

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITÉ MOULOUD MAMMERRI, TIZI-OUZOU



Faculté de Genie Electrique et D'informatique

Département d'Electronique

THÈSE DE DOCTORAT

en **Electronique**

Option Télédétection

Présentée par

M^{me} OUSLIMANI ép. OUTOUDERT Farida

Etude comparative des techniques de codage d'images en vue d'une segmentation

Soutenue le 21/10/ 2018 devant le jury d'examen composé de :

Mr.AMEUR Soltane	Professeur à l'UMMTO	Président
Mme.AMEUR Zohra	Professeur à l'UMMTO	Rapporteur
Mr.OUSLIMANI Achour	Professeur à l'ENSEA, France	Co/Rapporteur
Mr.CHITROUB Salim	Professeur à l'USTHB, Alger	Examineur
Mme.CHOUIREB Fatima	Professeur à l'UAT, Laghouat	Examinatrice
Mr.DJEDDI Mabrouk	Professeur à l'UMB, Boumerdes	Examineur

Remerciements

Le travail présenté dans cette thèse a été réalisé dans deux laboratoires de recherche : Le laboratoire LAMPA (Laboratoire d'Analyse et de Modélisation des Phénomènes Aléatoires) de l'Université Mouloud Mammeri de Tizi-Ouzou et le laboratoire Quartz (EA 7393) de l'Ecole Nationale Supérieure de l'Electronique et ses Applications (ENSEA), France.

J'adresse mes vifs remerciements à ma directrice de thèse Madame AMEUR Zohra, professeur à l'UMMTO d'avoir accepté de diriger mes travaux de thèse ainsi que pour son suivi, ses orientations et ses encouragements.

Je souhaite adresser ma profonde reconnaissance à Monsieur OUSLIMANI Achour, professeur à l'ENSEA également mon directeur de thèse, de m'avoir accueillie au sein du laboratoire Quartz. J'ai énormément appris en travaillant à ses côtés. Ses conseils, sa rigueur scientifique et ses valeurs humaines m'ont bien guidée tout au long de mon projet.

Je tiens à remercier Monsieur AMEUR Soltane, Professeur à l'UMMTO et directeur du laboratoire LAMPA pour avoir accepté de présider le jury de cette thèse, ainsi que pour ses précieux conseils et ses encouragements.

Mes remerciements vont également à Monsieur CHITROUB Salim, Professeur à l'USTHB, pour m'avoir fait honneur en acceptant de faire partie du jury de cette thèse.

Que Madame CHOUIREB Fatima, Professeur à l'UAT trouve ici l'expression de toute ma gratitude pour l'intérêt qu'elle a porté à mon travail en acceptant de faire partie du jury de cette thèse.

Je remercie vivement Monsieur DJEDDI Mabrouk, Professeur à l'UMB d'avoir accepté de participer au jury de ma thèse.

Je remercie tous les membres du laboratoire Quartz pour leur accueil et d'avoir mis à ma disposition tous les moyens pour réaliser mon travail. Grâce à l'ensemble du personnel, j'ai pu travailler dans un cadre particulièrement agréable.

J'exprime également mes remerciements à tous les collègues et amis de la faculté de Génie Electrique et d'Informatique et en particulier à ceux du département d'Electronique.

Merci à mes amis et à ma famille pour leur soutien et leurs encouragements ainsi que d'être toujours là pour moi.

Je remercie Karim pour sa présence, son écoute, son soutien et ses encouragements sans oublier mes adorables enfants Amayas et Ines.

Le dernier MERCI s'adresse à ma Mère, tous les mots du monde ne pourraient exprimer ma profonde gratitude pour tous les efforts et les sacrifices, que tu n'as jamais cessé de consentir pour tes enfants.

Liste des abbreviations

BRINT: Binary Rotation Invariant and Noise Tolerant

CLBP: Completed Local Binary Pattern

CUReT: Columbia-Utrecht Reflectance and Texture

DC: Directional Coding

DLBP: Dominant Local Binary Pattern

DNS: Dominant Neighborhood Structure

GLCM: Grey Level Co-occurrence Matrix

KNN: K Nearest Neighbor

LBC: Local Binary Count

LBP: Local Binary Pattern

LCP: Local Contrast Pattern

LDRC: Local Directional Rank Coding

LQP: Local Quinary Pattern

LTP: Local Ternary Pattern

RBF: Radial Basic Function

RMSE: Root Mean Squared Error

RRC: Regional Rank Coding

MBP: Median Binary Pattern

SIFT: Scale Invariant Feature Transform

SNR: Signal to Noise Ratio

SVH: Système Visuel Humain

SVM: Support Vector Machine

Table des figures

Fig.1.1. Perception des textures (a) Texture régulière et texture irrégulière (b) texture grossière et texture fine (c) texture directionnelle et texture non directionnelle (d) texture lisse et texture rigoureuse	6
Fig.1.2. Exemples de textures déterministes tirées de l'album de Brodatz [Brodatz,1965].	7
Fig.1.3. Exemples de textures stochastiques tirées de l'album de Brodatz [Brodatz,1965].	7
Fig.1.4. Exemples de textures directionnelles tirées de l'album de Brodatz [Brodatz,1965].	8
Fig.1.5. Exemple d'une même texture photographiée à 2 résolutions différentes.	8
Fig.1.6. a) Image composée de cinq régions texturées : "raphia", "eau", "coton", "sable", et "chevron" b) classification c) segmentation.	8
Fig.1.7. Deux images composées de paires de texture de même statistiques de second-ordre : (a) L'œil humain ne peut pas apercevoir les deux régions sans un examen minutieux, (b) les deux régions sont immédiatement discriminées par l'œil humain. [Tuceryan and Jain, 1998].	10
Fig.1.8. Calcul de Matrice de cooccurrence sur l'image test (a) utilisant quatre directions (b) 0° (c) 45° (d) 90° et (e) 135° avec la distance $d = 1$. # représente le nombre d'occurrences...	11
Fig.1.9. Partitionnement du spectre de puissance, (a) l'énergie calculée en chaque couronne fournit une information sur la finesse ou la grossièreté de la texture, (b) L'énergie calculée en chaque secteur angulaire fournit une information sur l'orientation de la texture	15
Fig.1.10. Modification de l'aspect imagé de la même texture (Plâtre B, texture # 30 de la base de données CURET [Dana et al, 1999]) avec la variation des conditions d'imagerie. Rangée du haut: angle de vue constant et éclairage variable. Rangée du bas: éclairage constant et angle de vue variable.	20
Fig.1.11. L'effet de l'échelle et de la rotation sur les textures. A gauche, la caméra a été éloignée du mur de briques. A droite, la surface a été tournée dans le plan de 90° alors que toutes les autres conditions d'imagerie ont été laissées inchangées [Varma et al, 2005].	20
Fig.1.12. Textures extraites de la base de données CURET [Dana et al,1999]. Dans la rangée supérieure, la première et la quatrième image sont de texture identique alors que toutes les autres images, même si elles se ressemblent, appartiennent à différentes classes. De même, dans la rangée du bas, les images apparaissent similaires et pourtant elles appartiennent à trois classes de textures différentes.	21
Fig.1.13. Banc de filtres MR8 composé de 38 filtres à la fois isotropiques et anisotropiques.	23
Fig.1.14. Principale différence entre les représentations VZ_Joint et VZ_MR8 est que les patches d'image source sont utilisés directement dans la représentation VZ_Joint par opposition aux réponses dérivées des filtres dans VZ_MR8 [varma et al, 2009].	24
Fig.1.15. Si on fixe $k = 1$, l'individu non classé appartient à la classe des ronds, si $k = 3$ l'individu appartient à la classe des carrés.	27
Fig.1.16. Exemple de fonctionnement du SVM dans le cas linéaire. Seuls 4 vecteurs de support sont nécessaires pour définir l'hyperplan optimal.	28

Fig.1.17. Exemple de transformation d'espace d'attributs utilisant une fonction noyau ϕ	28
Fig.1.18. Les 24 textures extraites de l'album de Brodatz.....	31
Fig.1.19. Echantillon des 24 textures des bases Outex_TC10 et Outex_TC12.....	32
Fig.1.20. 61 classes de textures de la base CURET.....	33
Fig.1.21. 11 différentes classes de texture de la base KTH-Tips2b.....	33
Fig.2.1. Pixel central g_c et ses P voisins uniformément repartis sur un rayon R	36
Fig.2.2. Illustration du calcul des motifs LBP.....	36
Fig.2.3. Trois voisinages de tailles différentes utilisées dans le calcul de LBP.....	37
Fig.2.4. Exemple de modèles uniformes et non uniformes, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP.....	38
Fig.2.5. Les 58 modèles uniformes, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al., 2002b].....	38
Fig.2.6. Les 36 modèles invariants par rotation définis pour P égale à 8, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al., 2002b].....	39
Fig.2.7. Les 9 modèles uniformes et invariants en rotation définis pour $P=8$. Les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al., 2002b].....	40
Fig.2.8. Deux exemples d'opérateur EBP [Liao et al, 2007].....	43
Fig.2.9. Un exemple de l'opérateur MB-LBP [Zhang et al, 2007].....	43
Fig.2.10. (a) voisinage 3×3 (b) différence local (c) signe de la différence (d) magnitude de la différence.....	44
Fig.2.11. Principe de LBP et CS-LBP pour un voisinage de 8 pixels [Heikkilä et al., 2009].....	45
Fig.2.12. Un exemple de la partition d'un code LTP en codes de LBP positifs et négatifs.....	47
Fig.2.13. Illustration du calcul du code (LQP) en utilisant quatre modèles binaires.....	48
Fig.3.1. Représentation de deux plateaux et leurs voisins.....	53
Fig.3.2. Images texturées codées, (a) images test, (b) codage d'extremalié régionale, (c) codage de rang régional.....	55
Fig.3.3. Images de références.....	57
Fig.3.4. Résultats de segmentation, (a) images test, (b) codage de rang régionale, (c) images segmentées.....	57
Fig.3.5. Exemples d'images texturées et leurs versions tournées (a)Rotation 0° (b) rotation 15° (c) rotation 75° (d) rotation 90°	59
Fig.3.6. La valeur absolue de la différence entre les histogrammes RRC de l'image originale $D1$ et ses versions tournées (a) 0° et 15° (b) 15° et 75° (c) 0° et 90° (d) 75° et 90°	60
Fig.3.7. Représentation d'un ensemble de pixels (en gris) et l'ensemble de ses voisins (en bleu) en utilisant (a) espace de recherche de taille 3×3 , (b) espace de recherche de taille 5×5	61
Fig.4.1. (a) pixel central x et ses 8 voisins, (b) Suite de voisinage utilisée pour le codage.....	68
Fig.4.2. Illustration du codage LDRC.....	69
Fig.4.3. Représentation d'un exemple exploitant 3 échelles et illustration de la transformation du voisinage $x_{s,8q}$ au voisinage $y_{s,q}$. Les différentes couleurs correspondent aux 4 voisinages directionnels utilisés par l'opérateur LDRC.....	71

Fig.4.4. Un exemple de calcul du codage LDRC sur trois échelles.....	71
Fig.4.5. Exemple d'images texturées (Brodarz D1, CURET #30, kthtip2#9) corrompus par un bruit gaussien de différentes intensités.	72
Fig.4.6. Taux de bonne classification en fonction du nombre d'échelles	73
Fig.4.7. Taux de classification obtenus sur la base de Brodatz bruitées en fonction du nombre d'échelles et pour différents SNR	75
Fig.4.8. Taux de bonne classification des textures de la base CURET en fonction des nombres d'échelles	76
Fig.4.9. Taux de classification obtenus sur la base de CURET bruitées en fonction du nombre d'échelles et pour différents SNR	78
Fig.4.10. Taux de bonne classification des textures de la base KTH-TIPSE2b en fonction des nombres d'échelles	79
Fig.4.11. Taux de classification sur la base KTH-TIPS2b bruitée en fonction du nombre d'échelles obtenus avec différents SNR	80
Fig.4.12. Deux exemples de calcul du codage DC	82

Liste des tableaux

Table 1.1. Propriétés des bases de données utilisées dans les tests	30
Table 2.1. Différentes variantes de LBP. Les méthodes utilisées dans nos tests sont indiquées en gras.	41
Table 3.1. Codage de rang régional en fonction de P_0 et les niveaux de gris.....	54
Table 3. 2. Taux de bonne classification des images segmentées	56
Table 3. 3. Le code RRC calculé en fonction du niveau de gris et le rapport P_0	58
Table3.4. L'erreur quadratique moyenne entre les caractéristiques extraites à partir du RRC correspondant à deux rotations de la même image de texture	60
Table 3. 5. Taux de classification (%) retrouvé sur les bases TC10 et TC12	64
Table 3. 6. Taux de classification (%) retrouvé sur la base CURET.	65
Table 4.1. Taux de classification retrouvés sur la base de Brodatz.....	74
Table 4. 2. Taux de classification obtenus sur la base de Brodatz bruitée en fonction de différents SNR	75
Table 4.3. Taux de classifications(%) retrouvées sur la base CURET.....	77
Table 4.4. Taux de classifications obtenues sur la base de CURET bruitées pour différents SNR.....	78
Table 4.5. Taux de classification(%) obtenus sur la base KTH-TIPS2	80
Table 4.6. Taux de classification en fonction du nombre d'échelles	83
Table 4.7. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur l'ensemble d'images TC10 et TC 12. Les deux plus hauts taux de classification sont donnés en gras	84
Table 4. 8. Performances de DC en fonction du nombre d'échelles, le taux de classification le plus élevé est indiqué en gras	84
Table 4. 9. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur la base CURET. Les deux taux les plus élevés de classification sont indiqués en gras	85
Table 4.10. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur la base KTH-TIPS2b. Les deux taux les plus élevés de classification sont indiqués en gras	86

Table des matières

Introduction	1
CHAPITRE 1: Méthodes d'analyse de la texture	
1.1. Préambule.....	5
1.2. Analyse de la texture.....	6
1.2.1. Notion de texture.....	6
1.2.2. Perception visuelle de la texture.....	9
1.2.3. Méthodes d'analyse de texture.....	10
1.2.3.1. Méthodes statistiques.....	11
1.2.3.2. Méthodes structurelles.....	13
1.2.3.3. Méthodes basées sur un modèle.....	14
1.2.3.4. Méthodes basées sur le traitement du signal.....	14
1.2.3.5. Comparaison des méthodes usuelles d'analyse de textures.....	17
1.3. Problème de classification des textures.....	19
1.4. Algorithmes de classification.....	25
1.4.1. La méthode des k plus proches voisins.....	26
1.4.2. Séparateurs à vaste marge.....	27
1.4.3. Etapes de classification des textures.....	29
1.5. Les bases des images utilisées : vue d'ensemble.....	30
1.5.1. Base de données de Brodatz.....	30
1.5.2. Base de données Outex.....	31
1.5.3. Base de données CURET.....	32
1.5.4. Base de données KTH_TIPS2b.....	33
1.6. Discussions.....	34
CHAPITRE 2: Les méthodes LBP	
2.1. Préambule.....	35
2.2. La méthode LBP de base.....	35
2.2.1. Modèle LBP uniforme.....	37
2.2.2. Modèle LBP invariant par rotation.....	38
2.2.3. Modèle LBP uniforme et invariants en rotation.....	39
2.3. Quelques variantes de LBP.....	41
2.3.1. Topologie du voisinage et échantillonnage.....	42
2.3.2. Seuillage.....	46
2.3.3. Codage et regroupement.....	49
2.3.4. Combinaison avec d'autres approches.....	50
2.4. Discussions.....	51

CHAPITRE 3 : Codage de Rang Régional

3.1. Préambule	52
3.2. Codage de rang régional	52
3.2.1. Principe	53
3.2.2. Conservation de l'information niveau de gris	55
3.2.3. Segmentation d'images texturées	56
3.3. Codage RRC	58
3.3.1. Invariance en rotation	59
3.3.2. Etude multi-résolution :	61
3.3.3. Tests et Résultats :	61
3.3.3.1. Résultats obtenus sur la base de données Outex :	63
3.3.3.2. Résultats obtenus sur la base de données CURET	64
3.4. Discussions	65

CHAPITRE 4 : Codage Directionnel

4.1. Préambule	67
4.2. Le codage de rang directionnel local	67
4.2.1. Principe du codage LDRC.....	68
4.2.2. Etude multi- résolution	70
4.2.3. Tests et Résultats	72
4.2.3.1. Résultats obtenus sur la base de Brodatz.....	73
4.2.3.2. Résultats obtenu sur la base CURET :	76
4.2.3.3. Résultats obtenus sur la base KTH-TIPS2b.....	78
4.3. Le codage directionnel	80
4.3.1. Tests et résultats.....	83
4.3.1.1. Résultats obtenus sur la base Outex	83
4.3.1.2. Résultats obtenus sur la base CURET.....	84
4.3.1.3. Résultats obtenus sur la base KTH-TIPS2b.....	85
4.3.1.4. Durée d'exécution	86
4.4. Discussions	86
Conclusion	88

Annexes

A. Séparateurs à vaste marge	90
A.1. Séparateurs à vaste marge	90
A.1.1. Classifieur linéaire	91
A.1.1.1. Marge maximale de l'hyperplan	91
A.1.1.2. Minimisation quadratique sous contraintes	94
A.1.2. SVM non-linéaires	96
A.1.3. Relaxation des contraintes	97
A.1.4. SVM multi classes	98
B. Bilan scientifique relatif à la thèse	99

Bibliographie	100
----------------------------	------------

INTRODUCTION

La texture est une composante visuelle riche en information. L'analyse de la texture vise à résoudre un large panel de problème en vision par ordinateur comme la classification, la segmentation, la modélisation et la synthèse de la texture [Tuceryan et Jain, 1998]. Les méthodes d'analyse de textures peuvent être scindées en quatre groupes : Les méthodes d'analyse basées sur l'analyse statistique des textures. La méthode de la matrice de cooccurrence est l'une des plus connue [Haralick et al., 1973]. Les méthodes d'analyse de texture qui utilisent les caractéristiques structurelles des images. Ces méthodes considèrent la texture comme un agencement d'éléments de base ou primitives qu'il faut extraire puis en déterminer la règle d'arrangement [Tuceryan et Jain 1998], les méthodes d'analyse qui définissent les textures par des modèles de probabilité comme le modèle Markovien [Cohen et al 1991, Yousefi et al 2011] ou le modèle autorégressif (AR) [Mao et al, 1992][Abadeni, 2010], les méthodes d'analyse des textures basées sur le filtrage d'image, les filtres de Gabor [Alvaro et al., 2014] et la transformée en ondelettes [Mallat, 1989] sont parmi les plus utilisées.

Contexte et problématique

La classification de la texture est largement étudiée dans la littérature. Elle a été utilisée dans de nombreuses applications incluant l'analyse d'images médicales [Nanni et al, 2010] l'inspection industrielle [Xie, 2008], la récupération d'images [Subrahmanyam, 2012] et la reconnaissance faciale [Tan et al., 2010].

En dépit des nombreux domaines d'application de la classification de la texture, les textures dans le monde réel ne sont pas souvent uniformes, en raison de changements d'orientation ou d'échelle, présence de bruit ou tout autre aspect visuel ce qui affecte les performances des méthodes d'analyse. Malgré les nombreux travaux existants consacrés à la description de texture, l'extraction de caractéristiques de textures invariantes à ces changements reste un problème important à résoudre. Cela a inspiré plusieurs études, qui intègrent généralement l'invariance par rapport à une ou au plus deux des propriétés [Zhang et al ,2002].

Les premières approches proposées pour la classification des textures invariantes en rotation comprennent les méthodes basées sur les matrices de cooccurrence généralisées [Davis et al, 1979], les polarogrammes [Davis, 1981] et l'anisotropie de texture [Chetverikov, 1982]. Des descripteurs de textures invariants à la rotation basés sur les filtres de Gabor [Fahriz et al, 2013] et les champs de Markov [Deng et al, 2004] sont aussi proposés.

Parmi les techniques les plus performantes d'analyse de textures invariantes en rotation, on en distingue principalement deux méthodes basées sur l'analyse locale des éléments de texture: le sac de mots ou BoW (en anglais *the Bag of Words*) [Leug et Malik, 2001] [Cula and Dana, 2004][Varma et Zisserman, 2005][Varma et Zisserman, 2009] et le modèle binaire local ou LBP (*Local Binary Pattern*) [kylberg et al, 2013] [Liu et al, 2014] [Hafiane et al, 2015] [Liu et al, 2017].

Varma et Zisserman [Vama et al, 2005] ont présenté une banque de filtres VZ_MR8. Une bibliothèque de textons invariants en rotation est d'abord construite à partir d'un ensemble d'apprentissage, une texture inconnue de l'image est classée en fonction de la distribution des textons. Plus tard, Varma et Zisserman [Varma et al, 2009] ont proposé la méthode VZ_Joint, qui extrait des textons à partir de l'intensité locale de l'image. Ces deux méthodes sont connues pour leurs bonnes performances de classification obtenues sur les bases de données volumineuses et complexes [Vama et al, 2005, Varma et al, 2009].

Ojala et al. [Ojala et al., 2002b] proposent le modèle binaire local (LBP) pour la classification de texture invariante en rotation. La méthode LBP est basée sur le signe de la différence entre le pixel central et ses voisins. Une fois le code LBP de chaque pixel de l'image est défini, un histogramme sera construit pour représenter l'image de la texture. La méthode LBP est un opérateur simple mais efficace pour décrire le motif local de l'image, des résultats de classification impressionnants sont obtenus sur des bases de données de texture représentatives [Ojala et al, 2002a]. En raison du grand succès de LBP en vision par ordinateur et la reconnaissance des formes, de nombreuses variantes sont proposées [Qi et al, 2014][Wang et al, 2015][Pan et al, 2015][Liu et al, 2016][Liu et al, 2017] afin d'améliorer le pouvoir discriminant mais aussi la robustesse au bruit.

Bien que des progrès significatifs soient réalisés, la plupart des variantes LBP ont des inconvénients essentiellement : Sensibilité au bruit [Liao et al, 2009][Khellah,2011][Liu et al, 2014], limitation des méthodes LBP à trois échelles, ce qui les empêche de capturer des informations de texture à longue portée [Liao et al, 2009][Khellah,2011][Hafiane et al, 2015].

Aussi, parfois les méthodes LBP classent différents motifs dans un même modèle, ce qui peut réduire leur capacité discriminative [Liao et al, 2009][Zhao et al,2013].

Ce travail s'inscrit dans le cadre du problème mentionné. Inspiré par les méthodes LBP, nous proposons dans ce travail deux méthodes d'analyse de texture. L'idée est d'exploiter des extensions de la notion d'extrema de gris afin d'extraire des descripteurs de texture compacts et discriminatifs.

Tout d'abord, le codage de rang régional ou RRC (*Regional Rank Coding*) est proposé. Cette méthode attribue à chaque pixel une valeur en fonction du rang de son niveau de gris sur un voisinage régional. Huit descripteurs invariants en rotation sont alors extraits. Pour l'étude multi-résolution, différentes dimensions de l'espace de recherche sont exploitées.

Ensuite, dans la seconde partie, nous présentons le codage local de rang directionnel (LDRC) et le codage directionnel (DC). Ces deux méthodes sont basées sur le signe de la différence entre le pixel central et ses voisins. Pour l'étude multi-résolution, les deux codages sont calculés en modifiant la taille de la fenêtre d'analyse autour du pixel traité, le nombre de voisins est limité à huit dont chacun représente une moyenne locale des niveaux de gris. La taille de l'histogramme des paramètres est ainsi maintenue réduite et constante. L'intérêt de ces deux méthodes est d'extraire des descripteurs de texture invariants en rotation et robuste au bruit.

Contributions

Pour évaluer les méthodes proposées nous avons utilisé une large gamme de bases de données : Brodatz [Brodatz,1965], Outex [Ojala et al, 2002a], CURET [Dana et al,1999],et KTH-TIPS2b[Mallikarjuna et al,2006]. Nous avons étudié différents problèmes liés à la reconnaissance de texture tels que l'effet de l'éclairage, la rotation et le changement d'échelle, la présence de bruit et aussi l'influence du nombre d'images d'apprentissage.

Les principaux avantages des méthodes proposées sont :

- simplicité de mise en œuvre,
- schéma de calcul rapide,
- faible dimension des paramètres,
- paramètres de textures très discriminatifs,
- invariance à l'éclairage, à la rotation et à l'échelle,
- robustesse au bruit.
- les méthodes proposées permettent de coder de manière compacte quelque soit l'échelle d'analyse ou la taille de la fenêtre de voisinage considérée.

Structure du mémoire

Le travail que nous présentons dans cette thèse est structuré en quatre chapitres : Dans le premier chapitre, les différentes méthodes d'analyse de textures sont présentées. Ensuite, les méthodes invariantes en rotation sont abordées. Enfin, les bases de données utilisées ainsi que les algorithmes de classifications exploités sont présentés.

Le deuxième chapitre est consacré à la présentation de la méthode LBP et plusieurs de ses variantes essentiellement celles auxquelles les méthodes proposées sont comparées.

Dans le troisième chapitre, nous présentons le codage de rang régional. Cette méthode est proposée pour extraire des paramètres invariants en rotation. Des tests sur les bases Outex et CURet sont effectués pour évaluer son efficacité par rapport à la méthode LBP.

Dans le dernier chapitre, nous présentons le codage local de rang directionnel. Cette méthode est implémentée pour la classification des textures en présence de bruit gaussien. Aussi, une extension de cette méthode est proposée pour l'invariance en rotation. Les différents tests effectués sont présentés.

En conclusion, nous proposons une synthèse concernant nos différentes contributions et nous discutons sur des perspectives qui pourraient apporter des améliorations à ces contributions et/ou proposer d'autres voies de continuité de ce travail.

CHAPITRE 1

Méthodes d'analyse de la texture

1.1. Préambule

La texture est une composante visuelle riche en information. Un objectif fondamental des travaux concernant l'analyse de texture est de formaliser le descriptif de la texture par des paramètres mathématiques qui serviraient à l'identifier.

La classification des textures est l'un des problèmes majeurs dans l'analyse de texture. Elle a reçu une attention considérable au cours des dernières décennies en raison de son importance dans la vision par ordinateur. La classification de texture est très utile dans de nombreuses applications, telles que la télédétection, l'analyse d'image biomédicale, la reconnaissance faciale, l'inspection industrielle, la classification de documents et la recherche d'images par le contenu.

Généralement, les images de textures capturées dans le monde réel ne sont pas souvent uniformes, en raison des changements d'orientation ou d'échelle, présence de bruit ou tout autre aspect visuel ce qui affecte les performances des méthodes d'analyse. Malgré les nombreux travaux consacrés à la description de texture, l'extraction de caractéristiques de textures invariantes à ces changements reste un problème important à résoudre.

Ainsi, dans une première partie de ce chapitre, un état de l'art sur les différentes méthodes d'analyse de texture est proposé, les travaux portant sur les méthodes invariantes à l'éclairage, la rotation et à l'échelle sont ensuite abordées. Dans une seconde partie, les méthodes de classification exploitées ainsi que les bases d'images texturées utilisées dans ce travail sont présentées.

1.2. Analyse de la texture

1.2.1. Notion de texture

La notion de texture est utilisée pour traduire un aspect homogène de la surface d'un objet sur une image. La texture se manifeste par une perception visuelle qui peut être décrite qualitativement selon des critères tels que la régularité, l'orientation, la granularité, la rugosité, ect... Les images de la figure 1.1 issues de l'album de Brodatz [Brodatz, 1965] illustrent quelques exemples.

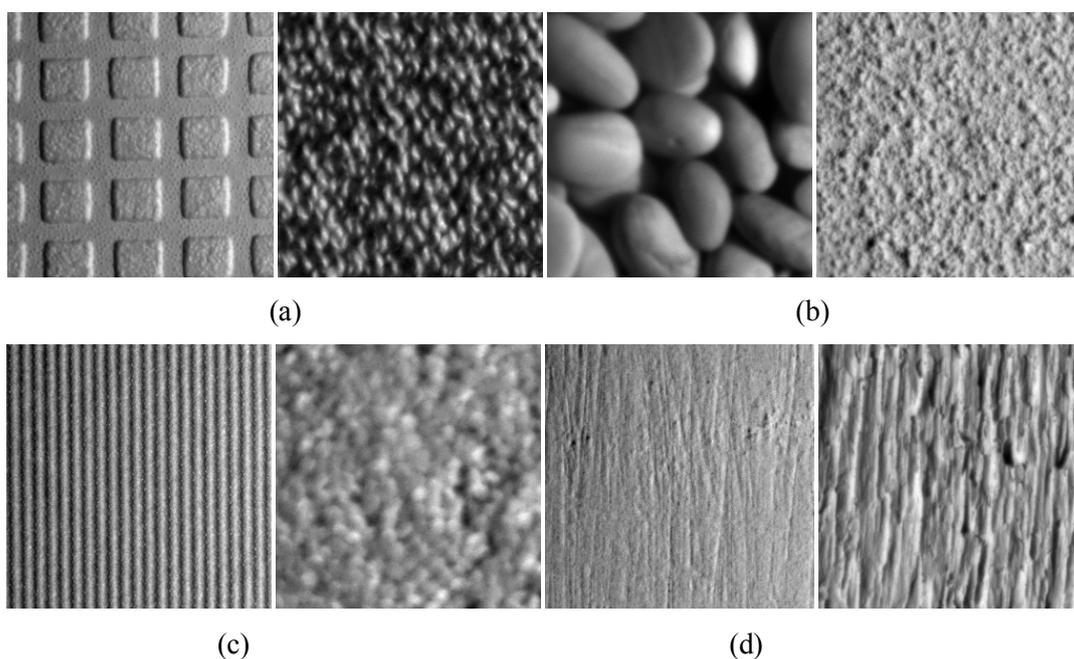


Fig.1.1. Perception des textures (a) Texture régulière et texture irrégulière (b) texture grossière et texture fine (c) texture directionnelle et texture non directionnelle (d) texture lisse et texture rigoureuse

Il est difficile de donner une définition précise et consensuelle à la notion de texture. Cette difficulté est démontrée par le nombre de définitions proposées dans la littérature.

Selon Julesz [Julesz, 1962], la texture est un ensemble d'images de mêmes valeurs statistiques d'ordre un et deux. Cette conjoncture a été réfutée par Julesz lui-même dans [Julesz et al., 1978] où les auteurs ont synthétisé des textures facilement distinguables ayant les mêmes statistiques d'ordre deux. Dans [Sklansky, 1978], Sklansky estime que la texture est une région de l'image dont l'ensemble des statistiques locales sont constantes, varient peu ou sont approximativement périodiques. Quant à Haralick [Haralick, 1979], il décrit l'image texturée par le nombre et le type de ses primitives ainsi que leur organisation spatiale ou agencement. Dans [Cross et Jain, 1983], Cross et Jain proposent de définir la texture par un champ

bidimensionnel stochastique, éventuellement périodique. Selon Unser [Unser, 1984], une texture est une région de l'image pour laquelle il est possible de définir une fenêtre de dimensions minimales, telle qu'une observation au travers de celle-ci se traduit par une perception (impression) visuelle identique pour toutes les translations possibles de cette fenêtre à l'intérieur de la région considérée.

Dans la littérature, deux grandes familles de texture s'imposent, les textures déterministes et les textures stochastiques.

Une texture est dite déterministe si elle est formée par un agencement régulier d'un motif géométrique. Ce type de texture est entièrement caractérisé par un motif et sa règle déterministe d'agencement. La figure 1.2 représente quelques exemples de textures déterministes.

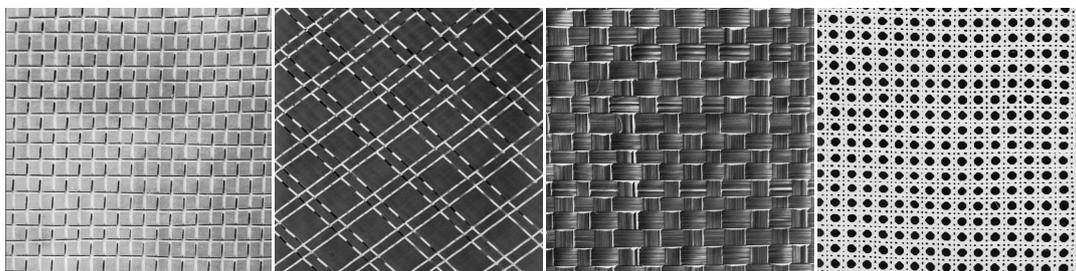


Fig.1.2. Exemples de textures déterministes tirées de l'album de Brodatz [Brodatz,1965].

Les textures dites stochastiques se caractérisent par un contenu irrégulier dont il est parfois difficile d'en extraire une primitive de base (motif non localisable). Elles sont décrites par des lois ou des modèles statistiques et sont considérées comme une réalisation d'un champ aléatoire bidimensionnel homogène. Des exemples de textures stochastiques sont présentés sur la figure 1.3.



Fig1.3. Exemples de textures stochastiques tirées de l'album de Brodatz [Brodatz,1965].

Hormis ces deux catégories de textures, il en existe d'autres comme, par exemple, les textures directionnelles (Figure 1.4). Ces textures ne sont pas totalement aléatoires et ne présentent pas d'éléments structurants de base. Néanmoins, elles se caractérisent par certaines orientations.



Fig1.4. Exemples de textures directionnelles tirées de l'album de Brodatz [Brodatz,1965].

Il est parfois difficile de classer une texture dans l'une ou l'autre des catégories d'autant plus que la résolution de l'image pose un problème supplémentaire, la figure 1.5 représente deux images tirées de la base KTH-TIPS2b [Mallikarjuna et al., 2006], une toile de lin est photographiée à deux résolutions différentes. La première image laisse à penser que la toile de lin est une texture déterministe, la seconde, qu'elle est plutôt stochastique.

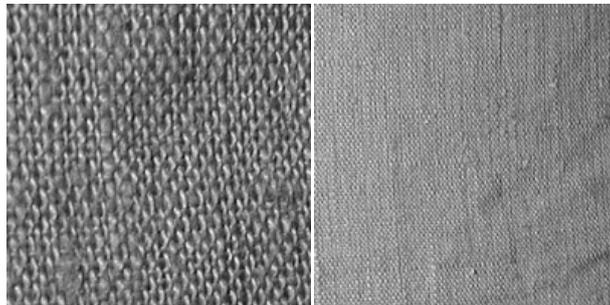


Fig. 1.5. Exemple d'une même texture photographiée à 2 résolutions différentes.

Une des applications de la notion de texture dans une image est la classification (Fig.1.6(b)). Le but de la classification est d'assigner chaque région uniformément texturée à la classe de texture à laquelle elle appartient suivant un certain critère de similitude. La segmentation de texture est le second type d'application en analyse de texture (Fig.1.6(c)). Le but est alors de découper une image en régions homogènes uniformément texturées ou possédant des propriétés proches ou identiques en faisant ressortir les frontières entre les classes.

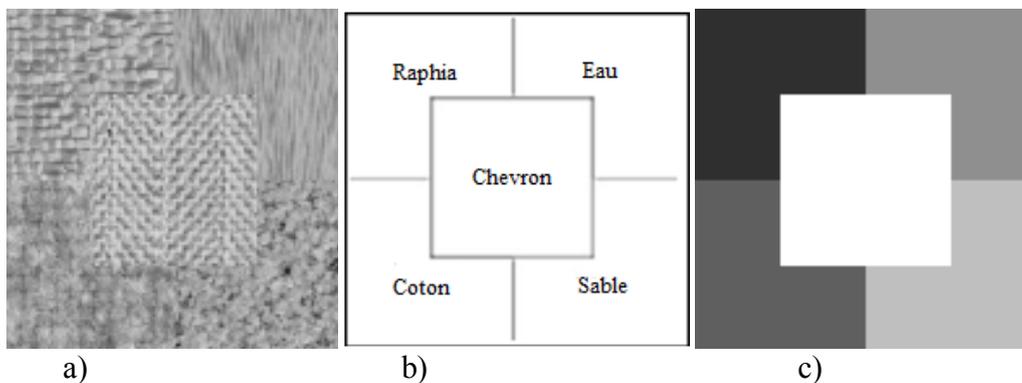


Fig. 1.6. a) Image composée de cinq régions texturées : "raphia", "eau", "coton", "sable", et "chevron"
 b) classification c) segmentation.

Un troisième aspect lié à la notion de texture est la modélisation de la texture. Elle consiste à déterminer les modèles de distribution propres à chaque texture. Ces modèles peuvent être aussi bien employés pour la synthèse que pour l'identification de textures.

Identifier les caractéristiques perceptibles de la texture dans une image constitue la première et la principale étape de l'analyse de texture. Il existe diverses approches d'analyse. Elles ont pour but de formaliser le descriptif de la texture par des paramètres mathématiques qui serviraient à l'identifier.

1.2.2. Perception visuelle de la texture

Beaucoup de travaux ont été menés pour étudier la perception de la texture par le système visuel humain (SVH). En vision par ordinateur, le SVH est simulé via la proposition de représentations mathématiques permettant un traitement informatisé dans des applications comme la classification, la segmentation ou la synthèse des textures.

Dans [Julesz, 1962], Julesz propose de définir mathématiquement une texture. Dans un premier temps, Julesz affirme que deux textures sont difficilement distinguables si leurs statistiques de second ordre sont identiques. Ceci est démontré par l'exemple de la figure 1.7(a). Cette image se compose d'une paire de régions texturées dont les statistiques de second ordre sont identiques. Les deux régions texturées sont difficilement différenciées. Plus tard, cette conjoncture a été réfutée par Julesz lui-même [Julesz et al., 1978] en synthétisant des textures ayant les mêmes statistiques de second ordre mais facilement différenciées comme le montre l'exemple de la figure 1.7 (b). Pour expliquer cette discrimination pré attentive de paires de texture, Julesz propose la théorie des « textons » [Julesz , 1981a][Julesz , 1981b]. Les textons sont des primitives de base liées aux caractéristiques locales de la texture qui sont décrites par des terminaisons, longueurs, largeurs et orientation des taches. Ces différentes caractéristiques sont détectées et utilisées pour la discrimination des textures. En utilisant la théorie de textons, Julesz explique les exemples de la figure 1.7 comme suit :

Sur la figure 1.7 (a), le nombre de terminaisons dans les deux régions est le même (c-à-d l'information de textons dans les deux régions n'est pas différente), donc le système visuel est incapable de distinguer pré attentivement les deux textures. D'autre part, sur la figure 1.7(b), le nombre de terminaisons dans la moitié supérieure est de trois, tandis que le nombre de terminaisons dans la moitié inférieure est de quatre. La différence dans ces textons rend les deux régions texturées discriminables.

Les approches suggérant que le VHS distingue les textures en employant des statistiques globales, ne sont pas les seules à avoir été proposées dans la littérature pour la perception des

textures. D'après des études psychophysiques, le cerveau humain effectue une analyse de fréquence de l'image [Harvey et Gervais, 1978][De Valois et al., 1982] et[Tuceryan et Jain , 1990], ces travaux ont motivé les chercheurs en vision par ordinateur à appliquer les filtres multi-échelles et multi- orientations pour l'analyse de la texture.

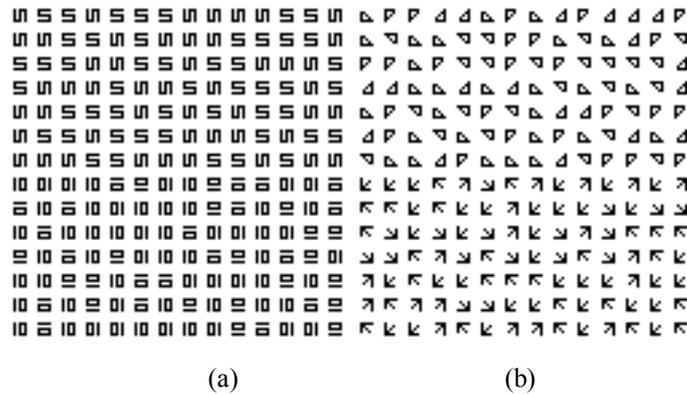


Fig.1.7. Deux images composées de paires de texture de même statistiques de second-ordre : (a) L'œil humain ne peut pas apercevoir les deux régions sans un examen minutieux, (b) les deux régions sont immédiatement discriminées par l'œil humain. [Tuceryan and Jain, 1998].

L'étude de la perception visuelle de la texture a conduit au développement de différentes théories pour l'interprétation de la vision humaine. À partir de ces différentes théories, plusieurs méthodes visant l'analyse de la texture sont développées. La conjecture de Julesz a été à l'origine des méthodes statistiques pour l'analyse des textures en les caractérisant à l'aide des descripteurs statistiques. La théorie des textons a inspiré les approches géométriques qui se proposent d'extraire des descripteurs pour la caractérisation de la texture. L'assimilation de la perception aux transformées multi-échelles et multi-orientations a permis le développement des méthodes de filtrage pour l'analyse des textures.

1.2.3. Méthodes d'analyse de texture

Selon Tuceryan et Jain [Tuceryan et Jain, 1998], les différentes méthodes d'analyse de texture peuvent se répartir en quatre catégories: méthodes statistiques, méthodes structurelles (géométriques), méthodes basées sur le traitement de signal et les méthodes basées sur des modèles.

1.2.3.1. Méthodes statistiques

Les méthodes statistiques sont basées sur les stochastiques de la distribution spatiale des niveaux de gris dans une image. Elles sont généralement classées en statistiques de premier ordre, en statistiques de second ordre et en statistiques d'ordre supérieur.

Les statistiques du premier ordre mesurent la probabilité d'apparition d'un niveau de gris à un emplacement aléatoire sans prendre en considération la dépendance spatiale entre pixels. Ces statistiques sont calculées à partir de la distribution de probabilité (histogramme) des niveaux de gris [Pinamonti et al. 1989][Schistad Solberg et Jain, 1997]. Les matrices de co-occurrence ou GLCM (*Grey Level Co-occurrence Matrix*) introduites par Haralick [Haralick et al., 1973] caractérisent les dépendances spatiales entre deux niveaux de gris. Le principe de cette matrice consiste à repérer dans une image le nombre d'occurrences de paires de niveaux de gris séparés par une distance d dans une direction définie par un vecteur de déplacement (dx, dy) . Le calcul de la GLCM pour une image I de dimension $N \times M$ se formalise comme suit :

$$GLCM_{dxdy}(i, j) = \sum_{n=1}^N \sum_{m=1}^M \begin{cases} 1, & \text{si } I(n, m) = i \text{ et } I(n + dx, m + dy) = j \\ 0, & \text{sinon} \end{cases} \quad (1.1)$$

où i et j sont respectivement les niveaux de gris du pixel de référence et du pixel voisin, n et m correspondent aux coordonnées des pixels dans l'image I .

La GLCM est souvent calculée sur quatre orientations différentes (0° , 45° , 90° , 135°) pour la même distance de déplacement d . La figure 1.8 illustre le processus de génération de quatre matrices de cooccurrence sur une image de taille 4×4 de 4 valeurs de niveaux de gris. Nous considérons $d=1$ et les quatre orientations (0° , 45° , 90° , 135°).

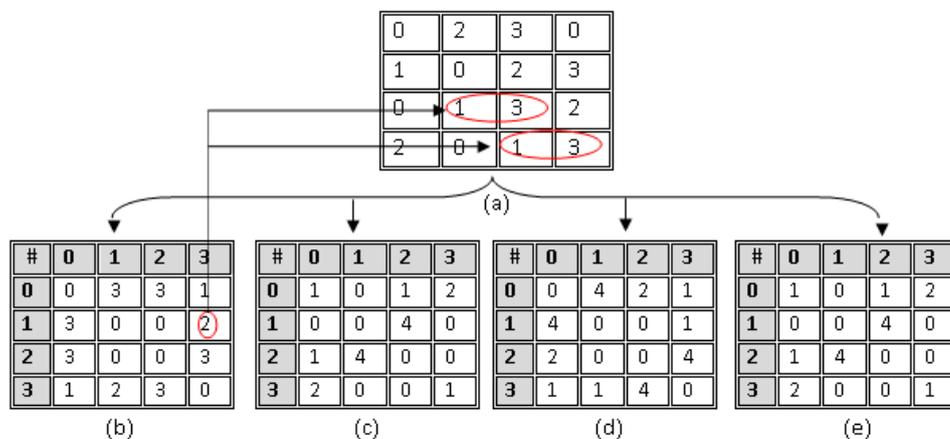


Fig.1.8. Calcul de Matrice de cooccurrence sur l'image test (a) utilisant quatre directions (b) 0° (c) 45° (d) 90° et (e) 135° avec la distance $d=1$. # représente le nombre d'occurrences.

Les matrices de cooccurrence contiennent beaucoup d'éléments (elles sont de taille 256×256 éléments pour une image sur 256 niveaux de gris) et sont donc difficilement manipulables. Dans la pratique, les images sont souvent altérées afin de ne considérer qu'une partie des niveaux de gris. Plusieurs mesures statistiques sont ensuite extraites [Haralick, 1979]. Les matrices de cooccurrence sont parmi les méthodes statistiques les plus couramment utilisées et ont inspiré une variété de méthodes entre autres [Roberti de Siqueira et al, 2013] [loris et al, 2015].

Les longueurs de plage de niveau de gris [Galloway, 1975] estiment la probabilité d'observer des plages à niveau de gris constant dans une direction donnée. Une plage est un ensemble de pixels consécutifs ayant le même niveau de gris. Cette méthode consiste donc, au premier lieu, à déterminer l'ensemble de plages présentes dans l'image, et en second lieu, d'extraire des statistiques sur leur longueur. Plusieurs matrices correspondant chacune à une orientation θ des plages dans l'image sont définies, les orientations couramment utilisées pour décrire les textures sont : 0° , 45° , 90° et 135° , des paramètres sont ensuite calculés à partir de ces matrices pour caractériser la texture. Tout comme pour les matrices de cooccurrences, les matrices de longueurs de plage permettent une bonne caractérisation des textures. D'autres méthodes statistiques sont aussi proposées telles que l'utilisation des histogrammes de la somme et la différence des niveaux de gris [Ameur Z et al., 2004], la différence de niveaux de gris [Weszka et al, 1976] ou la fonction d'autocorrélation [Bounekkar, 1996]. [Bi, 1997] propose d'utiliser la notion de vecteur de rang pour le codage de texture. L'intérêt de cette méthode est qu'elle prend très bien en compte l'information de distribution spatiale du niveau de gris. Cette méthode est aussi exploitée dans [Ameur Z, 2005] pour la caractérisation et la segmentation d'images texturées.

Aussi, la notion d'extrema de gris est énormément utilisée car elle est liée à beaucoup de méthodes de caractérisation de textures, mais aussi à beaucoup de méthodes générales d'analyse d'images. En effet, certains filtres sont basés sur la recherche de ces extrema pour ensuite en calculer la moyenne ou la médiane. Certaines opérations de la morphologie mathématique à niveaux de gris consistent à rechercher les extrema de gris autour des pixels de l'image [Peng et al, 1992]. Les extrema sont aussi très utilisés lors de la détection des contours. Bonnevey [Bonnevey, 1997] introduit la notion d'extrema afin de caractériser, puis de classifier des textures. Les résultats obtenus montrent qu'une texture peut être raisonnablement comparée à d'autres simplement à l'aide de la notion d'extrema de gris et met donc en avant leur importance en matière d'analyse de textures. Mavromatis [Mavromatis et al, 2004] introduit le concept d'extremum d'ordre maximum multiple pour la

caractérisation directionnelle d'une texture. Plus récemment, Subrahmanyam et al [Subrahmanyam, 2012] proposent de nouveaux motifs pour caractériser des régions locales de l'image. Cette méthode extrait des informations sur la base de la distribution des maximums locaux dans une image qui sont évalués en tenant compte de la magnitude de la différence locale entre le pixel central et ses voisins. [Vipparthi, 2014] propose un nouveau descripteur capable d'extraire quatre informations directionnelles. Les informations directionnelles sont ensuite combinées. Dans cette thèse, nous allons aussi exploiter la notion d'extrema de gris afin de caractériser la texture.

Dans [Ojala et al., 1996], les motifs binaires locaux ou LBP (en anglais *Local Binary Pattern*) sont proposés afin de décrire les textures présentes dans des images à niveaux de gris. Le LBP consiste à attribuer à chaque pixel de l'image une valeur caractérisant le motif local autour de ce pixel. Ces valeurs sont calculées en comparant le niveau de gris du pixel central aux valeurs des niveaux de gris des pixels voisins. Le concept du LBP est simple, il propose d'assigner un code binaire à un pixel en fonction de son voisinage. Ce code décrivant la texture locale d'une région est calculé par seuillage d'un voisinage avec le niveau de gris du pixel central. Tout comme la matrice de cooccurrences, le LBP est parmi les méthodes les plus utilisées pour l'analyse de texture [Liu et al 2017].

1.2.3.2. Méthodes structurelles

Ces méthodes considèrent la texture comme un agencement d'éléments de base ou primitives qu'il faut extraire puis en déterminer la règle d'arrangement. Les primitives ainsi que leur arrangement spatial sont utilisés pour caractériser les textures. Les primitives peuvent être extraites par filtrage de l'image sur différentes échelles [Voorhees et Poggio 1987] [Tuceryan et Jain, 1990], par extraction de région adaptatif [Tomita et Tsuji, 1990], ou par morphologie mathématique [Serra, 1982][Chen, 1994]. Une fois que les primitives sont identifiées, l'analyse est simplifiée, soit par calcul des statistiques sur les primitives (par exemple, l'intensité, l'allongement, et l'orientation) ou en définissant la règle de placement des éléments [Zucker, 1976][Fu,1982]. La structure et l'organisation des primitives peuvent également être présentées en utilisant des mosaïques de Voronoï [Ahuja 1982][Tuceryan et Jain 1990]. En général, les méthodes structurelles sont mieux adaptées pour les textures avec de larges structures et ne fonctionnent pas bien sur les textures stochastiques [Zhang et al ,2002].

1.2.3.3. Méthodes basées sur un modèle

Ces approches caractérisent statistiquement la texture à l'aide de modèles probabilistes. La texture est considérée comme un processus aléatoire qui peut être décrit par une loi de probabilité. Les modèles Markoviens sont très utilisés pour l'analyse des textures. Leur emploi permet la caractérisation de la dépendance entre pixels en définissant une structure de voisinage et des fonctions de potentiel associées [Cohen et al 1991][Yousefi et al 2011]. Le modèle autorégressif (AR) caractérise les interactions locales entre pixels en considérant chaque pixel central dans un voisinage comme une somme pondérée des voisins et d'un bruit additif indépendant et identiquement distribué [Mao et al, 1992][Abadeni, 2010]. Mandelbrot [Mandelbrot, 1983] propose de décrire les images en utilisant les fractales, un ensemble de fonctions auto-similaires caractérisées par ce qu'on appelle la dimension fractale [Pentland, 1984].

1.2.3.4. Méthodes basées sur le traitement du signal

D'après des études psychophysiques, le cerveau humain effectue une analyse de fréquence de l'image [Tuceryan et Jain, 1990]. La texture est particulièrement adaptée à ce type d'analyse en raison de ses propriétés. Diverses techniques d'analyse de texture se fondent sur des techniques de traitement du signal. Les approches de traitement du signal couvrent des méthodes de filtrage dans le domaine spatial ou fréquentiel.

La transformée de fourrier discrète bidimensionnelle est très utilisée pour classifier les textures. La transformée de Fourier F permet de convertir une image I de dimension $N \times M$ depuis son domaine spatial vers son domaine fréquentiel selon la formule suivante :

$$F(u, v) = \sum_{k=1}^N \sum_{l=1}^M I(k, l) \exp(-j2\pi(\frac{k}{N}u + \frac{l}{M}v)) \quad (1.2)$$

Les deux variables u et v désignent respectivement les fréquences spatiales de l'image suivant les directions horizontales et verticales.

Le résultat de la transformée de fourrier est utilisé sous la forme d'un spectre d'amplitude. Chaque coefficient de ce spectre correspond à une fréquence observée dans une orientation donnée. Plus on s'éloigne du centre du spectre, plus la fréquence observée est élevée. Ainsi, une texture lisse affichera des valeurs élevées autour du centre du spectre (basse fréquence) alors qu'une texture rugueuse affichera des valeurs plus étalées dans le spectre (haute

fréquence). Il est possible d'extraire des attributs de texture à l'aide de la transformée de Fourier comme par exemple l'énergie calculée dans une couronne ou bien en fonction de certaines directions. Le domaine des fréquences est alors divisé en anneaux ou en secteurs angulaires (voir Fig. 1.9) et l'énergie calculée dans ces régions définit alors une caractérisation de la texture.

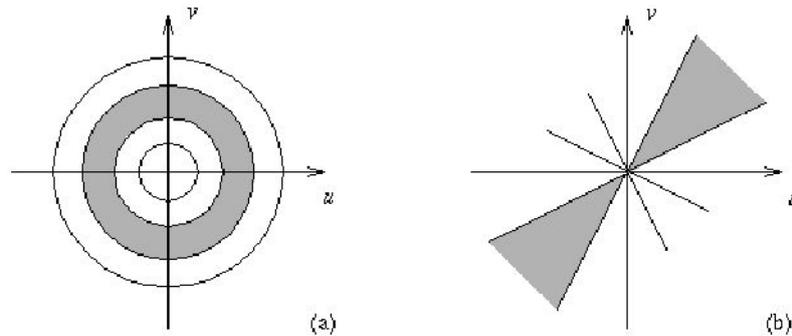


Fig.1.9. Partitionnement du spectre de puissance, (a) l'énergie calculée en chaque couronne fournit une information sur la finesse ou la grossièreté de la texture, (b) L'énergie calculée en chaque secteur angulaire fournit une information sur l'orientation de la texture

Une des principales limites de ce type d'approche est que le spectre de Fourier contient des informations localisées en fréquence et non dans l'espace. L'analyse du spectre ne permet donc pas de déduire directement la localisation spatiale dans l'image des textures dont la réponse fréquentielle est observée dans le spectre. Cette contrainte a mené à l'utilisation d'autres techniques telles que les filtres de Gabor et les transformées en ondelettes fournissant à la fois une localisation fréquentielle et spatiale dans l'image.

Les filtres de Gabor ont la capacité de modéliser la sensibilité fréquentielle et directionnelle du fonctionnement du cortex visuel humain en décomposant le spectre de l'image en images filtrées selon des plages étroites de fréquences et d'orientations. Un filtre de Gabor est une fonction Gaussienne modulée par une sinusoïde complexe. Dans le domaine spatial, un filtre de Gabor $h(x, y)$ se définit comme suit :

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{(x^2+y^2)}{2\sigma^2}\right] \exp(j2\pi F(x\cos\theta + y\sin\theta)) \quad (1.3)$$

Où σ^2 détermine l'étendue spatiale du filtre. Dans le domaine fréquentiel, après la transformée de Fourier de $h(x, y)$, le filtre de Gabor est une surface Gaussienne centrée sur la fréquence F et l'orientation θ :

$$H(u, v) = \exp [-2\pi^2\sigma^2((u - F\cos\theta)^2 + (v - F\sin\theta)^2)] \quad (1.4)$$

Les paramètres qui définissent chacun des filtres sont donc la fréquence F et l'orientation θ autour desquelles le filtre est centré dans le domaine fréquentiel et l'écart-type σ de la Gaussienne.

L'utilisation des filtres de Gabor en analyse de texture consiste à exploiter un banc de filtres, chacun centré sur une fréquence et une orientation distinctes en couvrant au mieux l'ensemble du domaine fréquentiel. Chaque pixel de l'image délivrera ainsi une réponse pour chaque filtre. L'aspect multi-échelles de cette représentation est particulièrement intéressant lorsque les structures que l'on cherche à identifier dans l'image sont caractérisées par des objets ou des motifs de taille variable. Il est en effet délicat dans ce cas de définir *a priori* une seule échelle d'analyse optimale. De même, la configuration multi-orientations trouve tout son sens dans le cas d'analyse de textures anisotropes et permet de conserver une information de directionnalité dans la signature texturale.

Plusieurs approches de classification de texture utilisant des bancs de filtres de Gabor sont proposées [Turner, 1986][Jain et Farrokhnia, 1991][Arivazhagan et al., 2006][Alvaro et al., 2014].

Tout comme les filtres de Gabor, les ondelettes permettent une bonne approximation du système visuel. L'utilisation de la transformée en ondelettes pour l'analyse de la texture a été initialement proposée par Mallat [Mallat, 1989]. Il prétend qu'elles permettent une analyse plus proche du système psycho-visuel par rapport à celle obtenue avec le filtre de Gabor.

La transformée en ondelettes décompose l'image en faisant appel à une série de fonctions créées par translation et changement d'échelle d'une fonction d'origine ψ , appelée ondelette-mère :

$$\psi_{s,u} = \frac{1}{\sqrt{s}} \psi\left(\frac{x-u}{s}\right) \quad (1.5)$$

où s et u représentent respectivement le facteur d'échelle et le facteur de translation. La décomposition en ondelettes d'une image $f(x,y)$ est alors le produit de convolution entre l'image $f(x,y)$ et les fonctions d'ondelettes $\psi_{s,u}$:

$$c_{i,j} = \int_{-\infty}^{+\infty} f(x,y) \frac{1}{\sqrt{s}} \psi\left(\frac{x-u}{s}, \frac{y-u}{s}\right) dx dy \quad (1.6)$$

Cette décomposition permet de calculer les coefficients d'ondelette $c_{i,j}$ et autorise également la reconstruction de l'image $f(x,y)$ à partir des coefficients $c_{i,j}$.

En pratique, l'utilisation de la décomposition en ondelettes sur une image discrète à deux dimensions revient à appliquer le produit de filtres passe-haut (H) et passe-bas (L) à une dimension. Dans l'analyse multi-résolutions définie par [Mallat, 1989], une approche de décomposition appelée transformée en ondelettes discrète ou *Discret Wavelet Transform* (DWT) est proposée. Celle-ci consiste à décomposer l'image en quatre sous-bandes sous-échantillonnées d'un facteur 2 à chaque échelle de décomposition. Ces sous-bandes sont le résultat de combinaisons entre filtre passe-haut et filtre passe bas: LL, LH, HL, HH. La sous-bande LL ou sous-bande d'approximation est une version moyennée de l'image d'origine alors que les sous-bandes HL, LH et HH ou sous-bandes de détails contiennent les hautes fréquences de l'image respectivement dans la direction de x (horizontale), dans la direction de y (verticale) ou dans les deux directions x et y (diagonale). Généralement, l'essentiel de l'information texturale étant lissée par l'application du filtre passe-bas, seules les sous-bandes de détails HL, LH et HH sont exploitées dans l'analyse texturale.

La DWT proposée par [Mallat, 1989] est une méthode standard d'application des ondelettes en analyse de textures. De nouvelles formes de décomposition inspirées de la DWT sont proposées dans la littérature. Citons par exemple la décomposition par paquets d'ondelettes créant une extension arborescente de la DWT [Chang et al, 1993].

1.2.3.5. Comparaison des méthodes usuelles d'analyse de textures

Différentes études comparatives des méthodes d'analyse de texture sont proposées dans la littérature. Cependant, il est difficile de proposer une hiérarchisation des approches car les comparaisons dépendent de l'application considérée et des données traitées.

Weska et al. [Weska, 1976] ont étudié et comparé les caractéristiques extraites du spectre de Fourier, des matrices de cooccurrence de niveaux de gris, des différences de niveaux de gris et des iso-segments de niveaux de gris en les intégrant dans une procédure de classification d'images aériennes et satellitaires. Les caractéristiques basées sur les matrices de cooccurrence de niveaux de gris et celles extraites des différences de niveaux de gris offrent des performances comparables. Ohanian et Dubes [Ohanian, 1992] ont comparé plusieurs paramètres de texture dont six extraits des matrices de cooccurrence, quatre dérivés des champs aléatoires de Markov, quatre caractéristiques fractales et d'autres issus des filtres de Gabor. L'évaluation, par une classification du k plus proches voisins (k -NN), a été faite sur quatre types d'images. Les matrices de cooccurrence ont donné le plus grand taux de bonne classification. Augustejin [Augustejin, 1995] a comparé les capacités de plusieurs méthodes

en utilisant une classification basée sur les réseaux de neurones. Les mesures texturales testées sont les matrices de cooccurrence, les différences de niveaux de gris, la transformée de Fourier, les filtres de Gabor et des statistiques locales. Il en a conclu qu'il n'y avait pas de meilleure méthode et que les performances de chacune dépendaient de l'application choisie. Dans [Manjunath et al, 1991], les auteurs comparent trois méthodes : l'énergie des sorties de filtres de Gabor, l'énergie des sous-bandes d'ondelettes orthogonales et le modèle autorégressif multi résolutions. Les meilleurs résultats en classification ont été obtenus avec le descripteur d'énergies des filtres de Gabor. Les auteurs dans [Ojala et al., 1996], étudient et comparent le modèle LBP, les matrices des différences de niveaux de gris, les descripteurs de filtres de Laws et les coefficients d'auto corrélation. Le modèle LBP présente de très bonnes performances pour une faible complexité par rapport aux autres méthodes. Dans sa thèse, Sharma [Sharma, 2001] s'est intéressée à la reconnaissance des objets dans des images représentant des scènes naturelles. Elle a étudié et comparé cinq approches texturales (fonction d'auto corrélation, matrice de cooccurrence, paramètres de Laws, densité de contours, iso-segments), elle en a conclu qu'il n'y avait pas de "meilleure approche" et que le résultat dépendait simultanément des paramètres employés et de l'approche de segmentation utilisée. [Stachowiak et al., 2005] compare les performances des méthodes d'analyse de textures en employant les ondelettes, les filtres de Gabor, la matrice de cooccurrence, champ aléatoire de markov et le LBP. Pour évaluer les performances en classification, les auteurs ont utilisé quatre ensembles d'images à l'aide du classificateur k plus proches voisins, l'analyse de texture basée sur les ondelettes affiche des meilleures performances. [Prasetyo et al, 2010] ont évalué les performances de la matrice de cooccurrence, la méthode LBP, les ondelettes, la granulométrie pour la classification des textures de bois exotique. Cinq classifieur sont alors utilisés (classifieur linéaire et quadratique, réseaux de neurones, support à vaste marge, et k plus proche voisins). En se basant sur différents facteurs tels que la complexité de calcul, taux de classification et la durée des calculs, les auteurs ont déduit que la méthode LBP est la plus performante pour ce type d'images. Dans [Chaieb et al, 2014], différents descripteurs de texture utilisés couramment dans les images mammaires sont comparés afin de tester leur capacité à caractériser les lésions mammaires. Les résultats obtenus ont montré que les approches statistiques ont fournis le meilleur résultat de classification.

Les différentes études comparatives ont montré qu'il est délicat de choisir une méthode pertinente. De plus, si l'on veut comparer les résultats de plusieurs études, il faut se placer exactement dans les mêmes conditions d'acquisition ou de traitement, ce qui est très difficile à

mettre en œuvre. Bien qu'elle soit très ancienne, la matrice de cooccurrence reste encore d'actualité. Elle est facile à mettre en œuvre et assure un bon compromis efficacité/temps de calcul. Les filtres de Gabor et les ondelettes sont très largement employés et exploités dans la littérature du fait qu'ils permettent de localiser spatialement le contenu fréquentiel et de fournir des informations à différentes résolutions. En raison de leur simplicité, légère complexité de calcul, l'opérateur LBP et ses variantes sont considérées comme des descripteurs de textures de premier plan dans de nombreuses applications.

1.3. Problème de classification des textures

Lors du choix de la méthode d'analyse de texture, un certain nombre de propriétés doivent être considérées dépendamment de l'application envisagée :

- L'invariance à l'éclairage qui se traduit par la sensibilité de la méthode aux variations des niveaux de gris. Ceci est particulièrement important par exemple en vision par machine industrielle, où les conditions d'éclairage peuvent être instables.
- L'invariance à l'échelle spatiale qui est une propriété importante essentiellement si l'échelle spatiale des échantillons inconnus à classer est différente de celle des données d'entraînement.
- L'invariance à la rotation qui se traduit par l'efficacité de l'algorithme à faire face à la rotation de l'image.
- La robustesse au bruit qui mesure la tolérance de la méthode au bruit dans les images d'entrée.
- Ainsi que la complexité de calcul qui permet d'estimer le coût d'un algorithme en fonction de la taille des données.

En effet, les textures peuvent subir un changement dans leur apparence imagée à cause des variations de l'éclairage ou de la position de la caméra (Fig 1.10).

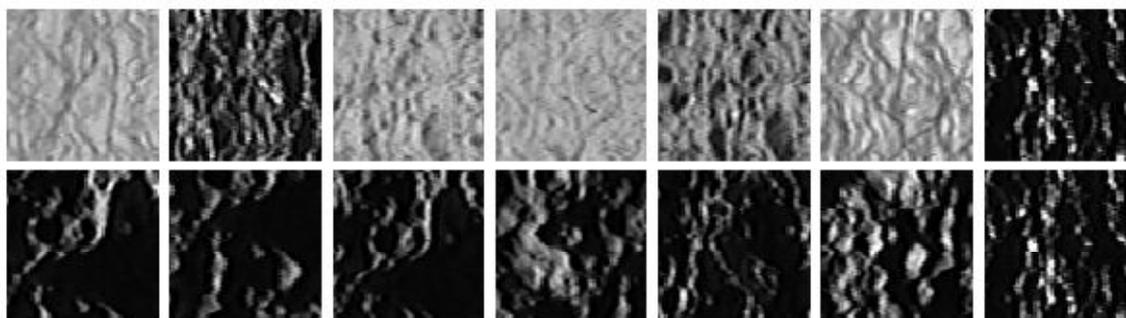


Fig.1.10. Modification de l'aspect visuel de la même texture (Plâtre B, texture # 30 de la base de données CURET [Dana et al, 1999]) suivant la variation des conditions d'acquisition. Rangée du haut: angle de vue constant et éclairage variable. Rangée du bas: éclairage constant et angle de vue variable.

Aussi, le changement de l'échelle ou la rotation de l'image entraînent une toute nouvelle texture avec nouveaux descripteurs et des statistiques différentes. Ceci est illustré dans la figure 1.11.

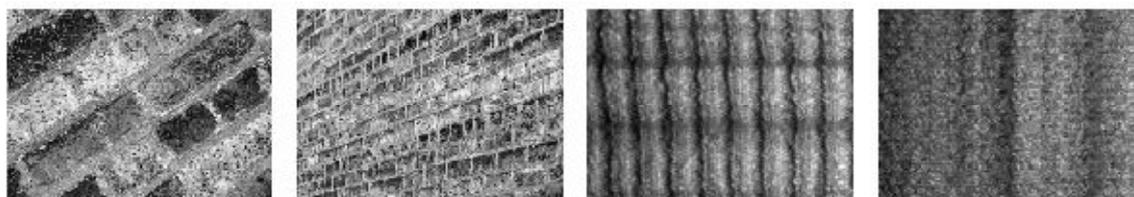


Fig 1.11. L'effet de l'échelle et de la rotation sur les textures. A gauche, la caméra est éloignée du mur de briques. A droite, la surface est tournée dans le plan de 90° alors que toutes les autres conditions d'acquisitions sont identiques [Varma et al, 2005].

Un autre facteur qui intervient est que, deux textures photographiées sous des conditions d'imagerie très différentes peuvent apparaître assez similaires (voir Fig1.12). Les petites différences inter classes entre les textures peuvent poser un problème plus difficile encore.

La combinaison de tous ces facteurs rendent le problème de classification des textures très difficile.

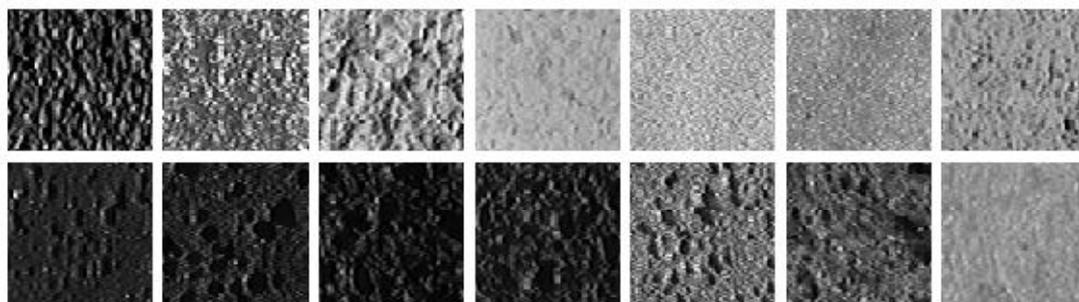


Fig.1.12.Textures extraites de la base de données CURET [Dana et al,1999].Dans la rangée supérieure, la première et la quatrième image sont de texture identique alors que toutes les autres images, même si elles se ressemblent, appartiennent à différentes classes. De même, dans la rangée du bas, les images apparaissent similaires et pourtant elles appartiennent à trois classes de textures différentes.

Dans de nombreuses applications, il est très difficile de garantir que les surfaces soient prises dans les mêmes conditions d'éclairage, de rotation et de résolution spatiale. Extraire des caractéristiques de texture invariantes à ses changements reste un des problèmes importants à résoudre. Cela a inspiré plusieurs études, qui intègrent généralement l'invariance par rapport à une ou au plus deux des propriétés. [Zhang et al ,2002] propose une revue sur les différentes méthodes. Des études comparatives sont proposées dans [Kandaswamy et al, 2011][Liu et al 2017]. Les principaux travaux peuvent être divisés en deux catégories : Méthodes statistiques et méthodes basées sur un modèle [Zhao et al, 2012].

Les statistiques de premier ordre invariantes en rotation les plus simples sont la valeur moyenne, la variance des intensités de pixels et l'histogramme d'intensité. Cependant, elles sont très pauvres en performance, car elles contiennent une quantité limitée d'informations texturales. Une certaine invariance des paramètres dérivés des matrices de cooccurrence ou d'histogrammes de différence peut être obtenue, par exemple, en faisant simplement la moyenne des matrices (histogrammes) ou des caractéristiques calculées suivant différents angles. Le problème avec cette approche réside dans le fait que la directionnalité, une propriété importante de la texture, est perdue lorsqu'une caractéristique isotrope est considérée. Une meilleure technique serait celle qui permettrait une caractérisation de la directionnalité de la texture, tout en évitant une dépendance à l'orientation de la texture.

Les premières approches proposées pour la classification des textures invariantes en rotation comprennent les méthodes basées sur les matrices de cooccurrence généralisées [Davis et al, 1979], les polarogrammes [Davis, 1981] et l'anisotropie de texture [Chetverikov, 1982]. Kashyap et al [Kashyap et al, 1986] ont développé un modèle autorégressif simultané

circulaire ou CSAM (en anglais *circular symmetric autoregressive model*) pour la classification de texture invariante en rotation. Dans [Cohen et al ,1991] la texture est caractérisée par les champs aléatoires de Markov gaussien, le maximum de probabilité est utilisé pour estimer les angles de rotation. Chen et Kundu [Chen et al, 1992] ont adressé l'invariance en rotation en utilisant la décomposition en sous-bandes multicanaux et le modèle de Markov caché ou HMM (*Hidden Markov Model*). Porter et al [Porter et al, 1997] ont exploité les ondelettes pour la classification de texture invariante en rotation. L'invariance à la rotation des attributs issus des filtres de Gabor est obtenue en conservant uniquement la réponse maximale pour toutes les orientations à une échelle donnée [Fahriz et al, 2013]. D'autres descripteurs invariants à la rotation basés sur les champs de Markov [Deng et al, 2004] sont aussi proposés.

Parmi les techniques les plus performantes d'analyse de texture invariantes en rotation, on en distingue principalement deux méthodes basées sur l'analyse locale des éléments de texture: le sac de mots ou BoW(en anglais *the Bag of Words*) et le modèle binaire local ou LBP (*Local Binary Pattern*) [Liu et al, 2014] [Hafiane et al, 2015] [Mehta et al, 2016].

Les méthodes BoW utilisent une phase d'apprentissage pour construire les descripteurs de texture. Dans un premier temps, les réponses locales à une banque de filtres [Varma et Zisserman, 2005] ou à des propriétés de voisinage [Varma et Zisserman, 2009] sont utilisées pour générer un vecteur de caractéristiques des images d'apprentissage. Ensuite, les vecteurs sont regroupés pour construire un modèle ou dictionnaire de textons. Ce modèle est utilisé pour construire le descripteur de chaque image. Divers schémas ont été développés en utilisant cette procédure.

Leung et Malik [Leug et Malik, 2001] proposent le banc de filtre Leung-Malik (LM). L'ensemble LM est un banc de filtres à la fois multi-échelle et multi-orientation. Il se compose de 48 filtres construits à partir des dérivées premières et secondes de Gaussiennes selon 6 orientations et 3 échelles, de 8 Laplaciens de Gaussiennes et de 4 Gaussiennes. Les auteurs utilisent ce banc de filtres dans un problème de classification de textures 3D avec des variations de luminosité et de surface. Leur méthode propose de construire un vocabulaire de textons, utilisant l'ensemble LM, afin d'obtenir une description générique des surfaces des textures. Ce vocabulaire est utilisé avec un algorithme de chaînes de markov afin de reconnaître les textures 3D.

Cula et Dana [Cula and Dana, 2004] ont modélisé la texture en fonction des directions de visualisation et d'eclairage représentées par un histogramme qui code la distribution

statistique des structures locales, la méthodologie globale est la même que celle utilisée dans [Leug et Malik, 2001] sauf que toutes les occurrences de textons 3D ont été remplacées par des textons 2D.

[Varma et Zisserman, 2005] ont développé un modèle statistique utilisant un banc de filtres de réponse maximum (VZ_MR8) pour créer un vocabulaire de textons. Le banc de filtres VZ_MR8 est similaire au banc de filtres LM, il se compose de 38 filtres (voir Fig1.13). Cependant, seulement 8 réponses de filtres sont conservées en prenant à chaque échelle, la réponse maximum des filtres anisotropiques selon toutes les orientations. Le nombre de réponses est ainsi réduit de 38 à 8 (3 échelles pour 2 filtres anisotropiques, plus 2 filtres isotropiques). Après avoir obtenu tous les vecteurs caractéristiques à partir d'un ensemble d'apprentissage, une bibliothèque de textons est conçue par classification. Ainsi, une image donnée est attribuée à un texton en recherchant le plus proche dans la bibliothèque, un histogramme statistique est construit en fonction de la distribution des textons.

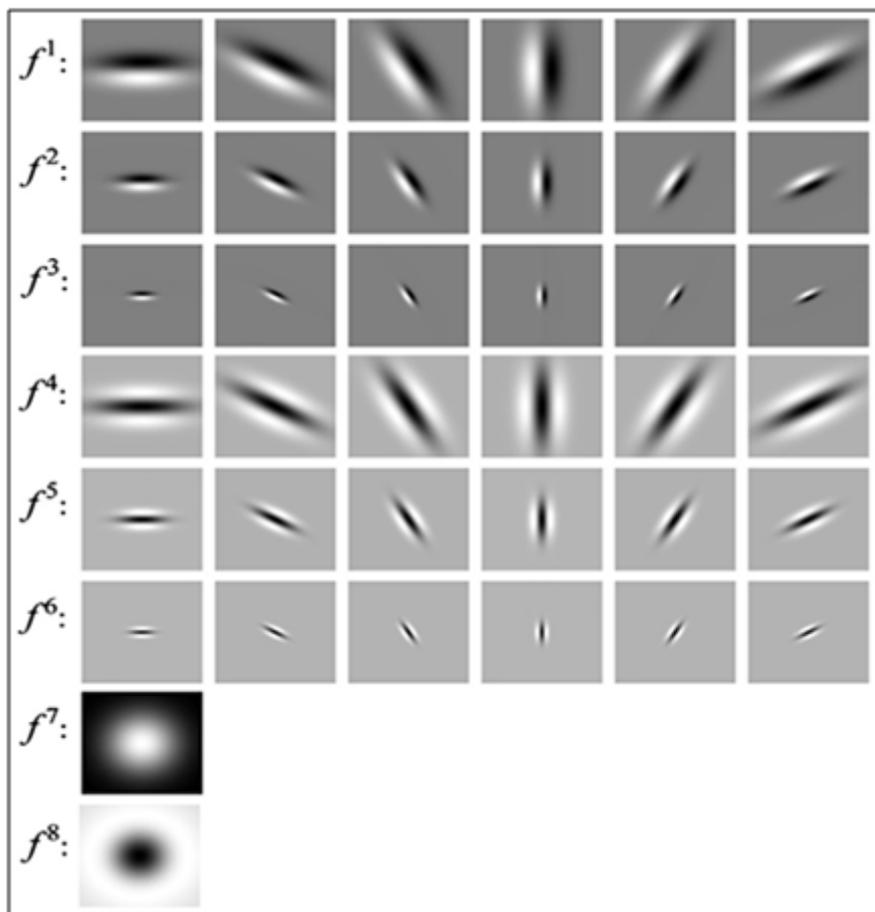


Fig.1.13. Banc de filtres MR8 composé de 38 filtres à la fois isotropiques et anisotropiques.

Dans un travail plus récent, les mêmes auteurs, [Varma et Zisserman, 2009] ont soutenu que la banque de filtres n'est pas nécessaire pour générer des textons et qu'elle peut être remplacée par des patches ou taches d'image du voisinage local en considérant les pixels et leurs intensités sur un voisinage de taille $N \times N$. D'une manière similaire avec VZ_MR8, un vocabulaire de textons est d'abord construit à partir d'un ensemble d'apprentissage, une image de texture inconnue est classée en fonction de la distribution des textons. Pour assurer l'invariance en rotation, le VZ_Joint considère l'orientation dominante du patch. La figure 1.14 représente les principales différences entre les représentations Joint et VZ-MR8.

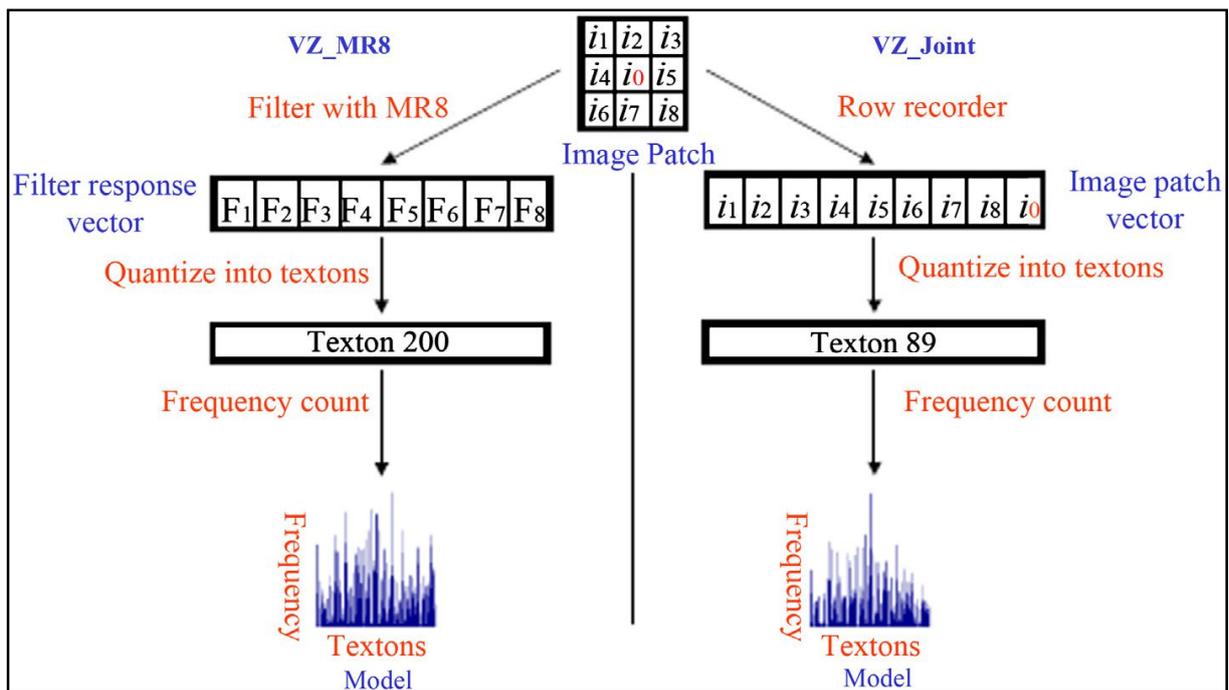


Fig. 1.14. Principale différence entre les représentations VZ_Joint et VZ_MR8 est que les patches d'image source sont utilisés directement dans la représentation VZ_Joint par opposition aux réponses dérivées des filtres dans VZ_MR8 [varma et al, 2009].

Les méthodes VZ_MR8 et VZ_Joint sont connues pour leurs bonnes performances de classification obtenues sur les bases de données volumineuses et complexes [Vama et al, 2005] [Varma et al, 2009], elles sont souvent prises comme des méthodes de références dans plusieurs travaux entre autres [Guo et al, 2010b][Liu et al, 2014][Hafiane et al, 2015].

D'autre part, la plupart des approches basées sur les modèles LBP [Ojala et al., 2002b] n'utilisent pas l'apprentissage pour construire le descripteur de texture. Rappelons que la méthode LBP est basée sur le signe de la différence entre le pixel central et ses voisins. Une fois le code LBP de chaque pixel de l'image est défini, un histogramme sera construit pour représenter l'image de la texture, cette méthode sera d'ailleurs exposée en détail dans le

chapitre suivant. La méthode LBP est très intéressante en raison de ses avantages remarquables: facilité de mise en œuvre, invariance aux changements d'éclairage monotones, invariance à la rotation et une faible complexité de calcul. En raison du grand succès de LBP en vision par ordinateur et la reconnaissance des formes, de nombreuses variantes sont proposées entre autres [Qi et al, 2014][Wang et al, 2015][Hafiane et al, 2015][Pan et al, 2015][Liu et al, 2016] et [Liu et al 2017].

Inspiré par les méthodes LBP, nous proposons dans ce travail deux méthodes d'analyse de texture. L'idée est d'exploiter des extensions de la notion d'extrema de gris afin d'extraire les descripteurs de texture.

Les principaux avantages des méthodes proposées sont : paramètres de texture très discriminatifs, invariance en rotation, invariance à l'échelle, simplicité de mise en œuvre, schéma de calcul rapide, faible dimension des paramètres, aussi les méthodes proposées permettent de coder de manière compacte quelque soit l'échelle d'analyse ou la taille de la fenêtre de voisinage considérée. Les méthodes proposées seront présentées dans les chapitres 3 et 4.

1.4. Algorithmes de classification

La classification de textures consiste à associer à une texture donnée, une classe ou un label en fonction des caractéristiques présentes. La première étape est l'extraction des caractéristiques de textures. Ces attributs sont calculés à partir des méthodes que nous avons présentées dans la section précédente. Ces caractéristiques sont regroupées sous forme de vecteurs numériques et sont, par la suite, directement utilisées par un classificateur. La deuxième étape du processus de classification est donc le choix du classificateur. Il est possible d'utiliser soit une approche supervisée, soit une approche non-supervisée.

Dans le cadre de la classification non-supervisée [Qin et al., 2008], les textures similaires sont regroupées à l'intérieur d'une même classe sans l'intervention de connaissances a priori et simplement à partir d'estimations de similarité entre caractéristiques. Dans le cadre de la classification supervisée [Li et al., 2003], le processus de classification des textures comporte deux phases : la phase d'apprentissage et la phase de reconnaissance ou de test. Dans la phase d'apprentissage, l'objectif est de construire un modèle de chaque classe de texture présente dans les données d'apprentissage, ce qui comprend généralement des images avec des

étiquettes de classe connue. La texture des images d'apprentissage est caractérisée par la méthode d'analyse de texture choisie, chaque image est décrite par un ensemble de caractéristiques de texture. Dans la phase de reconnaissance, la texture des images de l'échantillon inconnu est décrite en premier par la même méthode d'analyse de texture. Ensuite, les caractéristiques texturales de l'échantillon sont comparées à celles des images d'apprentissage avec un algorithme de classification défini, l'échantillon est assigné à la classe de la meilleure correspondance.

Dans cette thèse, seule la classification supervisée sera considérée, et la précision de la classification peut se référer au pourcentage d'échantillons de texture correctement classés. Dans ce qui suit, nous allons décrire sommairement les algorithmes de classifications les plus utilisés dans le cadre de la problématique exposée et qui sont la méthode des k plus proches voisins et les séparateurs à vaste marge (SVM) [Ojala et al., 2002b][Liao et al, 2009][Tan et al., 2010][Khellah, 2011][Liu et al., 2014].

1.4.1. La méthode des k plus proches voisins

L'algorithme des k plus proches voisins (noté parfois k-NN pour *k-Nearest Neighbor*) fait partie des méthodes de classification les plus couramment utilisées. Il a l'avantage d'être à la fois simple et efficace. L'algorithme k-NN permet de traiter des nuages de points non linéairement séparables. Le principe de cet algorithme consiste à déterminer pour chaque nouvelle donnée x à classer, la liste des k-plus proches voisins déjà classées. La classe de x est déterminée en fonction de la classe majoritairement représentée par les k voisins de l'observation x (Fig.1.15). La détermination de leur similarité est basée sur des mesures de distance.

La métrique la plus utilisée dans la classification des textures est la distance des Chi deux (χ^2) définie comme suit :

$$\chi^2(u, v) = \frac{1}{2} \sum_i \frac{(u_i - v_i)^2}{(u_i + v_i)} \quad (1.7)$$

u_i et v_i représentent respectivement les vecteurs caractéristiques des données u et v .

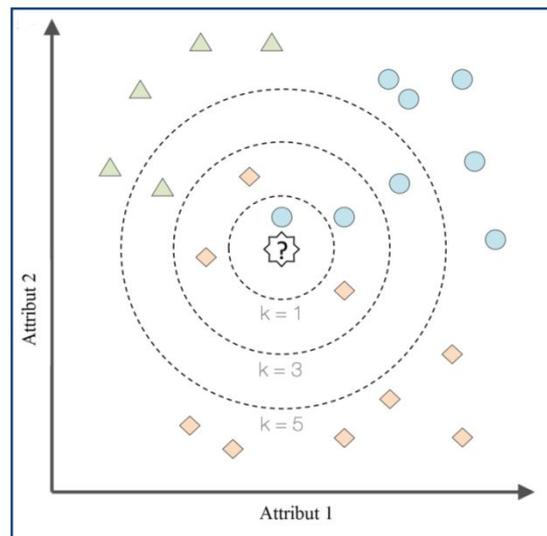


Fig1.15. Si on fixe $k=1$ l'individu non classé appartient à la classe des ronds, si $k=3$ l'individu appartient à la classe des carrés.

1.4.2. Séparateurs à vaste marge

La méthode des séparateurs à vaste marge ou SVM (*Support Vector Machine*) a été proposée par Vapnik en 1995 [Cortes et al, 1995] pour la classification des données. Initialement, elle a été définie pour résoudre un problème de discrimination à deux classes. Cette approche repose sur deux notions clés : la notion de marge maximale et la notion de noyau.

La marge est la distance séparant la frontière de séparation (l'hyperplan) et les prototypes les plus proches de cette frontière. Ces prototypes sont appelés vecteurs supports (Fig1.16). Dans les SVM, la frontière de séparation est choisie comme étant celle qui maximise la marge.

Parmi les modèles des SVM, il y a les cas linéairement séparable et les cas non linéairement séparable. Les premiers sont les plus simples car ils permettent de trouver facilement le classifieur linéaire.

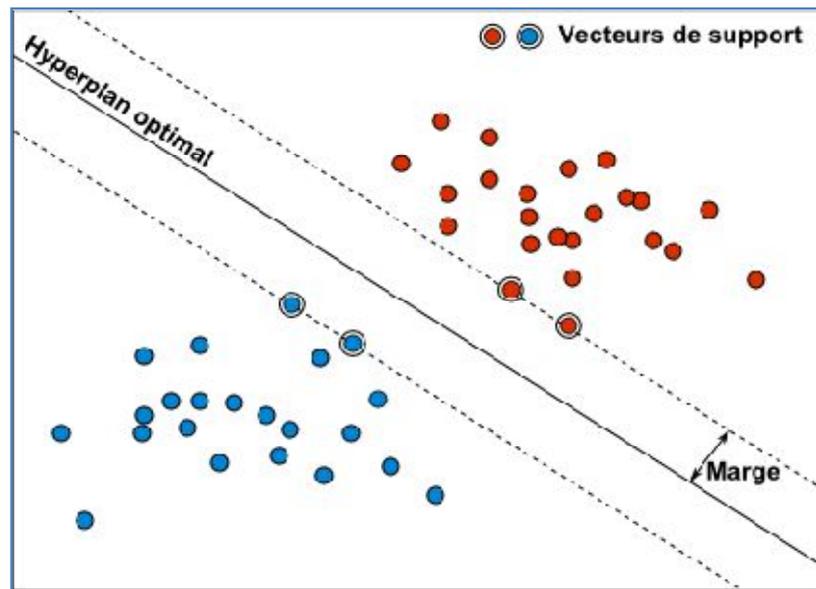


Fig1.16. Exemple de fonctionnement du SVM dans le cas linéaire. Seuls 4 vecteurs de support sont nécessaires pour définir l'hyperplan optimal.

Pour surmonter les inconvénients des cas non linéairement séparables, le principe des SVM est de changer l'espace des données. La transformation non linéaire des données peut permettre une séparation linéaire des exemples dans un nouvel espace. Il s'agit d'une transformation d'un problème de séparation non linéaire dans l'espace de représentation en un problème de séparation linéaire dans un espace de re-description de plus grande dimension. Cette transformation non linéaire est réalisée via une fonction noyau ϕ (voir Fig1.17).

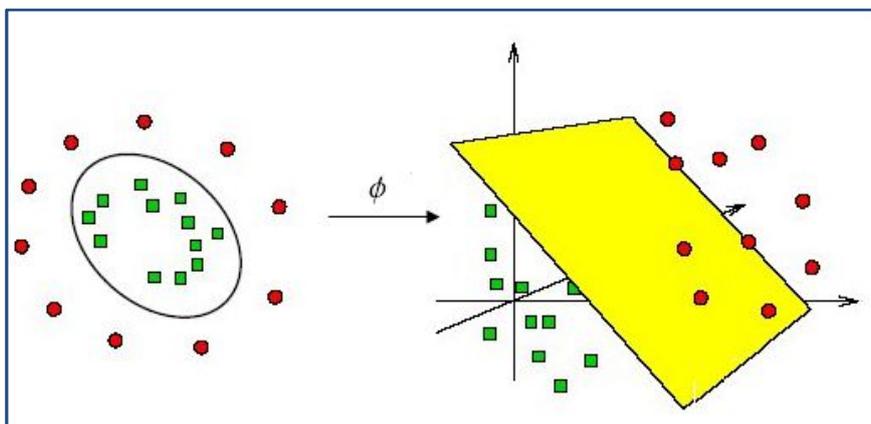


Fig1.17 Exemple de transformation d'espace d'attributs utilisant une fonction noyau ϕ

Dans cette thèse, les opérations impliquant les SVMs ont été réalisées au moyen de la bibliothèque LIBSVM [Chang et al, 2011]. Les fondements mathématiques des SVMs sont présentés en détail dans l'ANNEXE A.

1.4.3. Etapes de classification des textures

La classification de texture comprend généralement les étapes suivantes :

1. Sélection de la base de données : les données ou les images textures peuvent être artificielles ou naturelles, éventuellement obtenues dans une application du monde réel. Les textures tirées de l'album de Brodatz [Brodatz, 1965] sont probablement les données d'images le plus largement utilisées dans la littérature de l'analyse de texture.
2. Partitionnement des données images en sous-images: les bases de données sont souvent limitées en termes de nombre d'images sources disponibles, donc afin d'augmenter la quantité de données, les images sont divisées en sous-images chevauchantes ou disjointes, de taille prédéfinie.
3. Prétraitement des images : les images peuvent avoir différentes propriétés de niveaux de gris. Les images sont alors pré -traitées afin d'avoir une distribution uniforme des niveaux de gris, ou des statistiques de premier et de second d'ordre identiques, par égalisation d'histogramme, par exemple.
4. Partitionnement des (sous)images en bases d'apprentissage et bases de tests : Afin d'obtenir une estimation non biaisée de la performance de la classification de texture, les bases d'apprentissage et bases de tests devrait être indépendantes. Différentes approches peuvent être utilisées, les sous-images sont divisées en N ensembles disjoints, dont $N-n$ servent de données d'entraînement et les n restantes sont utilisées au cours des tests. Les sous-images peuvent aussi être partitionnées en données d'apprentissage et de test de façon aléatoire. Cette opération peut être répétée pour un certain nombre d'itérations pour une estimation plus fiable de la performance.
5. Sélection de l'algorithme de classification : l'algorithme de classification peut avoir une grande incidence sur la performance finale de la procédure de classification de texture. En plus de d'algorithme de classification d'autres sélections telles que les métriques utilisés sont aussi importantes.
6. Définition du critère de performance: les éléments de l'ensemble de tests sont classés, et la proportion d'éléments correctement classés (précision de la classification) ou la proportion d'éléments mal classés (erreur de classification) sont utilisées comme critère de performance.

1.5. Les bases des images utilisées : vue d'ensemble

Pour évaluer l'efficacité des méthodes proposées dans ce travail, plusieurs bases de données sont exploitées : Brodatz [Brodatz,1965], Outex [Ojala et al, 2002], CURET [Dana et al,1999], et KTH-TIPS2b [Mallikarjuna et al,2006]. La table 1.1 dresse les propriétés de chacune des bases.

1.5.1. Base de données de Brodatz

Pour la base de données Brodatz, nous avons exploité le même sous-ensemble d'images qui a été précédemment utilisé dans [Liao et al, 2009] et [Liu et al, 2014]. Comme il est montré sur la figure 1.18, il y a 24 classes de textures homogènes qui sont D1, D4, D16, D19, D21, D24, D28, D32, D53, D54, D57, D65, D68, D77, D82, D84, D92, D93, D95, D98, D101, D102, D106, D111. La taille de chaque image de texture est de 640 x 640 pixels. Chaque image est divisée en 25 sous-images non chevauchantes de taille 128x128 pixels, les sous images sont ensuite sous-échantillonnées à la taille 64x64 pixels. Au total 600 images sont utilisées.

Bases de données	rotation de l'image	variation de l'éclairage	variation de l'échelle	nombre de classes	nombre d'images par classe	nombre total d'échantillons	Taille de l'image (pixels)
Brodatz				24	25	600	64×64
Outex_TC10	✓			24	180	4320	128×128
Outex_TC12_000	✓	✓		24	200	4800	128×128
Outex_TC12_001	✓	✓		24	200	4800	128×128
CURET	✓	✓		61	92	5612	200×200
KTH-TIPS2b	✓	✓	✓	11	4×108	4752	200×200*

* la taille de quelques images échantillons est inférieure à 200×200 pixels

Table 1.1. Propriétés des bases de données utilisées dans les tests

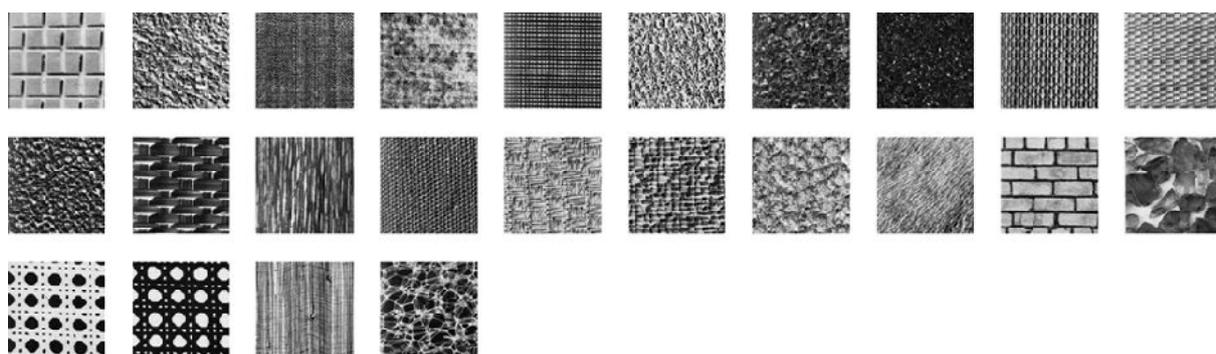


Fig.1.18. Les 24 textures extraites de l'album de Brodatz

1.5.2. Base de données Outex

Cette base comprend différentes suites de test pour l'évaluation de la classification des textures, nous utilisons Outex_TC 10, et Outex_TC 12. Ces deux suites de tests contiennent les mêmes 24 classes de texture (Fig1.18). Chaque texture est recueillie sous trois différentes conditions d'éclairage ("inca", "TL84 et "horizon") et neuf différents angles de rotation (0°, 5°, 10°, 15°, 30°, 45°, 60°, 75° et 90°). Il ya 20 échantillons de texture non chevauchante de taille 128×128 pour chaque classe et pour chaque réglage. Par conséquent, il y a trois ensembles de données dont chacun contient 4320 ($24 \times 20 \times 9$) texture échantillons. Les différents procédés expérimentaux pour les deux ensembles sont comme suit:

- 1) Outex_TC10 (TC10) : pour chaque classe, les échantillons capturés sous l'éclairage "inca" et l'angle 0° sont utilisés pour l'apprentissage du classificateur et les échantillons capturés sous les huit autres angles de rotation avec le même éclairage sont utilisés pour les tests. Ainsi, il y a 480 (24×20) modèles d'apprentissage et 3840 ($24 \times 8 \times 20$) échantillons de validation.
- 2) Outex_TC12 (TC12): le classificateur est formé avec les mêmes échantillons d'apprentissage que TC10, et testé avec tous les échantillons capturés sous l'éclairage "TL84 ou "horizon". Ainsi, Il y a 480 (24×20) modèles et 4320 échantillons ($24 \times 20 \times 9$) de validation pour chaque éclairage .

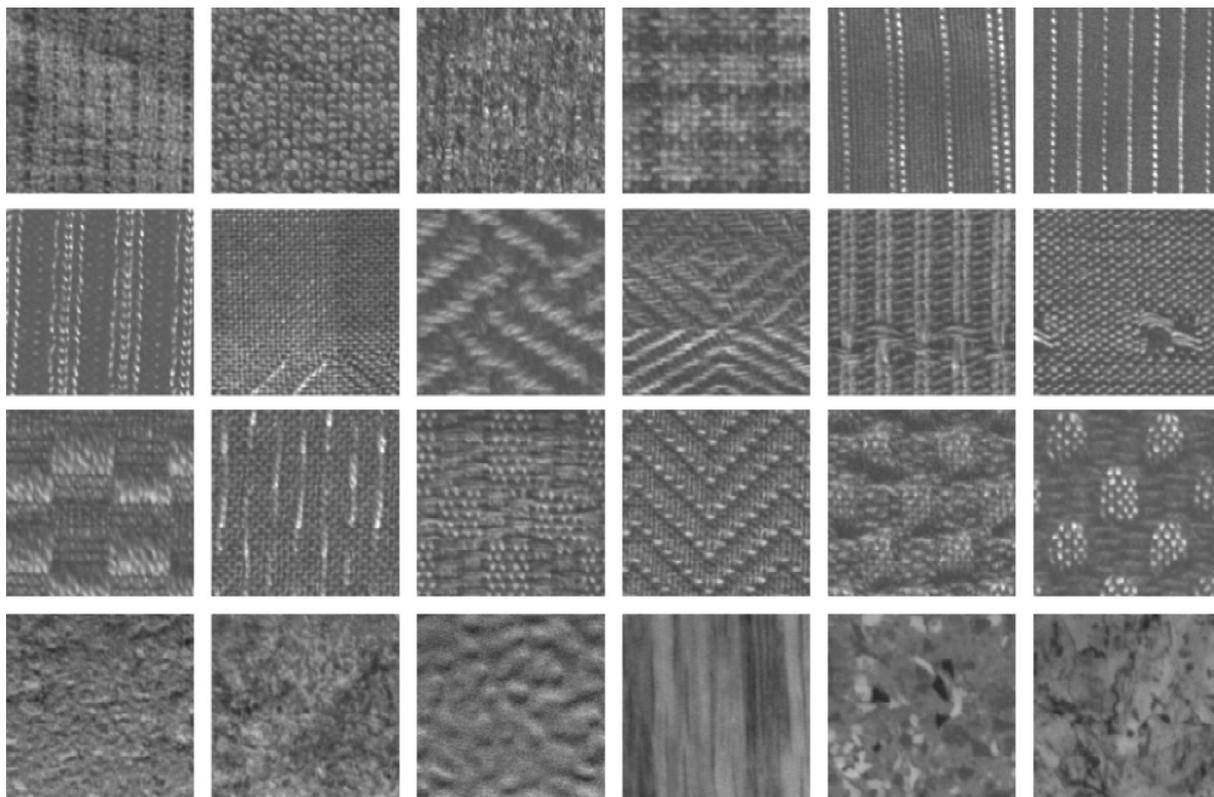


Fig.1.19. Echantillon des 24 textures des bases Outex_TC10 et Outex_TC12.

1.5.3. Base de données CURET

La base de données CURET (*Columbia-Utrecht Reflectance and Texture*) couvre 61 surfaces de matériaux différents, chacune est observée avec 205 combinaisons de visualisation et de direction d'éclairage, mais en général seulement 92 échantillons par classe sont sélectionnés pour la classification [Varma et al, 2005][Varma et al, 2009][Guo et al, 2010][Khellah, 2011][Liu et al, 2014]. La figure 1.20 représente un échantillon des 61 classes de texture de cette base.

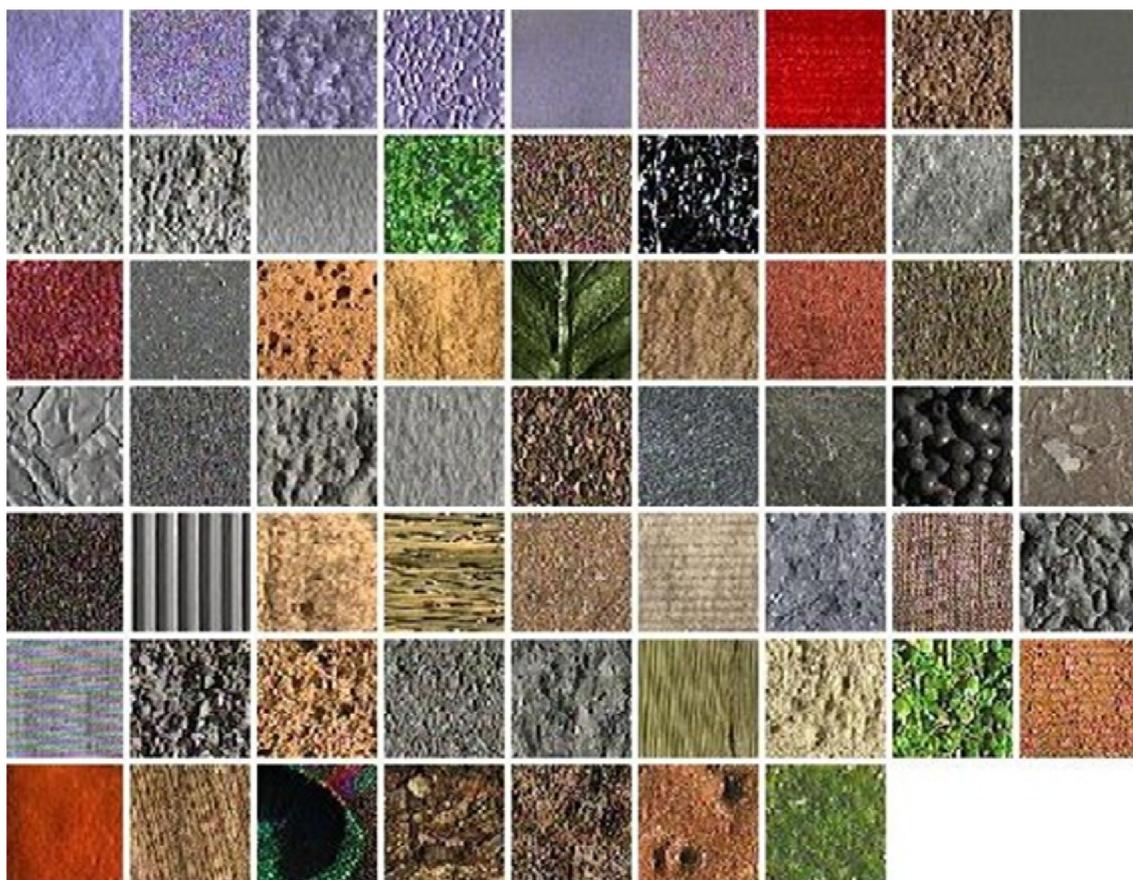


Fig.1.20. 61 classes de textures de la base CURET.

1.5.4. Base de données KTH_TIPS2b

La base KTH-TIPS2b est un autre ensemble de données créé pour répondre à certaines des lacunes de la base CURET, l'une des propriétés de cette base est la variation de l'échelle et l'utilisation de différents échantillons du même matériau. La base KTH-Tips2b dispose de 11 classes de textures. Dans chaque classe, quatre objets ont été imagés sous neuf échelles différentes, quatre directions d'éclairage différentes et trois poses différentes. Cette base contient donc au total 4752 images ($11 \times 108 \times 4$). La plupart des échantillons sont de taille 200×200 pixels, mais quelques uns sont plus petits en raison des problèmes d'échelle. La figure 1.21 montre les classes de texture de la base KTH-Tips2b.

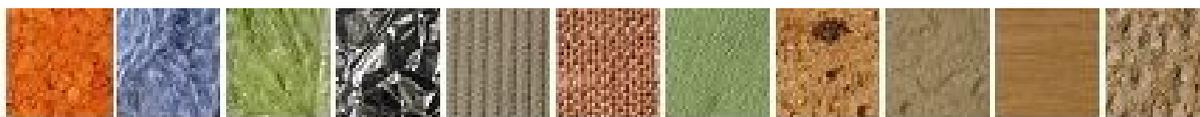


Fig1.21. 11 différentes classes de texture de la base KTH-Tips2b

Durant tous les tests, les textures de chaque base sont converties en niveaux de gris et normalisées à la moyenne zéro et l'écart type de valeur 1 suivant l'équation (1.8)

$$I = \frac{I - \mu}{\sigma} \quad (1.8)$$

Où μ est la moyenne de l'image I, $\mu = \frac{\sum_{i=1}^N \sum_{j=1}^M I(i,j)}{N \times M}$

et σ est l'écart type donné par $\sigma = \left(\frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M (I(i,j) - \mu)^2 \right)^{1/2}$

1.6. Discussions

Dans de nombreuses applications, il est très difficile de garantir que les surfaces de texture soient prises dans les mêmes conditions d'éclairage, de rotation et de résolution spatiale. Extraire des caractéristiques de texture invariantes à ses changements reste un des problèmes importants à résoudre. Parmi les méthodes d'analyse proposées dans la littérature, les méthodes VZ_MR8 et VZ_Joint sont connues pour leurs bonnes performances de classification obtenues sur les bases de données volumineuses et complexes. D'autre part, la méthode LBP est très intéressante en raison de ses avantages remarquables: facilité de mise en œuvre, invariance aux changements d'éclairage monotones, invariance à la rotation et une faible complexité de calcul. Des résultats de classification impressionnants sont obtenus sur des bases de données de texture représentatives. En raison du grand succès de LBP en vision par ordinateur et la reconnaissance des formes, de nombreuses variantes sont proposées afin d'améliorer le pouvoir discriminant mais aussi sa robustesse au bruit.

Inspiré par les méthodes LBP, nous proposons dans ce travail deux méthodes d'analyse de texture. Ainsi, le chapitre suivant sera consacré à la présentation des méthodes LBP et ses principales variantes notamment celles utilisées dans notre étude.

CHAPITRE 2

Les Méthodes LBP

2.1. Préambule

La méthode du motif binaire local en anglais *Local Binary Pattern* (LBP) a été initialement introduite par [Ojala et al., 1996]. Cette méthode est utilisée avec succès dans divers domaines comme la récupération d'images, la reconnaissance faciale, l'analyse d'image biomédicale [Murala et al, 2012, Ahonen et al, 2006, Huang and al, 2011, Nanni et al, 2010,]. En termes d'efficacité discriminante, cette méthode offre de bonnes performances et contient des informations structurelles et statistiques. L'opérateur LBP peut être invariant aux rotations et changements d'échelles et offre une grande tolérance face aux changements de luminosité.

Dans la première partie de ce chapitre, la méthode LBP de base ainsi que ses trois principaux modèles à savoir les modèles uniformes, les modèles invariants en rotation et les modèles uniformes et invariants en rotation seront exposés. Ensuite, dans une seconde partie, plusieurs variantes de LBP seront également présentées.

2.2. La méthode LBP de base

L'opérateur LBP est proposé par Ojala et al. [Ojala et al., 1996] pour appuyer la mesure du contraste local de l'image. La structure locale de la texture est caractérisée en utilisant un seuillage avec la valeur du pixel central sur un voisinage de taille 3×3 pixels.

Pour permettre l'analyse multi-résolution et l'invariance par rotation, une formulation plus générale définie sur un voisinage circulaire symétrique a été proposé dans [Ojala et al., 2002b]. Le LBP généralisé utilise P points échantillons uniformément répartis sur un rayon R autour d'un pixel central g_c de coordonnées (x_c, y_c) . La figure (2.1) montre un exemple d'un pixel central g_c et ses P voisins uniformément répartis sur un cercle de rayon R .

La position (x_p, y_p) d'un point voisin g_p , où $p \in \{0, \dots, P-1\}$ est donnée par l'équation (2.1). Les niveaux de gris qui ne coïncident pas avec le pixel central sont estimés par interpolation.

$$(x_p, y_p) = (x_c - R \sin\left(\frac{2\pi p}{P}\right), y_c + R \cos\left(\frac{2\pi p}{P}\right)) \quad (2.1)$$

Le Motif LBP est calculé en comparant la valeur du pixel centrale g_c avec celles de ses P voisins comme suit :

$$LBP_{R,P} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p \quad s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (2.2)$$

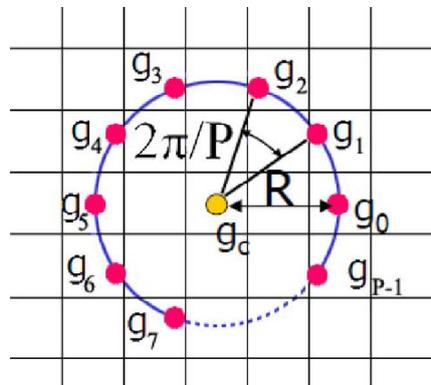


Fig.2.1. Pixel central g_c et ses P voisins uniformément repartis sur un rayon R.

La figure 2.2 (Fig.2.2) illustre un exemple de calcul du code LBP. Le motif binaire est calculé par seuillage des niveaux de gris des 8 voisins avec le niveau de gris du pixel central. La valeur de LBP est ensuite calculée en tenant compte des poids de chaque pixel du voisinage suivant l'équation (2.2).

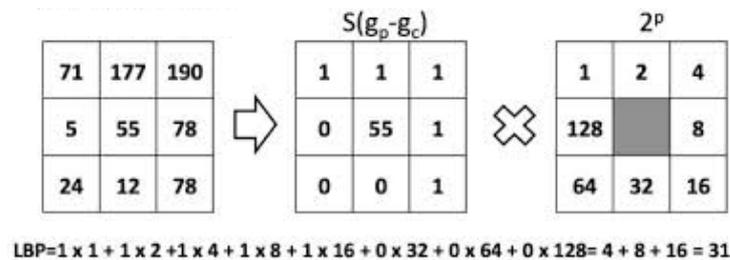


Fig.2.2 Illustration du calcul des motifs LBP

L'image texturée I de taille $N \times M$ est ainsi définie par l'histogramme $H(l)$ des motifs LBP, LBP est calculé sur chaque point (i,j) :

$$H(l) = \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M \delta(LBP_{R,P}(i,j) - k) \quad \delta(x) = \begin{cases} 1 & x = 0 \\ 0 & \text{sinon} \end{cases} \quad (2.4)$$

$0 < l < d$ ($d = 2^P$ est le nombre de motifs LBP).

Pour inclure des informations de texture à différentes échelles, l'opérateur de LBP a été étendu en utilisant des voisinages de différentes tailles [Ojala et al. 2002b], cet opérateur est noté $LBP_{R,P}$, les valeurs (R, P) sont sélectionnées comme suit : $(1,8)$, $(2,16)$, $(3, 24)$, ..., $(R, 8R)$, (voir Fig.2.3).

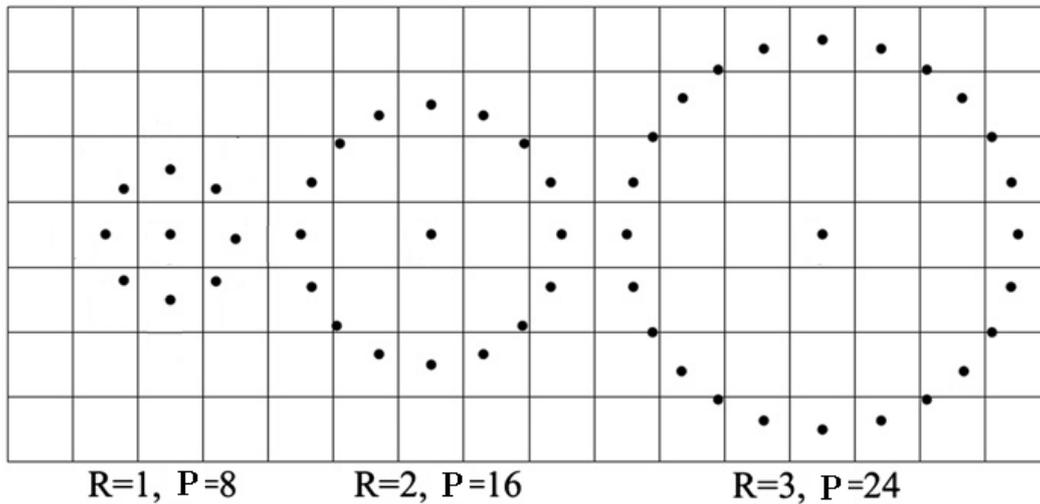


Fig.2.3. Trois voisinages de tailles différentes utilisées dans le calcul de LBP.

Le problème majeur de $LBP_{R,P}$ est la croissance exponentielle du nombre des modèles dépendamment de la taille du voisinage P . Par exemple dans le cas de $P=8$, le nombre de modèles LBP possible est de 2^8 soit 256. Pour remédier à ce problème, Ojala et al. [Ojal et al. 2002b] définissent trois modèles : les modèles uniformes, les modèles invariants en rotation et les modèles uniformes invariants en rotation.

2.2.1. Modèle LBP uniforme

La valeur d'un modèle LBP uniforme représente le nombre de transitions spatiales circulaires (binaire 0/1ou1/0), elle peut être mathématiquement calculée par l'équation (2.5) :

$$U(LBP_{R,P}) = \sum_{p=0}^{P-1} |s(g_p - g_c) - s(g_{p+1} - g_c)| \tag{2.5}$$

Si $U(LBP_{R,P})$ est inférieure ou égale à 2 ($U \leq 2$), le modèle est appelé modèle uniforme, noté $LBP_{R,P}^u$. La figure 2.4 donne un exemple de trois modèles LBP, Le modèle 00000001 (2

transitions) est uniforme, alors que les modèles 00100101(6 transitions) et 01010101 (8 transitions) sont non uniformes.

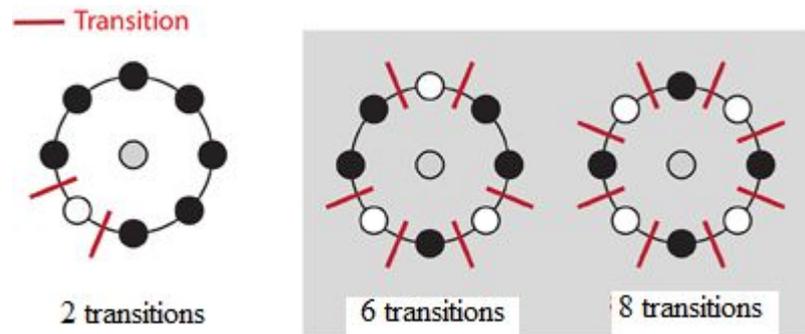


Fig.2.4. Exemple de modèles uniformes et non uniformes, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP.

Le $LBP_{R,P}^{u2}$ a $P \times (P-1) + 3$ valeurs de sortie. Ces modèles sont considérés comme des motifs fondamentaux au sein de l'image texturée [Ojala et al. 2002b]. La figure 2.5 représente les 58 modèles uniformes possibles dans le cas de $P=8$.

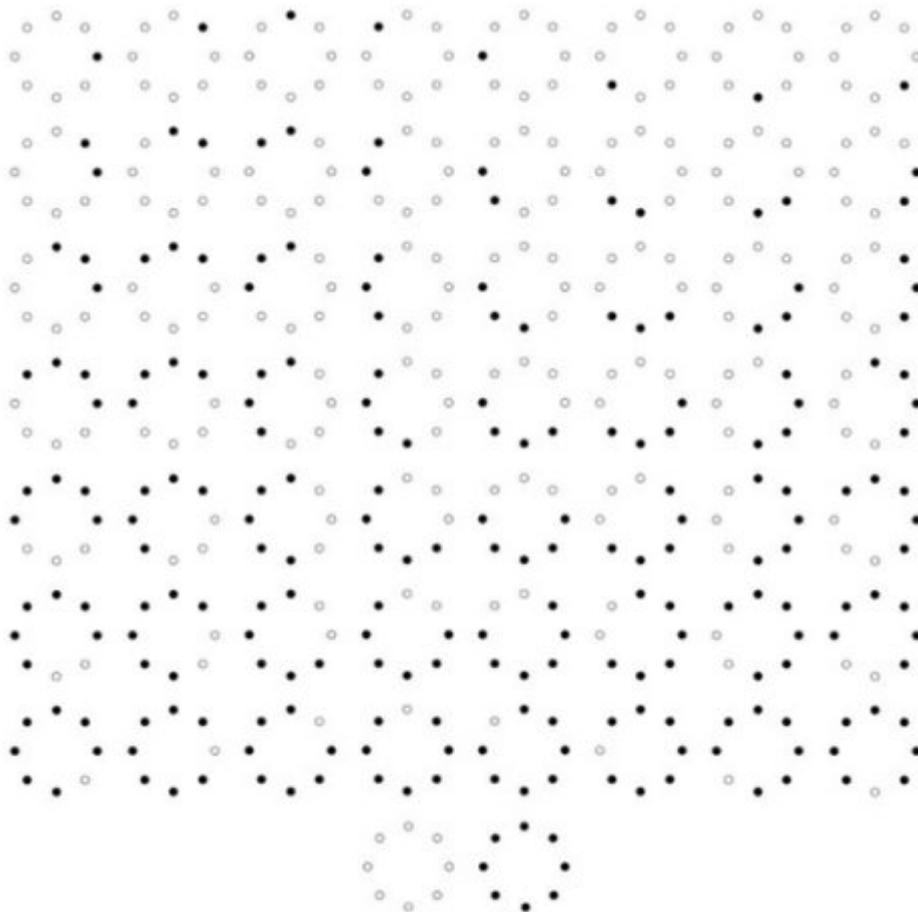


Fig.2.5. Les 58 modèles uniformes, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al., 2002b]

2.2.2. Modèle LBP invariant par rotation

Les motifs LBP invariants en rotation sont définis comme suit (2.6) :

$$LBP_{R,P}^{r,i} = \min (ROR(LBP_{R,P}, i) \quad i = 0, 1, \dots, P - 1) \quad (2.6)$$

(ROR (x, i)) correspond au résultat de i décalages circulaires successifs vers la droite des bits du code $LBP_{R,P}$ (codé sur P bits). Le $LBP_{R,P}^{r,i}$ quantifie ainsi les statistiques d'occurrence de modèles individuels invariants en rotation correspondant à certaines micro caractéristiques de l'image.

La figure 2.6 présente les 36 motifs binaires locaux invariants en rotation possibles dans le cas de $P = 8$. A titre d'exemple, le motif "0" permet de détecter les taches sombres, le motif "8" les taches claires et les surfaces monotones, le motif "3" les coins et le motif "4" les contours.

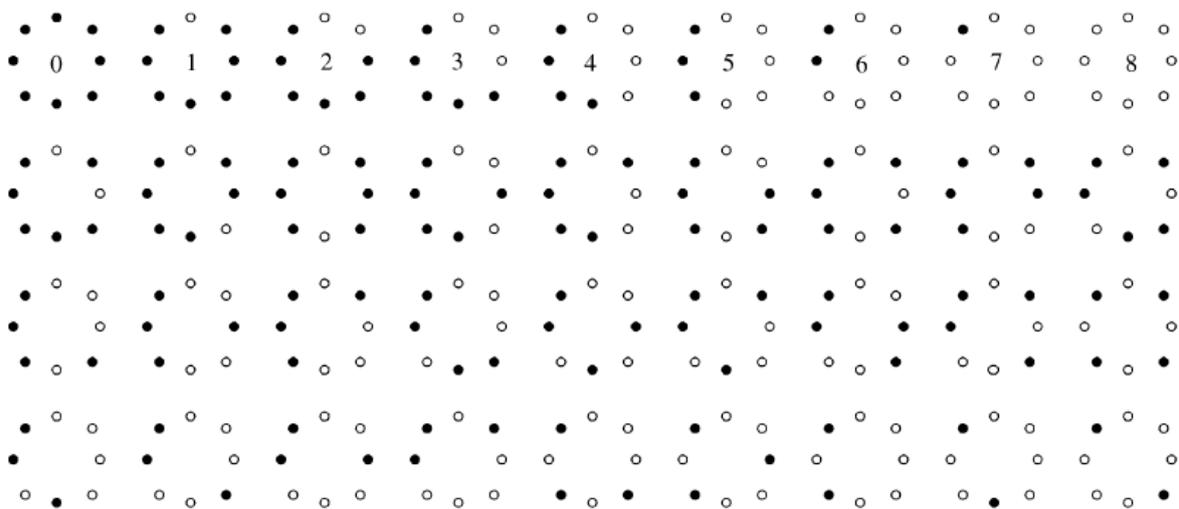


Fig.2.6. Les 36 modèles invariants par rotation définis pour P égale à 8, les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al , 2002b]

En introduisant la définition de l'invariance en rotation, la méthode LBP permet non seulement d'obtenir de très bonnes performances, mais aussi de réduire le nombre de motifs [Ojal et al., 2002b].

2.2.3. Modèle LBP uniforme et invariants en rotation

Afin d'obtenir une meilleure invariance en rotation et de réduire davantage la dimension de l'histogramme de LBP, Ojala et al [Ojala et al., 2002b] proposent les modèles uniformes invariants en rotation notés $LBP_{R,P}^{riu2}$ en se basant sur les modèles $LBP_{R,P}^i$ ayant une valeur d'uniformité U inférieure ou égale à 2. L'opérateur $LBP_{R,P}^{riu2}$ est défini comme suit :

$$LBP_{R,P}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c), & \text{si } U(LBP_{R,P}) \leq 2 \\ P + 1, & \text{sinon} \end{cases} \quad (2.7)$$

Le nombre de motifs uniformes et invariants en rotation est de $P+2$.

La figure 2.7 représente les 9 modèles binaires locaux uniformes et invariants en rotation dans le cas de $P = 8$.

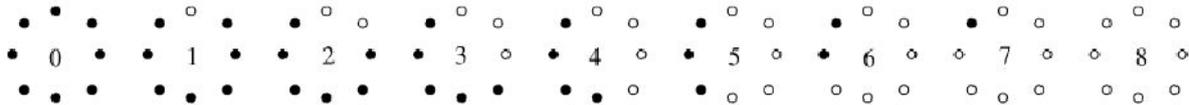


Fig. 2.7. Les 9 modèles uniformes et invariants en rotation définis pour $P=8$. Les cercles noirs et les cercles blancs correspondent respectivement à la valeur binaire 0 et 1 dans le code LBP [Ojala et al, 2002b].

Malgré le grand succès de l'opérateur LBP en vision par ordinateur et le traitement d'images, le descripteur LBP original a quelques limitations essentiellement sa capacité discriminative limitée et son manque de robustesse au bruit. Sur la base de ces problèmes, de nombreuses variantes de LBP ont été développées. Un livre complet sur la famille des descripteurs de texture LBP est proposé dans [Pietikäinen et al, 2011]. Plusieurs études comparatives peuvent aussi être trouvées dans [Nanni et al ,2012],[Song et al, 2013] ,[kylberg et al, 2013] [Brahnam et al ,2014] [Nguyen et al, 2016] et [Liu et al 2017].

Des études exhaustives sont introduites dans [Brahnam et al ,2014] [Liu et al 2017] et ont révélé l'énorme diversité des méthodes LBP.

2.3. Quelques variantes de LBP

Plusieurs variations sont portés sur la méthode LBP de base afin d'améliorer ses performances dans différentes applications [Liu et al 2017]. Ces variations se concentrent sur les différents aspects de la méthode LBP: (1) topologie de voisinage et échantillonnage, (2) seuillage, (3) codage et regroupement, (4) combinaison avec d'autres approches (Voir le tableau 2.1 pour une vue d'ensemble).

Catégorie	Variante	Abbr.	Références
LBP traditionnel	Local Binary Pattern	LBP	[Ojala et al.,1996]
	Uniform Local Binary Pattern	$LBP_{R,P}^{u2}$	[Ojala et al., 2002b]
	Rotation Invariant Local Binary Pattern	$LBP_{R,P}^{ri}$	[Ojala et al., 2002b]
	Rotation Invariant Uniform Local Binary Pattern	$LBP_{R,P}^{riu2}$	[Ojala et al., 2002b]
Topologie de voisinage et échantillonnage	Elongated Local Binary Pattern	EBP	[Liao et al, 2007]
	Local Line Binary Pattern	LLBP	[Petpon et al, 2009]
	Completed Local Binay Pattern	CLBP	[Guo el al, 2010b]
	Local Binary Count	LBC	[Zhao et al, 2012]
	Center Symmetric Local Binary Pattern	CS-LBP	[Heikkilä et al, 2009]
	Multi-block local binay pattern	MBLBP	[Zhang et al, 2007]
	Binary Rotation Invariant and Noise Tolerant	BRINT	[Liu et al., 2014]
	Median Robust Extended LBP Neighborhood Intensity	MRELBP_NI	[Liu et al, 2016]
Seuillage	Texture Spectrum	TS	[Wang et He, 1990]
	Median Local Binary Pattern	MLBP	[Hafiane et al, 2007]
	Local Ternary Pattern	LTP	[Tan et al., 2010]
	Local Quinary Pattern	LQP	[Nanni et al, 2010],
	Soft Local Binary Pattern	SoftLBP	[Ahonen et al, 2009]
	Fuzzy Local binary Pattern	FLBP	[Iakovidis, 2008]
	Noise resistant Local Binary Pattern	NRLBP	[Ret al, 2013]
	Completed Robust Local Binary Pattern	CRLBP	[Yang Zhao et al, 2013]

Table 2.1. Différentes variantes de LBP. Les méthodes utilisées dans nos tests sont indiquées en gras.

Catégorie	Variante	Abbr.	Références
Codage et regroupement	Novel extended local binary pattern	NELBP	[Zhou et al ,2008]
	Noise tolerant local binary pattern	NTLBP	[Fathi et al, 2012]
	Dominant Local Binary Pattern	DLBP	[Liao et al, 2009]
	Local Binary Count	LBC	[Zhao et al, 2012]
	Cooccurrence of Uniform LBP	UUCoLBP	[Qi et al, 2014]
	Cooccurrence of Rotation invariant Uniform LBP	RURUCoLBP	[Qi et al, 2012]
	Pairewise Rotation Invariant co – occurrence LBP	PRICoLBP	[Qi et al, 2014]
	Multiscale Joint encoding of Local Binary Patterns	MSJLBP	[Qi et al, 2013]
Combinaison avec d'autres approches	LBP and Var	LBP/Var	[Ojal et al., 2002b].
	Local Binary Pattern Variance	LBPV	[Guo et al, 2010a]
	CLBC-C CLBC-S CLBC-M	CLBC	[Zhao et al, 2012]
	Adaptative Local Binary Pattern	ALBP	[Guo et al, 2010c]
	LBP and Fourier features	LBP-HF	[Ahonen et al, 2009].
	DLBP and Gabor filters	DLBP-Gabor	[Guo et al, 2010c]
	LBP and Dominant Neighbors Structure	LBP- DNS	[Ahonen et al, 2009].
	LBP and SIFT	LBP and SIFT	[Klare et al, 2013]

Table 2.1. Différentes variantes de LBP(Suite). Les méthodes utilisées dans nos tests sont indiquées en gras.

2.3.1. Topologie du voisinage et échantillonnage

Le choix d'un voisinage approprié à une méthode LBP a un impact significatif sur la performance finale. Il implique le nombre de points d'échantillonnage, la répartition des points d'échantillonnage, la forme du voisinage et sa taille.

Les pixels voisins dans le LBP d'origine sont définis sur un cercle. La principale raison de définir le voisinage de manière isotrope est d'obtenir l'invariance en rotation pour la description de la texture ce qui n'est pas adapté à toutes les applications [Liao et al, 2007]; au contraire, l'anisotropie pourrait également être une caractéristique importante essentiellement dans la reconnaissance faciale. En conséquence [Liao et al, 2007] propose le LBP allongé ou

EBP (*Elliptical Binary Pattern*) dont les pixels voisins se trouvent sur une ellipse. Dans [Nanni et al, 2010] des voisinages généralisés de forme parabolique, hyperbolique et spirale sont aussi proposés. D'autres topologies ont été introduites dans [Petpon et al, 2009] qui utilise des lignes dans les directions verticale et horizontale pour les calculs de LBP.

La figure 2.8 donne deux exemples de EBP, où A, B désignent respectivement l'axe long et l'axe court, et m est le nombre de pixels voisins. Suivant le LBP original, la technique d'interpolation bilinéaire est adoptée pour les pixels voisins qui ne tombent pas exactement sur les pixels. L'opérateur EBP pourrait être tourné autour du pixel central, pour caractériser les structures locales allongées dans différentes orientations.

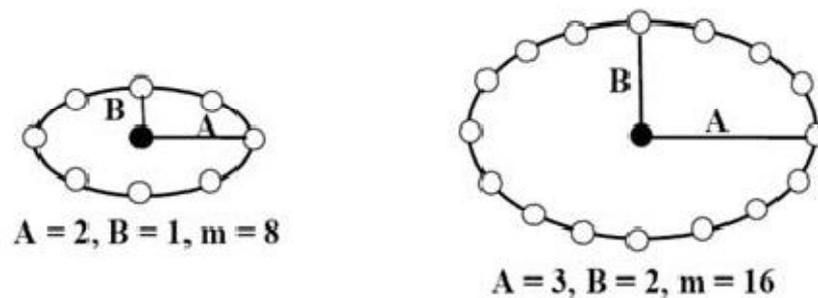


Fig.2.8 Deux exemples d'opérateur EBP [Liao et al, 2007]

Afin de capturer non seulement les microstructures mais aussi les macrostructures, Li et al. [Zhang et al, 2007] proposent le LBP multi-bloc (MB-LBP) qui consiste à comparer la moyenne des intensités des sous-régions voisines au lieu de comparer les pixels. Le LBP original peut être considéré comme un cas particulier du MB-LBP. La Figure 2.9 montre un exemple de MB-LBP, où chaque sous-région se compose de six pixels. La sous-région peut être un rectangle ou un carré.

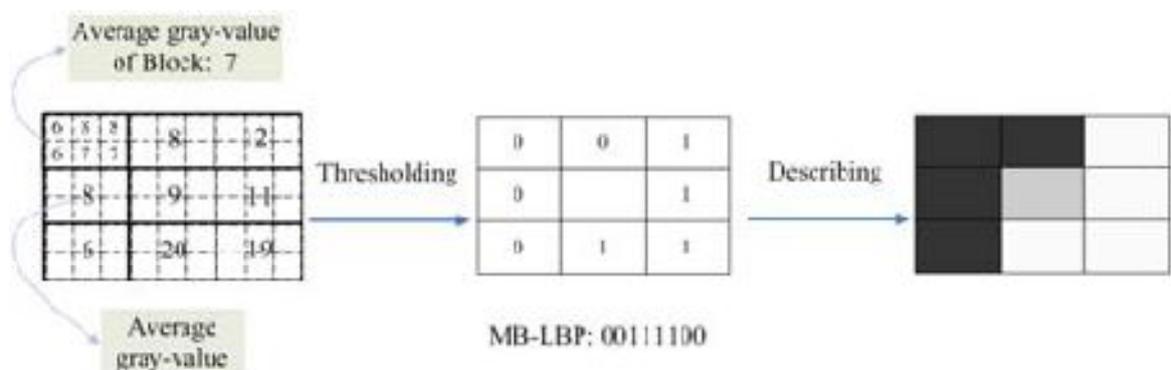


Fig.2.9. Un exemple de l'opérateur MB-LBP [Zhang et al, 2007].

Aussi, la règle de codage du LBP de base se limite au seuillage des pixels voisins par rapport au pixel central, en négligeant les relations entre les pixels du voisinage. Une classe de méthodes a été proposée pour générer le LBP en se basant sur les différences locales des pixels voisins. [Guo et al, 2010b] suggèrent le modèle LBP complet ou CLBP (*Completed Local Binary Pattern*) dans lequel ils utilisent non seulement le signe de la différence entre le pixel central et ses voisins, mais aussi l'amplitude de cette différence et l'amplitude du pixel central. Ils définissent trois opérateurs : CLBP_S, CLBP_M et CLBP_C. Le premier opérateur est le même que $LBP_{P,R}$, le motif binaire est calculé sur la base du signe de la différence du pixel central et ses voisins.

CLBP_M est une configuration binaire de l'amplitude de la différence entre le pixel central et ses voisins, il est défini comme suit :

$$CLBP - M_{P,R} = \sum_{p=0}^{P-1} s(m_p - C) \cdot 2^p \quad (2.8)$$

La valeur m_p représente l'amplitude de la différence entre le pixel central et le voisin x_p , C est un seuil adaptatif représentant la moyenne des valeurs m_p de l'image I.

Enfin, la valeur du pixel central x_C est intégrée dans l'opérateur CLBP_C comme suit :

$$CLBP - C_{P,R} = s(g_C - C_I) \quad (2.9)$$

où C_I est un seuil adaptatif réglé sur la valeur moyenne de l'image I.

La figure 2.10 montre un exemple des données exploitées par CLBP. Le LBP original utilise uniquement le vecteur de signe de la différence (Fig. 2.10(c)) codant le modèle local par une chaîne de 8 bits "11000111" ("- 1" est codé "0").

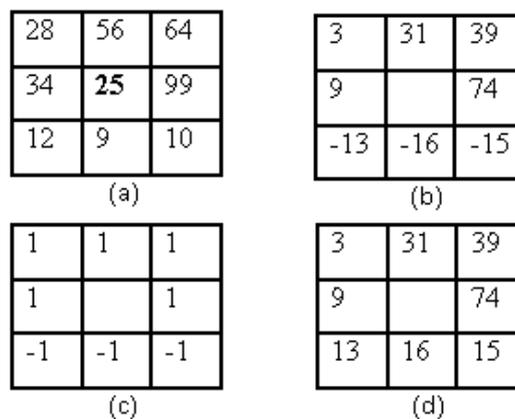


Fig.2.10. (a) voisinage 3×3 (b) différence local (c) signe de la différence (d) magnitude de la différence

L'histogramme conjoint et enchaîné des fréquences des trois modèles CLBP_S, CLBP_M et CLBP_C est utilisé pour caractériser la texture. L'approche CLBP adopte l'uniformité et l'invariance à la rotation utilisée dans le LBP de base. La combinaison jointe des trois descripteurs $CLBP_{R,P}^{riu2}$, $CLBP_C$ et $CLBP_{S,R,P}^{riu2}$ permet d'acquérir de meilleures performances que le LBP traditionnel. Le CLBP est appliquée à la classification des bases de données Outex et Curet. L'efficacité de CLBP a été comparée aux autres méthodes LBP mais aussi à VZ-MR8 et VZ-joint. L'inconvénient majeur de l'opérateur CLBP est le grand nombre de paramètres ce qui le rend difficilement exploitable sur des fenêtres de voisinage de grande dimension. [Liu et al., 2014] proposent alors l'opérateur BRINT (*Binary Rotation Invariant and Noise Tolerant*) pour la classification de texture. Ils utilisent le même principe que l'opérateur CLBP tout en proposant une nouvelle stratégie pour l'étude multi-résolution. Pour chaque résolution, le vecteur des pixels voisins est transformé par moyenne locale de telle sorte que le nombre d'échantillons reste constant. Cette procédure permet non seulement de réduire le nombre d'attributs mais aussi de rendre la méthode plus robuste au bruit. Dans [Heikkilä et al., 2009] le LBP à symétrie centrale (CS-LBP) est proposé en comparant des paires de pixels voisins qui sont dans le même diamètre du cercle et produit ainsi moins d'unités binaires que le LBP original. La différence entre CS-LBP et LBP avec 8 pixels voisins est donné sur la figure (Fig2.11).

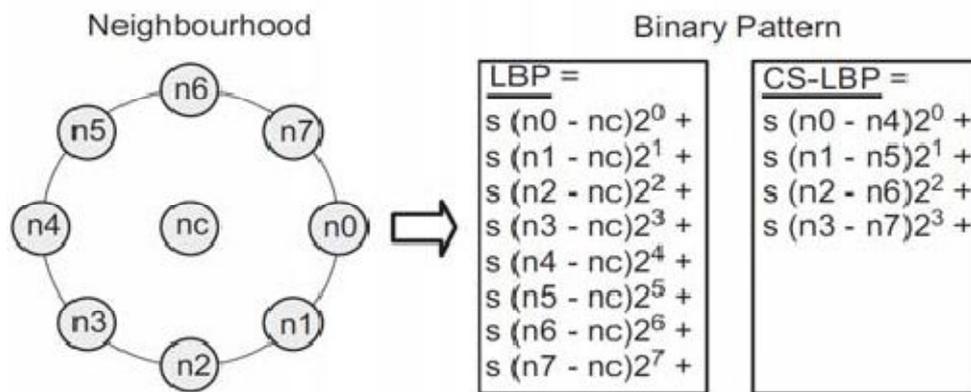


Fig.2.11. Principe de LBP et CS-LBP pour un voisinage de 8 pixels [Heikkilä et al., 2009].

2.3.2. Seuillage

Un des inconvénients de LBP d'origine est que l'opération de seuillage compare directement les valeurs de pixels, ce qui le rend très sensible au bruit. Plusieurs variantes de LBP sont alors proposées en changeant les schémas de seuillage pour améliorer le pouvoir discriminatif et surtout atteindre la robustesse au bruit.

Hafiane et al. [Hafiane et al, 2007] propose le motif binaire médian (MBP) pour la classification de texture. Le motif binaire est calculé par seuillage des pixels par la valeur médiane de tous les voisins. Le pixel central est inclus dans ce procédé. Le MBP est défini comme suit :

$$MBP = \sum_{p=0}^{P-1} s(g_p - g_{med}) 2^p + s(g_c - g_{med}) 2^p, \quad (2.10)$$

où $g_{med} = median(\{g_0, g_1, \dots, g_{P-1}, g_c\})$, s est la même fonction définie dans l'équation (2.2).

Récemment dans [Hafiane et al, 2015], le motif binaire médian adaptatif (AMBP) est proposé. La méthode AMBP est basée sur une fenêtre d'analyse adaptative des modèles locaux. L'idée principale est de convertir une petite parcelle d'image locale à un motif binaire en utilisant une sélection de seuil adaptatif à l'intérieur d'une fenêtre d'analyse de taille variable selon la microstructure locale de la texture. La variabilité de la fenêtre d'adaptation locale est incluse à titre d'information conjointe pour augmenter les propriétés discriminantes. Un nouveau régime multi-résolution est également proposé pour traiter le problème de la résolution de texture. Dans [Ahonen et al, 2009] la méthode LBP Soft (SLBP) est introduite, pour rendre l'approche traditionnelle LBP plus robuste au bruit. Elle permet de générer plusieurs modèles binaires locaux à chaque position de pixel. Cependant, la SLBP est coûteuse en temps de calcul et n'est plus strictement invariante aux changements monotones d'éclairage. Le spectre de textures ou TS (en anglais *Texture Spectrum*) est une approche présentée dans [Wang et He, 1990]. Cette méthode antérieure à LBP propose de décomposer une image en un ensemble d'unités appelées "unités de textures". Chacune d'elles se compose de n éléments ayant pour valeur 0, 1 ou 2. Pour un voisinage de pixels 3×3 (8 voisins), il est possible d'obtenir jusqu'à 3^8 soit 6561 unités de textures différentes. Motivé par TS et LBP, [Tan et al., 2010] a introduit un schéma de codage de partage afin de réduire la dimensionnalité et surtout faire face à la sensibilité au bruit du descripteur LBP. [Tan et al., 2010] propose alors le motif local ternaire

ou LTP (en anglais *Local Ternary Pattern*) . Le principe de LTP est de coder la différence entre le pixel central et ses voisins par 3 valeurs évaluées à l'aide d'un seuil fixe t :

$$LTP_{R,P} = \sum_{p=0}^P S_3(g_p, g_c, t)2^p, \quad (2.11)$$

$$S_3(g_p, g_c, t) = \begin{cases} 1, & g_p \geq g_c + t, \\ 0, & g_c - t \leq g_p < g_c + t \\ -1, & \text{sinon} \end{cases} \quad (2.12)$$

Au lieu d'utiliser un code en base 3 pour coder les trois états, LTP utilise deux codes binaires représentant les composantes positives et négatives du code ternaire (Fig.2.12)

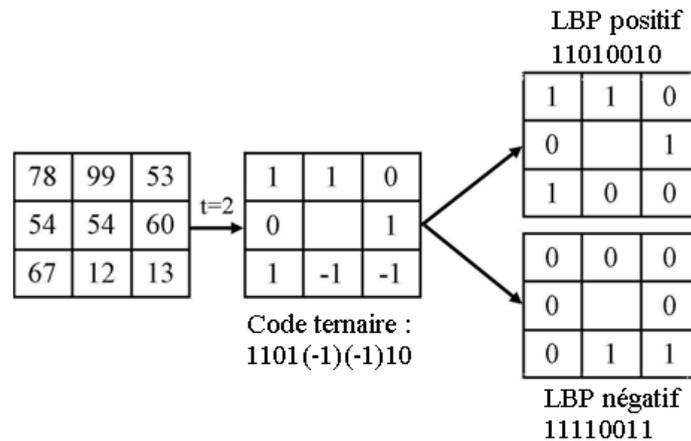


Fig.2.12. Un exemple de la partition d'un code LTP en codes de LBP positifs et négatifs

Les deux codes binaires sont recueillis dans deux histogrammes distincts. Les deux histogrammes sont ensuite concaténés pour former le vecteur caractéristique LTP.

La méthode LQP (Local Quinary Pattern) est introduite dans [Nanni et al, 2010]. Le principe de cette méthode est l'extension des différences locales à cinq valeurs correspondant à deux seuils t_1 et t_2 :

$$LQP_{p,R} = \sum_{p=0}^P s_5(g_p, g_c, t_1, t_2)2^p \quad (2.13)$$

Les deux seuils sont utilisés dans s_5 suivant la fonction (2.14) :

$$s_5(g_p, g_c, t_1, t_2)2^p = \begin{cases} 2, & g_p \geq g_c + t_2 \\ 1, & g_c + t_1 \leq g_p < g_c + t_2 \\ 0, & g_c - t_1 \leq g_p < g_c + t_1 \\ -1, & g_c - t_2 \leq g_p < g_c - t_1 \\ -2, & \text{otherwise} \end{cases} \quad (2.14)$$

Une fois le code LQP généré, il est divisé en quatre modèles binaires en considérant ses composantes positives, nulles et négatives, comme illustré sur la figure suivante. Les quatre histogrammes sont ensuite concaténés pour former le vecteur caractéristique LQP.

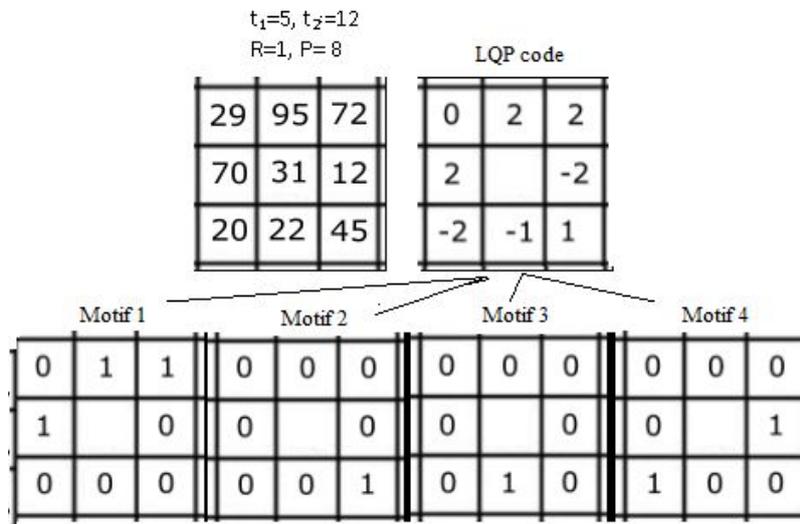


Fig.2.13 Illustration du calcul du code (LQP) en utilisant quatre modèles binaires

[Yang Zhao et al, 2013] propose le motif binaire local complet robuste ou CRLBP (*Completed Robust Local Binary Pattern*). Dans cette méthode, la valeur de chaque pixel central dans une fenêtre locale 3×3 est remplacée par le niveau de gris moyen. Pour rendre la méthode CRLBP plus robuste et plus stable, un niveau de gris local pondéré est introduit pour prendre la place de la valeur de gris traditionnel du pixel central. Les résultats expérimentaux obtenus à partir de quatre bases de données de textures représentatives montrent que la méthode proposée est robuste au bruit. [Ret al, 2013] propose la méthode NRLBP (*Noise Resistance Local Binary Pattern*) pour préserver les structures locales d'images en présence de bruit. Le principe de cette méthode est de coder la différence du pixel de petite valeur, puis déterminer sa valeur sur la base des autres bits du code LBP. Ainsi un mécanisme de correction d'erreur est développé pour récupérer les motifs d'images déformées.

2.3.3. Codage et regroupement

Le problème majeur de l'opérateur LBP original est la croissance exponentielle du nombre des modèles dépendamment de la taille du voisinage P soit 2^P modèles distincts, conduisant à un faible pouvoir discriminant et à un stockage important. Il est clair que les modèles locaux ne soient pas tous significatifs pour modéliser les caractéristiques des textures. Ainsi, les chercheurs ont tenté d'obtenir des paramètres compacts et plus discriminant en identifiant les modèles les plus significatifs. [Liao et al, 2009] propose l'opérateur DLBP (*Dominant Local Binary Pattern*) utilisant les modèles les plus fréquents pour capturer des informations descriptives de texture. Ils suggèrent de choisir les modèles dominants dans les textures au lieu des modèles uniformes.

A la différence des modèles uniformes, le nombre de motifs n'est pas constant et déterminé par le choix des modèles dominants qui représentent 80 % de motifs entiers :

$$k = \underset{k}{\operatorname{argmin}} \left(\frac{\sum_{p=0}^{k-1} H[p]}{\sum_{p=0}^{2^P-1} H[p]} \geq 80\% \right), \quad (2.15)$$

Où P est le nombre de voisins dans $LBP_{R,P}$, H est l'histogramme des modèles triés dans un ordre décroissant. L'histogramme final utilisé pour la classification est $H = [0 \dots k]$.

Différents tests de classification sur les bases de données Brodatz, Outex, Meastex et CURet ont démontré l'invariance en rotation et surtout la robustesse au bruit de l'opérateur DLBP [Liao et al, 2009] comparativement à plusieurs méthodes.

[Zhou et al, 2008] ont développé une extension de la méthode LBP basée sur les motifs "non uniformes" NELBP (*Novel Extended Local Binary Pattern*). Ils proposent de classifier et de combiner les modèles "non uniformes" en se basant sur l'analyse de leur structure et de la probabilité d'occurrence. Le nouvel opérateur utilise les informations de textures contenues dans les "modèles locaux non uniformes", qui sont ignorées par les opérateurs de LBP classiques, ce qui rend la méthode proposée plus robuste au bruit. Dans [Zhao et al., 2012] l'opérateur LBC (*Local Binary Count*) est proposé, en codant uniquement le nombre de voisins dont la valeur est plus élevée que le pixel central, Les performances de CLBP ont été évaluées sur trois bases de données comparativement aux méthodes LBP notamment CLBP.

La méthode utilisant les cooccurrences des motifs LBP (CoLBP) est aussi proposée. Cette approche est inspirée des matrices de cooccurrence (GLCM) [Haralick, 1973]. L'idée générale est de considérer la probabilité conjointe de paires de LBP à certains déplacements relatifs. [Qi et al, 2012] propose l'application de GLCM sur $LBP_{R,P}^{ri}$ et $LBP_{R,P}^{u2}$ respectivement

dans [Qi et al, 2014] [Qi et al, 2012]. Une extension pour l'étude multi-resolution et multi-orientation est aussi proposée dans [Qi et al, 2013].

2.3.4. Combinaison avec d'autres approches

Pour améliorer la capacité discriminative de LBP, l'histogramme conjoint de $LBP_{R,P}^{riu2}/VAR_{P,R}$ est proposé dans [Ojal et al., 2002b]. L'opérateur $VAR_{P,R}$ est défini comme suit :

$$VAR_{P,R} = \frac{1}{P} \sum_{p=0}^{P-1} (g_p - \mu)^2, \quad \mu = \frac{1}{P} \sum_{p=0}^{P-1} g_p \quad (2.16)$$

Guo et al. [Guo et al, 2010a] soutiennent que l'opération VAR(Variance) produit des valeurs continues et que la quantification correcte de ces valeurs est un défi. Alors, ils proposent le LBP Variance (LBPV) dans lequel l'histogramme de $LBP_{R,P}$ est pondéré par la valeur $VAR_{P,R}$ au lieu d'utiliser l'histogramme conjoint. L'histogramme de $LBPV_{P,R}$ est calculé comme suit (2.9) :

$$LBP_{P,R}(k) = \sum_{i=1}^N \sum_{j=1}^M W(LBP_{R,P}(i, j), k), \quad k \in [0, K], \quad (2.17)$$

Où N et M sont respectivement le nombre de lignes et de colonnes. Le poids W est calculé comme suit :

$$W = \begin{cases} VAR_{P,R}(i, j), & \text{si } LBP_{P,R} = k \\ 0, & \text{sinon} \end{cases} \quad (2.18)$$

Dans [Heikkilä et al., 2009] le LBP à symétrie centrale (CS-LBP) est combiné avec le SIFT (*Scale Invariant Feature Transform*) [Klare et al, 2013]. Dans [Liao et al, 2009], le filtre de Gabor et le LBP sont combinés. Le LBP capture la structure de texture locale, alors que les filtres de Gabor extraient des informations de textures globales. Ahonen et al. [Ahonen et al, 2009] proposent une approche nommée caractéristiques de l'histogramme LBP Fourier (LBP-HF) qui combine le LBP et la Transformée de Fourier discrète ou DFT (*Discret Fourier Transform*). Wang et al. Combinent l'histogramme de Gradients ou HOG (*Histogram Of Gradient*) avec LBP [Wang et al, 2009]. [Liao et al, 2009] utilisent les réponses aux filtres de Gabor normalisés (NGF) du spectre de fréquence pour améliorer le taux de classification. Hussain et Triggs combinent le LTP et le LBP [Hussain et al, 2010] . [Guo et al, 2010c] ont

introduit le LBP adaptatif (ALBP) en exploitant des caractéristiques statistiques directionnelles, en particulier la moyenne et l'écart type de la différence absolue locale pour améliorer l'efficacité de la classification du LBP. Les expériences effectuées sur la base de données de textures CURET montrent que la méthode ALBP peut améliorer de manière significative la précision de la classification du LBP. Klare et Jain exploitent la combinaison de LBP et de SIFT pour la reconnaissance de visage hétérogène [Klare et al, 2013]. Khellah [Khellah, 2011] introduit la structure de voisinage dominant ou DNS (*Dominant Neighborhood Structure*) pour extraire des caractéristiques globales et invariantes par rotation et compléter ainsi le modèle LBP. Les caractéristiques de textures proposées sont obtenues par la génération d'une carte globale estimée représentant la similarité mesurée entre une image pixel donnée et ses voisins à l'intérieur d'une certaine fenêtre de recherche. Khellah [Khellah, 2011] effectue plusieurs expérimentations et montrent l'efficacité du DNS et sa tolérance au bruit comparativement à DLBP et LBP.

2.4. Discussions

La méthode LBP est l'une des méthodes les plus performantes et les plus utilisées pour la classification de textures. Plusieurs travaux récents proposent des variantes de cette méthode afin d'améliorer son efficacité et sa robustesse au bruit tout en réduisant le nombre de descripteurs.

Bien que des progrès significatifs soient réalisés, la plupart des variantes LBP ont des inconvénients. Nous citons essentiellement : sa sensibilité au bruit [Liao et al, 2009][Khellah, 2011][Liu et al, 2014] limitation des méthodes LBP à trois échelles, ce qui les empêche de capturer des informations de texture à longue portée [Liao et al, 2009][Khellah,2011], [Hafiane et al,2015] . Aussi, parfois les méthodes LBP classent différents motifs dans un même modèle, ce qui peut réduire leur capacité discriminative Liao et al, 2009] [Zhao et al., 2013].

Nous proposons dans le chapitre suivant, une nouvelle méthode qui permet d'extraire de descripteurs de texture compacts, multi-échelles et invariants en rotation. Notre méthode est évaluée sur plusieurs bases de données, ses performances sont comparées à la méthode LBP.

CHAPITRE 3

Codage de Rang Régional

3.1. Préambule

Dans ce chapitre, nous proposons une nouvelle méthode d'analyse de textures utilisant le codage de rang régional. Dans une première partie, nous présentons le principe de cette méthode qui est basée sur le calcul du rang de niveau de gris d'un pixel sur un voisinage régional. Cette méthode est appliquée à la segmentation d'images texturées. Dans une seconde partie, une version améliorée de cette méthode est présentée. Huit paramètres invariants en rotation sont alors définis. Des tests de classification d'images sur les bases de données Outex et CURET seront effectués. Les différents résultats ainsi obtenus seront comparés à la méthode LBP.

3.2. Codage de rang régional

Contrairement au codage LBP, le codage de rang régional attribue à chaque pixel une valeur en fonction du rang de son niveau de gris sur un voisinage régional.

Dans ce paragraphe, les voisins d'un pixel x , désignent l'ensemble des pixels qui lui sont 8-connexes. On notera $B_8(x)$ l'ensemble des 8 pixels voisins de x incluant le pixel x lui-même. De plus, si on considère un ensemble $A \subset E$, l'ensemble des voisins de A sera l'ensemble

$$V = \{y \in \{A^c / B_8(y) \cap A \neq \emptyset\}\} \quad (3.1)$$

Notion de plateau

Un plateau est un ensemble 8-connexes de pixels de même niveau de gris.

Soit x un pixel de l'image I , on désigne par $ptI(x)$ le plateau contenant x .

La figure 3.1 représente une partition d'une image, nous pouvons apercevoir deux plateaux dont le niveau de gris est égale à 7 (couleur grise) entourés par l'ensemble des voisins (couleur

bleue). L'un des plateaux se compose de cinq pixels entourés par seize voisins, le deuxième plateau est composé de trois pixels entourés par neuf pixels voisins.

12	11	5	4	7	4	4	12
8	4	7	2	2	1	0	11
2	7	7	7	4	2	7	7
11	6	7	4	6	6	7	1
4	6	4	5	7	6	5	8

Fig.3.1. Représentation de deux plateaux et leurs voisins

3.2.1. Principe

Le codage de rang régional s'effectue comme suit :

Pour chaque pixel x de l'image, nous déterminons d'abord l'ensemble des pixels formant le plateau contenant x noté $pt(x)$.

Le codage de rang régional attribue la même valeur à tous les pixels du plateau $pt(x)$, en fonction des niveaux de gris de ses voisins. L'ensemble des voisins du plateau $pt(x)$ noté : $V(pt(x))$ est défini comme suit :

$$V(pt(x)) = y \in \{(pt(x))^c / B_8(y) \cap (pt(x)) \neq \emptyset\} \quad (3.2)$$

Le niveau de gris du plateau $pt(x)$ et l'ensemble de ses voisins sont classés dans l'ordre croissant. Nous attribuons à chaque plateau $pt(x)$ le rang correspondant à la position du niveau de gris de x dans la liste ordonnée. Le rang d'un pixel x est calculé suivant les équations (3.3) et (3.4).

Soit $V(pt(x)) = \{x_1, x_2, \dots, x_V\}$, l'ensemble des voisins du plateau $pt(x)$ et I est l'image en niveaux de gris .

$$rang(x) = \sum_{v=1}^V F(x, x_v) \quad (3.3)$$

$$F(x, x_v) = \begin{cases} 1, & \text{si } I(x) > I(x_v) \\ 0, & \text{sinon} \end{cases} \quad (3.4)$$

Si $\|V\|$ est le nombre de voisins, les valeurs possibles de rang sont incluses dans l'intervalle discret et fermé $[0, \dots, \|V\|]$. Si $\text{rang}(x) = 0$, le plateau contenant x ou $\text{Pt}(x)$ est un minimum régional. Par contre, si $\text{rang}(x) = \|V\|$, $\text{Pt}(x)$ est un maximum régional.

En d'autres termes : Un maximum (resp. minimum) régional M d'une image I est un plateau sans voisin de niveau de gris plus élevé (resp. moins élevé) que celui du plateau.

L'intérêt des extrema régionaux sur les extrema locaux est qu'ils reflètent de manière plus générale la structure globale de l'image sans être influencés par la forme d'un quelconque voisinage B .

Le codage d'extrémalité régional proposé dans [Bonnevay, 1997] est basé uniquement sur la valeur du rang (rang), l'information niveau de gris est alors complètement perdue. Par contre dans le codage proposé, le rang retrouvé ne sera pas utilisé directement pour coder la texture, car il varie en fonction du nombre des voisins. Cette valeur est alors normalisée, nous considérons le rapport $P_0 = \text{rang} / \|V\|$. Aussi, pour calculer le code final du rang régional, nous utilisons le rapport P_0 ainsi que le niveau de gris du plateau [Ouslimani, 2014] suivant le tableau ci-dessous. Les valeurs de niveaux de gris et le rapport P_0 sont quantifiés respectivement sur 8 et 2 valeurs. Il y a 16 valeurs possibles de code correspondant aux différentes combinaisons. La texture peut être caractérisée par l'histogramme des fréquences des 16 codes.

		Valeurs du rapport P_0	
		$[0- 0.5[$	$[0.5-1]$
Valeurs de niveaux de gris	0 -32	0	1
	32 -64	2	3
	64 -96	4	5
	96 -128	6	7
	128- 160	8	9
	160 -192	10	11
	192 -224	12	13
	224-255	14	15

Table 3.1. Codage de rang régional en fonction de P_0 et les niveaux de gris.

3.2.2. Conservation de l'information niveau de gris

Nous appliquons le codage d'extrémalité régionale [Bonnevay, 1997] et le codage de rang régional sur un ensemble d'images texturées. Comme nous pouvons le constater sur la figure 3.2, les deux codages conservent l'information structurelle de textures. Néanmoins, le codage de rang régional fait ressortir l'information texture tout en gardant les informations en niveaux de gris (Fig.3.2.c) contrairement au codage d'extrémalité régionale (Fig.3.2.b) où cette information est complètement perdue. Aussi, nous constatons que le codage de rang régional accentue quelques détails présents dans les textures.

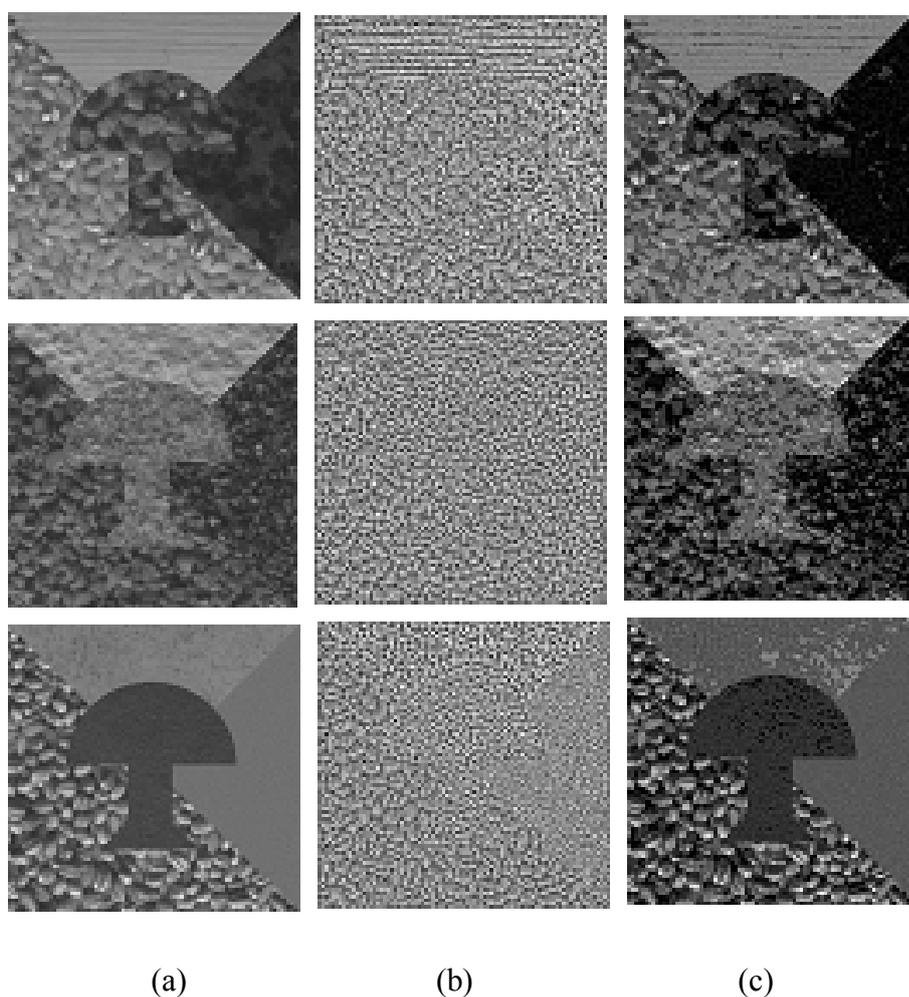


Fig. 3.2. Images texturées codées, (a) images test, (b) codage d'extremalié régionale, (c) codage de rang régional.

3.2.3. Segmentation d'images texturées

Le codage de rang régional est utilisé pour la segmentation de quatre images tests. Ces images sont composées de plusieurs régions texturées extraites de l'album de Brodatz [Brodatz, 1965]. La taille de chaque image est de 256×256 et a 256 niveaux de gris. La méthode de segmentation adoptée est la classification des paramètres par l'algorithme des K moyennes [Duda et al, 2001].

Les paramètres de textures sont extraits à partir de l'image codée. Le codage de rang régional est calculé sur des fenêtres d'analyse de taille 17×17 centrées sur chaque pixel traité. Ainsi, chaque pixel est caractérisé par l'histogramme de fréquence des 16 codes. Les résultats obtenus sont montrés sur la figure 3.4. Pour évaluer les résultats de la segmentation, nous estimons le taux de bonne classification CA en utilisant des images de référence présentées sur la figure 3.3. Le taux de bonne classification (CA) dépend du nombre de pixels correctement classifiés, il est évalué par la formule suivante

$$CA = \frac{t}{N \times M} \times 100 \quad (3.5)$$

Où t est le nombre de pixels correctement classés, et $N \times M$ est la taille de l'image considérée.

	Image 1	Image 2	Image 3	Image 4
CA (%)	98,19	97,33	93,46	87,62

Table 3.2. Taux de bonne classification des images segmentées

Comme le montrent les images de la figure 3.4, les résultats de segmentation sont satisfaisants. Le taux de classification estimé atteint respectivement 98,19 % et 97,33 % pour la première et la seconde image. Malgré la présence de certaines régions mal classées, la troisième image présente un résultat satisfaisant avec un taux de bonne classification égal à 93,46 %. Cependant, le taux de bonne classification de la quatrième image est de seulement 87,62 %. Ainsi, nous pouvons apercevoir plusieurs régions mal classées, ce qui peut être expliqué par le fait que le codage de rang régional a capté des détails présents dans la texture, comme nous pouvons le remarquer sur l'image code correspondante (Fig.3.4.b). Nous pensons que cette même propriété de codage de rang régional peut être bien exploitée dans des applications comme la détection d'un défaut ou d'une anomalie présente sur une texture. Les seize codes ont une bonne capacité discriminative pour caractériser les textures. Le codage de rang régional pourrait être exploré dans des applications de segmentation d'images.



Fig .3.3. Images de références

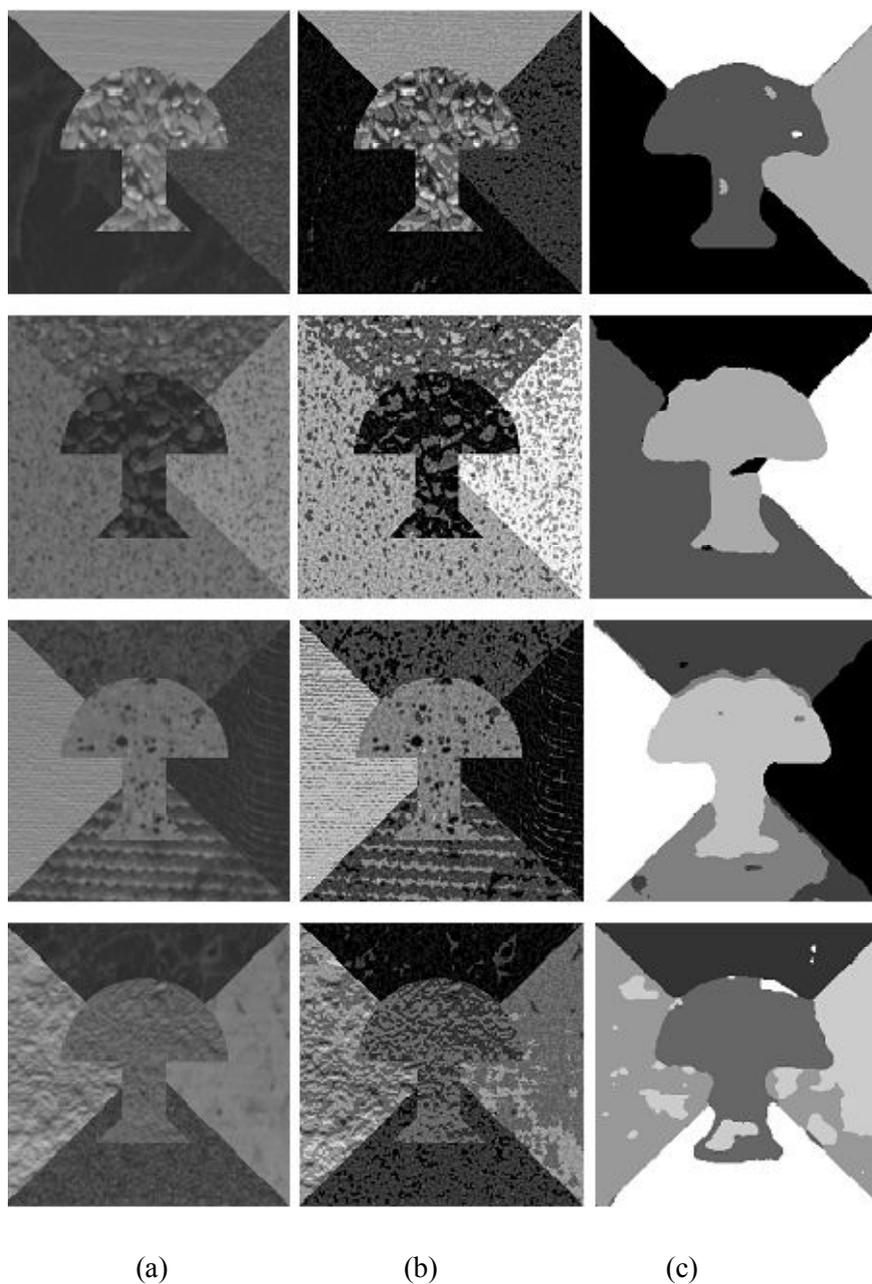


Fig.3.4. Résultats de segmentation, (a) images test, (b) codage de rang régionale, (c) images segmentées

3.3. Codage RRC

Le but de cette deuxième partie est de présenter de nouveaux attributs discriminatifs et invariants en rotation pour la classification des textures. Afin d'extraire des motifs de texture plus performants, des modifications sont effectuées sur le codage de rang régional. Tout d'abord, pour acquérir plus de précision, le rapport régional P_0 est quantifié sur quatre valeurs et non pas deux. Deuxièmement, en s'inspirant des travaux de [Guo and al, 2010], [Liu et al, 2014], l'information niveau de gris est exploitée après un seuillage global. Troisièmement, pour une étude multi-résolution, différentes dimensions de l'espace de recherche du voisinage sont exploitées. L'histogramme des caractéristiques à la même taille pour les différentes échelles et peuvent être facilement condensés pour améliorer la précision de la classification de textures. Pour calculer le codage final, la valeur correspondante au seuillage global du niveau de gris du pixel traité est combinée avec P_0 conformément à la table 3.3. Ainsi, huit attributs invariants en rotation sont extraits. Nous appelons cette version du codage de rang régional, le codage RRC [Ouslimani et al, 2015a].

		P_0			
		[0- 0.25[[0.25-0.5[[0.50-0.75[[0.75- 1]
$I(i,j) < \mu$	1	2	3	4	
$I(i,j) \geq \mu$	5	6	7	8	

Table 3.3. Le code RRC calculé en fonction du niveau de gris et le rapport P_0

μ est la moyenne des niveaux de gris de l'image I et $I(i,j)$ est le niveau de gris du pixel de coordonnées (i,j)

Après le calcul du code RRC pour chaque pixel de coordonnées (i, j) , l'image I est représentée par l'histogramme $H_{RRC(I)}$ donné par les équations (3.6) et (3.7)

$$H_{RRC(I)} = \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M f(RRC(i,j), l) , l \in [1,8] \quad (3.6)$$

$$f(x,y) = \begin{cases} 1 & , si x = y \\ 0 & , sinon \end{cases} \quad (3.7)$$

3.3.1. Invariance en rotation

Pour illustrer l'invariance en rotation du codage RRC, nous effectuons quelques tests. Nous considérons trois images (D1, D2 et D98) extraites de l'album de Brodatz [Brodatz, 1965] ainsi que leurs versions tournées suivant trois orientations : 15° , 75° et 90° (Fig.3.5).

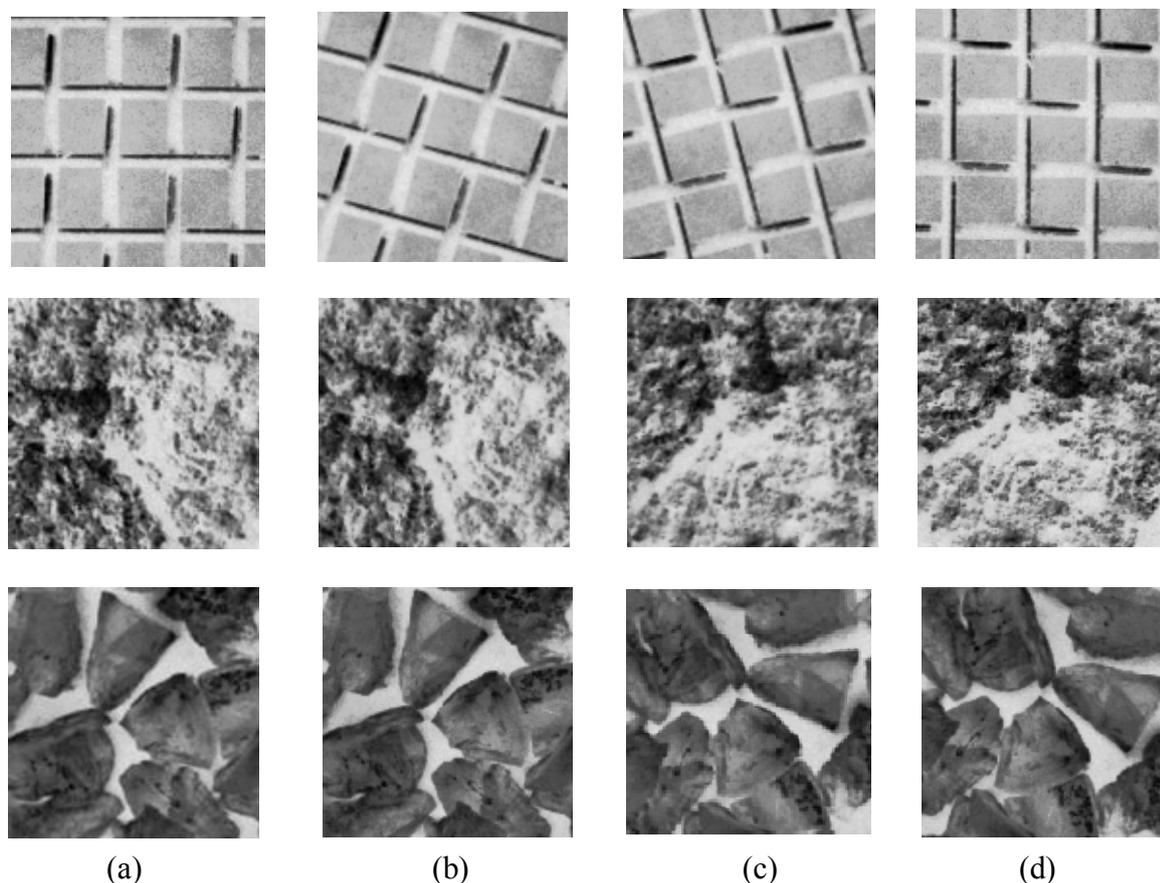


Fig.3.5. Exemples d'images texturées et leurs versions tournées (a)Rotation 0° (b) rotation 15° (c) rotation 75° (d) rotation 90°

La figure 3.6 représente la valeur absolue de la différence des histogrammes RRC extraient à partir de la texture D1 de l'album de Brodatz et ses versions tournées. Nous remarquons que quel que soit les orientations considérées, la différence entre les deux vecteurs est quasi nulle.

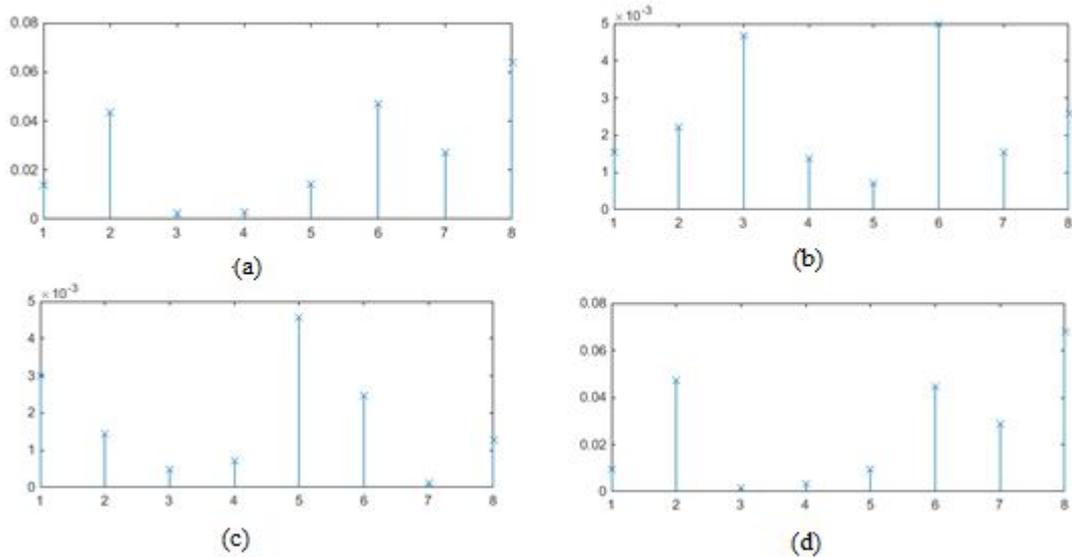


Fig.3.6. La valeur absolue de la différence entre les histogrammes RRC de l'image originale D1 et ses versions tournées (a) 0° et 15° (b) 15° et 75° (c) 0° et 90° (d) 75° et 90°

La table 3.4 représente l'erreur quadratique moyenne (RMSE) entre les caractéristiques d'une même image mais de rotation différentes θ_1 et θ_2 . Le RMSE est calculé sur trois textures de l'album de Brodatz représentées sur la figure 3.5. Les angles de rotation considérés sont 0°, 15°, 75° et 90°. Dans tous les cas, l'erreur quadratique moyenne est de l'ordre de 0. Ce qui montre l'invariance en rotation des attributs RRC.

$$RMSE_{\theta_1/\theta_2} = \sqrt{\frac{\sum_{i=1}^8 (H_{\theta_1}(i) - H_{\theta_2}(i))^2}{8}} \quad (3.8)$$

H_{θ_1} et H_{θ_2} représentent respectivement les histogrammes LDRC d'une image tournée suivant deux angles θ_1 et θ_2 .

	0°/15°	0°/75°	0°/90°	15°/75°	15°/90°	75°/90°
D1	0.0204	0.0268	0.0019	0.0065	0.0223	0.0288
D2	0.0124	0.0059	0.0002	0.0009	0.0071	0.0124
D98	0.0062	0.0093	0.009	0.0031	0.0053	0.0084

Table.3.4. L'erreur quadratique moyenne entre les caractéristiques extraites à partir du RRC correspondant à deux rotations de la même image de texture

3.3.2. Etude multi-résolution :

L'étude multi-résolution est indispensable pour acquérir plus d'informations, ainsi différentes échelles sont utilisées. L'ensemble des pixels de même niveau de gris et l'ensemble des voisins sont définis par extension de l'espace de recherche autour de chaque pixel. Nous utilisons des fenêtres carrés de taille $S \times S$ pixels. Ainsi, le voisinage exploité dans la section précédente correspond à $S = 3$.

La figure (3.7.b) représente une partition d'une image. En utilisant un espace de recherche de taille 5×5 pixels ($S=5$), nous pouvons apercevoir l'ensemble de pixels de même niveau de gris égal à 7 (couleur grise) entouré par l'ensemble de ses voisins (de couleur bleue).

La même procédure de codage est utilisée. Pour chaque fenêtre de recherche, l'ensemble de pixels et les voisins correspondants sont définis et le codage de rang régional est calculé suivant la table 3.3. Le nombre de motifs reste constant et égal à 8 quel que soit l'échelle considérée. RRC_S correspond aux attributs extraits en utilisant un voisinage de taille $S \times S$.

L'utilisation d'espace de recherche de différentes dimensions peut améliorer le pouvoir discriminatif des caractéristiques de textures. Ainsi, les caractéristiques extraites de chaque échelle sont enchaînées dans un seul histogramme de dimension $n \times 8$ où n représente le nombre d'échelle. Dans cette étude, nous nous limitons à trois échelles ($n=3$).

1	4	2	10	8	4	4	10	2
4	5	8	4	3	2	7	23	4
17	10	5	3	8	8	11	2	3
1	7	7	7	6	11	2	8	9
5	8	18	10	11	15	7	10	11
12	1	5	7	8	7	6	7	10
11	2	7	8	4	1	10	2	3
2	6	4	10	10	12	11	10	5
1	4	5	8	10	2	8	6	5

(a)

1	4	2	10	8	4	4	10	2
4	5	8	4	3	2	7	23	4
17	10	5	3	8	8	11	2	3
1	7	7	7	6	11	2	8	9
5	8	18	10	11	15	7	10	11
12	1	5	7	8	7	6	7	10
11	2	7	8	4	1	10	2	3
2	6	4	10	10	12	11	10	5
1	4	5	8	10	2	8	6	5

(b)

Fig.3.7. Représentation d'un ensemble de pixels (en gris) et l'ensemble de ses voisins (en bleu) en utilisant (a) espace de recherche de taille 3×3 , (b) espace de recherche de taille 5×5 .

3.3.3. Tests et Résultats :

Les performances de RRCs sont évaluées via la classification des textures. Nous utilisons les bases de données Outex_TC10, Outex_TC12 et la base CURET. Les détails concernant ces bases sont donnés dans le chapitre 1. Ces différentes bases de données

présentent un grand nombre de classes de textures avec des rotations significatives et des changements d'éclairage.

Le codage de rang régional RRCs est calculé suivant trois fenêtres de recherche ($S=3, 5$ et 7). Cette méthode est comparée aux motifs LBP uniforme et invariant en rotation $LBP_{R,P}^{riu2}$ pour $(R,P) \in \{(1,8), (2,16), (3,24)\}$.

La classification des textures est effectuée en utilisant la méthode du plus proche voisin (1-NN). Ainsi, chaque échantillon de la base test sera affecté à la classe correspondant au modèle le plus proche. La métrique utilisée est la distance Chi deux (χ^2) définie dans l'équation (1.7) du chapitre 1.

Aussi, la classification utilisant les SVM est introduite. Nous exploitons la librairie libsvm [Chang et al, 2011]. Nous utilisons la fonction noyau RBF (*Radial basic function kernel*) ou Noyau gaussien, cette fonction est définie par l'équation suivante :

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma}\right) \quad (3.9)$$

x_i et x_j représentent deux vecteurs caractéristiques.

Deux paramètres de la fonction RBF sont à fixer :

- Le facteur de régularisation C qui est une constante positive qui permet de contrôler l'importance de l'erreur que l'on s'autorise par rapport à la taille de la marge. Plus C est important, moins d'erreurs sont autorisées.
- Le paramètre σ permet de régler la largeur de la gaussienne. En prenant un σ grand, la similarité d'un exemple par rapport à ceux qui l'entourent sera assez élevée, alors qu'en prenant un σ tendant vers 0, l'exemple ne sera similaire à aucun autre.

Pour déterminer les valeurs de C et σ , une sorte de sélection de modèle (recherche de paramètres) doit être effectuée. Le but est d'identifier les meilleures valeurs de C et σ de sorte que le classificateur puisse prédire avec précision des données inconnues (c'est-à-dire des données de test). Après plusieurs tests, nous avons fixés C à 10^4 et σ à 0,1.

Les détails concernant les SVM sont donnés dans l'annexe A.

3.3.3.1. Résultats obtenus sur la base de données Outex :

La table 3.5 présente les différents résultats obtenus utilisant différentes combinaisons. D'après les résultats obtenus, nous pouvons faire les remarques suivantes :

- Le taux de classification de RRC_3 dépasse celui de $LBP_{1,8}^{riu2}$ considérablement et la méthode RRC acquiert la meilleure performance sur la base TC12-001.
- Les performances de RRC_S diminuent avec l'augmentation de S ceci peut s'expliquer par le fait que les informations acquises sont plus globales mais moins précises.
- L'histogramme enchaîné sur différentes échelles augmente la performance des deux méthodes. Le meilleur taux obtenu par la méthode proposée dépasse les **95 %** pour TC10 et atteint respectivement **88,96 %** et **89,17 %** pour les TC12-000 et TC12 -001.
- La méthode LBP dépasse légèrement la méthode proposée pour TC10 et TC12-000. Notons que pour la méthode LBP, le nombre de paramètres utilisé sur trois échelles est de 54 (10+ 18 + 26) soit deux fois plus le nombre de paramètres utilisé dans RRC_S qui est de 24 (3 x 8).
- Néanmoins, d'autres tests impliquant la combinaison de RRC avec la moyenne et la variance locale peut améliorer les résultats de classification. A titre d'exemple, la combinaison $RRC_3/RRC_7/VAR_3/MOY_3$ a permis d'atteindre un taux de classification de **97.81%** pour Outex TC10. Par contre, la combinaison $RRC_3/RRC_7/RRC_9/MOY_3$ a permis d'atteindre des taux respectifs de **90.02%** et **93.13%** pour Outex TC12-000 et TC12-001. Cette amélioration est considérable par rapport aux résultats obtenus par RRCs.

Méthodes		Taux de classification		
		TC10	TC12-000	TC12-001
$LBP_{1,8}^{riu2}$		84.81	65.46	63.68
$LBP_{2,16}^{riu2}$		89.40	82.26	75.20
$LBP_{3,24}^{riu2}$		95.07	85.04	80.78
$LBP_{8,1}^{riu2}/LBP_{2,16}^{riu2}$		93.20	84.32	79.35
$LBP_{8,1}^{riu2}/LBP_{3,24}^{riu2}$		97.65	90.76	85.25
$LBP_{2,16}^{riu2}/LBP_{3,24}^{riu2}$		96.40	87.82	82.89
$LBP_{1,8}^{riu2}/LBP_{2,16}^{riu2}/LBP_{3,24}^{riu2}$		97.21	89.21	84.32
1-NN	RRC ₃	92.79	84.42	84.14
	RRC ₅	85.23	76.97	76.55
	RRC ₇	80.96	71.48	70.28
	RRC ₃ / RRC ₅	93.88	87.94	87.52
	RRC ₃ / RRC ₇	95.16	88.77	88.94
	RRC ₅ / RRC ₇	89.84	81.78	83.43
	RRC ₃ / RRC ₅ / RRC ₇	95.10	88.96	89.17
SVM	RRC ₃	91.64	81.45	79.30
	RRC ₅	84.29	72.93	73.35
	RRC ₇	83.95	71.45	73.45
	RRC ₃ / RRC ₅	92.34	81.45	79.30
	RRC ₅ / RRC ₇	91.24	81.45	79.30
	RRC ₃ / RRC ₇	94.27	72.93	73.35
	RRC ₃ / RRC ₅ / RRC ₇	94.03	81.45	79.30

Table 3.5. Taux de classification (%) retrouvé sur les bases TC10 et TC12

3.3.3.2. Résultats obtenus sur la base de données CURET

Nous exploitons les mêmes sous-ensemble d'images utilisés dans [Liu et al, 2014], [Khellah, 2011] et [Varma et al, 2009] soit un ensemble de 61 classes de textures, chacune avec 92 échantillons, 46 images par classe sont choisies au hasard dans la phase

d'apprentissage et 46 images restantes sont choisies pour les tests. Pour retrouver le taux de classification moyen et l'écart type, nous effectuons 100 partitionnements aléatoires des 92 échantillons. La table 4.6 représente les résultats obtenus. Pour cette base de données, la méthode proposée permet d'atteindre un taux de classification modéré de l'ordre de 85 %. Nous pensons que ceci est principalement dû au fait qu'il existe des variations d'échelle affines dans la base de données CURET alors que RRC est un opérateur proposé pour la rotation et l'invariance du niveau de gris et qu'il a une capacité limitée pour traiter l'échelle et la variation affine.

Méthodes	Taux de classification (%)
$LBP_{8,1}^{riu2} / LBP_{16,2}^{riu2} / LBP_{24,3}^{riu2}$	95.84± 0.82
RRC ₃	79.43±1.05
RRC ₅	69.96± 1.21
RRC ₇	65.58 ±1.36
RRC ₃ / RRC ₅ / RRC ₇	85.00 ±1.10

Table 3.6. Taux de classification (%) retrouvé sur la base CURET.

3.4. Discussions

Dans ce chapitre, nous avons présenté de nouveaux descripteurs de textures basés sur le codage de rang régional. Cette méthode attribue à chaque pixel une valeur en fonction du rang de son niveau de gris sur un voisinage régional.

Dans une première partie, nous avons proposé le codage de rang régional, cette méthode a été appliquée à la segmentation d'images texturées. Les résultats de segmentation obtenus sont très satisfaisants. Le codage de rang régional peut être ainsi exploité dans des applications de segmentation d'images.

Dans une seconde partie, nous avons proposé de nouveaux attributs discriminatifs et invariants par rotation pour la classification des textures. Afin d'extraire des motifs de textures plus performants, des modifications sont effectuées sur le codage de rang régional. Huit descripteurs invariants par rotation sont extraits à partir du codage RRC. Pour l'étude multi-

résolution, différentes dimensions de l'espace de recherche sont exploitées. Ainsi, les descripteurs extraits sur chaque échelle sont enchaînés en un histogramme de dimension $n \times 8$ où n est le nombre d'échelle. Comme le montrent les résultats expérimentaux effectués sur la base de données Outex, les taux de classification obtenus avec la méthode RRC_3 dépassent ceux obtenus avec la méthode $LBP_{1,8}^{riu2}$ pour les bases TC10 et TC12. Le codage RRCs produit le meilleur taux de classification pour TC12-001 pour les différentes combinaisons. L'histogramme représentant les descripteurs extraits en utilisant différentes échelles permet d'améliorer les résultats. La combinaison de RRCs avec d'autres paramètres tels que la variance ou la moyenne permet d'améliorer son pouvoir discriminatif considérablement. Néanmoins, des études plus exhaustives utilisant les SVM pourraient améliorer les résultats de classification.

Le prochain chapitre portera sur une nouvelle technique de codage robuste au bruit. De plus, une extension à cette méthode est proposée pour l'invariance en rotation. Les différents résultats obtenus seront exposés, commentés et comparés aux différentes variantes du LBP.

CHAPITRE 4

Codage Directionnel

4.1. Préambule

Dans ce chapitre, deux méthodes d'analyse de textures sont proposées. Le codage local de rang directionnel (LDRC) et le codage directionnel (DC). Ces deux méthodes sont basées sur le signe de la différence entre le pixel central et ses voisins. Pour l'étude multi-résolution, les deux codages sont calculés en modifiant la taille de la fenêtre d'analyse autour du pixel traité. Le nombre de voisins est limité à huit dont chacun représente une moyenne locale des niveaux de gris. La taille de l'histogramme des paramètres est ainsi maintenue réduite et constante.

Afin d'évaluer les deux méthodes, plusieurs tests portant sur la classification des textures sont effectués en utilisant six bases de données à savoir : Brodatz, CURET, Outex TC10, Outex TC12-000, Outex TC12-001 et KTH-Tips2b. Les méthodes proposées sont comparées à des variantes de la méthode LBP ainsi qu'à d'autres méthodes d'analyse de textures.

Dans une première partie de ce chapitre, le codage LDRC est exposé. Différents tests effectués pour évaluer la robustesse au bruit de cette méthode sont ensuite présentés. La seconde partie présente le codage DC en évaluant l'invariance en rotation de ces descripteurs sur plusieurs bases de données.

4.2. Le codage de rang directionnel local

Le codage de rang directionnel local ou LDRC (*Local Directional Rank Coding*) est basé sur le signe de la différence entre le niveau de gris du pixel central et ses 8 voisins. Contrairement aux techniques présentées précédemment notamment les méthodes LBP, le codage LDRC permet d'extraire quatre informations directionnelles, représentant les rangs du niveau de gris du pixel central calculés sur quatre orientations sur une fenêtre de voisinage de taille 3x3 pixels. Les quatre rangs sont ensuite utilisés pour obtenir le code final [ouslimani et al, 2015b][ouslimani et al,2016]. Le codage LDRC fait ressortir les extrema de

gris sur chaque direction. Cette information est ainsi exploitée pour extraire les différents motifs.

4.2.1. Principe du codage LDRC

Pour effectuer ce codage, nous utilisons une suite de voisinages composée de quatre éléments. Chaque élément est défini par un point origine placé en X et deux pixels voisins alignés entourant ce dernier dans une direction particulière comme le montre la figure 4.1.b. Cette suite balaye les quatre directions principales : 0° , 45° , 90° et 135° .

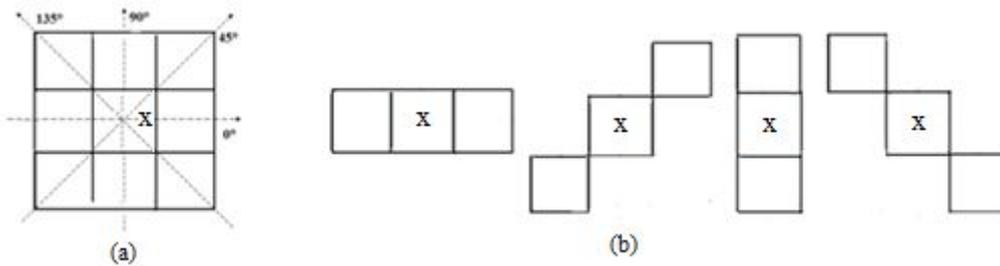


Fig.4.1. (a) pixel central x et ses 8 voisins, (b) Suite de voisinage utilisée pour le codage

Pour chaque voisinage, nous classons les trois pixels voisins par ordre croissant de leurs niveaux de gris. Nous attribuons au pixel central le rang correspondant à la position de son niveau de gris dans la liste ordonnée. Pour chaque orientation θ° , la valeur associée au pixel central est calculée suivant les équations (4.1) et (4.2).

Soit x le pixel central, pi ($i = 1, 2$) sont ses deux pixels adjacents suivant la direction θ° . $I(x)$ et $I(pi)$ sont les niveaux de gris correspondants.

$$Cod_\theta(x) = \sum_{i=1}^2 F(I(x), I(pi)); \quad \theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ \quad (4.1)$$

$$F(a, b) = \begin{cases} 1, & \text{si } a \geq b \\ 0, & \text{sinon} \end{cases} \quad (4.2)$$

Pour chaque orientation, la valeur Cod_θ associée est comprise entre 0 et 2. Notons que si $cod_\theta(x) = 0$, $I(x)$ représente le niveau de gris minimum selon la direction θ . En revanche, il représente le niveau de gris maximum si $cod_\theta(x) = 2$. Donc, la valeur de $cod_\theta(x)$ représente

une information directionnelle sur les extrema de niveau de gris et il serait intéressant de l'exploiter afin d'extraire des caractéristiques de textures.

Ainsi, pour trouver le code final, la chaîne de code ternaire $(\text{cod}_{135} \text{ cod}_{90} \text{ cod}_{45} \text{ Cod}_0)_3$ représentant les quatre valeurs calculées dans les quatre directions est convertie en son équivalent décimal suivant l'équation (4.3). Ainsi, l'opérateur LDRC produit 81 (3^4) motifs.

$$\text{LDRC}(x) = \text{cod}_0 \times 3^0 + \text{cod}_{45} \times 3^1 + \text{cod}_{90} \times 3^2 + \text{cod}_{135} \times 3^3 \quad (4.3)$$

Le LDRC est invariant au changement monotone de niveau de gris, puisque ce code est basé sur l'interaction spatiale dans une localité donnée.

Un exemple du codage LDRC est illustré sur la figure 4.2. Le niveau de gris du pixel central est classé suivant l'équation (4.1). Ainsi, le rang du pixel central dans le sens horizontal est égal à 0 ($\text{Cod}_0 = 0$), pour la direction diagonale (135°), le rang est égal à 2 ($\text{cod}_{135} = 2$), le code final affecté au pixel central est égal à 69 suivant l'équation (4.3).

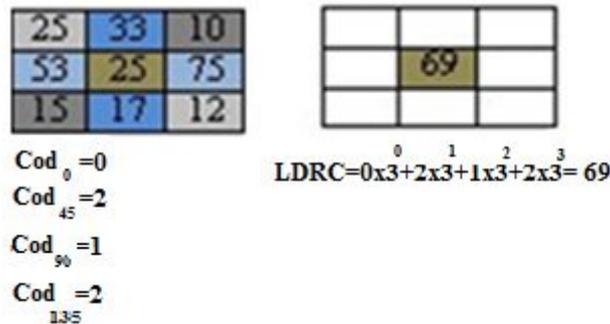


Fig.4.2. Illustration du codage LDRC

Après le calcul du code LDRC pour chaque pixel de coordonnées (i, j) , l'image I est représentée par l'histogramme $H_{\text{LDRC}(I)}$ donné par les équations (4.4) et (4.5).

$$H_{\text{LDRC}(I)} = \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M f(\text{LDRC}(i, j), l), \quad l \in [0, L], \quad (4.4)$$

$$f(x, y) = \begin{cases} 1, & \text{si } x = y \\ 0, & \text{sinon} \end{cases} \quad (4.5)$$

N et M représentent respectivement le nombre de lignes et de colonnes de l'image I, L est la valeur max de LDRC.

4.2.2. Etude multi- résolution

Pour l'étude multi-résolution, l'histogramme des descripteurs extraits à partir du codage LDRC est concaténé sur différentes échelles. Pour limiter l'augmentation de la taille de l'histogramme avec l'échelle et surtout extraire des motifs robustes au bruit, nous utilisons la même stratégie proposée dans [Liu et al, 2014]. Mais, dans LDRC, les voisins du pixel central x_c sont considérés sur une fenêtre carrée et non pas sur un voisinage circulaire.

Le nombre P de voisins est fixé à être un multiple de huit, donc $P = 8 \times q$ où q est un entier positif. Les voisins du pixel central x_c sur le voisinage de la taille $S \times S$ pixels sont représentés par le vecteur $\underline{x}_{s,8q} = [x_{s,8q,0}, \dots, x_{s,8q,8q-1}]$.

Pour chaque fenêtre, le vecteur des voisins $\underline{x}_{s,8q}$ est transformé par moyenne locale en un autre vecteur $\underline{y}_{s,q}$ sachant que $\underline{y}_{s,q} = [y_{s,q,0}, \dots, y_{s,q,7}]$. Le nombre de voisins dans $\underline{y}_{s,q}$ est égal à huit et les valeurs de $\underline{y}_{s,q}$ sont calculées suivant l'équation (4.6)

$$y_{s,q,i} = \frac{1}{q} \sum_{k=0}^{q-1} x_{s,8q,(qi+k)}, \quad i = 0, \dots, 7 \quad (4.6)$$

Par exemple : $y_{5,2,0} = (x_{5,16,0} + x_{5,16,1})/2$ et $y_{7,3,1} = (x_{7,24,3} + x_{7,24,4} + x_{7,24,5})/3$.

La figure 4.3 illustre cette transformation.

Ainsi, Le code LDRC est calculé sur différentes résolutions spatiales en modifiant la taille de la fenêtre $S \times S$. Sur chaque fenetre, la transformation de l'ensemble de voisins est effectuée suivant l'équation (4.6). Sur chaque échelle, le code LDRC est calculé par rapport au pixel central suivant les équations (4.1), (4.2) et (4.3) (voir figure 4.4).

L'histogramme des paramètres LDRC est concaténée sur de multiples échelles; cet histogramme est de taille $n \times 81$, n étant le nombre d'échelles.

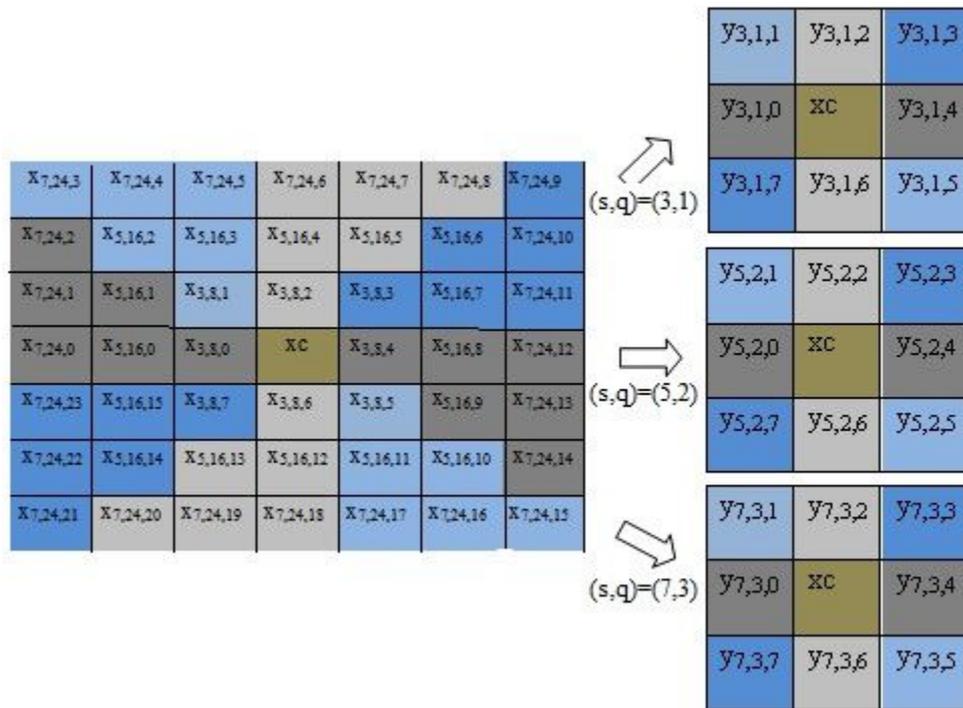


Fig.4.3. Représentation d'un exemple exploitant 3 échelles et illustration de la transformation du voisinage $x_{s,q}$ au voisinage $y_{s,q}$. Les différentes couleurs correspondent aux 4 voisinages directionnels utilisés par l'opérateur LDRC.

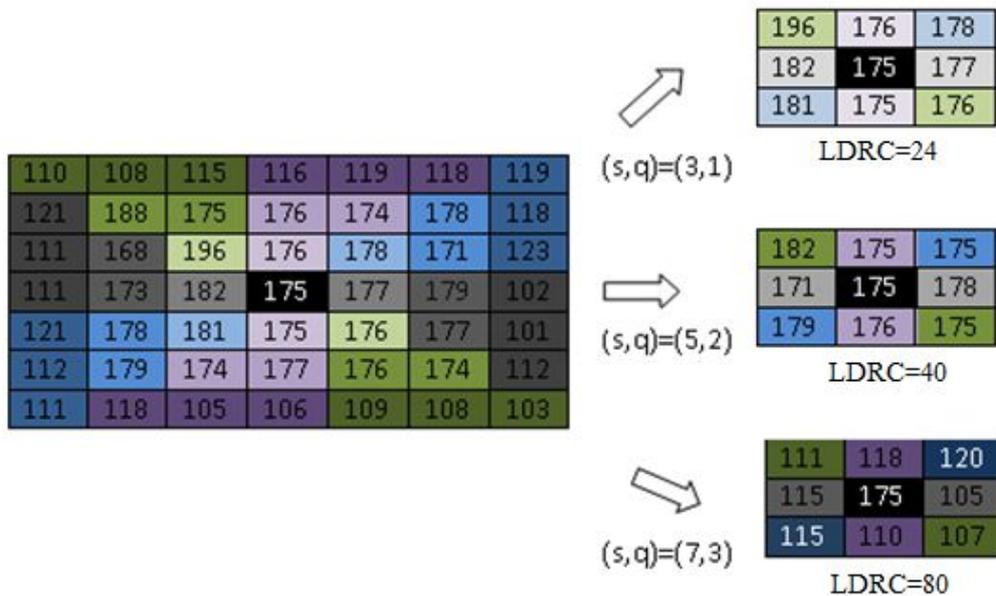


Fig.4.4. Un exemple de calcul du codage LDRC sur trois échelles.

4.2.3. Tests et Résultats

Pour évaluer la performance de la méthode proposée, nous utilisons trois bases de données : Brodatz, CURET et KTH- Tips2b décrites en détail dans le chapitre 1.

Comme dans les travaux présentés dans [Khellah,2011][Guo et al, 2010][Liu et al, 2014] [Vama et al, 2005] et [varma et al, 2009], la classification est effectuée en utilisant la méthode du plus proche voisin (1-NN). Chaque image échantillon est affectée à la classe correspondante au modèle le plus proche en utilisant la distance χ^2 définie dans l'équation (1-7) du chapitre 1.

Pour l'étude multi-résolution, l'histogramme des descripteurs LDRC est concaténé sur neuf échelles en modifiant la taille de la fenêtre de 3×3 à 19×19 . Cet histogramme est noté MSn où n représente le nombre d'échelles.

Nous effectuons deux tests : dans une première étape, la méthode proposée est appliquée sur les images originales. Puis, dans une seconde étape, nous évaluons la robustesse de la méthode proposée dans des conditions bruyantes où les images de textures originales sont corrompues par un bruit additif gaussien de moyenne nulle et d'écart type déterminé par différentes valeurs du rapport (signal/bruit) ou SNR. Les SNR utilisés sont 100, 30, 15, 10, 5 et 3 correspondant respectivement à 20db, 14.78db, 11.76db, 10db, 7db et 5db. Un exemple de bruit gaussien ajouté à trois textures (Fig. 4.5) illustre l'impact de la densité de bruit sur la qualité visuelle perceptive de la texture.

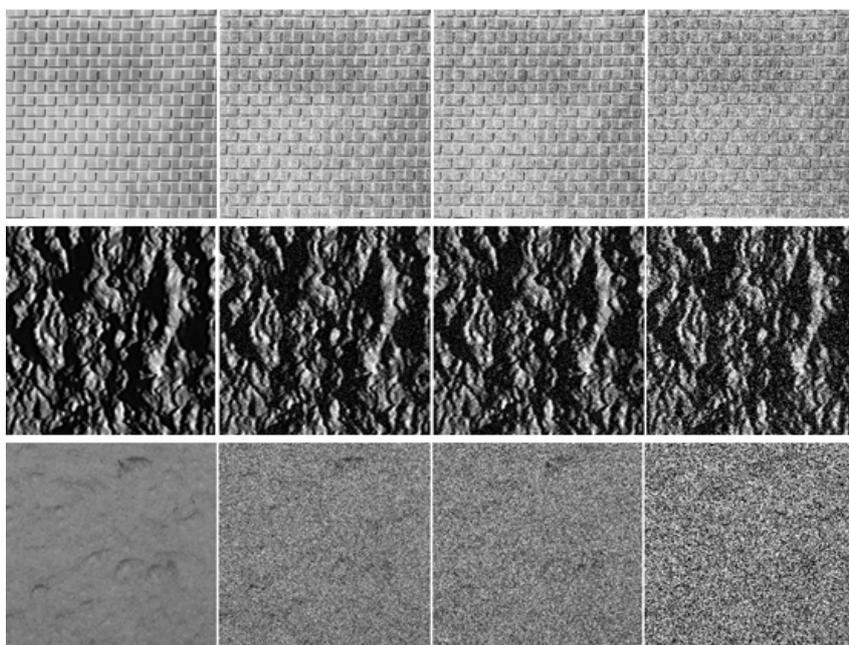


Fig.4.5 Exemple d'images texturées (Brodatz D1, CURET #30, kthtip2#9) corrompues par un bruit gaussien de différentes intensités.

Le LDRC est comparé aux méthodes: DLBP [Liao et al, 2009], DNS + LBP [Khellah, 2011], BRINT [Liu et al, 2014] et LCP [Song et al, 2014]. Ces méthodes sont considérées dans notre étude comme étant les approches les plus récentes basées sur LBP et qui revendiquent la robustesse au bruit.

4.2.3.1. Résultats obtenus sur la base de Brodatz

Nous utilisons 24 classes de textures homogènes extraites de l'album de Brodatz dont 25 images échantillons sont extraites pour chaque classe. Nous sélectionnons 13 échantillons pour l'apprentissage et les 12 autres pour les tests. Le taux de classification moyen et l'écart type sont calculés après 100 partitionnements aléatoires des échantillons d'apprentissage et de test.

Les tests sont d'abord effectués sur les images originales. La figure 4.6 illustre les performances de la classification utilisant LDRC en fonction du nombre d'échelles. Tous les résultats obtenus dépassent 99 %, mais le meilleur taux de classification est retrouvé en utilisant trois échelles (MS3).

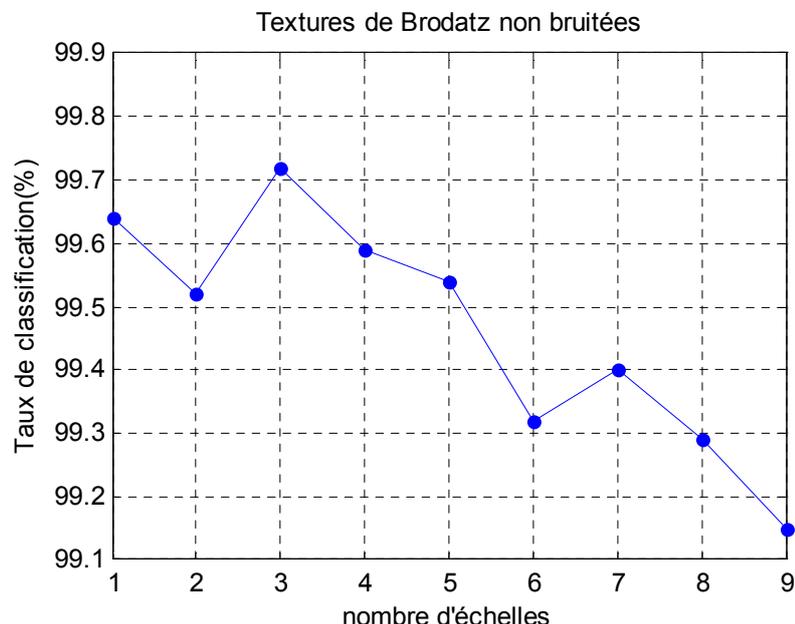


Fig.4.6. Taux de bonne classification en fonction du nombre d'échelles

La table 4.1 compare les performances de la classification de la méthode proposée avec BRINT (*Binary Rotation Invariant Noise Tolerant*), CLBP (*Completed Local Binary Pattern*) et DLBP (*Dominant Local Binary Pattern*). Toutes les méthodes testées atteignent un très

haut taux de classification. Cependant, même si BRINT effectue le meilleur taux avec 100 % de précision et qui est légèrement supérieur à la méthode proposée, la taille de l'histogramme des caractéristiques BRINT et LDRC sont respectivement 432 (144×3) et 243 (81×3) cela signifie que BRINT nécessite deux fois plus de paramètres que notre méthode LDRC .

Méthodes	Taux de classification (%)
DLBP+NGF[Liao et al, 2009]	99.54
CLBP [Guo et al, 2010]	99.72±0.33
BRINT [Liu et al, 2014]	100±0.00
LDRC(MS3)	99.72±0.32

Table 4.1. Taux de classification retrouvés sur la base de Brodatz

Pour tester les performances de la méthode proposée en présence de bruit, les images originales sont corrompues par un bruit gaussien avec différentes valeurs du SNR. La figure 4.7 représente les taux de classification de la méthode proposée pour différentes valeurs du SNR et en fonction du nombre d'échelles. Le LDRC montre de bons résultats de classification. L'histogramme obtenu en utilisant plusieurs résolutions augmente la robustesse au bruit. A partir de six échelles (MS6), le taux de classification dépasse les 80% pour tous les SNR. Le taux de la classification le plus élevé est acquis pour 9 échelles (MS9), pour le bruit moins intense (SNR = 30), le maximum est acquis pour seulement trois échelles (MS3) et atteint 97,68 %. Le LDRC donne de bons résultats dans des situations extrêmement bruyantes.

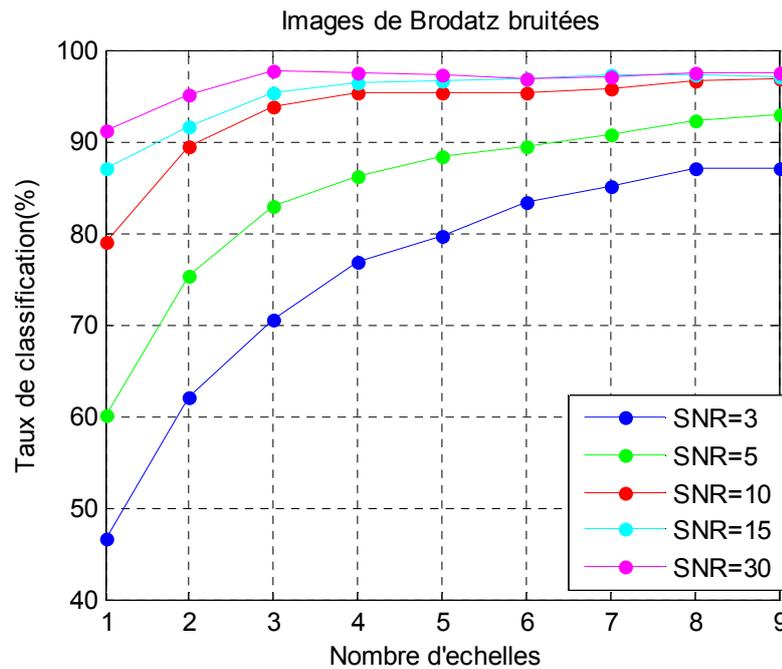


Fig.4.7. Taux de classification obtenus sur la base de Brodatz bruitées en fonction du nombre d'échelles et pour différents SNR

La table 4.2 compare les taux de classification obtenus par LDRC utilisant neuf échelles (MS9) avec les méthodes DLBP et BRINT.

Le LDRC est plus robuste au bruit, la performance relative augmente à mesure que le niveau de bruit augmente. La méthode proposée réalise de meilleures performances que DLBP et BRINT d'autant plus que le classificateur SVM utilisé dans DLBP et BRINT est plus robuste que 1-NN mis en œuvre dans ce travail. Aussi, le nombre de paramètres BRINT utilisé est de 1296 (144x9) soit environ deux fois la taille des paramètres LDRC 729 (81x9). Le taux de classification obtenu avec DLBP surpasse celui obtenu avec notre méthode pour SNR = 30, mais sous des bruit plus intenses (SNR <30), sa performance est nettement réduite comparée à la méthode proposée.

Méthodes	Taux de classification (%)				
	SNR=30	SNR=15	SNR=10	SNR=5	SNR=3
DLBP+NGF[Liao et al, 2009]	99.31±0.00	95.77±0.97	92.33±4.65	83.84±4.48	NAN
BRINT [Liu et al, 2014]	97.81±0.59	96.42±0.88	95.36±0.96	91.00±1.41	85.56±1.62
LDRC	97.98±0.78	97.23±0.84	96.79±0.96	92.91±1.35	87.07±1.31

Table 4.2. Taux de classification obtenus sur la base de Brodatz bruitée en fonction de différents SNR

4.2.3.2. Résultats obtenu sur la base CURET :

À l'origine, la base de données CURET contient 205 images pour chaque classe de textures acquises à différents points de vue et d'éclairage. Cela rend la base de données beaucoup plus difficile à classifier que sur les images de Brodatz. Nous utilisons les mêmes sous-ensembles d'images utilisés dans [Liu et al, 2014], [Khellah, 2011] et [Varma et al, 2009] soient 61 classes de texture. Chacune avec 92 échantillons. Les 46 images par classe sont choisies au hasard dans la phase d'apprentissage et 46 images restantes sont choisies pour les tests. Pour retrouver le taux de classification moyen et l'écart type, nous effectuons cent partitionnements aléatoires des 92 échantillons.

La figure 4.8 représente les résultats de la classification en fonction du nombre d'échelle. Nous remarquons que le taux de bonne classification augmente avec le nombre d'échelles et atteint le maximum égal à 97,41% en utilisant huit échelles (MS8).

La table 4.3 dresse la liste des résultats de la classification des différentes méthodes. Les méthodes BRINT et VZ-Joint sont légèrement plus efficaces que LDRC. Ceci est principalement dû à la légère variation de rotation lors de l'acquisition de la base de données CURET, tandis que le but visé en proposant LDRC est la robustesse au bruit. Le LDRC a une capacité limitée pour traiter l'invariance par rotation. Une amélioration de cette méthode permettant l'extraction de paramètres invariants par rotation est proposée dans la deuxième partie de ce chapitre.

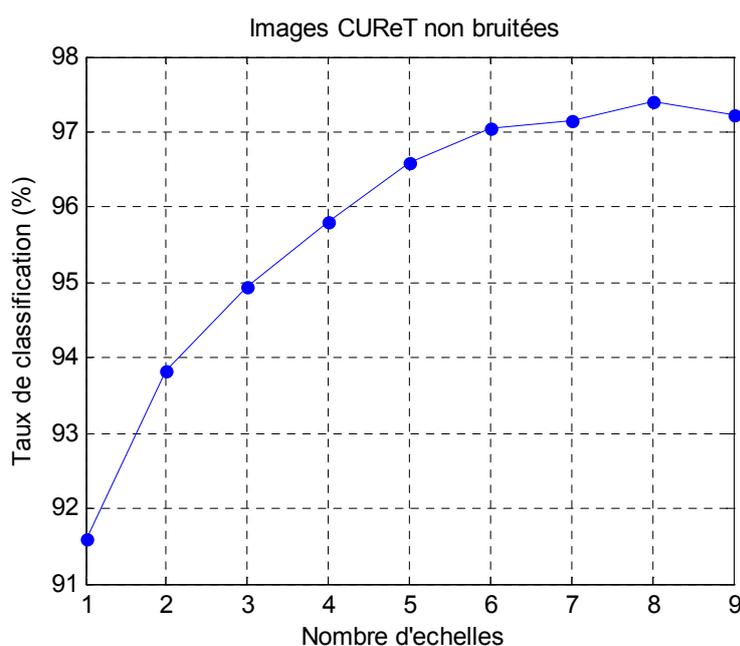


Fig.4.8. Taux de bonne classification des textures de la base CURET en fonction des nombres d'échelles

Méthodes	Taux de classification (%)
VZ- MR8 [Vama et al, 2005]	97.43
VZ-Joint [Varma et al, 2009]	98.03
CLBP [Guo et al, 2010]	97.39
BRINT [Liu et al, 2014]	97.86
LDRC (MS8)	97.41

Table 4.3. Taux de classifications(%) retrouvées sur la base CURET

Les images de la base CURET sont ensuite corrompues par un bruit gaussien additif avec différentes valeurs de SNR. La figure 4.8 montre la performance de la classification pour différents SNR et ceci en fonction du nombre d'échelles. Les résultats démontrent bien la robustesse de la méthode proposée au bruit. L'histogramme multi-résolution augmente les performances de LDRC. Les taux de classifications calculés suivant différents SNR dépassent les 95% à partir de MS5. La méthode LDRC donne de bons résultats dans des situations extrêmement bruyantes. Les résultats montrent clairement de légères dégradations des performances même dans le cas des bruits extrêmement élevés. Les meilleurs taux de classification sont obtenus en utilisant neuf échelles (MS9).

La table 4.4 énumère les résultats de la classification des différentes méthodes en présence de différents niveaux du bruit gaussien. La performance de la méthode proposée est comparée au DNS, BRINT et LCP. La méthode proposée permet d'obtenir une amélioration d'environ 15% par rapport à DNS et environ 3% de plus que LCP.

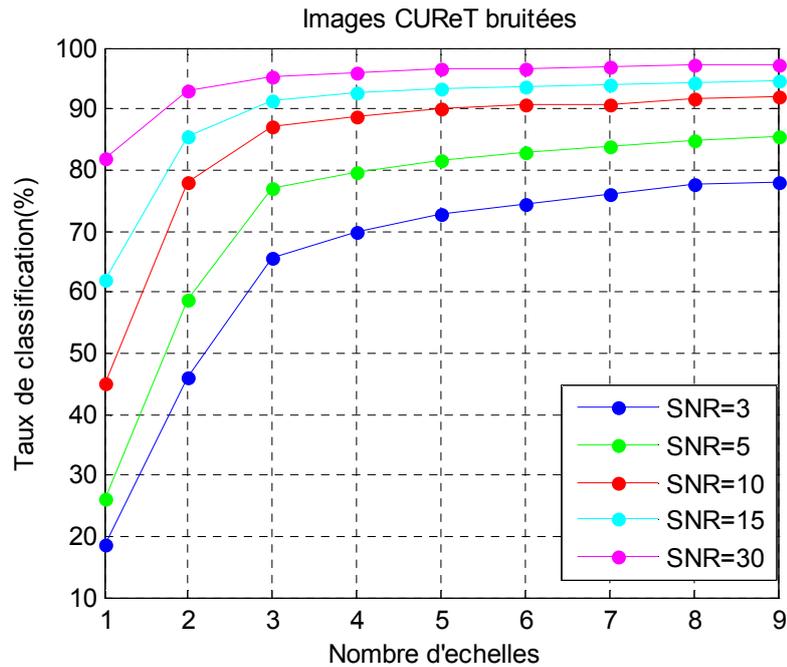


Fig.4.9 .Taux de classification obtenus sur la base de CURET bruitées en fonction du nombre d'échelles et pour différents SNR

Méthodes	Taux de classification (%)				
	SNR=30	SNR=15	SNR=10	SNR=5	SNR=3
DNS+LBP _{24,3} [Khellah,2011]	87.37±0.76	83.28±1.20	81.04±1.19	72.71±0.97	NAN
LCP[Song et al, 2014]	93.88	91.64	89.07	83.35	NAN
BRINT[Liu et al, 2014]	95.39± 0.84	93.69± 1.09	90.92± 2.55	86.11 ±1.17	80.45 ±1.05
LDRC(MS9)	97,19±0.48	94.62±0.66	91.87±0.72	85.43±0.86	78.10±0.9

Table 4.4.Taux de classifications obtenues sur la base de CURET bruitées pour différents SNR

4.2.3.3 Résultats obtenus sur la base KTH-Tips2b

Dans cette expérience, nous suivons le même schéma utilisé dans [caputo et al, 2005], [Chen et al, 2010] [Liu et al, 2014]. Plus précisément, les ensembles 1, 2, 3 et 4 de la base KTH-Tips2b sont combinés en couple (apprentissage, test). Rappelons que chaque ensemble contient 108 images. Nous utilisons trois ensembles d'images pour l'apprentissage et un ensemble d'images pour le test. Les combinaisons possibles sont alors (2U3U4,1), (1U3U4,2), (1U2U4,3) et (1U2U3,4). Nous considérons la moyenne des taux de classification retrouvés sur les quatre combinaisons.

La figure 4.10 montre bien la performance de la classification en fonction du nombre d'échelles. Le LDRC multi-résolution améliore le taux de bonne classification. Le meilleur taux est atteint en utilisant six échelles (MS6).

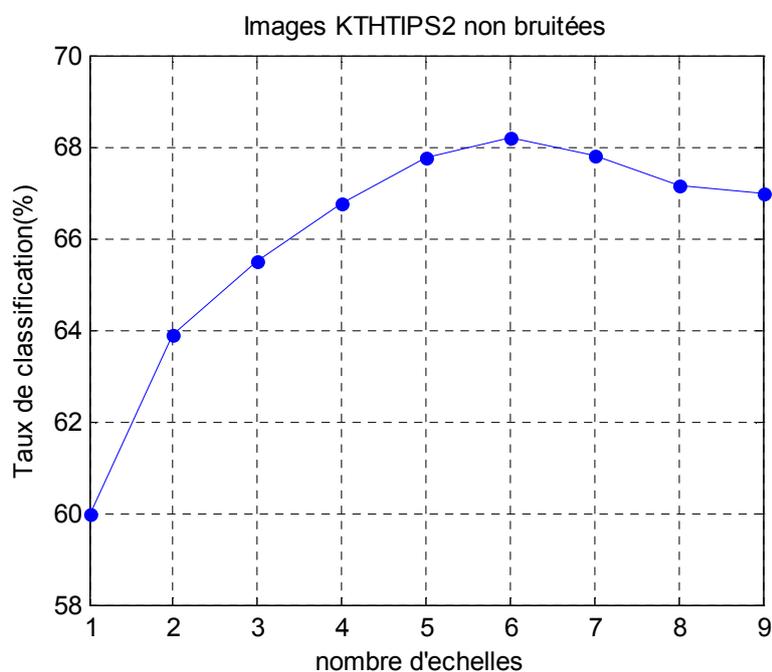


Fig.4.10. Taux de bonne classification des textures de la base KTH-TIPSE2b en fonction des nombres d'échelles

La table 4.5 compare les meilleurs taux de classification obtenus par LDRC avec d'autres méthodes assez récentes sur la base de données KTH-Tips2b. La méthode BRINT est légèrement plus efficace que le LDRC. Mais il convient de noter que LDRC utilise seulement (81×6) paramètres comparativement à BRINT qui nécessite 9×72 caractéristiques. D'autant plus que le procédé utilisé dans LDRC est beaucoup plus simple à mettre en œuvre que celui utilisé dans BRINT.

La figure 4.11 représente la performance de classification de la méthode proposée appliquée à KTH-Tips2b en présence d'un bruit gaussien. L'histogramme multi-résolution augmente les performances de la méthode proposée. Pour un $SNR > 10$, le maximum est atteint pour six échelles (MS6), pour un bruit plus élevé ($SNR \leq 10$) le taux maximum est atteint sur neuf échelles (MS9). Nous pouvons constater une légère réduction des taux de classification. Pour $SNR = 3$, le taux de classification a atteint 61,30 % soit une réduction de 10 % par rapport au taux atteint sur les images originales qui est de 68,18%.

Méthodes	Taux de classification (%)
DLBP [Liu et al,2017]	61.72
BRINT (MS9) [Liu et al, 2014]	70.3
LDRC(MS6)	68.18

Table 4.5. Taux de classification(%) obtenus sur la base KTH-TIPS2

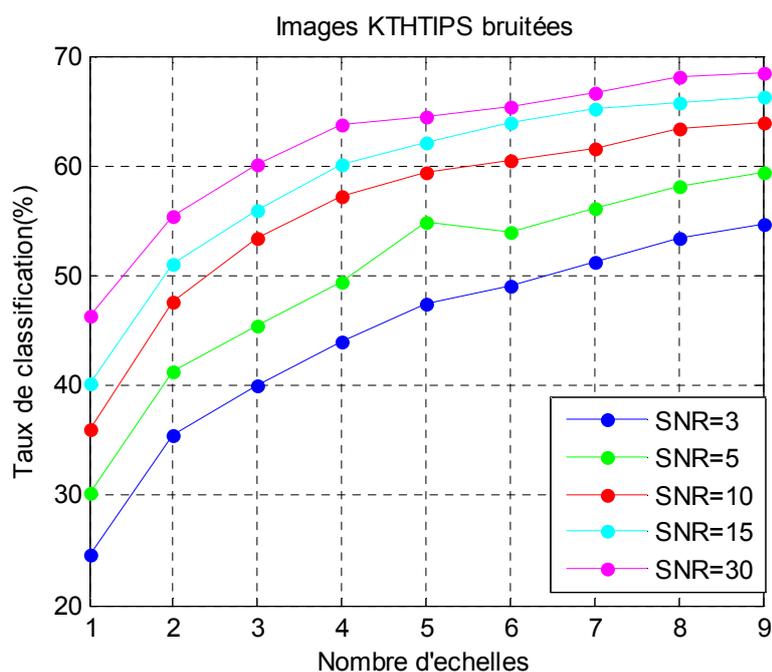


Fig.4.11. Taux de classification sur la base KTH TIPS2b bruitée en fonction du nombre d'échelles obtenus avec différents SNR

4.3. Le codage directionnel

Pour extraire des paramètres de textures plus discriminatifs et invariants en rotation tout en réduisant leur nombre, des modifications sont portées sur le codage de rang directionnel. Le nouveau codage est nommé DC (*Directional Coding*).

Dans le codage directionnel, les 4 valeurs cod_{θ} correspondant à la position du niveau de gris du pixel central sur les 4 orientations θ sont exploitées différemment.

D'abord, les quatre valeurs sont associées dans un vecteur noté : $Rank$, $Rank = [cod_{135} cod_{90} cod_{45} cod_0]$. Afin d'extraire les caractéristiques à partir du codage DC, les trois

valeurs possibles dans ce vecteur sont exploitées. Les n_1, n_2 et n_3 sont respectivement le nombre des valeurs 0, 1 et 2 dans le vecteur Rank et sont calculées par l'équation suivante.

$$n_l = \sum_{i=1}^4 f(\text{Rank}(i), l) \quad l \in [0, 2] \quad (4.7)$$

La fonction f est déjà définie dans l'équation (4.5)

Les valeurs n_l sont comprises dans l'intervalle fermé $[0, 4]$ et $\sum_l n_l = 4$. Il existe 15 combinaisons possibles des trois valeurs n_1, n_2 et n_3 . Ces combinaisons sont : 400, 040, 004, 211, 112, 121, 013, 031, 022, 202, 220, 310, 103, 310 et 130.

Vu l'importance de l'information niveau de gris dans l'analyse de textures [Guo et al, 2010, [Liu et al, 2014], les trois valeurs de n_l sont combinées conjointement avec l'information du pixel central. Cette dernière est exploitée après un seuillage global en utilisant la fonction C définie dans l'équation (4.8).

$$C = F(I(x) - \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M I(i, j)) \quad (4.8)$$

Où :

$N \times M$ représente la taille de l'image. La fonction F est déjà définie dans l'équation (4.2).

Le codage DC est enfin calculé suivant l'équation (4.9), l'opérateur DC produit 30 paramètres invariants par rotation.

$$DC(x) = n_1 \times 5^0 + n_2 \times 5^1 + n_3 \times 5^2 + C \times 5^3 \quad (4.9)$$

Le codage DC permet de distinguer les différentes distributions de pixels locaux. Ainsi, les statistiques des caractéristiques DC peuvent également être utilisées pour représenter les structures texturales. Ainsi, après le calcul du code DC pour chaque pixel (i, j) , l'image I est définie par l'histogramme des codes $H_{DC(I)}$ suivant l'équation (4.10)

$$H_{DC(I)} = \sum_{i=1}^N \sum_{j=1}^M f(DC(i, j), l) \quad , \quad l \in [0, L], \quad (4.10)$$

L est la valeur max de DC.

La figure 4.12 montre deux exemples de calcul du code DC. D'abord, le niveau de gris du pixel central est classé selon les quatre orientations : $0^\circ, 45^\circ, 90^\circ$, et 135° . Sur le premier

exemple, le pixel central de valeur 52 est entouré suivant la direction 0° de deux voisins de niveau de gris 64 et 32. Le rang du pixel central est égal à 1. Aussi, sur chacune des directions 45° et 90° , le pixel central a le niveau de gris maximum, son rang est alors égal à 2. Par contre, le rang du pixel central suivant la direction 135° est de 0 puisque le niveau de gris de ses deux voisins est supérieur. Le codage DC est finalement calculé en combinant les trois valeurs n_0 , n_1 et n_2 avec C suivant l'équation (4.9) (nous supposons dans ces exemples que $C = 0$). Ainsi, la valeur finale de DC est de 56. Dans le deuxième exemple, le niveau de gris du pixel central est minimum sur trois directions ($n_0 = 3$) et un maximum sur une direction ($n_2 = 1$). La valeur finale de DC est de 76.

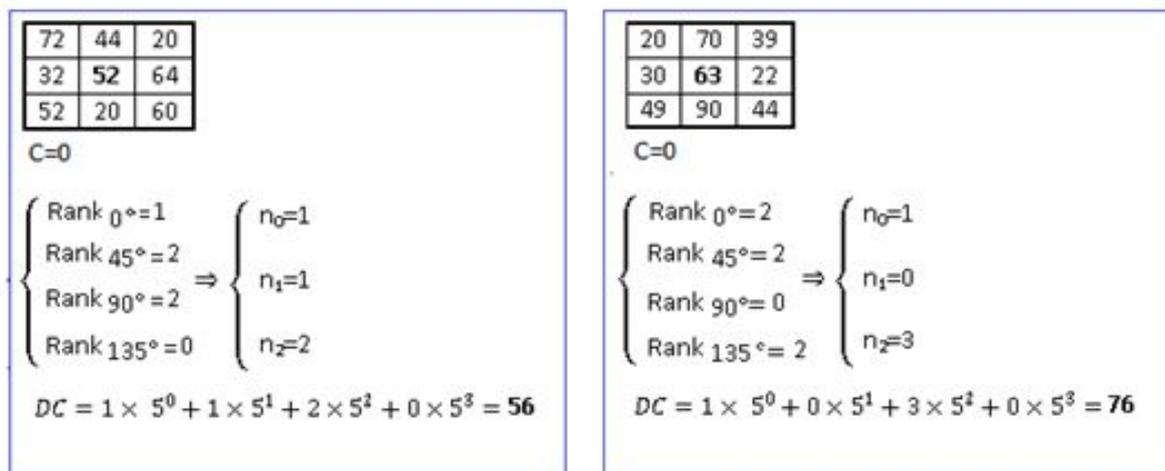


Fig.4.12. Deux exemples de calcul du codage DC

Pour l'étude multi résolution, la même procédure proposée pour LDRC est appliquée. Le code DC est calculé sur différentes résolutions spatiales en modifiant la taille de la fenêtre $S \times S$. Sur chaque échelle, la transformation des voisinages est effectuée suivant l'équation (4.6). Ainsi, le code DC est calculé par rapport au pixel central suivant l'équation (4.9). L'histogramme des paramètres DC est concaténé sur multiples échelles; cet histogramme est de taille $n \times 30$, n étant le nombre d'échelles.

4.3.1. Tests et résultats

Pour évaluer l'efficacité de la méthode DC, les tests sont effectués sur cinq bases de données : OUTEX TC10, OUTEX TC 12-000, OUTEX TC 12-001, CURET et KTH Tips2b. Ces bases de données ont différentes propriétés comme le nombre de classes, la rotation, l'éclairage et les poses. L'histogramme des caractéristiques de DC est concaténé sur cinq échelles en modifiant la taille de la fenêtre de 3x3 à 11x11.

Comme dans le cas du codage LDRC, le classificateur plus proche voisin (1-NN) utilisant la distance χ^2 est appliqué.

Notre approche est comparée à la méthode LBP et quelques unes de ses principales variantes comme DLBP [Liao et al, 2009], CLBP [Guo et al, 2010], DNS+LBP [Khellah, 2011]. Les méthodes VZ-MR8 [Vama et al, 2005] et VZ- joint [Vama et al, 2009] sont également utilisés dans la comparaison. Notons que tous les résultats référencés dans les tableaux sont reproduits à partir des articles cités.

4.3.1.1. Résultats obtenus sur la base Outex

La classification des textures utilisant la méthode proposée est évaluée en fonction du nombre d'échelles. La table 4.6 présente les résultats obtenus pour les différentes échelles utilisées (1à5). La meilleure performance est obtenue en utilisant trois échelles(MS3) pour TC10 et quatre échelles(MS4) pour TC12.

Bases de données	MS1	MS2	MS3	MS4	MS5
TC10	95.39	96.48	97.24	96.80	96.22
TC12_000	87.11	91.46	92.85	92.94	92.87
TC12_001	86.71	90.97	92.57	92.94	92.73

Table 4.6. Taux de classification en fonction du nombre d'échelles

Comparativement à plusieurs méthodes (table 4.7), la méthode proposée est compétitive avec des techniques très performantes. Essentiellement sur la base TC12, le taux de classification de DC apparaît parmi les meilleurs en utilisant moins de paramètres. Par exemple le CLBP est plus efficace mais le nombre de ses paramètres (2200) est d'environ dix-huit fois plus que le nombre de paramètres de DC (120), ce qui est considérable.

Méthodes	TC10	TC12_000	TC12_001	Nombre de paramètres
$LBP_{P,R}^{riu2}/Var$ [Ojal et al., 2002 a]	97.7	87.3	86.4	242
VZ_MR8[Vama et al, 2005]	93.59	92.55	92.82	610
VZ_joint[varma et al, 2009]	92.00	91.41	92.06	610
DLBP[Liao et al, 2009]	99.1	93.2	90.4	$K_{80\%}$
$CLBP_{P,R}^{riu2}$ [Guo et al, 2010]	99.14	95.18	95.55	2200
DNS+ $LBP_{24,3}$ [Khellah,2011]	99.27	94.40	92.85	242
DC(MS3)	97.24	92.85	92.57	90
DC(MS4)	96.80	92.94	92.94	120

Table 4.7. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur l'ensemble d'images TC10 et TC 12. Les deux plus hauts taux de classification sont donnés en gras

4.3.1.2 Résultats obtenus sur la base CURET

Dans cette expérience, nous utilisons différentes tailles d'échantillons d'apprentissage. Pour chaque classe, un nombre N d'échantillons d'apprentissage est choisi. Les valeurs de N considérées sont 46, 23, 12 et 6. Pour retrouver le taux de classification moyen, nous effectuons 50 partitionnements aléatoires des échantillons d'apprentissage et de test.

La table 4.8 présente les taux de classification obtenus en fonction du nombre d'échelles. Les résultats montrent l'augmentation du taux de classification avec le nombre d'échelles. Pour le nombre d'échantillons N égal à 46 ou 23, le maximum est obtenu pour cinq échelles (MS5). Cependant, si N est égal à 12 ou à 6, le meilleur taux est acquis pour MS3.

Les méthodes	Nombre d'échantillons d'apprentissage (N)			
	46	23	12	6
MS1	58.01	56.71	53.28	52.17
MS2	95.72	94.40	91.92	87.45
MS3	98.41	96.84	94.50	89.23
MS4	98.50	96.88	94.01	89.21
MS5	98.67	97.01	94.14	88.48

Table 4.8. Performances de DC en fonction du nombre d'échelles, le taux de classification le plus élevé est indiqué en gras

Comparée aux autres méthodes, la méthode proposée permet d'acquérir des résultats très impressionnants comme il est montré sur la table 4.9. La méthode proposée permet d'obtenir les meilleurs résultats dans tous les cas. Plus le nombre d'échantillons d'apprentissage est petit, plus l'efficacité de DC par rapport aux autres méthodes est meilleure. Par exemple pour $N = 46$, le taux obtenu par DC (MS3) est de 0,6% fois plus grand que celui obtenu avec VZ-MR8, alors que pour $N = 6$, la différence est de plus de 7%. Aussi, le nombre de paramètres VZ-MR8 est de 610, soit sept fois le nombre d'attributs de DC (90).

Méthodes	Nombre d'échantillons d'apprentissage (N)				Nombre de paramètres
	46	23	12	6	
VZ_MR8[Vama et al, 2005]	97.79	95.03	90.48	82.60	610
VZ_joint[varma et al, 2009]	97.66	94.58	89.40	81.06	610
DLBP+NGF[Liao et al, 2009]	84.1	-	-	-	$K_{80\%}$
DNS+LBP _{16,2} [Khellah,2011]	95.00	-	-	-	234
CLBP _{P,R} ^{riu2} [Guo et al, 2010]	97.39	94.19	88.72	79.88	2200
DC(MS3)	98.41	96.84	94.50	89.23	90
DC(MS5)	98.67	97.01	94.14	88.48	150

Table 4.9. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur la base CURET. Les deux taux les plus élevés de classification sont indiqués en gras

4.3.1.3 Résultats obtenus sur la base KTH-Tips2b

Contrairement aux tests effectués avec LDRC sur la base KTH- Tips2b, nous utilisons trois configurations avec un, deux et trois ensembles d'apprentissage par classe. Les ensembles 1, 2, 3 et 4 sont combinés en couple (apprentissage, test) en considérant toutes les combinaisons possibles. Pour une série d'apprentissage, nous avons les couples (1,2U3U4), (2,1U3U4), (3,1U2U4), (4,1U2U3). Pour deux séries d'apprentissage, nous considérons les couples (1U2,3U4), (3U4,1U2), (1U3,2U4), (2U4,1U3), (1U4,2 U3) et (2U3,1U4). Trois ensemble d'apprentissages sont définis par (2U3U4,1), (1U3U4,2), (1U2U4,3) et (1 U2U3,4). Pour chaque configuration, le taux de classification représente la moyenne des taux retrouvés sur toutes les combinaisons.

Comme nous pouvons l'apercevoir sur la Table 4.10, la méthode DC permet d'acquérir les meilleurs résultats atteignant respectivement 56.99%, 64.47% et 68.56% pour le nombre d'ensemble d'apprentissage égal à 1, 2 et 3.

Notons qu'avec uniquement une seule échelle ($S=3$) donc seulement 30 descripteurs, la méthode proposée a atteint des taux de classification de 54.17%, 61.08% et 64.88% c'est à dire des taux considérables par rapport à d'autres méthodes, ce qui prouve la pertinence des descripteurs du codage DC.

Méthodes	Nombre d'ensemble d'apprentissage (N)			Nombre de paramètres
	1	2	3	
VZ-MR8[Vama et al, 2005]	46.1	52.0	55.3	610
VZ-joint[varma et al, 2009]	53.5	60.0	61.0	610
DLBP[Liao et al, 2009]	49.3	55.1	58.0	$K_{80\%}$
$CLBP_{P,R}^{riu2}$ [Guo et al, 2010]	55.0	61.1	67.7	2200
DC(MS1) $s=3$	54.17	61.08	64.88	30
DC(MS2) $s=3,17$	56.99	64.47	68.56	60

Table 4.10. Comparaison de la méthode proposée avec d'autres méthodes d'analyse de la texture sur la base KTH-Tips2b. Les deux taux les plus élevés de classification sont indiqués en gras

4.3.1.4. Durée d'exécution

Tous les tests sont effectués sur un PC à processeur Intel Core i5, 4G RAM Windows 7 et en utilisant Matlab version 7.10 (2010a). Le temps consommé pour extraire les 30 descripteurs DC d'une image de texture de la taille 200 * 200 est environ 0.08 secondes.

4.4. Discussions

Dans ce chapitre, nous avons proposé deux méthodes d'analyse de textures.

Le codage local de rang directionnel (LDRC) permet d'extraire 81 caractéristiques robustes au bruit. Comme le montre les différents tests effectués sur les trois bases de données : Brodatz, CURET et KTH-Tips2b, la méthode LDRC est performante dans le cas d'images non bruitées. Elle surpasse considérablement toutes les techniques en présence de bruit gaussien. Comparativement aux autres méthodes, la performance relative du codage LDRC augmente à mesure que l'intensité du bruit augmente.

Le codage directionnel (DC) permet d'extraire des descripteurs de textures compactes, discriminatifs et invariants en rotation. Cette méthode a été évaluée sur une large gamme de textures. Nous avons étudié différents problèmes liés à la reconnaissance de textures tels que l'effet de l'éclairage, la rotation et changement d'échelle, et aussi l'influence du nombre d'images d'apprentissages. Comparée à d'autres méthodes de l'état de l'art, la méthode DC a acquis le taux de classification le plus élevé sur les bases de données CURET et KTH-Tips2b. La méthode DC a des résultats prometteurs sur les bases de données Outex essentiellement sur la base Outex-TC12.

CONCLUSION

Dans de nombreuses applications pratiques, il est très difficile ou impossible de s'assurer que les textures soient capturées dans les mêmes conditions d'éclairage, de rotation ou de résolution. Malgré les nombreux travaux existants consacrés à la description de textures, l'extraction de caractéristiques de textures invariantes aux changements d'éclairage, d'orientation ou d'échelle et la présence de bruit reste un problème important à résoudre. Ce travail s'inscrit dans le cadre du problème mentionné.

Nous avons proposé plusieurs méthodes d'analyse de textures. Nous avons développé le codage de rang régional (RRCs). Ce codage permet d'extraire huit descripteurs invariants en rotation. Les résultats de classification effectués sur les bases outex_TC10 et outex_TC12 montrent que RRC₃ a de meilleures performances que la méthode $LBP_{1,8}^{riu2}$. Nous avons remarqué que RRCs produit les meilleurs taux de classification sur la base TC12_001. Aussi l'utilisation conjointe de RRCs/Moy/Var permet d'améliorer considérablement ses performances.

Nous avons proposé le codage local de rang directionnel (LDRC). Ce codage permet d'extraire 81 caractéristiques robustes au bruit. Comme l'ont montré les différents tests effectués sur les trois bases de données : Brodatz, CURET et KTH-TIPS2b, en présence ou non de bruit, la méthode LDRC a permis d'obtenir les meilleurs taux de classification par rapport à d'autres variantes de LBP: DLBP [Liao et al, 2009], DNS + LBP [Khellah, 2011], BRINT [Liu et al, 2014] et LCP [Song et al, 2014].

Afin d'extraire des descripteurs de textures compactes, discriminatifs et invariants en rotation, nous avons proposé une extension du codage LDRC. Le codage DC a été évalué sur une large gamme de textures. Nous avons étudié différents problèmes liés à la reconnaissance de textures tels que l'effet de l'éclairage, la rotation et le changement d'échelle, et aussi l'influence du nombre d'images d'apprentissages. Comparée à d'autres méthodes de l'état de l'art : DLBP [Liao et al, 2009], CLBP [Guo et al, 2010], DNS [Khellah, 2011], VZ-MR8 [Vama et al, 2005] et VZ-joint [Vama et al, 2009], la méthode DC a acquis le meilleur taux

de classification sur les bases de données CURET et KTH-TIPS2b. Cette méthode a des résultats prometteurs sur les bases de données Outex essentiellement sur la base Outex-TC12. Aussi, les méthodes proposées sont simples à mettre en œuvre avec un schéma de calcul rapide, par exemple, l'extraction des 30 descripteurs DC d'une image de texture de la taille 200×200 ne dépasse pas les 8 milli-secondes.

Perspectives :

Les méthodes proposées ouvrent des possibilités pour d'autres chercheurs à travailler sur différentes parties comme l'emploi d'autres méthodes pour l'étude multi-résolution tel que le filtre de Gabor.

L'utilisation d'un voisinage circulaire pourrait améliorer l'invariance en rotation des paramètres.

Une exploitation plus exhaustive du classifieur SVM pourrait améliorer les différents résultats.

Des variantes de LBP pourraient être appliquées aux méthodes proposées et citons à titre d'exemple les méthodes : LBPV [Guo et al, 2010a], CLBP [Zhao et al, 2012], DNS [Ahonen et al, 2009] et PRICoLBP [Qi et al, 2014].

Les méthodes proposées pourraient être testées sur différentes autres bases de données.

L'application des méthodes proposées sur d'autres domaines du modèle binaire local pourrait également être explorée comme par exemple la reconnaissance faciale, le domaine du biomédical ou la recherche d'images par le contenu.

Annexe A : Séparateurs à vaste marge

A.1 Séparateurs à vaste marge :

Les séparateurs à vaste marge ou SVM (*Support Vector Machine*) sont des techniques d'apprentissage supervisé. Initialement, les SVM s'intéressent à la classification binaire en déterminant l'hyperplan qui divise deux classes, le principe est de maximiser la marge de l'hyperplan des deux classes suivant la théorie statistique de l'apprentissage conduite par Vapnik [Vapnik, 1995]. Les échantillons les plus proches de la marge sélectionnés pour déterminer l'hyperplan sont nommés les vecteurs de support. Les exemples utilisés lors de la recherche de l'hyperplan ne sont alors plus utiles et seuls ces vecteurs supports sont utilisés pour classer un nouveau cas, ce qui peut être considéré comme un avantage pour cette méthode. Le SVM multi-classe est construit en utilisant plusieurs SVM à deux classes. Les SVM sont particulièrement bien adaptés pour classer des données de très haute dimension.

A.1.1 Classifieur linéaire

Un classifieur est dit linéaire lorsqu'il est possible d'exprimer sa fonction de décision par une fonction linéaire en x . Cette fonction est exprimée par :

$$h(x) = \langle w, x \rangle + b = \sum_{i=1}^n w_i x_i + b \quad (\text{A.1})$$

où $w \in R^n$ est le vecteur de poids et $b \in R^0$, alors que x est la variable du problème. X est l'espace d'entrée et qui correspond à R^n , où n est le nombre de composantes des vecteurs contenant les données. Notons que l'opérateur $\langle w, x \rangle$ désigne le produit scalaire usuel dans R^n , w et b sont les paramètres à estimer de la fonction de décision $h(x)$.

Pour décider à quelle classe un exemple estimé x appartient, il suffit de prendre le signe de la fonction de décision: $y = \text{sign}(h(x))$. La fonction $\text{sign}(\cdot)$ est appelée classifieur. Géométriquement (voir figure A.1), cela revient à considérer un hyperplan correspond aux points x satisfaisant $\langle w, x \rangle + b = 0$. En orientant l'hyperplan, la règle de décision correspond à observer de quel côté de l'hyperplan se trouve l'exemple x . Le vecteur w définit la pente de

l'hyperplan (w est perpendiculaire à l'hyperplan). Le terme b quant à lui permet de translater l'hyperplan parallèlement à lui-même.

L'objectif de la discrimination linéaire est de trouver la bonne fonction de décision $h(x)$

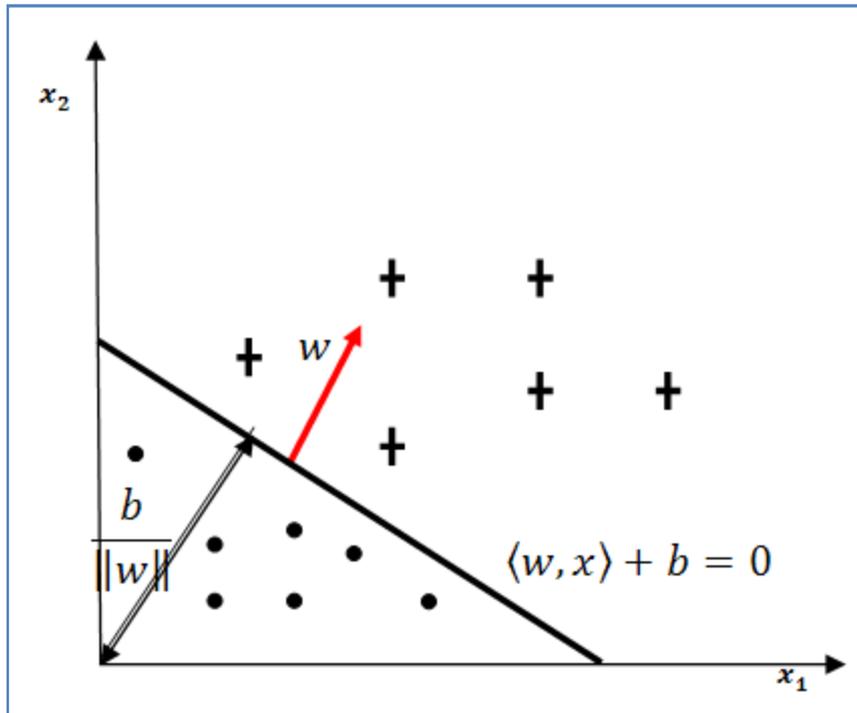


Fig.A.1 .Hyperplan séparateur $\langle w, x \rangle + b = 0$

A.1.1.1 Marge maximale de l'hyperplan

La marge est la distance entre la frontière de séparation et les échantillons les plus proches. Dans les SVM, la frontière de séparation est choisie comme celle qui maximise la marge.

La marge géométrique représente la distance euclidienne prise perpendiculairement entre l'hyperplan et l'exemple x_i . En prenant un point quelconque x_p se trouvant sur l'hyperplan, la marge géométrique peut s'exprimer par $\frac{w}{\|w\|} \cdot (x_i - x_p)$ retrouver comme suit :

$$\begin{aligned} \langle w, x_i \rangle + b &= f(x_i) \\ - \langle w, x_p \rangle + b &= 0 \\ \hline w \cdot (x_i - x_p) &= f(x_i) \end{aligned}$$

Donc :

$$\frac{w}{\|w\|} \cdot (x_i - x_p) = \frac{f(x_i)}{\|w\|} \quad (\text{A-2})$$

La figure (A-2) illustre un exemple de cette distance

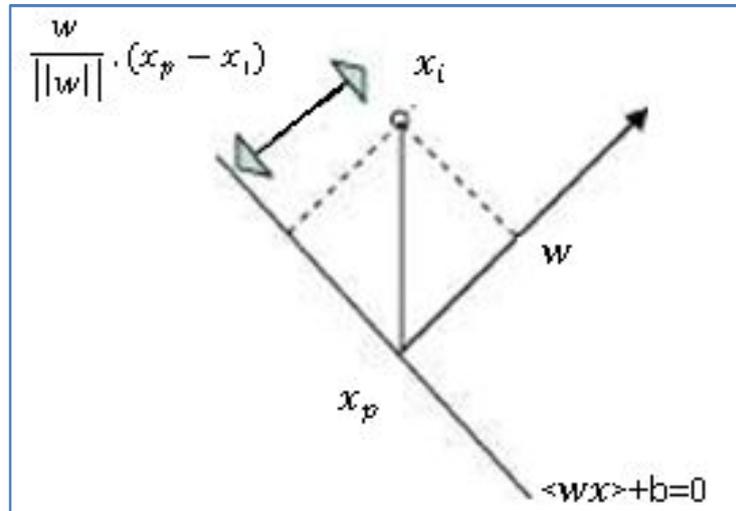


Fig. A.2. Illustration de la marge géométrique pour l'exemple x_i

Deux classes de l'échantillon d'apprentissage S sont linéairement séparables s'il existe des paramètres (w^*, b^*) tels que l'on a pour tout $i (=1,2,\dots,n)$:

$$\begin{aligned} w^* x_i + b^* &> 0 \text{ si } y_i = 1 \\ w^* x_i + b^* &< 0 \text{ si } y_i = -1 \end{aligned}$$

Ce qui est équivalent à :

$$y_i(w^* x_i + b^*) > 0; \forall i = 1, 2, \dots, n \quad (\text{A.3})$$

L'estimation des paramètres (w^*, b^*) de l'hyperplan qui maximise la marge se fait en résolvant le problème d'optimisation suivant :

$$(w^*, b^*) = \operatorname{argmax}_{(w,b)} \{ \min_i (y_i (w x_i + b)), \|w\| = 1 \} \quad (\text{A.4})$$

Donc, il existe un hyperplan séparant d'un côté toutes les données positives et de l'autre, toutes les données négatives. On peut définir deux plans se trouvant de part et d'autre de l'hyperplan et parallèles à celui-ci, sur lesquels reposent les exemples les plus proches. La figure (A.3) illustre cette situation.

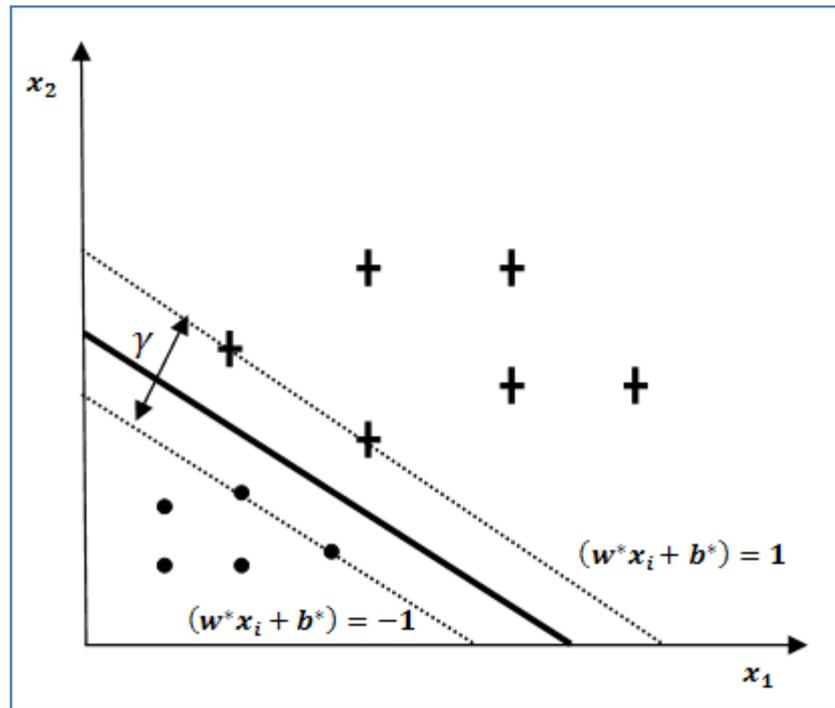


Fig. A-3. Hyperplans canoniques et marge maximale

Dans la définition de l'hyperplan, il est possible que différentes équations correspondent au même plan géométrique :

$$a(\langle w, x \rangle + b) = 0 \quad (\text{A.5})$$

a est une constante quelconque.

Il est donc possible de redimensionner (w^*, b^*) de telle sorte que les deux plans parallèles aient respectivement pour équations :

$$(w^*x_i + b^*) = 1 \quad (\text{A.6})$$

$$(w^*x_i + b^*) = -1 \quad (\text{A.7})$$

Ces deux hyperplans sont appelés hyperplans canoniques. Ainsi la marge γ entre ces deux plans est égale à :

$$\gamma = \frac{2}{\|w^*\|} \quad (\text{A.8})$$

A.1.1.2 Minimisation quadratique sous contraintes

Un problème d'optimisation mathématique est formulé tel que sa solution nous fournisse l'hyperplan optimal qui permet de maximiser la marge :

$$\begin{cases} \text{Minimiser} & \frac{1}{2} \|w\|^2 \\ \text{sous les contraintes} & y_i(w^* x_i + b^*) \geq 1 \end{cases} \quad (\text{A.9})$$

Il s'agit d'un problème quadratique convexe sous contraintes linéaires de forme primal dont la fonction objective est à minimiser. Cette fonction objective est le carré de l'inverse de la double marge. L'unique contrainte stipule que les exemples doivent être bien classés et qu'ils ne dépassent pas les hyperplans canoniques.

Dans cette formulation, les variables à fixer sont les composantes w_i et b . Le vecteur w possède un nombre de composantes égal à la dimension de l'espace d'entrée. Généralement dans ce type de cas on résout la forme duale du problème. Nous devons former ce que l'on appelle le Lagrangien. Il s'agit de faire rentrer les contraintes dans la fonction objective et de pondérer chacune d'entre elles par une variable dual :

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i (\langle w, x_i \rangle + b) - 1) \quad (\text{A.10})$$

Les variables duales α_i intervenant dans le Lagrangien sont appelées multiplicateurs de Lagrange. Ils représentent la force avec laquelle la solution s'appuie sur la contrainte i . Ainsi, un hyperplan qui violerait la contrainte pour x_i (il classe cet exemple du mauvais côté) rendrait α_i très grand ce qui ferait fortement augmenter la fonction objective L . Cette solution ne pourrait donc pas être retenue comme solution optimale. Notons que L doit être minimisé par rapport aux variables primales w_i et b et maximisé par rapport aux variables duales α_i .

Le point selle (minimal par rapport à une variable, maximal par rapport à l'autre) doit donc satisfaire les conditions nécessaires de stationnarité qui correspondent aux conditions Karush Kuhn et Tucker (KKT) :

$$\begin{aligned} \frac{\partial L(w, b, \alpha)}{\partial w} &= 0 \\ \frac{\partial L(w, b, \alpha)}{\partial b} &= 0 \end{aligned}$$

$$\frac{\partial L(w, b, \alpha)}{\partial w} = w - \sum_{i=1}^n \alpha_i y_i x_i = 0$$

$$\frac{\partial L(w, b, \alpha)}{\partial b} = - \sum_{i=1}^n \alpha_i y_i = 0$$

Ce qui permet d'obtenir :

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (\text{A.11})$$

$$\sum_{i=1}^n \alpha_i y_i = 0 \quad (\text{A.12})$$

Remarquons qu'avec cette formulation, w peut être calculé en fixant seulement n paramètres. L'idée est de formuler un problème dual dans lequel w est remplacé par sa nouvelle formulation. De cette façon, le nombre de paramètres à fixer est relatif au nombre d'exemples de l'échantillon d'apprentissage et non plus à la dimension de l'espace d'entrée. Pour se faire, nous substituons (A.1) et (A.12) dans le Lagrangien L , nous obtenons le problème dual équivalent suivant :

$$L(w, b, \alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \quad (\text{A.13})$$

A partir de quoi le problème dual est formulé :

$$\left\{ \begin{array}{l} \text{Maximiser } w(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \\ \text{avec } \left\{ \begin{array}{l} \sum_{i=1}^n \alpha_i y_i = 0 \\ \alpha_i \geq 0 \end{array} \right. \end{array} \right. \quad (\text{A.14})$$

Ce dernier problème peut être résolu en utilisant des méthodes standards de programmation quadratique. Une fois la solution optimale $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_n^*)$ du problème (A.14) obtenue, le vecteur de poids de l'hyperplan à marge maximale recherché s'écrit :

$$W^* = \sum_{i=1}^n \alpha_i^* y_i x_i \quad (\text{A.15})$$

Comme le paramètre b ne figure pas dans le problème dual, sa valeur optimale b^* peut être dérivée à partir des contraintes primales, soit donc :

$$b^* = -\frac{\max_{y=-1}(\langle w^*, x_i \rangle) + \min_{y=1}(\langle w^*, x_i \rangle)}{2} \quad (\text{A.16})$$

Une fois les paramètres α^* et b^* calculés, la règle de classification d'une nouvelle observation basée sur l'hyperplan à marge maximale est donnée par :

$$h(x) = \text{sign}(\sum_{i=1}^n y_i \alpha_i^* \langle x_i, x \rangle + b^*) \quad (\text{A.17})$$

Notons qu'un grand nombre de termes de cette somme est nul. En effet, seuls les α_i^* correspondants aux exemples se trouvant sur les hyperplans canoniques (sur la contrainte) sont non nuls. Ces exemples sont appelés Supports Vecteurs (SV). On peut les voir comme les représentants de leurs catégories car si l'échantillon d'apprentissage n'était constitué que des SV, l'hyperplan optimal que l'on trouverait serait identique.

A.1.2. SVM non-linéaires

Le paragraphe précédent décrit le principe des SVM dans le cas où les données sont linéairement séparables. Cependant, dans la plupart des problèmes réels, ce n'est pas toujours le cas et il est donc nécessaire de contourner ce problème (difficile de séparer n'importe quel jeu de données par un simple hyperplan). Si par exemple les données des deux classes se chevauchent sévèrement, aucun hyperplan séparateur ne sera satisfaisant.

L'idée est de projeter les points d'apprentissage x_i dans un espace T de dimension q , plus élevée que n grâce à une fonction non-linéaire ϕ qu'on appelle fonction noyau, choisie a priori et d'appliquer la même méthode d'optimisation de la marge dans l'espace T . L'espace T ainsi obtenu est appelé espace des caractéristiques ou aussi espace transformé.

Donc, il faut résoudre le problème (A.14) dans l'espace T , en remplaçant $\langle x_i, x_j \rangle$ par $\langle \phi(x_i), \phi(x_j) \rangle$. L'hyperplan séparateur obtenu dans l'espace T est appelé hyperplan optimal généralisé. Le produit scalaire $\langle \phi(x_i), \phi(x_j) \rangle$ peut se calculer facilement à l'aide d'une fonction symétrique k , dite noyau, définie par :

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \quad (\text{A.18})$$

Le noyau $K(x,y)$ est une fonction de deux variables, symétrique et positive. Dans ce cas, la frontière de décision devient :

$$h(x) = \text{sign}(\sum_{i=1}^n y_i \alpha_i^* K(x_i, x_j) + b^*) \quad (\text{A.19})$$

Une famille de ces fonctions noyaux qui sont très appropriées aux besoins des SVM peut être définie, en voici les plus utilisés :

Noyau polynomial d'ordre p :

$$K(x_i, x_j) = (\langle x_i, x_j \rangle + 1)^d \quad (\text{A.20})$$

La dimension de l'espace transformé induit par un noyau polynomial est de l'ordre $\frac{(p+d)!}{p!d!}$, où p est la dimension de l'espace de départ

Noyau linéaire :

$$K(x_i, x_j) = x_i x_j \quad (\text{A.21})$$

Noyau gaussien de largeur de bande σ :

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma}\right) \quad (\text{A.22})$$

Le paramètre σ permet de régler la largeur de la gaussienne. En prenant un σ grand, la similarité d'un exemple par rapport à ceux qui l'entourent sera assez élevée, alors qu'on prenant un σ tendant vers 0, l'exemple ne sera similaire à aucun autre.

A.1.3. Relaxation des contraintes

Quand le domaine du problème d'optimisation est vide et il n'admet donc pas de solution, dans ce cas les données sont non linéairement séparables. Pour tenter de résoudre ce problème, on relâche les contraintes (A.14) (2) dans le but d'autoriser quelques erreurs de classification. Cette généralisation de l'hyperplan à marge maximale a été proposée par Cortes et al. [Cortes et al,1995] en introduisant les variables d'écart à la marge . L'hyperplan optimal est celui qui satisfait les conditions suivantes :

- La distance entre les vecteurs bien classés et l'hyperplan optimal doit être maximale.
- La distance entre les vecteurs mal classés et l'hyperplan optimal doit être minimale.

Le problème (A.14) (1) devient alors :

$$\begin{aligned}
 & \text{Minimiser } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \\
 & \text{Tel que } y_i (\langle w, x_i \rangle + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, n \\
 & \xi_i \geq 0, \quad i = 1, 2, \dots, n
 \end{aligned}
 \tag{A.23}$$

Autrement dit, on cherche à maximiser la marge en s'autorisant pour chaque contrainte une erreur positive ξ_i , la plus petite possible. Le paramètre supplémentaire C est appelé paramètre de pénalisation du relâchement. C est une constante positive fixée à l'avance qui permet de contrôler l'importance de l'erreur que l'on s'autorise par rapport à la taille de la marge. Plus C est important, moins d'erreurs sont autorisées.

A.1.4. SVM multi classes

La plupart des problèmes ne se contentent pas de deux classes de données. Il existe plusieurs méthodes pour faire la classification multi classes. Citons les plus utilisées : La première est une méthode dite Un-contre-Un. Au lieu d'apprendre N fonctions de décisions, ici chaque classe est discriminée d'une autre.

La deuxième méthode est appelé Un-Contre-Tous. C'est une approche étendant la notion de marge aux cas multi classes. Cette formulation intéressante permet de poser un problème d'optimisation unique. Le problème fait intervenir N fonctions de décision.

Bilan scientifique relatif à la thèse

Articles publiés dans des revues internationales à comité de lecture

[Ouslimani et al, 2018] F. Ouslimani, A. Ouslimani , Z. Ameer. Rotation-Invariant texture classification based on Directional Coding, Neural Computing and Applications, <https://doi.org/10.1007/s00521-018-3462-9>

[Ouslimani et al, 2015b] F. Ouslimani, A. Ouslimani, Z. Ameer. Noise-robust texture features using local directional rank coding . Advances in computer science and engineering vol.15, no.1-2 ,pp. 1-12, 2015.

[Ouslimani et al, 2015a] F.Ouslimani, A.Ouslimani, Z. Ameer. Rotation Invariant Features Based on Regional Rank for Texture Classification. International Journal of Computational and Experimental Science and Engineering (IJCESEN) , vol. 1, no.2, pp. 31-35 , 2015

Articles publiés dans des conférences internationales, congrès et colloques

[Ouslimani et al, 2016] F. Ouslimani, A. Ouslimani , Z. Ameer, Directional Rank Coding For Multi-scale Texture Classification. International Conference of Computing for Engineering and Science (ICCES'2016) on 23-27 July 2016, Barcelone, Spain.

[Ouslimani et al, 2014] F.Ouslimani, Z. Ameer, A. Ouslimani. Texture analysis method using regional rank coding. Communication Systems, Networks & Digital Signal Processing (CSNDSP), 9th International Symposium on 23-25 July 2014, Manchester, UK.

Bibliographie

[Abadeni, 2010] N. Abbadeni. Texture representation and retrieval using the causal autoregressive model. *Journal of Visual Communication and Image Representation*, vol.21, pp. 651–664, 2010.

[Ahuja, 1982] N. Ahuja. Dot Pattern Processing Using Voronoi Neighborhoods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4, pp. 336-343, 1982.

[Ahonen et al, 2006] T. Ahonen , A. Hadid , M. Pietikainen. Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* ,vol.28, no.12, 2006 .

[Ahonen et al, 2009] T. Ahonen, J. Matas, C. He, and M. Pietikäinen,. Rotation invariant image description with local binary pattern histogram Fourier features. in *Proc. SCIA*, 2009, pp. 61–70.

[Alvaro et al., 2014] Alvaro G. Zuñiga , Joao B. Florindo , Odemir M. Bruno . Gabor wavelets combined with volumetric fractal dimension applied to texture analysis. *Pattern Recognition Letters*, vol. 36, pp.135–143, 2014.

[Ameur Z, 2005] Z.Ameur. Codage des images en vue d'une segmentation de haut niveau. Thèse de doctorat, UMMTO, 2005.

[Ameur Z et al, 2004] Z.Ameur, S. Ameur, A.Adane, H. Sauvageot, K. Bara. Cloud classification using the textural features of météosat images. *Int. J. Remote sensing*, vol.25, no.21, pp.4491-4504,2004.

[Arivazhagan et al., 2006] S. Arivazhagan , L. Ganesan , S. Padam Priyal. Texture classification using Gabor wavelets based rotation invariant features. *Pattern Recognition Letters*, vol.27 , pp.1976-1982, 2006.

[Augusteijn, 1995] M. F., Augusteijn, L.E . Clemens, K.A. Shaw. Performance evaluation of texture measures for ground cover identification in satellite images by means of a neural-network classifier. *IEEE Transactions on Geoscience and Remote Sensing*, vol.33, pp. 616-625, 1995.

[Bi, 1997] D. Bi. Segmentation d'images basée sur les statistiques de rangs des niveaux de gris. PhD thesis, Université de Tours, janvier 1997.

[Bonnevay,1997] S. Bonnevay. Texture features Extraction with the help of the Regional Extremality Coding. *IEEE Transactions Image Processing*, vol. 6, no. 6, pp. 888-895, 1997.

[Bounekkar, 1996] A. Bounekkar, M. Lamure, N. Nicoloyannis. Texture classification based upon spatial autocorrelation. In *Visual Communications and Image Processing*, 1996.

-
- [Brodatz, 1965] P. Brodatz,. Textures: A Photographic Album for Artists and Designers, Dover Publications, New York, 1965.
- [Chaieb et al, 2014] R. Chaieb , A. Bacha , K. Kalti , F. Ben Lamine. Soft Computing and Pattern Recognition (SoCPaR). 6th International Conference of 11-14 Aug. 2014, pp.203 – 208, Tunis.
- [Chang et al, 1993] T.Chang, C-C.J. Kuo. Texture analysis and classification with tree-structured wavelet transform. IEEE Transactions on Image Processing, vol. 2, no. 4, pp. 429-441. 1993
- [Chang et al, 2011] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, vol. 2, pp. 27:127:27, 2011, software available at <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>
- [Chen et al ,1992] J. L. Chen and A. Kundu. Rotation and gray scale transform invariant texture recognition using hidden Markov model. In *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vols. 1–5. 1992, pp.C69–C72.
- [Chen ,1994] Y. Chen , E. Dougherty. Grey-Scale Morphological Granulometric Texture Classification. *Optical Engineering*, vol.33, no. 8,pp.2713-2722, 1994
- [Chetverikov, 1982] D. Chetverikov. Experiments in the rotation-invariant texture discrimination using anisotropy features. *Proc. 6th International Conference on Pattern Recognition*, Munich, Germany, 1982, vol.2 pp.1071-1073.
- [Cohen et al, 1991] F. S. Cohen, Z. G. Fan, and M. A. Patel. Classification of rotated and scaled textured images using Gaussian Markov random field models. *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 2, pp. 192-202, 1991.
- [Cortes et al,1995] C. Cortes, V. Vapnik. Support-vector networks. *Mach. Learning* , vol.20 ,pp.273–297, 1995.
- [Cross et Jain, 1983] G. R . Cross, A. K. Jain. Markov random field texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5,vol.1, pp.25 –39,1983.
- [Cula and Dana, 2004] O.G. Cula, K.J.Dana. 3D texture recognition using bidirectional feature histograms. *Int.J.Comput.Vis.*vol.59, pp. 33–60, 2004.
- [Dana et al, 1999] K.Dana , B. Van-Ginneken , S.Nayar and J. Koenderink . Reectance and texture of real world surfaces, *ACM Trans.Graph.*,pp.181-34 ,1999, <http://www.cs.columbia.edu/CAVE//exclude/curet/>
- [Davis et al, 1979] L. S. Davis, SA. Johns and JK. Aggarwal. Texture analysis using generalized cooccurrence matrices. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol.1, pp.251-259,1979 .
- [Davis, 1981] L. S. Davis. Polarograms-a new tool for image texture analysis. *Pattern Recognit.*, vol. 13, no. 3, pp. 219–223, 1981.

- [Deng et al, 2004] H. Deng , D. A. Clausi . Gaussian MRF rotation-invariant features for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* ,vol.26 , no.7 pp. 951 – 955,2004
- [De Valois et al., 1982] R. L. De Valois, D. G. Albrecht, L. G. Thorell . Spatial frequency selectivity of cells in macaque visual cortex. *Vision Research*, vol.22, no.5, pp.545–559,1982.
- [Fahriz et al, 2013] Farhan Riaz , Ali Hassan , Saad Rehman , Usman Qamar. Texture Classification Using Rotation- and Scale-Invariant Gabor Texture Features. *IEEE Signal Processing Letters* , vol.20 ,no.6, pp. 607 – 610, 2013.
- [Fathi et al, 2012] A. Fathi, A.Naghsh-Nilchi. Noise tolerant local binary pattern operator for efficient texture analysis. *Pattern Recognit. Lett.*33, vol.9, pp.1093-1100, 2012.
- [Fu, 1982] K.S. Fu. *Syntactic Pattern Recognition and Applications*. Prentice-Hall, New Jersey, 1982.
- [Galloway, 1975] M. M. Galloway. Texture analysis using Gray Level Run Lengths. *Computer graphics and Image processing*, vol. 4, pp. 172-179, 1975.
- [Guo et al, 2010a] Z. Guo, L. Zhang, D. Zhang. Rotation invariant texture classification using LBP variance (LBPV) with global matching. *Pattern Recognition*.vol. 43, no.3, pp. 706-719, 2010.
- [Guo et al, 2010b] Z. Guo, L. Zhang, D. Zhang. A Completed Modeling of Local Binary Pattern Operator for Texture Classification. *IEEE Transactions on Image Processing* ,vol.19 ,no.6, pp. 1657 – 1663,2010.
- [Guo et al, 2010c] Z.Guo, L.Zhang,D.Zhang,S.Zhang. Rotation invariant texture classification using adaptive lbp with directional statistical features. In: *International Conference on Image Processing(ICIP)*,2010,pp.285–288.
- [Hafiane et al, 2007] A. Hafiane, G. Seetharaman, B. Zavidovique. Median binary pattern for textures classification. In *International Conference on Image Analysis and Recognition (ICIAR)*, *Lecture Notes in Computer Science*, vol. 4633, pp. 387–398, 2007.
- [Hafiane et al, 2015] A. Hafiane , K. Palaniappan , G. Seetharaman . Joint Adaptive Median Binary Patterns for texture classification. *Pattern Recognition* vol.48, pp.2609–2620 , 2015.
- [Heikkilä et al , 2009] M. Heikkilä, M. Pietikäinen , C. Schmid. Description of interest regions with local binary patterns. *Pattern Recognition*, vol. 42, no. 3, pp. 425-436, 2009.
- [Haralick et al., 1973] R.M. Haralick, K.Shanmugam, I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3, pp. 610-621, 1973.
- [Haralick, 1979] R.M. Haralick . Statistical and structural approaches to texture. *Proceedings of the IEEE*, vol. 67, no. 5, pp.786 – 804, 1979.

[Harvey et Gervais, 1978] L. O. Harvey, M. J. Gervais. Visual texture perception and fourier analysis. *Perception & Psychophysics*, vol. 24, no.6, pp. 534–542, 1978.

[Hegenbart et al, 2015] S. Hegenbart , A. Uhl. A scale-and orientation adaptive extension of Local Binary Patterns for texture classification. *Pattern Recognition* vol. 48, pp. 2633–2644, 2015.

[Huang et al, 2011] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen. Local binary patterns and its application to facial image analysis: A survey. *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 41, no. 6, pp. 765–781, 2011.

[Iakovidis, 2008] D. Iakovidis,E. Keramidas, D. Maroulis. Fuzzy local binary patterns for ultrasound texture characterization. In: A. Campilho, M. Kamel (Eds.),*Image Analysis and Recognition, Lecture Notes in Computer Science*, vol.5112, Springer, Berlin,Heidelberg, 2008, pp.750–759.

[Jain and Farrokhnia, 1991] A.K. Jain, F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, vol.24,no. 12, pp.1167-1186,1991

[Julesz, 1962] B. Julesz. Visual pattern discrimination. *IRE Transactions on Information Theory*, vol.8, no.2, pp.84 –92, 1962.

[Julesz et al., 1978] B. Julesz, E. N. Gilbert, J. D. Victor. Visual discrimination of textures with identical third-order statistics. *Biological Cybernetics*, vol. 3,no.3, pp.137–140, 1978.

[Julesz, 1981a] B. Julesz. Textons, the elements of texture perception, and their interaction. *Nature*, 290, pp. 91–97, 1981.

[Julesz, 1981b] B. Julesz. A theory of preattentive texture discrimination based on first-order statistics of textons. *Biological Cybernetics*, vol.41, no.2, pp.131–138, 1981

[Kandaswamy et al, 2011] U. Kandaswamy, S. A. Schuckers, D. Adjeroh. Comparison of Texture Analysis Schemes Under Non ideal Conditions *IEEE Transactions on Image processing* , vol .20, no. 6, pp. 2260-2275, 2011.

[Kashyap al, 1986] R. L. Kashyap and A. Khotanzad. A model-based method for rotation invariant texture classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 4, pp. 472–481, Jul. 1986.

[Khellah,2011] F.M.Khellah. Texture Classification Using Dominant Neighborhood Structure. *IEEE Tran. Image Process.*, vol.20, no.11, pp.3270-3279, Nov.2011.

[Klare et al, 2013] B. Klare, A. Jain. Heterogeneous face recognition using kernel prototype similarities. *IEEE Trans. Pattern Anal.Mach.Intell*, vol.35, no.6, pp .1410–1422, 2013.

[Kylberg et al, 2013] G. Kylberg , I-M. Sintorn. Evaluation of noise robustness for local binary pattern descriptors in texture classification. *EURASIP Journal on Image and Video Processing* vol.17,pp. 1-20, 2013.

-
- [Leung et Malik, 2001] T. Leung, J. Malik. Representing and Recognizing the Visual Appearance of Materials using Three-dimensional Textons. *International Journal of Computer Vision*, vol. 43, no.1, pp 29-44,2001.
- [Li et al., 2003] Li, S., Kwoka, J. T., Zhua, H. et Wang, Y. Texture classification using the support vector machines. *Pattern Recognition*, vol. 36, pp.2883–2893, 2003.
- [Liao et al, 2007] S. Liao and A. C. S. Chung. Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude. In *Proc. Asian Conf. Computer Vision (ACCV)*, 2007, pp. 672-679.
- [Liao et al, 2009] S. Liao, M W.K. Law , A. C.S. Chung . Dominant Local Binary Patterns for Texture Classification. *IEEE Transactions on Image Processing*, vol. 18, no.5, pp. 1107-1117, 2009.
- [Liu et al, 2014] L. Liu , Y. Long, P.W. Fieguth, S. Lao , G. Zhao. BRINT: Binary Rotation Invariant and Noise Tolerent Texture Classification. *IEEE transactions on Image Processing*, vol .23, no. 7,pp.3071-3084 , 2014.
- [Liu et al 2016] L. Liu , S. Lao , P. W. Fieguth ,Y. Guo. Median Robust Extended Local Binary Pattern for Texture Classification. *IEEE Transactions on Image Processing* , vol. 25 ,no. 3 , pp, 1368 – 1381,2016.
- [Liu et al 2017] L. Liu , P.W. Fieguth , Y. Guo ,X.Wang M. Pietikinen . Local binary features for texture classification. *Pattern Recognition* vol.62 , pp. 135 – 160,2017.
- [loris et al, 2015] N. Loris . , B.Sheryl , G. Stefano , M. Emanuele. Improving the descriptors extracted from the co-occurrence matrix using preprocessing approaches. *Expert Systems With Applications* , vol.42 pp. 8989–9000, 2015.
- [Lowe ,2004] D. Lowe. Distinctive Image Features from scale Invariant Key points. *International, Computer Vision* , vol.60, no 2.,pp.91-110,2004.
- [Mallat, 1989] S. Mallat. A theory for multi-resolution signal decomposition : the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no.7,pp. 674-693, 1989.
- [Mallikarjuna et al, 2006] P. Mallikarjuna ,M. Fritz., A. T. Targhi , E. Hayman , B. Caputo and J.Eklundh , The KTH-TIPS2 Databases, <http://www.nada.kth.se/cvap/databases/kth-tips/>
- [Mandelbrot,1983] B. B., Mandelbrot. *The Fractal Geometry of Nature*. Freeman, San Francisco, 1983.
- [Manjunath et al, 1991] B. S. Manjunath, R. Chellapa. Unsupervised texture segmentation using Markov Random Fields. *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13,pp 478-782,1991.
- [Mao et al, 1992] J. Mao, A.K. Jain. Texture classification and segmentation using multi resolution simultaneous autoregressive models. *Pattern Recognition* ,vol. 25 ,pp.173–188, 1992.

- [Mavromatis et al, 2004] S. Mavromatis, J-M Boï, R. Bulot, J. Sequeira. Texture analysis using directional local extrema. *Machine Graphics and Vision* , vol.13, no. 3 pp.289-302, 2004.
- [Mehta et al,2016] R. Mehta , K. Egiazarian. Texture Classification Using Dense Micro-Block Difference. *IEEE Transactions on Image Processing*, vol. 25, no. 4, 2016.
- [Murala et al, 2012] S. Murala, R. Maheshwari, R.Balasubramanian. Local tetra patterns: a new feature descriptor for content-based image retrieval. *IEEE Trans. Image Process.* vol.21, no.5, pp.2874-2886.2012.
- [Nanni et al, 2010] L.Nanni, A.Lumini, S.Braham. Local binary patterns variants as texture descriptors for medical analysis. *Artif.Intell.Med.*,vol.49, no.2, pp.117-125,2010
- [Nguyen et al, 2016] T. P. Nguyen, N-S. Vu, A. Manzanera. Statistical binary patterns for rotational invariant texture classification, *Neurocomputing*, vol. 173, no.3, 15 pp. 1565-1577, 2016.
- [Ohanian et al, 1992] P.P. Ohanian , R.C. Dubes. Performance evaluation for four classes of textural features. *Pattern Recognition*, vol. 25, pp.819-833,1992.
- [Ojala et al ,1996] T.Ojala , M.Pietikainen ,D.Harwood. A comparative study of texture measures with classification based on features distributions. *Pattern Recognition*, vol.29, no.1, pp.51-59, 1996.
- [Ojala et al, 2002a] T. Ojala , T. Maenpaa ,M. Pietikanen ,J. Viertola , J. Kyllonen , S. Huovinen. Outex New framework for empirical evaluation of texture analysis algorithms. In: *Int. Conf. Pattern Recognit.*,vol. 1, pp.701-706, 2002, <http://www.outex.oulu>
- [Ojala et al., 2002b]T.Ojala, M.Pietikainen ,T.Maenpaa. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.24, no.7, pp. 971–987 ,2002.
- [Ouslimani et al, 2014] F.Ouslimani, Z.Ameur, A.Ouslimani. Texture analysis method using regional rank coding. *Communication Systems, Networks & Digital Signal Processing (CSNDSP 2014)*, 9th International Symposium on 23-25 July 2014, Manchester, UK.
- [Ouslimani et al, 2015a] F.Ouslimani, A.Ouslimani, Z.Ameur. Rotation Invariant Features Based on Regional Rank for Texture Classification. *International Journal of Computational and Experimental Science and Engineering (IJCESEN)* , vol. 1, no.2, pp. 31-35 , 2015
- [Ouslimani et al, 2015b] F. Ouslimani, A. Ouslimani, Z. Ameur. Noise- robust texture features using local directional rank coding . *Advances in computer science and engineering* vol.15, no.1-2 ,pp. 1-12, 2015.
- [Ouslimani et al, 2016] F. Ouslimani, A. Ouslimani , Z. Ameur, Directional Rank Coding For Multi-scale Texture Classification. *International Conference of Computing for Engineering and Science (ICCES'2016)* on 23-27 July 2016, Barcelone, Spain.

-
- [Ouslimani et al, 2018] F. Ouslimani, A. Ouslimani , Z. Ameer, Rotation-Invariant texture classification based on Directional Coding, Neural Computing and Applications, <https://doi.org/10.1007/s00521-018-3462-9> .
- [Pan et al, 2015] Z. Pan , H. Fan , L. Zhang. Texture Classification Using Local Pattern Based on Vector Quantization. IEEE Transactions on Image Processing ,vol.24 no.12 , pp.5379 - 5388, 2015.
- [Pentland,1984] A. Pentland. Fractal-based description of natural scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-9, pp. 661-674, 1984.
- [Petpon et al, 2009] A. Petpon, S.Srisuk. Face recognition with local line binary pattern In: International Conference on Image and Graphics, 2009, pp.533–539.
- [Pietikäinen et al, 2011] M. Pietikäinen, A. Hadid, G. Zhao, T. Ahonen. Computer Vision Using Local Binary Patterns, vol.40 of Computational Imaging and Vision.(Springer, London, 2011).
- [Pinamonti et al., 1989] B. Pinamonti, Picano, E., Ferdeghini, M.,N. Lattanzi, F., Slavich, G., Landini, L., Camerini, F., Benassi, A., Distante, et A. L'Abbate. Quantitative texture analysis in two-dimensional echo cardiography : Application to the diagnosis of myocardial amyloidosis. Journal of the American College of Cardiology, vol.14, no.3, pp. 666–671,1989.
- [Pok et al,2005] G.Pok, K.H.Ryu , J. Lyu . Rotation and Gray-Scale Invariant Classification of Textures Improved by Spatial Distribution of Features. Lecture Notes in Computer Science, vol. 3588, pp. 250-259, 2005.
- [Porter et al, 1997] R. Porter and N. Canagarajah. Robust rotation invariant texture classification. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vols. 1–5. Jun. 1997, pp. 3157–3160.
- [Prasetiyo et al, 2010] T. Prasetiyo , M. Khalid , R. Yusof , F. Meriaudeau . A Comparative Study of Feature Extraction Methods for Wood Texture Classification . Signal-Image Technology and Internet-Based Systems (SITIS), 2010 Sixth International Conference on, 15-18 Dec. 2010, pp. 23 – 29, Kuala Lumpur.
- [Qi et al, 2014] X. Qi, R.Xiao, C.Li, Y.Qiao, J.Guo,X.Tang. Pairwise rotation invariant cooccurrence local binary pattern, IEEE Trans.Pattern Anal.Mach.Intell.,vol.36 ,no.11, pp. 2199–2213,2014.
- [Qi et al, 2012] X.Qi, R.Xiao, J.Guo, L.Zhang. Pairwise rotation invariant cooccurrence local binary pattern.In: European Conference on Computer Vision (ECCV), pp. 158–171, 2012.
- [Qi et al, 2013] X.Qi, Y.Qiao, C.Li, J.J.Guo. Multiscale joint encoding of local binary patterns for texture and material classification. In: Proceedings of British Machine Vision Conference(BMVC), 2013.

- [Qin et al., 2008] L. Qin, Q. Zheng, S. Jiang, Q. Huang, W. Gao. Unsupervised texture classification : Automatically discover and classify texture patterns. *Image and Vision Computing*, vol.26, pp.647–656,2008.
- [Ren et al, 2013] J.Ren, X. Jiang , J. Yuan. Noise-resistant local binary pattern with an embedded error-correction mechanism. *IEEE Trans Image Process.*,vol.22,no.10, pp.4049-4060, 2013.
- [Roberti de Siqueira, 2013] F. Roberti de Siqueira , W. R. Schwartz , H Pedrini. Multi-scale gray level co-occurrence matrices for texture description. *Neurocomputing*, vol. 120, pp.336-345 , 2013 .
- [Schistad Solberg et Jain, 1997] A. Schistad Solberg, A. Jain. Texture fusion and feature selection applied to SAR imagery. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no.2, pp. 475 –479, 1997.
- [Serra ,1982] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.
- [Sharma, 2001] M. Sharma. Performance evaluation of image segmentation and texture extraction methods in scene analysis. Thesis for the degree of Master of Philosophy in Computer Science, University of Exeter, 2001.
- [Sklansky, 1978] J. Sklansky. Image segmentation and feature extraction. *IEEE Transactions on Systems, Man and Cybernetics*, vol.8, no. 4, pp. 237–247,1978.
- [Song et al, 2013] K-C Song , Y-H Yan , W-H Chen , X. Zhang. Research and Perspective on Local Binary Pattern, *Acta Automatica Sinica* ,vol. 39, no.6, pp.730-744, 2013.
- [Subrahmanyam, 2012] M. Subrahmanyam , R.P.Maheshwari , R.Balasubramanian .Local maximum edge binary patterns: A new descriptor for image retrieval and object tracking. *Signal Processing* ,vol.92 , pp.1467–1479, 2012.
- [Tan et al., 2010] X.Tan and B.Trigg. Enhanced local texture features sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image processing* , vol .19, no. 6, pp. 1635-1650, 2010.
- [Tomita et Tsuji , 1990] F.Tomita, S. Tsuji. *Computer Analysis of Visual Textures*. Kluwer Academic Publishers, Boston, 1990.
- [Tuceryan et Jain, 1998] M. Tuceryan, A. Jain, Texture analysis. In Chen, C. H., Pau, L. F. et Wang, P. S. P., éditeurs : *The Handbook of Pattern Recognition and Computer Vision*, pp. 207–248. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2nd edition, 1998.
- [Turner, 1986] M. R Turner. Texture discrimination by gabor functions. *Biological cybernetics*, vol. 55, no 2-3,pp.71–82. PMID : 3801538,1986.
- [Unser, 1984] M. Unser. Description statistique de la texture. Application à l’inspection automatique. Thèse de doctorat, EPFL, Lausanne,1984 .

[Vama et al, 2005] M. Varma, A. Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, vol. 62, no.1-2, pp.61-81,2005.

[Varma et al, 2009] M. Varma, A. Zisserman. A statistical approach to material classification using image patch Exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence* ,vol.31, no.11, pp.2032–2047,2009.

[Vipparthi, 2014] S. K. Vipparthi , S.K. Nagar. Expert image retrieval system using directional local motif XoR patterns. *Expert Systems with Applications*, vol.41 , pp.8016-8026, 2014.

[Voorhees et Poggio, 1987] H. Voorhees, T. Poggio. Detecting textons and texture boundaries in natural images. In: *Proceedings of the First International Conference on Computer Vision*, pp. 250-258, London, 1987.

[Vapnik,1995] V. Vapnik. Consistency of Learning Processes. In *The Nature of Statistical Learning Theory*: Springer New York, 1995, pp. 33-64.

[Wang et He, 1990] L.Wang, D.C. He. Texture classification using texture spectrum. *Pattern Recognition*, vol. 23, no.8.pp. 905–910, 1990.

[Wang et al, 2015] S. Wang ,Q. Wu , X. He , J. Yang. Local Ary Pattern and Its Extension for Texture Classification. *IEEE Transactions on Circuits and Systems for Video Technology* ,vol.25 ,no. 9 , pp.1495 - 1506, 2015.

[Weszka et al, 1976] J.S.Weszka, C.R. Dyer, A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Trans.Syst.Man Cybernet* , vol.6, pp.269-285, 1976.

[Xie, 2008] X. Xie. A review of recent advances in surface defect detection using texture analysis techniques. *ELCVIA Electronic Letters on Computer Vision and Image Analysis*, vol.7, no.3, pp.1–22, 2008.

[Yousfi et al, 2011] S. Yousefi, N. Kehtarnavaz. A new stochastic image model based on Markov random fields and its application to texture modeling. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 1285–1288.

[Zhang et al ,2002] J. Zhang, T. Tan . Brief review of invariant texture analysis methods. *Pattern recognition*, vol. 35, pp. 735-747, 2002.

[Zhang et al, 2007] L. Zhang, R. Chu, S. Xiang, and S. Z. Li. Face detection based on Multi-Block LBP representation. In *Proc. Int. Conf. Biometrics (ICB)*, 2007, pp. 11-18.

[Zhao et al, 2013] Y. Zhao , W. Jia , R-X Hu , H. Min . Completed robust local binary pattern for texture classification . *Neurocomputing*,vol. 106, pp.68–76, 2013.

[Zhao et al, 2012] Y. Zhao, D. Huang, W.Jia. Completed local binary count for rotation invariant texture classification. *IEEE Trans. Image Process.*, vol.21, no.10, pp.4492–4497,2012.

[Zhao et al, 2013] Y. Zhao , W. Jia , R.-X. Huc, H. Min. Completed robust local binary pattern for texture classification . *Neurocomputing* ,vol.106 , pp.68-76,2013.

[Zhou et al , 2008] H. Zhou , R. Wang, C. Wang. A novel extended local-binary-pattern operator for texture analysis. *Information Sciences*, vol.178, pp.4314–4325, 2008.

[Zucker, 1976] S. W Zucker. Toward a model of Texture. *Computer Graphics and Image Processing*,vol.5, pp. 190-202, 1976.

Résumé :

La première partie de l'étude traite les descripteurs de texture invariants en rotation basés sur le codage de rang régional(RRC). Le codage RRC consiste à calculer le rang de chaque pixel sur une région dont la forme et la taille dépendent du niveau de gris du pixel traité et de ses voisins. Le code RRC est obtenu en combinant le rang du pixel traité et l'information niveau de gris, cette dernière est calculée après seuillage globale de l'image. Huit paramètres invariants en rotation sont alors obtenus. Le RRC multi-résolutions est calculé en utilisant plusieurs fenêtres de recherche. L'histogramme des caractéristiques à la même taille pour les différentes échelles. Comparée avec la méthode des motifs locaux binaire(LBP), la méthode proposée permet d'obtenir de bonnes performances de classification sur les bases Outex et CURET.

La seconde partie concerne le codage local de rang directionnel (LDRC) et le codage directionnel (DC). Ces deux méthodes sont basées sur le signe de la différence entre le pixel central et ses voisins. Les méthodes LDRC et DC permettent d'extraire quatre informations directionnelles, représentant les rangs du niveau de gris du pixel central calculés sur quatre orientations sur une fenêtre de voisinage de taille 3x3 pixels. Les quatre rangs sont ensuite utilisés pour obtenir le code final. L'opérateur LDRC produit 81 descripteurs robustes au bruit par contre DC produit 30 paramètres invariants par rotation. Pour l'étude multi résolutions, les deux codages sont calculés en modifiant la taille de la fenêtre autour du pixel traité, le nombre de voisins est limité à huit dont chacun représente une moyenne locale des niveaux de gris. La taille de l'histogramme des paramètres est maintenue réduite et constante. Afin d'évaluer les deux méthodes, plusieurs tests portant sur la classification des textures sont effectués en utilisant différentes bases de données.

Appliquée sur les bases Brodatz, CURET et KTH-TIPS2b, la méthode LDRC permet d'obtenir les meilleurs taux de classification par rapport à d'autres variantes de LBP essentiellement en présence de bruit. La méthode DC a été évaluée sur une large gamme de textures, nous avons étudié différents problèmes liés à la reconnaissance de texture tels que l'effet de l'éclairage, la rotation et changement d'échelle, et aussi l'influence du nombre d'images d'apprentissages. Comparée avec d'autres méthodes de l'état de l'art, DC a acquis le taux de classification le plus élevé sur les bases de données CURET et KTH-TIPS2b. La méthode DC a des résultats prometteurs sur la base de données Outex.

Mots cléf : texture, classification, invariance par rotation, codage, bruit gaussien, LBP, rang directionnel, rang régional.

Abstract:

The first part of the study deals with the rotation invariant descriptors based on regional rank coding (RRC) for texture classification. The regional rank presents the rank of the gray level of each pixel in a region whose size and shape depend on the gray level of the treaty pixel and its neighbors. Rotation invariant features are obtained by combining the rank which is found and the treaty pixel code. This latter is calculated by global thresholding. Eight discriminates and rotation invariant features are then obtained. The RRC multi-scales is calculated using several search windows. The features size doesn't increase with scale and kept constant. Compared to LBP method, the proposed method achieves good texture classification on the Outex and CURET databases.

The second part concerns the local directional rank coding (LDRC) and the directional coding (DC). Both methods are based on the sign of the difference between the central pixel and its neighbors. LDRC and DC methods extract four directional information, that represent the rank order of the central pixel calculated on a window of 3x3 pixels size. The four ranks are then used to obtain the final code. The LDRC operator products 81 descriptors robust to noise. But DC products 30 rotation invariant features. For multi-resolution study, the two methods are calculated by altering the window size around a central pixel. The number of samples is restricted to eight neighbors by local averaging. The size of the histogram parameter is kept small and constant. To assess both methods, several tests for texture classification are performed using different databases.

Applied on Brodatz, CURET and KTH-TIPS2b databases, the LDRC method provides the best classification rate compared to other LBP variants essentially in the presence of noise. The DC method was evaluated over a wide range of textures. We studied different problems related to texture recognition such as the effect of illumination, rotation and scale change, and also the influence of the number of the training samples. Compared to the other state of the art methods, the DC acquired the highest classification rate on CURET and KTH-TIPS2 databases. It also had promising results on Outex database.

Keywords: texture, classification, rotation invariance, coding, gaussian noise, LBP, directional rank, regional rank.