

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOU MAMMERI, TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de fin d'études
Présenté en vue de l'obtention
du Diplôme d'Ingénieur d'Etat d'automatique

Thème:
*Méthode d'analyse de la texture basée sur les
parcours optimaux, matricés de concurrence
et filtres de gabor*

Réalisé par :
Ousmail Faiz
Guerchouche said

Proposé et dirigé
k.Hammouche

Promotion 2008

Table des matières

Introduction générale	01
Chapitre I Notions sur la texture d'images	
I.1 Introduction.....	03
I.2 Définition de la texture	03
I.3 Quelques exemples des textures naturelles.....	05
I.4 Propriétés qualitatives de la texture.....	07
I.5 La régularité.....	09
I.6 Analyse et synthèse de la texture.....	10
I.6.1 Définition.....	10
I.6.2 Synthèse de la texture.....	11
I.6.3 Analyse de la texture.....	11
I.7 Méthodes d'analyse de la texture.....	12
I.7.1 Méthodes structurelles.....	12
I.7.2 Méthodes statistiques.....	12
I.7.3 Méthodes spatio-fréquentielles.....	13
I.7.4 Méthodes fractales.....	13
I.8 La méthode basée sur les matrices de cooccurrences.....	13
I.9 Analyse de la texture basée sur les filtres de Gabor.....	19
I.9.1 Rappels théoriques.....	19
I.9.1.1 Transformée de Fourier.....	19
I.9.1.2 Produit de convolution	20
I.9.1.3 Filtrage d'une image.....	21
I.9.2 Filtres de Gabor.....	21
I.10 Calcul des caractéristiques d'une texture par les filtres de Gabor.....	27
I.10.1 Calcul d'énergie.....	28
I.10.2 Calcul de la moyenne et de la variance.....	28
I.11 Domaines d'applications d'analyse de la texture.....	29
I.12 Conclusion.....	30
Chapitre II Segmentation d'images texturées	
II.1 Introduction.....	31
II.2 La segmentation	31
II.2.1 Définition de la segmentation.....	32
II.2.2 Méthodes de la segmentation.....	32
II.2.2.1 Segmentation en contours.....	32

II.2.2.1.1	L'approche Gradient...	33
II.2.2.1.2	L'approche Laplacienne.....	35
II.2.2.2	Segmentation en régions.....	37
II.2.2.2.1	Segmentation par croissance de régions.....	38
II.2.2.2.2	Segmentation par division.....	39
II.2.2.2.3	Segmentation par fusion.....	39
II.2.2.2.4	Segmentation par division/fusion.....	40
II.2.2.2.5	Segmentation basée sur la classification des pixels.....	41
II.2.2.2.5.1	Le seuillage.....	41
II.2.2.2.5.2	Définition de la classification.....	42
II.2.2.2.5.3	L'algorithme de C-Means.....	44
II.2.2.2.5.4	Classification floue.....	46
II.2.2.2.5.4.1	Introduction à la notion floue	46
II.2.2.2.5.5	L'algorithme de C-Means flu.....	47
II.3	Conclusion.....	50
Chapitre III Segmentation d'images texturées par l'analyse des motifs		
III.1	Introduction	51
III.2	Parcours locaux.....	51
III.3	Parcours optimal.....	53
III.4	Image de motifs.....	54
III.4.1	Matrice de cooccurrence des motifs (MCM).....	55
III.4.2	Propriétés de MCM.....	56
III.4.3.a	Invariance au changement de la luminance.....	56
III.4.3.b	Invariance à la translation.....	56
III.5	Description d'une image par les valeurs des parcours optimaux (VPO).....	58
III.5.1	Matrice des cooccurrences des VPO.....	58
III.5.2	Filtres de Gabor des VPO.....	58
III.6	Conclusion.....	59
Chapitre IV Tests et résultats		
IV.1	Introduction.....	60
IV.2	Création d'une base d'images.....	60
IV.3	Images motifs et images VPO.....	61
IV.4	Comparaison des attributs de texture extraits à partir des matrices de cooccurrence....	64
IV.5	Comparaison des attributs de texture issus des filtres de Gabor.....	66
IV.6	Segmentation.....	72

Conclusion générale.....	73
Bibliographie.	

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

Le cœur d'un système d'analyse automatique d'images est l'étape d'extraction des caractéristiques et celle de segmentation. Cette dernière constitue une opération de traitement d'images qui a pour but de rassembler des pixels ayant les mêmes caractéristiques. Les pixels sont ainsi regroupés en régions homogènes, mettant ainsi en évidence les différents objets que contient l'image.

L'étape d'extraction des caractéristiques consiste à décrire l'image entière ou un pixel de l'image par un ensemble d'attributs. Pour une image naturelle, les attributs de texture sont très indiqués pour ressortir ces propriétés.

Parmi toute une panoplie de méthodes d'analyses de la texture on retrouve les matrices de cooccurrence et les filtres de Gabor.

Notre travail, a pour objectif, l'analyse des images texturées basées sur les parcours optimaux par les filtres Gabor et les matrices de cooccurrence. Pour cela, nous avons réparti notre travail en quatre chapitres.

Le **chapitre I** est consacré à la représentation de quelques notions sur la texture d'images. Deux approches d'analyse de la texture qui sont la méthode basée sur les matrices de cooccurrence (approche statistique) et celle basée sur les filtres de Gabor (approche spatio-fréquentielle) seront particulièrement détaillées.

Le **chapitre II** propose des rappels sur la segmentation d'une image. Nous mettons l'accent sur les méthodes les plus utilisées pour la segmentation d'une image en portant un intérêt particulier sur l'approche de segmentation basée sur la classification des pixels en illustrant deux algorithmes de classification des pixels qui sont C-Means et Fuzzy C-Means.

Dans le **chapitre III**, nous avons présenté une nouvelle manière de quantifier les niveaux de gris de pixels voisins, qui est basée sur la disposition des pixels appartenant à un

même voisinage, ce qui engendre différents parcours locaux ou parcours optimaux, qui sont décrit par des parcours où motifs.

L'efficacité des attributs de texture extraits, par les deux techniques d'analyse de la texture présentées dans le chapitre I, sur les images transformées sera étudiée dans le **chapitre IV**. Nous allons essayer de montrer par la suite, que l'application de ces deux techniques sur l'image motif ou VPO présente bien des avantages par rapport a l'image originale.

I.1 Introduction

L'analyse de la texture est une étape primordiale dans le traitement d'une image numérique. Avant de procéder à la segmentation d'une image ou la reconnaissance de ses régions, il faut d'abord que le traiteur d'images analyse et isole les textures de cette image.

Dans ce présent chapitre, on définit la texture ainsi que les méthodes avec lesquelles on l'analyse et bien sûr son utilité et ses applications.

I.2 Définitions de la texture [1]

Dans le domaine du traitement de l'image et de la vision artificielle, il n'existe pas de définition satisfaisante de la texture. Les définitions mathématiques construites à partir de propriétés statistiques sont soit trop générales et imprécises soit restrictives pour s'adapter à la diversité des cas rencontrés. La définition la plus courante s'appuie sur une constatation expérimentale : « une texture est un champ de l'image qui apparaît comme un domaine cohérent et homogène. C'est-à-dire formant un tout pour un observateur. C'est cette propriété de cohérence de la texture placés dans son contexte d'être perçu comme un tout homogène par l'œil humaine qui sera recherché le plus souvent par le traiteur des images, dans le but d'isoler les textures, soit pour segmenter l'image, soit pour reconnaître des images.

La définition littéraire de la texture 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. La notion de texture est utilisée pour traduire un aspect homogène de la surface d'un objet sur une image. La texture se manifeste donc par une information visuelle qui permet de la décrire qualitativement à l'aide des adjectifs suivants: grossière, fine, lisse, tachetée, granuleuse, marbrée, régulière ou irrégulière.

La texture est une caractéristique importante de l'apparence des objets dans des scènes réelles et la comprendre est une partie essentielle de la compréhension de la vision humaine. Une texture représente, à une échelle donnée, le même aspect quelle que soit la zone observée. Dans ces conditions, on considère l'image comme la réalisation d'un processus stochastique *local* et *stationnaire*. C'est-à-dire que chaque pixel est caractérisé par un petit voisinage et que sa caractérisation est la même pour tous les pixels de l'image (Figure I.1).

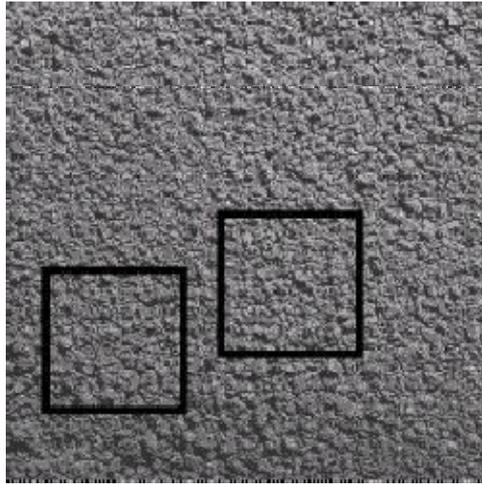


Figure I.1 : Une texture est stationnaire et locale: lorsqu'on déplace une certaine fenêtre au long de l'image, son contenu est perçu comme étant le même partout dans l'image

On rencontre deux types de définitions de la texture:

- La première est *déterministe* et fait référence à une répétition spatiale d'un motif de base dans différentes directions. Cette approche structurale correspond à une vision *macroscopique* des textures.
- La deuxième est *probabiliste* et cherche à caractériser l'aspect anarchique et homogène qui ne comprend ni de motif localisable, ni de fréquence de répétition principale.

HARALICK 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 desquels est formée la texture ; la deuxième dimension est relative à la description de l'organisation spatiale de ces primitives.

UNSER présente la texture comme une structure disposant de certaines propriétés spatiales homogènes et invariantes par translation. Une autre approche serait encore de définir la texture à partir de deux types d'informations essentielles que comporte l'image:

- Les contours , de type monodimensionnel , qui marquent les frontières entre régions homogènes :
- L'aspect de surface, de type bidimensionnel, qui définit les régions homogènes.

Cependant, la description d'une texture peut s'avérer erronée à une autre échelle d'observation, c'est à dire en changeant la résolution.

En pratique, on distingue deux grandes classes de textures, qui correspondent à deux niveaux de perception :

- Les macrotextures qui présentent un aspect régulier, sous formes de motifs répétitifs spatialement placés selon une règle précise (ex: peau de lézard, mur de brique). Elles nécessitent ainsi une approche structurelle déterministe pour son analyse.
- Les microtextures présentant des primitives "microscopiques" distribuées de manière aléatoire (ex: sable, laine tissée, herbe) d'ou une approche probabiliste cherchant à caractériser l'aspect anarchique et homogène.

GAGALOWICZ propose une synthèse des deux approches en considérant la texture comme « une structure spatiale constituée de l'organisation de primitives ayant chacune un aspect aléatoire, donc une structure hiérarchique a deux niveaux ».

De nombreuses études psycho-visuelles ont été faites sur la discrimination de la texture par le système visuel humain. Une conjecture importante et valide dans beaucoup de cas est que l'œil humain ne peut discerner instantanément deux textures dont les statistiques du second ordre sont identiques. Cependant, il existe des cas ou des textures ayant les mêmes statistiques du second ordre sont néanmoins discriminables sur la base de propriétés locales.

I.3 Quelques exemples des textures naturelles [1], [2]

Il existe une infinité des exemples de la texture naturelle (Figure I.2). La figure I.3 illustre la diversité des textures tirées de l'ouvrage de Brodatz, qui est reconnu aujourd'hui comme un ouvrage de référence. Il offre une collection des textures naturelles qui constituent d'excellents exemples et sont souvent utilisées pour tester les algorithmes et les méthodes d'analyse de la texture.

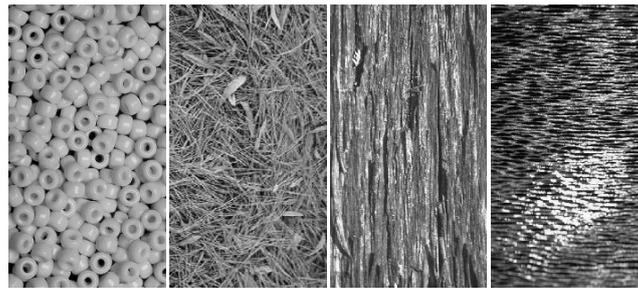


Figure I.2 : Quelques exemples des textures naturelles : Perles, tapis d'aiguilles de pin et de feuilles, écorce d'arbre, poil court.

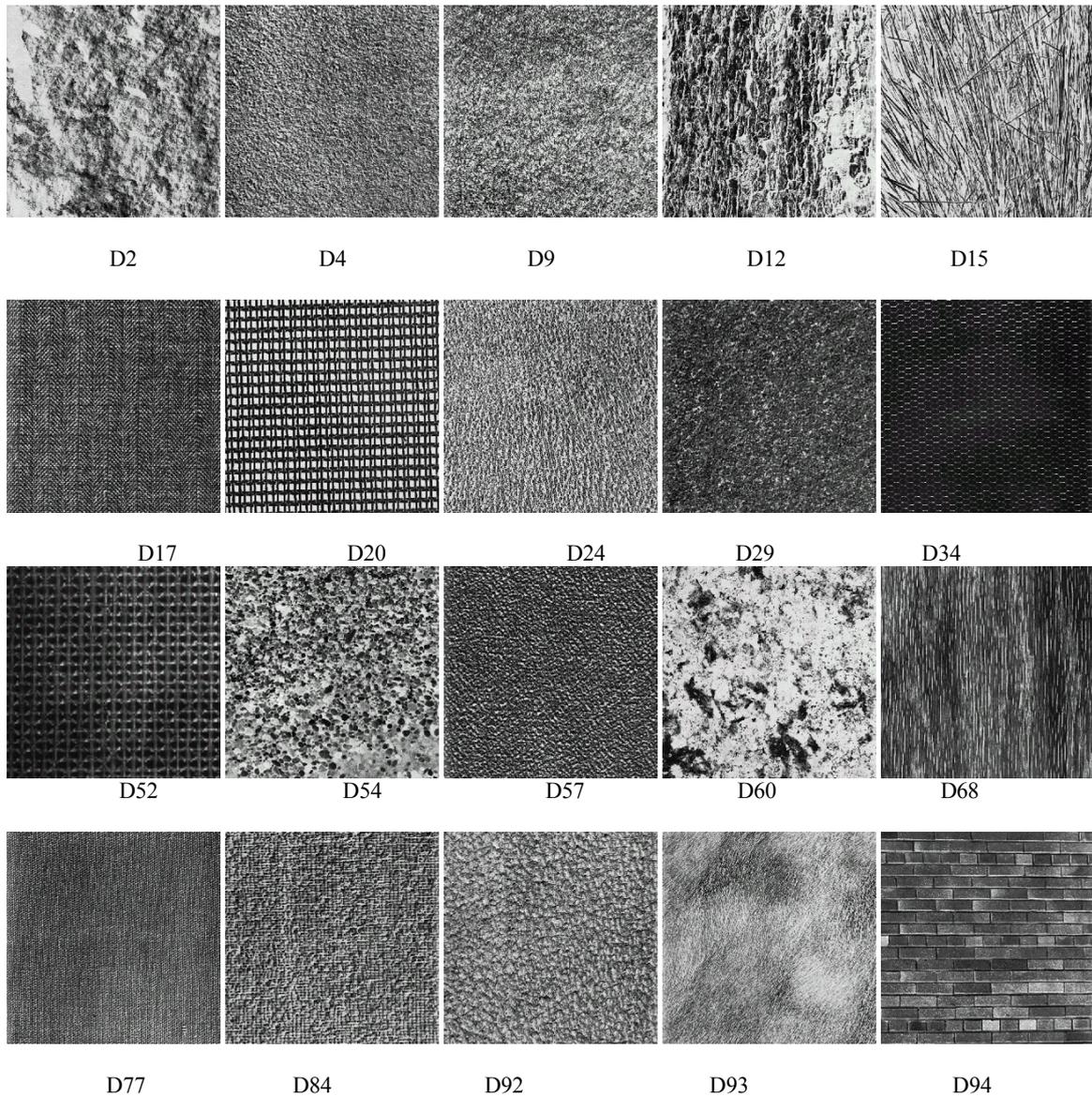


Figure I.3 : Images texturées naturelles de Brodatz;

D2 : pierre de champ(field stone), D4 : liège pressé(pressed cork), D9 : pelouse d'herbe(lawn grass), D12 : écorce (bark), D15 : paille(straw), D17 : arête tissée de hareng(herringbone weave), D20 :toile française(french canvas), D24 :cuire de veau pressé(pressed calf leather), D29 :sable de plage(beach sand), D34 :filet (netting), D52 :tissu de paille oriental(oriental straw cloth), D54 :cailloux de plage(beach pebbles), D57 :papier fait à la main (handmade paper), D60 :marbre européen (european marble), D68 : bois (wood), D77 : coton (cotton), D84 : raphia (raffia), D92 : peau de porc (pigskin), D93 : fourrure (fur), D94 :brique (brick).

I.4 Propriétés qualitatives de la texture [2]

Comme nous l'avons évoqué précédemment, on cherche à décrire la texture d'une manière courante par des adjectifs qualitatifs. Parmi eux, on distingue :

La complexité : Une texture complexe possède plusieurs types de primitives (groupe D23, D4, D9 de la figure I.4). Dans ce cas le contenu d'informations présent dans la texture est important.

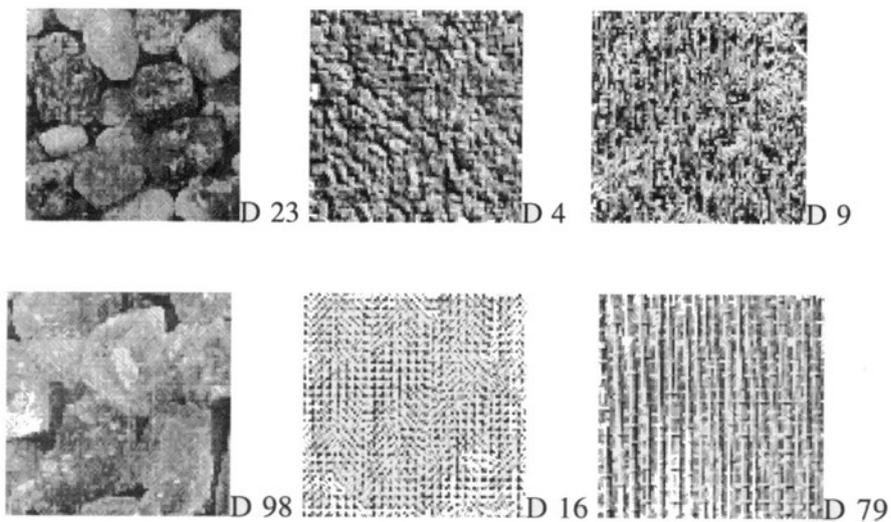


Figure I.4: Présentation du paramètre "complexité" sur différentes textures.

Le contraste : Une texture possède un contraste élevé si les différences d'intensité entre primitives sont importantes (groupe D27, D23 de la figure I.5).

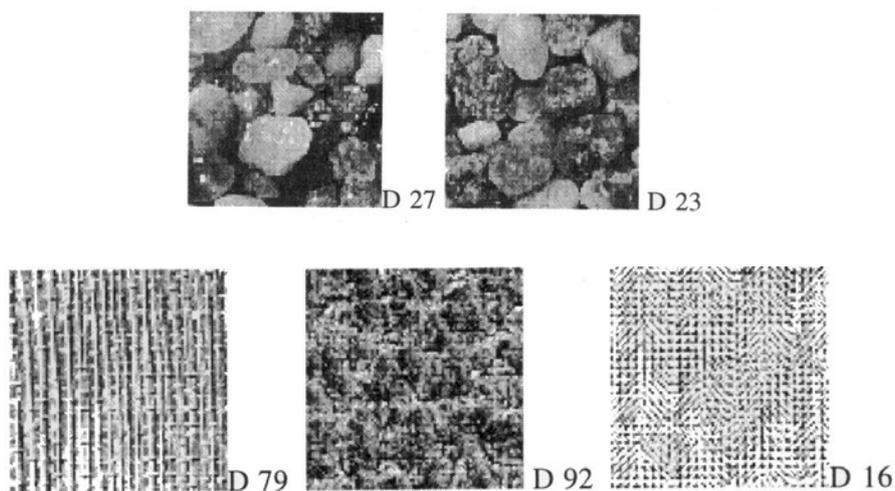


Figure I.5 : Présentation du paramètre "contraste" sur différentes textures.

La grossièreté : Une texture grossière (groupe D98, D23, D27) possède des primitives larges (figure I.6). Il existe alors peu de variations entre l'intensité d'un pixel et celle de ses pixels voisins.

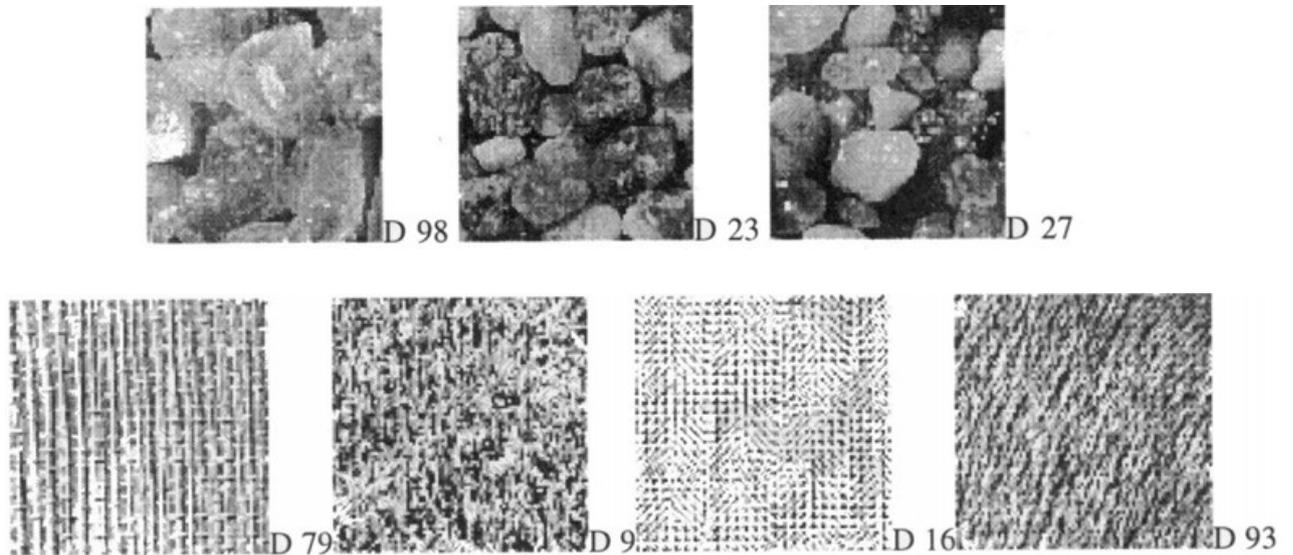


Figure I.6 : Présentation du paramètre "grossièreté" sur différentes textures.

La force : Plus la force est élevée et plus les primitives sont facilement définissables et visibles (groupe D23,D98,D27,D56 de la figure I.7).

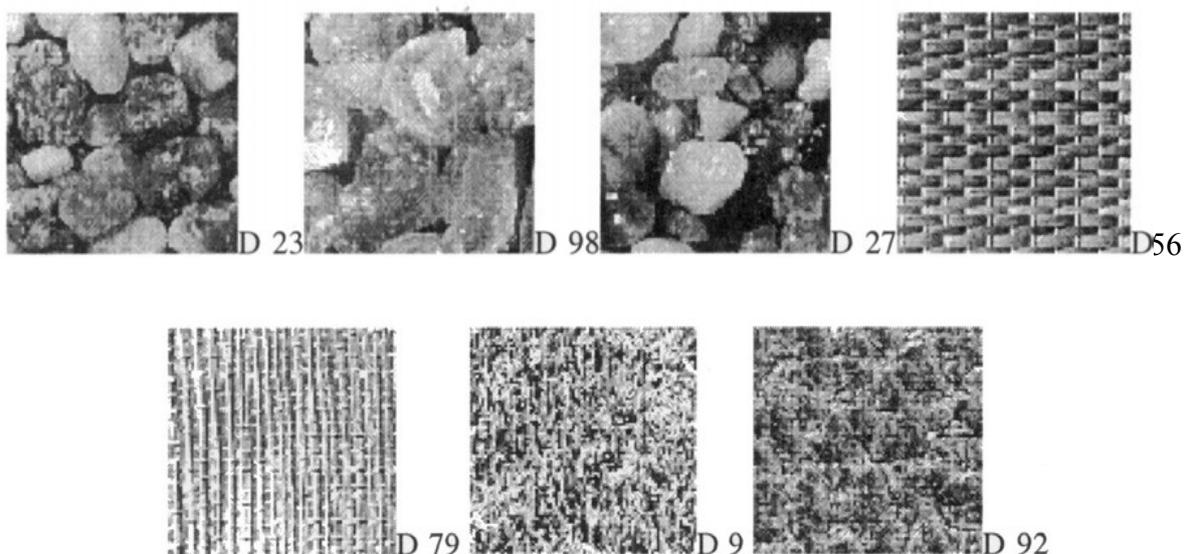


Figure I.7 : Présentation du paramètre "force" sur différentes textures.

I.5 La régularité [1],[2]

La texture est peut être distribuée d'une façon aléatoire ou d'une façon régulière. On distingue naturellement deux modèles de distribution de la texture :

- La texture régulière dans laquelle la périodicité de motif est connue (répétition d'un motif de base) exemples: tissus, mur de brique...etc. (Figure I.8)
- La texture aléatoire pour laquelle la distribution des intensités n'est l'objet d'aucune régularité apparente (pas de motif de base) exemples : herbe nuage, foule...etc. (Figure I.9)

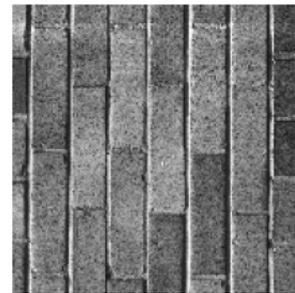
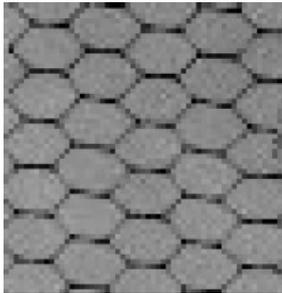


Figure I.8 : Exemples de texture régulière (périodique).

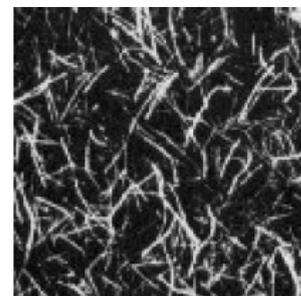
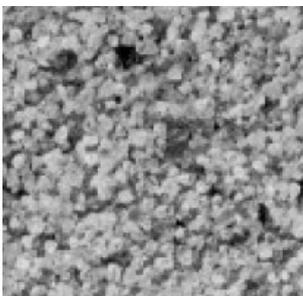


Figure I.9 : Exemple de texture aléatoire (non périodique).

La première famille sera bien décrite par des approches fréquentielles ou des approches structurales dans lesquelles on associera un motif et des règles de placement sur un pavage régulier.

La seconde approche sera plutôt décrite par les lois statistiques, des moments, une description spectrale en termes de densité de puissance, des propriétés de corrélation ou d'isotropie.

Il n'existe pas réellement une texture strictement périodique ni totalement aléatoire. Les deux modèles se complètent naturellement. L'une des difficultés est de les mêler dans des modèles capables de s'adapter à la variabilité des textures étudiées.

I.6 Analyse et synthèse de la texture [1], [3], [4]

L'analyse et la synthèse de texture est, depuis trois décennies, un thème actif de recherche ayant proposé de nombreuses méthodes. Le premier modèle de texture a été développé par Julesz dans les années 60. Il suggérait que la perception de la texture pouvait être expliquée par l'extraction des statistiques d'ordre k , c'est-à-dire des statistiques de cooccurrence pour les intensités des k -tuples de pixels. Les premiers travaux dans ce domaine ont utilisé cette conjecture, mais le problème rencontré était que la taille des données contenues dans les statistiques d'ordre k était immense et qu'il était très difficile de travailler pour $k > 2$. Cette théorie nous amène à une définition mathématique de la texture, en termes *d'ensemble de Julesz*.

I.6.1 Définition

Un ensemble de Julesz est un ensemble d'images qui partagent les mêmes valeurs statistiques sur certaines distributions de probabilité. Les images dans l'ensemble de Julesz sont définies sur des réseaux infinis (une idéalisation du point de vue mathématique étant $\mathbb{Z} \times \mathbb{Z}$).

D'autre part, les expériences neurophysiologiques montrent que le système visuel humain extrait des statistiques d'ordre plus grand que 2. Les tendances qui semblent s'imposer et qui permettent à l'heure actuelle de synthétiser la majorité des textures se basent sur deux approches. La première est la théorie des images filtrées, qui a été inspirée par le mécanisme de filtrage multicanaux découvert et accepté en neurophysiologie. Ce mécanisme suggère que le système visuel décompose l'image rétinienne dans un ensemble de sous-bandes, obtenues par la convolution de l'image avec un banc de filtres linéaires, suivie de certaines opérations non-linéaires. Cette théorie de filtrage inclut les filtres de Gabor et les pyramides en ondelettes. Elle montre des résultats remarquables dans la classification et la segmentation des images.

La seconde tendance est la modélisation statistique, qui définit les textures comme étant des réalisations de certaines lois de probabilité sur des champs aléatoires. Cela inclut des modèles de chaînes de Markov, des modèles de champs Markoviens aléatoires. Ces modèles prennent en compte seulement un nombre petit de paramètres et permettent une représentation concise des textures. En plus ils transforment la synthèse de textures en un problème bien posé. Les modèles statistiques permettent non seulement de trouver les

paramètres caractérisant les textures, mais aussi de synthétiser ces images de textures par l'échantillonnage des probabilités obtenues.

I.6.2 Synthèse de la texture

Le problème de la synthèse des textures peut être énoncé comme suit: étant donné un échantillon fini d'une texture (une image), on veut synthétiser de nouvelles images de la même texture. Elles doivent être suffisamment différentes par rapport à l'image d'origine, mais doivent sembler provenir du même processus stochastique.

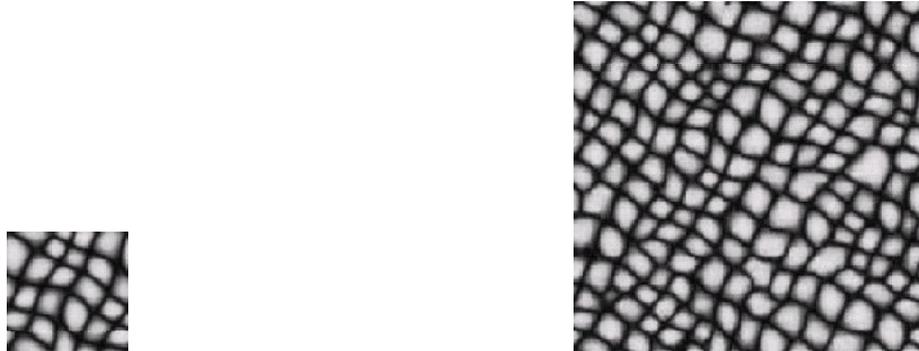


Figure I.10 : Génération de texture : on part d'une image échantillon et on veut générer une autre image ayant la même apparence texturale.

La synthèse de texture doit prendre en compte deux aspects:

- L'image obtenue doit suivre la même distribution de probabilité que l'image échantillon.
- Elle doit être différente de l'image d'origine au sens où l'on ne veut pas faire du copier-coller.

I.6.3 Analyse de la texture [1], [3]

L'analyse est un domaine très important dans le traitement de l'image. Les principales informations dans l'interprétation du message visuel pour un observateur humain sont les contours et/ou les textures. L'analyse de l'image consiste souvent à extraire un certain nombre de propriétés caractéristiques et à les exprimer sous forme paramétrique. L'étape d'extraction des paramètres précède souvent une étape de décision de manière à pouvoir répondre à des questions telles que: un matériau s'il est normal ou défectueux, un tissu s'il est biologique sain ou pathologique, quels sont les types de défauts...etc. Les

paramètres calculés permettent donc de décrire, de caractériser, de segmenter et d'analyser les images en question. Selon le cas, l'analyse peut être globale ou locale, la notion de localité prenant toute son importance avec la complexité de l'image. Il est évident que le choix des paramètres dépend surtout de l'application considérée, par exemple de lier ces paramètres avec les propriétés physiques et biologiques réelles afin de les quantifier ou alors de trouver des similitudes avec des textures de référence afin de les identifier. L'interprétation des informations dans un environnement naturel n'est pas un problème simple. En effet, les textures naturelles sont très irrégulières et ne peuvent être modélisées précisément par les techniques mathématiques actuelles.

I.7 Méthodes d'analyse de la texture

Il existe plusieurs méthodes pour analyser la texture :

I.7.1 Méthodes structurelles

Les méthodes structurelles 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.

I.7.2 Méthodes statistiques

Du point de vue des méthodes statistiques, la texture est considérée comme la réalisation d'un processus stochastique stationnaire. Des paramètres statistiques sont estimés pour chaque pixel de l'image.

Suivant la modalité des images à étudier, la signature la plus discriminante de la texture est à rechercher soit dans des méthodes qui exploitent directement les propriétés statistiques de la texture (matrices de cooccurrences, matrice de longueurs de plages, matrice de voisinage, fonction d'auto-corrélation, modèle de Markov, modèle autorégressif, modèles issues de la morphologie mathématique), soit dans des méthodes qui exploitent les propriétés statistiques à partir d'un plan transformé dans lequel on réécrit l'image de texture (densité spectrale, méthode des extrema locaux, méthodes de transformation de Fourier, Karhunen Loeve, Walsh Hadamard, Slant ou avec des filtres numériques).

I.7.3 Méthodes spatio-fréquentielles

Dans les méthodes spatio-fréquentielles, les représentations spatio-fréquentielles préservent à la fois les informations globales et locales donc elles sont bien adaptées aux signaux quasi périodiques. En effet, les textures sont des signaux quasi périodiques qui ont une énergie fréquentielle localisée.

I.7.4 Méthodes fractales

Du point de vue de la synthèse d'image, les méthodes fractales sont à part car elles permettent de synthétiser des images très proches de la réalité. En analyse de texture, la dimension fractale, qui est une mesure du degré d'irrégularité d'un objet, décrit une certaine propriété de la texture. Le modèle fractal est basé essentiellement sur l'estimation par des méthodes spatiales de la dimension fractale de la surface représentant les niveaux de gris de l'image.

Le but de l'analyse de texture par ces méthodes est de formaliser les descriptifs de la texture par des paramètres mathématiques qui serviraient à l'identifier. Dans ce sens, les critères visuels qui ont été retenues pour la texture sont: le contraste, la granularité, l'orientation, la forme, la finesse, la régularité et la rugosité.

Nous allons à présent présenter deux méthodes très populaires à savoir, la méthode basée sur les matrices de cooccurrence et celle basée sur les filtres de Gabor.

I.8 La méthode basée sur les matrices de cooccurrence [2],[3]

Les techniques en matière d'analyse de texture sont nombreuses. Parmi elles, la plus populaire, et sans doute la plus efficace, est la méthode basée sur les matrices des cooccurrences.

Elle demeure la méthode de référence dans le domaine de l'analyse de la texture. Cette méthode repose sur la stationnarité de l'image en loi conjointe, c'est-à-dire que la probabilité conjointe de niveaux i et j de pixels éloignés selon une translation T est un invariant qui caractérise donc la texture de l'image. La méthode des matrices de cooccurrence exploite les relations qui existent entre les pixels de l'image dans une direction θ et une distance donnée d . Il s'agit en fait de la probabilité de trouver, si on part d'un pixel de valeur i , un pixel de valeur j , quand on se déplace en suivant un vecteur dans

la direction θ d'une distance d . Les matrices de cooccurrence sont alors définies à partir d'une région R de l'image I et pour une translation T par :

$$P_T(i, j) = \frac{1}{N_c} \sum_R \delta(V_i(t), i) \cdot \delta(V_i(t + T), j) \tag{I.1}$$

Où $V_i(t)$ définit l'intensité du pixel d'indice (t) de l'image I , $\delta(i, j)$ représente le symbole de Kronecker et N_c est le nombre de couple $(t, t+T)$ de la région tel que :

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

La méthode des matrices de cooccurrence (ou méthode de dépendance spatiale des niveaux de gris), permet de déterminer la fréquence d'apparition d'un "motif" formé de deux pixels séparés par une certaine distance d dans une direction particulière par rapport à l'horizontale. Afin de limiter le nombre de calculs, on prend généralement comme valeurs $0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ$ et 1 pour la valeur de d .

La taille de la matrice est $N_g \times N_g$, où N_g correspond au maximum des niveaux de gris de l'image. Pour ne pas avoir une taille de matrice trop élevée, on réduit le plus souvent N_g à 8, 16 ou 32. A titre d'exemple, pour une image possédant 5 niveaux, la matrice de cooccurrence sera de taille 16 (Figure I.11).

1	4	4	3
4	2	3	2
1	2	1	4
1	2	2	3

Image initiale

0	2	0	2
1	1	2	0
0	1	0	0
0	1	1	1

matrice de cooccurrence associée ($d=1, \theta=0$)

Figure I.11 : Exemple de construction d'une matrice de cooccurrence.

A chaque direction θ et pour chaque valeur de d correspond une matrice de cooccurrence $P(d, \theta)$.

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

$$P(d, 0^\circ) = P^T(d, 180^\circ)$$

$$P(d, 45^\circ) = P^T(d, 225^\circ)$$

$$P(d, 90^\circ) = P^T(d, 270^\circ)$$

$$P(d, 135^\circ) = P^T(d, 315^\circ)$$

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

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

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 la matrice de cooccurrence, il est possible d'extraire plusieurs d'attributs. Ces attributs contiennent des informations sur la finesse, la directionnalité et la granularité de la texture.

Parmi ces attributs, on retrouve :

➤ **La moyenne**

$$Moy = \sum_i \sum_j P(i, j) \quad (I.3)$$

➤ **La variance**

$$VAR = \sum_i \sum_j (i - Moy)^2 P(i, j) \quad (I.4)$$

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

➤ **L'énergie ou second moment angulaire d'ordre deux ou uniformité**

Ce paramètre 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. Il est donné par la formule suivante :

$$E = \sum_{i,j} (P(i,j))^2 \quad (I.5)$$

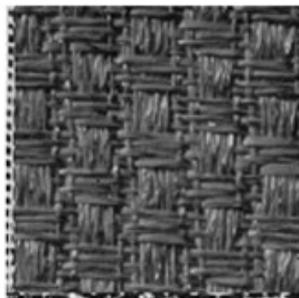
➤ **Le contraste (inertie)**

Il 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.

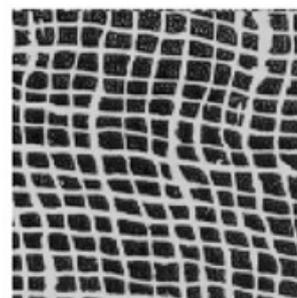
Ce paramètre permet aussi de caractériser la dispersion des valeurs de la matrice par rapport à sa diagonale principale. Il est formulé comme suit :

$$CONT = \sum_i \sum_j (i - j)^2 P(i,j) \quad (I.6)$$

La figure I.12 montre les valeurs du contraste pour deux textures différentes.



CONT=120.17

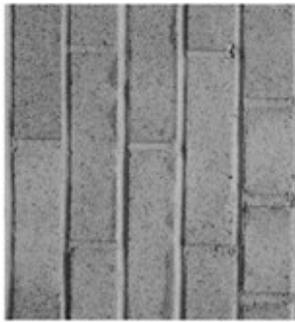


CONT=625.81

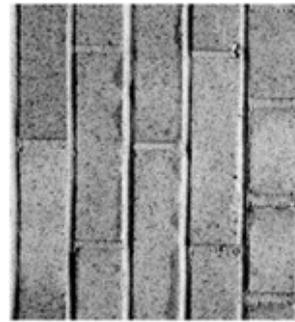
Figure I.12 : Illustration du contraste sur deux textures différentes.

La texture de droite possède plus de transitions que l'image de gauche et donc moins de régions homogènes. Son contraste est plus élevé.

La figure (I.13) montre qu'une texture apparaît plus nette lorsque le contraste est plus élevé, c'est à dire lorsque les transitions entre niveaux de gris sont plus visibles.



CONT=41.663



CONT=142.288

Figure I.13 : Illustration du contraste sur une même texture.

➤ **La corrélation**

Elle est définie de la manière suivante :

$$COR = \sum_i \sum_j (i - \mu_x)(j - \mu_y)p(i, j) \quad (I.7)$$

Où μ_x et μ_y représentent les moyennes respectivement des lignes et des colonnes de la matrice tels que :

$$\mu_x = \frac{1}{N_c} \sum_i \sum_j i p(i, j).$$

$$\mu_y = \frac{1}{N_c} \sum_i \sum_j j p(i, j).$$

➤ **La corrélation normalisée**

Elle permet de déterminer si certaines colonnes de la matrice sont égales. Plus les valeurs sont uniformément distribuées dans la matrice et plus la corrélation est importante. Elle est donnée par la formule suivante :

$$COR = \sum_{i,j} \frac{ijP(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \quad (I.8.a)$$

Où σ_x et σ_y représentent les écarts types respectivement des lignes et des colonnes de la matrice.

Pratiquement , on peut utiliser l'autocorrélation :

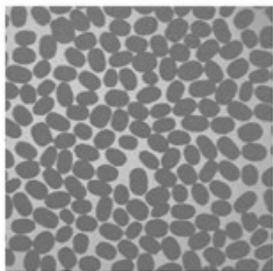
$$AUT = \sum_{i,j} ij P(i, j) \quad (I.8.b)$$

➤ **L'entropie**

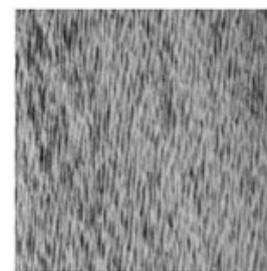
Cet attribut mesure la complexité de l'image. Lorsque les valeurs de la matrice sont presque toutes égales, l'entropie est élevée. Elle s'exprime comme suit :

$$ENT = \sum_i \sum_j P(i,j) \log P(i,j) \quad (I.9)$$

Elle permet de caractériser le degré de granulation de l'image. Plus l'entropie est élevée et plus la granulation est grossière. Comme le montre la figure I.14 où la texture de droite semble "moins lisse" que la texture de gauche.



ENT = 6.7



ENT= 4.74

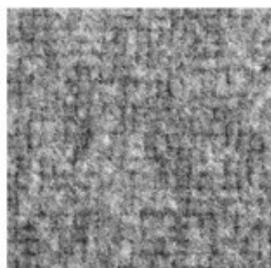
Figure I.14 : Illustration de l'entropie.

➤ **Moment différentiel inverse**

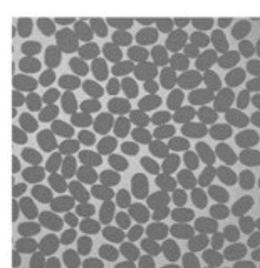
Il permet de mesurer l'homogénéité de la texture. Son expression est :

$$HOM = \sum_{i,j} \frac{1}{1+(i-j)^2} P(i,j) \quad (1.10)$$

Sur la figure (I.15), on voit que la texture de droite possède plus de régions homogènes que la texture de gauche, son paramètre HOM est plus élevé.



HOM= 0.0021



HOM= 0.03

Figure I.15 : Illustration de l'homogénéité.

On peut aussi définir d'autres attributs comme :

➤ **Le moment diagonal** $DIAG = \sum_i \sum_j \left(\frac{1}{2}|i - j|P(i, j)\right)^{1/2}$ (I.11)

➤ **Le "cluster shade"** $C SHADE = \sum_i \sum_j (i + j - 2Moy)^3 P(i, j)$ (I.12)

➤ **Le "cluster prominence"** $C PROM = \sum_i \sum_j (i + j - 2Moy)^4 P(i, j)$ (I.13)

I.9 Analyse de la texture basée sur les filtres de Gabor [6], [7], [8]

Un filtre de Gabor, du nom du physicien anglais d'origine hongroise Dennis Gabor, est un filtre linéaire. Il est défini par le produit entre une gaussienne et une sinusoïdale orientée, C'est un filtre passe bande avec une enveloppe gaussienne. La fonction sinusoïdale est caractérisée par sa fréquence et par son orientation. Les filtres de Gabor sont considérés comme des filtres précurseurs dans le domaine des méthodes de filtrage spatio-fréquentiel. Ils présentent d'excellentes propriétés de localisation fréquentielles et spatiales. En plus, des études physiologiques sur les mammifères ont montré qu'on peut assimiler le fonctionnement de certains neurones du cortex visuel à ce type de filtre. Leur efficacité dans le domaine de l'analyse de la texture a été montrée dans plusieurs travaux. Turner a été le premier en 1986 à montrer que les filtres de Gabor peuvent décrire des informations de texture. Il a également montré qu'en utilisant des images d'énergie en sortie des filtres, des différences importantes sur des filtres choisis spécifiquement pouvaient être utilisées pour réaliser une segmentation.

I.9.1 Rappels théoriques

I.9.1.1 Transformée de Fourier

La transformée de Fourier est un outil mathématique qui permet de représenter un signal dans le domaine fréquentiel. Il est défini comme suit :

- Pour un signal 1-D

$$F(V) = \int_{-\infty}^{+\infty} f(t)e^{-j2\pi Vt} dt \quad (I.14)$$

- pour un signal 2-D

$$F(V_1, V_2) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) e^{-j2\pi V_1 x} e^{-j2\pi V_2 y} dx dy \quad (I.15)$$

La transformée de Fourier discrète d'une image $f(x,y)$ de taille $(N \times M)$ est donnée par la formule suivante :

$$F(n, m) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) e^{-j2\pi nx/M} e^{-j2\pi my/N} \quad (I.16)$$

$$0 \leq n \leq N-1, 0 \leq m \leq M-1.$$

I.9.1.2 Produit de convolution

Le produit de convolution sert à définir la relation entrée/sortie dans un filtrage linéaire. Il permet de déduire le signal de sortie d'un système linéaire défini par sa réponse impulsionnelle $h(t)$ et son entrée $e(t)$. Il est défini comme suit :

$$S(t) = e(t) \otimes h(t) \quad (I.17)$$

➤ Pour un signal 1-D

$$s(t) = \int_{-\infty}^{+\infty} e(u)h(t-u)du \quad (I.18)$$

➤ Dans le cas d'un signal 2-D

$$s(t_1, t_2) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} e(u, v) h(t_1 - u, t_2 - v) dt_1 dt_2 \quad (I.19)$$

La convolution discrète d'une image $e(u,v)$ de taille $(N \times M)$ est régie par l'équation suivante :

$$s(n, m) = \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} e(u, v)h(n-u, m-v) \quad (I.20)$$

$$0 \leq n \leq N-1 \text{ et } 0 \leq m \leq M-1$$

I.9.1.3 Filtrage d'images

Un filtre est une transformation mathématique permettant, pour chaque pixel de la zone à laquelle il s'applique, de modifier sa valeur en fonction des valeurs des pixels avoisinants. Les filtres linéaires remplacent chaque pixel par une valeur égale à une somme pondérée de ses voisins. Pratiquement, filtrage d'une image consiste à convoluer sa fonction d'intensité $I(x, y)$ avec une fonction $h(x, y)$ réponse impulsionnelle du filtre.

Il existe plusieurs types de filtre parmi eux on trouve :

- **les filtres passe-bas** : ont un rôle de lissage du signal image: ils adoucissent les variations brutales d'intensité. Comme ils éliminent simplement le bruit, ils modifient assez peu l'image et gardent généralement intacte l'intensité moyenne de l'image.
- **les filtres passe-haut** : ont pour but de rehausser les frontières : ils mettent en valeur les points isolés ou les limites des objets.

I.9.2 Filtres de Gabor

Un filtre de Gabor est un filtre linéaire. Sa réponse impulsionnelle est définie par une fonction gaussienne modulée par une fonction sinusoïdale telle que:

$$h(x, \omega, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(\frac{-x^2}{2\sigma^2} + j\omega x\right)} \quad (\text{I. 21})$$

$\omega = 2\pi f$, f : fréquence de la sinusoïde.

σ : l'écart type.

Cette fonction, étant complexe, peut être décomposée en partie imaginaire et partie réelle.

La partie réelle du filtre (Figure I.16) est donnée par la formule suivante :

$$h(x, \omega, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(\frac{-x^2}{2\sigma^2}\right)} \cos(\omega x) \quad (\text{I. 22})$$

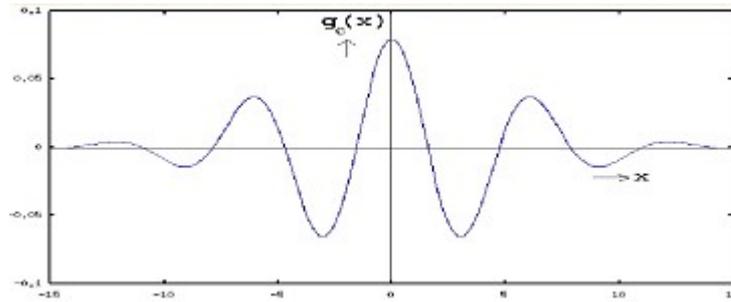


Figure I.16 : partie réelle du filtre de Gabor 1-D.

Tandis que la partie imaginaire, dont l'allure est donnée par la figure (I.17), est :

$$h(x, \omega, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(\frac{-x^2}{2\sigma^2}\right)} \sin(\omega x) \tag{I.23}$$

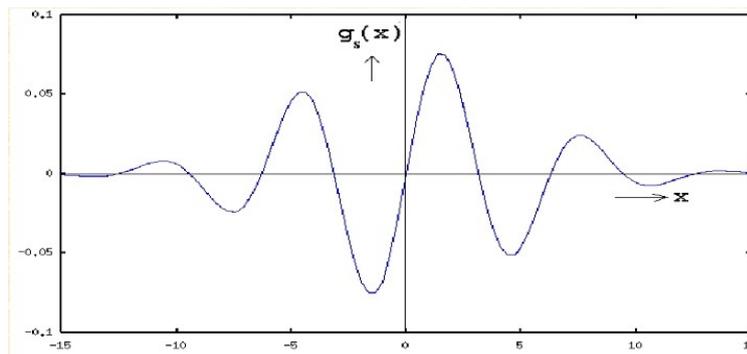


Figure I.17 : Partie imaginaire du filtre de Gabor 1-D.

Un filtre de Gabor bidimensionnelle (2D) est une fonction sinusoïdale complexe orientée d'un angle θ par rapport à l'axe des abscisses x et modulée par une enveloppe gaussienne. Dans le domaine spatial il est écrit ainsi :

$$h(x', y', \omega, \theta, \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left[\left(\frac{x'}{\sigma_x}\right)^2 + \left(\frac{y'}{\sigma_y}\right)^2\right]} + j\omega x' \tag{I.24}$$

Avec :

- $x' = x \cos \theta + y \sin \theta$
- $y' = -x \sin \theta + y \cos \theta$
- θ représente l'orientation du filtre.

- σ_x (respectivement σ_y) est la variance de la gaussienne selon l'axe X (respectivement Y), elles déterminent la largeur de bande du filtre de Gabor
- f la fréquence de la sinusoïde ($\omega = 2\pi f$).

Pour un filtre orienté à 0° ($\theta = 0^\circ$) a symétrie paire et a partie réelle, on aura :

$$h(x, y, \omega, \sigma_x, \sigma_y) = e^{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2\right]} \cdot \cos(\omega x) \tag{I.25}$$

Le filtre de Gabor à symétrie paire et à orientation non nulle ($\theta \neq 0$) est donné par la formule suivante :

$$h(x', y', \omega, \theta) = e^{-\frac{1}{2}\left[\left(\frac{x'}{\sigma_x}\right)^2 + \left(\frac{y'}{\sigma_y}\right)^2\right]} \cdot \cos(\omega x') \tag{I.26}$$

C'est ce filtre que nous avons choisi dans notre travail.

La figure (I.17) montre l'allure des filtres de Gabor bidimensionnels

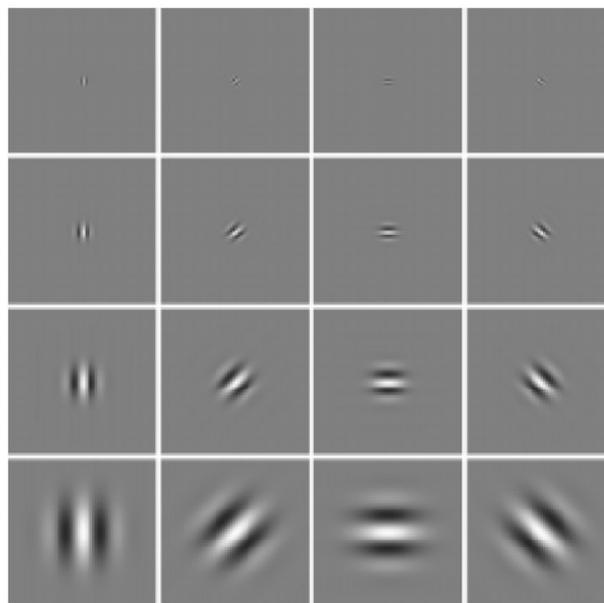


Figure I.18 : Partie réelle de fonctions de Gabor dans le domaine spatial suivant la variation de la fréquence (f) et les écarts types (σ_x, σ_y) ainsi que de l'orientation θ .

La figure (I.19) donne un exemple de filtrage d'images par les filtres de Gabor.

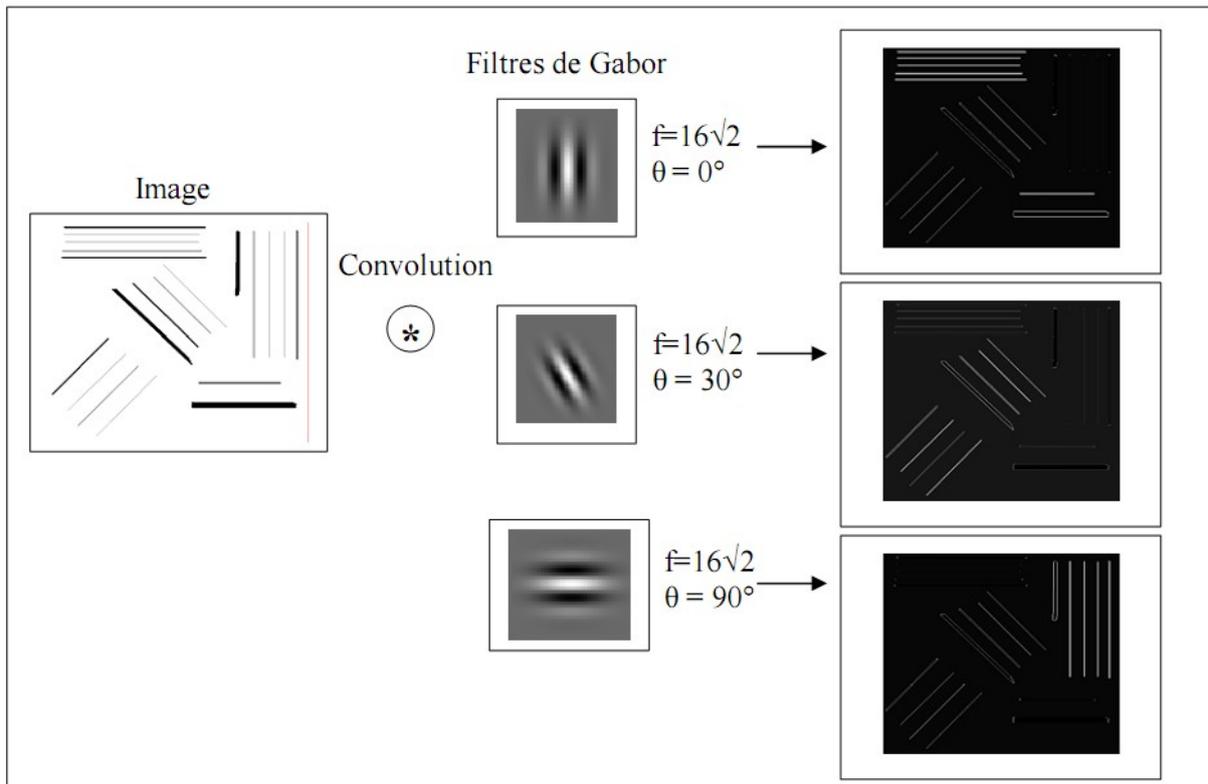


Figure I.19 : exemple d'image filtré par un filtre de Gabor dans le domaine spatial avec différentes fréquences et orientations

La transformée de Fourier du filtre de Gabor est donc donnée par la relation suivante :

$$H(U, V) = A \left(e^{-\frac{1}{2} \left[\left(\frac{U-f}{\sigma_u} \right)^2 + \left(\frac{V}{\sigma_v} \right)^2 \right]} + e^{-\frac{1}{2} \left[\left(\frac{U+f}{\sigma_u} \right)^2 + \left(\frac{V}{\sigma_v} \right)^2 \right]} \right) \quad (I.27)$$

Tels que

$$\sigma_u = \frac{1}{2} \pi \sigma_x, \quad \sigma_v = \frac{1}{2} \pi \sigma_y \quad \text{et} \quad A = 2 \pi \sigma_x \sigma_y$$

De ce fait, dans le domaine fréquentiel, le signal est représenté par deux gaussiennes le long de l'axe U, centrées en $+f$ et $-f$ (figure I.20).

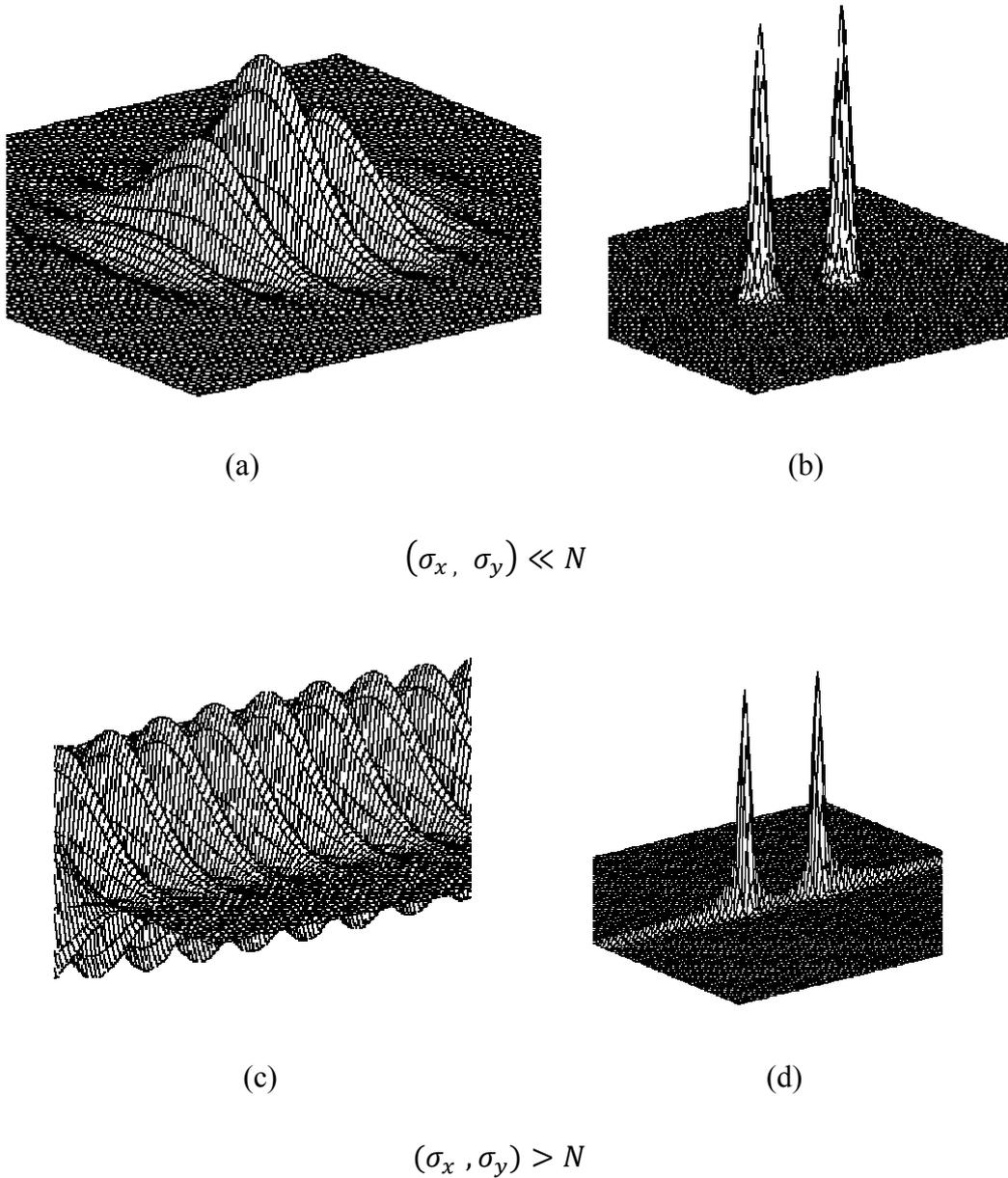


Figure I.20 : (a-c)Filtre de Gabor en espace (b-d) Filtre de Gabor en fréquence

Nous remarquons que dans le cas où l'écart type (σ_x, σ_y) est plus grand que la taille du filtre N , la transformée de Fourier n'est plus une gaussienne mais un sinus cardinal. (Figure I.19.d). Notons que la fréquence f dépend de la taille N du filtre ainsi que de sa période T ($f=N/T$).

La figure (I.21) montre comment appliquer les filtres de Gabor dans le domaine fréquentiel.

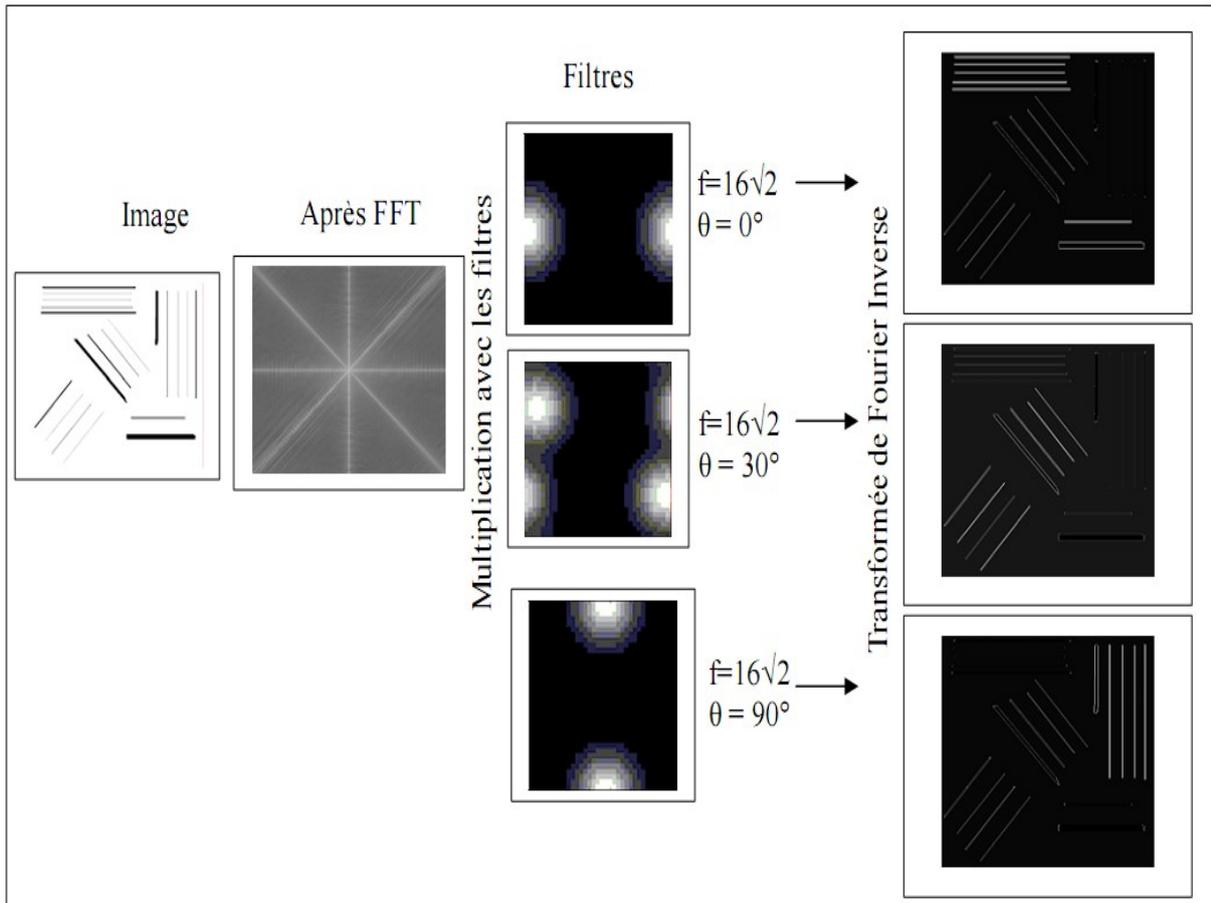


Figure I.21 : Exemple d'image filtrée par un filtre de Gabor dans le domaine fréquentiel avec différentes fréquences et orientations

L'aspect du filtre dépend de la fréquence, de son orientation et de ses écarts types.

- L'orientation du filtre définit la direction du filtre.
- La fréquence définit la sensibilité du filtre.
- Les écarts types définissent la réceptivité (largeur) du filtre.

L'association et l'agencement d'un ensemble de filtres de Gabor permet de couvrir tout l'espace fréquentiel, on parle alors de **filtrage multi-canal** (Figure I.22). Dans ce cas, le filtre finale est défini comme suit :

$$h = \sum_i h(x', y', \omega_i, \theta_i, \sigma_{ix}, \sigma_{iy}) \quad (I.28)$$

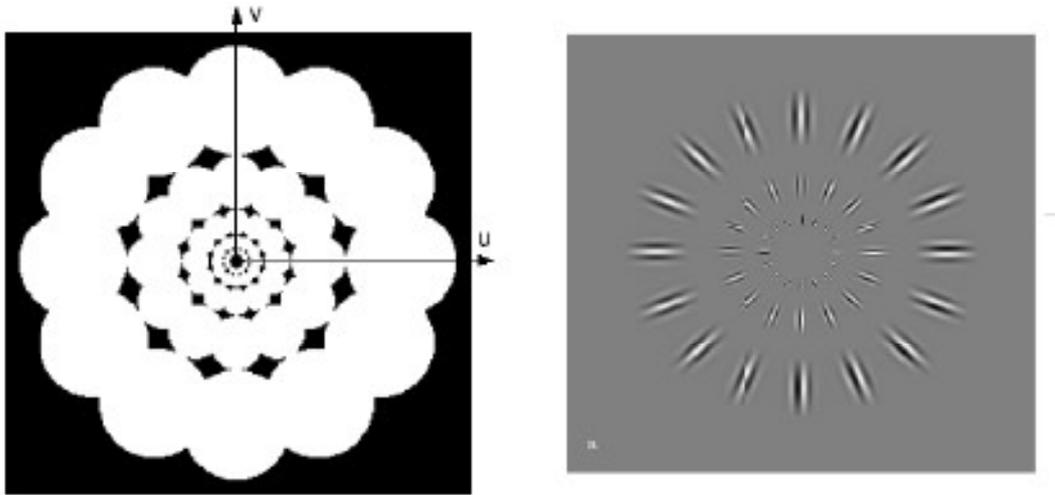


Figure I.22 : Ensemble de filtres de Gabor dans le domaine fréquentiel (le filtre multi-canal).

Le filtrage multi-canal est simplement et efficacement réalisé par le filtre de Gabor qui permet de couvrir tout le domaine fréquentiel et dans de multiples orientations. Chaque canal permet d'extraire les composantes de l'image dont les valeurs correspondent aux paramètres du filtre (fréquence et orientation).

Pour pouvoir couvrir au maximum l'espace fréquentiel, on peut utiliser plusieurs orientations θ tel que :

$$\theta_k = (k - 1)\pi / m, k=1\dots m, \text{ où } m \text{ représente le nombre de d'orientations.}$$

Ainsi pour 4 orientations nous aurons :

$$m=4, \quad k=1, 2, 3, 4, \quad \theta_1=0, \quad \theta_2=\pi/4, \quad \theta_3=\pi/2, \quad \theta_4=3\pi/4$$

L'idée de l'approche gaborienne est alors de concevoir un filtrage particulièrement sélectif en fréquence et orientation dans le but de caractériser au détail près les textures. Chaque filtre est alors appliqué à l'image d'origine.

I.10 Calcul des caractéristiques d'une texture par les filtres de Gabor

Pour extraire les caractéristiques d'une texture en utilisant les filtres de Gabor on procède comme suit :

I.10.1 Calcul d'énergie

L'énergie correspond à la somme des niveaux de gris de l'ensemble des éléments primitifs (pixels), elle mesure l'homogénéité de l'image.

Après application des filtres de Gabor sur l'image avec différentes orientations et échelles, nous obtenons un ensemble de grandeurs (magnitudes) sous la forme suivante:

$$E_i = \sum_x [G_i(x, y)]^2 \quad (\text{I.29})$$

$i = 1 \dots m$ (m étant le nombre de filtres)

où :

$M \times N$: taille du filtre

$(P \times Q)$: taille de l'image

$G_i(x, y)$: image filtrée

Ces grandeurs représentent la teneur en énergie à différentes balances (échelles) et orientations de l'image. Le but principal de la récupération texture est de trouver des images ou régions avec des textures semblables.

I.10.2 Calcul de la moyenne et de la variance

Deux paramètres de la transformée de Gabor sont utilisés pour représenter le dispositif de texture des régions ou images homogènes, la moyenne μ qui caractérise l'intensité lumineuse de l'image et qui correspond au niveau de gris moyen des éléments primitifs (pixels) et la variance σ qui caractérise le contraste de l'image. Ces paramètres sont calculés de la manière suivante :

$$\mu_i = \sum_x \sum_y G_i(x, y) / P \times Q \quad (\text{I.30})$$

$$\sigma_i = \sqrt{\sum_x \sum_y |G_i(x, y)| - \mu_i)^2 / P \times Q} \quad (\text{I.31})$$

Ainsi, on définit un vecteur d'attributs de texture pour une image par :

$$(\mu_1(I), \sigma_1(I), \mu_2(I), \sigma_2(I), \dots \dots \dots, \mu_m(I), \sigma_m(I)) \quad (I.32)$$

Où m représente le nombre de filtre.

I.11 Domaines d'application de l'analyse de la texture

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.

- L'analyse de 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...
- 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...).

I.12 Conclusion

Dans ce chapitre, nous avons présenté quelques notions théoriques sur la texture d'images ainsi que son utilité et ses différentes applications. On a également cité les différentes méthodes utilisées pour son analyse.

Par la suite nous avons présenté en détails les deux méthodes les plus utilisées dans le domaine de l'analyse de la texture : La méthode de Gabor basée sur les filtres de Gabor et la méthode Haralick basée sur les matrices de cooccurrence. Ces méthodes ont pour but d'extraire les caractéristiques les plus significatives d'une image texturée. Elles seront exploitées dans notre travail.

II.1 Introduction

Le cœur d'un système d'analyse automatique d'images est l'étape de segmentation ou d'extraction des caractéristiques. Cette étape est critique dans l'analyse d'une image vu son poids sur le résultat final.

Dans ce présent chapitre, nous définissons la segmentation et ses différentes méthodes.

II.2 La segmentation [4], [5]

La segmentation est une étape primordiale dans l'analyse d'image et la reconnaissance de formes. Elle constitue une opération de traitement d'images qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis (proximité spatiale, similarité). Les pixels sont ainsi regroupés en régions, qui constituent un pavage ou une partition de l'image. Il peut s'agir par exemple de séparer les objets du fond. Si le nombre de classes est égal à deux, elle est appelée aussi binarisation.

Si l'homme sait naturellement séparer, distinguer et reconnaître des objets dans une image c'est grâce à des connaissances de haut niveau (compréhension des objets et de la scène). Mettre au point des algorithmes de segmentation de haut niveau (chaque région est un objet sémantique) est encore un des thèmes de recherche les plus courants en traitement d'images. D'ailleurs, ces dernières années, beaucoup de travaux de recherche lui ont été consacrés mais, elle reste toujours un problème très compliqué et difficile dans son aspect universel. Donc, on admet que la segmentation a pour rôle de partitionner une image en un ensemble de régions (zones homogènes) en respectant une certaine cohérence visuelle et une continuité spatiale.

À ce jour, il existe de nombreuses méthodes de segmentation, que l'on peut regrouper en trois principales classes :

- Segmentation basée sur les régions (en anglais : *region-based segmentation*). On y trouve par exemple : la croissance de région (en anglais : *region-growing*), décomposition/fusion (en anglais : *split and merge*).
- Segmentation basée sur les contours (en anglais : *edge-based segmentation*).

- Segmentation basée sur la classification des pixels. Le seuillage (en anglais : *clustering*) est un cas particulier de cette approche.

II.2.1 Définition de la segmentation

En traitement d'images, on appelle segmentation d'une image, ou d'une séquence, lorsqu'on a partitionné l'espace ou elle est définie en régions au regard d'un critère que l'on s'est donné. Il existe plusieurs méthodes de la segmentation des images.

II.2.2 Méthodes de la segmentation [5]

La segmentation des images consiste à regrouper les pixels de ses images qui partagent une même propriété pour former des régions connexes. Il existe deux familles d'approche que l'on peut coopérer : les approches « contours » et les approches « régions ». Dans le premier cas, les régions (zones) sont délimitées par les contours des objets qu'elles représentent (séparation). Dans le seconde cas, les régions sont déterminées en fonction de leur propriétés intrinsèques (agrégation de pixels fonction d'un critère d'homogénéités). Les deux approches sont duales. Une image est donc une mosaïque de régions homogènes séparées par des régions de transition.

II.2.2.1 Segmentation en contours [5]

On l'appelle aussi segmentation des images par approche « frontière ». Cette approche cherche à exploiter le fait qu'il existe une transition détectable entre deux régions connexes.

Les méthodes les plus anciennes utilisent des opérateurs de traitement d'images, tels que le filtre de Canny, pour mettre en évidence les pixels qui semblent appartenir à un contour. La construction d'une partition est alors souvent difficile.

On peut aussi faire intervenir des modèles déformables à l'aide de courbes paramétriques (courbe de Bézier, spline...) ou de polygones (par exemple algorithme à bulle).

Pour initier le processus, on recherche des points remarquables de l'image, tels que des points à l'intersection de trois segments au moins. De tels points sont appelés des *graines* (*seeds* en anglais).

Le principe d'une chaîne d'extraction automatique des contours est le suivant :

1. L'extraction des points de contour
2. La binarisation des contours
3. La réduction de l'épaisseur des contours à un pixel
4. La fermeture et suivi de contours
5. Le codage et description des contours.

La dernière étape a pour objectif d'organiser les contours en structures simples telles que les suites numériques ; segments de droite ; arcs de cercle etc. Pour extraire les points, on cherche les variations des niveaux de gris ou de couleur (figure II.1). Une variation existe si la dérivée première est localement maximale et si la dérivée seconde présente un passage par zéro (figure II.2). Les deux familles d'algorithmes les plus connus pour la détection de contour exploitent soit une approximation de la dérivée première; ou gradient ; soit une approximation de la dérivée seconde ou Laplacien.

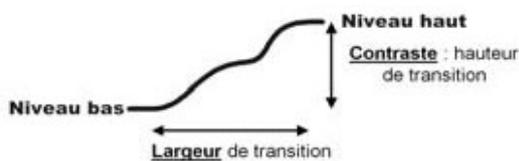


Figure II.1 : Variation de niveaux de gris ou de couleur provoque l'apparition d'un contour.

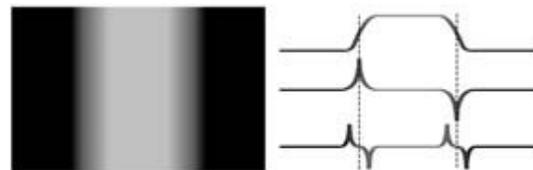


Figure II.2 : Une image avec profit associé (en haut) et ses dérivées premières (au milieu) et secondes (en bas)

II.2.2.1.1 L'approche gradient

L'estimation de la dérivée en un point de coordonnée (p ; q) selon les lignes x et les colonnes y peut être obtenue grâce aux formules suivantes :

$$\frac{\partial f}{\partial x} \Big|_{p,q} \approx \frac{1}{2} [f(p+1, q) - f(p-1, q)] \quad (\text{II.1})$$

$$\frac{\partial f}{\partial y} \Big|_{p,q} \approx \frac{1}{2} [f(p, q+1) - f(p, q-1)] \quad (\text{II.2})$$

Transformées en opérations matricielles, les formules (II.1) et (II.2) permettent d'aboutir au masque (taille 3×3) décrits dans les tableaux 1. Selon la décomposition des équations (II.1) et (II.2), il est possible d'obtenir des tailles des masques très élevées.

L'accroissement de taille permet d'améliorer la qualité d'estimation de la dérivée (débruitage), mais a pour conséquence d'augmenter le temps de calcul et de réduire la localisation du contour. Les deux masques sont convolués indépendamment avec l'image. Ensuite, pour chaque pixel un module et une direction du gradient peuvent être établis selon les formules (II.3), (II.4), (II.5).

$$\nabla f = \begin{bmatrix} Gx \\ Gy \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{II.3}$$

$$Module = \nabla f = \sqrt{Gx^2 + Gy^2} \tag{II.4}$$

$$Direction = arg(\nabla f) = \tan^{-1} \left[\frac{Gx}{Gy} \right] \tag{II.5}$$

Pratiquement, les gradient Gx et Gy sont obtenus en convoluant l'image avec les masques Hx et Hy comme ceux du tableau.

-1	-1	-1
0	0	0
1	1	1

Hx

-1	0	1
-1	0	1
-1	0	1

Hy

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

Hx

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

Hy

Tableau II.1 : Dérivés selon y (colonnes de gauche) et selon x (colonnes de droite), filtre Prewit (en haut) et Sobel (en bas).

Notons que la direction du gradient en un point est perpendiculaire à la direction du contour en ce point de contour, (équation II.5). A partir des valeurs du module, une extraction des valeurs importantes permet de ne retenir que les variations de niveaux de gris ou de couleur significative (supérieur à un seuil), donc à un contour. Une variante pour ce calcul de gradient en x et en y est le gradient directionnel, appelé aussi gradient boussole qui consiste à convoluer l'image avec huit masques obtenus par rotation successive de $+\pi/4$ du masque de base, indiquant ainsi les huit directions possibles pour un contour (tableau II.2).

5	5	5
-3	0	-3
-3	-3	-3

5	5	-3
5	0	-3
-3	-3	-3

5	-3	-3
5	0	-3
5	-3	-3

Tableau II.2 : les trois premiers masques de Kirsh.

II.2.2.1.2 L'approche Laplacien

L'approximation des dérivées secondes peut être effectuée de la même manière par un développement limité (équation (II.6), (II.7)).

$$\frac{\partial f}{\partial x^2} \Big|_{p,q} \approx \frac{1}{4} [f(p+1, q) - 2f(p, q) + f(p-1, q)] \tag{II.6}$$

$$\frac{\partial f}{\partial y^2} \Big|_{p,q} \approx \frac{1}{4} [f(p, q+1) - 2f(p, q) + f(p, q-1)] \tag{II.7}$$

Pratiquement, le Laplacien est obtenu par la convolution de l'image f avec l'un des masques du tableau II.3

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

Tableau II.3: Filtres Laplaciens.

Conformément aux propos précédents (figure II.2), l'extraction des contours grâce aux dérivées secondes se fait à partir d'une seule convolution et n'exploite pas les maxima mais les passages par zéro de la fonction bidimensionnelle. A la différence du gradient, la détection du contour produit une ligne double (Figure II.3). Néanmoins ces lignes de contours sont fermées et d'épaisseur 1 pixel, alors qu'avec un gradient le seuillage introduit des coupures dues aux effets du seuillage. Cependant l'opérateur Laplacien est sensible au bruit et ne donne aucune indication quant à la direction des contours.



Figure II.3 : Extraction de contours par le laplacien (zero-crossing).

Le passage à la couleur est plus délicat à cause de l'expression vectorielle de l'information. Une première solution consiste à travailler en marginal sur chaque composante puis à fusionner les cartes des contours. Les alternatives exploitent des méthodes vectorielles comme le gradient couleur de Di Zenzo. L'ensemble des méthodes est dans [1]. Il peut être nécessaire d'opérer un suivi de contours afin d'obtenir des régions fermées interprétables comme projection des objets de la scène. La description du contour passe par une modélisation de celui-ci ou un codage tel que celui de Freeman (Figure II.5) qui caractérise le passage d'un pixel à son voisin. La suite des codes locaux donne le codage du contour (exemple : 00564412).



Figure II.4 : image originale, filtré par un Sobel.

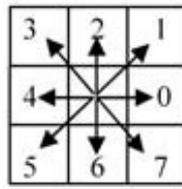


Figure II.5 : codage de l'orientation du contour (Codage de Freeman).

L'intérêt principal des méthodes de segmentation selon l'approche frontière est de minimiser le nombre d'opérations nécessaires en cas d'itération du processus sur des séries d'images peu différentes les unes des autres (cas des images vidéo notamment). En effet, une fois que les contours des régions ont été trouvés dans la première image, l'application du modèle déformable à l'image suivante est plus efficace que de tout recalculer, si la différence entre les images est peu importante.

III.2.2.2 Segmentation en région [5]

La segmentation en régions homogènes vise à partitionner l'image en se basant sur des propriétés intrinsèques des régions. Un critère simple est celui de niveau de gris ou de la couleur identique à ϵ près pour tous les pixels d'une même région. Mais les critères peuvent s'appuyer sur des statistiques beaucoup plus complexes de l'image, comme ceux décrivant la texture. Il est courant de visualiser le partitionnement d'une image en régions en leur couleur différente.

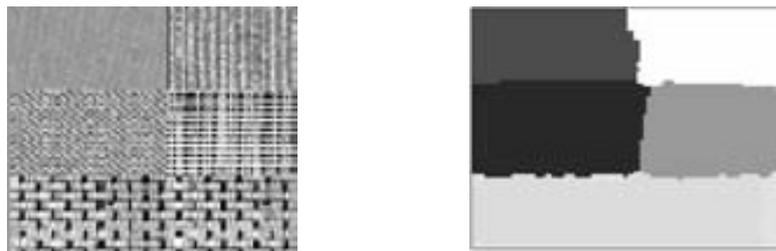


Figure II.6 : Segmentation par critère de texture.

Soit une image I partitionnée en un ensemble de régions R et soit P prédicat, ou critère, d'homogénéité. Une région, qui doit respecter des critères d'homogénéité et de connexité, signifie une zone homogène (région) dans l'image c'est-à-dire : tous les pixels qu'elle contient ont les mêmes caractéristiques (niveau de gris, couleur, texture, gradient, etc.).

Il existe un prédicat d'uniformité qu'on note $P(Rk)$ qui définit l'homogénéité d'une région. Il est vrai si Rk est homogène, faux si Rk ne l'est pas.

Les régions Rk doivent satisfaire certains critères ou conditions suivantes :

- Tout pixel appartient à une région : $I = \bigcup_{k=1}^K R_k$
- Aucun pixel n'appartient à plus d'une région : $R_i \cap R_j = \emptyset \quad (i \neq j)$
- Proximité (cohérence) spatiale : P connecté à P' quelque soit P, P' appartient à Rk
- Cohérence des attributs de région :
 - $P(R)$ est vrai pour $k : 1, 2, 3, \dots, K$
 - $P(R_i \cup R_j)$ est faux pour R_i, R_j adjacentes et $i \neq j$

II.2.2.2.1 Segmentation par croissance de régions [6],[7]

Cette méthode de segmentation consiste premièrement à choisir initialement des germes de régions, puis à faire croître chaque germe en intégrant progressivement des pixels voisins à ces germes. On trouve cette technique dans le système visuel humain : quand on regarde une image, on commence toujours par un point particulier puis on cherche ensuite à reconnaître l'objet autour de ce point.

Les algorithmes par croissance de régions partent d'un premier ensemble de régions, qui peuvent être calculées automatiquement (par exemple, les minima de l'image), ou fournies par un utilisateur de manière interactive. Les régions grandissent ensuite par incorporation des pixels les plus similaires suivant un critère donné, tel que la différence entre le niveau de gris du pixel considéré et le niveau de gris moyen de la région.

Dans cette méthode, deux facteurs agissent et influencent sur le résultat de la segmentation :

- Le premier est le choix des germes initiaux ;
- Le deuxième est la condition d'arrêt de la croissance de régions.

On peut noter que la croissance de régions exploite correctement les informations contextuelles et les informations topologiques de l'image.

L'inconvénient majeur de cette méthode est que le temps de calcul est très important et qu'elle produit toujours des régions trop nombreuses par rapport au nombre d'objets dans l'image.

III.2.2.2 Segmentation par division [5]

Le principe consiste à tester d'abord le critère d'homogénéité retenu sur l'image entière. Si le critère est valide, l'image est considérée comme segmentée ; sinon, l'image est découpée en zones plus petites et la méthode est réappliquée sur chacune des zones. Pour une image texturée, l'homogénéité est déterminée en fonction des attributs de texture. La division peut se faire en quatre parties, en six parties, en polygones, etc. Le plus souvent une méthode dite de *quadtree* est appliquée, à savoir que tant que tous les pixels ne sont pas identiques dans la région au sens du critère choisi, celle-ci est divisée par 4. L'inconvénient de ces méthodes est que deux parties adjacentes peuvent vérifier le même critère sans avoir été regroupé dans la même région. Ces méthodes font partie des approches descendantes.

II.2.2.2.3 Segmentation par fusion [5]

A l'inverse de la démarche précédente, l'image est explorée en partant de petites régions. Ces germes vont croître en taille en agglomérant des pixels ou des régions connexes satisfaisant un critère d'homogénéité.

Le balayage de l'image se fait à partir d'un germe initial et suivant un ordre déterminé. Un exemple d'une technique dédié aux images couleur est donné dans. Un pixel fusionne avec ses voisins si trois prédicats sont vérifiés. Soit R l'ensemble des pixels déjà agrégés dans la région considérée, soit V l'ensemble des pixels voisins du pixel courant et déjà à traiter. On note le vecteur \vec{Cp} sa couleur.

P est agrégé au pixel p' si :

$$d^2(\vec{Cp}, \vec{Cp}') < S1$$

$$d^2(\vec{Cp}, \vec{\mu v}) < S2$$

$$d^2(\vec{Cp}, \vec{\mu r}) < S3$$

Avec $\vec{\mu v}$ la couleur moyenne du voisinage V et $\vec{\mu r}$ la couleur moyenne de la région.

Si le pixel courant peut fusionner avec plusieurs voisins, il est affecté à la région qui minimise les trois distances. Si le pixel courant ne peut fusionner, il est le germe d'une nouvelle région. Un critère semblable peut être déterminé pour une image texturée.

Les inconvénients majeurs de cette méthode reposent sur l'influence de la position initiale du germe et du type de balayage en considérant le vecteur de paramètres de texture comme une valeur multispectrale.

II.2.2.2.4 Segmentation par division / fusion [5]

La segmentation par division induit une structure hiérarchisée qui permet d'établir de proximité entre les régions, mais qui peut fractionner une même région en plusieurs ensembles distincts. La segmentation par fusion produit un nombre minimal de régions connexes, mais décrit celles-ci dans une structure horizontale qui n'exprime pas de relation de proximité. La combinaison des deux approches (appelée *Split and Merge*) est d'exploiter un la fusion de régions à partir d'une division grossière obtenue par division de l'image. L'étape de division (*Split*) divise l'image en régions jusqu'à ce que la propriété d'homogénéité soit vraie dans la sous-image. Dans l'étape de fusion (*merge*), les régions adjacentes dont l'union vérifié un prédicat (pas forcément identique à celui de la première phase) sont regroupées.

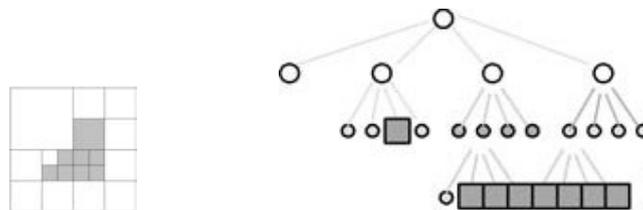


Figure II.7 : Principe de *Split and Merge*

La figure II.8 nous montre un exemple de segmentation par la méthode *Split and Merge*.

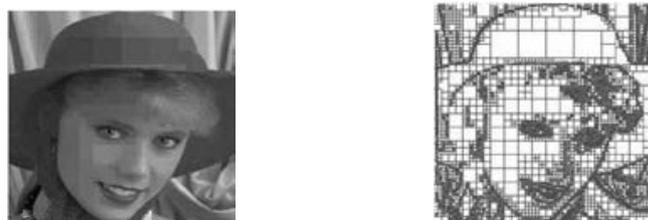


Figure II.8 : Split and Merge (4431 divisions pour obtenir 13294 régions homogènes puis fusion pour segmenter en 7995 en régions homogènes.

Les algorithmes de type division/fusion exploitent les caractéristiques propres de chaque région (surface, intensité lumineuse, colorimétrie, texture, etc.). On cherche des couples de régions candidates à une fusion et on les note en fonction de l'impact que cette fusion aurait sur l'apparence générale de l'image. On fusionne alors les couples de régions les mieux notés, et on réitère jusqu'à ce que les caractéristiques de l'image remplissent une condition prédéfinie : nombre de régions, luminosité, contraste ou texture générale donnée, ou alors jusqu'à ce que les meilleures notes attribuées aux couples de régions n'atteignent plus un certain seuil (dans ce dernier cas , on parle d'un algorithme avec *minimisation de fonctionnelle*).

II.2.2.2.5 Segmentation basée sur la classification des pixels

Dans cette approche, les pixels sont caractérisés par un ensemble d'attributs. Ces attributs correspondant au niveau de gris, ou aux attributs de texture calculés à partir des pixels voisins situé à l'intérieure d'une fenêtre de voisinage centrée sur chaque pixels ou encore, aux composantes couleurs s'il s'agit d'une image couleur. Les pixels sont projetés dans l'espace des attributs et forment de nuages de points.

La classification des pixels consiste alors à extraire des nuages de points compacts qui correspondent aux classes de pixels dans l'image. Les méthodes de classification construisent les classes de pixels : en présence dans l'image et affectent une étiquette à chaque pixels. La formation des régions homogènes dans l'image n'est obtenue qu'après l'analyse de connexité des pixels dans l'image étiquetée.

Les méthodes exploitant plusieurs attributs sont qualifiées de multidimensionnelles. Les méthodes ne prenant en compte qu'un seul attribut (niveau de gris) sont qualifiées de monodimensionnelles. Elles consistent à extraire automatiquement des seuils puis affectent les pixels à une classe par comparaison de leur niveau de gris à ces seuils. On les appelle communément méthodes de seuillage.

III.2.2.2.5.1 Le seuillage

Il a pour objectif de segmenter une image en plusieurs classes en n'utilisant que l'histogramme, qui est supposé caractéristique du contenu de l'image. Les creux ou vallées d'un histogramme correspondent à des niveaux de gris peu représentés dans l'image. A chaque pic de l'histogramme est associée une classe. On dit qu'un histogramme est

bimodal lorsque deux pics (deux modes) sont clairement identifiables. Comme l'histogramme ne contient pas d'information sur la répartition spatiale des couleurs, des pixels non connexes peuvent être associés à la même classe.

Ce type de méthode ne peut s'appliquer à segmenter une image texturée car pour cette dernière, chaque pixel est caractérisé pour un ensemble d'attributs de texture.

II.2.2.2.5.2 Définition 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 (Figure II.9)

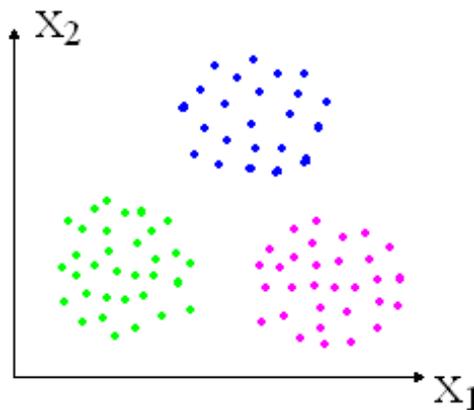


Figure (II.9) : Répartition des données dans un espace bidimensionnel.

Les observations ayant des attributs semblables appartiennent à la même classe et forment un nuage de point très dense dans l'espace P dimensions. Ainsi, la classification en générale est définie comme étant l'action de construire une collection d'objets similaires au sein d'un même groupe, dissimilaire 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.

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

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. $C_i \neq \phi$ Pour $i=1,2,\dots,K$
2. $C_i \cap C_j = \phi \quad \forall i \neq j$
3. $\bigcup_{i=1}^K C_i = \mathcal{Q}$

La première condition indique qu'une classe ne doit pas être vide, tandis que la deuxième condition stipule qu'aucun recouvrement entre les classes n'est toléré, alors que la troisième vérifie que l'union de toutes les classes doit aboutir au nombre d'observations disponibles.

Généralement, on exige de la classification de vérifier deux autres propriétés :

- Compacité : les points représentant une classe donnée sont plus proches entre eux que des points de toutes les autres classes.
- Séparabilité : les classes doivent être séparées le plus possible.

La figure (II.10) illustre un exemple graphique simple de classification :

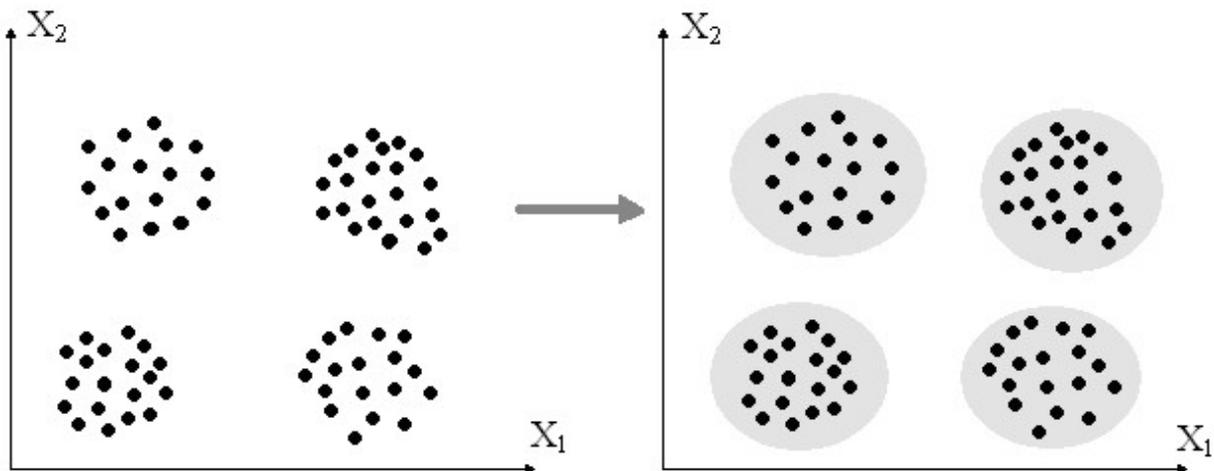


Figure (II.10) : Principe de la classification

Dans cet exemple, nous pouvons identifier facilement 4 classes où les données peuvent être regroupées selon un critère de similitude basé sur le calcul de *distance*. Deux objets peuvent appartenir à une même classe s'ils sont "proches" selon une distance donnée. Cette définition n'est cependant valable que dans le cas d'une *classification exclusive ou dure* "hard en anglais" où un objet doit appartenir à une seule classe contrairement à la *classification non exclusive*, également appelée *douce ou de recouvrement* "soft en anglais",

où un objet peut être assigné à plusieurs classes. La classification floue est un exemple de classification non exclusive où un objet peut appartenir à plusieurs classes avec des degrés d'appartenances différents. Par exemple, le regroupement des personnes en classe en fonction de leur âge ou sexe est exclusif, alors que leur regroupement par catégorie de maladies ne l'est pas puisqu'une personne peut avoir plusieurs maladies simultanément. Une autre catégorie de méthodes de classification est la *classification hiérarchique* qui tolère l'imbrication entre les classes.

Nous proposerons de décrire deux méthodes de classification non supervisée basées sur l'approche métrique et qui sont dénommées C- Means et Fuzzy C-Means. Ces deux algorithmes sont très populaires et sont les plus utilisés en pratique. Ils présentent l'avantage d'être à la fois simples et efficaces.

II.2.2.2.5.3 Algorithme de C- Means

Cet algorithme a été proposé par Macqueen en 1967. Son principe est de partitionner l'ensemble des points en un ensemble de classes prédéterminé. L'algorithme des C-means vise à minimiser la variance intra classes, qui se traduit par la minimisation de l'énergie suivante :

$$J(\pi) = \sum_{k=1}^K \sum_{i=1}^{N_k} d^2(x_i, v_k) \text{ avec } x_i \in C_k \quad (\text{II.8})$$

Où π est une partition composée de k classes.

$d(x_i, v_k)$ Représente la distance entre un individu X_i et une classe représentée par son centre v_k . Les coordonnées du centre de classe C_k sont données par l'expression suivante :

$$v_k = \frac{1}{N_k} \sum_{i=1}^{N_k} x_i \quad x_i \in C_k \quad \text{Où } N_k \text{ est le nombre d'observations de la classe } C_k .$$

La partition π^* qui optimise le critère est définie par : $J(\pi^*) = \min J(\pi)$.

Plusieurs types de distances peuvent être utilisés dans R^N (euclidienne, Mahalanobis, Minkowsky, etc...). Cependant en pratique, on utilise assez souvent la distance euclidienne définie comme suit :

$$d(x_i, v_k) = \|x_i - v_k\| = \sqrt{\sum_{j=1}^D (X_{ij} - v_{kj})^2} \quad (\text{II.9})$$

Pratiquement, l'algorithme C-means se déroule de la manière suivante :

Des points initiaux sont choisis aléatoirement pour constituer les centres de classes. Les autres points sont alors assignés à la classe dont le centre est le plus proche. Ensuite, on calcule le centre des classes et on répète le même processus jusqu'à avoir une partition optimale.

Géométriquement cela revient à partager l'espace des points en K zones définie par les plans médiateurs des segments de droite reliant deux centres de deux classes différentes (figure II.11)

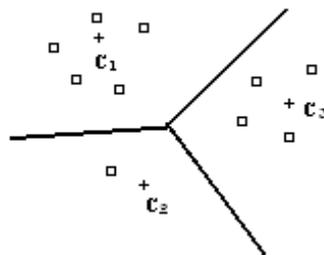


Figure (II.11) : Exemple de partition.

D'une manière plus générale, cet algorithme se déroule selon les étapes suivantes :

Soit un ensemble X de N individus caractérisés par P variables, à partitionner en K classes $\{C_1, \dots, C_k, \dots, C_K\}$.

Etape 0 : $t=0$: Initialisation

On choisit aléatoirement K centres de classes $\{v_1^t, \dots, v_k^t, \dots, v_K^t\}$. Ces K centres induisent une première partition π^t de l'ensemble des N individus en K classes $\{C_1^t, \dots, C_k^t, \dots, C_K^t\}$.

Etape 1 : Chaque individu est affecté à la classe la plus proche. Par exemple un individu x_i est affecté à la classe C_k^t si il est plus proche de v_k^t que de tous les autres centres.

$$x_i \in C_k^t \text{ si } d(x_i, v_k^t) < d(x_i, v_{k'}^t)$$

$$\text{Avec } k' = 1, \dots, k \text{ et } k' \neq k$$

Etape 2 : On détermine les K nouveaux centres de classes $\{v_1^{t+1}, \dots, v_k^{t+1}, \dots, v_K^{t+1}\}$.

Ces nouveaux centres induisent une nouvelles partition π^{t+1} de K classes $\{C_1^{t+1}, \dots, C_k^{t+1}, \dots, C_K^{t+1}\}$.

Etape 3 : $t=t+1$ et aller vers l'étape 1 si le critère d'arrêt n'est pas respecté.

t : indique le numéro d'itération. Plusieurs critères peuvent être utilisés :

- Deux itérations successives conduisent à la même partition.

- Un nombre maximal d'itérations, fixé a priori, est atteint.
- Les centres de classe ne change pas entre deux itérations successives.
- Minimisation d'une fonction coût (Equation II.8).

La figure (II.12) illustre le processus de partitionnement d'un ensemble d'individus en $K=3$ classes. Les centres de classes sont indiquées par le symbole (+).

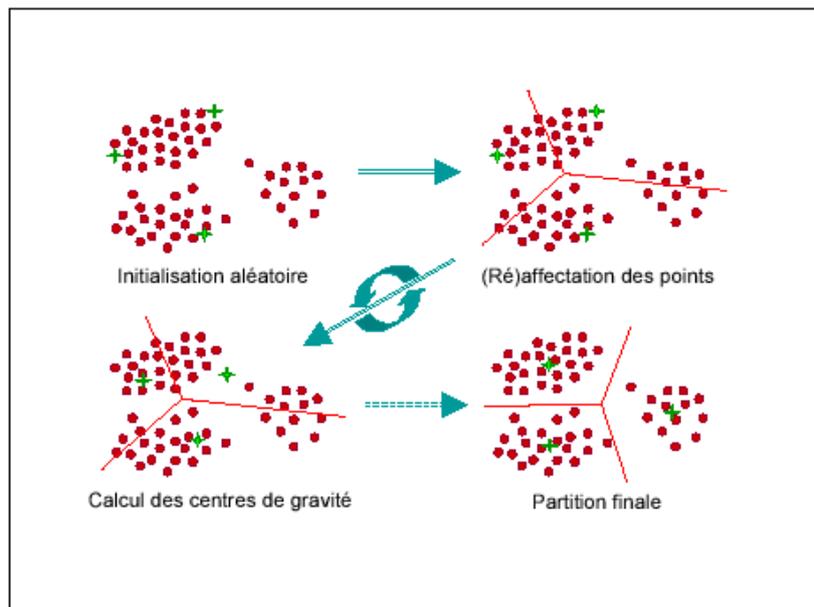


Figure (II.12) : Principe de l'algorithme de C-means.

II.2.2.2.5.4 Classification floue

II.2.2.2.5.4.1 Introduction à la notion floue

Un ensemble flou est un ensemble dont les bords sont mal définis. Cela se traduit par une fonction d'appartenance à valeur dans l'intervalle $[0, 1]$ tout entier, par opposition au cas classique où la fonction d'appartenance prend deux valeurs seulement : 0 ou 1. Cela permet de définir des ensembles d'une manière plus souple, tolérant à des informations imprécises, incomplètes et/ou incertaines. La figure II.13 montre un exemple d'ensemble flou.

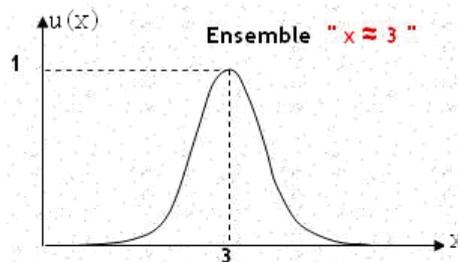


Figure II.13 : Exemple de l'ensemble flou.

Cet ensemble contient les nombres « proches » de 3. Ainsi la valeur 3 appartient complètement à cet ensemble et il a par conséquent, un degré égal à 1. Plus on s'éloigne de la valeur 3, plus le degré d'appartenance diminue.

Cette propriété est exploitée en traitement d'image, et en classification des données où les classes, appelée aussi régions, sont représentées par des ensembles flous. Cela est fort utile lorsque les régions ne peuvent pas être définies de manière nette et précise. La classification floue autorise le chevauchement des régions. La figure (II.14) montre un exemple de classification floue.

Exemple de classes floues :

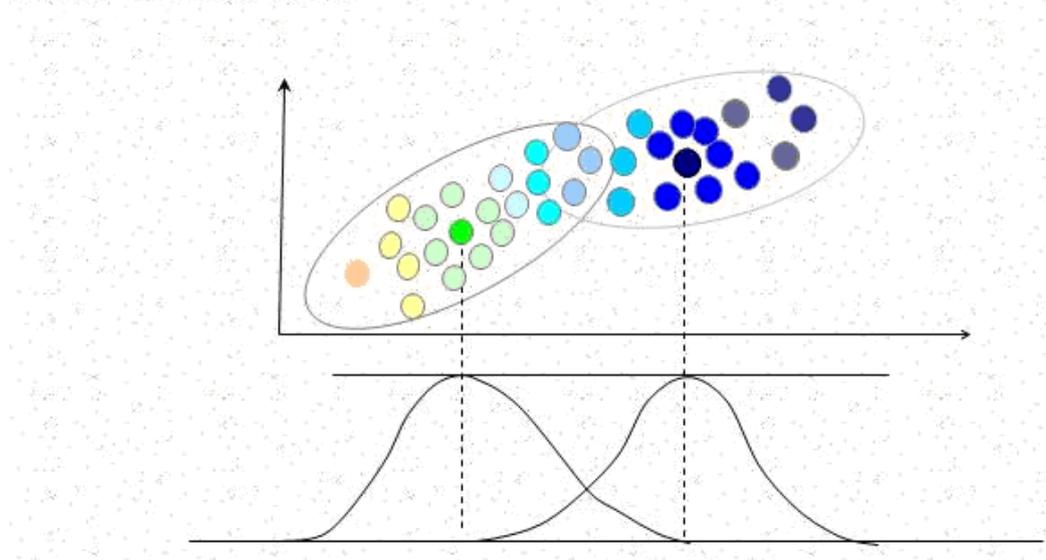


Figure II.14: Exemple de classification floue.

On présentera dans ce qui suit un algorithme de classification très populaire, basé sur la logique floue, connu pour son efficacité et sa robustesse.

II.2.2.2.5.5 L'algorithme de C-moyens flou (FCM)

L'algorithme de Fuzzy C-Means (FCM) est une extension directe de l'algorithme classique C-means, où la notion d'ensemble flou est introduite dans la définition des classes. Cet algorithme a été développé essentiellement par Bezdek, à partir des idées de Ruspini (clustering flou) et de Dunn (ISODATA flou). Contrairement à la classification exclusive où un individu peut appartenir à une seule classe, en classification floue un individu peut appartenir à plusieurs classes avec des degrés différents.

Le principe de base est de former à partir des individus non étiquetés K groupes qui soient les plus homogènes et naturels possible. « Homogène » et « naturel » signifient que les groupes obtenus doivent contenir des individus les plus semblables possible, tandis que des

individus des groupes différents doivent être le plus dissemblables possible. Le comportement flou en classification est alors présenté par une matrice qui quantifie le degré d'appartenance de chaque observation à chaque classe. Cette matrice notée u est de dimension $(N \times K)$ où N est le nombre d'individus et K est le nombre de classes. Les éléments u_{ij} ($i = 1, \dots, N$ et $j = 1, \dots, K$) de cette matrice sont des nombres réels dont les valeurs sont comprises entre 0 et 1. La matrice u obéit aux conditions suivantes :

$$u_{ij} \in [0,1] \quad \forall i, j.$$

$$\sum_{i=1}^K u_{ij} = 1 \quad \forall j.$$

$$0 < \sum_{j=1}^N u_{ij} < N.$$

Par exemple, considérons un ensemble X de N individus à partir de deux classes.

La matrice d'appartenance suivante montre une répartition floue des N individus en 2 classes :

$$u_{K \times N} = \begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ \dots & \dots \\ 0.9 & 0.1 \end{bmatrix}$$

Le premier élément x_1 possède un degré d'appartenance à la première classe égale à $u_{11} = 0.8$ et un degré d'appartenance à la deuxième classe égale à $u_{12} = 0.2$. Cette matrice vérifie toutes les conditions requises.

Il est facile de passer d'une partition floue à une partition exclusive en prenant la valeur maximale des éléments de la matrice u sur chaque ligne. La matrice d'appartenance devient alors :

$$u_{K \times N} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ \dots & \dots \\ 0 & 1 \end{bmatrix}$$

Dans ce cas, chaque observation appartient à seulement à une seule classe indiqué par la valeur 1.

L'algorithme de C-moyens flou (FCM) consiste à minimiser la fonction objective suivante :

$$J_m = \sum_{i=1}^N \sum_{k=1}^K u_{ik}^m d_{ik}^2 \quad 1 \leq m < \infty \quad (\text{II.10})$$

Où m est un nombre réel supérieur à 1, appelé facteur flou. u_{ik} est le degré d'appartenance de l'élément x_i à la classe C_k et d_{ik} est la distance séparant l'objet x_i et le centre v_k de la classe C_k . Cette distance exprime la similarité entre l'observation et le centre d'une classe.

$$d_{ik} = \|x_i - v_k\| > 0 \quad \forall i \text{ et } k .$$

La classification floue s'effectue par le biais d'une optimisation itérative de la fonction objective citée ci-dessus. Celle-ci s'effectue par une mise à jour permanente du degré d'appartenance u_{ik} et du centre de classe v_k .

Les centres de classes sont mis à jour après chaque nouvelle itération des observations. Cette mise à jour est obtenue en fonction des degrés d'appartenances selon la formule suivante :

$$v_k = \frac{\sum_{i=1}^N u_{ik}^m x_i}{\sum_{i=1}^N u_{ik}^m} \quad \forall i \quad (\text{II.11})$$

La mise à jour de ces centres de classes modifie la répartition des observations, par conséquent une mise à jour de la matrice u est nécessaire. Elle est effectuée en utilisant la formule suivante :

$$u_{ik} = \frac{1}{\sum_{j=1}^K \left(\frac{d_{ik}}{d_{ij}} \right)^{\frac{2}{m-1}}} \quad (\text{II.12})$$

L'algorithme de Fuzzy C-means est résumé par les étapes suivantes :

- 1- Fixer le nombre k de classes et le facteur flou m .
- 2- Initialisation des centres de classes, $t=0$.
- 3- Déterminer la matrice d'appartenance $u^{(t)}$.
- 4- $t=t+1$, détermination des nouveaux centres des classes.
- 5- Déterminer la matrice d'appartenance $u^{(k)}$.
- 6- Si $\|u^{(t+1)} - u^{(t)}\| \leq \varepsilon$, s'arrêter ; sinon aller à l'étape 3.

Le critère d'arrêt est indiqué par la dernière étape de l'algorithme, il consiste à vérifier si la répartition entre deux itérations est la même ou non. ε est un nombre réel dont la valeur est comprise entre 0 et 1, alors que la variable t indique le numéro de l'itération.

L'algorithme de FCM souffre des mêmes inconvénients que l'algorithme de C-means à savoir le choix de la répartition initial et le choix de nombre de classes.

II.6 Conclusion

Dans ce chapitre, nous avons exposé différentes approches de segmentation de façon générale.

Notre intérêt dans ce chapitre concerne l'approche de segmentation basée sur la classification des pixels car celles-ci sont bien adaptées pour la segmentation d'une image texturée. A cet effet, deux algorithmes de classification (C-means et Fuzzy C-means), qui sont de nos jours très utilisées en pratique, ont été décrites en détails.

III.1 Introduction

Souvent le résultat final d'un traitement d'image est d'aboutir à une classification ou bien une segmentation optimale. Dans ce même but, nous avons adopté une approche d'analyse des motifs d'une image texturée avant de procéder aux différentes étapes d'un traitement classique. Dans ce qui suit, nous allons essayer de détailler et d'expliquer l'approche adoptée.

III.2 Parcours locaux [9]

En analyse de la texture, on cherche souvent à quantifier les dépendances en niveaux de gris entre des pixels voisins. La matrice de cooccurrence, par exemple, permet de quantifier la dépendance entre niveaux de gris de pixels séparé par une distance σ selon une direction θ comme nous montre la figure III.1 suivante

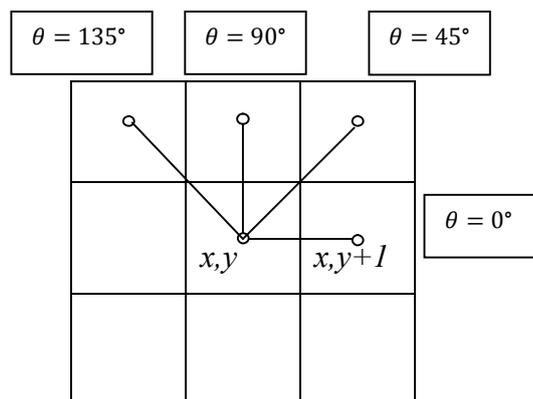


Figure III.1

Dans [9], les auteurs proposent une nouvelle manière de quantifier la dépendance entre niveaux de gris de plusieurs pixels proches l'un de l'autre. Elle est basée sur l'ordonnement des pixels appartenant à un voisinage donné. Cet ordonnancement engendre différents parcours locaux. Ces parcours sont alors décrits par des parcours ou motifs.

Pour illustrer cette notion de parcours locaux, on considère un bloc de quatre pixels voisins. (Figure III.2).

A partir de ces quatre pixels, on peut définir au maximum 24 parcours locaux ou motifs. D'où la définition suivante :

Définition n°1

Etant donné un ensemble de 4 points voisins $g = \{a,b,c,d\}$, un motif M est défini comme le chemin parcouru par une courbe φ passant par chaque point une seule fois.

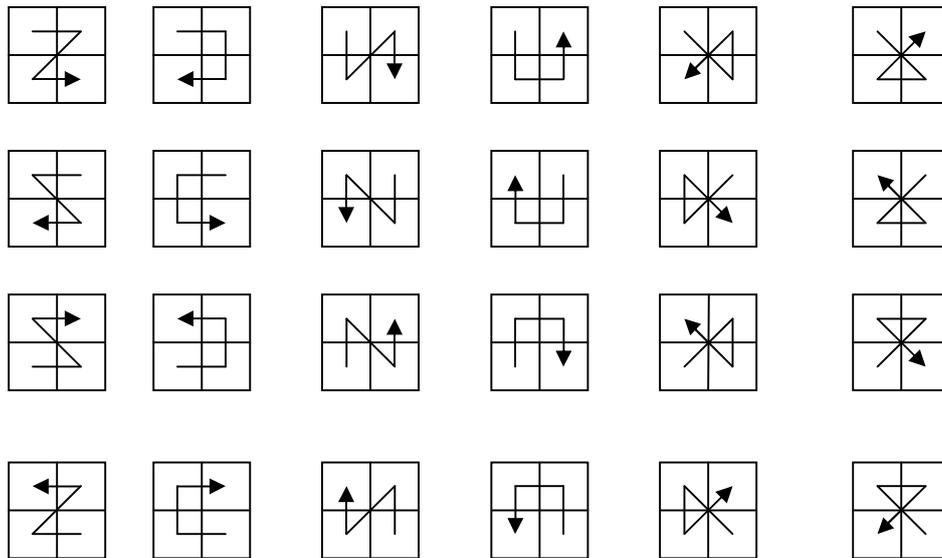
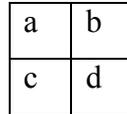


Figure III.2 : Parcours locaux possibles pour 4 points.

Sur la figure III.2 on peut remarquer que certain motifs sont redondant et différent que par les sens du parcours. Etant donné que le sens n'a aucune incidence sur les propriétés texturales, il est alors possible de réduire le nombre de motifs de moitié, soit 12 dans e cas de quatre pixels (figure III.3).

Chaque motif est décrit par le symbole formé, l'exposant (-1) indique que le symbole est inversé. D'où la définition suivante :

Définition n°2 :

Pour tous les points $\in g$, les 24 différents motifs sont réduits à 12.

$$M_t = \{Z, C, N, U, \alpha, \gamma, Z^{-1}, C^{-1}, N^{-1}, U^{-1}, \alpha^{-1}, \gamma^{-1}\} \tag{III.1}$$

M_t représente le nouvel ensemble de parcours locaux qui engendre tous les sous ensembles de motifs (figure III.3).

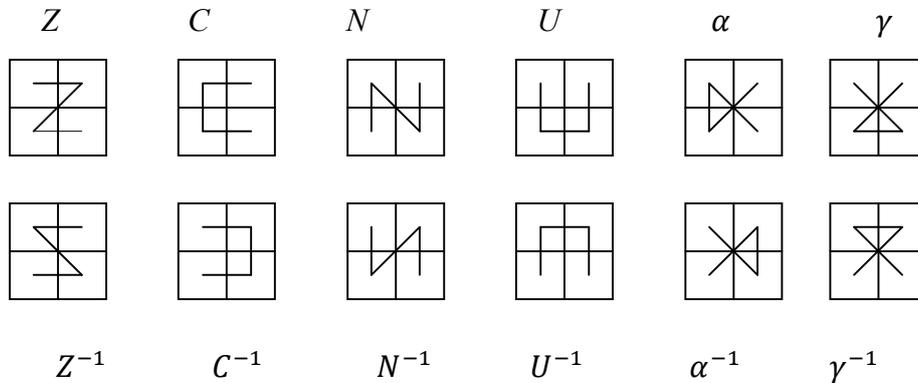


Figure III.3 : Motifs représentant tous les cas possibles.

Remarques

- En fixant le point de départ on obtient un sous ensemble de 6 motifs de base.
- La symétrie fait que les motifs s'apparient par sens de parcours.

III.3 Parcours optimal

Dans une image, chaque bloc de 4 pixels peut être décrit par un des 12 motifs de la figure III.3. La question qui se pose alors, c'est comment choisir le motif approprié. Rappelons que ce motif a pour objectif de définir la texture localement. Le choix du motif approprié est fait selon un critère d'optimalité qui exploite les variations des niveaux de gris dans le voisinage.

On considère l'intensité lumineuse d'un pixel comme un niveau d'énergie, et en supposant qu'une particule doit parcourir 4 points une seule fois. Cette particule doit choisir le parcours qui lui permettra de fournir le moindre effort (énergie). Le changement de niveau entraîne une perte d'énergie pour la particule, cette perte est liée aux différences entre niveaux. Une grande différence se traduit par une grande perte d'énergie et vice versa. Par conséquent la particule cherchera le chemin optimal de manière à réduire les transitions rapides des niveaux énergétiques. Le parcours doit engendrer la courbe la plus régulière et la plus adéquate parmi celles de l'ensemble.

Définition n°3 :

Un parcours optimal est défini comme le chemin qui minimise la variation d'intensité lumineuse le long de la courbe. Soit quatre pixels adjacents

P1	P2
P3	P4

le parcours optimal suit la permutation r^* qui correspond à :

$$\delta = \min_r \{ |p_{r_1} - p_{r_2}| + |p_{r_2} - p_{r_3}| + |p_{r_3} - p_{r_4}| \} \quad (\text{III.2})$$

La valeur minimale de δ détermine un parcours optimal. Par conséquent, un des 12 motifs de base représentera le bloc 2×2 pixels. L'application de ces parcours sur toute l'image produit l'image de motifs associée. Ce codage produit toujours la meilleure disposition entre les pixels voisins en défavorisant les transitions rapides. Il constitue localement un attribut de texture. En effet les motifs sélectionnés sont déterminés par les relations entre des pixels voisins.

Remarque :

Il y a des cas où plusieurs motifs peuvent parcourir la grille d'une manière optimale. On prend comme exemple le cas particulier où les quatre pixels du bloc sont identiques, c'est-à-dire, ils portent la même intensité lumineuse, donc tous les parcours sont optimaux.

III.4 Image de motifs

Il existe a priori deux variantes pour obtenir une image de motifs. Dans la première méthode, l'image est divisé en grille 2×2 pixels où chaque élément de la grille (bloc) est traversé par un parcours optimal est remplacée ainsi par un motif sans recouvrement. La taille de l'image résultat est réduite d'un facteur 2 en largeur et en hauteur par rapport à l'image originale. La deuxième méthode consiste à extraire pour chaque pixel et ses 3 voisins immédiats le motif correspondant. Ceci résulte en un opérateur local dont le résultat peut être par exemple un code du motif retenu. Par convention on commence par le point le plus haut à gauche.

La figure III.4 montre un exemple de calcul de l'image des motifs par la 1^{ère} méthode

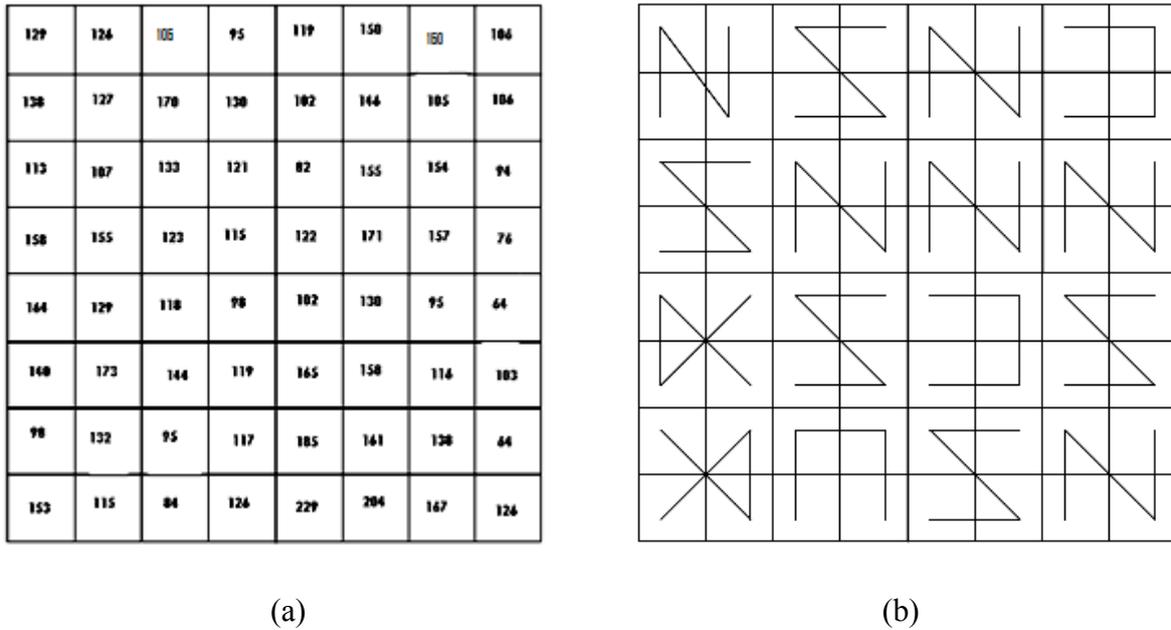


Figure III.4 : Exemple de calcul d'une image motif par la première méthode. (a) image originale, (b) image motif

La figure III.5 illustre le principe de calcul de l'image motif par la 2^{ème} méthode.

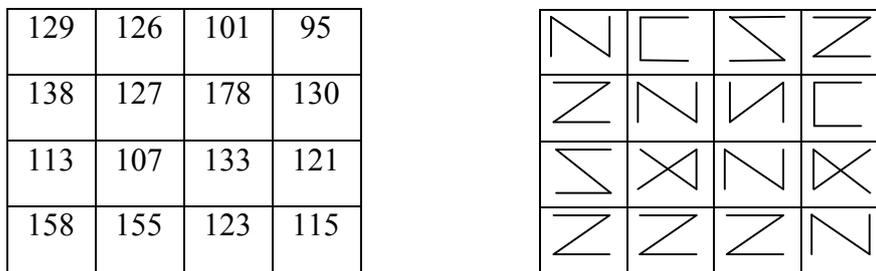


Figure III.5 : Exemple de calcul d'une image motif par la 2^{ème} méthode. (a) image originale,(b) image motif.

III.4.1 Matrice de cooccurrence des motifs (MCM)

Les parcours locaux nous offrent une primitive locale de texture en déterminant des relations entre les pixels parcourus. Cependant pour caractériser la texture de manière plus robuste, il est nécessaire d'extraire une information de texture d'une manière globale en exploitant la distribution spatiale des motifs. Pour cela les matrices de cooccurrence sont les plus indiquées. Les relations spatiales entre les paires de motifs sont déterminées en mesurant la probabilité qu'un motif soit associé à un autre. Un terme de la matrice de cooccurrence des motifs (MCM) est défini comme la probabilité de trouver un « *motif i* » à une distance donnée

d du « motif j ». Par conséquent, la MCM généralise la matrice de cooccurrence de niveaux de gris (MCNG). La distance d est définie selon les coordonnées polaires (d, θ) discrètes. En pratique θ prend les valeurs de 0° à 135° par pas de 45° . Comme le nombre de motifs est de 12, le vecteur de caractéristiques est représenté dans un espace de dimension 12×12 la MCM offre ainsi un avantage de compression par rapport à la MCNG dont la taille peut aller jusqu'à de 256×256 . Le fait de transformer d'abord les intensités lumineuses en une description plus symbolique engendre un ordre statistique supérieur au niveau des données et par conséquent la MCM code le troisième ordre statistique de l'intensité lumineuse alors que MCNG ne code que le deuxième ordre.

III.4.2 Propriétés de MCM

III.4.2.a Invariance au changement de luminance

Le changement des conditions d'illumination influence en général l'extraction des caractéristiques de l'image. Les attributs de textures extraits à partir de la MCNG sont affectés par le changement de luminance. Cela peut engendrer certaines incohérences dans l'information extraite (contours artificiels, mirages etc.). Il est intéressant de garantir une certaine consistance du phénomène. Supposons que la lumière incidente est uniforme et que le changement dans les conditions d'éclairage est monotone. On peut facilement vérifier que dans un tel cas l'histogramme de l'image ou MCNG seront affectés par ce changement. L'avantage de la MCM est d'être justement invariant par rapport à ces conditions. En effet supposant que c 'est une constante additive de l'intensité lumineuse, pour calculer le parcours optimal on se réfère à l'équation (III.2) ; on a donc :

$$\delta = \min_r \{ |(p_{r1} + c) - (p_{r2} + c)| + |(p_{r2} + c) - (p_{r3} + c)| + |(p_{r3} + c) - (p_{r4} + c)| \} \quad (III.3)$$

Il est évident que l'équation (III.3) est équivalente à l'équation (III.2). Dans les deux cas il en résulte le même motif.

III.4.2.b Invariance à la translation

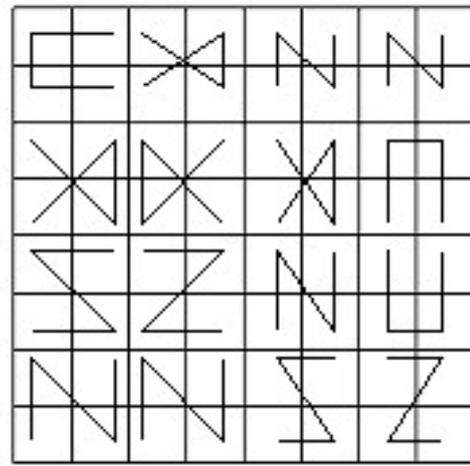
La MCM calculée sans recouvrement est très sensible aux translations. Une image décalée d'un nombre impair de pixels peut engendrer deux MCM différentes. En effet le décalage change les positions relatives des pixels voisins considérés sur la grille. Les parcours obtenus sont alors différents de ceux de l'image initiale (figure III.6). Cela produit des motifs différents et par conséquent leur distribution statistique conjointe en sera affectée. Afin

d'éviter ce problème, une solution consiste à décaler l'image originale d'un pixel horizontalement, verticalement et diagonalement afin de cerner tout les cas possibles. Pour chaque translation on calcule une MCM. On obtient 4 MCM dont, en cas de décalage, l'une correspond à l'originale.

Par contre pour une image motif obtenue en utilisant la 2^{ème} méthode (sans recouvrement), la MCM est invariante aux translations car tous les pixels sont visités et la relation entre voisins n'est pas affectée.

106	101	95	119	160	119	106	129
127	170	130	102	146	105	106	138
107	138	121	62	155	164	94	113
153	123	115	122	171	157	76	158
129	118	98	102	130	95	64	164
173	144	119	165	158	116	103	140
131	96	112	185	161	138	64	98
115	84	128	229	204	167	126	153

(a)



(b)

Figure III.6 : Effet de la translation. (a) Image après un décalage horizontal à gauche de l'image originale de la figure III.4, (b) Image motif.

III.4.3 Filtre de Gabor de l'image des motifs

On peut également envisager l'analyse de l'image des motifs en utilisant des filtres de Gabor. Ceux-ci nous permettrons de mieux apprécier l'effet du filtrage par les motifs sur l'image originale.

Il est alors possible de calculer des attributs de texture en calculant la valeur moyenne et la variance sur chaque image énergie de l'image filtrée.

III.5 Description d'une image par les valeurs des parcours optimaux

Dans cette section, nous allons introduire une autre technique d'analyse de la texture d'images qui est basée également sur les parcours locaux mais qui diffère de l'image des motifs en termes de représentation. En effet, chaque parcours appliqué sur un voisinage de 4 pixels engendre une valeur de parcours optimaux (VPO) désigné par δ (Equation III.2). Cette valeur peut être considérée comme un indice de variation des niveaux de gris dans un voisinage donné. Une grande valeur de δ traduit une forte variation des niveaux de gris et une faible valeur indique plutôt l'existence d'un voisinage lisse. Dans le cas particulier où δ est nul tous les pixels parcourus ont la même valeur. Il est donc intéressant de représenter un bloc de pixels par la valeur de δ . Cette représentation offre ainsi l'avantage d'être robuste au changement monotone de l'intensité comme il a été décrit dans la section III.5.1. Cette transformation peut également se concevoir comme un filtrage optimal obtenu par convolution avec l'un des 6 masques organisés selon 12 combinaisons (12 parcours). La figure III.5 montre le résultat du filtrage d'une image organisée en blocs (sans recouvrement) de 4 pixels (figure III.5.a) avec les masques choisis d'une manière optimale (figure III.5.b).

III.5.1 Matrice des Cooccurrences des VPO

Pour décrire la texture de l'image d'une manière globale, on peut calculer la matrice de cooccurrence à partir de l'image des VPO, qui sera notée (MCVPO). Les calculs sont effectués de la même manière que pour MCNG ou MCM. En revanche le principe résultant est différent puisqu'on détermine en fait la relation spatiale entre des indices de la variation des niveaux de gris. L'occurrence de δ est indépendante de l'occurrence des niveaux de gris. Par exemple, pour un groupe de pixels tels que $p_1=p_2=p_3=p_4=c$, $\delta=0$ quelque soit la valeur de c , alors que les valeurs d'une MCNG dépendent de la valeur de c .

III.5.2 Filtre de Gabor de l'image VPO

Pour mieux apprécier le changement dans la répartition spectrale, nous pouvons appliquer les filtres de Gabor sur les images de valeurs des parcours optimales (VPO). On constate que la transformation en VPO implique un changement de fréquence spatiale car le critère d'optimalité a pour but de régulariser en réduisant les faits des transitions rapides dans un voisinage. En effet la transformation en VPO remplace globalement les niveaux

par des gradients (équation III.2) donc privilégie les hautes fréquences spatiales. Une description efficace par filtre de Gabor de l'image originale nécessite une bonne résolution en fréquence et une bonne préservation des contours. Ces deux objectifs sont contradictoires, il peut s'avérer difficile de conjuguer les deux propriétés. L'image VPO réalise un certain compromis entre les deux critères. Une fois l'image filtré, on lui applique une fonction non linéaire pour calculer l'énergie de chaque pixels (équation III.3) et le résultat est prit pour caractéristique de la texture.

III .6 Conclusion

Dans ce chapitre, nous avons présenté une méthode d'analyse de la texture en se basant sur le codage de l'image par un motif ou par une valeur d'un parcours localement optimal.

L'application des techniques classiques, telles que la méthode basée sur les matrices de cooccurrence ou celle basée sur les filtres de Gabor, aux images des motifs ou des valeurs de parcours optimaux nous permettra d'extraire des attributs de texture.

La pertinence de ces attributs à discriminer des textures différentes sera évaluée dans le prochain chapitre.

IV.1 Introduction [10]

Nous allons étudier dans ce chapitre la capacité des attributs de texture, extraits à partir des matrices de cooccurrence et ceux déterminés par les filtres de Gabor, à discriminer les textures différentes.

Nous allons, également, étudier l'intérêt d'utiliser les images motifs où les images VPO.

Pour mener à bien cette étude, nous allons comparer les attributs de plusieurs images ayant des textures différentes, où procéder à la segmentation d'une image composées de plusieurs textures.

Les méthodes d'analyse de la texture basées sur les matrices de cooccurrence et les filtres de Gabor ainsi que le calcul des images motifs et VPO ont été implémenté sous l'environnement Matlab version 7.4. Notre choix est motivé par les plusieurs avantages de ce langage. Entre autre, nous pouvons citer :

- Un logiciel de calcul scientifique et de visualisation de données, dédié plus pratiquement aux applications numériques.
- Il a été conçu pour manipuler des données matricielles.

Matlab est aujourd'hui un langage de programmation complet dans un environnement de développement simple et pratique.

IV.2 Création d'une base d'images

Pour effectuer une comparaison entre les attributs de texture sur différentes textures, nous avons créé une base d'images de texture à partir de l'album de Brodatz.

Cette base contient 10 textures différentes. Il s'agit des textures D112, D74, D54, D35, D26, D24, D20, D11, D5 et D4 (figure IV.1). Celles-ci se différencient par leurs aspects microscopique et macroscopique ayant des propriétés différentes.

A partir de chaque image originale de taille (640×640), nous avons extraits différentes images, soit 30 échantillons de taille (512×512) à différents endroits de l'image originale.

Ensuite, nous avons redimensionné chacun de ces échantillons en imagerie de taille (64×64) ce qui nous donne au total 300 échantillons, soit 30 pour chaque texture.

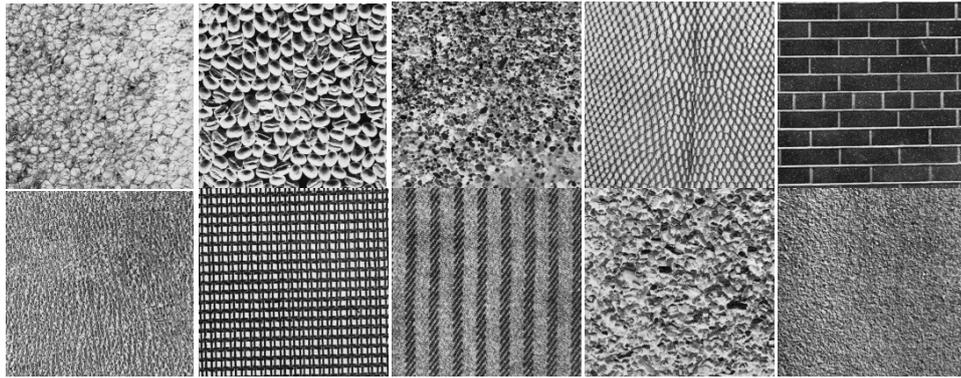


Figure IV.1 : Images test de gauche à droite et de haut en bas (D112, D74, D54, D35, D26, D24, D20, D11, D5, D4).

Par contre pour la segmentation, nous avons utilisé une image de taille (128×128), composée de 3 textures tirée de l'album de Brodatz, à savoir D24, D34 et D78 (Figure IV.2).

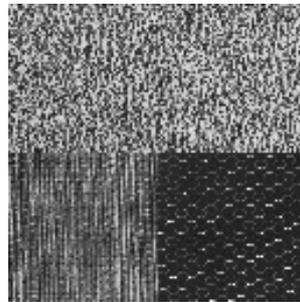


Figure IV.2 : Image à segmenter

IV.3 Images motifs et images VPO

Pour chaque image de la figure IV.1 et IV.2, nous avons déterminé le parcours optimal et ce pour chaque pixel (c.-à-d. en utilisant la 2^{ème} méthode).

Les figures IV.3 et IV.4 montrent respectivement les images motifs et les images VPO obtenues à partir des images de la figure IV.1.

La figure IV.5 montre l'image des motifs et celle des VPO de la figure IV.2

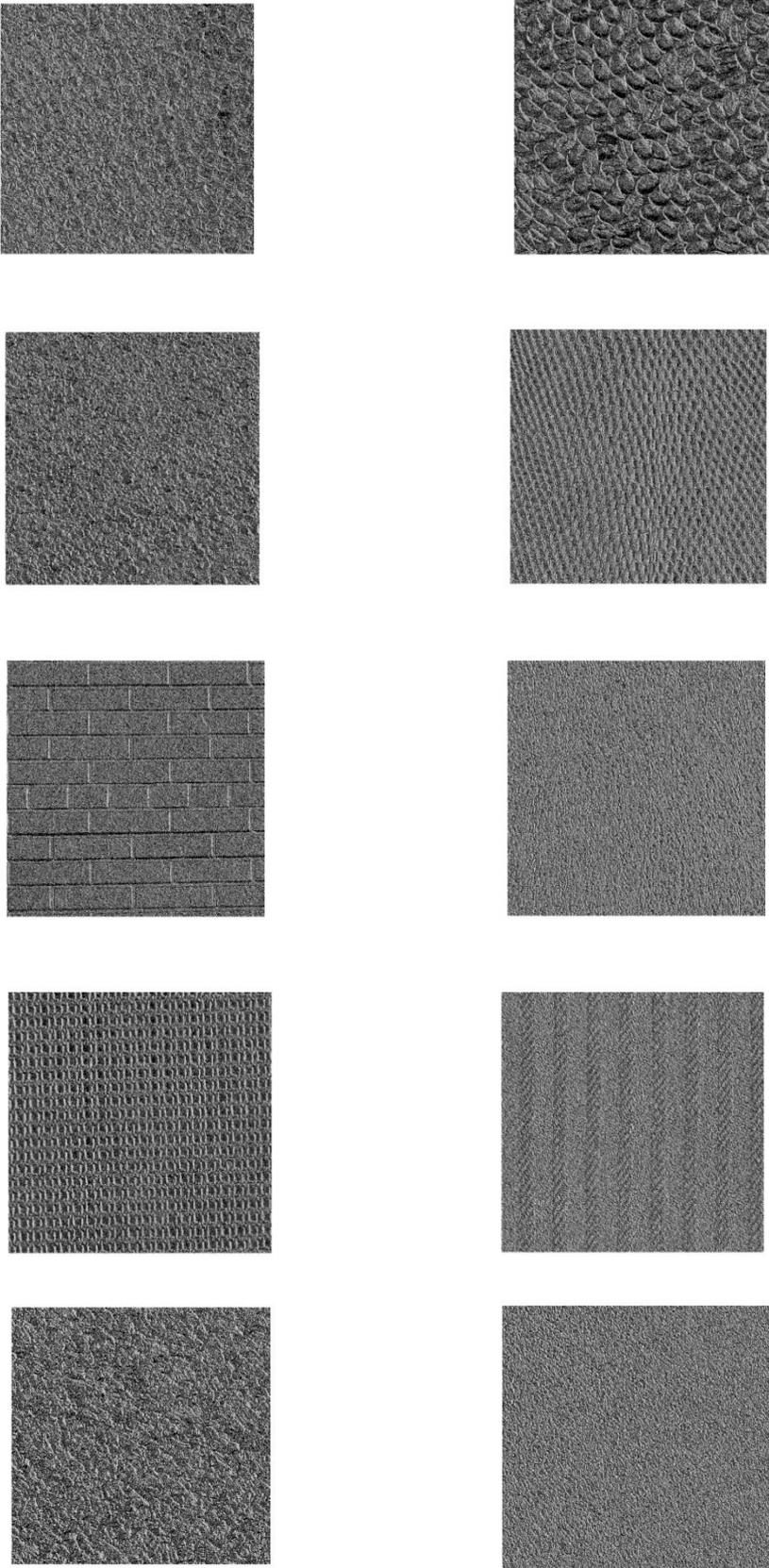


Figure IV.3 : Les images motifs des 10 images test de la figure IV.1.

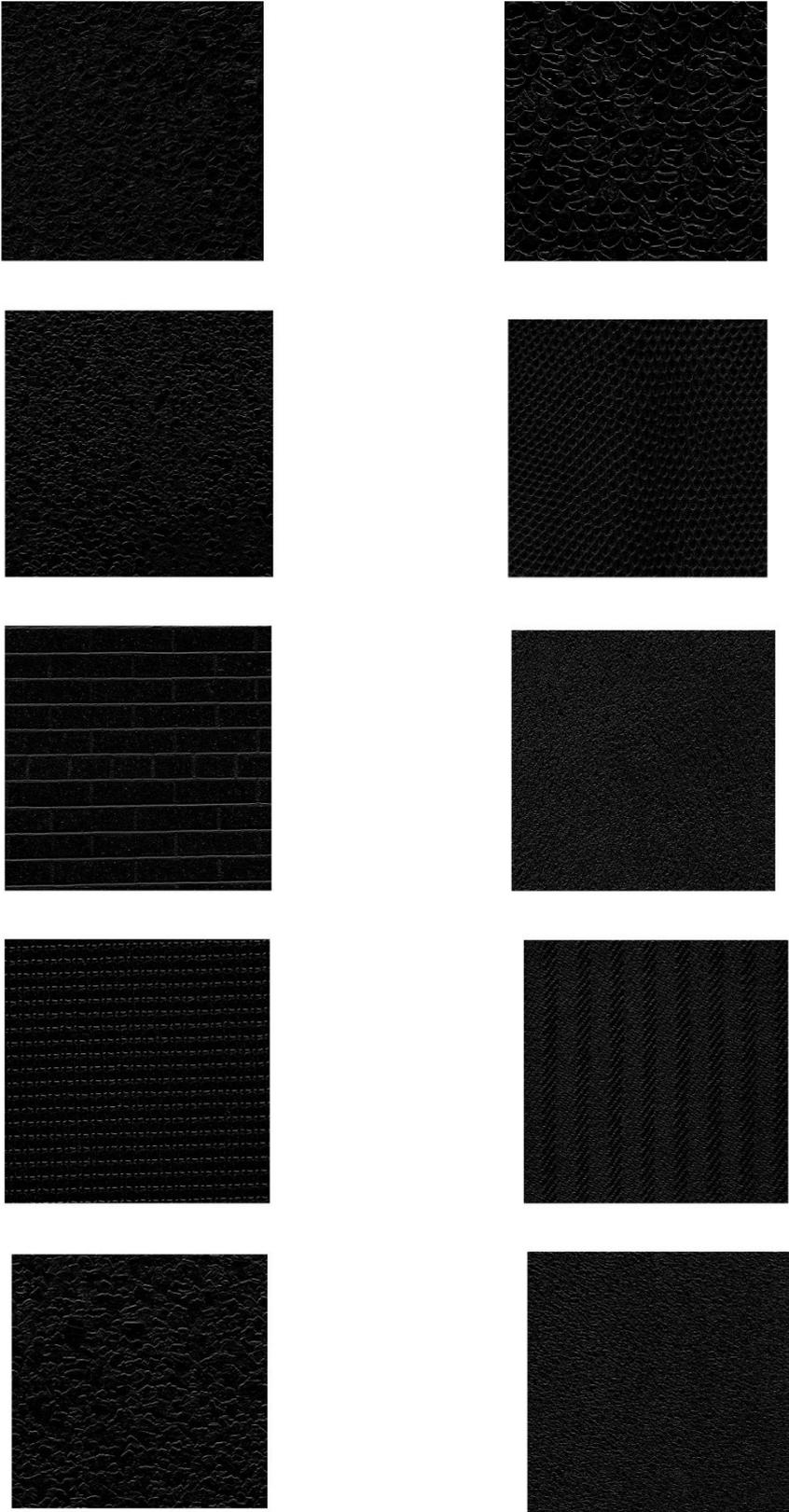


Figure IV.4 : Les images VPO des 10 images de la figure IV.1



Figure IV.5: (a) Image motifs, (b) Image des VPO de la figure IV.2.

Interprétation

Nous remarquons que les images VPO reproduisent fidèlement les variations des niveaux gris puisque les régions noirs foncées indiquent une petite variation où une variation nulle des niveaux de gris pendant que les zones claires ou blanches, indiquent une forte variation des niveaux de gris.

Tandis que les images motifs font clairement apparaître les textures de l'image.

IV.4 Comparaison des attributs de texture extraits à partir des matrices de cooccurrence

Pour étudier le pouvoir discriminant des attributs de texture extraits à partir des matrices de cooccurrence, nous avons déterminé les attributs de texture à partir de la matrice de cooccurrence de l'image motifs (MCM).

Ensuite, nous avons déterminé les valeurs moyennes des attributs appartenant à la même texture.

Le tableau IV.1 donne les valeurs moyennes des attributs de textures calculées à partir de MCM.

Notons que ces attributs correspondent au second moment angulaire (E), contraste (Cont), entropie (Ent), moment de différence inverse (Hom) et autocorrélation (Aut).

	E	Cont	Ent	Hom	Aut
D4	0.0169	15.4332	3.8303	0.2173	44.3548
D5	0.0170	15.6059	4.0300	0.2654	43.7213
D11	0.0179	14.0133	3.6445	0.2639	45.9501
D20	0.0373	15.0537	3.6205	0.2505	37.4434
D24	0.0219	16.7378	3.7142	0.2122	44.2004
D26	0.0212	16.4652	4.0245	0.2690	39.9047
D35	0.0304	16.1362	3.4573	0.2240	46.0502
D54	0.0167	15.7965	4.0098	0.2561	44.1424
D74	0.0169	15.9542	4.0035	0.2877	44.0196
D112	0.0194	15.7838	3.9517	0.2692	42.7729

Tableau IV.1 : Valeurs moyennes des attributs de texture extraites à partir de MCM

Interprétation

Les valeurs des différents attributs diffèrent d'une texture à l'autre. Les textures qui apparaissent visuellement identiques (D54 et D5) ou (D24 et D4) ont des attributs différents. Ceci démontre que les attributs extraits de la MCM permettent de discriminer convenablement les textures.

IV.5 Comparaison des attributs de texture issus des filtres de Gabor

Comme pour les matrices de cooccurrence, nous avons déterminé pour chaque image de la base, des attributs de texture extraits à partir des images filtrées, après application des filtres de Gabor.

Notons que ces attributs correspondent à la moyenne et la variance des images énergies (voir paragraphe I.10).

A titre d'exemple, les figures IV.6, IV.7 et IV.8 montrent quelques images énergies obtenues après application des filtres de Gabor avec différentes valeurs de f , σ_x , σ_y et θ sur l'image D4, les images motifs et VPO.

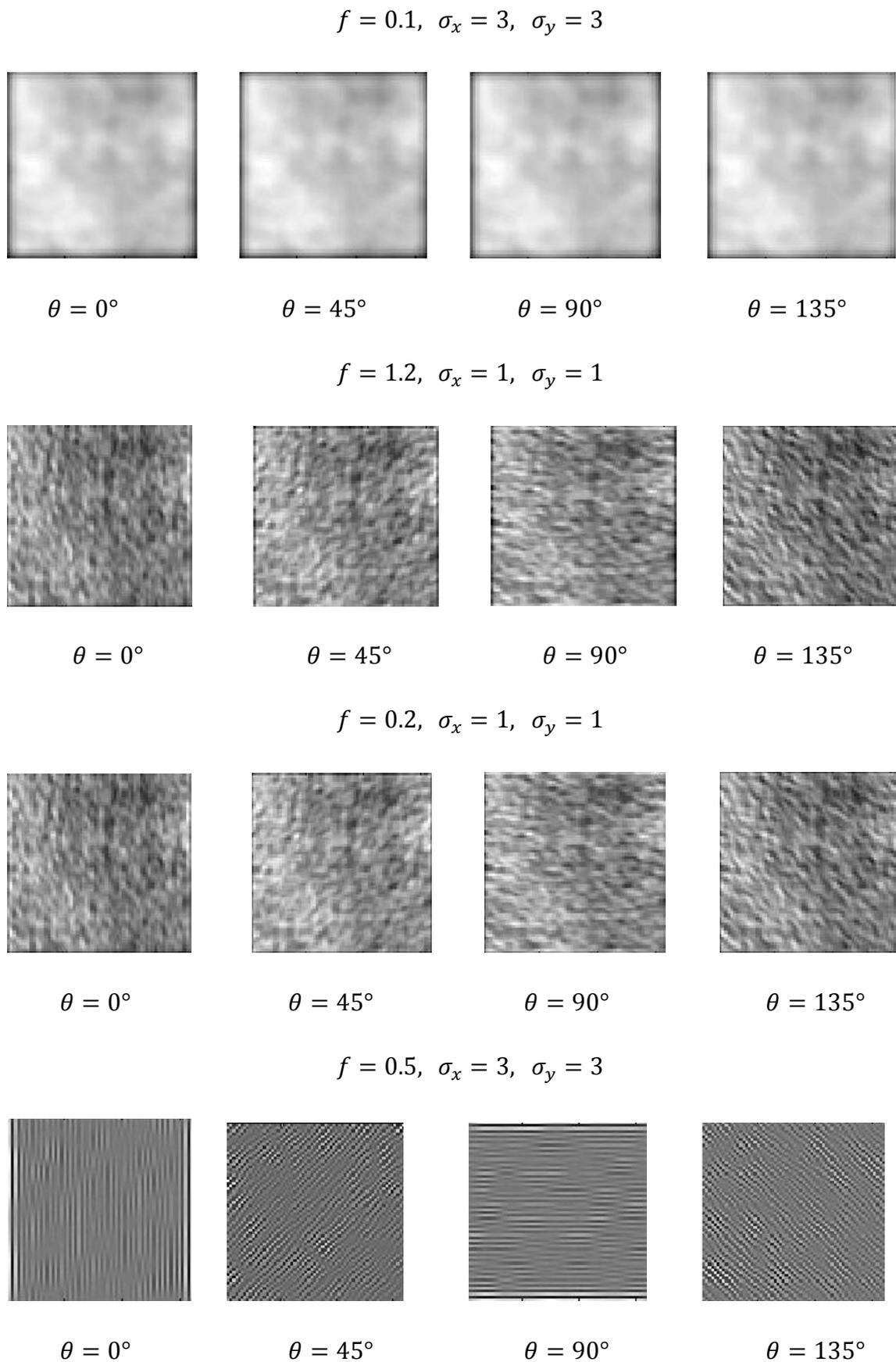
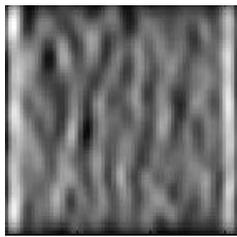
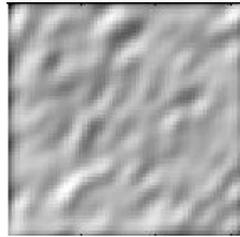


Figure IV.6 : Images énergie obtenues par application des filtres de Gabor sur l'image D4

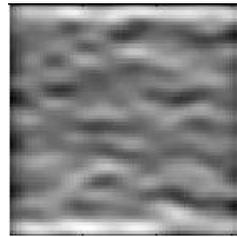
$$f = 0.1, \sigma_x = 3, \sigma_y = 3$$



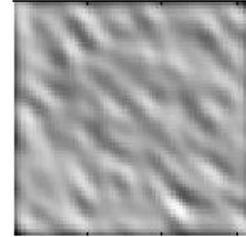
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

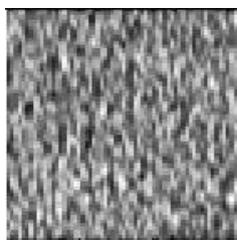


$$\theta = 90^\circ$$

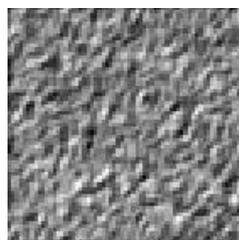


$$\theta = 135^\circ$$

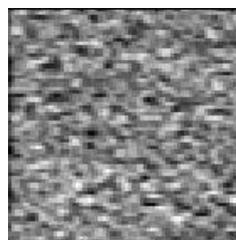
$$f = 1.2, \sigma_x = 1, \sigma_y = 1$$



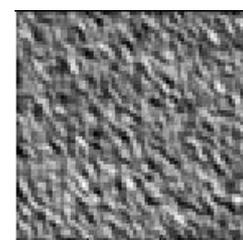
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

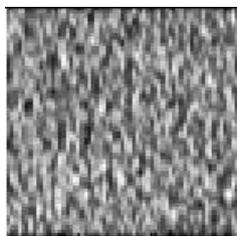


$$\theta = 90^\circ$$

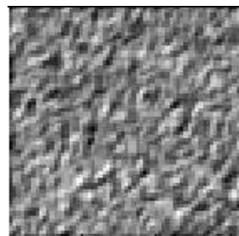


$$\theta = 135^\circ$$

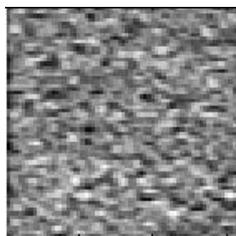
$$f = 0.2, \sigma_x = 1, \sigma_y = 1$$



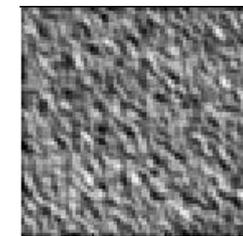
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

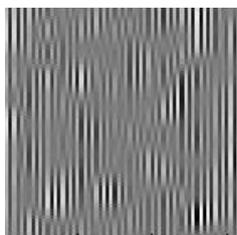


$$\theta = 90^\circ$$

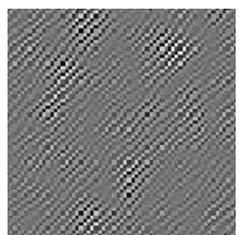


$$\theta = 135^\circ$$

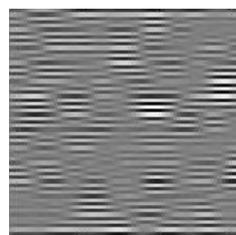
$$f = 0.5, \sigma_x = 3, \sigma_y = 3$$



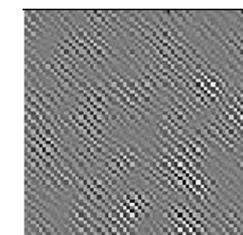
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$



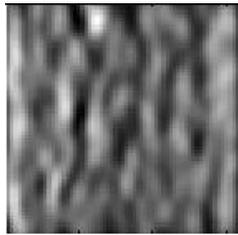
$$\theta = 90^\circ$$



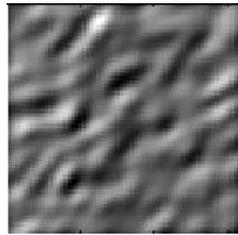
$$\theta = 135^\circ$$

Figure IV.7 : Images énergie obtenues par application des filtres de Gabor sur l'image motifs de D4

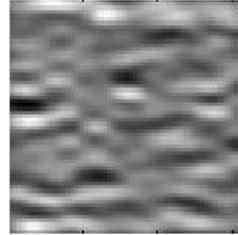
$$f = 0.1, \sigma_x = 3, \sigma_y = 3$$



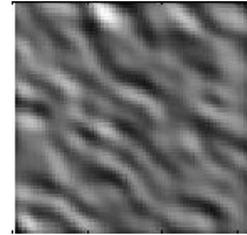
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

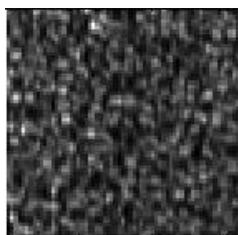


$$\theta = 90^\circ$$

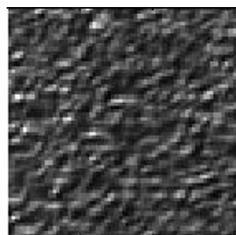


$$\theta = 135^\circ$$

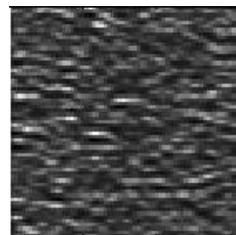
$$f = 1.2, \sigma_x = 1, \sigma_y = 1$$



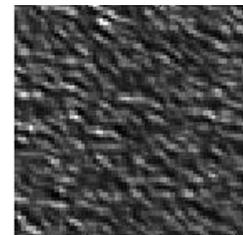
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

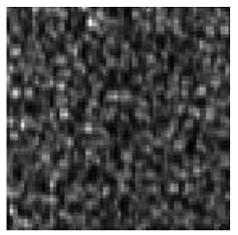


$$\theta = 90^\circ$$

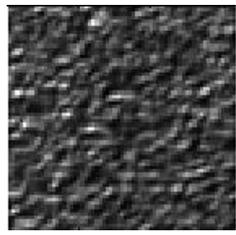


$$\theta = 135^\circ$$

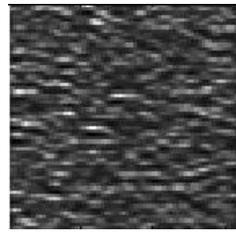
$$f = 0.2, \sigma_x = 1, \sigma_y = 1$$



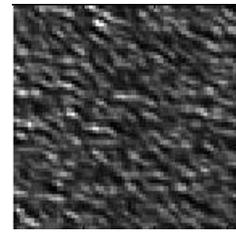
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$

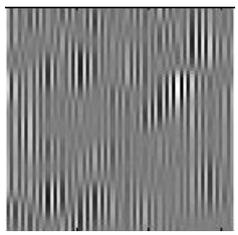


$$\theta = 90^\circ$$

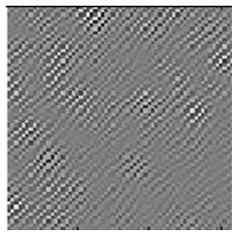


$$\theta = 135^\circ$$

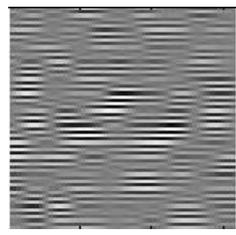
$$f = 0.5, \sigma_x = 3, \sigma_y = 3$$



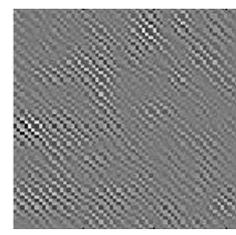
$$\theta = 0^\circ$$



$$\theta = 45^\circ$$



$$\theta = 90^\circ$$



$$\theta = 135^\circ$$

Figure IV.8 : Images énergie obtenues par application des filtres de Gabor sur l'image VPO de D4

Interprétation

Nous remarquons que le changement d'orientation θ , change considérablement l'image filtrée obtenue du moment que les textures détectées par les différents filtres sont celles qui sont opposées à l'orientation choisie (ex : une texture orienté à 90° est détecté par un filtre orienté à 0°).

La variation des écarts types σ_x , σ_y varient la largeur de bande du filtre ce qui varie la largeur entre un pixel détecté et un autre.

Quant à la fréquence f , elle définit la sensibilité des filtres donc elle a une influence directe sur la résolution des images obtenues.

Les tableaux IV.2 à IV.4 donnent les valeurs moyennes pour chaque texture de la moyenne (μ) et de la variance (s) des images énergie obtenues par l'application des filtres de Gabor sur les images de la base de test, leurs images motifs et leurs images VPO, respectivement.

Pour chaque image, on lui applique 4 filtres ayant la même fréquence $f=0.001$ et une même variance $\sigma_x = 5$ et $\sigma_y = 5$. Ces filtres se diffèrent que par l'orientation θ ($\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$). Chaque image analysée sera alors caractérisé par 8 attributs.

	$\theta = 0^\circ$		$\theta = 45^\circ$		$\theta = 90^\circ$		$\theta = 135^\circ$	
	μ $\times 10^{-4}$	s $\times 10^{-4}$						
D4	1.6402	0.3064	1.6402	0.3063	1.6402	0.3063	1.6402	0.3064
D5	1.8536	0.3371	1.8536	0.3371	1.8536	0.3371	1.8536	0.3371
D11	1.6057	0.3018	1.6057	0.3018	1.6057	0.3018	1.6057	0.3018
D20	1.4372	0.2680	1.4372	0.2680	1.4372	0.2680	1.4372	0.2680
D24	1.7933	0.3347	1.7933	0.3347	1.7933	0.3347	1.7933	0.3347
D26	9.3909	1.7808	9.3909	1.7808	9.3909	1.7808	9.3909	1.7808
D35	2.0268	0.3606	2.0268	0.3606	2.0268	0.3606	2.0268	0.3606
D54	1.7009	0.3203	1.7009	0.3203	1.7009	0.3203	1.7009	0.3203
D74	1.9058	0.3665	1.9058	0.3665	1.9058	0.3665	1.9058	0.3665
D112	2.2985	0.4411	0.2985	0.4411	2.2985	0.4411	2.2985	0.4411

Tableau IV.2 : Valeurs moyennes des attributs de texture issus des filtres de Gabor appliquées aux images originales.

	$\theta = 0^\circ$		$\theta = 45^\circ$		$\theta = 90^\circ$		$\theta = 135^\circ$	
	μ	s	μ	s	μ	s	μ	s
D4	879.0195	158.2918	879.0195	158.2919	879.0195	158.2919	879.0195	158.2918
D5	832.4272	151.1160	832.4272	151.1162	832.4270	151.1161	832.4271	151.1158
D11	916.0519	168.5089	916.0518	168.5087	916.0516	168.5083	916.0518	168.5085
D20	729.8329	134.7496	729.8329	134.7493	729.8329	134.7496	729.8329	134.7498
D24	872.5823	158.7766	872.5823	158.7768	872.5823	158.7771	872.5823	158.7769
D26	756.3903	139.7689	756.3902	139.7684	756.3901	139.7676	756.3902	139.7682
D35	885.5805	162.8504	885.5805	162.8501	885.5805	162.8500	885.5805	162.8504
D54	842.9955	154.8330	842.9953	154.8327	842.9952	154.8326	842.9954	154.8329
D74	837.9797	153.6887	837.9797	153.6893	837.9797	153.6897	837.9797	153.6891
D112	824.6906	151.1803	824.6906	151.1803	824.6907	151.1808	824.6907	151.1808

Tableau IV.3 : Valeurs moyennes des attributs de texture issus des filtres de Gabor appliquées aux images motifs.

	$\theta = 0^\circ$		$\theta = 45^\circ$		$\theta = 90^\circ$		$\theta = 135^\circ$	
	μ	s	μ	s	μ	s	μ	s
D4	956.7709	189.0975	956.7706	189.0976	956.7704	189.0978	956.7708	189.0977
D5 ($\times 10^{+3}$)	2.3549	0.5219	2.3549	0.5219	2.3549	0.5220	2.3549	0.5220
D11	975.9631	194.8048	975.9636	194.8066	975.9641	194.8078	975.9636	194.8060
D20($\times 10^{+3}$)	2.8193	0.5646	2.8193	0.5646	2.8193	0.5646	2.8193	0.5646
D24	873.2758	185.7841	873.2757	185.7850	873.2756	185.7831	873.2757	185.7823
D26($\times 10^{+3}$)	1.7799	0.3635	1.7799	0.3635	1.7799	0.3636	1.7799	0.3635
D35($\times 10^{+3}$)	1.6438	0.3484	1.6438	0.3484	1.6438	0.3484	1.6438	0.3484
D54($\times 10^{+3}$)	2.1011	0.4497	2.1011	0.4498	2.1011	0.4497	2.1011	0.4497
D74($\times 10^{+3}$)	3.2867	0.6320	3.2867	0.6320	3.2867	0.6320	3.2867	0.6320
D112($\times 10^{+3}$)	1.5446	0.3186	1.5446	0.3186	1.5446	0.3186	1.5446	0.3186

Tableau IV.4 : Valeurs moyennes des attributs de texture issus des filtres de Gabor appliquées aux images VPO.

Interprétation

On remarque approximativement la même chose que pour la MCM sauf que pour les filtres de Gabor appliqués aux images VPO et aux images motifs, les valeurs des différents attributs présentent une différence éminente d'une texture à l'autre. Ceci démontre que les attributs extraits à partir des filtres de Gabor appliqués aux images VPO et motifs permettent de discriminer plus pertinemment les textures, en particulier les filtres appliqués aux images VPO.

IV.6 Segmentation

Pour segmenter une image, nous avons déterminé un ensemble d'attributs pour chaque pixel. Ensuite, nous avons appliqué la méthode de classification non supervisée FCM, décrite au deuxième chapitre, afin de regrouper les pixels en zones homogènes. Les attributs de texture de chaque pixel sont extraits à partir des matrices de cooccurrence des niveaux de gris des pixels appartenant au voisinage du pixel traité. Ces pixels sont ceux situés à l'intérieur d'une fenêtre de voisinage de taille $(2w+1 \times 2w+1)$, centrée sur chaque pixel à traiter.

La figure IV.9 montre les images segmentées de l'image motifs de la figure IV.2 avec différentes valeurs de w .

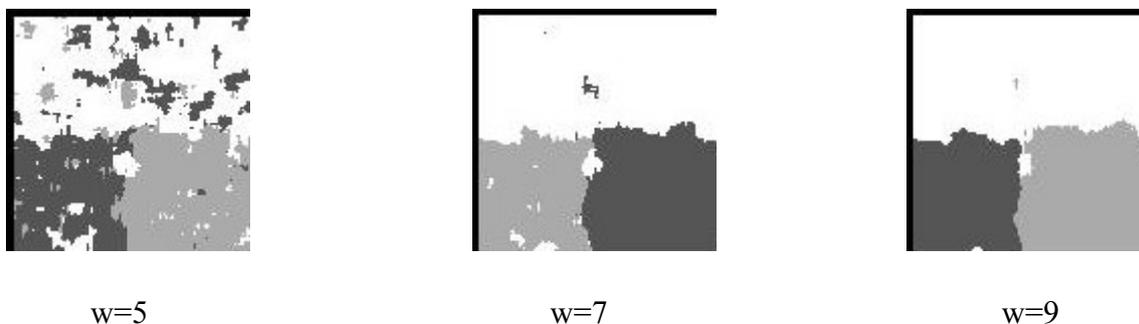


Figure IV.9 : Images segmentée à partir des attributs extraits de MCNG.

Interprétation

Nous pouvons remarquer que les trois zones texturées sont bien mises en évidence, plus la taille de la fenêtre de voisinage w augmente, plus le résultat de la segmentation est meilleure. Cependant, à partir d'une certaine valeur de w , les résultats deviennent médiocres. Il faut également noter que le fait d'utiliser l'image motif permet une segmentation plus rapide par rapport à celle de l'image originale ou VPO.

Par contre, pour les filtres de Gabor, nous avons appliquée 4 filtres ayant une même fréquence $f = 1,39$ une même variance $\sigma_x = \sigma_y = 3$ et des orientations différentes, $\theta = 0^\circ$ pour le 1^e, $\theta = 45^\circ$ pour le 2^e, $\theta = 90^\circ$ pour le 3^e et $\theta = 135^\circ$ pour le quatrième filtre.

Pour caractériser chaque pixel, nous avons calculé la valeur moyenne localement de chaque image énergie. Pour cela, on considère une fenêtre de voisinage de taille $(2w+1) \times (2w+1)$. Ainsi, chaque pixel calculé localement sur chaque image énergie.

La figure IV.10 montre le résultat du regroupement des pixels en 3 classes par l'algorithme FCM.

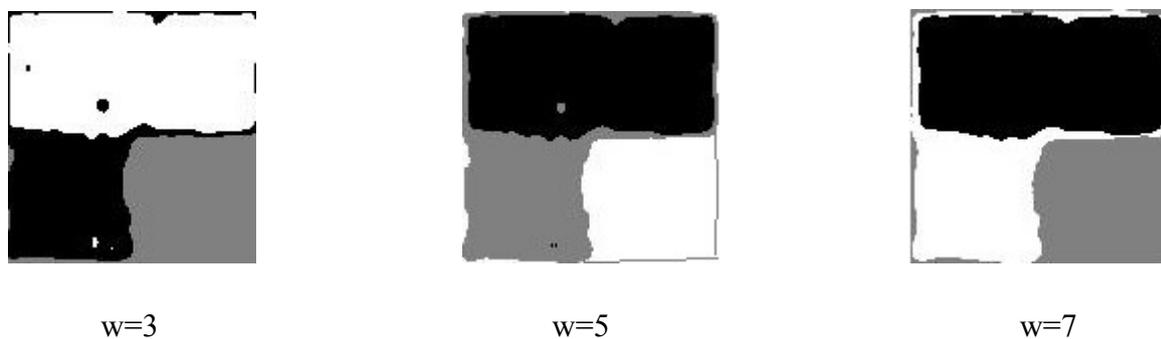


Figure IV.10 : Segmentation de l'image test en utilisant les attributs issus des filtres de Gabor.

Ici aussi, nous pouvons toujours remarquer une bonne détection des zones de même texture. On peut aussi voir une légère amélioration lorsqu'on augmente la taille de la fenêtre de voisinage de chaque pixel w , mais au-delà d'une certaine valeur les résultats deviennent assez mauvais.

IV.7 Conclusion

Cette partie expérimentale nous a permis de vérifier ce qu'on a avancé dans les chapitres précédents en constatant que travailler sur les images motifs ou VPO présente beaucoup d'avantages à savoir :

- C'est plus rapide que de travailler sur l'image originale.
- En travaillant sur les images VPO, nous avons constaté que les attributs obtenus, à partir des filtres de Gabor particulièrement sont très distincts d'une texture à l'autre ainsi le pouvoir de discrimination est plus fort par rapport à la MCM.
- La taille de la fenêtre de voisinage de chaque pixel a une influence directe sur le résultat de segmentation que ce soit par les filtres de Gabor ou par les matrices de cooccurrence des niveaux de gris.

Le travail, qui nous a été proposé, a pour objectif d'étudier la méthode d'analyse de la texture basée sur les parcours optimaux : les filtres de Gabor et les matrices de cooccurrence. Dans ce mémoire, nous avons présenté et détaillé ces deux techniques classiques qui ont pour but d'extraire les caractéristiques de la texture.

Dans le chapitre **I**, nous avons, tout d'abord, présenté quelques notions sur la texture d'images, et puis, nous avons donné les différentes méthodes les plus utilisées ou bien les plus connues dans l'analyse de la texture en détaillant l'essentiel sur les deux approches les plus populaires dans ce domaine qui sont : l'approche qui est basée sur les filtres de Gabor et celle basée sur les matrices de cooccurrence. Ces deux dernières permettent d'extraire les caractéristiques de n'importe quelle texture d'une image.

Nous avons introduit, dans le chapitre **II**, quelques rappels sur la segmentation d'images texturées, afin de mieux comprendre cette étape primordiale de traitement d'images. Nous avons, également, présenté les méthodes de segmentation en détaillant l'approche basée sur la classification des pixels (le seuillage). On utilise pour cette approche deux algorithmes de classification : le premier est l'algorithme C-Means qui classe les pixels dans des classes sans recouvrement, et le deuxième est l'algorithme des Fuzzy C-Means qui fait la même chose sauf que les classes résultantes sont avec recouvrement.

Nous avons consacré le chapitre **III** pour présenter une approche d'analyse de la texture en basant sur le codage d'une image par les motifs ou par une valeur d'un parcours local. Nous avons constaté que l'application des deux techniques classiques sur les images motifs nous permettra d'extraire les caractéristiques d'une texture.

Toutefois, ce travail nous a permis d'explorer le domaine du traitement d'images pour la première fois dans notre cursus et en particulier l'analyse et la segmentation d'images numériques.

Enfin, nous espérons que ce travail servira aux étudiants des promotions futures, et leur permettra d'aller plus loin dans ce domaine, en l'enrichissant par d'autres méthodes.