

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la A Recherche Scientifique

Université Mouloud Mammeri De Tizi-Ouzou



Faculté De Génie Electrique Et D'informatique
DEPARTEMENT D'AUTOMATIQUE

**Mémoire de Fin d'Etude
de MASTER ACADEMIQUE**
Spécialité : Commande des systèmes

Présenté par
Nacer HAMOU
Farid AIT ABDELLAH

Mémoire dirigée par **Sadia ALKAMA**

Thème

**Proposition des paramètres de texture
pour la segmentation d'images**

Mémoire soutenu publiquement le 04/07/2017 devant le jury composé de :

M Ahmed Dirami
M.C.B, UMMTO, Président

Mme Sadia Alkama
M.C.A, UMMTO, Rapporteur

Melle Ouardia Chilali
M.A.A, UMMTO, Examineur

Mme Ouiza Adjemout
M.A.A, UMMTO, Examineur

Remerciements

Une pensée pieuse à dieu qui a éclairé notre chemin et mener vers la réalisation de ce modeste travail.

Nous tenons à remercier notre promotrice Mme ALKAMA.S pour son aide, le temps qu'elle nous a consacré et surtout pour sa patience tout au long de ce travail.

Nos remerciements vont également aux membres de jury qui nous feront l'honneur de juger ce travail.

En fin Nous remercions toute les personnes qui nous ont soutenu et encouragé pour aller au bout de ce travail, en particulier nos familles et nos amis proches.

Dédicaces

Je dédie ce travail à :

*Mes très chers parents pour leur amour et pour le courage
et la volonté qu'ils m'ont inculqué ;*

A mes très chers frères ;

A mes très chères sœurs et leurs maries ;

A mon grand-père ;

A ma grand-mère ;

A ma nièce AYA et ces parents ;

A ma grande famille ;

A mon cher binôme Farid ;

A tous mes amis et collègues sans exception ;

Nacer

Dédicaces

Je dédie ce modeste travail à :

*Mes très chers parents pour leur amour
et pour le courage et la volonté qu'ils
m'ont inculqué ;*

A mes très chères sœurs et frères ;

A ma très chère amie Lylia et sa famille ;

*A tous mes amis(es) en particulier : Mon
binôme Nacer, Rafik, Rezek, Makhlouf ...*

Farid

Sommaire

Introduction générale	1
Chapitre I :analyse de textures	3
I. Introduction	3
II. Définition de la texture	3
III. Types de texture	4
III.1.Texture structurale	4
III.2. Texture aléatoire	5
III.3. Texture directionnelle	5
IV. Analyse de la texture	6
IV.1. Méthodes structurales	6
IV.2. Méthodes statistiques	7
V. Attributs de textures	7
V.1. Attributs géométriques	7
V.2. Attributs basés sur la modélisation spatiale	8
V.3. Attributs patio-fréquentiels	8
V.4. Attributs statistiques	8
V.4.1 Attributs statistiques du premier ordre	9
V.4.2. Attributs statistiques du econd ordre	10
V.4.3. Attributs statistiques d'ordre supérieur	13
VI. Domaine d'application de l'analyse de la texture	16
VII. Conclusion	16
Chapitre II : segmentation d'images	18
I. Introduction	18
II. Définition	18

Sommaire

IV.Approches de segmentation d'images.....	19
III.1. Approches contour	20
III.1.1. Opérateur gradient.....	21
III.1.2.Opérateur Laplacien	21
III.1.3. Opérateur de Prewitt.....	22
III.1.4. Opérateur de Sobel	22
III.2.Approches région	22
III.2.1.Segmentation par division de région.....	22
III.2.2. Segmentation par croissance de région	23
III.2.3.Segmentation par division-fusion de région	24
III.3. Classification.....	25
III.3.1.Classification supervisée (semi-automatique).....	26
III.3.2.Classification non supervisée (automatique)	28
IV. L'algorithme de classification K-means.....	28
V.Conclusion	32
Chapitre III :Méthodes adoptées, tests et résultats	33
I. Introduction	33
II.Proposition des paramètres de texture	33
II.1.Frontière des passages par zéro	36
II.2.Nombre de régions.....	39
II.3.Paramètre basé sur l'analyse cepstrale	40
III. Test des paramètres de texture	41
III.1. Résultats de la segmentation en utilisant le paramètre MC (maximum du cepstre).....	43
III.2.Résultats de la segmentation en utilisant le paramètre FPZ (frontières de passage par zéro)	45

Sommaire

III.3. Résultats de la segmentation en utilisant le paramètre NR (nombre de régions). 46	46
III.4. Résultats de la segmentation en utilisant les deux paramètres FPZ et NR 47	47
III.5. Résultats de la segmentation en utilisant l'ensemble des paramètres 48	48
IV. Conclusion 50	50
Conclusion générale 51	51
Bibliographie 52	52

Le traitement d'images est une discipline de l'informatique et de mathématiques appliquées qui étudie les images numériques et leurs transformations, dans le but d'améliorer leurs qualités ou d'en extraire de l'information.

Il s'agit d'un sous-ensemble du traitement du signal dédié aux images et aux données dérivées comme la vidéo, tout en opérant dans le domaine numérique.

Dans un système de traitement d'image, la segmentation est une étape primordiale puisqu'elle conditionne la qualité de l'interprétation d'une image. Un bon résultat de segmentation ne permet pas forcément une bonne interprétation, mais nous ne pouvons pas obtenir une bonne interprétation à partir d'un mauvais résultat de segmentation.

Les images contiennent la plus part des temps des zones texturées. Ainsi, de nombreuses méthodes de segmentation liées à la notion de texture sont apparues. Les méthodes de segmentation d'images texturées se basent d'abord sur l'analyse des textures pour extraire les différents attributs qui les décrivent.

L'analyse des textures joue un rôle important, dans l'identification et l'extraction des informations thématiques contenues dans l'image. Des exemples peuvent être trouvés dans des domaines variés, citons l'exemple de l'imagerie médicales et biomédicales, le contrôle de qualité, la télédétection, l'astronomie, etc.

L'étude des propriétés d'une image texturée, a généralement pour but de cerner la nature propre des textures contenues dans l'image, soit via des algorithmes de classification, soit via des algorithmes de synthèses.

Dans le domaine du traitement des images texturées, la segmentation est souvent réalisée en deux étapes, la première consiste à extraire les paramètres de texture, la deuxième consiste à soumettre ces paramètres à un classifieur. C'est cette démarche que nous adoptons dans ce travail. Nous proposons de définir des paramètres de texture inspirés des signaux monodimensionnels de la parole que nous adapterons ensuite aux images qui sont des signaux bidimensionnels.

Introduction générale

Par la suite nous soumettons ces paramètres à l'algorithme de classification K-means qui est largement utilisé dans le domaine du traitement d'images. Pour se rendre compte de l'efficacité de ces paramètres, nous comparons avec les résultats de la segmentation en utilisant les paramètres de la matrice de cooccurrence.

Compte tenu de cet objectif, nous avons opté pour le plan de travail suivant :

Le premier chapitre aborde les notions fondamentales de la texture et présente ses différentes méthodes d'analyse.

Le second chapitre est consacré à la segmentation d'images. Après la définition, nous donnons les principales méthodes de segmentation et développons la méthode de classification K-means car nous l'utilisons dans nos tests.

Le dernier chapitre est consacré à la méthode proposée. Après le développement des différentes étapes, nous présentons les résultats obtenus et nous les comparons avec ceux générés en utilisant les paramètres extraits de la matrice de cooccurrence.

Nous terminons ce mémoire par une conclusion générale.

CHAPITRE I

I. Introduction

L'analyse de la texture joue un rôle très important en traitement d'image car la texture est l'une des informations principales dans l'interprétation d'un message visuel pour un observateur humain.

L'objet de ce chapitre est de donner les notions fondamentales de la texture et son analyse.

Nous donnons une définition générale et quelques concepts de base de la texture, et présentons les différentes méthodes de l'analyse de la texture. Nous mettrons en évidence les principales méthodes d'extraction des paramètres permettant d'identifier les textures. L'accent sera mis sur l'utilisation des matrices de cooccurrence car elles ont une bonne caractérisation de la notion de texture et servira pour la comparaison avec la technique que nous allons adopter.

II. Définition de la texture

La texture est une caractéristique facile à reconnaître mais, en même temps, très difficile à définir. Sa définition littéraire est la suivante: "répétition spatiale d'un même motif dans différentes directions de l'espace". Cette définition est limitative car elle caractérise l'objet indépendamment d'un observateur humain. Cependant, une texture est un attribut représentant la distribution spatiale des niveaux des pixels d'une région. Elle se traduit aussi par un arrangement spatial des pixels que le niveau, seul, ne suffit pas à décrire [1].

La texture est définie aussi comme étant une région d'une image pour laquelle il est possible de définir une fenêtre de dimension minimale, tel qu'une observation au travers de celle-ci, à une perception (impression) visuelle identique pour toutes les translations possibles de cette fenêtre à l'intérieur de la région considérée.

Haralick la [2] décrit comme un phénomène à deux dimensions : la première concerne la description d'éléments de base ou primitives (le motif) à partir desquelles

est formée la texture, le deuxième est relatif à la description de l'organisation spatiale de ces primitives.

III. Types de texture

La complexité de la notion texture s'impose aussi dans la classification de l'ensemble des textures. En effet, il existe une multitude de textures qu'on peut regrouper en trois principales familles

- Texture structurelles
- Texture aléatoires
- Texture directionnelles

III.1. Texture structurelle

Les textures structurelles peuvent être considérées comme étant la répartition spatiale de motifs élémentaires de base dans différentes directions de l'espace suivant une certaine règle d'agencement

La Figure I.1 présente un exemple de deux images de textures structurelles. En effet, on s'aperçoit que l'image de droite, représente un mur de brique, elle est composée d'un ensemble d'éléments de base (élément structurant, les briques) disposées régulièrement de manière horizontale. L'image de gauche, est aussi composée de motifs de base agencés d'une manière particulière les uns à côté des autres.

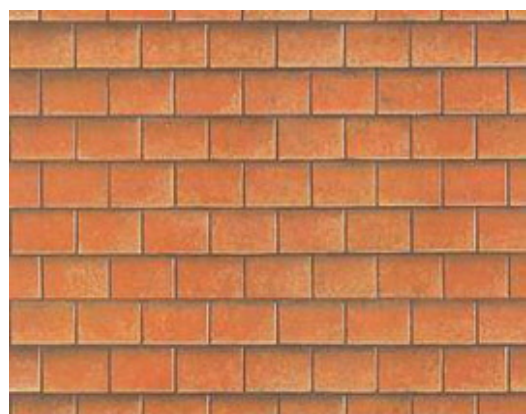
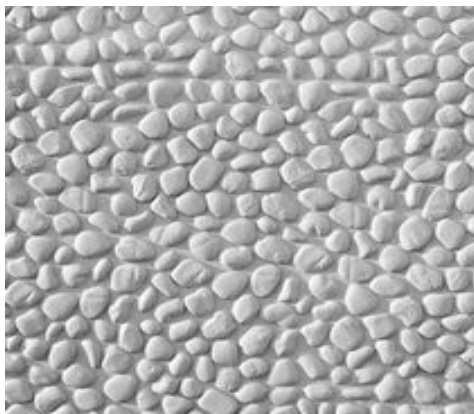


Figure I.1 : Exemples de textures structurelles

III.2. Texture aléatoire

Contrairement aux textures structurales, ce type de texture a un aspect anarchique, tout en restant globalement homogènes. On ne peut pas en extraire le motif de base qui se répète spatialement. Pour ce genre de texture, l'image est considérée comme étant associée à un processus aléatoire bidimensionnel.

La Figure I.2 présente un exemple de deux images de textures aléatoires.

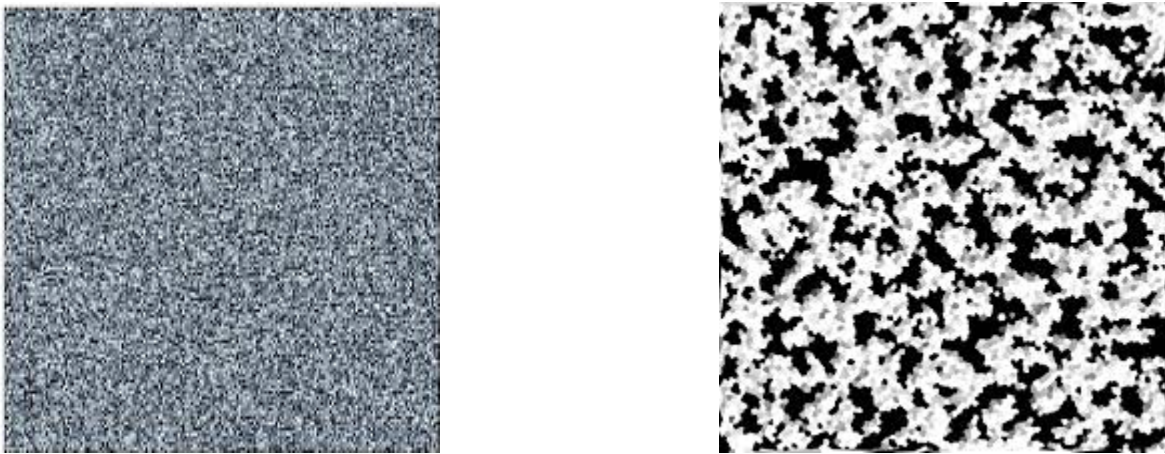


Figure I.2 : Exemples de textures aléatoires

III.3. Texture directionnelle

Les textures directionnelles ne sont pas totalement aléatoires et ne présentent pas d'éléments structurants de base. Elles se caractérisent principalement par certaines orientations privilégiées, qui forment des motifs organisés selon des directions bien précises.

Deux exemples de textures directionnelles sont donnés sur la Figure I.3



Figure I.3 : Exemples de textures directionnelles

IV. Analyse de la texture

L'objectif principal de l'analyse de la texture est la détermination de signatures texturales permettant une meilleure caractérisation de l'image, autrement dit, c'est la formalisation des descriptifs de la texture par des paramètres mathématiques qui serviraient à l'identifier [3].

L'analyse de texture est utilisée dans des domaines de plus en plus variés. La caractérisation, la segmentation des images ou la reconnaissance des formes représentent ses plus importantes applications.

Le choix d'une méthode de caractérisation de textures est étroitement lié à l'application visée et à la nature de la texture à discriminer. Parmi les méthodes utilisées dans l'analyse de la texture, nous pouvons citer les méthodes structurelles et les méthodes statistiques.

IV.1. Méthodes structurelles

Ces méthodes tiennent compte de l'information structurelle et contextuelle d'une forme et sont particulièrement bien adaptées aux textures macroscopiques. Les étapes d'analyse sont d'abord l'identification des éléments constitutifs, puis la définition des règles de placement. Les deux structures les plus importantes sont les structures de graphe et les structures syntaxiques.

IV.2. Méthodes statistiques

La caractérisation par ces méthodes, repose sur l'étude des variations locales d'intensité au sein de l'image et sur la description statistique de l'arrangement des pixels. Elles étudient les relations entre un pixel et ses voisins, dans le but d'extraire des paramètres texturaux.

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 pour les textures aléatoires.

V. Attributs de textures

Un attribut peut être défini comme étant une donnée de type numérique décrivant le contenu d'une image.

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.

De ce fait, on peut distinguer quatre classes connues d'attributs de texture : les attributs géométriques, les attributs basés sur la modélisation spatiale des textures, les attributs spatio-fréquentiels et les attributs statistiques.

V.1. Attributs géométriques

La description dite géométrique d'une texture tient compte de l'information structurelle et contextuelle de l'image. Elle est particulièrement bien adaptée à la description des textures lorsque celles-ci sont observées à un niveau macroscopique.

En effet, la description de la texture est faite par :

- une extraction explicite des primitives (régions, contours, motifs élémentaires, lors d'un prétraitement)
- des règles de placement de ces primitives [4].

V.2. Attributs basés sur la modélisation spatiale

Ces attributs sont issus de différentes modélisations de la répartition spatiale des niveaux de gris dans l'image [4].

Les principaux modèles utilisés sont :

- les modèles autorégressifs
- les modèles de Markov
- les motifs locaux binaires

V.3. Attributs spatio-fréquentiels

Les attributs spatio-fréquentiels se divisent en trois catégories: les attributs qui sont calculés dans le domaine spatial, ceux calculés dans le domaine fréquentiel ou bien ceux extraits à partir du domaine spatio-fréquentiel.

Le calcul d'attributs de texture dans le domaine spatial consiste à caractériser la texture par le nombre de transitions par unité de surface.

Le calcul d'attributs de texture dans le domaine fréquentiel consiste à effectuer une transformation du domaine spatial au domaine fréquentiel.

Le troisième type d'attributs de texture associe les deux domaines spatial et fréquentiel.

V.4. Attributs statistiques

Les attributs statistiques permettent de caractériser tout type de textures, même les textures fines et sans régularité apparente. Pour ce type d'attributs, la texture est définie en termes de variation de niveau de gris et ce sont également les relations entre les pixels et leurs voisins qui sont étudiées.

Ce type d'attributs peut être divisé 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, ce sont les couples de pixels qui sont considérés, ce sont des attributs d'ordre deux.

Les attributs statistiques peuvent être divisés donc en trois ensembles à savoir les attributs statistiques de premier ordre, les attributs statistiques de deuxième ordre et les attributs statistiques d'ordre supérieur.

V.4.1. Attributs statistiques du premier ordre

Pour les attributs statistiques de premier ordre on n'exploite que l'information de niveau de gris des pixels, en négligeant les relations spatiales entre ces derniers [5]. Les paramètres d'étude les plus souvent utilisés sont la moyenne, la variance, l'entropie, le rapport signal-bruit etc... Ces derniers sont calculés en utilisant les propriétés de l'histogramme des niveaux de gris de l'image.

- La moyenne donne le niveau de gris moyen dans une fenêtre de l'image.

Elle est calculée par l'équation suivante :

$$m = \frac{1}{N} \sum_{i,j} I(i,j) \quad (\text{I, 1})$$

Tel que :

$I(i, j)$ est la valeur de niveau de gris du pixel (i, j) .

N correspond au nombre total des pixels dans la fenêtre.

- La variance, donne des informations sur la dispersion et la variabilité des valeurs de niveau de gris de l'image.

$$\text{VAR} = \frac{1}{N} \sum_{i,j} (I(i,j) - m)^2 \quad (\text{I, 2})$$

- Le rapport signal-bruit rend compte de l'hétérogénéité d'une texture.

$$\text{SNR} = \frac{m}{\text{VAR}} \quad (\text{I. 3})$$

- La dissymétrie caractérise la forme du sommet de l'histogramme des niveaux de gris de l'image. Plus celle-ci est faible, plus le sommet est arrondi.

$$\text{KURT} = \frac{1}{N} \sum_{i,j} (I(i,j) - m)^4 \quad (\text{I. 4})$$

V.4.2. Attributs statistiques du second ordre

Ce type d'attributs contient des informations sur la localisation des pixels. Les méthodes qui les caractérisent sont celles qui exploitent les paires des pixels en incluant une certaine dépendance spatiale [1].

Parmi ces méthodes nous développons celle de la matrice de cooccurrence car nous l'utilisons dans notre travail.

Une matrice de cooccurrence mesure la probabilité d'apparition des paires de valeurs de pixels situés à une certaine distance dans l'image. Elle est basée sur le calcul de la probabilité $P(i, j, d, \theta)$ qui représente le nombre de fois où un pixel de niveau de gris i apparaît à une distance relative d d'un pixel de niveau de gris j et selon une orientation θ donnée [6].

Les directions angulaires θ souvent utilisées sont 0, 45, 90 et 135 degrés.

La Figure I.4 montre un exemple de calcul des $P(i, j, d, \theta)$ à partir d'une petite image 4×4 composée de quatre niveaux de gris (0, 1, 2, 3). Dans cet exemple on se limite de citer deux cas : $d = 1$; $\theta = 0^\circ$ et $\theta = 90^\circ$.

L'élément (1,0) de la matrice $P(i, j, 1, 0^\circ)$ est égal à 3, cela signifie qu'il existe 3 configurations dans l'image où un pixel de niveau de gris 1 est séparé horizontalement d'un autre pixel de niveau de gris 0 par une distance 1. Ces configurations sont représentées en trait horizontal sur l'image.

L'élément (1,3) de la matrice $P(i, j, 1, 90^\circ)$ est égal à 3, cela signifie qu'il existe 3 configurations dans l'image où un pixel de niveau de gris 1 est séparé verticalement

d'un autre pixel de niveau de gris 3 par une distance 1. Ces configurations sont représentées en trait vertical sur l'image.

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

Image 4x4

$$P(i, j, 1, 0^\circ) = \begin{pmatrix} 0 & 3 & 2 & 1 \\ 3 & 0 & 1 & 3 \\ 2 & 1 & 0 & 2 \\ 1 & 3 & 2 & 0 \end{pmatrix}; \quad P(i, j, 1, 90^\circ) = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 3 \\ 1 & 1 & 0 & 3 \\ 0 & 3 & 3 & 1 \end{pmatrix}$$

Figure I.4 : Exemple de matrices de cooccurrence construites à partir d'une image 4×4 composée de 4 niveaux de gris.

La plupart des images sont codées sur 256 niveaux de gris, par conséquent, la taille des matrices de cooccurrence est de 256×256 . On s'aperçoit ainsi que ces matrices comptabilisent une très grosse quantité d'informations difficile à exploiter directement.

Afin d'extraire de l'information de ces matrices pour mettre en évidence la structure des textures. Haralick et al [2] ont proposé quatorze paramètres, caractérisant les textures, issus de ces matrices. Voici 6 paramètres considérés comme étant les plus utilisés et les plus pertinents :

- L'énergie :

$$E = \sum_i \sum_j (P_{ij}(d, \theta))^2 \quad (I)$$

5)

Ce paramètre mesure l'uniformité de la texture. Il atteint de fortes valeurs lorsque la distribution des niveaux de gris est constante ou de forme périodique. Dans

ce dernier cas, les valeurs élevées d'énergie sont obtenues pour les matrices $P(d, \theta)$ lorsque (d, θ) correspond à la période

- Le contraste :

$$\text{CST} = \sum_i \sum_j ((i - j)^2 P_{ij}(d, \theta)) \quad (\text{I, 6})$$

Sa valeur est d'autant plus élevée que la texture présente des passages d'un pixel très clair à un autre très foncé dans l'image (fort contraste). Ce paramètre est fortement non corrélé à l'énergie.

- L'entropie :

$$\text{ENT} = - \sum_i \sum_j (\log P_{ij}(d, \theta) P_{ij}(d, \theta)) \quad (\text{I, 8})$$

Ce paramètre mesure le désordre dans l'image. Contrairement à l'énergie, l'entropie atteint de fortes valeurs lorsque la texture est complètement aléatoire (sans structure apparente). Elle est fortement corrélée (par l'inverse) à l'énergie.

- La variance :

$$\text{VAR} = \sum_i \sum_j ((i - \mu)^2 P_{ij}(d, \theta)) \quad (\text{I, 9})$$

Avec μ la moyenne de la matrice.

La variance mesure l'hétérogénéité de la texture. Elle augmente lorsque les niveaux de gris différent de leur moyenne. La variance est indépendante du contraste.

- La corrélation :

$$\text{COR} = \sum_i \sum_j \left(\frac{(i - \mu)(j - \mu) P_{ij}(d, \theta)}{\sigma^2} \right) \quad (\text{I, 10})$$

Où μ représente la moyenne de la matrice et σ représente l'écart type de la matrice.

Ce paramètre décrit les corrélations entre les lignes et les colonnes de la matrice de cooccurrence.

Il mesure la dépendance linéaire relative au déplacement "d" des niveaux de gris dans l'image. Ce paramètre n'est corrélé ni à l'énergie, ni à l'entropie.

- le moment de différence inverse :

$$\text{IDM} = \sum_i \sum_j \frac{P_{ij}(d,\theta)}{1+(i-j)^2} \quad (\text{I, 11})$$

Ce paramètre mesure l'homogénéité de l'image. Il reflète l'existence de plage uniforme de texture. Plus cette valeur est élevée, plus la texture est grossière.

V.4. 3. Attributs statistiques d'ordre supérieur

L'extraction de ces attributs repose sur l'étude des interactions entre plusieurs pixels de l'image, ils donnent des caractéristiques plus riches par rapport aux attributs de second ordre.

La méthode la plus connue est celle de la matrice de longueur de plages. Une plage est un ensemble connexe et linéaire de pixels ayant le même niveau de gris.

Cette méthode consiste donc, au premier lieu à déterminer l'ensemble des plages présentes dans l'image et en second lieu, d'extraire des statistiques sur leur longueur. On définit ainsi plusieurs matrices correspondant chacune à une orientation θ des plages dans l'image.

A chaque direction θ , on peut associer une matrice de longueurs de plages R_θ telle qu'un élément $R_\theta(i, l)$ de la matrice soit égal au nombre de plages de niveau de gris i et de longueur l dans la direction θ .

Comme les matrices de cooccurrence, les orientations souvent utilisées pour décrire les textures sont : 0° , 45° , 90° et 135° . La Figure I.5 présente un exemple simple de calcul de la matrice de longueur de plage obtenue pour $\theta = 0^\circ$. Il s'agit donc de rechercher le nombre maximum de pixels adjacents ayant le même niveau de gris dans cette direction.

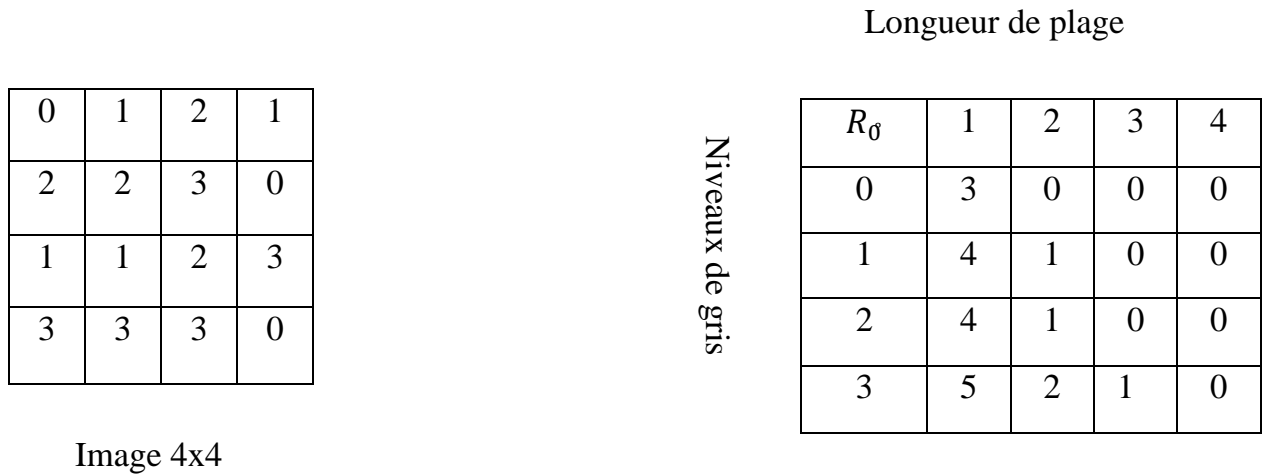


Figure I.5 : Exemple montrant la formation d'une matrice de longueur de plage, à partir d'une image 4x4

Pour une image I ayant ng niveaux de gris, soit nl le nombre de longueurs de plage identifiées dans l'image. La matrice de longueur de plage R_θ , dans la direction θ , est la matrice composée de ng lignes et nl colonnes telle que l'élément $R_\theta(i, l)$ de la matrice correspond au nombre de plages de niveau de gris i, et de longueur l, dans la direction θ .

Parmi les paramètres qu'on peut extraire de ces matrices, nous pouvons citer :

Le Poids des plages courtes (SRE) :

$$SRE(\theta) = \frac{\sum_{i=1}^{ng} \sum_{l=1}^{nl} \frac{R_\theta(i,l)}{l^2}}{\sum_{i=1}^{ng} \sum_{l=1}^{nl} R_\theta(i,l)} \quad (I, 12)$$

Ce paramètre caractérise la finesse de la texture. Plus SRE(θ) est grand, plus les régions ayant le même niveau de gris i sont petites.

- Le Poids des plages longues (LRE) :

$$LRE(\theta) = \frac{\sum_{i=1}^{ng} \sum_{l=1}^{nl} l^2 \times R_\theta(i,l)}{\sum_{i=1}^{ng} \sum_{l=1}^{nl} R_\theta(i,l)} \quad (I, 13)$$

Plus il est grand, plus il y a plus de régions étendues ayant le même niveau de gris i , et donc plus la texture est grossière.

- hétérogénéité des niveaux de gris (HNG) :

$$\text{HNG}(\theta) = \frac{\sum_{i=1}^{\text{ng}} \left(\sum_{l=1}^{\text{nl}} R_{\theta}(i,l) \right)^2}{\sum_{i=1}^{\text{ng}} \sum_{l=1}^{\text{nl}} R_{\theta}(i,l)} \quad (\text{I, 14})$$

Ce paramètre mesure l'uniformité de la distribution des plages. $\text{HNG}(\theta)$ est minimal lorsque les plages sont uniformément distribuées entre les différents niveaux de gris i .

- hétérogénéité des longueurs de plages (HLP) :

$$\text{HLP}(\theta) = \frac{\sum_{l=1}^{\text{nl}} \left(\sum_{i=1}^{\text{ng}} R_{\theta}(i,l) \right)^2}{\sum_{i=1}^{\text{ng}} \sum_{l=1}^{\text{nl}} R_{\theta}(i,l)} \quad (\text{I, 15})$$

Plus cet attribut est grand, plus le nombre de plages de même longueur est important.

- pourcentage de plage (PP) :

$$\text{PP}(\theta) = \frac{\sum_{i=1}^{\text{ng}} \sum_{l=1}^{\text{nl}} R_{\theta}(i,l)}{N \times M} \quad (\text{I, 14})$$

Ce paramètre représente le rapport entre le nombre total de plages et le nombre de pixels de l'image. Plus $\text{PP}(\theta)$ est important, plus la texture est homogène. $N \times M$ est le nombre de pixels de l'image.

Finalement, il est intéressant de noter que cette méthode permet une bonne caractérisation de texture des régions. Cependant, elle reste relativement lente, et est efficace sur des images comportant peu de niveaux de gris [3].

VI. Domaine d'application de l'analyse de la texture

L'analyse de la texture est utilisée dans des domaines de plus en plus variés. La caractérisation, la segmentation des images ou la reconnaissance des formes représentent ses plus importantes applications.

L'analyse de la texture est souvent employée comme un moyen pour finaliser la segmentation d'une image. En effet, dans le cas des images naturelles, la seule étude de la distribution des niveaux de gris est insuffisante pour caractériser les zones homogènes. C'est le cas pour les images multi spectrales réalisées en télédétection ou les estimateurs de textures permettent de différencier un champ, d'une forêt, d'une ville..... C'est le cas aussi en imagerie médicale ou ces mêmes estimateurs permettent de faire la distinction entre tissus sains et tissus pathologiques (détection de lésions, dépistage de pathologies, segmentation en IRM....).

En compression d'images, les attributs texturaux permettent de représenter l'image par un nombre minimal de paramètres et de ce fait permettent une reconstitution de l'information avec un minimum d'erreur. La caractérisation de texture à l'aide de paramètres pertinents permet également la restauration d'une partie dégradée ou manquante dans une image en la remplaçant par une version synthétique générée à partir du modèle textural élaboré. Dans le domaine de l'infographie ou de l'audiovisuel, la synthèse de texture découle naturellement de l'analyse et conduit à son utilisation pour le réalisme, l'art, le design....

En contrôle non destructif, l'analyse de texture est largement utilisée pour l'inspection des surfaces en contrôle de qualité (produits industriels, matériaux, produits alimentaires, détection de défauts...).

VII. Conclusion

Dans ce chapitre, nous avons présenté quelques notions sur la texture et l'analyse de texture en citant ses différentes méthodes d'analyse. Nous nous sommes également intéressés à la méthode des matrices de cooccurrence, compte tenu de sa fiabilité et la performance des résultats qu'elle offre, et vu que nous l'utilisons dans le dernier chapitre.

Nous réalisant une segmentation en utilisons des paramètres de texture que nous proposons et nous comparons les résultats avec la segmentation en utilisant les paramètres extraits de la matrice de cooccurrence.

Avant de fournir ces résultats, nous développons dans le chapitre suivant la notion de segmentation.

CHAPITRE II

I. Introduction

Le processus d'analyse d'images peut être défini comme l'ensemble des méthodes et d'outils permettant de décrire quantitativement le contenu d'une image. Il est généralement décomposé en plusieurs étapes : acquisition, numérisation, prétraitement, segmentation, interprétation.

L'étape de segmentation d'images est une étape très importante dans cette chaîne d'analyse car c'est à partir de l'image segmentée que les mesures sont effectuées pour l'extraction des paramètres discriminants en vue de la classification ou de l'interprétation.

II. Définition

Segmenter une image consiste à partitionner celle-ci en un ensemble de régions connexes, présentant une homogénéité selon un certain critère choisi (couleur, texture, niveau de gris... etc.). Elle a pour but l'extraction de l'information caractéristique contenue dans l'image. Le résultat de la segmentation est une image dans laquelle chaque pixel est affecté de l'étiquette correspondante au numéro de la région à laquelle il appartient dans l'image initiale.

Mathématiquement, la segmentation est considérée comme étant une décomposition de l'image I en N sous-ensembles R_i , appelés région. L'ensemble des régions $\{R_1, R_2, \dots, R_N\}$ est une segmentation de l'image I si :

- 1) $\bigcup_{i=1}^N R_i = I, i \in [1, N]$
- 2) $R_i \cap R_j = \emptyset \forall i \neq j$
- 3) R_i est connexe, $i = 1, 2, 3, \dots, N$.
- 4) $P(R_i) = \text{vrai} \forall R_i$ avec P : prédicat d'uniformité.
- 5) $P(R_i \cup R_j) = \text{faux}, \forall i \neq j$ et R_i adjacente à R_j

La première condition implique que chaque pixel de l'image doit appartenir à une région R_i , et l'union de toutes les régions correspond à l'image entière.

La deuxième condition stipule l'intersection de deux régions distinctes est égale à l'ensemble vide (c'est-à-dire qu'un pixel ne doit pas appartenir à deux régions différentes).

La troisième condition est relative à la structure des régions, elle définit une région comme un ensemble des pixels qui doivent être connexes.

La quatrième condition exprime le fait que chaque région doit respecter un prédicat d'uniformité P.

La dernière condition implique la non réalisation de ce prédicat pour la réunion de deux régions adjacentes [7].

III. Approches de segmentation d'images

La segmentation d'image est une étape essentielle en traitement d'image, elle fait référence aux notions de différence et de similarité perçues par le système visuel humain.

Il existe une multitude de méthodes de segmentation qu'on peut regrouper en trois principales catégories qui sont l'approche contour, l'approche région et la classification des pixels. En effet le choix d'une méthode est lié à la nature de l'image et aux primitives à extraire. Les différentes approches de la segmentation sont résumées dans l'organigramme suivant :

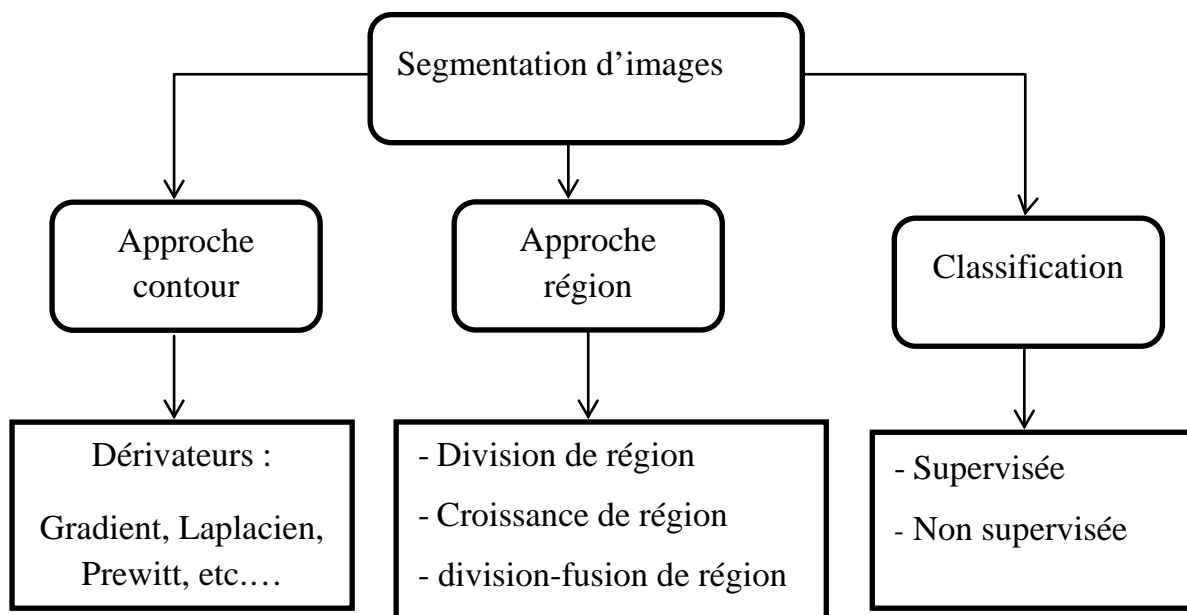


Figure II.1 : Approches de segmentation d'images

III.1. Approches contour

On définit un contour comme une frontière entre deux régions de niveaux de gris différents, il se manifeste dans une image par une forte transition des valeurs de luminance ou niveaux de gris.

L'approche contour recherche à localiser ces transitions. Elle utilise la discontinuité dans une image pour détecter les bords et les contours des régions.

Cette transition peut être détectée par le maximum d'une dérivée première de la fonction niveau de gris d'un point image ou par le passage par zéro de la seconde dérivée [7].

Les opérateurs les plus fréquemment utilisés pour détecter ces variations sont les opérateurs gradient, Laplacien, de Prewitt et de Sobel.

III.1.1. Opérateur gradient

Le gradient de l'image est considéré comme l'opérateur de base pour mesurer les contours dans une image. Il est défini dans un point de l'image par :

$$\nabla I(x, y) = \left[\frac{\partial I}{\partial x} \vec{u} + \frac{\partial I}{\partial y} \vec{v} \right] = G_x \vec{u} + G_y \vec{v} \quad (\text{II, 1})$$

Son approximation discrète est défini par :

$$\nabla I(x, y) = [I(x, y) - I(x-1, y)] \vec{u} + [I(x, y) - I(x, y-1)] \vec{v} \quad (\text{II, 2})$$

Tel que : \vec{u} et \vec{v} deux vecteurs unitaires selon les axes x et y.

Et $I(x, y)$: la fonction de niveau de gris de l'image.

Le module du gradient mesure la force de contour et définit par :

$$|\nabla I(x, y)| = \sqrt{\left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2} \quad (\text{II, 3})$$

Son orientation est :

$$\alpha = \arctg \frac{\partial I / \partial x}{\partial I / \partial y} \quad (\text{II, 4})$$

Les dérivées partielles peuvent être approximées par des différences finies suivant les directions i et j

$$|\nabla I(x, y)| = \sqrt{[I(x+1, y) - I(x-1, y)]^2 + [I(x, y) - I(x, y-1)]^2} \quad (\text{II, 5})$$

III.1.2. Opérateur Laplacien

Le laplacien est un opérateur dérivé de second ordre qui est utilisé pour extraire les contours dans une image.

Il est défini par :

$$\Delta I(x, y) = \frac{\partial^2}{\partial x^2} I(x, y) + \frac{\partial^2}{\partial y^2} I(x, y) \quad (\text{II, 6})$$

Dans l'approche de laplacien, le passage par zéro de la dérivée seconde du signal permet de mettre en évidence le contour.

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

$$L(x, y) = I(x, y) - \frac{1}{4} [I(x, y+1) + I(x, y-1) + I(x+1, y) + I(x-1, y)] \quad (\text{II, 7})$$

Cette opération de calcul de laplacien peut être appliquée à une image par l'intermédiaire d'un filtrage avec le masque 3*3 suivant :

$$\frac{1}{4} \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad (\text{II, 8})$$

III.1.3. Opérateur de Prewitt

Prewitt approxime le gradient par la convolution de l'image par deux masques l'un horizontal h et l'autre vertical v. Ces masques sont :

$$h = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} ; \quad v = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \quad (\text{II, 9})$$

III.1.4. Opérateur de Sobel

L'opérateur de Sobel est défini comme étant une combinaison entre les opérateurs de lissage et les opérateurs différentiels. Il utilise aussi un masque vertical v et un autre horizontal h.

$$h = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad v = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad (\text{II, 10})$$

III.2. Approches région

La notion de région fait référence à des regroupements de pixels ayant des propriétés communes. 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 [8]. Ces approches peuvent être classées en trois types de méthodes : division de régions, fusion de régions et division-fusion de région.

III.2.1. Segmentation par division de région

L'approche segmentation par division de régions consiste à diviser l'image originale en régions homogènes au sens d'un critère donné.

Ce processus est récursif et considère que la région initiale R correspond à l'image initiale à segmenter. Si une région ne respecte pas un prédicat d'homogénéité, elle est divisée 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 [9].

La figure suivante illustre le principe de ce processus.

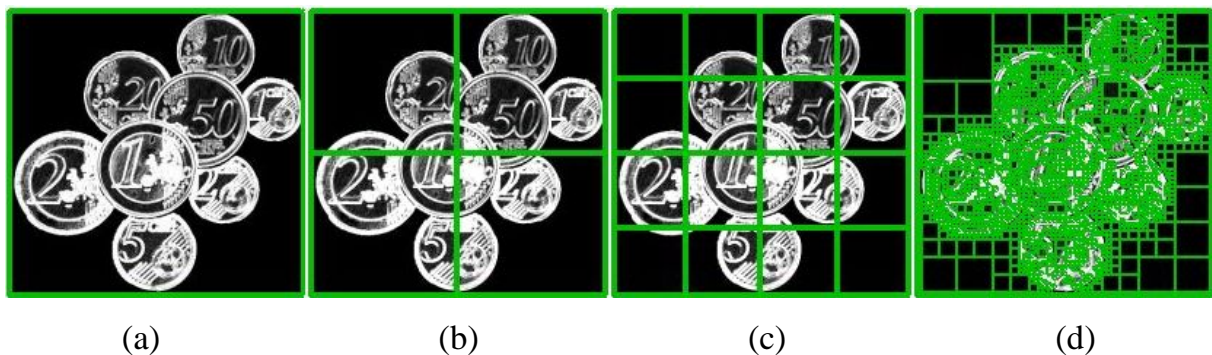


Figure II.2: Exemple de segmentation par division de région

- a) Image originale b) 1^{er} niveau de division c) 2^{ème} niveau de division
c) $n^{\text{ème}}$ niveau de division

Les régions qui contiennent des pixels qui ne sont pas identiques seront divisées en quatre.

A cause des divisions en quatre des régions, cette méthode est plutôt adaptée à des images carrées ayant un nombre de lignes et de colonnes égal à une puissance de deux, et dans lesquelles les régions sont de forme rectangulaire. D'autre part, cette méthode a tendance à faire apparaître des effets de blocs.

III.2.2. Segmentation par croissance de région

La méthode consiste à faire croître progressivement des régions distinctes à partir de pixels initiaux appelés germes, et notés g_i . Le nombre de régions ainsi obtenu est au plus égal au nombre de germes. A chaque région sont agrégés les pixels connexes qui obéissent au prédicat de segmentation.

La croissance d'une région, à partir de g_1 , s'arrête lorsqu'aucun pixel ne peut lui être agrégé. On fait croître alors une autre région autour de g_2 , etc., jusqu'à ce que tous les pixels de l'image aient été traités, en rajoutant des germes si nécessaire.

Le choix du nombre de germes, qui détermine le nombre maximum de régions segmentées, peut être guidé par l'histogramme de l'image [10].

La figure ci-dessous illustre un exemple de segmentation par croissance de région.

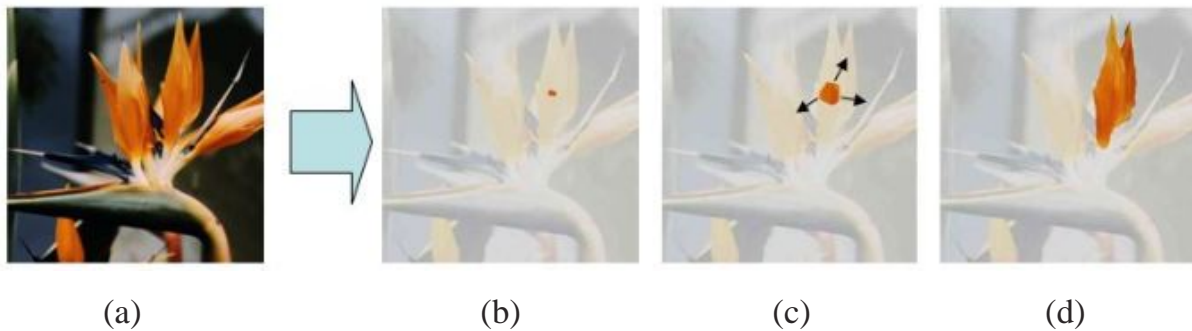


Figure II.3 : Exemple de segmentation par croissance de région

a) Image initiale ; b) germe ; c) croissance ; d) région finale

III.2.3. Segmentation par division-fusion de région

Ce processus de segmentation est itératif et alterne deux phases : une phase de division de toutes les régions non homogènes et une phase de fusion de toutes les régions adjacentes de sorte que la région résultante respecte toujours le critère d'homogénéité.

L'algorithme de division-fusion consiste, dans un premier temps, à découper l'image en zones de taille moyenne (le plus souvent en quatre quadrants), puis à parcourir chaque une des régions en cherchant si elle vérifie un critère d'homogénéité. Les régions non homogènes sont à leurs tours divisées en sous blocs. Ce processus de découpage de région est poursuivi jusqu'à obtenir que des régions homogènes.

Une étape de fusion permet par la suite de regrouper les zones adjacentes correspondant à la même région. Si deux régions adjacentes peuvent être fusionnées pour former une zone homogène, elles le sont [11].

La figure ci-dessous illustre un exemple de segmentation par division-fusion de région.

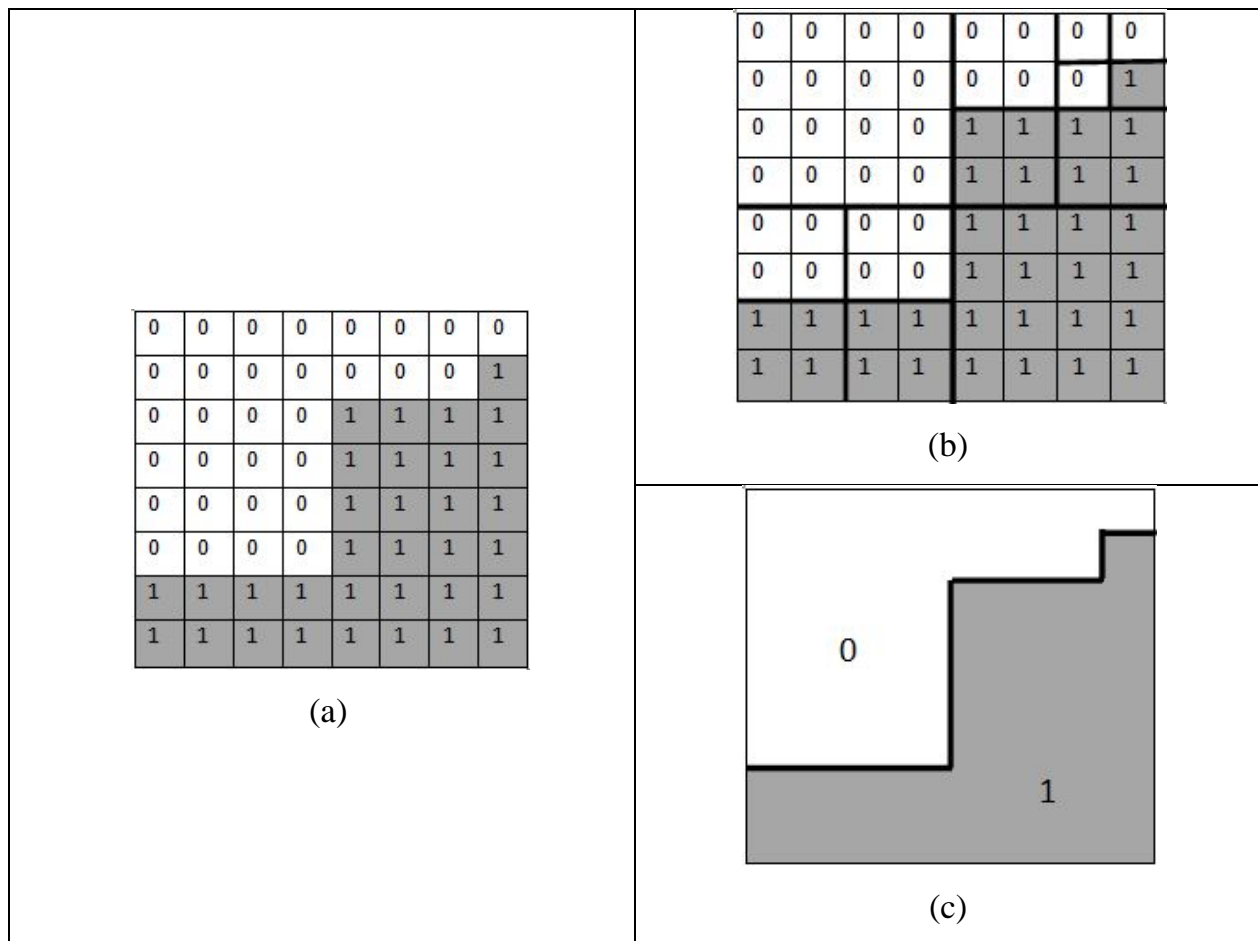


Figure II.4 : Exemple de segmentation par division-fusion de région

a) Image originale b) étape de division c) étape de fusion

III.3. Classification

Ces méthodes ont pour but de partitionner l'image en plusieurs classes. Chaque pixel est attribué à une et une seule classe. Elles consistent à associer une classe ou un label à un ensemble des pixels ayant des vecteurs d'attributs aussi similaires que possible et que les classes soient aussi éloignées (en terme d'attributs) que possible les unes des autres. Les pixels qui appartiennent à une même classe forment un nuage de points dans l'espace d'attributs. La classification consiste à retrouver ces nuages de points qui correspondent aux classes de pixels présentes dans l'image. La dimension de cet espace de représentation est égale au nombre d'attributs.

Le processus de classification est donc réalisé par l'exécution des étapes suivantes :

- Représentation des données de la classification (définition des attributs des pixels servant à la classification tel que les niveaux de gris, les paramètres de textures, de couleur, ...etc.).
- Définition d'une distance de similarité entre les pixels (par exemple, la distance euclidienne entre les niveaux de gris des pixels).
- Regroupement des pixels en des classes.
- Evaluation de la classification obtenue.

Il existe plusieurs méthodes de segmentation par classification des pixels que l'on peut regrouper en deux catégories : classification supervisée et classification non supervisée.

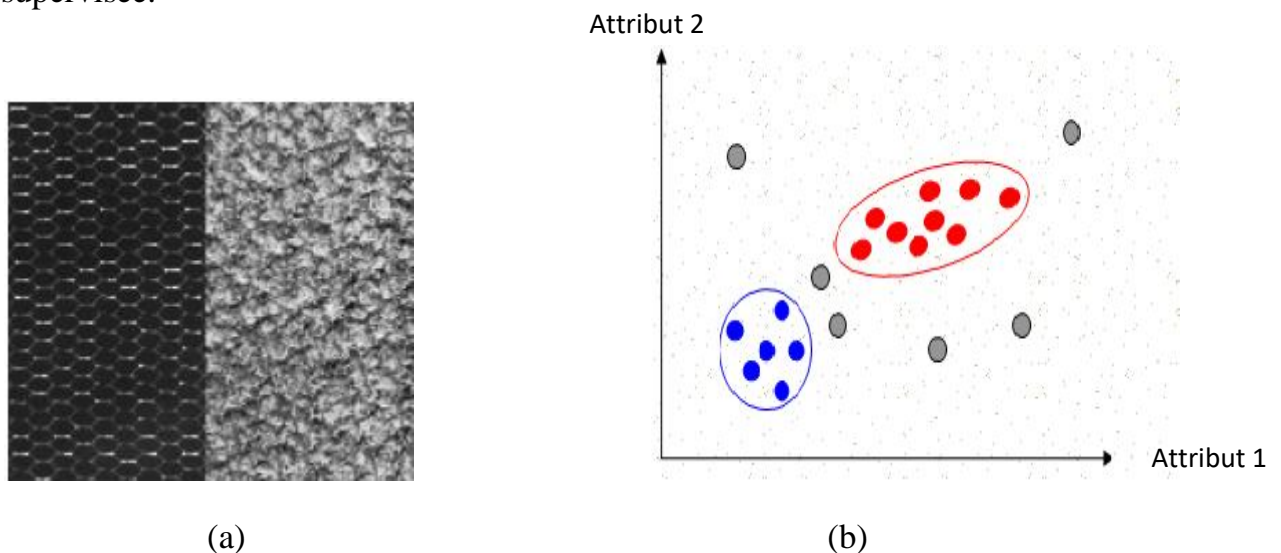


Figure II.5 : Exemple de nuage de points dans l'espace bidimensionnel

a) Image ; b) nuage de points

III.3.1. Classification supervisée (semi-automatique)

La classification est dite supervisée lorsque les différentes classes de l'image sont connues et l'appartenance de certains pixels (prototypes) à ces classes est connue a priori. Ce type de classification nécessite de fournir pour chaque classe un ensemble de pixels représentatifs de la segmentation que l'on souhaite obtenir. Ces pixels permettent de construire une fonction d'identification ou de discrimination pour les autres pixels. Cette fonction d'identification réalise un découpage de l'espace de

représentation. A chaque zone de ce découpage est affectée une classe a priori. Les autres pixels sont ensuite classés en fonction de leurs positions dans l'espace des paramètres.

La méthode des k plus proches voisins fait partie des méthodes de classification supervisées les plus couramment utilisées. Elle permet de traiter des nuages de points non linéairement séparables. Cette approche a l'avantage d'être à la fois simple et efficace. Dans un contexte de classification d'une nouvelle observation x , l'idée fondatrice est de faire voter les plus proches voisins de cette observation. La classe de x est déterminée en fonction de la classe majoritaire parmi les k plus proches voisins de l'observation x .

La figure ci-dessous illustre le principe de cette méthode. Parmi les cinq plus proches voisins du pixel x qu'on souhaite classer, deux pixels appartiennent à la classe 1 et trois pixels appartiennent à la classe 2, c'est donc la classe 2 qui sera attribuée au pixel x .

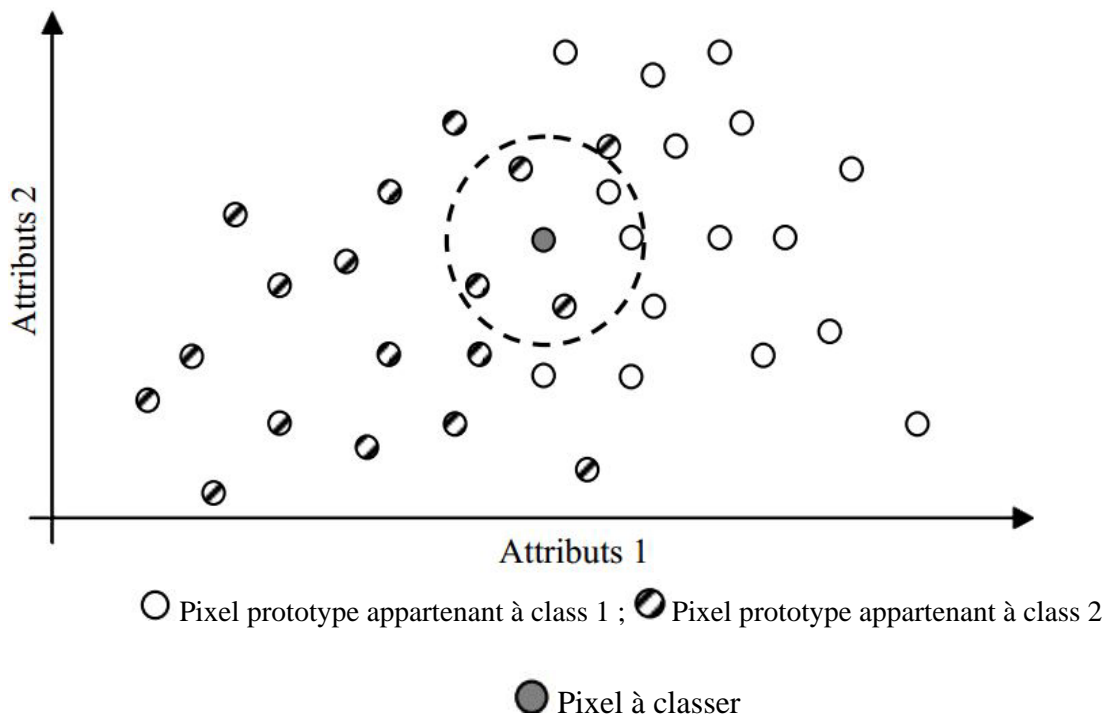


Figure II.6 : Illustration de la méthode des k plus proches voisins

III.3.2. Classification non supervisée (automatique)

Cette méthode de classification est aussi appelée "classification automatique", ou encore "regroupement". Aucune information a priori sur l'image n'est connue. On cherche alors à regrouper les différentes données à traiter en fonction de la valeur de leurs descripteurs de manière à créer des classes homogènes. On suppose qu'on dispose d'un ensemble d'objets que l'on note par $X = \{x_1, x_2, \dots, x_N\}$ caractérisé par un ensemble de descripteurs. L'objectif de la classification est de trouver pour chaque objet "x" la classe à laquelle il est affecté parmi un ensemble de classes qu'on note par $C = \{c_1, c_2, \dots, c_m\}$ avec $m \leq N$

Ce procédé nécessite généralement de fixer au préalable le nombre de classes désirées (m), que ce soit de manière empirique ou automatique. 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.

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.

Nous développons cette dernière méthode car c'est celle-ci que nous utilisons dans notre travail.

IV. L'algorithme de classification K-means

L'algorithme K-means est un des algorithmes les plus connus dans la communauté de classification des données. Dans cet algorithme, chaque classe est représentée par la moyenne (mean) ou la moyenne pondérée de ses points de données constituant ainsi son centre. K-means est un algorithme itératif. Il commence avec un ensemble de K points de référence (centres) choisis de manière aléatoire ou par l'utilisateur. Au début, les points de données sont partitionnés en K classes : un point appartient à une classe si le point de référence de cette classe est le plus proche de lui. La mise à jour des points de référence et l'affectation des points de données aux classes sont réalisées pendant les itérations successives [12].

Il y a plusieurs versions de K-means [13] qui diffèrent dans la mise à jour des

classes et/ou dans le critère pour faire cette mise à jour. Les plus connus sont les algorithmes Standard K-means, K-means de Lloyd et Continuous K-means [14].

Dans l'algorithme dit Standard K-means, la mise à jour des centres est faite pendant et après une itération. Si un point appartient à une classe et que pour lui, le centre de cette classe est le point de référence le plus proche, alors il n'y aura aucun ajustement. Mais si après avoir affecté un point y à une classe C_i , on trouve qu'il y a une autre classe C_j dont le centre est le point de référence le plus proche de y que celui de C_i , alors il faut réaffecter y à la classe C_j et recalculer les centres des deux classes. Les points de référence de ces deux classes se déplacent donc vers de nouveaux centres. Dans l'algorithme de Lloyd, les ajustements sont réalisés en calculant le centre pour chaque classe et en utilisant ces centres comme les points de référence dans l'itération suivante pour tous les points de données. La mise à jour des centres n'est faite qu'après une itération. Dans l'algorithme Continuous K-means proposé par McQueen [15] les K centres initiaux sont choisis arbitrairement parmi les points de la population entière. Si l'échantillon est assez gros, alors la distribution des points de référence initiaux pourrait refléter celle des points de la population. Dans cet algorithme, contrairement au standard K-means où tous les points sont séquentiellement examinés, on n'examine qu'un échantillon aléatoire des points. Si le jeu de données est gros et l'échantillon est représentatif du jeu de données, alors l'algorithme peut converger plus vite qu'un algorithme qui doit examiner séquentiellement tous les points [14].

Dans ce travail, nous utilisons l'algorithme standard des K-means, qui dans sa version la plus simple nécessite la connaissance préalable du nombre K de classes. Son principe consiste à choisir initialement K centres définissant ainsi une partition $C = \{C_1, C_2, \dots, C_K\}$ de K classes. Chacune des observations à classer est assignée à l'une des C_k classes selon une mesure de similarité entre chaque centre des classes et l'observation considérée. Une nouvelle partition C est ainsi obtenue et permet de redéfinir les nouveaux centres des classes. Ce processus itératif, prend fin lorsqu'un critère d'arrêt est satisfait. La méthode des K-means, également connue sous le nom de procédure de minimisation de l'erreur quadratique [15], fonctionne suivant l'algorithme de la figure II.7. Il s'agit de classer un ensemble de données

$Y = (y_1, y_2, \dots, y_n)$ avec $y_i \in \mathbb{R}^D$ en K classes où chaque classe C_k est définie par son centre $\overline{y}_k = (\overline{y}_{k1}, \overline{y}_{k2}, \dots, \overline{y}_{kp})$, avec :

$$\overline{y}_{kj} = \frac{1}{N_k} \sum_{y_i \in C_k} y_{ij} \quad (\text{II, 11})$$

j étant l'indice de l'attribut, N_k le nombre d'observations de la classe k et y_{ij} l'attribut j de l'individu y_i . Chaque individu y_i se trouvant dans la classe k est individuellement réassigné à la classe dont la distance euclidienne par rapport à son centre est la plus faible.

Les distances de l'individu y_i par rapport au différentes classes sont définie par :

$$d_k(y_i) = d(y_i, \overline{y}_k) = \|y_i - \overline{y}_k\|^2 \quad (\text{II, 12})$$

Le critère à optimiser est défini par :

$$J(C) = \sum_{k=1}^K \sum_{y_i \in C_k} d(y_i, \overline{y}_k)^2 = \sum_{k=1}^K J_k(C) \quad (\text{II, 13})$$

La partition C^* qui optimise ce critère est définie par : $J(C^*) = \min_C J(C)$.

L'algorithme de K-means est le suivant :

1. Donner le nombre K de classes, ainsi qu'une partition initiale C des n observations de l'échantillon en K classes.
2. Calculer $J(C)$ et \overline{y}_k , $k = 1, 2, \dots, K$.
3. Pour chaque observation y_i affectée initialement à la classe C_k , i variant de 1 à n :
 - Calculer $d_k(y_i)$, $\forall k = 1, 2, \dots, K$.
 - Affecter y_i à la classe C_s telle que : $d_s(y_i) \leq d_k(y_i)$, $\forall k = 1, 2, \dots, K$.
 - Calculer \overline{y}_k et \overline{y}_s
 - Recalculer $J(C)$
4. Si $J(C)$ a changé durant l'étape 3, revenir en 3, sinon arrêter.

Figure II.7 : Algorithme K-means

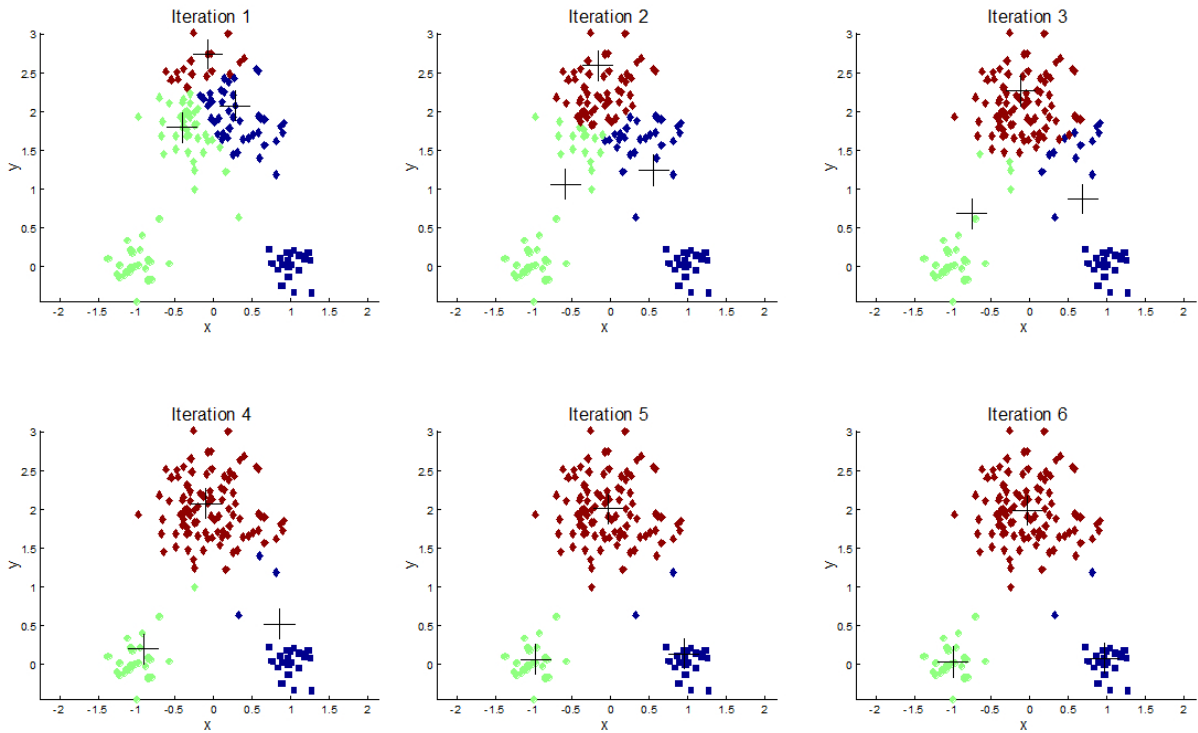


Figure II.8 : Exemple de classification en utilisant l’algorithme de K-means

Parmi les distances les plus utilisées, on peut citer : la distance euclidienne, la distance mahalanobis et la distance city block [16].

La distance Euclidienne est la distance la plus employée. Elle est utilisée généralement en présence de classe sphériques, et on aboutit à des résultats satisfaisants, elle est définie par :

$$d_e(y_1, y_2) = \|y_1 - y_2\| = \sqrt{\sum_{d=1}^D (y_{1d} - y_{2d})^2} \tag{II, 14}$$

Où : y_1 et $y_2 \in \mathbb{R}^D$

Avec : \mathbb{R}^D l'espace des paramètres.

La distance City block entre deux vecteurs y_1 et y_2 , elle est donnée par :

$$d_c(y_1, y_2) = \sum_{i=1}^D |y_{1d} - y_{2d}| \tag{II, 15}$$

Où D est le nombre de paramètres.

La distance Mahalanobis utilise la matrice de covariance des sommes.

$$d_k(y) = (y - y_k)^t T_k^{-1} (y - y_k) \tag{II, 16}$$

y est un vecteur d'attributs, y_k est la moyenne des vecteurs y sur la classe C_k .

T_k^{-1} est la matrice inverse de la matrice covariance associé à cette classe.

$(y - y_k)^t$ est le transposé du vecteur $(y - y_k)$.

V. Conclusion

Nous avons évoqué dans ce chapitre, d'une manière générale, quelques notions essentielles sur la segmentation. Nous nous sommes intéressés plus particulièrement à ses différentes approches.

Parmi ces approches, nous avons retenu celle de classification en se basant sur l'algorithme de K-means que nous avons développé car il sera utilisé dans le chapitre suivant pour la segmentation des images au niveau de gris en utilisant des paramètres de texture.

CHAPITRE III

I. Introduction

Ce travail a pour but de tester de nouveaux paramètres de texture, inspirés des méthodes utilisées pour le traitement de la parole. Ces paramètres sont extraits des signaux de la parole qui sont des signaux monodimensionnels. Or l'image est un signal bidimensionnel. Nous allons donc proposer une adaptation du monodimensionnel vers le bidimensionnel. Ces paramètres de texture seront ensuite soumis à un classifieur qui réalise la segmentation.

Pour rendre compte de la pertinence de ces paramètres, nous allons réaliser une comparaison de la segmentation par l'algorithme K-means obtenue par ces paramètres avec celle obtenue par les paramètres de texture extraits de la matrice de cooccurrence.

II. Proposition des paramètres de texture

Avants d'extraire les paramètres de texture, chaque pixel sera décrit par un signal bidimensionnel qui est en réalité une fenêtre de taille $n \times n$ centrée sur ce pixel.

Comme l'éclairage n'est pas toujours homogène dans la majorité des images, nous proposons de soustraire la moyenne locale pour chacune de ces fenêtres.

En effet pour la même texture nous obtenons des signaux différents si l'éclairage est différent, cette différence peut se répercuter sur les paramètres à extraire.

Sur l'image de la figure III.1, nous avons une texture unique avec un éclairage non homogène. Sur cette figure, nous observons deux fenêtres ouvertes autour de deux pixels qui sont indiqués par "+". L'un de ces pixels appartient à une zone claire et l'autre à une zone sombre. Ces fenêtres sont représentées ensuite par des signaux bidimensionnelles. Sur la figure III.2 nous observons les mêmes signaux mais en leurs soustrayant la moyenne locale.

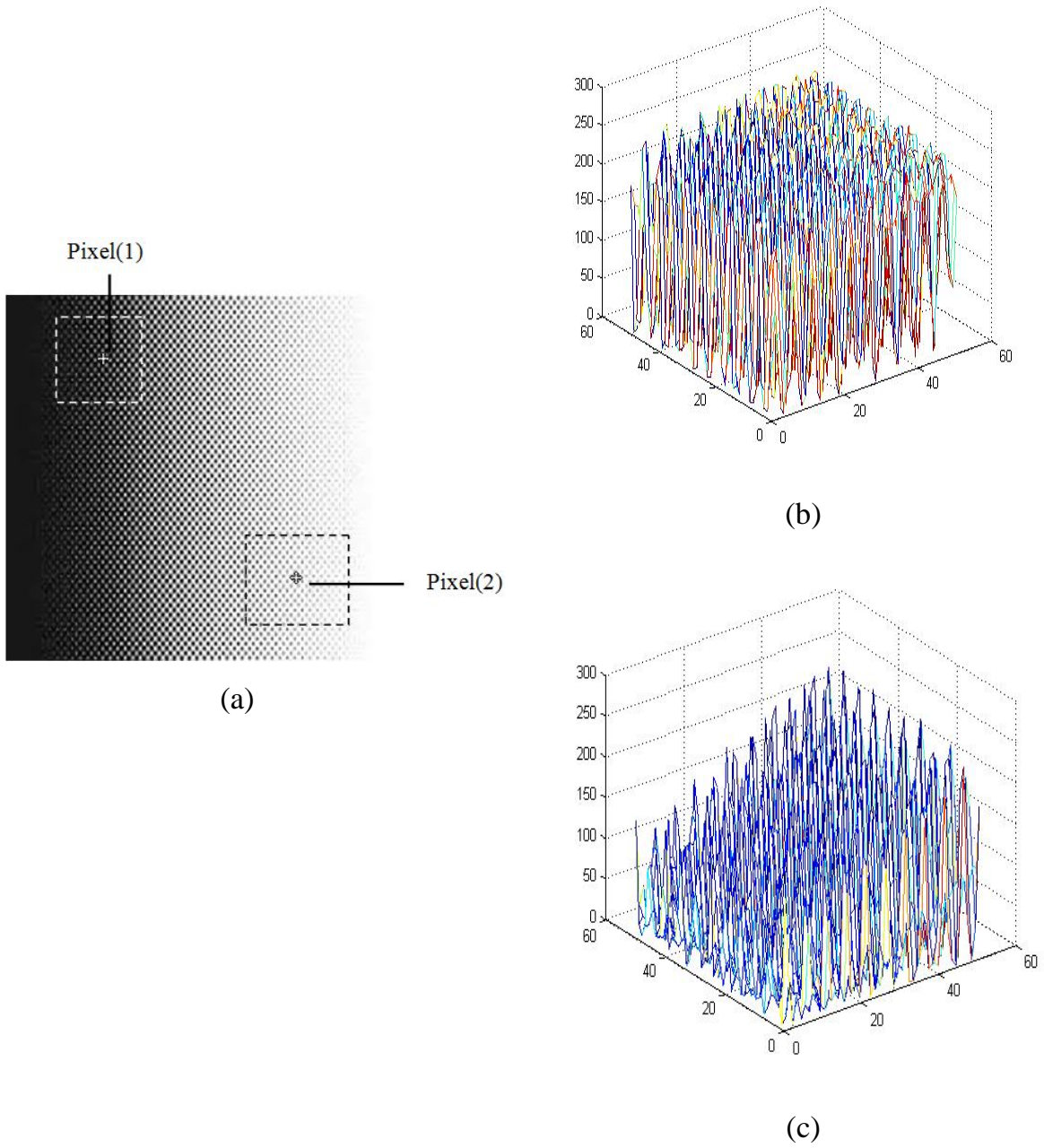


Figure III.1: Représentation des signaux bidimensionnels des fenêtres autour des pixels sélectionnés

a) Image I, b) Signal d'une fenêtre de la partie claire, c) Signal d'une fenêtre de la partie sombre

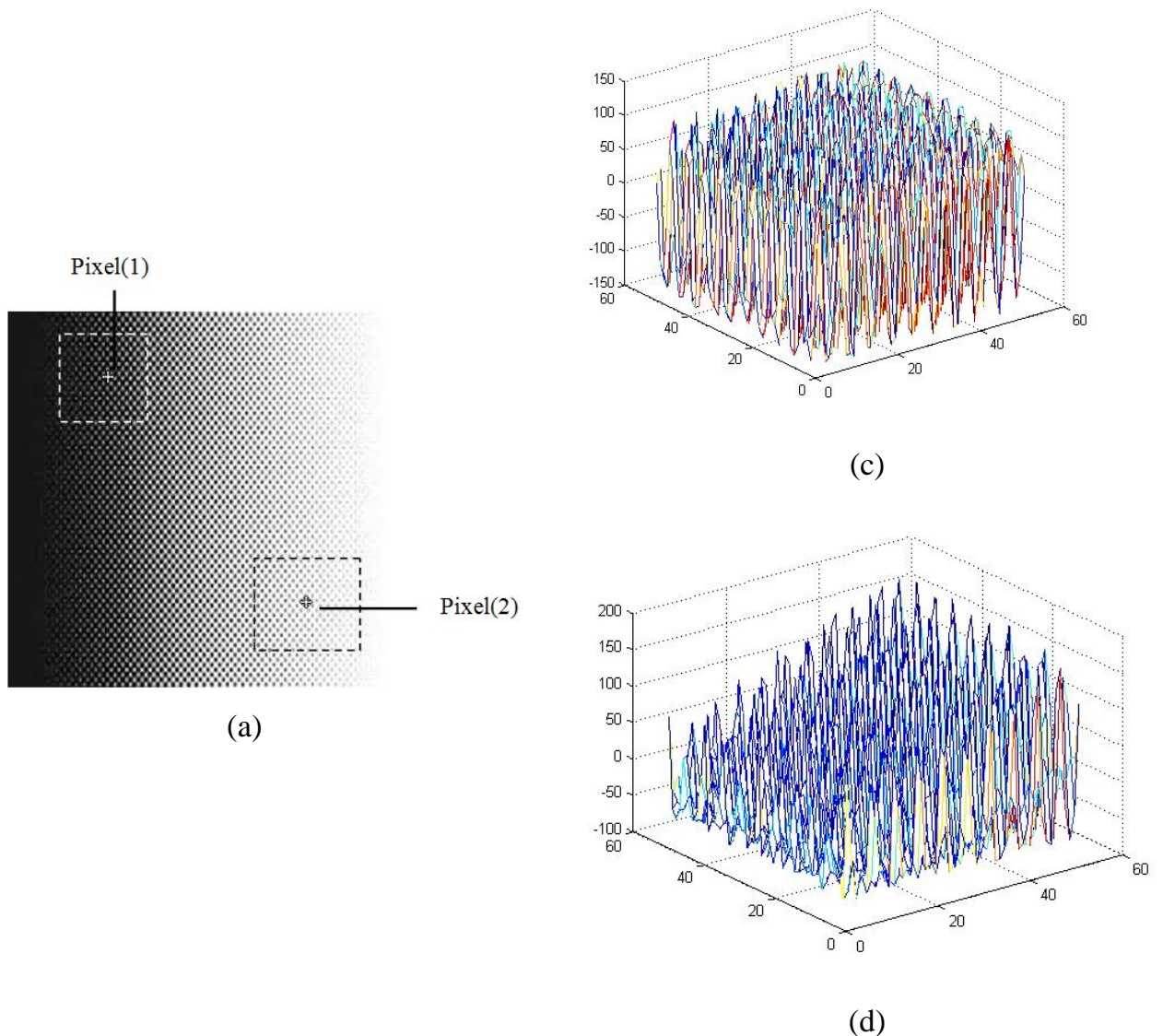


Figure III.2: Représentation des signaux bidimensionnels des fenêtres autour des pixels sélectionnés après soustraction de la moyenne locale.

a) Image I, b) Signal d'une fenêtre de la Partie claire, c) Signal d'une fenêtre de la Partie sombre

Après avoir représenté ces signaux bidimensionnels, nous pouvons extraire un certain nombre de paramètres.

Dans ce travail, nous proposons trois paramètres : les frontières de passage par zéro, le nombre des régions et un paramètre basé sur l'analyse cepstrale qui est le maximum des cepstres. Nous expliquerons chacun de ces paramètres et nous afficherons les résultats de segmentation obtenus sur des images réelles et synthétiques dans les paragraphes qui suivent.

II.1. Frontière des passages par zéro

Ce paramètre est inspiré du paramètre « nombre du passage par zéro » qui est utilisé dans la reconnaissance de la parole.

En effet dans le signal de la parole le nombre du passage par zéro correspond au nombre de fois que le signal dans sa représentation amplitude/temps, passe par la valeur zéro.

Il est fréquemment employé pour des algorithmes de détection de section voisée/non voisée dans un signal.

Rappelons que le signal de la parole est une alternance des zones assez périodiques et des zones bruitées appelées zones voisées et non-voisées, et du fait de sa nature aléatoire, le bruit possède généralement un taux de passage par zéro supérieur à celui des parties voisées [17].

En ce qui nous concerne, nous disposons d'un signal bidimensionnel. Donc nous n'avons plus des points de passages par zéro mais des frontières de passage par zéro, nous allons donc calculer la longueur de ces frontières.

Pour ce faire nous transformons le signal bidimensionnel à plusieurs niveaux en un signal binaire en affectant la valeur 1 à toutes les valeurs positives et la valeur 0 à toutes les valeurs négatives (voir figure III.4).



Figure III.3: Exemple d'image réelle (fleurs)

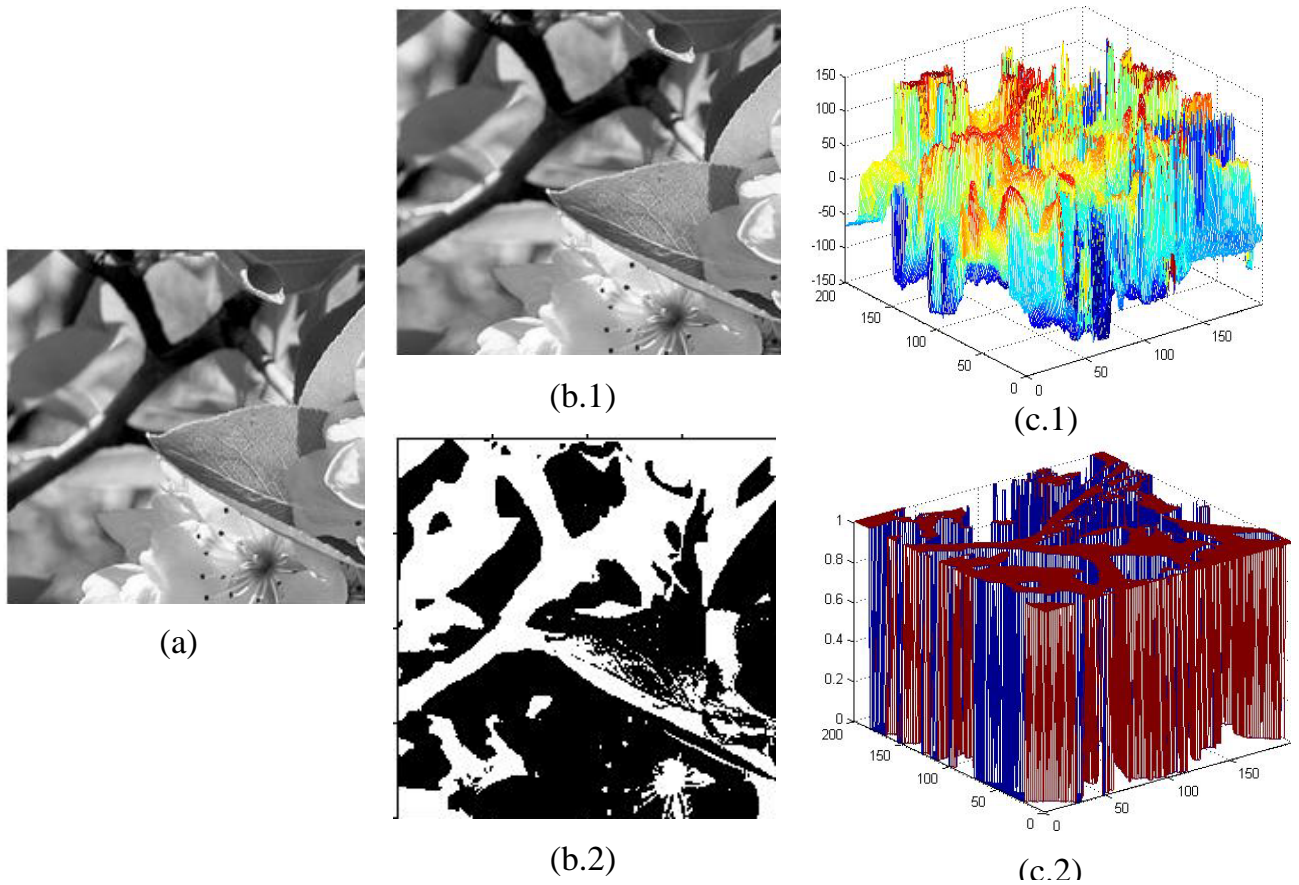


Figure III.4: Soustraction de la moyenne locale et binarisation d'une fenêtre de l'image fleurs et représentation de leur signaux bidimensionnels

(a) Fenêtre de l'image fleurs, (b.1) Fenêtre en image après avoir enlevé la moyenne, (b.2) Fenêtre en image binaire, (c.1) Signal de la fenêtre (b.1), (c.2) Signal de la fenêtre (b.2)

Les frontières de passage par zéro sont donc les frontières entre les régions de niveau 1 et les régions de niveau 0. Nous réalisons une détection de contour pour déterminer ces frontières, un exemple des résultats obtenus est affiché sur la figure III.5.



Figure III.5: Localisation des frontières des passages par zéro de la fenêtre de la figure précédente

a) Image binaire, b) image contours

Le paramètre de texture que nous allons utiliser dans nos tests sera la longueur totale des frontières de passage par zéro qui sera noté « FPZ » qui correspond dans l'exemple de la figure III.5.b au nombre de pixels de niveau 1.

Dans l'exemple de la figure III.6 nous afficherons 3 fenêtres sélectionnées dans l'image fleurs, et les valeurs de FPZ leur correspondant.

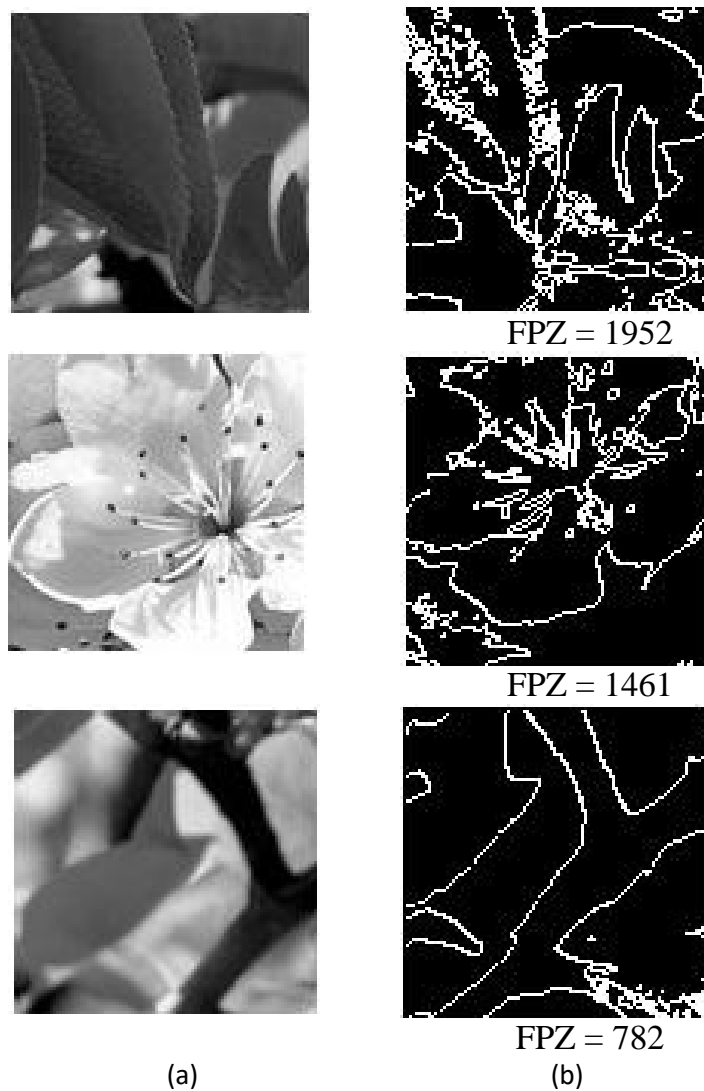


Figure III.6 : Exemples des valeurs de FPZ pour 3 fenêtres de l'image fleurs
a) Fenêtres de l'image fleurs, b) images des frontières de passage par zéro

Sur la figure III.7 Nous illustrons l'image du paramètre FPZ obtenu pour chaque pixel de l'image avec une taille de fenêtre 11x11.

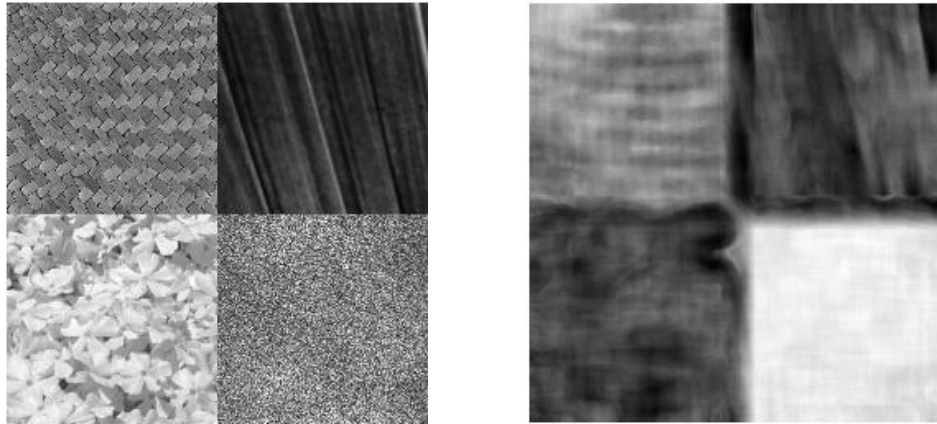


Figure III.7: Affichage en image du paramètre FPZ obtenu sur une image synthétique

II.2. Nombre de régions

Il se peut que le paramètre FPZ (longueur totale des frontières de passage par zéro) soit quasi identique pour des textures différentes comme cela est illustré dans l'exemple synthétique de la figure III.8.

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

(a)

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

(b)

Figure III.8:Exemple de deux fenêtres de textures différentes ayant la même valeur de FPZ et un nombre de régions différent

Nous pouvons constater que dans le cas de la figure III.8.a ce paramètre vaut 48 (FPZ=48). Il est de même pour le cas de la figure III.8.b, sauf que ces deux textures sont différentes.

Il nous parait donc nécessaire d'ajouter un autre paramètre pour faire la différence. Nous avons opté pour le calcul du nombre de région contenu dans ces fenêtres. Ce nombre de région noté «NR » vaut 6 dans le cas de la figure III.8.a, et 17 dans le cas de figure III.8.b.

Sur la figure III.9 Nous illustrons l'image du paramètre NR obtenu pour chaque pixel de l'image synthétique 1 avec une taille de fenêtre 11x11.

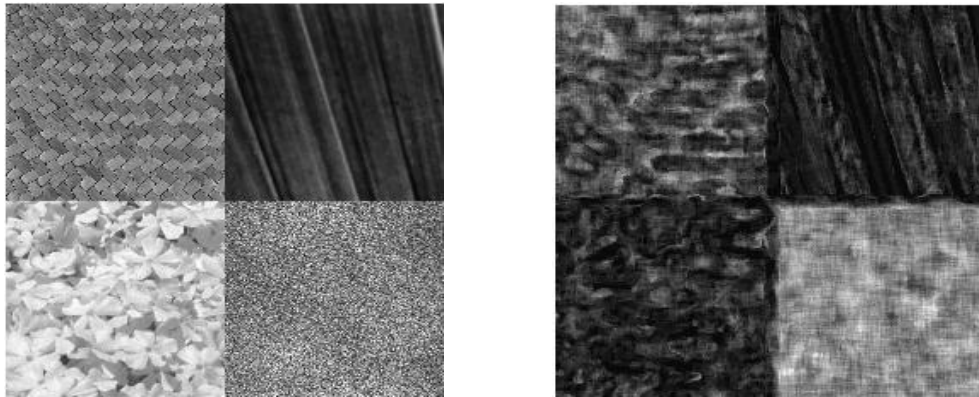


Figure III.9:Affichage en image du paramètre NR obtenu sur une image synthétique 1

II.3. Paramètre basé sur l'analyse cepstrale

L'analyse cepstrale est une technique non linéaire du traitement du signal qui est largement appliquée en traitement de la parole et en filtrage.

Le cepstre d'un signal $x(t)$ est une transformation de ce signal du domaine temporel vers un autre domaine analogue au domaine temporel. Il est défini comme étant la transformée de Fourier inverse appliquée au logarithme de la transformée de Fourier du signal [18].

$$C(\tau) = C(x(t)) = \text{TF}^{-1}(\log(|\text{TF}(x(t))|))$$

Avec : $x(t)$: signal défini dans le domaine temporel

TF : transformée de fourrier

TF^{-1} : transformée de fourrier inverse

Log : logarithme népérien

C: cepstre

τ : quéfrence

Pour rappeler le fait que l'on effectue une transformation inverse à partir du domaine fréquentiel, les dénominations des notions sont des anagrammes de celles utilisées en fréquentiel. Ainsi l'analogue du spectre est le cepstre, de la fréquence la quéfrece, du filtrage le lifrage, de la phase la saphe, de l'analyse l'alalyse (parfois).

L'analyse cepstrale est utilisée pour des signaux monodimensionnels, nous l'adaptions dans ce qui suit pour les signaux bidimensionnels. Il suffit dans ce cas d'utiliser la FFT à 2D et la FFT inverse en 2D. Plusieurs paramètres peuvent être extraits du signal bidimensionnels après l'application du cepstre. Dans notre cas nous choisissons de déterminer le maximum du signal cepstre.

Sur la figure III.10 Nous illustrons l'image de ce paramètre qui sera noté « MC », obtenu pour une image synthétique avec une taille de fenêtre 11x11.

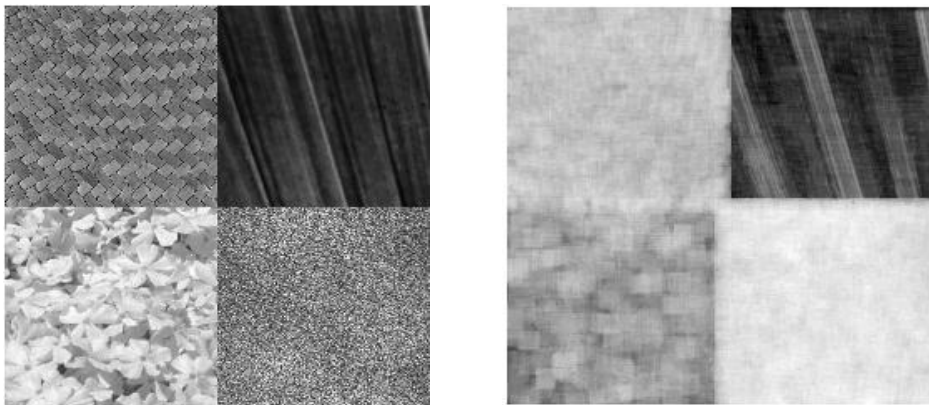


Figure III.9:Affichage en image du paramètre MC obtenu sur une image synthétique

III. Test des paramètres de texture

Pour évaluer la pertinence de ces paramètres de texture nous proposons de réaliser la segmentation par l'algorithme K-means en utilisant un à un ces différents paramètres et en les combinant.

Nous comparons à chaque fois les résultats obtenus avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence.

Il nous paraît possible que les résultats peuvent être modifiés si la taille de la fenêtre $n \times n$ est modifiée.

Nous affichons sur les figures III.11 les résultats de la segmentation en utilisant différentes tailles de fenêtre pour chacun des paramètres.

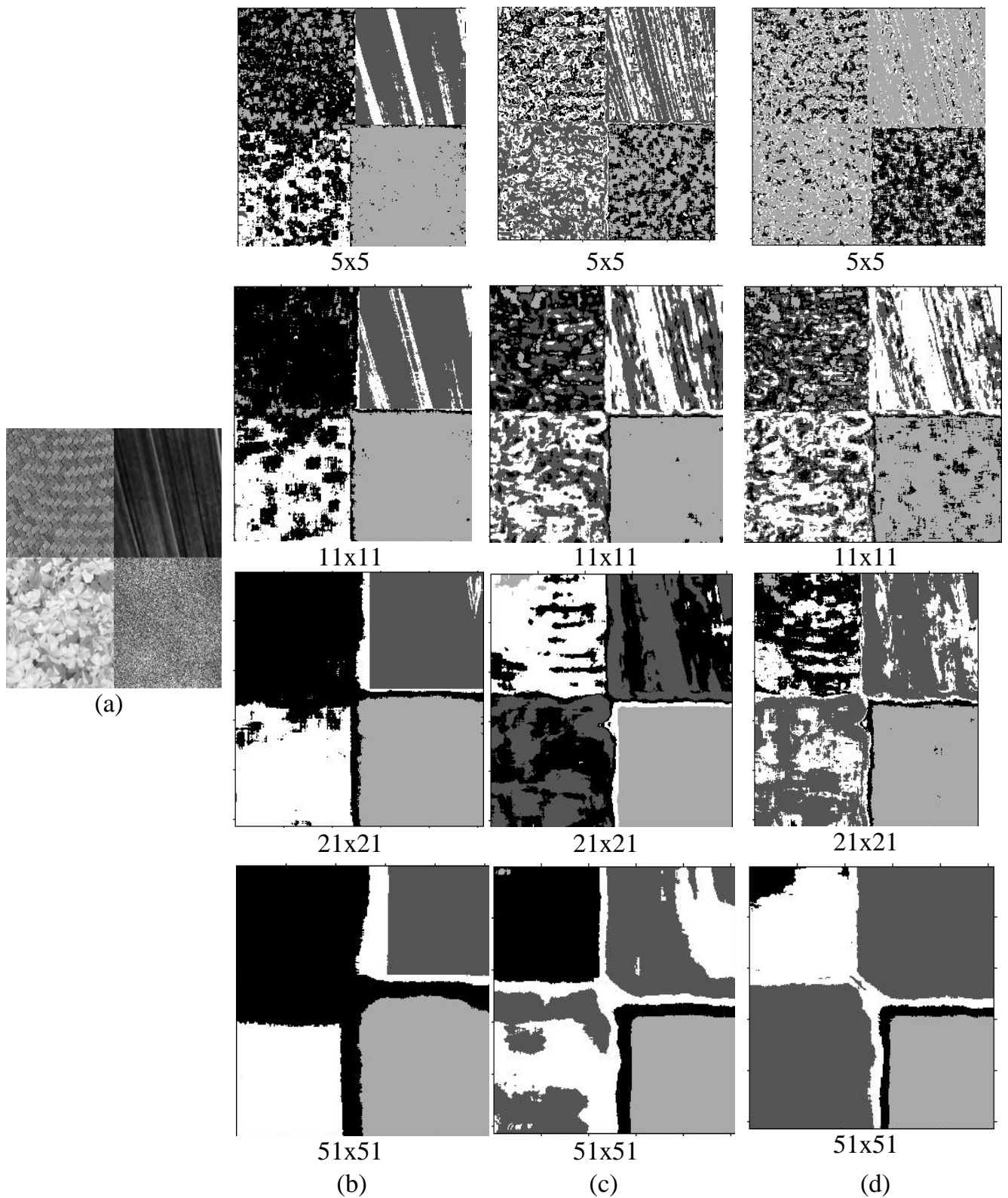


Figure III.11: Résultats de la segmentation en utilisant différents paramètres avec différentes tailles de fenêtres

a) image syntactique, b) MC, c) FPZ, d) NR

Au regard de la figure précédente nous constatons que la taille de la fenêtre influe sur les résultats de la segmentation. En effet dans le cas de l'image synthétique 1, de manière générale, nous observons que les meilleurs résultats sont pratiquement obtenus pour une taille de la fenêtre de 21x21.

Pour des tailles de fenêtre inférieures nous constatons que les résultats sont mauvais, car par exemple en utilisant les paramètres NR et FPZ (figure III.11.c,d) pour la taille 5x5 aucune texture n'est identifiée. Dans le cas du paramètre MC, une seule (sur quatre) texture est bien identifiée. Les résultats de la segmentation s'améliorent au fur et à mesure que la taille de la fenêtre augmente, par exemple pour le paramètre MC, pour une fenêtre de taille 11x11, une texture sur quatre est très bien identifiée, les trois autres sont moyennement identifiées. Le résultat est nettement meilleur dans le cas d'une fenêtre de taille 21x21.

Les régions (texture) sont assez bien segmenté dans le cas d'une fenêtre de taille 51x51, sauf que dans ce cas les bords des régions sont mal identifiés. Il ya donc un compromis à faire pour choisir la taille de la fenêtre. Il nous semble nécessaire que la taille de la fenêtre englobe au moins un motif élémentaire de la texture.

Pour la suite des tests nous optons pour une taille de fenêtre 21x21. En réalité ce choix ne doit pas être fixe car il dépend de la taille de l'image et de la nature des textures contenues dans cette image.

III.1.Résultats de la segmentation en utilisant le paramètre MC (maximum du cepstre)

Dans ce paragraphe nous réalisons une comparaison des résultats de la segmentation obtenus en utilisant le paramètre MC avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence. La figure ci-dessous illustre ces résultats sur deux images synthétiques tirées des benchmark MIT Vistex [19] et mosaic [20] et une image réelle tirées du benchmark berkely [21].

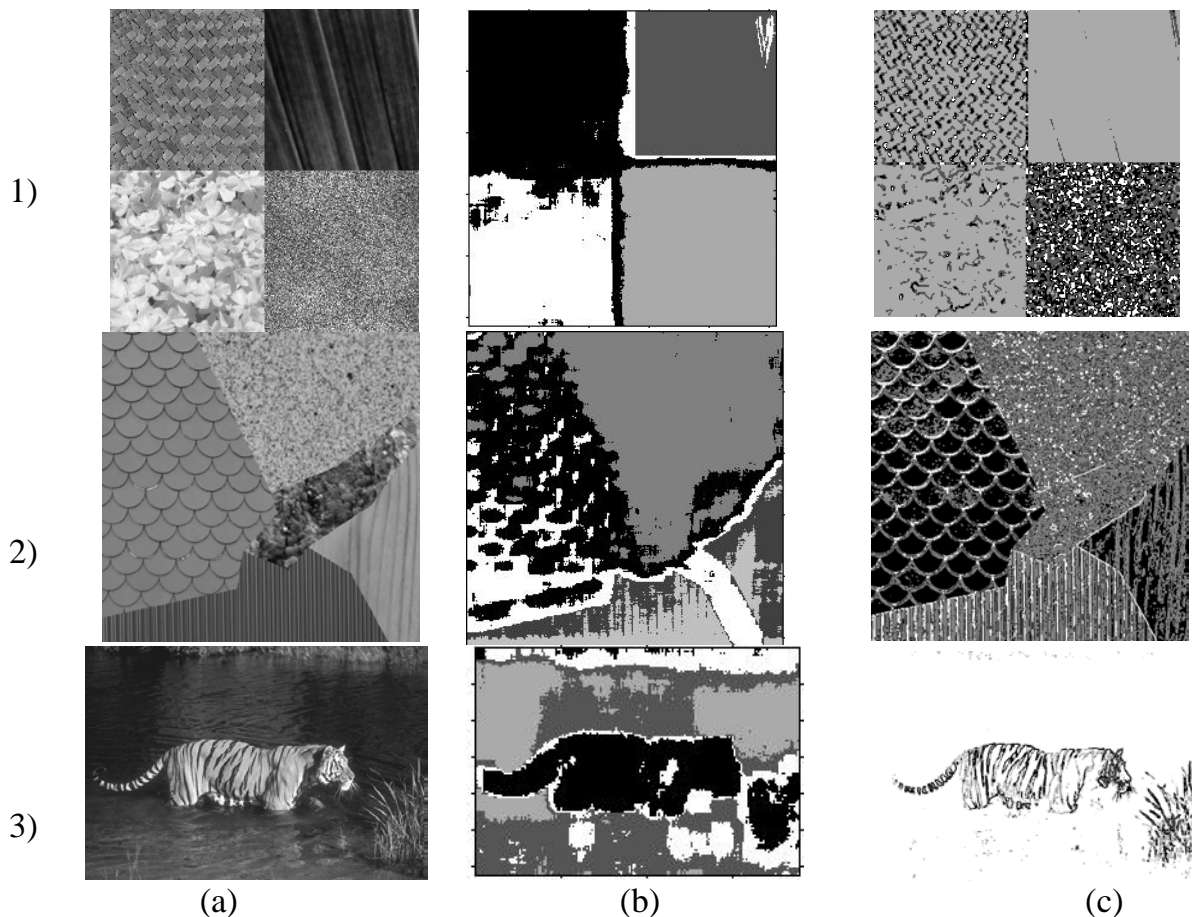


Figure III.12: Résultats de la segmentation en utilisant le paramètre MC et les paramètres de la matrice de cooccurrence

- 1) image synthétique 1, 2) image synthétique 2, 3) image réelle
 a) image d'origine, b) résultat de la segmentation en utilisant le paramètre MC
 c) résultat de la segmentation en utilisant le paramètre matrice de cooccurrence

Nous constatons que sur la figure précédente les résultats de la segmentation en utilisant le paramètre MC sont nettement meilleurs que ceux obtenus en utilisant les paramètres de la matrice de cooccurrence. En effet, par exemple dans le cas de l'image synthétique 1 les régions de l'image sont bien identifiées en utilisant le paramètre MC (voir figure III.12.1.b), alors qu'il n'y a qu'une seule région qui est bien identifiée en utilisant la matrice de cooccurrence (voir figure III.12.1.c).

Dans le cas de la figure III.12.2.b même si le résultat de la segmentation n'est pas très bon (la texture du milieu est confondue avec celle d'en haut à droite), il est nettement meilleur que celui obtenu en utilisant la matrice de cooccurrence car aucune texture n'est bien identifiée.

Nous retrouvons le même résultat dans la figure III.12.3, le tigre est bien identifié en utilisant le paramètre MC car, globalement, il a été identifié comme étant une seule

classe, alors qu'il est représenté par plusieurs classes en utilisant le paramètre de la matrice de cooccurrence.

III.2. Résultats de la segmentation en utilisant le paramètre FPZ (frontières de passage par zéro)

Dans ce paragraphe nous réalisons une comparaison des résultats obtenus en utilisant le paramètre FPZ avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence.

La figure ci-dessous illustre ces résultats sur les mêmes images que précédemment.

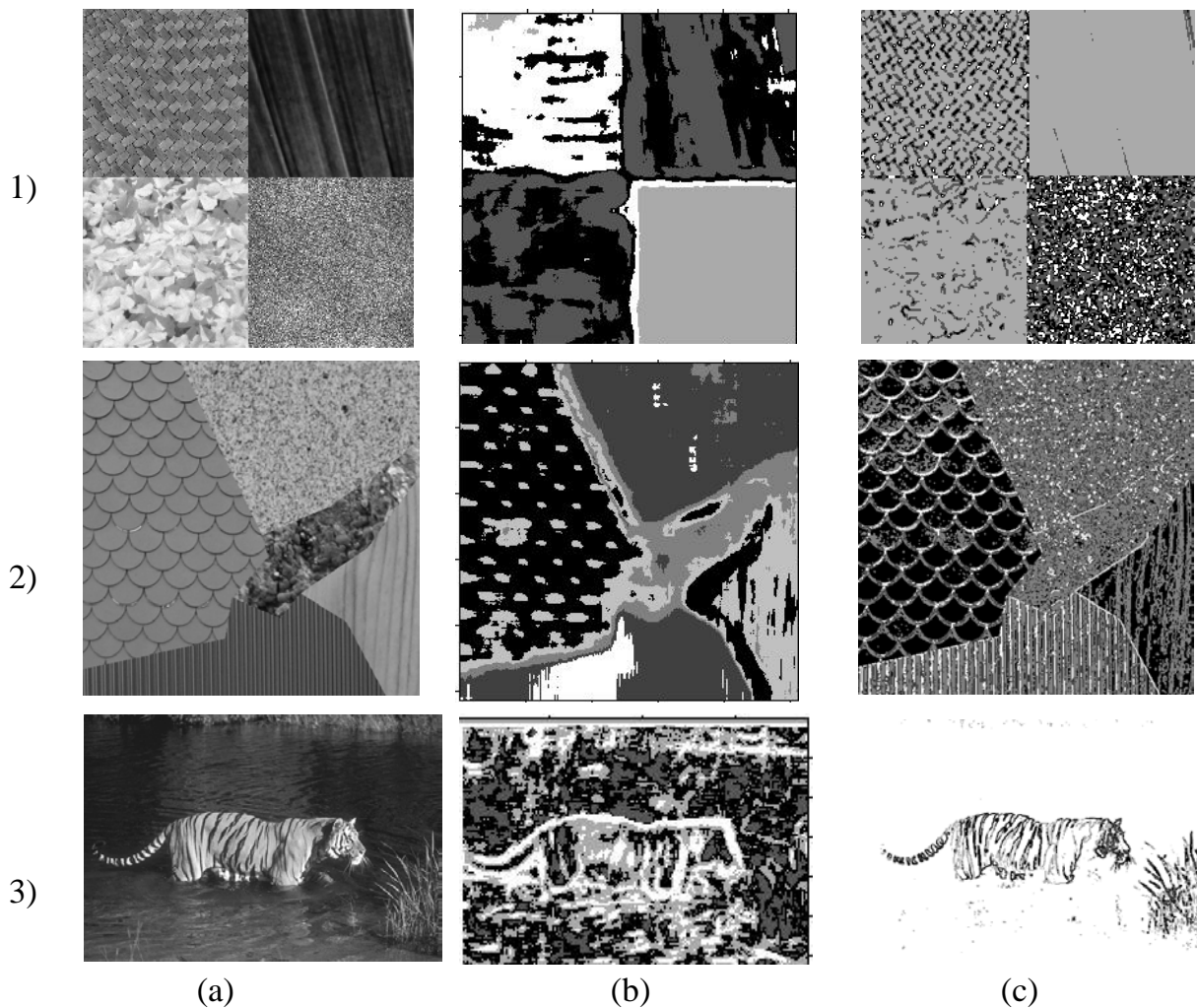


Figure III.13: Résultats de la segmentation en utilisant le paramètre FPZ et les paramètres de la matrice de cooccurrence

- 1) image synthétique 1, 2) image synthétique 2, 3) image réelle a) image d'origine, b) résultat de la segmentation en utilisant le paramètre FPZ
c) résultat de la segmentation en utilisant le paramètre matrice de cooccurrence

Dans le cas de l'utilisation du paramètre FPZ, nous constatons que même si les résultats sont moins bons que ceux obtenus en utilisant le paramètre MC, ils sont meilleurs que ceux obtenus en utilisant la matrice de cooccurrence.

Néanmoins, nous constatons dans le cas de la figure III.13.2.b que les différentes régions de texture sont mieux identifiées que dans le cas de l'utilisation du paramètre MC. En effet la région du centre et celle en haut à droite ne sont plus confondues.

Dans le cas de l'image du tigre le résultat de la segmentation en utilisant le paramètre FPZ (figure III.13.3.b) n'est pas concluant, il est de même en utilisant la matrice de cooccurrence.

III.3.Résultats de la segmentation en utilisant le paramètre NR (nombre de régions)

Dans ce paragraphe nous réalisons une comparaison des résultats obtenus en utilisant le paramètre NR avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence.

La figure III.14 illustre ces résultats sur deux images synthétiques et une image réelle précédentes.

Au regard de cette figure, nous remarquons que la segmentation obtenus en utilisant le paramètre NR n'est pas très concluante sauf qu'elle reste meilleure que celle obtenue en utilisant la matrice de cooccurrence.

Ce paramètre a été réfléchi comme complément au paramètre FPZ, c'est pour cela que nous allons donner dans le paragraphe suivant les résultats en combinant les deux paramètres FPZ et NR.

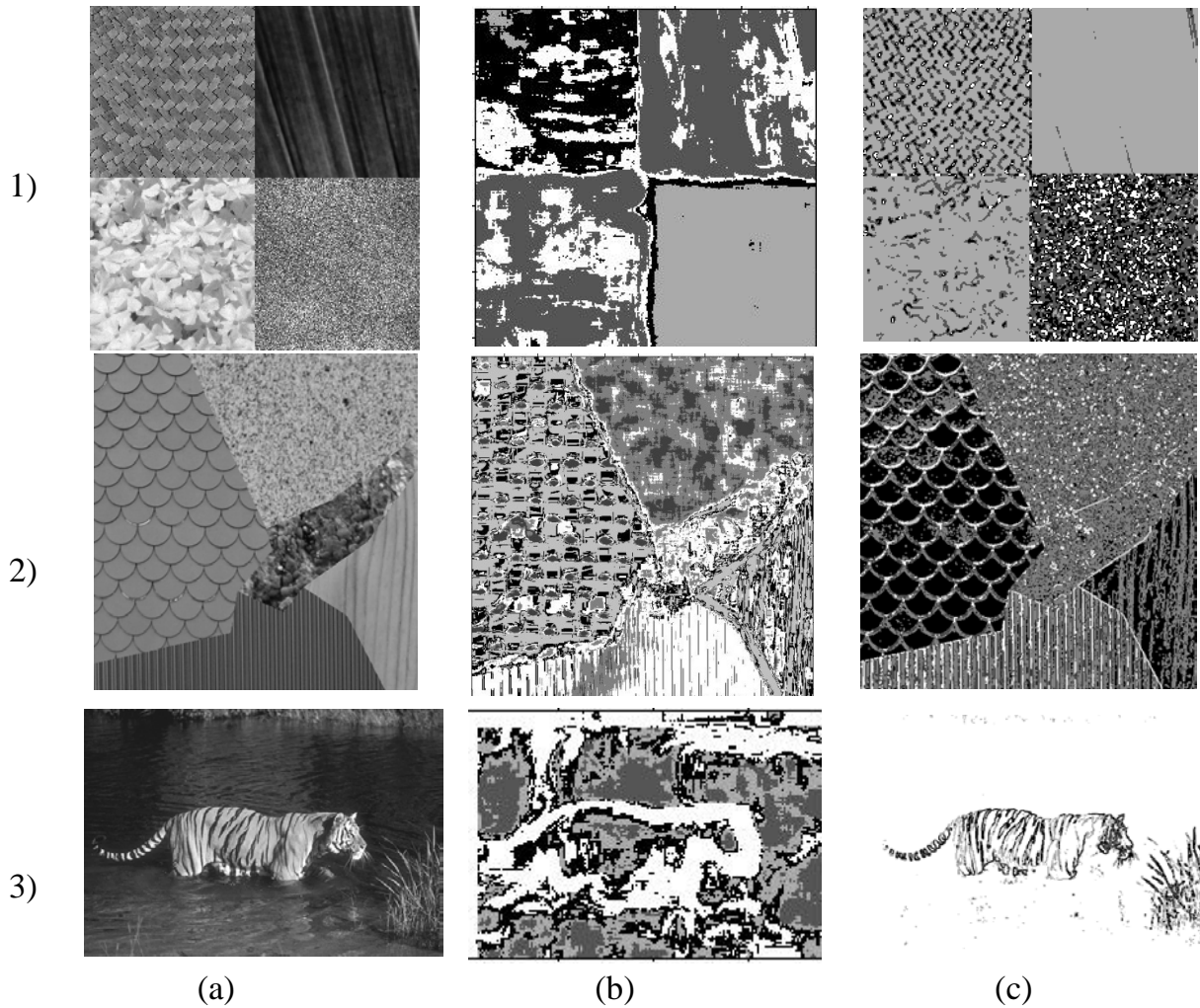


Figure III.14: Résultats de la segmentation en utilisant le paramètre NR et les paramètres de la matrice de cooccurrence

- 1) image synthétique 1, 2) image synthétique 2, 3) image réelle
 a) image d'origine, b) résultat de la segmentation en utilisant le paramètre NR
 c) résultat de la segmentation en utilisant le paramètre matrice de cooccurrence

III.4. Résultats de la segmentation en utilisant les deux paramètres FPZ et NR

Dans ce paragraphe nous réalisons une comparaison des résultats de la segmentation obtenus en utilisant les deux paramètres FPZ et NR avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence.

La figure ci-dessous illustre ces résultats.

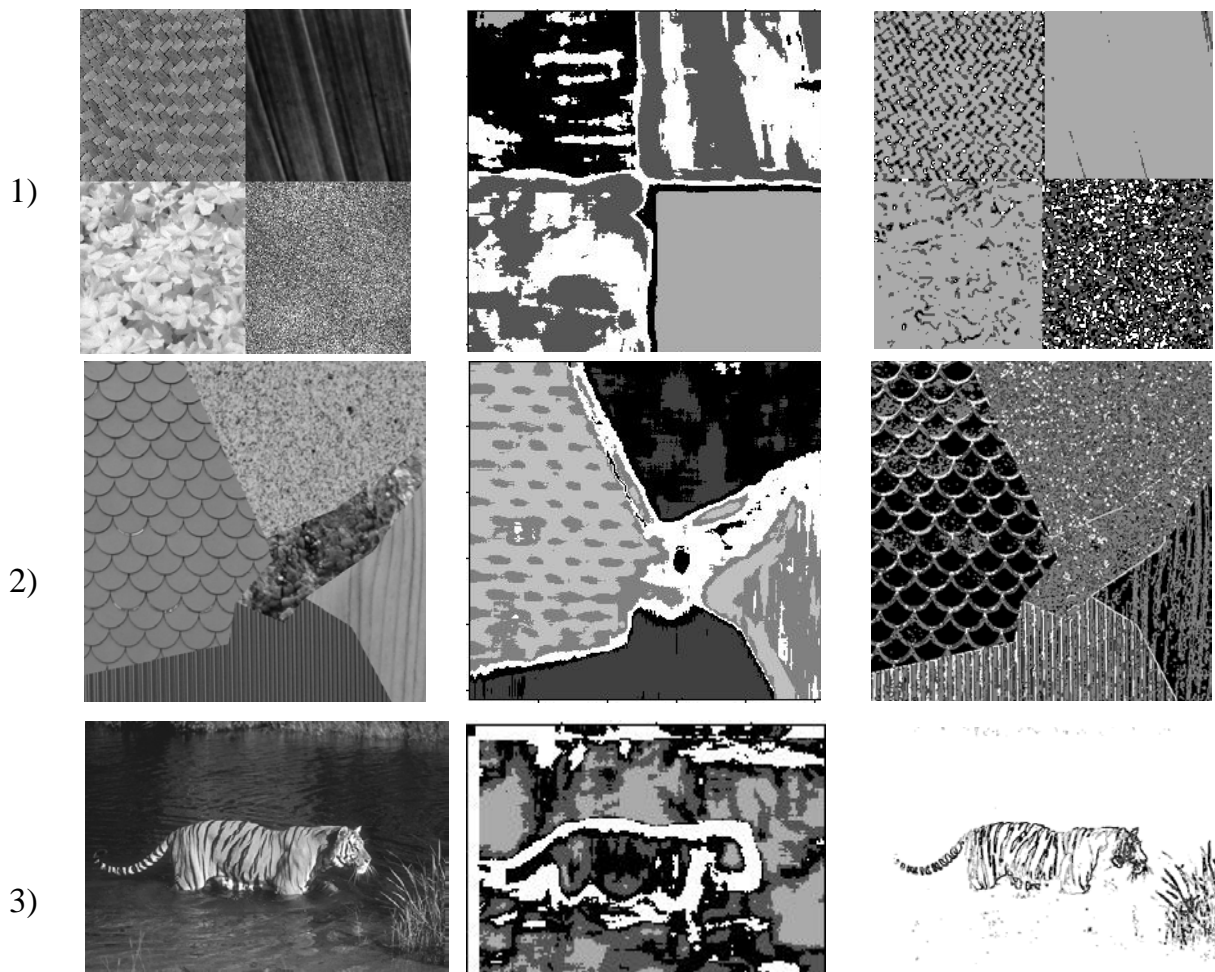


Figure III.15: Résultats de la segmentation en combinant les paramètres FPZ et NR et en utilisant les paramètres de la matrice de cooccurrence

1) image synthétique 1, 2) image synthétique 2, 3) image réelle

a) image d'origine, b) résultat de la segmentation en combinant les paramètres FPZ et NR

c) résultat de la segmentation en utilisant le paramètre matrice de cooccurrence

Nous constatons qu'en combinant les paramètres FPZ et NR, les résultats sont améliorés. Par exemple, nous pouvons constater cette amélioration dans le cas de la figure III.15.2.b. Cette dernière est bien meilleure que celles des figures III.13.2.b et III.14.2.b. Par exemple la région du bas est très bien identifiée dans ce dernier cas.

III.5. Résultats de la segmentation en utilisant l'ensemble des paramètres

Dans ce paragraphe nous réalisons une comparaison des résultats de la segmentation obtenus en utilisant la combinaison de l'ensemble des paramètres avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence.

La figure ci-dessous illustre ces résultats.

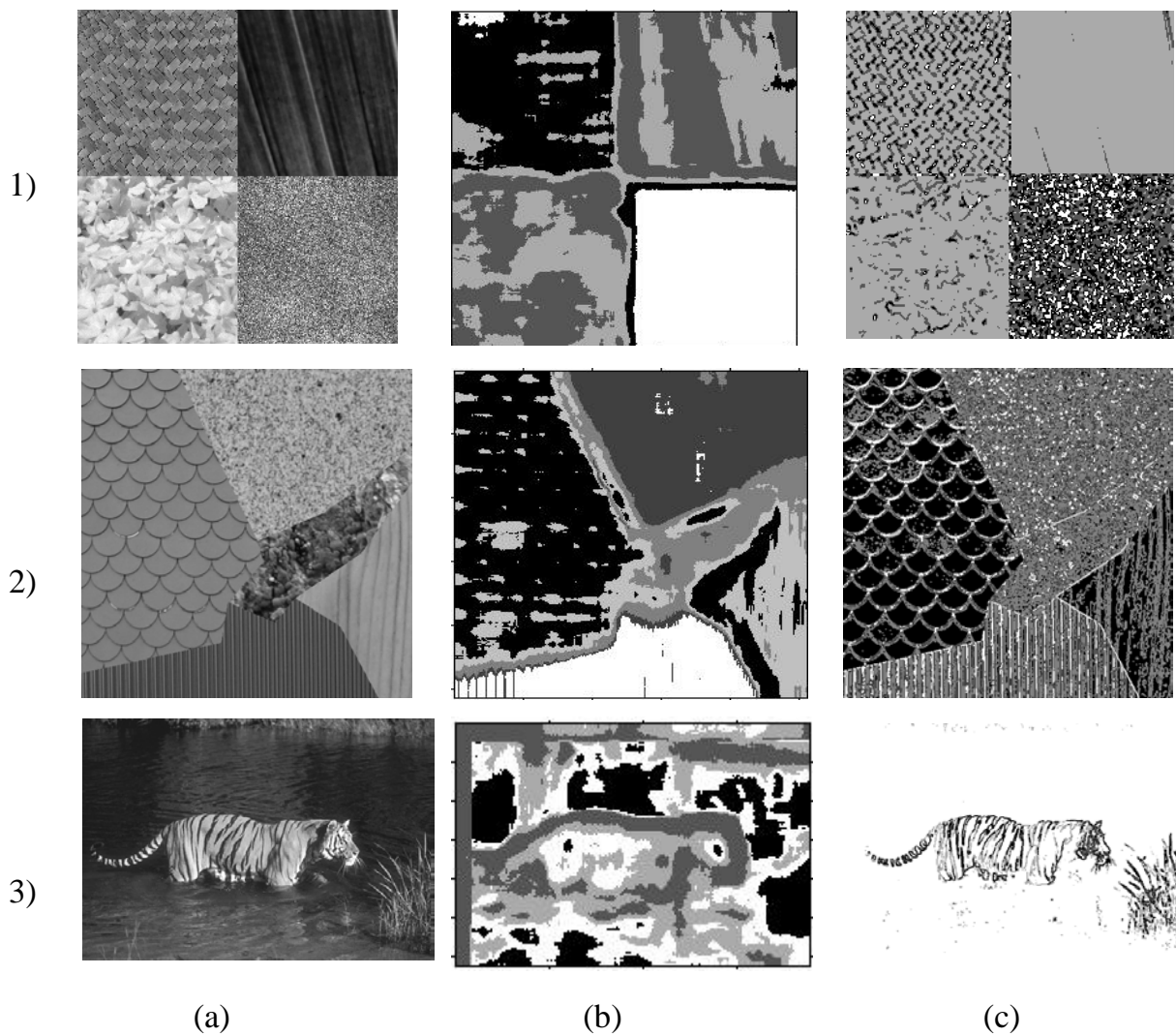


Figure III.16: résultats de la segmentation en utilisant l'ensemble des paramètres et les paramètres de la matrice de cooccurrence

- 1) image synthétique 1, 2) image synthétique 2, 3) image réelle
 a) image d'origine, b) résultat de la segmentation en utilisant l'ensemble des paramètres
 c) résultat de la segmentation en utilisant le paramètre matrice de cooccurrence

En combinant les trois paramètres, intuitivement, nous pouvons penser que les résultats vont s'améliorer, c'est le cas de la figure III.16.2.b. Pour les figure III.16.1.b et III.16.3.b, les résultats sont moins bon que ceux obtenus dans le cas de l'utilisation du paramètre MC tout seul.

IV. Conclusion

Dans ce chapitre, nous avons proposé des paramètres de texture inspirés des paramètres utilisés pour le traitement du signal de la parole qui est un signal monodimensionnel. Nous avons adapté ces paramètres au signal bidimensionnel pour les utiliser dans la segmentation d'images. Pour ce faire nous avons soumis ces paramètres à l'algorithme de classification K-means et nous avons comparé les résultats obtenus avec ceux obtenus en utilisant des paramètres extraits de la matrice de cooccurrence. Les résultats obtenus se sont avérés prometteurs.

Conclusion générale

Le travail présenté dans ce mémoire concerne un des domaines de recherche d'actualité qui est le traitement d'image et plus précisément celui de la segmentation d'images texturées.

Plusieurs types de paramètre de texture existent dans la littérature, dans ce mémoire, nous avons proposé et étudié trois paramètres de texture qui sont le maximum du cepstre, les frontières de passage par zéro et un autre paramètre complémentaire au deuxième qui est le nombre des régions. Après l'extraction de ces paramètres, nous les avons soumis à l'algorithme de classification K-means qui est largement utilisé dans le domaine de la classification et de la segmentation.

Pour rendre compte de l'efficacité de ces paramètres, nous avons réalisé plusieurs tests. Nous avons comparé les résultats de la segmentation obtenus en utilisant les paramètres proposés avec ceux obtenus en utilisant les paramètres de la matrice de cooccurrence. Les résultats obtenus se sont avérés être prometteurs.

Il serait intéressant de poursuivre les testes en comparant avec d'autres paramètres de la texture existants dans la littérature.

Il nous paraît aussi judicieux d'étudier plus profondément l'analyse cepstrale et de réfléchir à d'autres paramètres de texture à extraire du cepstre.

Bibliographie

- [1] S. Amrouni, Recherche et classement d'images dans une base de données de grande taille à partir de requêtes relatives à leurs descripteurs, Mémoire de magister, Université Mouloud Mammeri Tizi-Ouzou
- [2] R. M. Haralick, "Statistical and structural approaches to texture," Proceedings of the IEEE, vol. 67, 1979.
- [3] F. Amroun, Extraction de la composante texturée d'une image, Mémoire de magister, Université Mouloud Mammeri Tizi-Ouzou, 2013
- [4] Al. POREBSKI, Sélection d'attributs de texture couleur pour la classification d'images. Application à l'identification de défauts sur les décors verriers imprimés par sérigraphie, Université de Lille 1, 2009
- [5] L. Houam, contribution à l'analyse de textures de radiographies osseuses pour le diagnostic précoce de l'ostéoporose. Université d'Orléans, Université de Guelma, 2013
- [6] H. Majdoulayne, Extraction de caractéristiques de texture pour la classification d'images, Thèse de doctorat, Université de Toulouse, 2009
- [7] S. Lagab, F. Afettouche, segmentation d'images texturées à base des modèles autorégressifs bidimensionnels, mémoire de fin d'études, Université Mouloud Mammeri Tizi-Ouzou, 2011
- [8] K. Larbi, segmentation d'images basée sur la modélisation statique d'histogrammes, Mémoire de magister, Université Mouloud Mammeri Tizi-Ouzou, 2012.
- [9] N. Rabia, Revue des méthodes de segmentation d'images texturées: cas des images couleur, Mémoire de magister, Université Mouloud Mammeri Tizi-Ouzou, 2011.
- [10] S. Touhami, Classification non supervisée de pixels d'images couleur par analyse d'histogrammes tridimensionnels, Mémoire de DEA, Université D'Angers, 2005.
- [11] M. Lehamel, segmentation d'image texturée à partir des attributs fractals, mémoire de magister, Université Mouloud Mammeri Tizi-Ouzou.
- [12] S. Alkama, Segmentation des images multi-composantes, thèse de doctorat, Ecole nationale polytechnique, 2013.
- [13] P. Berkshire, Survey of Clustering Data Mining Techniques, 2002.
- [14] V. Faber, Clustering and Continuous K-means, Los Alamos Science, 1994.

Bibliographie

- [15] J. MacQueen, Some Methods for Classification and Analysis of Multivariate Observations, Proc.5th Berkeley Symp, 1965, pp. 281-297.
- [16] D. Aouali, K. Amzal, Segmentation d'images texturées par la méthode des matrices de cooccurrence,mémoire de fin d'études, Université Mouloud Mammeri, 2010.
- [17] T. Pellegrini, R. Duee, suivi de voix parlée grave aux modèles de Markov Cachés, Rapport de stage DEA ATIAM, IRCAM Paris, 2003.
- [18] [http://en.wikipedia.org/wiki/ le cepstre](http://en.wikipedia.org/wiki/le_cepstre).
- [19] MIT VisTex, 1995. Texture database, (URL vismod.media [http://.mit.edu/vismod/imagery /VisionTexture/vistex.html](http://.mit.edu/vismod/imagery/VisionTexture/vistex.html)).
- [20] <http://masaic.utia.cas.cz>.
- [21] Berkeley Segmentation Dataset. URL [http://www.eecs.berkeley.edu /Research/Projects/ CS/vision/bsds/BSDS300/html](http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html).

Résumé

Dans ce travail, nous avons proposé trois types de paramètres de texture extraits de signaux bidimensionnels tels que Maximum du cepstre, Frontière des passages par zéro, et le Nombre de régions. Par la suite nous avons soumis ces paramètres à l'algorithme de classification K-means qui est largement utilisé dans ce domaine.

Pour rendre compte de l'efficacité de ces paramètres, nous avons réalisé plusieurs tests et nous avons comparé avec les résultats de la segmentation obtenus en utilisant les paramètres de la matrice de cooccurrence.

Mots clés :

Analyse de textures, Segmentation par division de région, Segmentation par division-fusion de région, Classification supervisée, Classification non supervisée, algorithme K-means.