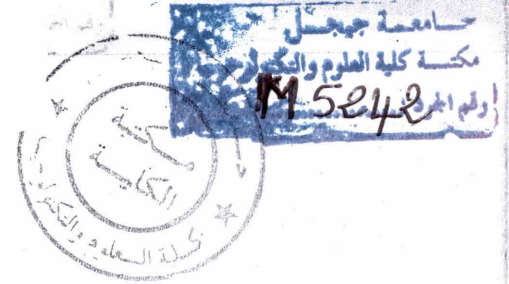


UNIVERSITE DE JIJEL



Faculté des Sciences et de la Technologie
Département d'Electronique

Mémoire présenté pour l'obtention du diplôme de
Master II en Electronique

Option :
Electronique et Analyse des Systèmes

Intitulé

**EXTRACTION ET CARACTÉRISATION DE
TEXTURES POUR LA CLASSIFICATION DES
IMAGES**

Encadré par :

✦ M^{me} BOUATMANE Sabrina

Présentés par :

✦ HAMDELLOU Haroun
✦ BOUROUINA Rabeh

Promotion Juin 2014

*R*emerciements

*Louange à dieu. Bienfaiteur miséricordieux.
Paix et bénédiction sur son prophète, Mohamed, ultime envoyé.*

*Avant tout nous remercions le Dieu qui nous a donné le courage
et la force pour continuer. C'est grâce à lui que notre chemin
est éclairé pour finir ce modeste travail.*

*nous remercions très vivement notre encadreur
M^{me}. BOUATMANE Sabrina*

*de nous avoir encadré, orienté, conseillé et corrigé tout au long
de notre travail.*

*nous remercions également, les membres de jury qui ont accepté
d'examiner et de juger notre travail.*

*nos vifs remerciements vont également à tous ceux qui ont de
près ou de loin contribué à l'accomplissement de ce travail.*

*notre reconnaissance s'adresse à nos familles qui ont su nous
apporter, sans relâcher les soutiens durant toutes ces longues
années d'études.*

*Enfin, nous remercions également, tous les enseignants, qui ont
assuré notre formation durant notre cycle universitaire.*

A tous, un grand Merci.



Dédicace

Je dédie ce modeste travail :

À ma mère le symbole de tendresse.

À mon père le symbole d'espoir.

*À mes frères et mes sœurs, pour leur
aide, soutien et encouragement.*

À mon ami et mon voisin

« Imad Hezouat zin ».

*À mes chers frères « siff ,abderraouf »
ma cousine « lamia » et mon binôme
« ribouh »*

*À tous mes amis et collègues chacun
de son nom et de ses particularités.*

Haroun



Dédicace

Je dédie ce modeste travail :

À ma mère le symbole de tendresse.

À mon père le symbole d'espoir.

*À mes frères et mes sœurs, pour
leur aide, soutien et
encouragement.*

À mon binôme « haroun »

À mon ami « Ahmed chriki »

À ma cousine « yasmine »

*À tous mes amis et mes
collègues*

« imad ,slamo,rahim,abdellah »

À tous un grand merci.

Rabeh

Introduction général	1
Chapitre I : <i>Extraction des caractéristiques de texture</i>	
I.1. Introduction	3
I.2. Définition de la texture	3
I.3. Techniques d'analyse de texture	4
I.3.1. Méthodes structurelles.....	5
I.3.2. Analyse spectrale.....	5
I.3.3. Analyse spatio-fréquentielle.....	5
I.3.4. Approche multi-niveaux.....	5
I.3.5. Géométrie fractale.....	5
I.3.6. Modélisation.....	6
I.3.7. Matrices de co-occurrence.....	6
I.4. Description des méthodes utilisées	6
I.4.1. Matrices de cooccurrences.....	6
I.4.2. Filtrés de Gabor.....	9
I.4.3. Définition de la Transformée de Fourier (TF).....	10
I.4.3.1. Cas de signaux 2D.....	11
I.4.3.2. Interprétation du spectre de Fourier d'une image.....	12
I.4.4. Filtrés de Gabor.....	14
I.4.4.1. Cas 1D.....	15
I.4.4.2. Cas 2D.....	16
I.5. Conclusion	21
Chapitre II : <i>Classification des données</i>	
II.1. Introduction	22
II.2. Définition de classifieur	22
II.3. Construction d'un classifieur	23
II.3.1. Base d'apprentissage et base de test.....	24
II.3.2. Types d'informations fournies par les classifieurs.....	25
II.4. Les outils de classification	25
II.4.1. Classification non supervisée.....	26
II.4.2. Classification supervisée.....	27
II.5. Méthodes de classification	29

II.5.1.Méthode Bayésienne.....	29
II.5.2.Machines à vecteurs support (SVM)	30
II.5.3.Analyse discriminante descriptive	31
II.5.4.Réseaux de neurones.....	33
II.5.5.K-plus proches voisins.....	34
II.5.5.1.Règle des k – plus proches voisins (k-ppv).....	34
II.5.5.2.L’algorithme K-ppv	35
II.6.Evaluation de la classification	36
II.6.1. Validation croisée.....	36
II.6.2.La courbe de ROC.....	38
II.7.Conclusion :.....	40
Chapitre III : Classification des images texturées	
III.1.Introduction.....	41
III.2. Fusion d’informations ou combinaison d’informations	41
III.3.Présentation des bases de données	41
III.3.1. Description de la base Brodatz.....	41
III.3.2.Description de la base UIUCTex.....	43
III.4.Implémentation et résultats.....	43
III.4.1. Extraction des caractéristiques	43
III.4.1.1.Filtre de Gabor	43
III.4.1.2.La matrice de co-occurrence (Gray Level Co-Occurrence Matrix GLCM).....	48
III.4.2.Résultats est discussions	50
III.5.Conclusion	57
Conclusion général.....	58
Bibliographies.....	60

Liste des figures

Figure I.1. Textures : le motif brique correspond à une structure régulière alors que l’herbe est plutôt aléatoire..... 4

Figure I.2. Exemple de familles de textures : [a-d] textures déterministes, [e-h] textures observables, [i-l] textures stochastiques 4

Figure I.3. Exemple de calcul de matrices de cooccurrence à partir d’une image de dimension 4×5 ayant 4 niveaux de gris. 7

Figure I.4. En haut Exemple de calcul de matrices de cooccurrence à partir d’une image de dimension 4×5 ayant 4 niveaux de gris. En bas : Distance de calcul des matrices de cooccurrence vs orientation. 7

Figure I.5. Exemple de matrices de cooccurrences de niveaux de gris sur des textures de Brodatz..... 9

Figure I.6. Les images originales en lignes (1) et leurs spectres de Fourier en ligne (2). 12

Figure I.7. Principe de décomposition d’une image en somme de sinusoides planes élémentaires de différentes orientations. 12

Figure I.8. Le spectre du babouin et le zoom sur les fréquences intéressantes 13

Figure I.9. L’image de texture (mure de briques) (a) et son spectre de Fourier (b) avec le zoom sur les fréquences intéressantes. 14

Figure I.10. La fonction gaussien..... 15

Figure I.11. Exemple d’un filtre de Gabor 2D de taille 256×256 dans le domaine spatial. 17

Figure I.12. Exemple des orientations différentes de la gaussienne et de la sinusoïde. (a): $\sigma_x = 30, \sigma_y = 10, \theta = 90^\circ$. (b): $\sigma_x = 10, \sigma_y = 30, \theta = 0^\circ$ 18

Figure I.13. Le filtre de Gabor dans le domaine spatial (a) et son spectre de Fourier (b)..... 18

Figure I.14. Convolution d’une image par un banc de filtre de Gabor 19

Figure I.15. Réponse des filtres à une échelle 20

Figure I.16. Filtre de Gabor à différentes échelles..... 20

Figure II.1. Principe général de classification 22

Figure II.2. La base de données. 24

Figure II.3. Classification des outils de classification. 26

Figure II.4. La classification supervisée..... 28

Figure II.5. Principe de la technique SVM..... 31

Figure II.6. Structure d’un neurone artificiel..... 33

Figure II.7. principe de k-plus proche voisin.	34
Figure II.8. Classification par k-ppv.	35
Figure.II.9. Estimation par validation croisée (K-fold cross-validation).	37
Figure II.10. FNF et FPF de la courbe de ROC.....	39
Figure II.11. Le seuil de la courbe de ROC.....	39
Figure.III.1. Exemples d'image des textures de Brodatz	42
Figure.III.2. 4 échantillons de chaque classe de texture de la base UIUCTex.....	44
Figure III.3. L'image D31 après le filtrage.....	45
Figure III.4. Amplitude de filtre de Gabor	45
Figure III.5. Partie réelle de filtre de Gabor.....	46
Figure III.6. Amplitude de l'image.....	46
Figure III.7. Partie réelle de l'image.....	47
Figure III.8. La méthode GLCM dans la base de données UIUCTex	48
Figure.III.9. Voisinage d'un pixel selon 4 directions principales et 3 distances pour l'extraction des matrices de co-occurrences.....	50
Figure III.10. Visualisation des caractéristiques en 2D	56

Liste des tableaux

Tableau.III.1. 4 Validation croisée (Brodatz)..... 51

Tableau.III.2. Validation croisée (UIUCTex) 51

Tableau.III.3. 4 validation croisé par les méthodes GLCM et Gabor sur Brodatz et
UIUTEX..... 52

Tableau.III.4. Valiation simple (Brodatz) 52

Tableau.III.5. Valiation simple (fusion Gabor + GLCM) (*Brodatz*) 53

Tableau.III.6. Valiation simple (UIUCTex)..... 53

Tableau.III.7. Valiation croisée (Brodatz)..... 53

Tableau.III.8. Valiation croisée (UIUCTex) 54

Résumé

Le thème de recherche abordé concerne la classification automatique d'images à niveaux de gris texturées. L'objectif de ces travaux est de caractériser les textures présentes dans des images à niveaux de gris en utilisant les attributs d'Haralick extraits de matrices de cooccurrences GLCM et les filtres de Gabor. La texture à niveaux de gris a été caractérisée par les quatre indices d'Haralick « Contraste, Corrélacion, Energie et Homogénéité » et les deux caractéristiques de filtre de Gabor moyenne d'amplitude et l'énergie des coefficients de Gabor. La méthode de classification KNN a été appliquée sur les bases de données d'images à niveaux de gris de texture Brodatz et UIUCTex. On a obtenu des taux de classification élevé pour le niveau de gris 8 et 16 en utilisant les attributs d'Haralick extraits des matrices de cooccurrences « GLCM ».

Introduction



Générale

Introduction générale

L'idée de construire des machines capables de simuler des êtres humains afin de les aider dans certaines tâches, voire de les remplacer, était antérieure aux ordinateurs. Leur apparition a permis d'étendre le spectre des tâches à simuler en ajoutant celles dont l'exécution relève de facultés mentales comme la perception et le raisonnement.

Le problème que cherche à résoudre la reconnaissance des formes est d'associer une étiquette à une donnée qui peut se présenter sous forme d'une image ou d'un signal. La Reconnaissance de Forme (RF) est un thème particulièrement vaste et pluridisciplinaire qui fait l'objet de nombreux travaux de recherches. La notion de modélisation d'un comportement intelligent est souvent sous-entendue dans un problème de RF et donne une idée de la difficulté de la tâche.

Des méthodes générales ont été développées en reconnaissance des formes pour extraire automatiquement des informations des données sensibles afin de caractériser les classes de formes (apprentissage) et d'assigner automatiquement des données à ces classes (reconnaissance).

L'analyse d'images texturées a pour but d'extraire des caractéristiques ou attributs de texture afin de classer ou de segmenter les images. La classification des images est utilisée dans différents domaines tels que la reconnaissance des empreintes, des visages, indexation, reconnaissance des tumeurs ... etc. Pour extraire ces caractéristiques il y a plusieurs méthodes, parmi ces méthodes on a les filtres de Gabor qui sont basés sur le produit d'une gaussienne par des sinus ou des cosinus pour différentes orientations et fréquences, et la matrice de cooccurrence GLCM (grey level co-occurrence matrix), son principe général est de calculer le nombre de transitions entre les niveaux de gris et extraire des indices appelés : attribut de Haralick [1].

Dans ce travail on a utilisé les deux méthodes **GLCM** et **filtre de Gabor** pour extraire les informations des images texturées. Et pour faire la classification on a utilisé l'algorithme de k-plus proche voisin KNN (K-Nearest Neighbor). ensuite on a fait une fusion entre les deux méthodes afin d'améliorer la classification.

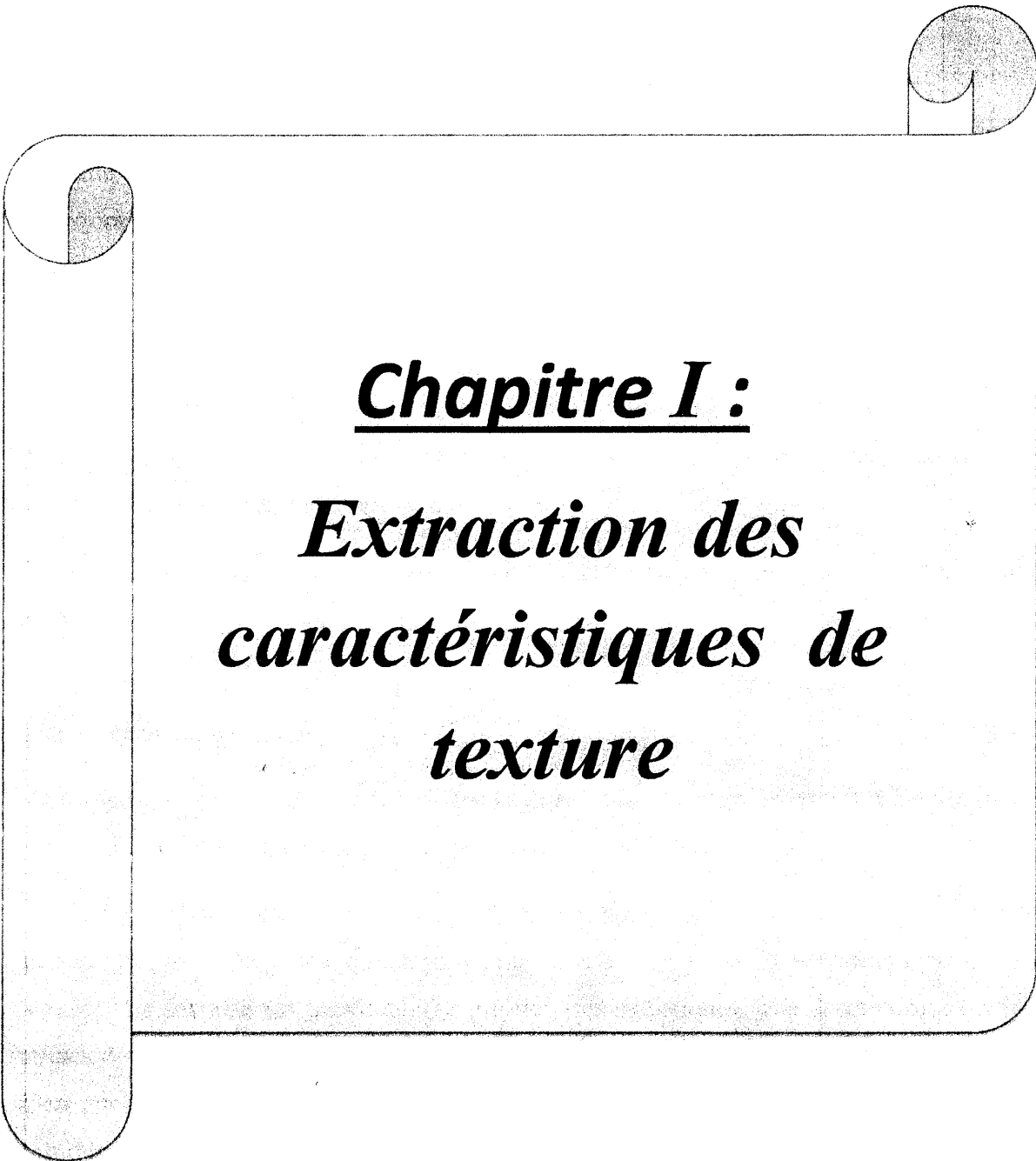
Notre travail comporte les parties suivantes :

Dans le chapitre I, nous exposons des définitions de la texture dans l'image et une description des méthodes de l'extraction de caractéristiques avec plus de détails sur les filtres de Gabor et la méthode GLCM.

Dans le deuxième chapitre, nous exposons des généralités sur la classification des données et nous rappelons quelques méthodes de classification.

Dans le chapitre III, nous présenterons les résultats et l'évaluation des techniques d'extraction de texture utilisée pour classifier les images.

Nous terminerons ce mémoire par une conclusion générale et des perspectives.



Chapitre I :
Extraction des
caractéristiques de
texture

I.1. Introduction

L'étude des textures rencontrées dans les images est un thème qui suscite beaucoup d'intérêt dans la communauté du traitement du signal et des images. Il existe une grande diversité dans les approches utilisées pour décrire ou identifier une texture. Certaines méthodes s'appuient sur des modèles stochastiques ou de prédiction linéaire. D'autres approches sont basées sur des calculs statistiques ou des transformations.

I.2. Définition de la texture

Plusieurs définitions de la notion de texture ont été présentées au cours des vingt dernières années. Ces définitions sont rarement génériques et portent plutôt sur un des aspects ou une des applications de la notion de texture.

En pratique, on peut distinguer deux grandes classes de textures, qui correspondent à deux niveaux de perception (figure I.1) :

- Les textures régulières, dans lesquelles la périodicité du motif est évidente, telle que les grilles, les murs, les tissus, etc [2].
- Les textures aléatoires pour lesquelles la distribution des intensités n'est l'objet d'aucune régularité apparente telle que le sable, les nuages, l'herbe, etc.

La première famille est bien décrite par des approches fréquentielles ou des approches structurelles dans lesquelles on associera un motif et des règles de placement sur un pavage régulier. La seconde est plutôt décrite par des lois statistiques, une description spectrale en termes de densité de puissance, des propriétés de corrélation ou d'isotropie. Mais une texture n'est jamais strictement périodique ni totalement aléatoire et 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.

Dans la littérature, il est souvent question de plusieurs familles de textures avec notamment les textures déterministes, les textures stochastiques et les textures observables (figure I.2. Les textures déterministes se caractérisent par la répétition d'un motif géométrique aussi appelé "texton". Ce genre de textures peut donc facilement être synthétisé à partir d'un motif si on connaît son orientation et sa taille [2].

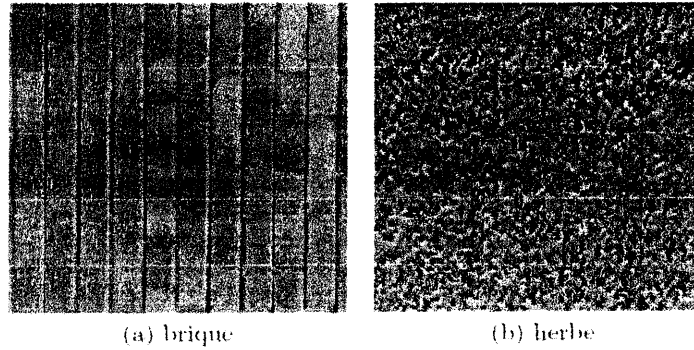


Figure I.1. Textures : le motif brique correspond à une structure régulière alors que l'herbe est plutôt aléatoire.

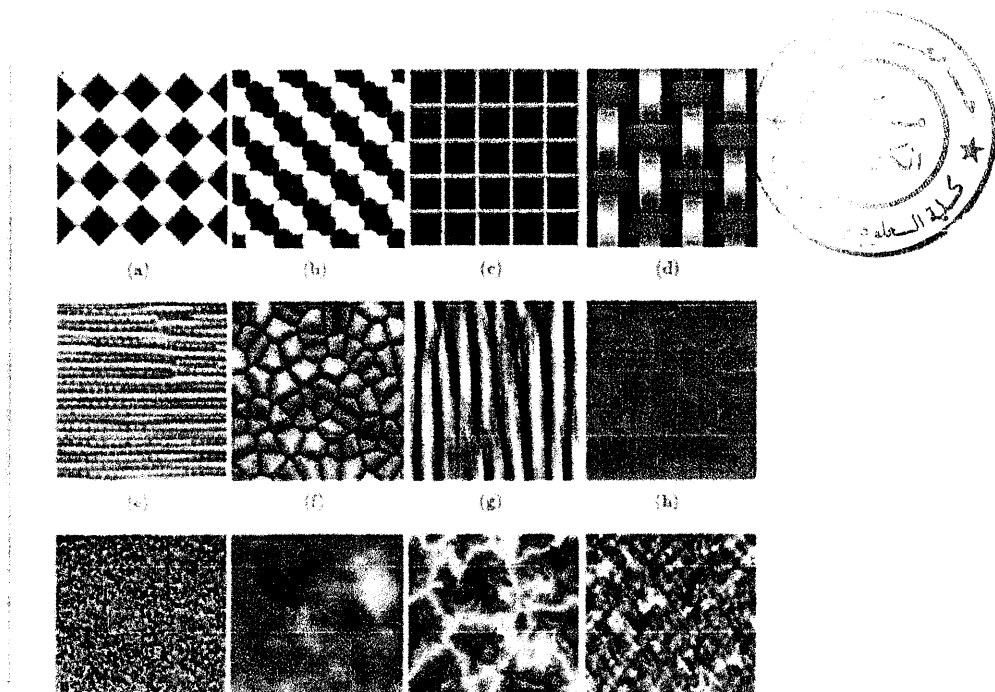


Figure I.2. Exemple de familles de textures : [a-d] textures déterministes, [e-h] textures observables, [i-l] textures stochastiques.

I.3. Techniques d'analyse de texture

Le but de l'analyse des textures est de décrire ces textures de façon formelle, en utilisant une formulation mathématique. Une multitude de méthodes, de variantes et de combinaisons de méthodes ont déjà été proposées dans la littérature et éprouvées en pratique. Dans la suite on va résumer les méthodes les plus connues [3].

I.3.1. Méthodes structurelles

Cette approche suppose que les surfaces étudiées sont composées d'éléments de bases arrangés selon une loi de placement. Les étapes d'analyse sont d'abord l'identification des éléments constitutifs, puis la définition des règles de placement. Il a été montré que cette méthode est applicable à l'étude de texture régulière donc elle ne s'adapte pas à toutes les textures du fait qu'un grand nombre de facteurs aléatoires sont à considérer.

I.3.2. Analyse spectrale

Cette analyse permet de mettre en évidence des correspondances entre les domaines spectral et spatial. Ainsi, le spectre d'une texture régulière de structure grossière (respectivement fine) est concentré dans les basses (respectivement hautes) fréquences. On obtient le spectre usuellement en utilisant la transformée de Fourier 2D.

I.3.3. Analyse spatio-fréquentielle

Cette technique offre la possibilité de mettre en évidence des propriétés locales sur une zone d'intérêt dans une image. Le principe d'extraction de données spatio-fréquentielles des textures consiste en la convolution de l'image avec un banc de filtres prédéfinis. A partir de ces résultats, un calcul de statistiques locales est effectué et le vecteur de paramètres texturels est obtenu par combinaison de ces statistiques. Les bancs usuels sont les filtres de Gabor 2D. Cette analyse est particulièrement efficace pour l'étude de textures périodiques [4].

I.3.4. Approche multi-niveaux

L'analyse d'image à différentes échelles est née de la nature même des images qui contiennent des objets de diverses tailles placés à diverses distances de l'observateur. Le principe de cette approche est d'étudier la texture à différents niveaux de résolution. La transformée en ondelettes et le modèle pyramidal sont les deux procédés les plus utilisés dans l'analyse multi-niveaux.

I.3.5. Géométrie fractale

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 [5].

I.3.6. Modélisation

Cette méthode suppose que l'image, présentée par l'ensemble de niveaux de gris, est une réalisation d'un champ de variables aléatoires. L'étude statistique des relations entre un pixel et son voisinage permet de définir un modèle de la texture. Les attributs sont alors les coefficients du modèle, dont le nombre est égal à la taille de voisinage considéré. Les deux modèles les plus utilisés sont : les modèles autorégressifs (AR) et les champs de Markov.

I.3.7. Matrices de co-occurrence

Les matrices de co-occurrence constituent une représentation simple des relations spatiales inter pixels. Elles permettent d'estimer les statistiques d'ordre 2 de l'image. Ces statistiques sont accumulées en un ensemble de matrices 2D, dont chacune mesure la dépendance spatiale de deux niveaux de gris, étant donné un vecteur de déplacement. On peut dériver de la matrice de co-occurrence des caractéristiques de la texture, comme l'énergie, l'entropie, le contraste, l'homogénéité et la corrélation [6].

Le but consiste à extraire d'une image présentant une texture unique, un nombre de paramètres, censés être représentatifs de la texture. Cette texture devra être aussi peu sensible que possible à des transformations de l'image qui laissent la texture inchangée pour un observateur humain (en particulier translation). L'étude de la texture a fait l'objet de nombreux travaux de recherche qui ont engendré une multitude de méthodes d'analyse.

Ces méthodes ont pour objet de caractériser, décrire, discriminer, segmenter les textures. Nous abordons ici différentes méthodes permettant l'extraction d'attributs texturaux dans les images à niveau de gris. L'accent est mis sur les matrices de co-occurrence, et le filtre de Gabor. Nous développons dans les sections suivantes les trois techniques [3].

I.4. Description des méthodes utilisées

I.4.1. Matrices de co-occurrences

Les matrices de co-occurrence constituent un outil d'analyse d'images en niveaux de gris. Elles rendent compte des transitions de niveaux dans l'image pour une direction et une distance données. Considérons une image rectangulaire de dimension (N_x, N_y) ayant N_g niveaux de gris. On définit la matrice de co-occurrence P , pour le vecteur $d = (d_x, d_y)$ et pour tout couple d'entier (i, j) appartenant à l'ensemble $[0, N_g[$, par :

$$P_d(i, j) = \#\{(x, y), (x', y') \mid x' = x + d_x, y' = y + d_y, \\ \text{et } (I(x, y) = i \ \& \ I(x', y') = j) \\ \text{ou } (I(x, y) = j \ \& \ I(x', y') = i)\}$$
(I.1)

Où $I(x, y) \in [0, Ng]$ [est le niveau de gris du pixel de coordonnées (x, y) dans l'image de départ et où la notation $\#$ représente le nombre d'éléments dans un ensemble. $P_d(i, j)$ contient alors le nombre de fois où les transitions $i \rightarrow j$ et $j \rightarrow i$ surviennent dans l'image entre deux pixels distants de (d_x, d_y) . La matrice P est donc symétrique et de dimension $Ng \times Ng$ [7].

La figure 1.3 montre un exemple de calcul des matrices de cooccurrence pour les vecteurs des plus proches voisins. Les matrices peuvent être normalisées par le nombre total de paires de voisins intervenant dans le calcul de la matrice (égal à la somme des termes de la matrice) [8].

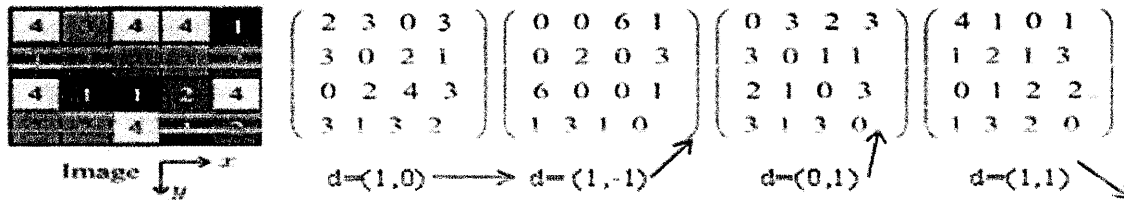


Figure 1.3. Exemple de calcul de matrices de co-occurrence à partir d'une image de dimension 4×5 ayant 4 niveaux de gris.

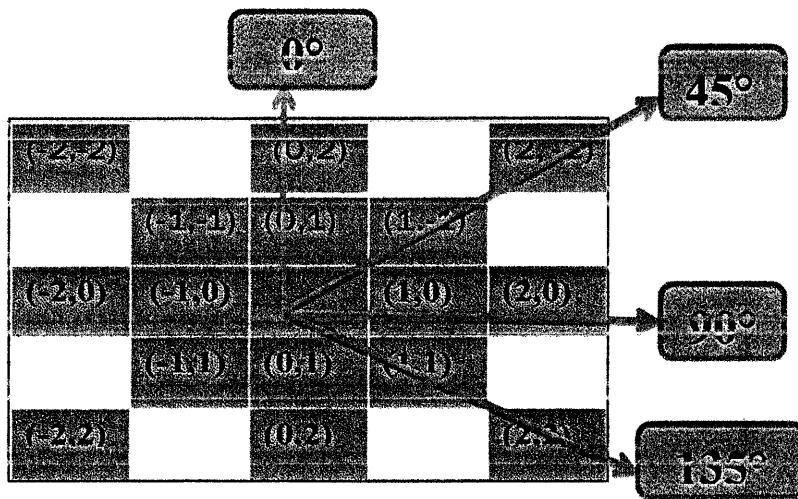


Figure 1.4. Distance de calcul des matrices de cooccurrence vs orientation.

Une définition non symétrique est parfois utilisée dans la littérature, en considérant les transitions $i \rightarrow j$ pour $P_d(i, j)$ d'une part, et $j \rightarrow i$ pour $P_d(j, i)$ d'autre part. De même, 4 matrices de co-occurrence ont initialement été définies dans Haralick et al [7]. pour des vecteurs correspondant aux plus proches voisins. Elles ont ensuite été généralisées à des

vecteurs quelconques, en utilisant une notation en coordonnées cylindriques, (r, θ) [9]. Cependant, le caractère discret des images (pixels) fait que la distance de calcul r dépend de l'angle θ (figure I.4) ; certaines directions impliquent de très grandes distances, généralement non adaptées à la détection de motifs texturés.

La plupart des images sont codées sur 256 niveaux de gris, par conséquent, la taille des matrices de co-occurrence est de 256×256 (figure I.5). On s'aperçoit ainsi que ces matrices comptabilisent une très grosse quantité d'informations difficile à exploiter directement. C'est pourquoi, un certain nombre d'auteurs comme Zucker ont essayé d'extraire de l'information de ces matrices afin de mettre en évidence la structure des textures. Mais c'est Haralick et al qui ont proposé les premiers 14 paramètres, caractérisant les textures, issus de ces matrices. Voici 4 paramètres considérés comme étant les plus utilisés et les plus pertinents [10] :

➤ **Le contraste**

$$CST = \sum_i \sum_j ((i - j)^2 p_{ij}(\delta, \theta)) \quad (I.2)$$

La valeur en est d'autant plus élevée que la texture présente un fort contraste. Ce paramètre est fortement non corrélié à l'énergie.

➤ **La corrélation**

$$COR = \sum_i \sum_j \frac{(i - \mu)(j - \mu)p_{ij}(\delta, \theta)}{\sigma^2} \quad (I.3)$$

COR mesure la dépendance linéaire (relativement à (δ, θ)) des niveaux de gris de l'image. La corrélation n'est corrélée ni à l'énergie, ni à l'entropie.

➤ **L'énergie**

$$ENE = \sum_i \sum_j (p_{ij}(\delta, \theta))^2 \quad (I.4)$$

Ce paramètre mesure l'uniformité de la texture. Il atteint de fortes valeurs lorsque la distribution des niveaux de gris est constante ou de forme périodique. Dans ce dernier cas, les valeurs élevées d'énergie sont obtenues pour les matrices $\bar{P}(\bar{\alpha}, \bar{\theta})$ lorsque $(\bar{\alpha}, \bar{\theta})$ correspond à la période.

➤ **Le moment inverse (ou homogénéité)**

$$IDM = \sum_i \sum_j \frac{p_{ij}(\delta, \theta)}{1 + (i - j)^2} \quad (I.5)$$

IDM (Inverse Différence Moment) mesure l'homogénéité de l'image. Ce paramètre est corrélé à une combinaison linéaire des variables *ENE* et *CST*.

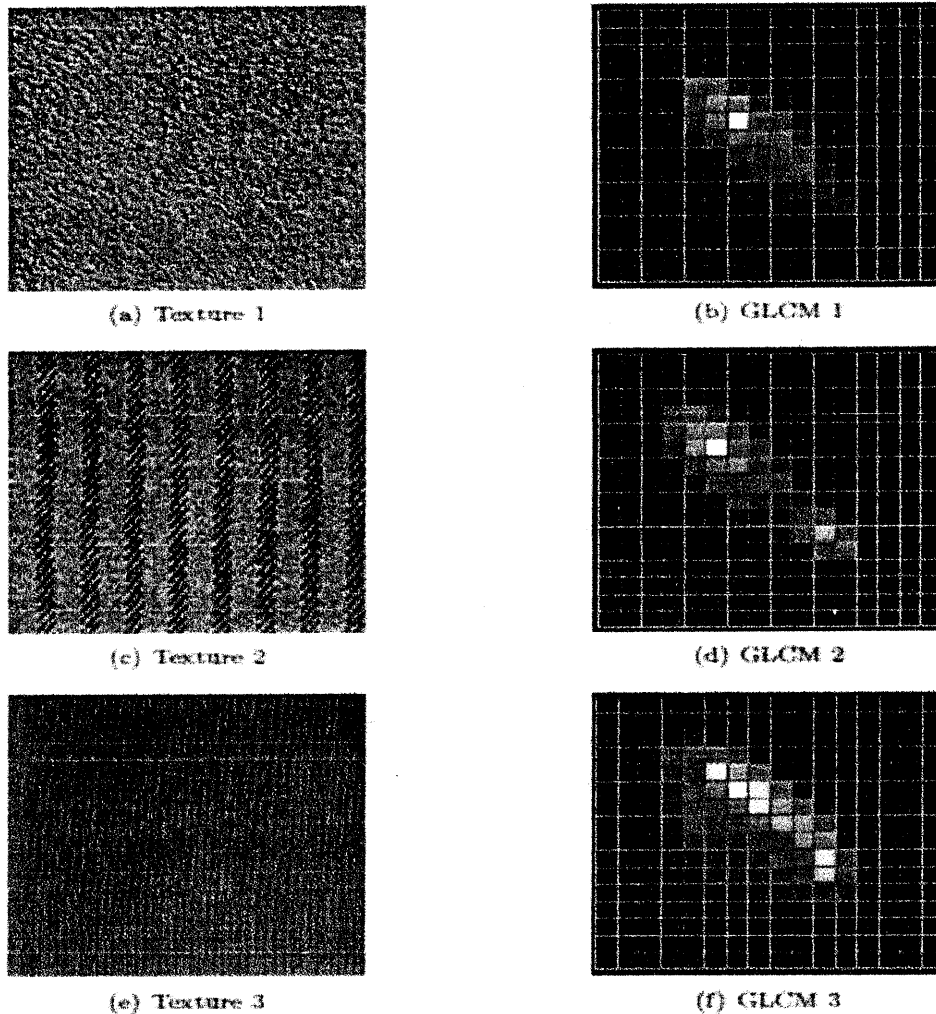


Figure I.5. Exemple de matrices de cooccurrences de niveaux de gris sur des textures de Brodatz

I.4.2. Méthodes fréquentielles : les filtres de Gabor

Les filtres de Gabor introduit par Gabor [11] ont montré des propriétés de localisation optimale dans les domaines spatial et fréquentiel. Ils sont donc adéquats pour des problèmes de classification de texture.

Dans le domaine spatial, un filtre de Gabor est une sinusoïde autour d'une fréquence (A, B) modulée par une gaussienne $g(x, y)$. C'est donc un filtre particulièrement adapté pour repérer une fréquence particulière en un lieu donné de l'image étudiée. On peut l'écrire [12] :

$$h(x, y) = g(x, y) e^{-j2\pi (Ax+By)} \quad (I.6)$$

Les méthodes fréquentielles, souvent utilisées en traitement du signal, permettent d'analyser une texture en identifiant les différentes fréquences qui la composent. Parmi elles, on retrouve notamment la transformée de Fourier, les filtres de Gabor ainsi que les ondelettes.

Ces méthodes, préservent à la fois les informations globales et locales. Elles sont bien adaptées aux signaux quasi périodiques, en effet, les textures sont des signaux quasi périodiques qui ont une énergie fréquentielle localisée. Ces méthodes permettent de caractériser la texture à différentes échelles. Analyser une texture de part son spectre fréquentiel nous donne beaucoup d'informations sur celle-ci.

La transformée de Fourier (TF) permet de passer d'une représentation de l'image dans le domaine spatial à sa représentation dans le domaine fréquentiel[13].

Transformée Cosinus Discret (DCT) La Transformée en cosinus discrète ou TCD (de l'anglais : DCT ou Discrète Cosine Transform) est une transformation proche de la transformée de Fourier discrète (DFT). Le noyau de projection est un cosinus et génère donc des coefficients réels, contrairement à la DFT, dont le noyau est une exponentielle complexe et qui génère donc des coefficients complexes. La DCT possède une excellente propriété de "regroupement" de l'énergie : l'information est essentiellement portée par les coefficients basses fréquences.

La transformée en Ondelettes, contrairement à Fourier, est beaucoup plus précise et riche en informations et en pertinence. Pour faire face aux difficultés d'analyse des textures naturelles, l'outil espace-fréquence et temps-échelle qui semble le plus approprié est la transformée en ondelettes. Celle-ci permet de prélever des paramètres locaux dans le domaine spatial, dans le domaine fréquentiel, à des résolutions multiples ainsi que des paramètres inter-échelles. De plus, son mode d'investigation, du global vers le détail, possède une similitude avec le système visuel humain.

I.4.3. Définition de la Transformée de Fourier (TF)

L'étude d'un signal nécessite souvent de combiner une analyse temporelle (ou spatiale) à une analyse fréquentielle. L'étude temporelle (ou spatiale) renseigne sur les discontinuités et la durée du signal. La deuxième permet de savoir quelles fréquences sont présentes dans ce signal. La transformée de Fourier fut le premier outil d'analyse fréquentielle utilisé en traitement d'images [14]. L'un des avantages de la transformée de Fourier est de

permettre de réaliser des filtrages par simple multiplication dans le domaine fréquentiel en évitant de réaliser le produit de convolution dans le domaine temporel ou spatial ce qui réduit le temps de calcul des filtrages surtout quand les masques de convolutions sont de grandes tailles.

I.4.3.1. Cas de signaux 2D

Les signaux 2D ou bidimensionnels sont des fonctions à deux dimensions, souvent notées x et y destinés pour décrire par exemple les images.

Les images numériques sont une version discrète de signaux 2D qui sont échantillonnés puis quantifiés. On peut appliquer une transformée de Fourier sur une image en niveaux de gris afin de connaître l'équivalent en fréquence de l'information contenue dans l'image spatiale. On choisit souvent d'afficher le résultat de la transformée de Fourier à l'aide d'une autre image en niveaux de gris (Figure I.6). On affiche en fonction des besoins le module, la phase, la partie réelle ou la partie imaginaire de la matrice complexe représentant le résultat de la transformation de Fourier de l'image originale.

- **La transformée de Fourier 2D discrète (TFD2D)** : La transformée 2D est donc souvent employée dans le domaine des images. Elle permet en effet de passer d'une représentation du domaine spatial (x,y) à une représentation dans le domaine fréquentiel (u,v) .

La TFD2D d'un signal $f[x, y]$ est définie sur \mathbb{Z}^2 , s'exprime sous la forme [15]:

$$F[u, v] = \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} f[x, y] e^{-2\pi j (ux, vy)} \quad (I.7)$$

En pratique, une image $f[m, n]$ est un ensemble de points (pixels) fini est borné, avec m et n entiers et $0 \leq m \leq M - 1$ et $0 \leq n \leq N - 1$. Dans ce cas l'expression de la TFD2D est donnée par :

$$g(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x}{\lambda} + \varphi\right) \quad (I.8)$$

Avec $0 \leq u \leq M - 1$ et $0 \leq v \leq N - 1$

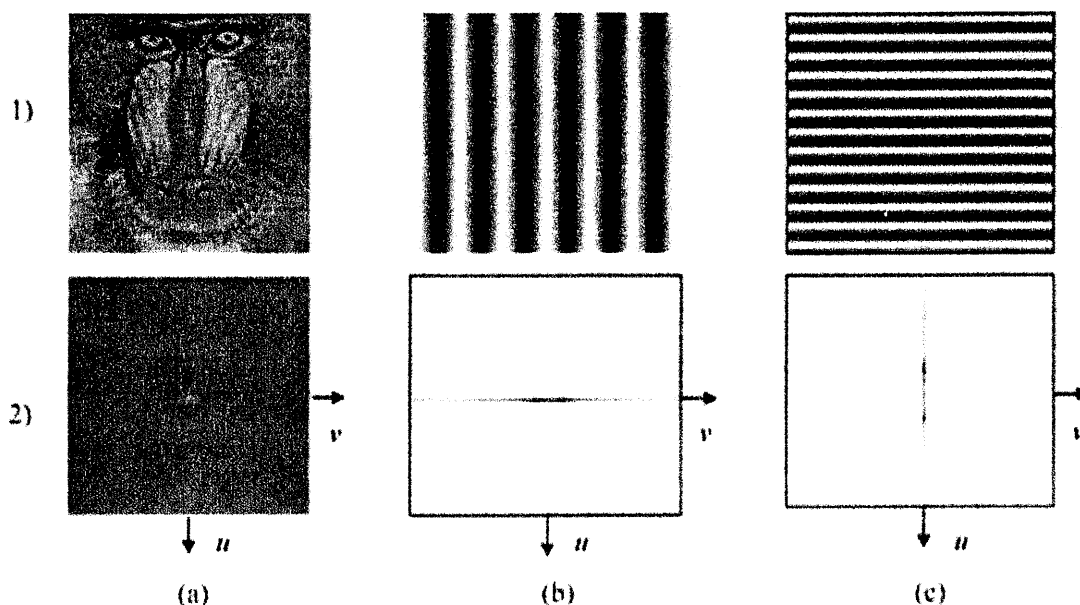


Figure I.6. Les images originales en lignes (1) et leurs spectres de Fourier en ligne (2).

Les deux variables u et v représentent les fréquences (spatiales) de l'image selon les directions verticales et horizontales respectivement. Elles s'expriment en cycles ou en radians par unité de longueur (m^{-1} ou $pixel^{-1}$), alors que les fréquences (temporelles) qui sont utilisées pour des signaux temporels s'expriment en cycles ou radians par unité de temps ($s^{-1} \equiv$ Hertz).

I.4.3.2. Interprétation du spectre de Fourier d'une image

La transformée de Fourier 2D d'une image représente le degré de ressemblance entre l'image originale et les fonctions cosinus et sinus à différentes fréquences et orientations. (Figure I.7)

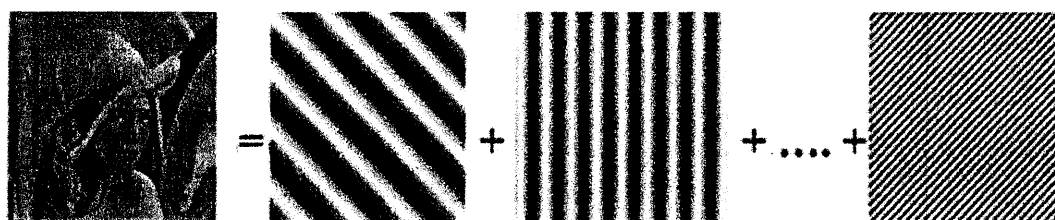


Figure I.7. Principe de décomposition d'une image en somme de sinusoides planes élémentaires de différentes orientations.

Dans la figure (Figure I.6) en ligne (1), l'image (b1) représente un sinus horizontale (0°) de fréquence f_1 , et (c1) un sinus verticale (90°) de fréquence f_2 . Chaque transformée de Fourier pour ces deux images est théoriquement constituée de trois points, qui sont [15]:

- le point central représente la moyenne de l'intensité de l'image d'origine. D'une façon générale, dans une image, l'information basse fréquence (autour du point central de la TF) code globalement les formes de l'image. Ces formes

seront floues si l'on ne conserve que les basses fréquences. C'est en rajoutant de l'information à plus haute fréquence que ces formes se distinguent par des contours de plus en plus nets.

➤ Les deux autres points en noir représentent la fréquence constituant les signaux. Il apparaît les deux pics en symétrie car l'image est réelle. Plus la fréquence d'oscillation des signaux est importante et plus les pics caractérisant cette fréquence dans le spectre sont éloignés du point central d'origine.

Cependant, il apparaît pour les deux images le phénomène de fenêtrage (effet de bords). Les effets de bords se manifestent dans les spectres de nos deux images par de nombreux autres pics d'amplitudes réduites alignés sur la direction de la sinusoïde. Ces lignes correspondent au sinus cardinal.

Revenons au spectre (a1) de l'image (a2). On voit bien qu'il comporte deux lignes, horizontale et verticale passent par l'origine, ces deux lignes dues aux effets de bords des images naturelles. Cela dit, il est difficile d'interpréter plus d'informations contenues dans le spectre d'énergie d'images complètes tant les détails fourmillent [15]. Le spectre de cette image

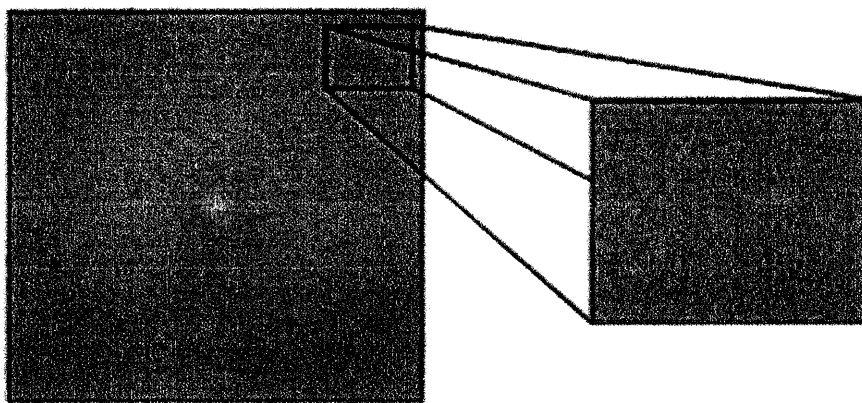


Figure I.8. Le spectre du babouin et le zoom sur les fréquences intéressantes

Présente cependant des pics d'intensité de manière symétrique dans les coins supérieurs. Ces pics d'amplitude se situent donc dans les hautes fréquences et pourraient donc correspondre par exemple au pelage du babouin. En effet, ce pelage est caractérisé par de l'information de texture qui est donc présente dans les hautes fréquences.

. En effet, c'est à travers ce spectre qu'on comprend comment les informations de texture apparaissent au sein du spectre. Une texture peut être par définition un motif qui se reproduit avec une certaine fréquence (Figure I.8), cette fréquence sera donc logiquement visible dans le spectre (en blanc) correspondant à l'image ou à la région texturées.

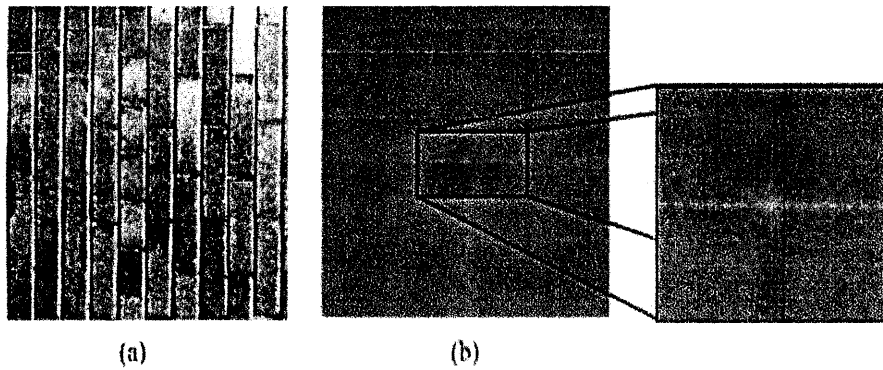


Figure I.9. L'image de texture (mure de briques) (a) et son spectre de Fourier (b) avec le zoom sur les fréquences intéressantes.

La figure (I.9) illustre aussi comment les informations de texture apparaissent au sein du spectre. On retrouve dans le spectre de l'image (a) qui est zoomé, des lignes horizontales et verticales qui correspondent à la répétition de ruptures verticales et horizontales rencontrées. Il faut faire attention cependant car certains pics d'amplitudes fréquentielles peuvent aussi être dus au sinus cardinal caractéristique des ruptures franches. Finalement, on comprend qu'il n'est pas facile d'interpréter l'information du spectre d'une image naturelle même texturée [15].

I.4.4. Filtrés de Gabor

Les filtres de Gabor sont introduits par Dennis Gabor en 1946. Ces filtres ont été largement utilisés à la fois comme fonctions de décomposition en ondelettes et comme outils d'analyse texturale.

Un filtre de Gabor agit comme un filtre gaussien passe-bande de largeur $\frac{1}{2\pi\sigma}$, qui peut être utilisé pour sélectionner un domaine de fréquences autour d'une fréquence particulière f_0 dans le spectre de Fourier. Le paramètre σ contrôle la sélectivité du filtre. En d'autres termes, une grande valeur de σ va décroître la précision de la localisation des bords

dans le domaine spatial alors qu'une valeur trop petite va décroître la précision en fréquence [17].

I.4.4.1.Cas 1D

La réponse impulsionnelle d'un filtre de Gabor est le produit d'une gaussienne et d'une exponentielle complexe. En 1D, l'expression d'un filtre de Gabor dans le domaine spatial ou temporel est donnée par :

$$g(x) = \left[\frac{1}{\sigma_x \sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma_x^2}\right) \right] \exp(2\pi j f_{x_0} x) \tag{I.9}$$

Où σ_x est l'écart-type de la gaussienne et f_{x_0} la fréquence centrale du filtre.

L'expression précédente est décomposable en deux termes, réel (à phase cosinus) et imaginaire (à phase sinus).

Dans le domaine de Fourier, le produit simple entre la gaussienne et l'exponentielle complexe devient un produit de convolution qui nous amène à l'expression finale de la transformée de Fourier de $g(x)$, donnée comme suit [21]:

$$G(f_x) = \exp(-2\pi^2 \sigma_x^2 (f_x - f_{x_0})^2) \tag{I.10}$$

Graphiquement, la transformée de Fourier d'un filtre de Gabor 1D est constituée d'une gaussienne centrée sur f_{x_0} (figure I.10).

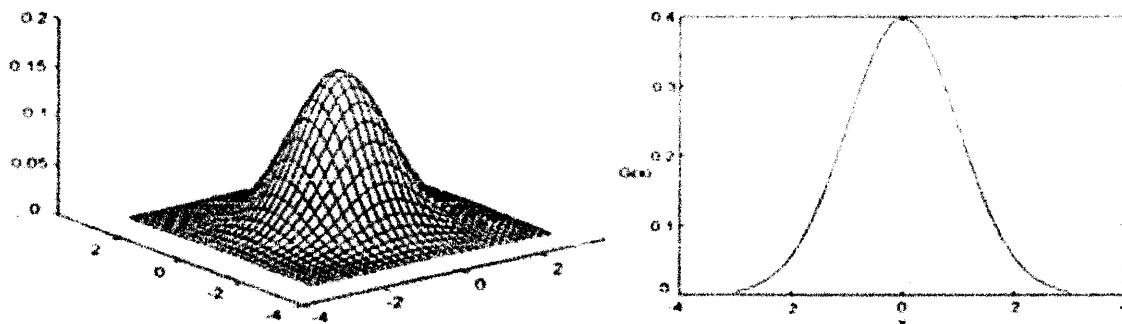


Figure I.10. La fonction gaussienne

Selon le principe d'incertitude d'Heisenberg ($\Delta t \Delta f > \frac{1}{4\pi}$), aucun signal ne peut être parfaitement localisé en temps et en fréquence de manière simultanée. Le filtre de Gabor minimise néanmoins cette incertitude en transformant l'inégalité en égalité [16]. Ainsi, on dit que ce filtre propose un compromis temps-fréquence optimale.

On contrôle la sélectivité de ce filtre passe-bande à travers. En effet, plus est élevé et plus le filtre est sélectif. Ce filtre est ainsi couramment utilisé en traitement de signal. Son application au traitement d'images nécessite sa transposition en 2D.

I.4.4.2. Cas 2D

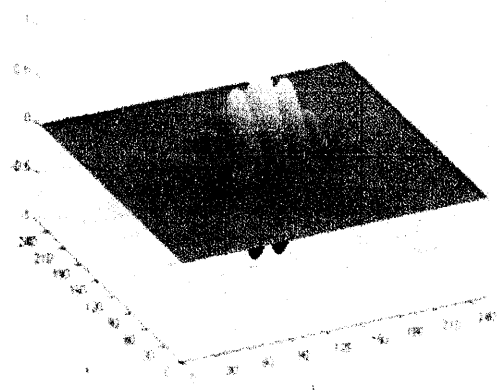
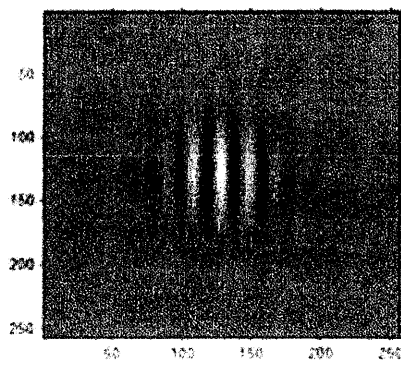
En 2D, le filtre de Gabor est défini par une gaussienne bidimensionnelle modulée par une fonction sinusoïdale plane (Figure I.11). L'expression du filtre de Gabor 2D est donnée par :

$$g(x, y) = \left[\frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right) \right] \exp(2\pi j(f_{x0}x + f_{y0}y)) \quad (I.11)$$

Où σ_x et σ_y sont les écart-types respectivement le long des axes x et y , f_{x0} et f_{y0} les fréquences centrales respectivement suivant x et y .

On peut modifier l'écriture de l'expression (I.11) en passant du repère cartésien au repère polaire où $f_0 = \sqrt{f_{x0}^2 + f_{y0}^2}$ est la fréquence centrale du filtre et $\theta = \arctan\left(\frac{f_{y0}}{f_{x0}}\right)$ l'angle entre l'axe des abscisses et le module de f_0 . Ainsi, on peut réécrire l'expression (I.11) sous cette forme :

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}\right) \exp(2\pi j f_0(x \cos\theta + y \sin\theta)) \quad (I.12)$$



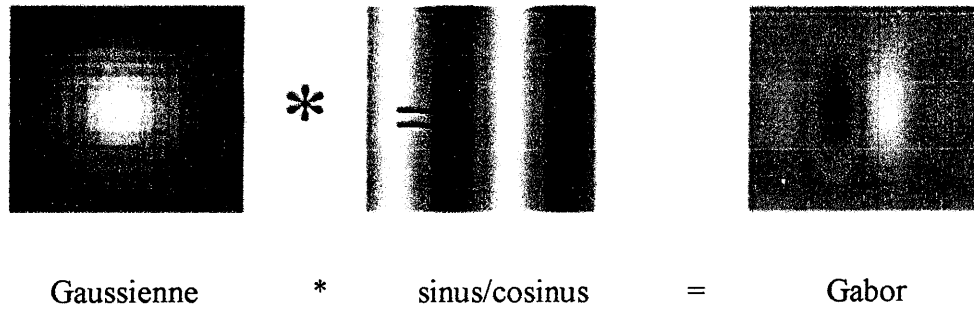


Figure I.11. Exemple d'un filtre de Gabor 2D de taille 256*256 dans le domaine spatial.

$$G_s(x, y) = \cos(w_x x + w_y y) e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (I.13) \quad G_a(x, y) = \sin(w_x x + w_y y) e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (I.14)$$

Jusqu'à maintenant, l'orientation de la gaussienne à l'aide σ_x et σ_y et l'orientation de la sinusoïde via θ sont indépendantes, c'est-à-dire elles peuvent prendre des orientations différentes (Figure I.12), ce qui constitue un problème.

Pour corriger le problème d'orientation (l'orientation de la gaussienne et de la sinusoïde), on doit opérer via l'angle θ . En effet, cela nous semble logique, ainsi on obtient une nouvelle expression du Filtre de Gabor 2D [16], donnée par l'expression suivante:

$$g(x, y) = \exp\left(-\frac{x'^2}{2\sigma_x^2} - \frac{y'^2}{2\sigma_y^2}\right) \exp(2\pi j f_0 x') \quad (I.15)$$

Où l'expression (I.16) qui exprime soit la partie réelle ou imaginaire de l'expression (I.15) tout dépend de la phase φ .

$$g(x, y) = \exp\left(-\frac{x'^2}{2\sigma_x^2} - \frac{y'^2}{2\sigma_y^2}\right) \cos(2\pi j f_0 x' + \varphi) \quad (I.16)$$

$$\begin{cases} x' = x \cdot \cos\theta + y \cdot \sin\theta \\ y' = -x \cdot \sin\theta + y \cdot \cos\theta \end{cases}$$

Si $\varphi = 0$ on aura la transformée de Fourier (Figure I.13) de l'expression (I.16) donnée par (I.7), (I.11):

$$G(f_x, f_y) = A \left\{ \exp\left(-\frac{(f_x - f_0)^2}{2\sigma_{fx0}^2} - \frac{f_y^2}{2\sigma_{fy}^2}\right) + \exp\left(-\frac{(f_x - f_0)^2}{2\sigma_{fx0}^2} - \frac{f_y^2}{2\sigma_{fy}^2}\right) \right\} \quad (I.17)$$

• Ou $\sigma_{fx} = \frac{1}{2\pi\sigma_x}, \sigma_{fy} = \frac{1}{2\pi\sigma_y},$ et $A = 2\pi\sigma_x\sigma_y$

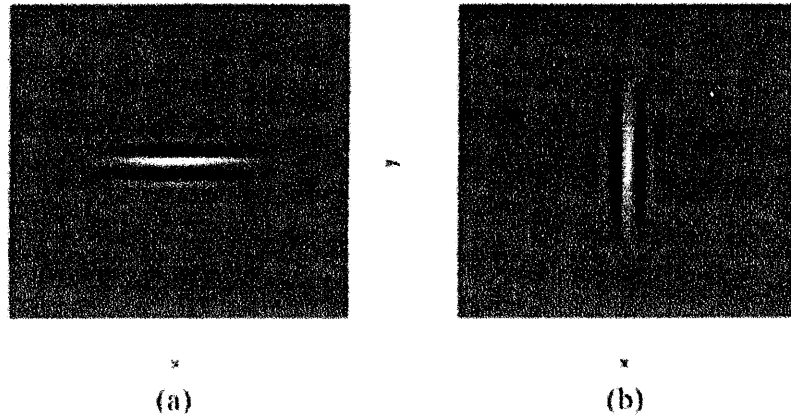


Figure I.12. Exemple des orientations différentes de la gaussienne et de la sinusöide.

(a): $\sigma_x = 30, \sigma_y = 10, \theta = 90^\circ.$ (b): $\sigma_x = 10, \sigma_y = 30, \theta = 0^\circ.$

A partir des expressions précédentes, une autre expression plus simple et plus pratique a été élaborée en introduisant de nouveaux paramètres afin d'intégrer l'aspect d'analyse multi-résolution (AMR). Ces paramètres sont utilisés dans l'expression finale suivante :

$$g(x,y) = \exp\left(\frac{-x'^2 + y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \varphi\right) \tag{I.18}$$

$$\begin{cases} x' = x \cdot \cos\theta + y \cdot \sin\theta \\ y' = -x \cdot \sin\theta + y \cdot \cos\theta \end{cases}$$

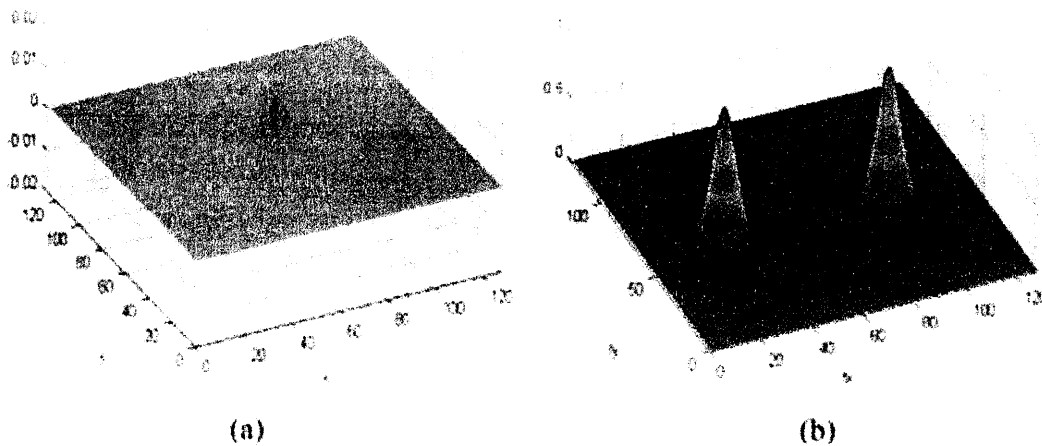


Figure I.13. Le filtre de Gabor dans le domaine spatial (a) et son spectre de Fourier (b).

Ainsi l'association d'un ensemble de filtres de Gabor permet de couvrir plus largement l'espace fréquentiel et de détecter un plus grand nombre d'orientations, donc d'extraire tous les contours de l'image. Le résultat d'un banc de filtres étant la moyenne des résultats des filtres de ce banc, nous pouvons tirer partie de la distributivité du produit de convolution afin d'alléger le traitement les figures I.14, I.15, I.16 donnent des exemples de la convolution des filtres de Gabor par une image.

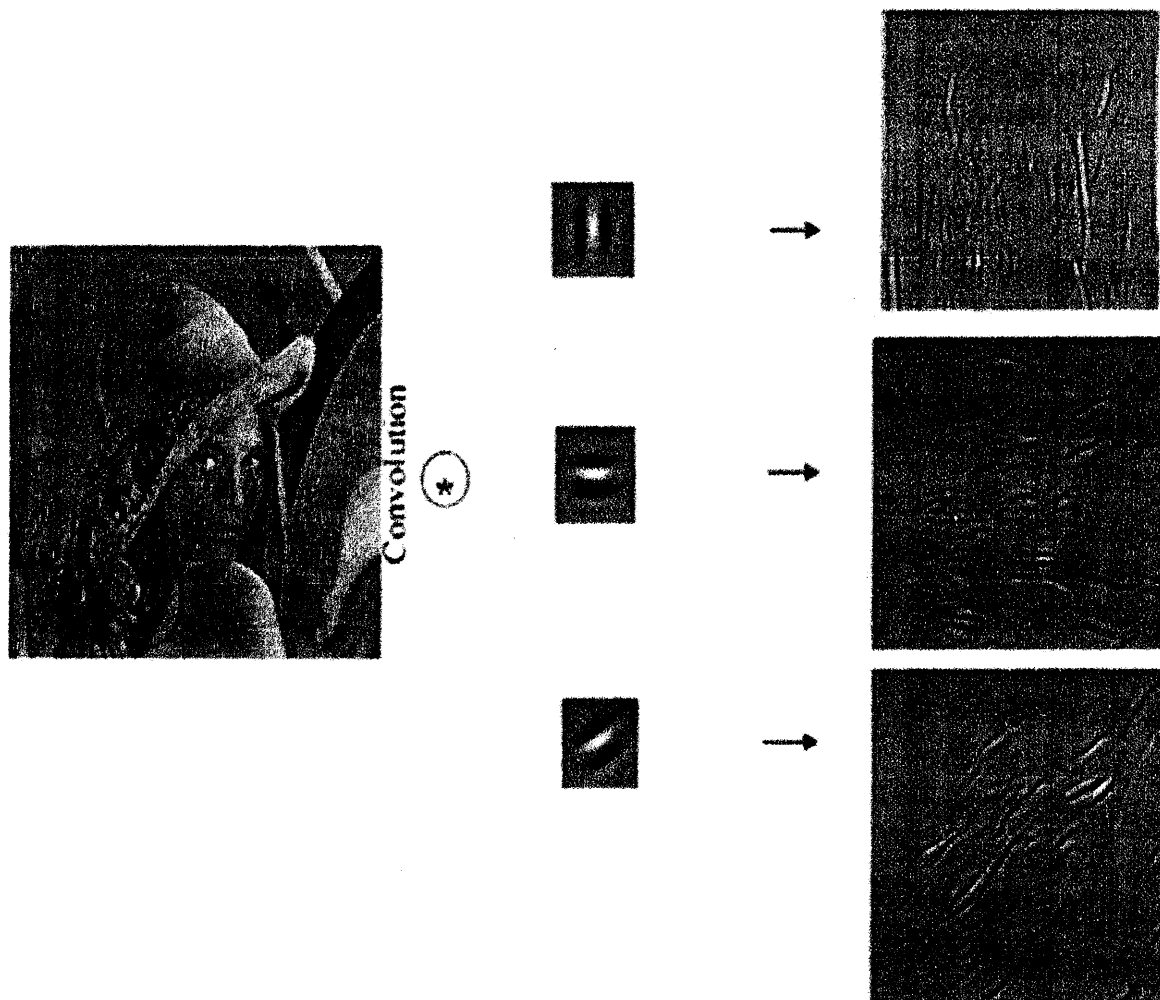


Figure I.14. Convolution d'une image par un banc de filtre de Gabor

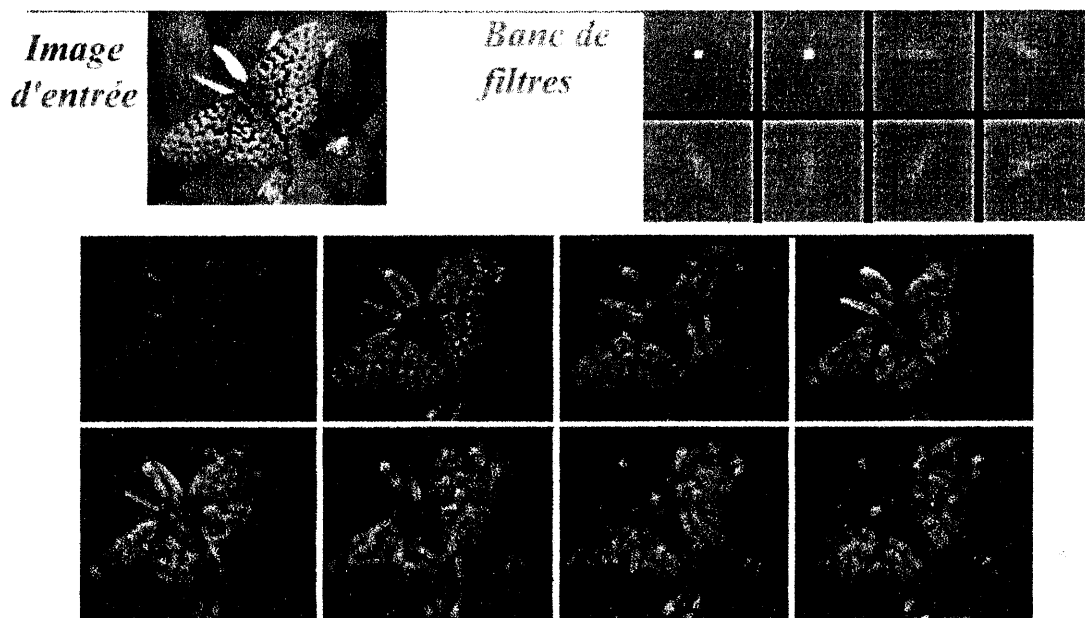


Figure I.15. Réponse des filtres à une échelle

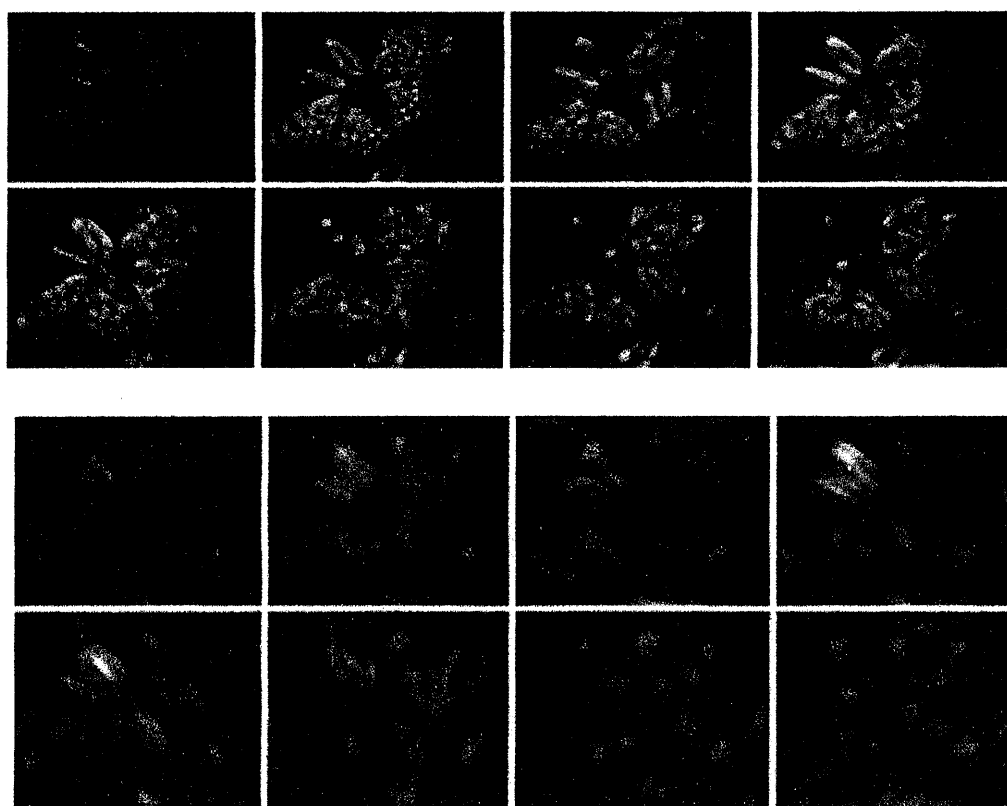
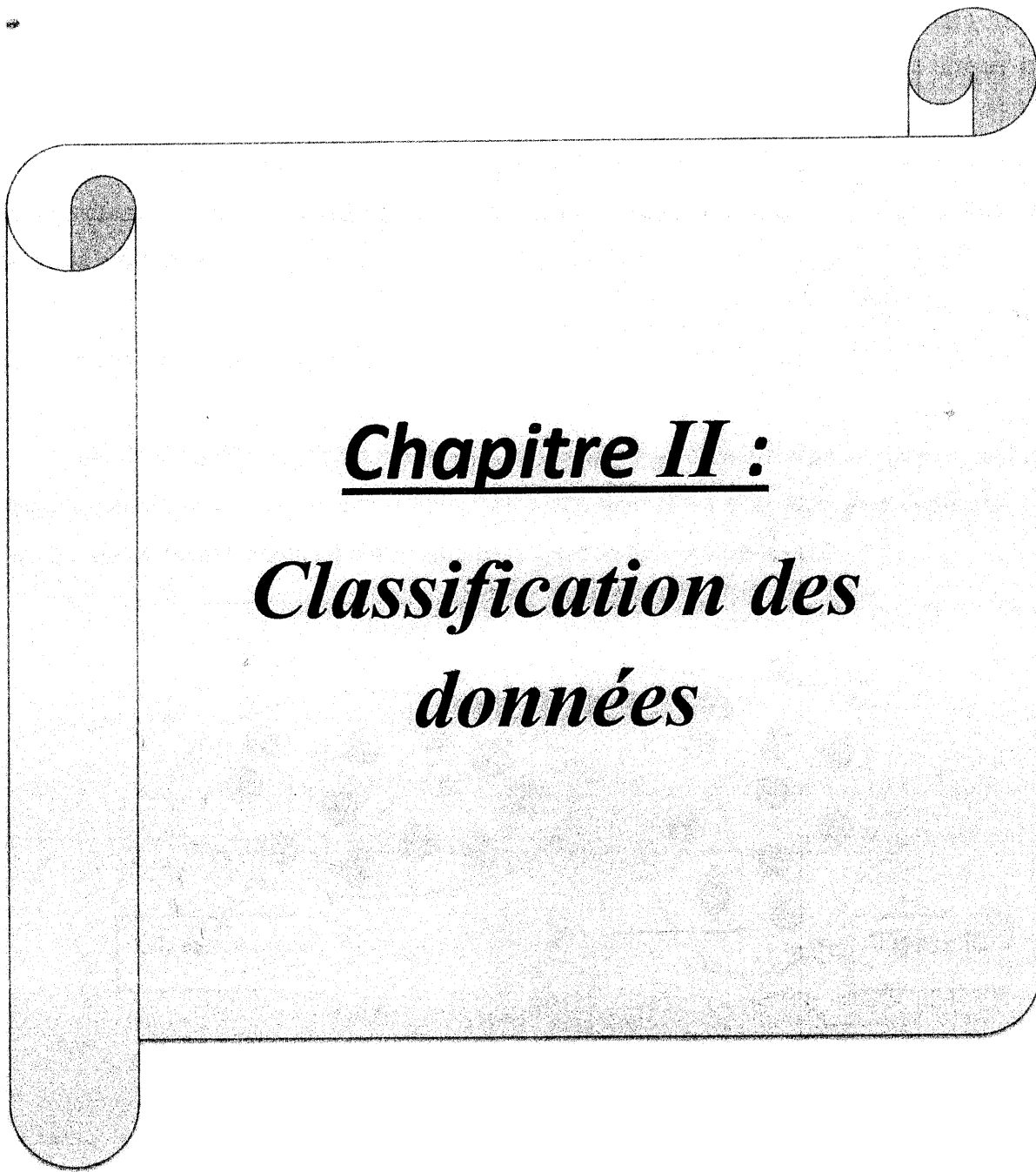


Figure I.16. Filtre de Gabor à différentes échelles

I.5. Conclusion

L'analyse de texture fait état d'un grand nombre d'approches et de méthodes existantes dans la littérature. Il est témoin d'un besoin croissant de connaissances plus approfondies sur la notion même de texture et sur le mécanisme humain de perception des textures. Ce chapitre donne une description de la notion de texture, et les différentes méthodes utilisées pour l'analyse de texture. Deux méthodes sont détaillées, l'une est la méthode GLCM qui est basé sur la dépendance spatiale entre les pixels, l'autre c'est une méthode fréquentielle ce sont filtres de Gabor.

A decorative scroll graphic with a vertical strip on the left and a horizontal strip on the top and right, all with rounded ends and a light gray fill. The text is centered on the horizontal strip.

Chapitre II :
***Classification des
données***

II.1. Introduction

Les problèmes de reconnaissance de formes tels que la classification des images sont très complexes en général. Le but de la classification est de regrouper les échantillons (image) en classes homogènes selon un critère bien défini. Plusieurs classifieurs peuvent donner des réponses différentes sur l'attribution d'un pixel de l'image à la classe qui lui correspond le plus. Ceci est dû d'une part à la qualité du contenu du pixel et d'autre part à l'erreur spécifique au classifieur. Cette erreur dépend du modèle de décision du classifieur et de la base de données utilisée pour son apprentissage.

II.2. Définition de classifieur

Afin de classifier les différentes images de textures, un classifieur opère généralement dans l'espace d'attributs de départ. Cependant, un nombre élevé d'attributs peut diminuer le taux de bonne classification de ces images et augmenter le temps de calcul.

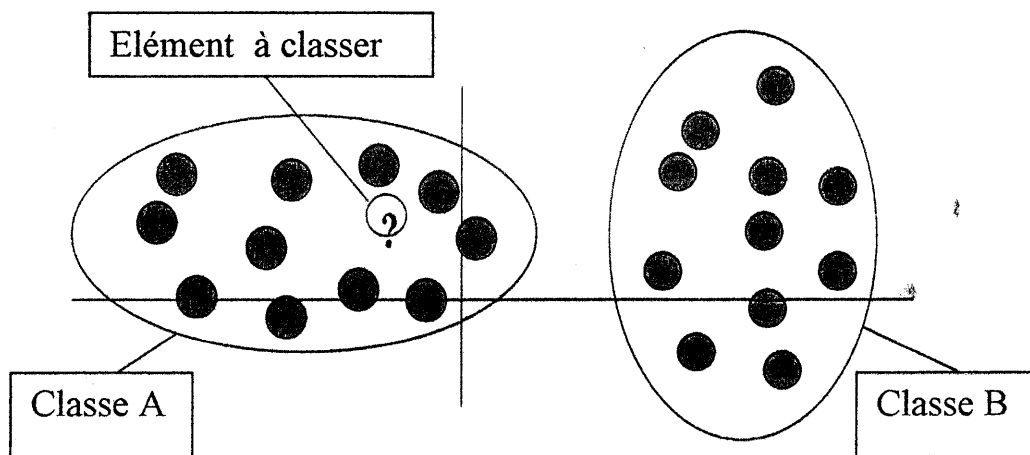


Figure II.1. Principe général de classification.

II.3. Construction d'un classifieur

La construction d'un classifieur pour un système considéré, nécessite des connaissances à propos de celui-ci. Ces connaissances peuvent avoir des formes (images) différentes dont les plus importantes sont:

- La structure probabiliste du système.
- Des connaissances acquises par des experts humains.
- Des modèles fonctionnant correctement sur des systèmes semblables.
- Un ensemble de formes générées par le système et leurs états correspondants.

Pour construire un classifieur, on peut utiliser un ou plusieurs de ces types de connaissance. Puis on utilise les connaissances disponibles, plus le classifieur ne sera adapté au système traité.

Dans de nombreuses applications, les trois premiers types de connaissance ne sont pas disponibles ou le sont partiellement. Cependant, le quatrième type de connaissance, autrement dit une base d'exemples, est souvent disponible.

La construction d'un classifieur est donc souvent faite à partir d'une base d'exemples et les autres types de connaissances disponibles servent généralement à choisir la méthode et le modèle ou à perfectionner le classifieur.

Le processus de construction du classifieur est donc un algorithme qui s'adapte à un système particulier en utilisant des exemples générés par celui-ci. Ce processus est nommé *apprentissage*.

Une fois le classifieur conçu, nous avons besoin de connaître ses performances pour savoir s'il est suffisamment performant pour l'application visée. La connaissance de ses performances nous permet également sa comparaison avec d'autres méthodes.

Une approche envisageable est d'estimer les performances du classifieur en les calculant à partir du modèle théorique utilisé et en remplaçant les paramètres manquants par des estimations. Cette approche donne dans la plupart des cas des estimations trop optimistes et s'avère souvent très difficile voire impossible à réaliser. Une autre approche est d'estimer les performances du classifieur en le testant sur des exemples réels générés par le système. Les vraies classes pour ces exemples doivent être connues pour pouvoir vérifier si le classifieur fonctionne correctement ou pas.

II.3.1. Base d'apprentissage et base de test

Comme on vient de voir, l'estimation des performances nécessite des exemples. Il est évident que ces exemples ne doivent pas être les mêmes que ceux sur lesquels l'apprentissage est effectué, sinon les performances estimées vont être sûrement trop optimistes puisque le classifieur a déjà appris ces exemples au cours de sa construction. Nous avons donc besoin de certains exemples pour l'apprentissage et d'autres pour le test.

L'ensemble des exemples utilisés pour l'apprentissage est nommé base d'*apprentissage* et noté par \mathbb{X}^A . L'ensemble des exemples utilisés pour tester le classifieur est nommé base de test et noté par \mathbb{X}^T .

Si le nombre d'exemples disponibles est limité, on utilise souvent une partie de la base disponible comme base d'apprentissage et le reste comme base de test. On considère que la vraie classe w_{vraie} est connue pour chaque exemple de la base d'apprentissage et de la base de test.

L'ensemble des exemples de la base d'apprentissage qui appartiennent à la classe w_i est noté par :

$$\mathbb{X}_{w_i}^A = \{x \in \mathbb{X}^A | w_{vraie} = w_i\} \quad (II.1)$$

L'ensemble des exemples de la base de test qui appartiennent à la classe w_i est noté par $\mathbb{X}_{w_i}^T$.

$$\mathbb{X}_{w_i}^T = \{x \in \mathbb{X}^T | w_{vraie} = w_i\} \quad (II.2)$$

base de donnée disponible

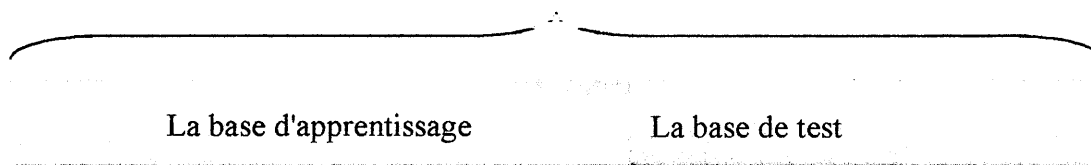


Figure II.2. La base de données.

II.3.2. Types d'informations fournies par les classifieurs

Les réponses fournies par les classifieurs peuvent être divisées en quatre types:

Type Classe : La réponse du classifieur est une classe. Il considère que la forme d'entrée x appartient à cette classe.

Type Ensemble : La réponse du classifieur est un ensemble de classes. Il considère que la forme x appartient à une des classes de cet ensemble, mais il ne donne aucune indication supplémentaire qui pourrait nous aider à discriminer parmi ces classes.

Type Rang : La réponse est un classement sur toutes les classes ou seulement sur une partie des classes. La classe mise au premier rang du classement est la classe qu'il considère comme la plus probable pour la x forme et la classe du dernier rang est la classe la moins probable. Le classifieur indique ce classement en donnant comme sortie un vecteur des rangs, de taille N_w .

Type Mesure : Le classifieur donne, pour toutes les classes, des mesures qui indiquent en quelque sorte, directement ou indirectement, à quel degré le classifieur considère chaque classe comme la vraie classe de x . La sortie du classifieur est donc un vecteur de mesures de taille N_w [17].

II.4. Les outils de classification

Les algorithmes de classification peuvent être répertoriés de plusieurs manières (Voir figure II.3). Un premier critère est le niveau de supervision, c'est-à-dire l'importance de l'intervention de l'utilisateur dans le processus de classification.

Il est également possible de séparer les algorithmes paramétriques associés à un modèle probabiliste des algorithmes non paramétriques[18].

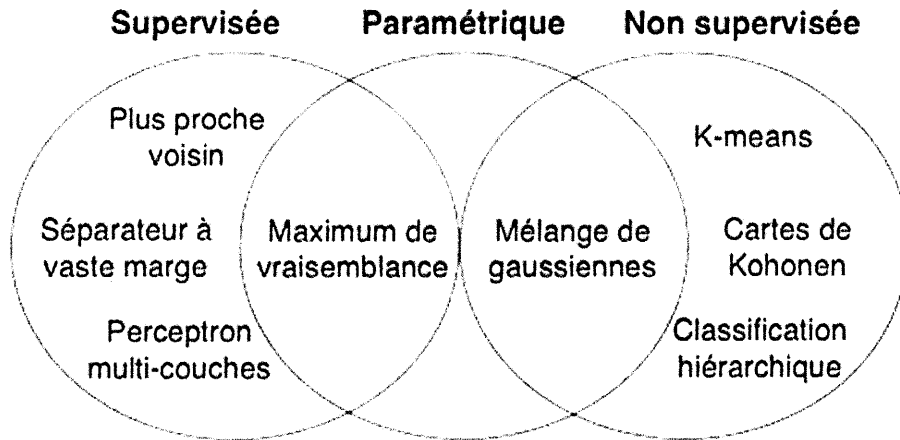


Figure II.3. Classification des outils de classification [18].

II.4.1. Classification non supervisée

Dans le cas d’une classification non supervisée, l’image est partitionnée sans utiliser de connaissance a priori (nombre de classes, ensembles d’échantillons déjà étiquetés).

De tels algorithmes de classification analysent la répartition des données au sein de leur espace de représentation de sorte à déterminer des groupements naturels correspondant à des zones de forte densité. Ces outils sont très prisés, principalement parce que l’acquisition de vérités terrain précises (un processus long et délicat) n’est pas systématiquement possible.

Il existe principalement deux grandes familles d’approches : les classifications par partitionnement et les classifications hiérarchiques.

La première considère que l’espace de représentation est divisible en un nombre fini de régions disjointes où toutes les observations appartiennent à la même classe. L’algorithme K-means [19] est le représentant le plus répandu de cette famille. Son objectif est de partitionner un ensemble de n observations (x_1, x_2, \dots, x_n) en k groupes (G_1, G_2, \dots, G_k) tout en minimisant la somme des distances à la moyenne :

$$\arg \min \sum_{i=1}^k \sum_{x_j \in G_i} \|x_j - \mu_i\|^2, \mu_i = \frac{1}{|G_i|} \sum_{x_j \in G_i} x_j \quad (II.3)$$

Les algorithmes de classification ascendante ou descendante hiérarchique génèrent, respectivement par fusion ou par division, une suite de partitions des observations allant de

la partition mono classe, où la même classe englobe la totalité des observations, à la partition où chaque observation correspond à une classe.

L'objectif de ces outils est de proposer à l'analyste de choisir manuellement ou automatiquement la partition de classe la plus réaliste. Il en existe plusieurs variantes qui se distinguent essentiellement par leur critère de fusion ou de division des classes (distance euclidienne, angle spectral, etc.).

II.4.2. Classification supervisée

L'approche supervisée consiste, à partir de plusieurs échantillons de l'image cible (que l'on appelle également données d'apprentissage) représentatifs de chacune des classes, à élaborer un modèle (probabiliste, géométrique ou structurel) à partir duquel le reste des éléments de l'image cible seront étiquetés. La pertinence du résultat de classification est en grande partie déterminée par la représentativité des données d'apprentissage, qu'il n'est pas forcément évident de rassembler de manière précise et qui dépendent le plus souvent de l'information thématique recherchée.

Parmi les méthodes non paramétriques le procédé le plus simple est celui des k plus proches voisins [20]. Ce dernier consiste à assigner au pixel étudié la classe majoritaire parmi les k échantillons qui lui sont le plus proche dans le nuage de points considéré. Ce procédé est simple mais une base d'apprentissage trop grande a tendance à augmenter très fortement les temps de calcul.

Plus récent, l'algorithme des Séparateurs à Vastes Marges [21] (SVM) utilise des hyperplans pour séparer les classes au sein de l'espace de représentation des données. Sa complexité est également dépendante de la taille de la base d'apprentissage.

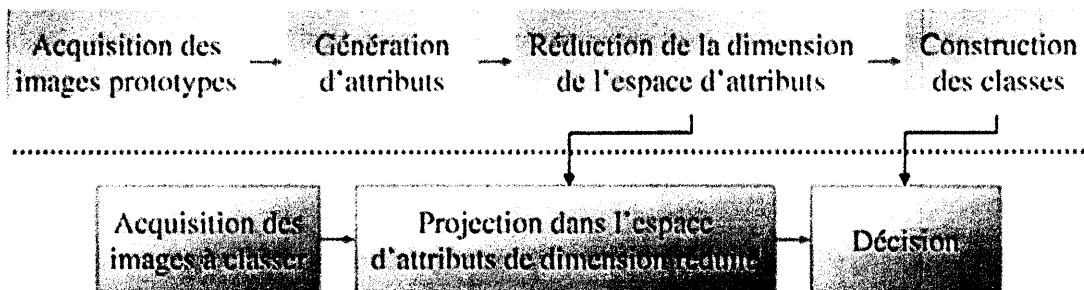
Les réseaux de neurones multi-couches (Multi-Layer Perceptron) [22] constituent une autre approche de classification supervisée non paramétrique. Ces réseaux sont constitués d'un minimum de trois couches de neurones. Afin de paramétrer les poids associés à ces derniers, l'algorithme d'apprentissage le plus répandu est cette fois-ci la rétro propagation du gradient.

Enfin, les outils de classification supervisée paramétriques déterminent les frontières de décision entre classes grâce à un modèle probabiliste défini par apprentissage. Partant de ce modèle, la phase de prédiction permet alors d'associer une classe au reste des pixels de l'image. L'exemple typique est la classification par maximum de vraisemblance

[23] qui associe une loi de probabilité conditionnelle présumée gaussienne à chacune des classes considérées, auxquelles vont correspondre une zone de probabilité maximale au sein de l'espace vecteur.

Le schéma présenté sur la figure II.4 illustre de manière générale la méthode de classification supervisée.

Apprentissage



Classification

Figure II.4. La classification supervisée.

Cette méthode est divisée en deux phases successives [24] :

- ❖ l'apprentissage, a pour but de construire des classes à partir de l'ensemble des prototypes (également appelé ensemble d'apprentissage)[25]. Pour cela, les textures présentes dans les images prototypes sont décrites par un ensemble d'attributs. Chaque texture est alors représentée par un vecteur d'attributs dans un espace d'attributs. Un classifieur est ensuite élaboré afin de déterminer la classe d'appartenance d'une image en fonction de ses caractéristiques.
- ❖ la classification, consiste à appliquer les règles de classification déterminées lors de l'apprentissage afin de classer l'image de la texture à contrôler.

II.5. Méthodes de classification

On distingue principalement deux approches de la reconnaissance des formes :

- les méthodes statistiques : l'extraction des caractéristiques produit des valeurs numériques qui sont confrontées aux modèles statistiques caractérisant chaque classe.
- les méthodes structurelles (ou syntaxiques) : l'extraction des primitives produit des valeurs symboliques et des relations qui font l'objet d'une analyse structurelle ou syntaxique. Les deux approches sont complémentaires et peuvent être combinées.

Les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs. Elles s'appliquent à un grand nombre d'activités humaines et conviennent en particulier au problème de la prise de décision automatisée. La procédure de classification sera extraite automatiquement à partir d'un ensemble d'exemples. Un exemple consiste en la description d'un cas avec la classification correspondante. Un système d'apprentissage doit alors, à partir de cet ensemble d'exemples, extraire une procédure de classification. Il s'agit en effet d'extraire une règle générale à partir des données observées. La procédure générée devra classer correctement les exemples de l'échantillon et avoir un bon pouvoir prédictif pour classer correctement de nouvelles descriptions.

Les méthodes utilisées pour la classification sont nombreuses, citons : la méthode des Machines à Vecteurs Supports (SVM), les Réseaux de Neurones, K Plus Proches Voisins (KPPV), etc. Nous présentons dans cette mémoire une étude détaillée de la technique KPPV. Cette méthode a montré son efficacité dans de nombreux domaines d'applications tels que le traitement d'image, la catégorisation de textes ou le diagnostics médicales, et ce même sur des ensembles de données de très grandes dimensions[26].

II.5.1. Méthode Bayésienne

L'approche du maximum de vraisemblance considère le vecteur de paramètres $\hat{\theta}_j$ comme une valeur fixe mais inconnue et essaie d'estimer la valeur qui est la plus cohérente avec des exemples disponibles.

L'approche Bayésienne en revanche considère le vecteur de paramètres θ_j comme une variable aléatoire. $p(\theta_j)$ la densité de probabilité a priori de θ_j , est supposée connue et contient toutes informations qu'on pourrait avoir à propos de θ_j avant l'observation des

exemples. L'observation de la base d'exemples nous permet d'améliorer notre connaissance à propos de θ_j et d'obtenir une nouvelle densité de probabilité $p(\theta_j | \mathbb{X}_{w_j}^A)$ à partir de la densité a priori $p(\theta_j)$ en utilisant la formule de Bayes :

$$p(\theta_j | \mathbb{X}_{w_j}^A) = \frac{p(\mathbb{X}_{w_j}^A | \theta_j) p(\theta_j)}{\int p(\mathbb{X}_{w_j}^A | \theta_j) p(\theta_j) d\theta_j} \quad (\text{II.4})$$

Comme dans l'approche du maximum de vraisemblance, on peut écrire :

$$p(\mathbb{X}_{w_j}^A | \theta_j) = \prod_{k=1}^{n_j^A} p(x_k | \theta_j) \quad (\text{II.5})$$

Et la connaissance de la forme paramétrique des $p(x_k | \theta_j)$ nous permet de calculer les densités $p(\theta_j | \mathbb{X}_{w_j}^A)$.

A la différence de l'approche du maximum de vraisemblance, l'approche Bayésienne n'essaye pas de trouver une seule valeur $\hat{\theta}_j$ qui optimise un critère quelconque et l'utilise pour calculer les densités $p(x | w_j)$. Elle calcule les densités $p(x | w_j)$ en intégrant sur toutes les valeurs de θ_j :

$$p(x | \mathbb{X}_{w_j}^A) = \int p(x, \theta_j | \mathbb{X}_{w_j}^A) d\theta_j \quad (\text{II.6})$$

II.5.2. Machines à vecteurs support (SVM)

La méthode "Machines à vecteurs supports" (SVM, de l'anglais Support Vector Machines (SVM)), proposée par Vapnik en 1995[27], a initialement été définie pour un problème de discrimination à deux classes. De manière simple, il s'agit de construire, dans l'espace des attributs, un hyper-plan séparant les points-image représentatifs des prototypes de chacune des deux classes. Cette technique, également appelée méthode des séparateurs à vastes marges, repose sur deux idées clés : la notion de marge maximale et la notion de noyau [28].

Chaque exemple doit être représenté par un vecteur de dimension n . La méthode cherche alors l'hyperplan qui sépare les exemples positifs des exemples négatifs, en garantissant que la marge entre le plus proche des positifs et des négatifs soit maximale. Intuitivement, cela garantit un bon niveau de généralisation car de nouveaux exemples pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être tout de même situés franchement d'un côté ou l'autre de la frontière.

L'inconvénient des SVM est le choix empirique de la fonction noyau adaptée au problème. Un deuxième inconvénient est le temps de calcul qui croît de façon cubique en fonction du nombre de données à traiter [29].

Le principe de la technique SVM est représenté dans la figure ci-dessous (figure II.5) :

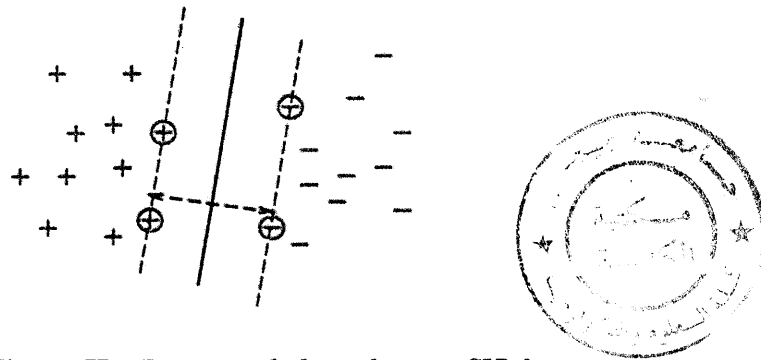


Figure II.5. Principe de la technique SVM.

II.5.3. Analyse discriminante descriptive

L'analyse discriminante descriptive est une méthode statistique supervisée qui, à partir d'un jeu de N observations décrites par M variables et réparties en K groupes, vise à déterminer l'espace de représentation des données qui séparera au mieux les K groupes. Cette méthode est essentiellement géométrique et ne repose sur aucune hypothèse probabiliste [18].

Soit n_i , $i \in \{1, \dots, K\}$ le nombre d'échantillons appartenant au groupe i . μ_i correspond au centroïde des échantillons du groupe i et W_i est la matrice de covariance de ce dernier.

L'objectif de l'algorithme est de déterminer une série d'axes non corrélés deux à deux qui discriminent au mieux les individus de groupes différents tout en rapprochant les individus de même groupe. Pour cela, nous considérons deux matrices de dispersion. La matrice de dispersion intraclasse :

$$\bar{W} = \frac{1}{N} \sum_{i=1}^K n_i \bar{W}_i \quad (II.7)$$

Correspond à la moyenne pondérée des matrices de covariance de chaque classe et représente la dispersion moyenne des individus d'une classe autour de leur centroïde. La matrice de dispersion inter-classe :

$$B = \frac{1}{N} \sum_{i=1}^K n_i (\mu_i - \mu)^T (\mu_i - \mu) \quad (II.8)$$

Où $\mu = \frac{1}{N} \sum_{i=1}^K n_i \mu_i$ est le centroïde de l'ensemble des échantillons, représente la dispersion moyenne entre centroïdes de classes différentes.

On recherche ainsi u , un vecteur directeur de l'espace tel que $u^T B u$ soit maximal et $u^T W u$ soit minimal. Cela consiste à maximiser $\frac{u^T B u}{u^T V u}$, où $V = B + W$ est la matrice de dispersion totale, ce qui revient à résoudre l'équation suivante :

$$V^{-1} B u = \lambda u \quad (II.9)$$

Au final, les axes recherchés sont ainsi les vecteurs propres correspondant aux valeurs propres non-nulles de la matrice $V^{-1} B$. Dans le cas où $N > M > K$, il existe $K - 1$ axes discriminants.

Cette méthode a l'avantage d'être rapide et de tenir compte des classes de l'image dans sa recherche d'axes discriminants. Cependant, elle nécessite de ce fait la connaissance d'un ensemble d'échantillons pour chaque classe et peut être prise en défaut si les classes ne sont pas assez uniformément réparties dans l'espace initial, notamment lorsque deux classes ont une moyenne très proche.

II.5.4. Réseaux de neurones

Les réseaux de neurones, fortement inspirés par le système nerveux biologique, sont composés d'éléments simples, appelés neurones, répartis sous forme de couches. Ces éléments sont connectés entre eux par des liaisons affectées de poids (les synapses). L'apprentissage de ces poids est effectué de sorte qu'à chaque vecteur d'attributs sélectionné $\hat{Y}^{i,j}$ présenté à l'entrée du réseau, corresponde en sortie la classe C^j à laquelle le prototype $w^{i,j}$ correspondant appartient.

La figure II.6 illustre la structure d'un neurone artificiel. Chaque neurone reçoit des valeurs provenant des neurones en amont via ses connexions synaptiques et traite ces valeurs via une fonction de combinaison. Le résultat de la combinaison est alors transformé par la fonction d'activation pour produire sa sortie. Cette fonction permet de seuiller le résultat pour obtenir une sortie binaire.

Le réseau de neurones le plus populaire est le perceptron multicouche (Multi-Layer Perceptron (*MLP*)). Sa structure est simple : il est constitué de plusieurs couches de neurones (une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées) entièrement connectées entre elles [30]. Pour ce type de réseau, la fonction de combinaison renvoie le produit scalaire entre le vecteur des entrées et le vecteur des poids $W = [W_1, \dots, W_d]^T$.

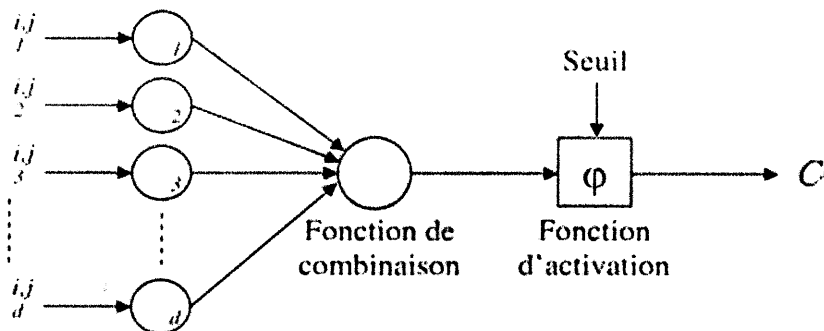
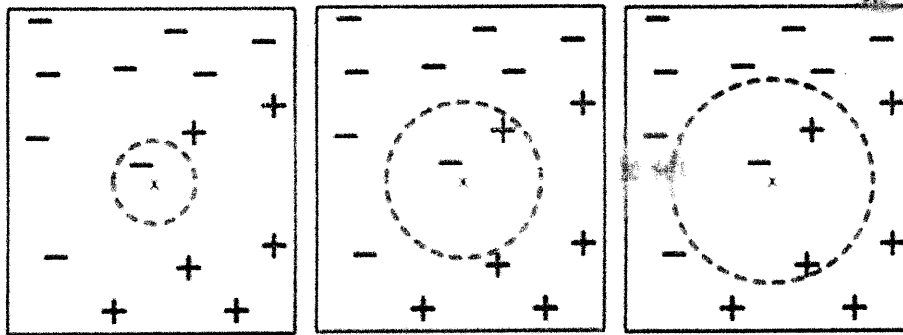


Figure II.6. Structure d'un neurone artificiel.

Il existe de nombreux autres types de réseaux de neurones disponibles à ce jour, comme par exemple les réseaux à fonction radiale (Radial Basic Functions (*RBF*)), qui ont pour architecture celle des *MLP* et pour fonction d'activation une gaussienne.

II.5.5. K-plus proches voisins

C'est une méthode non paramétrique basée sur l'existence des lois de probabilités régissant les observations et les classes. Toutefois si cette connaissance est incomplète, il est préférable d'orienter la procédure de décision vers l'estimation de la loi de probabilité ou de celle des probabilités a posteriori. Parmi les estimateurs usuels, on distingue l'estimateur de Parzen et l'estimateur des k-plus proches voisins (k - ppv).



(a).1-nearest neighbor (b).2- nearest neighbor (c).3- nearest neighbor

Figure II.7.principe de *k-plus proche voisin (k-ppv)*.

II.5.5.1. Règle des *k* – plus proches voisins (*k-ppv*)

Le principe de la règle des *k-ppv* est d'affecter X_u à la classe majoritairement représentée parmi ses *k*-plus proches une nouvelle observation voisins.

En réalité, La manière la plus simple d'élaborer cette règle, est de mesurer la distance entre la nouvelle observation et chacun des vecteurs de l'ensemble d'apprentissage. La nouvelle observation sera alors affectée à la classe majoritairement représentée parmi ses *k-ppv*. La définition de la notion de plus proche voisin est liée au choix d'une distance.

✓ **Les *k-PPV* nécessitent seulement :**

- Un entier *k*
- Une base d'apprentissage
- Une métrique pour la proximité

Exemple :

- Dans l'exemple suivant (figure II.8), on a 3 classes et le but est de trouver la valeur de la classe de l'exemple inconnu *x*
- On prend la distance Euclidienne et $K=5$ voisins
- Des 5 plus proches voisins, 4 appartiennent à ω_1 et 1 appartient à ω_3 , donc *x* est affecté à ω_1 , la classe majoritaire.

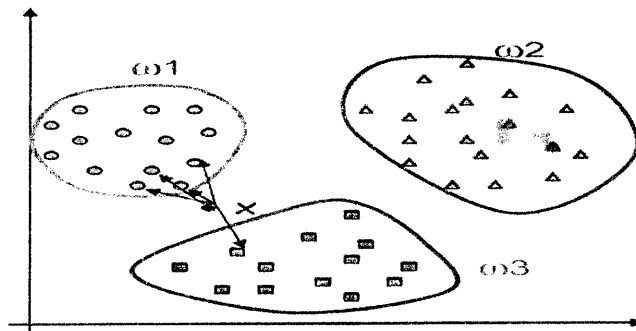


Figure II.6. Classification par k-ppv.

II.5.5.2. L'algorithme K-ppv

Une nouvelle observation classée dans la classe d'appartenance de l'observation de l'échantillon d'apprentissage qui lui est la plus proche, au regard des covariables utilisées. La détermination de leur similarité est basée sur des mesures de distance. Formellement, soit L l'ensemble de données à disposition ou échantillon d'apprentissage :

$$L = \{ (y, x), i = 1, \dots, n_L \} \tag{II.10}$$

Où $y_i \in \{1, \dots, c\}$ dénote la classe de l'individu i et le vecteur $x_i = (x_{i1}, \dots, x_{ip})$ représente les variables prédictives de l'individu i . La détermination du plus proche voisin est basée sur un fonction distance arbitraire $d(\cdot)$. La distance euclidienne ou dissimilarité entre deux individus caractérisés par p covariables est définie par:

$$d((x_1, x_2, \dots, x_p), (u_1, u_2, \dots, u_p)) = \sqrt{(x_1 - u_1)^2 + (x_2 - u_2)^2 + \dots + (x_p - u_p)^2} \tag{II.11}$$

Ainsi, pour une nouvelle observation (y, x) le plus proche voisin $(y(1), x(1))$ dans l'échantillon d'apprentissage est déterminé par :

$$d(x, x_{(1)}) = \min_i (d(x, x_i)) \tag{II.12}$$

➤ Algorithme des K-plus proches voisins:

Début

Pour chaque (exemple $(x', c) \in L$) faire

Calculer la distance $D(x, x')$

Fin pour

Pour chaque $\{x' \in kppv(x)\}$ faire

Compter le nombre d'occurrences de chaque classe

Fin pour

Attribuer x à la classe la plus fréquente

Fin

II.6. Evaluation de la classification

II.6.1. Validation croisée

Une des méthodes les plus utilisées en pratique est la validation croisée. Elle repose sur l'idée d'un découpage (aléatoire) des données en deux sous-échantillons. Supposons dans un premier temps ce découpage fixe. On utilise la première partie des données (appelée échantillon d'apprentissage, de taille N_e) pour construire une estimation de la « vraie » courbe avec chacun des modèles en compétition. Ensuite, avec les données restantes (formant l'échantillon de test, de taille $N_v = n - N_e$), on évalue la qualité de chacune des estimations en les comparant aux valeurs observées.

La raison fondamentale pour laquelle cette méthode fonctionne est l'indépendance entre échantillon d'apprentissage (entraînement) et échantillon de validation. Ainsi, on ne sous-estime pas l'erreur d'estimation des modèles les plus complexes. Il reste cependant un léger biais, car un estimateur fondé sur N_e données est moins performant que s'il pouvait en utiliser $n > N_e$. Pour obtenir une méthode de choix de modèles optimale, N_e doit être aussi proche de n que possible.

Une deuxième faiblesse de cette méthode est qu'elle repose fortement sur le choix d'un découpage, d'où une grande incertitude dans l'évaluation de la qualité de chaque modèle. En pratique, on fixe la valeur de N_e , puis l'on choisit aléatoirement B échantillons

d'apprentissages ($B > I$), et l'on évalue la qualité de chaque modèle en effectuant une moyenne sur ces B découpages (figure II.9). C'est pourquoi l'on parle de validation « croisée ». Pour des raisons de temps de calcul, on ne peut pas considérer tous les découpages possibles.

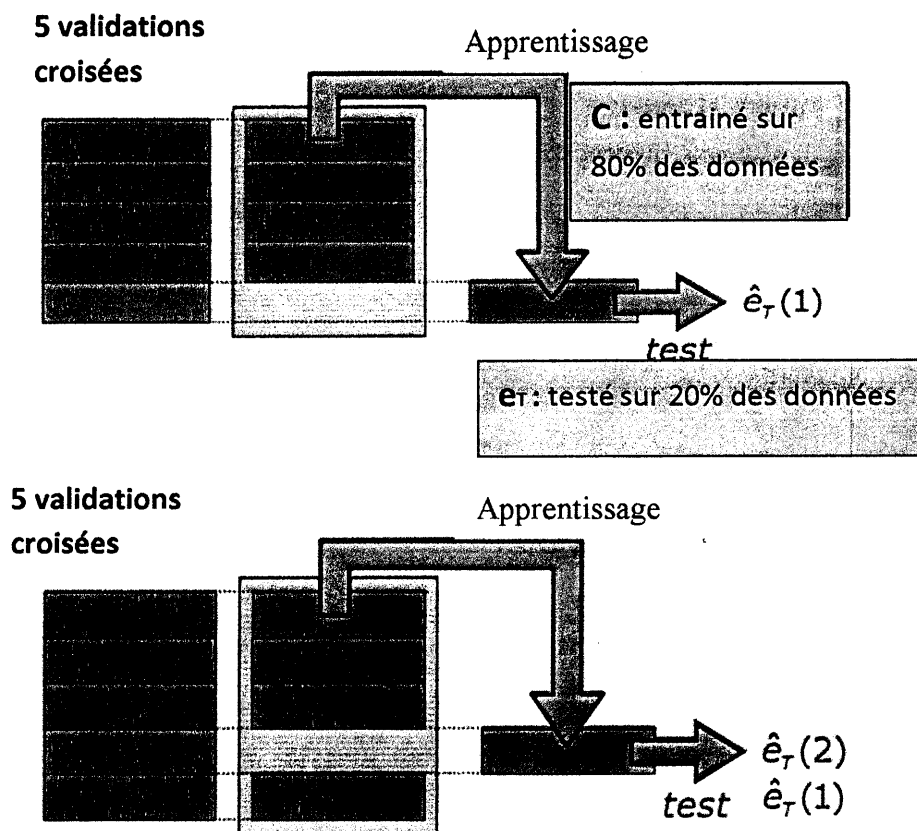


Figure II.9. Estimation par validation croisée (*K*-fold cross-validation).

Son mécanisme est donné par les étapes suivantes : [33]

- Partitionner l'ensemble d'apprentissage en k partitions égales.
- Appliquer l'apprentissage sur les $k-1$ partitions et utiliser la même partition pour tester.
- Répéter ce processus k fois selon la permutation circulaire.

Ainsi, l'erreur obtenue sera égale à la somme des erreurs trouvées dans les partitions.

II.6.2. La courbe de ROC

La fonction d'efficacité du récepteur ou caractéristique de fonctionnement du récepteur ou caractéristique de performance (d'un test) ou courbe sensibilité/spécificité ou, plus fréquemment, la courbe ROC (de l'anglais Receiver Operating Characteristic) est une mesure de la performance d'un classificateur binaire, c'est-à-dire d'un système qui a pour objectif de catégoriser des entités en deux groupes distincts sur la base d'une ou plusieurs de leurs caractéristiques. Graphiquement, on représente souvent la mesure ROC sous la forme d'une courbe qui donne le taux de vrais positifs (sensibilité : fraction des positifs qui sont détectés (correctement)) en fonction du taux de faux positifs (fraction des négatifs qui sont détectés (incorrectement)) pour ce même groupe (figure II.10).

Les courbes ROC furent inventées pendant la Seconde Guerre mondiale pour montrer la séparation entre les signaux radar et le bruit de fond.

Elles sont souvent utilisées en statistiques pour montrer les progrès réalisés grâce à un classificateur binaire lorsque le seuil de discrimination varie. Si le modèle calcule un score s qui est comparé au seuil S pour prédire la classe (c.-à-d. $(s < S) \rightarrow$ positif et $(s \geq S) \rightarrow$ négatif), et qu'on compare ensuite avec les classes réelles (Positif et Négatif), la sensibilité est donnée par la fraction des Positifs classés positifs (TPF), et l'anti-spécificité (1 moins la spécificité FPF) par la fraction des Négatifs classés positifs. On met l'anti-spécificité en abscisse et la sensibilité en ordonnée pour former le diagramme ROC. Chaque valeur de S fournira un point de la courbe ROC, qui ira de $(0, 0)$ à $(1, 1)$ (figure II.11).

À $(0, 0)$ le classificateur déclare toujours 'négatif' : il n'y a aucun faux positif, mais également aucun vrai positif. Les proportions de vrais et faux négatifs dépendent de la population sous-jacente.

À $(1, 1)$ le classificateur déclare toujours 'positif' : il n'y a aucun vrai négatif, mais également aucun faux négatif. Les proportions de vrais et faux positifs dépendent de la population sous-jacente.

Un classificateur aléatoire tracera une droite allant de $(0, 0)$ à $(1, 1)$.

À $(0, 1)$ le classificateur n'a aucun faux positif ni aucun faux négatif, et est par conséquent parfaitement exact, ne se trompant jamais.

À (1, 0) le classificateur n'a aucun vrai négatif ni aucun vrai positif, et est par conséquent parfaitement inexact, se trompant toujours. Il suffit d'inverser sa prédiction pour en faire un classificateur parfaitement exact.

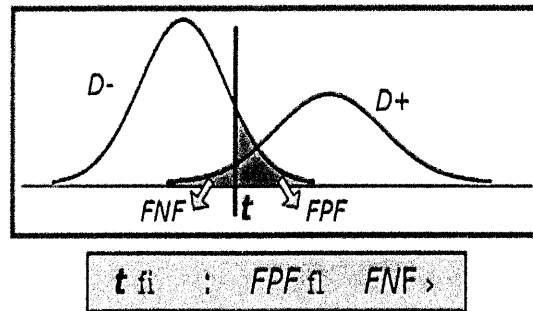


Figure II.10 : FNF et FPF de la courbe de ROC.

Classification:

- Variable implicite : Par exemple, règle de décision ou priors.
- Variable changeante : Différent valeurs pour la sensibilité et la sélectivité.

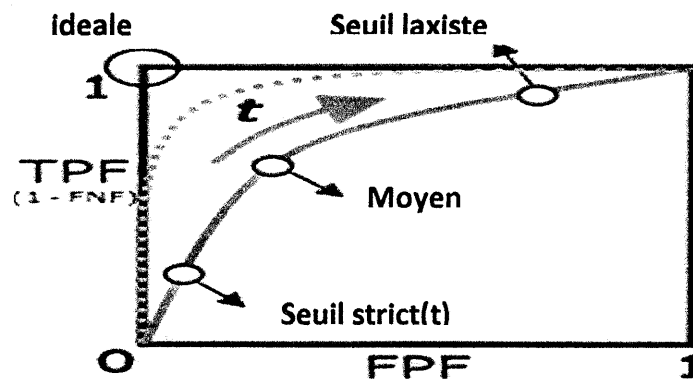
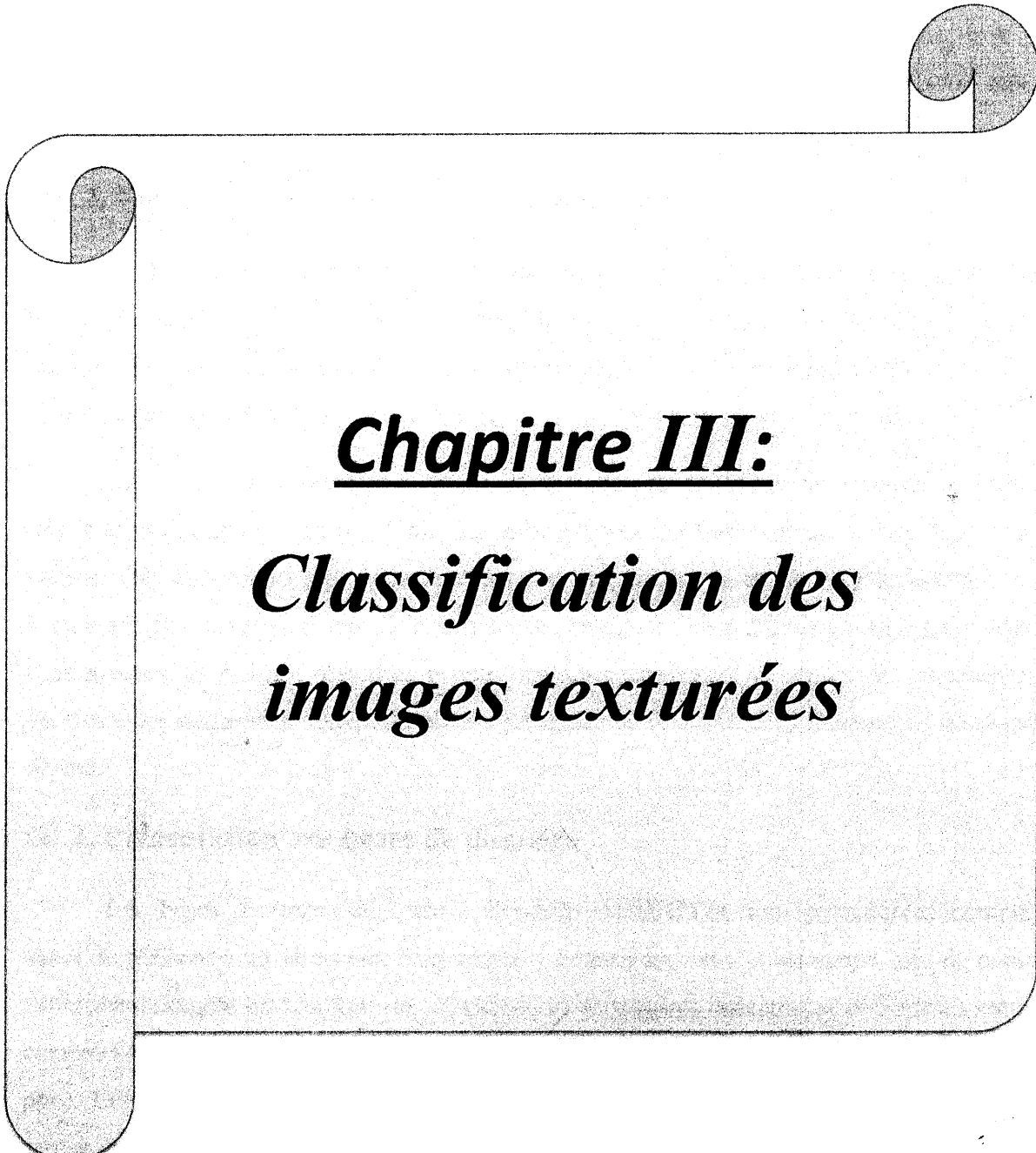


Figure II.11 : Le seuil de la courbe de ROC.

II.7. Conclusion :

L'étape de la classification est une étape importante dans les systèmes de reconnaissances. Ce chapitre est une introduction au domaine de classification des données. Il présente les différentes méthodes de classification avec plus de détails sur la méthode K-PPV ainsi que l'étape d'évaluation du Classifieur. Une bonne méthode de classification minimise les erreurs et augmente la robustesse du système de la reconnaissance parce qu'on prend la décision à partir du résultat de la classification.

A decorative graphic of a scroll with a vertical strip on the left side and rounded corners. The scroll is light gray with a thin black border. The text is centered within the scroll.

Chapitre III:
***Classification des
images texturées***

III.1. Introduction

Dans ce travail, l'objectif d'utiliser la classification est de vérifier la performance des descripteurs de texture choisis et sélectionnés pour le modèle de texture et de trouver les classes appropriées aux images fournies au système. Afin de pouvoir évaluer notre méthode, nous avons utilisé deux bases de données constituées des images de différentes textures.

III.2. Fusion d'informations ou combinaison d'informations

L'idée de combiner des informations venant de plusieurs sources et portant sur un même phénomène, est une idée très simple qui vient naturellement à l'esprit. Nous faisons souvent cette combinaison dans notre vie quotidienne. Par exemple, un expert humain reçoit l'information de plusieurs sources d'entrée, l'assimile et propose des conclusions.

En fait, l'idée principale de la fusion de données est de combiner différentes informations relatives à un problème, les informations fusionnées pouvant être de natures très variées. Ces informations peuvent être des données provenant de sources différentes ou bien présentant des caractéristiques différentes, extraites des mêmes données initiales. On identifie trois niveaux de fusion : fusion au niveau des données, fusion au niveau des caractéristiques décrivant les données et fusion au niveau de la décision. Nous nous intéressons au deuxième niveau.

III.3. Présentation des bases de données

Les bases d'images de texture **Brodatz** et **UIUCTex** sont considérées comme des bases de référence car elles sont fréquemment employées dans la littérature afin de comparer expérimentalement les résultats de classification de textures obtenus par différentes méthodes comme filtre de Gabor et GLCM avec l'algorithme de classification k plus proche voisin (k -ppv). Les paragraphes suivants décrivent ces bases d'images (Brodatz, UIUCTex) car elles seront utilisées pour présenter et comparer les résultats de notre travail[32].

III.3.1. Description de la base Brodatz

L'album de Brodatz est le premier standard pour l'évaluation des algorithmes de texture tels que la classification, la segmentation, la reconnaissance d'objets, etc. La base «Brodatz» est un ensemble de l'album composée de 112 classes. La Figure III.1 donne un exemple sur les différentes classes de cette base. Nous avons utilisé 32 classes, la taille des images est de 640×640 pixels en format GIF.

Cependant, le nombre d'échantillons pour chaque classe de textures est faible. C'est pourquoi, la plupart des auteurs qui utilisent cette base d'images sont amenés à diviser les images en "imagettes" afin d'augmenter le nombre d'échantillons.

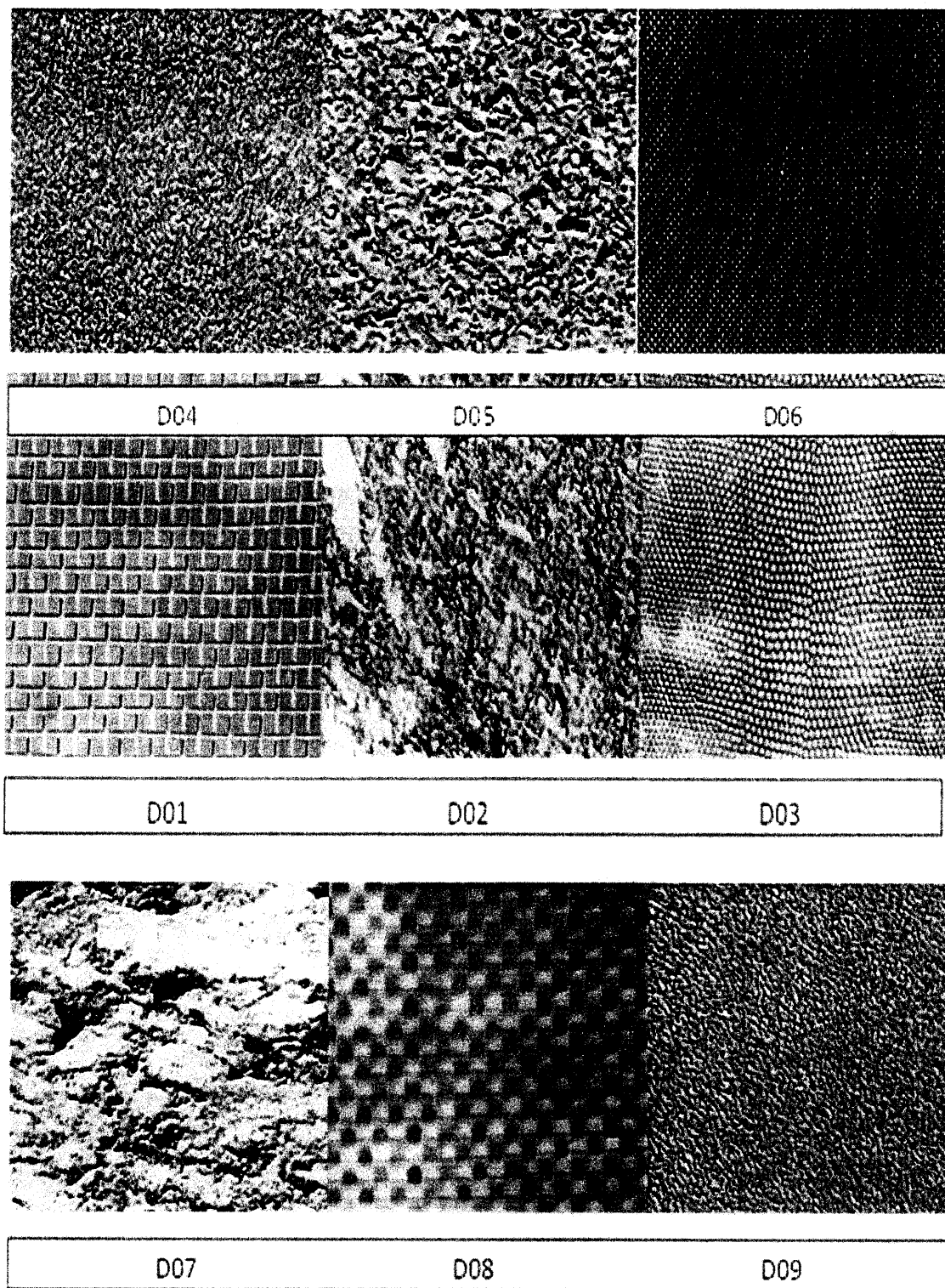


Figure.III.1. Exemples d'image des textures de Brodatz [34].

III.3.2. Description de la base UIUCTex

La base de données d'UIUCTex contient 25 classes de texture avec 40 images par classe. Les textures sont vues sous différentes échelles avec des changements de déformations et des variations d'apparence. Figure III.2 présente quatre images d'échantillon par classe, chacune montre une surface texturisée vue à des positions différentes.

III.4. Implémentation et résultats

III.4.1. Extraction des caractéristiques

L'extraction de caractéristiques de texture est la phase élémentaire de la classification et la segmentation de texture. De bon résultat à cette étape facilitent les étapes suivantes.

III.4.1.1. Filtre de Gabor

Les images sont divisées en imagerie de taille 128x128 (on aura pour chaque image 16 imagerie). On a utilisé le filtre de Gabor décrit dans le chapitre I par l'utilisation de 6 orientations (0° , 30° , 60° , 90° , 120° , 150°) et 5 fréquences.

Après l'application des filtres de Gabor, nous obtenons des coefficients qui décrivent la texture de l'image. Pour chaque paire de fréquence et d'orientation on a un nombre de coefficients de la même taille que l'image, c.-à-d., nous avons 128x128x5x6 coefficients pour chaque image. L'utilisation de ce nombre de coefficients n'est pas pratique pour la classification. Pour cette raison nous avons choisi d'utiliser deux paramètres pour chaque image : la **moyenne d'amplitude** et l'**énergie** des coefficients de Gabor. De ce fait on aura donc 60 caractéristiques pour chaque image ($5*6*2=60$). Les figures III.3, III.4, III.5, III.6, III.7 donne les résultats de l'application des filtres de Gabor sur une image de la base Brodatz. Avec les orientations 30° , 60° , 90° , 120° , 150° (de haut en bas).

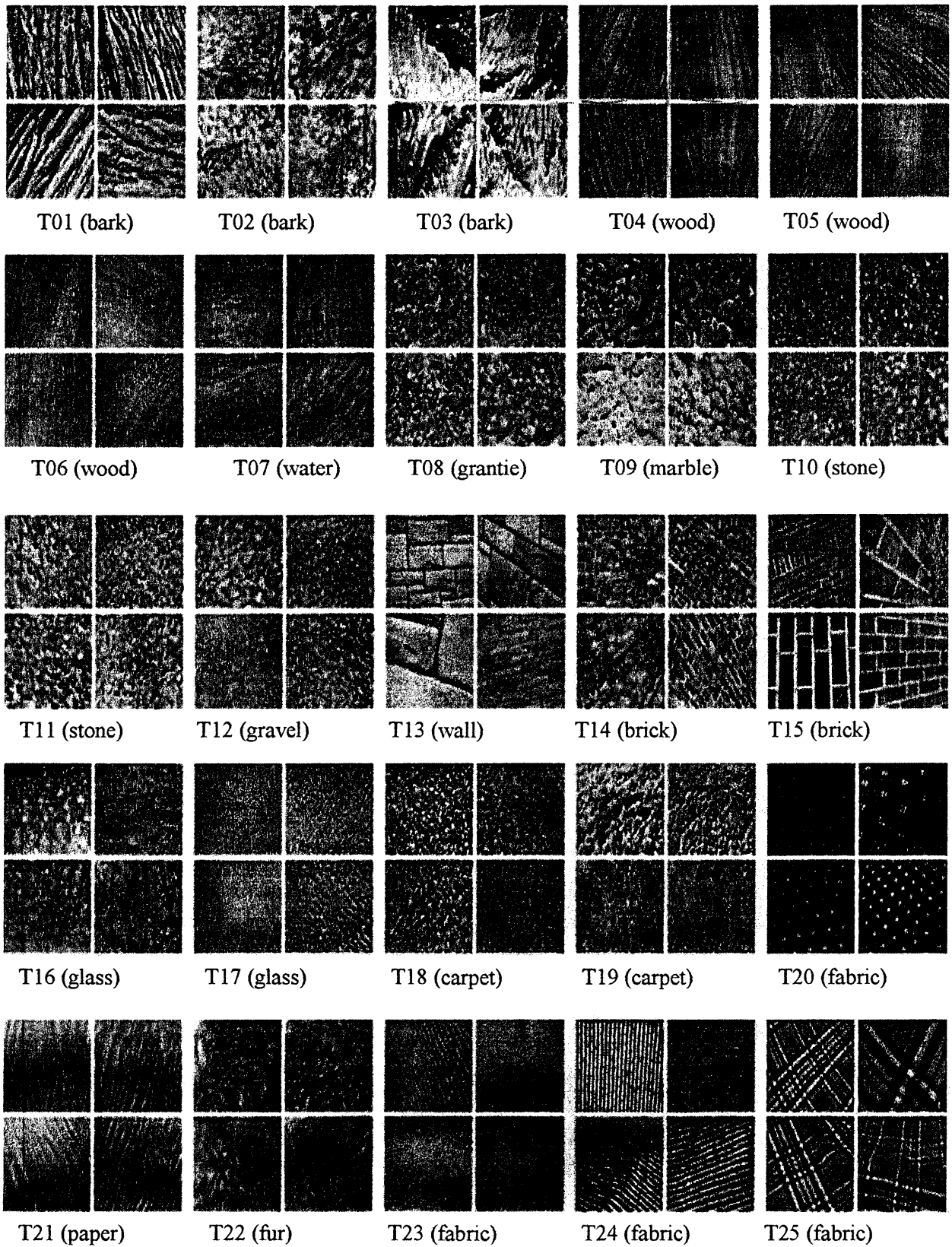


Figure.III.2. 4 échantillons de chaque classe de texture de la base UIUCTex[35].

Exemple de filtre de Gabor sur une image de base de données Brodatz :

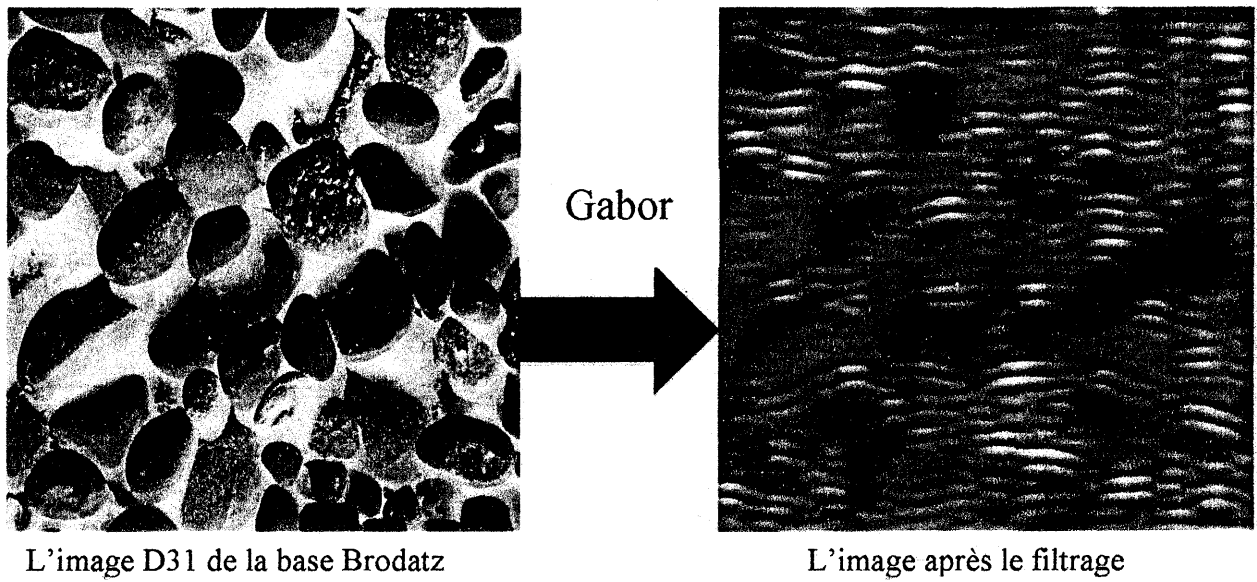


Figure III.3. L'image D31 après le filtrage

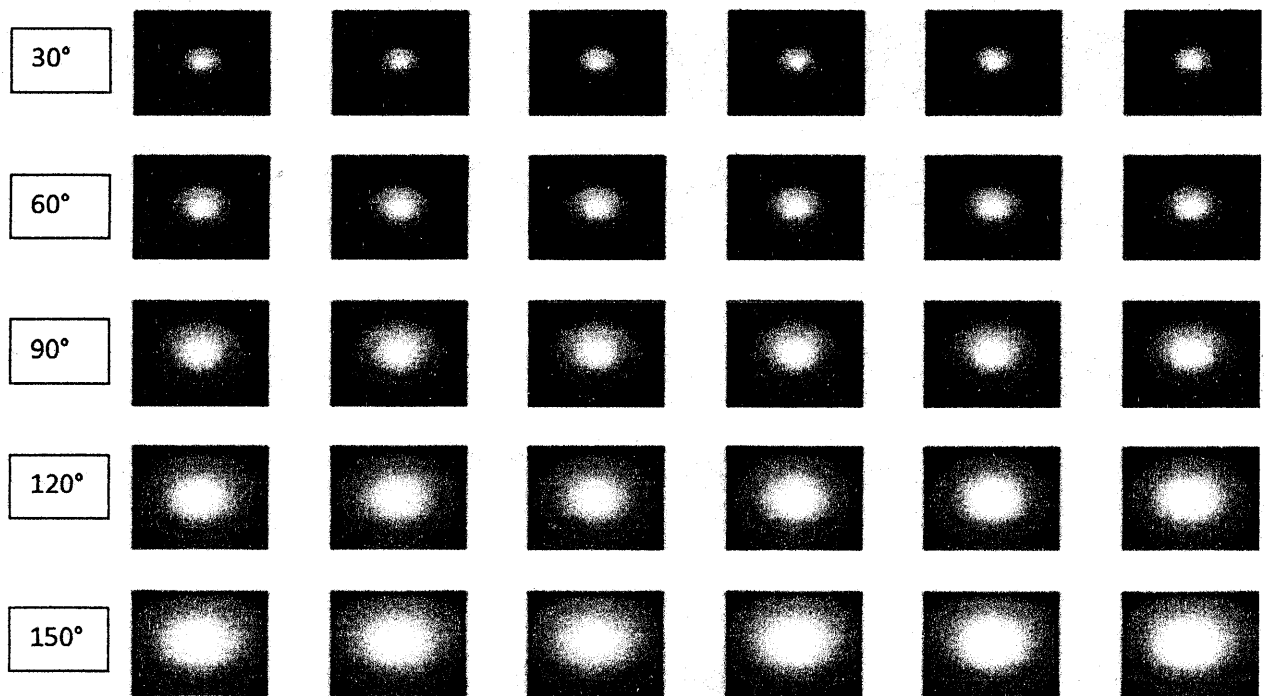


Figure III.4. Amplitude des filtres de Gabor

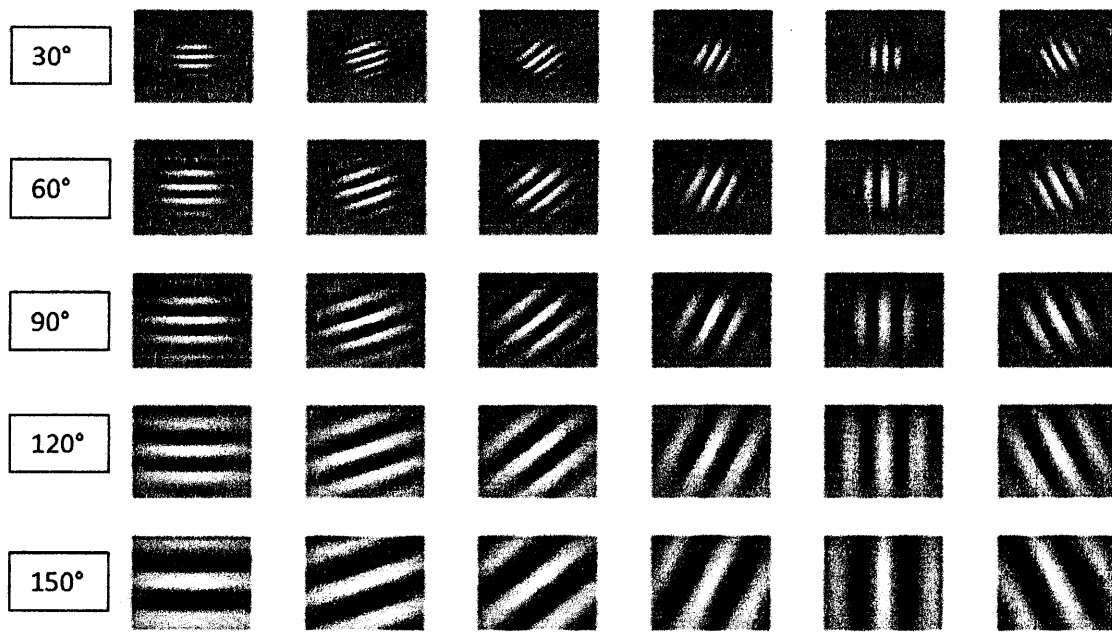


Figure III.5. Partie réelle des filtres de Gabor

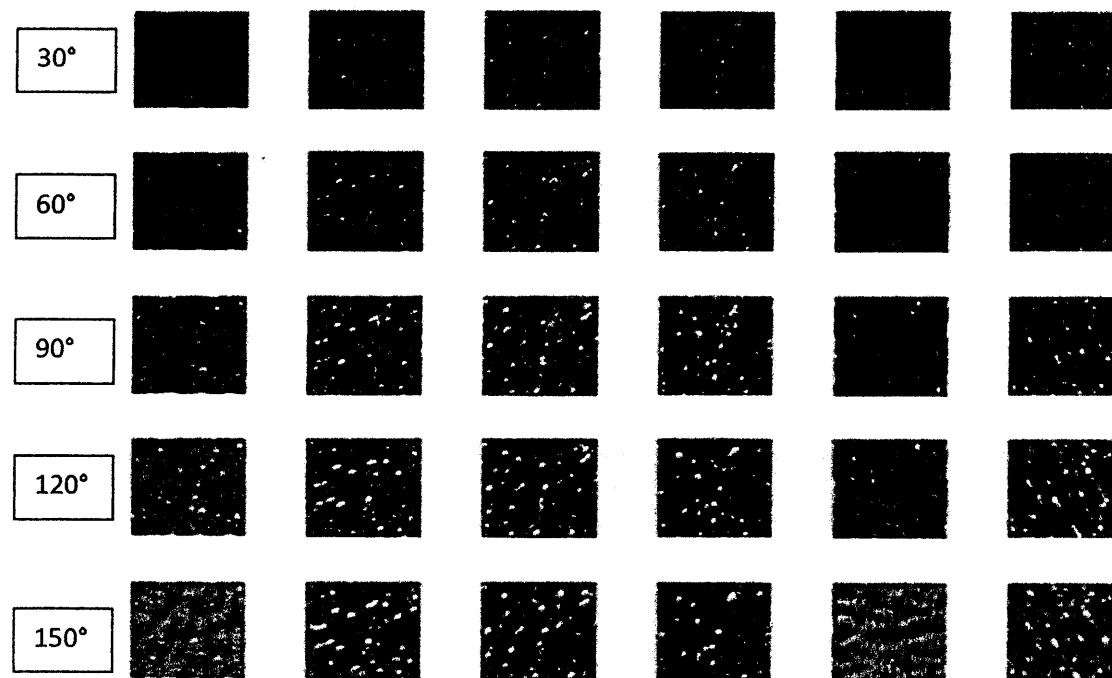


Figure III.6. Amplitude de l'image

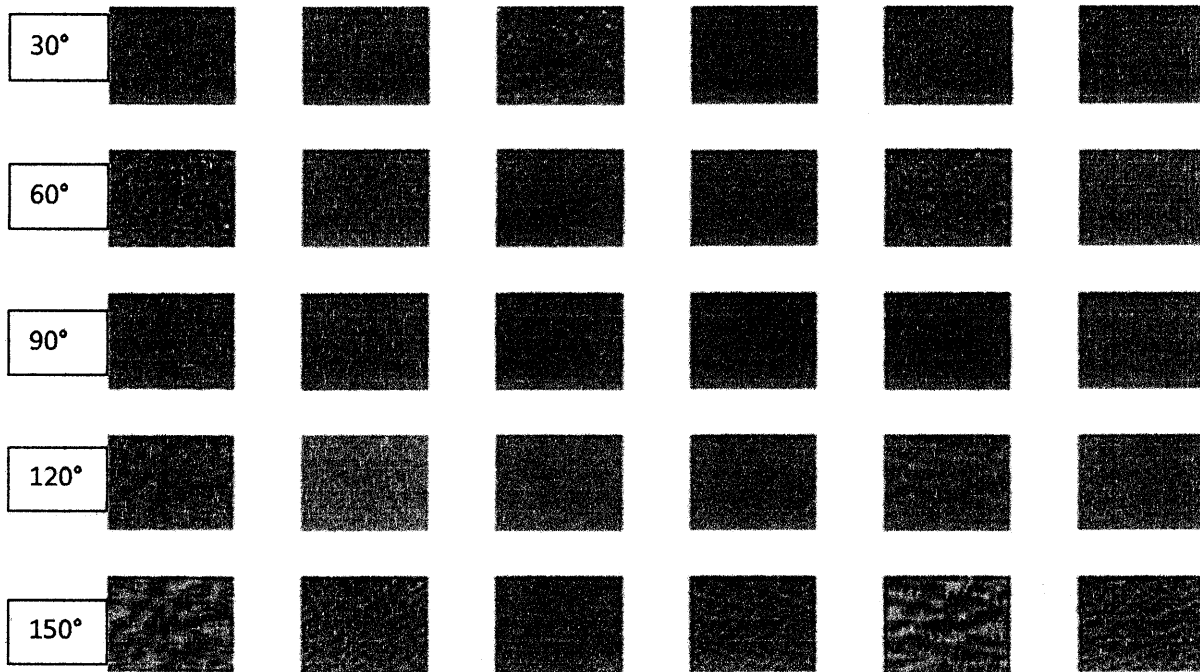


Figure III.7. Partie réelle de l'image

III.4.1.2. La matrice de co-occurrence

Le deuxième descripteur d'image utilisé est la matrice de co-occurrence introduit par Haralick [7]. Il permet de décrire la fréquence relative avec laquelle deux pixels séparés par une distance D sous angle θ se produisent dans une image en niveaux de gris (Figure.III.8).

L'utilisation d'une approche statistique comme la matrice de co-occurrence aidera à fournir des informations sur la position relative des pixels voisins dans une image.

On a divisée les images en imagerie de taille 128x128 (chaque image est a divisée en 16 imagerie). On utilise les paramètres suivants :

- Distance (1, 2, 3, 4).
- orientation (0° , 45° , 90° , 135°).

La figure III.9 montre les rapports spatiaux des pixels qui sont définis par ce choix, où D représente la distance du Pixel d'intérêt.

➤ l'exemple suivant explique la méthode GLCM dans la base de données UIUCTex.

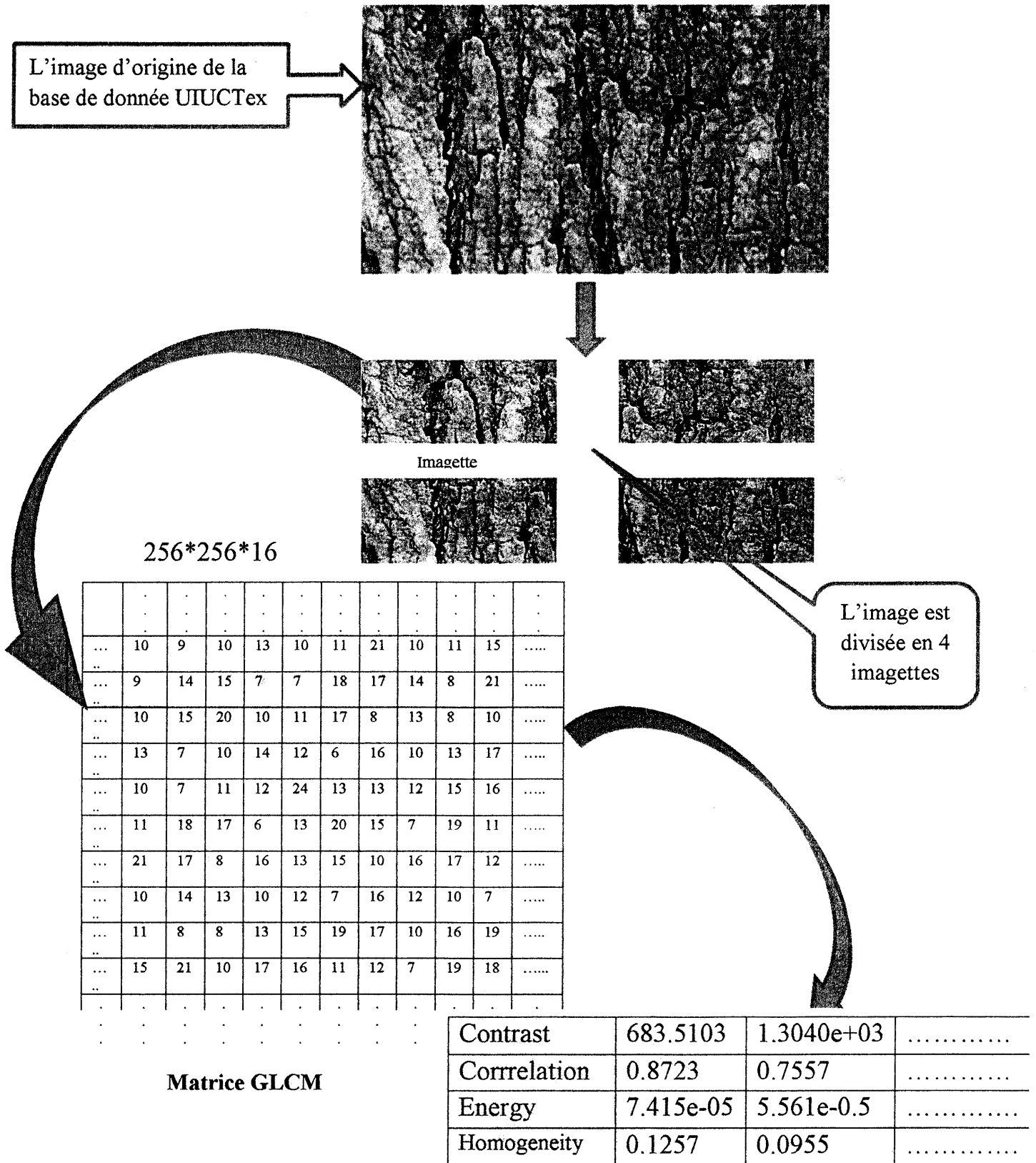
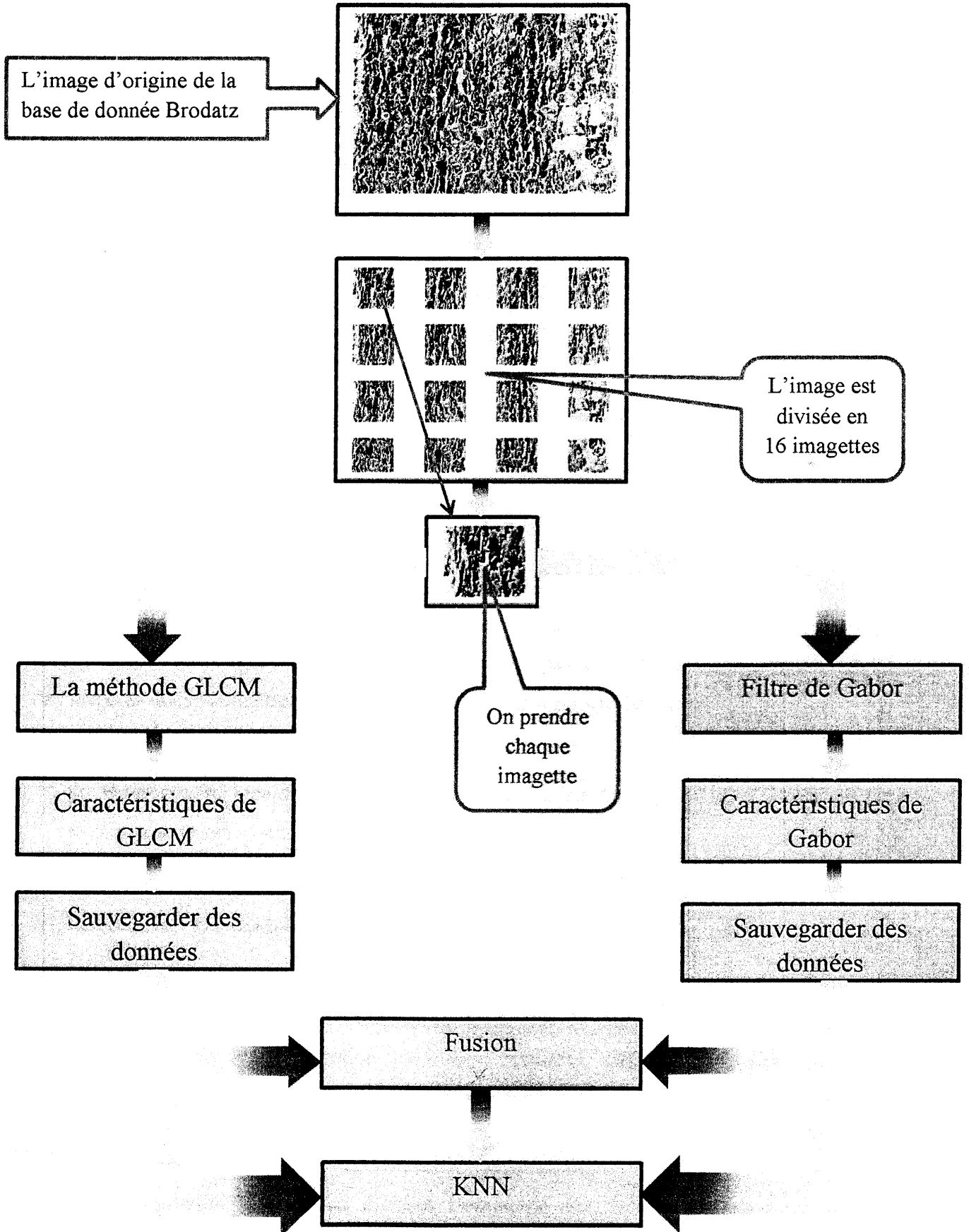


Figure III.8. La méthode GLCM dans la base de données UIUCTex

➤ on peut résumer notre travail par le diagramme suivant :



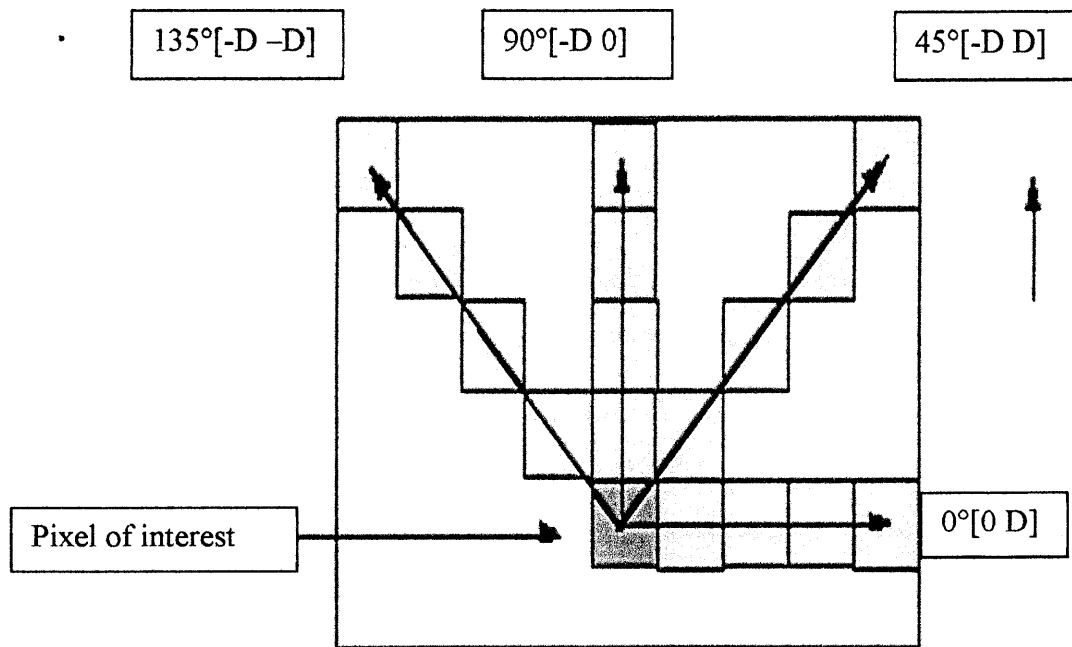


Figure.III.9. Voisinage d'un pixel selon 4 directions principales et 3 distances pour l'extraction des matrices de co-occurrences



Les matrices de co-occurrences contiennent une masse importante d'informations difficilement manipulable. C'est pour cela qu'elle n'est pas utilisée directement mais à travers des mesures dites paramètres d'Haralick : On a calculé 4 caractéristiques :

- ❖ **Énergie** (voir l'expression I.5 de chapitre I).
- ❖ **Homogénéité locale** (voir l'expression I.6 de chapitre I).
- ❖ **Contraste** (voir l'expression I.3 de chapitre I).
- ❖ **Corrélation** (voir l'expression I.4 de chapitre I).

III.4.2. Résultats et discussions

Nous avons testé les deux méthodes d'extraction de caractéristiques : les filtres de Gabor et les attributs de Haralick extraits de la matrice de GLCM. Pour évaluer les deux méthodes en termes de classification des images texturées, nous les avons testés sur deux bases d'images de texture différentes en utilisant la méthode de classification K-PPV.

L'évaluation de la classification est effectuée par deux méthodes :

- La validation simple : en divisant l'ensemble des échantillons en deux parties aléatoirement, une partie pour l'apprentissage (80%) et une partie pour le test (20%).

- La validation croisée : en divisant l'ensemble des échantillons en quatre parties aléatoirement, pour chaque itération (4 itérations) une partie est utilisée pour le test et les 3 autres parties sont utilisées pour l'apprentissage, l'erreur de classification est la moyenne des 4 erreurs obtenues pour les 4 parties de test

Les résultats de la validation simple et croisée sont présentés sur les tableaux suivants (1,2,3,4,5,6,7,8) pour la base de données UIUCTex pour les paramètres suivants

- La distance utilisée pour calculer la matrice GLCM
- Le niveau de quantification du niveau de gris N
- Le $k=1,3$ qui spécifie le nombre de voisins retenus pour le classifieur KNN

Les résultats sont exprimés en termes du taux d'erreurs(%)

Base de données	La méthode utilisée	1NN	3NN
BRODATZ	GLCM (N=256, d=1, 2, 3, 4)	23.36	26.58
	Gabor	19.13	20.58
	Gabor et GLCM (N=256, d=1, 2, 3, 4)	19.82	19.89

Tableau.III.1. 4 Validation croisée (Brodatz)

Base de données	La méthode utilisée	1NN	3NN
UIUCTex	GLCM (N=256, d=1, 2, 3, 4)	35.55	39.60
	Gabor	11.15	15.73
	Gabor et GLCM (N=256, d=1, 2, 3, 4)	11.33	14.90

Tableau.III.2. Validation croisée (UIUCTex)

<u>GLCM</u>	Base des données	1 KNN	3 KNN
N= 8, d=1, 2, 3, 4	uiuctex	1.0000e ⁻⁰¹	0.20
	brodatz	0.78	0.78
N= 16, d=1, 2, 3, 4	uiuctex	3.33	5.50
	brodatz	4.69	4.69
N= 64, d=1, 2, 3, 4	uiuctex	28.58	33.25
	brodatz	8.59	10.94
N= 128, d=1, 2, 3, 4	uiuctex	35.25	39.07
	brodatz	10.16	10.35
N= 256, d=1, 2, 3, 4	uiuctex	35.35	40.78
	brodatz	10.16	10.51
<u>Gabor</u> Niveau de gris : 256	Uiuctex	35.35	40.78
	brodatz	19.13	20.27

Tableau.III.3. 4 validation croisé par les méthodes GLCM et Gabor sur Brodatz et UIUTEX

	d=1					
	N=8	N=16	N=32	N=64	N=128	Gabor
1NN	8.33	9.38	12.50	14.58	16.67	9.32
3NN	11.43	10.42	15.63	17.71	18.75	11.46

Tableau.III.4.Valiation simple (Brodatz)

	d=1			
	N=8	N=16	N=32	N=64
1NN	7.29	8.33	9.38	10.42
3NN	9.38	10.42	14.58	12.50

Tableau.III.5. Valiation simple (fusion Gabor + GLCM) (Brodatz)

	d=1					
	N=8	N=16	N=32	N=64	N=128	Gabor
1NN	11.13	34.88	43.13	53.31	54.63	11.13
3NN	16.38	44.25	51.38	50.96	59.50	16.38

Tableau.III.6. Valiation simple (UIUCTex)

	d=1	d=2	d=1,2
N=8	0	0	0
N=16	0.39	0.98	1.17
N=32	3.13	6.64	4.88
N=64	7.23	10.94	7.03
N=128	13.48	12.89	9.96

Tableau.III.7. Valiation croisée (Brodatz)

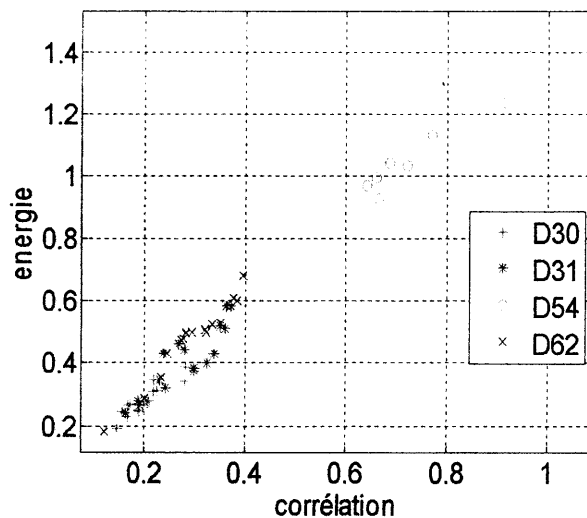
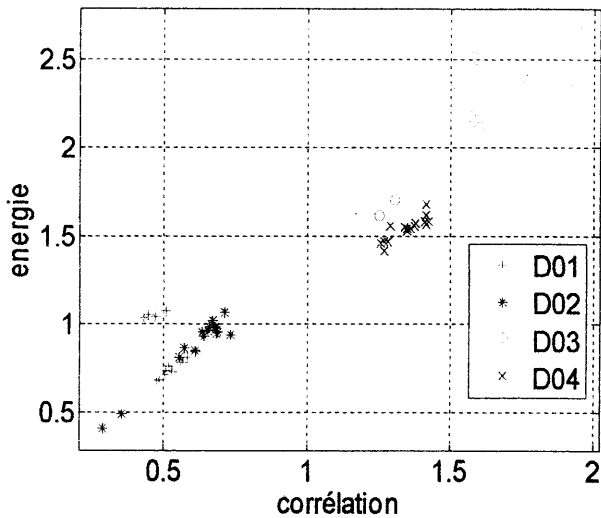
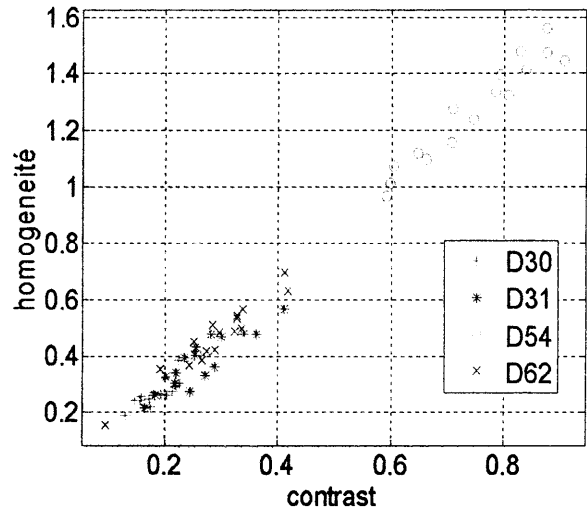
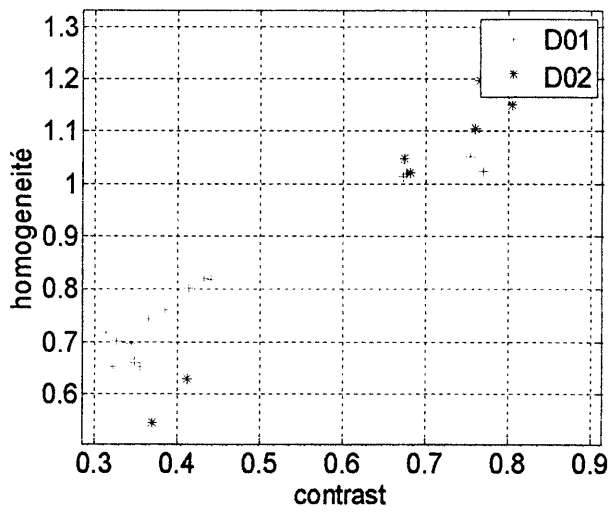
	d=1	d=2	d=1,2
N=8	0	0	7.5 e-002
N=16	0.35	0.98	2.43
N=32	6.33	11.30	4.27
N=64	24.32	28.23	29.48
N=128	42.40	44.05	41.03

Tableau. III.8. Valiation croisée (UIUCTex)

D'après les résultats obtenus, on peut conclure que :

- La performance du classifieur 1NN dépasse celle du 3NN à cause de la similarité de la texture pour les classes différentes.
- Les résultats montrent que la distance entre les pixels et le niveau de gris utilisés dans le calcul de la matrice GLCM peuvent affecter le taux d'erreur
- La distance spatiale entre les pixels est peut être changé selon les images et le type d'application désiré. Elle dépend de la distance spatiale qui donne une signification aux paires des pixels
- Pour la nôtre implémentation, les meilleurs résultats sont obtenu avec la distance égale à un (d=1) (avec les 4 orientations)
- Les niveaux de gris N=8, 16 utilisés dans la matrice GLCM donne les meilleurs résultats en termes de classification, parce qu'à ce niveau peu de niveau de gris sont quantifiés et visualisé par rapport aux niveaux de gris présenté sur les images allant de 0 à 256.
- Les filtres de Gabor donnent des bons résultats pour la classification. L'efficacité des filtres de Gabor est prouvée dans plusieurs domaines et c'est le cas pour la classification. En effet, cette méthode peut caractériser les différentes formes contenu dans l'images et pour des échelles différentes à cause de l'utilisation des 6 orientations et les 5 fréquences différentes ce qui donne des informations précieuses sur la texture présenté dans l'image.

- La fusion entre les deux méthodes GLCM et Gabor neuit légèrement réduire le taux d'erreur et améliorer la classification.
- Les deux méthodes sont des très bons outils pour l'extraction et la caractérisation de la texture à cause de leurs capacités de discriminer entre les différentes classes de texture.
- Cette capacité est illustrée sur les figures III.10 qui visualisent les caractéristiques utilisées en deux dimensions pour les différentes classes de texture des deux bases de données. Le regroupement des classes dans des zones séparées donne une idée sur la discrimination des caractéristiques utilisées.



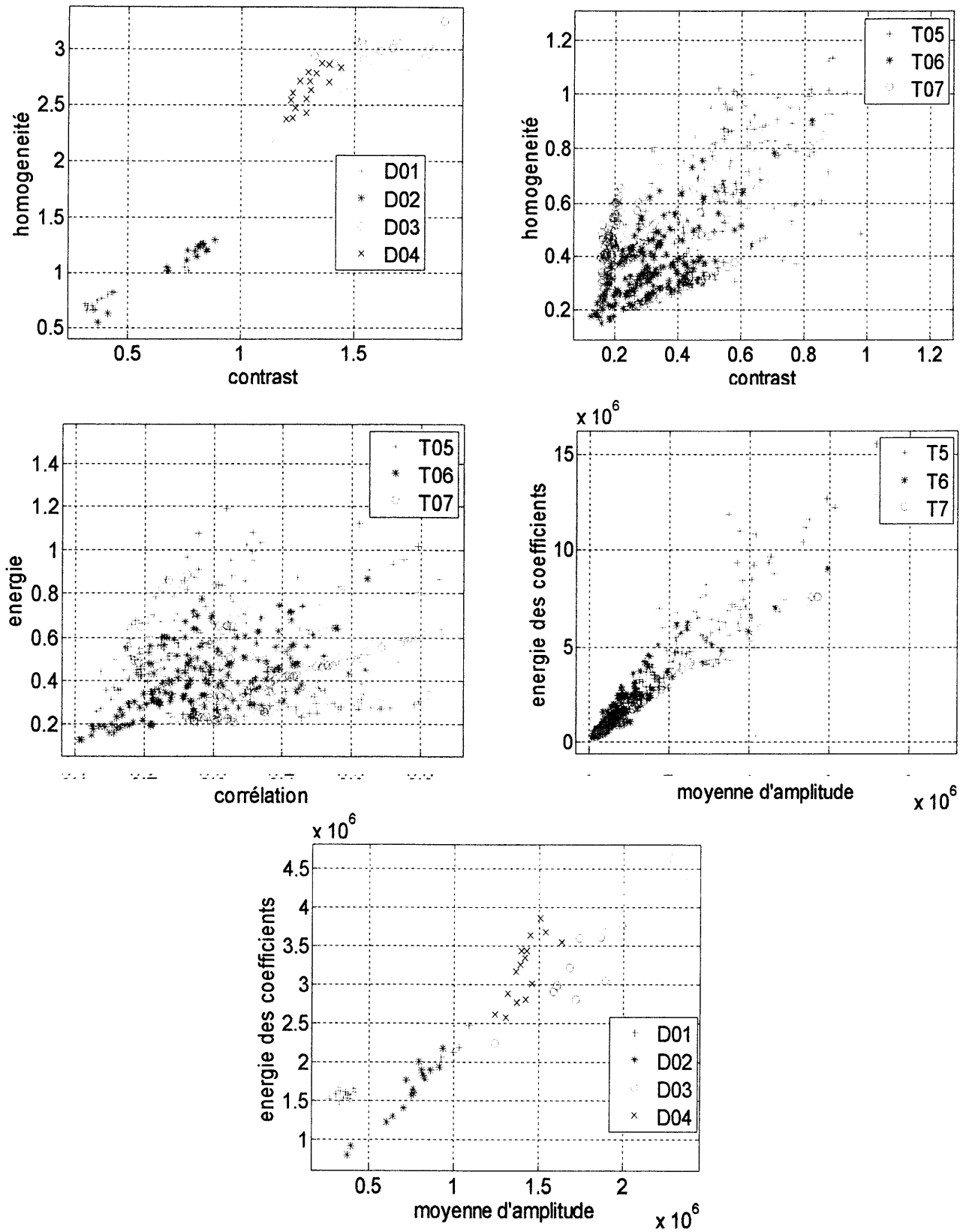


Figure III. 10. Visualisation des caractéristiques en 2D

III.5. Conclusion

Nous avons présenté dans ce chapitre l'implémentation et les résultats de notre travail.

La matrice de co-occurrence et les filtres de Gabor sont utilisés pour caractériser la texture. Nous avons évalué les deux méthodes pour la classification de texture par la méthode KNN, nous avons aussi testé la fusion des deux méthodes de bons résultats sont obtenus lors de l'évaluation prouvant ainsi l'efficacité des filtres de Gabor et la matrice GLCM dans la caractérisation et la classification des textures.

Conclusion



Générale

Conclusion général

Dans ce travail, nous avons fait la classification des images de textures en utilisant deux méthodes d'extraction des caractéristiques.

Il existe plusieurs méthodes pour caractériser la texture, on a choisi deux méthodes. La première est la matrice GLCM, cette méthode donne une description de la structure locale ainsi qu'une information sur le champ aléatoire qui contrôle les détails à petite échelle. Cette approche est basée sur la probabilité jointe de la distribution des pixels dans l'image. l'élément $P(i,j)$ de la matrice de co-occurrence définit la fréquence d'apparition des couples de niveaux de gris i,j pour les couples de pixels séparés par une distance d selon la direction θ . Donc le calcul de la matrice nécessite le choix d'une distance et d'un angle de déplacement.

La deuxième méthode est une méthode fréquentielle, ce sont les filtres de Gabor. Une fonction de Gabor bidimensionnels comprend une onde sinusoïdale d'une certaine fréquence et l'orientation modulée par une enveloppe gaussienne. Il s'agit des filtres extrayant une information fréquentielle localisée et orientée. Les filtres de Gabor sont le produit de Gaussiennes par des sinus ou des cosinus, le Sinus / cosinus : analyse en fréquences et la Gaussienne : différents niveaux de lissage (échelles).

L'étape de la classification vise à déterminer la classe de l'image. L'algorithme des k -plus proches voisins K-PPV est une méthode permettant de classer les objets en fonction des exemples d'entraînement étiquetés les plus proches dans l'espace des caractéristiques. Elle se base sur une comparaison directe entre le vecteur de caractéristiques de l'entité x à classer et les vecteurs de caractéristiques des entités de référence. La comparaison consiste en un calcul de distances entre ces entités. L'entité à classer est assignée à la classe majoritaire parmi les classes des k exemples les plus proches au sens de la distance utilisée. Dans notre étude, nous utilisons une distance euclidienne.

Dans ce mémoire, nous avons appliqué les méthodes citées ci-dessus sur deux bases de donnée (Brodatz et UIUCTex) constitués des images à niveau de gris. On a aussi combiné les deux méthodes d'extraction des caractéristiques afin d'améliorer la classification.

L'évaluation est effectuée par deux types de validation : validation simple et validation croisée, les résultats montrent des bons taux de classification pour les deux algorithmes d'extraction de caractéristiques malgré la diversité et la complexité des images de textures utilisées. La méthode de GLCM donne des résultats meilleurs pour des distances faibles et la fusion des deux méthodes peut améliorer le taux de classification.

Comme perspectives, on propose de :

- ❖ Utiliser d'autres méthodes d'extraction des caractéristiques.
- ❖ Exploiter les méthodes utilisées dans ce mémoire pour une autre application tel que l'indexation des images.

Bibliographies

- [1] C. Germain, M.KUNT. "Contribution à la caractérisation multi-échelle de l'anisotropie des images texturées", Université Bordeaux I, thèse de doctorat, Décembre 1997.
- [2] Yaovi GAGOU. "Cours de Traitement d'Image", Université de Picardie Jules Verne Année Académique 2007-2008.
- [3] F. Tomita, Y., and S. Tsuji. "Description of textures by a structural analysis". IEEE Transactions on Pattern Analysis and Machine Intelligence, 4(2):183-191, march 1982.
- [4] T. Matsuyama, S.-I. Miura and M. Shirai Nagao. "Structural analysis of natural textures by fourier transformation". Computer Vision, Graph. and Image Proc., 24:347-362, 1983.
- [5] F.M. Vilnrotter, R. Nevatia, and K.E. Price. "Structural analysis of natural textures". IEEE Trans. on Pat. Anal. and Mach. Intel., 8(1):76-89, january 1986.
- [6] A. Gagalowicz. "Vers un modèle de textures". PhD thesis, Université Pierre et Marie Curie Paris VI, mai 1983.
- [7] Shanmugam K, Dinstein I Haralick R.M, "Textural Features for Image Classification" , November 1973.
- [8] D. A. Clausi et Y. Zhao. "Grey level cooccurrence integrated algorithm (GLCIA): a superior computational method to rapidly determine cooccurrence probability texture Features". Computers and geosciences, 29:837-850, 2003.
- [9] S.W. Zucker and D. Terzopoulos. "Finding structure cooccurrence matrices for texture analysis". Computer Vision Graphics and Image Processing, 12:286-308, 1980.
- [10] D. Gabor, "Theory of communication". J. IEE, 93 : pp 492-457, 1941.
- [11] Weldon, T. P., Higgins, W. E. et Dunn, D. F. (1996). "Gabor filter design for multiple texture segmentation", Optical Engineering, 35(10): pp 2852-2863, 1996.
- [12] [www. Transformée de Fourier.htm](http://www.Transformée de Fourier.htm).
- [13] Christopher Rasmussen, "Texture", CIS 489/689 Computer Vision, Univ. of Delaware (USA).

- [14] Mr. RABIA Nacim. “Revue des méthodes de segmentation d’images texturées : cas des Images couleur”, Mémoire, UNIVERSITE MOULOUD MAMMERI DE TIZI OUZOU 30/06/2011.
- [15] [www. Transformée de Fourier.htm](#)
- [16] John G. Daugman 1985: “uncertainty relation for resolution in space, spatial frequency, and Orientation optimized by two-dimensional visual cortical filters“,
- [17] C. Rosenberger and K. Chehdi, “Genetic Fusion: Application to multi-components image segmentation”, ICASSP 2000, Vol. 3, pp. 2223-2226, Istanbul, Turkey, 2000.
- [18] Guillaume ROUSSEL. “Développement et évaluation de nouvelles méthodes de classification spatiale-spectrale d’images hyperspectrales”, Doctorat de l’université de Toulouse, 2012.
- [19] Y. Linde, A. Buzo, and R. M. Gray. “An algorithm for vector quantizer design”. IEEE Trans. Commun., pages 84–95, 1980.
- [20] T. M. Cover and P. E. Hart. “Nearest neighbor pattern classification”. IEEE Transactions on Information Theory, pages 21–27, 1967.
- [21] C. J. C. Burges. “A tutorial on support vector machines for pattern recognition”. Data Min. Knowl. Discov., 2 :121–167, 1989.
- [22] B. Krose and P. V. D. “Smagt. An introduction to neural network”, 1996.
- [23] A. H. Strahler. “The use of prior probabilities in maximum likelihood classification of remotely sensed data”. Remote Sensing of Environment, pages 135–163, 1980.
- [24] A.K. Jain, R.P.W. Duin, et J. Mao. “Statistical pattern recognition : A review. IEEE Transactions on Pattern Analysis and Machine Intelligence”, 22(1) :4–37, 2000.
- [25] A.K. Jain, M.N. Murty, et P.J. Flynn. “Data clustering : a review. ACM Computing Surveys”, 31(3) :264–323, 1999.
- [26] Imane NEDJAR. “CMBIR (Content Medical Based Image Retrieval) développement d’outil logiciel d’annotation d’images médicales, utilisant les méthodes d’indexation par