POLYPUBLIE

Segmentation des images texturées par estimation Bayésienne

Hong Hai Nguyen

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

Date	1989
Type	Mémoire ou thèse / Dissertation or Thesis Mémoire de maîtrise/Master's thesis
URL de PolyPublie PolyPublie URL	https://publications.polymtl.ca/58268/
Conditions d'utilisation Terms of use	© Tous droits réservés / All rights reserved
Référence Citation	Nguyen, H. H. (1989). Segmentation des images texturées par estimation Bayésienne [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/58268/

Détails de la thèse
Thesis details

Directeurs de recherche Advisors	
Programme Program	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal
https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

SEGMENTATION DES IMAGES TEXTURÉES PAR ESTIMATION BAYESIENNE

par

Hong Hai Nguyen

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION

DU GRADE DE MAÎTRISE ES SCIENCES APPLIQUÉES (M.Sc.A.)

Octobre 1989

c Hong Hai Nguyen 1989

l Library
ıda

ın Theses Service

Canada

Bibliothèque nationale
du Canada

Service des thèses canadiennes

author has granted an irrevocable non-
ısive licence allowing the National Library
ınada to reproduce, loan, distribute or sell
ɛs of his/her thesis by any means and in
orm or format, making this thesis available
terested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

author retains ownership of the copyright
is/her thesis. Neither the thesis nor
ɜtantial extracts from it may be printed or
rwise reproduced without his/her per-
ion.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-58193-X

nada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE

Cette thèse (ou ce mémoire) intitulé(e):

SEGMENTATION DES IMAGES TEXTURÉES PAR ESTIMATION BAYESIENNE

présenté(e) par: Hong Hai Nguyen

en vue de l'obtention du grade de: Maîtrise en Sciences Appliquées

a été dûment accepté(e) par le jury d'examen constitué de:

M. Paul Cohen . . . . . ., Ph. D., président

M. Guy Mailloux. . . . . ., Ph.D.

M. Hoang Hai Hoc . . . . . ., Ph.D.

SOMMAIRE

Ce mémoire présente une nouvelle stratégie pour segmenter des images texturées. Cette stratégie est basée sur un modèle markovien hiérarchique dans lequel les textures et le processus de formation des régions sont modélisés par des champs aléatoires markoviens. L'algorithme proposé comprend deux caractéristiques principales:

(1) les observations originales sont converties, par la transformation de Hadamard, en des caractéristiques locales non correlées. Cette conversion conduit à une décomposition du problème de segmentation en des décisions stochastiques locales basées sur des fenêtres d'analyse de taille limitée;

(2) le nombre de textures et les paramètres du modèle représentant chaque texture sont évalués directement à partir des observations par une technique d'aggrégation floue hiérarchique.

Cet algorithme a été appliqué avec succès à la segmentation d'images composées de textures naturelles.

ABSTRACT

This report presents a new segmentation strategy for textured images. The strategy is based on a hierarchical model using discrete Markov random fields. The proposed algorithm involves two main characteristics:

(1) the original observations are converted into uncorrelated local features, through Hadamard transformation; this conversion leads to the decomposition of the segmentation problem into local stochastic decisions based on limited-size analysis windows;

(2) the number of textures in the image and the model parameters for each texture are directly evaluated from the observation by a hierarchical fuzzy clustering technique.

This algorithm has been applied with success in the segmentation of images composed of natural textures.

REMERCIEMENTS

Je tiens à remercier mon directeur de recherche, Dr. Paul Cohen, pour ses conseils tout au long de ce travail.

Je tiens aussi à remercier mon épouse pour son support moral. Sa présence m'a beaucoup encouragé.

Finalement, je désire remercier aussi mes confrères Minh, Louis Patrick, J.Yves pour les discussions utiles durant la réalisation du projet.

TABLE DES MATIÈRES

SOMMAIRE iv

ABSTRACT v

REMERCIEMENTS vi

LISTE DES FIGURES xii

INTRODUCTION 1

CHAPITRE I - MÉTHODES DE CARACTÉRISATION DES TEXTURES

1.1 Introduction 7

1.2 Approche statistique 8

1.2.1 Introduction 8

1.2.2 Méthodes basées sur la fréquence spatiale de l'image . . . 9

1.2.2.1 Caractérisation des textures par la fonction d'autocorrélation 9

1.2.2.2 Caractérisation des textures par la transformation de Fourier 11

1.2.3 Méthodes basées sur les statistiques du premier ordre . . . 12

1.2.3.1 La distribution de premier ordre de luminance 12

1.2.3.2 Méthodes basées sur la différence des niveaux de gris (GLDM) 14

1.2.4 Méthodes basées sur les statistiques du deuxième ordre . . 16

1.2.5 Méthode basée sur les matrices de longueur de suites de niveaux de gris (GLRLM) 22

1.2.6 Méthode basée sur l'énergie de texture 24

1.2.7 Méthode basée sur des masques orthogonaux 26

1.2.8 Méthodes basées sur les modèles 27

1.2.8.1 Modèle markovien pour la caractérisation des textures. . 28

1.2.8.2 Modèles autorégressifs pour les textures 40

1.2.9 Conclusion 41

1.3 Approche structurale 43

1.3.1 Introduction 43

1.3.2 L'extraction des éléments primitifs 43

1.3.3 Description des relations spatiales 44

1.3.4 Méthodes structurales pour la caractérisation des textures faibles 44

1.3.5 Méthodes structurales pour la caractérisation des textures fortes 45

1.4 Conclusion 47

CHAPITRE II - MÉTHODES DE SEGMENTATION DES TEXTURES

2.1 Introduction 48

2.2 Méthodes basées sur l'extraction des arêtes texturales . . . 49

2.3 Méthodes de l'agrégation dans l'espace des caractéristiques. 54

2.4 Méthodes de croissance des régions 59

2.4.1 Méthode de liaison simple 59

2.4.2 Méthode de liaison hybride 60

2.4.3 Méthode de liaison centroïde 62

2.5 Méthode d'agrégation spatiale 64

2.6 Méthode de division - fusion 66

2.7 Méthodes de segmentation par estimation 67

2.8 Conclusion 77

CHAPITRE III - SEGMENTATION DES TEXTURES PAR ESTIMATION BAYESIENNE

3.1 Théorie du champ markovien à 2-D 79

3.1.1 Introduction 79

3.1.2 Système de voisinage sur une maille 2-D 80

3.1.3 Définition du champ markovien 81

3.1.4 Représentation du champ markovien 2-D 82

3.1.4.1 Modèle de Gibbs 83

3.1.4.2 Modèle gaussien 2-D 86

3.1.4.3 Modèle autobinomial 88

3.2 Application du modèle markovien dans la segmentation des images texturées 89

3.2.1 Introduction 89

3.2.2 Segmentation des images texturées par la programmation dynamique (Derin et Cole) 92

3.2.2.1 Modèle gibbsien hiérarchique 92

3.2.2.2 Procédure de segmentation par programmation dynamique . 94

3.2.2.3 Discussion 97

3.2.3 Segmentation des images texturées par l'algorithme hiérarchique et l'algorithme de relaxation (Cohen et Cooper) 98
3.2.3.1 Modèle de l'image 98
3.2.3.2 Modèle des régions 100
3.2.3.3 Procédure de segmentation hiérarchique 100
3.2.3.4 Procédure de segmentation par relaxation stochastique . 108
3.2.3.5 Discussion 112
3.3 Segmentation non supervisée des images texturées par estimation bayesienne: méthode proposée 113
3.3.1 Introduction 113
3.3.2 Modèle de l'image 115
3.3.2.1 Modèle hiérarchique. 115
3.3.2.2 Modèle des textures. 116
3.3.2.3 Modèle du processus de formation des régions 126
3.3.3 Procédure d'estimation 129
3.3.3.1 Stratégie du maximum de vraisemblance 129
3.3.3.2 Stratégie du maximum à posteriori 131
3.3.4 Estimation des paramètres 132
3.3.4.1 Introduction 132
3.3.4.2 Agrégation floue 134
3.3.4.3 Optimisation par des fonctions objectives. 135

3.3.4.4 Procédure d'agrégation floue hiérarchique dans l'estimation des paramètres 138

3.4 Conclusion 147

CHAPITRE IV - EXPÉRIMENTATION ET RÉSULTATS

4.1 Ressources matérielles et logicielles 148

4.2 Prétraitement 150

4.3 Procédure de segmentation 151

4.3.1 Segmentation avec connaissance à priori 151

4.3.2 Segmentation sans connaissance à priori 158

4.4 Résultats 161

4.5 Discussion 175

4.6 Conclusion 177

CONCLUSION 178

BIBLIOGRAPHIE 180

LISTE DES FIGURES

Figure 1.1 - Illustration de la relation spatiale entre deux entités selon la distance de bloc 21

Figure 1.2 - Des masques de Law 25

Figure 1.3 - Les quatre masques orthogonaux 26

Figure 1.4 - Système de voisinage 31

Figure 1.5 - Emplacement des pixels dans le système de voisinage. 33

Figure 1.6(a) - Système de voisinage ordre 1 et le type de clique correspondant 37

Figure 1.6(b) - Système de voisinage ordre 2 et le type de clique correspondant 38

Figure 2.1 - Opérateur de Robert pour les textures 51

Figure 2.2 - Segmentation par agrégation dans l'espace des caractéristiques 54

Figure 2.3 - Détection de modes dans un histogramme 56

Figure 2.4 - Construction d'une pyramide de deux niveaux 65

Figure 2.5 - Illustration de l'affectation des étiquettes à l'étape 2 74

Figure 2.6 - Illustration d'une division quaternaire à l'étape 3. 75

Figure 3.1 - Systèmes de voisinage 81

Figure 3.2 - Systèmes de voisinage et les types de clique associés 85

Figure 3.3 - Position des pixels selon l'ordre du voisinage . . . 89

Figure 3.4 - Illustration de la division quaternaire 101

Figure 3.5 - Illustration de deux types de blocs 103

Figure 3.6 - Illustration de la division du bloc Y 106

Figure 3.7 - Types de clique du système de voisinage d'ordre 2 . 118

Figure 3.8 - Fonction de densité de $\tau^k{}_{c*}$ 121

Figure 3.9 - Clique 2x2 123

Figure 3.10 - Masques orthogonaux 125

Figure 3.11 - Voisinage d'ordre 2 129

Figure 3.12 - Exemple d'une division hiérarchique 142

Figure 3.13 - Forme de la fonction de distribution F 143

Figure 4.1 - Les textures naturelles 148

Figure 4.2 - Configuration des images bitexturées à segmenter . . 149

Figure 4.3 - Configuration des images tri-texturées à segmenter . 150

Figure 4.4 - Illustration graphique du processus d'égalisation . 150

Figure 4.5 - Procédure de segmentation avec modèles connus à priori 152

Figure 4.6 - Procédure de segmentation sans connaissance à priori 158

Figure 4.7(a) - Illustration de la présentation des résultats d'une segmentation avec supervision 161

Figure 4.7(b) - Illustration de la présentation des résultats d'une segmentation sans supervision 162

Figure 4.8 - Résultat de la segmentation non-supervisée des images composées des textures D84 et D4 égalisées 163

Figure 4.9 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D24 égalisées. 164

Figure 4.10 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D4 égalisées 165

Figure 4.11 - Résultat de la segmentation non-supervisée des images composées des textures D84, D24 et D4 égalisées. . . 166

Figure 4.12 - Résultat de la segmentation non-supervisée des images composées des textures D84, D92 et D4 égalisées. . . 167

Figure 4.13 - Résultat de la segmentation non-supervisée des images composées des textures D29, D4 et D9 égalisées . . . 168

Figure 4.14 - Résultat de la segmentation supervisée et non-supervisée des images composées des textures D84, D4 égalisées 169

Figure 4.15 - Résultat de la segmentation supervisée et non-supervisée des images composées des textures D84, D92 et D4 égalisées 170

Figure 4.16 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D4 avec égalisation et sans égalisation 171

Figure 4.17 - Résultat de la segmentation non-supervisée des images composées des textures D84, D24 et D4 avec égalisation et sans égalisation 172

Figure 4.18 - Résultat de la segmentation non-supervisée des images composées des textures D29, D24 égalisées avec différentes tailles de fenêtre d'analyse . . . 173

Figure 4.19 - Résultat de la segmentation non-supervisée des images composées des textures D29, D4 et D9 égalisées avec différentes tailles de fenêtre d'analyse 174

Figure 4.20 - Illustration de l'influence de la forme des régions sur la taille de la fenêtre d'analyse 175

INTRODUCTION

La segmentation d'une image est un processus de partitionnement de l'image en des régions homogènes. L'homogénéité de chaque région peut être basée sur l'information de niveaux de gris ou des textures. Normalement les régions homogènes correspondent à des entités signifiantes de la scène comme des objets, ou à des faces d'objets délimitées par des discontinuités de niveaux de gris ou de textures.

Formellement, la segmentation d'une image peut être définie comme suit:

- Soit une image I(i,j) définie sur le domaine D:

$$D = \{(i,j) \mid 0 \leq 1 \leq N-1,\ 0 \leq j \leq N-1\}$$

- Soit P un prédicat d'homogénéité, c'est-à-dire une certaine fonction booléenne mesurant l'homogénéité des régions.

$$\forall\ R \subseteq D,\quad P(R) = \begin{cases} \text{vrai si la région est homogène} \\ \text{faux si la région est inhomogène} \end{cases}$$

- Une segmentation de l'image I est une partition de son domaine D en des régions R_1, $R_2 \ldots R_K$ telles que:

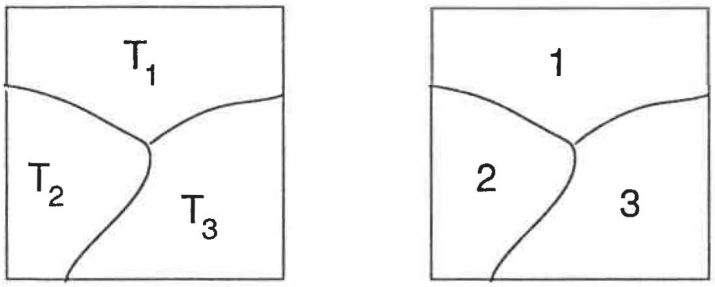
$$D = R_1 \cup R_2 \cup \ldots \cup R_K$$

$$R_m \cap R_n = \phi,\ \forall\ m,n \quad m \neq n$$

$$P(R_i) = \text{vrai}$$

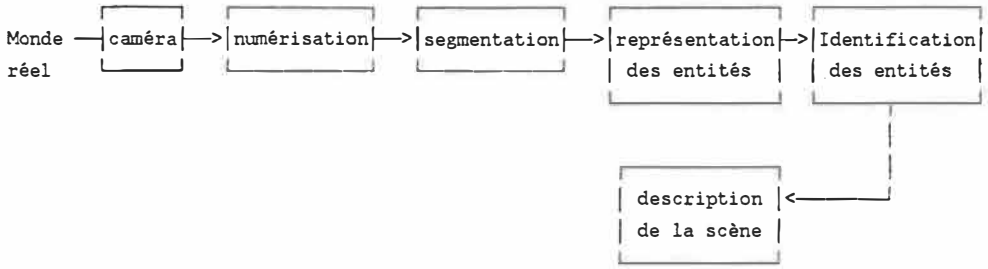
$P(R_m \cup R_n)$ = faux, $\forall m,n$ telles que R_m et R_n sont adjacentes.

L'exemple suivant illustre un processus de segmentation d'une image.



Ti: texture de type i

La segmentation joue un rôle très important dans un système de compréhension d'image. On entend par système de compréhension d'image un système qui prend comme entrée une image numérisée et donne à la sortie une description de la scène présente dans l'image. La figure suivante illustre le schéma-bloc d'un tel système.



Conceptuellement, la segmentation permet de passer une représentation volumineuse de l'information sous forme d'une fonction d'image I(x,y) en une représentation beaucoup plus réduite de

l'information sous forme d'un nombre restreint des entités signifiantes de la scène.

On peut trouver des applications de la segmentation dans divers domaines de la vision artificielle. Le processus de segmentation est requis dans l'isolation des défauts présents dans des produits comme les tissus dans l'industrie de textile, des planches de bois, dans l'industrie forestière, la viande et les filets de poisson dans le secteur alimentaire, les chaussées dans le domaine du transport... L'analyse des images aériennes a besoin aussi de la segmentation de l'image captée en des régions qui identifient des eaux, des forêts, des roches, des régions urbaines.

Ce présent mémoire traite du problème de segmentation par l'information texturale. Ce problème a été étudié par plusieurs chercheurs depuis une vingtaine d'années. Un bon nombre d'articles ont été publiés régulièrement chaque année, mais le problème est loin d'être résolu. La difficulté de la segmentation des textures réside dans les trois aspects suivants:

- une représentation adéquate des textures
- une technique formelle de la segmentation
- un critère objectif de l'évaluation des résultats de segmentation

Haralick [1979] a fait une étude extensive des diverses approches et modèles utilisés par des chercheurs pour représenter

des textures. Ce problème sera révisé en détail dans le chapitre I de ce mémoire. En ce qui concerne les techniques de segmentation, Haralick et Shapiro [1984] nous ont aussi fourni une vue générale de diverses techniques de segmentation et nous le présenterons plus en détail dans le chapitre II de ce mémoire.

Récemment, plusieurs travaux ont été réalisés sur la segmentation des images texturées en appliquant le champ aléatoire markovien dans la modélisation des textures; la segmentation a été effectuée par la méthode d'estimation. Ces nouvelles approches ont été reconnues comme ayant de bons résultats. Parmi ces travaux, on peut citer ceux de Derin et Cole [1986], Cohen et Cooper [1987] et Geman and Geman [1984]. Les travaux de ces auteurs seront révisés dans une section du chapitre III.

Nos travaux sur la segmentation des textures suivent les mêmes approches récemment proposées, c'est-à-dire, une modélisation des textures par le champ aléatoire markovien et la segmentation par la technique d'estimation. Nos contributions au problème de segmentation des textures réside dans deux aspects:

- A partir du modèle markovien des textures, un système de voisinage est défini, ce qui permet, par la transformation de Hadamard, de convertir des observations originales en un ensemble

de caractéristiques locales non-correlées. Cette conversion conduit à la décomposition du problème de segmentation en des décisions stochastiques locales basées sur des fenêtres d'estimation de taille limitée.

- Une procédure d'aggrégation floue hiérarchique réalisée au début du processus de segmentation nous permet d'identifier le nombre de classes de texture dans l'image et d'isoler des régions homogènes. Ces régions sont utilisées pour bâtir le modèle de chaque texture, ce qui rend notre méthode de segmentation complètement non-supervisée.

La méthode est appliquée avec succès à la segmentation d'images composées de textures naturelles.

Comme il a déjà été brièvement mentionné, l'organisation de ce mémoire est comme suit:

- Le Chapitre I est consacré à une bibliographie des principales méthodes pour la caractérisation des textures.

- Le Chapitre II fait un survol des techniques importantes de la segmentation des images, en particulier des images texturées, rapportées dans la littérature.

- Le Chapitre III présente, dans la première partie, l'étude de la technique de segmentation par estimation. La deuxième partie est consacrée à une revue des principaux travaux reliés à cette technique. Nous avons choisi une telle organisation pour le chapitre III afin d'avoir plus de cohérence et plus de facilité dans la comparaison de notre méthode avec les autres.

- Le Chapitre IV présente les résultats de notre méthode de segmentation et des discussions sur les résultats obtenus. Ce chapitre est suivi de la conclusion de notre mémoire.

CHAPITRE I

MÉTHODES DE CARACTÉRISATION DES TEXTURES

I.1 Introduction

La perception des textures est une composante très importante de la vision humaine. Les objets d'une scène peuvent souvent être distingués par leurs caractéristiques texturales, malgré leur similarité de formes et de couleurs. Les textures peuvent aussi fournir des informations sur l'orientation des surfaces. Pour être exploitable, une définition de la notion de texture doit incorporer toutes les caractéristiques importantes qui déterminent la variété des impressions visuelles produites par l'observation des textures.

Les textures peuvent être décrites comme une répétition d'un ou plusieurs motifs de base qui suivent une certaine règle de placement. Le contenu des motifs et la règle de placement peuvent être réguliers comme dans la plupart des textures artificielles, ou varier aléatoirement comme dans les textures naturelles. La plupart des textures apparaissant dans une scène se situent entre ces deux extrêmes: elles contiennent un certain degré de structure bien que présentant des variations aléatoires. Les primitives texturales et leurs interactions spatiales permettent de rendre compte de caractéristiques texturales telles que la granulosité, la finesse, la directivité, etc.

Les méthodes utilisées jusqu'à présent pour la caractérisation des textures sont généralement classifiées en deux catégories : les méthodes statistiques et les méthodes structurales.

L'approche statistique utilise comme caractéristiques les distributions spatiales des niveaux de gris ou de certaines propriétés locales du signal d'image pour décrire les textures, alors que l'approche structurale met l'importance sur la définition des motifs primitifs et des règles de placement qui les gouvernent.

Les sections qui suivent donnent un aperçu des approches statistiques et structurales dans l'analyse des textures et présentent les techniques les plus importantes de chaque approche.

I.2 Approche statistique

I.2.1 Introduction

Dans cette approche, les textures sont décrites à l'aide d'un ensemble des paramètres statistiques extraits à partir des caractéristiques locales de l'image texturée. Ces caractéristiques sont supposées demeurer invariantes pour des textures de même type. Nous allons dans cette section présenter les techniques les plus couramment utilisées. Elles peuvent être classifiées selon les catégories suivantes:

- Les techniques spectrales (Rosenfeld et Troy 1970; Haralick 1979; Levine 1985).

- Les techniques statistiques du premier ordre (Weszka, Dyer et Rosenfeld 1976).

- Les techniques statistiques du deuxième ordre (Julesz 1962; Haralick, Shammugam et Dinstein 1973; Davis, Houns et Aggarwal 1979).

- Les techniques basées sur des statistiques d'ordre plus élevé (Galloway 1975).

- La technique basée sur l'énergie de texture (Davis 1982).

- Et finalement, les techniques basées sur les modèles.

I.2.2 Méthodes basées sur la fréquence spatiale de l'image

Certaines classes de textures présentent un degré élevé de régularité. La fonction d'autocorrélation et la densité spectrale de puissance constituent alors des mesures de cette propriété.

I.2.2.1 Caractérisation des textures par la fonction d'autocorrélation

Levine (1985) présente une illustration de la fonction

d'autocorrélation dans la caractérisation des textures. Il suppose que deux transparents d'une même image sont décalés l'un par rapport à l'autre dans une certaine direction. L'ensemble est placé devant une source de lumière uniforme et l'énergie transmise est enregistrée. Cette variable, dessinée en fonction du déplacement bidimensionnel constitue la fonction d'autocorrélation, dont la forme fournit des informations sur la taille et sur la distance relative des éléments de base qui constituent la texture.

Mathématiquement, la fonction d'autocorrélation bidimensionnelle est donnée par :

$$Y(\Delta x,\Delta y) = \frac{\sum_{i,j} I(i,j)\ I(i+\Delta x, j+\Delta y)}{\sum_{i,j}[I(i,j)]} \qquad (1.1)$$

où I(i,j) est la luminance du pixel (i,j).

La granulosité des textures est spécifiée par la fréquence spatiale de Y(Δx,Δy): si les éléments de base sont petits, la fréquence est élevée et vice versa.

La régularité de la forme de la courbe fournit l'information de la régularité de la texture dans la direction considérée.

Haralick (1979) utilise les images aériennes dans une étude comparative de la granulosité des textures, telle perçue par des individus et calculée par la fonction d'autocorrélation. Sa mesure est le nombre de pixel par lequel l'image devrait être décalée pour que la

fonction d'autocorrélation tombe à 1/e de la valeur maximale. Ses études ont montré un degré élevé de concordance entre les mesures subjectives et les mesures calculées.

De leur côté, Rosenfeld et Troy [7] ont montré que la fonction d'autocorrélation n'est pas une bonne mesure de granulosité pour les textures naturelles.

I.2.2.2 Caractérisation des textures par la transformation de Fourier

Une autre façon d'examiner le contenu fréquentiel de l'image est d'utiliser la transformation de Fourier bidimensionnelle.

La transformation de Fourier de l'image I(i,j) est exprimée par:

$$J(u,v) = \frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j) \exp\left[- \frac{2\pi k}{N} (iu+jv) \right] \qquad (1.2)$$

où $u = 0, 1, 2, .., N-1$

$v = 0, 1, 2, .., N-1$

Le spectre de puissance de I(i,j) est donné par :

$$P(u,v) = \sqrt{(\text{Ré}[J(u,v)])^2 + (\text{Im}[J(u,v)])^2} \qquad (1.3)$$

La fonction bidimensionnelle P(u,v) caractérise la granulosité des textures. Une texture granuleuse donne une valeur élevée de P(u,v) autour de l'origine parce que les composantes de basses fréquences sont associées à des éléments de plus grande taille. Par contre, si la

distribution P(u,v) s'étend loin de l'origine, ceci signifie que la contribution des composantes de hautes fréquences est non négligeable, on est en présence d'une texture fine. L'information de directivité des textures est aussi fournie par P(u,v). En effet, si les motifs des textures sont plutôt orientés dans une direction particulière, des valeurs élevées de P(u,v) tendent alors à se concentrer dans une direction qui est perpendiculaire à celle des motifs. Par contre, si la texture est non directionnelle ou aléatoire, on ne s'attend pas à avoir une tendance directionnelle de la fonction P(u,v).

Les travaux les plus importants concernant l'utilisation des caractéristiques texturales dans le domaine fréquentiel sont rapportés dans (Weszka, Dyer et Rosenfield 1976). En générale, les descripteurs de Fourier ne sont pas assez puissants pour la discrimination des textures naturelles.

I.2.3 Méthodes basées sur les statistiques du premier ordre

Les caractéristiques texturales peuvent être capturées par les statistiques de premier ordre du niveau de gris ou des propriétés locales. Ces statistiques s'avèrent utiles dans certaines applications.

I.2.3.1 La distribution de premier ordre de luminance

L'histogramme de luminance d'une image est un résumé simple et concis de l'information contenue dans celle-ci. A partir de

l'histogramme on peut extraire des mesures appelées des moments centraux (Levine 1985).

Soit H(i), i=1, 2,.. K. L'histogramme normalisé des niveaux de gris ou K est le nombre d'intervalles qui couvrent la plage dynamique des luminances. On définit les mesures suivantes :

$$\text{La moyenne} \quad : \mu = \sum_{i=1}^{K} iH(i) \tag{1.4}$$

$$\text{La variance} \quad : \sigma_2 = \sum_{i=1}^{K} (i-\mu)^2 H(i) \tag{1.5}$$

$$\text{L'asymétrie} \quad : \mu_3 = \frac{1}{\sigma^3} \sum_{i=1}^{K} (i-\mu)^3 H(i) \tag{1.6}$$

$$\text{L'applatissement:} \quad \mu_4 = \frac{1}{4} \sum_{i=1}^{K} (i-\mu)^4 H(i) - 3 \tag{1.7}$$

La mesure de la variance indique l'uniformité de luminance de l'image. La dissymétrie donne une indication de la forme de la courbe en faveur d'une côté ou de l'autre de la moyenne. L'applatissement caractérise le degré d'acuité du sommet, en supposant que l'applatissement du gaussien est égal à 0.

La caractérisation des textures par les statistiques d'ordre un relatives aux niveaux de gris n'est pas efficace en général parce qu'elle dépend des conditions d'éclairage et que le résultat change d'une image à l'autre même lorsqu'on traite les mêmes textures.

I.2.3.2 Méthodes basées sur la différence de niveaux de gris (GLDM)

Au lieu d'utiliser les statistiques de premier ordre de luminance, on peut caractériser les textures par les statistiques extraites de l'histogramme des propriétés locales des niveaux de gris. Dans cette partie, on va considérer une classe particulière de propriétés locales basées sur la différence absolue entre les niveaux de gris d'une paire de pixels (Weszka, Dyer et Rosenfeld 1976)

Pour un vecteur de déplacement donné :

$$d = (\Delta x, \Delta y)$$

On définit la différence absolue entre deux pixels situés à une distance d l'un de l'autre, comme suit :

$$I'(x,y) = | I(x,y) - I(x+\Delta x, y+\Delta y) | \qquad (1.8)$$

Soit P, la distribution de la valeur de I'. S'il y a Ng niveaux de gris dans l'image originale I, cette distribution prend la forme d'un vecteur de Ng dimensions dont la $i^{ème}$ composante représente la probabilité que I' soit égale à i.

Si la texture est grossière et d est faible par rapport à la taille de l'élément primitif, les paires de point à une distance d ont, dans la plupart des cas, des niveaux de gris similaires, ce qui fait que les valeurs significatives de P seront essentiellement concentrées

près de l'origine. Inversement, pour une texture fine, avec d comparable à la taille des éléments, les niveaux de gris de deux pixels à une distance d seront en général différents, les valeurs significatives de P(i) étant réparties de façon plus étendue. Donc, l'observation de la forme de l'histogramme P(i) fournit une indication de la finesse des textures considérées. Weszka, Dyer et Rosenfeld (1976) proposent 5 mesures des caractéristiques texturales associées à cette méthode :

1/ Le contraste :

$$CONT = \sum_{i=1}^{Ng} i^2 P(i) \qquad (1.9)$$

C'est le deuxième moment de P, c'est-à-dire son moment d'inertie par rapport à l'origine. Plus la texture est fine, plus la valeur de CONT est élevée.

2/ Deuxième moment angulaire :

$$ASM = \sum_{i=1}^{Ng} P^2(i) \qquad (1.10)$$

Cette quantité est minimum si les composantes de P(i) sont toutes égales et élevée si les valeurs des composantes de P(i) sont irrégulières.

3/ Entropie :

$$ENT = - \sum_{i=1}^{Ng} [P(i)\mathrm{Log}P(i)] \qquad (1.11)$$

Cette mesure est encore élevée quand les P(i) sont répartis de façon plus uniforme.

4/ La moyenne :

$$MEAN = \frac{1}{Ng} \sum_{i=1}^{Ng} iP(i) \qquad (1.12)$$

Cette valeur est faible lorsque les valeurs élevées de P(i) sont près de l'origine et élevée quand les valeurs de P(i) sont plus ou moins irrégulières.

I.2.4 Méthodes basées sur les statistiques du deuxième ordre

Les méthodes basées sur les statistiques de premier ordre, dont on a discuté précédemment mesurent les propriétés ponctuelles des textures. Les statistiques de deuxième ordre tiennent compte aussi de la relation spatiale des motifs texturaux et donnent certainement une représentation plus riche de la texture.

Dans les études sur la vision humaine, Julesz (1962) est la première personne à avoir introduit et justifié les statistiques du deuxième ordre comme indicateurs de textures. D'après Julesz, l'homme ne peut discriminer les textures qui sont différentes par les statistiques de troisième ordre ou plus. Il a supposé que le système de vision humaine calcule les attributs texturaux en se basant sur les statistiques de deuxième ordre. Cependant Gagalowicz (1979) a montré des arguments contre l'affirmation de Julesz. En effet, il a donné plusieurs couples de textures composées de primitives discernables, où les statistiques du deuxième ordre prises localement diffèrent, alors que celles prises globalement sont identiques.

Gagalowicz (1979) conclut que l'oeil humain est incapable de différencier deux textures lorsque celles-ci présentent localement les mêmes statistiques du premier et du deuxième ordre. Ces études permettent de définir des limites dans l'analyse des textures.

Les matrices de dépendance spatiale des niveaux de gris (SGLDM) (les matrices de cooccurence) ont été introduites pour la première fois par Haralick et Dinstein (1972). Elles sont basées sur l'estimation des distributions conjointes f(i,j,d). Chaque f(i,j,d) mesure la probabilité que deux pixels situés à une distance relative d aient respectivement le niveau de gris i et le niveau de gris j. Si une image a Ng niveaux de gris, ces distributions peuvent être exprimées sous la forme des matrices de NgxNg. Chaque matrice est associée à un vecteur de distance d. La valeur de l'élément (i,j) d'une matrice est calculée en comptant le nombre de paires de pixels [(a,b),(c,d)], situées à une distance relative d, telles que I(a,b) = i, I(c,d) = j.

L'exemple suivant illustre le calcul d'une telle matrice.

Image :

0	0	1	1
0	0	1	1
0	2	2	2
2	2	3	3

Cette image a 4 niveaux de gris 0, 1, 2, 3.

La matrice de cooccurence :

	0	1	2	3
0	4	2	1	0
1	2	4	0	0
2	1	0	6	1
3	0	0	1	2

$P(i,j|d=(1,0))$

Soit R la constante de normalisation

$$f(i,j,d=(1,0)) = \frac{1}{R} P(i,j|d(1,0)) \qquad (1.13)$$

Quatre vecteurs de déplacement d sont souvent utilisés :

$$D = \{(1,0), (1,1), (0,1), (-1,1)\}$$

L'ensemble des matrices de cooccurence contient en principe toute l'information permettant de caractériser une texture. Haralick , Shanmugam et Dinstein (1973) ont proposé un ensemble de mesures qui peuvent être utilisées pour extraire l'information texturale à partir des matrices de cooccurence. Voici quelques mesures importantes:

1/ Energie :

$$E = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} [f(i,j,d)]^2 \qquad (1.14)$$

Cette mesure indique le degré d'homogénéité d'une image. Dans une image homogène, il y a très peu de transitions de niveaux de gris, par conséquent, les pixels de l'image I sont concentrés sur un nombre restreint des éléments de la matrice, ce qui donne une énergie élévée.

Pour une image non homogène, le même nombre de pixels est disséminé sur un nombre plus grand d'éléments de la matrice; par conséquent, une faible valeur d'énergie sera obtenue.

2/ Entropie :

$$ENT = - \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} [f(i,j,d) \; Log(f(i,j,d))] \qquad (1.15)$$

L'entropie donne une mesure de la complexité de l'image. Une image complexe aura une valeur d'entropie plus élévée qu'une image simple.

3/ L'inertie :

$$I = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} [(i-j)^2 f(i,j,d)] \qquad (1.16)$$

L'inertie est une mesure des variations locales dans l'image. La valeur élévée de cette mesure indique une variation locale importante.

4/ L'homogénéité locale :

$$LH = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} [f(i,j,d)/(1-(1+j)^2] \qquad (1.17)$$

Une valeur élévée de cette mesure indique la proximité géométrique des niveaux de gris semblables.

5/ Corrélation :

$$COR = [\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (ijf(i,j) - \mu_x \mu_y)]/\sigma_x \sigma_y \qquad (1.18)$$

où μ_x, σ_x représentent la moyenne et l'écart-type de la somme des lignes

et μ_y, σ_y représentent les mêmes éléments pour les colonnes.

C'est une mesure de la ressemblance des lignes (ou des colonnes) de la matrice. La mesure est élévée quand les valeurs sont distribuées uniformément dans la matrice et vice versa.

Jusqu'à présent, les statistiques du deuxième ordre sont calculées en considérant les pixels comme éléments texturaux. Les caractéristiques utilisées pour ces éléments sont la position du pixel et sa luminance. Ceci peut être généralisé à d'autres éléments ce qui donne une matrice de cooccurence généralisée (GCM) (Davis, Johns et Aggarwal 1979). En effet, au lieu de prendre les pixels comme éléments texturaux, on peut sélectionner des entités comme points du contour, ensembles des points connexes du contour ou régions uniformes. Le vecteur de caractéristiques associé à ces primitives texturales peut avoir un certain nombre de composantes. Par exemple, un élément du contour peut être caractérisé par les indicateurs suivants : position, contraste, orientation. En plus, la relation spatiale des entités peut être plus générale, par exemple le prédicat de contrainte spatiale peut être défini par deux éléments dont la distance de bloc est plus petite ou égale à 2. La figure 1.1 montre deux éléments en noir (parmi d'autres) qui satisfont la contrainte spatiale avec l'élément hachuré.

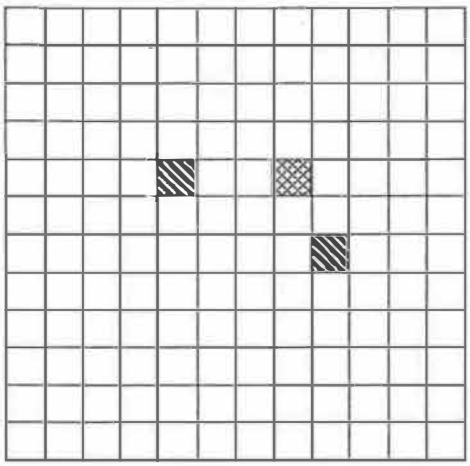


Figure 1.1 - Illustration de la relation spatiale entre deux entités selon la distance de bloc.

Une matrice de cooccurence généralisée est calculée, pour chaque composante du vecteur caractéristique, avec une relation spatiale donnée.

Des études théoriques et expérimentales (Connors, Harlow 1980) montrent que les mesures basées sur les statistiques du deuxième ordre sont très efficaces pour discriminer les textures naturelles.

Unser (1986) a proposé une alternative à la méthode des matrices de cooccurence, dans laquelle la somme et la différence de deux variables aléatoires de même variance sont décorrelées et définissent les axes principaux de leurs fonctions de probabilité conjointe

associées. Les histogrammes de somme et de différence sont construits pour analyser des textures. Il a aussi démontré l'équivalence théorique de ces deux méthodes et la supériorité de sa méthode au niveau de l'espace de mémoire requis et du temps de calcul et de la performance.

I.2.5 Méthode basée sur les matrices de longueur de suites de niveaux de gris (GLRLM)

Dans une image, des suites linéaires de pixels adjacents ayant le même niveau de gris peuvent se produire dans une direction donnée. Dans une texture granuleuse, on s'attend à avoir un grand nombre de longues suites. Par contre, les textures fines contiennent une proportion élevée des suites courtes.

Soit P(i,j), le nombre des suites de longueur j, de niveau de gris (i ± Δg), obtenu dans une certaine direction. Δg est un paramètre qui détermine l'homogénéité relative (Galloway 1975).

Les caractéristiques suivantes sont définies :

1/ Tendance de longues suites (Long Run Emphasis)

$$LRE = \sum_i \sum_j [j^2 P(i,j)] / \sum_i \sum_j P(i,j) \qquad (1.19)$$

Les textures ayant des primitives de grand taille auront une valeur élevée de LRE et vice versa.

2/ Tendance de suites courtes (Short Run Emphasis)

$$SRE = \sum_i \sum_j [P(i,j)/j^2] / \sum_i \sum_j P(i,j) \qquad (1.20)$$

3/ Non-uniformité de niveaux de gris (Gray Level Nonuniformity)

$$GLNU = \sum_i [\sum_j P(i,j)]^2 / \sum_i \sum_j P(i,j) \qquad (1.21)$$

Cette mesure est petite si le nombre de suites de chaque niveau de gris est distribué uniformément.

4/ Non-uniformité de longueurs de suites (Run Length Nonuniformity)

$$RLNU = \sum_j [\sum_i P(i,j)]^2 / \sum_i \sum_j P(i,j) \qquad (1.22)$$

Cette mesure est petite si la longueur des suites est distribuée uniformément.

5/ Pourcentage des suites (Run Percentage)

$$RPC = \sum_i \sum_j P(i,j)/N^2 \qquad (1.23)$$

où N^2 est le nombre de pixels dans l'image. Cette mesure est élevée si les suites sont courtes.

I.2.6 Méthode basée sur l'énergie de texture

Law Davis (1982) a proposé un ensemble de mesures appelées mesures de l'énergie des textures. Ces mesures sont dérivées de 3 vecteurs de base de longueur 3 :

L3 = (1,2,1)

E3 = (-1,0,1)

S3 = (-1,2,-1)

L3 représente une moyenne locale, E3 correspond à une détection de contour et S3 correspond à une détection de point. La convolution croisée de ces vecteurs nous donne 5 vecteurs de taille 5:

L5 = (1,4,6,4,1) = L3 * L3

S5 = (-1,0,2,0,-1) = E3 * E3 = L3 * S3

R5 = (1,-4,6,-4,1) = S3 * S3

E5 = (-1,-2,0,2,1) = L3 * E3

W5 = (-1,2,0,-2,1) = E3 * S3

Ici L5 représente la moyenne locale, E5 et S5 sont des détecteurs de point, K5 et W5 sont considérés respectivemeent comme détecteur de vague (wave detector) et détecteur d'ondulation (ripple detector).

Un ensemble de masques de deux dimensions est obtenu en convoluant un vecteur de longueur 3 (ou 5) avec le transposé d'un autre vecteur de longueur 3 (ou 5). Neuf masques de 3x3 sont ainsi obtenus :

$L3^t*L3$

1	2	1
2	4	2
1	2	1

$L3^t*E3$

-1	0	1
-2	0	2
-1	0	1

$L3^t*S3$

-1	2	-1
-2	4	-2
-1	2	-1

$E3^t*L3$

-1	-2	-1
0	0	0
1	2	1

$E3^t*E3$

1	-2	1
0	0	0
-1	0	1

$E3^t*R3$

1	-2	1
0	0	0
-1	2	-1

$S3^t*L3$

-1	-2	-1
2	4	2
-1	-2	-1

$S3^t*E3$

1	0	-1
-2	0	2
1	0	-1

$S3^t*S3$

1	-2	1
-2	4	-2
1	-2	1

Figure 1.2 - Des masques de Law.

La même opération pour les vecteurs de longueur 5 nous donne aussi un ensemble de masques 5x5.

Law a démontré que l'ensemble de masque 3x3 (ou 5x5) représente une base de l'espace des voisinages 3x3, c'est-à-dire que n'importe quel masque est égal à une combinaison linéaire de ces masques.

Pour utiliser ces masques dans l'analyse des textures, on convolue un sous-ensemble de ces masques à l'intérieur d'une fenêtre assez grande (15x15 ou plus). L'information texturale de la fenêtre est décrite par les statistiques des réponses des masques à la région contenue dans la fenêtre. Law suggère la somme des carrés des réponses ou la sommation des valeurs absolues des réponses comme caractéristiques des textures, d'où vient la notion de l'énergie de texture. Il a démontré que ces masques donnent un très bon résultat en classification des micro-textures.

I.2.7 Méthode basée sur des masques orthogonaux

Cohen, Le Dinh et Lacasse (1987) ont proposé une approche dans laquelle chaque voisinage local est transformé réversiblement, par la transformation de Hadamard, en des variables décorrelées.

Si la taille du voisinage est de 2x2, les nouvelles variables sont générées par la convolution de 4 masques orthogonaux (fig. 1.3) dans l'image :

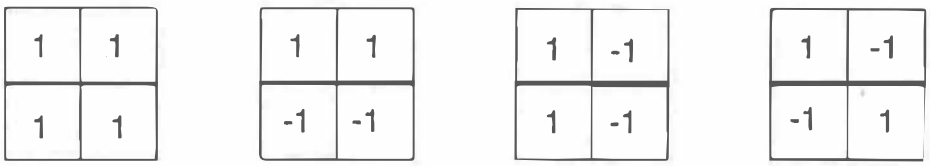


Figure 1.3 - Les quatre masques orthogonaux.

Les statistiques dérivées des 4 images intermédiaires sont utilisées pour la classification. Pour chaque réponse locale h_i, i=1, 2, ... 4, un histogramme correspondant est calculé sur l'image. Les mesures suivantes, dérivées de ces 4 histogrammes, sont proposées pour caractériser une texture :

1/ Dérivation absolue :

$$ABS(h_i) = \left[\frac{1}{N^2} \sum_{k=1}^{K_i} |k-\overline{h_i}| \; p_i(k) \right] \qquad (1.24)$$

d'où K_i est le nombre d'intervalles de l'histogramme de h_i et $\overline{h_i}$ représente la valeur moyenne de h_i.

2/ Ecart-type :

$$STD(h_i) = \sqrt{\frac{1}{N^2} \sum_{k=1}^{K_i} (k-\overline{h_i})^2 p_i(k)} \qquad (1.25)$$

3/ Asymétrie :

$$SKN(h_i) = \left[\frac{1}{N^2} \frac{1}{(STD)^{3/2}} \sum_{k=1}^{K_i} (k-\overline{h_i})^3 p_i(k) \right]^{1/3} \qquad (1.26)$$

Cette méthode donne une très bonne performance dans la discrimination des micro-textures naturelles.

I.2.8 Méthodes basées sur les modèles

La caractéristique commune des approches mentionnées ci-dessus est de chercher de façon ad hoc des mesures qui pourraient caractériser les textures. De ce fait, les mesures obtenues sont souvent imparfaites et ne peuvent s'appliquer avec le même degré de succès à tous les types de textures. Ceci provient du fait que la plupart des méthodes ne sont pas bâties sur des fondements mathématiques ou des modèles rigoureux.

Dans les dernières années, certains efforts de recherche ont été orientés vers la description des textures par des modèles mathématiques. Ces modèles sont souvent sous forme d'équations stochastiques qui définissent la dépendance du niveau de gris à un pixel de son voisinage. Dans cette approche, l'information texturale est obtenue en analysant mathématiquement la structure du modèle.

Plusieurs modèles ont été proposés jusqu'ici :

- Modèles autorégressifs (Souza 1982; Kashyap, Chellapa et Khotanzad 1982)
- Modèles mosaïque (Schacter 1978).
- Modèles markoviens ou son équivalence, le modèle de Gibbs (Cross et Jain 1983).
- Modèles linéaires (Deguchi et Mokishita 1978).
- Modèles de morphologie mathématique (Serra 1980).

Dans cette section, on présentera seulement les modèles markoviens et les modèles autorégressifs. En effet, ces modèles ont obtenu des succès remarquables récemment dans la caractérisation et la segmentation des textures.

I.2.8.1 Modèle markovien pour la caractérisation des textures

Les champs aléatoires markoviens ont été introduits à l'origine

par Ising (1925) dans ses études du phénomène ferromagnétique. Bien que ce modèle n'ait pas pu être appliqué avec succès dans le domaine ferromagnétique, il a donné de bons résultats pour des gaz naturels et des cristaux. Initialement, le modèle a été développé dans le cas de variables binaires ou gaussiennes sur la maille. Besag (1974) en a fait l'extension au cas où les variables ont des valeurs entières. Cette extension et l'apparition des procédures d'estimation des paramètres du modèle ont finalement permis l'application du champ aléatoire markovien dans la modélisation des textures.

a) Notions de base

Les définitions suivantes sont extraites de l'article de Besag (Besag 1974). Soit X(i,j), la luminance au point (i,j) dans une image de taille NxN. Pour simplifier, la notation X(i,j) est remplacée par X(i), i=1, 2... M où $M = N^2$.

Définition 1:

Soit L une maille bi-dimensionnelle. Une image peut être représentée par une fonction de deux variables X qui a comme domaine les points de L, et a comme valeurs des éléments dans l'ensemble {0, 1, 2, .., G-1} où G est le nombre total de niveaux de gris.

Définition 2

Le point j est un voisin du point i si :

$$P(X(i)|X(1), X(2), .., X(M)) \text{ dépend de } X(j).$$

Cette définition n'implique pas la proximité géométrique des pixels voisins par rapport au pixel courant.

Définition 3

Un champ aléatoire markovien est une fonction de densité de probabilité sur l'ensemble de toutes les réalisations possibles de X soumises aux conditions suivantes :

1/ Positivité :

$$p(X) > 0 \text{ pour tout } X$$

2/ Markovianité :

$$p(X(i)|\text{tout point sur la maille L}) = p(X(i)|\text{les voisins de i})$$

3/ Homogénéité :

$p(X(i)|$voisins de i) dépend seulement de la configuration des voisins et est invariant par rapport à la translation.

Définition 4

Soit un système de voisinage de configuration suivante :

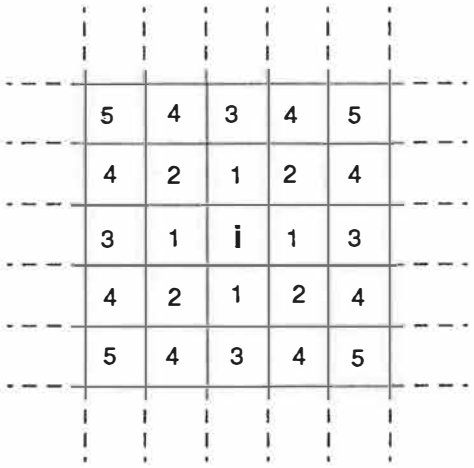


Figure 1.4 - Système de voisinage.

L'ordre d'un processus aléatoire markovien est $n^{ième}$ si p(X(i)/tout point sur la maille L) dépend seulement des voisins portant l'étiquette n du système de voisinage défini ci-dessus.

b) Champ aléatoire markovien autobinomial

Cross et Jain (1983) utilisent la distribution autobinomiale pour exprimer la dépendance probabiliste des variables aléatoires. Selon ce modèle, la probabilité p(X(i)|voisins de i) est binomiale avec le paramètre $\theta(T)$ ci-dessous et un nombre G-1 de tirages où G représente le nombre de niveaux de gris :

$$\theta(T) = \frac{\exp(T)}{1 + \exp(T)}$$

L'expression de T dépend de l'ordre du processus :

Premier ordre : $T = a + b(1,1)(t+t') + b(1,2)(u+u')$ (1.27)

(1.28)

Deuxième ordre : $T = a + b(1,1)(t+t') + b(1,2)(u+u') + b(2,1)(v+v') + b(2,2)(z+z')$

(1.29)

Troisième ordre: $T = a + b(1,1)(t+t') + b(1,2)(u+u') + b(2,1)(v+v') + b(2,2)(z+z') + b(3,1)(m+m') + b(3,2)(l+l')$

où :

a, b(p,q) sont des paramètres du modèle

t, t', u, u', v, v', z, z', m, m', l, l' sont les luminances des pixels dont les positions sont montrées dans la figure 1.5.

L'expression de T pour l'ordre plus élevé est obtenue d'une façon identique.

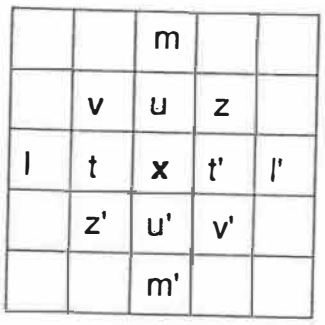


Figure 1.5 - Emplacement des pixels dans le système de voisinage.

En se basant sur cette représentation, Cross et Jain synthétisent des textures en faisant varier les paramètres du modèle. Par ce moyen, ils étudient l'influence des paramètres sur les caractéristiques de regroupement et d'anisotropie des textures.

Ils ont également étudié les correspondances entre des textures naturelles et celles synthétisées par le modèle : d'abord une procédure d'estimation des paramètres du modèle est faite sur les textures naturelles. La technique d'estimation est le codage (Cross et Jain 1983). A partir des paramètres obtenus, on synthétise la texture, et la performance est jugée visuellement en comparant la texture synthétique avec la texture naturelle. Les résultats obtenus avec le modèle autobinomial sont satisfaisants des points de vue suivants :

- Les paramètres du modèle sont suffisants pour synthétiser des images de textures (les approches basées sur la caractérisation statique n'ont pas la capacité de regénérer des images parce qu'elles ne définissent pas uniquement une texture)

- L'image texturée synthétisée est réaliste. Le modèle permet une transition naturelle des niveaux de gris.

- La directivité et la finesse peuvent être contrôlées par le choix des paramètres.

c) Champ aléatoire markovien gaussien

Le champ aléatoire markovien est caractérisé par la fonction de densité de la forme (Chellappa 1986):

$$p(y(x) \mid \text{tout } y(x+r),\ r \epsilon N)$$

d'où N est un ensemble symétrique des voisins sans compter x.

Dans le cas particulier où y(x) est gaussien, le modèle markovien est exprimé par une équation de différence qui détermine l'état du champ au point x de la maille L en fonction des états des proches voisins situés à une distance inférieure ou égale à D du point x, où D exprime l'ordre du processus.

La notion d'ordre du processus est semblable à celle que l'on a présentée au paragraphe précédent.

* Représentation du modèle

Supposons que $y(x)$, $x\epsilon\Omega$, $\Omega = \{x = (i,j) \mid 0 \leq i, j \leq N-1\}$ est gaussien de moyenne nulle et exprimé par l'équation de différence de la forme :

$$y(x) = \sum_{r\epsilon N_x} \theta_r [y(x+r) + y(x-r)] + e(x) \tag{1.30}$$

où : N et N_x sont reliés comme suit:

$$N = \{x \mid x\epsilon N_x\} \cup \{-x \mid x\epsilon N_x\} \tag{1.31}$$

$e(x)$ est un bruit gaussien stationnaire de moyenne nulle qui a les propriétés suivantes :

$$E[e(x), e(r)] = \begin{cases} 0 \text{ si } r \neq x \\ \upsilon \text{ si } r = x \end{cases} \tag{1.32}$$

La propriété $E[e(x) \mid \text{tout } y(r),\ r \neq x] = 0$ implique:

$$p(y(x) \mid \text{tout } y(r),\ r \neq x) = p(y(x) \mid \text{tout } y(x+r),\ r\epsilon N)$$

Cette équation exprime la propriété markovienne du champ aléatoire par rapport au système de voisinage N.

* Caractérisation des textures par un champ aléatoire markovien gaussien (GMRF)

L'ensemble des paramètres $\theta_r = \{\theta_r, r \epsilon N_s\}$ ainsi que υ constituent une représentation des différentes textures.

Chellapa et Chatterjee (1985) utilisent ce modèle dans la synthétisation et la classification des textures naturelles. La régénération des textures est jugée bonne visuellement et le taux de classification correcte atteint plus de 90%

Kaneko et Yodogawa (1982) utilisent le même modèle et obtiennent une performance de 100% dans la classification. La différence entre ces deux travaux est que Kaneko et Yodogawa ont utilisé un classificateur basé sur la distance de Bhattacharyya, alors que le classificateur de Chellapa et Chatterjee est basé sur la distance minimale, utilisant la moyenne et la variance des paramètres comme vecteur de caractéristiques pour chaque texture. De plus, Kaneko et Yodogawa ont optimisé le choix de voisinage, c'est-à-dire le nombre des paramètres, par la méthode AIC (Akaikes information criterion) pour évaluer le degré de justesse de chaque type de voisinage par rapport au modèle.

d) Champ aléatoire gibbsien (GRF)

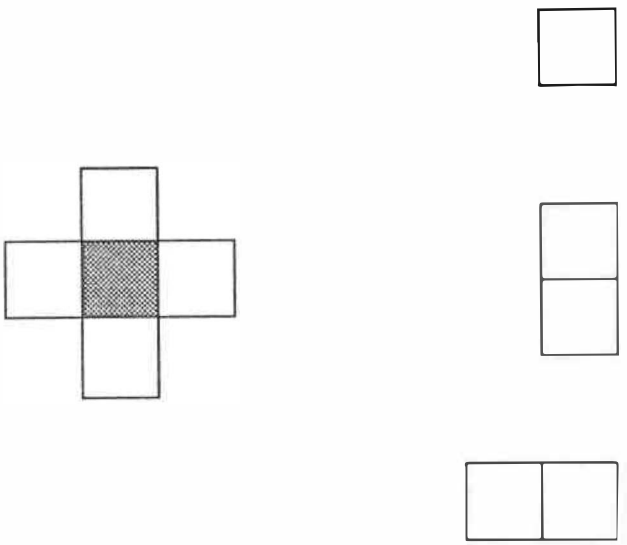
Pour définir un champ aléatoire gaussien (GRF), il est nécessaire

d'introduire la notion de clique associée à un système de voisinage d'ordre n.

Une clique c dans un système de voisinage d'ordre n est un sous ensemble de la maille L, tel que :

- c contient seulement un pixel du voisinage.
- deux pixels (i,j) et (k,l) appartiennent à une clique c si (i,j) est un voisin d'ordre n de (k,l).

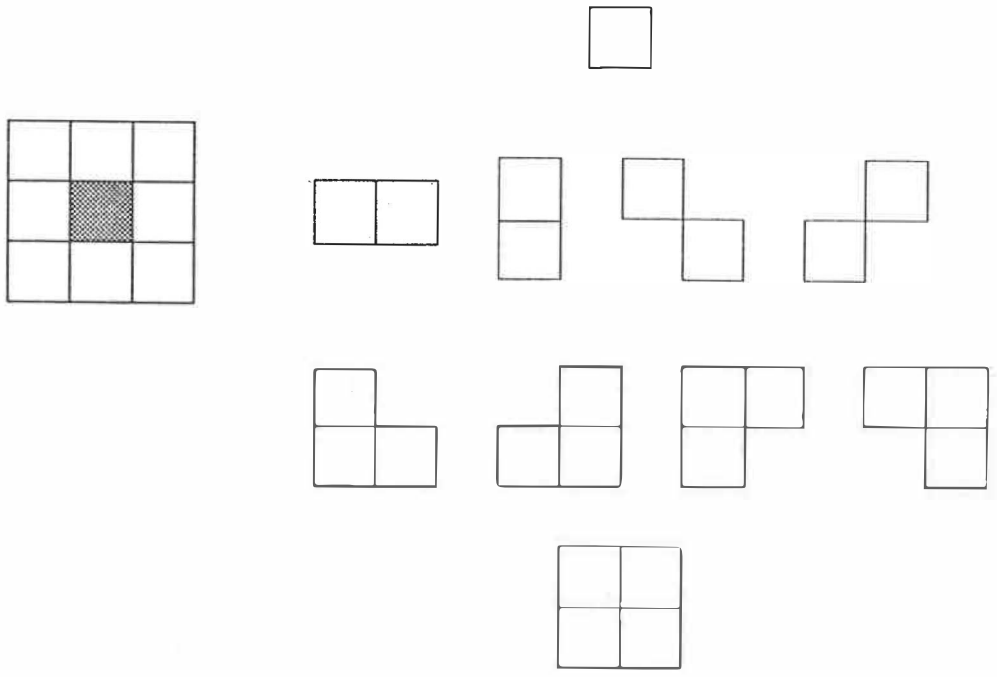
Les figures 1.6 a) et b) illustrent les différents types de clique associées à différents voisinages.



Système de voisinage ordre 1

Type de clique

Figure 1.6 (a) - Système de voisinage ordre 1 et le type de clique correspondant.



Système de voisinage ordre 2 — Type de clique

Figure 1.6 (b) - Système de voisinage ordre 2 et le type de clique correspondant.

Définition:

Soit L la maille et η^i un système de voisinage d'ordre i. Un champ aléatoire $X = \{X_{ij}\}$ défini sur L, suit une distribution de Gibbs par rapport à η^i, si et seulement si sa distribution conjointe est de la forme:

$$p(X=x) = \frac{1}{Z} e^{-U(x)} \tag{1.33}$$

où $U(x) = \sum_{c \in C} V_c(x)$ (1.34)

$V_c(x)$ est une fonction d'énergie associée à la clique c

C est l'ensemble de toutes les cliques.

Il existe une correspondance bi-univoque entre le champ aléatoire markovien (MRF) et le champ aléatoire gibbsien (GRF), établie par le théorème de Hammersley-Clifford.

Théorème Hammersley-Clifford:

Soit η^i un système de voisinage d'ordre i sur la maille L. Un champ aléatoire X est markovien par rapport à η^i si et seulement si sa distribution conjointe est une distribution de Gibbs avec les cliques associées à η^i.

Derin et Cole (1986) utilisent une classe particulière de la distribution de Gibbs. Pour un système de voisinage d'ordre 2, l'énergie potentielle associée aux cliques est définie comme suit :

$$V_c(x) = \begin{cases} -\theta \text{ si tous } x_{ij} \text{ dans c sont égaux} \\ \theta \text{ autrement} \end{cases} \qquad (1.35)$$

Les valeurs de θ sont différentes pour les différents types de clique

β_1 β_2 β_3 β_4 γ_1

γ_2 γ_3 γ_4 ζ_1

Pour les cliques d'un pixel :

$$V_c(x) = \alpha_k \text{ pour } x_{ij} = r_k \tag{1.36}$$

où r_k est le niveau de gris correspondant au $k^{ième}$ intervalle

En utilisant ce modèle, l'auteur a vérifié le rôle de chaque paramètre dans le contrôle de tailles et de directions du regroupement. Aucun essai n'a été fait sur la régénération et la classification des textures.

I.2.8.2 Modèles autorégressifs pour les textures

Kashyap, Chaellappa et Khotanzad (1982) utilisent le modèle SAR (Simultanous Autoregressive) pour la classification des textures naturelles.

* Représentation du modèle SAR

Soit $(y'(x), x = (i,j) \in \Omega)$ où $\Omega = ((i,j) \mid 0 < i, j < M-1)$ une image avec la moyenne α et la variance σ^2. $(y(x), x\epsilon\Omega)$ est une image avec la moyenne nulle et la même variance

$$y(x) = y'(x) - \alpha \tag{1.37}$$

$$\alpha = \frac{1}{M^2} \sum_{x\epsilon\Omega} y'(x) \tag{1.38}$$

$$\sigma^2 = \frac{1}{(M^2-1)} \sum_{x\epsilon\Omega} (y'(x)-\alpha)^2 \qquad (1.39)$$

Supposons que {y(x)} suit un modèle SAR défini par:

$$y(x) = \sum_{r\epsilon N} \theta_r y(x-r) = \sqrt{\rho\ W(x)} \qquad (1.40)$$

où N est l'ensemble de voisins et n'est pas nécessairement symétrique, W(x) est un bruit blanc gaussien de moyenne nulle et de variance égale à 1, l'ensemble de $\{\theta_r, r \epsilon N\}$ et ρ sont des paramètres du modèle.

La classification est réalisée en utilisant un classificateur bayesien, en supposant que les paramètres sont distribués selon une loi multinormale.

Le taux de classification correcte est de 99% pour 9 textures naturelles formant 72 échantillons de taille 64x64

Un autre type de modèle autorégressif appelé modèle autorégressif saisonnier (Seasonal Autoregressive Model), a été étudié par Souza (1982). Il utilise aussi ce type de modèle pour la classification des textures.

I.2.9 Conclusion

Les résultats obtenus par les différentes méthodes sont

difficiles à comparer. Conners et Harlow (1980) ont réalisé une étude comparative des méthodes des matrices de dépendance spatiale des niveaux de gris (SGLMD), des matrices de longueur de suites de niveaux de gris (GLRLM), des matrices de différence de niveaux de gris (GLDM) et de la méthode basée sur la puissance spectrale (PSM). Ils concluent que la méthode des matrices de dépendance spatiale des niveaux de gris (SGLMD) est la plus puissante et que la méthode des matrices de longueur de suites de niveaux de gris (GLRLM) donne la plus faible performance. D'après l'auteur, la faible performance de la méthode des matrices de longueur de suites de niveaux de gris (GLRLM) est due à la sensibilité, au bruit de la méthode et au manque d'information de transition de niveaux de gris.

Les résultats obtenus par Davis, Johns et Aggarwal (1979) montrent que les matrices de cooccurence généralisées (GCM) donnent un meilleur résultat que les matrices de cooccurence de niveaux de gris.

Les résultats des méthodes basées sur des modèles sont très bons. Normalement une performance de plus de 90% est rapportée.

Un avantage important des méthodes basées sur les modèles est que l'ensemble des paramètres permet de représenter une texture d'une façon unique, ce qui est prouvé par la ressemblance entre la texture regénérée et la texture originale, alors que les méthodes basées sur l'extraction des caractéristiques texturales ne le permettent pas.

I.3 APPROCHE STRUCTURALE

I.3.1 Introduction

Du point de vue structural, une texture est définie par des éléments primitifs qui suivent une certaine règle de placement. Donc, dans l'approche structurale, l'effet est mis sur l'extraction des primitifs et la dérivation des règles de placement spatial qui les gouvernent.

Généralement, l'approche structurale est appliquée avec succès dans la caractérisation des textures plus ou moins régulières. Haralick (1979) classifie les textures en deux classes du point de vue structurale: des textures faibles (Weak textures) et des textures fortes (Strong textures). La différence réside dans l'interaction spatiale entre les primitives. Une faible interaction peut être caractérisée par des statistiques locales, alors qu'une forte interaction demande des détails de la relation topologique des éléments primitifs.

I.3.2 L'extraction des éléments primitifs

Un élément primitif est un ensemble connexe de pixels caractérisé par une liste d'attributs (Haralick 1979). Les attributs peuvent inclure des mesures concernant la forme de l'élément, l'homogénéité de sa propriété locale, etc. Les arêtes sont de bons candidats pour les

primitives, leurs attributs pourraient être la direction, la longueur, etc.

I.3.3 Description des relations spatiales

Une fois les primitives obtenues, les relations spatiales comme A DROITE, A GAUCHE, AU-DESSUS, AU-DESSOUS, PROCHE DE, LOIN DE, ENTRE,.. peuvent être quantifiées (Freeman 1975). Les mesures comme la distance minimale entre un type de primitives et d'autres types sont aussi utilisées pour décrire les relations spatiales.

I.3.4 Méthodes structurales pour la caractérisation des textures faibles

Dans ce paragraphe, on rapporte seulement des travaux publiés récemment (Gool, Dewaele et Osterlinck 1985). Pour les travaux plus anciens, on peut se referer à l'étude de Haralick (1979).

Wang, Velasco, Wu et Rosenfeld (1981) présentent 3 méthodes qui extraient des primitives basées sur le seuillage de l'image. Puis, des mesures comme la surface, le périmètre, la compacité, la direction sont obtenues pour chaque primitif. Pour discriminer les textures, les caractéristiques dérivées des histogrammes des valeurs de propriété sont utilisées.

Hong, Dyer et Rosenfeld (1980) décrivent une méthode d'extraction

des primitives basée sur des arêtes. La classification des textures est réalisée en utilisant les statistiques de premier ordre de 6 propriétés mesurées à partir de ces primitives, ce sont: la superficie, le périmètre, la dispersion, l'allongement, la direction de l'axe principal, la moyenne de niveaux de gris.

Pietikainen et Rosenfeld (1982) présentent une méthode d'extraction de primitives dont les pixels constituants ne sont pas nécessairement connexes. Des mesures texturales sont aussi basées sur les statistiques du premier ordre des attributs.

Cross et Jain (1983) proposent une méthode d'extraction des primitives basée sur le modèle markovien. Ils travaillent surtout avec des textures binaires.

I.3.5 Méthodes structurales pour la caractérisation des textures fortes

Conners et Harlow (1980) déterminent les relations spatiales par un parallélogramme périodique. Pour ce faire, ils définissent une mesure d'inertie qui peut être calculée à partir des matrices de cooccurence de niveaux de gris.

$$A = \sum_{i=1}^{N_g} \sum_{i=1}^{N_g} (i-j)^2 \, f(i,j,d) \qquad (1.41)$$

On détermine les directions où les valeurs de A sont périodiques. Deux directions a et b qui ont des périodes minimales sont choisies. Les deux vecteurs qui déterminent la règle de placement sont censés être conformes à a et b tant du point de vue direction que du point de vue amplitude.

Matsuyama, Saburi et Nagao (1980) extraient deux vecteurs spatiaux représentant la règle de placement, en évaluant la distribution de l'énergie du spectre de puissance de Fourier.

Zucker et Terzopoulous (1980) décrivent une méthode pour extraire deux vecteurs de périodicité, basée sur la diagonalité de la matrice de cooccurence.

Tomita, Shirai et Tsuji (1982) proposent une méthode dans laquelle les primitives sont extraits par l'analyse des régions, le problème d'extraction est alors posé comme un problème de segmentation d'image en des régions homogènes. Les attributs suivants sont utilisés:

- La moyenne de luminance.
- La taille de la région.
- L'allongement.
- La courbure de la frontière.

Les différents types de primitives sont obtenus par l'analyse des histogrammes des attributs. Les règles de placement sont déterminées en

calculant un histogramme bidimensionnel des positions relatives de toute paire d'éléments primitifs. Cet histogramme est censé caractériser la relation spatiale des primitives.

I.4 Conclusion

Dans ce chapitre, on a tenté de donner un aperçu des principales techniques de caractérisation de texture. Des améliorations importantes ont été réalisées au cours des dernières années au niveau de la performance et au niveau théorique. Les nouvelles techniques sont basées surtout sur des modèles probabilistes de texture comme les modèles autorégressifs, les modèles markoviens, etc. L'analyse des textures a ainsi basé sur une formulation plus mathématique, évitant des techniques heuristiques basées sur l'extraction des caractéristiques statistiques.

Les recherches récentes dans le domaine des textures ont augmenté grandement la compréhension de la notion de texture, mais l'application de l'analyse des textures dans l'industrie est encore peu répandue en raison de l'importance des ressources matérielles nécessaires pour mener à bien cette analyse.

CHAPITRE II

MÉTHODES DE SEGMENTATION DES TEXTURES

II.1 Introduction

La segmentation est un processus de partitionnement d'une image en régions connexes dans laquelle chaque région possède un degré suffisant d'homogénéité par rapport à des caractéristiques déterminées. En ce qui concerne la segmentation d'images texturées, le processus de segmentation doit assurer la partition de l'image en des régions ayant chacune un type unique de texture.

La plupart des techniques de segmentation existantes sont fondamentalement ad hoc et dépendantes de la représentation adoptée pour les caractéristiques utilisées, qu'elles soient statistiques ou autres. Les caractéristiques locales de l'image utilisées dans divers algorithmes de segmentation sont la luminance, la couleur ou la texture. C'est presque certain que les différents types de textures présentes dans une scène aident le système de vision humain dans la segmentation. Cependant, du fait de la difficulté de la représentation des textures, la segmentation des images texturées constitue un problème difficile et non résolu.

La plupart des techniques de segmentation d'image, développées

jusqu'ici en utilisant des caractéristiques locales, peuvent être utilisées pour la segmentation des images texturées. En voici quelques-unes: aggrégation, croissance des régions (region growing), agrégation spatiale (spacial clustering) et division et fusion (split and merge). Les techniques mentionnées ci-dessus sont fondamentalement des techniques basées sur des régions (region based techniques). Une autre alternative consiste à considérer le problème de segmentation comme un problème de détection de contours. Dans le cas des textures, ceci revient à déterminer l'endroit où un changement brusque des caractéristiques texturales se produit. Ce genre de technique a été étudié par plusieurs chercheurs et sera discuté dans ce chapitre.

Au cours des dernières années, des progrès substantiels ont été réalisés en segmentation d'images par estimation stochastique. Dans cette approche, l'image est modélisée par un champ aléatoire dont la structure est déterminée par un modèle stochastique, et la segmentation est réalisée en cherchant la partition de l'image qui satisfait à un critère tel que le maximum de vraisemblance ou la probabilité à posteriori maximum. Cette nouvelle méthode sera aussi discutée en détail dans ce chapitre.

II.2 Méthodes basées sur l'extraction des arêtes texturales

Des arêtes texturales ou macro-arêtes sont des endroits où une discontinuité des propriétés texturales se produit. L'effort de la

segmentation basée sur l'extraction des arêtes texturales consiste à :

- Choisir des caractéristiques texturales discriminantes.
- Appliquer des opérateurs de détection d'arêtes texturales.
- Combiner les arêtes pour former les frontières entre des régions de type différent.

Thompson (1977) propose une fonction discriminante qui peut incorporer les différentes mesures de textures associées à une région. Soit $a_i(n)$, la $i^{ème}$ propriété texturale de la région n. On peut définir la différence entre les régions 1 et m basée sur cette propriété comme:

$$d_i(1,m) = |a_i(1) - a_i(m)| \qquad (2.1)$$

L'estimation de la dissemblance entre deux régions, basée sur l'ensemble des caractéristiques texturales, peut être exprimée par la somme pondérée des mesures associées à chaque propriété.

$$D(1,m) = c_1 d_1(1,m) + c_2 d_2(1,m) + \ldots + c_n d_n(1,m) \qquad (2.2)$$

Pour détecter des arêtes texturales, Thompson utilise un opérateur semblable à celui de Robert, mais les 4 pixels formant le masque sont remplacés par 4 régions a, b, c, d.

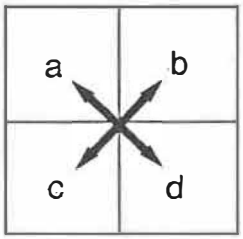


Figure 2.1 - Opérateur de Robert pour les textures.

La réponse obtenue en appliquant ce masque à un point sur l'image est donnée par :

$$T = D(a,d) + D(b,c) \tag{2.3}$$

Une valeur élevée de T signifie la présence d'une arête, et son orientation est calculée par :

$$Arg = \pm \text{Arctan}(D(a,d)/D(b,c)) \tag{2.4}$$

Triendl et Henderson (1980) proposent un modèle pour des arêtes texturales. Leur méthode de segmentation consiste à :

- Calculer n caractéristiques locales des textures et à les stocker dans n images intermédiaires.

- Appliquer un modèle d'arêtes aux images intermédiaires. Ce modèle tient compte aussi des effets de traînée, causés par l'opération de moyenne dans l'étape de calcul des caractéristiques locales.

- Grouper les arêtes pour former la frontière des textures.

Dans la première étape, 22 opérateurs sont appliqués à chaque pixel de l'image à segmenter. Ces opérateurs sont des vecteurs de gradients de 2x2 et 4x4 qui extraient le contenu de fréquence moyen et l'information de directivité moyenne des textures à l'intérieur d'une fenêtre d'observation de taille 11x11, centrée au pixel courant. Les 22 paramètres associés à chaque pixel sont réduits à 4 par la méthode d'analyse des composantes principales.

Le modèle d'arête utilisé suppose une arête de type échelon idéal estompé par l'effet de l'opération de moyenne dans la première étape. Ce modèle permet d'obtenir un détecteur d'arêtes qui maximise la corrélation entre l'arête et la petite fenêtre contenant l'arête. Ce détection une fois appliquée sur les images intermédiaires, donne la position, l'angle et la qualité des arêtes pour chacune des images intermédiaires. En combinant les arêtes de 4 images d'une façon appropriée, la frontière entre les différentes textures est obtenue.

Davis et Tychon utilisent deux mesures pour les caractéristiques locales des textures. La première mesure est basée sur une suite de n

niveaux de gris de n pixels dans une certaine direction:

$$P = p(1),\ p(2),\ \ldots\ p(n)$$

En utilisant la suite, on extrait des caractéristiques fréquentielles de la texture en comptant le nombre de différences [p(i)-p(i+1)] dont la valeur absolue dépasse un seuil et dont le signe est opposé à la différence précédente.

La deuxième mesure est basée sur la suite de différences :

$$D = d(1),\ d(2),\ ..,\ d(n)$$

où $d(i) = p(i) - p(i+1)$

On définit :

$$C = \sum_{i=1}^{n-1} |d(i)|/(n-1) \qquad (2.5)$$

C mesure le contraste local d'une texture, ainsi que la granulosité de la texture. Une texture grossière aura une valeur de C plus élevée que les textures fines.

Un opérateur de lissage est ensuite appliqué à l'image de caractéristiques pour réduire la fluctuation des mesures de D et C. Cette fluctuation est due à l'irrégularité des textures.

Finalement, une opération de seuillage est faite sur l'image résultant de l'étape précédente pour obtenir des arêtes.

II.3 Méthodes de l'agrégation dans l'espace des caractéristiques

Dans cette approche, chaque pixel est représenté par un point dans un espace à p dimensions qui correspond aux p caractéristiques associées au pixel. En première lieu, l'espace des caractéristiques est partitionné en différentes portions appelées agrégats, selon un certain critère d'agrégation. Les pixels appartenant au même agrégat portent une étiquette spécifique à cet agrégat. Un segment de l'image est alors défini comme l'ensemble des pixels connexes qui portent la même étiquette. La figure 2.2 illustre ce principe de segmentation.

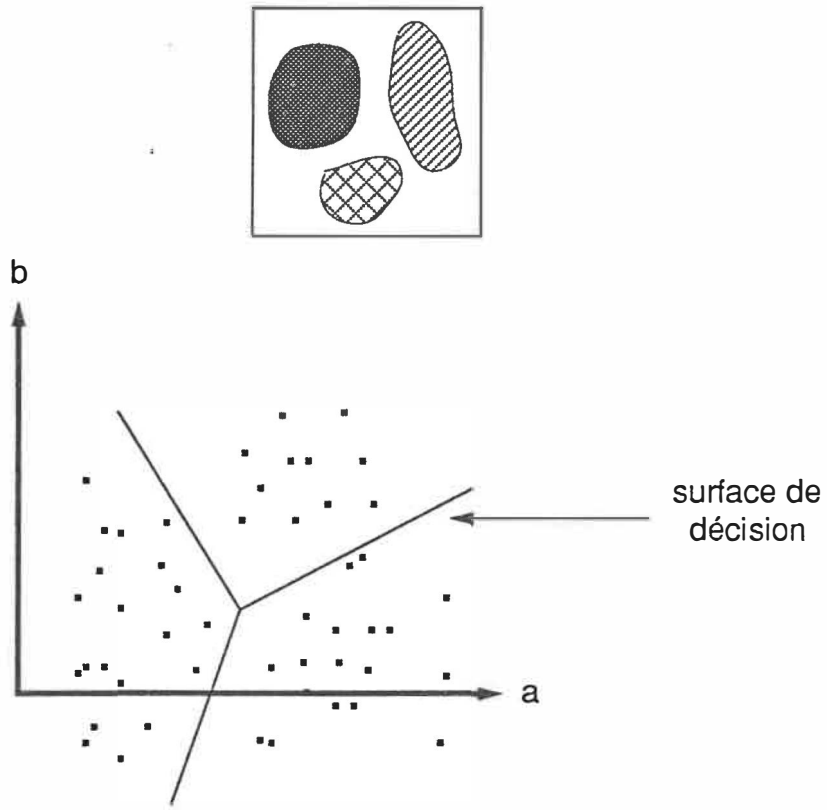


Figure 2.2 - Segmentation par agrégation dans l'espace des caractéristiques

Le succès de cette méthode de segmentation dépend d'une bonne séparation des agrégats dans l'espace des caractéristiques. Ceci exige un choix judicieux des caractéristiques associées à chaque pixel. Si l'objectif est de segmenter des régions de différents types de textures, les caractéristiques choisies doivent être discriminantes pour une grande variété de types de textures. En outre, la technique utilisée pour établir la partition de l'espace des caractéristiques doit également être choisie avec soin.

L'une des techniques utilisées est la recherche des modes dans un histogramme. Chaque mode définit un agrégat dans l'espace des caractéristiques et, par conséquent, définit un segment de l'image. Habituellement, les modes sont déterminés par les vallées d'un histogramme. La figure 2.3 illustre un exemple d'histogramme à 3 modes.

Dans cet exemple, l'agrégat 1 est l'ensemble de tous les pixels qui ont des valeurs de a inférieures à I_1, l'agrégat 2 est tel que $I_1 < a < I_2$ et l'agrégat 3 correspond à $a > I_2$.

Plusieurs techniques ont été proposées dans la détermination des seuils simples (2 modes) ou des seuils multiples (plus de 2 modes).

Cet histogramme d'une dimension contient 3 modes

1 : premier mode

2 : deuxième mode

3 : troisième mode

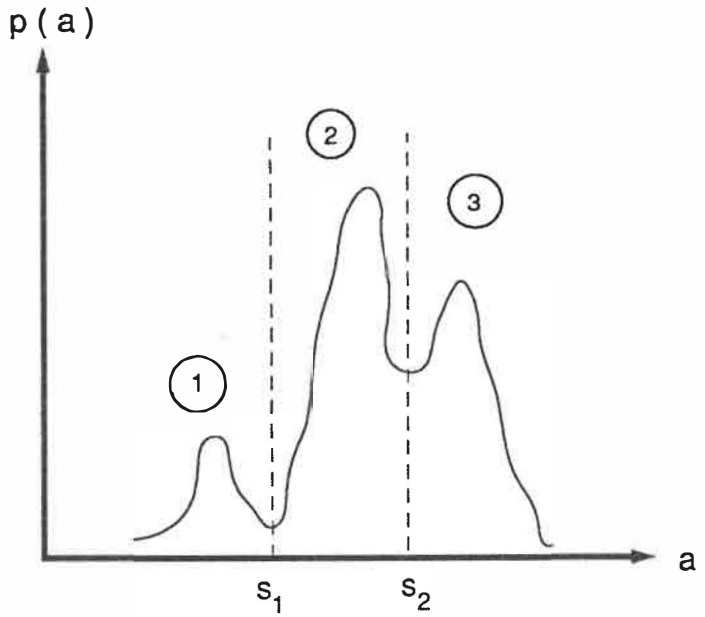


Figure 2.3 - Détection de modes dans un histogramme

Nobuyuki Otsu (1979) propose une méthode de seuillage optimal par la maximisation de la séparabilité des classes dans l'espace des mesures. Weszka et Rosenfeld (1978) appliquent une technique de transformation des histogrammes pour que les vallées soient plus profondes, ce qui rend les modes plus faciles à détecter.

Kittler et Illingworth (1986) proposent une méthode de seuillage

à erreur minimale, basée sur la supposition de la normalité des valeurs de caractéristiques. Les détails de ces techniques sont discutées dans les articles mentionnés.

Ohlander (Haralick, Shapiro 1985) raffinent la technique de la recherche des modes d'une façon récursive. Au début, l'histogramme de chaque caractéristique est calculé pour l'image entière. Parmi les histogrammes, on choisit celui qui a les modes les plus distincts et une segmentation préléminaire est réalisée en se basant sur la détection des vallées de cet histogramme. Pour chaque région ainsi identifiée, on calcule de nouveau l'histogramme de chaque caractéristique, mais cette fois-ci en se basant sur les pixels situés dans les régions identifiées par l'étape précédente. L'histogramme qui donne les modes les plus séparés est alors choisi et la segmentation de cette région en sous-régions est réalisée par la recherche des vallées comme dans l'étape précédente. Ce processus continue jusqu'à ce qu'aucune région n'ait plus de mode.

Une autre technique d'agrégation dans l'espace de caractéristiques consiste à partitionner l'espace d'une façon itérative jusqu'à la partition finale. Un critère d'optimisation est toujours requis.

Coleman et Andrews (1979) proposent une procédure de segmentation basée sur l'algorithme de K-means. Au début, on suppose que le nombre d'agrégats est connu, et les centres des agrégats sont positionnés

d'une façon arbitraire dans l'espace des caractéristiques. Les points sont alors attribués aux différents agrégats selon le critère de la distance minimale par rapport aux centres des agrégats. Après que tous ont été attribués, les centres d'agrégats sont recalculés en se basant sur les points. Ce processus continue jusqu'à ce qu'il n'y ait plus d'étiquette de changé.

Le défaut de cet algorithme réside dans le fait d'avoir à fournir le nombre d'agrégats à l'algorithme. Pour remédier à ce problème, l'auteur propose une mesure β, qui est calculée à partir des matrices des distances intra-agrégats et des distances inter-agrégats, pour mesurer la qualité de l'agrégation. Le nombre optimal d'agrégats est ainsi déterminé en choisissant le β le plus élevé. Pour connaître les détails concernant les matrices des distances intra-agrégats et des distances inter-agrégats et les différents choix des mesures de β, on peut se référer à l'article mentionné ci-dessus.

A titre comparatif, les techniques de recherche des modes sont plus rapides puisque les données ne sont balayées qu'une fois, mais la détermination des seuils multiples n'est pas triviale. De plus, l'extension à des histogrammes de dimensions élevées rend le problème de détermination de modes beaucoup plus difficile. Les techniques itératives sont plus longues d'exécution mais peuvent s'appliquer sans difficulté dans un espace de caractéristiques de dimensions élevées.

II.4 Méthodes de croissance des régions

Dans les méthodes de croissance des régions, on suppose au départ que chaque pixel constitue une région et que les régions adjacentes sont fusionnées itérativement selon un critère de ressemblance déterminé. Ainsi, chaque région croît en englobant des régions adjacentes d'allure similaire. Ce processus se termine lorsque le nombre de régions est suffisamment petit ou lorsque le critère de ressemblance n'est plus satisfait.

Il existe plusieurs techniques de croissance de régions. Ces méthodes se différencient principalement par le critère de fusionnement de deux régions ou l'inclusion d'un pixel dans une région. Les sections suivantes décrivent des principales techniques de cette approche.

II.4.1 Méthode de liaison simple

La méthode de liaison simple considère chaque pixel comme un noeud dans un graphe. Les pixels voisins, dont les propriétés sont quasi similaires, sont liés par un arc. Un segment est défini comme l'ensemble maximal de pixels appartenant à la même composante connexe. Cette méthode est simple mais présente un problème de chaînage : deux pixels semblables suffisent pour lier deux grandes régions et, par conséquent, la décision de liaison est très locale.

Cette technique ne peut pas être utilisée pour la segmentation

d'images de textures, parce que l'extraction de l'information texturale nécessite de prendre en compte un voisinage suffisamment grand autour du pixel. Les propriétés du pixel même ne sont donc pas suffisantes pour décrire les caractéristiques texturales.

II.4.2 Méthode de liaison hybride

La méthode de liaison hybride se différencie de la méthode de liaison simple par le critère de similarité. Dans cette méthode, un vecteur de caractéristiques est calculé pour chaque pixel en se basant sur le voisinage du pixel. La similarité entre deux pixels n'est pas basée sur la comparaison de deux vecteurs de caractéristiques associés à ces deux pixels, mais elle est basée sur la similarité de deux voisinages de ces deux pixels. Donc le critère utilisé est plus global, ce qui rend cette méthode plus robuste au bruit.

L'une des propriétés d'un pixel, basée sur son voisinage, consiste à l'indentifier comme situé ou non sur une arête. Les pixels voisins qui ne sont pas des arêtes sont fusionnés et un segment est défini comme l'ensemble des pixels connexes qui ne sont pas étiquetés comme des arêtes.

Evidemment, le succès de cette technique dépend du détecteur d'arêtes utilisé. Haralick et Dinstein (1984) utilisent des opérateurs simples comme l'opérateur de Roberts ou de Sobel. Puisque ces opérateurs ne donnent pas d'arêtes continues, les régions formées sont

grandes et difficiles à identifier. Pour remédier à ce problème, ils réalisent, à la suite de l'opération de détection d'arêtes, une opération de croissance des arêtes afin de combler les lacunes. Finalement, les pixels connexes qui ne sont pas des arêtes sont liés pour former des segments.

Haralick (1984) propose un détecteur d'arêtes basé sur le passage par zéro de la deuxième dérivée, en utilisant un critère d'erreur quadratique minimum (Least Square Error) pour faire concorder un voisinage avec un polynôme cubique de deux variables. Les dérivées partielles de premier ordre et deuxième ordre sont claculées pour effectuer la détection des arêtes. Une arête est déclarée présente lorsque la norme du vecteur gradient est élevé et qu'il y a un passage par zéro de la deuxième dérivée dans la direction du gradient déterminée par deux dérivées partielles de premier ordre. Dans cette technique, la continuité des arêtes est contrôlée par le seuillage de la norme du vecteur gradient.

Au lieu de déterminer si un pixel est une arête ou non, Levin et Leemet (1976) proposent un critère de similarité basé sur le degré de mélange des plus proches voisins. D'après ce critère, pour un pixel dans l'image, on regarde le voisinage et on établit une liste de pixels qui sont voisins du pixel en question et dont les propriétés sont les plus proches. Les propriétés et le critère de similarité sont arbitrairement choisis à cette étape. Deux pixels adjacents sont fusionnés si chaque pixel est dans la liste des pixels semblables aux

autres, et si les deux listes de pixels partagent un nombre suffisamment élevé de pixels communs.

La méthode de liaison hybride peut être appliquée efficacement aux textures, du fait que chaque pixel est associé à un vecteur de propriétés extraites à partir du voisinage du pixel. Ainsi, l'information texturale peut être incluse dans ce vecteur. Cependant, pour que les caractéristiques texturales soient fiables, un large voisinage est toujours requis, ce qui revient à produire deux vecteurs de caractéristiques très semblables pour deux pixels voisins, et donc à favoriser leur fusionnement. Si au lieu de considérer les pixels individuellement, on divise l'image en blocs disjoints, chaque bloc étant considéré comme une unité indivisible, on évite le problème de ressemblance des voisinages, mais on réintroduit l'effet de chaînage, car une seule erreur locale peut causer la fusion de deux grandes régions. De plus, la frontière entre deux textures est détériorée par la taille des blocs.

II.4.3 Méthode de liaison centroïde

A chaque position, le vecteur de propriétés d'un pixel est comparé au vecteur moyen de propriétés du segment en cours de formation. Si les deux vecteurs sont semblables selon un critère de similarité adéquat, le pixel est inclus dans le segment et le vecteur moyen du segment est remis à jour; sinon le pixel forme le noyau d'un nouveau segment. S'il existe déjà deux ou plusieurs segments en cours

de formation, le test de similarité est d'abord fait entre les segments pour décider s'ils sont fusionnables ou non, puis il est réalisé entre un pixel et les différents segments : ce pixel sera inclus dans un segment qui lui est connexe si leurs vecteurs de caractéristiques sont suffisamment proches.

Un exemple d'application de cette technique pour la segmentation d'images texturées est le travail de Wong et Raafat (1986). Ils divisent l'image en blocs de même taille. La semence d'une région est déterminée par une mesure de l'information texturale I qui donne une valeur faible pour des blocs qui contiennent juste une texture et une valeur élevée si plusieurs textures sont présentés dans le bloc.

L'algorithme comprend trois étapes :

- L'initialisation des régions.
- L'extension des régions par la technique de liaison centroïde.
- Le raffinement de la frontière.

Dans la phase d'initialisation des régions, la première semence d'une région est déterminée en choisissant le bloc dont la mesure I est la plus petite. Une tentation d'inclure les blocs homogènes de même type est faite en imposant un critère d'inclusion très sévère basé sur des seuils. Ceux-ci permettent la fusion des blocs qui ne sont pas nécessairement connexes.

L'extension des régions est faite par la fusion chacun des blocs restants à l'une des régions noyaux par la comparaison du vecteur de propriétés moyennes de chacune des régions à celui de chaque bloc.

Finalement, les blocs situés à la frontière des régions sont divisés en petits blocs, puis une procédure de fusion de ces blocs aux régions voisines, basée sur un test de similarité est effectuée.

II.5 Méthode d'agrégation spatiale

Cette méthode combine la technique d'agrégation dans l'espace de caractéristiques avec la technique de croissance spatiale des régions, la première servant à guider l'action de la seconde. Au début, un histogramme multidimensionnel des caractéristiques est calculé et les modes de l'histogramme sont isolés. Les pixels correspondant à chaque mode sont utilisés comme le noyau d'une région. On examine ensuite les pixels voisins de ce pixel et on évalue la proximité de ces pixels avec le pixel courant dans l'espace des caractéristiques. S'ils sont assez proches, ils sont joints par un arc.

Milgram (1979) définit un segment comme l'ensemble des pixels connexes dont les distances inter-pixels dans l'espace des caractéristiques se trouvent dans un certain intervalle I, et la frontière de ce segment coïncide bien avec les arêtes créees par un détecteur d'arêtes. Cette technique a l'avantage de ne pas explorer l'espace des caractéristiques dans la recherche des pixels semblables

autour du sommet, cependant elle doit être essayée avec plusieurs intervalles I différents pour avoir un segment dont la frontière coïncide bien avec les arêtes.

Burt, Hong et Rosenfeld (1981) proposent une technique d'agrégation spatiale en multirésolution. La pyramide est construite de telle façon que chaque niveau plus élevé de la pyramide est une image dont le nombre de lignes et de colonnes est égal à la moitié du nombre de lignes et de colonnes de l'image du niveau inférieur. L'image originale est à la base de la pyramide. Au début un noeud parent est lié avec 4x4=16 noeuds enfants. Deux noeuds parents voisins partagent 8 enfants communs. La figure 2.4 montre la construction d'une telle pyramide de deux niveaux.

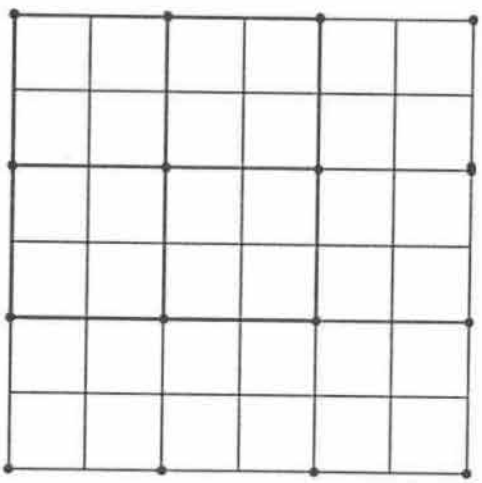


Figure 2.4 - Construction d'une pyramide de deux niveaux.

Au cours des itérations, le vecteur de propriétés d'un noeud père est calculé en prenant la moyenne des vecteurs de propriétés des noeuds enfants qui lui sont liés. Pour chaque noeud enfant, on compare son vecteur de propriétés avec ceux de ses parents et on établit un lien avec le noeud parent ayant le vecteur de propriétés le plus proche. Le processus converge quand plus aucun lien n'est changé. En traçant les liens à partir d'un noeud plus haut dans la pyramide, on peut isoler des segments correspondants dans l'image originale. Le nombre de segments obtenus dépend du niveau à partir duquel on commence le traçage. La même technique a été appliquée par Pietikainen et Rosenfeld (1981) aux images texturées, dans lesquelles le vecteur de propriétés de chaque noeud contient de l'information texturale.

II.6 Méthode de division-fusion

La méthode de division-fusion commence en considérant l'image originale comme une région initiale. Un test d'homogénéité est appliqué à la région. Si la région n'est pas suffisamment homogène, elle est divisée en petites régions selon une partition prédéfinie. Habituellement la structure d'un arbre quaternaire est utilisée. Chaque sous-région est ensuite soumise au test d'homogénéité. Le processus de division continue jusqu'au moment où aucun bloc ne peut plus être divisé, c'est-à-dire que tous les blocs sont homogènes. A ce moment, le processus de fusionnement commence. Pour chaque ensemble de 4 blocs homogènes adjacents, on teste si son union est uniforme, l'affirmation permet de fusionner ces 4 blocs en une région unique. Le processus

continue jusqu'au moment où aucune fusion ne peut plus se produire.

Les diverses techniques publiées, selon cette approche, se différencient par la nature du test d'homogénéité. Muerlen et Allen (Haralick et Shapiro 1985) utilisent le test de Kolmogorov-Smirnov, Chen et Pavlidis (Chen et Pavlidis 1980) proposent un test statistique basé sur la moyenne de la région en question et de chacune des 4 sous-régions. Dans l'application de la technique pour les images texturées, Chen et Pavlidis utilisent les coefficients de corrélation d'une région et ceux de ses sous-régions. Les mêmes auteurs proposent également une technique de Division-et-Fusion basée sur les matrices de cooccurence pour la segmentation des images texturées (Chen et Pavlidis 1982).

II.7 Méthodes de segmentation par estimation

Depuis quelques années, plusieurs efforts de recherche dans le domaine de la segmentation d'images ont été orientés vers une approche par l'estimation stochastique, dans laquelle la partition optimale de l'image est obtenue en maximisant un critère statistique défini par rapport à un modèle de l'image et un modèle de la formation des régions.

Cette façon de formuler le problème de segmentation nécessite 3 éléments de base:

- Le modèle de l'image et de la formation des régions.
- Les critères statistiques qui définissent la segmentation.
- Les techniques adoptées pour atteindre le critère.

Derin et Cole (1986) adoptent le modèle de Gibbs pour les textures et pour le processus de formation des régions. On peut se référer à la section I.2.8.1.d du chapitre précédent pour la description générale du modèle. Pour avoir du lien entre le modèle des textures des différentes régions avec le modèle de la formation des régions elles-mêmes, les auteurs proposent un modèle gibbsien hiérarchique.

Modèle gibbsien hiérarchique

L'image $y=\{y_{ij}\}$ est une matrice de $N_1 x N_2$ des observations d'une scène. Elle est une réalisation spécifique d'un champ aléatoire $Y=\{Y_{ij}\}$. La scène est également assimilée à une réalisation d'un champ aléatoire $X=\{X_{ij}\}$. Le modèle hiérarchique est formulé comme suit.

La scène X est un champ aléatoire discret où X_{ij} prend des valeurs dans $Q=\{1, 2, .., M\}$ pour chaque (i,j) dans l'image. Une réalisation de la variable aléatoire X_{ij} du champ $X=\{X_{ij}\}$ est x_{ij}. $x_{ij}=k$ signifie que le pixel (i,j) dans une réalisation de X appartient à la texture du type k. Ainsi, une réalisation x de X est simplement une partition de l'image en M régions, chaque région contenant un type de texture. Le champ aléatoire X est un processus de Gibbs dont la

fonction d'énergie associée à chaque clique est définie de la façon suivante:

$$V_c(x) = \begin{cases} -\theta \text{ si tous les pixels dans la clique c sont égaux} \\ 0 \text{ autrement} \end{cases}$$

où θ est un paramètre qui dépend du type de clique.

Chaque type de texture est modélisé par un processus de Gibbs, T^k qui appartient aussi à la classe particulière mentionnée ci-dessus. $T^k{}_{ij}$ prend ses valeurs dans $P=\{0, 1, 2, .., q_k\}$, $k=1, 2, 3, .., M$, d'où q_k est le nombre de niveaux de gris dans la texture T^k. On dénote t^k une réalisation particulière de T^k. On suppose aussi que le processus de région X et les textures T^k sont des champs aléatoires indépendants.

L'image texturée est modélisée comme un champ aléatoire $Y=\{Y_{ij}\}$ qui suit l'expression :

$$Y_{ij}(.) = T^k{}_{ij}(.), \text{ si } X_{ij}(.)=k, \; k=1, 2, .., M \qquad (2.6)$$

Ceci signifie que la réalisation de y au pixel (i,j) est la réalisation de la texture de type T^k_{ij} au pixel (i,j). Cette formulation représente le modèle gibbsien hiérarchique.

Critère de segmentation:

L'objectif est de déterminer la réalisation de la scène x, basée sur la réalisation y de l'image. L'auteur adopte la probabilité à posteriori maximum (MAP) comme critère d'optimisation, ce qui revient à déterminer l'estimation x de X qui maximisa p(X=x/Y=y).

Technique d'optimisation:

Selon la loi de Bayes:

$$p(X=x|Y=y) = \frac{p(Y=y|X=x)p(X=x)}{p(Y=y)} \qquad (2.7)$$

Puisque p(Y=y) ne dépend pas de la partition:

$$\text{Maximiser de } p(X=x|Y=y) \equiv \text{Maximiser } p(Y=y|X=x)p(X=x)$$

Pour une image de $N_1 x N_2$ dans laquelle M types de texture sont présents, il y a M^{N1xN2} configurations possibles de la scène. De plus, pour chaque région de D pixels de type k de la scène, il y a $q^D{}_k$ réalisations possibles de la texture du type k. Un tel nombre de configurations possibles rend la méthode de maximisation directe irréalisable. L'auteur propose d'utiliser une technique de programmation dynamique.

$$p(Y=y|X=x) = \prod_{(i,j)} [P(Y_{\nu ij}=y_{\nu ij}|X_{ij}=k)] \qquad (2.8)$$

où ν_{ij} est un voisinage du pixel (i,j). Ceci suppose l'indépendance statistique des voisinnages. De plus, pour réduire la dimension de l'espace d'état, l'estimation de la scène entière est réalisée bande par bande. Chaque bande est composée de D lignes consécutives et l'estimation de la scène associée à une bande est réalisée par l'algorithme de programmation dynamique, avec le nombre d'état égal à M^D au lieu de M^{N1} si l'on traite l'image entière.

Un tel traitement suppose que la corrélation des pixels qui sont assez éloignés les uns des autres est négligeable.

Des essais sur des textures synthétiques qui ont un nombre de niveaux de gris plus petit que 5 donnent un résultat remarquable. La limite du nombre de niveaux de gris est due à l'algorithme de programmation dynamique qui impose un nombre d'états limité.

Cohen et Cooper (1987) posent aussi le problème de segmentation comme un problème d'estimation stochastique dans lequel :

- Les textures sont modélisées par un champ aléatoire markovien qui suit le modèle gaussien.

$$p(Y_{ij}=y_{ij}\,|\,\nu_{ij},\text{type } k) = [2\pi\sigma^2(k)]^{-1/2}\exp\{-(1/2\sigma^2(k))[y_{ij}-\mu(k)-\sum_{(k,l)\in\nu_{ij}}\beta_{(i,j),(k,l)}(k)(y_{kl}-\mu(k))]^2\} \quad (2.9)$$

où $\mu(k)$: moyenne de niveaux de gris de la texture k

$\sigma^2(k)$: sa variance

$\beta_{(i,j),(k,l)}$ représente les paramètres du modèle. Ces paramètres expriment l'interaction des pixels selon leurs positions géométriques relatives dans l'image. Pour la définition de l'ordre de voisinage, on peut se référer à la section I.2.8.1 du chapitre précédent.

La probabilité conditionnelle du signal d'image sachant la texture de type k est exprimée par:

$$p(Y=y \mid \text{classe } k) = [2\pi\sigma^2(k)]^{-M/2} |B(k)|^{1/2} \exp\{-[(y-U(k))]^t B(k)[y-U(k)]/2\sigma^2(k)\} \quad (2.10)$$

d'où $U(k) = \mu(k)(1\ 1\ \ldots\ 1)^t$

y est un (Mx1) vecteur de colonne dont les éléments sont des niveaux de gris dans une région de M pixels de texture du type k.

B(k) est une matrice d'interaction dont l'élément b_{rv} est égal à β_{rv} d'où r = (i,j), v = (k,l).

- Pour le modèle de la formation des régions, l'auteur utilise le processus markovien autobinaire. Soit S le champ des régions, $S=\{s_{ij}\}$ si l'image contient seulement deux types de textures, s_{ij} prend la valeur 0 ou 1, indiquant l'allocation du pixel (i,j) à la texture du type 0 ou 1. La probabilité conditionnelle est exprimée alors par :

$$p(s_{ij} \mid S_{(i,j)}) = \exp\{(s_{ij} T_{(i,j)}/[1+\exp(T_{(i,j)})]\} \quad (2.11)$$

$$\text{d'où} \quad T_{(i,j)} = a + \sum_{(k,l)\in\nu_{ij}} b_{(i,j),(k,l)} s_{kl} \quad (2.12)$$

$S_{(i,j)}$ est un voisinage de (i,j) excluant le pixel (i,j)

Techniques de segmentation adoptées:

L'auteur propose deux algorithmes de segmentation : l'algorithme de segmentation hiérarchique et l'algorithme de relaxation.

Pour l'algorithme de segmentation hiérarchique, on divise l'image en des blocs de taille égale à NxN. Chaque bloc est traité comme une sous-image et les résultats sont combinés d'une façon appropriée.

Supposons qu'il existe juste deux types de textures dans chaque bloc. La segmentation est définie comme la partition qui maximise $p(Y=y|S=s)$, où y représente le vecteur d'observation du bloc et s le vecteur de partition dans le bloc.

Ce critère est appelé la stratégie du maximum de vraisemblance (ML). En supposant l'indépendance stochastique des régions de type 0 et 1, ceci est équivalent à maximiser $p(Y^0|0)p(Y^1|1)$, où Y^k représente le vecteur de la fonction d'image (les niveaux de gris) pour la texture de type k à l'intérieur du bloc.

L'algorithme de segmentation hiérarchique utilise l'arbre quaternaire comme structure de données. On prend l'image originale comme bloc initial. Chaque bloc est divisé en 4, l'affectation des étiquettes aux sous-blocs est faite de telle façon que la probabilité conjointe des données d'image soit maximisée. Plus explicitement, supposons qu'on est à l'étape 2 du processus de division et que

l'affectation des types de textures est montrée dans la figure 2.5.

0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1

Figure 2.5 - Illustration de l'affectation des étiquettes à l'étape 2.

On commence alors l'étape suivante dans laquelle chaque bloc de l'étape 2 est divisé en 4 et l'affectation des étiquettes pour chaque quadrant est telle que :

$$p(X^0 \mid Y^0, 0)p(X^1 \mid Y^1, 1) \text{ est maximisée}$$

où X représente le vecteur des données du bloc en question et Y le vecteur des données des blocs restants.

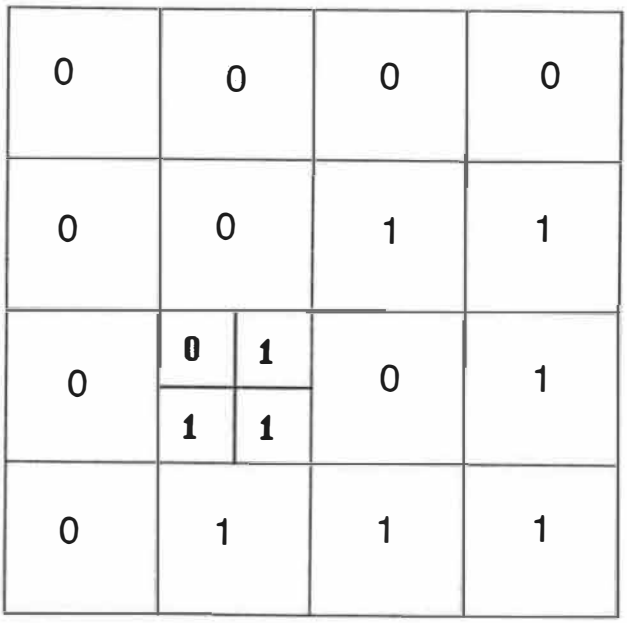


Figure 2.6 - Illustration d'une division quaternaire à l'étape 3.

Le modèle gaussien du champ markovien permet d'exprimer la probabilité conjointe conditionnelle comme suit :

$$p(X^k \mid Y^k, k) = (2\pi\sigma^2(k))^{-N_X/2} \mid B_x(k) \mid^{1/2} \exp\{-(1/2\sigma^2(k)) [X^t B_X X + 2X^t B_{XY} Y + Y^t B^t_{XY} B^{-1}_X B_{XY} Y]\} \quad (2.13)$$

où N_X est le nombre de pixel dans le bloc X

$B_X(k)$ représente matrice d'interaction définie comme précédemment, mais limitée au vecteur X.

B_{XY} représente la matrice d'interaction correspondant au vecteur de la concaténation du vecteur X et du vecteur Y. L'auteur considère juste les blocs Y au voisinage du bloc X.

L'algorithme de segmentation hiérarchique est basé seulement sur le modèle des textures. L'incorporation d'un modèle stochastique pour la formation des régions permet de formuler le problème de segmentation par la stratégie de la probabilité à posteriori maximum (MAP). Dans ce travail, le modèle utilisé pour la formation des régions est le modèle markovien autobinaire.

Dans la stratégie MAP, on cherche à maximiser $p(S|Y)$. Ceci est équivalent à maximiser $p(Y|S)p(S)$. Pour résoudre le problème l'auteur adopte un algorithme de relaxation qui permet de trouver un minimum local. Dans cet algorithme, l'état du pixel (i,j) à l'itération n, s^n_{ij} sera changé de façon à maximiser $p(S|Y)$. Une fois appliqué à tous les pixels de l'image, ceci produit une nouvelle estimation S^{n+1} qui est plus proche du maximum de $p(S|Y)$. Le point essentiel de l'algorithme de relaxation est de décomposer le problème de l'estimation globale en des problèmes de décisions locales:

$$p(S=s) = p(s_{ij},S_{ij}) = p(s_{ij}|S_{ij})p(S_{ij}) \qquad (2.14)$$

où S_{ij} est l'état du voisinage du pixel (i,j) sans compter ce pixel même.

Puisque $p(S_{ij})$ n'est pas une fonction de s_{ij}, on peut l'éliminer dans l'expression sans changer le résultat. L'autre terme, $p(Y|S)$, peut s'exprimer :

$$p(Y|S) = p(y_{ij},Y_{ij}|S) = p(y_{ij}|Y_{ij},S)p(Y_{ij}|S) \qquad (2.15)$$

Par le même raisonnement que précédemment, on peut négliger $p(Y_{ij}|S)$.

Donc le choix de l'état du pixel (i,j) à l'itération (n+1), $s^{n+1}{}_{ij}$, dépend de la valeur de R définie comme :

$$R = \frac{p(y_{ij}|Y_{ij},s^{n+1}=1,s^{n}{}_{ij})p(s^{n+1}{}_{ij}=1|s^{n}{}_{ij})}{p(y_{ij}|Y_{ij},s^{n+1}=0,s^{n}{}_{ij})p(s^{n+1}{}_{ij}=0|s^{n}{}_{ij})} \qquad (2.16)$$

Si $R > 1$: $s^{n+1}{}_{ij} = 1$

si non : $s^{n+1}{}_{ij} = 0$

Pour les autres travaux traitant le problème de segmentation comme une procédure d'estimation, on peut se référer entre autres aux travaux de Therrien, Quatieri et Dudgeon (1986), et de Corman et Geman (1984).

II.8 Conclusion

Dans ce chapitre nous avons présenté les principales méthodes de segmentation d'image. Pour chaque méthode nous avons discuté en détail de son fondement et des divers travaux qui lui sont reliés. Certaines méthodes s'appliquent mieux que d'autres aux images texturées, mais en

général, toutes les techniques utilisées pour la segmentation des images peuvent être applicables aux textures avec des modifications mineures. Ces modifications sont nécessaires parce qu'il faut tenir compte du fait que les caractéristiques texturales sont des propriétés de régions, non pas des points.

Les méthodes de segmentation par estimation sont basées sur un fondement mathématique plus solide, mais dans la plupart des cas, elles ont besoin d'efforts de calcul importants; des approximations sont donc toujours nécessaires. L'application de ces techniques aux textures naturelles se heurte à des difficultés reliées à l'estimation des paramètres du modèle et à la variation spatiale non négligeable des propriétés texturales.

CHAPITRE III

SEGMENTATION DES TEXTURES PAR ESTIMATION BAYESIENNE

III.1 Théorie du champ markovien à 2-D

III.1.1 Introduction

Les champs aléatoires markoviens ont été introduits à l'origine par Ising (1925) dans ses études sur le domaine ferromagnétique. Initialement, le modèle a été développé dans le cas des variables binaires ou gaussiennes. Besag (1974) a fait l'extension au cas des variables entières. L'extension du modèle se trouvait bien appropriée pour la modélisation de l'image. Il a été étudié de manière approfondie par Wood (1972). L'application du modèle dans le cas des textures est due à Cross (1980, 1983), Kashyap et Chellappa (1983).

Le modèle markovien 2-D suppose que chaque cellule dans une maille représente une variable aléatoire. Dans le cas des images digitales, les cellules correspondent aux pixels et les variables aléatoires associées aux cellules correspondent aux niveaux de gris. Le modèle markovien 2-D exploite la dépendance probabiliste d'une variable aléatoire de ses voisinages; donc, la caractérisation d'un modèle de ce type consiste à déterminer le modèle d'interaction spatiale entre la

variable considérée et son voisinage. Si le processus d'image est stationnaire, ce modèle d'interaction est identique en tout point de l'image.

Dans la spécification du modèle d'interaction, la notion de voisinage est requise, elle sera présentée dans la section suivante.

III.1.2 Système de voisinage sur une maille 2-D

Considérons un pixel r dans une image digitale, le voisinage de r, $D_p(r)$, est l'ensemble de tous les pixels autour de r. Dépendant de la distance spaciale entre r et les pixels voisins, on introduit la notion d'ordre d'un système de voisinage. On définit que D_p est un système de voisinage de l'ordre η si:

$$D_p(r) = \{v=(l,m), \text{ tel que: } ||r-v||^2 \leq N_\eta \text{ et } v \neq r\} \qquad (3.1)$$

où : N_η=1,2,3,4,5,.. correspond à η=1,2,3,4,..

En basant sur cette définition, on obtient les systèmes de voisinage suivants:

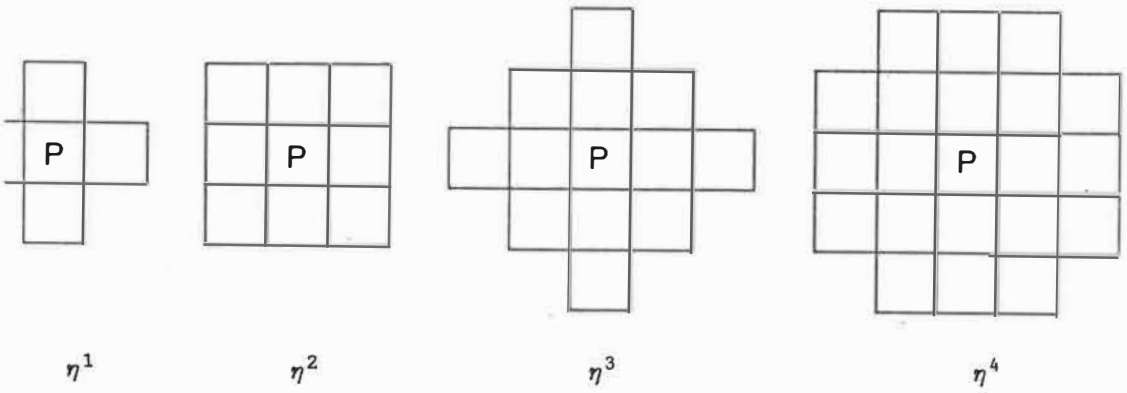


Figure 3.1 - Systèmes de voisinage.

III.1.3 Définition du champ markovien

Soit r = (i,j) où i et j représentent la position en ligne et en colonne dans une image, $1 \leq i,j \leq N$, où N est la taille de l'image. Soit (x_r) le champ aléatoire, avec x_r étant le champ au pixel r. Soit $X_{(r)}$ le champ de toute l'image, excepté le pixel r, (x_r) est un champ aléatoire markovien si :

$$P(x_r \,|\, x_{(r)}) = P(x_r \,|\, \forall x_v,\ v \epsilon D_p(r)) \qquad (3.2)$$

Ici, $D_p(r)$ désigne l'ensemble des voisins de r et dépend de l'ordre du système de voisinage comme présenté précédemment à la section III.1.2

Ceci signifie que la dépendance probabiliste de chaque variable aléatoire x_r est limitée à ses voisins immédiats et l'extension de voisinage est spécifié par son ordre. Dans le cas des images, cette dépendance limitée a du sens, parce que la réalisation du processus d'intensité à un pixel est loin d'être influencée par celle qui est loin de lui. Elle est, due à la symétrie des systèmes de voisinage, de nature isotropique.

La définition (3.2) a introduit la notion de dépendance probabiliste du champ markovien, mais n'a pas spécifié encore le modèle d'interaction entre la variable aléatoire x_r et ses voisins $\{x_v, v\epsilon D_p, v\neq r\}$. Plusieurs modèles d'interaction ont été proposés et font l'objet d'étude dans la section suivante.

III.1.4 Représentation du champ markovien 2-D

On entend, par la représentation d'un champ markovien, tout modèle mathématique qui spécifie la nature de l'interaction spatiale entre une variable du champ et son voisinage.

III.1.4.1 Modèle de Gibbs

Le modèle de Gibbs spécifie la probabilité conjointe du champ markovien par la distribution de Gibbs définie comme suit:

- Définition:

Soit L, la maille et η^i, un système de voisinage ordre i. Un champ aléatoire $X = \{x_r\}$ défini sur L, suit une distribution de Gibbs par rapport à η^i, si et seulement si sa distribution conjointe est de la forme :

$$p(X=x) = \frac{1}{Z} e^{-U(x)} \qquad (3.3)$$

On appelle U(x) la fonction d'énergie associée à la réalisation x du champ X, et Z est une constante de normalisation qui assure que l'expression (3.3) est une fonction de densité de probabilité.

La fonction d'énergie U(x) est déterminée par la contribution additive de l'énergie associée à des entités élémentaires extraites du voisinage η^i dont les formes représentent différents types d'interaction. Ces entités élémentaires sont appelées cliques.

- Définition de clique:

Une clique c dans un système de voisinage d'ordre i est un sous-ensemble de la maille L, tel que:

* c contient seulement un pixel du voisinage

* ou bien, deux pixels r et v appartiennent à une clique c si r est un voisin d'ordre i de v.

Revenons à la fonction d'énergie, elle peut s'écrire:

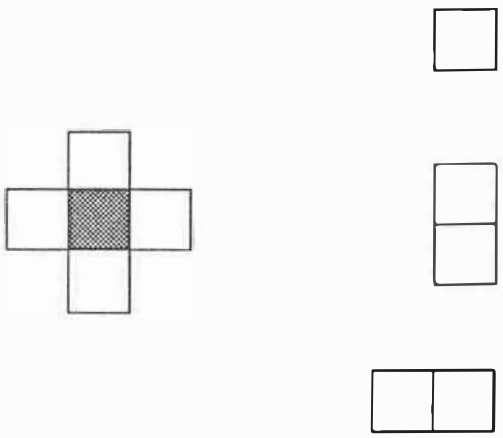
$$U(x) = \sum_{c \in C} V_c(x) \qquad (3.4)$$

où $V_c(x)$ est la fonction d'énergie associée à la clique c

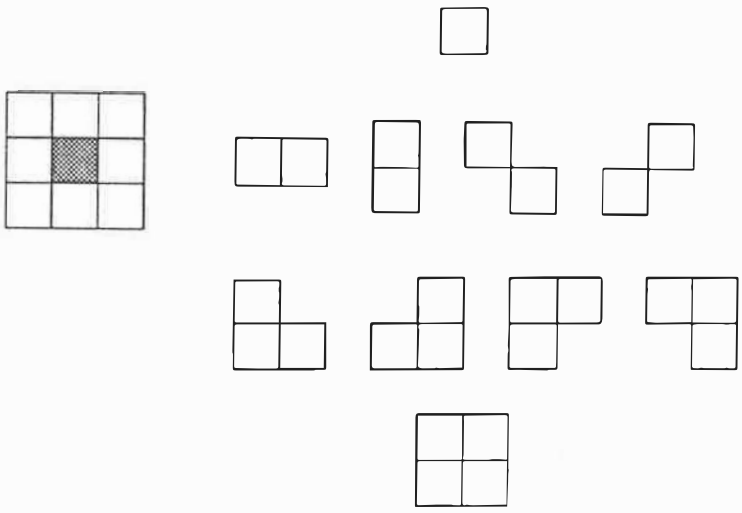
et C est l'ensemble de tous les types de clique associés au voisinage η^i.

La figure 3.2 illustre les différents types de clique associés à différents voisinages.

Chaque type de clique représente un type d'interaction et la valeur de la fonction d'énergie correspondante représente la force de ce type d'interaction.



η^1 Type de clique



η^2 Type de clique

Figure 3.2 - Systèmes de voisinage et les types de clique associés.

- Théorème Hammersley-Clifford:

Ce théorème établit la correspondance bi-univoque entre le champ aléatoire markovien et le champ aléatoire représenté par la distribution de Gibbs.

Soit η^i, un système de voisinage d'ordre i sur la maille L. Un champ aléatoire X est markovien par rapport à η^i si et seulement si sa distribution conjointe est une distribution de Gibbs avec des cliques associées à η^i

Ce théorème permet d'exprimer la fonction de distribution conjointe sous une forme simple et facile à manipuler mathématiquement.

III.1.4.2 Modèle gaussien 2-D

Le modèle gaussien a été introduit d'abord par Rosanov (1967), Woods (1972). Depuis, plusieurs travaux ont été réalisés en appliquant le modèle dans la restoration, synthétisation, codage, classification, segmentation d'image.

Dans cette section, nous présentons le modèle gaussien 2-D adapté par Cohen et Cooper (1987).

- Définition:

Soit X un champ aléatoire 2-D défini sur la maille L, soit D_p un système de voisinage d'ordre η^i, X est gaussien si la fonction de probabilité conditionnelle est exprimée comme suit :

$$p(x_r/X_{(r)}) = [2\pi\sigma^2]^{-1/2}\exp\left\{-\frac{1}{2\pi\sigma^2}[x_r-\mu-\sum_{v\epsilon D_p}\beta_{r-v}(x_v-\mu)]^2\right\} \qquad (3.5)$$

où : $\mu = E[x_r]$, $\forall r\epsilon L$

$\sigma^2 = E[(x_r-\mu)^2]$

Si on ordonne X en un vecteur par concaténation ligne par ligne et si le champ est supposé stationnaire, la fonction de probabilité conjointe est exprimée par [ ]:

$$p(X) = [2\pi\sigma^2]^{-M/2}\,|B|^{1/2}\exp\{-[X-U]^tB[X-U]/2\sigma^2\} \qquad (3.6)$$

où : M est le nombre de pixels dans l'image

X est un vecteur de M éléments

U = $\mu(1,1,\ldots,1)$, U est aussi un vecteur de M éléments

B est une matrice de MxM éléments, chaque élément de B, b_{rv}, représente la force d'interaction spatiale entre le point r et le point v du champ. Donc, B est une matrice symétrique dont les éléments diagonaux sont 1 et $b_{r,v}=b_{v,r}$. De plus $b_{r,v}=0$ si r et v ne sont pas voisins.

Le modèle gaussien a été utilisé largement dans la modélisation des textures et a connu un certain succès dans la classification et segmentation des textures.

III.1.4.3 Modèle autobinomial:

Ce modèle a été proposé par Cross et Jain (1983). Ils utilisent la distribution autobinomiale pour exprimer la dépendance probabiliste des variables aléatoires. Selon ce modèle, la probabilité $p(x_r/X_{(r)})$ est binomiale avec le paramètre $\theta(T)$ ci-dessous et un nombre (G-1) de tirages où G représente le nombre de niveaux de gris

$$\theta(T) = \frac{\exp(T)}{1 + \exp(T)} \qquad (3.7)$$

L'expression de T dépend de l'ordre du voisinage.

* Premier ordre :

$$T = a+b(1,1)(t+t')+b(1,2)(U+U')$$

* Deuxième ordre :

$$T = a+b(1,1)(t+t')+b(1,2)(U+U')+b(2,1)(v+v')+b(2,2)(z+z')$$

* Troisième ordre :

$$T = a+b(1,1)(t+t')+b(1,2)(u+u')+b(2,1)(v+v')+b(2,2)(z+z')+ b(3,1)(m+m')+b(3,2)(l+l')$$

où : a, b(α,β) représente la force d'interaction spatiale selon la distance relative et la position relative entre le pixel x et ses voisins.

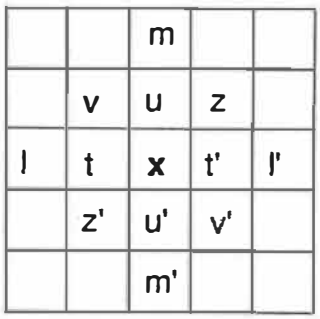


Figure 3.3 - Position des pixels selon l'ordre du voisinage.

III.2 Application du modèle markovien dans la segmentation des images texturées

III.2.1 Introduction

La modélisation de l'image par le champ aléatoire markovien trouve plusieurs applications dans l'analyse d'image : restauration, synthétisation, codage, classification, segmentation. La segmentation

des images en utilisant le modèle markovien est souvent basée sur les deux stratégies suivantes:

Stratégie de vraisemblance maximum

Cette stratégie consiste à trouver une partition de l'image qui maximise la probabilité conjointe de la réalisation x sachant la partition. La réalisation x du champ aléatoire X est l'image à segmenter. Une partition de l'image consiste à assigner des étiquettes aux pixels dans l'image, chaque étiquette représentant un type de région. Dans le cas des images multitexturées, chaque étiquette représente un type de texture présente dans l'image. Donc, pour une image de taille 2 NxN, l'existence de c classes de texture dans cette image donne c^{N^2} partitions possibles. Chercher une partition parmi c^{N^2} qui maximise la fonction de probabilité représente un énorme travail, la recherche exhaustive n'est jamais la méthode choisie. Souvent on se contente d'une solution sous optimale qui demande un effort de calcul appréciable.

Stratégie de probabilité à posteriori maximum

Dans la stratégie précédente, on considère la partition de l'image comme un processus déterministe et la modélisation des régions n'est pas nécessaire. Cependant la stratégie de probabilité à posteriori maximum exige un modèle stochastique pour des régions, parce que cette stratégie consiste à trouver une partition (une réalisation du champ des régions) qui maximise :

$$p(\text{partition}|x)$$

par la loi de Bayes :

$$p(\text{partition}|x) = \frac{p(x|\text{partition})\ p(\text{partition})}{p(x)} \qquad (3.8)$$

Parce que p(x) est la même pour toute position, donc:

$$\text{Maximiser}\{p(\text{partition}|x)\} \equiv \text{Maximiser}\{p(x|\text{partition})p(\text{partition})\}$$

Ici, on entend par partition une réalisation du champ aléatoire des régions. Des champs aléatoires de type markovien ont souvent été utilisés dans la litérature pour modéliser le processus de formation des régions.

Le modèle markovien des régions permet d'incorporer les caractéristiques des régions dans le modèle; par conséquent, le résultat obtenu avec cette stratégie est souvent meilleur que celui de la stratégie de vraisemblance maximum. Il est meilleur au niveau de la qualité des contours et au niveau de l'élimination des petites régions parce que la forme du contour et la taille des régions peuvent être incorporées dans le modèle.

Dans les trois sections qui suivent, on va présenter trois différentes approches rapportées récemment dans la litérature pour fins de segmentation et de restauration d'une image multitexturée. Ces approches partagent les points communs suivants:

- L'utilisation du processus aléatoire markovien pour modéliser l'image et de la formation des régions.

- L'utilisation des mêmes stratégies de segmentation, soit la stratégie de Vraisemblance Maximum ou la stratégie de Probabilité à Posteriori Maximum.

Elles se différencient par la méthode choisie pour arriver à obtenir un état qui satisfait à un des deux critères mentionnés ci-dessus. Concrètement, on va étudier les méthodes utilisées par Derin et Coles (1986), et Cohen et Cooper (1987) pour la segmentation des images texturées, et la méthode utilisée par Geman et Geman (1984) pour la restauration d'image. Elles sont choisies comme illustration de l'application du modèle markovien dans l'analyse d'image, notamment dans la segmentation et dans la restauration d'image. Elles sont choisies aussi parce que les techniques qu'elles proposent sont représentatives de l'ensemble des techniques utilisant le champ markovien dans l'analyse d'image. Elles servent de base de comparaison avec notre méthode de segmentation qui va être présentée dans la partie III.3 de ce chapitre.

III.2.2 Segmentation des images texturées par la programmation dynamique (Derin et Cole)

III.2.2.1 Modèle gibbsien hiérarchique

Soit $Y = \{y_{ij}\}$ un champ aléatoire markovien de deux dimensions défini sur une maille L. La maille L associée à ce champ est l'ensemble des emplacements des pixels satisfaisants:

$$L = \{(i,j) : 1 \leq i \leq N_1,\ 1 \leq j \leq N_2\}$$

où : $N_1 x N_2$ est la taille de l'image.

Sur la maille L, le système de voisinage est défini comme celui de la section III.1.2.

L'objectif est de construire un champ aléatoire Y qui caractérise l'image multitexturée.

Dans le modèle hiérarchique, le champ d'intensité Y est défini en termes du champ des régions $S = \{s_{ij}\}$. Le champ des régions S est un champ aléatoire défini aussi sur la maille L, et une réalisation s de S est simplement une partition de la maille L dans des régions de types différents. Chaque type est étiqueté par un nombre appartenant à l'ensemble {1,2,...,K}, où K est le nombre maximum des régions.

Sur la base de la réalisation de S, on construit le modèle de Y. La construction est que pour un pixel (i,j) dans la maille L, la réalisation de $s_{ij}=k$ signifie que y_{ij} suit un champ de texture de type T^k. Le champ de texture T^k est lui-même un champ aléatoire markovien.

Derin et Cole utilisent la distribution de Gibbs décrite à la section III.1.4.1 pour représenter les deux champs S et T^k. La fonction d'énergie choisie est simple. Pour chaque type de clique:

$$V_c(s) = \begin{cases} -\theta \text{ si tous } s_{ij} \text{ dans c sont égaux} \\ +\theta \text{ si autrement} \end{cases} \qquad (3.9)$$

θ représente la force d'interaction et la direction d'interaction est spécifiée par le type de clique. Par la définition ci-dessus, on s'attend à une tendance d'aggrégation importante dans la direction spécifiée par le type de la clique, si la valeur de θ associé à ce type de clique est élevée.

III.2.2.2 Procédure de segmentation par programmation dynamique

Ici, l'objectif recherché est de trouver une réalisation du champ S qui maximise sa probabilité sachant les données observables du champ Y, i.e.:

$$\text{Maximiser: } p(S=s|Y=y)$$

$$\text{ou Maximiser: } p(Y=y|S=s)p(S=s) \qquad (3.10)$$

Comme discuté précédemment, la maximisation par la recherche exhaustive est impossible parce que le nombre de partitions possibles est énorme $M^{N_1 x N_2}$.

Prenons le logarithme de l'expression (3.10)

$$\ln P(Y=y|S=s) + \ln P(S=s) \qquad (3.11)$$

Parce que la formation des régions est modélisée par un champ markovien, qui est représenté par la distribution de Gibbs, le deuxième terme de (3.11) peut s'écrire:

$$\ln P(S=s) = -\ln Z - \sum_{c} V_c(S) \qquad (3.12)$$

Parce que Z est la même pour toute partition de S, le terme lnZ peut être négligé dans l'expression (3.12), et le calcul du deuxième terme peut être fait récursivement en choisissant des cliques appropriées, donc il peut être maximisé par la programmation dynamique.

Le calcul du premier terme $\ln P(Y=y|S=s)$ ne peut être ramené à une formulation récursive à cause de la difficulté du calul de la fonction de partition $Z^k(S)$.

Pour pouvoir appliquer l'approche de programmation dynamique dans la procédure d'optimisation, on se contente d'une version sous-optimale de l'algorithme.

a) L'algorithme sous-optimal pour la segmentation d'une image bruitée

Dans une image bruitée, si Y, qui est maintenant le champ des bruits, est supposé indépendant, on peut écrire:

$$\ln P(Y=y|S=s) = \sum_{k=1}^{K} \sum_{(i,j)\in R_k} \ln p(y_{ij}|s_{ij}=k) \qquad (3.13)$$

où $R_k = \{(i,j)|(i,j)\in L \text{ et } s_{ij}=k\}$

Dans ce cas, l'expression (3.11) peut être ramenée à une formulation récursive:

$$l_0 = -\ln Z \tag{3.14}$$

$$l_n = l_{n-1} - \sum_{c \in C^{n-1,n}} V_c(S) + \sum_{k=1}^{K} \sum_{(i,j) \in R_k^n} \ln p(y_{ij} | s_{ij}=k) \tag{3.15}$$

où : $C^{n-1,n} = \{c | c$ est une clique dont les pixels, sont soit entièrement dans la colonne n, soit dans la colonne n et n-1$\}$

$R_k^n = \{(i,n) | s_{in}=k$ et $1 \leq i \leq N_1\}$

La valeur de l_{N2} correspond à $p(S=s|Y=y)$.

Si l'on procède colonne par colonne, l'espace d'état associé à l'algorithme de programmation dynamique est de dimension K^{N1} parce qu'il y a K^{N1} partitions possibles pour une colonne. Ceci signifie que l'algorithme doit passer N_2 itérations et chaque itération demande un effort de calcul de l'ordre K^{N1}.

Il est à noter que le terme $\ln P(y_{ij} | s_{ij}=k)$ dans l'expression (3.13) peut être calculé en supposant que le bruit est gaussien.

$$p(y/k) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{\frac{-(y-\mu(k))^2}{2\pi\sigma^2(k)}\right\} \tag{3.16}$$

b) <u>L'algorithme sous-optimal pour la segmentation des images texturées</u>

Pour être capable d'utiliser la procédure de segmentation par programmation dynamique, le premier terme de l'expression (3.11) est

remplacé par son approximation:

$$\ln P(Y=y|S=s) = \sum_{(i,j)\epsilon L} P(Y_{\upsilon_{ij}}=y_{\upsilon_{ij}}|s_{ij}=k) \qquad (3.17)$$

où υ_{ij} est le voisinage du pixel i,j.

Alors, la procédure de calcul récursif est identique au cas (III.2.2.2 a)

$$l_0 = -\ln Z \qquad (3.18)$$

$$l_n = l_{n-1} - \sum_{c\epsilon C^{n-1,n}} V_c(s) + \sum_{k=1}^{K} \sum_{(i,n)\epsilon R_k{}^n} \ln P(Y_{\upsilon_{ij}}=y_{\upsilon_{ij}}|s_{in}=k) \qquad (3.19)$$

Le terme $\ln P(Y_{\upsilon_{ij}}=y_{\upsilon_{ij}}|s_{in}=k)$ peut être calculé en utilisant la distribution de Gibbs qui caractérise K types de texture. Le calcul de la fonction de partition $Z_k(s)$ est possible si la taille du voisinage υ_{ij} est petite.

III.2.2.3 Discussion

Dans l'approche d'optimisation par programmation dynamique que propose l'auteur pour la segmentation des images texturées, on remarque les points suivants:

- La formation récursive nécessaire pour l'application de la méthode de programmation dynamique oblige l'adoption d'un modèle sous-optimal pour segmenter l'image.

- Même avec l'utilisation de la version sous-optimale, l'effort de calcul représente une difficulté énorme si la taille de l'image est grande et si le nombre de niveaux de gris dans l'image est élevé. Ceci délimite l'application de la méthode dans une certaine classe d'image. Cependant les auteurs ont contribué énormément à la formulation du problème de segmentation d'image modélisée par un champ aléatoire markovien et au développement d'une procédure concrète pour réaliser l'objectif recherché.

III.2.3 Segmentation des images texturées par l'algorithme hiérarchique et l'algorithme de relaxation (Cohen et Cooper)

III.2.3.1 Modèle de l'image

Pour une image multitexturée, chaque texture dans l'image est modélisée par un champ aléatoire markovien gaussien. Soit y_r, $r=(i,j)$, le champ de texture au point r, pour une texture k donnée, la fonction de densité de probabilité conditionnelle de y_r est donnée par :

$$p(y_r \mid Y_{(r)}, \text{ classe } k) = [2\pi\sigma^2(k)]^{1/2} \exp\{-(1/2\sigma^2(k)) [y_r - \mu(k) - \sum_{v \in D_p} \beta_{r-v}(y_v - \mu(k))]^2 \qquad (3.20)$$

S'il s'agit d'une image multitexturée, chaque texture dans l'image ont un modèle markovien propre et les champs aléatoires markoviens modélisant des différents types de texture sont supposés indépendants.

Le mécanisme de formation une image multitexturée est comme si la

nature réalise d'abord la formation des régions, ensuite pour une région d'étiquette k, un processus de texture de type k est réalisé.

Revenons à l'expression (3.20), la fonction de probabilité conjointe pour la texture de type k, en supposant que le champ de texture est stationnaire, est donnée par:

$$p(Y|\text{classe } k) = [2\pi\sigma^2(k)]^{-M/2}\,|B(k)|^{1/2}\exp\{-[Y-U(k)]^t B(k)\,[Y-U(k)]/2\sigma^2(k)\} \qquad (3.21)$$

où : Y est un vecteur formé par la concaténation des données d'image associées à la texture de type k.

M est le nombre de pixels dans la région de texture k

B(k) matrice de MxM dont chaque élément b_{rv} représente la force d'interaction entre deux éléments r et v

U(k) vecteur des moyennes de M éléments = $\mu(k)[1,1,1,\ldots,1]$

$\sigma^2(k)$ la variance du champ.

Un point à remarquer ici est que toutes les variables aléatoires formant le champ sont identiques, c.-à-d. que ce sont des gaussiens de moyenne μ et de variance σ^2. Ceci est dû à la propriété stationnaire du champ des textures.

III.2.3.2 Modèle des régions

En supposant qu'il existe seulement deux types de texture dans l'image, Cohen et Cooper utilisent le modèle markovien binaire pour modéliser les régions. En spécifiant des paramètres appropriés du modèle, on peut incorporer des propriétés géométriques des régions, comme la taille des régions, la courbure des contours de chaque région, etc.

Soit s, le champ des régions modélisé par un champ markovien autobinaire et non-causal. Pour une image de taille NxN contenant 2 types de texture, le champ s a N^2 éléments $\{s_{ij}\}$, chaque élément s_{ij} prend une valeur, soit 0, soit 1, indiquant la classe de texture à laquelle le pixel appartient. La fonction de probabilité conditionnelle est exprimée comme suit:

$$p(s_r | S_{(r)}) = \frac{1}{D} \exp\{(s_r T_r | [1+\exp(T_r)]\} \qquad (3.22)$$

où T est donné par :

$$T_r = a + \sum_{v \epsilon D_p} b_{rvsv} \qquad (3.23)$$

D est une constante de normalisation.

brv représente la force d'interaction spatiale entre le pixel r et v.

III.2.3.3 Procédure de segmentation hiérarchique

L'algorithme de segmentation hiérarchique proposé par Cohen et Cooper consiste à diviser hiérarchiquement l'image en 4 à chaque

niveau. Le niveau 0 correspond à l'image initiale, le niveau 1 correspond à 4 blocs obtenus par division de l'image initiale en 4, de même pour les niveaux 2, 3, 4, ... Le processus de division est terminé quand chaque bloc contient juste un pixel.

La figure 3.4 illustre cette procédure de division.

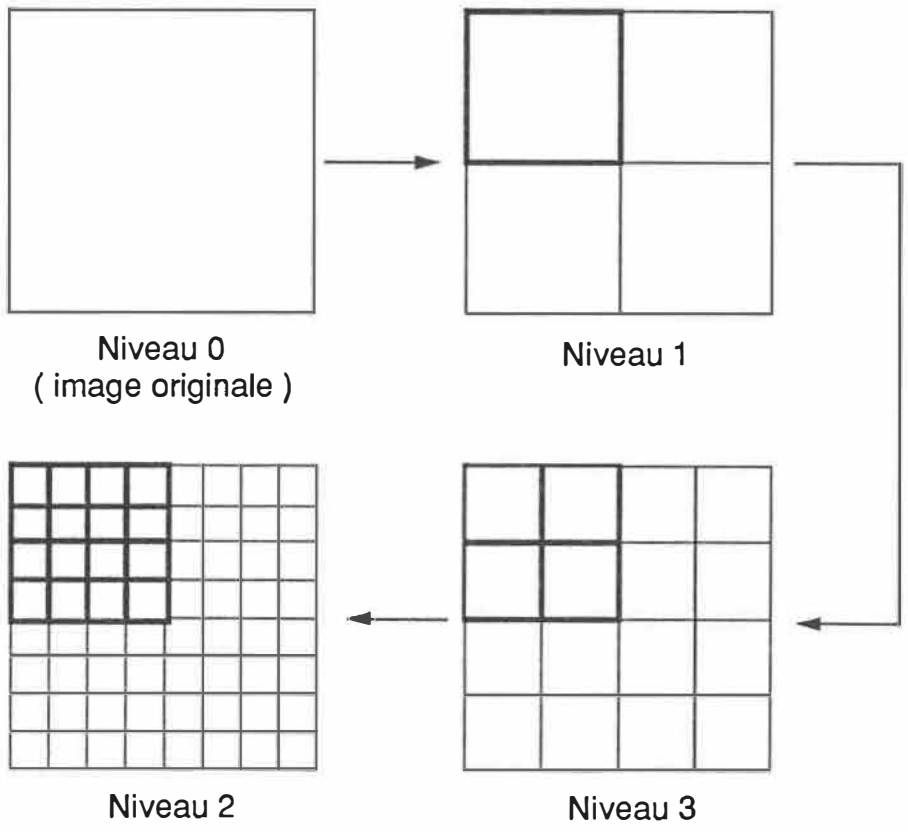


Figure 3.4 - Illustration de la division quaternaire

A chaque niveau, on cherche à déterminer la partition la plus vraisemblable. Parce qu'il est supposé y avoir deux types de texture dans l'image et parce qu'on utilise la structure quaternaire dans la division, alors, pour un bloc d'image donné, il y a $2^4=16$ partitions possibles. Dans les premières étapes de la division, on cherche à trouver une partition parmi 16 qui maximise:

$$p(Y|S) = p(Y^0|0)p(Y^1|1) \qquad (3.24)$$

où Y^k est le vecteur des données d'image pour la région de texture de type k.

Au fur et à mesure que la division progresse, pour que l'estimation soit plus exacte, la partition d'un bloc ne dépend pas seulement de l'information d'image dans ce bloc, mais aussi de l'information d'image des blocs qui l'entourent. Ceci revient à déterminer une partition d'un bloc qui maximise la probabilité des données d'image de ce bloc, conditionnée sur des blocs avoisinants:

$$p(X|Y)$$

où X est un vecteur contenant des données du bloc en question

Y est un vecteur contenant des données des blocs avoisinants.

Parce que deux champs aléatoires markoviens qui représentent deux textures sont indépendants, l'expression P(X|Y) peut s'écrire :

$$p(X|Y) = p(X^0|Y^0,0)p(X^1|Y^1,1) \qquad (3.25)$$

où X^k est un vecteur contenant des données de texture de type k dans le bloc X

Y^k est un vecteur contenant des données de texture de type k dans les blocs Y

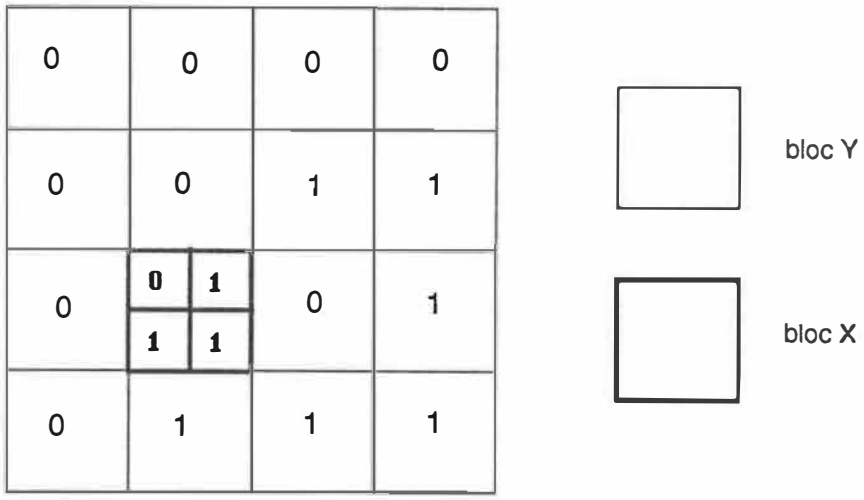


Figure 3.5 - Illustration de deux types de blocs.

a) Fonction de probabilité inconditionnelle

D'après l'expression (3.24), pour calculer $p(Y|S)$, on doit calculer $p(Y^k|k)$ où k est le type de texture, k=0,1 et Y^k est un vecteur contenant des données d'image d'un bloc assigné à l'étiquette k.

Pour une texture de type k, sans perdre la généralité, en supposant que la moyenne des niveaux de gris est nulle, l'expression (3.21) peut s'écrire:

$$p(Y|k) = (2\pi\sigma^2(k))^{-N^2/2}|B(k)|^{1/2} \exp\{-Y^t B(k)Y/2\sigma^2(k)\} \qquad (3.26)$$

où N^2 est le nombre de pixels d'un bloc contenant la texture de type k

$$|B(k)| = \prod_{i=0}^{N-1}\prod_{i=0}^{N-1}\left[1 - 2 \sum_{(l,m)\,\epsilon\,\Lambda} \beta_{lm}^k \cos\left(\frac{2\pi}{N} li + \frac{2\pi}{N} mj\right)\right] \qquad (3.27)$$

$$\Lambda = \{(1,0), (0,1), (1,1), (1,-1)\}$$

et
$$Y^t B(k)Y = \gamma^t I_W(Y) \qquad (3.28)$$

où
$$\gamma t = (1, -2\beta_{10}^k, -2\beta_{01}^k, -2\beta_{11}^k, -2\beta_{1-1}^k) \qquad (3.29)$$

et
$$I_W(Y) = (\Sigma y_{ij}^2, \Sigma y_{ij}y_{i+1,j}, \Sigma y_{ij}y_{i,j+1}, \Sigma y_{ij}y_{i+1,j+1}, \qquad (3.30)$$

$$\Sigma_{yijyi-1,j+1})^t$$

$I_B(Y_i, Y_j)$, appelé la somme d'interaction entre les blocs, est un vecteur de 4 éléments, chaque élément étant un produit de $y_r y_s$ où $r\epsilon Y_i$, $s\epsilon Yj$ et $\overrightarrow{rs}$ suit la direction déterminée par : β_{10}^k , β_{01}^k , β_{11}^k , β_{1-1}^k .

L'effort de calcul de $I_W(Y)$ et $I_B(Y_i, Y_j)$ est grandement réduit par le développement d'une formule récursive:

$$
\begin{aligned}
I_W(i,j:n-1) = & \sum_{k=0}^{1}\sum_{l=0}^{1} I_W(2i-1+k,2j-1+l;n) \\
& + I_B(2i-1,2j-1,2i-1,2j;n) \\
& + I_B(2i,2j-1,2i,2j-1;n) \\
& + I_B(2i-1,2j-1,2i,2j;n) \\
& + I_B(2i-1,2j,2i,2j-1;n) \\
& + I_B(2i-1,2j,2i,2j;n) \\
& + I_B(2i,2j-1,2i,2j;n)
\end{aligned}
\tag{3.31}
$$

$$
\begin{aligned}
I_B(i,j,i,j+1;n-1) = & I_B(2i-1,2j,2i-1,2j+1;n) \\
& + I_B(2i,2j,2i,2j+1;n) \\
& + I_B(2i-1,\ 2j,\ 2i,\ 2j+1;n) \\
& + I_B(2i,2j,2i-1,2j+1;n) \\
& + I_B(2i,2j,2i,2j+1;n)
\end{aligned}
\tag{3.32}
$$

où $I_W(i,j;n)$ est la somme d'interaction intrabloc à la position (i,j) au niveau de résolution n.

$I_B(i,j,k,l;n)$ est la somme d'interaction entre les blocs (i,j) et (l,m) au niveau de résolution n.

Le vecteur γ contient des paramètres du modèle, $I_W(Y)$ est appelé la somme d'interaction et peut être calculé en se basant seulement sur des données d'image.

Le calcul de $I_W(Y)$ demande un effort considérable. Pour réduire l'effort de calcul, Cohen et Cooper proposent la procédure de calcul suivante:

Le bloc Y est divisé en 4 cadrans Y_1, Y_2, Y_3 et Y_4 comme illustré dans la figure 3.6. La matrice B associée aux données du bloc Y peut être partionnée en:

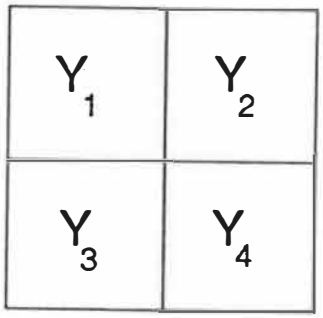


Figure 3.6 - Illustration de la division du bloc Y.

La matrice B associée aux données du bloc Y peut être partitionnée en:

$$B = \begin{bmatrix} B_1 & B_{12} & B_{13} & B_{14} \\ B_{21} & B_2 & B_{23} & B_{24} \\ B_{31} & B_{32} & B_3 & B_{34} \\ B_{41} & B_{42} & B_{43} & B_4 \end{bmatrix}$$

où B_j, $j=1,2,\ldots 4$ est la matrice d'interaction associée au bloc Y_j

B_{ij} est la matrice d'interaction entre le bloc Y_i et Y_j

Alors, l'expression (3.21) devient:

$$p(Y|k) = [2\pi\sigma 2(k)]^{-N^2/2}|B(k)|$$

$$\exp\left\{-(1/2\sigma 2(k))\left[\sum_{j=1}^{4} Y_j^t\ B_j(k)\ Y_j + 2\sum_{\substack{i,j \\ 1\le i<j\le 4}} Y_i^t B_{ij}(k) Y_j\right]\right\} \quad (3.33)$$

où
$$Y_j^t\ B_j(k) Y_j = \gamma^t I_W(Y_j) \quad (3.34)$$

et
$$Y_i^t B_{ij}(k) Y_j = \Phi^t I_B(Y_i, Y_j) \quad (3.35)$$

$$\gamma^t = \left(1,\ -2\beta^k_{10},\ -2\beta^k_{01},\ -2\beta^k_{11},\ -2\beta^k_{1-1}\right) \quad (3.36)$$

$$\Phi^t = \left(\beta^k_{10},\ \beta^k_{01},\ \beta^k_{11},\ \beta^k_{1-1}\right) \quad (3.37)$$

Les vecteurs $I_W(\cdot)$ et $I_B(\cdot,\cdot)$ pour des blocs de taille 2x2 sont calculés et stockés dans la mémoire. Ces vecteurs sont utilisés pour calculer $I_W(\cdot)$ et $I_B(\cdot,\cdot)$ pour des blocs de taille 4x4. Les $I_W(\cdot)$ et $I_B(\cdot,\cdot)$ des blocs de 4x4 sont utilisés ensuite pour calculer les $I_W(\cdot)$ et $I_B(\cdot,\cdot)$ des blocs 8x8, et le processus continue jusqu'à la taille de bloc désirée.

b) Fonction de probabilité conditionnelle

Comme on l'a mentionné précédemment, la probabilité d'une partition d'un bloc de petite taille est estimée plus exactement si elle est conditionnée par les données d'image des blocs avoisinants. Ceci est dû au fait que des blocs de petite taille pourraient contenir

seulement une portion d'un cycle des données texturales et, par conséquent, la classification ne peut être basée seulement sur les données contenues dans ce bloc.

La fonction de probabilité conditionnelle des données d'un bloc X, connaissant les données des blocs Y voisins de X, est donnée par:

$$p(X^k|Y^k) = (2\pi\sigma^2(k))^{-N^2/2}|B_X|^{1/2}\exp\{-(1/2\sigma^2(k) \left[X^t B_X(k)X + 2X^t B_{XY}(k)Y + Y^t B_{XY}(k)B_X^{-1}(k)B_{XY}(k)Y\right] \quad (3.38)$$

où $X^t B_X(k)X$ et $X^t B_{XY}(k)Y$ peuvent être calculés par la formule récursive décrite dans la section IV.2.3.3a. Le dernier terme ne peut être calculé récursivement, mais si X est de petite taille, l'effort de calcul pour ce terme est grandement réduit.

La procédure de calcul récursif ainsi obtenue permet de réduire le nombre d'opérations mathématiques d'un facteur de 10.

III.2.3.4 Procédure de segmentation par relaxation stochastique

Dans la section III.2.3.3, la segmentation des images par l'algorithme hiérarchique se base seulement les modèles de texture. L'inclusion de l'information géométrique des régions, telle que ses formes, ses tailles, etc. donne habituellement un meilleur résultat de segmentation parce que certaines contraintes spatiales de la formation

des régions sont modélisées par le modèle. Dans le cas où les contours des régions sont supposés lisses et que les courbures varient lentement, une spécification appropriée des paramètres du modèle permet d'avoir une fonction de probabilité conditionnelle à un point s_{ij}, connaissant son voisinage, ayant une basse valeur s'il y a une nette augmentation de la pente du contour à ce point, ou vice versa.

La stratégie généralement utilisée dans ce cas pour effectuer la segmentation est celle de la probabilité à posteriori maximum. L'algorithme utilisant ce critère cherche à trouver une partition qui maximise la probabilité de cette partition connaissant les données d'image:

$$\text{Maximiser:} \quad p(s=l \mid Y=y)$$

où S représente le champ aléatoire markovien modélisant la formation des régions, s est une réalisation de ce champ

Y représente le champ aléatoire markovien modélisant le processus d'intensité et y est une réalisation de Y (y correspond en fait à des données observables de l'image).

Comme décrit dans la section III.2.3.2, le modèle des régions utilisé est autobinaire et sa fonction de probabilité conditionnelle est de la forme:

$$p(s_r | S_{(r)}) = \frac{1}{D} \exp\{(s_r T_r)/[1 + \exp(T_r)]\} \qquad (3.39)$$

$$\text{où} \quad T_r = a + \sum_{U \epsilon D_p} b_{rv} S_v \qquad (3.40)$$

D est une constante de normalisation.

Pour réaliser la segmentation, les auteurs ont proposé un algorithme de relaxation qui converge comme démontré dans (Cohen et Cooper, 1987), à un maximum local. A partir de l'expression p(S/Y), on obtient:

$$p(S|Y) = \frac{p(Y|S)\ p(S)}{p(Y)} \qquad (3.41)$$

Trouver une partition qui maximise (4.41) est équivalent à trouver celle qui maximise:

$$p(Y|S)\ p(S)$$

parce que p(Y) est la même pour toutes les partitions.

On peut écrire:

$$p(Y|S)\ p(S) = p(Y|S_{(ij)},\ s_{ij} = k)\ p(S_{(ij)},\ s_{ij} = k) \qquad (3.42)$$

où $k = 0,1$.

Pour les deux termes dans le produit, on a:

$$p(S_{(ij)},\ s_{ij}) = p(s_{ij}|S_{(ij)})\ p(S_{(ij)}) \qquad (3.43)$$

Parce que le champs aléatoire des régions et celui des processus d'intensité sont supposés indépendants, on peut écrire:

$$p(Y|S_{(ij)}, s_{ij}=k)=p(y_{ij}|Y_{(ij)}, S_{ij}, s_{ij}=k)\ p(Y_{ij}|s_{(ij)}, S_{ij}=k) \quad (3.44)$$

L'algorithme

1/ Pour l'itération n:

* Pour tout pixel (i,j) dans l'image

* Choisir l'état de s_{ij} (=0 ou 1) qui maximise:

$$p(Y|S_{(ij)}, s_{ij}=k)\ p(S_{(ij)}, s_{ij} = k)$$

ou, selon (3.43) et (3.44), qui maximise:

$$p(y_{ij}|Y_{(ij)}, S_{(ij)}, s_{ij}=k)\ p(Y_{(ij)}|S_{(ij)}, s_{ij}=k)\ p(s_{ij}|S_{(ij)}\ p(S_{(ij)})$$

parce que:

$p(Y_{(ij)}|S_{(ij)}, s_{ij}=k)$ et $p(S_{(ij)})$ ne dépendent pas de l'état de s_{ij},

donc l'état de pixel s_{ij} est choisi de telle façon que:

$p(y_{ij}|Y_{(ij)}, S_{(ij)}, s_{ij}=k)\ p(s_{ij}|S_{(ij)})$ est maximisée.

Le premier terme du produit est calculé par (3.20) et le deuxième par (3.22).

2/ Comparer l'état des régions de l'itération n avec celui de n-1.

Si ils sont proches:

- Fini

Sinon

- n=n+1

- Retourner à 1

Par approximation, on peut utiliser l'état des régions résultant de la segmentation par stratégie de vraisemblance maximum, comme estimation pour la première itération (n=1).

III.2.3.5 Discussion

* La méthode de segmentation hiérarchique, par ses structures de données propres (arbre quaternaire), permet de limiter la taille, l'espace de recherche en un nombre très réduit (=16) pour chaque niveau (en supposant qu'il y a deux types de région). Pour une image de taille NxN, on a H niveaux hiérarchiques ($N=2^H$), la taille de l'espace de recherche peut être calculée par la formule:

$$T = \sum_{h=1}^{H} 2^{2(h-1)}$$

tandis que, pour l'approche de calcul exhaustif:

$$T = 2^{N^2}$$

Exemple: N=64

Par la méthode hiérarchique: T = 21840

Par la méthode de calcul exhaustif: $T' = 2^{64^2} \gg T$

* Dans la procédure hiérarchique de segmentation, le calcul de la fonction de probabilité inconditionnelle (III.2.3.3a) au niveau n ne se sert pas du résultat de la partition obtenue au niveau n-1. Par exemple, au niveau n-1, le bloc i est assigné

à la région de type k, la partition optimale du bloc i au niveau n ne dépend que des données d'image dans le bloc i et ne dépend pas de l'étiquette que le bloc i reçoit au niveau 1. Ceci représente une perte considérable de l'information.

Pour le calcul de la fonction de probabilité conditionnelle (III.2.3.3b), l'information sur la partition au niveau (n-1) n'intervient que partiellement au niveau n, parce que l'information sur le type de région du bloc i en question, au niveau n-1 est ignorée au niveau n mais l'information sur les types de région des blocs j ($j \neq i$) au niveau n-1 est incluse au niveau n.

III.3 Segmentation non-supervisée des images texturées par estimation baysienne; méthode proposée

III.3.1 Introduction

Dans les deux sections précédentes, nous avons fait un survol des méthodes typiques, rapportées récemment dans la littérature, pour segmenter une image modélisée par un champ aléatoire markovien 2-D. Ces méthodes ont contribué considérablement au développement des modèles ainsi qu'aux procédures à suivre pour réaliser la segmentation. Les résultats obtenus sont parmi les meilleurs malgré les défauts dont nous avons discuté en détail.

Dans cette section, nous présentons notre méthode de segmentation des images texturées. Cette méthode se base aussi sur le modèle markovien pour modéliser l'image et les régions, et la segmentation est aussi réalisée par une estimation statistique. L'originalité de notre méthode réside dans le choix approprié d'une fonction d'énergie associée à un type de clique qui, par sa disposition géométrique, permet d'englober le comportement des autres cliques et qui, plus important, permet de convertir, par la transformation de Hadamard, les observations originales en des caractéristiques locales décorrelées. La conversion simplifie considérablement l'expression de la distribution de Gibbs associée à chaque texture et conduit à la décomposition du problème de segmentation en des décisions stochastiques locales basées sur des fenêtres d'estimation de taille limitée. L'approche proposée s'adresse aussi au problème le plus difficile de la segmentation, celui de l'estimation des paramètres du modèle directement sur l'observation. Certaines méthodes utilisent une procédure adaptative de segmentation dans laquelle les paramètres du modèle sont estimés au fur et à mesure que la segmentation progresse. Cette procédure demande un énorme effort de calcul puisque plusieurs itérations doivent se faire pour raffiner les paramètres. Notre approche propose une méthode d'apprentissage basée sur l'extraction des régions homogènes par la technique d'aggrégation floue. Cette méthode s'est montrée efficace et souple dans l'apprentissage des paramètres du modèle.

Dans les sections qui suivent, nous allons présenter le modèle de l'image et des régions dans la section III.3.2, celle-ci inclut le

choix du système de voisinage et de la fonction d'énergie, ainsi que la transformation de Hadamard. Dans la section III.3.3, nous discuterons des procédures d'estimation basées sur différentes stratégies de segmentation et, finalement, à la section III.3.4, sera présentée notre procédure d'estimation des paramètres.

III.3.2 Modèle de l'image

III.3.2.1 Modèle hiérarchique

Nous adoptons le modèle gibbsien hiérarchique de Derin et Cole. Dans ce modèle, l'image observée est une réalisation conjointe de plusieurs champs de textures qui représentent les différents types de textures. Cette réalisation suppose que le processus de formation des régions précède celui des textures, d'où vient le mot hiérarchique.

Soit L la maille de taille $N_1 x N_2$ sur laquelle un champ aléatoire markovien $Y=\{y_{ij}\}$ est défini:

$$L = \{(i,j) \mid 1 \leq i \leq N, \quad 1 \leq j \leq N_2\} \tag{3.45}$$

Le champ $Y = \{y_{ij}\}$ est une combinaison des champs de textures $T^k = \{t^k_{ij}\}$, k=1,2... K. Une réalisation du champ $Y = \{y_{ij}\}$ est définie sur une réalisation du champ $S = \{s_{ij}\}$ qui représente le processus de formation des régions. Le champs S prend des valeurs dans l'ensemble:

$$Q = \{1,2... K\}$$

Le modèle hiérarchique est construit comme suit:

- A chaque emplacement (i,j), la valeur du pixel est égale à un échantillon du processus de texture T_k oy k est la valeur du champ S à cet emplacement.

III.3.2.2 Modèle des textures

a) Fonction d'énergie

Nous supposons que le processus d'intensité d'une texture de type k, T^k, sur la maille L suit le modèle de Gibbs dont la fonction de distribution est définie comme:

$$p(t^k) = \frac{1}{D^k} \exp\{-U(t_k)\} \qquad (3.46)$$

où t^k est une réalisation du champ T^k

D^k est une constante de normalisation $\sum_{\forall t^k} e^{-U(t^k)}$

$U(t^k)$: l'énergie associée à la réalisation t^k de T^k.

Pour la texture de type k, représentée par T^k, les réalisations de T^k ont comme domaine:

$$G^k = \left\{ g_1^k, g_2^k \ldots g_{q_k}^k \right\}$$

où q_k représente le nombre de niveaux de gris dans la texture de type k.

La fonction d'énergie $U(t^k)$ peut être exprimée comme:

$$U(t^k) = \sum_{(i,j)\in L} U(t^k_{ij}) \qquad (3.47)$$

où $U(t^k_{ij})$ est l'énergie associée à une réalisation $t^k_{ij} \in G^k$.

Avec le système de voisinage et la notion de clique présentés dans la section III.1.4.1. a, on peut exprimer:

$$U(t^k_{ij}) = \sum_{C} U_C(t^k_{ij}) \qquad (3.48)$$

où C est l'ensemble des types de clique associés au système de voisinage choisi.

Pour un système de voisinage d'ordre 2, on a des types de clique illustrés ci-dessous:

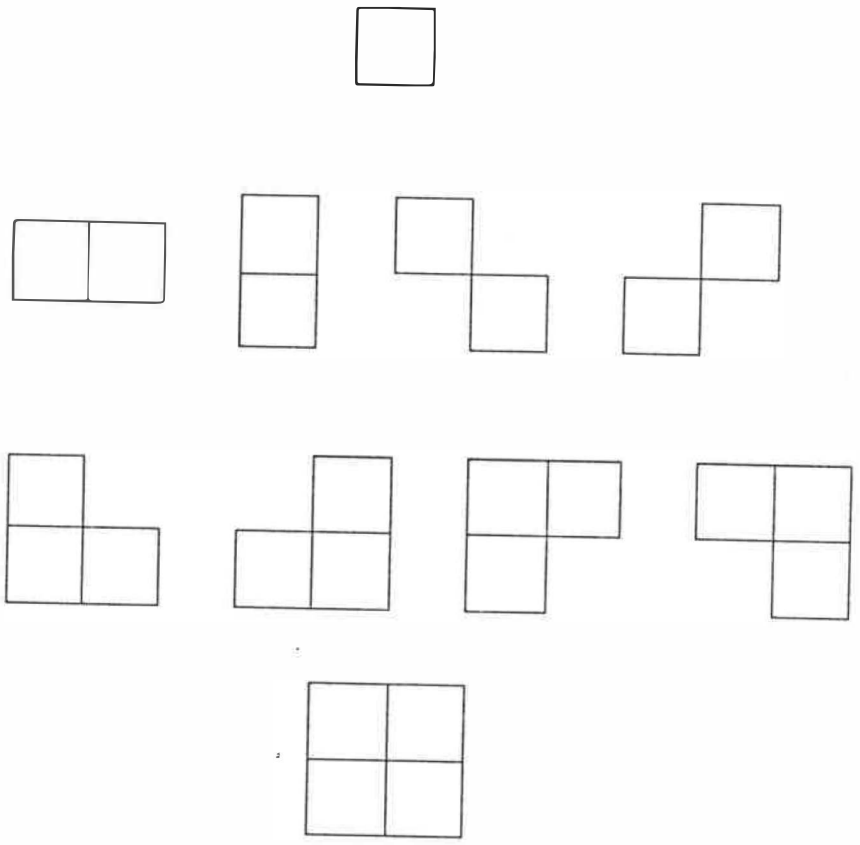


Figure 3.7 - Types de clique du système de voisinage d'ordre 2.

La valeur de $U_C(t^k_{ij})$ pour chaque type de clique représente l'énergie associée à ce type de clique pour une texture de type k donnée. Donc, les différents types de texture sont représentés par les différentes valeurs associées à un certain type de clique. Derin et $_k$ Cole ont défini $U_C(t_{ij})$ comme suit:

$$U_C(t^k_{ij}) = \begin{cases} -\theta, \text{ si tous } y_{ij} \text{ dans C sont égaux} \\ \theta \text{ autrement} \end{cases} \tag{3.49}$$

où $$\theta \in [\alpha^k, \beta^k_1, \beta^k_2 \ldots \gamma^k_1, \gamma^k_2 \ldots, \xi_1\} \tag{3.50}$$

Le paramètre α^k contrôle le pourcentage des pixels ayant un certain niveau de gris g^k_i, i ϵ {1,2... q_k}. Ceci signifie, pour un niveau de gris g^k d'une texture de type k, que α^k va prendre une valeur α^k_i qui contrôle le pourcentage des pixels ayant ce niveau de gris.

Les cliques de deux pixels et plus, représentées par β^k_1, $\beta^k_2 \ldots \beta^k_4$ γ^k_1, $\gamma^k_2 \ldots \gamma^k_4$, ξ^k_1, contrôlent, par la définition de (3.49), des aggrégations des niveaux de gris dans différentes directions.

Dans notre approche, l'énergie associée à un type de clique C est définie comme:

$$U_C(t_{ij}) = \sum_{c \epsilon C} - \ln[p(\tau_c)], \ \forall (i,j) \in L \tag{3.51}$$

où: $p(\tau_c)$ est la probabilité conjointe d'observer au pixel (i,j) une configuration τ_c de la clique de type C. S'il y a q niveaux de gris dans une texture, une clique de m pixels donne q^m configurations possibles de τ_c.

Cette définition relie la notion de l'énergie d'un type de clique à la notion de désordre local dans le champ de probabilité de cette

clique. Elle prend l'origine dans la définition d'entropie de Shannon (1948).

Considérons une distribution discrète de probabilité:

$$p_i \geq 0, \quad i=1,2\ldots n$$

$$\Sigma p_i = 1$$

L'entropie correspondante est la quantité.

$$H_n = H_n(p_1, p_2 \ldots p_n) = -\sum_{k=1}^{n} p_k \log p_k \tag{3.53}$$

Si nous associons p_i, $i=1,2\ldots n$ à des événements a_1, $a_2 \ldots a_n$ d'une expérience, selon la théorie de l'information, on peut définir:

$$I(a_k) = -\log p_k \tag{3.54}$$

où $I(a_k)$ représente la quantité d'information associée à l'événement a_k.

L'entropie H_n définit la quantité d'information moyenne obtenue à partir d'une expérience, elle atteint le maximum quand tous les p_k sont égaux, soit dans le cas où l'expérience ne fournit aucune information.

Revenons maintenant à la fonction d'énergie définie dans (3.51). Pour simplifier, nous assumons que l'énergie locale est associée uniquement à un type de clique C*. Les différentes configurations de C*, τ_C, définissent un champ de probabilité τ_C^n formé par n configurations τ_{C*}^1, τ_{C*}^2 ... τ_{C*}^n où $n = q^m$, m: le nombre de pixels dans la clique C*, q: nombre de niveaux de gris. Le modèle de Gibbs d'une

texture k est représenté dans notre approche par le logarithme de la fonction de densité de probabilité $p(\tau^i_{C*})$, i=1,2...n. Donc, selon notre modèle, la structure de la fonction de densité de probabilité $p(\tau^i_{C*})$, i=1,2... n définit un type de texture. Evidemment, pour être plus général, on doit tenir compte de tous les types de clique définis sur un voisinage déterminé.

La correspondance établie entre la fonction d'énergie et la densité de probabilité des types de clique est justifiée par le raisonnement suivant:

- Supposons qu'une texture de type k représentée par la fonction de densité $p(\tau^k_{C*})$ qui prend la forme:

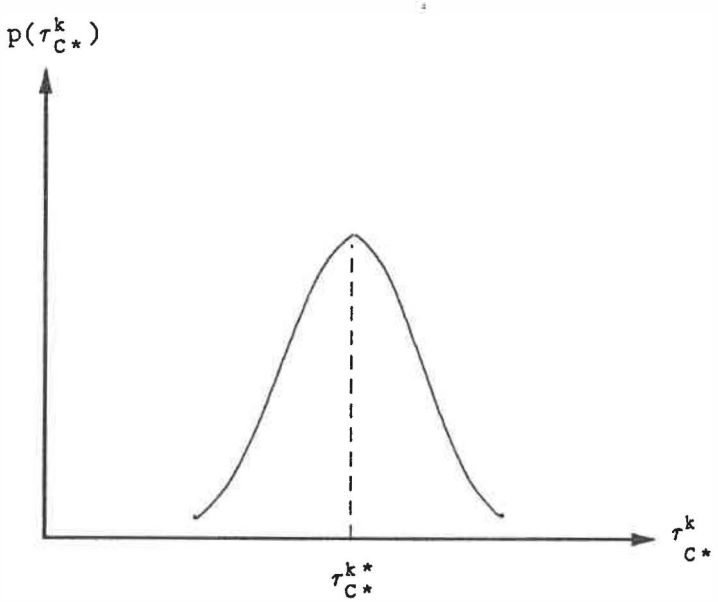


Figure 3.8 - Fonction de densité de τ^k_{C*}.

Plus la forme de la fonction de densité est concentrée, plus l'apparence de la texture k est déterminée par la configuration r_{C*}^{k*} parce que sa fréquence est fortement dominante. La fonction d'énergie définie en (3.51) favorise les réalisations t^k de T^k dont les fonctions de probabilité ont la même structure.

- Si la forme de la fonction de densité $p(r_{C*}^{k})$ est étendue, ceci signifie qu'aucune configuration n'est dominante, les réalisations t^k et T^k se présentent moins riches en termes de texture dans le sens donné par l'entropie: plus l'entropie est élevée, moins l'information texturale est abondante.

Les arguments mentionnés ci-dessus donnent un sens à la définition de la fonction d'énergie. De plus, notre définition permet de choisir le type de clique de taille 2x2 comme une clique unique dans la représentation de la fonction d'énergie, parce qu'il couvre toutes les caractéristiques contrôlées par les autres cliques. Ce n'est pas le cas dans la définition de la fonction d'énergie de Derin et Cole.

Revenons à l'expression (3.48), on peut écrire

$$U(t^k) = -4 \sum_{g_A} \sum_{g_B} \sum_{g_C} \sum_{g_D} N(g_A, g_B, g_C, g_D) \, \mathrm{Ln}[p^k(g_A, g_B, g_C, g_D)] \quad (3.55)$$

où g_A, g_B, g_C, g_D sont des niveaux de gris des pixels A, B, C et D dans une clique 2x2.

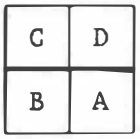


Figure 3.9 - Clique 2x2.

$p^k(g_A, g_B, g_C, g_D)$ représente la probabilité conjointe d'occurence de la configuration particulière (g_A, g_B, g_C, g_D) dans une clique 2x2.

et $N(g_A, g_B, g_C, g_D)$ désigne le nombre de cliques ayant la même configuration dans la réalisation t^k.

A ce point, l'expression (3.55) peut être simplifiée davantage par l'introduction d'une transformation qui a pour but de décorreler les variables locales correspondant aux positions ponctuelles des cliques de 2x2.

b) Transformation de Hadamard

Pour simplifier la notation, on définit les variables aléatoires correspondant à la clique 2x2 comme suit:

$$A = T^k_{ij} \quad , \quad B = T^k_{i-1,j} \quad , \quad C = T^k_{i-1,j-1} \quad , \quad D = T^k_{i,j-1}$$

Les 4 variables aléatoires A, B, C, D sont assurément correlées entre elles à cause de leur disposition géométrique. Nous cherchons, à partir des variables aléatoires A, B, C, D, quatre autres variables qui

sont non correlées. La transformation de Hadamard permet de fournir une solution à ce problème. Cette transformation trouve les axes principaux des variables A, B, C, D en se basant sur les vecteurs propres de la matrice de covariance associés aux variables A, B, C, D.

C	D
B	A

$$R = \sigma^2 \begin{bmatrix} 1 & \varphi_{AB} & \varphi_{AC} & \varphi_{AD} \\ \varphi_{BA} & 1 & \varphi_{BC} & \varphi_{BD} \\ \varphi_{CA} & \varphi_{CB} & 1 & \varphi_{CD} \\ \varphi_{DA} & \varphi_{DB} & \varphi_{DC} & 1 \end{bmatrix} \qquad (3.56)$$

Si on suppose que:

$$\varphi_{AB} = \varphi_{CD}$$

$$\varphi_{BC} = \varphi_{AD}$$

on peut écrire:

$$R = \sigma^2 \begin{bmatrix} 1 & \varphi_{10} & \varphi_{11} & \varphi_{01} \\ \varphi_{10} & 1 & \varphi_{01} & \varphi_{11} \\ \varphi_{11} & \varphi_{01} & 1 & \varphi_{10} \\ \varphi_{01} & \varphi_{11} & \varphi_{10} & 1 \end{bmatrix} \qquad (3.57)$$

où φ_{10} : coefficient de corrélation dans la direction horizontale

φ_{01} : coefficient de corrélation dans la direction verticale

φ_{11} : coefficient de corrélation dans la direction diagonale ou anti-diagonale.

$$\rho_{mn} = E\left\{ \left[T^k_{1+m,\,j+n} - \mu^k \right] \left[T^k_{ij} - \mu^k \right] \right\} / \sigma^2 \qquad (3.58)$$

$$\text{où} \quad \mu^k = E\left[T^k_{ij}\right] \quad \text{et} \quad \sigma^2 = E\left\{\left[T^k_{ij} - \mu^k\right]^2\right\} \tag{3.59}$$

Comme la matrice R est de forme diadique, elle est diagonalisable. Les 4 vecteurs propres de R définissent alors une transformation réversible des variables A, B, C et D en un ensemble de 4 nouvelles variables mutuellement décorrelées S_1, S_2, S_3 et S_4, dont les variances sont les 4 valeurs propres λ_1, λ_2, λ_3 et λ_4, respectivement.

On trouve:

$$\begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} \tag{3.60}$$

$$\begin{aligned} S_1 &= A + B + C + D \\ S_2 &= -A - B + C + D \\ S_3 &= -A + B + C - D \\ S_4 &= -A + B - C + D \end{aligned} \tag{3.61}$$

Par conséquent, les nouvelles variables S_1, S_2, S_3, S_4 sont obtenues en convoluant 4 masques orthogonaux de 2x2 sur l'image originale.

1	1
1	1

H_1

1	1
-1	-1

H_2

1	-1
1	-1

H_3

-1	1
1	-1

H_4

Figure 3.10 - Masques orthogonaux.

On remarque que le masque H_1 représente la moyenne locale des réalisations de T^k, H_2 l'information de la dérivée première en direction verticale, H_3 l'information de la dérivée première en direction horizontale et H_4 celle de la dérivée non-directionnelle du deuxième ordre. Les travaux de Cohen, Le et Lacasse (1988) ont montré que les variables décorrelées S_1, S_2, S_3 et S_4 sont très puissantes dans la discrimination des textures naturelles.

Finalement, si les variables A, B, C, D sont gaussiennes, S_1, S_2, S_3 et S_4 le sont également, et sont donc indépendantes. Revenons maintenant à l'expression (3.55), on peut alors écrire:

$$U(t^k) = -4 \sum_{n=1}^{4} \sum_{s_{nq}} N(s_{nq}) \, \mathrm{Ln} \left[p_n^k(s_{nq}) \right] \qquad (3.62)$$

où $S_n = (S_{n1}, S_{n2} \ldots S_{nq})$ représente l'espace de réalisation de la variable S_n.

$p_n^k(s_{nq})$ représente la fonction de distribution de la variable S_n

$N(s_{nq})$ est le nombre de cliques 2x2 qui donne $S_n = s_{nq}$

III.3.2.3 Modèle du processus de formation des régions

Dans notre approche, le champ de partition X est aussi modélisé par un champ aléatoire markovien sur la maille L avec le système de voisinage d'ordre 2. Donc X peut être représenté par une distribution de Gibbs:

$$p(X=x) = \frac{1}{F} \exp\left[-W(x)\right] \tag{3.63}$$

où F: constante de normalisation = $\sum_{\forall x} e^{-W(x)}$

W(x): fonction d'énergie associée à la réalisation x

On peut écrire:

$$W(x) = \sum_{(i,j)\epsilon L} \sum_{c\epsilon C} W_c[x_{ij}] \quad \text{pour } \forall \text{ c telle que } (i,j) \epsilon \text{ c} \tag{3.64}$$

Considérons de nouveau les cliques de 2x2 comme dans la figure 3.9, on définit que la fonction d'énergie locale comme suit:

$$W_c[x_{ij}] = \beta_1 \delta_{AB} + \beta_2 \delta_{AD} + \gamma \delta_{AC} \tag{3.65}$$

où δ_{UV} est défini comme:

$$\delta_{UV} = \begin{cases} -1 \text{ si } U = V \\ 1 \text{ si } U \neq V \end{cases} \tag{3.66}$$

Chaque terme dans (3.65) représente le modèle de la forme des régions suivant les différentes directions:

- δ_{AB} modélise l'agrégation des régions dans la direction horizontale, $|\beta_1|$ représente la force de l'agrégation. La valeur positive de β_1 favorise l'agrégation dans la direction horizontale, tandis que la valeur négative de β_1 défavorise l'agrégation dans cette direction.

- δ_{AD} et β_2 ont la même signification pour la direction verticale et δ_{AC}, γ pour la direction diagonale et anti-diagonale.

La combinaison linéaire de δ_{AB}, δ_{AC} et δ_{AD} dans (3.65) permet de modéliser la géométrie des régions de n'importe quelle forme en spécifiant des valeurs appropriées pour des paramètres β_1, β_2 et δ.

On peut exprimer P(x) dans (3.63) sous la forme de pseudo-vraisemblance

$$p(X=x) = \prod_{(i,j)} p[x_{ij} | x(\vartheta_{ij})] \qquad (3.67)$$

où $x(\vartheta_{ij})$ représente la réalisation du champ aléatoire X dans le voisinage ϑ_{ij} de (i,j)

Des classes de processus markoviens permettant une telle représentation ont été étudiées par Besag (1974). Ici nous choisissons une forme particulière:

$$p[x_{ij} | x(\vartheta_{ij})] = \frac{1}{G} e^{-\nu_{ij}} \qquad (3.68)$$

où ν_{ij} représente la fonction d'énergie locale. Si ϑ_{ij} est un voisinage d'ordre 2 (fig. 3.11) l'expression de ν_{ij} peut être obtenue à partir de (3.69):

$$\nu_{ij} = 2\beta_1(\delta_{AH} + \delta_{AD}) + 2\beta_2(\delta_{AB} + \delta_{AF}) + \gamma(\delta_{AC} + \delta_{AI} + \delta_{AE} + \delta_{AG}) \qquad (3.69)$$

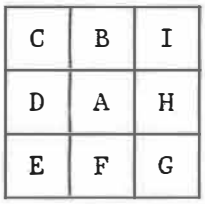


Figure 3.11 - Voisinage d'ordre 2.

III.3.3 Procédure d'estimation

Nous utilisons deux procédures d'estimation présentées en détail dans la section III.2.1: l'une se base sur la stratégie du maximum de vraisemblance et l'autre sur la stratégie du maximum à posteriori.

III.3.3.1 Stratégie du maximum de vraisemblance

Pour une image donnée, sous la forme d'un tableau d'observations $y=\{y_{ij}\}$, nous cherchons l'estimation

$$\hat{x} = \{\hat{x}_{ij}\}$$

qui maximise:

$$p(y|\hat{x})$$

Comme le processus de régions X et celui de texture T^k sont représentés par des champs aléatoires markoviens mutuellement indépendants, l'estimation globale peut être fractionnée en des estimations locales à condition qu'une fenêtre d'analyse appropriée soit définie.

Soit ω_{ij} une fenêtre d'analyse de taille (2M+1) x (2M+1), centrée au pixel (i,j)

$$\omega_{ij} = \{(u,v) \mid i-M \leq u \leq i+M,\ j-M \leq v \leq j+M\} \tag{3.70}$$

La taille de ω_{ij} doit être choisie de façon qu'elle soit suffisamment petite pour que, dans la plupart des cas, elle contienne seulement un type de texture, et suffisamment grande pour que les modèles de texture soient encore applicables avec fiabilité dans toute fenêtre.

Supposons qu'une fenêtre d'analyse de taille appropriée ait été définie. On peut décomposer:

$$p(y|\hat{x}) = \prod_{i,j} p(y(\omega_{ij})|\hat{x}(\omega_{ij})) \tag{3.71}$$

où $y(\omega_{ij})$ et $x(\omega_{ij})$ désignent l'observation et la partition dans la fenêtre ω_{ij}.

Comme, en général, ω_{ij} contient juste un type de texture, le terme $p[y(\omega_{ij})|\hat{x}(\omega_{ij})]$ désigne la probabilité d'observer $y(\omega_{ij})$ dans le processus de texture $T^{\hat{x}_{ij}}$. A partir des expressions (3.46) et (3.55), on peut écrire:

$$\mathrm{Ln}\{p[y(\omega_{ij})|\hat{x}(\omega_{ij})] = -\ln D_{\hat{x}_{ij}} + \sum_{n=1}^{4} \sum_{s_{nq}} N(s_{nq}) \mathrm{Ln}\left[p_n^{\hat{x}_{ij}}(s_{nq})\right] \tag{3.72}$$

où $N(s_{nq})$ désigne le nombre de cliques dans ω_{ij} avec $s_n = s_{nq}$.

Finalement, la procédure d'estimation par maximum de vraisemblance consiste à choisir, pour tout pixel (i,j), la texture de type k qui maximise la fonction de vraisemblance

$$\lambda_{ij}(k) = -\mathrm{Ln}D_k + \sum_{n=1}^{4} \sum_{s_{nq}} N(s_{nq}) \, \mathrm{Ln}\left[\hat{p}_n^{x_{ij}}(s_{nq})\right] \qquad (3.73)$$

En général, l'évaluation de D_k est difficile. Si on suppose que D_k est constant, on a alors:

$$\lambda'_{ij}(k) = \sum_{n=1}^{4} \sum_{s_{nq}} N(s_{nq}) \, \mathrm{Ln}\left[\hat{p}_n^{x_{ij}}(s_{nq})\right] \qquad (3.74)$$

III.3.3.2 Stratégie du maximum à posteriori

$$p(\hat{x}|y)$$

Ceci équivaut à maximiser:

$$p(y|\hat{x}) \; p(\hat{x})$$

Revenons aux expressions (3.67) et (3.71), nous pouvons écrire:

$$p(y|\hat{x}) \; p(\hat{x}) = \prod_{(i,j)} p[y(\omega_{ij})|x_{ij}] \; p[x_{ij}|x(\upsilon_{ij})] \qquad (3.75)$$

En combinant (3.69) et (3.74), la procédure d'estimation consiste à choisir au pixel (i,j) la texture de type k qui maximise la fonction:

$$\mu_{ij}(k) = \lambda'_{ij}(k) + \nu_{ij} \qquad (3.76)$$

III.3.4 Estimation des paramètres

III.3.4.1 Introduction

Dans notre approche, quelle que soit la stratégie utilisée, la connaissance des distributions de probabilité $\{p_n^k(\cdot)\}$ correspondant aux variables S_n de chaque type de texture k, est nécessaire.

Dans l'estimation supervisée, on suppose que ces distributions sont connues à priori, ce qui signifie qu'un apprentissage des types de textures présentes dans l'image à segmenter a été effectué avant la procédure d'estimation. Cependant, ce n'est pas toujours le cas dans la réalité, les textures présentes dans l'image ne sont pas disponibles avant la segmentation, donc le modèle de chaque type de texture doit être estimé directement à partir de l'image observée: c'est le cas de la segmentation non supervisée.

On peut réaliser la segmentation non supervisée par la technique adaptative de segmentation dans laquelle les modèles sont estimés et raffinés au fur et à mesure que le traitement progresse. Cette méthode demande habituellement un énorme effort de calcul et est très sensible au bruit.

Dans nos travaux, l'estimation des paramètres est faite par la méthode d'agrégation floue hiérarchique qui permet de déterminer le nombre de textures et d'identifier, en même temps, les régions occupées

par ces textures dans l'image observée. Chaque région homogène est supposée contenir un type de texture et l'établissement du modèle sera fait sur cette région.

L'agrégation est une technique pour partitionner l'ensemble des données discrètes représentées dans l'espace des caractéristiques. Ceci revient à dire que chaque élément de l'ensemble est représenté par un vecteur de caractéristiques associé à cet élément et, par conséquent, il peut être représenté par un point dans l'espace des caractéristiques. Les techniques d'agrégation permettent, à partir de cette représentation, de regrouper des éléments de caractéristiques semblables en des agrégats. Chaque agrégat contient des éléments qui sont considérés comme homogènes par rapport aux caractéristiques utilisées.

Dans un certain sens, le problème d'agrégation peut être posé comme un problème de recherche dans lequel une partition optimale doit être trouvée dans l'espace des partitions. Pour un ensemble de données $X = \{x_1, x_2 \ldots x_n\}$ où n est le nombre d'éléments, si on suppose que c agrégats existent dans l'ensemble, le nombre de partitions possibles est c^n. Donc, la taille de l'espace de recherche croît exponentiellement avec le nombre d'éléments.

Soit $\mathfrak{Z}$ l'espace de partitions, on définit sur $\mathfrak{Z}$ une application H qui prend l'ensemble des images dans R:

$$H: \mathfrak{Z} \longrightarrow R$$

La partition optimale est celle qui est le maximum de

$$H(p),\ \forall\ p \in \mathfrak{S}$$

On appelle H le critère d'agrégation. La définition formelle de l'espace de partition sera présentée dans la section suivante.

III.3.4.2 Agrégation floue

Définition 1: c-Partition dure

Soit $X = \{x_1, x_2 \ldots x_n\}$ un ensemble fini et V_{cn} un ensemble de matrices réelles c x n où c est un entier, $2 \leq c \leq n$. L'espace de c partition pour X est l'ensemble

$$M_c = \{ U \in V_{cn} / u_{ik} \in \{0,1\}\ \forall_{i,k};\ \sum_{i=1}^{c} u_{ik} = 1\ \forall_k;\ 0 < \sum_{k=1}^{n} u_{ik} < n\ \forall_i \}$$

Par cette définition:

U est appelée matrice de partition

u_{ik} est l'appartenance de l'élément k à la classe i (l'agrégat i)

$u_{ik} = 1 \iff k \in$ classe i

Définition 2: c-Partition floue

Soit $X = (x_1, x_2 \ldots x_n)$ un ensemble fini. V_{cn} est un ensemble de matrices réelles de taille c x n. L'espace de c-partition pour X est l'ensemble

$$M_{fc} = \{U \in V_{cn} \mid u_{ik} \in [0,1] \ \forall_{i,k};\ \sum_{i=1}^{c} u_{ik} = 1 \ \forall_k;\ 0 < \sum_{k=1}^{n} u_{ik} < n \ \forall_i\}$$

Comme on a remarqué, u_{ik} est continue dans [0,1], donc, l'appartenance d'un élément à une classe n'est plus de nature déterministe mais est plutôt exprimée en degré d'appartenance. Cette souplesse de la notion de l'ensemble flou permet d'avoir une représentation plus appropriée des phénomènes aléatoires.

Par agrégation floue, on entend une technique qui cherche à trouver une partition floue optimale dans l'espace de partition. Une partition optimale est celle qui extrémise un certain critère. Les critères d'optimisation font l'objet de la section suivante.

III.3.4.3 Optimisation par des fonctions objectives

a) Fonctionnelles de densité

Ces fonctionnelles proposées par Ruspini (1970) sont définies comme suit:

$$J_R(U) = \sum_{j=1}^{n} \sum_{k=1}^{n} \left\{ \left[\sum_{i=1}^{c} \sigma(u_{ij} - u_{ik})^2 \right] - d^2{}_{jk} \right\} \qquad (3.77)$$

où n : nombre d'éléments dans l'ensemble des données X

c : nombre de classes, $2 \leq c < n$

σ : constante

d_{ij} : distance entre l'élément i et j dans l'espace des caractéristiques.

Ruspini interprète J_R comme mesure de la qualité de l'agrégation basée sur la densité locale, parce que J_R est petit quand chaque terme dans (3.77) est petit.

La partition floue optimale de X est celle qui donne le minimum local de J_R. Ruspini (1970) a proposé un algorithme pour trouver la partition désirée.

b) Fonctionnelles du moindre carré

Dunn (1974) a proposé une fonctionnelle qui généralise la fonction de la somme de carrés des erreurs intra-groupe:

$$Jm(U,V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m \, d^2_{ik} \qquad (3.78)$$

où U : une partition floue de $X = \{x_1, x_2 \ldots x_n\}$

V : $(V_1, V_2, V_3 \ldots V_c)$ ensemble des centres d'agrégats

d_{ik} : $\|x_k - V_i\|$

m : exposant de pondération, $m \in]1, \infty[$

Parce que chaque terme de J_m est proportionnel à d_{ik}^2, J_m est appelé le critère d'agrégation au moindre carré.

Une partition floue optimale de X est celle qui minimise:

$$J_m(U,V)$$

Le théorème suivant permet de trouver une telle partition

Théorème

Pour $m \in]1,\infty[$, $(U,V)^*$ est un minimum global pour J_m si:

$$u_{ik} = 1/\left[\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{m-1}} \right] \qquad (3.79)$$

$$V_i = \sum_{k=1}^{n} (u_{ik})^m x_k / \sum_{k=1}^{n} (u_{ik})^m \quad , \quad \forall i \qquad (3.80)$$

La solution analytique pour trouver $(U,V)^*$, basée sur le théorème ci-dessus, est impossible. Bezdek (1981) a proposé un algorithme itératif qui permet de trouver une partition correspondante au minimum local de J_m.

Algorithme (Fuzzy c-Means, Bezdek)

1. Choisir c, $2 \leq c < n$, $m, 1 \leq m < \infty$. Initialiser $U^{(0)}$. Puis, à l'étape l, $l = 0,1,2...$

2. Calculer c centres d'agrégats $\{V_i^{(l)}\}$ par (3.80) en utilisant $U^{(l)}$.

3. Calculer $U^{(l+1)}$ par (3.79), en utilisant $(V_i^{(l)})$.

4. Comparer $U^{(l)}$ et $U^{(l+1)}$ par une norme matricielle appropriée.
   Si $\|U^{(l)} - U^{(l+1)}\| \leq \epsilon$, on arrête
   Sinon, retourner à 2.

III.3.4.4 Procédure d'agrégation floue hiérarchique dans l'estimation des paramètres

Pour estimer des paramètres directement à partir de l'observation, on doit résoudre deux problèmes:

- Déterminer le nombre de textures dans l'image
- Délimiter les régions homogènes contenant dans chacune un type de texture.

L'extraction des paramètres est donc faite sur ces régions homogènes.

Détermination du nombre de régions

L'approche générale de la détermination du nombre d'agrégats présents dans un ensemble de données, en utilisant l'algorithme d'agrégation floue, consiste à passer à l'algorithme les données et le nombre d'agrégats c, c = 2,3... Pour chaque valeur de c, une matrice d'appartenance U_c est obtenue. Un élément $u_{ik}^{(c)} \in U_c$ représente l'appartenance de l'élément k à la classe i. Pour déterminer le nombre réel d'agrégats qui existent dans l'ensemble des données, la stratégie habituellement adoptée est de chercher une fonctionnelle qui mesure la plausibilité d'une partition associée à une valeur de c particulière. Alors le nombre d'agrégats présents dans les données correspond à l'agrégation dont la mesure est extrémum.

Plusieurs fonctionnelles ont été proposées dans la littérature, on peut présenter ici celles qui sont les plus utilisées:

1) Coefficient de partition

$$F(U_C) = \sum_{i=1}^{c} \sum_{k=1}^{K} u_{ik}^2 / K \qquad 0 \leq u_{ik} \leq 1$$

2) Entropie de classification

$$H(U_C) = - \sum_{i=1}^{c} \sum_{k=1}^{K} u_{ik} \log u_{ik} / K$$

3) Exposant de proportion

$$P(U_C) = -\log_e \left\{ \prod_{k=1}^{K} \left[\sum_{j=1}^{[\Omega_k^{-1}]} (-1)^{j+1} \binom{c}{j} (1-j\Omega_k)^{c-1} \right] \right\}$$

où: $\Omega_k = \max_{1 \leq i \leq c} \{u_{ik}\}$

$[\Omega_k^{-1}]$ = entier le plus grand $\leq \frac{1}{\Omega_k}$.

Toutes ces mesures ont la même inconvénience; elles sont en fonction de c, donc leurs valeurs n'ignorent pas seulement la structure présente dans les données, mais elles varient aussi en fonction de c. On peut trouver des arguments dans les travaux de Libert (1983) et Bezdek (1981).

Des efforts ont été réalisés pour remédier au problème. En général, on cherche à compenser l'effet de c sur la mesure, on peut citer quelques exemples:

1) Indice NF (Non Fuzziness Index)

$$NFI = \frac{cF-1}{c-1}$$

où F est le coefficient de partition

c est le nombre d'agrégats.

2) $H' = \frac{K\ H}{K-c}$ ou $H'' = \frac{H}{\log_a(c)}$

où K: nombre d'échantillons

c: nombre d'agrégats

H: entropie de classification

a: base logarithmique arbitraire.

La justification pour ces choix n'a normalement pas de base théorique et est très spécifique à ce problème.

L'approche d'agrégation flou hiérarchique que nous avons adoptée pour résoudre le problème de validité des agrégats, représente un avantage évident sur l'ancienne approche. Elle ne pose pas le problème de l'influence de c sur la mesure de plausibilité (peu importe la mesure utilisée). Elle a aussi d'autres caractéristiques très intéressantes dont on discutera plus loin.

Méthode d'agrégation floue hiérarchique

La méthode consiste à diviser l'ensemble de données en deux par l'algorithme d'agrégation flou (Bezdek (1981)). Pour une division, une mesure de plausibilité est obtenue pour en quantifier la qualité. Si la division est plausible, les deux sous-ensembles obtenus par la division sont soumis à une autre division, sinon on considère l'ensemble à diviser comme homogène: il forme un agrégat dans l'espace des attributs.

Voici un exemple d'une procédure de division qui donne 3 agrégats:

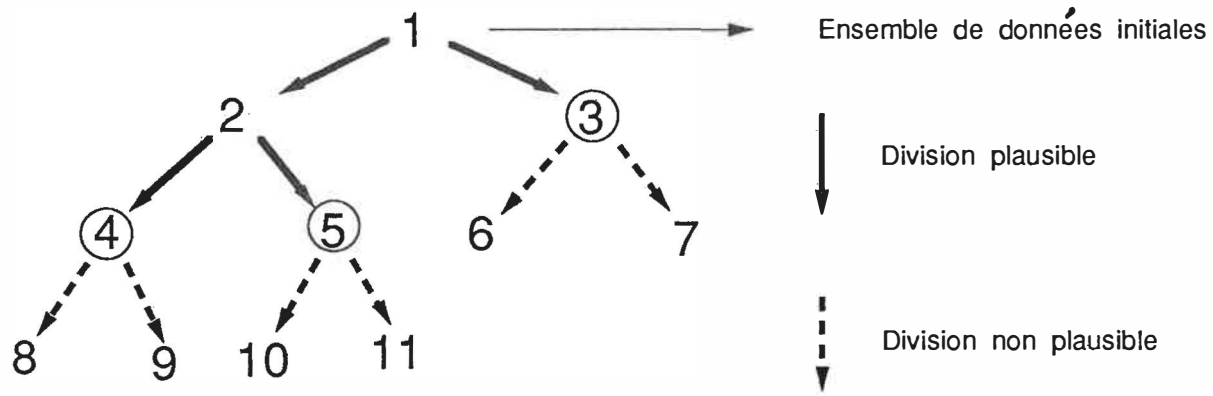


Figure 3.12 - Exemple d'une division hiérarchique.

Dans cet exemple, les agrégats étiquettés 3, 4 et 5 sont identifiés comme homogènes, on a donc 3 agrégats dans l'ensemble des données.

Comme on a mentionné ci-dessus, on peut prendre n'importe quelle mesure présentée au début de l'exposé, soit F, H, P ou même la mesure de Fisher. Prenons par exemple F, il est évident que F ne dépend pas de c (c=2 partout), mais F est en fonction du nombre d'échantillons des sous-ensembles à diviser, et ceci peut représenter un défaut de la mesure, parce que la taille des sous-ensembles diminue au fur et à mesure qu'on avance dans l'hiérarchie. Cependant, cet inconvénient n'existe plus si la taille du plus petit sous-ensemble est assez élevée.

Justification

Dans le cas c=2

$$F(U_{(2)}) = \sum_{k=1}^{K} \left[u_{1k}^{2} + u_{2k}^{2} \right]$$

Soit $S = u_{1k}^2 + u_{2k}^2$.

Si on peut considérer S comme une variable aléatoire dont la forme de la fonction de distribution représente la structure présente dans le sous-ensemble.

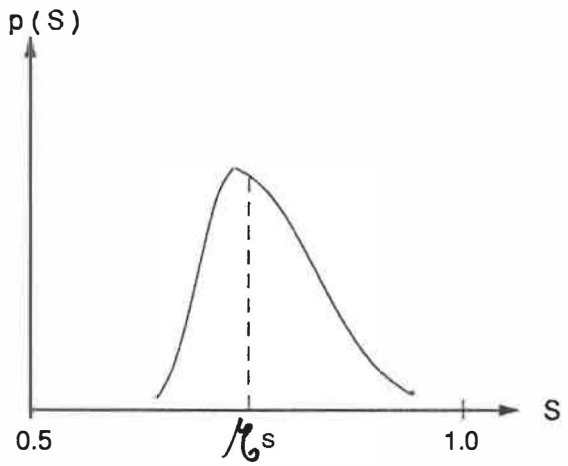


Figure 3.13 - Forme de la fonction de distribution F.

La quantité $F(U_{(2)})$ représente l'estimation de la moyenne de la distribution et c'est évident que

$$\hat{F}(U_{(2)}) = \lim_{K \to \infty} \left\{ \sum_{k=1}^{K} S/K \right\}$$

Dans le cas où K est assez élevée, l'estimation est proche de Ω_S. Donc, dans le cas où la taille du sous-ensemble est assez élevée, et dans le cas où c=2, la mesure F est impartiale et représente seulement l'indication de la structure de l'ensemble.

Discussion sur le seuil d'homogénéité

Dans l'approche d'agrégation floue hiérarchique, un seuil pour la mesure (de F par exemple) est nécessaire pour décider si le sous-ensemble est homogène ou non. On sait que la mesure de F dépend de la valeur de l'exposant de pondération m et on a démontré que F est une fonction monotone décroissante de m. Donc, le seuil choisi dépend aussi de m. Le problème est, pour une valeur donnée de m, comment choisir le seuil, ou bien, pour une valeur arbitrairement choisie du seuil, comment déterminer m. Empiriquement, pour les textures basées sur lesquelles nous avons fait des essais et pour les caractéristiques texturales utilisées, une valeur de m=2 ou 3 associée à un seuil S $\leq$0.41 est suffisante dans la plupart des cas étudiés.

Une procédure automatique pour déterminer m (en fixant le seuil) consiste à faire la division sur une texture quelconque mais de même classe (macro ou micro, même nombre de niveaux de gris, avec ou sans égalisation), la valeur de m est ajustée jusqu'au moment où la valeur de F atteint la valeur du seuil d'homogénéité.

Dans le même ordre d'idée, on peut déterminer m (en fixant S) en divisant l'image en des sous-images chevauchées de taille B x B. Pour chaque sous-image BxB, on cherche à diviser en deux en fixant m (=2, par exemple), et la valeur de F est calculée. Ensuite, une procédure d'agrégation floue avec c=2 est appliquée sur les valeurs de F pour isoler le groupe de faible valeur de F avec le groupe de forte valeur de F. L'identification des sous-images de faibles valeurs de F est utilisée pour ajuster m.

Justification

En choisissant la taille B appropriée, dans la plupart des cas, les sous-images B x B appartiennent à deux classes: homogène ou mixte, donc quel que soit m utilisé pour diviser en deux les sous-images B x B, les valeurs de F sont faibles s'ils sont homogènes ou fortes s'ils sont de nature mixte. Par conséquent, la division en deux basée sur les valeurs de F, va isoler des blocs homogènes pour fin d'ajustement de m.

Discussion sur la qualité des agrégats

Pour les données dont la structure est évidente (c.-à-d. que les agrégats sont bien séparés), la qualité des vrais agrégats identifiés par l'approche hiérarchique et l'approche conventionnelle est à peu près la même. On entend par qualité d'un agrégat, la ressemblance entre celui identifié par l'algorithme dans le domaine spatial et celui

réellement présent sur l'image. Mais dans le cas où les agrégats sont proches dans l'espace des attributs, nos essais montrent que l'agrégation hiérarchique est bien meilleur au niveau de la qualité des agrégats qu'au niveau de la performance: il y a des cas où l'approche hiérarchique identifie correctement les régions, tandis que l'approche conventionnelle donne des résultats aléatoires.

Discussion sur l'effort de calcul

Parce que le nombre d'échantillons des sous-populations diminue au fur et à mesure qu'on avance dans la hiérarchie, l'approche hiérarchique prend beaucoup moins de temps (5 fois) que l'approche conventionnelle.

Discussion sur la mesure de Fisher et le coefficient de partition F

Dans nos essais, nous utilisons aussi la mesure de Fisher définie comme suit (pour deux classes):

$$MF = \frac{K_1}{K_1 + K_2} \frac{O_1O_2}{R_1} + \frac{K_2}{K_1 + K_2} \frac{O_1O_2}{R_2}$$

Dans tous nos essais, on remarque que MF et F (ou H ou P) suivent la même tendance: une division qui donne une valeur de MF élevée donne aussi une valeur de F élevée. Donc, nos raisonnements dans les sections ci-dessus en prenant F comme exemple sont aussi vrais pour MF. Une seule différence entre MF et F est que

$$Min(MF) = 0$$

$$Min(F) = 0.5 \text{ pour } c=2$$

III.4 Conclusion

Nous avons présenté notre approche de segmentation des images texturées. Cette approche utilise le champ aléatoire markovien pour modéliser l'image et le processus de formation des régions. La stratégie du maximum de vraisemblance et celle du maximum à posteriori sont utilisées comme critères de segmentation. En choisissant un système de voisinage approprié, nous avons introduit la transformation de Hadamard pour convertir l'observation originale en des caractéristiques décorrelées qui, sous l'hypothèse de normalité des variables aléatoires formant le champ, permettent de déduire une expression compacte pour la distribution de Gibbs associée à chaque champ de texture. Le problème d'estimation des paramètres du modèle dans un environnement non supervisé a été résolu par la technique d'agrégation floue.

Un exposé de la théorie du champ aléatoire markovien au début du chapitre permet d'avoir une base théorique suffisante pour la continuité de la présentation. Deux travaux intéressants, rapportés récemment dans la littérature, et traitant du même problème de segmentation, ont été aussi présentés pour fins de comparaison avec notre approche.

Notre approche a été implantée et les résultats obtenus sont présentés et discutés dans le prochain chapitre.

CHAPITRE IV

EXPERIMENTATION ET RESULTATS

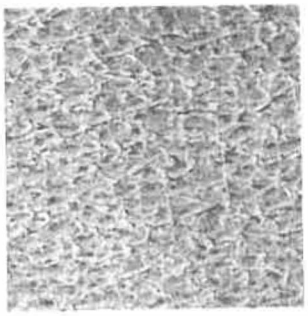
IV.1 Ressources matérielles et logicielles

Des images de microtextures naturelles, sélectionnées à partir de l'album de Brodatz, ont été utilisées pour la formation des images à segmenter. Les textures suivantes ont été choisies:

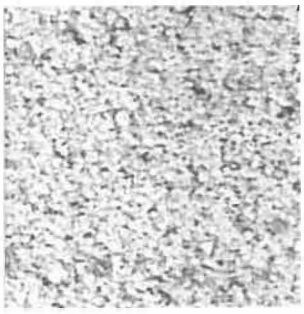
- Raffia (D84)



- Pigskin (D92)



- Beach Land (D29)

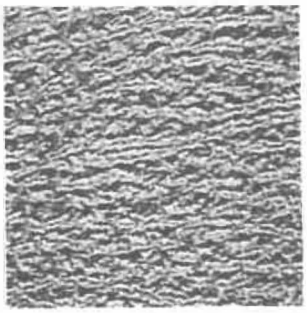


- Pressed Cork (D4)



Figure 4.1 - Les textures naturelles.

- Pressed Calf Leather (D24)



- Grass Lawn (D9)

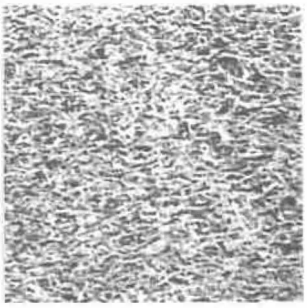
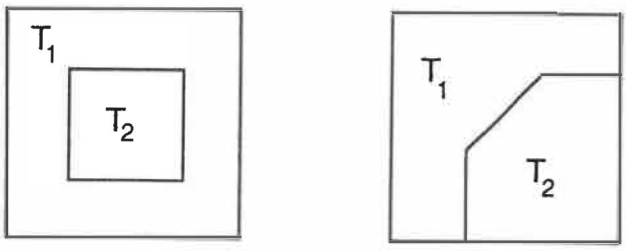


Figure 4.1 - Les textures naturelles (suite)

A partir de ces images, on a composé des images à segmenter de 256x256 pixels contenant chacune deux ou trois types de textures. Pour des images de deux textures, deux formes de régions ont été considérées, comme illustrées à la figure 4.2.

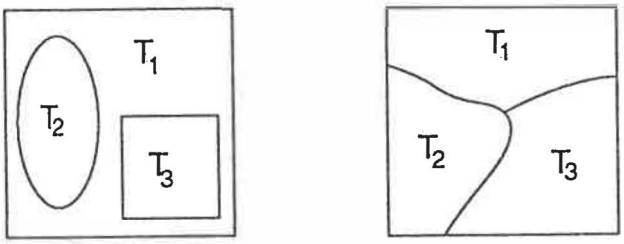
T_1 : texture de type 1



T_2 : texture de type 2

Figure 4.2 - Configuration des images bitexturées à segmenter.

Pour des images de trois textures, on considère les formes suivantes:



T_1 : texture de type 1

T_2 : texture de type 2

T_3 : texture de type 3

Figure 4.3 - Configuration des images tri-texturées à segmenter.

IV.2 Prétraitement

L'égalisation d'histogramme des niveaux de gris des textures originales est réalisée avant de former les images à segmenter. La technique consiste à transformer une image en une autre image, de façon telle que l'histogramme de la nouvelle image soit uniforme (fig.4.4). Normalement, il y a une réduction du nombre de niveaux de gris dans la nouvelle image. Dans nos essais, les 256 niveaux de gris des images originales ont été réduits à 32 niveaux par le processus d'égalisation.

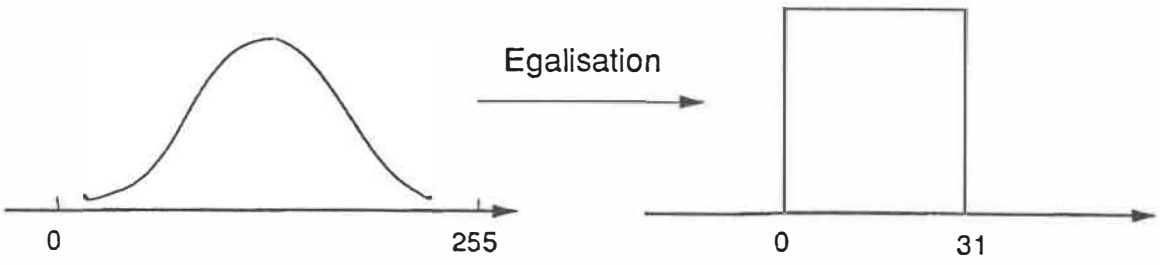


Figure 4.4 - Illustration graphique du processus d'égalisation.

L'égalisation a pour but d'obtenir des textures qui ont les mêmes statistiques du premier ordre et elle permet aussi de réduire les influences d'un mauvais éclairage dans le processus de prise d'images.

IV.3 Procédure de segmentation

Nous avons développé deux algorithmes de segmentation basés sur la théorie présentée à la section III.3, à savoir:

- un algorithme de segmentation avec modèles connus à priori
- un algorithme de segmentation sans connaissance à priori.

IV.3.1 Segmentation avec connaissance à priori

La figure 4.5 illustre le schéma-bloc de la procédure de segmentation avec connaissance à priori.

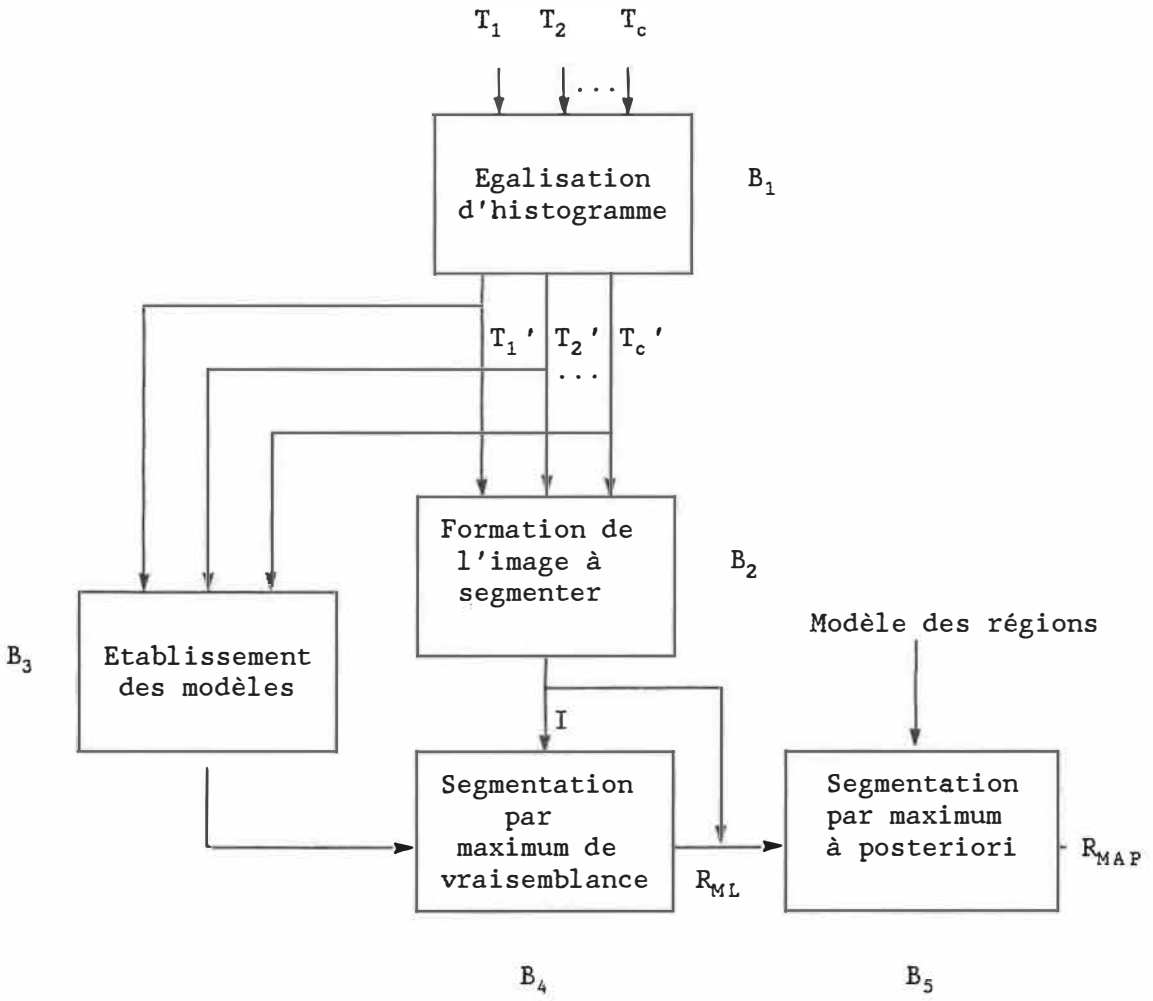


Figure 4.5 - Procédure de segmentation avec modèles connus à priori.

T_i : image de texture originale de type i. Ces images sont sélectionnées à partir de l'album de Bondatz; elles sont de taille 256x256 et ont 256 niveaux de gris.

T_i' : images égalisées du type i de taille 256x256 et de 32 niveaux de gris.

I : image à segmenter de taille 256x256 et de 32 niveaux de gris.

Modèles de T_1' et T_2' : 4 histogrammes de h1, h2... h4 pour chacune des textures (chapitre III).

R_{ML} : image résultante de la segmentation par stratégie de maximum de vraisemblance.

R_{MAP}: image résultante de la segmentation par stratégie de probabilité à postériori maximale.

Bloc B_1: réaliser une égalisation d'histogramme de niveaux de gris des textures T_i.

Algorithme:

Cet algorithme réalise une égalisation d'histogramme d'une image de taille NLINE x NCOL et de NGRIS de niveaux de gris. L'image résultante est de même taille et a un nombre de niveaux de gris réduit à NFGR.

1. Calculer l'histogramme des niveaux de gris de l'image originale. Cet histogramme a NGRIS intervalles: {his(n), n=1,2...NGRIS}.

2. Calculer le nombre de pixels pour chaque niveau de gris de la nouvelle image.

$$NPIX = \frac{NLINE \times NCOL}{NFGR}$$

3. Transformer le niveau de gris m dans l'image originale en niveau de gris n dans la nouvelle image conformément à la règle suivante:

$$\text{Si: } n \times NPIX \leq \sum_{i=0}^{m} hist(i) \leq (n+1) \times NPIX \Rightarrow m \text{ est transformé en } n.$$

Bloc B_3: Etablir les modèles des textures.

Algorithme:

1. Former 4 images intermédiaires en convoluant les 4 masques $H_1, H_2, \ldots H_4$ avec l'image de chaque texture T_k', k=1,2...,c.

2. Pour chaque texture T_k', k=1,2,...c, calculer les 4 histogrammes $h_1^{(k)}$, $h_2^{(k)} \ldots h_4^{(k)}$, à partir des 4 images intermédiaires résultantes de la convolution de 4 masques H_1, $H_2 \ldots H_4$ sur T_k'.

Bloc B_4 : Réaliser une segmentation par stratégie de Maximum de Vraisemblance.

Algorithme:

1. Lire les modèles des deux textures présentes dans l'image I.
2. Pour chaque pixel dans l'image I, calculer les 4 histogrammes locaux h_1' , h_2' ...h_4' à partir d'une fenêtre de taille NxN centrée sur ce pixel.
3. Calculer $U_K = \sum_{l=1}^{4} \sum_{m=1}^{NBINT} h_l'(m) \, L_n h_l^{(k)}(m)$, $= 1,2,...c$

   Si $U_i = \max \{U_k, k=1,2,...c\}$, assigner au pixel l'étiquette i.

   Remarque: NBINT est le nombre d'intervalles choisi pour chaque histogramme. Dans nos essais NBINT = 16 pour tous les histogrammes.

Bloc B_5 : Réaliser une segmentation par stratégie de probabilité à posteriori maximum.

Le processus de segmentation par MAP utilise l'image d'étiquettes résultant de la segmentation par ML comme état initial. A chaque itération une nouvelle image d'étiquettes est produite et est utilisée comme une entrée à l'itération suivante. Le processus continue jusqu'à ce que l'image d'étiquettes produite soit stable.

Algorithme:

1. Initialisation $R_{MAP}^{(0)} = R_{ML}$.

2. Pour chaque pixel (i,j) de l'image à segmenter I, calculer la probabilité d'avoir l'étiquette k au pixel (i,j), k=1,2,...c où c est le nombre de textures dans l'image I. Pour ce faire on suit les étapes suivantes:

- Soit $V_{ij}^{(n-1)}$ est un voisinage de 3x3 du pixel (i,j) dans l'image d'étiquette $R_{MAP}^{(n-1)}$ à l'itération (n-1), on transforme ce voisinage $\vartheta_{ij}^{(n-1)}$ tel que: si $p \in V_{ij}^{(n-1)}$ est le même étiquette que k, le pixel p dans $\vartheta_{ij}^{(n-1)}$ va porter l'étiquette 1, sinon il porte l'étiquette -1.

- Calculer:

$$W_k = U_k + \alpha\left[\vartheta_{ij}^{(n-1)}(i,j)\right]$$

$$+ \beta\left[\vartheta_{ij}^{(n-1)}(i-1,j) + \vartheta_{ij}^{(n-1)}(i+1,j) + \vartheta_{ij}^{(n-1)}(i,j-1) + \vartheta_{ij}^{(n-1)}(i,j+1)\right]$$

$$\gamma\left[\vartheta_{ij}^{(n-1)}(i-1,j-1) + \vartheta^{(n-1)}(i+1,j+1) + \vartheta^{(n-1)}(i+1,j-1)\right.$$

$$\left. + \vartheta^{(n-1)}(i-1,j+1)\right]$$

Si $W_S = \max\{W_k,\ k=1,2,...c\}$, assigner le pixel (i,j) à l'étiquette S.

- Le processus est répété pour tous les pixels dans l'image I, ce qui produit l'image d'étiquette $R_{MAP}^{(n)}$.

3. n=n+1. Retourner à 1. Le processus est terminé quand $R_{MAP}^{(n)}$ est stable.

Les paramètres α, β et γ représentent différentes interactions entre le pixel courant et les autres pixels selon différentes directions dans le voisinage V_{ij}. Dans nos essais, on choisit:

$$\alpha = \beta = \gamma = 1$$

IV.3.2 Segmentation sans connaissance à priori

La figure 4.6 illustre le schéma-bloc de la procédure de segmentation sans connaissances à priori.

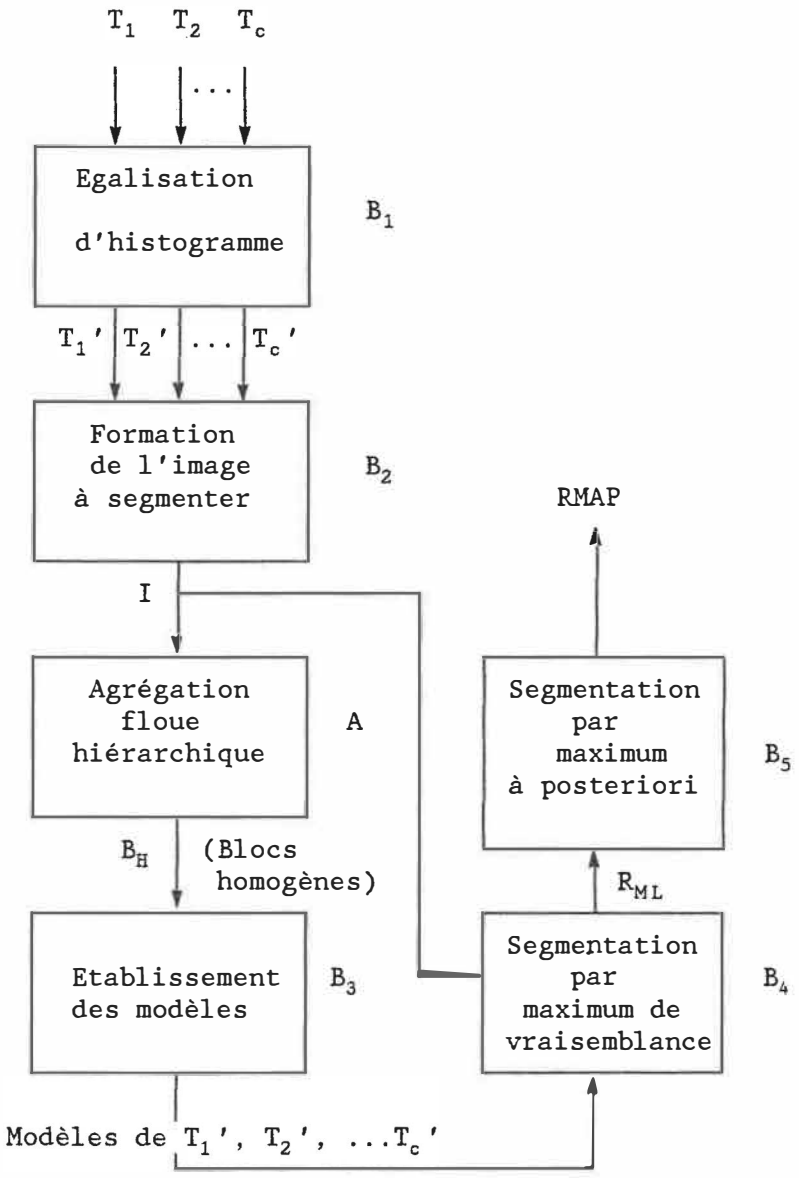


Figure 4.6 - Procédure de segmentation sans connaissance à priori.

Bloc A: Réaliser un processus d'agrégation floue hiérarchique.

Ce processus détermine le nombre de textures présentes dans l'image à segmenter I, et il fournit aussi des régions homogènes pour l'établissement des modèles.

Algorithme:

1. Diviser l'image I en blocs de taille MxM. Pour augmenter le nombre de blocs, un certain degré de recouvrement des blocs est accepté.

2. Extraire 4 histogrammes h_1, $h_2 \ldots h_4$ associés à chaque bloc. La concaténation des 4 histogrammes forme le vecteur des caractéristiques associé au bloc. Donc, chaque bloc est représenté par son vecteur de caractéristiques, l'ensemble des vecteurs de caractéristique constitue les données.

3. Appliquer l'algorithme d'aggrégation floue aux données, en supposant qu'il y a deux classes (c=2).

4. Calculer la mesure de Fisher pour mesurer la plausibilité de la division.

$$MF(\text{mesure Fisher}) = \frac{K_1}{K_1+K_2} \frac{O_1 O_2}{\frac{1}{K_1} \sum_{i \epsilon E_1} d_{1i}} + \frac{K_2}{K_1+K_2} \frac{O_1 O_2}{\frac{1}{K_2} \sum_{i \epsilon E_2} d_{2i}}$$

où O_1O_2 : distance euclidienne entre deux centres d'agrégats

E_1 : ensemble des éléments appartenant à la classe 1

E_2 : ensemble des éléments appartenant à la classe 2

K_1 : card (E_1)

K_2 : card (E_2)

5. Si MF est plus petit d'un seuil, on arrête le processus de division. Sinon, les deux agrégats sont soumis à la même procédure de division (retourner à l'étape 3).

6. Le processus de division hiérarchique est terminé quand aucune sous-population n'est divisible.

Dans nos essais, on choisit

- Taille des blocs (MxM) = 30x30 pixels

- Distance entre deux blocs: 7x7, ce qui donne 1024 blocs pour une image 256x256.

Paramètres de l'algorithme d'aggrégation floue:

- m=2
- norme matricielle

$$\left\| U(b) - U(b+1) \right\| = \text{Min} \left\{ \left| u_{ik}^{(b)} - U_{ik}^{(b+1)} \right| \right\}$$

où U(b) : matrice d'appartenance à l'itération b

U(b+1): matrice d'appartenance à l'itération (b+1)

$U_{ik}^{(b)}$: appartenance de l'élément k à la classe i à l'itération b

- seuil sur norme matricielle $\epsilon = 10^{-4}$

IV.4 Résultats

Les résultats sont présentés par des images. Il y a deux types d'image: le premier type présente le résultat d'une segmentation avec supervision.

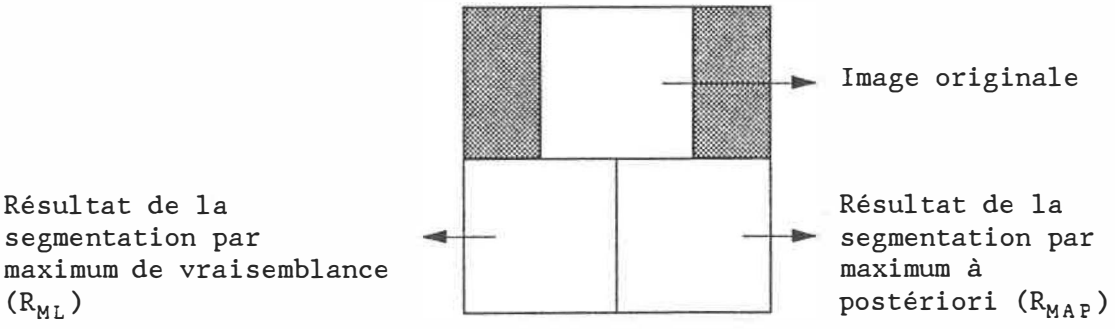


Figure 7a - Illustration de la présentation des résultats d'une segmentation avec supervision.

Le deuxième type présente le résultat d'une segmentation sans supervision.

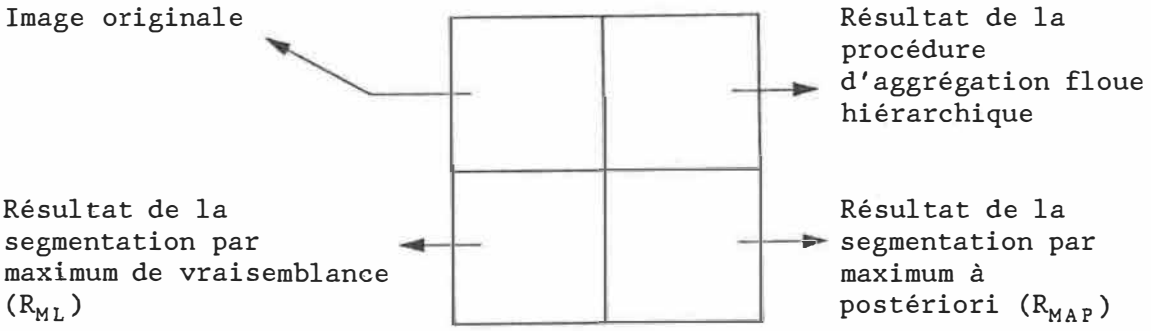


Figure 4.7b - Illustration de la présentation des résultats d'une segmentation sans supervision.

Dans des images R_{ML} et R_{MAP}, des régions de différents types de textures sont représentées par des contours différents. Pour l'image d'aggrégation floue hiérarchique, les blocs noirs sont des blocs non fiables à écarter dans le processus d'apprentissage.

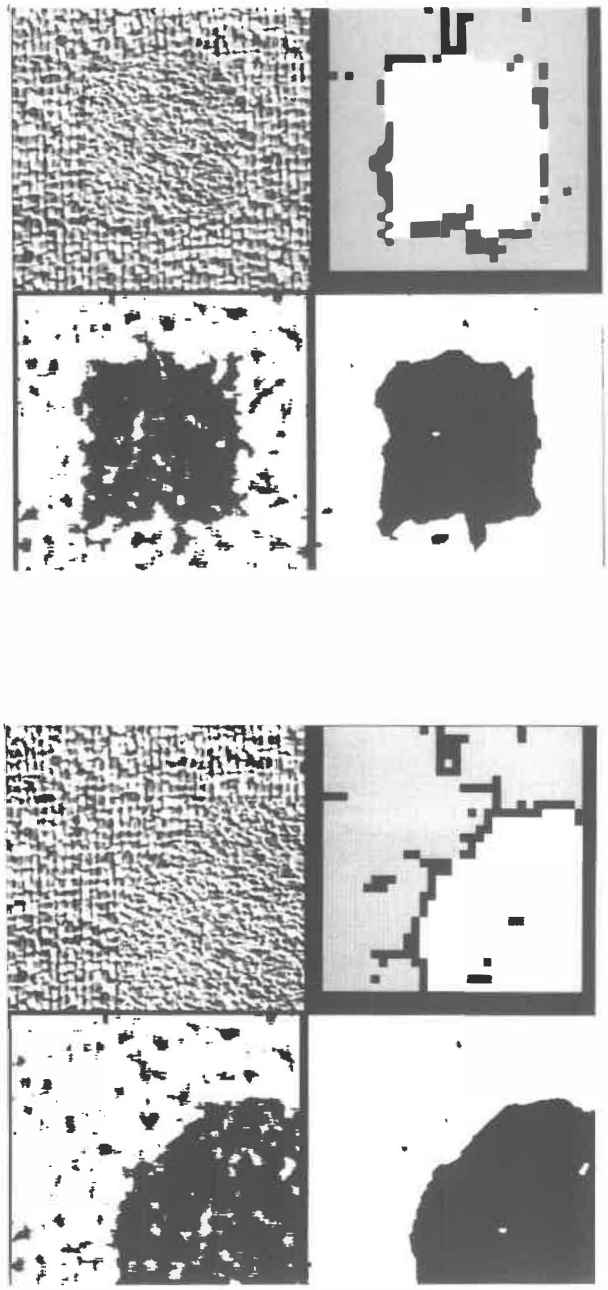


Figure 4.8 - Résultat de la segmentation non-supervisée des images composées des textures D84 et D4 égalisées.

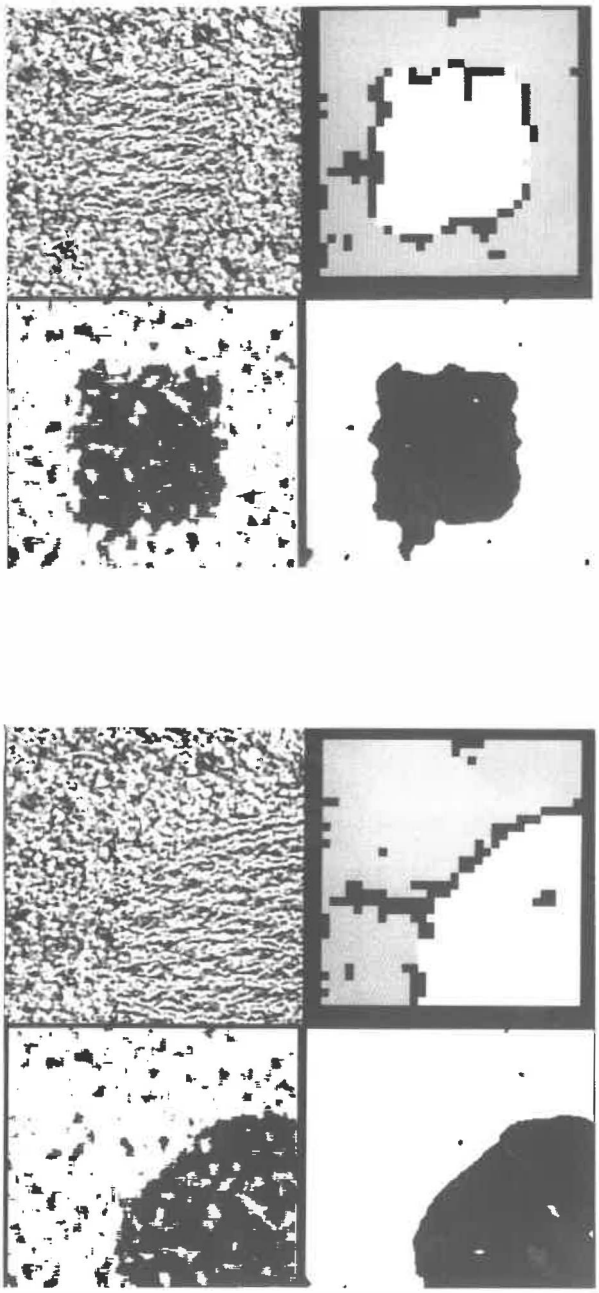


Figure 4.9 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D24 égalisées.

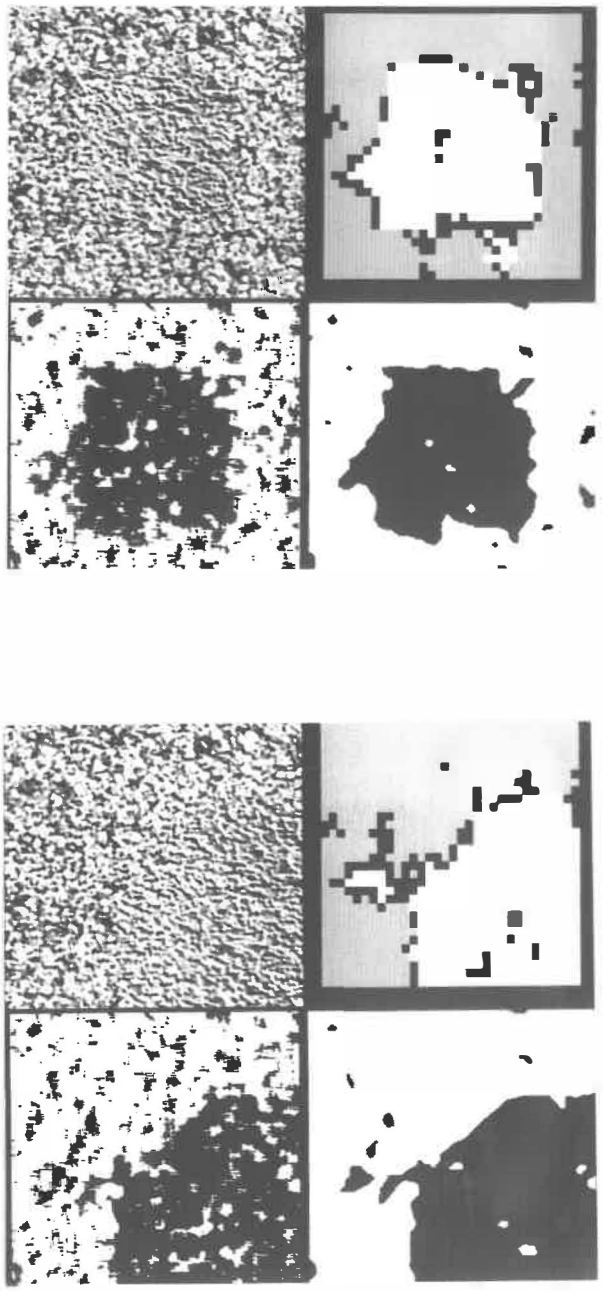


Figure 4.10 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D4 égalisées.

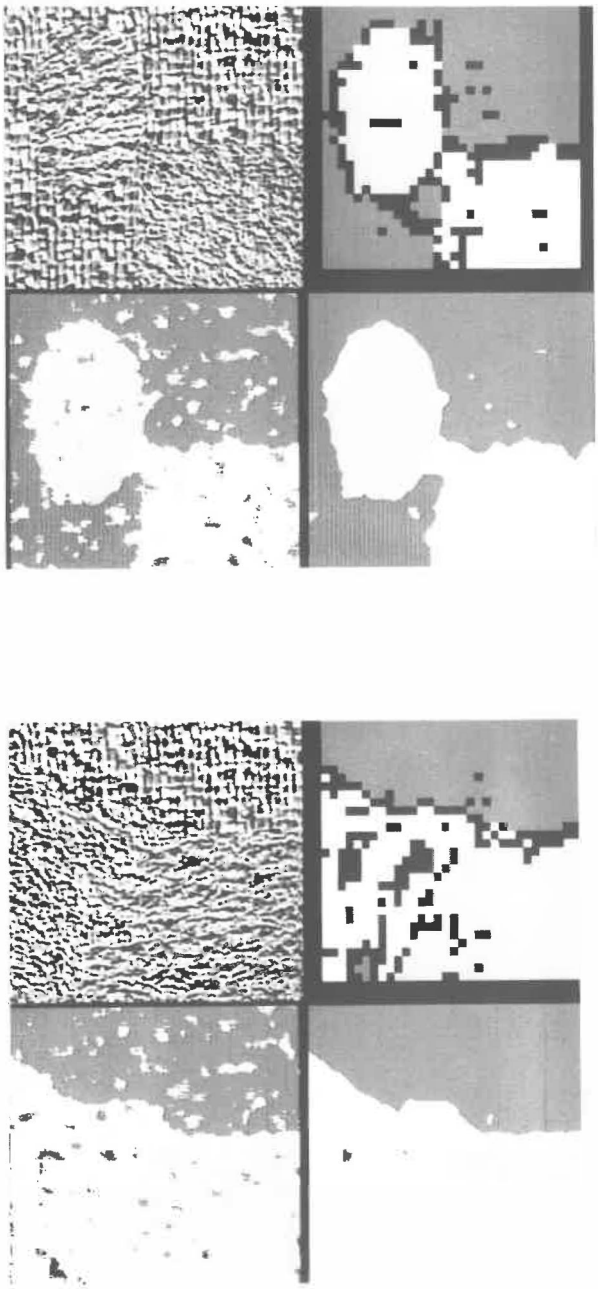


Figure 4.11 - Résultat de la segmentation non-supervisée des images composées des textures D84, D24 et D4 égalisées.

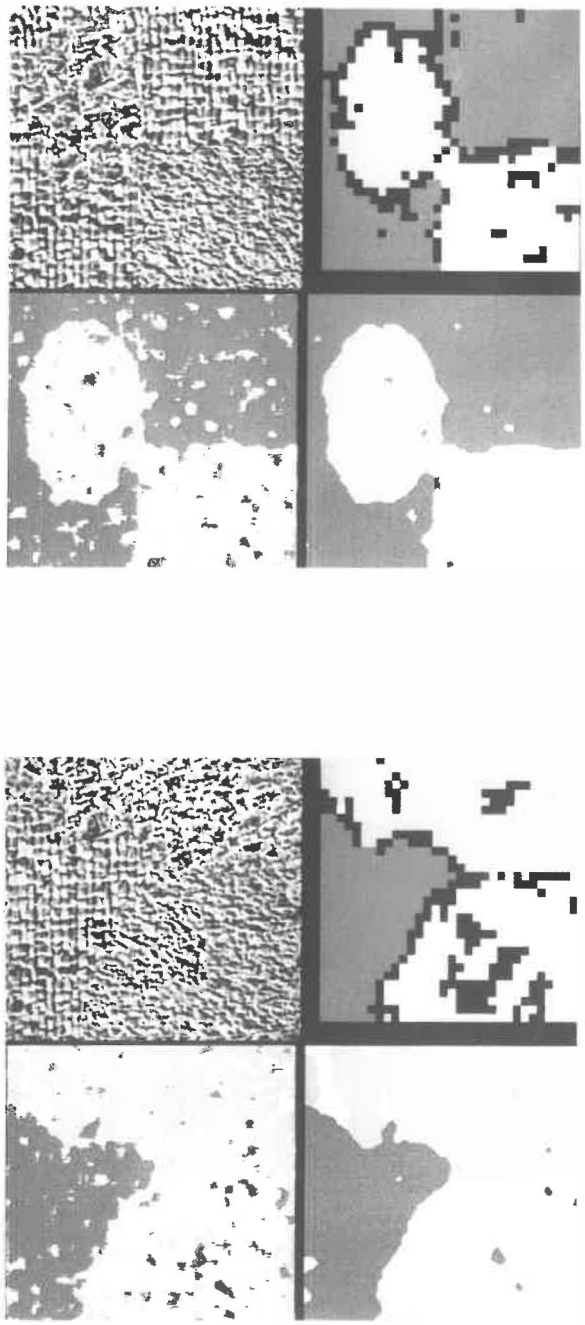


Figure 4.12 - Résultat de la segmentation non-supervisée des images composées des textures D84, D92 et D4 égalisées.

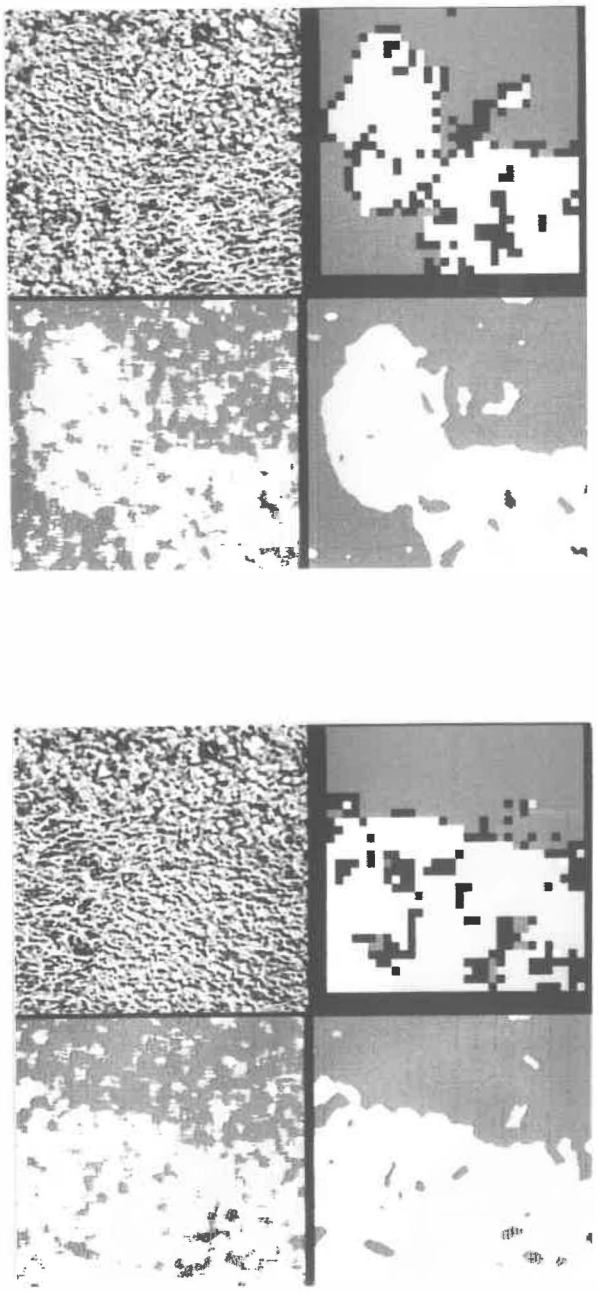


Figure 4.13 - Résultat de la segmentation non-supervisée des images composées des textures D29, D4 et D9 égalisées.

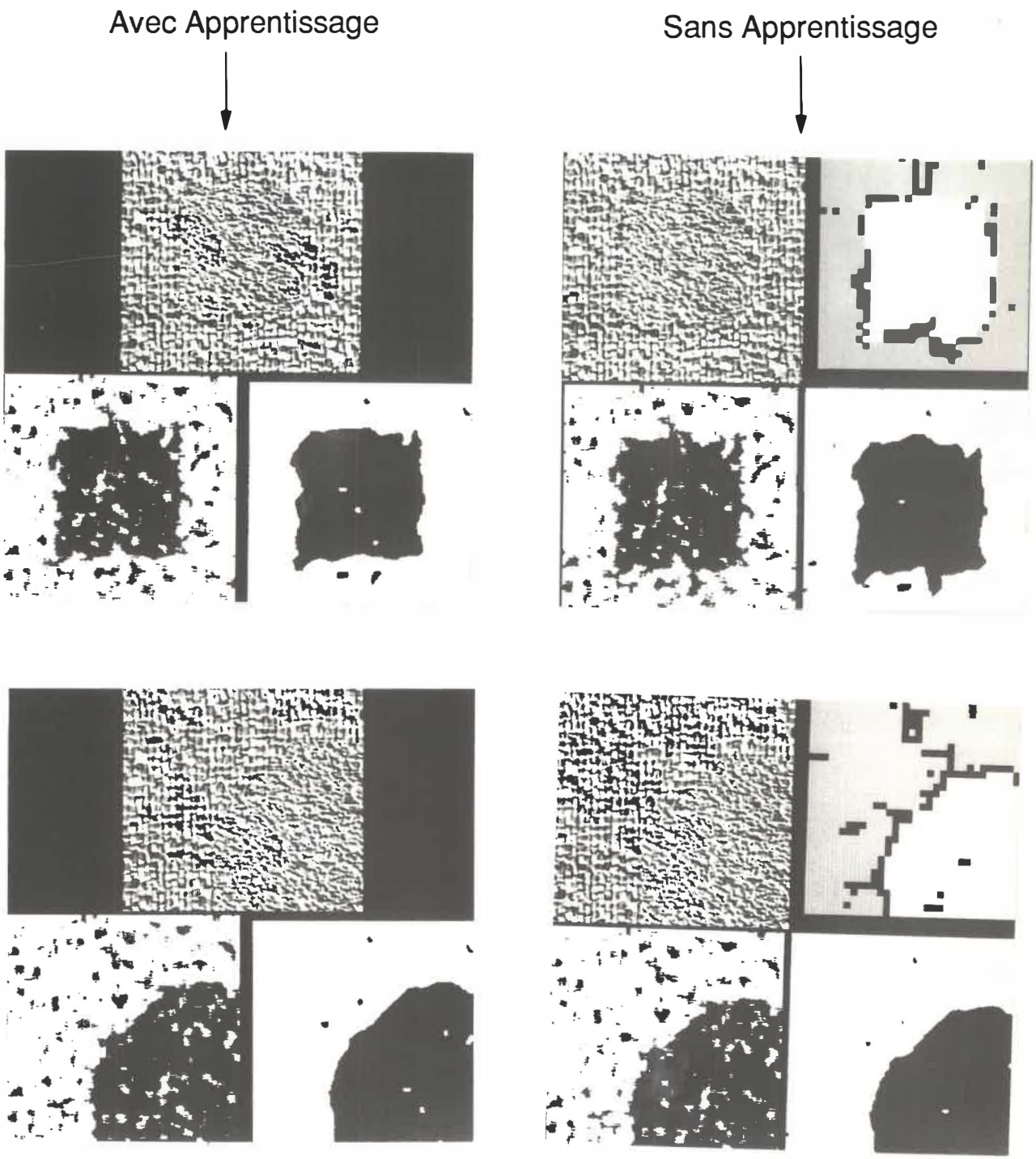


Figure 4.14 - Résultat de la segmentation supervisée et non-supervisée des images composées des textures D84 et D4 égalisées.

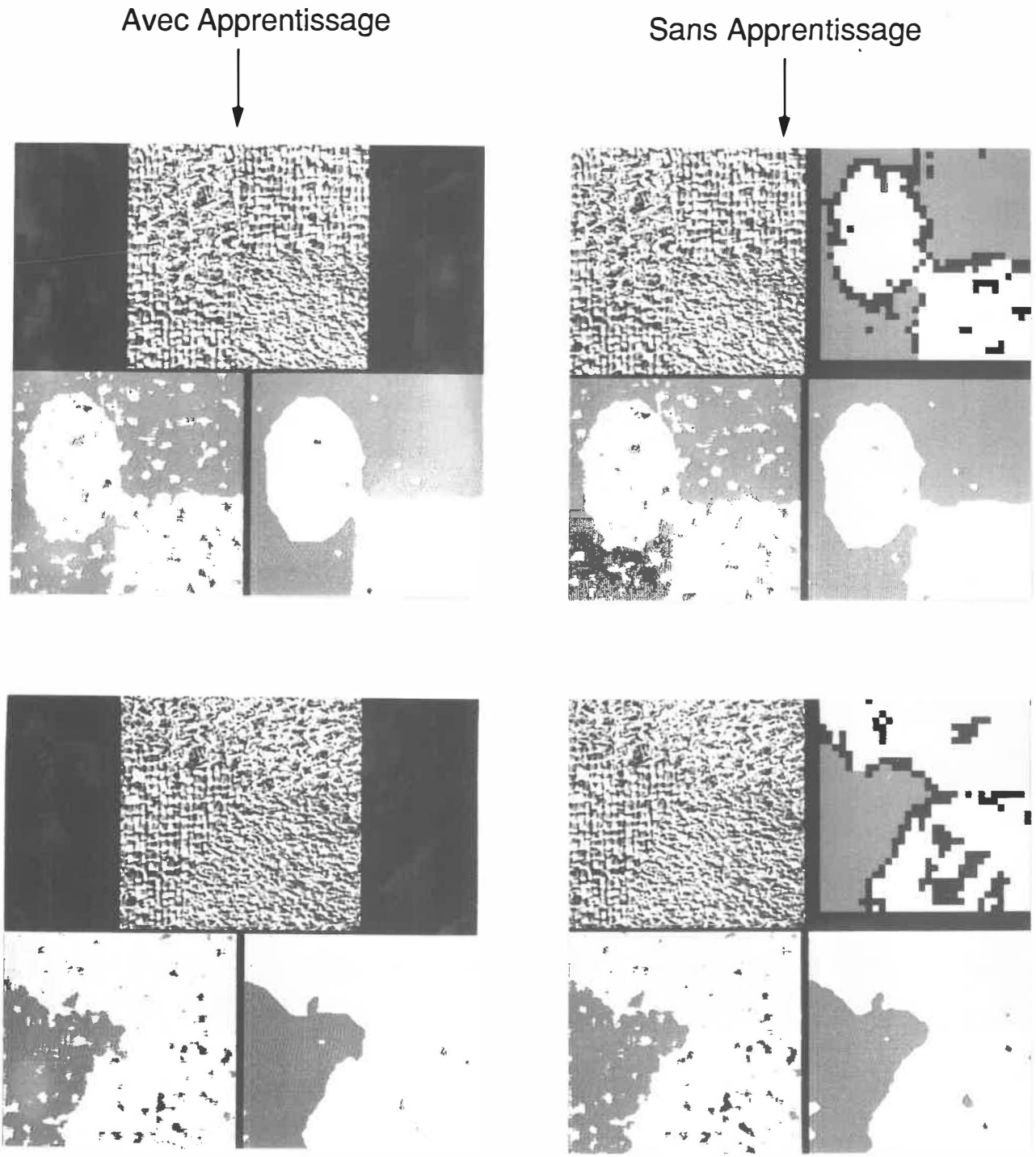
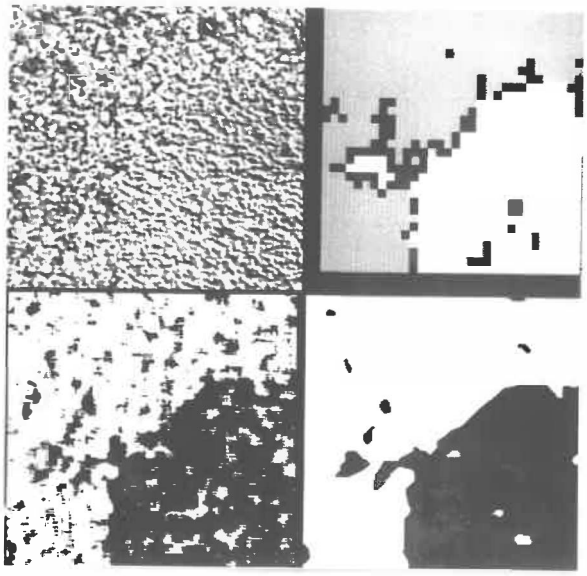


Figure 4.15 - Résultat de la segmentation supervisée et non-supervisée des images composée des textures D84, D92 et D4 égalisées.

Avec Egalisation



Sans Egalisation

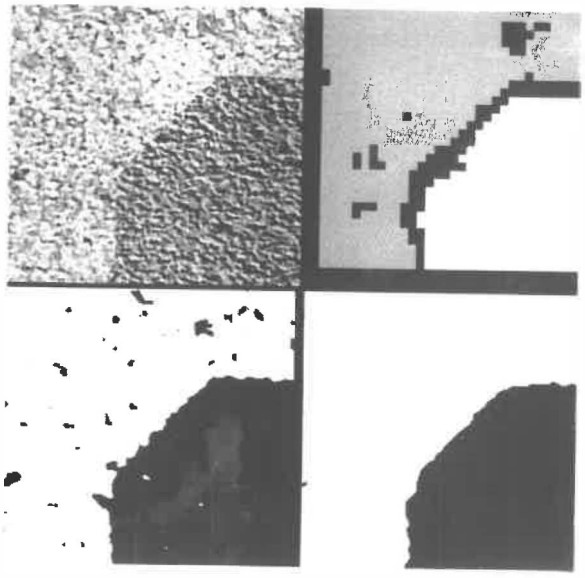
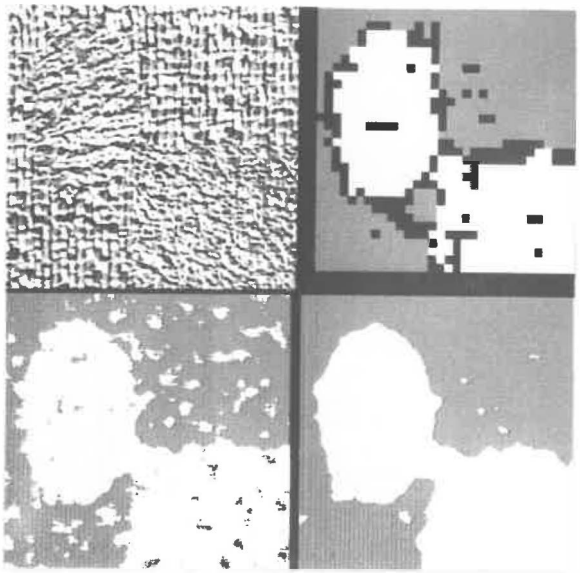


Figure 4.16 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D4 avec égalisation et sans égalisation.

Avec Egalisation



Sans Egalisation

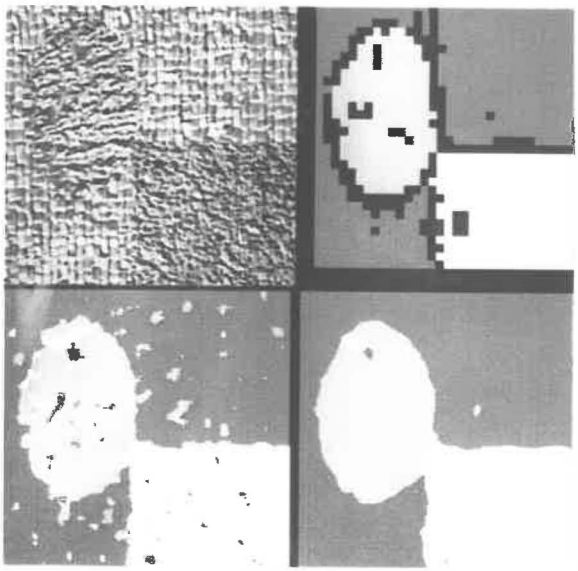
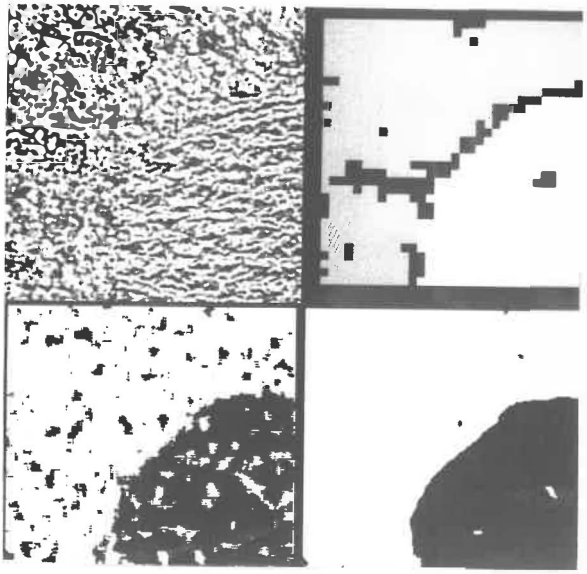
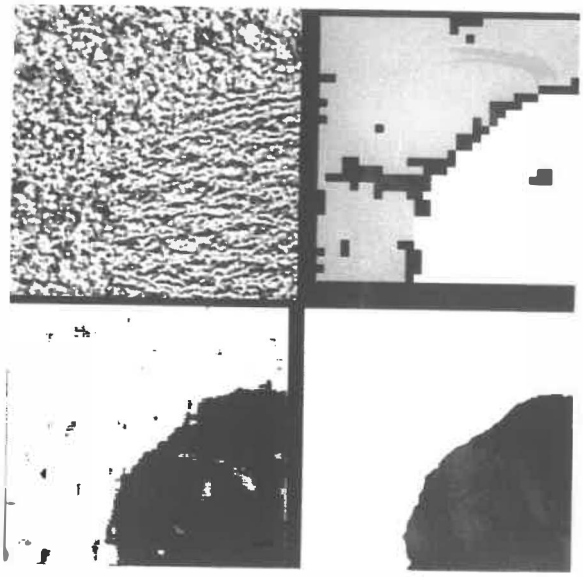


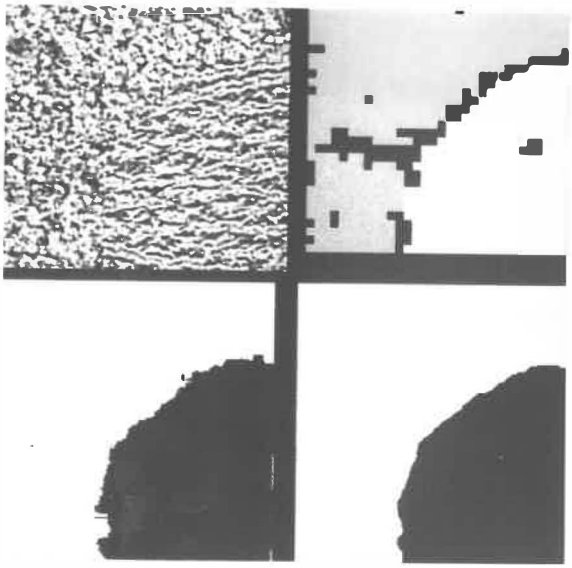
Figure 4.17 - Résultat de la segmentation non-supervisée des images composées des textures D84, D24 et D4 avec égalisation et sans égalisation.



9 x 9

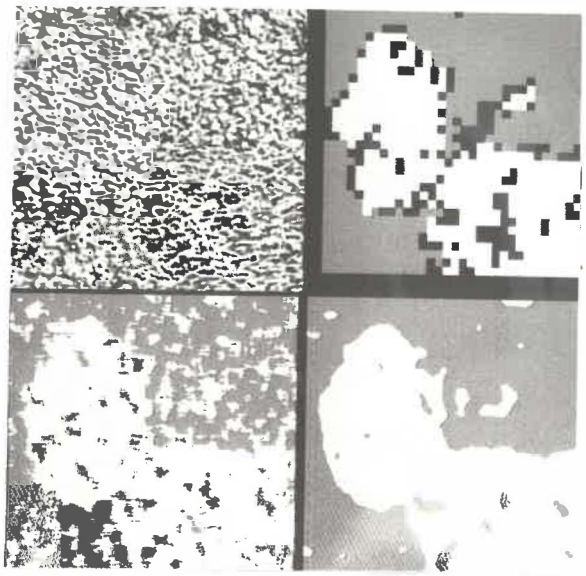


15 x 15

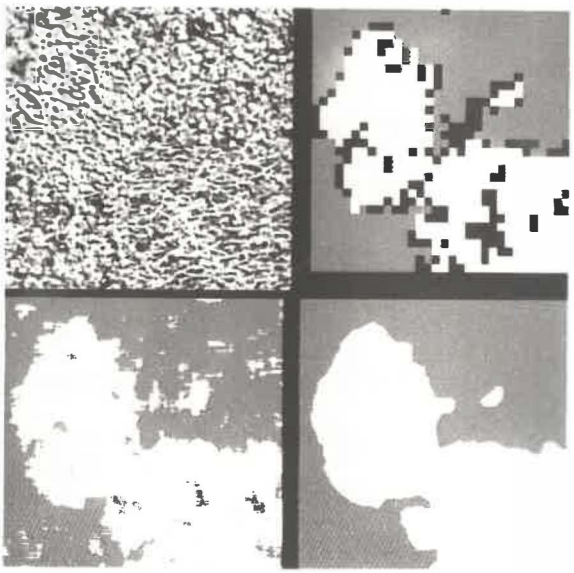


21 x 21

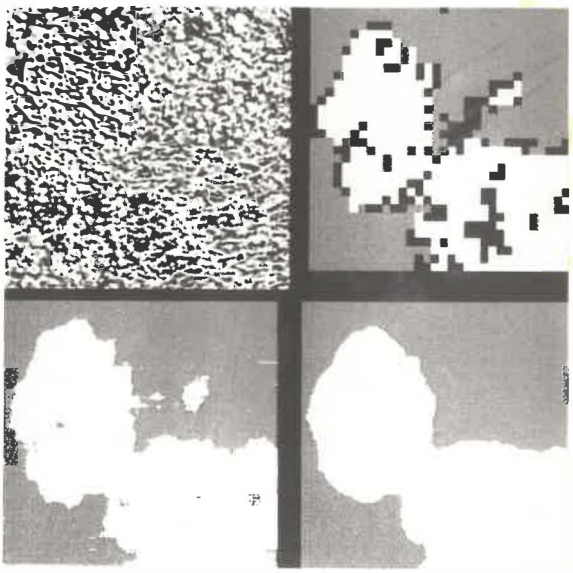
Figure 4.18 - Résultat de la segmentation non-supervisée des images composées des textures D29 et D24 égalisées avec différentes tailles de fenêtre d'analyse.



9 x 9



15 x 15



21 x 21

Figure 4.19 - Résultat de la segmentation non-supervisée des images composées des textures D29, D4 et D9 égalisées avec différentes tailles de fenêtre d'analyse.

IV.5 Discussion

Les résultats des pages précédentes nous montrent que la performance de la segmentation dépend de la taille de la fenêtre d'analyse: plus elle est grande, meilleur est le résultat. Ceci est normal parce que les statistiques obtenues dans le cas d'une grande fenêtre sont plus stables. Cependant, si la fenêtre est trop grande, la frontière détectée entre deux régions de textures différentes sera déteriorée surtout si la frontière a la forme d'une courbure plus ou moins accentuée.

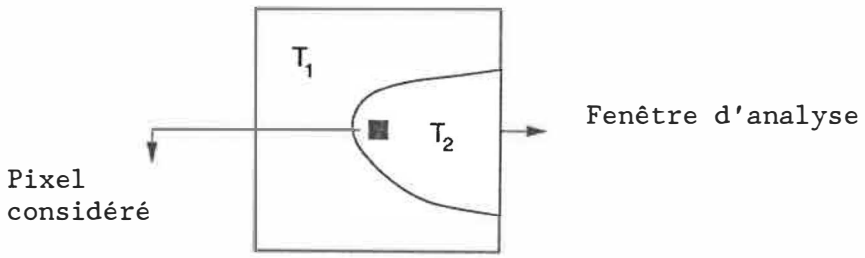


Figure 4.20 - Illustration de l'influence de la forme des régions sur la taille V de la fenêtre d'analyse.

Dans ce cas, le pixel en question a une forte probabilité d'être classifié au type 1, mais en réalité, il se trouve dans la région de type 2.

De plus, la forme des régions influence aussi le processus d'agrégation hiérarchique dans lequel la population est constituée par des blocs carrés extraits de l'image. Plus la frontière est courbe, plus le nombre de blocs contenant plus d'une texture augmente. Ces blocs mixtes estompent la séparation des agrégats dans l'espace de

caractéristiques. Pour remédier à ce problème, nous avons éliminé, pour une division, des blocs dont l'appartenance à la classe 1 ou 2 n'est pas très forte.

La taille des régions influence aussi le processus d'agrégation hiérarchique. Des régions de très petite taille occasionnent des difficultés pour la représentation de ces régions dans l'espace des caractéristiques, parce que la dimension de l'espace des caractéristiques est élevée (64); les petites régions ne fournissent pas suffisamment d'échantillons, ce qui influence beaucoup le test d'homogénéité à la fin de chaque division.

Mis à part ces défauts, les résultats obtenus dans plusieurs conditions sont stables. Nos résultats montrent une légère diminution de performance entre l'approche avec supervision et sans supervision, ce qui est très souhaitable, parce qu'en pratique le modèle des textures n'est pas disponible dans la plupart des cas. Par conséquent, une bonne performance de l'approche non supervisée est très souhaitable et notre méthode montre qu'elle est efficace dans l'environnement sans supervision.

Dans nos essais, nous avons testé aussi des cas où l'image à segmenter est composée de textures égalisées. Même avec l'élimination des statistiques d'ordre 1, nos résultats dans ce cas sont très intéressants, ce qui montre l'efficacité de notre modèle.

IV.6 Conclusion

Dans ce chapitre, nous avons présenté une procédure de segmentation des images texturées. Cette procédure réalise notre théorie présentée au chapitre IV. Nous avons aussi présenté et discuté des résultats obtenus par des tests intensifs sur des données établies dans différentes conditions. Les résultats obtenus sont jugés satisfaisants, ce qui montre la validité de notre approche.

Nous avons mentionné aussi les limites de notre approche. Le problème des petites régions est relié directement à la dimension de l'espace des caractéristiques choisies, aux problèmes du compromis entre entre la taille de la fenêtre d'analyse et la forme de la frontière.

Malgré ces limites, notre méthode de segmentation peut être appliquée à un bon nombre de situations dans la pratique.

CONCLUSION

Ce mémoire a présenté une méthode de segmentation non supervisée des images texturées. Dans cette méthode, les textures sont modélisées par le modèle markovien dont les paramètres sont évalués sur des régions homogènes extraites par la technique d'aggrégation floue. La technique de segmentation consiste à chercher une partition qui satisfait le critère de vraisemblance maximum ou le critère de la probabilité à posteriori maximum. Dans le deuxième cas, le processus de formation des régions est aussi censé de suivre le modèle markovien.

Les résultats présentés dans le chapitre précédent montrent que la technique développée est efficace pour la segmentation des textures naturelles. La technique fonctionne bien aussi dans le cas plus difficile où les textures sont égalisées pour supprimer les statistiques du 1^{er} ordre.

La procédure d'agrégation floue hiérarchique donne des résultats satisfaisants dans la plupart des cas et nous pensons qu'elle est fort appropriée pour la technique non supervisée, où les blocs homogènes extraits avec un haut degré de confiance sont utilisés pour l'établissement des modèles. De plus, la technique donne un résultat assez stable vis-à-vis les paramètres de la procédure d'agrégation.

Cependant, notre méthode connaît une certaine limite. Les petites régions texturées représentent une difficulté pour notre méthode, parce qu'elles ne fournissent pas suffisamment d'échantillons pour la procédure d'agrégation hiérarchique et, habituellement, elles sont noyées dans la région contenant une texture de caractéristiques statistiques plus proches.

Notre approche ne peut être applicable aux textures hautement irrégulières, parce que, dans ce cas, la procédure d'agrégation s'est montrée inefficace, elle a tendance à subdiviser ces textures en des régions contenant des textures plus ou moins régulières.

BIBLIOGRAPHIE

BESAG, J., "Spatial Interaction and the Statistical Analysis of Lattice Systems", J. Royal Statist. Soc., serie B, vol. 36, p. 192-236 (1974).

BEZDEK, J.C., "Pattern Recognition with Fuzzy Objective Function Algorithms", New York, Plenium (1981).

BURT, J., HONG, T.H. et ROSENFELD, A., "Segmentation and Estimation of Image Region Properties through Cooperative Hierarchical Computation", IEEE Trans. Syst. Man Cybernet., vol. SMC-11 (1981).

CHELLAPPA R., "Two-Dimensional Discrete Gaussian Markov Random Field Models for Image Processing", Progress in Pattern Recognition, North Holland (1985).

CHELLAPPA R. and CHATTERJEE S., "Classification of Textures Using Gaussian Markov Random Fields", IEEE Trans. Acoustics, Speech and Signal Processing, vol.ASSP-33, n°4, p. 959-963 (1985).

CHEN, P.C. et PAVLIDIS, T., "Segmentation by Texture Using a Co-occurence Matrix", Comput. Graphics Image Process., vol.10, p. 1972-1982 (1979).

CHEN, P.C. et PAVLIDIS, T., "Image Segmentation as an Estimation Problem", Comput. Graphics Image Process., vol.12, p. 153-172, (1980).

CHEN, P.C. et PAVLIDIS, T., "Segmentation by Texture Using Correlation", vol. PAMI-5, n°1, (1983).

COHEN, F.S. et COOPER, D.B., "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal Markovian Random Fields", IEEE Trans. on Pattern Anal. Mach. Intel., vol. PAMI-9, n°2, p. 195-217 (1987).

COHEN P., LE DINH C.T., LACASSE V., "Classification of Natural Textures by Means of Two-dimentional Orthogonal Masks", to appear in IEEE Trans. Accoustic, Speech and Signal Processing.

COHEN, H.S. et COOPER, D.B., "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal MRF", vol. PAMI-9, (1987).

COLEMAN, G.B. et ANDREWS, H.C., "Image Segmentation by Clustering", Proc. IEEE, vol. 67 (1979).

CONNORS R.W. and HARLOW C.A., "Toward A Structured Textural Analyzer Based on Statistical Methods", Comput. Graphics Image Processing, vol.12, p. 224-256 (1980).

CONNORS R.W. and HARLOW C.A., "A Theorical Comparison of Texture Algorithms", IEEE Trans. Patter. Anal. Mach. Intel., vol.PAMI-2, n°3, p. 204-222 (1980).

CROSS G.R. and JAIN A.K., "Measures of Homogeneity in Textures", IEEE Proc. of Computer Vision and Pattern Recognition, p. 211-216 (1983).

CROSS G.R. and JAIN A.K., "Markov Random Field Texture Models", IEEE Trans. Pattern. Anal. Mach. Intel., vol. PAMI-5, n°1, p. 25-38 (1983).

CROSS, R.J., "Markov Random Fields Texture Models", Dept. Comput. Sci., College of Eng., Michigan State Univ., Tech. Rep. 80-02 (1980).

DAVIS L.S., "Image Texture Analysis Recent Development", International Conference on Pattern Recognition, p. 214-216 (1982).

DAVIS L.S., CLEARMAN M. and AGGARWALL J.K., "An Empirical Evaluation of Generalized Co-occurence Matrices", IEEE Trans. Pattern Anal. Mach. Intel., vol. PAMI-3, n°2, p. 214-221 (1981).

DAVIS L.S., JOHNS S. and AGGARWAL J.K., "An Empirical Evaluation of Generalized Co-occurence Matrices", IEEE Trans. Patter. Anal. Mach. Intel., vol. PAMI-1, n°3, p. 251-258 (1979).

DAVIS, W.A. et TYCHON, G.G., "Texture Boundary in Digital Images", Proc. of the 8-ICPR, p. 402-404.

DEGUCHI K. et MOKISHITA I., "Texture Characterization and Texture Based Partitioning using Two-Dimensional Estimation Techniques", IEEE Trans. Inform. Theory, vol.IT-27, p. 738-745 (1978).

DERIN, H. et COLE, W.S., "Segmentation of Textured Images Using Gibbs Random Fields", Comp. Vision, Graphics and Image Processing, vol.35, p. 72-98 (1986).

DEVINE, M.D. et LEEMET, J., "A Method for Non-purposive Picture Segmentation", Proc. of the 3-International Joint Conference on PR (1976).

DUNN, J.C., "A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, Well Separated Clusters", J. Cybern., vol. 3, p. 32-57 (1974).

FREEMAN J., "The Modelling of Spatial Relations", Computer Graphics and Image Processing, vol.4, n°2, p. 156-171 (1975).

GAGALOWICZ, A., "Stochastic Texture Fields Synthesis from A Priori Given Second Order Statistics", Proc. IEEE Comp. Soc. Conf. on Pattern Recognition and Image Processing, Chicago, p. 796-804 (1979).

GALLOWAY M.M., "Texture Analysis using Gray Level Run Lengths", Comput. Graphics Image Processing, vol. 4, p. 172-199 (1975).

GEMAN, S. et GEMAN, D., "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", IEEE Trans. on Pattern Anal. Mach. Intel., vol. PAMI-6, p. 721-741 (1984).

GOOL L.V., DEWAELE P. et OSTERLINCK O., "Texture Analysis Anno 1983", CVGIP, Vol. , p. 336-357, (1985).

HARALICK, R.M., "Digital Step Edges from Zero Crossing of Second Directional Derivative", IEEE Trans. Pattern Anal. Mach. Intell., PAMI-6, p. 58-68 (1984).

HARALICK R.M., "Statistical and Structural Approches to Textures", Proc. IEEE, vol. 67, n°5, p. 786-804 (1979).

HARALICK, R.M. et DINSTEIN, I., "A Spatial Clustering Procedure for Multi-image Data", IEEE Trans. on Circuits and Systems, CAS-22, (1975).

HARALICK, R.M., SHANMUGAM K. et DINSTEIN I., "Textural Features for Image Classification", IEEE Trans. Syst., Man, Cybern, vol. SMC-3, p. 610-621 (1973).

HARALICK R.M., SHAMUGAM K.S. et DINSTEIN I., "On Some Quickly

Computable Features for Textures", Proc. of the 1972 Symposium on Computer Image Processing and Recognition, vol.2, p. 12.2.1 - 12.2.10 (1972).

HARALICK, R.M. et SHAPIRO, L., "Image Segmentation Techniques", CVGIP, (1985).

HONG T.H., DYER C.R. et ROSENFELD A., "Texture Primitive Extraction Using An Edge Based Approach", IEEE Trans. Systems Man Cybern., vol. SMC-10, n°10, p. 659-675 (1980).

JULESZ B., "Visual Pattern Discrimination", IRE Trans. Inf. Theory, vol. IT-8, n°2, p. 84-92 (1962).

KANEKO H. et YODOGAWA E., "A Markov Random Field Application to Texture Classification", Proc. Patter. Recog. and Image Processing, p. 221-225 (1982).

KASHYAP, R.J. et CHELLAPA, R., "Estimation and Choice of Neighbours in Spatial Interaction Models of Images", IEEE Trans. Inform. Theory, p. 60-72 (1983).

KASHYAP R.J., CHELLAPPA, R. et KHOTANZAD, A., "Texture Classification using Features Derived from Random Field Models", Pattern Recognition Lett.1, p. 43-50 (1982).

KITTLER, J. et ILLINGWORTH, J., "Minimum Error Thresholding", PR, vol. 19, n°1, p. 41-47 (1986).

LEVINE M.D., "Vision in Man and Machine", Mc Graw-Hill (1985).

LEVINE M.D., OHANDLWY D., et YAGI G., "Computer Determination of Depth Map", Computer Graphics and Image Processing, vol.2, n°2, p. 131-151 (1973).

MATSUYAMA T., SABURI K. et NAGAO M., "A Structural Description of Regularly Arranged Textures", Proc. 5th Int. Conf. on Pattern Recognition, p. 1115-1118 (1980).

MILGRAM, D.L., "Region Extraction using Convergent Evidence", Comput. Graphics Image. Process., vol.11, p. 1-12 (1979).

NOBUYUKI OTSU, "A Threshold Selection Method from Grey-Level Histograms", IEEE Trans. Syst. Man and Cybernet., vol. SMC-9, n°1, p. 62-66 (1979).

PIETIKAINEN, M. et ROSENFELD, A., "Image Segmentation by Texture using Pyramid Node Linking", IEEE Trans. System. Man Cybernet., vol. SMC-11 (1981).

PIETIKAINEN M. et ROSENFELD A., "Edge Based Texture Measures", Proc. 6th Int. Conf. on Pattern Recognition, p. 298-300 (1982).

ROSANOV, Y.A., "On Gaussian Field with Given Conditional Distributions", Theory of Prob. and its Applications, vol. II, p. 381-391 (1967).

ROSENFELD A. et TROY E.B., "Visual Texture Analysis", Conference Record for Symposium on Feature Extraction and Selection in Pattern Recognition, IEEE Publication 70C-51C, Argone, III., p. 115-124 (1970).

RUSPINI, E., "Numerical Methods for Fuzzy Clustering", Inf. Sci., vol. 2, p. 319-350 (1970).

SCHACTER B., ROSENFELD A. et DAVIS L.S., "Random Mosaic Models for Textures", IEEE Trans. Syst. Man and Cybern., vol. SMC-8, p. 694-702 (1978).

SERRA J., "Boolean Model and Random Set", Comput. Graphics Image Processing, vol.12, p. 99-126 (1980).

SOUZA de P. "Texture Recognition via Autoregression", Pattern Recognition, 15, n°6, p. 471-482 (1982).

THERRIEN, C.W., QUATIERI, T.F. et DUDGEON, D.E., "Statistical Model Based Algorithms for Image Analysis", Proc. IEEE, vol.74 (1986).

THOMPSON, W.B., "Textural Boundary Analysis", IEEE Trans. Computer, vol. C-26, March (1977).

TOMITA F., SHIRAI Y. et TSUJI S., "Description of Textures by a Structural Analysis", IEEE Trans. Pattern Anal. Mach. Intel., PAMI-4, n°2, p. 183-191 (1982).

TRIENDL, E. et HENDERSON, T. "A Model for Texture Edges", Proc. of the 5-ICPR, p. 1100-1102 (1980).

UNSER M., "Sum and Difference Histograms for Texture Classification", IEEE Trans. Pattern. Anal. Mach. Intel., PAMI-8, n°1, p. 118-125 (1986).

WANG S., VELASCO D., WU A.Y. et ROSENFELD A., "Relative Effectiveness of Selected Texture Primitive Statistics for Texture Discrimination", IEEE Trans. Systems. man Cybern., vol. SMC-11, n°5, p. 360-370 (1981).

WESZKA J.S., DYER C.R. et ROSENFELD A., "A Comparative Study of Texture Measures for Terrain Classifications", IEEE Trans. Syst., Man, Cybern., vol. SMC-6, p. 269-285 (1976).

WESZKA, J.S. et ROSENFELD, A., "Threshold Evaluation Techniques", IEEE Trans. Syst. Man Cybernet., vol. SMC-8, p. 622-629 (1978).

WOODS, J., "Two Dimensional Discrete Markovian Fields", IEEE Trans. Inform. Theory, vol. IT-18, p. 232-240 (1972).

WONG, A.K.C. et RAAFAT, H.M., "Texture-Based Image Segmentation", IEEE Conf. on Computer Vision and Pattern Reg., p. 469-475 (1986).

ZUCKER S.W. et TERZOPOULOS D., "Finding Structure in Co-occurence Matrices for Texture Analysis", Comput. Graphics Image Processing, vol.12, p. 286-308 (1980).

ÉCOLE POLYTECHNIQUE DE MONTRÉAL
3 9334 00290856 2