

**Département de géomatique appliquée
Faculté des lettres et sciences humaines
Université de Sherbrooke**

**Classification multi-échelle d'images à très haute
résolution spatiale basée sur une nouvelle approche
texturale**

Alexandre Delahaye

**Thèse présentée pour l'obtention du grade de Philosophiæ
Doctor (Ph.D.) en télédétection**

Juin 2016

© Alexandre Delahaye, 2016

Composition du jury

Cette thèse a été évaluée par un jury composé des personnes suivantes :

Pr. Goze Bertin Béné, codirecteur de recherche (Département de géomatique appliquée, FLSH, Université de Sherbrooke)

Dr. Mickaël Germain, codirecteur de recherche (Département de géomatique appliquée, FLSH, Université de Sherbrooke)

Pr. Dong-Chen He, examinateur interne (Département de géomatique appliquée, FLSH, Université de Sherbrooke)

Pr. Norman T. O'Neill, examinateur interne (Département de géomatique appliquée, FLSH, Université de Sherbrooke)

Dr. Samuel Foucher, examinateur externe (Centre de Recherche en Informatique de Montréal)

Résumé

Face à l'accroissement de la résolution spatiale des capteurs optiques satellitaires, de nouvelles stratégies doivent être développées pour classifier les images de télédétection. En effet, l'abondance de détails dans ces images diminue fortement l'efficacité des classifications spectrales; de nombreuses méthodes de classification texturale, notamment les approches statistiques, ne sont plus adaptées. À l'inverse, les approches structurelles offrent une ouverture intéressante : ces approches orientées objet consistent à étudier la structure de l'image pour en interpréter le sens. Un algorithme de ce type est proposé dans la première partie de cette thèse. Reposant sur la détection et l'analyse de points-clés (KPC : *KeyPoint-based Classification*), il offre une solution efficace au problème de la classification d'images à très haute résolution spatiale. Les classifications effectuées sur les données montrent en particulier sa capacité à différencier des textures visuellement similaires.

Par ailleurs, il a été montré dans la littérature que la fusion évidentielle, reposant sur la théorie de Dempster-Shafer, est tout à fait adaptée aux images de télédétection en raison de son aptitude à intégrer des concepts tels que l'ambiguïté et l'incertitude. Peu d'études ont en revanche été menées sur l'application de cette théorie à des données texturales complexes telles que celles issues de classifications structurelles. La seconde partie de cette thèse vise à combler ce manque, en s'intéressant à la fusion de classifications KPC multi-échelle par la théorie de Dempster-Shafer. Les tests menés montrent que cette approche multi-échelle permet d'améliorer la classification finale dans le cas où l'image initiale est de faible qualité. De plus, l'étude effectuée met en évidence le potentiel d'amélioration apporté par l'estimation de la fiabilité des classifications intermédiaires, et fournit des pistes pour mener ces estimations.

Abstract

Classifying remote sensing images is an increasingly difficult task due to the availability of very high spatial resolution (VHSR) data. The amount of details in such images is a major obstacle to the use of spectral classification methods as well as most textural classification algorithms, including statistical methods. However, structural methods offer an interesting alternative to this issue: these object-oriented approaches focus on analyzing the structure of an image in order to interpret its meaning. In the first part of this thesis, we propose a new algorithm belonging to this category: KPC (Key-Point-based Classification). KPC is based on keypoint detection and analysis and offers an efficient answer to the issue of classifying VHSR images. Tests led on artificial and real remote sensing images have proven its discriminating power.

Furthermore, many studies have proven that evidential fusion (based on Dempster-Shafer theory) is well-suited to remote sensing images because of its ability to handle abstract concepts such as ambiguity and uncertainty. However, few studies did focus on the application of this theory to complex textural data such as structural data. This issue is dealt with in the second part of this thesis; we focused on fusing multi-scale KPC classifications with the help of Dempster-Shafer theory. Tests have shown that this multi-scale approach leads to an increase in classification efficiency when the original image has a low quality. Our study also points out a substantial potential for improvement gained from the estimation of intermediate classifications reliability and provides ideas to get these estimations.

Table des matières

Résumé.....	ii
Abstract.....	iii
Liste des figures.....	viii
Liste des tableaux.....	x
Liste des annexes.....	xi
Notations.....	xii
Abréviations.....	xiii
Définitions.....	xiv
Remerciements.....	xv
1. Introduction générale.....	1
1.1. Contexte.....	1
1.2. Problématique.....	2
1.3. Objectifs.....	5
1.3.1. Objectif général.....	5
1.3.2. Objectifs secondaires.....	5
1.4. Hypothèses de recherche.....	6
1.5. Présentation du plan de la thèse.....	6
2. Cadre théorique.....	8
2.1. Bilan des méthodes actuelles en classification texturale.....	8
2.1.1. Méthodes statistiques.....	9
2.1.1.1. Statistiques du premier ordre.....	9
2.1.1.2. Statistiques du second ordre.....	9
2.1.2. Méthodes fréquentielles.....	12
2.1.2.1. Filtres basiques.....	12
2.1.2.2. Transformée de Fourier.....	13

2.1.2.3. Filtres de Gabor.....	14
2.1.2.4. Transformée en ondelettes.....	15
2.1.3. Méthodes à base de modèles.....	15
2.1.3.1. Champs aléatoires.....	16
2.1.3.2. Fractales.....	17
2.1.4. Méthodes géométriques.....	18
2.2. Éléments de vision par ordinateur.....	19
2.2.1. Généralités sur la détection et description de points-clés.....	19
2.2.2. Détection de points-clés.....	21
2.2.2.1. SIFT.....	22
2.2.2.2. SURF.....	27
2.2.3. Descripteurs de points-clés.....	28
2.2.3.1. SIFT.....	28
2.2.3.2. SURF.....	29
2.2.4. Points-clés et classification d'images.....	30
2.2.5. Statistiques spatiales ponctuelles.....	30
2.2.5.1. Statistiques du premier ordre.....	31
2.2.5.2. Statistiques du second ordre : fonction K de Ripley.....	31
2.3. Fusion de données.....	33
2.3.1. Contexte et définitions.....	33
2.3.1.1. Imperfections de l'information.....	35
2.3.1.2. Niveaux de fusion.....	37
2.3.2. Modèles probabilistes.....	38
2.3.3. Modèles possibilistes.....	39
2.3.4. Modèles évidentiels : théorie de Dempster-Shafer.....	40
2.3.4.1. Initialisation.....	42
2.3.4.2. Combinaison.....	45
2.3.4.3. Décision.....	49
2.3.4.4. Extensions à la théorie de Dempster-Shafer.....	51
2.4. Conclusion partielle.....	52
3. Méthodologie et données.....	53
3.1. Méthodologie.....	53
3.1.1. Classification texturale.....	54
3.1.1.1. Algorithme proposé.....	54

3.1.1.2. Algorithmes comparatifs.....	62
3.1.2. Fusion d'images.....	64
3.1.2.1. Généralités.....	64
3.1.2.2. Initialisation des fonctions de masse.....	65
3.1.2.3. Règles de combinaison.....	69
3.1.2.4. Règles de décision.....	70
3.1.3. Évaluation des performances.....	70
3.1.3.1. Taux de classification.....	70
3.1.3.2. Coefficient kappa.....	70
3.1.3.3. Matrice de confusion.....	71
3.2. Données.....	71
3.2.1. Mosaïque IKONOS.....	71
3.2.2. Mosaïque WorldView-2.....	72
3.2.3. Placette IKONOS.....	74
3.3. Conclusion partielle.....	75
4. Résultats.....	76
4.1. Classifications texturales.....	76
4.1.1. Influence des paramètres.....	76
4.1.2. Comparaison avec d'autres algorithmes de classification.....	84
4.1.2.1. Mosaïque IKONOS.....	84
4.1.2.2. Mosaïque WorldView-2.....	93
4.1.2.3. Placette IKONOS.....	97
4.2. Fusion d'images.....	100
4.2.1. Résultats généraux.....	100
4.2.2. Images bruitées.....	108
4.2.3. Fiabilité des sources.....	110
4.3. Discussion et conclusion partielle.....	114
5. Conclusion.....	118
5.1. Recherche effectuée.....	118
5.2. Limitations.....	121
5.3. Recommandations.....	122
5.3.1. Détection de frontières.....	122

5.3.2. Estimation du paramètre γ	122
5.3.3. Estimation du nombre de classes.....	123
5.3.4. Prise en compte des hypothèses composées.....	123
5.3.5. Diminution du temps de calcul.....	124
5.4. Perspectives.....	124
5.4.1. Modèles de fiabilité.....	124
5.4.2. Ajout de données multispectrales.....	124
5.4.3. Images à plus haute résolution spatiale.....	125
5.4.4. Classification par apprentissage.....	125
6. Annexes.....	126
7. Références.....	135

Liste des figures

Figure 2.1 : (a) Image 5×5 à trois niveaux de gris (0, 1 et 2). (b) Matrice de co-occurrence Pd pour $d = (1,1)$. Exemple en jaune. Tiré de Jain et al., 1995.....	10
Figure 2.2 : (a) Filtre laplacien (passe-haut). (b) Filtre gaussien (passe-bas). (c) Filtre moyenneur (passe-bas).....	13
Figure 2.3 : (a) Exemple de texture. (b) Densité spectrale de puissance associée.....	14
Figure 2.4 : banc de filtres de Gabor.....	15
Figure 2.5 : filtre LoG discret de taille 5×5 ($\sigma = 1$).....	18
Figure 2.6 : approche multi-échelle de l'algorithme SIFT. La partie à gauche représente l'image étudiée, celle de droite représente l'image filtrée à différentes échelles (D) (Lowe, 2004).....	23
Figure 2.7 : exemple de points d'intérêt SIFT détectés et de leur orientation.....	26
Figure 2.8 : exemples d'approximations de filtres gaussiens (dérivée seconde) utilisés par l'algorithme SURF. (a) Direction verticale. (b) Direction diagonale principale. Tiré de (Bay et al., 2006).....	27
Figure 2.9 : exemple de points d'intérêt SURF détectés et de leur orientation.....	28
Figure 2.10 : illustration de la méthode de calcul des descripteurs SIFT (Lowe, 2004)	29
Figure 2.11 : exemples de processus ponctuels. (a) Processus de Poisson homogène. (b) Processus de Poisson suivant un gradient circulaire. (c) Processus de Strauss.....	32
Figure 2.12 : illustration du comportement de la fonction K de Ripley en fonction de la distribution des points (tiré de Ripley, 1976).....	33
Figure 2.13 : schéma général d'un système de fusion d'images.....	35
Figure 2.14 : représentation des niveaux de fusion (Pohl et Van Genderen, 1998).....	38
Figure 2.15 : étapes du traitement évidentiel, précédées de l'identification des sources.....	42
Figure 3.1 : organigramme méthodologique général (intrants en bleu; KPC : keypoint-based classification).....	53
Figure 3.2 : étapes principales de l'algorithme de classification texturale.....	54
Figure 3.3 : exemple de points-clés SIFT détectés par l'algorithme SIFT++ sur une texture WorldView-2 (la taille des cercles indique l'échelle du point-clé, les couleurs n'ont pas de signification particulière).....	55
Figure 3.4 : exemple de répartition des voisinages au sein de l'image principale.....	58
Figure 3.5 : exemple d'une itération de l'algorithme de calcul de statistiques spatiales, pour cinq rayons.....	59
Figure 3.6 : organigramme méthodologique de l'étape de fusion multi-échelle. Les flèches pointillées désignent des étapes optionnelles.....	65
Figure 3.7 : image IKONOS de base.....	71
Figure 3.8 : mosaïque IKONOS (vérité-terrain à droite).....	72
Figure 3.9 : image WorldView-2 de base.....	73

Figure 3.10 : mosaïque WorldView-2 (vérité-terrain à droite).....	73
Figure 3.11 : placette IKONOS (vérité-terrain à droite).....	74
Figure 4.1 : taux de bonne classification en fonction du nombre de points-clés détectés. Les paramètres sont : $s = 5$, $w = 161$, $nC-KP = 20$, $nPCA = 0$, $R = [8, 16, 24, 32, 40]$	79
Figure 4.2 : TBC, kappa et durée de calcul en fonction du nombre de classes de points-clés nC-KP. Les autres paramètres sont : $s = 5$, $w = 161$, $nPCA = 0$, $R = [8,$ $16, 24, 32, 40]$	80
Figure 4.3 : TBC (en rouge), kappa (en bleu) et temps d'exécution (en vert) en fonction de la taille de voisinage w . Les autres paramètres sont : $s = 2$, $nC-KP = 20$, $nPCA$ $= 0$, $R = [8, 16, 24, 32, 40]$	81
Figure 4.4 : TBC en fonction de nPCA (courbe bleue). La ligne orange indique le TBC obtenu sans ACP.....	83
Figure 4.5 : résultats de la classification non dirigée de la mosaïque IKONOS.....	84
Figure 4.6 : classification KPC-SURF de la mosaïque IKONOS : classification normale (a) et classification avec pixels ambigus (en gris) (b).....	87
Figure 4.7 : différences de TBC entre classifications dirigées (k-NN en bleu, k-means en rouge) et non dirigées (k-means). Les pourcentages représentent les TBC des classifications dirigées.....	88
Figure 4.8 : résultats de la classification dirigée de la mosaïque IKONOS.....	90
Figure 4.9 : comparaison des temps de calcul de chaque étape entre les modes dirigés (k-NN) et non dirigés (k-means), pour les variantes SIFT et SURF.....	92
Figure 4.10 : résultats de la classification non dirigée de la mosaïque WorldView-2....	94
Figure 4.11 : classification KPC-SIFT de la mosaïque WorldView-2: classification normale (a) et classification avec pixels ambigus (en gris) (b).....	97
Figure 4.12 : résultats de la classification non dirigée de la placette IKONOS.....	99
Figure 4.13 : classification KPC-SIFT de la placette IKONOS : classification normale (a) et classification avec pixels ambigus (en gris) (b).....	99
Figure 4.14 : mosaïque IKONOS : résolutions 100 % (a), 50 % (b) et 25 % (c).....	101
Figure 4.15 : légende des hypothèses simples et composées.....	105
Figure 4.16 : mosaïque IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison conjonctive, décision par maximum de crédibilité) avec hypothèses simples (b) et composées (c), répartition du conflit (d).....	106
Figure 4.17 : mosaïque WorldView-2: classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison robuste, décision par maximum de plausibilité) avec hypothèses simples (b) et composées (c), répartition du conflit (d).....	107
Figure 4.18 : placette IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison conjonctive, décision par maximum de crédibilité) avec hypothèses simples (b) et composées (c),	

répartition du conflit (d).....	107
Figure 4.19 : Δ TBC en fonction de TBCsource pour chaque image.....	109
Figure 4.20 : mosaïque IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c).....	112
Figure 4.21 : mosaïque WorldView-2 : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c).....	113
Figure 4.22 : placette IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c).....	113
Figure 6.1 : masse maximale moyenne (γ , en bleu) et espérance du conflit (γ , en rouge) en fonction de γ	131
Figure 6.2 : Classifications des images source multi-échelle (colonne de gauche : résolution = 100 %, colonne centrale : résolution = 50 %, colonne de droite : résolution = 25 %).....	134

Liste des tableaux

Tableau 2.1 : indicateurs de Haralick.....	11
Tableau 3.1 : Paramètres SIFT utilisés.....	56
Tableau 3.2 : Paramètres SURF utilisés.....	56
Tableau 4.1 : paramètres utilisés (les paramètres analysés par la suite sont en gras).	77
Tableau 4.2 : influence du paramètre S sur le nombre de points-clés et le TBC.....	78
Tableau 4.3 : influence du paramètre t sur le nombre de points-clés et le TBC.....	78
Tableau 4.4 : TBC moyen en fonction des paramètres R et w. Pour chaque valeur de w, le meilleur TBC moyen est mis en gras.....	83
Tableau 4.5 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la mosaïque IKONOS. Les meilleures valeurs sont indiquées en gras.....	85
Tableau 4.6 : matrice de confusion de la classification de la mosaïque IKONOS par l'algorithme KPC-SURF.....	86
Tableau 4.7 : taux de bonne classification, coefficients kappa et durées d'exécution de la classification dirigée (k-NN) de la mosaïque IKONOS. Les meilleures valeurs sont indiquées en gras.....	89
Tableau 4.8 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la mosaïque WorldView-2. Les meilleures valeurs sont indiquées en gras.....	94
Tableau 4.9 : densité de points-clés détectés, par image et par algorithme de détection (en nombre de points-clés par 1 000 pixels).....	95
Tableau 4.10 : matrice de confusion de la classification de la mosaïque WorldView-2	

par l'algorithme KPC-SIFT. Un pixel sur deux a été classifié dans chaque direction (25 % de tous les pixels).....	96
Tableau 4.11 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la placette IKONOS. Les meilleures valeurs sont indiquées en gras.....	98
Tableau 4.12 : Δ TBC (%) issus de la fusion multi-échelle de la mosaïque IKONOS, sans coefficient de fiabilité. Les meilleures valeurs sont en gras.....	101
Tableau 4.13 : Δ TBC (%) issus de la fusion à multi-échelle de la mosaïque WorldView-2, sans coefficient de fiabilité. Les meilleures valeurs sont en gras..	102
Tableau 4.14 : Δ TBC (%) issus de la fusion à multi-échelle de la placette IKONOS, sans coefficient de fiabilité. La meilleure valeur est en gras.....	102
Tableau 4.15 : TBCsource (classification KPC-SIFT non dirigée) et Δ TBC pour chaque image (en %).....	104
Tableau 4.16 : Δ TBC et TBCsource (entre parenthèses) pour chaque image (en %)....	108
Tableau 4.17 : Δ TBC par rapport aux classifications sources (en %). Entre parenthèses : gains de TBC apportés par l'utilisation de modèles d'estimation du paramètre de fiabilité α	111
Tableau 4.18 : TBCsource (classification KPC-SIFT non dirigée) et Δ TBC, incluant l'estimation du paramètre de fiabilité, pour chaque image (en %).....	114
Tableau 6.1 : TBC (en %) et coefficients kappa (entre parenthèses) des images source multi-échelle.....	132
Tableau 6.2 : Résultats de la fusion à 3 échelles de la mosaïque IKONOS originale, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa.....	132
Tableau 6.3 : Résultats de la fusion à 3 échelles de la mosaïque WorldView-2, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa. Les meilleures valeurs sont en gras.....	133
Tableau 6.4 : Résultats de la fusion à 3 échelles de la placette IKONOS, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa. Les meilleures valeurs sont en gras.....	133

Liste des annexes

Annexe 1 – Étude du paramètre de distance γ	127
Annexe 2 – Détails des résultats de la fusion multi-échelle.....	132

Notations

C	Nombre de classes
d	Fonction de distance euclidienne
I	Image vérité (vérité-terrain)
$I_{x,y}$	Pixel de l'image I , colonne x , ligne y
K	Fonction K de Ripley
k	Conflit
S	Nombre de sources
X	Image observée
$X_{x,y}$	Pixel de l'image X , colonne x , ligne y
Θ	Ensemble des classes (ou hypothèses)
θ_i	Classe (hypothèse) numéro i ($1 \leq i \leq K$)
κ	Coefficient kappa de Cohen
ΔTBC	Différence entre le TBC à l'issue de la fusion multi-échelle et le TBC de la classification source (sans fusion).

Abréviations

ACP	Analyse en Composantes Principales
DST	Théorie de Dempster-Shafer
GLCM	Gray-Level Co-occurrence Matrix (indicateurs de Haralick)
KPC	Keypoint-based Classification
SFTA	Segmentation-based Fractal Texture Analysis
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
TBC	Taux de Bonne Classification
THRS	Très Haute Résolution Spatiale

Définitions

La liste suivante contient les définitions de certains termes spécifiques utilisés dans cette thèse :

Descripteur textural : vecteur caractérisant une texture, issu d'un algorithme d'analyse texturale.

Espace textural : espace vectoriel dans lequel peuvent être analysés les descripteurs texturaux. Il est de même dimension que ces derniers.

Partition : agrégat de vecteurs au sein d'un espace multidimensionnel. Traduction de l'anglais *cluster*.

Partitionnement : action de partager un ensemble de vecteurs en partitions. Traduction de l'anglais *clustering*.

Segment : ensemble spatialement connexe de pixels de même classe.

Voisinage : ensemble de pixels spatialement proches (selon une règle à définir) d'un pixel central.

Remerciements

Je tiens d'abord à exprimer ma profonde gratitude à mes codirecteurs, le Pr. Goze Bertin Béné et M. Mickaël Germain, qui ont accepté de diriger ma thèse. Sans leur patience, leurs conseils avisés et leur aide logistique, cette thèse n'aurait pu être menée à terme.

Je remercie également les membres du jury, Pr. Dong-Chen He, Pr. Norman T. O'Neill et M. Samuel Foucher pour m'avoir fait l'honneur d'accepter la charge d'évaluateurs.

Je remercie toute l'équipe de Géoimage Solutions pour les bons moments passés ensemble en parallèle de ce doctorat, avec une mention particulière à Claire et Yves pour les échanges fructueux que nous avons pu avoir concernant les études.

Pour finir, je remercie mes parents et le reste de ma famille pour m'avoir soutenu durant cette longue aventure canadienne. Merci à M. P. d'avoir assuré une animation quotidienne, à sa manière. Mes plus tendres remerciements vont à Paméla, pour avoir toujours été présente et m'avoir encouragé jusqu'au bout.

1. Introduction générale

1.1. Contexte

La classification d'images fait partie des problèmes les plus courants de télédétection. Elle consiste à regrouper les pixels d'une image en classes, dans l'espoir que les classes ainsi formées représentent au mieux l'occupation réelle du sol : végétation, champs, zones urbaines, milieux humides, etc. La qualité des données satellitaires est désormais suffisamment bonne pour permettre le développement d'une multitude d'applications exploitant la classification d'images : études environnementales, gestion des catastrophes naturelles, planification agricole, applications militaires, etc.

Parallèlement, la quantité grandissante d'images disponibles a encouragé la mise au point de méthodes de fusion d'images tirant parti de la redondance entre ces dernières pour accroître la qualité de leur analyse. La combinaison de plusieurs images représentant une même scène peut en effet permettre d'identifier avec une plus grande fiabilité les types d'occupation du sol, en mettant notamment en évidence les éventuels mélanges de classe, mais également d'améliorer la précision spatiale du découpage du sol.

Face à de tels défis, des méthodes de classification et de fusion de plus en plus sophistiquées doivent être mises en œuvre. Le but de cette recherche est de proposer un nouvel algorithme de classification non dirigé spécifiquement adapté aux images de télédétection à très haute résolution spatiale (THRS) – c'est-à-dire de limites de résolution inférieures à 1 m pour la bande panchromatique. Cet algorithme est basé sur une analyse texturale orientée objet, qui permet de différencier des thèmes visuellement très similaires. Ce type d'approche repose sur l'extraction de traits porteurs de sens (bâtiments, arbres, rochers, etc.) pour caractériser la texture présente au sein d'une zone donnée.

L'originalité de cet algorithme est l'étude de la répartition spatiale de ces traits, en plus de leurs simples caractéristiques géométriques et radiométriques, pour différen-

cier les textures présentes dans une image satellitaire.

Afin d'améliorer la qualité des classifications obtenues, une méthode de fusion multi-échelle est également proposée. Elle est basée sur la théorie de Dempster-Shafer, qui offre un cadre théorique intéressant pour modéliser les défauts de classifications courants en télédétection que sont l'imprécision et l'incertitude¹. Ces défauts, souvent ignorés ou sous-estimés, sont pourtant d'autant plus présents que la résolution spatiale de l'image est élevée, du fait de la netteté des zones de transitions entre différents milieux.

La méthode de fusion proposée est générique et conviendrait tout aussi bien à un contexte multi-source ou multi-temporel. Plusieurs innovations y sont incluses, telles qu'une nouvelle méthode d'initialisation des masses d'hypothèses doubles et deux modèles d'estimation de la fiabilité des classifications sources. De plus, les méthodes les plus courantes de combinaison et de décision ont été comparées entre elles afin d'évaluer leurs performances dans le cadre de la télédétection.

1.2. Problématique

Historiquement, les méthodes de classification se sont d'abord basées sur des approches pixel par pixel, radiométrique (étude des niveaux de gris) puis spectrale (étude de la valeur du pixel pour chaque bande de l'image). Ces approches se justifiaient par leur complexité réduite et la faible résolution spatiale des images alors disponibles. Cependant, la résolution spatiale des capteurs ne cesse de s'affiner, au point que l'on parle maintenant d'images à très haute résolution spatiale (THRS). Ainsi IKONOS (1999), Quickbird (2001) et GeoEye-1 (2008) offrent des limites de résolution spatiale d'environ 80, 60 et 50 cm respectivement en mode panchromatique et 4, 2,4 et 1,65 m en mode multispectral; WorldView-3 (2014) surpasse encore ces spécifications avec une limite de résolution maximale de 31 cm.

¹ Voir définitions en section 2.3.1.1.

L'augmentation de la résolution spatiale des images de télédétection a pour principal effet d'accroître la variabilité interne des classes observées. Comme le soulignent Carleer *et al.* (2005), l'augmentation de la variabilité interne diminue la séparabilité des classes de l'espace spectral, d'autant plus que les images THRS ont généralement une faible résolution spectrale (maximum 4 bandes, excepté WorldView-2 et 3). La quantité de détails contenus dans de telles images rend donc inefficaces la plupart des algorithmes de classification habituellement utilisés. Ces algorithmes sont généralement basés sur l'exploitation pixel par pixel des caractéristiques spectrales de l'image. De plus, Carleer *et al.* soulignent également que les méthodes de segmentation contextuelle (approche orientée objet) gèrent difficilement les zones de transition entre objets et mènent fréquemment à une sur-segmentation.

Par ailleurs, Puissant *et al.* (2005) ont montré l'importance de la prise en compte des caractéristiques texturales lorsque la limite de résolution spatiale est de l'ordre du mètre par pixel. Cependant, les méthodes classiques d'analyse de texture (et particulièrement les matrices de co-occurrence, utilisées par ces auteurs) ne sont pas adaptées à l'étude d'images dont la limite de résolution est de l'ordre de la dizaine de centimètres en raison de la trop grande hétérogénéité des pixels. En effet, Blaschke (2010) a mis en évidence l'importance d'une approche orientée objet dans ce contexte. Il rappelle de plus qu'une telle approche doit reposer sur l'extraction de descripteurs, la détection de frontières ainsi que sur des méthodes classiques de classification; or, bien qu'il existe depuis longtemps de nombreux algorithmes correspondant à ces différents axes, il reste difficile de bâtir une méthode d'analyse texturale orientée objet fiable (adaptée aux images THRS) et robuste (invariance face à la rotation, l'éclairement, l'inclinaison, etc.). Dans ce contexte, il est important de proposer une nouvelle méthode d'analyse texturale capable d'extraire des informations utiles d'images très détaillées, en adoptant une approche orientée objet au lieu d'une simple analyse statistique des niveaux de gris ou d'une approche de type géométrique.

Parallèlement, le besoin d'exploiter le maximum d'informations disponibles a encouragé la création de méthodes de fusion d'images de plus en plus sophistiquées : moyennage, analyse en composantes principales, filtrage passe-haut, décomposition pyramidale, transformée en ondelettes discrète, etc. (Thomas *et al.*, 2008). Outre le fait que ces méthodes emploient exclusivement des approches spectrales et non texturales, leur inconvénient majeur est qu'elles ne gèrent ni l'incertitude, ni l'imprécision portant sur les données. Autrement dit, le caractère probabiliste de l'information reçue est généralement négligé et on suppose que les images à fusionner correspondent parfaitement à la réalité, ce qui n'est pas du tout adapté à la fusion d'images de télédétection issues d'une classification.

Pour pallier cet inconvénient, ont été développées les méthodes de fusion bayésienne puis de fusion floue, qui ne prennent toutefois pas bien en compte l'imprécision des images à analyser. En télédétection, l'imprécision est surtout marquée sur les zones de transition ou les zones mixtes, où le problème est d'évaluer les limites entre classes ou entre mélanges de classes, et non seulement d'estimer la proportion de chaque classe; elle ne doit pas être confondue avec l'incertitude, qui décrit la conformité de l'information à la réalité (Bloch et Maître, 1994). Encore aujourd'hui, ce problème est rarement désigné de façon explicite dans la littérature, bien que l'augmentation de la résolution des images analysées le rende de plus en plus présent.

La théorie des croyances de Dempster-Shafer (DST – Shafer, 1976) permet une modélisation très souple et très riche des classes que l'on cherche à détecter, par le biais de fonctions de masse qui sont définies non seulement sur des singletons (classes seules) mais également sur des ensembles plus vastes (hypothèses composées). Elle représente bien l'imperfection des connaissances dont on dispose à partir des données source : incertitude, imprécision mais aussi ignorance, ambiguïté, conflit (Bloch et Maître, 1994). Toutefois, bien que la DST ait déjà été appliquée en fusion d'images, la spécificité des images THRS évoquée plus haut n'est pas prise en compte : l'image est étudiée pixel

par pixel (ou groupe de pixels par groupe de pixels) à partir de ses caractéristiques spectrales, sans analyse orientée objet.

En résumé, les propriétés des images THRS rendent obligatoire l'utilisation d'un algorithme d'analyse texturale avancé pour effectuer une classification ou une segmentation. Pour ce faire, il est préférable d'employer une approche orientée objet qui tienne compte du contenu sémantique de l'image plutôt que d'analyser uniquement la répartition des niveaux de gris des pixels, ou son spectre de Fourier. De plus, la classification pourrait être renforcée par la fusion des images issues de plusieurs capteurs à l'aide de la théorie de l'évidence : une telle fusion, effectuée sur les images pré-classifiées, permettrait d'exploiter et de représenter au mieux l'information associée à une zone. L'algorithme de classification ainsi proposé est de type non dirigé, tant au niveau de l'étape d'analyse texturale que celle de la fusion de données; toutefois, moyennant quelques ajustements mineurs, il est possible de le faire fonctionner de manière dirigée afin d'augmenter ses performances.

1.3. Objectifs

1.3.1. Objectif général

L'objectif général de cette thèse est de développer une méthode de classification basée d'une part sur l'analyse texturale d'images panchromatiques à très haute résolution spatiale (< 1 m), et d'autre part sur une fusion multi-échelle exploitant la théorie de l'évidence.

1.3.2. Objectifs secondaires

L'objectif général se divise en deux sous-objectifs : premièrement, développer un algorithme d'analyse texturale orienté objet adapté aux images de télédétection THRS, en mettant l'accent sur une classification complexe pour les images dans le domaine de la foresterie.

Cet algorithme devra répondre à certains critères : robustesse face aux images de télédétection (impliquant une invariance de contraste, de luminosité, d'échelle et de rotation), séparabilité des classes (capacité à différencier des textures similaires), complexité (durée de calcul), nombre de paramètres à régler manuellement.

Deuxièmement, il s'agit de mettre en œuvre une méthode de fusion des caractéristiques texturales issues de plusieurs échelles d'une même image, à l'aide de la théorie de l'évidence. Il s'agit de tenir compte du mieux possible des caractéristiques des données à fusionner, qui sont les descripteurs texturaux.

1.4. Hypothèses de recherche

Deux hypothèses scientifiques sont à la base de ce projet de recherche. La première est que l'extraction de caractéristiques texturales à l'aide d'un algorithme de recherche de points-clés de type SIFT améliore la classification d'images à très haute résolution spatiale (Puissant *et al.*, 2005). La seconde hypothèse est que l'utilisation de la théorie de l'évidence est bien adaptée à la fusion d'images de télédétection issues de plusieurs capteurs (Bouakache *et al.*, 2009; Lefebvre *et al.*, 2010) ou à la fusion multi-échelle, lorsqu'elles sont basées sur l'exploitation de descripteurs texturaux.

1.5. Présentation du plan de la thèse

La thèse s'articule sur cinq chapitres. Après la présente introduction sont exposées les bases théoriques nécessaires à la compréhension générale de la thèse, incluant une revue des travaux existants en analyse de texture et un exposé succinct des méthodes de fusion d'images.

Le troisième chapitre traite de la méthodologie et des données utilisées, tandis que le chapitre suivant expose et discute les résultats obtenus, d'abord en matière de classification texturale puis de fusion multi-échelle.

Enfin, on trouvera dans la conclusion un résumé des recherches effectuées ainsi que des pistes d'amélioration et des recommandations pour d'éventuelles études complémentaires.

2. Cadre théorique

2.1. Bilan des méthodes actuelles en classification texturale

Lorsque l'on observe une image complexe, le premier constat que l'on fait le plus souvent est que l'image est composée de plusieurs couleurs et de plusieurs textures. Bien que le système visuel humain ait une compréhension intuitive du concept de texture, il est difficile d'en fournir une définition précise et universelle : c'est la raison pour laquelle de multiples définitions existent. Néanmoins, l'idée générale que l'on retrouve dans chaque définition est la répétition spatiale (périodique ou aléatoire) d'un motif ou de caractéristiques locales de l'image. Ainsi, Sklansky énonce : « *A region [...] has a constant texture if a set of local statistics or other local properties of the picture function are constant, slowly varying, or approximately periodic* » (Sklansky, 1978). Haralick définit plutôt une texture comme étant un assemblage non nécessairement régulier de primitives : « *An image texture is described by the number and types of its (tonal) primitives and the spatial organization or layout of its (tonal) primitives* » (Haralick, 1979). Enfin, une texture peut être multibande ou panchromatique.

De nombreuses autres définitions sont énumérées par Tuceryan et Jain (1998), prouvant ainsi que le concept de texture dépend à la fois de la perception de l'observateur et du domaine d'application concerné. Toutes les images ne présentent d'ailleurs pas de texture analysable (Karu *et al.*, 1996). En télédétection, l'analyse de texture a pour but d'identifier différents types de terrains (champs, forêt dense, milieu urbain, plan d'eau, etc) que l'on suppose avoir des propriétés visuelles suffisamment distinctes.

Les diverses définitions que l'on peut retrouver correspondent en réalité à plusieurs interprétations de la notion de texture. Ces différentes interprétations ont mené au développement d'algorithmes d'analyse de texture très variés, que l'on peut classer en quatre grandes catégories développées ci-après : les méthodes statistiques, fréquen-

tielles, à base de modèles et orientées objet (ou géométriques). L'objectif de ces méthodes est toujours le même : associer à une texture située dans le voisinage d'un pixel un ensemble de métriques, ou vecteur descripteur, afin d'effectuer une classification qui pourra éventuellement mener à une segmentation.

2.1.1. Méthodes statistiques

Les méthodes statistiques ont pour point commun d'étudier la distribution des niveaux de gris au sein d'une texture. Historiquement, ce sont les premières méthodes employées car ce sont les plus simples à implémenter. Aujourd'hui encore, elles sont largement répandues car elles ont un bon rapport efficacité/complexité.

2.1.1.1. *Statistiques du premier ordre*

Les méthodes statistiques du premier ordre consistent simplement à étudier les propriétés de l'histogramme des niveaux de gris d'une texture. Les propriétés les plus courantes sont : moyenne, médiane, mode, minimum, maximum, écart-type, etc. Peu d'informations spatiales sont prises en compte, ce type de méthode ne peut donc fonctionner que sur des textures très homogènes ou régulières. En pratique, les statistiques du premier ordre ne sont jamais utilisées seules pour caractériser une texture, car elles ne permettent pas de représenter la dépendance spatiale des niveaux de gris qui est un trait fondamental de l'analyse texturale; cependant, elles peuvent fournir un complément d'information utile, notamment dans le cas des images couleur.

2.1.1.2. *Statistiques du second ordre*

Contrairement aux statistiques du premier ordre, les statistiques du second ordre exploitent la dépendance spatiale des niveaux de gris. Plus précisément, les méthodes de ce type visent à évaluer dans quelle mesure l'intensité de l'image suit un motif périodique, dans une direction donnée.

La méthode de ce type la plus connue est basée sur l'analyse des matrices de co-occur-

rence, proposée par Haralick *et al.* (1973). Une matrice de co-occurrence est la représentation de la fréquence des différentes combinaisons de paires de pixels, dans une direction donnée \mathbf{d} . Un exemple est fourni par la figure 2.1 : la matrice de co-occurrence (b) indique qu'il existe dans l'image (a) deux paires de valeurs (1,0) dans la direction $\mathbf{d} = (1,1)$ (représentées en jaune). Une matrice de co-occurrence doit aussi être normalisée de manière à ce que la somme de ses éléments vaille 1 (d'où la division par 16 dans la figure 2.1 (b)).

Une matrice de co-occurrence a donc autant de lignes et de colonnes que le nombre de niveaux de gris de l'image analysée, ce qui entraîne une complexité très élevée et souvent inutile. Pour cette raison, on regroupe généralement les niveaux de gris en un faible nombre de classes (souvent huit) par partition de l'histogramme. De plus, il existe autant de matrices de co-occurrence que de directions analysées; on se contente donc habituellement d'utiliser quelques directions « standard », sauf cas particulier.

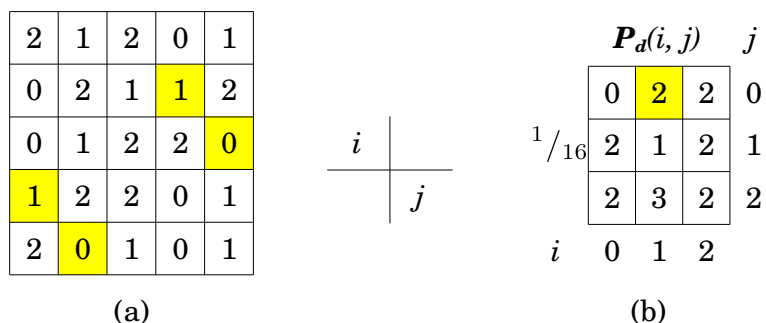


Figure 2.1 : (a) Image 5×5 à trois niveaux de gris (0, 1 et 2). (b) Matrice de co-occurrence \mathbf{P}_d pour $\mathbf{d} = (1,1)$. Exemple en jaune. Tiré de Jain *et al.*, 1995.

Une fois les matrices de co-occurrence calculées pour une image ou une portion d'image, on en extrait plusieurs indicateurs (dits de Haralick) tels que ceux présentés dans le tableau 2.1 :

Tableau 2.1 : indicateurs de Haralick.

Indicateur	Formule
Énergie	$\sum_i \sum_j P_d^2(i, j)$
Entropie	$-\sum_i \sum_j P_d(i, j) \log P_d(i, j)$
Contraste	$\sum_i \sum_j (i - j)^2 P_d(i, j)$
Homogénéité	$\sum_i \sum_j \frac{P_d(i, j)}{1 + i - j }$
Corrélation	$\frac{\sum_i \sum_j (i - \mu_x)(j - \mu_y) P_d(i, j)}{\sigma_x \sigma_y}$
Valeur absolue	$\sum_i \sum_j i - j P_d(i, j)$
Maximum	$\max_{i, j} P_d(i, j)$

$P_d(i, j)$ représente ici la valeur située à la ligne i et à la colonne j d'une matrice de co-occurrence. μ_x et μ_y représentent les moyennes des lignes et des colonnes, et σ_x et σ_y leurs écarts-types.

Ainsi, si N indicateurs et D directions sont choisis, un ensemble de $D \times N$ caractéristiques sera associé à chaque texture. Bien que potentiellement élevé, ce nombre reste faible par rapport à d'autres méthodes d'analyse de texture.

Le problème majeur présenté par cet outil est qu'il impose de faire un choix manuel des vecteurs d de direction ainsi que des indicateurs de Haralick utilisés. De plus, il ne fonctionne efficacement que dans le cas des textures régulières et n'est donc pas adapté aux textures complexes. C'est donc un outil simple et générique, mais limité.

Un autre outil d'analyse statistique du second ordre parfois utilisé est la matrice d'autocorrélation. Cette matrice, définie dans l'équation 2.1, permet de calculer la corréla-

tion (i.e. la similarité des niveaux de gris) entre les pixels de toutes les paires de pixels existantes et met très bien en évidence l'existence de motifs réguliers. Elle présente un pic à l'emplacement (0,0) puis décroît plus ou moins rapidement dans chaque direction suivant la rugosité de la texture. Dans le cas d'une texture très régulière, la matrice d'autocorrélation présente des pics à intervalles réguliers qui permettent de détecter la périodicité de la texture.

$$\rho(x, y) = \frac{\sum_{u=0}^N \sum_{v=0}^N \mathbf{I}(u, v) \mathbf{I}(u + x, v + y)}{\sum_{u=0}^N \sum_{v=0}^N \mathbf{I}^2(u, v)} \quad (2.1)$$

Toute l'information contenue dans l'image est donc concentrée dans une seule matrice; toutefois il est difficile d'extraire des descripteurs pertinents à partir de cette matrice, c'est pourquoi elle est relativement peu employée. En revanche, comme on le verra dans la prochaine partie, elle permet de calculer la densité spectrale de puissance d'une image, qui est un outil important.

Notons enfin l'existence d'approches basées sur le calcul de statistiques locales de l'image, tel que l'algorithme de spectre de texture (He et Wang, 1990).

2.1.2. Méthodes fréquentielles

Les méthodes fréquentielles ont comme point commun de s'intéresser à la répartition et à l'intensité des fréquences spatiales d'une image. Elles partent du constat que le système visuel humain est sensible aux fréquences spatiales (et particulièrement aux hautes fréquences). Contrairement aux méthodes statistiques qui considèrent une image comme un assemblage de pixels sans signification particulière, les méthodes fréquentielles cherchent donc à extraire une partie du sens de l'image.

2.1.2.1. *Filtres basiques*

Il est possible d'utiliser de simples filtres linéaires passe-haut ou passe-bas (filtre gaussien, filtre laplacien, etc.) pour caractériser certaines textures. L'idée est de calculer

certaines caractéristiques statistiques de l'image étudiée filtrée. Cela permet d'extraire des indicateurs simples liés par exemple à l'homogénéité ou à la quantité de variations brusques d'intensité dans l'image.

$$\begin{array}{ccc} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} & \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} & \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \\ \text{(a)} & \text{(b)} & \text{(c)} \end{array}$$

Figure 2.2 : (a) Filtre laplacien (passe-haut). (b) Filtre gaussien (passe-bas). (c) Filtre moyennneur (passe-bas).

Il est également possible d'utiliser des groupes de filtres afin de calculer des moments statistiques d'ordre quelconque (Tuceryan, 1994).

Unser et Eden (1990) proposent une approche plus subtile basée sur l'utilisation de filtres spatiaux non linéaires permettant de faire ressortir des caractéristiques texturales plus complexes, et donc d'améliorer la classification de ces textures.

Ces diverses solutions restent cependant peu employées car le filtrage est généralement trop peu paramétrable et trop rudimentaire.

2.1.2.2. *Transformée de Fourier*

La transformée de Fourier est l'outil d'analyse fréquentiel le plus courant. Elle permet d'exprimer une image sous la forme d'une somme d'ondes sinusoïdales de différentes fréquences et d'afficher la valeur des coefficients de ces ondes sous forme d'une image, qui représente l'espace fréquentiel. L'étude de l'espace fréquentiel met alors en évidence les fréquences dominantes de l'image (figure 2.3). Il est ensuite possible de caractériser l'image en calculant la proportion du signal qui se trouve dans chaque direction et chaque bande fréquentielle.

En pratique, on utilise la densité spectrale de puissance (DSP) plutôt que la transformée de Fourier, pour des raisons mathématiques. La DSP fournit pratiquement les

mêmes renseignements que la transformée de Fourier, mais son estimation de l'espace spectral convient mieux aux signaux finis et discrets. La DSP peut être simplement obtenue en calculant la transformée de Fourier discrète de la matrice d'autocorrélation.

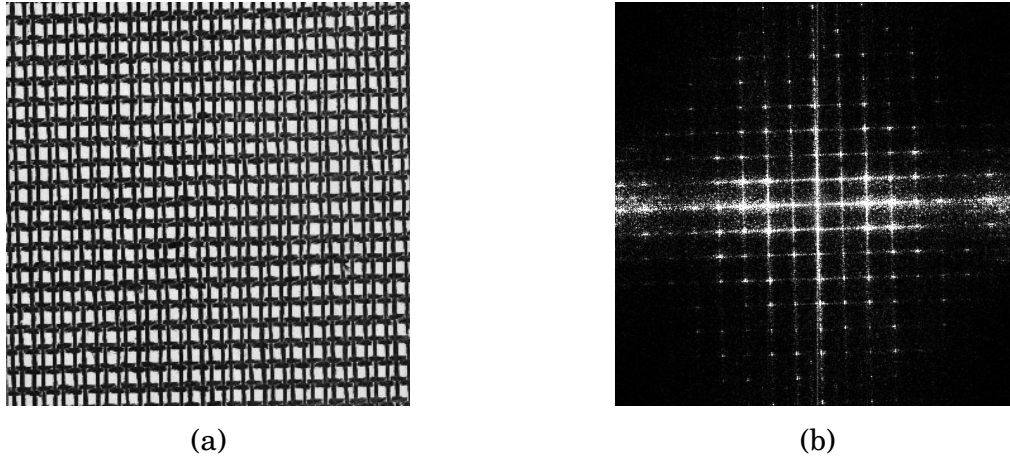


Figure 2.3 : (a) Exemple de texture. (b) Densité spectrale de puissance associée.

2.1.2.3. *Filtres de Gabor*

Contrairement à la méthode précédente qui se limite à une analyse globale de l'image, les filtres de Gabor autorisent une approche à la fois spatiale et fréquentielle. Il s'agit d'un ensemble de filtres complexes dont la forme générale est :

$$h(x, y) = \left(\frac{1}{2\pi\lambda\sigma^2} \right) \cdot \exp\left(-\frac{(x'/\lambda)^2 + y'^2}{2\sigma^2}\right) \cdot \exp(2\pi i(\nu x + \mu y)) \quad (2.2)$$

où $x' = x\cos(\Phi) + y\sin(\Phi)$, $y' = -x\sin(\Phi) + y\cos(\Phi)$ et $\lambda, \nu, \mu, \sigma$ et Φ sont des paramètres.

Ces filtres ont été exploités pour la première fois par Bovik *et al.* (1990) dans le cadre de l'analyse de textures. L'utilisation de tels filtres se base sur la vision humaine (Daugman, 1980), qui est sensible aux répétitions périodiques de motifs, d'abord à grande échelle puis à petite échelle. En faisant varier les paramètres de l'équation générale, on obtient un banc de filtres multi-fréquences, multi-échelles et multi-directionnels. De plus, il s'agit d'un filtre spatial, c'est-à-dire qu'on obtient le résultat du filtrage en convoluant le support du filtre avec chaque pixel de l'image.

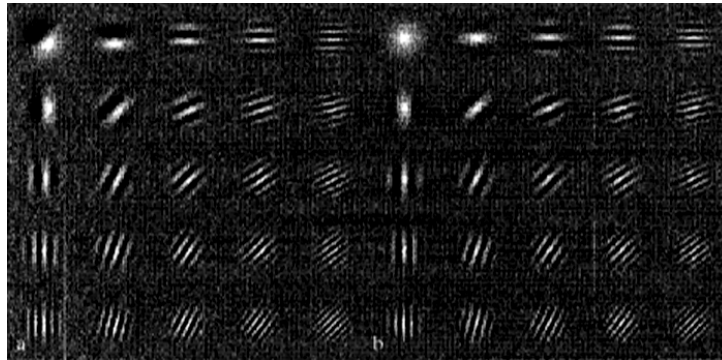


Figure 2.4 : banc de filtres de Gabor

L'analyse texturale se déroule de la manière suivante : l'image à caractériser est passée au travers de chacun des filtres séparément, donnant ainsi autant d'images de sortie qu'il y a de filtres. Une opération est ensuite effectuée sur chaque image de sortie (habituellement, la somme des niveaux de gris ou l'énergie de l'image) pour attribuer une valeur à chacune d'entre elles; le vecteur descripteur comporte donc autant d'éléments qu'il y a de filtres définis. L'inconvénient de ces filtres est qu'ils exigent de choisir à l'avance les fréquences à étudier, il est donc difficile de mettre au point des filtres généraux fonctionnant pour tous les types de textures.

2.1.2.4. *Transformée en ondelettes*

Comme les filtres de Gabor, la transformée en ondelettes permet une analyse spatiale et fréquentielle simultanément, ainsi que multi-échelle. Proposée par Mallat (1989), elle présente l'avantage de réduire considérablement le temps de calcul au détriment de la précision de l'analyse : en effet, seules trois directions peuvent être étudiées (horizontale, verticale et diagonale) et les échelles d'analyse suivent une progression géométrique, au détriment des hautes fréquences.

2.1.3. Méthodes à base de modèles

Ces méthodes ont pour point commun de tenter de développer des modèles de textures analytiques, basés sur des jeux de paramètres. Ces modèles permettent à la fois de dé-

crire des textures et de les synthétiser; les paramètres permettent alors de caractériser les textures analysées.

2.1.3.1. Champs aléatoires

Les champs aléatoires de Markov sont un outil populaire de modélisation de textures. Proposés par Cross et Jain (1983), ils reposent sur l'idée que le niveau de gris d'un pixel suit une loi de probabilité dépendante du niveau de gris des pixels voisins. Le théorème de Hammersley-Clifford permet d'affirmer qu'en matière d'imagerie numérique, un champ de Markov est également un champ de Gibbs, c'est-à-dire un champ pour lequel on peut définir une densité de probabilité globale. La probabilité qu'un pixel ait un certain niveau de gris s'exprime alors :

$$P(X = x) = \frac{1}{Z} e^{-U(x)} \quad (2.3)$$

où U est une fonction d'énergie (exprimée en fonction) et Z une constante de normalisation.

Plusieurs fonctions d'énergie ont été proposées dans la littérature, les plus connus étant les modèles d'Ising, de Potts et le modèle gaussien.

L'estimation des paramètres d'un champ de Markov se fait en calculant les propriétés statistiques du voisinage de chaque pixel; en général, seules la moyenne et l'autocorrélation sont nécessaires, mais des modèles plus avancés exigent des calculs plus complexes.

L'avantage d'une telle approche est qu'elle permet de gérer des modèles très flexibles approchant au plus près les textures réellement observées. En revanche, elle présente plusieurs inconvénients : le choix du modèle peut s'avérer délicat, tous n'ayant pas les mêmes propriétés; et l'estimation des paramètres est parfois complexe, notamment pour les modèles les plus sophistiqués.

2.1.3.2. *Fractales*

De nombreuses textures possèdent des propriétés d'auto-similarité interne, on peut donc les qualifier de fractales. Pentland (1984) a d'ailleurs démontré que la plupart des surfaces naturelles peuvent être modélisées sous forme de fractales. Par cette approche, on peut donc utiliser un concept-clé de l'analyse fractale pour caractériser des textures : la dimension fractale (équation 2.4). La dimension fractale fournit une mesure de la rugosité ou de la complexité d'une texture. Dans le cas d'une image, cette dimension fractale est comprise entre 2 et 3, une dimension de 2 correspondant à une image parfaitement régulière. En considérant qu'un objet est constitué de N sous-objets identiques au premier, mais réduit d'un facteur r , la dimension fractale D est définie de la manière suivante :

$$D = \frac{\log N}{\log(1/r)} \quad (2.4)$$

Le calcul concret de la dimension fractale d'une texture est complexe et plusieurs méthodes existent pour y parvenir (Myint, 2003) : isarithme, prisme triangulaire, variogramme, etc.

Il a été démontré que la dimension fractale ne peut pas être toujours suffisante pour différencier certains types de texture. Une autre caractéristique est donc souvent utilisée en complément : la lacunarité. Cette grandeur mesure la granularité de la texture : une valeur élevée indique une texture très hétérogène, avec de gros « blocs » bien différenciés.

Pour que l'analyse soit pertinente, il est préférable d'avoir une image de taille assez grande et dont la texture ne varie pas trop rapidement. Néanmoins, l'analyse fractale fournit de bons résultats, y compris dans le domaine de la télédétection. Parmi les algorithmes d'analyse fractale existants, citons SFTA (*Segmentation-based Fractal Texture Analysis*) proposé par Costa *et al.* (2012) et utilisé dans le cadre de cette recherche.

2.1.4. Méthodes géométriques

Les méthodes géométriques se basent sur l'idée qu'une texture est composée d'« éléments texturaux » appelés aussi textons, texels ou primitives. La caractérisation de la texture se fait alors soit en extrayant les propriétés géométriques du texton lui-même (taille, surface, intensité, etc.), soit en analysant la répartition spatiale des textons (méthodes structurelles). C'est de ce dernier type d'approche dont il est question dans la suite de cette section.

Les méthodes structurelles sont très variées ; cependant, elles exigent toujours de commencer par extraire les éléments texturaux d'une image². Voorhees et Poggio (1988) suggèrent d'opérer par détection de blob (un blob est une région d'une image dans laquelle certaines propriétés restent relativement uniformes). Pour ce faire, ils proposent d'effectuer un filtrage de l'image à analyser par un filtre laplacien de gaussienne (LoG) multi-échelle (équation 2.5).

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2 + y^2}{2\sigma^2}\right) e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (2.5)$$

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & -16 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Figure 2.5 : filtre LoG discret de taille 5×5 ($\sigma = 1$)

Ce filtre a pour effet de calculer la dérivée seconde de l'image, horizontalement et verticalement. Les zones de gradient uniforme ressortent donc avec une faible valeur : elles peuvent ainsi être détectées par seuillage. On peut noter que ce filtre est également utilisé en détection de contours.

² Ces éléments pouvant souvent correspondre à des objets physiques de l'image, on parle souvent d'approche orientée objet pour désigner les approches structurelles. C'est notamment le cas dans cette thèse, car les points-clés détectés correspondent à des objets ou des parties d'objets (cf. section 2.2).

À l'heure actuelle, on emploie plus fréquemment des filtres basés sur des différences de gaussiennes plutôt que sur des laplaciens de gaussiennes, pour des raisons d'efficacité calculatoire. Le résultat est cependant sensiblement le même.

Une fois l'extraction des éléments texturaux effectuée, il reste à déterminer leurs caractéristiques. De nombreuses approches ont été proposées, donc certaines sont mentionnées ci-dessous.

Goyal *et al.* (1995) proposent une méthode d'évaluation du périmètre et de la compacité d'un élément textural, invariant à la rotation et aux erreurs de quantification. Ils proposent également un histogramme de propriétés des éléments texturaux invariant aux changements d'échelle.

Eichmann et Kasparis (1988) utilisent la transformée de Radon pour extraire certaines caractéristiques des éléments texturaux : le nombre d'orientations différentes au sein de la texture, les angles de ces orientations et l'écartement entre les orientations de même angle. Cette méthode ne fonctionne bien que pour les textures très régulières de type artificiel.

Enfin, plusieurs approches basées sur la morphologie mathématique ont été proposées, souvent en complément d'autres méthodes. Citons par exemple Wang *et al.* (1993) qui utilisent une méthode récursive afin de décomposer une image en éléments structurants, avant d'analyser ces éléments par des méthodes statistiques.

2.2. Éléments de vision par ordinateur

2.2.1. Généralités sur la détection et description de points-clés

En vision par ordinateur, un point-clé (ou point d'intérêt, respectivement *keypoint* et *interest point*) est un point d'une image dont la position est déterminée de façon précise et dont le voisinage possède un contenu informationnel riche (tel que des arêtes ou une texture spécifique, par exemple). Pour être exploitable, un point-clé doit de plus être

autant que possible invariant aux conditions d'illumination, de rotation et d'échelle de l'image. En effet, un point-clé doit permettre d'identifier les caractéristiques structurales du contenu de l'image, et ce, quelles que soient les conditions dans lesquelles l'image a été acquise.

La détection de points-clés est historiquement issue de la détection de coins (*corner detection*), c'est la raison pour laquelle les deux termes sont encore très liés dans la littérature scientifique. Les coins sont les points les plus simples à détecter et les plus apparents sur le plan visuel.

La détection de points-clés est un problème important, car de nombreuses applications en dépendent :

- Reconnaissance d'objets (incluant la reconnaissance faciale) : permet de catégoriser et/ou d'identifier automatiquement les objets photographiés, à la suite d'un apprentissage ;
- Assemblage d'images : recollage de plusieurs images d'une même scène pour aboutir à une image de type panorama par exemple ;
- Modélisation de scène 3D : permet de localiser les éléments d'un environnement 3D;
- Localisation robotique : localisation automatique de la position d'une caméra dans un environnement 3D ;

En matière d'analyse par points-clés, il faut distinguer deux étapes distinctes et complémentaires : la *détection* à proprement parler, qui consiste à identifier des points ou des pixels répondant à des critères spécifiques, et la *description* dont le but est d'associer un ensemble de descripteurs (souvent appelé vecteur descripteur) à chaque point-clé détecté.

De nombreux algorithmes de détection proposent leurs propres vecteurs descripteurs, mais il est souvent possible de se baser sur les points-clés détectés par un algorithme

de détection quelconque pour calculer des descripteurs à l'aide d'un algorithme complètement différent. Par exemple, il est possible d'extraire des points-clés SIFT et de les décrire par l'algorithme BRIEF, bien que SIFT propose sa propre méthode de description.

2.2.2. Détection de points-clés

De nombreuses méthodes de détection de points-clés ont été proposées, évoluant avec la puissance de calcul disponible.

Parmi les détecteurs de points-clés les plus récents ou les plus utilisés, trois grandes catégories peuvent être identifiées (Gauglitz *et al.*, 2011; Schmid *et al.*, 2000; Tuytelaars et Mikolajczyk, 2008) :

- Détecteurs de coins : déjà évoqués plus haut, il s'agit du premier type de détecteurs à avoir vu le jour. Il s'agit en général d'étudier les propriétés statistiques du voisinage de chaque pixel de l'image. Le premier détecteur de coins proposé est celui de Moravec (1980) : il consiste à calculer la similarité entre le voisinage d'un pixel et les voisinages des pixels voisins puis à retenir les pixels dont la similarité inter-voisinage est faible.

Cependant, le détecteur de coin ayant connu le plus grand succès est celui proposé par Harris et Stephens (1988) : basé sur celui de Moravec, il lui ajoute une fenêtre gaussienne et une matrice d'autocorrélation pour calculer les seconds moments du voisinage. Il est invariant aux rotations et aux variations affines de luminosité. Il a ensuite été à nouveau modifié par Mikolajczyk et Schmid (2001) pour le rendre invariant aux changements d'échelle.

À quelques exceptions près, la plupart des détecteurs de coins reposent donc sur des calculs de gradients bidirectionnels.

- Détecteurs de blobs : un « blob » est une région d'une image dans laquelle certaines propriétés restent relativement uniformes. Les détecteurs de blobs

tentent donc d'identifier dans une image des zones dont la réponse à un filtre donne un extremum local d'intensité. Les filtres utilisés sont souvent des approximations plus ou moins fines de laplaciens de gaussiennes (LoG; voir équation 2.5). Plusieurs détecteurs de ce type sont détaillés dans les sections suivantes. L'algorithme SIFT, sur lequel se base ce projet de recherche, appartient à cette catégorie.

- Détecteurs invariants par transformation affine : Mikolajczyk et Schmid (2002) ont proposé en 2002 un nouveau type de détecteur adapté aux distorsions affines de grande ampleur. Ils sont basés sur les détecteurs de coins, mais utilisent un traitement plus complexe qui accroît le temps de calcul.

Deux algorithmes de détection de points-clés sont présentés plus en détail dans les sections suivantes. Ce sont tous deux des détecteurs basés sur la recherche de blobs.

2.2.2.1. SIFT

L'algorithme SIFT (*Scale Invariant Feature Transform*) a été publié par Lowe (1999) pour la première fois en 1999. Breveté par l'université de la Colombie-Britannique, son implémentation est décrite par le même auteur en 2004.

Il repose sur les étapes suivantes :

- 1. Détection des extrema spatiaux et multi-échelle** : il s'agit d'employer un filtre de type « différence de gaussiennes » (équation 2.6) afin de rechercher tous les points d'intérêt potentiels.

Plus précisément, en notant I l'image analysée, σ un facteur d'échelle et k une constante multiplicative, on obtient (les équations suivantes sont tirées de (Lowe, 2004)) :

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad (2.6)$$

avec

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (2.7)$$

Dans l'équation 2.6, D est une image filtrée dont on va extraire des extrema locaux à la prochaine étape. Le choix d'utiliser une différence de gaussienne vient du fait que cette fonction est une bonne approximation du filtre LoG (équation 2.5) normalisé par σ^2 :

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2 LoG(x, y) \quad (2.8)$$

Plus k est proche de 1, plus cette approximation est bonne. En pratique, Lowe suggère d'utiliser la valeur $2^{1/s}$, où s est le nombre d'échelles par octave (voir figure 2.6). À chaque nouvelle octave, l'image est sous-échantillonnée d'un facteur 2.

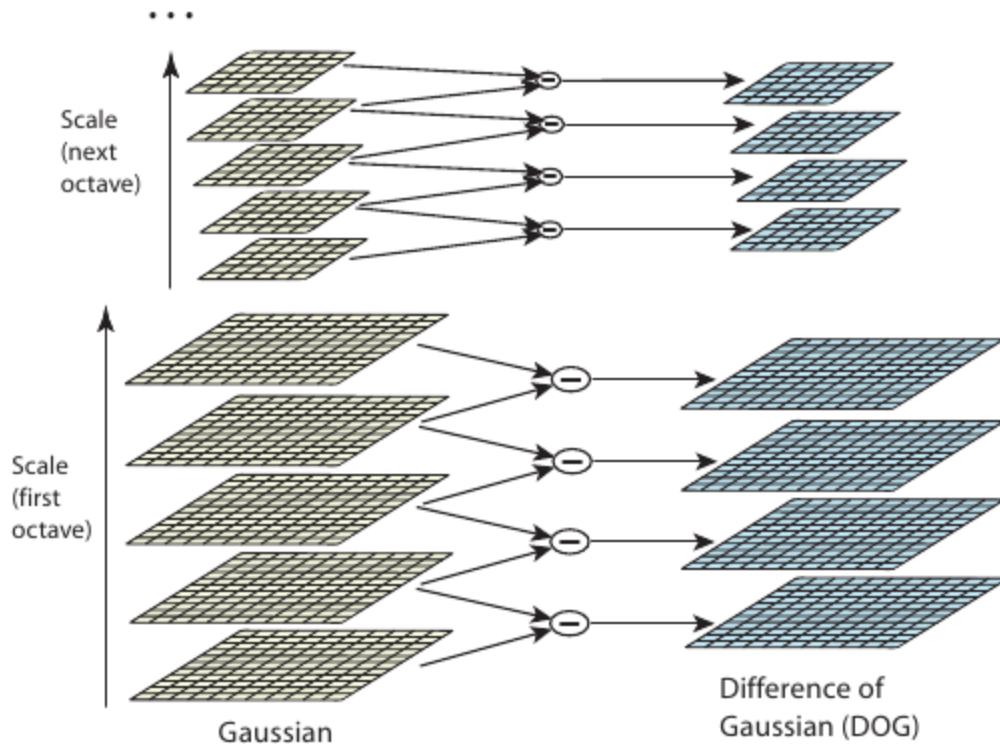


Figure 2.6 : approche multi-échelle de l'algorithme SIFT. La partie à gauche représente l'image étudiée, celle de droite représente l'image filtrée à différentes échelles (D) (Lowe, 2004)

Une première phase de localisation grossière des points d'intérêt a ensuite lieu. Cette localisation se fait au pixel près, en sélectionnant les extrema (minima et maxima) au sein d'un voisinage de 26 pixels (8 pixels à l'échelle courante et 9 pixels aux échelles supérieure et inférieure).

2. Localisation précise des points-clés : après avoir trouvé des pixels candidats au statut de point d'intérêt, il reste à en déterminer précisément l'emplacement, l'échelle et le ratio des courbures principales. Cette dernière information permet d'éliminer les points qui n'apportent pas d'information intéressante (points situés le long d'une arête par exemple).

Cette étape se fait par interpolation de $D(x,y,\sigma)$ en fonction de x , y et σ (représentés par un vecteur \mathbf{x}), à l'aide d'un développement en série de Taylor (T représente ici la transposée) :

$$D(\mathbf{x} - \mathbf{x}_0) \approx D(\mathbf{x}_0) + \frac{\partial D}{\partial \mathbf{x}}(\mathbf{x}_0)^T (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \frac{\partial^2 D}{\partial \mathbf{x}^2}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) \quad (2.9)$$

Ce qui permet d'estimer la position réelle du point-clé ($\hat{\mathbf{x}}$) :

$$\hat{\mathbf{x}} = \mathbf{x}_0 - \frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1}(\mathbf{x}_0) \frac{\partial D}{\partial \mathbf{x}}(\mathbf{x}_0) \quad (2.10)$$

La valeur de $|D(\hat{\mathbf{x}})|$ représente alors le contraste des niveaux de gris à l'emplacement du point d'intérêt. Dans l'algorithme original de Lowe, les points dont la valeur de $|D(\hat{\mathbf{x}})|$ est inférieure à 0,03 sont éliminés, mais ce seuil est réglable dans la plupart des implémentations de SIFT.

Il ne suffit pas d'éliminer les points dont le contraste avec le voisinage est trop faible. Il faut également éliminer les points qui se situent sur une simple arête (contraste important dans une seule direction) au lieu de se situer dans une région porteuse d'information telle qu'un coin (contraste dans deux directions dif-

férentes). Il s'agit donc de calculer le ratio des courbures principales de niveaux de gris, c'est-à-dire le ratio de l'intensité des gradients les plus forts dans deux directions.

Cela se fait en calculant le ratio des valeurs propres de la matrice hessienne de D , calculée à \hat{x} :

$$\mathbf{H}(\hat{x}) = \begin{bmatrix} \frac{\partial^2 D}{\partial x^2}(\hat{x}) & \frac{\partial^2 D}{\partial x \partial y}(\hat{x}) \\ \frac{\partial^2 D}{\partial x \partial y}(\hat{x}) & \frac{\partial^2 D}{\partial y^2}(\hat{x}) \end{bmatrix} \quad (2.11)$$

Le ratio r des valeurs propres vérifie alors l'équation suivante, ce qui permet d'éviter le calcul direct des valeurs propres (Tr et Det représentent respectivement la trace et le déterminant) :

$$\frac{Tr(\mathbf{H}(\hat{x}))^2}{Det(\mathbf{H}(\hat{x}))} = \frac{(r+1)^2}{r} \quad (2.12)$$

On peut ainsi fixer un seuil minimal r_0 pour r , en comparant $\frac{Tr(\mathbf{H}(\hat{x}))^2}{Det(\mathbf{H}(\hat{x}))}$ et $\frac{(r_0+1)^2}{r_0}$. Lowe conseille d'utiliser la valeur $r_0 = 10$, mais là encore ce seuil est paramétrable.

3. Choix de l'orientation :

L'orientation des points d'intérêt apporte une information supplémentaire (en plus de la position et de l'échelle) concernant les traits saillants de l'image. Elle permet d'analyser diverses transformations liées aux rotations possibles, et, dans le cadre de l'analyse de textures, de détecter la direction d'une texture.

Pour un pixel donné, le calcul de l'angle et de l'intensité de l'orientation se fait simplement à partir des gradients horizontaux et verticaux. L'angle de l'orientation est donc la direction dans laquelle le contraste est le plus fort, et son inten-

sité est proportionnelle à ce contraste.

Le calcul précis de l'orientation d'un point SIFT se fait de la manière suivante : à l'intérieur du voisinage de ce point, les orientations de plusieurs points sont calculées. Un histogramme des angles de ces orientations est alors établi, en pondérant chaque orientation par son intensité et par une fenêtre gaussienne. Le plus haut pic de l'histogramme est alors retenu comme orientation principale du point-clé; les pics dont la valeur est d'au moins 80 % celle du pic principal sont aussi conservés, de manière à créer un nouveau point d'intérêt de même position mais d'orientation différente du premier.

La figure 2.7 illustre la détection de la position des points-clés SIFT et de leur orientation, à partir d'une image test.

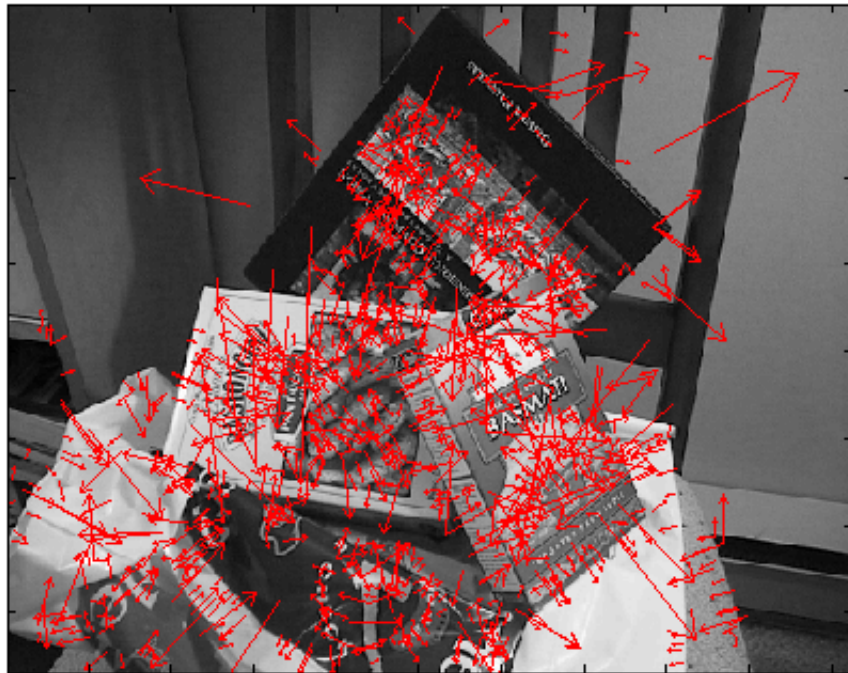


Figure 2.7 : exemple de points d'intérêt SIFT détectés et de leur orientation

4. Calcul des descripteurs : voir section 2.2.3.1.

2.2.2.2. SURF

L'algorithme SURF (*Speeded Up Robust Features*) est proposé par Bay *et al.* (2006) puis révisé en 2008. Il s'inspire de SIFT mais présente des différences significatives. Son but premier est de diminuer le temps de calcul nécessaire à la détection et la description des points-clés.

Alors que SIFT utilisait un filtre de type *DoG* (*Difference of Gaussian*) – approximation d'un filtre laplacien multi-échelle – pour détecter des points-clés, SURF utilise une approximation rapide de la matrice hessienne (voir équation 2.11) de l'image originale.

Pour accélérer la détection, l'idée de prétraiter l'image en la convoluant par un filtre gaussien est abandonnée. À la place, les auteurs proposent de filtrer directement l'image par des filtres simplifiés (figure 2.8). Les filtres sont précalculés pour chaque écart-type désiré; dans l'exemple de la figure 2.8, l'écart-type vaut 1,2. Le temps de calcul est alors considérablement réduit du fait de la taille limitée des filtres et des valeurs discrètes qu'ils contiennent.

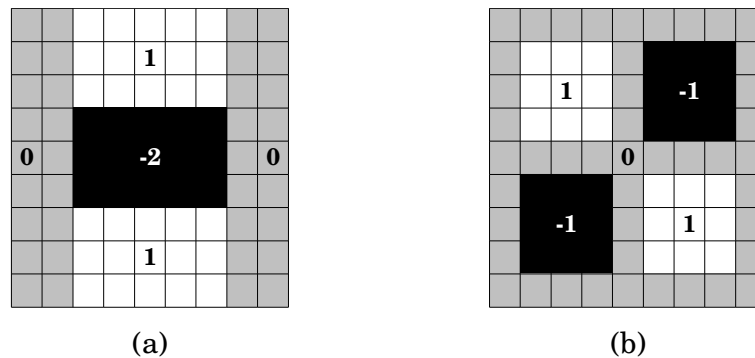


Figure 2.8 : exemples d'approximations de filtres gaussiens (dérivée seconde) utilisés par l'algorithme SURF. (a) Direction verticale. (b) Direction diagonale principale. Tiré de (Bay *et al.*, 2006)

La matrice hessienne est simplement calculée par convolution entre ces filtres simplifiés (au nombre de quatre par échelle) et l'image originale. Le déterminant de cette

matrice, pour chaque pixel et chaque échelle, permet de détecter les points-clés : comme pour SIFT, une recherche des extrema locaux du déterminant au sein d'un voisinage de 26 pixels (8 pixels à l'échelle courante et 9 pixels aux échelles supérieure et inférieure) est effectuée. La position exacte des points-clés est ensuite obtenue par interpolation.

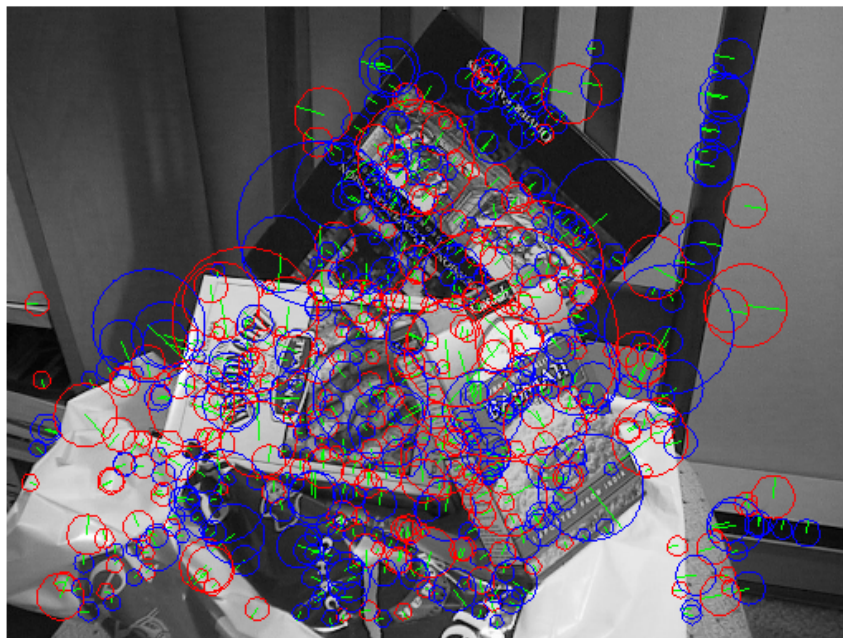


Figure 2.9 : exemple de points d'intérêt SURF détectés et de leur orientation

Enfin, l'orientation des points-clés est obtenue de manière assez similaire à SIFT, excepté qu'elle est basée sur le filtrage du voisinage de chaque point par des ondelettes de Haar au lieu de calculer les gradients locaux de l'image.

2.2.3. Descripteurs de points-clés

2.2.3.1. SIFT

L'algorithme SIFT attribue à chaque point-clé cinq caractéristiques de base : localisation (x et y), échelle, orientation (angle et intensité). Il offre également une méthode de calcul d'un vecteur descripteur aussi invariant que possible aux conditions d'acqui-

tion de l'image, et calquée sur un modèle de vision biologique. Ainsi, le descripteur se base avant tout sur l'orientation des gradients des niveaux de gris dans le voisinage du point-clé.

Plus précisément, on considère autour du point-clé une fenêtre de 16 pixels de côté. À l'intérieur de cette fenêtre, l'orientation de chaque pixel (angle et intensité) est calculée. À l'aide de ces données, on construit ainsi un histogramme des orientations dans 4×4 blocs de 4×4 pixels, où l'orientation de chaque pixel est pondérée par un facteur gaussien (voir figure 2.10). Dans chaque histogramme, les orientations sont réparties en huit valeurs possibles. Le descripteur final comporte donc $4 \times 4 \times 8 = 128$ valeurs.

Divers paramètres peuvent être changés de manière à modifier la taille finale du descripteur : le nombre de valeurs de l'histogramme (8 par défaut) et le nombre de « sous- » histogrammes utilisés (16 par défaut).

Pour finir, le vecteur descripteur est normalisé et les valeurs sont seuillées de manière à être toujours inférieures ou égales à 0,2 (afin de diminuer la sensibilité aux variations de luminosité).

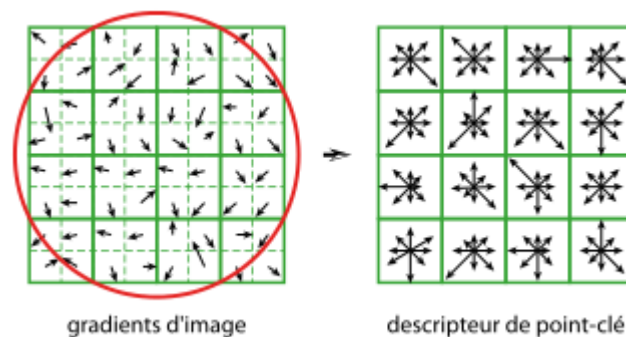


Figure 2.10 : illustration de la méthode de calcul des descripteurs SIFT (Lowe, 2004)

2.2.3.2. *SURF*

Comme pour SIFT, les descripteurs des points-clés sont calculés à partir des orientations de gradient dans des sous-régions des voisinages des points. Les orientations sont cependant calculées à partir de la réponse du voisinage à une série d'ondelettes de

Haar, ce qui ne permet pas une description aussi précise de l'angle d'orientation. Pour cette raison, la taille du descripteur de SURF est plus faible que celle de SIFT : 64 au lieu de 128, avec les paramètres par défaut. Toutefois, une version « étendue » du descripteur SURF à 128 éléments existe; elle offre les meilleures performances de description, au détriment du coût calculatoire.

2.2.4. Points-clés et classification d'images

Plusieurs méthodes de classification d'image basée sur la détection de points-clés ou de zones d'intérêt ont déjà été proposées. Ces approches reposent généralement sur l'analyse des descripteurs des points-clés afin de construire une « signature » de l'image, par le biais d'une méthode de partitionnement (Lazebnik *et al.*, 2005). En particulier, sont apparues récemment les méthodes très populaires de type « *bag-of-keypoints* », adaptées de l'approche « *bag-of-words* », dans lesquelles on vise à construire un dictionnaire de points-clés permettant d'identifier différents types d'images (Csurka *et al.*, 2004; Zhang *et al.*, 2007).

Signalons par ailleurs l'existence de méthodes basée sur la production de SIFT dense (Dalal et Triggs, 2005), dans lesquelles on calcule des descripteurs SIFT sur chaque point d'une grille préétablie (et donc sans passer par la phase de détection SIFT). Ces méthodes s'avèrent particulièrement efficaces en matière de classification d'images (Bosch *et al.*, 2008; Liu *et al.*, 2008).

Les approches précédentes ont toutefois pour inconvénient de ne pas exploiter les relations spatiales entre points-clés, dont l'analyse constitue l'originalité de KPC. Le prochain paragraphe détaille la méthode de calcul des statistiques spatiales de ces points-clés.

2.2.5. Statistiques spatiales ponctuelles

La statistique spatiale consiste en l'étude des phénomènes ayant lieu au sein d'un en-

semble spatial, qu'il soit 2D ou 3D. Dans le cadre de cette thèse, nous nous intéressons uniquement aux statistiques spatiales ponctuelles, c'est-à-dire l'étude de la répartition de points à l'intérieur d'un espace limité (processus ponctuel).

Dans le reste de cette section, on note I l'image concernée, $n(I)$ le nombre de points inclus dans cette image, S la surface de l'image.

2.2.5.1. *Statistiques du premier ordre*

La seule statistique du premier ordre applicable à un processus ponctuel (PP) est son intensité. L'intensité d'un PP, notée λ , représente le nombre moyen de points localement (Gaetan et Guyon, 2008). Si le PP est stationnaire, c'est-à-dire si la répartition des points est uniforme, alors λ peut être estimé par :

$$\hat{\lambda} = \frac{n(I)}{S}. \quad (2.13)$$

Dans le cas d'une répartition des points non uniforme, le même calcul peut être effectué sur des sous-parties de l'image de base.

Cette statistique, bien qu'utile, est limitée car elle n'analyse pas assez finement la répartition des points au sein d'une image. Ainsi, elle ne permet pas de faire la différence entre une répartition de points purement aléatoire et une répartition régulière.

2.2.5.2. *Statistiques du second ordre : fonction K de Ripley*

L'étude statistique au second ordre d'une répartition de points consiste à estimer la distribution de la distance inter-points. Il s'agit d'un bon indicateur de la répartition des points : des points uniformément distribués ont des distances inter-points très peu variées, alors que des points répartis de manière plus aléatoire ont des distances inter-points prenant de nombreuses valeurs différentes.

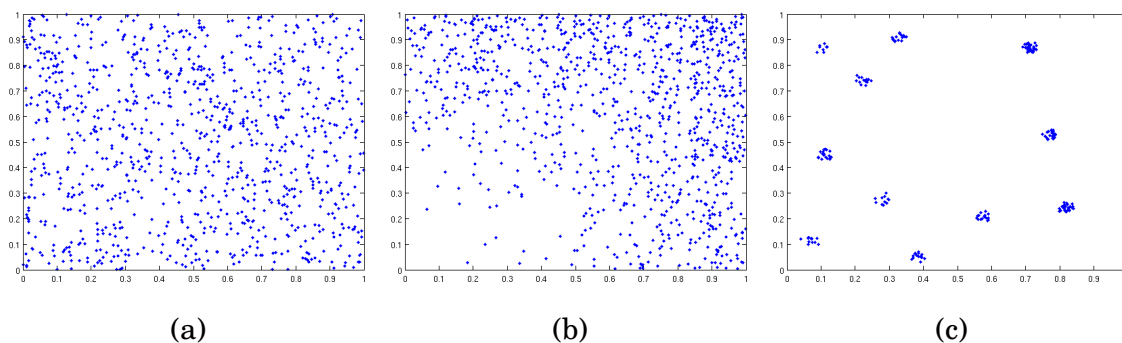


Figure 2.11 : exemples de processus ponctuels. (a) Processus de Poisson homogène. (b) Processus de Poisson suivant un gradient circulaire. (c) Processus de Strauss.

La fonction K de Ripley (1976), introduite par ce dernier en 1976 pour évaluer la répartition des arbres au sein d'une forêt, offre un bon moyen de mesurer la distribution de la distance inter-points. Elle est définie par l'équation suivante, modifiée par Irons et Petersen (1981) pour des raisons pratiques :

$$\hat{K}(r) = \frac{S}{N^2} \sum_{x \neq y} w_{x,y}^{-1} \mathbb{1}(d(x,y) < r) \quad (2.14)$$

Ici, r représente la distance inter-points dont on veut estimer la fréquence, N le nombre total de points et S la surface de l'image. x et y sont des points, $w_{x,y}$ une fonction de poids pour gérer les effets de bord, et $\mathbb{1}(d(x,y) < r)$ une fonction valant 1 si la distance entre x et y est inférieure à r .

La figure 2.12 fournit une illustration de la manière dont la fonction K permet de caractériser une distribution de points.

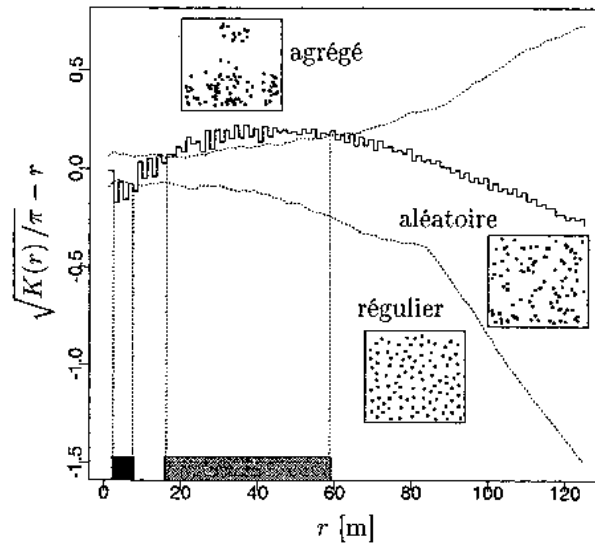


Figure 2.12 : illustration du comportement de la fonction K de Ripley en fonction de la distribution des points (tiré de Ripley, 1976)

La fonction K peut être utilisée pour représenter des grandeurs concrètes. En réutilisant l'intensité λ définie dans l'équation 2.13, les deux grandeurs suivantes peuvent être utilisées :

- $\lambda K(r)$ représente le nombre moyen de points dans un disque de rayon r ;
- $\lambda^2 K(r)$ représente le nombre moyen de paires de points (distincts) dont la distance est inférieure à r , par unité de surface.

2.3. Fusion de données

2.3.1. Contexte et définitions

La fusion de données, ou fusion d'informations, « constitue un cadre formel dans lequel s'expriment les moyens et techniques permettant l'alliance des données de sources diverses. Elle vise à l'obtention d'information de plus grande qualité; la définition exacte de "plus grande qualité" dépendra de l'application » (Wald, 1999). Cette définition générale, mise au point par le groupe de travail européen *Association of Remote Sensing*

Laboratories, vise à synthétiser les nombreuses définitions énoncées au fil des années : en effet, bien que le concept de fusion d'information soit aisé à comprendre, il reste difficile d'en énoncer une définition générique.

Bloch et Maître (1994) fournissent une définition plus « opérationnelle » sur le plan technique, spécifique au traitement d'images :

« La fusion d'images est l'utilisation conjointe d'images hétérogènes pour l'aide à la prise de décision. »

La notion de prise de décision est en effet capitale dans le processus de fusion. C'est précisément dans le but de prendre une décision qu'a lieu toute fusion de données; la fusion elle-même est décrite par le terme générique « utilisation conjointe ».

Pour la même raison qui rend complexe l'établissement d'une définition de la fusion de données, il est difficile de modéliser le processus de fusion à l'aide d'étapes simples ou d'un diagramme général. Un modèle fréquemment utilisé est celui du JDL (*Joint Directors of Laboratories*), conçu dans le but d'établir un cadre de travail commun entre théoriciens, ingénieurs et autres intervenants des systèmes de fusion (Hall et Llinas, 1997).

Ce modèle très général n'est toutefois pas assez concret pour être utilisable, d'autant plus que la fusion d'images exploite des outils spécifiques.

On peut donc imaginer le modèle suivant qui, tout en restant générique, permet de représenter les principales étapes indispensables à tout système de fusion d'images :

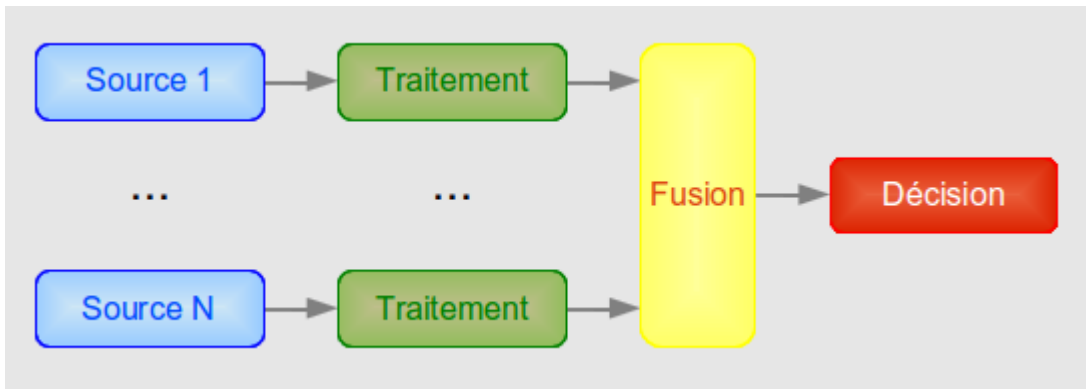


Figure 2.13 : schéma général d'un système de fusion d'images

2.3.1.1. *Imperfections de l'information*

La raison pour laquelle la fusion d'informations est fréquemment employée est simple : il s'agit de pallier les multiples imperfections auxquelles l'information peut être sujette. En combinant plusieurs sources d'information acquises dans des conditions différentes, on espère pouvoir réduire ces imperfections grâce à des règles de fusion bien choisies.

Deux types d'imperfections sont unanimement reconnues : il s'agit de l'imprécision et de l'incertitude. Bloch (2003) en ajoute d'autres : l'incomplétude, l'ambiguïté et le conflit.

- **Imprécision et incertitude** : ces deux notions sont étroitement liées et sont souvent confondues, bien qu'étant fondamentalement différentes. (Bloch et Maître, 1994) résumant ainsi ces termes : « *l'imprécision [...] porte sur un défaut quantitatif de connaissance, [...] tandis que l'incertitude est relative à la vérité d'une information* ». Une région d'une image peut ainsi être imprécise (mélange de deux classes difficiles à distinguer) sans être incertaine (absence de bruit sur l'image); inversement, l'incertitude peut être très élevée (présence de brume) mais très précise (nombre réduit de classes aisément différenciables en l'absence de bruit). Bien entendu, ces deux imperfections peuvent se combiner, comme

c'est souvent le cas.

Dans la mesure où ces deux aspects sont souvent rencontrés en traitement d'image, il est important pour un algorithme de fusion de pouvoir les traiter explicitement.

- **Incomplétude** : l'incomplétude est liée à l'absence totale ou partielle d'information. Chaque source offre généralement une information incomplète en ne fournissant que certaines caractéristiques du phénomène observé. Ainsi, un capteur optique embarqué sur un satellite ne donne accès qu'à des bandes spectrales limitées.
- **Ambiguïté** : l'ambiguïté désigne la possibilité qu'offre une source de mener à deux interprétations différentes. Elle peut être issue d'autres imperfections, telles que l'incomplétude ou l'imprécision, ou bien être intrinsèque à la source d'information. C'est le cas de deux classes qui auraient la même réponse spectrale, par exemple.
- **Conflit** : on dit qu'il y a conflit lorsque deux sources d'information observant le même phénomène mènent à deux interprétations contradictoires. L'origine du conflit est souvent complexe à déterminer : il peut aussi bien découler d'une source imprécise ou incertaine que d'une méthode de combinaison des sources non adaptée. De plus, un conflit élevé peut avoir plusieurs significations suivant le contexte de l'étude; il peut parfois constituer une information utile sur le système étudié. La présence de conflit doit donc toujours être analysée avec soin.

Bloch ajoute d'autres caractéristiques des sources, celles-ci positives : la redondance et la complémentarité.

- **Redondance** : on parle de redondance lorsque la même information est fournie par plusieurs sources. Lorsque les sources sont imprécises, la redondance est ex-

exploitée afin d'augmenter la qualité de la fusion.

- **Complémentarité** : la complémentarité est la propriété des sources qui donnent des informations sur des caractéristiques différentes du phénomène étudié. En télédétection, les images optiques et radar constituent un exemple évident de complémentarité. Comme la redondance, elle est le plus possible exploitée par les méthodes de combinaison pour accroître la qualité du résultat.

2.3.1.2. *Niveaux de fusion*

On considère classiquement qu'il existe trois niveaux de fusion possibles en matière de fusion d'images (Pohl et Van Genderen, 1998) : niveau pixel, niveau objet et niveau décision.

- **Niveau pixel** : il s'agit du plus bas niveau de fusion possible, qui est également le plus utilisé. Travailler au niveau pixel revient à fusionner des caractéristiques physiques directement mesurables. Le recalage géométrique et éventuellement le ré-échantillonnage des images fusionnées joue alors un rôle essentiel.
- **Niveau objet** : la fusion au niveau objet exige de préalablement segmenter les images étudiées. Les objets extraits peuvent alors être fusionnés à partir de leurs caractéristiques (superficie, taille, intensité, etc.). La difficulté consiste alors à segmenter convenablement les images, de manière à extraire des objets qui ont un sens sur le plan physique.
- **Niveau décision** : la fusion au niveau décision consiste à calculer séparément les décisions au niveau de chaque image, puis fusionner l'ensemble des décisions à partir de la fiabilité de chaque source.

La figure 2.14 ci-dessous résume l'approche choisie pour chacun des trois niveaux. Elle permet de constater que plus le niveau de fusion est bas, plus la quantité d'informations à fusionner est élevée, mais moins le prétraitement des sources est complexe.

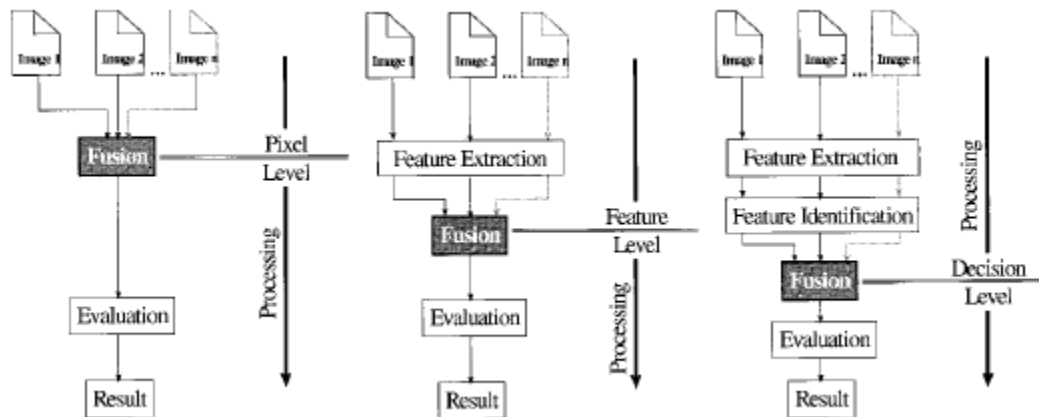


Figure 2.14 : représentation des niveaux de fusion (Pohl et Van Genderen, 1998)

Dans l'ensemble de cette étude, on s'intéressera seulement à la fusion au niveau pixel en raison de sa plus grande simplicité. C'est donc l'approche qui est choisie dans les sections suivantes.

L'image vérité sera notée I , dont les pixels ont pour valeur $I_{x,y}$; l'image observée est notée X , et les sources multiples X^i , $1 \leq i \leq S$, S étant le nombre de sources. L'ensemble des C classes recherchées dans X est $\Theta = \{\theta_1, \dots, \theta_C\}$.

2.3.2. Modèles probabilistes

Les premiers modèles de fusion d'images, et qui restent aujourd'hui encore les plus utilisés, sont les modèles probabilistes. Ces modèles reposent sur une approche bayésienne du problème, c'est-à-dire que la probabilité qu'un pixel appartienne à une classe est exprimée sous forme de probabilité conditionnelle, grâce à la règle de Bayes. Les imperfections de l'information sont donc modélisées sous la forme de densités de probabilité, ce qui constitue le principal défaut de cette approche.

En effet, il n'y a pas de moyen « naturel » dans cette approche de modéliser l'absence d'information. De plus, l'incertitude et l'imprécision sont intrinsèquement confondues. Considérons par exemple le cas de figure suivant : si l'on obtient pour un pixel une probabilité de 50% pour la classe « eau » et de 50% pour la classe « végétation », cela ne

permet pas de décider si la zone représentée par le pixel est en partie couverte d'eau et en partie de végétation (imprécision), ou si cette zone est entièrement couverte de l'une des deux classes sans que l'on puisse décider laquelle (incertitude), ou bien encore s'il s'agit d'une situation intermédiaire. Devant une telle ambiguïté, il est donc difficile de tirer des conclusions sur le résultat de la fusion obtenu.

2.3.3. Modèles possibilistes

On désigne par le terme « possibiliste » les approches basées sur la théorie des ensembles flous, développée par Zadeh (1965). Sans entrer dans les détails de cette théorie, on peut la résumer de la manière suivante : par opposition à la théorie ensembliste « classique », un élément x d'un univers de référence U a la possibilité d'appartenir en partie à un sous-ensemble quelconque X . La mesure d'appartenance à X est évaluée par une fonction indicatrice μ_X dont la valeur est comprise entre 0 et 1.

Ainsi, $\mu_X(x) = 1$ signifie que x appartient complètement à X , et $\mu_X(x) = 0$ signifie que x n'appartient pas du tout à X . Les valeurs intermédiaires de μ_X traduisent une appartenance partielle à l'ensemble X . On appelle alors « ensemble flou » la donnée de X et μ_X . Il est immédiatement visible que le concept d'ensemble flou est particulièrement adapté à la représentation de l'imprécision : en effet, il s'agit d'un outil idéal pour modéliser des idées ou objets sans définition quantitative précise.

À partir des fonctions indicatrices μ , il est possible de définir des opérateurs de comparaison, d'union et d'intersection entre ensembles flous, de manière similaire aux ensembles classiques. On peut également définir des opérateurs de combinaison, utilisés pour fusionner des ensembles flous issus de sources différentes.

Ces notions de théorie des ensembles flous permettent d'aborder la théorie des possibilités. Cette dernière a été introduite par Zadeh en 1978 et développée par Dubois et Prade dans les années 1980. Elle fait appel à deux mesures fondamentales qui permettent d'évaluer le niveau de certitude d'une information : la *possibilité* et la *nécessi-*

té, notés respectivement Π et Λ .

Π est défini à partir des contraintes suivantes :

$$\begin{cases} \Pi(\emptyset) = 0 \\ \Pi(U) = 1 \\ \forall X_i \subseteq U, \Pi(\cup X_i) = \max \Pi(X_i) \end{cases} \quad (2.15)$$

Il suffit donc de définir Π pour les singletons de U en se basant sur des ensembles flous (étape de modélisation floue, qui peut s'avérer délicate) pour munir l'ensemble des sous-ensembles de U d'une mesure de possibilité.

La nécessité Λ est ainsi définie :

$$\Lambda(X) = 1 - \Pi(\bar{X}) \quad (2.16)$$

On peut vérifier à partir de ces formules que $\Pi(X)$ est toujours supérieur ou égal à $\Lambda(X)$. Pour mettre en relation les vocabulaires probabiliste et possibiliste, on peut donc dire que la possibilité (resp. nécessité) d'un ensemble d'hypothèses est un indicateur de la probabilité maximale (resp. minimale) que cet ensemble soit vrai. L'écart entre la possibilité et la nécessité est lié à l'incertitude relative à l'observation : plus cet écart est grand, plus l'incertitude est élevée.

La possibilité et la nécessité ont pour objectif final de simplifier l'étape de décision en quantifiant l'incertitude liée aux observations. En effet, cette décision peut être simplement effectuée à partir de l'hypothèse offrant le maximum de possibilité, ou le maximum de nécessité, avec ou sans seuil d'incertitude.

2.3.4. Modèles évidentiels : théorie de Dempster-Shafer

La théorie de Dempster-Shafer (DST) (Dempster, 1968; Shafer, 1976) peut être vue comme une généralisation du modèle probabiliste permettant une représentation bien plus fine de l'information et de ses imperfections. En effet, la DST manipule des ensembles d'hypothèses (correspondant aux classes, dans un contexte de classification)

au lieu de simples singletons; ainsi, en cas d'incertitude totale entre deux hypothèses, il suffit d'accorder un poids à l'ensemble composé des deux hypothèses et non aux hypothèses simples.

Dans la suite de ce chapitre, on notera $\Theta = \{\theta_1, \dots, \theta_K\}$ l'ensemble des hypothèses (ou classes) en faveur desquelles une décision peut être prise. La confiance en la véracité d'un sous-ensemble d'hypothèses est modélisée par une fonction de masse m dont les valeurs sont comprises entre 0 et 1.

Le traitement évidentiel consiste en trois grandes étapes (figure 2.15) qui sont détaillées dans les sections suivantes :

- 1. Initialisation des fonctions de masse :** pour chaque source, il s'agit d'attribuer une masse à chaque hypothèse ou ensemble d'hypothèses. C'est l'étape la plus délicate, car c'est à ce moment que l'on modélise les connaissances que nous apportent les sources. Cette étape peut difficilement être complètement automatisée dans la mesure où les spécificités de chaque source d'information doivent être prises en compte.
- 2. Combinaison :** cette étape consiste à rassembler les fonctions de masse de chaque source pour aboutir à un unique jeu de fonctions de masse. Il s'agit de l'opération de fusion proprement dite. La combinaison la plus souvent retenue est de type conjonctive.
- 3. Décision :** en se basant sur les fonctions de masse agrégées, il faut effectuer une décision sur l'hypothèse ou le sous-ensemble d'hypothèses retenu. C'est là qu'entrent en jeu des fonctions particulières, la *plausibilité* et la *crédibilité*. La première mesure la confiance maximale que l'on peut avoir sur un sous-ensemble d'hypothèses, la seconde en mesure la confiance minimale. La différence entre les deux mesure l'ignorance que l'on a sur ce sous-ensemble. La décision se fait fréquemment à partir du maximum de plausibilité ou de crédibilité, mais

d'autres critères existent.

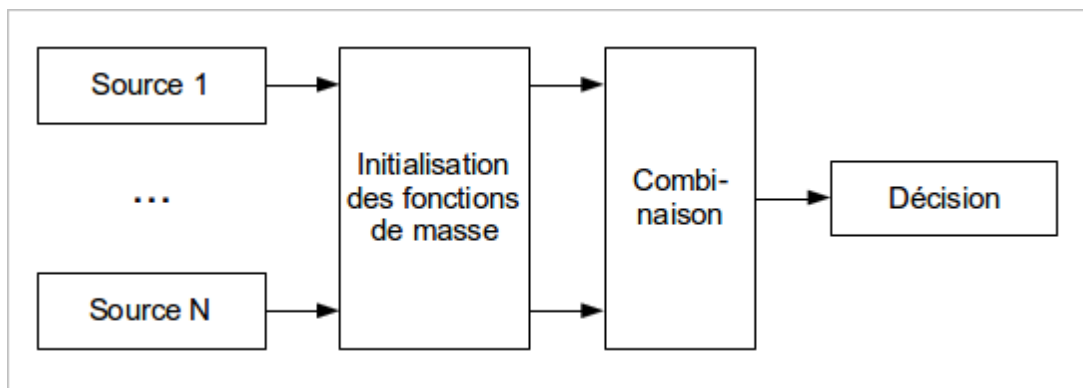


Figure 2.15 : étapes du traitement évidentiel, précédées de l'identification des sources

2.3.4.1. *Initialisation*

Comme il a été mentionné dans la section précédente, la DST se base sur l'utilisation de fonctions de masses. Une fonction de masse m est une fonction de 2^Θ (l'ensemble des sous-ensembles de Θ) dans $[0, 1]$. Elle attribue donc un poids à chaque sous-ensemble d'hypothèses, y compris les singletons qui représentent les hypothèses simples. m vérifie les critères suivants :

$$\begin{cases} m(\emptyset) = 0 \\ \forall A \subseteq \Theta, m(A) \geq 0 \\ \sum_{A \subseteq \Theta} m(A) = 1 \end{cases} \quad (2.17)$$

Le premier critère correspond à une hypothèse de monde clos, c'est-à-dire que toutes les hypothèses possibles sont répertoriées dans Θ . Il s'agit d'une hypothèse forte qui doit être prise en compte avec attention. En effet, cela signifie que s'il existe dans l'image une classe non répertoriée dans la liste des hypothèses prévues et suffisamment distincte des autres classes, les pixels de la classe supplémentaire seront mal attribués dans chacune des sources et la combinaison n'aura plus beaucoup de sens.

Les deuxième et troisième critères ont simplement un but de normalisation des

masses.

Le choix des valeurs à attribuer à la fonction de masse revêt une importance considérable. Il est bien sûr possible de se contenter d'attribuer des valeurs uniquement aux hypothèses simples ($m(\{\theta_1\})$, $m(\{\theta_2\})$, ...), comme cela se fait dans le cadre probabiliste. On parle alors de fonctions de masses bayésiennes. Cela ne présente toutefois pas un grand intérêt dans la mesure où cette approche n'apporte pas d'informations supplémentaire sur l'incertitude des informations. L'enjeu est donc de pouvoir attribuer une masse à des hypothèses composées; pour cela, plusieurs méthodes existent.

Méthode d'affaiblissement

La méthode d'affaiblissement (Shafer, 1976) se base sur l'approche probabiliste : on suppose donc que les masses des hypothèses simples ont été préalablement calculées par la méthode bayésienne. Il s'agit ensuite de réduire les masses des hypothèses simples et d'attribuer la masse restante à Θ , toutes les autres sous-ensembles étant omis :

$$\begin{cases} m'(\theta_i) = \alpha m(\theta_i) \\ m'(\Theta) = 1 - \alpha \end{cases} \quad (2.18)$$

α est ici un coefficient d'affaiblissement (qui sera appelé par la suite « coefficient de fiabilité ») compris entre 0 et 1 : plus α est faible, plus les masses des hypothèses simples sont réduites en faveur de l'univers des hypothèses. Cela permet de rendre compte de la fiabilité d'une source, car une masse importante sur Θ signifie un niveau d'ignorance élevé quant à l'ensemble des hypothèses.

Cette méthode, bien que simple à mettre en œuvre, est bien souvent trop limitée car elle n'attribue aucun poids aux hypothèses composées (en-dehors de Θ). En revanche, elle peut être utilisée en conjonction avec d'autres méthodes si une source est supposée peu fiable et que l'on souhaite affaiblir tous les sous-ensembles d'hypothèses.

Autres modèles de base probabiliste

Appriou (1993) a proposé d'autres modèles d'initialisation de masses basés sur la connaissance explicite des probabilités conditionnelles $p(x_{x,y}|\theta_k)$ (équations 2.19 et 2.20).

Dans le premier modèle (équation 2.19), la masse des hypothèses simples devient alors une fonction croissante de leurs probabilités conditionnelles respectives. L'avantage de ce modèle est qu'il permet également d'estimer les fonctions de masses des ensembles $\Theta \setminus \theta_i$, ce qui revient à diminuer la croyance en θ_i .

$$\begin{cases} m'(\theta_i)(x_{x,y}) = \frac{\alpha_i R p(x_{x,y}|\theta_i)}{1 + R p(x_{x,y}|\theta_i)} \\ m'(\Theta \setminus \theta_i)(x_{x,y}) = \frac{\alpha_i}{1 + R p(x_{x,y}|\theta_i)} \\ m'(\Theta) = 1 - \alpha_i \end{cases} \quad (2.19)$$

Comme dans le modèle précédent, α_i est un coefficient d'affaiblissement que l'on peut définir classe par classe, ce qui permet d'affiner la fiabilité des sources. R est un coefficient de pondération des probabilités conditionnelles.

Le second modèle (équation 2.20) attribue une masse nulle à toutes les hypothèses simples. Il est à utiliser dans le cas où la probabilité conditionnelle $p(x_{x,y}|\theta_k)$ apporte plus d'information sur ce qui n'appartient pas à la classe θ_k .

$$\begin{cases} m'(\theta_i)(x_{x,y}) = 0 \\ m'(\Theta \setminus \theta_i)(x_{x,y}) = \alpha_i (1 - R p(x_{x,y}|\theta_i)) \\ m'(\Theta) = 1 - \alpha_i (R p(x_{x,y}|\theta_i) - 1) \end{cases} \quad (2.20)$$

Il souffre du même inconvénient que le précédent, à savoir la connaissance explicite des probabilités conditionnelles.

2.3.4.2. *Combinaison*

Lors de l'étape de combinaison, les jeux de masse de chaque source sont rassemblés pour aboutir à une unique fonction de masse globale. Il existe différentes règles de combinaison, la plus utilisée étant la combinaison conjonctive, aussi appelée règle de Dempster ou règle orthogonale (Shafer, 1976). Dans tous les cas, la gestion du conflit (masse de l'absence d'hypothèse) joue un rôle important dans la qualité du résultat final.

Combinaison conjonctive de Dempster

Soient m_1 et m_2 deux fonctions de masses (définies sur le même univers d'hypothèses Θ), et A un ensemble d'hypothèses de Θ . L'opérateur de combinaison conjonctive est noté \oplus et il est défini par l'équation suivante :

$$\forall A \subseteq \Theta, (m_1 \oplus m_2)(A) = \sum_{B_1 \cap B_2 = A} m_1(B_1)m_2(B_2) \quad (2.21)$$

Cet opérateur est associatif et commutatif : il est donc possible de fusionner plus de deux sources d'information sans que l'ordre des sources ait d'importance.

Par la suite, on notera m_F la fonction de masse résultant de la combinaison des fonctions de masse sources : $m_F = m_1 \oplus m_2$. La règle de Dempster conduit souvent à attribuer une masse à l'ensemble vide, c'est-à-dire $m_F(\emptyset) \neq 0$. Cette masse particulière est appelée conflit; elle provient d'un désaccord entre les sources d'information. Ce désaccord peut venir de l'absence de fiabilité de certaines des sources, ou du fait que les sources fournissent des informations de nature différente (Bloch et Maître, 1994). Dans le premier cas, on résout généralement le désaccord en redistribuant la masse du conflit sur les hypothèses simples ou composées, tel que détaillé dans le prochain paragraphe. Dans le second cas, la fusion n'a simplement pas de sens.

Lorsque les sources d'information accordent environ les mêmes masses aux différentes

hypothèses, la règle de Dempster conduit au renforcement des hypothèses dont la masse est élevée et à l'affaiblissement des autres. L'expression « combinaison conjonctive » vient du fait que le produit des masses de deux sous-ensembles B_1 et B_2 de Θ est attribué à $B_1 \cap B_2$ (par exemple, le produit des masses de $\{\theta_1, \theta_2\}$ et $\{\theta_1, \theta_3\}$ est attribué à $\{\theta_1\}$). Les masses ont donc tendance à se « concentrer » sur les hypothèses simples et sur l'ensemble vide : pour cette raison, ce type de combinaison fait souvent apparaître un conflit entre sources. L'aspect conjonctif de cette règle a pour autre conséquence de réduire l'intervalle entre crédibilité et plausibilité, ce qui traduit une confiance plus élevée dans les hypothèses de masse élevée.

Combinaison disjonctive

La combinaison disjonctive (Dubois et Prade, 1986, 1992) est une autre méthode de combinaison dont l'expression est donnée par l'équation 2.22. À l'inverse de la combinaison conjonctive, cette méthode attribue le produit des masses de deux sous-ensembles B_1 et B_2 à $B_1 \cup B_2$, ce qui a pour effet de répartir les masses vers les hypothèses composées et Θ . Aucun conflit ne peut donc apparaître, excepté si l'ensemble vide possède déjà un poids parmi plusieurs sources.

$$\forall A \subseteq \Theta, (m_1 \oplus_{\cup} m_2)(A) = \sum_{B_1 \cup B_2 = A} m_1(B_1)m_2(B_2) \quad (2.22)$$

Ce type de combinaison est moins focalisé dans la mesure où, en cas d'ambiguïté entre B_1 et B_2 , la masse résultante est confiée à $B_1 \cup B_2$, ce qui revient à dire que la vérité se situe dans l'une de ces deux hypothèses. Par opposition, la combinaison conjonctive aurait attribué la masse résultante à \emptyset , ce qui signifie que la vérité n'est ni dans B_1 , ni dans B_2 . La combinaison disjonctive constitue donc une manière de ne pas trancher entre deux hypothèses contradictoire (pour cette raison, elle est parfois appelée « règle de compromis »). Comme ce comportement n'est en général pas souhaité car on cherche

à faire en sorte que le résultat de la fusion soit aussi fiable que possible, cette méthode est plus rarement utilisée.

Combinaison robuste

Mise au point par Florea *et al.* (2009), la combinaison robuste consiste à calculer une moyenne entre les résultats des combinaisons conjonctive et disjonctive, pondérée en fonction du conflit. Partant du constat que la combinaison conjonctive a naturellement tendance à produire des résultats plus précis (car les masses sont concentrées sur les hypothèses simples), et qu'au contraire la combinaison disjonctive fournit des résultats plus vagues privilégiant les ensembles d'hypothèses, l'idée est d'accorder un poids d'autant important à la première que le conflit est faible. Ainsi, lorsque le conflit est nul, la combinaison robuste est équivalente à la combinaison conjonctive. À l'inverse, lorsque le conflit vaut 1, elle revient à effectuer une combinaison disjonctive. Cette approche permet donc de trancher automatiquement entre deux types de combinaisons couramment utilisées.

Autres combinaisons

Plusieurs autres méthodes de combinaison existent, souvent dérivées de l'approche conjonctive. Sentz et Ferson (2002) en ont établi une liste assez complète; citons notamment la méthode de moyennage qui consiste à calculer une moyenne pondérée des masses de chaque sous-ensemble d'hypothèses pour obtenir le résultat de la fusion.

Certaines méthodes de fusion ne se basent pas sur les fonctions de masses, mais plutôt sur les fonctions de croyance ou de plausibilité présentées dans la section 2.3.4.3 (par exemple : Shafer, 1976). Les deux approches sont similaires et reviennent généralement au même, la différence se situant surtout dans la manière d'aborder le problème.

Conflit

Le conflit, souvent noté k , est la masse de l'ensemble vide :

$$k = m_F(\emptyset) = \sum_{B_1 \cap B_2 = \emptyset} m_1(B_1)m_2(B_2) \quad (2.23)$$

La manière la plus simple de supprimer le conflit est d'effectuer l'opération suivante, nommée « normalisation » :

$$\begin{cases} m'(\emptyset) = 0 \\ \forall A \subseteq \Theta, m'(A) = \frac{m(A)}{1-k} \end{cases} \quad (2.24)$$

Cela revient à redistribuer la masse de l'ensemble vide sur tous les autres sous-ensembles de Θ . Cette méthode est souvent utilisée en raison de sa simplicité, mais elle doit être employée avec précaution. En effet, si le conflit est élevé, la normalisation a pour effet d'accroître énormément les masses des autres hypothèses, ce qui n'a pas beaucoup de sens. De manière générale, il convient donc de vérifier la valeur du conflit lors de chaque fusion : si ce dernier est trop élevé, l'idéal est d'en rechercher les causes au cas par cas et de modifier la modélisation des fonctions de masse des sources.

D'autres méthodes de suppression du conflit existent, parmi lesquelles :

- affecter toute la masse de \emptyset à Θ (Yager, 1987);
- affecter la masse de \emptyset à tous les sous-ensembles d'hypothèses, proportionnellement à leurs masses respectives (Inagaki, 1991);
- affecter la masse de \emptyset à tous les sous-ensembles d'hypothèses, suivant diverses approches de pondération (Lefevre *et al.*, 2002)
- affecter le conflit issu de deux hypothèses contradictoires à l'ensemble constitué de l'union des deux hypothèses (Dubois et Prade, 1988), ce qui permet de représenter l'ignorance quant à ces deux hypothèses.

2.3.4.3. *Décision*

L'étape de décision consiste à sélectionner une hypothèse (simple ou composée) parmi l'ensemble des hypothèses de départ, en se basant sur la fonction de masse établie lors de l'étape de combinaison. La décision fait généralement intervenir deux fonctions évoquées plus haut : la *crédibilité* et la *plausibilité*.

La crédibilité mesure la confiance minimale que l'on peut avoir sur un ensemble d'hypothèses inclus dans Θ . Elle est définie par l'équation suivante :

$$\forall A \subseteq \Theta, Bel(A) = \sum_{B \subseteq A, B \neq \emptyset} m(B) \quad (2.25)$$

La crédibilité d'un ensemble d'hypothèses A est donc la somme des masses de tous les sous-ensembles de A . Si l'on travaille dans un cadre de monde fermé (c'est-à-dire $m(\emptyset) = 0$), alors $Bel(\Theta) = 1$.

La plausibilité mesure au contraire la confiance maximale que l'on a sur un ensemble d'hypothèses (équation 2.26). Il s'agit de la somme des masses des sous-ensembles qui ont au moins une hypothèse commune avec A .

$$\forall A \subseteq \Theta, Pls(A) = \sum_{A \cap B \neq \emptyset} m(B) \quad (2.26)$$

La plausibilité est toujours supérieure à la crédibilité :

$$\forall A \subseteq \Theta, Pls(A) \geq Bel(A) \quad (2.27)$$

Il est intéressant de noter que la plausibilité et la crédibilité sont égales (et égales à la fonction de masse) lorsque seules les hypothèses simples possèdent un poids. Cela revient à se placer dans un cadre probabiliste simple qui ne tire pas parti des possibilités offertes par la DST.

L'intervalle $[Bel(A), Pls(A)]$ est une bonne mesure de la quantité d'information que l'on possède sur l'hypothèse A : plus il est large, plus l'ignorance sur A est élevée.

Le calcul de la plausibilité et de la crédibilité fournit des mesures permettant de prendre une décision en faveur d'une hypothèse (simple ou composée). Plusieurs règles de décision existent (Bloch, 2003) parmi lesquelles :

- **Maximum de plausibilité** : il s'agit de sélectionner l'hypothèse dont la plausibilité est la plus élevée. Il faut préalablement décider quel nombre d'hypothèses on souhaite retenir, les choix les plus courants étant une seule (hypothèse simple) ou deux (hypothèse composée de deux hypothèses simples). Ce choix est important car une hypothèse composée a toujours une plausibilité plus élevée qu'une hypothèse simple.
- **Maximum de crédibilité** : de manière similaire à la règle précédente, il s'agit de retenir l'hypothèse dont la crédibilité est la plus élevée.
- **Maximum de crédibilité sans risque d'erreur** : on choisit l'hypothèse dont la crédibilité est supérieure ou égale aux plausibilités de toutes les autres hypothèses. En pratique, cette règle est bien souvent trop stricte et conduit rarement à un résultat.
- **Maximum de crédibilité avec rejet** (Le Hégarat-Masclé *et al.*, 1997) : on retient l'hypothèse dont la crédibilité est la plus élevée et dont la crédibilité est supérieure ou égale à la crédibilité de l'hypothèse complémentaire (par exemple, dans le cas d'une hypothèse simple θ_i , l'hypothèse complémentaire est $\Theta \setminus \{\theta_i\}$, c'est-à-dire l'ensemble constitué de toutes les autres hypothèses).
- **Maximum de probabilité pignistique** (Smets, 1989) : pour toute hypothèse simple A , la probabilité pignistique est définie de la manière suivante :

$$\forall A \subseteq \Theta, \text{Bet}P(A) = \sum_{B \ni A} \frac{m(B)}{|B|(1 - m(\emptyset))} \quad (2.28)$$

Cette règle permet de se replacer dans un contexte probabiliste, ce qui peut être utile lorsque l'on souhaite combiner le résultat de la fusion avec d'autres informations probabilistes.

2.3.4.4. Extensions à la théorie de Dempster-Shafer

Malgré ses nombreux points forts, la théorie de Dempster-Shafer possède quelques limitations que plusieurs approches tentent de dépasser. Par exemple, Gordon et Shortliffe (1985) ont proposé d'utiliser un modèle hiérarchique pour représenter l'ensemble des hypothèses, en partant de l'observation que toutes les combinaisons d'hypothèses ne sont pas forcément possibles dans certains contextes. Ce changement du mode de représentation des hypothèses implique des modifications des règles de combinaison.

Par ailleurs, Yen (1986) fait remarquer que la DST ne permet pas d'exprimer de manière formelle les relations entre sources d'information et hypothèses; pour pallier ce problème, il propose d'intégrer entre ces derniers des relations de probabilités partielles.

Mais l'extension de la DST la plus utilisée est probablement la théorie de Dezert-Smarandache (DSmT), formalisée par Dezert et Smarandache (2002). Cette théorie vise à éliminer trois grands points faibles de la théorie de Dempster-Shafer : la contrainte d'hypothèses exclusives (les hypothèses sont supposées totalement incompatibles entre elles), le principe du tiers exclu (si une hypothèse fait partie du cadre de discernement, son complémentaire au sens ensembliste en fait également partie), et surtout l'apparition de résultats incohérents en cas de conflit élevé. Ce cadre théorique a, entre autres, été utilisé pour fusionner des données issues du milieu de la télédétection (Corgne *et al.*, 2003; Hachicha et Chaabane, 2009; Mora *et al.*, 2011).

2.4. Conclusion partielle

La classification texturale est un vaste domaine où coexistent de nombreuses approches. Chacune possède ses avantages et s'avère efficace dans un cadre particulier; néanmoins, seules les approches orientées objet (approches géométriques ou structurales) sont adaptées aux images THRS, car elles considèrent une image comme résultant de la combinaison d'un ensemble d'objets détectables et analysables.

Ces approches reposent sur des algorithmes de détection d'objets ou de caractéristiques précises dans une image. Parmi ces algorithmes, le détecteur de points-clés SIFT (ainsi que ses variantes) possède certaines qualités le rendant particulièrement adapté aux images de télédétection : invariance aux changements de contraste, d'illumination, d'échelle et de rotation. Pour cette raison, il semble intéressant de mettre au point une méthode de classification texturale d'images THRS basée sur SIFT. Cette méthode sera détaillée dans le prochain chapitre.

Par ailleurs, la fusion d'information est un ensemble de méthodes connues pour améliorer la classification d'une image dans un contexte multi-sources. Parmi les grands types de fusion, seule la fusion évidentielle (basée sur la théorie de Dempster-Shafer) gère convenablement les concepts d'incertitude et d'imprécision, fréquemment rencontrés en télédétection. Cette approche, souvent mise en œuvre dans des cas simples de fusion des niveaux radiométriques, est toutefois peu appliquée à la fusion de descripteurs texturaux complexes. Nous verrons également dans le prochain chapitre comment ce type de fusion peut être décrit.

3. Méthodologie et données

3.1. Méthodologie

Le travail de recherche effectué est partagé en deux grandes étapes méthodologiques, représentées dans la figure 3.1. La première étape concerne la mise au point et la validation de l'algorithme de classification texturale proposé (nommé KPC, pour *Key-Point-based Classification*, dans les prochaines sections). La seconde étape est liée à l'élaboration d'une méthode de fusion d'images basée sur l'information texturale.

La validation de chaque étape est basée sur la vérité terrain disponible.

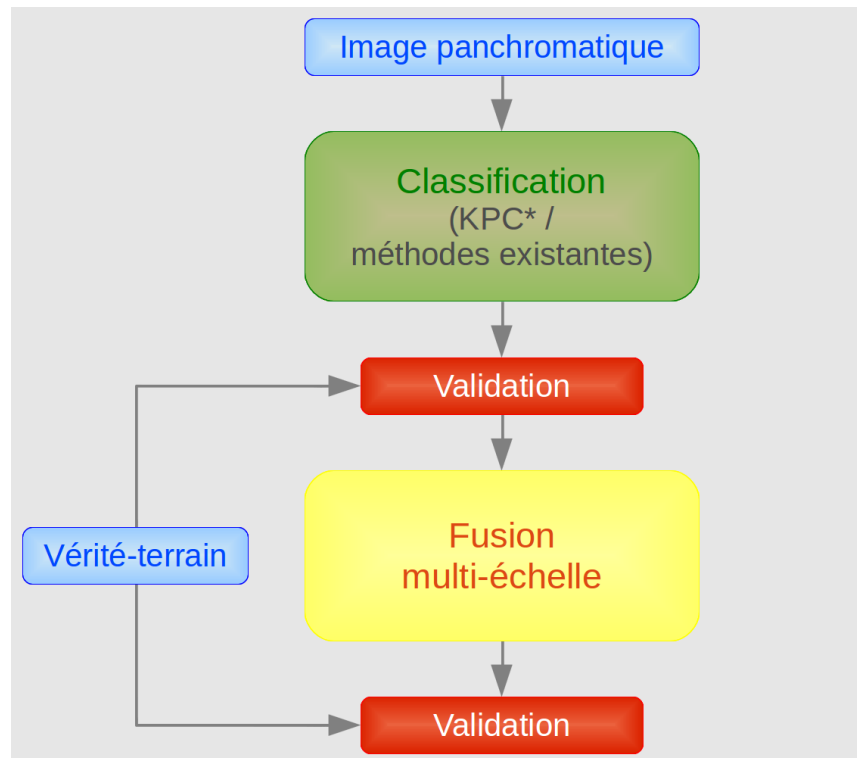


Figure 3.1 : organigramme méthodologique général (intrants en bleu; KPC : *keypoint-based classification*)

En résumé, l'algorithme KPC consiste à identifier et analyser un ensemble de points-clés (*keypoints*) dans le but de caractériser la texture présente à l'intérieur du voisi-

nage de chaque pixel de l'image soumise. Deux variantes en sont proposées, basées sur deux algorithmes de détection de points-clés différents : SIFT et SURF. Afin de valider les classifications issues de KPC, quatre autres algorithmes de classification texturale sont également testés : matrices de cooccurrence (indicateurs de Haralick), filtres de Gabor, spectre de texture et SFTA. Les résultats obtenus sont comparés objectivement à l'aide de la vérité-terrain.

Dans un second temps, l'algorithme KPC sert de base à une classification multi-échelle basée sur la théorie de l'évidence. Pour ce faire, l'image source est d'abord sous-échantillonnée d'un facteur 2 puis 4, afin d'obtenir plusieurs résolutions (ou échelles) de cette image. Chaque échelle est ensuite classifiée à l'aide de KPC; les classifications résultantes sont fusionnées de manière à obtenir une classification finale de l'image source. L'enjeu est ici d'adapter la théorie de l'évidence à un contexte de fusion de caractéristiques texturales.

3.1.1. Classification texturale

3.1.1.1. Algorithme proposé

a. Résumé

L'algorithme de classification texturale proposé est divisé en quatre étapes principales, décrites dans les paragraphes suivants. Le schéma ci-dessous résume ces étapes :

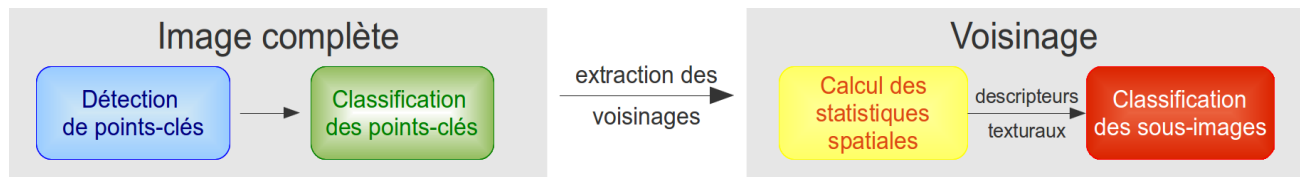


Figure 3.2 : étapes principales de l'algorithme de classification texturale

Notons que les deux premières étapes concernent l'image étudiée dans sa globalité, alors que les deux suivantes ont lieu au niveau des multiples voisinages (ou sous-

images) extraites de l'image globale. L'extraction des voisinages est également détaillée dans la section d.

b. Détection de points-clés

Deux algorithmes de détection de points-clés ont été utilisés : il s'agit de SIFT (*Scale Invariant Feature Transform*) et SURF (*Speeded Up Robust Features*). Le choix du détecteur SIFT a été guidé par ses performances (Mikolajczyk et Schmid, 2005). SURF a été testé en raison de sa rapidité d'exécution, bien que ses performances soient souvent inférieures à SIFT (Wu *et al.*, 2013).

Les implémentations choisies sont les suivantes : SIFT++ 0.8.1 d'Andrea Vedaldi³ et OpenCV 2.4.5 pour SURF⁴. Bien que la librairie OpenCV contienne une implémentation de SIFT, cette dernière n'a pas été retenue à cause de limitations concernant son paramétrage. À l'inverse, SIFT++ a des performances très proches de l'algorithme SIFT original de David Lowe (Bauer *et al.*, 2007).

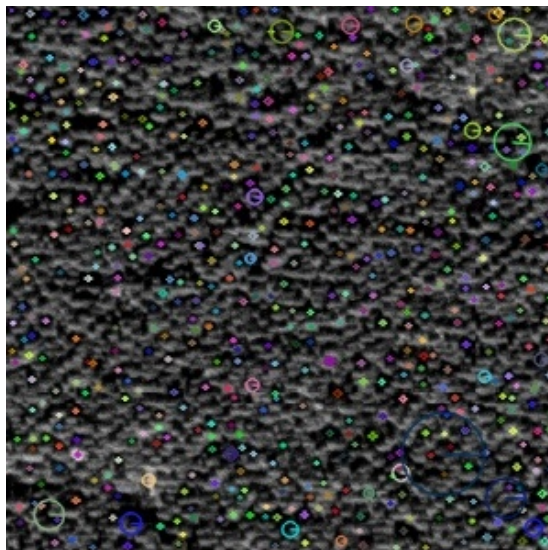


Figure 3.3 : exemple de points-clés SIFT détectés par l'algorithme SIFT++ sur une texture WorldView-2 (la taille des cercles indique l'échelle du point-clé, les couleurs n'ont pas de signification particulière)

³ <http://www.robots.ox.ac.uk/~vedaldi/code/siftpp.html>

⁴ <http://opencv.org/>

Paramètres

Les deux détecteurs utilisés requièrent de nombreux paramètres. Le Tableau 3.1 et le Tableau 3.2 résument les paramètres utilisés :

Tableau 3.1 : Paramètres SIFT utilisés.

Paramètre	Symbole	Description	Valeur
NumOctaves	O	Nombre d'octaves	5
FirstOctave	-	Index de la première octave	0
NumLevels	S	Nombre de niveaux dans une octave	3
Sigma0	-	Lissage de base	$1,6 * \sqrt[3]{2} \approx 2$
SigmaN	-	Lissage de chaque niveau	0,5
Threshold	t	Seuillage de l'intensité des extrema locaux	$6,7 * 10^{-3}$
EdgeThreshold	-	Seuillage directionnel des extrema locaux	10
Magnif	-	Facteur d'agrandissement de la zone de calcul de l'histogramme des directions	3

Tableau 3.2 : Paramètres SURF utilisés.

Paramètre	Symbole	Description	Valeur
hessianThreshold	-	Seuillage de l'intensité des extrema locaux	0,01
nOctaves	O	Nombre d'octaves	4
nOctaveLayers	S	Nombre de niveaux dans une octave	2
extended	-		true
upright	-		false

Chacun de ces détecteurs possède un algorithme descripteur, qui a pour rôle d'associer un vecteur descripteur de taille fixe à chaque point-clé. Dans la configuration choisie, les descripteurs SIFT comportent 128 éléments, alors que les descripteurs SURF en

possèdent 64.

Les descriptions techniques des détecteurs et descripteurs SIFT et SURF sont présentées dans le chapitre 2.

c. Classification des points-clés

Les vecteurs descripteurs des points-clés sont ensuite classifiés par un algorithme de type *k-means* (k-moyennes). Le nombre de classes de points-clés est fixé à 20, valeur qui mène aux meilleurs résultats pour tous les types d'image employés. Le but de cette opération est de mieux caractériser les points-clés dans la suite de l'algorithme.

Les centroïdes des partitions⁵ sont initialisés suivant la méthode d'initialisation (*seeding*) aléatoire d'Arthur et Vassilvitskii (2007). L'algorithme *k-means* cesse ses itérations lorsque l'une des deux limites suivantes est atteinte : l'un des centroïdes se déplace de moins de 0,001 unités de distance, ou le nombre d'itérations dépasse 10. Ces valeurs assurent un bon compromis entre rapidité et stabilité du partitionnement.

À l'issue de cette étape, on dispose donc d'un ensemble de plusieurs centaines à plusieurs milliers de points-clés regroupés en 20 classes. Le choix du nombre de classes sera exposé dans la section 4.1.1.

d. Extraction des voisinages

L'algorithme proposé étant basé sur l'analyse de l'information texturale, chaque pixel ne peut être classifié que par la prise en compte de son voisinage (et non de ses caractéristiques spectrales propres). Pour cette raison, un voisinage est extrait autour de chaque pixel que l'on souhaite classifier; l'analyse présentée au prochain paragraphe (calcul des statistiques spatiales) n'est effectuée qu'au sein de ce voisinage.

Vu le temps de calcul élevé que peut représenter la classification de tous les pixels

⁵ Le terme « partition » est ici utilisé comme traduction du mot « *cluster* » présent dans la littérature anglophone, et désigne un groupe de descripteurs au sein de l'espace textural. De la même manière, le terme « *clustering* » est ici traduit par « partitionnement ».

d'une image, la plupart des tests de performance ont été menés en ne classifiant qu'un pixel sur N (N étant généralement égal à 2 ou 4).

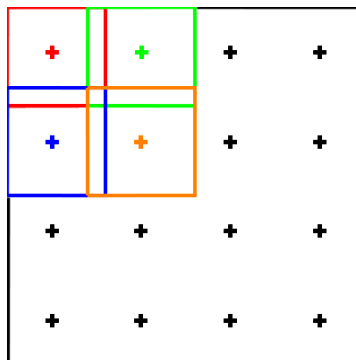


Figure 3.4 : exemple de répartition des voisinages au sein de l'image principale

La figure 3.4 illustre la répartition de ces voisinages, représentés par des cadres colorés; les croix symbolisent les pixels à classifier, exagérément espacés dans cet exemple. La taille des voisinages, c'est-à-dire la longueur de leurs côtés, joue un rôle fondamental dans les performances obtenues. C'est donc le principal paramètre qui affecte le taux de classification.

Les voisinages situés près des limites de l'image principale ont une taille plus petite que ceux situés à l'intérieur; cet effet de bord est pris en compte dans la suite de l'algorithme.

e. Statistiques spatiales

La partie la plus novatrice de l'algorithme proposé concerne le calcul de statistiques spatiales sur les points-clés pour caractériser les textures. La fonction K de Ripley (présentée dans la section 2.2.5.2) permet de modéliser numériquement la répartition des points-clés, à la fois en termes de densité mais aussi et surtout de régularité et d'homogénéité.

De manière générale, cette fonction permet de calculer la moyenne du nombre de points à l'intérieur d'un disque de rayon r . C'est donc un bon indicateur de la distance

inter-points moyenne, qu'elle permet d'ailleurs également de calculer. En calculant la valeur de la fonction K pour plusieurs valeurs de r , il est possible de détecter si la répartition des points est aléatoire, régulière, agrégée, etc. Ainsi, il suffit d'un vecteur contenant les valeurs de $K(r)$, avec r prenant plusieurs valeurs, pour caractériser une texture.

Concrètement, le calcul de $K(r)$ se fait en itérant sur tous les points-clés contenus dans une sous-image; à chaque itération, on détermine le nombre de points situés à une distance inférieure à r du point de référence, puis on divise le nombre obtenu par la surface du disque de rayon r en tenant compte des effets de bord. Ce principe est illustré par la figure 3.5, pour cinq valeurs de r et un seul point de référence.

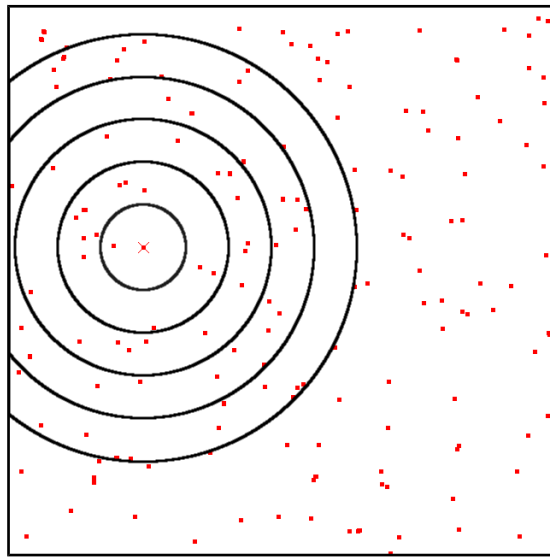


Figure 3.5 : exemple d'une itération de l'algorithme de calcul de statistiques spatiales, pour cinq rayons

En pratique, le nombre de valeurs de r est toujours fixé à 5, car les expérimentations menées sur plusieurs images et plusieurs jeux de paramètres ont montré qu'il s'agit toujours de la quantité optimale.

Rappelons que les points-clés ont préalablement été divisés en 20 classes (cf. paragraphe c). Grâce à cette classification, il est possible de calculer $K(r)$ pour chaque classe

de points-clés i ($1 \leq i \leq 20$). De manière générale, il est possible de calculer $K(r)$ pour une classe de points-clé j alors que le point de référence est de classe i . Les valeurs ainsi obtenues seront notées $K_{i,j}(r)$, $1 \leq i, j \leq 20$.

Si l'on prend en compte les différentes valeurs de r , que nous noterons r_k $1 \leq k \leq 5$, on constate que le calcul des statistiques spatiales résulte en $20 \times 20 \times 5 = 2\,000$ valeurs différentes de $K_{i,j}(r_k)$. L'ensemble de ces valeurs peut être arrangé en un vecteur, qui constitue le descripteur textural final de la sous-image étudiée.

Face à un vecteur de dimensionnalité aussi élevée, il est naturel de chercher à employer une méthode de réduction dimensionnelle telle que l'analyse en composantes principales (ACP). Cependant, les tests effectués en intégrant l'ACP n'ont pas permis de mettre en évidence une amélioration des classifications finales. Pour cette raison, la dimension des vecteurs descripteurs n'a pas été réduite.

f. Classification des descripteurs de texture

Les descripteurs texturaux ont été classifiés par deux méthodes de partitionnement concurrentes : l'algorithme *k-means* (en mode dirigé et non-dirigé) et l'algorithme *k-NN* (en mode dirigé). La majorité des tests ont été effectués avec *k-means* en mode non-dirigé, car l'un des objectifs de notre approche était de pouvoir fonctionner sans intervention humaine (excepté le choix du nombre de classes). Pour cette raison, des approches d'apprentissage plus sophistiquées (SVM, réseaux de neurones, *deep learning*, etc) n'ont pas été utilisées.

Deux types de résultats peuvent être obtenus à l'issue d'une telle classification : une classification dure et une classification douce.

Classification dure

Une classification dure (« *hard classification* ») consiste simplement à attribuer une classe unique pour chaque pixel. Dans le cas de l'approche par partitionnement, il

s'agit d'étiqueter chaque pixel par le numéro de la partition dont il fait partie. Cette approche simple, qui ne tient pas compte des effets de mélange de classes intra-pixel, a le mérite de fournir un résultat facilement exploitable et dont la qualité est aisément quantifiable (à l'aide de la vérité terrain, notamment). Pour comparer les performances des divers algorithmes testés, les classifications ont donc été effectuées de manière dure.

Classification douce

On parle de classification douce (« *soft classification* ») lorsque, pour chaque pixel, on attribue une probabilité – ou un poids – à chacune des classes existantes. Cette approche, plus fine que la précédente, permet de représenter les effets de l'indécision ou de l'incertitude, sans pour autant distinguer ces deux concepts. Il est difficile d'établir une mesure objective de la qualité d'une telle classification à moins de posséder une vérité-terrain dans laquelle figurent les zones ambiguës ou de transition, ce qui est rarement le cas. Même ainsi, le résultat d'une classification douce suppose de mettre en place des seuils arbitraires, et donc contestables : par exemple, à partir de quand un pixel est-il considéré comme pur, c'est-à-dire comme étant composé d'une seule classe?

En revanche, cette approche permet de juger visuellement le résultat d'une classification en mettant en évidence les zones d'indécision, qui peuvent être soit des zones de faiblesse de l'algorithme testé, soit de réelles zones de transition.

Plus important encore, la classification douce peut servir de base au processus de fusion évidentielle. En effet, le résultat de cette classification permet d'initialiser les masses de toutes les hypothèses, simples comme composées. À l'opposé, une classification dure ne permet d'initialiser que la masse d'une hypothèse, ce qui présente un intérêt très limité dans le cadre évidentiel. Or, il a été mentionné dans la section 2.3.4.1 l'importance particulière de l'étape d'initialisation des masses.

L'approche par partitionnement employée présente l'avantage d'être naturellement

compatible avec la classification douce. En effet, la distance séparant un échantillon des centroïdes des différentes partitions constitue une bonne mesure d'appartenance à une classe : plus cette distance est faible, plus l'échantillon peut être considéré comme faisant partie de la classe concernée. Pour cette raison, les diverses méthodes d'initialisation des masses présentées dans la section 3.1.2.2 suivent toutes des modèles de distance.

3.1.1.2. *Algorithmes comparatifs*

a. Matrices de co-occurrence

Les paramètres employés pour calculer les indicateurs de Haralick sont les mêmes pour chaque image. Les niveaux de gris ont été regroupés en huit classes de manière à ce que les matrices de cooccurrence conservent une taille raisonnable. Quatre directions ont été choisies : $d = (1,0), (1,1), (0,1), (-1,1)$ (cf. section 2.1.1.2). Enfin, quatre indicateurs ont été utilisés : contraste, corrélation, énergie et homogénéité. Les descripteurs texturaux ont donc une taille de 16 éléments.

b. Filtres de Gabor

Construire un banc de filtres de Gabor requiert un jeu de six paramètres (dans l'implémentation utilisée⁶). Certains de ces paramètres ont dû être modifiés d'une image à l'autre pour atteindre la meilleure classification possible; les autres sont demeurés constants d'une image à l'autre.

- k_s : taille du noyau (paramètre variable).
- σ : écart-type du filtre gaussien (paramètre variable).
- θ : liste d'orientations des filtres. Ce paramètre est constant et vaut $[0^\circ, 90^\circ]$.
- γ : liste des ratios spatiaux (longueur/largeur). Ce paramètre est constant et vaut $[1, 2]$.
- λ : liste des « longueurs d'onde » (c'est-à-dire $2\pi/\text{fréquence}$). Ce paramètre est

6 <https://github.com/dominiklessel/opencv-gabor-filter/blob/master/gaborFilter.cpp>

constant et vaut [1].

- ψ : liste des phases. Ce paramètre est constant et vaut $[0^\circ, 45^\circ, 90^\circ, 135^\circ]$.

On voit donc que la combinaison de tous les paramètres laisse $1 \times 1 \times 2 \times 2 \times 1 \times 4 = 16$ filtres à tester. Dans l'implémentation choisie, l'énergie de l'image résultante de chaque filtre constitue le descripteur textural. Ces derniers ont donc une taille de 16 éléments.

c. Spectre de texture

L'algorithme de spectre de texture, développé par He et Wang (1990), repose sur le concept d'unité texturale (« *texture unit* »), qui entre dans la catégorie des approches statistiques. Il attribue un « numéro d'unité texturale » N_{TU} à chaque pixel, calculé à partir des différences entre le niveau de gris du pixel et les niveaux de gris de ses voisins immédiats. Le spectre de texture d'une fenêtre est ensuite défini comme l'histogramme des numéros d'unité texturale au sein de la fenêtre. Ce spectre constitue donc un descripteur textural. Comme il existe 6 561 numéros d'unité texturale différents, les descripteurs possèdent 6 561 éléments. Cet algorithme ne nécessite pas de paramétrage.

Il existe une version simplifiée de cet algorithme (He et Wang, 2010), qui présente notamment l'avantage d'avoir un spectre de texture beaucoup moins long et donc un temps de traitement plus court. Cette version n'a toutefois pas été utilisée, car les tests effectués ont montré que la version originale aboutit à de meilleures classifications.

d. SFTA

SFTA (*Segmentation-based Fractal Texture Analysis*) est un algorithme créé par (Costa *et al.*, 2012). Il comporte deux principales étapes : décomposition d'une image en plusieurs images binaires à l'aide de plusieurs seuillages basés sur les niveaux de gris, suivi d'un calcul de certaines caractéristiques de ces images binaires (dont notamment la dimension fractale).

Cet algorithme comprend un seul paramètre à régler : il s'agit du nombre de seuils n_t , déterminant le nombre d'images binaires utilisées. Ce paramètre a été fixé à 4 pour toutes les images testées, car c'est la valeur donnant les meilleurs résultats.

Contrairement aux autres algorithmes d'analyse de texture qui ont été testés dans le cadre de ce projet de recherche, SFTA a été testé uniquement sous Matlab, à l'aide du code rendu public par son auteur⁷. Pour cette raison, la durée d'exécution de cet algorithme ne peut être comparée aux autres approches.

3.1.2. Fusion d'images

3.1.2.1. Généralités

La seconde partie principale de ce travail concerne la fusion de classifications texturales d'images THRS. Pour des raisons à la fois pratiques et liées aux résultats de la première partie, il a été décidé de s'orienter vers la fusion multi-échelle d'images panchromatiques, à l'aide de la théorie de l'évidence.

La figure 3.6 détaille l'approche méthodologique de cette partie. L'étape de fusion multi-échelle est mise en évidence dans le cadre gris. Elle comporte trois phases principales détaillées dans les prochaines sections : initialisation des fonctions de masse, combinaison de ces fonctions, décision.

Elle est précédée par une étape dite de sous-échantillonnage dont le but est de supprimer les fréquences les plus élevées de l'image originale (bruit, détails). Ce sous-échantillonnage linéaire, d'un facteur 2 et 4, est suivi d'une étape de sur-échantillonnage (non représentée dans la figure 3.6) bicubique afin que toutes les images sources possèdent la même taille.

⁷ <http://www.mathworks.com/matlabcentral/fileexchange/37933-sfta-texture-extractor>

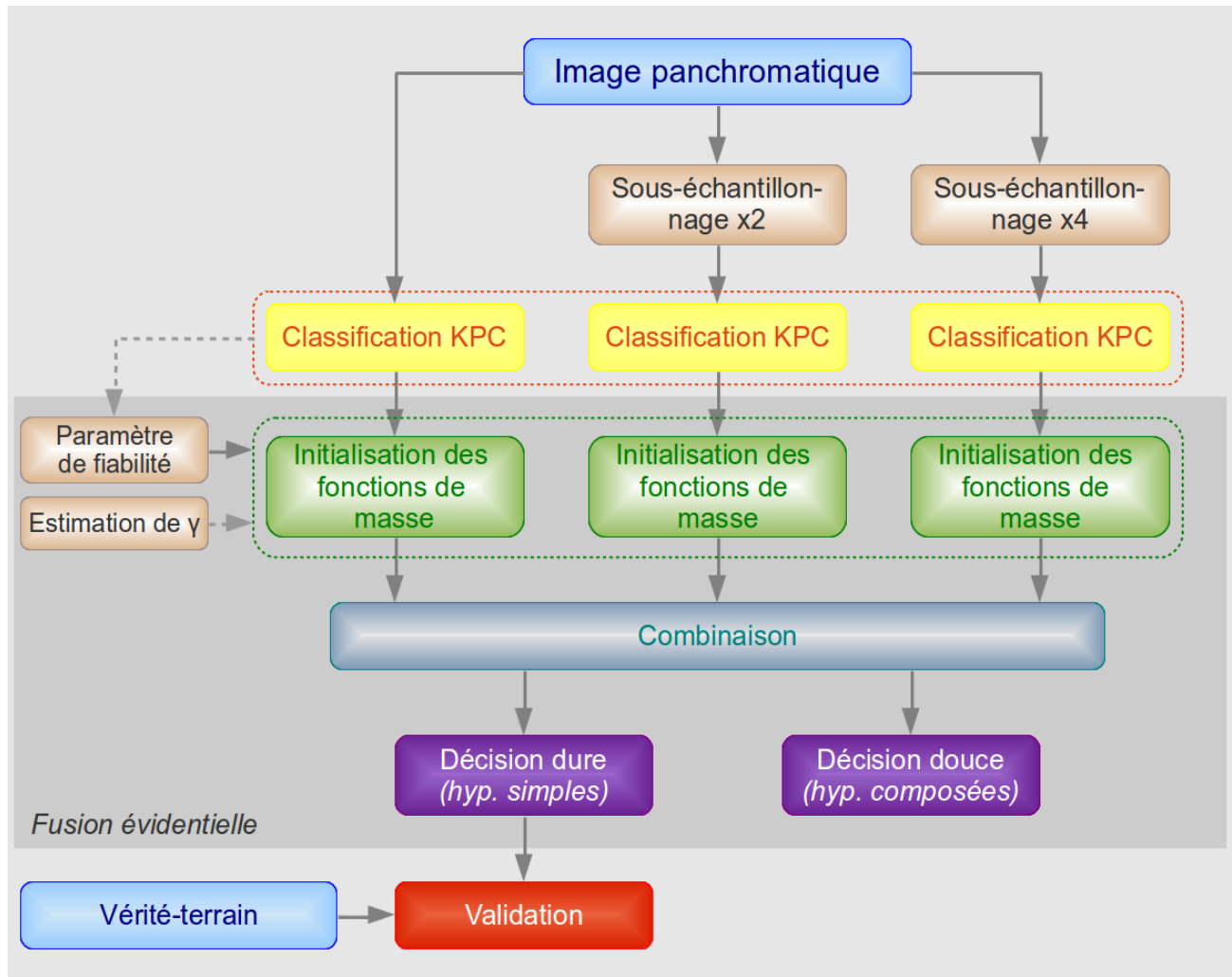


Figure 3.6 : organigramme méthodologique de l'étape de fusion multi-échelle. Les flèches pointillées désignent des étapes optionnelles

3.1.2.2. *Initialisation des fonctions de masse*

Le choix des masses à attribuer à chaque hypothèse (classe) ou ensemble d'hypothèses, pour chaque pixel de chaque image source, est délicat et doit faire l'objet d'une attention particulière. Il dépend de l'information disponible au sujet de chaque pixel à classer.

Chaque pixel est caractérisé par une information texturale se présentant sous la forme d'un vecteur descripteur de dimensionnalité variable, pouvant aller d'une dizaine à plus de mille éléments. Ces vecteurs, regroupés sous forme de partitions, permettent à

leur tour la classification des pixels. L'approche la plus naturelle pour juger de la masse à attribuer aux différentes hypothèses consiste donc à se baser sur les distances entre un vecteur descripteur et les centres des partitions détectées : plus un vecteur est proche d'une partition, plus la masse affectée à l'hypothèse représentée par cette partition sera élevée. La masse affectée à la partition dont fait partie le vecteur descripteur sera donc la plus élevée.

En notant $C_i, 1 \leq i \leq C$ (où C est le nombre de partitions, c'est-à-dire le nombre de classes identifiées) le centroïde de chaque partition et $p_{x,y}$ le vecteur descripteur du pixel situé aux coordonnées x, y et N la dimensionnalité de ces vecteurs, on définit la distance entre $p_{x,y}$ et C_i de la manière suivante :

$$d_n(C_i, p_{x,y}) = \left[\sum_{j=1}^N (C_i(j) - p_{x,y}(j))^n \right]^{1/n} \quad (3.1)$$

Suivant la valeur de n , plusieurs formes de distances peuvent être utilisées. La valeur $n = 2$, qui correspond à la distance euclidienne, a été retenue car il s'agit de l'une des distances qui donnent les meilleurs résultats dans un contexte de partitionnement. La distance d_2 sera simplement notée d par la suite.

a. Modèles de distance

Modèle de distance simple

Le premier modèle de distance retenu permet de définir des masses pour les hypothèses simples ($m(\{\theta_1\})$, $m(\{\theta_2\})$, etc) ainsi que pour l'ensemble de toutes les hypothèses $\Theta = \{\theta_1, \dots, \theta_K\}$. Développé par Denoeux (1995), il se base sur l'idée que plus un vecteur est éloigné d'une partition, plus la masse affectée à l'hypothèse représentée par cette partition doit être faible :

$$\begin{cases} m(\theta_i)(p) = \alpha e^{-\gamma d^2(C_i, p)} \\ m(\Theta)(p) = 1 - \sum_i m(\theta_i)(p) \end{cases} \quad (3.2)$$

α est un paramètre d'atténuation permettant de modéliser la fiabilité de la source. En l'absence de méthode simple pour juger de cette fiabilité, α est dans un premier temps laissé égal à 1, avant d'être analysé plus en détail. γ a trait à la forme des fonctions de masse et doit être évalué manuellement. Une valeur de 0,01 convient dans la plupart des cas, car les distances sont de l'ordre de quelques dizaines d'unités. Une justification plus avancée du choix de cette valeur se trouve dans l'annexe 1.

Modèle de distance composé

Le modèle de distance simple présente l'inconvénient de n'initialiser que les hypothèses simples, en plus de Θ . Cette approche n'exploite donc pas les possibilités offertes par la théorie de l'évidence. Nous proposons une autre approche, dans laquelle certaines hypothèses composées seront également initialisées : il s'agit du modèle de distance composé.

Ce dernier consiste à évaluer non seulement les distances séparant un vecteur des centroïdes des partitions, mais également les distances le séparant des points situés à mi-chemin de chaque couple de centroïdes. L'idée est que cette dernière distance reflète le poids à accorder aux hypothèses composées binaires.

Le système d'équations (3.2) devient donc :

$$\begin{cases} m(\{\theta_i\})(p) = \alpha e^{-\gamma d^2(C_i,p)} \\ m(\{\theta_i, \theta_j\})(p) = \alpha e^{-\gamma d^2((C_i+C_j)/2,p)} \end{cases} \quad (3.3)$$

Les masses sont ensuite normalisées.

b. Coefficient de fiabilité

Le coefficient de fiabilité (aussi appelé coefficient d'affaiblissement), rarement étudié dans la littérature, a fait l'objet d'une attention particulière. En effet, plusieurs auteurs (Fabre *et al.*, 2001; Mora, 2009) ont souligné le potentiel d'amélioration de la fusion dû à ce coefficient, en mentionnant toutefois la difficulté de l'estimer correctement

(Lefevre *et al.*, 2002). Plusieurs modélisations ont été testées pour tenter d'estimer la fiabilité de la classification des images sources.

Fiabilité multi-échelle

Cette approche, valide dans le cas d'une fusion multi-échelle uniquement, consiste à définir la fiabilité comme étant une fonction décroissante de l'échelle de l'image source. On part du principe qu'une image de résolution élevée devrait aboutir à une classification de meilleure qualité.

La fiabilité α est ainsi modélisée :

$$\alpha = 1 - \delta e \quad (3.4)$$

Ici, e est un entier supérieur ou égal à 0 représentant l'index de l'échelle : 0 correspond à la résolution de base, 1 à la résolution immédiatement inférieure, etc. δ représente un paramètre qui a été fixé à 0,2 dans les expérimentations (essais de plusieurs valeurs).

Fiabilité par estimation a priori de la répartition des classes

Une autre approche, un peu plus générique que la précédente, a été développée. Elle repose sur une estimation a priori de la fréquence des classes à détecter. On peut ensuite mesurer la dissimilarité entre cette estimation et la classification source pour estimer la fiabilité de cette dernière. En notant p_i^e la proportion estimée de la classe i au sein de l'image source et p_i la proportion de cette même classe dans la classification obtenue, la fiabilité peut être calculée de la manière suivante :

$$\alpha = \left(1 - \frac{K-1}{K} \sum_{i=0}^K (p_i^e - p_i) \right)^d \quad (3.5)$$

K est le nombre de classes et d est un paramètre fixé à 20 afin de permettre une variation assez large de α .

Autres approches

Dans le but d'éviter d'avoir recours à des informations a priori qui ne sont pas toujours connues, d'autres approches ont été tentées. Elles reposent sur le calcul de plusieurs indicateurs liés à la phase de partitionnement, tels que la densité des partitions et la distance les séparant. Ces approches n'ayant pas fourni de résultats concluants, elles ne seront pas détaillées.

De manière générale, plusieurs niveaux de finesse peuvent être envisagés lors de l'estimation de α :

- estimation par pixel : α varie non seulement en fonction de la classification source, mais aussi en fonction de certaines caractéristiques propres à chaque pixel. C'est un niveau extrêmement fin, impossible à atteindre actuellement, car il n'existe pas d'approche permettant de juger de la fiabilité locale au sein d'une image.
- estimation par classe : il s'agit ici de faire varier α pour chaque classe et chaque classification source. Cette approche, plus générale que la précédente, pourrait être mise en place à partir du calcul d'indicateurs liés aux partitions de descripteurs texturaux.
- estimation globale : α est constant à l'intérieur de chaque classification source. Il s'agit de l'approche la plus grossière mais la plus réalisable en pratique. C'est celle qui a été retenue, car elle est la seule compatible avec les deux premières modélisations présentées dans cette section.

3.1.2.3. Règles de combinaison

Trois règles de combinaison ont été mises en œuvre : les combinaisons conjonctive, disjonctive et robuste. Ces trois combinaisons offrent en général des résultats proches.

3.1.2.4. *Règles de décision*

Trois règles de décision ont été testées : le maximum de plausibilité, le maximum de crédibilité et le maximum de probabilité pignistique. Là encore, les résultats obtenus sont généralement proches.

3.1.3. *Évaluation des performances*

3.1.3.1. *Taux de classification*

Le taux de classification est le critère quantitatif d'évaluation de la performance de classification le plus simple. Il est basé sur le nombre de pixels dont la classe correspond à celle de la vérité-terrain, rapporté au nombre total de pixels. C'est un indicateur grossier qui ne permet notamment pas de distinguer les performances de classification interclasses, mais qui permet de comparer rapidement deux classifications effectuées sur la même image.

3.1.3.2. *Coefficient kappa*

Le coefficient kappa de Cohen (κ) permet d'éliminer les « fausses » bonnes classifications dues au hasard. En notant M la matrice de confusion définie dans la section suivante et N le nombre total de pixels, le coefficient kappa est calculé de la manière suivante :

$$\kappa = \frac{P_a - P_e}{1 - P_e} \quad (3.6)$$

où P_a est le taux de pixels correctement classifiés (cela correspond au taux de classification précédemment défini) :

$$P_a = \frac{1}{N} \text{Tr}(M) \quad (3.7)$$

et P_e est la proportion de pixels correctement classifiés « par chance », déterminé par le calcul des probabilités marginales de classification par classe :

$$Pe = \frac{1}{N^2} \sum_i \left[\sum_j M(i, j) \sum_j M(j, i) \right] \quad (3.8)$$

Un coefficient kappa supérieur à 60 % est considéré comme bon, alors qu'un taux supérieur à 80 % est excellent.

3.1.3.3. *Matrice de confusion*

La matrice de confusion est l'outil le plus complet pour juger de la qualité d'une classification. Elle permet d'analyser classe par classe les performances de classification d'un algorithme, ainsi que de calculer le taux de classification et le coefficient kappa de Cohen.

3.2. *Données*

3.2.1. **Mosaïque IKONOS**

Nous avons utilisé une image IKONOS panchromatique, acquise en 2001 et représentant le secteur de Happy Valley – Goose bay, au Labrador (Canada). Sa limite de résolution spatiale est de 82 cm. Cette image (figure 3.7) a fait l'objet d'une photointerprétation, permettant de définir cinq zones de texture uniforme assemblées dans une mosaïque (figure 3.8). Cette mosaïque a pour dimensions 212×344 px.



Figure 3.7 : image IKONOS de base

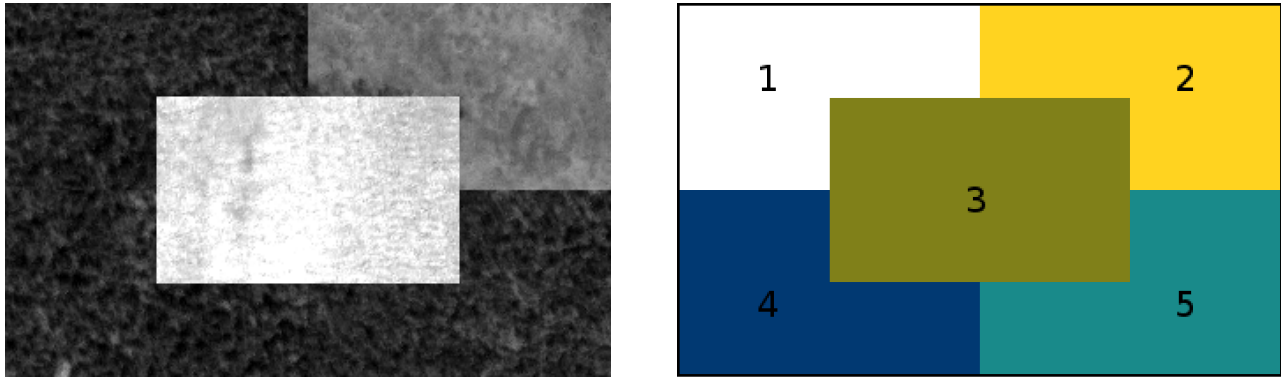


Figure 3.8 : mosaïque IKONOS (vérité-terrain à droite)

Les cinq textures apparaissant dans la mosaïque sont :

1. eau (centre);
2. forêt d'épinettes noires, recouvrement de 26 à 50 % (coin supérieur gauche);
3. végétation arbustive (coin supérieur droit);
4. forêt d'épinettes noires et de sapins, recouvrement de 51 à 75 % (coin inférieur gauche);
5. forêt d'épinettes noires, recouvrement de 51 à 75 % (coin inférieur droit).

On peut constater qu'au sein de la mosaïque, les trois zones de forêt sont difficiles à distinguer, même pour l'œil humain. Cette image constitue donc un bon test de classification.

3.2.2. Mosaïque WorldView-2

Cette image de la région de Sherbrooke a été acquise le 30 août 2010 par le satellite WorldView-2 en mode panchromatique (figure 3.9). Sa limite de résolution spatiale est de 52 cm. La photointerprétation des zones rurales a permis d'identifier six classes et d'extraire un carré de 300 × 300 px pour chaque classe. Une mosaïque de six textures a ainsi pu être élaborée (figure 3.10).

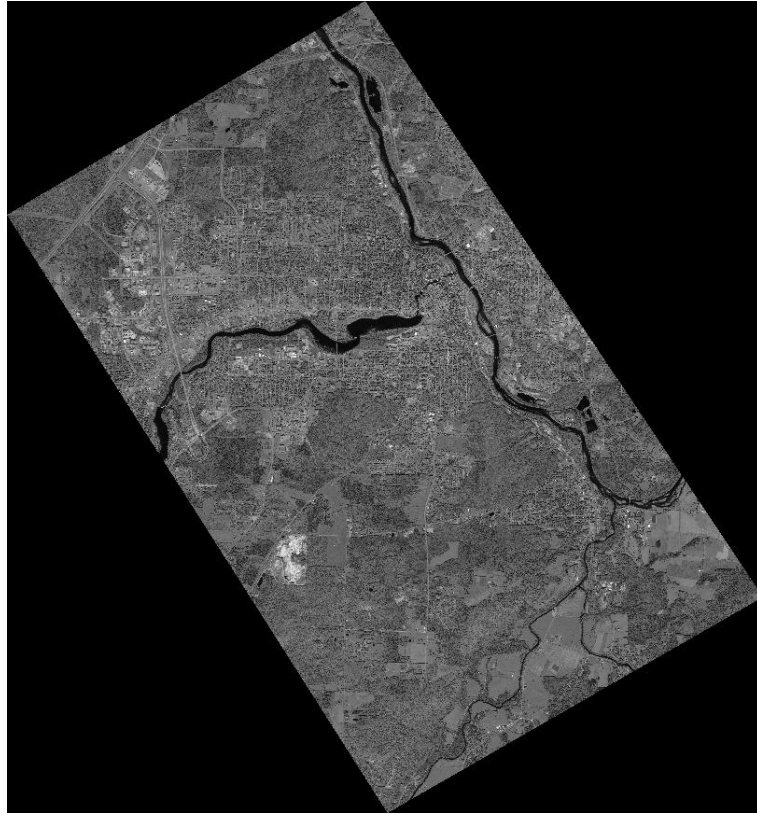


Figure 3.9 : image WorldView-2 de base

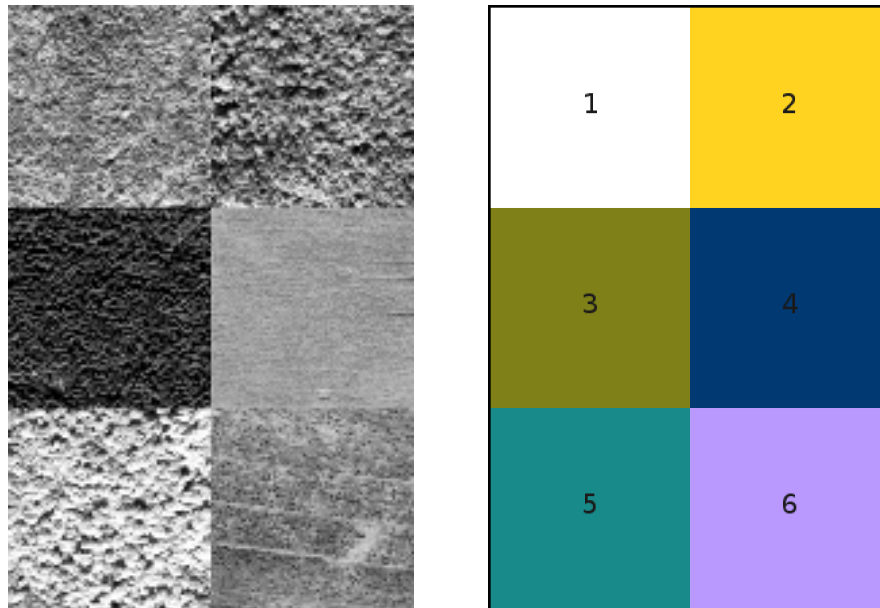


Figure 3.10 : mosaïque WorldView-2 (vérité-terrain à droite)

Les six textures de la mosaïque sont, de gauche à droite et de haut en bas :

1. Jeune forêt de feuillus indéterminés;
2. Forêt de feuillus incluant érables à sucre, feuillus intolérants et tolérants à l'ombre;
3. Plantation mature d'épinettes blanches;
4. Champ de maïs;
5. Forêt d'érables à sucre mature;
6. Jeune plantation de sapins baumiers.

3.2.3. Placette IKONOS

La dernière image utilisée est une placette extraite de l'image IKONOS principale (Fig. 3.7), de dimensions 444×360 px. Les thèmes présents sont similaires :

1. forêt d'épinettes noires, recouvrement de 51 à 75 % :
2. forêt d'épinettes noires, recouvrement inférieur à 25 %;
3. forêt d'épinettes noires, recouvrement de 26 à 50 %;
4. forêt d'épinettes noires et de sapins, recouvrement de 51 à 75 %;
5. zone en régénération;
6. végétation arbustive.

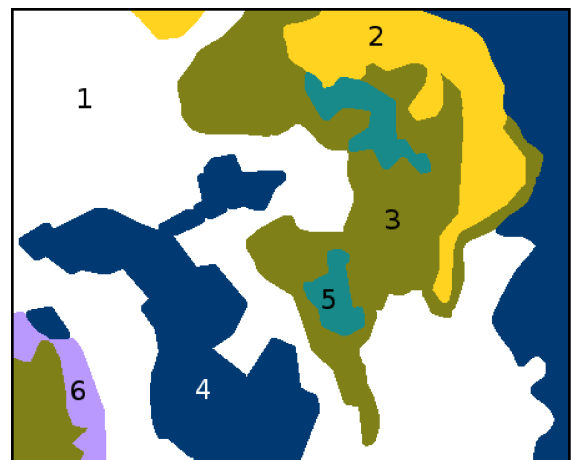
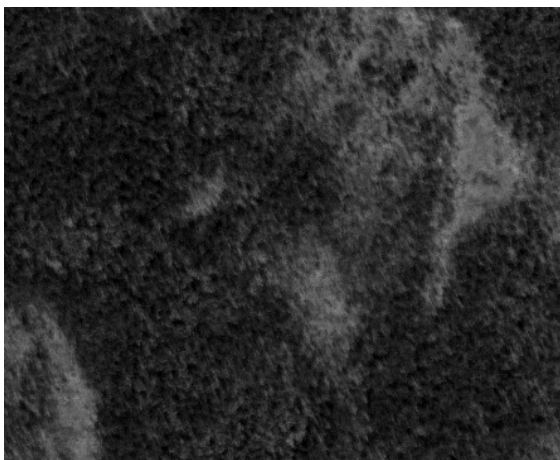


Figure 3.11 : placette IKONOS (vérité-terrain à droite)

3.3. Conclusion partielle

Une nouvelle méthode de classification texturale a été proposée, basée sur la détection, la classification et l'analyse spatiale de points-clés au sein des images étudiées. Deux variantes de cette méthode existent, en fonction de l'algorithme de détection de points-clés choisi (SIFT ou SURF). SURF étant inspiré de SIFT, les deux variantes devraient fournir des résultats proches.

En complément de cette méthode, une approche de fusion multi-échelle est exposée. Elle repose sur la théorie de l'évidence (théorie de Dempster-Shafer) qui permet de représenter les concepts importants en télédétection que sont l'incertitude et l'imprécision. Cette approche devrait permettre de compenser au moins en partie l'imprécision des frontières inter-texturales détectées, due à la nature de l'algorithme (taille élevée des voisinages). La proposition d'un nouveau modèle d'initialisation des fonctions de masse (modèle de distance composé) et de modèles d'estimation du coefficient de fiabilité devrait encore améliorer les résultats.

Les algorithmes comparatifs sont variés et appartiennent à des catégories différentes : méthodes statistiques (matrices de co-occurrence, spectre de texture), méthodes fréquentielles (filtres de Gabor) et méthodes fractales (SFTA). Ils devraient donc fournir des comparaisons pertinentes et permettre une validation raisonnable.

Les images utilisées sont représentatives des données rencontrées en télédétection optique THRS. Deux images synthétiques sont essentiellement constituées de textures de foresterie et agricoles, issues d'IKONOS et de WorldView-2. La troisième image est directement extraite d'une scène IKONOS et présente le même type de textures, ainsi que des frontières beaucoup plus complexes. Ces images constituent donc une base solide de test, permettant d'apprécier au mieux les forces et faiblesses de l'algorithme de classification proposé.

4. Résultats

4.1. *Classifications texturales*

Cette section est consacrée à l'étude des performances de l'algorithme de classification texturale proposé, dans un cadre mono-échelle. La prochaine sous-section vise à analyser le rôle des divers paramètres liés à cet algorithme, en l'appliquant sur l'image constituée d'un assemblage de textures IKONOS (voir section 3.2.1). La troisième sous-section a pour but de comparer l'algorithme proposé aux autres algorithmes présentés dans le chapitre 2. Enfin, la dernière sous-section met en relief les différences de classification entre les deux images étudiées.

4.1.1. **Influence des paramètres**

L'algorithme de classification texturale proposé est régi par de nombreux paramètres. Pour rappel, on peut le décomposer en cinq étapes (cf. section 3.1.1.1) :

- Détection de points-clés;
- Classification des points-clés;
- Extraction des sous-images;
- Calcul des statistiques spatiales;
- Classification des descripteurs de texture.

Les paramètres liés aux quatre premières étapes sont étudiés dans cette section, alors que la dernière étape est analysée dans la section suivante. Les paramètres de la première étape (détection de points clés) sont listés dans les tableaux 3.1 et 3.2. Le tableau 4.1 résume les paramètres liés aux étapes deux à quatre :

Tableau 4.1 : paramètres utilisés (les paramètres analysés par la suite sont en gras).

Paramètre	Étape	Description
n_{C-KP}	2. Classification des points-clés	Nombre de classes de points-clés
ε_{KP}	2. Classification des points-clés	Critère d'arrêt de la classification (variation minimum des centroïdes)
max_{iter}	2. Classification des points-clés	Critère d'arrêt de la classification (nombre maximum d'itérations)
w	3. Extraction des sous-images	Taille de voisinage (en pixels)
s	3. Extraction des sous-images	Distance entre pixels classifiés
R	4. Calcul des statistiques spatiales	Distances utilisées pour le calcul de la fonction K de Ripley (vecteur)
n_{PCA}	4. Calcul des statistiques spatiales	Nombre de composantes principales conservées

Dans toute cette section, l'impact de chaque paramètre sera testé sur l'assemblage de textures IKONOS, en mode non dirigé. Les résultats sont présentés de la manière suivante : on fait varier chaque paramètre alors que les autres paramètres sont fixes. La valeur des paramètres fixes est choisie de manière à mettre en évidence l'effet de la variation du paramètre étudié.

Détection de points-clés

En ce qui concerne la détection de points-clés, les paramètres les plus importants sont le nombre d'échelles à l'intérieur d'une octave (S), et le seuil de détection t (cf. tableaux 3.1 et 3.2) à la fois pour SIFT et SURF.

Le tableau 4.2 illustre le rôle du paramètre S sur le résultat de la classification. Les valeurs des autres paramètres sont proches des valeurs assurant les meilleurs taux de bonne classification (TBC), de sorte que l'évolution du TBC en fonction de S est représentative. On constate ainsi que bien que l'augmentation de S entraîne celle du nombre de points-clés détectés – ce qui est cohérent avec la manière dont le détecteur SIFT

fonctionne – le TBC n’augmente pas nécessairement. En règle générale, un pic de TBC est atteint pour les valeurs de S égales à 3 ou 4; la valeur 3 est retenue dans la suite des tests.

Tableau 4.2 : influence du paramètre S sur le nombre de points-clés et le TBC.

S	Nombre de points-clés détectés	Taux de bonne classification (%)
2	173	67
3	275	74
4	362	71
5	417	71
6	468	73

Le seuil de détection des points-clés t a quant à lui une faible influence sur la qualité finale de la classification (tableau 4.3). La valeur originale du paramètre ($6,67 \cdot 10^{-3}$) a donc été conservée.

Tableau 4.3 : influence du paramètre t sur le nombre de points-clés et le TBC.

t	Nombre de points-clés détectés	Taux de bonne classification (%)
10^{-3}	275	74
$6,67 \cdot 10^{-3}$	275	74
10^{-2}	275	74
10^{-1}	275	74
1	273	71

La figure 4.1 met en lumière une propriété intéressante du détecteur SIFT, en établissant un lien entre le nombre de points-clés détectés et le taux de bonne classification.

Chaque point correspond à une classification effectuée avec différents paramètres du détecteur SIFT, dont les classifications des tableaux 4.2 et 4.3. On observe qu'il existe une corrélation positive entre ces deux variables, avec un coefficient de corrélation de 0,40. Bien que ce taux ne soit pas élevé, on peut en conclure qu'un nombre accru de points-clés SIFT est un facteur d'amélioration de la classification.

Il faut toutefois nuancer ce constat, car un nombre très élevé de points-clés ne mène pas nécessairement à une meilleure classification. En effet, l'algorithme SIFT n'interdit pas que plusieurs points-clés soient détectés au même emplacement ou, si les paramètres de détection sont trop flexibles, à des emplacements très proches (au sein du même pixel, par exemple). Ce phénomène de redondance est à opposer à la pertinence des points détectés : il est préférable d'obtenir un nombre réduit de points qui décrivent adéquatement la structure de la texture, plutôt qu'un grand nombre de points dépourvus de sens. C'est pour cette raison qu'il n'est pas souhaitable de « pousser » les paramètres de détection à leurs limites dans le but de détecter le maximum de points-clés.

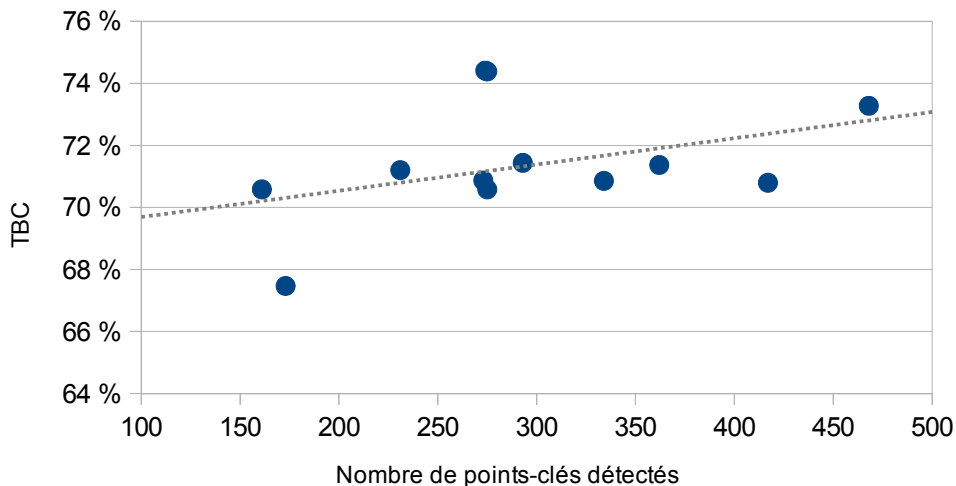


Figure 4.1 : taux de bonne classification en fonction du nombre de points-clés détectés.

Les paramètres sont : $s = 5$, $w = 161$, $n_{C-KP} = 20$, $n_{PCA} = 0$, $R = [8, 16, 24, 32, 40]$

Nombre de classes de points-clés (n_{C-KP})

Les points-clés détectés par SIFT sont classifiés par l'algorithme *k-means*, par l'intermédiaire de leurs descripteurs. Le principal paramètre utilisé est le nombre de classes de points-clés. La figure 4.2 illustre la variation du TBC et du coefficient kappa en fonction de ce paramètre. Bien que les courbes soient globalement croissantes, elles présentent une variabilité importante.

On peut également noter que ce paramètre possède une forte influence sur le temps de calcul (courbe verte). Pour cette raison, la valeur retenue de n_{C-KP} n'est pas la valeur donnant le meilleur TBC lors de la phase de paramétrage, mais résulte plutôt d'un compromis entre temps de calcul et qualité de classification. Cette valeur est de 20 : c'est cette valeur qui fournit généralement le meilleur résultat, au sein de la fourchette [5; 30].

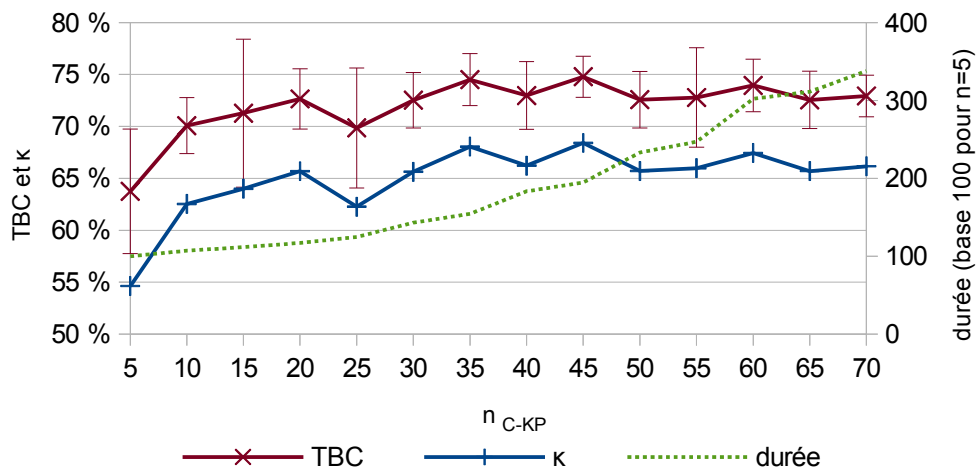


Figure 4.2 : TBC, kappa et durée de calcul en fonction du nombre de classes de points-clés n_{C-KP} . Les autres paramètres sont : $s = 5$, $w = 161$, $n_{PCA} = 0$, $R = [8, 16, 24, 32, 40]$

Taille de voisinage (w)

La taille de voisinage, c'est-à-dire la longueur en pixels du côté des sous-images centrées sur chaque pixel à classifier, est le facteur le plus déterminant du processus de classification. Il varie d'une image à l'autre, et doit être déterminé manuellement à chaque fois. La figure 4.3 met en évidence la variation typique du TBC en fonction de w :

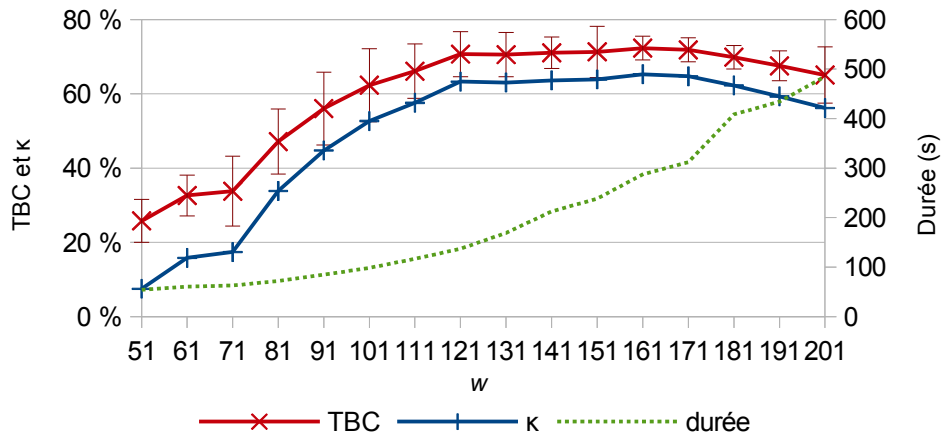


Figure 4.3 : TBC (en rouge), kappa (en bleu) et temps d'exécution (en vert) en fonction de la taille de voisinage w . Les autres paramètres sont : $s = 2$, $n_{C-KP} = 20$, $n_{PCA} = 0$, $R = [8, 16, 24, 32, 40]$

La courbe de variation du TBC en fonction de w présente toujours la même allure : une augmentation rapide jusqu'à la valeur maximale, suivie d'une stagnation ou d'une diminution lente. Ces deux phases s'expliquent de la manière suivante : les faibles valeurs de w font en sorte que les sous-images sont trop petites pour contenir suffisamment de points-clés pour caractériser les textures. Le nombre de points-clés inclus dans les sous-images augmente ensuite proportionnellement à w^2 , ce qui mène à un accroissement rapide du TBC. Une fois la valeur maximale atteinte, l'augmentation de w souffre des effets de bord : plus w est élevé, plus le risque d'avoir un mélange de textures au sein des sous-images est grand, et donc moins le descripteur textural qui en

résulte sera pertinent.

Pour résumer, l'augmentation de la valeur du paramètre w a pour conséquence deux effets opposés : d'une part, un plus grand nombre de points-clés est inclus dans les sous-images, ce qui a un effet positif sur la classification; d'autre part, un mélange de textures (et donc de classes) de plus en plus fort, ce qui a un effet négatif. La valeur optimale de w est donc celle qui offre le meilleur compromis entre ces deux effets.

Il faut enfin noter le lien existant entre la taille de voisinage et le temps de calcul : ce dernier est en effet proportionnel au carré de w . C'est un facteur qui joue un rôle important lorsque l'on se fixe comme priorité la durée de la classification : on peut ainsi diminuer cette durée au coût d'une baisse (parfois minime) de la qualité de la classification.

Distances de la fonction K de Ripley (R)

R est un vecteur contenant les distances utilisées pour le calcul de la fonction K de Ripley (cf. section 2.2.5.2). Pour des raisons de simplicité, ces distances sont régulières; autrement dit, $R = [r, 2r, 3r, \dots]$. Le nombre d'éléments de R possède une légère influence sur la qualité de la classification : de manière générale, plus ce nombre est élevé et plus la classification est bonne. Cependant, le temps de calcul est proportionnel au carré du nombre d'éléments de R . Un bon compromis est de garder ce nombre à 5 : ainsi, R s'écrit $[r, 2r, 3r, 4r, 5r]$. La valeur optimale de r dépend essentiellement de l'image à étudier et non des autres paramètres. C'est ce qu'indique le tableau 4.4 : lorsque le paramètre w varie, le meilleur taux de bonne classification est généralement atteint pour $R = [14, 28, 42, 56, 70]$.

Tableau 4.4 : TBC moyen en fonction des paramètres R et w . Pour chaque valeur de w , le meilleur TBC moyen est mis en gras.

R	w			
	131	141	151	161
[8, 16, 24, 32, 40]	77	76	78	77
[14, 28, 42, 56, 70]	79	78	78	78
[16, 32, 48, 64, 80]	77	76	78	78
[18, 36, 54, 72, 90]	77	77	77	78

Nombre de composantes principales (n_{PCA})

Il est possible d'effectuer une analyse en composantes principales (ACP) pour réduire la dimensionnalité des vecteurs descripteurs de texture, qui comportent 2 000 éléments (cf. section 3.1.1.1. e), et ainsi améliorer la classification de ces vecteurs. Le graphique suivant résumé les résultats obtenus pour plusieurs nombres de composantes principales retenues :

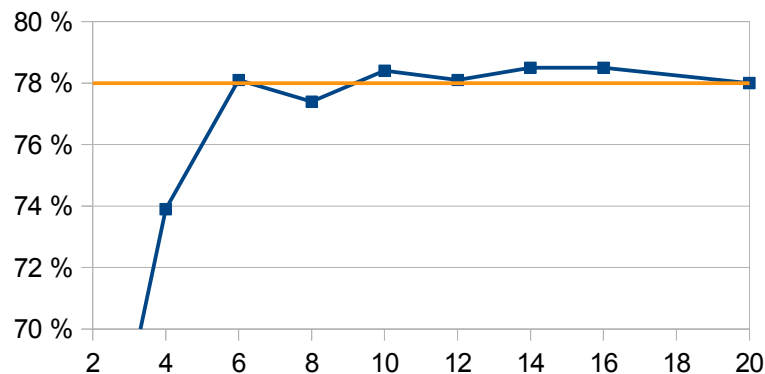


Figure 4.4 : TBC en fonction de n_{PCA} (courbe bleue). La ligne orange indique le TBC obtenu sans ACP

Avec le jeu de paramètres utilisé pour obtenir la figure 4.4, le TBC sans ACP est de 78 %. Avec ACP, le meilleur TBC est de 79 %, obtenu pour 16 composantes principales. En revanche, l'utilisation de l'ACP multiplie par environ 10 le temps de calcul requis. Cet inconvénient majeur, rapporté au gain minime de TBC, fait que l'ACP n'a en pratique pas été utilisée pour le reste des tests.

4.1.2. Comparaison avec d'autres algorithmes de classification

4.1.2.1. Mosaïque IKONOS

a. Classification non dirigée

Le tableau suivant résume les résultats obtenus pour la classification non dirigée de la mosaïque IKONOS, à l'aide de l'algorithme de partitionnement *k-means*. L'initialisation (*seeding*) de *k-means* étant aléatoire, chaque classification peut mener à un résultat légèrement différent. Pour cette raison, les taux de bonne classification (TBC) et coefficients kappa affichés sont des moyennes issues de plusieurs classifications. Les écarts-types de TBC sont également listés afin de donner une idée de la variabilité des résultats.

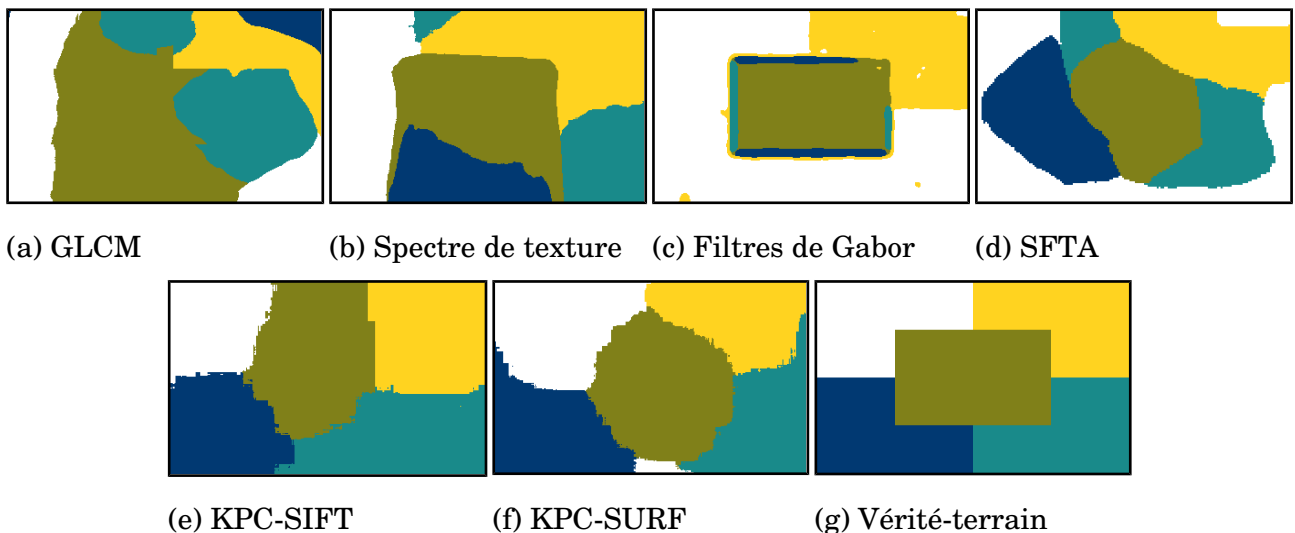


Figure 4.5 : résultats de la classification non dirigée de la mosaïque IKONOS

Tableau 4.5 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la mosaïque IKONOS. Les meilleures valeurs sont indiquées en gras.

		TBC (%)	σ_{TBC} (%)	κ	Durée moy. (s)
KPC	SIFT	79	0,0	0,73	210
	SURF	82	0,2	0,78	925
GLCM		48	2,4	0,34	600
Spectre de texture		72	0,0	0,65	370
Filtres de Gabor		58	0,0	0,48	1 080
SFTA		57	0,2	0,46	-

La figure 4.5, qui représente les résultats obtenus, permet de constater la présence d'une importante confusion interclasse pour la plupart des algorithmes testés. Cette confusion est due à la forte ressemblance que certaines textures présentent entre elles, notamment les textures 1, 4 et 5 (voir figure 3.8). Cela est particulièrement visible sur la figure 4.5 (c). Le nombre de classes à détecter (ici 5) étant imposé, cette confusion mène à la détection de classes inexistantes ou au mélange de plusieurs classes en une seule (figures 4.5 (a) à (d)).

Seuls les algorithmes KPC parviennent à discriminer correctement les textures présentes (figures 4.5 (e) et (f)), comme le reflètent les taux de bonne classification du tableau 4.5. Ces TBC sont supérieurs de 6 à 10 % au meilleur TBC des algorithmes concurrents. De même, le coefficient kappa présente un gain de 8 à 12 % par rapport au meilleur kappa concurrent. Il est important de constater que l'algorithme KPC parvient à différencier des textures situées dans une même image, par opposition à la classification de jeux d'images. En effet, contrairement à l'application précitée, la classification de textures au sein d'une image force l'algorithme de classification à être confronté au problème de mélange de textures au sein des voisinages des pixels situés aux frontières de ces textures. Ce problème est d'ailleurs d'autant plus présent que les voisinages sont grands, comme c'est le cas pour l'algorithme KPC. Ce dernier parvient

cependant à surmonter cette difficulté.

Bien que les frontières soient simples, elles ne sont toutefois pas parfaitement détectées. Comme nous le verrons à plusieurs reprises, ce problème constitue l'inconvénient majeur de l'algorithme KPC. En effet, cet algorithme repose sur l'analyse de voisinages assez vastes pour contenir au moins plusieurs dizaines de points-clés. Par exemple, les voisinages utilisés dans les figures 4.5 (e) et (f) sont respectivement de 131 et 151 pixels, ce qui est du même ordre de grandeur que la taille des morceaux de textures (106×172 px). Pour obtenir une détection plus fine des frontières, il faudrait que les textures aient une taille significativement plus élevée que les voisinages utilisés; ce cas de figure n'est cependant envisageable que pour des images de résolution extrêmement élevée (de l'ordre de la dizaine de centimètres par pixel). Comme on le verra dans la section consacrée à la mosaïque WorldView-2, l'algorithme KPC livre donc de meilleurs résultats à mesure que la résolution spatiale des images augmente.

Enfin, il est intéressant de comparer les temps de calcul nécessaires à chaque algorithme. Pour la mosaïque IKONOS, l'algorithme le plus rapide est la variante SIFT de KPC.

Tableau 4.6 : matrice de confusion de la classification de la mosaïque IKONOS par l'algorithme KPC-SURF.

CLASSIFICATION N	VÉRITÉ-TERRAIN					Erreurs de commission
	Région 1	Région 2	Région 3	Région 4	Région 5	
Classe 1	12 561	0	1 952	928	426	3 306 (21 %)
Classe 2	154	11 669	580	0	0	734 (6 %)
Classe 3	566	484	14 739	1 259	2 883	5 192 (26 %)
Classe 4	393	0	957	11 487	0	1 350 (11 %)
Classe 5	0	1 521	4	0	10 365	1 525 (13 %)
Erreurs d'omission	1 113 (8 %)	2 005 (15 %)	3 493 (19 %)	2 187 (16 %)	3 309 (24 %)	12 107 (17 %)

La matrice de confusion de la meilleure classification (KPC-SURF) est représentée dans le tableau 4.6. On peut y observer que la région la mieux détectée est la région 1 (coin supérieur gauche), malgré sa forte ressemblance visuelle avec les classes 4 et 5 (coins inférieurs gauche et droit).

À l'opposé, la classe 3 (correspondant à la région centrale) est surreprésentée et débordante notamment sur les régions 4 et 5 : cet effet peut être attribué à la mauvaise détection des frontières plutôt qu'à des erreurs de caractérisation des textures.

Enfin, il peut être intéressant d'observer la répartition des pixels « ambigus » : on considérera qu'un pixel est ambigu lorsque la distance entre le vecteur descripteur du pixel et le deuxième plus proche centroïde est égale à au moins 90 % de la distance entre ce même vecteur et le centroïde le plus proche (qui est par définition le centroïde de la classe attribuée). Autrement dit, un pixel ambigu est presque à mi-distance entre deux classes différentes. Un grand nombre de pixels ambigus témoigne donc d'une faible différenciation entre les partitions de descripteurs texturaux.

La figure 4.6 (b) représente la répartition de ces pixels ambigus. On constate qu'ils sont logiquement situés au niveau des frontières entre textures, mais également le long de certains bords de l'image, témoignant ainsi de la présence d'effets de bord difficiles à maîtriser et nuisant à la classification de l'image.

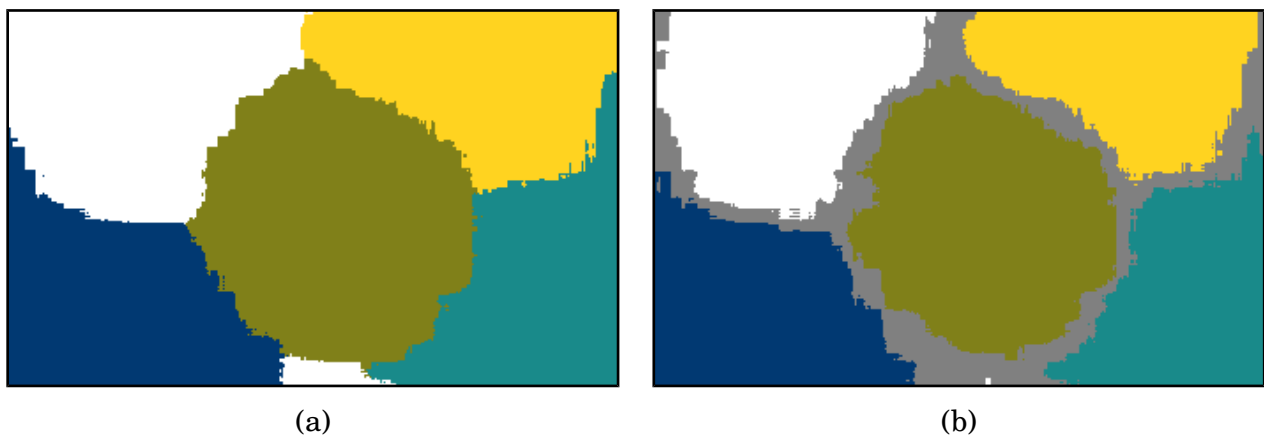


Figure 4.6 : classification KPC-SURF de la mosaïque IKONOS : classification normale (a) et classification avec pixels ambigus (en gris) (b)

Dans le cas de la classification KPC-SURF, les pixels ambigus représentent 15 % de tous les pixels de l'image. Cette proportion est élevée : elle est de 11 % pour KPC-SIFT et inférieure à 8 % pour les autres algorithmes. C'est une situation paradoxale : on constate que les algorithmes menant aux meilleurs TBC sont aussi ceux faisant apparaître le plus de pixels ambigus. On peut tenter de l'expliquer par la grande dispersion des vecteurs descripteurs : bien que les partitions soient correctement identifiées (et donc les textures bien discriminées), les vecteurs descripteurs sont trop dispersés pour assurer une séparabilité satisfaisante des partitions.

b. Classifications dirigées

Il a été mentionné dans le chapitre portant sur la méthodologie que deux méthodes de partitionnement dirigé des descripteurs texturaux ont été testées : il s'agit de *k-means* et de *k-NN*. La figure 4.7 résume les résultats obtenus avec ces trois méthodes. On peut y observer que parmi les deux approches dirigées, *k-NN* fournit les meilleurs résultats. C'est donc sur cette approche que nous nous concentrerons dans la suite de cette section.

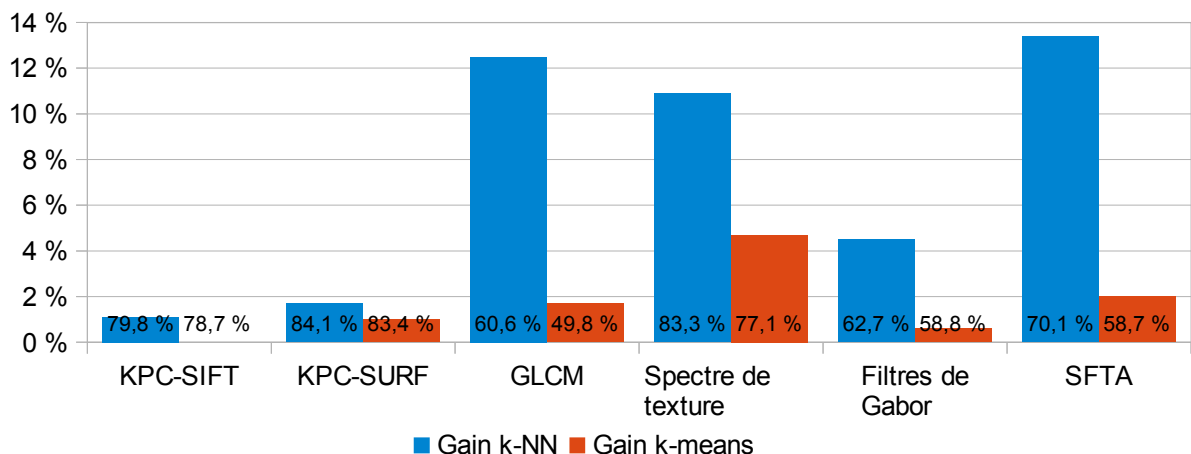


Figure 4.7 : différences de TBC entre classifications dirigées (*k-NN* en bleu, *k-means* en rouge) et non dirigées (*k-means*). Les pourcentages représentent les TBC des classifications dirigées.

Un seul pixel d'entraînement a été sélectionné pour chaque texture : il s'agit à chaque fois du pixel le plus éloigné possible des autres textures.

Le tableau 4.7 détaille les résultats généraux obtenus pour la classification dirigée des descripteurs texturaux. L'initialisation (*seeding*) n'étant plus aléatoire (en raison de l'approche dirigée), les résultats des classifications ne présentent plus de variabilité. Les TBC et coefficients kappa sont donc déterministes, d'où l'absence d'écart-types dans le tableau 4.7.

Tableau 4.7 : taux de bonne classification, coefficients kappa et durées d'exécution de la classification dirigée (*k-NN*) de la mosaïque IKONOS. Les meilleures valeurs sont indiquées en gras.

		TBC (%)	κ	Durée moy. (s)
KPC	SIFT	80	0,75	172
	SURF	84	0,80	695
GLCM		61	0,50	695
Spectre de texture		83	0,79	102
Filtres de Gabor		63	0,53	1 107
SFTA		70	0,62	-

Encore une fois, c'est la variante SURF de l'algorithme KPC qui produit les meilleurs résultats. Les gains de TBC par rapport aux algorithmes concurrents sont moins élevés que dans l'approche non dirigée, en raison des bonnes performances de l'algorithme de spectre de texture. Ainsi, KPC-SURF apporte un léger gain de seulement 1 % par rapport à ce dernier algorithme, alors que ce gain était de 10 % en mode non dirigé.

Les figures 4.8 (b), (e) et (f) sont donc assez similaires : on y observe une classification approximativement correcte, avec peu de confusion interclasse. Les frontières paraissent un peu plus régulières que dans la classification non dirigée. Globalement, les

mêmes constats qu'en mode non dirigé peuvent donc être faits : bonne discrimination texturale, mais qualité moyenne des frontières.

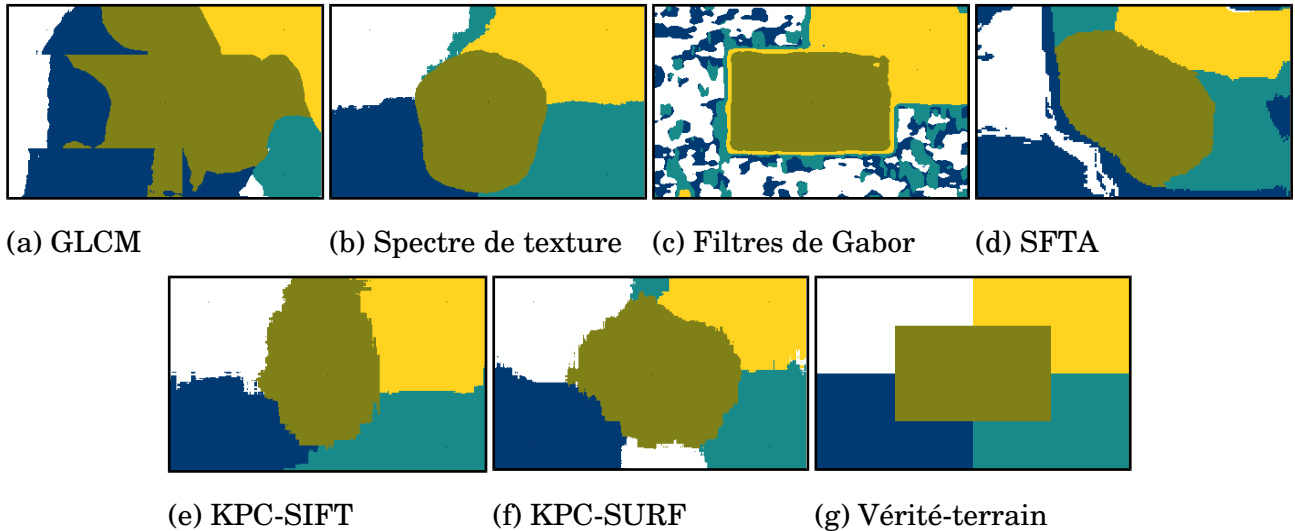


Figure 4.8 : résultats de la classification dirigée de la mosaïque IKONOS

On peut se poser la question de savoir pourquoi les deux variantes de KPC ne bénéficient que d'un gain minime lors du passage du mode non dirigé au mode dirigé (comme le montre la figure 4.7), alors que les autres algorithmes voient leurs TBC augmenter de 4 à 13 %. Cela pourrait s'expliquer par la structure des partitions de descripteurs texturaux de l'algorithme KPC : ces descripteurs étant plus dispersés que ceux des autres algorithmes (en raison de la dimensionnalité de l'espace textural), il est possible qu'ils soient plus facilement classifiables à l'aide de *k-means* (dirigé ou non). À l'inverse, *k-NN* pourrait davantage profiter aux autres algorithmes, dont les descripteurs sont plus concentrés. Vu sous cet angle, on peut considérer que le supplément d'informations apporté par les pixels d'entraînement est peu utile à l'algorithme KPC puisque la structure de son espace textural encourage le partitionnement non dirigé. D'une certaine manière, KPC est donc optimisé pour la classification non dirigée.

Enfin, on peut également se demander pourquoi l'algorithme *k-NN* produit de bien meilleurs résultats que l'algorithme *k-means* dirigé, comme le montre aussi la

figure 4.7. La réponse est simple : dans la mesure où *k-means* est un algorithme itératif qui ne s'achève qu'une fois un état d'équilibre trouvé, l'initialisation des centroïdes – aléatoire ou prédéfinie – joue un rôle relativement mineur, et ce d'autant plus lorsque les partitions sont faiblement différenciées. Ainsi, il est naturel que les classifications *k-means* dirigées et non dirigées fournissent un résultat proche. Inversement, *k-NN* repose exclusivement sur le choix des échantillons d'entraînement. Suivant la qualité de ce choix, le résultat final peut être très bon ou très mauvais; il peut surpasser celui de *k-means* si les échantillons d'entraînement sont représentatifs de leurs classes respectives. Comme c'est le cas ici, il est compréhensible que *k-NN* fournisse de meilleurs résultats que *k-means*.

Temps de calcul

Il est intéressant de se pencher sur les différences de durées d'exécution entre les variantes SIFT et SURF, mais aussi entre les modes dirigés et non dirigés. La figure 4.9 permet de mieux comprendre ces écarts de temps de calcul en mettant en lumière les principales phases de l'algorithme KPC. Les durées totales d'exécution qui y figurent peuvent légèrement varier par rapport aux durées répertoriées dans les tableaux 4.5 et 4.7, car certains paramètres ont été normalisés afin que les valeurs affichées dans la figure 4.9 soient plus facilement comparables entre elles. En particulier, la taille de voisinage est la même pour les quatre types de classification (131 px).

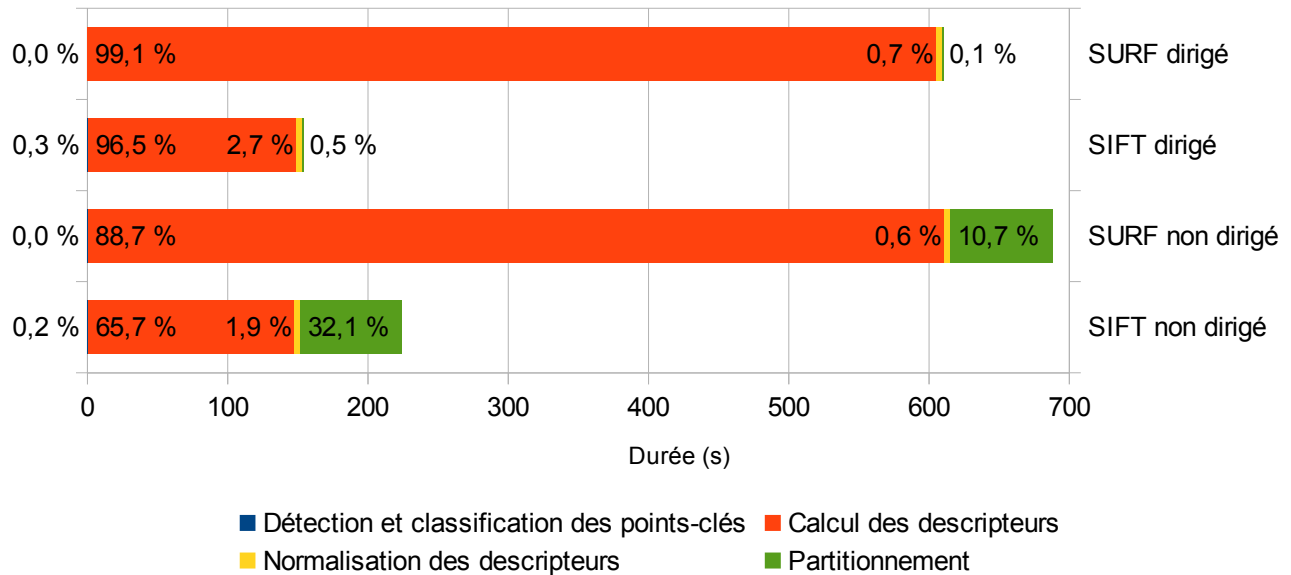


Figure 4.9 : comparaison des temps de calcul de chaque étape entre les modes dirigés (k -NN) et non dirigés (k -means), pour les variantes SIFT et SURF

Plusieurs constats peuvent être faits. D’abord, la première étape (détection et classification des points-clés) représente un coût négligeable en comparaison de la durée totale de l’algorithme. Ce constat est important car le fait que la variante SURF de KPC soit beaucoup plus lente que la variante SIFT n’est pas intuitif : en effet, SURF a justement été conçu (en partie) dans le but d’être plus rapide que SIFT. L’analyse des temps de calcul le confirme : cette première étape est environ deux fois plus rapide pour SURF que pour SIFT (en moyenne 240 contre 510 ms). Seulement, ces durées sont minimes et n’influent donc en rien la durée totale de l’algorithme (d’autant plus que les implémentations GPU souvent disponibles accélèrent grandement cette étape).

Deuxièmement, la comparaison dirigé / non dirigé montre que la différence de durée d’exécution entre ces deux modes est seulement due à l’étape finale de partitionnement, ce qui est prévisible. On voit ainsi que k -means (non dirigé, mais la version dirigée produit les mêmes résultats) est environ 100 fois plus lente que k -NN : en moyenne 73 s contre 820 ms. Cela met en évidence la difficulté pour k -means de converger vers

un résultat stable. De plus, on constate aussi que même en utilisant *k-means*, la phase de partitionnement n'est pas à l'origine de la majeure partie de la durée d'exécution (au maximum 32 %, dans un cas extrême).

Enfin, la comparaison SIFT / SURF met en évidence la cause principale de la différence de temps de calcul entre ces deux variantes. Il est en effet clair que c'est la phase de calcul des descripteurs texturaux qui constitue la plus grande part du temps d'exécution. Les descripteurs sont basés sur le calcul de statistiques spatiales de second ordre entre les points-clés, c'est-à-dire sur l'étude de la répartition des distances entre chaque paire de points-clés. Le temps de calcul des descripteurs est donc approximativement proportionnel au carré de nombre de points-clés détectés; or SURF détecte environ le double de points-clés que SIFT (655 contre 275). On comprend donc aisément pourquoi cette étape dure ~600 s pour SURF et ~150 s pour SIFT.

Ainsi, il apparaît que les pistes d'optimisation du temps de calcul de l'algorithme KPC sont davantage à chercher du côté du calcul des statistiques spatiales plutôt que du partitionnement des descripteurs texturaux.

4.1.2.2. Mosaïque WorldView-2

Le tableau 4.8 résume les résultats obtenus pour la classification non dirigée de la mosaïque WorldView-2. Pour cette image, c'est la variante SIFT de KPC qui produit le meilleur résultat. Les différences de performance des algorithmes testés sont encore plus marquées que dans la section précédente : les variantes de KPC surpassent de 11 à 13 % le meilleur des algorithmes concurrents. Ce gain était de 6 à 12 % dans le cas de la mosaïque IKONOS.

Tableau 4.8 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la mosaïque WorldView-2. Les meilleures valeurs sont indiquées en gras.

		TBC (%)	σ_{TBC} (%)	κ (%)	Durée moy. (s)
KPC	SIFT	86	0,1	0,83	9 940
	SURF	85	1,3	0,82	36 180
GLCM		72	0,7	0,67	530
Spectre de texture		74	0,0	0,68	3 420
Filtres de Gabor		64	0,0	0,57	19 060
SFTA		65	0,0	0,58	-

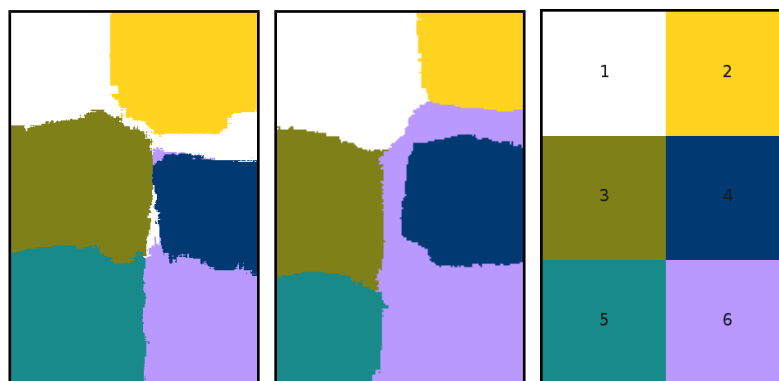


(a) GLCM

(b) Spectre de texture

(c) Filtres de Gabor

(d) SFTA



(e) KPC-SIFT

(f) KPC-SURF

(g) Vérité-terrain

Figure 4.10 : résultats de la classification non dirigée de la mosaïque WorldView-2

Cette amélioration des performances peut être expliquée de deux manières. Premièrement, la résolution de la mosaïque WorldView-2 est supérieure à celle de la mosaïque IKONOS : 52 cm/px contre 82 cm/px. Le nombre accru de détails a un effet direct sur le nombre et la densité des points-clés détectés. Comme l'indique le tableau 4.9, cette densité augmente significativement dans le cas de la mosaïque WorldView-2. Cela permet à l'algorithme KPC d'améliorer la caractérisation des textures et donc la qualité finale de la classification.

Deuxièmement, la géométrie de la mosaïque WorldView-2 est davantage favorable à la classification : les frontières entre textures sont légèrement plus simples, et surtout les morceaux de textures sont bien plus grands que dans l'image IKONOS : ces derniers font en effet 300 pixels de côté. La taille des voisinages utilisés pour classifier les textures est ainsi sensiblement plus faible que celle des morceaux de texture, permettant une meilleure détection des frontières. Cela rejoint l'idée développée dans la section a : plus la résolution spatiale de l'image source est haute, plus le potentiel de KPC est élevé.

Tableau 4.9 : densité de points-clés détectés, par image et par algorithme de détection (en nombre de points-clés par 1 000 pixels).

	SIFT	SURF
Mosaïque IKONOS	3,8	9,0
Placette IKONOS	5,2	9,8
Mosaïque WorldView-2	6,9	12,8

La matrice de confusion de la classification par KPC-SIFT (figure 4.10 (e)) est représentée dans le tableau 4.10. Les classes avec les plus faibles erreurs de commission sont les classes 4 et 6; elles correspondent aux textures les plus régulières (champs de maïs et plantation de sapins) et donc les plus simples à classifier. Contrairement aux

algorithmes de classification concurrents, les classes 1 et 2 sont bien différenciées et présentent peu d'erreurs de commission l'une par rapport à l'autre. Enfin, il est intéressant de remarquer qu'il y a assez peu de confusion entre les classes 2, 3 et 5 qui représentent toutes des forêts matures et denses.

Tableau 4.10 : matrice de confusion de la classification de la mosaïque WorldView-2 par l'algorithme KPC-SIFT. Un pixel sur deux a été classifié dans chaque direction (25 % de tous les pixels).

CLASSIFICATION	VÉRITÉ-TERRAIN						Erreurs de commission (%)
	Région 1	Région 2	Région 3	Région 4	Région 5	Région 6	
Classe 1	16 348	1 280	0	4 084	0	0	5 364 (25 %)
Classe 2	3 282	20 843	0	5	0	0	3 287 (14 %)
Classe 3	2 570	77	20 999	2 907	147	0	5 701 (21 %)
Classe 4	0	0	0	14 690	0	900	900 (6 %)
Classe 5	0	0	1 501	81	22 203	1 906	3 488 (14 %)
Classe 6	0	0	0	733	0	19 544	733 (4 %)
Erreurs d'omission (%)	5 852 (26 %)	1 357 (6 %)	1 501 (7 %)	7 801 (35 %)	147 (1 %)	2 806 (13 %)	19 473 (15 %)

La figure 4.11 indique la répartition des pixels ambigus (voir section a). Cette fois-ci, les pixels ambigus représentent 12 % (16 % pour la variante SURF) de l'ensemble des pixels classifiés, ce qui est supérieur à la proportion de pixels ambigus pour la mosaïque IKONOS. On peut observer qu'un grand nombre de pixels attribués par erreur à la classe 1 (en blanc) dans la région 4 ne sont pas considérés ambigus. Cela trahit un problème particulier de caractérisation de la classe 1, qui peut être dû à son caractère irrégulier et hétérogène.

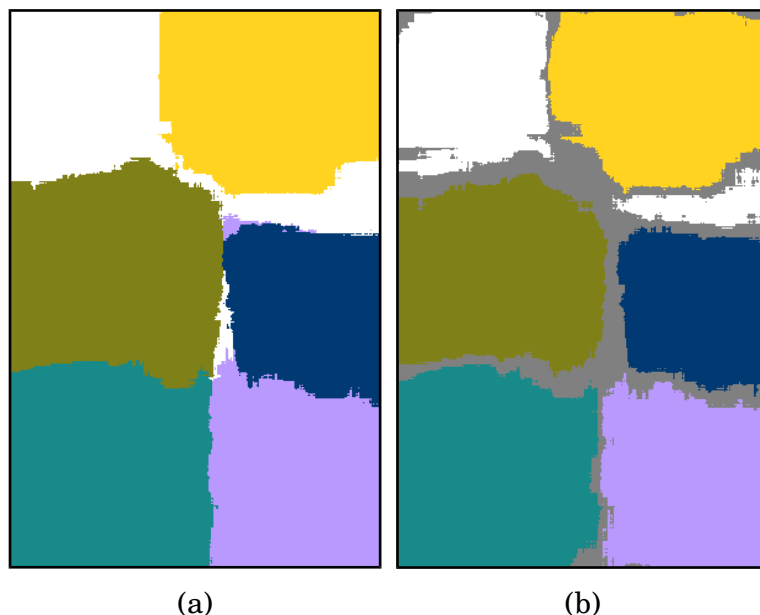


Figure 4.11 : classification KPC-SIFT de la mosaïque WorldView-2: classification normale (a) et classification avec pixels ambigus (en gris) (b)

4.1.2.3. *Placette IKONOS*

La dernière image testée est la placette IKONOS. Le tableau 4.11 et la figure 4.12 résumement les résultats des classifications de cette image.

Ces résultats sont bien plus mitigés que pour les images précédentes. Bien que KPC-SIFT parvienne à décrocher le plus haut TBC et KPC-SURF le coefficient kappa le plus élevé, les résultats de KPC ne sont pas significativement meilleurs que ceux des algorithmes concurrents, compte tenu des écarts-types élevés.

Les hautes valeurs des écarts-types peuvent être attribués à la complexité de l'image source : à cause du fait que les segments sont beaucoup plus petits et irréguliers que dans les images précédentes, le phénomène de mélange de textures à l'intérieur des voisinages est bien plus présent. Cela a pour effet d'empêcher l'apparition de partitions facilement détectables dans l'espace textural, créant ainsi une plus forte instabilité lors du partitionnement par *k-means*.

Tableau 4.11 : taux de bonne classification moyens, écarts-types du TBC, coefficients kappa moyens et durées moyennes d'exécution de la classification non dirigée de la placette IKONOS. Les meilleures valeurs sont indiquées en gras.

		TBC (%)	σ_{TBC} (%)	κ (%)	Durée moy. (s)
KPC	SIFT	39	3,0	0,22	538
	SURF	38	2,0	0,24	2 959
GLCM		37	0,4	0,21	338
Spectre de texture		36	3,9	0,18	1 230
Filtres de Gabor		38	1,4	0,24	743
SFTA		34	1,5	0,15	-

De plus, la figure 4.12 met en évidence la principale faiblesse de l'algorithme KPC, qui est la détection de frontières complexes. La taille des segments est ici de l'ordre de quelques dizaines de pixels, alors que les voisinages utilisés font environ une centaine de pixels de côté. La bonne capacité de KPC à discriminer les textures ne parvient pas à compenser la mauvaise détection de frontières, contrairement aux images précédentes. Pour cette raison, les performances de KPC ne dépassent pas significativement celles des autres algorithmes.

Il est cohérent de constater que la proportion de pixels ambigus est plus élevée que jamais (figure 4.13) : pour la classification KPC-SIFT, elle atteint 26 %. Cela traduit la faible différenciation des descripteurs dans l'espace textural et la difficulté à classifier les mélanges de texture.

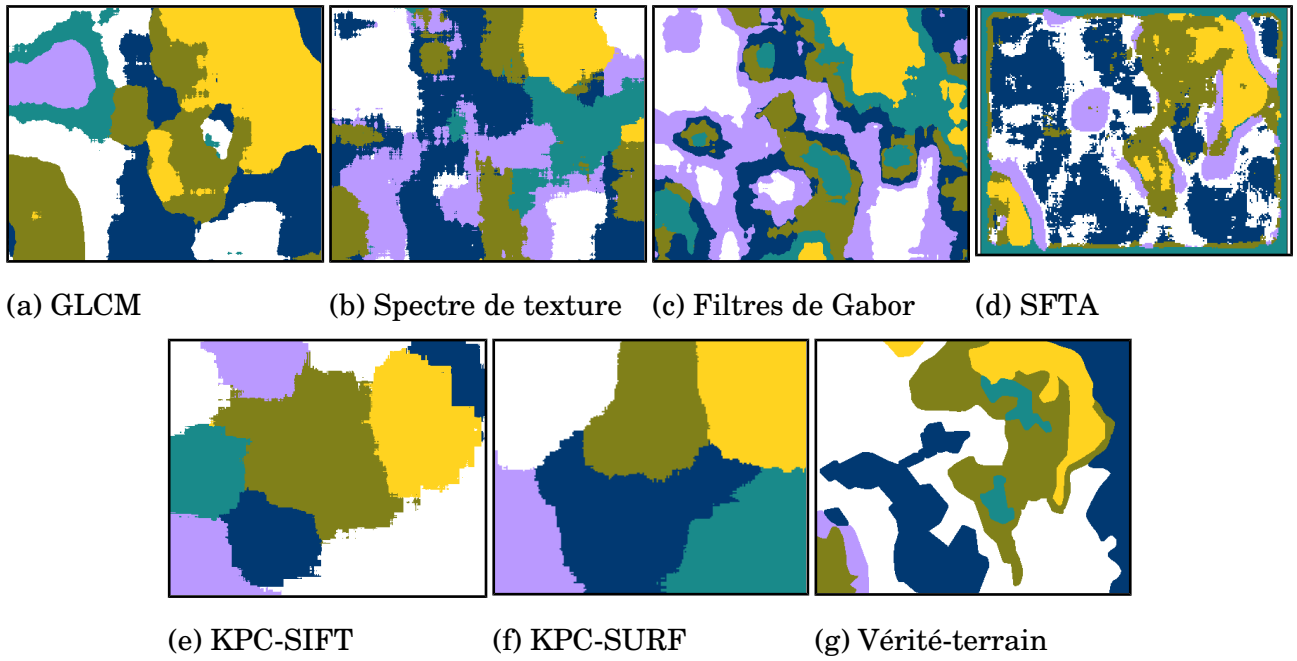


Figure 4.12 : résultats de la classification non dirigée de la placette IKONOS

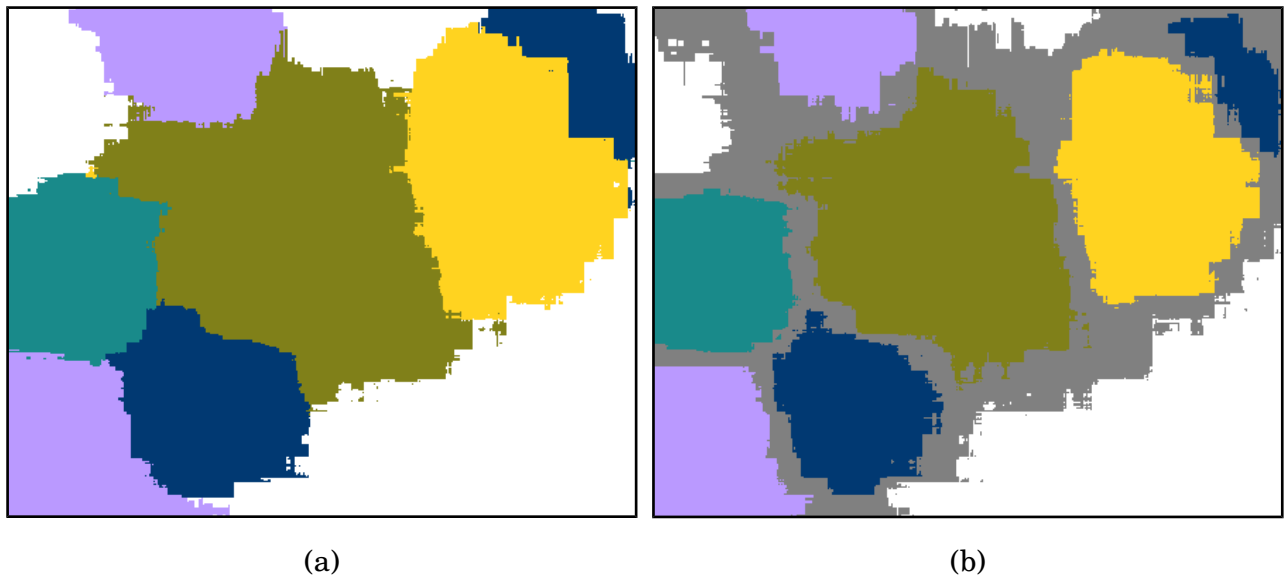


Figure 4.13 : classification KPC-SIFT de la placette IKONOS : classification normale (a) et classification avec pixels ambigus (en gris) (b)

4.2. Fusion d'images

La section 4.1 a mis en évidence les bonnes performances de l'algorithme KPC, essentiellement dues à sa capacité à discriminer des textures similaires. En revanche, la mauvaise détection des frontières entre textures constitue la principale faiblesse de cet algorithme. Pour pallier ce problème, nous nous sommes orientés vers une approche multi-échelle.

La section 4.2.1 résume les résultats généraux obtenus lors de la fusion multi-échelle des trois images test. La section 4.2.2 vise à étudier le cas spécifique de la fusion multi-échelle d'une image dégradée par un bruit important, afin de mieux évaluer le potentiel d'une telle approche. Enfin, la section 4.2.3 analyse dans quelle mesure le concept de fiabilité des sources permet d'améliorer le résultat final.

4.2.1. Résultats généraux

Les tableaux 4.12, 4.13 et 4.14 résument les résultats obtenus pour la fusion de trois échelles de chacune des images de test (respectivement la mosaïque IKONOS, la mosaïque WorldView-2 et la placette IKONOS). Les valeurs affichées dans ces tableaux correspondent au ΔTBC , qui est la différence entre le TBC de la classification fusionnée et le TBC de la classification source en résolution de base. Par exemple, si les trois échelles de résolution d'une même image ont des TBC de 48, 42 et 43 % et que la fusion résultant de ces trois échelles a un TBC de 50 %, alors ΔTBC vaut 2 %.

Les trois échelles correspondent aux résolutions suivantes : 100 %, 50 % et 25 % (résolution de base et résolution divisée par 2 et 4). Dans cette section, le coefficient de fiabilité est ignoré. Les classifications source sont toutes issues de KPC-SIFT. Les résultats complets peuvent être trouvés dans l'annexe 2.

La figure 4.14 représente, à titre d'exemple, les sources multi-échelle correspondant à la mosaïque IKONOS :

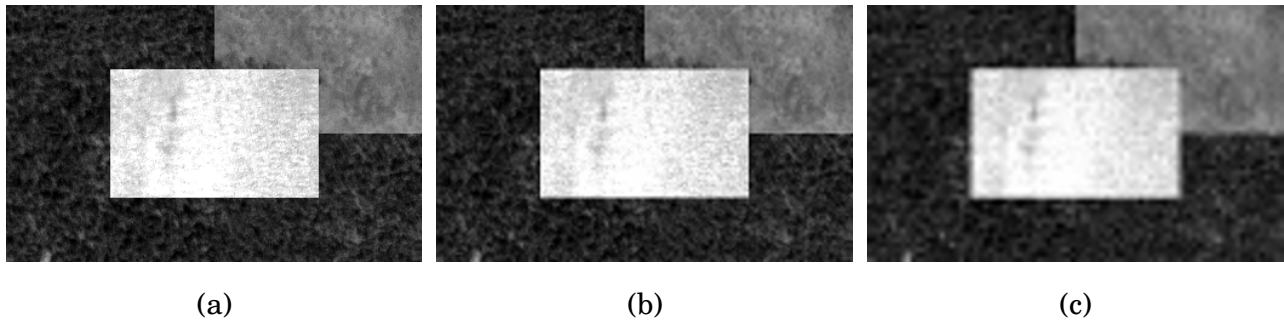


Figure 4.14 : mosaïque IKONOS : résolutions 100 % (a), 50 % (b) et 25 % (c)

Tableau 4.12 : Δ TBC (%) issus de la fusion multi-échelle de la mosaïque IKONOS, sans coefficient de fiabilité. Les meilleures valeurs sont en gras.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	-2	-2	-2
	Robuste	-2	-3	-3
	Disjonctive	-2	-3	-3
Modèle de distance composé	Conjonctive	-2	-2	-2
	Robuste	-2	-2	-2
	Disjonctive	-2	-3	-3

Tableau 4.13 : Δ TBC (%) issus de la fusion à multi-échelle de la mosaïque WorldView-2, sans coefficient de fiabilité. Les meilleures valeurs sont en gras.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	-10	-10	-10
	Robuste	-10	-8	-8
	Disjonctive	-10	-8	-8
Modèle de distance composé	Conjonctive	-10	-10	-10
	Robuste	-10	-9	-9
	Disjonctive	-10	-11	-10

Tableau 4.14 : Δ TBC (%) issus de la fusion à multi-échelle de la placette IKONOS, sans coefficient de fiabilité. La meilleure valeur est en gras.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	+7	+6	+6
	Robuste	+7	-1	-3
	Disjonctive	+7	-1	-3
Modèle de distance composé	Conjonctive	+7	+7	+7
	Robuste	+7	-0	+1
	Disjonctive	+7	-2	-2

Plusieurs conclusions peuvent être tirées de ces résultats. Commençons par l'étude des étapes du processus de fusion :

- Initialisation : parmi les deux approches testées (modèles de distance simple et composé), le modèle simple ressort clairement. Il est possible que le modèle com-

posé, dont le but est d'établir les masses de paires d'hypothèses au lieu des hypothèses simples, apporte trop d'ambiguïté dans la modélisation des hypothèses.

- Combinaison : dans la majorité des cas, la combinaison conjonctive mène aux meilleurs résultats. Contrairement à ce qui était attendu, la combinaison robuste ne constitue pas systématiquement un bon compromis entre fusions conjonctive et disjonctive. En effet, le conflit entre les classifications sources étant souvent élevé, la combinaison robuste aboutit souvent aux mêmes résultats que la combinaison disjonctive.
- Décision : bien que la décision par maximum de crédibilité semble être en général la plus appropriée, les résultats sont assez partagés. Il n'est pas possible de dégager une règle de décision qui serait systématiquement meilleure que les autres. Toutefois, les écarts de résultats d'une règle de décision à une autre sont la plupart du temps assez faibles.

Le second constat majeur que l'on peut dresser est que la fusion multi-échelle « simple » (c'est-à-dire sans paramètre de fiabilité) n'améliore pas la qualité de la classification pour toutes les images. En particulier, cette approche ne donne pas de bons résultats pour les deux mosaïques, avec des ΔTBC de -2 et -8 %. En revanche, le gain de TBC obtenu par la fusion multi-échelle de la placette IKONOS est remarquable : $+7$ %.

Le tableau 4.15 met en parallèle, pour chaque image, le TBC de la classification SIFT non dirigée (résolution native), noté $\text{TBC}_{\text{source}}$, et le gain maximum de TBC à l'issue de la fusion. Bien que le nombre d'images soit trop faible pour établir une corrélation significative entre ces deux variables, il est clairement visible que le gain de TBC est d'autant plus élevé que $\text{TBC}_{\text{source}}$ est faible.

Tableau 4.15 : TBC_{source} (classification KPC-SIFT non dirigée) et ΔTBC pour chaque image (en %).

Image	TBC_{source}	ΔTBC
Mosaïque IKONOS	79	-2
Mosaïque WorldView-2	86	-8
Placette IKONOS	41	+7

Ce phénomène peut s'expliquer par le raisonnement suivant : lorsqu'une classification est de très bonne qualité en résolution native, il est probable que les classifications de la même image en résolution plus faible seront de moins bonne qualité (voir le tableau 6.1 de l'annexe 2 pour illustration). La fusion multi-échelle aura donc pour effet de dégrader la classification originale. Inversement, si la classification en résolution native est déjà de faible qualité, les classifications aux échelles inférieures ne seront probablement pas sensiblement différentes et la fusion pourra jouer son rôle efficacement.

Ainsi, si l'on prend comme référence pour le calcul de ΔTBC le TBC_{moyen} (parmi les trois échelles) au lieu du *meilleur* TBC (échelle de base), alors ΔTBC vaut +10 % pour la mosaïque WorldView-2 au lieu de -8 %.

On peut ainsi en tirer une conclusion importante : l'approche de fusion multi-échelle employée ne présente d'intérêt que pour les images complexes et difficiles à classifier. Bien entendu, la question suivante se pose alors : où se situe le seuil de complexité? Partant du principe que l'on travaille en mode non dirigé, comment savoir si la fusion multi-échelle risque ou non de dégrader la classification initiale? Il n'y a pas de réponse simple à ce problème; la solution la plus directe consiste à effectuer une comparaison visuelle avant/après fusion afin de conserver le résultat qui paraît le plus satisfaisant.

Les figures 4.16, 4.17 et 4.18 permettent de se faire une idée des résultats des fusions multi-échelle. Pour chaque image source sont représentées :

1. la classification KPC-SIFT de l'image source en résolution originale (les classifications des autres résolutions sont répertoriées dans l'annexe 2, figure 6.2).
2. le résultat de la fusion multi-échelle (avec 3 résolutions, diminuées d'un facteur 2 pour chaque échelle), comprenant uniquement des hypothèses simples.
3. le même résultat que précédemment, en faisant également apparaître les hypothèses composées (seules les hypothèses duales sont représentées pour des raisons de lisibilité). La crédibilité / plausibilité des hypothèses composées étant toujours supérieure ou égale à celle des hypothèses simples, il a fallu choisir une règle permettant de conserver ces dernières afin que les hypothèses composées ne représentent que les situations d'incertitude et/ou d'imprécision. La règle retenue est la suivante : pour chaque pixel, une hypothèse simple est privilégiée par rapport aux hypothèses composées si sa crédibilité (ou sa plausibilité) est supérieure à un certain seuil, généralement égal à 0,8 mais susceptible de varier d'une image à une autre. Notons que pour cette raison, la proportion de pixels dont l'hypothèse est composée n'est pas indicative de l'incertitude / imprécision générale de la classification, mais résulte uniquement d'un seuillage arbitraire.
4. la répartition du conflit, pixel par pixel, entre les images sources.

La légende est la suivante :

 θ_1	 θ_4	 $\theta_1 \cup \theta_2$	 $\theta_1 \cup \theta_5$	 $\theta_2 \cup \theta_4$	 $\theta_3 \cup \theta_4$	 $\theta_4 \cup \theta_5$
 θ_2	 θ_5	 $\theta_1 \cup \theta_3$	 $\theta_1 \cup \theta_6$	 $\theta_2 \cup \theta_5$	 $\theta_3 \cup \theta_5$	 $\theta_4 \cup \theta_6$
 θ_3	 θ_6	 $\theta_1 \cup \theta_4$	 $\theta_2 \cup \theta_3$	 $\theta_2 \cup \theta_6$	 $\theta_3 \cup \theta_6$	 $\theta_5 \cup \theta_6$

Figure 4.15 : légende des hypothèses simples et composées

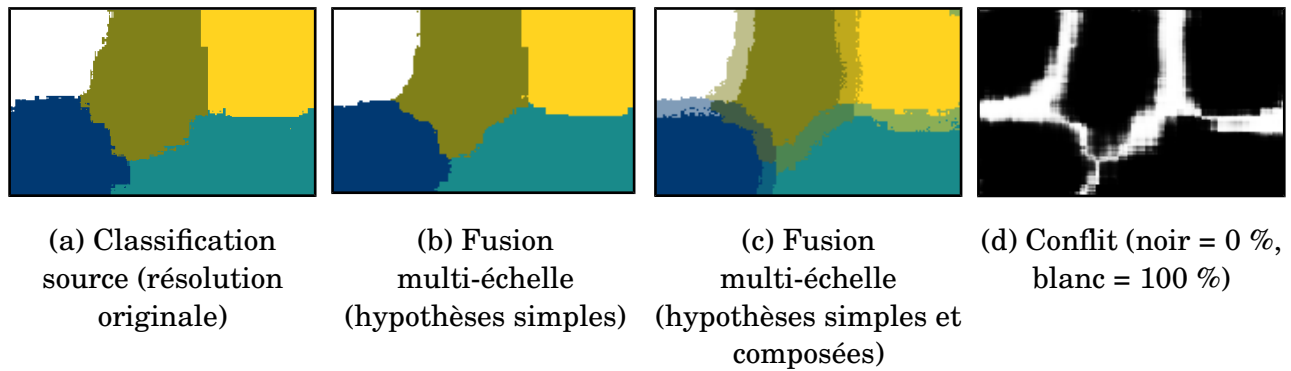


Figure 4.16 : mosaïque IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison conjonctive, décision par maximum de crédibilité) avec hypothèses simples (b) et composées (c), répartition du conflit (d)

La figure 4.16, qui représente la mosaïque IKONOS, ne présente pas de résultats surprenants. Le résultat de la fusion multi-échelle est très proche de la classification source : la fusion s’est donc montrée plutôt robuste dans ce cas. L’affichage des hypothèses composées témoigne simplement de l’incertitude détectée par la théorie de l’évidence au niveau des frontières entre textures, ce qui est cohérent.

La mosaïque WorldView-2 est celle qui a le plus souffert de la fusion multi-échelle (figure 4.17), principalement à cause de la mauvaise détection des textures 4 et 5 (bleu foncé et turquoise) au niveau des échelles inférieures. Cela se reflète d’ailleurs au niveau de la carte du conflit, qui n’est pas limité aux frontières entre textures. La figure 4.17 (c) montre toutefois que la prise en compte des hypothèses composées permet de délimiter plus clairement les principaux segments. Le fort conflit au niveau de la texture 4 peut s’expliquer par le fait qu’il s’agit d’une texture très uniforme (champ de maïs) qui présente peu de traits caractéristiques. Un important phénomène de mélange de classes peut donc avoir lieu au niveau des frontières avec les autres textures, particulièrement à faible résolution. Cela met en évidence une limite de l’approche multi-échelle.

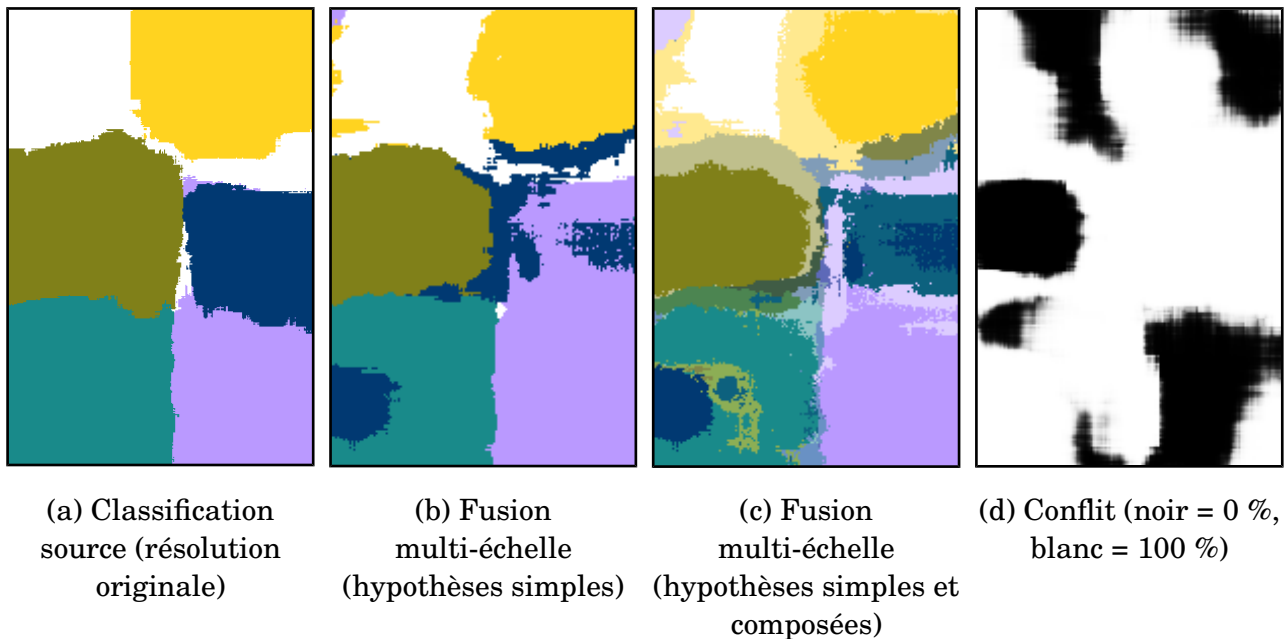


Figure 4.17 : mosaïque WorldView-2: classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison robuste, décision par maximum de plausibilité) avec hypothèses simples (b) et composées (c), répartition du conflit (d)

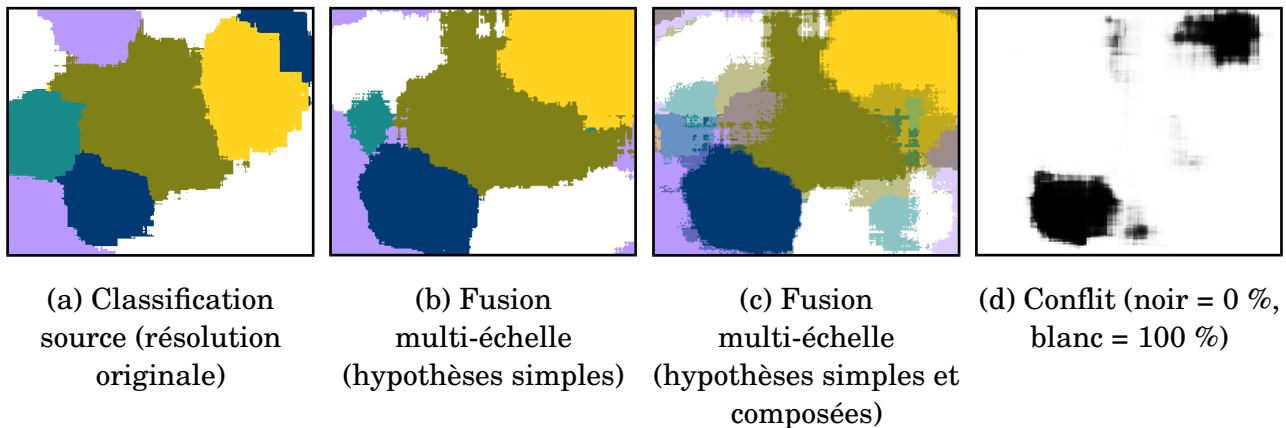


Figure 4.18 : placette IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle (modèle de distance simple, combinaison conjonctive, décision par maximum de crédibilité) avec hypothèses simples (b) et composées (c), répartition du conflit (d)

La figure 4.18 est la plus difficile à analyser en raison de la complexité des frontières

de la placette IKONOS. L'amélioration de 7 % du TBC entre les figures 4.18 (a) et (b) n'est pas évidente à constater (cf vérité-terrain en figure 4.12 (g)). La carte des conflits montre que seules, deux zones sont correctement classifiées à toutes les échelles, correspondant approximativement aux segments jaune et bleu. La complexité des frontières est à l'origine de conflits dans la majeure partie de l'image. Le conflit moyen est d'ailleurs de 88 %, ce qui est une valeur remarquablement élevée : par comparaison, il est de seulement 19 % pour la mosaïque IKONOS.

4.2.2. Images bruitées

Afin de mieux étudier le lien entre la qualité des images sources et le gain de TBC apporté par la fusion multi-échelle, il est nécessaire de disposer d'images supplémentaires de qualité variée. Pour ce faire, divers niveaux de bruit blanc gaussien ont été ajoutés aux images de test. Au total, 12 images ont ainsi été construites, auxquelles s'ajoutent les trois images de test non artificiellement bruitées. Chacune de ces images a fait l'objet de la même fusion multi-échelle que dans la section précédente (classification KPC-SIFT de trois échelles). Les résultats sont récapitulés dans le tableau 4.16 et la figure 4.19.

Tableau 4.16 : Δ TBC et TBC_{source} (entre parenthèses) pour chaque image (en %).

Image	Écart-type du bruit (en niveaux de gris)					Δ TBC moyen
	0	20	40	60	80	
Mosaïque IKONOS	-2 (79)	+13 ⁸ (52)	+3 (70)	-3 (66)	-0 (59)	-1
Mosaïque WorldView-2	-8 (86)	+9 (68)	+5 (67)	-4 (84)	+18 ⁸ (62)	+1
Placette IKONOS	+7 (41)	+9 (32)	+3 (38)	+4 (36)	+0 (48)	+5

8 Ces deux valeurs extrêmes (Δ TBC = +18 % et +13 %) ont été exclues lors de l'analyse des données (calcul du taux de corrélation et des moyennes). Elles sont dues à des écarts anormaux entre les TBC obtenus aux différentes échelles de classification.

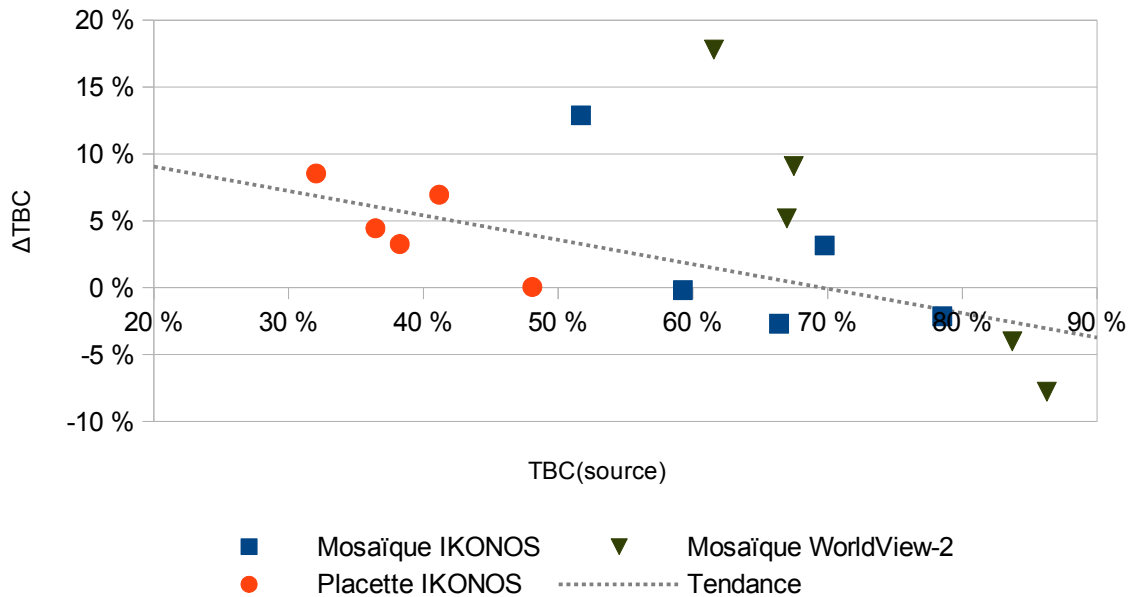


Figure 4.19 : ΔTBC en fonction de TBC_{source} pour chaque image

Ces résultats permettent de mettre en évidence une corrélation de $-0,66$ entre la qualité des classifications source et le gain de TBC apporté par la fusion multi-échelle. Cela confirme la tendance observée dans la section précédente : plus une classification est de mauvaise qualité (en raison de la complexité de l'image source), plus la fusion multi-échelle apporte un gain important.

L'approximation linéaire de la tendance permet d'estimer que le TBC_{source} représentant la limite entre ΔTBC positif et négatif est d'environ 70 % (bien qu'il soit visible que des gains peuvent être obtenus au-delà de cette valeur). Autrement dit, si le TBC de la classification initiale est inférieur à 70 %, il est probable que la fusion multi-échelle aura pour conséquence une amélioration de la qualité de la classification, et inversement.

Globalement, la valeur moyenne de ΔTBC est de $+2$ %, avec des valeurs s'échelonnant entre -8 et $+9$ %. Cela montre que, de manière générale, l'hypothèse selon laquelle la fusion multi-échelle permet d'améliorer la qualité de la classification est vérifiée.

Comme indiqué plus haut, il faut toutefois garder à l'esprit que cette amélioration n'est pas systématique et dépend notamment de la qualité de l'image source.

4.2.3. Fiabilité des sources

Le modèle de distance, utilisé pour initialiser les masses des hypothèses pour chaque source, permet l'utilisation d'un paramètre de fiabilité α , compris entre 0 et 1. Ce paramètre représente la confiance que l'on estime pouvoir accorder à chaque source et en pondère les masses lors de la phase de combinaison.

Jusqu'ici, ce paramètre n'était pas pris en compte (sa valeur était systématiquement réglée à 1). Dans cette section sera étudié le gain que ce paramètre peut apporter dans un contexte de fusion multi-échelle.

Comme indiqué dans le chapitre méthodologique, deux méthodes d'estimation de α ont été mises en œuvre : un modèle multi-échelle (α_{ME}) dans lequel intervient un paramètre δ , et un modèle plus générique basé sur l'estimation a priori de la fréquence des classes dans l'image source (α_{DISS}) où intervient un paramètre d . Ces deux modèles ont été appliqués aux trois images de test.

Les estimations *a priori* de la fréquence des classes sur laquelle le modèle α_{DISS} repose sont les suivantes :

- mosaïque IKONOS : estimation idéale correspondant aux fréquences réelles des classes (1/4 pour la classe centrale, 3/16 pour les autres);
- mosaïque WorldView-2 : estimation idéale correspondant aux fréquences réelles des classes (1/6 pour chaque classe);
- placette IKONOS : estimation basée sur un arrondi des fréquences réelles des classes ([0,40; 0,10; 0,20; 0,20; 0,05; 0,05]).

En ce qui concerne les deux premières images de test, on s'est donc placé dans une si-

tuation idéale qui surestimera probablement les performances du modèle α_{DISS} en conditions réelles. L'estimation des fréquences dans le cas de la placette IKONOS vise à rendre les résultats un peu plus réalistes.

Afin de rendre les tests aussi génériques que possible, il a été choisi de rendre les paramètres d et δ constants (bien que les valeurs optimales varient d'une image à l'autre). Leurs valeurs ont ainsi été respectivement fixées à 10 et 0,2.

Tableau 4.17 : ΔTBC par rapport aux classifications sources (en %). Entre parenthèses : gains de TBC apportés par l'utilisation de modèles d'estimation du paramètre de fiabilité α .

Modèle d'estimation d' α	Image		
	Mosaïque IKONOS	Mosaïque WorldView-2	Placette IKONOS
Aucun ($\alpha=1$)	-2 (-)	-8 (-)	+7 (-)
α_{ME} ($\delta=0,2$)	-1 (+2)	+1 (+9)	+5 (-2)
α_{DISS} ($d=10$)	-2 (+1)	+1 (+9)	+4 (-3)

Le tableau 4.17 résume les résultats obtenus en intégrant les modèles d'estimation du paramètre de fiabilité dans l'algorithme de fusion multi-échelle. Les valeurs entre parenthèses correspondent à la différence de TBC entre l'absence et la présence de ces modèles; les autres valeurs correspondent au ΔTBC final (différence entre le TBC de la classification source KPC-SIFT) et le TBC final.

On constate que les modèles d'estimation d' α apportent tous deux un gain de TBC aux images dont le ΔTBC initial (sans fiabilité) était négatif, c'est-à-dire aux deux mosaïques. Dans le cas de la mosaïque WorldView-2, ce gain est suffisamment important pour que le ΔTBC devienne positif, et donc pour que la fusion mène à une réelle amélioration de la classification finale.

Au contraire, l'utilisation du paramètre de fiabilité diminue la qualité de la classification finale dans le cas de la placette IKONOS; le ΔTBC reste cependant positif.

En moyenne, le modèle α_{ME} a apporté un gain de TBC d'environ 2,9 %, contre 2,3 % pour le modèle α_{DISS} . Bien que ces chiffres portent seulement sur trois images, cela laisse penser que le modèle α_{ME} , plus générique, est plus adapté à l'estimation de la fiabilité; d'autant plus que les estimations de fréquence des classes sur lesquelles le modèle α_{DISS} s'est basé étaient, pour deux images, idéales et non approximatives comme ce serait le cas dans une application réelle.

Les classifications finales sont représentées dans les figures 4.20 à 4.22 avec, à titre de comparaison, la classification source (c'est-à-dire sans fusion multi-échelle) (a), et la classification issue de la fusion multi-échelle sans paramètre de fiabilité (b). Enfin, le tableau 4.18 récapitule les gains finaux apportés par la fusion en tenant en compte l'estimation du paramètre de fiabilité.

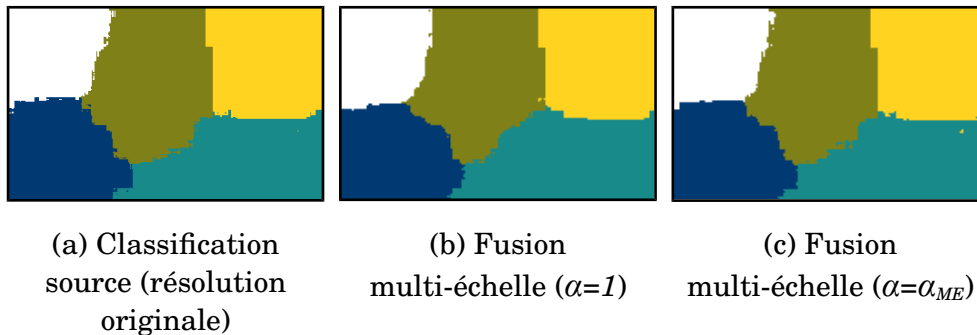


Figure 4.20 : mosaïque IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c)

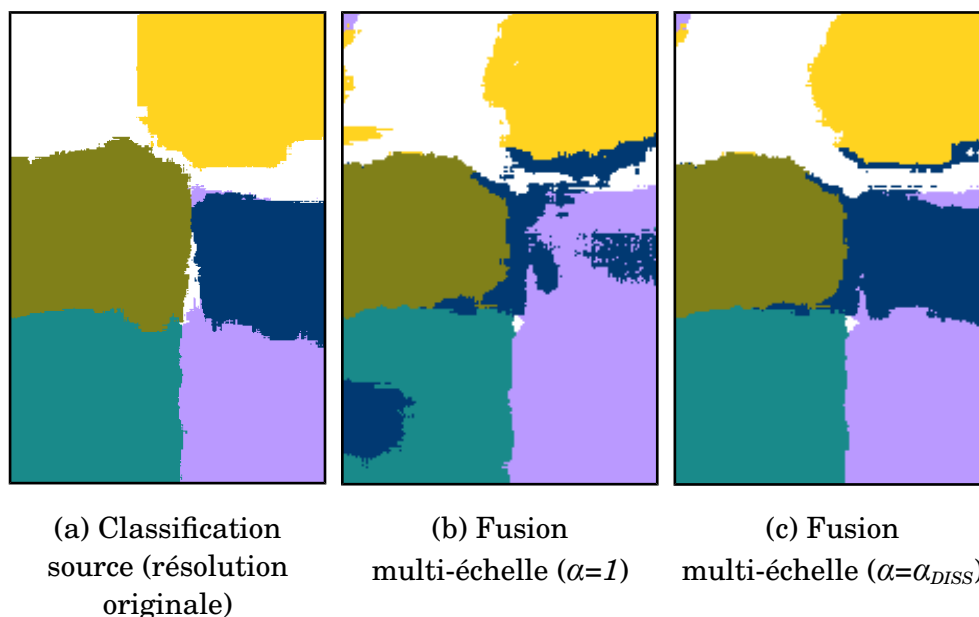


Figure 4.21 : mosaïque WorldView-2 : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c)

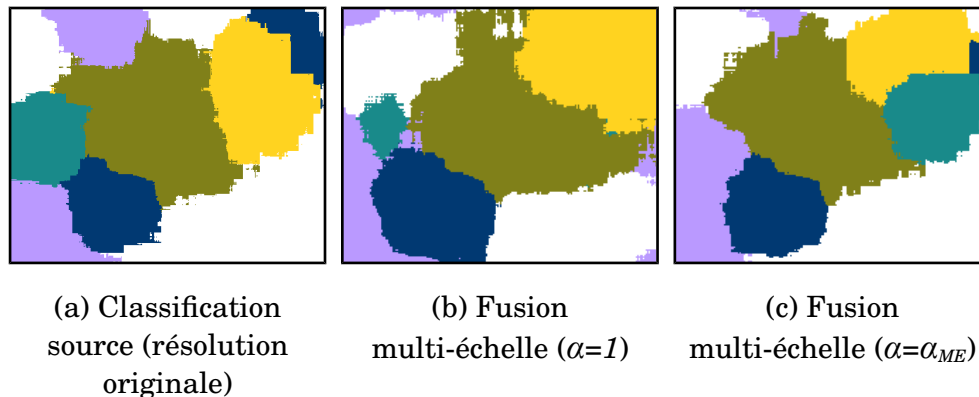


Figure 4.22 : placette IKONOS : classification KPC-SIFT originale (a), résultat de la fusion multi-échelle sans (b) et avec estimation du paramètre de fiabilité (c)

Tableau 4.18 : TBC_{source} (classification KPC-SIFT non dirigée) et ΔTBC , incluant l'estimation du paramètre de fiabilité, pour chaque image (en %)

Image	TBC_{source}	ΔTBC
Mosaïque IKONOS	79	-1
Mosaïque WorldView-2	86	+1
Placette IKONOS	41	+5

Les figures 4.20 et 4.22 ne permettent pas d'observer un réel changement de la qualité de la classification (malgré un ΔTBC élevé dans le cas de la placette IKONOS). En revanche, on constate des modifications intéressantes dans le cas de la mosaïque WorldView-2 (figure 4.21) : en comparant les figures 4.21(a) et (c), on observe principalement une plus grande hétérogénéité (dans la classe bleue notamment) bien que ΔTBC soit positif. Ce phénomène pourrait probablement être corrigé par des méthodes de morphologie mathématique (érosion/dilatation), par exemple. Ce post-traitement n'a pas été mis en œuvre à cause d'un nombre trop restreint d'images complexes qui permettrait d'en valider l'efficacité.

4.3. Discussion et conclusion partielle

Les tests menés sur les images source ont montré la pertinence de l'algorithme KPC pour la classification d'images texturées complexes. Ces résultats restent assez généraux, d'autant plus que deux images source sont des images synthétiques « idéales », mais permettent néanmoins de mettre en évidence la capacité discriminatoire de KPC. Dans les deux mosaïques, les textures ont été correctement identifiées, avec des résultats variables quant à la détection des frontières.

Il faut noter que la structure des textures représentées dans ces images est naturellement adaptée aux classifications orientées objet : il s'agit en effet majoritairement de textures forestières granuleuses, homogènes aux échelles élevées mais hétérogènes à

haute résolution. Il apparaît d'ailleurs clairement dans les résultats de la classification de l'image WorldView-2 que ces textures sont les plus difficiles à différencier pour les algorithmes comparatifs.

De même, certaines textures de la mosaïque WorldView-2 sont de type agricole : champ de maïs, plantation d'épinettes et de sapins. De telles textures présentent un plus haut niveau de régularité et constituent un défi moins important mais, malgré tout, non trivial comme le montrent également les résultats de la classification.

Bien que l'un des objectifs de ce travail soit de mettre en place une méthode de classification non dirigée, une alternative supervisée a été évaluée avec deux algorithmes de partitionnement différents (*k-means* et *k-NN*). Deux résultats ressortent de cette étude : l'approche supervisée apporte un gain mineur à l'algorithme KPC (comparativement aux autres algorithmes), et l'algorithme *k-NN* permet un gain substantiel de temps de calcul. Ce second résultat est dû à la faible complexité de *k-NN*; en effet, l'étape la plus coûteuse de KPC reste la phase de calcul des descripteurs texturaux, ce qui minimise l'impact de *k-NN*.

Les bonnes performances de l'algorithme KPC sont toutefois à nuancer, du fait de l'important problème lié à la taille des fenêtres de classification (voisinages). Cette difficulté, classique dans ce type d'approche, est liée au fait qu'un maximum de points-clés doivent être inclus dans chaque voisinage pour pouvoir caractériser aussi finement que possible la texture présente. Outre le fait que cela entraîne une imprécision dans la détection des frontières inter-texturales, un autre problème se pose : plus le voisinage est grand, plus le risque de mélange de textures est présent, diminuant d'autant plus l'efficacité de la classification.

Pour pallier ce problème, une approche de type multi-échelle a été proposée. La fusion des échelles de classification a été réalisée à l'aide de la théorie de l'évidence (DST), afin de résoudre les difficultés liées aux mélanges de classe (incertitude), en plus de l'imprécision des frontières. Cette approche a mené à des résultats généralement satis-

faisants : sur les trois images de test, deux ont vu leur TBC augmenter (de 1 et 5 %). La troisième (la mosaïque IKONOS) a en revanche subi une légère baisse de son TBC, inférieure à 1 %. Ces résultats sont toutefois prometteurs, car c'est l'image « naturelle » (l'image IKONOS) qui a subi la plus forte augmentation de TBC. Il ressort ainsi des résultats que l'apport de la fusion multi-échelle est d'autant plus grand que l'image est difficile à classifier. Sans prendre en compte l'introduction du paramètre de fiabilité, nous avons constaté que la fusion multi-échelle améliore la qualité de la classification lorsque le TBC initial (c'est-à-dire en résolution normale) est inférieur à environ 70 %. Ce résultat a été obtenu en introduisant un nouveau lot d'images construites en ajoutant divers niveaux de bruit aux trois images sources.

Plus généralement, il appert que la théorie de l'évidence est bien adaptée à la fusion de descripteurs texturaux complexes. Cette observation n'est pas triviale : la théorie de l'évidence est le plus souvent utilisée pour fusionner des données scalaires (niveaux de gris, probabilités, etc.) et non vectorielles, à plus forte raison lorsque les vecteurs en question ont une dimensionnalité élevée (jusqu'à 2000 éléments dans le cas présent). La nature de ces données impose des contraintes fortes à l'étape d'initialisation des fonctions de masse, étape cruciale dans le processus de fusion. En effet, la seule approche envisageable ici consiste à employer un modèle de distance pour initialiser les masses. Un tel modèle repose sur la distance entre un vecteur descripteur et les centroïdes des partitions : par conséquent, il ne permet d'initialiser que les hypothèses simples, ce qui ne tire pas pleinement parti de la puissance de la théorie de l'évidence. Un modèle de distance composé a été proposé afin d'initialiser également les masses des hypothèses duales; cependant, ce modèle ne parvient pas à dépasser le précédent. On constate donc qu'il existe une marge de progrès à ce niveau, qui nécessiterait une étude bien plus poussée.

L'efficacité de la fusion multi-échelle est en partie due aux méthodes d'estimation du paramètre de fiabilité. Parmi toutes les méthodes proposées dans le cadre de ce travail,

seules deux ont été détaillées et évaluées (les autres ne donnant pas de résultats préliminaires satisfaisants) : une approche par estimation *a priori* de fréquence des classes et une approche générique multi-échelle. La seconde fournit des résultats légèrement meilleurs et est probablement plus adaptée à une utilisation pratique, car elle ne nécessite pas d'estimations *a priori* sur la nature et la fréquence des classes de l'image source. Elle exige toutefois de spécifier un paramètre dont la valeur a une influence non négligeable sur le résultat final.

Enfin, il apparaît clairement qu'un avantage important de la théorie de l'évidence est qu'elle fait ressortir les zones d'ambiguïté à l'intérieur d'une image, malgré les limitations de la méthode d'initialisation des fonctions de masse. Ces zones sont, logiquement, principalement localisées aux frontières inter-textures mais peuvent également apparaître à l'intérieur de certaines textures présentant peu de traits caractéristiques. Dans ce dernier cas, le phénomène d'ambiguïté intra-texturale est amplifié par l'approche multi-échelle, la baisse de résolution entraînant une aggravation du phénomène de mélange de classes.

5. Conclusion

5.1. Recherche effectuée

La principale originalité de ce travail est la description d'un nouvel algorithme d'analyse texturale, nommé KPC, combiné avec une méthode de fusion évidentielle multi-échelle. Cet algorithme repose sur une approche orientée objet, dont le principal intérêt est d'être particulièrement adapté aux images à très haute résolution spatiale (THRS). En effet, ces images contiennent une grande quantité de détails pouvant être plus ou moins aisément détectés et analysés, permettant *in fine* de caractériser les textures présentes. Pour cette raison, les approches orientées objets offrent un potentiel élevé pour l'analyse des images fines de télédétection et font l'objet de nombreuses recherches.

L'algorithme KPC repose sur la détection, la classification et l'analyse spatiale de points-clés. Deux variantes de la phase de détection sont proposées : l'une se base sur l'algorithme SIFT, et l'autre sur SURF; ces deux algorithmes sont couramment employés en vision par ordinateur. L'analyse spatiale est l'étape la plus novatrice : elle a pour but de ne pas baser la classification texturale uniquement sur les caractéristiques intrinsèques des points-clés, mais également sur la manière dont ces derniers sont répartis au sein d'une région donnée.

Dans un premier temps, plusieurs paramètres de l'algorithme KPC ont été étudiés (section 4.1.1) afin d'en déterminer l'influence précise et de mettre en évidence des pistes d'optimisation. Il a notamment été montré que quelques paramètres (nombre de classes de points-clés, nombre de composantes d'ACP) peuvent mener à une amélioration de la qualité de la classification, au prix d'un accroissement considérable de la durée d'exécution de l'algorithme. Afin de garder cette dernière à un niveau raisonnable, certaines valeurs sous-optimales de paramètres ont été utilisées.

Les performances de l'algorithme KPC ont ensuite été comparées à celles de quatre algorithmes de classification texturale de référence (section 4.1.2). La capacité supérieure de KPC à discriminer des textures au sein d'une même image a ainsi été démontrée. En particulier, KPC s'est avéré capable de différencier des textures forestières très similaires, contrairement aux autres algorithmes testés. De plus, les résultats obtenus à l'aide de la mosaïque WorldView-2 laissent penser que l'efficacité de KPC est d'autant plus élevée que la résolution spatiale de l'image testée est haute.

Toutes les classifications précédentes ont été réalisées de manière non supervisée; des tests en mode supervisé ont été menés. Il en ressort principalement que l'approche supervisée apporte un gain minime à KPC, comparativement aux autres algorithmes de classification. On peut en déduire que KPC est plus autonome que les autres algorithmes, dans le sens où il est capable d'extraire davantage d'informations des images étudiées. Il est donc particulièrement adapté aux contextes de classification automatisée.

Pour pallier le problème important de détection des frontières dû à la large taille des voisinages utilisés, une méthode de fusion évidentielle multi-échelle a été proposée. La version initiale de cette méthode a fourni des résultats mitigés : sur les trois images testées, seule une (l'image IKONOS) a vu son TBC augmenter. Les deux autres ont subi une baisse de qualité parfois importante. De plus, cette première approche a permis de montrer que le modèle de distance composé, utilisé pour initialiser les masses, ne dépasse pas le modèle de distance simple, contrairement à ce qui était attendu. Enfin, il a été constaté que les méthodes de combinaison des masses et de décision fournissent des résultats variables d'une image à l'autre, de sorte qu'il est impossible de dégager une série de méthodes « universelles » qui pourraient être appliquées à n'importe quelle image tout en obtenant un résultat optimal.

Dans un second temps, deux méthodes d'estimation du coefficient de fiabilité ont été incorporées à l'algorithme de fusion multi-échelle. Ces méthodes ont permis d'amélio-

rer les résultats obtenus (le gain moyen est compris entre 2 et 3 %). Avec l'ajout de ces méthodes, le gain moyen apporté par la fusion multi-échelle devient positif, alors qu'il était auparavant négatif. La fusion multi-échelle proposée devient donc un outil viable, d'autant plus que les gains de qualité qu'elle apporte sont d'autant plus élevés que l'image source est difficile à classifier.

Le premier objectif secondaire énoncé consistait à développer un algorithme d'analyse texturale orienté objet adapté aux images de télédétection THRS, répondant à certains critères. Il a été montré que l'algorithme KPC est effectivement bien adapté à cette tâche, comparativement aux autres méthodes testées. Examinons à présent les critères initialement retenus :

- robustesse face aux images de télédétection (impliquant une invariance de contraste, de luminosité, d'échelle et de rotation) : les détecteurs SIFT et SURF présentent une telle robustesse, et le reste de l'algorithme KPC est indépendant des conditions d'acquisition de l'image. Ce critère est donc respecté.
- séparabilité des classes (capacité à différencier des textures similaires) : les tests effectués ont montré la bonne capacité de KPC à discriminer des textures similaires au point d'être difficiles à différencier par l'œil humain.
- complexité (durée de calcul) : bien que plusieurs paramètres aient été choisis de manière à conserver un temps de calcul raisonnable, KPC présente une complexité supérieure aux approches statistiques (matrices de co-occurrence, spectre de texture). Le choix d'un algorithme de partitionnement simple tel que k -NN permet toutefois de diminuer le temps de calcul.
- nombre de paramètres à régler manuellement : l'algorithme KPC comprend trois paramètres obligatoires à régler manuellement (cf section 5.2). Ce nombre est comparable à d'autres algorithmes et ne semble donc pas déraisonnable, même s'il serait souhaitable de le diminuer.

On constate donc que, dans l'ensemble, le premier sous-objectif est atteint.

Le second objectif secondaire consistait à proposer une méthode de fusion texturale multi-échelle basée sur la théorie de l'évidence, dans le but d'améliorer les classifications obtenues. On a vu que cet objectif a pu généralement être atteint, le résultat variant d'une image à l'autre.

5.2. Limitations

Les algorithmes de classification texturale et de fusion multi-échelle proposés souffrent de plusieurs limitations, dont certaines sont courantes dans ce type d'approche. Le problème principal a été évoqué à plusieurs reprises : il s'agit de la mauvaise détection des frontières entre textures. Il s'agit d'une difficulté inévitable en classification orientée objet, due à la large taille des voisinages permettant de caractériser une texture. Cet effet entraîne de plus un fort mélange de classes au niveau des frontières, réduisant d'autant plus la qualité de la classification. La fusion multi-échelle permet de limiter dans une faible mesure ce problème, notamment en mettant en évidence les zones de mélange de classes par le biais des hypothèses composées. Le résultat n'est toutefois pas encore satisfaisant dans des conditions réelles.

Une autre limitation, classique, concerne l'estimation des paramètres utilisés. L'algorithme KPC comprend trois paramètres qui doivent être réglés manuellement d'une image à l'autre : le nombre de classes, la taille des voisinages et le vecteur de distances inter-points pour le calcul des statistiques spatiales. L'algorithme de fusion multi-échelle comporte deux paramètres : le paramètre de distance γ et le paramètre du modèle de fiabilité (d ou δ , suivant le modèle). Il est possible d'envisager des pistes d'estimation automatique de certains de ces paramètres, qui seront détaillées dans la section suivante.

On peut enfin évoquer le problème du temps de calcul. Bien que l'objectif n'ait jamais été de faire de la classification en temps réel, ou du moins dans une durée très limitée,

on pourrait néanmoins espérer que l'algorithme KPC ait une complexité du même ordre de grandeur que les algorithmes comparatifs. Cela n'est pas le cas pour de grandes images, où la variante KPC-SURF peut être plusieurs dizaines de fois plus lente que d'autres algorithmes. L'analyse des raisons de cette lenteur laisse penser que le temps de calcul devrait augmenter de façon exponentielle avec la taille de l'image étudiée.

5.3. *Recommandations*

Afin de répondre aux limites évoquées dans la section précédente, plusieurs pistes de solutions sont ici suggérées.

5.3.1. Détection de frontières

L'amélioration de la détection des frontières est un problème qui n'a pas de solution simple. Comme mentionné précédemment, l'approche multi-échelle n'a pas permis de résoudre efficacement cette difficulté. La solution la plus « évidente » consisterait à utiliser une approche indépendante pour détecter les frontières, soit par un filtre de détection de contours (de type Sobel ou Canny, par exemple), soit par une méthode de classification texturale à faible voisinage, comme les matrices de co-occurrence ou les filtres de Gabor. La difficulté serait alors de fusionner cette information de contour avec la classification KPC. On pourrait envisager un algorithme qui associerait à chaque « segment » détecté par l'algorithme de contour une classe détectée par KPC, sur des critères de proximité spatiale.

5.3.2. Estimation du paramètre γ

La justification du choix de la valeur de γ est longuement évoquée dans l'annexe 1. Il est possible de tirer de cette analyse une méthode d'estimation automatique de γ . Cela nécessiterait toutefois de corriger certaines approximations effectuées lors des calculs.

5.3.3. Estimation du nombre de classes

L'automatisation du choix du nombre de classes est liée à la méthode de partitionnement final utilisée. L'algorithme *k-means* utilisé exige de connaître ce paramètre *a priori*. Une autre méthode doit donc être mise en œuvre. Les approches de type classification hiérarchiques peuvent s'avérer efficaces dans ce contexte, bien que leur complexité calculatoire soit élevée.

5.3.4. Prise en compte des hypothèses composées

Bien que la fusion multi-échelle ait fourni des résultats intéressants, l'approche utilisée n'a pas permis de tirer pleinement parti des possibilités de la théorie de l'évidence, pour deux raisons : premièrement, il n'a pas été possible de mettre au point un modèle efficace d'initialisation des fonctions de masses permettant d'assigner des valeurs aux masses composées, malgré la mise en œuvre d'un modèle de distance composé. L'inefficacité de ce dernier provient probablement de la dimensionnalité élevée de l'espace textural. Une méthode de réduction de la dimensionnalité pourrait peut-être accroître l'efficacité du modèle proposé.

Le travail de Germain (2006) permet d'envisager une autre piste pour assigner des valeurs aux hypothèses composées, par le biais de la théorie des ensembles flous. Cela exigerait toutefois de construire une métrique reflétant l'appartenance d'un vecteur descripteur à l'ensemble des partitions.

Deuxièmement, il a été montré dans les résultats que la fusion multi-échelle permet de dégager non seulement des classes simples, comme cela est habituellement le cas dans ce type d'application, mais également des classes composées (union de plusieurs classes) indiquant une zone d'incertitude. Il serait intéressant d'exploiter ces segments de classes composées pour améliorer la qualité de la classification.

5.3.5. Diminution du temps de calcul

L'analyse menée dans la section 4.1.2.1. b a montré que l'étape la plus longue de KPC est la construction des vecteurs descripteurs. Ces vecteurs sont bâtis à l'aide du calcul des statistiques spatiales d'ordre 2 des points-clés. On pourrait envisager de simplifier ces statistiques (en éliminant le calcul de valeurs peu significatives, par exemple) : en plus d'accélérer la classification, cela permettrait de réduire la dimensionnalité des données et pourrait donc contribuer à résoudre d'autres problèmes.

5.4. Perspectives

Pour finir sont listées ici quelques suggestions de recherches complémentaires, permettant d'étendre le cadre du présent travail.

5.4.1. Modèles de fiabilité

Les modèles de fiabilité proposés sont très génériques et ne tiennent pas compte des spécificités des classifications sources. Il serait intéressant d'y intégrer des métriques telles que la dispersion des vecteurs descripteurs ou la dissimilarité entre les sources. La première de ces deux approches n'a pas fourni de résultats préliminaires pertinents – probablement en raison de la dimensionnalité élevée des données – mais constitue néanmoins une piste potentielle. La seconde, décrite par Florea et Bossé (2009), est essentiellement applicable en présence de nombreuses sources (ou échelles, dans le cas présent). Il est donc envisageable d'utiliser cette approche si le nombre d'échelles augmente, en cas par exemple d'images à résolution spatiale plus élevée.

5.4.2. Ajout de données multispectrales

Dans ce travail, nous nous sommes exclusivement concentrés sur des images panchromatiques, car les détecteurs SIFT et SURF ne sont pas adaptés aux données multispectrales. Il est certain que la prise en compte des bandes spectrales dans l'analyse

texturale devrait améliorer la classification finale, la résolution limitée du mode multispectral pouvant être simplement améliorée par des opérations de type pansharpening. De nombreux détecteurs de points-clés pour des images en couleur ont déjà été mis au point; une revue en a été faite par (Van De Sande *et al.*, 2010). Parmi les plus connus, citons notamment C-SIFT d'Abdel-Hakim et Farag (2006), HSV-SIFT de (Bosch *et al.*, 2008) ou RGB-SIFT (Burghouts et Geusebroek, 2009).

5.4.3. Images à plus haute résolution spatiale

Les résultats de la section 4.1.2 ont, comme on l'a vu, mis en évidence le fait que l'image avec la plus haute résolution spatiale a été la mieux classifiée. Cela suggère l'idée que l'algorithme KPC est d'autant plus performant que l'image source présente un grand nombre de détails, ce qui est naturel étant donné qu'il s'agit d'un algorithme orienté objet. Il serait bon de démontrer ce phénomène en se servant d'images aéroportées ou issues de satellites tels que WorldView-3 ou GeoEye-2.

Incidemment, il serait intéressant de tenter de classifier des thèmes différents de ceux utilisés dans ce travail (principalement foresterie et agricole). Analyser une image comprenant une grande proportion de zones urbaines permettrait de mieux juger les capacités de l'algorithme KPC.

5.4.4. Classification par apprentissage

La classification des descripteurs texturaux a été menée par de simples algorithmes de partitionnement (*k-means*, *k-NN*). Bien que l'on sorte du cadre de la classification non dirigée, il serait intéressant de mettre en place des méthodes d'apprentissage plus sophistiquées, telles que SVM (*Support Vector Machine*) ou un réseau de neurones. Les approches de type *deep learning*, en pleine expansion, pourraient également être adaptées.

6. Annexes

Annexe 1 – Étude du paramètre de distance γ

γ est un paramètre employé dans les modèles distances (cf. section 3.1.2.2. a) afin de diminuer ou d'augmenter la valeur moyenne attribuée aux fonctions de masse lors de l'initialisation de ces dernières. L'initialisation des masses est une étape importante, car elle influence fortement l'issue de la fusion. Le choix de γ est donc délicat : de faibles variations de sa valeur peuvent grandement diminuer la qualité du résultat final.

En pratique, et pour toutes les images, les valeurs comprises entre 10^{-2} et 10^{-3} aboutissent aux meilleurs résultats. La valeur 10^{-2} a ainsi été choisie pour le calcul des résultats de la section 4.2.

Dans cette annexe, nous allons justifier ce choix par une analyse théorique du modèle de distance et du conflit dans la théorie de l'évidence. Pour obtenir des valeurs numériques, nous nous placerons dans le cas de la fusion multi-échelle de la mosaïque WorldView-2, le raisonnement restant valide pour les autres images.

Commençons par établir la relation entre le conflit k et γ . Comme indiqué dans la section 2.3.4.2, k est par définition égal à la masse de l'ensemble vide :

$$k = m(\emptyset) = \sum_{A_1 \cap \dots \cap A_S = \emptyset} m_1(A_1) \times \dots \times m_S(A_S) \quad (6.1)$$

Ici, S est le nombre de sources, les A_i ($1 \leq i \leq S$) sont des sous-ensembles d'hypothèses, et m_i ($1 \leq i \leq S$) les fonctions de masse des sources.

Si l'on fait l'hypothèse que seules les hypothèses simples sont initialisées (excepté l'ensemble de toutes les hypothèses Θ), ce qui est le cas du modèle de distance simple, il est possible de déterminer l'espérance de k , qui sera par la suite notée \bar{k} :

$$\bar{k} = \sum_{A_1 \cap \dots \cap A_S = \emptyset} (\bar{m}^S + D) \quad (6.2)$$

où \bar{m} représente l'espérance de la masse d'un singleton (hypothèse simple) et D un terme de covariance (dû au fait que les m_i ne sont pas des variables indépendantes) que nous ne chercherons pas à déterminer et qui sera considéré constant.

Pour S sources et C classes, le nombre de singletons dont l'intersection totale est vide est égal à $C^S - C$. L'équation (6.2) devient donc :

$$\bar{k} = (C^S - C) \cdot (\bar{m}^S + D) \quad (6.3)$$

D'après l'équation (3.2) du modèle de distance, et en ne tenant pas compte du paramètre de fiabilité α , la valeur moyenne \bar{m} de la masse d'un singleton peut être approximée par :

$$\bar{m} \approx \frac{1}{C} \sum_{i=1}^C e^{-\gamma \bar{d}_i^2} \quad (6.4)$$

où \bar{d}_i^2 représente le carré de la distance moyenne entre un descripteur et le i -ème plus proche centroïde de partition. Dans la mesure où le terme exponentiel décroît très vite avec la distance, on peut simplifier (6.4) en retenant seulement le premier terme (\bar{d}_1^2 sera simplement noté \bar{d}^2) :

$$\bar{m} \approx \frac{1}{C} e^{-\gamma \bar{d}^2} \quad (6.5)$$

En injectant (6.5) dans (6.3), on obtient :

$$\bar{k} \approx (C^S - C) \cdot (e^{-S\gamma \bar{d}^2} / C^S + D) \quad (6.6)$$

Ainsi, il est possible d'estimer \bar{k} (ou, du moins, son ordre de grandeur) à l'aide d'un nombre réduit de variables et de paramètres : C , S , γ et \bar{d}^2 .

Une simple inversion de (6.6) permet d'exprimer γ en fonction de \bar{k} :

$$\gamma \approx -\frac{1}{S\bar{d}^2} \cdot \ln \left[C^S \left(\frac{\bar{k}}{C^S - C} - D \right) \right] \quad (6.7)$$

L'équation (6.7) est le premier résultat intéressant. En effet, le conflit k est une valeur que l'on souhaite habituellement être la plus faible possible, car un conflit élevé diminue mécaniquement les masses des hypothèses non nulles; or, ce sont ces masses qui sont significatives et permettent d'aboutir à une décision fiable.

L'équation (6.7) montre que γ est une fonction décroissante de \bar{k} : plus l'on veut que \bar{k} soit faible, plus γ devra être élevé. En affectant aux paramètres S , \bar{d}^2 et C les valeurs déduites de la fusion de la mosaïque WorldView-2, on obtient l'exemple suivant : pour avoir un conflit de 1 %, γ doit être de l'ordre de 4×10^{-3} .

Il semble donc naturel d'augmenter γ de manière à aboutir à un conflit quasi-nul : cependant, l'équation du modèle de distance montre bien que les fonctions de masses diminuent aussi lorsque γ augmente, à tel point que toutes les masses peuvent également devenir quasi-nulles. Cela n'est évidemment pas souhaitable pour les raisons évoquées plus haut.

On peut tenter de déterminer le bon ordre de grandeur de γ en calculant la relation entre la masse maximale moyenne (c'est-à-dire la masse moyenne de l'hypothèse simple la plus proche d'un échantillon donné) et γ . Cette relation est simplement donnée par le modèle de distance :

$$\bar{m}_{max} \approx e^{-\gamma \bar{d}^2} \quad (6.8)$$

Il s'agit encore une fois d'une approximation, basée sur l'hypothèse que les distances entre un échantillon et le centroïde le plus proche sont assez homogènes.

En injectant (6.7) dans (6.8), on obtient :

$$\bar{m}_{max} \approx \exp \left(\frac{1}{S\bar{d}^2} \cdot \ln \left[C^S \left(\frac{\bar{k}}{C^S - C} - D \right) \right] \cdot \bar{d}^2 \right) \quad (6.9)$$

Puis, en simplifiant :

$$\begin{aligned}\bar{m}_{max} &\approx \exp\left(\frac{1}{S} \cdot \ln\left[C^S \left(\frac{\bar{k}}{C^S - C} - D\right)\right]\right) \\ &\approx C \left(\frac{\bar{k}}{C^S - C} - D\right)^{1/S}\end{aligned}\quad (6.10)$$

On obtient alors \bar{k} en fonction de \bar{m}_{max} :

$$\bar{k} \approx (C^S - C) \cdot \left[\left(\frac{\bar{m}_{max}}{C}\right)^S + D\right] \quad (6.11)$$

Enfin, on peut injecter (6.11) dans (6.7) :

$$\begin{aligned}\gamma &\approx -\frac{1}{Sd^2} \cdot \ln\left[C^S \left(\frac{(C^S - C) \cdot \left[\left(\frac{\bar{m}_{max}}{C}\right)^S + D\right]}{C^S - C} - D\right)\right] \\ &\approx -\frac{1}{d^2} \ln(\bar{m}_{max})\end{aligned}\quad (6.12)$$

L'équation (6.12) montre que γ est d'autant plus faible que \bar{m}_{max} est élevé. Ainsi, si l'on souhaite que les masses les plus fortes aient un ordre de grandeur de 10^{-1} , alors γ doit être d'environ 5×10^{-3} (toujours en utilisant la valeur de \bar{d}^2 obtenue pour la mosaïque WorldView-2). Cette contrainte, qui encourage à choisir une valeur de γ aussi faible que possible, entre en conflit avec la contrainte issue de l'équation (6.7), qui forçait γ à être aussi élevé que possible.

Il s'agit donc de choisir pour γ une valeur intermédiaire satisfaisant ces deux contraintes. La figure 6.1 permet de mieux appréhender le problème : on y observe que pour les faibles valeurs de γ , le conflit est trop proche de 1; lorsque γ est élevé, la masse maximale moyenne est trop proche de 0. La valeur idéale de γ semble se trouver entre 10^{-3} et 10^{-2} , ce qui est confirmé par les expérimentations. En pratique, la valeur 10^{-2} donne des résultats légèrement meilleurs que 10^{-3} .

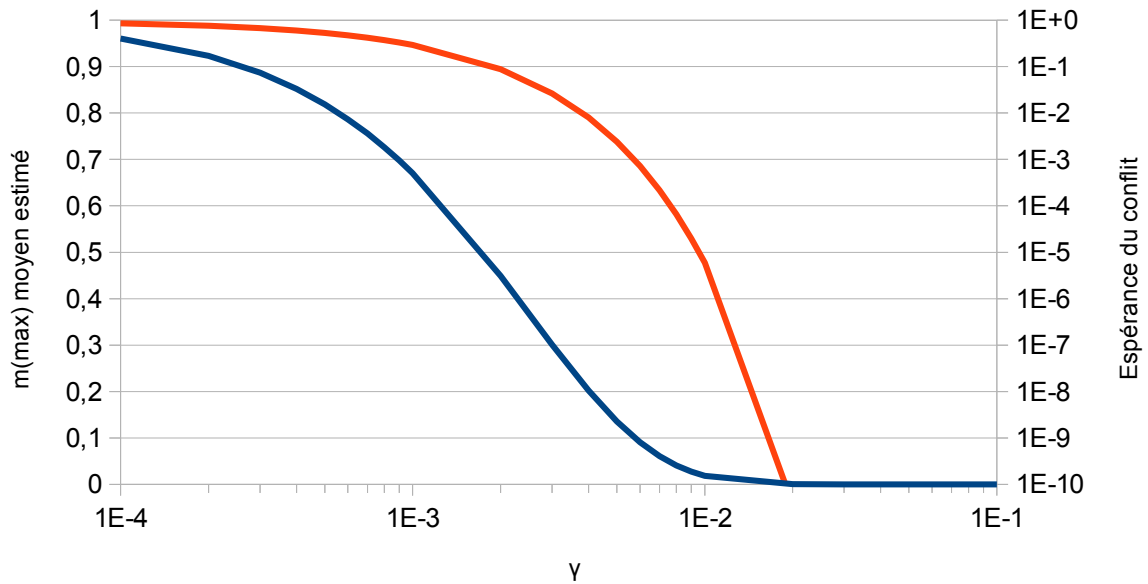


Figure 6.1 : masse maximale moyenne (\bar{m}_{max} , en bleu) et espérance du conflit (\bar{k} , en rouge) en fonction de γ .

Les paramètres utilisés pour construire la figure 6.1 ainsi que pour les exemples sont les suivants :

- $\bar{d}^2 = 400$
- $S = 3$
- $C = 6$
- $D = 0$

Annexe 2 – Détails des résultats de la fusion multi-échelle

Tableau 6.1 : TBC (en %) et coefficients kappa (entre parenthèses) des images source multi-échelle.

Image	Résolution		
	100 %	50 %	25 %
Mosaïque IKONOS	79 (0,73)	71 (0,64)	74 (0,68)
Mosaïque WorldView-2	86 (0,84)	67 (0,61)	52 (0,42)
Placette IKONOS	41 (0,25)	28 (0,10)	41 (0,28)

Tableau 6.2 : Résultats de la fusion à 3 échelles de la mosaïque IKONOS originale, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	76 (0,70)	76 (0,70)	76 (0,70)
	Robuste	76 (0,70)	76 (0,70)	76 (0,70)
	Disjonctive	76 (0,70)	76 (0,70)	76 (0,70)
Modèle de distance composé	Conjonctive	76 (0,70)	76 (0,70)	76 (0,70)
	Robuste	76 (0,70)	76 (0,70)	76 (0,70)
	Disjonctive	76 (0,70)	76 (0,70)	76 (0,70)

Tableau 6.3 : Résultats de la fusion à 3 échelles de la mosaïque WorldView-2, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa. Les meilleures valeurs sont en gras.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	77 (0,72)	77 (0,72)	77 (0,72)
	Robuste	77 (0,72)	78 (0,74)	79 (0,74)
	Disjonctive	77 (0,72)	78 (0,73)	79 (0,74)
Modèle de distance composé	Conjonctive	77 (0,72)	77 (0,72)	77 (0,72)
	Robuste	77 (0,72)	78 (0,73)	77 (0,73)
	Disjonctive	77 (0,72)	76 (0,71)	76 (0,71)

Tableau 6.4 : Résultats de la fusion à 3 échelles de la placette IKONOS, sans coefficient de fiabilité. La première valeur représente le TBC (en %), la seconde le coefficient kappa. Les meilleures valeurs sont en gras.

		Décision		
Initialisation	Combinaison	Maximum de crédibilité	Maximum de plausibilité	Maximum de probabilité pignistique
Modèle de distance simple	Conjonctive	48 (0,33)	48 (0,33)	48 (0,33)
	Robuste	48 (0,33)	40 (0,25)	38 (0,23)
	Disjonctive	48 (0,33)	40 (0,25)	38 (0,23)
Modèle de distance composé	Conjonctive	48 (0,33)	48 (0,33)	48 (0,33)
	Robuste	48 (0,33)	41 (0,26)	42 (0,27)
	Disjonctive	48 (0,33)	39 (0,24)	39 (0,24)

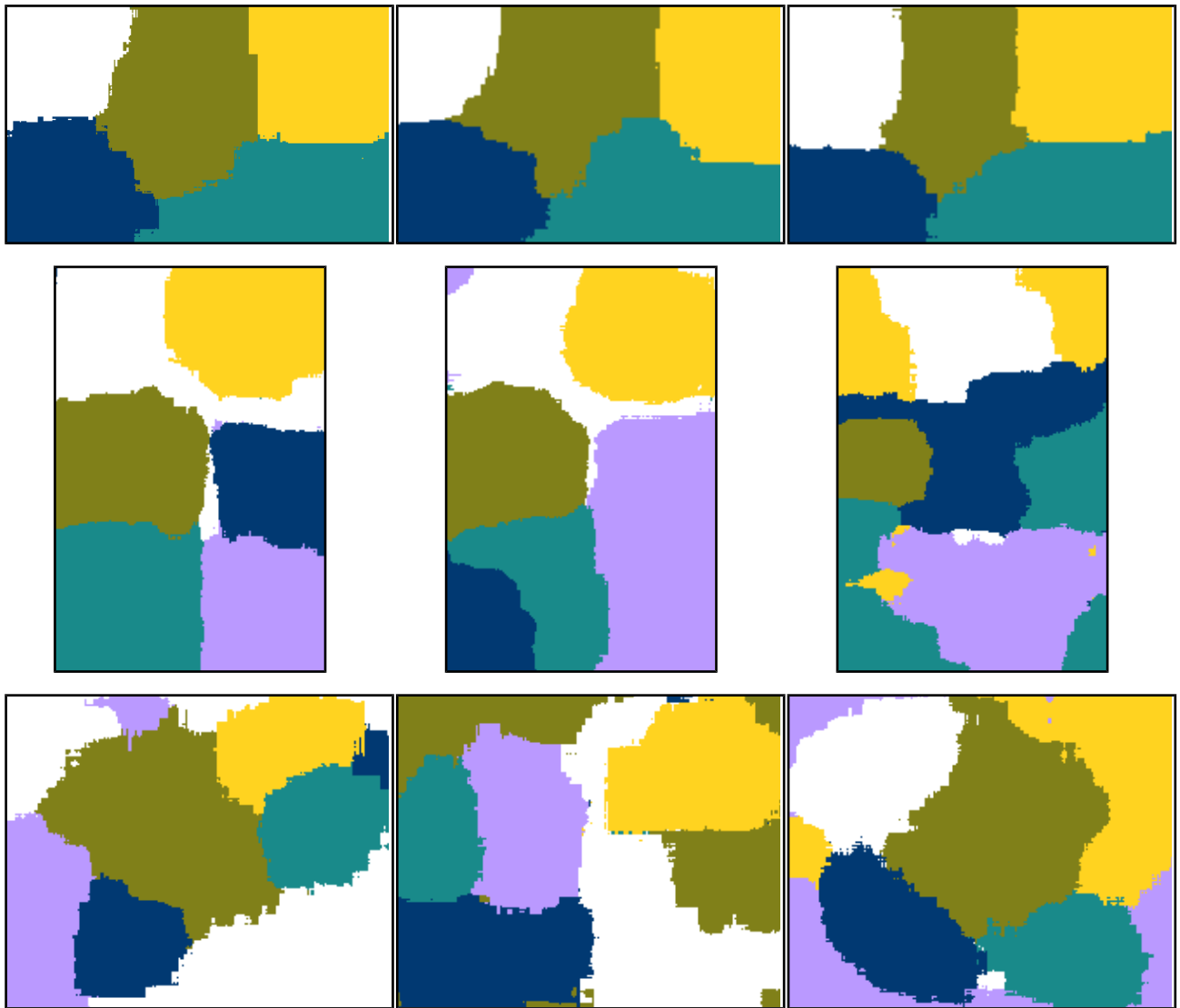


Figure 6.2 : Classifications des images source multi-échelle (colonne de gauche : résolution = 100 %, colonne centrale : résolution = 50 %, colonne de droite : résolution = 25 %).

7. Références

- Abdel-Hakim, A. E. et Farag, A. A. (2006) CSIFT: A SIFT Descriptor with Color Invariant Characteristics. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, p. 1978-1983.
- Appriou, A. (1993) Formulation et traitement de l'incertain en analyse multi-senseurs. 14° Colloque sur le traitement du signal et des images, FRA, 1993. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images.
- Arthur, D. et Vassilvitskii, S. (2007) K-means++: The Advantages of Careful Seeding. Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07, p. 1027–1035. Philadelphia, PA, USA : Society for Industrial and Applied Mathematics.
- Bauer, J., Sunderhauf, N. et Protzel, P. (2007) Comparing several implementations of two recently published feature detectors. Proc. of the International Conference on Intelligent and Autonomous Systems, vol. 6.
- Bay, H., Ess, A., Tuytelaars, T. et Van Gool, L. (2008) Speeded-up robust features (SURF). Computer vision and image understanding, vol. 110, n°3, p. 346–359.
- Bay, H., Tuytelaars, T. et Van Gool, L. (2006) Surf: Speeded up robust features. Computer Vision–ECCV 2006, p. 404–417.
- Blaschke, T. (2010) Object based image analysis for remote sensing. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 65, n°1, p. 2-16.
- Bloch, I. (2003) Fusion d'informations en traitement du signal et des images. Hermès Science Publications.
- Bloch, I. et Maître, H. (1994) Fusion de données en traitement d'images: modèles d'information et décisions. Traitement du Signal.
- Bosch, A., Zisserman, A. et Muñoz, X. (2008) Scene Classification Using a Hybrid

- Generative/Discriminative Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, n°4, p. 712-727.
- Bouakache, A., Belhadj-Aissa, A. et Mercier, G. (2009) Satellite image fusion using Dezert-Smarandache theory. *In* F. Smarandache et J. Dezert (dir.), *Advances and Applications of DsmT for Information Fusion* (Vol. 3, p. 549-564). American Research Press.
- Bovik, A. C., Clark, M. et Geisler, W. S. (1990) Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, n°1, p. 55–73.
- Burghouts, G. J. et Geusebroek, J.-M. (2009) Performance evaluation of local colour invariants. *Computer Vision and Image Understanding*, vol. 113, n°1, p. 48–62.
- Carleer, A., Debeir, O. et Wolff, E. (2005) Assessment of very high spatial resolution satellite image segmentations. *Photogrammetric Engineering and Remote Sensing*, vol. 71, n°11, p. 1285–1294.
- Corgne, S., Hubert-Moy, L., Dezert, J. et Mercier, G. (2003) Land cover change prediction with a new theory of plausible and paradoxical reasoning. *Proc. of Fusion*, p. 8–11.
- Costa, A. F., Humpire-Mamani, G. et Traina, A. J. M. (2012) An Efficient Algorithm for Fractal Analysis of Textures. *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, p. 39-46.
- Cross, G. R. et Jain, A. K. (1983) Markov Random Field Texture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, n°1, p. 25-39.
- Csurka, G., Dance, C., Fan, L., Willamowski, J. et Bray, C. (2004) Visual categorization with bags of keypoints. *Workshop on statistical learning in computer vision, ECCV*, vol. 1, p. 1–2. Prague.
- Dalal, N. et Triggs, B. (2005) Histograms of oriented gradients for human detection.

- 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, p. 886-893 vol. 1.
- Daugman, J. G. (1980) Two-dimensional spectral analysis of cortical receptive field profiles. *Vision research*, vol. 20, n°10, p. 847–856.
- Dempster, A. P. (1968) A generalization of Bayesian inference. *Journal of the Royal Statistical Society. Series B (Methodological)*, p. 205–247.
- Denoeux, T. (1995) A k-nearest neighbor classification rule based on Dempster-Shafer theory. *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 25, n°5, p. 804–813.
- Dezert, J. (2002) Foundations for a new theory of plausible and paradoxical reasoning. *Information and Security*, vol. 9, p. 13–57.
- Dubois, D. et Prade, H. (1986) A set-theoretic view of belief functions logical operations and approximations by fuzzy sets. *International Journal Of General System*, vol. 12, n°3, p. 193–226.
- Dubois, D. et Prade, H. (1988) Representation and combination of uncertainty with belief functions and possibility measures. *Computational Intelligence*, vol. 4, n°3, p. 244-264.
- Dubois, D. et Prade, H. (1992) On the Combination of Evidence in Various Mathematical Frameworks. *In J. Flamm et T. Luisi (dir.), Reliability Data Collection and Analysis, Eurocourses* (p. 213-241). Springer Netherlands.
- Eichmann, G. et Kasparis, T. (1988) Topologically invariant texture descriptors. *Computer vision, graphics, and image processing*, vol. 41, n°3, p. 267–281.
- Fabre, S., Appriou, A. et Briottet, X. (2001) Presentation and description of two classification methods using data fusion based on sensor management. *Information Fusion*, vol. 2, n°1, p. 49-71.
- Florea, M. C. et Bossé, E. (2009) Crisis Management using Dempster Shafer Theory: Using dissimilarity measures to characterize sources' reliability. *C3I for Crisis*,

Emergency and Consequence Management.

- Florea, M. C., Jousselme, A.-L., Bossé, É. et Grenier, D. (2009) Robust combination rules for evidence theory. *Information Fusion*, vol. 10, n°2, p. 183–197.
- Gaetan, C. et Guyon, X. (2008) *Modélisation et statistique spatiales*. Springer Verlag.
- Gauglitz, S., Höllerer, T. et Turk, M. (2011) Evaluation of interest point detectors and feature descriptors for visual tracking. *International journal of computer vision*, vol. 94, n°3, p. 335–360.
- Germain, M. (2006) *Classification multisource par la fusion évidentielle avec une nouvelle approche statistique floue*. Thèse de doctorat, Université de Sherbrooke.
- Gordon, J. et Shortliffe, E. H. (1985) A method for managing evidential reasoning in a hierarchical hypothesis space. *Artificial Intelligence*, vol. 26, n°3, p. 323-357.
- Goyal, R. K., Goh, W. L., Mital, D. P. et Chan, K. L. (1995) Scale and rotation invariant texture analysis based on structural property. *Industrial Electronics, Control, and Instrumentation, 1995., Proceedings of the 1995 IEEE IECON 21st International Conference on*, vol. 2, p. 1290–1294. IEEE.
- Hachicha, S. et Chaabane, F. (2009) Application of DSM theory for SAR image change detection. *2009 16th IEEE International Conference on Image Processing (ICIP)*, p. 3733-3736.
- Hall, D. L. et Llinas, J. (1997) An introduction to multisensor data fusion. *Proceedings of the IEEE*, vol. 85, n°1, p. 6-23.
- Haralick, R. M. (1979) Statistical and structural approaches to texture. *Proceedings of the IEEE*, vol. 67, n°5, p. 786–804.
- Haralick, R. M., Shanmugam, K. et Dinstein, I. H. (1973) Textural features for image classification. *IEEE Transactions on systems, man and cybernetics*, vol. 3, n°6, p. 610–621.
- Harris, C. et Stephens, M. (1988) A combined corner and edge detector. *Alvey vision*

- conference, vol. 15, p. 50. Citeseer.
- He, D.-C. et Wang, L. (1990) Texture unit, texture spectrum, and texture analysis. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 28, n°4, p. 509–512.
- He, D.-C. et Wang, L. (2010) Simplified texture spectrum for texture analysis. *Journal of Communication and Computer*, vol. 7, n°8, p. 44–53.
- Inagaki, T. (1991) Interdependence between safety-control policy and multiple-sensor schemes via Dempster-Shafer theory. *IEEE Transactions on Reliability*, vol. 40, n°2, p. 182–188.
- Irons, J. R. et Petersen, G. W. (1981) Texture transforms of remote sensing data. *Remote Sensing of Environment*, vol. 11, p. 359–370.
- Jain, R., Kasturi, R. et Schunck, B. G. (1995) *Machine vision (Vol. 5)*. McGraw-Hill New York.
- Karu, K., Jain, A. K. et Bolle, R. M. (1996) Is there any texture in the image? *Pattern Recognition*, vol. 29, n°9, p. 1437-1446.
- Lazebnik, S., Schmid, C. et Ponce, J. (2005) A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, n°8, p. 1265-1278.
- Le Hégarat-Masclé, S., Bloch, I. et Vidal-Madjar, D. (1997) Application of Dempster-Shafer evidence theory to unsupervised classification in multisource remote sensing. *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 35, n°4, p. 1018-1031.
- Lefebvre, A., Corpetti, T. et Hubert-Moy, L. (2010) Segmentation of very high spatial resolution panchromatic images based on wavelets and evidence theory. *SPIE Remote Sensing Conference*, vol. 7830.
- Lefevre, E., Colot, O. et Vannoorenberghe, P. (2002) Belief function combination and conflict management. *Information Fusion*, vol. 3, n°2, p. 149-162.
- Liu, C., Yuen, J., Torralba, A., Sivic, J. et Freeman, W. T. (2008) SIFT Flow: Dense

- Correspondence across Different Scenes. *In* D. Forsyth, P. Torr, et A. Zisserman (dir.), *Computer Vision – ECCV 2008, Lecture Notes in Computer Science* (p. 28-42). Springer Berlin Heidelberg.
- Lowe, D. G. (1999) Object recognition from local scale-invariant features. *Proceedings of the 7th IEEE International Conference on Computer Vision*, p. 1150-1157.
- Lowe, D. G. (2004) Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, vol. 60, n°2, p. 91–110.
- Mallat, S. G. (1989) A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, n°7, p. 674-693.
- Mikolajczyk, K. et Schmid, C. (2001) Indexing based on scale invariant interest points. *Proceedings of the Eighth IEEE International Conference on Computer Vision, 2001.*, vol. 1, p. 525-531.
- Mikolajczyk, K. et Schmid, C. (2002) An Affine Invariant Interest Point Detector. *In* A. Heyden, G. Sparr, M. Nielsen, et P. Johansen (dir.), *Computer Vision — ECCV 2002, Lecture Notes in Computer Science* (p. 128-142). Springer Berlin Heidelberg.
- Mikolajczyk, K. et Schmid, C. (2005) A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1615–1630.
- Mora, B. (2009) Cartographie de paramètres forestiers par fusion évidentielle de données géospatiales multi-sources application aux peuplements forestiers en régénération et feuillus matures du Sud du Québec. Thèse de doctorat, Université de Sherbrooke, 153 p.
- Mora, B., Fournier, R. A. et Foucher, S. (2011) Application of evidential reasoning to improve the mapping of regenerating forest stands. *International Journal of Applied Earth Observation and Geoinformation*, vol. 13, n°3, p. 458–467.
- Moravec, H. P. (1980) Obstacle avoidance and navigation in the real world by a seeing

- robot rover. DTIC Document.
- Myint, S. W. (2003) Fractal approaches in texture analysis and classification of remotely sensed data: Comparisons with spatial autocorrelation techniques and simple descriptive statistics. *International Journal of Remote Sensing*, vol. 24, n°9, p. 1925-1947.
- Pentland, A. P. (1984) Fractal-Based Description of Natural Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, n°6, p. 661-674.
- Pohl, C. et Van Genderen, J. L. (1998) Review article Multisensor image fusion in remote sensing: Concepts, methods and applications. *International Journal of Remote Sensing*, vol. 19, n°5, p. 823-854.
- Puissant, A., Hirsch, J. et Weber, C. (2005) The utility of texture analysis to improve per-pixel classification for high to very high spatial resolution imagery. *International Journal of Remote Sensing*, vol. 26, n°4, p. 733-745.
- Ripley, B. D. (1976) The second-order analysis of stationary point processes. *Journal of Applied Probability*, vol. 13, n°2, p. 255–266.
- Schmid, C., Mohr, R. et Bauckhage, C. (2000) Evaluation of Interest Point Detectors. *International Journal of Computer Vision*, vol. 37, n°2, p. 151-172.
- Sentz, K. et Ferson, S. (2002) Combination of evidence in Dempster-Shafer theory (Vol. 4015). Citeseer.
- Shafer, G. (1976) A mathematical theory of evidence. Princeton : Princeton University Press.
- Sklansky, J. (1978) Image Segmentation and Feature Extraction. *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, n°4, p. 237-247.
- Smarandache, F. (2002) Neutrosophic set—a generalization of the intuitionistic fuzzy set. University of New Mexico. Citeseer.
- Smets, P. (1989) Constructing the Pignistic Probability Function in a Context of

- Uncertainty. *Uncertainty in Artificial Intelligence*, p. 29–40.
- Thomas, C., Ranchin, T., Wald, L. et Chanussot, J. (2008) Synthesis of Multispectral Images to High Spatial Resolution: A Critical Review of Fusion Methods Based on Remote Sensing Physics. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, n°5, p. 1301-1312.
- Tuceryan, M. (1994) Moment-based texture segmentation. *Pattern Recognition Letters*, vol. 15, n°7, p. 659–668.
- Tuceryan, M. et Jain, A. K. (1998) Texture analysis. *The handbook of pattern recognition and computer vision* (Vol. 2, p. 207–248).
- Tuytelaars, T. et Mikolajczyk, K. (2008) Local invariant feature detectors: a survey. *Foundations and Trends in Computer Graphics and Vision*, vol. 3, n°3, p. 177–280.
- Unser, M. et Eden, M. (1990) Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering. *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, n°4, p. 804-815.
- Van De Sande, K. E., Gevers, T. et Snoek, C. G. (2010) Evaluating color descriptors for object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, n°9, p. 1582–1596.
- Voorhees, H. et Poggio, T. (1988) Computing texture boundaries from images. *Nature*, vol. 333, n°6171, p. 364–367.
- Wald, L. (1999) Some terms of reference in data fusion. *IEEE Transactions on Geosciences and Remote Sensing*, vol. 37, n°3, p. 1190–1193.
- Wang, D., Haese-Coat, V., Bruno, A. et Ronsin, J. (1993) Texture Classification and Segmentation Based on Iterative Morphological Decomposition. *Journal of Visual Communication and Image Representation*, vol. 4, n°3, p. 197-214.
- Wu, J., Cui, Z., Sheng, V. S., Zhao, P., Su, D. et Gong, S. (2013) A Comparative Study of SIFT and its Variants. *Measurement Science Review*, vol. 13, n°3, p. 122–131.

- Yager, R. R. (1987) On the Dempster-Shafer framework and new combination rules. *Information Sciences*, vol. 41, n°2, p. 93-137.
- Yen, J. (1986) A reasoning model based on an extended Dempster-Shafer theory. *AAAI*, p. 125–131.
- Zadeh, L. A. (1965) Fuzzy sets. *Information and control*, vol. 8, n°3, p. 338–353.
- Zhang, J., Marszałek, M., Lazebnik, S. et Schmid, C. (2007) Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. *International Journal of Computer Vision*, vol. 73, n°2, p. 213-238.