

ÉCOLE DOCTORALE Mathématiques, Sciences de l'Information et de l'Ingénieur

Laboratoire ICube

THÈSE présentée par :

Bruno BELARTE

soutenue le : 19 septembre 2014

pour obtenir le grade de : **Docteur de l'université de Strasbourg**

Discipline / Spécialité : Informatique

Extraction, analyse et utilisation de relations spatiales entre objets d'intérêt pour une analyse d'images de télédétection guidée par des connaissances du domaine

THÈSE dirigée par :

M. WEMMERT Cédric
Mme. WEBER Christiane

Maître de conférence HDR, Université de Strasbourg
Directrice de recherche, Université de Strasbourg

RAPPORTEURS :

M. LEBBAH Mustapha
M. MAINGUENAUD Michel

Maître de conférence HDR, Université Paris 13
Professeur, INSA Rouen

AUTRES MEMBRES DU JURY :

M. PASSAT Nicolas

Professeur, Université de Reims Champagne-Ardenne

Table des matières

Introduction	1
Partie I	
Contexte des travaux	5
1 Analyse d'images de télédétection	7
1 Approches orientées régions pour l'analyse d'images	8
2 Utilisation de connaissances expertes pour la détection d'objets d'intérêt	19
3 Le CNES, Pléiades et le programme d'accompagnement ORFEO	30
4 Positionnement	33
2 La fouille de données pour l'analyse d'images	35
1 La fouille de données pour aider l'expert	35
2 Utilisation de connaissance dans un processus de classification	41
3 Positionnement	48
Partie II	
Analyse d'images de télédétection guidée par la connaissance	51
3 La logique floue comme moteur de raisonnement	53
1 Logique floue	54
2 La logique floue comme moteur de raisonnement	55
3 Exemple	58
4 État de l'art	61
5 Application à la télédétection	65
Conclusion et contributions	66
4 Utilisation de connaissances expertes pour l'analyse d'images de télédétection	67
1 Représentation des données	68
2 Représentation de la base de connaissances	70
3 Classification des objets d'intérêt	71
4 Extraction d'objets composés	72
5 Application	75
Conclusion et contributions	75
Partie III	
Approche multi-niveaux pour l'apprentissage de règles et l'intégration d'information contextuelle	79
5 Apprentissage de règles de composition pour l'analyse d'images de télédétection à très haute résolution spatiale	81

1	Méthode proposée : Apprentissage de règles de composition	82
2	Apprentissage de règles floues	87
3	Validation expérimentale de la méthode <i>ARC</i>	92
	Conclusion et contributions	96
6	Approche montante-descendante pour la mise à jour de la classification	99
1	Injecter de la connaissance dans une étape de post-traitement	99
2	Proposition : approche montante-descendante	100
3	Expérimentations sur des images Pléiades	102
	Conclusion et contributions	104
7	Validation expérimentale	109
1	Présentation du jeu de données	109
2	Évaluation des paramètres de la méthode <i>ARC</i>	110
3	Approche montante-descendante	115
4	Application sur l'image de Strasbourg	117
5	Complexité algorithmique et temps de calcul	117
	Conclusion	118
	Conclusion	123
	Bibliographie	127
	Annexes	139
	A Jeux de données	139
	B Segmentations associées aux images Pléiades	145

Table des figures

1.1	Images satellitaires à différentes résolutions.	8
1.2	Illustration du principe de l'algorithme LPE. (Najman et Schmitt, 1996)	9
1.3	Visualisation du filtrage et de la segmentation MEANSHIFT sur des données en niveau de gris. (Comaniciu et Meer, 2002a)	11
1.4	Graphe extrait d'une segmentation à trois niveaux. (Inglada et Michel, 2009)	12
1.5	Segmentation multi-résolution. (Gaetano <i>et al.</i> , 2009)	13
1.6	Segmentation multi-résolution. (Gaetano <i>et al.</i> , 2009)	13
1.7	Représentation d'une hiérarchie de segments. (Blaschke, 2003)	14
1.8	Représentation d'une hiérarchie de composantes connexes. (Akçay et Aksoy, 2008)	15
1.9	Exemple de vérité terrain, avec les 7 classes associées. (Akçay et Aksoy, 2008)	17
1.10	Déroulement de l'algorithme optimized probashed. (Derivaux <i>et al.</i> , 2010)	17
1.11	Résultats obtenus par l'algorithme <i>optimized probashed</i> . (Derivaux <i>et al.</i> , 2010)	18
1.12	Extrait d'une ontologie associée à un paysage urbain. (Forestier <i>et al.</i> , 2008)	18
1.13	Résultats obtenus par la méthode de Bacher et Mayer (2005).	20
1.14	Résultats obtenus avec la méthode de Jeon <i>et al.</i> (2002).	20
1.15	Illustration des travaux de Tupin <i>et al.</i> (2002).	21
1.16	Application de la méthode de Poulain (2010) après fusion des détecteurs.	22
1.17	Illustration de la méthode de Lhomme (2005).	23
1.18	Exemple de résultats pour la méthode de Peng <i>et al.</i> (2005).	23
1.19	Illustration de la méthode de Kim et Muller (1999)	24
1.20	Application de la méthode de Poulain (2010) après fusion des détecteurs.	24
1.21	Exemple d'utilisation de la méthode de Awrangjeb <i>et al.</i> (2010)	25
1.22	Illustration de la méthode de Rottensteiner <i>et al.</i> (2005).	26
1.23	Illustration de la méthode de Ziems <i>et al.</i> (2007).	27
1.24	Exemple d'application de la méthode de Gerke et Heipke (2008).	28
1.25	Exemple de mise à jour de SIG avec la méthode de Qiaoping et Couloigner (2004).	29
1.26	Bandes spectrales du capteur PLEIADES.	31
2.1	Les huit relations de la table 2.1 illustrées par des exemples. (Renz et Nebel, 1999)	44
2.2	Huit contraintes sur la relation directe entre deux points. (Papadias et Egenhofer, 1997)	47
2.3	Exemple d'un réseau de trois contraintes et cinq points. (Papadias et Egenhofer, 1997)	48
3.1	Courbes associées aux fonctions gaussienne, triangulaire, trapézoïdale et booléenne.	56
3.2	Schéma du processus de raisonnement basé sur la logique floue.	57
3.3	Fonctions d'appartenance correspondant aux variables linguistiques <i>vitesse</i> , <i>distance</i> et <i>comportement</i>	59
3.4	Déroulement de l'algorithme de raisonnement.	61
3.5	Résultat comparatif des deux opérateurs de défloutage.	62
3.6	Exemple de fonctions et de règles décrit par un fichier XML.	66
4.1	Différentes étapes de création du graphe d'adjacence.	68

4.2	Extrait d'une hiérarchie de concepts et des relations associées.	72
4.3	Illustration du résultat des algorithmes 4 et 5 sur une imagerie issue de l'image PLEIADES	75
4.4	Relation entre objets d'intérêt et segments.	76
5.1	Entrées et sortie de la méthode <i>ARC</i>	82
5.2	Organigramme résumant le processus d'apprentissage.	85
5.3	La nouvelle hiérarchie de concepts après apprentissage.	85
5.4	Différentes étapes de l'algorithme d'apprentissage.	86
5.5	Arbres binaires de partition produits par les trois CAH.	89
5.6	Les fonctions d'appartenance sur les trois propriétés après l'apprentissage.	90
5.7	Arbre binaire de partition produits par la CAH pour l'apprentissage des règles.	90
5.8	Application de la méthode sur une image QUICKBIRD.	94
5.9	Résultats sur les deux images PLEIADES.	95
5.10	Deux exemples de hiérarchies de concepts appris par la méthode <i>ARC</i>	96
6.1	Entrées et sorties de la chaîne de traitement.	100
6.2	L'approche montante-descendante.	101
6.3	L'approche montante-descendante avec les trois niveaux utilisés dans le cas de cette étude.	102
6.4	Résultats obtenus après les différentes étapes de l'approche montante-descendante pour l'image de <i>Cronenbourg</i>	106
6.5	Résultats obtenus après les différentes étapes de l'approche montante-descendante pour l'image du <i>quartier des Quinze</i>	107
7.1	Influence du taux d'erreur et des exemples choisis.	111
7.2	Exemple du processus d'extraction des segments.	111
7.3	Analyse des différentes valeurs du paramètre d'affinage.	113
7.4	Analyse des différentes valeurs du critère de Ward pour l'apprentissage des fon- ctions d'appartenance et pour l'apprentissage des règles.	114
7.5	Illustration du résultat obtenu avec le jeu de paramètres identifié dans cette section.	116
7.6	Base de règles apprise par la méthode <i>ARC</i> sur la région d'intérêt.	120
7.7	Résultats des différentes étapes de la chaîne de traitement.	121
7.8	Résultats de la chaîne de traitement sur la segmentation de travail.	122
A.1	Image QUICKBIRD utilisée par la validation expérimentale.	139
A.2	Extraits de l'image PLEIADES.	140
A.3	Image PLEIADES de Strasbourg (quatre bandes).	141
A.4	Image PLEIADES de Strasbourg (trois bandes).	142
A.5	Région d'intérêt extrait de l'image A.4.	143
B.1	Extraits des segmentations de la zone de Strasbourg.	146

Liste des tableaux

1.1	Méthodes d'extraction de routes classées en fonction des images sur lesquelles elles sont appliquées.	29
1.2	Méthodes d'extraction de bâtiments classées en fonctions des images sur lesquelles elles sont appliquées.	30
1.3	Les principales caractéristiques de la composante optique PLEIADES.	32
2.1	Huit relations de la méthode RCC-8. (Renz et Nebel, 1999)	44
2.2	Opérateurs applicables sur l'ensemble des relations RCC-8.	44
2.3	Six relations de base de RCC-8 encodées en logique modale.	46
2.4	Liste non exhaustive de symboles utilisés par les logiques de description.	46
2.5	Six relations de base de RCC-8 encodées en logique de description.	47
3.1	Transformation des variables numériques en variables floues.	60
3.2	Résultats du défloutage.	60
5.1	Tableau présentant les valeurs de chaque segment pour chacune des trois propriétés.	89
5.2	Les exemples après redéfinition.	90
5.3	Propriétés des différents jeux de données utilisés.	91
5.4	Résultats pour différents jeux de données.	92
5.5	Évaluation des résultats sur les images PLEIADES.	95
6.1	Les différentes phases des deux itérations successives de l'approche montante-descendante.	103
6.2	Les trois règles de composition utilisées pour l'approche montante-descendante.	103
6.3	Résultats de l'extraction des pavillons sur l'image de <i>Cronenbourg</i>	104
6.4	Résultats de l'extraction des pavillons sur l'image du <i>quartier des Quinze</i>	104
7.1	Présentation des valeurs initiales des différents paramètres de la méthode <i>ARC</i>	110
7.2	Résultats de l'analyse sur l'influence du paramètre d'affinage.	112
7.3	Évaluation des résultats en fonction des différentes propriétés utilisées.	114
7.4	Présentation des valeurs finales des différents paramètres de la méthode <i>ARC</i> qui ont été identifiées dans cette section.	115
7.5	Les différentes phases des deux itérations successives de l'approche montante-descendante.	115
7.6	Les trois règles de composition utilisées pour l'approche montante-descendante.	115
7.7	Résultats de l'extraction des pavillons sur l'image du <i>quartier des Quinze</i> extrait de l'image PLEIADES en fausses couleurs.	117

Remerciements

Avant de présenter les travaux réalisés au cours de ma thèse, je tiens à remercier toutes les personnes qui ont rendu cette expérience possible.

Je voudrais tout particulièrement remercier Cédric Wemmert d'avoir co-dirigé cette thèse. Notre collaboration a débuté voilà cinq ans déjà lorsqu'il a accepté de m'encadrer pour mon projet d'étude lors de ma première année de master, et cette collaboration n'a cessé depuis. Il a su me guider tout en me laissant la liberté nécessaire pour mener mes travaux de recherche, et pour cela je lui suis reconnaissant. Je tiens à remercier tout aussi chaleureusement Christiane Weber d'avoir co-dirigé cette thèse. Nos nombreux échanges m'ont permis de mieux comprendre les problématiques et les enjeux de la télédétection.

Merci à Manuel Grizonnet d'avoir encadrer cette thèse et de m'avoir apporter toute son expertise technique et ses précieux conseils lors des phases de conception et de développement. Mes différents séjours au CNES dans le département DCT/SI/AP ont toujours été profitables. Enfin, je tiens à remercier Germain Forestier d'avoir encadré cette thèse. Prendre un train en route n'est pas évident, son intérêt pour mes travaux et son regard neuf m'ont permis de pousser plus loin mon raisonnement sur un sujet aussi passionnant que complexe.

Ces recherches ont vu le jour grâce au soutien financier du Centre National d'Étude Spatiales et de la Région Alsace. À ce titre je tiens à remercier Pierre Gançarski pour avoir trouvé ces financements et rendu cette thèse possible.

Merci à Michel Mainguenaud et Mustapha Lebbah d'avoir rapporté cette thèse, ainsi qu'à Nicolas Passat d'avoir pris part à l'évaluation de cette thèse et présidé le jury de soutenance.

Mes travaux ont été réalisés au sein de l'équipe BFO du laboratoire ICube. Je tiens à remercier l'ensemble du personnel de l'équipe qui m'a accompagné depuis quatre ans. Merci à Camille et François d'avoir partagé leur savoir, ainsi qu'à Ogier, Matthieu et Emmanuel d'avoir accompagné dans la bonne humeur le début de cette thèse. Merci à Clément, Andrés, Karim, Igor, Carlos, Manuela de m'avoir supporté et encouragé lors de la dernière année et d'avoir contribué à l'aboutissement de ces trois ans de travaux.

Merci à mes parents de m'avoir laissé la liberté de faire des erreurs. Long est le chemin qui sépare un jeune bachelier d'un doctorat en informatique, chemin qu'il m'aurait été impossible d'arpenter sans le soutien et la confiance dont ils ont fait preuve.

Enfin je ne pourrais pas terminer ces remerciements sans adresser un grand merci à Aurélie, ma compagne depuis sept ans. Merci de m'avoir supporté, au sens propre comme au sens figuré, et de m'avoir fait confiance toutes ces années.

Introduction

Les années 1980 ont vu apparaître les premiers capteurs aéroportés capables d’acquérir des images d’observation de la terre. Nous considérons aujourd’hui ces images comme étant à basse résolution spatiale¹ (résolution supérieure à 30 mètres). Les années 1980 et 1990 ont vu apparaître progressivement des capteurs à moyenne résolution spatiale (entre 30 et 5 mètres) et des capteurs à haute résolution spatiale (entre 5 et 1 mètre). Ceux-ci ont été suivis dans les années 2000 par les capteurs à très haute résolution spatiale (résolution inférieure au mètre). Ces progrès sont significatifs puisque les détails observables sur une image ont augmenté considérablement à chaque étape. Une image à basse résolution permet par exemple d’avoir une vue globale d’un paysage, en mettant en avant les parcelles agricoles ou les tissus urbains, tandis que dans une image à haute résolution, il est possible d’observer des bâtiments individuels ou les réseaux routiers. Le niveau de détail d’une image est d’autant plus fin dans une image à très haute résolution spatiale (THRS), dans laquelle il devient même possible de discerner les voitures sur la route.

Pour analyser les premières images issues de la télédétection, les méthodes de traitement d’images classiques étaient employées. Dans ce type de méthodes, dites *orientées pixels*, seules les valeurs radiométriques du pixel (éventuellement sa position absolue) étaient prises en compte. Ceci permettait une analyse thématique basée sur la radiométrie. Ce type d’approche était suffisant dans le cas des images basse ou moyenne résolution, mais elles ont montré leurs limites lorsqu’elles ont été appliquées à des images à haute ou très haute résolution. De là a émergé le besoin d’utiliser les méthodes d’analyse d’images dites *orientées régions*, apparues dans la littérature dédiée dans les années 1980. Dans ce type d’approches, les pixels sont d’abord regroupés en composantes connexes homogènes d’après un critère radiométrique. Ces méthodes, appelées *segmentation*, permettent de découper une image en régions, chaque région étant définie par sa signature spectrale à travers des attributs calculables (par exemple valeur moyenne de chaque bande, écart-type, indices de texture, d’homogénéité) mais aussi par des attributs de forme (par exemple taille, élongation, périmètre). Ceci permet d’enrichir considérablement la description des objets dans les méthodes de fouille de données et de réfléchir en termes d’*objets d’intérêt* plutôt que pixels. Les méthodes de segmentation sont aujourd’hui communément utilisées pour l’analyse d’images de télédétection à haute ou très haute résolution spatiale, notamment au sein de l’équipe BFO² qui encadre cette thèse.

Le bond technologique provoqué par l’arrivée des capteurs à très haute résolution spatiale a entraîné des complications dans le domaine de l’analyse d’images de télédétection. Les images obtenues sont plus précises et de plus grande taille, une image représentant jusqu’à plusieurs milliards de pixels. Avec des capacités d’acquisition quotidienne, les capteurs peuvent fournir des images de cette qualité plusieurs fois par semaine. Le volume de données à traiter est devenu considérable. De plus, la finesse des images THRS est telle que des petits objets dans la scène peuvent venir brouter l’image (par exemple, une fenêtre de toit ou une parabole sur un pavillon). Ceci nuit à l’homogénéité des objets d’intérêt pour l’expert. De part ce volume et ce bruit, la masse de données à traiter est devenue beaucoup trop importante et trop complexe pour qu’un expert ou groupe d’experts photointerprètes puissent analyser manuellement toutes les données

1. Par résolution spatiale, nous entendons *définition de la qualité de l’image*. Nous pouvons voir la résolution comme le côté du carré représentant un pixel de l’image. Ceci est à opposer à la résolution *spectrale*, qui est définie par le nombre et la largeur des bandes radiométriques.

2. Bioinformatique théorique, Fouille de données et Optimisation stochastique

produites dans un temps réaliste. L'expert dispose de connaissances avancées sur le domaine d'application. Cette information de haut niveau sur ce que cherche l'expert dans une image de télédétection est d'un intérêt particulier puisqu'elle permet de restreindre ou de guider l'analyse de l'image. De nombreuses méthodes basées sur la connaissance sont apparues dans la littérature ces dernières années.

Dans le but de traiter efficacement les images THRS, il devient nécessaire de disposer de méthodes entièrement automatiques prenant en compte la connaissance de l'expert, que ce soit à travers l'utilisation d'exemples, le réglage de paramètres ou l'évaluation de la qualité des résultats. Au cours des dernières années, le laboratoire de géographie de l'Université de Strasbourg³ a travaillé à la création d'un dictionnaire d'objets urbains. Dans ce dictionnaire, les objets sont à la fois décrits par des contraintes sur des attributs propres (taille, signature spectrale) et par des contraintes sur leurs relations spatiales (adjacence, composition). Pour pouvoir utiliser la connaissance d'un expert, il faut pouvoir la formaliser rigoureusement. Ceci est élémentaire lorsque nous parlons d'attributs propres à un objet mais devient plus compliqué lorsque nous nous intéressons aux relations spatiales entre ces objets.

Cette thèse s'inscrit dans une collaboration débutée en 2000 entre les laboratoires d'informatique⁴ et de géographie de l'Université de Strasbourg. Les travaux menés conjointement portent sur l'application de méthodes de fouille de données sur des données issues de la télédétection. La présente thèse, cofinancée par le CNES⁵, s'intéresse à l'utilisation de connaissances, plus particulièrement d'information contextuelle, dans un processus d'analyse d'image de télédétection. Ceci passe par l'apprentissage de règles de composition floues et l'injection de connaissances de haut niveau sémantique dans un processus hiérarchique de classification d'objets d'intérêt multi-niveaux issus de la segmentation.

Problématique

Dans un premier temps, nous cherchons à utiliser la connaissance de l'expert pour identifier et classer des objets d'intérêt dans une image segmentée par une approche simple. Nous nous intéressons tout particulièrement à l'apport d'informations contextuelles (voisinage, appartenance) pour améliorer la classification. Nos premiers travaux portent sur la formalisation de la connaissance spatiale pour une intégration dans un processus d'analyse d'images de télédétection orientée région.

Cependant, nous verrons dans le chapitre 4 que l'utilisation de la connaissance, qu'elle soit spatiale ou non, est difficilement utilisable dans une analyse d'images THRS. En effet, la segmentation d'images THRS ne permet pas d'identifier les objets d'intérêt pour l'expert, alors que ceux-ci sont pourtant parfaitement discernables dans l'image. Le problème vient du phénomène de sursegmentation, particulièrement présent dans la segmentation d'images THRS. Le problème de la sursegmentation rejoint le problème plus large de la différence entre la représentation des objets d'intérêt tels que décrits par l'expert et la représentation des objets d'intérêt dans l'image. Ainsi, la principale problématique de cette thèse est de chercher un lien entre ces deux représentations de manière à pouvoir intégrer efficacement un raisonnement basé sur la connaissance de l'expert dans un processus d'analyse d'images THRS.

Pour remédier à ce problème, nous proposons une approche de classification hiérarchique dans le but d'apprendre comment des objets d'intérêt sont décomposés dans une image segmentée. Nous présentons notre méthode dans le chapitre 5. Nous verrons dans ce chapitre que

3. Laboratoire Image Ville Environnement – UMR 7362

4. Laboratoire des sciences de l'Ingénieur, de l'Informatique et de l'Imagerie – UMR 7357

5. Centre National d'Études Spatiales

l'information contextuelle n'apporte que peu de choses dans l'apprentissage de cette règle de composition. Pour intégrer au mieux l'information contextuelle, nous proposons dans le chapitre 6 une approche hiérarchique itérative cherchant à ajouter de l'information de haut niveau dans la hiérarchie, dans le but de mettre à jour la classification à bas niveau.

Axes de recherche

Les travaux présentés dans cette thèse s'orientent autour de deux axes de recherche. Le premier axe porte sur la recherche d'un lien entre la représentation de la connaissance de l'expert et la segmentation d'une image dans le domaine de la télédétection. Le second axe porte sur l'intégration de l'information contextuelle dans un processus d'analyse d'image de télédétection.

Lien entre la connaissance et la segmentation. Au cours de nos travaux, nous allons observer qu'il existe un fossé entre la description d'un objet d'intérêt fait par un expert et la représentation de ce même objet dans une image segmentée. En effet, l'expert est capable de décrire un objet d'intérêt avec une large variété de descripteurs, qu'il s'agisse de descripteurs génériques (par exemple la signature spectrale) ou de descripteurs liés au domaine d'application (par exemple l'emprise au sol). Or un algorithme de segmentation ne tient pas compte de la sémantique sous-jacente à l'image qu'il traite, il va découper l'image en régions radiométriquement homogènes. Ceci devient particulièrement problématique dans le traitement des images de télédétection à très haute résolution spatiale, dans lesquelles un objet d'intérêt du plus bas niveau (un pavillon par exemple) peut se retrouver découpé en une ou plusieurs régions à cause de la complexité de sa forme ou de son exposition au soleil. Le principal problème lié à cette différence de représentation réside dans le fait que la représentation n'est pas duale (domaine d'expertise / image) mais multiple. En effet, un même objet dans le domaine d'expertise va être représenté différemment selon le type d'image (panchromatique, multispectrale, hyperspectrale, radar), l'algorithme de segmentation utilisé ou encore les paramètres de l'algorithme de segmentation. Il n'est donc pas possible que l'expert donne une description de la composition d'un objet d'intérêt dans une segmentation, puisque cette description est différente pour chaque image et chaque segmentation.

Pour résoudre ce problème de représentation, nous proposons un algorithme d'apprentissage qui, en partant d'exemples fournis par un expert, est capable d'apprendre une description en fonction de leurs compositions dans une image segmentée.

Intégration de l'information contextuelle. Dans un paysage urbain, les objets d'intérêt sont en grande partie définis par leur contexte. Nous verrons dans le chapitre 2 que la littérature sur la formalisation et l'utilisation des relations spatiales est riche. Ce type d'information est extrêmement pertinent pour l'expert, bien qu'il ne soit que peu utilisé dans le domaine de l'analyse automatique d'images de télédétection. L'information contextuelle peut être de deux types. Premièrement nous avons les relations spatiales entre objets d'un même niveau. Il s'agit des relations d'adjacence ou de distance par exemple. Deuxièmement, nous avons les relations entre objets de différents niveaux. Il s'agit de la composition, la superposition totale ou partielle, ou de l'inclusion par exemple. Ces deux types d'informations contextuelles peuvent être combinés pour créer une hiérarchie d'objets liés spatialement et contextuellement.

Pour intégrer l'information contextuelle dans un processus de raisonnement, nous proposons une approche hiérarchique qui cherche dans un premier temps à extraire des objets composés de haut niveau à partir d'une classification et de la connaissance de l'expert, et dans un second temps

qui utilise l'information extraite pour mettre à jour une classification à bas niveau. Ce processus hiérarchique n'est basé que sur l'extraction et la réutilisation d'informations contextuelles.

Structure du document

Ce document est découpé en trois parties, chacune composée de plusieurs chapitres. Nous présentons ici le contenu de chacune de ces parties.

Première partie : Contexte des travaux

Dans cette partie nous abordons les deux thématiques portées par cette thèse. Le chapitre 1 présente le cadre de l'analyse d'images, plus particulièrement de l'analyse d'images de télédétection. Nous présentons différents types de méthodes de segmentation et nous détaillons l'utilisation de la connaissance de l'expert pour guider l'interprétation d'images de télédétection. Nous évoquons les problèmes liées aux images à très haute résolution spatiale. Le chapitre 2 présente le cadre de la fouille de données et de l'utilisation de connaissances pour guider des algorithmes de classification. Nous nous intéressons tout particulièrement à la connaissance portant sur les relations spatiales et leur formalisation. Nous évoquons l'application de méthodes de fouilles de données pour l'analyse d'images de télédétection.

Deuxième partie : Analyse d'images de télédétection guidée par la connaissance

Dans cette partie nous présentons la méthode d'analyse d'images de télédétection que nous avons mis en place au cours de cette thèse. Pour gérer l'imprécision relative à la segmentation et à l'analyse d'images, nous choisissons d'utiliser la logique floue comme formalisation de la connaissance et comme moteur de raisonnement. Le chapitre 3 présente les notions de base de la logique floue ainsi qu'un bref état de l'art sur l'utilisation de la logique floue moteur d'inférence et sur l'apprentissage de règles floues. Le chapitre 4 présente la méthode d'analyse d'images proposée. Nous présentons la méthode de classification basée sur la logique floue et la méthode d'extraction des objets composés que nous avons proposées. Nous concluons cette partie en évoquant les limites d'une telle approche pour l'analyse d'images à très haute résolution spatiale.

Troisième partie : Approche multi niveaux pour l'apprentissage de règles et l'intégration d'information contextuelle

Dans cette partie nous présentons une chaîne de traitement pour résoudre les problèmes identifiés dans la seconde partie. Le chapitre 5 présentera un algorithme d'apprentissage de règles de composition guidé par des exemples fournis par l'expert dans le but de faire le lien entre les objets d'intérêt et une segmentation. Les différentes étapes sont détaillées et validées expérimentalement, avant de tester la méthode sur des images de télédétection. Le chapitre 6 présente l'approche *montante-descendante* que nous avons proposée pour extraire de l'information contextuelle de haut niveau à partir de la classification d'une segmentation et l'utilisation de cette information de haut niveau pour mettre à jour la classification à bas niveau. Cette méthode est utilisée à la suite de l'algorithme d'apprentissage de règles de composition pour améliorer les résultats produits par la chaîne de traitement. Le chapitre 7 présente une validation expérimentale détaillée des différentes étapes de la chaîne de traitement et illustre l'application de la méthode sur divers jeux de données.

Première partie

Contexte des travaux

★ Résumé ★

Dans cette partie, nous proposons une étude bibliographique des deux axes de recherche qui composent cette thèse. Dans le premier chapitre nous présentons l'analyse d'images et son application à la télédétection. Nous décrivons ensuite l'apport de la connaissance de l'expert et de son intégration dans un processus de raisonnement. Nous parlons des problématiques liées aux images à très haute résolution spatiale, tout particulièrement les images PLEIADES, et le programme d'accompagnement ORFEO. Dans le second chapitre, nous présentons la fouilles de données, et plus précisément des méthodes de classification supervisées et non supervisées. Nous évoquons l'application de ces méthodes sur des problèmes de télédétection. Nous parlons ensuite de l'utilisation de connaissances dans les algorithmes de fouille de données. Nous étudions tout particulièrement la connaissance spatiale et les formalismes associés.



Analyse d'images de télédétection

Sommaire

1	Approches orientées régions pour l'analyse d'images	8
1.1	Approches simples	8
1.2	Approches complexes	10
1.3	Approches guidées par la connaissance	16
2	Utilisation de connaissances expertes pour la détection d'objets d'intérêt	19
2.1	Détection de routes	19
2.2	Détection de bâtiments	22
2.3	Utilisation de système d'information géographique	26
2.4	Synthèse	28
3	Le CNES, Pléiades et le programme d'accompagnement ORFEO	30
3.1	Les images Pléiades	30
3.2	Le programme d'accompagnement ORFEO	32
4	Positionnement	33

Avec l'arrivée des premières images de télédétection issues de satellites orbitaux est apparu, pour le spécialiste, le besoin d'analyser efficacement cette nouvelle source de données.

Les premiers satellites offraient des images de basse résolution, un pixel représentant un carré de 80 mètres de côté environ (Landsat-1, 1972). Les techniques de segmentation usuelles en analyse d'images, utilisant une approche basée pixel, furent employées pour segmenter ces images. Ces techniques visaient à extraire et analyser les informations inhérentes aux pixels : l'information spectrale (extraction d'histogramme de couleurs, de niveau de gris, bande infrarouge, ...) et l'information spatiale (pixels voisins, position, ...). Les années 80 ont vu émerger des satellites capables de produire des images de Haute Résolution (HR), dont la résolution était comprise entre 30 et 10 mètres (Landsat-4, 1982 ; SPOT3, 1993), tandis que dans les années 2000 sont apparus les satellites Très Haute Résolution (THR) dont la résolution spatiale est inférieure à 5 mètres (QUICKBIRD, 2001 ; PLEIADES, 2011) Les satellites THR permettent actuellement d'obtenir des images dont la résolution va jusqu'à 0.5 mètres par pixel sur la bande panchromatique. Ces images offrent un niveau de détail bien supérieur aux images haute résolution. Sur une image représentant un paysage urbain, il est désormais possible de distinguer plus finement la plupart des objets d'intérêt (maisons individuelles, arbres, ...), détails impossibles à observer avant.

La figure 1.1 illustre ces différences de résolutions spatiales. Sur l'image 1.1(a) nous distinguons plusieurs groupes de pixels plus clairs, sur l'image 1.1(b) nous pouvons distinguer des groupes de bâtiments, sur l'image 1.1(c) nous pouvons identifier les bâtiments et les routes.

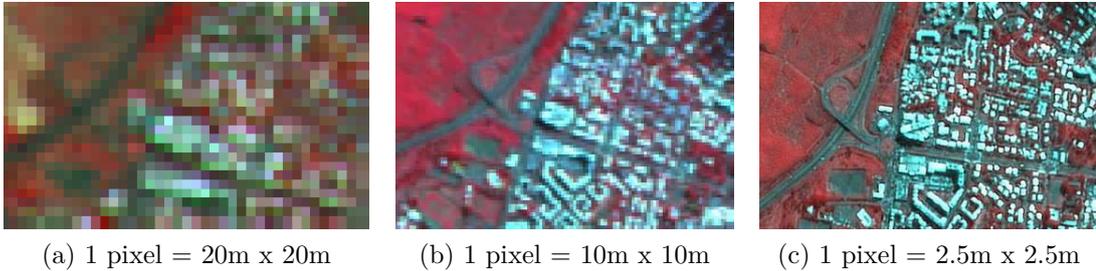


FIGURE 1.1 – Images satellitaires à différentes résolutions. 1.1(a) et 1.1(b) font partie de la catégorie des images HR tandis que 1.1(c) est une image THR.

Ce bond technologique a cependant entraîné certaines difficultés. Nous pouvons observer deux problèmes majeurs. Premièrement, pour deux images représentant le même terrain, l'image THR sera composée de beaucoup plus de pixels, augmentant considérablement la taille de l'image, donc *a fortiori* le temps de calcul. Deuxièmement, la résolution étant beaucoup plus fine, nous voyons apparaître une certaine quantité de bruit qui vient perturber l'analyse (excès de détails, véhicules sur les routes, ...). Les segmentations reposant sur une approche basée pixel montrent clairement leurs limites ici, puisque le bruit vient parasiter le résultat de la segmentation et de l'analyse qui en découle.

C'est dans ce contexte qu'ont été développés les modèles de segmentation reposant sur une approche basée région. Les informations inhérentes au pixel sont utilisées pour regrouper les pixels semblables en régions homogènes, l'approche basée pixel servant de pré-traitement à la segmentation. L'objet d'intérêt n'est plus le pixel mais la région. Les différentes techniques présentées ici sont toutes des approches orientées régions, et nous utilisons indifféremment les termes région et objet. De plus, nous définissons une image multi-spectrale comme une image couleur (plusieurs bandes) et une image panchromatique comme une image en niveau de gris (une seule bande spectrale). Les images panchromatiques ont une résolution supérieure aux images multi-spectrales associées.

1 Approches orientées régions pour l'analyse d'images

1.1 Approches simples

Par approche simple, nous entendons approche basée sur une seule image à une résolution donnée. Une image peut être en niveau de gris, c'est à dire une seule bande dans laquelle les pixels prennent une valeur entre 0 et la valeur maximale (cette valeur dépend de l'encodage des pixels). Une image couleur est la combinaison de plusieurs images en niveaux de gris (typiquement une image multispectrale à quatre bandes, représentant respectivement le bleu, le vert, le rouge et le proche infrarouge).

1.1.1 Ligne de partage des eaux

L'algorithme de Ligne de Partage des Eaux (abrégé LPE), développé par Vincent et Soille (1991), est une technique de segmentation basée sur la morphologie mathématique. Initialement, l'algorithme considère une image en niveau de gris, mais a été étendu pour gérer les images couleurs. Cette image est vue comme une carte de hauteur, le niveau de gris d'un pixel représentant sa *hauteur* sur la carte. L'algorithme va *faire monter le niveau de l'eau*. À chaque fois que le niveau atteint un minimum local, un *bassin versant* est créé. Lorsque deux bassins se rencontrent, l'algorithme crée une ligne entre ces deux bassins pour en empêcher la fusion. Ces lignes de partage des eaux représentent à la fin de l'algorithme les contours de chaque région. Dans le cas d'une image en couleur, l'algorithme s'applique indépendamment sur chaque bande, puis les résultats sont fusionnés. La figure 1.2 illustre le principe de l'algorithme sur une bande.

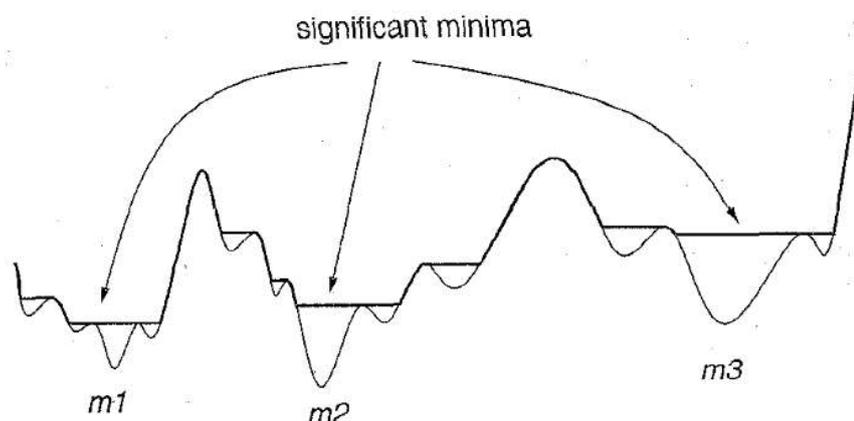


FIGURE 1.2 – Illustration du principe de l'algorithme LPE. Les traits horizontaux correspondent au niveau de l'eau. Les maxima locaux sont ainsi détectés et des lignes de partage sont créées. (Najman et Schmitt, 1996)

Le principal inconvénient de la LPE est qu'elle crée un nombre trop important de régions. L'image est sur-segmentée. Ceci est d'autant plus vrai avec les images THR, qui regorgent de détails et de bruit pouvant être considérés comme des minima locaux.

Pour pallier ce problème, Chen *et al.* (2005) utilisent la LPE comme pré-traitement dans un algorithme de croissance de régions. À chaque étape de la LPE, si deux régions adjacentes minimisent une fonction d'hétérogénéité (inhérente à l'image ou au type de segmentation), ces deux régions sont fusionnées, et à l'inverse si une région maximise cette fonction, la région est séparée en deux. Cette technique réduit en effet la sur-segmentation, mais a l'inconvénient de propager les éventuelles erreurs de la LPE.

1.1.2 MeanShift

L'algorithme de segmentation MeanShift est basé sur le filtre du même nom, décrit par Comaniciu et Meer (1997) et Comaniciu et Meer (2002a). Le principe du filtre MeanShift repose sur la correction du bruit blanc Gaussien. On appelle bruit blanc Gaussien une perturbation dont la densité de probabilité de l'amplitude suit une loi gaussienne centrée. Cela signifie que la probabilité pour que la perturbation soit d'une amplitude x est de :

$$P(x) = \frac{1}{\sigma \cdot \sqrt{2\pi}} \cdot \exp\left(-\frac{x^2}{2 \cdot \sigma^2}\right)$$

où σ est un paramètre qui définit la forme de la courbe (l'écart-type de la gaussienne).

L'espace des données est composé des pixels, représentés par un histogramme de niveaux de gris. Dans le cas d'une image couleur (multi-spectrale à trois bandes : rouge, vert, bleu par exemple), il ne s'agit pas d'appliquer l'algorithme sur une courbe, mais sur une hypersurface gaussienne à trois dimensions (correspondant aux trois bandes). Il est alors possible de considérer des informations supplémentaires (une autre bande spectrale, par exemple), il faut pour cela augmenter le nombre de dimensions de l'hypersurface. Le principe du filtre Mean Shift est de faire converger un point vers le maximum local le plus proche. Le voisinage d'un pixel comprend tous les pixels dont la distance au pixel de référence est inférieure à un seuil donné en paramètre.

Le filtre regroupe les pixels en bassins d'attraction autour d'un barycentre. Pour finir la segmentation de l'image, l'algorithme va regrouper tous les barycentres qui sont proches, en dessous d'un certain seuil. Chaque pixel de l'image se voit attribuer une étiquette correspondant à l'isobarycentre le plus proche. Sur l'image segmentée, les pixels adjacents ayant la même étiquette font partie d'une même région. La figure 1.3 illustre le déroulement de l'algorithme de segmentation.

L'avantage de cette technique est qu'elle permet une classification de l'espace de données. En effet, les pixels sont étiquetés par l'isobarycentre de la classe à laquelle ils appartiennent. En utilisant uniquement un histogramme de niveaux de gris (une seule bande), il est possible de mettre en avant certains objets d'intérêt, mais les objets dont l'information spectrale est semblable (eau et ombre, par exemple) seront regroupés sous une même étiquette.

1.2 Approches complexes

Par approche complexe, nous entendons approche basée sur une ou plusieurs images, capturées à une ou plusieurs résolutions. Nous pouvons noter que malgré l'utilisation de termes différents (multi-échelle, multi-niveau, hiérarchique, ...) les techniques se recoupent souvent.

1.2.1 Approches multi-échelle

Une segmentation basée sur une approche multi-échelle prend en compte une même image à différents niveaux. C'est notamment le cas des travaux de [Inglada et Michel \(2009\)](#). Les auteurs définissent un niveau en fonction de la taille minimale des éléments structurants d'une image. Leur algorithme se base sur une image panchromatique, à laquelle ils appliquent deux fonctions conçues à partir d'opérateurs morphologiques par reconstruction sur une métrique géodésique. Ces deux fonctions sont utilisées pour mettre en avant les structures concaves et convexes de l'image. La fonction convexe (respectivement concave) donne la probabilité qu'un pixel donné fasse parti d'une structure plus claire (respectivement plus foncée) que l'arrière plan, dont la taille est plus petite que la taille minimale imposée aux éléments structurants d'un niveau donné. Ces deux fonctions sont définies mathématiquement comme suit :

Fonction convexe : $\mu_N(I) = I - \gamma_N^*(I)$

Fonction concave : $\eta_N(I) = \phi_N^*(I) - I$

où I est l'image en entrée, N la taille minimale, γ l'opérateur d'ouverture par reconstruction, ϕ l'opérateur de fermeture par reconstruction. À partir de ces deux fonctions, ils définissent la fonction de mise à niveau (leveling function) comme suit :

Fonction de mise à niveau : $\psi_N(I) = \begin{cases} \gamma_N^*(I) & : \mu_N(I) > \eta_N(I) \\ \phi_N^*(I) & : \mu_N(I) < \eta_N(I) \\ I & : \mu_N(I) = \eta_N(I) \end{cases}$

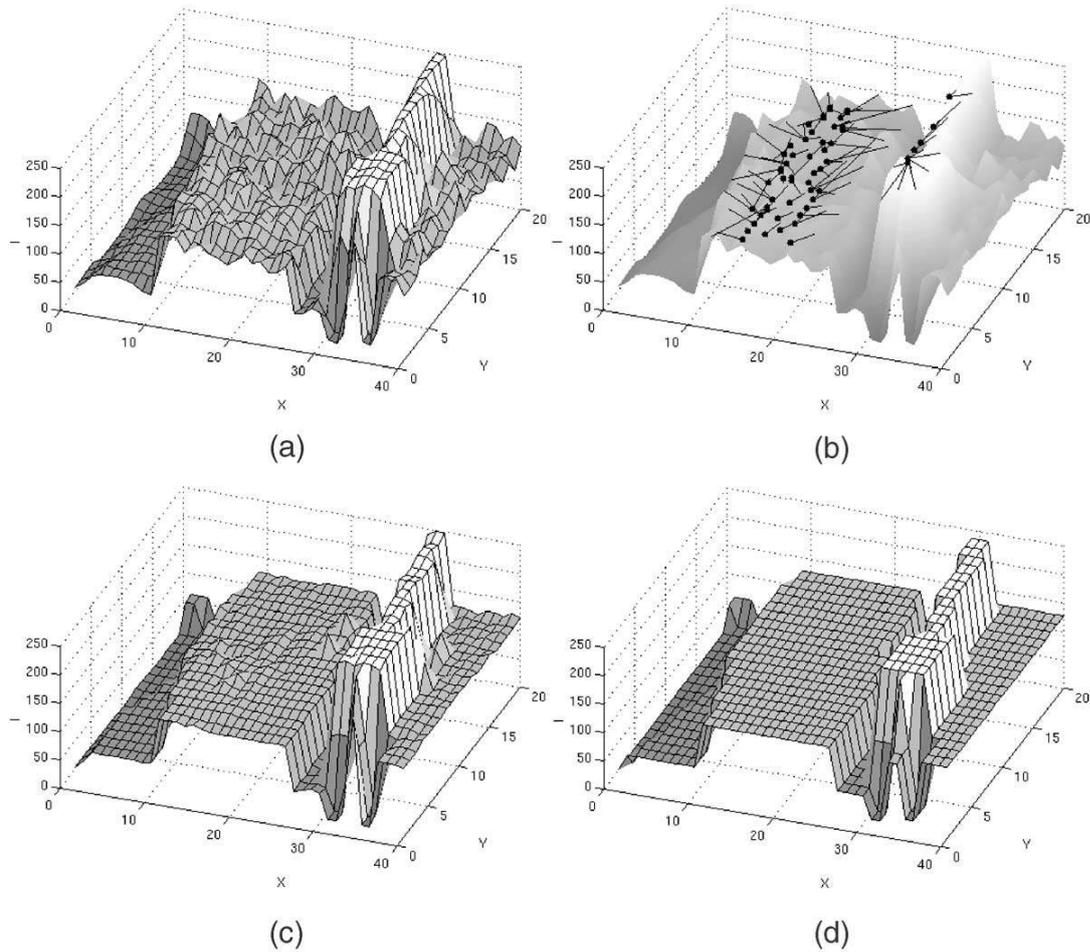


FIGURE 1.3 – Visualisation du filtrage et de la segmentation MEANSHIFT sur des données en niveau de gris. a) Hypersurface gaussienne en entrée, b) les points représentent les barycentres, les traits leurs chemins, c) Résultat du filtrage, d) Résultat de segmentation. (Comaniciu et Meer, 2002a)

À partir de ces outils mathématiques, ils définissent la méthode décrite par l'algorithme 1.

Le résultat obtenu permet de construire un graphe qui met en relation les éléments structurants des différents niveaux (chevauchement, ...). Ce graphe est ensuite analysé à l'aide de la méthode RCC8, décrit dans le chapitre 2 section 2.2.2. Un exemple d'un tel graphe est donnée dans la figure 1.4. Nous pouvons y voir une hiérarchie d'objets structurants à trois niveaux. Chaque niveau contient les segments extraits par les fonctions concaves et convexes (représentées par deux couleurs distinctes par niveau).

Algorithme 1: Segmentation par fonction de mise à niveau (Inglada et Michel, 2009)

pour tous les niveaux i faire

 Segmenter l'image I_i avec la fonction concave

 Segmenter l'image I_i avec la fonction convexe

 Créer l'image $I_{i+1} = \psi_N(I_i)$ à l'aide de la fonction de mise à niveau

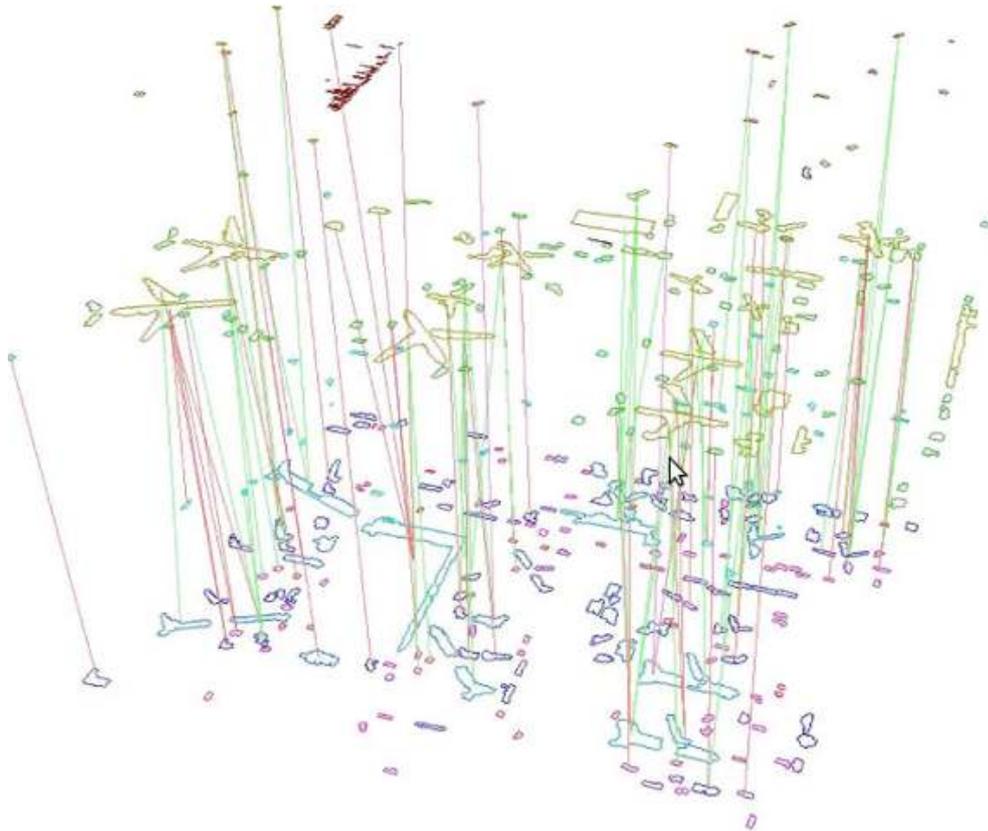


FIGURE 1.4 – Graphe extrait d'une segmentation à trois niveaux. (Inglada et Michel, 2009)

1.2.2 Approches multi-résolution

Le principe d'une segmentation multi-résolution repose sur l'analyse d'une même image capturée à plusieurs résolutions différentes. Gaetano *et al.* (2009) utilisent deux résolutions pour une même image. Premièrement, une image THR panchromatique d'une résolution d'un mètre par pixel, soit une image d'une qualité très fine. Deuxièmement, une image THR multi-spectrale d'une résolution de 4 mètres par pixel. Les deux images sont traitées de manière différentes. Les résultats sont ensuite superposés et corrélés avant d'être analysés.

D'abord l'image panchromatique est segmentée à l'aide de Tree Structured Markov Random Field (TS-MRF). Le principe des TS-MRF (décrit par D'Elia *et al.* (2003)) repose sur la construction d'un arbre binaire. L'image initiale est segmentée en deux classes, puis chaque classe

est segmentée en deux jusqu'à atteindre un critère d'arrêt. On obtient ainsi, en plus de l'image segmentée, un arbre dont les feuilles représentent des classes, et dont les noeuds représentent différents niveaux d'abstraction des classes obtenues. Ceci fait, chaque région se voit attribuer une signature spectrale correspondant à la moyenne de tous les pixels de l'image MS qui chevauchent cette région. Ceci est illustré dans la figure 1.5.

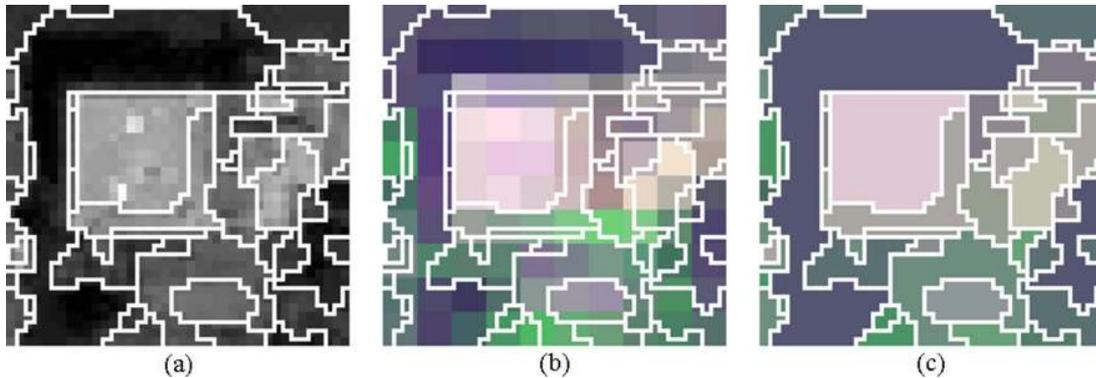


FIGURE 1.5 – Segmentation multi-résolution. a) Segmentation de l'image panchromatique, b) Résultat précédent superposé à l'image multi-spectrale, c) Chaque région se voit attribuer une signature spectrale. (Gaetano *et al.*, 2009)

L'étape suivante consiste à réaliser un clustering dans le domaine spectral, dans le but de raffiner la segmentation en séparant les différentes classes sémantiques initialement regroupées à cause de leurs niveaux de gris proche. Le clustering est réalisé sur chaque classe de niveau de gris. Les résultats obtenus à l'issue de cette phase sont très proches des classes en entrée de l'algorithme, mais cela permet de raffiner la segmentation tout en jouant sur la sur-segmentation qui sera de toute façon corrigée dans l'étape suivante.

La dernière étape consiste en un clustering spatial, basé sur des chaînes de Markov. À chaque étape du clustering, la région avec le plus faible score de texture est fusionnée avec son voisin dominant. On obtient ainsi à nouveau une structure en arbre binaire représentant la hiérarchie de classes.

La force de cet algorithme est de pouvoir différencier deux classes ayant la même signature spectrale en fonction de leur voisinage. La figure 1.6 illustre cet exemple.

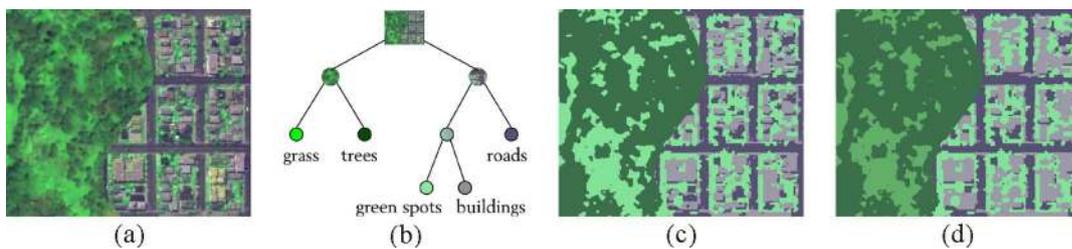


FIGURE 1.6 – Segmentation multi-résolution. a) Image originale, b) Hiérarchie de classes, c) Segmentation avant clustering spatial (*grass* confondu avec *green spots*), d) Segmentation après clustering spatial (*grass* différent de *green spots*). (Gaetano *et al.*, 2009)

1.2.3 Approches hiérarchiques

Une approche hiérarchique se base sur une segmentation multi-niveau dans laquelle un objet d'un niveau donné est directement mis en relation avec l'objet de niveau supérieur (dans lequel il est contenu), avec les objets de niveau inférieur (qu'il contient) et avec ses voisins dans le niveau courant. Dans un tel cas, il n'y a pas de chevauchement partiel, un objet est intégralement compris dans l'objet de niveau supérieur.

À partir de cette définition, il est possible de construire un arbre complet (dont toutes les feuilles sont à même distance de la racine), dans lequel chaque niveau représente une segmentation à une échelle donnée, et dont la hiérarchie représente l'appartenance d'un objet à des concepts de niveau supérieur. La figure 1.7 illustre cette idée.

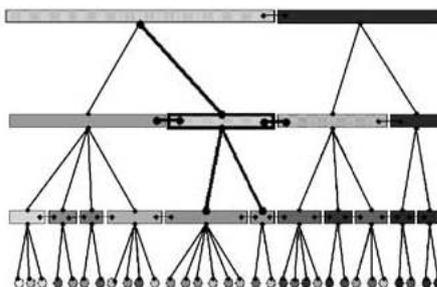


FIGURE 1.7 – Représentation d'une hiérarchie de segments. Chaque segment d'un niveau donné contient plusieurs segments du niveau inférieur. (Blaschke, 2003)

Un tel type d'approche est présenté par Burnett et Blaschke (2003) et utilisé par Blaschke (2003) pour analyser des zones agricoles et forestières. Les images sont segmentées à l'aide d'un processus multi-échelle dans lequel chaque niveau est à un niveau d'abstraction différent. Ce point là a été particulièrement soulevé par Burnett et Blaschke (2003). En effet, les auteurs mettent en avant le fait que bien qu'un objet soit composé de plusieurs autres dans les niveaux inférieurs, il représente tout de même une entité propre dans le niveau d'abstraction donné.

Pour illustrer ceci, nous pouvons prendre l'exemple d'une forêt. Une forêt est un objet d'intérêt en tant que tel, bien qu'elle soit composée d'arbres. Au niveau du paysage, l'ensemble des arbres composant la forêt n'ont aucun intérêt individuellement. Un avantage de cette méthode est sa complexité, puisque les informations relatives à chaque objet (signature spectrale, taille, ...) peuvent être calculées à partir des noeuds fils dans l'arbre.

L'intérêt d'une telle approche réside dans la structure hiérarchique des objets d'intérêt. Nous pouvons cependant noter un manque de souplesse dans la structure de données, une classe d'objets pouvant être détectée dans un niveau mais être plus pertinente dans un autre.

Une autre segmentation utilisant cette approche a été développée par Akçay et Aksoy (2008). Leur algorithme de segmentation s'applique individuellement sur chaque bande de l'image, à partir des informations spectrales, morphologiques et de voisinage. Premièrement, des opérations d'ouverture et de fermeture morphologiques sont appliquées sur les bandes spectrales en utilisant des éléments structurant de taille croissante pour générer des profils morphologiques. À partir de là sont produits autant de hiérarchies que de bandes, une hiérarchie représentant un ensemble de composantes connexes. L'étape suivante consiste à regrouper les éléments structurants les plus significatifs. La figure 1.8 illustre cette idée.

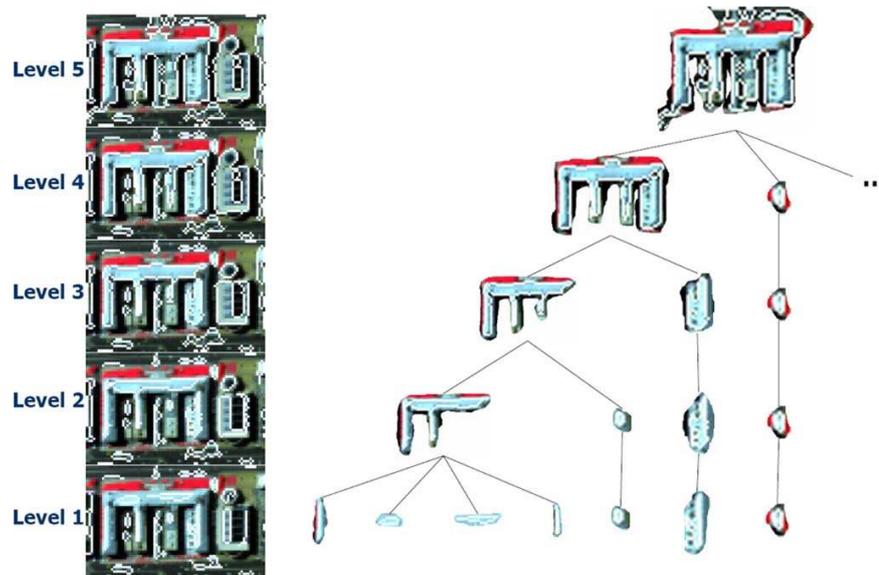


FIGURE 1.8 – Représentation d'une hiérarchie de composantes connexes. Les composantes connexes les plus significatives sont fusionnées. (Akçay et Aksoy, 2008)

Cela est fait avec l'algorithme *Bottom Up / Top Down*. Cet algorithme nécessite une mesure qui prend en compte la signature spectrale ainsi que la taille de l'objet à traiter. L'algorithme se déroule en deux passes. Il est décrit par l'algorithme 2. Dans cet algorithme, on considère que les feuilles représentent le niveau 1 de la hiérarchie et que la racine représente le niveau n (n étant la profondeur de la hiérarchie). À la fin de l'algorithme, tous les noeuds marqués lors de la passe *Top Down* représentent les composantes connexes les plus significatives.

L'étape suivante de la détection d'objet consiste à grouper les éléments structurants détectés précédemment pour trouver des groupes cohérents qui correspondent à des objets d'intérêt dans l'image initiale. Pour ce faire, ils utilisent l'analyse sémantique latente probabiliste (abrégé PLSA en anglais) décrit par Hofmann (2001). PLSA est une méthode de traitement automatique des langues inspirée de l'analyse sémantique latente, qui inclut un modèle statistique particulier. PLSA est appliqué ici aux éléments structurants extraits précédemment dans le but de détecter les objets d'intérêt.

Kurtz *et al.* (2011a,b) proposent une approche hiérarchique basée sur les arbres binaires de partitions. À partir d'une segmentation initiale, les objets adjacents sont fusionnés deux à deux selon un critère d'homogénéité radiométrique et structurel. L'utilisation d'information structurale permet de garder une organisation géométrique des segments correspondant aux objets d'intérêt. Pour définir la segmentation finale, l'arbre binaire de partition est coupé manuellement par un expert.

Les auteurs soulèvent le problème de l'échelle dans une segmentation d'image THRS. En effet, dans une telle image nous pouvons distinguer plusieurs types d'objets à plusieurs niveaux (par exemple pavillon et îlot résidentiel). Pour palier ce problème, ils proposent d'adapter la segmentation par arbres binaires de partition à des zones locales de l'image représentant un même type d'objets d'intérêt, homogène du point de vue radiométrique. L'expert peut ainsi

Algorithme 2: Bottom Up / Top Down (Akçay et Aksoy, 2008)

```

// (Passe Bottom Up)
pour tous les noeud  $N$  de niveau 1 faire
  └ Marquer  $N$ 
pour tous les niveau  $i = 2$  à  $n$  faire
  └ pour tous les noeud  $N$  du niveau  $i$  faire
    └ si  $N$  a une mesure plus grande que tous ses fils alors
      └ Marquer  $N$ 
// (Passe Top Down)
pour tous les noeud  $N$  de niveau  $n$  faire
  └ si  $N$  a été marqué lors de la passe précédente alors
    └ Marquer  $N$ 
pour tous les niveau  $i = n - 1$  à 1 faire
  └ pour tous les noeud  $N$  du niveau  $i$  faire
    └ si  $N$  a été marqué lors de la passe précédente et si aucun de ses ancêtres n'a été
      marqué lors de la passe courante alors
        └ Marquer  $N$ 

```

définir une coupe pour une imagerie donnée et cette coupe sera utilisée pour analyser les arbres binaires de partitions associés aux images similaires.

Pour permettre une segmentation plus poussée à différents niveaux sémantiques dans une même image, les auteurs proposent d'appliquer cette méthode à plusieurs échelles dans le but de pouvoir obtenir une bonne partition globale de l'image.

1.3 Approches guidées par la connaissance

Une approche guidée par la connaissance n'est pas une technique de segmentation en tant que telle, mais plutôt un outil pour raffiner ou améliorer une segmentation en fonction de connaissances données par un expert. Ces connaissances expertes peuvent se matérialiser sous différentes formes.

1.3.1 Vérité terrain

Une vérité terrain représente une information mesurée sur le terrain par un expert. Cette information peut être de toute sorte (nature du sol, forme d'un bâtiment, contour d'un quartier, ...). La figure 1.9 représente une vérité terrain ainsi que les classes associées.

L'utilisation d'une vérité terrain peut se faire dans différents cas. Akçay et Aksoy (2008), Gaetano *et al.* (2009), Derivaux *et al.* (2008) et Forestier *et al.* (2008) proposent des méthodes utilisant des vérités terrains pour vérifier que les résultats de leurs segmentations correspondent aux observations de l'expert.

Dans d'autres cas, une vérité terrain est utilisée pour entraîner l'algorithme de segmentation. Derivaux *et al.* (2010) proposent deux méthodes, appelées *probashed* et *optimized probashed* utilisant une vérité terrain à cette fin. Ces algorithmes sont basés sur l'algorithme de ligne de partage des eaux (voir section 1.1.1).

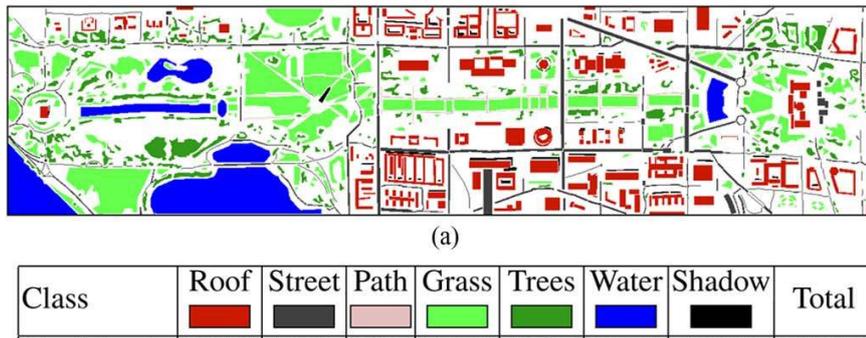


FIGURE 1.9 – Exemple de vérité terrain, avec les 7 classes associées. (Akçay et Aksoy, 2008)

La vérité terrain est formalisée par autant de masques binaires que de classes référencées par l'expert. Dans un premier temps (algorithme *probashed*), l'espace des données est modifié. Les pixels de l'image de référence sont projetés dans un nouvel espace de données dans lequel les vérités terrain sont composées de pixels homogènes. Une classification floue est ensuite effectuée, dont le résultat est une carte d'appartenance en niveau de gris. Une segmentation classique de ligne de partage des eaux est ensuite appliquée sur cette carte d'appartenance.

La difficulté avec un algorithme de segmentation tel que la ligne de partage des eaux est de lui fournir les bons paramètres pour obtenir des résultats correspondant à une réalité sémantique. Pour pallier ce problème, les auteurs proposent d'utiliser un algorithme génétique pour trouver ces *bons* paramètres.

La méthode précédente est ainsi optimisée à l'aide d'un algorithme génétique (algorithme *optimized probashed*). La phase de segmentation simple est remplacée par un algorithme génétique, dont la fonction d'évaluation cherche à maximiser les régions correspondant à des exemples issus de la vérité terrain et cherche à minimiser la sous-segmentation et la sur-segmentation. Le principe de cet algorithme est illustré dans la figure 1.10, et un exemple est donné dans la figure 1.11.

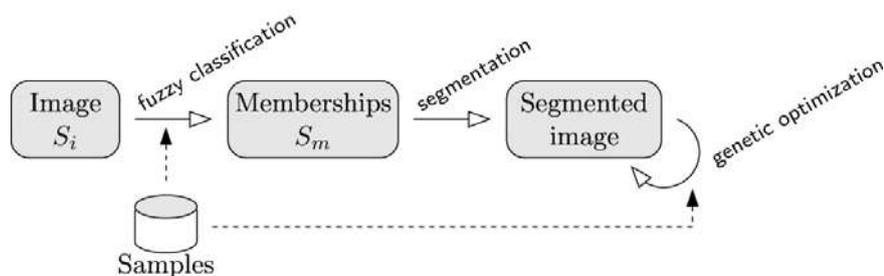


FIGURE 1.10 – Déroulement de l'algorithme *optimized probashed*. (Derivaux *et al.*, 2010)

1.3.2 Ontologie

Une ontologie est une spécification abstraite, une vue simplifiée du monde représentée dans un but précis. Une ontologie définit un ensemble de concepts organisés de manière hiérarchique.

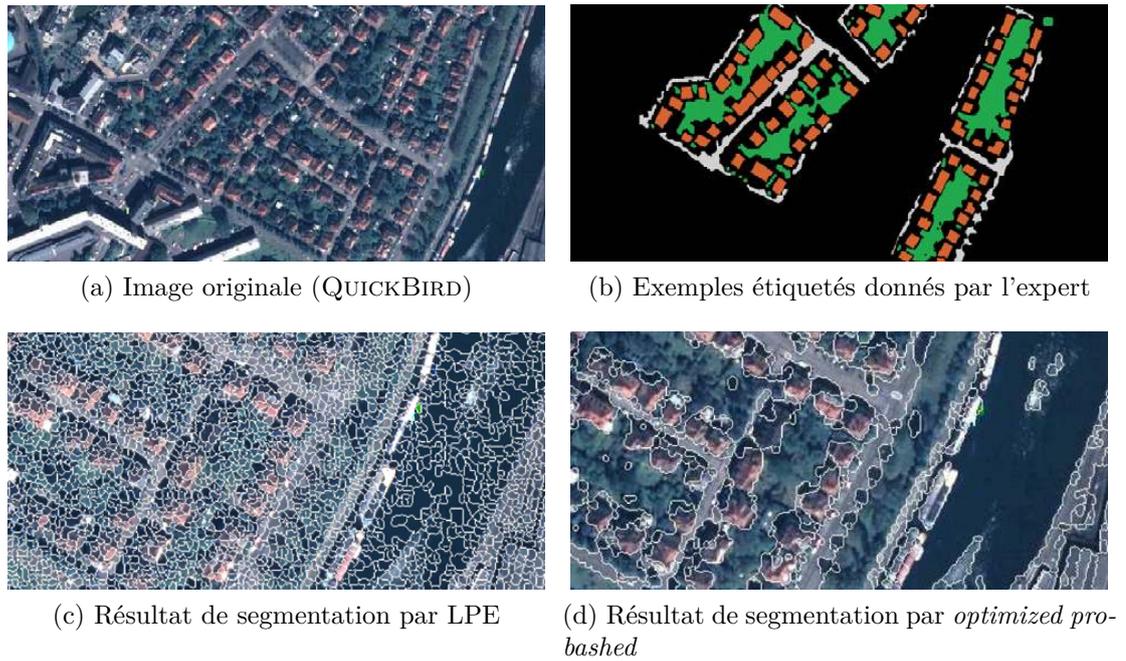


FIGURE 1.11 – Exemple d'une image QUICKBIRD avec le masque associé à plusieurs classes, et résultats obtenus par ligne des partage de eaux et l'algorithme *optimized probashed*. (Derivaux *et al.*, 2010)

Les concepts y sont décrits par leurs caractéristiques et leurs relations entre eux. La figure 1.12 donne un extrait simplifié d'une ontologie associée à un paysage urbain.

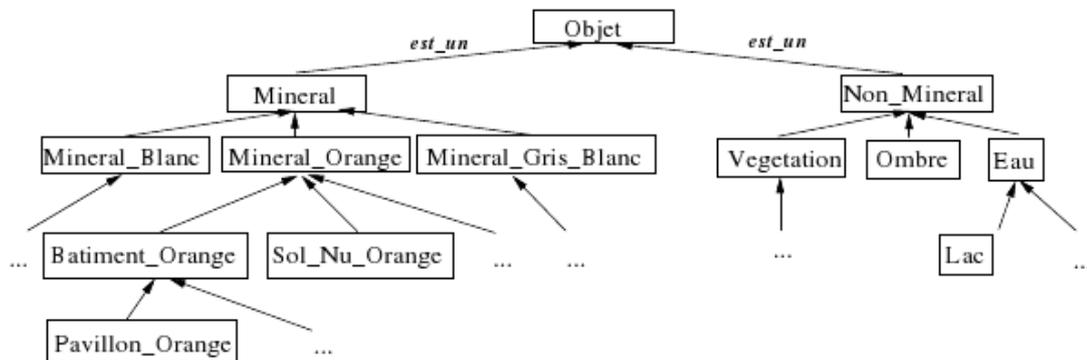


FIGURE 1.12 – Extrait d'une ontologie associée à un paysage urbain. (Forestier *et al.*, 2008)

Forestier *et al.* (2008) et Derivaux *et al.* (2008) utilisent une ontologie produite par l'expert pour guider la segmentation. À chaque concept de l'ontologie est associé un certain nombre de

contraintes, modélisées par des intervalles de valeurs admises, qui vont caractériser ce concept et les concepts héritant de celui-ci. À ces contraintes sont associés des exemples définis par l'expert. Ces exemples sont composés d'une image et d'un masque binaire définissant les parties représentatives du concept donné. Ces exemples servent à enrichir l'ontologie. En plus d'enrichir l'ontologie, ces exemples vont permettre de valider les intervalles de données fournis par l'expert, et enrichir les concepts dont les données n'ont pu être renseignées. L'utilisation d'exemples reste complémentaire au savoir de l'expert.

Cette ontologie enrichie va servir à construire des détecteurs. Un détecteur est un algorithme capable, à partir d'une image, d'associer un ensemble d'instances de la classe qu'il recherche. Ces détecteurs ont pour but d'améliorer la détection d'objets associés aux concepts de l'ontologie, améliorant ainsi la qualité de la segmentation. Pour plus de précision sur l'utilisation d'exemples dans le processus de segmentation et l'extraction de détecteurs, le lecteur intéressé pourra consulter la thèse de Sébastien Derivaux (Derivaux, 2009).

Une telle technique fait grandement appel à l'expert, car seul lui peut remplir efficacement l'ontologie de manière à ce qu'elle soit exploitable. L'avantage est de pouvoir restreindre les concepts avec des contraintes définies par l'expert (limite de taille pour une maison, ...).

2 Utilisation de connaissances expertes pour la détection d'objets d'intérêt

La connaissance de l'expert peut intervenir à différents niveaux. Elle peut être utilisée pour créer des détecteurs capable d'identifier une classe particulière d'objets d'intérêt. Elle peut aussi s'exprimer sous la forme d'un système d'information géographique.

Dans le cadre de nos travaux, nous cherchons à étudier les paysages urbains. Les détecteurs qui nous intéressent sont les détecteurs de routes et les détecteurs de bâtiments.

2.1 Détection de routes

Il existe dans la littérature un grand nombre de méthodes visant à extraire les routes à partir d'une image. Les algorithmes les plus efficaces sont ceux reposant sur une détection de lignes dans l'image. Nous nous concentrons ici sur ce type d'approches. Parmi les travaux portant sur l'extraction de lignes, nous pouvons notamment citer les travaux de Steger (1998), dans lesquels l'auteur propose un algorithme de détection de lignes qui utilise une approche géométrique différentielle et un modèle explicite pour les lignes.

Dans leurs travaux, Bacher et Mayer (2005) proposent une méthode d'extraction de routes non supervisée utilisant la méthode de Steger (1998) pour extraire des lignes à partir de chacune des bandes d'une image multi-spectrale. Ces segments sont considérés comme des hypothèses de route. Un algorithme de classification permet de sélectionner les hypothèses les plus fiables d'un point de vue local (élongation, radiométrie, ...). Les hypothèses provenant de différentes bandes sont fusionnées pour éliminer la redondance. Les hypothèses restantes sont utilisées pour construire un réseau prenant en compte les propriétés globales d'un réseau routier (transport optimal d'un point à un autre). Ceci est fait en deux étapes. Premièrement, les segments proches sont liés à l'aide d'une méthode basée sur un graphe pondéré. Les connexions manquantes sont détectées en cherchant le plus court chemin entre deux paires de points à l'aide de l'algorithme de Dijkstra. Deuxièmement, les connexions potentielles restantes sont vérifiées à l'aide d'un modèle de contour actif (Kass *et al.*, 1988).

Les auteurs appliquent leur méthode sur des images multi-spectrales (QUICKBIRD, IKONOS). Les résultats obtenus, illustrés dans la figure 1.13, montrent que la méthode proposée est efficace

dans des zones ouvertes (campagne, ...) mais peu efficace dans des environnement urbains.

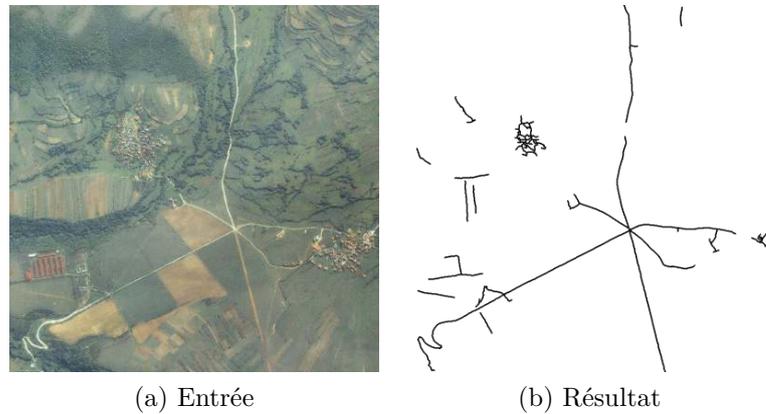


FIGURE 1.13 – Résultats obtenus par la méthode de [Bacher et Mayer \(2005\)](#).

Les travaux de [Jeon *et al.* \(2002\)](#) présentent une méthode similaire, reposant sur l'algorithme de détection de lignes de [Steger \(1998\)](#). Les auteurs définissent les facteurs de proximité et de co-curvilinearité comme mesure pour fusionner les segments proches. Dans un deuxième temps un algorithme génétique basé sur un processus de croissance de région est utilisé pour grouper de plus grands ensembles de segments. Les graines du processus de croissance de régions sont les segments ou groupes de segments suffisamment grands pour être considérés comme appartenant à un réseau routier. Pour combiner les nouveaux ensembles de segments obtenus par l'algorithme génétique, les auteurs utilisent un modèle de contour actif ([Lai et Chin, 1995](#)). Ils obtiennent ainsi, de manière non supervisée, un réseau routier correspondant.

Les auteurs procèdent à une série d'expériences sur des images SAR d'une résolution de 30m représentant des paysages montagneux et des zones à proximité de villes. La méthode proposée permet d'extraire les réseaux routiers ainsi que les routes individuelles.

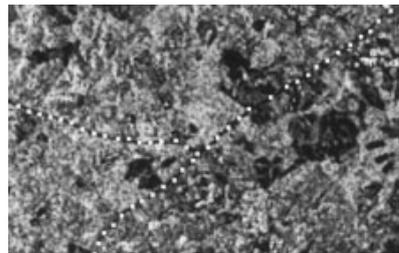


FIGURE 1.14 – Résultats obtenus avec la méthode de [Jeon *et al.* \(2002\)](#). En pointillé, le réseau détecté.

Les travaux de [Tupin *et al.* \(2002\)](#) s'intéressent à l'extraction de réseaux routiers dans des paysages urbains denses. La méthode proposée est composée de deux étapes. Premièrement des segments sont détectés par l'algorithme proposé par [Tupin *et al.* \(1998\)](#). Cet algorithme repose sur les tavelures (*i.e. speckles* en anglais) appliqués aux images SAR. Dans une deuxième étape, un graphe est créé avec ces segments. Ce graphe modélise un champ aléatoire de Markov. La

2. UTILISATION DE CONNAISSANCES EXPERTES POUR LA DÉTECTION D'OBJETS D'INTÉRÊT

recherche de cliques dans ce graphe va produire un réseau routier. Cette approche est appliquée à plusieurs échelles et les résultats sont combinés pour plus de pertinence.

La méthode proposée est appliquée sur des couples d'images SAR prises à différentes orientations. Un exemple est donné dans la figure 1.15. Les expériences montrent que selon l'orientation des capteurs, les résultats sont plus ou moins pertinents. Si pour un couple d'images les capteurs ont des orientations différentes, un réseau complet peut être extrait. Si les capteurs sont parallèles, les résultats sont améliorés par rapport à une simple image mais ne sont pas aussi complets que dans le cas précédent.



FIGURE 1.15 – Illustration des travaux de [Tupin et al. \(2002\)](#) en prenant en compte deux images orthogonales.

Dans sa thèse, [Poulain \(2010\)](#) propose de combiner plusieurs détecteurs de routes. Il utilise la méthode proposée par [Christophe et Inglada \(2007\)](#) sur une image multi-spectrale et la méthode proposée par [Amberg et al. \(2005\)](#) sur une image radar. Ces deux méthodes génèrent des hypothèses de routes, contenant beaucoup de faux positifs. Des détecteurs (absence de végétation, d'ombre projetée, parallélisme avec un groupe de bâtiments, ...) sont utilisés pour attribuer des scores à chaque hypothèse. Les scores produits par ces détecteurs sont fusionnés avec la théorie de Dempster-Shafer ([Shafer, 1976](#)) pour ne garder que les hypothèses de routes les plus probables. Un réseau routier est extrait à partir des segments restant à l'aide d'un modèle de contour actif ([Williams et Shah, 1992](#)).

Une telle méthode permet une plus grande exhaustivité des résultats car elle prend en compte des sources de données multiples. De plus, elle tient compte de la connaissance experte via l'utilisation des détecteurs et permet une modélisation pertinente des phénomènes d'ignorance et d'imprécision à travers l'utilisation de la théorie des croyances de Dempster-Shafer. Un exemple de résultat est donné dans la figure 1.16.

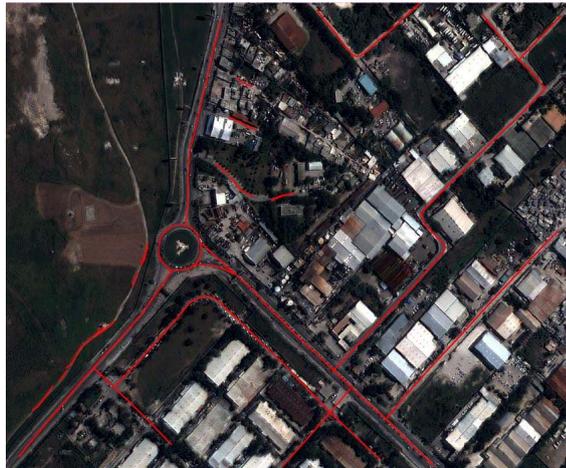


FIGURE 1.16 – Application de la méthode de Poulain (2010) après fusion des détecteurs.

2.2 Détection de bâtiments

La détection de bâtiment est un sujet de recherche actif. De très nombreuses méthodes sont proposées dans la littérature. Nous regroupons les méthodes en fonction des primitives extraites (contours, lignes, région, ...).

2.2.1 Approches par détection de contours

La méthode proposée par Lhomme dans sa thèse (Lhomme, 2005) se base sur la variance pour définir les contours d'un bâtiment. La variance est calculée à partir de l'image panchromatique. Pour des valeurs de taille et une orientation de bâtiment recherchées, les valeurs moyennes de la variance pour les zones de corps et des zones de contours de bâtiments sont extraites en faisant varier la fenêtre de convolution en fonction de la taille et de l'orientation. Une nouvelle image est créée dans laquelle la valeur d'un pixel est donné par le rapport entre la valeur du corps et la valeur du contour calculé précédemment. Les minima locaux, correspondant aux centres des hypothèses de bâtiment, sont extraits puis seuillés. Pour réduire les erreurs, des masques d'ombre et végétation sont calculés puis combinés au processus à l'aide d'un raisonnement basé sur la logique floue. Les bâtiments sont alors reconstruits en fonction des paramètres de taille et d'orientation utilisés.

Cette méthode est appliquée sur une image IKONOS d'une résolution de un mètre. L'avantage de cette méthode est d'être totalement indépendante de la signature spectrale des bâtiments et est applicable sur différents types d'images. Cependant, nous pouvons nous poser la question de son application en milieu urbain dense. En effet, la détection des contours repose sur la variance, or deux bâtiments adjacents vont avoir une variance faible et donc pouvoir être considérés comme formant un seul bâtiment de plus grande taille. Un exemple est donné dans la figure 1.17.

La méthode proposée par Peng *et al.* (2005) est une extension du modèle de contours actifs (Kass *et al.*, 1988) qui prend en compte les propriétés radiométriques et géométriques des bâtiments. L'algorithme original est modifié en deux points. Premièrement, le placement des points initiaux est modifié pour être proche des contours du bâtiment à identifier et prendre en compte l'illumination et l'ombre, dans le but de minimiser les étapes de calcul. Deuxièmement, la fonction d'énergie extérieure est modifiée en considérant que la forme d'un bâtiment est caractérisée

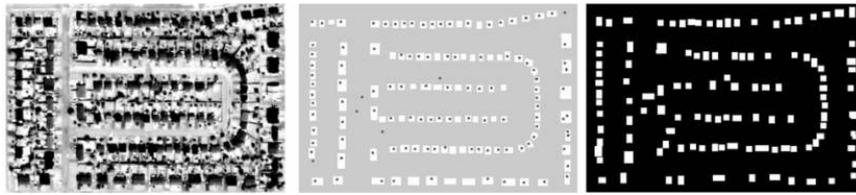


FIGURE 1.17 – Illustration de la méthode de [Lhomme \(2005\)](#). À gauche, l'image en entrée ; au centre, l'identification des centres ; à droite, la reconstruction des bâtiments.

par des segments de lignes droites et par des coins. Un post-traitement est effectué pour éliminer les erreurs dues à l'illumination de certains bâtiments.

La méthodologie est testée sur des images aériennes. Les résultats obtenus montrent que les bâtiments aux formes complexes ainsi que les bâtiments adjacents sont correctement identifiés, ce qui met en avant les améliorations apportées à l'algorithme original. Cependant, tous les bâtiments ne sont pas extraits. Un exemple est donné dans la figure 1.18.



FIGURE 1.18 – Exemple de résultats pour la méthode de [Peng *et al.* \(2005\)](#).

2.2.2 Approches par détection de lignes

[Kim et Muller \(1999\)](#) proposent une méthode d'extraction de bâtiments basée sur l'utilisation de lignes. Premièrement les contours sont détectés par les méthodes de [Canny \(1986\)](#); [Petrou et Kittler \(1991\)](#) puis transformés en lignes ([Kim et Muller, 1996](#)). Ces lignes sont seuillées (taille et angle) pour ne garder que les plus significatives. Deuxièmement un graphe est créé avec les lignes comme sommets et leurs liens (relations géométriques entre les lignes) comme arêtes. Les hypothèses de routes sont construites en cherchant des cycles dans le graphe. Dans une dernière étape, les hypothèses sont validées par fusion en prenant en compte l'ombre et les lignes verticales.

Cette méthode a été appliquée sur des images aériennes. Les résultats montrent l'efficacité de la méthode au niveau de la précision des bâtiments détectés, mais aussi le manque d'exhaustivité des résultats positifs. Un exemple est donné dans la figure 1.19.

2.2.3 Combinaison segmentation / classification

Dans le but d'extraire les bâtiments à partir d'une image multi-spectrale, [Poulain \(2010\)](#) propose d'appliquer l'algorithme MeanShift ([Comaniciu et Meer, 2002b](#)) avec plusieurs jeux



FIGURE 1.19 – Illustration de la méthode de [Kim et Muller \(1999\)](#)

de paramètres distincts (en faisant varier le rayon spectral) pour obtenir une segmentation à plusieurs niveaux. L'ensemble des régions produites est considéré comme hypothèse de bâtiments. Des détecteurs (absence de végétation, présence d'ombre projetée, de contours linéaire, ...) sont utilisés pour attribuer des scores à chaque hypothèse. Les scores produits par ces détecteurs sont fusionnés avec la théorie de Dempster-Shafer ([Shafer, 1976](#)) pour ne garder que les hypothèses de routes les plus probables.

Cette méthode propose les mêmes avantages que la méthode proposée par le même auteur pour l'extraction de routes (source de données multiples, utilisation de connaissance experte, modélisation de l'imprécision, voir section 2.1). Un exemple de résultat est donné dans la figure 1.20.



FIGURE 1.20 – Application de la méthode de [Poulain \(2010\)](#) après fusion des détecteurs.

[Zhang \(1999\)](#) propose une méthode de classification basée pixel pour identifier le bâti. À partir d'une image panchromatique, quatre matrices de co-occurrences sont calculées dans quatre

directions. Des mesures de contraste, d'énergie, d'homogénéité et d'entropie sont calculées dans le but de produire des images en nuances de gris. Ces images sont divisées en cinquante clusters avec l'algorithme ISODATA (voir chapitre 1 section 1.2.1). Ces clusters sont interprétés manuellement et six d'entre eux sont identifiés comme une partie du bâti et fusionnés.

L'utilisation d'une matrice de co-occurrence permet d'obtenir de meilleurs résultats qu'une classification basée uniquement sur l'image panchromatique car elle permet de prendre en compte une information spatiale, la matrice de co-occurrence étant dépendante de la direction.

2.2.4 Images laser

La télédétection par laser (LIDAR) est une technologie de mesure optique basée sur les propriétés d'une lumière laser. Cette technologie permet notamment de générer des modèles numériques de terrain (MNT, ou DTM en anglais). Une MNT est une représentation de la topographie (altimétrie et/ou bathymétrie) d'une zone terrestre. Les images LIDAR sont beaucoup étudiées pour l'extraction de bâti car elles permettent d'analyser la différence d'élévation entre deux points et donc la hauteur des bâtiments.

Dans leurs travaux, [Awrangjeb *et al.* \(2010\)](#) construisent deux masques à partir d'une image LIDAR, le premier plus restrictif que le second, représentant les objets dont la hauteur dépasse un seuil donné. À partir du premier masque, les auteurs construisent les contours des objets extraits avec le filtre proposé par [Canny \(1986\)](#) et transforment ces contours en segments après avoir détecté les angles des contours. Pour éliminer un maximum de faux positifs, les segments les plus petits et ceux avec un indice de végétation (calculé à partir d'une image multi-spectrale) trop élevé sont supprimés. Les lignes restantes sont alors étendues pour former des rectangles représentant les bâtiments. Dans une dernière étape, les rectangles sont encore étendus en prenant cette fois en compte le second masque comme limite, ainsi que les valeurs radiométriques du bâtiment dans l'image multi-spectrale.

Cette méthode offre de bons résultats en milieu urbain, mais impose de disposer à la fois d'une image LIDAR et d'une image multi-spectrale. De plus, les résultats sont sensibles aux variations d'élévation (terrain en pente). Un exemple est donné dans la figure 1.21.

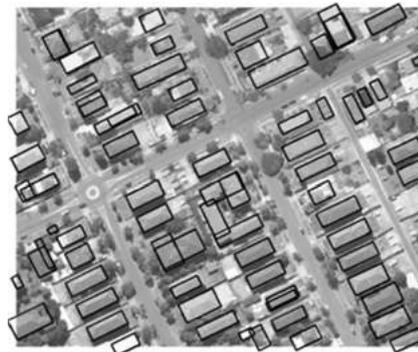


FIGURE 1.21 – Exemple d'utilisation de la méthode de [Awrangjeb *et al.* \(2010\)](#) pour la détection de bâtiments.

[Rottensteiner *et al.* \(2005, 2007\)](#) proposent une méthode de classification pixels et régions basée sur la théorie de Dempster-Shafer ([Shafer, 1976](#)). Des mesures de texture et de hauteurs sont calculées à partir des données LIDAR ainsi que la valeur de l'indice NDVI pour chaque pixel. Ces mesures sont évaluées et fusionnées à l'aide de la méthode de Dempster-Shafer, chaque pixel

est ainsi classifié. Une ouverture morphologique est appliquée pour éliminer les zones les plus petites. Les composantes connexes sont calculées pour créer des régions étiquetées. Une nouvelle classification sur les régions est menée à l'aide de la méthode de Dempster-Shafer. Un masque de bâtiments est obtenu.

Cette méthode donne de bons résultats mais reste sujette aux mêmes limites que la méthode de Awrangjeb *et al.* (2010) (utilisation d'images LIDAR et multi-spectrale, variation de pente). Cependant, l'utilisation de la méthode de Dempster-Shafer permet une grande modularité au niveau des paramètres. Un exemple est donné dans la figure 1.22.



FIGURE 1.22 – Illustration de la méthode de Rottensteiner *et al.* (2005). En gris, les bâtiments correctement détectés ; en noir, les faux positifs.

2.3 Utilisation de système d'information géographique

Qu'est ce qu'un SIG ? Un Système d'Information Géographique (abrégié SIG) est un système d'information capable d'organiser et de présenter des données alphanumériques géoréférencées, ainsi que de produire des plans ou des cartes. Un SIG contient généralement des données 2D et permet le traitement et la diffusion d'informations géographiques. Les données peuvent être géométriques, descriptives, graphiques, *et caetera*.

L'usage courant d'un SIG est la représentation plus ou moins réaliste de l'environnement spatial en se basant sur des primitives géométriques (points, vecteurs, polygones, ...) auxquelles sont associées des informations telles que la nature (forêt, voie ferrée, ...) ou toute autre information contextuelle (nombre d'habitants, superficie, ...). On peut notamment y trouver d'autres informations telles que des cartes topographiques, des images de télédétection ou encore des informations temporelles.

L'intérêt d'un SIG est qu'il donne la possibilité de centraliser toute sorte de données aussi diverses que riches. Ces données peuvent alors être utilisées par un processus de segmentation, voire mises à jour par un tel processus. Une fonctionnalité importante des SIG est de permettre le calcul de relations spatiales entre les objets géoréférencés.

Parmi les SIG existant nous pouvons notamment citer la BD Topo de l'IGN¹ et OpenStreet-

1. <http://professionnels.ign.fr/bdtopo>, données disponibles gratuitement pour les missions de services publiques, pour la recherche et l'enseignement

Map².

Un SIG pour guider la détection d'objets d'intérêt. L'utilisation d'un Système d'Information Géographique peut se faire à différents niveaux. *Ziems et al. (2007)* utilisent la position et l'épaisseur des segments de routes présents dans un SIG pour contraindre un algorithme de segmentation. Pour chaque segment de route présent dans le SIG, une région correspondant à cette route est produite, ainsi que deux régions correspondant aux régions à gauche et à droite. Les propriétés radiométriques (homogénéité, contraste, ...) de ces régions sont apprises et vont servir à définir les paramètres d'un algorithme de détection de routes (*Wiedmann, 2001, 2003*) basé sur l'algorithme de détection de lignes de *Steger (1998)*. Un exemple de résultat est donné dans la figure 1.23.

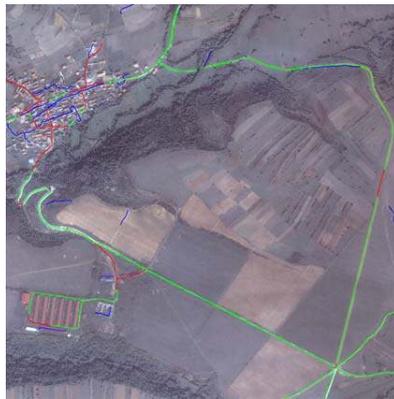


FIGURE 1.23 – Illustration de la méthode de *Ziems et al. (2007)*. En vert, les routes correctement extraites ; en bleu, les faux positifs ; en rouge, les routes manquantes.

Les données d'un SIG peuvent être utilisées conjointement à l'image. *Tönjes et Growe (1998)* construisent un réseau sémantique pour représenter la connaissance experte du domaine à travers l'utilisation de différents capteurs et la fusion des résultats. Le parcours de ce réseau permet de retrouver les hypothèses de routes les plus pertinentes. *Burnett et Blaschke (2003)*; *Blaschke (2003)* se basent sur un SIG pour obtenir des images et les traiter à l'aide de toute l'information disponible dans la base de données de manière à l'enrichir.

Mise à jour et validation de SIG à l'aide d'images. Les travaux de *Gerke et Heipke (2008)* visent à valider les routes présentes dans un SIG. Cette évaluation se base sur les routes extraites à partir d'une image aérienne ou satellite. L'algorithme de détection de routes utilisé est celui de *Wiedmann (2001)*. Considérant qu'une route peut être partiellement ou totalement recouverte par un alignement d'arbres, ceux-ci sont aussi extraits par la méthode proposée par *Gimel'farb (1996)*. Les routes présentes dans le SIG sont évaluées individuellement à partir de critères géométriques et topologiques, combinés avec une mesure probabiliste. Une étape d'analyse globale cherche le plus court chemin (algorithme A*) dans un graphe dont les arrêtes représentent les hypothèses de routes. Dans une dernière étape les hypothèses de routes sont à nouveau évaluées individuellement avec des paramètres plus souples.

2. <http://www.openstreetmap.org/>, base de données collaborative en accès libre

Dans cette méthode, la qualité de l'algorithme de détection de routes influence largement le processus d'évaluation du SIG. La prise en compte d'informations contextuelles ainsi que l'analyse de graphe permet d'obtenir un faible taux d'erreurs. Un exemple est donné dans la figure 1.24.

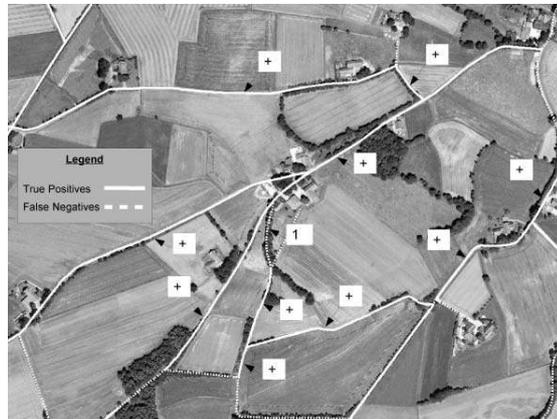


FIGURE 1.24 – Exemple d'application de la méthode de Gerke et Heipke (2008).

Qiaoping et Couloigner (2004) proposent une méthode pour la mise à jour de SIG. Dans un premier temps, les routes sont extraites d'une image panchromatique. Les intersections entre routes sont extraites avec des ondelettes Chui (1992). Ces valeurs sont seuillées pour ne garder que les hypothèses les plus probables. Ces intersections définissent des segments de routes, les centres de ces segments sont extraits et un réseau est construit. Les lignes sont ensuite simplifiées pour diminuer leur nombre. Ces données sont stockées dans un SIG.

Dans un second temps, le SIG construit à partir de l'image est comparé au SIG de référence. Pour cela, les points du réseau sont appairés avec les points correspondant dans le SIG. Les points de chaque paire sont combinés pour obtenir les nouvelles coordonnées de chaque point. Le même processus est répété pour les polygones. Le SIG de référence est mise à jour avec ces nouvelles primitives. Les anciennes valeurs sont gardées dans un historique spatio-temporel. Un exemple est donné dans la figure 1.25.

D'autres méthodes visent à mettre à jour les bâtiments (Phalke et Couloigner, 2005; Straub *et al.*, 2000) ou les routes (Baumgartner *et al.*, 1996; Mena et Malpica, 2005) présents dans des SIG. Les différentes méthodes reposent sur un modèle commun et permettent à la fois de vérifier l'algorithme de détection en le comparant au SIG mais aussi de vérifier le SIG en prenant en compte une image plus récente que le SIG.

2.4 Synthèse

L'état de l'art dans les sections précédentes ne vise pas à être exhaustif mais à présenter différents types d'approches. Nous allons maintenant classer ces approches en fonction des images en entrée ainsi que de l'utilisation ou non d'un SIG dans le processus d'analyse.

Extraction de routes. Nous avons vu dans les sections 2.1 et 2.3 plusieurs approches visant à extraire les réseaux routiers à partir d'une ou plusieurs images. Nous nous sommes concentrés sur les approches basées sur la détection de segments de droites, car ce sont ces méthodes qui offrent



FIGURE 1.25 – Exemple de mise à jour de SIG avec la méthode de Qiaoping et Couloigner (2004). En rouge, l'ancien SIG ; en vert, le SIG généré.

les meilleurs résultats. Le tableau 1.1 présente les différentes approches classées en fonction du type d'image sur lesquelles elles sont appliquées ainsi que de l'utilisation ou non d'un SIG.

	Type			SIG	Urbain
	Aérienne	Pan-MS	SAR		
Tönjes et Growe (1998)		x	x	x	
Jeon <i>et al.</i> (2002)			x		
Tupin <i>et al.</i> (2002)			x		x
Qiaoping et Couloigner (2004)		x		x	
Bacher et Mayer (2005)		x			
Ziems <i>et al.</i> (2007)		x		x	
Gerke et Heipke (2008)	x			x	
Poulain (2010)		x	x		x

TABLE 1.1 – Méthodes d'extraction de routes classées en fonction des images sur lesquelles elles sont appliquées.

Nous pouvons noter ici que les approches basées sur la détection de lignes s'appliquent sur tous types d'images (multi-spectrale, SAR, ...). Cependant, pour être portable une méthode donnée va devoir être finement réglée en fonction des caractéristiques de l'image (nombre de bandes, signature spectrale de l'objet recherché, ...) ce qui est fastidieux pour l'expert qui doit superviser l'algorithme. De plus, la détection de lignes est extrêmement sensible à la résolution de l'image : une image à une résolution de 10 mètres permettra d'extraire uniquement le réseau routier principal (autoroutes, grandes nationales) alors qu'une image d'une résolution inférieure au mètre permettra d'extraire beaucoup plus exhaustivement l'ensemble des routes, mais les résultats seront en contrepartie fortement bruités. Une même approche n'est donc pas exploitable en l'état sur différents jeux de données sans une interaction forte avec l'expert ou sans l'utilisation d'un SIG pour contraindre la méthode.

Extraction de bâtiments. Nous avons vu dans les sections 2.2 et 2.3 plusieurs approches visant à extraire le bâti à partir d'une ou plusieurs images. Le tableau 1.2 présente les différentes

approches classées en fonction du type d'images sur lesquelles elles sont appliquées ainsi que de l'utilisation ou non d'un SIG. Ces méthodes sont appliquées sur des images THRS.

	Aérienne	Type Pan-MS	SAR	SIG	Urbain
Kim et Muller (1999)	x				x
Zhang (1999)		x			x
Straub <i>et al.</i> (2000)		x		x	x
Lhomme (2005)		x			x
Peng <i>et al.</i> (2005)	x				x
Rottensteiner <i>et al.</i> (2005)		x	x		x
Phalke et Couloigner (2005)		x		x	x
Awrangjeb <i>et al.</i> (2010)		x	x		x
Poulain (2010)		x			x

TABLE 1.2 – Méthodes d'extraction de bâtiments classées en fonctions des images sur lesquelles elles sont appliquées.

Les bâtiments, plus particulièrement les habitats individuels, ne sont correctement discernables que sur des images à très haute résolution spatiale. De par la complexité d'un paysage urbain, une image à une telle résolution va être difficile à analyser (détail, bruit, ...) que ce soit par une approche segmentation / classification ou par l'extraction de primitives géométriques. L'utilisation de données laser est efficace pour l'extraction de bâti. En effet, la possibilité d'analyser la différence de hauteur entre deux points permet de différencier facilement les bâtiments du terrain. Cependant, de telles méthodes sont sensibles aux variations de l'inclinaison du terrain ainsi qu'à la présence d'objets (arbres, ...).

Pour lever cette limite, certaines approches combinent plusieurs images ou plusieurs détecteurs au sein d'une même image à l'aide de méthodes de fusion telles que Dempster-Shafer ou la logique floue. Ceci donne de bons résultats mais implique de disposer de plusieurs images ou détecteurs ainsi que de la connaissance d'un expert pour fusionner correctement ces données.

3 Le CNES, Pléiades et le programme d'accompagnement ORFEO

Le Centre National d'Études Spatiales (CNES) est un établissement public à caractère industriel et commercial dont le but est de coordonner et d'animer les activités spatiales françaises. Le CNES a cinq grands thèmes d'application, qui sont : accès à l'espace ; terre, environnement et climat ; applications grand public ; sécurité et défense ; sciences spatiales et préparation de l'avenir. La mission du CNES passe par la conception, le développement et l'exploitation de capteurs satellitaires, notamment des capteurs optiques à haute résolution (SPOT) et très haute résolution (PLEIADES).

3.1 Les images Pléiades

Le projet PLEIADES a commencé en 2000 en partenariat avec l'agence spatiale italienne (ASI). Il a pour objectif de combiner des outils optiques et radars. La partie optique (PLEIADES) est gérée par le CNES, la partie radar (COSMO-SKYMED) est gérée par l'agence spatiale italienne. La réalisation des capteurs PLEIADES a commencée en 2003. Le premier capteur PLEIADES-1A

a été lancé le 17 décembre 2011 et a été suivi par le capteur PLEIADES-1B le 2 décembre 2012. Les deux capteurs sont actuellement en phase opérationnelle.

Les capteurs PLEIADES forment une constellation de deux satellites fonctionnant en orbite héliosynchrone³, phasée et quasi circulaire à une altitude de 694km. Les deux capteurs sont placés à 180° l'un de l'autre et permettent l'acquisition de plus de 450 images par jour. De plus, PLEIADES offre une capacité d'acquisition stéréoscopique. Leurs petites tailles et leurs mobilités permettent un accès journalier en tout point du globe et une forte capacité de réaction en situation de crise.

Chaque capteur a une fauchée de 20km au nadir⁴. La résolution panchromatique est de 70cm, pour une résolution multispectrale quatre fois supérieure. Les images multispectrales sont composées de quatre bandes : bleu, vert, rouge et proche infrarouge. Les valeurs radiométriques des bandes spectrales sont présentées dans la figure 1.26. Les images brutes peuvent être traitées par fusion dans le but de produire une image multispectrale à la résolution de l'image panchromatique. L'image peut ensuite être orthorectifiée pour obtenir une image géoréférencée au système de coordonnées géographiques ou cartographiques. Ces images peuvent ensuite être ré-échantillonnées dans des produits de 70cm à 50cm. Ce type d'images est le produit final destiné à une utilisation dans un SIG. Des exemples d'images orthorectifiées sont visibles dans l'annexe A.

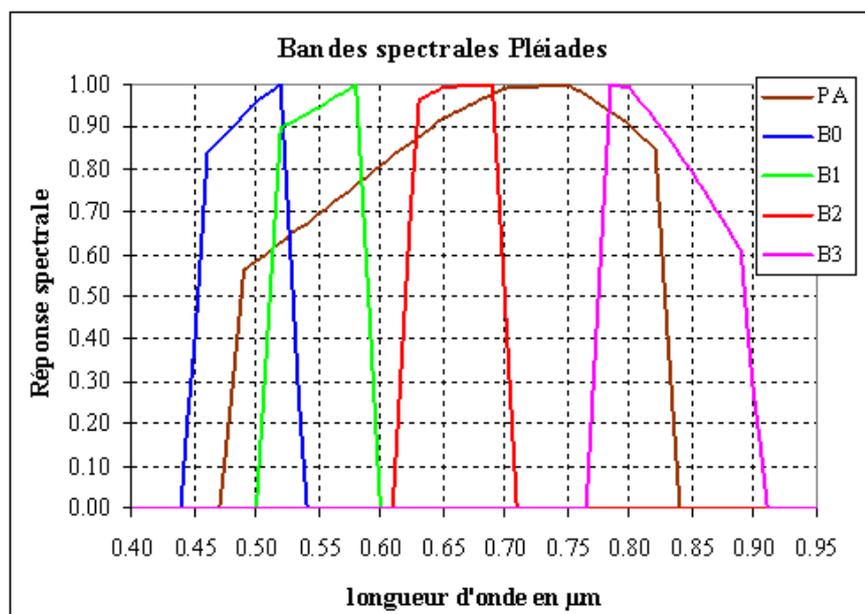


FIGURE 1.26 – Bandes spectrales du capteur PLEIADES (source http://smc.cnes.fr/PLEIADES/Fr/GP_mission.htm).

Les principales caractéristiques de la composante optique PLEIADES sont présentées dans le tableau 1.3.

3. Orbite géocentrique dont l'altitude et l'inclinaison sont choisies de façon à ce que l'angle entre le plan d'orbite et la direction du soleil soit constant. Un satellite placé sur une telle orbite repasse au-dessus d'un point donné à la même heure solaire locale (source : http://fr.wikipedia.org/wiki/Orbite_héliosynchrone).

4. Le point de la sphère céleste représentatif de la direction verticale descendante (à l'opposé du zénith).

Spécification image	résolution panchromatique	0.7m
	résolution multispectrale	2.8m
	champ de vue	20km
	couleurs	bleu, vert, rouge, proche infrarouge
Localisation des images	avec point d'appui	1m / 1000km
	sans point d'appui	<10m
Capacité d'acquisition	km ² par orbite (maxi)	30 000km ²
	mosaïque	120km × 120km
	stéréoscopie	20km × 300km
	cibles max sur 100km × 200km	30 (mode crise)

TABLE 1.3 – Les principales caractéristiques de la composante optique PLEIADES (source http://smc.cnes.fr/PLEIADES/Fr/GP_mission.htm).

3.2 Le programme d'accompagnement ORFEO

Avec l'arrivée de ces images optiques à très haute résolution spatiale (THRS), il est nécessaire de mettre en place un programme de suivi afin de préparer, promouvoir et accompagner l'utilisation et l'exploitation des capteurs PLEIADES.

L'accent est mis sur l'adaptation des méthodes existantes aux données PLEIADES et le développement de nouvelles méthodes adaptées. Ceci pour accompagner les nouveautés apportées par le programme ORFEO : nouveaux capteurs à très haute résolution spatiale, acquisitions journalières, stéréoscopiques, qualité des données, masse importante des données. Et ce, conjointement avec l'utilisateur final, dans le but de prendre en compte les besoins et spécifications de ces utilisateurs.

Le programme d'accompagnement ORFEO (de l'anglais *Optical and Radar Federated Earth Observation*) est composé d'un volet thématique et d'un volet méthodologie.

Le volet thématique a pour but de spécifier et de valider les produits à valeur ajoutée et services qui pourront être réalisés avec les données ORFEO afin de mieux répondre aux besoins des utilisateurs. Le volet thématique comprend huit groupes thématiques composés d'experts extérieurs au CNES :

GT1 : Mer et littoral

GT2 : Risques et aide humanitaire

GT3 : Cartographie et aménagement du territoire

GT4 : Géologie et géophysique

GT5 : Hydrologie

GT6 : Forêt

GT7 : Agriculture

GT8 : Défense

Le volet méthodologique a pour but de spécifier et développer les outils nécessaires à l'exploitation des images THRS optiques et radars. Les objectifs suivent l'évolution des besoins des utilisateurs. Le volet méthodologique est composé de deux groupes de travail : le premier dédié à la 2D, le second à la 3D.

Dans le cadre du volet méthodologique, le CNES a décidé de développer une bibliothèque logicielle regroupant les outils nécessaires au traitement des données ORFEO. Cette bibliothèque est composée d'un ensemble de briques algorithmiques allant de la lecture et l'écriture d'images géoréférencées à la classification d'images segmentées, couvrant ainsi l'intégralité des tâches réalisables sur des données ORFEO.

Cette bibliothèque logicielle, appelée ORFEO ToolBox (OTB) est écrite en langage C++ et repose sur la bibliothèque de traitement d'images ITK en y ajoutant les outils nécessaires au traitement d'images de télédétection, à l'aide de bibliothèques logicielles dédiées (Gdal, Ossim). OTB est une bibliothèque libre sous la licence CeCILL⁵. Les fonctionnalités principales de OTB sont le support des très grandes images via l'utilisation de méthodes de parallélisation et de streaming, le traitement d'images (filtrage, segmentation, extraction d'information), l'analyse de données (classification, détection de changement). La documentation (API, cas d'utilisation, applications) est complète et détaillée. De plus, OTB dispose d'une grande communauté de développeurs et d'utilisateurs, très efficace et très réactive.

4 Positionnement

Nous avons vu dans la section 1 qu'il existe un grand nombre de méthodes pour segmenter et analyser les images. Les approches *simples* utilisent une seule image à une résolution donnée alors que les approches *complexes* reposent sur l'utilisation conjointe de plusieurs images, plusieurs résolutions ou encore plusieurs sources de données (images optiques, radar, laser, SIG). Ces méthodes sont parfaitement applicables aux images de télédétection.

Nous avons vu dans la section 2 que l'analyse de paysages urbains est un domaine de recherche actif. L'utilisation d'images à très haute résolution spatiale est une évolution à double tranchant. Bien qu'elle donne accès à des images bien plus détaillées, la quantité de bruit en contrepartie est non négligeable, et l'augmentation de la résolution complexifie l'analyse de par le volume important de données à traiter. Les méthodes les plus efficaces nécessitent plusieurs sources d'informations ou reposent sur la connaissance d'un expert. Un processus intégralement automatique serait impossible à mettre en place et un processus guidé nécessiterait que l'expert règle manuellement des paramètres en fonction de l'image ou de la zone à analyser. L'utilisation d'un système d'information géographique permet de s'affranchir de certaines limites (paramétrisation, limitation de l'espace des données, ...) ce qui représente un avantage important dans la mise en place d'un processus le plus automatisé possible.

Cette thèse cofinancée par le CNES s'inscrit directement dans les problématiques du programme ORFEO. L'analyse de paysages urbains est un axe de travail du groupe thématique GT3 (cartographie et aménagement du territoire) et l'intégration d'informations contextuelles dans un processus de raisonnement fait partie des outils que le CNES aimerait intégrer dans OTB.

5. Licence de logiciel libre développée par les organismes de recherche français (CEA, CNRS, INRIA), similaire à la GPL. CeCILL est l'acronyme de **C**EA **C**NRS **I**NRIA **L**ogiciel **L**ibre.

2

La fouille de données pour l'analyse d'images

Sommaire

1	La fouille de données pour aider l'expert	35
1.1	Algorithmes de classification supervisée	36
1.2	Algorithmes de classification non supervisés	37
1.3	La fouille de données appliquée à la télédétection	39
2	Utilisation de connaissance dans un processus de classification . .	41
2.1	Clustering guidé par la connaissance	41
2.2	Connaissance spatiale	42
3	Positionnement	48

À l'heure actuelle, l'analyse de données n'est plus réalisable manuellement en temps fini par un individu ou un groupe d'individus. Qu'il s'agisse d'images de télédétection à très haute résolution spatiale ou de bases de données dans le nuage, la masse de données à traiter est telle qu'il faudrait un temps infini. Avec plus de 1.2 milliard d'utilisateurs mensuels actifs, 800 millions d'utilisateurs actifs quotidiens et 10 milliards de messages envoyés par jour dans le monde¹, Facebook en est le parfait exemple.

Pour pouvoir analyser ces données massives nous avons besoin d'algorithmes capables de traiter un grand volume d'informations en peu de temps. Nous présentons dans la section 1 plusieurs familles de méthodes de classification, dont certaines font appel à la connaissance d'un expert du domaine. Nous parlons dans la section 2 de certaines de ces méthodes et nous évoquons l'utilisation de ces méthodes sur des problèmes liés à la télédétection. Nous parlons de connaissances spatiales et des façons de la formaliser. Enfin, nous concluons en positionnant nos travaux par rapport aux travaux présentés dans ce chapitre.

1 La fouille de données pour aider l'expert

Il existe deux grandes familles de méthodes de fouille de données : les méthodes de classification supervisée et les méthodes de classification non supervisée (algorithmes de clustering).

1. Chiffres officiels au premier trimestre 2014

1.1 Algorithmes de classification supervisée

Les méthodes de classification supervisée sont utilisées lorsqu'il existe une catégorisation *a priori* des données. Ces méthodes reposent sur l'utilisation d'exemples étiquetés. Le but est de créer un modèle représentant au mieux l'ensemble des données. Ce modèle est appris à l'aide des exemples étiquetés. Les exemples sont des instances de classe, une classe ayant une sémantique dans le domaine d'expertise. Par exemple, dans le domaine de la télédétection, une classe peut être *bâtiment* ou encore *route*.

Le but de ces méthodes est d'apprendre dans un premier temps un modèle prédictif à partir d'un jeu d'exemples, et d'utiliser dans un second temps ce modèle pour la classification d'un jeu de données. Parmi les grandes familles de méthodes de classification supervisée, nous pouvons notamment citer les séparateurs à vaste marge, les réseaux de neurones et les arbres de décision.

1.1.1 Séparateurs à Vaste Marge

Le principe des méthodes basées sur les séparateurs à vaste marge (abrégé SVM, de l'anglais Support Vector Machine, (Schölkopf et Smola, 2002)) repose sur la recherche d'un hyperplan séparant au mieux des exemples étiquetés dans l'espace des données. Les SVM se basent sur deux idées principales : la recherche de la marge qui maximise la séparation entre deux échantillons de classes différentes et la transformation de l'espace des données pour augmenter le nombre de dimensions.

La marge maximale permet à elle seule de résoudre les problèmes linéairement séparables. Lorsque le problème n'est pas linéairement séparable, l'augmentation du nombre de dimensions permet de le transformer en un problème linéairement séparable. Cette dernière étape est réalisée à l'aide d'une fonction noyau qui doit respecter le théorème de Mercer (1909)².

1.1.2 Réseaux de neurones

Les méthodes basées sur les réseaux de neurones artificiels (Bishop, 1996) reposent sur le paradigme du neurone formel. Un neurone reçoit plusieurs signaux en entrée et les combine en un signal de sortie. Un réseau de neurones est organisé en couches (trois minimum). La première couche représente les entrées (un neurone par variable du problème), la dernière couche représente les sorties (un neurone par classe, pour un problème de classification). Les autres couches, appelées couches cachées, réalisent la combinaison des entrées jusqu'à la couche de sortie. Le nombre de couches cachées et de neurones par couche sont variables, ils peuvent être fixés par l'utilisateur ou peuvent évoluer pendant l'exécution.

Un neurone est défini par deux fonctions : la fonction de combinaison et la fonction d'activation. Dans les réseaux de type *perceptron multi-couche* la combinaison est linéaire, dans les réseaux de type *fonction à base radiale* la combinaison calcule la distance entre les entrées. La fonction d'activation sert à briser la linéarité dans le fonctionnement d'un neurone. La fonction sigmoïde, la tangente hyperbolique et la fonction de Heaviside sont parmi les fonctions d'activation les plus courantes dans la littérature. La phase d'apprentissage est réalisée par rétropropagation du gradient de l'erreur, en partant de la couche de sortie jusqu'à la couche d'entrée.

1.1.3 Arbres de décision

Un arbre de décision (Breiman *et al.*, 1984; Quinlan, 1993) est une méthode de partitionnement récursive. Le modèle créé par ce type de méthodes prend la forme d'un arbre. À chaque

2. Le théorème de Mercer est une représentation d'une fonction symétrique de type positif par le carré d'une série convergente de produits de fonctions.

noeud interne, une variable d'entrée est sélectionnée (selon un critère de segmentation prédéfini), partageant le jeu de données en groupes homogènes. Une feuille de l'arbre correspond à une variable cible (la classe à attribuer). Le partitionnement s'arrête lorsque tous les éléments d'un groupe ont la même variable cible ou lorsque le critère de segmentation ne permet plus d'améliorer les résultats. Un chemin de la racine à une feuille donne la combinaison des variables d'entrées nécessaires pour que la classe correspondant à la feuille soit affectée à un élément du jeu de données.

La création de l'arbre optimal est un problème NP-complet³, le critère de segmentation dépend donc d'une heuristique. Il n'est donc pas possible de vérifier que l'arbre de décision appris est l'optimum global.

1.1.4 Limites de la classification supervisée

Bien que très efficaces, les méthodes de classification supervisée sont limitées par plusieurs points. Premièrement, la classification repose sur un étiquetage *a priori* de tout ou partie des données. Cet étiquetage est dépendant de l'utilisateur et de sa connaissance des données et du problème. Cependant cela induit un biais dans le processus car deux utilisateurs différents peuvent interpréter différemment les données. De plus, cet étiquetage des données peut se révéler fastidieux pour des données massives ou complexes. Finalement, la portabilité d'un modèle appris sur un jeu de données appliqué sur un autre jeu de données n'est pas assurée.

Un autre point important à soulever est que les modèles produits par certains de ces algorithmes sont difficiles à interpréter par un expert. Les réseaux de neurones et les séparateurs à vaste marge fonctionnent comme une *boite noire* qui prend des données en entrées et donne un résultat en sortie, mais le raisonnement interne de l'algorithme est trop complexe (*i.e.* fonctionnement d'un neurone, augmentation des dimensions du problème) pour pouvoir être lié à une sémantique dans le domaine d'expertise. Un modèle produit par un arbre de décision est *a contrario* facile à comprendre. Cependant, ce type de méthodes est sujette au surapprentissage : le modèle appris est trop spécifique et est difficilement généralisable. De plus certains problèmes sont difficiles à exprimer sous forme d'arbre de décision, et le modèle produit peut devenir très large.

1.2 Algorithmes de classification non supervisés

Les méthodes de classification non supervisée, ou clustering, sont utilisées lorsqu'il n'y a pas d'exemples ni de connaissances *a priori* sur le jeu de données. Ces méthodes reposent sur l'analyse globale du jeu de données dans le but de faire émerger des groupes homogènes au sein des données. Un groupe, ou cluster, n'a pas de sémantique associée (contrairement aux classes, voir section 1.1) mais est un regroupement d'objets qui maximise un critère d'homogénéité.

L'objectif de ces méthodes est de proposer des regroupements minimisant l'inertie intra-classe (chaque groupe est le plus homogène possible) et maximisant l'inertie inter-classe (les groupes sont le plus différents les uns des autres). Pour ce faire, les méthodes de classification non supervisée reposent sur l'utilisation d'une mesure de distance (ou dissimilarité) entre objets. Parmi les distances les plus utilisées, nous pouvons notamment citer la distance euclidienne, la distance de Manhattan et la distance de Ward. Cependant, la minimisation de l'inertie intra-classe est optimale lorsque les groupes ne sont composés que d'un seul élément, et la maximisation de l'inertie inter-classe est optimale lorsque tous les éléments sont regroupés dans un seul groupe.

3. Un problème NP-complet est un problème dont il existe une solution vérifiable en un temps polynomial, mais dont il n'est pas possible de trouver cette solution en temps polynomial.

Le problème est alors de trouver un juste milieu entre maximisation des critères d'inertie et regroupement des données.

Nous pouvons séparer les principales méthodes de classification non supervisée en deux grandes familles : les méthodes de partitionnement et les méthodes hiérarchiques.

1.2.1 Méthodes de partitionnement

Dans ce type de méthode, l'ensemble des données est partagé en groupes homogènes. La littérature est très riche dans le domaine et présente des méthodes basées sur des prototypes. Un prototype est le représentant d'un groupe d'objets. Il peut faire partie ou non de ce groupe.

Parmi les méthodes basées sur l'utilisation de prototype, nous pouvons notamment citer l'algorithme K-MEANS (MacQueen, 1967; Jain, 2010), qui cherche à partager le jeu de données en k groupes. Pour ce faire, k centres sont créés aléatoirement dans l'espace des données. Chaque objet du jeu de données se voit attribuer au centre le plus proche (d'après la distance choisie) et le centre est ensuite réaffecté à la moyenne du groupe qui lui est attribué. Le processus itère jusqu'à un critère d'arrêt (*i.e.* nombre d'itération maximum, pas de changement entre deux itérations).

Il existe plusieurs variations de cette méthode. Parmi celles-ci, nous pouvons citer l'algorithme K-MÉDIANE (Jain et Dubes, 1988), dans lequel l'objet moyen d'un groupe est déplacé vers la médiane du groupe plutôt que la moyenne. Nous pouvons aussi citer l'algorithme K-MÉDOÏDE (Kaufman et Rousseeuw, 1987), dans lequel les centres ne sont pas choisis aléatoirement dans l'espace des données mais directement dans les données. Le centre de chaque groupe est l'individu le plus représentatif du groupe. Le principe général de ces méthodes est résumé dans l'algorithme 3.

Algorithme 3: Méthodes de partitionnement basées sur des centres

Entrées : $k \rightarrow$ le nombre de groupes
 choisir aléatoirement k centres
tant que le critère d'arrêt n'est pas atteint **faire**
 | affecter chaque objet au centre le plus proche
 | modifier les centres en fonction des objets associés

Le principal problème de ces méthodes repose sur le choix du nombre de groupes. En effet, le choix de k n'est pas anodin et a une forte influence sur les résultats des algorithmes : si k est trop petit certains groupes ne seront pas suffisamment homogènes, si k est trop grand certains objets similaires se verront séparés en deux groupes. Le choix du nombre de groupes est l'une des principales problématiques dans la littérature sur le clustering.

Pour s'affranchir de cette limite, Ball *et al.* (1965) proposent l'algorithme ISODATA qui est une évolution de l'algorithme K-MEANS. Dans cet algorithme le nombre de centres est variable lors de l'exécution. Deux centres jugés trop proches ou avec trop peu d'objets peuvent être fusionnés. À l'inverse, le centre d'un groupe jugé trop hétérogène peut se voir séparer en deux. Ceci permet de répondre à la problématique du choix du nombre de centres. Cependant la méthode introduit un grand nombre de paramètres (*i.e.* seuil de proximité, seuil d'homogénéité), déplaçant le problème du choix du nombre de centres au choix des valeurs des paramètres.

1.2.2 Méthodes hiérarchiques

Dans ce type de méthodes, les données sont regroupées entre elles pour former des groupes homogènes. Parmi ces méthodes nous pouvons citer la Classification Ascendante Hiérarchique, abrégé CAH (Lance et Williams, 1967; Müllner, 2011). Cet algorithme part des données et fusionne itérativement les deux individus ou groupe d'individus les plus proches (selon une mesure de distance ou dissimilarité).

Lorsque l'algorithme manipule des groupes d'objets, il existe plusieurs solutions pour calculer la distance entre les groupes. Nous pouvons citer le saut minimum (single link, distance minimale entre les objets des deux groupes), le saut maximum (complete link, distance maximale entre les objets des deux groupes), le lien moyen (mean link, la distance entre la moyenne des deux groupes), la distance de Ward (Ward (1963), la différence d'inertie avant et après fusion de deux classes). Le lecteur intéressé peut se référer à l'article de Müllner (2011) pour une présentation détaillée de ces méthodes de calculs de distance.

CAH produit en sortie un arbre binaire de partition, séparant récursivement l'ensemble des données en deux jusqu'à arriver aux objets individuels. Le principal problème de ce type d'approche est de choisir comment couper l'arbre pour obtenir des groupes pertinents. Ceci peut être fait en fixant un nombre de groupes recherchés, par une coupe de l'arbre faite par un expert ou encore en fixant un seuil de distance. Dans tous les cas, cela implique un réglage des paramètres ou la supervision d'un expert.

1.2.3 Combinaison de méthodes non supervisées

Nous venons de voir que les méthodes de classification non supervisée dépendent d'un critère d'arrêt ou d'un nombre de classes initial. Ceci rend ces méthodes difficiles à exploiter lorsque l'utilisateur ne dispose pas de suffisamment d'informations sur le jeu de données. Pour palier ce problème, Wemmert (2000) propose de combiner les résultats de plusieurs classifieurs. Chaque classifieur vote et si un consensus est trouvé, les résultats sont unifiés. Ceci permet de mettre à jour les classifieurs dans un processus itératif. Le résultat final est un ensemble de classifications aux résultats similaires (mais pas nécessairement identiques). Cette méthode permet de résoudre le problème du choix des paramètres initiaux en faisant converger les classifieurs vers une classification commune. Cette méthode a été améliorée pour gérer les données multi-sources (Gançarski et Wemmert, 2007) et l'intégration de la connaissance d'un expert (Forestier, 2010).

1.3 La fouille de données appliquée à la télédétection

Les méthodes de fouille de données sont utilisables pour analyser des images de télédétection, que ce soit au niveau pixel ou au niveau région.

Dans leurs travaux, Benediktsson *et al.* (1990) comparent les réseaux de neurones aux méthodes statistiques (classification Bayésienne). Ils partent d'images multispectrales et intègrent des informations multisources (données radar, pente, élévation) dans une classification des pixels. Les auteurs montrent la supériorité des réseaux de neurones en terme de précision de classification. Heermann et Khazenie (1992) s'intéressent à la pertinence de l'utilisation des réseaux de neurones pour des problèmes de classification pixel sur des images multispectrales LANDSAT à trois bandes. Pour cela ils utilisent des images de 2048×2048 pixels. Les auteurs montrent que les réseaux de neurones sont beaucoup plus rapides que des méthodes statistiques ou de clustering.

Wang (1990) propose une classification floue appliquée sur des images LANDSAT de 256×256 pixels. Sept classes sont définies. La méthode proposée produit sept matrices d'appartenance (une par classe) représentant le degré d'appartenance des pixels à chacune des classes. L'auteur

souligne l'importance de la représentation des données dans le cas de l'analyse d'images de télédétection.

Yang *et al.* (2003) s'intéressent à la pertinence de l'utilisation des SVM pour la classification d'images hyperspectrales. Les auteurs comparent leurs résultats à une méthode basée sur les réseaux de neurones et une autre basée sur les k plus proches voisins. Ils s'intéressent plus particulièrement aux approches multiclassées combinant plusieurs SVM (un contre tous, un contre un, arbre binaire équilibré, arbre binaire un contre tous). Ils testent leur méthode sur une image aérienne à 220 bandes et concluent sur l'avantage des SVM sur les autres types d'approches. Ils soulignent le fait que les stratégies multiclassées n'ont qu'un impact mineur sur les performances. Melgani et Bruzzone (2004) proposent une méthode active pour minimiser le nombre de pixels nécessaires à l'entraînement d'un SVM. La méthode propose à l'expert d'étiqueter des objets très proches et très éloignés de l'hyperplan séparant les données. Les auteurs montrent l'efficacité de leur méthode en soulignant sa faible tolérance aux erreurs (pixels mal étiquetés).

Dans le chapitre 1 nous avons présenté des méthodes de segmentation. Les algorithmes de fouille de données peuvent très bien s'appliquer sur des segments.

Une méthode de classification non supervisée peut être utilisée pour proposer une classification des régions. Gaetano *et al.* (2009) réalisent une segmentation basée sur deux classifications pixels. Dans un premier temps, les auteurs utilisent l'algorithme K-MEANS pour faire un clustering dans le domaine spectral. Dans un second, ils raffinent la segmentation précédente en effectuant un deuxième K-MEANS dans le domaine spatial pour classer les pixels selon leurs voisinages. Ceci leur permet de générer une segmentation hiérarchique. Kurtz *et al.* (2010, 2012) proposent une méthode multirésolution pour caractériser un résultat de segmentation hiérarchique. Les auteurs utilisent l'algorithme K-MEANS sur des histogrammes de composition pour classer récursivement les régions à chaque niveau de la hiérarchie. La méthode est validée sur une image QUICKBIRD et permet d'extraire des objets d'intérêt à différents niveaux (pavillon, îlot urbain).

Une méthode de classification supervisée peut être utilisée pour entraîner des détecteurs. Bacher et Mayer (2005) proposent une méthode de détection de routes basée sur un algorithme de détection de lignes. Les lignes détectées servent à entraîner un algorithme de classification supervisée floue. La méthode est testée sur des images QUICKBIRD et IKONOS.

Plusieurs méthodes peuvent être combinées. Poulain (2010) utilise plusieurs détecteurs et algorithmes de classification (SVM) sur une segmentation produite par l'algorithme MEANSHIFT et les combine à l'aide de la théorie des croyances de Dempster-Shafer. L'intérêt de cette méthode est de pouvoir gérer l'incertitude propre aux différents processus entrant en jeu dans le processus d'analyse (segmentation, détecteurs, classifieurs). Gañçarski et Wemmert (2007) combinent les résultats de plusieurs méthodes de classification non supervisée pour classifier une image de télédétection. Le processus itère de manière à faire émerger un consensus entre les différents classifieurs. La méthode est testée sur une image SPOT d'une résolution de 20m.

Problèmes liés à l'analyse d'images de télédétection. Dans ce chapitre, nous avons pu voir que la classification et plus particulièrement la classification d'images de télédétection a fait ses preuves depuis plusieurs décennies. L'application d'une analyse basée pixel sur des images basse ou moyenne résolution donne de bons résultats. L'arrivée de la haute résolution a été suivie par l'émergence des méthodes d'analyse d'images orientées région et les algorithmes de classifications se sont adaptés au travail conjoint avec la segmentation. La connaissance de l'expert a pu être intégrée dans les algorithmes de segmentation et de classification.

Cependant, avec l'arrivée de la très haute résolution spatiale (*i.e.* PLEIADES), la masse de données à traiter est devenue considérable. Une image multispectrale PLEIADES représente plusieurs giga octets de données et la segmentation d'une telle image produit des régions très sur-

segmentées. De plus, les capteurs PLEIADES ont une capacité d'acquisition de plusieurs centaines d'images par jour. Ainsi le volume de données à traiter est très important et les algorithmes de classification doivent s'adapter à ce volume de données. Ceci est d'autant plus vrai dans le cas des approches combinant plusieurs sources (*i.e.* image THRS + SIG, multi-résolution, multi-temporel).

2 Utilisation de connaissance dans un processus de classification

Une méthode de fouille de données est utilisée pour proposer une solution à un problème particulier dans un domaine d'expertise donné. Nous avons vu dans la section précédente qu'il existe des méthodes supervisées par un expert et d'autres méthodes non supervisées. Dans les méthodes supervisées, l'expert intervient directement en donnant des exemples.

L'utilisation de la connaissance de l'expert ne passe pas obligatoirement par son intervention directe. Nous avons vu dans le chapitre 1 que la connaissance de l'expert peut être utilisée pour aider des algorithmes de segmentation. Cette connaissance peut prendre la forme d'une base de données (*i.e.* SIG dans le cas de l'analyse d'image de télédétection), d'une ontologie (ou plus généralement, d'une hiérarchie de concepts), d'une vérité terrain. Ces mêmes outils peuvent être utilisés pour aider des algorithmes de fouille de données.

2.1 Clustering guidé par la connaissance

Les méthodes de clustering utilisant de la connaissance sont appelées méthodes *semi-supervisées* ou méthodes *hybrides*. Ces méthodes peuvent être regroupées en deux familles : les méthodes par contraintes et les méthodes utilisant des objets étiquetés. La formalisation de la connaissance dans ces méthodes peut être simple (description de contraintes, étiquetage manuel d'objets) mais elle peut aussi passer par l'utilisation de dictionnaires de données, de hiérarchies de concepts ou d'ontologies.

2.1.1 Clustering par contraintes

Dans ce type de clustering, la connaissance est représentée par des contraintes exprimant la relation qui existe entre deux objets du jeu de données. Cette relation peut être de deux types. La relation *must-link* indique que les deux objets doivent être dans le même cluster à la fin de l'algorithme. La relation *cannot-link* indique que les deux objets doivent être dans deux clusters distincts à la fin de l'algorithme. L'avantage de ce type de méthode est qu'il n'y a pas besoin d'avoir d'information *a priori* sur le nombre de classes, les étiquettes ou les propriétés de ces classes, seule la relation entre paires d'objets est prise en compte.

Wagstaff *et al.* (2001) proposent des versions contraintes des algorithmes K-MEANS et COBWEB. Les relations *must-link* et *cannot-link* sont utilisées pour biaiser les classifieurs. Les deux méthodes ne retournent un résultat que si toutes les contraintes sont satisfaites.

Bilenko *et al.* (2004) utilisent les deux types de contraintes pour apprendre une fonction de distance. La distance diminue avec un lien *must-link* et augmente avec un lien *cannot-link*. La méthode proposée, appelée MPCK-KMEANS, apprend différentes fonctions de distance pour chacun des clusters.

Kumar et Kummamuru (2008) proposent d'utiliser un autre type de connaissances, les comparaisons relatives, à travers un algorithme de classification non supervisée. Leur étude expérimentale démontre que l'algorithme proposé est plus précis et robuste avec ces relations qu'avec les contraintes *must-link* et *cannot-link*.

2.1.2 Clustering utilisant des objets étiquetés

Forestier *et al.* (2010) proposent une méthode de clustering collaboratif dans laquelle des objets étiquetés interviennent à trois niveaux. Dans le premier niveau, une étiquette est attribuée à chaque cluster (plutôt qu'aux objets eux-mêmes). Cette étape est facile à mettre en place car elle ne nécessite qu'un traitement *a posteriori* des résultats. Dans le deuxième niveau, la connaissance est utilisée pour guider le processus collaboratif en évaluant le résultat des classifications individuelles et en guidant le choix des modifications à apporter aux algorithmes. Dans le troisième niveau, la connaissance est utilisée directement au sein des différentes méthodes de clustering. Cette étape est la plus coûteuse car elle implique de modifier, parfois en profondeur, le fonctionnement des algorithmes de clustering.

Kurtz *et al.* (2013) propose d'intégrer la connaissance de l'expert dans la mesure de distance. Les objets sont définis par un histogramme de composition. À l'aide d'une matrice de dissimilarité fournie par l'expert, les auteurs calculent hiérarchiquement une distance barre à barre dans les histogrammes de composition. La méthode proposée, appelée HSBD (de l'anglais Hierarchical Semantic-Based Distance), est utilisée pour introduire de la connaissance dans la méthode de Kurtz *et al.* (2012). Dans cette méthode, des objets sont étiquetés à un niveau et servent à définir l'histogramme de composition au niveau supérieur.

2.2 Connaissance spatiale

Nous venons de voir qu'il existe des méthodes de fouille de données qui utilisent la connaissance d'un expert. Nous cherchons à utiliser la connaissance de l'expert pour analyser des images de télédétection. Nous avons vu dans ce chapitre et dans le chapitre 1 qu'il existe de nombreuses méthodes de segmentation qui utilisent de la connaissance, dont certaines font collaborer segmentation et classification.

Parmi la connaissance disponible dans le domaine de la télédétection, un type de connaissance nous semble particulièrement intéressant, bien qu'encore peu utilisé à l'heure actuelle. Les objets d'intérêt pour l'expert ont la particularité d'être organisés spatialement. En effet, une image de télédétection peut représenter une ville. Cette ville va contenir des quartiers, qui eux mêmes sont composés d'îlots urbains. Ces îlots contiennent des bâtiments qui peuvent être alignés, parallèles, plus ou moins densément espacés, adjacent à des zones de végétation ou proche d'une route. Ainsi les objets d'intérêt sont placés dans un contexte riche, défini par des relations spatiales dans un même niveau (*i.e. adjacence, distance*) ou des relations de composition (*i.e. compose, est composé de*).

2.2.1 Alignement et parallélisme

Un motif particulièrement intéressant dans un paysage urbain est celui de l'alignement d'objets complexes, tel l'alignement de maisons, d'arbres ou encore de voitures. C'est dans ce cadre précis que se placent les travaux de Vanegas *et al.* (2010a,b). Le principe de cette méthode est de regrouper des couples d'objets ayant la même orientation. Cependant, nous pouvons noter que dans un paysage urbain, les objets d'un même type n'ont pas forcément la même taille, la même forme, ou ne sont pas forcément parfaitement alignés. C'est pourquoi il faut compter sur un raisonnement flou, dans lequel des objets peuvent être *approximativement* alignés. Les auteurs proposent pour cela une mesure de similarité entre histogrammes d'orientation. Cette mesure de similarité peut être étendue pour évaluer la similarité entre un ensemble d'histogrammes d'orientation.

Pour plus d'efficacité, cette méthode peut être couplée à une méthode de détection d'objets parallèles. Une telle méthode est présentée par Vanegas *et al.* (2009). Cette méthode définit

d'une part le parallélisme entre deux objets, et entre un objet et un alignement d'objets. Encore une fois, en considérant à la fois l'imprécision de l'image et le fait que dans le monde réel tout ne soit pas rigoureusement aligné, il convient d'utiliser un raisonnement flou, dans lequel on s'intéresse plus au degré de parallélisme entre deux objets ou groupe d'objets qu'au parallélisme au sens mathématique du terme, précis, symétrique et transitif. En utilisant cette méthode, ainsi que l'algorithme de détection de routes défini par [Christophe et Inglada \(2007\)](#), [Vanegas et al. \(2010a\)](#) raffinent la détection d'alignement de maisons en considérant qu'un alignement de maisons est forcément parallèle et proche d'une route. De ce fait, les auteurs réduisent le nombre de groupes et éliminent les faux positifs autant du côté des alignements de maisons que des morceaux de routes.

2.2.2 Relation entre objets de différents niveaux

Inclusion. Lorsque nous nous plaçons dans une hiérarchie d'objets, nous pouvons considérer la notion d'inclusion d'un objet de niveau inférieur dans un objet de niveau supérieur. [Papadias et Egenhofer \(1997\)](#) propose une méthode utilisant cette notion d'inclusion pour inférer la position relative d'objets en montant ou descendant dans la hiérarchie.

Pour qualifier la position relative d'objets, les auteurs utilisent les neuf primitives (quatre points cardinaux, quatre combinaisons de points cardinaux consécutifs, identité) définies par [Freksa \(1992\)](#). En considérant la notion de tangence à un axe défini par un côté d'un objet, ainsi que les treize relations mutuellement exclusives entre deux intervalles définies par [Allen \(1983\)](#) et étendues dans le plan, les auteurs obtiennent 169 relations primitives. À partir de ces définitions, les auteurs proposent deux algorithmes, l'un pour inférer les positions d'objets du niveau le plus bas (point, dans la suite de cette section), l'autre pour inférer les positions d'objets d'autres niveaux (région, dans la suite de cette section). Ces deux méthodes sont présentées succinctement ici. Dans le cas des régions, les auteurs restreignent la relation entre deux objets à l'aide d'une table de composition. Dans le cas des points, l'inférence peut être formulée comme un problème de *path consistency* dans un réseau de contraintes. Chaque contrainte est une disjonction de relations primitives qui représente les relations permises entre deux points. Un tel algorithme a besoin de règles de compositions pour les relations de direction. Les règles utilisées ici permettent d'inférer la position relative de deux points par rapport à un troisième ([Freksa, 1992](#)).

Region connection calculus. La méthode du Region Connection Calculus (RCC) décrite par [Randell et al. \(1992\)](#) et utilisée entre autre par [Renz et Nebel \(1999\)](#) et [Inglada et Michel \(2009\)](#) est une méthode qui permet à la fois d'analyser les relations entre objets d'un même niveau et les objets de niveaux distincts (sans notion de hiérarchie). Les auteurs décrivent les objets par leurs relations entre eux. Dans cette section nous nous intéressons plus particulièrement à RCC-8, un sous ensemble de RCC qui considère huit relations basiques entre deux objets. Ces huit relations sont décrites dans la table 2.1. Ces relations couvrent l'ensemble des relations possibles entre deux objets. Autrement dit, quelque soit A et B deux objets, il existe au moins une relation entre A et B .

Ces huit relations sont illustrées dans la figure 2.1.

Ces relations formant un ensemble, nous pouvons définir les opérateurs d'union (\cup), d'intersection (\cap), d'inverse ($^{-1}$) et de composition (\circ). Ces opérateurs sont décrits formellement dans la table 2.2. Pour plus de précision sur la notion d'ensemble comme formalisation de la connaissance, se référer à la section 2.2.3.

Il existe d'autres sous-ensemble de RCC. Citons par exemple RCC-5, dans lequel la notion de tangence n'est pas considérée. L'ensemble de règles de RCC-5 est le suivant

Notation	Relation
DC	Non connexe
EC	Connexe
EQ	Égal
PO	Superposition partielle
TPP	Partie propre tangente
TPP^{-1}	Partie propre tangente inverse
$NTPP$	Partie propre non tangente
$NTPP^{-1}$	Partie propre non tangente inverse

TABLE 2.1 – Huit relations de la méthode RCC-8. (Renz et Nebel, 1999)

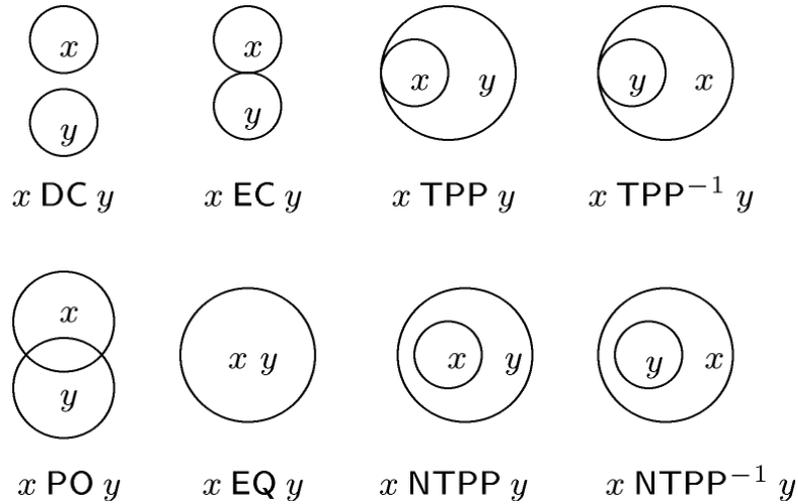


FIGURE 2.1 – Les huit relations de la table 2.1 illustrées par des exemples. (Renz et Nebel, 1999)

$$\begin{aligned}
\forall X, Y : X(R \cup S)Y &\leftrightarrow XRY \vee XSY \\
\forall X, Y : X(R \cap S)Y &\leftrightarrow XRY \wedge XSY \\
\forall X, Y : XR^{-1}Y &\leftrightarrow YRX \\
\forall X, Y : X(R^\circ S)Y &\leftrightarrow \exists Z : (XRZ \wedge ZSY)
\end{aligned}$$

TABLE 2.2 – Opérateurs applicables sur l'ensemble des relations RCC-8.

$\{DC, PO, EQ, NTPP, NTPP^{-1}\}$. Ou encore RCC-7, dans lequel la relation de superposition partielle est exclue.

2.2.3 Représentation de la connaissance spatiale

Dans cette section, nous décrivons différents formalismes de représentation de la connaissance, applicables aux relations spatiales. Dans certains cas, les relations spatiales n'ont pas besoin d'être formalisées. Lorsqu'une relation particulière est recherchée, un simple indice suffit pour signifier que des objets sont liés ou non par la relation en question. C'est notamment le cas dans les

travaux de Vanegas *et al.* (2010a,b) où un indice entre 0 et 1 est donné pour mesurer le degré d'alignement d'un groupe d'objets, ou encore dans les travaux de Vanegas *et al.* (2009) où un indice donne le degré de parallélisme entre deux objets / groupes d'objets.

Dans les cas où les relations sont plus complexes ou multiples, le formalisme prend toute son importance. C'est ce que nous allons présenter dans cette section.

La logique mathématique, ou logique formelle, est une discipline des mathématiques qui définit et étudie les représentations formelles du langage mathématique. Les objets fondamentaux de la logique mathématique sont les formules modélisant les énoncés mathématiques, les dérivations modélisant les raisonnements mathématiques et les sémantiques établissant les liens entre ces modèles et les objets qu'ils sont censés représenter.

Un système logique ou système de déduction est constitué de trois composantes. Les deux premières définissent sa syntaxe, la troisième sa sémantique :

- Un ensemble de formules, ou faits ; dans les systèmes de logique classique ou intuitionniste, les formules représentent des énoncés mathématiques exprimés formellement. Les formules sont définies par des moyens combinatoires : suites de symboles, arbres étiquetés, graphes...
- Un ensemble de déductions ; les déductions sont également définies par des moyens combinatoires. Une déduction permet de dériver des formules (les formules prouvables ou théorèmes) à partir de formules de départ (les axiomes) au moyen de règles (les règles d'inférence).
- Une interprétation des formules ; il s'agit d'une fonction associant à toute formule un objet dans une structure abstraite appelée modèle, ce qui permet de définir la validité des formules.

Nous proposons ici de définir formellement l'ensemble de relations de RCC-8 (voir section 2.2.2) à l'aide de deux logiques. Premièrement à l'aide de la logique modale, deuxièmement à l'aide de la logique de description.

Logique modale. Une logique modale LM est une logique à laquelle on ajoute des modificateurs, qui vont permettre de modifier le sens d'une proposition. Dans le cas de RCC-8, nous considérons que la sémantique d'une proposition est un objet de l'espace, les formules représentent les relations entre les objets.

Renz et Nebel (1999) utilisent un tel formalisme en considérant qu'un objet ne peut pas être vide, que deux objets sont connexes s'ils partagent un point et se superposent s'il partagent un point intérieur. Pour définir l'intérieur il est nécessaire de définir l'opérateur logique modale I . Cet opérateur doit satisfaire les axiomes suivants :

- $IX \rightarrow X$
- $IIX \leftrightarrow IX$
- $I\top \leftrightarrow \top$
- $I(X \wedge Y) \leftrightarrow IX \wedge IY$

où X et Y sont des propositions et \top représente la tautologie (proposition toujours vraie).

Soit f une formule de LM , on appelle *contrainte de modèle* toute formule telle que $f = \top$ et *contrainte d'implication* toute formule telle que $f \neq \top$. Ces contraintes sont encodées en logique modale dans le tableau 2.3.

Logique de description. Une logique de description LD fait partie de la famille des logiques modales. Une logique de description est un langage de représentation de connaissance qui peut être utilisé pour représenter la connaissance terminologique d'un domaine d'application d'une façon structurée et formelle. Une caractéristique d'une telle logique est sa sémantique formelle définie en logique du premier ordre.

La plupart des logiques de description divisent la connaissance en deux parties :

Relation	Contraintes de modèle	Contraintes d'implication
$DC(X, Y)$	$\neg(X \wedge Y)$	$\neg X, \neg Y$
$EC(X, Y)$	$\neg(IX \wedge IY)$	$\neg(X \wedge Y), \neg X, \neg Y$
$PO(X, Y)$	–	$\neg(IX \wedge IY), X \rightarrow Y, Y \rightarrow X, \neg X, \neg Y$
$TPP(X, Y)$	$X \rightarrow Y$	$X \rightarrow IY, Y \rightarrow X, \neg X, \neg Y$
$NTPP(X, Y)$	$X \rightarrow IY$	$Y \rightarrow X, \neg X, \neg Y$
$EQ(X, Y)$	$X \rightarrow Y, Y \rightarrow X$	$\neg X, \neg Y$

TABLE 2.3 – Six relations de base de RCC-8 encodées en logique modale. Les deux relations manquantes sont données par $TTP^{-1}(X, Y) = TPP(Y, X)$ et $NTTP^{-1}(X, Y) = NTPP(Y, X)$.

- les informations terminologiques : définition des notions basiques ou dérivées et de comment elles sont reliées entre elles. Ces informations sont "génériques" ou "globales", vraies dans tous les modèles et pour tous les individus.
- les informations sur les individus : ces informations sont "spécifiques" ou "locales", vraies pour certains individus particuliers.

Toutes les informations connues sont alors modélisées comme un couple $\langle T, A \rangle$, où T est un ensemble de formules relatives aux informations terminologiques (la T-Box) et A est un ensemble de formules relatives aux informations sur les assertions (la A-Box). Une autre manière de voir la séparation entre ces informations, est d'associer la T-Box aux règles qui régissent notre monde (e.g., la physique, la chimie, la biologie, ...), et d'associer les individus de notre monde à la A-Box (e.g., Jean, Marie, un chat, ...).

Les logiques de description utilisent un certain nombre de symboles, dont certains sont définis dans le tableau 2.4, où C et D sont des concepts et R est une relation.

Symbole	Exemple	Description
\top	\top	Concept universel
\perp	\perp	Concept vide
\sqcup	$C \sqcup D$	Union de concepts
\sqcap	$C \sqcap D$	Intersection de concepts
\neg	$\neg C$	Négation de concept
\forall	$\forall R.C$	Quantificateur universel
\exists	$\exists R.C$	Quantificateur existentiel
\sqsubseteq	$C \sqsubseteq D$	Inclusion de concept

TABLE 2.4 – Liste non exhaustive de symboles utilisés par les logiques de description.

À partir des définitions précédentes, Randell *et al.* (1992) formalisent pour la première fois RCC-8 sous forme d'une logique triée, et Kutz *et al.* (2004), l'adaptent en logique de description, qui est un formalisme similaire. Les règles sont données dans le tableau 2.5.

L'avantage d'un tel formalisme est qu'il permet d'exprimer des contraintes quantitatives. Nous pouvons par exemple imaginer une relation *très éloigné* (TE), telle que :

$$TE(X, Y) : \top = DC(X, Y) \sqcap R_{>10}(X, Y)$$

où $R_{>10}(X, Y)$ est la relation " X et Y sont situés à une distance de plus de 10 kilomètres".

$$\begin{aligned}
DC(X, Y) &: \top = \neg\exists(X \sqcap Y) \\
EQ(X, Y) &: \top = (X \sqsubseteq Y) \sqcap (Y \sqsubseteq X) \\
EC(X, Y) &: \top = \exists(X \sqcap Y) \sqcap \neg\exists(IX \sqcap IY) \\
PO(X, Y) &: \top = \exists(IX \sqcap IY) \sqcap \exists(IX \sqcap \neg Y) \sqcap \exists(IY \sqcap \neg X) \\
TPP(X, Y) &: \top = (\neg X \sqcup Y) \sqcap \exists(X \sqcap \neg IY) \sqcap \exists(\neg X \sqcap Y) \\
NTPP(X, Y) &: \top = \forall(\neg X \sqcup IY) \sqcap \exists(\neg X \sqcap Y)
\end{aligned}$$

TABLE 2.5 – Six relations de base de RCC-8 encodées en logique de description. I représente ici aussi la relation *intérieur*. Les deux relations manquantes sont données par $TPP^{-1}(X, Y) = TPP(Y, X)$ et $NTPP^{-1}(X, Y) = NTPP(Y, X)$.

Réseau de contraintes. Dans la section 2.2.3, les différentes logiques formalisaient un ensemble de contraintes. Ces contraintes sont exprimées de manière similaire par Papadias et Egenhofer (1997). Par exemple $RN(A, B)$ signifie que le point A est strictement au nord du point B . Ces contraintes sont illustrées dans la figure 2.2. Les seize contraintes de base sont ainsi formalisées par un ensemble de contraintes. L’apport des auteurs dans le domaine est la formalisation des relations entre les objets en un graphe dans lequel les sommets sont les points et les arcs sont étiquetés par la relation qui lie les deux sommets aux extrémités. Un tel graphe, appelé *réseau de contraintes* est illustré dans la figure 2.3(a), où P_1, P_2, P_3, P_4 et P_5 sont des points liés par les relations $NW(P_2, P_1), NW(P_3, P_2)$ et $RW(P_4, P_3)$.

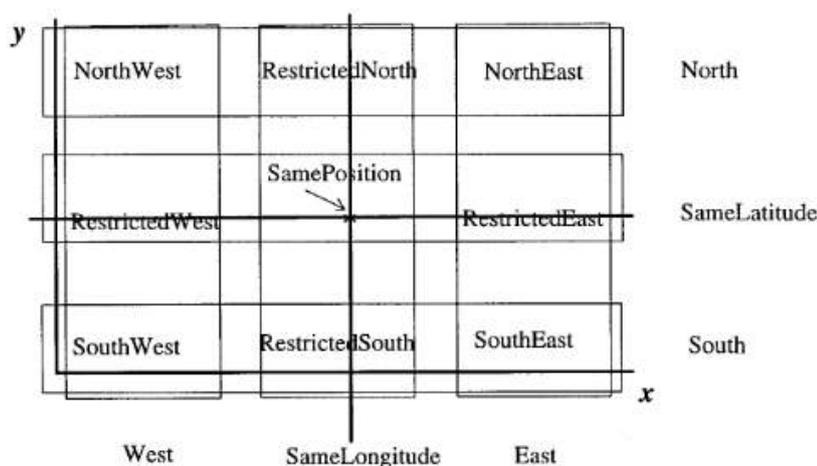


FIGURE 2.2 – Huit contraintes sur la relation directe entre deux points. (Papadias et Egenhofer, 1997)

À l’aide de l’algorithme mis au point par les auteurs, nous pouvons par exemple inférer la relation qui lie P_2 à P_4 . On ajoute ainsi une contrainte au réseau. Ceci est illustré dans la figure 2.3(b).

Algèbre relationnel. Moratz *et al.* (2000) définissent un certain nombre de relations entre deux dipôles, ou plus précisément entre un dipôle et un couple de points. Pour pouvoir raisonner sur ces règles et inférer des nouvelles connaissances, les auteurs formalisent ces règles par un

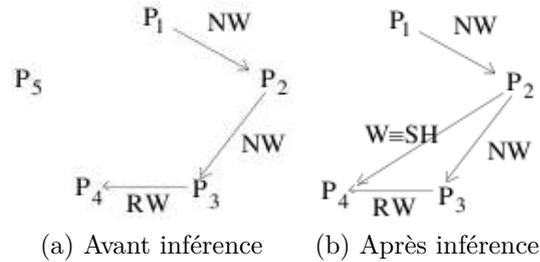


FIGURE 2.3 – Exemple d’un réseau de trois contraintes et cinq points. L’exemple montre l’inférence de la position relative de P_2 et P_4 . (Papadias et Egenhofer, 1997)

algèbre relationnel. Les relations spatiales sont représentées sous forme vectorielle. Les auteurs utilisent un système d’équation pour inférer les positions relatives de manière calculatoire.

Rough set. Les *rough sets* ont été décrit pour la première fois par Pawlak (1982). Un *rough sets* est une approximation formelle d’un ensemble mathématique classique. Il est composé de deux ensembles correspondant à une approximation inférieure et à une approximation supérieure de l’ensemble que l’on cherche à décrire. Ce modèle repose sur deux approximations. L’*approximation inférieure* est l’ensemble des objets qui sont classés comme appartenant à un ensemble de manière certaine. L’*approximation supérieure* est l’ensemble des objets qui appartiennent potentiellement à un ensemble, sans certitude. La *région limite* est l’ensemble contenant les objets dont on ne sait pas s’ils font partie ou non de l’ensemble. Dans l’espace, des objets peuvent être représentés à différents niveaux de précision, en fonction de leur résolution. Worboys (1998) utilise des *rough sets* pour raisonner sur l’imprécision d’objets dans l’espace.

3 Positionnement

Nous avons vu dans ce chapitre qu’il existe des méthodes de fouilles de données guidées par la connaissance. Dans la première section de ce chapitre, nous avons présenté les deux grandes familles de méthodes de fouille de données et celles ci peuvent être utilisées pour analyser des images de télédétection. Nous avons aussi vu que certaines de ces méthodes peuvent être guidées par la connaissance. Dans la seconde section, nous avons parlé plus en détail de la connaissance de l’expert, plus particulièrement de la connaissance spatiale et des formalismes associés.

Dans ce chapitre, nous avons abordé divers formalismes pour modéliser les relations spatiales. L’analyse d’image de télédétection étant intrinsèquement imprécise, il faut que le formalisme choisi prenne en compte cette imprécision. Pour cette raison, nous choisissons d’utiliser la logique floue pour modéliser les connaissances. En effet, les formalismes présentés dans ce chapitre ne sont pas capables de gérer l’imprécision, car dérivés de la logique classique (Renz et Nebel, 1999; Kutz et al., 2004; Randell et al., 1992). La logique floue ressemble aux *rough sets* dans le sens où tous deux sont dérivés de la logique classique pour gérer l’imprécision. Le choix de la logique floue a été motivé par sa puissance et le fait qu’elle soit utilisée dans beaucoup de domaines d’application (de la régulation de systèmes complexes à des problèmes de classification). De plus elle est expressive dans le langage naturel, ce qui permet un travail simplifié avec l’expert. Nous détaillons le principe de fonctionnement de la logique floue dans le chapitre 3. Nous avons présenté la méthode RCC-8. Elle permet de modéliser toutes les relations possibles entre deux

objets quelque soit le niveau. Cependant, ces relations sont uniquement qualitatives et il n'est donc pas possible de les quantifier. Ainsi l'utilisation de RCC-8 pour raisonner entraîne une perte d'information considérable (distance entre objets distincts, ratio de superposition, ratio d'adjacence). Pour cette raison nous choisissons de ne pas utiliser RCC-8 mais de redéfinir un ensemble de relations quantifiées élémentaires, celles ci sont présentées dans le chapitre 4 section 1.2.

Le fait que nous nous intéressions aux relations spatiales permet d'orienter nos travaux. Une relation spatiale peut se modéliser par un lien entre deux objets, un objet pouvant être lié à plusieurs autres objets. La structure de données qui s'impose naturellement pour modéliser nos données est donc le graphe. Par la suite, nous modélisons nos résultats de segmentation dans un graphe, dans lequel un sommet représente un segment et une arête représente une relation spatiale. Ceci est détaillé dans le chapitre 4.

Deuxième partie

Analyse d'images de télédétection guidée par la connaissance

★ Résumé ★

Dans cette partie, nous présentons notre méthode d'analyse d'images de télédétection guidée par la connaissance du domaine. Notre méthode repose sur le prétraitement de l'image par un algorithme de segmentation et sur une analyse orientée objet de cette segmentation. Pour palier l'imprécision inhérente à l'analyse d'images, nous choisissons de travailler avec la logique floue, que nous détaillerons dans le premier chapitre de cette partie. L'utilisation de la connaissance experte nécessite de formaliser rigoureusement la connaissance ainsi que les données à analyser de manière à pouvoir classer les données de manière optimale. Notre représentation et notre méthode de raisonnement utilisent la logique floue pour pouvoir classer les objets d'intérêt. Dans un second temps, nous utiliserons le résultat de cette classification pour extraire des objets composés de plus haut niveau sémantique. La méthode d'analyse d'images est présentée dans le second chapitre de cette partie.

3

La logique floue comme moteur de raisonnement

Sommaire

1	Logique floue	54
1.1	Base théorique : les ensembles flous	54
1.2	Fonction d'appartenance	54
1.3	Opérations sur les ensembles	55
2	La logique floue comme moteur de raisonnement	55
2.1	Première étape : le floutage	56
2.2	Deuxième étape : le raisonnement	57
2.3	Troisième étape : le défloutage	58
3	Exemple	58
4	État de l'art	61
4.1	Réseaux de neurones	62
4.2	Algorithmes évolutionnaires	63
4.3	Méthodes hybrides	64
4.4	Règles pondérées	64
5	Application à la télédétection	65
	Conclusion et contributions	66

L'objectif premier de nos travaux est de proposer une méthode de fouille de données guidée par la connaissance du domaine, appliquée à l'analyse d'images de télédétection. Pour pouvoir mettre en place de tels outils, il est nécessaire de définir et formaliser la représentation de la connaissance que nous allons utiliser. L'analyse d'images est intrinsèquement imprécise, la représentation sous forme de pixels étant une discrétisation d'une longueur d'onde continue. Il convient de bien choisir le formalisme de représentation de la connaissance pour pouvoir gérer cette imprécision. Pour ces raisons, nous choisissons d'utiliser la logique floue. La logique floue (Zadeh, 1975) repose sur la théorie des ensembles flous, introduite par Zadeh (1965). C'est une théorie mathématique qui étend la théorie ensembliste classique en permettant la gestion de l'imprécision et de l'erreur.

Dans ce chapitre, nous rappelons les bases théoriques de la logique floue et nous définissons le formalisme que nous utiliserons dans la suite de ce manuscrit.

1 Logique floue

Dans cette section nous rappelons les bases de la théorie des ensembles flous et de la logique floue. Nous présentons les notions d'ensemble flou, de fonction d'appartenance, et les opérations associées.

La logique floue (Zadeh, 1975) est un domaine de recherche actif. Elle permet de modéliser la connaissance à l'aide de variables linguistiques. Elle est notamment utilisée depuis la fin des années 70 comme moteur de raisonnement dans des systèmes experts (*i.e.* système dont le but est de reproduire le raisonnement d'un expert). Mamdani et Assilian (1975) et Mamdani (1977) introduisent le raisonnement basé sur la logique floue qui est présenté dans ce chapitre.

Lee (1990a,b) propose une revue de différentes applications de contrôleurs flous ainsi que des différentes stratégies et mécanismes de raisonnement. Novák (2012) propose un état de l'art des différentes théories et de leurs évolutions. Dans ce chapitre, nous présenterons uniquement les éléments que nous utiliserons dans la suite de nos travaux.

1.1 Base théorique : les ensembles flous

Dans la théorie ensembliste classique, le degré d'appartenance d'un élément à un ensemble est donné par une valeur binaire : l'élément appartient ou non à l'ensemble. Le degré d'appartenance à un ensemble flou est décrit à l'aide d'une fonction d'appartenance retournant une valeur réelle dans l'intervalle $[0, 1]$, 1 signifiant que l'élément appartient totalement à l'ensemble, 0 signifiant que l'élément n'appartient pas du tout à l'ensemble. Toutes les valeurs intermédiaires permettent de nuancer le degré d'appartenance. Un ensemble flou est défini par deux ensembles, le noyau et le support.

Définition 1 : Noyau

Le noyau est l'ensemble des éléments appartenant totalement à l'ensemble (*i.e.* le degré d'appartenance est égal à 1).

Définition 2 : Support

Le support est l'ensemble des éléments appartenant au moins un peu à l'ensemble (*i.e.* le degré d'appartenance est strictement positif).

Par conséquent, un élément peut appartenir à deux ensembles flous en même temps, à des degrés identiques ou différents (par exemple, un élément peut être *petit* avec un degré de 0.86 et *moyen* avec un degré de 0.27).

Définition formelle. Soit E l'espace des données, $A \subset E$ un ensemble flou, n_A le noyau et s_A le support de l'ensemble A , x un élément de E , μ_A la fonction d'appartenance modélisant l'ensemble flou A . Le noyau et le support sont définis dans les équations 3.1 et 3.2.

$$n_A = \{x \in E | \mu_A(x) = 1\} \quad (3.1)$$

$$s_A = \{x \in E | \mu_A(x) > 0\} \quad (3.2)$$

1.2 Fonction d'appartenance

Soit E l'espace des données et $A \subset E$ un ensemble flou. Une fonction d'appartenance est une application de E dans $[0, 1]$. Elle donne le degré d'appartenance d'un élément à l'ensemble A . Plusieurs fonctions d'appartenance peuvent décrire un ensemble flou (même noyau et un même support). N'importe quelle fonction mathématique de E dans $[0, 1]$ peut représenter un ensemble

fou. Nous présentons ici les quatre fonctions que nous avons retenues. Soit m et σ respectivement la moyenne et l'écart-type de A .

La fonction gaussienne est décrite par l'équation 3.3 et est représentée par la figure 3.1(a).

Cette fonction a les propriétés suivantes : $n_A = [m]$ et $s_A =] - \infty, +\infty[$. La fonction gaussienne est centrée sur la moyenne et décroît rapidement lorsqu'on s'éloigne de la moyenne. Cette fonction n'est jamais nulle, mais tend vers 0 dès qu'on s'éloigne de 5 fois l'écart-type.

$$f(x) = \exp\left(-\frac{(x - m)^2}{2\sigma^2}\right) \quad (3.3)$$

La fonction triangulaire est représentée par la figure 3.1(b). Cette fonction a les propriétés suivantes : $n_A = [m]$ et $s_A = [m - \sigma, m + \sigma]$. Comme la fonction gaussienne, la fonction triangulaire est centrée sur la moyenne. Cependant elle décroît linéairement et très rapidement lorsqu'on s'éloigne de la moyenne. Cette fonction est nulle lorsqu'on s'éloigne de 1 fois l'écart-type.

La fonction trapézoïdale est représentée par la figure 3.1(c). Cette fonction a les propriétés suivantes : $n_A = [m - 2\sigma, m + 2\sigma]$ et $s_A = [m - 3\sigma, m + 3\sigma]$. La fonction trapézoïdale décroît linéairement entre les bornes du noyau et du support.

La fonction booléenne est représentée par la figure 3.1(d). Cette fonction a les propriétés suivantes : $n_A = s_A = [m - 2\sigma, m + 2\sigma]$. La fonction booléenne est équivalente à un ensemble classique dans le sens où elle n'a que deux valeurs possible : 0 et 1.

Un ensemble flou étant étroitement lié à la fonction d'appartenance le représentant, nous utiliserons par la suite indifféremment les termes *ensemble flou* et *fonction d'appartenance*.

1.3 Opérations sur les ensembles

Les opérateurs classiques d'union, d'intersection et de complément sont étendus aux ensembles flous. L'intersection est modélisée par une T-norme et l'union par une T-conorme. Dans son article original, Zadeh (1965) propose comme T-norme l'opérateur minimum et comme T-conorme l'opérateur maximum. Soit E le domaine, $A \subset E$ et $B \subset E$ deux ensembles flous. Les équations 3.4, 3.5 et 3.6 illustrent ces opérations.

$$\mu_{A \cap B} = \min(\mu_A, \mu_B) \quad (3.4)$$

$$\mu_{A \cup B} = \max(\mu_A, \mu_B) \quad (3.5)$$

$$\mu_{\bar{A}} = 1 - \mu_A \quad (3.6)$$

Comme pour les opérations ensemblistes classiques, les opérations d'union et d'intersection sont commutatives et associatives, l'intersection est distributive par rapport à l'union. Cependant, le principe du tiers exclu est contredit ($\mu_A \cup \mu_{\bar{A}} \neq 1$). D'autre part, un élément peut appartenir à un ensemble et à son complément (mais à des degrés différents).

2 La logique floue comme moteur de raisonnement

La logique floue est une technique de raisonnement en intelligence artificielle. Elle repose sur la théorie des ensembles flous présentée dans la section précédente. Le principe de la logique floue est basé sur l'utilisation de variables linguistiques pour modéliser la connaissance. Nous pouvons voir la connaissance de l'expert comme des contraintes sur un ensemble de valeurs que peut prendre un objet d'intérêt. Plutôt que de définir une contrainte par un intervalle de valeurs,

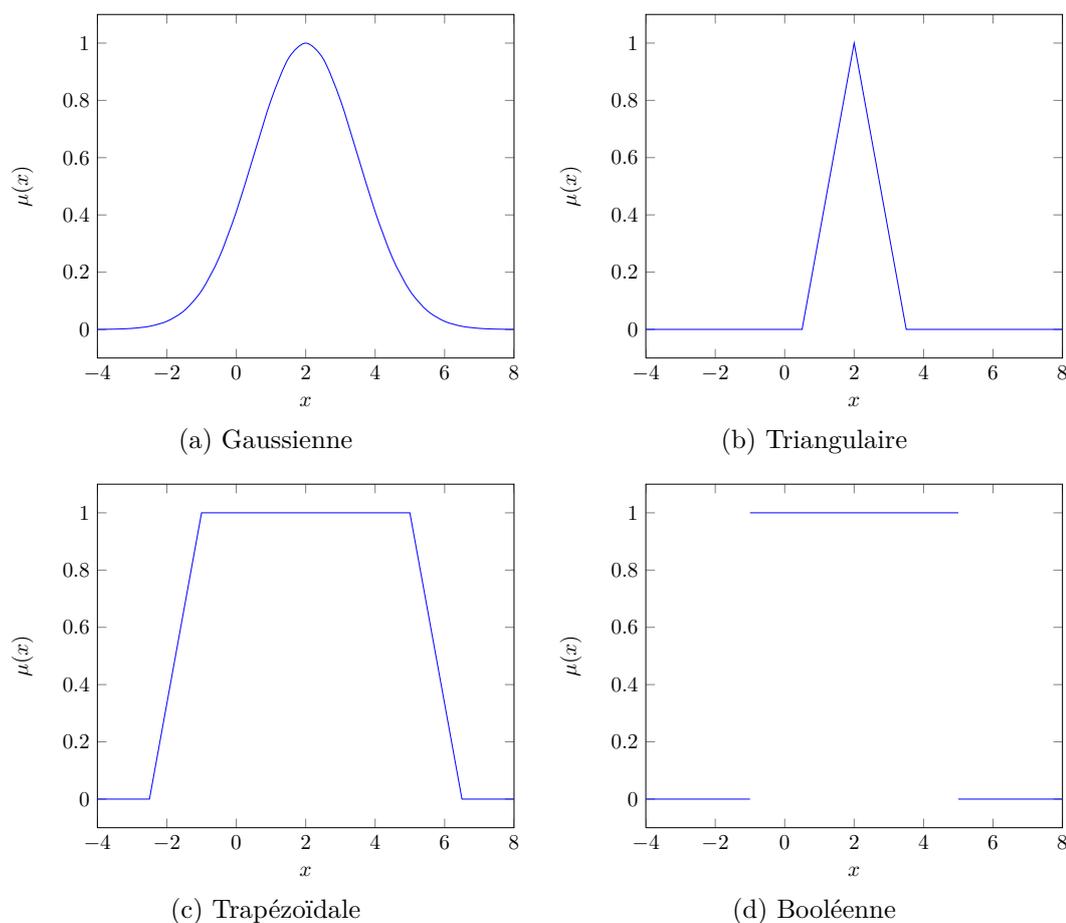


FIGURE 3.1 – Courbes associées aux fonctions gaussienne, triangulaire, trapézoïdale et booléenne. Dans ces exemples la moyenne est $m = 2$ et l'écart-type est $\sigma = 1.5$.

elle est définie par une fonction d'appartenance. Ainsi, une variable telle que la *taille* va pouvoir prendre comme valeur *petit* ou *grand*. Un expert peut alors associer des ensembles flous à des variables linguistiques du domaine qu'il sera possible d'utiliser dans un second temps pour former des règles. Le fait que ces variables linguistiques soient basées sur des ensembles flous permet de gérer l'imprécision des mesures à travers l'utilisation du degré d'appartenance.

Le raisonnement à l'aide de la logique floue fonctionne en trois étapes. Premièrement les variables numériques en entrée sont floutées. Deuxièmement le moteur d'inférence utilise les variables linguistiques pour raisonner. Troisièmement la sortie du moteur d'inférence est défloutée, ce qui permet à l'utilisateur de prendre une décision. Ces trois étapes utilisent la base de connaissances. Ce processus est illustré dans la figure 3.2.

2.1 Première étape : le floutage

Nous voulons évaluer un système à l'aide d'un moteur d'inférence flou. Le système est décrit par des valeurs numériques. Pour pouvoir raisonner dans le langage naturel, il est nécessaire

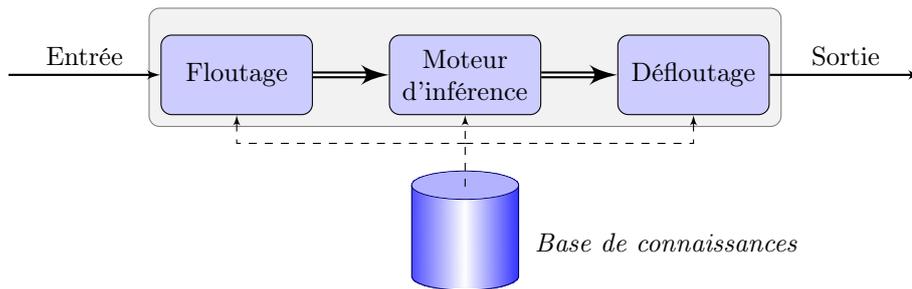


FIGURE 3.2 – Schéma du processus de raisonnement basé sur la logique floue. L'entrée correspond aux variables à évaluer, la sortie correspond à la décision à prendre.

de faire le lien entre les valeurs numériques et le langage naturel. Les fonctions d'appartenance décrites précédemment nous permettent de faire ce lien. Nous pouvons définir une variable linguistique sur un domaine de référence à l'aide de fonctions d'appartenance.

Définition 3 : Variable linguistique

Une variable linguistique est un triplet (V, X, T_V) telle que :

- X est le domaine de référence
- V est une variable définie sur X
- T_V est un ensemble fini de sous-ensembles flous

Par exemple, nous pouvons définir la variable linguistique *age* de la manière suivante ($age, [0, 150], \{nourisson, enfant, adolescent, adulte\}$), où *nourisson*, *enfant*, *adolescent* et *adulte* sont des fonctions d'appartenance.

Une fois le système flouté, nous ne disposons plus de variables numériques (par exemple $age = 21$) mais de variables floues (par exemple $age = adulte$ avec un degré d'appartenance de 1.0). Ces variables floues sont ensuite utilisées par le moteur d'inférence.

2.2 Deuxième étape : le raisonnement

En combinant différentes variables linguistiques, un expert peut former des règles cohérentes et complexes. De par leurs définitions linguistiques, ces règles sont compréhensibles par un être humain. Leurs traductions en termes logiques les rendent compréhensibles par un algorithme et donc utilisables dans un processus de raisonnement automatique.

L'objectif de cette étape est d'évaluer les différentes règles et de combiner chaque résultat en un résultat unique.

Définition 4 : Règle

Une règle est une combinaison de fonctions d'appartenance à l'aide des opérateurs logiques **et**, **ou** et **non**.

L'opérateur **et** est modélisé par une T-norme, l'opérateur **ou** est modélisé par une T-conorme, l'opérateur **non** est modélisé par le complément, ceci de manière similaire aux opérateurs de la théorie des ensembles flous. Parmi les T-normes les plus utilisées dans la littérature, nous pouvons citer l'intersection ensembliste de Zadeh (équation 3.4). Parmi les T-conormes les plus utilisées dans la littérature, nous pouvons citer l'union ensembliste de Zadeh (équation 3.5).

Il existe d'autres opérateurs dans la littérature (Lee, 1990a,b) tels que l'union et l'intersection probabiliste.

Soit *allongé*, *brillant*, *petit* et *moyen* quatre fonctions d'appartenance. Un exemple de règle combinant ces fonctions à l'aide des différents opérateurs est donné par la formule suivante :

$$r = \textit{allongé et non brillant et (petit ou moyen)}$$

La transformation de cette règle en expression mathématique utilisant les opérateurs ensemblistes de Zadeh est donnée dans l'équation 3.7.

$$\mu_r = \min(\mu_{\textit{allongé}}, \min(1 - \mu_{\textit{brillant}}, \max(\mu_{\textit{petit}}, \mu_{\textit{moyen}}))) \quad (3.7)$$

À chaque règle est associé un ensemble flou en sortie. Les règles sont évaluées et les ensembles flous résultants sont agrégés à l'aide de l'opérateur maximum (voir équation 3.5). Le moteur d'inférence repose sur l'évaluation des règles et l'agrégation des résultats. La paramétrisation du moteur d'inférence (le choix des opérateurs flous) est le point clé du système de raisonnement.

2.3 Troisième étape : le défloutage

La dernière étape du processus consiste à attribuer au résultat la bonne variable linguistique (processus de defuzzification, ou défloutage). Pour cela, il existe un grand nombre de méthodes dans la littérature (Lee, 1990a,b). Les deux plus utilisées sont le centre de gravité (*CdG*, équation 3.8) et la moyenne des maxima (*MM*, équation 3.9, où $S = \{y_x \in E, \mu(y_x) = \max_{y \in E}(\mu(y))\}$, E étant le domaine de définition de la variable de sortie). Soit d la variable de décision retournée par l'algorithme de raisonnement.

Lorsque la logique floue est utilisée pour évaluer un système expert la variable de sortie est ordonnée. Dans le cas d'une application à un problème de classification, l'opérateur de défloutage doit être adapté car la variable de sortie n'est pas ordonnée, une classe n'étant pas *supérieure* ou *inférieure* à une autre. La solution la plus simple est de prendre une variante de l'opérateur moyenne des maxima. Plutôt que de calculer la moyenne des maxima, nous gardons simplement la variable correspondant au score maximum. La formule utilisée est donnée par l'équation 3.10, où o est l'objet à étiqueter, x_j est la variable correspondant à la règle μ_j et n est le nombre de concepts évalués.

$$d_{CdG} = \frac{\int y \cdot \mu(u) \cdot dy}{\int \mu(u) \cdot dy} \quad (3.8)$$

$$d_{MM} = \frac{\int_S y \cdot dy}{\int_S dy} \quad (3.9)$$

$$d_{\max} = \{x_j | \mu_j = \max(\mu_i(o)), i \in [1, n]\} \quad (3.10)$$

Le processus de défloutage va donner la valeur de sortie de l'algorithme de raisonnement à partir de l'agrégat d'ensembles flous obtenu lors de l'évaluation. Cette valeur peut maintenant être utilisée dans un processus de décision.

3 Exemple

Illustrons le principe de fonctionnement d'un algorithme de raisonnement basé sur la logique floue. On veut réguler la vitesse d'une voiture sur une route nationale. Ce système repose sur deux

propriétés : la vitesse actuelle de la voiture et sa distance à la voiture devant elle. On veut décider si on doit conserver la vitesse actuelle, accélérer ou décélérer. La vitesse réglementaire sur une route nationale est de 90km/h, la distance de sécurité est de 50m. À partir de ces informations, nous pouvons décrire le système par trois variables linguistiques qui sont définies comme suit :

$$(vitesse, [0, 130], \{lente, bonne, rapide\})$$

$$(distance, [0, 100], \{proche, éloigné\})$$

$$(comportement, [-90, 90], \{ralentir, conserver, accélérer\})$$

Les fonctions d'appartenance correspondant à chacune de ces variables linguistiques sont représentées dans la figure 3.3. Nous utilisons des fonctions triangulaires et trapézoïdales dans les ensembles de définition de ces propriétés.

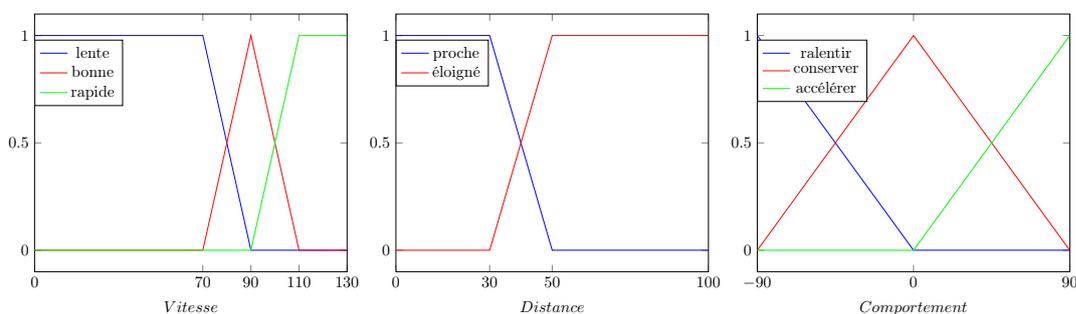


FIGURE 3.3 – Fonctions d'appartenances correspondant aux variables linguistiques *vitesse*, *distance* et *comportement*.

Pour contrôler le système, nous définissons trois règles :

$$r_{conserver} = \text{bonne et éloigné} \quad (3.11)$$

$$r_{ralentir} = \text{rapide ou proche} \quad (3.12)$$

$$r_{accélérer} = \text{lent et éloigné} \quad (3.13)$$

Entrées : Considérons que la voiture que nous voulons réguler roule à 105 kilomètres par heure et suit une autre voiture à une distance de 45 mètres. Nous représentons cette voiture par un objet $o = \{vitesse = 105, distance = 45\}$.

Floutage : Décrivons l'objet o à l'aide des variables linguistiques *vitesse* et *distance*. Le tableau 3.1 illustre l'opération de floutage.

Inférence : Nous choisissons les opérateurs ensemblistes de Zadeh (fonctions minimum et maximum) pour représenter nos opérateurs flous. L'évaluation de chacune des règles est résumée par les équations 3.14, 3.15 et 3.16. L'opérateur d'agrégation est la fonction maximum. L'évaluation détaillée du moteur d'inférence est représentée dans la figure 3.4.

Propriété	Variable	Fonction	Degré d'appartenance
vitesse	lente	$\mu_{lente}(105)$	0
	bonne	$\mu_{bonne}(105)$	0.25
	rapide	$\mu_{rapide}(105)$	0.75
distance	proche	$\mu_{proche}(45)$	0.25
	éloigné	$\mu_{éloigné}(45)$	0.75

TABLE 3.1 – Transformation des variables numériques en variables floues.

$$\mu_{conserver}(o) = \min(\mu_{bonne}(o), \mu_{éloigné}(o)) = \min(0.25, 0.75) = 0.25 \quad (3.14)$$

$$\mu_{ralentir}(o) = \max(\mu_{rapide}(o), \mu_{proche}(o)) = \max(0.75, 0.25) = 0.75 \quad (3.15)$$

$$\mu_{accélérer}(o) = \min(\mu_{lent}(o), \mu_{éloigné}(o)) = \min(0, 0.75) = 0 \quad (3.16)$$

Défloutage : À partir de la courbe calculée dans l'étape d'inférence, nous pouvons effectuer le défloutage. Les résultats pour les opérateurs *centre de gravité*, *moyenne des maxima* et *maximum* sont présentés dans le tableau 3.2 et sont illustrés dans la figure 3.5. Nous pouvons voir que le choix de l'opérateur de défloutage influe significativement le résultat.

Méthode de défloutage	Résultat
Centre de gravité	-25.3125
Moyenne des maxima	-78.75
Maximum	<i>ralentir</i>

TABLE 3.2 – Résultats du défloutage. Les résultats sont donnés pour les trois opérateurs de défloutage présentés dans ce chapitre. Le résultat correspond au comportement à adopter (modification de la vitesse de la voiture).

Sortie : À partir de la valeur de sortie de l'étape de défloutage, le système peut prendre une décision. Si l'opérateur *centre de gravité* est choisi, le système va baisser la vitesse de la voiture de 25.3125 kilomètres par heure. Si l'opérateur *moyenne des maxima* est choisi, le système va baisser la vitesse de la voiture de 78.75 kilomètres par heure. Ceci montre que le choix de l'opérateur de défloutage est une étape importante car cet opérateur a une forte influence sur la valeur retournée et donc sur la décision à prendre.

Le problème présenté ici est un problème décisionnel classique dont la variable de sortie était une valeur numérique (donc ordonnée). Les opérateurs *centre de gravité* et *moyenne des maxima* avaient tout leur sens ici. Cependant nous pouvons aussi voir ce problème comme un problème de classification. Dans ce cas il s'agit d'utiliser l'opérateur *maximum* pour trouver l'étiquette à retourner (*conserver*, *ralentir* ou *accélérer*), et c'est à l'utilisateur d'agir en conséquent.

Dans la suite de ce manuscrit, nous utiliserons la logique floue exclusivement pour des problèmes de classification, il ne sera plus pertinent d'utiliser les deux opérateurs *centre de gravité* et *moyenne des maxima*. Nous utiliserons donc uniquement l'opérateur *maximum*.

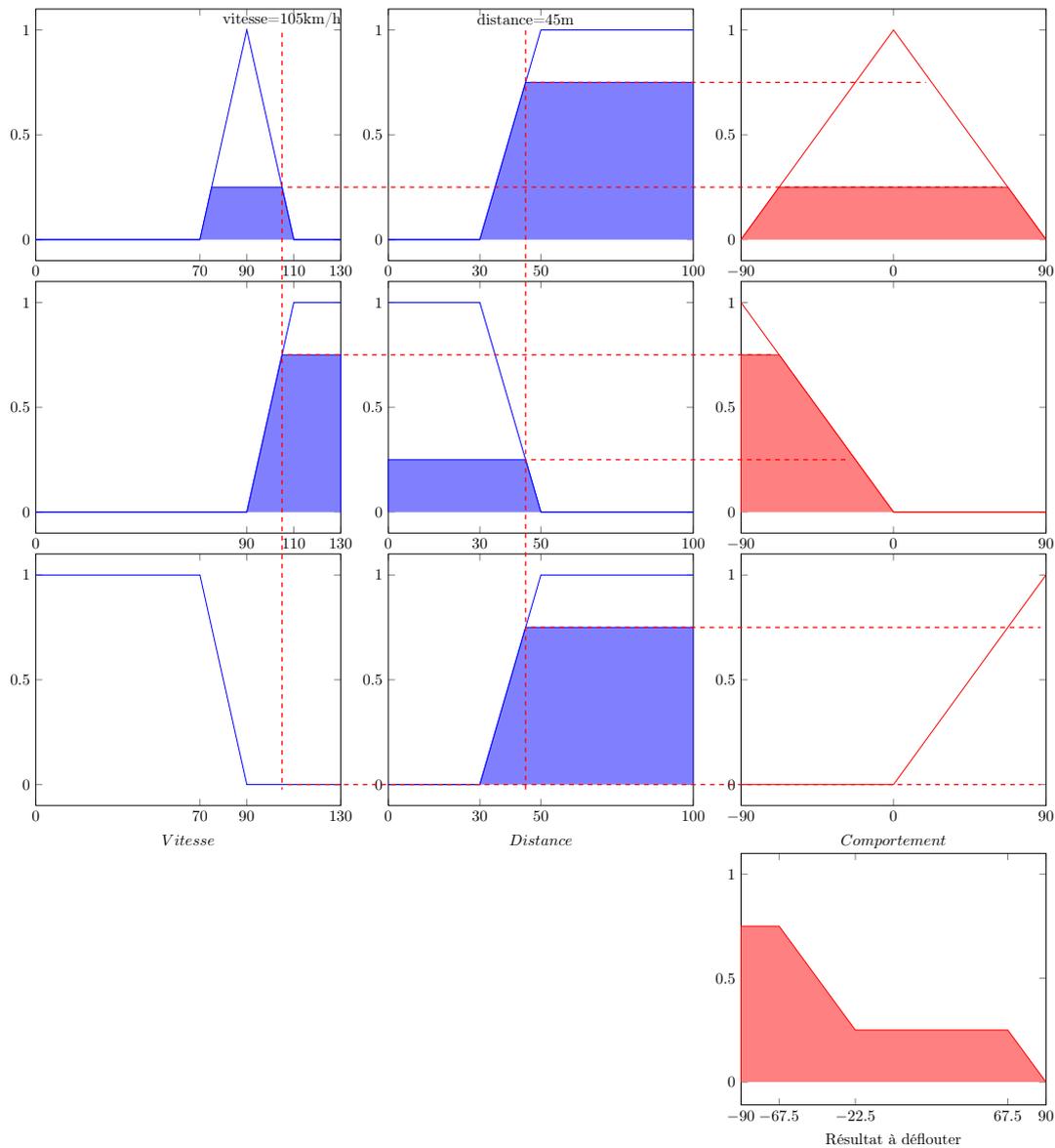


FIGURE 3.4 – Déroulement de l'algorithme de raisonnement. La première ligne correspond à la règle *conserver*, la seconde ligne correspond à la règle *ralentir*, la troisième ligne correspond à la règle *accélérer*. La quatrième ligne correspond à l'agrégation des résultats qu'il faudra déflouter.

4 État de l'art

La logique floue utilise la connaissance de l'expert dans un processus décisionnel. Lorsqu'il n'existe pas de connaissance *a priori*, il est nécessaire d'apprendre des fonctions d'appartenance et/ou des règles à partir d'exemples. De nombreuses méthodes d'apprentissage de règles existent dans la littérature. Parmi les familles de méthodes les plus populaires, nous pouvons noter les

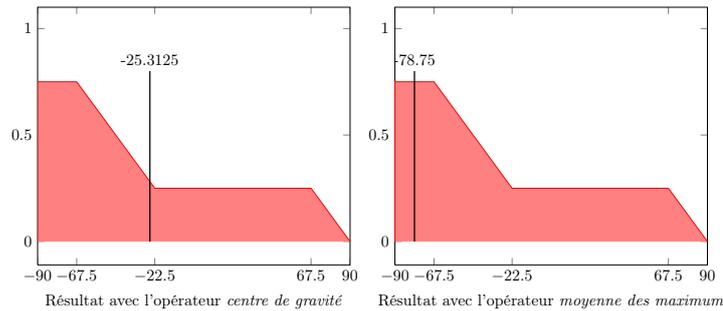


FIGURE 3.5 – Résultat comparatif des deux opérateurs de défloutage.

méthodes basées sur des réseaux de neurones (neuro-fuzzy) et les méthodes basées sur des algorithmes évolutionnaires.

4.1 Réseaux de neurones

Un réseau de neurones est un modèle de calcul dont le fonctionnement est inspiré du cerveau humain. Des neurones sont répartis sur plusieurs couches, les neurones de différentes couches étant connectés aux neurones des couches suivantes et précédentes. Un réseau possède une couche de neurones en entrée (pour modéliser les variables du problème), une couche de neurones en sortie (pour modéliser la ou les réponses) et une ou plusieurs couches cachées. L'apprentissage repose sur le principe de rétropropagation : un exemple est évalué par le réseau, l'erreur en sortie est calculée puis utilisée pour corriger la sortie du neurone en fonction de la sortie attendue. Cette erreur est ensuite propagée aux couches précédentes pour que tous les neurones soient corrigés. Un réseau de neurones peut être utilisé pour modéliser un moteur de raisonnement par logique floue. Les règles sont modélisées par des neurones et les fonctions d'appartenance sont modélisées par des neurones ou par des connexions.

Nauck et Kruse (1997) proposent une méthode appelée NEFCLASS (neuro-fuzzy classification) dérivée de leur précédente méthode NEFCON (neuro-fuzzy control). Un NEFCLASS est un réseau à trois couches. La couche interne modélise les règles, les connexions entre la couche d'entrée et la couche interne représentent les fonctions d'appartenance. Ces connexions peuvent être partagées entre plusieurs règles. L'utilisateur doit spécifier un nombre de fonctions d'appartenance sur les entrées, ainsi qu'un nombre maximum de règles à apprendre. Pour chaque entrée, le réseau évalue quelle règle classe le mieux cette entrée. Si une telle règle n'existe pas, elle est créée par le système. Sinon le système évalue l'ensemble des règles avant de corriger les connexions lors de la phase de rétropropagation. Les fonctions d'appartenance utilisées sont de type triangulaire. Lors de l'apprentissage, une fonction est modifiée en déplaçant ses paramètres (centre, support). Dans un second temps, les auteurs proposent d'utiliser des connaissances *a priori* (base de règles pré-existante) pour initialiser le réseau.

Cpalka (2009) utilise l'algorithme fuzzy c-means pour générer les fonctions d'appartenances initiales. Un réseau est ensuite construit à partir de ces fonctions. Les fonctions sont des gaussiennes. Après apprentissage, le système est capable de réduire le nombre de règles et de fusionner les entrées et sorties similaires pour réduire la complexité du réseau sans pour autant perdre en précision.

Dans les méthodes suivantes, les fonctions d'appartenance ne sont plus modélisées par des connexions mais par des neurones dans la première couche cachée.

Wang et Lee (2002) proposent une méthode appelée SANFIS (self adaptative neuro-fuzzy inference system). Trois architectures sont proposées, selon le modèle suivant. La première couche cachée représente les fonctions d'appartenance pour chaque entrée. La seconde couche cachée représente les règles. Chaque neurone de cette couche est connecté à un seul neurone de la couche précédente pour une entrée donnée. La troisième couche cachée effectue une normalisation des sorties de la couche précédente. Les neurone de ces deux couches sont entièrement connectés. Parmi les trois architectures proposées, deux utilisent une dernière couche cachée pour effectuer une opération de défloutage. Les fonctions d'appartenance utilisées par cette méthode sont des fonctions gaussiennes. L'apprentissage est divisé en deux étapes. Premièrement, la structure interne du réseau est apprise à l'aide de l'algorithme de clustering MCA (mapping-constrained agglomerative clustering). Ensuite, une méthode des moindres carrés linéaire/non linéaire réursive est utilisée pour accélérer la convergence et régler finement le réseau.

Chakraborty et Pal (2004) et Wu *et al.* (2010) proposent chacun une méthode à deux couches cachées, la première modélise les fonctions d'appartenance et la seconde les règles, avec des fonctions d'appartenances gaussiennes. Chakraborty et Pal (2004) génèrent dans un premier temps toutes les combinaisons possibles de règles. Dans un second temps, le système élague le réseau en éliminant les règles redondantes, incompatibles et peu utilisées pour ne conserver que les règles les plus pertinentes. Wu *et al.* (2010) voient leur réseau comme un système d'inférence de Takagi-Sugeno d'ordre zéro dans lequel les poids des connexions de la couche d'entrées à la première couche cachée comme les paramètres des fonctions d'appartenance correspondantes. Les poids des connexions entre les deux couches cachées sont constants et valent 1. Une fois initialisé, le système utilise un algorithme de descente de gradient pour optimiser le réseau.

Suresh et Subramanian (2011) proposent la méthode Mc-FIS (meta-cognitive fuzzy inference system). Leur réseau comporte trois couches cachées dont les neurones représentent respectivement les fonctions d'appartenance, les règles et des neurones de normalisation. La méthode proposée repose sur l'autorégulation du système. Si un exemple n'apporte pas de nouvelles informations, il est simplement supprimé de l'échantillon. Si un exemple apporte suffisamment d'informations, il est utilisé pour mettre à jour le réseau. Si aucun des cas précédents n'est satisfait, l'exemple est placé à la fin du jeu d'exemples pour être réévalué plus tard. Le système est ainsi capable de décider ce qu'il doit apprendre, comment il doit apprendre et quand il doit apprendre.

4.2 Algorithmes évolutionnaires

Les algorithmes évolutionnaires sont des algorithmes basés sur la notion d'une population évoluant au fil du temps pour converger vers une bonne solution lorsqu'il n'est pas possible de calculer la solution exacte en temps fini. Les algorithmes évolutionnaires répondent à des problèmes d'optimisation. Ils sont basés sur un modèle génétique et reposent sur la notion de sélection naturelle.

Castillo *et al.* (2001) proposent la méthode SLAVE (*structural learning algorithm in vague environment*). Cette méthode repose sur le principe suivant. Premièrement, un algorithme génétique est utilisé pour générer une règle unique. Cette règle est ensuite ajoutée à la base de règles et est pénalisée. Si la base de règles est satisfaisante l'algorithme s'arrête, sinon le processus recommence. Les individus de l'algorithme génétique sont représentés par deux parties. La première représente les variables en entrée, la seconde représente les fonctions d'appartenance. La fonction d'évaluation prend en compte la complétude et la consistance des règles. Des contraintes sont utilisées pour simplifier la base de règles.

Ishibuchi et Yamamoto (2004) proposent deux méthodes multi-objective MOGA et MOGLS. Dans un premier temps des fonctions d'appartenances triangulaires sont générées pour chaque attribut. Ces fonctions sont combinées en règles de petite taille. Les mesures de confiance et de support sont utilisées pour éliminer les règles les moins satisfaisantes. Les règles restantes sont ensuite réparties en différents groupes (une par classe recherchée) et triées pour ne garder que les meilleures pour chaque classe. Les règles restantes vont ensuite alimenter un algorithme génétique multi-objectif (MOGA) qui a pour but de maximiser la précision de la classification, de minimiser le nombre de règles et de minimiser la taille globale des règles. Ceci est fait à l'aide d'un critère de non dominance. Cet algorithme est ensuite étendu en un algorithme de recherche de minimum local multi-objectif (MOGLS).

Fernández *et al.* (2010) décomposent les problèmes de classification multi-classes en utilisant des approches par paires. Des classifieurs sont entraînés pour chaque paire de classes et les résultats sont ensuite combinés pour prendre une décision. Les auteurs introduisent une nouvelle méthode de comparaison basée sur un critère de non dominance et la compare à deux méthodes de comparaison par vote. Deux classifieurs sont comparés, L'algorithme SLAVE présenté précédemment et l'algorithme FH-GBML. Les expériences réalisées montrent que le critère de non dominance donne de meilleurs résultats.

4.3 Méthodes hybrides

Il existe des méthodes hybrides mêlant algorithmes évolutionnaires et réseaux de neurones. Russo (2000) propose un classifieur basé sur un algorithme génétique. Les individus représentent des règles. Pour éviter une convergence prématurée, les individus sont répartis en plusieurs sous-populations. Pour chaque sous-population, une étape de l'évolution se décompose comme suit. Un nouvel individu est créé par croisement et mutation. Si cet individu améliore la note de la sous-population, il est utilisé pour initialiser un réseau de neurones. Le réseau est entraîné puis est transformé en règle (individu) avant d'être réinjecté dans la population initiale. Cette méthode permet d'apprendre des règles plus simples et par conséquent de réduire le temps de calcul.

4.4 Règles pondérées

Les méthodes présentées précédemment reposent sur la modification des fonctions d'appartenance pour faire converger le système. Une autre option est d'associer un poids aux fonctions et aux règles et de faire varier ces poids plutôt que les fonctions elles-mêmes.

Nozaki *et al.* (1996) génèrent pour chaque attribut k fonctions d'appartenance triangulaires pondérées et les combinent en règles. L'espace des données est ainsi découpé en une grille. Le poids des règles va être modifié par correction de l'erreur calculée pour chaque règle. Par conséquent la grille est modifiée pour mieux correspondre aux classes recherchées. La base de règles est ensuite élaguée en supprimant les règles inutiles. La qualité du résultat est fortement dépendant du paramètre k .

Ishibuchi et Nakashima (2001) proposent une méthode similaire. Cependant Ishibuchi et Yamamoto (2005) montrent que leur méthode est limitée dans les cas d'apprentissage multi classes avec un grand nombre de classes. Pour remédier à ce problème Nakashima *et al.* (2007) proposent une nouvelle méthode de spécification du poids des règles, plus efficace dans les cas multi classes.

5 Application à la télédétection

Nous avons vu dans ce chapitre que la logique floue peut être utilisée comme algorithme de classification. Elle peut ainsi être utilisée dans l'analyse d'images de télédétection pour classer les pixels ou les segments.

Les premières méthodes proposaient des classifications sur les pixels. [Zhu et al. \(1996\)](#) combinent la logique floue avec un système d'information géographique et un système expert pour inférer la classe au sol à partir de conditions environnementales. Cette méthode génère pour chaque point un vecteur de similarité de sol montrant la similarité de ce point par rapport à des points connus. Cette méthode a comme avantages par rapport aux méthodes standards un coût de production plus faible et des résultats plus détaillés au niveau spatial. [Foody \(1996\)](#) compare un classifieur probabiliste, un classifieur flou, et un réseau de neurones. Il montre que les méthodes floues, de part leur gestion de l'imprécision, permettent une meilleure estimation de la classe d'un pixel. Les auteurs testent leur méthode sur des zones de végétation. [Binaghi et al. \(1997\)](#) proposent une méthode de classification basée sur la connaissance conçue pour intégrer l'information contextuelle dans une analyse multi sources. La combinaison des différentes sources se fait à l'aide de la logique floue. Les auteurs appliquent leur méthode à la surveillance de glaciers. [Laha et al. \(2006\)](#) utilisent une carte auto adaptative de Kohonen pour générer des prototypes qui sont traduits en règles floues. Une étape de classification est ensuite effectuée en prenant en compte le voisinage des pixels. La prise de décisions est effectuée à l'aide de la théorie des croyances de Dempster-Shafer. La méthode est testée sur deux images LANDSAT-TM de 512×512 pixels. Les images contiennent sept bandes. Les classes recherchées sont *bâtiment*, *route* et *végétation*. Les résultats obtenus montrent un taux d'erreur moindre qu'une approche basée sur les champs aléatoires de Markov.

Les méthodes plus récentes travaillent sur des résultats de segmentation. [Ke et al. \(2008\)](#) utilisent un classifieur flou sur une segmentation multi échelle à quatre niveaux pour extraire et surveiller la végétation urbaine à partir d'images QUICKBIRD. Les quatre niveaux de segmentation permettent d'extraire des objets à différentes échelles et de créer un réseau d'objets hiérarchiques. Les objets de ce réseau sont ensuite classés par la logique floue, en utilisant des informations spectrales, spatiales, de texture et contextuelles. Pour caractériser les bâtiments sans toit, [Gueguen et al. \(2012\)](#) utilisent le filtrage morphologique, plus particulièrement la transformation *top-hat*, pour produire une carte d'appartenance. Dans un deuxième temps les auteurs utilisent la logique floue pour agréger des structures de la carte d'appartenance. La méthode est testée sur une image panchromatique WORLDVIEW d'une résolution de 0.5 mètres. [Sebari et He \(2013\)](#) utilisent des images à très haute résolution spatiale IKONOS pour analyser des paysages urbains à l'aide de méthodes orientées objets. Les auteurs sélectionnent un ensemble de propriétés. Pour chacune de ces propriétés deux fonctions sont définies, *propriété satisfaite* et *propriété non satisfaite*, une fonction étant la négation de l'autre. Ces fonctions sont combinées en règle par un expert et servent à créer une base de connaissances. Le seuil définissant la satisfaction d'une propriété est appris à partir des données. De nouvelles règles sont apprises puis injectées dans la base de règles. Les auteurs montrent l'efficacité de leur méthode ainsi que la bonne capacité de transfert des règles apprises.

Nous pouvons voir que la logique floue est régulièrement utilisée pour gérer l'imprécision dans des problèmes d'analyse d'images de télédétection, et ce notamment ces dernières années pour traiter les images THRS ([Ke et al., 2008](#); [Gueguen et al., 2012](#); [Sebari et He, 2013](#)). Ceci nous conforte dans notre choix d'utiliser la logique floue comme méthode de classification dans notre méthode.

Conclusion et contribution

Dans ce chapitre, nous avons présenté les bases de la logique floue et de son utilisation dans un système expert. L'avantage de la logique floue par rapport à la logique classique est de permettre plus de souplesse lors de la définition d'un ensemble. Avec un degré d'appartenance plutôt qu'une valeur booléenne, nous pouvons gérer l'imprécision dans un système expert, ce qui est particulièrement utile dans notre cas puisque nous voulons analyser des images qui sont imprécises par nature.

Nous avons vu que la littérature dans le domaine est extrêmement riche, autant au niveau théorique qu'appliqué. De plus, la logique floue est utilisée dans le domaine de la télédétection pour palier les problèmes d'imprécision. Il est donc pertinent de l'utiliser pour l'analyse d'images de télédétection, tout particulièrement les images à très haute résolution spatiale.

Dans le chapitre suivant, nous utiliserons la logique floue pour formaliser la connaissance d'un expert géographe et nous utiliserons notre moteur d'inférence pour analyser des images de télédétection.

Réalisation logicielle : Une implémentation du moteur d'inférence présenté dans ce chapitre a été réalisée dans une bibliothèque logicielle écrite en langage C++. Cette bibliothèque est hautement paramétrable : la base de connaissance peut être écrite dans un fichier XML¹ sous la forme de balises et d'attributs, avec des règles écrites dans le langage naturel ; la forme des fonctions d'appartenance peut être choisie indépendamment pour chaque fonction ; les opérateurs peuvent être spécifiés à l'exécution indépendamment pour chaque règle. Un exemple de fichier XML est donné dans la figure 3.6.

```
<functions>
  <function name="adjVegetation "
    relation="adjacency" concept="vegetation" shape="trapezoidal "
    minSupport="0" maxSupport="1" minCore="0.25" maxCore="0.75" />
  <function name="petit "
    attribute="taille" shape="gaussian "
    mean="100." sigma="25." />
  <function name="compPavillonDense "
    composition="ratio" concept="pavillon" shape="trapezoidal "
    minSupport="0.25" maxSupport="1" minCore="0.5" maxCore="0.75" />
</functions>

<rules>
  <rule name="pavillon" definition="adjVegetation et petit" />
</rule>
```

FIGURE 3.6 – Exemple de fonctions et de règles décrit par un fichier XML.

1. <http://www.w3.org/XML/>

4

Utilisation de connaissances expertes pour l'analyse d'images de télédétection

Sommaire

1	Représentation des données	68
1.1	Graphe d'objets d'intérêt	68
1.2	Objet d'intérêt	68
2	Représentation de la base de connaissances	70
2.1	Concepts	70
2.2	Hiérarchie de concepts	71
2.3	Illustration de la base de connaissances	71
3	Classification des objets d'intérêt	71
4	Extraction d'objets composés	72
4.1	État de l'art	72
4.2	Proposition : Algorithme d'extraction des objets composés	74
5	Application	75
	Conclusion et contributions	75

L'objectif de nos travaux est l'intégration d'information contextuelle dans un processus d'analyse d'images de télédétection orienté objet, appliqué à l'analyse de paysages urbains. Pour rejoindre les problématiques du LIVE, nous nous intéressons à l'extraction de bâtiments, plus particulièrement de pavillons, et à l'extraction d'îlots de type résidentiels.

Nous rappelons que les images que nous utilisons sont des images panchromatiques et multispectrales à très haute résolution spatiale. Elles proviennent de deux capteurs : le capteur QUICK-BIRD (multispectral : 2.44m ; panchromatique : 0.61m) et le capteur PLEIADES (multispectral : 2m ; panchromatique : 0.5m). Les images sont fusionnées pour obtenir des images multispectrales à la résolution de la panchromatique. Les images fusionnées sont segmentées, ceci nous permet d'utiliser des méthodes *orientées objet*.

La méthode d'analyse d'images que nous proposons repose sur l'identification d'objets d'intérêt à partir de leur définition par un expert du domaine. Pour cela, il est nécessaire de formaliser la représentation des objets d'intérêt et la base de connaissances. Dans un premier temps, nous allons présenter la formalisation des objets d'intérêt que nous proposons avant de décrire la

notion de concept et l'organisation hiérarchique de ces concepts. Nous présenterons ensuite les algorithmes utilisés pour l'extraction des objets composés.

1 Représentation des données

Dans cette section nous présentons les données sur lesquelles nous travaillons et le formalisme utilisé pour les représenter. Les données brutes que nous souhaitons analyser sont des images de télédétection. Notre représentation des données est basée sur la notion d'*objets d'intérêt* et de leurs propriétés.

1.1 Graphe d'objets d'intérêt

Nous nous intéressons à l'analyse des relations spatiales entre objets d'intérêt dans une image segmentée. Pour pouvoir stocker et manipuler aisément nos données, nous choisissons de les représenter par un graphe d'adjacence. Chaque objet d'intérêt ainsi que l'ensemble des informations contextuelles sont contenues dans le graphe. Celui-ci contenant toute l'information utile, l'image n'est plus nécessaire. C'est ce graphe qui sera traité par les divers algorithmes mis en place par la suite.

Le graphe est construit à partir d'une segmentation. Chaque segment va générer un objet d'intérêt qui sera représenté par un sommet dans le graphe. Une arête est ajoutée dans le graphe lorsque deux objets sont adjacents (*i.e.* leurs segments respectifs sont adjacents dans l'image segmentée). Ceci est illustré dans la figure 4.1.

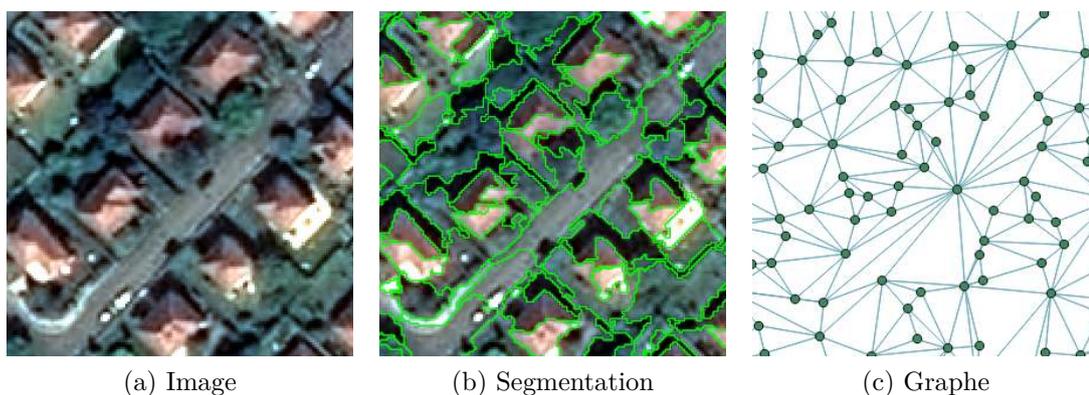


FIGURE 4.1 – Différentes étapes de création du graphe d'adjacence. L'image en entrée 4.1(a) est d'abord segmentée 4.1(b) puis transformée en graphe d'adjacence 4.1(c).

1.2 Objet d'intérêt

Un objet d'intérêt est un objet ayant une réalité sémantique dans le domaine d'expertise. Il peut être de deux types : un objet d'intérêt simple ou composé.

Définition 5 : Objet d'intérêt simple

Un objet d'intérêt simple représente la granularité la plus fine pour l'expert. Il est représenté par sa forme géométrique (le polygone englobant cet objet) et par des propriétés numériques.

Définition 6 : Objet d'intérêt composé

Un objet d'intérêt composé est un agrégat d'objets simples et/ou d'autres objets composés.

Dans le contexte de ces travaux, un objet simple peut être par exemple une maison, un arbre, une route ou encore une étendue d'eau ; un objet composé peut être par exemple un îlot pavillonnaire, un parc, une zone d'activité ou encore un aéroport. Ces objets sont géoréférencés. De ce fait, nous pouvons très facilement interagir avec un système d'information géographique.

Les propriétés numériques décrivant les objets d'intérêt sont de trois types : les attributs, les relations spatiales et les relations de composition.

Définition 7 : Attribut

Un attribut est une propriété qui ne dépend que de l'objet en lui-même.

Les attributs utilisés dans nos travaux sont les suivants : la valeur moyenne de chacune des quatre bandes, Normalized Difference Vegetation Index (NDVI : un indice de végétation fréquemment utilisé dans la littérature), Normalized Difference Water Index (NDWI2 : indice d'eau), Soil Brightness Index (SBI : indice de brillance des sols), Color Index (CI : indice de couleur qui met en avant les objets rouges tels que les champs non cultivés ou les toits de tuiles rouges), la taille, l'élongation, la longueur et la largeur, le périmètre, l'indice de Miller (indice de circularité), l'indice de Morton (indice de forme carrée).

Les indices radiométriques sont définis par les équations 4.1, 4.2, 4.3 et 4.4, où *rouge*, *bleu*, *vert* et *pir* (proche infrarouge) représentent les valeurs moyennes de chacune des quatre bandes spectrales de l'image. Les indices de Miller et Morton sont définis par les équations 4.5 et 4.6, où *a*, *L*, *p* représentent respectivement l'aire, la longueur et le périmètre d'un objet.

$$NDVI = \frac{pir - rouge}{pir + rouge} \quad (4.1)$$

$$NDWI2 = \frac{vert - pir}{pir + vert} \quad (4.2)$$

$$CI = \frac{rouge - vert}{rouge + vert} \quad (4.3)$$

$$SBI = 0.2623 \cdot bleu + 0.6432 \cdot vert + 0.6302 \cdot rouge + 0.3471 \cdot pir \quad (4.4)$$

$$I_{Miller} = \frac{4 \cdot \pi \cdot a}{p^2} \quad (4.5)$$

$$I_{Morton} = \frac{4 \cdot a}{\pi \cdot L^2} \quad (4.6)$$

Définition 8 : Relation spatiale

Une relation spatiale est une propriété qui lie un objet à son voisinage.

Parmi l'ensemble de relations spatiales présentées dans le chapitre 2, nous en retenons deux : la relation d'adjacence et la relation de distance. Ces deux relations sont celles utilisées dans le dictionnaire d'objets urbains proposé par le LIVE. La *relation d'adjacence* à un objet de type A est définie par le rapport entre les côtés du polygone courant partagés avec des objets de type A sur le périmètre de l'objet courant. La *relation de distance* à un objet de type B est définie par la distance en mètres à l'objet de type B le plus proche de l'objet courant.

Définition 9 : Relations de composition

Une relation de composition est une propriété qui lie un objet à un ou des objets de différents niveaux.

Nous définissons deux relations de composition : *compose* et *est composé de*. Ces deux relations sont antisymétrique : si o_i est composé de o_j alors o_j compose o_i . La relation de composition d'un objet composé par des objets de type C est définie par le rapport de la surface de chacun des objets de type C composant l'objet courant sur la taille de l'objet courant.

2 Représentation de la base de connaissances

Nous venons de décrire notre formalisation des données. Nous allons maintenant décrire notre formalisation de la connaissance experte. Elle repose sur la notion de *concepts* ainsi que sur l'organisation hiérarchique de ces concepts entre eux.

2.1 Concepts

À partir de la définition précédente d'un objet d'intérêt, nous pouvons définir la notion de concept et de concept composé.

Définition 10 : Concept Simple

Un concept est une représentation abstraite d'un objet d'intérêt. Il définit une classe et les contraintes que doivent satisfaire les instances de cette classe. Les contraintes sont modélisées par des règles floues qui portent sur les attributs et les relations spatiales uniquement.

Nous appelons *concept simple* un concept défini à un niveau d'interprétation donné (c'est à dire qui n'est pas défini en terme de composition). L'expert dispose d'un grand nombre de variables linguistiques dans son domaine d'expertise. Par exemple, la propriété *élongation* est définie par les variables linguistiques *compact*, *moyennement allongé* et *très allongé*. L'expert peut formuler des règles à partir des différentes variables linguistiques. Chaque règle décrit un concept. Par exemple, le concept *arbre* est défini par la règle *petit et compact et NDVI élevé*.

Dans un paysage urbain, l'expert peut définir des concepts de plus haut niveau. Un objet correspondant à un tel concept composé est un agrégat d'objets élémentaires ou d'autres objets composés.

Définition 11 : Concept composé

Un concept composé est défini par une règle portant sur tout type de propriétés (attributs, relations spatiales, relations de composition).

Par exemple, le concept composé *îlot pavillonnaire* est défini par la règle *grand et composé de pavillons et composé de végétation et entouré par de la route*.

La manière la plus simple de représenter un concept est de le représenter sous la forme d'une liste de contraintes sur les propriétés d'un objet. De là, émergent plusieurs questions. Comment formaliser ces contraintes pour qu'elles puissent prendre en compte l'imprécision inhérente au traitement d'images ? Quel formalisme utiliser pour que l'expert puisse produire des règles concises et réutilisables ? Afin de répondre à ces questions, nous avons choisi de représenter nos contraintes par des ensembles flous car ils permettent de gérer l'imprécision tout en étant faciles à mettre en place. Ceux-ci sont présentés dans le chapitre 3. L'expert définit des variables linguistiques sur les différentes propriétés. Les fonctions d'appartenance sur ces variables linguistiques forment les contraintes. La règle définissant un concept est par conséquent une règle floue.

Un exemple de concepts simples et de concepts composés est donné dans la figure 4.2. Nous pouvons y voir des concepts avec leurs contraintes sur les attributs et les relations spatiales.

2.2 Hiérarchie de concepts

L'expert définit l'ensemble des concepts pertinents pour l'étude. Il forme ainsi un dictionnaire de concepts qui pourra être utilisé par un algorithme de classification basé sur la connaissance. Parmi ces concepts, l'expert peut définir un concept qui est un cas particulier d'un autre concept. Par exemple, le concept *étang* est un cas particulier du concept *étendue d'eau*, ou, le concept *végétation* est une généralisation du concept *pelouse*. Pour pouvoir ordonner les concepts en conséquence, ils sont placés dans une hiérarchie dans laquelle une feuille est un concept particulier et chaque noeud est une généralisation de ses sous-arbres. Un exemple de hiérarchie est donné dans la figure 4.2. Dans cet exemple, nous pouvons voir des concepts simples et un concept composé ainsi que les relations de composition et les relations spatiales définissant le concept *îlot résidentiel*.

Notons Θ le dictionnaire des concepts, $\mathcal{C}_i \in \Theta$ le concept i . Définissons la relation de sous-concept. Cette relation d'ordre partielle, notée \preceq_{Θ} , est définie par l'équation 4.7 dans laquelle \mathcal{C}_i est un sous-concept de \mathcal{C}_j .

$$\forall \mathcal{C}_i, \mathcal{C}_j \in \Theta^2, \mathcal{C}_i \preceq_{\Theta} \mathcal{C}_j \quad (4.7)$$

On note \preceq_{Θ}^k la relation d'ordre k . Ainsi, $\mathcal{C}_i \preceq_{\Theta}^1 \mathcal{C}_j$ signifie que \mathcal{C}_i est un sous-concept descendant directement de \mathcal{C}_j .

Corollaire : \mathcal{C}_j est un super-concept de \mathcal{C}_i . C'est un concept plus générique.

Par exemple, *pavillon* \preceq_{Θ} *bâtiment*. *Maison individuelle* est un sous-concept de *Bâtiment*, et *Bâtiment* regroupe tous les bâtiments, dont font partie, entre autre, les maisons individuelles.

2.3 Illustration de la base de connaissances

Pour illustrer la base de connaissances, nous utilisons une notation proche d'un diagramme de classes UML¹. L'UML (de l'anglais Unified Modeling Language) est un langage de modélisation graphique utilisé en développement logiciel et en conception.

Un concept est représenté par une boîte bleue contenant son nom et les contraintes associées. Une flèche triangulaire représente la relation de sous-concept. Une flèche en diamant représente la relation de composition. Un lien en pointillé représente une relation spatiale.

L'exemple présenté dans la figure 4.2 montre les différents concepts intervenant dans la définition du concept composé *îlot résidentiel*. Le concept le plus générique est le concept *objet*, tous les concepts descendent de lui. Certains concepts définissent des contraintes sur les attributs (le concept *végétation* par exemple), des relations spatiales (par exemple la relation d'adjacence entre le concept *végétation* et le concept *pavillon*) ou des relations de composition (par exemple la relation de composition entre *îlot résidentiel* et *pavillon*).

Les concepts présents dans la hiérarchie sont issus d'un dictionnaire de concepts décrits et formalisés par les experts du laboratoire de géographie de l'université de Strasbourg (LIVE).

3 Classification des objets d'intérêt

Un concept est défini par une règle floue. Il est donc simple d'évaluer la correspondance d'un objet d'intérêt à un concept : il suffit d'évaluer la règle correspondante. Soit $\mu_A(o) \in [0; 1]$ la fonction d'évaluation de la règle associé au concept A . Nous choisissons les fonctions minimum, maximum et complément pour représenter respectivement les opérateurs **et**, **ou** et **non** de la

1. <http://www.uml.org/>

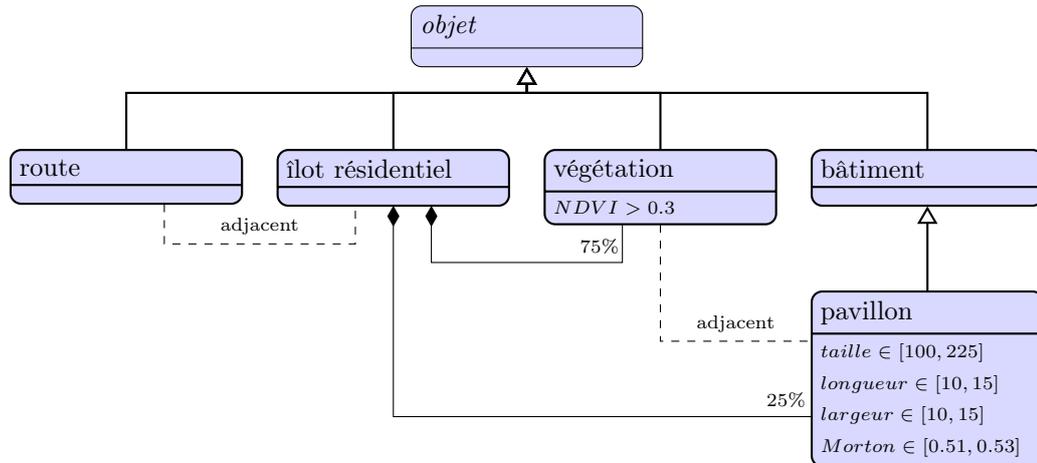


FIGURE 4.2 – Extrait d’une hiérarchie de concepts et des relations associées. En bleu les différents concepts. La notation utilisée est inspirée de la notation UML.

logique floue. Notre problème étant un problème de classification, l’opérateur de défloutage est l’opérateur maximum, présenté dans l’équation 3.10, que nous rappelons ici :

$$d_{\max} = \{x_j | \mu_j = \max(\mu_i(o)), i \in [1, n]\}$$

Dans cette configuration, x_j correspond au score du concept j (celui dont l’évaluation a produit le meilleur score), n est le nombre de concepts à évaluer. d_{\max} est le score déflouté qui permet de prendre une décision. La prise de décision est décrite dans l’algorithme 4, elle dépend d’un seuil minimum.

Avec cette méthode et une base de connaissances fournie par l’expert, nous pouvons classer les objets d’intérêt. Nous obtenons en sortie le graphe d’objets étiquetés. Nous pouvons maintenant utiliser ces objets étiquetés pour analyser le graphe à un plus haut niveau dans le but d’extraire des objets composés.

4 Extraction d’objets composés

À partir du graphe contenant des objets d’intérêt étiquetés par des concepts simples, nous pouvons extraire des objets composés en fusionnant des objets simples selon des règles de composition définies par l’expert. L’extraction des objets composés repose sur un algorithme de croissance de régions adapté aux graphes d’objets d’intérêt.

4.1 État de l’art

Les algorithmes de croissances de régions sont des algorithmes de segmentation dont le principe consiste à faire grandir des régions en y ajoutant des pixels/régions adjacentes qui maximisent un critère de fusion. Le fonctionnement de ces algorithmes repose sur trois points : le choix

Algorithme 4: Raffinement de concept attribué

```

soit  $o$  le concept à évaluer
soit  $\mathcal{C}$  le concept attribué à l'objet  $o$ 
soit  $seuil$  le score minimum pour raffiner l'étiquette du concept courant
tant que  $\mathcal{C}$  a des sous-concepts faire
   $d = \{x_j | \mu_{\mathcal{C}_j} = \max(\mu_{\mathcal{C}_i}(o)), \forall \mathcal{C}_i \preceq_{\Theta} \mathcal{C}\}$ 
  si  $d > seuil$  alors
     $\mathcal{C} = \mathcal{C}_j$ 
  sinon
    arrêter

```

des régions initiales (appelées graines), le critère de fusion, le critère d'arrêt. [Fan et al. \(2005\)](#) proposent une revue des algorithmes de croissance de régions par graines.

[Adams et Bischof \(1994\)](#) proposent une méthode de croissance de régions par graines. Les graines sont choisies manuellement. Le critère de fusion est un critère d'homogénéité basé sur le niveau de gris des pixels. L'algorithme s'arrête lorsqu'il n'y a plus de pixel à évaluer. Les auteurs valident leur méthode sur des images médicales en nuances de gris et concluent que leur algorithme est rapide, efficace et facilement adaptable aux images couleurs. Cependant [Mehnert et Jackway \(1997\)](#) soulignent que les résultats de la méthode proposée par [Adams et Bischof \(1994\)](#) dépend de l'ordre de traitement des pixels. Ils proposent une amélioration de l'algorithme de croissance de régions par graines indépendante de l'ordre de traitement des pixels. Cette méthode est basée sur l'utilisation d'une file de priorité. Les résultats sont similaires à la méthode originale, mais ne dépendent plus de l'ordre des graines.

[Hojjatoleslami et Kittler \(1998\)](#) proposent une méthode similaire dans laquelle à chaque étape il n'y a qu'un seul pixel qui présente les propriétés nécessaires pour être fusionné à la région courante. Pour contrôler le processus de croissance, les auteurs utilisent des mesures de discontinuité appelées contraste moyen et contraste périphérique.

Pour s'affranchir du problème du choix des graines, [Shih et Cheng \(2005\)](#) proposent une méthode automatique. Pour qu'un pixel soit considéré comme une graine, il doit satisfaire deux conditions : avoir une similarité à ses voisins supérieure à un seuil donné et une distance euclidienne à tous ses voisins inférieure à un seuil donné. Les régions sur-segmentées sont fusionnées dans un post-traitement. Les auteurs testent leur méthode sur des images couleurs.

[Fan et al. \(2001\)](#) combinent un détecteur de contour isotropique amélioré et une technique de seuillage entropique rapide pour extraire les contours d'une image. Les centroïdes des contours générés sont utilisés comme graines d'un algorithme de croissance de régions. Les auteurs utilisent cette méthode de segmentation pour extraire automatiquement des visages sur des images en nuances de gris.

Nous pouvons voir que les méthodes basées sur la croissance de régions sont souvent utilisées en analyse d'images lorsque l'on souhaite fusionner des objets selon un critère. Les objets peuvent être des pixels ou des segments. Dans notre cas, le choix du critère de fusion est simple : nous utilisons l'évaluation de la règle de composition. De même, le critère d'arrêt va dépendre de la règle de composition.

4.2 Proposition : Algorithme d'extraction des objets composés

La méthode d'extraction des objets composés que nous proposons repose sur un algorithme de croissance de régions par graines adapté aux graphes d'objets d'intérêt. Les graines choisies sont les objets qui maximisent le score de la règle définissant le concept à extraire. Le critère de fusion est la fonction de gain présenté dans l'équation 4.8. L'algorithme s'arrête lorsque le gain maximal est inférieur à zéro.

La règle de composition nous donne les étiquettes des objets pouvant entrer dans la composition d'un objet composé. La première étape de l'algorithme consiste en l'extraction de toutes les composantes connexes composées uniquement de ces étiquettes (*i.e.* végétation et pavillon pour le concept composé *ilot résidentiel*). Ensuite pour chaque composante connexe extraite, la croissance de régions contrainte prend place. Toutes les paires d'objets adjacents sont évaluées : si la fusion de deux objets donne un score supérieur à chacun des scores des deux objets fusionnés, cet objet fusionné est conservé et le processus itère. Lorsqu'il n'y a plus de fusion possible, tous les candidats sont évalués par la règle de composition. Si le score obtenu est supérieur à un seuil, l'objet est conservé et ajouté définitivement dans le graphe. Dans cet algorithme nous avons besoin de calculer le gain lors de la fusion de deux objets d'intérêt. La formule du calcul du gain est donnée dans l'équation 4.8, où o est l'objet composé et $o1$ et $o2$ sont les objets composant o .

$$\text{gain}(o) = \mu(o) - \max(\mu(o1), \mu(o2)) \quad (4.8)$$

L'algorithme 5 présente l'algorithme de croissance de régions contrainte.

Algorithme 5: Extraction d'objets composés par croissance de régions contrainte

Entrées : *seuil* → score minimal pour qu'un objet soit considéré comme satisfaisant la règle

soit CC une composante connexe contenant uniquement des objets simples

soit $objList = CC$ la liste des objets candidats

soit $compList = \emptyset$ la liste des objets composés temporaires

pour tous les couples d'objets o_1 et o_2 adjacents dans $objList$ faire

 créer o_c l'objet composé de o_1 et o_2

 ajouter o_c dans $compList$

tant que $compList \neq \emptyset$ faire

$o = \text{meilleurCandidat}(compList)$

si $\text{gain}(o) \geq 0$ alors

 soit $o_1 = \text{filsGauche}(o)$

 soit $o_2 = \text{filsDroit}(o)$

$objList = objList \setminus \{o_1, o_2\}$

 supprimer de $compList$ tous les objets qui contiennent o_1 ou o_2

 mettre à jour $compList$ avec les paires d'objets adjacents à o

 insérer o dans $objList$

pour tous les objets $o \in objList$ faire

si $\mu(o) > \text{seuil}$ alors

 ajouter o dans le graphe

5 Application

Nous souhaitons tester notre méthode sur deux segmentations. Celles-ci sont présentées dans l'annexe B. Nous utilisons le dictionnaire de données fourni par l'expert. Celui-ci contient notamment une définition des concepts *pavillon* et *îlot urbain résidentiel*. Nous utilisons l'algorithme 4 pour identifier les objets simples, puis nous utilisons l'algorithme 5 pour extraire les objets composés de type *îlot résidentiel*.

Dans l'exemple présenté dans la figure 4.3 nous souhaitons extraire les îlots urbains dans un quartier résidentiel. L'image en entrée est donnée dans la figure 4.3(a). Nous testons notre méthode sur les deux segmentations. Les figures 4.3(b) et 4.3(c) montrent respectivement le résultat de l'identification des pavillons et l'extraction des îlots résidentiels sur la segmentation-classification. Les figures 4.3(d) et 4.3(e) montrent respectivement le résultat de l'identification des pavillons et l'extraction des îlots résidentiels sur la segmentation de travail. Les pavillons sont représentés en vert (pour contraster avec l'image en fausses couleurs), les îlots résidentiels sont représentés en bleu.

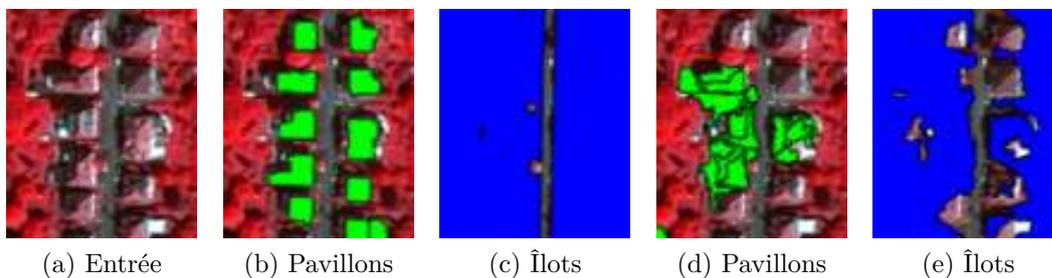


FIGURE 4.3 – Illustration du résultat des algorithmes 4 et 5 sur une imagerie issue de l'image PLEIADES. La figure 4.3(a), montre l'image en entrée. Les figures 4.3(b) et 4.3(c) montrent respectivement le résultat de l'identification des pavillons et l'extraction des îlots résidentiels sur la segmentation-classification. Les figures 4.3(d) et 4.3(e) montrent respectivement le résultat de l'identification des pavillons et l'extraction des îlots résidentiels sur la segmentation de travail.

Nous pouvons observer que dans le cas de la segmentation parfaite, l'identification des pavillons à l'aide de l'algorithme 4 est efficace et donne de bons résultats. L'extraction des îlots résidentiels à partir de cette étape intermédiaire montre la capacité de l'algorithme 5 à extraire les objets composés à partir d'une bonne classification préalable. Cependant, lorsque nous appliquons la même chaîne de traitement avec les mêmes règles sur la segmentation de travail, les résultats sont mauvais. En effet, dans le cas de la segmentation de travail, l'algorithme 4 n'est pas capable d'identifier correctement tous les pavillons car l'image est sursegmentée. Les pavillons n'étant pas correctement identifiés, il est très difficile de bien extraire les objets composés.

Cette expérience nous montre que notre algorithme d'extraction d'objets composés est efficace, dans le cas où nous avons une bonne segmentation et une bonne classification préalables des objets d'intérêt. Si cette classification n'est pas correcte, il n'est pas possible d'extraire les objets composés.

Conclusion et contributions

À partir de la base de connaissances fournie par l'expert, il est théoriquement possible de classer les objets d'intérêt. Ceci repose sur l'hypothèse qu'un algorithme de segmentation soit

capable d'extraire correctement des objets d'intérêt (un segment correspond à un objet d'intérêt). Or cette hypothèse n'est pas juste, rendant extrêmement difficile l'utilisation de connaissances expertes pour l'analyse d'image de télédétection. En effet, les expériences menées lors de mon stage de master et au début de cette thèse ont montré que les segments produits par un algorithme de segmentation *ne correspondent pas* aux objets d'intérêt décrits par l'expert. Ceci est dû au phénomène de sur- ou sous-segmentation. La classification des objets d'intérêt est ainsi biaisée par ce manque de correspondance. Ceci est particulièrement vrai dans le cas des images à très haute résolution spatiale telles que les images PLEIADES où le phénomène de sur-segmentation est très marqué.

Par conséquent l'extraction des objets composés est très difficilement réalisable car elle repose sur un étiquetage supposé correct des objets d'intérêt. En fait, toute utilisation de connaissances dans un processus d'analyse d'image de télédétection repose sur une détection correcte des objets d'intérêt dans la segmentation. Mais les algorithmes de segmentation usuels ne sont pas capables d'extraire des objets géographiques, ils sont uniquement capables de regrouper ensemble des pixels respectant un critère d'homogénéité (par exemple radiométrique). Dans une image à très haute résolution spatiale, telle que les images QUICKBIRD ou PLEIADES, les objets d'intérêt tel que les pavillons sont très détaillés et ne sont pas vus comme homogènes : ils se retrouvent découpés en un ou plusieurs segments. L'exemple le plus frappant est la différence radiométrique entre deux pans de toit dont l'orientation au soleil est différente, comme illustré dans la figure 4.4. Les morceaux de toit sont toujours rouges, mais pas de la même nuance de rouge. Il n'y a pas de relation bijective entre segmentation et objets d'intérêt. La connaissance de l'expert n'est ainsi pas utilisable en l'état dans un processus d'analyse d'image de télédétection à très haute résolution spatiale.

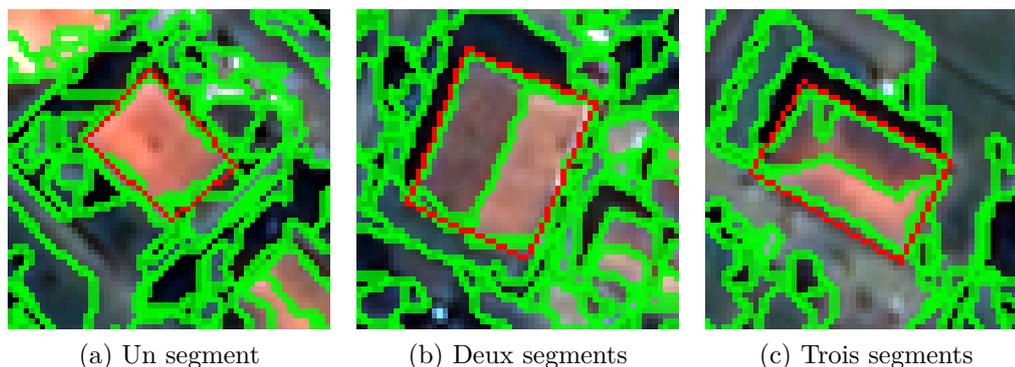


FIGURE 4.4 – Relation entre objets d'intérêt et segments. En vert les contours de la segmentation, en rouge des exemples de pavillons. Ces trois images sont extraites de la même segmentation d'une image QUICKBIRD. Nous voyons dans chaque image un pavillon découpé différemment.

La solution que nous proposons pour remédier à ce problème est de baisser le niveau d'abstraction. En considérant un objet d'intérêt comme un objet composé par un ou plusieurs segments, les méthodes d'analyse d'image basées sur la connaissance sont toujours valides. Il suffit de décrire un objet d'intérêt par une règle de composition. Cependant, l'expert n'est pas capable de donner une telle règle pour tous les objets, car cette règle va dépendre du type et de la résolution de l'image en entrée, de l'algorithme de segmentation utilisé et de ses paramètres, de l'orientation ou encore de l'heure d'acquisition de l'image. Avec une telle combinaison de facteurs, il est

impossible pour l'expert de donner une règle de composition générique décrivant un concept particulier. Pour obtenir une telle règle, nous proposons d'utiliser un algorithme d'apprentissage. La méthode utilise des exemples fournis par l'expert pour extraire les segments correspondants et les classes pour ensuite apprendre une règle de composition par rapport à la classification préalable. La règle apprise peut finalement être utilisée pour extraire les objets composés correspondants à l'aide de l'algorithme 5, qu'ils soient composés d'un ou de plusieurs segments. Il est ainsi possible de faire le lien entre le résultat d'une segmentation et la connaissance de l'expert. La méthode d'apprentissage de règles de composition est présentée dans le chapitre 5.

Valorisation scientifique : Les conclusions présentées dans ce chapitre ont donné lieu à une publication lors de l'atelier *Fouille de données complexes* de la conférence *Extraction et Gestion de Connaissances 2012* (Belarte et Wemmert, 2012).

Réalisation logicielle. Le travail réalisé dans ce chapitre a été réalisé dans une bibliothèque logicielle écrite en langage C++. La partie *traitement d'image* utilise la bibliothèque OTB. Le fonctionnement de l'application repose sur le design pattern *injection de dépendance* : le déroulement de l'application est intégralement paramétré par un fichier de configuration écrit en langage XML. Ceci permet une parfaite modularité du code et permet de conserver un historique des expériences effectuées.

Troisième partie

Approche multi-niveaux pour l'apprentissage de règles et l'intégration d'information contextuelle

★ Résumé ★

Dans cette partie, nous proposons une chaîne de traitement pour l'extraction d'objets d'intérêt dans une image segmentée en combinant un algorithme d'apprentissage de règles de composition et une approche montante-descendante. Dans le premier chapitre, nous présentons l'algorithme d'apprentissage de règles de composition, la méthode *ARC*, et nous l'appliquons pour extraire des pavillons dans des quartiers résidentiels. Nous observons que l'information contextuelle n'est que peu pertinente dans la méthode *ARC*. Dans le second chapitre, nous présentons l'approche montante-descendante. Cette méthode repose sur l'identification d'objets composés de haut niveau d'interprétation (tel que des îlots urbains) pour ajouter de l'information contextuelle et utiliser cette information pour mettre à jour la classification aux niveaux inférieurs. Dans le troisième chapitre, nous proposons un protocole expérimental complet pour valider méthodiquement chaque paramètre entrant en compte dans le déroulement de la chaîne de traitement. Des tests seront effectués sur une image PLEIADES de 9211×11275 pixels représentant le nord-est de Strasbourg.

5

Apprentissage de règles de composition pour l'analyse d'images de télédétection à très haute résolution spatiale

Sommaire

1	Méthode proposée : Apprentissage de règles de composition . . .	82
1.1	Extraction des exemples	82
1.2	Apprentissage de règles de composition	83
2	Apprentissage de règles floues	87
2.1	Classification Ascendante Hiérarchique (CAH)	87
2.2	Proposition : Apprentissage de règles floues basé sur CAH.	88
2.3	Exemple	88
2.4	Évaluation de la méthode ARF_{CAH}	91
3	Validation expérimentale de la méthode ARC	92
3.1	Protocole expérimental	92
3.2	Expérimentations préliminaires sur une image QuickBird	93
3.3	Expérimentations sur des images Pléiades	93
	Conclusion et contributions	96

Nous avons vu dans le chapitre précédent qu'il n'existe pas de relation bijective entre la segmentation d'une scène et les objets d'intérêt contenus dans cette même scène, car un objet d'intérêt n'est pas décrit de la même façon par l'expert et par l'image.

Dans ce chapitre, nous proposons une méthode d'analyse d'images qui fait le lien entre ces deux descriptions d'une même scène. Pour ce faire nous proposons d'apprendre comment une classe d'objets d'intérêt est décomposée dans l'image segmentée. Cette méthode repose sur l'apprentissage de règles floues pour classer les segments, ainsi que sur l'apprentissage d'une règle de composition décrivant la classe recherchée en fonction de sa composition dans la classification précédente. Ce processus automatique utilise des exemples. Nous commençons par présenter la méthode de manière générale avant de détailler les différentes étapes. Nous présentons ensuite les résultats obtenus par notre méthode sur les zones de tests que nous avons sélectionnées.

1 Méthode proposée : Apprentissage de règles de composition

Nous souhaitons apprendre comment sont composés les objets correspondant à une classe d'objets d'intérêt en fonction de la segmentation. La méthode que nous proposons est l'algorithme d'Apprentissage de Règles de Composition (abrégé *ARC*). La méthode *ARC* prend en entrée un graphe d'adjacence créé à partir d'une image segmentée (voir chapitre 4). Les objets contenus dans le graphe correspondent aux segments identifiés par la segmentation. Deux entrées supplémentaires sont nécessaires : les exemples (fournis par l'expert) ainsi qu'un algorithme d'apprentissage de règles floues. Les entrées et sortie de l'algorithme ainsi que l'interaction avec la base de connaissances sont présentées dans la figure 5.1.

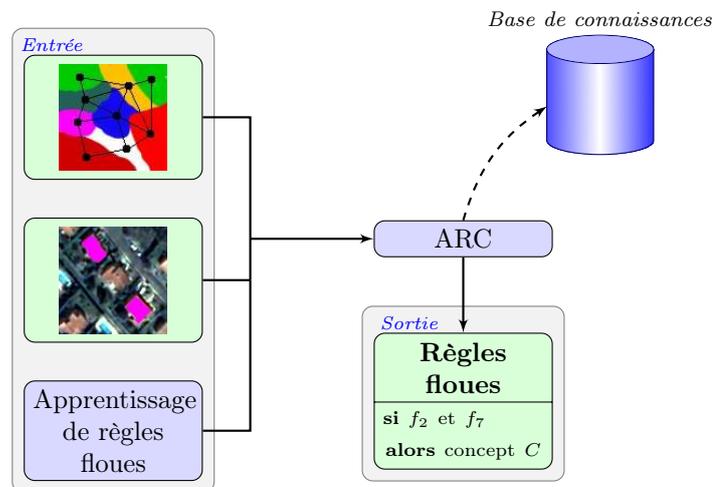


FIGURE 5.1 – Entrées et sortie de la méthode *ARC*. Les entrées sont représentées à gauche. La base de connaissance et la base de règles sont enrichies avec les concepts (respectivement les règles) générés en sortie.

Avant d'appliquer la méthode *ARC*, les objets correspondant aux exemples sont extraits du graphe. Ensuite l'apprentissage prend place : des règles floues sont apprises et utilisées pour étiqueter les objets correspondant aux exemples. Finalement, une règle de composition unique est apprise à partir de l'étiquetage des exemples. La méthode *ARC* a deux paramètres : le taux d'erreur pour l'extraction des exemples et le paramètre d'affinage pour la génération de la règle de composition. Les différentes parties de la méthode et les paramètres sont détaillées dans la suite de ce chapitre.

1.1 Extraction des exemples

L'expert donne les exemples sous la forme d'une liste de polygones géoréférencés. Les objets correspondant à ces exemples sont extraits. Le graphe est parcouru sommet par sommet et les objets correspondant à chacun des exemples sont conservés. Ce processus est décrit dans l'algorithme 6. Le seuil indique quel pourcentage de recouvrement minimal est nécessaire pour qu'un objet soit considéré comme faisant partie d'un exemple, et le recouvrement minimal nécessaire

pour qu'une agglomération d'objets soit considérée comme étant un exemple. Ceci nous permet de ne pas considérer les segments ne recouvrant pas assez le polygone exemple. De plus nous ne considérons pas non plus les objets composés dont la taille est faible par rapport au polygone exemple. Cet algorithme a été ajouté à la chaîne de traitement après les premières expériences (voir section 3.2), celles-ci ayant montré que l'algorithme initial (qui considérait comme objet candidat tout objet intersectant le polygone exemple) pouvait conserver des objets aberrants (trop grand ou trop petit). L'expert donne un taux d'erreur acceptable et le seuil est calculé en conséquence.

Ce paramètre varie selon la segmentation et la nature de la scène à analyser. Nos expériences nous ont permis de voir que dans le cas de l'analyse de paysages urbains, une valeur acceptable pour le taux d'erreur est 0.5. Une étude détaillée sur l'influence du taux d'erreur est présentée dans le chapitre 7, section 2.1.

Algorithme 6: Sélection des segments candidats, avec $taille(o)$ la fonction qui renvoie la taille de l'objet o et $taux_{erreur}$ le paramètre renseigné par l'expert.

```

seuil = 1 - taux_erreur
liste = ∅
pour tous les exemples  $ex$  faire
    taille = 0
    liste_temporaire = ∅
    pour tous les objets d'intérêt  $o$  du graphe faire
        si  $\frac{taille(o \cap ex)}{taille(o)} > seuil$  alors
            liste_temporaire = liste_temporaire  $\cup$   $\{O\}$ 
            taille = taille + taille( $o$ )
        si  $|taille - taille(ex)| < seuil$  alors
            liste = liste  $\cup$  liste_temporaire
retourner liste

```

1.2 Apprentissage de règles de composition

Dans cette section nous détaillons la méthode *ARC*. Elle repose sur l'utilisation d'un algorithme d'apprentissage de règles floues. L'objectif de la méthode *ARC* est de proposer une règle de composition représentant l'ensemble des exemples fournis par l'expert. Pour cela, il est nécessaire de pouvoir étiqueter les segments composant ces exemples. Ces étiquettes sont apprises automatiquement : l'algorithme d'apprentissage de règles produit des règles, celles-ci vont générer des concepts qui vont enrichir la base de connaissances. Une première version de la méthode est illustrée dans l'algorithme 7.

Algorithme 7: Méthode *ARC*, première version

```

apprentissage de règles floues (algorithme 11)
étiquetage des segments (algorithme 4)
génération d'une règle de composition unique (algorithme 9)

```

Nous nous intéressons à l'influence du contexte spatial (relation de voisinage) entre les objets d'intérêt. Nous choisissons de prendre en compte les relations d'adjacence lors de l'apprentissage. Cependant, au début du processus, les segments ne sont pas étiquetés, il n'est donc pas possible d'apprendre les règles spatiales entre ces segments. La phase d'apprentissage doit être découpée en deux étapes : premièrement des règles sont apprises sur les attributs propres ; deuxièmement, après ré-étiquetage lors de la phase précédente, les règles spatiales sont apprises. Cette seconde version est illustrée dans l'algorithme 8.

Algorithme 8: Méthode *ARC*, seconde version

```

apprentissage de règles floues sur les attributs propres (algorithme 11)
étiquetage des segments (algorithme 4)
apprentissage de règles floues sur les relations spatiales (algorithme 11)
réétiquetage des segments (algorithme 4)
génération d'une règle de composition unique (algorithme 9)

```

La méthode d'apprentissage de règles simples n'intervient pas dans la structure de l'algorithme et peut donc être n'importe quelle méthode d'apprentissage de règles. Nous utilisons la méthode *ARFCAH* comme algorithme d'apprentissage de règles. L'algorithme de génération de la règle de composition est présentée dans l'algorithme 9. Le facteur d'affinage permet de moduler l'écart-type de la classe pour lui donner plus ou moins d'amplitude.

Algorithme 9: Génération de la règle de composition, paramétrée par le facteur d'affinage.

```

pour tous les propriétés  $p_i$  faire
  | calculer l'écart-type de  $p_i$ 
soit  $r_{comp}$  une règle de composition
pour tous les exemples  $e_j \in E$  faire
  | générer une règle  $r_j$ 
  | pour tous les propriétés  $p_i$  faire
  |   | générer la fonction d'appartenance gaussienne  $\mu_i(e_j)$ 
  |   |   (moyenne = valeur moyenne de la propriété  $i$  de  $e_j$ )
  |   |   (écart-type = écart-type de la propriété  $p_i \times$  facteur d'affinage)
  |   |_ ajouter  $\mu_i(e_j)$  à  $r_j$  avec l'opérateur et
  |_ ajouter  $r_j$  à  $r_{comp}$  avec l'opérateur ou

```

La figure 5.2 illustre le processus d'apprentissage dans son intégralité. L'algorithme d'apprentissage de règles de composition est découpé en deux étapes d'apprentissage de règles floues simples (en pointillé dans la figure 5.2).

Avec deux apprentissages successifs, la base de connaissances est enrichie par deux niveaux de concepts. Le premier niveau est composé de concepts générés par la phase d'apprentissage sur les attributs propres, le second niveau est généré par la phase d'apprentissage sur les relations spatiales. Un exemple d'une hiérarchie à la sortie de la méthode *ARC* est donné dans la figure 5.3. Le concept à apprendre est le concept *pavillon*. La phase d'apprentissage sur les attributs propres a généré deux règles qui ont chacune généré un concept (A et B), la phase d'apprentissage sur les relations spatiales a généré trois règles qui ont permis de générer quatre concepts (C, D, E et F). En effet, une règle spatiale peut être discriminante pour deux concepts dans la hiérarchie.

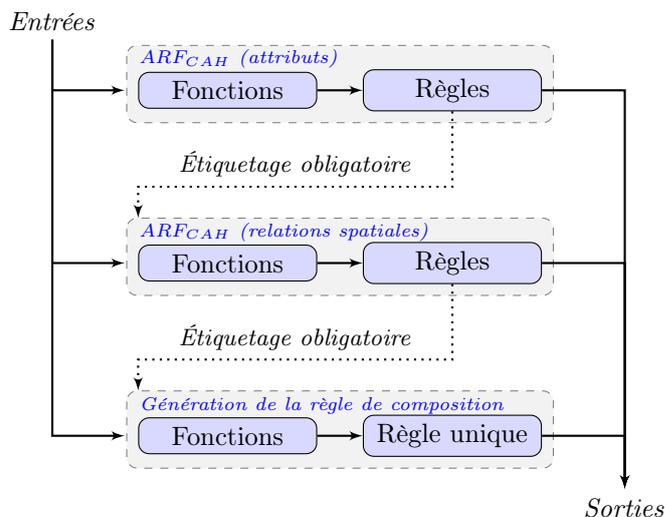


FIGURE 5.2 – Organigramme résumant le processus d'apprentissage. Les pointillés illustrent les différentes étapes de l'algorithme : apprentissage pour les propriétés et pour les relations spatiales, génération de la règle de composition. Un exemple de hiérarchie produite par l'algorithme est illustré dans la figure 5.3.

Il faut noter ici que le concept *D* par exemple n'est pas défini uniquement par la règle 4, mais par la règle 1 **et** la règle 4, ce qui le différencie du concept *E*.

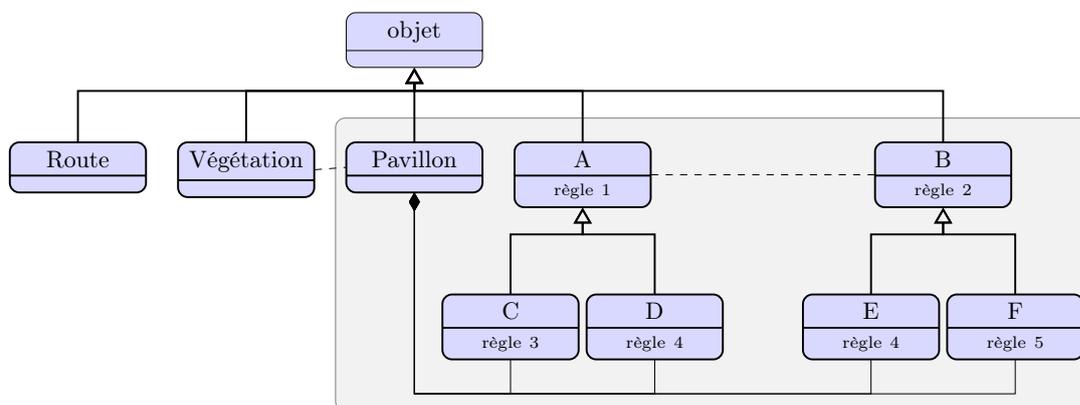


FIGURE 5.3 – La nouvelle hiérarchie de concepts après apprentissage. Les nouveaux concepts sont encadrés. Le premier niveau (concepts A et B) correspond à la sortie de la phase d'apprentissage sur les attributs propres, le second (concepts C, D, E et F) correspond à la sortie de la phase d'apprentissage sur les relations spatiales.

Un exemple de synthèse résumant la méthode est donné dans la figure 5.4.

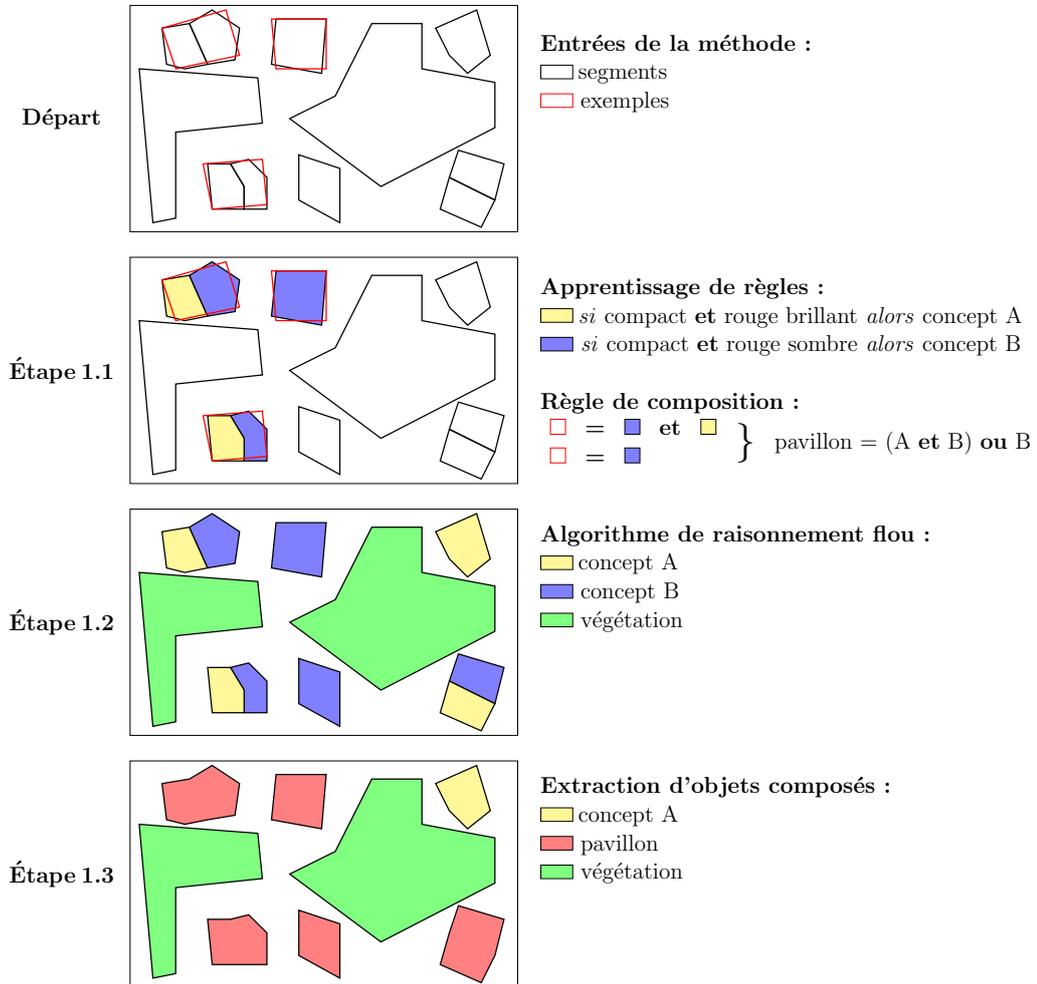


FIGURE 5.4 – Différentes étapes de l'algorithme d'apprentissage. Les entrées et les paramètres sont représentés dans l'étape *départ*. L'*étape 1.1* représente l'apprentissage des règles floues : d'abord la caractérisation des segments, ensuite l'apprentissage de la règle de composition. L'*étape 1.2* illustre la classification des segments à partir des règles apprises précédemment. L'*étape 1.3* montre le résultat après extraction des objets composés.

Une fois l'apprentissage terminé, la nouvelle règle de composition peut être utilisée pour extraire les objets composés par l'algorithme d'extraction des objets composés présenté dans le chapitre 4. Dans la section 3 nous mettrons en pratique la méthode *ARC* sur des images à très haute résolution spatiale. Dans la section suivante, nous présentons l'algorithme d'apprentissage de règles floues *ARF_{CAH}*.

2 Apprentissage de règles floues

Nous souhaitons apprendre des règles floues de manière automatique. Nous avons montré dans le chapitre 3 que la littérature est riche dans ce domaine. Nous proposons ici une méthode qui apprend indépendamment des fonctions d'appartenance pour chaque propriété et qui dans un second temps combine ces fonctions pour apprendre des règles minimalistes. La méthode que nous proposons utilise l'algorithme de classification ascendante hiérarchique. Ce choix est motivé par le fait que la classification ascendante hiérarchique est un algorithme automatique qu'il est possible de paramétrer sans connaissance *a priori* du problème (par exemple, avec l'utilisation du critère de Ward).

2.1 Classification Ascendante Hiérarchique (CAH)

CAH est un algorithme de classification automatique (Lance et Williams, 1967). Müllner (2011) propose une revue des différentes méthodes de classification ascendante hiérarchique. Cet algorithme repose sur une mesure de dissimilarité D entre classes qui est utilisée pour déterminer quelles sont les classes à fusionner. La dissimilarité que nous avons choisie est la distance euclidienne au carré, notée d_2^2 . Au départ, les objets sont chacun répartis dans une classe. À chaque étape, les deux classes les plus proches sont fusionnées. La sortie de cet algorithme est un arbre binaire de partition. Le nombre de classes à conserver dépend du critère d'arrêt. Le fonctionnement de la classification hiérarchique ascendante est présenté dans l'algorithme 10.

Algorithme 10: Classification hiérarchie ascendante

Entrées : graphe G

Sorties : arbre binaire de partition contenant le résultat de la classification

pour tous les objets d'intérêt O de G faire

 | générer une classe à un seul élément contenant uniquement O

tant que le critère d'arrêt n'est pas atteint faire

 | Trouver deux classes C_i et C_j dont la distance $D(i, j)$ est la plus faible

 | Fusionner C_i et C_j

Le critère d'arrêt peut être un nombre de classes prédéfini ou un critère automatique. Dans notre cas, nous ne connaissons pas le nombre de classes recherchées, nous choisissons donc une méthode automatique. La mesure de dissimilarité de Ward (1963) calcule la différence d'inertie intra-classe (ou inter-classe) avant et après fusion de deux classes. L'équation 5.1 donne la formule de la dissimilarité de Ward, où d_W est la mesure calculée, C_u et C_v sont deux classes, $n_i = \text{card}(C_i)$ et $D(C_u, C_v)$ est la dissimilarité entre deux classes.

$$d_W(C_u, C_v) = \frac{n_u \cdot n_v}{n_u + n_v} \cdot D(C_u, C_v) \quad (5.1)$$

Lorsque la dissimilarité de Ward entre deux classes est trop élevée (*i.e.* l'augmentation de l'inertie inter-classe ou la perte d'inertie intra-classe trop élevée), la fusion de ces deux classes produirait une nouvelle classe trop hétérogène et donc non cohérente. Ce seuil critique dépend des données, par conséquent nous utiliserons ce seuil comme paramètre de l'algorithme CAH.

2.2 Proposition : Apprentissage de règles floues basé sur CAH.

La méthode d'apprentissage de règles que nous proposons repose sur l'algorithme CAH et nécessite deux paramètres : $s_{fonction}$ le seuil du critère de Ward pour l'apprentissage des fonctions et $s_{règle}$ le seuil du critère de Ward pour l'apprentissage des règles. Nous appelons cette méthode ARF_{CAH} . L'objectif est d'extraire rapidement un ensemble de règles. Dans un premier temps, les fonctions d'appartenance sont apprises indépendamment pour chaque propriété. Pour cela, une CAH est effectuée pour chacune des propriétés du système. Chaque CAH produit une ou plusieurs fonctions d'appartenance. Dans un second temps, les fonctions apprises sont utilisées pour définir les objets, chaque objet est défini par un vecteur dont chaque élément est la valeur de retour d'une des fonctions d'appartenance apprises. Une nouvelle CAH est effectuée sur les objets nouvellement définis. Le résultat permet de générer une règle par classe.

Apprentissage des fonctions d'appartenance. Une CAH est effectuée pour chacune des propriétés. Pour chaque classe générée par la CAH, une fonction trapézoïdale est générée sur la propriété correspondante. La moyenne et l'écart-type de la classe sont calculés. Le noyau et le support sont calculés comme défini dans le chapitre 3 : $n_A = [m - 2\sigma, m + 2\sigma]$ et $s_A = [m - 3\sigma, m + 3\sigma]$.

Apprentissage des règles. Pour apprendre les règles, chaque segment se voit caractérisé par le résultat de toutes les fonctions d'appartenance apprises précédemment. Un segment est alors caractérisé par un vecteur de réels entre 0 et 1. Une nouvelle CAH est effectuée sur ces objets.

L'algorithme d'apprentissage de règles floues ARF_{CAH} est résumé dans l'algorithme 11.

Algorithme 11: Apprentissage de règles floues

soit P l'ensemble des propriétés, soit Ω l'ensemble des objets décrits par P
 soit $F = \emptyset$ l'ensemble des fonctions d'appartenance, $R = \emptyset$ l'ensemble des règles
pour tous les propriétés p de P faire
 | faire une CAH pour apprendre les fonctions d'appartenances sur p
 | ajouter les fonctions apprises à F
pour tous les objets d'intérêt O de Ω faire
 | décrire O avec toutes les fonctions de F
 faire une CAH sur Ω
 ajouter les règles apprises à R

2.3 Exemple

Apprentissage des fonctions d'appartenance. Supposons que le système ait trois propriétés (l'indice NDVI, la taille et l'élongation) et que l'expert ait défini trois fonctions sur ces propriétés (*élevé*, *compact* et *allongé*). Nous utilisons la notation définie dans le chapitre 3. La base de fonctions d'appartenance avant apprentissage est la suivante :

$$(NDVI, [-1, 1], \{\text{élevé}\})$$

$$(taille, [1, 1000], \{\})$$

(*élongation*, [1, 5], {*compact*, *allongé*})

Supposons que les exemples aient permis d'extraire n segments notés s_1 à s_n . Le tableau 5.1 résume les données en entrée de l'étape de classification.

		s_1	s_2	s_3	...	s_n
NDVI	[-1, 1]	0.1	-0.1	0	...	0.05
taille	[1, 1000]	103	258	186	...	227
élongation	[1, 5]	1.2	3.9	1.8	...	1.4

TABLE 5.1 – Tableau présentant les valeurs de chaque segment pour chacune des trois propriétés.

Pour apprendre les fonctions d'appartenance, trois CAH sont effectuées (une par propriété). Les arbres binaires de partitions produits par ces CAH sont illustrés dans la figure 5.5. La ligne en pointillés rouges représente le seuil du critère de Ward. Dans le cas de la CAH sur le NDVI, il n'y a qu'une seule classe en dessous du critère de Ward, par conséquent l'apprentissage sur le NDVI va générer une fonction d'appartenance sur cette propriété. Nous notons cette fonction f_0 . Dans le cas de la CAH sur la taille (respectivement l'élongation), il y a deux classes en dessous du critère de Ward, par conséquent l'apprentissage sur la taille (respectivement l'élongation) va générer deux fonctions d'appartenance que nous notons f_1 et f_2 (respectivement f_3 et f_4). Ces fonctions sont représentées dans la figure 5.6. Nous pouvons voir dans la figure 5.6(c) que l'algorithme peut apprendre des fonctions qui chevauchent des fonctions existantes.

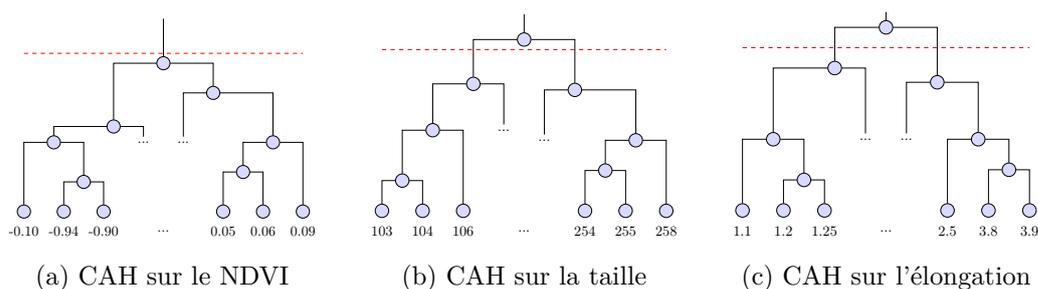


FIGURE 5.5 – Arbres binaires de partition produits par les trois CAH.

La base de fonctions après l'apprentissage inclut ces nouvelles fonctions.

(*NDVI*, [-1, 1], {*élevé*, f_0 })

(*taille*, [1, 100], { f_1 , f_2 })

(*élongation*, [1, 5], {*compact*, *allongé*, f_3 , f_4 })

Apprentissage des règles. Les fonctions d'appartenance f_0 à f_4 sont utilisées pour redéfinir les n segments. Un segment est défini par un vecteur dont chaque élément est le degré d'appartenance de ce segment pour une des fonctions d'appartenance. Le tableau 5.2 présente les données après transformation.

Une CAH est effectuée sur les segments nouvellement définis. Les segments étant définis par des vecteurs de nombres réels dans l'intervalle [0; 1] il n'est pas nécessaire de normaliser et la

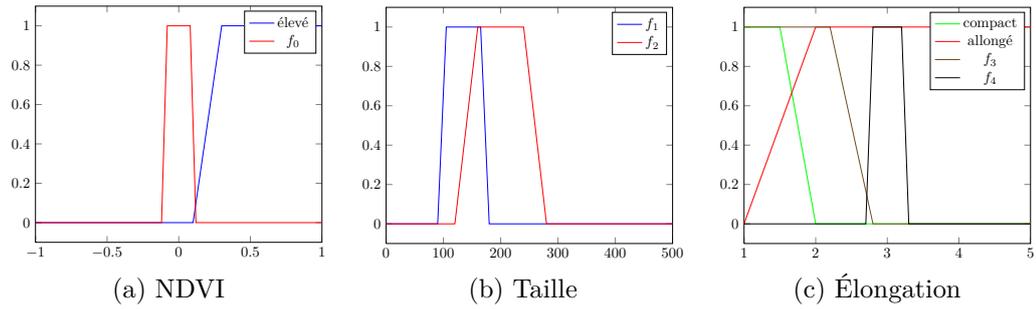


FIGURE 5.6 – Les fonctions d'appartenance sur les trois propriétés après l'apprentissage. Les fonctions apprises et les fonctions fournies par l'expert sont illustrées.

	[f_0, f_1, f_2, f_3, f_4]
s_1	[0.9, 0.5, 0.4, 0.3, 0.4]
s_2	[0.91, 0.9, 0.9, 0.1, 0.2]
\vdots	\vdots
s_n	[0.95, 0.6, 0.1, 0.5, 0.8]

TABLE 5.2 – Les exemples après redéfinition : les segments sont caractérisés par les valeurs de sortie des fonctions apprises précédemment.

distance euclidienne est directement utilisable. L'arbre binaire de partition produit par cette CAH est donné dans la figure 5.7.

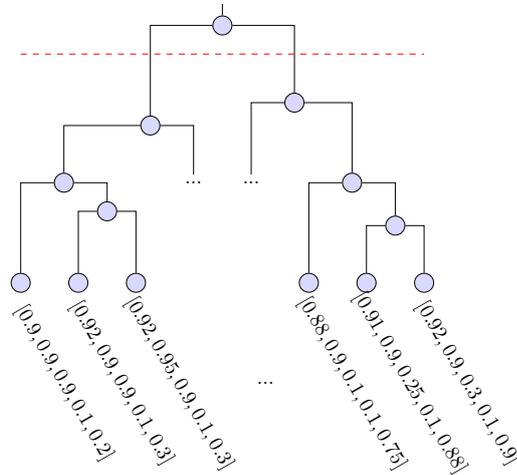


FIGURE 5.7 – Arbre binaire de partition produits par la CAH pour l'apprentissage des règles.

Pour chaque classe à la fin de la CAH, une règle est générée à partir de l'objet moyen définissant cette classe. Seules les fonctions dont la valeur de retour dépasse un seuil fixé par l'expert sont utilisées pour générer une règle. Nous fixons empiriquement ce seuil à 0.85. Par

exemple, supposons que les objets moyens définissant les deux classes de notre exemple soient $[0.92, 0.96, 0.88, 0.15, 0.27]$ et $[0.95, 0.9, 0.33, 0.12, 0.88]$, les règles générées seront :

$$r_0 = f_0 \text{ et } f_1 \text{ et } f_2$$

$$r_1 = f_0 \text{ et } f_1 \text{ et } f_4$$

2.4 Évaluation de la méthode ARF_{CAH}

Pour tester la méthode ARF_{CAH} , nous choisissons d'utiliser des jeux de données de l'UCI¹. Les jeux de données testés sont présentés dans le tableau 5.3. Notre méthode étant basée sur l'algorithme CAH, nous évaluons aussi cet algorithme pour nous y comparer. Nous évaluons la CAH avec différentes valeurs du critère de Ward, ARF_{CAH} avec différentes valeurs du critère de Ward pour l'apprentissage des fonctions et des règles. Différentes mesures de qualité sont utilisées : précision, rappel, F-mesure, indice de Rand, indice de Jaccard. Ces mesures sont détaillées dans les équations suivantes, où vp , vn , fp et fn sont respectivement le nombre de vrais positifs, vrais négatifs, faux positifs et faux négatifs.

$$\text{précision} = \frac{vp}{vp + fp} \quad (5.2)$$

$$\text{rappel} = \frac{vp}{vp + fn} \quad (5.3)$$

$$F - \text{mesure} = 2 \cdot \frac{\text{précision} \cdot \text{rappel}}{\text{précision} + \text{rappel}} \quad (5.4)$$

$$\text{indice de Rand} = \frac{vp + vn}{vp + fp + vn + fn} \quad (5.5)$$

$$\text{indice de Jaccard} = \frac{vp}{vp + fp + fn} \quad (5.6)$$

dataset	#instances	#attributs	#classes
iris	150	4	3
column 2C	310	6	2
column 3C	310	6	3
wine	178	13	3
breast cancer	699	9	2
wdbc	569	30	2
yeast	1484	8	10
parkinson	195	22	2

TABLE 5.3 – Propriétés des différents jeux de données utilisés : nombre d'instances, nombre d'attributs et nombre de classes.

Les résultats obtenus sont présentés dans le tableau 5.4. Nous pouvons voir que les deux algorithmes CAH et ARF_{CAH} produisent des résultats moyens dans l'ensemble. ARF_{CAH} montre des résultats globalement meilleurs sur cinq jeux de données contre un seulement pour CAH. De plus, ARF_{CAH} se démarque sur le jeu de données *breast cancer* avec une F-mesure supérieure à 0.9.

1. University of California Irvine, www.ics.uci.edu/~mlearn/

dataset	méthode	P	R	FM	RI	JI
iris	CAH	0.436	0.492	0.462	0.624	0.301
	ARF_{CAH}	0.625	0.683	0.653	0.761	0.484
column 2C	CAH	0.562	1	0.719	0.562	0.562
	ARF_{CAH}	0.562	1	0.719	0.562	0.562
column 3C	CAH	0.374	1	0.544	0.374	0.374
	ARF_{CAH}	0.471	0.697	0.563	0.595	0.391
wine	CAH	0.500	0.694	0.581	0.662	0.410
	ARF_{CAH}	0.418	0.897	0.570	0.542	0.398
breast cancer	CAH	0.794	0.801	0.798	0.777	0.663
	ARF_{CAH}	0.910	0.922	0.916	0.907	0.845
wdbc	CAH	0.678	0.745	0.710	0.676	0.550
	ARF_{CAH}	0.702	0.865	0.775	0.733	0.633
yeast	CAH	0.251	0.544	0.344	0.537	0.208
	ARF_{CAH}	0.271	0.326	0.296	0.655	0.174
parkinson	CAH	0.629	1	0.771	0.627	0.627
	ARF_{CAH}	0.745	0.836	0.788	0.717	0.650

TABLE 5.4 – Résultats pour différents jeux de données. Deux méthodes non supervisées sont testées : CAH classique et ARF_{CAH}^{auto} . P=précision, R=rappel, FM=F-mesure, RI=indice de Rand, JI=indice de Jaccard.

Il est important de remarquer ici que nous ne cherchons pas à obtenir des résultats parfaits mais à obtenir un algorithme simple et efficace qui donne des résultats cohérents et lisibles dans un problème de faible taille (peu d'instances et peu de classes). L'objectif étant d'utiliser cette méthode dans un processus d'apprentissage plus complexe : la méthode *ARC* (voir section 1.2). En cela les résultats pour les jeux de données *iris*, *yeast*, *breast cancer*, *wdbc* et *parkinson* sont satisfaisants. Ce choix a aussi été motivé par la rapidité de notre méthode dont le temps d'exécution est bien inférieur à la seconde. Nous avons vu dans l'état de l'art du chapitre 3 que de nombreuses méthodes d'apprentissage de règles floues basées sur des réseaux de neurones existent et seraient peut-être plus adaptées dans notre cas. Cependant, ces méthodes fonctionnent comme des boîtes noires (réseaux de neurones, algorithmes évolutionnaires) ou produisent des règles peu lisibles et non exploitables facilement par l'expert.

3 Validation expérimentale de la méthode *ARC*

Dans cette expérience nous nous concentrons sur l'extraction de pavillons dans un quartier résidentiel. Comme énoncé dans le chapitre 4, la segmentation de telles images ne peut que produire une sur-segmentation. Nous choisissons donc de considérer les pavillons comme des objets composés et notre but est d'apprendre pour chaque image une règle de composition décrivant le concept *pavillon*.

3.1 Protocole expérimental

Un bâtiment urbain étant en partie défini par son voisinage, nous choisissons d'utiliser la relation d'adjacence à la végétation et à la route comme relations spatiales car ce sont les relations utilisées pour décrire le concept *pavillon* dans le dictionnaire d'objets urbains proposé par le LIVE. L'algorithme de segmentation utilisé est l'algorithme *meanshift* (Comaniciu et Meer,

2002a). La segmentation est transformée en graphe, la végétation et les routes sont extraites. La végétation est extraite à l'aide de l'algorithme présenté dans le chapitre 4, les routes sont extraites par comparaison aux routes présentes dans le système d'information géographique OpenStreet-Map².

Nous choisissons de tester notre méthode sur trois images. La première est une image QUICK-BIRD d'une résolution de 0,7 mètres, acquise en 2002, représentant un quartier à Saint-Orens, dans la banlieue de Toulouse. Les deux autres sont des images PLEIADES d'une résolution de 0,5 mètres, acquises en 2012, représentant deux quartiers de Strasbourg (le quartier des Quinze et une zone résidentielle de Cronembourg). L'ensemble des jeux de données est présenté en détail dans l'annexe A.

3.2 Expérimentations préliminaires sur une image QuickBird

Dans un premier temps nous cherchons à vérifier l'efficacité de notre méthode sur une image à très haute résolution spatiale QUICKBIRD d'une résolution de 0.7 mètres.

Une fois l'image segmentée et transformée en graphe, le processus d'apprentissage prend place : les exemples sont extraits, les règles sont apprises et les concepts correspondants sont générés puis utilisés pour étiqueter les objets. La figure 5.8(a) montre que deux concepts ont été appris (en jaune et bleu sur l'image). La végétation est représentée en vert, les routes en gris. Enfin, la règle de composition est apprise et le concept composé correspondant est généré. Le résultat est présenté dans la figure 5.8(b). Les pavillons sont extraits avec une précision de 0.82 après comparaison avec une vérité terrain fournie par l'expert géographe. Nous pouvons voir que certains pavillons pourtant évident pour un œil humain n'ont pas été identifiés (par exemple les pavillons notés *a*, *b* et *c* sur l'image 5.8(b)). Après observation il apparaît que l'objet composé correspondant au pavillon *a* est plus grand que les exemples fournis par l'expert et donc ne satisfait pas la contrainte de taille de la règle apprise, les segments composant le pavillon *b* sont mal identifiés dans la phase précédente, l'objet composé correspondant au pavillon *c* ne satisfait pas les contraintes de composition de la règle apprise. Il s'avère que la nature des pavillons n'est pas la même dans l'image : les îlots sont découpés différemment et contiennent des pavillons de différentes formes et tailles. Un biais est induit par le fait que les exemples ne sont pas représentatifs de l'ensemble des types de pavillons présents sur cette image. Ceci montre que le choix et la qualité des exemples est cruciale. Ces conclusions sont à l'origine de l'introduction du paramètre d'affinage dans la génération de la règle de composition.

Les résultats montrent un certain nombre de faux positifs que nous pouvons qualifier de valeurs aberrantes (objets trop petits, ou trop allongés). En effet il existe un biais dans l'extraction des segments correspondant aux exemples. Cette expérience a permis d'identifier ce biais et de le corriger pour éliminer ce type de valeurs aberrantes dans les expériences futures (voir algorithme 6).

Les résultats présentés dans la figure 5.8 ont été obtenus avec les meilleurs paramètres (pour l'algorithme de segmentation et la méthode *ARC*) identifiés par une analyse empirique. Nous évaluerons dans le chapitre 7 l'influence des différents paramètres de la méthode *ARC* sur la *segmentation de travail* du LIVE (voir annexe B).

3.3 Expérimentations sur des images Pléiades

Les résultats des deux expériences suivantes sont présentés dans le tableau 5.5 et dans la figure 5.9.

2. OpenStreetMap : <https://openstreetmap.fr/projet>



(a) En bleu et jaune, les deux concepts appris par la méthode. La végétation en vert, les routes en gris

(b) Les pavillons extraits

FIGURE 5.8 – Application de la méthode sur une image QUICKBIRD (©DigitalGlobe 2001). À gauche le résultat de la phase d'apprentissage : deux concepts sont appris (en bleu et jaune sur l'image). En vert la végétation, en gris les routes. À droite, le résultat de l'extraction des objets composés *pavillon*.

Cronenbourg. Cette première expérience sur une image PLEIADES permet de tester la portabilité de notre méthode. En effet cette petite image représente un quartier résidentiel contenant des pavillons homogènes. Ces pavillons sont plus faciles à identifier. Le résultat de l'apprentissage est donné dans la figure 5.9(a) et évalué dans le tableau 5.5. Nous pouvons voir que les pavillons sont extraits avec une F-mesure de 0.746 après comparaison avec la vérité terrain, ce qui est un bon résultat. Le taux de faux positifs est très faible : seul deux pavillons sont extraits en dehors des deux îlots centraux. La précision et le rappel inférieur à 0.8 sont expliqués par le fait que des segments sont incorrectement étiquetés *composant de pavillon* (allée de garage, terrasse, à cause de leur signature spectrale) et ainsi l'extraction des objets composés est partiellement biaisée.

Quartier des Quinze. Pour cette seconde expérience, la scène est plus grande et plus variée. Le paysage est plus complexe : les pavillons sont différents en taille et signature spectrale. Les îlots sont plus complexes : grands jardins, présence de piscines ou de terrasses par exemple. Par conséquent, cette expérience nous permet de tester l'efficacité de notre méthode en situation réelle. La figure 5.9(b) montre les pavillons extraits par la règle de composition apprise par notre méthode et les résultats sont évalués dans le tableau 5.5. Les pavillons sont extraits avec une F-mesure de 0.572. Ce chiffre est expliqué par le grand nombre de faux positifs (pavillons reconnus comme tel en dehors de tout îlot résidentiel pour la plupart) et donc par la faible précision. Dans l'ensemble les pavillons appartenant aux îlots résidentiels sont bien reconnus, malgré le biais identifié dans l'expérience sur Cronenbourg (*composant de pavillon* mal identifié).

	Précision	Rappel	F-Mesure	Indice de Jaccard
Cronenbourg	0.733	0.760	0.746	0.595
Quinze	0.519	0.636	0.572	0.400

TABLE 5.5 – Évaluation des résultats sur les images PLEIADES.

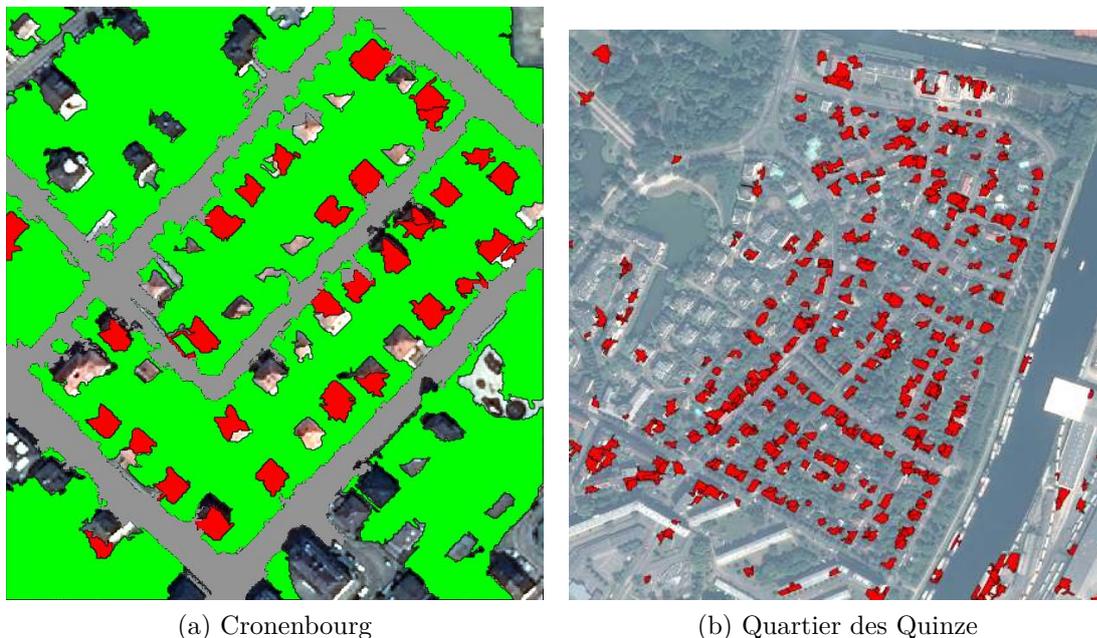


FIGURE 5.9 – Résultats sur les deux images PLEIADES. Les résultats sur *Cronenbourg* sont bons, mais la précision est faible pour le *quartier des Quinze*. Ceci s'explique par le grand nombre de faux positifs en dehors de tout îlot résidentiel. En rouge les pavillons extraits. En vert et noir sur l'image de Cronenbourg, la végétation et la route.

Les résultats présentés dans cette section montrent que la méthode proposée permet d'appréhender une règle de composition capable de reconnaître des pavillons dans des images représentant des paysages urbains. Les résultats sur *Cronenbourg* sont très encourageants, car ils illustrent le bon déroulement de l'algorithme dans un cas *favorable* (*i.e.* classe homogène, objets bien séparés). Cependant, les résultats ne sont pas aussi bons sur le *quartier des Quinze*. Dans une image hétérogène telle que celle-ci, le choix des exemples est crucial. L'expert doit être sûr que les exemples qu'il choisit soient représentatifs de l'ensemble de la classe *pavillon*. De plus, l'image étant plus grande elle contient d'autres objets qui ne sont ni pavillon, ni végétation, ni route. Ceci contribue à l'augmentation du nombre de faux positifs.

En regardant en détail les hiérarchies de concepts générées par la méthode *ARC* pour les différentes expériences, nous nous apercevons que les propriétés spatiales choisies n'ont pas eu d'impact lors de l'apprentissage automatique de règles : les relations d'adjacence ne permettent pas d'identifier différents types de composant de pavillon. L'ajout d'information contextuelle n'a pas été pertinent dans ces expériences. Ceci est illustré dans la figure 5.10.

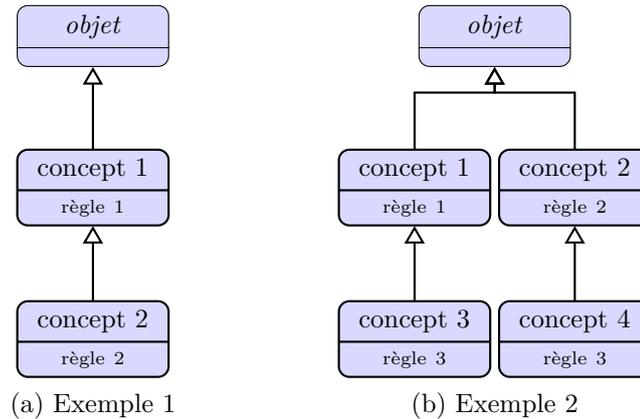


FIGURE 5.10 – Deux exemples de hiérarchies de concepts appris par la méthode *ARC*. Dans les deux cas, l'apprentissage sur les relations spatiales n'a pas permis de différencier les concepts appris dans l'étape précédente.

Conclusion et contributions

Dans ce chapitre nous avons proposé et présenté un algorithme d'apprentissage de règles de composition qui a pour but de faire le lien entre la représentation de la connaissance experte et l'image. La méthode *ARC* apprend automatiquement des règles floues à partir d'exemples et à l'aide de la méthode *ARF_{CAH}* et génère des concepts qui permettent de classer les segments dans le but de pouvoir les regrouper à l'aide d'une règle de composition unique. La méthode a été testée sur une image *QUICKBIRD* ainsi que sur deux extraits d'une image *PLEIADES*.

Les résultats obtenus montrent que la méthode est capable d'apprendre une règle permettant l'extraction des pavillons dans des images représentant des quartiers résidentiels. Pour cela, la qualité des exemples est importante, et ceux-ci doivent être représentatifs de l'ensemble des instances de la classe à apprendre. Cependant malgré des exemples représentatifs, les F-mesures obtenues lors des différentes expériences ne sont pas à la hauteur de nos espérances. Beaucoup de faux positifs sont retenus (ceci est particulièrement vrai dans l'expérience sur le quartier des Quinze). Ceci s'explique par le fait que les propriétés spatiales ne sont pas retenues par l'algorithme *ARF_{CAH}* car elles ne permettent pas d'identifier différents composants de la classe *pavillon*. La relation d'adjacence seule ne permet pas d'apporter suffisamment d'information contextuelle.

Pour remédier à cela et tirer pleinement parti de l'information contextuelle à différents niveaux, nous proposons d'utiliser la méthode *ARC* en tant que première étape d'une approche montante-descendante dans laquelle des objets composés de plus haut niveau sémantique sont extraits et permettent de mettre à jour la classification des objets aux niveaux inférieurs. Cette approche permettra de prendre en compte à la fois les relations spatiales et les relations de composition entre objets de différents niveaux (*i.e.* îlot urbain). Cette méthode est présentée dans le chapitre 6.

Valorisation scientifique : Les travaux présentés dans ce chapitre ont donné lieu à une publication lors de la conférence *IGARSS 2013* (Belarte *et al.*, 2013a) ainsi qu'une publication à l'atelier fouille de données complexes de la conférence *EGC 2013* (Belarte *et al.*, 2013b). Cette dernière a été suivie par une publication dans le numéro spécial de la revue francophone *RNTI*

(à paraître).

Réalisation logicielle : Pour réaliser les travaux présentés dans ce chapitre, la bibliothèque logicielle développée pour les travaux présentés dans le chapitre 4 a été étendue pour gérer l'apprentissage de règles de composition et la mise à jour de la base de connaissance.

6

Approche montante-descendante pour la mise à jour de la classification

Sommaire

1	Injecter de la connaissance dans une étape de post-traitement . . .	99
2	Proposition : approche montante-descendante	100
2.1	Phase montante	101
2.2	Phase descendante	101
2.3	Combiner la phase montante et la phase descendante	102
3	Expérimentations sur des images Pléiades	102
3.1	Protocole expérimental	103
3.2	Cronenbourg	103
3.3	Quartier des Quinze	104
	Conclusion et contributions	104

Dans le chapitre précédent nous avons proposé une méthode pour classer les régions d'une segmentation et extraire des objets composés. Les résultats montrent que la méthode est efficace. Cependant, l'information contextuelle n'est que peu prise en compte lors de l'apprentissage. Dans ce chapitre nous nous concentrons sur cette information contextuelle. Dans un premier temps, nous cherchons à identifier les relations de composition à plusieurs niveaux. Dans un second temps, nous utiliserons les objets composés extraits pour mettre à jour la classification aux différents niveaux. La méthode que nous proposons, l'*approche montante-descendante*, se place en post-traitement de la méthode *ARC*. L'objectif de cette approche est d'utiliser l'information contextuelle (relation *compose* et *est composé de*) pour mettre à jour la classification des pavillons en sortie de la méthode *ARC*.

1 Injecter de la connaissance dans une étape de post-traitement

Dans le chapitre 4 nous proposons une méthode pour extraire les objets composés dans une image segmentée. Or nous avons montré que l'extraction n'était pas possible puisque les segments ne correspondaient pas aux objets d'intérêt pour le géographe. Dans le chapitre 5 nous avons

proposé la méthode *ARC*, qui est capable de faire le lien entre segments et objets d'intérêt. Nous disposons maintenant des outils nécessaires pour l'extraction d'objets composés de plus haut niveau (*i.e.* îlot résidentiel).

Nous avons aussi vu dans le chapitre 5 que les relations spatiales n'ont pas apporté d'informations significatives lors du processus d'apprentissage de la méthode *ARC*. Cependant nous pensons que ce type d'informations contextuelles peut beaucoup apporter dans un processus d'analyse à plusieurs niveaux. Il nous semble donc pertinent d'utiliser ce type d'information dans un post traitement de la méthode *ARC*.

Nous proposons d'utiliser l'information contextuelle pour mettre à jour la classification. Cette méthode, que nous appelons *approche montante-descendante*, est présentée dans le reste de ce chapitre. Ce processus prend en entrée la sortie de la méthode *ARC*. La chaîne de traitement dans son intégralité est présentée dans la figure 6.1. Nous pouvons voir sur cette figure le lien entre la méthode *ARC* et l'approche montante-descendante : en plus de travailler sur le même graphe, l'approche montante-descendante utilise la base de connaissances mise à jour par la méthode *ARC*.

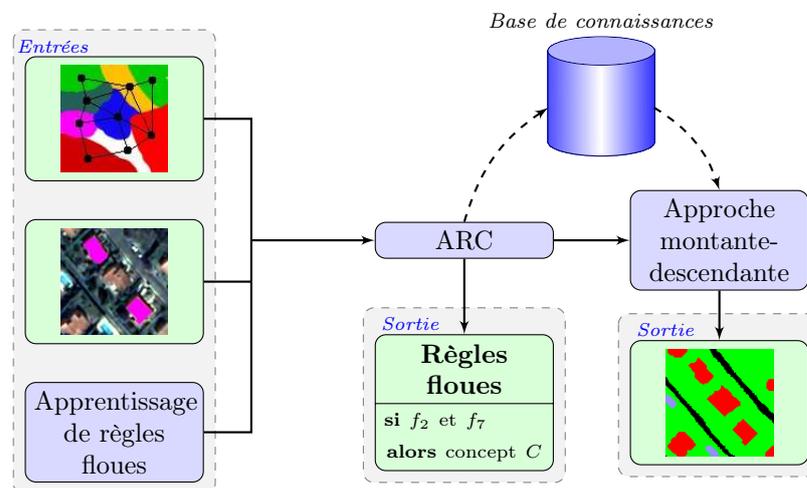


FIGURE 6.1 – Entrées et sorties de la chaîne de traitement. L'approche montante-descendante prend place après la méthode *ARC* dans le but d'améliorer la classification des objets d'intérêt après la phase d'apprentissage.

2 Proposition : approche montante-descendante

L'approche montante-descendante se compose de deux phases : la phase montante et la phase descendante. Dans la phase montante, des règles de compositions (appprises ou renseignées par l'expert) sont utilisées pour extraire des objets composés. Ceci permet d'ajouter au graphe des informations contextuelles. Dans la phase descendante, l'information contextuelle est utilisée pour mettre à jour la classification aux niveaux inférieurs. La figure 6.2 illustre l'approche montante-descendante et ses deux phases.

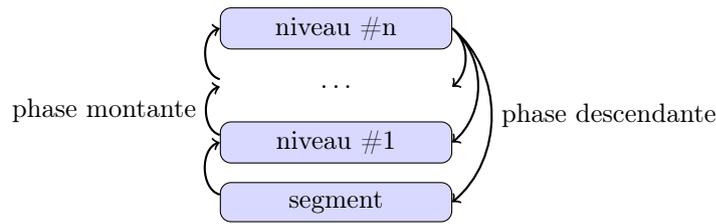


FIGURE 6.2 – L’approche montante-descendante. Après classification des segments, des règles de compositions sont utilisées pour extraire des objets de plus hauts niveaux (phase montante). Cette nouvelle information contextuelle est utilisée pour mettre à jour la classification aux niveaux inférieurs (phase descendante).

2.1 Phase montante

Dans la phase montante nous partons du plus bas niveau (*i.e.* la segmentation) et nous ajoutons des niveaux en fusionnant les objets des couches précédentes. L’intérêt de cette fusion est d’apporter une nouvelle information contextuelle, absente dans une analyse à un seul niveau. L’ajout de niveaux se fait par une utilisation itérative de l’algorithme d’extraction des objets composés présenté dans la section 4 du chapitre 4. Pour cette étape, les règles de composition utilisées peuvent être fournies par l’expert ou apprises par la méthode *ARC*. Le déroulement de la phase montante est donné dans l’algorithme 12.

Algorithme 12: Phase montante

Entrées : liste \rightarrow liste triée de concepts composés à extraire
pour tous les *concepts* $c \in$ *liste* **faire**
 | extraire les objets composés (algorithme 5)

L’ordre des concepts composés dans la liste est importante. En effet, la liste est parcourue dans l’ordre. Il faut donc veiller à ce que les concepts de niveau k soient placés avant les concepts de niveau $k + 1$ dans la liste pour qu’ils puissent être extraits correctement.

2.2 Phase descendante

Dans la phase descendante, nous partons du plus haut niveau et nous descendons pour mettre à jour la classification aux niveaux inférieurs. Cette étape est réalisée à l’aide le l’algorithme 4 présenté dans le chapitre 4, section 3. Outre l’amélioration de la classification de la segmentation, la classification des niveaux intermédiaires est aussi améliorée. Ceci permet une meilleure analyse globale de la scène. Le déroulement de la phase descendante est présenté dans l’algorithme 13.

Algorithme 13: Phase descendante

pour tous les *objets* o *dans le graphe* **faire**
 | mettre à jour la classification (algorithme 4)

La phase descendante repose sur l'utilisation de l'information contextuelle apportée par la phase montante : une fois que des objets de hauts niveaux sont extraits, il est possible d'utiliser l'appartenance ou l'adjacence à ces objets. Par exemple, nous pouvons utiliser la relation *compose un objet de niveau k* pour mettre à jour la classification d'objets au niveau $k-1$. Ainsi les concepts utilisés ici tirent pleinement parti de l'information contextuelle apportée par l'expert.

2.3 Combiner la phase montante et la phase descendante

Le principe de l'approche montante-descendante repose sur la succession de phases montantes et de phases descendantes. Plusieurs itérations peuvent avoir lieu, chaque itération permettant de mettre à jour la classification aux différents niveaux. Le nombre d'itérations est spécifié par l'expert. Chaque itération a besoin d'une liste de concepts composés à extraire lors de la phase montante. Le déroulement de l'approche montante-descendante est présenté dans l'algorithme 14.

Algorithme 14: Approche montante-descendante

Entrées : liste \rightarrow liste de listes de concepts composés à extraire lors des phases montantes
pour tous les itérations i faire
 réaliser la phase montante avec la liste $liste[i]$ (algorithme 12)
 réaliser la phase descendante (algorithme 13)

3 Expérimentations sur des images Pléiades

Les expérimentations présentées dans ce chapitre font suite aux expériences sur les images PLEIADES présentées dans le chapitre 5 section 3.1.

Dans le cas de cette étude, nous choisissons d'effectuer deux itérations. Dans la première itération les pavillons sont extraits, puis les îlots résidentiels. Cette information est ensuite utilisée pour identifier les composants de pavillon qui n'ont pas été extraits correctement. Dans la seconde itération les pavillons manquant sont extraits à partir des candidats identifiés au cours de la première itération. Les îlots sont ensuite à nouveau extraits pour prendre en compte les nouveaux pavillons. L'appartenance à un îlot résidentiel est ensuite utilisée pour mettre à jour la classification des pavillons et éliminer les faux positifs. La figure 6.3 illustre les trois niveaux utilisés pour cette étude.

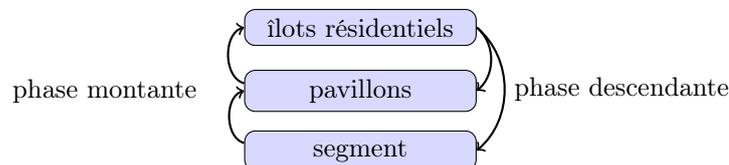


FIGURE 6.3 – L'approche montante-descendante avec les trois niveaux utilisés dans le cas de cette étude.

3.1 Protocole expérimental

L'approche montante-descendante est appliquée directement en sortie de la méthode *ARC*. Deux itérations sont effectuées. Dans les deux cas la phase montante se fait en deux étapes : extraction des pavillons puis extraction des îlots.

Nous utilisons pour la phase montante un modèle à trois niveaux : la segmentation, le niveau des pavillons, le niveau des îlots urbains. La segmentation est le niveau initial. Les pavillons sont extraits à partir du graphe étiqueté, à l'aide de la règle de composition apprise par la méthode *ARC*. Les îlots urbains sont extraits du graphe après extraction des pavillons, à l'aide de la règle de composition $R_{\text{îlot}}$ renseignée par l'expert. Nous utilisons pour la phase descendante l'appartenance ou l'adjacence à un îlot résidentiel (règle $R_{\text{composant}}$) lors de la première itération et l'appartenance à un îlot (règle R_{pavillon}) lors de la deuxième itération pour mettre à jour la classification au niveau segmentation et au niveau pavillons. Les différentes phases sont résumées dans le tableau 6.1. Les trois règles renseignées par l'expert sont présentées dans le tableau 6.2.

itération	phase	concept recherché	règle utilisée
1	montante	pavillon	apprise par <i>ARC</i>
	montante	îlot résidentiel	$R_{\text{îlot}}$
	descendante	composant de pavillon	$R_{\text{composant}}$
2	montante	pavillon	apprise par <i>ARC</i>
	montante	îlot résidentiel	$R_{\text{îlot}}$
	descendante	pavillon	R_{pavillon}

TABLE 6.1 – Les différentes phases des deux itérations successives de l'approche montante-descendante.

$R_{\text{îlot}} = \text{composé d'un tiers de pavillon et composé de deux tiers de végétation et grand}$
 $R_{\text{composant}} = \text{entouré d'îlot résidentiel ou (entouré d'îlot résidentiel et de route)}$
 $R_{\text{pavillon}} = \text{appartient à îlot résidentiel}$

TABLE 6.2 – Les trois règles de composition utilisées pour l'approche montante-descendante.

Dans la suite de ce chapitre, la méthode *ARC* est représentée par l'étape 1 et les deux itérations de l'approche montante-descendante sont représentées par les étapes 2.a et étape 2.b. Les tableaux 6.3 et 6.4 rappellent les résultats obtenus par la méthode *ARC* dans le chapitre 5.

3.2 Cronenbourg

Les résultats de cette expérience sont présentés dans le tableau 6.3. Sur cette image la méthode *ARC* retournait des bons résultats (F-mesure de 0.756). Après la première itération de l'approche montante-descendante les résultats sont meilleurs : la précision et le rappel augmentent. L'augmentation significative du rappel (+0.101) montre que la première itération a permis d'extraire des pavillons qui ne l'étaient pas précédemment. Ceci est illustré dans la figure 6.4 : la figure 6.4(a) montre le résultat après la méthode *ARC*, la figure 6.4(b) montre le résultat de la première phase montante, les deux îlots extraits sont représentés en bleu. La figure 6.4(c) montre le résultat après la première phase descendante, les pavillons extraits sont représentés en rose. Après la seconde itération, le rappel stagne, mais la précision augmente. Ceci est expliqué par l'élimination des faux positifs : deux pavillons identifiés en dehors des deux îlots résidentiels sont éliminés par cette itération. La figure 6.4(d) représente l'évaluation de la méthode après les deux

itérations. En orange les vrais positifs, en rouge les faux positifs, en jaune les faux négatifs. Au final, la F-mesure a augmenté de 0.077 soit une augmentation de plus de 7%.

	Précision	Rappel	F-mesure	Indice de Jaccard
Étape 1	0.733	0.760	0.746	0.595
Étape 2.a	0.757	0.861	0.805	0.674
Étape 2.b	0.789	0.861	0.823	0.700

TABLE 6.3 – Résultats de l'extraction des pavillons sur l'image de *Cronenbourg*. L'étape 1 montre le résultat initial, après la méthode *ARC*. L'étape 2.a montre le résultat après la première itération (identification de nouveaux pavillons). L'étape 2.b montre le résultat après la seconde itération (élimination des faux positifs).

3.3 Quartier des Quinze

Les résultats de cette expérience sont présentés dans le tableau 6.4. Les résultats obtenus par la méthode *ARC* sur cette image étaient moins bons (F-mesure de 0.572.). Ceci est dû à une faible précision causée par un grand nombre de faux positifs, comme nous pouvons le voir sur la figure 6.5(a). Après la première itération de l'approche montante-descendante la F-mesure augmente légèrement (+0.028). Cependant le rappel augmente de façon significative (+0.154). La faible augmentation de la F-mesure est expliquée par une baisse de la précision. Plus de pavillons sont reconnus, le nombre de faux positifs augmente en conséquent. Après la deuxième itération, la F-mesure augmente de 0.081 pour atteindre une valeur de 0.681. La précision augmente significativement (+0.131). Ceci s'explique par l'élimination de tous les faux positifs en dehors des îlots résidentiels. Au final la F-mesure a augmenté de 0.109 soit une augmentation de presque 11%.

	Précision	Rappel	F-mesure	Indice de Jaccard
Étape 1	0.519	0.636	0.572	0.400
Étape 2.a	0.483	0.790	0.600	0.428
Étape 2.b	0.614	0.763	0.681	0.516

TABLE 6.4 – Résultats de l'extraction des pavillons sur l'image du *quartier des Quinze*. L'étape 1 montre le résultat initial, après la méthode *ARC*. L'étape 2.a montre le résultat après la première itération (identification de nouveaux pavillons). L'étape 2.b montre le résultat après la seconde itération (élimination des faux positifs).

Conclusion et contributions

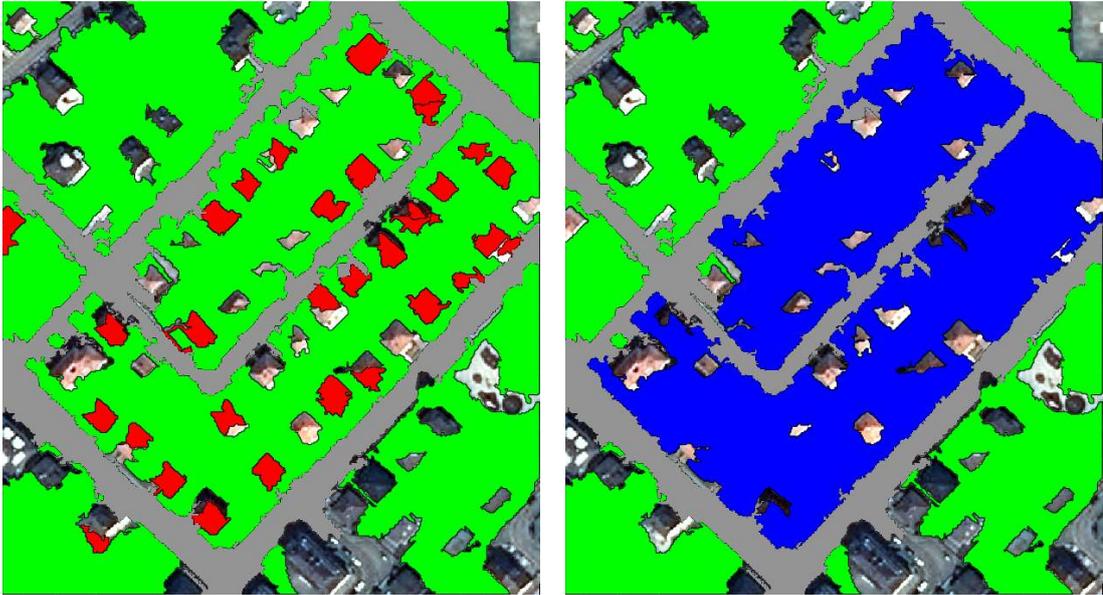
Dans ce chapitre nous avons proposé une méthodologie pour la découverte et l'utilisation d'informations contextuelles lors de l'analyse d'une image de télédétection guidée par la connaissance du domaine. L'approche montante-descendante que nous proposons part d'une classification et utilise des règles de composition pour extraire l'information contextuelle avant d'utiliser cette même information pour mettre à jour la classification. Notre approche travaille sur plusieurs niveaux. Le premier niveau est celui de la segmentation, et autant de niveaux que nécessaire sont ajoutés, selon les besoins de l'expert.

Les expériences réalisées dans ce chapitre montrent une nette amélioration de la classification. Cette amélioration dépasse les 10% pour l'image du *quartier des Quinze*. Ceci montre que dans un cas d'utilisation normal (image hétérogène, de grande taille) notre méthode est capable d'améliorer une classification préalable. Pour effectuer la classification, nous avons utilisé la méthode *ARC* développée dans le chapitre 5. Nous avons ainsi créé une chaîne de traitement, capable de faire le lien entre la segmentation et les objets d'intérêt, et capable d'utiliser l'information contextuelle pour améliorer la classification.

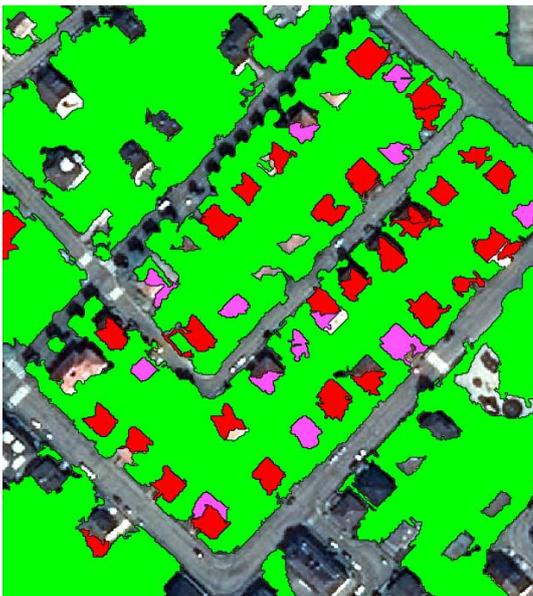
Dans le chapitre suivant, nous utiliserons la méthode *ARC* et l'approche montante-descendante pour valider sur un cas réel et concret notre approche.

Valorisation scientifique : La méthode montante-descendante présentée dans ce chapitre a donné lieu à une soumission dans la revue CEUS.

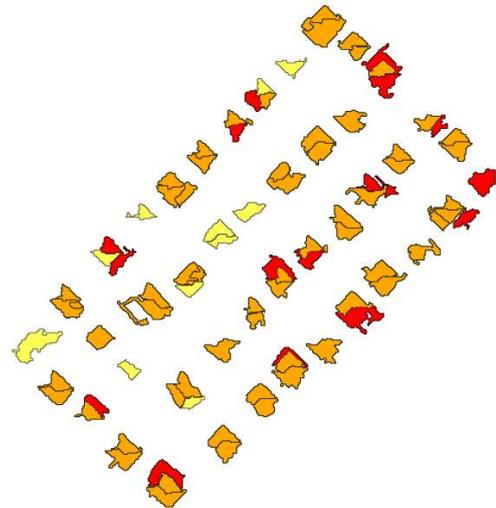
Réalisation logicielle : La bibliothèque développée pour le chapitre 4 et étendue dans le chapitre 5 a à nouveau été étendue pour gérer l'approche montante-descendante. L'application permet la mise à jour de la base de connaissance lors de l'exécution.



(a) Résultats après l'apprentissage. En rouge les pavillons, en vert la végétation, en noir les routes.
 (b) Extraction des îlots résidentiels : approche montante de l'étape 2.a.



(c) Résultats après l'étape 2.a. En rose les pavillons identifiés par cette étape.

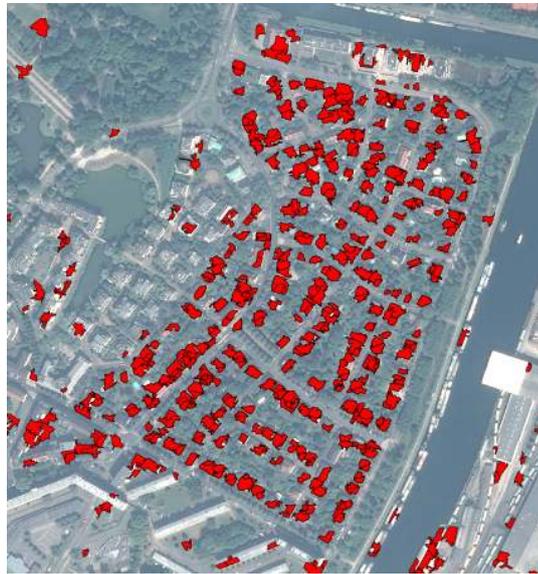


(d) Évaluation du résultat final : en orange les vrais positifs, en rouge les faux positifs, en jaune les faux négatifs.

FIGURE 6.4 – Résultats obtenus après les différentes étapes de l'approche montante-descendante pour l'image de *Cronenbourg* : 6.4(a) après l'apprentissage, 6.4(b) après la première approche montante, 6.4(c) après la première approche descendante. La figure 6.4(d) montre l'évaluation de l'intégralité du processus après deux itérations de l'approche montante-descendante.



(a) Résultats après apprentissage



(b) Résultats après l'étape 2.a



(c) Résultats après l'étape 2.b



(d) Évaluation du résultat final : en orange les vrais positifs, en rouge les faux positifs, en jaune les faux négatifs.

FIGURE 6.5 – Résultats obtenus après les différentes étapes de l'approche montante-descendante pour l'image du *quartier des Quinze*.

7

Validation expérimentale

Sommaire

1	Présentation du jeu de données	109
2	Évaluation des paramètres de la méthode <i>ARC</i>	110
2.1	Taux d'erreur pour l'extraction des exemples	110
2.2	Évaluation des paramètres de la méthode <i>ARC</i>	112
2.3	Étude sur les propriétés	113
2.4	Méthode <i>ARC</i> : résultats sur la région d'intérêt	115
3	Approche montante-descendante	115
4	Application sur l'image de Strasbourg	117
5	Complexité algorithmique et temps de calcul	117
	Conclusion	118

Dans les chapitres précédents, nous avons proposé une méthode pour faire le lien entre la segmentation et la connaissance de l'expert, ainsi qu'une méthode intégrant l'information contextuelle multi niveaux pour mettre à jour une classification. Nous avons validé ces méthodes sur des segmentations en essayant de trouver les meilleurs paramètres (de segmentation, de la méthode *ARC*, ...) de manière empirique. Dans ce chapitre, nous présentons une suite d'expérimentations visant à analyser méthodiquement chacun des paramètres entrant en jeu dans le déroulement de la chaîne de traitement.

Dans un premier temps, nous évaluons les différents paramètres propre à la méthode *ARC* : taux d'erreur pour l'extraction des segments-exemples, paramètre d'affinage pour la génération de la règle de composition, valeurs seuils du critère de Ward pour l'apprentissage des fonctions d'appartenance et des règles.

Dans un second temps, nous évaluons l'approche montante-descendante. Nous utilisons la base de règles présentée dans le chapitre 6.

1 Présentation du jeu de données

L'ensemble des jeux de données utilisés dans ce chapitre sont présentés dans l'annexe A. L'image sur laquelle nous testons notre méthode est une image PLEIADES du nord-est de Strasbourg, d'une résolution de 9211×11275 pixels. Cette image est accompagnée de deux segmentations, présentées dans l'annexe B.

Pour évaluer les paramètres, nous extrayons une région d'intérêt de cette image. Cette région d'intérêt a une taille de 1701×1782 pixels et une résolution de 0.5 mètre et correspond à une zone proche de l'image du *quartier des Quinze* utilisée dans les chapitres 5 et 6. Nous disposons ainsi d'une vérité terrain pour évaluer l'extraction des pavillons et d'un point de comparaison pour estimer l'efficacité des différentes étapes de la chaîne de traitement. La région d'intérêt est présentée dans la figure A.5 dans l'annexe A. Nous testons notre méthode sur la segmentation de travail.

2 Évaluation des paramètres de la méthode *ARC*

Dans cette section, nous évaluons les différents paramètres propres à la méthode *ARC*. Les paramètres sont fixés à leurs valeurs maximales, et chaque expérience va travailler sur un ou plusieurs paramètres. Les valeurs initiales sont présentées dans le tableau 7.1. Le taux d'erreur de 0 signifie qu'un segment doit correspondre parfaitement à un exemple. Le paramètre d'affinage de 1 signifie que les fonctions gaussiennes générées lors de la génération de la règle de composition ont un écart-type non altéré. Les valeurs de 1 pour les critères de Ward signifient que les apprentissages ne généreront qu'une seule classe.

Dans ces expériences, les propriétés utilisées pour l'apprentissage des règles sont uniquement les attributs radiométriques et structurels, les relations spatiales ne sont pas utilisées. Les attributs radiométriques sont : moyenne de chaque bande (proche infrarouge, rouge, vert), NDVI, CI. Les attributs structurels sont : aire, périmètre, longueur, largeur, élongation, indice de Miller, indice de Morton.

Paramètre	Valeur initiale
Taux d'erreur	0
Paramètre d'affinage	1
Critère de Ward pour l'apprentissage des fonctions	1
Critère de Ward pour l'apprentissage des règles	1

TABLE 7.1 – Présentation des valeurs initiales des différents paramètres de la méthode *ARC*.

2.1 Taux d'erreur pour l'extraction des exemples

Le premier paramètre entrant en jeu lors de l'apprentissage est le taux d'erreur dans le processus d'extraction des exemples (voir algorithme 6 dans le chapitre 5). Ainsi, nous cherchons à trouver la valeur minimale de l'erreur qui maximise l'identification correcte de chacun des exemples.

Nous disposons de 17 exemples fournis par l'expert, ceux-ci sont visibles dans la figure A.5 dans l'annexe A. Nous cherchons à trouver un taux d'erreur qui permette d'en extraire le plus possible. Pour cela, nous utilisons la méthode *ARC* réglée avec les valeurs initiales du paramètre d'affinage et des critères de Ward. Dans cette expérience, nous testons individuellement chaque exemple indépendamment avec plusieurs valeurs du taux d'erreur (de 0.1 à 0.9 par pas de 0.1). L'objectif est de trouver une valeur du taux d'erreur qui permette d'extraire chacun des exemples. Les résultats sont donnés dans la figure 7.1.

Dans l'histogramme de la figure 7.1(a), les barres représentent la F-mesure moyenne pour chaque exemple (moyenne en fonction du taux d'erreur). Nous pouvons voir que certains exemples sont extraits plus fréquemment (les exemples 1, 2, 11, 15 et 17). Dans l'histogramme de la figure 7.1(b), les barres représentent la F-mesure moyenne pour chaque valeur du taux d'erreur

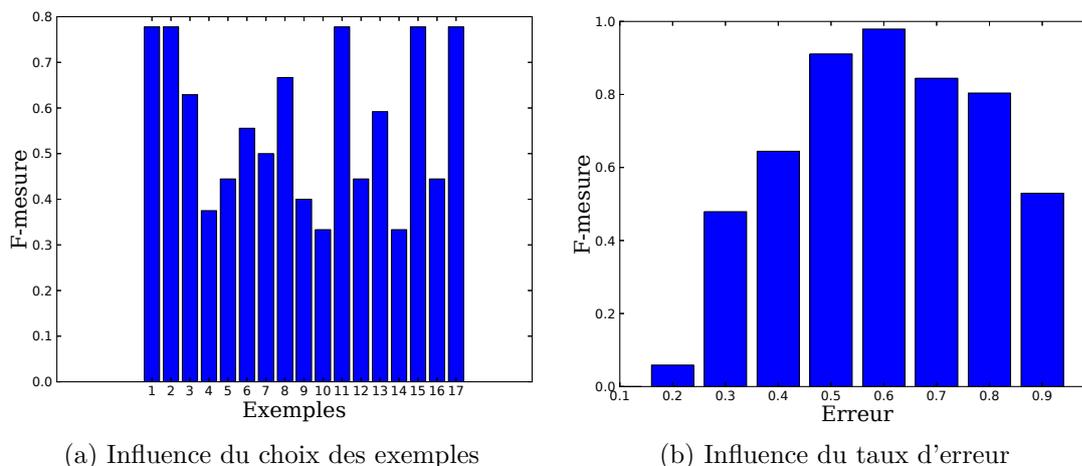


FIGURE 7.1 – Influence du taux d'erreur et des exemples choisis. La figure 7.1(a) montre l'influence du choix des exemples. La figure 7.1(b) montre que pour un taux d'erreur de 0.6 tous les exemples sont retrouvés indépendamment.

(moyenne en fonction des exemples). Nous pouvons observer un pic pour la valeur 0.6. La F-mesure n'atteint pas 1 car pour un des exemples, la méthode *ARC* a extrait un second pavillon, faisant baisser la précision et donc la F-mesure pour cet exemple. En cela, les résultats sont excellents et permettent de fixer le taux d'erreur à 0.6 pour la suite des expériences. Un exemple est donné dans la figure 7.2. Nous pouvons y voir un exemple tel que fourni par l'expert, les segments correspondants extraits par l'algorithme avec un taux d'erreur de 0.6, et le pavillon correspondant identifié par la méthode *ARC*.

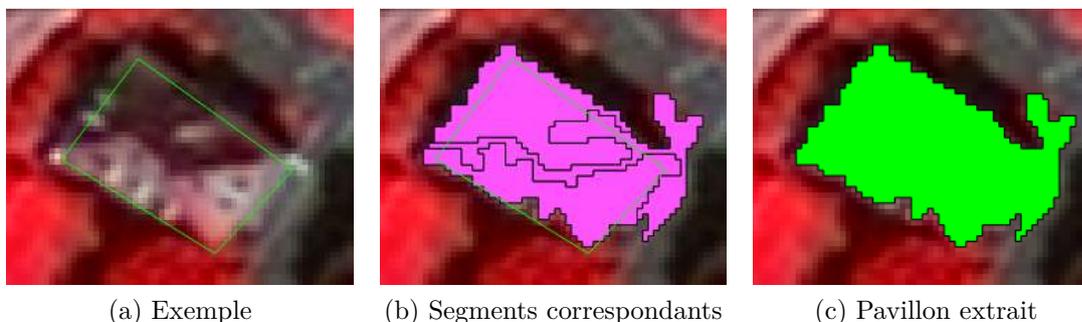


FIGURE 7.2 – Exemple du processus d'extraction des segments. La figure 7.2(a) montre l'exemple tel que renseigné par l'expert. La figure 7.2(b) montre les segments correspondants extraits par l'algorithme d'extraction (pour l'apprentissage et pour l'évaluation). La figure 7.2(c) montre le pavillon extrait par la méthode *ARC*.

Il est à noter que l'algorithme d'extraction des segments utilisé pour extraire les exemples est aussi utilisé pour extraire les segments correspondant à la vérité terrain lors du processus d'éva-

luation. En fixant ainsi le paramètre, nous nous assurons d'extraire correctement les exemples et de pouvoir correctement les évaluer en les comparant aux segments correspondant à la vérité terrain.

2.2 Évaluation des paramètres de la méthode *ARC*

La méthode *ARC* nécessite trois paramètres : le paramètre d'affinage pour la génération de la règle de composition, valeurs seuils du critère de Ward pour l'apprentissage des fonctions d'appartenance et des règles. Nous nous intéressons dans un premier temps uniquement au paramètre d'affinage et dans un second temps aux valeurs du critère de Ward conjointement.

Paramètre d'affinage. Ce paramètre intervient dans la dernière étape de la méthode *ARC* : la génération de la règle de composition, présentée dans l'algorithme 9 du chapitre 5. Le facteur d'affinage permet de moduler l'écart-type calculé à partir des différents exemples. Pour une valeur de 1, l'écart-type n'est pas altéré, pour une valeur inférieure à 1, l'écart-type est réduit et la règle est par conséquent plus restrictive. Pour un écart-type supérieur à 1, l'écart-type est plus grand et par conséquent la règle est moins restrictive.

Pour cette expérience, nous choisissons de tester 10 valeurs du paramètre d'affinage : de 1 à 5.5 par pas de 0.5. Pour chaque valeur, nous effectuons des expérimentations en faisant varier les critères de Ward pour l'apprentissage des fonctions et des règles (100 expériences pour une valeur du paramètre d'affinage, comme expliqué dans le paragraphe suivant). Ceci dans le but d'obtenir un échantillon de résultats pour chaque valeur du paramètre d'affinage.

Les résultats sont présentés dans le tableau 7.2 et dans la figure 7.3. Les résultats présentent pour chaque valeur du paramètre d'affinage le meilleur résultat obtenu ainsi que le résultat moyen. Nous pouvons voir que pour un paramètre d'affinage de 1 les résultats sont faibles. La F-mesure augmente lorsque le paramètre d'affinage augmente, ceci jusqu'à la valeur 3. Ensuite la F-mesure (moyenne et maximum) décroît légèrement.

Paramètre d'affinage	Minimum	Maximum	Moyenne	Écart-type
1	0	0.326	0.175	0.131
1.5	0	0.505	0.293	0.218
2	0	0.539	0.336	0.249
2.5	0	0.573	0.349	0.259
3	0	0.597	0.365	0.272
3.5	0	0.573	0.352	0.262
4	0	0.576	0.353	0.263
4.5	0	0.575	0.346	0.262
5	0	0.562	0.343	0.257
5.5	0	0.560	0.343	0.258

TABLE 7.2 – Résultats de l'analyse sur l'influence du paramètre d'affinage. Pour chaque valeur du paramètre, des expériences sont faites avec plusieurs jeux de paramètres (Ward, voir section suivante).

Ces résultats permettent de fixer le paramètre d'affinage à une valeur de 3, puisque les expériences réalisées dans cette section nous ont montré que c'est avec cette valeur que les résultats sont les meilleurs (en moyenne et dans le meilleur des cas).

Critère de Ward. Pour cette expérience le paramètre d'affinage est fixé à 3. Nous faisons varier le critère de Ward pour l'apprentissage des fonctions d'appartenance et des règles. Les valeurs

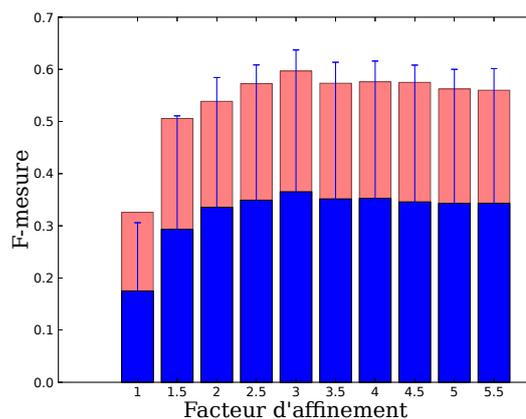


FIGURE 7.3 – Analyse des différentes valeurs du paramètre d’affinage. En bleu, la moyenne et l’écart-type pour chaque valeur. En rouge, le maximum.

vont de 0.1 (signifiant que pour que deux classes soient fusionnées, la perte d’inertie intra-classe ne doit pas dépasser 10%) à 1.0 (signifiant que pour que deux classes soient fusionnées, la perte d’inertie intra-classe ne doit pas dépasser 100%) par pas de 0.1. Pour une valeur de 1.0, il ne reste qu’une seule classe et donc une seule règle est générée. Nous avons donc 10 valeurs du critère de Ward pour l’apprentissage des fonctions d’appartenance, 10 valeurs du critère de Ward pour l’apprentissage des règles, soit en combinant les deux 100 paires de valeurs à tester. Les résultats obtenus sont présentés dans la figure 7.4.

Nous pouvons observer que les résultats sont similaires pour chaque valeur du critère de Ward pour l’apprentissage des règles. La F-mesure est égale à zéro lorsque le critère de Ward pour l’apprentissage des fonctions d’appartenance est inférieur ou égal à 0.4. Nous observons un plateau lorsque ce paramètre atteint une valeur de 0.5. De plus, nous pouvons observer un léger pic pour la paire ($fonction = 0.5; règles = 0.2$). Ceci nous permet de réduire la plage de paramètres à observer pour la suite. Nous pouvons simplement fixer les paramètres en conservant la paire (0.5, 0.2).

2.3 Étude sur les propriétés

Dans cette section nous nous intéressons à l’influence du choix des propriétés sur les résultats de la méthode *ARC*. Nous séparons les propriétés en trois groupes distincts : les attributs radiométriques (notés R dans cette expérience), les attributs structurels (notés F dans cette expérience), et les relations spatiales (notées S dans cette expérience). Nous testons séparément les attributs radiométriques (R) et les attributs structurels (F), puis nous combinons les deux (RF). Nous effectuons une deuxième série de tests en ajoutant les relations spatiales aux trois tests précédents ($R + S$, $F + S$, $RF + S$). Les paramètres d’apprentissage utilisés sont ceux identifiés précédemment dans cette section. Les résultats sont présentés dans le tableau 7.3.

Nous pouvons voir que les attributs structurels seuls (F) ne permettent pas de compléter la méthode *ARC*. Les règles identifiées avec ces seules propriétés ne permettent pas de différencier les composants de pavillons d’autres objets dans l’images. L’ajout de relations spatiales ($F + S$) ne peut rien apporter à des résultats nuls. L’utilisation des attributs radiométriques seuls (R) donne de bons résultats, mais l’ajout des relations spatiales ($R + S$) fait baisser très légèrement les résultats. L’utilisation conjointe des attributs radiométriques et structurels (RF) donne les

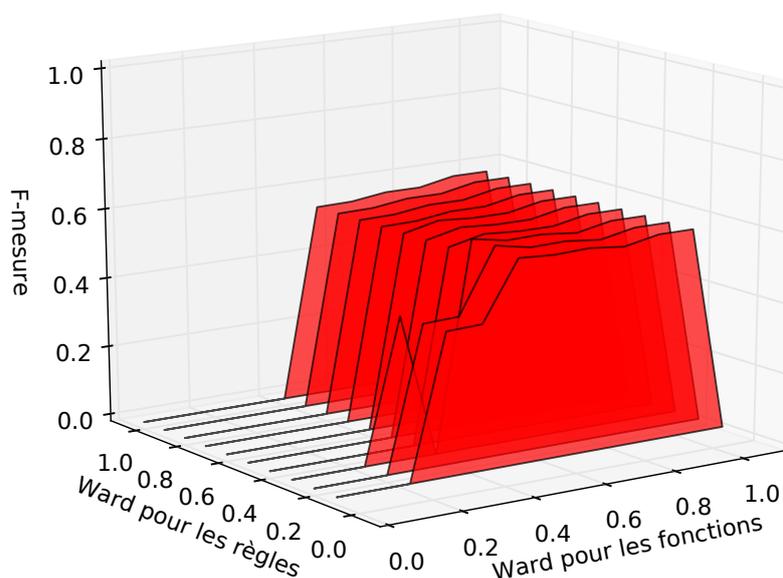


FIGURE 7.4 – Analyse des différentes valeurs du critère de Ward pour l'apprentissage des fonctions d'appartenance et pour l'apprentissage des règles.

Propriétés	Précision	Rappel	F-mesure	Indice de Jaccard
F	0.000	0.000	0.000	0.000
$F + S$	0.000	0.000	0.000	0.000
R	0.545	0.648	0.592	0.421
$R + S$	0.568	0.561	0.565	0.394
RF	0.562	0.635	0.597	0.425
$RF + S$	0.552	0.496	0.523	0.354

TABLE 7.3 – Évaluation des résultats en fonction des différentes propriétés utilisées, parmi des combinaisons entre les attributs radiométriques R , les attributs structurels F , les relations spatiales S .

meilleurs résultats, mais encore une fois l'ajout des relations spatiales ($RF + S$) fait baisser les résultats.

Cette expérience montre que dans le cas de l'analyse de paysages urbains, les critères radiométriques apportent plus d'information que les critères structurels ou les relations spatiales. L'information structurelle apporte une légère amélioration lorsqu'elle est utilisée conjointement à l'information radiométrique. Cependant la conclusion la plus notable est le fait que les résultats baissent légèrement lorsque nous ajoutons l'utilisation des relations spatiales. Après analyse, cela semble logique car dans un paysage de type urbain résidentiel non dense, les pavillons sont systématiquement entourés de végétation, proche de la route et d'autres pavillons. Dans ce sens, les relations spatiales telles que nous les avons utilisées n'apportent que peu d'information à la méthode *ARC* pour l'apprentissage et la classification des composants de pavillons. La relation

d'adjacence seule ne permet pas d'aider à discriminer différents types de composant de pavillon. C'est pour cette raison que nous avons choisi d'utiliser l'approche montante-descendante pour tirer parti de l'information contextuelle multi niveaux.

2.4 Méthode *ARC* : résultats sur la région d'intérêt

Les différents paramètres évalués dans cette section peuvent être fixés. Ceux-ci sont résumés dans le tableau 7.4.

Paramètre	Valeur finale
Taux d'erreur	0.6
Paramètre d'affinage	3
Critère de Ward pour l'apprentissage des fonctions	0.5
Critère de Ward pour l'apprentissage des règles	0.2

TABLE 7.4 – Présentation des valeurs finales des différents paramètres de la méthode *ARC* qui ont été identifiées dans cette section.

Le résultat de la méthode *ARC* pour l'extraction des pavillons sur la région d'intérêt est donné dans la figure 7.5. Les pavillons extraits sont illustrés en vert, pour contraster avec l'image en fausses couleurs. Nous pouvons voir que beaucoup de pavillons sont extraits correctement. Cependant, nous pouvons voir un certain nombre de faux positifs en dehors des îlots résidentiels. Ceci pourra être corrigé par l'utilisation de l'approche montante-descendante, que nous détaillerons dans la section suivante. Un extrait des différentes règles apprises par la méthode *ARC* est donné dans la figure 7.6 (toutes ne sont pas représentées par soucis de place et de lisibilité).

3 Approche montante-descendante

Pour tester l'approche montante-descendante sur la région d'intérêt, nous utilisons le même protocole de test que pour la validation de la méthode dans le chapitre 6. Nous utilisons les résultats obtenus dans la section précédente comme première étape. Les itérations et les règles utilisées sont rappelées dans les tableaux 7.5 et 7.6.

itération	phase	concept recherché	règle utilisée
1	montante	pavillon	apprise par <i>ARC</i>
	montante	îlot résidentiel	$R_{\text{îlot}}$
	descendante	composant de pavillon	$R_{\text{composant}}$
2	montante	pavillon	apprise par <i>ARC</i>
	montante	îlot résidentiel	$R_{\text{îlot}}$
	descendante	pavillon	R_{pavillon}

TABLE 7.5 – Les différentes phases des deux itérations successives de l'approche montante-descendante.

$R_{\text{îlot}}$ =composé d'un tiers de pavillon **et** composé de deux tiers de végétation **et** grand
 $R_{\text{composant}}$ =entouré d'îlot résidentiel **ou** (entouré d'îlot résidentiel **et** de route)
 R_{pavillon} =appartient à îlot résidentiel

TABLE 7.6 – Les trois règles de composition utilisées pour l'approche montante-descendante.

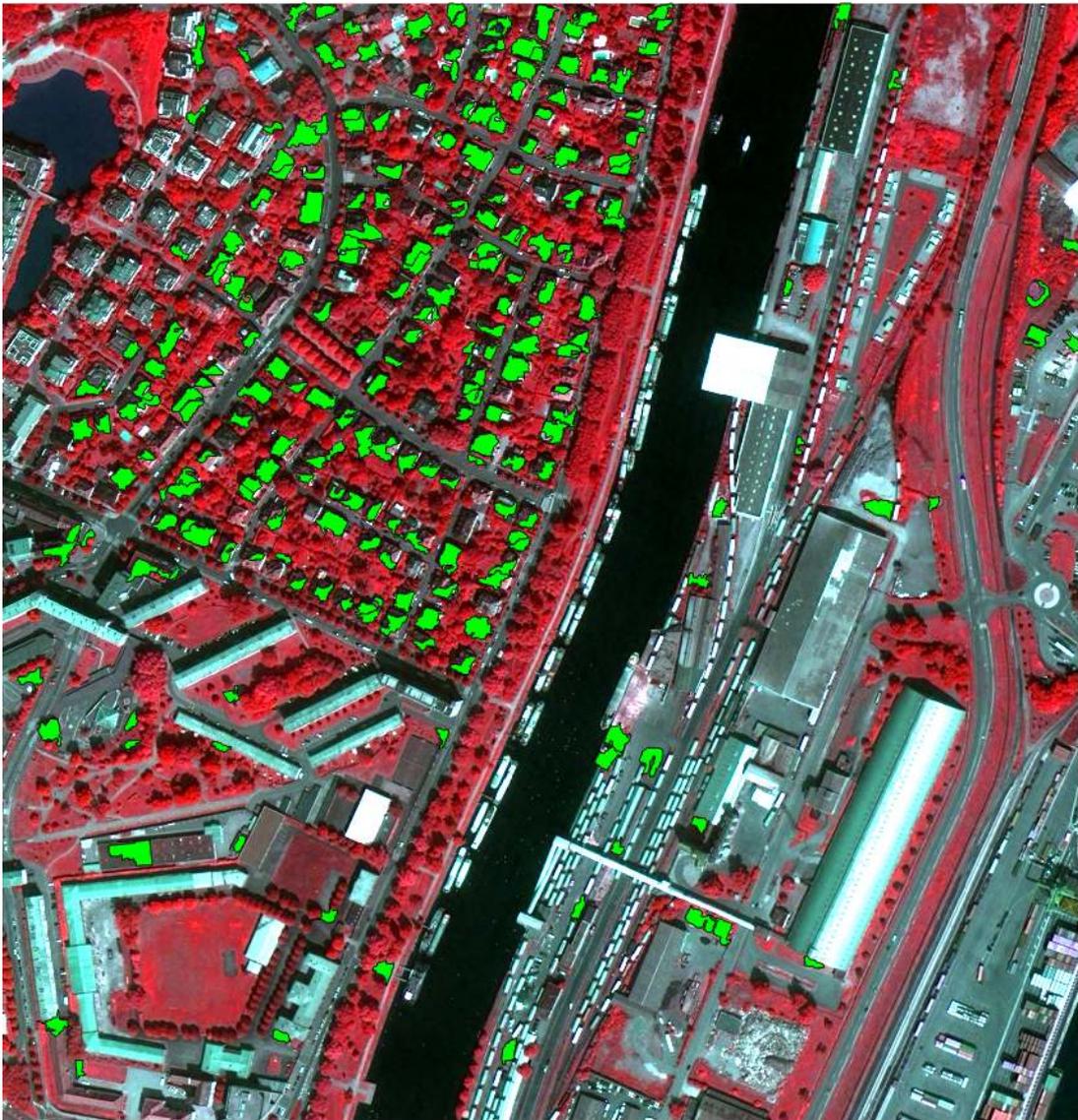


FIGURE 7.5 – Illustration du résultat obtenu avec le jeu de paramètres identifié dans cette section. En vert (pour contraster avec l'image en fausse couleur), les pavillons extraits.

Les résultats obtenus sont présentés dans le tableau 7.7 et illustrés dans la figure 7.7. Nous pouvons voir que la première itération de l'approche montante-descendante permet une légère amélioration de la F-mesure (+ 0.012). La deuxième itération permet une forte augmentation de la précision à travers l'élimination des faux positifs hors des îlots résidentiels (+ 0.118). Cependant le rappel diminue (- 0.064) ce qui au final permet une amélioration non significative de la F-mesure (+ 0.028). La diminution du rappel est due au fait que certains pavillons aux bords d'un îlot ne soient pas rattachés à cet îlot lors de l'extraction des objets composés. Ne faisant pas partie d'un îlot, ces vrais positifs sont éliminés par la deuxième itération de l'approche montante-

descendante. Le fait que certains pavillons ne soient pas rattachés à des îlots est causé par la présence de segments non classés (ni pavillon, ni végétation, ni ombre). Ces segments peuvent correspondre à des terrasses ou allées de garage, qui ne sont pas pris en compte dans la règle de composition $R_{\text{îlot}}$.

	Précision	Rappel	F-mesure	Indice de Jaccard
Étape 1	0.562	0.635	0.597	0.425
Étape 2.a	0.565	0.660	0.609	0.438
Étape 2.b	0.683	0.596	0.637	0.467

TABLE 7.7 – Résultats de l'extraction des pavillons sur l'image du *quartier des Quinze* extrait de l'image PLEIADES en fausses couleurs. L'étape 1 montre le résultat initial, après la méthode *ARC*. L'étape 2.a montre le résultat après la première itération (identification de nouveaux pavillons). L'étape 2.b montre le résultat après la seconde itération (élimination des faux positifs).

4 Application sur l'image de Strasbourg

Nous venons de valider la chaîne de traitement sur une région d'intérêt de la segmentation de test de travail. Nous allons maintenant utiliser la règle apprise par la méthode *ARC* et l'approche montante-descendante sur l'intégralité de la segmentation de travail. Le résultat est présenté dans la figure 7.8. Dans cette figure, la végétation est représentée en vert, les pavillons extraits après la première itération de l'approche montante-descendante sont représentés en rouge et les îlots résidentiels extraits après la seconde itération sont représentés en bleu. Il est important de noter que les îlots résidentiels contiennent bien évidemment des pavillons, mais ceux-ci ne sont pas représentés ici pour ne pas rendre trop confuse l'illustration. Nous ne disposons malheureusement pas d'une vérité terrain de la classe *pavillon* sur l'intégralité de la zone, il nous est donc pas possible d'évaluer numériquement ce résultat.

Nous pouvons voir que les îlots résidentiels du quartier des Quinze (région d'intérêt, en bas au centre) sont à nouveau correctement extraits. La plupart des îlots extraits dans le reste de l'image correspondent bien à des îlots dans l'image. Les faux positifs les plus évidents sont les quelques îlots extraits au sud est, au niveau du port autonome de Strasbourg. Les pavillons représentés correspondent généralement à des habitats individuels. Beaucoup d'îlots ne sont pas extraits. Ceci est dû au fait que la règle de composition utilisée pour l'extraction des îlots est probablement trop spécifique à la région d'intérêt.

Cette expérience nous permet de vérifier deux choses. Premièrement, la portabilité des règles apprises à partir de la région d'intérêt sur l'intégralité de la zone de Strasbourg est vérifiée. La règle de composition pour le concept *pavillon* a permis d'extraire les objets correspondant sur toute la zone. Deuxièmement, notre méthode passe à l'échelle.

5 Complexité algorithmique et temps de calcul

Dans cette section nous étudions la complexité algorithmique de la chaîne de traitement. Les trois étapes importantes sont l'étiquetage des objets (algorithme 4), l'extraction des objets composés (algorithme 5) et l'apprentissage de la règle de composition. Dans la suite de cette section, nous considérons un graphe à n objets.

Étiquetage des objets. Le nombre de concepts présents dans la base de connaissance est faible par rapport au nombre d'objets dans le graphe. En ce sens, l'étiquetage d'un objet a une complexité négligeable et la complexité de l'algorithme 4 est donc facteur du nombre d'objets dans le graphe, soit une complexité moyenne de $O(n)$.

Apprentissage de la règle de composition. Supposons que nous ayons k exemples composés par s segments (avec $k \leq s$) et p propriétés. La complexité moyenne de l'algorithme CAH est $O(n^2 \log(n))$. Au sein de la méthode *ARC*, nous effectuons une CAH par propriété plus une pour apprendre la règle de composition. Ceci donne $p \times O(s^2 \log(s)) + O(k^2 \log(k))$, soit une complexité moyenne de $O(s^2 \log(s))$. La complexité de la méthode *ARC* va être très faible si les exemples sont composés de peu de segments.

Extraction des objets composés. L'extraction des objets composés est basée sur un algorithme de croissance de région appliqué dans une composante connexe du graphe. Cette composante connexe peut être composée d'un seul objet comme elle peut contenir tout le graphe (selon la classe à extraire et l'étiquetage du graphe). L'extraction d'un objet composé se matérialise dans le graphe par un parcours en largeur d'abord, ce qui correspond à une complexité moyenne linéaire de $O(n)$. Or dans une composante connexe, toutes les paires d'objets sont testées en même temps, ce qui nous donne une complexité moyenne quadratique de $O(n^2)$.

À cela il faut ajouter le coût de fusion de deux objets d'intérêts, opération qui intervient à chaque étape de la croissance de région. Celle-ci nécessite de calculer les propriétés structurelles de l'objet fusionné. Certaines sont accessibles directement après fusion des polygones (par exemple taille, périmètre), d'autre nécessite de calculer les vecteurs propres du polygone (par exemple la longueur et la largeur, et les propriétés calculables à partir de celles-ci : élongation, indice de Morton). Or le coût de calcul des vecteurs propres dépend du nombre de points dans le polygone, nombre qui devient très grand lorsque l'objet devient grand. D'autant plus que les polygones représentent directement le contour des segments, avec le crénelage et donc l'augmentation du nombre de point dans le polygone que cela implique. Le calcul des vecteurs propres multiplie le temps de calcul de l'extraction des objets composés par un facteur qui augmente avec la taille des composantes connexes.

Temps de calcul. En pratique, les temps de calcul de l'étiquetage des objets et de l'apprentissage sont négligeables (inférieur à la seconde sur la région d'intérêt). Le temps d'exécution de la chaîne de traitement est directement dépendant de l'extraction des objets composés. En effet, si l'extraction des maisons est réalisable en quelques secondes (beaucoup de très petites composantes connexes) l'extraction des îlots résidentiels est beaucoup plus longue, jusqu'à plusieurs dizaines de minutes sur une machine récente (processeur huit coeurs, huit giga octets de mémoire vive) pour un graphe contenant 200 000 objets à l'étape initiale.

Conclusion

Dans ce chapitre, nous avons validé méthodiquement les différentes étapes de la chaîne de traitement proposée dans les chapitres 5 et 6. Nous avons commencé par évaluer l'influence des paramètres de la méthode *ARC* dans le but de pouvoir les fixer pour les expériences ultérieures. Nous avons ensuite utilisé l'approche montante-descendante sur les résultats obtenus par la méthode *ARC* pour valider la chaîne de traitement sur un exemple concret. Nous avons finalement utilisé la règle de composition apprise par la méthode *ARC* dans la région d'intérêt pour appliquer la chaîne de traitement sur l'intégralité de l'image correspondant à la segmentation de

test. Finalement, nous avons estimé la complexité algorithmique de la chaîne de traitement et identifié une étape critique en terme de temps de calcul.

De ce chapitre, nous pouvons retenir plusieurs choses. Premièrement, nous avons vu dans la section 2.3 que dans notre étude, l'intérêt des relations spatiales pour la caractérisation des composants de pavillons est faible. En effet, les relations d'adjacence seules ne suffisent pas à distinguer plusieurs classes de composant de pavillon. Le manque d'influence des relations spatiales dans la méthode *ARC* est compensé dans l'approche montante-descendante, où l'extraction et l'utilisation de l'information contextuelle permettent de mettre à jour l'étiquetage des objets et d'améliorer les résultats obtenus par la méthode *ARC*.

Deuxièmement, l'utilisation de la chaîne de traitement sur l'intégralité de l'image de test (correspondant à la segmentation de travail du LIVE) nous a permis de vérifier la portabilité de la base de connaissance apprise sur la zone de Strasbourg Nord-Est. Les règles apprises sur la région d'intérêt ont permis l'extraction de pavillons et d'îlots résidentiels. Ceci nous a montré que notre méthode est capable de passer à l'échelle et permet d'analyser une large zone.

Troisièmement, le temps de calcul de la chaîne de traitement est dépendant de l'extraction des objets composés. Si ce temps est court pour l'extraction de petits objets tels que les pavillons, il est bien plus long lorsqu'il s'agit d'extraire des objets plus conséquents tels que les îlots résidentiels.

Nous pouvons cependant noter que les résultats sont perfectibles. L'apprentissage et l'extraction des objets composés est toujours dépendant de la qualité de la segmentation. L'approche montante-descendante permet d'améliorer les résultats mais son bon fonctionnement dépend de l'extraction des objets composés selon des règles fournies par l'expert. Ces règles décrivent la composition d'une classe d'objets dans une segmentation. L'ensemble des segments doivent être correctement étiquetés pour que l'extraction des objets composés fonctionne correctement. Or ceci est difficile dans les paysages urbains car les îlots résidentiels ne sont pas homogènes et leurs composition peut varier d'une zone d'étude à une autre. L'expert se doit d'être attentif à ces différences.

```

<functions>
  <function name="fun00" attribute="b2" shape="trapezoidal"
    minSupport="21.6" maxSupport="214.1" minCore="53.7" maxCore="182.0" />
  <function name="fun01" attribute="b2" shape="trapezoidal"
    minSupport="164.3" maxSupport="277.4" minCore="183.2" maxCore="258.5" />
  <function name="fun02" attribute="b3" shape="trapezoidal"
    minSupport="28.6" maxSupport="205.7" minCore="58.1" maxCore="176.2" />
  <function name="fun03" attribute="b3" shape="trapezoidal"
    minSupport="154.0" maxSupport="288.5" minCore="176.4" maxCore="266.1" />
  <function name="fun04" attribute="b4" shape="trapezoidal"
    minSupport="37.3" maxSupport="160.9" minCore="57.9" maxCore="140.3" />
  <function name="fun05" attribute="b4" shape="trapezoidal"
    minSupport="88.7" maxSupport="281.1" minCore="120.7" maxCore="249.1" />
  <function name="fun06" attribute="ndvi_index" shape="trapezoidal"
    minSupport="-0.09" maxSupport="0.18" minCore="-0.05" maxCore="0.13" />
  ...
  <function name="f00" attribute="size"
    shape="gaussian" mean="307.5" sigma="217.418" />
  <function name="f01" attribute="perimeter"
    shape="gaussian" mean="109" sigma="48.3732" />
  <function name="f02" attribute="miller_index"
    shape="gaussian" mean="0.325239" sigma="0.176509" />
  <function name="f03" composition="ratio" concept="concept00"
    shape="gaussian" mean="0.856911" sigma="0.607491" />
  <function name="f04" composition="ratio" concept="concept01"
    shape="gaussian" mean="0.143089" sigma="0.41578" />
  <function name="f05" attribute="size"
    shape="gaussian" mean="170" sigma="217.418" />
  <function name="f06" attribute="perimeter"
    shape="gaussian" mean="91" sigma="48.3732" />
  <function name="f07" attribute="miller_index"
    shape="gaussian" mean="0.257974" sigma="0.176509" />
  <function name="f08" composition="ratio" concept="concept00"
    shape="gaussian" mean="0.922059" sigma="0.607491" />
  <function name="f09" composition="ratio" concept="concept01"
    shape="gaussian" mean="0.0779412" sigma="0.41578" />
  ...
</functions>
<rules>
  <rule name="rule00"
    definition="fun00_and_fun02_and_fun06_and..." />
  <rule name="rule01"
    definition="fun01_and_fun03_and_fun05_and_fun06..." />
  <rule name="compRule00"
    definition="(f00_and_f01_and_f02_and_f03_and_f04)
    or (f05_and_f06_and_f07_and_f08_and_f09)
    or..."
  </rule>

```

FIGURE 7.6 – Base de règles apprises par la méthode *ARC* sur la région d'intérêt. Deux règles sont apprises pour l'étiquetage des segments. Une règle de composition unique est apprise.



(a) Pavillons extraits après apprentissage



(b) Pavillons extraits après première itération



(c) Îlots extraits après deuxième itération



(d) Pavillons extraits après deuxième itération

FIGURE 7.7 – Résultats des différentes étapes de la chaîne de traitement. Les pavillons sont représentés en vert sur les images 7.7(a), 7.7(b) et 7.7. Les îlots sont représentés en bleu sur l'image 7.7(c).

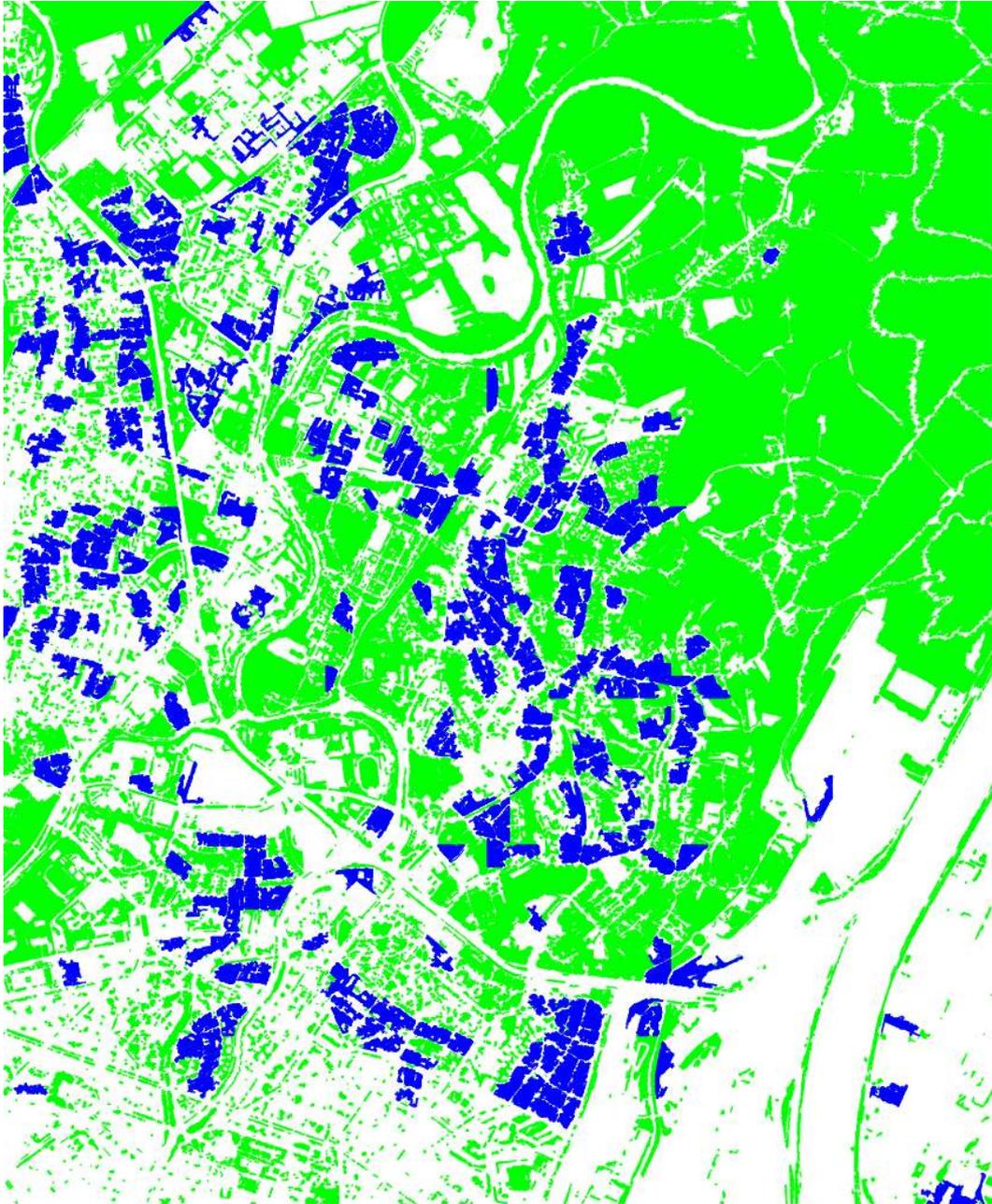


FIGURE 7.8 – Résultats de la chaîne de traitement sur la segmentation de travail. En vert, la végétation. En Bleu, les îlots résidentiels extraits après la deuxième itération.

Conclusion

Au cours de cette thèse, nous avons travaillé à réduire le fossé sémantique séparant la connaissance de l'expert et la représentation des objets d'intérêt dans une image de télédétection à très haute résolution spatiale. Les thèmes abordés dans cette thèse se développent autour de deux axes de recherche. Le premier axe est la recherche d'un lien entre la représentation de la connaissance de l'expert et la segmentation d'une image dans le domaine de la télédétection. Le second axe de recherche est l'intégration de l'information contextuelle dans un processus d'analyse d'images de télédétection. Pour cela, nous avons proposé une approche innovante qui a été concrétisée par le développement d'une chaîne de traitement complète pour l'apprentissage de règles de composition et l'intégration de l'information contextuelle dans un processus d'analyse d'images de télédétection à THRS, le tout basé sur une formalisation de la connaissance à l'aide de la logique floue, pour gérer l'imprécision inhérente au traitement d'images.

Axe 1 : Lien entre la connaissance et la segmentation. Dans un premier temps, nous nous intéressions à la pertinence de l'utilisation de l'information contextuelle en tant que connaissance experte pour l'analyse de segmentation d'images THRS. Cependant, nous avons vu dans le chapitre 4 qu'il est en pratique impossible d'utiliser un raisonnement basé sur la connaissance pour la classification de segments car les segments ne correspondent pas ou peu à des objets d'intérêt. Pour lever ce verrou scientifique, nous avons proposé de changer le niveau d'interprétation et de ne plus considérer les objets d'intérêt simple pour le géographe comme des objets simples (des régions) dans la segmentation mais comme des objets composés. Une classe d'objets composés est ainsi décrite par une règle de composition floue. Cependant, il est difficile pour l'expert de donner une telle règle de composition, car celle-ci dépend de l'image en entrée et de la segmentation utilisée. Pour pallier cette difficulté, nous avons proposé dans le chapitre 5 la méthode *ARC*, un algorithme d'apprentissage de règles de composition indépendant du type d'images et de la segmentation. Nous avons vu que l'apprentissage de règles de composition est efficace, mais que les relations spatiales sélectionnées dans cette analyse n'étaient pas pertinentes dans le processus d'apprentissage.

Axe 2 : Intégration de l'information contextuelle. Les conclusions sur la faible utilité des relations spatiales au sein de la méthode *ARC* nous ont amené à nous concentrer sur la représentation multi-niveau de la connaissance spatiale. Dans un second temps, nous nous sommes intéressés à l'intégration d'information contextuelle dans un post-traitement à la méthode *ARC*. L'approche montante-descendante proposée dans le chapitre 6 est une méthode hiérarchique reposant sur l'extraction d'objets composés de haut niveau d'interprétation à partir de la connaissance de l'expert, et sur l'utilisation de ces relations de composition pour mettre à jour la classification des objets à plus bas niveaux. Nous avons montré qu'en utilisant uniquement l'information contextuelle, nous sommes capable d'améliorer un résultat de classification, là où notre méthode d'apprentissage n'était pas capable de tirer profit de ces informations.

En travaillant conjointement sur ces deux axes de recherche, nous avons identifié un verrou majeur séparant la représentation de l'information dans le domaine de l'image et dans celui de la connaissance experte. Pour lever ce verrou, nous avons proposé une méthodologie pour représenter les objets d'intérêt du domaine par des objets composés dans la segmentation. Cette

méthodologie a été concrétisée par l'implémentation de la méthode *ARC*, méthode qui vise à apprendre les règles de composition lorsque l'expert ne peut pas fournir une telle règle. Les résultats obtenus montrent que la méthode *ARC* permet d'apprendre avec peu d'exemples une règle utile pour extraire une classe d'objets d'intérêt dans l'image de télédétection. Pour améliorer ces résultats, nous avons proposé une approche itérative multi-niveaux pour intégrer l'information contextuelle. La chaîne de traitement implémentant cette méthode permet d'améliorer les résultats en utilisant l'appartenance ou la proximité à un objet de haut niveau pour raffiner l'étiquetage des objets d'intérêt à plus bas niveau. La validation de la chaîne de traitement dans le chapitre 7 montre la complémentarité de la méthode *ARC* et de l'approche montante-descendante pour l'apprentissage et l'utilisation de la connaissance de l'expert. Les résultats obtenus montrent l'efficacité de la chaîne de traitement pour l'analyse d'une image de télédétection à très haute résolution spatiale.

Bilan scientifique

Les travaux présentés ici ont donné lieu à plusieurs contributions dans les domaines de la fouille de données et de la télédétection.

Nos premiers travaux visaient à utiliser la connaissance spatiale dans un processus de classification d'objets issus de la segmentation. Nous avons montré dans le chapitre 4 que cela n'était pas possible en l'état. Ces conclusions ont été publiées dans les actes de l'atelier *fouille de données complexes* de la conférence EGC 2012.

La méthode *ARC* est un algorithme d'apprentissage de règles de composition floues. La partie méthodologique de la méthode a donné lieu à deux contributions dans le domaine de la fouille de données. Premièrement, une publication dans les actes de l'atelier *fouille de données complexes* de la conférence EGC 2013. Deuxièmement, une publication dans un numéro spécial de la revue RNTI. La partie applicative a donné lieu à une publication dans les actes de la conférence IGARSS 2013.

L'approche montante-descendante est une application directe de l'utilisation de connaissances thématiques. Ces travaux ont donné lieu à une soumission dans le domaine de la télédétection dans la revue CEUS.

Développement logiciel

Pour mettre en pratique les méthodes proposées dans cette thèse, nous avons conçu et développé une bibliothèque logicielle regroupant toutes les étapes de la chaîne de traitement. Cette bibliothèque est écrite en C++ et utilise les bibliothèques libres OTB pour le traitement des images, GDAL/OGR¹ pour le traitement des données vectorielles, Boost² pour les pointeurs intelligents et la gestion des graphes, TinyXML³ pour la gestion des fichiers de configuration, et Eigen⁴ pour le calcul matriciel.

L'application repose sur l'utilisation d'un fichier de configuration en langage XML. L'arborescence XML nous sert de langage de description : les différentes étapes et paramètres y sont décrits. L'utilisateur décrit le déroulement de l'algorithme dans le fichier de configuration puis le programme lit le fichier et enchaîne les différents maillons tels que spécifiés dans le fichier. En

1. <http://www.gdal.org/index.html>

2. <http://www.boost.org/>

3. <http://sourceforge.net/projects/tinyxml/>

4. <http://eigen.tuxfamily.org/index.php>

cela, la bibliothèque est hautement paramétrable et l'application ne nécessite que des bases en informatique pour être utilisée.

Perspectives de recherche

Les résultats obtenus au cours de nos travaux nous ont permis d'identifier certaines perspectives de recherche.

L'objectif de cette thèse était l'analyse de paysages urbains résidentiels à partir d'images THRS. Pour cela, nous avons eu besoin d'apprendre une règle de composition pour décrire le concept *pavillon* en terme de composition dans une segmentation. Nous nous sommes concentrés sur un algorithme d'apprentissage de règles floues, l'algorithme *ARFCAH* conçu et développé pour nos travaux. Nous avons vu dans le chapitre 3 que la littérature dans le domaine est riche. Il pourrait être pertinent de tester et d'évaluer la méthode *ARC* avec d'autres méthodes d'apprentissage de règles. Les méthodes d'apprentissage actives (*active learning* en anglais) nous semblent pertinentes. Une règle pourrait être proposée à l'expert, et celui-ci pourrait moduler cette règle en enlevant ou ajoutant des fonctions d'appartenance.

Notre intérêt s'est porté sur les paysages résidentiels. Or ceux-ci sont par nature intrinsèquement complexes, de par leur forme (définie par le réseau routier) et leur composition, elle-même définie par des objets complexes (forme des pavillons, présence ou non de terrasse ou de jardin). Il nous semble pertinent de tester notre méthode sur d'autres types de paysages, voir d'autres types d'images (radar, médicales par exemple). Ceci permettrait de mieux identifier les limites propres à la méthode et celles inhérentes au domaine d'application. D'un point de vue plus générique, il serait intéressant d'étudier le portage de la méthode *ARC* sur des problèmes non liés à l'image. Le fondement de la méthode repose sur l'information contextuelle et les relations entre objets. Ce type de relations, quantifiables ou qualitatives, peut se retrouver dans une base de donnée relationnelle.

Les données sur lesquelles nous avons travaillé sont des images THRS segmentées à simple niveau (voir chapitre 1), combinées à des données issues de SIG, que nous avons traité avec un algorithme de classification hiérarchique. Une piste qui nous semble intéressante est l'intégration de processus multi-sources, que cela se fasse à travers l'ajout de plusieurs sources de données (plusieurs images, plusieurs résolutions) ou par le biais de processus collaboratifs (par exemple combinaison de classifieurs pour l'étiquetage des segments, application de la méthode *ARC* à plusieurs niveaux).

Nos travaux reposent sur l'information contextuelle. Le dictionnaire d'objets urbains du LIVE décrit les relations entre objets en terme de distance, d'adjacence et de composition. Nous avons vu dans le chapitre 2 qu'il existe des méthodes pour formaliser et traiter des relations spatiales plus complexes telles que l'alignement d'objets ou le parallélisme. Ce type de relations pourrait apporter beaucoup de richesse à l'information contextuelle.

Le point clé de l'axe de recherche 1 a été la recherche d'un lien entre les objets d'intérêt tels que décrits par l'expert et la représentation de ces objets dans une image segmentée. Dans cette thèse, nous avons formalisé ce lien par une règle de composition floue. Des travaux sur la formalisation d'une ontologie image, ou d'une collection d'ontologie image, ont récemment débuté dans l'équipe BFO. Ces travaux visent à décrire les mêmes objets d'intérêt en fonction de leur représentation (sémantique dans le domaine d'expertise, représentation à THRS, à moyenne résolution, *et caetera*). Les outils développés dans cette thèse ont leur place dans une telle étude.

Les résultats obtenus dans cette thèse montrent qu'une segmentation considérée comme acceptable par l'expert est encore problématique à utiliser. Une thèse dont l'objectif est d'évaluer

la qualité d'une segmentation a débuté dans l'équipe BFO. Ces travaux pourraient permettre de mieux choisir la segmentation sur laquelle tester les méthode d'analyses d'images THRS telle que la nôtre.

D'un point de vue purement applicatif, il nous semble important d'améliorer les performances de l'algorithme d'extraction des objets composés. Nous avons vu que cette étape est critique en terme de temps de calcul. Un travail de fond est nécessaire pour revoir le déroulement de l'algorithme. De plus, celui-ci pourrait être parallélisé car le traitement de chaque composante connexe est indépendant. Ceci permettrait de diviser le temps de calcul par 8 sur une machine de bureau moderne.

En conclusion, les travaux menés au cours de cette thèse ont permis de lever un verrou majeur quant au fossé sémantique séparant la représentation des objets d'intérêt par la connaissance de l'expert et la représentation de ces mêmes objets dans une image segmentée. L'importance de l'information contextuelle et son impact sur l'étiquetage des objets d'intérêt a été souligné et des perspectives ont été proposées pour une meilleure intégration de la connaissance spatiale dans un processus de raisonnement. Ceci nous a permis de faire un pas en avant vers un formalisme générique de la connaissance utilisable dans une approche globale d'analyse d'images de télédétection à très haute résolution spatiale.

Bibliographie

- ADAMS, R. et BISCHOF, L. (1994). Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–647.
- AKÇAY, H. G. et AKSOY, S. (2008). Automatic detection of geospatial objects using multiple hierarchical segmentations. *IEEE Transaction on Geoscience and remote sensing*, 46:2097–2111.
- ALLEN, J. F. (1983). Maintaining knowledge about temporal intervals. *Commun. ACM*, 26:832–843.
- AMBERG, V., COULON, M., MARTHON, P. et SPIGAI, M. (2005). Improvement of road extraction in high resolution sar data by a context-based approach. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 490–493.
- AWRANGJEB, M., RAVANBAKSH, M. et FRASER, C. S. (2010). Automatic detection of residential buildings using lidar data and multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(5):457–467.
- BACHER, U. et MAYER, H. (2005). Automatic road extraction from multispectral high resolution satellite images. *International Archives of Photogrammetry and Remote Sensing*, 33:51–57.
- BALL, G. H., HALL, D. J., STANFORD RESEARCH INSTITUTE, UNITED STATES, OFFICE OF NAVAL RESEARCH et INFORMATION SCIENCES BRANCH (1965). *Isodata, a novel method of data analysis and pattern classification*. Stanford Research Institute, Menlo Park, Calif.
- BAUMGARTNER, A., STEGER, C., WIEDEMANN, C., MAYER, H., ECKSTEIN, W. et HEBNER, H. (1996). Update of roads in gis from aerial imagery: Verification and multi-resolution extraction. volume 31, pages 53–58.
- BELARTE, B. et WEMMERT, C. (2012). Apport des relations spatiales dans l'extraction automatique d'informations à partir d'image. In *Atelier Fouille de données complexes, Conférence EGC 2012*. Bordeaux, France.
- BELARTE, B., WEMMERT, C., FORESTIER, G., GRIZONNET, M. et WEBER, C. (2013a). Learning fuzzy rules to characterize objects of interest from remote sensing images. In *Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International*, pages 2986–2989. IEEE.
- BELARTE, B., WEMMERT, C., FORESTIER, G. et WEBER, C. (2013b). Apprentissage de règles floues pour caractériser des objets d'intérêt dans une image de télédétection. In *Atelier Fouille de données complexes, Conférence EGC 2013*. Toulouse, France.
- BENEDIKTSSON, J., SWAIN, P. H. et ERSOY, O. K. (1990). Neural network approaches versus statistical methods in classification of multisource remote sensing data. *IEEE Transactions on geoscience and remote sensing*, 28(4):540–552.
- BILENKO, M., BASU, S. et MOONEY, R. J. (2004). Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of 21st International Conference on Machine Learning (ICML-2004)*, pages 81–88, Banff, Canada.

- BINAGHI, E., MADELLA, P., GRAZIA MONTESANO, M. et RAMPINI, A. (1997). Fuzzy contextual classification of multisource remote sensing images. *Geoscience and Remote Sensing, IEEE Transactions on*, 35(2):326–340.
- BISHOP, C. M. (1996). *Neural Networks for Pattern Recognition*. Oxford University Press, USA, 1 édition.
- BLASCHKE, T. (2003). Object-based contextual image classification built on image segmentation. *Advances in Techniques for Analysis of Remotely Sensed Data, 2003 IEEE Workshop on*, pages 113–119.
- BREIMAN, L., FRIEDMAN, J., STONE, C. J. et OLSHEN, R. A. (1984). *Classification and regression trees*. CRC press.
- BURNETT, C. et BLASCHKE, T. (2003). A multi-scale segmentation/object relationship modelling methodology for landscape analysis. *Ecological modelling*, 168:233–249.
- CANNY, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 8(6):679–698.
- CASTILLO, L., GONZÁLEZ, A. et PÉREZ, R. (2001). Including a simplicity criterion in the selection of the best rule in a genetic fuzzy learning algorithm. *Fuzzy Sets and Systems*, 120(2):309–321.
- CHAKRABORTY, D. et PAL, N. (2004). A neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. *IEEE Transactions on Neural Networks*, 15(1):110–123.
- CHEN, Q., ZHOU, C., LUO, J. et MING, D. (2005). Fast segmentation of high-resolution satellite images using watershed transform combined with an efficient region merging approach. *Lecture Notes in Computer Science*, 3322:621–630.
- CHRISTOPHE, E. et INGLADA, J. (2007). Robust road extraction for high resolution satellite images. *In Image Processing, 2007. IICIP 2007. IEEE International Conference on*, volume 5, pages 437–440.
- CHUI, C. K. (1992). *An Introduction to Wavelets*. Academic Press.
- COMANICIU, D. et MEER, P. (1997). Robust analysis of feature spaces: color image segmentation. *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 750–755.
- COMANICIU, D. et MEER, P. (2002a). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.
- COMANICIU, D. et MEER, P. (2002b). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619.
- CPALKA, K. (2009). A new method for design and reduction of neuro-fuzzy classification systems. *IEEE Transactions on Neural Networks*, 20(4):701–714.
- DERIVAUX, S. (2009). *Construction et classification d'objets à partir d'images de télédétection par une approche itérative guidée par les connaissances du domaine*. Thèse de doctorat, Université de Strasbourg.

- DERIVAUX, S., FORESTIER, G., WEMMERT, C. et LEFEVRE, S. (2008). Extraction de détecteurs d'objets urbains à partir d'une ontologie. In *Atelier Extraction de Connaissance à partir d'Images (ECOI), Journées Francophones Extraction et Gestion des Connaissances (EGC 2008)*, pages 71–81, Sophia Antipolis, France.
- DERIVAUX, S., FORESTIER, G., WEMMERT, C. et LEFÈVRE, S. (2010). Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation. *Pattern Recognition Letters*, 31(15):2364–2374.
- D'ELIA, C., POGGI, G. et SCARPA, G. (2003). A tree-structured markov random field model for bayesian image segmentation. *IEEE Transaction on Image Processing*, 12(10):1259–1273.
- FAN, J., YAU, D., ELMAGARMID, A. et AREF, W. (2001). Automatic image segmentation by integrating color-edge extraction and seeded region growing. *Image Processing, IEEE Transactions on*, 10(10):1454–1466.
- FAN, J., ZENG, G., BODY, M. et HACID, M.-S. (2005). Seeded region growing: an extensive and comparative study. *Pattern Recognition Letters*, 26(8):1139–1156.
- FERNÁNDEZ, A., CALDERÓN, M., BARRENECHEA, E., BUSTINCE, H. et HERRERA, F. (2010). Solving multi-class problems with linguistic fuzzy rule based classification systems based on pairwise learning and preference relations. *Fuzzy Sets and Systems*, 161(23):3064–3080.
- FOODY, G. (1996). Fuzzy modelling of vegetation from remotely sensed imagery. *Ecological Modelling*, 85(1):3–12. Fuzzy Logic in Ecological Modelling.
- FORESTIER, G. (2010). *Connaissances et clustering collaboratif d'objets complexes multisources*. Thèse de doctorat, Université de Strasbourg.
- FORESTIER, G., DERIVAUX, S., WEMMERT, C. et GANÇARSKI, P. (2008). Interprétation d'images basée sur une approche évolutive guidée par une ontologie. In *Journées Francophones Extraction et Gestion des Connaissances (EGC 2008)*, volume 2, pages 469–474, Sophia Antipolis, France.
- FORESTIER, G., GANÇARSKI, P. et WEMMERT, C. (2010). Collaborative clustering with background knowledge. *Data and Knowledge Engineering*, 69(2):211–228.
- FREKSA, C. (1992). Using orientation information for qualitative spatial reasoning. In FRANK, A., CAMPARI, I. et FORMENTINI, U., éditeurs : *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, volume 639 de *Lecture Notes in Computer Science*, pages 162–178. Springer Berlin / Heidelberg.
- GAETANO, R., SCARPA, G. et POGGI, G. (2009). Hierarchical texture-based segmentation of multiresolution remote-sensing images. *IEEE Transaction on Geoscience and remote sensing*, 47:2129–2141.
- GANÇARSKI, P. et WEMMERT, C. (2007). Collaborative multi-step mono-level multi-strategy classification. *Multimedia Tools and Applications*, 35(1):1–27.
- GERKE, M. et HEIPKE, C. (2008). Image-based quality assessment of road databases. *International Journal of Geographical Information Science*, 22:871–894.
- GIMEL'FARB, G. (1996). Texture modeling by multiple pairwise pixel interactions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18:1110–1114.

- GUEGUEN, L., PESARESI, M., GERHARDINGER, A. et SOILLE, P. (2012). Characterizing and counting roofless buildings in very high resolution optical images. *Geoscience and Remote Sensing Letters, IEEE*, 9(1):114–118.
- HEERMANN, P. et KHAZENIE, N. (1992). Classification of multispectral remote sensing data using a back-propagation neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 30(1):81–88.
- HOFMANN, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196.
- HOJJATOLESLAMI, S. et KITTLER, J. (1998). Region growing: a new approach. *IEEE Transactions on Image processing*, 7(7):1079—1084.
- INGLADA, J. et MICHEL, J. (2009). Qualitative spatial reasoning for high-resolution remote sensing image analysis. *IEEE Transaction on Geoscience and remote sensing*, 47:599–612.
- ISHIBUCHI, H. et NAKASHIMA, T. (2001). Effect of rule weights in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, 9(4):506–515.
- ISHIBUCHI, H. et YAMAMOTO, T. (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems*, 141(1):59–88.
- ISHIBUCHI, H. et YAMAMOTO, T. (2005). Rule weight specification in fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, 13(4):428–435.
- JAIN, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666.
- JAIN, A. K. et DUBES, R. C. (1988). *Algorithms for Clustering Data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- JEON, B.-K., JANG, J.-H. et HONG, K.-S. (2002). Road detection in spaceborne sar images using a genetic algorithm. *Transactions on Geoscience and Remote Sensing*, 40(1):22–29.
- KASS, M., WITKIN, A. et TERZOPOULOS, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331.
- KAUFMAN, L. et ROUSSEEUW, P. (1987). Clustering by means of medoids. In DODGE, Y., éditeur : *Statistical Data Analysis Based on the L1-Norm and Related Methods*, pages 405–416. North-Holland.
- KE, C.-Q., XIA, L., TANG, G.-D. et CAO, X. (2008). Detecting urban vegetation using a object-oriented method with quickbird imagery. volume 3, pages 138–141.
- KIM, T. et MULLER, J.-P. (1996). Automated urban area building extraction from high resolution stereo imagery. *Image and Vision Computing*, 14(2):115–130.
- KIM, T. et MULLER, J.-P. (1999). Development of a graph-based approach for building detection. *Image and Vision Computing*, 17(1):3–14.
- KUMAR, N. et KUMMAMURU, K. (2008). Semisupervised clustering with metric learning using relative comparisons. *Knowledge and Data Engineering, IEEE Transactions on*, 20(4):496–503.

- KURTZ, C., GANÇARSKI, P., PASSAT, N. et PUISSANT, A. (2013). A hierarchical semantic-based distance for nominal histogram comparison. *Data and Knowledge Engineering*, 87(1):206–225.
- KURTZ, C., PASSAT, N., GANÇARSKI, P. et PUISSANT, A. (2010). Multi-resolution region-based clustering for urban analysis. *International Journal of Remote Sensing*, 31(22):5941–5973.
- KURTZ, C., PASSAT, N., GANÇARSKI, P. et PUISSANT, A. (2012). Extraction of complex patterns from multiresolution remote sensing images: A hierarchical top-down methodology. *Pattern Recognition*, 45(2):685–706.
- KURTZ, C., PASSAT, N., PUISSANT, A. et GANÇARSKI, P. (2011a). Hierarchical segmentation of multiresolution remote sensing images. In SOILLE, P., PESARESI, M. et OUZOUNIS, G., éditeurs : *Proceedings of the International Symposium on Mathematical Morphology – ISMM 2011*, volume 6671, pages 343–354, Intra (Italy). Springer.
- KURTZ, C., PUISSANT, A., PASSAT, N. et GANÇARSKI, P. (2011b). An interactive approach for extraction of urban patterns from multisource images. In *Proceedings of the IEEE Joint Urban Remote Sensing Event – JURSE 2011*, pages 321–324, Munich (Germany). IEEE Geoscience and Remote Sensing Society.
- KUTZ, O., LUTZ, C., WOLTER, F. et ZAKHARYASCHEV, M. (2004). E-connections of abstract description systems. *Artificial Intelligence*, 156(1):1–73.
- LAHA, A., PAL, N. et DAS, J. (2006). Land cover classification using fuzzy rules and aggregation of contextual information through evidence theory. *IEEE Transactions on Geoscience and Remote Sensing*, 44(6):1633–1641.
- LAI, K. et CHIN, R. (1995). Deformable contours: modeling and extraction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17:1084–1090.
- LANCE, G. N. et WILLIAMS, W. T. (1967). A general theory of classificatory sorting strategies: 1. hierarchical systems. *The Computer Journal*, 9(4):373–380.
- LEE, C. (1990a). Fuzzy logic in control systems: fuzzy logic controller. I. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):404–418.
- LEE, C. (1990b). Fuzzy logic in control systems: fuzzy logic controller. II. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):419–435.
- LHOMME, S. (2005). *Identification du bâti à partir d’images satellitaires à très haute résolution spatiale*. Thèse de doctorat, Université Louis Pasteur, Strasbourg, France.
- MACQUEEN, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. California, USA.
- MAMDANI, E. (1977). Application of fuzzy logic to approximate reasoning using linguistic synthesis. *Computers, IEEE Transactions on*, C-26(12):1182–1191.
- MAMDANI, E. et ASSILIAN, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13.
- MEHNERT, A. et JACKWAY, P. (1997). An improved seeded region growing algorithm. *Pattern Recognition Letters*, 18(10):1065–1071.

- MELGANI, F. et BRUZZONE, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1778–1790.
- MENA, J. et MALPICA, J. (2005). An automatic method for road extraction in rural and semi-urban areas starting from high resolution satellite imagery. *Pattern Recognition Letters*, 26(9):1201–1220.
- MERCER, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 209:415–446.
- MORATZ, R., RENZ, J. et WOLTER, D. (2000). Qualitative spatial reasoning about line segments. In *ECAI 2000. Proceedings of the 14th European Conference on Artificial Intelligence*, pages 234–238. IOS Press.
- MÜLLNER, D. (2011). Modern hierarchical, agglomerative clustering algorithms. *arXiv:1109.2378 [cs, stat]*.
- NAJMAN, L. et SCHMITT, M. (1996). Geodesic saliency of watershed contours and hierarchical segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(12):1163–1173.
- NAKASHIMA, T., SCHAEFER, G., YOKOTA, Y. et ISHIBUCHI, H. (2007). A weighted fuzzy classifier and its application to image processing tasks. *Fuzzy Sets and Systems*, 158(3):284–294.
- NAUCK, D. et KRUSE, R. (1997). A neuro-fuzzy method to learn fuzzy classification rules from data. *Fuzzy Sets and Systems*, 89(3):277–288.
- NOVÁK, V. (2012). Reasoning about mathematical fuzzy logic and its future. *Fuzzy Sets and Systems*, 192(0):25–44.
- NOZAKI, K., ISHIBUCHI, H. et TANAKA, H. (1996). Adaptive fuzzy rule-based classification systems. *IEEE Transactions on Fuzzy Systems*, 4(3):238–250.
- PAPADIAS, D. et EGENHOFER, M. J. (1997). Algorithms for hierarchical spatial reasoning. *GeoInformatica*, 1:251–273. 10.1023/A:1009760430440.
- PAWLAK, Z. (1982). Rough sets. *International Journal of Parallel Programming*, 11:341–356. 10.1007/BF01001956.
- PENG, J., ZHANG, D. et LIU, Y. (2005). An improved snake model for building detection from urban aerial images. *Pattern Recognition Letters*, 26(5):587–595.
- PETROU, M. et KITTLER, J. (1991). Optimal edge detectors for ramp edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):483–491.
- PHALKE, S. et COULOIGNER, I. (2005). Change detection of linear man-made objects using feature extraction technique. In *Proceedings of 3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas (URBAN 2005)*, volume 1, pages 14–16.
- POULAIN, V. (2010). *Fusion d'images optique et radar à haute résolution pour la mise à jour de bases de données cartographiques*. Thèse de doctorat, Institut National Polytechnique de Toulouse, Toulouse, France.

- QIAOPING, Z. et COULOIGNER, I. (2004). Automatic road change detection and gis updating from high spatial remotely-sensed imagery. *Geo-Spatial Information Science*, 7:89–95.
- QUINLAN, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- RANDELL, D., R, D. A., CUI, Z., COHN, A. et JT, L. (1992). A spatial logic based on regions and connection. *In 3rd Int. Conf. on knowledge representation and reasoning*, pages 165–176.
- RENZ, J. et NEBEL, B. (1999). On the complexity of qualitative spatial reasoning: A maximal tractable fragment of the region connection calculus. *Artificial Intelligence*, 108(1-2):69–123.
- ROTTENSTEINER, F., TRINDER, J., CLODE, S. et KUBIK, K. (2005). Using the dempster-shafer method for the fusion of lidar data and multi-spectral images for building detection. *Information Fusion*, 6(4):283–300.
- ROTTENSTEINER, F., TRINDER, J., CLODE, S. et KUBIK, K. (2007). Building detection by fusion of airborne laser scanner data and multi-spectral images: Performance evaluation and sensitivity analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62(2):135–149.
- RUSSO, M. (2000). Genetic fuzzy learning. *IEEE Transactions on Evolutionary Computation*, 4(3):259–273.
- SCHÖLKOPF, B. et SMOLA, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.
- SEBARI, I. et HE, D.-C. (2013). Automatic fuzzy object-based analysis of VHRS images for urban objects extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 79:171–184.
- SHAFER, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.
- SHIH, F. Y. et CHENG, S. (2005). Automatic seeded region growing for color image segmentation. *Image and Vision Computing*, 23(10):877–886.
- STEGER, C. (1998). An unbiased detector of curvilinear structures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20:113–125.
- STRAUB, B., WIEDEMANN, C. et HEIPKE, C. (2000). Towards the automatic interpretation of images for gis update. *International Archives of Photogrammetry and Remote Sensing*, 33(B2; PART 2):525–532.
- SURESH, S. et SUBRAMANIAN, K. (2011). A sequential learning algorithm for meta-cognitive neuro-fuzzy inference system for classification problems. *In The 2011 International Joint Conference on Neural Networks (IJCNN)*, pages 2507–2512.
- TÖNJES, R. et GROWE, S. (1998). Knowledge based road extraction from multisensor imagery. *International Archives of Photogrammetry and Remote Sensing*, 32:387–393.
- TUPIN, F., HOUSHMAND, B. et DATCU, M. (2002). Road detection in dense urban areas using sar imagery and the usefulness of multiple views. *IEEE Transactions on Geoscience and Remote Sensing*, 40:2405–2414.
- TUPIN, F., MAITRE, H., MANGIN, J.-F., NICOLAS, J.-M. et PECHERSKY, E. (1998). Detection of linear features in sar images: application to road network extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 36:434–453.

- VANEGAS, M., BLOCH, I. et INGLADA, J. (2010a). Detection of aligned objects for high resolution image understanding. In *Geoscience and Remote Sensing Symposium, 2010 IEEE International, IGARSS 2010*.
- VANEGAS, M., BLOCH, I. et INGLADA, J. (2010b). Searching aligned groups of objects with fuzzy criteria. In HÜLLERMEIER, E., KRUSE, R. et HOFFMANN, F., éditeurs : *Computational Intelligence for Knowledge-Based Systems Design*, volume 6178 de *Lecture Notes in Computer Science*, pages 605–613. Springer Berlin / Heidelberg.
- VANEGAS, M., BLOCH, I., MAÎTRE, H. et INGLADA, J. (2009). Approximate parallelism between fuzzy objects: Some definitions. In DI GESÙ, V., PAL, S. et PETROSINO, A., éditeurs : *Fuzzy Logic and Applications*, volume 5571 de *Lecture Notes in Computer Science*, pages 12–19. Springer Berlin / Heidelberg.
- VINCENT, L. et SOILLE, P. (1991). Watersheds in digital spaces : an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598.
- WAGSTAFF, K., CARDIE, C., ROGERS, S. et SCHROEDL, S. (2001). Constrained k-means clustering with background knowledge. In *Proceeding of the International Conference of Machine Learning*, pages 577–584. Morgan Kaufmann.
- WANG, F. (1990). Fuzzy supervised classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 28(2):194–201.
- WANG, J.-S. et LEE, C. S. G. (2002). Self-adaptive neuro-fuzzy inference systems for classification applications. *IEEE Transactions on Fuzzy Systems*, 10(6):790–802.
- WARD, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244.
- WEMMERT, C. (2000). *Classification hybride distribuée par collaboration de méthodes non supervisées*. Thèse de doctorat, Université de Strasbourg.
- WIEDMANN, C. (2001). *Extraktion von Straßennetzen aus optischen Satellitenbilddaten*. Thèse de doctorat, Technische Universität München, Munich, Allemagne.
- WIEDMANN, C. (2003). Internal evaluation of road networks. *International Archives of Photogrammetry and Remote Sensing*, 34:93–98.
- WILLIAMS, D. J. et SHAH, M. (1992). A fast algorithm for active contours and curvature estimation. *CVGIP: Image Understanding*, 55:14–26.
- WORBOYS, M. (1998). Imprecision in finite resolution spatial data. *GeoInformatica*, 2:257–279. 10.1023/A:1009769705164.
- WU, W., LI, L., YANG, J. et LIU, Y. (2010). A modified gradient-based neuro-fuzzy learning algorithm and its convergence. *Information Sciences*, 180(9):1630–1642.
- YANG, C.-C., PRASHER, S. O., ENRIGHT, P., MADRAMOOTOO, C., BURGESS, M., GOEL, P. K. et CALLUM, I. (2003). Application of decision tree technology for image classification using remote sensing data. *Agricultural Systems*, 76(3):1101–1117.
- ZADEH, L. (1965). Fuzzy sets. *Information and Control*, 8(3):338–353.

- ZADEH, L. (1975). Fuzzy logic and approximate reasoning. *Synthese*, 30(3):407–428.
- ZHANG, Y. (1999). Optimisation of building detection in satellite images by combining multi-spectral classification and texture filtering. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(1):50–60.
- ZHU, A.-X., BAND, L. E., DUTTON, B. et NIMLOS, T. J. (1996). Automated soil inference under fuzzy logic. *Ecological Modelling*, 90(2):123–145. Fuzzy Modelling in Ecology.
- ZIEMS, M., GERKE, M. et HEIPKE, C. (2007). Automatic road extraction from remote sensing imagery incorporating prior information and colour segmentation. *International Archives of Photogrammetry and Remote Sensing*, 36:141–147.

Annexes

A

Jeux de données



(a) L'image et les exemples



(b) La segmentation

FIGURE A.1 – Image QUICKBIRD utilisée par la validation expérimentale. Cette image est une fusion Pan-MS d'une image QUICKBIRD (©DigitalGlobe 2001) d'une résolution de 0,7 mètres acquise en 2002, représentant un quartier à Saint Orens, dans la banlieue de Toulouse. L'image fait 508×507 pixels et contient quatre bandes : bleue, vert, rouge et infrarouge.



(a) L'image et les exemples

(b) La segmentation



(c) L'image et les exemples

(d) La segmentation

FIGURE A.2 – Extraits de l'image PLEIADES. Cette image est une fusion Pan-MS d'une image PLEIADES d'une résolution de 0,5 mètres acquise en 2012, de l'intégralité de Strasbourg. L'image contient quatre bandes : bleue, vert, rouge et infrarouge. Deux extraits sont utilisés. Le premier fait 438×488 pixels et représente un quartier résidentiel de Cronenbourg (A.2(a) image et exemples ; A.2(b) segmentation). Le second fait 1479×1546 pixels et représentant le quartier des Quinze (A.2(c) image et exemples ; A.2(d) segmentation).



FIGURE A.3 – Cette image PLEIADES est une fusion Pan-MS d'une résolution de 0,5 mètres acquise en 2012, représentant la zone nord est de Strasbourg. L'image fait 9211×11275 pixels et contient quatre bandes : rouge, vert, bleu et proche infrarouge.



FIGURE A.4 – Cette image PLEIADES est une fusion Pan-MS d'une résolution de 0,5 mètres acquise en 2012, représentant la zone nord est de Strasbourg. L'image fait 9211×11275 pixels et contient trois bandes : infrarouge, rouge et vert. Les couleurs sont rééchantillonnées en huit bits pour faciliter l'affichage.

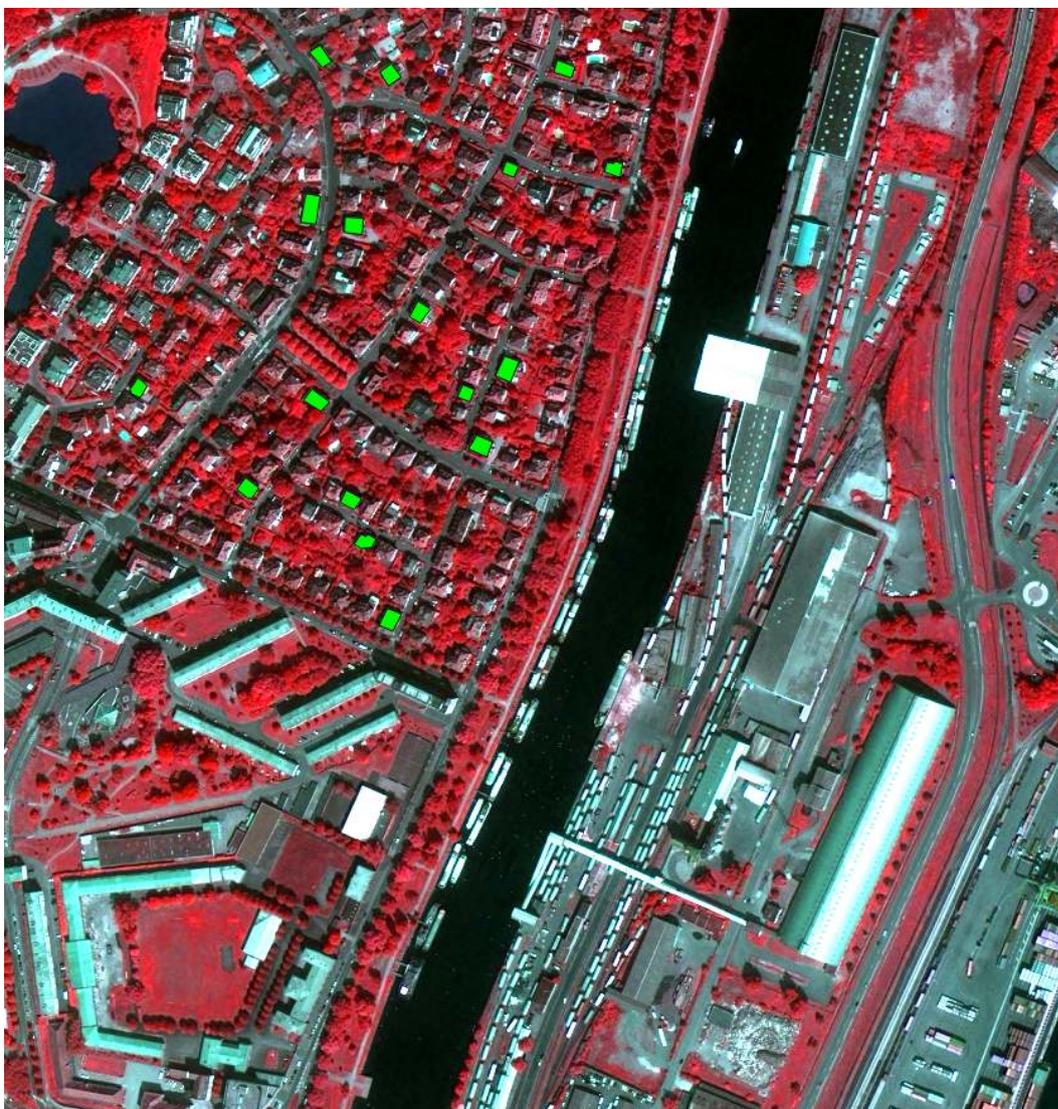


FIGURE A.5 – Région d'intérêt extrait de l'image A.4. En vert, les dix sept exemples du concept *pavillon* utilisé dans le chapitre validation expérimentale.

B

Segmentations associées aux images Pléiades

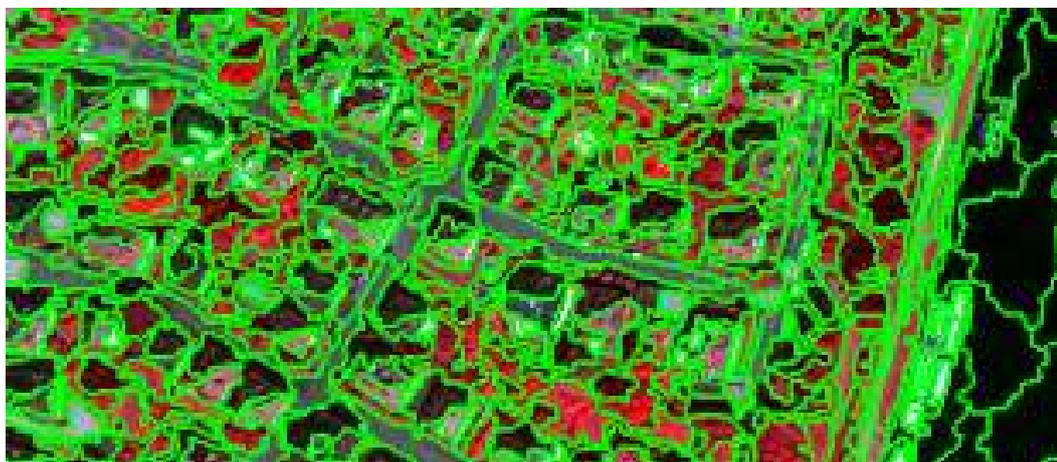
Les images A.4 et A.3 présentées dans l'annexe A servent de base d'étude au projet COCLICO¹ sur la zone de Strasbourg. Deux segmentations sont associées à cette zone. Des extraits de ces segmentations sont utilisés dans ce manuscrit. Un exemple est donnée dans la figure B.1.

Segmentation de travail. Cette segmentation a été réalisée avec le logiciel eCognition² par un expert du LIVE. L'algorithme de segmentation utilisé est un algorithme multiresolution avec paramètre d'échelle réglé à 160 et le shape factor réglé à 0.5. Cette segmentation est considérée comme la *segmentation de travail* par les expert du projet COCLICO. Le résultat est sursegmenté, ce qui nous permet d'appliquer les méthodes proposées au cours de cette thèse. Un exemple est donnée dans la figure B.1(a).

Segmentation-Classification. Cette segmentation est issue d'une classification / interprétation réalisée par le SERTIT à l'échelle de l'objet (1/2000). Ces données peuvent être considérées comme des données de référence sur l'ensemble de la zone. Dans cette segmentation les bâtiments et les routes sont pour la plupart correctement découpés. Nous considérons cette segmentation comme une segmentation *parfaite* et nous nous en servons comme point de comparaison pour nos méthodes. Un exemple est donnée dans la figure B.1(b).

1. <http://icube-coclico.unistra.fr/index.php/Coclico>

2. <http://www.ecognition.com/>



(a) Segmentation de travail



(b) Segmentation-Classification

FIGURE B.1 – Extraits des segmentations de la zone de Strasbourg. Nous pouvons voir ici que la segmentation de travail est très sursegmentée tandis que dans la segmentation-classification les régions sont mieux découpées (routes droites, pavillons rectangulaires, végétation fusionnée).

Extraction, analyse et utilisation de relations spatiales entre objets d'intérêt pour une analyse d'images de télédétection guidée par des connaissances du domaine

Résumé

Les nouveaux capteurs satellitaires permettent l'acquisition d'images d'un très haut niveau de détail à des cadences élevées, produisant ainsi une importante masse de données. Le traitement manuel de ces données étant devenu impossible, de nouveaux outils sont nécessaires afin de les traiter automatiquement. Des algorithmes de segmentation efficaces sont nécessaires pour extraire des objets d'intérêt de ces images. Cependant les segments produits ne correspondent pas aux objets d'intérêt, rendant difficile l'utilisation de connaissances expertes.

Dans le cadre de cette thèse nous proposons de changer le niveau d'interprétation d'une image afin de voir les objets d'intérêt pour l'expert comme des objets composés par des segments. Pour cela, nous avons mis en place un processus d'apprentissage multi-niveaux dans le but d'apprendre ces règles de composition. Une règle de composition ainsi apprise peut ensuite être utilisée pour extraire les objets d'intérêt correspondant.

Dans un second temps, nous proposons d'utiliser l'algorithme d'apprentissage de règles de composition comme première étape d'une approche montante-descendante. Cette chaîne de traitement a pour objectif d'améliorer la classification à partir des informations contextuelles et de connaissances expertes. Des objets composés de plus haut niveau sémantique sont extraits à partir de règles apprises ou fournies par l'expert, et cette nouvelle information est utilisée pour mettre à jour la classification des objets aux niveaux inférieurs.

L'ensemble de ces travaux ont été testés et validés sur des images Pléiades représentant la ville de Strasbourg. Les résultats obtenus montrent l'efficacité de l'apprentissage de règles de composition pour faire le lien entre connaissance experte et segmentation, ainsi que l'intérêt de l'utilisation d'informations contextuelles dans l'analyse d'images de télédétection à très haute résolution spatiale.

Mots clés : Analyse d'images, télédétection, image à très haute résolution spatiale, classification guidée par la connaissance

Abstract

The new remote sensors allow the acquisition of very high spatial resolution images at high speeds, thus producing a large volume of data. Manual processing of these data has become impossible, new tools are needed to process them automatically. Effective segmentation algorithms are required to extract objects of interest of these images. However, the produced segments do not match to objects of interest, making it difficult to use expert knowledge.

In this thesis we propose to change the level of interpretation of an image in order to see the objects of interest of the expert as objects composed of segments. For this purpose, we have implemented a multi-level learning process in order to learn composition rules. Such a composition rule can then be used to extract corresponding objects of interest.

In a second step, we propose to use the composition rules learning algorithm as a first step of a bottom-up top-down approach. This processing chain aims at improving the classification from contextual knowledge and expert information. Composed objects of higher semantic level are extracted from learned rules or rules provided by the expert, and this new information is used to update the classification of objects at lower levels.

The proposed method has been tested and validated on Pléiades images representing the city of Strasbourg. The results show the effectiveness of the composition rules learning algorithm to make the link between expert knowledge and segmentation, as well as the interest of the use of contextual information in the analysis of remotely sensed very high spatial resolution images.

Keywords : Image analysis, remote sensing, very high spatial resolution image, knowledge driven classification

