



HAL
open science

Classification d'images satellitaires hyperspectrales en zone rurale et périurbaine

Olivier Pony, Xavier Descombes, Josiane Zerubia

► **To cite this version:**

Olivier Pony, Xavier Descombes, Josiane Zerubia. Classification d'images satellitaires hyperspectrales en zone rurale et périurbaine. [Rapport de recherche] RR-4008, INRIA. 2000, pp.64. inria-00072636

HAL Id: inria-00072636

<https://inria.hal.science/inria-00072636>

Submitted on 24 May 2006

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.

Classification d'images satellitaires hyperspectrales en zone rurale et périurbaine

Olivier Pony — Xavier Descombes — Josiane Zerubia

N° 4008

Septembre 2000

THÈME 3



*Rapport
de recherche*

Classification d'images satellitaires hyperspectrales en zone rurale et périurbaine

Olivier Pony , Xavier Descombes, Josiane Zerubia

Thème 3 —Interaction homme-machine,
images, données, connaissances
Projet Ariana

Rapport de recherche n° 4008 —Septembre 2000 —64 pages

Résumé : L'observation satellitaire en zone rurale et périurbaine fournit des images hyperspectrales exploitables en vue de réaliser une cartographie ou une analyse du paysage. Nous avons appliqué une classification par maximum de vraisemblance sur des images de zone agricole. Afin de régulariser la classification, nous considérons la modélisation d'image par champs de Markov, dont l'équivalence avec les champs de Gibbs nous permet d'utiliser plusieurs algorithmes itératifs d'optimisation : l'ICM et le recuit simulé, qui convergent respectivement vers une classification sous-optimale ou optimale pour une certaine énergie. Un modèle d'énergie est proposé : le modèle de Potts, que nous améliorons pour le rendre adaptatif aux classes présentes dans l'image. L'étude de la texture dans l'image initiale permet d'introduire des critères artificiels qui s'ajoutent à la radiométrie de l'image en vue d'améliorer la classification. Ceci permet de bien segmenter les zones périurbaines, la forêt, la campagne, dans le cadre d'un plan d'occupation des sols. Trois images hyperspectrales et une vérité terrain ont été utilisées pour réaliser des tests, afin de mettre en évidence les méthodes et le paramétrage adéquats pour obtenir les résultats les plus satisfaisants.

Remerciements : Ce travail a été en partie financé par Astrium (Matra Marconi Space). Les auteurs remercient Umberto Polverini et Laurent Gautret pour les échanges stimulants, et pour avoir fourni des images hyperspectrales et une vérité terrain.

Mots-clés : télédétection, image satellitaire hyperspectrale, classification supervisée, champs de Markov, champs de Gibbs, modèle de Potts, recuit simulé, analyse de texture, plan d'occupation des sols

Hyperspectral satellite image classification in rural and semiurban areas

Abstract: Satellite observation in rural and semiurban areas provides hyperspectral images which enable us to make a map or an analysis of the landscape. Herein, we applied a maximum likelihood classification on agricultural images. In order to improve this procedure, it is possible in each pixel to use contextual information. Thus, we consider Markov random fields image modeling. The equivalence between Markov and Gibbs fields allows us to use some iterative algorithms of optimisation : ICM and simulated annealing, which converge respectively towards a suboptimal or an optimal classification for a given energy. An energy model is proposed : the Potts model, which can be improved to be adaptive to the classes defined in the image. Texture analysis on the initial image is used to introduce artificial criteria, added to the original image, in order to improve classification. This proves to be useful for segmenting semiurban regions, forests, and the countryside, within the framework of a land-use plan. We use three hyperspectral images and a ground truth to carry out tests, in order to highlight the best methods and parameter setting to obtain the most satisfactory results.

Acknowledgements: This work has been partly funded by Astrium (Matra Marconi Space). The authors would like to thank Umberto Polverini and Laurent Gautret for stimulating discussions and for providing the data, including ground truth.

Key-words: remote sensing, hyperspectral satellite image, supervised classification, Markov random fields, Gibbs random fields, Potts model, simulated annealing, texture analysis, land-use plan

Table des matières

1	Introduction	5
1.1	L'image satellitaire	5
1.2	La classification d'image hyperspectrale	6
1.3	La régularisation	6
1.4	La texture	7
2	Classification d'une image hyperspectrale	8
2.1	Principe, définitions	8
2.1.1	L'espace des états	8
2.1.2	Les zones d'apprentissage	8
2.1.3	L'apprentissage	9
2.1.4	Le classifieur	9
2.2	Classification par maximum de vraisemblance	9
2.2.1	Notations	9
2.2.2	Hypothèse gaussienne	10
2.2.3	Hypothèse d'indépendance conditionnelle des probabilités	10
2.2.4	Ecriture du problème	10
2.3	Mise en pratique	11
2.3.1	Algorithme de classification	11
2.3.2	Présentation de deux images et classification	13
2.4	Limites de l'approche	17
3	Régularisation par champs de Markov	19
3.1	Position du problème	19
3.2	Champs de Markov - Champs de Gibbs	19
3.2.1	Notations, définitions	19
3.2.2	Application	21
3.3	Première approche	22
3.3.1	Le modèle de Potts	22
3.3.2	L'ICM	23
3.4	Un algorithme convergent : le recuit simulé	23
3.4.1	L'algorithme de Métropolis	23
3.4.2	Le recuit simulé	24
3.5	Mise en œuvre	24

4	Application à quelques images	26
4.1	Critères d'évaluation	26
4.1.1	La vérité terrain	26
4.1.2	Autres indicateurs d'erreurs	26
4.2	Régularisation : Topologie et géométrie	28
4.2.1	Voisinage en 4-connexité et 8-connexité	28
4.2.2	Topologie à trous	29
4.2.3	Le paramètre β	30
4.3	Régularisation : ICM et recuit	32
5	Des paramètres de régularisation adaptatifs	36
5.1	Motivations	36
5.2	Formulation	36
5.3	Application	38
5.4	Problème d'automatisation	39
6	Des critères de texture	42
6.1	Différents modèles de texture	43
6.1.1	Les matrices de cooccurrence	43
6.1.2	Les filtres de Gabor	45
6.1.3	Un modèle fondé sur la variance locale	46
6.2	Résultats pour un plan d'occupation des sols	49
6.3	Résultats sur l'image de campagne	51
6.4	Remarques sur la texture	54
7	Pondération des critères	55
7.1	Première approche : déplacement des frontières de décision	55
7.2	Deuxième approche : fission et fusion de critères	56
7.2.1	Principe	56
7.2.2	Choix de bandes discriminantes et non corrélées	57
7.2.3	Découpage du spectre	58
7.3	Synthèse	58
8	Conclusion et perspectives	61

1 Introduction

L'agriculture de précision est une technique prometteuse en pleine expansion. Elle consiste en une cartographie des parcelles agricoles, qui peut se faire de manière automatique par télédétection. Ainsi, les grands groupes agro-industriels et les entreprises spécialisées dans les satellites d'observation de la Terre empruntent ensemble cette nouvelle voie. Or, l'imagerie satellitaire, qui a vu le jour au début des années 60, est le vecteur qui relie les différents acteurs de tels projets.

1.1 L'image satellitaire

Une image satellitaire naît d'un capteur situé sur le satellite en question, et d'une scène observée. Un capteur est caractérisé par un certain nombre de bandes, de 1 à 200, dont chacune correspond à une longueur d'onde. Le capteur synthétise pour chaque bande une image que l'on peut représenter en niveaux de gris et dont la résolution est de quelques mètres par pixel. La résolution d'un capteur étant inversement proportionnelle à sa taille, donc à son coût, l'idée d'utiliser la signature spectrale d'un pixel a permis, à moindre coût, d'augmenter la quantité d'information captée [15].

Le terme *multispectral* qualifie une image dotée de plusieurs bandes. Une image couleur classique est typiquement multispectrale, avec trois bandes. Une image *hyperspectrale* est une image pour laquelle nous disposons d'un spectre continu pour chaque pixel. En pratique, une image hyperspectrale est une image qui contient un très grand nombre de bandes, plusieurs dizaines ou centaines. Ici, nous emploierons le terme "hyperspectral" comme terme général.

Le but de l'observation de la Terre par un capteur multibandes est de recueillir un certain nombre de données sur un paysage. Nous nous intéresserons plus particulièrement aux paysages ruraux et périurbains. Ce sont essentiellement des zones agricoles, boisées, montagneuses, ou maritimes. En zone agricole, une image satellitaire représente des champs cultivés, et éventuellement quelques habitations. Or, la chlorophylle, caractéristique de la végétation, émet abondamment dans l'infra-rouge, et surtout dans le proche infra-rouge. Dans un souci de recueillir l'information la plus utile, le spectre couvert par les bandes des capteurs est toujours plus étendu que le spectre visible.

Par exemple, le satellite d'observation SPOT 2 comporte trois bandes. La première bande se situe entre 500 et 580 μm , ce qui correspond au vert. La bande 2, qui correspond au rouge est entre 610 et 680 μm . Enfin, une bande dans l'infra-rouge est sensible aux longueurs d'ondes entre 790 et 910 μm .

1.2 La classification d'image hyperspectrale

L'image en soi est inutilisable en pratique, sans traitement supplémentaire pour la rendre exploitable. L'objectif du traitement d'image satellitaire est d'en extraire le maximum d'information qui intéresse le futur utilisateur de l'image, et d'évacuer tout ce qui est superflu. Un but réaliste est la classification des cultures sur une image satellitaire. A chaque point de l'image, on souhaite pouvoir attribuer une étiquette parmi une collection définie auparavant. Cette collection d'étiquettes est ce que l'on appelle l'ensemble des classes. Un ensemble de classes pourrait être : ville, campagne, mer, montagne. En zone agricole, un ensemble de classes serait : blé, orge, soja, maïs, forêt, route.

Les données hyperspectrales permettent en chaque pixel de réaliser une étude de son spectre de réflectance. Il est souvent nécessaire de traiter l'image pour corriger les déformations liées aux capteurs, et les phénomènes atmosphériques et éventuellement topographiques. L'image traitée présente l'avantage de ne plus dépendre de phénomènes externes, ni des instruments de mesure, en théorie. La signature spectrale de chaque pixel est souvent caractéristique d'une certaine classe, et donc l'étiquetage peut se faire.

Il existe deux grandes familles de techniques de classification d'images : supervisées et non supervisées. La classification non supervisée ne demande aucune connaissance *a priori* de l'utilisateur. Soit il existe une collection universelle de classes déjà définies indépendamment de l'image, soit le nombre de classes et leurs caractéristiques sont définies automatiquement lors de la classification. Loin de cette automatisation dont quelques techniques ont déjà été développées, mais dont la mise en œuvre à grande échelle sur des données réelles est difficile, nous nous limiterons ici à une classification supervisée. Celle-ci demande à un utilisateur d'instruire le système en désignant des zones de l'image comme étant des échantillons représentatifs des classes à extraire. La classification est donc précédée d'un apprentissage.

Pour instruire le système, encore faut-il avoir une bonne connaissance des cultures du terrain observé. Pour cela, il nous faut soit savoir repérer visuellement la nature des cultures, soit disposer d'une vérité terrain. Une vérité terrain est un ensemble de données garanties exactes sur les cultures observées. Elle est réalisée au sol, par une observation rapprochée du terrain. C'est un outil pour réaliser un bon apprentissage, ainsi que pour valider une classification. Il est suffisant de disposer d'une vérité terrain sur une portion de l'image où tous les types de terrain sont représentés.

1.3 La régularisation

Il existe déjà des logiciels commerciaux, comme ENVI [3], qui implémentent plusieurs méthodes de classification. Cependant, les spécialistes ne sont pas toujours satisfaits du

résultat d'une classification. En effet, celle-ci est fondée sur l'analyse spectrale d'un pixel, et certains types de terrains ont des réponses spectrales voisines. On détecte alors dans l'image classifiée des aberrations, qui ne se font sentir que lorsque l'on dispose d'une vue globale. Une fois le pixel replacé dans son contexte, nous savons dire si la classe attribuée est réaliste ou non. L'information contextuelle d'un pixel est donc un outil de correction.

La régularisation est la procédure qui consiste à corriger une image classifiée pour obtenir une structure acceptable. Une image de zone rurale est composée généralement de grandes parcelles homogènes, et une image classifiée acceptable doit respecter cette propriété. Ainsi, la régularisation insistera sur la cohérence entre la classe d'un pixel et celle de ces voisins. Les champs de Markov consistent à tenir compte de l'influence du voisinage d'un point sur ce dernier. C'est donc tout naturellement cet outil mathématique que nous utiliserons pour régulariser la classification d'image.

1.4 La texture

L'information contextuelle d'un pixel présente un intérêt pour corriger la classification, mais un autre type d'information contextuelle est exploitable dans l'image hyperspectrale initiale : la texture. Un paysage de forêt ou une zone urbaine ne possèdent pas toujours une réponse radiométrique simple à caractériser. La ville est un mélange de divers matériaux aux spectres de réflectance variés. La surface d'une forêt est irrégulière et ne renvoie pas la lumière solaire de la même manière d'un point à l'autre. Par contre, à certaines échelles, ces paysages semblent avoir une texture homogène. On arrive nettement à différencier visuellement les textures des champs, des forêts et des villes.

Pour parvenir à exploiter la texture des images, il existe divers modèles qui permettent de caractériser ces textures. L'étude radiométrique du voisinage de chaque pixel permet d'extraire de l'image initiale une nouvelle bande qui localise les régions texturées. Ces bandes constituent de nouveaux critères qui s'ajoutent à la signature spectrale des pixels pour améliorer la classification.

2 Classification d'une image hyperspectrale

2.1 Principe, définitions

Nous avons repris ici la description et le vocabulaire introduits par Pierre Soille dans le chapitre sur la classification issu de [24].

Le principe de la classification d'image est d'attribuer à chaque point une classe choisie parmi une collection de classes possibles. Si le nombre de classes et leurs règles d'appartenance sont fixées au départ, on parle de classification supervisée. Si, par contre, on désire réaliser une classification sans connaissance *a priori* du nombre de classes ou de leurs caractéristiques, on effectue alors une classification dite non supervisée.

Dans notre cas, nous effectuons une classification supervisée. Celle-ci s'organise en quatre étapes

2.1.1 L'espace des états

De l'image hyperspectrale, on extrait des critères jugés suffisamment discriminants pour la classification à suivre. Le niveau de gris dans une bande de l'image est un critère possible. Cet ensemble définit un espace, dont la dimension est égale au nombre de critères extraits. A chaque pixel de l'image correspond un vecteur de cet espace.

Dans un premier temps, on identifiera l'ensemble des critères à l'ensemble des bandes de l'image.

2.1.2 Les zones d'apprentissage

On définit sur l'image des zones d'apprentissage, qui sont des régions de l'image que l'on juge représentatives d'une certaine classe. Par exemple, en vue de classer une image de zone agricole, on sélectionnera un champ de blé pour réaliser l'apprentissage de la classe "blé".

Le choix des régions d'apprentissage mérite beaucoup d'attention, même si dans ce rapport nous n'y consacrons que peu de temps. Il ne suffit pas toujours de choisir une seule parcelle par classe. En effet, il se peut qu'une classe telle que "forêt" contienne plusieurs types de paysages correspondant à divers types d'arbres. Dans ce cas, il faut soit créer des sous-classes, ce qui complexifie les calculs, soit définir au moins une région d'apprentissage par type de paysage. En outre, il se peut que malgré la correction de l'image, deux paysages semblables aient des spectres de réflectance légèrement différents, à cause d'une exposition au soleil différente.

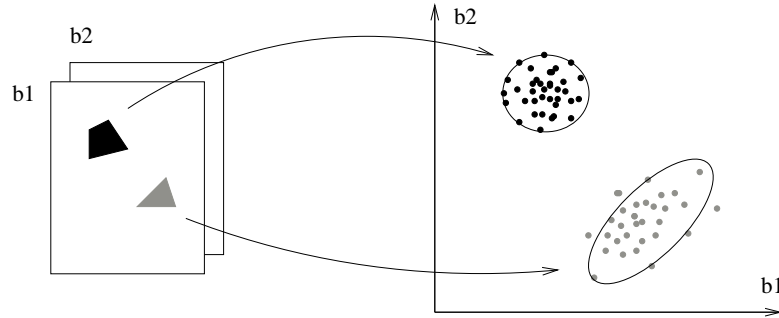


FIG. 1 – Image hyperspectrale (2 bandes) et espace des états

2.1.3 L'apprentissage

Il consiste, pour une classe donnée, à rassembler l'ensemble des pixels situés dans ses zones d'apprentissage, et d'étudier la répartition de leurs vecteurs associés dans l'espace des états. Ceux-ci sont idéalement situés dans la même zone de l'espace. Dans la figure 1, on n'a sélectionné que deux critères ou bandes pour simplifier la représentation.

Il est raisonnable, par exemple, de calculer pour chaque classe un vecteur moyen, et éventuellement les statistiques d'ordres supérieurs, telles que la matrice de covariance.

2.1.4 Le classifieur

La classification est le problème inverse de l'apprentissage. Le classifieur utilise l'information issue de l'apprentissage pour attribuer à chaque point de l'image une catégorie ou classe.

Le but du problème de la classification est de trouver des séparations *ad hoc*, ou frontières de décision, entre les régions caractéristiques des classes dans l'espace des états, qui minimisent les erreurs de classification. Un bon choix initial de critères facilite cette étape.

2.2 Classification par maximum de vraisemblance

2.2.1 Notations

Une image est constituée d'un ensemble S de sites s_i ou pixels. Soit Λ l'ensemble des classes parmi lesquelles on choisit les étiquettes de chaque pixel. Une classe sera notée λ_i .

L'espace des états est noté E , sa dimension est égale au nombre de critères sélectionnés.

Soit y_s le vecteur de E associé au site s de l'image S .

Soit $y = (y_{s_1}, \dots, y_{s_{|S|}})$.
 Soit x_s la classe dans Λ attribuée à s .
 Soit $x = (x_{s_1}, \dots, x_{s_{|S|}})$.
 Y_s est le processus aléatoire associé à y_s .
 Y est le processus aléatoire associé à y .
 X_s est le champ aléatoire associé à x_s .
 X est le champ aléatoire associé à x .

2.2.2 Hypothèse gaussienne

Dans un but de simplification, nous allons effectuer l'hypothèse suivante : Pour chaque classe λ_i , la loi de probabilité $P(y_s/x_s = \lambda_i)$ est gaussienne.

Ainsi, chaque classe λ_i est définie par un vecteur moyenne μ_i et une matrice de covariance Σ_i dans l'espace des états E . L'apprentissage consistera à approximer les statistiques de chaque classe par les statistiques sur les régions d'apprentissage correspondantes. On calculera donc les vecteurs moyennes et les matrices de covariance dans l'espace des états, pour chacune des classes.

2.2.3 Hypothèse d'indépendance conditionnelle des probabilités

Dans un premier temps, on suppose que la réponse spectrale en un site sachant la classe correspondante est indépendante de la réponse spectrale aux sites voisins.

$$P(Y = y/X = x) = \prod_s P(Y_s = y_s/X_s = x_s)$$

Cette hypothèse peut être erronée dans le cas d'une image, qui est une représentation d'une scène continue. La radiométrie d'un pixel peut dépendre de celle de ses voisins. Nous verrons ceci de plus près dans le chapitre suivant.

2.2.4 Ecriture du problème

Sous ces deux hypothèses, nous cherchons à déterminer la meilleure classification des pixels de l'image, x_{opt} . Pour cela, on dispose de l'état global de l'image, y , et on cherche à déterminer :

$$x_{opt} = \arg \max_x P(X = x/Y = y)$$

$$x_{opt} = \arg \max_x \frac{P(Y = y/X = x)P(X = x)}{P(Y = y)}$$

$$x_{opt} = \arg \max_x P(Y = y/X = x)P(X = x)$$

En supposant que l'on ne dispose d'aucune information *a priori* sur les probabilités relatives d'occurrence des classes, $P(X_s = \lambda)$ est constant, pour tout s et tout λ .

$$x_{opt} = \arg \max_x P(Y = y/X = x)$$

$$x_{opt} = \arg \max_x \prod_s P(Y_s = y_s/X_s = x_s)$$

Le problème global est donc ramené à un problème d'optimisation local, grâce à l'hypothèse d'indépendance conditionnelle des pixels.

$$x_{s\ opt} = \arg \max_{x_s} P(Y_s = y_s/X_s = x_s)$$

Et sous l'hypothèse gaussienne, on obtient l'équation suivante :

$$x_{s\ opt} = \arg \max_{\lambda_i \in \Lambda} \frac{\exp\left(-\frac{1}{2}(y_s - \mu_i)^T [\Sigma_i]^{-1} (y_s - \mu_i)\right)}{\det(\Sigma_i)^{\frac{1}{2}} \cdot (2\pi)^{\frac{N}{2}}}$$

où N est la dimension de l'espace des états, soit le nombre de critères choisis. Pour des raisons pratiques, on préférera calculer le logarithme de l'expression ci-dessus.

$$x_{s\ opt} = \arg \min_{\lambda_i \in \Lambda} \left((y_s - \mu_i)^T [\Sigma_i]^{-1} (y_s - \mu_i) + \log \det(\Sigma_i) \right)$$

2.3 Mise en pratique

2.3.1 Algorithme de classification

A la lumière des paragraphes précédents il est à présent aisé d'envisager un premier programme de classification d'image hyperspectrale. Les entrées de ce programme sont :

- une image hyperspectrale à Nb bandes,
- un masque de régions d'apprentissage.

En sortie, ce programme produit :

- une image de classification,
- pour chaque classe, une carte de vraisemblance d'appartenance de chaque pixel à la classe en question.

Le schéma en figure 2 résume la classification par maximum de vraisemblance.

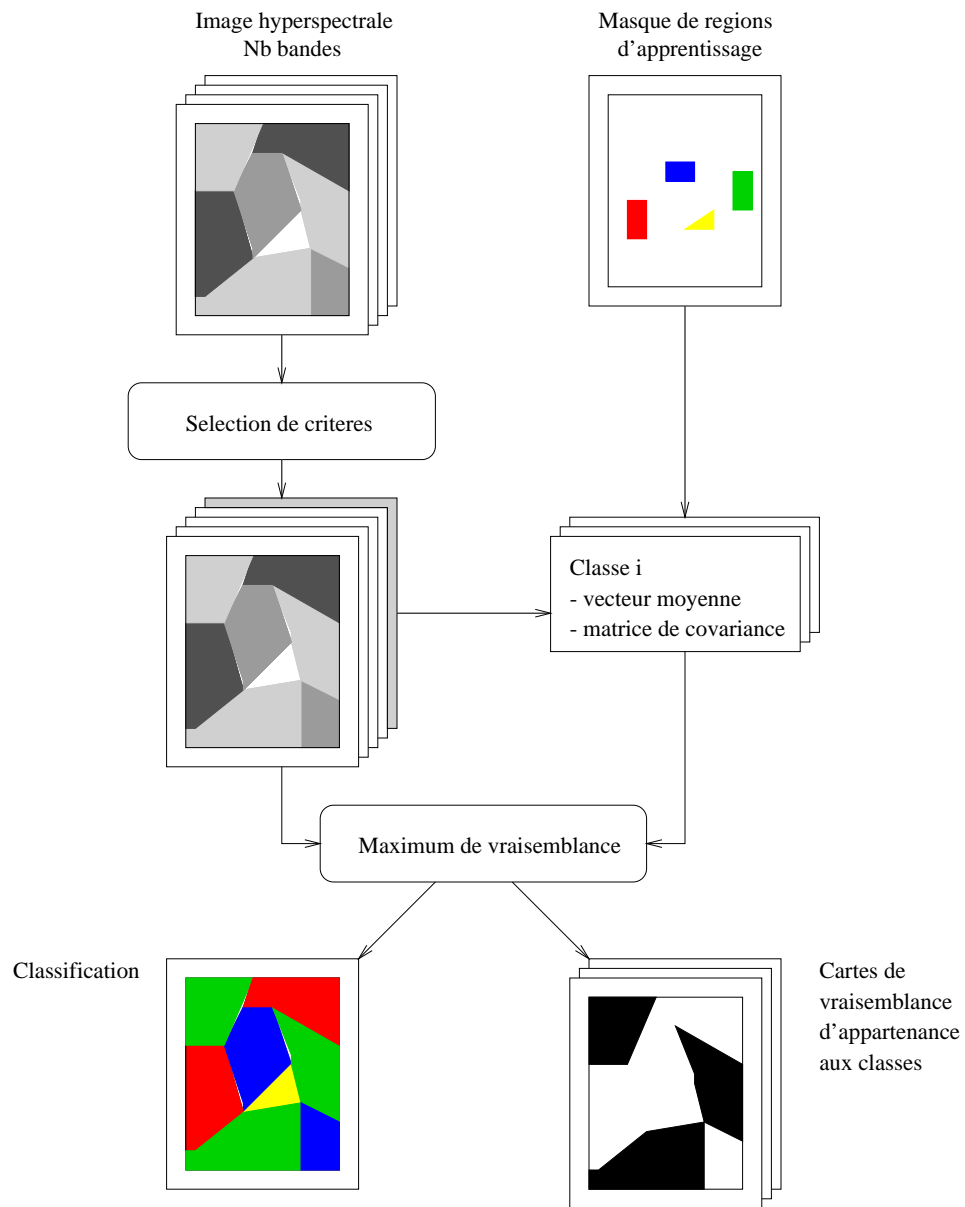


FIG. 2 – Classification par maximum de vraisemblance

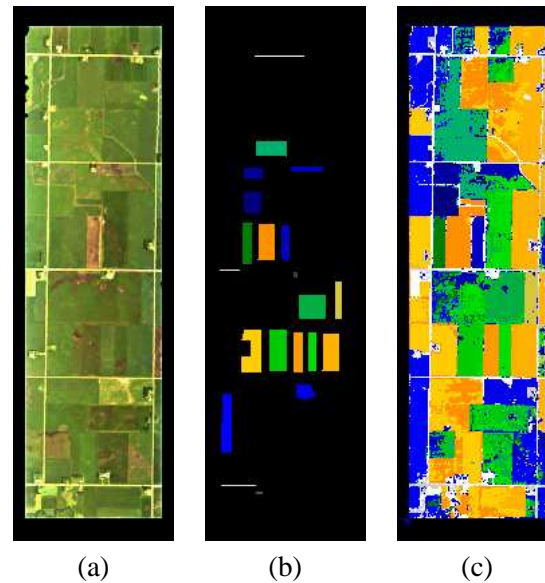


FIG. 3 – Classification sur l'image 1 (US) - (a) Image originale - (b) Régions d'apprentissage - (c) Image classifiée

2.3.2 Présentation de deux images et classification

La première image est une image de zone agricole américaine avec une résolution horizontale et verticale de 20 mètres par pixel. Elle comporte 11 bandes, allant du bleu ($0,44 \mu\text{m}$) au proche infra-rouge ($0,90 \mu\text{m}$). Le capteur est un capteur CASI avec 48 bandes de 10 nm, mais l'image est simulée XSTAR, qui est un projet de capteur à 11 bandes utilisé pour le "pilotage" agricole. Voir figure 3.

A gauche se trouve l'image originale, dont trois canaux sur onze ont été sélectionnés pour la coloriser. Au milieu, les régions d'apprentissage, et à droite la classification par maximum de vraisemblance. Ce sont 11 critères correspondant précisément aux 11 bandes que l'on a utilisées pour la classification.

Les zones oranges sont des champs de blé, avec une distinction entre trois types de blé. A cela s'ajoutent trois types de soja, en vert. La classe route est divisée en trois sous-classes blanches ou grises. Les teintes bleues sont associées à des classes diverses que l'on ne cherche pas à identifier. Les régions d'apprentissage sont relativement petites par rapport à l'image, mais cela n'empêche pas une bonne classification si les échantillons sont représentatifs.

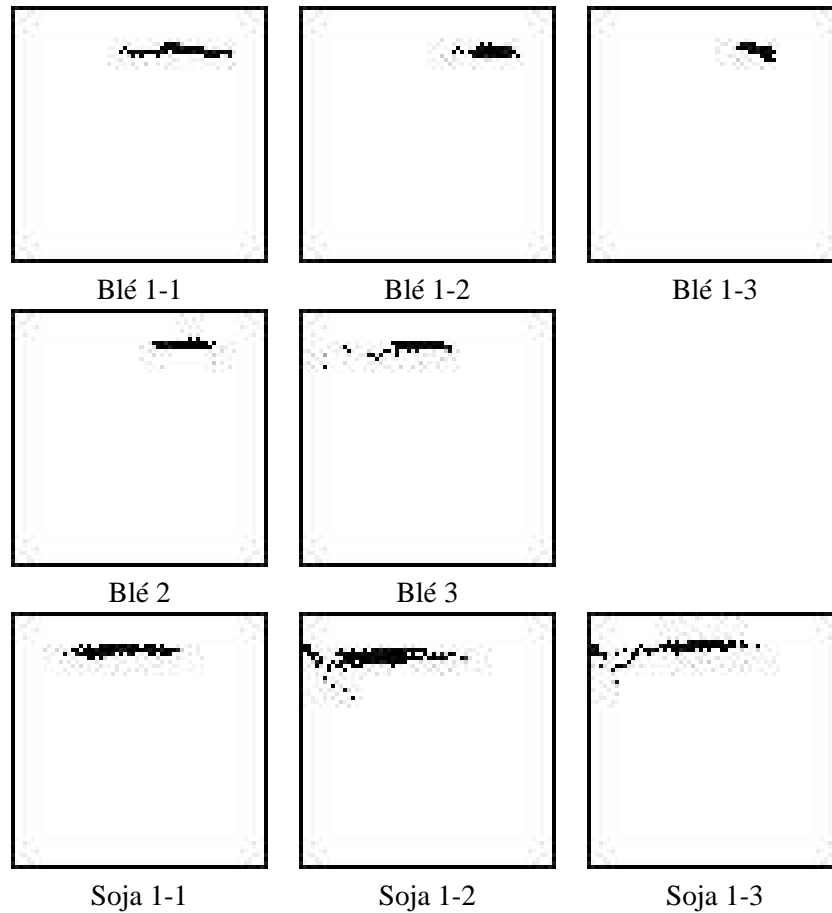


FIG. 4 – *Projection de l'espace des états pour les classes de l'image US*

Nous avons représenté sur les figure 4 et 5 une projection sur deux bandes, l'une dans le visible, l'autre dans le proche infra-rouge, des nuages de chaque classe dans l'espace des états. Cela donne une idée de la facilité ou de la difficulté de bien séparer les classes. On voit par exemple qu'à l'aide des deux canaux sélectionnés, on peut aisément séparer la route des autres classes. Les classes "divers" sont relativement bien localisées. Par contre, les classes "blé" et "soja" peuvent avoir tendance à se mélanger.

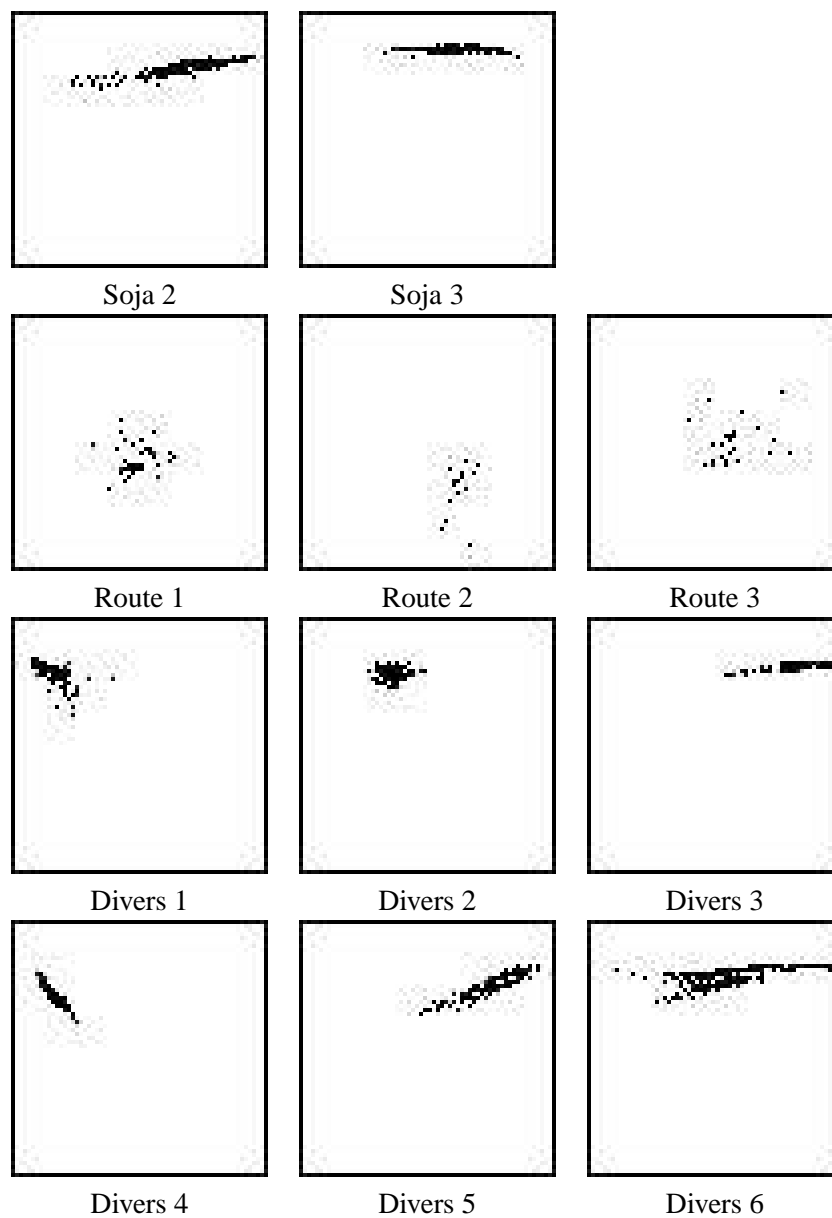


FIG. 5 – Projection de l'espace des états pour les classes de l'image US

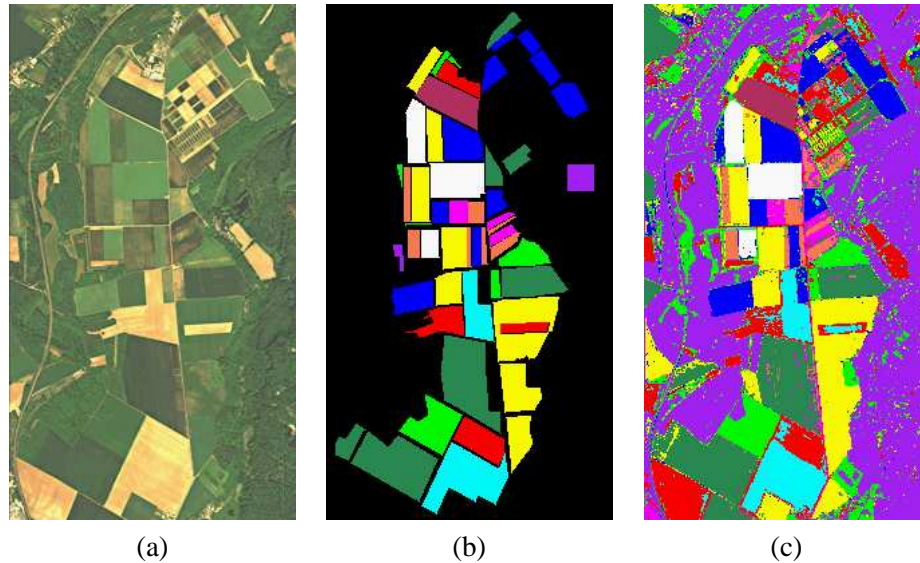


FIG. 6 – Classification sur l’image 2 (environs de Boigneville) - (a) Image originale - (b) Régions d’apprentissage - (c) Image classifiée

La deuxième image représente une zone agricole et forestière proche de Boigneville dans l’Essonne. Elle comporte 10 bandes, du bleu ($0,44 \mu m$) au proche infra-rouge ($0,82 \mu m$). Sa résolution est de 10 mètres par pixel. Le capteur est un capteur MIVIS 20 bandes de 20 nm. L’image a été simplifiée à 10 bandes (voir figure 6).

On y trouve diverses cultures : blé, maïs, orge, colza, pois, betteraves. Il existe parfois une distinction entre les cultures d’hiver et celles de printemps. Deux classes ont aussi été réservées pour les terres en jachère et la forêt. Les régions d’apprentissage sont ici très grandes. Même si, en pratique, on ne réalise l’apprentissage que sur une petite portion de l’image, nous disposons ici d’une grande quantité de données sur le terrain que nous exploiterons lors de l’estimation des erreurs de classification.

En premier lieu, nous n’avons pas réalisé d’apprentissage spécifique pour les routes et les habitations. La classification attribue donc la classe qu’il juge la plus probable le long des routes, ce qui inclut quelques aberrations dans l’image finale. La classe “route” sera introduite tardivement dans le chapitre 5. Comme pour l’image précédente, nous avons projeté l’espace des états sur deux bandes pour chacune des régions d’apprentissage. On se rend compte que l’hypothèse gaussienne peut être critiquée pour certaines classes telles que le blé (voir figure 7).

2.4 Limites de l'approche

Dans cette partie, nous avons fait l'hypothèse de l'indépendance d'un pixel vis-à-vis de son environnement. A la vue des deux exemples d'images classifiées, on se rend facilement compte des défauts de cette approche.

Visuellement, les images de classification présentent souvent des points isolés au milieu de vastes zones homogènes. Or, il est inconcevable qu'un type de culture se mélange à un champ d'une culture différente. Devant cette situation, on saisit l'intérêt de traiter cette image en tenant compte en chaque pixel de la classe de ses voisins. C'est pourquoi nous avons utilisé une modélisation par champs de Markov, décrite notamment dans l'article de Geman et Graffigne [11].

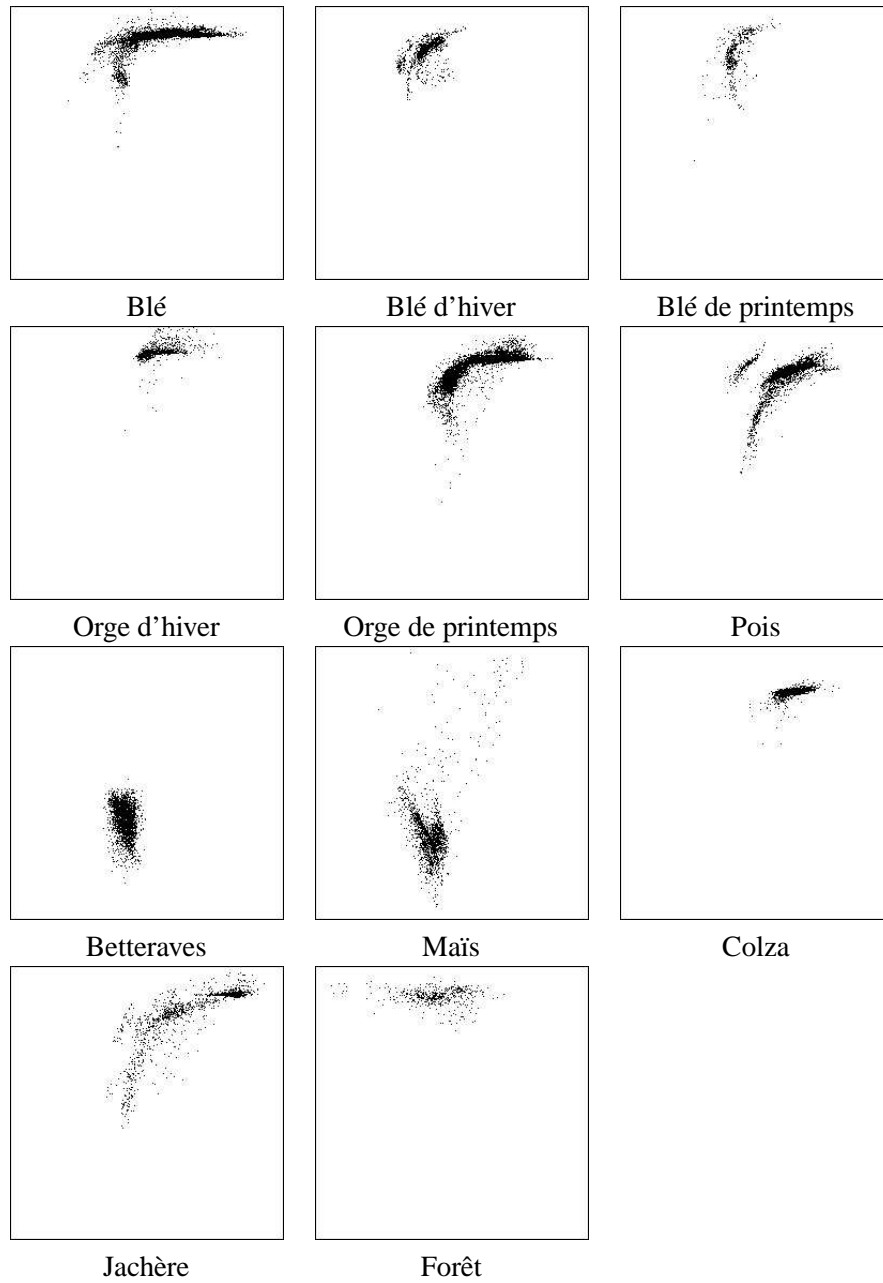


FIG. 7 – Projection de l'espace des états pour les classes de l'image de Boigneville

3 Régularisation par champs de Markov

Le lecteur intéressé peut également se reporter aux ouvrages de G. Winkler [27], S.Z. Li [17], et J.P. Cocquerez et S. Philipp [5] pour plus de détails sur l'utilisation des champs de Markov en traitement d'image.

3.1 Position du problème

Une première approche de la régularisation d'image est celle des opérateurs morphologiques utilisés par le logiciel de traitement d'image ENVI: *sieve* et *clump*. Le premier consiste à déclasser les ensembles connexes de pixels de même classe de taille inférieure à un certain seuil. La relation de connexité entre deux pixels est liée au choix entre un système de voisinage en 4-connexité et en 8-connexité (voir figure 8). Le deuxième opérateur est une fermeture sur l'image, qui comble une partie des pixels non classés.

Les résultats de ce traitement sont en effet une régularisation de l'image dans le sens où les ensembles de pixels isolés dans la classification disparaissent. Cependant, nous pouvons émettre plusieurs critiques. Il se peut qu'une région de petite taille dans l'image classifiée ne soit pas une erreur, comme un arbre dans un champ par exemple. Inversement, un ensemble assez conséquent peut s'avérer être une erreur de classification.

En plus de l'image classifiée, nous disposons des probabilités ou vraisemblances d'appartenance des pixels aux classes. Il est souhaitable d'employer un modèle qui prenne en compte cette information.

3.2 Champs de Markov - Champs de Gibbs

3.2.1 Notations, définitions

Un système de voisinage V sur S est défini ainsi :

$$\begin{cases} s \notin V_s \\ t \in V_s \Rightarrow s \in V_t \end{cases}$$

Ici, nous nous limiterons aux cas des systèmes en 4-connexité et en 8-connexité. Si l'on identifie les pixels d'une image au couple de leurs coordonnées (i,j) ,

$$V_{i,j}^4 = \{(k,l) \in S / 0 < (i-k)^2 + (j-l)^2 \leq 1\}$$

$$V_{i,j}^8 = \{(k,l) \in S / 0 < (i-k)^2 + (j-l)^2 \leq 2\}$$



FIG. 8 – Voisinage en 4-connexité et en 8-connexité

Une *clique* c est un ensemble non vide de sites vérifiant :

$$\forall (s, t) \in c, (s \neq t \Rightarrow s \in V_t)$$

L'ensemble des cliques est noté \mathcal{C} . Notons que les singletons sont aussi des cliques d'après cette définition.

Dans le système en 4-connexité, les cliques sont de taille 1 ou 2. Dans le système en 8-connexité, les cliques sont de taille 1, 2, 3 ou 4.

Rappelons que nous notons x_s la classe dans l'ensemble des classes Λ attribuée au site s de l'image S . Nous notons désormais x^s la configuration de l'image exceptée le site s . X est le champ aléatoire associé à x .

DÉFINITION 1 (CHAMP DE MARKOV)

X est un champ de Markov sur S relativement à V si et seulement si il vérifie :

$$\begin{cases} \forall x \in \Lambda^{|S|}, P(X = x) > 0 \\ \forall s \in S, P(X_s = x_s / x^s) = P(X_s = x_s / x_t, t \in V_s) \end{cases}$$

Autrement dit, dans un champ de Markov, la probabilité conditionnelle en un site n'est fonction que de la configuration du voisinage de ce site. Cette hypothèse est réaliste pour les images qui nous intéressent.

DÉFINITION 2 (MESURE DE GIBBS)

P est une mesure de Gibbs sur S relativement à V si elle est de la forme :

$$P(X = x) = Z^{-1} \exp - \sum_{c \in \mathcal{C}} U_c(x)$$

où

$$Z = \sum_{x \in \Lambda^{|S|}} \exp \left(- \sum_{c \in \mathcal{C}} U_c(x) \right)$$

$U_c(x)$ est l'énergie d'une clique c , $\sum_{c \in C} U_c(x)$ est l'énergie totale de la configuration x . L'expression de l'énergie des cliques est un problème en soi, nous en donnerons un modèle ultérieurement.

DÉFINITION 3 (CHAMP DE GIBBS)

Le champ de Gibbs de potentiel associé au système de voisinage V est le champ aléatoire X dont la probabilité est une mesure de Gibbs associée au système de voisinage.

$$P(X = x) = Z^{-1} \exp - \sum_{c \in C} U_c(x)$$

Il est hélas presque impossible d'accéder à la valeur de $P(X = x)$ car la constante Z est, en pratique, incalculable. C'est une somme sur toutes les configurations de $\Lambda^{|S|}$. Dans le cas de la première image présentée en fin de chapitre précédent, qui est une petite image (400x120) avec 19 classes, le nombre de classifications possibles est supérieur à 10^{60000} .

3.2.2 Application

Un résultat fondamental est l'équivalence des champs de Markov et des champs de Gibbs grâce au théorème suivant :

THÉORÈME 1 (HAMMERSLEY-CLIFFORD)

Sous les hypothèses :

- S est fini ou dénombrable
- Le système de voisinage V est borné
- L'espace des états E est discret

X est un champ de Markov relativement à V si et seulement si X est un champ de Gibbs de potentiel associé à V .

Ecrivons la probabilité conditionnelle en un site s de S :

$$P(X_s = x_s / X^s = x^s) = \frac{\exp(-U(x_s, x^s))}{\sum_{x_s \in \Lambda} \exp(-U(x_s, x^s))}$$

Définissons l'énergie locale U_s par :

$$U_s(x_s / x_t, t \in V_s) = \sum_{c \in C / s \in c} U_c(x_s, x_t, t \in V_s)$$

Notons désormais cette quantité $U_s(x_s/V_s)$. L'énergie globale $U(x)$ s'écrit :

$$U(x) = \sum_{c \in \mathcal{C}/s \notin c} U_c(x) + U_s(x_s/V_s)$$

On obtient :

$$P(X_s = x_s / X^s = x^s) = \frac{\exp(-U_s(x_s/V_s))}{\sum_{x_s \in \Lambda} \exp(-U_s(x_s/V_s))}$$

Cette expression ne fait intervenir que les potentiels des cliques contenant le site s . Le calcul de la probabilité conditionnelle locale est donc rendu possible. Ceci nous permettra de simuler des champs markoviens.

3.3 Première approche

3.3.1 Le modèle de Potts

Nous sortons du cas général de l'énergie des cliques pour présenter un exemple bien adapté au problème de classification d'image. Cet exemple est le modèle de Potts [23]. On choisit un système de voisinage en 4- ou 8-connexité, avec la simplification qui consiste à considérer que l'énergie d'une clique c , U_c , est non nulle uniquement pour les cliques d'ordre 1 et 2. Sur les cliques d'ordre 2, on a :

$$\begin{aligned} U_{c=(s,t)}(x_s, x_t) &= -\beta \text{ si } x_s = x_t \\ &= +\beta \text{ si } x_s \neq x_t \end{aligned}$$

β est un paramètre positif. Nous verrons son influence sur la régularisation ultérieurement.

Le potentiel sur les cliques d'ordre 1 est un terme d'attache aux données. C'est la log-vraisemblance *a posteriori* $\log(P(x_s/y_s))$. A une constante additive près, sous l'hypothèse d'égalité des probabilités $P(x)$ et $P(y)$, c'est la log-vraisemblance *a priori* $\log(P(y_s/x_s))$. Cette constante sera évacuée dans l'expression de la probabilité conditionnelle $P(x_s/x^s)$. L'expression a été donnée au chapitre précédent :

$$U_s(\lambda) = ((y_s - \mu_\lambda)^T [\Sigma_\lambda]^{-1} (y_s - \mu_\lambda) + \log \det(\Sigma_\lambda))$$

On en déduit l'expression de l'énergie *a posteriori* :

$$U(x/y) = \sum_s ((y_s - \mu_{x_s})^T [\Sigma_{x_s}]^{-1} (y_s - \mu_{x_s}) + \log \det(\Sigma_{x_s})) + \beta \sum_{(s,t) \in \mathcal{C}_2} \delta_{x_s, x_t}$$

où δ_{x_s, x_t} vaut -1 si $x_s = x_t$, 1 sinon.

Cette expression intègre un terme d'attache aux données et un terme de régularisation pondéré par le paramètre β . Plus β est grand, et plus le terme de régularisation est fort. Concrètement, une grande valeur de β favorise le reclassement d'un pixel isolé dans la classe la plus présente parmi ses voisins. Une valeur de β faible nous ramène au problème de classification brute, qui correspondrait à $\beta = 0$, c'est-à-dire à une classification par maximum de vraisemblance.

3.3.2 L'ICM

Une fois choisi le modèle de l'énergie, le but est de trouver la configuration optimale x qui minimise cette énergie. Nous présentons ici une méthode simple qui permet d'obtenir une régularisation de la classification, mais sans atteindre le minimum de l'énergie.

L'ICM est un algorithme de régularisation utilisant les champs de Markov assez intuitif, proposé par Besag [1] [2]. ICM signifie "*Iterated Conditional Mode*".

Il consiste tout simplement à balayer l'ensemble des sites de l'image, et pour chacun des sites, choisir pour x_s la classe λ dans Λ qui minimise l'énergie. Cette opération doit être itérée un certain nombre de fois pour atteindre un état stationnaire. Cette technique est rapide, mais dépend fortement des conditions initiales et du choix de l'ordre de balayage. L'algorithme converge vers le premier minimum local rencontré à partir des conditions initiales.

Il existe cependant des méthodes, plus lentes, pour lesquelles on dispose d'une démonstration de convergence vers un optimum global. Le recuit simulé [25] en est une.

3.4 Un algorithme convergent : le recuit simulé

3.4.1 L'algorithme de Métropolis

L'algorithme de Métropolis [20] [25] [5], élaboré dans les années 50, permet la simulation de champs de Markov. C'est un algorithme de relaxation probabiliste qui consiste, à partir d'une image initiale, à en produire une autre selon la loi d'un champ de Markov donné.

A chaque étape, on choisit un site de l'image, et on tire au hasard une classe λ à lui affecter. Dans le cas de l'ICM, on choisirait le meilleur λ dans Λ , c'est-à-dire celui qui minimise l'énergie de l'image. Dans l'algorithme de Métropolis, on choisit une classe aléatoirement, que l'on affecte selon la règle suivante :

- Si la variation d'énergie de l'image $\Delta U < 0$, alors le changement de classe est accepté,
- Si $\Delta U \geq 0$, le changement est accepté avec la probabilité $p = \exp(-\Delta U)$

3.4.2 Le recuit simulé

Cette méthode [25] est inspirée du phénomène physique du recuit en physique statistique. on intègre un paramètre de température T qui règle la probabilité de changement de classe en cas d'augmentation de l'énergie.

La distribution de Gibbs est ainsi adaptée :

$$P_T(X = x) = \frac{1}{Z(T)} \exp -\frac{U(x)}{T}$$

Avec $T > 0$ et $Z(T)$ constante de normalisation.

Lorsque $T \rightarrow \infty$, P_T converge vers la probabilité uniforme sur $\Lambda^{|\mathcal{S}|}$, toutes les configurations sont équiprobables. Lorsque $T \rightarrow 0$, $P_T(x) = 0$ partout, sauf pour les configurations correspondant aux minima d'énergie de U , qui sont équiprobables.

L'algorithme de recuit simulé *inhomogène* se déroule ainsi :

- Choix d'une température initiale $T^{(0)}$ "grande",
- Choix d'une configuration initiale $x^{(0)}$ quelconque,
- A l'étape n , on simule une configuration $x^{(n)}$ pour la loi de Gibbs d'énergie $\frac{U(x)}{T^{(n)}}$ à partir de la configuration $x^{(n-1)}$ par l'algorithme de Métropolis, et ceci pour tous les pixels de l'image,
- Diminution de la température : $T^{(n)} = k * T^{(n-1)}$, typiquement $k \approx 0,99$,
- Arrêt lorsque le taux de changement est faible (typiquement <1%).

L'algorithme de recuit simulé *homogène* consiste, à température fixée, à itérer le processus jusqu'à stabilité du taux de changement. Uniquement à cette condition, on diminue la température. Dans nos tests, nous emploierons le recuit simulé inhomogène.

Le recuit simulé accepte les remontées en énergie, mais de moins en moins lorsque la température diminue. La convergence de cet algorithme vers le minimum d'énergie globale repose sur cette propriété, mais aussi sur la vitesse de décroissance de la température, qui doit se faire suffisamment lentement (pour une loi de décroissance théorique exacte, se référer à l'ouvrage de Aarts et van Laarhoven [25]).

3.5 Mise en œuvre

Le problème de régularisation se pose de manière plus claire. Une configuration d'énergie minimale est un bon compromis entre la vraisemblance de la classification de chaque pixel, et la régularité de cette classification. Ce compromis s'effectue par une pondération du critère de régularité grâce au paramètre β .

Le paramètre de température T règle la tolérance des configurations désavantageuses dans le processus d'évolution de la classification, par l'algorithme de Métropolis. Grâce à une décroissance adéquate de la température, le recuit simulé, converge vers un minimum global.

On utilise ici les données de vraisemblance d'appartenance de chaque pixel aux classes pour l'attache aux données, et l'image classifiée par maximum de vraisemblance comme état initial dans le recuit simulé. On choisit un critère d'arrêt du recuit car la convergence est lente. Il est ensuite envisageable d'achever la convergence par un ICM, très sensible aux conditions initiales, qui dans ce cas sont favorables.

4 Application à quelques images

4.1 Critères d'évaluation

Dans ce chapitre, nous verrons plusieurs exemples de classification et régularisation d'images. Afin de pouvoir comparer deux résultats, une appréciation visuelle critique permet un premier jugement sur la qualité d'une image par rapport à l'autre. Nous avons tout de même besoin de méthodes plus rigoureuses pour repérer les erreurs, et évaluer la qualité de la classification obtenue.

4.1.1 La vérité terrain

La classification repose sur un apprentissage sur des régions définies par l'utilisateur. Les régions et la classe attribuée sont des informations pertinentes, qui correspondent à la réalité. Nous appelons un tel ensemble de données une *vérité terrain*. Il est alors possible de détecter les erreurs.

Une première possibilité d'expression des erreurs est une image des erreurs. Elle représente l'image des régions d'apprentissage, et affiche de manière visible les pixels où la classe attribuée après classification diffère de son "étiquette" dans la vérité terrain. Un exemple sur une classification brute sans régularisation de l'image de Boigneville est montré sur la figure 9.

Enfin, on peut calculer le pourcentage de pixels mal classés sur l'ensemble des régions dont on a une vérité terrain. Sur le même exemple, on obtient un taux d'erreur de 7,64%. Nous avons calculé ce taux d'erreur sur les images régularisées par les deux opérateurs morphologiques "*sieve*" et "*clump*". Les résultats figurent sur le tableau 11.

4.1.2 Autres indicateurs d'erreurs

Une forme plus utilisable du taux d'erreur est la *matrice de confusion* [19], qui calcule les mêmes statistiques élémentaires sur les erreurs, mais en faisant la différence entre les classes. C'est une matrice carrée de taille égale au nombre de classes $|\Lambda|$, et dont un élément m_{ij} est égal au pourcentage de pixels classés λ_j qui auraient dû appartenir à la classe λ_i .

Normalement, une matrice de confusion a des valeurs proches de 1 sur la diagonale, et des valeurs proches de 0 ailleurs. Idéalement, c'est la matrice identité. A titre d'exemple, la

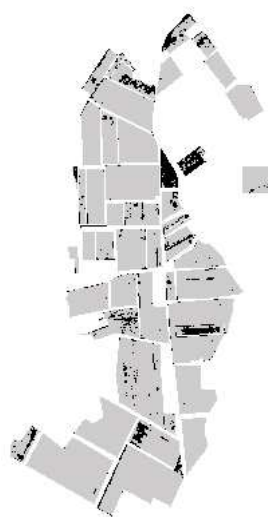


FIG. 9 – Exemple d'image des erreurs

	Erreur (%)
US	0,73
Boigneville	7,64

FIG. 10 – Taux d'erreur après une simple classification par maximum de vraisemblance

	Erreur (%)
US	0,15
Boigneville	6,49

FIG. 11 – Taux d'erreur après régularisation par les opérateurs morphologiques "sieve" et "clump"

matrice de confusion relative à la classification sur l'image de Boigneville est la suivante :

$$\begin{pmatrix} 0.87 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.07 & 0.00 & 0.04 & 0.00 \\ 0.00 & 0.97 & 0.00 & 0.00 & 0.00 & 0.01 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.97 & 0.00 & 0.00 & 0.00 & 0.01 & 0.00 & 0.01 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.95 & 0.00 & 0.05 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.01 & 0.00 & 0.98 & 0.00 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.01 & 0.00 & 0.23 & 0.00 & 0.74 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.01 & 0.02 & 0.00 & 0.00 & 0.00 & 0.01 & 0.95 & 0.00 & 0.00 & 0.00 & 0.01 \\ 0.04 & 0.01 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 & 0.90 & 0.01 & 0.03 & 0.00 \\ 0.01 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.98 & 0.00 & 0.00 \\ 0.05 & 0.00 & 0.01 & 0.00 & 0.00 & 0.00 & 0.00 & 0.07 & 0.03 & 0.85 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.02 & 0.00 & 0.00 & 0.00 & 0.97 \end{pmatrix}$$

Alors que le taux de pixels mal classés est de moins de 8%, on voit grâce à la matrice de confusion que les régions d'apprentissage de la classe 6 ont été mal classées à 26%. Ceci correspond aux deux champs noirs un peu au-dessus du milieu de l'image 9. Ce sont des champs définis comme blé, classés majoritairement en blé d'hiver.

4.2 Régularisation : Topologie et géométrie

Dans cette partie, nous allons étudier l'influence du système de voisinage, ainsi que celle du paramètre β sur la régularisation par modèle de Potts. Pour effectuer des tests comparables, nous n'utiliserons dans toute cette partie que l'algorithme ICM, assez rapide, et qui a la propriété d'être déterministe. Le critère d'arrêt est une stabilisation du nombre de pixels qui changent de classe sur 10 itérations (<1%).

4.2.1 Voisinage en 4-connexité et 8-connexité

Dans cette partie, nous fixons β à 1. Nous allons comparer l'influence du système de voisinage dans la régularisation sur les deux images présentées en première partie. L'image 1 est la région agricole américaine. L'image 2 est la région agricole de Boigneville. L'avantage de l'image 2 est que l'on dispose d'une vérité terrain associée. Pour l'image 1, la vérité terrain est réduite aux régions d'apprentissage qui sont peu étendues.

Dans tous les cas, l'ICM stagne à partir d'un certain moment. Dans le tableau de résultats 12, nous recueillons le nombre d'itérations nécessaires pour arriver à cette situation stable, et le taux d'erreur sur la vérité terrain.

On remarque que le pourcentage d'erreur n'est pas du même ordre d'une image à l'autre. Lorsque l'on réalise des tests, il n'est pas légitime de comparer deux images distinctes. Les

	Voisinage	Nb iter	Erreur (%)
US	4-connexité	3	0,40
	8-connexité	5	0,35
Boigneville	4-connexité	4	7,00
	8-connexité	8	6,50

FIG. 12 – Comparaison des voisinages en 4- et 8-connexité

seules comparaisons acceptables se font pour la même image, entre deux méthodes différentes. Sur l'image de campagne américaine les zones d'apprentissage sont réduites, petites, et les classes sont bien séparables, ce qui assure une bonne classification. Sur l'image de Boigneville, les régions d'apprentissage recouvrent une grande partie de l'image, et certaines classes semblent difficiles à séparer, ce qui explique un pourcentage d'erreur résiduel.

On pourrait se satisfaire d'une de ces deux topologies. Cependant, en regardant l'image des erreurs commises, on observe des petites taches de pixels mal classés. Il est possible d'évacuer en partie ces structures à l'aide d'une topologie dite à trous.

4.2.2 Topologie à trous

La topologie à trous est introduite dans l'article de Descombes, Moctesuma, Maître et Rudant [7] en imagerie radar, pour exploiter la corrélation du bruit. Ici, nous l'utilisons pour "détruire" certaines structures stables mais indésirables.

Une topologie à trous consiste à ajouter au système de voisinage en 4- ou 8-connexité des éléments plus éloignés. On choisit une topologie classique de base, en 4-connexité par exemple, et une longueur de saut, 5 par exemple. On ajoute au voisinage d'un pixel les éléments qui se situent 5 pixels plus loin dans les 4 directions des 4 pixels voisins d'origine (voir figure 13).

Ainsi, les pixels dans une tache jugée mal classée auront plus de chance d'être reclassés correctement si l'on ajoute des pixels un peu éloignés à leur voisinage. Les résultats sont présentés dans le tableau 14, à comparer avec le tableau 12. Pour que les résultats soient comparables, il a fallu fixer $\beta = 0,5$, puisque nous avons doublé la taille du voisinage de chaque pixel, donc doublé le nombre de cliques de l'image, et l'expression de l'énergie est :

$$U(x/y) = \sum_s ((y_s - \mu_{x_s})^T [\Sigma_{x_s}]^{-1} (y_s - \mu_{x_s}) + \log \det(\Sigma_{x_s})) + \beta \sum_{(s,t) \in C_2} \delta_{x_s, x_t}$$

On peut se rendre compte visuellement de l'avantage de cette topologie en comparant les images des erreurs figure 15. On remarque qu'une partie des petites zones d'erreur sont

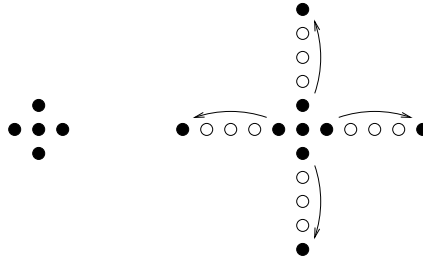


FIG. 13 – Topologie en 4-connectivité classique et à trous

	Voisinage	Nb iter	Erreur (%)
US	4 à trous	4	0,45
	8 à trous	5	0,30
Boigneville	4 à trous	6	6,81
	8 à trous	17	5,97

FIG. 14 – Comparaison des voisinages dans le cas de la topologie à trous

éliminées. Cependant, la vitesse de la régularisation dépend linéairement de la taille du voisinage des pixels. Ainsi, en passant d'un système en 4-connectivité à un système en 8-connectivité, on double le temps de calcul. On le double à nouveau en passant à la topologie à trous.

4.2.3 Le paramètre β

Le paramètre β pondère le terme de régularisation. On s'attend donc à obtenir de plus grandes parcelles homogènes pour des valeurs de β élevées. Le tableau 16 montre le pourcentage d'erreur pour différentes valeurs. Dans toute cette partie, on choisit une topologie en 8-connectivité à trous.

Dans l'image 1 (US), on obtient des résultats presque parfaits pour une valeur de 5.

Dans l'image 2 (Boigneville), les meilleurs résultats sont obtenus pour β entre 5 et 10. Mais en observant les images classifiées pour les valeurs de β entre 3 et 5, on se rend compte que, en dehors des zones d'apprentissage, une régularisation trop forte détruit certaines petites parcelles (voir figure 17). On préférera donc des valeurs inférieures à 3.



FIG. 15 – Images des erreurs en topologie classique (a) et à trous (b)

	β	Nb iter	Erreur (%)
US	1	11	0,20
	3	24	0,10
	5	27	0,02
	10	33	0,05
	100	25	0,94
Boigneville	1	19	5,19
	3	23	4,18
	5	40	3,92
	10	30	3,90
	100	37	6,79

FIG. 16 – Comparaison pour différentes valeurs de β en 8-connexité à trous

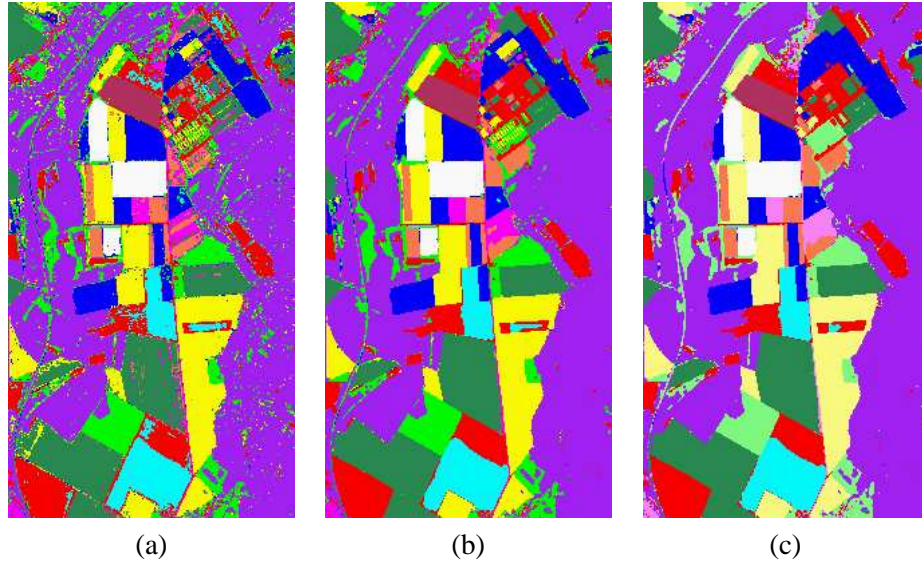


FIG. 17 – Image classifiée brute (a), et images régularisées par ICM avec une topologie en 8-connectivité à trous, pour $\beta=3$ (b) et $\beta=5$ (c)

4.3 Régularisation : ICM et recuit

Dans cette partie, nous fixons β à 2 pour l'image 2 (Boigneville), et à 5 pour l'image 1 (US). La topologie est en 8-connectivité à trous. Nous allons comparer l'algorithme ICM et le recuit simulé pour plusieurs valeurs de la température initiale. L'évolution de la température sera géométrique, plus précisément $T^{(n+1)} = 0,98 * T^{(n)}$. Nous nous limitons au recuit simulé inhomogène, c'est-à-dire que, à chaque itération, la température est diminuée. Un recuit homogène serait plus long, car on itère jusqu'à stabilité à température donnée avant de réduire T .

Nous insistons sur le fait que, dans un but d'accélérer le recuit, nous avons dans un premier temps modifié la procédure décrite dans le chapitre précédent. Le choix d'une classe à affecter à un pixel ne se fait pas au hasard dans l'ensemble des classes, mais uniquement parmi les classes représentées dans le voisinage.

Le recuit simulé est arrêté lorsque sur 10 itérations successives, moins de 1% des pixels de l'image changent de classe à chaque itération. Nous décidons de terminer par un ICM qui, certes, ne possède pas la propriété de convergence du recuit, mais permet tout de même d'accélérer la diminution d'énergie à la fin du calcul pour atteindre un minimum local. Les résultats sont présentés dans le tableau 18.

	$T^{(0)}$	Nb iter recuit	T final	Nb iter ICM	Erreur (%)
US $\beta = 5$	ICM			27	0,02
	2	14	1,51	27	0,02
	5	15	3,70	11	0
	10	20	6,68	10	0
	20	35	9,86	9	0
	50	66	13,18	10	0
	100	94	14,97	15	0
Boigneville $\beta = 2$	ICM			20	4,32
	2	13	1,54	13	4,22
	5	16	3,61	10	3,98
	10	32	5,23	12	3,78
	20	62	5,72	29	3,92
	50	104	6,12	13	5,55
	100	136	6,28	15	6,30

FIG. 18 – Comparaison ICM et recuit simulé adapté pour différentes températures initiales

Pour l'image de Boigneville, la température initiale idéale semble être environ 10. Les résultats sont meilleurs qu'avec un ICM classique, comme nous l'avions prévu. Les 3,78% d'erreurs sont essentiellement dus à l'ambiguïté entre deux types de blés relativement semblables. Il se peut que la vérité terrain soit erronée sur les deux champs concernés. Pour les températures supérieures, nous constatons une augmentation dans le taux d'erreur, ce qui va à l'encontre du principe du recuit simulé. Ceci est lié à notre modification du recuit, que nous avons signalée au début de cette partie.

Pour l'image 1 (US), le taux d'erreur ne permet pas de juger de la qualité de la régularisation. Un test visuel pencherait pour des températures très grandes. Pour se rapprocher d'un cas idéal, nous avons prolongé le recuit pour une température de 100, le résultat est meilleur. Nous l'avons comparé avec le cas $T = 10$ en figure 19.

De manière générale, une température initiale trop faible fait perdre au recuit simulé son originalité, car on est plus restrictif sur les sauts d'énergie, et l'algorithme tombe plus facilement dans un minimum local de l'énergie.

Pour s'assurer du bon fonctionnement du recuit simulé, nous l'avons appliqué de manière stricte à l'image de Boigneville, et les résultats figurent dans le tableau 20. Il est bon de noter que le taux d'erreur et la température finale sont plus stables. Lorsqu'on impose à un pixel de ne choisir sa nouvelle classe que parmi celles de ses voisins, on peut atteindre des

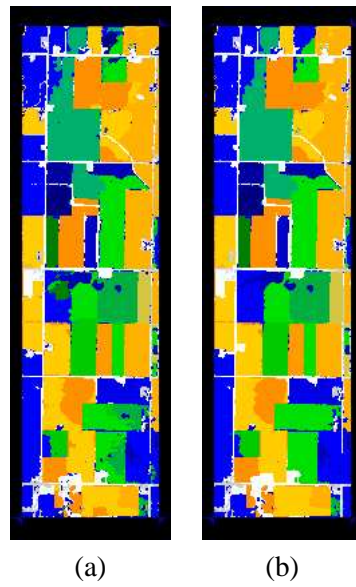


FIG. 19 – *Classifications régularisées par recuit adapté pour $\beta = 5$, avec une température initiale $T = 10$ (a) et $T = 100$ (b)*

	$T^{(0)}$	Nb iter recuit	T final	Nb iter ICM	Erreur (%)
Boigneville $\beta = 2$	ICM			20	4,32
	2	10	1,63	19	4,27
	5	10	4,09	18	4,23
	10	39	4,54	15	4,26
	20	73	4,58	12	4,05
	50	120	4,42	14	4,06
	100	153	4,55	14	4,01

FIG. 20 – Comparaison ICM et recuit simulé pour l'image de Boigneville

résultats légèrement meilleurs en temps de calcul et en taux d'erreur, pour une température initiale ni trop grande ni trop faible. Cette modification semble avoir un effet régularisant. Cependant, un vrai recuit simulé est moins hasardeux.

5 Des paramètres de régularisation adaptatifs

5.1 Motivations

Dans l’image de Boigneville, nous avons rajouté une classe “route” pour tenter de détecter une route qui traverse la forêt. Or, la régularisation telle que nous l’avons décrite, qui s’appuie sur l’observation du voisinage de chaque pixel, risque de détruire la classe “route”. En effet, à notre échelle, l’épaisseur de la route est de 1 pixel, et un pixel de route est davantage entouré de pixels de la classe “forêt” que de pixels de la classe “route”. Il se peut donc que le terme de régularisation favorise la classe “forêt”. On pressent que les routes et les classes dont la structure est fine en général (lignes d’épaisseur 1 ou 2 pixels) risquent de manquer de robustesse. C’est donc dans cette optique que nous avons introduit des paramètres de régularisation adaptés aux classes.

Cependant, dans nos exemples, les structures fines telles que les routes ont une réponse spectrale suffisamment différente de celles des paysages agricoles et forestiers environnants, ce qui permet une bonne détection malgré le problème évoqué précédemment. Par contre, nous avons vu que le taux d’erreur résiduel sur l’image de Boigneville correspond à la confusion entre le blé et le blé d’hiver (voir la figure 21). Il est possible de rendre la classe blé plus “forte” lors de la régularisation. Pour y parvenir, nous modifions le modèle de Potts pour qu’une clique d’ordre 2 de type “blé-blé” ait une énergie plus faible qu’une clique de type “blé d’hiver-blé d’hiver”.

5.2 Formulation

Reprenons l’expression de l’énergie du modèle de Potts :

$$U(x/y) = \sum_s ((y_s - \mu_{x_s})^T [\Sigma_{x_s}]^{-1} (y_s - \mu_{x_s}) + \log \det(\Sigma_{x_s})) + \beta \sum_{(s,t) \in C_2} \delta_{x_s, x_t}$$

Pour chaque classe λ , nous fixons un paramètre β_λ qui vaut 1 par défaut, et qui est d’autant plus grand que l’on souhaite que la classe soit robuste. Le terme de régularisation est ainsi modifié :

$$\beta \times \sum_{(s,t) \in C_2} \beta_{x_s} \beta_{x_t} \delta_{x_s, x_t}$$

Sur la figure 22, nous avons représenté un pixel situé sur une route qui traverse une forêt, ainsi que ses huit voisins connexes, dans une image dont la résolution donne à cette route une épaisseur de 1 pixel. On voudrait que le pixel central soit classé en “route”. Même si le terme d’attache aux données favorise cette classe, dans le cas d’une forte valeur de β , le terme de régularisation pourrait *in fine* classer ce pixel dans la classe “forêt”.



FIG. 21 – Image de confusion après régularisation

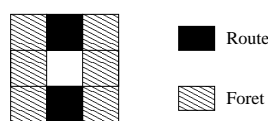


FIG. 22 – Classification d'un pixel de route

En effet, si l'on revient au modèle de Potts tel que nous l'avons introduit au chapitre 3, avec un système de voisinage en 8-connexité, la différence d'énergie entre les deux configurations, en notant y le vecteur associé au pixel dans l'espace des états, x sa classe, r la classe route, f la classe forêt, $d(y, \lambda)$ la distance de Mahalanobis entre le vecteur y et la classe λ , est donnée par :

$$\begin{aligned} U(x = r) - U(x = f) &= (d(y, r) + (6 - 2)\beta) - (d(y, f) + (2 - 6)\beta) \\ &= (d(y, r) - d(y, f)) + 8\beta \end{aligned}$$

En supposant que le terme $d(y, r) - d(y, f)$ soit négatif, le signe de la quantité globale dépend de β .

Dans le cas de paramètres adaptatifs, en fixant par exemple $\beta_r = 4$ et $\beta_f = 1$ on a :

$$\begin{aligned} U(x = r) - U(x = f) &= (d(y, r) + (6 \times \beta_f \times \beta_r - 2 \times \beta_r \times \beta_r)\beta) \\ &\quad - (d(y, f) + (2 \times \beta_r \times \beta_f - 6 \times \beta_f \times \beta_f)\beta) \\ &= (d(y, r) + (6 \times 1 \times 4 - 2 \times 4 \times 4)\beta) \\ &\quad - (d(y, f) + (2 \times 4 \times 1 - 6 \times 1 \times 1)\beta) \\ &= (d(y, r) - 8\beta) - (d(y, f) + 2\beta) \\ &= (d(y, r) - d(y, f)) - 10\beta \end{aligned}$$

Le pixel central sera donc introduit dans la bonne classe, quelque soit la force de la régularisation. Nous sauvegardons ainsi les structures linéiques des routes en instruisant le système avec les paramètres β_λ .

5.3 Application

Nous avons appliqué à l'image de Boigneville une régularisation très forte ($\beta = 20$) par recuit simulé pour un voisinage en 8-connexité. Les deux topologies classique et "à trous" ont été testées. Nous avons isolé la classe "route" qui traverse la forêt sur la gauche de l'image, et étudié sa résistance à la régularisation entre le modèle de Potts et le modèle avec paramètres adaptatifs (voir figure 23).

On se rend compte que la topologie à trous affaiblit davantage ces structures linéiques que la topologie classique, comme nous pouvions nous y attendre. Les paramètres β_λ adaptatifs, fixés à 1 pour l'ensemble des classes et à 4 pour la classe "route", améliorent effectivement la détection de la route. Par contre, le paramétrage doit être fait avec précaution, on voit pour $\beta_{route} = 5$ que la classe "route" commence déjà à déborder sur les classes voisines.

Revenons au problème de la confusion entre blé et blé d'hiver. Nous avons donné à la classe "blé" plus de force que la classe "blé d'hiver" par un jeu de paramètres. Nous

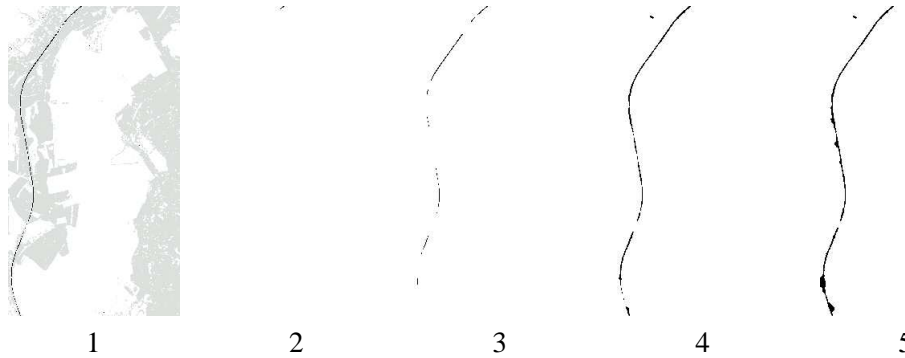


FIG. 23 – Résistance de la classe route à la régularisation ($\beta = 20$, recuit) : 1. classification sans régularisation ; 2. modèle de Potts classique et topologie à trous ; 3. modèle de Potts classique et topologie classique ; 4. $\beta_{route} = 4$ et topologie classique ; 5. $\beta_{route} = 5$ et topologie classique

avons aussi avantage le maïs par rapport aux betteraves, qui avaient tendance à en recouvrir certaines parcelles. Ces démarches sont artificielles, elles ne traduisent aucune réalité géométrique sur les parcelles des classes considérées. Elles ont éventuellement une justification qu'il faut chercher dans la forme des clusters dans l'espace des états. Nous obtenons toutefois des résultats satisfaisants.

Auparavant, nous avions un taux d'erreur d'environ 4% au mieux. Avec un paramétrage adaptatif, on le réduit à 2%. On peut comparer la nouvelle image des erreurs de classification (figure 24) avec celle que l'on obtenait pour un modèle de Potts classique (figure 21). On voit bien que l'on a résolu certains problèmes de confusion de classes.

5.4 Problème d'automatisation

Même si ce modèle améliore la classification, il introduit autant de nouveaux paramètres que de classes définies à l'apprentissage. L'intérêt de cette méthode serait assuré si nous pouvions fixer ces paramètres de manière automatique. Dans un premier temps, nous proposons une méthode pour fixer ces paramètres de manière grossière.

La classification d'un pixel lors de la régularisation se joue sur la différence entre les termes de régularisation de l'énergie pour deux configurations différentes. Cette grandeur dépend du nombre de pixels d'une certaine classe λ dans le voisinage, multiplié par le paramètre de cette même classe β_λ . On voudrait donc que, pour toutes les classes, ce produit soit du même ordre de grandeur.



FIG. 24 – Image de confusion après régularisation, avec un paramétrage adaptatif

Au début de la phase de régularisation, on dispose d'une image classifiée par maximum de vraisemblance. L'observation visuelle permet de se faire une idée des structures des classes. On perçoit distinctement les formes linéiques, telles que les routes, et les grandes parcelles, telles que les cultures. Typiquement, un pixel de route possède deux pixels de la même classe dans son voisinage. Quant à un pixel de culture, il possède habituellement huit pixels de la même classe.

Ainsi, tel que nous l'avons proposé dans un exemple précédent, le paramètre β_{route} devrait être environ 4 fois supérieur au paramètre β_{champ} . Une première méthode pour estimer ces paramètres consiste à étudier sur l'image classifiée brute, pour chaque classe, le nombre moyen pour les pixels de cette classe, de voisins de même type. Le nombre obtenu est compris entre 0, pour une classe totalement incohérente et éparpillée, et 8, pour une classe qui s'étend sur de vastes surfaces uniformes.

Le paramètre β_λ est fixé sur l'inverse du nombre obtenu par cette étude de voisinage, à un facteur multiplicatif près, 8 dans nos applications.

Même si cette méthode fonctionne bien en théorie pour les structures linéiques et planaires, on est confronté en pratique à certains problèmes. La classification brute est bruitée, et il se peut que des pixels de deux classes ambiguës se mélangent et faussent l'étude de la structure de la classe. On peut toujours corriger ce problème en n'estimant les paramètres β_λ qu'après quelques étapes de régularisation, qui auront un effet de débruitage sur l'image.

On est aussi parfois confronté à des classes dont la structure n'est pas bien définie. Une classe route non différenciée pourrait inclure à la fois des segments fins de route et des grandes surfaces correspondant par exemple à des parkings. Enfin, lorsque nous utilisons les paramètres adaptatifs pour mettre sensiblement plus de poids sur une classe qui se confond avec une autre, il devient plus compliqué de définir une méthode automatique. Le chien-modèle [8] de Xavier Descombes apporte une amélioration sur ce plan.

Pour pousser le modèle un peu plus loin, on pourrait à nouveau modifier le modèle en introduisant des paramètres qui caractérisent la compatibilité entre deux classes. On aurait ainsi une matrice de paramètres au lieu d'un vecteur, avec des avantages pour la régularisation, mais aussi une certaine quantité d'inconvénients pour l'estimation de ces paramètres.



FIG. 25 – Image de texture locale d’une bande de l’image de Boigneville

6 Des critères de texture

Dans les méthodes de classification vues jusqu’à présent, nous nous intéressons à l’information contextuelle uniquement après classification, via les champs de Markov dans l’image classifiée. Or, les bandes de l’image hyperspectrale initiale contiennent également de l’information contextuelle. Certains paysages, tels que les forêts, sont relativement texturés. Ces zones présentent de fortes variations radiométriques, et ceci peut perturber l’apprentissage.

En ajoutant une image synthétique dont la valeur en chaque pixel est caractéristique de la texture locale, les classes texturées et les classes homogènes sont mieux séparables. Pour ce faire, il existe plusieurs modèles. Ces modèles consistent à sélectionner une bande, et à étudier, en chaque pixel de l’image, la texture locale, en niveau de gris, dans une fenêtre centrée sur le pixel considéré.

Un modèle très simple consiste à étudier la variance autour de chaque pixel dans une fenêtre de taille 3x3 autour de celui-ci. Lorsque l’on réalise cela sur une bande située dans l’infra-rouge, la forêt apparaît distinctement. Par contre, on extrait aussi les bords des champs, ce qui risque de gêner la classification (voir figure 25). Nous allons proposer ci-dessous des méthodes plus performantes.

6.1 Différents modèles de texture

Pour une liste plus exhaustive de l'ensemble des méthodes d'extraction de texture, nous conseillons au lecteur de se référer à la thèse d'Anne Lorette [18].

6.1.1 Les matrices de cooccurrence

Cette méthode d'extraction de texture est une méthode anisotrope introduite par Haralick [12]. On choisit un vecteur de déplacement dans l'image $d(d_x, d_y)$. On se place sur un site de l'image, et l'on considère une fenêtre carrée centrée sur ce pixel. La matrice de cooccurrence $M_d = [m_{gg'}]$ en ce site est alors construite en estimant les probabilités de passer d'un niveau de gris g à un niveau de gris g' en se déplaçant d'un vecteur d . L'estimation se fait sur toute la fenêtre autour du pixel.

$$m_{gg'} = P[X_s = g, X_t = g' | t_x = s_x + d_x, t_y = s_y + d_y]$$

où X_s et X_t sont respectivement les niveaux de gris des sites s et t .

Les niveaux de gris sont quantifiés, afin de traiter des matrices de taille raisonnable. Ayant choisi une direction, nous ne souhaitons privilégier aucun des deux sens d ou $-d$. On rend donc la matrice de cooccurrence M_d symétrique en y ajoutant sa transposée et en divisant par 2.

Voici 5 exemples de paramètres que l'on peut extraire d'une telle matrice :

$$\begin{aligned} \text{Energie} &= \sum_{gg'} m_{gg'}^2 \\ \text{Inertie} &= \sum_{gg'} (g - g')^2 m_{gg'}^2 \\ \text{"Cluster shade"} &= \sum_{gg'} ((g - \mu_g) - (g' - \mu_{g'}))^3 m_{gg'} \\ \text{"Cluster prominence"} &= \sum_{gg'} ((g - \mu_g) - (g' - \mu_{g'}))^4 m_{gg'} \\ \text{"Inverse different moments"} &= \sum_{gg'} \frac{1}{1 + (g - g')^2} m_{gg'} \\ \text{où } \mu_g &= \sum_g g \sum_{g'} m_{gg'} \\ \text{où } \mu_{g'} &= \sum_{g'} g' \sum_g m_{gg'} \end{aligned}$$

La figure 26 représente deux images obtenue par le paramètre "*cluster prominence*" sur une bande de l'image de Boigneville dans le proche infra-rouge. Pour la première, le vecteur $d = (0,1)$, pour la deuxième, $d = (1,0)$. On voit nettement que la direction du vecteur d favorise la detection des bords et routes perpendiculaires. De plus, les zones à micro-texture, comme la forêt à cette échelle, sont toujours détectées. On entrevoit donc la possibilité, en utilisant cette méthode pour plusieurs directions, de bien faire la différence de manière automatique entre la forêt et un bord de champ.

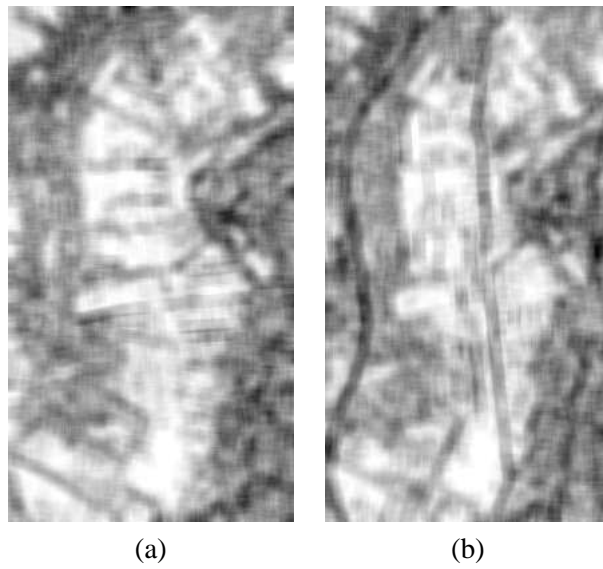


FIG. 26 – Images du paramètre de “cluster prominence” des matrices de cooccurrence dans les directions $(0,1)$ (a) et $(1,0)$ (b) sur une bande du proche infra-rouge de l’image de Boigneville

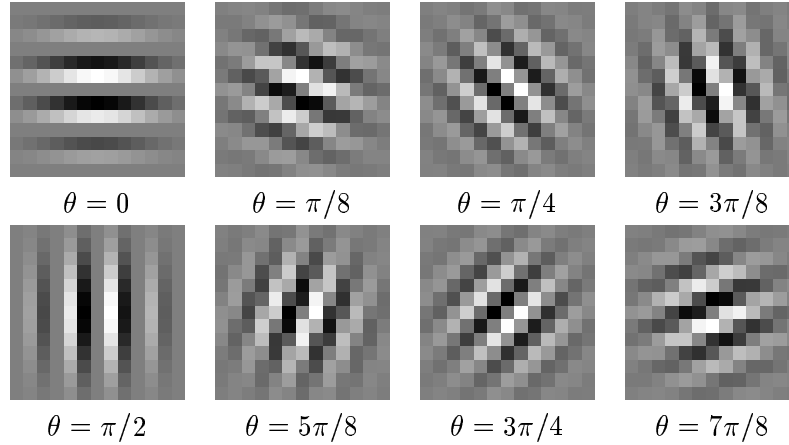


FIG. 27 – Quelques filtres de Gabor, $\lambda = 3$, $\sigma = 3$, $\phi = \frac{\pi}{2}$

6.1.2 Les filtres de Gabor

La méthode, anisotrope, des filtres de Gabor consiste à convoluer une image en niveaux de gris avec les filtres en question. Un filtre de Gabor est une onde plane sinusoïdale modulée par une enveloppe gaussienne :

$$G(s_x, s_y / \theta = 0, \lambda, \phi) = \exp \frac{-(s_x^2 + s_y^2)}{2\sigma^2} \cos\left(\frac{2\pi}{\lambda} s_x + \phi\right)$$

θ est l'angle du filtre de Gabor, entre 0 et π . L'angle ϕ prendra soit la valeur 0, soit $\frac{\pi}{2}$, selon que l'on souhaite avoir un filtre pair ou impair. λ est une longueur d'onde de la sinusoïde, et σ est l'écart-type de la gaussienne.

La figure 27 montre des exemples de tels filtres sur une fenêtre 13x13, avec $\lambda = 3$, $\sigma = 3$, $\phi = \frac{\pi}{2}$.

L'anisotropie des filtres de Gabor a pour conséquence la détection ou non des bords des champs selon leur orientation. Ceci nous permet donc, en utilisant 8 directions différentes pour les filtres de Gabor, de caractériser les bords des champs. Ainsi, dans un but de n'extraire que la forêt, on applique une méthode développée dans la thèse d'Anne Lorette [18].

En chaque pixel s , on calcule le résultat de la convolution de son voisinage par chacun des filtres de Gabor. On classe les 8 résultats dans l'ordre croissant, $I_1 < I_2 < I_3 < I_4 < I_5 < I_6 < I_7 < I_8$, et on conserve la valeur médiane, soit $I_m = \frac{I_4 + I_5}{2}$. La forêt aura une réponse forte dans toutes les directions, I_m sera fort. Dans un champ, toutes les réponses



FIG. 28 – Image du paramètre de texture issu des filtres de Gabor

sont faibles, donc I_m est faible. Au bord d'un champ, la réponse ne sera forte que dans 2 ou 3 directions, d'après l'expérience, $\frac{I_4+I_5}{2}$ aura donc une valeur plus proche de la valeur typique des champs que de celle de la forêt.

L'image 28 montre le produit de cette opération, après avoir appliqué un filtre médian.

6.1.3 Un modèle fondé sur la variance locale

Nous proposons ici un autre modèle isotrope de texture qui se fonde sur le calcul de la variance locale (comme sur la figure 25). La figure 29 montre une coupe d'une bande de l'image dans le proche infra-rouge. On peut voir que la variance en niveaux de gris dans la forêt à droite est plus forte que sur les champs du milieu. Par contre, les pixels au bord des champs ont aussi une forte variance locale, ce qui gênera notre classification en répondant fortement aux bords des champs.

Nous allons extraire un paramètre qui caractérise le bord d'un champ. Sur la coupe de la figure 29, on voit que les bords des champs correspondent à des sauts en niveaux de gris, auxquels s'ajoute un signal dont la variance est négligeable par rapport à la hauteur du saut. Certains travaux traitent spécialement de la classification de micro-textures par des paramètres fractals, comme l'article de Limas Serafim [22].



FIG. 29 – Coupe d'une bande du proche infra-rouge de l'image de Boigneville

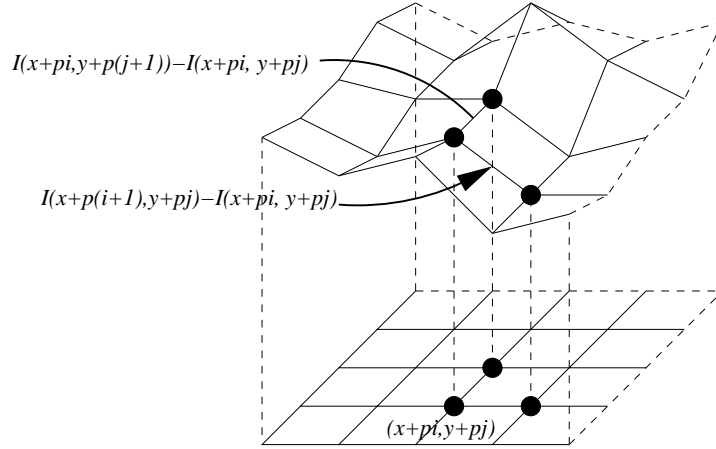


FIG. 30 – Illustration du calcul de variation totale sur une fenêtre

Dans une fenêtre 13×13 autour du pixel, on choisit un pas p , qui correspond à la taille d'une grille que l'on plaque sur l'image. On impose que p divise 12, et on calcule une variation totale au carré V_p sur la fenêtre. Si la fenêtre est centrée en (x, y) , on a :

$$V_p = \sum_{i=-p}^{p-1} \sum_{j=-p}^{p-1} (I(x + p \cdot (i + 1), y + p \cdot j) - I(x + p \cdot i, y + p \cdot j))^2 + (I(x + p \cdot i, y + p \cdot (j + 1)) - I(x + p \cdot i, y + p \cdot j))^2$$

La figure 30 illustre cette équation.

On réalise une telle mesure pour toutes les valeurs de p qui divisent 12 : 1, 2, 3, 4, 6, 12. On estime ensuite un paramètre d , proche de la notion de dimension de Hausdorff [22], qui vérifie $V_p \cdot p^{d-2} = cte$. On en déduit : $(d - 2) \log(p) + \log(V_p) = cte$.

On estime donc d ainsi :

$$d - 2 = \frac{d \log(V_e)}{d \log(e)}$$

Pour une surface plane, $d = 2$, et pour une surface bruitée, $2 < d < 3$. Un saut tel qu'un bord de champ aura une réponse de type surface plane, alors qu'une micro-texture aura une réponse plus forte, notamment pour les forêts et les champs. En multipliant le paramètre $(d - 2)$ par la variance locale, on obtient un nouveau paramètre qui sélectionne les zones à forte variance, et évacue les bords des champs.



FIG. 31 – Image de texture d'une bande dans le proche infra-rouge issue de l'image de Boigneville

L'image 31 montre le résultat de ce traitement pour une bande dans le proche infra-rouge issue de l'image de Boigneville, après avoir appliqué un filtre médian. Ce filtre s'avère être utile pour rendre l'image plus lisse, et on constate expérimentalement que la classification fonctionne mieux.

6.2 Résultats pour un plan d'occupation des sols

Dans une optique de plan d'occupation des sols, la classification consiste à séparer par exemple la ville, la campagne, la forêt, la mer et la montagne. Nous disposons d'une image qui comporte à la fois de la ville, de la forêt et de la campagne.

C'est une image de la ville de Boigneville, qui prolonge l'image de la campagne que nous avons étudiée jusqu'à présent. Elle comporte 20 bandes, du bleu ($0,44 \mu m$) au proche infra-rouge ($0,82 \mu m$). Sa résolution est de 2,5 mètres par pixel, mais nous avons d'abord simulé une image à 10 mètres par pixel. Le capteur est un capteur MIVIS 20 bandes de 20 nm.

Les images 32 représentent la scène et les régions d'apprentissage pour la ville, la forêt et la campagne.

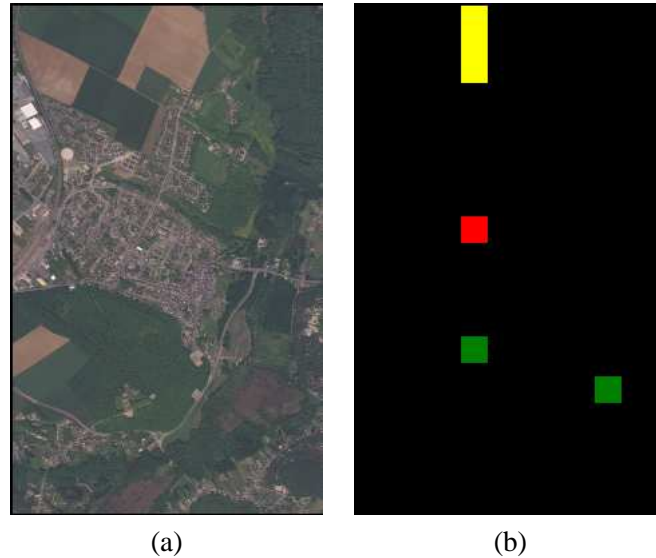


FIG. 32 – Image 3 : ville de Boigneville (a), et régions d'apprentissage : ville, campagne, forêt (b)

A cette échelle, nous considérons la ville comme une microtexture, car nous distinguons à peine les objets. Ceci nous permet d'utiliser les modèles de texture présentés ci-dessus. On conçoit que l'on ne peut pas segmenter la ville du reste de l'image autrement qu'en extrayant un paramètre de texture. La réponse radiométrique de la ville est totalement chaotique et ne constitue pas un caractère suffisamment discriminant pour bien la séparer des autres classes. A la limite, on pourrait mieux séparer la forêt de la campagne avec une simple étude spectrale, mais des erreurs seraient commises.

La figure 33 exhibe deux images du critère de texture fondé sur la variance locale appliqué sur deux bandes de l'image : l'une dans le spectre visible, l'autre dans le proche infra-rouge. On peut déjà conclure que les critères de texture sur une bande du spectre visible permettent de différencier la ville de la forêt et la campagne. Et La texture sur une bande dans le proche infra-rouge permet de discriminer la campagne de la ville et la forêt. En théorie, nous avons deux critères suffisants pour réaliser une classification de type plan d'occupation des sols.

On procède donc à la classification, en ajoutant aux bandes de l'image nos deux bandes de texture. On compare 4 images :

- 20 bandes radiométriques

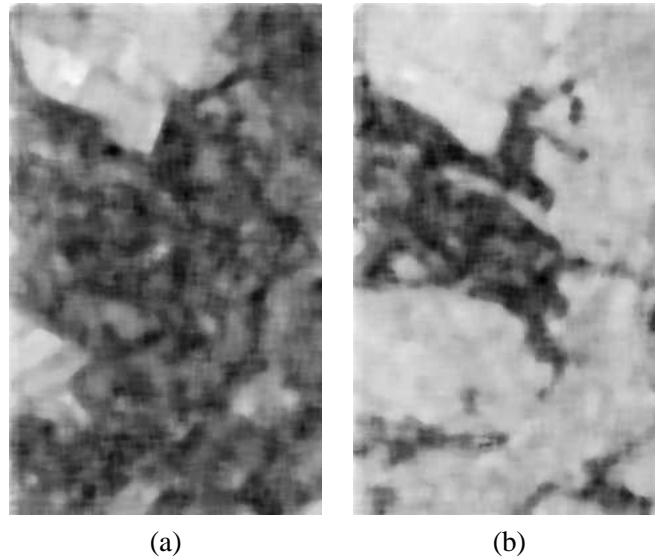


FIG. 33 – Images du critère de texture sur deux bandes : (a) proche infra-rouge, (b) bleu

- 20 bandes radiométriques + 2 de texture
- 2 bandes de texture
- 3 bandes radiométriques + 2 de texture

Les images sont regroupées en figure 34.

Les images montrent que la radiométrie seule ou la texture seule ne permettent pas de réaliser une bonne classification. Il faut manifestement mélanger les deux types d'information. Cependant, cela ne suffit pas, on constate qu'entre les deux images mixtes, on a un meilleur résultat en retirant quelques bandes radiométriques.

6.3 Résultats sur l'image de campagne

Sur l'image de campagne proche de Boigneville, nous avons réalisé une classification avec des bandes de texture. Nous comparons les résultats dans le tableau 35. Cependant, il n'apparaît pas dans ce tableau que la route n'est plus détectée. Elle est en effet noyée dans une texture de forêt. Par contre, nous avons évoqué le problème des parcelles ambiguës de blé et blé d'hiver. La texture résout ce problème partiellement, les paramètres de régularisation adaptatifs améliorent encore la classification.

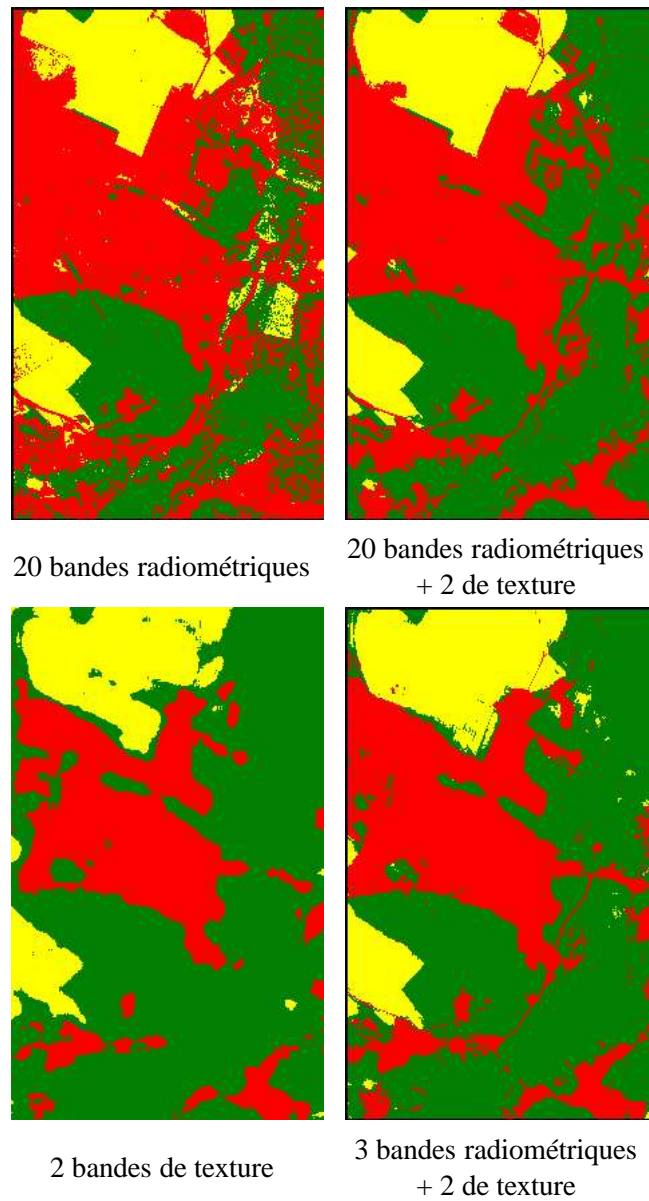


FIG. 34 – *Classification d'images qui combinent information radiométrique et critères de texture*

	Erreur (%)
classification 10 bandes	7,64
classification 10 bandes +“sieve” et “clump”	6,49
classification 10 bandes +ICM ($\beta = 2$)	4,32
classification 10 bandes +recuit simulé ($\beta = 2$)	4,00
classification 10 bandes +texture bandes 1 et 10 : variance	6,39
classification 10 bandes +texture bandes 1, 7, 10 : variance	6,27
classification 10 bandes +texture bandes 1, 7, 10 : Gabor	6,49
classification 10 bandes +texture bandes 1, 7, 10 : variance et Gabor	4,77
classification précédente +recuit ($\beta = 2$)	2,75
classification précédente +recuit ($\beta = 2$) + β adaptatifs	1,42

FIG. 35 – Qualité de la classification sur l'image de campagne proche de Boigneville, pour diverses méthodes

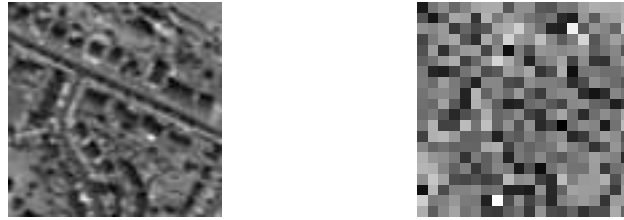


FIG. 36 – *Detail de Boigneville aux résolutions 2,5m et 10m*

6.4 Remarques sur la texture

Nous avons constaté dans cette partie l'importance d'une étude de texture sur l'image initiale pour améliorer la classification de zones de forêt et zones urbaines. Les 4 images de la figure 34 montrent aussi que notre modèle de texture ne permet pas de détecter les routes, et que nous avons besoin d'information radiométrique pour affiner la classification.

Le problème du plan d'occupation des sols est, paradoxalement, que nous avons très peu de classes. La difficulté qui en découle se situe au niveau de l'apprentissage. Par exemple, la classe "champs" contient aussi bien des champs de blé, de maïs, les champs en jachère, ou toute autre culture. D'un point de vue radiométrique, tous ces types de récoltes ont des réponses spectrales très différentes. L'hypothèse gaussienne devient irréaliste, et il est parfois tentant de définir des sous-classes, au risque de sortir du cadre de plan d'occupation des sols.

Le choix du modèle de texture est intimement lié à la résolution de l'image. A une résolution de 10m, la ville peut être considérée comme une micro-texture. A une résolution de 2,5m, on commence à distinguer les maisons, les rues, les jardins. La ville est un mélange de petites structures (voir figure 36). La classe "ville" devient alors nettement plus difficile à définir, autrement que par une collection de sous-classes. On pourrait alors envisager un traitement multi-échelle.

Krishnamachari et Chellappa [14] proposent une méthode de segmentation de la texture fondée sur la modélisation par champs de Markov. Cette analyse de texture se fait de plus par un traitement multi-échelle, qui améliore la classification.

Enfin, après la sélection de critères radiométriques et de texture, se présente une nouvelle difficulté. Il faut correctement pondérer les critères sélectionnés. Quel poids faut-il donner à la texture par rapport à l'information radiométrique? Et comment traduire cela lors de la classification? Nous allons essayer d'y répondre dans le chapitre suivant.

7 Pondération des critères

Dans ce chapitre, nous proposons diverses approches pour pondérer les différents critères sélectionnés pour la classification (radiométrie, texture, ...). Il n'est pas évident de dire s'il existe un rapport idéal entre le poids de l'information de texture et celui de l'information radiométrique. Nous nous limitons à décrire quelques manières d'intégrer des poids fixés manuellement dans l'énergie de l'image à classifier par maximum de vraisemblance.

7.1 Première approche : déplacement des frontières de décision

Supposons que nous avons 2 critères : un critère de radiométrie correspondant à la bande 1 de l'image, et un critère de texture. Supposons que nous avons 2 classes C_1 et C_2 . L'espace des états est un plan, où les deux classes sont idéalement représentées par deux gaussiennes. Entre ces deux gaussiennes se trouve la frontière de décision entre C_1 et C_2 , qui est l'ensemble des points à égale distance (de Mahalanobis) des deux classes.

A présent, imaginons que nous voulions donner plus de poids à l'information de texture. Une manière intuitive est de déplacer la frontière de décision, de telle manière qu'un point sur cette frontière soit affecté à la classe dont il est le plus proche en ne considérant que le critère de texture. La figure 37 éclaire ces explications.

Déplacer la frontière de décision comme sur la figure 37 revient en fait à déformer les gaussiennes définies par les matrices de covariance. Les transformations correspondantes sont des affinités. Choisissons deux coefficients k_1 et k_2 , un pour chaque critère, tels que $k_1 k_2 = 1$. Les matrices de cooccurrence subissent cette transformation :

$$\Sigma' = \begin{pmatrix} \frac{1}{k_1} \\ \frac{1}{k_2} \end{pmatrix}^T \Sigma \begin{pmatrix} \frac{1}{k_1} \\ \frac{1}{k_2} \end{pmatrix}$$

Soit :

$$[\Sigma'^{-1}] = \begin{pmatrix} k_1 \\ k_2 \end{pmatrix}^T [\Sigma^{-1}] \begin{pmatrix} k_1 \\ k_2 \end{pmatrix}$$

On a aussi :

$$\det(\Sigma'^{-1}) = k_1 k_2 \det(\Sigma^{-1}) = \det(\Sigma^{-1})$$

Et le rapport $\frac{k_1}{k_2}$ représente le rapport de force entre les critères 1 et 2. Par exemple, si l'on veut donner plus de force à la texture qu'à la radiométrie, on a $\frac{k_{texture}}{k_{radiometrie}} > 1$. La valeur du rapport en soi n'a pas une signification bien précise, il permet juste de dire si un critère est renforcé par rapport à l'autre.

On peut aussi se permettre de sélectionner un certain nombre de bandes, éliminer les autres de l'image, et réaliser la classification sur cette image réduite. La pondération se fait

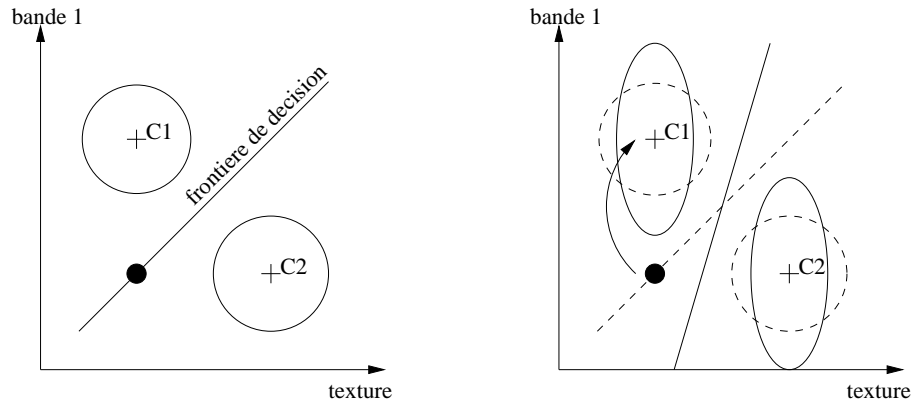


FIG. 37 – *Déplacement d'une frontière de décision en faveur de la texture*

avec la même méthode. Cette simplification de l'image accélère les calculs, mais on élimine une part de l'information initiale.

7.2 Deuxième approche : fission et fusion de critères

7.2.1 Principe

Le principe de la fission/fusion est décrit dans la thèse de Xavier Descombes [6].

L'idée de la démarche fission/fusion est d'extraire des paramètres de l'image, tels que radiométrie et paramètres de texture, et de sélectionner les plus discriminants d'entre eux, éventuellement de les pondérer, et de faire l'hypothèse que ces critères sont indépendants 2 à 2.

Chaque paramètre ou critère i prend une valeur i_s au site s , et on associe, pour une classe j , une énergie $V^{(i,j)}(i_s)$. L'énergie est celle que l'on utilise pour la classification par maximum de vraisemblance.

La fusion des paramètres se fait en un site s par une somme pondérée des énergies décrites ci-dessus. Soit λ une classe affectée au site.

$$V_s(\lambda) = \frac{\sum_i k_i V^{(i,\lambda)}(i_s)}{\sum_i k_i}$$

Partant d'une image hyperspectrale à laquelle on a ajouté par exemple des bandes de texture, il y a deux manières de définir et sélectionner les critères.

Seuil de corrélation	Nombre de bandes restantes
1	20
0.999	17
0.995	10
0.99	8
0.95	4
0.9	3
0.5	1

FIG. 38 – Nombre de bandes restantes après simplification du spectre par une étude de corrélation, pour différents seuils

7.2.2 Choix de bandes discriminantes et non corrélées

Selon les classes définies au départ, certaines bandes sont mieux adaptées pour séparer ces classes. Nous avons vu notamment comment les images de critères de texture permettaient de bien séparer la ville, la campagne, et la forêt. De tels critères méritent d'être choisis lors de l'étape de fission du spectre.

Il est bon de ne pas s'embarrasser d'information redondante. Une analyse en composante principale (ACP) met en évidence l'information essentielle et évacue les redondances [9]. Il existe aussi des modèles de redondance d'images [26].

Une manière plus simple d'opérer est de procéder à une analyse de corrélation des bandes. La corrélation entre deux images est comprise entre 0 (décorrélées) et 1 (corrélées). En se fixant un seuil (par exemple 0,99) à partir duquel on considère que deux bandes sont très corrélées, on forme des ensembles de bandes corrélées. De chaque ensemble, on ne garde qu'un seul représentant. Le tableau 38 montre combien de bandes il reste après une simplification du spectre de l'image de Boigneville à 20 bandes. On se rend compte que les bandes sont relativement corrélées de manière générale.

Après avoir sélectionné les bandes, l'expression de l'énergie pour une bande i , en un pixel s qui prend la valeur i_s et auquel on affecte la classe λ est :

$$V^{(i,\lambda)}(i_s) = \left(\frac{i_s - \mu_\lambda^i}{\sigma_\lambda^i} \right)^2 + 2 \log(\sigma_\lambda^i)$$

où μ_λ^i et σ_λ^i sont respectivement la moyenne et l'écart type de la classe λ sur le paramètre i .

$$\Sigma = \begin{pmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \sigma_{13}^2 & \sigma_{14}^2 & \sigma_{15}^2 & \sigma_{16}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \sigma_{23}^2 & \sigma_{24}^2 & \sigma_{25}^2 & \sigma_{26}^2 \\ \sigma_{31}^2 & \sigma_{32}^2 & \sigma_{33}^2 & \sigma_{34}^2 & \sigma_{35}^2 & \sigma_{36}^2 \\ \sigma_{41}^2 & \sigma_{42}^2 & \sigma_{43}^2 & \sigma_{44}^2 & \sigma_{45}^2 & \sigma_{46}^2 \\ \sigma_{51}^2 & \sigma_{52}^2 & \sigma_{53}^2 & \sigma_{54}^2 & \sigma_{55}^2 & \sigma_{56}^2 \\ \sigma_{61}^2 & \sigma_{62}^2 & \sigma_{63}^2 & \sigma_{64}^2 & \sigma_{65}^2 & \sigma_{66}^2 \end{pmatrix}$$

FIG. 39 – Matrice de covariance d'une classe dans une image à 6 bandes

7.2.3 Découpage du spectre

Dans la méthode précédente, nous avons retiré de l'information de l'image. Pour conserver l'intégrité de l'image, il est possible de procéder légèrement différemment. Après l'étude de corrélation des bandes, on dispose de groupements de bandes corrélées. Eventuellement, on peut réaliser ce groupement de manière différente. Nous suggérons par exemple de réaliser deux groupements : un groupement pour l'information radiométrique, un groupement pour l'information de texture.

Les critères ne sont plus des bandes mais les groupements de bandes. L'expression des énergies élémentaires utilise l'information de covariance, comme auparavant. Pour un groupement i , l'énergie en un pixel s qui prend pour ensemble de valeurs le vecteur y_s^i , auquel on affecte la classe λ vaut :

$$V^{(i,\lambda)}(y_s^i) = ((y_s^i - \mu_i)^T [\Sigma_i]^{-1} (y_s^i - \mu_i) + \log \det(\Sigma_i))$$

μ_i et Σ_i sont respectivement le vecteur moyenne et la matrice de covariance de la classe λ sur le groupement de bandes i .

Lors de l'étape de fusion, il est alors aisé de pondérer l'information de texture et l'information radiométrique. On peut aussi donner plus d'importance à une partie du spectre qu'à une autre.

7.3 Synthèse

Toutes les méthodes décrites dans cette partie se ressemblent, mais présentent bien des caractéristiques différentes. Certaines conservent l'intégralité de l'information, d'autres éliminent l'information redondante. On fait parfois l'hypothèse d'indépendance des critères, parfois non. Chacune de ces méthodes traduit en fait une transformation particulière sur les matrices de covariance des classes. Nous résumons de manière visuelle ces transformations diverses (voir figures 39, 40, 41, 42, 43).

$$\Sigma' = \begin{matrix} & \times k_1 & \times k_2 & \times k_3 & \times k_4 & \times k_5 & \times k_6 \\ \times k_1 & \left(\begin{array}{cccccc} \sigma_{11}^2 & \sigma_{12}^2 & \sigma_{13}^2 & \sigma_{14}^2 & \sigma_{15}^2 & \sigma_{16}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \sigma_{23}^2 & \sigma_{24}^2 & \sigma_{25}^2 & \sigma_{26}^2 \\ \sigma_{31}^2 & \sigma_{32}^2 & \sigma_{33}^2 & \sigma_{34}^2 & \sigma_{35}^2 & \sigma_{36}^2 \\ \sigma_{41}^2 & \sigma_{42}^2 & \sigma_{43}^2 & \sigma_{44}^2 & \sigma_{45}^2 & \sigma_{46}^2 \\ \sigma_{51}^2 & \sigma_{52}^2 & \sigma_{53}^2 & \sigma_{54}^2 & \sigma_{55}^2 & \sigma_{56}^2 \\ \sigma_{61}^2 & \sigma_{62}^2 & \sigma_{63}^2 & \sigma_{64}^2 & \sigma_{65}^2 & \sigma_{66}^2 \end{array} \right) \\ \times k_2 & & & & & & \\ \times k_3 & & & & & & \\ \times k_4 & & & & & & \\ \times k_5 & & & & & & \\ \times k_6 & & & & & & \end{matrix}$$

FIG. 40 – Pondération des bandes sans perte d'information

$$\Sigma' = \begin{matrix} & \times k_1 & \times k_2 & \times k_3 \\ \times k_1 & \left(\begin{array}{cc|cc|cc} \sigma_{11}^2 & \sigma_{12}^2 & 0 & 0 & 0 & 0 \\ \sigma_{21}^2 & \sigma_{22}^2 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \sigma_{33}^2 & \sigma_{34}^2 & 0 & 0 \\ 0 & 0 & \sigma_{43}^2 & \sigma_{44}^2 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & \sigma_{55}^2 & \sigma_{56}^2 \\ 0 & 0 & 0 & 0 & \sigma_{65}^2 & \sigma_{66}^2 \end{array} \right) \\ \times k_2 & & & & & & \\ \times k_3 & & & & & & \end{matrix}$$

FIG. 41 – Pondération des bandes par groupements avec hypothèse d'indépendance de ces groupements

$$\Sigma' = \begin{matrix} & \times k_1 & \times k_2 & \times k_3 \\ \times k_1 & \left(\begin{array}{ccc} \sigma_{11}^2 & \sigma_{13}^2 & \sigma_{15}^2 \\ \sigma_{31}^2 & \sigma_{33}^2 & \sigma_{35}^2 \\ \sigma_{51}^2 & \sigma_{53}^2 & \sigma_{55}^2 \end{array} \right) \\ \times k_2 & & & \\ \times k_3 & & & \end{matrix}$$

FIG. 42 – Sélection de critères

$$\Sigma' = \begin{matrix} & \times k_1 & \times k_2 & \times k_3 \\ \times k_1 & \left(\begin{array}{ccc} \sigma_{11}^2 & 0 & 0 \\ 0 & \sigma_{33}^2 & 0 \\ 0 & 0 & \sigma_{55}^2 \end{array} \right) \\ \times k_2 & & & \\ \times k_3 & & & \end{matrix}$$

FIG. 43 – Sélection de critères supposés indépendants 2 à 2

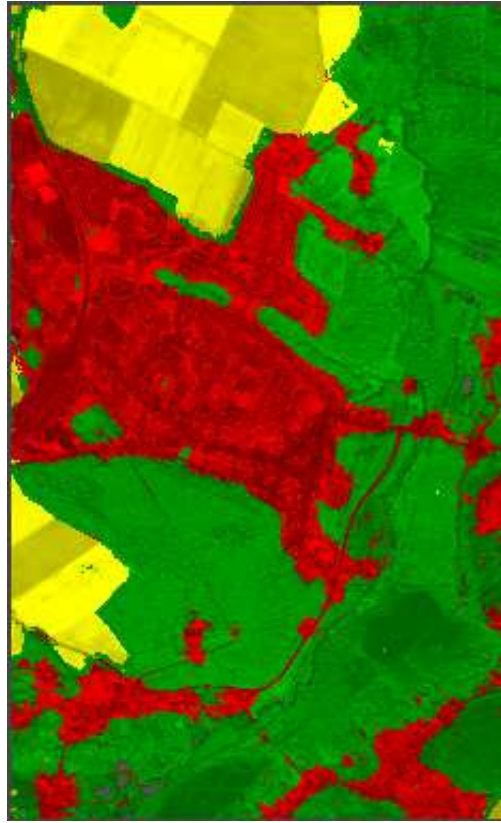


FIG. 44 – *Fusion d'une classification et d'une bande de l'image de Boigneville*

Dans ce chapitre, nous avons fait intervenir sous différentes formes des facteurs de pondération des bandes ou de groupements de bandes. Il est possible à cette étape de changer les paramètres pour chaque classe. On traduit alors la capacité d'un paramètre spécifique à caractériser une classe spécifique. Nous n'avons pas expérimenté cet aspect.

Pour conclure, nous présentons l'image classifiée, régularisée de Boigneville après une simplification du spectre en 3 bandes radiométriques et 2 bandes de texture (voir figure 44).

8 Conclusion et perspectives

Les résultats obtenus sur les images tests permettent de juger la méthode de régularisation par champs de Markov. Nous sommes partis d'une méthode de classification par maximum de vraisemblance, régularisée par deux opérateurs morphologiques sur l'image classifiée. Le formalisme markovien, le modèle de Potts, les algorithmes d'ICM et de recuit, ont nettement amélioré nos résultats. La modification du modèle de Potts, pour le rendre adapté aux classes, et l'étude de la texture, ont résolu des erreurs résiduelles et rendu possible une approche "plan d'occupation des sols".

Nous avons restreint notre étude à un certain cadre qui exclut d'"entrer dans la ville". La classification du milieu urbain à une résolution qui permet de distinguer les structures internes, est très différente de celle que l'on applique à une image rurale ou peri-urbaine. D'autre part, nous n'avons pas envisagé la classification sur une image hyperspectrale à plusieurs centaines de bandes. Dans ce cas, un prétraitement des données devient obligatoire afin de limiter la redondance de l'information qui ralentit inutilement tout calcul.

Pour certains paysages, il n'est pas raisonnable d'affecter une classe à un pixel. Il est parfois intéressant d'affecter un pourcentage de chacune des classes, et travailler sur une régularisation de ces pourcentages. Par exemple, une image de forêt à basse résolution peut présenter divers types d'arbres par pixel, et la classification peut viser à étudier la répartition de chaque espèce d'arbre, comme dans la thèse de Sandrine Mathieu [19]. Cela a d'ailleurs été testé en hyperspectral dans le cadre d'un contrat NSF sur une simulation MODIS [13].

Nous avons bien senti l'intérêt d'introduire de l'information contextuelle pour un résultat meilleur. Mais dans la perspective d'appliquer nos algorithmes à des images de taille beaucoup plus importante que nos images tests, la réduction du temps de calcul lors de la classification est un enjeu important. Ainsi, nous devons établir un compromis entre la rapidité des calculs et la qualité du résultat, en concentrant nos efforts sur la transformation de l'information initiale [16].

Le concept de paramètres adaptés à la classe est sans doute l'axe de recherche le plus complexe, mais aussi le plus intéressant. Nous avons évoqué la possibilité d'intégrer des paramètres qui caractérisent non pas les classes, mais la compatibilité entre classes. Cela se ferait néanmoins au prix d'une augmentation du nombre de paramètres.

A propos de paramétrisation, nous avons introduit des paramètres dépendants des bandes lors de la classification, et dépendants des classes lors de la régularisation. Devant le nombre de paramètres introduits, il serait appréciable pour l'utilisateur de ne pas avoir à les fixer manuellement. Ainsi se pose le problème complexe du réglage automatique des paramètres.

Nous avons occulté une grande partie de la classification supervisée qui concerne l'apprentissage. En effet, le choix des classes, l'éventualité de définir des sous-classes (blé mur, blé jeune), le choix des régions d'apprentissage sont autant de problèmes complexes à se

poser lors d'une classification. De plus, cette étape dépend également du prétraitement de l'image. Par exemple, dans un paysage de montagne, on devra définir une classe montagne au soleil, et une classe montagne ombragée. Une correction topographique évite cette séparation en sous-classes qui rend le procédé plus complexe.

Enfin, il serait profitable de comparer les résultats obtenus avec ceux qu'auraient fournis des méthodes variationnelles, développées notamment par Christophe Samson [21], ou des modèles stochastiques [8] [4].

Références

- [1] Besag (J.). – Spatial interaction and the statistical analysis of lattice systems. *J. Royal Statist. Soc. B*, n36, 1974, pp. 192–326.
- [2] Besag (J.). – On the statistical analysis of dirty pictures. *J. Royal Statist. Soc. B*, vol. 3, n48, 1986, pp. 259–302.
- [3] Better Solutions Consulting LLC. – *ENVI Tutorials*, 1999.
- [4] Chardin (A.). – *Modèles énergétiques hiérarchiques pour la résolution des problèmes inverses en analyse d'images*. – Thèse de doctorat - Université de Rennes I, 2000.
- [5] Cocquerez (J.) et Philipp (S.). – *Analyse d'images : filtrage et segmentation*. – Masson, 1995.
- [6] Descombes (X.). – *Champs Markoviens en analyse d'images*. – Thèse de doctorat - ENST, 1993.
- [7] Descombes (X.), Moctezuma (M.), Maître (H.) et Rudant (J.). – Coastline detection by a markovian segmentation on sar images. *Signal Processing*, vol. 55, n1, 1996, pp. 123–132.
- [8] Descombes (X.), Morris (R.) et Zerubia (J.). – Quelques améliorations à la segmentation d'images bayésienne - première partie : modélisation. *Traitement du Signal*, vol. 14, n4, 1997, pp. 373–382.
- [9] Dunteman (G. H.). – *Principal Components Analysis*. – Sage Inc., 1989.
- [10] Geman (S.) et Geman (D.). – Stochastic relaxation, gibbs distribution and the bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, n6, Novembre 1987, pp. 721–741.
- [11] Geman (S.) et Graffigne (C.). – Markov random field image models and their applications to computer vision. *Proceedings of the International Congress of Mathematicians : Berkeley*, Août 1986, pp. 1496–1517.
- [12] Haralick (R.). – Statistical and structural approaches to texture. *Proc. of the IEEE*, vol. 67, n5, Mai 1979, pp. 786–804.
- [13] Kalluri (S. N. V.), Huang (C.), Mathieu-Marni (S.), Townshend (J. R. G.), Yang (K.), Chellappa (R.) et Fleig (A.). – A comparison of mixture modeling algorithms and their applicability to the modis data. *IGARSS'97. 1997 International Geoscience and Remote Sensing Symposium. Remote Sensing - A Scientific Vision for Sustainable Development (Cat. No.97CH36042)*, vol. 1, 1997, pp. 171–173.
- [14] Krishnamachari (S.) et Chellappa (R.). – Multiresolution Gauss-Markov random field models for texture segmentation. *IEEE Trans. on Image Processing*, vol. 6, n2, Février 1997, pp. 251–267.

-
- [15] Landgrebe (D.). – *Information Extraction Principles and Methods for Multispectral and Hyperspectral Image Data*. – Rapport technique, Purdue University, Indiana, 1998.
- [16] Landgrebe (D.) et Jimenez (L. O.). – Supervised classification in high-dimensional space : Geometrical, statistical, and asymptotical properties of multivariate data. *IEEE Transactions on Systems, Man, Cybernetics - Part C*, vol. 28, n1, Février 1998, pp. 39–53.
- [17] Li (S.). – *Markov random field modeling in computer vision*. – Springer Verlag, 1995.
- [18] Lorette (A.). – *Analyse de texture par méthodes markoviennes et par morphologie mathématique : application à l'analyse des zones urbaines sur des images satellitales*. – Thèse de doctorat - UNSA, 1999.
- [19] Mathieu (S.). – *Aide à la Détermination de Faciès Pédopaysagers à partir de Données Satellitaires*. – Thèse de doctorat - Paris 7, 1994.
- [20] Metropolis (N.), Rosenbluth (A.), Rosenbluth (N.), Teller (A.) et Teller (E.). – Equation of state calculations by fast computing machines. *J. Chemical Physics*, n21, 1953, pp. 1087–1091.
- [21] Samson (C.), Blanc-Féraud (L.), Aubert (G.) et Zerubia (J.). – A variational model for image classification and restoration. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, n5, Mai 2000, pp. 460–472.
- [22] Serafim (A. L.). – Segmentation of natural microtextures by joining local and global fractal model parameters. *Proc. SPIE - Int. Soc. Opt. Eng. (USA), Proceedings of the SPIE - The International Society for Optical Engineering*, vol. 3101, 1997, pp. 38–48.
- [23] Sigelle (M.) et Tupin (F.). – *Champs de Markov en Traitement d'Image*. – Ecole Nationale Supérieure des Télécommunications, 1999.
- [24] Soille (P.). – *Morphological Image Analysis*. – Springer, 1999.
- [25] Van Laarhoven (P. J.) et Aarts (E. H.). – *Simulated annealing : theory and applications*. – D.Reidel, 1987.
- [26] Volden (E.), Giraudon (G.) et Berthod (M.). – Modelling image redundancy. *Rapport de recherche - INRIA*, n2440, Décembre 1994.
- [27] Winkler (G.). – *Image analysis, random fields and dynamic Monte-Carlo methods*. – Springer Verlag, 1995.



Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Lorraine : Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)

<http://www.inria.fr>

ISSN 0249-6399