



HAL
open science

Détection et classification de changements sur des scènes urbaines en télédétection

Alexandre Fournier

► **To cite this version:**

Alexandre Fournier. Détection et classification de changements sur des scènes urbaines en télédétection. Interface homme-machine [cs.HC]. Institut Supérieur de l'Aéronautique et de l'Espace, 2008. Français. NNT: . tel-00463593

HAL Id: tel-00463593

<https://theses.hal.science/tel-00463593v1>

Submitted on 12 Mar 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'**Institut Supérieur de l'Aéronautique et de l'Espace**
Spécialité : Signal, image et optimisation

Présentée et soutenue par **Alexandre FOURNIER**
le 31 octobre 2008

Détection et classification de changements
sur des scènes urbaines en télédétection

JURY

M. Jean-Yves Tourneret, président
M. Jacques Blanc-Talon
M. Mihai Datcu, rapporteur
M. Xavier Descombes
Mme Christine Graffigne, rapporteur
M. Jordi Inglada
M. Emmanuel Zenou
Mme Josiane Zerubia, directrice de thèse

École doctorale : **Aéronautique-Astronautique**

Unité de recherche : **INRIA Sophia Antipolis**

Directrice de thèse : **Mme Josiane Zerubia**

Table des matières

Table des matières	3
Introduction	9
1 État de l'art	11
1.1 Différents types de problématiques	11
1.2 Prétraitements	12
1.2.1 Recalage	12
1.2.2 Mise en correspondance radiométrique	14
1.3 Méthodes pixelliques	14
1.3.1 Opérateurs naïfs	14
1.3.2 Cas multispectral	15
1.4 changement de contraste globaux	16
1.4.1 Opérateurs statistiques	16
1.4.2 Réseaux de neurones	16
1.4.3 Prise de décision	17
1.5 Approches « contextuelles »	18
1.5.1 Subdivision de l'image	18
1.5.2 Voisinages	18
1.5.3 Approche markovienne, régularisation	18
1.5.4 Texture	19
1.5.5 Ensembles de niveau	19
1.5.6 δ -classification	19
1.6 Systèmes Experts	20
1.7 Conclusion	20
2 Obtention d'un masque de changements	21
2.1 Un algorithme de normalisation de contraste	22
2.1.1 Description de l'algorithme	22
2.1.2 Régularisation par champs de Markov	24
2.1.3 Résultats	25

2.1.4	Conclusion	26
2.2	Une approche fondée sur les lignes de niveau	29
2.2.1	introduction	29
2.2.2	Notations, définitions et modélisation	32
2.2.3	Invariance des lignes de niveau	36
2.2.4	Algorithme invariant aux changements d'illumination pour la détection de changements	47
2.2.5	Un algorithme d'égalisation de contraste	50
2.2.6	Résultats	51
2.3	Conclusion	62
3	Classification fondée sur la radiométrie	63
3.1	Algorithme de <i>k-moyennes</i> et dérivés	64
3.1.1	Problème des <i>k-moyennes</i>	64
3.1.2	Description de l'algorithme des <i>k-moyennes</i>	65
3.1.3	Propriétés de Convergence	67
3.2	Définition de l'espace d'état	67
3.2.1	Espace d'état	67
3.2.2	Distance	67
3.3	Évaluation du nombre de classes	68
3.3.1	<i>k-moyennes</i> entropiques	70
3.3.2	Algorithme des <i>k-moyennes</i> entropiques	72
3.4	Régularisation par champs de Markov	74
3.4.1	Notations	74
3.4.2	Modèle <i>a priori</i> (modèle de Potts)	74
3.4.3	Attache aux données (vraisemblance)	75
3.4.4	Probabilité <i>a posteriori</i>	75
3.4.5	Recuit Simulé	76
3.5	Résultats	76
3.5.1	Zone commerciale d'Abidjan	76
3.5.2	Aéroport d'Abidjan	78
3.6	Conclusion	85
4	Approche objet	87
4.1	Approximation polygonale	87
4.1.1	Principes mathématiques	88
4.1.2	Énoncés des problèmes d'approximation polygonale	89
4.1.3	Choix de l'approximation polygonale	89
4.1.4	Formulation Mathématique	89
4.1.5	Algorithme d'optimisation locale	90
4.1.6	Initialisation	94

4.1.7	Résultats expérimentaux	96
4.2	Classification Polygonale	98
4.2.1	Métrie de l'espace des orientations	99
4.2.2	Combinaison de la radiométrie et de la géométrie	100
4.2.3	Résultats	100
4.3	Conclusion	103
Conclusion		107
A Rappels sur les champs de Markov		111
A.1	Modélisation	111
A.2	Optimisation	113
Table des figures		117
Bibliographie		121
Publications		129

Remerciements

Écrire des remerciements est toujours un exercice difficile. N'oublier personne sans tomber dans le catalogue. Éviter les platitudes sans tomber dans la grivoiserie. Je crois que la première personne que je souhaiterais remercier est Beth Gibbons, dont la voix chaude et irréaliste a su m'accompagner et m'isoler du monde pendant les longues périodes de rédaction de ce manuscrit.

Le chemin de la thèse, parfois laborieux et difficile est aussi l'occasion de rencontres. J'étais loin d'imaginer quand j'ai fait la connaissance de Pierre Weiss, il y a maintenant plus de trois ans, qu'il aurait une aussi grande influence sur ma méthode de travail, ma rigueur et mon goût pour la recherche. Je remercie Pierre du fond du cœur pour la collaboration intense que nous avons eu pendant cette dernière année et j'espère le retrouver inchangé à son retour d'Asie.

Cette thèse n'aurait jamais vu le jour sans la présence des mes deux encadrants : Josiane Zerubia et Xavier Descombes qui m'ont accueilli au sein du projet Ariana et guidé pendant ces années. Leurs caractères radicalement opposés ont été le Yin et le Yang qui équilibraient mon cheminement. En particulier, je souhaiterais remercier Josiane pour avoir su m'encourager dans les moments de désillusion et Xavier, dont les conseils avisés m'ont guidé tout au long de ce parcours. Plus qu'un directeur de thèse, je suis heureux de le compter maintenant comme un ami.

Je tiens également à remercier Christine Graffigne et Mihai Datcu pour avoir accepté d'être les rapporteurs de ma thèse. Leurs conseils ont été d'une grande valeur pour moi et pour la lisibilité, la clarté de mon manuscrit. Un grand Merci également à Jacques Blanc-Talon, Jordy Inglada, Emmanuel Zenou et Jean-Yves Tourneret pour avoir fait partie de mon jury.

J'ai une pensée toute particulière pour deux compagnons de route : Olivier Zammit, d'abord avec qui j'ai partagé mon bureau, et qui, chaque jour ouvrable pendant ces trois ans, a stoïquement supporté mes blagues minables, mes coups de gueule et mes sautes d'humeur. De l'autre côté de l'écran, je salue Alban « lalarve » Linard. Ma maman m'a dit un jour que nous étions ensemble en maternelle, ce qui en ferait mon plus vieil ami... Compagnon de messagerie instantanée à toutes heures, surtout les plus tardives, Lalarve a su m'inculquer les arcanes de la programmation en C++ (dont l'enseignement était malheureusement assorti de réflexions dont la grivoiserie aurait fait rougir le légionnaire

le plus aguerri). Ses leçons ont été d'une valeur inestimable pour mon travail de thèse. À ces deux personnes en particulier, je souhaite tout le bonheur et la réussite professionnelle qu'il est possible d'imaginer, ils le méritent cent fois.

De manière générale, je souhaite également saluer toutes les personnes avec qui j'ai pu échanger des pensées, des joies, des peines à l'INRIA de Sophia Antipolis, pendant ces années : Vincent Lunot, José Grimm, Romain Primet, Demian Wassermann, Damien Lepiller . . . Mes pensées vont en particulier aux doctorants, stagiaires et permanents de l'équipe projet ARIANA : Ian, Laure, Corinne, Pierre, Ting, Alexis, Maria, Éric, Praveen, Michaël, Giuseppe, Raffaele, Daniele, Florent, Guillaume et Florent . . .

Parmi les personnes qui ont énormément compté pendant ces années de thèse, je voudrais saluer mes colocataires : Guillaume, Marie, Florian, Omid et Pooran. Merci pour les instants magiques que nous avons partagés, je crois qu'ils ont contribué à me faire grandir. Un énorme merci à Omid en particulier, à la fois pour son aide inestimable pour la résolution de certains problèmes mathématiques, et pour son sourire, son caractère incroyablement heureux, qui ont fait de la vie en colocation un pur bonheur.

Enfin, un grand merci aux membres de ma famille, qui m'ont apporté un soutien, une confiance et un amour sans faille pendant ces trois années.

मेरी सपना,
कैसे कहूँ कि तुम्हारे बिना जिंदगी कितनी अधूरी है! मेरी अंधेरी जिंदगी को अपनी खूबसूरत हंसी से रोशन करने का बहुत बहुत शुक्रिया! जिंदगी का सफर बहुत लंबा लगता था, लेकिन तुम्हारे साथ के जादू से सारी मंजिल करीब लगती है!!! कभी अपना अप मत बदलना, है गुजारिश यही तुमसे मेरी!

Introduction

Nuts!

GENERAL MC AULIFFE

La détection de changements en traitement d'images est un ensemble de problèmes regroupant des domaines aussi variés que la télédétection ou la microscopie. Il a fait l'objet d'une quantité importante de publications et nombre d'approches méthodologiques ont été introduites afin de tenter de résoudre ce problème, souvent dans des conditions particulières. La principale difficulté de la détection de changements, dans la plupart des applications, est que le problème n'est pas de détecter les changements sur des images, mais sur des scènes sous-jacentes. La résolution de ce problème dans un cadre générique semble quasiment impossible, dans le sens où elle fait appel à un haut niveau d'interprétation. Les résultats les plus encourageants dans la problématique de la détection de changements sont donc tout naturellement liés à des applications bien particulières (médicine de précision, surveillance de construction de bâtiments, . . .). Ceci est souvent lié au fait que l'on peut dans ces cas définir des hypothèses plus fines quant à la modélisation, ce qui facilite notre interprétation.

L'objectif de ce travail de thèse n'est pas de résoudre le problème général de la détection de changements, mais de manière plus réaliste, de tenter d'apporter certaines réponses et méthodologies sur un domaine bien particulier : la télédétection sur des zones urbaines. La variété de scènes urbaines existantes rend déjà cet objectif très complexe à réaliser.

Ce document s'articule en quatre chapitres. Dans le premier chapitre, on fait un descriptif rapide de l'état de l'art au niveau méthodologique (de la détection de changements). Malgré tout, on ne se cantonne pas au domaine de la télédétection, en essayant de présenter certaines méthodologies qui pourraient être utilisables dans notre cadre. Les trois chapitres suivants sont ordonnés dans le sens logique d'une chaîne de traitement « idéale », depuis les deux images brutes, jusqu'à des « objets » de plus haut niveau correspondant à des changements sur la scène.

Le Chapitre 2 correspond à un travail réalisé en collaboration avec Pierre Weiss, également doctorant en dernière année dans le projet Ariana. Il présente une méthodologie originale, fondée sur l'invariance des lignes de niveau pour déterminer un masque pixelique de changements. Cette recherche a été motivée par le fait que les « objets », étudiés

dans les chapitres ultérieurs, ne correspondent pas toujours à des objets réels dans les scènes. Cette nouvelle approche a permis de réduire considérablement ce problème (en proposant deux cartes de changements au lieu d'une, correspondant chacune à des objets existant exclusivement sur une des deux scènes).

Le Chapitre 3 correspond à une première classification fondée sur les radiométries des pixels « changés » fournissant un ensemble de zones connexes. On s'intéresse dans le Chapitre 4 aux propriétés géométriques de ces zones et en particulier, on cherche à affiner notre classification en utilisant leurs orientations.

Chapitre 1

État de l'art

1.1 Différents types de problématiques

Les auteurs de [Radke et al., 2005] définissent le problème général de détection de changements comme *la détection de régions changées dans plusieurs images de la même scène prise à des instants différents*. Cette définition, assez large, inclut n'importe quelle scène pouvant générer une image à l'aide de capteurs conçus par l'homme, ce qui regroupe des applications dans des domaines aussi divers que le biomédical, la sécurité urbaine ou la télédétection. Par ailleurs, les images utilisées pour la comparaison peuvent être réduites au nombre de deux ou, au contraire, former une large séquence vidéo.

Enfin, à scène fixée, les capteurs eux-mêmes peuvent être différents d'une prise de vue à l'autre, augmentant encore l'éventail des problèmes liés à la détection de changements.

En conséquence, la diversité des problèmes évoqués a induit le développement de solutions fondées sur des approches très différentes.

Plusieurs états de l'art ont été publiés ces dernières années sur le sujet. Les auteurs de [Radke et al., 2005] proposent une description des méthodes générales employées en détection de changements, mais ne traitent pas des méthodes de « comparaison post-classification » (*delta classification*), c'est-à-dire des méthodes consistant à classer les différentes images pour ensuite détecter des différences sur les images classifiées. Ils s'intéressent au problème de détection de changements du point de vue méthodologique, mais ne s'intéressent pas à des applications particulières. Cette publication propose aussi quelques pistes en ce qui concerne le travail difficile de l'évaluation de performance en détection de changements. Des revues plus spécifiques sur la détection de changements en télédétection sont également proposées dans [Lu et al., 2004; Coppin et al., 2004]. D'autres publications, sont axées sur des problématiques plus particulières comme les détections de changements sur les images aériennes [Singh, 1989] ou la surveillance d'écosystèmes [Coppin et al., 2004]. Plus récemment, dans le cadre d'une thèse de doctorat, A. Robin présente une taxonomie des différentes méthodes de détection de changements dans [Robin, 2007,

Chapitre 2].

L'objectif de ce travail est la détection de changements sur des paires d'images pan-chromatiques d'observation de la Terre à haute résolution, en particulier en milieu urbain. Il est peu aisé de classer et/ou identifier tous les changements, qui peuvent subvenir en milieu urbain, mais une attention particulière est portée sur les changements au niveau des bâtiments et des véhicules. Les applications envisagées se situent dans les domaines de la surveillance d'occupation des sols et de la sécurité.

En conséquence, ce chapitre n'a pas pour vocation d'effectuer un état de l'art exhaustif de toutes les méthodes utilisées pour la détection de changements, mais plutôt de présenter les grandes familles de méthodes qui ont été employées pour traiter les problèmes de détection de changements dans un couple d'images, et en particulier en télédétection.

La chaîne de traitements successifs, quand il s'agit de détection et de classification de changements, comprend, en général, une étape de prétraitements (incluant notamment le recalage), une étape d'obtention d'un masque de changements, puis une ou plusieurs étapes de classification.

1.2 Prétraitements

1.2.1 Recalage

Le recalage est la mise en correspondance géométrique d'un couple ou d'une série d'images. En effet, les prises de vues d'une même scène peuvent varier suivant la position ou la nature du capteur, ce qui rend nécessaire cette étape. Il s'agit d'un élément important parmi les prétraitements nécessaires à la détection de changements : par exemple, les auteurs de [Day et Khorram, 1998] proposent d'étudier l'impact des erreurs de recalage sur une méthode de détection de changements sur un plan d'occupation des sols. Ils montrent qu'une erreur de recalage de l'ordre d'un cinquième de pixel peut entraîner jusqu'à 10 % d'erreurs de détection. Des résultats du même ordre avaient été également publiés dans [Townshend et al., 1992] en utilisant une méthodologie différente. Ces résultats sont liés à des applications précises et ne peuvent certainement pas être transposés directement à toutes les applications, ni à toutes les techniques de détection de changements ; néanmoins, ils donnent une idée de l'influence du recalage sur la qualité du résultat obtenu en détection de changements. La réalisation d'un recalage correct est un problème en soi et a fait l'objet de diverses publications. Plusieurs revues sont disponibles, comme [Brown, 1992] ou plus récemment [Zitová et Flusser, 2003] . Nous allons maintenant donner un bref aperçu des problèmes et techniques liés à cette opération.

La réalisation du recalage se divise en deux grandes étapes :

La première étape est la recherche et la sélection de points de contrôle , ou points d'amer, correspondant à la même localisation sur les deux images. Cette étape peut être réalisée manuellement ou automatiquement. Dans le second cas, on distingue deux ca-

tégories d'algorithmes, l'appariement fondé sur l'aire ou *area-based matching* (ABM) et l'appariement fondé sur les caractéristiques ou *feature based matching* (FBM). La première, et plus ancienne technique, consiste à comparer une fenêtre de pixels sur une image avec une fenêtre sur l'image de référence. Des critères comme la corrélation croisée ou les moindres carrés sont utilisés pour mesurer le degré de correspondance des deux fenêtres. Les points de contrôle sont alors les centres des fenêtres correspondantes. La FBM consiste à appairer certains objets caractéristiques des deux images (contours, intersections de linéiques, coins ...). La seconde catégorie donne, en général, de meilleurs résultats en matière de correspondance, mais ne permet pas d'avoir une précision subpixelique dans l'appariement.

Une fois les points de contrôle sélectionnés, il reste à effectuer un appariement dense des deux images en appliquant un modèle de transformation :

- Si l'on modélise la surface de la scène à recaler comme un plan, alors, il s'agit d'un problème de recalage rigide (dans des domaines différents de la télémétrie, d'autres conditions peuvent être nécessaires, du fait, notamment, de distorsions du signal). Ce type de problèmes a été largement étudié et plusieurs solutions ont été proposées pour répondre à ce besoin, et sont disponibles dans les logiciels professionnels de traitement d'images (ENVI, ITK ...).
- Ce type de recalage donne des résultats convenables tant que cette modélisation n'est pas aberrante ; toutefois, sur des zones géographiques où le relief est marqué, le recalage rigide produit des erreurs importantes. Dans ce type de situations, le problème devient alors plus complexe. On fait normalement appel aux techniques de recalage non rigide. Ces techniques sont utilisées à l'heure actuelle dans le domaine de l'imagerie médicale, mais souvent restreintes à des problèmes précis où le modèle de surface est connu à l'avance ([Can et al., 2002]). En télédétection, le relief n'est pas forcément connu lors du traitement de données, ce qui complique le recalage . Certains modèles de recalage automatique et non rigide ont déjà été expérimentés. Par exemple, [Hong et Zhang, 2005] se fondent sur une décomposition en ondelettes pour appliquer successivement une FBM sur la composante basse-fréquence, puis une AFM sur la composante haute-fréquence. Ceci permet d'obtenir un grand nombre de points de contrôle pour un maillage fin de la scène. D'autres méthodes de recalage élastique ont été également proposées –entre autres– dans [Kybic et Unser, 2003; Liu et al., 2006; Monasse, 1999]. Récemment, une méthode de détection de changements [Li et al., 2006] a intégré un algorithme novateur de recalage, grâce à une méthode fondée sur les arêtes des bâtiments, avec des résultats très encourageants.

Considérer le problème de recalage comme un simple prétraitement à la détection de changements n'est pas forcément la meilleure solution. Le fait que les différences entre des images puissent nuire à la qualité du recalage montre bien que ces problèmes sont imbriqués l'un dans l'autre. Par exemple, P. Monasse dans [Monasse, 1999] et [Monasse et Guichard, 2000] utilise des méthodes similaires pour traiter ces deux problèmes. En

outre, certaines publications [Chatelain et al., 2007] adressent simultanément ces deux problèmes.

En ce qui concerne ce travail particulier, les images fournies ont été recalées manuellement en deux temps. Dans un premier temps à l'aide du logiciel ENVI, des points de contrôle ont été placés manuellement, puis un recalage en rotation et échelle a été effectué. La composante de translation a été retrouvée grâce à une recherche de similarités dans le domaine spectral à l'aide d'un script Matlab. Néanmoins, la qualité du recalage ne peut pas être considérée comme parfaite et nous avons pris ce facteur en compte dans la mise au point de nos algorithmes.

1.2.2 Mise en correspondance radiométrique

La plupart des opérateurs de bas niveau sont fondés sur la comparaison des radiométries au niveau du pixel. Il est donc essentiel que les radiométries des pixels correspondants soient cohérentes. Parmi les facteurs qui peuvent faire varier les radiométries, on peut citer, des changements d'illumination globale, des différences de nature des capteurs ou bien le vieillissement du capteur entre les différentes prises de vue. La rectification radiométrique des images est d'une importance capitale pour la qualité de la détection de changements, mais ce type de correction est étroitement lié aux techniques ponctuelles évoquées dans la partie suivante de ce manuscrit. Dans notre travail, cela est pris en compte, dans les algorithmes de détection de changements que nous proposons. que nNous ne nous attarderons donc pas sur les corrections radiométriques.

1.3 Méthodes pixelliques

Historiquement, les premières méthodes, encore largement utilisées en détection de changements, reposent sur des opérateurs de comparaison pixel à pixel, plus ou moins sophistiqués. Dans un premier temps, une quantité de changement (CQ) est calculée entre deux pixels de même position sur chaque image. Il s'ensuit une étape de décision sur cette quantité de changement afin de classer chaque pixel comme « changé » ou « non-changé ».

1.3.1 Opérateurs naïfs

Nous allons d'abord présenter les premiers opérateurs, les plus simples, qui ont été utilisés. L'opérateur de quantité de changement le plus direct consiste à comparer les deux images, I_1 et I_2 , pixel à pixel, par une simple différence d'intensité.

$$CQ(I_1, I_2) = I_1 - I_2$$

On pourra notamment utiliser la valeur absolue pour avoir une quantité de changement strictement positive :

$$CQ(I_1, I_2) = |I_1 - I_2|$$

D'autres opérateurs un peu plus complexes ont également été proposés comme, par exemple, dans le domaine de l'imagerie Radar, le ratio d'images [Rignot et van Zyl, 1993], ce qui s'explique par la caractéristique multiplicatif du bruit dans les images radar :

$$CQ = \frac{I_1}{I_2}$$

De façon similaire, les auteurs de [Coppin et Bauer, 1994] proposent de normaliser la différence d'image par l'intensité moyenne entre les deux images.

$$CQ(I_1, I_2) = \frac{I_1 - I_2}{I_1 + I_2}$$

Cette technique permet notamment de discerner les changements intervenus dans des zones d'ombre où la luminosité moyenne est plus faible.

Ces opérateurs pixelliques peuvent fournir des résultats intéressants (*i.e.* caractériser des zones de changement important) lorsque les conditions entre les deux prises de vues sont relativement proches voire identiques, au niveau de l'illumination générale ou des caractéristiques des capteurs. Néanmoins, ces conditions représentent un cas « idéal ». Aussi, quand ces conditions varient, il est souvent intéressant de faire appel à des opérateurs plus souples, souvent issus de la statistique.

1.3.2 Cas multispectral

Bien que ce travail s'applique à des images panchromatiques, il est intéressant d'étudier la façon dont l'information multispectrale est utilisée pour la détection de changements. La logique la plus simple utilisable dans le cas d'une image multicanaux serait de sommer la différence des valeurs du pixel pour chaque canal.

Lorsque l'on a affaire à des images multispectrales, on peut moduler cet opérateur en tenant compte de chaque bande :

$$CQ(I_1, I_2) = \sum_i |I_1^{(i)} - I_2^{(i)}|$$

où les i représentent les indices des bandes. Toutefois, chaque bande n'a pas la même importance selon l'application envisagée et l'on pourra alors considérer d'attribuer un poids spécifique à chaque bande selon sa pertinence :

$$CQ(I_1, I_2) = \sum_i \alpha_i |I_1^{(i)} - I_2^{(i)}|$$

Pour des applications particulières, des indices plus complexes ont été mis au point et sont aujourd'hui utilisés comme indicateurs. Il est donc naturel d'utiliser ces indices

comme information de base pour la comparaison pixel à pixel. Parmi ceux-ci, on peut citer, notamment, le NDVI (Indice Différentiel de Végétation Normalisé) qui exploite les informations des bandes spectrales « rouge » et « proche infra-rouge » :

$$NDVI = \frac{NIR - RED}{NIR + RED}$$

Proposé en premier lieu dans [Rouse et al., 1973], il a été utilisé dans le contrôle de la déforestation notamment, dans [Lyon et al., 1998]. Dans un autre contexte, [Stefanov et al., 2001] utilisent le ratio SAVI, introduit dans [Huete, 1988] :

$$SAVI = \frac{NIR - RED}{NIR + RED + L}(1 + L)$$

au sein d'un système expert de détection de changements d'occupation des sols dans la périphérie de Phoenix, Arizona.

1.4 changement de contraste globaux

1.4.1 Opérateurs statistiques

Afin d'améliorer la robustesse à des variations globales d'illumination entre les prises de vue, on fait souvent appel à des techniques dérivées de l'analyse en composantes principales (ACP). Bien qu'ils prennent en compte des données relatives à l'ensemble de l'image, les algorithmes fondés sur l'ACP agissent, en général, au niveau du pixel et non pas au niveau de zones, on peut donc considérer ces méthodes comme pixelliques. Cet outil a été utilisé dans plusieurs configurations :

- analyse multibandes-monotemporelle ;
- analyse monobande multitemporelle ;
- analyse multibandes-bitemporelle.

On cherche de cette manière à discriminer les pixels changés. Ces différentes applications sont passées en revue dans [Inglada, 2001]

Cette approche atteint rapidement ses limites lorsque les conditions générales d'illumination de la scène varient. Pour pallier ce problème, l'opérateur principal utilisé est

1.4.2 Réseaux de neurones

D'autres outils plus récents, comme les réseaux de neurones ont également été mis en œuvre pour détecter les changements. Ainsi, dans [Clifton, 2003], des couples d'images de moyenne résolution (Landsat) sont analysés. Ici, l'objectif est d'utiliser l'analyse vectorielle afin de discriminer les changements « naturels » des changements non prévisibles. Des

essais ont aussi été effectués sur des images aériennes à très haute résolution $< 0.5m$ avec des résultats intéressants.

Plus récemment, d'autres études [Patra et al., 2007] ont utilisé une approche similaire sur des images multispectrales (n bandes), en automatisant la phase d'apprentissage du réseau de neurones. Pour cela, ils effectuent dans un premier temps une classification par k -moyennes sur le vecteur de changements :

$$CQ = (int) \sqrt{\sum_{i=1}^n (I_1^{(i)} - I_2^{(i)})^2} \quad (1.1)$$

Ensuite, les auteurs s'appuient sur le fait que les pixels non-changés et changés auront respectivement une faible et une forte QC pour sélectionner pour la base d'apprentissage des pixels pour dont ils sont certains de l'état. En pratique, ils choisissent les pixels non-changés et changés dans les sphères de centre respectif 0 et 255 et de rayon respectifs $|\mu_{NC}|$ et $|255 - \mu_{CH}|$, où μ_{NC} et μ_{CH} sont les centroïdes respectifs des classes « non-changé » et « changé ».

1.4.3 Prise de décision

Une fois la quantité de changement calculée, une étape de décision est nécessaire afin de déterminer de façon dure si un pixel est considéré comme « changé » ou « non-changé ». Là aussi, plusieurs méthodologies ont été envisagées et mises en application. Parmi les techniques les plus simples, on peut citer, par exemple, les simples seuillages empiriques. D'autres techniques font appel à des composantes markoviennes pour donner une détection régulière.

Seuillages

Le seuillage est l'opérateur le plus direct utilisé pour la prise de décision. La difficulté relative à cette méthode est évidemment l'estimation du seuil optimal qui peut être variable selon les images utilisées. Dans un cadre tout à fait différent, des seuillages plus complexes ont été utilisés comme le seuillage par hystérésis [Canny, 1986] pour la détection de zones brûlées [Zammit et al., 2008].

Décision Bayésienne

La décision bayésienne peut être un outil intéressant pour déterminer le seuil optimal. Elle permet d'utiliser une connaissance *a priori* (souvent relative à un modèle) dans la décision. Par exemple, elle est utilisée dans [Bruzzone et Prieto, 2002], pour décider de la probabilité de changements à l'aide d'une approche par champs de Markov (il s'agit d'une approche un peu similaire à celle que nous proposons dans le Chapitre 2).

1.5 Approches « contextuelles »

Les approches fondées sur les pixels ont des limitations importantes dues à leur nature même. Plusieurs auteurs ont proposé d'utiliser des informations contextuelles ou géométriques afin d'améliorer la qualité de la détection. On cherche avec ces approches, non plus à détecter des pixels changés, mais des zones plus étendues. Ceci limite également le nombre de pixels isolés qui font l'objet d'une mauvaise classification. On peut faire appel à des structures de plus ou moins haut niveau d'interprétation pour améliorer la classification. On peut citer notamment les champs de Markov, les lignes de niveau, la notion de texture, ou des objets de haut niveau d'interprétation.

1.5.1 Subdivision de l'image

Parmi les méthodes contextuelles classiques, une des plus simples consiste à diviser l'image en zones régulières (le plus souvent à l'aide d'une grille, parfois à l'aide d'une segmentation préalable) et à donner une estimation du changement au niveau de chaque zone et non plus au niveau des pixels [Borchani et al., 2004]. Ce type d'approches présente deux défauts principaux : une perte de précision importante et une segmentation relativement arbitraire (dans le cas de la grille) ou un choix difficile au niveau de la segmentation à adopter dans le second cas.

1.5.2 Voisinages

On a souvent recours à la notion de voisinage, pour avoir une carte de changements régulière et éviter les phénomènes de bruit. Ainsi, dans [Patra et al., 2007], chaque entrée du réseau de neurones n'est pas un pixel isolé, mais un pixel accompagné de son voisinage 8-connexe. Cet exemple est assez révélateur de la façon dont la notion de voisinage est intégrée au sein des systèmes plus complexes de détection de changements.

1.5.3 Approche markovienne, régularisation

Les champs de Markov ont été utilisés à plusieurs reprises. Dans le cadre de la vidéo-surveillance, par exemple dans [Aach et Kaup, 1995], on a recours aux champs de Markov afin d'introduire une connaissance *a priori* sur le masque de changements et ainsi améliorer le compromis entre fausses alarmes et faux négatifs. Dans [Wiemker, 1997], on fait appel à une composante markovienne, afin de modéliser la densité de probabilité conditionnelle d'un algorithme de décision bayésienne.

1.5.4 Texture

Plusieurs publications proposent d'utiliser l'information texturale afin de faciliter la détection de changements. Dans le cadre de la surveillance de l'utilisation des terres arables, des filtres de Gabor ont été utilisés dans [Yang et Lishman, 2003] pour créer un ensemble de descripteurs de texture. Ces descripteurs sont ensuite utilisés comme base pour un classifieur.

1.5.5 Ensembles de niveau

Une autre notion de voisinage, mais basée sur les ensembles de niveau a été utilisée en détection de changements. Dans [Monasse et Guichard, 2000], chaque image est transformée en un arbre de formes qui sont les composantes connexes des ensembles de niveau. La détection de changements se fonde sur la reconnaissance de similarités entre les deux arbres. Cette technique semble donner des résultats convaincants sur des images de vidéo surveillance en milieu urbain. Nous reviendrons un peu plus en détail sur ces techniques dans le Chapitre 2.

1.5.6 δ -classification

Certains auteurs ont eu recours aux techniques de comparaison « post-classification ». Par exemple, dans [Borchani et al., 2004], on utilise les informations statistiques liées à la texture pour classer, à l'aide d'un réseau de neurones, des parcelles d'images en deux catégories : urbain et rural. Une fois cette étape effectuée sur des couples d'images, les zones correspondantes de chaque paire sont confrontées afin de détecter des changements via des classifications contradictoires. Afin d'éviter des fausses alarmes, seuls des groupes de zones connexes sont finalement retenus. Cette approche présente plusieurs inconvénients : d'une part, elle nécessite un apprentissage du réseau de neurones et deuxièmement, elle ne permet pas une localisation fine des changements intervenus. Dans certains cas, comme dans [Bruzzone et Prieto, 2000], la segmentation *a priori* a un intérêt dans le sens où les parcelles segmentées représentent directement l'information recherchée.

De manière plus générale, ces techniques présentent l'inconvénient majeur de requérir un traitement préalable de chaque image du couple ou de la série multitemporelle, et par conséquent la résolution d'un problème non-trivial de détection ou de classification. L'information de différence n'est en général pas utilisée dans ce traitement préalable. En revanche, elles sont largement utilisées en coordination avec un Système d'Information Géographique (SIG), souvent, pour mettre à jour les informations contenues dans les SIG. On fait alors appel assez souvent à des méthodes de haut niveau introduisant des relations entre les objets présents dans le SIG et les objets détectés par la prise de vue. Par exemple, dans [Alboody et al., 2008], les auteurs décrivent des relations topologiques qualitatives entre les objets obtenus à l'aide, entre autres, du système *region connection*

calculus (RCC-8). Ce système décrit six relations possibles entre deux objets : disjonction, tangence, intersection partielle, égalité stricte, tangence interne et inclusion stricte. Les deux dernières relations sont réciproques, portant le total à huit.

1.6 Systèmes Experts

Des systèmes experts, beaucoup plus complexes ont également été mis en œuvre. L'objectif d'un système expert est, en général, d'utiliser une base de connaissances *a priori* afin de valider des hypothèses émises sur les données. La conception des systèmes est suffisamment modulaire afin d'ajouter facilement de nouvelles connaissances (enrichissement d'un SIG, par exemple). Dans le cadre de la détection de changements, des exemples de systèmes experts ont été proposés dans [Wang, 1993], ou plus récemment dans [Stefanov et al., 2001]. Des travaux sont également en cours dans les instituts géographiques, dans le cadre de la mise à jour des SIG.

1.7 Conclusion

Les méthodes de détection de changements sont assez variées et dépendent très souvent de l'application pratique sous-jacente (application militaire, surveillance de croissance urbaine, surveillance des parcelles agricoles . . .). On peut cependant distinguer une grande famille de méthodes pixelliques qui s'attachent à calculer une quantité de changements dans un premier temps, puis à proposer une étape de décision. D'autres méthodes font appels à des objets de plus haut niveau, que ce soit dans l'interprétation (détection des arêtes de bâtiments, delta-classification) ou grâce à l'utilisation de connaissances *a priori*. Par ailleurs, on remarque que les meilleurs résultats sont souvent obtenus en faisant des hypothèses assez fines sur le problème étudié. Nous verrons dans les chapitres suivants que nous ne dérogeons pas à cette règle, en proposant des méthodes assez spécialisées pour la détection de bâtiments, d'ombres, de véhicules.

Chapitre 2

Obtention d'un masque de changements

Rick : You are closing me ? On what grounds ?

Renaud : Shocked, I am shocked to discover that gambling is going on, here

croupier : your winnings, sir.

Renaud : Oh, thank you!

CASABLANCA

Dans ce chapitre, nous présentons des méthodes de bas niveau permettant d'obtenir un masque de changements. Nous avons vu dans le Chapitre 1 que la difficulté est de trouver un opérateur à la fois suffisamment discriminant au niveau des faux positifs, et particulièrement robuste aux changements de paramètres (illumination, qualités et position du capteur ...). Nous présentons, dans un premier temps un opérateur proposé à la fin des années 90, fondé sur l'analyse en composantes principales et testons les qualités et limites de cette approche. Dans une deuxième partie, nous présentons une modélisation originale des scènes urbaines et un algorithme tirant parti de cette modélisation pour détecter les changements. Ce travail se fonde sur l'analyse de l'invariance vis-à-vis des changements d'illumination des lignes de niveau d'une image. Nous montrons que si la surface d'une scène a une composante de réflectance lambertienne et que la source de lumière est directionnelle, alors une condition nécessaire pour que les lignes de niveau de l'image soient invariantes aux changements d'illumination est que la scène 3D soit développable et que son albédo satisfasse certaines contraintes géométriques. Ensuite, nous montrons que les lignes de niveau sont « quasi » invariantes pour des scènes composées de surfaces développables par morceaux. De telles surfaces correspondent à la plupart des structures urbaines. Ceci nous permet, alors, de proposer un algorithme très rapide de détection de

changements sur des paires d'images représentant des scènes urbaines en télédection. Enfin, cet algorithme est testé sur des exemples jouets (des scènes modélisées en OpenGL) et sur des images réelles panchromatiques. La comparaison des performances de l'algorithme avec d'autres approches classiques lui donne l'avantage sur les plans théorique aussi bien qu'expérimental.

2.1 Mise en œuvre d'un algorithme de normalisation globale de contraste

Dans cette partie, nous présentons un algorithme bas-niveau de détection de changements. Cet algorithme, qui repose principalement sur l'analyse en composantes principales, a été proposé par R. Wiemker dans [Wiemker, 1997]. À l'origine, il a été utilisé pour détecter les changements dans des images multispectrales. Néanmoins, le recours à l'analyse en composantes principales a déjà été envisagé dans le cadre d'images panchromatiques [Inglada, 2001].

2.1.1 Description de l'algorithme

Les techniques fondées sur l'analyse en composantes principales cherchent à déterminer une caractérisation linéaire ou affine des pixels dans l'espace des couples des radiométries. Cette caractérisation suit normalement la forme générale du nuage de points étudié. Dans le cas de la détection de changements, le nuage que l'on cherche à détecter est le nuage correspondant aux pixels non-changés. D'autres pixels secondaires, représentant certains pixels changés sont également présents. Or, ces pixels secondaires sont souvent pris en compte dans l'analyse en composantes principales. Une des contributions de l'algorithme présenté est l'utilisation d'un procédé itératif permettant de se soustraire à cette influence. Le principe de l'algorithme se fonde sur une hypothèse assez réaliste, dans le cas où les changements entre les deux scènes ne sont pas généralisés. Dans ce contexte, une simple analyse en composantes principales devrait être biaisée par les nuages de points secondaires, mais néanmoins être relativement proche du nuage principal. L'idée est donc de procéder itérativement à des analyses en composantes principales, tout en pondérant à chaque itération les points de l'espace de façon décroissante vis-à-vis de la distance qui les sépare de l'axe principal à l'itération précédente.

Dans cette partie, on note S l'ensemble des pixels (ou sites). $\forall s \in S$, on appelle $CQ(s)$ la quantité de changement du pixel s . Plus précisément, la quantité de changement est définie comme l'intensité de la deuxième composante principale.

$$CQ(s) = g_1(I_1(s) - \mu_1) + g_2(I_2(s) - \mu_2) \quad (2.1)$$

où $G = (g_1, g_2)^T$ est le deuxième vecteur propre de la matrice de covariance des couples

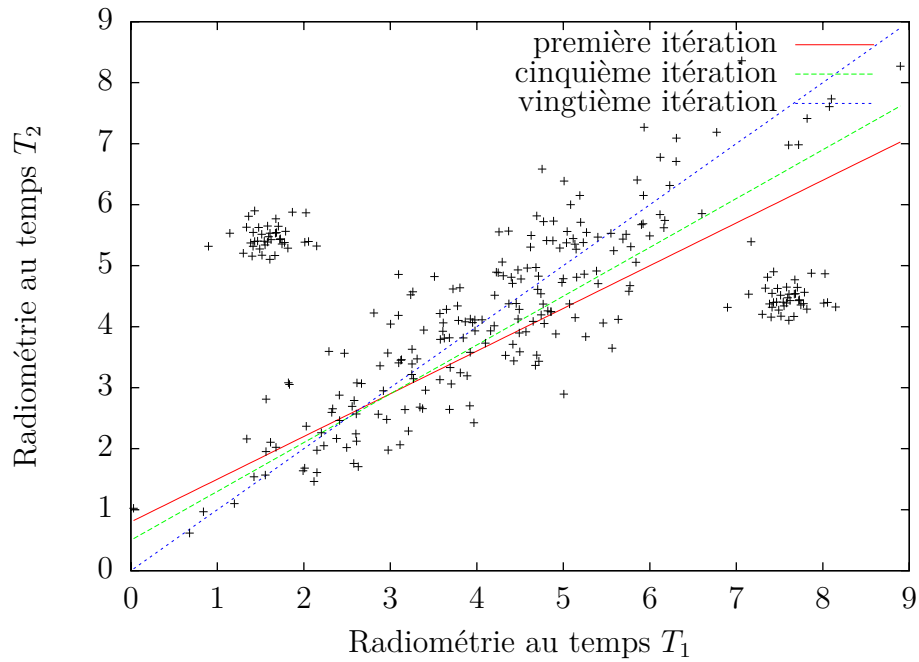


FIGURE 2.1 – Graphe des radiométries au temps T_2 en fonction des radiométries au temps T_1 illustrant le principe de l’algorithme de R. Wiemker : la première analyse en composantes principales est fortement biaisée par les nuages des points « changés » et les itérations successives corrigent ce biais en pondérant les points avec leur distance à l’axe principal.

de radiométries $(I_1(s), I_2(s))$. Le poids de chaque pixel sera donc une fonction décroissante de la distance pour l'itération suivante. Dans notre cas, l'auteur propose une approche bayésienne : les populations de pixels changés et non-changés sont modélisées par des distributions gaussiennes :

$$p(s|\Lambda) = \frac{1}{\sqrt{2\pi\sigma_\Lambda^2}} \exp\left(-\frac{CQ(s)}{2\sigma_\Lambda^2}\right) \quad (2.2)$$

où Λ est la classe à laquelle appartient le pixel : CH (changé) ou NC (non-changé). Une probabilité d'appartenance à chaque classe *a priori* est donnée en considérant les probabilités *a posteriori* obtenues à l'itération précédente :

$$p(\Lambda) = \sum_X \frac{p(\Lambda|X)}{\text{card}(I_1)} \quad (2.3)$$

où $p(\Lambda|X)$ est la probabilité *a posteriori* obtenue à l'itération précédente et $\text{card}(I)$ représente la taille de l'image.

La probabilité *a posteriori* est calculée en fonction des deux probabilités précédentes :

$$p(\Lambda|X) = \frac{p(\Lambda)p(X|\Lambda)}{p(X)} \quad (2.4)$$

2.1.2 Régularisation par champs de Markov

L'approche suggérée dans [Wiemker, 1997] utilise également une composante markovienne dans l'algorithme, mais les résultats obtenus sont alors assez discutables et la méthodologie plus complexe. Aussi, nous avons préféré adopter une régularisation *a posteriori* également fondée sur les champs de Markov. Ce post-traitement permet d'obtenir *in fine* le masque de changements.

Les données utilisées sont alors les probabilités de changement par pixel *fournies par la méthode précédente*. On les note dans cette partie $\mathbf{P}_{CH}(s)$ et $\mathbf{P}_{NC}(s)$.

Le champ de Markov μ que l'on va utiliser a pour espace des configurations $(\Omega, \mathcal{F}) = (\{CH, NC\}, \varepsilon)^S$ (voir Annexe A). On se fonde sur le modèle d'Ising [voir Kindermann et Snell, 1980, Chapitre 2] pour établir les probabilités *a priori* d'appartenance à une classe. Pour toute configuration $\omega \in \Omega$, et pour tout site $s \in S$ on a :

$$p(\omega_s|\omega_{s'}, s' \in V(s)) = \frac{1}{Z} \exp\left(\sum_{s' \in V(s)} (1 - \delta(\omega_s, \omega_{s'}))\right) \quad (2.5)$$

où Z est une constante de normalisation, $V(s)$ représente le voisinage du pixel s et $\delta(\omega_s, \omega_{s'})$ est égal à 1 si $\omega_s = \omega_{s'}$ et 0 sinon.

Le modèle d'attache aux données utilise les probabilités données précédentes afin de calculer la vraisemblance d'un pixel :

$$p(\mathbf{P}_{\omega_s}(s)|\omega_s) = \exp\left(-\frac{1}{\mathbf{P}_{\omega_s}(s)}\right) \quad (2.6)$$

La probabilité *a posteriori* est définie en fonction du terme *a priori* et de l'attache aux données :

$$p(\omega_s|\mathbf{P}_{\omega_s}(s)) = \frac{1}{Z'} \exp\left(-\frac{1}{\mathbf{P}_{\omega_s}(s)} - \beta \sum_{s' \in V(s)} (1 - \delta(\omega_s, \omega_{s'}))\right) \quad (2.7)$$

où le paramètre $\beta > 0$ permet de régler le poids relatif des deux termes dans l'expression finale de la probabilité *a posteriori*. Pour une configuration ω , la probabilité *a posteriori* s'écrit :

$$p(\omega|\mathbf{P}_\omega) = \frac{1}{Z'} \exp\left(-\sum_{s \in \mathcal{S}} \frac{1}{\mathbf{P}_{\omega_s}(s)} - \beta \sum_{c \in \mathcal{C}} (1 - \delta(\omega_{s_c}, \omega_{s'_c}))\right) \quad (2.8)$$

où \mathcal{C} représente l'ensemble des cliques de cardinal 2 du champ de Markov, s_c et s'_c représentent les deux sites de la clique c . Afin de trouver la configuration qui maximise cette probabilité, on utilise le schéma de recuit simulé, décrit dans l'Annexe A (voir en particulier la description de l'Algorithme A.1).

2.1.3 Résultats

L'algorithme a été testé sur les images de l'aéroport et du centre ville d'Abidjan, en Côte d'Ivoire, fournies par la DGA. Elles sont issues du capteur Quickbird, à la résolution de 61 cm.

Aéroport d'Abidjan

Les images originales de l'aéroport sont représentées sur la Figure 2.2. À titre de comparaison, une simple différence des radiométries a été placée à côté de la quantité de changement sur la Figure 2.3. On remarque que l'algorithme de Wiemker est légèrement plus robuste au niveau des différences globales d'illumination (notamment au niveau du bitume) que la simple différence. En ce qui concerne les probabilités de changement et le masque de changements final, (cf. Figure 2.4), on note que les différences sont beaucoup plus marquées. Par exemple, bien que les avions soient correctement détectés, on note une quantité importante de fausses alarmes au niveau de la végétation entourant le tarmac. D'autres erreurs sont notamment liées aux ombres portées des bâtiments de l'aéroport.

La régularisation par champ de Markov donne des résultats satisfaisants en éliminant la plupart des fausses alarmes isolées, par rapport à l'image de probabilité de changement. Les fausses alarmes au niveau des bâtiments de la zone aéroportuaire, en revanche, ont une taille trop importante pour être éliminées.

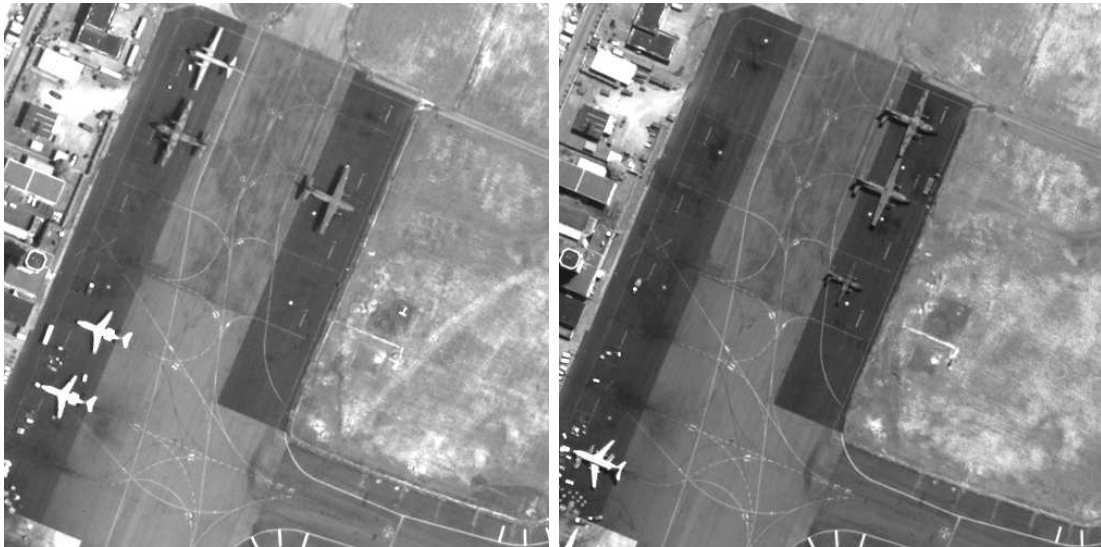


FIGURE 2.2 – Prises de vue de l'aéroport d'Abidjan. Gauche : le 04/02/2003, droite : 07/07/2003. Images Quickbird (61 cm de résolution), © DigitalGlobe, distribution Spot Image, fournies par la DGA.

Zone commerciale d'Abidjan

Nous présentons les images de la zone commerciale d'Abidjan sur la Figure 2.5. Elles ont été prises aux mêmes dates que les images de l'aéroport. Entre ces deux images, la plupart des changements réels est due aux mouvements de véhicules. Les résultats de l'algorithme de R. Wiemker sont présentés sur la Figure 2.6. On remarque que la plupart des fausses alarmes sont dues à des ombres portées d'une part, et des changements locaux de contraste qui ne suivent pas la composante principale. Ces derniers sont assez importants, principalement au niveau des toits. Si les véhicules sont, en général, détectés correctement, on atteint un taux de fausse alarme assez important à cause de ce facteur. D'autres fausses alarmes sont dues à des erreurs de parallaxe (malheureusement plus difficiles à corriger).

2.1.4 Conclusion

Si dans certaines conditions, les approches fondées sur l'analyse en composantes principales donnent des résultats intéressants, on arrive rapidement aux limites de ce type d'algorithmes lorsque l'on a des différences importantes dans les conditions d'illumination. À partir de là, deux stratégies sont envisageables. La première stratégie consiste à essayer de classer les changements détectés de façon suffisamment précise pour discriminer ce type de fausses alarmes, soit vis-à-vis de leurs composantes radiométriques, soit



FIGURE 2.3 – Traitements sur l’aéroport d’Abidjan. Haut : Simple différence (en valeur absolue), bas : Quantité de changements (valeur absolue)

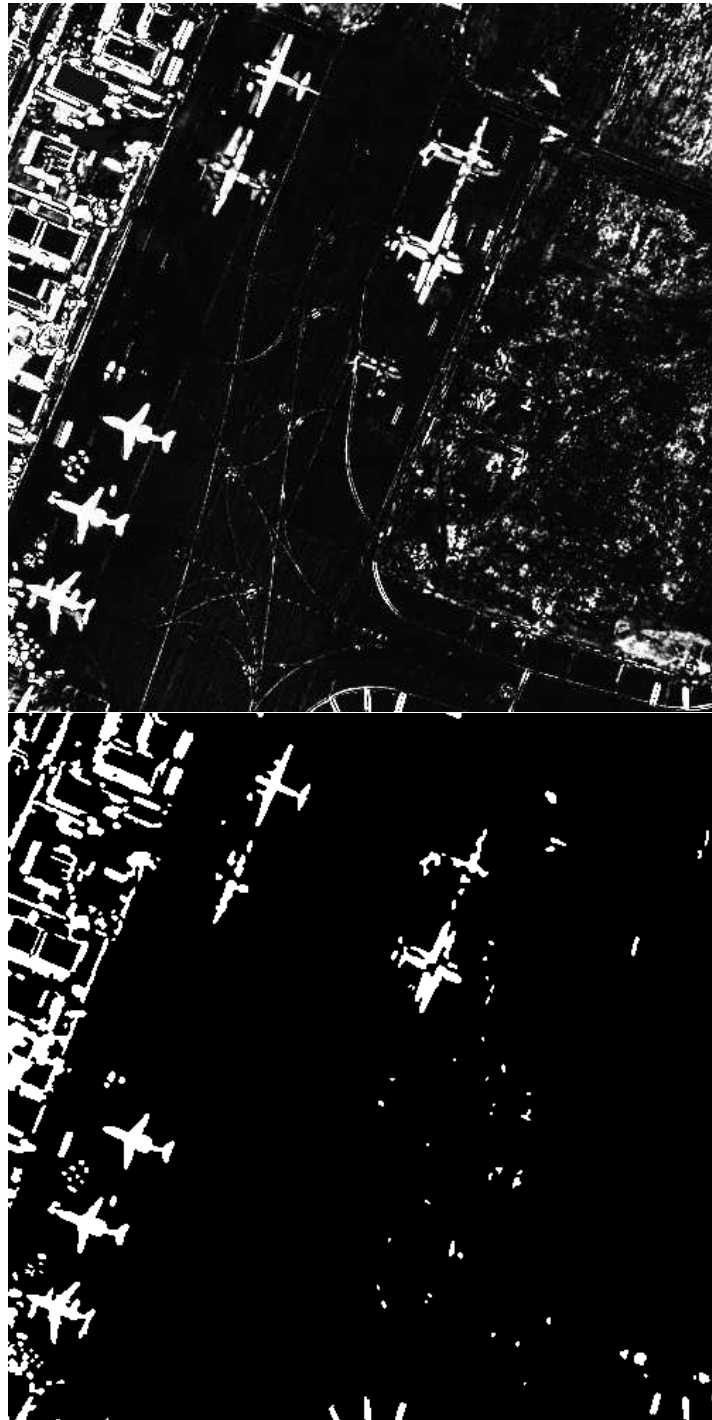


FIGURE 2.4 – Aéroport d'Abidjan , Haut : Probabilité de changements après convergence.
Bas : Masque de changement obtenu après régularisation par un champ de Markov.



FIGURE 2.5 – Images originales d’une zone commerciale d’Abidjan , © DigitalGlobe, distribution Spot Image fournies par la DGA.

en intégrant des données plus contextuelles, telles que la position ou des informations géométriques. C’est cette approche qui est décrite dans les Chapitres 3 et 4. Néanmoins, on peut également essayer de suivre une stratégie différente : la recherche d’un nouvel algorithme bas niveau de détection de changements, plus robuste aux changements dus aux illuminations. Cette stratégie a été adoptée pendant la dernière partie du travail de thèse. Les résultats théoriques et expérimentaux sont décrits dans la Partie 2.2 ci-dessous.

2.2 Une approche fondée sur les lignes de niveau

2.2.1 introduction

La recherche d’une caractéristique invariante à l’illumination est un problème récurrent en traitement d’images (on pourra se référer par exemple à [Chen et al., 2008; Mundy et Zisserman, 1992]. Nombre d’applications, comme la reconnaissance, l’égalisation et rehaussement de contraste, le recalage d’images et la détection de changements bénéficieraient de tels attributs.

À cet égard, les ensembles de contours sont probablement l’attribut le plus utilisé [Marr, 1982]. Les contours sont généralement dus à des discontinuités au niveau de l’élévation ou de l’albédo de la surface et sont donc invariants aux changements d’illumination dans le sens où ils apparaissent sur une image pour la plupart des conditions d’illumination. Récemment, Caselles et Col.[Caselles et al., 1999a] ont proposé une composante

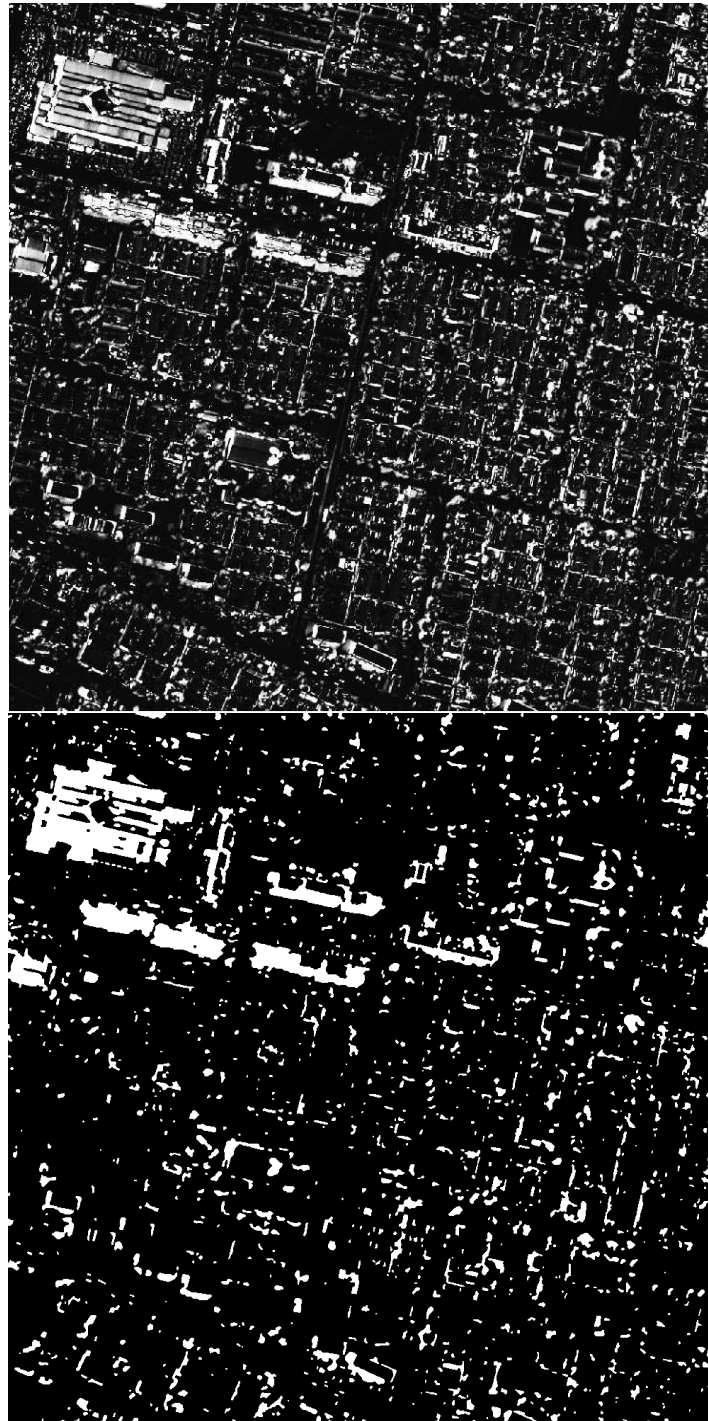


FIGURE 2.6 – Haut : probabilité de changements après convergence sur une zone commerciale d'Abidjan, bas : masque de changements après régularisation par champs de Markov

alternative. Les auteurs montrent que les lignes de niveau sont invariantes aux changements de contraste locaux. L'ensemble des lignes de niveau (ou carte topographique) présente deux avantages importants par rapport aux contours. Premièrement, en modélisation discrète, la définition des contours repose en général sur le réglage d'un seuil contrairement à la carte topographique. En second lieu, la carte topographique permet de reconstruire une image. Certaines applications des cartes topographiques – comme la détection de changements [Monasse et Guichard, 2000; Ballester et al., 2000], le recalage [Monasse, 1999] ou le rehaussement de contraste [Caselles et al., 1999b] ont été récemment proposées avec de bons résultats. Néanmoins, les changements de contraste locaux [Caselles et al., 1999a] ne peuvent pas modéliser tous les changements d'illumination. (*i.e.* les variations des conditions d'illumination). La Figure 2.7 présente un exemple de surface dont les lignes de niveau ne sont pas invariantes vis-à-vis des changements d'illumination. En conséquence, ces méthodes ne sont pas complètement justifiées

Dans cette partie, nous présentons les conditions nécessaires sur la géométrie de la scène pour que les lignes de niveau soient invariantes aux variations de la direction d'incidence de la lumière. Dans un premier temps, nous montrons qu'elles sont effectivement invariantes si et seulement si la scène est développable et que son albédo varie uniquement selon certaines directions. Ces surfaces présentent un intérêt limité, car elles modélisent peu de scènes réelles, ce qui nous mène à l'étude de l'invariance pour des surfaces développables par morceaux. Nous montrons que les images de ces objets ont des lignes de niveau « quasi » invariantes. Le résultat le plus proche de nos conclusions est fourni dans [Chen et al., 2008], dans lequel les auteurs montrent que la direction du gradient est « quasi » invariante par rapport à la direction d'illumination pour la plupart des scènes.

Nous tirons ensuite parti des lignes de niveau pour réaliser un algorithme adapté à la détection de changements sur des couples d'images recalées. La plupart des structures urbaines (voire anthropiques) peuvent être considérées comme développables par morceaux. Par conséquent, cet algorithme est particulièrement bien adapté pour la détection de changements sur des paires d'images d'infrastructures urbaines ou des images urbaines haute résolution en télédétection. Précédemment, d'autres auteurs ont tenté d'élaborer des algorithmes de détection de changements robustes aux variations d'illumination. La revue sur la détection de changement, [Lu et al., 2004] ou plus récemment [Radke et al., 2005] en donnent quelques exemples. Les premières tentatives utilisent, en général, l'égalisation linéaire de contraste [Wiemker, 1997], des moyennes locales et la normalisation de la variance [Lillestrand, 1972], l'utilisation du ratio entre les deux images [Toth et al., 2000; Watanabe et al., 1998], ou le rehaussement global de contraste [Theiler et Perkins, 2007]. Ces approches ont en commun le défaut de ne pas prendre en compte toutes les variations d'illumination, ce qui a abouti à un assez grand nombre de fausses alarmes au cours de nos tests. Certains travaux sont plus proches de notre approche. Dans [Watanabe et al., 1998], les auteurs supposent les toits plats et leur réflectance uniquement lambertienne. Sous ces hypothèses, ils montrent que le ratio entre les deux images peut

être utilisé afin de détecter les ombres et les changements quelle que soit la direction d'illumination. Leurs hypothèses sur la scène sont néanmoins bien plus restrictives que les nôtres. Dans [Monasse et Guichard, 2000; Ballester et al., 2000], les auteurs proposent un algorithme de détection de changements intéressant, qui s'appuie largement sur les lignes de niveau. Néanmoins, on montre que la définition qu'ils fournissent des lignes de niveau ne les rend pas invariantes aux conditions d'illumination.

Enfin, on compare les différentes approches sur des images de synthèse et des images Quickbird haute résolution. On constate alors que ces expériences donnent sensiblement l'avantage à notre méthode. Notons toutefois que bien d'autres algorithmes sont basés sur des caractéristiques invariantes. Par exemple, [Tang et Prinet, 2007] utilisent des caractéristiques telles que le détecteur de coins de Harris, alors que [Black et al., 2000] évalue un ré-échelonnement non local de l'illumination. Ces méthodes restent difficiles à comparer à notre algorithme, dans le sens où leur mise en œuvre requiert plusieurs étapes de décision et plusieurs paramètres intuitifs.

2.2.2 Notations, définitions et modélisation

Notations

Soit Ω un sous-ensemble connexe de \mathbb{R}^2 . Soit $u : \Omega \rightarrow \mathbb{R}$ une fonction C^2 . $\nabla u = [u_1, u_2]$ représente le gradient de u , $\nabla^2 u = \begin{bmatrix} u_{11} & u_{12} \\ u_{12} & u_{22} \end{bmatrix}$ représente son hessien. Soit $p : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $J(p)$ représente le jacobien de p . Soient x et y des éléments de \mathbb{R}^n . $x \parallel y$ signifie que x et y sont colinéaires (0 est colinéaire à tout élément de \mathbb{R}^n). Soit $\omega \subset \Omega$. On note alors $\bar{\omega}$ la fermeture de ω (vis à vis de la topologie induite par la métrique euclidienne). L'intérieur de ω (défini comme le plus grand ouvert de ω au sens de l'inclusion) sera désigné indifféremment par $\overset{\circ}{\omega}$ ou $\text{int}(\omega)$. $\mu_{\mathbb{R}^n}$ est la mesure de Lebesgue dans \mathbb{R}^n . $\mathcal{M}_{m,n}$ correspond à l'espace des matrices à m lignes et n colonnes.

Les notations qui suivent sont illustrées sur la Figure 2.8. Ω représente le plan image. C'est un ouvert connexe de \mathbb{R}^2 . $\mathbf{\Omega}$ représente le plan « objet ». $s : \mathbf{\Omega} \rightarrow \mathbb{R}$ désigne l'élévation de la scène. $N(\mathbf{x})$ représente la normale à la surface de la scène au point $(\mathbf{x}, s(\mathbf{x}))$. $P : (\mathbf{x}, z) \mapsto x$ est la projection de perspective sur Ω . p est alors l'application définie par :

$$\begin{aligned} p : \mathbf{\Omega} &\rightarrow \Omega \\ \mathbf{x} &\mapsto P(\mathbf{x}, s(\mathbf{x})) \end{aligned} \tag{2.9}$$

On suppose que p est un difféomorphisme C^1 . Un difféomorphisme étant bijectif, cela implique que la caméra peut « voir » tous les points de la surface.

Tout au long de cette partie, on prend comme convention de noter en gras les symboles relatifs au plan objet et en régulier les symboles relatifs au plan image. Par exemple, $\mathbf{\Omega} =$

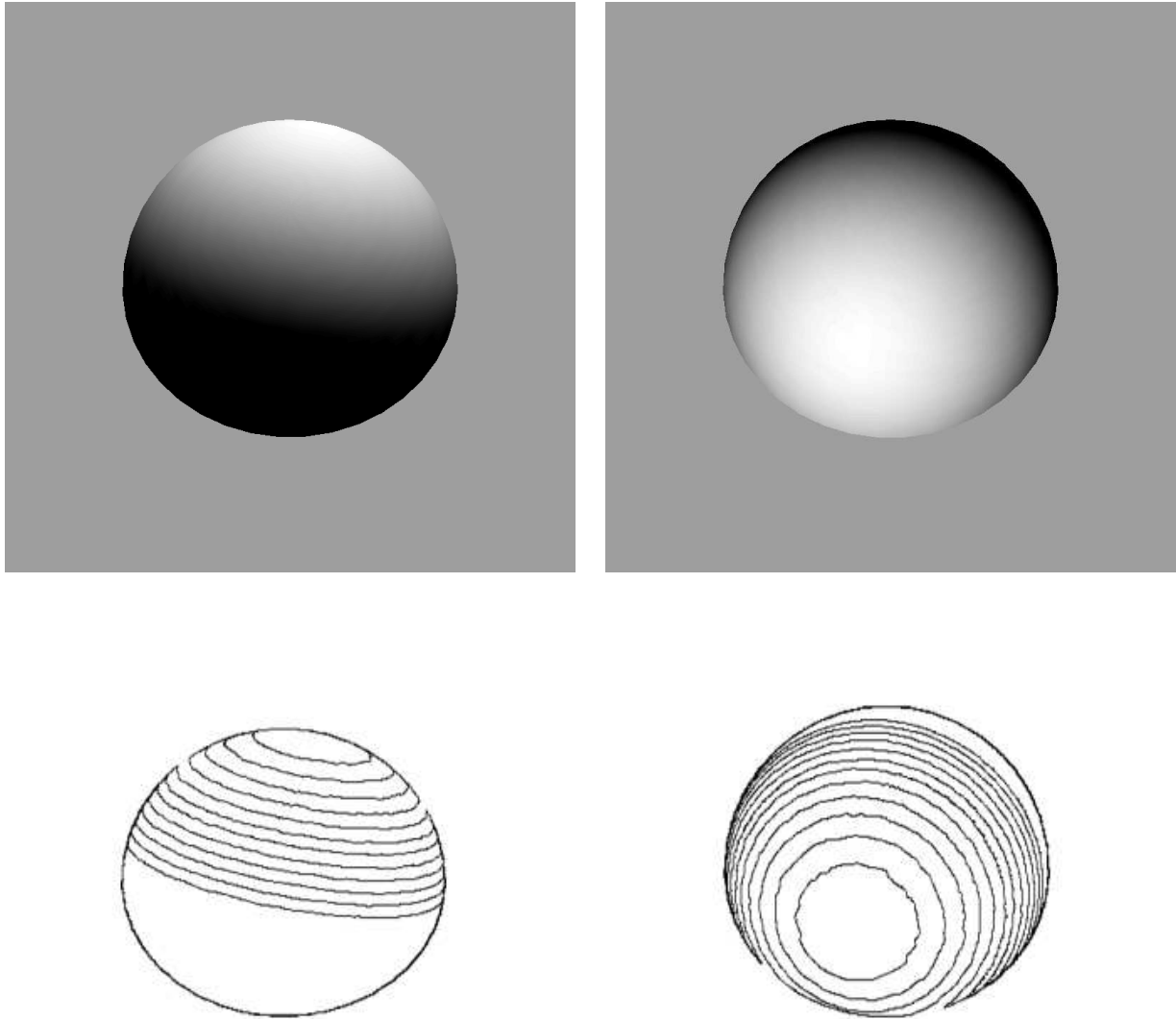


FIGURE 2.7 – Haut : images d'un dôme en 3 dimensions illuminé d'après le modèle de Phong [Phong, 1975] en utilisant deux directions d'incidence différentes. Bas : quelques lignes de niveau associées aux illuminations respectives.

$p^{-1}(\Omega)$ représente le plan objet. De même si $x \in \Omega$, on définit implicitement $\mathbf{x} = p^{-1}(x)$ qui est un point de Ω .

Enfin, l est un vecteur de \mathbb{R}^3 . $\frac{l}{|l|}$ représente la direction d'incidence de la lumière et $|l|$ représente son intensité.

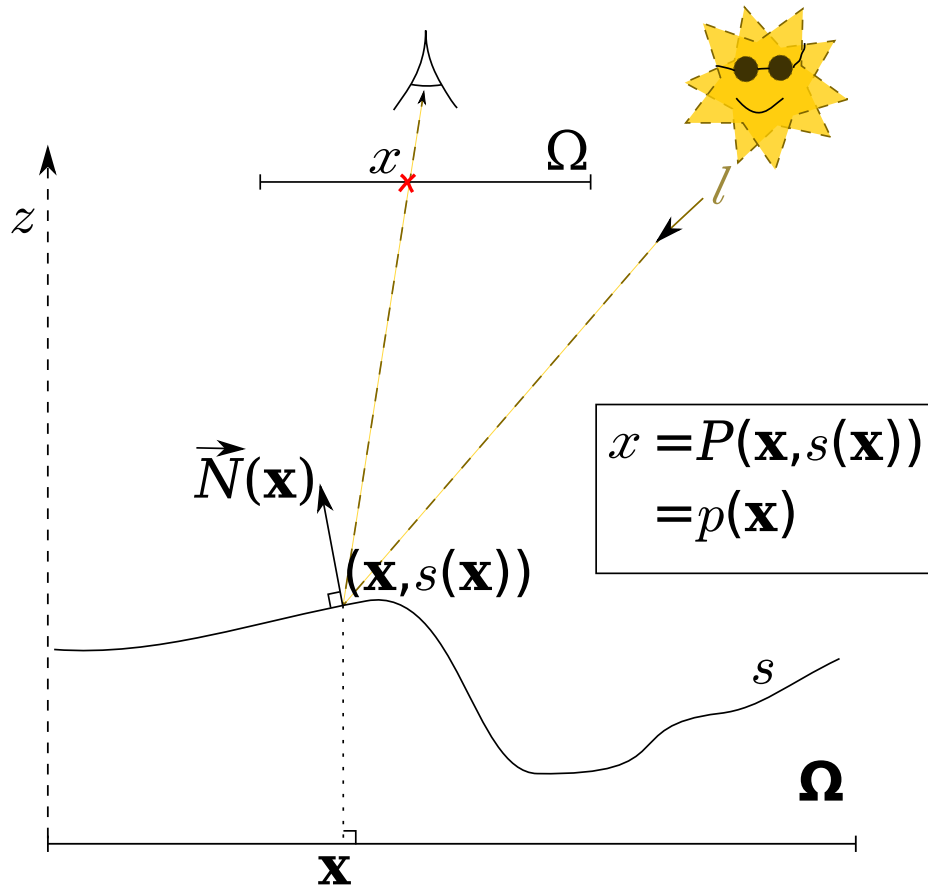


FIGURE 2.8 – Notations

Hypothèses sur la surface et modèle d'illumination

On utilise le modèle de Phong [Phong, 1975] pour modéliser les interactions entre la surface et la lumière. On admet les hypothèses suivantes sur la surface :

Hypothèse 1. On considère que la lumière est composée de lumière ambiante d'amplitude γ (composante présente uniformément sur toute la scène) et d'une lumière directionnelle (tous les rayons de lumière incidente sont parallèles et d'intensité égale).

Hypothèse 2. *On suppose que la surface est lambertienne avec un albédo variable $\alpha : \mathbb{R}^2 \rightarrow \mathbb{R}_*^+$ [Horn et Brooks, 1989].*

Hypothèse 3. *Afin d'éviter la présence d'ombres, on suppose que l'angle entre l et N est strictement inférieur à $\pi/2$ sur un ensemble $\mathcal{L} \subset \mathbb{R}^3$. L'ensemble des illuminations possibles $[l, \gamma]$ est noté $\mathcal{L} = \mathcal{L} \times \mathbb{R}^+$.*

Avec ces hypothèses, une scène S est décrite complètement par $S = (s, \alpha)$ et les conditions d'illumination sont complètement décrites par le vecteur $L = [l, \gamma] \in \mathcal{L}$. D'après l'hypothèse lambertienne, l'image u de la scène S sous les conditions d'illumination L peut s'écrire comme [Phong, 1975] :

$$u_{S,L}(x) = (\langle l, N(\mathbf{x}) \rangle + \gamma) \cdot \alpha(\mathbf{x}) \quad (2.10)$$

où $\langle \cdot, \cdot \rangle$ représente le produit scalaire canonique

Les résultats ci-après sont également valables pour un modèle d'image de la forme :

$$u_{S,L}(\mathbf{x}) = \varphi((\langle l, N(\mathbf{x}) \rangle + \gamma) \cdot \alpha(\mathbf{x})) \quad (2.11)$$

où $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction strictement monotone qui modélise un changement global de contraste. Cependant, pour simplifier les notations, on utilise uniquement le modèle décrit dans l'Équation 2.10.

Définition des lignes de niveau

Soit $u : \Omega \rightarrow \mathbb{R}$ une fonction quelconque. On suppose que u est définie sur tout Ω . Les lignes de niveau sont définies de la façon suivante :

Définition 1 (Lignes de niveau). *Les lignes de niveau de u sont les composantes connexes [Janisch, 1984] des iso-niveaux $\{\mathbf{x} \in \Omega, u(\mathbf{x}) = \alpha\}$*

Cette définition est généralement utilisée pour les fonctions C^1 pour lesquelles le gradient est toujours non nul. Pour de telles fonctions, on peut montrer que les lignes de niveau sont des courbes de Jordan soit fermées, soit résidant sur la frontière de Ω . Pour les autres fonctions, les lignes de niveau peuvent prendre la forme de n'importe quel objet connexe du plan tel que des points, des courbes, des fractales ... Le terme « ligne » est donc un abus de langage. Les auteurs de [Caselles et al., 1999a; Guichard et Morel, 2001] donnent une autre définition des lignes de niveau :

Définition 2 (lignes de niveau [Caselles et al., 1999a; Guichard et Morel, 2001]). *Soit u une fonction semi-continue supérieurement. Les lignes de niveau de u sont définies comme les frontières des composantes connexes des ensembles de niveau $\{x \in \Omega \leq \lambda\}$.*

Les lignes de niveau sont alors définies comme des courbes ; néanmoins, nous montrons plus loin que cette définition ne garantit pas l'invariance aux changements d'illumination, et ce, même sur des scènes relativement simples. Par la suite, nous utiliserons donc, sauf mention contraire, la première définition.

2.2.3 Invariance des lignes de niveau

Dans ce qui suit, nous caractérisons les scènes qui présentent des lignes de niveau invariantes à l'illumination. On suppose $s \in C^2$ et $\alpha \in C^1$. En conséquence, $u_{s,L}$ est C^1 pour tout $L \in \mathcal{L}$. Rappelons quelques définitions importantes en géométrie différentielle :

Définition 3 (Courbure gaussienne). *La courbure gaussienne d'une surface de \mathbb{R}^3 est définie comme le produit de ses deux courbures principales. Avec nos notations, la courbure gaussienne de s est définie comme $\det(\nabla^2 s)$.*

Définition 4 (Surface développable). *On appelle surface développable une surface C^2 dont la courbure gaussienne est nulle partout [Spivak, 1999].*

On peut exhiber des exemples simples de surfaces développables tels que des plans, des cylindres ou des cônes. Une surface développable possède les propriétés suivantes [Spivak, 1999] :

- Chaque point de la surface réside sur une ligne (la génératrice) appartenant à la surface ;
- Le plan tangent à la surface est le même en chaque point de la génératrice.

Munis des deux définitions précédentes, nous pouvons maintenant décrire l'ensemble des scènes générant des images dont les lignes de niveau sont invariantes :

Définition 5. *Soit Θ l'ensemble des scènes $S = (s, \alpha)$ telles que s est C^2 développable, α est C^1 , et varie uniquement dans la direction orthogonale aux génératrices de s . Sur les points où la surface de la scène est plane, α peut varier dans n'importe quelle direction.*

Nous allons montrer l'invariance des lignes de niveau sur cette ensemble étape par étape. Dans un premier temps, on se concentre sur les propriétés locales des images, (*i.e.* la direction du gradient). Plusieurs algorithmes invariants aux changements de contraste se fondent sur cette caractéristique. Par exemple, les auteurs de [Lisani et Morel, 2003] l'utilisent en détection de changements, les auteurs de [Cao et Bouthemy, 2006; Chen et al., 2000, 2008] l'utilisent pour calculer des mesures de dissimilarité entre deux images et les auteurs de [Droske et Rumpf, 2004] l'utilisent en recalage d'images médicales. Le théorème suivant caractérise les scènes pour lesquelles ces approches sont intégralement justifiées mathématiquement :

Théorème 1. *Soient $s \in C^2(\Omega)$ et $\alpha \in C^1(\Omega)$ Les propositions suivantes sont équivalentes :*

- Prop. 1 : $\forall (L_1, L_2) \in \mathcal{L} \times \mathcal{L}, \forall x \in \Omega, \nabla u_{s,L_1}(x) \parallel \nabla u_{s,L_2}(x)$
- Prop. 2 : $S \in \Theta$

Démonstration. On exprime d'abord le gradient de u en fonction de s, α et (l, γ) . Des considérations simples en géométrie donnent $N(\mathbf{x}) = \Psi(\nabla s(\mathbf{x})) = \frac{(-s_1(\mathbf{x}), -s_2(\mathbf{x}), 1)}{\sqrt{s_1^2(\mathbf{x}) + s_2^2(\mathbf{x}) + 1}}$, où s_1

et s_2 sont les dérivées partielles de s le long des deux axes du plan objet. Soit $\mathbf{x} = p^{-1}(x)$. En utilisant la règle de dérivation des fonctions composées et l'Équation 2.10, on obtient :

$$\nabla u_{S,L}(x) = \left(\underbrace{l}_{\in \mathcal{M}_{1,3}} \cdot \left[\underbrace{\Psi'(\nabla s(\mathbf{x}))}_{\in \mathcal{M}_{3,2}} \cdot \underbrace{\nabla^2 s(\mathbf{x})}_{\in \mathcal{M}_{2,2}} \underbrace{\alpha(\mathbf{x})}_{\in \mathbb{R}} + \underbrace{\Psi(\nabla s(\mathbf{x})) \cdot \nabla \alpha(\mathbf{x})}_{\in \mathcal{M}_{3,1}} + \underbrace{\gamma \nabla \alpha(\mathbf{x})}_{\in \mathcal{M}_{1,2}} \right] \cdot \underbrace{J(p^{-1})(x)}_{\in \mathcal{M}_{2,2}} \right)$$

Ce qui peut être réécrit comme :

$$\nabla u_{S,L}(x) = (l \cdot A(\mathbf{x}) + \gamma \nabla \alpha(\mathbf{x})) \cdot J(p^{-1})(x) \quad (2.12)$$

avec :

$$A(\mathbf{x}) = \left([M_1, M_2] \cdot \begin{bmatrix} s_{11} & s_{12} \\ s_{12} & s_{22} \end{bmatrix} + N \cdot [\alpha_1, \alpha_2] \right) (\mathbf{x}) \quad (2.13)$$

$M_1(\mathbf{x}), M_2(\mathbf{x}) \in \mathcal{M}_{3,1}$ sont les deux colonnes de $\Psi'(\nabla s(\mathbf{x}))\alpha(\mathbf{x})$ et $N(\mathbf{x}) = \Psi(\nabla s(\mathbf{x}))$.

Nous avons donné les fondements pour démontrer *Prop. 1* \Rightarrow *Prop. 2*. Soit S une scène telle que :

$$\forall (L_1, L_2) \in \mathcal{L} \times \mathcal{L}, \quad \forall x \in \Omega, \quad \nabla u_{S,L_1}(x) \parallel \nabla u_{S,L_2}(x) \quad (2.14)$$

Soit $D : \Omega \rightarrow \mathbb{S}^2$ la direction d'invariance de $\nabla u_{S,L}$. Sur les points x où $\forall L \in \mathcal{L}, \nabla u_{S,L}(x) = 0$, D peut correspondre à n'importe quel vecteur de \mathbb{R}^2 .

Comme l'Équation 2.14 est vraie pour tout $([l_1, \gamma_1], [l_2, \gamma_2]) \in \mathcal{L} \times \mathcal{L}$, elle est vraie en particulier pour $\gamma_1 = \gamma_2 = 0$. Il vient alors La Relation 2.15 :

$$\forall l \in \mathcal{L}, \quad \forall x \in \Omega, \quad l \cdot A(\mathbf{x}) \cdot J(p^{-1})(x) \parallel D(x) \quad (2.15)$$

p étant un difféomorphisme, $J(p^{-1})(x)$ est une matrice 2×2 inversible. Alors, la Propriété 2.15 ne dépend pas de $J(p^{-1})$ et est vraie seulement si la matrice $A(\mathbf{x})$ présente deux lignes parallèles. Ceci peut être réécrit de la façon suivante :

$$\begin{cases} (s_{11}M_1 + s_{12}M_2 + \alpha_1N)(\mathbf{x}) = a(\mathbf{x})C(\mathbf{x}) \\ (s_{12}M_1 + s_{22}M_2 + \alpha_2N)(\mathbf{x}) = b(\mathbf{x})C(\mathbf{x}) \end{cases} \quad (2.16)$$

où $(a, b) : \Omega \rightarrow \mathbb{R}^2$ et $C : \Omega \rightarrow \mathcal{M}_{3,1}$ sont des fonctions C^0 . Ceci est vrai seulement si \mathcal{L} a une certaine épaisseur (ouvert non-vide), mais la supposition inverse amène à considérer des cas « dégénérés » ayant peu d'intérêt. Un calcul élémentaire (mais fastidieux) mène à

$\det([M_1, M_2, N](\mathbf{x})) > 0$, de telle façon que les vecteurs $M_1(\mathbf{x}), M_2(\mathbf{x}), N(\mathbf{x})$ forment une base de \mathbb{R}^3 . Par conséquent, Le système d'Équations 2.16 implique :

$$\begin{pmatrix} s_{11} \\ s_{12} \\ \alpha_1 \end{pmatrix}(\mathbf{x}) \parallel \begin{pmatrix} s_{12} \\ s_{22} \\ \alpha_2 \end{pmatrix}(\mathbf{x}) \quad (2.17)$$

de sorte que :

$$\begin{pmatrix} s_{11} \\ s_{12} \end{pmatrix}(\mathbf{x}) \parallel \begin{pmatrix} s_{12} \\ s_{22} \end{pmatrix}(\mathbf{x}) \parallel \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}(\mathbf{x}) \quad (2.18)$$

Finalement, l'Équation 2.18 conduit à $\forall \mathbf{x} \in \Omega$, $\det(\nabla^2 s(\mathbf{x})) = 0$ ce qui implique que s est développable. On peut vérifier facilement que le vecteur propre de $\nabla^2 s(\mathbf{x})$ est $\begin{pmatrix} s_{12} \\ s_{22} \end{pmatrix}(\mathbf{x})$. Ce vecteur propre est dans la direction orthogonale à la génératrice de s passant par le point \mathbf{x} . L'Équation 2.18 indique alors que $\alpha(\mathbf{x})$ varie dans la direction orthogonale à cette génératrice. Pour les points tels que $\nabla^2 s(\mathbf{x}) = 0$, aucune condition n'est imposée à $\alpha(\mathbf{x})$.

Montrons maintenant *Prop. 2* \Rightarrow *Prop. 1*. Soit $S = (s, \alpha) \in \Theta$. En conséquence, $\forall \mathbf{x} \in \Omega$, $\det(\nabla^2 s(\mathbf{x})) = 0$. Il existe donc $(c_1, c_2, a, b) : \Omega \rightarrow \mathbb{R}^4$ tels que :

$$\begin{bmatrix} s_{11} & s_{12} \\ s_{12} & s_{22} \end{bmatrix} = \begin{bmatrix} c_1 a & c_2 a \\ c_1 b & c_2 b \end{bmatrix} \quad (2.19)$$

avec $c_1 b = c_2 a$. De plus, comme le gradient varie orthogonalement aux génératrices, il existe c_3 tel que :

$$[\alpha_1, \alpha_2] = c_3 [a, b] \quad (2.20)$$

À l'aide des Équations 2.19 et 2.20, on peut écrire la matrice A (voir Équation 2.13) sous la forme :

$$A = [aC, bC] \quad (2.21)$$

avec $C = M_1 c_1 + M_2 c_2 + N c_3$. On peut alors simplifier l'Équation 2.12 sous la forme :

$$\nabla u_{S,L}(x) = ((\langle l, C \rangle + \gamma c_3)[a, b])(\mathbf{x}) \cdot J(p^{-1})(x) \quad (2.22)$$

ce qui nous amène à $\nabla u_{S,L}(x) \parallel [a, b](\mathbf{x}) \cdot J(p^{-1})(x)$ pour tout $L \in \mathcal{L}$.

□

Le Théorème 1 implique que la direction du gradient est une caractéristique invariante aux changements d'illumination si et seulement si la scène appartient à Θ . Des résultats

analogues ont, par ailleurs, été également obtenus dans [Chen et al., 2000, 2008]. Les auteurs montrent que la direction du gradient est presque invariante aux changements d'illumination dans le sens où sa distribution par rapport à l'orientation de la lumière est concentrée le long d'un vecteur pour la plupart des scènes.

On se concentre maintenant sur les résultats concernant l'invariance des lignes de niveau. Le Théorème 1 amène aisément la proposition suivante :

Corollaire 1. *Les lignes de niveau sont invariantes aux changements d'illumination seulement si la scène appartient à Θ .*

Démonstration. Il s'agit d'une conséquence directe du Théorème 1 et du fait que deux fonctions C^1 u_1 et u_2 ont les mêmes lignes de niveau seulement si $\forall x \in \Omega$, $\nabla u_1(x) \parallel \nabla u_2(x)$. \square

Néanmoins, toutes les scènes appartenant à Θ n'ont pas des lignes de niveau invariantes aux changements d'illumination. Par exemple, la Figure 2.9 montre un cône avec un albédo constant. Si la lumière incidente est dirigée exactement dans la direction de l'axe du cône (gauche), la radiométrie du cône est uniforme et son image est, en conséquence, composée d'une seule ligne de niveau. Dans tous les autres cas, (image de droite), les lignes de niveau correspondent aux génératrices du cône.

Nous allons maintenant énoncer un résultat plus faible, mais qui complète néanmoins le Corollaire 1. Pour ce faire, il est nécessaire d'énoncer préalablement certains résultats liés à la topologie.

Lemme 1. *Soit $C : \Omega \rightarrow \mathbb{R}^{4*}$ une application sur laquelle aucune hypothèse de régularité n'est émise.*

On note $L \in \mathcal{L}$, L^\perp un hyperplan orthogonal à L .

Soit $\omega_L = \text{int}(\{x \in \Omega, C(x) \in L^\perp\})$.

Pour presque tout $L \in \mathcal{L}$ (vis-à-vis de la mesure de Lebesgue de \mathbb{R}^4) $\omega_L = \emptyset$.

Démonstration. On note $Y = \{L \in \mathcal{L}, \omega_L \neq \emptyset\}$ et \mathbb{Q} l'ensemble des nombres rationels. Soit $\Omega_{\mathbb{Q}} = \{\mathbb{Q}^2 \cap \Omega\}$. On a $\Omega_{\mathbb{Q}} \subset \mathbb{Q}^2$, $\Omega_{\mathbb{Q}}$ est alors un ensemble dénombrable.

Soit $a_i \in \Omega_{\mathbb{Q}}$ et $Y_i = \{L \in \mathcal{L}, a_i \in \omega_L\}$. Y_i est un sous ensemble d'un hyperplan de \mathbb{R}^4 . Dans le cas contraire, il existerait 4 éléments de Y_i , L_1, L_2, L_3, L_4 formant une base de \mathbb{R}^4 . Étant donné que $C(a_i) \perp L_j \forall j \in \{1, 2, 3, 4\}$, cela impliquerait $C(a_i) = 0$, ce qui contredirait $C(a_i) \in \mathbb{R}^{4*}$.

Alors $\mu_{\mathbb{R}^4}(Y_i) = 0$ et $\mu_{\mathbb{R}^4}(\bigcup_{a_i \in \Omega_{\mathbb{Q}}} Y_i) = 0$. De plus, $\bigcup_{a_i \in \Omega_{\mathbb{Q}}} Y_i = Y$ (puisque chaque ouvert non-vide ω_L contient un élément de \mathbb{Q}^2). Par conséquent, $\mu_{\mathbb{R}^4}(Y) = 0$. \square

Lemme 2. *Soit $\omega \in \Omega$ un ouvert. Soient u_1 et u_2 deux fonctions $C^1(\Omega)$ telles que : $\forall x \in \Omega$, $\nabla u_1(x) \parallel \nabla u_2(x)$, $\nabla u_1(x) \neq 0$ et $\nabla u_2(x) \neq 0$. Alors u_1 et u_2 ont les mêmes lignes de niveau sur ω .*

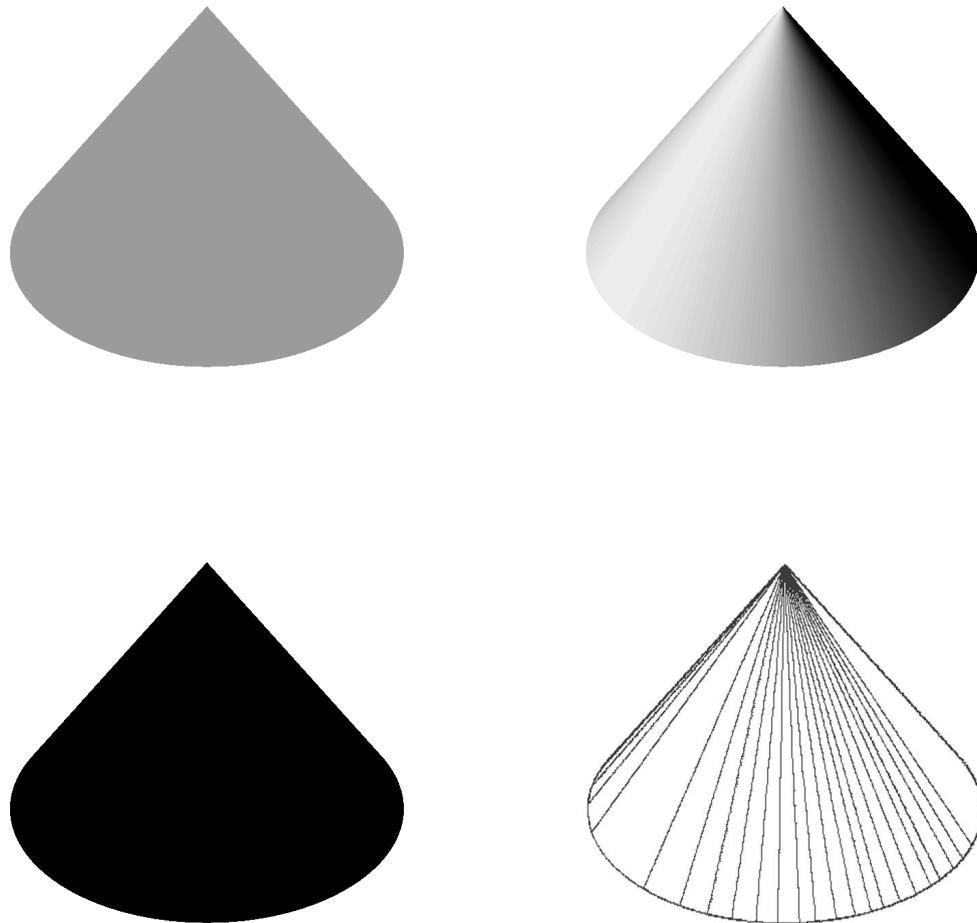


FIGURE 2.9 – Haut : images d'un cône sous deux lumières d'incidences différentes. Bas : quelques unes de leurs lignes de niveau.

Démonstration. ∇u_1 et ∇u_2 étant non-nulles sur ω , le théorème des fonctions implicites implique que les lignes de niveau de u_1 et u_2 sont des courbes C^1 . De plus, ces courbes peuvent être définies uniquement grâce à leurs normales respectives : $\frac{\nabla u_1}{|\nabla u_1|}$ et $\frac{\nabla u_2}{|\nabla u_2|}$ qui sont égales. \square

Lemme 3. Soient u_1 et u_2 deux fonctions $C^1(\Omega)$ telles que : $\forall x \in \Omega, \nabla u_1(x) \parallel \nabla u_2(x)$. Les propositions suivantes sont équivalentes :

- Prop. 1 : u_1 et u_2 n'ont pas les mêmes lignes de niveau ;
- Prop. 2 : Il existe un ouvert non-vide $\omega \subset \Omega$ tel que :

$$\forall x \in \omega, \quad \begin{cases} \nabla u_1(x) = 0 \\ \nabla u_2(x) \neq 0 \end{cases} \quad (2.23)$$

$$\text{ou } \forall x \in \omega, \quad \begin{cases} \nabla u_1(x) \neq 0 \\ \nabla u_2(x) = 0 \end{cases} . \quad (2.24)$$

Démonstration. La démonstration Prop. 2 \Rightarrow Prop. 1 est directe :

Étant donné que $\nabla u_1 = 0$ sur ω , u_1 est constante sur ω et est donc incluse dans une ligne de niveau de u_1 , alors que $\nabla u_2 \neq 0$ implique que $\exists(x, y) \in \omega^2$ tels que $u_2(x) \neq u_2(y)$. En conséquence, ω n'est pas inclus dans une ligne de niveau de u_2 .

On montre maintenant Prop. 1 \Rightarrow Prop. 2 par l'absurde. On suppose qu'il existe deux fonctions $C^1(\Omega)$: u_1 et u_2 telles que :

$$(H_1) \quad \forall x \in \Omega, \nabla u_1(x) \parallel \nabla u_2(x)$$

(H₂) il n'existe pas d'ensemble ouvert ω tel que :

$$\forall x \in \omega, \quad \begin{cases} \nabla u_1(x) = 0 \\ \nabla u_2(x) \neq 0 \end{cases} \quad (2.25)$$

$$\text{or } \forall x \in \omega, \quad \begin{cases} \nabla u_1(x) \neq 0 \\ \nabla u_2(x) = 0 \end{cases} \quad (2.26)$$

(H₃) Les ligne de niveau de u_1 sont différentes des lignes de niveau de u_2

Soient :

$$\Omega_{1,0} = \{x \in \Omega, \nabla u_1(x) = 0\}$$

$$\Omega_{2,0} = \{x \in \Omega, \nabla u_2(x) = 0\}$$

1. Dans un premier temps, on remarque que $\mathring{\Omega}_{1,0} = \mathring{\Omega}_{2,0}$.

En effet, supposons :

$$\begin{cases} \mathring{\Omega}_{1,0} \subset \Omega_{2,0} \\ \mathring{\Omega}_{2,0} \subset \Omega_{1,0} \end{cases} \quad (2.27)$$

$\mathring{\Omega}_{1,0}$ étant un ouvert, on a $\mathring{\Omega}_{1,0} \subset \mathring{\Omega}_{2,0}$. En appliquant le même raisonnement, on obtient $\mathring{\Omega}_{2,0} \subset \mathring{\Omega}_{1,0}$, et finalement $\mathring{\Omega}_{1,0} = \mathring{\Omega}_{2,0}$.

Supposons maintenant que le Système d'Équations 2.27 est faux :

$$\mathring{\Omega}_{1,0} \not\subset \mathring{\Omega}_{2,0} \text{ or } \mathring{\Omega}_{2,0} \not\subset \mathring{\Omega}_{1,0} \quad (2.28)$$

En conséquence, quitte à inverser les indices, il existe $x \in \mathring{\Omega}_{1,0} \setminus \mathring{\Omega}_{2,0}$ tel que $\nabla u_1(x) = 0$ et $\nabla u_2(x) \neq 0$. Par conséquent, d'après la continuité de ∇u_2 il existe un ouvert $\omega_x \subseteq \mathring{\Omega}_{1,0}$ contenant x tel que $\nabla u_2 \neq 0$ sur ω_x . Par ailleurs $\omega_x \subseteq \mathring{\Omega}_1^0$ implique que $\nabla u_1(x) = 0$ sur ω_x . Ce qui contredit (H_2) .

2. On note $\tilde{\Omega}^+ = \Omega \setminus \overline{\mathring{\Omega}_{1,0}}$. Sur $\tilde{\Omega}^+$, les lignes de niveau de u_1 sont différentes de celles de u_2 .

Encore une fois, on utilise une démonstration par l'absurde. On suppose que les lignes de niveau de u_1 sont les mêmes que les lignes de niveau de u_2 sur $\tilde{\Omega}^+$. D'après le paragraphe précédent, ∇u_1 et ∇u_2 sont nulles sur $\mathring{\Omega}_{1,0} = \mathring{\Omega}_{2,0}$. Par continuité, ceci est également valable sur $\overline{\mathring{\Omega}_{1,0}}$. Par conséquent u_1 et u_2 sont toutes deux constantes sur chaque composante connexe U_i de $\overline{\mathring{\Omega}_{1,0}}$. Chaque U_i est alors une ligne de niveau de u_1 et u_2 sur $\overline{\mathring{\Omega}_{1,0}}$. Considérons maintenant une ligne de niveau A de u_1 dans $\tilde{\Omega}^+$. On peut considérer deux cas :

- $\forall i, U_i \cap A = \emptyset$. Dans ce cas, A est clairement une ligne de niveau de u_1 dans Ω . Le même raisonnement pouvant être appliqué à u_2 , A est également une ligne de niveau de u_2 dans Ω .
- $\exists i, U_i \cap A \neq \emptyset$. A et U_i sont tous deux des ensembles connexes, alors $A \cup U_i$ est un ensemble connexe. u_1 étant continue, il vient que u_1 est constante sur $A \cup U_i$. De même, u_2 est constante sur $A \cup U_i$.

Alors, les lignes de niveau de u_1 et u_2 sont les mêmes sur Ω ce qui est contradictoire avec nos hypothèses.

3. Soit $\omega_1 \subset \Omega$ un ouvert tel que $\nabla u_1 = 0$ sur ω_1 . Nécessairement, $\omega_1 \subset \mathring{\Omega}_1$ et donc $\omega_1 \cap \tilde{\Omega}^+ = \emptyset$. Alors, il n'existe pas d'ouvert dans $\tilde{\Omega}^+$ sur lequel $\nabla u_1 = 0$. Ce résultat est également valable avec u_2 . Soit $\Omega_1^+ = \{x, \nabla u_1(x) \neq 0\}$ (respectivement $\Omega_2^+ = \{x, \nabla u_2(x) \neq 0\}$). Ω_1^+ et Ω_2^+ sont des ouverts denses dans $\tilde{\Omega}_+^+$. Alors, par le théorème de Baire, il vient que, $\Omega_1^+ \cap \Omega_2^+$ est un ouvert dense dans $\tilde{\Omega}^+$.
4. D'après le Lemme 2, on sait que les lignes de niveau de u_1 et u_2 sont identiques sur $\overline{\Omega_1^+ \cap \Omega_2^+}$. La continuité de u_1 et u_2 assure qu'elles sont également identiques sur $\Omega_1^+ \cap \Omega_2^+ = \tilde{\Omega}^+$. Ce résultat est en contradiction avec la conclusion du 2.

□

Nous avons maintenant en main tous les éléments nécessaires à la démonstration du Théorème 2.

Théorème 2. Soit $S \in \Theta$. Pour presque toutes les paires de conditions d'illumination $(L_1, L_2) \in \mathcal{L} \times \mathcal{L}$ (dans le sens de la mesure de Lebesgue de $\mathbb{R}^4 \times \mathbb{R}^4$) les lignes de niveau de u_{S,L_1} sont les mêmes que celles de u_{S,L_2}

Démonstration. On suppose $S \in \Theta$. Ceci implique (cf. démonstration du Théorème 1)) que $\nabla u_{S,L} = \langle L, C \rangle \cdot [a, b]$ avec $C : \Omega \rightarrow \mathbb{R}^{4*}$, et $[a, b] : \Omega \rightarrow \mathbb{R}^2$, des fonctions $C^0(\Omega)$. On obtient alors $\forall (L_1, L_2) \in \mathcal{L} \times \mathcal{L}, \forall x \in \Omega, \nabla u_{S,L_1}(x) \parallel \nabla u_{S,L_2}(x)$.

Soit $\Omega^+ = \{x \in \Omega, [a, b](x) \neq 0\}$. Cet ensemble est un ouvert car a et b sont $C^0(\Omega)$. Soit

$$\omega_L = \text{int}(\{x \in \Omega^+, \nabla u_{S,L}(x) = 0\}) \quad (2.29)$$

Cet ensemble est également caractérisé par :

$$\omega_L = \text{int}(\{x \in \Omega^+, C(x) \in L^\perp\}) \quad (2.30)$$

D'après le lemme 1, pour presque tout $L_1 \in \mathcal{L}, \omega_{L_1} = \emptyset$. En conséquence, pour presque tout $L_1 \in \mathcal{L}, \forall \omega$ ouvert de $\Omega^+, \exists x \in \omega$ tel que $\nabla u_{S,L_1}(x) \neq 0$. La continuité de $\nabla u_{S,L_1}$ induit l'existence d'un ouvert $\omega_x \subset \Omega^+$ contenant x et sur lequel $\nabla u_{S,L_1} \neq 0$. De plus, pour presque tout $L_2 \in \mathcal{L}, \omega_{L_2} = \emptyset$, de sorte qu'il existe $x' \in \omega_x$ tel que $\nabla u_2(x') \neq 0$.

Finalement, pour presque tout $(L_1, L_2) \in \mathcal{L} \times \mathcal{L}, \forall \omega \subseteq \Omega^+, \exists x' \in \omega$ tel que :

$$\begin{cases} \nabla u_1(x') \neq 0 \\ \nabla u_2(x') \neq 0 \end{cases}$$

De plus, $\forall x \in \Omega \setminus \Omega^+, \nabla u_1(x) = \nabla u_2(x) = 0$.

On remarque, par ailleurs, que la contraposée du Lemme 3 est :

$$u_1 \text{ et } u_2 \text{ ont les mêmes lignes de niveau} \Leftrightarrow \quad (2.31)$$

$$\forall \omega \text{ ouvert de } \Omega^+, \exists x \in \omega, \quad (2.32)$$

$$\begin{cases} \nabla u_1(x) \neq 0 \\ \nabla u_2(x) \neq 0 \end{cases} \text{ ou } \begin{cases} \nabla u_1(x) = 0 \\ \nabla u_2(x) = 0 \end{cases} \quad (2.33)$$

On en déduit que pour presque tout $(L_1, L_2) \in \mathcal{L} \times \mathcal{L}, u_{S,L_1}$ et u_{S,L_2} ont les mêmes lignes de niveau. \square

Ce théorème et le Corollaire 1 montrent qu'il existe « presque » une équivalence entre ces deux propositions :

- La scène a des lignes de niveau invariantes ;
- La scène appartient à Θ .

Malheureusement, l'espace Θ contient trop peu de surfaces pour modéliser toutes les scènes réalistes¹. Ceci nous amène à analyser l'invariance des lignes de niveau, lorsque S est une fonction C^2 développable par morceaux et que son albédo varie orthogonalement aux génératrices sur chaque morceau. On commence par donner une définition précise de cet espace :

Définition 6. Ξ est l'espace des scènes $\bar{S} = (\bar{s}, \bar{\alpha})$ tel qu'il existe un ensemble fini $\{\omega_i\}_{i \in I}$ et une scène $S = (s, \alpha)$ qui satisfont :

- $\forall i \in I, \omega_i \subset \Omega$ est un ouvert connexe de mesure non-nulle.
- $\forall (i, j), \omega_i \cap \omega_j = \emptyset$.
- $\cup_{i \in I} \bar{\omega}_i = \Omega$.
- $\forall i, S|_{\omega_i} \in \Theta$ (la restriction de S à ω_i appartient à Θ).
- Finalement, on suppose que $S|_{\omega_i}$ (de même que $N|_{\omega_i}$) admet une limite à la frontière de ω_i . Ceci permet de définir $\bar{S} = (\bar{s}, \bar{\alpha})$ partout. Par exemple, on pourrait la définir comme suit :

$$\begin{cases} (\bar{s}, \bar{\alpha})(\mathbf{x}) = (s, \alpha)(\mathbf{x}) & \text{si } \mathbf{x} \in \cup_{i \in I} \omega_i \\ \left\{ \begin{array}{l} \bar{s}(\mathbf{x}) = \lim_{r \rightarrow 0} \sup_{\mathbf{y} \in B(\mathbf{x}, r)} s(\mathbf{y}) \\ \bar{\alpha}(\mathbf{x}) = \lim_{r \rightarrow 0} \sup_{\mathbf{y} \in B(\mathbf{x}, r)} \alpha(\mathbf{y}) \end{array} \right. & \text{sinon} \end{cases}$$

Pour les images générées par des scènes de Ξ , les lignes de niveau ont des propriétés d'invariance plus faibles que précédemment. La Figure 2.10 illustre ce problème. Sur la partie gauche, la plupart des lignes de niveau du toit en forme de cylindre sont des segments du toit. Selon l'orientation de la lumière, un ou plusieurs de ces segments fusionnent avec le « mur » du bâtiment. En conséquence, les lignes de niveau ne sont pas invariantes. La partie droite de la Figure 2.10 montre l'image d'un bâtiment avec un toit simple à deux versants plats. Si la direction de la lumière appartient au plan bisecteur de ces deux versants, alors, ces derniers auront la même radiométrie. Pour la plupart des illuminations, le toit sera alors constitué de deux lignes de niveau (jaune et rouge), alors que pour un ensemble de mesure nulle d'illuminations, il sera constitué d'une seule ligne de niveau. Dans ce qui suit, on traduit ces conclusions de manière plus formelle.

Soit ω un sous-ensemble de Ω . On pose les notations suivantes :

- $\kappa(\mathbf{x}, L)$ est la ligne de niveau de $u_{S,L}$ telle que $x = p(\mathbf{x}) \in \kappa(\mathbf{x}, L)$;
- $\kappa_{\omega}(\mathbf{x}, L)$ est la ligne de niveau de $u_{S,L|\omega}$ telle que $x \in \kappa_{\omega}(\mathbf{x}, L)$.

Proposition 1. Soit S un élément de Ξ . Soient ω_i et ω_j deux morceaux adjacents. Deux lignes de niveau adjacentes de $u_{S,L|\omega_i}$ et $u_{S,L|\omega_j}$ ne fusionnent pour presque aucune L .

1. Une exception notable est celle des documents courbés. Une feuille de papier courbée est une surface développable. On peut donc imaginer utiliser le Théorème 2 afin de redresser les photographies de documents courbés (livres), et ce indépendamment des conditions d'éclairage. Cette idée a déjà été exploitée par quelques auteurs [Courteille et al., 2007; Tan et al., 2006; Wada et al., 1997]

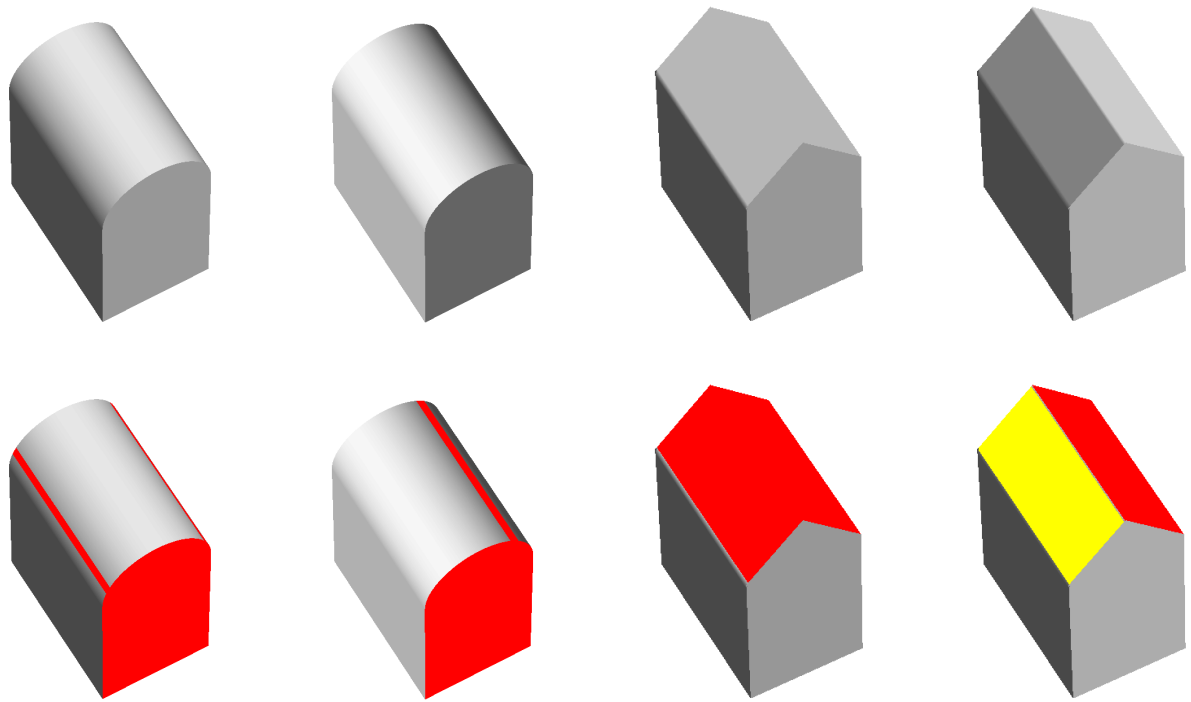


FIGURE 2.10 – Exemples de non-invariance des lignes de niveau dans le cas C^2 par morceaux. Les éléments colorés représentent des lignes de niveau isolées.

Démonstration. Soit $z \in \bar{\kappa}_{\omega_i}(\mathbf{x}_i, L) \cap \bar{\kappa}_{\omega_j}(\mathbf{x}_j, L)$ et $\mathbf{z} = p^{-1}(z)$.

On a :

$$\forall x \in \kappa_{\omega_i}(\mathbf{x}_i, L), u(x_i) = u(x)$$

Et, en particulier :

$$\begin{aligned} u(x_i) &= \lim_{\substack{x \rightarrow z \\ x \in \omega_i}} u(x) \\ u(x_i) &= \lim_{\substack{x \rightarrow z \\ x \in \omega_i}} ((\langle l, N \rangle + \gamma) \alpha)(\mathbf{x}) \end{aligned}$$

$z \in \bar{\kappa}_{\omega_i}(\mathbf{x}_i, L) \cap \bar{\kappa}_{\omega_j}(\mathbf{x}_j, L)$ donc, $z \in \bar{\omega}_i \cap \bar{\omega}_j$. Comme p est un difféomorphisme, $\mathbf{z} \in \bar{\omega}_i \cap \bar{\omega}_j$ et $\lim_{\substack{x \rightarrow z \\ x \in \omega_i}} \mathbf{x} = \mathbf{z}$. En conséquence, d'après les hypothèses précédentes, on a :

$$u(x_i) = \langle l, \lim_{\substack{x \rightarrow z \\ x \in \omega_i}} (\alpha N)(\mathbf{x}) \rangle + \gamma \lim_{\substack{x \rightarrow z \\ x \in \omega_i}} \alpha(\mathbf{x}) \quad (2.34)$$

$$= \langle l, \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_i}} (\alpha N)(\mathbf{x}) \rangle + \gamma \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_i}} \alpha(\mathbf{x}) \quad (2.35)$$

$$(2.36)$$

De même, on a :

$$u(x_j) = \langle l, \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_j}} (\alpha N)(\mathbf{x}) \rangle + \gamma \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_j}} \alpha(\mathbf{x}) \quad (2.37)$$

Alors $u(x_i) = u(x_j)$ si et seulement si :

$$\underbrace{\langle l, \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_i}} (\alpha N)(\mathbf{x}) - \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_j}} (\alpha N)(\mathbf{x}) \rangle}_{T_1} + \gamma \left(\lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_i}} \alpha(\mathbf{x}) - \lim_{\substack{\mathbf{x} \rightarrow \mathbf{z} \\ x \in \omega_j}} \alpha(\mathbf{x}) \right) = 0 \quad (2.38)$$

On considère alors deux cas :

- si $\lim_{\substack{\mathbf{x} \rightarrow \mathbf{y} \\ x \in \omega_i}} (N, \alpha)(\mathbf{x}) \neq \lim_{\substack{\mathbf{x} \rightarrow \mathbf{y} \\ x \in \omega_j}} (N, \alpha)(\mathbf{x})$, alors l'Équation 2.38 est vérifiée si et seulement si L réside dans un hyperplan particulier de \mathbb{R}^4 . Un tel espace est de mesure de Lebesgue nulle.
- si $\lim_{\substack{\mathbf{x} \rightarrow \mathbf{y} \\ x \in \omega_i}} (N, \alpha)(\mathbf{x}) = \lim_{\substack{\mathbf{x} \rightarrow \mathbf{y} \\ x \in \omega_j}} (N, \alpha)(\mathbf{x})$, alors on se reporte au cas C^2 sur Ω entier, en conséquence : $u_{S,L}(x_i)$ et $u_{S,L}(x_j)$ forment une unique ligne de niveau qui est invariante pour presque toutes les conditions d'illumination (vis-à-vis de la mesure de Lebesgue).

□

2.2.4 Algorithme invariant aux changements d'illumination pour la détection de changements

On s'intéresse maintenant au problème de détection de changements en essayant de mettre à profit les résultats précédents. D'après des résultats récents, il est impossible (avec l'hypothèse d'une scène lambertienne) de décider si deux images représentent la même scène sous deux conditions d'illumination ou bien deux scènes différentes [Chen et al., 2000]. En pratique, aucun algorithme n'est à notre connaissance capable de traiter un grand nombre d'images de télédétection avec un nombre raisonnable de fausses alarmes. Les principaux facteurs générateurs de fausses alarmes ont été évoqués dans le Chapitre 1 (une illustration des problèmes de parallaxe est donnée dans la Figure 2.11). À cela s'ajoutent les difficultés dues à la quantité importante de détails haute-résolution ce qui a poussé certains auteurs à se concentrer uniquement sur les changements importants du paysage urbain.

Les opérateurs humains font appel à des interprétations sémantiques, ce qui pousse les auteurs à introduire des hypothèses *a priori* sur la géométrie des scènes étudiées. Par exemple, certains auteurs [Fournier et al., 2008; Li et al., 2006; Lillestrand, 1972] tentent de détecter des objets dont les bords sont des lignes ou des polygones (cf. également Chapitre 4), objets qui ont de grandes probabilités de correspondre à des bâtiments. D'autres auteurs supposent que l'élévation de la scène est constante par morceaux [Watanabe et al., 1998]. Dans ce chapitre, nous supposerons que les scènes appartiennent à Ξ . Nous proposons un algorithme simple qui égalise le contraste de deux images. Après ce pré-traitement, une simple différence pixel à pixel fournit des résultats encourageants, que ce soit sur des images issues de scènes en OpenGL ou des images réelles (Quickbird).

Hypothèses

On rappelle et justifie les hypothèses faites sur les scènes :

Hypothèse 1 les surfaces ont une réflectance lambertienne ;

Hypothèse 2 l'illumination a deux composantes : une composante ambiante et une composante directionnelle ;

Hypothèse 2 La scène appartient à Ξ ;

Hypothèse 4 Les deux images sont parfaitement recalées ;

Hypothèse 5 On néglige le problème des ombres.

- L'Hypothèse 1 est à peu près cohérente, dans le sens où la plupart des surfaces urbaines sont principalement mates (asphalte, béton). Cette hypothèse n'est pas valable dans le cas de surfaces mouillées ou de vitres.

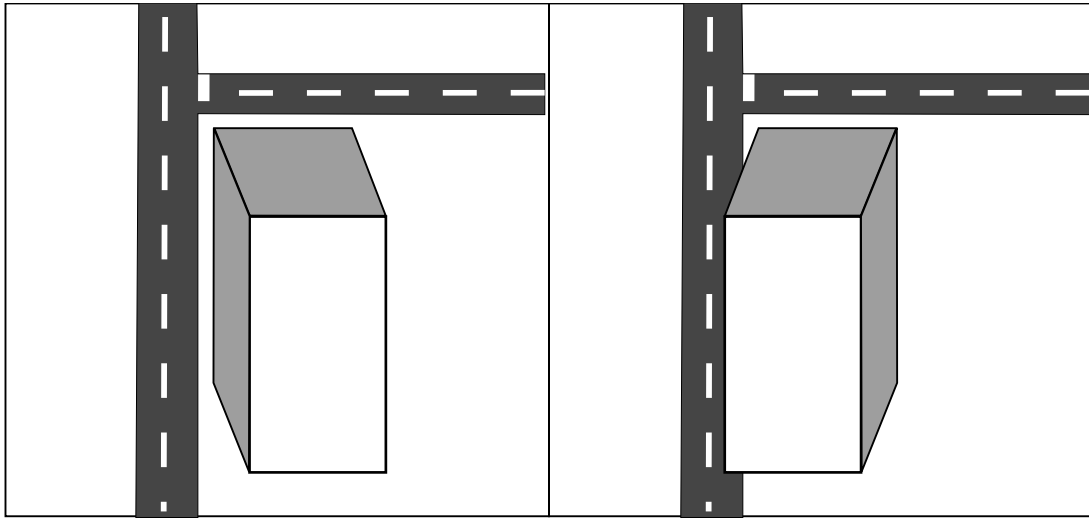


FIGURE 2.11 – Illustration des erreurs de parallaxe : deux scènes identiques sont représentées avec des géométries différentes.

- L’Hypothèse 2 est naturelle, car on peut considérer qu’il n’y a qu’une seule source ponctuelle de lumière à l’infini (on néglige l’ouverture angulaire du soleil). La lumière ambiante est due à la diffusion de la lumière solaire sur les divers objets de la scène.
- L’Hypothèse 4 s’appuie sur la structure géométrique des scènes urbaines. Elle est plus dure à justifier puisqu’elle dépend des normes locales d’architecture dans chaque partie du monde. Toutefois, de manière générale, on remarque que les scènes sont développables par morceaux. Les toits en forme de dôme sont relativement rares. La Figure 2.12 est issue de [Lafarge et al., 2008] où les auteurs utilisent un dictionnaire correspondant à des sections de bâtiments de style européen. On remarque qu’ils satisfont tous cette hypothèse.
- L’Hypothèse 4 est une hypothèse forte. En pratique, il est assez rare que deux images aériennes ou satellitaires soient prises depuis exactement la même position. En outre, des erreurs de parallaxe sont introduites (voir Figure 2.11). Quelques références adressant ce problème sont données dans le Chapitre 1. Certaines fausses alarmes détectées par notre méthode sont expliquées par cette hypothèse qui n’est pas toujours vérifiée.
- L’hypothèse sur les ombres est principalement là pour simplifier le modèle (lumière directionnelle présente partout dans la scène). Certaines publications [Watanabe et al., 1998] proposent des méthodes pour détecter et corriger l’impact des ombres des bâtiments

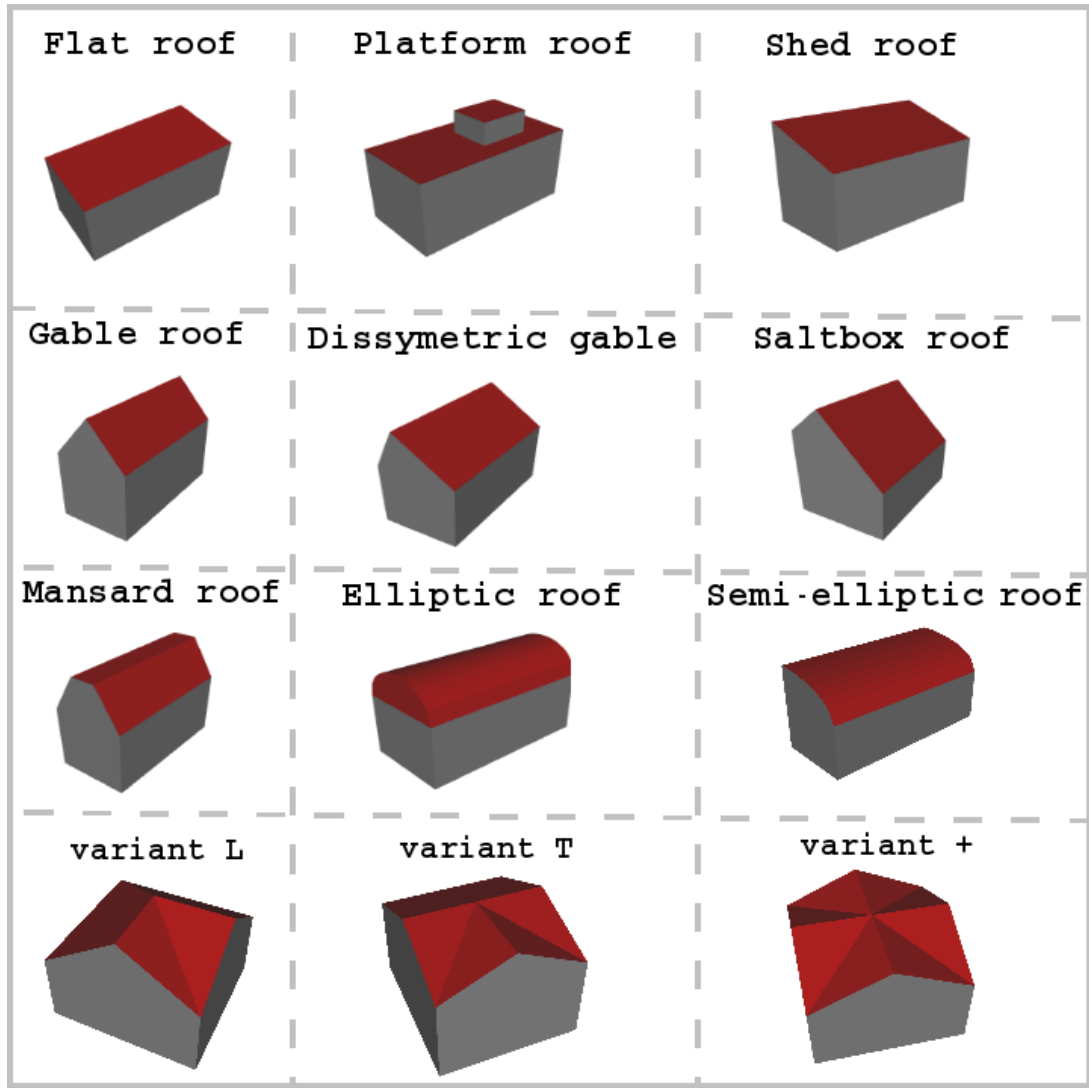


FIGURE 2.12 – Dictionnaire de toits utilisés dans [Lafarge et al., 2008] pour modéliser des villes d’architecture européenne. Elles sont toutes constituées de morceaux de surfaces développables. Images fournies gracieusement par F. Lafarge.

2.2.5 Un algorithme d'égalisation de contraste

Sous les hypothèses précédemment évoquées, on a vu que les lignes de niveau devaient être « presque » invariantes aux changements d'illumination. Nous proposons une méthode de normalisation de contraste et une procédure de détection de changements qui tirent parti de ce résultat. Soient u_1 et u_2 deux images parfaitement recalées, prises sous des conditions d'illuminations différentes, respectivement L_1 et L_2 aux temps respectifs t_1 et t_2 . Soit S_1 la scène 3D au temps t_1 . Sous ces notations, on peut écrire :

$$\begin{cases} u_1 = u_{S_1, L_1} \\ u_2 = u_{S_1, L_2} + c_{1,2} \end{cases} \quad (2.39)$$

où $c_{1,2}$ représente les changements de l'image u_1 à l'image u_2 . Dans cette équation, u_{S_1, L_2} et $c_{1,2}$ sont inconnus. Afin de les déterminer, on introduit des *a priori*. D'après les considérations précédentes, il est naturel de considérer que u_{S_1, L_2} appartient à l'espace des images qui ont les mêmes lignes de niveau que u_1 . On appelle cet espace Ξ_{u_1} . On introduit également un *a priori* sur les changements $J(c)$. Dans la plupart des applications, les changements sont rares. Ici, la norme L_1 étant reconnue comme favorisant les structures parcimonieuses, on pose alors $J(c) = \|c\|_1$. Pour déterminer $c_{1,2}$, le problème à résoudre est le suivant :

$$\inf_{u \in \Xi_{u_1}} (\|u_2 - u\|_1) \quad (2.40)$$

en posant $c_{1,2} = u_2 - \bar{u}$ où \bar{u} est la solution de l'Équation 2.40. Cette dernière peut donc être formulée comme : « trouver l'image u la plus proche de u_2 qui possède les mêmes lignes de niveau que u_1 ». Pour résoudre le Problème 2.40, il faut discrétiser Ξ_{u_1} . On propose la stratégie suivante :

1. On pose $u_Q = \lfloor \frac{u_1}{\Delta} \rfloor \Delta$ (quantification uniforme)
2. Pour chaque niveau $k\Delta$ ($k \in \mathbb{Z}$), on sépare les composantes connexes $\Omega_{k,j}$ de l'ensemble $\Omega_k = \{x \in \mathbb{R}^n, u_Q(x) = k\Delta\}$. Dans les simulations, les composantes connexes se fondent sur le système de voisinage 8-connexe.

On définit Ξ_{u_1} l'ensemble des images constantes sur la composante $\Omega_{k,j}$. Avec cette définition, la solution du Problème 2.40 est sous sa forme analytique :

$$\bar{u}|_{\omega_{k,j}} = \text{mediane}(u_2|_{\omega_{k,j}}) \quad (2.41)$$

Ce type d'algorithme a déjà été utilisé et analysé dans le cadre d'une autre application dans [Caselles et al., 2002]. C'est un algorithme très rapide (moins de 0.4 secondes sur une image 1000×1000 sur un Processeur Intel Xeon à 1.86 GHz).

Un détail important est que cet algorithme est asymétrique. On peut donc résoudre le problème suivant :

$$\inf_{u \in \mathcal{X}_{u_2}} (\|u_1 - u\|_1) \quad (2.42)$$

afin de déterminer une seconde image de changements $c_{2,1} = u_1 - \bar{u}$. Cette asymétrie permet notamment de déterminer à quelle scène appartient un objet détecté (cf. Figure 2.15)

2.2.6 Résultats

Dans cette partie, nous comparons l'approche par lignes de niveau avec deux algorithmes classiques : la projection monotone [Moisan, 2005] et la FLST (« Fast LevelSet Transform ») [Monasse et Guichard, 2000; Ballester et al., 2000]. Le premier est un outil fondamental en traitement d'images alors que le second est fondé sur des principes assez similaires au nôtre. Nous commençons par les décrire brièvement, puis nous les comparons sur des images de synthèse ainsi que sur des images réelles.

Description des approches alternatives

Projection monotone La projection monotone, décrite dans [Moisan, 2005], est similaire à l'égalisation globale de contraste. On la choisit pour les comparaisons parce qu'elle est plus générale qu'une égalisation de contraste linéaire [Radke et al., 2005; Wiemker, 1997]. Comme dans la partie 2.2.5, on suppose que les deux images peuvent être écrites comme :

$$\begin{cases} u_1 = u_{S_1, L_1} \\ u_2 = u_{S_1, L_2} + c_{1,2} \end{cases} \quad (2.43)$$

La projection monotone s'appuie sur l'hypothèse suivante : deux images de la même scène prises sous des conditions d'illuminations différentes diffèrent uniquement par un changement de contraste global et monotone. Ceci implique que $u_{S_1, L_2} = g \circ u_{S_1, L_1}$ ou $g : \mathbb{R} \rightarrow \mathbb{R}$ est une fonction croissante. Pour déterminer les changements, on cherche alors la fonction \bar{g} qui minimise l'énergie :

$$\bar{g} = \arg \min_{g \text{ croissante}} (\|g \circ u_1 - u_2\|_2^2) \quad (2.44)$$

Finalement, on pose $c_{1,2} = \bar{g} \circ u_1 - u_2$. Le Problème 2.44 peut être résolu en $O(n)$ opérations où n est le nombre de pixels [Moisan, 2005].

Comparaison des arbres d'ensembles de niveau Les auteurs de [Ballester et al., 2000; Monasse et Guichard, 2000] proposent un algorithme invariant aux changements de contraste pour la détection de changements. Cet algorithme est proche du notre étant donné qu'il fait appel aux lignes de niveau de façon importante.

Le principe de l'algorithme est le suivant : premièrement, chaque image est décomposée en un arbre de « formes » (les composantes connexes des ensembles de niveau). La Figure 2.13 représente la FLST de deux fonctions différentes. Les deux arbres sont ensuite comparés. Une forme dans l'arbre de u_1 sera considérée comme correspondant à l'arbre de u_2 s'il existe une forme dans cet arbre qui a approximativement les mêmes moments (position, surface, ...). Sur la Figure 2.13, les formes sans correspondance sont celles possédant l'étiquette « 2 ». Les images de changements $c_{1,2}$ et $c_{2,1}$ sont alors recomposées à partir des formes sans correspondance. Nous invitons le lecteur à se référer à [Ballester et al., 2000; Monasse et Guichard, 2000] pour plus de détails.

Images de synthèse

Afin d'illustrer les résultats présentés dans ce chapitre, nous avons développé une application très simple, permettant de générer des scènes en OpenGL correspondant à notre modèle. Cette application permet, en outre, de visualiser facilement ces scènes sous des conditions d'illumination différentes. Les images en haut de la Figure 2.15 montrent des prises de vues de deux scènes « urbaines ». Dans cet exemple, quelques bâtiments ont disparus, d'autres sont apparus, certains ont été déplacés et enfin, quelques uns ont été modifiés.

Dans cet exemple jouet, la scène n'appartient pas à Ξ (en effet, un des bâtiments est un dôme), mais toutes les autres hypothèses de notre modèle sont satisfaites. Ces résultats montrent clairement les avantages de notre méthode :

- Le résultat de l'algorithme de projection monotone est clairement peu satisfaisant. Un changement global de contraste ne peut pas reproduire les inversions locales de radiométrie dues aux changements de direction d'illumination. Ceci explique la grande quantité de fausses alarmes sur les toits en forme de dents de scie et les toits simples à deux versants.
- Les mauvais résultats de la FLST sont plus subtils à expliquer. Premièrement, dans nos simulations, nous comparons les formes uniquement par le biais de leurs barycentres et leur surface. Ces caractéristiques sont trop naïves pour donner des résultats satisfaisants. Dans [Monasse et Guichard, 2000], les auteurs suggèrent l'utilisation de moments d'ordre supérieur. Néanmoins, même munis d'une bonne mesure de comparaison, la méthode devrait échouer parce que la FLST n'est pas invariante aux changements d'illumination. Par exemple, la Figure 2.13 montre deux images d'un toit simple à deux versants sous deux illumination différentes, pour lesquelles les arbres ne sont pas comparables. En conséquence, certaines formes vont être marquées comme « changées » par l'algorithme et générer un nombre important de fausses alarmes.
- En revanche, le résultat de notre algorithme est très satisfaisant. On remarque qu'il échoue au niveau du dôme (fausses alarmes dues à la courbure gaussienne non-nulle). Ceci était prévisible étant donné que le dôme n'est pas une surface développable. On

remarque qu'entre les deux prises de vue, le bâtiment au toit en forme de cylindre a bougé le long de son axe, seules les parties qui ne se recouvrent pas ont été détectées, les autres parties générant des faux négatifs.

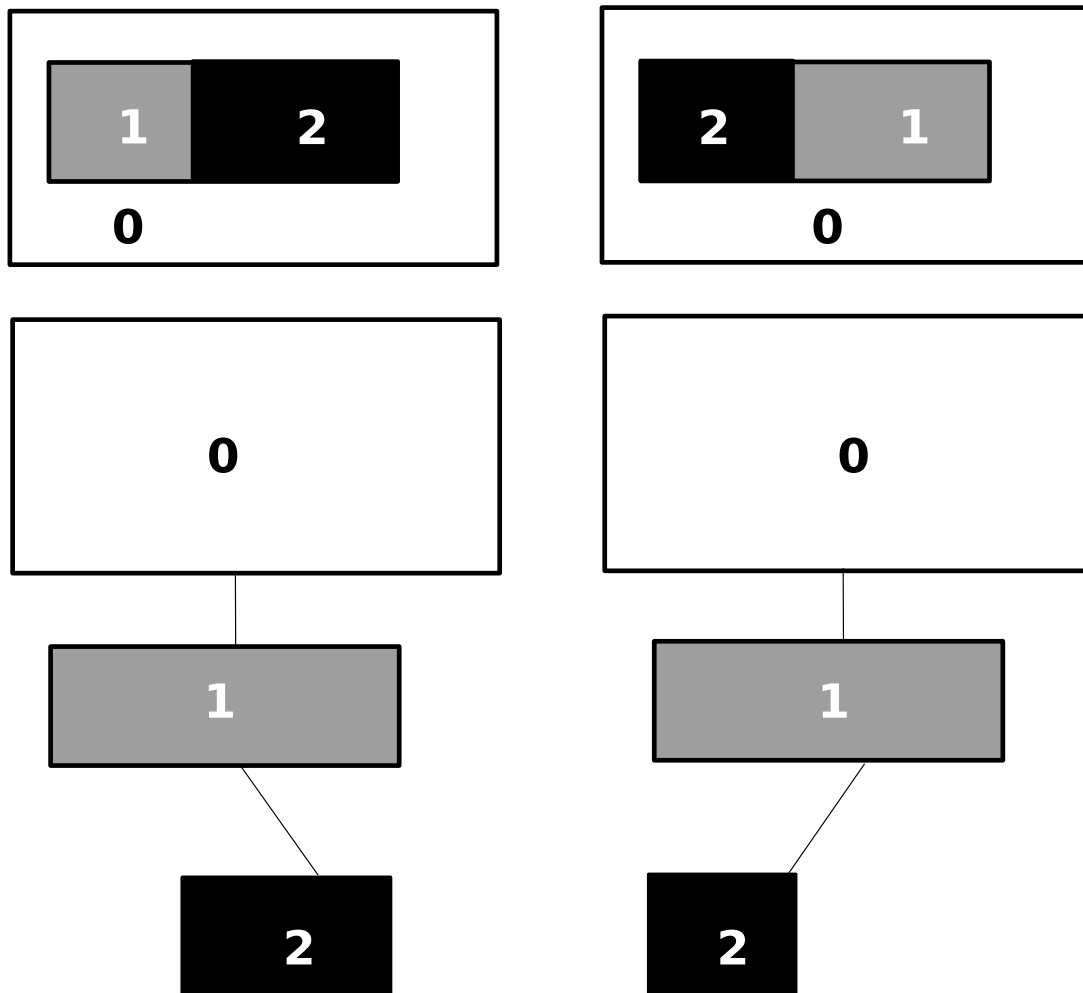


FIGURE 2.13 – Haut : images d'un toit simple à deux versants sous deux illuminations différentes. Bas : FLST de chaque image.

Image de télédétection

On s'intéresse maintenant à des images réelles. Les hypothèses que nous avons émises sur la surface des scènes ne sont valables qu'à des échelles importantes. Les tuiles des toits, par exemple, peuvent rarement être considérées comme développables. Pour appliquer l'algorithme que nous avons décrit précédemment, on ajoute une étape de décomposition

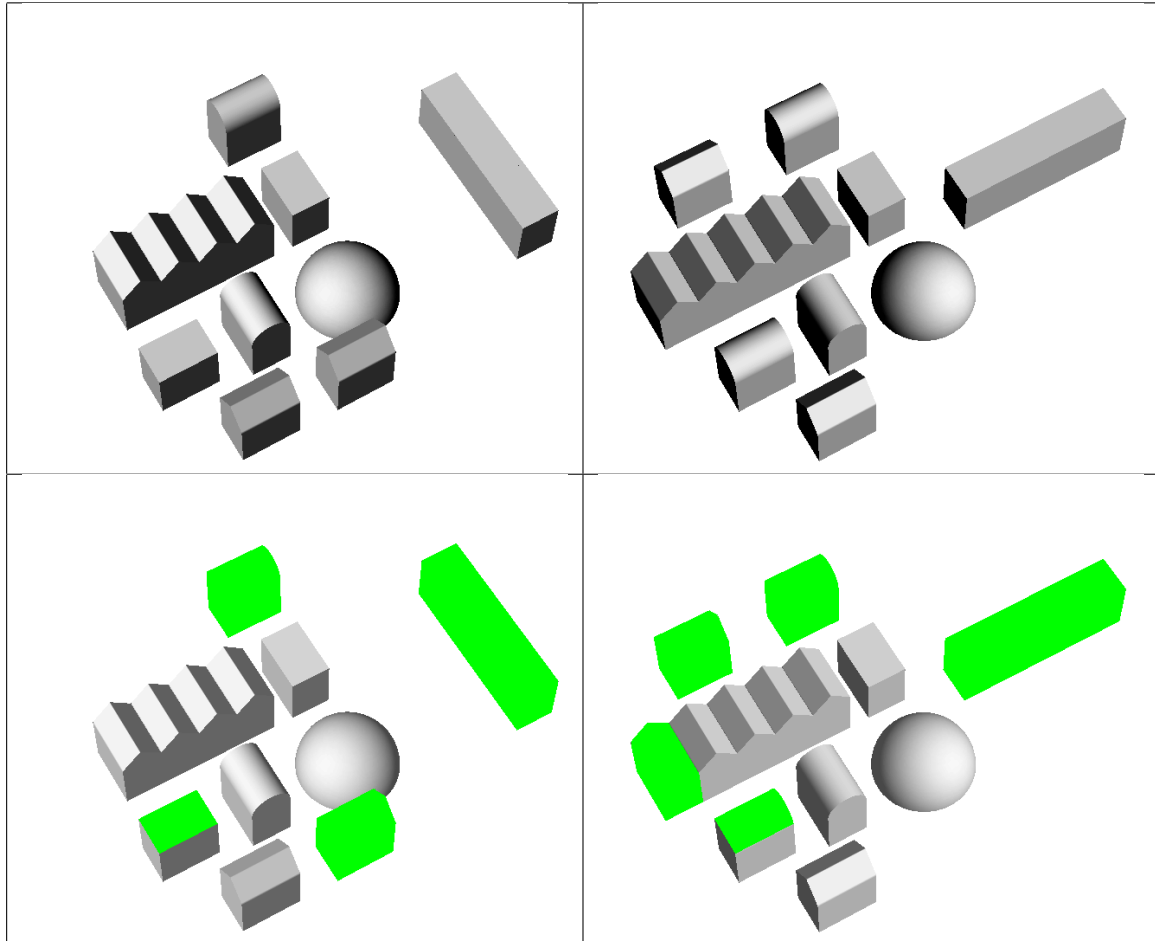


FIGURE 2.14 – Exemple jouet. Haut : Deux images sous des conditions différentes d'illumination, avec par ailleurs quelques changements sur les scènes. Bas : vérités de terrain (les changements sont indiqués en vert).

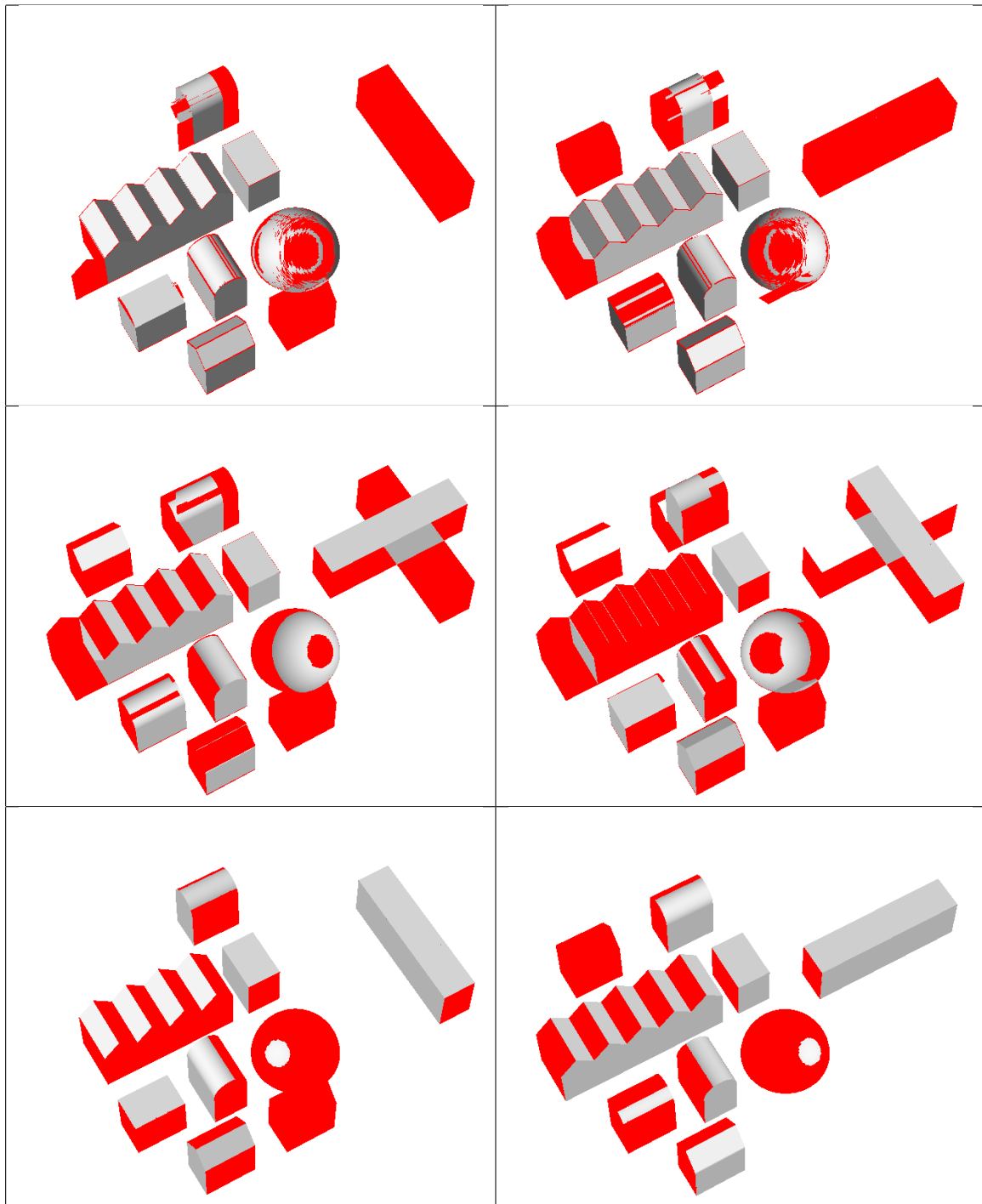


FIGURE 2.15 – Exemple jouet. Première ligne, détection de changements en utilisant notre algorithme. Deuxième ligne : détection de changements en utilisant une égalisation de contraste par moindres carrés. Troisième ligne : détection de changements en utilisant la FLST.

en « cartoon+texture » [Weiss et al., 2007] et on travaille sur la composante « cartoon ». Dans nos simulations, nous avons utilisé le modèle de Rudin-Osher-Fatemi [Rudin et al., 1992]. Par ailleurs, nous n'avons pas considéré les ombres dans notre modèle. Les régions ombragées sont uniquement éclairées par la lumière ambiante. Leur intensité radiométrique peut être considérée comme de l'ordre d'un dixième de celle des régions éclairées par la lumière du soleil. On peut facilement retirer les changements dus aux ombres en omettant les changements de faible intensité. Des techniques bien plus sophistiquées pour ôter les ombres ont été proposées [Watanabe et al., 1998].

Nous proposons des comparaisons sur deux jeux d'images Quickbird. Les trois méthodes dépendent d'un seuil. On fournit donc des courbes ROC pour comparer les performances des algorithmes quand les seuils varient. On rappelle que les vérités de terrain ont été construites à la main.

Aéroport d'Abidjan Sur cette image, la détection de changement est assez aisée et les trois algorithmes donnent de bons résultats. Comparé à une approche classique (*i.e.* égalisation globale de contraste suivie d'une différence pixel-par-pixel), notre algorithme donne des résultats satisfaisants, avec peu de faux négatifs et encore moins de fausses alarmes, et ce en très peu de temps (3 secondes pour cette image avec un processeur Intel Xeon à 2 GHz). Notre implantation de la FLST ne donne pas de résultats satisfaisants comparés à d'autres approches. Ceci est probablement dû au fait que nous ne comparons que les aires et barycentres des zones (en omettant les moments d'ordres supérieur). Ce problème est important, dans le sens où l'algorithme de recherche de correspondances s'exécute en 30 secondes. Les courbes ROC (cf. Figure 2.18) montrent que notre algorithme est sensiblement plus performant que les autres approches. Par exemple, pour obtenir 85 % de vrais positifs, on obtient 5 % de fausses alarmes, alors que les autres algorithmes génèrent tous deux plus de 20 % de fausses alarmes.

Les principales causes d'échec de notre algorithme sont les erreurs de recalage : quelques lignes peintes sur la piste ne sont pas recalées parfaitement et quelques parois d'immeubles ne coïncident pas exactement. Enfin, quelques fausses alarmes sont dues aux changements saisonniers de la végétation autour du parking.

Ville de Pékin Ce couple d'images est beaucoup plus difficile à traiter que le précédent. En effet, de très importantes modifications ont eu lieu sur la scène entre les deux prises de vue. Aussi, nous n'avons réussi à générer qu'une vérité de terrain à une échelle grossière. Notre algorithme donne néanmoins des résultats globalement satisfaisants. Avec un seuil correctement choisi, on obtient 75 % de vrais positifs avec 25 % de fausses alarmes. Les autres méthodes génèrent un taux bien supérieur de fausses alarmes et les résultats qu'elles fournissent semblent difficiles à utiliser (cf. les courbes ROC, Figure 2.20).

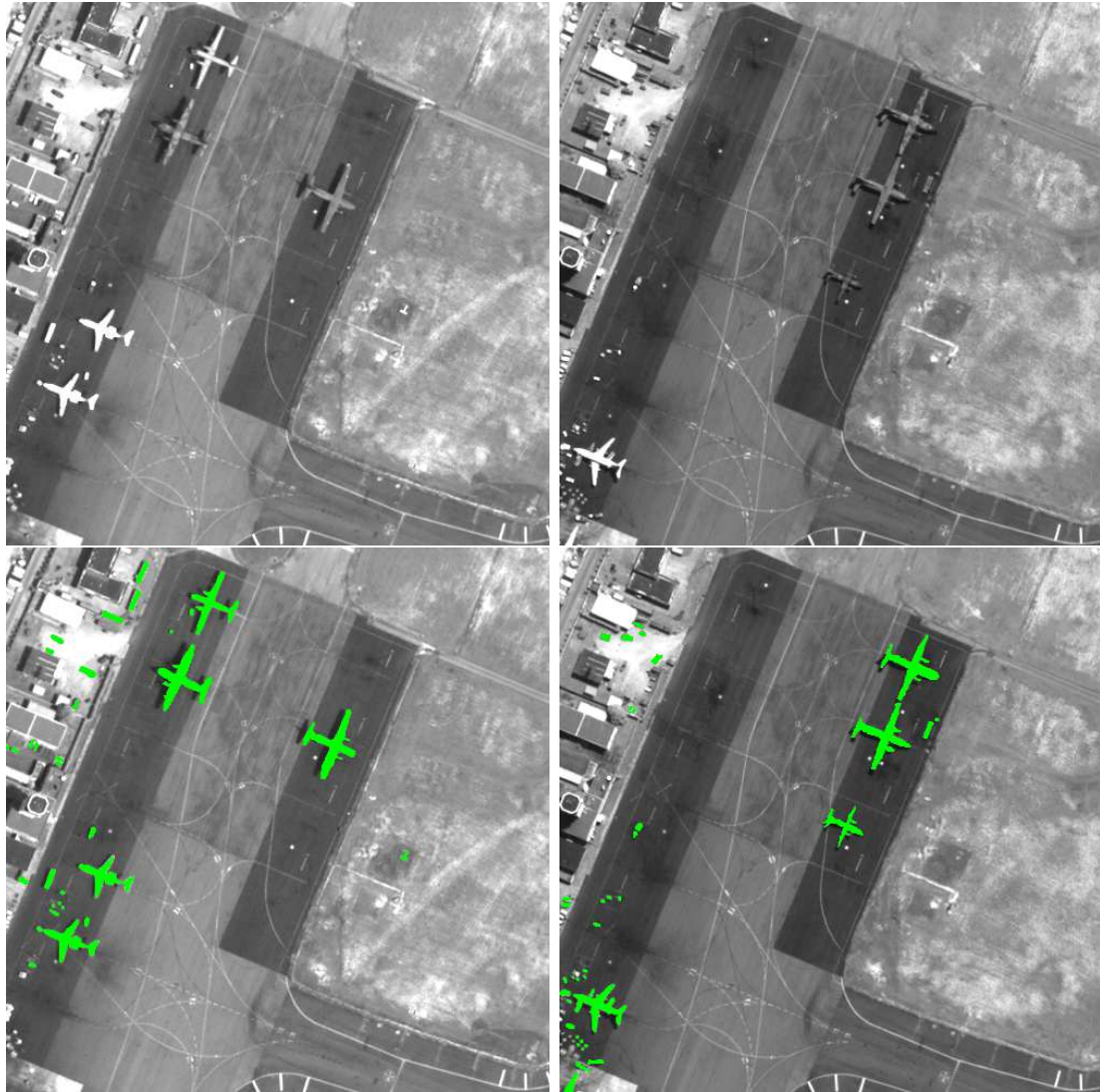


FIGURE 2.16 – Images Quickbird (61 cm de résolution) de l'aéroport d'Abidjan. Haut : l'aéroport le 04/02/2003 et le 07/07/2003. Bas : vérités de terrain (les changements sont indiqués en vert).

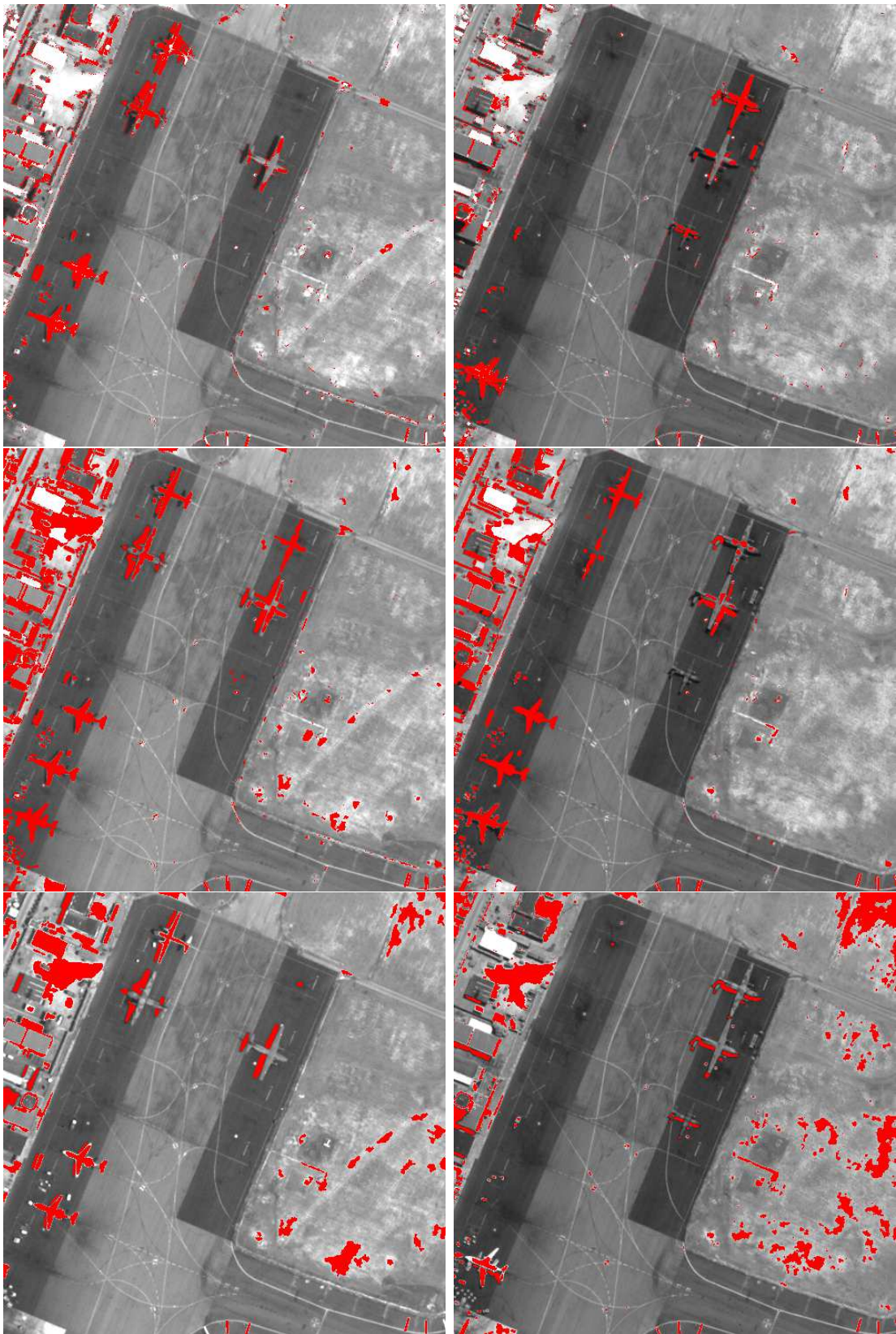


FIGURE 2.17 – Images Quickbird (61 cm de résolution) de l'aéroport d'Abidjan. Première ligne : détection de changements en utilisant notre algorithme. Deuxième ligne : détection de changements en utilisant une égalisation de contraste aux moindres carrés. Troisième ligne : détection de changements en utilisant la FLST.

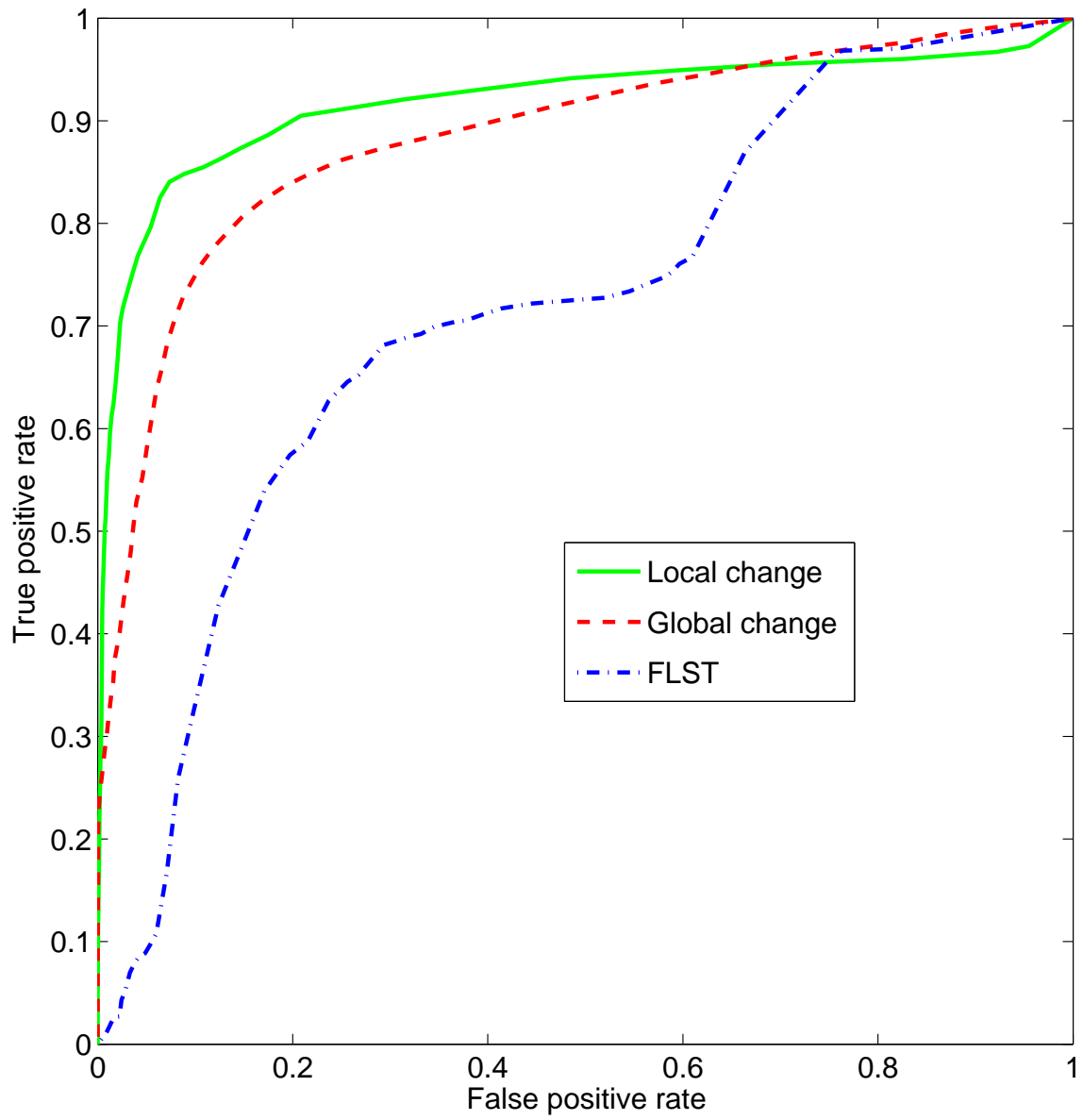


FIGURE 2.18 – Courbes ROC pour l'aéroport d'Abidjan.

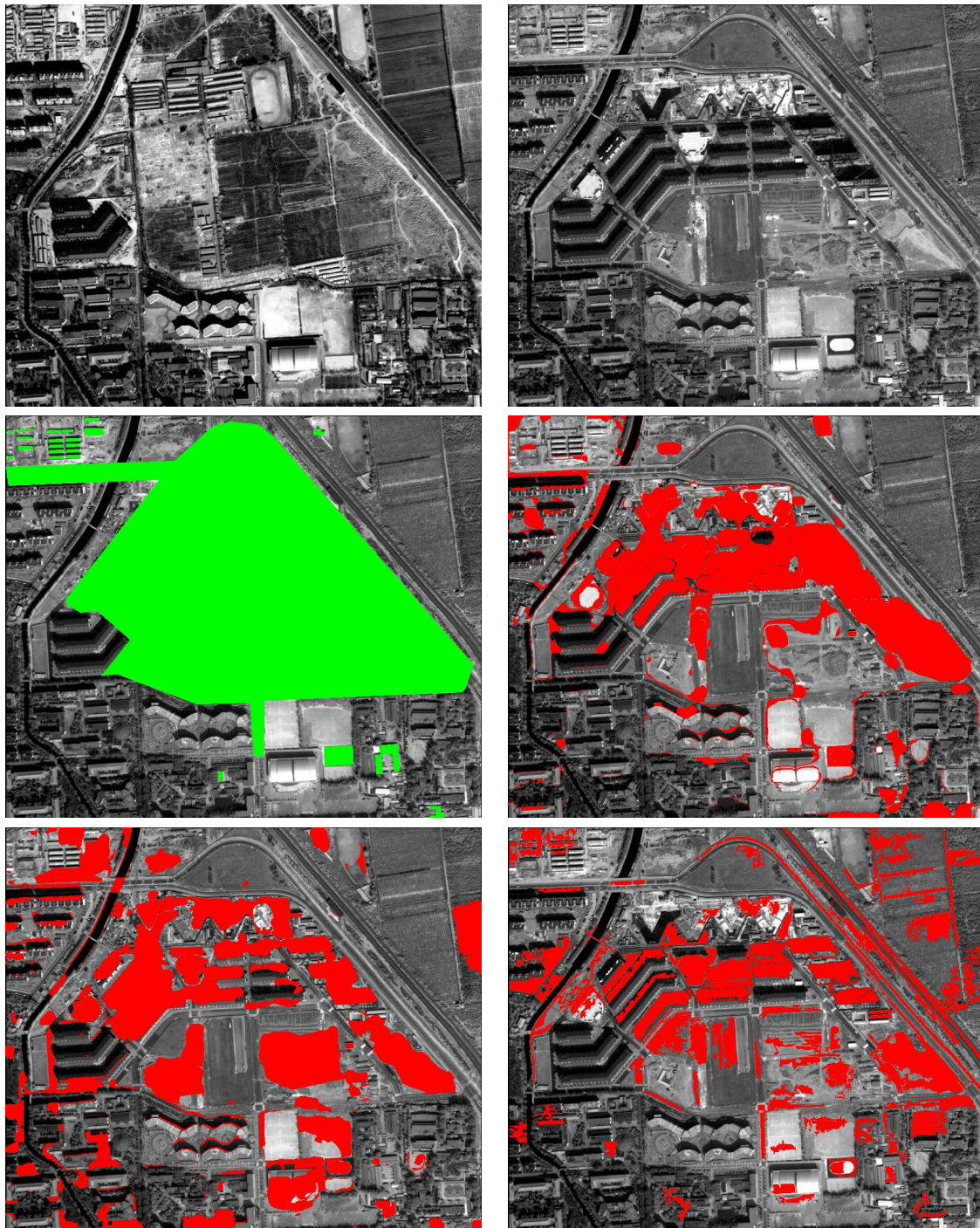


FIGURE 2.19 – Images Quickbird (61cm de résolution) de Pékin. Première ligne : Pékin en 2001 (gauche) et 2003 (droite). Deuxième ligne - gauche : vérité de terrain (les changements sont indiqués en vert) - droite : Détection de changements en utilisant notre algorithme. Troisième ligne - gauche : détection de changements en utilisant un projection monotone - droite : détection de changements en utilisant la FLST.

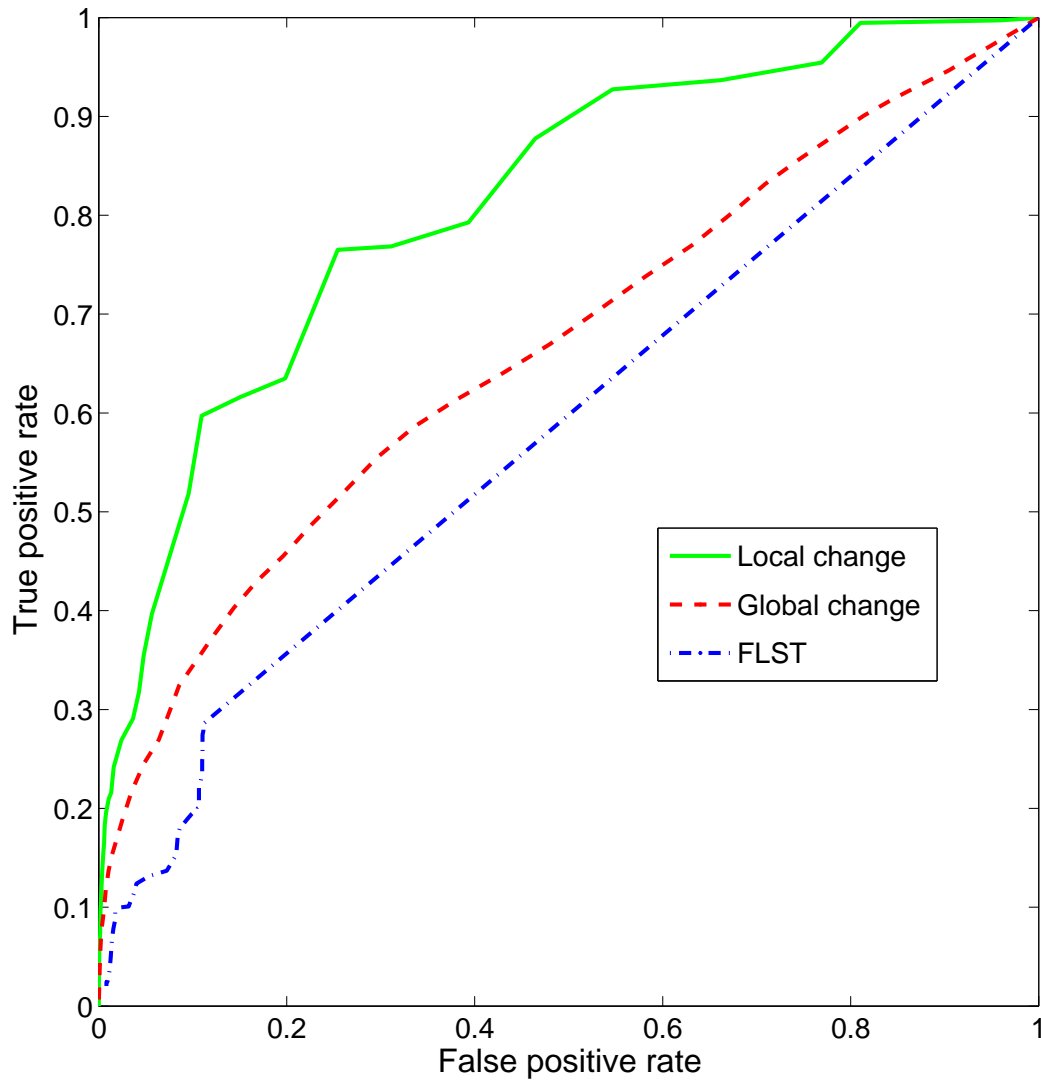


FIGURE 2.20 – Courbes ROC pour le couple d'images de la banlieue de Pékin.

2.3 Conclusion

Dans ce chapitre, nous avons abordé différentes méthodologies permettant d'obtenir un premier masque de changements. Nous avons mis en œuvre des méthodes classiques et agissant de manière globale sur les paires d'images, et nous avons mis en avant les limites de ce type d'algorithmes. Ces limites nous ont conduit à rechercher des invariants en modélisant les scènes à l'aide d'hypothèses relativement fortes. Ainsi, nous avons pu exhiber l'invariance des lignes de niveau selon nos hypothèses. Enfin, nous avons validé ces hypothèses à l'aide d'un algorithme relativement simple.

Les résultats présentés dans ce chapitre sont très encourageants en comparaison avec ceux obtenus par les algorithmes plus classiques généralement utilisés. Les performances de notre algorithme semblent meilleures à la fois sur des images synthétiques et sur des images réelles.

Les perspectives issues de ce travail sont variées, tant sur le plan théorique que sur le plan pratique. Sur le plan théorique, les hypothèses que nous avons posées sont parfois fortes (surfaces développables, composante de réflexion exclusivement lambertienne ...). Une des axes de recherche envisageables est donc l'obtention de résultats théoriques pour des hypothèses un peu moins restrictives, mais plus proches des conditions réelles en télédétection ou en imagerie urbaine. D'autre part, l'algorithme proposé qui tire parti de l'invariance des lignes de niveau est extrêmement simple ; des algorithmes plus sophistiqués fondés sur les mêmes résultats théoriques ont donc de grandes chances de donner de meilleurs résultats en termes de performances.

Chapitre 3

Classification fondée sur la radiométrie : Classification par *k-moyennes*

War don't ennoble men. It turns
them into dogs... poisons the soul.

PRIVATE WITT, THE THIN RED
LINE

Dans le chapitre précédent, nous avons décrit plusieurs algorithmes permettant d'obtenir un masque de détection de changements. En général, il s'agit de la première étape avant une classification des zones détectées, soit en fonction de leur radiométrie, soit en fonction de données plus contextuelles (géométrie, position, connaissances *a priori*). L'étape suivante de notre chaîne de traitement est la réalisation d'une première classification des pixels détectés précédemment comme « changés », uniquement en fonction de leurs radiométries sur les deux prises de vue. Pour ce type de classifications, plusieurs outils sont disponibles, comme les réseaux de neurones, les séparateurs à vaste marge (SVM), les algorithmes de *k-moyennes* et de *k-médianes*, ou les mélanges de gaussiennes [Bishop, 2006]. Les algorithmes fondés sur les *k-moyennes* possèdent plusieurs avantages :

- Par rapport aux réseaux de neurones et aux SVM, ils ne requièrent pas de phase d'apprentissage, et sont en général non-supervisés ;
- Ils sont relativement simples à mettre en œuvre ;
- Leur vitesse de convergence est relativement élevée [Bishop, 2006].

Comme nous le verrons plus loin (cf. Chapitre 4), ces méthodes présentent une qualité supplémentaire : être suffisamment génériques pour s'adapter à des espaces assez variés. En conséquence, c'est cette famille d'algorithmes qui a été retenue pour effectuer la classification. Dans ce chapitre, nous allons d'abord présenter divers aspects théoriques

du problème des k -moyennes et de l'algorithme des k -moyennes (ou algorithme de Lloyd [Lloyd, 1982]). Dans un deuxième temps, nous allons présenter l'algorithme de k -moyennes entropique introduit dans [Palubinskas et al., 1998] qui permet d'estimer le nombre optimal de classes dans la population. Enfin, les résultats de classification sur les masques de changements précédemment obtenus seront présentés et analysés.

3.1 Algorithme de k -moyennes et dérivés

L'algorithme des k -moyennes est un algorithme très classique de classification. Dans un premier temps, nous allons présenter le problème des k -moyennes et montrer sa pertinence pour la classification de points dans un espace métrique¹, puis nous décrirons de manière générique l'algorithme des k -moyennes (ou algorithme de Lloyd). Enfin, nous décrirons quelques propriétés importantes (notamment de convergence) de l'algorithme des k -moyennes.

3.1.1 Problème des k -moyennes

Le problème des k -moyennes se fonde sur la notion de distance d'un espace métrique pour induire une classification d'une suite finie de points résidant dans cet espace. Une des particularités de ce problème est que l'on considère le nombre de classes existantes comme une donnée connue *a priori*. Nous proposons plus loin (voir Partie 3.3.1) une méthode fondée sur la notion d'entropie qui permet d'intégrer un estimateur de ce nombre au sein même de l'algorithme.

Le problème des k -moyennes se fonde sur la notion de moyenne de Karcher (introduite dans [Karcher, 1977]), qui est liée à un espace métrique :

Définition 7 (Moyenne de Karcher). *Soit Ω un espace métrique et d la distance qui lui est associée. Soit $n \in \mathbb{N}^*$ et $x : (x_i)_{1 \leq i \leq n}$ une suite finie d'éléments de Ω . La moyenne de Karcher de x s'écrit alors :*

$$\mu_{\text{Karcher}}(x) = \arg \min_{y \in \Omega} \sum_{i=1}^n \frac{1}{2} d(x_i, y)^2 \quad (3.1)$$

Une des premières remarques que l'on peut faire est que ni l'existence, ni l'unicité de la moyenne de Karcher ne sont assurées dans le cas général. Par exemple, une suite de n points régulièrement espacés dans l'espace des directions $[0, 2\pi[$ possède n moyennes de Karcher.

Par ailleurs, dans un espace euclidien, on remarquera que la moyenne de Karcher et la moyenne arithmétique $\mu(x) = \sum_{i=1}^n \frac{x_i}{n}$ sont confondues.

1. Des résultats récents ont montré [Teboulle et al., 2006] qu'il était possible d'adapter l'algorithme pour des fonctions plus générales, incluant notamment la divergence de Bregman.

Formulation « naturelle »

Munis de la Définition 7, nous sommes maintenant en mesure d'énoncer le problème des *k-moyennes*. La manière la plus intuitive d'énoncer le problème des *k-moyennes* est la suivante :

Problème 1 (Problème des *k-moyennes*). Soit Ω un espace métrique et d la distance associée. Soit (x_i) , $i \in [1..n]$ une suite finie d'éléments de Ω . Soit $k \in \mathbb{R}^{+*}$. On cherche la partition $S : S_{1 \leq j \leq k}$ de Ω composée de k parties minimisant la fonction suivante :

$$E(S) = \sum_{i=1}^k \sum_{x_j \in S_i} \frac{1}{2} d(x_j, \mu_i)^2 \quad (3.2)$$

Où la suite $\mu : (\mu_j)_{1 \leq j \leq k}$ est la suite des moyennes de Karcher des éléments de x appartenant à S_j . Par la suite, nous appellerons les éléments de cette suite les centroïdes.

Formulation alternative

En remplaçant la moyenne de Karcher par sa définition, on peut aisément reformuler le problème des *k-moyennes* en omettant les S_j et en ne tenant compte que des centroïdes :

Problème 2 (Problème des *k-moyennes* (bis)). Soit Ω un espace métrique et d la distance associée. Soit $x : (x_i)_{1 \leq i \leq n}$, une suite finie d'éléments de Ω . Soit $k \in \mathbb{N}^+$ et $\mu : \mu_{1 \leq j \leq k}$ une suite de k éléments de Ω . Le problème des *k-moyennes* revient à déterminer la suite $\bar{\mu}$ définie comme :

$$\bar{\mu} = \arg \min_{\mu \in \Omega^k} \sum_{i=1}^n \frac{1}{2} \min_j d^2(x_i, \mu_j) \quad (3.3)$$

Une fois le minimum connu, il suffit de trouver le centroïde le plus proche de chaque x_i pour déterminer la classification optimale.

3.1.2 Description de l'algorithme des *k-moyennes*

L'algorithme des *k-moyennes* est constitué de plusieurs étapes :

1. la phase d'initialisation affecte des valeurs de départ aux centroïdes. Les méthodologies d'initialisation sont multiples. On peut, par exemple, choisir d'initialiser les centroïdes à des positions occupées par des éléments de x choisis au hasard. D'autres stratégies d'initialisation ont également été envisagées [Arthur et Vassilvitskii, 2007].
2. boucle principale : Cette boucle est constituée de deux étapes importantes :
 - mise à jour des classes** Au cours de cette phase, chaque x_i est affecté à la classe dont le centroïde est le plus proche.

mise à jour des centroïdes On évalue chaque centroïde μ_j comme la moyenne de Karcher des x_i qui ont été affectés à la classe S_j .

À l'issue de cette mise à jour, l'énergie E est comparée à l'énergie obtenue à l'étape précédente. Si elles sont égales, on quitte la boucle principale.

Cette division de la boucle principale en deux phases distinctes n'est pas particulière aux algorithmes de *k-moyennes*. Dans [Bottou et Bengio, 1995], les auteurs montrent que cette spécificité fait de l'algorithme des *k-moyennes* un algorithme très proche des algorithmes de type *Espérance-Maximisation*. Une description plus précise de l'algorithme des *k-moyennes* en pseudo code est donné ci-dessous.

Algorithme 3.1 : Algorithme des *k-moyennes*

Données : $(x_i)_{1 \leq i \leq n}$: des points de Ω , $k > 0$

Résultat : $(\mu_j)_{1 \leq j \leq k}$ les centroïdes des classes optimales, $(\mu'_i)_{1 \leq i \leq n}$: pour chaque x_i , le μ_j le plus proche

début

```

Initialisation des centroïdes  $\mu_j$  (voir Partie 3.1.3);
/*initialisation des classes                                     */
pour chaque  $1 \leq i \leq n$  faire
    |  $\mu'_i \leftarrow \arg \min_{\mu_j} d(x_i, \mu_j)$ 
fin
/*Calcul de l'énergie totale :                                   */
 $E_0 \leftarrow \sum_{i=1}^n \frac{1}{2} d(x_i, \mu'_i)^2$ ;
répéter
    |  $t \leftarrow t + 1$ ;
    | /*mise à jour des classes :                                 */
    | pour chaque  $1 \leq i \leq n$  faire
    | |  $\mu'_i \leftarrow \arg \min_{\mu} d(x_i, \mu_j)$ 
    | fin
    | /*mise à jour des centroïdes :                             */
    | pour chaque  $1 \leq j \leq k$  faire
    | |  $\mu_j \leftarrow \mu_{\text{Karcher}}(x_i \text{ tels que } \mu'_i = \mu_j)$ ;
    | fin
    | /*Calcul de l'énergie totale                               */
    |  $E_t \leftarrow \sum_{i=1}^n \frac{1}{2} d(x_i, \mu'_i)^2$ ;
jusqu'à  $E_t = E_{t-1}$  ;

```

fin

Il s'agit ici d'une description assez naïve de l'algorithme des *k-moyennes* et beaucoup d'améliorations ont été proposées pour accélérer la convergence de l'algorithme. Entre autres, lorsque les éléments de x sont fortement redondants (notamment dans le cas des

valeurs des pixels d'une image), il est intéressant d'avoir une approche « par histogramme » qui travaille sur chaque valeur prise par x en lui affectant un poids p_i .

3.1.3 Propriétés de Convergence

Chacune des deux phases de la boucle principale a pour résultat une diminution de l'énergie totale E_t . En conséquence, on a une garantie de décroissance de cette énergie.

En réalité, la formulation du Problème 2 est utilisée dans [Bottou et Bengio, 1995] pour mettre en évidence l'algorithme des *k-moyennes* comme un algorithme de descente de gradient. Deux conséquences essentielles découlent de cette considération :

- on a une garantie de convergence vers un minimum de la fonction d'énergie $E(\mu)$,
- ce minimum est un minimum local et la qualité de ce minimum peut être fortement dépendante de l'initialisation de l'algorithme.

Pour essayer de contourner ce défaut majeur de l'algorithme, la solution utilisée le plus souvent est de tirer parti de la rapidité de convergence de l'algorithme pour le lancer plusieurs fois avec des initialisations différentes et choisir la classification la meilleure au sens de l'énergie $E(\mu)$. Des résultats récents [Arthur et Vassilvitskii, 2007] montrent que l'algorithme des *k-moyennes* n'offre aucune garantie sur la qualité de la classification finale (c'est à dire que le ratio $\frac{E_{finale}}{E_{optimale}}$ n'est pas borné, même à k et à n fixés). Les auteurs proposent alors une méthode d'initialisation aléatoire de l'algorithme qui garantit une énergie finale inférieure à $8(\ln k + 2)E_{optimale}$.

3.2 Définition de l'espace d'état

3.2.1 Espace d'état

Dans notre application, l'espace d'état est relativement simple. Il s'agit des radiométries aux deux instants de prise de vue des pixels de l'image. Les points x_i possèdent alors deux coordonnées x_i^1 et x_i^2 . Dans notre cas particulier, les images traitées étaient codées sur 8 bits, les valeurs des radiométries s'échelonnant alors entre 0 et 255.

3.2.2 Distance

Pour compléter la description de l'espace d'état, il est nécessaire de définir une distance. On pourrait utiliser la distance euclidienne. Néanmoins, comme nous l'avons vu dans le chapitre 2, les radiométries suivent une dynamique particulière. Nous essayons donc de suivre cette dynamique en utilisant la distance de Mahalanobis entre les pixels au lieu de la distance euclidienne :

$$d_{Mah}(x_1, x_2) = \sqrt{(x_1 - x_2)^T \Sigma^{-1} (x_1 - x_2)}. \quad (3.4)$$

où Σ représente la matrice de covariance de la suite x . Un rapide calcul montre que la moyenne de Karcher associée à cette distance correspond également à la moyenne arithmétique. Dans la pratique, on applique l'homomorphisme associé à la matrice $\sqrt{\Sigma}^{-1}$ aux x_i et μ_j d'entrée afin de travailler directement avec la distance euclidienne puis la transformation inverse est appliquée après convergence.

3.3 Évaluation du nombre de classes

Comme nous l'avons vu précédemment, l'algorithme des *k-moyennes* présente un grand intérêt lorsque le nombre de classes désirées est connu à l'avance. Dans le cas contraire, il devient nécessaire d'estimer ce nombre. Il s'agit d'une opération assez fastidieuse qui fait appel à des notions souvent arbitraires. Prenons par exemple une distribution de points dont la densité de probabilité serait représentée Figure 3.1. On peut alors considérer que l'on a affaire à deux populations bien marquées et donc initialiser à 2 le nombre de partitions de l'algorithme de *k-moyennes*.

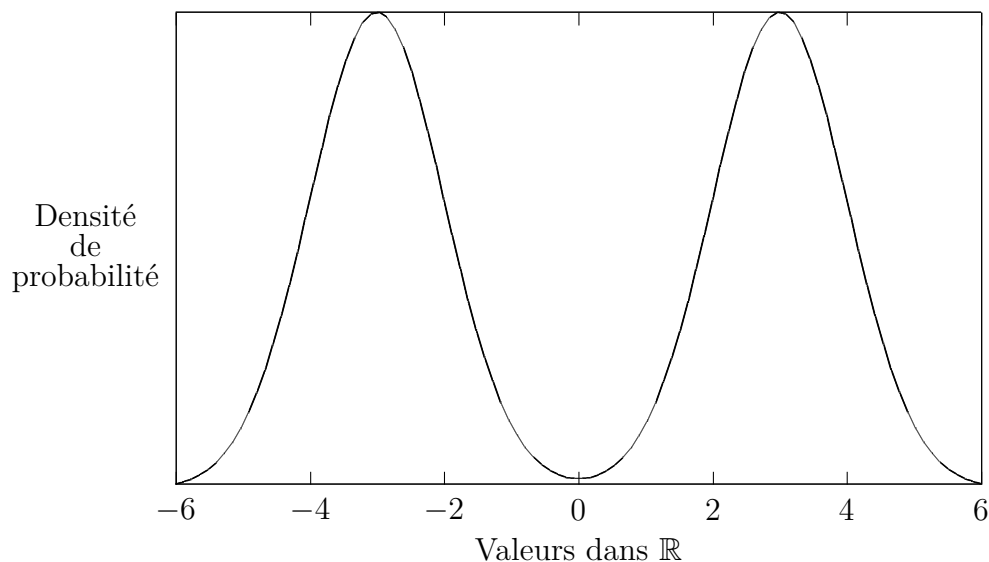


FIGURE 3.1 – Deux populations « bien marquées »

En revanche, si l'on considère une population de points qui suit la densité de probabilité donnée Figure 3.2, alors le nombre « idéal » de classes est sujet à débat. Si l'on cherche à avoir une classification « fine », on estimera le nombre de classes à 4, mais si l'on cherche à avoir une classification plus grossière, alors l'on aura tendance à estimer ce nombre à 2.

De manière générale, plus le nombre de centroïdes est élevé, plus l'énergie $E_{Optimale}$ fournie par la solution du problème des *k-moyennes* est basse ; dans le cas-limite où le

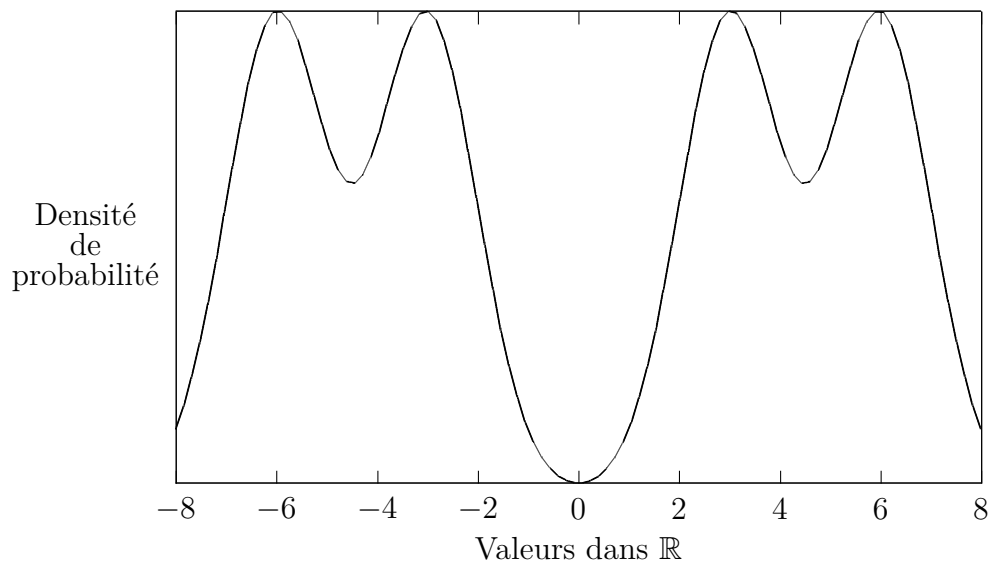


FIGURE 3.2 – Cas « litigieux »

nombre de centroïdes est égal au nombre de points uniques dans la suite x_i , ainsi l'énergie du minimum est nulle. Mais la classification alors obtenue n'a guère de sens.

Entre les deux extrêmes, il est donc assez difficile de donner une définition précise d'un nombre pertinent de classes. Dans le cas où des problèmes d'échelles similaires sont à traiter, une piste à envisager pour la comparaison des résultats serait l'utilisation d'une quantité dite de « granularité ». Une propriété essentielle de ce type d'expression serait alors une bonne robustesse au nombre de populations. Une expression naïve de ce type de quantité pourrait être alors de la forme :

$$G = \frac{D(\mu, x)}{D(\mu)} \quad (3.5)$$

où $D(\mu, x)$ représente une distance caractéristique entre un point et le centroïde qui lui est associé dans la classification et $D(\mu)$ représente une distance caractéristique entre deux centroïdes différents de μ (on pourra prendre, par exemple, la distance médiane ou moyenne). On remarque, par exemple, que la granularité diminue lorsque les différences entre les classes sont bien marquées (grande distance entre les centroïdes), mais augmente lorsque les classes sont en moyenne assez dispersées.

3.3.1 *k*-moyennes entropiques

Terme d'entropie

Afin d'automatiser, ou du moins de faciliter l'évaluation du nombre de classes, l'ajout d'un terme d'entropie à l'algorithme des *k*-moyennes a été proposé dans [Palubinskas et al., 1998]. Il s'agit d'une application assez différente de la notre (classification des pixels d'une image afin de détecter des tissus du cerveau humain), néanmoins, nous avons repris la méthodologie proposée afin de l'appliquer à notre application.

Plus précisément, les auteurs considèrent la classification des pixels comme un problème bayésien. Ils cherchent alors à maximiser la probabilité *a posteriori* de classification :

$$P(\mu|x) = \frac{P(x, \mu)P(\mu)}{P(x)} \quad (3.6)$$

où l'on considère x et μ comme des variables aléatoires (par la suite, nous utilisons la même notation pour ces variables aléatoires et leur réalisation). Le terme de vraisemblance est sous une forme très classique ; on suppose les pixels indépendants les uns des autres, d'où un terme de vraisemblance de la forme :

$$P(X|Y) = \exp\left(\sum_{i=1}^n \ln p(x_i|\mu_i)\right) \quad (3.7)$$

Le choix de $p(x_i|\mu_i)$ permet de retrouver l'expression classique des *k*-moyennes :

$$p(x_i|\mu_i) \propto \exp\left(-\frac{1}{2}d(x_i, \mu_i)^2\right) \quad (3.8)$$

sur la fonction d'énergie à minimiser. Ce terme a pour effet l'inclusion du nombre de classes en tant que variable de la fonction. Dans un premier temps, ils proposent d'initialiser le nombre de classes au nombre d'éléments *uniques* de x .

L'originalité de la contribution réside dans le choix de terme de probabilités *a priori*. Ce terme doit pouvoir favoriser les solutions comportant un nombre de classes assez faible (à population totale fixée). Les auteurs ont donc fait appel à la définition de l'entropie au sens de Shannon [Shannon, 1948]. Pour une classification μ donnée, on peut définir l'entropie comme :

$$\mathcal{S} = -\sum_{j=1}^k p_j \ln(p_j) \quad (3.9)$$

où $p_j = \frac{\text{card}(S_j)}{n}$ est la probabilité *a priori* pour un élément de x d'appartenir à la classe S_j . Par exemple, on remarque que l'entropie est minimale avec un partitionnement pour lequel une classe est prédominante et les autres classes sont vides. Au contraire, elle est maximale lorsque la population de x est uniformément répartie sur les différents μ_j . L'entropie est utilisée directement dans l'expression de la probabilité *a priori* :

$$P(\mu) \propto \exp \left(\alpha_E \sum_{j=1}^k p_j \ln p_j \right) \quad (3.10)$$

La probabilité *a posteriori* s'écrit alors :

$$P(\mu|x) \propto \exp \left(\sum_{i=1}^n \ln p(x_i|\mu_i) + \alpha_E \sum_{j=1}^k p_j \ln p_j \right) \quad (3.11)$$

On cherche un maximum de probabilité *a posteriori*. Pour l'atteindre, il faut donc minimiser le terme d'énergie :

$$E(\mu) = - \sum_{i=1}^n \ln p(x_i|\mu_i) - \alpha_E \sum_{j=1}^k p_j \ln p_j \quad (3.12)$$

où le terme α_E correspond à un compromis entre le terme d'entropie et l'énergie que l'on cherchait à minimiser auparavant, *i.e.* un compromis entre l'homogénéité et le nombre des classes. Le choix de α_E va directement influencer sur la granularité que nous recherchons, et « pilote » indirectement le nombre de classes.

Par ailleurs, on remarque que l'influence du premier terme de $E(\mu)$ croît avec la taille de l'image, contrairement au terme d'entropie (qui ne dépend que de probabilités). Afin d'assurer une invariance à la taille de l'image, on note alors $\alpha_E = \alpha n$. On peut alors réécrire l'énergie à minimiser :

$$E(\mu) = - \sum_{i=1}^n \ln p(x_i|\mu_i) - \alpha n \sum_{j=1}^k p_j \ln p_j \quad (3.13)$$

$$E(\mu) = - \sum_{i=1}^n \ln p(x_i|\mu_i) - \alpha \sum_{j=1}^k \text{card}(S_j) \ln p_j \quad (3.14)$$

$$E(\mu) = - \sum_{j=1}^k \sum_{x_i \in S_j} (\ln p(x_i|\mu_i) + \alpha \ln p_j) \quad (3.15)$$

Enfin, en remplaçant, dans l'Équation 3.15, les vraisemblances par leur expression, proposée dans l'Équation 3.8, on obtient, à une constante (omise) près :

$$E(\mu) = \sum_{j=1}^k \sum_{x_i \in S_j} \left(\frac{1}{2} d(x_i, \mu_j)^2 - \alpha \ln p_j \right) \quad (3.16)$$

Une remarque importante, est que bien que le nombre k des classes soit fixé, de même que précédemment, il est tout à fait envisageable d'obtenir des classes complètement vides. Auquel cas, on pourra simplement les supprimer de l'algorithme. Ce faisant, cette nouvelle énergie propose un nombre k' de classes non vide variable et majoré par k .

On peut alors formuler un nouveau problème des k -moyennes entropiques, dont la solution donnera un certain compromis entre nombre de classes et finesse de la classification :

Problème 3 (Problème des k -moyennes entropiques). Soit Ω un espace métrique et d la distance associée. Soit (x_i) , $i \in [1..n]$ une suite finie d'éléments de Ω . Soit $k \in \mathbb{R}^{+*}$. On cherche la partition $S : S_{1 \leq j \leq k}$ de Ω composée de k parties minimisant la fonction suivante :

$$E(S) = \sum_{i=1}^k \sum_{x_j \in S_i} \frac{1}{2} d(x_j, \mu_i)^2 - \alpha \ln \left(\frac{\text{card}(S_j)}{n} \right) \quad (3.17)$$

3.3.2 Algorithme des k -moyennes entropiques

Une nouvelle forme d'énergie ayant été proposée, nous pouvons maintenant proposer un nouvel algorithme de k -moyennes entropiques, permettant d'atteindre un compromis sur la finesse de la classification.

Le choix du paramètre α est assez problématique. Une méthode heuristique fondée sur les lois d'échelle est proposée dans [Palubinskas et al., 1998].

Dans notre cas particulier, nous avons opté pour un α variable. Anne Lorette a travaillé sur des problèmes de classification sensiblement équivalents. [Lorette, 1999] reprend la méthodologie proposée dans [Frigui et Krishnapuram, 1996] en initialisant α à une valeur relativement grande afin d'obtenir un nombre pertinent de classes, puis en diminuant sa valeur suivant un schéma de décroissance exponentielle. En effet, une fois le nombre de classes obtenu, rien ne justifie *a priori* de biaiser la classification par le terme d'entropie.

L'Algorithme 3.2 des k -moyennes entropiques est une adaptation de l'algorithme des

k -moyennes au problème des k -moyennes entropiques, décrit ci-après :

Algorithme 3.2 : Algorithme des k -moyennes entropique

Données : $(x_i)_{1 \leq i \leq n}$: des points de Ω , $k \gg k_{estimate}$, α_0 : valeur initiale de α , τ constante de temps du schéma de décroissance de α

Résultat : $(\mu_j)_{1 \leq j \leq k}$ les centroïdes des classes optimales, $(\mu'_i)_{1 \leq i \leq n}$: pour chaque x_i , le μ_j le plus proche

début

```

Initialisation des centroïdes  $\mu_j$  (voir Partie 3.1.3);
/*initialisation des classes                                     */
pour chaque  $1 \leq i \leq n$  faire
    |  $\mu'_i \leftarrow \arg \min_{\mu_j} d(x_i, \mu_j) - \alpha \ln \left( \frac{\text{card}(\mu_j)}{n} \right)$ 
fin
/*Calcul de l'énergie totale :                                  */
 $E_0 \leftarrow \sum_{i=1}^n \frac{1}{2} d(x_i, \mu'_i)^2$ ;
répéter
    |  $t \leftarrow t + 1$ ;
    | /*mise à jour des classes :                               */
    | pour chaque  $1 \leq i \leq n$  faire
    | |  $\mu'_i \leftarrow \arg \min_{\mu} d(x_i, \mu_j) - \alpha \ln \left( \frac{\text{card}(\mu_j)}{n} \right)$ 
    | fin
    | /*mise à jour des centroïdes :                             */
    | pour chaque  $1 \leq j \leq k$  faire
    | |  $\mu_j \leftarrow \mu_{\text{Karcher}}(x_i \text{ tels que } \mu'_i = \mu_j)$ ;
    | fin
    | /*Calcul de l'énergie totale                               */
    |  $E_t \leftarrow \sum_{i=1}^n \frac{1}{2} d(x_i, \mu'_i)^2$ ;
    |  $k \leftarrow k_0 \exp \left( \frac{-t}{\tau} \right)$ ;
jusqu'à  $E_t = E_{t-1}$  ;

```

fin

En réalité, le critère d'arrêt défini ci-dessus n'est pas forcément idéal. En effet, il est possible que l'algorithme ait convergé, mais que l'énergie soit modifiée du fait de la variation de α . On peut alors envisager plusieurs solutions :

- spécifier un seuil de variation de l'énergie au dessous duquel on considère que l'algorithme a convergé,
- comparer l'état des centroïdes entre deux itérations successives (c'est la technique que nous avons retenu).

Implantation de l'algorithme

Les algorithmes de *k-moyennes* et de *k-moyennes* entropiques ont été écrits en C++ au cours de ce travail de thèse. L'objectif de cette écriture était de la mise au point d'un algorithme de *k-moyennes* complètement générique, pouvant s'adapter de façon rapide à tout type d'espace métrique.

En conséquence, l'algorithme a été implanté en utilisant de façon extensive les patrons (« templates ») afin de passer les foncteurs représentant la distance et les opérations de moyenne en arguments de ce problème. Cette conception générique du code a permis de réutiliser le programme pour des applications totalement différentes de la détection de changements, comme la détection de feux de forêt [Zammit, 2008]. Une application plus originale de cette généricité est, par ailleurs, décrite dans le Chapitre 4.

Enfin, cette implantation a été intégrée à la bibliothèque de traitement d'images ARIANA-LIB, développée par un ingénieur associé de l'EPI ARIANA.

3.4 Régularisation par champs de Markov

3.4.1 Notations

Afin d'améliorer la classification, nous avons opté pour une régularisation des résultats *a posteriori*. À l'issue de la classification par *k-moyennes*, nous disposons de n classes (et implicitement, de la classification de chaque pixel, par le biais de la distance de notre espace métrique). Nous souhaitons alors ajouter une composante de régularité à la classification. La régularisation par champs de Markov fait appel à la notion de voisinage et prend en charge le concept de classes. Cette technique semble donc bien adaptée à ce problème. Des rappels théoriques sur les définitions des champs de Markov en traitement d'images sont présentés dans l'Annexe A. Dans cette partie, on note S la trame de l'image et $s \in S$ les pixels de l'image. On note $(x_s)_{s \in S}$ l'ensemble des couples de radiométries des pixels dans les images originales.

Le champ de Markov que nous utilisons pour cette régularisation s'appuie sur les classes $C = c_j$ obtenues grâce à l'algorithme de *k-moyennes* entropiques (et caractérisées par leurs centroïdes μ_j). L'espace Ω des configurations est donc évidemment C^S .

3.4.2 Modèle *a priori* (modèle de Potts)

On reprend les notations de l'Annexe A. Ici, les variables aléatoires (les sites du champ de Markov) sont les pixels de l'image et leur espace d'état est l'ensemble des classes auxquelles peut appartenir chaque pixel. Le modèle de Potts permet de favoriser les configurations présentant des pixels appartenant à la même classe, ce qui favorise la régularité de l'image. On aurait également pu proposer un modèle plus complexe favorisant les classes les plus proches, mais cela induit des effets « de bord » au niveau des frontières entre des

zones classées différemment. D'après le modèle de Potts, la probabilité conditionnelle d'appartenance à une classe j pour un pixel X_i , sachant les classes auxquelles appartiennent ses voisins, s'écrit de la façon suivante :

$$p(\omega_s | \{\omega_{s'}, s' \in V(s)\}) = \frac{1}{Z} \exp \left(- \sum_{k \in V(i)} (1 - \delta(\omega_s, \omega_{s'})) \right) \quad (3.18)$$

où Z est une constante de normalisation et $\delta(\omega_s, \omega_{s'})$ est égal à 1 si $\omega_s = \omega_{s'}$, et à 0 sinon. La régularité se traduit par le fait que la probabilité conditionnelle qu'un pixel soit assigné à une classe est maximale lorsque tous ses voisins sont également assignés à cette classe.

3.4.3 Attache aux données (vraisemblance)

L'attache aux données est introduite par le biais de cliques de cardinal 1 (singletons). La vraisemblance est fondée sur la distance (dans l'espace des couples de radiométries) entre la réalisation w_s sur un site et le couple de radiométries x_s de l'image originale. Afin d'éviter de biaiser les résultats, la distance utilisée est toujours la distance de Mahalanobis, présentée précédemment. L'expression de l'attache aux données pour un pixel s particulier s'écrit alors :

$$p(x_s | \omega_s) = \frac{1}{Z} \exp(-d(\mu(\omega_s), x_s)) \quad (3.19)$$

où $\mu(\omega_s)$ représente le centroïde de la classe ω_s .

3.4.4 Probabilité *a posteriori*

La probabilité *a posteriori* s'écrit finalement sous la forme :

$$p(\omega_s | x_s) = \frac{1}{Z} \exp \left(-d(\mu(\omega_s), x_s) - \beta \sum_{s' \in V(s)} (1 - \delta(\omega_s, \omega_{s'})) \right) \quad (3.20)$$

La probabilité *a posteriori* pour une image complète s'écrit alors :

$$p(w|x) = \frac{1}{Z} \exp \left(- \sum_{s \in S} d(\mu(\omega_s), x_s) - \beta \sum_{k \in K_2} (1 - \delta(k)) \right) \quad (3.21)$$

où K_2 est l'ensemble des cliques de cardinal 2 et $\delta(k)$ est égal à 1 si les deux sommets de k ont la même valeur et 0 sinon.

3.4.5 Recuit Simulé

De même que dans le Chapitre 2, on a recours au recuit simulé afin de déterminer la configuration correspondant au maximum de probabilité *a posteriori*. Cet algorithme est décrit de façon générique dans l'Annexe A.

3.5 Résultats

Les algorithmes ont été testés sur plusieurs jeux d'images. Néanmoins, la validation des résultats est principalement qualitative. En effet, des vérités de terrain auraient été nécessaires pour pouvoir évaluer de manière quantitative les résultats de la classification, mais nous n'avons pas pu en acquérir ou obtenir l'intervention d'un photo-interprète. Pour l'obtention d'un masque de changements (voir Chapitre 2), les vérités de terrain ont été générées manuellement, mais le travail est beaucoup trop long et les résultats trop hasardeux pour renouveler l'opération au niveau de la classification.

L'objectif de ces expériences n'est pas forcément de partir du masque de changements « idéal » pour classer les différents types de changements, mais également de déterminer s'il est possible de discriminer les fausses alarmes détectées par les algorithmes précédents dans la chaîne de traitement (cf. Chapitre 2) des changements « réels » survenus sur la scène.

Les programmes ont été testés sur un processeur Intel Xeon cadencé à 3,00 GHz.

3.5.1 Zone commerciale d'Abidjan

Le premier jeu d'images testées est une zone commerciale d'Abidjan. Les vues sont prises à 4 mois d'intervalle. Ces images, de taille 1000×1000 , sont présentées Figure 3.3. Les pixels détectés comme changés par la méthode de Wiemker ont été classifiés (voir le masque de changements présenté Figure 3.3). Pour cette image, l'algorithme des *k-moyennes* entropiques converge en 8 secondes. Suivant les paramètres, le nombre de classes de couples de radiométries détectées oscille entre 4 et 10. Il est néanmoins nécessaire de rappeler que le nombre de centroïdes présents influence directement la vitesse de convergence de l'algorithme. Les premières itérations, au cours desquelles le nombre de centroïdes est élevé, sont donc beaucoup plus lentes que les suivantes, lorsque le nombre de centroïdes a diminué grâce à l'influence du terme entropique.

Nous montrons sur les Figures 3.4 et 3.5 les classes résultantes pour des valeurs de paramètres différentes. L'analyse des résultats est un peu plus ardue. Bien que les vérités de terrain n'aient pas été établies à ce niveau, on peut faire une courte typologie des changements détectables : apparitions/disparitions de véhicules, apparitions/disparitions de bâtiments. Parmi les fausses alarmes intéressantes, on dénote les problèmes de parallaxe, les différences locales d'illumination (le masque de changements étant celui fourni par l'al-

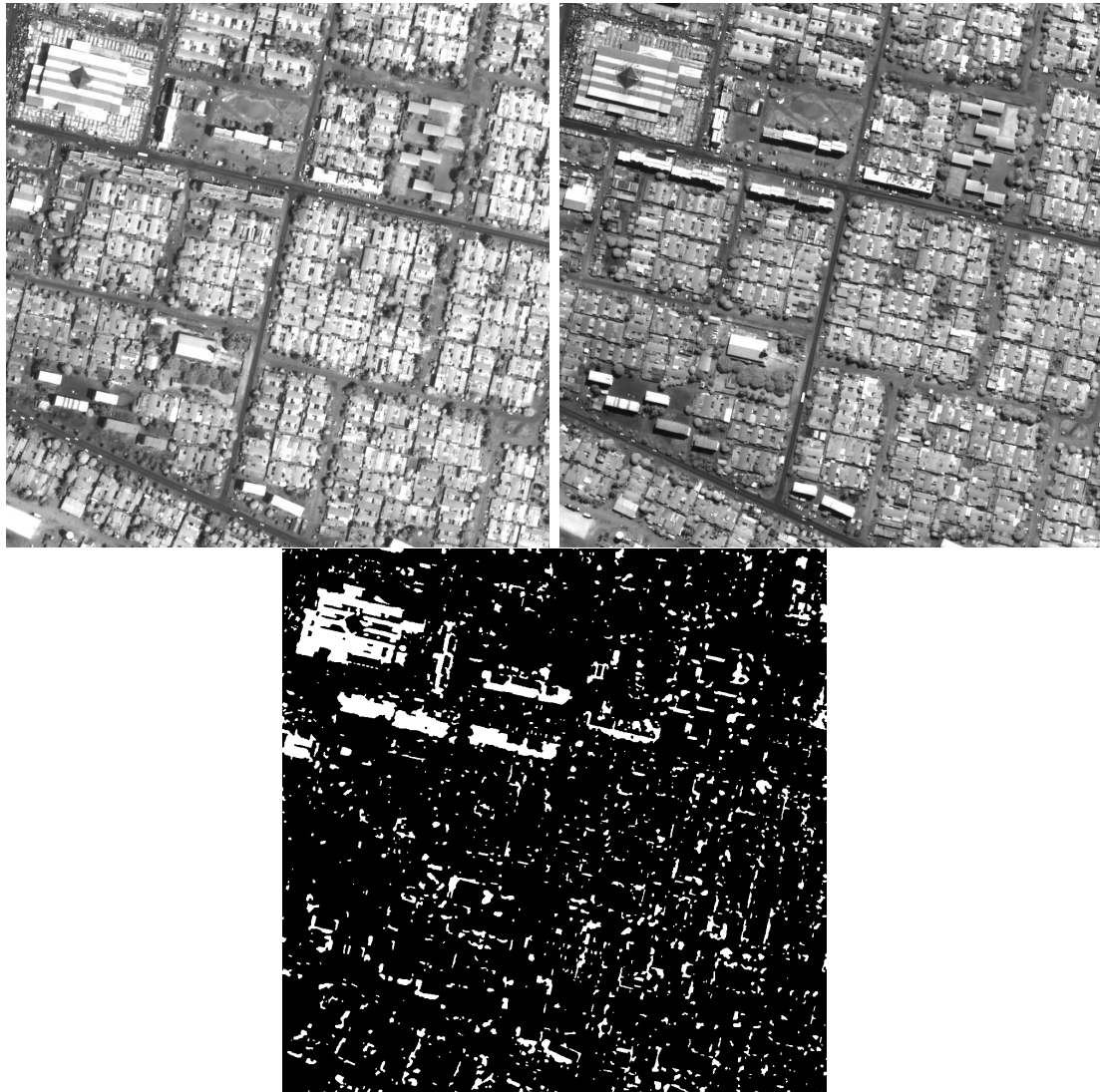


FIGURE 3.3 – Haut : Images originales d'Abidjan, © Spot Image fournie par la DGA.
Bas : Masque de changements obtenu par la méthode de R. Wiemker (cf. Chapitre 2)

gorithme de [Wiemker, 1997]), et les ombres. D'autres fausses alarmes dues notamment à des modifications de végétation peuvent être localisées.

Sur la première classification, les ombres, de manière générale, sont toutes détectées par l'algorithme de *k-moyennes* entropiques comme appartenant à la même classe. L'information radiométrique apportée est, en effet, très forte au niveau des ombres; elle a été utilisée de manière beaucoup plus systématique dans [Watanabe et al., 1998] pour discriminer les ombres des changements réels dans des scènes urbaines présentant des immeubles de taille considérable (villes très denses telles que Tokyo). Toutefois, la classe dans laquelle résident les ombres appartenant à la seconde image est aussi occupée par de la végétation. Sur la seconde classification, les ombres et la végétation semblent être mieux discriminées, mais certaines zones d'ombres sont manquantes dans la classe principale. Des problèmes similaires peuvent être observés sur les autres classes, montrant que, bien que la classification fondée uniquement sur la radiométrie soit porteuse de sens, elle n'est pas suffisante pour une interprétation satisfaisante de l'image.

Les résultats de la régularisation par champs de Markovs sont indiqués sur la Figure 3.5, on remarque que la sursegmentation (au niveau des ombres, notamment) a tendance à être plus réduite.

3.5.2 Aéroport d'Abidjan

L'algorithme des *k-moyennes* entropiques a également été testé sur le tarmac de la zone aéroportuaire d'Abidjan, cette fois-ci en utilisant le masque de changements obtenu par notre méthode fondée sur les lignes de niveau (cf. Chapitre 2). Les changements les plus importants sur ce couple d'images correspondent aux mouvements des avions et des véhicules de service (camions de ravitaillement, bus, camionnettes). Les images originales et le masque de changements obtenu sont présentés sur la Figure 3.7. Il est nécessaire de préciser que ce masque est obtenu en appliquant l'opérateur logique « OU » sur les deux masques de changements obtenus normalement par notre méthode de détection. Ce faisant, une partie de l'information a été perdue. Ceci est dû au fait que la méthode de classification a été développée chronologiquement avant la méthode de détection de masque de changements fondée sur les lignes de niveau. Il persiste donc un manque d'adaptation important pour pouvoir tirer parti de toutes les qualités de l'algorithme de détection de masque de changements.

Ici, des valeurs raisonnables des paramètres d'entropie fournissent un nombre de classes variant entre 4 et 8. les images correspondantes sont fournies respectivement Figure 3.8 et Figure 3.9.

Nous avons également fourni pour comparaison sur la Figure 3.10 le résultat de la classification et de la régularisation par champs de Markov sur le masque de changements utilisant l'analyse en composantes principales. On remarque que le fait d'avoir régularisé le masque de changements (voir chapitre 2) améliore le rendu de la classification, mais en contrepartie, on doit utiliser un masque qui ne détecte pratiquement pas une substitution

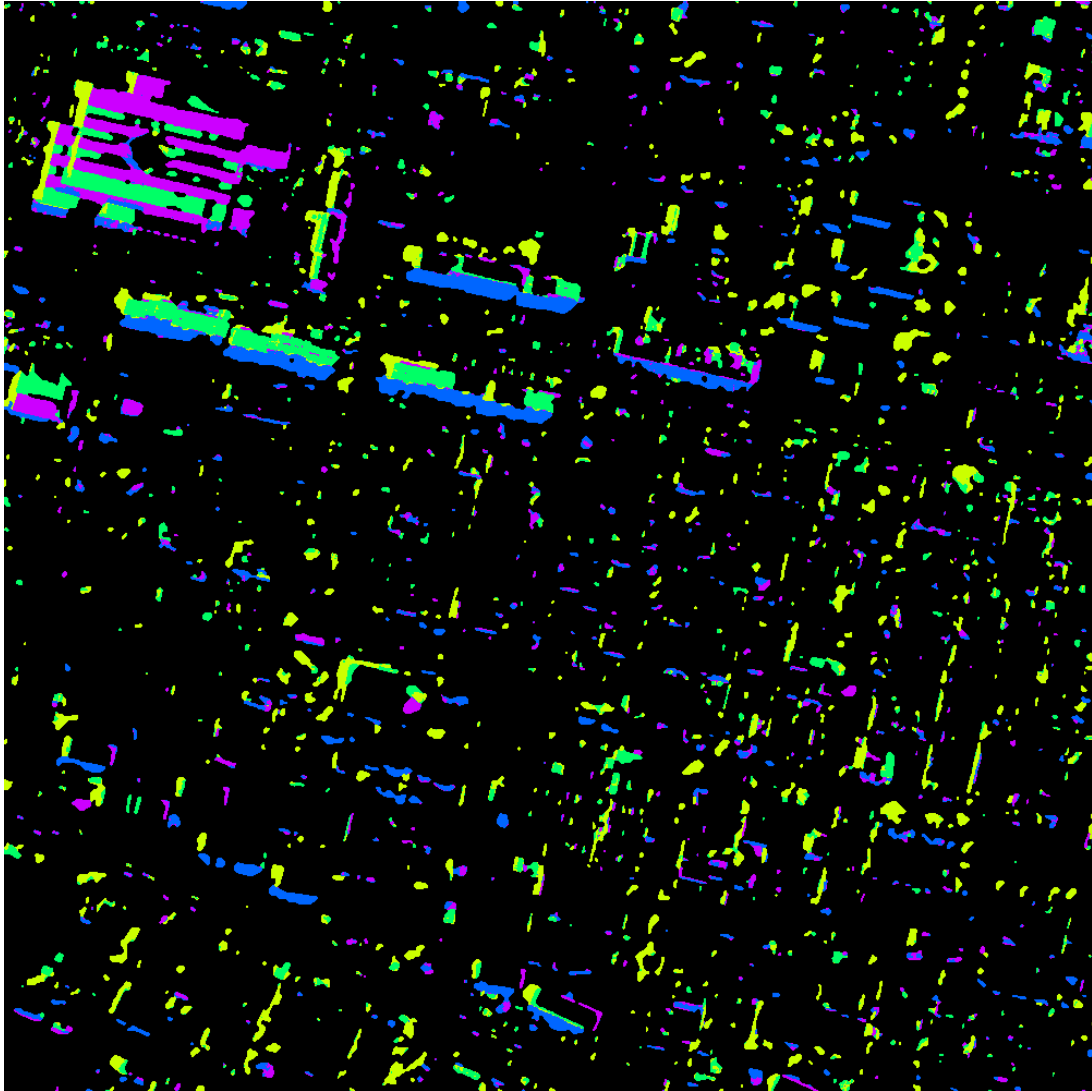


FIGURE 3.4 – Première classification par *k-moyennes*, ici quatre classes sont obtenues. On distingue notamment les ombres parmi les objets détectés.

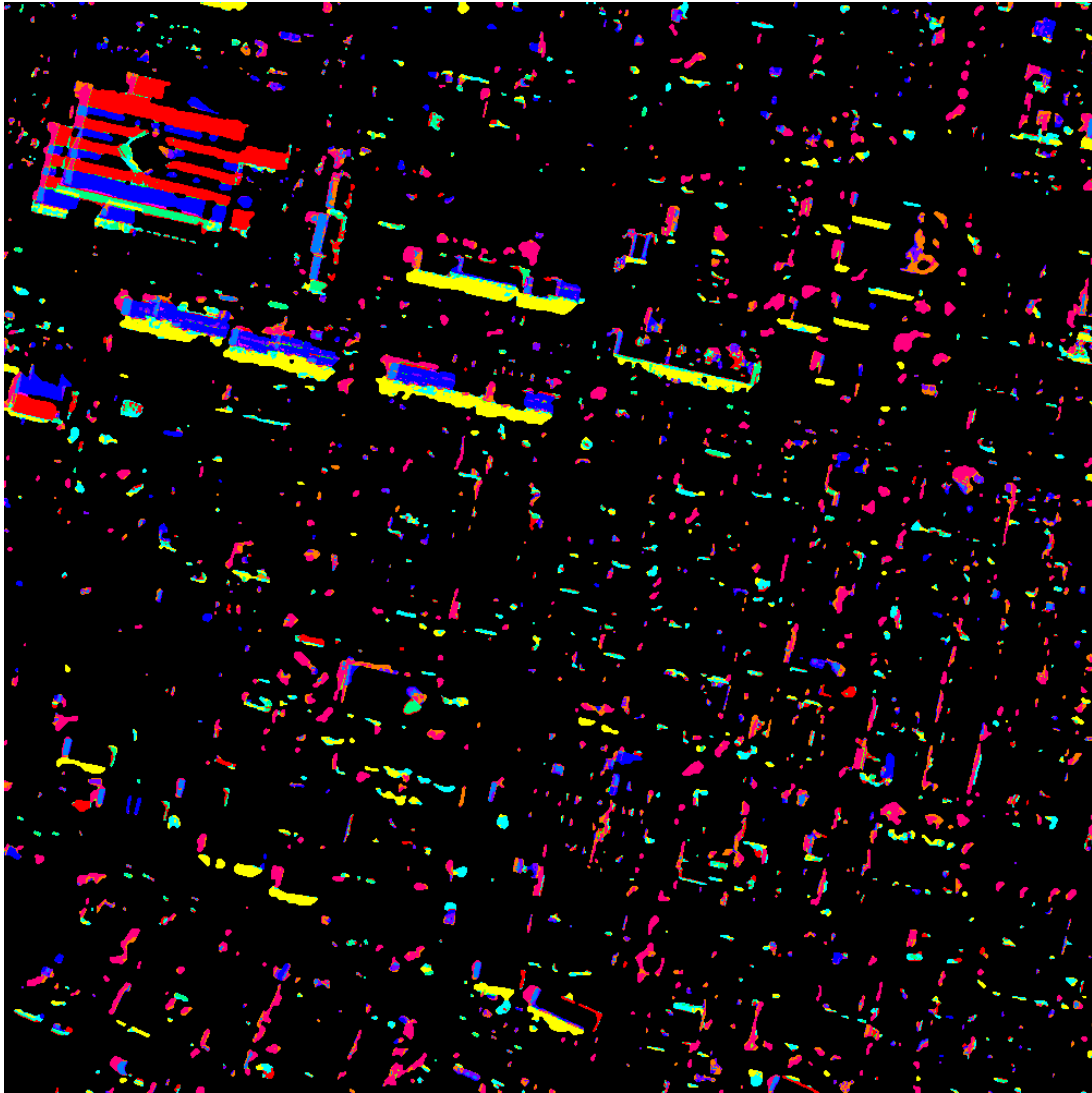


FIGURE 3.5 – Classification par k -moyennes entropiques (11 classes). On remarque que certaines classes (ombres, notamment) contiennent des pixels isolés.

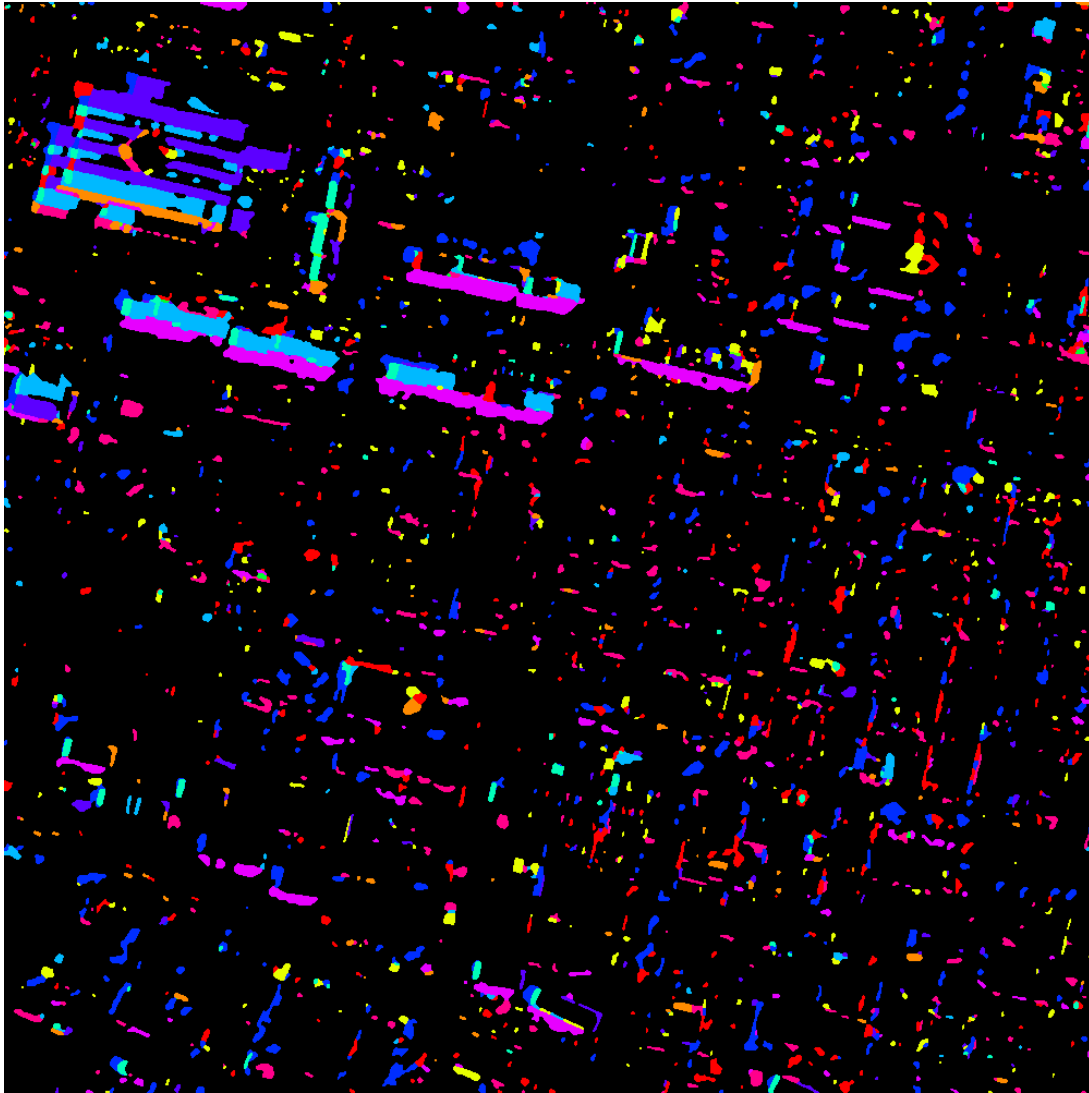


FIGURE 3.6 – Après régularisation par champs de Markov de la classification obtenue sur la Figure 3.5, on remarque que les classes (notamment les ombres) sont plus homogènes.

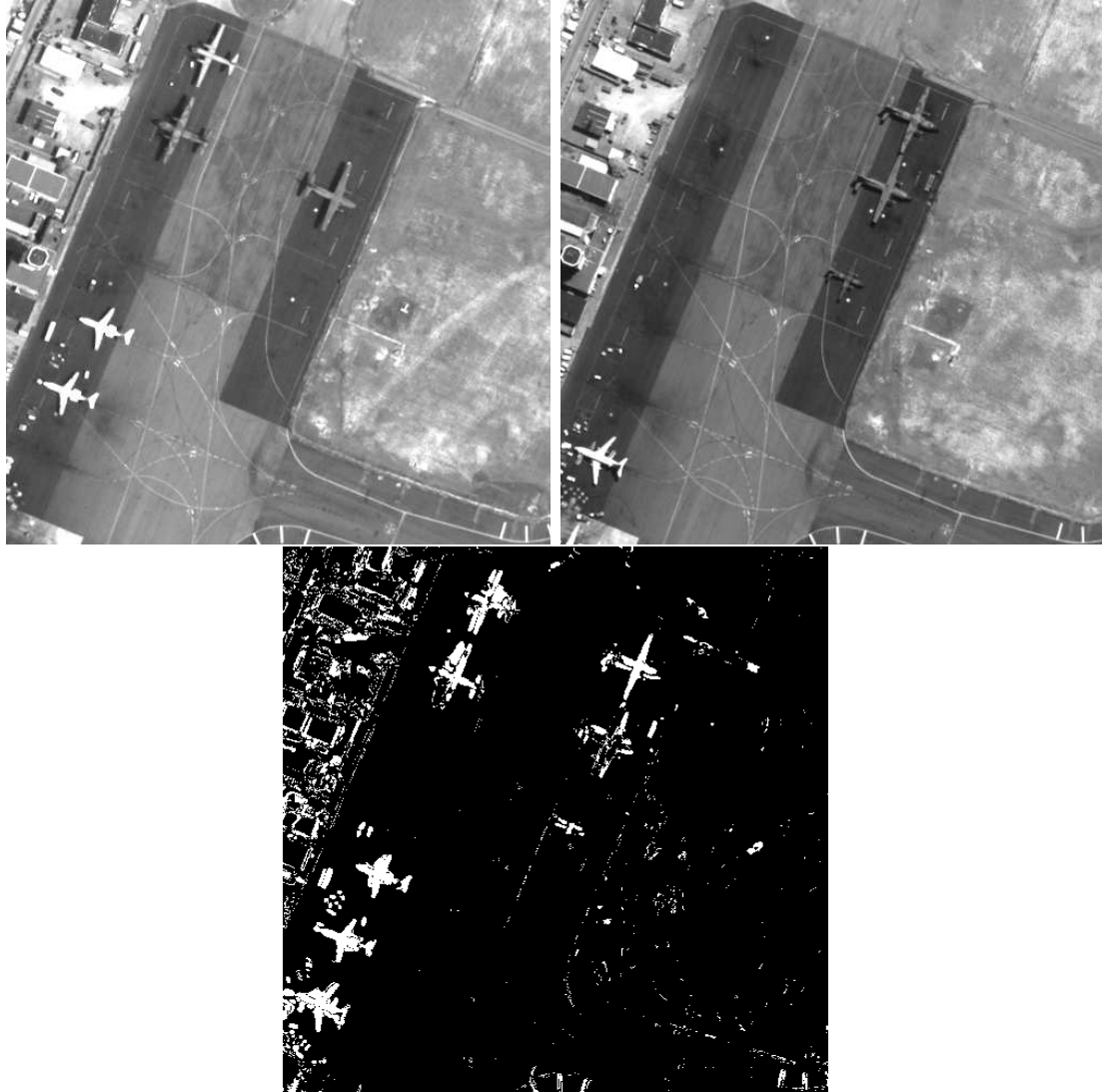


FIGURE 3.7 – Haut : Images originales de l'aéroport d'Abidjan, prises en février et juillet 2003 © Spot Image, fournies par la DGA. Bas : masque de changements issu de l'approche par lignes de niveau (voir Chapitre 2)

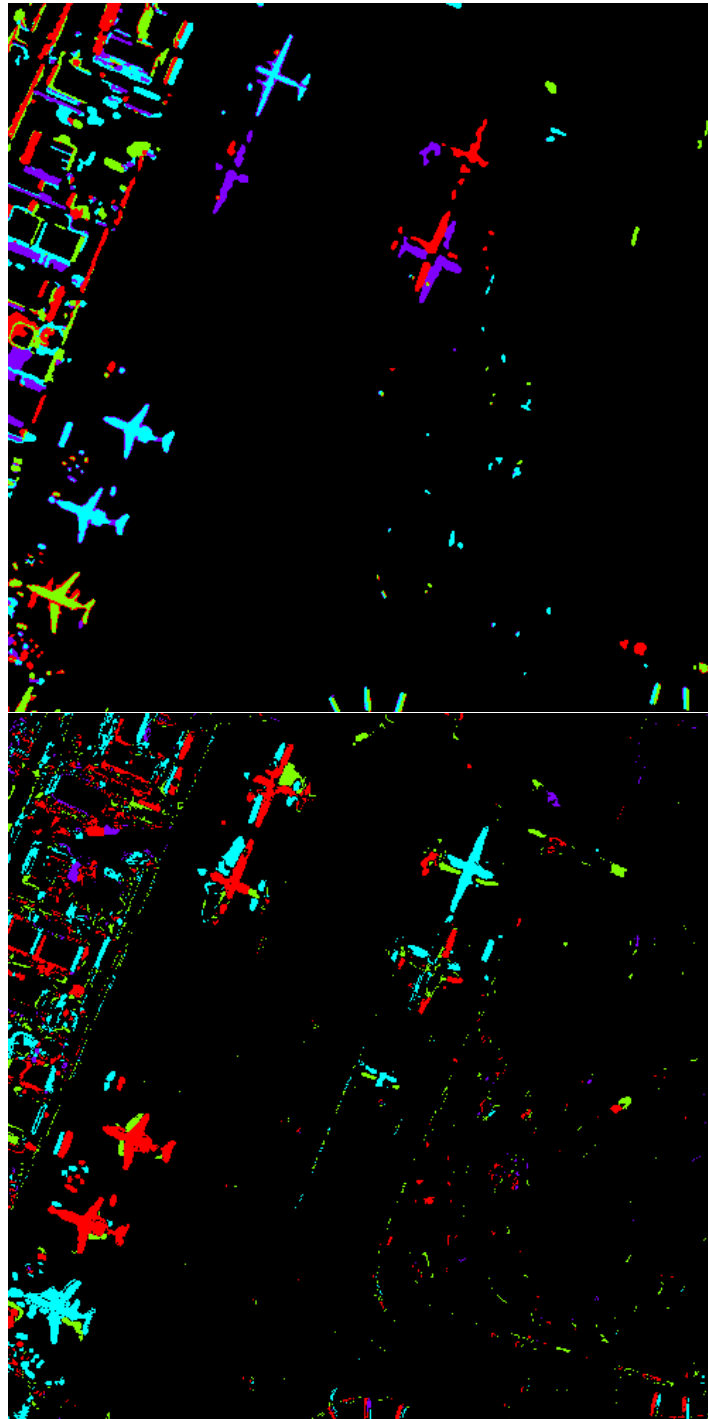


FIGURE 3.8 – Haut : classification par k -moyennes entropiques avec 4 classes : les avions figurant sur la première image originales (voir Figure 3.7) sont divisés en plusieurs classes selon leur couleur (blanc ou camouflage), les ombres sur la première image originale sont confondues avec les avions apparaissant sur la deuxième image originale. Bas : Régularisation par champs de Markov de la classification des changements sur l'aéroport d'Abidjan : on remarque que les avions sont détectés plus uniformément. On peut plus facilement discriminer les apparitions et les disparitions. En revanche, les fausses alarmes liées aux ombres restent problématiques.

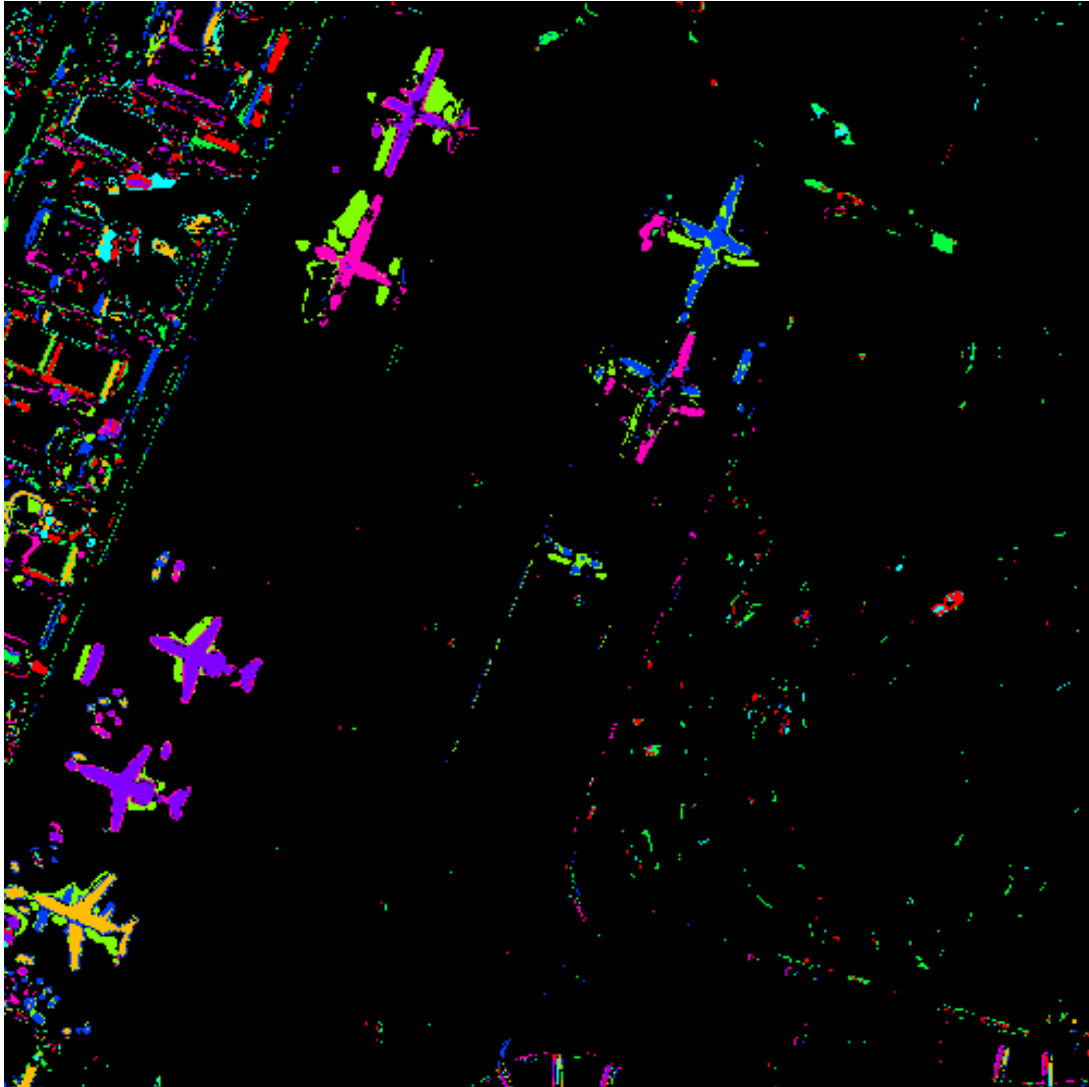


FIGURE 3.9 – Classification avec 8 classes : les avions figurant sur la première image originale (voir Figure 3.7) sont divisés en plusieurs classes selon leur couleur (blanc ou camouflage), les ombres sur la première image originale sont confondues avec les avions apparaissant sur la deuxième image originale.

d'un avion.

Pour avoir une comparaison plus juste de ces deux résultats, une régularisation des valeurs détectées par notre algorithme par lignes de niveau devrait être effectuée au préalable.

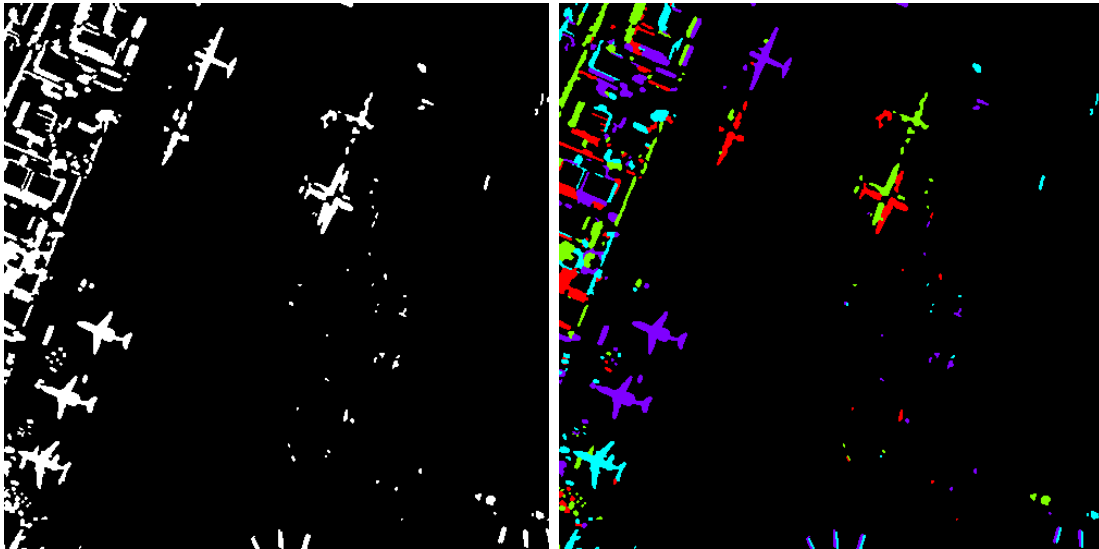


FIGURE 3.10 – Gauche : Masque de changements obtenu en appliquant la méthode de R. Wiemker et résultat après application de notre classifieur et la régularisation par champs de Markov

3.6 Conclusion

Dans ce chapitre, nous avons tenté de tirer parti des informations radiométriques afin de classifier les zones changées. Nous avons pris le parti d'utiliser une méthodologie assez simple (*i.e.* adapter l'algorithme des *k-moyennes* à la métrique de notre problème). Cette méthodologie est donc largement perfectible et d'autres algorithmes plus complexes (mélanges de gaussiennes ...) pourraient donner de meilleurs résultats.

Un point important, qui a montré des limitations dans l'exemple est l'estimation du nombre de classes. Si l'approche « entropique » donne des résultats raisonnablement bons quant au nombre de classes, il semble difficile de déterminer ce nombre avec précision. Une approche fondée sur un estimateur plus robuste (peut-être semblable à la notion de granularité évoquée précédemment), ou l'introduction d'*a priori*, dans la mesure du possible, sont autant de pistes restant à explorer.

Les résultats sont assez satisfaisants. Dans certains cas, les classes obtenues par cet algorithme englobent des « objets » réels appartenant à la même catégorie (bâtiments,

véhicules, ombres). Malheureusement, quel que soit le nombre de classes évalué, on remarque des imperfections dans la classification (sursegmentation ou, au contraire, non-discrimination). Ces défauts sont en partie « gommés » par une régularisation par champs de Markov, c'est à dire en introduisant de l'information spatiale dans le modèle, mais cela n'est pas toujours suffisant. Cette constatation, combinée à l'observation que certaines formes et certaines directions sont récurrentes dans les scènes urbaines, nous a motivés à tenter de caractériser les zones connexes détectées à l'aide de polygones, puis de tenter d'incorporer les informations qu'ils apportent dans la classification. Ce travail est présenté dans le chapitre suivant.

Chapitre 4

Approche objet

Dans le chapitre précédent, une première classification des zones détectées comme changées, fondée essentiellement sur la radiométrie, a été proposée. Si les résultats sont intéressants, on remarque qu'elle ne suffit pas, loin s'en faut, pour proposer une description précise des changements (et d'identifier les objets sous-jacents). Des informations importantes, notamment spatiales, résident dans les zones détectées et ne sont pas prises en compte par la classification au niveau pixelique décrite précédemment. Parmi les informations spatiales, on pense souvent aux informations de taille qui permettent de discerner facilement un véhicule d'une ombre portée (souvent appartenant à la même classe lors de l'évaluation précédente). Mais les informations d'orientation sont plus rarement mises en avant.

Ce chapitre propose une méthode pour exploiter ces informations en deux phases. Dans un premier temps, nous proposons un algorithme d'approximation polygonale fondée sur une mesure originale de l'erreur, dans le but d'être robuste aux valeurs aberrantes présentes dans les formes détectées. Dans un second temps, nous proposons une méthode de classification, fondée sur les *k-moyennes*. Tout d'abord en tenant compte uniquement des informations directionnelles, puis en pondérant les informations radiométriques et directionnelles.

4.1 Approximation polygonale

L'approximation polygonale consiste à approximer une courbe ouverte ou fermée à l'aide d'un polygone. Il s'agit d'un problème classique et la littérature décrivant les diverses approches est abondante. Parmi les publications intéressantes, on peut citer notamment [Kolesnikov, 2003] qui présente une revue assez complète des différents problèmes rencontrés au sein de l'approximation polygonale.

Dans un premier temps, les principes mathématiques liés à l'approximation polygonale ainsi que les différentes formalisations de ce problème sont rappelés ci-dessous, puis nous

présentons une mesure de l'erreur d'approximation adaptée aux zones à approximer. Enfin, on décrit un algorithme de descente de gradient minimisant cette erreur.

4.1.1 Principes mathématiques

Polygones

On note un polygone P comme la suite de ses sommets $(P_k)_{1 \leq k \leq n+1}$, avec $P_{n+1} = P_0$.

Mesure d'erreur

Une des composantes fondamentales de l'approximation polygonale est la mesure d'erreur. Cette mesure permet d'évaluer la qualité d'une approximation, et donc de comparer quantitativement plusieurs approximations entre elles. On peut la définir formellement de la façon suivante :

Définition 8 (Mesure d'erreur). *Soit \mathcal{C} l'espace des courbes de longueur finie de \mathbb{R}^2 . Une mesure d'erreur d'approximation est une application $f : \mathcal{C} \times \mathcal{C} \rightarrow \mathbb{R}^+$, telle que $f(C, C) = 0$.*

Cette définition est relativement large. Elle peut être adaptée à des espaces plus restreints en prenant $f : \mathcal{C} \times \mathcal{P} \rightarrow \mathbb{R}^+$, avec \mathcal{P} l'espace des courbes polygonales finies. Dans notre cas, nous nous attachons à des courbes fermées, nous prendrons alors comme définition pour une mesure d'erreur :

Définition 9 (Mesure d'erreur (bis)). *Soit \mathcal{C} l'espace des courbes fermées de longueur finie de \mathbb{R}^2 et \mathcal{P} l'ensemble des polygones de \mathbb{R}^2 . Une mesure d'erreur d'approximation est une application $f : \mathcal{C} \times \mathcal{P} \rightarrow \mathbb{R}^+$, telle que $f(C, C) = 0$.*

Le choix d'une mesure d'erreur pertinente est évidemment crucial pour l'approximation polygonale. Parmi les mesures d'erreur classiques, on remarque les erreurs en norme L_p lorsque C est une courbe de classe C^1 . On note $d : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^+$ la distance euclidienne dans \mathbb{R}^2 . Les erreurs en norme L_p s'écrivent alors sous la forme :

$$E_{L_p}(C, P) = \oint_C d(x, P)^p dx \text{ avec } d(x, P) = \min_{t \in P} d(x, t) \quad (4.1)$$

[Kolesnikov, 2003] remarque que, quand la courbe à approximer est elle-même un polygone, on utilise assez souvent une forme simplifiée :

$$E_{L_p}(C, P) = \sum_{k=1}^n d(C_k, P)^p \text{ avec } d(C_k, P) = \min_{t \in P} d(C_k, t) \quad (4.2)$$

4.1.2 Énoncés des problèmes d'approximation polygonale

En réalité, le terme d'« approximation polygonale » regroupe deux problèmes mathématiques distincts. Pour ces deux problèmes, on note pour $n \in \mathbb{N}$, \mathcal{P}_n l'ensemble de polygones à n côtés.

Définition 10 (Problème Min- ε). *Soit C une courbe fermée de \mathbb{R}^2 et D une mesure d'erreur. Soit $n \in \mathbb{N}^*$. Trouver $P \in \mathcal{P}_n$ tel que :*

$$P = \arg \min_{P' \in \mathcal{P}_n} D(C, P') \quad (4.3)$$

Il s'agit donc, à nombre de côtés fixés, de trouver le polygone qui minimise l'erreur d'approximation.

Le second problème est plus délicat :

Définition 11 (Problème Min- \sharp). *soit C une courbe fermée de \mathbb{R}^2 et D une mesure d'erreur. Soit $\varepsilon \in \mathbb{R}^{+*}$. Trouver $n \in \mathbb{N}$ tel que :*

$$n = \min_{\mathbb{N}} \{n', \min_{P \in \mathcal{P}_{n'}} D(C, P) \leq \varepsilon\} \quad (4.4)$$

Il s'agit ici, de trouver le nombre minimal de côtés nécessaires pour atteindre une précision d'approximation ε donnée.

Dans ce qui suivra, nous allons tenter de reconnaître et comparer des quadrilatères. Le problème à résoudre sera donc celui des min- ε .

4.1.3 Choix de l'approximation polygonale

Pour le problème particulier de l'approximation d'objets, il faut dans un premier temps choisir une norme pertinente pour l'approximation désirée. Dans notre cas particulier, les formes détectées sont souvent assez peu précises et des valeurs aberrantes sont parfois présentes. Il convient donc de proposer une erreur d'approximation qui soit robuste à ces valeurs aberrantes. Quelques essais montrent que les normes utilisées habituellement sont assez sensibles à ce type d'irrégularités. Nous proposons ici de tirer parti du fait que l'on cherche à approximer des formes et non pas des contours afin de proposer une erreur d'approximation originale.

4.1.4 Formulation Mathématique

on note respectivement $\overset{\circ}{C}$ et $\overset{\circ}{P}$ l'intérieur (au sens géométrique) de C et de P .

Définition 12 (Ou exclusif). *Soient E_1 et E_2 deux sous ensembles de \mathbb{R}^2 . On définit l'opérateur « ou exclusif » $\text{xor}(E_1, E_2)$ comme :*

$$\text{xor}(E_1, E_2) = (E_1 \cap \neg E_2) \cup (\neg E_1 \cap E_2) \quad (4.5)$$

où \neg représente la négation logique.

L'erreur d'approximation est fondée sur l'opérateur « ou exclusif ». On l'écrit sous la forme :

$$E_{\text{xor}}(C, P) = \mu(\text{xor}(\mathring{C}, \mathring{P})) \quad (4.6)$$

où μ est la mesure de Lebesgue dans \mathbb{R}^2 . Une illustration de cette mesure est donnée Figure 4.1. L'intérêt de cette erreur est qu'elle est intrinsèquement liée à la notion d'aire et non plus de distance. Elle offre donc une plus grande robustesse aux « valeurs aberrantes » qui engendrent des irrégularités dont l'aire assez faible. Ceci est illustré sur la Figure 4.2.

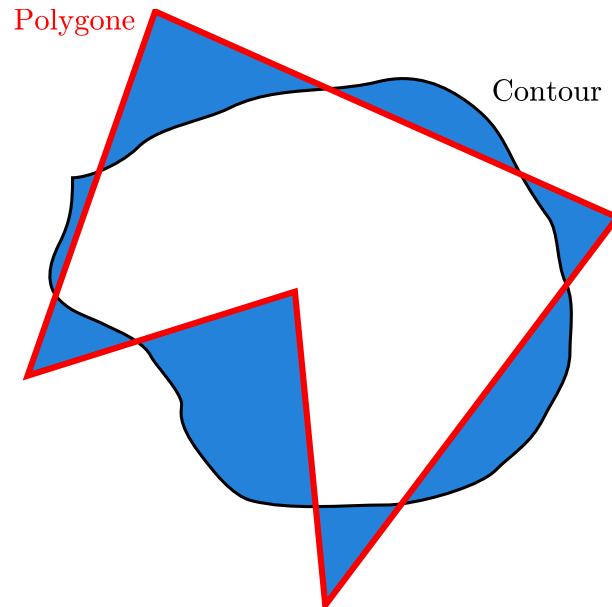


FIGURE 4.1 – Illustration de l'erreur en norme xor

4.1.5 Algorithme d'optimisation locale

Afin de minimiser l'erreur définie dans la Partie 4.1.3, on a recours à l'algorithme classique de descente de gradient. On fait l'hypothèse que l'énergie à minimiser est convexe. Il est assez facile de vérifier que l'énergie que l'on cherche à minimiser n'est pas convexe. La convergence de l'algorithme vers une approximation pertinente dépend donc fortement des conditions d'initialisation. Dans la Partie 4.1.6, nous présentons une heuristique donnant des résultats raisonnables pour l'initialisation.

Hypothèses et notations

Pour le calcul du gradient, l'espace est constitué par les m points P_i du polygone P . On calcule donc l'influence sur l'erreur du déplacement d'un point P_i . Cette influence est

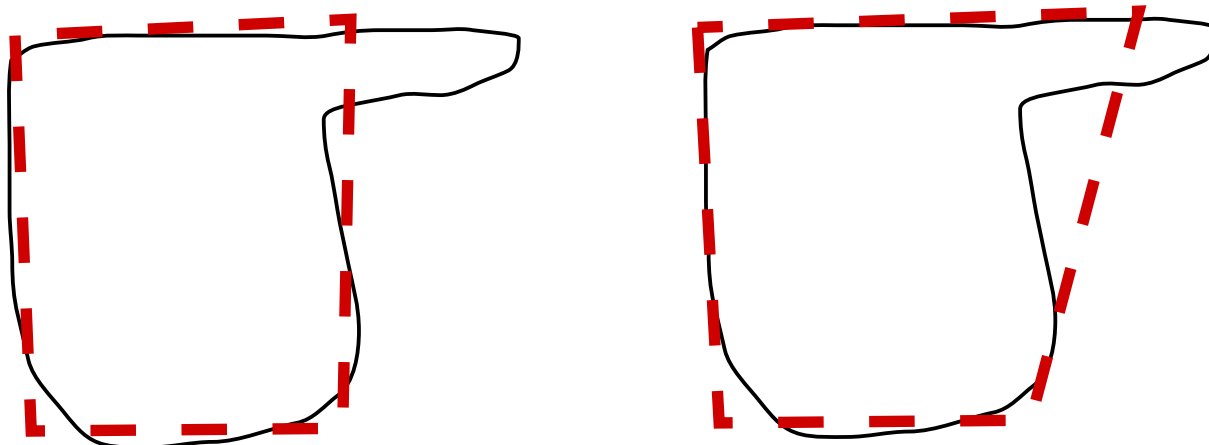


FIGURE 4.2 – Approximations polygonales d’une forme présentant des points aberrants. Gauche : approximation optimale avec notre mesure d’erreur, droite : approximation optimales avec une mesure d’erreur « classique ».

illustrée sur la Figure 4.3.

Afin de simplifier le calcul, on admet quelques hypothèses supplémentaires :

- On suppose, d’une part, le polygone P *non-croisé*,
- On suppose que P a uniquement des intersections « strictes » avec C (en particulier, on suppose l’absence de points de tangence et de segments sur lesquels P et C sont confondus).

On ordonne les P_i dans le sens direct (anti-horaire). Pour $1 \leq i < n$, On note alors S_i , le côté $[P_i, P_{i+1}]$ et on note S_m le côté $[P_n, P_0]$.

Les intersections entre S_i et C sont appelées $(I_{i,j})_{1 \leq j \leq p_i}$, et classées dans l’ordre croissant selon la distance $d(P_i, I_{i,j})$.

Calcul du gradient

La variation dE_i de l’erreur d’approximation due au mouvement de P_i se traduit uniquement à travers les mouvements des segments S_{i-1} et S_i (S_n et S_1 si $i = 1$). Si le mouvement de P_i est élémentaire, on remarque aisément que les variations de E_{xor} dues à ces derniers se traduisent au premier ordre par deux termes indépendants. On appelle dE_i^- le terme lié au mouvement du segment S_{i-1} et dE_i^+ le terme lié au mouvement du segment S_i .

On se concentre dans un premier temps sur le calcul de dE_i^+ . On suppose que P_i est à l’extérieur de \hat{C} et que l’on déplace P_i orthogonalement à S_i et vers l’extérieur de P (bouger P_i parallèlement à S_i n’a pas d’influence au premier ordre sur dE_i^+).

En observant la Figure 4.3, on constate que ce mouvement déplace les $I_{i,j}$ en $I'_{i,j}$. Les

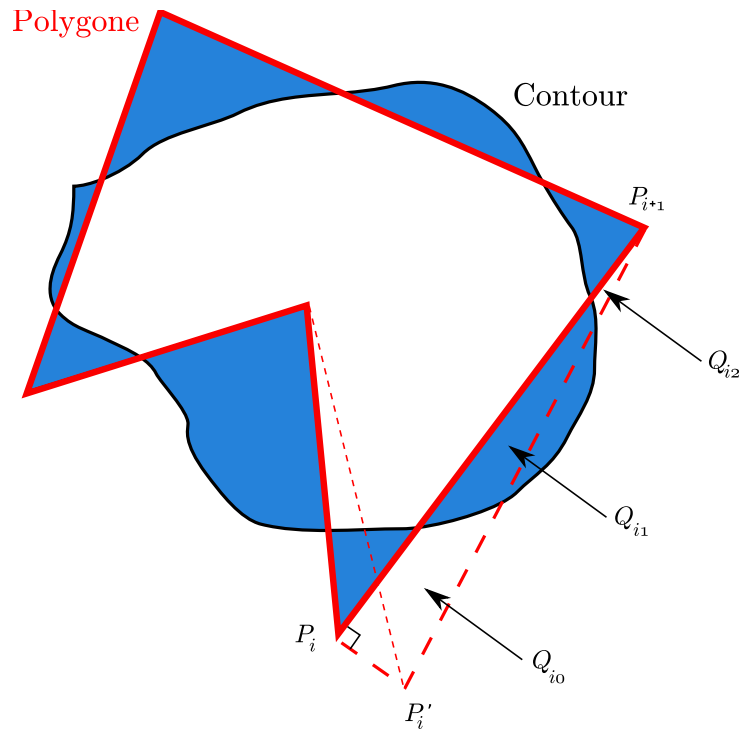


FIGURE 4.3 – Influence du déplacement d'un point P_i sur l'erreur d'approximation polygonale. Les aires successives des zones Q_{i0} à Q_{in} sont alternativement ajoutées et retranchées à l'erreur d'approximation.

aires $A_{i,j}$ des zones $Q_{i,j}$ balayées par ce mouvement doivent être successivement ajoutées et retranchées afin de calculer la variation de E_{xor} . Lorsque le déplacement est élémentaire, on remarque facilement que ces zones sont des trapèzes (à l'exception de $Q_{i,p+1}$ qui est un triangle). Le calcul de la variation de l'erreur est alors assez simple :

$$dE_i^+ = - \sum_{j=0}^p (-1)^j A_{i,j} \quad (4.7)$$

Supposons maintenant P_i à l'intérieur de C , on remarque alors que l'influence de chaque $Q_{i,j}$ sur E_{xor} est inversée vis-à-vis de la situation précédente. L'expression de dE_i^+ est alors :

$$dE_i^+ = \sum_{j=0}^p (-1)^j A_{i,j} \quad (4.8)$$

On pose alors γ tel que :

$$\begin{cases} \gamma(P_i) = -1 & \text{si } P_i \in \mathring{C} \\ \gamma(P_i) = 1 & \text{si } P_i \notin \mathring{C} \end{cases} \quad (4.9)$$

L'expression de dE_i^+ devient :

$$dE_i^+ = \sum_{j=0}^{p_i} (-1)^j A_{i,j} \quad (4.10)$$

On calcule maintenant l'aire de $A_{i,j}$. On pose :

$$L_i = d(P_i, P_{i+1}) \text{ pour } 1 \leq i \leq n \quad (4.11)$$

la longueur du segment S_i , et

$$x_{i,j} = \frac{d(P_i, I_{i,j})}{L_i} \quad (4.12)$$

l'abscisse relative du point courant sur le segment S_i . On définit également (u_i, v_i) le repère orthogonal direct centré en P_i tel que u_i et $P_i P_{i+1}$ soient parallèles et $\langle u_i, P_i P_{i+1} \rangle > 0$. On note x_i, y_i les coordonnées d'un point dans ce repère.

L'expression de $A_{i,j}$ en fonction de dy_i pour $2 < i < p_i$ est :

$$A_{i,j} = \frac{L_i dy_i}{2} \left((1 - x_{i,j})^2 - (1 - x_{i,j+1})^2 \right) \quad (4.13)$$

Les expressions de $A_{i,1}$ et A_{i,p_i} se calculent de la même manière :

$$A_{i,1} = \frac{L_i dy_i}{2} \left(1 - (1 - x_1)^2 \right) \quad (4.14)$$

$$A_{i,p} = \frac{L_i dy_i}{2} (1 - x_{i,p_i})^2 \quad (4.15)$$

L'expression de dE_i^+ s'écrit alors :

$$dE_i^+ = \frac{\gamma(P_i)L_i dy_i}{2} \quad (4.16)$$

$$\cdot \left(1 + 2 \sum_{j=1}^{p_i-1} [(-1)^j (1 - x_{i,j})^2] + (-1)^{p_i} \cdot (1 - x_{i,p})^2 \right) + o(dy_i)$$

$$= \frac{\gamma(P_i)L_i}{2} \langle dP_i, u_i \rangle \quad (4.17)$$

$$\cdot \left(1 + 2 \sum_{j=1}^{p_i-1} [(-1)^j (1 - x_{i,j})^2] + (-1)^{p_i} \cdot (1 - x_{i,p})^2 \right) + o(|dP_i|)$$

De la même manière, on peut calculer dE_i^- :

$$dE_i^- = \frac{\gamma(P_i)L_{i-1}}{2} \langle dP_{i-1}, v_{i-1} \rangle \cdot \left(1 + 2 \sum_{j=1}^{p_i} [(-1)^{p_i+1-j} x_{i-1,j}^2] \right) + o(|dP_i|) \quad (4.18)$$

L'expression du gradient de E_{xor} s'écrit :

$$\frac{dE_{\text{xor}}}{dP_i} = \frac{\gamma(P_i)}{2} \cdot \left[L_{i-1} \left(1 + 2 \sum_{j=1}^{p_i-1} [(-1)^{p_i+1-j} x_{i-1,j}^2] \right) v_{i-1} + L_i \left(1 + 2 \sum_{j=1}^{p_i-1} [(-1)^j (1 - x_{i,j})^2] + (-1)^{p_i} \cdot (1 - x_{i,p})^2 \right) v_i \right] \quad (4.19)$$

Les P_i étant indépendants dans l'espace d'état, nous avons alors l'expression du gradient. Une descente de gradient classique permet d'atteindre un minimum de l'erreur d'approximation E_{xor} (une bonne approximation de la courbe avec le polygone P).

4.1.6 Initialisation

L'algorithme de descente de gradient est, par nature, un algorithme d'optimisation locale. Il est donc particulièrement sensible à l'initialisation. Dans le cas de l'approximation polygonale, avec la mesure d'erreur présentée dans la Partie 4.1.3, la plupart des minima locaux correspondent à un polygone P réduit à un seul point – pour s'en assurer, on pourra prendre l'exemple d'un polygone initialisé de telle façon que $\overset{\circ}{C} \cap \overset{\circ}{P} = \emptyset$ (P complètement à l'extérieur de C) – ce qui correspond à une situation à éviter. D'autres minima locaux peuvent être atteints lorsque la forme du contour C est trop irrégulière (nombre de points du polygone sous évalué).

Dans le cadre plus général de l'approximation polygonale, et dans le cas où la courbe à approximer est elle-même polygonale, les techniques classiques de résolution ne font pas appel à une descente de gradient, mais plutôt à de la programmation dynamique. Ces techniques ont permis d'obtenir des optima globaux [Perez et Vidal, 1994]. Dans le cas de courbes fermées, trouver dans ces conditions un point d'initialisation pour l'algorithme est particulièrement coûteux, cependant une solution permettant de s'approcher de la solution optimale et ayant un coût algorithmique relativement raisonnable est proposée dans [Kolesnikov et Fränti, 2007].

Dans cette partie, nous proposons une heuristique pour l'initialisation du polygone. Le but est de tenter d'uniformiser la répartition des points initiaux du polygone sur le contour : On choisit comme premier point du polygone P_1 , le point du contour le plus éloigné du centre de gravité O . Ensuite, le point le plus éloigné du segment $[O, P_1]$ est choisi pour P_2 , et le point le plus éloigné du triangle (O, P_1, P_2) est choisi pour P_3 . À partir de P_4 , on choisit le point le plus éloigné du polygone (P_0, \dots, P_{i-1}) comme point P_i . Un exemple d'initialisation avec cette heuristique est représenté sur la Figure 4.4. Un des problèmes importants de cet algorithme est la phase de réordonnancement des points. En effet, la solution pour réordonner des points de façon à former un polygone non-croisé et direct n'est pas unique. Pour remédier à ce problème, on propose de paramétrer le contour C et de réordonner les points du polygone par rapport au paramètre sur le contour.

Algorithme 4.1 : Algorithme d'initialisation du polygone P approximant le contour C

Données : C une courbe C_1 , fermée

Résultat : P le polygone initial approximant C

début

 Chercher O , le centre de gravité de \hat{C} ;

$P_1 \leftarrow \arg \max_{x \in C} d(O, x)$;

$P_2 \leftarrow \arg \max_{x \in C} d([O, P_1], x)$;

$P_3 \leftarrow \arg \max_{x \in C} d((O, P_1, P_2), x)$;

pour chaque $i \in [4..n]$ **faire**

$P_i \leftarrow \arg \max_{x \in C} d((P_1, \dots, P_{i-1}), x)$;

 Réordonner les P_i de façon à obtenir un polygone non-croisé;

fin

fin

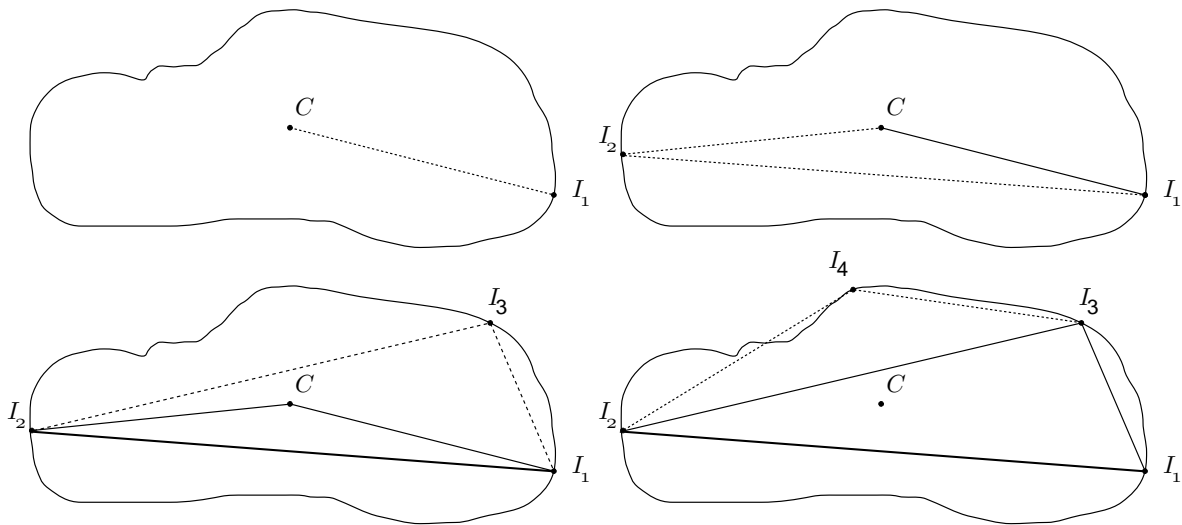


FIGURE 4.4 – Initialisation

4.1.7 Résultats expérimentaux

Exemples jouets

L'algorithme d'approximation polygonale a d'abord été testé sur un « exemple jouet ». Sur cet exemple, le contour à approximer est un carré qui présente deux valeurs aberrantes prononcées. L'objectif est ici de vérifier la pertinence de la mesure utilisée et la convergence de l'algorithme d'optimisation. On notera, par exemple, que le minimum en norme L_2 va être influencé par la présence des valeurs aberrantes. Les résultats expérimentaux montrent (cf. Figure 4.5) que l'algorithme a bien le comportement attendu. L'heuristique d'initialisation place certains points sur les valeurs aberrantes, toutefois, après quelques itérations, la forme principale est atteinte par le polygone.

Un autre exemple a été utilisé, cette fois-ci avec une forme plus complexe : un avion a été extrait sur les images de l'aéroport d'Abidjan. L'algorithme est testé pour des polygones à 9 et à 11 côtés successivement. Les résultats sont présentés sur la Figure 4.6. On remarque, dans les deux cas, que l'approximation polygonale converge vers un résultat pertinent. Néanmoins, dans le cas du polygone à neuf côtés, on observe que l'empennage n'est pas compris dans l'intérieur du polygone d'approximation. Dans le cas du polygone à 11 côtés, en revanche, l'empennage est correctement détecté.

Après avoir vérifié son comportement sur ces « exemples jouets », l'algorithme d'approximation polygonale est alors utilisé afin de qualifier les géométries des zones classées par radiométrie par les algorithmes présentés dans le Chapitre 3. L'objectif était, avant tout, de retrouver les orientations principales des zones détectées (en particulier les fausses

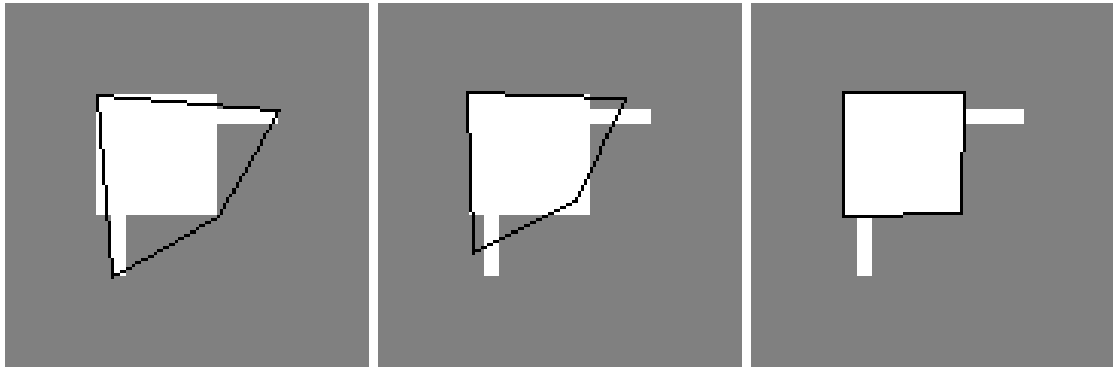


FIGURE 4.5 – Approximation polygonale sur un « exemple jouet ». Le quadrilatère finit par converger sur le carré après quelques itérations.

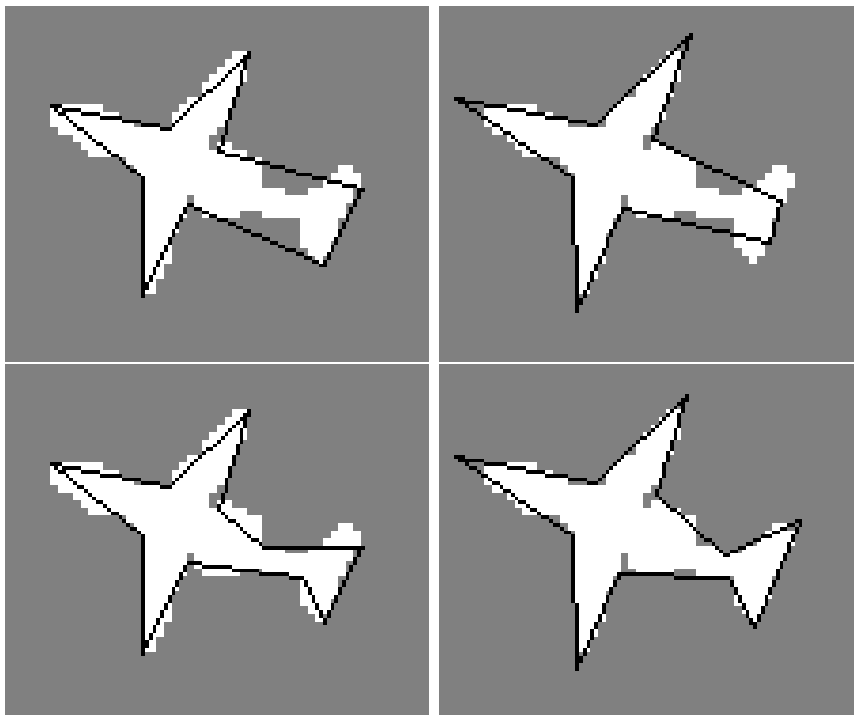


FIGURE 4.6 – Test sur une forme d’avion extraite sur l’aéroport d’Abidjan pour un $n = 9$ (haut) et $n = 11$ (bas). À gauche : initialisation, à droite : convergence.

alarmes que constituent les zones d'ombre). Afin de rendre les résultats plus lisibles d'une part, et de travailler sur des zones pour lesquelles une forme est franchement caractérisée, les zones dont la taille dépasse quelques pixels ont été analysées à ce stade. Un détail des zones classifiées sur la ville d'Abidjan est donné sur la Figure 4.7 et les résultats sont représentés sur la Figure 4.8

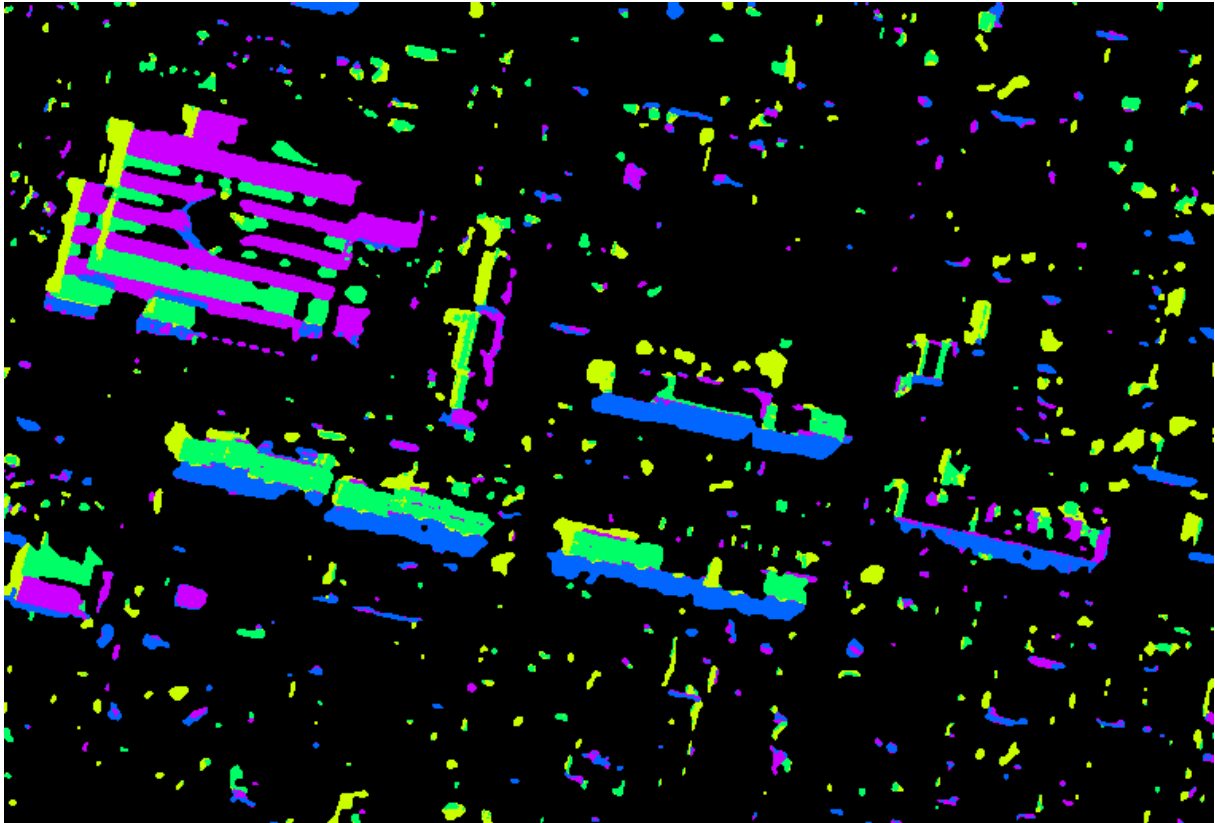


FIGURE 4.7 – Détail de la classification de la ville d'Abidjan obtenue au Chapitre 3.

4.2 Classification Polygonale

L'objectif de cette étude est de tirer parti des informations géométriques afin d'affiner la classification. En particulier, les orientations des polygones qui approximent les zones extraites par les méthodes présentées dans le Chapitre 2 peuvent être porteuses d'informations. Ci-dessous, on décrit une adaptation de l'algorithme des *k-moyennes* entropique, à l'espace des orientations d'un polygone. Pour cela on utilise une métrique issue des statistiques directionnelles.

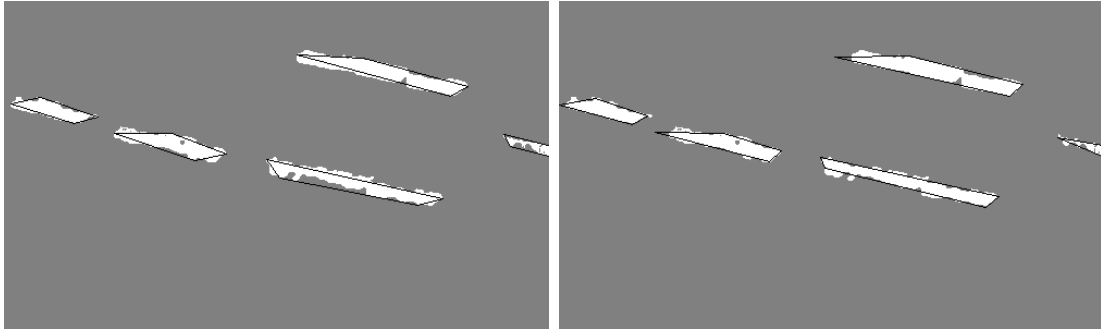


FIGURE 4.8 – Approximation polygonale sur la classification (Figure 4.7) : Gauche : initialisation, Droite : convergence. On remarque qu’après convergence, les directions des ombres et des bâtiments sont extraites correctement.

4.2.1 Métrique de l’espace des orientations

Les espaces des orientations sont des espaces assez particuliers. La distance euclidienne n’y correspond à aucune notion intuitive et n’est pas utilisable en l’état. La notion la plus intuitive de distance est, en général, définie de cette façon en deux dimensions :

Définition 13 (Distance angulaire). *Soient V_1 et V_2 deux éléments de \mathbb{R}^2 , θ_1 et θ_2 leurs arguments respectifs. La distance angulaire $d_{ang} : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ est alors définie comme :*

$$d(\theta_1, \theta_2) = \inf(|\theta_1 - \theta_2|, 2\pi - |\theta_1 - \theta_2|) \quad (4.20)$$

Cette définition peut être étendue à de plus grandes dimensions, simplement en considérant le plan contenant les deux vecteurs dont on considère l’orientation et en appliquant la définition pour l’espace \mathbb{R}^2 .

Néanmoins, cette distance présente le défaut d’avoir une moyenne de Karcher induite assez difficile à calculer. Les mathématiciens et statisticiens travaillant dans les espaces directionnels ont l’habitude de travailler sur la distance cordale :

Définition 14 (Distance cordale). *Soient V_1 et V_2 deux vecteurs de norme 1 de \mathbb{R}^n . La distance cordale de $\mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ est alors donnée par :*

$$d_{cord}(V_1, V_2) = \|V_1 - V_2\|_1 \quad (4.21)$$

Cette distance est assez proche de la distance angulaire. Par exemple, on remarque qu’elle est maximale lorsque $V_1 = -V_2$. La moyenne de Karcher associée se calcule aisément [Mardia et al., 1979; Mardia et Jupp., 2000] :

$$\mu_{Karcher}(V) = \overline{\sum_{i=1}^n V_i} \quad (4.22)$$

où \bar{x} est le vecteur x normé à 1. Bien que la moyenne soit facilement calculable, on remarque que si la somme des V_i est nulle, la moyenne de Karcher, telle qu'elle est décrite dans la formule, est indéterminée. En réalité, la moyenne de Karcher existe, mais n'est pas unique. Ainsi, la moyenne de Karcher de deux vecteurs unitaires opposés est l'intersection de l'hyperplan perpendiculaire aux deux vecteurs et de l'hypersphère de rayon unité. Dans la pratique, nous travaillons sur des suites V de faible dispersion angulaire (*i.e.* $\max_{i,j} d_{\text{cord}}(V_i, V_j) < \sqrt{2}$, ce qui permet d'éviter les cas comme celui qui est donné en exemple).

Des illustrations de la distance cordale ainsi que de la moyenne qui lui est associée sont représentées sur la Figure 4.9.

Espace d'orientations multiples

Si l'on se place dans \mathcal{P}_n , l'espace des polygones à n côtés. On représente un polygone par les n orientations de ses côtés. L'objectif est donc de classer les polygones de cet espace vis-à-vis de leurs orientations. On représente alors les polygones dans l'espace $[0, 2\pi]^n$. On utilise une combinaison de distances cordales afin de définir une distance entre les polygones : $d_{\text{xcord}} : \mathcal{P}_n \times \mathcal{P}_n \rightarrow \mathbb{R}^+$

$$d_{\text{xcord}}(P, P') = \sum_{i=1}^n d_{\text{cord}}(s_i, s'_i) \quad (4.23)$$

où s_i et s'_i sont respectivement les i -èmes segments des polygones P et P' .

4.2.2 Combinaison de la radiométrie et de la géométrie

Il est intéressant d'utiliser les deux informations (géométrique et radiométrique) afin d'affiner la classification. Pour ce faire, on adjoint l'information de radiométrie (c'est à dire la classe à laquelle appartient la zone classifiée dans le Chapitre 3) aux informations géométriques. Les objets ainsi obtenus résident alors dans l'espace $E = [0, 2\pi] \times \mathbb{R}^2$.

4.2.3 Résultats

Exemple jouet

Dans un premier temps, nous avons testé les algorithmes d'approximation et de classification sur un exemple jouet. Nous avons généré un ensemble de quadrilatères présentant des géométries différentes, mais pouvant être regroupés par familles d'orientation. Il s'agit d'un exemple « parfait » dans le sens où l'incertitude sur les orientations est quasi nulle. Les quadrilatères sont présentés sur la Figure 4.10.

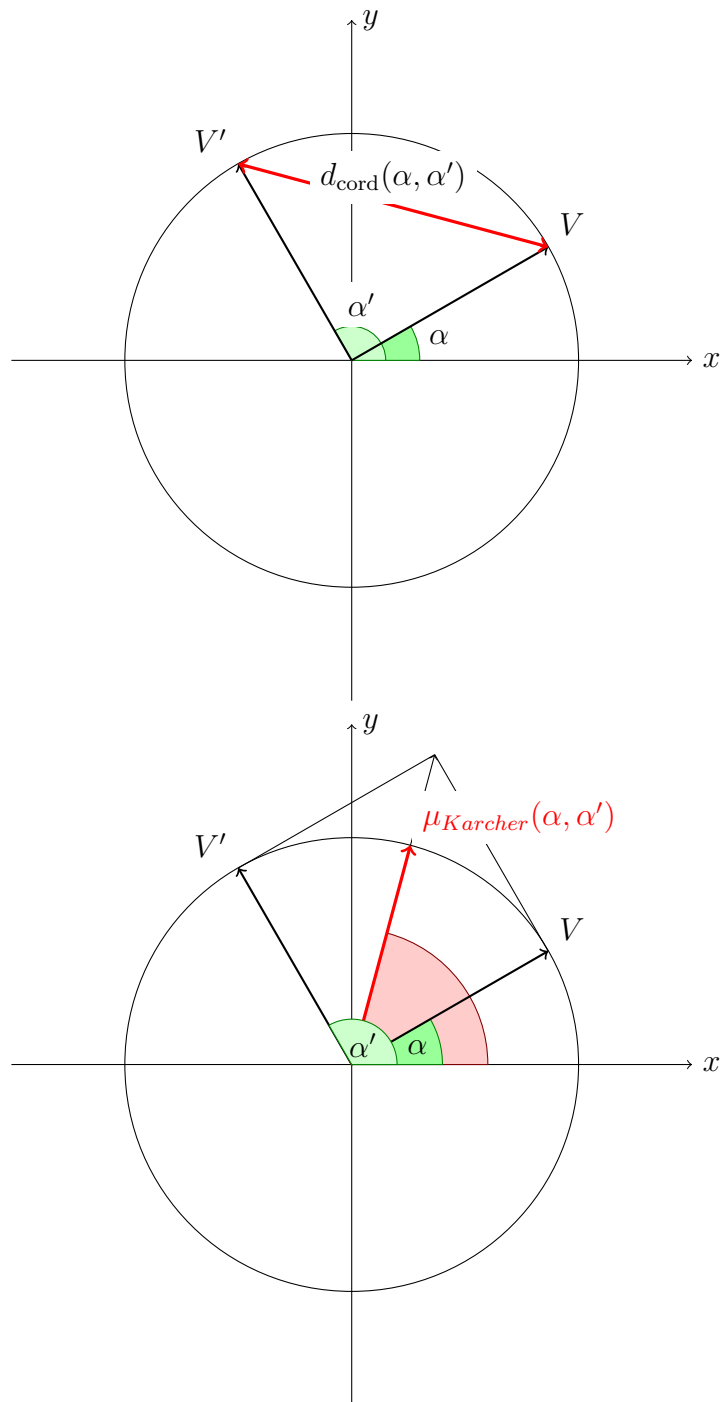


FIGURE 4.9 – Distance cordale (en haut) et moyenne de Karcher associée (en bas) dans l'espace (2D) des orientations.

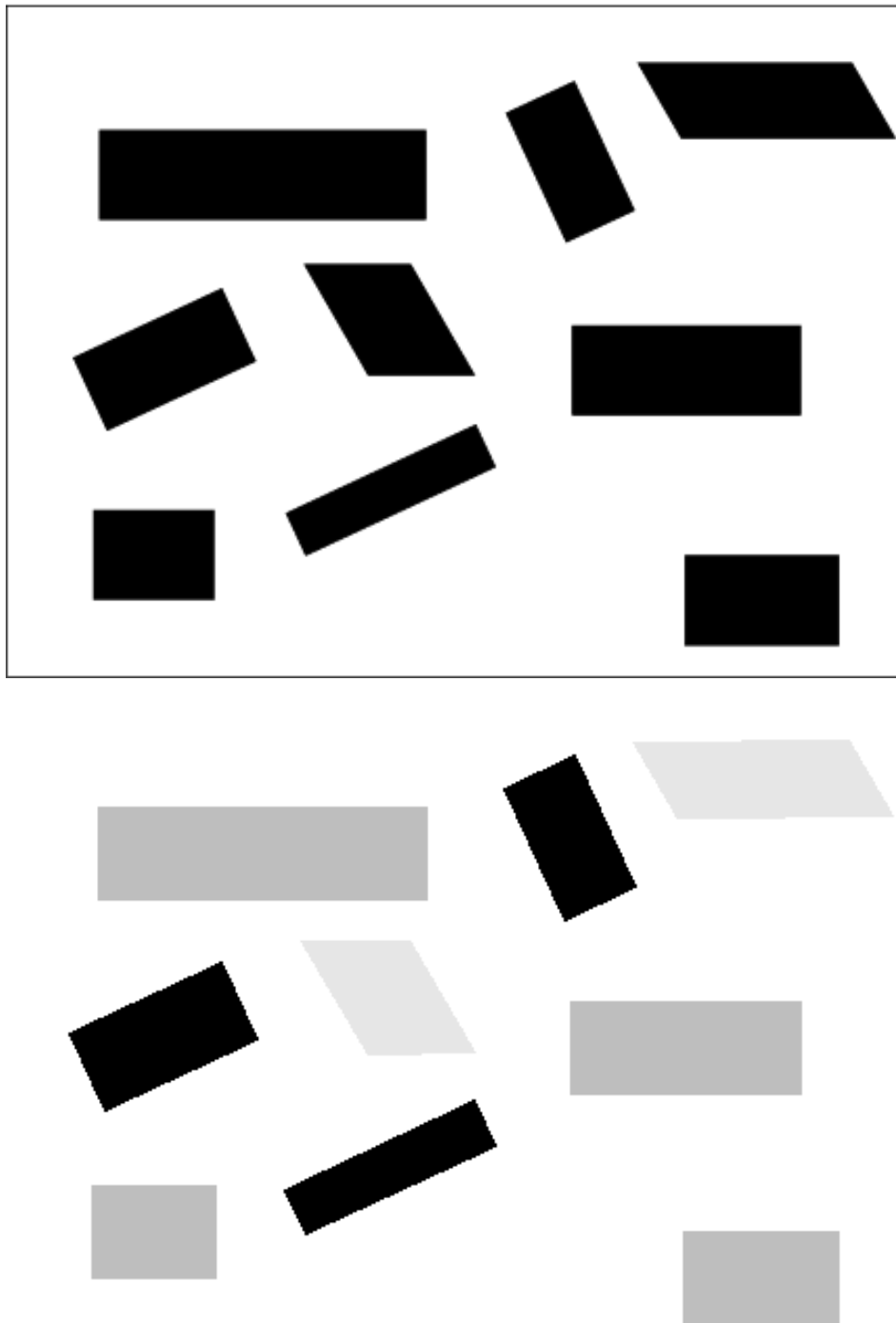


FIGURE 4.10 – Haut : Différents quadrilatères. Bas : classification par orientations de leurs côtés , les trois niveaux de gris représentent les classes auxquelles appartiennent les quadrilatères.

Images d'Abidjan

Nous avons testé l'algorithme de classification sur les zones connexes de la ville d'Abidjan (voir Figure 3.4) en tenant compte exclusivement des informations géométriques. L'objectif n'était pas d'obtenir une classification parfaite, mais de vérifier la pertinence de l'utilisation des formes pour la classification. L'algorithme détermine plusieurs classes. Les classes pertinentes sont indiquées sur la Figure 4.11. On remarque que la première correspond aux formes les plus régulières et proches des bâtiments. Malgré cela, certaines ombres sont classifiées par erreur dans cette classe. La seconde classe a plus de représentants parmi les ombres. D'autres représentants des ombres figurent dans la dernière classe.

Ces résultats sont perfectibles. Les ombres, qui sont pourtant caractérisées par des directions bien particulières ne sont pas regroupées dans la même classe. Ceci peut être imputé à deux problèmes : d'une part, la dépendance de notre algorithme d'approximation polygonale, vis-à-vis de l'initialisation. On remarque, par exemple, que certaines ombres perdent complètement cette direction justement à cause de l'initialisation. D'autre part, la segmentation proposée n'est pas forcément parfaite et des ombres peuvent être coupées en deux zones connexes, donnant lieu à des polygones dont l'information directionnelle est beaucoup moins porteuse de sens. Enfin, et c'est un des défauts principaux de notre algorithme, les quadrilatères ne sont pas forcément adaptés à toutes les zones que nous cherchons à caractériser. Dans le cas des ombres notamment, les ombres ont tendance à avoir une forme en « L ». Cette imprécision est rédhibitoire pour la précision de la classification, puisque la classification est fondée sur la somme des distances cordales pour chacun des côtés des polygones.

Nous avons ici évalué la seule information d'orientation. Il va de soi que dans la pratique, il est nécessaire de fusionner cette information avec l'information radiométrique.

Pour améliorer les résultats, plusieurs pistes doivent donc être explorées : d'une part améliorer l'algorithme d'approximation polygonale (et son initialisation), d'autre part, proposer un plus grand degré de liberté au niveau du nombre de côtés des polygones utilisés pour la classification. Enfin, on pourrait également envisager la possibilité de fusionner certaines zones proches pour former une seule zone connexe, ce qui pourrait s'avérer assez ardu.

4.3 Conclusion

Dans ce chapitre, nous avons pu voir que les informations d'orientation peuvent être intéressantes dans le cadre de la classification de changements. Néanmoins, afin d'obtenir une classification raisonnable, il est essentiel de trouver une pondération entre informations géométriques et radiométriques. Pour ce qui est de discriminer les ombres notamment, les informations radiométriques semblent considérablement plus intéressantes que

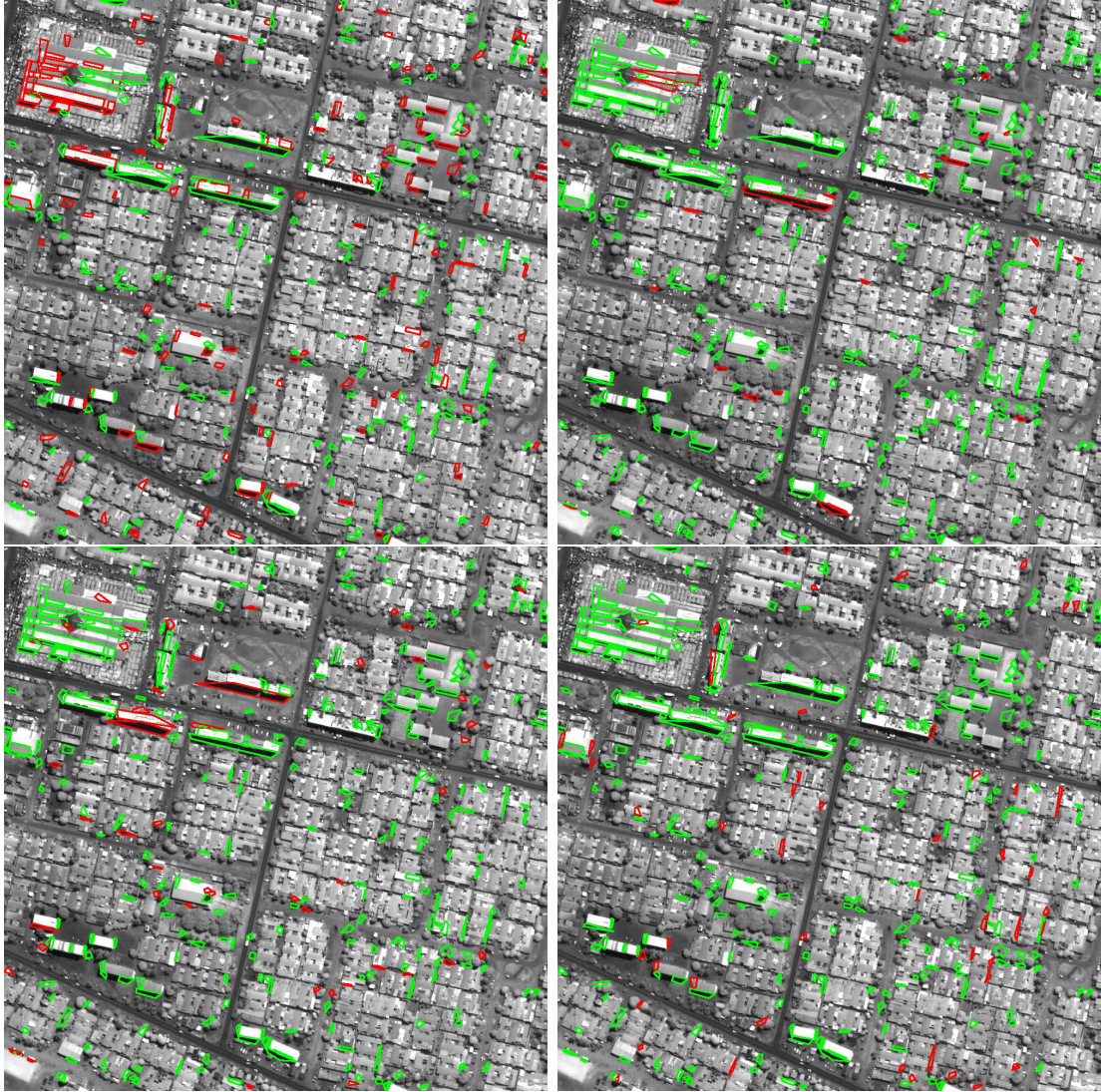


FIGURE 4.11 – Classes principales détectées par l’algorithme de classification polygonale. Pour chaque image, une classe différente est considérée, et ses représentants sont en rouge, les représentants des autres classes sont en vert.

les informations géométriques.

Toutefois, la limitation fondamentale de cet algorithme est que, contrairement à des approches de type « processus ponctuels », le nombre de côtés de l'ensemble des polygones est fixé une fois pour toute. Or, il s'est avéré bien souvent que des quadrilatères ne suffisent pas à caractériser tous les contours extraits dans les phases précédentes de notre chaîne de traitement. En conséquence, l'approximation polygonale sur ces zones est souvent de qualité médiocre. Pour améliorer ce type d'approche, plusieurs axes peuvent être étudiés :

- l'étude du problème min- \sharp adapté à la mesure d'erreur proposée dans ce chapitre, ce qui permettrait de proposer des polygones plus adaptés aux formes. Afin d'éviter des incohérences, un des points à étudier est le problème du choix de l'erreur d'approximation ε à utiliser. Une des solutions est probablement l'utilisation d'un ε dépendant de la taille (surface) de chaque forme.
- la proposition d'un algorithme de classification (très probablement non-fondé sur les *k-moyennes*) travaillant non plus sur l'espace \mathcal{P}_n , mais sur l'espace entier des polygones \mathcal{P} . Une des difficultés pour la conception de ce type d'algorithme est probablement de trouver une mesure de similarité entre des polygones ayant un nombre de côtés différents, mais présentant des similarités importantes au niveau des orientations.

Enfin, pour tenir compte de façon plus exhaustive de la géométrie, les notions de taille et de connexité doivent également être prises en comptes dans un algorithme de détection et de classification de changements.

Conclusion

Oh, can't anybody see,
We've got a war to fight,
Never found our way,
Regardless of what they say

BETH GIBBONS

Nous avons présenté tout au long de ce manuscrit des méthodes et des algorithmes pour la détection de changements à différents niveaux de traitement, depuis le couple d'images jusqu'au concept d'« objet-changement ».

Dans un premier temps, nous nous sommes penchés sur le problème de détection d'un masque de changements. Afin d'arriver à des résultats intéressants, il a été nécessaire de mettre au point une modélisation mathématique des scènes et des illuminations. Cette modélisation, quoi que rudimentaire (modèle d'illumination simplifié, hypothèses assez fortes sur les géométries des objets) nous a permis de donner un cadre plus formel pour définir un changement. C'est dans ce cadre que nous avons décrit une méthode originale pour construire un masque de changements pertinent. Ce dernier se caractérise par sa robustesse vis-à-vis des changements de contrastes locaux liés à des variations d'illumination (une des difficultés récurrentes dans notre problématique). L'algorithme proposé donne des résultats très encourageants et laisse espérer que des algorithmes plus complexes reposant sur les mêmes fondations théoriques pourront faire partie, à terme, des outils utilisés habituellement dans la détection de changements dans les différentes applications liées au domaine.

Dans une deuxième partie, nous avons pris comme point de départ les masques générés précédemment afin d'effectuer une première classification. Pour ce faire, nous avons pris le parti d'exploiter au maximum l'information radiométrique. Paradoxalement, il s'est avéré que la radiométrie est souvent porteuse d'une quantité importante d'informations pour la détection de changements. Ce type d'approche, en toute logique, a également montré ses limites, notamment au niveau de certaines classes qui regroupaient des objets fondamentalement différents. Ces différents algorithmes ont été testés sur des couples d'images réelles (Abidjan, Pékin ...).

Naturellement, les limites exposées précédemment nous ont amenés à nous interroger

sur l'importance d'informations moins locales, comme les informations de voisinage, ou la notion d'« objet » dans une scène. Une des informations qui nous ont paru intéressantes était la notion d'orientation des formes détectées, qui correspondaient en général à des bâtiments, des ombres, ou des façades. Nous avons alors adapté des algorithmes d'approximation polygonale et de classification à notre problématique, puis nous avons tenté de mettre en évidence la pertinence de cette information dans le cadre de la détection de changements, avec des résultats mitigés.

Comme nous avons pu le voir le long de ce manuscrit, l'interprétation est fondamentale en traitement d'images et la détection de changements n'est pas une exception. En effet, sur une image, un changement en soi n'a aucun sens, mais doit être rapporté à l'objet ou la partie de scène qu'il modifie, qu'il s'agisse d'une disparition, d'une apparition ou d'une modification de cet objet.

À ce titre, les trois approches proposées précédemment correspondent à des niveaux différents d'interprétation. La classification par *k-moyennes*, proposée dans le Chapitre 3, correspond à un travail au niveau pixélique. C'est à dire à un niveau particulièrement bas. Un post-traitement est alors nécessaire pour obtenir des résultats exploitables. La méthode exposée dans le Chapitre 4 consiste à extraire des zones à partir des pixels classifiés et à construire des objets. Ensuite, une classification plus fine, utilisant les propriétés d'orientation de ces objets est proposée. Il s'agit donc d'une technique complémentaire de la première, puisqu'elle travaille alors au niveau de l'objet. Néanmoins, les limitations de cette dernière technique sont assez importantes et sont dues notamment aux hypothèses *a priori* trop fortes sur la nature de ces objets (polygones réguliers).

La technique proposée dans le Chapitre 2 repose, quant-à-elle, sur les lignes de niveau. Cette structure est intéressante, parce qu'elle semble s'adapter aux objets. En ce sens les lignes de niveau peuvent être considérées comme des structures intermédiaires (et variables) entre les pixels et les objets.

En outre, cette approche est intéressante pour plusieurs raisons :

- Le premier avantage de cette méthode est qu'elle repose sur une modélisation relativement simple de l'image en fonction de la scène observée. On dispose alors d'une panoplie intéressante d'outils mathématiques pour résoudre.
- Par rapport à la méthode proposée dans le Chapitre 4, les hypothèses sur la géométrie des objets présents dans la scène sont bien moins restrictives.

Ainsi, cette approche s'appuie sur une modélisation et des hypothèses fortes au niveau de l'illumination des scènes, mais s'affranchit quasi-totalement d'hypothèses au niveau des géométries des objets 2D détectés, ce qui semble lui conférer ses qualités de robustesse. Cet aspect devra être pris en compte dans le développement de nouvelles approches en détection et classification de changements.

Les perspectives à l'issue de ce travail sont nombreuses. Des améliorations importantes sont envisageables par rapport à l'algorithme de détection de changements présenté dans

le Chapitre 2, notamment au niveau du choix de la quantification qui est resté dans notre cas relativement simple.

En ce qui concerne les classifications présentées dans les chapitres ultérieurs, elles ne prétendent en aucun cas être complètes. Par exemple, déterminer des classes en se restreignant aux informations d'orientation, sans considérer les informations de taille, ou de voisinage des formes serait complètement illusoire. Pour obtenir des résultats réellement probants, il faudra intégrer la notion d'orientation, dans un ensemble plus vaste de caractéristiques.

Enfin, un point important est l'utilisation du double résultat fourni par l'algorithme présenté dans le Chapitre 2. En effet, pour l'instant, les algorithmes suivants ne prennent qu'un seul masque de changements en entrée et ne tirent pas parti de la classification préétablie par le premier. Il s'agit d'un point essentiel sur lequel des améliorations considérables sont envisageables.

Annexe A

Rappels théoriques sur les champs de Markov en traitement d'images

Les champs de Markov sont un outil assez classique en traitement d'images, ils ont été introduits dans [Besag, 1974] et popularisés par [Geman et Geman, 1984]. Cette annexe a pour objectif de présenter brièvement les notions fondamentales nécessaires à leur compréhension et leur utilisation dans le cas particulier du traitement d'images. Néanmoins l'énoncé de certains résultats et définitions a été simplifié en tenant compte des spécificités de notre application. Pour des définitions plus générales, le lecteur pourra se référer à [Guyon, 1993], et une description plus adaptée au contexte du traitement d'images est fournie dans [Graffigne et Zerubia, 1995].

Dans une deuxième partie, on décrit le recuit simulé, un algorithme permettant d'atteindre une configuration qui correspond au maximum de probabilité d'un champ de Markov.

A.1 Modélisation

Définition 15 (Configuration). *Soit S un ensemble dénombrable de points. Soit E un espace d'état métrique complet séparable et ε sa tribu borélienne. Une configuration sur S est un élément ω de l'espace des configurations $(\Omega, \mathcal{F}) = (E, \varepsilon)^S$.*

Définition 16 (Champ aléatoire). *Soit S un ensemble dénombrable de points, E un espace d'état métrique complet séparable et ε sa tribu borélienne, un champ aléatoire sur S est une mesure de probabilité μ sur son espace de configurations $(\Omega, \mathcal{F}) = (E, \varepsilon)^S$.*

Définition 17 (Système de voisinages). *Soit S un ensemble dénombrable de points. On*

défini le système de voisinages comme une fonction $V : S \rightarrow P(S)$ telle que :

$$\forall s \in S, \begin{cases} \text{card}(V(s)) < \infty \\ s \notin V(s) \\ \forall s' \in V(s), s \in V(s') \end{cases} \quad (\text{A.1})$$

Cette définition correspond aux arêtes E d'un graphe simple (*i.e.* sans boucle, ni arêtes parallèles) non-orienté G pour lequel l'ensemble des sommets est S . On appelle $V(s)$ le voisinage de s .

Nous pouvons maintenant introduire les champs de Markov :

Définition 18 (Champ de Markov). *Soit S un ensemble dénombrable de points muni d'un système de voisinages V . Soit μ un champ aléatoire sur (Ω, \mathcal{F}) . μ est un champ de Markov si pour tout point s de S , tout événement A de \mathcal{F}_s , et toute configuration ω de Ω :*

$$\mu_s(A|\omega(S - \{s\})) = \mu_s(A|\omega(V(s))) \quad (\text{A.2})$$

En d'autres termes, pour un champ de Markov, la probabilité conditionnelle de la restriction d'une configuration à un site ne dépend alors que de la restriction de la configuration aux sites voisins de ce dernier.

Définition 19 (Potentiel d'interaction). *Soit μ un champ aléatoire sur (Ω, \mathcal{F}) . Un potentiel d'interaction est une famille $\varphi = (\varphi_A, A \in \mathcal{S})$ d'applications $\varphi_A : \Omega(A) \rightarrow \mathbb{R}$ telles que :*

- Pour tout A de \mathcal{S} , φ_A est $\mathcal{F}(A)$ -mesurable ;
- Si $\Lambda \in \mathcal{S}$ et $\omega \in \Omega$, la somme suivante existe :

$$U_\Lambda^\varphi(\omega) = \sum_{A \in \mathcal{S} \text{ tel que } A \cap \Lambda \neq \emptyset} \varphi_A(\omega) \quad (\text{A.3})$$

Définition 20 (Clique). *Soit S un ensemble fini de points muni d'une relation de voisinage V . Une clique k de S est un sous ensemble de S tel que $\forall (s, s') \in S^2$ tel que $s \neq s'$, on a $s' \in v(s)$.*

Théorème 3 (Théorème de Hammersley-Clifford). *Soit μ un champ aléatoire sur (Ω, \mathcal{F}) , muni d'une structure de voisinage V . Les deux propositions suivantes sont équivalentes :*

- μ est un champ de Markov tel que :

$$\forall x \in \Omega, \mu(x) > 0 \quad (\text{A.4})$$

– il existe un potentiel d'interaction φ tel que $\forall x \in \Omega$, on a :

$$\mu(x) = \exp \left(- \sum_{A \in \mathcal{C}} \varphi_A(x_A) \right) \quad (\text{A.5})$$

ou \mathcal{C} est la famille des cliques de μ .

Par le biais de ce théorème, on peut manipuler un champ de Markov en raisonnant via les potentiels définis sur des cliques.

Dans le cas du traitement d'images, l'ensemble des points représente le plus souvent les pixels de l'image. L'espace des configurations (Ω, F) représente les réalisations en terme d'images ou de classes. La relation de voisinage, quant à elle, peut être plus variée. On peut imaginer des relations de voisinage exotiques, mais les relations de voisinage les plus utilisées sont des relations simples et souvent intuitives. Parmi elles, les plus classiques sont :

le voisinage 4-connexe : Dans ce type de voisinage, chaque pixel de l'image est connecté par une arête à ses voisins directs à gauche, droite, en haut et en bas. Dans ce cas, les cliques du graphe sont les cliques de cardinal 1 et 2 et représentées sur la Figure A.1.



FIGURE A.1 – Gauche : exemple de voisinage 4-connexe, droite : Cliques dans un graphe 4-connexe

le voisinage 8-connexe : Ce voisinage ajoute aux voisins définis dans le cas précédent, les pixels diagonaux (*i.e.* les pixels situés immédiatement en haut-à droite, en haut-à gauche, en bas-à droite, en bas-à gauche). Les cliques relatives à ce système de voisinage sont de cardinal 1, 2, 3 et 4. Elles sont représentées, avec un exemple d'un pixel et ses 8-voisins, sur la Figure A.2.

A.2 Optimisation

Une des utilisations classiques des champs de Markov en traitement d'images est le problème de classification et de régularisation des pixels d'une image. En général, on dispose d'une image I que l'on cherche à régulariser et d'un espace des configurations Ω pour les pixels. On propose alors un champ de Markov qui tient compte, d'une part de

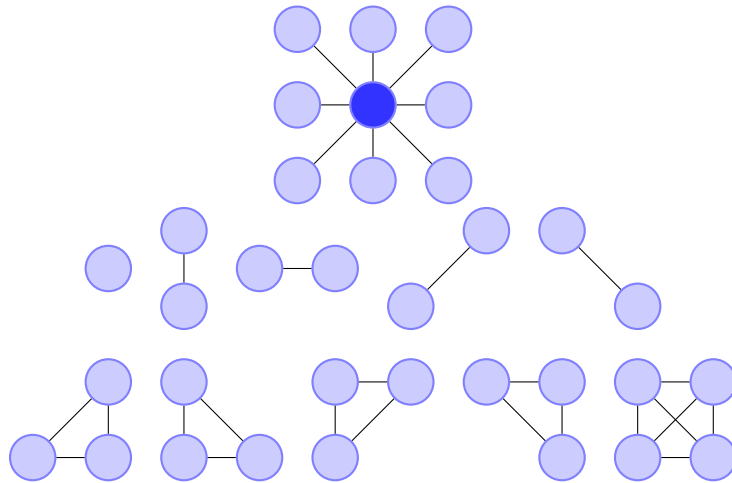


FIGURE A.2 – Haut : exemple de voisinage 8-connexe. Bas : cliques dans un graphe 8-connexe

l'image I par le biais d'un terme d'attache aux données et d'une contrainte de régularité (souvent liée aux potentiels sur les cliques). La meilleure classification ou configuration correspond alors au maximum de probabilités de ce champ, *i.e.* au minimum de l'énergie associée. Afin de faire converger l'énergie vers le minimum dans l'espace des configurations, on applique le schéma de recuit simulé [Geman et Geman, 1984; Graffigne et Zerubia, 1995]. Ce dernier est décrit dans l'Algorithme A.1 avec une dynamique de Metropolis

[Metropolis et al., 1953] :

Algorithme A.1 : Schéma de recuit simulé

Données : $(I_s)_{s \in S}$: l'image à régulariser. μ un champ de Markov sur les sites s de S , T_0 : la température initiale

Résultat : La configuration $\bar{\omega}$ de μ maximisant la probabilité définie dans l'Équation A.5

début

$t \leftarrow 0$;

 Tirer aléatoirement une configuration initiale $\omega_0 \in \Omega$;

$T \leftarrow T_0$;

tant que $T > \varepsilon$ **faire**

pour chaque site $s \in S$ **faire**

 Choix aléatoire d'une valeur s'_i (suivant une loi uniforme);

si $E(I') - E(I) < 0$ *ou* $\text{rand}() < \exp\left(-\frac{E(I') - E(I)}{k.T}\right)$ **alors**

$s_i \leftarrow s'_i$;

fin

 /*diminution de la température

*/

fin

$T \leftarrow \ln\left(T_0 \frac{t}{t+1}\right)$;

$t \leftarrow t + 1$;

fin

fin

En pratique, le calcul des énergies dans l'Algorithme A.1 est très simple : grâce à la probabilité markovienne, il n'est nécessaire de calculer que la variation d'énergie des cliques auxquelles appartient le site s .

Table des figures

2.1	Graphe des radiométries au temps T_2 en fonction des radiométries au temps T_1 illustrant le principe de l'algorithme de R. Wiemker : la première analyse en composantes principales est fortement biaisée par les nuages des points « changés » et les itérations successives corrigent ce biais en pondérant les points avec leur distance à l'axe principal.	23
2.2	Prises de vue de l'aéroport d'Abidjan. Gauche : le 04/02/2003, droite : 07/07/2003. Images Quickbird (61 cm de résolution), © DigitalGlobe, distribution Spot Image, fournies par la DGA.	26
2.3	Traitements sur l'aéroport d'Abidjan. Haut : Simple différence (en valeur absolue), bas : Quantité de changements (valeur absolue)	27
2.4	Aéroport d'Abidjan , Haut : Probabilité de changements après convergence. Bas : Masque de changement obtenu après régularisation par un champ de Markov.	28
2.5	Images originales d'une zone commerciale d'Abidjan , © DigitalGlobe, distribution Spot Image fournies par la DGA.	29
2.6	Haut : probabilité de changements après convergence sur une zone commerciale d'Abidjan, bas : masque de changements après régularisation par champs de Markov	30
2.7	Haut : images d'un dôme en 3 dimensions illuminé d'après le modèle de Phong [Phong, 1975] en utilisant deux directions d'incidence différentes. Bas : quelques lignes de niveau associées aux illuminations respectives.	33
2.8	Notations	34
2.9	Haut : images d'un cône sous deux lumières d'incidences différentes. Bas : quelques unes de leurs lignes de niveau.	40
2.10	Exemples de non-invariance des lignes de niveau dans le cas C^2 par morceaux. Les éléments colorés représentent des lignes de niveau isolées.	45
2.11	Illustration des erreurs de parallaxe : deux scènes identiques sont représentées avec des géométries différentes.	48
2.12	Dictionnaire de toits utilisés dans [Lafarge et al., 2008] pour modéliser des villes d'architecture européenne. Elles sont toutes constituées de morceaux de surfaces développables. Images fournies gracieusement par F. Lafarge.	49

2.13	Haut : images d'un toit simple à deux versants sous deux illuminations différentes. Bas : FLST de chaque image.	53
2.14	Exemple jouet. Haut : Deux images sous des conditions différentes d'illumination, avec par ailleurs quelques changements sur les scènes. Bas : vérités de terrain (les changements sont indiqués en vert).	54
2.15	Exemple jouet. Première ligne, détection de changements en utilisant notre algorithme. Deuxième ligne : détection de changements en utilisant une égalisation de contraste par moindres carrés. Troisième ligne : détection de changements en utilisant la FLST.	55
2.16	Images Quickbird (61 cm de résolution) de l'aéroport d'Abidjan. Haut : l'aéroport le 04/02/2003 et le 07/07/2003. Bas : vérités de terrain (les changements sont indiqués en vert).	57
2.17	Images Quickbird (61 cm de résolution) de l'aéroport d'Abidjan. Première ligne : détection de changements en utilisant notre algorithme. Deuxième ligne : détection de changements en utilisant une égalisation de contraste aux moindres carrés. Troisième ligne : détection de changements en utilisant la FLST.	58
2.18	Courbes ROC pour l'aéroport d'Abidjan.	59
2.19	Images Quickbird (61cm de résolution) de Pékin. Première ligne : Pékin en 2001 (gauche) et 2003 (droite). Deuxième ligne - gauche : vérité de terrain (les changements sont indiqués en vert) - droite : Détection de changements en utilisant notre algorithme. Troisième ligne - gauche : détection de changements en utilisant un projection monotone - droite : détection de changements en utilisant la FLST.	60
2.20	Courbes ROC pour le couple d'images de la banlieue de Pékin.	61
3.1	Deux populations « bien marquées »	68
3.2	Cas « litigieux »	69
3.3	Haut : Images originales d'Abidjan, © Spot Image fournie par la DGA. Bas : Masque de changements obtenu par la méthode de R. Wiemker (cf. Chapitre 2)	77
3.4	Première classification par <i>k-moyennes</i> , ici quatre classes sont obtenues. On distingue notamment les ombres parmi les objets détectés.	79
3.5	Classification par <i>k-moyennes</i> entropiques (11 classes). On remarque que certaines classes (ombres, notamment) contiennent des pixels isolés.	80
3.6	Après régularisation par champs de Markov de la classification obtenue sur la Figure 3.5, on remarque que les classes (notamment les ombres) sont plus homogènes.	81
3.7	Haut : Images originales de l'aéroport d'Abidjan, prises en février et juillet 2003 © Spot Image, fournies par la DGA. Bas : masque de changements issu de l'approche par lignes de niveau (voir Chapitre 2)	82

3.8	Haut : classification par k -moyennes entropiques avec 4 classes : les avions figurant sur la première image originales (voir Figure 3.7) sont divisés en plusieurs classes selon leur couleur (blanc ou camouflage), les ombres sur la première image originale sont confondues avec les avions apparaissant sur la deuxième image originale. Bas : Régularisation par champs de Markov de la classification des changements sur l'aéroport d'Abidjan : on remarque que les avions sont détectés plus uniformément. On peut plus facilement discriminer les apparitions et les disparitions. En revanche, les fausses alarmes liées aux ombres restent problématiques.	83
3.9	Classification avec 8 classes : les avions figurant sur la première image originale (voir Figure 3.7) sont divisés en plusieurs classes selon leur couleur (blanc ou camouflage), les ombres sur la première image originale sont confondues avec les avions apparaissant sur la deuxième image originale.	84
3.10	Gauche : Masque de changements obtenu en appliquant la méthode de R. Wiemker et résultat après application de notre classifieur et la régularisation par champs de Markov	85
4.1	Illustration de l'erreur en norme xor	90
4.2	Approximations polygonales d'une forme présentant des points aberrants. Gauche : approximation optimale avec notre mesure d'erreur, droite : approximation optimales avec une mesure d'erreur « classique ».	91
4.3	Influence du déplacement d'un point P_i sur l'erreur d'approximation polygonale. Les aires successives des zones Q_{i_0} à Q_{i_n} sont alternativement ajoutées et retranchées à l'erreur d'approximation.	92
4.4	Initialisation	96
4.5	Approximation polygonale sur un « exemple jouet ». Le quadrilatère finit par converger sur le carré après quelques itérations.	97
4.6	Test sur une forme d'avion extraite sur l'aéroport d'Abidjan pour un $n = 9$ (haut) et $n = 11$ (bas). À gauche : initialisation, à droite : convergence.	97
4.7	Détail de la classification de la ville d'Abidjan obtenue au Chapitre 3.	98
4.8	Approximation polygonale sur la classification (Figure 4.7) : Gauche : initialisation, Droite : convergence. On remarque qu'après convergence, les directions des ombres et des bâtiments sont extraites correctement.	99
4.9	Distance cordale (en haut) et moyenne de Karcher associée (en bas) dans l'espace (2D) des orientations.	101
4.10	Haut : Différents quadrilatères. Bas : classification par orientations de leurs côtés , les trois niveaux de gris représentent les classes auxquelles appartiennent les quadrilatères.	102
4.11	Classes principales détectées par l'algorithme de classification polygonale. Pour chaque image, une classe différente est considérée, et ses représentants sont en rouge, les représentants des autres classes sont en vert.	104

A.1	Gauche : exemple de voisinage 4-connexe, droite : Cliques dans un graphe 4-connexe	113
A.2	Haut : exemple de voisinage 8-connexe. Bas : cliques dans un graphe 8-connexe	114

Bibliographie

- Aach, T. et Kaup, A. (1995). Bayesian algorithms for adaptive change detection in image sequences using Markov random fields. *Signal Processing : Image Communication*, 7(2) :147–160.
- Alboody, A., Sedes, F., et Inglada, J. (2008). Post-classification and spatial reasoning : new approach to change detection for updating GIS database. Dans *3rd International Conference on Information and Communication Technologies : From Theory to Applications (ICTTA)*, Brest, pages 1–7. ENST Brest.
- Arthur, D. et Vassilvitskii, S. (2007). k-means++ : the advantages of careful seeding. Dans *Proc. of ACM-SIAM Symposium On Discrete Algorithms*, Nouvelle Orléans, Louisiane, États-Unis, pages 1027–1035, Philadelphie, Pennsylvanie, État-Unis. Society for Industrial and Applied Mathematics.
- Ballester, C., Cubero-Castan, E., Gonzalez, M., et Morel, J.-M. (2000). Contrast invariant image intersection. *Advanced Mathematical Methods in Measurement and Instrumentation*, pages 41–55.
- Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, 36(2) :192–236.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.
- Black, M. J., Fleet, D. J., et Yacoob, Y. (2000). Robustly estimating changes in image appearance. *Computer Vision and Image Understanding*, 78(1) :8–31.
- Borchani, M., Cloppet, F., Volkan, A., et Stamon, G. (2004). Change detection in aerial images. Dans *Canadian Conference on Computer and Robot Vision*, Québec, Canada, pages 354–360, Washington, DC, États-Unis. IEEE Computer Society.
- Bottou, L. et Bengio, Y. (1995). Convergence properties of the k-means algorithms. Dans *Proc. of the 1994 conference on Advances in Neural Information Processing Systems*, Denver, Colorado, États-Unis, pages 585–592.

- Brown, L. G. (1992). A survey of image registration techniques. *ACM Comput. Surv.*, 24(4) :325–376.
- Bruzzone, L. et Prieto, D. (2002). An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images. *IEEE Trans. on Image Processing*, 11(4) :452–466.
- Bruzzone, L. et Prieto, D. F. (2000). An adaptive parcel-based technique for unsupervised change detection. *International Journal of Remote Sensing*, 21(4) :817–822.
- Can, A., Stewart, C. V., Roysam, B., et Tanenbaum, H. L. (2002). A feature-based robust hierarchical algorithm for registration pairs of images of the curved human retina. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(3) :347–364.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6) :679–698.
- Cao, F. et Bouthemy, P. (2006). A general principled method for image similarity validation. Dans *Proc. Int. Workshop on Adaptive Multimedia Retrieval*, Genève, Suisse, volume 4398, pages 57–70.
- Caselles, V., Coll, B., et Morel, J.-M. (1999a). Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33.
- Caselles, V., Coll, B., et Morel, J.-M. (2002). Geometry and color in natural images. *JMIV*, 16 :89–107.
- Caselles, V., Lisani, J., Morel, J.-M., et Sapiro, G. (1999b). Shape preserving local histogram modification. *IEEE Trans. on Image Processing*, 8 :220–230.
- Chatelain, F., Tourneret, J., Inglada, J., et Ferrari, A. (2007). Bivariate gamma distributions for image registration and change detection. *IEEE Trans. on Image Processing*, 16(7) :1796–1806.
- Chen, H., Belhumeur, P., et Jacobs, D. (2000). In search of illumination invariants. *Proc. of Intl. Conference on Computer Vision and Pattern Recognition*, 2 :254–261.
- Chen, H., Belhumeur, P., et Jacobs, D. (2008). In search of illumination invariants. *soumis à Int. Journal of Computer Vision*.
- Clifton, C. (2003). Change detection in overhead imagery using neural networks. *Applied Intelligence*, 18(2) :215–234.
- Coppin, P. R. et Bauer, M. E. (1994). Processing of multitemporal landsat TM imagery to optimize extraction of forest cover change features. *IEEE Trans. on Geoscience and Remote Sensing*, 32(4) :918–927.

- Coppin, P. R., Jonckheere, I., Nackaerts, K., Muys, B., et Lambin, E. (2004). Digital change detection methods in ecosystem monitoring : a review. *International Journal of Remote Sensing*, 25(9) :1565–1596(32).
- Courteille, F., Crouzil, A., Durou, J.-D., et Gurdjos, P. (2007). Shape from shading for the digitization of curved documents. *Machine Vision and Applications*, 18(5) :301–316.
- Day, X. et Khorram, S. (1998). The effects of image misregistration on the accuracy of remotely sensed change detection. *IEEE Trans. on Geoscience and Remote Sensing*, 36 :1566–1577.
- Droske, M. et Rumpf, M. (2004). A variational approach to non-rigid morphological registration. *SIAM Appl. Math.*, 64(2) :668–687.
- Fournier, A., Descombes, X., et Zerubia, J. (2008). Mixing geometric and radiometric features for change classification. Dans Bouman, C. A., Miller, E. L., et Pollak, I., éditeurs, *Proc. SPIE Symposium on Electronic Imaging*, San José, Californie, États-Unis, volume 6814, page 681408. SPIE.
- Frigui, H. et Krishnapuram, R. (1996). A robust clustering algorithm based on competitive agglomeration and soft rejection of outliers. Dans *Proc. of Intl. Conference on Computer Vision and Pattern Recognition*, San Francisco, Californie, États-Unis, pages 550–555.
- Geman, S. et Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(6) :721–741.
- Graffigne, C. et Zerubia, J. (1995). Champs markoviens et segmentation. Dans Cocquerez, J.-P. et Philipp, S., éditeurs, *Analyse d'images : filtrage et segmentation*, pages 379–388. Masson.
- Guichard, F. et Morel, J.-M. (2001). Image analysis and P.D.E.s.
- Guyon, X. (1993). *Champs aléatoires sur un réseau, modélisations, statistique et applications*. Masson.
- Hong, G. et Zhang, Y. (2005). The image registration technique for high resolution remote sensing image in hilly area. Dans *IEEE/ISPRS 3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas*.
- Horn, B. et Brooks, M. (1989). *Shape from shading*. MIT Press.
- Huete, A. R. (1988). A soil-adjusted vegetation index(SAVI). *Remote Sensing of Environment*, 25(3) :295–309.

- Inglada, J. (2001). État de l'art en détection de changements sur les images de télédétection. Rapport technique, CNES, Toulouse.
- Janisch, K. (1984). *Topology*. Springer.
- Karcher, H. (1977). Riemannian center of mass and mollifier smoothing. *Communications on Pure and Applied Mathematics*, 30 :509–541.
- Kindermann, R. et Snell, L. J. (1980). *Markov random fields and their applications*. American Mathematical Society.
- Kolesnikov, A. (2003). *Efficient algorithms for vectorization and polygonal approximation*. Thèse de doctorat, Université de Joensuu, Finlande.
- Kolesnikov, A. et Fränti, P. (2007). Polygonal approximation of closed discrete curves. *Pattern Recognition*, 40(4) :1282–1293.
- Kybic, J. et Unser, M. (2003). Fast parametric elastic image registration. *IEEE Trans. on Image Processing*, 12(11) :1427–1442.
- Lafarge, F., Descombes, X., Zerubia, J., et Pierrot-Deseilligny, M. (2008). Building reconstruction from a single dem. Dans *Proc. of Intl. Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, États-Unis, pages 1–8.
- Li, W., Li, X., Wu, Y., et Hu, Z. (2006). A novel framework for urban change detection using vhr satellite images. Dans *Proc. of International Conference on Pattern Recognition*, Hong Kong, République populaire de Chine, volume 2, pages 312–315.
- Lillestrand, R. (1972). Techniques for change detection. *IEEE Trans. on Computers*, 21(7) :654–659.
- Lisani, J. et Morel, J.-M. (2003). Detection of major changes in satellite images. Dans *ICIP*, Barcelone, Espagne, volume 1, pages 941–944.
- Liu, D., Gong, P., Kelly, M., et Guo, Q. (2006). Automatic registration of airborne images with complex local distortion. *Photogrammetric Engineering & Remote Sensing*, 72(9) :1049–1060.
- Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Trans. on Information Theory*, 28(2) :129–137.
- Lorette, A. (1999). *Analyse de texture par méthodes markoviennes et par morphologie mathématique : application à l'analyse des zones urbaines sur des images satellitaires*. Thèse de doctorat, Université de Nice-Sophia Antipolis.

- Lu, D., Mausel, P., Brondiacutezio, E., et Moran, E. (2004). Change detection techniques. *International Journal of Remote Sensing*, 25(12) :2365–2401.
- Lyon, J. G., Yuan, D., Lunetta, R. S., et Elivdge, C. D. (1998). A change detection experiment using vegetation indices. *Photogrammetric Engineering & Remote Sensing*, 64(2) :143–150.
- Mardia, K. et Jupp., P. (2000). *Directional Statistics*. Wiley.
- Mardia, K., Kent, J., et Bibby, J. (1979). *Multivariate Analysis*. Academic Press.
- Marr, D. (1982). *Vision : A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman and Company, NY.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., et Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6) :1087–1092.
- Moisan, L. (2005). Modeling and image processing. Notes de cours de l'ENS Cachan.
- Monasse, P. (1999). Contrast invariant registration of images. Dans *Proc. of Intl. Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona , États-Unis, volume 6, pages 3221–3224.
- Monasse, P. et Guichard, F. (2000). Fast computation of a contrast-invariant image representation. *IEEE Trans. on Image Processing*, 9 :860–872.
- Mundy, J. et Zisserman, A. (1992). *Geometric Invariance in Computer Vision*. MIT Press, Cambridge.
- Palubinskas, G., Descombes, X., et Kruggel, F. (1998). An unsupervised clustering method using the entropy minimization. Dans *Proc. of International Conference on Pattern Recognition*, Brisbane, Australie, volume 2, pages 1816–1818.
- Patra, S., Ghosh, S., et Ghosh, A. (2007). Semi-supervised Learning with Multilayer Perceptron for Detecting Changes of Remote Sensing Images. Dans *Proc. of International Conference on Pattern Recognition and Machine Intelligence*, volume 4815 de *Lecture Notes in Computer Science*, pages 161–168. Springer.
- Perez, J. et Vidal, E. (1994). Optimum polygonal approximation of digitized curves. *Pattern Recognition Letters*, 15(8) :743–750.
- Phong, B. T. (1975). Illumination for computer generated pictures. *Comm. ACM*, 18(6) :311–317.

- Radke, R. J., Andra, S., Al-Kofahi, O., et Roysam, B. (2005). Image change detection algorithms : a systematic survey. *IEEE Trans. on Image Processing*, 14(3) :294–307.
- Rignot, E. et van Zyl, J. (1993). Change detection techniques for ERS-1 SAR data. *IEEE Trans. on Geoscience and Remote Sensing*, 31(4) :896–906.
- Robin, A. (2007). *Détection de changements et classification sous-pixelliques en imagerie satellitaire. Application au suivi temporel des surfaces continentales*. Thèse de doctorat, Université René Descartes - Paris V.
- Rouse, J. W., Haas, R. H., Schell, J. A., et W.Deering, D. (1973). Monitoring vegetation systems in the great plains with ERTS. Dans *Third ERTS Symposium*.
- Rudin, L., Osher, S., et Fatemi, E. (1992). Nonlinear Total Variation Based Noise Removal. *Physica D*, 60 :259–268.
- Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(4) :379–423.
- Singh, A. (1989). Digital change detection techniques using remotely-sensed data. *International Journal of Remote Sensing*, 10(6) :989–1003.
- Spivak, M. (1999). *A comprehensive Introduction to Differential Geometry*, volume 3. Publish or Perish, Inc.
- Stefanov, W., Ramsey, M. S., et Christensen, P. R. (2001). Monitoring urban land cover change : An expert system approach to land cover classification of semiarid to arid urban centers. *Remote Sensing of Environment*, 77(2) :173–185.
- Tan, C., Zhang, L., Zhang, Z., et Xia, T. (2006). Restoring warped document images through 3D shape modeling. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(2) :195–208.
- Tang, F. et Prinet, V. (2007). Computing invariants for structural change detection in urban areas. Dans *Urban Remote Sensing Joint Event, 2007*, pages 1–6.
- Teboulle, M., Berkhin, P., Dhillon, I., Guan, Y., et Kogan, J. (2006). Clustering with entropy-like k-means algorithms. Dans Kogan, J., Nicholas, C., et Teboulle, M., éditeurs, *Grouping Multidimensional Data, Recent Advances in Clustering*, pages 127–160. Springer.
- Theiler, J. et Perkins, S. (2007). Resampling approach for anomalous change detection. Dans *Proceedings of SPIE*, Orlando, Floride, États-Unis, volume 6565, page 65651U. SPIE.

- Toth, D., Aach, T., et Metzler, V. (2000). Illumination-invariant change detection. Dans *Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium*, pages 3–7.
- Townshend, J., Justice, C., Gurney, C., et McManus, J. (1992). The impact of mis-registration on change detection. *IEEE Trans. on Geoscience and Remote Sensing*, 30(5) :1054–1060.
- Wada, T., Ukida, H., et Matsuyama, T. (1997). Shape from shading with interreflections under a proximal light source : distortion-free copying of an unfolded book. *Int. Journal of Computer Vision*, 24(2) :125–135.
- Wang, F. (1993). A knowledge-based vision system for detecting land changes at urban fringes. *IEEE Trans. on Geoscience and Remote Sensing*, 31(1) :136–145.
- Watanabe, S., Miyajima, K., et Mukawa, N. (1998). Detecting changes of buildings from aerial images using shadow and shading model. Dans *Proc. of International Conference on Pattern Recognition*, volume 2, pages 1408–1412.
- Weiss, P., Blanc-Féraud, L., et Aubert, G. (2007). Efficient schemes for total variation minimization under constraints in image processing. RR 6260, INRIA.
- Wiemker, R. (1997). An iterative spectral-spatial Bayesian labeling approach for unsupervised robust change detection on remotely sensed multispectral imagery. Dans *Proc. Computer Analysis of Images and Patterns (CAIP)*, Kiel, Allemagne.
- Yang, F. et Lishman, R. (2003). Land cover change detection using gabor filter texture analysis. Dans *Proc. of International on texture analysis and synthesis*, pages 78–83.
- Zammit, O. (2008). *Détection de zones brûlées après un feu de forêt à partir d'une seule image satellitaire SPOT5 par techniques SVM*. Thèse de doctorat, Université de Nice, Sophia Antipolis.
- Zammit, O., Descombes, X., et Zerubia, J. (2008). Combining one-class support vector machines and hysteresis thresholding : application to burnt area mapping. Dans *Proc. European Signal Processing Conference (EUSIPCO)*, Lausanne, Suisse.
- Zitová, B. et Flusser, J. (2003). Image registration methods : a survey. *Image and Vision Computing*, 21(11) :977–1000.

Publications et activités de recherche

Publications dans des conférences avec actes

- A. Fournier, X. Descombes, et J. Zerubia. Vers une détection et une classification non-supervisées des changements inter-images. Dans *Proc. Traitement et Analyse de l'Information - Méthodes et Applications (TAIMA)*, Hammamet, Tunisie, 2007.
- A. Fournier, P. Weiss, L. Blanc-Féraud, et G. Aubert. A contrast equalization procedure for change detection algorithms : applications to remotely sensed images of urban areas. Dans *International Conference on Pattern Recognition (ICPR)*, Tampa, États-Unis, 2008.

Publication invitée dans des conférences avec actes

- A. Fournier, X. Descombes, et J. Zerubia. Mixing geometric and radiometric features for change classification. Dans *Proc. SPIE Symposium on Electronic Imaging*, San Jose, États-Unis, 2008.

Publication dans un journal

- P. Weiss, A. Fournier, L. Blanc-Féraud, et G. Aubert. On the illumination invariance of the level lines under directed light, application to change detection. *soumis*.

Intervention invitée dans un workshop

- Using change analysis for target recognition in high resolution data. *CNES/DLR Workshop on Information Extraction and Scene Understanding for Meter Resolution Images*, Oberpfaffenhofen, Allemagne 28 - 30 Mars 2007

Détection et classification de changements sur des scènes urbaines en télédétection

Cette thèse aborde le problème de la détection de changements sur des images de scènes urbaines en télédétection. Les expériences ont été menées sur des couples d'images satellitaires panchromatiques haute résolution (< 1 m).

À travers ce thème général, plusieurs problématiques, correspondant aux divers niveaux d'une chaîne de traitement, sont abordés, depuis la création d'un masque de changements jusqu'au raisonnement à un niveau objet.

Dans ce manuscrit, nous abordons premièrement le problème de la détermination d'un masque de changements. Après avoir étudié les limites d'un algorithme de détection de changements, fondé sur l'analyse en composantes principales, nous proposons un algorithme tirant parti de l'invariance des lignes de niveau, fondé sur un modèle d'illumination et des hypothèses sur la régularité de la scène.

Par la suite, nous abordons la classification des zones détectées comme changées au cours de l'étape précédente. D'abord, nous nous fondons uniquement sur les radiométries des couples de pixels. Enfin, nous étudions l'intérêt d'une composante géométrique dans la classification. Plus précisément, nous appliquons un algorithme d'approximation polygonale sur les zones connexes issues de la classification précédentes, puis nous classifions les formes obtenues compte tenu des orientations des côtés des polygones obtenus.

Mots Clés : détection de changements, images satellitaires, lignes de niveau, classification, k-moyennes, zones urbaines, statistiques directionnelles.

This thesis addresses the problem of change detection on remotely sensed urban scenes. The experiences were run on couples of high resolution (< 1 m) panchromatic satellite images.

Through this general theme, different problems, corresponding to different levels of a processing chain were addressed, from the determination of a change mask to an object level reasoning.

In this work, we first address the problem of determining a change mask. We study the assets and limits of a change detection algorithm based on a Principal Component Analysis. We then propose a new algorithm that relies on the invariance of the level lines. It is based on a simple illumination model and some hypotheses on the scene regularity.

Then we address the classification of the zones detected as changed during our first step. This is done by only considering the radiometries of each pixel couple. Finally, we study the interest of a geometric component in our classification. More precisely, we apply a polygonal approximation algorithm on the connected zones generated by the first classification, then we classify the obtained shapes according to the orientations of the polygon edges.

Keywords: change detection, satellite images, level lines, urban areas classification, K-means, directional statistics.

