

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERY, Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Automatique

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

D'Ingénieur d'Etat en Automatique

Thème

**Segmentation par classification
d'une partie de l'image**

Proposé et dirigé par :
Mme HAMMOUCHE Sadia

Présenté par
Melle MEDROUK hayet
Melle TOUCHERIFT souad

Soutenu le : / 07 /2010

Promotion 2010



REMERCIEMENT

Remerciements

Louage à DIEU qui nous a comblé de faveurs, et apporté son aide afin de réaliser le travail exposé tout le long de cette thèse.

Nous tenons à exprimer toute notre reconnaissance et notre gratitude à Mme HAMOUCHE.

Pour avoir accepté de nous encadrer, leurs qualités humaines et pour tout ce que nous avons appris en travaillant avec elle.

Nos vifs remerciements vont aussi aux membres de jury qui ont accepté aimablement d'examiner et de juger notre travail.

Nous tenons également à remercier tous ceux qui ont contribué à la réalisation de ce travail.



SOMMAIRE

Sommaire

Introduction générale.....	1
 Chapitre 1 : généralités sur le traitement d'image	
I.1 Introduction.....	3
I.2 Notions d'image.....	3
I.3 Différents types d'une image numérique.....	4
I.3.1 Image binaire.....	4
I.3.2 Image en niveau de gris.....	4
I.3.3 Image couleur.....	5
I.4 Architecture générale d'un système de traitement d'image.....	6
I.4.1 Acquisition.....	7
I.4.2 Traitement et pos- traitement.....	7
I.4.3 Le traitement numérique.....	7
I.4.4 Visualisation, transmission et stockage.....	7
I.5 Segmentation.....	8
I.6 Approches de segmentation.....	8
I.6.1. Approches contour.....	9
I.6.1.1 Opérateur gradient.....	9
I.6.1.2 Opérateur de Sobel.....	10
I.6.1.3 Opérateur Prewitt.....	10
I.6.1.4 Opérateur Laplacien	11
I.6.2 Approches région.....	11
I.6.2.1 Segmentation par division de région.....	11
I.6.2.2 Segmentation par croissance de région.....	12
I.6.2.3 Segmentation par division-fusion de région.....	13
I.6.3 Classification.....	13
I.7 Définition de la texture.....	13
I.8 Analyse de texture.....	14
I.8.1 Méthodes statistiques.....	14
I.8.1.Méthodes structurelles.....	14
I.9 Les attributs statistiques de texture.....	15

I.9.1	Les attributs statistiques du premier ordre.....	15
I.9.1.1	La moyenne	16
I.9.1.2	La variance	16
I.9.1.3	Le moment d'ordre 3	17
I.9.1.4	Le moment d'ordre 4.....	18
I.9.2	Les attributs statistiques de deuxième ordre.....	19
I.9.3	Les attributs statistiques d'ordre supérieur.....	22
I.10	Conclusion.....	24
 Chapitre 2: Méthodes de classification		
II.1	Introduction.....	25
II.2	Définition de la classification.....	25
II.3	Les techniques de classification.....	26
II.3.1	La classification supervisée.....	26
II.3.2	La classification non supervisée	27
II.4	Quelques méthodes de classification.....	27
II.4.1	Méthode de K plus proche voisins.....	28
II.4.2	Approche densité des informations.....	29
II.4.3	Approche hiérarchique.....	30
II.4.4	La méthode de k means.....	31
II.4.5	La classification floue	34
II.5	Conclusion.....	35
 Chapitre 3 : Tests et résultats		
III.1	Introduction.....	36
III.2	Principe de la méthode proposée.....	36
III.3	Influence du taux de pixels sélectionnés T	41
III.4	Influence de la taille de la fenêtre d'analyse $W \times W$	45
III.5	Influence de la direction θ utilisée dans le calcul des paramètres de texture.....	49
III.6	Influence de la taille de la fenêtre utilisée pour calculer la matrice de cooccurrence $V \times V$.	53
III.7	Conclusion.....	57
	Conclusion générale.....	58



INTRODUCTION
GENERALE

Avec la parole, l'image constitue l'un des moyens les plus importants qu'utilisent les êtres humains pour communiquer. C'est un moyen qui leur permet de tout âge et de toutes cultures de se comprendre en analysant l'image chacun à sa manière, pour en dégager une impression et en extraire les informations souhaités.

Les images sont largement utilisés dans de nombreux domaines tel que la médecine, l'industrie, la météorologie, l'agriculture, etc.

Avec le développement technologique et l'apparition de l'image numérique, la discipline traitement d'image à vu le jour. Elle a donc des objectifs variés, dont le but est à la fois simple dans son concept et difficile dans sa réalisation. Simple en effet, puisqu'il s'agit de reconnaître des objets que notre système visuel aperçoit rapidement, difficile cependant car dans la grande quantité d'informations contenue dans l'image, il faut extraire des éléments pertinents pour l'application visée et ceci indépendamment de la qualité de l'image. Le traitement d'image est alors doté d'outils et de méthodes puissants tels que les mathématiques et l'informatique.

Parmi les opérations de traitement d'images, la segmentation est l'une des plus utilisée. Elle consiste à séparer les objets entre eux et du fond de l'image.

Notre travail s'inscrit dans ce domaine, nous proposons de segmenter des images texturées en utilisant des méthodes de classification. La classification consistera alors à organiser les pixels de l'image en un ensemble fini de classes. Elle peut être réalisée sur les niveaux de gris des pixels ou sur d'autres paramètres tels que les attributs de texture extraits d'un voisinage de ces pixels.

L'extraction des attributs pour tous les pixels de l'image en plus de la technique de classification utilisée sur toutes ces données peut prendre un temps de calcul important.

Nous proposons, dans ce travail, de réduire ce temps de calcul en réduisant le nombre de pixel à traiter tout en essayant d'avoir un résultat identique, sinon proche de celui qu'on obtiendrait si on utilise tous les pixels de l'image.

Pour ce faire, nous sélectionnons de manière aléatoire un certain nombre de pixels de l'image. Nous extrayons ensuite les paramètres de texture de ces pixels, nous les classons par la suite par la méthode des k-means. Au final, les pixels qui n'ont pas été sélectionnés, donc qui ne sont pas encore classés, nous leur attribueront la classe majoritaire des pixels déjà classés se trouvant dans leur voisinage.

Notre mémoire est organisé en trois chapitres.

Le premier chapitre est consacré aux notions de base concernant l'image, les différentes méthodes de segmentation, les principales méthodes d'analyse de textures et la définition des attributs statistiques de texture.

Dans le second chapitre nous expliquons les principales méthodes de classification et nous nous attardons sur la méthode de classification non supervisée que nous avons utilisée à savoir la méthode des k-means.

Dans le dernier chapitre nous expliquons la méthode proposée avant de l'appliquer sur des images texturées synthétiques et réelles.

Nous terminons par une conclusion générale.



CHAPITRE I

*Notions de base de
traitement d'image*

I.1 Introduction

Dans ce chapitre nous définirons l'image et rappellerons les notions de base du traitement d'image. Nous présentons les principales méthodes de segmentation d'image puis nous terminons par développer la notion de texture et donner quelques méthodes d'extraction des attributs de texture.

I.2 Notions d'image

Une image est une représentation bidimensionnelle d'objets tridimensionnelles de nature diverses. Elle peut être considérée comme un signal continu bidimensionnel ou présentée sous forme d'un ensemble de points répartis dans une surface donnée. Elle peut être définie comme étant une matrice rectangulaire, où chaque élément appelé pixel extraction de l'expression anglaise « Picture élément », il a comme caractéristique un niveau de gris ou de couleur prélevé à l'emplacement correspondant dans l'image réelle ou calculé à partir d'une description interne de la scène à représenter.

Les principales caractéristiques d'une image numérique sont [5] :

- **La dimension** qui définit la taille de l'image et qui correspond au nombre total de pixels de l'image.
- **La résolution** qui correspond à la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production de l'image, elle est exprimée en nombre de pixels par unité de mesure.
- **La luminance** qui est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface.
- **Le contraste** qui est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images. Si L_1 et L_2 sont les degrés de luminosité respectivement de deux zones voisines d'une image, le contraste C est défini par le rapport :

$$C = \frac{L_1 + L_2}{L_1 - L_2} \quad (\text{I.1})$$

- **Les contours** qui représentent les frontières entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative.

- **Le voisinage** qui est défini comme étant l'ensemble des pixels qui entourent dans toutes les directions un pixel donné.
- **L'histogramme** qui est un graphique statistique permettant de représenter la distribution des niveaux de gris des pixels d'une image.

I.3 Différents types d'une image numérique [6]

Principalement il existe trois types d'images numériques qui sont des images binaires, en niveau de gris et en couleurs.

I.3.1 Image binaire

Une image binaire est une matrice rectangulaire dont le nombre de niveaux de gris est réduit à deux éléments 1 et 0, où le niveau 0 représente le noir et le niveau 1 représente le blanc, donc on utilise qu'un seul bit pour coder un niveau.

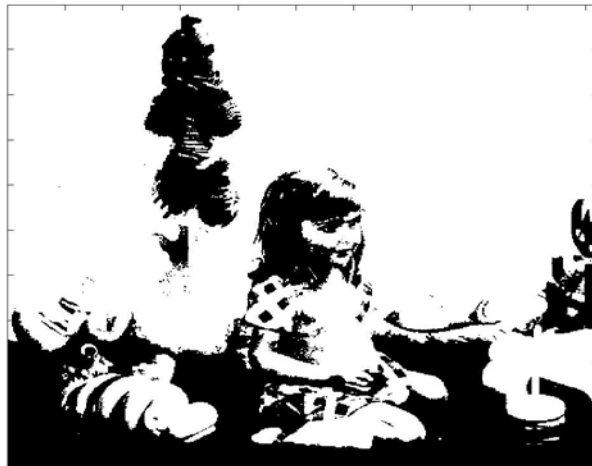


Figure I.1 : Exemple d'image binaire.

I.3.2 Image en niveau de gris

Le niveau de gris est la valeur de l'intensité lumineuse en un point. Le pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter des images en niveau de gris, on peut attribuer à chaque pixel de l'image une valeur correspondante à la quantité de la lumière renvoyée. Cette valeur est souvent comprise entre 0 et 255, chaque pixel est donc codé par 8 bits (un octet).



Figure I.2 : Exemple d'image en niveau de gris.

I.3.3 Image couleur

L'image couleur est obtenue par une combinaison de trois couleurs de base généralement (RVB), elle consiste à utiliser 24 bits pour représenter chaque point de l'image, 8 bits sont employés pour décrire la composante rouge (R), 8 pour décrire la composante verte (V) et 8 pour décrire la composante bleue (B).

Il est ainsi possible de représenter environ 16,7 millions de couleurs différentes simultanément. Ce qui est pratiquement impossible.



Figure I.3 : Exemple d'image couleur sur 24 bits.

Afin de diminuer la charge de travail, qui consiste à manipuler des images en 24 bits, on peut utiliser le mode de représentation en couleur indexée. Le principe consiste à déterminer

le nombre de différentes couleurs utilisées dans l'image, puis à créer une table de ces couleurs en attribuant à chacune une valeur numérique qui correspond à sa position dans la table, appelée palette.

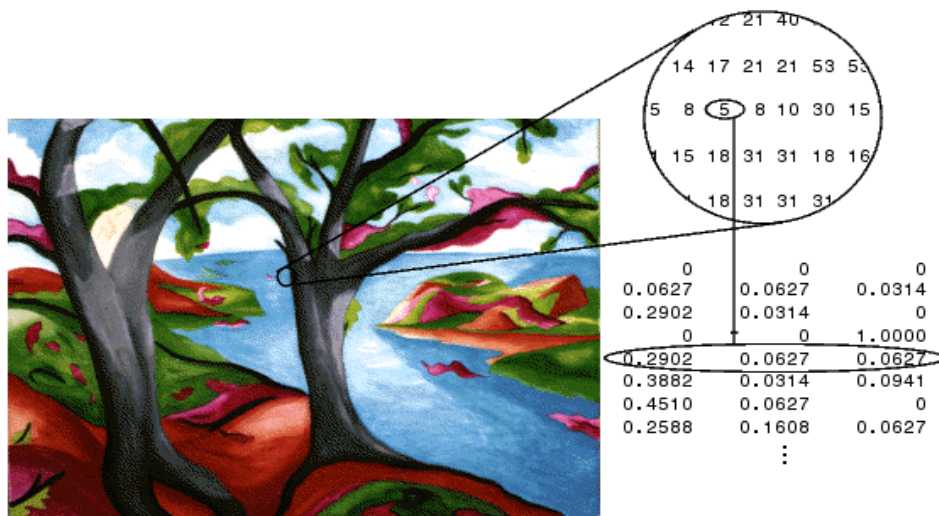


Figure I.4 : Exemple d'image indexée.

I.4 Architecture générale d'un système de traitement d'images

Le traitement d'images désigne une discipline de l'informatique et de mathématiques appliqués, qui étudie les images numériques et leurs transformations dans le but d'améliorer leur qualité ou d'extraire des informations pertinentes.

Un système de traitement d'image est composé de plusieurs parties, la figure I.5 montre les différentes étapes de traitement :

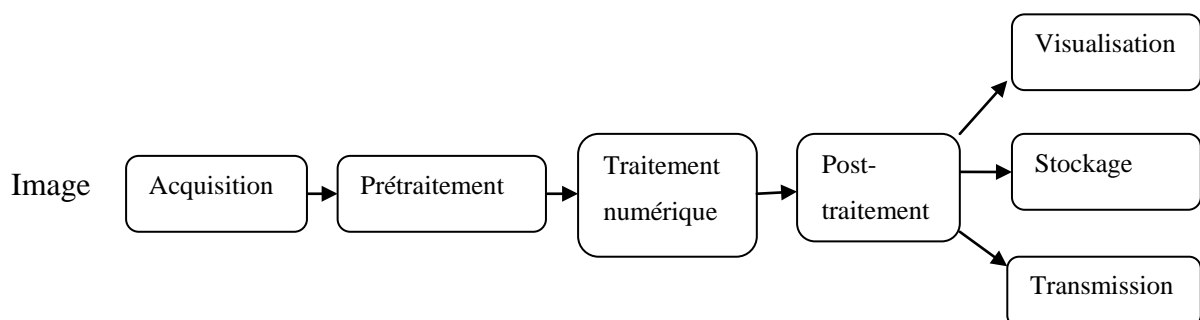


Figure I.5 : Schéma synoptique d'un système de traitement d'image

I.4.1 Acquisition

L'acquisition d'une image est l'opération qui permet le passage de l'information réelle à une représentation numérique. Elle est réalisée en 3 étapes :

La transformation du signal optique en un signal analogique (électrique), l'échantillonnage qui consiste à multiplier le signal analogique par une série d'impulsion unité dans le but d'avoir des échantillons du signal, et la quantification qui est une traduction des échantillons en valeurs numériques selon une règle de codage choisi.

I.4.2 Prétraitement et post-traitement

Le prétraitement concerne les images acquises tandis que le post-traitement concerne les images traitées. Dans les deux étapes, les images ne peuvent échapper aux effets de dégradation due essentiellement aux phénomènes physiques.

L'opération de prétraitement a pour but principal la correction de l'imperfection du système d'acquisition, alors que l'opération post-traitement vise surtout à maitre en forme pour éventuellement une visualisation, transmission ou un stockage.

Les techniques de prétraitement les plus utilisées reposent sur le filtrage numérique et la modification de l'histogramme.

I.4.3 Le traitement numérique

Le traitement numérique d'image est l'ensemble des méthodes qui permettent de décrire quantitativement le contenu d'une image.

On distingue deux types de traitement, les traitements de bas niveau et les traitements de haut niveau. Cette distinction est liée au contenu sémantique des entités traitées et extraites de l'image.

Les traitements de bas niveau opèrent en général sur les grandeurs calculées à partir des valeurs attachées à chaque point de l'image sans faire la liaison avec la réalité. Ils opèrent plutôt sur des données de nature numérique.

Les traitements de haut niveau s'appliquent à des entiers de nature symbolique associés à une représentation de la réalité extraite de l'image ils sont relatifs à l'interprétation.

I.4.4 Visualisation, transmission et stockage

La visualisation est une opération qui permet de transformer le signal numérique qui est la matrice image en un signal analogique visible par l'œil de l'observateur à travers un dispositif d'affichage.

L'image peut être transmise vers une station d'archivage ou de traitement qui peut être différente de la station d'acquisition.

Les images sont archivées sur des supports destinés à cet effet comme par exemple les bandes magnétiques, les disques durs,....

I.5 Segmentation [8]

La segmentation d'image est une opération de traitement d'image qui a pour but de séparer des objets du fond d'une image numérique et entre eux, ou d'une manière plus générale de rassembler les différents pixels ayant des propriétés communes. Si la segmentation, sépare deux classes de pixels, elle est appelée binarisation.

L'homme sait naturellement séparer des objets dans une image, il se base pour cela des connaissances de haut niveau qui lui permettent de détecter ce qui l'intéresse dans l'image.

En traitement d'image la segmentation est le partitionnement d'une image I en N régions R_i disjointes et homogènes selon un prédicat, ou critère P déterminé (couleur, intensité texture, indice,....).

Formellement, l'ensemble des régions $\{R_1, R_2, \dots, R_N\}$ est une segmentation de l'image I si :

$$-\bigcup_{i=1}^N R_i = I, \quad i = 1, \dots, N. \quad (\text{I.2})$$

$$-R_i \cap R_j = \emptyset \quad \forall i \neq j \quad (\text{I.3})$$

$$-R_i \text{ est connexe} \quad \forall i \in \{1, \dots, N\} \quad (\text{I.4})$$

$$-P(R_i) = \text{vrai} \quad \forall i \in \{1, \dots, N\} \quad (\text{I.5})$$

$$-P(R_i \cup R_j) = \text{faux} \quad \forall i \neq j \text{ et } R_i \text{ adjacence à } R_j \quad (\text{I.6})$$

Plusieurs algorithmes de segmentation existent selon les diversités des propriétés recherchés, les problèmes rencontrés et la manière d'opérer le regroupement des pixels.

I.6 Approches de segmentation

Plusieurs approches de la segmentation des images existent et nous citons :

- Approches frontière (contour).
- Approches région.
- Classification des pixels.

I.6.1 Approches contour [8]

Pour une image en niveau de gris, nous définissons un contour comme une frontière entre deux régions de niveaux de gris différents et relativement homogènes.

L'approche contour consiste donc à identifier les transitions entre régions.

Un contour est alors déterminé par une variation rapide des niveaux de gris des pixels consécutifs. Cette variation peut être détectée par le maximum d'une dérivée première de la fonction niveau de gris d'un point ou par le passage par zéro de la seconde dérivée (voir figure I.6)

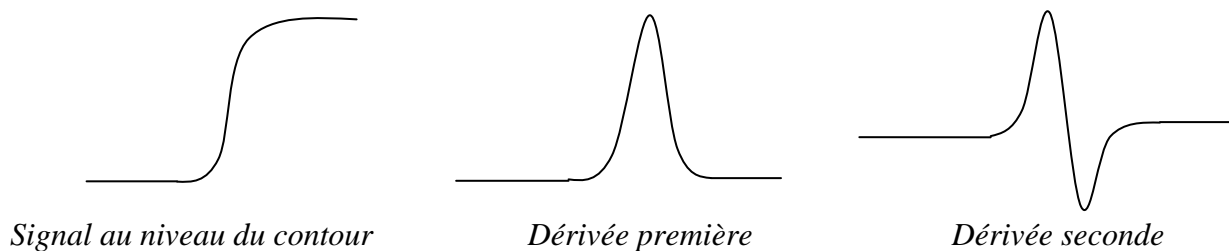


Figure I.6 : Méthode de détection de variations de niveaux de gris

Les opérations les plus utilisées pour détecter ces variations sont les opérateurs gradients, de Sobel, le Prewitt et Laplacien.

I.6.1.1 Opérateur gradient

Le gradient en un point est défini par :

$$\nabla I(x, y) = \left[\frac{dI}{dx} \vec{u} + \frac{dI}{dy} \vec{v} \right] = G_x \vec{u} + G_y \vec{v} \quad (I.7)$$

Sous forme numérique, ce vecteur sera :

$$\nabla I(x, y) = [I(x, y) - I(x-1, y)] \vec{u} + [I(x, y) - I(x, y-1)] \vec{v} \quad (I.8)$$

Avec : \vec{u} et \vec{v} deux vecteurs unitaires sur x et y.

$I(x, y)$: Fonction de niveau de gris de l'image.

Le module du gradient est :

$$|\nabla I(x, y)| = \sqrt{\left(\frac{dI}{dx}\right)^2 + \left(\frac{dI}{dy}\right)^2} \quad (\text{I.9})$$

Son orientation est :

$$\alpha = \text{arctg} \frac{dI/dx}{dI/dy} \quad (\text{I.10})$$

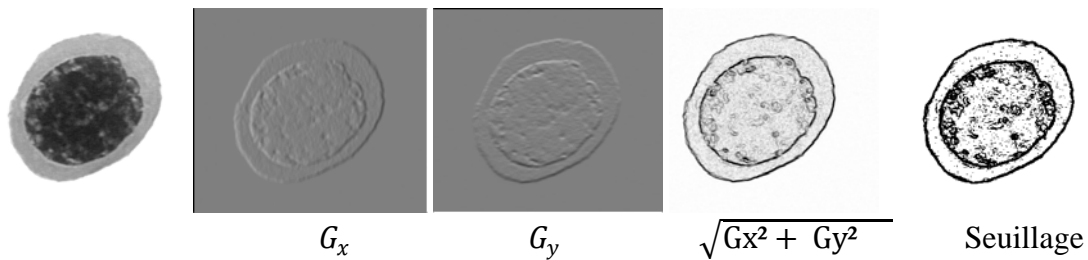


Figure I.7 : Extraction des contours à partir du gradient.

I.6.1.2 Opérateur de Sobel

Cet opérateur combine les opérateurs de lissage et les opérateurs différentiels. Il est alors l'une des solutions du problème du bruit, il utilise les deux masques :

$$G_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad G_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

I.6.1.3 Opérateur de Prewitt

Prewitt approxime le gradient par la convolution de l'image avec les masques de convolution suivants :

$$G_x = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \quad G_y = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}$$

I.6.1.4 Opérateur Laplacien

C'est un opérateur différentiel isotrope définie par :

$$\nabla^2 I = \frac{d^2 I}{dx^2} + \frac{d^2 I}{dy^2} \quad (\text{I.11})$$

Une approximation discrète de celui-ci est donnée par la fonction $L(x,y)$ définie par :

$$L(x,y) = I(x,y) - \frac{1}{4}[I(x,y+1) + I(x,y-1) + I(x+1,y) + I(x-1,y)] \quad (\text{I.12})$$

Dans ce cas, on cherche le passage par zéro de cette fonction. La façon la plus simple est de prendre les points qui vérifient $L(x,y)=0$. Cet opérateur n'est pas utilisé car le résultat est très sensible aux bruits.

On approxime le Laplacien par :

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

I.6.2 Approches région [5]

Une région est un ensemble connexe de pixels voisins de l'image ayant des propriétés communes (intensité, texture,...) qui les différencient des pixels des régions voisines.

Les principales techniques utilisées dans la segmentation des images par approche région sont :

- la segmentation par division de région.
- la segmentation par croissance de région.
- la segmentation par division-fusion de région.

I.6.2.1 Segmentation par division de région

Les méthodes de division sont aussi appelées méthodes *quadtree*.

On part d'une région initiale qui est l'image entière, si cette image vérifie un critère d'homogénéité (similarité), l'algorithme s'arrête, sinon on divise en quatre sous régions de taille égale. Chaque sous région est ensuite analysée, l'algorithme récursif s'arrête lorsque toutes les régions respectent le prédicat d'homogénéité.

A titre d'illustration, nous nous proposons de segmenter l'image de la figure I.8 par division de région.

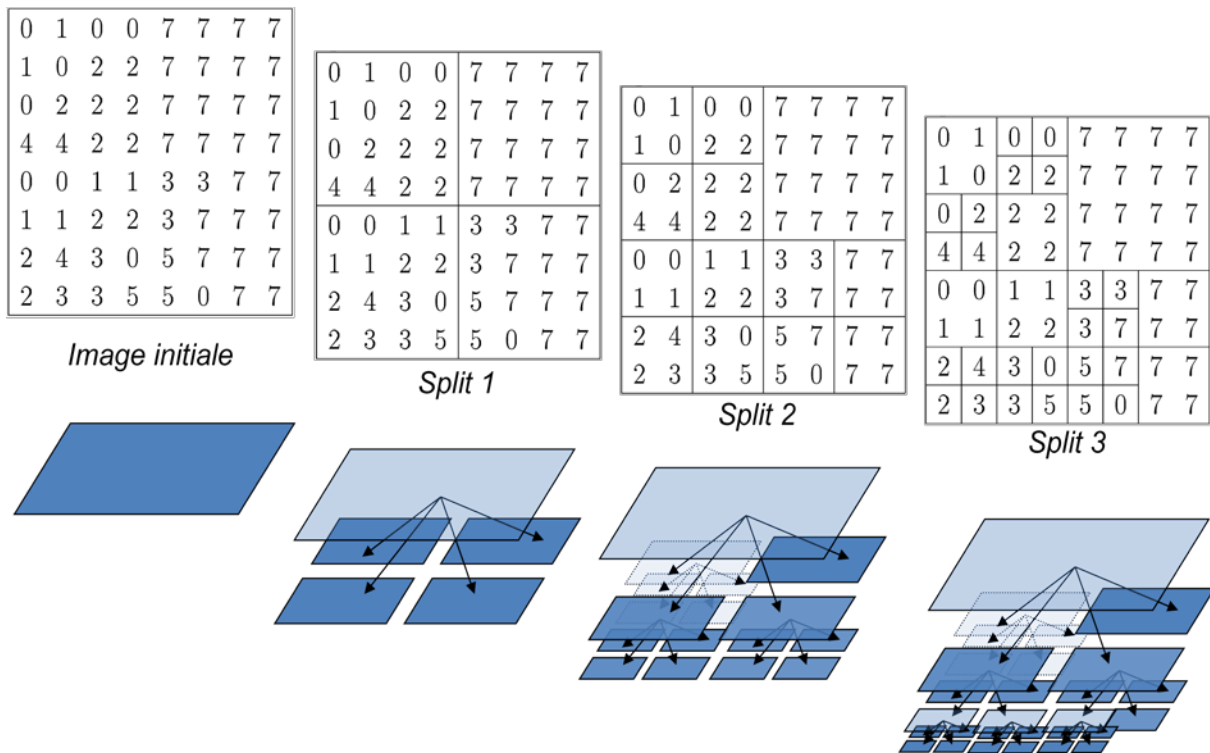


Figure I.8 : Exemple de segmentation par division de région

I.6.2.2 Segmentation par croissance de région

Ce type de segmentation consiste à faire croître des régions en y ajoutant successivement les pixels adjacents qui satisfont un critère d'homogénéité. La croissance s'arrête lorsque tous les pixels ont été traités.

L'avantage de la croissance de région est de préserver la forme de chaque région de l'image. Cependant une mauvaise sélection des régions entraîne des phénomènes de sous segmentation ou de sur segmentation. Un exemple de segmentation par croissance de région est donné par la Figure I.9

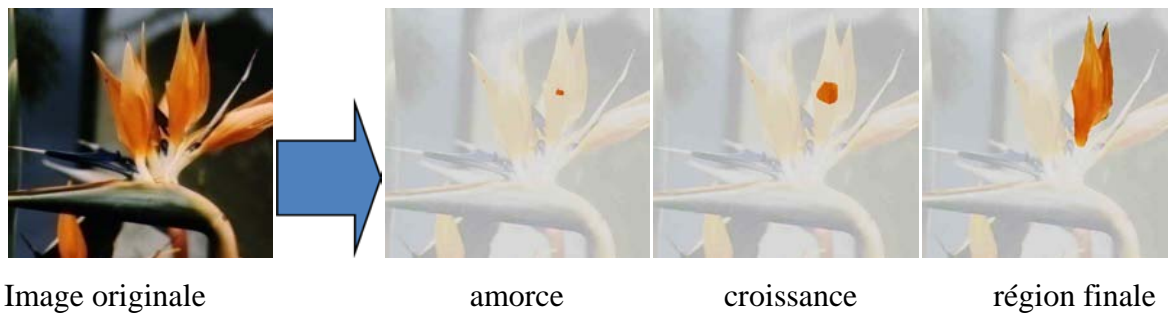


Figure I.9 : Exemple de segmentation par croissance de régions

I.6.2.3 Segmentation par division-fusion de région

La procédure de découpage décrite dans la méthode de division de région aboutit à un nombre de régions trop élevé. La cause fondamentale de cette sur-segmentation est que l'algorithme découpe les régions de manière arbitraire. Il se peut qu'il coupe de cette façon une zone homogène en deux ou en quatre parties. Alors La phase de regroupement consiste à connecter les régions adjacentes et homogènes.

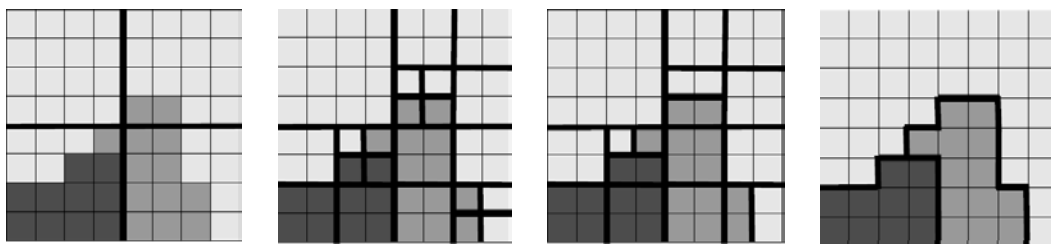


Figure I.10 : Exemple de division-fusion de région

I.6.3 La classification [8]

Dans cette approche la segmentation sera vue comme étant un problème de classification de données. Les techniques de classification identifient les classes des pixels homogènes dans l'espace des caractéristiques utilisées (niveau de gris, paramètres de texture,...) sous la forme de nuages de points opaques. Ensuite chaque classe est étiquetée donnant ainsi l'image des étiquettes.

Les techniques de classification automatique sont détaillées au chapitre suivant.

I.7 Définition de la texture

La définition littéraire de la texture est la "répétition spatiale d'un même motif dans différentes directions de "l'espace".

Haralick [7] élargit la définition en décrivant une texture comme un phénomène à deux dimensions: la première concernant la description d'éléments de base ou primitives (le motif) à partir des quels est formée la texture, la deuxième dimension est relative à la description de l'organisation spatiale de ces primitives.

En pratique, nous distinguons deux grandes classes de textures, qui correspondent à deux niveaux de perception: les macro-textures et les micro-textures.

Les macro-textures présentent un aspect régulier, sous forme de motifs répétitifs spatialement placés selon une règle précise (ex: peau de lézard, mur de brique) elles nécessitent une approche structurale déterministe pour leur analyse [3].

Les micro-textures présentent des primitives "microscopiques" distribuées de manière aléatoire (ex: sable, laine tissée, herbe) [3].

I.8 Analyse de texture

L'analyse de texture regroupe un ensemble de techniques mathématiques permettant de quantifier les différents niveaux de gris présents dans une image en termes d'intensité et de distribution. Le but de l'analyse de texture est de formaliser les descriptifs de la texture par des paramètres mathématiques qui serviraient à l'identifier [4].

Comme il existe deux grandes classes de texture, deux approches seront donc nécessaires: les méthodes statistiques et les méthodes structurales.

I.8.1 Les méthodes statistiques

Elles étudient les relations entre un pixel et ses voisins. Elles sont utilisées pour caractériser des structures fines, sans régularité apparente. L'étude se fait sur des structures tout à fait aléatoires et le plus souvent non homogènes, c'est pourquoi ce type de méthode sera préférentiellement utilisé. Dans ce cas, la texture est décrite par les statistiques de la distribution de ces niveaux de gris (ou intensité).

I.8.2 Les méthodes structurales

Ces méthodes tiennent compte de l'information structurale et contextuelle d'une forme et sont particulièrement bien adaptées aux textures macroscopiques. Elles permettent de décrire la texture en définissant les primitives et les "règles" d'arrangement qui les relie. En

effet, les textures ordonnées possèdent des primitives qui se répètent dans les images en des positions selon une certaine loi.

I.9 Les attributs statistiques de texture [2]

La distinction aisément faite par l'œil humain entre plusieurs textures est une tâche difficile à réaliser en vision par ordinateur, dans la mesure où il existe un nombre infini de texture et où chacune possède ses propres caractéristiques. Il n'existe pas de méthodes capables de classifier toutes les textures aussi bien que le ferait un observateur humain, cependant un certain nombre de propriétés définissant les impressions visuelles peuvent être extraites et permettent par conséquent de reproduire au mieux la classification de texture effectuée par cet observateur humain.

Ces propriétés sont à la base de nombreuses méthodes d'analyse de texture et ont permis de définir un grand nombre d'attributs caractérisant les textures présentes dans les images. Ces attributs peuvent être géométriques, basés sur la modélisation spatiale des textures, spatiaux-fréquentiels et statistiques.

Les attributs statistiques permettent de caractériser tout types de textures, même les textures fines et sans régularité apparente. Ces attributs peuvent être divisés en plusieurs catégories selon leur "ordre".

L'ordre des attributs dépend du type d'interaction spatiale entre les pixels considérés, il est donné par le nombre de pixels mis en jeu dans le calcul des paramètres. Par exemple pour les histogrammes d'image, on ne s'intéresse qu'au pixel lui-même, ce paramètre appartient donc à la catégorie des attributs statistiques de premier ordre.

Quant au calcul des matrices de cooccurrence, se sont les couples de pixels qui sont considérés, ce sont des attributs d'ordre deux.

I.9.1 Attributs statistiques du premier ordre [8]

De nombreux attributs peuvent directement être extraits de l'image afin de caractériser la texture qu'elle contient. Les attributs statistiques du premier ordre donnent des informations générales caractérisant une image en fonction de la variation de l'intensité des pixels qui la compose.

Les plus courants sont :

- La moyenne
- La variance
- Le moment d'ordre 3

- Le moment d'ordre 4

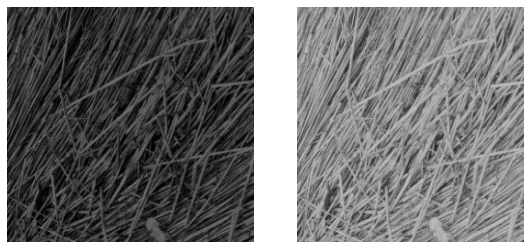
I.9.1.1 La moyenne

La valeur moyenne (ou intensité moyenne) des niveaux de gris de tous les pixels est donnée par :

$$MOY = \frac{1}{N} \sum_{x,y} I(x, y) \quad (\text{I.13})$$

avec $I(x, y)$ représente la valeur du niveau de gris du pixel (x, y) et N correspond au nombre total des pixels.

La figure I.11 illustre un exemple de la même image texturée mais ayant des moyennes différentes.



a) $MOY=50.48$ b) $MOY=137.01$

Figure I.11: Illustration de la moyenne sur une même image de Brodatz.

I.9.1.2 La variance

Elle correspond au moment d'ordre 2. Elle mesure la répartition des niveaux de gris autour de la valeur moyenne. Sa valeur est donnée par :

$$VAR = \frac{1}{N} \sum_{X,Y} (I(X, Y) - MOY)^2 \quad (\text{I.14})$$

La figure I.12 fournit deux exemples d'images ayant des variances différentes.

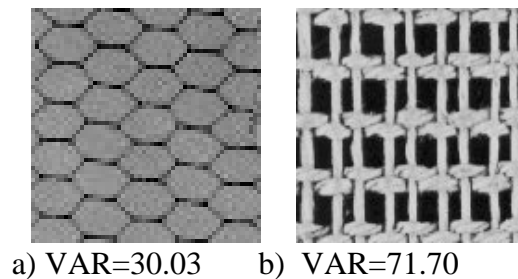


Figure I.12 : Illustration de différentes valeurs de la variance

Les images ci-dessus correspondent à deux textures ayant la même moyenne (127.7) mais des variances différentes. Plus la variance est élevée figure I.12.b et plus il y a d'écart importants entre les valeurs et la moyenne

I.9.1.3 Le moment d'ordre trois

Le moment d'ordre 3 (en anglais skewness) centré autour de la moyenne mesure la déviation de la distribution des niveaux de gris par rapport à une distribution symétrique. Il est donné par :

$$M3 = \frac{1}{N} \sum_{x,y} (I(x,y) - MOY)^3 \quad (I.15)$$

Il mesure le degré (d'asymétrie) des niveaux par rapport à leur moyenne. Pour une déviation vers les valeurs élevées le moment d'ordre trois est positif, alors que pour une déviation vers les basses valeurs, il est négatif.

La figure I.13 fournit deux exemples d'images et leurs histogrammes ayant des skewness différents.

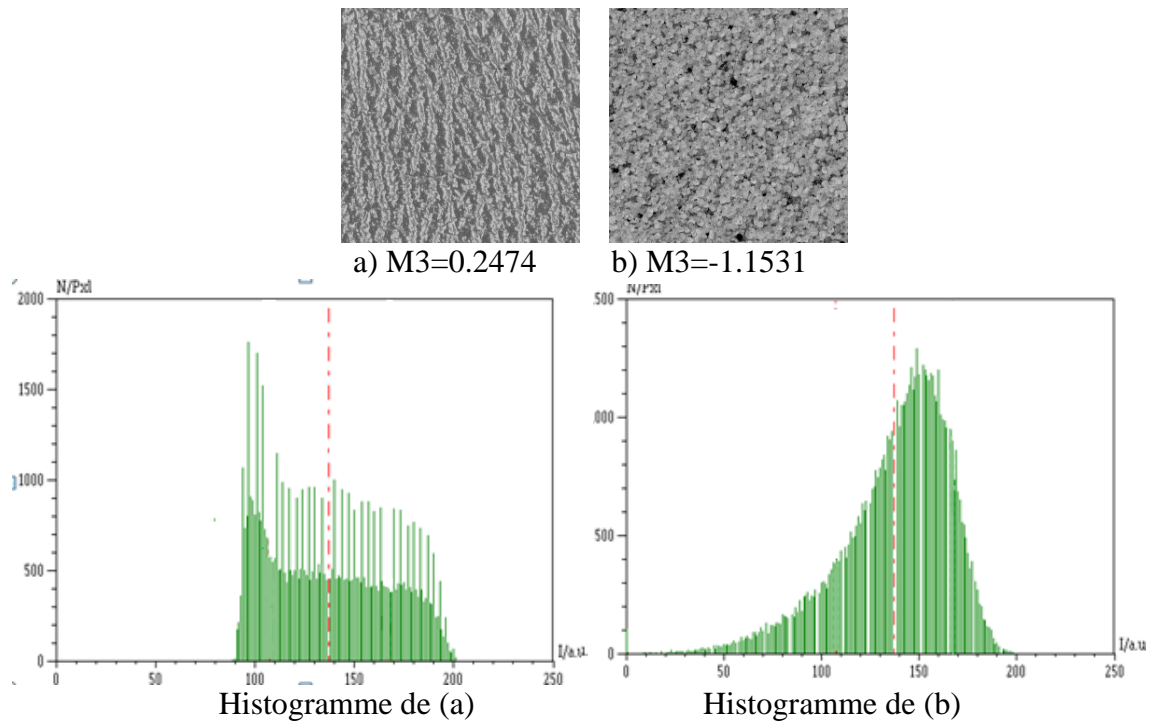


Figure I.13 : Illustration du moment d'ordre trois

Ces deux images possèdent la même moyenne(138). En revanche leurs skewness sont différents. La texture de la figure I.13.a qui possède un skewness positif possède un histogramme décalé vers la droite.

I.9.1.4 Le moment d'ordre quatre

Le moment d'ordre 4 centré autour de la moyenne (en anglais kurtosis).

$$KURT = \frac{1}{N} \sum_{x,y} (I(x,y) - MOY)^4 \quad (I.16)$$

Ce paramètre estime le degré de concavité ou convexité des niveaux par rapport à la moyenne. La figure I.14 illustre deux exemples d'images et leurs histogrammes ayant des kurtosis différents.

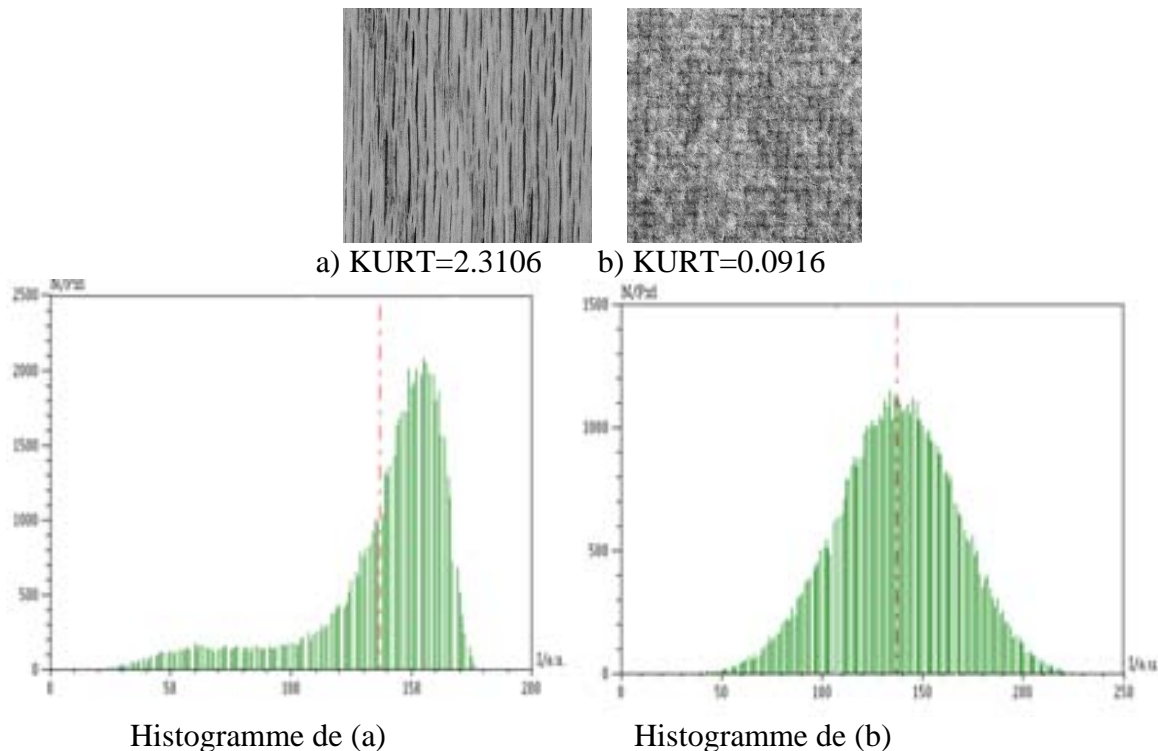


Figure I.14: Illustration sur les différentes valeurs du moment d'ordre quatre

On peut constater que la texture qui possède un moment d'ordre 4 plus élevé (figure I.14.a) possède une distribution décentrée vers les valeurs élevées de niveaux de gris.

I.9.2 Attributs statistiques du second ordre [2]

Dans les attributs de premier ordre, il n'y a pas d'informations sur la localisation du pixel. Il est donc nécessaire d'utiliser des méthodes d'ordre supérieur pour une analyse plus précise.

On retiendra, en particulier la méthode de la matrice de cooccurrence, dite aussi méthode de dépendance spatiale des niveaux de gris.

Elle permet de déterminer la fréquence d'apparition d'un "motif" formé de deux pixels séparés par une certaine distance "d" dans une direction particulière θ . Afin de limiter le nombre de calculs, on prend généralement comme valeurs de la direction θ 0° , 45° , 90° , 135° , 180° et 1 pour la valeur de "d".

Ces indices, bien que corrélés, réduisent l'information contenue dans la matrice de cooccurrence et permettent une meilleure discrimination des textures.

Le choix du vecteur de déplacement, et de la taille de la fenêtre du voisinage sur laquelle s'effectue la mesure, sont les paramètres sur lesquels reposent la réussite de la

méthode. La difficulté à surmonter lors de l'application de cette technique réside justement dans le choix de ces paramètres car ils varient selon le type d'images et de textures.

A chaque direction θ et pour chaque valeur de d correspond une matrice de cooccurrence $\varphi(d, \theta)$. La figure I.15 illustre un exemple de construction de la matrice de cooccurrence pour $\theta=0^\circ$ et $d=1$

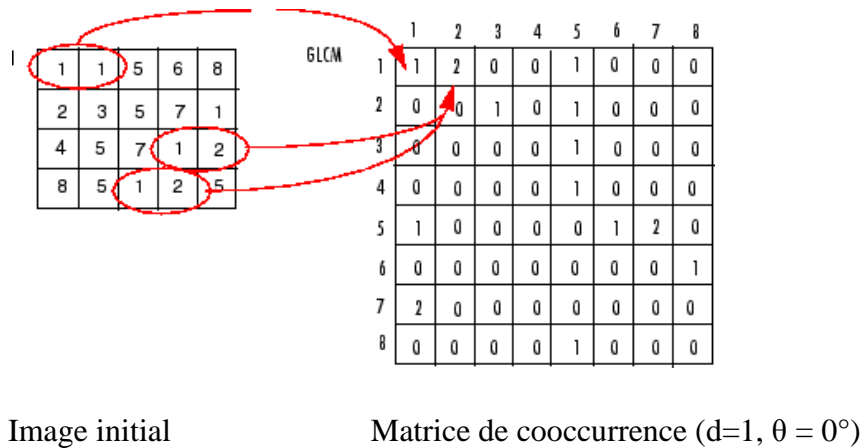


Figure I.15 : Exemple de construction d'une matrice de cooccurrence

On définit généralement les matrices symétriques de cooccurrence. Elles sont construites à partir des constatations suivantes :

$$\begin{aligned} \varphi(d, 0^\circ) &= \varphi^t(d, 180^\circ) \\ \varphi(d, 45^\circ) &= \varphi^t(d, 225^\circ) \\ \varphi(d, 90^\circ) &= \varphi^t(d, 270^\circ) \\ \varphi(d, 135^\circ) &= \varphi^t(d, 315^\circ) \end{aligned}$$

Par exemple, la matrice symétrique associée à la direction 0° sera de la forme :

$$S_0(d) = \frac{1}{2} [\varphi(d, 0^\circ) + \varphi(d, 180^\circ)] \tag{I.17}$$

Une fois la matrice symétrique est réalisée, il est possible d'en extraire une quinzaine de paramètres qui contiennent des informations sur la finesse et la granularité de la texture.

Pour une texture grossière, les valeurs de la matrice sont concentrées sur la diagonale principale. Au contraire, pour une texture fine, les valeurs de la matrice seront dispersées : en effet, pour une telle texture il existe beaucoup de transitions de niveaux de gris.

A partir de cette matrice de cooccurrence on peut extraire la moyenne, La variance, l'énergie, le contraste, la corrélation et l'homogénéité.

La moyenne est donnée par la relation :

$$\text{MOY} = \frac{1}{Ng \times Ng} \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} p(i, j) \quad (\text{I.18})$$

$p(i, j)$: correspond aux éléments de la matrice de cooccurrence, c'est à dire à la probabilité de passer d'un pixel de niveau de gris i à un pixel de niveau de gris j .

i, j : deux niveaux de gris

Ng : correspond au maximum des niveaux de gris de l'image.

La variance est définit par :

$$\text{VAR} = \frac{1}{Ng \times Ng} \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (i - \text{MOY})^2 p(i, j) \quad (\text{I.19})$$

Elle caractérise la distribution des niveaux de gris autour de la valeur moyenne MOY calculée précédemment.

L'énergie ou moment angulaire d'ordre deux mesure l'homogénéité de l'image. L'énergie a une valeur d'autant plus faible qu'il y a peu de zones homogènes : dans ce cas, il existe beaucoup de transitions de niveaux de gris.

$$E = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} p(i, j)^2 \quad (\text{I.20})$$

Le contraste ou (l'inertie) mesure les variations locales des niveaux de gris. Si elles sont importantes (c'est à dire s'il existe peu de régions homogènes), alors le contraste sera élevé. Ce paramètre permet aussi de caractériser la dispersion des valeurs de la matrice par rapport à sa diagonale principale.

$$\text{CONT} = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (i - j)^2 p(i, j) \quad (\text{I.21})$$

La corrélation est donnée par :

$$\text{COR} = \frac{\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} ij p(i,j) - \mu_i \mu_j}{\sigma_i \sigma_j} \quad (\text{I.22})$$

Où μ_i et μ_j représentent les moyennes respectivement des lignes et des colonnes de la matrice et σ_i et σ_j représentent les écarts types respectivement des lignes et des colonnes de la matrice.

L'homogénéité a un comportement inverse du contraste. Plus la texture possède de régions homogènes, plus le paramètre est élevé. Elle est donnée par :

$$\text{HOM} = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} \frac{1}{1+|i-j|} p(i,j) \quad (\text{I.23})$$

I.9.3 Attributs statistiques d'ordre supérieur [10]

Les attributs statistiques d'ordre supérieur sont extraits en étudiant les interactions entre plusieurs pixels. La méthode des longueurs de plages de niveaux de gris est l'une des plus connues. Elle consiste à compter le nombre de plages d'une certaine longueur j , de niveau de gris i dans une direction θ donnée (généralement θ vaut $(0^\circ, 45^\circ, 90^\circ$ ou $135^\circ)$). Une **plage** de niveaux de gris correspond donc à l'ensemble des pixels d'une image ayant la même valeur de niveau de gris. La longueur de la plage correspond au nombre de pixels appartenant à la plage. A chaque direction θ correspondra donc une matrice $R(\theta)$ donnée par :

$$R(\theta) = [r_\theta(i, j)] \quad (\text{I.24})$$

avec $r(i, j)$: élément de la matrice $R(\theta)$, représente nombre de plages de niveau de gris i , de longueur j dans la direction θ . La figure suivante donne un exemple de construction de la matrice de longueur de plages de niveau de gris.

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

Image initiale

$i \backslash j$	1	2	3	4
0	4	0	0	0
1	4	2	1	0
2	3	0	0	0
3	5	1	0	0

Matrice $R(0^\circ)$ de longueur de plages associé

Figure I.16 : Exemple de construction de la matrice de longueur de plages de niveaux de gris

Les principaux paramètres issus de la matrice de longueur de plages sont le "poids" des plages courtes, le "poids" des plages longues et la distribution des niveaux de gris.

Le "poids" des plages courtes caractérise la finesse de la texture. Plus la valeur de cet attribut est grande, plus les zones ayant le même niveau sont petites. Il est défini par la relation :

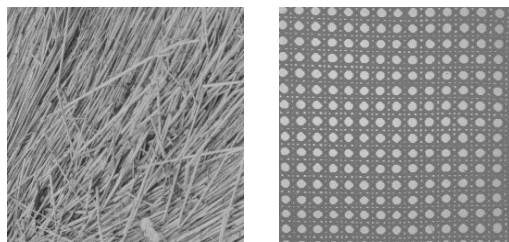
$$\text{SRE} = \frac{1}{nr} \sum_{i=0}^{M-1} \sum_{j=1}^N \frac{r(i,j)}{j^2} \quad (\text{I.25})$$

Avec $nr = \sum_{i=0}^{M-1} \sum_{j=1}^N r(i,j)$: représente le nombre total de plages de l'image.

M correspond au nombre de niveaux de gris dans l'image et N à la longueur de la plage maximale.

Le "poids" des plages longues caractérise les textures grossières. Plus la valeur de cet attribut est grande plus il y'a des zones étendues ayant le même niveau. Il est donnée par :

$$\text{LRE} = \frac{1}{nr} \sum_{i=0}^{M-1} \sum_{j=1}^N j^2 r(i,j) \quad (\text{I.26})$$



LRE=0.053

LRE=0.72

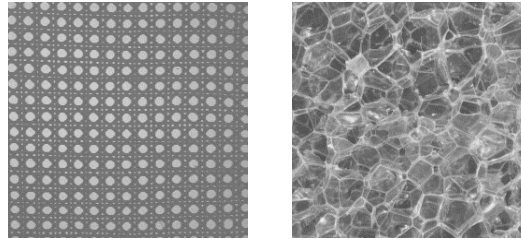
Figure I.17 : Illustration du paramètre LRE

Nous constatons bien que l'image qui possède un LRE plus grand possède plus de plages, c'est à dire plus de zones étendues ayant le même niveau de gris.

La distribution des niveaux de gris mesure l'uniformité de la distribution des plages. Cet attribut est minimal lorsque les plages sont également distribuées entre les différents niveaux. Elle est défini par :

$$\text{RLDIST} = \frac{1}{nr} \sum_{i=0}^{M-1} \left[\sum_{j=1}^N r(i, j) \right]^2 \quad (\text{I.27})$$

Ce paramètre augmente si le nombre de plages de même longueur augmente



RLDIST=17.93

RLDIST=35.75

Figure I.18 : illustration du RLDIST

I.10 Conclusion

Dans ce chapitre nous avons présenté d'une manière générale quelques notions essentielles de l'image, les notions et les attributs de texture qui décrivent les régions, puis les principales méthodes d'analyse de texture. Nous avons défini aussi les différentes techniques de segmentation.

L'une des stratégies possibles pour la segmentation est celle de la classification des pixels. Nous utiliserons dans notre travail une méthode de classification non supervisée à savoir la méthode des k-means pour classer les pixels sélectionnés de l'image et une méthode supervisée ressemblant à la méthode des K plus proches voisins pour classer les pixels non sélectionnés. Ces deux méthodes et d'autres seront détaillées dans le prochain chapitre.



CHAPITRE II

Méthodes de classification

II.1 Introduction

Ce chapitre présente quelques méthodes consistant à classer des individus en exploitant leur similarité. Les individus peuvent être :

- ✓ Les pixels eux-mêmes : les attributs sont alors ponctuels (comme le niveau de gris) ou calculés sur un voisinage du pixel (comme la moyenne des intensités ou un indice de texture calculé sur une même fenêtre centrée sur le pixel).
- ✓ Une sous image de taille fixe : les attributs sont calculés pour cette sous image, par exemple on peut découper l'image en fenêtres ne se chevauchant pas et classer chaque fenêtre.
- ✓ Une région obtenue par une méthode de segmentation que l'on cherche à classer.

Il est nécessaire de disposer d'un critère de comparaison des objets tel que la distance euclidienne.

II.2 Définition de la classification

La classification consiste à organiser un ensemble de données multidimensionnelles en un ensemble fini de classes selon un ou plusieurs critères de classification à l'aide d'un classifieur [9].

La figure II.1 illustre le principe de la classification. Les données sont représentées par des points (vecteurs) dans un espace à n dimensions (dans notre exemple $n=2$). En sortie de la classification, nous obtenons m classes de points selon les critères donnés et le classifieur choisi (dans notre exemple, $m=3$).

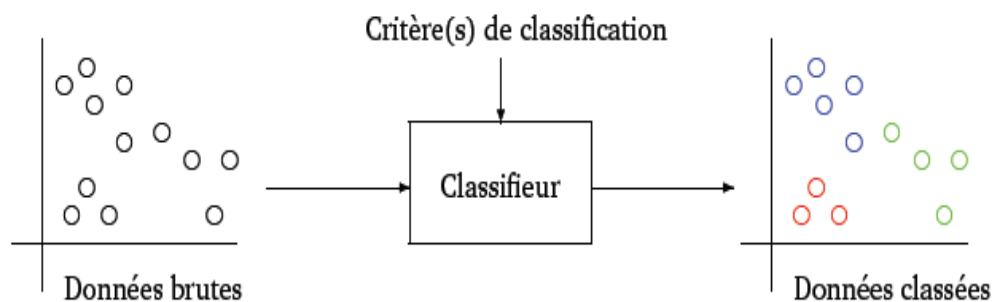


Figure II .1 : Le principe de la classification

En classification, les objets ou observations sont définis par un vecteur de mesures. Ce vecteur de dimension P correspond à un ensemble de P attributs ou variables (caractéristiques) définies a priori. Ces observations sont généralement représentées comme des points projetés dans un espace à P dimensions.

Les observations ayant des attributs semblables appartiennent à la même classe et forment un nuage de points très dense dans l'espace P dimensions. Ainsi, la classification en général est définie comme étant l'action de construire une collection d'objets similaires au sein d'un même groupe, dissimilaires quand ils appartiennent à des groupes différents. Elle est communément connue sous le nom anglophone « Clustering ».

Mathématiquement la classification est définie de la manière suivante :

Soit $X = \{ X_1, X_2, \dots, X_N \}$ l'ensemble des N observations à classer.

Chaque observation X_i est caractérisée par P paramètres.

$X_i = \{ x_{i1}, x_{i2}, \dots, x_{ip} \}$ tel que $X_i \in R^P$

Soit $C = \{ C_1, C_2, \dots, C_K \}$ l'ensemble des K classes.

La classification consiste à répartir l'ensemble des N observations en K classes tel que :

1. $C_i \neq \emptyset$ pour $i=1,2,\dots,K$
2. $C_i \cap C_j = \emptyset \forall i \neq j$
3. $\bigcup_{i=1}^K C_i = X$

II.3 Les techniques de classifications

Pour partitionner l'ensemble des observations en un certain nombre de classes distinctes, on peut faire appel aux différentes méthodes de classification automatique qui peuvent être distinguées par leur caractère supervisé et non supervisé.

II.3.1 Classification supervisée (semi automatique) [9]

Elles sont dites supervisées si l'on dispose a priori de l'information telle le nombre de classes possibles et l'appartenance de chaque individu à l'une de ces classes. La plupart des algorithmes d'apprentissage supervisés tentent donc de trouver un **modèle** qui explique le lien entre des données d'entrée et les classes de sortie.

Par exemple dans le cas de la classification de documents, on fournit donc à la machine des exemples sous la forme (Document, Classe). Cette méthode de raisonnement est appelée **inductive** car on induit de la connaissance à partir des données d'entrée et de sortie. Grâce à

ce modèle, on peut alors déduire les classes de nouvelles données : le modèle est utilisé pour prédire. Le modèle est bon s'il permet de bien prédire.

II.3.2 Classification non supervisées (automatique) [1]

L'expression non supervisée fait référence au fait qu'aucun superviseur ou label est utilisé pour préciser à quelle classe appartient un individu. En conséquence, le nombre de classes existant dans un ensemble d'individus est a priori inconnu. De ce fait, l'un des problèmes les plus délicats à propos des méthodes de classification non supervisées concernent le choix du nombre de classes à retenir. Pour palier cet écueil, il existe des artifices permettant d'approcher le bon nombre de classes. Une fois le nombre de classes est choisi, le prochain pas dans le processus de classification consiste à évaluer la qualité de la partition obtenue.

La classification non supervisée (classification automatique, regroupement ou (clustering en anglais) a pour but de regrouper des individus en classes homogènes en fonction de l'analyse des caractéristiques qui décrivent les individus. Par classes homogènes, on entend regrouper les individus qui se ressemblent et séparer ceux qui sont dissemblables.

II.4 Quelques méthodes de classification

Nous proposons de décrire quelques méthodes de classification supervisées et non supervisées qui peuvent être monodimensionnelles ou multidimensionnelles [8].

Les méthodes monodimensionnelles ne prennent en compte qu'un seul attribut (en général le niveau de gris). Elles reposent généralement sur l'exploitation de l'histogramme. La plupart d'entre elles extraient automatiquement des seuils puis affectent les pixels à une classe par comparaison de leur niveau de gris à ces seuils.

Les méthodes multidimensionnelles exploitent plusieurs attributs, tel que chaque individu est associé à un vecteur d'attributs.

Parmi les différentes méthodes de classification nous pouvons citer, La méthode de K plus proches voisins, l'approche densité des informations, l'approche hiérarchique, la classification floue et la méthode des k-means.

II.4.1 Méthode des K plus proches voisins

Elle est connue en anglais sous le nom K-nearest neighbor (KNN) (ou encore Memory Based Reasoning)[9]. C'est une méthode de classification supervisée sauf qu'elle diffère des traditionnelles méthodes d'apprentissage par le fait qu'aucun modèle n'est induit à partir des exemples. Les données restent telles quelles : elles sont simplement stockées en mémoire.

Pour prédire la classe d'un nouvel individu, l'algorithme cherche les K plus proches voisins de ce nouvel individu et prédit la réponse la plus fréquente de ces K plus proches voisins. La méthode utilise donc deux paramètres.

- ✓ Le nombre K.
- ✓ La fonction de similarité pour comparer le nouvel individu aux individus déjà classés.

Pour déterminer les K plus proches voisins d'un nouvel individu, nous calculons souvent la distance de cet individu aux différents individus connus.

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (\text{II.1})$$

Avec

$a_r(x_i)$: Attribut du nouvel individu.

$a_r(x_j)$: Attribut de $j^{\text{ième}}$ individu classé.

Avec $j=1, \dots, N$ et N : nombre d'individus classés

r : Indice de l'attribut

n : Nombre d'attribut

$d(x_i, x_j)$ Distance entre le nouveau individu i et le $j^{\text{ième}}$ individu classé.

Dans KNN de base, on choisit la classe majoritairement représentée par les K plus proches voisins. Une autre solution est de pondérer la contribution de chaque K plus proche voisin en fonction de sa distance avec le nouvel individu à classer.

Notez qu'avec cette méthode de pondération, on pourrait utiliser les N exemples au lieu des K plus proches voisins : en effet, plus un exemple sera éloigné du nouvel individu à classer et moins sa classe contribuera au résultat final.

Les expériences menées avec les KNN montrent qu'ils résistent bien aux données bruitées. Par contre, ils requièrent de nombreux exemples classés.

Notez aussi que, si le temps d'apprentissage est inexistant puisque les données sont stockées telles quelles, la classification d'un nouveau individu est par contre coûteuse puisqu'il faut comparer ce cas à tous les exemples déjà classés.

La figure II.2 illustre un exemple de classification par KPV. On doit classer le nouvel individu X_q . Si on choisit $K = 1$, X_q sera classé +. Si $K = 5$, le même X_q sera classé - On voit donc que le choix de K est très important dans le résultat final.

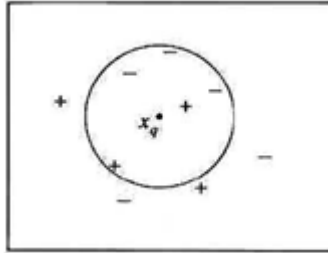


Figure II.2: Exemple de classification par les K- plus proche voisins

II.4.2 L'approche densité des informations [9]

Les méthodes basées sur la densité sont des méthodes de classification non supervisées dans les quelles, on cherche à regrouper l'ensemble des points situés dans un voisinage d'un point non classé selon une certaine distance. Pour atteindre ce but, on définit une distance maximale entre deux points (voisinage) ainsi qu'un nombre de voisins minimal (densité). Ensuite, on procède au classement des données selon l'algorithme suivant :

- 1- Choisir au hasard un point non classé
- 2- Former un groupe contenant ce point et les points de son voisinage dense
- 3- Chercher le voisinage dense de chaque point du groupe et l'ajouter au groupe,
- 4- Tant que le groupe n'est pas stable aller à 3
- 5- S'il y a des points non classés aller à 1 sinon finir

La figure II.3 illustre un exemple de classification par l'approche densité des informations

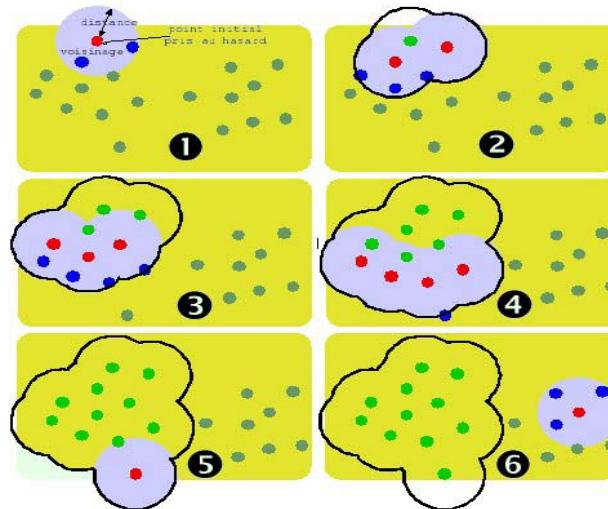


Figure II.3 : Exemple de classification à l'aide des voisinages denses

Dans cette figure en (1) on choisit au hasard un point non classé. Ce point et son voisinage sont classés dans le même groupe. En (2) on prend le voisinage de chacun des points du groupe (2) et on les classe dans le même groupe et ainsi de suite (3), (4) et (5) jusqu'à la stabilité du groupe obtenu en (6). On recommence ensuite avec un autre point de départ non classé (6) et on réitère le processus jusqu'à ce que tous les points soient classés.

Cette méthode est précise pour trouver le nombre de classes présentes dans les données sans indication préalable, cependant, elle souffre de certains inconvénients : elle nécessite de nombreux calculs, elle est très sensible au choix de la distance et du nombre minimum de voisins.

II.4.3 L'approche hiérarchique

L'approche hiérarchique [9] consiste à construire les familles par agglomération ascendante de classes. On construit les classes par agglomération successive de deux groupes de données dont les centres de gravité sont plus proches qu'une distance d_{\max} donnée.

L'algorithme de classification hiérarchique (ou agglomération) est le suivant :

- 1- Regrouper deux par deux les points les plus proches
- 2- Tant que les groupes ne sont pas stables
 - Calculer le centre de gravité des groupes ainsi créés.
 - Regrouper deux par deux les groupes dont les centres ont une distance inférieure à d_{\max} .

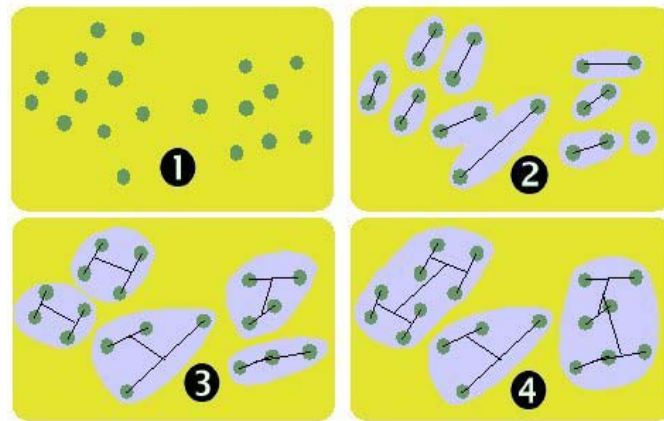


Figure II.4 : Exemple de classification hiérarchique

La figure II.4 présente un exemple d'agglomération de classes. L'ensemble des points (1) est traité pour regrouper les points les plus proches par classes de deux (2). Les centres des classes sont calculés et les classes regroupées par deux (3). Enfin, quand les groupes obtenus sont trop éloignés pour être fusionnés, l'algorithme s'arrête.

Cette approche ne nécessite pas beaucoup de calculs et converge vite. Par contre, elle ne sait traiter que des ensembles de données de cardinalité proche de 2^n .

II.4.4 La méthode de k-means [9]

Les méthodes de classification non hiérarchique ou de partitionnement ont pour but de fournir une partition des éléments à classer. Le nombre de classes dans la partition à générer doit être fixé au départ. Le principe des méthodes de partitionnement est le suivant :

- ✓ A partir d'un ensemble E de n individus, on cherche à construire une partition $P = (C_1, \dots, C_k)$ contenant k classes homogènes et distinctes, tout en respectant un critère basé sur une distance entre les individus.
- ✓ Doivent se trouver dans une même classe les individus qui se ressemblent et en classes différentes les individus divergents en termes du critère adopté.
- ✓ Une partition optimale peut être obtenue à partir de l'énumération exhaustive de toutes les partitions, ce qui devient cependant prohibitif en termes de temps de calcul. Comme solution alternative à ce problème, des méthodes de partitionnement basées sur l'optimisation itérative du critère à respecter permettent d'obtenir des groupes bien distincts en un temps de calcul raisonnable. Ces méthodes d'optimisation utilisent une réaffectation afin de redistribuer itérativement les individus dans les k classes.

Parmi les nombreuses méthodes de classification non-hiérarchique, la méthode des k-means est largement utilisée car elle est inspirée de la stratégie de regroupement de la vision humaine. Les k-means fonctionnent sur le principe de la recherche par raffinements successifs. Cette méthode offre une certaine similitude avec la méthode humaine de recherche par tâtonnement : on classe d'abord les individus en paquets grossiers, puis on cherche à affiner le résultat du classement en tendant vers une solution idéale.

L'algorithme de k-means est le suivant :

- 1- Choisir aléatoirement k centres de classe.
- 2- Affecter les individus aux classes dont la distance est minimale
- 3- Calculer les nouveaux centres de gravité des classes
- 4- Affecter chaque individu à la nouvelle classe
- 5- Tant que le critère d'arrêt n'est pas vérifié aller à 3

Le centre de gravité d'une classe est calculé comme suit :

$$g_j = \frac{1}{N} \sum_{i=1}^N x_{ij} \quad (\text{II.2})$$

j : indice d'attribut.

N : nombres d'observations (individus ou pixels).

x_{ij} : attribut j de l'individu x_i

On choisit souvent comme critère d'arrêt, la stabilité des classes, ce critère peut être déterminé par plusieurs techniques, parmi ces techniques nous avons :

- ✓ Lorsque les centres des classes ne changent pas durant deux itérations successives, c'est-à-dire lorsque

$$g(n, p) = g(n, p+1) \quad (\text{II.3})$$

$n = 1, \dots, k$

Avec $g(n, p)$: Centre de gravité de classe n à la $p^{\text{ème}}$ itération.

La figure II.5 illustre un exemple de classification par la méthode des k-means

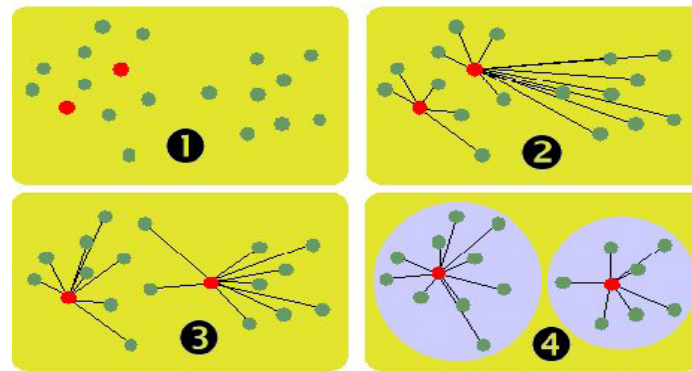


Figure II.5 Exemple de classification à l'aide des k-means

L'idée est de placer au hasard un nombre de centre de classes correspondant au nombre de classes souhaitées. Pour que l'algorithme donne de bons résultats, il est préférable d'avoir une connaissance a priori du nombre de classes présentes dans les données à regrouper. La classification est automatiquement obtenue par le déplacement itératif des centres de classes.

Cette technique n'implique pas de longs calculs et converge rapidement. Par contre, elle impose de donner le nombre de classes désirées au début de l'algorithme et elle est sensible aux conditions initiales.

Un effet de bord très connu des k-means est que les points isolés (*outliers*) qui ont tendance à attirer vers eux les centres de classes et donc à dénaturer le résultat de la classification.

Un autre problème connu est la séparation stricte des classes qui pénalise grandement les images en périphérie de classe.

La figure II.6 fournit un exemple de classification de l'image en niveau de gris

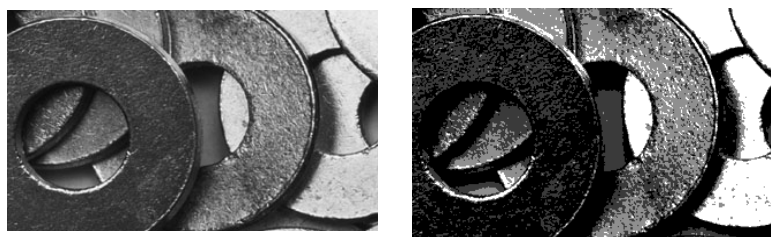


Image originale

Image classifiée

Figure II.6 : Exemple de classification d'une image en niveau de gris en 4 classes par la méthode des k-means

II.4.5 Classification floue [9]

À la fin de la classification par les k-means, nous obtenons des classes strictes, c'est-à-dire que chaque individu n'appartient qu'à une seule classe. Les individus situés en périphérie d'une classe, près de la frontière d'une autre classe, posent le problème de l'appartenance stricte à une classe unique.

Il serait intéressant d'introduire une notion plus floue d'appartenance à une classe pour permettre aux individus placés entre deux classes d'appartenir un peu à chacune d'elles. Afin de rendre la classification floue, nous allons effectuer après la classification un calcul de degré d'appartenance d'une image à une classe.

Soit un point M des données à classées et soient $d_i, i = 1 \dots k$ les distances du point M à chacun des k centres des classes. Le degré d'appartenance du point M à la classe i est défini

$$\text{Deg}_{M \in i} = \frac{100}{\sum_{p=1}^k \frac{d_i}{d_p}} \quad (\text{II.4})$$

Grâce au calcul du degré d'appartenance (qui est compris entre 0 et 100), nous avons un moyen de rendre la classification floue. On peut par exemple décider que l'individu appartient à toutes les classes pour les quelles son degré d'appartenance est supérieur à 30.

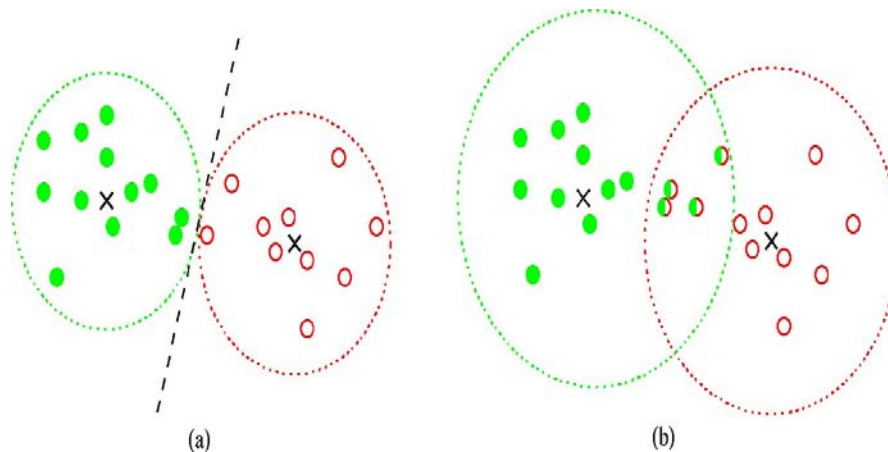


Figure II.6 Introduction de flou dans la classification. (a) Classification stricte, (b) Classification floue

La figure II.6 illustre le niveau de flou induit dans la classification. En (a), la classification est stricte et les points éloignés du centre de gravité de chacune des familles

sont classés dans une seule famille. En (b), la classification est floue, nous autorisons certains individus à appartenir à plusieurs familles.

Il existe une version floue de l'algorithme des k-means. Cet algorithme revient à calculer un degré d'appartenance flou dont le flou est réglé par un paramètre.

II.5 Conclusion

Nous avons évoqué dans ce chapitre les différentes méthodes de classification automatique en insistant particulièrement sur l'approche classification non supervisée et plus particulièrement sur la méthode des (k-means). Cette méthode représente l'avantage d'être rapide et simple, mais son inconvénient majeur est que la solution obtenue dépend étroitement des centres initiaux et aboutissent souvent à un optimum local.

Dans notre travail qui est développé dans le chapitre suivant, les centres initiaux seront choisis par l'utilisateur pour s'assurer de prendre de chaque classe un centre.



CHAPITRE III

Tests et résultats

III.1 Introduction

Le but de ce travail est de réaliser une segmentation d'image en ne considérant au préalable qu'un certain nombre de pixels. Il existe des méthodes de segmentation donnant des résultats satisfaisants mais qui nécessitent un temps de calcul important. Dans ce travail nous allons tester une méthode qui nous permettra de réduire ce temps de calcul. Au lieu de réaliser une classification pour tous les pixels de l'image, nous allons classer une partie de ces pixels et diffuser l'information sur le reste des pixels.

III.2 Principe de la méthode proposée

Nous allons dans ce travail segmenter des images en niveau de gris en utilisant des paramètres de texture calculés à partir de la matrice de cooccurrence. Les paramètres que nous utiliserons sont :

Le contraste

$$\text{CONT} = \sum_i \sum_j (i - j)^2 p(i, j) \quad (\text{III.1})$$

L'homogénéité

$$\text{HOM} = \sum_{i,j} \frac{1}{1+|i-j|} p(i, j) \quad (\text{III.2})$$

L'énergie

$$E = \sum_{i,j} (p(i, j))^2 \quad (\text{III.3})$$

La corrélation

$$\text{COR} = \frac{\sum_i \sum_j ij p(i, j) - \mu_i \mu_j}{\sigma_i \sigma_j} \quad (\text{III.4})$$

Tel que :

$p(i, j)$: correspond aux éléments de la matrice de cooccurrence, c'est à dire à la probabilité de passer d'un pixel de niveau de gris i à un pixel de niveau de gris j .

i, j : deux niveaux de gris.

i : allant 1 jusqu'à Ng.

j : allant 1 jusqu'à Ng.

Ng : correspond au maximum des niveaux de gris de l'image.

μ_i et μ_j et représentent les moyennes respectivement des lignes et des colonnes de la matrice.

Où σ_i et σ_j représentent les écarts types respectivement des lignes et des colonnes de la matrice.

Une fois que ces paramètres sont calculés pour chacun des pixels de l'image, nous réalisons une classification en utilisant l'algorithme des k-means (voir chapitre II. Paragraphe 4.4).

Pour réduire le temps d'analyse, nous proposons de calculer les paramètres de texture non pas pour tous les pixels de l'image mais pour un nombre nettement inférieur. Le choix de ces derniers se fera de manière aléatoire. Nous pourrions ainsi choisir le taux T de pixels sélectionnés. Sur les figures III.1, III.2 nous affichons des exemples de choix du taux T pour l'image LENNA et pour une image synthétique. Sur ces images, les pixels non sélectionnés sont représentés en noir et les pixels sélectionnés gardent leur niveau de gris.



Figure III.1: Image LENNA et sa représentation avec différents taux T de pixels sélectionnés
a) Image originale ($T=100\%$) b) $T=10\%$ c) $T=50\%$ d) $T= 75\%$.

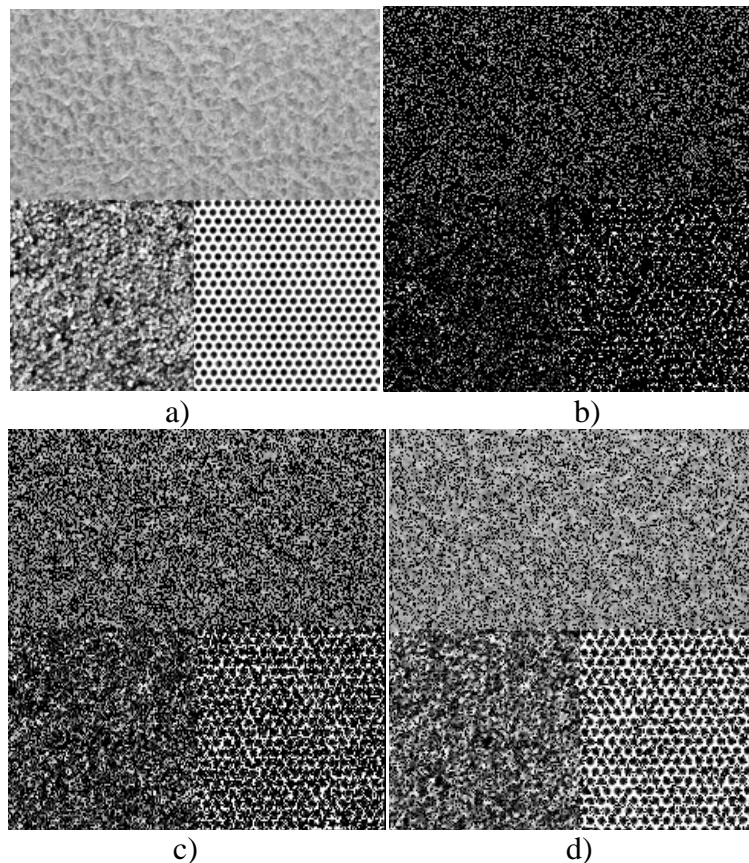


Figure III.2 : Image synthétique1 et sa représentation avec différents taux T de pixels sélectionnés

a) Image originale (T=100%) b) T=20% c) T=50% d) T= 75%.

Nous réalisons donc l'extraction des attributs de texture de ces pixels sélectionnés en utilisant la matrice de cooccurrence. Nous les classons par la suite par la méthode de classification non supervisée k-means. Cette méthode nécessite une initialisation des centres des classes qui peuvent être aléatoire ou non. Dans notre cas, nous réalisons une initialisation aléatoire pour les images réelles et une initialisation manuelle pour les images synthétiques. Sur les figures III.3 et III.4 nous affichons les images originales en a) suivies des images représentant les pixels sélectionnés en b). En c) nous donnons les images ou les pixels non sélectionnés (donc n'ayant subis aucune classification) sont représentés en noir et les pixels qui ont été classés sont représentés par le niveau de gris attribués à leurs classes.

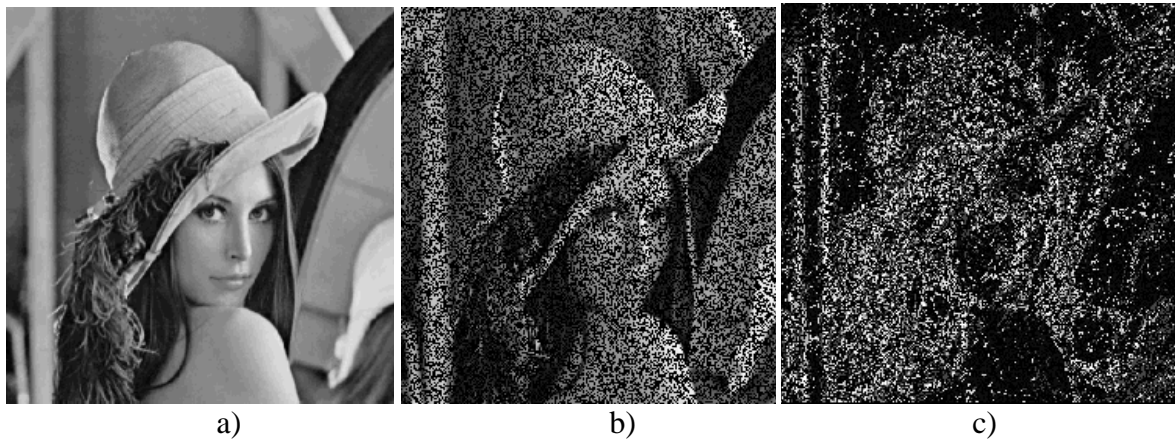


Figure III.3 : Classification des pixels sélectionnés de l'image LENA
 a) Image originale b) représentation des 50% de pixels sélectionnés c) représentation après classification en 10 classes des pixels sélectionnés

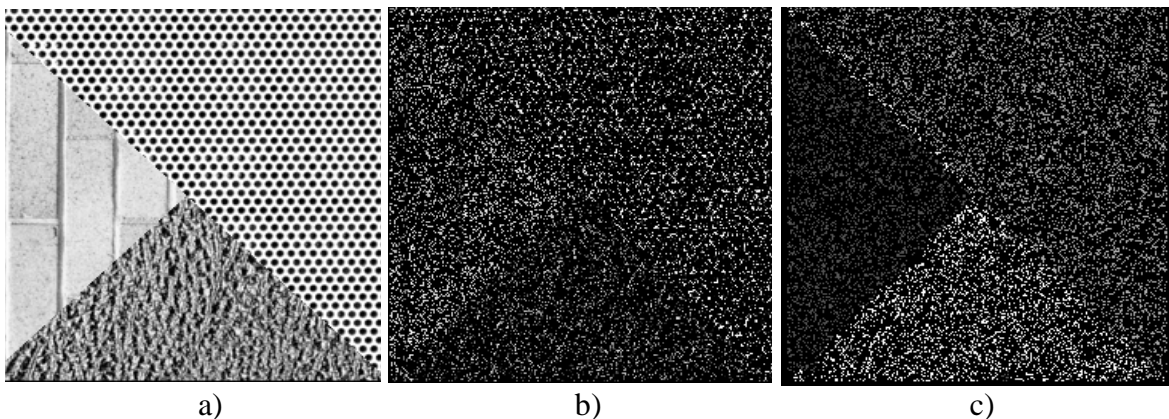


Figure III.4 : Classification des pixels sélectionnés de l'image synthétique2
 a) Image originale b) représentation des 20% des points sélectionnés c) représentation après classification en 3 classes des pixels sélectionnés

Une fois les pixels sélectionnés classés, nous passons à la classification des pixels non sélectionnés.

Sur l'image des étiquettes, les pixels non sélectionnés sont affectés de la valeur 0 et les pixels sélectionnés sont affectés de la valeur de leurs classes donc variant entre 1 et k tel que k représente le nombre de classes présent dans l'image.

Pour chaque pixel non sélectionné, nous comptons le nombre de pixels de la classe prédominante dans une fenêtre de taille $W \times W$ centrée en ce pixel. Si ce nombre est supérieur à $(W \times W)/2$, nous affectons à ce pixel le numéro de cette classe, dans le cas contraire, nous n'affectons aucune classe à ce pixel qui sera classé aux itérations suivantes. A la fin de cette étape, tous les pixels seront classés et nous obtenons ainsi l'image segmentée. La figure III.5 illustre la classification du pixel représenté en gris selon une fenêtre de voisinage 5×5 .

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

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

Figure III.5 : Exemple de classification d'un pixel non sélectionné

Sur la figure ci-dessous nous donnons les résultats de la classification des pixels non sélectionnés à des niveaux d'itérations différents. Les pixels non classés sont toujours représentés en noir.

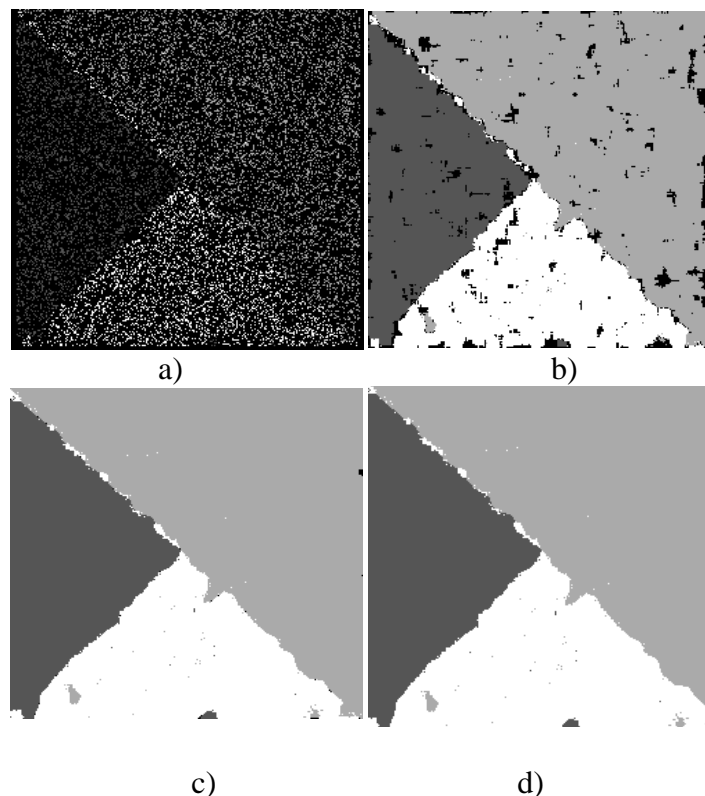


Figure III.6: Exemple de classification des pixels non sélectionnés sur l'image synthétique2.
 a) Classification des pixels sélectionnés b) 1^{ère} itération c) 2^{ème} itération
 d) 3^{ème} itération

En résumé, la méthode proposée se déroule en plusieurs étapes à savoir :

- La sélection d'un certain nombre de pixels,
- L'extraction des paramètres de texture de ces pixels,
- Leur classification et
- La classification des pixels non sélectionnés.

Nous avons donc implémenté cette méthode sous Matlab 7.8 et son utilisation nécessite le choix d'un certain nombre de paramètres qui sont :

- Le taux de pixels sélectionnés T ,
- La taille de la fenêtre de voisinage pour la classification des pixels sélectionnés $W \times W$,
- La direction θ utilisée pour calculer les paramètres de texture et
- La taille de la fenêtre $V \times V$ utilisée pour calculer la matrice de cooccurrence.

Dans ce qui suit, nous allons examiner les résultats de la segmentation lorsque nous varions un à un ces différents paramètres.

Les tests sont effectués sur deux images synthétiques et une image réelle en niveau de gris de même taille 256×256 pixels².

III.3 Influence du taux de pixels sélectionnés T

Dans ce paragraphe, nous proposons d'étudier l'influence du taux T de pixels sélectionnés sur la segmentation des images synthétiques et réelles. Pour ce faire, nous fixerons les valeurs des autres paramètres. Ainsi dans le cas des figures III.7, III.8 et III.9 la taille de la fenêtre d'analyse sera $W \times W = 3 \times 3$ pour Lenna, $W \times W = 9 \times 9$ pour les images synthétiques, la direction θ utilisée pour calculer les paramètres de texture est prise dans toutes les directions et la taille de la fenêtre utilisée pour déterminer la matrice de cooccurrence est $V \times V = 9 \times 9$. Le nombre de classe n est fixé suivant le nombre de textures présentes dans les images synthétiques et sera égale à 15 dans le cas de l'image Lenna. Pour chaque résultat nous donnons le temps total t mis pour réaliser la segmentation.

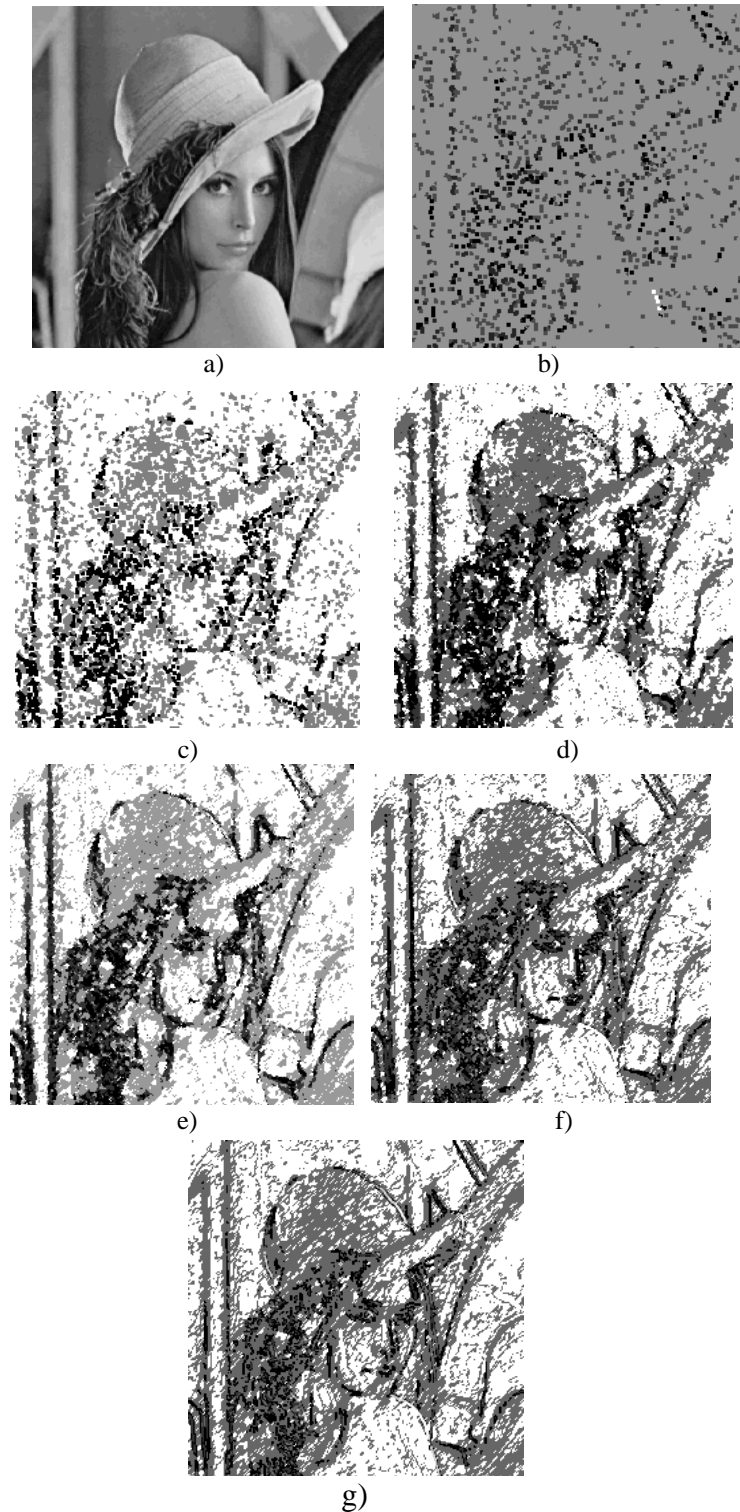


Figure III.7 : Exemple sur l'influence du taux de pixels sélectionnés T sur la segmentation de l'image LENA

a) Image originale b) $T=5\%$, $t=37.7246s$ c) $T=15\%$, $t=93.9584s$ d) $T=30\%$, $t=148.5372s$
e) $T=50\%$, $t=248.1762s$ f) $T=75\%$, $t=397.9908s$ g) $T=100\%$, $t=594.5124s$.

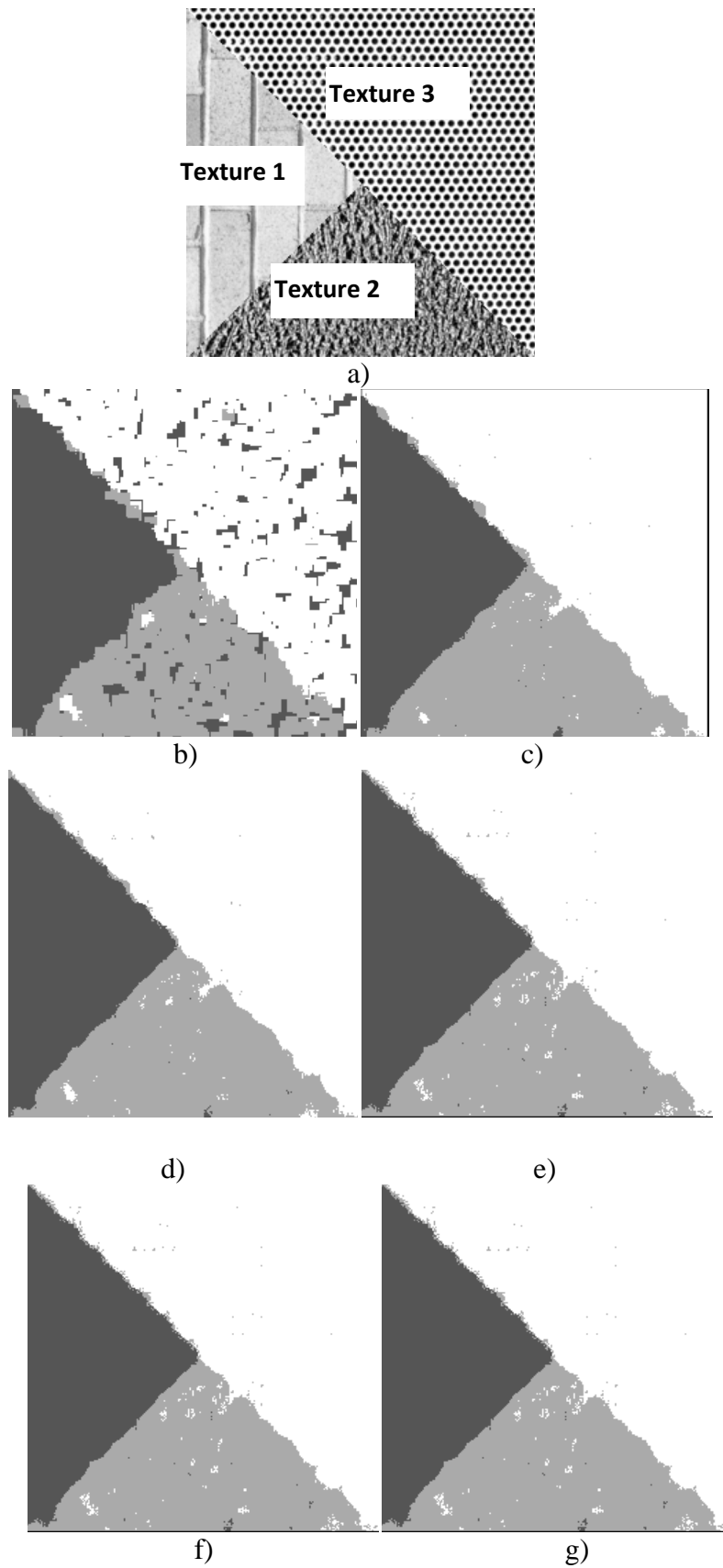


Figure III.8 : Exemple sur l'influence du taux de pixels sélectionnés T sur la segmentation de l'image synthétique2

a) Image originale b) $T=5\%$, $t=24,1254s$ c) $T=15\%$, $t=32,1451s$ d) $T=30\%$, $t=71,8259s$
 e) $T=50\%$, $t=116,5132 s$ f) $T=75\%$, $t=178,4705 s$ g) $T=100\%$, $t=246,6779s$.

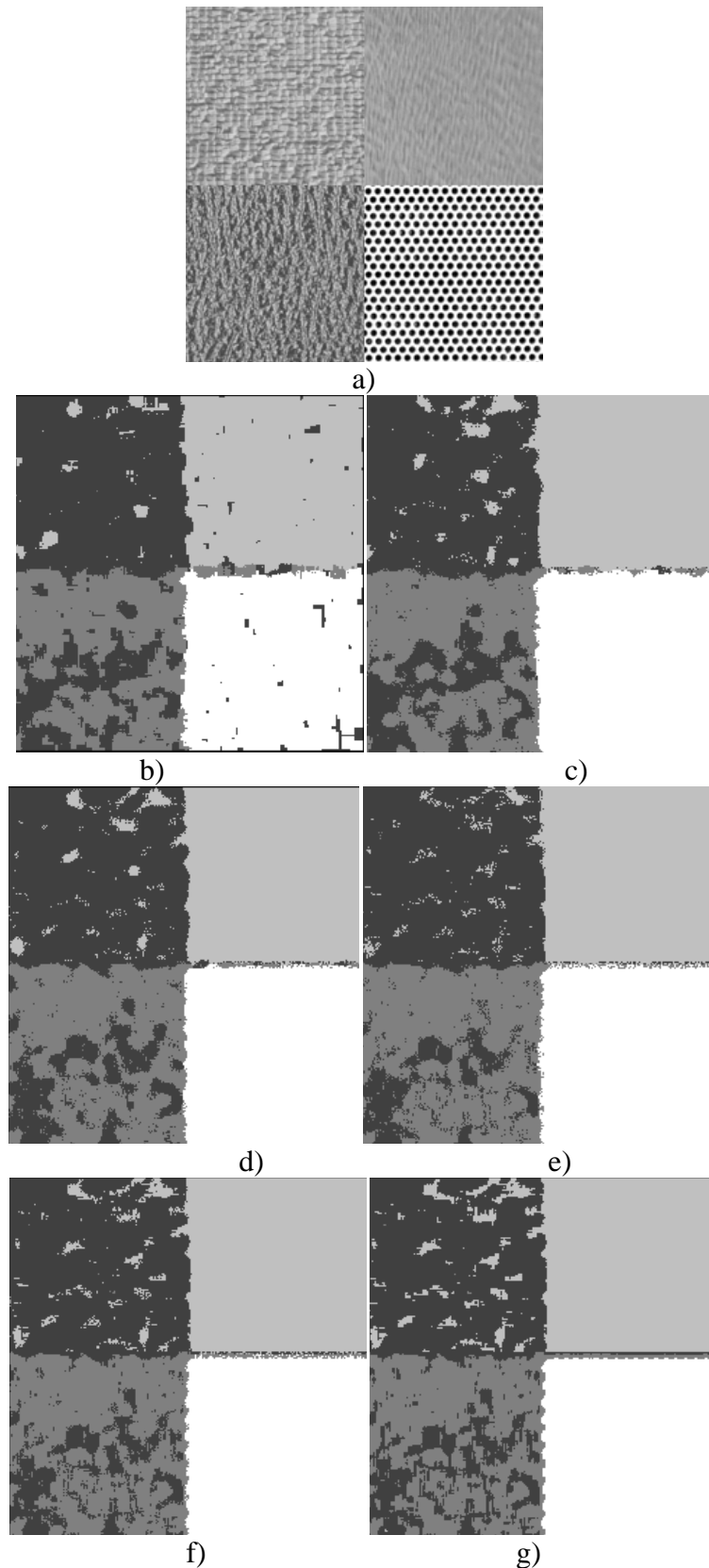


Figure III.9 : Exemple sur l'influence du taux de pixels sélectionnés T sur la segmentation de l'image synthétique3.

a) Image originale b) $T=5\%$, $t=25.1466s$ c) $T=15\%$, $t=31.7329s$ d) $T=30\%$, $t=68.4136s$
 e) $T=50\%$, $t=118.1872s$ f) $T=75\%$, $t=181.9010s$ g) $T=100\%$, $t=255.0188s$

En observant les résultats affichés sur les figures précédentes, nous remarquons que pour des taux de pixels sélectionnés assez faibles les résultats de la segmentation sont mauvais, ce qui était prévisible. Si nous observons la figure III.8. b) par exemple nous remarquons que les différentes textures sont mal séparées, en effet nous retrouvons des régions de texture1 dans la partie de l'image où nous devons retrouver que de la texture3. Le résultat est déjà meilleur si nous considérons un taux $T=15\%$ (figure III.8. c)).

Dans le cas de la figure III.9, nous tirons les mêmes conclusions, et nous observons que les textures représentées en blanc et en gris clair sont nettement séparées à partir de $T=15\%$ (figure III.9. c)).

Dans le cas d'une image réelle (voir figure III.7) nous remarquons que la classification est mauvaise dans le cas où $T=5\%$, elle s'améliore pour $T=15\%$, mais elle reste non satisfaisante, alors qu'à partir de $T=30\%$, les résultats sont bons comparativement aux résultats obtenus pour $T=100\%$.

Ainsi, dans tous les cas traités, nous constatons qu'un taux de 30% fournit des résultats appréciables. Il est alors préférable de réaliser la segmentation avec $T=30\%$ qu'avec $T=100\%$ car le temps de calcul est nettement inférieur. En effet pour $T=30\%$ le temps de calcul vaut $t=68.4136$ s pour l'image synthétique3 et $t=71.8259$ s pour l'image synthétique2 et pour LENA $t=148.5372$ s. Pour $T=100\%$, il vaut $t=255.0188$ s pour l'image synthétique3, $t=246.6779$ s, pour l'image synthétique 2, $t=594.5124$ s, pour l'image LENA.

Notre but est d'avoir un bon résultat en un minimum de temps, pour ces raisons, nous considérons $T=30\%$ pour la suite des tests.

III.4. Influence de la taille de la fenêtre d'analyse $W \times W$

Dans ce paragraphe, nous proposons d'étudier l'influence de la taille de la fenêtre d'analyse $W \times W$ sur la segmentation des images synthétiques et réelles. Pour ce faire, nous fixerons les valeurs des autres paramètres. Ainsi dans le cas des figures III.10, III.11, III.12 la direction θ utilisée pour calculer les paramètres de texture est prise dans toutes les directions, le taux de pixels sélectionné est $T=30\%$, la taille de fenêtre utilisée pour calculer la matrice de cooccurrence est $V \times V=9 \times 9$ et le nombre de classe n est fixé suivant le nombre de textures présentes dans les images synthétiques et est égale à 15 dans le cas de l'image LENA.

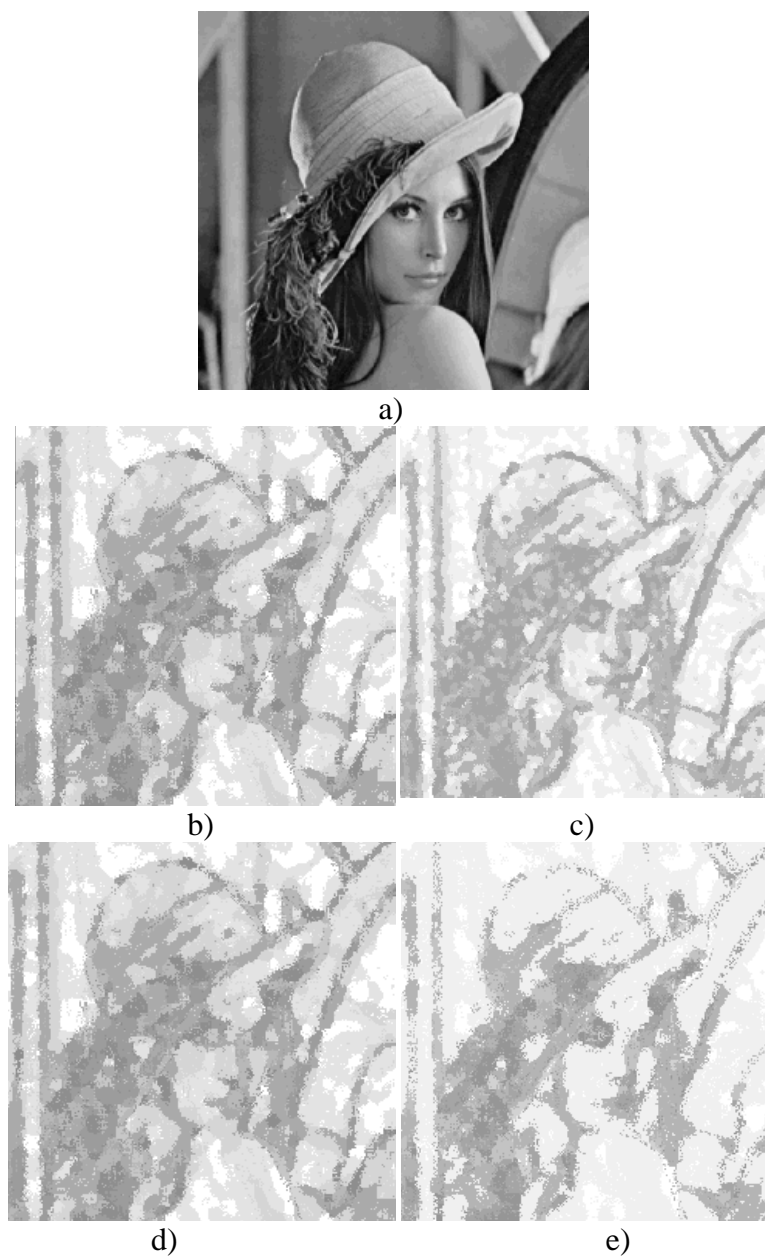


Figure III.10 : Influence de la taille fenêtre d'analyse sur la segmentation de l'image LENA

- a) Image originale b) $W \times W = 3 \times 3$, $t = 148.5372s$ c) $W \times W = 5 \times 5$, $t = 128.4655s$
d) $W \times W = 9 \times 9$, $t = 119.4298s$ e) $W \times W = 11 \times 11$, $t = 150.9079s$.

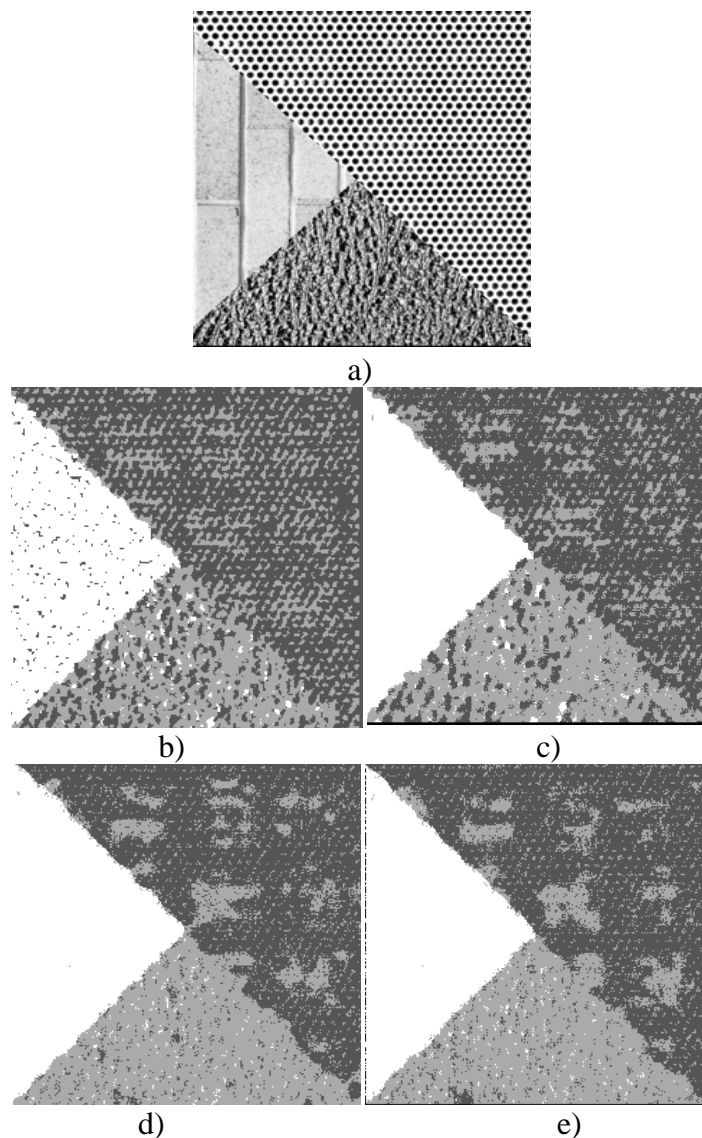


Figure III.11 : Influence de la fenêtre d'analyse sur la segmentation de l'image syntetique2

a) Image originale b) $W \times W = 3 \times 3$, $t = 76.9569$ s c) $W \times W = 5 \times 5$, $t = 75.5453$ s

d) $W \times W = 9 \times 9$, $t = 74.8794$ s e) $W \times W = 11 \times 11$, $t = 81.5073$ s.

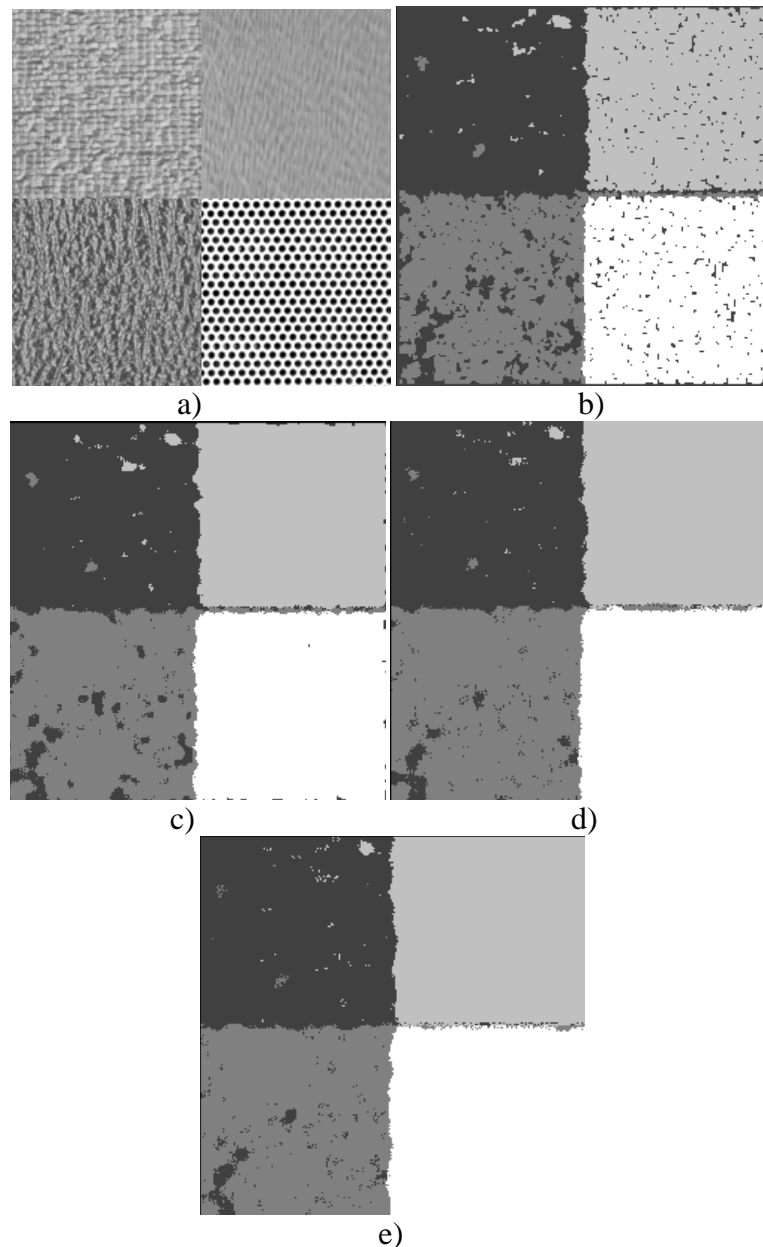


Figure III.12 : Influence de la fenêtre d'analyse sur la segmentation de l'image synthétique3

a) Image originale b) $W \times W = 3 \times 3$, $t = 84.3318s$ c) $W \times W = 5 \times 5$, $t = 83.6619s$

d) $W \times W = 9 \times 9$, $t = 75.4579s$ e) $W \times W = 11 \times 11$, $t = 58.7561s$.

En observant la Figure III.11.b) donnant le résultat de la segmentation de l'image synthétique2 pour une fenêtre d'analyse $W \times W = 3 \times 3$, nous constatons que le résultat est mauvais par rapport à des tailles de la fenêtre d'analyse supérieur. En effet si nous observons la zone de texture représentée en blanc, nous retrouvons un grand nombre de régions comportant peu de pixels d'une autre texture. Or généralement, lorsque nous réalisons une segmentation, en tente d'éliminer les régions comportant un faible nombre de pixels.

Si nous considérons une fenêtre d'analyse de taille faible, à l'intérieur d'une région texturée, le nombre de pixels utilisé pour classer notre point central sera faible. Par contre, si nous considérons une taille plus importante pour la fenêtre d'analyse, et si nous sommes toujours dans la même région texturée, le nombre de pixels utilisé pour classer notre point central sera plus important, donc la probabilité d'avoir un meilleur classement sera plus importante dans ce dernier cas. Nous retrouvons sur les figures III.11, III.12, de mauvais résultats pour de faibles tailles de la fenêtre d'analyse et ces derniers s'améliorent en augmentant la taille de fenêtre, ce qui est en accord avec le raisonnement apporté ci-dessus.

Ce raisonnement ne peut s'appliquer dans le cas où les zones texturées sont de dimensions faibles car en augmentant la taille de la fenêtre, nous risquons de prendre en considération des pixels qui sont de différentes classes. Il y'a donc un compromis à faire sur la taille de cette fenêtre. Pour ces différentes raisons, nous considérerons pour la suite des tests $W \times W = 9 \times 9$ pour les images synthétiques et $W \times W = 5 \times 5$ pour l'image LENNA.

III.5 Influence de la direction θ utilisée dans le calcul des paramètres de texture

Dans ce paragraphe, nous proposons d'étudier l'influence de la direction utilisée dans le calcul de paramètres de texture sur la segmentation des images synthétiques et réelles. Pour ce faire, nous fixerons les valeurs des autres paramètres. Ainsi dans le cas des figures III.13, III.14, III.15, nous considérons $W \times W = 9 \times 9$ pour les images synthétiques et 5×5 pour l'image réelle, $V \times V = 9 \times 9$, $T = 30\%$ et n est fixé suivant le nombre de textures présentes dans les images synthétiques et sera égale à 15 dans le cas de l'image LENNA.

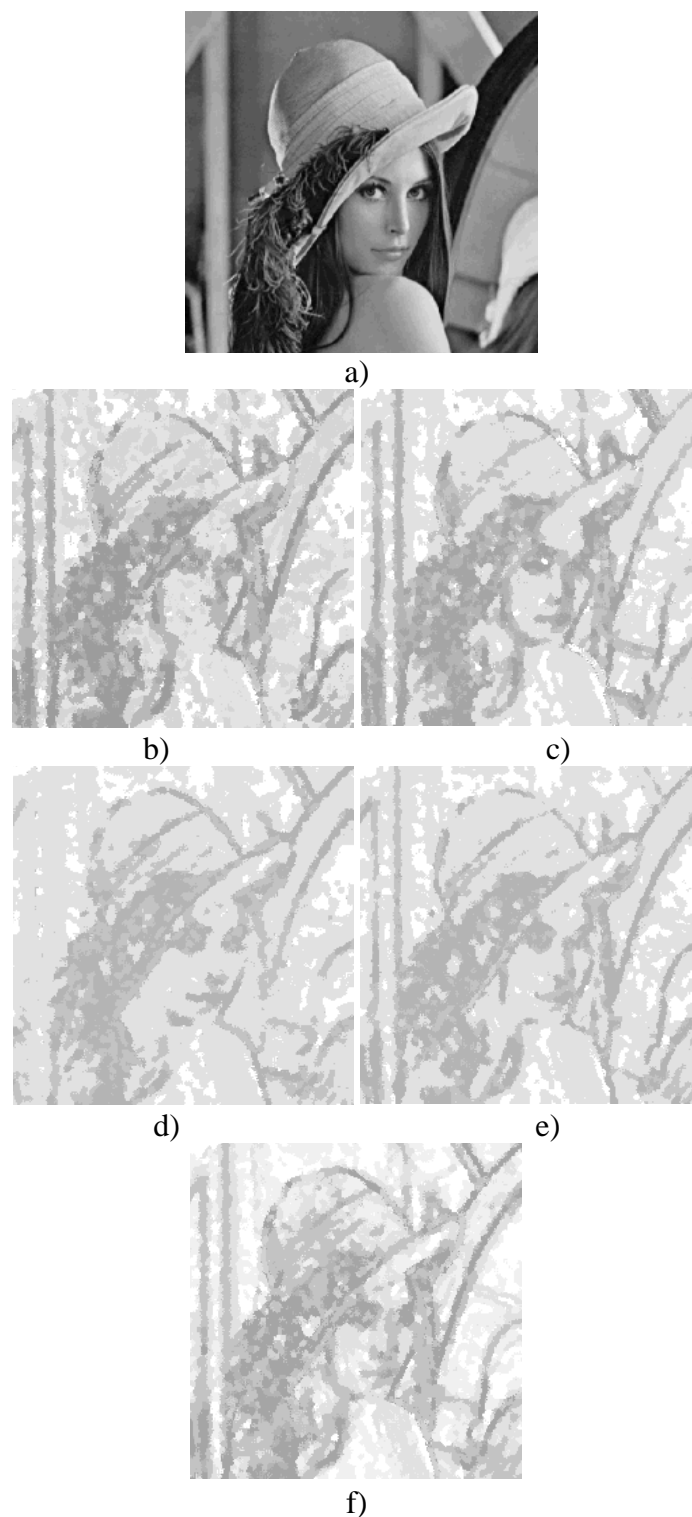


Figure III.13 : Influence de la direction du calcul de paramètres de texture sur la segmentation de l'image LENA

a) Image originale b) $\theta=0^\circ$, $t= 143.7412s$ c) $\theta=45^\circ$, $t=148.5372s$ d) $\theta=90^\circ$, $t=138.7262s$
e) $\theta=135^\circ$, $t=165.3304s$ f) Toutes les directions

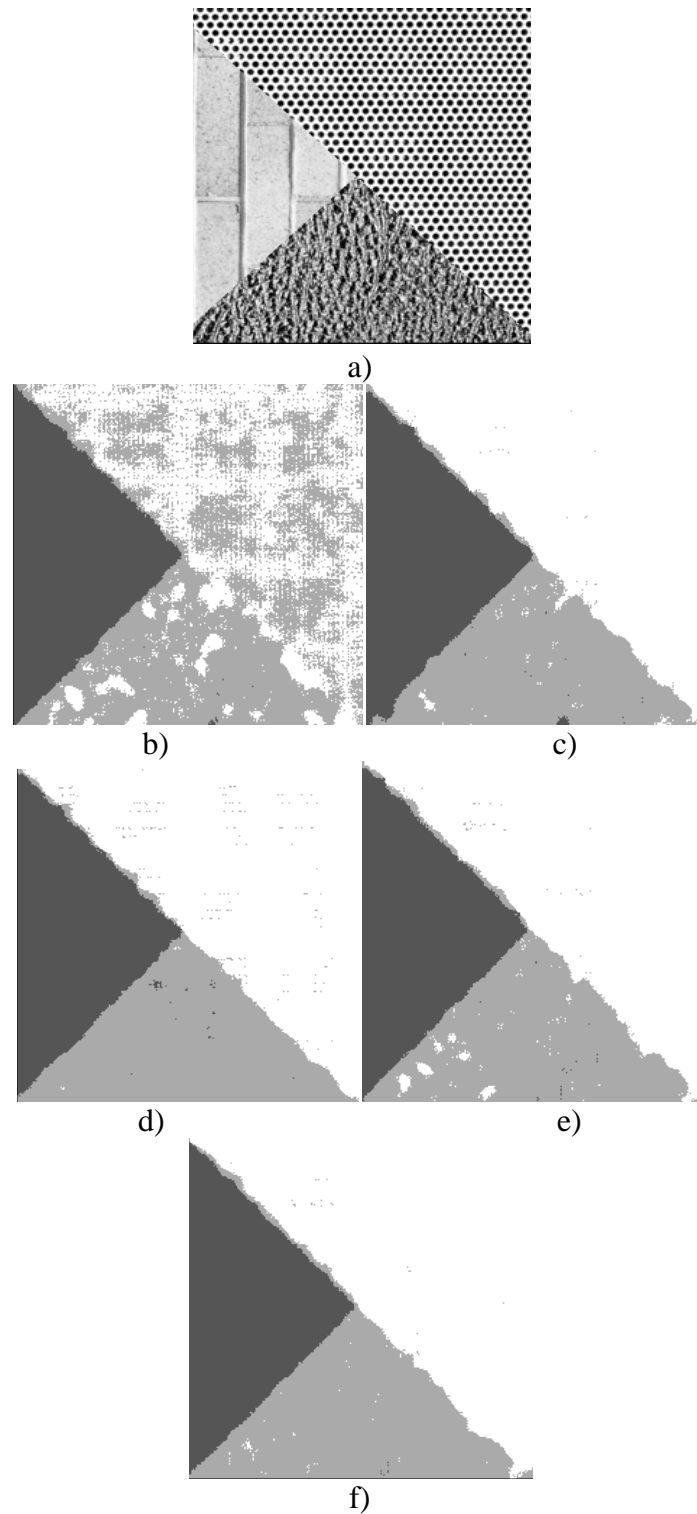


Figure III.14 : Influence de la direction du calcul de paramètres de texture sur la segmentation de l'image synthétique2

a) Image originale b) $\theta=0^\circ$, $t = 76.9685s$ c) $\theta=45^\circ$, $t= 59.0329s$ d) $\theta=90^\circ$, $t= 61.4644s$
e) $\theta=135^\circ$, $t= 64.4777 s$ f) toutes les directions, $t=88.5233 s$.

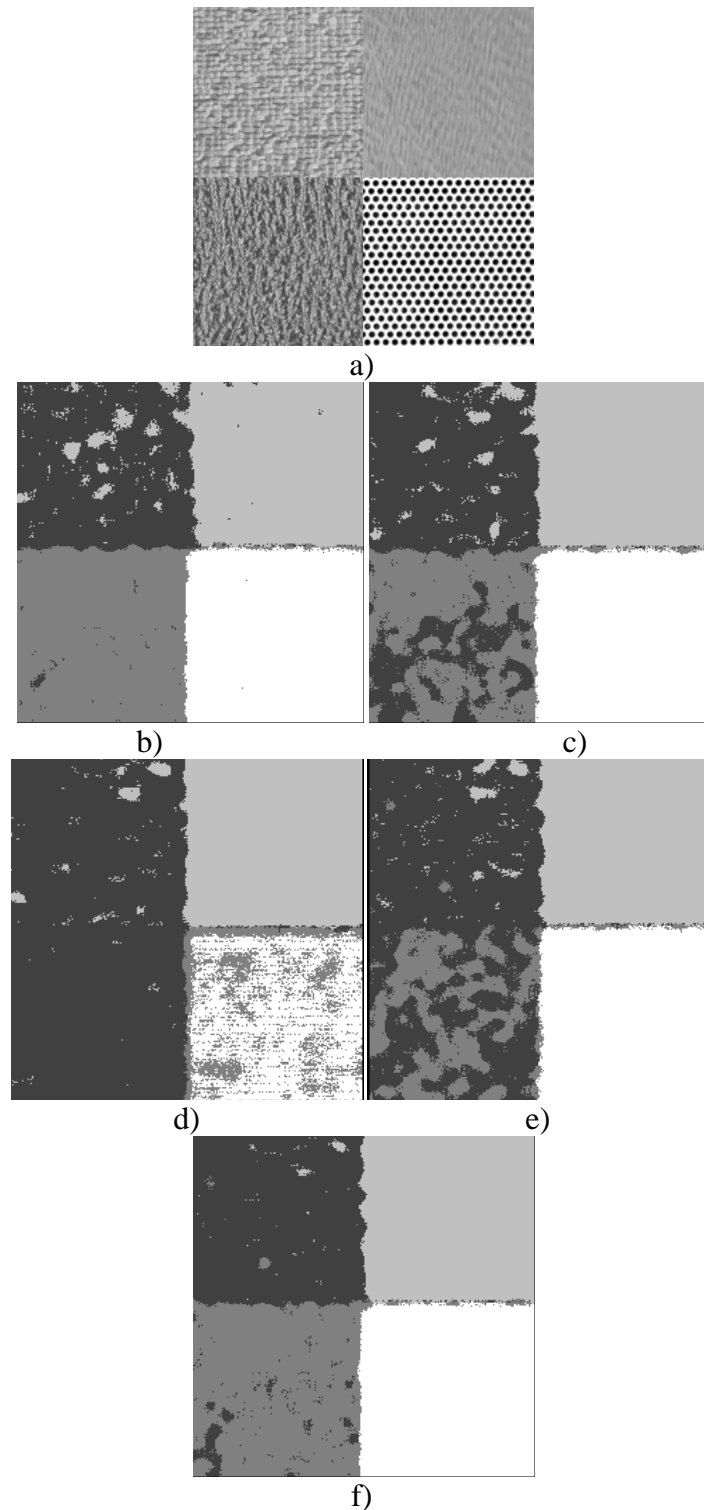


Figure III.15 : Influence de la direction du calcul de paramètres de texture sur la segmentation de l'image synthétique3

- a) Image originale b) $\theta=0^\circ$, $t=86.5022s$ c) $\theta=45^\circ$, $t=68.4136s$ d) $\theta=90^\circ$, $t=65.9170s$
 e) $\theta=135^\circ$, $t=60.2539s$ f) toutes les directions, $t= 90.6127 s$.

En observant les figures III.14, III.15 donnant les résultats de la segmentation des images synthétiques nous constatons que le meilleur résultat est obtenu en considérant toutes

les directions dans le calcul de la matrice de cooccurrence. En effet dans le cas par exemple de la figure III.14.b) nous constatons que les textures représentées en blanc et en gris claire se confondent pour $\theta=0^\circ$, le résultat est nettement meilleur par exemple dans le cas ou $\theta=90^\circ$, sauf que pour d'autres images nous observons un mauvais résultat dans ce cas (voir figure III.15.d).

Nous obtenons de bons résultats pour des directions privilégiées dans certaines images mais ces directions sont différentes d'une image à une autre. Il est donc préférable d'utiliser dans le calcul de la matrice de cooccurrence toutes les directions car dans ce cas, nous sommes sûre de prendre cette direction privilégiée.

III.6 Influence de la taille de la fenêtre utilisée $V \times V$ pour calculer la matrice de cooccurrence

Dans ce paragraphe, nous proposons d'étudier l'influence de la taille de la fenêtre de la matrice de cooccurrence utilisée dans le calcul des paramètres de texture sur la segmentation des images synthétiques et réelles. Pour ce faire, nous fixerons les valeurs des autres paramètres. Ainsi dans le cas des figures III.16, III.17, III.18, $T=30\%$, $W \times W=9 \times 9$ pour des images synthétiques et 5×5 pour l'image réelle, θ est prise dans toutes les directions et n est fixé suivant le nombre de textures présentes dans les images synthétiques et sera égale à 15 dans le cas de l'image LENA.

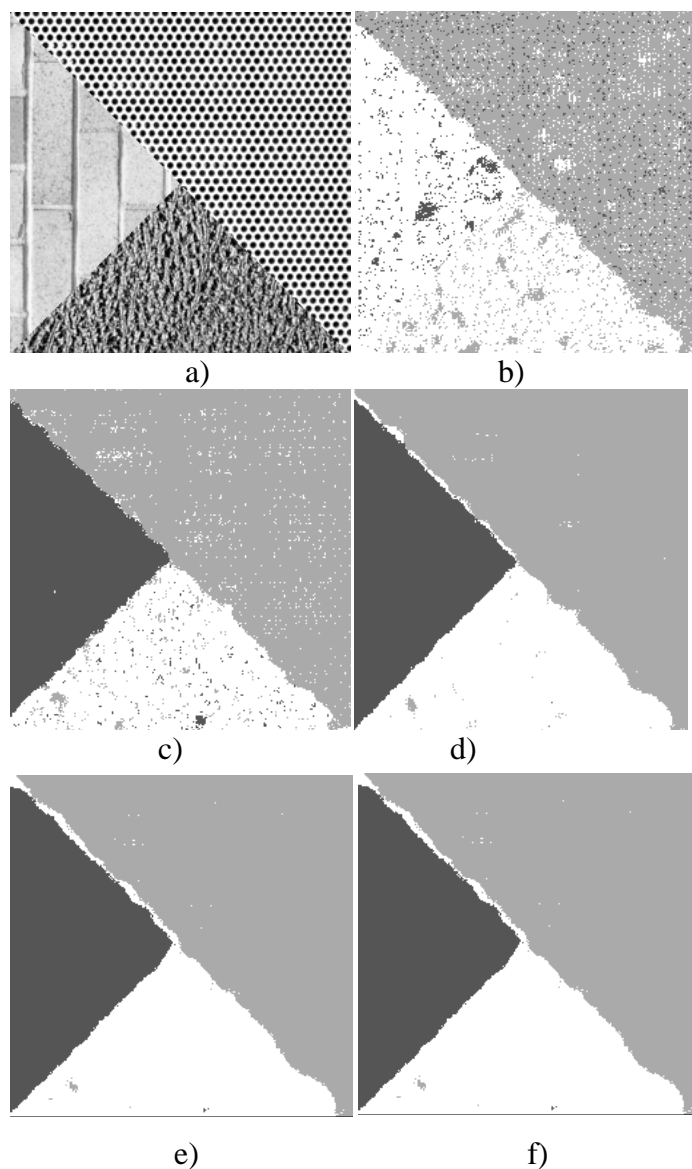


Figure.III.16 : Influence de la taille de la matrice de cooccurrence sur la segmentation de l'image syntetique2

a) Image originale b) $V \times V = 3 \times 3$ c) $V \times V = 5 \times 5$ d) $V \times V = 9 \times 9$ e) $V \times V = 11 \times 11$
f) $V \times V = 13 \times 13$.

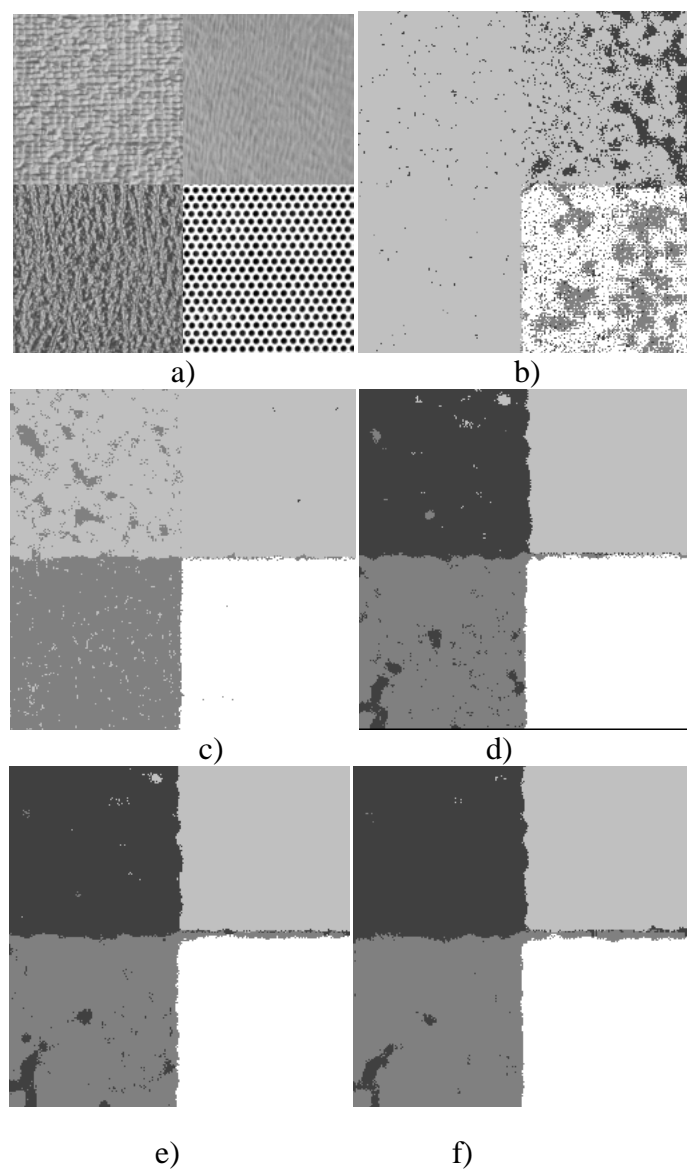


Figure.III.17 : Influence de la taille de la matrice de cooccurrence sur la segmentation de l'image syntetique3

a) Image originale b) $V \times V = 3 \times 3$ c) $V \times V = 5 \times 5$ d) $V \times V = 9 \times 9$ e) $V \times V = 11 \times 11$
f) $V \times V = 13 \times 13$

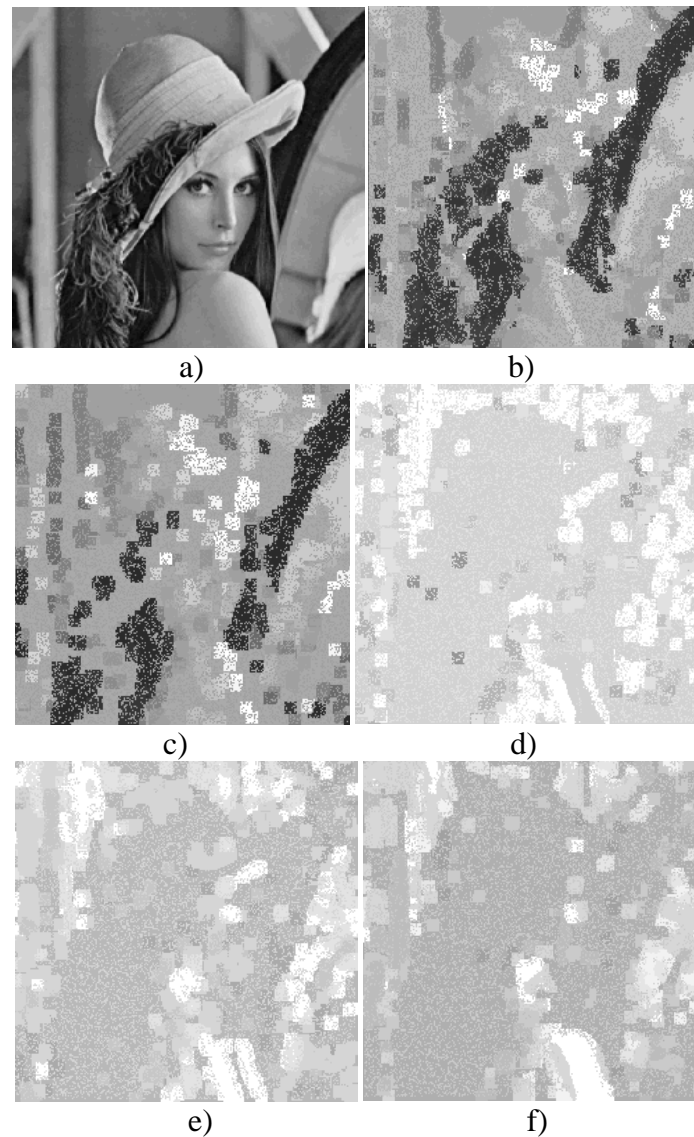


Figure.III.18 : Influence de la taille de la matrice de cooccurrence sur la segmentation de l'image LENA
 a) Image originale b) $V \times V = 3 \times 3$ c) $V \times V = 5 \times 5$ d) $V \times V = 9 \times 9$ e) $V \times V = 11 \times 11$
 f) $V \times V = 13 \times 13$

En observant les résultats affichés sur les Figures précédentes, nous constatons que pour une faible taille de la fenêtre utilisée pour le calcul de la matrice de cooccurrence, le résultat est mauvais car les différentes textures se confondent.

Il est prévisible de trouver des améliorations pour les images synthétiques en augmentant la taille de cette fenêtre car pour avoir de bons résultats, il faut ouvrir des fenêtres qui contiennent au moins un motif de base de la texture étudiée, ce qui est nettement observée sur les figures III.16, III.17.

Nous ne pouvons pas augmenter cette taille indéfiniment surtout pour des images réelles (voir figure III.18) car nous risquons de considérer plusieurs textures différentes dans une même fenêtre.

III.9 Conclusion

Dans ce chapitre, nous avons exposé la méthode proposée et fournie des résultats expérimentaux sur des images texturées réelles et synthétiques. Nous avons constaté que ces résultats sont satisfaisants puisque la réduction du volume de données à traiter fournit malgré cela une bonne segmentation. Nous avons aussi testé dans ce chapitre l'influence des paramètres ajustables de la méthode.



***CONCLUSION
GENERALE***

Conclusion générale

Nous avons proposé dans ce travail une méthode pour accélérer la segmentation des images en ne classant au préalable que quelques pixels de ces images. En effet la segmentation par classification en utilisant les paramètres de texture, de forme,... etc peut prendre un temps assez important, dans ce mémoire, nous pouvons présenter une méthode qui le réduit. Cette méthode opère en deux étapes.

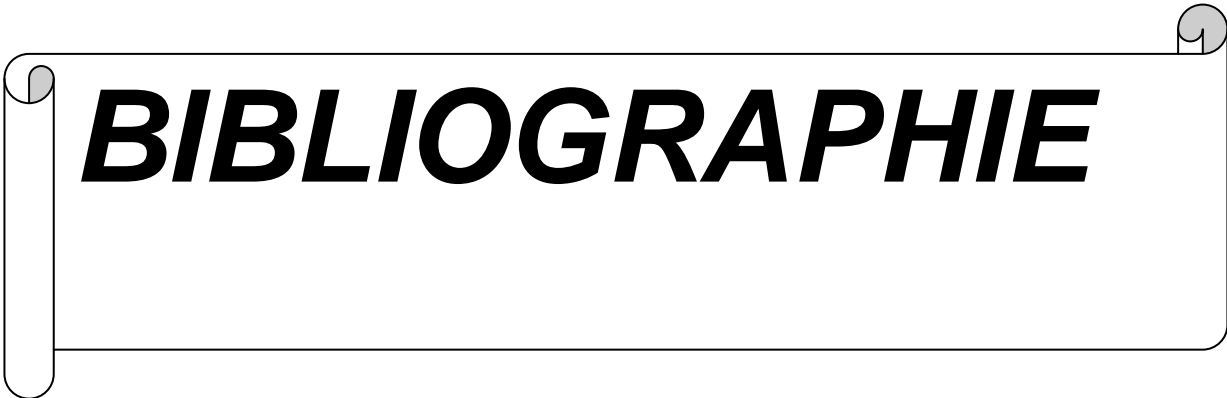
Dans la première étape, une sélection aléatoire d'un certain taux de pixels de l'image est réalisée, leurs attributs de texture en utilisant la matrice de cooccurrence sont ensuite calculés puis une classification par la méthode des k-means est effectuée.

Dans la deuxième étape, les pixels non sélectionnés sont classés en considérant les classes des pixels sélectionnés dans la première étape et se trouvant dans leur voisinage.

L'utilisation de cette méthode a nécessité l'ajustement de certains paramètres tels que, le taux de pixels sélectionnés, la taille de la fenêtre de voisinage pour la classification des pixels non sélectionnés en plus des paramètres pour le calcul de la matrice de cooccurrence.

Principalement nous avons constaté que pour toutes les images testées, un taux de pixels sélectionné de 30% est largement suffisant pour obtenir des résultats très proches du cas où on considère 100% des pixels. Ainsi nous pouvons donc réduire le temps de calcul et le volume des données à traiter de 70% ce qui n'est pas négligeable lorsque nous travaillons sur des images de grande dimension ou/et en utilisant des méthodes de classification qui prennent beaucoup de temps.

La technique proposée n'est pas restreinte à la méthode de classification par k-means ou à l'emploi des paramètres de texture extraits par la matrice de cooccurrence mais elle peut être utilisée avec d'autres méthodes de classification et d'autres paramètres décrivant le pixel.



BIBLIOGRAPHIE

- [1]-ALZENNYR GOMES DA SILVA, analyse des données évolutives : application aux données d'usage du Web, Thèse de doctorat d'état université Paris IX Dauphine école Doctorale Décision, Informatique, Mathématiques et Organisation (EDDIMO), 2009.
- [2]-AMAZIT DALILA, Segmentation d'image texturées par approche statistique, thèse de magister, UMMTO, 2005.
- [3]-BELISSON Andy Anderson, LE VIGOUROUX Swann, M1 SIS Luminy, http://en.wikipedia.org/wiki/Texture_synthesis 2007-2008.
- [4]-BELKACEM ET R BOUBRIT, Segmentation d'images par analyse de la texture, Mémoire d'ingénieur d'état, UMMTO, 1998.
- [5]-Mr BENANMROUZ.S, KETTANE.S, segmentation d'images par méthode adaptative basée sur les matrices de cooccurrences, mémoire d'ingénieur (eln,aut),UMMTO,2009.
- [6]-GILLE BUREL, Introduction en traitement d'image, Lavoisier 2001.
- [7]-R.M HARALICK,L.G Shapiro, image segmentation technique ,CVGIP.29(1) :pp.100-132.1985.
- [8]-JEAN PIERRE COCQUEREZ et Sylvie Philippe, Analyse d'image : Filtrage et segmentation, Masson, 1995.
- [9]-JEROM LANDRÉ, analyse multi résolution pour la recherche l'indexation d'images par le contenu dans les bases images application à la base d'image paléontologique trans'tyfiral, université de bourgogne, U.F.R. sciences et technique école doctorale buffon images et modélisation des objets naturels, 2006.
- [10]- M.DEHBLI.R, CHERIFI.F, Segmentation d'images texturées par les statistiques de rangs des niveaux de gris, Mémoire d'ingénieur d'état UMMTO, 1999.

Le résumé

Nous proposons, dans ce travail, de réduire ce temps de calcul en réduisant le nombre de pixel à traiter tout en essayant d'avoir un résultat identique, sinon proche de celui qu'on obtiendrait si on utilise tous les pixels de l'image.

Pour ce faire, nous sélectionnons de manière aléatoire un certain nombre de pixels de l'image. Nous extrayons ensuite les paramètres de texture de ces pixels, nous les classons par la suite par la méthode des k-means. Au final, les pixels qui n'ont pas été sélectionnés, donc qui ne sont pas encore classés, nous leur attribueront la classe majoritaire des pixels déjà classés se trouvant dans leur voisinage.

Notre mémoire est organisé en trois chapitres. Le premier chapitre est consacré aux notions de base concernant l'image, les différentes méthodes de segmentation, les principales méthodes d'analyse de textures et la définition des attributs statistiques de texture. Dans le second chapitre nous expliquons les principales méthodes de classification et nous nous attardons sur la méthode de classification non supervisée que nous avons utilisée à savoir la méthode des k-means. Dans le dernier chapitre nous expliquons la méthode proposée avant de l'appliquer sur des images texturées synthétiques et réelles. Nous terminons par une conclusion générale.