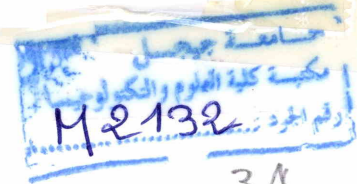


**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur**  
**et de la Recherche Scientifique**



Université de Jijel

Faculté des Sciences et de la Technologie

Département d'Electronique

*Mémoire de fin d'Etudes pour l'Obtention du Diplôme de*

*Master II en Electronique*

*Option :*

*Electronique et Analyse des Systèmes*

*Thème :*

**Segmentation des Images par Champ  
Aléatoire de Markov**

**Présenté par :**

**Melle : DAAMOUCHE Rafika**

**Melle : ABDELTIF Loubna**

**Encadré par :**

**Meme BOUATMANE Sabrina**

**Promotion : Juin 2015**

# REMERCIEMENTS

*Ce travail a été effectué au département d'électronique, faculté des sciences et de la technologie*

*Nous remercions en premier lieu ALLAH le tout puissant et miséricordieux, de nous avoir donné le courage, la force et la patience d'accomplir ce travail "EL HAMDOULI ALLAH".*

*Nous remercions M<sup>me</sup> Bouatmane S. à l'université de Mohamed Assadik Ben Yahia de Jijel pour avoir assuré notre encadrement ainsi que pour son aide précieuse .*

*Nous tenons également à remercier le président et les membres de jury pour nous avoir fait l'honneur d'évaluer notre travail.*

*Nos tenons à remercier vivement tous les enseignants du département d'électronique pour leurs efforts considérables.*

*Nous remercions s'adressent également à tous les amis qui ont contribué de près ou de loin pour réaliser ce travail et créer une ambiance d'étude*

# *Dédicace*

*Je dédie ce modeste travail à :*

*Ma chère mère*

*Mon chère Père*

*Mes sœurs*

*Mes frères*

*A tous mes amis*

*A tous mes enseignants*

*A tous mes collègues de la promotion 2015*

*“Loubna”*

# *Dédicace*

*Je dédie ce modeste travail à :*

*Ma chère mère*

*Mon chère Père*

*Mes frères*

*Mes sœurs*

*A tous mes amis*

*A tous mes enseignants*

*A tous mes collègues de la promotion 2015.*

*“Rafika”*

**TABLE DES MATIERES**

Remerciement .....	i
Dédicace.....	ii
Table des matières.....	iv
Liste des figures.....	vii
Liste des tableaux.....	ix
Introduction générale.....	1

**Chapitre I: Segmentation d'images : Généralité**

<b>I.1 Introduction .....</b>	<b>3</b>
I.2 Définition d'une image.....	4
I.3 Définition d'un histogramme .....	5
I.5 Définition d'une région .....	5
I.4 La segmentation d'images .....	6
I.6 Quelques exemples d'applications.....	7
I.7 Différences approches de la segmentation des images .....	8
I.7.1 Approche contour .....	9
I.7.1.1 Principe .....	9
I.7.1.2 La détection de contours .....	10
I.7.1.3 Le gradient d'une image .....	11
I.7.1.4 Pré -filtrage par l'opérateur de Sobel .....	12
I.7.2 Approche région .....	13
I.7.2.1 Segmentation par croissance des régions .....	14
I.7.2.2 Segmentation par division/fusion .....	14
I.7.2.3 Segmentation d'image par classification.....	15
a .Segmentation par seuillage .....	15
b. Algorithme K-Means .....	16
<b>I.8 Conclusion .....</b>	<b>17</b>

**Chapitre II : Segmentation Markovienne et stratégie bayésienne**

**II.1 Introduction** ..... 18

II.2 Définition du problème de la classification d'images..... 19

II.3 Stratégie Bayésienne ..... 19

II.4 Modélisation du champ des observations..... 21

II.5 Modélisation Markovienne du champ des classes ..... 22

    II .5.1 Principe du champ de markov ..... 22

    II.5.2 Système de voisinage et de clique ..... 23

        II.5.2.1 Système de voisinage ..... 23

        II.5.2.2 Système de clique..... 23

II.6 Equivalence entre champ de Markov et champ de Gibbs ..... 24

II.7 Fonction d'énergie globale ..... 26

II.8 Optimisation de la probabilité a posteriori ..... 27

    II.8.1 L'échantillonneur de Gibbs ..... 28

    II.8.2 L'algorithme de Métropolis ..... 28

    II.8.3 Algorithme *ICM (Itérative Conditional Mode)*..... 29

**II.9 Conclusion**..... 30

**Chapitre III : Testes et résultats**

**III.1 Introduction** ..... 31

III.2 Algorithme *ICM* ..... 32

    III.2.1 Exemple du seuillage ..... 32

    III.2.2 Exemple sur l'algorithme de k-Means ..... 33

III.3 Critères d'évaluation ..... 34

    III.3.1 Evaluation supervisée ..... 34

    III.3.2 Evaluation non supervisée ..... 34

III.4 La mesure *PRI (Probabiliste rand Index)* ..... 35

III.5 Résultats et discussion ..... 35

    III.5.1 Les images de synthèse ..... 35

---

III.5.2 Les images réelles .....	41
<b>III.6 Conclusion</b> .....	44
<b>Conclusion générale</b> .....	45
<b>Bibliographique</b> .....	47

**LISTE DES FIGURES**

<b>Figure I.1</b>	Image numérique .....	4
<b>Figure I.2</b>	Illustration d'un pixel.....	4
<b>Figure I.3</b>	Histogramme d'une image.....	5
<b>Figure I.4</b>	Exemple de segmentation d'image couleur.....	6
<b>Figure I.5</b>	Différentes régions d'une image .....	7
<b>Figure I.6</b>	Image par résonance magnétique (IRM) du cerveau .....	7
<b>Figure I.7</b>	Images satellite .....	8
<b>Figure I.8</b>	Exemple d'images en contrôle de qualité .....	8
<b>Figure I.9</b>	Détection de contours sur Lena .....	10
<b>Figure I.10</b>	Illustration de contours à fermer sur Lena.....	10
<b>Figure I.11</b>	Gradient d'une image découpé en deux zones distinctes.....	11
<b>Figure I.12</b>	Dérivées partielles selon $x$ et $y$ .....	13
<b>Figure I.13</b>	Les 3 masques de dérivation de Sobel .....	13
<b>Figure I.14</b>	Principe de la croissance des régions .....	14
<b>Figure I.15</b>	La segmentation par le seuillage.....	16
<b>Figure II.1</b>	Les cliques associées à deux systèmes de voisinage en dimension 2.....	24
<b>Figure III.1</b>	Segmentation d'une image par le seuillage.....	32
<b>Figure III.2</b>	Segmentation d'une image par k-Means .....	33
<b>Figure III.3</b>	Organigramme de la méthode <i>ICM</i> utilisée .....	35
<b>Figure III.4</b>	Images de synthèse.....	36
<b>Figure III.5</b>	Illustration des résultats obtenus sur l'image (1) par les différents algorithmes.....	38
<b>Figure III.6</b>	Illustration des résultats obtenus sur l'image (2) par les différents algorithmes.....	39
<b>Figure III.7</b>	Illustration des résultats obtenus sur l'image (3) par les différents algorithmes.....	40
<b>Figure III.8</b>	Images réelles utilisées.....	42
<b>Figure III.9</b>	Illustration des résultats obtenus sur l'image (1) par les différents des deux algorithmes <i>ICM</i> et K-Means.....	42



<b>FigureIII.10</b>	Illustration des résultats obtenus sur l'image (2) par les différents de deux algorithmes <i>ICM</i> et <i>K-Means</i> .....	43
<b>FigureIII.11</b>	Illustration des résultats obtenus sur l'image (3) par les différents des deux algorithmes <i>ICM</i> et <i>K-Means</i> .....	44



**Liste des tableaux**

<b>Tableaux III.1</b> Les valeurs de variance ( $\sigma$ ) .....	37
<b>Tableaux III.2</b> Les résultats <i>PRI</i> obtenus.....	40
<b>Tableaux III.3</b> Influence du bruit.....	41
<b>Tableaux III.4</b> Effet de paramètre $\beta$ avec $\sigma = 3$ pour l'image (1), $\sigma = 15$ pour l'image (2)et $\sigma = 0.3$ pour l'image (3).....	41
<b>Tableaux III.5</b> Les résultats <i>PRI</i> obtenus .....	44



***Introduction***

---

***Générale***

## ***Introduction Générale***

La segmentation d'une image est souvent nécessaire pour fournir une description compacte et opportune de son contenu. Elle consiste en l'affectation des pixels à des sous ensembles homogènes et disjoints, formant une partition de l'image. Les pixels qui appartiennent à une même région partagent une propriété commune, dite critère d'homogénéité de région. Les méthodes de segmentation se divisent en deux catégories distinctes:

- Les méthodes de segmentation en contours.
- Les méthodes de segmentation en régions.

Pour la première catégorie, les méthodes orientées contour sont bien connues pour leur coût de calcul faible cependant elles sont très sensibles au bruit et nécessitent des post-traitements complexes pour le chaînage et la fermeture de contours. Les méthodes orientées région utilisent les propriétés de surface pour grouper les pixels ayant les mêmes propriétés, dans des parties connexes et disjointes. Comparées aux méthodes orientées contour, les méthodes orientées région sont plus stables et moins sensibles au bruit. Cependant, leur efficacité dépend fortement de la sélection des graines initiales de régions. De plus, cette approche ne facilite pas la parallélisations et la distribution des traitements. Ceci rend le coût de calcul très élevé et ne permet pas à ces méthodes d'être utilisées pour des applications temps réel.

Dans ce mémoire nous avons adapté une méthode de segmentation en se basant sur les champs aléatoires de Markov et sur la distribution de Gibbs. Les champs aléatoires de Markov fournissent un cadre de recherche permettant à l'information spatiale d'être incorporée. Le modèle des champs aléatoires de Markov et la distribution de Gibbs exploitent la dépendance entre les pixels voisins dans les images. La propriété la plus importante de ces modèles est que les informations a priori et les données observées peuvent être combinées à travers l'utilisation de fonctions d'énergie appropriées. L'affectation de chaque classe dans la segmentation markovienne est effectuée en se basant sur les données observées et le modèle a priori. Le critère populaire utilisé pour trouver la classe la plus probable est le maximum de

Distribution a posteriori (MAP). Cela correspond à minimiser la solution d'une fonction d'énergie.

Plusieurs algorithmes sont utilisés pour trouver une solution itérative à ce problème tel que des algorithmes de relaxation stochastique et des algorithmes de relaxation déterministes dont l'algorithme *ICM (Iterative Conditional Mode)*. La méthode de relaxation introduit une adaptation itérative d'appartenance de chaque pixel à une classe donnée en se basant sur l'information contextuelle spatiale du voisinage d'un pixel.

Ce mémoire est organisé en trois chapitres qui nous permettront de présenter les différents aspects de notre travail.

Dans le premier chapitre, nous présenterons les différentes approches de la segmentation d'image en générale, et brièvement un état de l'art des principales méthodes de segmentation dans le traitement d'image.

Au second chapitre nous présenterons une méthode de segmentation basée sur les champs aléatoires de Markov et la distribution de Gibbs.

Enfin, le troisième chapitre présente l'implémentation et l'application de l'algorithme *ICM* sur les différentes images synthétiques et réelles et l'évaluation des résultats expérimentaux obtenus.

Nous terminerons ce mémoire par une conclusion.

***CHAPITRE I :***

***SEGMENTATION DES :***  
***IMAGES GÉNÉRALITE***

## I.1 Introduction

L'analyse et l'interprétation des images sont actuellement très utilisées dans différents domaines allant de l'aide au diagnostic en médecine, à la navigation autonome des véhicules en passant par la reconnaissance des visages, le contrôle de qualité des produits manufacturables (bois, tissus, verre, pièces mécaniques)... [1].

Dans un système d'analyse des images ; la segmentation joue un rôle prépondérant dans le traitement d'image. Elle est réalisée avant les étapes d'analyse et de prise de décision dans plusieurs processus d'analyse d'image, tel que la détection des objets. Elle aide à localiser et à délimiter les entités présentes dans l'image. Le but de la segmentation d'image est de faciliter l'extraction des éléments qui la composent. Pour cela, plusieurs approches ont été proposées.

Dans ce premier chapitre, nous présentons une brève revue de ces approches.

## I.2 Définition d'une image

Une image numérique est un tableau de valeurs [2]. Chaque case de ce tableau, qui stocke une valeur, se nomme un pixel. On peut la décrire par une fonction  $f(x, y)$  où :

$f(x, y)$  : La couleur ou le niveau de gris en ce point.

$x, y$  : Coordonnées cartésiennes d'un point de l'image.

$f$ : Est la fonction d'intensité lumineuse définie dans un domaine borné.

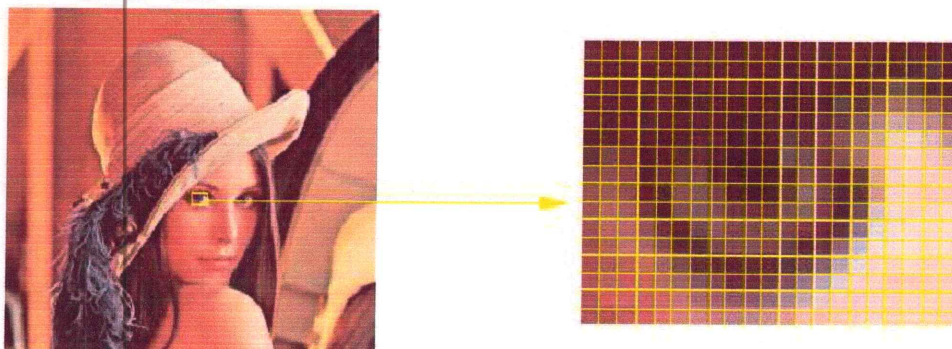


Figure I.1 Image numérique [2].

Un pixel est l'unité de base d'une image numérique. Il constitue l'unité minimale adressable par le contrôleur vidéo. A chaque pixel est associée une couleur, elle-même décomposée en 3 composantes primaires qui sont le Rouge, le Vert et le Bleu (RGB) ou une valeur de niveau de gris.

Les pixels ont une forme rectangulaire proche du carré, comme le montre le schéma suivant :

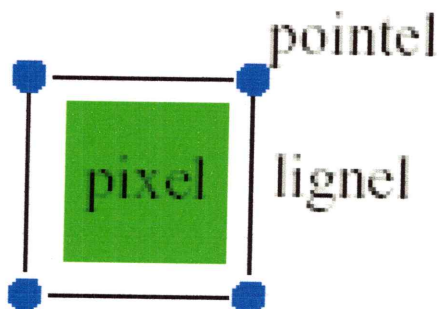


Figure I. 2 Illustration d'un pixel.

Ici, la couleur associée est le vert et on voit bien la forme rectangulaire voire carrée de cet élément de base d'une image numérique.

### I.3 Définition d'un histogramme

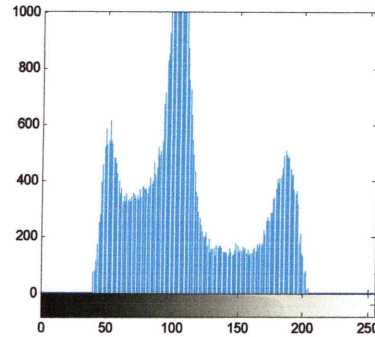
Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse. Par convention un histogramme représente le niveau d'intensité en abscisse en allant du plus foncé (à gauche) au plus clair (à droite) [3]. Ainsi, l'histogramme d'une image en 256 niveaux de gris sera représenté par un graphique possédant 256 valeurs en abscisses, et le nombre de pixels de l'image en ordonnées.

Prenons par exemple l'image suivante composée de niveaux de gris,





(a)



(b)

Figure I.3 (b) est l'histogramme de l'image (a).

#### I.4 La segmentation d'image

La segmentation consiste à diviser l'ensemble de pixels en régions connexes, homogènes et bien séparées. Ces régions possèdent une certaine uniformité pour une ou plusieurs caractéristiques (intensité, couleur, texture, ...) et sont différentes pour au moins une de ses caractéristiques des régions voisines [4].

Formellement, la segmentation d'une image numérique  $I$  consiste à chercher une partition de  $I$  en un sous-ensemble  $R = \{R_1, R_2, \dots, R_n\}$  telle que :

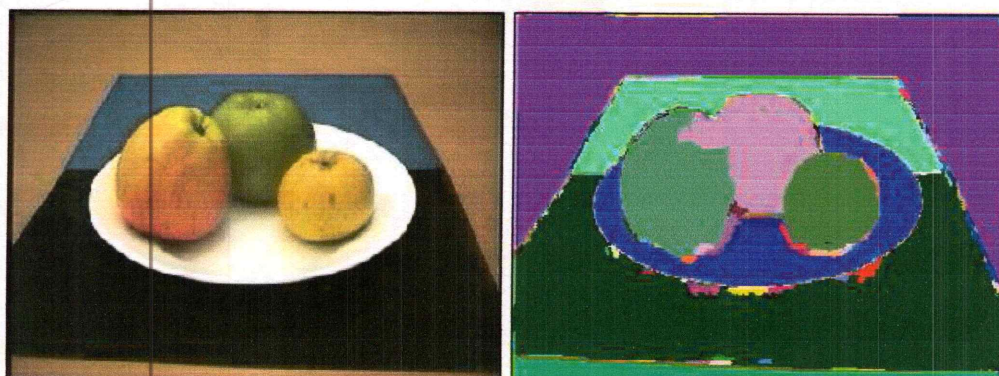
$$\forall i, R_i \neq \emptyset$$

$$\forall i \neq j, R_i \cap R_j = \emptyset$$

$$I = \cup_i R_i$$



Voici un exemple de segmentation d'images dans la figure I.4 permettant de montrer ce que l'on entend par régions homogènes.

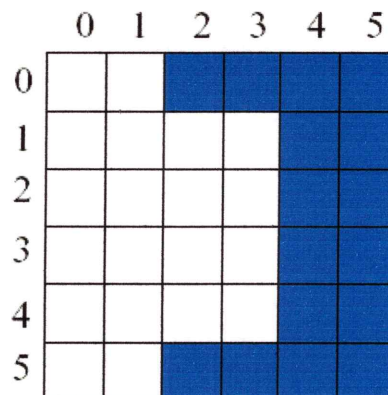


**Figure I.4 Exemple de segmentation d'image couleur [2].**

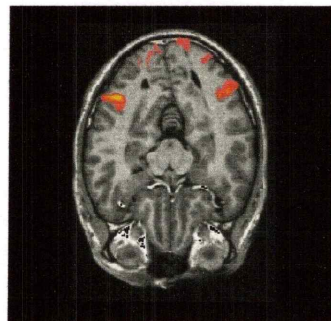
Sur cette image, on voit bien que chaque objet de l'image se voit attribué une couleur, d'où la séparation en régions dites homogènes. Cependant on voit qu'il peut y avoir des défauts de reconnaissance et donc y avoir des confusions entre les régions comme c'est le cas ici entre la pomme du milieu et celle se situant à sa gauche car elles ont une zone en violet commune [5].

### I.5 Définition d'une région

La notion de région dans le traitement d'images, comme évoquée ci-dessus, est de regrouper des zones possédant les mêmes caractéristiques (dans notre cas, la couleur). C'est-à-dire que si plusieurs pixels adjacents s'avèrent être de couleur identique alors la zone qu'ils forment est une région. Ci-après, une illustration montrant deux régions de pixels différentes [2] :

**Figure I.5 Différentes régions d'une image [2].**

### I.6. Quelques exemples d'applications

**Figure I.6 Image par résonance magnétique (IRM) du cerveau [6].**

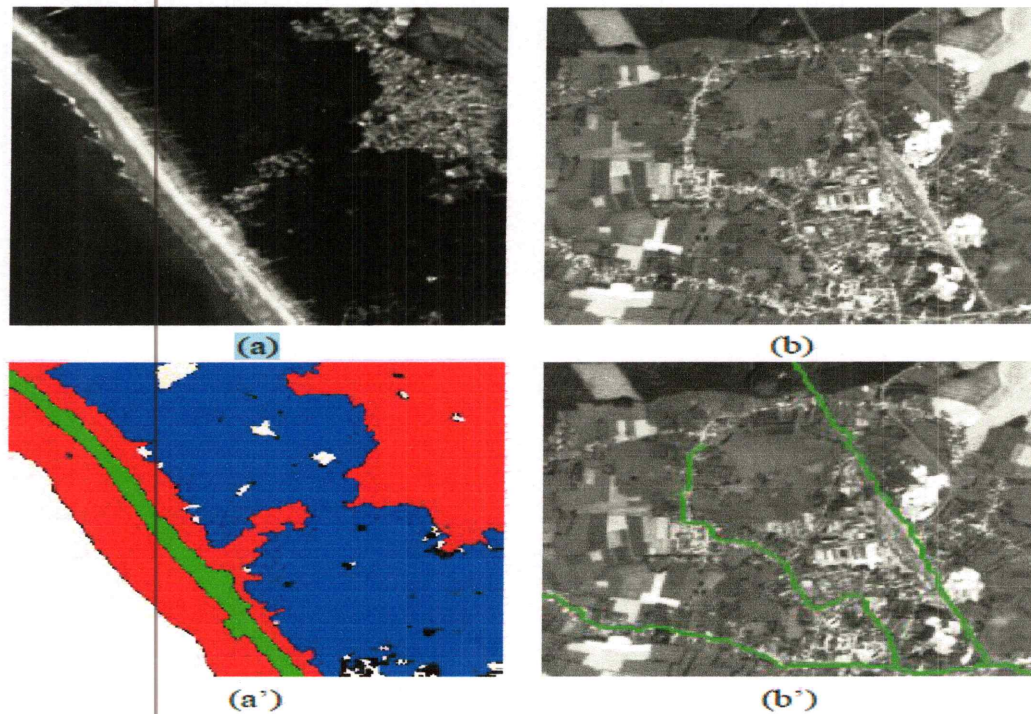


Figure I.7 Images satellite [6].

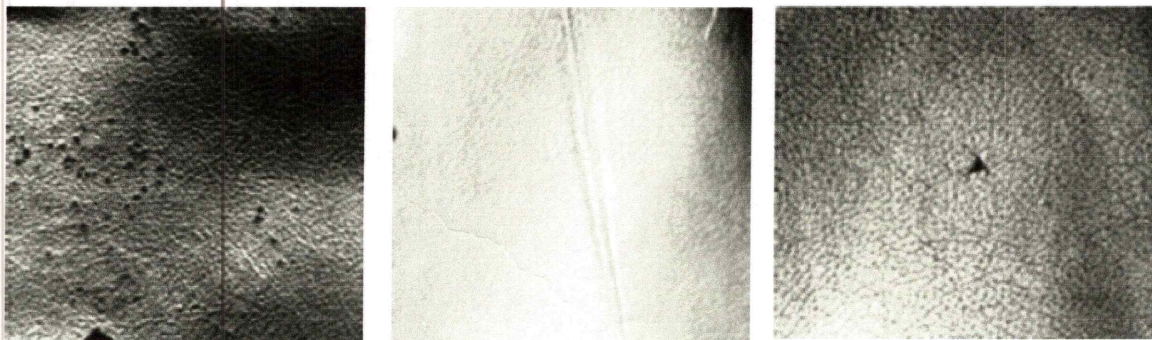


Figure I.8. Exemple d'images en contrôle de qualité [6].

### I.7 Différentes approches de la segmentation des images

La segmentation fait référence aux notions de différence et de similarité perçues par le système visuel humain. Ceci donne naissance à deux approches couramment qualifiées d'approche 'frontière' et d'approche 'région' [7]. L'approche région s'attache à faire apparaître des régions homogènes selon un critère (niveaux de gris ou texture), alors que

l'approche frontière tente de trouver des contours ou frontières de régions présentant une variation rapide du même critère.

Un algorithme de segmentation s'appuie donc sur :

1. la recherche de discontinuités afin de mettre en évidence les contours ;
2. la recherche d'homogénéité locale pour définir les régions ;
3. ou encore sur la coopération des deux principes.

### **I.7.1. Approche contour**

#### **I.7.1.1 Principe**

La segmentation par approche contours [8, 9] s'intéresse aux contours de l'objet dans l'image. La plupart des algorithmes qui lui sont associés sont locaux, c'est-à-dire qu'ils fonctionnent au niveau du pixel.

Des filtres détecteurs de contours sont appliqués à l'image et donnent généralement un résultat difficile à exploiter sauf si les images sont très contrastées. Les contours extraits sont la plupart du temps morcelés et peu précis, il faut alors utiliser des techniques de reconstruction de contours par interpolation ou connaître a priori la forme de l'objet recherché. Formellement, ce type d'algorithme est proche des méthodes d'accroissement de régions fonctionnant au niveau du pixel. Ces techniques purement locales sont en général trop limitées pour traiter des images bruitées et complexes.

Dans la segmentation par approches contours, il y a deux problématiques à résoudre, à savoir :

- Caractériser la frontière entre les régions.
- Fermer les contours.



Figure I.9 Détection de contours sur Lena [2].



Figure I.10 Illustration de contours à fermer sur Lena [2].

### I.7.1.2 La détection de contours

La détection de contours est une étape préliminaire à de nombreuses applications de l'analyse d'images. Les contours constituent des indices riches, au même titre que les points d'intérêts, pour toute interprétation ultérieure de l'image [10].

Les contours dans une image proviennent des :

- Discontinuités de la fonction de réflectance (texture, ombre).
- Discontinuités de profondeur (bords de l'objet).

Pour les détecter, il existe deux types d'approches :

- Approche gradient : Détermination des extrema locaux dans la direction du gradient.
- Approche Laplacien : Détermination des passages par zéro du Laplacien.

### I.7.1.3 Le gradient d'une image

Le gradient d'une image est le vecteur  $\nabla I(x, y)$  défini par [10] :

$$\nabla I(x, y) = \left( \frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y} \right) \quad (\text{I.1})$$

Il est donc caractérisé par un module  $m$  et une direction  $\emptyset$  dans l'image :

$$m = \sqrt{\left( \frac{\partial I(x, y)}{\partial x} \right)^2 + \left( \frac{\partial I(x, y)}{\partial y} \right)^2} \quad (\text{I.2})$$

$$\emptyset = \tan^{-1} \left( \frac{\partial I(x, y)}{\partial y} / \frac{\partial I(x, y)}{\partial x} \right) \quad (\text{I.3})$$

La direction du gradient maximise la dérivée directionnelle.

La dérivée de  $I(x, y)$  dans une direction donnée  $d$  s'écrit :

$$\nabla I(x, y) \cdot d. \quad (\text{I.4})$$

$$GI^t(x, y) = \nabla(I(x, y) * h(x, y)) = \nabla I(x, y) * h(x, y) = I(x, y) * \nabla h(x, y) \quad (\text{I.5})$$

Voici une illustration montrant le gradient d'une image découpée en deux zones distinctes :

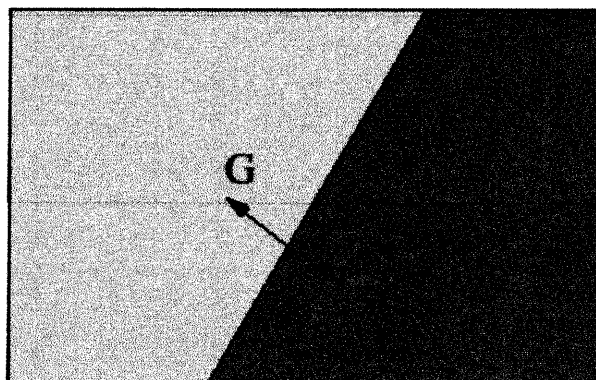


Figure I.11 Gradient d'une image découpée en deux zones distinctes [10].

Le gradient n'est autre qu'une dérivée vectorielle de l'image et permet de détecter les contours du fait que les contours correspondent à des discontinuités d'ordre 0 de la fonction d'intensité.

Cependant, le calcul de la dérivée nécessite un pré filtrage de l'image étudiée.

Ce pré-filtrage peut être réalisé à l'aide de l'opérateur de Sobel

#### I.7.1.4 Pré-filtrage par l'opérateur de Sobel

Pour filtrer nos images, nous devons utiliser des filtres que l'on appelle filtres étroits dont l'approche la plus courante est le filtre gradient dans lequel l'opérateur de Sobel est utilisé. On rappelle que le gradient, en un pixel d'une image numérique, est un vecteur caractérisé par son amplitude et sa direction. L'amplitude est directement liée à la quantité de variation locale des niveaux de gris [10].

La direction du gradient est orthogonale à la frontière qui passe au point considéré. La méthode la plus simple pour estimer un gradient est donc de faire un calcul de variation monodimensionnelle, *i.e.* en ayant choisi une direction donnée.

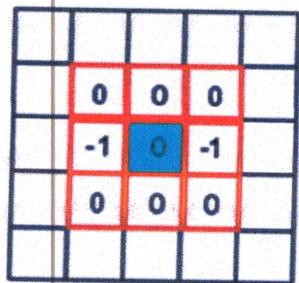
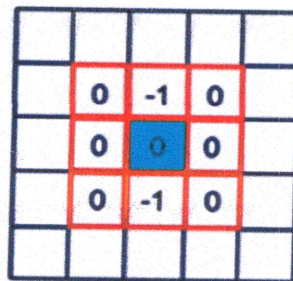
On a alors le schéma suivant :

$$G_d(x, y) = (I * W_d)(x, y) \quad (I.6)$$

Où  $W_d$  désigne l'opérateur de dérivation dans la direction  $d$  et  $*$  le produit de convolution.

Il existe de très nombreux opérateurs différents (Roberts, Sobel, Prewit, Kirsch,...) qui ont globalement les mêmes propriétés. Le gradient étant un vecteur, l'approche la plus classique pour estimer le gradient consiste à choisir deux directions privilégiées (naturellement celles associées au maillage, *i.e.* ligne et colonne) orthogonales, sur lesquelles on projette le gradient. En chaque point  $(x, y)$  de l'image, on peut donc calculer le vecteur gradient. Sa direction maximise la dérivée directionnelle et sa norme est la valeur de cette dérivée.

Appliquée à une image, considérée comme une fonction de deux variables, on peut définir deux dérivées partielles, suivant  $x$  (colonnes) et suivant  $y$  (lignes) :

Dérivée horizontale  $\left(\frac{\partial I(x,y)}{\partial x}\right)$ Dérivée verticale  $\left(\frac{\partial I(x,y)}{\partial y}\right)$ Figure I.12 Dérivées partielles selon  $x$  et  $y$  [10].

C'est ici que l'opérateur de Sobel entre en jeu car la dérivation augmentant le bruit, il a été nécessaire de créer des filtres plus robustes comme celui de Sobel. L'opérateur de Sobel, créé en 1972, est un filtre très populaire permettant d'utiliser des masques de dérivation dans les directions horizontales, verticales et obliques. Voici d'ailleurs les 3 masques de dérivation de l'opérateur de Sobel :

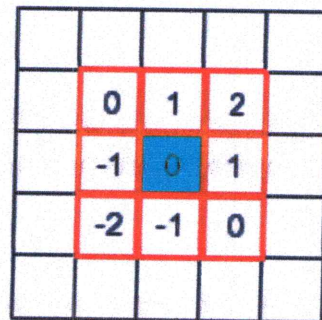
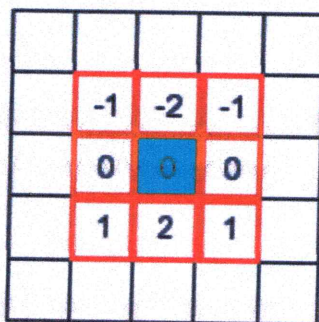
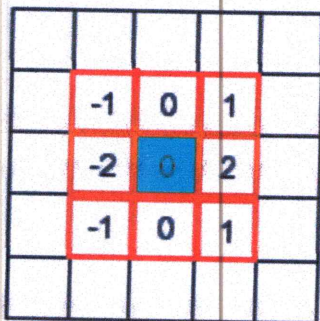


Figure I.13 Les 3 masques de dérivation de Sobel [10].

### I.7.2 Approche région

L'approche région cherche à regrouper les pixels en régions homogènes. Elle se caractérise par la mesure d'uniformité des régions construites dans l'image. Ces régions sont construites en évaluant la similarité entre les pixels ou entre un pixel et ceux d'une même région. On distingue les méthodes par croissance des régions, par division-fusion et par classification.



### I.7.2.1 Segmentation par croissance des régions

Cette approche consiste à choisir initialement un ensemble de pixels pour les considérer comme des germes ou noyaux des régions à partir desquels la croissance des régions s'effectuera. Chaque région doit respecter un critère d'uniformité ou d'homogénéité défini par exemple par la variance des niveaux de gris. La croissance des régions s'effectue par agrégation de nouveaux pixels aux germes initiaux. Elle consiste à fusionner successivement aux régions les pixels voisins qui sont similaires et à condition que les régions fusionnées continuent à vérifier le critère d'uniformité **figure(I.14)**. Notant que des régions adjacentes peuvent être fusionnées si leur fusion vérifie le critère d'uniformité et que de nouveaux germes peuvent être créés dans les régions qui ne peuvent pas être fusionnées avec les régions existantes [11].

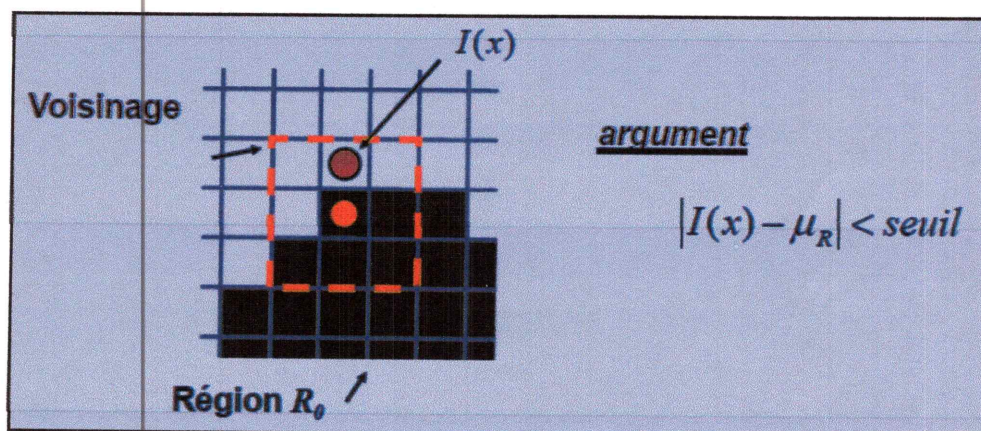


Figure I.14 Principe de la croissance des régions.

### I.7.2.2 Segmentation par division/fusion

Ce type de méthode consiste à diviser l'image. Considérée comme une région initiale. En régions de plus en plus petites. Le principe consiste à tester d'abord le critère d'homogénéité retenu sur l'image entière. Si le critère est valide. L'image est considérée comme segmentée ; sinon. L'image est découpée en zones plus petites et la méthode est réappliquée sur chacune des zones nouvellement obtenues.

La division peut se faire en quatre parties, en six parties, en polygones, etc. la méthode la plus connue est la méthode de *quadtree* [12] où chaque zone est divisée en 4. L'inconvénient de ces méthodes est que deux parties adjacentes peuvent vérifier le même critère sans avoir été regroupées dans la même région.

Pour éviter ce problème, Une procédure de fusion des petites régions similaires au sens d'un prédicat de regroupement est appliquée.

La fusion de régions est principalement fondée sur l'analyse d'un graphe d'adjacence de région qui analyse une image pré segmentée. Constituée d'un ensemble de régions. C'est une structure de données constituée d'un graphe non-orienté dont chaque nœud représente une région et chaque arête représente une adjacence entre deux régions. Le procédé consiste à fusionner deux nœuds reliés par une arête à condition qu'ils respectent un critère de fusion [13].

### **1.7.2.3 Segmentation d'image par classification**

La classification est l'une des procédures les plus utilisées en analyse des données. Elle permet de partitionner un ensemble de donnée multidimensionnelle en un ensemble de  $k$  classes disjointes. En segmentation d'image, Les données multidimensionnelles correspondent aux pixels de l'image ou chaque pixel est caractérisé par un vecteur d'attributs tels que les attributs de texture ou les composantes couleurs. **Chaque classe regroupe des pixels ayant des vecteurs de caractéristiques aussi similaires que possible.** Sachant que les pixels de deux classes distinctes ont des attributs très différents [14]. Contrairement aux méthodes précédentes, les techniques de segmentation par classification ne permet pas en compte la disposition spatiale des pixels et ne considèrent que le vecteur de caractéristiques. Elle identifie les classes de pixels présentes dans l'image et affectent à chaque pixels une étiquette indiquant la classe à laquelle il appartient. Ainsi les pixels appartenant à une classe peuvent former plusieurs **régions non adjacentes dans l'image mais partageant les même propriétés.** Une segmentation en région n'est obtenue qu'après analyse de la connexité des pixels dans l'image étiquetée.

On trouve dans la littérature une multitudes de méthodes de classification parmi lesquelles on peut citer, L'algorithme K-means et le seuillage

#### **a. Segmentation par seuillage**

Le seuillage est une technique de segmentation très populaire à cause de sa facilité de mise en œuvre et sa rapidité. Elle permet d'extraire les objets du fond de l'image. Dans le cas le plus classique, les pixels de l'image sont classés en deux classes par

l'intermédiaire d'un niveau de gris  $S$  appelé seuil. La première classe regroupe les pixels du fond et la deuxième classe regroupe les pixels de l'objet.

Soit l'image  $I(M * N)$ , supposons que  $f(x, y)$  représente le niveau de gris d'un pixel de coordonnées  $(x, y)$  [15 – 17].  $0 \leq x \leq M$ .  $0 \leq y \leq N$  et  $S$  est le seuil choisi.

Les pixels de l'objet sont ceux dont le niveau de gris est supérieur à  $S$  et les pixels dont le niveau de gris est inférieur à  $S$  appartiennent au fond. L'image segmentée  $G$  est définie pour chaque pixel de coordonnées  $(x, y)$  par :

$$g(x, y) = \begin{cases} 0 & \text{si } f(x, y) > S \\ 1 & \text{si } f(x, y) \leq S \end{cases} \quad (I.7)$$

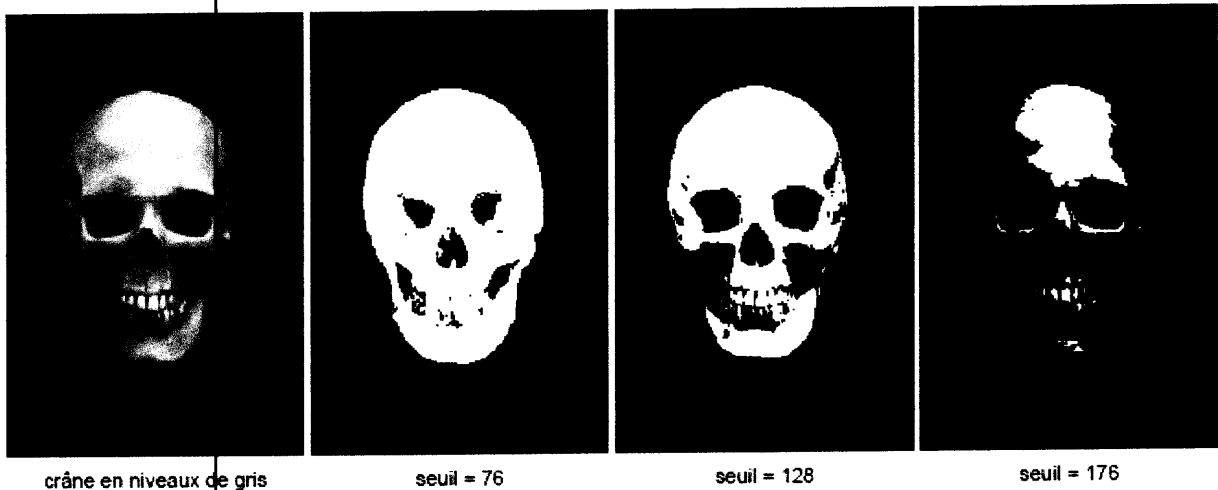


Figure I.15 la segmentation par le seuillage.

### b. Algorithme K-Means

Il vise à produire un partitionnement des pixels de manière à ce que les pixels d'une même classe soient semblables et les pixels issus de deux classes différentes soient dissemblables. L'idée principale est de définir  $K$  centroïdes, un pour chaque classe  $\{C_k\}_{1 \leq k \leq K}$ . Chaque classe  $C_k$  est ainsi caractérisée par son centre noté  $\mu_k$  et le nombre d'éléments  $N_k$ . L'algorithme K-means dans sa formulation originale cherche à minimiser une fonction :  $J = \sum_{k=1}^K \sum_{(x,y), i \in C_k} |f(x, y) - \mu_k|$  (I.8)

de coût global définie par :

Où  $f(x, y)$  représente le niveau de gris du pixels de coordonnées  $(x, y)$ .

Il se déroule selon les étapes suivantes :

1. Initialisation de chaque centre  $\mu_k$ .
2. Pour chaque pixel  $(x, y)$ , calculer la distance  $d(f(x, y), \mu_k)$  aux différents centres des classes  $\mu_k$ , et affecter à la classe la plus proche  $C_{l=\arg \min_k} d(f(x, y), \mu_k)$  avec

$$D(f(x, y), \mu_k) = (|f(x, y) - \mu_k|) \quad (\text{I.9})$$

3. Mise à jour de nombre de pixels et des centres  $\mu_k$  des classes ;

$$\mu_k = \frac{\sum_{(x,y) \in C_k} f(x,y)}{N_k} \quad (\text{I.10})$$

4. Arrêt si  $N_k = N_{k+1} \forall (x, y) \in C_k$ , sinon retour à l'étape 2.

Le principal inconvénient de cette méthode est que la classification finale dépend du choix de la partition initiale. Le minimum global n'est pas obligatoirement atteint, on est seulement certain d'obtenir la meilleure partition à partir du départ choisie [18].

De nombreuses variantes peuvent être rencontrées. Par exemple, au lieu de calculer le centre des classes, après avoir affecté tout les pixels, les centres de gravité peuvent être calculés immédiatement après chaque affectation. La méthode des K-means a été généralisée sous l'appellation de la "méthode des nuées dynamiques" [19].

### **I.8 conclusion**

Nous avons présenté dans ce chapitre quelques notions sur la segmentation des images. Il ressort que le domaine de la segmentation est tellement riche et diversifié qu'il est difficile de donner une étude exhaustive.

Deux grands approches pour la segmentation des images sont définies dans la littérature, il s'agit de l'approche contour et l'approche région. L'approche région contient un grand nombre de méthode dont la plus part d'entre elles se basent sur la classification des pixels.

***CHAPITRE II :***

***SEGMENTATION  
MARKOVIENNE ET  
STRATÉGIE BAYSIENNE***

$$P(X/Y) \equiv P(Y/X) P(X) \quad (\text{II.3})$$

Donc pour obtenir de la classe la plus probable " $\hat{x}$ " de l'équation (II.1) en utilisant la maximisation de la probabilité a posteriori il est, nécessaire de réaliser les deux étapes suivantes :

1. La première étape est la modélisation du champ des observations et celle du champ des classes, c'est-à-dire trouver le modèle statistique de  $P(Y = y/X = x)$  et le modèle de  $P(X = x)$ . Dans notre recherche, la modélisation des observations est basée sur le modèle gaussien. Alors que la modélisation du champ de classes est basée sur un champ markovien permettant de prendre en compte les dépendances statistiques spatiales entre sites adjacents. Ces deux modèles seront explicités dans les sections (II.4) et (II.5).
2. La deuxième étape est l'optimisation de la probabilité à posteriori. Cette optimisation sera effectuée en utilisant une approche de type déterministe réalisée par l'algorithme *ICM*. Cette approche sera décrite dans la section (II.6) [21].

#### II.4. Modélisation du champ des observations

La densité de probabilité conditionnelle  $P(Y = y/X = x)$  de l'équation (II.3) peut être modélisée par une loi gaussienne. Dans le cas d'une image la probabilité  $P(Y_s = y_s/X_s = k)$  (avec  $y_s$  désigne la valeur observée du site "s") Peut être modélisée sous la forme suivante [21]:

$$P(Y_s = y_s/X_s = k) = pi(y_s/k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(y_s - \mu_k)^2}{2\sigma_k^2}} \quad (\text{II.4})$$

$\sigma_k^2$  : La variance de la classe  $k$ .

$\mu_k$  : La moyenne de la classe  $k$ .

La détermination de cette probabilité est liée à l'ensemble des moyennes et des variances des classes. Cet ensemble forme le vecteur des paramètres à estimer.



Dans le cas des images à plusieurs canaux, on utilise des lois gaussiennes multidimensionnelles. La probabilité  $P(Y_s = y_s / X_s = k)$  sera donnée par l'expression suivante :

$$P(Y_s = y_s / X_s = k) = \pi(y_s / k) = \frac{1}{(2\pi|A_k|)^{M/2}} e^{-\frac{1}{2}(y_s - \mu_k)^T A_k^{-1} (y_s - \mu_k)} \quad (\text{II.5})$$

Où

$y_s$  : Le vecteur constitué par les valeurs du pixel "s" sur les  $M$  canaux utilisés.

$A_k$  : La matrice de covariance entre les canaux pour la classe  $k$ .

$\mu_k$  : Le vecteur des moyennes pour la classe  $k$ .

L'ensemble des matrices de covariance et des moyennes de classes désigne le vecteur  $\alpha$  à estimer. En supposant que  $P(Y_s = y_s / X_s = k)$  soit strictement supérieure à zéro, on pose :

$$E_1(y/x) = -\ln p(Y_s = y_s / X_s = k) \quad (\text{II.6})$$

$E_1$  sera considéré comme une composante de l'énergie globale qui sera utilisée pour déterminer la classe la plus probable.

Selon l'utilisation de l'expression exacte de la probabilité conditionnelle n'améliore pas d'une façon significative les résultats, en regard de la complexité et du temps de calcul supplémentaire. Par ailleurs, l'apport de l'information contextuelle est introduit tout naturellement dans le modèle markovien du champ des classes. Ce dernier point sera développé dans la section (II.5) [22].

## II.5. Modélisation markovienne du champ des classes

Généralement, dans les méthodes de classification conventionnelles telles que la méthode du maximum de vraisemblance, la probabilité à priori est supposée identique pour toutes les classes.

Cela est incorrect en général, en ce qui concerne les objets d'une scène naturelle. De plus, le produit final de classification peut inclure des régions isolées à cause de la

présence du bruit. Afin d'éviter ces problèmes, l'utilisation de l'information du voisinage permet de jouer un grand rôle dans la détermination des probabilités a priori.

Il s'agit d'assimiler le champ des classes à un champ markovien. Avant d'utiliser le champ markovien, nous allons donc définir le principe du champ de Markov et un système de voisinage adapté aux images.

### II.5.1.Principe du champ de Markov

$X$  est un champ de Markov si la probabilité conditionnelle locale en un site n'est fonction que de la configuration du voisinage du site considéré. Ce qui s'exprime de façon formelle par :

$$P(X_s = x_s / x^s) = P(X_s = x_s / x_t, t \in \vartheta_s) \quad (II.7)$$

$x_s$ : la valeur du descripteur prise au site  $s$  et  $x^s = (x_t)_{t \neq s}$  la configuration de l'image excepté le site  $s$ .

Ainsi, le niveau de gris en un site ne dépend que des niveaux de gris des pixels voisins de ce site.

On dit alors que  $X$  est un champ aléatoire de Markov (MRF) sur  $S$  relativement au système de voisinage  $V_s$ . La propriété de " markoviennité " exprime que le comportement de la variable aléatoire portée par un site est entièrement déterminé par les réalisations des variables aléatoires voisines.

Ainsi dans la pratique, l'introduction des champs markoviens permet la modélisation d'ensembles de variables aléatoires dont les interactions mutuelles résultent uniquement de la combinaison d'interactions locales

### II.5.2. Système de voisinage de et cliques

#### II.5.2.1.Système de voisinage

En analyse d'images, l'image est représentée par une grille :

$$S = \{s(i, j); 0 \leq i \leq n - 1; 0 \leq j \leq m - 1\} \quad (II.8)$$



Rectangulaire finie bidimensionnelle de taille  $n \times m$ , contenant un ensemble de "sites"  $s \in S$ , appelés :  $V = V_{S \in S}$  vérifiant les propriétés suivantes :

$S \notin V_s$  : I.e. un site n'est pas voisin à lui-même.

$S \in V_t \Leftrightarrow t \in V_s$  : Qui signifie que la relation de voisinage est symétrique.

A titre d'exemple, l'ensemble des "voisins" d'un site  $s$  peut être défini comme un ensemble de sites situés dans un rayon  $r$  :

$$V_s = \{t \in S / d(s, t) \leq r; t \neq s\} \quad (\text{II.9})$$

Où  $d(s, t)$  est la distance euclidienne entre le pixel  $s$  et le pixel  $t$ , et  $r$  est à valeur dans l'ensemble des entiers [23].

Notons que les systèmes de voisinage les plus usuels, représentés en Figure II.1, sont :

- Système "4 - connexité", également appelé voisinage du premier ordre.
- Système "8 - connexité", également appelé voisinage du second ordre.

### II.5.2.2. Clique

À partir d'un système de voisinage, un système de cliques peut être déduit : une clique est soit un singleton de  $S$ , soit un ensemble de sites tous voisins les uns des autres.

En fonction du système de voisinage utilisé, le système de cliques sera différent et fera intervenir plus ou moins de sites comme illustré sur la figure II.1.

On notera  $C$  l'ensemble des cliques relatif à  $V$ , et  $C_k$  l'ensemble des cliques de cardinal  $k$ .

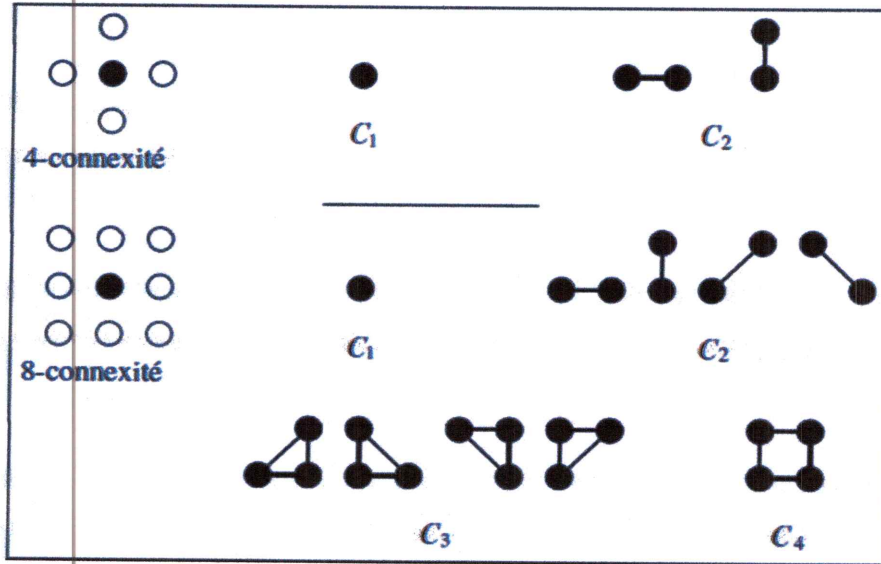


Figure II.1. Les cliques associées à deux systèmes de voisinage en dimension 2.

Les interactions locales entre niveaux de gris (ou descripteurs) de sites voisins peuvent alors s'exprimer comme un potentiel de clique.

Soit  $c$  une clique, on lui associe le potentiel  $U_c$  dont la valeur dépend des niveaux de gris (ou descripteurs) des pixels constituant la clique.

### II .6. Equivalence entre champs de Markov et champs de Gibbs

La modélisation markovienne prend toute sa puissance grâce au théorème que nous allons voir maintenant. En effet, celui-ci permettra d'accéder aux expressions des probabilités conditionnelles locales.

Il nous faut au préalable définir un certain nombre de notions relatives aux mesures et champs de Gibbs.

- **Définition champ de Gibbs**

Champ de Gibbs de potentiel associé au système de voisinage  $V$  : c'est le champ aléatoire  $X$  dont la probabilité est une mesure de Gibbs associée au système de voisinage  $V$ , ce qui implique :

$$P(X = x) = \frac{1}{z} e^{-U(x)} = \frac{1}{z} e^{\sum_{c \in C} U_c(x)} \tag{II.10}ml$$

Avec :

$$U(x) = \sum_{c \in C} U_c(x) \quad (\text{II.11})$$

Où  $C$  est le système de cliques associé au système de voisinage  $V$  de  $U^2$

$$Z = \sum_{x \in \Omega} e^{-U(x)} \quad (\text{II.12})$$

est une constante de normalisation appelée fonction de partition de Gibbs. En pratique, il est quasi impossible de calculer cette constante à cause du très grand nombre de configurations possibles.

Ne serait-ce que dans le cas d'une image binaire ( $Card(E) = 2$ ) et de taille =  $512 \times 512$ , on a  $2^{262144}$  configurations possibles.

L'énergie globale d'un champ de Gibbs possède donc la propriété de se décomposer sous forme d'une somme d'énergies locales, qui comme on le verra par la suite permettront d'accéder aux probabilités conditionnelles locales.

Notons ici que plus une configuration d'un champ de Gibbs a une énergie faible, plus elle est probable.

- **Le théorème d' Hammersley – Clifford** : [24] Permet d'exprimer la fonction d'énergie  $U(x)$  en fonction des valeurs de " $x$ " sur "les cliques" du voisinage ( $V_s$ ) utilisé [25]. La fonction d'énergie  $U(x)$  est obtenue sous la forme d'une somme de potentiels locaux  $V_c(x)$  sur les cliques  $c$  de  $C$ , de la façon suivante :

$$U(x) = E_2(x) = \sum_{c \in C} V_c \quad (\text{II.13})$$

Où  $V_c$  est la fonction de potentiel associée à chaque clique  $c$ .  $C$  est l'ensemble de cliques correspondant au voisinage choisi. La fonction de potentiel permet de caractériser la différence ou la ressemblance entre les pixels de chaque clique.

En utilisant un système de voisinage des quatre ou huit plus proches voisins et en considérant seulement les cliques à deux pixels, la fonction de potentiel  $V_c$  est choisie comme suite [26]:

- $V_c = \beta$  Si les deux pixels de la clique ont des valeurs différentes.
- $V_c = -\beta$  Si les deux pixels de la clique ont la même valeur.

Où

$\beta$  est un paramètre positif à estimer.

Cette fonction de potentiel impose une contrainte de connectivité spatiale sur la classification. L'affectation du potentiel permet de donner une probabilité élevée pour les paires de pixels ayant des étiquettes identiques et une probabilité faible pour les paires ayant de pixels ayant des étiquettes distinctes.

L'énergie de  $E_2(x)$  de l'équation (II.13) peut être considérée comme un terme énergétique comparable à  $E_1(y/x)$  de l'équation (II.6).

## II.7. Fonction d'énergie globale

Comme nous l'avons vu dans les équations précédentes (II.5) et (II.10).  $P(Y/X)$  et  $P(X)$  sont des fonctions exponentielles. Si nous supposons que :

$$E_1(y/x) = -\ln P(Y = y/X = x) \quad (\text{II.14})$$

et

$$E_2 = -\ln P(X = x) \quad (\text{II.15})$$

L'estimateur du maximum a posteriori MAP de l'équation (II.1) devient :

$$\hat{x} = \text{Arg max}_{x \in \Omega} (\exp - \{E_1(y/x) + E_2(x)\}) \quad (\text{II.16})$$

Lorsque, seules, les fonctions d'énergie sont considérées, nous avons:

$$\hat{x} = \text{Arg min}_{x \in \Omega} (E_1(y/x) + E_2(x)) \quad (\text{II.17})$$

Donc, la configuration de la classe la plus probable est celle qui présente une énergie globale minimale  $E_G$ , avec :

$$E_G(x, y) = E_1(y/x) + E_2(x) \quad (\text{II.18})$$

Nous constatons que la fonction globale d'énergie a deux composantes : L'une utilise l'intensité des régions obtenues à partir des données observées et l'autre impose la connectivité spatiale entre les classes.

Par conséquent, l'estimation de la classification ne dépend pas uniquement de l'intensité des images, mais aussi des propriétés spatiales imposées par le modèle du champ aléatoire de Markov.

L'utilisation de la distribution de Gibbs revient à apporter l'information à priori à travers le modèle énergétique Markovien.

## **II.8. Optimisation de la probabilité a posteriori**

La définition d'un champ de Markov passe par la définition de sa fonction d'énergie  $U$ . Celle-ci nécessite la définition d'un système de voisinage, qui définit alors le système de cliques, et de fonctions de potentiel associées aux cliques.

Ces fonctions de potentiel permettent d'accéder à la probabilité globale d'une configuration, et aux probabilités conditionnelles locales.

Le problème qui se pose alors est, étant défini un champ de Markov, comment pouvons-nous réaliser le tirage d'une configuration en suivant la loi de probabilité de Gibbs caractéristique de ce champ.

Plusieurs algorithmes ont été proposés pour synthétiser des réalisations d'un champ de Markov parmi ces algorithmes, on trouve :

- L'échantillonneur de Gibbs.
- L'algorithme de Métropolis.
- *ICM (Itérative Conditional Mode)*

### **II.8.1.L'échantillonneur de Gibbs**

Est un algorithme de relaxation probabiliste, qui se propose de réaliser ce tirage par une méthode itérative, convergeant vers une configuration suivant la loi de Gibbs.

A l'itération  $n$ , la construction de la réalisation courante se fait à l'aide des étapes suivantes [27] :

Choix d'un site  $s$  (par tirage selon une loi uniforme sur  $E$ )

➤ Calcul de la probabilité conditionnelle :

$$P(X_s = x_s | X_r = x_r, r \in V_s) = \frac{e^{-U(x_s, x_r, r \in V_s)}}{\sum_{e \in E} e^{-U(e, x_r, r \in V_s)}} \quad (\text{II.19})$$

Suivant la configuration des sites  $r \in V_c$  à l'itération précédente.

➤ Mise à jour du site  $s$  par tirage aléatoire selon :

$$P(X_s = x_s | X_r = x_r, r \neq s) \quad (\text{II.20})$$

Chaque site  $s$  est visité une infinité de fois, on peut montrer que les réalisations générées sont des réalisations de la loi de Gibbs, indépendamment de la configuration initiale utilisée pour l'algorithme.

### II.8.2. L'algorithme de Métropolis

L'algorithme de Métropolis issu de la physique statistique, est également un algorithme de relaxation probabiliste [27]. Le principe est donc itératif, et repose sur les étapes suivantes :

- Choix d'un site  $s$  (par tirage selon une loi uniforme sur  $E$ ).
- Tirage aléatoire d'un descripteur  $\sigma \in E$  selon une loi uniforme.
- Calcul de la variation d'énergie induite par le changement d'état du site  $s$  (de son état à l'itération précédente à l'état  $\sigma$ ).

$$\Delta U = U_s(X_s = \delta | X_r = x_r^{n-1}, r \in V_s) - U_s(X_s = x_s^{n-1} | X_r = x_r^{n-1}, r \in V_s) \quad (\text{II.21})$$

**Si** ( $\Delta U < 0$ ) **Alors**

Le changement d'état est accepté pour le site  $s$ .

**Si non**

Le changement est accepté avec une probabilité  $U^{-\Delta U}$ .

Les critères de convergence sont similaires à ceux de l'échantillonneur de Gibbs, mais cette dernière intervient cependant plus tardivement car les transitions ne sont ici pas toujours acceptées. En revanche, l'itération est plus rapide que celle de l'échantillonneur de Gibbs, car la variation d'énergie ne se calcule qu'entre deux configurations données.

### II.8.3. Algorithme ICM (Itérative Conditional Mode)

Des algorithmes sous-optimaux ont été proposés pour accélérer la convergence, et c'est le cas par exemple de l'algorithme des modes conditionnels itérés.

Le principe est toujours itératif, mais cette fois-ci la modification du descripteur  $x_s$  du  $s$  se fait de manière déterministe [27].

On recherche ici une approximation du maximum a posteriori  $x$ , résultat de la convergence des itérations. Pour une itération  $k$  donnée, le déroulement de la mise à jour de l'ensemble d'une configuration se fait, pour tout site  $s$ , en mettant à jour  $x_s$  par :

$$\text{Arg max}_{e \in E} P(X_s = e | \hat{x}_r(k), r \in V_c) \quad (\text{II.22})$$

L'algorithme converge lorsque le nombre de changements d'états des sites est faible (par exemple en pourcentage de  $\text{Card}(S)$ ).

Si on montre que l'énergie globale de  $x$  diminue, on ne peut affirmer que la convergence aboutit à un optimum global, l'algorithme dépendant fortement de son initialisation. L'algorithme *ICM* sera expliqué en détail dans le chapitre III.

## II.9. Conclusion

Nous avons abordé dans ce chapitre les notions fondamentales du formalisme markovien. Nous avons vu comment se définissait un champ de Markov et comment l'équivalence champ de Markov - champ de Gibbs permettait de définir sa loi de probabilité en termes d'énergie.

Nous avons abordé les algorithmes de synthèse *ICM*, Métropolis et échantillonneur de Gibbs permettant d'échantillonner selon une loi de Gibbs, et le recuit simulé permettant de trouver une configuration d'énergie minimale.

***CHAPITRE III :***

***TESTS ET RÉSULTATS***



### III.1 Introduction

Dans ce chapitre, nous allons détailler et examiner l'efficacité de l'algorithme *ICM* sur des images synthétiques et des images réelles en termes de la segmentation. Cet algorithme a été développé par Besag [28] sous le nom *Iterated Conditional Modes (ICM)*.

Le processus de segmentation de cet algorithme consiste à maximiser la probabilité a posteriori en se basant sur les données observées et sur l'information du voisinage de chaque pixel segmenté. Cette information du voisinage est utilisée pour calculer la probabilité a priori basée sur un modèle simple de fonction appelé la fonction d'énergie.

### III.2 Algorithme *ICM*

L'algorithme *ICM* est un algorithme itératif qui permet de minimiser la fonction d'énergie à chaque pixel, connaissant la valeur observée du pixel et la classe courante de tous les autres pixels à l'intérieur de la fenêtre du voisinage.

L'ensemble des pixels de la fenêtre du voisinage est utilisé dans le calcul de la probabilité a priori basée sur la loi de Gibbs.

*ICM* s'avère important puisqu'il se base sur une méthode déterministe qui a la propriété de converger rapidement. Il est déterministe dans le sens où, l'examen d'un site "s" permet systématiquement d'affecter au site la classe pour laquelle l'énergie est minimale.

Le résultat de l'algorithme *ICM* dépend du choix de l'image d'initialisation. Par conséquent, un bon algorithme d'initialisation doit être utilisé. L'initialisation, ou la pré-classification, est une étape fondamentale dans la procédure de classification.

Avant l'application de l'algorithme *ICM* il faut segmenter l'image une segmentation initiale, pour cela nous choisissons le seuillage et nous faisons une comparaison le résultat de l'algorithme *ICM* avec l'algorithme de K-Means.

Pour segmenter une image par le seuillage on réalise les étapes suivantes :

- Déterminer le nombre de classe.

- Déterminer les seuils par l'histogramme tels que les seuils dépendent au nombre des classes.

### III.2.1 Exemple du seuillage

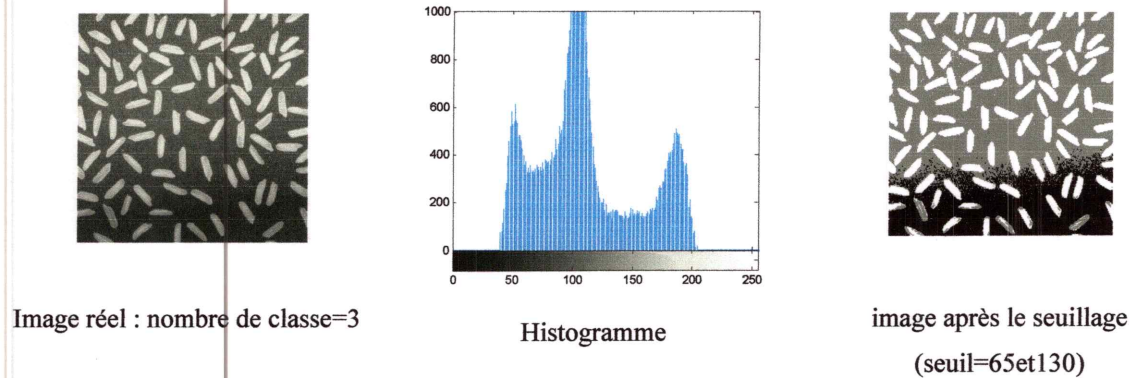


Figure III.1 Segmentation d'une image par le seuillage.

### III.2.2 Exemple sur l'algorithme de K-Means

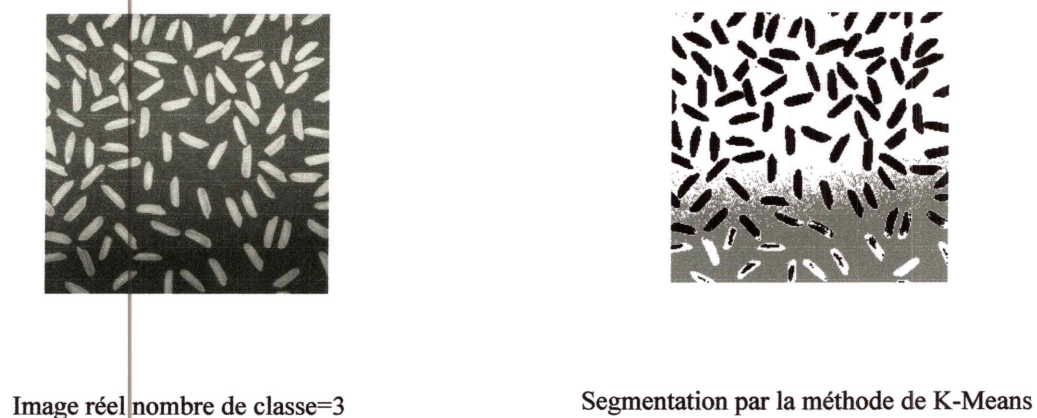


Figure III.2 Segmentation d'une image par K-Means.

Le déroulement de cet algorithme se présente comme suit :

❖ **Etape 1 segmentation initiale**

- Il s'agit de fournir à l'algorithme *ICM* une image classifiée et d'estimer les paramètres de la classification initiale.
- Les paramètres correspondent à la moyenne  $\mu$  et à la variance de chaque classe  $\sigma$ .

❖ **Etape 2 : calcul d'énergie**

- A partir d'une configuration initiale, les étapes suivantes sont réalisées pour chaque pixel.

a. Pour chaque classe :

- calcul de l'énergie de clique  $U_c$  tel que cette énergie c'est la somme de comparaison d'énergie de pixel avec les 8 pixels voisins.

$$U_c = \sum_{c \in C} v_c \quad (\text{III.1})$$

- $U_c = \beta$  si les deux pixels de la clique ont des valeurs différentes
- $U_c = -\beta$  si les deux pixels de la clique ont des même valeurs.

- Calcul la probabilité d'énergie  $E_p$  par la formule suivant :

$$P(Y_s = y_s / X_s = k) = pi(y_s/k) = (y_s/k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(y_s - \mu_k)^2}{2\sigma_k^2}} \quad (\text{III.2})$$

$$E_p(y/x) = -\ln P(Y = y/X = x) \quad (\text{III.3})$$

- Calcul l'énergie totale tel que  $E_t = U_c + E_p$ .

b. Recherche de la classe d'énergie minimale.

❖ **Etape 3 : affectation de la classe**

- On affecte l'étiquette de la classe, trouvée dans l'étape (b), au site courant.
- S'il y a eu des changements sur l'image classifiée et si le nombre de changements est supérieur à un seuil défini par l'utilisateur, on retourne à l'étape (2). Sinon, l'algorithme ICM s'arrête.

L'organigramme suivant résume les étapes de l'algorithme utilisé

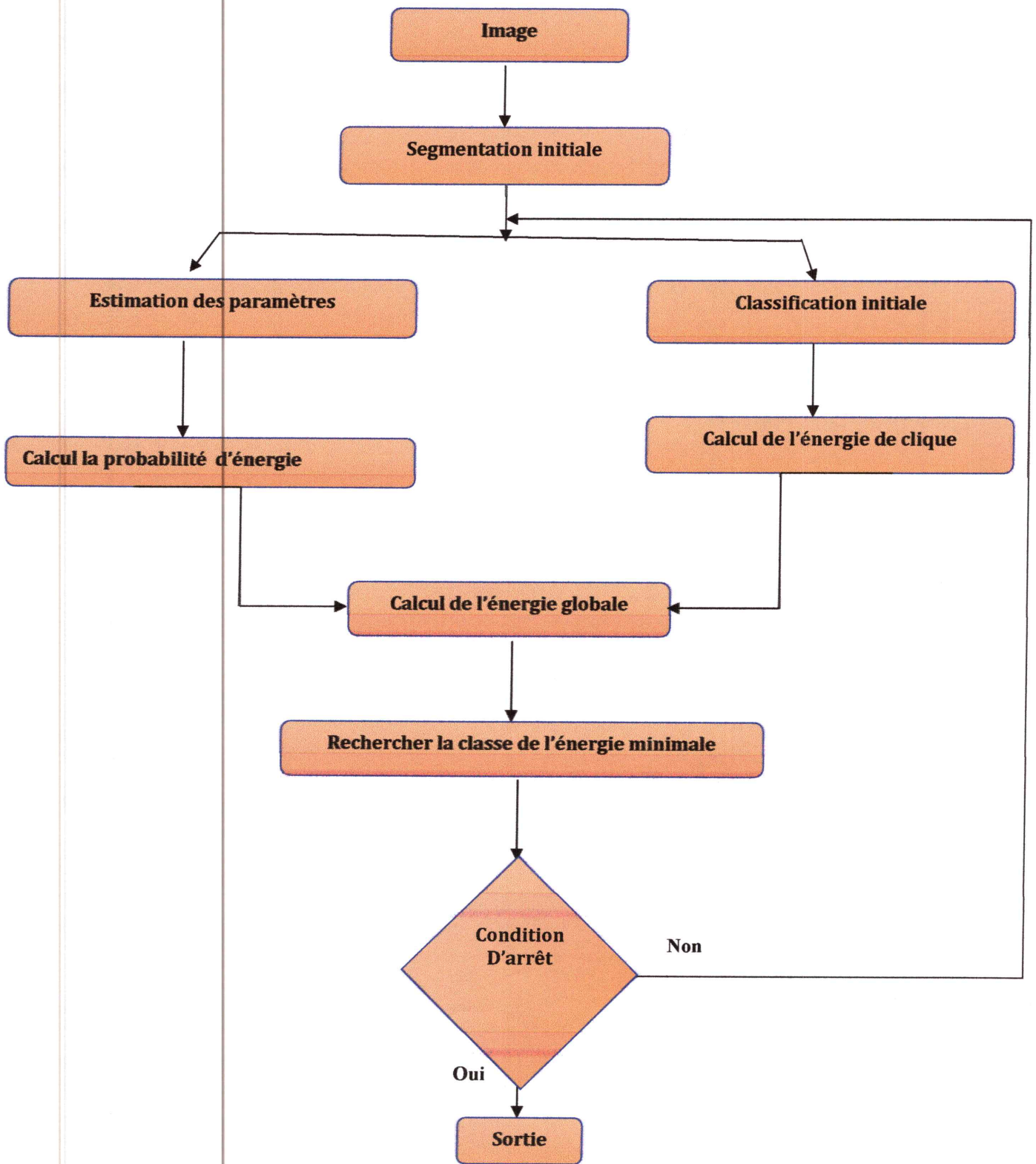


Figure III.3 Organigramme de la méthode *ICM* utilisée.

### III.3 Critères d'évaluation

Les critères d'évaluation quantitative peuvent être groupés en deux classes, selon que l'on possède ou non à une « vérité-terrain » qui constitue une segmentation de référence. Celle-ci est directement accessible dans le cas d'images de synthèse, mais elle doit être construite « à la main » par un expert du domaine de l'application dans le cas d'images réelles. Si l'on veut comparer de manière objective les méthodes, il est plus simple d'utiliser des images de synthèse, pour lesquelles une « vérité terrain » est parfaitement connue, à savoir la segmentation qui a servi à synthétiser l'image. L'inconvénient d'une telle démarche est que ces images ne représentent pas toutes les situations possibles d'une utilisation réelle.

#### III.3.1 Evaluation supervisée

Quand on dispose d'une vérité-terrain, l'évaluation des segmentations s'effectue à l'aide de critères comparant chaque segmentation avec l'image de référence. Dans ce cas, on parle alors d'évaluation supervisée.

#### III.3.2 Evaluation non supervisée

En l'absence de vérité-terrain, des critères quantitatifs absolus sont utilisés pour comparer les différents résultats de segmentation. L'évaluation est alors dite non supervisée. De nombreux critères ont été proposés pour quantifier la qualité de la segmentation de l'image, ces critères en deux catégories: les critères de contraste entre les régions et les critères d'uniformité à l'intérieur des régions.

### III.4 La mesure *PRI (Probabilistic Rand Index)*

L'indice de Rand a été introduit pour évaluer l'opération *clustering*. Son fonctionnement est basé sur la comparaison de la compatibilité des affectations entre les paires d'éléments dans les groupes. Pour la segmentation, l'indice de Rand entre une image de test *S* et une autre image vérité terrain *G (groundTruth)* est donné par la somme des nombres de paires de pixel qui ont la même classe dans *S* et *G* et ceux qui ont des classes différentes dans les deux segmentations, divisé par le nombre totales des paires de pixels.

Considérant l'ensemble des segmentations vérité terrain  $\{G_k\}$ , l'indice de Rand probabiliste  $PRI$  est défini par :

$$PRI(S, \{G_k\}) = \frac{1}{T} \sum_{i < j} [c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij})] \quad (III.4)$$

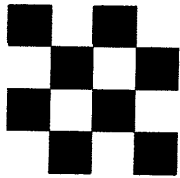
Où  $c_{ij}$  est l'événement que les pixels  $i$  et  $j$  ont la même classe et  $p_{ij}$  son probabilité.  $T$  est le nombre total des paires de pixels  $S$

### III.5 Résultats et discussion

Pour évaluer les performances de la méthode de segmentation par champs de Markov proposée et montrer son efficacité, des images de synthèse et des images réelles à 256 niveaux de gris ont été traitées sur un PC Intel Core2 Duo, CPU- 2.2 GHz avec 4 Go de RAM. Les algorithmes sont implémentés en Matlab 7.10.

#### III.5.1 Les images de synthèse

La taille d'image (256×256)



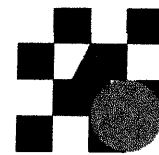
Image(1) : 2 classes

La taille d'image (128×256)



Image(2) : 3 classes

La taille d'image (128×128)



Image(3) : 4 classes

**Figure III.4 Images de synthèse.**

On a utilisé trois images synthétiques avec 2, 3, et 4 classes. Les images sont représentées sur la figure III.4. Nous avons ajouté aux images un bruit gaussien de moyenne nulle et de variance  $\delta$  qui trouvé dans le tableau suivant:

Images	Image(1)	Image(2)	Image(3)
$\delta$	3	15	0.3

**Tableau III.1** Les valeurs de la variance ( $\sigma$ ).

On mesure les performances à l'aide d'un programme en Matlab fournit par Allen Y. Yang [29]. Ce programme va comparer les images segmentées par notre méthode avec celles qui sont contenues dans la base de données de *Berkeley*.

Ce programme donne en sortie 3 indices de mesure qui permettent d'évaluer la différence entre l'image segmentée par notre méthode et l'image de la base de données de Berkeley. On va utiliser uniquement la mesure *PRI* (*Probabilistic Rand Index*) qui quantifie le nombre de pixels bien classés.

Nous avons testé l'algorithme *ICM* sur les différentes images en utilisant la méthode de seuillage comme segmentation initiale. Pour comparer la méthode *ICM*, nous avons choisi d'appliquer la méthode K-means pour les mêmes images. La mesure *PRI* est calculée pour les deux méthodes. Les résultats sont donnés sur les figures III.5, III.6, et III.7.

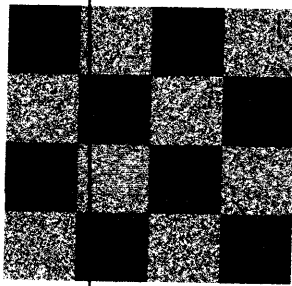
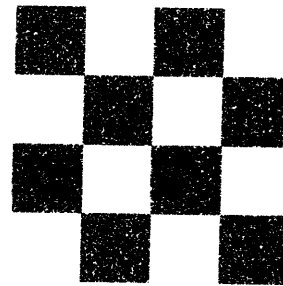
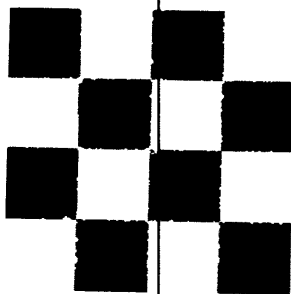


Image bruitée



Segmentation par le seuillage (seuil=1)



Segmentation par la méthode d'*ICM*  
avec nombre d'itération=15  
( $\beta = 0.5$ )

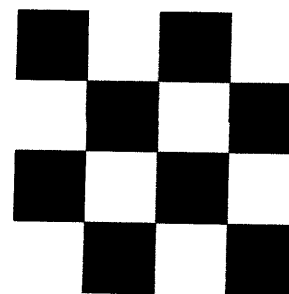


Image originale (1)

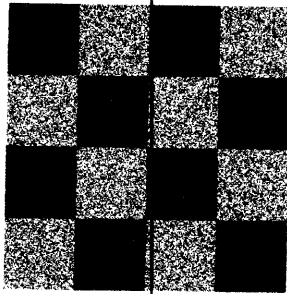
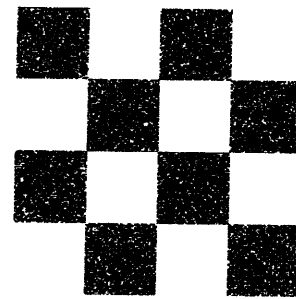


Image bruitée



Segmentation par la méthode de  
K-Means

Figure III.5 Illustration des résultats obtenus sur l'image (1) par les différents algorithmes.

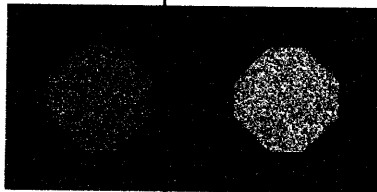
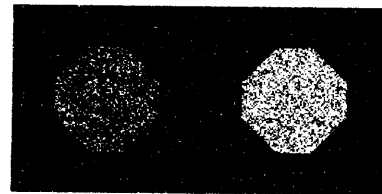


Image bruitée



Segmentation par le seuillage (seuils=12et19)



Segmentation par la méthode d'ICM  
avec nombre d'itération=30  
( $\beta = 1.5$ )



Image originale (2)



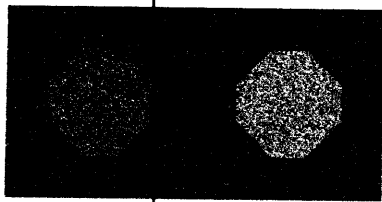
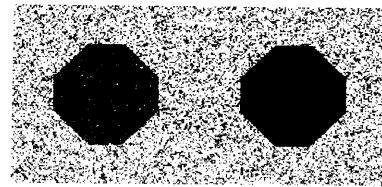


Image bruitée



Segmentation par la méthode de K-Means

Figure III.6 Illustration des résultats obtenus sur l'image (2) par les différents algorithmes.

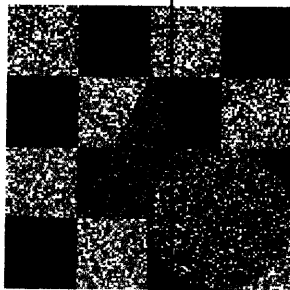
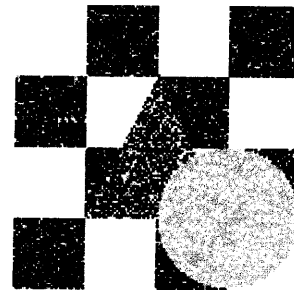
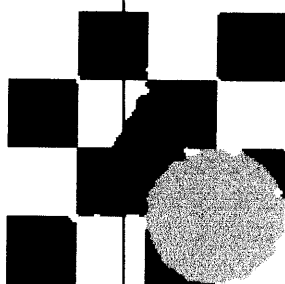


Image bruitée



Segmentation par le seuillage

(seuils=0.2, 0.6 et 1)



Segmentation par la méthode d'ICM

avec nombre d'itération =30

( $\beta = 1$ )

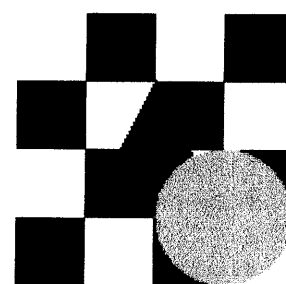


Image originale(3)

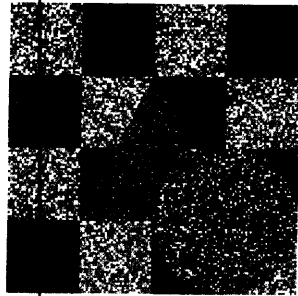
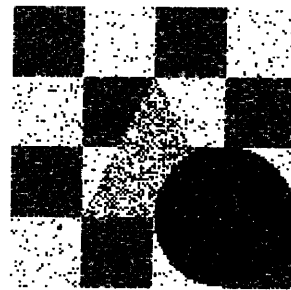


Image bruitée



Segmentation par la méthode de  
K-Means

Figure III.7 Illustration des résultats obtenus sur l'image (3) par les différents algorithmes.

images	Image(1)	Image(2)	Image(3)
<i>PRI</i> (k-means)	0.7136	0.6907	0.8349
<i>PRI</i> (ICM)	0.9878	0.9938	0.9806

Tableaux III.2 Les résultats *PRI* obtenus.

- À partir des résultats représentés sur les figures précédentes et les résultats du paramètre *PRI* dans le tableau III.2, nous pouvons constater que les performances de l'algorithme *ICM* sont toujours meilleures à celle du K-Means.
- Pour étudier l'influence du bruit sur la segmentation, nous avons effectué une série de tests en variant la variance du bruit blanc  $\sigma$ . Les résultats sont représentés sur le tableau III.3. On peut voir la dégradation du *PRI* lorsqu'on augmente le  $\sigma$ . Cela est plus remarquable lorsque le nombre de classe augmente comme il est le cas pour l'image 3.
- Le paramètre  $\beta$  est un paramètre important qui en changeant sa valeur on peut avoir des résultats différents de segmentation (tableau III.4). Le rôle de ce paramètre est de favoriser l'homogénéité spatiale entre les pixels voisins.

		Image(1)			Image(2)			Image(3)		
$\delta$	1	3	6	1	15	20	0.1	0.3	0.6	
<i>PRI</i>	0.9966	0.9878	0.9731	1	0.9938	0.9849	0.9937	0.9806	0.9212	

Tableaux III.3 Influence du bruit.

	Image(1)			Image(2)			Image(3)		
$\beta$	0.5	4	10	1.5	5	10	1	5	10
<i>PRI</i>	0.9878	0.9632	0.9670	0.9938	0.9886	0.9883	0.9806	0.9530	0.9673

Tableaux III.4 Effet de paramètre  $\beta$  avec  $\delta = 3$  pour l'image (1),  $\delta = 15$  pour l'image (2) et  $\delta = 0.3$  pour l'image (3).

### III.5.2 Les images réelles

Pour mesurer les performances de notre méthode, on va utiliser des exemples d'images réelles contenues dans la base de données de Berkeley (figure III.8). Cette base de données rassemble 12000 segmentations faites par 30 individus. À cause de la complexité des images réelles, nous avons choisi d'utiliser la méthode K-Means comme segmentation initiale. Les résultats obtenus sont présentés sur les figures III.9, III.10, III.11.



Image(1)



Image(2)



Image(3)

Figure III.8 Images réelles utilisées.

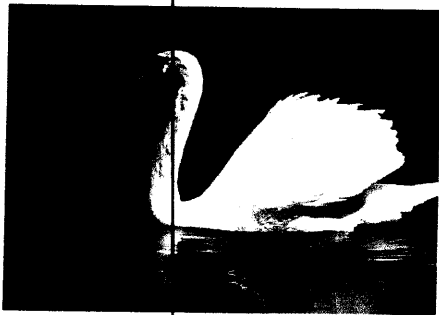
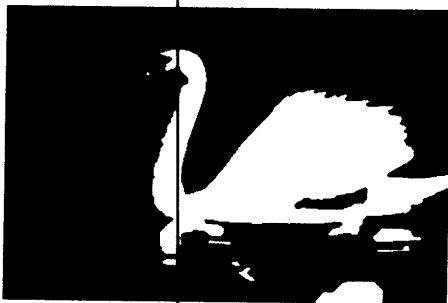


Image (1)



Segmentation par la méthode de K-Means



Segmentation par la méthode d'ICM avec  
nombre d'itération=15 ( $\beta = 6$ )



Segmentation par vérité terrain

**Figure III.9** Illustration des résultats obtenus sur l'image (1) par les deux algorithmes *ICM* et *K-Means*.

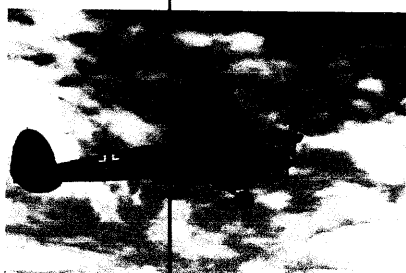
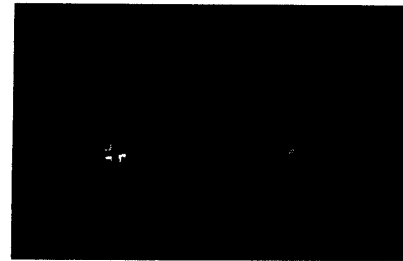


Image (2)



Segmentation par la méthode de K-Means



Segmentation par la méthode d'ICM avec  
nombre d'itération= 15 ( $\beta = 1$ )

Segmentation par vérité terrain

**Figure III.10** Illustration des résultats obtenus sur l'image (2) par les deux algorithmes *ICM* et *K-Means*.

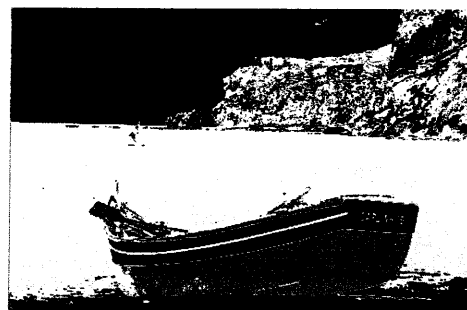
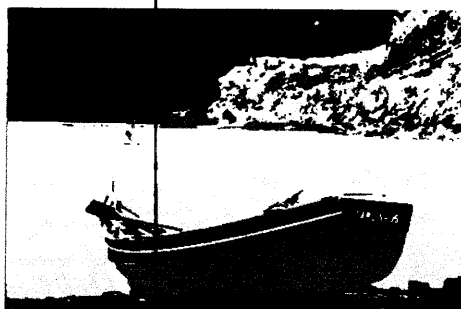


Image (3)

Segmentation par la méthode de K-Means



Segmentation par la méthode d'ICM

Segmentation par vérité terrain

**Figure III.11** Illustration des résultats obtenus sur l'image (3) par les deux algorithmes *ICM* et *K-Means*.

image	Image (1)	Image (2)	Image (3)
PRI	0.9059	0.9748	0.8241

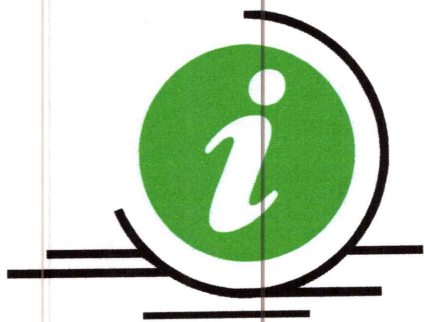
**Tableaux III.5** Les résultats *PRI* obtenus.

D'après les résultats obtenus, nous pouvons tirer les résultats suivants :

- Le résultat de l'algorithme *ICM* dépend du choix de l'image d'initialisation. Par conséquent, un bon algorithme d'initialisation doit être utilisé.
- Le paramètre  $\beta$  correspond au facteur de régularisation de l'homogénéité, L'augmentation du paramètre  $\beta$  correspond à une augmentation du lissage. Un choix correct de sa valeur, peut améliorer le résultat de la segmentation.

### III.6 Conclusion

L'algorithme *ICM* est une méthode itérative permettant l'estimation des caractéristiques de chaque classe et la segmentation des images. Pour tester la performance de l'algorithme *ICM* des images synthétiques et des images réelles ont été utilisées. Une étude comparative entre l'algorithme *ICM* et autre algorithmes K-Means a été effectuée. La performance de la classification est calculée en terme *PRI*. L'algorithme *ICM* produit une meilleure segmentation de l'image.



***Conclusion***

---

***Générale***

## Conclusion Générale

Dans ce mémoire, nous avons abordé le problème de la segmentation par champ Markov aléatoire en utilisant une stratégie Bayésienne. Les traitements statistiques d'images fondés sur des modèles Markoviens peuvent présenter des qualités exceptionnelles.

La segmentation markovienne est une méthode qui exploite la notion de voisinage et la notion de dépendance locale des pixels voisins. Elle est basée sur les champs aléatoires de Markov et sur la distribution de Gibbs.

Le modèle des champs aléatoires de Markov et la distribution de Gibbs exploitent la dépendance entre les pixels voisins dans les images. Le processus de classification de cet algorithme consiste à maximiser la probabilité a posteriori MAP en se basant sur les données observées et sur l'information du voisinage de chaque pixel segmenté. Cette information du voisinage est utilisée pour calculer la probabilité a priori basé sur un modèle simple de fonction d'énergie.

Pour optimiser la segmentation au sens de MAP, nous avons choisi d'étudier et d'implémenter un algorithme itératif, c'est la méthode ICM.

Nous avons pu examiner les performances de ces techniques dans le troisième chapitre sur des images synthétiques et des images réelles. L'évaluation de la segmentation a été effectuée d'une manière subjective en jugeant la qualité visuelle de la segmentation et d'une manière objective en calculant le paramètre PRI qui quantifie le pourcentage des pixels bien classés.

L'étude d'évaluation de la segmentation markovienne par rapport à la méthode K-Means a montré que l'algorithme ICM est plus précis malgré la présence du bruit.

Nous avons également montré que le paramètre  $\beta$  qui correspond au facteur de régularisation de l'homogénéité est déterminé par étude expérimentale et l'augmentation du paramètre  $\beta$  correspond à une augmentation du lissage.

Pour éviter de choisir ce paramètre, on propose de l'estimer à partir des images parce qu'il est lié à la nature de chaque image (présence de texture ou des zones homogènes ...).

Nous proposons aussi d'utiliser les techniques de segmentation par MRF avec d'autres méthodes d'optimisation comme les recuits simulés.



- [1] O. Abdelli. *Segmentation d'images par seuillage d'histogrammes bidimensionnels*. Mémoire de Magister, en automatique, Université mouloud Mammeri de Tizi-Ouzou, (2011).
- [2] J. dubois. *Segmentation par approche contours*. Université de Rouen(1973).
- [3] M. Merzougui, M. Nasri. *Contrôle de qualité par segmentation évolutionniste d'images*(1976).
- [4] K. larbi. *Segmentation d'image basée sur la modélisation statique d'histogrammes*. Mémoire de Magistère, en automatique, Université Mouloud Mammeri, de Tizi-Ouzou,(2012).
- [5] S. Zuker. *Region growing childhood and adolescence*. Computer Graphics and imae dans processing 5, pp. 382-399, (1976).
- [6] F. Jian. *Analyse d'images par champ de Markov*. Mémoire de Magister, Université de Rennes 1(1974).
- [7] Ludovic Macaire, Sylvie philipp-Foliguet .*Segmentation des images couleur*(1998).
- [8] Alexandre Hamon. *Segmentation couleur de plans cadastraux De la ville de paris au XIXème siècle* (2000).
- [9] S. L. Horowitz and S. Pavlidis. *Picture segmentation by a directed split and merge procedure*. In 2<sup>nd</sup> Int. Joint Conf. On pattern Recognition, pp. 424-433, (1974).
- [10] S. nebti. *Optimisation par écosystèmes artificiels : application à la segmentation d'image*. Mémoire de Magister, en informatique, Université Mentouri, de Constantine(1999).
- [11] A. Boucher, c. Garbay, *des agents spécialisés pour la compréhension de séquences d'images*. RFIA. 98, vol. II, pp. 275-284,(1998).
- [12] M. Fontaine. *Segmentation non supervisée d'images couleur par analyse de la connexité des pixels*. Thèse de doctorat, Université de Lille 1. (2001).
- [13] A. Nakib. *Conception de méta heuristiques d'optimisation pour la segmentation d'images. Application à des images biomédicales*. Thèse de doctorat, Université de paris 12-Val de Marne, (2007).
- [14] A. K. Jain, M. N. Murty and p. J. Flynn. . *Data clustering a review*. ACM computing survey, vol. 31, n°3, pp-264, 322, (1999).
- [15] N. Monmarché. *Algorithmes des fourmis artificielles : applications à la classification et à l'optimisation*. Thèse de doctorat, Université de Tours, (2000).
- [16] R. Horaud et O. Monga. *Vision par ordinateur : outils fondamentaux*. Editions Hermès, (1993).
- [17] A. S. Abutaleb. *Automatic thresholding of grey-level pictures using two-dimensional entropy*. Comput. Using Graphics Image Process. 47 22-32, (1989).

- [18] J. Gong, L. Li and W. Chen. *Fast recursive algorithms for two-dimensional thresholding*. Pattern Recognition, vol. 31, no. pp. 295-300, (1998).
- [19] J. Bernsen. *Dynamic thresholding of grey-level images proc. Internat. Conf. pattern Recognition*. Pp. 1251-1255, (1986).
- [ 20] W. Pieczynski. *Segmentation statistique d'images*. Notes de cours, Institut National des Télécommunications, France, 26p ,(1992).
- [ 21] H. Derin. and H. Elliott .*Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 1, pp. 39-55 ,(1987).
- [ 22] E. Rignot and R. Chhappa .*Segmentation of Synthetic Aperture Radar Complex Data*. J. Opt Soc. Amer., A, pp. 1499-1509,(1991).
- [23] S. Z. Li. *Markov Random Field Modelling in Computer Vision*, Springer, (1995).
- [24] J .Besag. *Spatial Interaction and the Statistical Analysis of Lattice Systems*. J.Royal Statist. Soc, Serie B, vol 36, pp. 192-236,(1974).
- [25 ] S .Geman, and D. Geman. *Stochastic Relaxation, Gibbs Distributions, and The Bayesian Restoration of Images. IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI4, no. 6, pp. 721-741, (1984)
- [ 26] Pappas, T. *An Adaptive de Clustering for Image Segmentation. IEEE Transactions on Signal Processing*, vol. 40, no. 4, pp. 901-914, (1992).
- [27 ] Metropolis et al. Metropolis, N, Rosenbluth, A. W, Rosenbluth, N. M., Teller, A. H., and Teller, E. (1953). *Equation of state calculations by fast computing machines*. J. Chemical Physics, 21, 1087-1091,(1953)
- [28] Besag, J. On the Statistical Analysis of Diay Pictures. Journal of Royal Statistic Soaety, vol. 48, no. 3, pp. 259-302,(1986)
- [29]