) Images (356 × 210 pixels) - (b) Partition obtenue avec une quantification sur  $Q = 10$  niveaux après avoir égalisé les histogrammes - (c) Les niveaux de gris dans chaque région ont été remplacés par la valeur moyenne de la région. Grilles initiales : murs de briques de taille 4.

plutôt que dans l'espace RVB usuel [83]. Dans cet espace, la composante Y représente la luminance, H la teinte (*hue* en Anglais) et S la saturation. La partition en régions homogènes est effectuée seulement sur la composante H, ce qui permet de conserver l'information couleur tout en s'affranchissant des problèmes d'éclairage. Dans l'espace YHS, cette composante H est analogue à un angle et est donc  $2\pi$ -périodique, ce qui signifie que des valeurs élevées (proches de  $\pi$ ) et des valeurs faibles (proches de  $-\pi$ ) correspondent à la même couleur. Nous présentons sur la fig. 5.10 (ligne 1) la partition en régions homogènes de l'image couleur d'une disquette informatique. Cette partition a été obtenue en quantifiant sur  $Q = 10$  niveaux la composante H de cette image couleur. Soulignons que puisque l'ensemble de valeurs que peuvent prendre les niveaux de gris est borné ( $\mathcal{D} = ]-\pi, \pi[$ ), il n'a pas été nécessaire d'effectuer d'égalisation d'histogramme.

En utilisant la polarisation de la lumière, il est également possible de déterminer l'angle de l'état principal de polarisation de la lumière réfléchie [43, 89]. On obtient ainsi en chaque pixel de l'image, un angle qui est fonction de l'orientation de la surface 3D de l'objet observé. Cette image a donc les mêmes propriétés que la composante H des images couleurs : la partition d'une telle image est présentée sur la fig. 5.10 (ligne 2), en utilisant la même procédure que dans le cas des images couleurs (toujours en quantifiant sur  $Q = 10$  niveaux).

### 5.3.4 Contrainte de la topologie par contours actifs

Comme nous l'avons vu dans le chapitre précédent, il est possible d'imposer une contrainte sur la topologie de la grille en utilisant des contours actifs mono ou multi-régions.

Nous avons ainsi utilisé sur la fig. 5.11, l'algorithme décrit au chapitre 4.3 pour segmenter successivement une image couleur (ligne 1) et une image sonar (ligne 2).

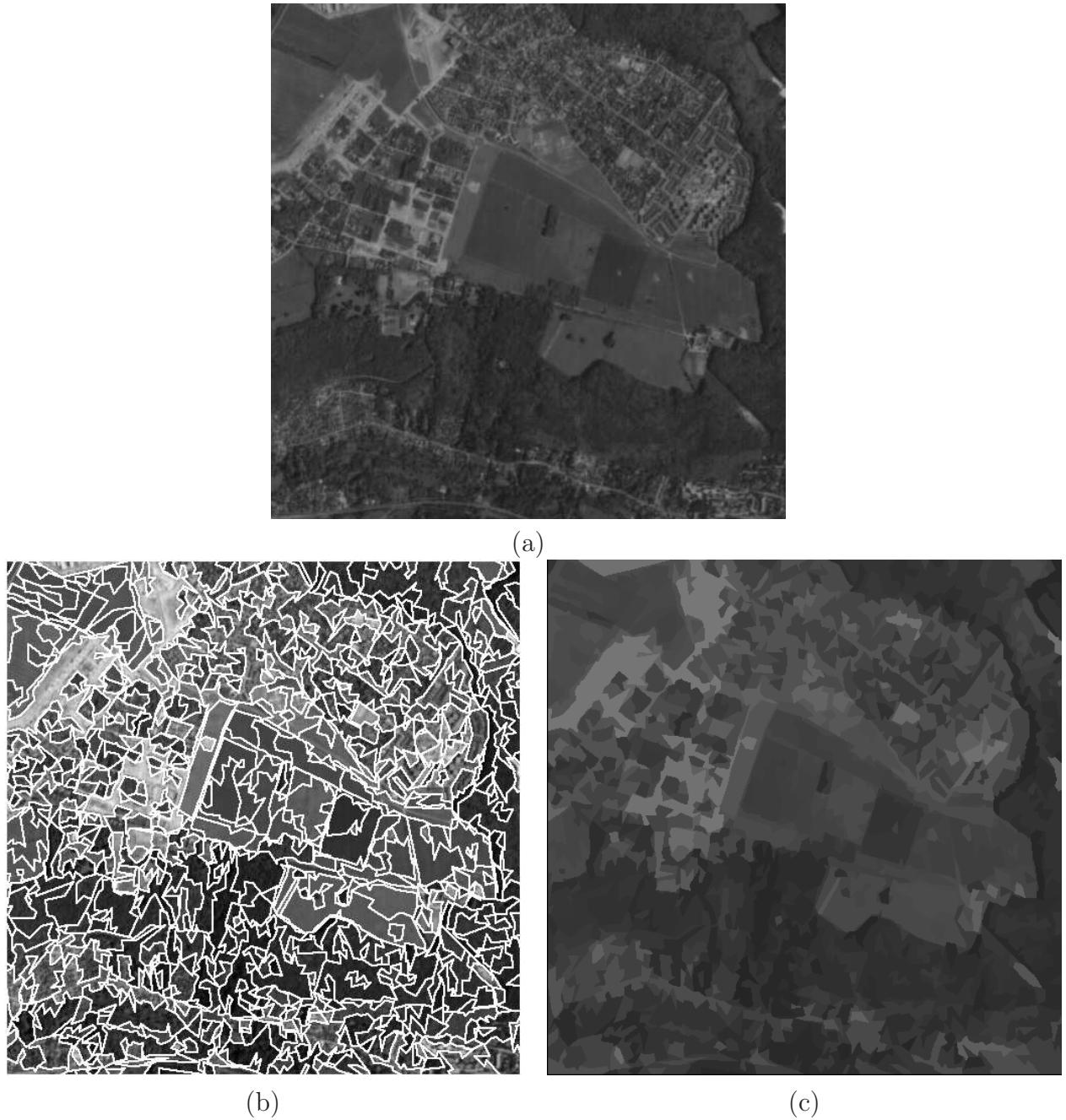


FIGURE 5.9 – Partition d'une image optique aérienne ISTAR (la même que sur la fig. 5.3). (a) Image ISTAR comportant  $512 \times 512$  pixels (fournie par le GDR-ISIS) - (b) Partition obtenue après égalisation d'histogramme et quantification sur  $Q = 10$  niveaux - (c) Les niveaux de gris dans chaque région ont été remplacés par la valeur moyenne de la région. Grille initiale : mur de briques de taille 8.

## 5.4 Conclusion

Dans ce chapitre, nous avons dans un premier temps proposé une généralisation du modèle aux images prenant leurs valeurs parmi un ensemble de taille  $Q$ , non nécessairement ordonné. Nous avons ensuite montré qu'il était possible d'estimer la loi de probabilité des niveaux de gris dans chaque région, et donc d'obtenir l'expression de la complexité stochastique. L'algo-

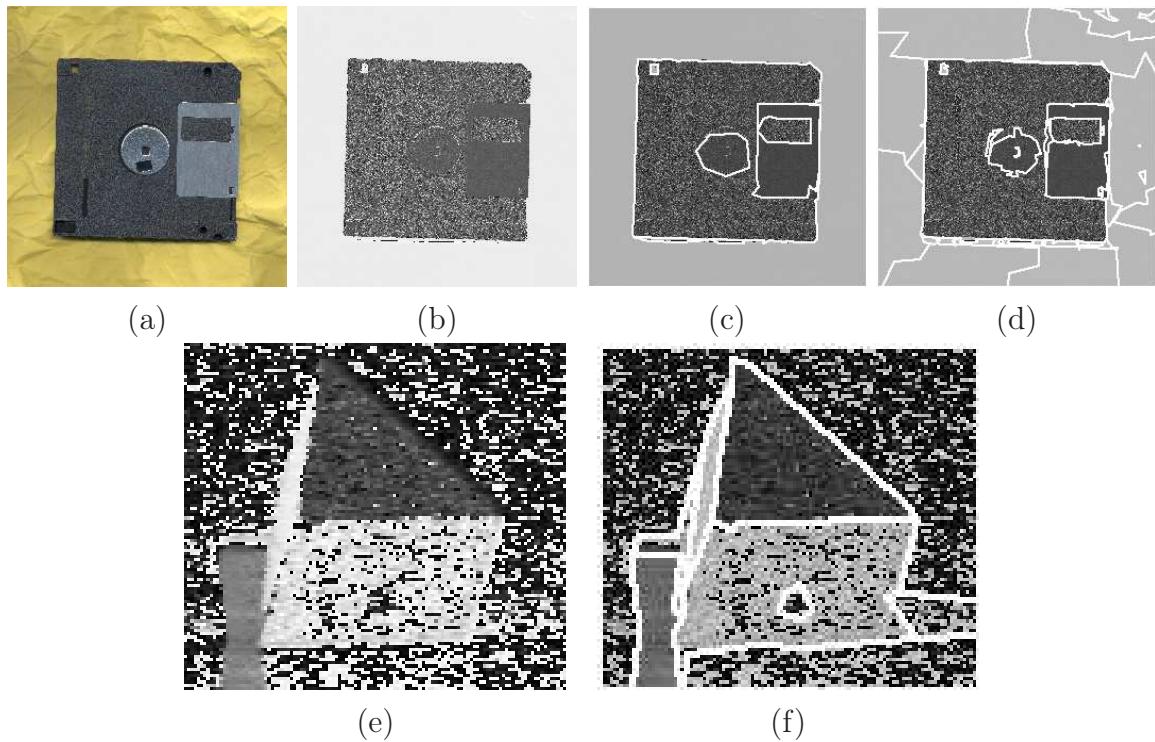


FIGURE 5.10 – *Images couleur et polarimétrique. Ligne 1 : Partition d'une image couleur (264 × 264 pixels) d'une disquette informatique à partir de la composante de chrominance  $H$  : (a) image couleur - (b) teinte  $H$  - (c) partition de l'image (b) quantifiée sur  $Q = 10$  niveaux - (d) partition de (b) en utilisant un critère optimal pour un bruit gaussien. Ligne 2 : Partition d'une image d'angle de polarisation (119 × 108 pixels) : (e) image d'angle de polarisation - (f) partition de l'image (e) quantifiée sur  $Q = 10$  niveaux de gris. Grilles initiales : murs de briques de taille 3 pour la première ligne et de taille 4 pour la deuxième.*

rithme de partition obtenu a alors été testé sur plusieurs images réelles pré-classifiées.

Nous avons mis en œuvre dans un deuxième temps une procédure systématique de pré-classification, consistant tout d'abord à égaliser l'histogramme de l'image (sauf si le domaine de définition des niveaux de gris de l'image est déjà borné<sup>21</sup>), avant de le quantifier uniformément sur  $Q = 10$  niveaux. Cette pré-classification permet ainsi une approximation - assez grossière - de l'histogramme de l'image sur  $Q = 10$  quantiles, l'égalisation d'histogramme ayant permis que tous les quantiles contiennent environ le même nombre de pixels. Nous avons alors montré que cette pré-classification permettait d'obtenir de bons résultats de partition, similaires à ceux obtenus en utilisant un critère adapté aux ddp présentes dans l'image, sans pour autant avoir nécessité la connaissance de la ddp du bruit présent dans l'image. La méthode de partition ainsi obtenue, robuste, nous a alors permis de partitionner différentes images ayant des ddp inconnues et différentes dans chaque région et n'appartenant pas nécessairement à la famille exponentielle, c'est-à-dire des images qu'il n'aurait pas été possible de partitionner avec les approches paramétriques utilisées dans les chapitres précédents.

21. Soulignons cependant que même si les niveaux de gris de l'image sont bornés, rien n'empêche l'utilisation d'une égalisation d'histogramme : les résultats obtenus peuvent dans certains cas être de meilleure qualité.

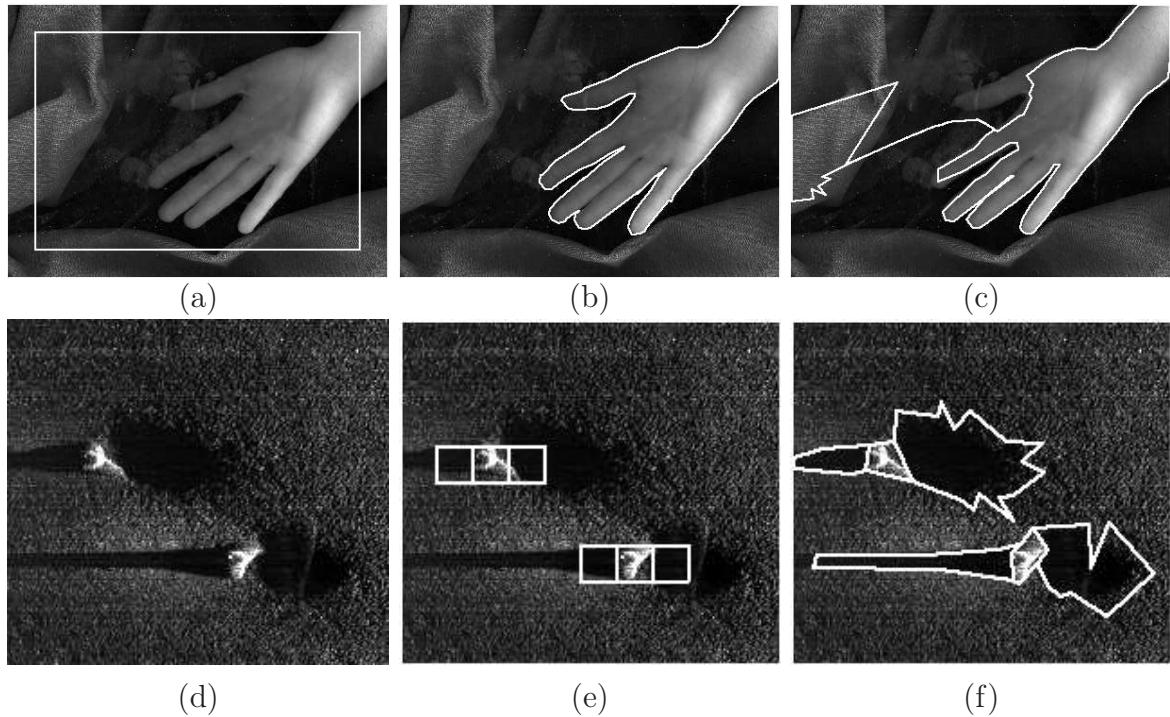


FIGURE 5.11 – Segmentation par contours actifs : Ligne 1 : Segmentation par contour actif mono-région d’une image couleur ( $405 \times 291$  pixels) à partir de sa composante  $H$  (teinte) dans l’espace de couleur YHS : (a) image en niveaux de gris (composante  $Y$ ) et contour initial - (b) segmentation de la composante  $H$  quantifiée sur  $Q = 10$  niveaux - (c) segmentation de la composante  $Y$  en utilisant un critère gaussien. Soulignons que les résultats (b et c) sont affichés sur la composante  $Y$ . Ligne 2 : Segmentation par contours actifs multi-régions d’une image sonar ( $212 \times 192$  pixels) fournie par l’ONR et le SACLANT Center : (d) image - (e) contours initiaux - (f) segmentation après égalisation d’histogramme et quantification sur  $Q = 10$  niveaux.



# Conclusion et perspectives

Au début de ce travail de thèse, différentes méthodes reposant sur des contours actifs statistiques polygonaux avaient été développées et il était ainsi possible de segmenter rapidement des objets simplement connexes, à condition que le nombre d'objets présents dans l'image, leur topologie et le type de ddp caractérisant leurs niveaux de gris soient connus. Au cours de cette thèse, nous nous sommes attachés à relaxer ces différentes contraintes.

Nous avons dans un premier temps montré que l'utilisation du principe de la minimisation de la complexité stochastique nous permettait d'estimer simultanément le nombre de régions présentes dans l'image, leur topologie ainsi que la position et la régularité de leurs contours. De plus, l'accélération d'algorithme proposée dans [16, 38], adaptée jusqu'à présent uniquement aux objets simplement connexes, a pu être généralisée à des objets de topologie quelconque. La technique de partition obtenue, qui repose sur l'optimisation d'un critère sans paramètre à régler de la part de l'utilisateur, permet ainsi de partitionner des images de  $256 \times 256$  pixels en moins d'une seconde sur un PC standard équipé d'un processeur cadencé à 2.8 GHz.

Dans les premières études de cette thèse (présentées dans les chapitres 2 à 4), cette approche reposait sur la connaissance de la loi de probabilité caractérisant les niveaux de gris de l'image, ce qui nous permettait notamment d'obtenir un algorithme adapté aux images SAR. Nous avons alors montré (voir le chapitre 5), qu'il était également possible d'utiliser une méthode non paramétrique permettant la partition d'images sans connaissance *a priori* de la loi de probabilité. Cette approche, robuste, permet alors de partitionner différentes images dont la loi de probabilité peut être inconnue, différente dans chaque région et ne pas appartenir à la famille exponentielle.

Dans ce manuscrit, nous avons également cherché à généraliser cette technique, initialement présentée dans le cadre de la partition d'images SAR scalaires modélisées à l'aide de ddp Gamma, à d'autres ddp de la famille exponentielle et à des images vectorielles dont les différentes composantes peuvent être corrélées ou non. Par ailleurs, nous avons montré que lorsque l'utilisateur dispose de certaines connaissances *a priori* sur l'image qu'il cherche à segmenter, il lui est possible d'injecter ces informations dans le modèle, améliorant ainsi les performances de l'algorithme. Cet aspect a particulièrement été mis en avant dans le cas de la segmentation par contours actifs multi-régions, où une connaissance forte sur le nombre de régions et leur topologie est injectée dès l'initialisation. Toujours dans le cadre de contours actifs, nous avons également vu l'intérêt que pouvait apporter la connaissance des paramètres de la cible, par exemple dans le cas de la poursuite d'un objet, ou la connaissance des familles de ddp lorsqu'elles ne sont pas nécessairement identiques dans chaque région.

De plus, l'utilisation du principe de minimisation de la complexité stochastique nous a permis d'estimer automatiquement différents paramètres inconnus, comme la taille de la grille initiale, l'ordre du speckle et la valeur des seuils intervenant dans l'étape de classification par seuillage utilisée notamment pour la segmentation de nappes de pétrole sur des images SAR.

Il serait bien entendu nécessaire de mieux quantifier l'apport de chacun des points évoqués précédemment. Cependant, dans notre travail nous avons préféré tout d'abord identifier des applications pratiques. Nous avons ainsi travaillé avec le laboratoire I3D de l'université de Lille I, dans le cadre d'un projet "jeunes chercheurs" au sein du GDR-ISIS et dont le sujet était la segmentation d'images polarimétriques passives. Cette étude, non présentée dans ce manuscrit, a notamment abouti à une publication [AI.3] et une communication [CF.1] (voir annexe C). Nous avons également établi une collaboration très fructueuse avec la société Starlab de Barcelone, qui a conduit à l'approche présentée dans la partie 3.3 relative à la segmentation de nappes de pétrole sur des images SAR [AI.4, CI.4] (voir annexe C). Nos travaux ont permis d'établir une collaboration avec l'ONERA qui a débouché sur la mise en place d'une thèse portant sur l'imagerie SAR polarimétrique interférométrique. Enfin, nos travaux sont à la base de la collaboration avec le groupe Thalès Communications France (TCF), dont le sujet est l'étude d'applications de notre méthode de partition aux images radars avec l'introduction de connaissances de haut niveau.

## Perspectives

Comme nous venons de le souligner, la première perspective est de caractériser la qualité de la partition obtenue par la méthode proposée. Cependant, cette tâche s'avère plus difficile qu'il n'y paraît à première vue, dans la mesure où la définition d'un critère de qualité pertinent est non triviale. Ainsi, si dans le cas de contours actifs mono-régions, l'utilisation du nombre de pixels mal classés ou de distances de Hausdorff [44] ont permis notamment à Germain [33] et à Ruch [81] de caractériser les performances du contour actif statistique polygonal, cette approche n'est pas directement transposable dans le cas où le nombre de régions est inconnu. En effet, dans la mesure où une erreur peut être commise lors de l'estimation du nombre de régions, le résultat de la partition et la vérité terrain ne comportent plus nécessairement le même nombre de régions. Il est donc nécessaire d'utiliser d'autres critères de qualité, par exemple le nombre de régions détectées, comme nous l'avons fait dans la partie 4.2.2 (fig. 4.18), ou la mesure de Vinet [17]. Ce problème important est abordé dans le cadre d'une thèse en cours au laboratoire.

Une seconde perspective, très prometteuse, est relative aux travaux du chapitre 5, concernant la partition d'images égalisées puis quantifiées sur  $Q$  niveaux. En effet, sur les exemples considérés, l'algorithme obtenu, de type non paramétrique, semble donner des résultats comparables à ceux obtenus avec une méthode paramétrique où l'on impose une ddp adaptée. Il est cependant fort probable que la méthode optimale donne de meilleurs résultats à faible contraste ou pour des régions de petite taille. Une étude quantitative des inconvénients et des avantages de chacune de ces deux méthodes s'impose donc.

De plus, l'étape de pré-classification, qui permet une approximation de l'histogramme de l'image, repose sur une égalisation d'histogramme suivie d'une quantification sur  $Q = 10$  niveaux. Il serait sans doute intéressant d'estimer automatiquement cette valeur de  $Q$ .

Dans le cadre du contour actif multi-régions, plusieurs perspectives sont également envisageables. En premier lieu, nous avons rapidement évoqué son intérêt pour la poursuite de cibles dans le cas de ddp gaussiennes et l'approche consistant à injecter les paramètres de la cible dans le modèle semble prometteuse. Outre la nécessité de quantifier l'apport de cette méthode, il pourra également être intéressant de nous placer, non plus dans le cas de ddp gaussiennes

---

mais dans le cas de ddp non paramétriques, c'est-à-dire reposant sur une quantification de l'image sur  $Q$  niveaux. La cible serait alors définie à l'aide de  $Q$  paramètres et non plus seulement à l'aide de sa moyenne et de sa variance comme c'était le cas avec des ddp gaussiennes. De plus, il sera peut-être aussi possible de choisir en fonction de la cible considérée, la façon de quantifier l'image et le nombre de niveaux  $Q$  à considérer afin de limiter la dégradation de son contraste par rapport au fond de l'image.

Un autre point à développer est l'utilisation conjointe de la partition en régions homogènes avec un algorithme de classification, comme dans le cadre de la segmentation de nappes de pétrole, mais avec cette fois-ci un algorithme de classification plus élaboré qu'un simple seuillage. Cela permettrait en effet de vérifier si l'utilisation de la complexité stochastique permet toujours l'estimation automatique des différents paramètres contenus dans l'algorithme de classification, comme c'était le cas pour le seuillage.

Parallèlement à ce type de méthodes reposant sur des post-traitements nécessaires lors de la segmentation d'images plus complexes, il serait également intéressant de pouvoir prendre en compte une corrélation des grains de speckle ou la présence d'une texture sous-jacente. Soulignons cependant que ce n'est pas nécessairement en complexifiant les modèles que l'on obtient les meilleurs résultats, comme le démontre d'ailleurs clairement le principe de la minimisation de la complexité stochastique.



# Annexe A

## Calculs et algorithmes annexes

### Sommaire

---

<b>A.1</b>	<b>Détermination du terme de codage de la grille <math>\Delta_G(w)</math></b>	<b>128</b>
<b>A.2</b>	<b>Détermination de la log-vraisemblance généralisée d'une loi gamma d'ordre <math>L</math></b>	<b>131</b>
<b>A.3</b>	<b>Algorithme de minimisation de la complexité stochastique</b>	<b>132</b>
A.3.1	Fusion de régions	132
A.3.2	Déplacement des nœuds	132
A.3.3	Suppression des nœuds	133
A.3.4	Temps de calcul	134
<b>A.4</b>	<b>Définition non ambiguë de l'intérieur des régions par translation de la grille</b>	<b>137</b>
<b>A.5</b>	<b>Algorithme rapide : présentation, généralisation et mise en œuvre</b>	<b>140</b>
A.5.1	Rappels	140
A.5.2	Généralisation aux régions non simplement connexes	140
A.5.3	Mise en œuvre	142
A.5.4	Gestion de la topologie	143
<b>A.6</b>	<b>Cas où la moyenne n'est plus constante pas morceaux</b>	<b>145</b>
A.6.1	ddp gaussienne et espérance mathématique distribuée selon une fonction linéaire	145
A.6.2	ddp gaussienne et espérance mathématique distribuée selon une fonction quadratique	146
A.6.3	ddp gamma et espérance mathématique distribuée selon une fonction linéaire	147
<b>A.7</b>	<b>Contour actif et transmission des paramètres de la cible</b>	<b>149</b>
<b>A.8</b>	<b>Estimation de la ddp d'une image à <math>Q</math> niveaux</b>	<b>150</b>

---

## A.1 Détermination du terme de codage de la grille $\Delta_G(w)$

Dans cette partie, nous allons détailler l'obtention du terme de codage de la grille  $\Delta_G$  fourni dans l'équation 2.7.

Afin de coder une grille, décomposons-la dans un premier temps en  $n$  sous-graphes Eulériens<sup>22</sup>, où  $n$  est le nombre minimum de graphes Eulériens nécessaires pour décrire la grille entièrement. De façon plus intuitive,  $n$  est le nombre minimum de fois où il faut lever le crayon pour dessiner la grille sans jamais retracer deux fois le même segment. La valeur de  $n$  est donnée par la formule suivante :

$$n = \frac{n_{IMP}}{2} + n_{CI} \quad (\text{A.1})$$

où  $n_{IMP}$  est le nombre de noeuds de multiplicité impaire (c'est-à-dire reliés à un nombre impair de segments) et où  $n_{CI}$  est le nombre de graphes isolés ne comportant aucun noeud de multiplicité impaire<sup>23</sup> (fig. A.1).

Comme un graphe Eulérien peut être parcouru continûment en passant une et une seule fois par ses segments, le codage de chacun des  $n$  graphes Eulériens peut être divisé en trois étapes :

- a) Codage du noeud de départ : comme l'image contient  $N$  pixels, cela peut être fait avec  $\log(N)$  nats.
- b) Codage du nombre de segments appartenant au graphe Eulérien : ce nombre pouvant varier entre 1 et  $p$  (où  $p$  est le nombre de segments dans la grille), il est possible de le coder en première approximation avec  $\log(p)$  nats.
- c) Codage de la liste des segments du graphe Eulérien. Ces segments étant tous reliés, il suffit pour cela de donner une liste de vecteurs chaînés<sup>24</sup>.

Pour coder la grille entière, il faut finalement coder  $n$  noeuds de départ,  $n$  nombres de segments, ainsi que les vecteurs associés aux segments de chacun des  $n$  graphes Eulériens, c'est-à-dire au nombre total de segments dans la grille, noté  $p$ .

Détaillons maintenant le codage de ces  $p$  vecteurs associés aux segments, appelés dans la suite vecteurs du graphe.

Il est évident que le codage de ces vecteurs du graphe peut être effectué en  $p \log(N)$  nats<sup>25</sup>. Cependant, ceci ne constitue pas une bonne estimation du nombre minimum de nats nécessaires au codage de ces vecteurs, dans la mesure où cette approximation repose sur l'hypothèse de nœuds uniformément répartis sur l'image, indépendamment de la position des nœuds auxquels ils sont reliés.

Une meilleure approximation peut être obtenue en utilisant un code entropique [84] pour les coordonnées des vecteurs du graphe. Pour déterminer la longueur de ce code, il faut au

22. Un graphe Eulérien est un graphe qui peut être parcouru continûment sans avoir parcouru deux fois le même segment.

23. L'équation (A.1) s'obtient en remarquant qu'un noeud d'ordre impair doit nécessairement être le noeud d'arrivé ou de départ d'un graphe Eulérien. On obtient ainsi  $n_{IMP}/2$  noeuds de départ. Mais il peut arriver qu'une composante de graphe isolée ne comporte aucun noeud d'ordre impair. Il faut donc ajouter  $n_{CI}$  noeuds de départ, ce qui nous conduit à  $n = \frac{n_{IMP}}{2} + n_{CI}$ .

24. Il est donc inutile de donner les coordonnées des deux extrémités de chaque segment.

25. Pour définir les segments de chacun des graphes Eulériens, il suffit de donner la liste des coordonnées des nœuds dans l'ordre où on les rencontre lors du parcours de ce graphe. En supposant toutes les positions de nœuds équiprobables (et indépendantes de la position du nœud précédent), chaque coordonnée de nœuds coûte  $\log(N)$  nats, ce qui donne bien  $p \log(N)$  nats pour le codage des  $p$  segments.

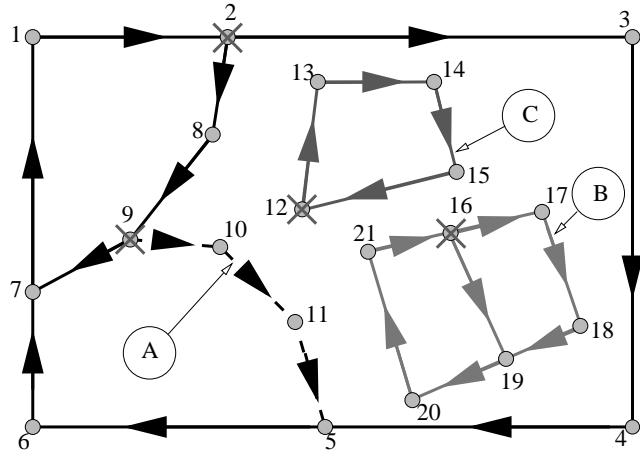


FIGURE A.1 – *Nombre minimum de sous-graphes Eulériens nécessaires pour décrire la grille. Pour parcourir une et une seule fois chaque segment, on peut parcourir les nœuds suivants : (2,3,4,5,6,7,1,2,8,9,7), puis (9,10,11,5), puis (16,17,18,19,20,21,16,19) et enfin (12,13,14,15,12). Le nombre de sous-graphes Eulériens, et donc de points de départs (marqués d'une croix) est égal à 4. Ce nombre inclus  $n_{IMP}/2 = 3$  nœuds de multiplicité impaire et un nœud de multiplicité paire (12) qui appartient au seul graphe isolé ne comportant aucun nœud de multiplicité impaire (C).*

préalable choisir la distribution associée aux vecteurs du graphe. Soient  $d_x(i)$  et  $d_y(i)$  l'abscisse et l'ordonnée du vecteur numéro  $i$  :  $d_x(i) = x_2(i) - x_1(i)$  et  $d_y(i) = y_2(i) - y_1(i)$  où  $(x_1(i), y_1(i))$  et  $(x_2(i), y_2(i))$  sont les coordonnées des extrémités du vecteur numéro  $i$ .

Supposons que  $d_x(i)$  (respectivement  $d_y(i)$ ) est distribué avec une ddp  $P_{m_x}(d)$  (respectivement  $P_{m_y}(d)$ ) de paramètre  $m_x$  (respectivement  $m_y$ ). Connaissant le paramètre  $m_u$  de la ddp  $P_{m_u}(d_u)$  (où  $u = x$  ou  $y$ ), la longueur de code moyenne nécessaire pour coder les composantes verticales et horizontales des  $p$  vecteurs du graphe peut être approchée par l'opposée de la log-vraisemblance  $\mathcal{L}_e[\chi_u|m_u]$  où  $\chi_u = (d_u(i))_{i \in [1, p]}$ . Nous obtenons donc le terme de codage de la grille suivant :

$$\Delta_G = n(\log N + \log p) - \mathcal{L}_e[\chi_x|m_x] - \mathcal{L}_e[\chi_y|m_y] + \Delta(m_x) + \Delta(m_y) \quad (\text{A.2})$$

où  $\chi_x = (d_x(i))_{i \in [1, p]}$  et  $\chi_y = (d_y(i))_{i \in [1, p]}$  et où  $\Delta(m_x)$  et  $\Delta(m_y)$  sont les nombres de nats nécessaires pour coder les paramètres  $m_x$  et  $m_y$ .

Ces paramètres  $m_x$  et  $m_y$  étant inconnus, nous les remplaçons par leurs estimés  $\hat{m}_x$  et  $\hat{m}_y$  au sens du MV. D'après [79], comme  $m_x$  et  $m_y$  sont estimés sur un échantillon de taille  $p$ , chacun peut être codé avec  $\log \sqrt{p}$  nats. On obtient ainsi :

$$\Delta_G = n(\log N + \log p) - \mathcal{L}_e[\chi_x | \hat{m}_x] - \mathcal{L}_e[\chi_y | \hat{m}_y] + \log p \quad (\text{A.3})$$

Il reste maintenant à choisir la loi de distribution  $P_{mu}(d_u)$  (où  $u = x$  ou  $y$ ) pour les composantes horizontales et verticales des vecteurs du graphe. Nous nous proposons d'appliquer le principe du maximum d'entropie, de façon à nous placer dans le cas le plus désavantageux (c'est-à-dire nécessitant le plus de nats). De plus, nous supposerons que la longueur moyenne des projections horizontales et verticales des vecteurs est connue, c'est-à-dire la moyenne  $m_r$ .

et  $m_y$  de  $|d_x|$  et  $|d_y|$ . En d'autres termes, nous devons déterminer la ddp  $P_m(x)$  minimisant l'entropie  $S = - \int_{-\infty}^{+\infty} P_m(x) \log P_m(x) dx$  sous la contrainte que la moyenne de la valeur absolue soit égale à  $m$  :  $\int_{-\infty}^{+\infty} |x| P_m(x) dx = m$ . La ddp correspondante étant une loi de Laplace, nous obtenons donc

$$P_{m_u}(d_u) = \frac{1}{2m_u} e^{\frac{-|d_u|}{m_u}} \quad (\text{A.4})$$

et le terme de codage de la grille s'écrit alors :

$$\Delta_G = n (\log N + \log p) + \log p + p (2 + \log (2\hat{m}_x) + \log (2\hat{m}_y)) \quad (\text{A.5})$$

où  $\hat{m}_x = \frac{1}{p} \sum_{i=1}^p |d_x(i)|$  et  $\hat{m}_y = \frac{1}{p} \sum_{i=1}^p |d_y(i)|$ .

## A.2 Détermination de la log-vraisemblance généralisée d'une loi gamma d'ordre $L$

A titre d'exemple, nous allons détailler ici le calcul de la log-vraisemblance généralisée dans le cas d'une loi Gamma d'ordre  $L$  connu. La ddp de cette loi est la suivante :

$$P_{\theta_r}(x) = \frac{L^L}{\theta_r^L \Gamma(L)} x^{L-1} e^{-\frac{L}{\theta_r}x} \quad (\text{A.6})$$

où  $\theta_r$  est le paramètre scalaire de la loi Gamma dans la région  $\Omega_r$ , c'est-à-dire l'espérance mathématique de  $x$ .

La log-vraisemblance dans la région  $\Omega_r$  s'écrit alors :

$$\begin{aligned} \mathcal{L}_e[\Omega_r | \theta_r] &= \sum_{(x,y) \in \Omega_r} \log [P_{\theta_r}(s(x,y))] \\ &= N_r [L \log L - L \log \theta_r - \log \Gamma(L)] - \frac{L}{\theta_r} \sum_{(x,y) \in \Omega_r} s(x,y) \\ &\quad + (L-1) \sum_{(x,y) \in \Omega_r} \log [s(x,y)] \end{aligned} \quad (\text{A.7})$$

Remarquons que dans cette dernière équation,  $\mathcal{L}_e[\Omega_r | \theta_r]$  dépend du vecteur de paramètres  $\theta_r$  de la ddp. Dans la plupart des applications,  $\theta_r$  est inconnu et doit donc être considéré comme un paramètre de nuisance. Nous allons donc l'estimer. Comme nous l'avons dit dans la partie 2.1.2, nous choisirons de l'estimer au sens du MV.

Dans le cas étudié, à savoir une loi Gamma d'ordre  $L$ , l'estime  $\hat{\theta}_r$  du paramètre  $\theta_r$  au sens du MV n'est autre que la moyenne empirique calculée sur la région  $\Omega_r$  :

$$\hat{\theta}_r = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} s(x,y) \quad (\text{A.8})$$

En remplaçant  $\theta_r$  par  $\hat{\theta}_r$  dans (A.7), nous obtenons alors la log-vraisemblance généralisé suivante :

$$\mathcal{L}_e[\Omega_r | \hat{\theta}_r] = N_r L \log L - N_r L \log \hat{\theta}_r - N_r \log \Gamma(L) - N_r L + (L-1) \sum_{(x,y) \in \Omega_r} \log [s(x,y)] \quad (\text{A.9})$$

En sommant sur les  $R$  régions, on obtient alors :

$$\begin{aligned} \mathcal{L}_e[\mathbf{s} | \mathbf{w}, \hat{\boldsymbol{\theta}}] &= \sum_{r=1}^R \mathcal{L}_e[\Omega_r | \hat{\theta}_r] \\ &= N L \log L - L \sum_{r=1}^R N_r \log \hat{\theta}_r - N \log \Gamma(L) - N L \\ &\quad + (L-1) \sum_{(x,y) \in \text{Image}} \log [s(x,y)] \end{aligned} \quad (\text{A.10})$$

où nous avons noté  $\mathcal{L}_e[\mathbf{s} | \mathbf{w}, \hat{\boldsymbol{\theta}}]$  la log-vraisemblance généralisée calculée sur toute l'image, c'est-à-dire sur les  $R$  régions, et où  $\hat{\boldsymbol{\theta}} = (\hat{\theta}_r)_{r \in \{1, 2, \dots, R\}}$ . Dans cette équation, nous pouvons constater que seul le terme  $-L \sum_{r=1}^R N_r \log \hat{\theta}_r$  varie avec la partition  $w$ . Nous pouvons donc écrire :

$$\mathcal{L}_e[\mathbf{s} | \mathbf{w}, \hat{\boldsymbol{\theta}}] = -L \sum_{r=1}^R N_r \log \hat{\theta}_r - K(\mathbf{s}, L) \quad (\text{A.11})$$

où  $K(\mathbf{s}, L)$  est un terme additif indépendant de la partition  $w$ , mais qui dépend bien sûr de l'image et de l'ordre  $L$  de la loi Gamma.

## A.3 Algorithme de minimisation de la complexité stochastique

Dans cette partie, nous allons décrire plus en détail l'algorithme de minimisation de la complexité stochastique présenté en partie 2.2.

### A.3.1 Fusion de régions

L'algorithme de fusion MCS (voir partie 2.2.1) consiste à effectuer les fusions permettant une diminution de la complexité stochastique. Dans le cas de la fusion GLRT, seules les fusions qui n'entraînent pas d'augmentation de la log-vraisemblance supérieure à un seuil  $\tilde{S}$  sont acceptées. Il est maintenant nécessaire de définir l'ordre dans lequel nous allons effectuer ces différentes fusions.

Le plus simple à mettre en œuvre consisterait à choisir l'ordre des fusions aléatoirement. Cependant, nous avons mis en œuvre une méthode qui permet d'obtenir de meilleurs résultats. Nous avons vu dans l'équation 2.19, que dans le cas de fusion MCS aussi bien que GLRT, deux régions  $\Omega_A$  et  $\Omega_B$  pouvaient être fusionnées si

$$\mathcal{L}_e \left[ \Omega_A | \hat{\theta}_A \right] + \mathcal{L}_e \left[ \Omega_B | \hat{\theta}_B \right] - \mathcal{L}_e \left[ \Omega_{AB} | \hat{\theta}_{AB} \right] < S \quad (\text{A.12})$$

où nous avons noté  $\hat{\theta}_{AB}$  l'estimée (au sens du MV) du vecteur de paramètres de la région  $\Omega_{AB} = \Omega_A \cup \Omega_B$ . Dans le cas d'une fusion MCS, nous devons prendre  $S = S(w, w')$  alors que dans le cas GLRT, nous avons  $S = \tilde{S}$  (voir partie 2.2.1).

Nous avons donc choisi de classer ces différentes fusions possibles en fonction de la variation de log-vraisemblance  $\mathcal{L}_e \left[ \Omega_A | \hat{\theta}_A \right] + \mathcal{L}_e \left[ \Omega_B | \hat{\theta}_B \right] - \mathcal{L}_e \left[ \Omega_{AB} | \hat{\theta}_{AB} \right]$  qu'elles engendrent : nous commencerons donc par examiner les régions dont la fusion n'entraîne qu'une faible diminution de la log-vraisemblance<sup>26</sup> (c'est-à-dire une faible augmentation du terme  $\Delta_L$ ).

Ce tri sur la variation de log-vraisemblance offre l'avantage de pouvoir être mis rapidement à jour après chaque fusion de régions, dans la mesure où seuls quelques calculs, impliquant les régions autour des deux régions qui viennent d'être fusionnées, doivent être effectués. Ce ne serait pas le cas si nous avions choisi un tri reposant sur la complexité stochastique. En effet, après chaque fusion de régions, différents paramètres de la grille varient, notamment le nombre de segments  $p$  : il serait donc nécessaire après chaque fusion de recalculer toutes les variations de complexité stochastique afin de déterminer la fusion conduisant à sa plus forte diminution.

### A.3.2 Déplacement des nœuds

Comme nous l'avons évoqué en partie 2.2.2, il n'est pas possible lors de la phase de déplacement des nœuds d'examiner tous les déplacements possibles pour chaque nœud en un temps raisonnable. C'est pourquoi nous avons choisi de n'examiner qu'un nombre restreint de déplacements.

26. En effet, lors de la fusion de deux régions, la log-vraisemblance ne peut que diminuer (au mieux, elle ne varie pas).

L'algorithme de déplacement des nœuds consiste alors à examiner les nœuds un par un, et à considérer pour chacun d'entre eux 8 déplacements particuliers d'amplitude  $a$  fixée comme le présente la fig. 2.3 de la partie 2.2.4. L'objectif étant de minimiser la complexité stochastique, pour un nœud donné, c'est la position correspondant à la plus petite valeur de la complexité stochastique qui est sélectionnée, parmi les 9 positions possibles (la position initiale plus les 8 déplacements testés).

Le critère d'arrêt est là encore très simple : l'algorithme de déplacement des nœuds est stoppé lorsqu'après avoir parcouru tous les nœuds, aucun des déplacements testés n'a entraîné de diminution de la complexité stochastique.

Cet algorithme requiert le réglage d'un paramètre, l'amplitude de déplacement  $a$ . Il est cependant possible de définir cette amplitude afin que l'utilisateur n'ait pas à la régler. Pour chaque nœud  $P_i$  de la grille, on impose une amplitude  $a_i$  différente. Cette amplitude est choisie égale à  $E^+(m_i/2)$  où  $m_i$  est la longueur moyenne des segments reliés au nœud  $P_i$  considéré, et où nous avons noté  $E^+$  l'arrondi à l'entier supérieur<sup>27</sup>. L'utilisation d'une amplitude dépendant du nœud considéré permet de gérer une grille où les longueurs des segments peuvent être très variables selon la partie de l'image considérée.

Soulignons maintenant un deuxième point important. Utiliser une assez grande amplitude au début de la phase de déplacement permet une convergence rapide vers la bonne position du contour et évite de plus de rester piégé dans un minimum local de la complexité stochastique. Il est en revanche nécessaire de diminuer cette amplitude en fin de convergence afin d'estimer la position du contour avec une meilleure précision, l'idéal étant de terminer avec une amplitude d'un pixel.

Nous avons donc mis en place une stratégie multi-résolution qui consiste à considérer les amplitudes  $a_i$  précédemment définies dans un premier temps, puis, dès que plus aucun nœud ne peut être déplacé, une nouvelle étape de déplacement des nœuds est mise en place, mais cette fois-ci avec toutes les amplitudes  $a_i$  divisées par 2. Cette procédure de division des amplitudes de déplacement est ainsi poursuivie jusqu'à ce que toutes les amplitudes soient égales à 1 pixel. Pour plus de clarté, cet algorithme de déplacement des nœuds est présenté dans le tableau A.1.

### A.3.3 Suppression des nœuds

Nous proposons d'analyser l'ordre dans lesquelles vont être effectuées les suppressions de nœuds.

Nous avons remarqué qu'afin de garantir une meilleure convergence, il est intéressant de supprimer à chaque étape le nœud engendrant la plus forte diminution de la complexité stochastique. De plus, la suppression d'un nœud entraînant une modification locale du contour, il est nécessaire de déplacer les nœuds qui étaient reliés au nœud qui vient d'être supprimé. Pour cela, l'algorithme de déplacement des nœuds présenté dans la partie précédente est utilisé.

Cet algorithme nécessite cependant un temps de calcul important. En effet, le tri effectué pour savoir quelle suppression de nœud doit être examinée, nécessite le calcul de la variation de la complexité stochastique associée à chaque nœud. Or ces variations de complexité stochastique dépendent de termes qui varient au cours des suppressions, comme par exemple le nombre de segments  $p$ . Après chaque suppression de nœud, il est donc nécessaire de recalculer les variations de complexité stochastique associées à chaque nœud.

---

27. En effet, dans la mesure où  $a_i$  doit être un entier non nul, nous arrondissons sa valeur à l'entier supérieur.

- (1) Pour chaque nœud  $P_i$ ,  $a_i = E^+(m_i/2)$
  - (2) Parcourir un par un chaque nœud  $P_i$  :
    - Déplacer le nœud  $P_i$  avec une amplitude  $a_i$  dans 8 directions
    - Choisir la meilleure position du nœud tant qu'au moins un nœud a été déplacé.
  - (3) S'il existe des nœuds associés à une amplitude  $a_i > 1$  :
    - Pour ces nœuds  $P_i$ ,  $a_i \leftarrow E^+(a_i/2)$
    - Retourner en (2)
 Sinon, la convergence est terminée.
- 

TABLE A.1 – *Algorithme de déplacement des nœuds.* Nous avons noté  $E^+$  l'arrondi à l'entier supérieur et  $m_i$  la longueur moyenne des segments reliés au nœud  $P_i$ .

Cette phase de suppression de nœuds est donc très lente, sa complexité algorithmique étant approximativement en  $O(k^2)$  où  $k$  est le nombre de nœuds sur la grille. Il est donc nécessaire d'utiliser une autre approche, notamment lorsque le nombre de nœuds est très élevé.

Pour résoudre ce problème, nous avons choisi de ne plus sélectionner le nœud dont la suppression engendrerait la plus forte diminution de la complexité stochastique. Nous classons dès le début tous les nœuds doubles<sup>28</sup> en partant de celui entraînant la plus forte diminution de complexité stochastique en cas de suppression jusqu'à celui entraînant la plus faible diminution (qui est souvent d'ailleurs une augmentation). Chaque nœud est ensuite examiné dans cet ordre, et l'algorithme de suppression des nœuds s'arrête lorsque tous les nœuds ont été examinés.

Nous considérons donc deux algorithmes différents : l'un, plus rigoureux où à chaque étape le test de suppression s'effectue sur le nœud qui conduit à la plus grande diminution de la complexité stochastique, alors que l'autre, beaucoup plus rapide se contente de déterminer l'ordre des tests à effectuer une fois pour toute. Nous utiliserons ainsi l'algorithme de suppression rapide au début de la procédure d'optimisation, quand la grille comporte un grand nombre de nœuds, alors que la dernière phase de suppression de nœuds sera effectuée avec l'algorithme plus rigoureux.

### A.3.4 Temps de calcul

Nous allons maintenant aborder la question des temps de calcul. Tous les temps de calcul fournis dans ce manuscrit ont été obtenus sur un PC standard sous Linux 2.4 (gcc 2.96) équipé d'un processeur Xéon cadencé à 2.8 GHz.

Soulignons tout d'abord un point important. Nous avons vu dans les sections précédentes qu'un certain nombre de choix ont été faits concernant les algorithmes de fusion, de déplacement et de suppression. A titre d'exemple, le seuil de fusion GLRT est systématiquement fixé à la valeur  $\tilde{S} = 3$ , l'amplitude des déplacements est diminuée progressivement jusqu'à la valeur de 1 pixel, chaque suppression de nœud est suivie de déplacements de ses nœuds

28. En effet, seuls les nœuds doubles peuvent être supprimés lors de la phase de suppression de nœuds, puisqu'un nœud double est un nœud relié à seulement deux autres nœuds.

voisins, etc. Tous ces choix sont motivés à la fois par la volonté d'avoir le moins de paramètres possibles à régler dans l'algorithme de minimisation, tout en souhaitant obtenir un algorithme aussi général que possible, c'est-à-dire un algorithme de minimisation conduisant à de bons résultats quelle que soit l'image à partitionner.

Cependant, dans le cas d'applications particulières, il est possible d'optimiser cette stratégie ou de réaliser quelques simplifications de l'algorithme, ce qui peut avoir une grande influence sur le temps de calcul. Citons à titre d'exemple quelques modifications possibles dans l'algorithme permettant d'obtenir des gains de temps de calcul :

- Ne plus faire décroître l'amplitude de déplacement des nœuds au cours d'une même phase de déplacement mais la faire diminuer seulement phase après phase. Ainsi, à la première phase de déplacement des nœuds, on fixera l'amplitude de déplacement du nœud  $P_i$  à la valeur  $a_i = E^+(m_i/2)$ , puis à la phase de déplacement suivante (après une phase de fusion puis de suppression des nœuds) on prendra  $a_i = E^+(m_i/4)$ .
- Ne plus utiliser de déplacement de nœuds entre chaque suppression de nœuds.

Toutes ces simplifications ont ainsi pour but d'améliorer le temps de calcul de l'algorithme de minimisation pour un type d'image bien particulier. Les temps de calcul fournis dans cette partie, comme dans tout ce manuscrit, ont souvent été obtenus en utilisant de telles simplifications.

A titre d'exemple, nous présentons sur la fig. A.2 les temps de calcul obtenus lors de la partition en régions homogènes de trois images SAR réelles prises lors du naufrage du Prestige au large des côtes espagnoles (pour plus de précision sur ces images et leur segmentation, se référer à la partie 3.3). Ces temps de calcul, fournis pour plusieurs tailles de grilles initiales (de type mur de briques), ont été obtenus en utilisant rigoureusement le même algorithme d'optimisation pour chaque image.

Nous présentons ensuite sur la fig. A.3 différents temps de calcul en fonction du nombre de nœuds de la grille initiale. Ces temps de calcul ont été obtenus sur diverses images de tailles différentes et avec diverses grilles initiales, mais en utilisant l'algorithme non optimisé en temps de calcul. Nous pouvons constater sur ce graphique que pour une image donnée, le temps de calcul varie quasi-linéairement en fonction du nombre de nœuds présents dans la grille initiale. De plus, nous remarquons que la pente de cette droite dépend de l'image considérée : ceci était prévisible, dans la mesure où le nombre de régions et leur complexité sont différents pour chaque image.

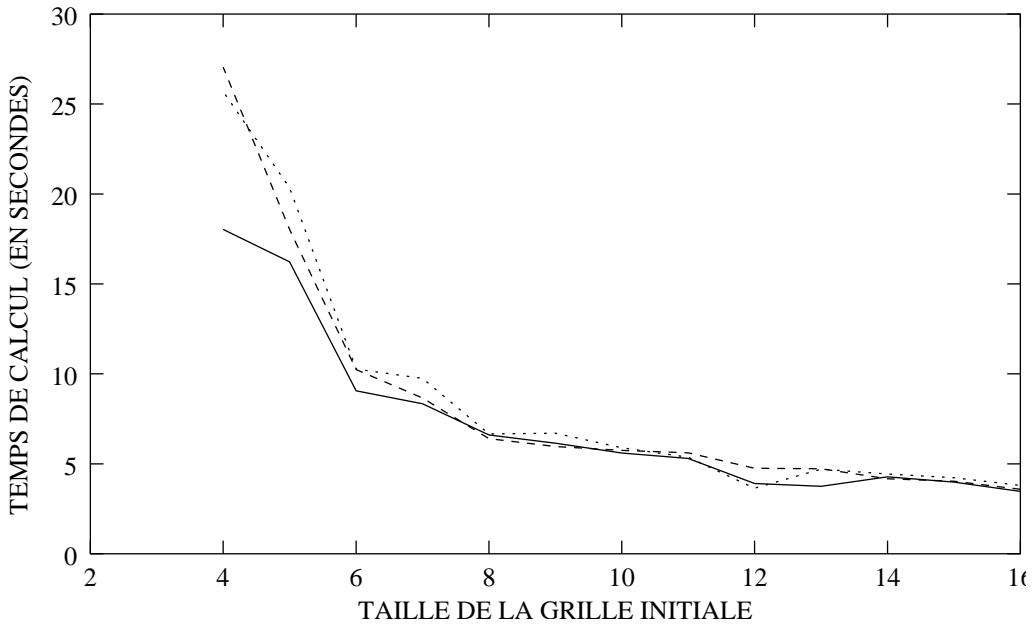


FIGURE A.2 – Temps de calcul obtenus lors de la partition en régions homogènes des 3 images SAR de la marée noire causée par le Prestige (voir partie 3.3, fig 3.11.a, 3.13.a et 3.15.a), en fonction de la taille (en pixels) de la grille initiale (de type mur de briques). Ces images ont des tailles de  $801 \times 581$ ,  $875 \times 598$  et  $878 \times 601$  pixels.

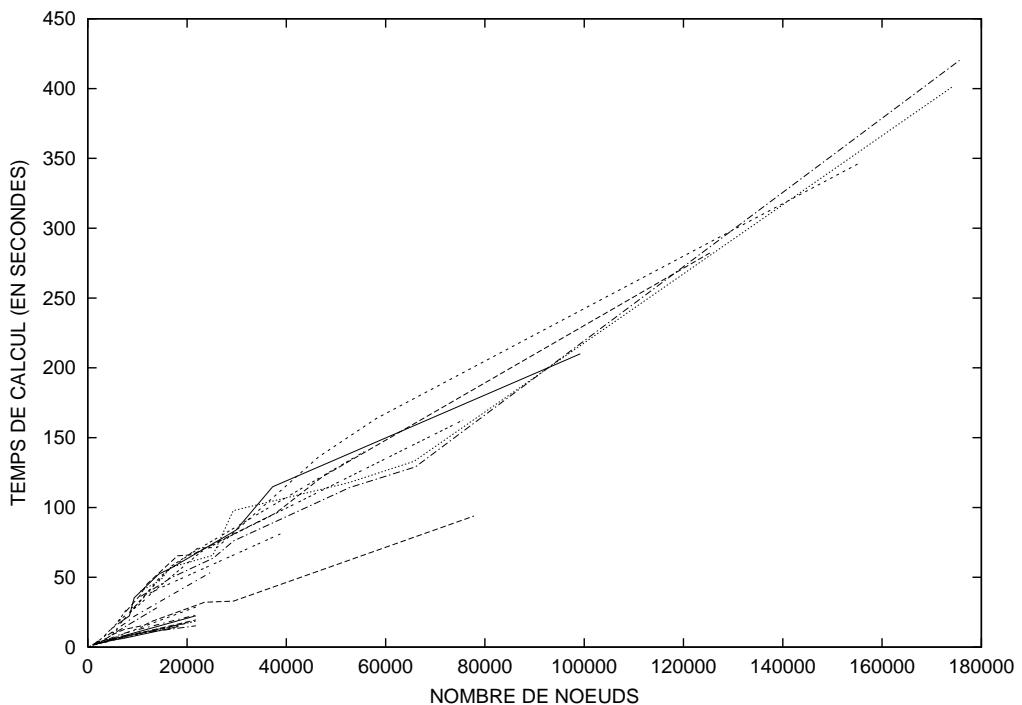


FIGURE A.3 – Temps de calcul obtenus lors de la partition en régions homogènes de différentes images, en fonction du nombre de nœuds dans la grille initiale. Chaque courbe correspond à l'ensemble des temps de calcul obtenus pour une même image mais avec des grilles initiales différentes.

## A.4 Définition non ambiguë de l'intérieur des régions par translation de la grille

Dans cette partie, nous allons détailler la façon dont est déterminé l'intérieur de chaque région à partir de la partition  $w$ .

Nous allons dans un premier temps expliquer comment la partition se projette sur le réseau discret lorsque l'on connaît la position de chacun des noeuds de la grille. Pour cela, il est utile de préciser quelques notations sur les entités géométriques que nous manipulons.

- On se place dans le plan continu  $\mathbb{R}^2$  et on considère le réseau discret  $\mathbb{Z}^2$  auquel appartiennent les noeuds d'un polygone.
- A partir des noeuds du polygone (fig. A.4.a), il est possible de déterminer *le contour polygonal discret*  $\mathcal{C}$  (fig. A.4.b). Celui-ci est constitué des segments discrets joignant les noeuds consécutifs, construits par l'algorithme de Bresenham [15]. C'est donc un sous-ensemble de  $\mathbb{Z}^2$ .
- A partir de  $\mathcal{C}$ , on définit le *lacet continu*  $\tilde{\mathcal{C}}$  qui est la courbe de  $\mathbb{R}^2$  joignant les points successifs de  $\mathcal{C}$  (fig. A.4.c).
- Nous noterons  $\text{Int} [\tilde{\mathcal{C}}]$  l'intérieur strict de  $\tilde{\mathcal{C}}$  qui est une surface de  $\mathbb{R}^2$  (fig. A.4.d).

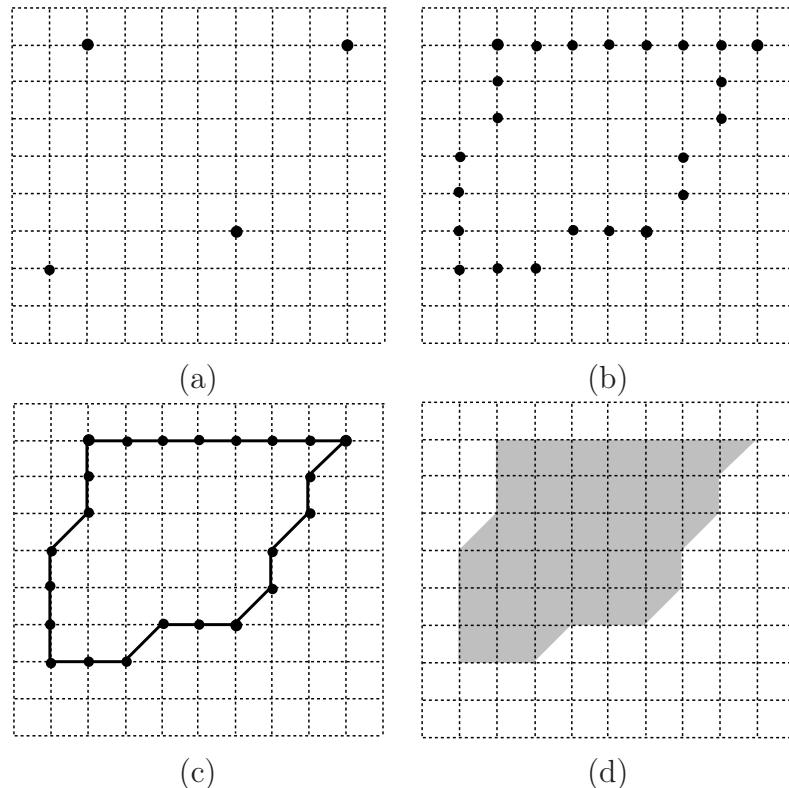


FIGURE A.4 – Notations employées pour définir l'intérieur d'un contour polygonal discret. (a) Nœuds du polygone. (b) Contour polygonal discret  $\mathcal{C}$ . (c) Lacet continu correspondant  $\tilde{\mathcal{C}}$ . (d) Intérieur strict du lacet continu  $\text{Int} [\tilde{\mathcal{C}}]$ .

Commençons par rappeler la convention la plus simple que l'on pourrait être tenté d'utiliser pour définir la partition en deux domaines (intérieur et extérieur du contour). En utilisant les notations précédentes, l'ensemble des pixels intérieurs au contour polygonal  $\mathcal{C}$  s'écrirait alors

formellement :

$$\Omega = \left( \mathbb{Z}^2 \cap \text{Int} \left[ \tilde{\mathcal{C}} \right] \right) \cup \mathcal{C} \quad (\text{A.13})$$

Autrement dit, l'intérieur du contour polygonal discret  $\mathcal{C}$  comprend les pixels strictement à l'intérieur de  $\tilde{\mathcal{C}}$  auxquels s'ajoutent les pixels de  $\mathcal{C}$ . Soulignons d'ailleurs que c'est la convention qui avait été retenue dans [16] dans le cadre de la segmentation par contour actif polygonal.

Il est clair que cette convention n'est pas applicable dans le cas de la grille. Si c'était le cas, les pixels situés sur le segment discret séparant deux régions appartiendraient simultanément à l'une et à l'autre de ces deux régions et on ne pourrait pas définir une partition de l'image avec la grille. Une nouvelle convention a donc été choisie [38] pour lever l'ambiguïté sur les pixels des segments discrets. Il est simple de montrer<sup>29</sup> que le lacet  $\tilde{\mathcal{C}}$  translaté d'une quantité  $(1/2, 1/4)$  ne possède aucune intersection avec le réseau discret  $\mathbb{Z}^2$ . On peut donc définir  $\Omega$  sans ambiguïté comme l'ensemble des pixels strictement à l'intérieur du lacet translaté. Formellement, cette définition s'écrit :

$$\Omega = \mathbb{Z}^2 \cap \text{Int} \left[ \mathcal{T}_{(1/2, 1/4)} \left( \tilde{\mathcal{C}} \right) \right] \quad (\text{A.14})$$

où  $\mathcal{T}_{(1/2, 1/4)}$  représente l'opérateur de translation selon  $(1/2, 1/4)$ . Cette convention revient en fait à considérer que les contours discrets ne sont plus localisés sur le réseau direct  $\mathbb{Z}^2$  mais sur un réseau dual de  $\mathbb{Z}^2$ . Une représentation de ce réseau dual est obtenue en translatant le réseau direct de  $(1/2, 1/4)$ .

Avec cette convention, les pixels d'un contour discret polygonal  $\mathcal{C}$  n'appartiennent pas tous à l'intérieur  $\Omega$  (fig. A.5).

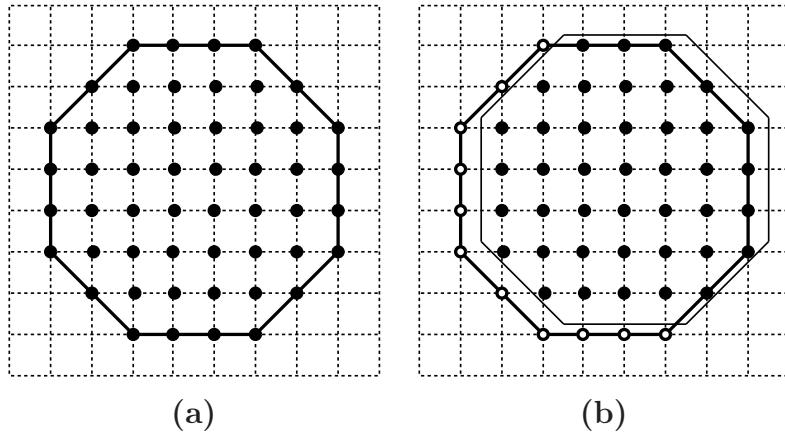


FIGURE A.5 – Comparaison des deux conventions pour la définition de l'intérieur d'un contour polygonal discret (les pixels à l'intérieur sont représentés en noir). (a) Convention la plus simple (équation A.13). (b) Convention "grille active" (équation A.14).

Pour finir, la fig. A.6 donne un exemple de partition du réseau discret à partir de la grille.

29. Le lacet continu est une succession de segments continus entre des points voisins du réseau discret. Il suffit donc de montrer la proposition sur un segment continu quelconque entre deux points voisins du réseau discret. Soient  $M(x, y)$  et  $M'(x', y')$  deux points voisins sur le réseau discret  $\mathbb{Z}^2$ . Le segment continu  $[MM']$  après translation a pour équation : 
$$\begin{cases} (x' - x)\zeta + x + 1/2 \\ (y' - y)\zeta + y + 1/4 \end{cases} \quad \text{avec } \zeta \in [0, 1].$$
 Comme  $M$  et  $M'$  sont voisins,  $(x' - x)$  et  $(y' - y)$  ne peuvent que prendre les valeurs 0, 1 ou -1. Il est donc facile de se persuader que les deux coordonnées ne peuvent pas être entières simultanément.

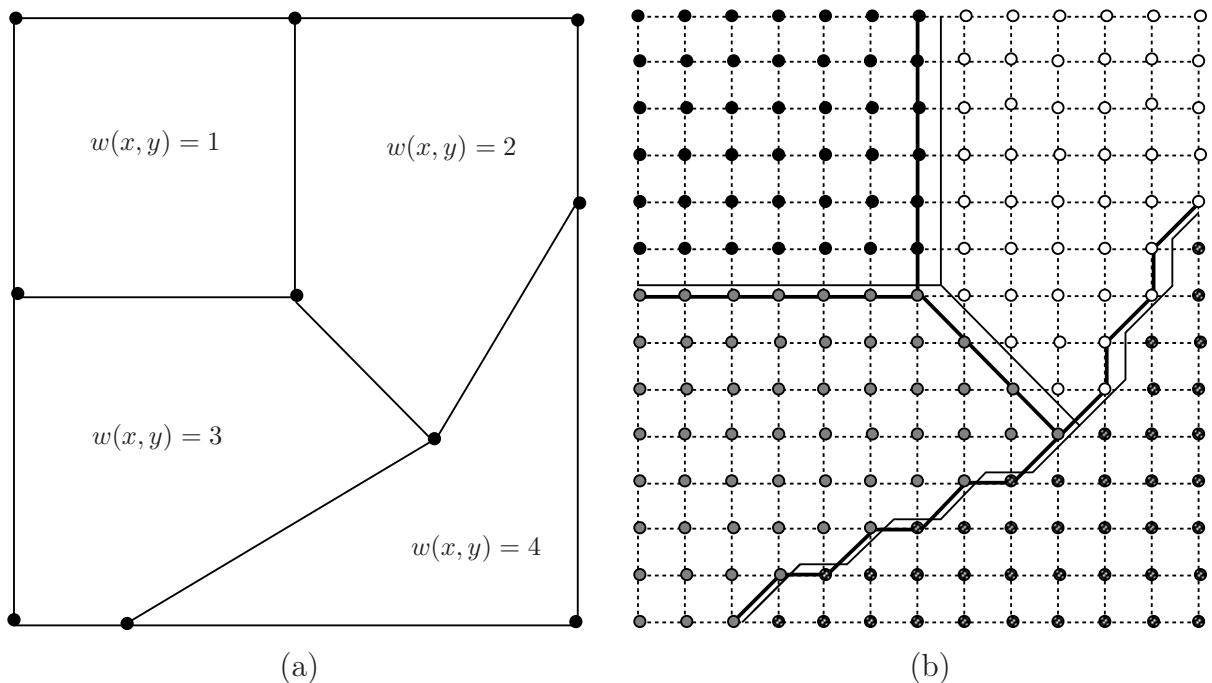


FIGURE A.6 – Projection de la partition sur le réseau discret  $\mathbb{Z}^2$ . (a) Grille polygonale comprenant 10 nœuds et 4 régions. (b) Partition correspondante. On a représenté les lacets continus correspondants aux contours discrets polygonaux (en traits épais) ainsi que leurs translations selon  $(1/2, 1/4)$  (traits fins). En appliquant la convention (A.14), on définit sans ambiguïté les quatre régions (pixels en noir, blanc, gris clair et gris sombre).

## A.5 Algorithme rapide : présentation, généralisation et mise en œuvre

Dans un premier temps (partie A.5.1), nous allons rappeler le principe de l'algorithme rapide tel qu'il a été présenté dans [38] dans le cas de régions simplement connexes. Nous allons ensuite montrer qu'une généralisation aux régions non simplement connexes est possible (partie A.5.2) avant de présenter sa mise en œuvre du point de vue informatique (parties A.5.3 et A.5.4).

### A.5.1 Rappels

Soit  $\Omega$  une région simplement connexe. Nous avons vu (équation 2.20) qu'une sommation surfacique sur  $\Omega$  était nécessaire pour calculer la statistique  $T_k(\Omega)$  :

$$T_k(\Omega) = \sum_{(x,y) \in \Omega} t_k[s(x,y)] \quad (\text{A.15})$$

Rappelons, à titre d'exemple, que dans le cas d'une ddp gamma, nous avons :

$$t_k[s(x,y)] = [s(x,y)]^k, \quad k \in \{0, 1\} \quad (\text{A.16})$$

La région étant simplement connexe, cette sommation 2D peut être remplacée par une sommation 1D sur le contour de  $\Omega$  :

$$T_k(\Omega) = \sum_{(x,y) \in \Omega} t_k[s(x,y)] = \sum_{(x,y) \in \mathcal{C}} c(x,y) F_k(x,y) \quad (\text{A.17})$$

où  $\mathcal{C}$  est le contour de  $\Omega$  parcouru dans le sens trigonométrique et où  $F_k$  est une image pré-calculée une fois pour toute à l'initialisation de l'algorithme et obtenue en effectuant une sommation selon les lignes de l'image que l'on souhaite partitionner :

$$F_k(x,y) = \sum_{x_1=1}^x t_k[s(x_1,y)] \quad (\text{A.18})$$

Le coefficient  $c(x,y)$  en un point  $(x,y)$  du contour dépend seulement des positions de ses deux pixels voisins sur le contour  $\mathcal{C}$ , et prend ses valeurs parmi l'ensemble  $\{-1, 0, 1\}$ . Avec les notations de la fig. A.7, les valeurs de  $c(x,y)$  sont données dans le tableau A.2.

Pour plus de détails sur la façon dont sont obtenues les valeurs de  $c(x,y)$ , on pourra se référer à [38].

### A.5.2 Généralisation aux régions non simplement connexes

Nous allons maintenant généraliser cette technique au cas d'une région  $\Omega$  quelconque, c'est-à-dire non nécessairement simplement connexe. Cela signifie qu'une telle région peut maintenant comporter des trous ou être séparée en plusieurs parties disjointes. Comme nous pouvons le voir sur la fig. A.8, une région non simplement connexe peut être décomposée en 2 familles de régions simplement connexes : les régions pleines, notées  $\Omega_i^P$  et les régions trous, notées  $\Omega_i^T$ .

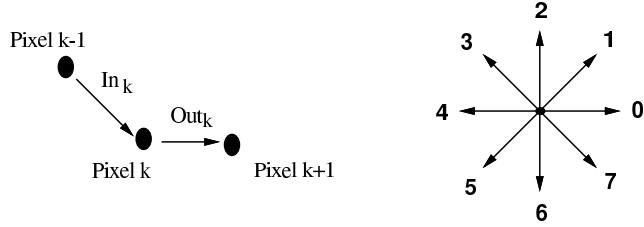


FIGURE A.7 – Notations utilisées pour décrire la configuration de 3 pixels consécutifs du contour. En parcourant le contour dans le sens trigonométrique,  $In_k$  et  $Out_k$  sont les vecteurs définis à partir du pixel  $k$  et de ses 2 pixels voisins : le précédent ( $k - 1$ ) et le suivant ( $k + 1$ ). Les directions de ces 2 vecteurs sont données à partir du code de Freeman (à droite).

$In_k \setminus Out_k$	0	1	2	3	4	5	6	7
0	0	0	0	0	0	-1	-1	-1
1	1	1	1	1	1	0	0	0
2	1	1	1	1	1	0	0	0
3	1	1	1	1	1	0	0	0
4	0	0	0	0	0	-1	-1	-1
5	0	0	0	0	0	-1	-1	-1
6	0	0	0	0	0	-1	-1	-1
7	0	0	0	0	0	-1	-1	-1

TABLE A.2 – Valeur  $c(x, y)$  du pixel  $(x, y)$  situé sur le contour, lorsque l'on se déplace le long du contour discret dans le sens trigonométrique, connaissant la position du pixel précédent et du pixel suivant.

On peut alors écrire :

$$T_k(\Omega) = \sum_{(x,y) \in \Omega} t_k[s(x,y)] = \sum_i \left( \sum_{(x,y) \in \Omega_i^P} t_k[s(x,y)] \right) - \sum_j \left( \sum_{(x,y) \in \Omega_j^T} t_k[s(x,y)] \right) \quad (\text{A.19})$$

Comme les régions  $(\Omega_i^P)_i$  et  $(\Omega_j^T)_j$  sont simplement connexes, il est possible d'utiliser (A.17) et de remplacer les sommes sur les surfaces par des sommes sur les contours :

$$T_k(\Omega) = \sum_i \left( \sum_{C_i^P} c(x,y) F_k(x,y) \right) - \sum_j \left( \sum_{C_j^T} c(x,y) F_k(x,y) \right) \quad (\text{A.20})$$

où  $C_i^P$  et  $C_j^T$  sont les contours de  $\Omega_i^P$  et  $\Omega_j^T$  parcourus dans le sens trigonométrique. D'après cette équation, la contribution d'une région pleine doit être considérée positive alors que la contribution d'une région trou doit être considérée négative.

Nous avons précisé dans la sous-partie précédente que pour calculer  $T_k(\Omega)$  selon (A.17) lorsque  $\Omega$  est simplement connexe, les contours devaient être parcourus dans le sens trigonométrique. Cependant, d'après le tableau de codage de  $c(x,y)$  (tableau A.2), il est possible de montrer que si l'on parcourt ce contour dans le sens trigonométrique inverse, nous obtenons

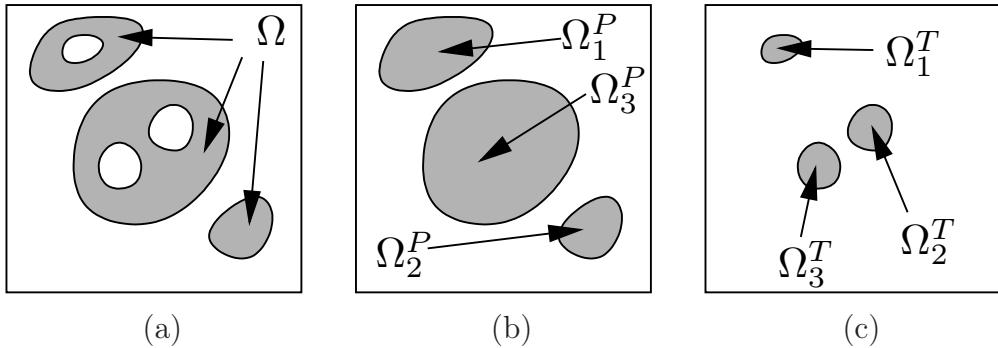


FIGURE A.8 – Décomposition de la région non simplement connexe  $\Omega$  (en gris) (a) en 2 familles de régions simplement connexes : les régions pleines  $\Omega_i^P$  (b) et les régions trous  $\Omega_i^T$  (c).

la valeur opposée :

$$\sum_{\mathcal{C}\{+\}} c(x, y) F_k(x, y) = - \sum_{\mathcal{C}\{-\}} c(x, y) F_k(x, y) \quad (\text{A.21})$$

où  $\mathcal{C}\{+\}$  (respectivement  $\mathcal{C}\{-\}$ ) est le contour de  $\Omega$  parcouru dans le sens trigonométrique direct (respectivement inverse)<sup>30</sup>. L'équation (A.20) peut alors être ré-écrite :

$$T_k(\Omega) = \sum_i \left( \sum_{\mathcal{C}_i^P\{+\}} c(x, y) F_k(x, y) \right) + \sum_j \left( \sum_{\mathcal{C}_j^T\{-\}} c(x, y) F_k(x, y) \right) \quad (\text{A.22})$$

De plus, nous pouvons remarquer que lorsqu'un contour est parcouru en gardant  $\Omega$  sur sa gauche, les régions pleines sont parcourues dans le sens trigonométrique directe alors que les régions trous sont parcourues dans le sens trigonométrique inverse.

Finalement, nous pouvons donc écrire :

$$T_k(\Omega) = \sum_{\mathcal{C}\{\text{gauche}\}} c(x, y) F_k(x, y) \quad (\text{A.23})$$

où  $\mathcal{C}\{\text{gauche}\}$  est l'union de tous les contours de  $\Omega$ , c'est-à-dire  $\mathcal{C} = (\bigcup_i \mathcal{C}_i^P) \cup (\bigcup_j \mathcal{C}_j^T)$ , parcourus en gardant  $\Omega$  sur la gauche.

### A.5.3 Mise en œuvre

Nous pouvons maintenant présenter la façon dont sont calculées les statistiques de chaque région en utilisant cette accélération.

Notons  $[A \rightarrow B]$  le segment  $[AB]$  parcouru du point  $A$  vers le point  $B$ . Pour chaque segment  $[AB]$  de la grille, il est possible de déterminer sa contribution à la statistique  $T_k$  en calculant  $\sum_{[A \rightarrow B]} c(x, y) F_k(x, y)$ . Cette contribution doit être ajoutée à  $T_k(\Omega_G)$  et doit être soustraite à  $T_k(\Omega_D)$  où  $\Omega_G$  est la région sur la gauche lorsque l'on va de  $A$  vers  $B$  et  $\Omega_D$  la région sur

30. Il est important de noter que ceci n'est pas vrai avec le codage  $c(x, y)$  proposé dans [16] et adapté au cas d'un contour actif, dans lequel le contour est défini comme faisant partie de l'objet à segmenter. Ce n'est qu'après avoir translaté la grille de  $(1/2, 1/4)$  afin de proposer une définition d'intérieur et d'extérieur adaptée au cas d'une grille (voir annexe A.4) que l'on obtient le codage  $c(x, y)$  du tableau A.2, qui vérifie alors bien l'équation (A.21).

la droite. Il est ainsi possible de déterminer  $T_k(\Omega_r)$  - et donc le nombre de pixels  $N_r$  et les paramètres  $\hat{\theta}_r$  - pour toutes les régions  $\Omega_r$  en examinant une fois seulement chaque segment<sup>31</sup>.

De plus, une fois que les statistiques  $T_k$  dans chaque région ont été calculées pour une configuration de grille particulière, elles peuvent être mises à jour très rapidement au cours du processus d'optimisation. En effet, lorsqu'un nœud est déplacé ou supprimé<sup>32</sup>, il suffit de modifier la contribution des segments liés à ce nœud pour mettre à jour les statistiques de toutes les régions qui ont été modifiées.

Cette approche permet ainsi de déterminer et de mettre à jour très rapidement les paramètres des ddp des niveaux de gris dans chaque région, même en présence de régions non simplement connexes.

#### A.5.4 Gestion de la topologie

Nous venons de voir que l'on pouvait calculer de façon rapide les statistiques à l'intérieur d'une région  $\Omega$  quelconque, pourvu que l'on parcourt sa frontière en gardant  $\Omega$  sur la gauche. De plus, au cours de l'algorithme de minimisation, il est nécessaire de mémoriser la grille  $w$  de façon à pouvoir accéder rapidement aux informations importantes. Nous allons donc présenter ici la façon dont la grille est mémorisée.

Nous avons choisi de décrire la grille à l'aide de deux structures. La première, très simple, fournit simplement les coordonnées de chacun des nœuds de la grille. Il reste alors à donner la structure topologique de la grille.

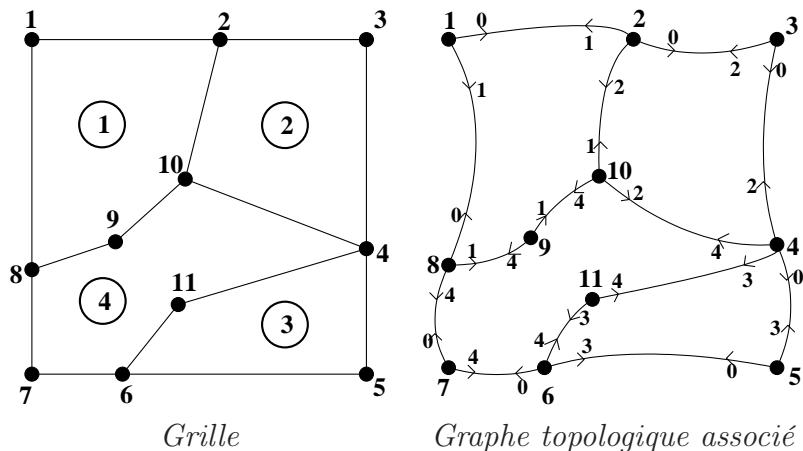


FIGURE A.9 – Représentation de la topologie de la grille sous la forme d'un graphe valué et orienté : exemple d'une grille comportant 4 régions et 11 nœuds.

Cette structure topologique est décrite à l'aide d'un graphe valué et orienté (fig. A.9), appelé dans la suite graphe topologique. Dans la grille, les nœuds et les régions sont repérés par un numéro<sup>33</sup>. Chaque nœud de la grille correspond à un sommet dans le graphe. Lorsque deux

31. Cela signifie que si l'on a calculé la contribution du segment  $[A \rightarrow B]$ , il est inutile de calculer celle du segment  $[B \rightarrow A]$ , puisque l'on sait qu'elles sont opposées l'une de l'autre.

32. La fusion de régions ne pose aucun problème, avec ou sans algorithme rapide, puisque si l'on fusionne deux régions  $\Omega_1$  et  $\Omega_2$ , ses statistiques se déduisent immédiatement de celles de  $\Omega_1$  et  $\Omega_2$  :  $T_k(\Omega_1 \cup \Omega_2) = T_k(\Omega_1) + T_k(\Omega_2)$ .

33. Par convention, la région extérieure à l'image est numérotée 0.

nœuds  $M$  et  $M'$  sont reliés par un segment, ils constituent une frontière entre deux régions : l'une à droite et l'autre à gauche lorsque le segment est parcouru de  $M$  vers  $M'$ . Les deux sommets correspondants du graphe sont alors liés par un arc dont la valuation est le numéro de la région à gauche. Il devient ainsi très facile de parcourir la frontière d'une région  $\Omega$  en la laissant sur la gauche.

Le graphe topologique permet ainsi d'accéder à deux informations nécessaires au cours de l'algorithme d'optimisation. Il fournit en effet la liste des nœuds reliés à un noeud donné, ainsi que les régions situées à gauche des segments connaissant le sens de leur parcours.

## A.6 Cas où la moyenne n'est plus constante pas morceaux

Nous allons nous intéresser ici au cas où certains paramètres des ddp ne sont plus constants sur la région mais peuvent varier de manière linéaire ou quadratique. De plus, nous nous limiterons à deux exemples de ddp qui nous intéressent particulièrement : les ddp gaussiennes qui interviennent en imagerie vidéo et les ddp gamma, dans la mesure où elles sont particulièrement utiles en imagerie SAR. Ainsi, jusqu'à présent nous avons supposé que pour ces deux ddp, l'espérance mathématique  $m_r$  dans chaque région  $\Omega_r$  était constante. Nous supposerons maintenant que les espérances mathématiques dépendent de la position du pixel considéré : dans la région  $\Omega_r$ , elle sera ainsi notée  $m_r(x, y)$ . Nous allons dans un premier temps supposer que cette espérance mathématique  $m_r(x, y)$  est décrite par une fonction linéaire. Nous montrerons alors que cette généralisation ne pose aucun problème pour une ddp gaussienne et peut même être étendue au cas où la moyenne  $m_r(x, y)$  est décrite par une fonction quadratique. En revanche, nous verrons que cette généralisation n'est pas possible pour une ddp gamma.

### A.6.1 ddp gaussienne et espérance mathématique distribuée selon une fonction linéaire

Analysons tout d'abord le cas de ddp gaussiennes. Nous supposerons que l'espérance mathématique dans une région  $\Omega_r$  donnée, dépend de la position  $(x, y)$  selon l'équation :

$$m_r(x, y) = a_r x + b_r y + c_r \quad (\text{A.24})$$

En revanche, nous supposerons que l'écart-type  $\sigma_r$  est toujours constant sur toute la région. La ddp d'une telle loi gaussienne s'écrit donc

$$P_{(m_r(x, y), \sigma_r)} [s(x, y)] = \frac{1}{\sqrt{2\pi}\sigma_r} e^{-\frac{1}{2}\left(\frac{s(x, y) - m_r(x, y)}{\sigma_r}\right)^2} \quad (\text{A.25})$$

et la log-vraisemblance dans la région  $\Omega_r$  devient

$$\begin{aligned} \mathcal{L}_e[\Omega_r | a_r, b_r, c_r, \sigma_r] &= -N_r \log 2\pi - N_r \log \sigma_r - \frac{1}{2\sigma_r^2} \sum_{(x, y) \in \Omega_r} [s(x, y) - m_r(x, y)]^2 \\ &= -N_r \log 2\pi - N_r \log \sigma_r - \frac{1}{2\sigma_r^2} \sum_{(x, y) \in \Omega_r} [s(x, y) - a_r x - b_r y - c_r]^2 \end{aligned} \quad (\text{A.26})$$

Les paramètres  $a_r$ ,  $b_r$  et  $c_r$  étant inconnus, on considérera leur estimée au sens du MV. Nous devons donc résoudre les trois équations suivantes :

$$\begin{aligned} \frac{\partial \mathcal{L}_e[\Omega_r | \hat{a}_r, \hat{b}_r, \hat{c}_r, \sigma_r]}{\partial \hat{a}_r} &= 0 \iff \sum_{\Omega_r} x (s(x, y) - \hat{a}_r x - \hat{b}_r y - \hat{c}_r) = 0 \\ \frac{\partial \mathcal{L}_e[\Omega_r | \hat{a}_r, \hat{b}_r, \hat{c}_r, \sigma_r]}{\partial \hat{b}_r} &= 0 \iff \sum_{\Omega_r} y (s(x, y) - \hat{a}_r x - \hat{b}_r y - \hat{c}_r) = 0 \\ \frac{\partial \mathcal{L}_e[\Omega_r | \hat{a}_r, \hat{b}_r, \hat{c}_r, \sigma_r]}{\partial \hat{c}_r} &= 0 \iff \sum_{\Omega_r} (s(x, y) - \hat{a}_r x - \hat{b}_r y - \hat{c}_r) = 0 \end{aligned} \quad (\text{A.27})$$

que l'on peut ré-écrire sous forme matricielle

$$\begin{pmatrix} \sum_{\Omega_r} x^2 & \sum_{\Omega_r} xy & \sum_{\Omega_r} x \\ \sum_{\Omega_r} xy & \sum_{\Omega_r} y^2 & \sum_{\Omega_r} y \\ \sum_{\Omega_r} x & \sum_{\Omega_r} y & \sum_{\Omega_r} 1 \end{pmatrix} \begin{pmatrix} \hat{a}_r \\ \hat{b}_r \\ \hat{c}_r \end{pmatrix} = \begin{pmatrix} \sum_{\Omega_r} x s(x, y) \\ \sum_{\Omega_r} y s(x, y) \\ \sum_{\Omega_r} s(x, y) \end{pmatrix} \quad (\text{A.28})$$

Une simple inversion matricielle nous permet donc de trouver les valeurs de  $\hat{a}_r$ ,  $\hat{b}_r$  et  $\hat{c}_r$ . Il ne reste maintenant plus qu'à estimer  $\sigma_r$  au sens du MV. On obtient facilement :

$$\hat{\sigma}_r^2 = \frac{1}{N_r} \sum_{(x, y) \in \Omega_r} (s(x, y) - \hat{a}_r x - \hat{b}_r y - \hat{c}_r)^2 \quad (\text{A.29})$$

où nous avons évidemment remplacé  $a_r$ ,  $b_r$  et  $c_r$  par leurs estimées. En remplaçant dans (A.26) les paramètres vrais par leurs estimées, nous obtenons la log-vraisemblance généralisée suivante :

$$\mathcal{L}_e[\Omega_r | \hat{a}_r, \hat{b}_r, \hat{c}_r, \hat{\sigma}_r] = -\frac{N_r}{2} \log \hat{\sigma}_r^2 - \frac{N_r}{2} - N_r \log 2\pi \quad (\text{A.30})$$

Nous voyons ainsi que l'équation donnant la log-vraisemblance généralisée est rigoureusement la même, que l'on suppose l'espérance mathématique constante sur la région ou distribuée selon une fonction linéaire (équation A.30 et tableau 4.2). La seule différence est dans le calcul des paramètres estimés de l'espérance mathématique et de la variance.

Remarquons de plus que même si, dans le cas où l'espérance mathématique est distribuée selon une fonction linéaire, il est nécessaire de calculer trois paramètres pour estimer l'espérance mathématique (équation A.28), tous les termes intervenant dans ce calcul et nécessitant des sommes sur la région s'écrivent sous la forme

$$T_{i,j,k}(s(x, y)) = \sum_{(x,y) \in \Omega_r} x^i y^j [s(x, y)]^k \quad (\text{A.31})$$

avec  $i$ ,  $j$  et  $k$  des entiers appartenant à  $\{0, 1, 2\}$ . Il est donc possible de calculer ces termes à l'aide de l'algorithme rapide détaillé en partie 2.2.5. De même, après un développement de (A.29), nous pouvons voir que le calcul de  $\hat{\sigma}_r^2$  peut lui aussi s'effectuer à l'aide des sommes fournies par (A.31) et peut donc être calculé à l'aide de l'algorithme rapide.

Analysons maintenant ce qu'il en est du terme de codage des paramètres  $\Delta_P$ , ce qui revient en fait d'après (2.8) à trouver le nombre de paramètres  $\alpha$  de la ddp. Il suffit pour cela de remarquer que dans le cas où l'espérance mathématique est distribuée selon une fonction linéaire, la ddp dépend non plus de 2 mais de 4 paramètres qui sont  $\sigma_r$  ainsi que  $a_r$ ,  $b_r$  et  $c_r$ . Nous avons donc dans ce cas  $\alpha = 4$ .

En conclusion de ce paragraphe, nous pouvons dire que dans le cas d'une ddp gaussienne, le modèle se généralise très facilement lorsque l'espérance mathématique est supposée varier selon une fonction linéaire. Le terme de codage des niveaux de gris (c'est-à-dire l'opposée de la log-vraisemblance généralisée) est alors identique au cas où l'espérance mathématique est constante sur la région : la seule différence réside dans le calcul des paramètres de la ddp au sens du MV, à savoir l'espérance mathématique et la variance. Dans tous les cas, il est important de souligner que l'algorithme rapide peut être utilisé, permettant un calcul rapide de ces paramètres.

## A.6.2 ddp gaussienne et espérance mathématique distribuée selon une fonction quadratique

Nous allons montrer très succinctement qu'il est possible de généraliser de façon similaire ces résultats au cas d'une ddp gaussienne dont la moyenne  $m_r$  varie non plus selon une fonction linéaire, mais selon une fonction quadratique :

$$m_r(x, y) = a_r x^2 + b_r y^2 + c_r xy + d_r x + e_r y + f_r \quad (\text{A.32})$$

En reprenant exactement la même démarche que dans le paragraphe précédent, nous pouvons estimer les paramètres  $\hat{a}_r$ ,  $\hat{b}_r$ ,  $\hat{c}_r$ ,  $\hat{d}_r$ ,  $\hat{e}_r$  et  $\hat{f}_r$  de l'espérance mathématique au sens du

MV :

$$\begin{pmatrix} \sum x^4 & \sum x^2 y^2 & \sum x^3 y & \sum x^3 & \sum x^2 y & \sum x^2 \\ \sum x^2 y^2 & \sum y^4 & \sum x y^3 & \sum x y^2 & \sum y^3 & \sum y^2 \\ \sum x^3 y & \sum x y^3 & \sum x^2 y^2 & \sum x^2 y & \sum x y^2 & \sum x y \\ \sum x^3 & \sum x y^2 & \sum x^2 y & \sum x^2 & \sum x y & \sum x \\ \sum x^2 y & \sum y^3 & \sum x y^2 & \sum x y & \sum y^2 & \sum y \\ \sum x^2 & \sum y^2 & \sum x y & \sum x & \sum y & \sum 1 \end{pmatrix} \begin{pmatrix} \hat{a}_r \\ \hat{b}_r \\ \hat{c}_r \\ \hat{d}_r \\ \hat{e}_r \\ \hat{f}_r \end{pmatrix} = \begin{pmatrix} \sum x^2 s(x, y) \\ \sum y^2 s(x, y) \\ \sum x y s(x, y) \\ \sum x s(x, y) \\ \sum y s(x, y) \\ \sum s(x, y) \end{pmatrix} \quad (\text{A.33})$$

où le symbole  $\sum$  est en fait une sommation sur  $\Omega_r$ , c'est-à-dire  $\sum_{(x,y) \in \Omega_r}$ . Ainsi, nous voyons que l'utilisation de l'accélération introduite en partie 2.2.5 peut là encore être utilisée pour calculer tous les termes intervenant dans les deux matrices de gauche et de droite de l'équation A.33, permettant ainsi le calcul rapide - au moyen d'une simple résolution d'un système  $6 \times 6$  - des 6 paramètres de l'espérance mathématique.

De même, nous pouvons estimer la variance au sens du MV :

$$\hat{\sigma}_r^2 = \frac{1}{N_r} \sum_{(x,y) \in \Omega_r} \left( s(x, y) - \hat{a}_r x^2 - \hat{b}_r y^2 - \hat{c}_r x y - \hat{d}_r x - \hat{e}_r y - \hat{f}_r \right)^2 \quad (\text{A.34})$$

expression qui après un simple développement peut se calculer en utilisant l'accélération présentée en partie 2.2.5.

Quant à la log-vraisemblance généralisée, elle s'écrit toujours sous la forme donnée dans (A.30), en utilisant bien sûr l'expression de  $\hat{\sigma}_r^2$  donnée ci-dessus.

Pour résumer, dans le cas d'une ddp gaussienne, le terme de codage des niveaux de gris reste toujours le même (voir tableau 4.2) : seules les expressions des estimées des paramètres varient. Dans le cas où l'espérance mathématique est constante, il suffit de prendre comme paramètres estimés ceux fournis dans le tableau 4.2, alors que lorsque l'espérance mathématique suit une fonction linéaire (respectivement, une fonction quadratique), les paramètres sont donnés par (A.28) (respectivement, par l'équation A.33). De plus, il faut aussi prendre en compte dans le calcul du terme de codage des paramètres, le nombre de paramètres  $\alpha$  de la ddp : dans le cas où l'espérance mathématique est constante,  $\alpha = 2$ , pour le cas linéaire,  $\alpha = 4$  et pour le cas quadratique,  $\alpha = 7$ .

### A.6.3 ddp gamma et espérance mathématique distribuée selon une fonction linéaire

Nous allons maintenant voir que dans le cas d'une ddp gamma, une telle généralisation n'est plus possible. Nous supposerons donc ici que l'espérance mathématique  $m_r(x, y)$  de la ddp gamma dans la région  $\Omega_r$  suit l'équation d'une fonction linéaire, donnée dans (A.24).

Afin de simplifier les calculs, nous supposerons que l'ordre  $L$  de cette ddp est égal à 1 : il s'agit donc en fait d'une ddp exponentielle qui s'écrit dans notre cas

$$P_{m_r(x,y)} [s(x, y)] = \frac{1}{m_r(x, y)} e^{-\frac{s(x, y)}{m_r(x, y)}} \quad (\text{A.35})$$

La log-vraisemblance dans la région  $\Omega_r$  s'obtient aisément :

$$\begin{aligned} \mathcal{L}_e[\Omega_r | m_r] &= -\sum_{(x,y) \in \Omega_r} \log m_r(x, y) - \sum_{(x,y) \in \Omega_r} \frac{s(x, y)}{m_r(x, y)} \\ &= -\sum_{(x,y) \in \Omega_r} \log(a_r x + b_r y + c_r) - \sum_{(x,y) \in \Omega_r} \frac{s(x, y)}{a_r x + b_r y + c_r} \end{aligned} \quad (\text{A.36})$$

Comme précédemment, nous cherchons à estimer  $a_r$ ,  $b_r$  et  $c_r$  au sens du MV, ce qui conduit aux trois équations ci-dessous :

$$\begin{aligned}\frac{\partial \mathcal{L}_e[\Omega_r|m_r]}{\partial a_r} = 0 &\iff \sum_{(x,y) \in \Omega_r} \left( \frac{x}{a_r x + b_r y + c_r} + \frac{x s(x,y)}{(a_r x + b_r y + c_r)^2} \right) = 0 \\ \frac{\partial \mathcal{L}_e[\Omega_r|m_r]}{\partial b_r} = 0 &\iff \sum_{(x,y) \in \Omega_r} \left( \frac{y}{a_r x + b_r y + c_r} + \frac{y s(x,y)}{(a_r x + b_r y + c_r)^2} \right) = 0 \\ \frac{\partial \mathcal{L}_e[\Omega_r|m_r]}{\partial c_r} = 0 &\iff \sum_{(x,y) \in \Omega_r} \left( \frac{1}{a_r x + b_r y + c_r} + \frac{s(x,y)}{(a_r x + b_r y + c_r)^2} \right) = 0\end{aligned}\quad (\text{A.37})$$

Il ressort de ce système deux conclusions. La première est qu'il n'est pas possible de résoudre analytiquement un tel système et donc de trouver d'expression pour les estimées  $\hat{a}_r$ ,  $\hat{b}_r$  et  $\hat{c}_r$  de  $a_r$ ,  $b_r$  et  $c_r$  au sens du MV. Évidemment, nous pourrions imaginer un algorithme de descente permettant d'estimer rapidement ces valeurs  $\hat{a}_r$ ,  $\hat{b}_r$  et  $\hat{c}_r$ . Cependant, et cela constitue la deuxième conclusion, chacune des sommes sur  $\Omega_r$  qui intervient dans le système d'équations et qu'il sera nécessaire de calculer avec un tel algorithme de descente, dépend des valeurs de  $a_r$ ,  $b_r$  et  $c_r$ . Les termes de ces sommes étant à chaque itération différents, il n'est plus possible d'utiliser l'algorithme rapide de la partie 2.2.5<sup>34</sup>. Ainsi, chaque fois qu'il sera nécessaire de recalculer  $\hat{a}_r$ ,  $\hat{b}_r$  et  $\hat{c}_r$  (c'est-à-dire à chaque modification de la grille), des calculs nécessitant des sommes des niveaux de gris sur des régions de l'image devront être effectuées, ce qui est rédhibitoire du point de vue du temps de calcul.

Nous voyons ainsi que pour une ddp gamma, il n'est plus possible de généraliser notre algorithme tout en conservant une approche rapide, lorsque l'espérance mathématique n'est plus constante sur chaque région mais est distribuée linéairement.

---

34. En effet, l'algorithme rapide repose sur le pré-c calcul d'une image  $F_k$  (annexe A.5) une fois pour toute lors de l'initialisation. Cependant, comme dans le cas présent l'expression à sommer varie continuellement, il sera nécessaire de recalculer cette image  $F_k$  à chaque modification de la grille. L'utilisation de cet algorithme n'est donc plus justifiée.

## A.7 Contour actif et transmission des paramètres de la cible

Dans cette annexe, nous allons déterminer la complexité stochastique associée à un contour actif dont les paramètres de la ddp de la cible sont connus.

Supposons que l'image soit composée de deux régions, la région cible notée  $\Omega_a$ , et la région fond notée  $\Omega_b$ , avec des niveaux de gris distribués selon des ddp gaussiennes :

$$P_{(m_r, \sigma_r)}(x) = \frac{1}{\sqrt{2\pi}\sigma_r} e^{-\frac{1}{2}(\frac{x-m_r}{\sigma_r})^2} \quad (\text{A.38})$$

où l'on doit remplacer  $r$  par  $a$  dans la région cible et par  $b$  sur le fond. La log-vraisemblance dans la région  $\Omega_r$  ( $\Omega_r = \Omega_a$  ou  $\Omega_b$ ) s'écrit alors

$$\mathcal{L}_e[\Omega_r | m_r, \sigma_r] = -\frac{1}{2}N_r \log 2\pi - \frac{1}{2}N_r \log \sigma_r^2 - \frac{1}{\sigma_r^2} \sum_{(x,y) \in \Omega_r} [s(x,y) - m_r]^2 \quad (\text{A.39})$$

Les paramètres du fond  $m_b$  et  $\sigma_b$  étant supposés inconnus, nous les remplaçons par leur estimé  $\hat{m}_b$  et  $\hat{\sigma}_b$  au sens du MV (dont les valeurs sont fournies dans le tableau 4.2). Ainsi, pour la région fond  $\Omega_b$ , nous obtenons la log-vraisemblance généralisée suivante :

$$\mathcal{L}_e[\Omega_b | \hat{m}_b, \hat{\sigma}_b] = -\frac{1}{2}N_b \log 2\pi - \frac{1}{2}N_b \log (\hat{\sigma}_b)^2 - \frac{N_b}{2} \quad (\text{A.40})$$

En revanche, nous supposerons que les paramètres de la cible sont connus. Il est donc inutile de les estimer et la log-vraisemblance de la cible  $\Omega_a$  s'écrit :

$$\begin{aligned} \mathcal{L}_e[\Omega_a | m_a, \sigma_a] &= -\frac{1}{2}N_a \log 2\pi - \frac{1}{2}N_a \log \sigma_a^2 \\ &\quad -\frac{1}{\sigma_a^2} \left( \sum_{(x,y) \in \Omega_a} s(x,y)^2 - 2m_a \sum_{(x,y) \in \Omega_a} s(x,y) + N_a m_a^2 \right) \end{aligned} \quad (\text{A.41})$$

Le terme de codage des niveaux de gris sur les deux régions  $\Omega_a$  et  $\Omega_b$  s'écrit donc

$$\begin{aligned} \Delta_L(\mathbf{s}, w, (m_a, \sigma_a, \hat{m}_b, \hat{\sigma}_b)) &= -\mathcal{L}_e[\Omega_a | m_a, \sigma_a] - \mathcal{L}_e[\Omega_b | \hat{m}_b, \hat{\sigma}_b] \\ &= \frac{1}{\sigma_a^2} \left( \sum_{(x,y) \in \Omega_a} s(x,y)^2 - 2m_a \sum_{(x,y) \in \Omega_a} s(x,y) + N_a m_a^2 \right) \\ &\quad + N_a \log \sigma_a + \frac{1}{2}N_b \log (\hat{\sigma}_b)^2 + \frac{N_b}{2} + \frac{1}{2}N \log 2\pi \end{aligned} \quad (\text{A.42})$$

Nous voyons que dans cette expression, toutes les sommes peuvent se calculer en utilisant l'algorithme rapide proposé dans la partie 2.2.5.

## A.8 Estimation de la ddp d'une image à $Q$ niveaux

Notre objectif dans cette partie est d'estimer la ddp  $P_r = (p_r(q))_{q \in \{1, 2, \dots, Q\}}$  d'une image à  $Q$  niveaux au sens du MV.

Nous devons maximiser

$$\mathcal{L}_e \left[ \Omega_r \mid (p_r(q))_{q \in \{1, 2, \dots, Q\}} \right] = \sum_{(x, y) \in \Omega_r} \sum_{q=1}^Q \delta[s(x, y), \lambda_q] \log p_r(q) \quad (\text{A.43})$$

sous la contrainte

$$\sum_{q=1}^Q p_r(q) = 1 \quad (\text{A.44})$$

Introduisons le paramètre de Lagrange  $\mu$  et la fonction de Lagrange  $\Psi(\mathbf{p})$  :

$$\Psi(\mathbf{p}) = \sum_{(x, y) \in \Omega_r} \sum_{q=1}^Q \delta[s(x, y), \lambda_q] \log p_r(q) - \mu \left( \sum_{q=1}^Q p_r(q) - 1 \right) \quad (\text{A.45})$$

avec  $\mathbf{p} = (p_1, p_2, \dots, p_Q)^T$  où  $T$  est l'opérateur de transposition. L'estimée de  $p_r(q)$  au sens du MV, notée  $\hat{p}_r(q)$ , est obtenue lorsque  $\partial \Psi(\mathbf{p}) / \partial p_r(q) = 0$  pour tout  $q$  appartenant à  $\{1, \dots, Q\}$ , ce qui conduit à

$$\sum_{(x, y) \in \Omega_r} \frac{\delta[s(x, y), \lambda_q]}{\hat{p}_r(q)} - \mu = 0 \quad (\text{A.46})$$

ou

$$\hat{p}_r(q) = \frac{1}{\mu} \sum_{(x, y) \in \Omega_r} \delta[s(x, y), \lambda_q] \quad (\text{A.47})$$

Comme  $N_r = \sum_{(x, y) \in \Omega_r} \sum_{q=1}^Q \delta[s(x, y), \lambda_q]$ , on en déduit  $\mu = N_r$  et

$$\hat{p}_r(q) = \frac{1}{N_r} \sum_{(x, y) \in \Omega_r} \delta[s(x, y), \lambda_q] \quad (\text{A.48})$$

## Annexe B

### Acronymes

<b>ASAR</b>	: <i>Advanced Synthetic Aperture Radar</i>
<b>ddp</b>	: densité de probabilité (employée aussi bien pour les lois continues que discrètes)
<b>ENVISAT</b>	: <i>Environmental Satellite</i>
<b>ERS</b>	: <i>European Remote Sensing satellite</i>
<b>GLRT</b>	: <i>Generalized Likelihood Ratio Test</i>
<b>MCS</b>	: Minimisation de la Complexité Stochastique
<b>MDL</b>	: <i>Minimum Description Length</i>
<b>MV</b>	: Maximum de Vraisemblance
<b>RVB</b>	: Espace de couleur : R = Rouge - V = Vert - B = Bleu
<b>SAR</b>	: <i>Synthetic Aperture Radar</i>
<b>YHS</b>	: Espace de couleur : Y = Luminance - H = teinte ( <i>Hue</i> ) - S = Saturation
<b>YUV</b>	: Espace de couleur : Y = Luminance - U et V : composantes colorimétriques



## Annexe C

### Liste des travaux

#### Articles publiés dans des revues internationales

- [AI.1] F. Galland, N. Bertaux and Ph. Réfrégier. *Minimum Description Length Synthetic Aperture Radar image segmentation*, IEEE Image Processing, vol 12, n 9, pp 995-1006, 2003.
- [AI.2] F. Galland and Ph. Réfrégier. *Information theory-based snake adapted to multi-region objects with different noise models*, Optics Letters, Volume 29, n 14, pp 1611-1613, 2004.
- [AI.3] F. Goudail, P. Terrier, Y. Takakura, L. Bigué, F. Galland et V. Devlaminck. *Target detection with a liquid crystal-based passive Stokes polarimeter*, Applied Optics, Volume 43-2, pp. 274-282, 2004.
- [AI.4] F. Galland, Ph. Réfrégier and O. Germain. *Synthetic Aperture Radar oil spill segmentation by stochastic complexity minimization*, IEEE Geoscience and Remote Sensing Letters, vol 1, issue 4, pp 295-299, oct 2004
- [AI.5] F. Galland, N. Bertaux and Ph. Réfrégier. *Multicomponent image segmentation in homogeneous regions based on description length minimization : application to speckle, Poisson and Bernoulli noise*, Accepté pour publication dans Pattern Recognition.

#### Conférences internationales avec comité de sélection

- [CI.1] F. Galland, N. Bertaux and Ph. Réfrégier. *Merge, move and remove MDL based segmentation for Synthetic Aperture Radar images*, ACIVS 2002, Gand (Belgique), 9-11 septembre 2002, pp 307-314.
- [CI.2] F. Galland, N. Bertaux and Ph. Réfrégier. *Active grid segmentation by stochastic complexity minimisation adapted to different noise models*, PSIP 2003, Grenoble (France), 29-31 janvier 2003, pp 73-76.
- [CI.3] F. Goudail, F. Galland and Ph. Réfrégier. *A general framework for designing image processing algorithms for coherent polarimetric images*, ICIP 2003, Barcelone (Espagne), 14-17 septembre 2003, Volume III, pp153-156.
- [CI.4] F. Galland, O. Germain, N. Bertaux and Ph. Réfrégier. *Fast and automatic oil spill segmentation in Synthetic Aperture Radar images*, ESA EUSC 2004, Madrid (Espagne), 17-18 mars 2004.
- [CI.5] Ph. Réfrégier, F. Galland, P. Martin, G. Delyon, N. Bertaux, F. Goudail. *Minimum Description Length segmentation adapted to different image sensors*, SPIE 2004, Londres (Angleterre), 25-28 octobre 2004. Conférence invitée.

**Conférences françaises avec comité de sélection**

- [CF.1] P. Terrier, O. Lossan, V. Devlaminck, F. Galland, F. Goudail et Ph. Réfrégier. *Segmentation non supervisée d'images polarimétriques passives*, Colloque GRETSI, Paris, 8-11 septembre 2003.
- [CF.2] F. Galland, N. Bertaux et Ph. Réfrégier. *Segmentation d'image par minimisation de la complexité stochastique*, MAJECSTIC 2003, Marseille, 29-31 octobre 2003. Papier sélectionné parmi les meilleurs du colloque afin d'être publié dans la revue en ligne ISDM (<http://isdm.univ-tln.fr>)

# Bibliographie

- [1] W. Alpers and H. Huhnerfuss. Radar signatures of oil films floating on the sea surface and the marangoni effect. *J. Geophys. Res.*, 93 :3642–3648, 1988.
- [2] P. Y. Arquès. *Décisions en traitement du signal*. Masson, Paris, 1982.
- [3] G. Auber and P. Kornprobst. Mathematical problems in image processing. partial differential equations and the calculus of variations. *Applied Mathematical Sciences, Springer Verlag*, 147, 2001.
- [4] A. Azzalini. *Statistical Inference - Based on the likelihood*. Chapman and Hall, New York, 1996.
- [5] B. C. Barber. Theory of digital imaging from orbital synthetic aperture radar. *International Journal of Remote Sensing*, 6 :1009–1057, 1985.
- [6] R. Bernier and H. Arsenault. Deconvolution of two-dimensional images with zeros in the transfert function. *Appl. Opt.*, 30(35) :5163–5168, 1991.
- [7] S. Beucher and F. Meyer. The morphological approach to segmentation : the watershed transformation. In *Mathematical Morphology in Image Processing*. Marcel Dekker, 1992.
- [8] A. C. Bovik. On detecting edges in speckle imagery. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(10) :1618–1627, 1988.
- [9] A. C. Bovik and D. C. Munson. Optimal detection of object boundaries in uncorrelated speckle. *Optical Engineering*, 25(11) :1246–1252, 1986.
- [10] J. Bruniquel. *Contribution de données multi-temporelles à l'amélioration radiométrique et à l'utilisation d'images de radars à synthèse d'ouverture*. Thèse de doctorat, Université Paul Sabatier de Toulouse, 1996.
- [11] J. Bruniquel and A. Lopès. On the true multilook intensity distribution in SAR imagery. In *Proc. International Geoscience and Remote Sensing Symposium (IGARSS'98)*, Seattle, USA, 1998.
- [12] D. Casasent, D. Weber, and M. Sipe. Detection filters and algorithm fusion for ATR. *IEEE Trans. on Image Processing*, 6 :114–125, Jan 1997.
- [13] A. Chakraborty, L. H. Staib, and J. S. Duncan. Deformable boundary finding in medical images by integrating gradient and region information. *IEEE Trans. on Medical Imaging*, 15(6) :859–870, 1996.
- [14] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10(22) :266–277, Feb. 2001.
- [15] J.M. Chassery and A. Montanvert. *Géométrie discrète en analyse d'images*. Editions Hermès, Paris, 1991.
- [16] C. Chesnaud, Ph. Réfrégier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. *IEEE Trans. Pattern Anal. and Machine Intell.*, 21(11) :1145–1157, 1999.

- [17] J. P. Cocquerez and S. Philipp. *Analyse d'images : filtrage et segmentation*. Masson, Paris, 1995.
- [18] L. D. Cohen. On active contour models and balloons. *CVGIP : Image Understanding*, 53(2) :211–218, 1991.
- [19] R. Cook and I. Mc Connell. MUM (merge using moments) segmentation for SAR images. In *Conf. on SAR Data Processing for Remote Sensing, Rome, Italy*, volume 2316, pages 92–103. SPIE, 1994.
- [20] R. Cook, I. Mc Connell, D. Stewart, and C. Oliver. Segmentation and simulated annealing. In *SPIE Proc. European Symposium on Remote Sensing, SAR Image Analysis, Modelling and Techniques*, Taormina, Italy, volume 2955, pages 30–37, 1996.
- [21] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-interscience, New York, 1991.
- [22] F. Del Frate, A. Petrocchi, J. Lichtenegger, and G. Calabresi. Neural networks for oil spill detection using ers-sar data. *IEEE TGARS*, 38(5), 2000.
- [23] ESA-CNES. *CEOS SAR Workshop, Land and Ocean applications*, Toulouse, France, volume ESA SP-450, Octobre 1999.
- [24] T. S. Ferguson. *Mathematical Statistics, a decision theoretic approach*. Academic Press, New York and London, 1967.
- [25] P. Ferrand, M. Egen, R. Zentel, J. Seekamp, S. G. Romanov, and C. M. Sotomayor Torres. Structuring of self-assembled three-dimensional photonic crystals by electron-beam lithography. *Applied Physics Letters*, 83(25) :5289–5291, 2003.
- [26] M. Figueiredo, J. Leitão, and A. K. Jain. Unsupervised contour representation and estimation using B-splines and a minimum description length criterion. *IEEE Trans. Image Processing*, 9 :1075–1087, 2000.
- [27] R. Fjørtoft. *Segmentation d'images radar par détection de contours*. Thèse de doctorat, Institut National Polytechnique de Toulouse (France), 1999.
- [28] R. Fjørtoft, A. Lopès, P. Marthon, and E. Cubero-Castan. An optimum multiedge detector for SAR image segmentation. *IEEE Trans. Geoscience and Remote Sensing*, 36(3) :793–802, 1998.
- [29] V. S. Frost, Shanmugan K.S., and Holtzman J. C. Edge detection for SAR and other noisy images. In *Proc. International Geoscience and Remote Sensing Symposium, Munich, Germany*, volume FA2, pages 4.1–4.9, 1982.
- [30] P.H. Garthwaite, I.T. Jolliffe, and B. Jones. *Statistical Inference*. Printice Hall Europe, London, 1995.
- [31] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6 :721–741, 1984.
- [32] J.J. Gerbrands and E. Backer. Split and merge segmentation of slar imagery : Segmentation consistency. *IEEE 7th Int. Conf. Pattern Recognition*, 1(21) :284–286, 1984.
- [33] O. Germain. *Segmentation d'images radar : caractérisation de détecteurs de bords et apport des contours actifs*. Thèse de doctorat, Université Aix-Marseille III, Marseille, France, 2001.
- [34] O. Germain and Ph. Réfrégier. Optimal snake-based segmentation of a random lumiance target on a spatially disjoint background. *Opt. Lett.*, 21(22) :1845–1847, 1996.

- 
- [35] O. Germain and Ph. Réfrégier. Snake-based method for the segmentation of objects in multichannel images degraded by speckle. *Optics Letters*, 24(12) :814–816, June 1999.
  - [36] O. Germain and Ph. Réfrégier. On the bias of the Likelihood Ratio edge detector for SAR images. *IEEE Trans. Geoscience and Remote Sensing*, 38(3) :1455–1458, 2000.
  - [37] O. Germain and Ph. Réfrégier. Edge location in SAR images : Performance of the Likelihood Ratio filter and accuracy improvement with an active contour approach. *IEEE Trans. Image Processing*, 10(1) :73–78, Jan. 2001.
  - [38] O. Germain and Ph. Réfrégier. Statistical active grid for segmentation refinement. *Pattern Recognition Letters*, 22(3) :1125 –1132, 2001.
  - [39] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Fast geodesic active contours. *IEEE Transactions on Image Processing*, 10 :1467–1475, 2001.
  - [40] R. C. Gonzalez and R. E. Woods. *Digital image processing*. Addison-Wesley, Reading, 1992.
  - [41] J. W. Goodman. *Statistical Properties of laser Speckle Patterns*, chapter Laser Speckle and Related Phenomena, pages 9–75. Springer-Verlag (Topics in Applied Physics Vol. 9), Heidelberg, 1975.
  - [42] J. W. Goodman. *Statistical Optics*. John Wiley and Sons, Inc., New York, 1985.
  - [43] S. Huard. *Polarization of light*. Wiley, Masson, Paris, 1997.
  - [44] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9) :850–863, 1993.
  - [45] E. Jakeman and P. N. Pusey. A model for non-Rayleigh sea echo. *IEEE Trans. on Antennas and Propagation*, AP-24(6) :806–814, 1976.
  - [46] E. Jakeman and J. A. Tough. Generalized K-distribution : a statistical model for weak scattering. *Journal of the Optical Society of America A*, 4(9) :1764–1772, 1987.
  - [47] J. K. Jao. Amplitude distribution of composite terrain radar clutter and the K-distribution. *IEEE Trans. on Antennas and Propagation*, AP-32(10) :1049–1062, 1984.
  - [48] S. Jehan-Besson, G. Aubert, and M. Barlaud. A 3-step algorithm using region-based active contours for video objects detection. *EURASIP Journal on Applied Signal Processing*, 6 :572–581, 2002.
  - [49] T. Kanungo, B. Dom, W. Niblack, and D. Steele. A fast algorithm for mdl-based multi-band image segmentation. *Proc. Comp. Vision and Patt. Recognition*, CVPR :609–616, 1994.
  - [50] M. Kass, A. Witkin, and D. Terzopoulos. Snakes : Active contour models. *International Journal of Computer Vision*, 1 :321–331, 1988.
  - [51] M. Kass, A. Witkin, and D. Terzopoulos. Snakes : Active contour models. *International Journal of Computer Vision*, 1 :321–331, 1988.
  - [52] P. A. Kelly, H. Derin, and K. D. Hartt. Adaptative segmentation of speckled images using a hierarchical random field model. *IEEE Trans. Acoustics, Speech and Signal Processing*, 36(10) :1628–1641, 1988.
  - [53] C. Kervrann and F. Heitz. A hierarchical statistical framework for the segmentation of deformable objects in image sequences. *Proc. IEEE Conf. Comp. Vision Pattern Recognition, Seattle*, pages 724–728, 1994.

- [54] A. D. Lanterman. Schwarz, wallace, and rissanen : Intertwining themes in theories of model selection. *International Statistical Review*, 69(2) :185–212, 2001.
- [55] D. Lebedeff. *Etude de la quantification vectorielle des données brutes issues d'un radar à synthèse d'ouverture*. Thèse de doctorat, Université de Nice - Sophia Antipolis, 1995.
- [56] Y.G. Leclerc. Constructing simple stable descriptions for image partitioning. *Computer Vision*, 3 :73–102, 1989.
- [57] J. Lee and I. Jurkevich. Segmentation of SAR images. *IEEE Trans. Geoscience and Remote Sensing*, 27(6) :674–680, 1989.
- [58] J. S. Lee, M. R. Grunes, and R. Kwok. Classification of multi-look polarimetric SAR imagery based on complex Wishart distribution. *Int. J. Remote Sensing*, 15(11) :2299–2311, 1994.
- [59] T.C.M. Lee. Segmenting images corrupted by correlated noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5) :481–492, 1998.
- [60] A. Lopès, J. Bruniquel, F. Séry, J. C. Souyris, and F. Adragna. Optimal target detection using one channel SAR complex imagery : Application to ship detection. In *Proc. International Geoscience and Remote Sensing Symposium (IGARSS'98)*, Seattle, USA, 1998.
- [61] R. Malladi, J. Sethian, and B. Venuri. Shape modelling with front propagation. *IEEE Trans. Pattern Anal. and Machine Intell.*, 17 :158–175, 1995.
- [62] P. Martin, Ph. Réfrégier, F. Goudail, and F. Guérault. Influence of the noise model on level set active contour segmentation. *IEEE Trans. Pattern Anal. and Machine Intell.*, 26 :799–803, 2004.
- [63] A. Marzouki. *Segmentation statistique d'images radar*. Thèse de doctorat, Université des sciences et technologies de Lille, 1996.
- [64] D. Massonet, M. Rossi, C. Carmona, F. Adragna, G. Peltzer, K. Feigl, and T. Rabaute. The displacement field of the landers earthquake mapped by radar interferometry. *Nature*, 364 :138–142, 1993.
- [65] J. Montagnat, H. Delingette, and N. Ayache. A review of deformable surfaces : topology, geometry and deformation. *Image and Vision Computing*, 19(14) :1023–1040, 2001.
- [66] R. J. Muirhead. *Aspects of multivariate statistical theory*. Wiley, New York, 1982.
- [67] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42 :577–684, 1989.
- [68] C. Nguyen, I. Herlin, and C. Graffigne. A deformable region model using stochastic processes applied to echadiographic images. In *Conf. on Computer Vision and Pattern Recognition, Urbana Chapagn, USA*. IEEE, 1992.
- [69] J.M. Nicolas, M. Sigelle, C. Thuillier, and F. Tupin. Images de radar à ouverture synthétique : Transformée de mellin et multirésolution. *Seizieme colloque GRETSI*, 2 :797–800, 1997.
- [70] C. J. Oliver, D. Blacknell, and R. G. White. Optimum edge detection in SAR. *IEE Proc. Radar Sonar Navigat.*, 143 :31–40, Feb. 1996.
- [71] C. J. Oliver, A. Blake, and R. White. Optimum texture analysis of synthetic aperture radar images. In *SPIE Proc. International Conf. on Algorithms for SAR imagery*, volume 2230, pages 389–398, 1994.

- 
- [72] C. J. Oliver and S. Quegan. *Understanding SAR images*, chapter Fundamental Properties of SAR images, pages 75–122. Artech House, Norwood, 1998.
  - [73] C. J. Oliver and S. Quegan. *Understanding SAR images*, chapter Principles of SAR image formation, pages 11–42. Artech House, Norwood, 1998.
  - [74] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *International Journal of Computer Vision*, 46(3) :223, 2002.
  - [75] P. Pavlakis, D. Tarchi, and A. J. Sieber. On the monitoring of illicit vessel discharges using spaceborn sar remote sensing - a reconnaissance study in the mediterranean sea. *Ann. Telecommun.*, 56(11-12) :700–718, 2001.
  - [76] H. V. Poor. *An introduction to signal detection and estimation*, chapter Elements of hypothesis testing, pages 5–39. Springer Verlag, New York, 1994.
  - [77] E. Rignot and R. Chellappa. Segmentation of synthetic-aperture-radar complex data. *J. Opt. Soc. Amer. A*, 8(9) :1499–1509, 1991.
  - [78] J. Rissanen. Modeling by shortest data description. *Automatica*, 14 :465–471, 1978.
  - [79] J. Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.
  - [80] R. Ronfard. Region-based strategies for active contour models. *International Journal of Computer Vision*, 2(13) :229–251, 1994.
  - [81] O. Ruch. *Reconnaissance des formes par Contour Actif Statistique - Application à l'imagerie optronique active*. Thèse de doctorat, Université Aix-Marseille III, Marseille, France, 2001.
  - [82] O. Ruch and Ph. Réfrégier. Minimal-complexity segmentation with a polygonal snake adapted to different optical noise models. *Opt. Lett.*, 26(13) :977–979, 2001.
  - [83] S. J. Sangwine and R. E. N. Horne, editors. *The Colour Image Processing Handbook*. Optoelectronics Imaging and Sensing. Chapman and Hall, London, 1998. ISBN 0-412-80620-7.
  - [84] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27 :379–423 ; 623 – 656, 1948.
  - [85] P. C. Smits and S. G. Dellepiane. Synthetic aperture radar image segmentation by a detail preserving markov random field. *IEEE Trans. Geoscience and Remote Sensing*, 35(4) :844–857, 1997.
  - [86] A. H. S. Solberg, G. Storvik, R. Solberg, and E. Volden. Automatic detection of oil spills in ers sar images. *IEEE TGARS*, 37(4) :1916–1924, 1999.
  - [87] L.H. Staib and J. S. Duncan. Boundary finding with parametrically deformable contour models. *IEEE Trans. Pattern Anal. and Machine Intell.*, 14(11) :1061–1075, 1992.
  - [88] G. Storvik. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(10) :976–986, 1994.
  - [89] P. Terrier and V. DeVlaminck. Robust and accurate estimate of the orientation of partially polarized light from a camera sensor. *Applied Optics - IP*, 40(29) :5233–5239, 2001.
  - [90] A. N. Tikhonov and V. Y. Arsenin. *Solutions of ill-posed problems*. John Wiley and Sons, Inc., New York, 1977.

- [91] T. L. Toan, F. Ribbes, L. F. Wang, N. Flouzy, K. H. Ding, J. A. Kong, M. Fujita, and T. Kurosu. Rice crop mapping and monitoring using ERS-1 data based on experiment and modeling results. *IEEE Trans. Geoscience and Remote Sensing*, 35 :41–55, 1997.
- [92] R. Touzi, A. Lopès, and P. Bousquet. A statistical and geometrical edge detector for SAR images. *IEEE Trans. Geoscience and Remote Sensing*, 26(6) :764–773, 1988.
- [93] E. Trouvé. *Imagerie interférentielle en radar à visée latérale*. Thèse de doctorat, Ecole Nationale Supérieure des Télécommunications, 1996.
- [94] F. Tupin. *Reconnaissance des formes et analyse de scènes en imagerie radar à ouverture synthétique*. Thèse de doctorat, Ecole Nationale Supérieure des Télécommunications, 1997.
- [95] F. Tupin, H. Maître, J. F. Mangin, J. M. Nicolas, and E. Pechersky. Detection of linear structures in SAR images : Application to road network extraction. *IEEE Trans. Geoscience and Remote Sensing*, 36(2) :434–453, 1998.
- [96] L. Vincent and P. Soille. Watersheds in digital spaces : an efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(6) :583–598, 1991.
- [97] R. G. White. Change detection in SAR imagery. *Int. Journal of Remote Sensing*, 12(2) :339–360, 1991.
- [98] D. J. Williams and M. Shah. A fast algorithm for active contours and curvature estimation. *CVGIP : Image Understanding*, 55(1) :14–26, 1992.
- [99] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Processing*, 7 :359–369, 1998.
- [100] S. C. Zhu and A. Yuille. Region competition : unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Trans. Pattern Anal. and Machine Intell.*, 18(9) :884–900, 1996.