



Habilitation à diriger des recherches

Discipline : Informatique

Clustering collaboratif et connaissances expertes – Application à l'analyse d'images

présentée par

Cédric Wemmert

soutenue publiquement le 4 juillet 2012

Membres du jury

Garant scientifique : Pierre Gançarski, Professeur, Université de Strasbourg

Rapporteurs : Florence Sedès, Professeur, Université de Toulouse III

Antoine Cornuejols, Professeur, AgroParistech

Arnaud Martin, Professeur, Université de Rennes I

Examinateuse : Christiane Weber, Directrice de recherche, Université de Strasbourg

Préambule

Depuis toujours, l'homme et bien d'autres espèces animales ont compris qu'il était plus aisné de s'associer pour réaliser une tâche difficile plutôt que de l'effectuer seul. Le verbe collaborer, du latin *cum* (avec) et *laborare* (travailler), signifie travailler de concert. L'objectif de la **collaboration** peut être double. Il peut s'agir de réaliser une tâche pour laquelle la participation de plusieurs est indispensable, un seul individu étant incapable de la réaliser seul, ou d'améliorer la réalisation selon un ou plusieurs critères de qualité (rapidité, confort, facilité, qualité/précision du résultat).

Dans notre cas, nous utilisons ce terme dans le contexte d'un ensemble d'experts travaillant conjointement à la résolution d'un problème particulier commun, l'objectif étant d'améliorer la qualité du résultat. Pour cela, il est communément admis qu'il est important que le groupe présente une certaine hétérogénéité, la collaboration entre individus identiques ne permettant pas d'améliorer les performances de l'individu lui-même (mis-à-part la rapidité d'exécution et de résolution du problème éventuellement). Ainsi, chaque expert de l'ensemble apporte ses spécificités et ses connaissances propres, contribuant à la réussite et à l'amélioration de la qualité du résultat. Ces connaissances sont naturellement incomplètes et peuvent être imprécises voire fausses ; dans le cas contraire, cela supposerait l'existence d'un expert ayant l'intégralité des connaissances du problème et des méthodes pour le résoudre, ce qui rendrait inutile toute collaboration avec autrui.

Ces **connaissances** peuvent être caractérisées en fonction de ce sur quoi elles portent :

- le savoir-faire, qui correspond à la capacité de résoudre selon une méthode propre tout ou partie du problème ;
- les connaissances du domaine, qui correspondent à l'ensemble des informations générales sur le domaine dont relève le problème ;
- le contexte du problème, qui correspond aux données du problème ainsi qu'à des informations particulières sur ces données et le problème lui-même.

De plus, dans le domaine de la classification, de nombreux travaux menés en particulier dans notre équipe, dans le cas supervisé et non supervisé, ont mené à la conclusion qu'il n'existe pas de méthode optimale permettant la résolution de tout type de problème et que la méthode devait être adaptée aux données afin d'obtenir de bons résultats. Dans le cas de la classification supervisée, plusieurs méthodes de combinaison et de collaboration ont rapidement été proposées. Les informations sur les classes à obtenir étant connues, il est plus facile de comparer et combiner différents résultats que dans le cas non supervisé pour lequel aucune information a priori n'est connue.

C'est pourquoi nous avons décidé de proposer une solution à ce problème de la collaboration dans un ensemble de méthodes non supervisées, ayant des stratégies différentes, dans le but d'obtenir automatiquement un meilleur résultat.

Cette définition de la collaboration et de la connaissance distribuée entre experts nous amène naturellement au domaine de l'intelligence artificielle distribuée et aux systèmes multi-agents. Dans ces disciplines, de nombreux travaux ont été menés afin de définir de manière générique comment établir une collaboration entre experts ou comment représenter et raisonner avec de la connaissance distribuée. Ainsi, nos travaux d'origine

sur ce sujet se sont inspirés de ce paradigme de manière naturelle. Nous y avons modélisé une approche de clustering collaboratif sous forme de système multi-agents.

Les géosciences et plus particulièrement l'observation de la Terre via les images satellites ou aériennes forment un domaine d'étude très intéressant et complexe, pour lequel il subsiste de nombreux problèmes en classification automatique. Pensant que ce type d'approche collaborative est pertinente et peut apporter des solutions dans ce domaine, nous avons validé nos travaux dans ce cadre. Malheureusement, nous nous sommes rapidement heurtés à un problème de passage à l'échelle lors de l'application de nos méthodes de classification à ces données réelles, et notamment à des données de type image. En effet, les limites de modèles multi-agents, représentant des modèles de cognition généraux et de très haut niveau sémantique, apparaissent dès lors que l'on essaie de les appliquer à des données de grand volume comme des images de télédétection. Nous nous sommes alors tourné vers un formalisme plus spécifique et adapté aux problèmes qui nous intéressent, et en particulier à la classification d'images de télédétection.

Ainsi, les travaux présentés dans le cadre de ce mémoire d'habilitation à diriger des recherches, basés sur ce paradigme de la collaboration et de la connaissance, proposent des contributions à la fois dans le domaine de la classification non supervisée (*clustering*) collaborative guidée par des connaissances du domaine, et dans celui de l'extraction automatique d'informations à partir d'images de télédétection. Nous avons articulé la présentation de nos travaux dans ce domaine autour de ces deux thèmes.

Clustering collaboratif guidé par des connaissances

Ce premier thème de recherche est le plus théorique des deux et rassemble nos contributions sur la collaboration de méthodes de clustering dans un cas général, ainsi que sur l'injection de connaissances expertes dans ce processus. Ainsi, nous traitons les deux aspects, collaboration au niveau des savoir-faire et à celui des connaissances du domaine. Le troisième aspect, concernant le contexte du problème, n'a quant à lui été traité que dans la partie applicative de nos travaux présentée plus loin.

Savoir-faire Il existe de nombreuses méthodes de clustering pouvant suivre des objectifs différents et donc des stratégies différentes, et ainsi proposer pour un même jeu de données des résultats différents. Cette profusion de méthodes impose à l'expert de devoir choisir *a priori* une méthode ainsi que ses paramètres, sachant que ce choix va conditionner le résultat. Même si l'utilisateur dispose de connaissances *a priori* sur ses données et a une bonne expertise en matière d'algorithmes de clustering, choisir la *méilleure* méthode avec les *meilleurs* paramètres pour obtenir le *meilleur* clustering reste extrêmement difficile. Cette notion d'optimalité est déjà une question difficile en soi que nous aborderons par la suite.

Ainsi sont apparues à la fin des années 90 les premières propositions de travail sur la collaboration et sur l'apport de différentes méthodes de clustering à la résolution d'un problème donné. C'est dans ce cadre que nous avons proposé la méthode originale de

classification collaborative non supervisée SAMARAH. Cette méthode consiste à mettre en œuvre la collaboration de différentes méthodes de clustering afin de trouver un consensus sur le clustering du jeu de données passé en paramètre. L'objectif de cette collaboration est de réduire l'impact du choix d'une méthode particulière et de ses paramètres lors du processus de classification, et de le rendre plus robuste. Étant donné un ensemble de résultats de clustering initiaux, l'idée est de modifier de façon itérative et collaborative les résultats initiaux afin d'en améliorer la similarité et la qualité. Cela autorise la construction d'un consensus final plus pertinent et de meilleure qualité.

Les recherches menées autour de cette méthode de collaboration nous ont permis de mieux comprendre le processus de clustering collaboratif. Nous avons notamment réalisé une étude détaillée des stratégies possibles pour la sélection des conflits à résoudre entre les méthodes collaborant. Celle-ci a conduit à la définition d'un cadre générique de formalisation de stratégies de collaboration. C'est dans ce cadre qu'une version de SAMARAH basée sur un algorithme génétique a été proposée, celle-ci permettant d'obtenir de meilleurs résultats que la stratégie initiale.

Connaissances du domaine Comme indiqué précédemment, la notion d'optimalité pour un résultat est une question difficile en soi. Pour tenter d'y répondre, nous avons mené une étude sur les critères et techniques d'évaluation de résultats de clustering, afin de permettre de mieux comprendre la problématique de la combinaison de résultats. Cette étude donne de premiers arguments en faveur de l'introduction de connaissances du domaine dans le processus de classification, car elle conclut sur la difficulté de définir un critère universel de qualité d'un résultat de clustering, sans connaissance *a priori* des objectifs de l'utilisateur.

Le processus de clustering est par définition une approche non supervisée, c'est-à-dire qu'il se base uniquement sur les données et n'utilise pas ou peu de connaissances *a priori*. Or, sans aucune supervision, les algorithmes peuvent aboutir à des solutions non pertinentes. Pour résoudre ce problème, des travaux actuels dans plusieurs équipes se concentrent sur la définition d'approches permettant de guider le processus de clustering par des connaissances du domaine, tout en limitant l'implication de l'expert humain lors du processus de classification. Ainsi, plusieurs études ont montré le rôle important joué par ces connaissances du domaine ainsi que par l'expert dans le processus de fouille de données. Elles expliquent notamment que l'extraction de connaissances à partir de données ne peut pas être totalement automatique et qu'il est nécessaire d'étudier les mécanismes permettant la combinaison entre, d'une part les traitements automatiques (sans connaissance *a priori*), et d'autre part des traitements supervisés par l'utilisateur. Afin d'automatiser malgré tout le processus, il est nécessaire alors de parvenir à représenter les connaissances expertes et de les intégrer dans le système. Cependant, en fonction des domaines, la représentation et le type de connaissances à utiliser peuvent être très hétérogènes et plus ou moins complexes.

Deux représentations de connaissances ont principalement été proposées dans la littérature pour le clustering. La première vise à permettre l'utilisation de contraintes

entre objets et la seconde l'utilisation d'objets étiquetés. Nous avons étudié ces deux représentations ainsi que différentes mesures de pureté qui permettent de tirer parti de la connaissance des étiquettes d'un ensemble d'objets.

Nous avons alors défini une intégration de ce type de critère dans le processus de clustering collaboratif via plusieurs techniques, lors de l'estimation de la qualité d'un clustering notamment et aussi dans une nouvelle approche évolutionnaire. Les résultats obtenus confirment l'intérêt de l'utilisation de connaissances en clustering et plus particulièrement en clustering collaboratif.

Connaissances expertes en observation de la Terre

Le second thème de recherche concerne l'utilisation de méthodes collaboratives guidées par des connaissances dans le domaine de l'observation de la Terre. En effet, comme indiqué précédemment, l'observation de la Terre *via* la télédétection et l'imagerie aérienne est un domaine d'étude très complexe pour lequel il existe de nombreuses applications (étude de l'occupation du sol, des dynamiques urbaines, ou de l'agriculture), et qui fournit de nombreuses données hétérogènes et complexes. L'image de télédétection est le prototype même d'une donnée complexe de par sa structure physique, mais aussi par le fossé sémantique entre les informations de plus bas niveau (radiométrie des pixels) et les informations à extraire (occupation du sol par exemple). Ce fossé sémantique est défini comme le manque de concordance entre l'information bas niveau et l'interprétation faite par un expert. Ceci est d'autant plus vrai avec l'apparition des images à très haute résolution spatiale (sub-métrique). En effet dans ce cas, les objets à classer (bâtiments, routes, etc.) sont représentés par des ensembles de pixels et non plus par des pixels isolés. Affecter une étiquette à chacun des pixels indépendamment des autres n'a plus de sens et ne permet pas d'extraire les objets d'intérêt recherchés par l'expert. Les étiquettes des classes doivent être appliquées à des composantes connexes de l'image représentant un objet réel. Ces objets n'étant pas définis et identifiés *a priori*, ils doivent être tout d'abord construits (segmentation) puis caractérisés avant de pouvoir être classés (suivant des critères spectraux ou géométriques). Dans ce cas, on parle communément de classification basée objet ou région.

Les résultats décevants de la classification automatique face à ce type de données ont poussé la communauté scientifique à proposer de nouvelles approches. Ainsi, des méthodes tentent de tirer parti de l'ensemble des connaissances et informations disponibles afin d'améliorer les performances des algorithmes de classification. Nos travaux se sont placés dans ce cadre et se sont concentrés sur la définition d'une approche de construction et de classification guidée par des connaissances expertes ; la connaissance étant utilisée lors de toutes les étapes du processus :

- le clustering qui consiste à extraire de manière totalement automatique des informations à partir des données et qui consiste souvent en un premier traitement des données afin de mieux les comprendre ;
- la segmentation qui consiste à agréger des pixels connexes afin de construire des objets qui idéalement doivent correspondre aux objets d'intérêt de l'image ;

-
- la classification qui associe une étiquette de classe à chaque objet ;
 - la détection enfin, qui cherche à construire les objets d'une classe d'intérêt dans l'image.

Pour chacune de ces tâches, nous avons proposé une ou plusieurs méthodes génériques d'extraction d'informations à partir d'images, utilisant les connaissances disponibles. Ces méthodes ont ensuite été appliquées dans le domaine de la classification d'images de télédétection mais restent adaptées à différents domaines.

Deux types de connaissance y sont utilisés :

- les connaissances de l'expert, qui sont implicites et nécessitent d'être formalisées, *via* des méthodes de représentation des connaissances afin de les rendre opérables ;
- les exemples fournis par l'expert qui représentent une connaissance indirecte.

Ces différentes sources de connaissance sont par nature hétérogènes et nécessitent des traitements spécifiques afin de pouvoir être exploitées.

Plan du mémoire

Le présent mémoire d'habilitation synthétise l'ensemble de mes travaux dans le domaine de la classification non supervisée collaborative ainsi que dans l'intégration de connaissances dans le processus d'extraction d'informations, et plus précisément de classification automatique d'images. Les travaux présentés sont issus principalement de mon implication en tant qu'encadrant dans les thèses de Sébastien Derivaux et de Germain Forestier, ainsi que dans les différents projets de recherche auxquels j'ai participé (ces projets sont décrits de manière plus précise en section 1.2.2). Le manuscrit présente mes contributions dans ces deux domaines particuliers et se décompose de la manière suivante.

Tout d'abord, le premier chapitre précise le contexte dans lequel se sont déroulés mes travaux de recherche. Il présente l'historique de mes activités de recherche ainsi que leur évolution thématique. La suite du chapitre comporte un *curriculum vitae* étendu de mes autres activités d'enseignant-chercheur, à savoir mes activités d'enseignement, mes charges administratives, ainsi que mes activités d'animation de la recherche (encadrement de jeunes chercheurs, implication dans la communauté scientifique, contrats et projets de recherche). Enfin, une liste de mes publications vient clore ce chapitre.

Le second chapitre présente les contributions théoriques de nos travaux dans le domaine de la collaboration de méthodes de clustering. Le contexte de l'apparition de ces métaméthodes de clustering collaboratif est précisé en introduction, en présentant les enjeux et objectifs du clustering, ainsi que les limites des méthodes classiques. S'en suit un état de l'art et une étude comparative des différentes méthodes existantes de combinaison de clustering, ainsi qu'une présentation de nos contributions dans ce domaine. Ensuite, nous montrons comment la collaboration avec des méthodes de clustering peut enrichir un processus de classification semi-supervisée. Finalement, nous présentons une étude sur l'utilisation de la connaissance dans le processus de clustering, et dans un processus de clustering collaboratif. Cette étude est complétée par une réflexion sur

l'évaluation d'un résultat de clustering en fonction des connaissances et une comparaison des différents critères de qualité existant.

Le troisième chapitre quant à lui, s'intéresse à l'application de nos méthodes dans le cadre de la classification automatique d'images, et plus particulièrement d'images de télédétection. La première section présente nos choix sur la représentation des connaissances expertes dans ce domaine. En collaboration avec des géographes, une ontologie d'objets géographiques a été définie, complétée par un processus d'appariement qui permet d'effectuer la comparaison entre une région construite lors d'une segmentation et les différents concepts définis dans l'ontologie. Ensuite, nous présentons comment nous avons proposé d'intégrer cette connaissance dans les différents processus d'extraction d'informations à partir d'une image, à savoir le clustering, la segmentation puis la classification ou la détection d'objets particuliers.

Enfin nous concluons dans le dernier chapitre, sur les apports de nos travaux dans le domaine de la collaboration de méthodes de clustering ainsi que l'extraction d'informations à partir de données complexes, guidées par des connaissances du domaine, et nous proposons plusieurs pistes de recherche à moyen et plus long terme, ouvertes par ces travaux.

Table des matières

1 Contexte	1
1.1 Thématiques de recherche	1
1.2 Curriculum vitae	2
1.2.1 Enseignement	3
1.2.2 Recherche	5
1.2.3 Liste des publications	10
2 Clustering collaboratif	17
2.1 Collaboration entre méthodes de clustering	18
2.1.1 Combinaison de résultats de clustering	18
2.1.2 Approches multiobjectives	26
2.1.3 Approche collaborative SAMARAH	27
2.2 Collaboration entre clustering et classification semi-supervisée	34
2.2.1 Méthodes semi-supervisées	36
2.2.2 Apprentissage semi-supervisé enrichi par de multiples clusterings . .	41
2.3 Connaissances et clustering	44
2.3.1 Utilisation de connaissances en clustering	46
2.3.2 Évaluation d'un clustering	47
2.3.3 Évaluation des différents critères de qualité	51
2.3.4 Utilisation de la pureté pour guider un clustering collaboratif . . .	53
2.4 Contributions et valorisation	57
3 Connaissances expertes en observation de la Terre	59
3.1 Ontologie d'objets géographiques	60
3.1.1 Description de l'ontologie	60
3.1.2 Appariement de région	61
3.2 Connaissances et segmentation	63
3.2.1 Optimisation du paramétrage d'une segmentation	66
3.2.2 Segmentation supervisée	69
3.2.3 Approche hybride : optimisation de segmentation supervisée . . .	72
3.2.4 Comparatif des méthodes	75
3.3 Connaissances et classification	75
3.3.1 Problématique	76
3.3.2 Caractérisation des régions	77
3.3.3 Critères d'évaluation	78
3.3.4 Méthode de classification par connaissances du domaine	79
3.4 Connaissances et détection	81
3.4.1 Interprétation par ensemble de détecteurs	82
3.4.2 Extraction de détecteurs spécifiques à partir de la base de connaissances	83

TABLE DES MATIÈRES

3.5	Connaissances et clustering collaboratif	86
3.5.1	Problématique	86
3.5.2	Étiquetage des clusters	86
3.6	Contributions et valorisation	88
4	Conclusion et perspectives	91
4.1	Conclusion	91
4.2	Perspectives	92
4.2.1	Application au domaine des images microscopiques	93
4.2.2	Vers un cadre générique de représentation de la connaissance	94
A	Méthodes de classification semi-supervisée	111
A.1	Static Labeling	112
A.2	Dynamic labeling	112
A.3	Étiquetage des clusters à la majorité	112
A.4	Single Representative Insertion/Deletion Steepest Descent Hill Climbing with Randomized Restart	113
A.5	Supervised Clustering using Evolutionary Computing	114
A.6	Refined clustering	115
A.7	Seeded-Kmeans	116
A.8	Constrained-Kmeans	116
B	Données et résultats pour la méthode SLEM	119
B.1	Données utilisées	119
B.2	Résultats comparatifs	120
C	Articles principaux	127

Une certaine identification du scientifique et de l'objet de son étude est non seulement tolérable mais souhaitable. L'IA est la première science qui a illustré cette position, d'où son odeur de soufre.

Yves Kodratoff (1986)

Contexte

1.1 Thématiques de recherche	1
1.2 Curriculum vitae	2
1.2.1 Enseignement	3
1.2.2 Recherche	5
1.2.3 Liste des publications	10

1.1 Thématiques de recherche

Mes travaux de recherche ont débuté en 1996 lors d'un stage qui avait pour objet d'étudier et de proposer une solution de collaboration entre méthodes de classification non supervisée. L'idée était d'utiliser le paradigme multi-agents pour représenter des instances de méthode de classification via des agents intelligents et modéliser un comportement de collaboration entre eux. Cette première étude de faisabilité a été concrétisée et validée sur la classification automatique d'images de télédétection lors de mes travaux de thèse qui ont suivi (Wemmert et al. 2000, Gançarski & Wemmert 2007).

Depuis ma prise de fonction en tant qu'enseignant-chercheur en 2001, j'ai eu la chance de pouvoir continuer à travailler dans la même équipe de recherche et l'opportunité d'encadrer deux thèses. Celles-ci ont permis d'une part d'étendre le système de clustering collaboratif développé précédemment (Forestier et al. 2008a) et d'y intégrer des mécanismes de gestion des connaissances expertes (Forestier, Gançarski & Wemmert 2010). D'autre part, l'utilisation de connaissances de haut-niveau a été étudiée et mise en œuvre dans le cas particulier de la classification automatique d'images de télédétection (Wemmert, Puissant, Forestier & Gançarski 2009, Derivaux et al. 2010, Forestier et al. 2012).

Je poursuis actuellement mes recherches selon plusieurs axes. Tout d'abord, je m'intéresse à l'aspect théorique de la collaboration entre méthodes de classification semi-supervisées et de clustering supervisé.

Parallèlement, je continue à m'intéresser à l'intégration de connaissances dans le processus d'extraction d'information à partir d'images avec notamment l'intégration de contraintes spatiales dans le processus d'extraction.

CHAPITRE 1. CONTEXTE

Enfin, j'étends mon champ d'application en validant les approches développées dans d'autres domaines comme celui des images cellulaires microscopique ou encore des images thermiques de bâtiments.

1.2 Curriculum vitae

État civil

Nom et prénom : WEMMERT Cédric
Date et lieu de naissance : 13 août 1973 à Strasbourg
Nationalité : Française
Situation familiale : Marié, 2 enfants
Adresse professionnelle : LSIIT - UMR 7005
Pôle API - Bd Sébastien Brant
BP 10413
67412 Illkirch CEDEX
Téléphone : +33 (0) 368 854 581
Adresse électronique : wemmert@unistra.fr
Page personnelle : lsiit-cnrs.unistra.fr/bfo-fr/index.php/Cedric_Wemmert

Situation actuelle (depuis septembre 2001)

Maître de Conférences en Informatique à l'Université de Strasbourg (UdS) :

Enseignant à l'IUT Robert Schuman, Département Informatique
Chercheur au Laboratoire des Sciences de l'Image, de l'Informatique
et de la Télédétection
(LSIIT - UMR 7005 - CNRS / UdS), dans l'équipe BFO (Bioinformatique théorique, Fouille de données et Optimisation stochastique)

Formation Universitaire

- 1997-2000 **Thèse de doctorat : Classification hybride distribuée par collaboration de méthodes non supervisées**
Université Louis Pasteur (Strasbourg) - 14 décembre 2000
- directeur de thèse : Jerzy Korczak, Professeur à l'ULP Strasbourg
- co-encadrant : Pierre Gançarski, Maître de conférences à l'ULP Strasbourg
- 1995-1996 **Diplôme d'Études Approfondies en Informatique Générale**, mention bien
Université Louis Pasteur (Strasbourg)
Mémoire : Une architecture multi-agents pour la classification hybride en télédétection
Laboratoire d'accueil : LSIIT - UMR 7005 - CNRS/ULP (Strasbourg)

Thèmes de recherche

- clustering collaboratif, extraction d'informations à partir d'images

Enseignements

- programmation objets, architectures multi-tiers, fouille de données, programmation en géomatique

Publications

7 revues internationales, 1 revue nationale, 2 chapitres de livre, 20 manifestations internationales, 11 manifestations nationales. La liste complète de mes publications est donnée en fin de la section 1.2.3.

1.2.1 Enseignement

J'effectue la majorité de mon enseignement à l'IUT Robert Schuman, principalement en licence professionnelle Concepteur Développeur en Environnement Distribué. En effet, cette licence professionnelle a été créée lors de mon recrutement et j'en ai pris la responsabilité pédagogique dès ma titularisation.

J'ai cependant toujours veillé à avoir un service d'enseignement le plus large possible et à enseigner dans différentes formations de l'Université. Ainsi, j'enseigne actuellement

CHAPITRE 1. CONTEXTE

en DUT informatique, Licence professionnelle CDED (Concepteur Développeur en Environnement Distribué) et Licence professionnelle QCI (Qualification Complémentaire en Informatique) au sein de ma composante de rattachement. J'effectue aussi des enseignements en Master ILC (Ingénierie des Logiciels et des Connaissances - UFR de Mathématique et Informatique), Master OTG (Observation de la Terre et Géomatique - UFR de Géographie), ainsi qu'en deuxième année de l'École Nationale Supérieure d'Informatique pour l'Industrie et l'Entreprise (ENSIIE).

Ce large spectre me permet d'appréhender tous les aspects de l'offre de formation de l'Université en Informatique. De plus, j'ai toujours essayé de varier les matières enseignées comme le montre le tableau ci-dessous.

Module	Formation	Période	Volume annuel
Programmation C	DUT informatique	2001 - 2004	40h
Systèmes d'exploitation	DUT Informatique	2001 - 2008	40h
Réseaux informatiques	DUT Informatique	2006 - 2009	40h
Programmation iPhone	DUT Informatique	2011 -	35h
Programmation objets	LP CDED	2001 -	50h
	LP QCI	2010 -	50h
Architectures multi-tiers	LP CDED	2001 -	70h
Modélisation UML	LP CDED	2001 -	25h
Programmation web	LP QCI	2008 - 2009	50h
Algorithmique et programmation	Master OTG	2009 -	30h
Programmation distribuée	ENSIIE	2010 -	25h
Intelligence artificielle	Maîtrise d'Informatique	2002 - 2006	20h
Fouille de données	Master ILC	2008 -	15h

Ainsi j'enseigne ou ai enseigné des matières très techniques (système d'exploitation, réseaux, etc.), des matières plus conceptuelles (modélisation et analyse UML), et aussi des matières liées à mon domaine de recherche (fouille de données, intelligence artificielle).

Enfin, je participe à l'encadrement de nombreux projets : projets tuteurés de DUT informatique (2 équipes de 3 étudiants par an) et de Licence professionnelle CDED (2 équipes de 8 étudiants par an), projets TER en Master ILC, projets 150h en Licence et Master d'informatique.

Responsabilités pédagogiques Dès ma titularisation, j'ai pris en charge des responsabilités pédagogiques. Ainsi, j'ai assuré la responsabilité pédagogique de la licence professionnelle CDED de 2002 à 2008. Depuis 2009, je suis responsable pédagogique de la licence professionnelle QCI en formation continue à temps partagé. Enfin, depuis septembre 2011, je suis aussi co-responsable du Master ILC de l'Université de Strasbourg. Du fait de ces responsabilités, j'ai participé à la rédaction des 3 dossiers d'habilitation de ces diplômes.

Fonctions d'intérêt collectif

- Membre élu du Conseil de Laboratoire du LSIIT depuis 2009
- Membre élu du Comité d'Experts 27ème section de l'Université de Strasbourg depuis 2009
- Représentant de l'équipe BFO au Conseil Scientifique du Département Informatique du futur institut ST2I (ICube) depuis 2011
- Vice président MCF élu de la Commission de Spécialistes 27ème section de l'Université Robert Schuman de 2004 à 2008
- Membre du comité Recherche de l'IUT Robert Schuman depuis 2009

1.2.2 Recherche

Je fais partie de la thématique Fouille de données de l'équipe Bioinformatique théorique, Fouille de données et Optimisation stochastique du LSIIT. Mes travaux de recherche portent principalement sur la fouille de données multistratégie. Il s'agit d'étudier des mécanismes de collaboration entre méthodes de classification existantes et plus précisément de combinaison de classificateurs non supervisés. Plus récemment, je m'intéresse à l'apport de connaissances de haut niveau sémantique au processus, afin de le guider.

Ces travaux sont réalisés en grande partie en collaboration avec le Laboratoire Image, Ville, Environnement (ERL 7230) et ont été validés dans le cadre de la télédétection. Ainsi, le domaine d'application principal de mes méthodes est la classification automatique d'images de télédétection et plus largement l'extraction automatique d'informations à partir d'images.

Activité d'encadrement

Durant ma carrière, j'ai eu l'opportunité d'encadrer plusieurs étudiants en stage de Master recherche et en thèse, et des chercheurs post-doctorants. L'ensemble de ces activités est résumé dans le tableau ci-dessous.

<i>Personne en- cadrée</i>	<i>Période</i>	<i>Taux</i>	<i>Co-encadrant(s)</i>
Thèses			
1. Sébastien Derivaux			
	2005-2009	33%	J.J. Korczak, S. Lefèvre
<i>Intégration de connaissances dans la construction et la classification d'objets d'image de télédétection</i>			
2. Germain Forestier	2007-2010	80%	P. Gançarski
<i>Connaissances et clustering collaboratif de données complexes</i>			

CHAPITRE 1. CONTEXTE

3. Aymen Sel- 2010- **33%** K. Bsaises, A. Deruyver laouti
Méthode collaborative de segmentation et classification d'objets à partir d'images de télédétection à très haute résolution spatiale (co-tutelle avec l'Université de Tunis)
 4. Bruno Belarte 2011- **60%** P. Gançarski, C. Weber
Extraction, analyse et utilisation de relations spatiales entre objets d'intérêt pour une analyse d'images de télédétection guidée par des connaissances du domaine
-

Post-doctorats

1. Nicolas Durand 2008-2009 **30%** P. Gançarski, A. Puissant
Navigation et appariement d'objets géographiques dans une ontologie
 2. Juliane Krueger 2010-2012 **80%** P. Gançarski
Caractérisation automatique des relations spatiales entre cellules dans des images microscopiques à très haute résolution
-

Stages de Master

1. Anthony Sutton 2001 **50%** P. Gançarski
Négociation entre agents dans un système de classification hybride non supervisée (cas de la télédétection)
 2. Alexandre Blansché 2003 **30%** P. Gançarski
Sélection automatique d'attributs et combinaison de classifieurs pour des objets complexes
 3. Dia Matar 2004 **50%** C. Weber, P. Gançarski
Intégration de connaissances lors du processus d'apprentissage hybride : cas des régions d'intérêt
 4. Mansour Beye 2004 **30%** C. Weber, P. Gançarski
Utilisation d'un expert lors du processus d'apprentissage hybride
 5. Pierre Chris- 2005 **50%** P. Gançarski tensen
Détection et utilisation de textures pour la classification d'objets dans des images hyperspectrales
 7. Germain Fores- 2007 **50%** P. Gançarski tier
Classification collaborative et intégration de connaissances
 8. Mickaël Fro- 2009 **50%** G. Forestier meyer
Collaboration de méthodes de classification non supervisées
 9. Bruno Belarte 2010 **100%**
-

Extraction et analyse de relations spatiales entre objets d'intérêt dans les images de télédétection guidées par des connaissances du domaine

10. Maurice Lan- 2010 **100%**
selle

Opportunités d'exploitation de l'inférence évidentielle en classification collaborative

Activités contractuelles

Je participe et ai participé très activement à la rédaction, l'animation et la réalisation de plusieurs projets de recherche. L'ensemble des projets est listé ci-dessous avec un bref descriptif de ma contribution dans le projet.

FOSTER : projet ANR Cosinus (2011 - 2014)

- taux de participation : 30%
- budget total : 900 000€- LSIIT-BFO : 218 000€
- participants : LSIIT Strasbourg, PPME Nouvelle Calédonie, LIRIS Lyon, LISTIC Annecy, Bluecham SAS

Fouille de données spatio-temporelles : application à la compréhension et à la surveillance de l'érosion. Ce projet a pour objectif de concevoir, développer et mettre en oeuvre des nouveaux processus d'analyse adaptés aux masses de données spatio-temporelles (MNT, images satellites, données météorologiques,...) dans l'optique d'une gestion améliorée de l'environnement. Deux tâches critiques de ce processus seront plus particulièrement étudiées : la segmentation des images satellitaires basée sur des méthodes collaboratives, et la construction de modèles descriptifs (motifs, clustering, ...) et/ou prédictifs (arbres de décision,...) intégrant de l'information spatio-temporelle. Mon rôle consiste notamment en la responsabilité de la tâche concernant les approches collaboratives en classification et segmentation d'images.

Roche Diagnostics GmbH : Histopathological image analysis (2010 - 2012)

- contrat de Recherche et Développement
- budget : 30 000€

Ce projet concerne l'analyse d'images histopathologiques pour la validation de l'efficacité d'une molécule anti-cancéreuse de chimio-thérapie. L'objectif est d'extraire des images et de classer automatiquement les différentes cellules présentes, puis d'en étudier les relations spatiales. Mon rôle consiste à piloter le projet et encadrer une chercheuse en post-doctorat (Juliane Krueger) basée dans l'équipe Experimental Pathology, TRS du service Pharma Research and Early Development (pRED) de la société Roche Diagnostics GmbH à Penzberg (Allemagne).

CHAPITRE 1. CONTEXTE

CNES : Etude ORFEO GT3 Extraction et analyse de relations spatiales entre objets d'intérêt dans les images de télédétection guidées par des connaissances du domaine (2010 - 2012)

- responsable scientifique de l'étude
- budget : 15 000€

L'environnement proposé dans le cadre de cette étude devra permettre une utilisation conjointe d'un ensemble de données multisources, hétérogènes et complexes (optique, photo, altitude, multi-résolution). L'objectif est de parvenir à extraire, modéliser et utiliser des informations sur des relations spatiales entre objets d'intérêt pour guider le processus d'extraction d'information et de classification d'images à partir de cet ensemble de données. Pour cette étude un apprenti du Master Ingénierie des Logiciels et des Connaissances a été embauché et j'en ai assuré l'encadrement à 100%.

FRESQUEAU : projet ANR Modèles numériques (2011 - 2014)

- taux de participation : 15%
- budget total : 850 000€ - LSIIT-BFO : 278 000€
- partenaires : LSIIT et LHYGES Strasbourg - TETIS Paris - LIRMM et AQUASCOP Montpellier - AQUABIO Saint-Germain-du-Puch

L'objectif du projet est de répondre à deux enjeux spécifiques : (1) approfondir la connaissance du fonctionnement des cours d'eau par l'analyse des taxons à la base des indices biologiques (2) relier les sources de pressions sur le milieu à la qualité physico-chimique et biologique des cours d'eau. Je participe principalement à la tâche 3 de ce projet qui a pour objectif d'étudier la complémentarité et la combinaison des méthodes de fouilles de données proposées pour traiter les données du projet.

CNES : contrat n° 70904/00 Système interactif d'aide à l'interprétation d'images (2007 - 2008)

- co-responsable du projet avec Pierre Gançarski (LSIIT) et Anne Puissant (LIVE Strasbourg)
- budget : 63 000€

L'objectif de ce projet était de fournir un système interactif d'aide à l'interprétation d'images satellite. De plus, une méthode générique permettant l'utilisation conjointe d'images de différents types et différentes résolutions a été développée et testée. Les expérimentations se sont focalisées sur des données SPOT (haute résolution) et type Pléiades (très haute résolution). Ce contrat a permis l'embauche au sein du LSIIT-BFO d'un post-doctorant pendant 6 mois ainsi que d'une apprentie ingénieur de développement pendant 12 mois.

ECOSGIL : projet ANR Jeunes Chercheuses Jeunes Chercheurs (2005 - 2008)

- taux de participation : 60%

- responsable local (LSIIT) du projet
- budget : 55 000€
- partenaires : LSIIT et LIVE Strasbourg – IDEES/Geosyscom, GREYC DODOLA, LETG Geophen et M2C Caen

L'objectif de ce projet pluridisciplinaire est de développer une plateforme interactive d'archivage, d'identification, d'analyse et de représentation de plusieurs objets d'études environnementaux à partir de données d'information géographiques, en particulier des données images à très haute résolution. J'ai été responsable de la tâche *Développement de méthodes automatiques d'extraction*.

FODOMUST : projet ACI Masse de données (2004 - 2007)

- taux de participation : 40% de mon temps de recherche
- budget total : 420 000€ - LSIIT-BFO : 236 000€
- partenaires : LSIIT et LIVE Strasbourg – ERIC Lyon

Ce projet porte sur la fouille de données multistratégie pour extraire et qualifier la végétation urbaine à partir de bases de données d'images. J'ai travaillé plus particulièrement sur les axes *Construction des objets urbains* et *Classification multistratégie* avec notamment le co-encadrement de la thèse de Sébastien Derivaux. De plus, ce projet a permis d'obtenir une bourse ministérielle pour Germain Forestier qui a effectué sa thèse d'octobre 2007 à septembre 2010 et que j'ai co-encadrée.

Activités diverses

Relecture d'articles Je suis relecteur réguliers dans divers journaux et conférences internationales et nationales dont :

- *Journaux internationaux*
 - IEEE Transactions on Geoscience and Remote Sensing (TGRS)
 - IEEE Sensors Journal
 - IEEE Geoscience and Remote Sensing Letters (GRSL)
 - IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)
 - Pattern Recognition (Elsevier)
- *Conférences nationales et internationales*
 - IEEE International Geoscience and Remote Sensing Symposium - IGARSS 2008, 2009, 2010, 2011
 - 15th AGILE Conference on Geographic Information Science - AGILE 2012
 - Colloque International de Géomatique et d'Analyse Spatiale - SAGEO 2006, 2007, 2008, 2009, 2010, 2011
 - Conférence francophone sur l'Extraction et Gestion des Connaissances - EGC 2008, 2009, 2010, 2011
 - Reconnaissance de Formes et Intelligence Artificielle - RFIA 2012

CHAPITRE 1. CONTEXTE

- *Divers*

- numéros spéciaux *Fouille de données complexes* - revue RNTI 2005, 2010
- chapitre pour le livre : *Urban Remote Sensing: Monitoring, Synthesis and Modeling in the Urban Environment* - Éditions Wiley
- revue Télédétection - Éditions des archives contemporaines, Paris

Participation à l'organisation de conférences ou groupes de travail

- Journées Fouille et Visualisation de Données Massives - 18-19 juin 2012 (Tours)
- Atelier Fouille de données complexes - conférence EGC 2012 (Bordeaux)
- Journées Fouille de données complexes et de grands graphes - 21-22 juin 2011 (Paris)
- Atelier Fouille de données complexes - conférence EGC 2011 (Brest)
- Assises du GDR I3 (Information-Interaction-Intelligence) - 2010 (Strasbourg)
- Conférence francophone sur l'Extraction et Gestion des Connaissances - EGC 2009 (Strasbourg)
- Évolution Artificielle - EA 2009 (Strasbourg)
- Congrès de la Société des Personnels Enseignants et Chercheurs en Informatique de France - SPECIF 2008 (Strasbourg)
- 2ième Journées sur l'Information Géographique et l'Observation de la Terre - JIGOT 2008 (Marseille)
- Colloque International de Géomatique et d'Analyse Spatiale - SAGEO 2006 (Strasbourg)

Par ailleurs, je suis membre de plusieurs groupes de travail nationaux : ORFEO, GDR ISIS, GDR I3, GDR SIGMA.

1.2.3 Liste des publications

L'ensemble de mes publications sont données dans cette section. Celles marquées d'un symbole * sont des publications à l'interface entre la télédétection et l'informatique.

Revues internationales (7)

- [1]* *Knowledge-based region labeling for remote sensing image interpretation*
G. Forestier, A. Puissant, **C. Wemmert**, P. Gançarski
Computers, Environment and Urban Systems, Elsevier, 2012 - à paraître
- [2] *Collaborative clustering with background knowledge*
G. Forestier, P. Gançarski, **C. Wemmert**
Data & Knowledge Engineering, Vol. 69, Num. 2, Elsevier, pp 211–228 - 2010
- [3] *Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation*

- S. Derivaux, G. Forestier, **C. Wemmert**, S. Lefèvre
 Pattern Recognition Letters, Vol. 31, Num. 15, Elsevier, pp 2364–2374 - 2010
- [4]★ *Multiresolution Remote Sensing Image Clustering*
C. Wemmert, A. Puissant, G. Forestier, P. Gançarski
 IEEE Geoscience and Remote Sensing Letters, Vol. 6, Num. 3, IEEE Edition,
 pp 533 - 537 - July 2009
- [5] *Multi-source Images Analysis Using Collaborative Clustering*
 G. Forestier, **C. Wemmert**, P. Gançarski
 EURASIP Journal on Advances in Signal Processing - Special issue on
 Machine Learning in Image Processing, Vol. 2008, Hindawi (Article ID
 374095), pp 11 - 2008
- [6] *Collaborative Multi-step Mono-level Multi-strategy Classification*
 P. Gançarski, **C. Wemmert**
 Journal on Multimedia Tools and Applications, Vol. 35, Num. 1, Springer
 Ed., pp 1–27 - October 2007
- [7] *A Collaborative Approach to Combine Multiple Learning Methods*
C. Wemmert, P. Gançarski, J. Korczak
 International Journal on Artificial Intelligence Tools, Vol. 9, Num. 1, Ed.
 World Scientific, pp 59–78 - May 2000
- Communications à des manifestations internationales à comité de lecture (20)**
- [8] *Hierarchical Classification-based Region Growing (HCBRG) : A Collaborative Approach for Object Segmentation and Classification*
 A. Sellaouti, A. Hamouda, A. Deruyver, **C. Wemmert**
 International Conference on Image Analysis and Recognition (ICIAR),
 Aveiro, Portugal, June 2012
- [9] *Background knowledge integration in clustering using purity indexes*
 G. Forestier, **C. Wemmert**, P. Gançarski
 International Conference on Knowledge Science, Engineering & Management,
 Springer, Lecture Notes in Computer Science, pp 28–38, Vol. 6291,
 Belfast, Ireland - September 2010
- [10] *Towards conflict resolution in collaborative clustering*
 G. Forestier, **C. Wemmert**, P. Gançarski
 IEEE International Conference on Intelligent Systems, pp 361–366, Lon-
 don, Great-Britain - July 2010
- [11]★ *Mining Multiple Satellite Sensor Data Using Collaborative Clustering*

- G. Forestier, **C. Wemmert**, P. Gançarski, J. Inglada
Workshop on Mining Multiple Information Sources, pp 501–506, Miami,
USA - December 2009
- [12]* *Mining spectral libraries to study sensors' discrimination ability*
G. Forestier, J. Inglada, **C. Wemmert**, P. Gançarski
SPIE Europe Remote Sensing, pp 9 pages, Vol. 7478, Berlin, Germany - September 2009
- [13] *Semi-supervised collaborative clustering with partial background knowledge*
G. Forestier, **C. Wemmert**, P. Gançarski
Workshop on Mining Complex Data, pp 211–217, Pisa, Italy - December 2008
- [14]* *On Combining Unsupervised Classification and Ontology Knowledge*
G. Forestier, **C. Wemmert**, P. Gançarski
IEEE International Geoscience and Remote Sensing Symposium (IGARSS),
pp 395–398, Boston, USA - July 2008
- [15] *Improving Supervised Learning with Multiple Clusterings*
S. Derivaux, G. Forestier, **C. Wemmert**
Workshop on Supervised and Unsupervised Ensemble Methods and Their Applications in conjunction with ECAI, pp 57–60, Patras, Greece - July 2008
- [16] *An Evolutionary Approach for Ontology Driven Image Interpretation*
G. Forestier, S. Derivaux, **C. Wemmert**, P. Gançarski
Tenth European Workshop on Evolutionary Computation in Image Analysis and Signal Processing, Springer, Lecture Notes in Computer Sciences, pp 295–304, Vol. 4974, Napoli, Italy - March 2008
- [17]* *On the Complementarity of an Ontology and a Nearest Neighbour for Remotely Sensed Image Interpretation*
S. Derivaux, N. Durand, **C. Wemmert**
IEEE International Geosciences and Remote Sensing Symposium (IGARSS), pp 3983–3986, Barcelona, Spain - July 2007
- [18]* *Ontology-based Object Recognition for Remote Sensing Image Interpretation*
N. Durand, S. Derivaux, G. Forestier, **C. Wemmert**, P. Gançarski, O. Boussaïd, A. Puissant
IEEE International Conference on Tools with Artificial Intelligence, IEEE Computer Society, pp 472–479, Vol. 1, Patras, Greece - October 2007

- [19] *On Machine Learning In Watershed Segmentation*
 S. Derivaux, S. Lefèvre, **C. Wemmert**, J. Korczak
 IEEE International Workshop on Machine Learning for Signal Processing,
 pp 187–192, Thessaloniki, Greece - August 2007
- [20] *Collaborative Multi-Strategical Classification for Object-Oriented Image Analysis*
 G. Forestier, **C. Wemmert**, P. Gançarski
 Workshop on Supervised and Unsupervised Ensemble Methods and Their Applications in conjunction with IbPRIA, pp 80–90, Girona, Spain - June 2007
- [21]* *Watershed Segmentation of Remotely Sensed Images Based on a Supervised Fuzzy Pixel Classification*
 S. Derivaux, S. Lefèvre, **C. Wemmert**, J. Korczak
 IEEE International Geosciences And Remote Sensing Symposium (IGARSS) 2006, Denver, USA, pp 3712–3715 - July 2006
- [22]* *Deriving Classification Rules from Multiple Sensed Urban data with Data Mining*
 D. Sheeren, A. Puissant, C. Weber, P. Gançarski, **C. Wemmert**
 1rst Workshop of the EARSel Special Interest Group Urban Remote Sensing, 9 pages, Berlin, Germany - March 2006
- [23] *Collaborative Multi-strategy Classification : Application to per-pixel Analysis of Images*
 P. Gançarski, **C. Wemmert**
 ACM SIGKDD - Sixth International Workshop on Multimedia data mining, ACM Digital Library, pp 15–22, Chicago, USA - August 2005
- [24] *MuStICOS : Multi-Step Image Classification Operating System*
C. Wemmert, P. Gançarski
 IASTED Artificial Intelligence and Applications Conference, pp 23–32, Benalmadena, Spain - September 2003
- [25]* *Urban Thematical Zones Construction from Remote Sensing Data by Unsupervised Classification*
C. Wemmert, P. Gançarski
 23rd Symposium on Urban Data Management, 6 pages, Prague, Czech Republic - October 2002
- [26] *A Multi-View Voting Method to Combine Unsupervised Classifications*
C. Wemmert, P. Gançarski

CHAPITRE 1. CONTEXTE

Artificial Intelligence and Applications, pp 447–452, Malaga, Spain - September 2002

- [27] *An unsupervised collaborative learning method to refine classification hierarchies*
C. Wemmert, P. Gançarski, J. Korczak
11th IEEE International Conference on Tools with Artificial Intelligence, IEEE Computer Society, pp 465–470, Chicago, USA - November 1999

Revues nationales (1)

- [28] *SLEM C : Apprentissage semi-supervisé enrichi par de multiples clusterings*
C. Wemmert, G. Forestier
Revue des Nouvelles Technologies de l'Information, RNTI, numéro spécial "Fouille de données complexes", RNTI E.21, pp 147–169, 2011

Ouvrages ou participation à des ouvrages (2)

- [29] *Applications of Supervised and Unsupervised Ensemble Methods*
C. Wemmert, G. Forestier, S. Derivaux
Springer, chap. Improving Supervised learning with Multiple Clusterings, pp 135–148, Studies in Computational Intelligence, Vol. 245 - 2009
- [30] *Supervised and Unsupervised Ensemble Methods and their Applications*
G. Forestier, **C. Wemmert**, P. Gançarski
Springer, chap. Collaborative Multi-Strategical Clustering for Object-Oriented Image Analysis, pp 71–88, Studies in Computational Intelligence, Vol. 126/2008 - 2008

Communications à des manifestations nationales à comité de lecture (11)

- [31] *Comparaison de critères de pureté pour l'intégration de connaissances en clustering semi-supervisé*
G. Forestier, **C. Wemmert**, P. Gançarski
Journées Francophones Extraction et Gestion des Connaissances (EGC 2010), pp 127–132, Hammamet, Tunisie - Janvier 2010
- [32] *Étude de données multisources par simulation de capteurs et clustering collaboratif*
G. Forestier, **C. Wemmert**, P. Gançarski

- Atelier Fouille de données complexes, Journées Francophones Extraction et Gestion des Connaissances (EGC 2010), pp A143–A152, Hammamet, Tunisie - january 2010
- [33] * *Extraction de détecteurs d'objets urbains à partir d'une ontologie*
 S. Derivaux, G. Forestier, **C. Wemmert**, S. Lefèvre
 Atelier Extraction de Connaissance à partir d'Images (ECOI), Journées Francophones Extraction et Gestion des Connaissances (EGC 2008), pp 71–81, Sophia Antipolis, France - Janvier 2008
- [34] *Interprétation d'images basée sur une approche évolutive guidée par une ontologie*
 G. Forestier, S. Derivaux, **C. Wemmert**, P. Gançarski
 Journées Francophones Extraction et Gestion des Connaissances (EGC 2008), pp 469–474, Vol. 2, Sophia Antipolis, France - Janvier 2008
- [35] *Segmentation par ligne de partage des eaux basée sur des connaissances texturelles*
 S. Derivaux, S. Lefèvre, **C. Wemmert**, J. Korczak
 XXIème colloque GRETSI, Traitement du Signal et des Images, pp 913–916 - Septembre 2007
- [36] *Paramétrisation de méthodes de segmentation par utilisation de connaissances et approche génétique*
 S. Derivaux, **C. Wemmert**, S. Lefèvre, J. Korczak
 Atelier Extraction de Connaissance à partir d'Images (ECOI), Journées Francophones Extraction et Gestion des Connaissances (EGC), pp 11, Namur, Belgium - Janvier 2007
- [37] *Apport d'une classification non supervisée floue à la segmentation par ligne de partage des eaux*
 S. Derivaux, S. Lefèvre, **C. Wemmert**, J. Korczak
 Colloque International de Géomatique et d'Analyse Spatiale (SAGEO), Strasbourg, 2006, pp 11 - Septembre 2006
- [38] * *Amélioration des connaissances sur l'environnement urbain : intérêt de l'intégration de règles et de l'utilisation de classifications multi-formalismes*
 A. Puissant, D. Sheeren, C. Weber, **C. Wemmert**, P. Gançarski
 Colloque international Nature-Sociétés : Analyses et modèles, La Baule - Mai 2006
- [39] *Un système de classification hybride et son application à la télédétection*
C. Wemmert, P. Gançarski, J. Korczak

CHAPITRE 1. CONTEXTE

Société Francophone de Classification, SFC'99, pp 273-280, Nancy - Septembre 1999

- [40] *Un système de raffinement non-supervisé d'un ensemble de hiérarchies de classes*

C. Wemmert, P. Gançarski, J. Korczak

Conférence d'Apprentissage, CAP'99, pp 153-160, Palaiseau - Mai 1999

- [41] *Collaborations entre méthodes de classification*

C. Wemmert, P. Gançarski

Actes des cinquièmes rencontres de la Société Francophone de Classification, Lyon - Septembre 1997

Nous classifions trop et ne jouissons pas assez.

Okakura Kakuzo (1906)

2

Clustering collaboratif

2.1	Collaboration entre méthodes de clustering	18
2.1.1	Combinaison de résultats de clustering	18
2.1.2	Approches multiobjectives	26
2.1.3	Approche collaborative SAMARAH	27
2.2	Collaboration entre clustering et classification semi-supervisée	34
2.2.1	Méthodes semi-supervisées	36
2.2.2	Apprentissage semi-supervisé enrichi par de multiples clusterings .	41
2.3	Connaissances et clustering	44
2.3.1	Utilisation de connaissances en clustering	46
2.3.2	Évaluation d'un clustering	47
2.3.3	Évaluation des différents critères de qualité	51
2.3.4	Utilisation de la pureté pour guider un clustering collaboratif . . .	53
2.4	Contributions et valorisation	57

Un nombre important de nouveaux algorithmes de clustering a été développé ces dernières années, et des méthodes existantes ont également été modifiées et améliorées. Cette abondance de méthodes peut être expliquée par la difficulté de proposer des méthodes génériques s'adaptant à tous les types de données disponibles. En effet, chaque méthode comporte un biais induit par l'objectif choisi pour créer les clusters. Par conséquent, deux méthodes différentes peuvent proposer des résultats de clustering très différents à partir des mêmes données. De plus, le même algorithme peut fournir des résultats différents en fonction de son initialisation ou de ses paramètres.

Pour résoudre ce problème, certaines méthodes proposent d'utiliser plusieurs résultats de clustering différents pour mieux refléter la diversité potentielle des résultats. Ces approches tirent parti des informations fournies par les différents résultats de manière sensiblement différente.

L'objectif de ce chapitre est de situer nos travaux sur l'apport de la collaboration de méthodes de clustering, à la fois dans le cadre totalement non supervisé (section 2.1)

et semi-supervisé (2.2). De plus, nous présentons aussi nos apports dans le cadre de l'intégration de connaissances expertes dans le processus de clustering (section 2.3).

2.1 Collaboration entre méthodes de clustering

La collaboration entre méthodes de clustering peut mener à deux types de résultats, soit un partitionnement unique des données, soit un ensemble de résultats de clustering. Le premier cas qui est le plus étudié actuellement, demande la mise en œuvre de techniques de fusion ou de combinaison de résultats de clustering. Le second cas représente les méthodes dites de *clustering multi-objectif* qui consistent à optimiser simultanément plusieurs critères plutôt qu'un seul puis à donner l'ensemble des résultats parmi lesquels il faudra choisir celui proposant un compromis optimal entre les critères à optimiser.

Les méthodes de combinaisons de décisions issues de plusieurs méthodes de classification non supervisée s'inspirent du travail important mené dans le domaine de la combinaison de méthodes supervisées (Kuncheva 2004, Kittler et al. 1998). Dans le cadre de la combinaison supervisée, le travail est simplifié par l'existence d'une référence commune de classe. Il est alors possible d'effectuer un vote à la majorité parmi les différentes classes proposées par les classificateurs pour une instance.

Cependant, dans le cadre de la collaboration entre méthodes non supervisées, il n'existe pas de liens évidents entre les clusters des différents résultats et les résultats n'ont pas forcément le même nombre de clusters. D'autres approches ont donc été envisagées avec l'utilisation d'hypergraphes ou des méthodes de ré-étiquetage. Ces méthodes de classification par ensemble s'intéressent principalement à fusionner les partitions créées en utilisant uniquement les résultats des clusterings initiaux, c'est à dire que le processus de fusion n'a plus accès aux données mais uniquement à l'affectation des clusters aux objets. C'est pourquoi nous avons proposé une méthode de clustering collaboratif qui remet en cause le résultat de chacun des clusterings et tire ainsi parti des informations sur les données et sur les partitions proposées par toutes les méthodes.

Dans cette section, nous définissons la problématique liée à la classification par ensemble de classificateurs et exposons un aperçu des méthodes de création d'un consensus à partir d'un ensemble de résultats de classification non supervisée. Nous présentons ensuite succinctement les méthodes de clustering multi-objectives, et finalement, notre méthode de collaboration de clusterings, SAMARAH.

2.1.1 Combinaison de résultats de clustering

- Soit $X = \{x_1, \dots, x_n\}$ l'ensemble des n objets à classer ;
- Soit $\mathbb{C} = \{\mathcal{C}^{(1)}, \dots, \mathcal{C}^{(N)}\}$ un ensemble de N résultats de clustering de X ;
- Soit $\mathcal{C}^{(i)} = \{C_1^{(i)}, \dots, C_{K^{(i)}}^{(i)}\}$ un résultat de clustering de X en $K^{(i)}$ clusters ;
- Soit $n_k^{(i)} = |C_k^{(i)}|$ le nombre d'objets du cluster $C_k^{(i)}$;
- Soit $\alpha_{k,l}^{(i,j)} = |C_k^{(i)} \cap C_l^{(j)}|$ le nombre d'objets commun au cluster $C_k^{(i)}$ et au cluster $C_l^{(j)}$.

La classification par ensemble peut être formellement définie comme la recherche d'un résultat de clustering $\mathcal{C}^{(*)}$ représentant au mieux la structure des données X , à partir de l'ensemble des informations disponibles dans les N résultats de \mathbb{C} . Pour cela, il est nécessaire de définir une fonction Φ prenant en entrée les N résultats de \mathbb{C} et produisant un résultat de clustering $\mathcal{C}^{(*)}$ appelé communément *consensus* :

$$\Phi : \mathbb{C} \rightarrow \mathcal{C}^{(*)} \quad [2.1]$$

Le résultat $\mathcal{C}^{(*)}$ peut être vu comme une moyenne des résultats de l'ensemble, et donc comme étant le clustering le plus similaire (selon une fonction de similarité *sim*) à l'ensemble des N résultats de \mathbb{C} parmi l'ensemble des résultats de clustering possibles $\check{\mathbb{C}} = \{\check{\mathcal{C}}^{(1)}, \check{\mathcal{C}}^{(2)}, \dots, \check{\mathcal{C}}^{(m)}\}$ de X :

$$\mathcal{C}^{(*)} = \arg \max_{\check{\mathcal{C}}^{(i)} \in \check{\mathbb{C}}} \sum_{q=1}^N \text{sim}(\check{\mathcal{C}}^{(i)}, \mathcal{C}^{(q)}) \quad [2.2]$$

Il est cependant impossible d'effectuer une recherche exhaustive parmi tous les résultats possibles $\check{\mathbb{C}} = \{\check{\mathcal{C}}^{(1)}, \check{\mathcal{C}}^{(2)}, \dots, \check{\mathcal{C}}^{(m)}\}$ leur nombre étant dissuasif : $m = \frac{1}{K!} \sum_{k=1}^K \binom{K}{k} (-1)^{K-k} k^n$ (par exemple 171 798 901 possibilités de former 4 groupes à partir de 16 objets). De plus, quand ce problème est vu sous la forme d'un problème d'optimisation, la recherche d'un consensus devient un problème NP-complet (Megiddo & Supowit 1984) car se rapportant à un problème de coloration de graphe.

Les travaux sur le clustering par ensemble s'intéressent essentiellement à deux aspects. Le premier est l'étude des méthodes permettant de générer les différents résultats composant \mathbb{C} . Le second est la définition de la fonction Φ permettant de générer le consensus. Dans la suite, nous présentons uniquement les différentes approches abordées dans la littérature pour définir la fonction Φ permettant de trouver un consensus seront présentées, classées de la manière suivante :

- méthodes basées sur les graphes,
- méthodes basées sur une matrice de co-association,
- méthodes basées sur la définition d'un nouvel espace de définition,
- méthodes basées sur le ré-étiquetage et le vote,
- méthodes spécifiques au clustering flou.

Méthodes de combinaison basées sur les graphes

L'une des premières approches pour le clustering par ensemble a été proposée par (Strehl & Ghosh 2002) qui introduisent différents cadres d'application de ces méthodes et décrivent trois algorithmes permettant de calculer un consensus à partir d'un ensemble de résultats. La recherche d'un consensus est définie comme la recherche d'un résultat $\mathcal{C}^{(*)}$ qui partage le plus d'information avec les résultats de l'ensemble \mathbb{C} . Pour quantifier ce partage d'information, les auteurs utilisent la notion d'information mutuelle (*mutual information*), qui est une mesure symétrique pour quantifier statistiquement l'information partagée par deux distributions (Cover & Thomas 2006). L'objectif du clustering par

ensemble tel que défini par (Strehl & Ghosh 2002) est de trouver le consensus maximisant l'information mutuelle normalisée (NMI) avec les membres de l'ensemble :

$$\mathcal{C}^{(*)} = \arg \max_{\check{\mathcal{C}}^{(i)} \in \check{\mathbb{C}}} \sum_{q=1}^N NMI(\check{\mathcal{C}}^{(i)}, \mathcal{C}^{(q)}) \quad [2.3]$$

Pour trouver ce consensus, les trois algorithmes proposés utilisent la notion d'hypergraphe pour représenter l'ensemble des résultats de clustering. Un hypergraphe est un ensemble de sommets et d'hyper-arêtes, une hyper-arête étant une généralisation du concept d'arête pouvant être connectée à un ensemble de sommets. Pour chaque résultat $\mathcal{C}^{(i)} \in \mathbb{C}$, une matrice binaire d'appartenance $H^{(i)}$ est créée, composée d'une colonne pour chaque cluster du résultat (voir exemple tableau 2.1). La concaténation de l'ensemble des matrices $H = (H^{(1)} \dots H^{(N)})$ représente la matrice d'adjacence d'un hypergraphe à N sommets et $\sum_{i=1}^N K^{(i)}$ hyper-arêtes. Chaque colonne h_i définit une hyper-arête où 1 indique que cette hyper-arête contient le sommet et 0 qu'elle ne le contient pas. Les trois algorithmes proposés utilisent cette représentation.

La première méthode appelée CSPA (*Cluster-based Similarity Partitioning Algorithm*), est basée sur la création d'une mesure de similarité entre les objets à partir des informations contenues dans les résultats de l'ensemble. Pour représenter cette similarité, une matrice S de taille $n \times n$ (n étant le nombre d'objets) est calculée à partir de la matrice H tel que $S = \frac{1}{N} HH^T$. Cette matrice est ensuite utilisée dans un algorithme de clustering classique pour générer le consensus.

La seconde approche appelée HPGA (*HyperGraph Partitioning Algorithm*), utilise directement la méthode de partitionnement d'hypergraphe METIS (Karypis et al. 1997) pour partitionner l'hypergraphe H représentant l'ensemble des clusterings. L'objectif de cette méthode est de créer un clustering consensus qui coupe le moins d'hyper-arêtes.

Enfin, la troisième approche appelée MCLA (*Meta-Clustering Algorithm*) crée des méta-groupes au sein de l'hypergraphe des clusterings en fusionnant des clusters similaires. Ces méta-groupes sont ensuite utilisés pour déterminer le clustering final. La similarité entre deux clusters est calculée à partir du nombre d'instances qui sont regroupées ensemble dans ces deux clusters en utilisant l'indice de Jaccard (voir Annexe B). Le graphe où les clusters ont été fusionnés est ensuite partitionné en utilisant la méthode de partitionnement d'hypergraphe METIS. Un vecteur d'association entre les instances et les clusters est créé au sein de chaque méta-groupe. Les instances sont ensuite classées dans le meta-cluster ayant le plus fort degré d'association.

Dans une autre approche proposée par (Fern & Brodley 2004), nommée HBGF (*Hybrid Bipartite Graph Formulation*), le problème est ramené à trouver une partition d'un graphe bipartite pour la formation du consensus. L'algorithme CSPA modélise l'ensemble comme un graphe dont les sommets représentent les instances, alors que l'algorithme MCLA modélise l'ensemble comme un graphe de clusters. L'algorithme HBGF combine ces deux approches et représente l'ensemble par un graphe bipartite où les instances ainsi que les clusters forment des sommets. Le graphe est bipartite car il n'y a pas d'arête entre les deux différents types de sommets (instance et cluster). Des règles sont définies pour

2.1. COLLABORATION ENTRE MÉTHODES DE CLUSTERING

	$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$
x_1	1	3	1
x_2	1	3	1
x_3	2	1	1
x_4	2	1	1
x_5	3	2	2
x_6	3	2	2

Résultats de clustering

	$H^{(1)}$			$H^{(2)}$			$H^{(3)}$	
	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
x_1	1	0	0	0	0	1	1	0
x_2	1	0	0	0	0	1	1	0
x_3	0	1	0	1	0	0	1	0
x_4	0	1	0	1	0	0	1	0
x_5	0	0	1	0	1	0	0	1
x_6	0	0	1	0	1	0	0	1

Matrice de transition du graphe

Tableau 2.1

Exemple de représentation par hypergraphe de trois résultats de clustering

affecter un poids $W(i, j)$ entre deux sommets (i, j) , sur les arêtes du graphe :

- $W(i, j) = 0$ si i et j sont deux clusters ou deux instances
- $W(i, j) = 0$ si l'instance i n'appartient pas au cluster j
- $W(i, j) = 1$ si l'instance i appartient au cluster j

Le graphe bipartite est ensuite partitionné pour former le consensus. Cette classification est effectuée en utilisant l'algorithme METIS ou une méthode de clustering spectral (Dhillon 2001).

Méthodes de combinaison basées sur la matrice de co-association

Une autre approche dans le clustering par ensemble est présentée par (Fred & Jain 2005), qui introduisent le concept d'accumulation de preuves (*evidence accumulation*). L'idée principale de cette approche est d'utiliser la *matrice de co-association* calculée à partir des résultats de l'ensemble. Cette matrice, de taille $n \times n$ (n étant le nombre d'objets), donne l'information du nombre de fois où deux objets ont été classés ensemble dans le même cluster dans les différents résultats de l'ensemble. Un exemple de calcul de cette matrice est donné dans le tableau 2.2. La matrice de co-association contient à l'indice (i, j) le nombre de fois où les objets (x_i, x_j) ont été classés ensemble dans les différents résultats :

$$\text{co-assoc}(i, j) = \frac{1}{K} \sum_{k=0}^K V(\mathcal{C}^{(k)}, x_i, x_j) \quad (2.4)$$

où $V(.)$ renvoie 1 si les instances (x_i, x_j) ont été classées ensemble dans le résultat $\mathcal{C}^{(k)}$ et 0 sinon. Une méthode de clustering hiérarchique est ensuite utilisée en prenant la matrice de co-association comme une matrice de distance avec un seuil t qui détermine le similitude minimale à partir duquel deux clusters seront fusionnés. Ce type d'approche permet

	$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$
x_1	1	3	1
x_2	1	3	1
x_3	2	1	1
x_4	2	1	1
x_5	3	2	2
x_6	3	2	2

Résultats de clustering

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	-	3	1	1	0	0
x_2	3	-	1	1	0	0
x_3	1	1	-	3	0	0
x_4	1	1	3	-	0	0
x_5	0	0	0	0	-	3
x_6	0	0	0	0	3	-

Matrice de co-association

Tableau 2.2
Exemple de calcul de la matrice de co-association.

notamment de trouver des clusters ayant des formes non conventionnelles. Il est cependant nécessaire que les clusters soient clairement séparables dans l'espace des données sous peine de ne former qu'un cluster unique lors du clustering final. De plus, la construction et l'utilisation de la matrice de co-association implique une complexité quadratique en temps et en mémoire en fonction du nombre d'observations $O(n^2)$, ce qui rend l'utilisation de ces méthodes peu attractive pour traiter de grands volumes de données.

Une méthode pour introduire une pondération dans la matrice de co-association a également été proposée par (Duarte et al. 2005). Elle permet de pondérer la décision de chaque résultat de l'ensemble en fonction d'un critère de qualité interne affecté à chaque résultat. Cet indice est ensuite utilisé pour mettre à jour la matrice de co-association pondérée :

$$w\text{-co-assoc}(i, j) = \frac{1}{K} \sum_{k=0}^K V(\mathcal{C}^{(k)}, x_i, x_j) \times Q(\mathcal{C}^{(k)}) \quad (2.5)$$

où $V(\cdot)$ renvoie 1 si les instances (x_i, x_j) ont été classées ensemble dans le résultat $\mathcal{C}^{(k)}$ et 0 sinon, et $Q(\cdot)$ renvoie l'indice de qualité du résultat $\mathcal{C}^{(k)}$.

Méthodes de combinaison basées sur la définition d'un nouvel espace de données

Une autre proposition (Topchy et al. 2003) consiste à construire un nouvel espace de données à partir de l'ensemble des clusterings. Le problème du clustering par ensemble est modélisé sous la forme d'un problème de clustering à attributs catégoriels. Chaque résultat de l'ensemble donne lieu à un attribut catégoriel représentant l'objet. Ce nouvel espace de données est ensuite utilisé pour effectuer le clustering final. Une fonction d'utilité est introduite qui permet d'évaluer la qualité du consensus en fonction des différents attributs. (He et al. 2005) ont défini formellement le lien entre le clustering par ensemble et le clustering de données catégorielles. Dans des travaux similaires, (Hu et al. 2006) utilisent un champ de Markov aléatoire ainsi qu'une estimation du maximum de vraisem-

blance pour définir une métrique entre les résultats de clustering. Ils présentent deux méthodes basées sur cette nouvelle similarité permettant de calculer un consensus.

Méthodes basées sur le ré-étiquetage et le vote

Une autre approche dans le clustering par ensemble consiste à effectuer un vote au sein des différents résultats pour produire le consensus. Ainsi, (Ayad & Kamel 2008) présentent un algorithme de vote cumulatif qui permet de trouver un consensus parmi plusieurs partitions à nombre différent de clusters. Ils décrivent plusieurs approches de vote pondéré qui permettent de calculer une densité de probabilité résument les clustering initiaux. Une étude empirique montre que leur approche semble donner de meilleurs résultats que les approches par graphe. D'autres approches par vote sont présentées par (Nguyen & Caruana 2007) : IVC (*Iterative Voting Consensus*), IPVC (*Iterative Probabilistic Voting Consensus*) et IPC (*Iterative Pairwise Consensus*). Ces algorithmes utilisent une carte de caractéristiques calculée à partir de l'ensemble de résultats de clustering et utilisent une approche de type *Expectation-Maximisation* pour calculer le consensus. Un problème important dans ces approches par vote est la nécessité de mettre en correspondance les différents clusters des clusterings initiaux. La plupart des méthodes (Ayad & Kamel 2008, Zhou & Tang 2006) sélectionnent un membre de l'ensemble servant de *base* pour le ré-étiquetage des autres résultats. Une fois cette base choisie, les clusters des autres résultats sont mis en correspondance en observant le recouvrement des clusters. Cependant, le choix de l'heuristique de sélection de la base a d'importantes conséquences sur les résultats obtenus et détermine notamment le choix du nombre de clusters. Pour résoudre ce problème, (Long et al. 2005) proposent une méthode basée sur un mécanisme de correspondance floue entre les différents clusters des résultats. L'algorithme proposé produit un consensus mais également une matrice de correspondance qui fournit les liens entre les clusters des différents résultats.

Une autre méthode proposée par (Zhou & Tang 2006) ne s'intéressent qu'aux partitions ayant le même nombre de clusters (voir l'exemple tableau 2.3). Une fois les clusters ré-étiquetés, un vote à la majorité est effectué pour construire le consensus. Des pondérations sont appliquées à chacune des décisions en fonction de son taux d'accord avec les autres méthodes. (Dimitriadou et al. 2002) présentent un algorithme basé sur la minimisation de l'erreur au carré entre les résultats de l'ensemble. L'algorithme de vote trouve une solution approchée, la recherche de la meilleure solution du problème de minimisation étant impossible, car il est nécessaire d'énumérer toutes les permutations possibles. L'algorithme effectue une recherche séquentielle, où pour chaque partition de l'ensemble, la meilleure permutation d'étiquette est effectuée.

Une autre approche proposée par (Oliveira & Pedrycz 2007) consiste à utiliser une correspondance douce entre les clusters pour refléter le fait qu'un cluster d'un résultat peut être lié avec plusieurs clusters d'un autre résultat. Une nouvelle fonction de consensus appelée ITK (*Information Theoretic KMEANS*) est proposée. Celle-ci prend en entrée le degré d'appartenance de chaque objet aux clusters des résultats et non pas une appartenance dure comme dans les autres travaux sur l'ensemble clustering. Le fait de posséder cette in-

	$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$
x_1	1	3	1
x_2	1	3	1
x_3	2	1	1
x_4	2	1	3
x_5	3	2	3
x_6	3	2	2

Résultats initiaux

	$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$
x_1	1	1	1
x_2	1	1	1
x_3	2	2	1
x_4	2	2	2
x_5	3	3	2
x_6	3	3	3

Résultats ré-étiquetés

<i>Objet</i>	x_1	x_2	x_3	x_4	x_5	x_6
Vote	(3,0,0)	(3,0,0)	(1,2,0)	(0,3,0)	(0,1,2)	(0,0,3)
Résultat	1	1	2	2	3	3

Résultat du vote

Tableau 2.3
Exemple de ré-étiquetage suivi d'un vote.

formation d'appartenance permet de définir une fonction de consensus plus précise qui prend en compte l'information floue proposée par les algorithmes de clustering. Il est cependant nécessaire d'utiliser des algorithmes de clustering proposant ce type de résultat en sortie. L'algorithme EM est par exemple bien adapté car il est possible d'obtenir un degré d'appartenance à chaque loi composant le résultat.

Afin de construire un consensus parmi les résultats d'un ensemble de clusterings n'ayant pas nécessairement le même nombre de clusters, nous avons développé une méthode de vote multivue originale (Wemmert & Gançarski 2002a). Dans cette méthode, chaque objet vote pour le cluster auquel il appartient et pour un cluster dans chacun des autres résultats (le plus similaire du cluster qu'il a lui-même choisi). Les clusters ayant obtenu le maximum de votes sont retenus pour construire le consensus. Cette approche est présentée en détails dans la section 2.1.3.

Cas des méthodes de clustering flou

À la différence des méthodes de clustering dites *dures* qui proposent un partitionnement des éléments, le clustering flou propose pour chaque élément à classer un degré d'appartenance à chaque cluster. Ainsi, les méthodes de clustering par ensemble présentées précédemment ne peuvent être appliquées directement et des méthodes spécifiques ont été étudiées et proposées.

Une architecture de clustering flou a été introduite par (Pedrycz 2002) dans laquelle plusieurs sous-ensembles d'objets d'un jeu de données initial sont traités dans le but de trouver une structure commune. Les différents sous-ensembles sont tout d'abord traités indépendamment, puis, chaque partition est modifiée en accord avec les autres partitions : chaque résultat produit à partir d'un sous-ensemble est modifié par rapport aux informations obtenues sur les autres sous-ensembles. Des expériences ainsi que plus de détails sur la méthode sont fournis dans (Pedrycz & Rai 2008). Une application de cette approche pour l'analyse de contenu web est proposée par (Loia et al. 2007). Les auteurs présentent une méthode collaborative basée sur la proximité des objets et montrent comment cette information peut être utilisée pour découvrir la structure d'informations sur le web dans des espaces sémantiques différents.

Une autre plateforme de clustering collaboratif flou est proposée par (Mitra et al. 2006) où les ensembles bruts (*rough sets*) sont utilisés pour créer un paradigme collaboratif. Un algorithme de clustering est développé en intégrant les avantages des ensembles flous et des ensembles bruts. Une analyse quantitative et des résultats expérimentaux sont aussi présentés sur des données artificielles et réelles. Ces approches collaboratives floues apportent des fondements théoriques intéressants mais sont limitées sur de nombreux aspects : chaque partition doit avoir le même nombre de clusters ; la question difficile de la correspondance entre les clusters est supposée résolue ; la distance entre chaque point et le centre des clusters dans chaque solution doit être connue. Malgré ces contraintes, il a été montré, au moins sur des exemples simples à deux ou trois clusters que la collaboration a un effet positif sur la qualité des clusters.

Des travaux récents (Grozavu & Bennani 2010) utilisent l'algorithme SOM pour effectuer cette tâche de clustering distribué de plusieurs sous-ensembles d'objets. La méthode est décomposée en deux phases. La première consiste à appliquer l'algorithme SOM sur les différentes données indépendamment. La seconde phase consiste à faire collaborer ces différentes cartes pour les enrichir. Des travaux similaires sont proposés par (Cleuziou et al. 2009) avec une approche permettant de traiter des données multi-représentées, c'est-à-dire un même ensemble d'individus décrits par plusieurs représentations.

2.1.2 Approches multiobjectives

Le clustering multiobjectif a pour but d'optimiser simultanément plusieurs critères de clustering. L'idée est de mieux saisir la notion de cluster en définissant explicitement différentes fonctions objectives. Les algorithmes sont ainsi capables de produire un ensemble de solutions qui sont des compromis des différents objectifs utilisés.

Ainsi, la méthode MOCK (*Multi-Objective Clustering with automatic K-determination*) (Handl & Knowles 2007) utilise deux objectifs : le premier est de maximiser la compacité des clusters, et le second leur connectivité. Un algorithme évolutionnaire multiobjectif est utilisé pour optimiser ces deux critères simultanément. La méthode utilise un front de Pareto (Konak et al. 2006) qui consiste à sélectionner les solutions non dominées, c'est-à-dire celles respectant le mieux les deux objectifs de manière simultanée. À la fin de l'évolution, les solutions présentes sur le front de Pareto forment l'ensemble des solutions fournies par l'algorithme. Une heuristique est ensuite utilisée pour sélectionner la meilleure solution potentielle en utilisant le nombre de clusters des solutions présentes sur le front. Dans (Handl & Knowles 2006), les auteurs présentent un moyen d'intégrer des connaissances du domaine à travers un troisième objectif basé sur un ensemble d'objets étiquetés. Cette version semi-supervisée donne de meilleurs résultats que la version sans connaissance.

(Faceli et al. 2006) ont proposé une autre méthode appelée MOCLE (*Multi-Objective Clustering Ensemble*) qui intègre les deux mêmes fonctions objectives (maximisation de la compacité et la connectivité des clusters) que MOCK. Cependant, un opérateur de croisement spécial est ajouté qui utilise des techniques de clustering par ensemble. Le but de la méthode MOCLE est de produire un ensemble de solutions représentant des compromis entre les deux objectifs alors que MOCK cherchait à trouver une unique solution. Une extension semi-supervisée de MOCLE a également été proposée par (Faceli et al. 2007). La connaissance à propos de la structure des données est également intégrée à l'aide d'un objectif additionnel. Enfin, une autre approche proposée par (Law et al. 2004) utilise également plusieurs objectifs. Cependant, la solution finale est produite en sélectionnant les meilleurs clusters dans les résultats produits. Une méthode d'échantillonnage est utilisée pour estimer la qualité des clusters en se basant sur la stabilité de leur présence sur plusieurs exécutions.

2.1.3 Approche collaborative Samarah

La méthode SAMARAH que nous avons proposée (Wemmert et al. 2000) est basée sur le principe d'un raffinement mutuel et itératif de plusieurs résultats de clustering. Le système peut être décomposé en trois grandes étapes :

1. La génération des résultats initiaux ;
2. Le raffinement des différents résultats ;
3. La combinaison des résultats raffinés.

La première étape de la méthode consiste à générer les résultats initiaux qui seront ensuite utilisés par celle-ci. Dans cette étape, des algorithmes de clustering sont appliqués sur les données. Différents algorithmes ou paramétrages du même algorithme peuvent être utilisés.

Lors de la deuxième étape, chacun des résultats va être comparé avec l'ensemble des résultats proposés par les autres méthodes. Le but est d'évaluer la similarité entre les différents résultats pour observer les différences dans les regroupements des objets. Une fois ces différences (appelées par la suite *conflits*) identifiées, l'objectif est de modifier les résultats pour tenter de réduire ces différences, c'est-à-dire résoudre les conflits. En fonction des informations obtenues sur les différences entre les résultats, des modifications (fusion de clusters, scission de clusters, reclustering de clusters) sont appliquées de manière itérative aux résultats. Cette étape peut être vue comme une remise en cause en fonction des informations fournies par les autres acteurs de la collaboration. Après plusieurs itérations de raffinement, il est attendu des résultats qu'ils soient plus similaires qu'avant cette étape collaborative.

Lors de la troisième étape, les différents résultats raffinés sont combinés si nécessaire pour proposer un résultat unique. Ce calcul d'un résultat consensuel est simplifié par la forte similarité des résultats.

La figure 2.1 présente un schéma des différentes étapes du clustering collaboratif. Dans les sections suivantes nous allons étudier en détail les différentes étapes de la méthode.

Génération des résultats initiaux

La génération des résultats initiaux consiste à appliquer plusieurs méthodes de clustering aux données. Il est nécessaire lors de cette étape de faire un choix sur le nombre de méthodes impliquées dans la collaboration ainsi que sur leur type. Nous avons vu dans le chapitre 2 qu'il existe un grand nombre de méthodes de clustering. La plupart des méthodes de clustering les plus courantes peuvent être utilisées dans la méthode SAMARAH. Nous verrons toutefois qu'il est nécessaire de modifier légèrement les méthodes pour pouvoir les utiliser de manière efficace dans le système. On obtient donc à l'issue de cette étape, un ensemble de clusterings différents $\mathcal{C} = \{\mathcal{C}^{(1)}, \dots, \mathcal{C}^{(N)}\}$, chaque clustering $\mathcal{C}^{(i)}$ ayant été généré à partir d'une méthode et/ou d'un paramétrage spécifique.

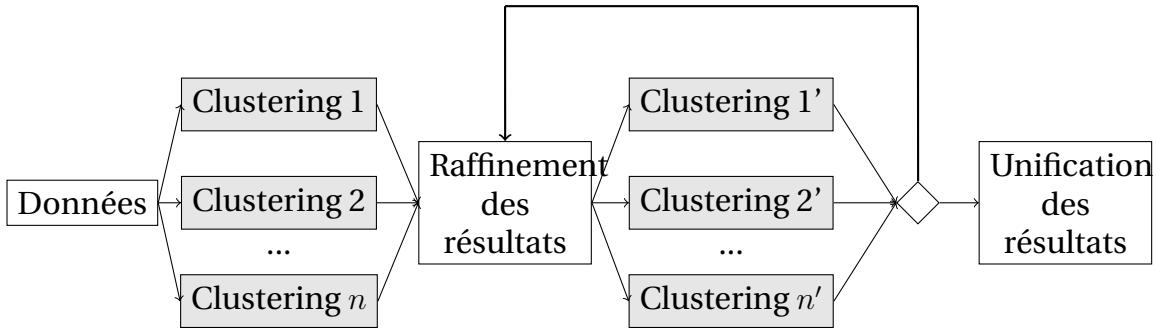


Figure 2.1
Schéma illustrant les différentes étapes du clustering collaboratif SAMARAH.

Raffinement des résultats

Le raffinement des résultats consiste à comparer les différents résultats et à observer la répartition des objets dans les différents clusters des différents résultats. L'objectif est d'identifier sur quelle partie du regroupement des données les méthodes sont en désaccord (et respectivement en accord). Pour se faire il est nécessaire de comparer les résultats entre eux. Pour pouvoir observer les similitudes et les différences de chaque résultat par rapport à tous les autres résultats, ceux-ci sont comparés deux à deux. Pour effectuer cette comparaison, la matrice de confusion (Ω) est calculée pour chaque couple de clusterings ($\mathcal{C}^{(i)}, \mathcal{C}^{(j)}$) :

$$\Omega^{(i,j)} = \begin{pmatrix} \alpha_{1,1}^{(i,j)} & \dots & \alpha_{1,K^{(j)}}^{(i,j)} \\ \vdots & \ddots & \vdots \\ \alpha_{K^{(i)},1}^{(i,j)} & \dots & \alpha_{K^{(i)},K^{(j)}}^{(i,j)} \end{pmatrix} \text{ où } \alpha_{k,l}^{(i,j)} = \frac{|C_k^{(i)} \cap C_l^{(j)}|}{|C_k^{(i)}|} \quad (2.6)$$

La matrice de confusion $\Omega^{(i,j)}$ contient la répartition des objets dans les clusters de deux résultats de clustering. Cette matrice permet d'observer si les objets d'un cluster d'un résultat ont été regroupés de manière similaire dans l'autre résultat ou au contraire ont été répartis dans plusieurs clusters et dans quelles proportions.

Cette matrice de confusion est le cœur de la comparaison des résultats en clustering collaboratif. À partir de cette information, il est possible d'évaluer la similarité des résultats, et plus encore, d'identifier avec précision leurs désaccords. Un autre intérêt de cette matrice est qu'elle est totalement indépendante de la méthode utilisée pour créer les clusters, car celle-ci ne nécessite que l'information de l'affectation des objets aux clusters pour être calculée. Nous allons voir dans la suite de cette section un ensemble de critères statistiques utilisant cette matrice de confusion et qui permettent d'obtenir des informations sur la similarité de deux résultats de clustering. Le premier critère est un critère de similarité inter-cluster qui permet la comparaison de deux clusters de deux résultats différents. Pour se faire deux matrices de confusion sont calculées. En effet, comme les

2.1. COLLABORATION ENTRE MÉTHODES DE CLUSTERING

résultats n'ont, d'une part, pas forcément le même nombre de clusters, et que d'autre part, les coefficients de la matrice sont normalisés par la taille des clusters d'un des deux résultats, on obtient $\Omega^{i,j} \neq \Omega^{j,i}$. Il est donc nécessaire de calculer cette matrice dans les deux sens ($\Omega^{i,j}$ et $\Omega^{j,i}$) pour pouvoir comparer les deux résultats. Pour calculer la similarité S entre les deux clusters, on peut ensuite utiliser ces deux matrices, en observant les deux coefficients correspondants aux deux clusters à comparer :

$$S(C_k^{(i)}, C_l^{(j)}) = \alpha_{k,l}^{(i,j)} \alpha_{l,k}^{(j,i)} \quad [2.7]$$

Cependant, ce critère ne prend en compte que l'intersection entre les deux clusters considérés sans prendre en compte la répartition des objets dans les autres clusters. Si par exemple les objets d'un cluster d'un résultat se retrouve à 50% dans le cluster d'un autre résultat, la répartition des 50% d'objets restants n'est pas prise en compte. Que ces objets soient répartis dans un autre cluster ou dans plusieurs autres clusters, la valeur du coefficient est identique. Pour prendre en compte cette distribution, la similarité entre deux clusters est définie telle que :

$$S(C_k^{(i)}, C_l^{(j)}) = \rho_k^{(i,j)} \alpha_{l,k}^{(j,i)} \quad [2.8]$$

où

$$\rho_k^{(i,j)} = \sum_{r=1}^{K^{(j)}} (\alpha_{k,r}^{(i,j)})^2 \quad [2.9]$$

Une fois cette similarité définie entre les clusters des différents résultats, il est possible de définir une fonction qui va mettre en correspondance un cluster $C_k^{(i)}$ d'un résultat $\mathcal{C}^{(i)}$ et son cluster le plus similaire dans un autre résultat $\mathcal{C}^{(j)}$:

$$\psi(C_k^{(i)}, \mathcal{C}^{(j)}) = \arg \max_{C_l^{(j)} \in \mathcal{C}^{(j)}} S(C_k^{(i)}, C_l^{(j)}) \quad [2.10]$$

Grâce à la fonction de correspondance, il est possible d'identifier si les différents résultats sont en accord sur le regroupement des données. Nous définissons le concept de *conflit* entre un cluster d'un résultat et un autre résultat, si ce cluster ne se retrouve pas de manière parfaite dans l'autre résultat.

L'étape de détection des conflits consiste à chercher dans \mathbb{C} tous les couples $(C_k^{(i)}, \mathcal{C}^{(j)})$, $i \neq j$, tel que $S(C_k^{(i)}, \psi(C_k^{(i)}, \mathcal{C}^{(j)})) < 1$, ce qui signifie que le cluster $C_k^{(i)}$ ne peut pas être trouvé exactement dans le résultat $\mathcal{C}^{(j)}$. La liste des conflits de \mathbb{C} peut se définir comme :

$$\text{conflits}(\mathbb{C}) = \{(C_k^{(i)}, \mathcal{C}^{(j)}) : i \neq j, S(C_k^{(i)}, \psi(C_k^{(i)}, \mathcal{C}^{(j)})) < 1\} \quad [2.11]$$

Chaque conflit $\mathcal{K}_k^{(i,j)}$ est lié à un cluster $C_k^{(i)}$ et un résultat $\mathcal{C}^{(j)}$. Son importance CI , est évaluée grâce à la similarité entre le cluster du premier résultat et les clusters du deuxième résultat (équation (2.8)).

$$CI(\mathcal{K}_k^{(i,j)}) = 1 - S(C_k^{(i)}, \psi(C_k^{(i)}, \mathcal{C}^{(j)})) \quad [2.12]$$

La suite du processus collaboratif consiste à essayer de résoudre ces conflits. Dans sa version initiale, la méthode SAMARAH cherche toujours à résoudre le conflit le plus important. C'est cette approche que nous allons étudier dans un premier temps.

Le conflit le plus important est sélectionné parmi tous les conflits détectés entre tous les couples de résultats. La résolution d'un conflit $\mathcal{K}_k^{(i,j)}$ consiste à appliquer un opérateur sur chacun des résultats impliqués dans le conflit : $\mathcal{C}^{(i)}$ et $\mathcal{C}^{(j)}$. Les opérateurs qui peuvent être appliqués aux résultats sont :

- la **fusion** de clusters : plusieurs clusters sont fusionnés ensemble.
- la **scission** de cluster en sous-clusters : un clustering est appliqué aux objets d'un cluster pour créer des sous-clusters.
- la **reclustering** d'un cluster : un cluster est retiré et ses objets sont distribués dans les autres clusters restants.

L'opérateur à appliquer est choisi grâce au nombre de clusters impliqués dans le conflit, c'est-à-dire tel que $S(\mathcal{C}_k^{(i)}, \mathcal{C}_l^{(j)}) > p_{cr}$, où $0 \leq p_{cr} \leq 1$ est un paramètre choisi. Par exemple si $p_{cr} = 0.2$ cela signifie que si $\mathcal{C}_k^{(i)} \cap \mathcal{C}_l^{(j)}$ représente moins de 20% des objets de $\mathcal{C}_k^{(i)}$, $\mathcal{C}_l^{(j)}$ n'est pas considéré comme un représentant de $\mathcal{C}_k^{(i)}$.

Les détails du processus de sélection de l'opérateur à appliquer sont présentés dans l'algorithme 1. Un opérateur est appliqué à chacun des deux résultats impliqués dans le conflit. Cependant, l'application de ces deux opérateurs (un sur chaque résultat) n'est pas toujours pertinente. En effet, cela n'entraîne pas toujours une augmentation de la similarité des résultats impliqués dans le conflit. De plus, une itération de cette résolution de conflit peut mener à des solutions triviales et non voulues. Par exemple, tous les résultats ayant un unique cluster, ou un cluster pour chaque objet. Ces solutions non pertinentes doivent être évitées.

Algorithme 1: Résolution d'un conflit dans SAMARAH

Entrées : \mathbb{C} les différents résultats de clustering, $\mathcal{K}_k^{i,j}$ le conflit à résoudre

Sorties : $\mathbb{C}^* = \text{conflictResolution}(\mathbb{C}, \mathcal{K}_k^{i,j})$ le nouvel ensemble après la résolution

soit $\kappa = \{\mathcal{C}_l^{(j)}, \forall 1 \leq l \leq K^{(j)} : S(\mathcal{C}_k^{(i)}, \mathcal{C}_l^{(j)}) > p_{cr}\}$

si $|\kappa| > 1$ **alors**

$$\begin{cases} \mathcal{C}^{(i')} = \mathcal{C}^{(i)} \setminus \{\mathcal{C}_k^{(i)}\} \cup \text{scission}(\mathcal{C}_k^{(i)}, |\kappa|) \\ \mathcal{C}^{(j')} = \mathcal{C}^{(j)} \setminus \kappa \cup \text{fusion}(\kappa, \mathcal{C}^{(j)}) \end{cases}$$

sinon

$$\mathcal{C}^{(i')} = \text{reclustering}(\mathcal{C}^{(i)} \setminus \{\mathcal{C}_k^{(i)}\})$$

$$\{\mathcal{C}^{(i*)}, \mathcal{C}^{(j*)}\} = \arg \max \gamma^{(I,J)} \text{ avec } I \in \{i, i'\}, J \in \{j, j'\}$$

$$\mathbb{C}^* = \mathbb{C} \setminus \{\mathcal{C}^{(i)}, \mathcal{C}^{(j)}\} \cup \{\mathcal{C}^{(i*)}, \mathcal{C}^{(j*)}\}$$

Pour cela, la méthode SAMARAH utilise un *coefficient local d'évaluation*, qui permet de contrôler la convergence de la collaboration entre deux clusterings. Ce coefficient est basé

2.1. COLLABORATION ENTRE MÉTHODES DE CLUSTERING

sur la similarité entre tous les clusters de deux résultats, et prend en compte un coefficient de qualité des clusters eux-mêmes (δ), sélectionné par l'expert. Ce coefficient de qualité évalue la qualité du clustering pour éviter d'obtenir une solution triviale :

$$\gamma(\mathcal{C}^{(i)}, \mathcal{C}^{(j)}) = \frac{1}{2} \left(p_s \cdot \left(\frac{1}{K^{(i)}} \sum_{k=1}^{K^{(i)}} \omega_k^{(i,j)} + \frac{1}{K^{(j)}} \sum_{k=1}^{K^{(j)}} \omega_k^{(j,i)} \right) + p_q \cdot (\delta^{(i)} + \delta^{(j)}) \right) \quad (2.13)$$

où

$$\omega_k^{(i,j)} = S(C_k^{(i)}, \psi(C_k^{(i)}, \mathcal{C}^{(j)})) \quad (2.14)$$

et, p_q et p_s sont des paramètres du système ($p_q + p_s = 1$). Nous verrons en détail la définition de l'indice de qualité δ qui nous permettra notamment d'intégrer des connaissances dans la méthode dans le chapitre suivant (Chapitre 4).

Soit $\mathcal{C}^{(i')}$ (respectivement $\mathcal{C}^{(j')}$) le résultat de l'application d'un opérateur sur $\mathcal{C}^{(i)}$ (respectivement $\mathcal{C}^{(j)}$). Le coefficient local d'évaluation (équation (2.13)) est calculé pour chacun des couples : $(\mathcal{C}^{(i)}, \mathcal{C}^{(j)}), (\mathcal{C}^{(i')}, \mathcal{C}^{(j')}), (\mathcal{C}^{(i')}, \mathcal{C}^{(j)}), (\mathcal{C}^{(i)}, \mathcal{C}^{(j')})$. Le couple ayant la meilleure évaluation est conservée comme solution à la résolution du conflit.

Après cette étape de résolution locale d'un conflit (locale car effectuée pour un couple de résultats alors que plus de deux clusterings peuvent être impliqués dans la collaboration), une étape d'évaluation globale est engagée. Cette étape globale d'évaluation a pour but de valider ou non les décisions prises au niveau local. Le *coefficient global d'évaluation* est calculé en fonction de tous les coefficients locaux d'évaluation pour chaque couple de résultats :

$$\Gamma(\mathbb{C}) = \frac{1}{N} \sum_{i=1}^N \Gamma(\mathcal{C}^{(i)}) \quad (2.15)$$

où

$$\Gamma(\mathcal{C}^{(i)}) = \frac{1}{N-1} \sum_{\substack{j=1 \\ j \neq i}}^N \gamma(\mathcal{C}^{(i)}, \mathcal{C}^{(j)}) \quad (2.16)$$

Trois cas peuvent survenir concernant l'évolution de ce critère pendant la phase de collaboration :

1. Cette étape de résolution de conflit permet d'obtenir une meilleure solution au niveau global. Dans ce cas, la solution courante est remplacée par cette nouvelle solution. La liste des conflits est recalculée, et une nouvelle itération est engagée ;
2. L'étape de résolution propose la même solution qu'avant l'application des opérateurs, ce qui signifie que la résolution de ce conflit n'est pas pertinente. Ce conflit est retiré de la liste, et une nouvelle itération est engagée ;
3. Si la solution proposée par la résolution du conflit donne un résultat globalement moins pertinent, ce résultat est tout de même considéré pour éviter de tomber dans

un extremum local. Cependant, si dans la suite des itérations, aucune résolution de conflit ne permet d'obtenir une meilleure solution (après avoir épuisé la moitié des conflits à résoudre), tous les résultats sont réinitialisés à la meilleure solution courante et le conflit est retiré.

Ce processus itère jusqu'à épuisement de la liste des conflits, c'est-à-dire que tous les conflits ont été résolus, ou que la résolution des conflits restants ne permet pas d'améliorer les résultats. L'algorithme 2 présente ce processus en détail.

Algorithme 2: Méthode collaborative SAMARAH

```

soit  $\breve{\mathbb{C}} = \{\mathbb{C}^i\}_{1 \leq i \leq m}$  l'ensemble initial de résultats de clustering
soit  $\breve{K} = \text{conflicts}(\breve{\mathbb{C}})$  l'ensemble des conflits sur  $\breve{\mathbb{C}}$  (équation (2.11))
soit  $\breve{\mathbb{C}}^{\text{best}} = \breve{\mathbb{C}}$  la meilleure solution temporaire
soit  $\breve{K}^{\text{best}} = \breve{K}$  les conflits de la meilleure solution temporaire
tant que  $|\breve{K}| \geq 0$  faire
     $\mathcal{K}_k^{i,j} = \arg \max_{\mathcal{K}_l^{r,s} \in \breve{K}} CI(\mathcal{K}_l^{r,s})$ 
     $\breve{\mathbb{C}} = \text{conflictResolution}(\breve{\mathbb{C}}, \mathcal{K}_k^{i,j})$  (Algorithme 1)
    si  $\Gamma(\breve{\mathbb{C}}) > \Gamma(\breve{\mathbb{C}}^{\text{best}})$  alors
         $\breve{\mathbb{C}}^{\text{best}} = \breve{\mathbb{C}}$ 
         $\breve{K}^{\text{best}} = \breve{K} = \text{conflicts}(\breve{\mathbb{C}})$ 
         $bt = 0$ 
    sinon si  $\breve{\mathbb{C}}^{t+1} = \breve{\mathbb{C}}^t$  alors
         $\breve{K} = \breve{K} \setminus \mathcal{K}_k^{i,j}$ 
    sinon
         $bt := bt + 1$ 
         $\breve{K} = \breve{K} \setminus \mathcal{K}_k^{i,j}$ 
        si  $bt > |\breve{K}|$  alors
             $\breve{\mathbb{C}} = \breve{\mathbb{C}}^{\text{best}}$ 
             $\breve{K} = \breve{K}^{\text{best}} \setminus \mathcal{K}_k^{i,j}$ 
    
```

calcul du consensus

Combinaison des résultats raffinés

Après l'étape de raffinement, tous les résultats sont relativement similaires, avec un nombre de clusters proche. Chaque résultat peut être individuellement intéressant pour l'expert, comme chaque résultat a été créé d'une part grâce à son algorithme propre, et d'autre part par les informations obtenues des autres méthodes pendant la collaboration.

Cependant, si l'expert est intéressé par un résultat unique, représentant le mieux possible l'ensemble des résultats, il est possible d'appliquer des algorithmes d'ensemble clustering vue précédemment. Nous présentons ici une méthode que nous avons développée (Wemmert & Gançarski 2002b) et qui utilise les concepts et les méthodes mis en place en clustering collaboratif.

Dans cet algorithme de vote, chaque résultat $\mathcal{C}^{(i)}$ effectue un vote pour chaque objet x . Il va tout d'abord voter pour le cluster $C_k^{(i)}$ auquel est associé x dans son résultat, ainsi que pour tous les clusters correspondants $\psi(C_k^{(i)}, \mathcal{C}^{(j)})$ dans les autres résultats de clustering impliqués dans la collaboration. Une fois tous les votes effectués pour chaque résultat et chaque objet, la valeur maximale indique le meilleur cluster pour l'objet x , par exemple $C_l^{(j)}$. Cela signifie que l'objet x doit être dans le cluster $C_l^{(j)}$ d'après l'opinion majoritaire des résultats.

Pour chaque objet x , un ensemble de vecteurs de vote est calculé :

$$\mathcal{V}(x) = \left\{ (v_1^{(i)}(x), \dots, v_{K^{(i)}}^{(i)}(x)), 1 \leq i \leq N \right\} \quad (2.17)$$

où

$$v_k^{(i)}(x) = \sum_{j=1}^N \text{vote}(x, C_k^{(i)}, \mathcal{C}^{(j)}) \quad (2.18)$$

et

$$\text{vote}(x, C_k^{(i)}, \mathcal{C}^{(m)}) = \begin{cases} 1 & \text{si } (i = m \text{ et } x \in C_k^{(i)}) \\ & \text{ou } x \in \psi(C_k^{(i)}, \mathcal{C}^{(m)}) \\ 0 & \text{sinon} \end{cases} \quad (2.19)$$

L'objet x est affecté au cluster $\check{\mathcal{V}}$, défini comme :

$$\check{\mathcal{V}}(x) = \arg \max_{C_k^{(i)}} v_k^{(i)}(x) \quad (2.20)$$

Le tableau 2.4 présente un exemple de l'application de cet algorithme de vote. Une fois les différents vecteurs de \mathcal{V} calculés, le maximum est cherché pour assigner un cluster d'un résultat à chaque objet (voir équation (2.20)). En cas d'égalité, le premier cluster trouvé pour un objet est conservé. Ces égalités rendent possible le fait que plusieurs clusters soient créés dans le résultat final mais qu'ils représentent en fait des clusters très similaires dans deux résultats différents. Il est possible d'utiliser des heuristiques pour réduire le nombre de clusters dans le résultat final, comme par exemple établir une correspondance entre les clusters très similaires et effectuer un ré-étiquetage en post-traitement.

	$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$		$\mathcal{C}^{(1)}$	$\mathcal{C}^{(2)}$	$\mathcal{C}^{(3)}$
x_1	1	3	1		$\mathcal{V}(x_1)$	$\{(3, 1, 0), (1, 0, 3), (1, 2, 0)\}$	
x_2	1	3	1		$\mathcal{V}(x_2)$	$\{(3, 1, 0), (1, 0, 3), (1, 2, 0)\}$	
x_3	2	1	1		$\mathcal{V}(x_3)$	$\{(1, 3, 0), (3, 0, 1), (1, 2, 0)\}$	
x_4	2	1	1		$\mathcal{V}(x_4)$	$\{(1, 3, 0), (3, 0, 1), (1, 2, 0)\}$	
x_5	3	2	2		$\mathcal{V}(x_5)$	$\{(0, 0, 3), (0, 3, 0), (0, 0, 3)\}$	
x_6	3	2	2		$\mathcal{V}(x_6)$	$\{(0, 0, 3), (0, 3, 0), (0, 0, 3)\}$	

Résultats initiaux

Votes pour les différents objets

<i>Objet</i>	x_1	x_2	x_3	x_4	x_5	x_6
<i>Résultat</i>	1	1	2	2	3	3

Résultat du vote

Tableau 2.4

Exemple de l'application de l'algorithme de vote à la fin du processus collaboratif SAMA-RAH.

2.2 Collaboration entre clustering et classification semi-supervisée

Le nombre d'exemples étiquetés est un paramètre essentiel lors de la phase d'apprentissage en classification supervisée. Si trop peu d'exemples sont disponibles, le modèle prédictif induit par l'apprentissage aura des performances relativement faibles. Malheureusement, dans de nombreuses applications réelles, les objets étiquetés sont difficiles à obtenir.

En effet, cela s'explique souvent par le coût important de l'intervention humaine dans le processus d'identification et de sélection des exemples. Nous pouvons citer par exemple la recherche d'objets dans une base de données à partir de quelques échantillons saisis par l'utilisateur (recherche d'images basée sur le contenu, recommandation de sites web, etc.). Dans ces cas, très peu d'exemples étiquetés sont disponibles alors qu'un nombre important de données non-étiquetées sont disponibles (toutes les autres instances de la base de données).

Pour l'exemple de la recommandation de sites web, il est difficile d'imaginer de demander à l'utilisateur d'étiqueter plus de sites qu'il n'en connaît, puisque son objectif est précisément de trouver des sites similaires, mais qu'il ne connaît pas encore. Le même problème apparaît en vente à distance, pour les systèmes de recommandation d'ar-

2.2. COLLABORATION ENTRE CLUSTERING ET CLASSIFICATION SEMI-SUPERVISÉE

ticles, qui se basent uniquement sur les achats déjà réalisés par l'acheteur. Enfin, dans les applications nécessitant une identification visuelle par l'utilisateur pour la création des exemples, leur nombre est généralement assez faible par rapport au volume des données dans lesquelles la recherche va s'effectuer.

Un autre problème important est de parvenir à réaliser une classification très précise, lorsque le rapport entre le nombre d'exemples étiquetés et le nombre d'attributs les décrivant est très faible. Si le nombre d'attributs est élevé, les classificateurs standards nécessitent beaucoup d'exemples pour obtenir de bons taux de classification. Cette observation est connue comme étant le phénomène de Hughes (Hughes 1968).

Dans le cadre de la télédétection, les capteurs hyperspectraux peuvent désormais produire des données comportant un nombre important de bandes spectrales (jusqu'à 200 valeurs réelles par pixel de l'image). Avec de telles données, plus de détails peuvent être observés sur la couverture du sol, c'est-à-dire que le nombre de classes d'intérêt augmente. Plus d'attributs et plus de classes implique un besoin plus important en terme d'exemples, qui sont généralement coûteux à acquérir. La même observation peut être réalisée avec les méthodes d'analyse basées sur les objets des images à très haute résolution spatiale (taille d'un pixel inférieur à 5m). Avec ce type d'images, une première étape de segmentation est d'abord réalisée afin de construire des régions. Celles-ci sont alors décrites par un ensemble d'attributs spectraux, spatiaux ou contextuels trop importants en regard du nombre d'exemples disponibles.

Alors que les données étiquetées sont rares et souvent insuffisantes par rapport à la dimension de l'espace de recherche, les données non-étiquetées sont disponibles en très grande quantité. Des travaux récents montrent que ces données peuvent être utilisées afin d'améliorer la qualité d'une classification supervisée (Blum & Mitchell 1998, Nigam et al. 2000, Seeger 2002). On parle alors de *classification semi-supervisée*.

La méthode que nous présentons dans cette section est sensiblement différente des approches existantes, car nous utilisons plusieurs classificateurs non supervisés afin de créer une nouvelle description des exemples basée sur les clusters proposés. Ensuite, un classificateur supervisé est appliqué dans ce nouvel espace de données. Comme l'objectif des classificateurs non-supervisés est de créer des clusters qui maximisent la similarité intra-cluster conjointement à la dissimilarité inter-cluster, aucun exemple étiqueté n'est nécessaire, mais aucune étiquette n'est attribuée aux clusters trouvés. La classification non-supervisée (ou *clustering*) peut être vue comme un moyen de résumer la distribution des objets étiquetés dans leur espace.

Cette section se compose de deux parties, la première présentant un état de l'art des méthodes semi-supervisées regroupées selon trois familles, les méthodes de pré-étiquetage, les méthodes de post-étiquetage et les méthodes de *clustering semi-supervisé*. La seconde partie s'intéresse à une méthode originale d'enrichissement d'un apprentissage semi-supervisé à partir de multiples clusterings.

2.2.1 Méthodes semi-supervisées

Comme indiqué plus haut, plusieurs travaux ont montré que les données non-étiquetées pouvaient permettre d'améliorer la qualité de la classification lorsque peu d'exemples étaient disponibles (Goldman & Zhou 2000, Blum & Mitchell 1998, Joachims 1999, Nigam et al. 2000, Bennett et al. 2002, Chawla & Karakoulas 2005, Zhou et al. 2007, Raskutti et al. 2002, Deodhar & Ghosh 2007, Cai et al. 2009, Karem et al. 2012).

Gabrys & Petrakieva (2004) ou Bouchachia (2007) proposent de classer ces méthodes en trois catégories principales : les *méthodes de pré-étiquetage*, pour lesquelles les données non-étiquetées sont étiquetées à l'aide d'un classifieur initial entraîné sur l'ensemble des exemples disponibles, les *méthodes de post-étiquetage*, qui consistent à étiqueter des clusters construits sur l'ensemble des données disponibles en fonction de leur composition en terme d'exemples étiquetés, et les *approches semi-supervisées*, qui utilisent à la fois les données étiquetées ou non durant le processus de clustering.

Soit X un ensemble de n objets $x_j \in X$. Nous nous plaçons dans un cas de classification à q classes avec m exemples étiquetés et l objets non-étiquetés. Nous faisons l'hypothèse que m est très faible et $l \gg m$.

Soit L l'ensemble des objets étiquetés de X :

$$L = \{(x_1, y_1), \dots, (x_m, y_m)\} \quad [2.21]$$

avec $y_i \in \{1, \dots, q\}$ les étiquettes des classes des exemples.

Soit U l'ensemble des objets non-étiquetés de X :

$$U = \{(x_{m+1}, 0), \dots, (x_{m+l}, 0)\} \quad [2.22]$$

avec 0 signifiant qu'aucune étiquette n'est associée à cet objet.

L'objectif de la classification semi-supervisée est de construire un classifieur basé sur tout l'ensemble d'apprentissage X . Ce classifieur peut être vu comme une fonction associant une des q classes à chaque objet de x . Il peut être défini formellement de la manière suivante :

$$y = C_X(x) : y \in \{1, \dots, q\} \quad [2.23]$$

Méthodes de pré-étiquetage

La première famille de méthodes étudiée se composent des *méthodes de pré-étiquetage*. L'idée est de réaliser une première classification C_L uniquement sur les données étiquetées (L). Ensuite, les données non-étiquetées sont étiquetées en fonction de cette classification.

Une première manière d'utiliser les objets non-étiquetées est mise en œuvre dans la méthode dite de *co-training* développée par Blum & Mitchell (1998). L'idée principale est d'utiliser deux classifications complémentaires afin d'étiqueter itérativement les données non-étiquetées. Cela implique qu'il existe deux ensembles d'attributs indépendants et complémentaires pour décrire les données.

2.2. COLLABORATION ENTRE CLUSTERING ET CLASSIFICATION SEMI-SUPERVISÉE

Afin d'étendre cette méthode et de permettre d'éviter l'indépendance et la redondance entre les ensembles d'attributs, ce qui n'est pas très réaliste dans le cas d'applications réelles, Goldman & Zhou (2000) ont présenté une stratégie de co-training qui utilise les données non-étiquetées afin d'améliorer les performances d'un classifieur supervisé. Leur méthode utilise aussi deux classificateurs supervisés différents qui sélectionnent les futurs objets à étiqueter. Les expériences montrent que la méthode augmente significativement la précision de l'algorithme ID3. Dans (Nigam et al. 2000), une version basée sur l'algorithme *Expectation Maximization* (EM) est proposée et appliquée à la classification de textes.

Plus récemment, Raskutti et al. (2002) ont présenté une méthode de co-training qui ne nécessite pas obligatoirement d'utiliser deux classificateurs complémentaires. L'idée est de produire une "vue" différente des données en calculant une classification non-supervisée sur tout le jeu de données (étiqueté et non-étiqueté). Alors, les données originales et la nouvelle représentation sont utilisées pour créer deux prédicteurs indépendants et appliquer le co-training.

Dans (Zhou et al. 2007), les auteurs présentent une approche de co-training qui suppose d'avoir deux représentations des données. La méthode utilise la corrélation entre les deux vues afin de produire de manière itérative de nouveaux exemples positifs et négatifs. Les expériences montrent que cette technique surpassé les autres approches de co-training, sur des cas pour lesquels un seul exemple étiqueté existe.

Contrairement au co-training, ASSEMBLE (Bennett et al. 2002) ne nécessite pas d'avoir des vues multiples sur les données. Il incorpore itérativement des exemples auto-étiquetés par un processus de boosting. À chaque itération, des objets non-étiquetés le sont grâce à l'ensemble d'exemples existant et ajoutés à l'ensemble d'apprentissage.

Dans la méthode de *static labeling* (**SL**) (Gabrys & Petrikieva 2004), les données non-étiquetées sont toutes étiquetées en une étape en appliquant simplement C_L à U . Un nouvel ensemble de données W est alors construit :

$$W = \{(x_j, y_j) : y_j = C_L(x_j), x_j \in U\} \quad 2.24$$

Ensuite, lors d'une seconde étape, la classification finale est calculée comme :

$$y = C_{L \cup W}(x) \quad 2.25$$

L'algorithme décrivant cette méthode est présenté en annexe A par l'algorithme 7.

Une autre méthode de pré-étiquetage, appelée *dynamic labeling* (**DL**), est proposée dans (Gabrys & Petrikieva 2004). Comme pour la méthode précédente, un classificateur C_L est construit à partir de l'ensemble d'apprentissage. Ensuite, plutôt que d'étiqueter l'ensemble des objets de U en une fois, ils le sont de manière itérative, un à la fois. Un objet x_j de U est choisi et étiqueté en fonction de C_L . Il est alors ajouté dans L et un nouveau classificateur est entraîné à partir de $L \cup \{x_j\}$. Le processus est itéré jusqu'à ce que tous les objets non-étiquetés soient étiquetés. Cet algorithme est présenté en annexe A par l'algorithme 8. L'ordre dans lequel sont choisis les objets est défini en fonction de leur degré de confiance dans la classification de chacun.

Méthodes de post-étiquetage

À l'inverse des méthodes de pré-étiquetage qui utilisent les données non-étiquetées pour améliorer un classifieur initial, les *méthodes de post-étiquetage* commencent par construire un clustering sur tous les objets de l'ensemble, mais sans tenir compte des étiquettes et exemples disponibles. Ensuite, les données non-étiquetées de chaque cluster sont étiquetées avec l'étiquette majoritaire de leur cluster.

Soit K_l , $l = 1, \dots, k$, les clusters produits par la première étape de classification non-supervisée, et c_{lj} , $j = 1, \dots, q$ le nombre d'objets étiquetés avec la classe j dans le cluster l :

$$c_{lj} = \|\{(x_i, y_i) \in L : (x_i, y_i) \in K_l, y_i = j\}\| \quad (2.26)$$

Étiquetage des clusters à la majorité La méthode de post-étiquetage présentée ici est appelée *étiquetage des clusters à la majorité* (CLM) (Gabrys & Petrakieva 2004) et est décrite en annexe A par l'algorithme 9. Elle se compose de trois étapes :

- La première consiste à étiqueter tous les clusters contenant au moins un exemple étiqueté. L'étiquette affectée à chaque objet du cluster est l'étiquette majoritairement présente parmi les objets étiquetés dans le cluster.
- La seconde étape affecte l'étiquette du cluster le plus similaire aux clusters ne contenant aucun exemple. La mesure de similarité $\Delta(K_j, K_k)$ dépend de la méthode de clustering utilisée et estime la distance entre deux K_l et K_k .
- Enfin, lors de la troisième étape, le classifieur final est construit en fonction du nouvel ensemble d'objets étiquetés.

Optimisation de la pureté Une autre famille de méthodes (Eick et al. 2004) est basée sur l'optimisation d'un critère de *pureté* d'un clustering K construit initialement à partir des données. L'évaluation de la pureté Π d'un cluster est basée sur deux critères :

- l'impureté de classe qui mesure le pourcentage d'*exemples minoritaires* dans les différents clusters de K ;
- le nombre de clusters k qui doit être maintenu à un niveau plutôt bas dans la majorité des cas.

Les *exemples minoritaires* sont des objets étiquetés appartenant à la classe qui n'est pas la classe majoritaire dans le cluster. Comme défini précédemment, la classe majoritaire d'un cluster K_l est $y_{K_l} = \arg \max_{j \in \{1 \dots q\}} (c_{lj})$. Ainsi, les *exemples minoritaires* $m(K_l)$ d'un cluster K_l peuvent être défini par :

$$m(K_l) = \{(x_i, y_i) \in K_l : y_i \neq y_{K_l}\} \quad (2.27)$$

et les *exemples minoritaires* $M(K)$ du clustering K par :

$$M(K) = \{m(K_l) : \forall l \in [1 \dots k]\} \quad (2.28)$$

La pureté se définit alors comme :

$$\Pi(K) = \text{impurity}(K) + \eta \times \text{penalty}(k) \quad (2.29)$$

avec

$$\text{impurity}(K) = \frac{\|M(K)\|}{n}$$

et

$$\text{penalty}(k) = \begin{cases} \sqrt{\frac{k-q}{n}} & k \geq q \\ 0 & k < q \end{cases}$$

Le paramètre η ($0 < \eta < 2$) détermine la pénalité associée au nombre de clusters k afin qu'il ne devienne pas trop important.

Le premier algorithme défini par Eick et al. (2004) est un algorithme glouton appelé *Single Representative Insertion/Deletion Steepest Descent Hill Climbing with Randomized Restart* (**SRIDHCR**). Plusieurs objets sont sélectionnés aléatoirement (entre q et $2q$ objets) pour être les représentants initiaux des clusters, qui sont créés en leur affectant les objets leur étant les plus proches. Ensuite, un objet est ajouté ou supprimé de l'ensemble des représentants. La qualité $\Pi(K)$ est évaluée et l'algorithme itère jusqu'à ce qu'il n'y ait plus d'amélioration significative de la qualité du résultat. En général, l'algorithme est lancé r fois et la meilleure solution est conservée. Une description précise est donnée en annexe A par l'algorithme 10.

Le second algorithme, *Supervised Clustering using Evolutionary Computing* (**SCEC**), essaie de trouver les meilleurs représentants des clusters par une approche évolutionnaire. Un premier ensemble de p clusterings est généré aléatoirement puis des opérateurs génétiques sont appliqués pour créer les générations futures. Trois opérateurs sont utilisés dans SCEC :

- mutation : un représentant est remplacé par un autre objet qui n'est pas déjà un représentant dans une des solutions ;
- croisement : cet opérateur crée une nouvelle solution à partir de deux solutions existantes ; l'intersection des représentants des deux solutions est insérée dans la nouvelle solution, et chaque représentant restant de chacune des solutions est inséré ou non avec une probabilité de 50% dans la nouvelle solution ;
- copie : une solution de la génération actuelle est copiée dans la génération suivante.

Les nouveaux représentants sont choisis aléatoirement par un tournoi d'ordre k sur l'ensemble des représentants construits à partir des opérateurs. Le processus est itéré sur un nombre fixe de N générations. L'algorithme complet est décrit en annexe A par l'algorithme 11.

Clustering semi-supervisé

Le dernier type de méthodes, appelées approches par *clustering semi-supervisé*, utilisent les données étiquetées ou non en même temps. L'idée est que le clustering des données doit être guidé par les exemples étiquetés.

Basu et al. (2002) ont défini deux variantes de l'algorithme K-means, qui permettent d'utiliser des données étiquetées pour améliorer la classification. L'idée principale est d'utiliser les objets étiquetés pour guider la phase d'initialisation des noyaux. Dans la première méthode, les noyaux initiaux sont déterminés uniquement grâce aux données

étiquetées puis l'algorithme se déroule normalement. Tous les objets (étiquetés ou non) se déplacent à chaque itération et sont affectés au cluster le plus proche. Dans la seconde variante, les objets étiquetés restent dans leur cluster initial quelque soit la manière dont évolue leurs centres.

Dans (Chawla & Karakoulas 2005), une étude empirique de plusieurs techniques d'apprentissage semi-supervisé appliquées à différents type de jeux de données est présentée. Plusieurs expériences permettent d'évaluer l'influence de la taille des ensembles d'objets étiquetés et non-étiquetés, et l'effet du bruit dans les données exemples. L'article conclut que la performance des différentes méthodes dépend fortement du domaine d'application et de la nature des données. Cependant, l'utilisation conjointe de données étiquetées et non-étiquetées augmente significativement la qualité des résultats dans l'ensemble des cas.

Cai et al. (2009) proposent l'idée de calculer simultanément le clustering et la classification supervisée, plutôt que de procéder en deux étapes. Afin de le réaliser, les auteurs définissent une fonction d'objectif qui évalue à la fois la qualité de classification et de clustering en mélangeant deux termes : le taux de mauvaise classification (pour la partie supervisée) et l'impureté des clusters (pour l'aspect non-supervisé). Une méthode à peu près similaire est présentée dans (Deodhar & Ghosh 2007) et appliquée à des données réelles en marketing.

Une approche plus récente est exposée dans (Karem et al. 2012), qui proposent de fusionner via la théorie de Dempster-Shafer des résultats de clustering et de classification supervisée obtenus sur un même jeu de données.

Refined clustering La méthode *refined clustering* (RC) (Gabrys & Petrakieva 2004) se compose de deux étapes :

- création d'un nouvel ensemble de données totalement étiqueté à partir de toutes les données disponibles ($L \cup U$) ;
- application d'un algorithme des k-plus proches voisins pour construire le classifieur final.

La première étape est évidemment la plus cruciale et peut être décomposée en deux sous-parties :

- la création d'un nouvel ensemble de données, incorporant à la fois les objets étiquetés et non-étiquetés et représentant un maximum d'information ;
- l'étiquetage de ce nouvel ensemble par la méthode CLM décrite précédemment (Section 2.2.1).

La création du nouvel ensemble est réalisée grâce à une approche divisive. L'idée est de scinder les clusters existants contenant des objets étiquetés par différentes classes. Un seuil est défini comme le ratio représentatif minimal d'une classe dans un cluster.

De plus, afin d'éviter la disparition des classes minoritaires, celles n'étant présentes que dans un seul cluster sont conservées en scindant le cluster en deux. L'algorithme complet est présenté en annexe A par l'algorithme 12.

Méthodes de seeding Finalement, nous utilisons aussi dans cette étude les méthodes dites de *seeding* proposées par Basu et al. (2002). Ces méthodes sont des variantes de l'algorithme de partitionnement K-means. L'objectif de K-means est de générer un k -partitionnement $K = \bigcup_{i=1}^k K_i$ de l'ensemble des données X . Chaque cluster K_i est représenté par son centre de gravité μ_i . Comme indiqué précédemment, nous nous intéressons à un problème de classification à q classes à partir d'un ensemble de données $X = L \cup U$ où L est l'ensemble des données étiquetées. L'idée est de guider l'algorithme K-means en utilisant les objets étiquetés L comme noyaux initiaux. Un premier q -partitionnement $\{S_i\}_{i=1,\dots,q}$ de L est calculé en regroupant les objets de L ayant la même étiquette dans un même cluster. Une hypothèse forte est qu'il existe au moins un objet pour chacun des clusters S_i , c'est-à-dire que l'on dispose d'au moins un exemple par classe.

La première méthode, *Seeded-Kmeans* (**SK**), utilise simplement cette initialisation des noyaux plutôt qu'une initialisation aléatoire. L'algorithme K-means est ensuite déroulé normalement sans aucune modification. Une présentation complète de SK est donnée en annexe A par l'algorithme 13.

Dans la seconde méthode, appelée *Constrained-Kmeans* (**CK**), le partitionnement des objets étiquetés est utilisés comme dans la méthode *Seeded-Kmeans* pour initialiser le clustering. Cependant, les objets étiquetés ne sont pas réassignés à d'autres clusters durant l'exécution de l'algorithme. Ils sont contraints dans leur cluster d'origine. Cet algorithme est décrit en annexe A par l'algorithme 14.

2.2.2 Apprentissage semi-supervisé enrichi par de multiples clusterings

La méthode que nous proposons (Wemmert & Forestier 2011), appelée *Semi-supervised learning enhanced by multiple clusterings* (**SLEMC**), peut être considérée comme faisant partie de la catégorie des méthodes de post-étiquetage. En effet, elle essaie d'améliorer la classification en produisant tout d'abord un clustering sur l'ensemble de données. Celui-ci, calculé sur tout l'ensemble (étiqueté et non-étiqueté), regroupe les objets similaires ensemble. Ainsi, si les classes du problème sont bien séparées dans leur espace d'attributs, il est relativement simple d'affecter à chaque cluster la classe des objets exemplaires qui la composent.

Malheureusement, dans les problèmes réels, les classes ne sont pas bien séparées. Il arrive donc fréquemment d'avoir plusieurs objets étiquetés par différentes classes dans un même cluster, ou des clusters sans aucun exemple. Afin de résoudre ce problème, nous proposons d'utiliser un ensemble de clusterings.

Nous considérons b clusterings de l'ensemble de données X . Soit K cet ensemble de clusterings, $K = \{K_1, \dots, K_b\}$. L'idée est d'affecter à chaque exemple étiqueté (x_i, y_i) , $\forall i : 1 < i < m$ (avec m le nombre d'exemples étiquetés), un nouvel ensemble d'attributs :

$$v(x_i) = (K_1^i, \dots, K_b^i, y_i) \quad (2.30)$$

avec K_j^i le cluster affecté par la j^{eme} méthode de clustering K_j à x_i . Ensuite, un modèle

prédictif $C_V : X \rightarrow \{1, \dots, q\}$ est dérivé de ce nouvel ensemble de données $V = \{v(x_i)\}_{i=1}^m$, en utilisant un algorithme d'apprentissage supervisé classique. Finalement, l'étiquette $C_V(x_i)$ est affectée à chacun des objets non-étiquetés x_i de U .

L'algorithme complet est présenté par l'algorithme 3.

Algorithme 3: *Apprentissage semi-supervisé enrichi par de multiples clusterings (SLEMC)*

```

soit  $L$  l'ensemble des exemples étiquetés disponibles
construire  $b$  clusterings  $K = \{K_1, \dots, K_b\}$  sur l'ensemble des données  $X$ 
pour tous les  $(x_i, y_i) \in L$  faire
     $\lfloor v(x_i) = (K_1^i, \dots, K_b^i, y_i)$ 
    construire un modèle prédictif  $C_V$  à partir de  $V = \{v(x_i)\}_{i=1}^m$  en utilisant un
    algorithme classique d'apprentissage
    utiliser  $C_V$  pour étiqueter  $U$ 
```

Comparatif des méthodes

Dans cette section, nous comparons les méthodes semi-supervisées présentées précédemment sur différents jeux de données de l'UCI. Pour plus de lisibilité, les différents résultats numériques sont donnés en annexe A. Notamment, le tableau B.1 présente en détail les caractéristiques des jeux de données utilisés.

Pour chaque expérience, les données ont été découpés en deux sous-ensembles, composés chacun de 50% des objets. Le premier ensemble est utilisé comme ensemble de données non-étiquetées pour effectuer le clustering dans les méthodes semi-supervisées. Le second ensemble permet de choisir quelques exemples étiquetés, considérés comme les connaissances disponibles. Le reste du second ensemble est utilisé pour évaluer la méthode (i.e. le calcul de la précision). Nous avons choisi d'évaluer les méthodes avec 2, 4, 8 et 16 exemples par classe.

Par exemple, pour le premier jeu de données (*iris*) qui contient trois classes, nous avons testé la méthode avec 6, 12, 24 et 48 objets étiquetés. Comme le nombre d'exemples est très faible, la qualité du résultat dépend fortement de leur sélection. C'est pourquoi, nous avons répété les expériences 30 fois et les résultats ont été moyennés. À chaque exécution, le jeu de données est coupé aléatoirement en deux.

Pour l'application la méthode SLEMC, nous avons tout d'abord choisi le nombre de clusterings à exécuter (c'est-à-dire combien d'attributs auraient les objets dans le nouvel espace des données), puis les méthodes à appliquer. Nous avons défini quatre configurations différentes :

1. *Simple* : un algorithme EM (Expectation-Maximization (Dempster et al. 1977))
2. *Low* : un algorithme EM et un algorithme K-Means (MacQueen 1967)
3. *Medium* : deux algorithmes EM et deux algorithmes K-Means

2.2. COLLABORATION ENTRE CLUSTERING ET CLASSIFICATION SEMI-SUPERVISÉE

4. *High* : c algorithmes EM et c algorithmes K-Means (avec c le nombre de classes de l'ensemble de données).

De plus, nous avons testé une variante de la méthode SLEMC, qui conserve les descriptions initiales des objets et ajoutent les nouveaux attributs. Ces configurations sont appelées par la suite *Simple+*, *Low+*, *Medium+* et *High+*.

Chaque méthode a été lancée avec un nombre de clusters égal au nombre de classes réellement présentes dans le jeu de données, à l'exception de la configuration *High* pour laquelle les méthodes de clustering avaient chacune k clusters, $k \in \{2, \dots, c\}$, choisi aléatoirement.

Afin de faire un étude complète, nous avons aussi exécuté des méthodes de classification supervisées classiques sur les jeux de données testés. Pour chaque configuration, les exemples disponibles ont été utilisés et les données non-étiquetées ignorées. Nous avons choisi plusieurs algorithmes de différents types : arbre de décision (C4.5), bayésien naïf (NB) et un 1-plus proche voisin (1-NN).

Les résultats pour chaque jeu de données et pour chaque expérience sont présentés dans les tableaux B.2, B.3, B.4 et B.5, où les valeurs données sont les moyennes et écarts-types des précisions sur les 30 exécutions pour chaque configuration et pour chaque jeu de données. Pour mémoire, le nombre d'exemples utilisé est indiqué au début de chaque ligne.

Le tableau B.6 résume pour plus de lisibilité les quatre tableaux précédents en présentant pour chaque configuration et chaque jeu de données, les trois meilleures méthodes. Dans ce tableau, nous pouvons observer que la méthode proposée, SLEMC, donne de meilleurs résultats que les méthodes supervisées et les autres approches semi-supervisées lorsque le nombre d'exemples est très faible (2 ou 4 exemples par classe).

Sur les données *iris* et *wine* nos méthodes sont en première position pour 3 des 4 configurations. Sur les données *wine*, nos méthodes sont présentes 10 fois sur 12 parmi les trois meilleurs résultats en considérant toutes les configurations. Pour certaines configurations sur certains jeux de données, la différence de précision entre nos méthodes et les autres méthodes semi-supervisées semble importante. Par exemple, pour la configuration (2) sur le jeu de données *wine*, notre méthode avec la configuration *Medium* obtient une précision de 91.888% alors que DL n'obtient que 80.924% en précision. Cela apparaît aussi sur le jeu de données *remote* pour la configuration (2) car notre méthode obtient une précision de 93.065% alors que la méthode DL obtient uniquement 86.015%.

Concernant les différentes configurations que nous avons proposées pour SLEMC, la configuration *Medium* (deux EM et deux KMeans) semble donner les meilleurs résultats puisqu'elle apparaît 10 fois sur l'ensemble du tableau présentant les trois meilleurs résultats. Cela renforce notre intuition qu'ajouter plus de clusterings améliore le résultat final, car les objets sont décrits avec plus de détails (i.e. avec plus d'attributs). Cependant, la méthode ne semble pas tirer profit de la variation du nombre de clusters, car la configuration *High* ne parvient pas à être plus performante que la configuration *Medium* qui elle utilise un nombre constant de clusters. Cela peut s'expliquer par le fait que les jeux de

données utilisés montrent généralement une relative correspondance entre les classes et les clusters.

Nous observons aussi que dans la plupart des cas, les configurations qui ne conservent pas la description des objets (*Low*, *Simple*, *Medium* et *High*) donnent de bons résultats lorsque le nombre d'objets étiquetés est très faible (2 et 4), alors que les autres configurations conservant les anciens attributs (*Low+*, *Simple+*, *Medium+* et *High+*) donnent de meilleurs résultats quand le nombre d'exemples est plus grand (8 et 16). Cela peut s'expliquer en observant que quand le nombre d'exemples est bas, leur description est trop faible (manque d'information) pour construire un classifieur performant. Cependant, comme les attributs ajoutés par les clusterings sont calculés à partir de 50% des données, les descriptions enrichies comportent plus d'information. Quand le nombre d'exemples augmente, la description des objets est plus significative et plus précise que l'information brute apportée par les clusterings.

L'intuition que si l'espace des données n'est pas corrélé avec l'information de classe, l'utilisation d'un ou plusieurs clusterings est inutile, est confirmée par les résultats obtenus sur les jeux de données *anneal* et *diabetes*, pour lesquels les résultats des approches semi-supervisées sont inférieurs à ceux du 1-plus proche voisin, bien que le nombre d'exemples soit très faible.

Parmi les autres méthodes semi-supervisées, DL a donné des résultats particulièrement bons, puisqu'il est présent 16 fois dans le tableau B.6. En second vient la méthode SL qui est présente 4 fois. Ces résultats mettent en avant que les techniques de pré-étiquetage semblent être plus performantes que les autres, notamment sur ce type de données.

Nous avons aussi réalisé une expérience pour étudier l'importance de la quantité de données non-étiquetées dans nos méthodes. À nouveau, nous avons utilisé 50% des données comme données non-étiquetées, mais aussi réalisé la même expérience avec 25% et 10% des données. Les tests ont été effectués sur le jeu de données *iris* et avec 2, 4 et 8 exemples par classe. Les résultats sont présentés sur la figure 2.2. Chaque courbe y représente l'évolution de la précision en fonction de la quantité de données non-étiquetées disponibles. Comme attendu, la méthode tire profit de l'ensemble des données non-étiquetées, puisque l'on remarque aisément que la qualité des résultats augmentent en fonction du nombre d'objets étiquetés disponibles.

2.3 Connaissances et clustering

Le processus de clustering est par définition une approche non supervisée, c'est-à-dire qu'il se base uniquement sur les données et n'utilise pas de connaissance. Cependant, sans aucune supervision, les algorithmes peuvent aboutir à des solutions non pertinentes. Les travaux actuels se concentrent donc sur la définition d'approches permettant de guider le processus de clustering par des connaissances du domaine. Plusieurs études (Anand et al. 1995, Kopanas et al. 2002) ont montré le rôle important joué par ces connaissances du domaine ainsi que par l'expert dans le processus de fouille de données. Elles

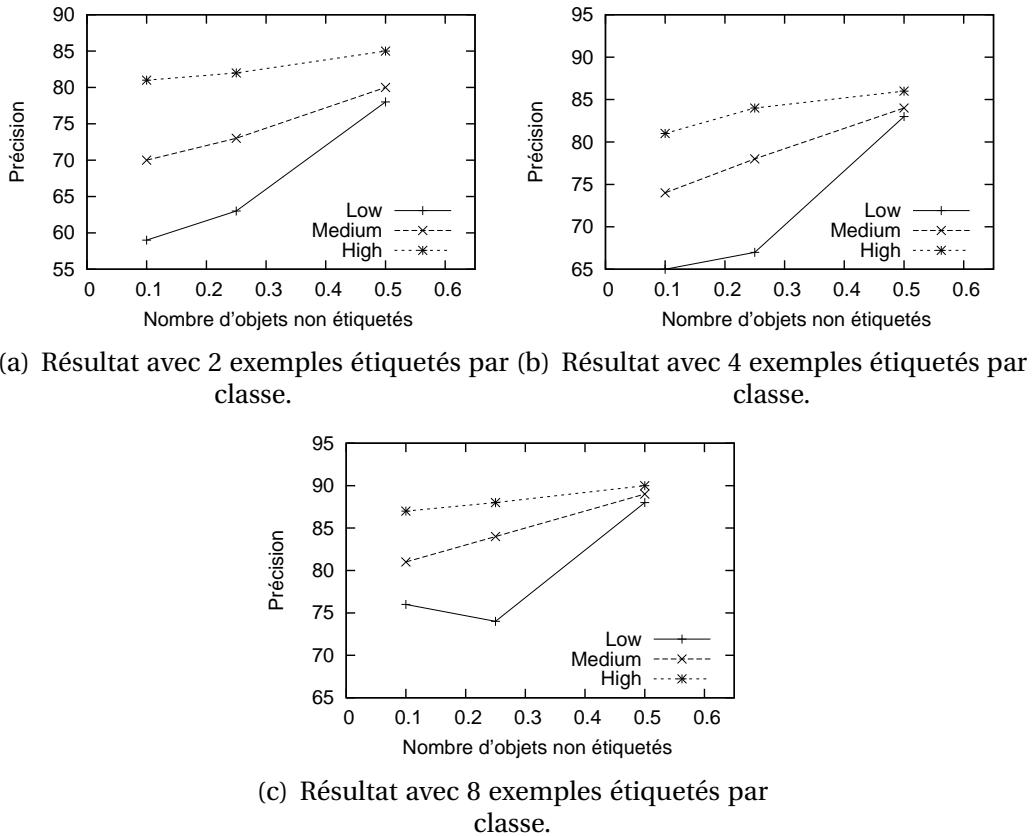


Figure 2.2

Résultats de la méthode de classification semi-supervisée SLEM-C avec un nombre variable d'objets étiquetés et non-étiquetés.

expliquent notamment que l'extraction de connaissances à partir de données ne peut pas être totalement automatique et qu'il est nécessaire d'étudier les mécanismes permettant les interactions entre d'une part les traitements automatiques (sans connaissance a priori) et d'autre part la supervision par l'expert pour guider ces traitements. Afin d'automatiser malgré tout le processus, il est nécessaire alors de parvenir à représenter les connaissances expertes et de les intégrer dans le système. Cependant, en fonction des domaines, la représentation et le type de connaissances à utiliser peuvent être très hétérogènes et plus ou moins complexes.

Deux représentations de connaissances principalement ont été proposées jusqu'à présent dans le cadre du clustering. La première concerne l'utilisation de contraintes entre objets et la seconde l'utilisation d'objets étiquetés. Ces deux représentations ont été étudiées dans différents travaux et différentes mesures de pureté proposées, permettant de tirer parti de la connaissance des étiquettes d'un ensemble d'objets. Il en ressort que les critères évaluant la pureté sans prendre en compte le nombre de clusters peuvent rapidement surévaluer la qualité des résultats. Pour résoudre ce problème, il est possible de prendre en compte une mesure qui va pénaliser les résultats avec un nombre de clusters trop important. D'autres types de critères qui comparent les regroupements de couple d'objets prennent en compte implicitement le nombre de clusters.

Dans cette section, nous présentons comment peuvent être intégrés ces critères dans la méthode SAMARAH afin de la guider. Plusieurs techniques d'intégration sont proposées et évaluées. Les résultats obtenus confirment l'intérêt d'utiliser des connaissances en clustering et plus particulièrement en clustering collaboratif.

2.3.1 Utilisation de connaissances en clustering

De nombreuses approches ont été proposées pour intégrer des connaissances dans les algorithmes de clustering. En clustering constraint (*constrained clustering*), la connaissance est exprimée sous forme de liens *must-link* et *cannot-link* entre les objets. Une contrainte *must-link* signifie que deux objets devraient être groupés dans le même cluster, et une contrainte *cannot-link* signifie le contraire. Ce type de connaissances est parfois plus simple à obtenir qu'un étiquetage classique des objets (i.e. affectation d'une classe à un objet). Wagstaff et al. (2001) ont présenté une version contrainte de l'algorithme Kmeans qui tire parti de ce type de contraintes pour biaiser l'affectation des objets aux clusters. À chaque itération, l'algorithme essaie de satisfaire les contraintes données par l'utilisateur.

Ces contraintes peuvent aussi être utilisées pour apprendre une fonction de distance biaisée par la connaissance entre les couples d'objets (Bilenko et al. 2004). La distance entre deux objets est artificiellement réduite pour une contrainte *must-link* et augmentée pour une contrainte *cannot-link*. Les travaux récents en clustering constraint s'intéressent à évaluer l'utilité potentielle d'un ensemble de contraintes (Davidson et al. 2006, Wagstaff 2007). En effet, quand le nombre de contraintes disponibles est important, il est possible que certaines soient contradictoires, introduisent du bruit et de fait, complexi-

fient le regroupement. Klein et al. (2002) ont également étudié l'intérêt de permettre aux contraintes entre les objets d'avoir des implications au niveau de l'espace des données. Cette approche a amélioré les résultats par rapport aux autres versions de Kmeans déjà proposées en requérant généralement moins de contraintes. Basu, Banerjee & Mooney (2004) ont présenté une plateforme de clustering constraint permettant la sélection active de contraintes. Dans cette approche, des objets difficiles à classer sont présentés à l'utilisateur et celui-ci exprime un *must-link* ou un *cannot-link* de manière interactive. Des résultats empiriques ont montré que choisir les contraintes de manière active améliore de manière significative les résultats et demande un effort limité de la part de l'utilisateur.

Une autre manière d'intégrer des connaissances est d'utiliser un ensemble d'objets étiquetés. Basu et al. (2002) ont proposé un algorithme qui utilise ces objets pour initialiser les clusters de l'algorithme Kmeans. Deux algorithmes, Seeded Kmeans et Constrained Kmeans sont proposés. Dans le premier, les objets étiquetés sont utilisés pour initialiser les clusters, puis les clusters sont mis à jour de manière identique à l'algorithme classique Kmeans. Dans le second, les objets étiquetés utilisés pendant l'initialisation restent ensuite dans le cluster qui leur est affecté et seul les objets non étiquetés sont mis à jour pendant la phase d'affectation des objets aux clusters. Le choix entre ces deux approches est fait en fonction de la connaissance du niveau de bruit dans les exemples, la deuxième approche y étant plus sensible.

Gao et al. (2006) ont également proposé une approche permettant d'incorporer des connaissances partielles dans un algorithme de clustering quand les objets étiquetés disponibles ne disposent pas exactement des mêmes attributs que les objets non étiquetés. Les auteurs introduisent deux algorithmes d'apprentissage pour résoudre ce problème qui sont basés sur du clustering flou et dur. Une étude empirique montre que les algorithmes proposés améliorent les résultats malgré le nombre limité d'exemples disponibles. Basu, Bilenko & Mooney (2004) ont également proposé un modèle probabiliste basé sur les champs de Markov aléatoire, qui tire parti d'objets étiquetés.

Une autre approche, appelée clustering supervisé (Eick et al. 2004), utilise l'information de la classe des objets comme une donnée supplémentaire pour construire des clusters avec une grande pureté. Le but du clustering supervisé est d'identifier des clusters de classe uniforme ayant une forte densité. Le clustering supervisé est également utilisé pour créer des résumés de jeux de données et pour améliorer des algorithmes de classification supervisée existants.

2.3.2 Évaluation d'un clustering

L'évaluation de la pureté ou de la qualité d'un clustering consiste à évaluer si le résultat du clustering est cohérent par rapport à la connaissance disponible sur les données. Nous considérons ici que la connaissance est un ensemble d'objets étiquetés. Définissons tout d'abord un ensemble de notations qui serviront à formaliser les différents critères présentés :

- Soit N le nombre d'objets étiquetés
- Soit $\mathbb{C} = \{c_1, c_2, \dots, c_K\}$ les clusters trouvés par l'algorithme de clustering

- Soit $\mathbb{W} = \{w_1, w_2, \dots, w_C\}$ les classes des objets étiquetés
- Soit c_k les objets appartenant au cluster k et w_k l'ensemble des objets de la classe k
- Soit $|c_k|$ le nombre d'objets du cluster k
- Soit $n_j^i = |w_i \cap c_j|$ les objets à la fois dans le cluster i et de la classe j

Calcul de pureté

La façon la plus simple de calculer la pureté est de chercher la classe majoritaire dans chacun des clusters et de sommer le nombre d'objets de cette classe pour chacun des clusters (Manning et al. 2008). La pureté d'un clustering se définit comme :

$$\Pi_{\text{simple}}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_i^K \arg \max_j (n_j^i) \quad (2.31)$$

Ce calcul de la pureté revient à estimer le pourcentage d'objets appartenant à la classe majoritaire de leur cluster pour l'ensemble du clustering. Sa valeur est bornée dans $[0; 1]$, 1 indiquant que les clusters sont tous purs, i.e. ils ne contiennent que des objets d'une unique classe.

Une autre façon de calculer la pureté des clusters est proposée dans le domaine du traitement de données audio où l'objectif est d'étudier si des sons sont produits par la même source. (Solomonoff et al. 1998) le formulent comme la probabilité, étant donné un cluster, que deux objets tirés au hasard sans remise soient de la même classe. La probabilité que le premier objet tiré du cluster i soit de la classe j est de $n_j^i / |c_i|$. La probabilité que le second soit également de la classe j est de $(n_j^i / |c_i|)^2$. Si ces deux objets proviennent de la même classe, alors ils viennent soit tous les deux de la classe 1, soit tous les deux de la classe 2, etc. Ces événements étant exclusifs, les probabilités peuvent être sommées pour évaluer la pureté d'un cluster.

$$\pi_{\text{prob}}(c_i) = \sum_j^C \left(\frac{n_j^i}{|c_i|} \right)^2 \quad (2.32)$$

Ce qui donne pour un clustering :

$$\Pi_{\text{prob}}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_i^K |c_i| \pi_{\text{prob}}(c_i) \quad (2.33)$$

Cette mesure a l'avantage, par rapport à la pureté simple (Eq. 2.31), de prendre en compte la distribution des classes minoritaires d'un cluster (i.e. les classes autres que la classe majoritaire), et favorise donc les clusters présentant un nombre limité de classes différentes. Sa valeur est également bornée dans $[0; 1]$, 1 indiquant que les clusters sont tous purs.

Ces deux mesures de pureté ont cependant un inconvénient majeur, qui est de surévaluer la qualité d'un clustering avec un nombre important de clusters. En effet, la pureté est maximisée dans le cas extrême où l'on observe un nombre de clusters égal au

nombre d'objets. De fait, si cette mesure de pureté est utilisée dans un algorithme où le nombre de clusters peut évoluer, l'algorithme tendra à créer un nombre plus important de clusters pour s'assurer de leur pureté. Différentes propositions ont été faites pour résoudre ce problème. Par exemple, (Ajmera et al. 2002) ont proposé de calculer la pureté des clusters en terme de classes ainsi que la pureté des classes en terme de clusters, c'est à dire pour chaque classe sa répartition dans les différents clusters. Ces deux valeurs sont ensuite combinées pour donner une évaluation globale du clustering. Considérer également la pureté des classes permet de pénaliser un nombre trop important de clusters. La pureté des classes se calcule de manière similaire à la pureté des clusters mais en observant la distribution des clusters des objets dans chaque classe :

$$\pi_{\text{prob}}^{\sim}(w_i) = \sum_j^C \left(\frac{n_j^i}{|w_i|} \right)^2 \quad (2.34)$$

ce qui donne pour un clustering :

$$\Pi_{\text{prob}}^{\sim}(\mathbb{C}, \mathbb{W}) = \frac{1}{N} \sum_i^K |c_k| \pi_{\text{prob}}^{\sim}(w_i) \quad (2.35)$$

La pureté des clusters et la pureté des classes sont ensuite combinées :

$$\Pi_{\text{overall}}(\mathbb{C}, \mathbb{W}) = \sqrt{\Pi_{\text{prob}}(\mathbb{C}, \mathbb{W}) \times \Pi_{\text{prob}}^{\sim}(\mathbb{C}, \mathbb{W})} \quad (2.36)$$

Une autre approche consiste à considérer également une mesure de la qualité du clustering à partir des données. Demiriz et al. (1999) utilisent un algorithme pour optimiser la pureté des clusters appelé Gini et similaire à Eq. 2.32. Pour éviter que l'algorithme ne génère une solution avec un nombre trop important de clusters, la fonction objective est une moyenne arithmétique de la pureté des clusters et de la qualité du clustering. La qualité du clustering est évaluée grâce à l'indice de Davies-Bouldin (Davies & Bouldin 1979) qui favorise les clusters compacts bien séparés dans l'espace des données. La combinaison de ces deux critères permet d'éviter des solutions trop extrêmes.

Enfin, Eick et al. (2004) ont également proposé d'introduire une notion de pénalité par rapport au nombre de clusters de la solution proposée afin de résoudre ce problème. Cette méthode permet de pénaliser une solution ayant un nombre de clusters trop important par rapport au nombre de classes.

$$\text{penalty}(K) = \begin{cases} \sqrt{\frac{K-C}{N}} & \text{si } K \geq C \\ 0 & \text{sinon} \end{cases} \quad (2.37)$$

avec K le nombre de clusters, C le nombre de classes et N le nombre d'objets. Cette pénalité est retranchée de l'indice de pureté choisi, pondérée par un paramètre β , comme suit :

$$\Pi_{\text{penalty}}(\mathbb{C}, \mathbb{W}) = \Pi_{\text{simple}}(\mathbb{C}, \mathbb{W}) - \beta \text{penalty}(K) \quad (2.38)$$

Une autre solution est d'utiliser le cadre de la théorie de l'information et d'évaluer l'information mutuelle normalisée entre les connaissances et le clustering :

$$\Pi_{\text{nmi}}(\mathbb{C}, \mathbb{W}) = \frac{I(\mathbb{C}, \mathbb{W})}{[H(\mathbb{C}) + H(\mathbb{W})]/2} \quad (2.39)$$

I est l'information mutuelle :

$$I(\mathbb{C}, \mathbb{W}) = \sum_i \sum_j n_j^i \log \frac{n_j^i}{|c_i|/N \times |w_j|/N} \quad (2.40)$$

$$= \sum_i \sum_j \frac{n_j^i}{N} \log \frac{n_j^i}{|c_i| \times |w_j|} \quad (2.41)$$

H est l'entropie :

$$H(\mathbb{W}) = - \sum_k \frac{|w_k|}{N} \log \frac{|w_k|}{N} \quad (2.42)$$

L'information mutuelle I (Eq. 2.40) mesure la quantité d'information que l'on obtient à propos des classes étant donné un clustering. Le dénominateur dans Eq. 2.39 permet de normaliser le critère qui prend alors ses valeurs dans $[0; 1]$, 1 indiquant des clusters purs. Le résultat maximal de ce critère étant obtenu dans le cas où le nombre de clusters est égal au nombre de classes, le critère ne possède pas les inconvénients des indices de pureté présentés précédemment.

Comparaison de partitions

Un autre indice couramment utilisé est l'indice de *Rand* (Rand 1971) qui permet de comparer des partitions. Cet indice consiste à comparer des couples d'objets et vérifier si ils sont classés de manière similaire dans deux partitions. Dans notre cas, il consiste à vérifier si les couples d'objets de la même classe d'après les connaissances disponibles, ont été placé dans un même cluster. On dit qu'un couple d'objets est un vrai positif (VP) si les deux objets sont de la même classe et sont placés dans le même cluster, et un vrai négatif (VN) quand les deux objets sont de classes différentes et sont placés dans deux clusters différents. Un faux positif (FP) correspond à deux objets de classes différentes placés dans le même cluster. Un faux négatif (FN) correspond à deux objets de la même classe dans deux clusters différents. Il peut être défini de la manière suivante :

$$\Pi_{\text{rand}}(\mathbb{C}, \mathbb{W}) = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.43)$$

$(TP + FP + FN + TN)$ représentant tous les couples possibles d'objets et $(TP + TN)$ les couples d'objets correctement classés. L'indice de *Rand* donne cependant un poids égal aux faux positifs et aux faux négatifs.

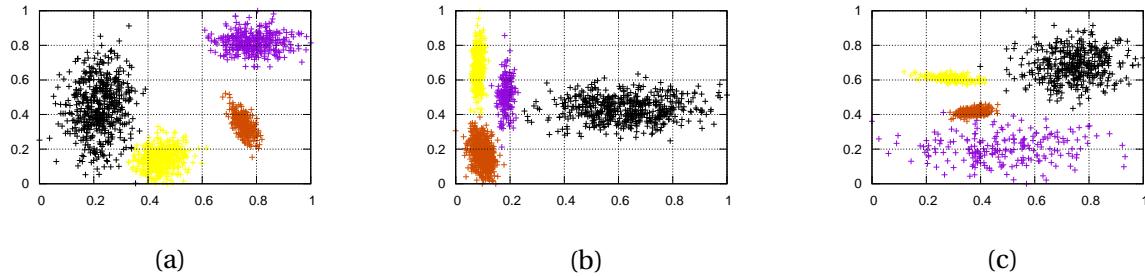


Figure 2.3
Représentations graphiques des trois jeux de données artificiels utilisés.

La *F-Mesure* (van Rijsbergen 1979) quant à elle, permet de pondérer ces deux valeurs en tenant compte de la précision (P) et du rappel (R) :

$$P = \frac{VP}{VP + FP} \quad R = \frac{VP}{VP + FN}$$

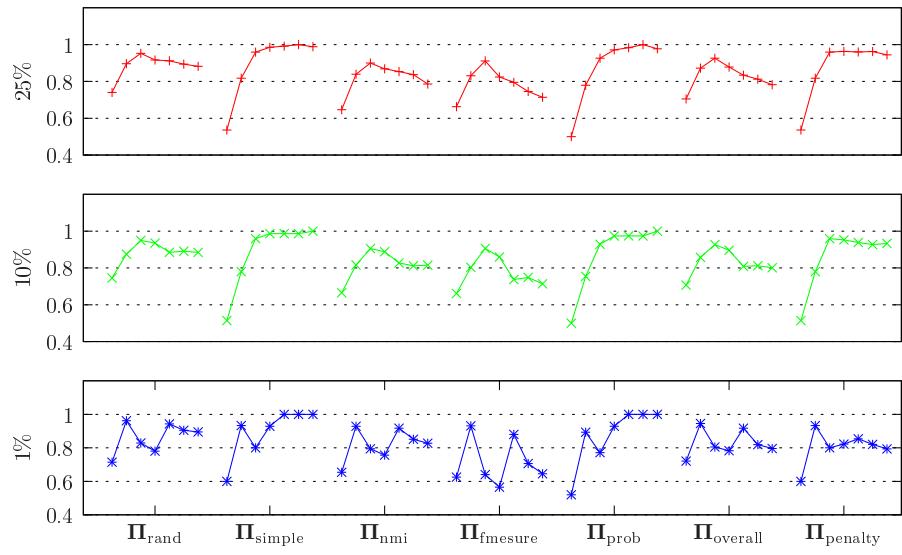
$$\Pi_{fmesure}(\mathbb{C}, \mathbb{W}) = \frac{(\beta^2 + 1)P \times R}{\beta^2 P + R} \quad (2.44)$$

Le paramètre β peut être utilisé pour pénaliser plus fortement les faux négatifs que les faux positifs en sélectionnant une valeur $\beta > 1$. Si $\beta = 1$, la précision et le rappel ont alors la même importance.

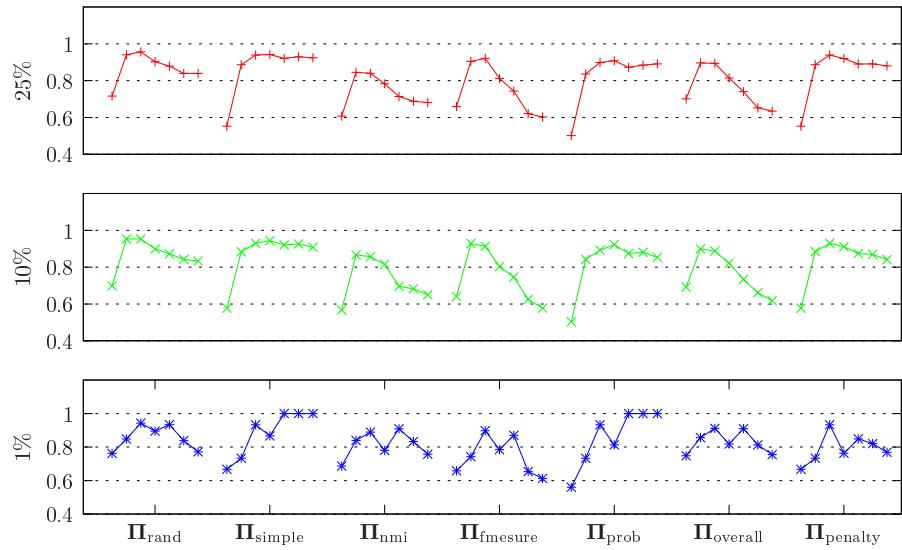
L'avantage de ces deux indices (Π_{rand} et $\Pi_{fmesure}$) est qu'ils intègrent implicitement le nombre de clusters, en défavorisant naturellement les solutions avec un nombre de clusters trop important. En effet, plus le nombre de clusters va augmenter, plus le regroupement des couples d'objets tendra à diverger de la connaissance disponible.

2.3.3 Évaluation des différents critères de qualité

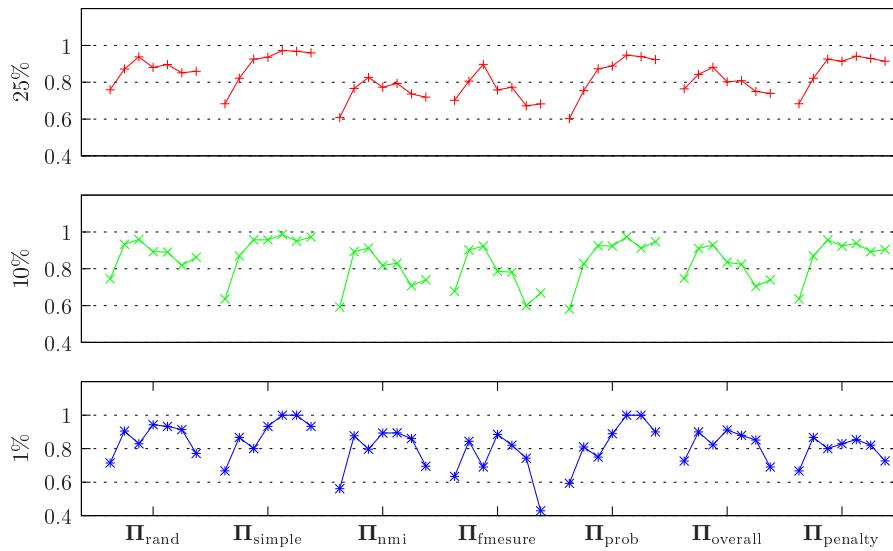
Dans cette section nous allons évaluer les critères présentés dans la section précédente sur différents jeux de données de test. La figure 2.3 présente trois jeux de données artificiels représentant chacun quatre clusters dans un espace à deux dimensions. L'algorithme Kmeans a été utilisé sur ces jeux de données avec des nombres de clusters variant de 2 à 8. Pour chaque clustering, les mesures présentées dans les sections précédentes ont été calculées. Trois configurations différentes ont été évaluées, la première avec 1% des données étiquetées, la seconde avec 10% des données étiquetées et enfin la dernière avec 25% des données étiquetées. Chaque expérience a été effectuée 100 fois avec des initialisations aléatoires de l'algorithme, puis les résultats ont été moyennés. Les figures 2.4, 2.5 et 2.6 représentent les résultats respectivement pour les jeux de données des figures 2.3(a), 2.3(b) et 2.3(c).


Figure 2.4

Évolutions des critères de pureté en fonction du nombre de clusters pour le jeu donné Figure 2.3(a) pour les trois configurations proposées (1% des données étiquetées en bleu, 10% en vert et 25% en rouge).


Figure 2.5

Évolutions des critères de pureté en fonction du nombre de clusters pour le jeu donné Figure 2.3(b) pour les trois configurations proposées (1% des données étiquetées en bleu, 10% en vert et 25% en rouge).


Figure 2.6

Évolutions des critères de pureté en fonction du nombre de clusters pour le jeu donné Figure 2.3(c) pour les trois configurations proposées (1% des données étiquetées en bleu, 10% en vert et 25% en rouge).

Quand le nombre d'objets étiquetés disponibles est faible (1%), la majorité des critères ont un comportement très aléatoire. En effet, il n'est pas du tout garanti d'avoir des objets pour chacune des classes du jeu de données. C'est pourquoi ces critères sont difficilement utilisables quand vraiment très peu de connaissances sont disponibles. Quand le nombre d'objets étiquetés augmente (10%), il est plus probable d'avoir des objets étiquetés pour chaque classe. Par conséquent, les courbes deviennent plus caractéristiques. Le problème présenté précédemment sur le fait que les mesures de pureté surévaluent la qualité du clustering quand le nombre d'objets augmente peut être observé. En effet, la pureté simple (Π_{simple}) ainsi que la pureté par cluster (Π_{prob}) ne font qu'augmenter avec le nombre de clusters. Les autres indices (Π_{rand} , Π_{nmi} , Π_{fmesure} , Π_{overall} , Π_{penalty}) ont tendance à diminuer avec l'augmentation du nombre de clusters. Les plus caractéristiques étant Π_{fmesure} , Π_{overall} et le Π_{nmi} . Les critères Π_{rand} et Π_{penalty} diminuent de façon moins caractéristique. Il est intéressant de noter qu'il n'y a pas de différence importante entre les résultats obtenus avec 10% d'objets étiquetés et 25% d'objets étiquetés.

2.3.4 Utilisation de la pureté pour guider un clustering collaboratif

Comme présenté dans la section 2.1.3, nous avons proposé une nouvelle méthode de clustering collaboratif, basée sur un processus d'amélioration mutuelle et itérative des solutions proposées par les différentes méthodes engagées dans la collaboration. La première étape du clustering collaboratif consiste à effectuer plusieurs clusterings

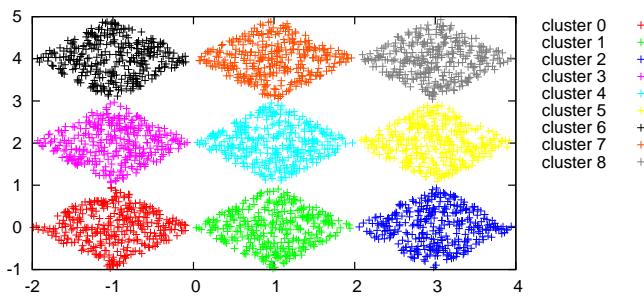


Figure 2.7
Représentation graphique du jeu de données 9-Diamonds

différents des données, ces différents résultats étant alors modifiés pendant une étape de raffinement. Lors de cette étape, chaque résultat est remis en cause à partir des informations proposées par les autres résultats. Pour évaluer l'amélioration des solutions au cours du raffinement, un critère appelé *coefficient local d'évaluation* a été défini (équation 2.13). Ce critère évalue la similarité ainsi que la qualité d'un couple de résultats de clustering. La similarité est évaluée en observant la répartition des objets au sein des clusters dans les deux résultats. La qualité est évaluée en utilisant un critère d'évaluation qui est généralement choisi en fonction de la méthode utilisée (par exemple la compacité). Une des approches pour intégrer de la connaissance dans le clustering collaboratif est de modifier ce critère de qualité pour y intégrer une évaluation en fonction des connaissances du domaine disponibles.

Évaluation de l'apport de la pureté

Pour évaluer l'intérêt des différents critères de pureté des clusters en clustering collaboratif, nous avons fait collaborer deux méthodes. L'algorithme Kmeans a été utilisé sur le jeu de données 9-Diamonds (voir figure 2.7) présentant neuf clusters en forme de losanges (Salvador & Chan 2004). Ces données proviennent de l'équipe Apprentissage de l'Université de Houston¹.

Des clusterings ont été calculés avec un nombre de clusters allant de 2 à 18, et une initialisation aléatoire des centroïdes. Pour chacun de ces clusterings, les valeurs de pureté ont été calculées avec 10 % des données étiquetées. Pour chaque couple de résultats, la moyenne des deux évaluations de la pureté a été calculée. Cette valeur moyenne est utilisée en clustering collaboratif pour évaluer la qualité d'un couple de résultats. Soit \mathbb{C}_x un résultat avec x clusters, \mathbb{C}_y un résultat avec y clusters et Π une fonction de pureté, la moyenne des deux évaluations est définie telle que :

$$f(x, y) = \frac{1}{2}(\Pi(\mathbb{C}_x, \mathbb{W}) + \Pi(\mathbb{C}_y, \mathbb{W})) \quad 2.45$$

1. <http://www.tlc2.uh.edu/dmmlg/>

	Π_{simple}	Π_{prob}	Π_{overall}	Π_{penalty}	Π_{NMI}	Π_{rand}	Π_{fmesure}
c_{\max}	0.003	0.012	0.0	0.003	0.0	0.0	0.0
c_{\sup}	0.744	0.703	0.709	0.708	0.730	0.839	0.661

Tableau 2.5

Résultats des évaluations des deux critères c_{\max} et c_{\sup} pour les différents critères de pureté.

La figure 2.8 représente cette fonction $f(x, y) \forall x, y \in [2; 18]$ pour chacun des critères de pureté. Plus la valeur retournée par la fonction est élevée, plus le couple de résultats est considéré comme pur, c'est à dire de bonne qualité. Dans notre exemple, le nombre de clusters du jeu de données étant 9, il est attendu de cette fonction d'être maximale lorsque les deux résultats comparés ont 9 clusters. Pour évaluer la qualité de ces critères en fonction de ces différents résultats, deux critères ont été développés. Le premier évalue la différence entre la valeur maximale de la fonction et la valeur de la fonction quand les méthodes fournissent des résultats ayant tous les deux le nombre de clusters réel du jeu de données :

$$c_{\max} = \arg \max_{x,y} (f(x, y)) - f(C, C) \quad 2.46$$

Ce critère est maximisé si la valeur de f est maximale quand les deux résultats ont un nombre de clusters identique à celui du jeu de données. Plus ce critère est faible, plus le critère permet d'identifier le nombre de clusters.

Le deuxième critère évalue l'ensemble de la fonction, en pénalisant des valeurs importantes quand le nombre de clusters proposé par les méthodes s'éloigne du nombre de clusters réel. Plus le critère est faible, plus la fonction aura le comportement attendu.

$$c_{\sup} = \frac{\sum_{x,y}^C ((|x - C| + |y - C|) \times f(x, y))}{\sum_{x,y}^C ((|x - C| + |y - C|))} \quad 2.47$$

Ces deux critères ont été calculés à partir des résultats illustrés en figure 2.8 et sont présentés dans le tableau 2.5.

Seul trois des sept indices évalués ne possèdent pas leur maximum à l'endroit attendu. Parmi ces trois indices, sont présents les deux premiers indices de pureté dont la tendance à surévaluer un résultat avec un nombre trop important de clusters a déjà été soulevée. L'évaluation de l'ensemble des résultats à l'aide du deuxième critère permet de prendre en compte le comportement global des critères. Sur ce jeu de données, l'indice Π_{fmesure} permet d'obtenir le meilleur résultat (0.661). L'indice Π_{rand} est le plus mauvais avec un score de 0.839.

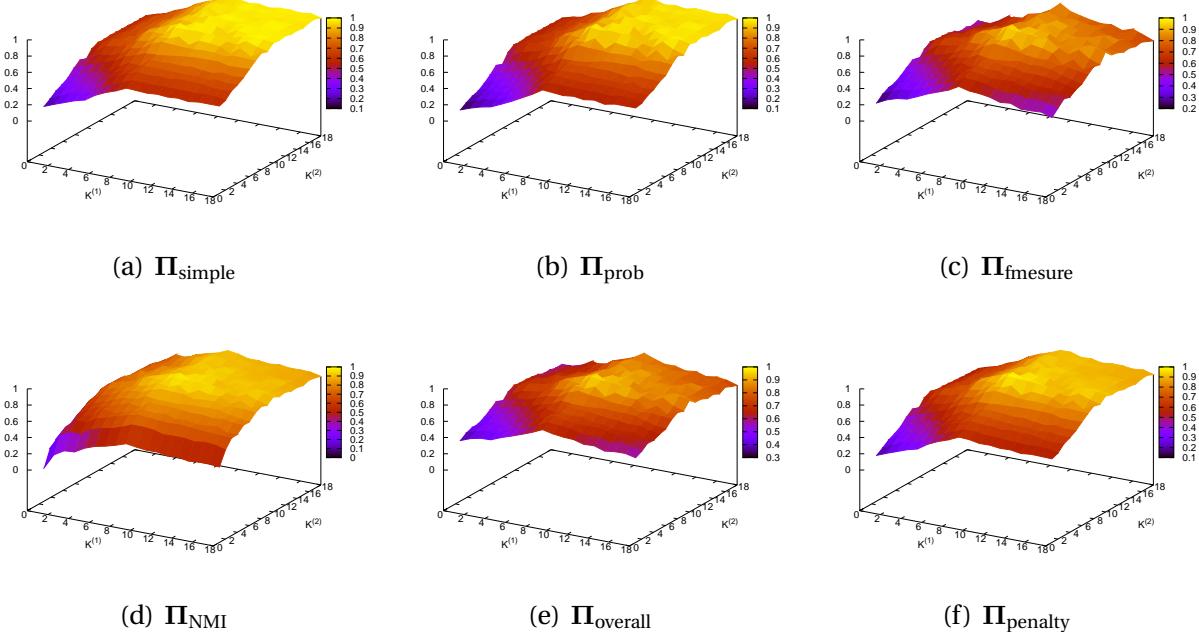


Figure 2.8
Utilisation des différents critères d'évaluation en clustering collaboratif.

2.4 Contributions et valorisation

Dans ce chapitre nous avons présenté nos contributions dans le domaine de la collaboration de méthodes de clustering. Ces recherches ont été validées et valorisées par différentes publications.

Dans (Wemmert et al. 2000), nous présentons la première version de la méthode collaborative SAMARAH. Sa méthode originale de vote multi-points de vue pour la fusion de résultats de clustering est définie plus formellement et évaluée dans (Wemmert & Gançarski 2002*a*). Ensuite, dans (Forestier et al. 2008*a*), nous présentons une extension de SAMARAH permettant l'utilisation de données multi-sources lors de la collaboration. Une étude théorique poussée sur le choix des conflits à résoudre a été menée et est présentée dans (Forestier, Wemmert & Gançarski 2010*b*). Enfin, nous avons défini une extension du cadre collaboratif aux méthodes semi-supervisées et non supervisées qui est publiée dans (Wemmert & Forestier 2011).

Concernant l'utilisation de connaissances expertes, (Forestier et al. 2008*b*) présente une première méthode de clustering semi-supervisé et (Wemmert, Forestier & Derivaux 2009) une méthode combinant classification supervisée et non supervisée, les deux permettant de tirer parti d'un petit ensemble d'exemples étiquetés. Dans (Forestier, Gançarski & Wemmert 2010), nous décrivons de manière formelle comment l'intégration de connaissances expertes peut se faire dans le cas du clustering collaboratif.

La seule vraie science est la connaissance des faits.

Georges-Louis Leclerc de Buffon (1749)

3

Connaissances expertes en observation de la Terre

3.1 Ontologie d'objets géographiques	60
3.1.1 Description de l'ontologie	60
3.1.2 Appariement de région	61
3.2 Connaissances et segmentation	63
3.2.1 Optimisation du paramétrage d'une segmentation	66
3.2.2 Segmentation supervisée	69
3.2.3 Approche hybride : optimisation de segmentation supervisée . . .	72
3.2.4 Comparatif des méthodes	75
3.3 Connaissances et classification	75
3.3.1 Problématique	76
3.3.2 Caractérisation des régions	77
3.3.3 Critères d'évaluation	78
3.3.4 Méthode de classification par connaissances du domaine	79
3.4 Connaissances et détection	81
3.4.1 Interprétation par ensemble de détecteurs	82
3.4.2 Extraction de détecteurs spécifiques à partir de la base de connaissances	83
3.5 Connaissances et clustering collaboratif	86
3.5.1 Problématique	86
3.5.2 Étiquetage des clusters	86
3.6 Contributions et valorisation	88

Ce chapitre s'intéresse à nos travaux sur la classification automatique d'images, et

plus précisément d'images de télédétection. En effet, comme indiqué en préambule, les géosciences et plus particulièrement l'observation de la Terre via les images satellitaires, forment un domaine d'étude très intéressant, présentant de nombreuses applications (étude de l'occupation du sol, des dynamiques urbaines, ou de l'agriculture), et il fournit de nombreuses données hétérogènes et complexes.

Nous présentons tout d'abord nos travaux dans le domaine de la modélisation des connaissances expertes sous forme d'une ontologie d'objets géographiques. En effet, afin de pouvoir utiliser efficacement les connaissances de l'expert, il est nécessaire de les formaliser. Cette première section présente le modèle de représentation choisi ainsi que les opérateurs permettant d'interroger cette base de connaissances.

Ensuite, nous exposons comment ces connaissances peuvent être utilisées pour guider les différentes méthodes d'extraction d'informations à partir d'images. Tout d'abord la segmentation, qui constitue la première étape du processus de classification basée région pour les images à très haute résolution spatiale. C'est elle qui permet de construire les objets qui seront ensuite classés. L'utilisation des connaissances expertes dès cette première étape semble donc très importante. Nous présentons nos propositions sur ce sujet dans la section 3.2. Ensuite, dans la section 3.3, nous présentons comment les connaissances peuvent être utilisées lors de l'étape de classification, étape suivant la segmentation. La section 3.4 quant à elle, expose comment cette base de connaissances peut être utilisée afin de créer un ensemble de détecteurs spécifiques à chaque classe recherchée, plutôt que de procéder de manière classique par segmentation puis classification. Finalement, nous présentons dans la section 3.5 comment la méthode SAMARAH présentée dans le chapitre précédent peut être adaptée afin de pouvoir être guidée par ces connaissances.

3.1 Ontologie d'objets géographiques

La représentation de connaissances géographiques est actuellement au cœur des recherches dans les communautés de représentation des connaissances et d'observation de la Terre. Nous proposons ici un modèle permettant la représentation d'objets géographiques (pavillon, route, végétation ...) sous la forme d'une ontologie. Cette ontologie se présente comme une hiérarchie de concepts ainsi que des relations entre ces concepts. Nous proposons également un processus d'appariement qui permet d'effectuer la comparaison entre une région construite lors d'une segmentation et les différents concepts définis dans l'ontologie. Cet appariement permet d'identifier les concepts dont la description est la plus proche de la région à appairer.

3.1.1 Description de l'ontologie

L'ontologie réalisée est formée d'une hiérarchie de concepts dont la figure 3.1 illustre un extrait. Dans cette hiérarchie chaque noeud correspond à un concept. Chaque concept a une étiquette (par exemple *pavillon*) et est défini par des attributs. Chaque attribut est associé à un intervalle de valeurs acceptées pour cet attribut (par exemple [50; 60])

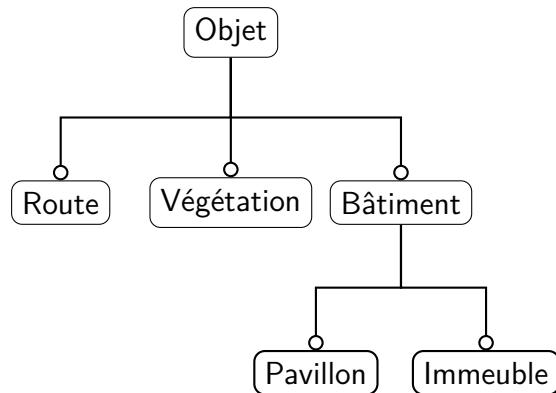


Figure 3.1
Exemple de hiérarchie de concepts d'objets géographiques urbains.

ainsi qu'une pondération (comprise entre 0 et 1) représentant son importance pour reconnaître l'objet géographique correspondant à ce concept (1 indiquant que cet attribut est très pertinent). Le tableau 3.1 présente un exemple avec le concept *pavillon*. Les valeurs de ces concepts ont par exemple été renseignées par les experts géographes grâce à leur connaissance de la morphologie des objets urbains. Certaines informations ont été extraites de bases de données topographiques ou de connaissances sur les réponses spectrales de certains types de matériaux (tuile, bitume, etc.).

3.1.2 Appariement de région

Un mécanisme d'appariement de région permet d'évaluer la similarité entre une région construite lors d'une segmentation et les concepts définis dans la hiérarchie de l'ontologie. Ce mécanisme permet d'obtenir la signification sémantique d'une région si les caractéristiques de celle-ci se rapproche de la description d'un des concepts définis dans l'ontologie. L'appariement d'une région consiste à vérifier la validité des caractéristiques extraites de celle-ci (réponse spectrale, taille, elongation ...) en fonction des propriétés et des contraintes définies dans les concepts de l'ontologie. Le *score d'appariement* d'une région à un concept est calculé sous la forme d'une valeur réelle entre 0 et 1, 1 signifiant que les caractéristiques de la région correspondent tout à fait à la description du concept. Une région peut s'apparier a priori avec n'importe quel concept. Cependant les caractéristiques permettant l'appariement ne sont pas identiques en fonction des concepts. Par exemple le concept *pavillon* est défini par plusieurs types d'indices (spectraux, formes) alors que le concept *ombre* est défini uniquement par des valeurs spectrales. Pour prendre en compte ces spécificités une mesure d'appariement est proposée, basée sur la similarité entre les caractéristiques extraites d'une région et les concepts de l'ontologie. Cette mesure qui calcule la similarité entre les caractéristiques $v_1^r \dots v_n^r$ d'une région r et les attributs $a_1^k \dots a_n^k$ d'un concept k est formée d'une composante locale (à partir des caractéristiques internes du concept) et une composante globale (qui évalue la

Type	Nom de l'attribut	Valeurs
spectral	signature_spectrale_bleue	[21.7-62.3]
	signature_spectrale_verte	[19.4-80.1]
	signature_spectrale_rouge	[29.7-135.1]
	signature_spectrale_proche_infra-rouge	[34.8-139]
	signature_spectrale_SBI	[14.6-60.1]
	signature_spectrale_NDVI	[50.2-108]
spatial	diamètre(m)	[13-61]
	aire (m ²)	[10-600]
	périmètre (m)	[28-116]
	élongation (m)	[1-3.1]
	indice de Miller	[0.5-0.8]
	indice de Solidité	[0.85-1]

Tableau 3.1
Exemple de valeurs pour les attributs du concept “Pavillon”.

pertinence par rapport à la hiérarchie des concepts).

Le *degré de validité* $Valid(v_i^r, a_i^k)$ quantifie la validité entre la caractéristique extraite v_i de la région r et les bornes des valeurs acceptées de l'attribut a_i du concept k .

$$Valid(v_i^r, a_i^k) = \begin{cases} 1 & \text{si } v_i^r \in [min(a_i^k); max(a_i^k)] \\ \frac{v_i^r}{min(a_i^k)} & \text{si } v_i^r < min(a_i^k) \\ \frac{max(a_i^k)}{v_i^r} & \text{si } v_i^r > max(a_i^k) \end{cases} \quad (3.1)$$

La mesure de *similarité locale* $Sim(r, k)$ compare l'ensemble des caractéristiques communes de la région r avec les attributs du concept k . La valeur λ_i^k est le poids de a_i^k et exprime le rôle de a_i^k pour reconnaître k .

$$Sim(r, k) = \frac{\sum_{i=1}^n \lambda_i^k Valid(v_i^r, a_i^k)}{\sum_{i=1}^n \lambda_i^k} \quad (3.2)$$

Le *score d'appariement* $Score(r, k)$ évalue la pertinence de l'appariement entre la région r et le concept k dans la hiérarchie de concepts. Le score d'appariement est une combinaison linéaire des mesures des similarités locales obtenues avec les concepts k_j du chemin partant de la racine de l'ontologie vers le concept étudié k_m . Les similarités locales sont propagées par héritage aux concepts plus spécifiques. Dans ce calcul est intégré un coefficient β_i basé sur la profondeur des concepts. De cette manière, la mesure fa-

vorise la spécialisation des concepts en considérant qu'il est toujours plus intéressant sémantiquement d'obtenir un concept plus bas dans la hiérarchie.

$$Score(r, k_m) = \frac{\sum_{j=1}^m \beta_j Sim(r, k_j)}{\sum_{j=1}^m \beta_j} \quad [3.3]$$

Pour parcourir l'ontologie à la recherche de concepts s'appariant avec la région étudiée, un mécanisme de parcours de l'ontologie a également été mis en place. Ce mécanisme permet de trouver les meilleurs concepts pour la région étudiée. Lors du parcours plusieurs heuristiques sont utilisées pour explorer la hiérarchie de l'ontologie et réduire l'espace de recherche. Il est par exemple possible de définir un score minimum à partir duquel on souhaite qu'une région soit considérée comme appartenant à un concept.

Le processus général de l'exploration est le suivant : si la région peut correspondre à la classe courante, l'algorithme va descendre d'un niveau dans la hiérarchie définie par l'ordre partiel \preceq_Θ . Si la région ne peut pas correspondre à la classe courante, celle-ci est abandonnée et ses sous-classes ne seront pas explorées.

Nous définissons le seuil $minScore$ comme étant la valeur minimale du score de correspondance entre une région et une classe pour considérer possible que la région soit effectivement un objet de cette classe.

Soit $\mathcal{L} : \Theta \rightarrow \Theta$ tel que $\mathcal{L}(R)$ est l'ensemble des classes qui peuvent correspondre à la région R tenant compte du seuil $minScore$.

$$\mathcal{L}(R) = \{(C_i, Score(R, C_i)) \mid Score(R, C_i) \geq minScore\} \quad [3.4]$$

L'exploration de la hiérarchie de classes est présentée dans l'algorithme 4. Ce processus peut être répété pour chaque région d'une image segmentée afin de donner une interprétation de l'image complète.

3.2 Connaissances et segmentation

Dans le cadre de la télédétection, l'augmentation rapide de la résolution spatiale (taille de l'image) ainsi que de la résolution spectrale (nombre de bandes) accroît la complexité des images disponibles et nécessite le développement de méthodes spécifiques, les méthodes classiques de classification par pixels devenant inefficaces sur ce type de données. L'approche la plus prometteuse et la plus étudiée actuellement est l'approche dite *orientée objet* qui consiste à identifier dans l'image des objets composés de plusieurs pixels connexes (régions ou segments) et ayant un intérêt pour l'expert du domaine.

Cette approche se compose de deux étapes :

- la segmentation : cette étape consiste à regrouper les pixels voisins dans l'image ayant une valeur spectrale homogène. L'ensemble des régions ainsi trouvé forme un pavage de l'image. Chacune de ces régions est ensuite caractérisée par un en-

Algorithme 4: Exploration de la hiérarchie de classes pour l'affectation d'une région

Entrées : une région R , une hiérarchie de classes Θ , un seuil $minScore$

$\mathcal{L}(R) = \emptyset;$

$\mathcal{RC} = \{\text{racine}\};$

tant que ($\mathcal{RC} \neq \emptyset$) **faire**

meilleursProfondeur = \emptyset ;

pour tous les $C_i \in \mathcal{RC}$ **faire**

$s = \text{Score}(R, C_i);$

si ($s \geq minScore$) **alors**

$\mathcal{L}(R) = \mathcal{L}(R) \cup \{(C_i, s)\};$

meilleursProfondeur = meilleursProfondeur $\cup \{C_i\};$

$\mathcal{RC} = \emptyset;$

pour tous les $C_j \in \text{meilleursProfondeur}$ **faire**

$\mathcal{RC} = \mathcal{RC} \cup \{C_i | C_i \preceq_{\Theta} C_j\};$

renvoyer $\mathcal{L}(R)$

semble d'attributs liés aux réponses spectrales des pixels (e.g. textures) composants la région ou à la forme (e.g. taille, élongation) de la région.

- la classification : cette étape consiste à affecter à chaque région obtenue une classe sémantique en se basant sur leurs caractéristiques.

La ligne de partage des eaux (Vincent & Soille 1991) est la méthode principale de segmentation de morphologie mathématique. L'image est considérée comme une carte d'altitude représentant les intensités des pixels (i.e. réponse spectrale). Le gradient de l'image est utilisé pour distinguer les zones homogènes et hétérogènes de l'image étudiée. Plus un pixel est situé dans une zone hétérogène, plus son gradient et donc ici son élévation seront importants. Ce relief est ensuite inondé à partir de ses minima pour créer les régions. Lorsque deux bassins de rétention se rencontrent, une ligne de partage des eaux est créée pour les séparer. Un exemple de coupe d'une image d'élévation et de ses minima est donné en figure 3.2(a).

Pour construire l'image de gradient, chaque pixel est remplacé par la différence entre la valeur maximale et minimale d'une fenêtre de taille 3×3 centrée sur le pixel. L'image d'élévation finale est obtenue en combinant les élévations des différentes bandes spectrales à l'aide de la norme euclidienne. Si le gradient de la i^{e} bande est noté G^i et N_B le nombre de bandes, le gradient final est défini par :

$$G = \sqrt{\sum_{i=0}^{N_B} G_i^2} \quad 3.5$$

La ligne de partage des eaux a l'avantage d'être une méthode complètement non supervisée et sans paramètre. Néanmoins, elle fournit généralement une image sur-

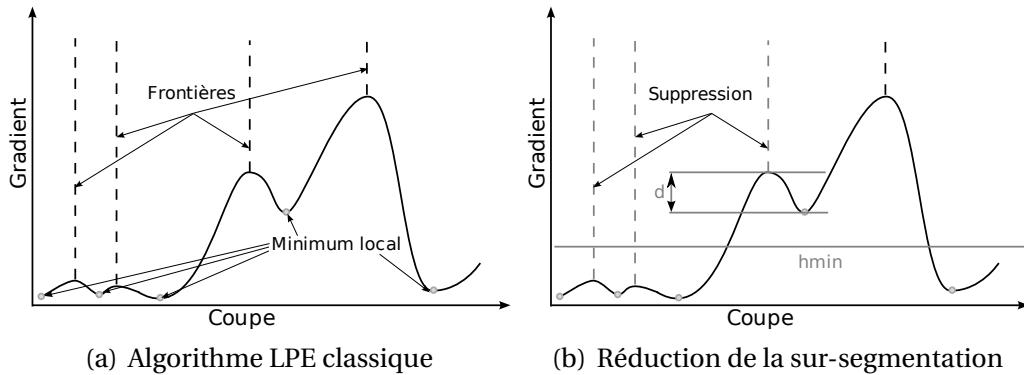


Figure 3.2

Principe de fonctionnement de l'algorithme de segmentation de la ligne de partage des eaux et de son extension paramétrée pour la réduction de la sur-segmentation.

segmentée, c'est-à-dire une image où chaque objet (i.e une maison) est représenté par plusieurs régions (i.e. les deux versants de son toit). Pour résoudre ce problème, plusieurs techniques peuvent être utilisées indépendamment ou conjointement.

Une fois l'image de gradient calculée, un seuillage du gradient (Haris et al. 1998) peut être effectué. Tout pixel ayant une valeur inférieure à un seuil est mis à zéro. Ainsi, les petites variations au sein des zones homogènes sont supprimées. Sur la figure 3.2(b), la ligne h_{min} représente le seuil en dessous duquel la valeur des pixels sera considérée comme nulle.

Une autre technique consiste en l'utilisation de la profondeur des bassins (Najman & Schmitt 1996). Soit m_r l'altitude du minimum local du bassin r et d_r l'altitude minimale à partir de laquelle il sera séparé d'un autre bassin par une ligne de partage des eaux. Tous les minima locaux pour lesquels on a $d_r - m_r < d$, avec d un seuil donné, ne seront pas considérés pendant l'étape d'immersion des bassins. Sur la figure 3.2(b), le minimum local ne sera pas pris en compte lors de l'immersion car sa dynamique, la différence entre m_r et d_r , n'est pas assez importante.

Enfin, il est également possible d'utiliser une technique de fusion de régions (Haris et al. 1998). Deux régions peuvent être séparées par une zone hétérogène (se traduisant par la génération d'une frontière par la ligne de partage des eaux) tout en étant néanmoins spectralement similaires. Pour résoudre ce problème, il est possible d'utiliser un filtre qui fusionne les régions connexes dont la distance euclidienne entre moyennes spectrales est inférieure à un seuil.

Ces différentes techniques peuvent être utilisées simultanément pour réduire la sur-segmentation provoquée par la ligne de partage des eaux et nécessite donc de définir 3 paramètres (le niveau de h_{min} , le seuil d et le seuil de fusion). Les valeurs optimales de ces paramètres sont difficiles à trouver car la valeur pour un paramètre donné dépend des valeurs choisies pour les autres paramètres. De plus, il existe de nombreux optima locaux, ce qui accroît la difficulté de trouver la meilleure solution.

Problématique Un premier problème à résoudre lors de la phase de segmentation est celui du réglage des paramètres. Le choix des paramètres de segmentation est un problème important (Darwish et al. 2003), l'utilisateur devant en général procéder par essais-erreurs jusqu'à obtenir un résultat convenable. La méthode que nous proposons détermine automatiquement les paramètres à l'aide d'un algorithme évolutionnaire guidé par des exemples d'apprentissage. Elle permet de simplifier la tâche de l'utilisateur et d'obtenir de meilleurs résultats.

Ensuite, l'hypothèse souvent faite par les algorithmes de segmentation considérant qu'un objet sémantique donné est composé de pixels connexes similaires, n'est pas valide dans le cas de l'extraction d'objets hétérogènes et complexes. Cette hypothèse est remise en cause en introduisant une connaissance exogène dans le processus, sous forme d'une classification floue basée pixels afin de transformer l'espace des données. Nous utilisons ainsi une vision alternative et plus proche de la sémantique des données. Chaque pixel est représenté non plus par ses attributs visuels propres à l'image, mais par des appartenances à différentes classes sémantiques. Appliquer un algorithme de segmentation sur un tel espace permet d'obtenir des segmentations plus conformes aux attentes.

Enfin, nous proposons d'utiliser conjointement ces deux méthodes (optimisation stochastique des paramètres et transformation de l'espace des données). Les résultats obtenus mettent en avant une forte synergie entre les méthodes. Nous développons ainsi une nouvelle méthode de segmentation ne nécessitant aucun paramètre mais guidée par des exemples donnés par l'utilisateur expert. Les résultats s'en trouvent plus pertinents et correspondent mieux à ses attentes.

Cette section présente tout d'abord une méthode de paramétrage automatique d'un algorithme de segmentation par utilisation de connaissances, puis le *probashed* qui, via une transformation de l'espace de représentation de l'image, permet de construire des segments non nécessairement composés de pixels homogènes en terme de réponse spectrale, mais plus proches des objets sémantiques attendus par l'utilisateur, et finalement une approche hybride intégrant les deux méthodes proposées.

3.2.1 Optimisation du paramétrage d'une segmentation

Les algorithmes génétiques (Goldberg 1989) représentent une solution générique au problème de recherche des paramètres optimaux. En effet, ils peuvent être utilisés pour optimiser un ensemble d'attributs dès lors qu'une fonction d'évaluation des paramètres est disponible. Les méthodes existantes d'optimisation de segmentation par approche génétique (Pignalberi et al. 2003, Bhanu et al. 1995, Song & Ciesielski 2003, Feitosa et al. 2006) se basent sur des fonctions d'évaluations demandant des exemples d'objets segmentés fournis par l'expert. Si aucun exemple n'est disponible, il est possible d'utiliser des critères non supervisés (Bhanu et al. 1995, Feitosa et al. 2006), c'est-à-dire jugeant la qualité intrinsèque que doit avoir une segmentation (par exemple l'homogénéité des régions). Néanmoins ces critères non supervisés sont souvent insuffisants pour obtenir une segmentation de bonne qualité notamment pour l'analyse d'images complexes.

Nous proposons d'utiliser des connaissances du domaine afin d'évaluer la qualité d'une segmentation. En effet, l'approche orientée objet permet à l'expert d'exprimer ses connaissances sur les objets de l'image. Ce nouveau cadre de discernement permet de raisonner sur des régions et non sur des pixels, ce qui autorise l'utilisation d'une description intuitive et naturelle des objets pouvant être présents dans une image. Comme indiqué précédemment (section 3.1), une ontologie, ou base de connaissances, peut être utilisée pour définir les différents types d'objets du domaine (i.e. concepts) de l'image ainsi que leurs caractéristiques. Il devient alors possible d'évaluer la cohérence d'une segmentation par rapport aux concepts définis dans cette ontologie. Cette approche a l'avantage de ne pas nécessiter d'exemples et utilise la connaissance définie dans l'ontologie.

Nous avons vu précédemment qu'un algorithme de segmentation efficace nécessite la définition de plusieurs paramètres. Nous nous intéressons ici à l'utilisation d'un algorithme génétique afin de trouver les paramètres de l'algorithme de segmentation de manière automatique, en utilisant les connaissances de l'ontologie. Nous montrons tout d'abord le fonctionnement générique d'un algorithme génétique puis la fonction d'évaluation choisie pour résoudre ce problème.

Algorithme génétique

Un algorithme génétique peut être vu comme une méthode d'optimisation. Supposons connue \mathbb{F} une fonction d'évaluation prenant un paramètre noté g à valeurs dans un espace \mathbb{G} . Le but d'une fonction d'optimisation est de trouver la valeur de g pour laquelle $\mathbb{F}(g)$ est maximale. Les algorithmes génétiques sont réputés pour être efficaces même lorsque \mathbb{G} est vaste et contient de nombreux maxima locaux.

Un algorithme génétique nécessite une population initiale qu'il va faire évoluer pour arriver à des solutions maximisant la fonction d'évaluation. La population initiale est ici choisie aléatoirement sauf un individu pour lequel toutes les valeurs des paramètres seront fixées. Ceci permet de garantir que la solution proposée par l'algorithme génétique sera aussi bon que les paramètres définis par défaut.

Une fois la population initiale définie, l'algorithme va appliquer les étapes suivantes qui représentent le passage d'une génération à une autre :

1. évaluation des individus par la fonction d'évaluation ;
2. sélection des individus pour la reproduction pondérée par leur classement obtenu lors de l'évaluation afin de privilégier les meilleurs individus ;
3. reproduction : deux individus parents (p_1 et p_2) se reproduisent en combinant leurs gènes et génèrent un individu enfant e . Pour chaque paramètre g_i , $g_i(e)$ aura une chance égale de prendre pour valeur soit $g_i(p_1)$, soit $g_i(p_2)$ ou soit $0.5 \times g_i(p_1) + 0.5 \times g_i(p_2)$;
4. mutation : chaque paramètre g_i de chaque individu a une probabilité \mathbb{P}_m d'être remplacé par une valeur aléatoire ;

Ici, g , aussi appelé génotype dans le cadre des algorithmes génétiques, représente le vecteur des paramètres de la méthode de segmentation. Nous considérons que les valeurs de ces paramètres sont définies entre zéro et un, et on a donc $\mathbb{G} = [0; 1]^3$ pour la ligne de partage des eaux (le niveau de h_{min} , le seuil d et le seuil de fusion).

Nous utilisons un taux de mutation $\mathbb{P}_m = 1\%$ et un nombre de générations de 14, des tests ayant montré que plus de générations n'amélioraient pas les résultats.

Choix de la fonction d'évaluation

La définition de la fonction d'évaluation est une des étapes les plus importantes dans un système d'évolution génétique. Nous cherchons ici à utiliser les connaissances de l'ontologie pour guider le processus évolutif et trouver les paramètres qui permettent de maximiser la découverte d'objets dans l'image. Nous allons donc utiliser comme fonction d'évaluation, le pourcentage de la surface de l'image qui est identifié par l'ontologie. Chaque génotype g va donc produire un phénotype correspondant à une segmentation de l'image. Soit \mathcal{R}^g les régions d'une segmentation et \mathcal{R}_o^g les régions identifiées par l'ontologie ($\mathcal{R}_o^g \subseteq \mathcal{R}^g$). Le pourcentage de la surface de l'image reconnu par l'ontologie est défini par :

$$\mathbb{F}(g) = \frac{\sum_{r \in \mathcal{R}_o^g} Aire(r)}{\sum_{r \in \mathcal{R}^g} Aire(r)} \quad [3.6]$$

avec $Aire(r)$ une fonction renvoyant la surface en pixel de la région r . La surface des régions identifiées a été préféré à leur nombre pour évaluer le résultat. En effet, une segmentation peut produire de nombreuses petites régions qui n'ont pas de signification sémantique et qui ne sont donc pas reconnus par l'ontologie. Ces petites régions peuvent perturber un calcul basé sur le nombre de régions. Une grande partie de l'image peut être reconnue si de grandes régions (par exemple l'étendue végétale) sont reconnus par l'ontologie. Ce résultat peut être faussé si de petites régions sans sémantique sont présentent dans le reste de l'image segmentée. En effet, les régions ont généralement des tailles différentes et de nombreuses petites régions ne sont pas identifiées. Se baser sur leur nombre ne permet donc pas de juger correctement la qualité de la reconnaissance.

Le critère basé sur la surface des régions permet de quantifier la qualité de la segmentation par rapport à la connaissance représentée dans l'ontologie. L'augmentation de ce critère signifie que les régions construites par la segmentation correspondent de mieux en mieux à la description des objets géographiques présents dans l'ontologie. En maximisant ce critère nous nous assurons de construire une segmentation représentant les connaissances des experts sur les objets géographiques. Nous cherchons ici à maximiser la surface de l'image reconnue par l'ontologie. Une série d'expérimentations a été menée pour valider cette approche dont les résultats sont présentés dans la section suivante.

L'étape la plus sensible dans un système par évolution génétique est la définition de la fonction d'évaluation. L'évaluation de la qualité d'une segmentation est une tâche ardue tant il existe de caractéristiques d'évaluations (Zhang 1996). Nous avons choisi d'évaluer la qualité de la segmentation par la précision d'une classification supervisée en se basant sur la segmentation obtenue. Cela nous semble particulièrement pertinent puisque

notre objectif final est l'interprétation de l'image à l'aide d'une classification supervisée. La fonction d'évaluation est donc :

$$\mathbb{F}(g) = \frac{\sum_{i=0}^{\bar{C}} K_{ii}^g}{\sum_{i=0}^{\bar{C}} \sum_{j=0}^{\bar{C}} K_{ij}^g} \quad (3.7)$$

avec K^g la matrice de corrélation issue de la classification en \bar{C} classes de la segmentation obtenue par l'individu g .

L'évaluation de cette précision s'effectue sur un jeu de données différent de celui utilisé pour l'apprentissage du classifieur et si besoin de l'algorithme de segmentation. Une évaluation par validation croisée est utilisée. Ainsi, si l'on dispose de n zones d'apprentissage, chacune de celles-ci est utilisée comme zone d'évaluation pendant que les $n - 1$ restantes sont utilisées pour l'apprentissage de l'algorithme de segmentation et du classifieur. L'évaluation finale est obtenue en prenant la moyenne de chacune des n évaluations ainsi réalisées.

3.2.2 Segmentation supervisée

À la section précédente, nous avons proposé une méthode permettant d'optimiser des paramètres de segmentation en fonction de connaissances du domaine. Dans l'évaluation analytique de l'algorithme il apparaît que les limites de l'algorithme de segmentation de base utilisé ne peuvent être évitées par cette approche. Dans cette section, nous proposons une approche différente, utilisant des méthodes de fouille de données au sein même de l'algorithme de segmentation.

Pour introduire cette approche, commençons par rappeler les limites des algorithmes classiques de segmentation. Comme nous l'avons vu, un algorithme de segmentation cherche à créer des regroupements de pixels spatialement connectés selon un critère d'homogénéité. L'hypothèse est donc faite que des pixels connexes ayant des valeurs différentes n'appartiennent pas au même objet. Cette hypothèse est valide pour les images où les objets d'intérêt sont représentées par des pixels similaires. Pour des images complexes, cette hypothèse n'est plus valide. Un objet peut être très hétérogène ; si l'on prend le cas d'un toit de maison, celui-ci peut avoir des fenêtres (de couleur différente au reste du toit) et une illumination différente de chaque côté du toit selon l'exposition au soleil. L'hypothèse faite n'est pas valide dans le cas de ces images et les résultats ne sont donc pas pertinents.

Nous proposons une nouvelle méthode de segmentation qui utilise des régions de référence pour apprendre une nouvelle notion de pixels homogènes. Pour ce faire, nous projetons les données à segmenter dans un espace de représentation plus adéquat où l'hypothèse de similarité est valide. Dans l'espace ainsi créé, les algorithmes classiques de segmentation peuvent être appliqués et doivent fournir des résultats plus proches de ceux escomptés.

Nous allons tout d'abord donner un état de l'art des méthodes pouvant se rapprocher de notre proposition et nous positionner par rapport à celles-ci. Ensuite, nous expliciterons le fonctionnement de l'algorithme.

État de l'art et positionnement

Plusieurs méthodes se proposent d'introduire des connaissances via des régions de références dans un algorithme de segmentation. Dans l'algorithme de segmentation par ligne de partage des eaux, la connaissance est souvent introduite en utilisant des marqueurs (Lefèvre 2007, Meyer & Beucher 1990) et/ou en appliquant la ligne de partage des eaux sur une image modifiée. Les marqueurs modifient légèrement le fonctionnement de la ligne de partage des eaux. Au lieu d'être inondée à partir de ses minima locaux, la surface topographique sera inondée uniquement à partir des marqueurs.

Haker et al. (2000) utilisent des images préalablement segmentées afin d'extraire les probabilités a priori de l'appartenance aux classes d'intérêt et les combinent selon la règle de Bayes. D'autres connaissances peuvent être intégrées de cette façon comme les positions spatiales approximatives des objets d'intérêt. Si la position d'un objet est connue, il est possible d'augmenter la probabilité a priori de la classe de l'objet recherché aux positions où l'objet est attendu. Cette méthode est similaire à une classification supervisée et donc sujette à la sous-segmentation, mais permet d'améliorer les résultats quand l'utilisateur sait déterminer approximativement la localisation des différents objets.

Levner & Zhang (2007) proposent une méthode utilisant des cartes de probabilité. Ils utilisent une première classification, basée sur des régions de référence érodées pour trouver des marqueurs. Une autre classification est effectuée en utilisant les régions de référence non modifiées fournissant une carte de probabilités. L'inverse de cette carte de probabilités est utilisée comme une surface topographique pour appliquer la ligne de partage des eaux en utilisant les marqueurs obtenus précédemment. Cette méthode n'est appliquée que pour de la classification binaire (une classe d'intérêt et un fond). Elle fait l'hypothèse qu'un marqueur sera détecté pour chaque objet d'intérêt. Si aucun marqueur n'est détecté pour un objet, il ne pourra être segmenté.

Grau et al. (2004) utilisent une carte de probabilité pour chaque classe d'intérêt. Des marqueurs sont générés en utilisant un atlas. Chaque marqueur a une classe associée. Une approche par croissance de région est utilisée pour simuler le processus d'inondation de la surface topographique. L'élévation entre deux pixels est dépendante du marqueur courant. L'approche utilise la différence de probabilité de ces pixels dans la carte de probabilités associée à la classe du marqueur (processus markovien). Cette approche nécessite de connaître la position des marqueurs.

D'autres formes d'utilisation de connaissances dans le processus de segmentation ont été proposées. Hamarneh & Li (2007) effectuent une ligne de partage des eaux classique. Cette première segmentation est donc affectée par le problème de sur-segmentation inhérent à la ligne de partage des eaux. Ils utilisent ensuite un algorithme de clustering de type k-means regroupant les régions à partir de leur intensité et de leur position. En utilisant des connaissances sur l'apparence des objets recherchés, ils sélectionnent le regrou-

pement de régions approprié et alignent un histogramme de forme pour ôter les régions inadéquates. Cette approche repose fortement sur l'assertion que les objets recherchés ont des valeurs d'intensité homogènes.

Chen et al. (2003) extraient un modèle de forme et d'intensité de l'objet d'intérêt à partir d'un ensemble de régions de référence. Après cette étape d'apprentissage, ils utilisent un modèle de contour actif pour segmenter l'objet recherché en se basant sur le modèle de forme et d'intensité défini précédemment. Cette méthode n'est applicable que pour les problèmes de détection d'un seul objet dont la localisation approximative est connue.

De cet état de l'art, nous pouvons noter que l'idée de l'introduction de connaissances dans le processus de segmentation a donné lieu récemment à plusieurs propositions. Nous constatons que toutes les méthodes (hormis la classification supervisée) soit nécessitent de connaître la localisation des objets, soit ne peuvent segmenter qu'un seul objet (ou une classe d'objets).

L'approche que nous proposons diffère de ces approches sur plusieurs points :

- la capacité à prendre en compte plusieurs classes d'objets ;
- aucune connaissance n'est nécessaire sur la localisation des objets.

Présentation de l'algorithme

L'algorithme proposé, appelé *probashed* par la suite, fonctionne en deux étapes. La première étape consiste à transformer l'espace des données de l'image d'entrée. Cette étape est réalisée par une classification floue supervisée afin d'introduire de la sémantique dans la représentation des données. Cet algorithme utilise les pixels des régions de référence données et affecte à chaque pixel de l'image une probabilité d'appartenir à chacune des classes possibles. La seconde étape est la segmentation de ces cartes d'appartenances, réalisée par un algorithme classique de segmentation.

Afin d'expliquer le paradigme utilisé dans cet algorithme, considérons que nous disposons d'un classifieur parfait (c'est-à-dire qui ne fait pas d'erreur). Une classe est assignée à chaque pixel en entrée. Ainsi, une région est créée à partir de chaque composante connexe de pixels avec des étiquettes identiques. Il n'y a pas de sous-segmentation, puisque deux pixels de classes différentes ne peuvent être regroupés ensemble. Il n'y a pas de sur-segmentation non plus car deux pixels connexes de la même classe sont toujours regroupés ensemble.

Soit S_i l'espace d'entrée défini par l'équation 3.8 (de dimension $\Omega(i)$). Il n'existe pas de fonction qui pour un pixel de S_i affecte une classe à ce pixel sans jamais faire d'erreur. Comme il n'existe que des fonctions approximatives, soit S_m l'espace des appartennances défini par l'équation 3.9. Dans l'espace des appartennances, chaque classe d'objets (soit $\Omega(\mathcal{C})$ le nombre de classes) contenue dans l'image est une dimension. La valeur dans chaque dimension dénote l'appartenance du pixel à la classe d'objets correspondante.

$$\begin{aligned} S_i : \quad I &\rightarrow \mathbb{R}^{\Omega(i)} \\ p &\mapsto S_i(p) \quad \text{avec } S_i(p) \text{ la valeur} \\ &\quad \text{spectrale du pixel } p \end{aligned} \tag{3.8}$$

$$\begin{aligned} S_m : I &\rightarrow [0; 1]^{\Omega(\mathcal{C})} \\ p &\mapsto S_m(p) \quad \text{avec } S_m(p) \text{ le vecteur} \\ &\quad \text{d'appartenances du pixel } p \end{aligned} \tag{3.9}$$

La transformation de S_i à S_m est apprise en utilisant des outils de fouille de données capables d'apprendre cette fonction à partir des pixels de référence. Dans notre cas, nous utilisons l'algorithme des k plus proches voisins (Aha et al. 1991). Pour chaque pixel d'entrée p , les k (avec $k = 5$) plus proches voisins dans l'espace S_i sont sélectionnés. Chaque pixel voisin p_n augmente le niveau d'appartenance du pixel p pour la classe à laquelle il appartient, pondéré par l'inverse de la distance $d(p, p_n)$ dans l'espace des données. L'appartenance du pixel p à la classe C_i , notée m_{p,C_i} , est obtenue par :

$$m_{p,C_i} = \left(\sum_{n=1}^k \sum_{l=1}^{\Omega(\mathcal{C})} w_{n,l} \right)^{-1} \sum_{n=1}^N w_{n,C_i} \tag{3.10}$$

$$\text{où } w_{n,C_i} = \begin{cases} d(p, p_n)^{-1} & \text{si } p_n \text{ est étiqueté par } C_i \\ 0 & \text{sinon} \end{cases}$$

Les données des pixels projetés dans l'espace S_m peuvent être considérées comme des images multi-valuées. En effet, chaque pixel est associé à un vecteur d'appartenances aux classes. On peut donc créer une image où chaque bande contient les appartenances de chaque pixel à une classe donnée.

Cette image ainsi obtenue peut donc être segmentée par des techniques classiques.

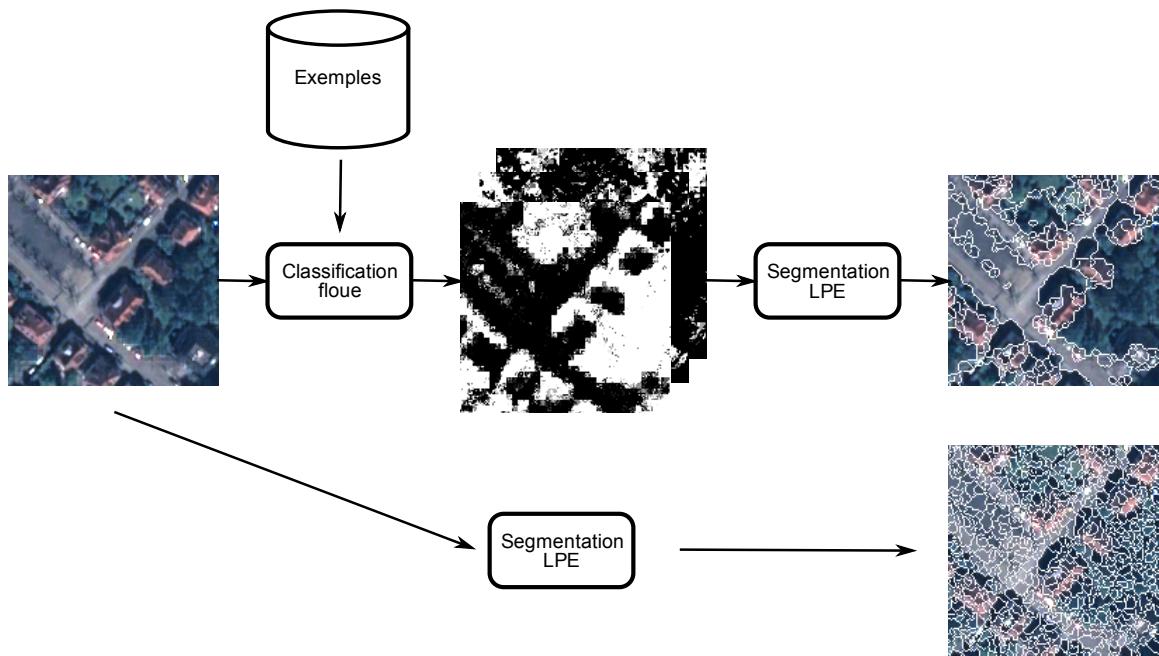
Le schéma sur la figure 3.3 présente l'approche *probashed* en comparaison avec la ligne de partage des eaux classique sur un exemple d'image de télédétection.

3.2.3 Approche hybride : optimisation de segmentation supervisée

Nous avons vu une méthode d'apprentissage des paramètres à la section 3.2.1 et une méthode de segmentation supervisée à la section 3.2.2. Dans cette section, nous présentons une approche hybride, combinant les avantages de chacune d'entre elles.

La méthode (Derivaux et al. 2007) est constituée de deux phases comme illustrée à la figure 3.4. Dans la première phase, l'algorithme apprend comment segmenter une image en utilisant un jeu d'apprentissage. Le processus d'apprentissage intervient en deux endroits, pour la transformation d'espace et pour le choix des paramètres. Une fois l'apprentissage terminé, l'algorithme de segmentation ainsi créé (composé de l'étape de transformation d'espace et de l'algorithme de segmentation de base) peut être utilisé pour segmenter des images. Il n'y a plus besoin d'exemples d'apprentissage à cette phase. De plus, la méthode ne nécessite aucun paramètre dans les deux phases.

Nous avons déjà vu la méthode de transformation d'espace à la section 3.2.2. Cette méthode transforme l'espace des valeurs brutes de l'image en un espace d'appartenances aux classes plus sémantique. L'utilisation de la méthode d'apprentissage de paramètres


Figure 3.3

Comparaison du probashed avec un algorithme de ligne de partage des eaux classique sur une image de télédétection.

présentée à la section 3.2.1 va permettre d'introduire une nouvelle étape. En effet, au lieu d'inférer les appartenances des pixels aux classes à partir des valeurs brutes des pixels, nous allons utiliser les valeurs brutes complétées par de nouveaux attributs. Ayant plus d'informations, notre objectif est d'avoir des appartenances aux classes plus précises. Nous introduisons donc l'espace S_e défini par :

$$S_e : \mathbb{R}^{\Omega(i)} \rightarrow \mathbb{R}^{\Omega(e)}$$

$$S_i(p) \mapsto S_e(p) \text{ avec } S_e(p) \text{ le vecteur étendu des attributs du pixel } p \quad (3.11)$$

En plus des valeurs brutes des pixels de l'image, l'espace S_e contient les attributs suivants :

- intensité relative : chaque bande c , prend la valeur de celle-ci divisée par la somme des valeurs des bandes ;
- intensité moyenne : c'est la moyenne de la valeur des bandes ;
- analyse en composante principale : l'ACP est une méthode qui recherche les directions de l'espace qui représentent le mieux les corrélations (Volle 1997).

Le problème d'une telle approche tient à la méthode d'apprentissage que nous avons utilisée, celle des k plus proches voisins. Cette méthode a pour défaut d'affecter une importance identique à tous les attributs caractérisant les pixels. Cela pose un problème quand on augmente le nombre d'attributs. En effet, certains attributs sont moins im-

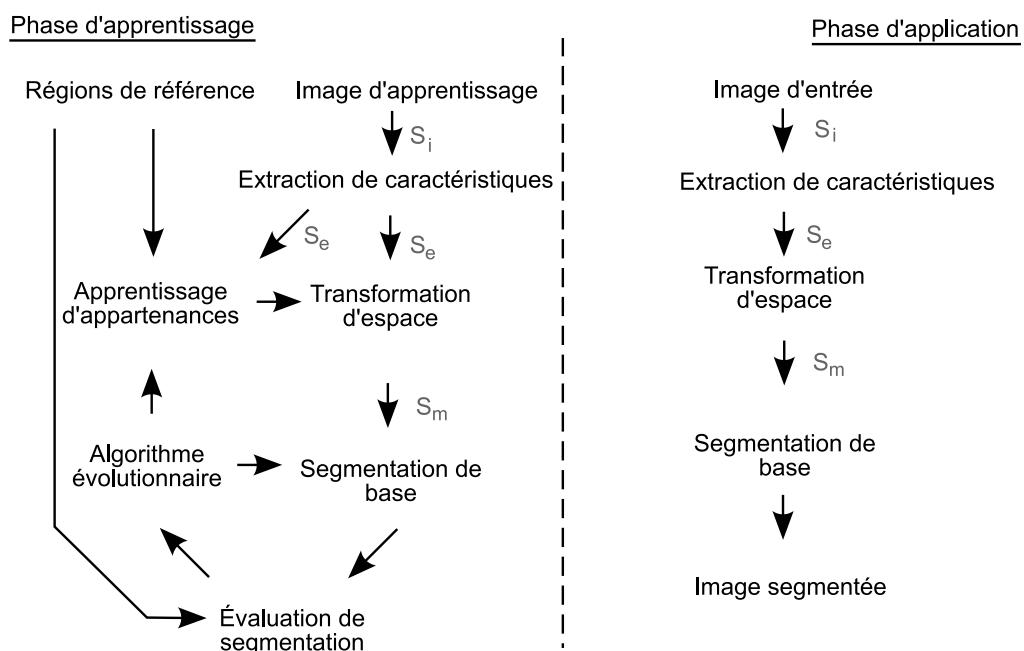


Figure 3.4

Diagramme de flux de l'algorithme proposé pour la phase d'apprentissage et la phase d'application.

portants que d'autres. Certains peuvent n'être que du bruit (pour le problème donné) ou plusieurs attributs peuvent être corrélés (Blansché et al. 2005). Pour éviter ce type de problèmes, l'étape d'*extraction de caractéristiques* considère aussi un vecteur de poids, un poids pour chaque dimension de l'espace S_e , afin de pondérer les attributs (par multiplication de la valeur de l'attribut par la pondération). Ainsi, les attributs n'apportant pas d'informations pertinentes pourront voir leur pondération réduite et donc ne pas influencer le résultat. La modification de la fonction de distance permet d'accroître la précision de la méthode des k plus proches voisins (Paredes & Vidal 2006). Le nombre d'attributs (nombre de dimensions de S_e) étant $(3 \times B) + 1$, avec B le nombre de bandes de l'image, il faut donc un nombre de poids identique.

Pour l'algorithme de segmentation de base, nous utilisons le même algorithme évolutionnaire que celui présenté à la section 3.2.1, c'est à dire une ligne de partage des eaux suivie d'une étape de fusion de régions. Cet algorithme de segmentation prend trois paramètres. La différence est que le nombre de paramètres à optimiser passe de 3 à $(3 \times B) + 4$ (dont $(3 \times B) + 1$ pour la pondération des attributs et 3 pour l'étape de segmentation).

3.2.4 Comparatif des méthodes

Nous présentons ici (figure 3.5) un résultat permettant de comparer sur un exemple les différences entre les trois approches proposées et l'algorithme de la ligne de partage des eaux standard. La méthode d'optimisation des paramètres de segmentation (section 3.2.1) conduit à une image moins sur-segmentée que la LPE, mais qui reste très sur-segmentée malgré tout. Le résultat proposé par le *probashed* (segmentation supervisée présentée section 3.2.2) est de meilleure qualité (évaluation visuelle comparativement à une vérité terrain). La route est mieux construite ainsi que les pavillons. De même la région de végétation au nord-ouest est reconstruite en un seul segment. Finalement, on peut constater que l'approche hybride (optimisation de segmentation supervisée, section 3.2.3) propose le meilleur résultat visuellement, les objets d'intérêts étant relativement bien identifiés.

3.3 Connaissances et classification

La classification est une étape postérieure à la segmentation. Elle consiste à affecter une étiquette de classe à chacune des régions issues de l'étape de segmentation. Il a été démontré qu'affecter des étiquettes à des régions est plus intéressant que de travailler directement à partir des pixels, notamment pour l'imagerie satellitaire à très haute résolution (Cleve et al. 2008, Gigandet et al. 2005, Whiteside & Ahmad 2005, Liu et al. 2006).

Dans cette section, nous présentons la problématique de la classification de régions issues d'une segmentation. Nous abordons ensuite la définition des différents attributs permettant de caractériser les régions de la segmentation et introduisons des mesures



Figure 3.5

Comparaison des différentes approches de segmentation proposées à la ligne de partage des eaux classiques.

permettant d'évaluer la qualité des classifications obtenues. Enfin, nous décrivons notre proposition de méthode de classification guidée par des connaissances du domaine.

3.3.1 Problématique

Soit \mathcal{X} l'espace des objets à classer (dans notre cas des régions issues de la segmentation) et \mathbb{C} l'ensemble des classes possibles. L'objectif est de créer une fonction c , qui, à un objet x de \mathcal{X} , associe une classe C_x de \mathbb{C} . Plus formellement :

$$\begin{aligned} c: \quad \mathcal{X} &\rightarrow \mathbb{C} \\ f(x) &\mapsto C_x \end{aligned} \tag{3.12}$$

Il existe deux approches classiques pour construire cette fonction c . La construction par un expert et l'inférence à partir d'exemples.

La création par un expert, approche utilisée dans les systèmes experts, fonctionne de la manière suivante : l'expert formalise ses connaissances du domaine sous forme de règles, d'ontologies (Wang et al. 2002) ou de systèmes à base de frames (Minsky 1975). Ces connaissances sont ensuite opérationnalisées afin de créer un modèle de classification de régions.

A contrario, l'approche à partir d'exemples se base sur des objets déjà classés pour inférer la fonction c : systèmes à base de règles (induites des exemples), arbres de décision, réseaux de neurones, approches par prototypes, ... Le lecteur intéressé par un descrip-

tif plus exhaustif des algorithmes d'apprentissage à partir d'exemples peut se référer à (Mitchell 1997, Witten & Frank 2005).

Dans les deux cas, la classification se base sur un ensemble d'attributs permettant de caractériser les régions.

3.3.2 Caractérisation des régions

Pour classer les régions, il faut définir un formalisme permettant de les décrire. Initialement, une région est définie par la composition d'un certain nombre de pixels qui forment une composante connexe. Chacun de ces pixels est défini par une position dans l'image (ou une position géolocalisée) et un vecteur de valeurs qui dénote l'intensité de la réponse spectrale du pixel dans différentes longueurs d'onde du spectre. Cette représentation des régions utilise un formalisme multi-relationnel, c'est-à-dire un formalisme où plusieurs types d'entités (régions, pixels) existent et sont reliées entre elles par une relation *un à plusieurs* (une région comporte un ensemble de pixels). Il peut aussi exister une relation *un à plusieurs* entre les régions pour modéliser l'adjacence. Même si des algorithmes de classification capables de traiter ce type de formalisme existent (Dzeroski & Lavrac 2001), ils n'ont (à notre connaissance) jamais été utilisés pour de la classification de régions.

Une autre approche consiste à propositionnaliser (passage d'un formalisme multi-relationnel à un formalisme attribut-valeur) les données afin de n'avoir que des attributs décrivant l'entité région. Deux familles de propositionnalisation existent : la sélection (Krogel & Wrobel 2002, Kira 1992) (par exemple un attribut booléen *existe-t-il un pixel dont l'intensité est inférieure à 0.1 ?*) et l'agrégation (Perlich & Provost 2003, Knobbe et al. 2001) (par exemple la surface qui compte le nombre de pixels de la région). La propositionnalisation peut s'avérer être un exercice difficile (Lachiche 2005) et entraîne généralement une perte d'information.

Dans notre cas, la classification concerne des régions de segmentation d'images de télédétection à très haute résolution. Nous utilisons principalement l'agrégation pour définir les propriétés internes des régions, et la sélection pour définir des propriétés d'adjacence entre les régions. En effet, les caractéristiques propres d'un pixel d'une région ont peu d'intérêt, a priori, pour classer des régions. Nous utilisons aussi des attributs qui ne sont pas issus d'agrégation ou de sélection, mais qui sont spécifiques au domaine. Par exemple, nous construisons l'attribut *elongation* qui permet de mesurer l'élargissement d'une région.

L'ensemble des attributs ainsi utilisés peut se décomposer en trois catégories :

- les attributs spectraux, regroupant des attributs statistiques sur les valeurs spectrales des pixels (la moyenne et l'écart-type sur chacune des bandes de l'image) ;
- les attributs spatiaux, regroupant des attributs caractérisant la forme de la région (élargissement par exemple), sa taille, ... ;
- les attributs contextuels, caractérisant l'environnement de la région (présence d'objets d'une classe particulière à proximité, ...).

3.3.3 Critères d'évaluation

Les critères d'évaluation d'une classification d'image peuvent être assez similaires à ceux d'évaluation d'une segmentation d'image déjà vus précédemment. Pour la classification, il est néanmoins possible de se limiter à quatre critères d'évaluation : la précision, le rappel, la F-mesure et le Kappa.

Ces trois critères sont basés sur une matrice de confusion. Nous rappelons ici comment créer une matrice de confusion K entre une classification et des régions de référence. Pour chaque pixel d'évaluation d'une classe C_i , étant étiqueté par la classe C_j dans la classification, la valeur de la cellule K_{ij} est incrémentée de 1 (les cellules étant initialisées à zéro). Dans une classification, une région (et donc les pixels sous-jacents) peut n'avoir aucune étiquette attribuée. Dans ce cas, les pixels sous-jacents de la région ne seront pas ajoutés dans la matrice de confusion.

Précision

La précision est le nombre de pixels bien étiquetés divisé par le nombre total de pixels étiquetés, c'est-à-dire :

$$EA = \frac{1}{\Omega(\mathcal{C})} \times \sum_{i=1}^{\Omega(\mathcal{C})} \frac{K_{ii}}{\sum_{j=1}^{\Omega(\mathcal{C})} K_{ji}} \quad (3.13)$$

Il peut aussi être intéressant de se focaliser sur la précision pour une classe donnée. On utilise alors les pixels qui ont été étiquetés par la classe donnée. Ce critère est défini par :

$$EA(C_i) = \frac{K_{ii}}{\sum_{j=1}^{\Omega(\mathcal{C})} K_{ji}} \quad (3.14)$$

Rappel

Le rappel est le nombre de pixels bien classés divisé par le nombre de pixels d'exemples, c'est-à-dire :

$$R = \frac{1}{\Omega(\mathcal{C})} \times \sum_{i=1}^{\Omega(\mathcal{C})} \frac{K_{ii}}{\Omega(C_i)} \quad (3.15)$$

avec $\Omega(C_i)$ le nombre de pixels des régions de référence de la classe C_i .

À nouveau, on peut choisir de se focaliser sur le rappel pour une classe donnée. Ce critère est défini par :

$$R(C_i) = \frac{K_{ii}}{\Omega(C_i)} \quad (3.16)$$

F-Mesure

La F-mesure est un critère qui combine le rappel et la précision afin de fournir une évaluation unique. Sa formule est la suivante :

$$\text{F-mesure} = \frac{2 \times R \times EA}{R + EA} \quad [3.17]$$

À nouveau il est possible de l'utiliser pour une classe en particulier :

$$\text{F-mesure}(C_i) = \frac{2 \times R(C_i) \times EA(C_i)}{R(C_i) + EA(C_i)} \quad [3.18]$$

Kappa

Le Kappa est un critère d'évaluation qui permet d'évaluer une classification par rapport à une classification aléatoire. Il est défini par :

$$\kappa = \frac{P_0 - P_e}{1 - P_e} \quad [3.19]$$

avec P_0 la proportion d'accord entre la classification et les exemples et P_e la proportion d'accord aléatoire. Nous utilisons la précision pour P_0 et nous calculons P_e par :

$$P_e = \frac{\sum_{i=1}^{\Omega(\mathcal{C})} K_{i.} K_{.i}}{\left(\sum_{i=1}^{\Omega(\mathcal{C})} \sum_{j=1}^{\Omega(\mathcal{C})} K_{ij} \right)^2} \quad [3.20]$$

La valeur de ce critère peut aller de -1 à +1. Un score proche de zéro indique que la classification équivaut à une classification aléatoire. Plus la valeur est proche de 1, plus le résultat est corrélé aux exemples. Inversement, un score inférieure à zéro indique une corrélation inverse aux exemples.

3.3.4 Méthode de classification par connaissances du domaine

Les systèmes à base de connaissances ont prouvé leur efficacité pour l'identification d'objets complexes (Liu et al. 1994, Chetty & Deshpande 1995), l'analyse d'image (Matsuyama & Hwang 1990, Perner 1994) ou l'interprétation d'images (Ogiela & Tadeusiewicz 2008). Par exemple, les systèmes SIGMA (Matsuyama & Hwang 1990) et Schema (Draper et al. 1989) effectuent des analyses d'images aériennes en utilisant plusieurs descripteurs de régions. Ces systèmes permettent d'accéder à un niveau sémantique élevé. Néanmoins, comme le soulignent Crevier & Lepage (1997), ces systèmes sont très dépendants du domaine car ils intègrent des connaissances a priori sur l'image à analyser. Leurs inconvénients sont que les connaissances du domaine ne

sont pas clairement séparées de la partie procédurale et que la base de connaissances est difficile à produire.

C'est pourquoi des travaux récents ont proposé d'utiliser des systèmes d'ontologies pour décrire plus clairement les connaissances du domaine. Zlatoff et al. (2004) utilisent les relations spatiales entre les concepts pour fusionner les régions et identifier les objets de classes. L'utilisation quasi-exclusive des relations spatiales se prête difficilement à notre domaine d'application (imagerie satellitaire à très haute résolution).

Panagi et al. (2006) proposent un algorithme génétique d'analyse sémantique basé sur une ontologie. Des attributs de bas niveau sont extraits de l'image et utilisés pour correspondre avec l'ontologie. Un ensemble d'hypothèses (région, liste des classes possibles et degré de confiance) sont testées avec un algorithme génétique pour déterminer l'interprétation acceptable de l'image. Seules les relations spatiales sont utilisées dans ce système.

Inspiré par ces travaux, nous avons proposé un algorithme original qui calcule un score de correspondance entre une région à identifier et des classes contenues dans la base de connaissances. Le résultat final est, pour chaque région, une liste d'étiquettes de classes possibles et un score de confiance pour chaque attribution d'étiquette.

Cette méthode utilise la fonction de score présentée précédemment (équation 3.3) et l'algorithme de parcours (algorithme 4).

Remise en cause de l'étape de segmentation par la classification

La qualité de la segmentation est un point crucial et la qualité finale de la classification en dépend directement. C'est pourquoi nous proposons d'utiliser une classification par connaissances du domaine afin d'améliorer la qualité de la segmentation. Il s'agit de permettre à l'étape de classification de remettre en cause la segmentation.

La méthode se base sur celle présentée à la section 3.2.1 à la différence que nous considérons que nous ne disposons pas d'exemples. L'idée est d'utiliser la base de connaissances pour évaluer la qualité de la classification et la segmentation plutôt que des exemples. Comme nous ne pouvons pas utiliser les mesures classiques d'évaluation qui utilisent des exemples, nous introduisons une autre mesure de qualité de la classification.

Nous allons utiliser comme fonction d'évaluation le pourcentage de la surface de l'image qui est identifiée par la classification. Soit \mathcal{X} les régions d'une segmentation et \mathcal{X}_o les régions identifiées par la classification ($\mathcal{X}_o \subseteq \mathcal{X}$). Le pourcentage de la surface de l'image reconnue par la classification est défini par :

$$\text{Reco}(\mathcal{X}, \mathcal{X}_o) = \frac{\sum_{R \in \mathcal{X}_o} \text{Aire}(R)}{\sum_{R \in \mathcal{X}} \text{Aire}(R)} \quad (3.21)$$

avec $\text{Aire}(R)$ étant la surface de la région R . La surface des régions identifiées a été préférée à leur nombre pour évaluer le résultat. En effet, une segmentation peut produire de nombreuses petites régions qui n'ont pas de signification sémantique et qui ne sont donc pas

reconnues par la méthode de classification. Ces petites régions peuvent perturber un calcul basé sur le nombre de régions. Une grande partie de l'image peut être reconnue si de grandes régions (par exemple l'étendue végétale) sont reconnues par la classification. Ce résultat peut être faussé si de petites régions sans sémantique sont présentes dans le reste de l'image segmentée.

Le critère basé sur la surface des régions permet de quantifier la qualité de la segmentation par rapport à la classification basée sur les connaissances du domaine. L'augmentation de ce critère signifie que les régions construites par la segmentation correspondent de mieux en mieux à la description des objets géographiques présents dans la base de connaissances. En maximisant ce critère nous nous assurons de construire une segmentation en accord avec les connaissances du domaine fournies par l'expert.

Il est à noter que pour que cette méthode fonctionne, il est essentiel de mettre le seuil minScore de la méthode de classification basée sur les connaissances du domaine à 1. Ce paramétrage oblige la méthode à ne reconnaître que les régions qui y correspondent parfaitement. Réduire cette contrainte aurait pour effet de permettre la classification de plus de régions, mais réduirait la confiance que l'on peut exprimer sur une telle classification.

3.4 Connaissances et détection

La détection consiste à rechercher des objets d'intérêt dans une image. On appelle détecteur un algorithme capable, à partir d'une image, d'associer un ensemble d'instances de la classe qu'il recherche. Dans cette section, nous présentons l'intérêt que peut avoir une approche utilisant un ensemble de détecteurs par rapport à l'approche plus classique consistant en une segmentation globale suivie d'une classification des régions, présentée précédemment.

La littérature contient de nombreuses méthodes (Jin & Davis 2005, Lefèvre et al. 2006, Peteri et al. 2003, Yager & Sowmya 2004, Zhao et al. 2002) pour rechercher les objets de certaines classes d'objets particulières (route, immeubles, etc.). Ces méthodes sont très spécifiques et intègrent généralement les connaissances implicites de leurs concepteurs sur les classes d'objets recherchées. Ces connaissances sont ensuite difficiles à extraire si l'on souhaite les modifier ou les améliorer.

Nous proposons une approche plus générique, permettant d'extraire des détecteurs pour chaque classe d'objets à partir d'une base de connaissances. Les connaissances et l'algorithmique sont ainsi clairement séparées. L'objectif est de permettre à l'expert du domaine d'exprimer facilement ses connaissances sans être spécialiste en traitement d'image ou en extraction de connaissances.

Enfin, nous abordons le problème de la communication entre plusieurs détecteurs. Les détecteurs peuvent en effet être vus comme des agents capables de créer de l'information. Il apparaît naturel que cette information puisse être utilisée par d'autres détecteurs. Nous verrons notamment le cas des détecteurs d'objets composites (par exemple un îlot résidentiel est composé de maisons, routes et végétation).

3.4.1 Interprétation par ensemble de détecteurs

Un détecteur d_i , spécialisé pour la classe C_i , peut être formalisé comme une fonction qui à un pixel p_j associe une valeur binaire issue de $\mathbb{B} = \{\text{vrai}, \text{faux}\}$ signifiant l'appartenance ou non du pixel p_j à la classe C_i :

$$d_i(p_j) = \begin{cases} \text{vrai} & \text{si } p_j \text{ appartient à } C_i \\ \text{faux} & \text{sinon} \end{cases} \quad [3.22]$$

Par extension, en appliquant la fonction à tous les pixels de l'image, il est possible d'obtenir la liste des composantes connexes, les régions, appartenant à C_i . On note ainsi \mathcal{X}_i l'ensemble des objets que le détecteur d_i propose comme étant de la classe C_i .

Chaque détecteur permet la détection d'une seule classe. Afin de classifier l'image, il faut utiliser un détecteur par classe d'intérêt dans l'image. On obtient un ensemble de détecteurs $\mathcal{D} = \{d_i | C_i \in \mathcal{C}\}$ (avec \mathcal{C} l'ensemble des classes d'objets).

Il est à noter que certains pixels de l'image peuvent n'appartenir à aucune classe. Plus formellement, il peut exister un pixel p_j tel que $\forall i | C_i \in \mathcal{C}, d_i(p_j) = \text{faux}$. De même, un pixel peut être considéré comme appartenant à plusieurs classes. C'est à dire, qu'il peut exister un pixel p_j tel que $d_a(p_j) = \text{vrai}$ et $d_b(p_j) = \text{vrai}$ avec $a \neq b$ et $C_a, C_b \in \mathcal{C}$.

Le fait que des pixels puissent n'être associés à aucune classe a déjà été rencontré pour la classification d'objets par connaissances du domaine à la section 3.3.4. Cela est naturel car les images, notamment de télédétection, peuvent contenir des objets non recherchés par l'utilisateur ou des pixels mixtes (pixels appartenant partiellement à plusieurs classes) ou de bruit.

De même, nous pouvons avoir deux objets x_a et x_b tels que $x_a \in \mathcal{X}_i$ et $x_b \in \mathcal{X}_j$ avec $x_a \neq x_b$ et $x_a \cap x_b \neq \emptyset$. Ces deux objets de classes différentes se chevauchent donc partiellement. Cet effet peut provenir d'une erreur d'un des algorithmes de détection ou d'une difficulté intrinsèque à l'image pour fixer une frontière entre deux objets. Cela peut arriver dans le cas de pixels mixtes. Comme les différents détecteurs ne tiennent pas compte les uns des autres, ils ne cherchent pas à résoudre ce type d'incohérences.

Comparaison avec l'approche par segmentation suivie d'une classification - La différence principale entre l'approche par ensemble de détecteurs et celle de la segmentation suivie d'une classification des régions est le type de séparation effectué dans les traitements. L'approche segmentation puis classification opère une séparation séquentielle, c'est-à-dire qu'elle divise en deux le traitement, d'un côté la segmentation et de l'autre la classification. De son côté, l'approche par ensemble de détecteurs propose une séparation parallèle, i.e. le traitement de chaque classe est séparé.

L'avantage de l'approche par détecteurs est justement la possibilité d'appliquer un traitement spécifique à chaque classe. Dans l'approche par segmentation puis classification, toutes les classes sont traitées par le même algorithme. C'est pourquoi, il est difficile de prendre en compte les spécificités de chaque classe, ou bien les connaissances particulières disponibles.

L'approche par ensemble de détecteurs a aussi un avantage calculatoire. En effet, le traitement de chaque classe se faisant à part, le processus de détection est facilement parallélisable. Il est de plus possible d'utiliser des détecteurs dont la complexité algorithmique est corrélée à la complexité de détection de la classe. Par exemple, la végétation est une classe facilement détectable et ne demande pas un détecteur algorithmiquement très complexe.

Enfin, cette approche a un avantage conceptuel. Il est en effet plus simple de se focaliser sur la création d'un détecteur pour une classe que de créer un algorithme capable de découvrir toutes les classes d'intérêt.

Le seul inconvénient de l'approche par détecteurs, est que deux régions de classes différentes peuvent se chevaucher partiellement. Dans le cas de la segmentation suivie d'une classification, par définition, cela ne peut pas se produire, chaque pixel n'appartenant qu'à une région après la segmentation. Cet inconvénient est néanmoins à relativiser. En effet, si l'étape de segmentation n'affecte qu'une seule région à chaque pixel, ce n'est pas forcément la région la plus appropriée qui est choisie. Comme nous l'avons vu précédemment, une image peut être sous-segmentée, c'est-à-dire qu'une région peut couvrir deux objets d'intérêt de classes différentes.

Nous pouvons donc conclure que malgré cet inconvénient, l'approche par détecteurs dispose de nombreux atouts qui doivent permettre d'obtenir de meilleures performances qu'une approche par segmentation puis classification. Nous allons à présent présenter une application de cette approche.

3.4.2 Extraction de détecteurs spécifiques à partir de la base de connaissances

De nombreuses méthodes spécifiques existent pour détecter des objets particuliers dans une image, tels que la route (Peteri et al. 2003, Yager & Sowmya 2004, Zhao et al. 2002) ou le bâti (Jin & Davis 2005, Lefèvre et al. 2006). Ces méthodes intègrent les connaissances implicites qu'ont leurs concepteurs sur les types d'objets recherchés. Cette connaissance n'a pas été modélisée et se retrouve imbriquée dans l'algorithmique des détecteurs ainsi créés. Il en résulte que ces connaissances sont ensuite difficiles à retrouver si l'on souhaite améliorer ou adapter l'algorithme.

D'un autre côté, afin de modéliser ces connaissances implicites, l'utilisation d'ontologie ou de bases de connaissances devient de plus en plus courante, par exemple dans les systèmes d'informations géographiques (Fonseca et al. 2002) ou pour l'analyse d'image (Bittner & Winter 1999). L'objectif est d'obtenir une spécification abstraite, une vue simplifiée du monde représentée dans un but précis (Gruber 1995). Avec la base de connaissances définie précédemment (section 3.1), nous disposons d'un ensemble de classes d'intérêt, leurs caractéristiques et des relations entre elles. La plupart des méthodes (Durand et al. 2007, Maillot 2005, Mezaris et al. 2004, Panagi et al. 2006) formalisent les classes pouvant être présentes dans une image puis proposent une analyse sémantique de celle-ci en cherchant à identifier les objets de ces classes dans l'image.

C'est pourquoi, nous proposons une méthode d'extraction de détecteurs pour des objets spécifiques à partir d'une base de connaissances. L'originalité de cette approche est de séparer la partie connaissance de la partie algorithmique alors que dans les méthodes existantes, la connaissance est mêlée au processus d'identification. La méthode que nous proposons s'inspire des travaux menés dans les classifications de régions à l'aide de connaissances (section 3.3.4). Les connaissances sont vues comme des contraintes qui permettent d'affirmer que des parties de l'image ne peuvent appartenir à une classe donnée. En effet, une zone de l'image qui ne valide pas une contrainte ne peut appartenir à la classe du détecteur. Une fois toutes les contraintes appliquées, il ne reste plus que les zones pouvant appartenir à cette classe.

Structure générale des détecteurs - Les détecteurs que nous allons créer ont une structure de base identique. Ils commencent par considérer que toute l'image correspond à la classe qu'ils recherchent. Ensuite, une succession de filtres sont appliqués afin de retirer les zones de l'image qui ne correspondent pas à la classe recherchée. Une fois ces filtres appliqués, il ne reste plus que les zones de l'image pouvant appartenir à la classe recherchée.

Les filtres sont extraits de la base de connaissances, en prenant les connaissances comme des contraintes qui doivent être vérifiées par les objets de la classe recherchée. La problématique consiste à traduire les différentes connaissances en des contraintes à respecter et enfin en des filtres opérationnels.

Nous avons vu à la section 3.1 que les connaissances sont souvent exprimées sous la forme d'intervalles sur des attributs. Dans ce cas il est fort aisément de développer un filtre associé. En effet, il suffit de vérifier si la valeur pour cet attribut est bien dans l'intervalle permis. Si ce n'est pas le cas, il faut supprimer la partie de l'image. Nous allons étudier en détail les différents cas.

Description des filtres pouvant composer un détecteur - Pour le niveau spectral, deux méthodes de filtrage sont possibles selon les connaissances disponibles. La première méthode, détaillée à l'algorithme 5, utilise les intervalles des valeurs spectrales admises pour la classe recherchée. Pour chaque pixel, si ses valeurs spectrales ne sont pas dans l'intervalle défini, ce pixel est supprimé de la liste des pixels détectés. La seconde méthode est utilisable si des exemples sont disponibles et que l'expert a indiqué qu'il est possible d'en extraire des connaissances spectrales. Dans ce cas, il est possible d'effectuer une classification supervisée floue basée pixel en utilisant les attributs spectraux. Le détail du processus de filtrage est donné à l'algorithme 6. La fonction `classificationFloue(p, e_1, e_2)` renvoie la probabilité d'appartenance du pixel p à la classe représentée par les exemples e_1 , les exemples e_2 étant utilisés comme exemples négatifs. Le paramètre seuil permet de définir à partir de quelle probabilité d'appartenance à la classe recherchée on conserve le pixel. Nous prendrons ici comme valeur arbitraire 0,4, c'est-à-dire que même si un pixel semble appartenir plus à la classe représentée par les exemples e_2 , il peut quand même être conservé dans celle représentée par e_1 . La raison de ce choix est d'éviter de suppri-

mer des pixels alors qu'ils appartiennent à la classe, la classification basée pixels pouvant commettre des erreurs. Dans les deux cas de filtrage spectral, nous appliquons une fermeture binaire, c'est-à-dire que nous remettons dans la liste des pixels détectés toutes les zones connexes non détectées qui sont plus petites qu'une forme prédefinie, ici un carré de 3×3 pixels, ceci afin de gérer le bruit poivre et sel ou les petits objets sans intérêts présents sur l'image.

Algorithme 5: *Filtrage spectral par contraintes*

soit C_c la classe recherchée
 soit d_c le détecteur de la classe C_c
 soit \mathcal{P} l'ensemble des pixels de l'image à interpréter
 soit \mathcal{A}_s l'ensemble des attributs spectraux avec un intervalle spécifié dans la base de connaissances pour la classe C_c
pour tous les $p \in \mathcal{P}$ faire

$$d_c(p) := \begin{cases} \text{vrai} & \text{si } \forall a \in \mathcal{A}_s, \text{valeur}(p, a) \in [\min(C_c, a), \max(C_c, a)] \\ \text{faux} & \text{sinon} \end{cases}$$

Algorithme 6: *Filtrage spectral par classification floue supervisée*

soit C_c la classe recherchée
 soit d_c le détecteur de la classe C_c
 soit \mathcal{P} l'ensemble des pixels de l'image à interpréter
 soit \mathcal{C}_c l'ensemble des classes dont les exemples doivent être utilisés pour détecter C_c
pour tous les $p \in \mathcal{P}$ faire
pour tous les $C_a \in \mathcal{C}_c \setminus C_c$ faire

$$\begin{cases} \text{si } \text{classificationFloue}(p, \text{exemples}(C_c), \text{exemples}(C_a)) > \text{seuil} \text{ alors} \\ \quad d_c(p) := \text{faux} \end{cases}$$

Pour la forme, nous ne pouvons utiliser directement les intervalles. En effet, nous avons à notre disposition une image binaire (pixels détectés comme étant de la classe recherchée et ceux non détectés). Nous ne pouvons utiliser directement les composantes connexes présentes dans l'image. Différentes composantes connexes représentant des objets peuvent être reliées par des pixels détectés par erreur comme appartenant à la classe recherchée. Dans ce type de situations, utiliser les attributs des composantes connexes ne permet pas de détecter ces objets. Nous proposons de construire des filtres sous forme de traitements morphologiques sur l'image binaire.

Enfin, il reste les attributs de régions. Pour les traiter, une segmentation de l'image est effectuée, chaque composante connexe de l'image de détection correspondant à une région. Des attributs de régions peuvent alors être calculés. Si la valeur d'un des attributs

ne correspond pas à l'intervalle autorisé pour la classe donnée, la composante connexe est retirée de la détection.

À la fin de l'application de ces différents filtres, l'image binaire ne contient plus que les zones considérées par le détecteur comme étant des objets de la classe recherchée.

Unification des résultats de détecteurs - Selon l'utilisation faite des résultats, il peut être nécessaire de faire une unification des différentes cartes de détection afin de fournir une classification de l'image. Pour cela, la méthode consiste à attribuer une pondération à chaque résultat pour déterminer la priorité de chacun des résultats sur les autres. Le problème est ici simplifié car chaque résultat à fusionner ne comporte qu'une seule classe. Chaque pixel qui est détecté par au moins un détecteur, est associé à la classe dont le détecteur a la plus grande priorité. L'ordre de priorité des différents détecteurs est donné par l'expert.

3.5 Connaissances et clustering collaboratif

Dans cette section, nous présentons comment nous avons adapté la méthode de clustering collaboratif vue section 2.1.3 pour effectuer l'étiquetage de clusters de régions en vue d'une meilleure identification des objets dans l'image.

3.5.1 Problématique

Rappelons que la méthode d'identification vue en début de chapitre (section 3.1) permet d'affecter un concept à des régions à l'issue d'une étape de segmentation. Cependant, la méthode ne permet d'identifier que peu de régions avec certitude. Le paramètre *minScore* qui permet d'augmenter le nombre d'objets reconnus en réduisant le respect des contraintes imposées par les concepts lève partiellement ce verrou. Cependant, quand ce seuil baisse, de nombreuses erreurs d'identification apparaissent. Nous proposons ici une autre approche consistant à utiliser la méthodes de clustering collaboratif sur les régions issues d'une segmentation. Chaque cluster se verra ensuite affecté une étiquette en fonction des objets lui appartenant. Cette approche a deux intérêts : le premier est de proposer un étiquetage automatique des clusters, et la seconde est de permettre une augmentation significative du nombre d'objets identifiés.

3.5.2 Étiquetage des clusters

Le processus de clustering collaboratif offre à l'expert des mécanismes efficaces de regroupement d'objets similaires sous la forme de clusters. Cependant, il ne permet pas d'affecter directement une sémantique aux clusters, c'est-à-dire une correspondance directe avec un concept ou une classe du monde réel. Pour pouvoir affecter cette sémantique, l'expert doit utiliser sa connaissance pour faire correspondre un groupe d'objets à un concept. Cette tâche, dite d'étiquetage des clusters, n'est pas une tâche facile

pour l'expert. De plus, elle est généralement fastidieuse et peu gratifiante. La méthode d'étiquetage d'objets géographiques vues précédemment va nous permettre d'automatiser cette tâche. L'objectif est d'identifier pour chacun des clusters découverts par la méthode collaborative, si celui-ci correspond à un concept d'objets géographiques. Pour cela, les objets appartenant aux clusters, c'est-à-dire les régions, vont être utilisés en entrée de la méthode d'identification. Puis, dans chaque cluster, un vote à la majorité prenant en compte le score d'appariement (voir équation (3.3)) est effectué pour identifier le concept majoritaire.

À l'issue de la segmentation, les régions sont caractérisées par des attributs spectraux et de forme. Le clustering collaboratif est alors appliqué sur ces régions caractérisées, qui sont ensuite identifiées en utilisant la base de connaissances.

Soit \mathcal{X} l'ensemble des régions d'une segmentation et \mathcal{X}_o l'ensemble des régions identifiées par la méthode utilisant les connaissances du domaine ($\mathcal{X}_o \subseteq \mathcal{X}$). L'objectif est d'utiliser ces régions identifiées pour étiqueter les clusters d'un résultat de clustering $C = \{C_1, \dots, C_K\}$. Le système d'étiquetage des clusters prend en compte le score d'appariement (voir équation (3.3)) affecté par la méthode d'identification aux régions en fonction du concept identifié. Le cluster est alors étiqueté par l'étiquette du concept dont la somme des scores est la plus importante parmi les régions présentes dans ce cluster :

$$\text{étiquetage}(C) = \arg \max_{\mathcal{C}} \sum_{R \in C} \text{Score}(R, \mathcal{C}) \quad | \quad \mathcal{C} \neq \text{inconnu} \quad 3.23$$

À l'issue de cette étape d'étiquetage, chaque cluster C du résultat aura soit un concept \mathcal{C} qui lui aura été affecté, soit le concept `inconnu` si aucune région du cluster n'a été identifiée. En effet, en fonction du seuil sélectionné, certaines régions peuvent rester inconnues. Il est conseillé de choisir un seuil relativement élevé ($\geq 0,90$) pour éviter de trop nombreuses erreurs d'identification. Une étape de ré-étiquetage des régions est ensuite engagée. Lors de cette étape, dans chaque cluster ayant été étiqueté par un concept, les régions n'ayant pas encore de concept se voient affecter le concept assigné à leur cluster d'appartenance. Ce mécanisme permet d'augmenter considérablement le nombre de régions reconnues. Ainsi, de nombreuses régions vont passer du concept `inconnu` au concept associé à leur cluster lors de cette étape. Il est également envisageable de ré-étiqueter les régions présentant un concept différent que le concept majoritaire de leur cluster d'appartenance. Ce choix est laissé à l'utilisateur et ces régions (peu nombreuses en pratique) sont généralement traitées au cas par cas.

La figure 3.6 présente une illustration dans l'espace des données de ce mécanisme d'étiquetage des clusters. Chaque point sur ces figures représente une région, décrite ici par les attributs correspondant à la moyenne des valeurs sur la Bande 1 des pixels composant la région et à la taille de la région. La figure 3.6 (a) présente le résultat de l'identification utilisant les connaissances. Les bornes des concepts Pavillon et Route sont également illustrées. La figure 3.6 (b) présente le résultat de clustering obtenu en considérant les régions comme les objets à classer. La figure 3.6 (c) présente la combinaison de ces deux informations (clustering + connaissances). Enfin, la figure 3.6 (d) présente le résultat final,

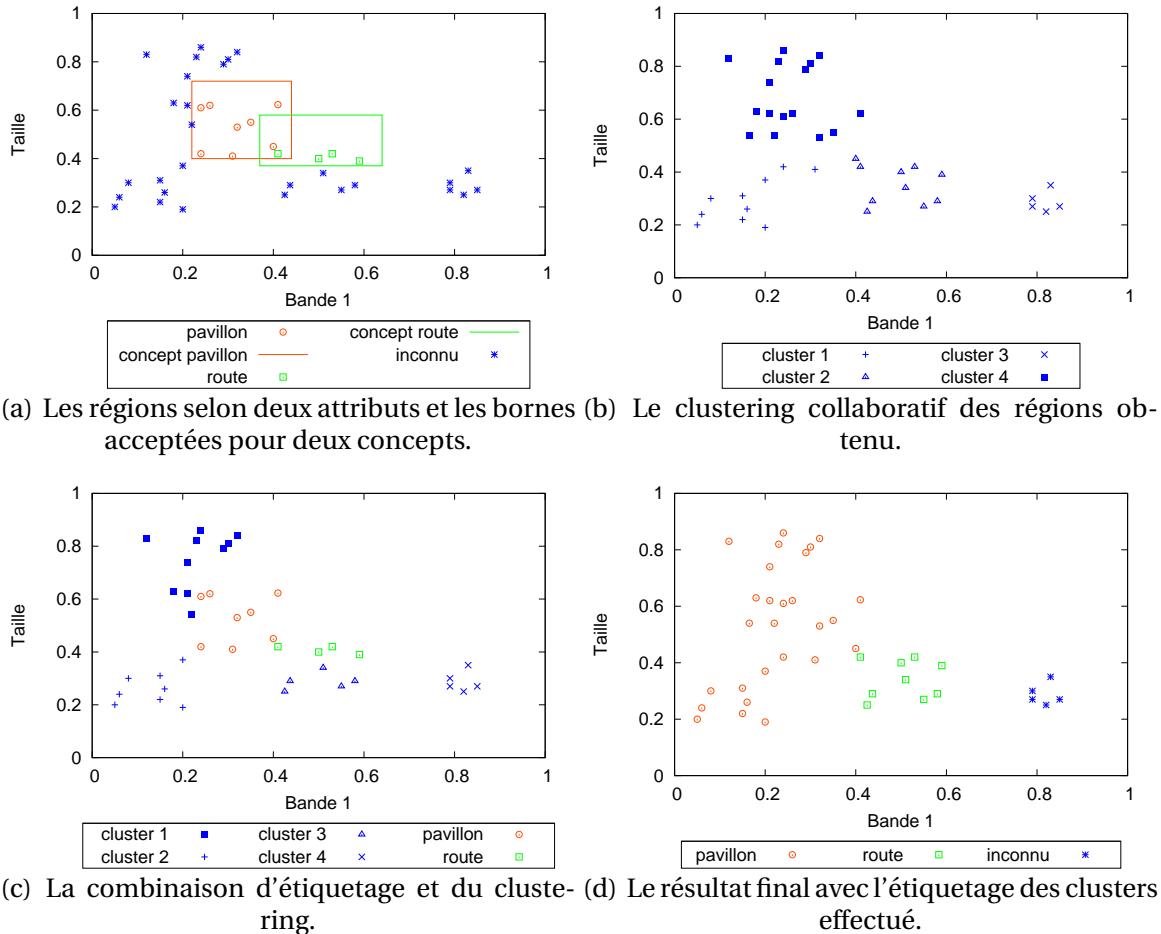


Figure 3.6
Illustration du processus d'étiquetage des clusters dans l'espace des données.

une fois la règle d'étiquetage des clusters (voir équation (3.23)) appliquée et les régions rétiquetées avec le concept majoritaire de leur cluster.

3.6 Contributions et valorisation

Ce chapitre a présenté une partie de nos contributions dans le domaine de l'utilisation de connaissances expertes dans un processus d'extraction d'informations à partir d'images de télédétection. Ces contributions ont été validées par plusieurs publications.

Nous avons proposé un mécanisme d'appariement de régions à une ontologie d'objets géographiques, représentant des connaissances expertes. Une première définition de cette méthode a été présentée dans (Durand et al. 2007) puis étendue dans (Forestier et al. 2012). Un article présentant l'utilisation d'un ensemble de détecteurs dans le cas de

la classification d'images de télédétection a été publié dans (Derivaux et al. 2008). Enfin, la méthode originale de segmentation supervisée, *probashed*, a été définie dans (Derivaux et al. 2010).

Concernant les travaux sur la méthode SAMARAH et son extension pour permettre l'utilisation de connaissances expertes, ils ont été validés par une première publication (Forestier et al. 2008b) regroupant l'intégration a posteriori des connaissances dans la méthode collaborative. Une seconde publication (Forestier, Wemmert & Gançarski 2010a) présente l'intégration de connaissances en clustering sous la forme de critères de pureté. Enfin, un article en revue (Forestier, Gançarski & Wemmert 2010) décrit en détail les différentes formes d'intégration de connaissances possibles dans la méthode collaborative et les compare.

4

Le genre humain a toujours été en progrès et continuera toujours de l'être à l'avenir : ce qui ouvre une perspective à perte de vue dans le temps.

Emmanuel Kant (1781)

Conclusion et perspectives

4.1 Conclusion	91
4.2 Perspectives	92
4.2.1 Application au domaine des images microscopiques	93
4.2.2 Vers un cadre générique de représentation de la connaissance	94

4.1 Conclusion

Ce mémoire présente mes principales contributions théoriques ou pratiques dans le domaine de la collaboration de méthodes de clustering et d'intégration de connaissances dans un processus d'extraction d'informations à partir d'images.

D'un point de vue plus théorique, j'ai mené dans un premier temps, des études sur les méthodes de clustering et les mécanismes de collaboration permettant de tirer parti de leurs avantages propres et surtout de leur complémentarité. Dans ce cadre, j'ai tout d'abord proposé l'approche générique de collaboration de méthodes de clustering SAMARAH basée sur la résolution de conflits entre les résultats de clustering. Les résultats proposés convergent avant d'être finalement fusionnés par un algorithme de vote. Une méthode originale de vote multi-points de vue pour la fusion de résultats de clustering a aussi été définie, afin de pouvoir prendre en compte différents résultats ne proposant pas nécessairement le même nombre de clusters. De plus, une étude théorique poussée sur le choix des conflits à résoudre a été menée et a abouti à la définition d'un schéma de collaboration basé sur le paradigme de l'optimisation stochastique permettant d'obtenir de meilleurs résultats qu'avec la méthode de collaboration précédente.

Nos travaux sur le clustering collaboratif se sont ensuite portés sur les aspects théoriques, mais aussi pratiques, de l'intégration de connaissances dans le processus de clustering. Ces recherches ont permis de faire évoluer la méthode collaborative existante en lui permettant d'être guidée par des connaissances. Une des originalités fortes de cette proposition est que ces connaissances peuvent être données sous différentes formes : exemples étiquetés, contraintes sur les classes ou sur des relations entre exemples, description "formelle" des objets d'intérêt. De nombreux développements ont été réalisés au

sein de la méthode pour permettre cette intégration.

Nous avons ensuite proposé une extension du cadre collaboratif aux méthodes semi-supervisées et non supervisées. Dans ce cadre, nous avons proposé la méthode *Semi-supervised learning enhanced by multiple clusterings* (SLEM). Cette méthode améliore la classification supervisée basée sur un ensemble d'exemples étiquetés en produisant tout d'abord un clustering sur l'ensemble de données étiquetées ou non. Les objets similaires étant regroupés dans les mêmes clusters, il est ensuite possible d'affecter à chacun de ces cluster la classe correspondant aux objets exemples qu'il contient.

D'un point de vue plus applicatif, nous nous sommes intéressés à l'extraction d'informations à partir d'images, et notamment les images de télédétection. Dans ce domaine, l'arrivée récente de nouveaux capteurs à très haute résolution spatiale a mis à mal les méthodes existantes. Rapidement, les méthodes dites "orientées objet", proposées pour résoudre cette difficulté, se sont heurtées au problème du fossé sémantique existant entre les segments extraits et les objets d'intérêt recherchés. C'est pourquoi nous avons défini une base de connaissances géographiques et un mécanisme d'appariement permettant de faire le lien entre les segments extraits de l'image et les concepts du domaine (objets urbains par exemple) décrits entre autres, par des caractéristiques radiométriques et géométriques.

Pour lever les verrous scientifiques liés à la dépendance forte de la méthode de classification à la segmentation utilisée, nous avons proposé une première solution consistant à utiliser les connaissances directement pendant l'étape de segmentation pour construire des objets plus facilement étiquetables par l'expert ou de manière automatique en utilisant cette base de connaissances. Une seconde approche a aussi été définie, basée sur le clustering collaboratif afin de créer des clusters de régions similaires. Ces clusters de régions similaires sont ensuite étiquetés en observant la répartition des concepts présents dans ces clusters. Loin de pouvoir fournir un système totalement automatique, la solution proposée permet d'aider l'expert dans le processus d'interprétation en lui fournissant des outils lui facilitant l'interprétation des images et la découverte de nouveaux concepts.

Contrairement aux approches séquentielles classiques de segmentation puis classification d'objets qui effectuent une séparation séquentielle entre la segmentation puis la classification, nous avons proposé une approche par détecteur qui effectue une extraction indépendante et parallèle de chaque classe d'objets d'intérêt. Cela permet de définir des traitements spécifiques, et donc potentiellement plus efficaces, pour chaque classe sans avoir à se conformer au schéma de segmentation puis classification.

4.2 Perspectives

Les méthodes présentées ci-dessus ont été validées dans le domaine de la classification d'images de télédétection et ont donné de bons résultats. Néanmoins, elles souffrent de nombreuses limitations théoriques et pratiques, et aussi liées au domaine d'application. De fait, elles nécessitent de nouvelles études à court et plus long terme.

Ainsi, à plus court terme, il nous semble très important de valider nos méthodes dans un autre cadre applicatif que celui de la télédétection. En effet, la plupart des propositions faites sont directement applicables à l'extraction d'informations à partir d'images, quelque soit le contexte. Notre première perspective est de valider nos approches dans le domaine de l'imagerie cellulaire microscopique qui nous semble avoir de nombreux points communs avec la télédétection : la détection automatique d'objets d'intérêt dans l'image avec la possibilité de formaliser la connaissance experte sur ces objets, leur caractérisation par des attributs colorimétriques ou de forme ainsi que leur répartition dans l'image.

La caractérisation des relations spatiales entre les objets d'intérêt et la construction d'objets composites dans le domaine de l'extraction d'informations à partir d'images est une seconde perspective ouverte par nos travaux. En effet, peu de méthodes actuelles utilisent efficacement les informations sur les relations spatiales lors de la détection et la construction des objets d'intérêt. Cependant, cette connaissance est souvent primordiale pour permettre d'identifier certaines classes d'objets lorsque les informations de forme ou de radiométrie sont insuffisantes. De plus, nous pensons que ces informations spatiales devraient permettre de construire des objets composites via la description de leur composition et de l'organisation interne des éléments les composant.

L'évolution technologique des appareils d'acquisition d'images et l'augmentation des capacités de stockage des données permettent actuellement de construire des séquences temporelles d'images de plus en plus longues avec des fréquences d'acquisition de plus en plus élevées, qui amènent de nouvelles problématiques liées aux applications telles que la détection de changements ou la classification d'évolutions. L'étude des différents types de relation temporelle entre les objets est un enjeu très important afin de pouvoir répondre à ces applications.

Ainsi, à plus long terme, il est à notre avis primordial de réfléchir à une meilleure formalisation de la connaissance utilisée afin de pouvoir intégrer toutes ces nouvelles connaissances mais surtout être capable de les utiliser et en inférer de nouvelles de manière semi-automatique. L'idée est de parvenir à une description, la plus complète possible, à la fois des connaissances du domaine, mais aussi des processus d'extraction et des données disponibles pour réaliser les objectifs de l'utilisateur. Il s'agira aussi d'être capable d'inférer de nouvelles connaissances automatiquement, afin d'enrichir la base de connaissances.

4.2.1 Application au domaine des images microscopiques

L'ensemble des méthodes proposées ont toutes été validées dans le domaine de la classification automatique d'images de télédétection ou plus largement dans le domaine de l'observation de la Terre. Cependant, elles ont toutes été définies afin de pouvoir s'adapter à tout type de problème d'extraction d'informations à partir d'images.

Récemment, un projet Recherche et Développement dont j'ai eu la responsabilité scientifique a été initié avec la société Roche GmbH, portant sur l'analyse automatique d'images microscopiques représentant des coupes de tissu animal et les cellules le com-

posant. Ce projet a pour objectif de développer une méthode d'interprétation automatique d'images microscopiques cellulaires afin de valider l'efficacité thérapeutique d'un traitement de chimiothérapie. Cette nouvelle collaboration est l'occasion d'évaluer la robustesse des méthodes développées dans le cadre de la télédétection sur un autre domaine de recherche pour lequel les problèmes sont fortement similaires (segmentation et classification d'objets dans des images à très haute résolution spatiale).

Les premiers résultats obtenus montrent la pertinence d'utiliser nos méthodes dans ce cadre là. Cependant, ceci a aussi généré plusieurs nouvelles problématiques qu'il me semble intéressant d'étudier dans la suite de mes travaux.

La première concerne la modélisation et l'utilisation de relations spatiales entre les objets à extraire (ici des cellules). En effet, il est important pour le médecin de comprendre et de caractériser l'agencement spatial des différents types de cellules détectées. De plus, nous proposons d'utiliser ces relations spatiales afin de construire des objets d'intérêts de niveau supérieur ou objets composites (tumeur, vaisseau, etc.).

La seconde porte sur la segmentation des images qui permet de construire les objets primitifs (cellules). En effet, il subsiste de nombreuses erreurs, les images comportant beaucoup de bruit. Il nous semblerait pertinent d'étudier une nouvelle méthode de segmentation permettant la collaboration entre les étapes de segmentation et de classification, sujet qui reste ouvert et peu étudié. En effet, nos résultats montrent la pertinence de la remise en cause de la segmentation afin d'obtenir une meilleure classification. Celle-ci est cependant sommaire et ne permet de modifier que les paramètres de la segmentation. Une idée serait de conserver les objets (segments) bien construits et identifiés et de ne resegmenter que le reste de l'image. Ainsi, nous pourrions avoir une méthode de segmentation adaptative en fonction du contexte.

4.2.2 Vers un cadre générique de représentation de la connaissance

À plus long terme, nous proposons de travailler sur la conception d'une base de connaissances modélisant l'ensemble des connaissances de l'extraction d'informations à partir de données complexes telles que des images (médicales, de télédétection ou autres).

Comme indiqué dans le préambule de ce manuscrit, ces connaissances peuvent être caractérisées en fonction de ce sur quoi elles portent :

- le savoir-faire, qui correspond à la capacité de résoudre selon une méthode propre tout ou partie du problème ;
- les connaissances du domaine, qui correspondent à l'ensemble des informations générales sur le domaine dont relève le problème ;
- le contexte du problème, qui correspond aux données du problème ainsi qu'à des informations particulières sur ces données et sur le problème lui-même.

Dans les travaux que nous avons menés jusqu'à présent, nous nous sommes beaucoup focalisés sur l'intégration des connaissances du domaine dans le processus d'extraction d'informations. Nous nous sommes intéressés à modéliser les connaissances sur

les objets d'intérêts à extraire des images (décris par différents attributs spectraux et géométriques) et avons proposé plusieurs mécanismes permettant de les utiliser afin de guider l'extraction de connaissances. Ce domaine est actuellement très actif. Cependant les approches proposées n'utilisent que des informations spectrales ou géométriques sur les objets d'intérêt à extraire. Or, dans certains cas, ces informations sont insuffisantes pour différencier certaines classes d'objets. Il serait alors intéressant d'être capable de formaliser et d'utiliser d'autres types de connaissances. Parmi celles qui nous semblent primordiales dans notre domaine, nous pouvons citer les relations spatiales entre les objets, mais aussi les relations temporelles (cas du suivi d'objet ou de l'étude d'un paysage et de son évolution).

De façon complémentaire et tout aussi primordiale, les autres types de connaissances (savoir-faire et contexte) doivent aussi être intégrées dans cette base de connaissances afin de pouvoir d'une part, permettre aux méthodes d'accéder à l'ensemble des données disponibles pour résoudre un problème, et d'autre part, proposer automatiquement des solutions répondant aux objectifs de l'utilisateur. Pour cela, les données, les processus et les résultats devront être formalisés dans la base de connaissances.

Intégration de relations spatiales Les relations spatiales entre les objets ne sont que rarement prises en compte par les méthodes de segmentation ou d'extraction d'information à partir d'images, alors qu'elles représentent une connaissance forte des experts, notamment pour la recherche d'objets ou de structures complexes (parc, zones industrielles, aéroport, etc.). Parmi les travaux sur ce sujet, nous pouvons citer ceux de Lopez-Ornelas & Sèdes (2007) qui proposent une approche morphologique de segmentation basée sur une description de l'image par des graphes d'adjacence, Guo et al. (2009) qui modélisent les relations spatiales uniquement par l'existence ou non d'un type d'objets dans une zone (pas de relations entre les objets "à côté de...", etc.), Inglada & Michel (2009) qui utilisent le formalisme de représentation de relations spatiales RCC-8 pour la segmentation multi-échelles d'images de télédétection ainsi que Alboody et al. (2010) qui proposent d'étendre ce formalisme à un environnement dans lequel une incertitude sur les objets et leurs relations est modélisée.

Le cadre générique que nous proposons d'étudier devra permettre une utilisation conjointe d'un ensemble de données multisources, hétérogènes et complexes (optique, photo, altitude, multi-résolution). Toutes les informations sur une zone d'étude seront utilisées simultanément afin d'extraire et de caractériser au mieux les objets d'intérêt. À ces informations s'ajouteront des connaissances du domaine modélisées à partir des connaissances des experts comme la texture, la signature spectrale ou le voisinage, enrichies par les études précédentes d'une zone similaire. L'originalité de l'approche proposée est de s'abstraire de la résolution des images pour travailler uniquement sur les objets détectés et leurs relations spatiales. Contrairement aux méthodes existantes (travaillant principalement au niveau pixel), nous proposons d'extraire les objets composant la scène étudiée puis de représenter leur agencement.

Les verrous scientifiques résident alors dans la détermination des méthodes de seg-

CHAPITRE 4. CONCLUSION ET PERSPECTIVES

mentation, de caractérisation et de classification des objets guidées par une modélisation du domaine, dans la prise en compte de données hétérogènes dans les différents processus (plusieurs images à différentes dates, différentes échelles, obtenues par différents capteurs), ainsi que dans la modélisation et l'enrichissement itératif de la base des connaissances du domaine. Les verrous technologiques résident principalement dans la paramétrisation automatique des méthodes de segmentation et caractérisation, et dans la quantité et l'hétérogénéité des informations à traiter (passage à l'échelle de méthodes existantes ou à créer).

De plus, la méthode devra apporter une aide à la compréhension et à l'apprentissage des règles d'agencement spatial des objets d'intérêt dans les zones étudiées. À partir de la détection automatique des objets, de l'étude de leur contexte spatial et des règles déjà disponibles dans la base de connaissances, de nouvelles règles pourront être inférées ou des règles proposées par les experts pourront être confirmées ou infirmées de manière automatique. Le verrou scientifique réside dans une modélisation efficace des règles d'agencement spatial afin de pouvoir les utiliser dans un processus d'apprentissage automatique et d'en inférer de nouvelles ou d'en valider la pertinence.

Des recherches liées à ces perspectives ont débuté via une bourse de thèse en collaboration avec le CNES (Centre National des Études Spatiales) sur le sujet "Extraction et analyse de relations spatiales entre objets d'intérêt dans les images de télédétection guidées par des connaissances du domaine" et au dépôt d'un projet ANR porté par le LIVE - ERL 7530 (Strasbourg). Les objectifs de ce projet sont (1) de proposer une base de connaissances des formes urbaines qui permettra, à partir d'une base de données géo-historiques et multi-sources (plans, cartes, photographies aériennes, images), d'extraire, stocker et identifier, selon leur échelle de représentation, les ressources, les méthodes et les indicateurs les plus pertinents afin d'analyser les formes urbaines et leur dynamique et (2) de définir les méthodes et outils permettant de maintenir et enrichir cette base de connaissances.

Intégration de relations temporelles Le projet cité précédemment s'intéresse aussi aux relations temporelles existant entre les objets à extraire dans le cadre d'une analyse d'une série de données. Cette dimension supplémentaire apparaît comme décisive dans de nombreuses applications actuelles d'interprétation automatique d'images de télédétection. En effet, ces dernières années, la multiplication des sources de données (vecteur et raster) combinée aux documents anciens existants (plans et cartes) est une véritable opportunité pour la représentation d'un espace (par exemple urbain) aux différentes échelles spatiales (de la tache urbaine au bâtiment par exemple). Toutefois, ces données sont elles-mêmes hétérogènes, multi-échelles, et incomplètes. De plus, les techniques d'extraction actuelles sont souvent spécifiques à un capteur (ou à un type d'images) ou à une base de données. L'exploitation conjointe de cette multitude de données spatio-temporelle pose donc un certain nombre de verrous que nous souhaitons aborder dans la suite de nos travaux de recherche.

Modélisation des données, des processus et des résultats Enfin, comme indiqué précédemment, il est indispensable de disposer d'un formalisme pour représenter les connaissances sur le savoir-faire (les processus d'extraction d'informations) et le contexte du problème (données disponibles, résultats précédents, ...). L'idée est de disposer d'une base de connaissances complète, modélisant l'ensemble des processus (méthodes) disponibles et permettant de savoir le type de données qu'elles manipulent, leurs paramètres, et le type de résultats qu'elles proposent. De même, les données disponibles seraient décrites par un ensemble de méta-données et associées à des résultats provenant de l'application d'un ou plusieurs processus. Ainsi, nous pouvons imaginer un processus d'interrogation de la base via une description de ses objectifs par l'utilisateur, lui proposant automatiquement une chaîne de traitement, les données et les paramètres à utiliser afin de répondre aux mieux à ses attentes.

Les principaux verrous de cette approche sont de modéliser de manière efficace et réellement opérable l'ensemble de ces données et de définir une interface d'interaction avec l'utilisateur, lui permettant de préciser rapidement ses objectifs à partir d'une description de haut niveau sémantique et de visualiser les solutions proposées par le système.

Un premier travail sur le stockage et la navigation dans un ensemble de résultats de classification et de segmentation dans un entrepôt de données a débuté (Favre et al. 2012) récemment. Nous proposons un modèle permettant la représentation des données images décrites par un ensemble de méta-données (taille, zone géographique, date, résolution, nombre de canaux, etc.) associées à des résultats de classification ou de segmentation caractérisés par la méthode utilisée, les paramètres, des critères de qualité, etc. De plus, un outil d'interrogation permettra à l'utilisateur de naviguer efficacement dans l'ensemble de ses données et résultats.

La constitution d'un tel modèle et sa mise-en-œuvre posent un ensemble de verrous scientifiques, conceptuels et pratiques, à l'interface de plusieurs domaines des STIC comme le traitement d'images, la représentation des connaissances et la fouille de données. Cela représente un défi important que je suis prêt à relever dans les prochaines années.

Bibliographie

- Aha, D. W., Kibler, D. F. & Albert, M. K. (1991), 'Instance-based learning algorithms', *Machine Learning* **6**, 37–66.
- Ajmera, J., Bourlard, H., Lapidot, I. & McCowan, I. (2002), Unknown-multiple speaker clustering using hmm, *in* 'International Conference on Spoken Language Processing', pp. 573–576.
- Alboody, A., Sèdes, F. & Inglada, J. (2010), Modeling Topological Relations between Uncertain Spatial Regions in Geo-spatial Databases : Uncertain Intersection and Difference Topological Model (regular paper), *in* 'International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA), The Three Valleys, F, 11/04/2010-16/04/2010', IEEE Computer Society, pp. 56–68.
- Anand, S., Bell, D. & Hughes, J. (1995), The role of domain knowledge in data mining, *in* 'Proceedings of the fourth international conference on Information and knowledge management', ACM, pp. 37–43.
- Ayad, H. & Kamel, M. S. (2008), 'Cumulative voting consensus method for partitions with variable number of clusters.', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**(1), 160–173.
- Basu, S., Banerjee, A. & Mooney, R. J. (2002), Semi-supervised clustering by seeding, *in* 'Proceedings of the 19th International Conference on Machine Learning (ICML-2002)', Sydney, Australia, pp. 19–26.
- Basu, S., Banerjee, A. & Mooney, R. J. (2004), Active semi-supervision for pairwise constrained clustering, *in* 'SIAM International Conference on Data Mining'.
- Basu, S., Bilenko, M. & Mooney, R. J. (2004), A probabilistic framework for semi-supervised clustering, *in* 'International Conference on Knowledge Discovery and Data Mining', pp. 59–68.
- Bennett, K. P., Demiriz, A. & Maclin, R. (2002), Exploiting unlabeled data in ensemble methods., *in* 'ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 289–296.
- Bhanu, B., Lee, S. & Das, S. (1995), 'Adaptive image segmentation using genetic and hybrid search methods', *IEEE Transactions on Aerospace and Electronic Systems* **31**(4), 1268–1291.
- Bilenko, M., Basu, S. & Mooney, R. J. (2004), Integrating constraints and metric learning in semi-supervised clustering, *in* 'International Conference on Machine Learning', pp. 81–88.
- Bittner, T. & Winter, S. (1999), On ontology in image analysis, *in* 'International Workshop On Integrated Spatial Databases ISD'99', Vol. 1737, Lecture Notes in Computer Science, pp. 168–191.
- Blansché, A., Gançarski, P. & Korczak, J. J. (2005), Genetic algorithms for feature weighting : Evolution vs. coevolution and darwin vs. lamarck, *in* 'Proceedings of the 4th Mexican

- International Conference on Artificial Intelligence', Vol. 3789 of *Lecture Notes in Computer Science*, pp. 682–691.
- Blum, A. & Mitchell, T. (1998), Combining labeled and unlabeled data with co-training, in 'Proceedings of the Workshop on Computational Learning Theory', pp. 92–100.
- Bouchachia, A. (2007), 'Learning with partly labeled data.', *Neural Computing and Applications* **16**(3), 267–293.
- Cai, W., Chen, S. & Zhang, D. (2009), 'A simultaneous learning framework for clustering and classification', *Pattern Recognition* **42**(7), 1248–1286.
- Chawla, N. V. & Karakoulas, G. J. (2005), 'Learning from labeled and unlabeled data : An empirical study across techniques and domains.', *Journal of Artificial Intelligence Research* **23**, 331–366.
- Chen, Y., Huang, F., Tagare, H., Rao, M., Wilson, D. & Geiser, E. (2003), Using prior shape and intensity profile in medical image segmentation, in 'Proceedings of the Ninth IEEE International Conference on Computer Vision', IEEE Computer Society, pp. 1117–1124.
- Chetty, G. & Deshpande, N. (1995), Knowledge-based object recognition system, in 'Proceedings of the 8th international conference on industrial and engineering applications of artificial intelligence and expert systems', Gordon and Breach Science Publishers, Inc., pp. 459–468.
- Cleuziou, G., Exbrayat, M., Martin, L. & Sublemontier, J.-H. (2009), Cofkm : A centralized method for multiple-view clustering, in 'IEEE International Conference on Data Mining', pp. 752–757.
- Cleve, C., Kelly, M., Kearns, F. R. & Moritz, M. (2008), 'Classification of the wildland-urban interface : A comparison of pixel- and object-based classifications using high-resolution aerial photography', *Computers, Environment and Urban Systems* **32**(4), 317–326.
- Cover, T. M. & Thomas, J. A. (2006), *Elements of Information Theory*, 2 edn, John Wiley & Sons, Inc., Hoboken, NJ, USA.
- Crevier, D. & Lepage, R. (1997), 'Knowledge-based image understanding systems : a survey', *Computer Vision and Image Understanding* **67**(2), 161–185.
- Darwish, A., Leukert, K. & Reinhardt, W. (2003), Image segmentation for the purpose of object-based classification, in 'Proceedings of the IEEE International Geoscience and Remote Sensing Symposium 2003', Vol. 3, IEEE Computer Society, pp. 2039–2041.
- Davidson, I., Wagstaff, K. L., & Basu, S. (2006), Measuring constraint-set utility for partitional clustering algorithms, in 'European Conference on Principles and Practice of Knowledge Discovery in Databases', pp. 115–126.
- Davies, D. & Bouldin, D. (1979), 'A cluster separation measure', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-1**(2), 224–227.

- Demiriz, A., Bennett, K. & Embrechts, M. (1999), Semi-supervised clustering using genetic algorithms, *in* 'Intelligent Engineering Systems Through Artificial Neural Networks', pp. 809–814.
- Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the EM algorithm', *Journal of the Royal Statistical Society* **39**(1), 1–38.
- Deodhar, M. & Ghosh, J. (2007), A framework for simultaneous co-clustering and learning from complex data, *in* 'ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 250–259.
- Derivaux, S., Forestier, G., Wemmert, C. & 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)', Sophia Antipolis, France, pp. 71–81.
- Derivaux, S., Forestier, G., Wemmert, C. & Lefevre, S. (2010), 'Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation', *Pattern Recognition Letters* **31**(15), 2364–2374.
- Derivaux, S., Lefevre, S., Wemmert, C. & Korczak, J. (2007), On machine learning in watershed segmentation, *in* 'IEEE International Workshop on Machine Learning for Signal Processing', pp. 187–192.
- Dhillon, I. S. (2001), Co-clustering documents and words using bipartite spectral graph partitioning, *in* 'KDD '01 : Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining', pp. 269–274.
- Dimitriadou, E., Weingessel, A. & Hornik, K. (2002), 'A combination scheme for fuzzy clustering.', *International Journal of Pattern Recognition and Artificial Intelligence* **16**(7), 901–912.
- Draper, B., Collins, A., Brolio, J., Hanson, A. & Riseman, E. (1989), 'The schema system', *International Journal of Computer Vision* **2**(3), 209–250.
- Duarte, F., Fred, A., Lourenco, A. & Rodrigues, M. (2005), 'Weighting cluster ensembles in evidence accumulation clustering', *Portuguese Conference on Artificial Intelligence* pp. 159–167.
- Durand, N., Derivaux, S., Forestier, G., Wemmert, C., Gançarski, P., Boussaid, O. & Puissant, A. (2007), Ontology-based object recognition for remote sensing image interpretation, *in* 'IEEE International Conference on Tools with Artificial Intelligence', Vol. 1, IEEE Computer Society, Patras, Greece, pp. 472–479.
- Dzeroski, S. & Lavrac, N. (2001), *Relational Data Mining*, Springer.
- Eick, C., Zeidat, N. & Zhao, Z. (2004), Supervised clustering—algorithms and benefits, *in* 'IEEE International Conference on Tools with Artificial Intelligence', pp. 774–776.
- Faceli, K., de Carvalho, A. & de Souto, M. (2006), 'Multi-objective clustering ensemble', *International Conference on Hybrid Intelligent Systems* pp. 51–51.

BIBLIOGRAPHIE

- Faceli, K., de Carvalho, A. & de Souto, M. (2007), Multi-objective clustering ensemble with prior knowledge, *in* 'Advances in Bioinformatics and Computational Biology', Vol. 4643, Springer, pp. 34–45.
- Favre, C., Wemmert, C. & Forestier, G. (2012), Olap for navigating within the results of remote sensing images mining, *in* 'Workshop Complex Data Mining in a GeoSpatial Context - AGILE International Conference on Geographic Information Science'.
- Feitosa, R. Q., Costa, G. A., Cazes, T. B. & Feijo, B. (2006), A genetic approach for the automatic adaptation of segmentation parameters, *in* 'Proceedings of the International Conference on Object-based Image Analysis'.
- Fern, X. Z. & Brodley, C. E. (2004), Solving cluster ensemble problems by bipartite graph partitioning, *in* 'ICML '04 : Proceedings of the twenty-first international conference on Machine learning', ACM, New York, NY, USA, p. 36.
- Fonseca, F., Egenhofer, M., Agouris, P. & Camara, G. (2002), 'Using ontologies for integrated geographic information systems', *Transactions in Geographic Information Systems* **6**(3), 231–257.
- Forestier, G., Gançarski, P. & Wemmert, C. (2010), 'Collaborative clustering with background knowledge', *Data and Knowledge Engineering* **69**(2), 211–228.
- Forestier, G., Puissant, A., Wemmert, C. & Gançarski, P. (2012), 'Knowledge-based region labeling for remote sensing image interpretation', *Computers, Environment and Urban Systems*.
- Forestier, G., Wemmert, C. & Gançarski, P. (2008a), 'Multi-source images analysis using collaborative clustering', *EURASIP Journal on Advances in Signal Processing* **2008**, 11.
- Forestier, G., Wemmert, C. & Gançarski, P. (2008b), Semi-supervised collaborative clustering with partial background knowledge, *in* 'Workshop on Mining Complex Data', IEEE International Conference on Data Mining, Pisa, Italy, pp. 211–217.
- Forestier, G., Wemmert, C. & Gançarski, P. (2010a), Background knowledge integration in clustering using purity indexes, *in* 'International Conference on Knowledge Science, Engineering & Management', Vol. 6291 of *Lecture Notes in Computer Science*, Springer, Belfast, pp. 28–38.
- Forestier, G., Wemmert, C. & Gançarski, P. (2010b), Towards conflict resolution in collaborative clustering, *in* 'IEEE International Conference on Intelligent Systems', London, pp. 361–366.
- Fred, A. & Jain, A. (2005), 'Combining multiple clusterings using evidence accumulation', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(6), 835–850.
- Gabrys, B. & Petrikieva, L. (2004), 'Combining labelled and unlabelled data in the design of pattern classification systems', *International journal of approximate reasoning* **35**(3), 251–273.
- Gançarski, P. & Wemmert, C. (2007), 'Collaborative multi-step mono-level multi-strategy classification', *Multimedia Tools and Applications* **35**(1), 1–27.

- Gao, J., Tan, P. & Cheng, H. (2006), Semi-supervised clustering with partial background information, in 'SIAM International Conference on Data Mining', pp. 489–493.
- Gigandet, X., Cuadra, M., Pointet, A., Cammoun, L., Caloz, R. & Thiran, J.-P. (2005), Region-based satellite image classification : method and validation, in 'IEEE International Conference on Image Processing', Vol. 3, IEEE Computer Society, pp. 832–836.
- Goldberg, D. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Kluwer Academic Publishers.
- Goldman, S. & Zhou, Y. (2000), Enhancing supervised learning with unlabeled data, in 'International Conference on Machine Learning', pp. 327–334.
- Grau, V., Mewes, A., Alcaniz, M., Kikinis, R. & Warfield, S. (2004), 'Improved watershed transform for medical image segmentation using prior information', *IEEE Transactions on Medical Imaging* **23**(4), 447–458.
- Grozavu, N. & Bennani, Y. (2010), Classification collaborative non supervisée, in 'Conférence francophone sur l'apprentissage automatique (CAP)'.
- Gruber, T. (1995), 'Toward principles for the design of ontologies used for knowledge sharing', *International Journal of Human Computer Studies* **43**(5/6), 907–928.
- Guo, D., Xiong, H., Atluri, V. & Adam, N. (2009), 'Object discovery in high-resolution remote sensing images : a semantic perspective', *Knowledge and Information Systems* **19**(2), 211–233.
- Haker, S., Sapiro, G. & Tannenbaum, A. (2000), 'Knowledge-based segmentation of SAR data with learned priors', *IEEE Transactions on Image Processing* **9**(2), 299–301.
- Hamarneh, G. & Li, X. (2007), 'Watershed segmentation using prior shape and appearance knowledge', *Image and Vision Computing*.
- Handl, J. & Knowles, J. (2007), 'An evolutionary approach to multiobjective clustering', *IEEE Transactions on Evolutionary Computation* **11**(1), 56–76.
- Handl, J. & Knowles, J. D. (2006), On semi-supervised clustering via multiobjective optimization, in 'Genetic and Evolutionary Computation Conference', pp. 1465–1472.
- Haris, K., Efstradiadis, S. N., Maglaveras, N. & Katsaggelos, A. K. (1998), 'Hybrid image segmentation using watersheds and fast region merging', *IEEE Transaction On Image Processing* **7**(12), 1684–1699.
- He, Z., Xu, X. & Deng, S. (2005), 'A cluster ensemble method for clustering categorical data', *Information Fusion* **6**, 143–151.
- Hu, T., Yu, Y., Xiong, J. & Sung, S. Y. (2006), 'Maximum likelihood combination of multiple clusterings.', *Pattern Recognition Letters* **27**(13), 1457–1464.
- Hughes, G. (1968), 'On the mean accuracy of statistical pattern recognizers', *IEEE Transactions on Information Theory* **14**(1), 5 – 63.
- Inglaada, J. & Michel, J. (2009), 'Qualitative spatial reasoning for high-resolution remote sensing image analysis', *Geoscience and Remote Sensing, IEEE Transactions on* **47**(2), 599–612.

BIBLIOGRAPHIE

- Jin, X. & Davis, C. H. (2005), 'Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information', *EURASIP Journal on Applied Signal Processing* **2005**(14), 2196–2206.
- Joachims, T. (1999), Transductive inference for text classification using support vector machines, in 'International Conference on Machine Learning', pp. 200–209.
- Karem, F., Dhibi, M. & Martin, A. (2012), Combination of supervised and unsupervised classification using the theory of belief functions, in 'Proceedings of the International Conference on Belief Functions'.
- Karypis, G., Aggarwal, R., Kumar, V. & Shekhar, S. (1997), Multilevel hypergraph partitioning : Application in vlsi domain., in 'DAC', pp. 526–529.
- Kira, K. and Rendell, L. A. (1992), A practical approach to feature selection, in 'Proceedings of the ninth international workshop on Machine learning', Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 249–256.
- Kittler, J., Hatef, M., Duin, R. P. W. & Matas, J. (1998), 'On combining classifiers.', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(3), 226–239.
- Klein, D., Kamvar, S. & Manning, C. (2002), From instance-level constraints to space-level constraints : Making the most of prior knowledge in data clustering, in 'In The Nineteenth International Conference on Machine Learning', pp. 307–314.
- Knobbe, A., de Haus, M. & Siebes, A. (2001), 'Propositionalisation and Aggregates', *Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD-2001)* pp. 277–288.
- Konak, A., Coit, D. & Smith, A. (2006), 'Multi-objective optimization using genetic algorithms : A tutorial', *Reliability Engineering & System Safety* **91**(9), 992–1007.
- Kopanas, I., Avouris, N. & Daskalaki, S. (2002), 'The role of domain knowledge in a large scale data mining project', *Methods and Applications of Artificial Intelligence* pp. 746–746.
- Krogel, M.-A. & Wrobel, S. (2002), Feature selection for propositionalization, in 'Proceedings of the 5th International Conference on Discovery Science', Lecture Notes in Computer Science, pp. 430–434.
- Kuncheva, L. I. (2004), *Combining Pattern Classifiers : Methods and Algorithms*, Wiley-Interscience.
- Lachiche, N. (2005), Good and bad practices in propositionalisation, in 'Proceedings of Advances in Artificial Intelligence, 9th Congress of the Italian Association for Artificial Intelligence (AI*IA'05)', pp. 50–61.
- Law, M., Topchy, A. & Jain, A. (2004), Multiobjective data clustering, in 'IEEE Conference on Computer Vision and Pattern Recognition', Vol. 2, pp. 424–430.
- Lefèvre, S. (2007), Knowledge from markers in watershed segmentation, in 'IAPR International Conference on Computer Analysis of Images and Patterns (CAIP)', Vol. 4673 of *Lecture Notes in Computer Sciences*.

- Lefèvre, S., Weber, J. & Sheeren, D. (2006), Automatic building extraction in vhr images using advanced morphological operators, *in* 'Proceedings of the IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (URBAN)'.
- Levner, I. & Zhang, H. (2007), 'Classification-driven watershed segmentation', *IEEE Transaction on Image Processing* **16**(5), 1437–1445.
- Liu, S., Thonnat, M. & Berthod, M. (1994), Automatic classification of planktonic foraminifera by a knowledge-based system, *in* 'Proceedings 10th Conference on Artificial Intelligence for Applications', IEEE Computer Society, pp. 358–364.
- Liu, Y., Li, M., Mao, L., Xu, F. & Huang, S. (2006), 'Review of remotely sensed imagery classification patterns based on object-oriented image analysis.', *Chinese Geographical Science* **16**(3), 282–288.
- Loia, V., Pedrycz, W. & Senatore, S. (2007), 'Semantic web content analysis : A study in proximity-based collaborative clustering', *Fuzzy Systems, IEEE Transactions on* **15**(6), 1294–1312.
- Long, B., Zhang, Z. M. & Yu, P. S. (2005), Combining multiple clusterings by soft correspondence., *in* 'International Conference on Data Mining', IEEE Computer Society, pp. 282–289.
- Lopez-Ornelas, E. & Sèdes, F. (2007), 'Génération de descripteurs d'images satellitaires à thrs', *Revue européenne de géographie* **1**(385), (en ligne).
- URL:** <http://www.cybergeo.eu/index8212.html>
- MacQueen, J. (1967), Some methods for classification and analysis of multivariate observations, *in* 'Berkeley Symposium on Mathematical Statistics and Probability', pp. 281–297.
- Maillet, N. (2005), Ontology Based Object Learning and Recognition, Thèse de doctorat, Université de Nice.
- Manning, C. D., Raghavan, P. & Schütze, H. (2008), *Introduction to Information Retrieval*, Cambridge University Press.
- Matsuyama, T. & Hwang, V.-S. (1990), *SIGMA - A Knowledge-Based Aerial Image Understanding System*, Plenum Press New York USA.
- Megiddo, N. & Supowit, K. J. (1984), 'On the complexity of some common geometric location problems.', *SIAM Journal on Computing* **13**(1), 182–196.
- Meyer, F. & Beucher, S. (1990), 'Morphological segmentation', *Journal of Visual Communication and Image Representation* **1**(1), 21–46.
- Mezaris, V., Kompatsiaris, I. & Strintzis, M. G. (2004), 'Region-based image retrieval using an object ontology and relevance feedback', *EURASIP Journal on Advances in Signal Processing* **2004**(1), 886–901.
- Minsky, M. (1975), A framework for representing knowledge, *in* 'The Psychology of Computer Vision', McGraw-Hill, pp. 211–281.

BIBLIOGRAPHIE

- Mitchell, T. (1997), *Machine Learning*, McGraw Hill, chapter 6.
- Mitra, H., Banka & Pedrycz, W. (2006), ‘Rough-fuzzy collaborative clustering’, *IEEE Transactions on Systems, Man, and Cybernetics* **36**, 795–805.
- Najman, L. & Schmitt, M. (1996), ‘Geodesic saliency of watershed contours and hierarchical segmentation’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18**(12), 1163–1173.
- Nguyen, N. & Caruana, R. (2007), Consensus clusterings., in ‘International Conference on Data Mining’, IEEE Computer Society, pp. 607–612.
- Nigam, K., McCallum, A. K., Thrun, S. & Mitchell, T. M. (2000), ‘Text classification from labeled and unlabeled documents using EM’, *Machine Learning* **39**(2/3), 103–134.
- Ogiela, M. R. & Tadeusiewicz, R. (2008), *Modern Computational Intelligence Methods for the Interpretation of Medical Images*, Springer.
- Oliveira, J. V. d. & Pedrycz, W. (2007), *Advances in Fuzzy Clustering and its Applications*, John Wiley & Sons, Inc., New York, NY, USA.
- Panagi, P., Dasiopoulou, S., Papadopoulos, G. T., Kompatsiaris, I. & Strintzis, M. G. (2006), A genetic algorithm approach to ontology-driven semantic image analysis, in ‘IEEE International Conference of Visual Information Engineering (VIE 2006)’, IEEE Computer Society, pp. 132–137.
- Paredes, R. & Vidal, E. (2006), ‘Learning weighted metrics to minimize nearest-neighbor classification error’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**(7), 1100–1110.
- Pedrycz, W. (2002), ‘Collaborative fuzzy clustering’, *Pattern Recognition Letters* **23**, 1675–1686.
- Pedrycz, W. & Rai, P. (2008), ‘A multifaceted perspective at data analysis : A study in collaborative intelligent agents’, *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on* **38**(4), 1062–1072.
- Perlich, C. & Provost, F. (2003), ‘Aggregation-based Feature Invention and Relational Concept Classes’, *Proceedings of the Ninth International Conference on Knowledge Discovery and Data Mining (KDD-2003)* pp. 167–176.
- Perner, P. (1994), ‘A knowledge-based image-inspection system for automatic defect recognition, classification, and process diagnosis’, *Machine vision and applications* **7**(3), 135–147.
- Peteri, R., Celle, J. & Ranchin, T. (2003), Detection and extraction of road networks from high resolution satellite mages, in ‘Proceedings of the IEEE International Conference on Image Processing’, IEEE Computer Society, pp. 301–304.
- Pignalberi, G., Cucchiara, R., Cinque, L. & Levialdi, S. (2003), ‘Tuning range image segmentation by genetic algorithm’, *EURASIP Journal on Applied Signal Processing* **2003**(8), 780–790.

- Rand, W. M. (1971), 'Objective criteria for the evaluation of clustering methods', *Journal of the American Statistical Association* **66**, 622–626.
- Raskutti, B., Ferri, H. L. & Kowalczyk, A. (2002), Combining clustering and co-training to enhance text classification using unlabelled data., in 'ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 620–625.
- Salvador, S. & Chan, P. (2004), Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms, in 'IEEE International Conference on Tools with Artificial Intelligence', pp. 576–584.
- Seeger, M. (2002), Learning with labeled and unlabeled data, Technical report, University of Edinburgh.
- Solomonoff, A., Mielke, A., Schmidt, M. & Gish, H. (1998), Clustering speakers by their voices, in 'Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on', Vol. 2, pp. 757–760.
- Song, A. & Ciesielski, V. (2003), Fast texture segmentation using genetic programming, in 'Proceedings of the IEEE Congress on Evolutionary Computation', Vol. 3, IEEE Computer Society, pp. 2126–2133.
- Strehl, A. & Ghosh, J. (2002), 'Cluster ensembles – a knowledge reuse framework for combining multiple partitions', *Journal on Machine Learning Research* **3**, 583–617.
- Topchy, A. P., Jain, A. K. & Punch, W. F. (2003), Combining multiple weak clusterings., in 'International Conference on Data Mining', IEEE Computer Society, pp. 331–338.
- van Rijsbergen, C. J. (1979), *Information Retrieval*, Butterworths, London.
- Vincent, L. & Soille, P. (1991), 'Watersheds in digital spaces : An efficient algorithm based on immersion simulations', *IEEE Pattern Analysis and Machine Intelligence* **13**(6), 583–598.
- Volle, M. (1997), *Analyse des données*, Economica.
- Wagstaff, K., Cardie, C., Rogers, S. & Schroedl, S. (2001), Constrained k-means clustering with background knowledge, in 'International Conference on Machine Learning', pp. 557–584.
- Wagstaff, K. L. (2007), Value, cost, and sharing : Open issues in constrained clustering, in 'International Workshop on Knowledge Discovery in Inductive Databases', pp. 1–10.
- Wang, L., Khan, L. & Breen, C. (2002), Object boundary detection for ontology-based image classification, in 'Proceedings of the 3rd International Workshop on Multimedia Data Mining (MDM/KDD2002)', pp. 51–61.
- Wemmert, C. & Forestier, G. (2011), 'Slemc : Apprentissage semi-supervisé enrichi par de multiples clusterings', *Revue des Nouvelles Technologies de l'Information* **E.21**, 147–169.
- Wemmert, C., Forestier, G. & Derivaux, S. (2009), *Applications of Supervised and Unsupervised Ensemble Methods, chapter Improving Supervised learning with Multiple Clusterings*, Vol. 245 of *Studies in Computational Intelligence*, Springer, pp. 135–148.

BIBLIOGRAPHIE

- Wemmert, C. & Gançarski, P. (2002a), A multi-view voting method to combine unsupervised classifications, *in* 'Artificial Intelligence and Applications', Malaga, Spain, pp. 447–452.
- Wemmert, C. & Gançarski, P. (2002b), Urban thematical zones construction from remote sensing data by unsupervised classification, *in* '23rd Symposium on Urban Data Management', (6 pages), Prague, Rep. Tchèque.
- Wemmert, C., Gançarski, P. & Korczak, J. (2000), 'A collaborative approach to combine multiple learning methods', *International Journal on Artificial Intelligence Tools* **9**(1), 59–78.
- Wemmert, C., Puissant, A., Forestier, G. & Gançarski, P. (2009), 'Multiresolution remote sensing image clustering', *IEEE Geoscience and Remote Sensing Letters* **6**(3), 533 – 537.
- Whiteside, T. & Ahmad, W. (2005), A comparison of object-oriented and pixel-based classification methods for mapping land cover in northern australia, *in* 'Proceedings Spatial Sciences Institute Biennial Conference (SSC 2005)', pp. 1225–1231.
- Witten, I. H. & Frank, E. (2005), *Data mining : practical machine learning tools and techniques*, Morgan Kaufmann.
- Yager, N. & Sowmya, A. (2004), Support vector machines for road extraction from remotely sensed images, *in* 'Computer Analysis of Images and Patterns', pp. 285–292.
- Zhang, Y. J. (1996), 'A survey on evaluation methods for image segmentation', *Pattern Recognition* **29**(8), 1335–1346.
- Zhao, H., Kumagai, J., Nakagawa, M. & Shibasaki, R. (2002), Semi-automatic road extraction from high-resolution satellite image, *in* 'Proceedings of the ISPRS Symposium on Photogrammetry and Computer Vision', p. A : 406.
- Zhou, Z.-H. & Tang, W. (2006), 'Clusterer ensemble.', *Knowledge-Based Systems* **19**(1), 77–83.
- Zhou, Z.-H., Zhan, D.-C. & Yang, Q. (2007), Semi-supervised learning with very few labeled training examples., *in* 'AAAI International Conference on Artificial Intelligence', pp. 675–680.
- Zlatoff, N., Tellez, B. & Baskurt, A. (2004), Image understanding using domain knowledge, *in* 'Proceedings of Recherche d'information Assistée par Ordinateur (RIA0)', pp. 277–290.

Annexes

A

Méthodes de classification semi-supervisée

A.1	Static Labeling	112
A.2	Dynamic labeling	112
A.3	Étiquetage des clusters à la majorité	112
A.4	Single Representative Insertion/Deletion Steepest Descent Hill Climbing with Randomized Restart	113
A.5	Supervised Clustering using Evolutionary Computing	114
A.6	Refined clustering	115
A.7	Seeded-Kmeans	116
A.8	Constrained-Kmeans	116

Soit X un ensemble de n objets $x_j \in X$. Nous nous plaçons dans un cas de classification à q classes avec m exemples étiquetés et l objets non-étiquetés. Nous faisons l'hypothèse que m est très faible et $l >> m$.

Soit L l'ensemble des objets étiquetés de X :

$$L = \{(x_1, y_1), \dots, (x_m, y_m)\} \quad (\text{A.1})$$

avec $y_i \in \{1, \dots, q\}$ les étiquettes des classes des exemples.

Soit U l'ensemble des objets non-étiquetés de X :

$$U = \{(x_{m+1}, 0), \dots, (x_{m+l}, 0)\} \quad (\text{A.2})$$

avec 0 signifiant qu'aucune étiquette n'est associée à cet objet.

L'objectif de la classification semi-supervisée est de construire un classifieur basé sur tout l'ensemble d'apprentissage X . Ce classifieur peut être vu comme une fonction associant une des q classes à chaque objet de x . Il peut être défini formellement de la manière suivante :

$$y = C_X(x) : y \in \{1, \dots, q\} \quad (\text{A.3})$$

A.1 Static Labeling

Algorithme 7: *Static labeling (SL)*

construire une classification C_L à partir d'un ensemble d'exemples étiquetés L
soit $W = \{(x_j, y_j) : y_j = C_L(x_j), x_j \in U\}$
construire le classifieur final $C_{L \cup W}$

A.2 Dynamic labeling

Algorithme 8: *Dynamic labeling (DL)*

soit $U' = U$ et $W' = \emptyset$
construire un classifieur $C_{L \cup W'}$
pour tous les $x_j \in U'$ *choisi en fonction de leur degré de confiance dans la classe faire*
 $\left[\begin{array}{l} W' := W' \cup \{(x_j, t_j) : t_j = C_{L \cup W'}(x_j)\} \\ U' := U' \setminus \{(x_j, 0)\} \end{array} \right]$
construire la classification finale $C_{L \cup W'}$

A.3 Étiquetage des clusters à la majorité

A.4. SINGLE REPRESENTATIVE INSERTION/DELETION STEEPEST DECENT HILL CLIMBING WITH RANDOMIZED RESTART

Algorithme 9: Étiquetage des clusters à la majorité (CLM)

construire un clustering $K = \{K_l, l = 1 \dots k\}$ à partir de X

soit $LK := \emptyset$ **pour tous les** $K_l, l = 1 \dots k$ **faire**

si $\sum_{j=1}^q c_{lj} \neq 0$ **alors**

$$y_{K_l} = \arg \max_{j \in \{1 \dots q\}} (c_{lj})$$

$$W_l = \{(x_i, y_{K_l}), x_i \in K_l\}$$

$$LK := LK \cup K_l$$

pour tous les $K_u : \sum_{j=1}^q c_{uj} = 0$ **faire**

$$K_m = \arg \max_{K_l \in LK} (\Delta(K_l, K_u))$$

étiqueter tous les objets du cluster K_u avec l'étiquette y_{K_m}

$$W_u = \{(x_i, y_{K_m}), x_i \in K_u\}$$

construire le classifieur final $C_{W_1 \cup \dots \cup W_k}$

A.4 Single Representative Insertion/Deletion Steepest Decent Hill Climbing with Randomized Restart

Algorithme 10: *Single Representative Insertion/Deletion Steepest Descent Hill Climbing with Randomized Restart (SRIDHCR)*

```

pour  $i = 1$  à  $r$  faire
    rep :=  $\{(x_i, y_i) \in L : \text{choisis aléatoirement}\}$  et  $q \leq \|rep\| \leq 2q$ 
    tant que non fin faire
        soit  $s^1 = rep \cup (x_i, y_i) : x_i \notin rep$ 
        soit  $s^2 = rep \setminus (x_i, y_i) : x_i \in rep$ 
        soit  $S = \arg \min_{s \in \{s^1, s^2\}} \Pi(s)$ 
        si  $\Pi(S) < \Pi(rep)$  alors
            rep := S
        sinon si  $\Pi(S) = \Pi(rep)$  et  $\|S\| > \|rep\|$  alors
            rep := S
        sinon
            fin
    
```

la meilleure solution est conservée

A.5 Supervised Clustering using Evolutionary Computing

Algorithme 11: Supervised Clustering using Evolutionary Computing (**SCEC**)

```

 $pop_0 = \{S_r = \{s_i^r = (x_i, y_i) \in L : \text{choisis aléatoirement}\}$ 
 $1 \leq r \leq ps\}$ 
pour  $i = 1$  à  $N$  faire
     $mu_i = \text{mutation}(pop_{i-1})$ 
     $cr_i = \text{croisement}(pop_{i-1})$ 
     $co_i = \text{copie}(pop_{i-1})$ 
     $pop_i = \text{select\_by\_k\_tournoi}(mu_i, cr_i, co_i)$ 
    pour tous les  $S_r \in pop_i$  faire
        soit  $K^r$  le clustering correspondant aux représentants  $S_r$ 
        pour tous les  $l \in [1 \dots k]$  faire
             $K_l^r = \{(x_j, y_j) : \text{dist}(x_j, s_l^r) \text{ est minimal}\}$ 
            calculer  $\Pi(S_r)$ 

```

la meilleure solution S_r est conservée

A.6 Refined clustering

Algorithme 12: *Refined clustering (RC)*

construire un clustering $K = \{K_l, l = 1 \dots k\}$ sur X avec k relativement petit
pour tous les $K_l, l = 1 \dots k$ **faire**

```

si  $\sum_{j=1}^q c_{lj} \neq 0$  alors
    tant que  $nbc > 1$  faire
         $nbc := 1$ 
        pour tous les  $m \in [1 \dots q]$  faire
            soit  $r_m = \frac{c_{lm}}{\sum_{j=1}^q c_{lj}}$ 
            si  $r_m < \Theta$  et  $\forall i \in [1 \dots k], i \neq l, g_{im} = 0$  alors
                 $K' := scission(K, l)$ 
            sinon si  $r_m \geq \Theta$  alors
                 $nbc := nbc + 1$ 
                 $K' := scission(K, l)$ 

```

appliquer l'algorithme **CLM** à K'

A.7 Seeded-Kmeans

Algorithme 13: *Seeded-Kmeans (SK)*

$$\mu_i^{(0)} = \frac{1}{|S_i|} \sum_{x \in S_i} x \text{ pour } i = 1 \dots q$$

$$t = 0$$

répéter

```

 $K^{(t+1)} = \left\{ K_i^{(t+1)}, i = 1 \dots q \right\}$ 
avec  $K_i^{(t+1)} = \left\{ x \in X : i = \arg \min_h \|x - \mu_h^{(t)}\|^2 \right\}$ 
 $\mu_i^{(t+1)} = \frac{1}{|K_i^{(t+1)}|} \sum_{x \in K_i^{(t+1)}} x \text{ pour } i = 1 \dots q$ 
 $t = t + 1$ 

```

jusqu'à convergence

A.8 Constrained-Kmeans

Algorithme 14: *Constrained-Kmeans (CK)*

$$\mu_i^{(0)} = \frac{1}{|S_i|} \sum_{x \in S_i} x \text{ pour } i = 1 \dots q$$

$$t = 0$$
répéter

$K^{(t+1)} = \left\{ K_i^{(t+1)}, i = 1 \dots q \right\}$ avec $K_i^{(t+1)} = S_i \cup \left\{ x \in X : i = \arg \min_h \ x - \mu_h^{(t)}\ ^2 \right\}$ $\mu_i^{(t+1)} = \frac{1}{ K_i^{(t+1)} } \sum_{x \in K_i^{(t+1)}} x \text{ pour } i = 1 \dots q$

$$t = t + 1$$
jusqu'à convergence

B

Données et résultats pour la méthode SLEM C

B.1	Données utilisées	119
B.2	Résultats comparatifs	120

B.1 Données utilisées

Données	#classes	#attributs	#objets
iris	3	4	150
wine	3	13	178
ionosphere	2	34	351
diabetes	2	8	768
breast-w	2	9	699
anneal	5	38	898
remote	7	36	6435

Tableau B.1
Informations sur les différents jeux de données utilisés.

B.2 Résultats comparatifs

B.2. RÉSULTATS COMPARATIFS

	1-NN	NB	C45	SL	DL
iris(2)	78, 02(± 13 , 48)	67, 53(± 15 , 72)	56, 52(± 11 , 13)	71, 06(± 13 , 85)	79, 27(± 14 , 18)
iris(4)	85, 61(± 11 , 36)	76, 08(± 13 , 58)	72, 01(± 17 , 77)	81, 80(± 14 , 02)	88, 84(± 11 , 64)
iris(8)	91, 50(± 4 , 94)	93, 59(± 3 , 39)	90, 59(± 5 , 51)	93, 01(± 3 , 98)	93, 33(± 2 , 93)
iris(16)	93, 46(± 5 , 38)	94, 20(± 3 , 54)	91, 73(± 4 , 85)	93, 70(± 3 , 33)	94, 20(± 3 , 91)
wine(2)	76, 67(± 13 , 889)	54, 62(± 14 , 19)	51, 32(± 15 , 07)	72, 53(± 17 , 22)	80, 92(± 15 , 29)
wine(4)	88, 92(± 8 , 046)	77, 88(± 11 , 89)	70, 39(± 11 , 10)	87, 27(± 9 , 57)	90, 69(± 9 , 22)
wine(8)	91, 95(± 5 , 250)	88, 77(± 8 , 85)	79, 90(± 7 , 44)	93, 43(± 4 , 29)	94, 76(± 3 , 52)
wine(16)	93, 50(± 3 , 636)	95, 53(± 3 , 73)	85, 04(± 6 , 10)	95, 52(± 3 , 67)	96, 09(± 3 , 65)
breast-w(2)	81, 88(± 16 , 93)	82, 39(± 15 , 30)	70, 68(± 17 , 74)	85, 22(± 16 , 45)	84, 72(± 17 , 98)
breast-w(4)	87, 08(± 18 , 72)	86, 65(± 19 , 26)	80, 83(± 17 , 94)	87, 94(± 19 , 24)	89, 91(± 18 , 54)
breast-w(8)	89, 14(± 17 , 13)	91, 08(± 16 , 99)	84, 02(± 16 , 09)	90, 74(± 17 , 01)	91, 01(± 17 , 06)
breast-w(16)	93, 18(± 3 , 27)	94, 69(± 0 , 80)	89, 03(± 2 , 50)	94, 48(± 1 , 02)	94, 75(± 1 , 26)
diabetes(2)	60, 62(± 9 , 40)	59, 23(± 10 , 48)	55, 18(± 11 , 99)	59, 14(± 9 , 82)	60, 45(± 10 , 95)
diabetes(4)	62, 92(± 6 , 01)	61, 06(± 7 , 74)	62, 11(± 7 , 59)	62, 06(± 6 , 07)	64, 93(± 7 , 30)
diabetes(8)	65, 00(± 5 , 05)	66, 24(± 4 , 76)	64, 35(± 6 , 53)	65, 40(± 5 , 78)	67, 67(± 5 , 35)
diabetes(16)	66, 07(± 3 , 93)	69, 56(± 4 , 29)	66, 53(± 4 , 14)	69, 13(± 3 , 35)	69, 61(± 2 , 78)
ionos.(2)	57, 36(± 10 , 62)	56, 35(± 9 , 96)	50, 35(± 6 , 58)	58, 12(± 12 , 21)	59, 70(± 14 , 00)
ionos.(4)	70, 08(± 10 , 59)	68, 18(± 9 , 84)	66, 20(± 12 , 61)	72, 09(± 10 , 01)	71, 51(± 11 , 49)
ionos.(8)	74, 08(± 10 , 19)	79, 93(± 7 , 62)	77, 21(± 7 , 04)	79, 24(± 7 , 42)	76, 77(± 6 , 32)
ionos.(16)	77, 92(± 15 , 98)	82, 70(± 16 , 16)	82, 09(± 16 , 19)	80, 35(± 15 , 79)	76, 90(± 15 , 21)
anneal(2)	76, 71(± 4 , 54)	75, 88(± 4 , 06)	74, 87(± 7 , 55)	75, 26(± 7 , 19)	70, 35(± 10 , 35)
anneal(4)	79, 74(± 3 , 91)	78, 13(± 3 , 34)	78, 51(± 4 , 95)	78, 01(± 5 , 64)	72, 33(± 9 , 63)
anneal(8)	85, 19(± 3 , 00)	81, 53(± 3 , 78)	82, 59(± 5 , 21)	80, 49(± 4 , 79)	77, 56(± 6 , 68)
anneal(16)	90, 91(± 1 , 46)	87, 27(± 3 , 42)	89, 84(± 4 , 53)	87, 16(± 2 , 89)	83, 45(± 3 , 88)
remote(2)	85, 51(± 13 , 29)	65, 40(± 13 , 46)	51, 41(± 12 , 94)	83, 52(± 15 , 28)	86, 01(± 13 , 21)
remote(4)	95, 59(± 6 , 70)	79, 63(± 11 , 16)	72, 96(± 8 , 74)	94, 07(± 8 , 14)	95, 67(± 6 , 64)
remote(8)	98, 98(± 1 , 55)	94, 10(± 7 , 04)	84, 39(± 5 , 93)	98, 40(± 1 , 41)	99, 27(± 0 , 72)
remote(16)	99, 70(± 0 , 75)	98, 81(± 2 , 20)	88, 44(± 4 , 94)	99, 18(± 1 , 57)	99, 40(± 1 , 13)

Tableau B.2
Précision des résultats - partie I

ANNEXE B. DONNÉES ET RÉSULTATS POUR LA MÉTHODE SLEM

	CLM	RC	SCEC	SRIDHCR	SK
iris(2)	64, 15($\pm 14, 09$)	64, 54($\pm 13, 67$)	72, 46($\pm 12, 89$)	73, 38($\pm 15, 32$)	64, 97($\pm 13, 85$)
iris(4)	74, 92($\pm 14, 48$)	74, 86($\pm 14, 12$)	81, 16($\pm 12, 52$)	83, 43($\pm 13, 04$)	74, 60($\pm 14, 39$)
iris(8)	91, 63($\pm 5, 18$)	90, 26($\pm 6, 74$)	87, 58($\pm 8, 47$)	85, 62($\pm 7, 58$)	89, 47($\pm 7, 65$)
iris(16)	93, 21($\pm 4, 97$)	92, 84($\pm 4, 96$)	91, 97($\pm 7, 41$)	92, 09($\pm 4, 94$)	92, 59($\pm 6, 12$)
wine(2)	55, 82($\pm 14, 58$)	56, 18($\pm 14, 87$)	73, 65($\pm 15, 86$)	78, 47($\pm 15, 64$)	54, 73($\pm 13, 55$)
wine(4)	82, 98($\pm 13, 65$)	82, 98($\pm 13, 33$)	86, 45($\pm 9, 73$)	86, 53($\pm 12, 49$)	82, 29($\pm 13, 11$)
wine(8)	89, 84($\pm 9, 12$)	89, 53($\pm 8, 72$)	91, 28($\pm 5, 72$)	92, 46($\pm 3, 60$)	89, 43($\pm 8, 58$)
wine(16)	94, 87($\pm 3, 79$)	94, 55($\pm 4, 11$)	93, 25($\pm 5, 65$)	93, 41($\pm 5, 31$)	94, 22($\pm 3, 90$)
breast-w(2)	79, 44($\pm 16, 79$)	79, 48($\pm 16, 76$)	72, 59($\pm 29, 74$)	73, 56($\pm 29, 98$)	71, 91($\pm 28, 48$)
breast-w(4)	87, 44($\pm 19, 62$)	87, 60($\pm 19, 59$)	82, 76($\pm 25, 13$)	80, 89($\pm 25, 21$)	84, 06($\pm 24, 94$)
breast-w(8)	91, 27($\pm 17, 03$)	91, 12($\pm 17, 00$)	88, 37($\pm 17, 01$)	88, 11($\pm 17, 09$)	90, 90($\pm 16, 98$)
breast-w(16)	94, 62($\pm 1, 28$)	94, 67($\pm 1, 35$)	90, 36($\pm 16, 86$)	89, 73($\pm 16, 88$)	91, 58($\pm 17, 05$)
diabetes(2)	58, 61($\pm 11, 21$)	58, 38($\pm 11, 31$)	60, 43($\pm 8, 83$)	59, 35($\pm 11, 51$)	58, 14($\pm 11, 44$)
diabetes(4)	59, 09($\pm 9, 88$)	58, 02($\pm 10, 14$)	60, 03($\pm 10, 02$)	58, 60($\pm 10, 58$)	57, 53($\pm 10, 36$)
diabetes(8)	66, 15($\pm 5, 70$)	65, 73($\pm 5, 10$)	62, 41($\pm 13, 10$)	60, 41($\pm 13, 02$)	62, 88($\pm 12, 63$)
diabetes(16)	68, 76($\pm 4, 81$)	67, 79($\pm 6, 31$)	63, 90($\pm 13, 45$)	61, 63($\pm 12, 31$)	65, 03($\pm 13, 64$)
ionos.(2)	55, 96($\pm 11, 46$)	55, 51($\pm 10, 72$)	48, 09($\pm 21, 33$)	47, 91($\pm 22, 05$)	48, 07($\pm 21, 88$)
ionos.(4)	65, 74($\pm 14, 88$)	65, 40($\pm 15, 53$)	59, 70($\pm 11, 03$)	63, 95($\pm 12, 40$)	65, 88($\pm 16, 11$)
ionos.(8)	72, 74($\pm 8, 33$)	71, 34($\pm 9, 76$)	60, 10($\pm 19, 42$)	67, 08($\pm 21, 45$)	66, 75($\pm 20, 40$)
ionos.(16)	75, 78($\pm 15, 01$)	74, 10($\pm 15, 33$)	68, 11($\pm 16, 77$)	74, 98($\pm 16, 51$)	73, 14($\pm 15, 28$)
anneal(2)	71, 51($\pm 9, 97$)	69, 97($\pm 10, 34$)	71, 15($\pm 10, 97$)	68, 49($\pm 11, 98$)	69, 89($\pm 10, 01$)
anneal(4)	76, 40($\pm 4, 25$)	75, 10($\pm 4, 33$)	75, 92($\pm 5, 59$)	74, 32($\pm 7, 81$)	74, 06($\pm 4, 68$)
anneal(8)	78, 24($\pm 6, 03$)	76, 89($\pm 6, 25$)	79, 23($\pm 5, 38$)	76, 11($\pm 7, 38$)	75, 66($\pm 6, 64$)
anneal(16)	84, 06($\pm 3, 88$)	81, 80($\pm 5, 60$)	82, 89($\pm 3, 75$)	80, 99($\pm 4, 72$)	80, 21($\pm 6, 66$)
remote(2)	68, 65($\pm 14, 41$)	68, 73($\pm 14, 46$)	73, 21($\pm 13, 37$)	79, 73($\pm 15, 25$)	67, 89($\pm 14, 04$)
remote(4)	80, 94($\pm 10, 64$)	81, 07($\pm 10, 15$)	86, 95($\pm 11, 07$)	93, 90($\pm 7, 69$)	80, 04($\pm 9, 25$)
remote(8)	94, 97($\pm 6, 54$)	94, 87($\pm 6, 45$)	93, 47($\pm 7, 30$)	98, 30($\pm 2, 42$)	93, 96($\pm 7, 23$)
remote(16)	98, 22($\pm 2, 83$)	98, 29($\pm 2, 48$)	99, 48($\pm 0, 94$)	98, 74($\pm 2, 41$)	97, 85($\pm 2, 78$)

Tableau B.3
Précision des résultats - partie II

B.2. RÉSULTATS COMPARATIFS

	CK	Simple+	Low+	Medium+	High+
iris(2)	65, 31($\pm 14, 01$)	68, 01($\pm 15, 49$)	68, 11($\pm 15, 43$)	69, 08($\pm 15, 38$)	68, 40($\pm 15, 53$)
iris(4)	74, 44($\pm 14, 57$)	76, 19($\pm 13, 63$)	75, 71($\pm 13, 51$)	75, 82($\pm 13, 57$)	75, 82($\pm 13, 54$)
iris(8)	88, 69($\pm 8, 32$)	93, 98($\pm 3, 43$)	93, 13($\pm 3, 74$)	92, 35($\pm 3, 96$)	92, 48($\pm 3, 54$)
iris(16)	92, 59($\pm 6, 48$)	94, 32($\pm 4, 56$)	92, 84($\pm 4, 27$)	91, 11($\pm 4, 83$)	92, 34($\pm 4, 58$)
wine(2)	54, 13($\pm 13, 27$)	54, 90($\pm 14, 32$)	54, 98($\pm 14, 30$)	55, 18($\pm 14, 34$)	54, 90($\pm 14, 28$)
wine(4)	82, 20($\pm 13, 13$)	78, 31($\pm 12, 04$)	78, 44($\pm 12, 03$)	79, 35($\pm 11, 86$)	78, 78($\pm 11, 87$)
wine(8)	89, 07($\pm 8, 52$)	89, 69($\pm 8, 34$)	90, 30($\pm 7, 84$)	90, 76($\pm 7, 86$)	90, 05($\pm 7, 83$)
wine(16)	94, 30($\pm 4, 04$)	95, 61($\pm 3, 69$)	95, 36($\pm 3, 90$)	95, 28($\pm 4, 17$)	95, 20($\pm 4, 05$)
breast-w(2)	71, 87($\pm 28, 44$)	75, 01($\pm 28, 58$)	75, 04($\pm 28, 59$)	75, 51($\pm 28, 60$)	75, 03($\pm 28, 58$)
breast-w(4)	84, 09($\pm 24, 95$)	83, 86($\pm 24, 75$)	83, 92($\pm 24, 74$)	84, 42($\pm 24, 67$)	83, 95($\pm 24, 74$)
breast-w(8)	90, 76($\pm 16, 97$)	91, 22($\pm 17, 02$)	91, 17($\pm 17, 01$)	91, 26($\pm 17, 02$)	91, 11($\pm 17, 00$)
breast-w(16)	91, 50($\pm 17, 04$)	91, 66($\pm 17, 04$)	91, 58($\pm 17, 03$)	91, 57($\pm 17, 03$)	91, 63($\pm 17, 04$)
diabetes(2)	58, 00($\pm 11, 56$)	59, 19($\pm 10, 47$)	59, 22($\pm 10, 46$)	59, 31($\pm 10, 36$)	59, 19($\pm 10, 49$)
diabetes(4)	57, 35($\pm 10, 38$)	61, 30($\pm 7, 84$)	61, 30($\pm 7, 80$)	61, 41($\pm 8, 08$)	61, 37($\pm 7, 93$)
diabetes(8)	62, 07($\pm 12, 42$)	64, 25($\pm 12, 68$)	64, 23($\pm 12, 73$)	63, 97($\pm 12, 77$)	64, 07($\pm 12, 80$)
diabetes(16)	64, 08($\pm 13, 52$)	67, 50($\pm 13, 13$)	67, 37($\pm 13, 13$)	66, 79($\pm 13, 00$)	67, 33($\pm 13, 10$)
ionos.(2)	47, 87($\pm 21, 70$)	48, 44($\pm 21, 29$)	48, 44($\pm 21, 31$)	48, 63($\pm 21, 45$)	48, 49($\pm 21, 33$)
ionos.(4)	65, 86($\pm 16, 38$)	68, 32($\pm 9, 88$)	68, 36($\pm 9, 94$)	69, 16($\pm 9, 94$)	68, 40($\pm 9, 94$)
ionos.(8)	66, 52($\pm 20, 43$)	75, 24($\pm 21, 30$)	75, 34($\pm 21, 22$)	75, 13($\pm 21, 17$)	75, 26($\pm 21, 23$)
ionos.(16)	72, 49($\pm 15, 25$)	82, 68($\pm 16, 14$)	82, 54($\pm 16, 04$)	80, 76($\pm 15, 62$)	82, 35($\pm 15, 96$)
anneal(2)	69, 60($\pm 10, 21$)	76, 03($\pm 3, 96$)	75, 91($\pm 4, 00$)	75, 80($\pm 3, 86$)	75, 64($\pm 3, 78$)
anneal(4)	73, 13($\pm 5, 49$)	78, 12($\pm 3, 35$)	78, 07($\pm 3, 41$)	77, 78($\pm 3, 39$)	77, 04($\pm 3, 82$)
anneal(8)	74, 84($\pm 6, 61$)	81, 54($\pm 3, 92$)	81, 30($\pm 4, 16$)	80, 84($\pm 4, 13$)	79, 44($\pm 5, 03$)
anneal(16)	79, 14($\pm 7, 09$)	87, 17($\pm 3, 26$)	86, 89($\pm 3, 38$)	85, 84($\pm 3, 27$)	83, 80($\pm 4, 48$)
remote(2)	67, 39($\pm 13, 91$)	65, 40($\pm 13, 46$)	65, 40($\pm 13, 46$)	65, 47($\pm 13, 32$)	65, 44($\pm 13, 39$)
remote(4)	79, 54($\pm 8, 88$)	79, 58($\pm 11, 10$)	79, 67($\pm 11, 16$)	79, 67($\pm 11, 16$)	79, 67($\pm 11, 16$)
remote(8)	93, 43($\pm 7, 51$)	94, 15($\pm 7, 05$)	94, 15($\pm 7, 04$)	94, 30($\pm 7, 08$)	94, 20($\pm 7, 01$)
remote(16)	97, 77($\pm 3, 09$)	98, 81($\pm 2, 20$)	98, 81($\pm 2, 12$)	98, 81($\pm 2, 12$)	98, 88($\pm 2, 12$)

Tableau B.4
Précision des résultats - partie III

ANNEXE B. DONNÉES ET RÉSULTATS POUR LA MÉTHODE SLEM

	Simple	Low	Medium	High
iris(2)	78, 11($\pm 14, 73$)	80, 29($\pm 12, 99$)	85, 70($\pm 8, 03$)	82, 31($\pm 11, 33$)
iris(4)	83, 17($\pm 16, 13$)	84, 97($\pm 14, 41$)	86, 24($\pm 9, 46$)	81, 69($\pm 12, 19$)
iris(8)	88, 95($\pm 8, 31$)	89, 34($\pm 5, 42$)	90, 06($\pm 4, 17$)	85, 88($\pm 10, 43$)
iris(16)	90, 74($\pm 5, 12$)	88, 76($\pm 5, 36$)	87, 90($\pm 5, 14$)	86, 91($\pm 7, 01$)
wine(2)	84, 41($\pm 14, 92$)	84, 90($\pm 14, 70$)	91, 88($\pm 9, 84$)	84, 05($\pm 13, 73$)
wine(4)	93, 85($\pm 5, 84$)	94, 97($\pm 2, 16$)	95, 02($\pm 2, 09$)	88, 13($\pm 11, 21$)
wine(8)	95, 12($\pm 3, 25$)	95, 07($\pm 2, 84$)	94, 66($\pm 3, 48$)	91, 69($\pm 6, 20$)
wine(16)	95, 77($\pm 3, 66$)	95, 36($\pm 3, 52$)	95, 61($\pm 3, 52$)	91, 38($\pm 8, 00$)
breast-w(2)	73, 84($\pm 31, 14$)	82, 87($\pm 28, 80$)	84, 47($\pm 28, 17$)	79, 89($\pm 30, 05$)
breast-w(4)	86, 79($\pm 24, 45$)	87, 58($\pm 23, 43$)	88, 10($\pm 23, 57$)	86, 56($\pm 24, 48$)
breast-w(8)	91, 50($\pm 17, 02$)	91, 24($\pm 16, 98$)	91, 14($\pm 16, 96$)	89, 69($\pm 18, 65$)
breast-w(16)	91, 08($\pm 16, 93$)	90, 85($\pm 16, 88$)	90, 84($\pm 16, 88$)	87, 84($\pm 19, 44$)
diabetes(2)	60, 16($\pm 10, 44$)	60, 43($\pm 10, 58$)	59, 31($\pm 11, 15$)	59, 49($\pm 10, 81$)
diabetes(4)	60, 74($\pm 9, 73$)	59, 43($\pm 11, 30$)	59, 32($\pm 11, 30$)	59, 05($\pm 12, 00$)
diabetes(8)	61, 22($\pm 13, 06$)	61, 69($\pm 13, 07$)	60, 38($\pm 14, 04$)	61, 42($\pm 13, 04$)
diabetes(16)	62, 41($\pm 11, 73$)	63, 12($\pm 12, 20$)	62, 79($\pm 12, 25$)	62, 73($\pm 12, 16$)
ionos.(2)	51, 54($\pm 24, 04$)	56, 55($\pm 25, 86$)	61, 69($\pm 26, 44$)	57, 56($\pm 26, 34$)
ionos.(4)	72, 57($\pm 9, 77$)	72, 69($\pm 8, 26$)	73, 75($\pm 9, 18$)	70, 39($\pm 9, 57$)
ionos.(8)	67, 73($\pm 20, 71$)	68, 70($\pm 19, 75$)	71, 53($\pm 19, 42$)	67, 61($\pm 20, 00$)
ionos.(16)	74, 07($\pm 14, 92$)	71, 84($\pm 13, 70$)	72, 37($\pm 13, 64$)	69, 86($\pm 15, 17$)
anneal(2)	76, 51($\pm 4, 17$)	74, 65($\pm 6, 08$)	69, 10($\pm 8, 30$)	60, 49($\pm 8, 47$)
anneal(4)	76, 80($\pm 4, 11$)	75, 59($\pm 5, 57$)	71, 55($\pm 7, 85$)	60, 82($\pm 12, 52$)
anneal(8)	78, 47($\pm 4, 10$)	77, 51($\pm 5, 29$)	75, 13($\pm 5, 56$)	63, 49($\pm 8, 20$)
anneal(16)	80, 20($\pm 3, 44$)	79, 66($\pm 4, 01$)	75, 55($\pm 6, 27$)	62, 84($\pm 9, 66$)
remote(2)	83, 60($\pm 14, 66$)	86, 89($\pm 10, 80$)	93, 06($\pm 8, 31$)	88, 39($\pm 12, 85$)
remote(4)	88, 47($\pm 7, 63$)	93, 04($\pm 8, 06$)	95, 10($\pm 5, 01$)	94, 81($\pm 5, 74$)
remote(8)	93, 52($\pm 4, 32$)	96, 03($\pm 4, 14$)	95, 89($\pm 2, 24$)	93, 91($\pm 5, 79$)
remote(16)	94, 07($\pm 4, 88$)	96, 88($\pm 2, 33$)	96, 88($\pm 2, 19$)	97, 11($\pm 2, 51$)

Tableau B.5
Précision des résultats - partie IV

B.2. RÉSULTATS COMPARATIFS

données	1	2	3	Données enrichies	Données d'origine	\nearrow
iris(2)	85.70(Medium)	82.320(High)	78,02(1-NN)	85.70(Medium)	78,02(1-NN)	+
iris(4)	88.84(DL)	86.24(Medium)	85,61(1-NN)	88.84(DL)	85,61(1-NN)	+
iris(8)	93.99(Simple+)	93,59(NB)	93,59(DL)	93.99(Simple+)	93,59(NB)	+
iris(16)	94.32(Simple+)	94.20(DL)	94.20(NB)	94.32(Simple+)	94.20(NB)	+
wine(2)	91.89(Medium)	84.90(Low)	84.42(Simple)	91.89(Medium)	76.67(1-NN)	+
wine(4)	95.02(Medium)	94.98(Low)	93.85(Simple)	95.022(Medium)	88.92(1-NN)	+
wine(8)	95.13(Simple)	95.08(Low)	94.77(DL)	95.13(Simple)	91.95(1-NN)	+
wine(16)	96.10(DL)	95.77(Simple)	95.61(Medium)	96.10(DL)	95.53(NB)	+
breast-w(2)	85.23(SL)	84.72(DL)	84.47(Medium)	85.23(SL)	82.40(NB)	+
breast-w(4)	89.91(DL)	88.10(Medium)	87.59(Low)	89.91(DL)	87.09(1-NN)	+
breast-w(8)	91.50(Simple)	91.26(Medium+)	91.24(Low)	91.50(Simple)	91.08 (NB)	+
breast-w(16)	94.75(DL)	94.68(RC)	91.66(Simple+)	94.75(DL)	94.69 (NB)	+
diabetes(2)	60.62(1-NN)	60.46(DL)	60.44(SCEC)	60.46(DL)	60.62(1-NN)	-
diabetes(4)	64.94(DL)	62.93(1-NN)	62.24(C45)	64.94(DL)	62.93(1-NN)	+
diabetes(8)	67.67(DL)	66.24(NB)	66.16(CLIM)	67.67(DL)	66.24(NB)	+
diabetes(16)	69.61(DL)	69.56(NB)	69.14(SL)	69.61(DL)	69.56(NB)	+
ionos.(2)	61.70(Medium)	59.71(DL)	58.59(SL)	61.70(Medium)	57.37(1-NN)	+
ionos.(4)	73.75(Medium)	72.69(Low)	72.67(Simple)	73.75(Medium)	70.08(1-NN)	+
ionos.(8)	79.94(NB)	79.24(SL)	77.21(C45)	79.24(SL)	79.94(NB)	-
ionos.(16)	82.70(NB)	82.68(Simple+)	82.54(Low+)	82.68(Simple+)	82.70(NB)	-
anneal(2)	76.71(1-NN)	76.51(Simple)	76.03(Simple+)	76.51(Simple)	76.71(1-NN)	-
anneal(4)	79.75(1-NN)	78.51(C45)	78.13(NB)	78.12(Simple+)	78.13(NB)	-
anneal(8)	85.19(1-NN)	82.59(C45)	81.55(Simple+)	81.55(Simple+)	85.19(1-NN)	-
anneal(16)	90.92(1-NN)	89.85(C45)	87.27(NB)	87.18(Simple+)	90.92(1-NN)	-
remote(2)	93.06(Medium)	88.39(High)	86.90(Low)	93.06(Medium)	85.52(1-NN)	+
remote(4)	95.68(DL)	95.60(1-NN)	95.10(Medium)	95.68(DL)	95.60(1-NN)	+
remote(8)	99.27(DL)	98.99(1-NN)	98.41(SL)	99.27(DL)	98.99(1-NN)	-
remote(16)	99.70(1-NN)	99.48(SCEC)	99.41(DL)	99.41(DL)	99.70(1-NN)	-

Tableau B.6

Les trois meilleurs résultats pour les expériences sur les différents jeux de données.

C

Articles principaux

Dans cette annexe sont insérés les trois articles principaux décrivant les contributions les plus importantes présentées au titre de mon habilitation à diriger des recherches.

- [1] *Knowledge-based region labeling for remote sensing image interpretation*
G. Forestier, A. Puissant, **C. Wemmert**, P. Gançarski
Computers, Environment and Urban Systems, Elsevier, 2012 - à paraître
- [2] *Collaborative clustering with background knowledge*
G. Forestier, P. Gançarski, **C. Wemmert**
Data & Knowledge Engineering, Vol. 69, Num. 2, Elsevier, pp 211–228 - 2010
- [3] *Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation*
S. Derivaux, G. Forestier, **C. Wemmert**, S. Lefèvre
Pattern Recognition Letters, Vol. 31, Num. 15, Elsevier, pp 2364–2374 - 2010



Contents lists available at SciVerse ScienceDirect

Computers, Environment and Urban Systems

journal homepage: www.elsevier.com/locate/compenvurbssys

Knowledge-based region labeling for remote sensing image interpretation

G. Forestier^a, A. Puissant^b, C. Wemmert^{a,*}, P. Gançarski^a^aLSIIT – Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection UMR 7005 CNRS – Uds, University of Strasbourg, France^bLIVE – Laboratoire Image, Ville, Environnement ERL 7230 CNRS – Uds, University of Strasbourg, France

ARTICLE INFO

Article history:

Received 15 October 2010

Received in revised form 18 January 2012

Accepted 19 January 2012

Available online xxxx

Keywords:

Urban object

Knowledge base

High resolution

Remote sensing images

Semantic interpretation

Region labeling

ABSTRACT

The increasing availability of High Spatial Resolution (HSR) satellite images is an opportunity to characterize and identify urban objects. Thus, the augmentation of the precision led to a need of new image analysis methods using region-based (or object-based) approaches. In this field, an important challenge is the use of domain knowledge for automatic urban objects identification, and a major issue is the formalization and exploitation of this knowledge. In this paper, we present the building steps of a knowledge-base of urban objects allowing to perform the interpretation of HSR images in order to help urban planners to automatically map the territory. The knowledge-base is used to assign segmented regions (*i.e.* extracted from the images) into semantic objects (*i.e.* concepts of the knowledge-base). A matching process between the regions and the concepts of the knowledge-base is proposed, allowing to bridge the semantic gap between the images content and the interpretation. The method is validated on Quickbird images of the urban areas of Strasbourg and Marseille (France). The results highlight the capacity of the method to automatically identify urban objects using the domain knowledge.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Urban planners are interested in up-to-date land cover and land use information on urban objects at several spatial (1:100,000–1:5000) and temporal scales. Acquiring automatically this information is complex, difficult and time-consuming if traditional data sources (e.g. ground survey techniques) are used. The increasing availability of remotely sensed images with Medium Spatial Resolution (MSR) from 30 to 10 m or High Spatial Resolution (HSR) from 5 to 1 m is an opportunity to characterize and identify these objects into urban and peri-urban areas (Wu et al., 2009). Images can be exploited to provide this spatial information, which can also be easily integrated in urban GIS platforms.

Image interpretation is a difficult task and can be defined as the extraction of the image semantic. It consists in obtaining useful spatial and thematic information on the objects by using human knowledge and experience (Lillesand et al., 2003; Moller-Jensen, 1997). In this domain, differences are observed between the visual interpretation of the spectral information and the semantic interpretation of the pixels, mainly due to different levels of abstraction. The semantic is not always explicitly contained in the image and depends on domain knowledge and on the context. This problem is known as the *semantic gap* (Smeulders et al., 2000) and is defined as the lack of concordance between low-level information (*i.e.* automatically extracted from the images) and high-level information

(*i.e.* analyzed by urban experts). In order to reduce the semantic gap, image analysis methods using *region-based* (or *object-based*) approaches with domain knowledge are developed (Benz et al., 2004; Herold et al., 2002). These methods involve the segmentation of the images into homogeneous regions and the characterization of the regions with a set of spectral (e.g. spectral signature, spectral index), spatial (e.g. shape index) and topological (e.g. adjacency, inclusion) features. Region-based classification is known to achieve better results than pixel-based classification (Cleve et al., 2008) for processing HSR images. However, only few initiatives have focused on the use of domain knowledge for classifying urban objects (Baltasavias, 2004), and a major issue in these approaches is therefore domain knowledge formalization and exploitation. Building a knowledge-base is a difficult task because the knowledge is most of the time implicit and held by the domain experts.

The aim of this paper is to highlight the benefits of using a knowledge-base (KB) for automatic regions labeling in order to store expert knowledge and to use it to automate image interpretation. The contribution of this paper is twofold. First, we present the building steps of a knowledge-base adapted to the interpretation of HSR images. A key issue is to identify appropriate concepts in terms of external structure (*i.e.* a hierarchy) and, in terms of internal definition (*i.e.* the attributes and their domain values) to describe the thematic objects for mapping the territory. In particular, we describe an attributes-filling mechanism used to feed the knowledge-base. The second contribution lies in the validation of a matching method which uses the knowledge-base for automatic image interpretation. The purpose of this method is to label regions

* Corresponding author. Tel.: +33 03 68 85 45 81; fax: +33 03 68 85 44 55.

E-mail address: wemmert@unistra.fr (C. Wemmert).

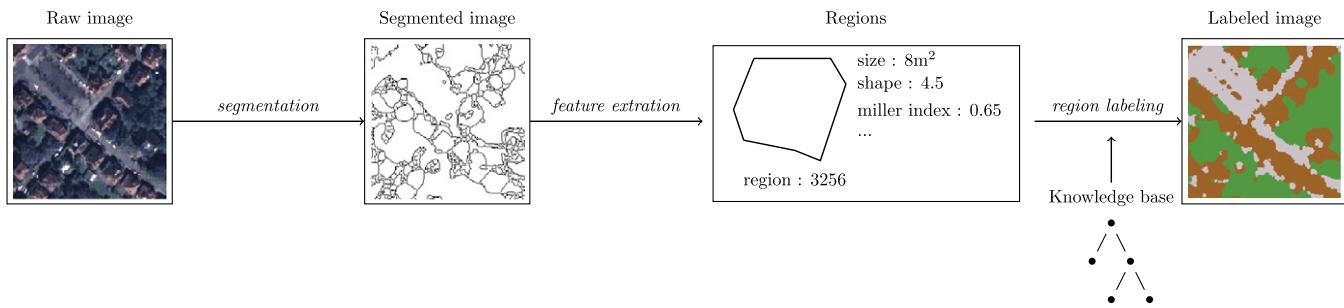


Fig. 1. The region labeling workflow: from a raw image to a labeled image.

extracted from remotely sensed images. The method starts by associating a set of low-level characteristics to each region built using a segmentation algorithm. Then, the knowledge-base is used to assign a semantic to the regions. Fig. 1 illustrates the different steps of the approach. We also present experimental results to highlight the relevance of our method on multiple HSR images.

The paper is organized in six sections. First, approaches using domain knowledge in image analysis are discussed (Section 2). Second, the steps to build the knowledge-base adapted to image interpretation is presented (Section 3). Third, the knowledge based region labeling process is detailed (Section 4). Then, some experiments on Quickbird (Digital Globe[®]) images with a spatial resolution of 0.61 m, on the urban areas of Strasbourg and Marseille (France), are proposed (Section 5). Finally, we conclude and present some perspectives (Section 6).

2. Knowledge-based systems for image analysis

Knowledge-based systems (KBS) are becoming more and more important in various domains despite the fact that they are still complex to produce (Gomez & Segami, 2007). Indeed, acquiring and representing the knowledge of a domain is often a tedious process and the multiple steps involved in the creation of the knowledge-base can be very different according to the studied domain. This heterogeneity led to an abundance of propositions and the expert is often lost when the time comes to choose a solution. However, the advantages of representing and storing domain knowledge are undeniable. Indeed, it is then possible to produce intelligent systems based on the use of the acquired knowledge and to better explain and understand the domain under consideration.

Knowledge-based systems have proved to be effective for complex object recognition and for image analysis. For instance, the Sigma (Matsuyama & Hwang, 1990) and Schema (Draper, Collins, Brolio, Hanson, & Riseman, 1989) systems performed image analysis on aerial images by using several descriptors of the objects. These systems give access to a high semantic level but are strongly domain-dependent as they integrate prior knowledge on the image (Crevier & Lepage, 1997). Their main drawback is that the knowledge is not clearly separated from the procedure. Alternatively, Cataldo and Rinaldi (2010) proposed a model of knowledge in the framework of landscape planning, with a particular emphasis on cultural landscape, to resolve conceptual misunderstandings and semantic ambiguities, and to provide a precise and accurate description of the current state of the knowledge. In the domain of image segmentation and object labeling, there exist some previous work trying to benefit from a representation of expert knowledge on the objects to extract and label. For example, Athanasiadis, Mylonas, Avrithis, and Kollias (2007) proposed a new framework for automatic image annotation, guided by expert knowledge represented by an ontological knowledge base. A region growing segmentation algorithm is driven by new similarity measures and merging criteria defined at a semantic level. In Castellano, Fanelli,

and Torsello (2011), the authors presented a fuzzy shape annotation approach for automatic image labeling. The method is based on a fuzzy clustering algorithm, partially supervised by information on the shape of the object and textual labels related to semantic categories. In the remote sensing field, the Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung¹ made a lot of effort since many years for incorporating *a priori* knowledge into the image interpretation process (Tiijnes, Glowe, Bücknel, & Liedtke, 1999; Bückner, Pahl, Stahlnut, & Liedtke, 2002; Bückner et al., 2002). Their GeoAIDA system uses a semantic net to model *a priori* knowledge on the objects in the studied scene. A multi-level semantic segmentation is proposed, built by the collaboration of multiple segmentation algorithms controlled by external operators evaluating the interpretation hypothesis made by the different methods.

A classical way to build a knowledge-base is to use an ontology. An ontology can be defined as a simplified view of the world, which is represented for specific purpose (Gruber, 1995). It defines a set of representational terms called concepts, their characteristics and their relationships. It is the result of a consensus in a user community to clarify the communication. An ontology can have a different representation according to its level of expressivity. It can simply be composed of a taxonomy but can also carry complex axioms about the domain concepts. Depending on the building process, an ontology can be generic or domain-dependent. Therefore, recent works have proposed to use ontologies to describe more clearly the knowledge of the studied domain. In Zlatoff et al. (2004), spatial relations between concepts are used to merge regions and to recognize objects. The exclusive use of spatial relations is however not possible in the case of remotely sensed images. This work points out the differences between domain knowledge and procedures. In a same way, Maillot and Thonnat (2008) proposed an ontology-based object learning and recognition system for image analysis. An interesting point is the separation of a local matching and a global matching procedure (*i.e.* the global matching combines the probabilities computed during the local matching). The descriptors used for the matching correspond to *visual concepts* which are acquired during the learning phase. The matching function is then dependent of these visual concepts. The authors state that the global matching should take into account the hierarchy of the ontology. Although, this kind of system needs a time consuming learning step, and also requires the expert to produce examples for each of the concept he is looking for.

Many other works on image analysis tried to benefit from building an ontology. In Dasiopoulou et al. (2005), an ontology-based object detection using a segmentation process for video analysis is proposed. Breen et al. (2002) used a neural network method to classify objects in pre-defined classes. Both systems determine if the image may be classified by a concept from an ontology. In Panagi et al. (2006), the authors proposed a genetic algorithm of

¹ TNT, University of Hannover, Germany.

Table 1

Extract of the Corine Land Cover Nomenclature used to map urban area.

1:100,000 Level 1	1:100,000 Level 2	1:50,000 Level 3
1. Artificial surfaces		
1.1. Urban fabric	1.1.1. Continuous urban fabric	
	1.1.2. Discontinuous urban fabric	
1.2. Industrial, commercial and transport units	1.2.1. Industrial or commercial unit	
	1.2.2. Road and rail networks	
	1.2.3. Port areas	
	1.2.4. Airports	
1.3. Mine, dump and construction sites	1.3.1. Mineral extraction sites	
	...	
1.4. Artificial, non agricultural vegetated areas	1.4.1. Green urban areas	
2. Agricultural areas		
3. Forest and semi-natural areas		
4. Wetlands		
5. Waterbodies		
5.1. Inland waters	5.1.1. Water courses	
	5.1.2. Water bodies	
5.2. Marine waters	...	

ontology-driven semantic image analysis. Some low-level descriptors are extracted from the image and are used to match with the ontology. A set of hypothesis (*i.e.* a list of possible concepts and their degrees of confidence) are then tested with a genetic algorithm to determine the optimal image interpretation. Only spatial relations (eight directional relations) are used by the system. In Athanasiadis et al. (2007), the authors present a framework for simultaneous image segmentation and object labeling using an ontology in the domain of multimedia analysis.

In the field of remote sensing several propositions involving the construction of an ontology exist. For example, Fonseca et al. (2002) presented a reflexion about the construction and the use of ontologies at different levels of Geographic Information System (GIS). They proposed an ontology-driven GIS that acts as a system integrator. In this system, an ontology is a component, such as the database, cooperating to fulfill the system's objectives. In another initiative, Uitermark et al. (1999) proposed a framework for ontology-based geographic data set integration, an ontology being a collection of shared concepts. Components of this formal approach are an ontology for topographic mapping (*i.e.* a domain ontology), an ontology for every geographic data sets involved



Fig. 2. Single and aggregate objects from Quickbird image. (a) Single objects, each corresponding to one group of pixels (houses), (b) Aggregate object composed of some groups of homogeneous pixels (houses, gardens, road).

(*i.e.* the application ontologies), and abstraction rules (*i.e.* capture criteria). It is common in GIS to use multiple ontologies to represent different levels of knowledge. The main advantage is to efficiently separate the different kind of knowledge but it leads to complex systems which are difficult to understand as a whole.

Although these work using ontologies are interesting, they rarely tackle the problem of actually identify the concepts present in the created ontology. Indeed, they often describe in details meta-data about the representation, the hierarchy of concepts but often omit an important part: does the modeled knowledge can be used in remote sensing image interpretation? Our goal in this paper is to propose an actionable representation of the knowledge for image interpretation. In the following sections we present the different steps of the construction and the use of our knowledge-base.

3. Construction of the knowledge-base

The use of domain-dependent knowledge-base (KB) for object analysis from HSR images presents two main challenges: the first is the extraction of the semantic (or thematic) concepts adapted to HSR images and the second is the actual construction of the KB. There are no standard type of KB available for all the domains of application (Noy et al., 2000; Waterson and Preece, 1999). In agreement with Uschold and King (1995), we used a 3-steps methodology to construct our KB. We started by identifying the concepts needed for mapping the urban territory from HSR images. In Section 3.1, we detail the gap between these concepts and their identification in HSR images. Then, we describe in Section 3.2 the

Table 2

Extract of the taxonomy added to map urban area on MSR and HSR images.

1:25,000 Level 4: Area level	1:10,000 Level 5: Block level	1:5000 Level 6: Urban object level
• High-density urban fabric	• Continuous urban blocks	• Building/roofs: orange tile roof, ... light gray residential roof
• Low-density urban fabric	• Discontinuous urban blocks	• Vegetation: green vegetation, non-photosynthetic veg, ...
• Industrial areas	– Individual urban blocks	• Transportation: street, parking lots, ...
• Forest zones	– Collective urban blocks	• Water surfaces: river, natural water bodies, ...
• Agricultural zones	• Industrial urban blocks	• Bare soil
• Water surfaces	• Urban vegetation	• Shadow
• Bare soil	• Forest	
	• Agricultural zones	
	• Water surfaces	
	• Road	

urban objects identifiable in such images. Finally, we present in Section 3.3 an implementation of the KB in a computer-readable form.

3.1. Step1: Identification of the concepts

A lot of land cover/land use terms exist, which represents the linguistic expression of the urban scene knowledge. Nevertheless, several terms correspond to urban objects which are not always identifiable on the images depending of their spatial resolution. In fact, there is a wide range of object nomenclatures for remotely sensed data such as the CORINE LAND COVER nomenclature defined for LANDSAT images (30 m spatial resolution), the SPOT Thema nomenclature defined for SPOT images (5–20 m) or the French national land-cover database BDCARTO IGN® (defined for aerial photographs and SPOT images). All these nomenclatures built from MSR images are adapted to map urban areas from 1:100,000 to 1:50,000 (Table 1). A fourth level is commonly added by users to map urban area with a scale of 1:25,000 allowing for instance to specify the density of an urban fabric (Autran, 2007) (Table 2, left column). Nowadays, it is possible to extract urban objects (e.g. house, garden and road) from HSR images. This allows to map individual objects with their material (e.g. houses with orange tile roof) corresponding to a scale near of 1:5000 (Table 2, right column).

In the domain of urban planning and management, some users also need to map the territory at the scale of the urban blocks (*i.e.* which can be defined as a minimal cycle closed by communication way) corresponding to a scale near of 1:10,000. In this case, there is no existing available land cover/land use product. The MSR images have a too coarse spatial resolution and HSR images have a too fine spatial resolution to map urban blocks. Thus, it is necessary to add an intermediate level (Table 2, middle column).

3.2. Step 2: Formalization of the concepts

These new urban concepts based on HSR images have to be translated into objects directly identifiable on images at this specific spatial resolution. These objects are called single object if one group of homogeneous pixels (referred here as region) is sufficient to identify one of them. For example, at a metric resolution, an object can correspond to a house, level 6 (Fig. 2a). Alternatively, it is called an aggregate object if several groups of homogeneous regions are necessary to identify it. For example, at a metric resolution, an aggregate object can correspond to an individual urban block, level 5, composed of houses, gardens, streets, etc. (Fig. 2b).

In this work, we described these objects using a dictionary, adapted from Pantazis et al. (2002), which contains three categories of information:

1. Some characteristics to identify the objects: name, representation in a GIS database (e.g. point, polyline or polygon), type of object (e.g. single, aggregate), range of spatial resolution at which the object is identifiable;
2. A qualitative description of the object (e.g. textual definition);
3. A list of relevant indicators used in the photo-interpretation domain to characterize these objects, classed by their relative importance respectively: color, shape, texture (Momm et al., 2009), context or spatial relationships (Table 3, left column)

These qualitative descriptors have to be translated into quantitative low descriptors according to the analyzed image. For example, the low-level descriptors associated to the color information depend on the radiometric reflectance of the objects and on some indexes calculated from this reflectance. The Table 3 presents the low-descriptors we used in this work.

Table 3

The descriptor classes and the low-level descriptors identified to characterize the regions.

Class of descriptor	Numbers of low level desc.	Low-level descriptors
Color	4	Reflectance: range of observed values in four spectral bands: – Blue (B) – Green (G) – Red (R) – Near-InfraRed (NIR)
	2	Spectral index: range of observed values of – Normalized Difference Vegetation Index (NDVI) – Soil Brightness Index (BI)
Shape	5	Range of observed values of area, perimeter, elongation,
	2	Range of observed values of the homogeneity index and the variance derived from the co-occurrence gray-levelmatrix (Haralick, 1979)
Context	4	Relationships: adjacency, inclusion, composition, neighborhood

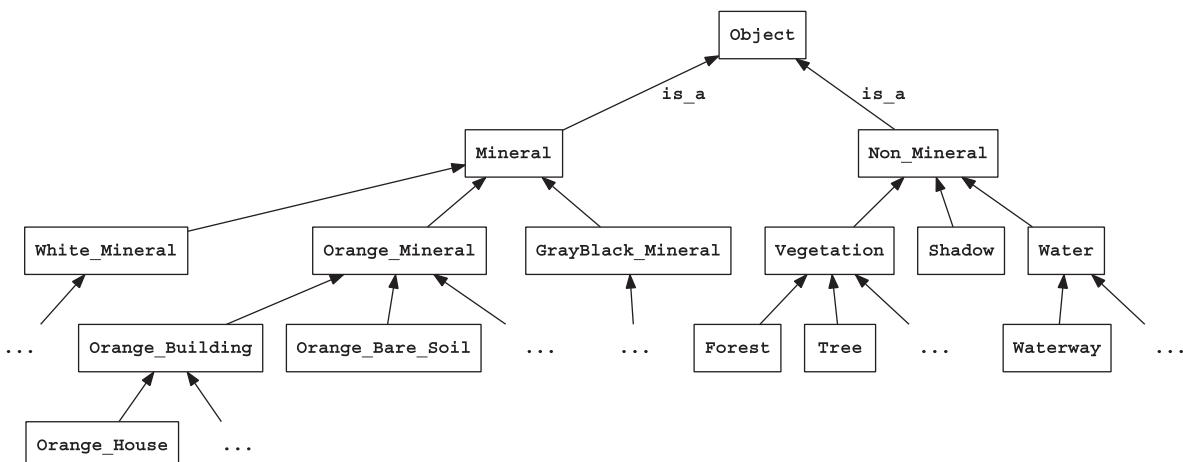


Fig. 3. Excerpt of the hierarchy of concepts.

Each object type is associated to an urban concept and each qualitative descriptor is associated with a low-level quantitative attribute. We focused our work on identify single object as it is the first step before considering trying to identify aggregate objects. We present in the following the implementation of the knowledge-base.

3.3. Step 3: Implementation of the knowledge-base

Experiments carried out at the 6th level shown that it is very difficult on the one hand, to define exactly the right range of accepted values for each attribute for each concept, and on the other hand, to extract objects from the HSR images. In fact, some concepts can be difficult to discriminate. For example, it is difficult to discriminate between house with orange tile roof and orange bare soil from tennis court or between water and shadow. The ability to discriminate forest and tree depends for example of the quality of the image segmentation. To address these problems, we built a KB which allows to generalize the urban concepts that are difficult to discriminate. To build the hierarchy of concepts, we used the order of importance of the descriptors. The KB we created (Fig. 3) corresponds to the 6th level and is composed of 91 concepts. Each concept has a label (e.g. Orange_House for individual houses with orange roof tiles) and is defined by attributes corresponding to the low-level descriptors. To precisely describe how the hierarchy of concepts is built and used, let us introduce some notations and definitions (Durand et al., 2007).

Definition 1 (concept, sub-concept, depth). Let Θ be the set of concepts, \preceq_Θ is a partial order between concepts. $\forall(C_i, C_j) \in \Theta^2$, $C_i \preceq_\Theta C_j$ means that C_i is a sub-concept of C_j . $\rho(C)$ is the depth of the concept C in the hierarchy.

For example, $C_i = \text{Orange_House}$ is a sub-concept of $C_j = \text{Orange_Building}$. $\rho(C_i) = 5$ (see Fig. 3).

Definition 2 (specific attributes of a concept). Let $\mathcal{F}_\alpha(C)$ be the set of attributes of the classes in α , specifically associated with the concept $C \in \Theta$.

For instance, for the concept $C = \text{Orange_House}$, if the spectral attributes (spectral_signature_Blue,...) and their values are inherited by the Orange_Building , they are not present in $\mathcal{F}_\alpha(C)$. But an attribute overridden in C is present in $\mathcal{F}_\alpha(C)$.

Definition 3 (values and weight of an attribute). Let $a \in \mathcal{A}_\alpha$ be an attribute of a class in $\alpha \in \Phi$. We define $\mathcal{V}_C : \mathcal{A}_\alpha \rightarrow [\mathbb{R}; \mathbb{R}]$ so that $\mathcal{V}_C(a)$ is the range of values for ' a ' in the concept $C \in \Theta$. Let $\omega(a, C)$ be the weight associated to the attribute ' a ' for the concept C .

Definition 4 (set of regions). Let Γ be the set of regions.

Definition 5 (feature value of a region). Let $a \in \mathcal{A}$ be a feature of a (segmented) region $R \in \Gamma$. We define $\mathcal{V}'_R : \mathcal{A}_\alpha \rightarrow \mathbb{R}$ so that $\mathcal{V}'_R(a)$ is the value of ' a ' for the region R .

The conception phase of the KB consisted in defining Θ , \preceq_Θ , $\mathcal{F}_\alpha(C)$, $\omega(a, C)$, Φ and $\mathcal{V}_C(a)$. For all the concepts C , all the attribute values $\mathcal{V}_C(a)$ have to be provided by the expert or using learning algorithms. This allows to reduce the semantic gap between expert knowledge and image content. An example through the concept Orange_House , is described in Table 4. Note that in general, it is rather difficult to draw knowledge from domain experts. The experts are rarely able to directly supply an explicit description of the knowledge they use for objects identification. In addition, acquiring knowledge this way is usually time consuming. This is a well-known problem within the artificial intelligence commu-

nity. Thus, in order to ease the creation of the KB, we used machine learning techniques to automatically extract knowledge from the raw images. For example, to learn interpretable rules and build a reusable knowledge base, we used symbolic tools (Sheeren, Puis-sant, et al., 2006; Sheeren, Quirin et al., 2006). This step was very important for the discussion with the experts, and helped to create the geographical KB content. The proposed KB was developed using Protégé (Noy et al., 2000), a free open-source software that provides tools to construct domain models and knowledge-based applications.

4. Knowledge-based region labeling

The proposed method which associates each region of an image to a concept of the KB (i.e. to assign a semantic label to each region) is composed of two main steps: the construction of the regions (Section 4.1) and the matching of the regions with the KB to assign a semantic to each region (Section 4.2).

4.1. Regions building using a segmentation algorithm

A segmentation algorithm is applied on the image in order to obtain a set of regions. A region is a set of connected and spectrally homogeneous pixels. The regions are then characterized by assigning a set of low-level descriptors to each of them. A numerical value is calculated for each attribute. It is important to note that any segmentation method can be used. However, this step is a critical point of the global identification method. Indeed, the quality of the produced segmentation is very important and is strongly linked to the quality of the identification process. This point is discussed in further details in Section 5.1.

Algorithm 1. Traversing algorithm of the KB.

Input: a region R , a KB (Θ , Φ , $\mathcal{V}_C(a)$, ...), a set of attribute classes (α), maxDepth and minScore .

Output: the best label (s) and the matching score value.

$\text{depth} = 1$; $\text{scoreMax} = \text{minScore}$;

$\mathcal{L}_\alpha(R) = \emptyset$;

$\mathcal{RC} = \{\text{root}\}$; $\text{scoreDepth} = 0$; $\text{bestsDepth} = \emptyset$;

while ($\mathcal{RC} \neq \emptyset$ and $\text{depth} \leqslant \text{maxDepth}$) **do**

$\text{scoreDepth} = 0$; $\text{Best} = \emptyset$;

for all $C \in \mathcal{RC}$ **do**

$s = \text{Score}_\alpha(R, C)$;

if ($s == \text{scoreMax}$) **then**

$\mathcal{L}_\alpha(R) += \{C\}$;

end if

if ($s > \text{scoreMax}$) **then**

$\mathcal{L}_\alpha(R) = \{C\}$; $\text{scoreMax} = s$;

end if

if ($s == \text{scoreDepth}$) **then**

$\text{bestsDepth} += \{C\}$;

end if

if ($s > \text{scoreDepth}$) **then**

$\text{bestsDepth} = \{C\}$; $\text{scoreDepth} = s$;

end if

end for

$\mathcal{RC} = \emptyset$;

for all $C_j \in \text{bestsDepth}$ **do**

$\mathcal{RC} = \mathcal{RC} \cup \{C_i | C_i \preceq_\Theta C_j\}$;

end for

$\text{depth}++$;

end while

return $\{\mathcal{L}_\alpha(R), \text{score}\}$;

Table 4

Concept Orange_House.

Descriptor	Descriptor-associated attribute	Weight	Values	
			Min	Max
Color	Blue	1	21.7	62.3
	Green	1	19.4	80.1
	Red	1	29.7	135.1
	Near-InfraRed	1	34.8	139
	NDVI	1	50.2	108
	SBI	0.5	14.6	60.1
Shape	Diameter (m)	0.8	13	61
	Area (m ²)	1	10	600
	Perimeter (m)	1	28	116
	Elongation (m)	0.6	1	3.1
	Miller index	0.5	0.5	0.8
	Solidity index	1	0.85	1

4.2. Regions labeling using the knowledge base

The regions and their features are the inputs of the KB-based object recognition. The aim of this step is to find the concepts of the KB that best match the regions. To carry out this comparison, we defined a matching measure and a traversing method of the hierarchy of concepts.

Matching score. The proposed matching mechanism is a feature-oriented approach. It consists in checking the validity of feature values of the region, according to the properties and the constraints defined in the concepts. However, as a region does not have a semantic structure, we cannot directly use measures like MDSM (Rodriguez and Egenhofer, 2003), or other matching measures (Schwering and Raubal, 2005). A region can be matched with any concepts and the features of a region allowing the matching are not identical according to the studied concept. For example, the concept `Orange_House` is defined by several indexes (e.g. elongation, shape, etc.) and spectral attributes, while the concept `Shadow` is only defined with spectral attributes. Without *a priori* knowledge, this asymmetry involves to compute all the features for each region, even if the majority of them will not be used by the matching process. In order to take into account all these specificities, a matching measure based on a distance between the extracted features of a region and the observed values of the descriptors was proposed. The measure computes the relevance of a matching and is composed of a local component and a global component (*i.e.* evaluating the pertinence in the hierarchy of concepts).

The *matching score* $Score(R, C_i)$ between a region R and a concept C_i is based on the definition of a local similarity measure, that evaluates the similarity between a region and a specific concept of the hierarchy. Each attribute of the concept is compared to the corresponding attribute calculated on the region.

Definition 6 (degree of validity). Let $Valid(a, C, R)$ be the validity degree of an attribute ' a ' between a region R and a concept C .

$Valid(a, C, R)$ is equal to:

$$\begin{cases} 1 & \text{if } V'_R(a) \in [min(V_C(a)); max(V_C(a))] \\ \frac{V'_R(a)}{min(V_C(a))} & \text{if } V'_R(a) < min(V_C(a)) \\ \frac{max(V_C(a))}{V'_R(a)} & \text{if } V'_R(a) > max(V_C(a)) \end{cases}$$

Definition 7 (local similarity). Let $Sim_\alpha(R, C)$ be the local similarity between a region R and a concept C using the attributes of each class in α .

$$Sim_\alpha(R, C) = \frac{\sum_{a \in \mathcal{F}_\alpha(C)} \omega(a, C) Valid(a, C, R)}{\sum_{a \in \mathcal{F}_\alpha(C)} \omega(a, C)}$$

Definition 8 (matching score). Let $Score_\alpha(R, C)$ be the matching score between a region R and a concept C , and $\mathcal{P}(C)$ be the path starting from the root of the hierarchy and ending at the concept C . $\mathcal{P}(C) = \{C_j \mid C \preceq_\theta \dots \preceq_\theta C_2 \preceq_\theta C_1\}$.

$$Score_\alpha(R, C) = \frac{\sum_{C_j \in \mathcal{P}(C)} \rho(C_j) Sim_\alpha(R, C_j)}{\sum_{C_j \in \mathcal{P}(C)} \rho(C_j)}$$

Traversing the hierarchy of concepts. To match a region with the KB, it is necessary to navigate in the hierarchy to find the best concept (s) for a region. A level-wise algorithm (**Alg. 1**) was developed to navigate in the hierarchy of concepts using heuristics to reduce the search space: if the region matches the current concept, the algorithm will go deeper in the hierarchy; if the matching fails, the current concept is dropped and its sub-concepts will not be explored. The *maxDepth* value defines the exploration maximal depth (*i.e.* the degree of detail). The *minScore* threshold is the minimal value of the matching score between a region and a concept to allocate the corresponding label to the region.

Definition 9 (labels identified for a region). We define $\mathcal{L}_\alpha : \Gamma \rightarrow \Theta$ so that $\mathcal{L}_\alpha(R)$ is the set of concepts (seen as labels) identified for the region R according to the attributes of \mathcal{A}_α and the *minScore* value.

$$\mathcal{L}_\alpha(R) = \{C_i \mid \rho(C_i) \leq maxDepth \text{ and } Score_\alpha(R, C_i) \geq minScore \text{ and } \nexists C_j (\neq C_i) Score_\alpha(R, C_j) > Score_\alpha(R, C_i)\}$$

5. Experiments on remote sensing images

In order to illustrate how the knowledge-base can be used for automatic image interpretation, we carried out two series of experiments on three urban districts of Strasbourg (North-East of France) and on a district of Marseille (South of France) using Quickbird images (Digital Globe[®]). The Quickbird sensors produce two kind of images: panchromatic images with low spectral resolution but a high spatial resolution, and multispectral images with a good spectral resolution but a low spatial resolution. Consequently, each panchromatic image (at 0.61 m spatial resolution) were merged with the multispectral image (at 2.44 m spatial resolution), using the UWT-M2 method (Puissant et al., 2003) to obtain an image at 0.61 m spatial resolution with four spectral bands (**Fig. 4**).

All the districts are mainly composed of road (or parking), vegetation, water and small houses with gray or orange roofing tiles. Consequently, we focused our analysis to recognize the regions belonging to the concepts `Vegetation`, `Water`, `Road`, `Orange_House` and `Gray_House` of the KB. These concepts are the most relevant concepts to identify in urban areas according to geographer experts. For both cities, a set of manually labeled regions (*i.e.* ground truth) given by the expert were available. Some of the samples along with external information (*i.e.* topographic databases, expert knowledge, etc.) were used to fill the knowledge-base as described in the previous sections.

The Strasbourg knowledge-base was directly used for the labeling of the regions of the three districts of Strasbourg. For the district of Marseille, we performed the region labeling step by using three different knowledge-bases. Firstly, with the knowledge-base already used for the experiment on the Strasbourg districts. Secondly, with a knowledge-base created from information about the Marseille district. And finally, with a knowledge-base where the knowledge from Strasbourg and Marseille were merged. This experiment aimed at highlighting the ability of the KB to evolve and to leverage from new knowledge sources.

In the experiments, the segmentations of the images were computed using a supervised segmentation algorithm (Derivaux et al., 2010). This segmentation algorithm uses a fuzzy pixel classification using a k-Nearest Neighbor classifier and the watershed transform (Soille, 2003) applied on the fuzzy classification result.



Fig. 4. The districts used for the experiment extracted from Quickbird images.

5.1. Experiment on Strasbourg districts

The first step of the identification process is to segment the image in order to produce the set of regions to identify. An extract of the segmentation obtained from the District I and corresponding to the red square² in Fig. 4a is presented in Fig. 5b. For a better understanding of the scene, an aerial photography corresponding to the considered area is displayed in Fig. 5c.

Once this segmentation produced, we used the KB to identify the regions. In order to evaluate if the results obtained using our method were in agreement with the ground truth given by the expert (Fig. 6a), we computed the *precision*, *recall* and the *F-measure* (van Rijsbergen, 1979), according to different values of the *minScore* parameter (Section 4). For the *F-measure*, a value of 1 means that the result is in agreement with the ground truth. Table 5 presents the average values of the three indexes for the three districts of Strasbourg with in bold, the maximal value of each index. As the method is deterministic, with the same parameters set, two runs provide exactly the same results. From these results, one can see that the precision increases with the *minScore* while the recall decreases. This result is consistent as the method tends to be more

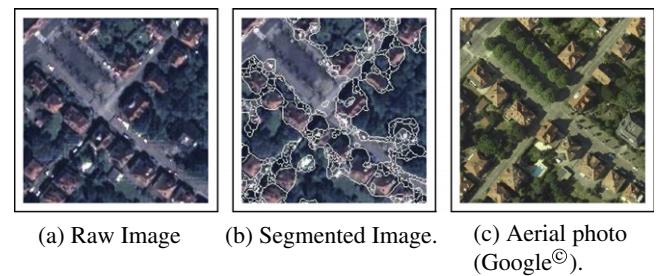


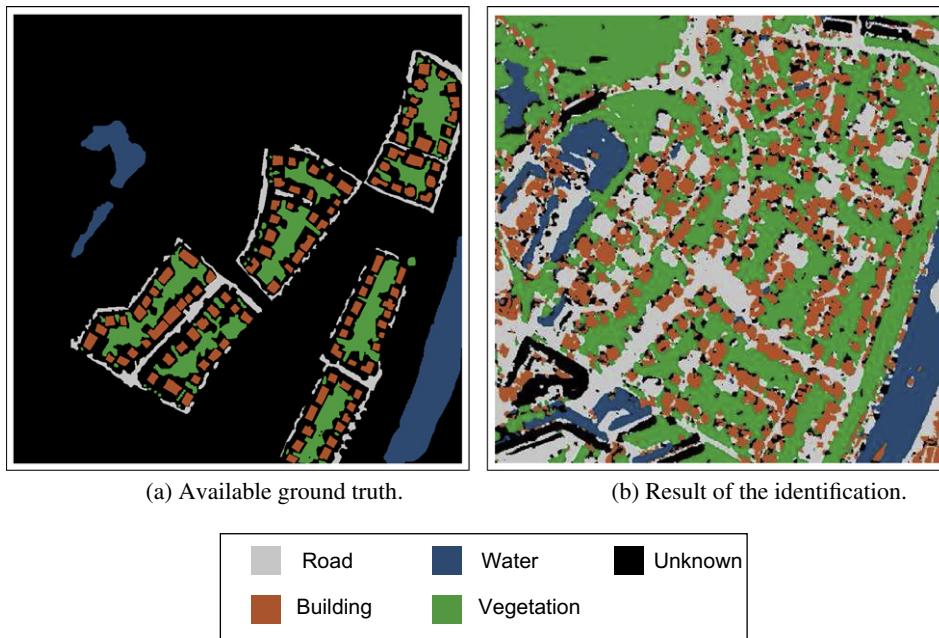
Fig. 5. Extract of the segmentation of the District I.

restrictive when the *minScore* increases. This result means that the method identified a fewer number of regions but with a higher confidence. One can also observe from the results that the best value for *minScore* is in [0.75, 0.85] regardless to the image.

5.1.1. Detailed results for District I

The Table 6 presents the detailed results for the District I: for each concept, for different values of *minScore*, it presents the values of the indexes according to the ground truth. From this table, one can see that the concepts *Vegetation* and *Water* are very well identified, except when *minScore* = 1. In that case, the recall rate

² For interpretation of color in Figs. 1, 2, 4–6, 8, and 9 the reader is referred to the web version of this article.

**Fig. 6.** Available ground truth and result identification of District I.**Table 5**

Assessment of the results of the identification according to the different Strasbourg districts. The higher value of each index according to the minScore has been highlighted in bold.

District (Fig. 4)	minScore	Precision	Recall	F-measure
Strasbourg I	0.75	0.859	0.859	0.859
	0.80	0.859	0.858	0.859
	0.85	0.861	0.857	0.859
	0.90	0.864	0.854	0.859
	0.95	0.876	0.837	0.856
	1.00	0.881	0.660	0.755
Strasbourg II	0.75	0.824	0.824	0.824
	0.80	0.826	0.824	0.825
	0.85	0.829	0.821	0.825
	0.90	0.836	0.816	0.826
	0.95	0.858	0.777	0.816
	1.00	0.999	0.533	0.695
Strasbourg III	0.75	0.862	0.861	0.861
	0.80	0.864	0.858	0.861
	0.85	0.864	0.855	0.860
	0.90	0.915	0.576	0.707
	0.95	0.956	0.164	0.281
	1.00	1.000	0.067	0.126

for the Water class is 0.276. It can easily be explained: when the expert defined the ground truth, he made one and only one area representing the river visible on the right of the image. In addition, reflections of the sun and turbulence in the water show the surface clearer than defined in the knowledge-base. The concept Road has good precision values and the recall values are acceptable. The precision values for the Building concept are relatively good but the recall values are very low.

The percentage of recognized objects and the percentage of the corresponding area in the image (*i.e.* the number of pixels from all the recognized objects) according to the minScore values, are illustrated in Fig. 7. The curves show that a major part of the image is recognized, and thus labeled. With minScore = 1, 18.9% of the objects are recognized corresponding to 53.7% of the image area.

Table 6

Results according to different minScore values for District I.

Class	Index					
	1.00	0.95	0.90	0.85	0.80	0.75
<i>Precision</i>						
Building	0.708	0.690	0.699	0.695	0.695	0.696
Vegetation	0.993	0.991	0.985	0.980	0.977	0.976
Road	0.850	0.843	0.832	0.826	0.824	0.823
Water	0.972	0.978	0.942	0.941	0.941	0.941
<i>Recall</i>						
Building	0.595	0.620	0.675	0.690	0.694	0.695
Vegetation	0.969	0.973	0.976	0.976	0.976	0.976
Road	0.801	0.815	0.823	0.823	0.823	0.823
Water	0.276	0.940	0.941	0.941	0.941	0.941
<i>F-measure</i>						
Building	0.647	0.653	0.687	0.692	0.694	0.695
Vegetation	0.981	0.982	0.980	0.978	0.976	0.976
Road	0.825	0.829	0.827	0.824	0.823	0.823
Water	0.430	0.959	0.941	0.941	0.941	0.941

With $\text{minScore} = 0.98$, 37.9% of the objects are identified and 66.2% of the image area. These results are promising: the majority of unlabeled objects correspond to small objects built from not properly segmented regions. Fig. 6b shows the result of the identification of the District I with a minScore of 0.98.

It is also important to note that some houses are not correctly segmented: the corresponding regions are sometimes composed of some pixels from shadow and vegetation. Thus, these houses could present features which do not correspond to the values defined in the knowledge-base, especially for the elongation indexes. Furthermore, very close buildings are sometimes grouped into only one single region and consequently, these regions cannot match with any concept of the knowledge-base. The opposite problem is encountered with the roads which are often over-segmented. In the following, we studied the influence of the segmentation step on the quality of the identification results.

5.1.2. Influence of the segmentation on identification results

As introduced previously, the identification results depends on the quality of the segmentation. Thus, in order to study and

Table 7

Results from different segmentations. The higher value of each index according to the minScore has been highlighted in bold.

Segmentation (Fig. 8)	minScore	Precision	Recall	F-measure
Watershed	0.75	0.815	0.815	0.815
	0.80	0.815	0.814	0.815
	0.85	0.815	0.813	0.814
	0.90	0.834	0.758	0.794
	0.95	0.838	0.734	0.783
	1.00	0.853	0.538	0.660
Supervised segmentation	0.75	0.842	0.842	0.842
	0.80	0.842	0.841	0.841
	0.85	0.843	0.840	0.842
	0.90	0.847	0.836	0.841
	0.95	0.857	0.819	0.837
	1.00	0.860	0.642	0.735
Supervised segmentation with some user modifications	0.75	0.859	0.859	0.859
	0.80	0.859	0.858	0.859
	0.85	0.861	0.857	0.859
	0.90	0.864	0.854	0.859
	0.95	0.876	0.837	0.856
	1.00	0.881	0.660	0.755
eCognition 5.0	0.75	0.806	0.815	0.810
	0.80	0.805	0.813	0.809
	0.85	0.805	0.812	0.808
	0.90	0.813	0.681	0.741
	0.95	0.851	0.625	0.721
	1.00	0.930	0.319	0.475
ENVI EX 4.8	0.75	0.848	0.861	0.854
	0.80	0.848	0.860	0.855
	0.85	0.848	0.858	0.853
	0.90	0.855	0.818	0.837
	0.95	0.871	0.776	0.821
	1.00	0.919	0.546	0.685

evaluate the influence of the segmentation, we carried out experiments on the extract of District I presented in Fig. 8a using five different segmentation approaches:

1. The watershed algorithm (Soille, 2003) (Fig. 8b).
2. A supervised segmentation algorithm (Derivaux et al., 2010) (Fig. 8c).
3. A supervised segmentation algorithm with manual corrections an expert made by splitting or merging regions (Fig. 8d).
4. The eCognition 5.0 software ³ (Fig. 8e) (parameters: scale = 50, color = 0.7, shape = 0.3).
5. The ENVI EX 4.8 software ⁴ (Fig. 8f) (parameters: scale = 30, merge = 90).

Table 7 presents the results of the identification using the five segmentations according to different values of the minScore parameter. The best F-measure value is obtained using the supervised segmentation algorithm with expert modifications, followed by the segmentation from ENVI EX, the supervised segmentation, the watershed, and finally the segmentation from eCognition. Note that the segmentations obtained using commercial softwares (eCognition and ENVI EX) offered good identification accuracies but low recalls when the minScore was high. This result means that few regions are identified but they are identified with a high degree of confidence. These results confirm that the better the segmentation, the better the results. Furthermore, our method turned out

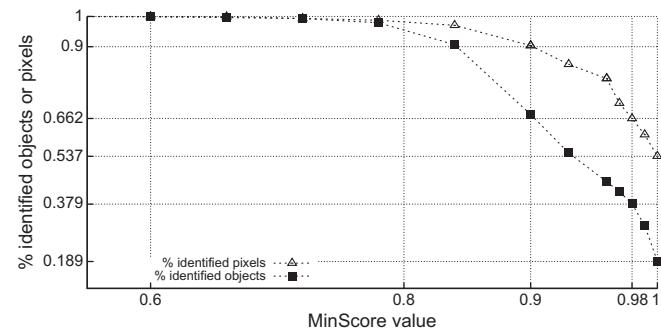


Fig. 7. Percentage of labeled objects and pixels according to the minScore value.

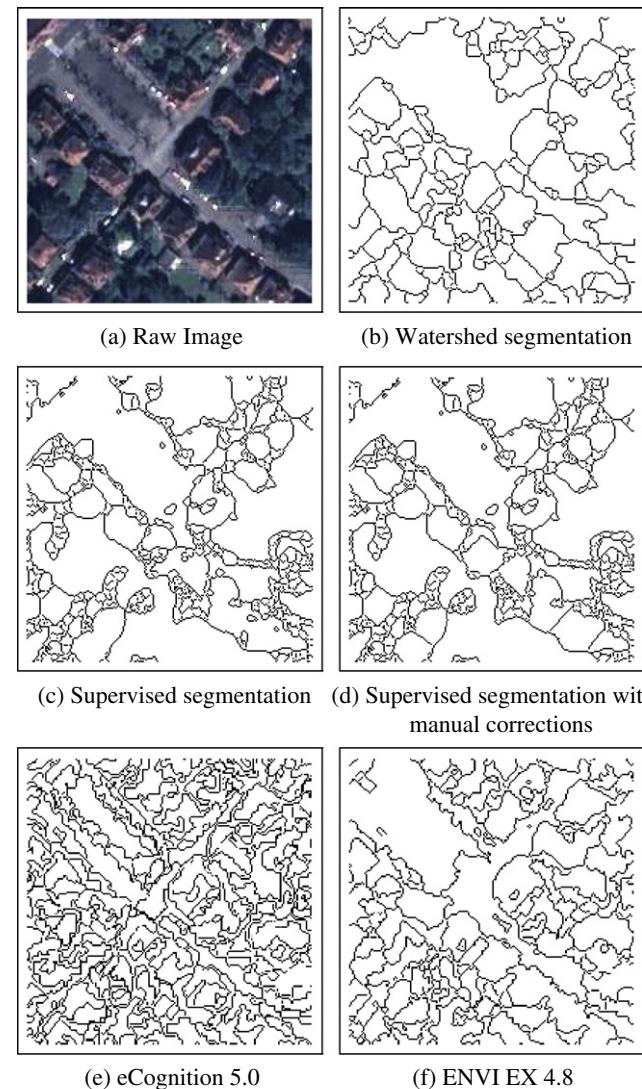


Fig. 8. Five segmentation extracts of the District I.

to be highly generic and the results were not as dependent of the segmentation as expected. Indeed, even with the over-segmented result proposed by the eCognition software, our identification method performed well. Finally, for all of the studied segmentations, the best F-measure value is obtained with a minScore value of 0.75 or 0.80 which means that the method is able to leverage from this parameter to soften the matching with the

³ <http://www.ecognition.com/>

⁴ <http://www.itvis.com/>

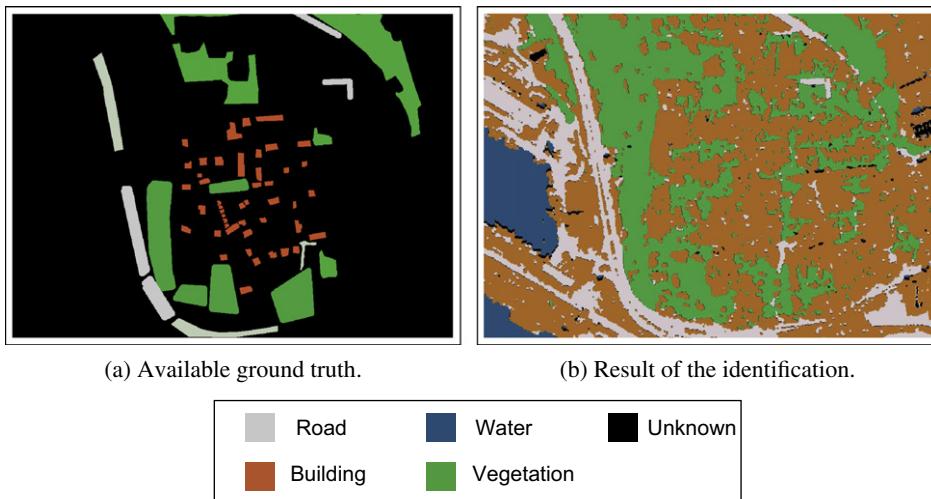


Fig. 9. Results for the Marseille district using both knowledge-bases.

Table 8

Precision (Prec.), recall (Recall) and F-Measure (F-M) from results on Marseille.

KB used	Marseille			Strasbourg			Both cities		
	Index			Prec.	Recall	F-M	Prec.	Recall	F-M
Class	Prec.	Recall	F-M	Prec.	Recall	F-M	Prec.	Recall	F-M
Building	0.353	0.892	0.506	0.589	0.751	0.660	0.600	0.963	0.739
Vegetation	0.976	0.725	0.832	0.995	0.873	0.930	0.988	0.905	0.945
Road	0.991	0.927	0.958	0.732	0.971	0.835	0.996	0.891	0.940
Means	0.774	0.848	0.809	0.772	0.865	0.816	0.861	0.919	0.889

knowledge-base leading to a better identification. A trade-off between quality of the identification and the amount of recognized regions is thus easily obtained.

5.2. Experiment on Marseille district

In this section, experiments carried out on the district of Marseille are presented. The aim of these experiments is to highlight the ability of our approach to reuse the knowledge acquired from previous experience. Consequently, we used the knowledge acquired from the Strasbourg images to identify regions in the image of Marseille.

The ground truth provided by the expert for the Marseille district suffered of two problems. First, the number of examples was very low and, second, there was no example of the water class. To evaluate the ability of our approach to deal with these data, we first segmented the Marseille district using the samples from Strasbourg. Then, the regions were labeled firstly, with the KB already used for the experiment on the Strasbourg districts. Secondly, with a KB created from information about the Marseille district. And finally, with a KB where the knowledge from Strasbourg and Marseille were merged. Fig. 9 presents the results obtained in that last experiment. The quality of the results were evaluated using the samples from Marseille. Note that, as there were no example of water in these examples, this class was not evaluated. The results presented in Table 8 show that even if there is no knowledge available on the studied image, our approach can be used and rely on the knowledge acquired in the past. Indeed, by using only the knowledge extracted from Strasbourg images, we were able to identify regions in the Marseille image. Furthermore, the results show that when we enriched the KB with the Marseille knowledge,

the detection performed even better. This result is consistent as the KB including also the Marseille knowledge had more information about regions extracted from Marseille district.

6. Conclusion

In this paper, the steps to build an urban knowledge-base applied to HSR image analysis were presented and a new knowledge representation was introduced. The approach is based on a domain-dependent knowledge-base developed by experts of the domain. A similarity measure and an exploration procedure of the knowledge-base were used in order to affect a semantic to the regions of a segmented image. The experimental results highlighted the effectiveness of the method, and the obtained results were compared using different segmentation approaches, including commercial softwares. The results also showed that even if there was no knowledge available on a studied area, our approach could be used and rely on the knowledge acquired in the past.

In the future, we will plan several experiments on different types of urban images using other segmentation algorithms. We also wish to integrate the method into a framework of collaborative clustering. Indeed, in Forestier et al. (2008b), collaborative clustering and the knowledge extracted from a knowledge-base were used together. Furthermore, we also plan on using directly the knowledge during the segmentation step (Forestier et al., 2008a) in order to build regions easily identifiable. In order to improve and to enrich the content of the knowledge-base, machine learning techniques continues to be developed in order to automatically extract information from the HSR images. In particular, we are focusing on topological relations based on the RCC-8 (Region Connection Calculus) theory.

References

- Athanasiadis, T., Mylonas, P., & Avrithis, Y. (2007). Semantic image segmentation and object labeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 17, 298–312.
- Athanasiadis, T., Mylonas, P., Avrithis, Y., & Kollias, S. (2007). Semantic image segmentation and object labeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 17, 298–312.
- Autran, J. (2007). Extension de la nomenclature corine land cover pour la description de l'occupation du sol urbain à grande échelle. In *Journée francophone sur les ontologies*. Sousse, Tunisia.
- Baltasavias, E. (2004). Object extraction and revision by image analysis using existing geodata and knowledge: Current status and steps towards operational systems. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58, 129–151.
- Benz, U., Hofmann, P., Willhauck, G., Lingenfelder, I., & Heynen, M. (2004). Multi-resolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58, 239–258.
- Breen, C., Khan, L., & Ponnusamy, A. (2002). Image classification using neural networks and ontologies. In *Proceedings of 13th international workshop on database and expert systems applications, co-located with DEXA 2002* (pp. 98–102). Aix-en-Provence, France.
- Bückner, J., Pahl, M., Stahlhut, O., & Liedtke, C.-E. (2002). A knowledge-based system for context dependent evaluation of remote sensing data. In L. J. V. Gool (Ed.), *DAGM-symposium. Lecture notes in computer science* (Vol. 2449, pp. 58–65). Springer.
- Castellano, G., Fanelli, A. M., & Torsello, M. A. (2011). Fuzzy image labeling by partially supervised shape clustering. In A. Knig, A. Dengel, K. Hinkelmann, K. Kise, R. J. Howlett, & L. C. Jain (Eds.), *KES (2). Lecture notes in computer science* (Vol. 6882, pp. 84–93). Springer.
- Cataldo, A., & Rinaldi, A. M. (2010). An ontological approach to represent knowledge in territorial planning science. *Computers, Environment and Urban Systems*, 34, 117–132.
- Cleve, C., Kelly, M., Kearns, F.R., & Moritz, M. (2008). Classification of the wildland-urban interface: A comparison of pixel- and object-based classifications using high resolution aerial photography. *Computers, Environment and Urban Systems*, doi:10.1016/j.compenvurbsys.2007.10.001.
- Crevier, D., & Lepage, R. (1997). Knowledge-based image understanding systems: a survey. *Computer Vision and Image Understanding*, 67, 161–185.
- Dasiopoulou, S., Mezaris, V., Kompatiari, I., Papastathis, V. K., & Strintzis, M. G. (2005). Knowledge-assisted semantic video object detection. *IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Analysis and Understanding for Video Adaptation*, 15, 1210–1224.
- Derivaux, S., Forestier, G., Wemmert, C., & Lefèvre, S. (2010). Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation. *Pattern Recognition Letters*, 31, 2364–2374.
- Draper, B., Collins, A., Brolio, J., Hanson, A., & Riseman, E. (1989). The schema system. *International Journal of Computer Vision*, 2, 209–250.
- Durand, N., Derivaux, S., Forestier, G., Wemmert, C., Gançarski, P., & Boussaid, O., et al. (2007). Ontology-based object recognition for remote sensing image interpretation. In *IEEE international conference on tools with artificial intelligence* (Vol. 1, pp. 472–479). Patras, Greece: IEEE Computer Society.
- Fonseca, F., Egenhofer, M., Agouris, P., & Camara, G. (2002). Using ontologies for integrated geographic information systems. *Transactions in GIS*.
- Forestier, G., Derivaux, S., Wemmert, C., & Gançarski, P. (2008a). An evolutionary approach for ontology driven image interpretation. In *Tenth European workshop on evolutionary computation in image analysis and signal processing. Lecture Notes in Computer Sciences* (vol. 4974, pp. 295–304). Napoli, Italy: Springer.
- Forestier, G., Derivaux, S., Wemmert, C., & Gançarski, P. (2008b). On combining unsupervised classification and ontology knowledge. In *IEEE geoscience and remote sensing symposium*. Boston: Massachusetts.
- Gomez, F., & Segami, C. (2007). Semantic interpretation and knowledge extraction. *Knowledge-Based Systems*, 20, 51–60.
- Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human Computer Studies*, 43, 907–928.
- Haralick, R. (1979). Statistical and structural approaches to texture. *Proceeding of the IEEE*, 67, 45–69.
- Herold, M., Scepan, J., Muller, A., & Gunter, S. (2002). Object-oriented mapping and analysis of urban landuse/cover using ikonos data. In *Proceedings of 22nd Earsel symposium geoinformation for European-wide integration* (pp. 531–538). Prague.
- Lillesand, T. M., Kiefer, R. W., & Chipman, J. W. (2003). *Remote sensing and image interpretation*. Wiley.
- Maillot, N., & Thonnat, M. (2008). Ontology based complex object recognition. *Image and Vision Computing*, 26, 102–113.
- Matsuyama, T., & Hwang, V.-S. (1990). *SIGMA – a knowledge-based aerial image understanding system*. New York, USA: Plenum Press.
- Moller-Jensen, L. (1997). Classification of urban land cover based on expert systems, object models and texture. *Computers, Environment and Urban Systems*, 21, 291–302.
- Momm, H., Easson, G., & Kuszmahl, J. (2009). Evaluation of the use of spectral and textural information for an evolutionary algorithm for multi-spectral imagery classification. *Computers, Environment and Urban Systems*, 33, 463–471.
- Noy, N.F., Ferguson, R.W., & Musen, M.A. (2000). The knowledge model of protege-2000: combining interoperability and flexibility. In *Proceedings of 12th international conference on knowledge engineering and knowledge management (EKAW 2000)* (pp. 17–32). Juan-les-Pins, France.
- Panagi, P., Dasiopoulou, S., Papadopoulos, G.T., Kompatiari, I., & Strintzis, M.G. (2006). A genetic algorithm approach to ontology-driven semantic image analysis. In *Proceedings of 3rd IEE international conference of visual information engineering (VIE 2006)* (pp. 132–137). Bangalore, India.
- Pantazis, D., Cornelis, B., Billen, R., & Sheeren, D. (2002). Establishment of a geographic data dictionary: A case study of urbis 2, the brussels regional government gis. *Computers, Environment and Urban Systems*, 26, 3–17.
- Puissant, A., Ranchin, T., Weber, C., & Serradj, A. (2003). Fusion of Quickbird MS and Pan data for urban studies. In *Proceedings of European association of remote sensing laboratories symposium (EARSeL)* (pp. 77–83). Gent, Belgium.
- Rodriguez, M. A., & Egenhofer, M. J. (2003). Determining semantic similarity among entity classes from different ontologies. *IEEE Transactions on Knowledge and Data Engineering*, 15, 442–456.
- Schwing, A., & Raubal, M. (2005). Measuring semantic similarity between geospatial conceptual regions. In *Proceedings of 1st international conference on geospatial semantics (GeoS). Lecture Notes in Computer Science* (Vol. 3799, pp. 90–106). Mexico City, Mexico.
- Sheeren, D., Puissant, A., Weber, C., Gançarski, P., & Wemmert, C. (2006). Deriving classification rules from multiple remotely sensed data with data mining. In *Proceedings of 1st workshop of the EARSeL special interest group on urban remote sensing*. CDROM 9p, Berlin.
- Sheeren, D., Quirin, A., Puissant, A., Gançarski, P., & Weber, C. (2006). Discovering rules with genetic algorithms to classify urban remotely sensed data. In *Proceedings of IEEE international geoscience and remote sensing symposium (IGARSS'2006)* (pp. 3919–3922). Denver, Colorado.
- Smeulders, A., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 1349–1380.
- Soille, P. (2003). *Morphological image analysis* (2nd ed.). Springer-Verlag.
- Tiijnes, R., Glowe, S., Biicknel, J., & Liedtke, C. (1999). *Knowledge-based interpretation of remote sensing Images using semantic nets*.
- Uitermark, H., van Oosterom, P., Mars, N. J. I., & Molenaar, M. (1999). Ontology-based geographic data set integration. In *Spatio-temporal database management. Lecture notes in computer science* (vol. 1678, pp. 60–78). Springer.
- Uschold, M., & King, M. (1995). Towards a methodology for building ontologies. In *Workshop on basic ontological issues in knowledge sharing*. Montreal, Canada.
- van Rijsbergen, C. (1979). *Information retrieval*. London: Butterworths.
- Waterson, A., & Preece, A. (1999). Verifying ontological commitment in knowledge-based systems. *Knowledge-Based Systems*, 12, 45–54.
- Wu, H., Li, Y., Li, Q., & Chen, X. (2009). Research on fractal model of urban land use considering the appropriate spatial resolution for remote sensing imagery. *SPIE*, 7498, 749816.
- Zlatoff, N., Tellez, B., & Baskurt, A. (2004). Image understanding and scene models: a generic framework integrating domain knowledge and gestalt theory. In *Proceedings of IEEE international conference on image processing (ICIP 2004)* (pp. 2355–2358). Singapore.



Collaborative clustering with background knowledge

G. Forestier, P. Gançarski*, C. Wemmert

University of Strasbourg, LSIIT, UMR7005, Pôle API, Bd Sébastien Brant, BP 10413, 67412 Illkirch, Cedex, France

ARTICLE INFO

Article history:

Received 23 March 2009

Received in revised form 5 October 2009

Accepted 5 October 2009

Available online 27 October 2009

Keywords:

Collaborative clustering

Unsupervised learning

Classification

Pattern recognition

Knowledge-guided clustering

ABSTRACT

The aim of collaborative clustering is to make different clustering methods collaborate, in order to reach at an agreement on the partitioning of a common dataset. As different clustering methods can produce different partitioning of the same dataset, finding a consensual clustering from these results is often a hard task. The collaboration aims to make the methods agree on the partitioning through a refinement of their results. This process tends to make the results more similar.

In this paper, after the introduction of the collaboration process, we present different ways to integrate background knowledge into it. Indeed, in recent years, the integration of background knowledge in clustering algorithms has been the subject of a lot of interest. This integration often leads to an improvement of the quality of the results. We discuss how such integration in the collaborative process is beneficial and we present experiments in which background knowledge is used to guide collaboration.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Clustering is an important machine learning tool for discovering hidden patterns, structures and relationships between data objects in an unsupervised way. It has been widely used in pattern recognition fields, mainly to classify groups of measurements or observations. It finds applications in bioinformatics, information retrieval, medicine, image analysis or financial markets, to structure information and discover hidden knowledge.

Over the last 50 years, a huge number of new clustering algorithms have been developed, and existing methods have been modified and improved [1–5]. This abundance of methods can be explained by the ill-posed nature of clustering. Indeed, each clustering algorithm is biased by the objective function used to build the clusters. Consequently, different methods can, from the same data, produce very different clustering results. Furthermore, even the same algorithm can produce different results, according to its parameters and initialization. A relatively recent approach to circumvent the problem is based on the idea that the information offered by different sources and different clustering, are complementary [6]. Thus, the combination of different clusterings may increase their efficiency and accuracy. A single classification is produced from results of methods having different points of view: each individual clustering opinion is used to find a consensual decision. Each decision can be processed from a different source or media.

In the same way, we address the problem of the collaboration between different clustering methods. Collaboration is a process where two or more actors work together to achieve a common goal by sharing knowledge. In our collaborative clustering method, called SAMARAH [52], different clustering methods work together to reach an agreement on their clustering. Each clustering modifies its results according to all the other clusterings until all the clusterings proposed by the different methods are strongly similar. Thus, they can be more easily unified, for example, through a voting algorithm.

Different studies showed that this method provides interesting results on artificial data sets and on real life problems [52]. However, a lot of work is currently focusing on integrating background knowledge into the clustering process. This

* Corresponding author. Tel.: +33 3 90 24 45 76; fax: +33 3 90 24 44 55.

E-mail addresses: forestier@unistra.fr (G. Forestier), gancarski@unistra.fr (P. Gançarski), wemmert@unistra.fr (C. Wemmert).

work highlights the benefits of knowledge integration into this process by showing that the use of such knowledge to drive the process leads to more accurate results. The aim of this paper is to present an extension of SAMARAH, which takes into account background knowledge during the collaboration of the methods. The background knowledge is used to drive the collaboration between the clustering methods and allows an improvement of the final results.

To evaluate benefits of using the collaborative clustering method SAMARAH with (and without) background knowledge integration, we used classical approaches based on quality indexes and a more recent approach called *cascade evaluation*. Cascade evaluation [7] is a new approach to evaluate the quality and the interest of clustering results. The method is based on the enrichment of a set of labeled datasets by the results of clusterings, and the use of a supervised method to evaluate the benefit of adding such new information to the data sets.

The cascade evaluation of the SAMARAH method highlights that collaboration increases the quality of the refined results in both cases: with and without the use of background knowledge during the collaboration.

This paper is organized as follows. First, we describe work related to our approach (Section 2). In particular, we introduce clustering methods using background knowledge (Section 2.4). Section 3 describes the collaborative clustering method SAMARAH and presents the knowledge integration into it. Then, in Section 4, after a brief introduction to cascade evaluation principles, the evaluations of the collaborative clustering on various datasets from the UCI repository are detailed and discussed. Finally, conclusions and perspectives are drawn in Section 5.

2. Related works

In recent years, a lot of work has focused on the use of multiple clusterings to improve the unsupervised classification process. Indeed, many different algorithms exist and may provide different results from the same dataset. Consequently, it is often difficult to design a single algorithm whose results reflect what users need and expect. To cope with this problem, the ensemble clustering approaches (Section 2.1) consist in designing a function which summarizes several clusterings into a single one. The aim is to find the average partition which is the most similar to all the results of the ensemble. However, the ensemble clustering methods only focus on the creation of the consensus and do not modify or create new partitions. Consequently, they consider that the initial provided partitions are the only ones necessary.

Alternatively, the multi-objective approaches (Section 2.2) see the clustering process as an optimization of different objectives. In this methods, a vast amount of different clustering results are explored, and the ones which best match the objectives are kept. A genetic algorithm is generally used and new clusterings are created by mixing different results together. However, the objective use in the optimization are not always the objective of the algorithm used to create the initial partition, and thus induce a new bias.

In the fuzzy collaborative clustering approaches (Section 2.3), the fuzzy c-means algorithm is use to cluster different views of the data in a collaborative way. This work only focuses on the fuzzy c-means algorithm which limits its use. However, its provide strong theoretical basis which better highlight the benefit of the collaboration in specific environment.

The use of different clusterings result often require a strong implication of the user, as there is a lot of parameter to choose. To reduce the need of the user and improve the quality of the results, a lot of work has also focused on the use of background knowledge (Section 2.4). Background knowledge has many different representations like a set of labeled patterns, link between patterns, the number of expected clusters, the expected size of the clusters, etc.

2.1. Ensemble clustering

The aim of ensemble clustering is to generate multiple clusterings and to merge them to produce a final consensus clustering. The initial clusterings are generally generated by applying different algorithms, using different parameters of the same algorithm, or by random sampling of the dataset. Ensemble clustering is often used to improve the accuracy of the data clustering or to find complex shaped clusters. Indeed, using multiple clusterings allows ensemble clustering algorithms to better grasp the underlying distribution of the data space.

Strehl et al. [8] formulated the consensus clustering as the partition that maximize the shared information among the ensemble of initial clusterings. This information is measured through the Average Normalized Mutual Information (ANMI) which uses the information theory framework. Three different clustering ensemble strategies based on graph theory are presented : Cluster-based Similarity Partitioning Algorithm (CSPA), Hyper Graph Partitioning Algorithm (HGPA), and Meta Clustering Algorithm (MCLA). Further work reveals that these approaches are sensitive to cluster sizes and seek only for balanced-size clusters (i.e. all the clusters tend to have the same number of data objects).

Another approach, called evidence accumulation, is presented by Fred et al. [9]. The main idea is to produce a co-association matrix from the different initial clusterings. This matrix gives the information of the number of times that two data objects have been put together in the same cluster. A hierarchical clustering is then used, using the co-association matrix as a distance matrix, to cluster the objects into the final partition. Various approaches using the co-association matrix have been presented in the literature and seem to outperform the graph based methods.

Topchy et al. [10] described how to create a new feature space from an ensemble clustering by interpreting the multiple clusterings as a new set of categorical features. The KMEANS algorithm is applied on this new standardized feature space using

a category utility function to evaluate the quality of the consensus. Topchy et al. [11] also demonstrated the efficiency of combining partitions generated by weak clustering algorithms that use data projections and random data splits.

Hadjitodorov et al. [12] presented an evaluation of different heuristics to produce the initial set of clusterings. The most often selected heuristics were random feature extraction, random feature selection and random number of clusters assigned to each member of the ensemble.

Ayad and Kamel [13] present a cumulative voting method to come to a consensus from partitions with a variable number of clusters. They described several cumulative vote weighting schemes and corresponding algorithms, to compute an empirical probability distribution summarizing the partitions. The empirical study shows the efficiency of the method compared to others consensus clustering algorithms.

More voting methods are given by Nguyen et al. [14]. Three iterative algorithms are presented: Iterative Voting Consensus (IVC), Iterative Probabilistic Voting Consensus (IPVC) and Iterative Pairwise Consensus (IPC). These algorithms use a feature map built from the set of base clusterings and apply an EM-like consensus clustering.

To directly address the correspondence problem (among the clusters of different clusterings) in combining multiple clusterings, Zhang et al. [15] introduced a new framework based on soft correspondence. The proposed algorithm provides a consensus clustering method as well as correspondence matrices that give the relations between the clusterings of the ensemble.

Hu et al. [16] proposed a method which uses Markov random fields and maximum likelihood estimation to define a metric distance between clusterings. They present two combining methods based on this new similarity to find a consensus.

More recently, Pedrycz et al. [17] proposed a consensus-driven fuzzy clustering method. The authors consider the proximity matrices induced by the corresponding partition matrices. An optimization scheme is presented in detail along with a way of forming a pertinent criterion. This criterion governs an intensity of collaboration to guide the process of consensus formation.

The ensemble clustering approaches do not generally address the problem of the generation of the initial results, and the algorithms used to create the initial results are not used in the combination process. Consequently, ensemble clustering approaches introduce a new bias, relative to the objective function chosen when merging the different clusterings.

2.2. Multi-objective clustering

The aim of multi-objective clustering is to optimize simultaneously several clustering criteria. The idea is to have a better grasp of the notion of cluster by explicitly defining them with different objective functions. Algorithms are able to produce a set of trade-off solutions between the different objectives. These solutions correspond to different compromises of the objectives used.

Thus, the method `MOCK` (Multi-Objective Clustering with automatic K-determination) [18] uses two objectives: the first one is to maximize the compactness of the clusters, and the second one their connectivity. A multi-objective evolutionary algorithm is used to optimize these two criteria simultaneously. The method uses a Pareto based approach [19] which consists in selecting the non-dominated solutions of the Pareto front. At the end of the evolution, the solutions on the Pareto front are the set of solutions provided by the algorithm. A heuristic is then used to select the best potential solution by using the number of clusters of the solutions on the front. In [20], the authors present how to integrate background knowledge through a third objective optimization which uses a subset of labeled samples. This semi-supervised version outperformed the solution without background knowledge.

Faceli et al. [21] described the multi-objective method called `MOCLE` (Multi-Objective Clustering Ensemble) which integrates the same objective function (maximization of compactness and connexity of the clusters) as `MOCK`. But a special cross-over operator which uses ensemble clustering techniques is added: the purpose of the `MOCLE` method is to produce a set of solutions which are a trade-off of the different objectives, while the `MOCK` method produces a single solution.

Law et al. [22] proposed a method which uses different clustering methods using different objectives. The final result is produced by selecting clusters among the results proposed by the different methods. A resampling method is used to estimate the quality of the clusters.

A semi-supervised extension of `MOCLE` has also been proposed recently [23]. The prior knowledge about a known structure of the data is integrated by means of an additional objective function that takes external information into account. The new objective aims at creating pure clustering according to known object class label. The objective is optimized along with the previous one on the data and the use of background knowledge improved the result of the method.

An evolutionary version of the `KMEANS` algorithm is used by [24], driven by a semi-supervised objective. This objective is a weighted sum of the mean squared error (MSE) and the purity of the clusters according to a subset of available samples. Different criteria are investigated to quantify this purity.

In [22], the authors proposed a method which uses different clustering methods using different objectives. The final result is produced by selecting clusters among the results proposed by the different methods. A resampling method is used to estimate the quality of the clusters.

2.3. Fuzzy collaborative clustering

A fuzzy clustering architecture is introduced by Pedrycz et al. [25], in which several subsets of patterns can be processed together to find a common structure to all of them. In this system, different subsets of the initial data are processed

independently. Then, each partition matrix is modified according to the other matrices found: each result produced on a subset is modified according to results found on the other subsets. Extensive experiments of the method are also proposed in [26] along with algorithmic details. An application of this collaborative fuzzy clustering method to semantic web content analysis has been proposed in [27]. The authors discuss a collaborative proximity-based fuzzy clustering and show how this type of clustering is used to discover a structure of web information in the spaces of semantics and data.

A fuzzy collaborative framework is also proposed [28], where rough sets are used to create a collaborative paradigm in which several subsets of patterns are processed together to find a common structure. A clustering algorithm is developed by integrating the advantages of both fuzzy sets and rough sets. A quantitative analysis of the experimental results is also provided for synthetic and real-world data.

To tackle the problem of distributed data, [29] proposed a framework to cluster distributed classifier. They show that clustering distributed classifiers as a pre-processing step for classifier combination enhances the achieved performance of the ensemble.

2.4. Clustering with background knowledge

Many approaches have been investigated to use background knowledge to guide the clustering process.

In constrained clustering, knowledge is expressed as *must-link* and *cannot-link* constraints and is used to guide the clustering process. A *must-link* constraint gives the information that two data objects should be in the same cluster, and *cannot-link* means the opposite. This kind of knowledge is sometimes easier to obtain than a classical subset of labeled samples. Wagstaff et al. [30] presented a constrained version of the KMEANS algorithm which uses such constraints to bias the assignment of the objects to the clusters. At each step, the algorithm tries to agree with the constraints given by the user. These constraints can also be used to learn a distance function biased by the knowledge about the links between the data objects [31]. The distance between two data objects is reduced for a must-link and increased for a cannot-link. Huang et al. [32] presented an active learning framework for semi-supervised document clustering with language modeling. The approach uses a gain-directed document pair selection method to select cleverly the constraints. In order to minimize the amount of constraints required, Griga et al. [33] defined an active mechanism for the selection of candidate constraints. The active fuzzy constrained clustering method is presented and evaluated on a ground truth image database to illustrate that the clustering can be significantly improved with few constraints. Recent works on constrained clustering are focused on evaluating the utility (i.e. the potential interest) of a set of constraints [34,35].

Kumar and Kummamuru [36] introduced another kind of knowledge through a clustering algorithm that uses supervision in terms of relative comparisons, e.g. x is closer to y than to z . Experimental studies on high-dimensional textual data sets demonstrated that the proposed algorithm achieved higher accuracy and is more robust than similar algorithms using pairwise constraints (*must-link* and *cannot-link*) for supervision.

Klein et al. [37] allowed instance-level constraints (i.e. *must-link*, *cannot-link*) to have space level inductive implications in order to improve the use of the constraints. This approach improved the results of the previously studied constrained KMEANS algorithms and generally requires less constraints to obtain the same accuracies.

Basu et al. [38] presented a pairwise constrained clustering framework as well as a new method for actively selecting informative pairwise constraints, to get improved clustering performance. Experimental and theoretical results confirm that this active querying of pairwise constraints significantly improves the accuracy of clustering, when given a relatively small amount of supervision.

Another way to integrate background knowledge is to use a small set of labeled samples. Basu et al. [39] used a set of samples to *seed* (i.e. to initialize) the clusters of the KMEANS algorithm. Two algorithms, Seeded KMEANS and Constrained KMEANS, are presented. In the first one, the labeled samples are used to initialize the clusters and the clusters are updated during the clustering process such as in the KMEANS algorithm. In the second one, the labeled samples used during the initialization stay in their assigned cluster, and only the unlabeled samples can change of cluster during the cluster affectation step of KMEANS. The choice between these two approaches must be done according to the knowledge about noise in the dataset.

To tackle the problem of incorporating partial background knowledge into clustering, when the labeled samples have moderate overlapping features with the unlabeled data, Gao et al. [40] formulated a new approach as a constrained optimization problem. The authors introduced two learning algorithms to solve the problem, based on hard and fuzzy clustering methods. An empirical study shows that the proposed algorithms improve the quality of clustering results despite a limited number of labeled samples.

Basu et al. [41] also proposed a probabilistic model for semi-supervised clustering, based on Hidden Markov Random Fields (HMRF), that provides a principled framework for incorporating supervision into prototype-based clustering. Experimental results on several text data sets demonstrate the advantages of this framework.

Another approach, called supervised clustering [42], uses the class information about the objects as an additional feature, to build clusters with a high class-based purity. The goal of supervised clustering is to identify class-uniform clusters having high probability densities. Supervised clustering is used to create summaries of datasets and for enhancing existing classification algorithms.

Different kinds of background knowledge are introduced by Pedrycz et al. [43], namely partial supervision, proximity-based guidance and uncertainty driven knowledge hints. The authors discuss about different ways of exploiting and effectively incorporating these background knowledge (known as *knowledge hints*) in the fuzzy c-means algorithm. In [44],

Bouchachia and Pedrycz presented an extension of the fuzzy collaborative clustering which takes into account background knowledge through labeled objects. One of the advantages of the method is to take into account the classes splitted in several clusters. During the collaboration step, the method identify if a class correspond to various clusters and add or remove clusters according to this information. More recently, Pedrycz [45] presented some concepts and algorithms to collaborative and knowledge-based fuzzy clustering. The fuzzy c-means algorithm (FCM) was used as an operational model to explain the approach. Interesting linkages between information granularity, privacy and security of data in collaborative clustering were also discussed. The problem of data privacy when clustering multiple datasets was also recently discussed in [46]. An application of fuzzy clustering with partial knowledge to organize and classify digital images is also proposed in [27]. The author present an operational framework of fuzzy clustering using the fuzzy c-means algorithm with an augmented objective function using background knowledge. Experiments are carried out on collections of images composed of 2000 images.

In this section, we presented different works on using multiple clusterings: ensemble clustering (Section 2.1), multi-objective clustering (Section 2.2) and collaborative fuzzy clustering (Section 2.3). The ensemble clustering approaches do not generally address the problem of the generation of the initial results, and the algorithms used to create the initial results are not used in the combination process. Consequently, ensemble clustering approaches introduce a new bias, relative to the objective function chosen when merging the different clusterings. The same problem appears with multi-objective clustering approaches where the optimized objectives are not the objectives of the methods used to generate the initial results. Collaborative fuzzy clustering offers strong theoretical basis on collaborative clustering but is only developed for fuzzy c-means which limit its use.

Finally, we presented several work on the integration of background knowledge in clustering algorithm (Section 2.4). Different kinds of representation of the knowledge and different kinds of integration exist. However, every work claims that using background knowledge improves substantially the results of clustering algorithms. The challenge task is to design methods able to leverage different kinds of knowledge. We propose such a method in the next section, where the collaborative clustering method SAMARAH is presented along with the different ways to integrate background knowledge into it.

3. Knowledge-guided collaborative clustering

As seen in Section 2, many techniques for combining clusterings exist. Unfortunately, only a few of them can handle the combination of clusterings having different numbers of clusters, because there is no obvious correspondence between the clusters of the different results.

Moreover, the proposed methods almost always aim to build a consensus among an ensemble of partitions or clusterings, without casting doubt on their quality. We think that a first step of collaboration of the clustering method before the consensus computation can help to obtain better results.

Thus, we propose a method consisting of a collaborative clustering process, based on an automatic and mutual refinement of several clustering results.

In this section, we first present the existing unsupervised collaborative clustering method called SAMARAH. Then, we present how we integrate knowledge into this collaborative clustering process.

3.1. Collaborative process overview

Computing a consensual result from clustering results having different numbers of clusters is a difficult task. This is mainly due to the lack of a trivial correspondence between the clusters of the different results. To address this particular problem, we present a framework where different clustering methods work together, in a collaborative way, to find an agreement about their proposals.

This collaborative process consists of an automatic and mutual refinement of the clustering results, until all the results have almost the same number of clusters, and all the clusters are statistically similar with a good internal quality. At the end of this process, as the results have comparable structures, it is possible to define a correspondence function between the clusters, and to apply a unifying technique, such as a voting method [47].

Before the description of the collaborative method, we introduce the correspondence function and the similarity measure used in the system.

There is no problem to associate classes of different supervised classifications, as a common set of class labels is given for all the classifications. Unfortunately, in the case of clustering, the results may not have a same number of clusters, and no information is available about the correspondence between different clusters of different clusterings.

To address this problem, we have defined an intercluster correspondence function, which associates to each cluster from a result, a cluster from each of the other results. This cluster, in each result, is called the corresponding cluster.

Let $\check{\mathcal{R}} = \{\mathcal{R}^i\}_{1 \leq i \leq m}$ be the set of results given by the m different algorithms. Let $\{C_k^i\}_{1 \leq k \leq n_i}$ be the clusters of the result \mathcal{R}^i . The corresponding cluster $CC(C_k^i, \mathcal{R}^j)$ of C_k^i in the result \mathcal{R}^j , $i \neq j$, is defined as

$$CC(C_k^i, \mathcal{R}^j) = \arg \max_{C_l^j \in \mathcal{R}^j} S(C_k^i, C_l^j), \quad (1)$$

where S is the intercluster similarity which evaluates the similarity between two clusters of two different results.

A large number of criteria exists to evaluate the similarity between two clustering results, like the Kappa index [48] or the Rand index [49], and more recently the Jaccard index [50] or the Fowlkes–Mallows index [51]. However, these criteria only give a global evaluation of the similarity between two partitions. In order to find the most similar cluster of one result in another one, we have to use an index based on the similarity between the two results *and* the similarity between each cluster of each result. To achieve this goal, we introduced the intercluster similarity between two clusters of two different results, which takes into account both aspects: the similarity through the confusion matrix between the two results (α in Eq. (3)), and the distribution of the cluster into the clusters of the second result through a distribution coefficient (ρ in Eq. (3)).

The confusion matrix (or matching matrix) is commonly used to compare two partitions or clustering results. The confusion matrix Ω^{ij} between two results \mathcal{R}^i and \mathcal{R}^j is a $n_i \times n_j$ matrix defined by:

$$\Omega^{ij} = \begin{pmatrix} \alpha_{1,1}^{ij} & \cdots & \alpha_{1,n_j}^{ij} \\ \vdots & & \vdots \\ \alpha_{n_i,1}^{ij} & \cdots & \alpha_{n_i,n_j}^{ij} \end{pmatrix} \quad \text{where } \alpha_{k,l}^{ij} = \frac{|\mathcal{C}_k^i \cap \mathcal{C}_l^j|}{|\mathcal{C}_k^i|}. \quad (2)$$

The *intercluster similarity* between two clusters \mathcal{C}_k^i and \mathcal{C}_l^j is evaluated by observing their intersection (Eq. (2)) and by taking into account the distribution ρ (Eq. (4)) of the cluster \mathcal{C}_k^i in all the clusters of \mathcal{R}^j as follows:

$$S(\mathcal{C}_k^i, \mathcal{C}_l^j) = \rho_k^{ij} \alpha_{l,k}^{ij}, \quad (3)$$

where

$$\rho_k^{ij} = \sum_{r=1}^{n_j} (\alpha_{k,r}^{ij})^2. \quad (4)$$

The entire collaborative clustering process is broken down in three main phases:

- (1) *Initial clusterings* – Each method computes its result independently.
- (2) *Results refinement* – A phase of convergence of the results, which consists of conflict evaluations and resolutions, is iterated as long as the quality of the results and their similarity increase:
 - (a) Detection of the conflicts, by evaluating the dissimilarities between couples of results;
 - (b) Local resolution of some conflicts;
 - (c) Global management of the local modifications in the global result (if they are relevant).
- (3) *Consensus computation* – The refined results are unified using a voting algorithm.

The entire algorithm of the method is detailed and explained in Algorithm 1.

Algorithm 1. Collaborative clustering process

```

1: Let  $\check{\mathcal{R}} = \{\mathcal{R}^i\}_{1 \leq i \leq m}$  be the initial set of clusterings
2: Let  $\check{\mathcal{K}} = \text{conflicts}(\check{\mathcal{R}})$  be the set of conflicts in  $\check{\mathcal{R}}$  as defined in Eq. (5)
3: Let  $\check{\mathcal{R}}^{\text{best}} := \check{\mathcal{R}}$  be the best temporary solution
4: Let  $\check{\mathcal{K}}^{\text{best}} := \check{\mathcal{K}}$  be the conflicts of the best temporary solution
5: while  $|\check{\mathcal{K}}| \geq 0$  do
6:    $\mathcal{K}_k^{ij} := \arg \max_{\mathcal{K}_l^{rs} \in \check{\mathcal{K}}} CI(\mathcal{K}_l^{rs})$ 
7:    $\check{\mathcal{R}} := \text{conflictResolution}(\check{\mathcal{R}}, \mathcal{K}_k^{ij})$  Algorithm 2
8:   if  $\Gamma(\check{\mathcal{R}}) > \Gamma(\check{\mathcal{R}}^{\text{best}})$  then
9:      $\check{\mathcal{R}}^{\text{best}} := \check{\mathcal{R}}$ 
10:     $\check{\mathcal{K}}^{\text{best}} := \check{\mathcal{K}} := \text{conflicts}(\check{\mathcal{R}})$ 
11:     $bt := 0$ 
12:   else if  $\check{\mathcal{R}}^{t+1} = \check{\mathcal{R}}^t$  then
13:      $\check{\mathcal{K}} := \check{\mathcal{K}} \setminus \mathcal{K}_k^{ij}$ 
14:   else
15:      $bt := bt + 1$ 
16:      $\check{\mathcal{K}} := \check{\mathcal{K}} \setminus \mathcal{K}_k^{ij}$ 
17:   if  $bt > |\check{\mathcal{K}}|$  then
18:      $\check{\mathcal{R}} := \check{\mathcal{R}}^{\text{best}}$ 
19:      $\check{\mathcal{K}} := \check{\mathcal{K}}^{\text{best}} \setminus \mathcal{K}_k^{ij}$ 

```

Algorithm (continued)**Algorithm 1.** Collaborative clustering process

```

20:   end if
21:   end if
22: end while
23: consensus computation

```

3.1.1. Initial clusterings

Each clustering method computes a clustering of the data using its initial parameters: all data objects are grouped into different clusters. According to the base method selected, different parameters need to be set.

3.1.2. Results refinement

3.1.2.1. Detection of the conflicts. The detection of the conflicts consists in seeking in \check{R} all the couples (C_k^i, R^j) , $i \neq j$, such as $S(C_k^i, CC(C_k^i, R^j)) < 1$, which means that the cluster C_k^i cannot be exactly found in the result R^j .

$$\text{conflicts}(\check{R}) = \left\{ (C_k^i, R^j) : i \neq j, S(C_k^i, CC(C_k^i, R^j)) < 1 \right\}. \quad (5)$$

Each conflict K_k^{ij} is identified by one cluster C_k^i and one result R^j . Its importance, $CI(K_k^{ij})$, is computed according to the inter-cluster similarity:

$$CI(K_k^{ij}) = 1 - S(C_k^i, CC(C_k^i, R^j)). \quad (6)$$

3.1.2.2. Local resolution of some conflicts. The conflict resolution algorithm is detailed precisely in Algorithm 2.

The most important conflict (i.e. having the greatest conflict importance) is selected in the set of existing conflicts according to the conflict importance coefficient (Eq. (6)).

The local resolution of a conflict K_k^{ij} consists in applying an operator on each result involved in the conflict, R^i and R^j , to try to improve their similarity. The operators which can be applied to a result are the following:

- **Merging** of clusters: some clusters are merged together;
- **Splitting** of a cluster into subclusters: a clustering is applied to the objects of a cluster to produce subclusters;
- **Reclustering** of a group of objects: one cluster is removed and its objects are reclassified in all the other existing clusters.

The operator to apply is chosen according to the number of clusters involved in the conflict, i.e. the number of clusters such as $S(C_k^i, C_l^j) > p_{cr}$, where $0 \leq p_{cr} \leq 1$ is given by the user. The p_{cr} parameter represents the percentage above which the intersection between the two clusters is considered as significant. For example, $p_{cr} = 0.2$ means that if $C_k^i \cup C_l^j$ represents less than 20% of the objects of C_k^i , C_l^j is not considered as a significant representative of C_k^i .

However, the application of the two operators (each one on a different result) is not always relevant. Indeed, it does not always increase the similarity of the results involved in the processed conflict. Moreover, the iteration of the conflict resolutions step may lead to a trivial but consensual solution. For example, the clusterings can converge towards a solution where all the results have only one cluster, including all the objects to classify, or towards a solution where all the results have one cluster for each object. These two solutions are not relevant and must be avoided.

So, we defined a criterion γ , called *local similarity criterion*, to evaluate the similarity between two results and their quality. It is based on the intercluster similarity S (Eq. (3)) and a quality criterion δ (detailed in Section 3.2, Eq. (23)). The criterion δ evaluates the quality of the clustering (e.g. inertia, number of clusters, ...) to avoid that the method ends up with a trivial solution as those presented before:

$$\gamma^{ij} = \frac{1}{2} \left(p_s \cdot \left(\frac{1}{n_i} \sum_{k=1}^{n_i} \omega_k^{ij} + \frac{1}{n_j} \sum_{k=1}^{n_j} \omega_k^{ji} \right) + p_q \cdot (\delta^i + \delta^j) \right), \quad (7)$$

where

$$\omega_k^{ij} = S(C_k^i, CC(C_k^i, R^j)) \quad (8)$$

and p_q and p_s are given by the user ($p_q + p_s = 1$).

Let $R^{i'}$ (resp. $R^{j'}$) be the result R^i (resp. R^j) after having applied the operators. The local similarity criterion is computed on each of the four couples of results: (R^i, R^j) , $(R^{i'}, R^j)$, $(R^i, R^{j'})$, $(R^{i'}, R^{j'})$. The best couple is accepted as the local solution of the conflict.

Algorithm 2. Conflict resolution

Require: \check{R} the ensemble of clusterings
Require: \mathcal{K}_k^{ij} the conflict to solve
Ensure: $\check{R}^* = \text{conflictResolution}(\check{R}, \mathcal{K}_k^{ij})$ the new ensemble after the resolution

let $\kappa = \left\{ C_l^i, \forall 1 \leq l \leq n_j : S(C_k^i, C_l^i) > p_{cr} \right\}$

if $|\kappa| > 1$ **then**

$\mathcal{R}'^i = \mathcal{R}^i \setminus \{C_k^i\} \cup \text{split}(C_k^i, |\kappa|)$

$\mathcal{R}'^j = \mathcal{R}^j \setminus \kappa \cup \text{merge}(\kappa, \mathcal{R}^j)$

else

$\mathcal{R}'^i = \text{recluster}(\mathcal{R}^i \setminus \{C_k^i\})$

end if

$\{\mathcal{R}^{i*}, \mathcal{R}^{j*}\} = \arg \max \gamma^{IJ}$ for $I \in \{i, i'\}, J \in \{j, j'\}$

$\check{R}^* = \check{R} \setminus \{\mathcal{R}^i, \mathcal{R}^j\} \cup \{\mathcal{R}^{i*}, \mathcal{R}^{j*}\}$

3.1.2.3. Global management of the local modifications. After the resolutions of the local conflicts, a global application of the modifications proposed by the refinement step is decided if their application improve the quality of the global result. The *global agreement coefficient* Γ is evaluated according to all the local similarities between each couple of results as follows:

$$\Gamma = \frac{1}{m} \sum_{i=1}^m \Gamma^i, \quad (9)$$

where

$$\Gamma^i = \frac{1}{m-1} \sum_{\substack{j=1 \\ j \neq i}}^m \gamma^{ij}. \quad (10)$$

Three cases can occur:

- The resolution step gives a better solution than all previous ones (line 8). In this case, the best temporary solution is the one proposed by the conflict resolution step. As the global results have changed, the conflicts list is recomputed (line 10).
- The resolution step proposes the same solution which means that no operators application is relevant to solve this conflict (line 12). Then, the conflict is removed from the list and the algorithm iterates.
- If the solution proposed by the conflict resolution gives a worth global agreement coefficient, it is accepted to avoid to fall in a local maximum (line 14). But, if no conflict resolution enables to find a better solution (after having resolved the first half part of the conflicts list), all the results are reinitialized to the best temporary solution (line 17).

The process is iterated until some conflicts still remain in the conflicts list (line 5).

3.1.3. Consensus computation

After the refinement step, all the results tend to have the same number of clusters, which should be similar. Thus, in a final step, we use an original voting algorithm to compute a consensus result from the different results. This multi-view voting algorithm enables to combine in one unique result, many different clusterings that do not have necessarily the same number of clusters.

The basic idea is that for each object to cluster, each result \mathcal{R}^i votes for the cluster it has found for this object, C_k^i for example, and for its corresponding cluster $CC(C_k^i, \mathcal{R}^j)$ in each other result \mathcal{R}^j . The maximum of these values indicates the *best cluster* for the object, for example C_l^j . It means that this object should be in the cluster C_l^j according to the opinion of the majority of the methods.

For each object p a voting matrix is computed as:

$$\mathcal{V}(p) = \left\{ \left(v_1^i(p), \dots, v_{n_i}(p) \right), 1 \leq i \leq m \right\}, \quad (11)$$

where

$$v_k^i(p) = \sum_{j=1}^m \text{vote}(p, C_k^i, \mathcal{R}^j) \quad (12)$$

and

$$\text{vote}(p, \mathcal{C}_k^i, \mathcal{R}^m) = \begin{cases} 1 & \text{if } (i = m \text{ and } p \in \mathcal{C}_k^i) \\ & \text{or } p \in CC(\mathcal{C}_k^i, \mathcal{R}^m) \\ 0 & \text{else} \end{cases} \quad (13)$$

The object p is then assigned to the cluster $\check{\mathcal{V}}$, defined as:

$$\check{\mathcal{V}}(p) = \arg \max_{\mathcal{C}_k^i} \nu_k^i(p). \quad (14)$$

3.2. Background knowledge integration

In this section, we explain how we integrated background or domain knowledge into our collaborative clustering process. The aim is to make the method able to deal with two types of constraints: class label-based constraints and relationship between objects-based constraints (also called link-constraints). In class label-based constraints, a small subset of labeled samples is available. The goal is to try to have only one class represented in each cluster. In link-constraints, the goal is to respect the constraints provided on the objects.

Firstly (Section 3.2.1), we will see some examples of background knowledge integration present in the literature and then (Section 3.2.2) we will see how we integrated it into the SAMARAH method.

3.2.1. Examples of knowledge integration

In the literature, different methods have already been proposed to take into account background knowledge. For example, to integrate link-constraints in the algorithm Pairwise Constrained KMEANS, Basu et al. [41] defines the following objective function:

$$obj_{pckm} = \sum_{j=1}^{n_k^i} \text{dispersion}(\mathcal{C}_j^i) + \sum_{(x_i, x_j) \in \mathbb{M}} [l_i \neq l_j] + \sum_{(x_i, x_j) \in \mathbb{C}} [l_i = l_j], \quad (15)$$

where \mathbb{M} is the set of must-link constraints and \mathbb{C} is the set of cannot-link constraints. This objective function includes a dispersion measure computed as the classical mean squared error (first term), but also two other components (second and third terms) reflecting the agreement according to the sets of available constraints (\mathbb{M} and \mathbb{C}). The functions $[l_i \neq l_j]$ and $[l_i = l_j]$ return 1 if the constraint between the couple of objects (x_i, y_j) is respected, and 0 else.

An approach is proposed by Demiriz et al. [24] to integrate class label-based constraints. This method uses a genetic algorithm to minimize an objective function, which is a geometric mean between cluster dispersion and cluster impurity according to available samples. It is defined as:

$$obj_{gen} = \sum_{j=1}^{n_k^i} (\alpha \times \text{dispersion}(\mathcal{C}_j^i) + \beta \times \text{impurity}(\mathcal{C}_j^i)), \quad (16)$$

where the cluster dispersion is usually the mean squared error, and the cluster impurity is a measure of the impurity of a cluster according to its composition of available labeled samples. This impurity measure is low if all the known samples of a cluster are from the same class. On the contrary, the value increases as the cluster contains objects from various classes (and the cluster will be considered as impure). This impurity is evaluated thanks to the Gini index:

$$\text{gini}(\mathcal{C}_k^i) = 1 - \sum_{l=0}^{n_c} \left(\frac{P_{kl}}{n_k^i} \right)^2, \quad (17)$$

where P_{kl} is the number of objects belonging to the l th class in the cluster \mathcal{C}_k^i , and n_k^i is the number of objects in the cluster \mathcal{C}_k^i . The weights α and β in Eq. (16) allow the user to choose if he prefers to give more or less importance to the available knowledge.

In supervised clustering [42], a similar idea is used. A penalty measure is added to the impurity measure to deal with the case of various number of clusters in the results and avoid results with very high or very low number of clusters:

$$obj_{sc} = \sum_{j=1}^{n_k^i} (\alpha \times \text{impurity}(\mathcal{C}_j^i) + \beta \times \text{penalty}(\mathcal{C}_j^i)), \quad (18)$$

where the impurity measure is defined by:

$$\text{impurity}(\mathcal{C}_k^i) = \frac{1}{n} \sum_{\substack{l=0 \\ l \neq cmax}}^{n_c} P_{kl}, \quad (19)$$

where P_{kl} is the number of objects belonging to the l th class in the cluster \mathcal{C}_k^i , n_k^i is the number of objects in the cluster \mathcal{C}_k^i , and $cmax$ is the most represented class in the cluster \mathcal{C}_k^i

$$cmax = \arg \max_l (P_{kl}). \quad (20)$$

The penalty is defined as

$$\text{penalty}(\mathcal{C}_k) = \begin{cases} \sqrt{\frac{n_k - n_c}{n}} & \text{if } n_k \geq n_c, \\ 0 & \text{otherwise,} \end{cases} \quad (21)$$

where n_c is the number of classes in the known samples. Here again, the weights α and β in Eq. (18) are chosen by the user.

3.2.2. Knowledge integration in the SAMARAH method

In the SAMARAH method, during the refinement step, the local similarity criterion γ^{ij} (Eq. (7)) is used to evaluate if the modifications of a couple of results is relevant. This criterion includes a quality criterion δ^i which represents the quality of the result \mathcal{R}^i . This criterion is used to balance the refinement step between the similarity and the quality of the expected results. It is computed for two aspects of the results: the internal and external qualities. The internal evaluation consists in evaluating the quality of the result through a unsupervised measure. The external evaluation consists in evaluating the quality of a result thanks to external knowledge, such as an estimation of the number of clusters, some labeled samples or some constraints.

The previous version of SAMARAH already includes internal knowledge but only includes an estimation of the number of clusters as external knowledge. To take into account more external knowledge, we have extended the quality criterion by integrating an evaluation of the agreement of the results with different kinds of constraints as follows:

$$\delta^i = \sum_{c=1}^{N_c} q_c(\mathcal{R}^i) \times p_c, \quad (22)$$

where N_c is the number of constraints to respect, q_c is the criterion used to evaluate the result according to the c th constraint ($q_c(\cdot) \in [0, 1]$) and p_c is the relative importance given by the user to the c -th constraint ($p_1 + p_2 + \dots + p_{N_c} = 1$). By default, each constraint is given with a weight of $\frac{1}{N_c}$.

Thus, any constraint can be integrated in the process if it can be defined as a function taking its values in $[0, 1]$. We described below some frequently encountered constraints that can be used.

3.2.2.1. Cluster quality-based constraints. These constraints are based on the intrinsic quality of the clusters such as inertia or predictivity and also take into account the number of clusters. Indeed criterion such as inertia or compacity need to be balanced with an evaluation of the number of clusters. An example of a criterion which includes quality of the clusters and the number of clusters is given below:

$$q_{qb}(\mathcal{R}^i) = \frac{p^i}{n^i} \sum_{k=1}^{n^i} \tau_k^i, \quad (23)$$

where n^i is the number of clusters of \mathcal{R}^i , τ_k^i defines the internal quality of the k th cluster and p^i is the external quality of the result. The internal quality of the k th cluster is given as:

$$\tau_k^i = \begin{cases} 0 & \text{if } \frac{1}{n_k^i} \sum_{l=1}^{n_k^i} \frac{d(x_{k,l}^i, g_k^i)}{d(x_{k,l}^i, g^i)} > 1, \\ 1 - \frac{1}{n_k^i} \sum_{l=1}^{n_k^i} \frac{d(x_{k,l}^i, g_k^i)}{d(x_{k,l}^i, g^i)} & \text{else,} \end{cases} \quad (24)$$

where n_k^i is the cardinal of \mathcal{C}_k^i , g_k^i is the gravity center of \mathcal{C}_k^i , g^i is the gravity center of another cluster, the closest from $x_{k,l}^i$ and d is the distance function. The measure is computed on each cluster to evaluate the overall quality of the clustering result. To take into account the number of clusters n^i , the criterion p^i is defined as:

$$p^i = \frac{n_{\text{sup}} - n_{\text{inf}}}{|n_i - n_{\text{inf}}| + |n_{\text{sup}} - n_i|}, \quad (25)$$

where $[n_{\text{inf}}, n_{\text{sup}}]$ is given by the user, as the range of expected number of clusters.

3.2.2.2. Class label-based constraints. These constraints correspond to the case where a few sets of labeled samples are available. To evaluate the agreement between results and such constraints, we can use any index which enables us to evaluate the similarity between a clustering and a labeled classification (where all the classes are known, and each object belongs to one of these classes). In our case, we only compare results with a given partial partition \mathbb{R} which represents the known labeled

objects. In the implementation of the SAMARAH method, we mainly used the Rand index [49] and another index known as WG agreement index [52]. Information theoretic measures could also be used [53].

The Rand index is a measure of the similarity between two data partitions defined by:

$$Rand(\mathcal{R}^i, \mathbb{R}) = \frac{a + b}{\binom{n}{2}}, \quad (26)$$

where n is the number of objects to classify, a is the number of pairs of objects which are in the same cluster in \mathcal{R}^i and in the known result, and b is the number of pairs of objects which are not in the same cluster in the proposed result \mathcal{R}^i and in the known result \mathcal{R}^j . The sum of these two measurements (a and b) can be seen as the number of times that the two partitions are in agreement. This index takes values in [0,1]: 1 indicates that the two partitions are identical. The defined constraint is:

$$q_{rand}(\mathcal{R}^i) = Rand(\mathcal{R}^i, \mathbb{R}). \quad (27)$$

The WG agreement index is defined by

$$WG(\mathcal{R}^i, \mathbb{R}) = \frac{1}{n} \sum_{k=1}^{n_i} S(C_k^i, \mathcal{R}^j) |C_k^i|, \quad (28)$$

where n is the number of objects to classify and \mathcal{R}^j is the reference partition (e.g. labeled classification, another clustering, etc.). This index takes values in [0,1]: 1 indicates that all the objects in the clustering \mathcal{R}^i are well classified according to the class label of the objects in \mathcal{R}^j . The defined constraint is:

$$q_{wg}(\mathcal{R}^i) = WG(\mathcal{R}^i, \mathbb{R}). \quad (29)$$

3.2.2.3. Link-based constraints. These constraints correspond to the case where knowledge is expressed as *must-link* and *cannot-link* constraints between objects (see [30,31]). In this case, the ratio of respected constraints against violated constraints can easily be computed as

$$q_{link}(\mathcal{R}^i) = \frac{1}{n_r} \sum_{j=1}^{n_r} v(\mathcal{R}^i, l_j), \quad (30)$$

where n_r is the number of constraints between the objects, l_j is a *must-link* or *cannot-link* constraint and $v(\mathcal{R}^i, l_j) = 1$ if \mathcal{R}^i respects the constraint l_j , 0 otherwise.

Note that such constraints can be extracted from class-label constraints. For example, a must-link constraint could be created for all the couples of objects belonging to the same cluster, and a cannot-link constraint could be created for all the couples of objects belonging to different clusters.

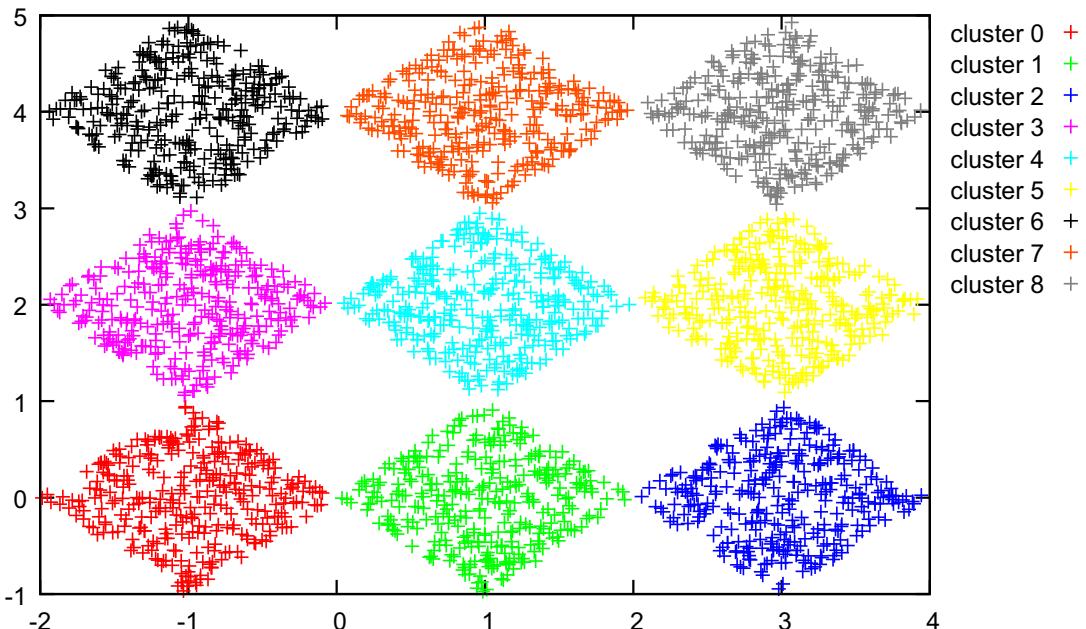


Fig. 1. 9-Diamonds dataset.

3.3. Example

In this section, we present a simple example on a 2D dataset. The aim is to illustrate how the different kinds of knowledge presented in the previous section can improve the collaboration between the clustering methods. We used the 9-Diamonds dataset (Fig. 1) from [54] which is available for download on the website of the machine learning group of the University of Houston.¹

We used the KMEANS algorithm to cluster this dataset in various number of clusters ranging from 2 to 18 (9 being the actual number of cluster). Then we considered couple of clustering results and we computed for each couple a value of agreement between the two clustering results. The goal is illustrate the search-space of the different possible solutions (i.e. couple of clustering results). In the followings figures (Figs. 2–5) the higher the values are, the higher the agreement is. We illustrated here how different kinds of knowledge can modify the shape of the search-space and consequently, help the collaboration.

Fig. 2 presents the local agreement using only the similarity of the result (7). One can see that using only the similarity creates several local minima in this search-space as the results with a low number of clusters are strongly similar. To reduce this problem, the knowledge of the range of expected number of clusters can be used.

Fig. 3 presents the local agreement using the similarity along with some knowledge about the number of clusters (25) (i.e. set here to [7; 11]). One can see that the search-space leverages this information and that the value for the number of cluster outside the range are strongly reduced.

If some labeled patterns are available, a measure of quality of the clustering can be added to the evaluation to guide the collaboration. This is illustrated on Fig. 4 where the WG index (29) is computed assuming 5% of labeled objects. On can see that this information improves substantially the shape of the search space.

Finally, different kinds of knowledge can be used together as in Fig. 5 where the range of expected number of clusters along with the knowledge of some labeled objets are used. The resulting search-space is cleary easier to explore and contains less local minimas, the optimal solution (nine clusters) being strongly highlighted.

4. Evaluation of the proposed method

4.1. Protocol of experiments

In this section, we present two evaluations of the method proposed in this paper. The aim of these experiments is twofold. Firstly, we want to show the relevance of collaborative clustering to improve the quality of a set of clustering results, thanks to our collaborative process. Secondly, we want to show the interest in integrating background knowledge in this collaboration, to produce even better results.

Two kinds of experiment have been carried out:

- (1) The first one consisted in the evaluation of the quality of a set of clustering results, first, without collaboration, then with collaboration, and finally, with collaboration integrating background knowledge. Different classical quality indexes were used to evaluate the quality of these sets of clustering results. The details of these experiments are described in Section 4.3.
- (2) The second one consisted in the evaluation using the Cascade Evaluation approach [7]. This approach is based on the enrichment of a set of datasets by the clustering results, and the use of a supervised method to evaluate the interest of adding such new information to the datasets. The details of these experiments are described in Section 4.4.

For all the experiments with SAMARAH, we used the KMEANS algorithm [55] as base clustering method (Section 3.1). Five methods were randomly initialized with a number of clusters randomly picked in [2;10]. The refinement step was set up to find results with a number of clusters in [2; 10] (i.e. $n_{\inf} = 2, n_{\sup} = 10$ in Eq. (25)).

To evaluate benefits of the background knowledge integration in the refinement step as presented in Section 3.2, we randomly picked up 10% of the datasets as a subset of available samples. This subset was used to drive the refinement step through the quality evaluation of the results. The datasets used in both the experiments are presented in the next section.

4.2. Data sets

Seven different datasets from the UCI machine learning repository [56] were used in the experiments:

- (1) Iris data set, which contains three classes of 50 instances, where each class refers to a type of iris plant;
- (2) Wine database, which contains three classes of wines, characterized by 13 chemicals attributes (178 instances);
- (3) Ionosphere database, which consists in 351 information about 16 high-frequency antennas classified into two classes;
- (4) Pima Indians Diabetes database, referred as Pima, which consists in 768 patients discriminated into two classes according to World Health Organization criteria;

¹ <http://www.tlc2.uh.edu/dmmlg>.

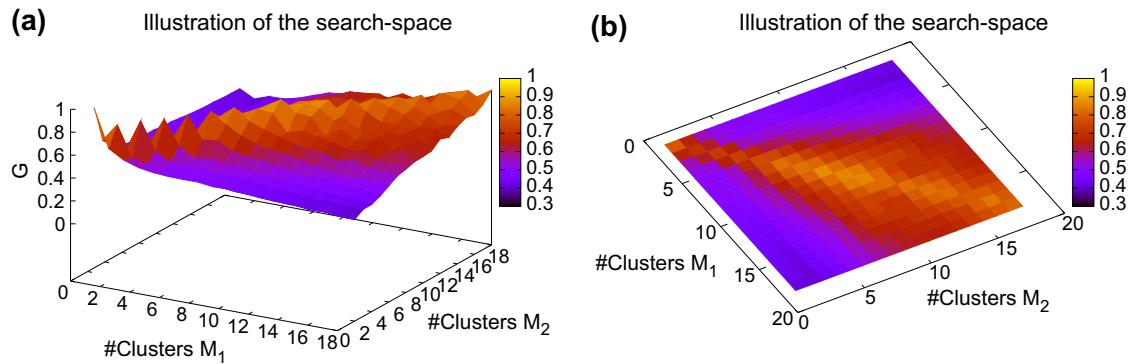


Fig. 2. The search-space using only the similarity.

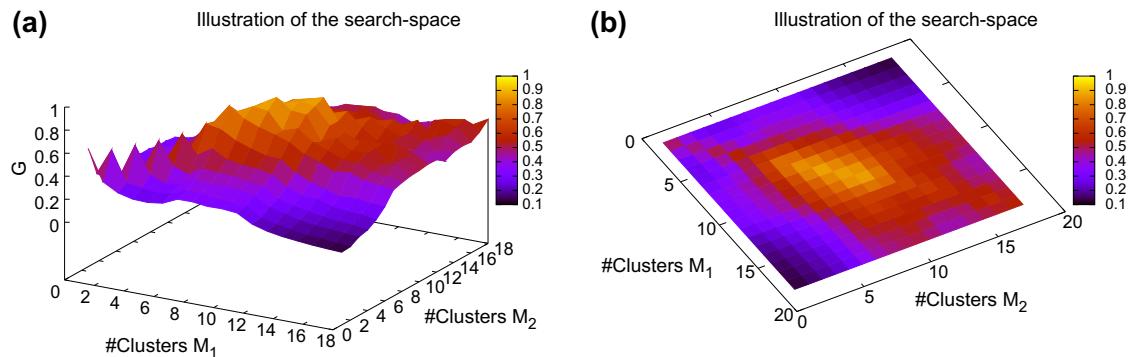


Fig. 3. The search-space using the similarity and the range of expected number of clusters.

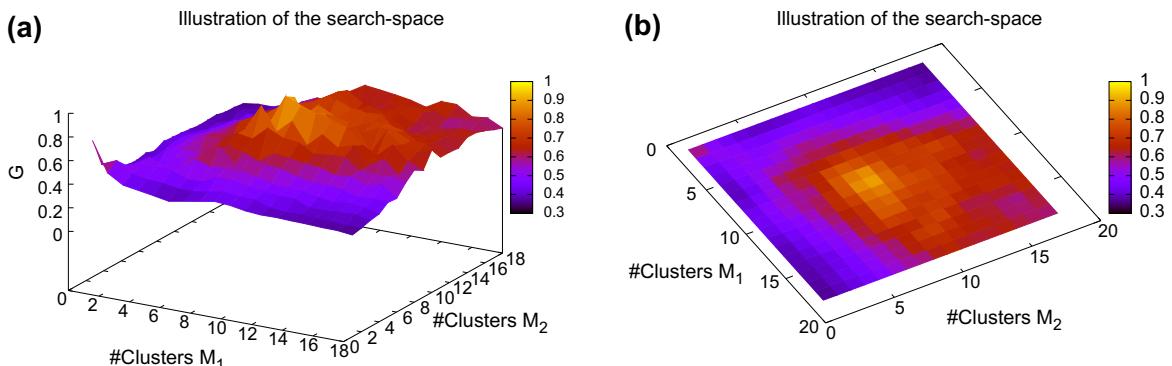


Fig. 4. The search-space using the knowledge of some labeled patterns.

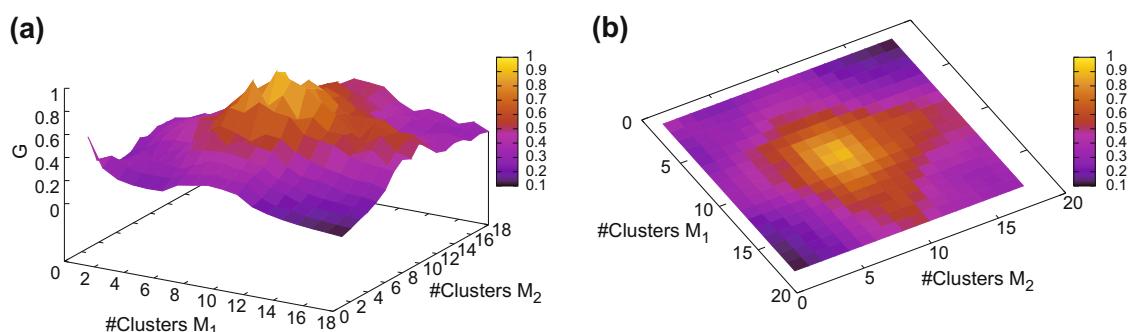


Fig. 5. The search-space using the range of expected number of clusters and the knowledge of labeled patterns.

- (5) Sonar, which has been used to learn to discriminate between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock (two classes, 307 instances), using 60 real attributes (each attribute represents the energy within a particular frequency band);
- (6) Vehicle data set, which contains four classes of 946 vehicles to classify given a set of 18 features extracted from their silhouette at different angles;
- (7) Segment database, composed of 2500 instances of 3×3 regions extracted from seven images of texture; each region is characterized by 19 attributes.

4.3. Comparison using quality indexes

In this first experiment, we evaluated the quality of the sets of clustering results, without collaboration (referred as $\neg\text{col}$), refined with collaboration (referred as col), and refined with collaboration using background knowledge (referred as $k\text{col}$). We used six different quality indexes to evaluate the quality of each set:

- The Rand index [49].
- The Jaccard index [50].
- The Falks–Mallow index [51].
- The Wemmert–Gançarski index [52].
- The F-Measure index [51].
- The Kappa index [48].

For each set of clusterings ($\neg\text{col}$, col and $k\text{col}$), the mean of the quality of each element of the set was computed, to define the quality of the set. We carried out 100 runs computing at each run the set of initial clusterings (with the parameters defined in Section 4.1) and the refined set through the collaborative process and through the collaborative process using background knowledge. The evaluation of the results are given in Table 1, where the values correspond to the means, the standard deviations and the maximum accuracy of the results on the 100 runs. On Fig. 6, we illustrate the results for the Rand index for all the experiments and all the datasets.

As one can see, the quality of the refined set (col) is almost always better than the quality of the unrefined set ($\neg\text{col}$). Indeed, for each dataset the quality is better according to at least five out of the six quality indexes. Furthermore, the quality of the sets provided by the collaboration process using background knowledge ($k\text{col}$) gives even better results.

All these results show, firstly, that the refinement step of our collaborative clustering method improve the quality of the results, and secondly, that the use of background knowledge in the collaboration helps the process to produce better results.

4.4. Comparison using cascade evaluation

The cascade evaluation [7] is a new approach to evaluate the quality and the interest of clustering results. The method is based on the enrichment of a set of datasets by the results of clustering, and the use of a supervised method to evaluate the interest of adding such new information to the datasets.

The method consists in evaluating and comparing the result of a supervised classifier when it is helped or not by the information issued from a clustering. If the result of the classifier is improved by the information added by the clustering, the authors assume that the clustering embeds a meaningful information. Furthermore, different clustering results can be compared, the one improving the most the result according to the classifier accuracy is consequently the one embedding the most interesting information.

We used this paradigm to conduct a cascade evaluation of the collaborative clustering. The aim of this evaluation is to show the relevance of the refinement step presented above. We want to see if the refinement improves the different results through the collaborative step. We are consequently interested to show that the set of refined results contains a more accurate knowledge compared to the initial unrefined set of results.

To evaluate the benefits of using the information provided by our method in supervised algorithms, we created different datasets from the initial one, and then we classified them with a supervised algorithm. Let D be the initial dataset to classify : $D = \{o_i\}_{1 \leq i \leq l}$ where the object o_i is characterized by the m attributes $A_1(o_i), \dots, A_m(o_i)$. Let $C_j(o_i)$ be the cluster of the object o_i in the j th *initial* clustering result. Let $R_j(o_i)$ be the cluster of the object o_i in the j th *refined* clustering result. Let $K_j(o_i)$ be the cluster of the object o_i in the j th *refined with knowledge* clustering result. From each initial dataset D , three datasets were created to integrate knowledge provided from the different clusterings:

- $D^1 = \{O_i^1\}_{1 \leq i \leq l}$ where $O_i^1 = (A_1(o_i), \dots, A_m(o_i), C_1(o_i), \dots, C_n(o_i))$.
- $D^2 = \{O_i^2\}_{1 \leq i \leq l}$ where $O_i^2 = (A_1(o_i), \dots, A_m(o_i), R_1(o_i), \dots, R_n(o_i))$.
- $D^3 = \{O_i^3\}_{1 \leq i \leq l}$ where $O_i^3 = (A_1(o_i), \dots, A_m(o_i), K_1(o_i), \dots, K_n(o_i))$.

Table 1Quality of the results of the different sets of results $\neg\text{col}$, col and $k\text{col}$.

		Rand	jacc	FM	WG	F-M	K
iris	($\neg\text{col}$)	0.73(± 0.03) \diamond 0.78	0.44(± 0.02) \diamond 0.46	0.64(± 0.02) \diamond 0.65	0.54(± 0.03) \diamond 0.59	0.60(± 0.01) \diamond 0.62	0.43(± 0.01) \diamond 0.45
	(col)	0.85(± 0.00) \diamond 0.87	0.64(± 0.01) \diamond 0.67	0.78(± 0.00) \diamond 0.80	0.69(± 0.03) \diamond 0.72	0.78(± 0.01) \diamond 0.80	0.67(± 0.01) \diamond 0.70
	($k\text{col}$)	0.86(± 0.01) 0.87	0.65(± 0.02) 0.68	0.79(± 0.01) 0.81	0.71(± 0.03) 0.74	0.79(± 0.02) 0.81	0.69(± 0.02) 0.72
wine	($\neg\text{col}$)	0.68(± 0.09) \diamond 0.76	0.46(± 0.05) \diamond 0.55	0.66(± 0.03) \diamond 0.72	0.59(± 0.04) \diamond 0.66	0.62(± 0.04) \diamond 0.69	0.39(± 0.10) \diamond 0.52
	(col)	0.88(± 0.04) \diamond 0.94	0.71(± 0.09) \diamond 0.83	0.83(± 0.06) \diamond 0.91	0.75(± 0.06) \diamond 0.83	0.83(± 0.06) \diamond 0.91	0.74(± 0.10) \diamond 0.86
	($k\text{col}$)	0.90(± 0.02) 0.94	0.76(± 0.05) 0.83	0.86(± 0.03) 0.91	0.79(± 0.07) 0.87	0.86(± 0.03) 0.91	0.78(± 0.04) 0.86
ionosphere	($\neg\text{col}$)	0.56(± 0.01) \diamond 0.58	0.34(± 0.03) \diamond 0.38	0.53(± 0.03) \diamond 0.57	0.30(± 0.03) 0.32	0.50(± 0.04) \diamond 0.55	0.14(± 0.02) \diamond 0.16
	(col)	0.59(± 0.03) 0.61	0.34(± 0.08) \diamond 0.41	0.53(± 0.07) \diamond 0.59	0.21(± 0.05) \diamond 0.28	0.50(± 0.09) \diamond 0.58	0.20(± 0.04) \diamond 0.23
	($k\text{col}$)	0.59(± 0.01) 0.60	0.37(± 0.03) 0.39	0.55(± 0.02) 0.57	0.22(± 0.03) \blacklozenge 0.26	0.54(± 0.03) 0.56	0.21(± 0.02) 0.22
pima	($\neg\text{col}$)	0.48(± 0.00) \diamond 0.49	0.22(± 0.02) \diamond 0.25	0.38(± 0.02) \diamond 0.41	0.17(± 0.01) \diamond 0.18	0.35(± 0.03) \diamond 0.39	0.40(± 0.01) 0.41
	(col)	0.49(± 0.00) \diamond 0.50	0.29(± 0.00) 0.30	0.46(± 0.00) 0.46	0.20(± 0.01) 0.21	0.45(± 0.01) 0.46	0.37(± 0.00) \diamond 0.38
	($k\text{col}$)	0.50(± 0.00) 0.50	0.29(± 0.01) 0.30	0.46(± 0.01) 0.47	0.20(± 0.01) 0.20	0.45(± 0.01) 0.46	0.37(± 0.00) \blacklozenge 0.38
sonar	($\neg\text{col}$)	0.51(± 0.01) 0.52	0.25(± 0.04) \diamond 0.27	0.41(± 0.04) \diamond 0.47	0.22(± 0.04) 0.29	0.39(± 0.05) \blacklozenge 0.45	0.02(± 0.01) 0.04
	(col)	0.51(± 0.00) 0.51	0.26(± 0.02) \diamond 0.28	0.42(± 0.02) \diamond 0.44	0.20(± 0.01) \diamond 0.21	0.41(± 0.02) \diamond 0.43	0.01(± 0.01) \diamond 0.02
	($k\text{col}$)	0.51(± 0.01) 0.52	0.31(± 0.05) 0.37	0.47(± 0.06) 0.54	0.16(± 0.03) \blacklozenge 0.18	0.47(± 0.06) 0.54	0.01(± 0.01) \blacklozenge 0.04
vehicle	($\neg\text{col}$)	0.57(± 0.06) \diamond 0.65	0.21(± 0.01) \diamond 0.22	0.37(± 0.02) \diamond 0.40	0.25(± 0.05) \diamond 0.31	0.35(± 0.01) \diamond 0.36	0.50(± 0.10) \diamond 0.62
	(col)	0.58(± 0.07) \diamond 0.69	0.22(± 0.01) 0.24	0.38(± 0.02) 0.40	0.25(± 0.06) 0.22	0.35(± 0.01) \diamond 0.39	0.51(± 0.12) \diamond 0.68
	($k\text{col}$)	0.65(± 0.02) 0.66	0.22(± 0.01) 0.24	0.37(± 0.01) \blacklozenge 0.39	0.16(± 0.01) \blacklozenge 0.17	0.37(± 0.01) 0.39	0.63(± 0.02) 0.64
segment	($\neg\text{col}$)	0.65(± 0.06) \diamond 0.71	0.30(± 0.03) \diamond 0.34	0.49(± 0.03) \diamond 0.53	0.46(± 0.03) \diamond 0.49	0.45(± 0.04) \diamond 0.49	0.61(± 0.07) \diamond 0.68
	(col)	0.83(± 0.02) \diamond 0.85	0.38(± 0.03) \diamond 0.41	0.56(± 0.03) \diamond 0.59	0.48(± 0.05) \diamond 0.54	0.55(± 0.03) \diamond 0.58	0.83(± 0.03) 0.84
	($k\text{col}$)	0.84(± 0.03) 0.87	0.39(± 0.03) 0.42	0.58(± 0.03) 0.60	0.52(± 0.07) 0.60	0.56(± 0.03) 0.60	0.83(± 0.03) 0.87

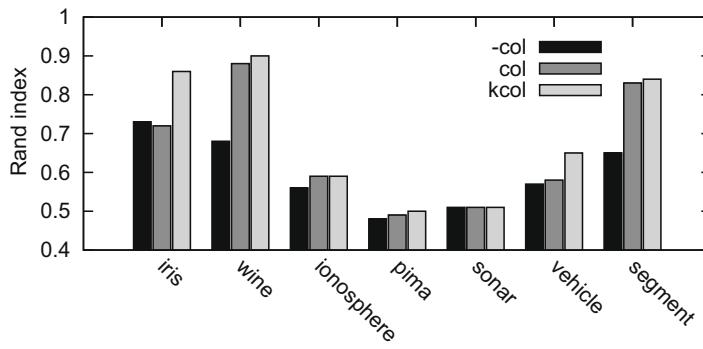


Fig. 6. Rand quality index for the three experiments ($\sim\text{col}$, col and kcol) for all datasets.

Table 2

Evaluation of the integration of background knowledge using cascade evaluation.

	D	D^1	D^2	D^3
iris	93.33%	94.67%(± 0.47) \diamond 94.67	95.47%(± 0.30) \diamond 96.00	96.40%(± 0.60) \diamond 96.67
wine	92.13%	93.48%(± 1.52) \diamond 95.51	95.73%(± 0.50) \diamond 96.07	96.18%(± 0.73) \diamond 97.19
ionosphere	88.03%	89.00%(± 1.25) \diamond 90.60	89.40%(± 0.93) \diamond 90.88	90.94%(± 0.31) \diamond 91.17
pima	63.41%	64.35%(± 0.57) \diamond 65.23	64.92%(± 0.76) \diamond 65.89	65.89%(± 0.95) \diamond 67.45
sonar	71.15%	70.58%(± 0.79) \diamond 71.63	71.44%(± 0.87) \diamond 71.63	72.50%(± 0.86) \diamond 74.04
vehicle	69.47%	69.17%(± 0.87) \diamond 70.35	69.77%(± 0.91) \diamond 71.23	70.55%(± 0.82) \diamond 71.61
segment	95.93%	95.79%(± 0.17) \diamond 96.02	95.77%(± 0.15) \diamond 95.97	95.79%(± 0.14) \diamond 95.97

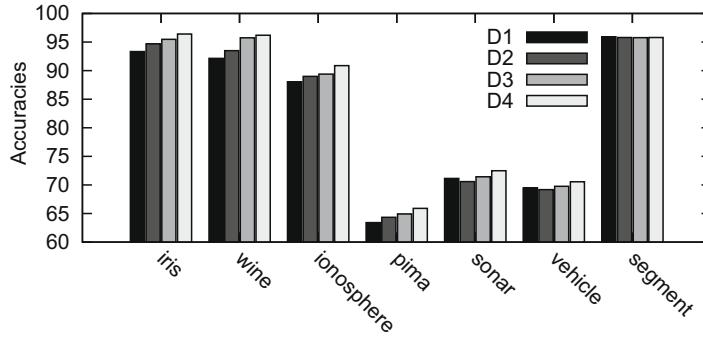


Fig. 7. Accuracies of cascade evaluation for all datasets.

As supervised algorithm, we chose the tree-based classifier C4.5 [57] for his ability to handle numeric and categorical attributes, and we made 100 runs of 10-fold cross validations, each on the three versions of each dataset.

The results of this evaluation are presented in Table 2 where the values are the average values, the standard deviations and the maximum values of the accuracy for the four versions of each dataset (i.e. the normal dataset D , the one embedded with the clustering D^1 , the one embedded with the refined clustering D^2 , and the one embedded with the refined clustering integrating background knowledge D^3). The histograms of the accuracies are presented on Fig. 7. One can see that the datasets embedded by the refined clusterings gives the best results on six of the seven datasets. One can notice that the refinement step degrades the results without clustering only when the initial clustering results also degrade the result. This can be explained by the lack of concordance between the class of the objects and their distribution in the data space. Consequently, adding the clustering information just added noise to the dataset.

Furthermore, one can observe an increase of the stability, as the standard deviation significantly decreases, when the refined results are used instead of the initial ones. The results refined using background knowledge are better in means than the results obtained without it. However, the results are less stable (higher standard deviation) on the half of the datasets. This can be explained by the high degree of randomness in the selection of the samples used as background knowledge. If these samples are well distributed on the data space and among the different clusters, they will carry a better information and will make helping the collaboration easier. We assume that this issue can be solved if the samples are actually provided by the expert itself (and not selected randomly). The expert should be able to provide high quality examples. An active learning approach could also be used where the expert could, during the collaboration, gives information about the clusters involved in strong conflict.

5. Conclusion

In many clustering problems, the user is able to provide some background knowledge which can be used to guide the algorithm to obtain more accurate results. Moreover, it has been accepted that ensemble clustering algorithms give more

robust results to such problems. Unfortunately, no ensemble clustering method that includes information given by the user has been defined yet.

In this article, we have presented a new method of collaborative clustering, that integrates and benefits from background knowledge on the given problem. The user can express the knowledge through different kinds of constraints. To illustrate this, we have proposed a formalization of three main types of constraints: cluster quality, class label and link-based constraints. Then, we presented how this information can be used during the collaboration step, to guide the different methods in their search for a better solution.

Finally, we have shown by different experiments that the collaboration between the clustering methods provides better results than the single classical method, and that introducing knowledge to control the collaboration gives even more accurate results.

References

- [1] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, *ACM Computing Surveys* 31 (3) (1999) 264–323.
- [2] A. Rauber, E. Pampalk, J. Paralic, Empirical evaluation of clustering algorithms, *Journal of Information and Organizational Sciences (JIOS)* 24 (2) (2000) 195–209.
- [3] P. Berkhin, Survey of clustering data mining techniques, Technical Report, Accrue Software, San Jose, CA, 2002.
- [4] R. Xu, D. Wunsch, Survey of clustering algorithms, *IEEE Transactions on Neural Networks* 16 (3) (2005) 645–678.
- [5] Cen Li, Gautam Biswas, Unsupervised learning with mixed numeric and nominal data, *IEEE Transactions on Knowledge and Data Engineering* 14 (4) (2002) 673–690.
- [6] J. Kittler, On combining classifiers, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (3) (1998) 226–239.
- [7] L. Candillier, I. Tellier, F. Torre, O. Bousquet, Cascade evaluation of clustering algorithms, *European Conference on Machine Learning*, vol. 4212, Springer, Berlin/Heidelberg, 2006, pp. 574–581.
- [8] A. Strehl, J. Ghosh, Cluster ensembles – a knowledge reuse framework for combining multiple partitions, *Journal on Machine Learning Research* 3 (2002) 583–617.
- [9] A.I.N. Fred, A.K. Jain, Combining multiple clusterings using evidence accumulation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (6) (2005) 835–850.
- [10] A.P. Topchy, A.K. Jain, W.F. Punch, Combining multiple weak clusterings, in: *International Conference on Data Mining*, IEEE Computer Society, 2003, pp. 331–338.
- [11] A. Topchy, A.K. Jain, W. Punch, Clustering ensembles: models of consensus and weak partitions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (12) (2005) 1866–1881.
- [12] S.T. Hadjitodorov, L.I. Kuncheva, Selecting diversifying heuristics for cluster ensembles, in: *International Workshop on Multiple Classifier Systems*, vol. 4472, 2007, pp. 200–209.
- [13] H. Ayad, M.S. Kamel, Cumulative voting consensus method for partitions with variable number of clusters, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (1) (2008) 160–173.
- [14] N. Nguyen, R. Caruana, Consensus clusterings, in: *International Conference on Data Mining*, IEEE Computer Society, 2007, pp. 607–612.
- [15] B. Long, Z. Zhang, P.S. Yu, Combining multiple clusterings by soft correspondence, in: *International Conference on Data Mining*, IEEE Computer Society, 2005, pp. 282–289.
- [16] T. Hu, Y. Yu, J. Xiong, S.Y. Sung, Maximum likelihood combination of multiple clusterings, *Pattern Recognition Letters* 27 (13) (2006) 1457–1464.
- [17] W. Pedrycz, K. Hirota, A consensus-driven fuzzy clustering, *Pattern Recognition Letters* 29 (9) (2008) 1333–1343.
- [18] J. Handl, J. Knowles, An evolutionary approach to multiobjective clustering, *IEEE Transactions on Evolutionary Computation* 11 (1) (2007) 56–76.
- [19] A. Konak, D. Coit, A. Smith, Multi-objective optimization using genetic algorithms: a tutorial, *Reliability Engineering & System Safety* 91 (9) (2006) 992–1007.
- [20] J. Handl, J.D. Knowles, On semi-supervised clustering via multiobjective optimization, in: *Genetic and Evolutionary Computation Conference*, 2006, pp. 1465–1472.
- [21] K. Faceli, A. de Carvalho, M. de Souto, Multi-objective clustering ensemble, in: *International Conference on Hybrid Intelligent Systems*, 2006, pp. 51–51.
- [22] M.H. Law, A. Topchy, A.K. Jain, Multiobjective data clustering, in: *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 424–430.
- [23] K. Faceli, A.C.P. Ferreira de Carvalho, M.C. Pereira de Souto, Multi-objective Clustering Ensemble with Prior Knowledge, vol. 4643, Springer, 2007. pp. 34–45.
- [24] A. Demiriz, K.P. Bennett, M.J. Embrechts, Semi-supervised clustering using genetic algorithms, *Artificial neural networks in engineering (ANNIE-99)* (1999) 809–814.
- [25] W. Pedrycz, Collaborative fuzzy clustering, *Pattern Recognition Letters* 23 (2002) 1675–1686.
- [26] W. Pedrycz, P. Rai, A multifaceted perspective at data analysis: a study in collaborative intelligent agents, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 38 (4) (2008) 1062–1072.
- [27] V. Loia, W. Pedrycz, S. Senatore, Semantic web content analysis: a study in proximity-based collaborative clustering, *IEEE Transactions on Fuzzy Systems* 15 (6) (2007) 1294–1312.
- [28] H. Mitra, Banka, W. Pedrycz, Rough-fuzzy collaborative clustering, *IEEE Transactions on Systems, Man, and Cybernetics* 36 (2006) 795–805.
- [29] Grigoris Tsoumacas, Lefteris Angelis, Ioannis Vlahavas, Clustering classifiers for knowledge discovery from physically distributed databases, *Data & Knowledge Engineering* 49 (3) (2004) 223–242.
- [30] K. Wagstaff, C. Cardie, S. Rogers, S. Schrodl, Constrained K-means clustering with background knowledge, in: *International Conference on Machine Learning*, 2001, pp. 557–584.
- [31] M. Bilenko, S. Basu, R.J. Mooney, Integrating constraints and metric learning in semi-supervised clustering, in: *International Conference on Machine Learning*, 2004, pp. 81–88.
- [32] Ruizhang Huang, Wai Lam, An active learning framework for semi-supervised document clustering with language modeling, *Data & Knowledge Engineering* 68 (1) (2009) 49–67.
- [33] Nizar Grira, Michel Crucianu, Nozha Boujemaa, Active semi-supervised fuzzy clustering, *Pattern Recognition* 41 (5) (2008) 1851–1861.
- [34] I. Davidson, K.L. Wagstaff, S. Basu, Measuring constraint-set utility for partitional clustering algorithms, in: *European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2006, pp. 115–126.
- [35] K.L. Wagstaff, Value, cost, and sharing: Open issues in constrained clustering, in: *International Workshop on Knowledge Discovery in Inductive Databases*, 2007, pp. 1–10.
- [36] Nimit Kumar, Krishna Kummamuru, Semisupervised clustering with metric learning using relative comparisons, *IEEE Transactions on Knowledge and Data Engineering* 20 (4) (2008) 496–503.
- [37] D. Klein, S. Kamvar, C. Manning, From instance-level constraints to space-level constraints: making the most of prior knowledge in data clustering, in: *The 19th International Conference on Machine Learning*, 2002, pp. 307–314.

- [38] S. Basu, A. Banerjee, R.J. Mooney, Active semi-supervision for pairwise constrained clustering, in: SIAM International Conference on Data Mining, 2004, pp. 333–344.
- [39] S. Basu, A. Banerjee, R.J. Mooney, Semi-supervised clustering by seeding, in: International Conference on Machine Learning, 2002, pp. 19–26.
- [40] J. Gao, P. Tan, H. Cheng, Semi-supervised clustering with partial background information, in: SIAM International Conference on Data Mining, 2006.
- [41] S. Basu, M. Bilenko, R.J. Mooney, A probabilistic framework for semi-supervised clustering, in: International Conference on Knowledge Discovery and Data Mining, 2004, pp. 59–68.
- [42] C.F. Eick, N. Zeidat, Z. Zhao, Supervised clustering – algorithms and benefits, in: International Conference on Tools with Artificial Intelligence, 2004, pp. 774–776.
- [43] W. Pedrycz, Fuzzy clustering with a knowledge-based guidance, *Pattern Recognition Letters* 25 (4) (2004) 469–480.
- [44] Abdelhamid Bouchachia, Witold Pedrycz, Data clustering with partial supervision, *Data Mining and Knowledge Discovery* 12 (1) (2006) 47–78.
- [45] W. Pedrycz, Collaborative and knowledge-based fuzzy clustering, *International Journal of Innovative Computing, Information and Control* 1 (3) (2007) 1–12.
- [46] Benjamin C.M. Fung, Ke Wang, Lingyu Wang, Patrick C.K. Hung, Privacy-preserving data publishing for cluster analysis, *Data & Knowledge Engineering* 68 (6) (2009) 552–575.
- [47] C. Wemmert, P. Gançarski, A multi-view voting method to combine unsupervised classifications, *Artificial Intelligence and Applications* (2002) 362–424.
- [48] J. Cohen, A coefficient of agreement for nominal scales, *Educational and Psychological Measurement* 20 (1) (1960) 37.
- [49] W.M. Rand, Objective criteria for the evaluation of clustering methods, *Journal of the American Statistical Association* 66 (1971) 622–626.
- [50] P. Tan, M. Steinbach, V. Kumar, *Introduction to Data Mining*, first ed., Addison-Wesley, Longman Publishing Co., Inc., Boston, MA, USA, 2005.
- [51] M. Halkidi, M. Vazirgiannis, Y. Batistakis, On clustering validation techniques, *Journal of Intelligent Information Systems* 17 (2–3) (2001) 107–145.
- [52] C. Wemmert, P. Gançarski, J. Korczak, A collaborative approach to combine multiple learning methods, *International Journal on Artificial Intelligence Tools (World Scientific)* 9 (1) (2000) 59–78.
- [53] Ping Luo, Hui Xiong, Guoxing Zhan, Junjie Wu, Zhongzhi Shi, Information-theoretic distance measures for clustering validation: generalization and normalization, *IEEE Transactions on Knowledge and Data Engineering* 21 (9) (2009) 1249–1262.
- [54] S. Salvador, P. Chan, Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms, in: IEEE International Conference on Tools with Artificial Intelligence, 2004, pp. 576–584.
- [55] J.B. Macqueen, Some methods of classification and analysis of multivariate observations, in: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.
- [56] A. Asuncion, D.J. Newman, Uci machine learning repository , 2007.
- [57] J. Quinlan, Improved use of continuous attributes in {C4.5}, *Journal of Artificial Intelligence Research* 4 (1996) 77–90.



Germain Forestier received the M.Sc. degree in Computer Science from the University of Strasbourg, France in 2007. He is currently working towards the Ph.D. degree at the LSIIT, University of Strasbourg, France. His work is focused on data mining, knowledge discovery and collaborative clustering with applications in remote sensing image analysis.



Pierre Gançarski received his Ph.D. degree in Computer Science from the University of Strasbourg. He is currently an Associate Professor at the Department of Computer Science of the Strasbourg University. His current research interests include collaborative multistrategical clustering with applications to complex data mining and remote sensing analysis.



Cédric Wemmert received in 2000 the Ph.D. degree from the University of Strasbourg. He is currently an Assistant Professor in the Department of Computer Science and the LSIIT, University of Strasbourg. His research interests are in ensemble clustering and application in the remote sensing field.



Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation

S. Derivaux, G. Forestier, C. Wemmert *, S. Lefèvre

Image Sciences, Computer Sciences and Remote Sensing Laboratory, LSIIT UMR 7005, CNRS–University of Strasbourg, Pôle API, Blvd Sébastien Brant, P.O. Box 10413, 67412 Illkirch Cedex, France

ARTICLE INFO

Article history:

Received 10 July 2009

Available online 23 July 2010

Communicated by Y.J. Zhang

Keywords:

Supervised image segmentation

Watershed transform

Fuzzy classification

Genetic algorithm

ABSTRACT

Automatic image interpretation is often achieved by first performing a segmentation of the image (i.e., gathering neighbouring pixels into homogeneous regions) and then applying a supervised region-based classification. In such a process, the quality of the segmentation step is of great importance in the final classified result. Nevertheless, whereas the classification step takes advantage from some prior knowledge such as learning sample pixels, the segmentation step rarely does. In this paper, we propose to involve such samples through machine learning procedures to improve the segmentation process. More precisely, we consider the watershed transform segmentation algorithm, and rely on both a fuzzy supervised classification procedure and a genetic algorithm in order to respectively build the elevation map used in the watershed paradigm and tune segmentation parameters. We also propose new criteria for segmentation evaluation based on learning samples. We have evaluated our method on remotely sensed images. The results assert the relevance of machine learning as a way to introduce knowledge within the watershed segmentation process.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

The goal of image understanding is to identify meaningful objects (from a user point of view) within an image. This process usually relies on two distinct steps: segmentation and classification. The segmentation clusters pixels into regions (i.e., it assigns to each pixel a region label) whereas classification clusters regions into classes (i.e., it assigns to each region a class label). A region is a set of connected pixels from which rich features can be extracted (e.g., shape, textural indexes, etc.). These features, which cannot be extracted at pixel level, are expected to improve the classification accuracy. Nowadays, this kind of approach is widely used, in particular in the remote sensing field (Blaschke, 2010).

To build an accurate classification, the segmentation should return a set of regions with a one-to-one mapping to the semantic objects (from a user perspective) present within the image. However, this is hardly possible due to image complexity. Indeed, since a segmentation algorithm is usually designed to cluster connected pixels according to a homogeneity criterion, achieving a good segmentation needs to involve such a relevant homogeneity criterion. Common criteria (e.g., graylevel or spectral homogeneity, but also textural indexes) may not be relevant when processing complex images, such as very high resolution remotely sensed images

where semantic objects have no spectral homogeneity (e.g., a house may be quite heterogeneous, due to the presence of windows on the roof, or a different illumination on each side of the roof). The lack of relevant segmentation criteria leads to two main problems encountered during the segmentation process. On the one hand, undersegmentation may occur when a given region spans over objects of different classes. Whatever the subsequent classifier is, some parts of the region will necessarily be misclassified. Thus, undersegmentation leads to segmentation errors that cannot be recovered in the classification step. On the other hand, oversegmentation may occur when a semantic object is covered by many regions. In this case, extracted attributes, especially shape and topological properties, are far less representative of the object class. The classification, using such noisy attribute values will produce a lower quality result. Designing a segmentation method able to avoid both under and oversegmentation is then very challenging.

To cope with this problem, and to achieve a one-to-one correspondence between the segmented regions and the semantic objects defined by user knowledge, homogeneity criteria involved in the segmentation process need to be related to the user's knowledge. In the context of image understanding, this knowledge is often brought by the user through learning samples given as an input to the (supervised) classification step. It seems very interesting to also exploit these samples in the segmentation step and to elaborate more semantic homogeneity criteria. By analogy with

* Corresponding author. Tel.: +33 (0) 3 90 24 45 81; fax: +33 (0) 3 90 24 44 55.
E-mail address: wemmert@unistra.fr (C. Wemmert).

supervised classification, segmentation methods guided by learning samples are called here *supervised segmentation* algorithms.

In this paper, we propose a new supervised segmentation method relying on learning samples (also called ground truth) in two different ways. Firstly, ground truth information is used to learn how to project the source image in a more relevant data space, where the homogeneity assumption between connected pixels is true and where a well-known segmentation method (i.e., the watershed transform) can be applied. Secondly, ground truth is used to learn an adequate set of segmentation parameters using a genetic algorithm. Genetic algorithms were chosen here to optimize the segmentation parameters, because they are very efficient methods commonly used for objective functions optimization (Goldberg and Holland, 1988). Moreover, they have already been used in the context of segmentation parameters optimization, as mentioned in Section 2.2. Similarly to some recent studies (Lezoray et al., 2008), our contributions show that designing machine learning-based image processing algorithms is a very promising way to rely on user knowledge.

We start by recalling the main principles of watershed segmentation and briefly reviewing how this method has been supervised. We then describe several ways to perform supervised segmentation: space transformation (Section 3), segmentation parameters optimization (Section 4) and finally an hybrid method combining the two approaches (Section 5). In Section 4, we also deal with the problem of segmentation evaluation and introduce several new criteria which will be used as fitness function within the genetic algorithm. Then, we provide both an analytical evaluation of the algorithms and an experimental and quantitative evaluation in remote sensing. Finally, conclusions and some research directions are drawn.

2. Watershed segmentation and its supervision

In this section, we recall the main principles of the watershed transform, a widely used morphological approach for image segmentation. We also present related work, i.e., attempts to introduce user knowledge in the watershed-based image segmentation.

2.1. Watershed segmentation

The watershed transform has been chosen as the base segmentation algorithm in our approach, which may however be applied with any segmentation algorithm (and especially those needing parameter settings, see Section 4). It is a well-known segmentation method which considers the image to be processed as a topographic surface. In the immersion paradigm from Vincent and Soille (1991), this surface is flooded from its minima, thus generating different growing catchment basins. Dams are built to avoid merging water from two different catchment basins. The segmentation result is defined by the locations of the dams (i.e., the watershed lines) when the whole image has been flooded, as illustrated in Fig. 1.

In this approach, the topographic surface is most often built from an image gradient, since object edges (i.e., watershed lines) are most probably located at pixels with high gradient values. Different techniques can be involved to compute the image gradient. Since it does not affect our study, we consider here as an illustrative example, the morphological gradient (Soille, 2003) computed marginally (i.e., independently) for each image band and combined through an Euclidean norm. Vectorial morphological approaches may of course be involved (Aptoula and Lefèvre, 2007).

In its original, marker-free version, the watershed segmentation is proven to easily generate an oversegmentation (i.e., a segmentation where the number of regions created is far larger than the number of actual regions in the image). A smoothing filter is often

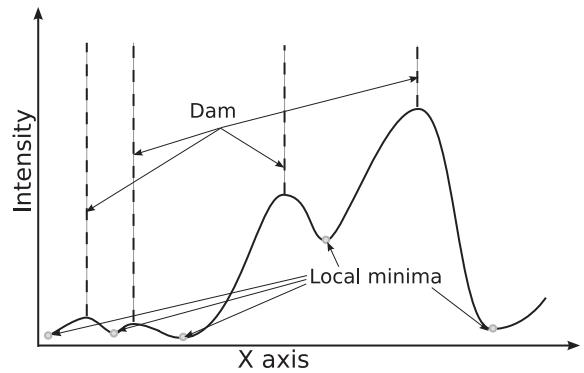


Fig. 1. Illustration of the watershed segmentation principle. For each pixel, the elevation relies here on the intensity within the image.

applied on the input image to overcome this problem. Here we have decided to process marginally all image bands with a median filter (of size 3×3 pixels, which is adequate for our task) in order to preserve image edges.

To further reduce oversegmentation, we may use other, more advanced methods. In this paper we consider three well-established techniques but our proposal is not limited to those approaches.

First, the gradient thresholding method (Haris et al., 1998) is used. On the grayscale gradient image considered as the topographic surface, each pixel with a value below a given threshold (written h_{min}) is set to zero. This step removes small heterogeneity effects. On Fig. 2, this step is represented by the h_{min} line: all values under this line are set to null, and thus, two watersheds are removed.

The concept of dynamics (Najman and Schmitt, 1996) is also involved. Catchment basins with a dynamic (written d) under a given threshold are filled. On Fig. 2 this step is represented by the catchment basin which starts from A. If its dynamic d is below the considered threshold, this catchment basin is filled and the left watershed is removed.

The last method involved here is region merging (Haris et al., 1998). For each region produced by the watershed transform, the average spectral signature is computed from its pixels and considered as a feature vector. If the Euclidean distance between vectors of two neighbouring regions is below a given threshold (written M), these two regions are merged.

2.2. Supervised segmentation

Another way to improve the quality of the segmentation is to leverage the knowledge or examples available on the image. This family of methods is called *supervised segmentation* methods.

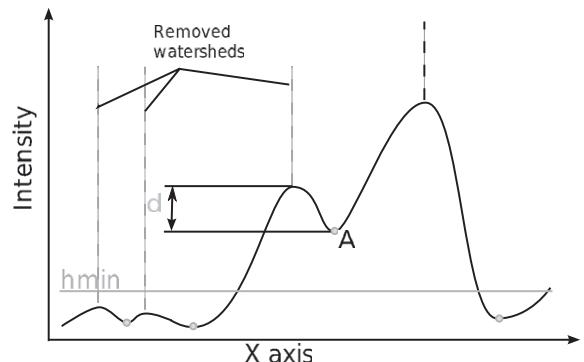


Fig. 2. Illustration of watershed-related oversegmentation reduction methods considered in this paper.

The most frequent use of examples (or ground truth in the field of remote sensing) is to perform an optimization to find the best segmentation parameters (Bhanu et al., 1995; Pignalberi et al., 2003; Song and Ciesielski, 2003; Martin and Maillot, 2006; Feitosa et al., 2006). This kind of methods involves a common segmentation algorithm which can be tuned by a set of parameters. The genetic algorithm finds a set of parameters which optimize a fitness function. Different fitness functions were proposed using different segmentation criteria based on ground truth. We will focus on this strategy in Section 4.

A completely different approach was proposed by Meyer and Beucher (1990), where knowledge is introduced using markers in the watershed algorithm. Many methods have been proposed for the choice of markers using knowledge. In these methods, the user may locate the markers, which are used only as the initial positions of the catchment basins, i.e., the regions to be segmented. Recently, Lefèvre (2007) proposed another marker-based watershed method where the segmentation process also relies on the contents of the markers. Marker pixels are involved in a supervised pixel classification process whose result is merged with the gradient of the input image to build the topographic surface. This approach share some properties with the strategy proposed in Section 3, but requires the user to set relevant markers for all the objects to be segmented (which cannot be achieved in many contexts, e.g., remote sensing).

It is also possible (but less common) to apply the watershed on a modified input image. As our approach could be classified in this category of methods, we review the related major contributions hereafter.

Haker et al. (2000) use manually segmented images to extract, for each object, a priori membership probabilities to belong to the different classes of interest. Then, they are combined using Bayes rules. Other kinds of data knowledge can be included in the process, for example spatial relations between objects of interest. This approach is comparable to a supervised classification, thus it faces the same problem of undersegmentation. Nevertheless, it produces better results if the user can approximately determine the position of the objects in the scene.

In a similar way, Levner and Zhang (2007) propose a method working with probability maps. They use a first classification, based on an eroded ground truth to find some seeds. Another classification is applied using original ground truth and the resulting inverted probability map is used as an elevation. This approach is currently applied only on binary classification. Also, this method assumes the detection of all seeds. If a seed is missed then the underlying object is not segmented.

Another method proposed by Grau et al. (2004) uses a probability map for each class of interest. In this approach, markers are generated using an atlas. Each marker has an associated class. A region growing approach is used to simulate flooding. The elevation between two pixels relies on the original marker class as it uses the probability difference between these pixels in the probability map for the marker class (i.e., it is a markovian process). This approach also needs the knowledge of markers locations.

Other ways to introduce knowledge within the segmentation process have been proposed. Hamarneh and Li (2007) perform a watershed segmentation with the classical oversegmentation problem. They use a modified k -means algorithm in order to cluster segments by intensity and position. Using appearance knowledge, they select the appropriate cluster and iteratively align a shape histogram over the result to remove irrelevant remaining segments. This approach relies heavily on the assumption that objects have homogeneous intensity values, assumption which cannot be made in our context.

Chen et al. (2003) extract a shape and intensity model of the object of interest from a set of reference segmentations. After the learning step, they use an active contour model in order to segment

the objects in respect with the shape and intensity model previously defined. This method works only for single object detection and approximative location needs to be known.

From this brief review of related work, we can notice that involving knowledge into the segmentation process is a relevant idea which leads to several approaches recently proposed. In order to highlight our contribution and the goals of this paper, we point out the main properties which differs our work from other existing approaches:

- ability to deal with many classes;
- knowledge about the position of objects is not needed;
- ability to deal with spectrally inseparable classes i.e., where marker creation using classification is not possible.

3. Supervised segmentation by space transformation

Segmentation algorithms aim to produce an image partition (i.e., a segmentation) which ensures several fundamental properties. Thus, all regions of the segmentation have to fulfil a predefined segmentation criterion. In other words, extracted objects are expected to be homogeneous, i.e., they are built by gathering adjacent pixels with similar values (spectral similarity is most often considered, but other criteria may be used, e.g., texture). However, when dealing with very high resolution remotely sensed images, this assumption does not hold any more. Indeed, too many details appear in such images (e.g., cars are visible on the roads, shadows of the buildings appear, etc.). Thus, we propose here another approach, called probashed, that modifies the data space in which the segmentation is applied.

The main idea is to use the examples given by the user to define a new homogeneity between the pixels. For this, we project the pixels in a new data space in which the sample regions are composed of homogeneous pixels. Then, classical segmentation algorithms can be applied and should give better results (according to the samples given by the user).

To produce the new data space based on the examples, we apply a supervised classification method on the data. Applying a hard classification technique would produce a binary membership map, which is of limited usage when given as an input to a segmentation algorithm. As we are considering to apply a watershed segmentation on the membership map, we rather need a more descriptive data representation. Thus, we perform a fuzzy classification of the data, in order to obtain a grayscale membership map which can then be processed by the watershed transform.

A graphical representation of the supervised segmentation process is presented in Fig. 3(b). The proposed method breaks down into two parts:

- fuzzy classification: based on the samples given by the user;
- watershed segmentation: the segmentation is applied on the membership map given by the fuzzy classification (not on the original image).

Let us describe more precisely the space transformation strategy. We write S_i the input space:

$$S_i : E \rightarrow \mathbb{R}^i \\ x \mapsto S_i(x) \text{ with } S_i(x) \text{ the spectral signature of the pixel } x \quad (1)$$

As we are facing complex images, we cannot assume that a perfect decision function (i.e., a function able to assign the correct class for every pixel from S_i) exists. Since only approximation functions exist, we consider the space of membership values and write it S_m :

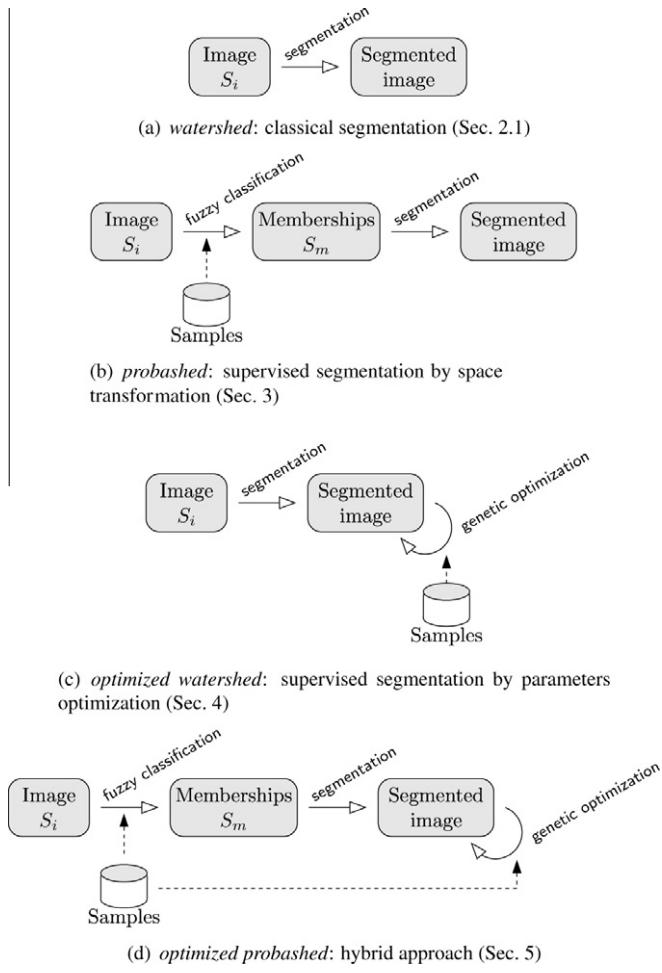


Fig. 3. The different segmentation processes presented in this paper.

$$S_m : E \rightarrow [0; 1]^{\Omega(C)} \\ x \mapsto S_m(x) \text{ with } S_m(x) \text{ the membership vector of the pixel } x \quad (2)$$

with $\Omega(C)$ the number of classes. In this membership space, each class of objects contained in the image and provided by the user is assumed to be a dimension of the space. Thus the value in each dimension denotes the membership of the pixel to the corresponding class of objects.

In order to build the membership space S_m from the input space S_i , we propose to rely on data mining tools and to perform a learning process based on the available ground truth.

As an illustrative example, we use here a N nearest neighbours classifier (Aha et al., 1991) to achieve the fuzzy classification and compute the membership values. For each input pixel p , the N nearest labeled pixels in the S_i space are selected. Each neighbouring pixel p_n will increase the membership degree of the class it has been labeled with, weighted by the inverse of the distance $d(p, p_n)$ in the feature space, with $d : \mathbb{R}^i \times \mathbb{R}^i \rightarrow \mathbb{R}^+$ a given distance measure, e.g., the Euclidean distance. The memberships $m_{p,k}$ are then obtained by:

$$m_{p,k} = \left(\sum_{n=1}^N \sum_{l=1}^K w_{n,l} \right)^{-1} \sum_{n=1}^N w_{n,k} \quad (3)$$

where $w_{n,k} = \begin{cases} d(p, p_n)^{-1} & \text{if } p_n \text{ is labeled with class } k \\ 0 & \text{otherwise.} \end{cases}$

In this section, we have presented the probashed supervised segmentation method which consists in applying a watershed seg-

mentation on a transformed data space. This transformation is computed using a fuzzy classification of the data from which fuzzy probability membership maps are built. Consequently, the watershed is applied on the membership maps instead of the raw data, which allows the method to better grasp the complexity of the image and leverage the available knowledge. An evaluation and an application of this method are given in Section 6.

4. Supervised segmentation by parameters optimization

In the previous section, learning examples provided by the user have been used to compute a new similarity criterion between pixels. The segmentation algorithm is then applied on a modified input image where spectral values have been replaced by class memberships. Another way to improve the segmentation is to rely on the learning samples to automatically find the best parameters required for the algorithm. This can be achieved using an optimization framework, and we propose to use here a genetic algorithm.

A genetic algorithm (GA) is an optimization method (Gersho and Gray, 1992), based on a function to maximize, called the *fitness function*. The definition of this fitness function is a critical point of these methods. Indeed, the fitness has to evaluate the solutions proposed by the GA, in order to drive it to the *best* solutions.

In this section, we first describe the parameters optimization algorithm, and then present and compare different kinds of segmentation evaluation criteria that could be used as fitness functions.

4.1. Parameters optimization algorithm

Let us emphasize that the watershed segmentation method (and its parameters) considered in this paper is just a simple example to illustrate our contribution which consists in a general evolutionary framework for optimizing segmentation parameters. Another segmentation algorithm could have been used instead.

As it has been underlined previously, the base segmentation algorithm (and more precisely the oversegmentation reduction techniques) requires several parameters to be set. We explain here how the genetic algorithm proceeds to tune these parameters.

Given an evaluation function $f(G)$ where G (the genotype in the genetic framework) is taken in a space \mathbb{G} , the GA searches the optimal value of G , i.e., $\arg \max_{G \in \mathbb{G}} f(G)$. GA are known to be effective even if $f(G)$ contains many local minima. This optimization can be considered as a learning process, if and only if it is performed on a learning set but can be generalized to other (unlearned) datasets.

The genotype G is defined as an array containing the parameters that have to be automatically tuned in the watershed segmentation process, i.e., $G = [\omega_1, \dots, \omega_n]$, with all parameters normalized into $[0; 1]$.

A GA requires an initial population defined as a set of genotypes, to perform the evolutionary process. In this process, the population evolves to obtain better and better genotypes, i.e., solutions of the optimization problem under consideration. In order to build the initial population, each genotype is randomly chosen in the space \mathbb{G} .

Once the initial population has been defined, the algorithm relies on the following steps, which represent the transition between two generations:

1. assessment of genotypes in the population: genotypes are sorted by their relevance;
2. selection of genotypes for crossover weighted by their rank;
3. crossover: two genotypes (G_1 and G_2) breed by combining their parameters (or genes in the genetic framework) to give a child E . The resulting child is E with $E[i] = G_{p_i}[i] + \alpha_i \times |G_1[i] - G_2[i]|$

where α_i and p_i are randomly selected in $[-1; 1]$ and $\{1, 2\}$ respectively. We apply an elitist procedure and keep the best solution of the current generation in the next generation;

4. mutation: each parameter may be replaced by a random value with a probability P_m . Thus, we avoid the GA to be trapped in a local minimum. As indicated previously, the best genotype of a generation is kept unchanged.

In our study, we use the following parameters for the GA: a population size of 15 genotypes, a mutation probability P_m of 1%, and an evolution number $N = 30$ generations (experiments shown that no significant improvement is obtained with more generations). The results are presented in Section 6.

Any segmentation evaluation function can be used as fitness function ($f(G)$). Different segmentation evaluation are presented in the following section.

4.2. Segmentation evaluation

In the literature, many criteria for segmentation quality evaluation have been proposed. The reader can refer to Zhang (1996, 2001) for some surveys of this topic. In this paper, we do not consider all existing criteria, but rather focus on criteria based on discrepancy, i.e., comparing a resulting segmentation with some reference regions. This is particularly relevant since we are interested here in evaluation of GA methods in the context of optimal segmentation parameters learning. Criteria which are not based on learning samples are useless when investigating machine learning capabilities of the GA solutions.

Let us define reference samples as a set of connected components $R = \{R_i\}_{i \in [1; \Omega(R)]}$ where each connected component R_i is labeled with a class $C_k = c(R_i)$ from the set $C = \{C_k\}_{k \in [1; \Omega(C)]}$, with Ω the cardinality operator and c the class assignment function. For instance, we could define $C = \{\text{house}, \text{road}, \text{vegetation}\}$ in the remote sensing context. If no class are meaningful, we assign a new class to each reference sample, thus $c(R_i) = C_i$ and $\Omega(R) = \Omega(C)$. We also note R^{C_k} the set of reference samples, sharing the same class label, i.e., $R^{C_k} = \{R_i : c(R_i) = C_k\}$.

We can define three types of discrepancy criteria: classification errors criteria, matching criteria and generalization criteria. In our study, we illustrate these categories by a few representative criteria which will now be described.

4.2.1. Classification errors criteria

These criteria are based on the classification error principle. An image segmentation can be seen as an image classification process, and then, the percentage of misclassified pixels can be used. Since labels are assigned to both produced and reference regions, the number of pixels with different labels between the segmentation and the reference image can be computed.

The criterion used here is derived from the E criterion from Carleer et al. (2005). In the original paper, each reference region has a unique label. In our case, we assign to each reference region a class label. This way, reference regions sharing the same semantic, have the same label. To each segmented region is then assigned the label of the most overlapping reference region (i.e., the region sharing the greatest number of pixels). We define here the TMA criterion (Theoretical Maximum Accuracy), which uses class labels instead of a label for each region. If a segmented region spans over two reference regions of the same class, the TMA criterion does not track an error, whereas the E criterion does, as each reference region has a different label. For each class, error is measured and weighted by the inverse number of reference pixels in order to give the same importance to each class. Then, a per-pixel confusion matrix K is computed. For each evaluation pixel of a class C_i , assigned to a label C_j by the matching, the value of the cell K_{ij} is incremented by

$(\Omega(C_i))^{-1}$ where $\Omega(C_i)$ is the number of reference pixels for class C_i . Thus, the evaluation function TMA is the classifier precision (the overall accuracy):

$$TMA = \frac{1}{\Omega(C)} \sum_{i=1}^{\Omega(C)} K_{ii} \quad (4)$$

The TMA criterion gives the best available accuracy of a subsequent classification step of the resulting segments.

4.2.2. Matching criteria

Matching criteria measure spatial differences between segmented and reference regions. They rely on a matching function $m(R_i, S_j)$ which computes a matching score between a reference region R_i and a segmented region S_j , where $S = \{S_j\}_{j \in [1; \Omega(S)]}$ is the set of segmented regions. Let us additionally define R_{S_j} the set of reference regions overlapping S_j , and inversely S_{R_i} the set of segmented regions overlapping R_i . To apply these criteria on a complete segmentation, the average matching value μ_m of the best matching score for each reference region is computed:

$$\mu_m = \frac{1}{\Omega(R)} \sum_{i=1}^{\Omega(R)} \text{best}_{1 \leq j \leq \Omega(S)}(m(R_i, S_j)) \quad (5)$$

where the best function is the optimum function, i.e., minimum or maximum function depending on the matching criterion.

The first criterion used here is taken from Feitosa et al. (2006) and defined by:

$$F(R_i, S_j) = \frac{\Omega(R_i \setminus (R_i \cap S_j)) + \Omega(S_j \setminus (R_i \cap S_j))}{\Omega(R_i)} \quad (6)$$

where \setminus represents the set difference operator, i.e., $A \setminus B = \{x : x \in A, x \notin B\}$.

We observe that the F criterion favours oversegmentation over undersegmentation and should be minimized to obtain the best segmentation.

The second criterion is taken from Janssen and Molenaar (1995). It is quite similar to F but does not have the bias to avoid oversegmentation. It considers reference and segmented regions in the same way and should be maximized

$$J(R_i, S_j) = \sqrt{\frac{\Omega(R_i \cap S_j)^2}{\Omega(R_i) \times \Omega(S_j)}} \quad (7)$$

In this formulation, if a segmented region S_j spans over two reference regions R_i and $R_{i'}$ of the same class C_k , both matching scores $J(R_i, S_j)$ and $J(R_{i'}, S_j)$ will be low. Nevertheless, as R_i and $R_{i'}$ belongs to R^{C_k} , they could be merged, thus resulting in a high matching score $J(R_i \cup R_{i'}, S_j)$.

This principle leads to a new criterion J_C which relies on class labels. For a given couple (R_i, S_j) , we consider the subset of $R^{c(R_i)} = \{R_{i'} : c(R_{i'}) = c(R_i)\}$ (i.e., the union of all reference regions $R_{i'}$ sharing the label assigned to R_i) overlapping S_j , or $R_{S_j}^{c(R_i)} = R^{c(R_i)} \cap S_j$. The modified criterion is then:

$$J_C(R_i, S_j) = \sqrt{\frac{\Omega(R_{S_j}^{c(R_i)})^2}{\Omega(R_i) \times \Omega(S_j)}} \quad (8)$$

A similar evaluation criterion is the Jaccard index (Jaccard, 1912) which should also be maximized. It is defined as the ratio between the cardinalities of the intersection and the union of the two sets:

$$J'(R_i, S_j) = \frac{\Omega(R_i \cap S_j)}{\Omega(R_i \cup S_j)} \quad (9)$$

Here, we also extend this criterion to handle class labels:

$$JC'(R_i, S_j) = \frac{\Omega(R^{C(R_i)} \cap S_j)}{\Omega(R_i \cup S_j)} \quad (10)$$

We can also mention the ultimate measurement accuracy criterion (Zhang and Gerbrands, 1992), which measures the difference between features extracted from R_i and S_j . Since it strongly depends on the regional features extracted, and thus, is hardly compatible with a generic solution for parameter tuning, we do not consider this criterion in our study.

4.2.3. Generalization criteria

Generalization criteria measure the coarseness of the segmentation.

The *Gen* criterion (Carleer et al., 2005) measures oversegmentation through a simple ratio between the number of segmented and reference regions, i.e., $Gen = \Omega(S)/\Omega(R)$.

Here we consider only segmented regions spanning over a reference one, in order to deal with an incomplete reference segmentation. Moreover, we take into account class information and compute the average oversegmentation for all classes. Thus the proposed criterion *OV* is defined as:

$$OV = \frac{1}{\Omega(C)} \sum_{k=1}^{\Omega(C)} \frac{\Omega(S_{R^{C_k}})}{\Omega(R^{C_k})} \quad (11)$$

where $S_{R^{C_k}}$ denotes the set of segmented regions overlapping at least one of the reference region assigned to the class C_k while R^{C_k} is the set of reference regions assigned to the class C_k .

Another criterion belonging to this category is the average region size (noted *p/r*), i.e., $\Omega(I)/\Omega(S)$ where $\Omega(I)$ and $\Omega(S)$ represent respectively the number of pixels in the image and the number of regions produced by the segmentation. It is rather simplistic and does not involve any sample. Nevertheless, it allows to compare two segmentations to determine the coarsest one.

4.2.4. Hybrid criteria

Among the previous criteria, some criteria measure mainly oversegmentation (e.g., *OV* and *p/r*) while others measure mainly undersegmentation (e.g., *TMA*). So it is relevant to combine these criteria to build some aggregated criteria. Combination is one solution for resolving multi-objective optimization. Another solution is to use the Pareto front (Fonseca and Fleming, 1996). The Pareto front returns a set of results representing different trade-offs between all the considered criteria. Thus, handling a set of results needs more user interaction, which is out of the scope of this paper.

We propose here two multi-objective criteria, combining *TMA* and *OV*.

The first one *TMA/OV*, avoids mainly undersegmentation (using *TMA*) and secondarily oversegmentation (using *OV*). It is simply defined by weighting *OV* with a small coefficient (ε):

$$TMA/OV = TMA + \varepsilon \frac{1}{OV} \quad (12)$$

The second criterion is *TMA* \oplus *OV*(α). It also primarily relies on undersegmentation (using *TMA*), but limits its effect with the α parameter:

$$TMA \oplus OV(\alpha) = \min(TMA, \alpha) + \varepsilon \frac{1}{OV} \quad (13)$$

Of course the α parameter is dependent of the application. It represents the amount of errors (measured by the *TMA* criterion) tolerated by the user or system. For instance, if the *TMA* quality should be at least 95%, the user sets $\alpha = 0.95$.

5. Hybrid approach

In this section, we describe a hybrid method, integrating the two previous ideas presented in Sections 3 and 4. In an offline phase, the method learns how to segment an image using a learning set (composed of images and masks corresponding to objects of interest). The learning process occurs in two steps: a space transformation step and a core segmentation step. Once the learning is finished, a segmentation algorithm (i.e., the space transformation step and the core segmentation step) is produced and can be used to segment images. No learning set is needed in this application phase. The proposed method does not need input parameters in neither phases. A flow chart is shown on Fig. 3(d).

The learning set is composed of learning images and corresponding learning masks. A learning mask is a semantic interpretation of a learning image made by a human expert. For each object, the corresponding pixels in the image are labeled with a class C_k where $k \in [1 \dots K]$ and K is the number of classes. Some pixels could be left unlabeled, denoting the inability to label them.

5.1. Segmentation supervision by genetic algorithm

Here we propose a genetic algorithm in order to handle the parameters from the segmentation step. As already stated in Section 4, the watershed algorithm needs three parameters to be set: *hmin* to ignore low gradient values, *d* for the basin dynamics and *M* as the threshold for the region merging step. In the space transform segmentation algorithm, another parameter is added, which is the same as the *M* threshold, but applied with the mean of membership maps: this new threshold is written *M_m*. Thus, we have four parameters to optimize.

5.2. Evaluation function

As already discussed in Section 4.2, a critical point of the genetic algorithm optimization method is the way the quality of the potential solutions (i.e., genotypes) is estimated. Here, as we are interested in evaluation of segmentation results, we focus on empirical discrepancy evaluation methods following the work from Carleer et al. (2005). Nevertheless, our criteria are adapted to both mixed and user-meaningless pixels which do not appear in such a manual reference segmentation. They are compatible with partially segmented images defined as (incomplete) sets of labeled pixels. We use the term region for a labeled reference region given by the user and the term segment for a region produced by a segmentation.

From the evaluation criteria introduced in Section 4.2, we can define the evaluation function. We can choose to optimize one of the two criteria or a combination of them. Here, we chose to optimize a criterion which represents oversegmentation and undersegmentation using:

$$\mathbb{F}(g) = \frac{1}{OV(g)} \times \max(0, TMA(g) - 0.98) \quad (14)$$

In the proposed function, $\mathbb{F}(g)$ increases as $OV(g)$ is reaching 1 (no oversegmentation) and decreases when $TMA(g)$ decreases. The function is null if $TMA(g)$ is under 98%, i.e., the maximum accuracy is 98% well classified pixels. This threshold was set to give more importance to avoid undersegmentation. It could be modified by the user depending on the image noise and complexity. Ninety-eight percentage seems a good compromise in our experiments. If $TMA(g)$ falls below this threshold the resulting segmentation will be useless.

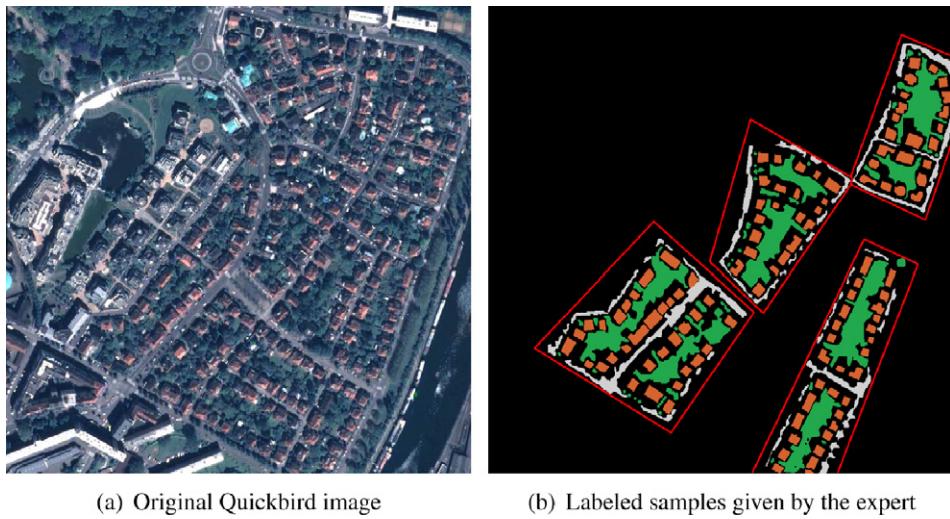


Fig. 4. Remotely sensed image of a part of Strasbourg (France).

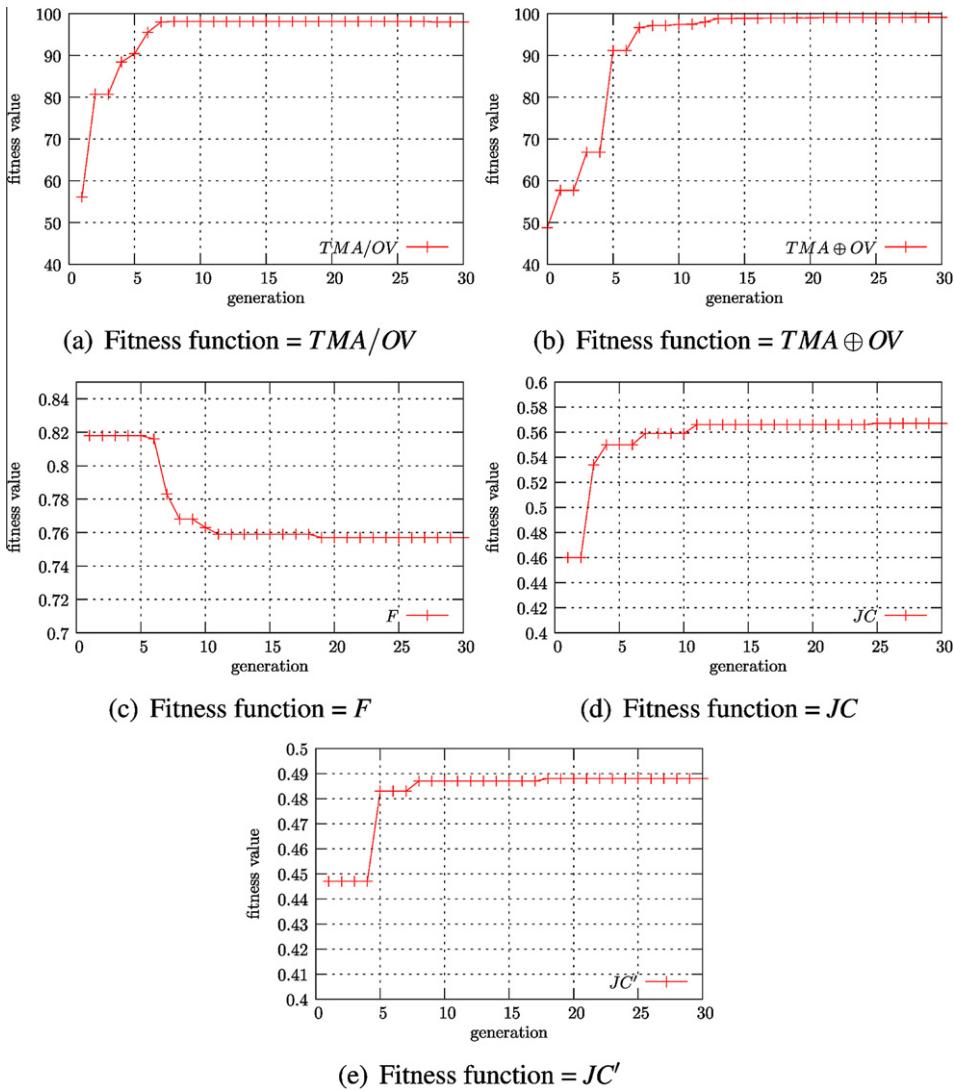


Fig. 5. Evolution of the fitness functions according to the number of generations.

6. Evaluation

The evaluation of the proposed algorithm follows the evaluation scheme proposed by Zhang (1996), using both an analytical evaluation and an empirical discrepancy evaluation. Let us observe that the empirical goodness evaluation is not performed, since it is not relevant here: indeed it usually assumes that segments are spectrally homogeneous.

6.1. Analytical evaluation

The first part of the evaluation is an analytical review of the proposed algorithm. Such a review is helpful to know if the algorithm is suitable to an image or not. The proposed algorithm requires some knowledge from the user to be able to segment an image:

- Class knowledge: the user needs to know the classes of objects which are sought in the image.
- Samples for each class: some samples of each class are needed for the learning step. The *fuzzy classification* step can work with isolated samples, but the *genetic optimization* step requires labeling of image parts.

There are also some limits which should be noted in the proposed algorithm:

- Connected objects of the same class: if two objects of the same class are spatially connected and have similar memberships to classes, they will be merged together (i.e., undersegmentation). The same problem arises in usual segmentation methods when two objects have similar spectral values.
- Objects having heterogeneous spectral values and membership values: in such a case, the algorithm produces an oversegmentation.

Nevertheless, these limits are weaker than those of classical segmentation algorithms. If an object has heterogeneous spectral and membership values, it will be oversegmented by classical segmentation methods. The case where two spatially connected objects have similar membership values and dissimilar spectral values and each object has homogeneous spectral values seems less frequent than objects with heterogeneous spectral values. It is a tradeoff that should be analyzed depending on the application.

6.1.1. Computational complexity

The computational complexity of this algorithm depends on four parameters: n the number of pixels in the image, $\Omega(C)$ the number of labeled examples, p the population size and N the number of generations of the genetic algorithm. At each step of the GA, the costly part of the algorithm is the evaluation of the genotypes (i.e., the computation of the fuzzy classification followed by the watershed algorithm and the calculation of the evaluation criteria). The fuzzy classification algorithm has a $\mathcal{O}(n\Omega(C))$ complexity. But, as it is only executed once at the beginning of the algorithm, we decided to ignore it in the following. The watershed segmentation algorithm is linear according to n . The evaluation of the fitness function depends on the chosen criterion. In the case of TMA, it is linear according to $\Omega(C)$. Thus, the complexity of the evaluation of one genotype is in $\mathcal{O}(n + \Omega(C))$ which can be approximated by $\mathcal{O}(n)$ if we consider that the segmentation is totally recomputed at each evaluation (worst case) and that $\Omega(C) \ll n$ (which seems realistic in most of the cases). Finally, the complexity of the method is in $\mathcal{O}(N \times p \times n)$.

Table 1

Watershed parameters optimization (for readability reasons, F , JC and JC' indexes were multiplied by 100).

Fitness functions	Evaluation criteria				
	TMA	OV	100 × F	100 × JC	100 × JC'
TMA/OV	98.03	48.01	77.4	52.9	44.2
TMA ⊕ OV	99.12	95.10	81.5	43.4	36.4
F	98.56	61.28	75.7	53.4	44.8
JC	96.74	34.83	78.6	56.7	48.4
JC'	96.91	41.12	78.8	55.5	48.8

Bold indicates the best value of each evaluation criterion.

Table 2

Probashed parameters optimization (for readability reasons, F , JC and JC' indexes were multiplied by 100).

Fitness functions	Evaluation criteria				
	TMA	OV	100 × F	100 × JC	100 × JC'
TMA/OV	98.05	4.51	88.2	66.1	52.7
TMA ⊕ OV	99.50	28.79	68.8	64.6	57.5
F	99.40	23.59	68.4	65.3	57.6
JC	98.27	7.58	81.2	68.9	57.2
JC'	99.17	12.88	72.8	67.6	59.2

Bold indicates the best value of each evaluation criterion.

Table 3

Comparison of the different approaches proposed with two commercial segmentation softwares and a supervised per-pixel classification (for readability reasons, F , JC and JC' indexes were multiplied by 100).

Segmentation methods	Evaluation criteria				
	TMA	OV	100 × F	100 × JC	100 × JC'
Watershed	99.18	99.04	17.1	41.5	30.0
Optimized watershed	98.57	61.29	24.3	53.4	44.8
Probashed	99.52	24.33	31.7	65.5	48.3
Optimized probashed	99.41	23.59	31.7	65.3	57.6
eCognition	91.42	35.26	12.9	48.3	51.2
ENVI FX	84.95	2.75	1.3	47.3	59.8
Pixel + median	97.41	2.77	5.7	5.82	56.4
Pixel	97.48	6.69	5.3	5.85	55.5

Bold indicates the best value of each evaluation criterion.

6.2. Application to a real urban image

In the last decade automatic interpretation of remotely sensed images became an increasingly active domain since sensors are now able to produce images with a *very high spatial resolution* (VHSR) (i.e., 1 m resolution). This increasing precision disturbs the classical per-pixel classification procedures and knowledge based systems have been more attentively investigated during the last few years, to improve VHSR image interpretation. Indeed, the so called *object-oriented* (Blaschke et al., 2000; Blaschke, 2010) approach provides a new paradigm of reasoning by focusing on the objects present within an image, and not only on the pixels. The images are segmented and the segments are classified using spectral and spatial attributes (e.g. shape index, texture, etc.).

This case study is a typical example of VHSR image interpretation in remote sensing, where a segmentation is first performed before applying a supervised region-based classification.

The input data is a pan-sharpened Quickbird¹ image of the city of Strasbourg (France) with four spectral bands representing a zone of 15.4 km × 13.3 km, with a spatial resolution of 0.7 m per pixel.

The experiment was performed on the whole zone Derivaux (2009), but we only present here the results on an 900 × 900 pixels extract of the image (Fig. 4(a)). In four areas of the studied zone,

¹ Image provided by the LIVE laboratory from University of Strasbourg.

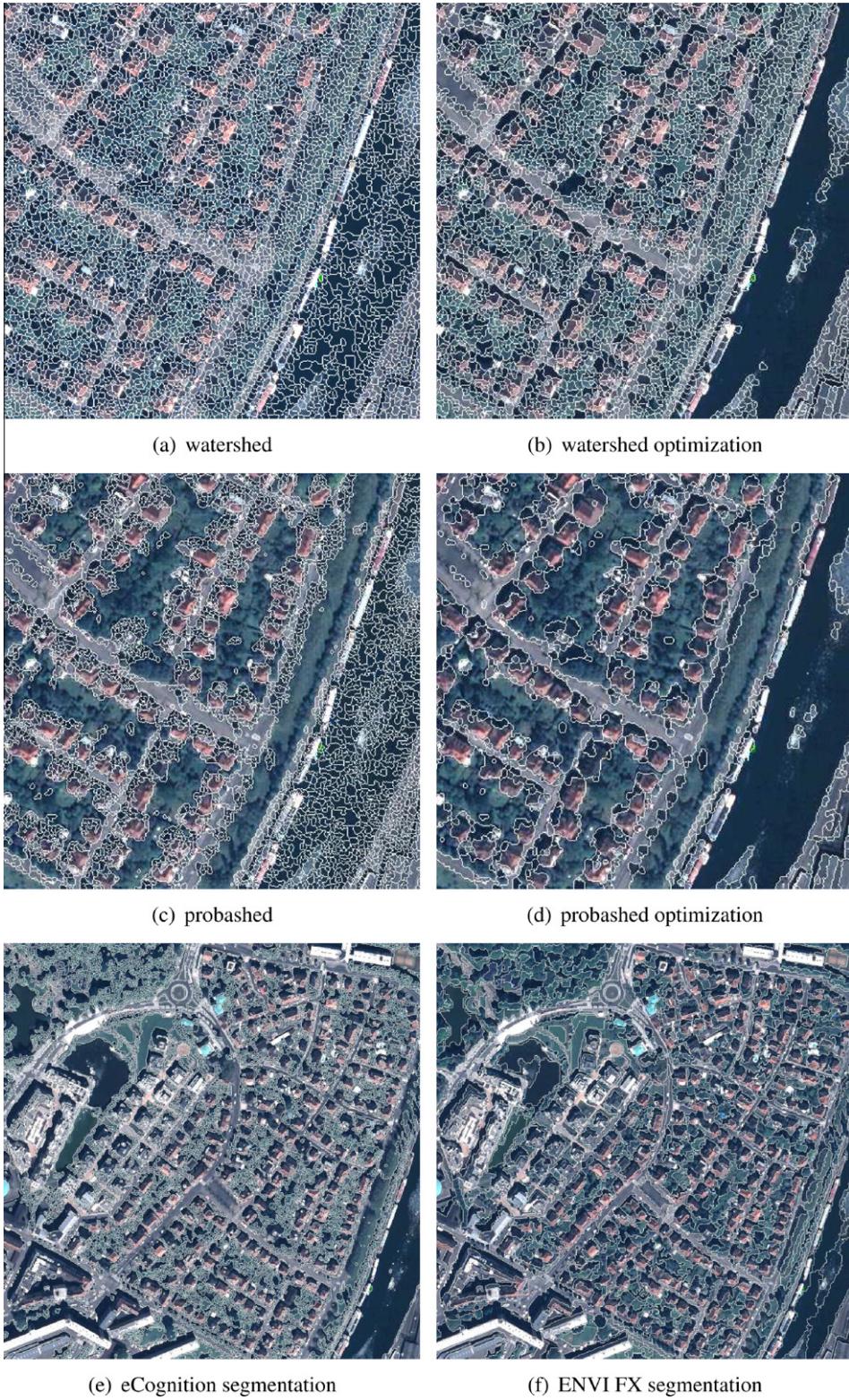


Fig. 6. Segmentation results obtained by the different approaches proposed (extract from the studied image).

some regions (representing 13% of the extract) have been labeled by the expert in three classes: road, vegetation and house (Fig. 4(b)).

6.2.1. Choice of the fitness function

The aim of the first set of experiments carried out on this data was to evaluate the influence of the choice of the fitness function.

Indeed, we presented in Section 4.2 many criteria that could be used as fitness function to optimize the parameters of the segmentation methods. The question is *which criteria shall we optimize to obtain the best result?* We performed a genetic optimization on two segmentation algorithms proposed before: classical watershed and *probashed* (which corresponds to the space transformation

method given in Section 3). For the watershed algorithm, three parameters have to be tuned as stated in Section 4: $hmin$, d , and M . For the probashed algorithm, four parameters are used (Section 5): $hmin$, d , M and M_m .

In our experiments, we consider the following parameters for the genetic algorithm: a population size of 15 genotypes, a mutation probability P_m of 1% and an evolution number equals to 30 generations. Experiments show that stability and convergence is achieved at this step. Fig. 5 shows the trend of the fitness functions with respect to the number of generations. It shows that the convergence is relatively fast and that 30 generations are enough as no significant improvement arises after 20 generations.

We present in Tables 1 and 2 the results obtained by optimizing the parameters of the segmentation method. The first column shows the criterion that has been used as fitness function. Then, each column corresponds to the value obtained by the final result for each evaluation criterion.

It is important to notice that three criteria have to be maximized ($0 < TMA < 100$, $0 < JC < 1$, $0 < JC' < 1$), while two have to be minimized ($0 < F < 1$ and $0 < OV$).

The first remark concerns the three last lines of the two tables. It is obvious that optimizing one criterion will produce the best result for this criterion. This is verified on these results for the three criteria F_{JC} and JC' .

Concerning the hybrid criteria, $TMA \oplus OV$ seems to be a better compromise as TMA/OV because it optimizes well the TMA criterion, without having bad results with the other ones.

6.2.2. Comparison of the different approaches proposed

The second experiment tries to compare the different approaches proposed in this paper. To have a more thorough study, we also included two results given by two commercial remote sensing segmentation software: eCognition™ from Definiens² and ENVI FX from ITT Visual Information Solutions.³ These results were manually computed by a geographer expert. We also computed a supervised per-pixel classification using a 5 nearest neighbours classifier for comparison purpose. The results are presented for a raw per-pixel classification and a per-pixel classification after the application of a median filter (with a window of 3×3 pixels).

Again, we present in Table 3 the evaluations calculated from the different criteria on the results given by the different proposed methods. For the optimized methods, we only give the result with F as fitness function for a better readability. We choose F because it has good results with quite all the evaluation criteria.

Concerning the TMA criterion, no significant improvement is shown compared to the classical or optimized version of the watershed. But compared to the two commercial softwares, the probashed algorithm gives better results. For OV , F and JC , the two probashed algorithms present better results as the other methods. The space transformation brings a significant contribution to the quality of the solution. Finally, results for the JC' criterion are comparable with those given by the commercial softwares and better than those given by the watershed. In conclusion, the probashed algorithms seem to perform better results according to the different quality criteria proposed here.

As it is difficult to grasp the influence of a small change on a criterion, we show in Fig. 6 the segmentations produced by the different methods. Thus, it is possible to have a visual appreciation of the quality of the results. It is clear that the watershed, even in its optimized version, produces results that could not be used directly in the classification step. For example, the vegetation zones in the blocks are really oversegmented as well as the houses. It is then

very difficult to use geometrical attributes in the classification, as the shape of the regions does not necessarily correspond to the expected one.

When comparing the probashed method and its optimized version, the values for the evaluation criteria are comparable or better for the optimized version. But the main differences are visible on the segmentation results (Fig. 6). It is obvious that the river (East of the image) is better delimited as the houses in the blocks.

7. Conclusion

In this article, we presented and compared different criteria to optimize segmentation parameters, when examples are available. We also exposed another way to take advantage of ground truth, in changing the data space before applying the segmentation algorithm. The space transformation is performed by a fuzzy classification based on the examples given by the expert. It has been shown that using this knowledge to guide the segmentation enables to produce better results, even better than manually produced segmentations by an expert.

In future work, we would like to focus on the study of the integration of other kinds of knowledge (not only examples) in the segmentation process. For example, a hierarchy of concepts describing the objects of interest could help to better identify which regions are well segmented. We also plan to use several segmentation algorithms and make them collaborate to find a better segmentation.

References

- Aha, D.W., Kibler, D.F., Albert, M.K., 1991. Instance-based learning algorithms. *Machine Learn.* 6, 37–66.
- Aptoula, E., Lefèvre, S., 2007. A comparative study on multivariate mathematical morphology. *Pattern Recognition* 40 (11), 2914–2929.
- Bhanu, B., Lee, S., Das, S., 1995. Adaptive image segmentation using genetic and hybrid search methods. *IEEE Trans. Aerosp. Electron. Systems* 31 (4), 1268–1291.
- Blaschke, T., 2010. Object based image analysis for remote sensing. *ISPRS J. Photogramm. Remote Sens.* 65 (1), 2–10.
- Blaschke, T., Lang, S., Lorup, E., Strobl, J., Zeil, P., 2000. Object-oriented image processing in an integrated GIS/remote sensing environment and perspectives for environmental applications. *Environ. Inform. Plan. Polit. Public* 2, 555–570.
- Carleer, A.P., Debeir, O., Wolff, E., 2005. Assessment of very high spatial resolution satellite image segmentations. *Photogramm. Eng. Remote Sens.* 71 (11), 1285–1294.
- Chen, Y., Huang, F., Tagare, H., Rao, M., Wilson, D., Geiser, E., 2003. Using prior shape and intensity profile in medical image segmentation. In: Proc. 9th IEEE Internat. Conf. on Computer Vision, pp. 1117–1124.
- Derivaux, S., 2009. Construction et classification d'objets à partir d'images de télédétection par une approche itérative guidée par des connaissances du domaine. Ph.D. Thesis, University of Strasbourg.
- Feitosa, R.Q., Costa, G.A., Cazes, T.B., Feijó, B., 2006. A genetic approach for the automatic adaptation of segmentation parameters. In: Internat. Conf. on Object-based Image Analysis.
- Fonseca, C.M., Fleming, P.J., 1996. An overview of evolutionary algorithms in multiobjective optimization. *Evolut. Comput.* 1 (3), 1–16.
- Gersho, A., Gray, R.M., 1992. Vector Quantization and Signal Compression. Kluwer Academic Publishers..
- Goldberg, D., Holland, J., 1988. Genetic algorithms and machine learning. *Machine Learn.* 3 (2), 95–99.
- Grau, V., Mewes, A., Alcaniz, M., Kikinis, R., Warfield, S., 2004. Improved watershed transform for medical image segmentation using prior information. *IEEE Trans. Med. Imag.* 23 (4), 447–458.
- Haker, S., Sapiro, G., Tannenbaum, A., 2000. Knowledge-based segmentation of SAR data with learned priors. *IEEE Trans. Image Process.* 9 (2), 299–301.
- Hamarneh, G., Li, X., 2007. Watershed segmentation using prior shape and appearance knowledge. *Image Vision Comput.* doi:10.1016/j.imavis.2006.10.009.
- Haris, K., Efstratiadis, S.N., Maglaveras, N., Katsaggelos, A.K., 1998. Hybrid image segmentation using watersheds and fast region merging. *IEEE Trans. Image Process.* 7 (12), 1684–1699.
- Jaccard, P., 1912. The distribution of flora in the alpine zone. *New Phytol.* 11 (2), 37–50.
- Janssen, L., Molenaar, M., 1995. Terrain objects, their dynamics and their monitoring by the integration of GIS and remote sensing. *IEEE Trans. Geosci. Remote Sens.* 33, 749–758.
- Lefèvre, S., 2007. Knowledge from markers in watershed segmentation. In: IAPR Internat. Conf. on Computer Analysis of Image and Patterns (CAIP). Lecture

² <http://earth.definiens.com/>.

³ <http://www.itvis.com/>.

- Notes in Computer Sciences, vol. 4673. Springer, Vienna, pp. 579–586 <<http://lsiiit.cnrs.unistra.fr/Publications/2007/Lef07>>.
- Levner, I., Zhang, H., 2007. Classification-driven watershed segmentation. *IEEE Trans. Image Process.* 16 (5), 1437–1445.
- Lezoray, O., Charrier, C., Cardot, H., Lefèvre, S. (Eds.), 2008. Machine Learning in Image Processing. EURASIP J. Adv. Signal Process.
- Martin, V., Maillot, N.M.T., 2006. A learning approach for adaptive image segmentation. In: IEEE Internat. Conf. on Computer Vision Systems, pp. 40–48.
- Meyer, F., Beucher, S., 1990. Morphological segmentation. *J. Visual Commun. Image Represent.* 1 (1), 21–46.
- Najman, L., Schmitt, M., 1996. Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Trans. Pattern Anal. Machine Intell.* 18 (12), 1163–1173.
- Pignalberi, G., Cucchiara, R., Cinque, L., Levialdi, S., 2003. Tuning range image segmentation by genetic algorithm. *EURASIP J. Appl. Signal Process.* 2003 (8), 780–790.
- Soille, P., 2003. Morphological Image Analysis, second ed. Springer-Verlag.
- Song, A., Ciesielski, V., 2003. Fast texture segmentation using genetic programming. *IEEE Congress Evolut. Comput.* 3, 2126–2133.
- Vincent, L., Soille, P., 1991. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Pattern Anal. Machine Intell.* 13 (6), 583–598.
- Zhang, Y.J., 1996. A survey on evaluation methods for image segmentation. *Pattern Recognition* 29 (8), 1335–1346.
- Zhang, Y.J., 2001. A review of recent evaluation methods for image segmentation. In: Internat. Symposium on Signal Processing and Its Applications, vol. 1, pp. 148–151.
- Zhang, Y.J., Gerbrands, J.J., 1992. Segmentation evaluation using ultimate measurement accuracy. In: Proc. SPIE, Image Processing Algorithms and Techniques III, vol. 1657, pp. 449–460.