Université de Montréal

Estimation de paramètres de champs markoviens cachés avec applications à la segmentation d'images et la localisation de formes

par

François Destrempes

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures
en vue de l'obtention du grade de
Philosophiae Doctor (Ph.D.)
en informatique

février, 2006

Université de Montréal

Faculté des études supérieures

Cette thèse intitulée :

Estimation de paramètres de champs markoviens cachés avec applications à la segmentation d'images et la localisation de formes

présentée par

François Destrempes

a été évaluée par un jury composé des personnes suivantes:

Yoshua Bengio
(Président-rapporteur et représentant du doyen)

Max Mignotte
(Directeur)

Jean-François Angers
(Co-directeur)

Sébastien Roy
(Membre)

Wojciech Pieczynski
(Examinateur externe)

Thèse acceptée le _____

À ma famille,

RÉSUMÉ

Cette thèse porte principalement sur l'estimation des paramètres de modèles markoviens adaptés à la segmentation d'images naturelles obtenues par acquisition optique. La segmentation d'images est une opération dite de bas-niveau qui permet d'en donner une représentation simplifiée, afin de procéder ensuite à des tâches de haut-niveau, telle la localisation de formes.

Notre travail a porté sur trois volets: 1) l'optimisation stochastique; 2) la modélisation markovienne d'images en vue de la segmentation et l'estimation des paramètres de ces modèles au sens du Maximum A Posteriori (MAP); 3) la localisation de formes.

Nous nous sommes principalement intéressé à un algorithme d'optimisation stochastique de O. François appelé algorithme d'Exploration/Selection (E/S). Cet algorithme évolutionnaire est un recuit simulé généralisé (RSG). Nous démontrons que l'algorithme E/S est valide dans le cadre général d'un graphe d'exploration connexe, mais pas nécessairement symétrique, et d'une distribution d'exploration positive quelconque. Nous présentons également des bornes en temps fini dans le cas d'un graphe d'exploration complet et d'une distribution d'exploration positive quelconque.

Dans le cadre de la modélisation markovienne d'images, nous proposons dans un premier temps un modèle de couple de champs markoviens, formé du champ observable des couleurs et du champ discret caché des régions. Les distributions de vraisemblance sont définies à partir de distributions unimodales, mais non gaussiennes. Le nombre de régions est supposé inconnu. De plus, nous ajoutons des contraintes globales sur la taille et le nombre de régions. Nous proposons également un modèle plus élaboré de triplet de champs markoviens: le champs observable joint des attributs de textures et de couleurs; le champs discret caché des régions; et le champs discret caché

des classes de couleurs et de textons. En plus de considérer un nombre de régions inconnu et une contrainte globale sur la taille et le nombre de régions, nous supposons un hyper-paramètre inconnu pour le modèle markovien du processus des régions. Ce deuxième modèle présente l'avantage d'offir une paramétrisation approchée des ensembles de Julesz plus facile à estimer que le modèle FRAME ("Filters, Random fields And Maximum Entropy"). Le modèle proposé s'applique au cadre plus général de fusion de données. Pour l'estimation des paramètres, notre cadre formel est le paradigme bayésien. Nous proposons une variante originale de l'algorithme E/S qui consiste à utiliser comme noyau d'exploration, un noyau approché de l'échantillonneur de Gibbs. Cette méthode permet de calculer le MAP des deux modèles mentionnés ci-dessus. Nous comparons la méthode proposée avec un MCMC ("Monte Carlo Markov Chains") et un recuit simulé (RS) dans le cadre d'un mélange de noyaux gaussiens en nombre variable. Les tests effectués indiquent un avantage de notre méthode sur les deux autres.

Nous proposons un modèle des déformations d'un patron de formes dont l'*a priori* est un mélange de PPCA ("Probabilistic Principal Component Analysis"), ainsi qu'une méthode de type MCMC pour en estimer les paramètres. De plus, nous proposons une contrainte globale qui exploite une segmentation préalable de l'image en régions basées sur les couleurs. Cette contrainte présente l'avantage majeur de faciliter grandement la recherche stochastique de la forme dans l'image, effectuée à l'aide de l'algorithme E/S. Nous comparons notre méthode de segmentation avec deux autres méthodes dans le cadre de la localisation de formes. Les tests effectués indiquent un net avantage de la méthode proposée au point de vue de la robustesse et de la précision de la localisation.

Mots clés : modèles markoviens, estimation bayésienne, optimisation stochastique, segmentation d'images, modèle de couleurs et textures, fusion, localisation de formes.

ABSTRACT

This thesis relates mainly to estimation of parameters of Markovian models that are adapted to segmentation of natural images obtained by optical acquisition. Image segmentation is a so-called low-level operation that allows giving a simplified representation of the image, in order to perform subsequently high-level tasks, such as localization of shapes.

Our work deals with three aspects: 1) stochastic optimization; 2) Markovian modeling of images in view of segmentation and estimation of the parameters of those models in the sense of the Maximum A Posteriori (MAP); 3) localization of shapes.

We were particularly interested in a stochastic optimization algorithm of O. François called Exploration/Selection algorithm (E/S). This evolutionary algorithm is a generalized simulated annealing (GSA). We show that the E/S algorithm is valid in the general framework of a connected, but not necessarily symmetric, exploration graph, and any positive exploration distribution. We also present finite time bounds in the case of a complete exploration graph, and any positive exploration distribution.

As for Markovian modeling of images, we propose first of all a model made of a couple of Markov random fields; namely, the observable field of colors and a discrete hidden field of regions. The likelihood distributions are defined from unimodal, but non-Gaussian, distributions. The number of regions is supposed unknown. Moreover, we consider global constraints on the size and the number of regions. We propose also a more elaborate model made of a triplet of Markov random fields; namely, the observable field of the joint color and texture features, the discrete hidden field of regions, and the discrete hidden fields of color and texton classes. In addition to

considering an unknown number of regions and a global constraint on the size and the number of regions, we assume an unknown hyper-parameter for the Markovian region process. This second model presents the advantage of offering an aproximate parametrization of Julesz ensembles that is easier to estimate than the FRAME model ("Filters, Random fields And Maximum Entropy"). The proposed model can be applied to the more general framework of data fusion. For the estimation of parameters, our formal framework is the Bayesian paradigm. We propose an original variant of the E/S algorithm that consists of using as exploration kernel an approximate Gibbs sampler. This method allows computing the MAP of the models mentioned above. We compare the proposed method with an MCMC ("Monte Carlo Markov Chains") and a simulated annealing (SA) in the framework of mixtures of an unknown number of Gaussian kernels. Our tests indicate an advantage of our method over the other two methods.

We propose a model for the deformations of a shape for which the *prior* distribution is a mixture of PPCA ("Probabilistic Principal Component Analysis"), as well as an MCMC algorithm for the estimation of the parameters. Moreover, we propose a global constraint that exploits a preliminary color-based segmentation of the image into regions. This constraint presents the major advantage of facilitating greatly the stochastic search of the shape in the image, using the E/S algorithm. We compare our segmentation method with two other methods in the context of localization of shapes. Our tests indicate a clear advantage of the proposed method in terms of robustness and precision of the localization.

Keywords: Markov random field models, Bayesian estimation, stochastic optimization, image segmentation, color and texture model, fusion, localization of shapes.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLES

REMERCIEMENTS

Je souhaite remercier tous ceux qui m'ont apporté le support si nécessaire à la création de cette thèse, tout particulièrement mon directeur de thèse Max Mignotte et mon co-directeur Jean-François Angers. J'aimerais remercier M.M. Yoshua Bengio et Jean Meunier pour m'avoir apporté des suggestions lors de ma soutenance pré-doctorale. J'aimerais également remercier tous les autres membres de mon jury, M.M. Yoshua Bengio, Sébastien Roy, et Wojciech Pieczynski qui a accepté d'être examinateur externe.

Finalement, je remercie tous les membres de ma famille pour leur support constant dans mes études en informatique.

Chapitre 1

INTRODUCTION GÉNÉRALE

Cette thèse porte principalement sur l'estimation des paramètres de modèles markoviens adaptés à la segmentation d'images naturelles obtenues par acquisition optique. La segmentation d'images consiste à classifier les pixels d'une image en régions formées de parties cohérentes d'objets et de l'arrière-plan qui sont présents dans la scène captée par l'image. Cette opération dite de bas-niveau permet d'en donner une représentation simplifiée, afin de procéder ensuite à des tâches de plus haut niveau. Parmi ces dernières tâches, on distingue en outre la détection et reconnaissance d'objets, la détection de mouvement dans des séquences vidéo, la reconstruction d'objets 3-D, l'indexation d'images dans des bases de données, ou encore la restauration d'images. De plus, les paramètres estimés autres que les étiquettes des régions, peuvent s'avérer utiles pour rendre plus robustes les tâches de haut-niveau, ou encore pour aider directement à la détection d'objets comportant des caractéristiques connues *a priori*. L'estimation des paramètres et la segmentation d'images sont donc deux problèmes essentiels en traitement d'images et en vision par ordinateur.

Pour l'estimation des paramètres, nous adoptons le paradigme bayésien. Il s'agit d'un cadre formel qui permet de pondérer les connaissances *a priori* et la vraisemblance des données observables. Nous évitons donc ainsi d'ajuster arbitrairement les paramètres de méthodes algorithmiques. De plus, l'inférence bayésienne permet de considérer diverses distributions *a priori* sur les paramètres à estimer, ainsi que diverses fonctions de pertes. Les propriétés mathématiques de l'approche bayésienne sont utiles, et la sémantique en est riche. L'estimateur proposé ici est un Maxi-

mum *A Posteriori* (MAP) pondéré par des contraintes globales sur le processus des régions. Pour calculer le MAP, nous avons recours à un algorithme d'optimisation stochastique.

Plusieurs méthodes ont été proposées pour segmenter les images naturelles. Il semble difficile de comparer ces diverses méthodes autrement que par une appréciation visuelle, qui est forcément subjective. Dans cette thèse, nous utilisons un test de Turing pour tester la méthode de segmentation proposée, c'est-à-dire que nous nous donnons comme objectif de pouvoir localiser dans une image des objets dont la forme est connue *a priori*. En effet, nous sommes portés à penser que le problème de la segmentation d'images n'a de sens qu'en vue d'une tâche de haut-niveau particulière.

Notre travail a donc porté sur trois volets: 1) l'optimisation stochastique; 2) la modélisation markovienne d'images en vue de la segmentation et l'estimation des paramètres de ces modèles au sens du MAP; 3) la localisation de formes. Les deux premiers articles portent sur les deux premiers volets. Le troisième article est dédié au dernier volet. Nous détaillons maintenant chacun des trois aspects de notre recherche.

Optimisation stochastique

Nous nous sommes principalement intéressé à un algorithme d'optimisation stochastique de O. François appelé algorithme d'Exploration/Sélection (E/S) [59]. Cet algorithme évolutionnaire est un recuit simulé généralisé (RSG) [76]. On retrouve également dans cette famille d'algorithmes le recuit simulé classique (RS) [72], une version parallèle du RS [136], ainsi que l'algorithme génétique de R. Cerf [21, 22]. L'avantage de l'algorithme E/S sur les autres recuits simulés généralisés mentionnés ci-dessus, est que les paramètres internes de l'algorithme E/S ne dépendent (à toute fin pratique) que du diamètre d'un graphe d'exploration, et non de la fonction à minimiser.

Dans la version originale de l'E/S, le graphe d'exploration est supposé connexe et symétrique. De plus, la distribution d'exploration est supposée uniforme sur chaque

voisinage du graphe d'exploration. Dans cette thèse, il est utile de se donner une version plus souple et générale de l'algorithme E/S, c'est-à-dire que nous voulons :

1. un graphe d'exploration connexe, mais pas nécessairement symétrique ;

2. une distribution d'exploration positive quelconque sur chaque voisinage du graphe d'exploration.

Nous allons montrer dans le premier article (section 4.6) que la convergence asymptotique de l'E/S est encore valide dans ce cadre plus souple. En fait, nous allons considérer pour nos besoins une version légèrement plus générale (section 4.2.3).

De plus, nous allons montrer dans le deuxième article (section 6.3.2) comment ramener un graphe d'exploration quelconque au cas d'un graphe complètement connexe (c'est-à-dire, de diamètre égal à 1). Sous cette hypothèse, mais avec une distribution d'exploration quelconque, nous donnerons à la section 6.7 des bornes en temps fini pour la convergence de l'E/S. Comme nous nous ramenons toujours au cas d'un diamètre égal à 1 dans cette thèse, nous aurions pu nous contenter de ce dernier résultat. Toutefois, les résultats de la section 4.6 peuvent s'avérer utiles dans de (nombreuses) autres applications en informatique, comme seul nous le dira l'avenir.

Modélisation d'images en vue de la segmentation et estimation des paramètres de ces modèles au sens du MAP

Dans le premier article (chapitre 4), nous modélisons une image par un couple de champs markoviens, formé du champ observable des couleurs et du champ discret caché des régions. Nous considérons des distributions de vraisemblance définies à partir de distributions unimodales, mais non gaussiennes. La difficulté principale consiste à supposer le nombre de régions inconnu. En effet, l'espace des paramètres à estimer est alors scindé en sous-espaces de dimension variable, ce qui en rend l'exploration plus difficile par les algorithmes de RJMCMC ("Reversible Jump Monte Carlo Markov

Chain") [66] utilisés dans ce contexte. De plus, nous ajoutons à un modèle markovien *a priori* sur le processus de régions, des contraintes globales sur la taille et le nombre de régions. Par contre, l'hyper-paramètre du modèle markovien est fixé.

Dans le deuxième article (chapitre 6), nous considérons un modèle plus élaboré, c'est-à-dire, un triplet de champs markoviens : le champs observable joint des attributs de textures et de couleurs ; le champs discret caché des régions ; et le champs discret caché des classes de couleurs et de textons. Cette fois-ci, les distributions de vraisemblance sont des mélanges de noyaux gaussiens. De plus, nous supposons l'hyper-paramètre du modèle markovien inconnu. Ce deuxième modèle d'images s'inscrit dans un cadre plus général de fusion de données basée sur la notion d'ensembles de Julesz [146], c'est-à-dire un champ de données observables caractérisé par la distribution de ces attributs (et non des attributs eux-mêmes). En outre, notre cadre formel s'aplique à d'autres types d'imagerie. L'innovation principale consiste à considérer une modélisation paramétrique approchée des ensembles de Julesz, plutôt que le modèle FRAME ("Filters, Random fields And Maximum Entropy") [148, 149], qui est beaucoup plus difficile à estimer.

Le problème fondamental est alors d'estimer les paramètres des deux modèles proposés sans phase d'apprentissage supervisé, c'est à dire que nous ne supposons pas la segmentation connue, ni les autres paramètres. La segmentation en régions est alors considérée comme une partie des paramètres à estimer. Dans les deux premiers articles, nous nous proposons de calculer conjointement une segmentation et une estimation des paramètres dans le sens du MAP, y compris le nombre de régions.

Pour ce faire, nous proposons une variante originale de l'algorithme E/S. La nouveauté consiste à utiliser comme noyau d'exploration, un noyau approché de l'échantillonneur de Gibbs. La méthode proposée porte l'acronyme ESE pour Exploration/Sélection/Estimation.

Nous présentons à l'annexe B une comparaison de notre méthode avec un algorithme de type MCMC et un algorithme de type RS dans le cadre d'un mélange de

noyaux gaussiens en nombre variable. Nos tests démontrent que la procédure ESE est en moyenne plus efficace que la méthode du MCMC habituellement utilisée en statistiques, ou que le RS.

Dans l'annexe D, nous comparons notre méthode avec l'algorithme E/S dans le cadre du calcul du mode d'un mélange de deux noyaux gaussiens. Nos tests démontrent que l'algorithme E/S est plus efficace lorsque le domaine de définition englobe de près la solution cherchée. Par contre, notre méthode est plus efficace que l'algorithme E/S sur un domaine de définition plus étendu.

Localisation de formes

Le problème de la localisation de formes consiste à localiser dans une image un objet dont la forme est connue *a priori*. Dans mon mémoire de maîtrise, nous avions présenté un modèle qui comportait une distribution *a priori* basée sur les résultats d'une PPCA ("Probabilistic Principal Component Analysis") [134], et une distribution de vraisemblance qui utilise un modèle statistique des contours dans une image. Dans cette thèse, nous proposons un modèle des déformations dont la distribution *a priori* peut être un mélange de PPCA, ainsi qu'une méthode de type MCMC pour en estimer les paramètres. De plus, outre le modèle statistique des contours, nous proposons une contrainte globale qui exploite une segmentation préalable de l'image en régions basées sur les couleurs. Cette contrainte présente l'avantage majeur de faciliter grandement la recherche stochastique de la forme dans l'image, effectuée à l'aide de l'algorithme E/S. Nous comparons notre méthode de segmentation avec deux autres méthodes dans le cadre de la localisation de formes. Les tests effectués indiquent un net avantage de la méthode proposée relativement à la fiabilité et qualité de la localisation. Par contre, le temps de calcul requis est plus important.

Organisation de cette thèse

Nous présentons d'abord au chapitre 2 les notions portant sur le recuit simulé généralisé, et plus particulièrement l'algorithme E/S. Nous donnons quelques précisions techniques sur la contribution de cette thèse à l'algorithme E/S.

Au chapitre 3, nous rappellons brièvement quelques éléments du formalisme bayésien. Nous présentons quelques méthodes de simulations de distributions, dont le RJMCMC qui est utilisé dans le contexte de modèles à dimension variable. Nous rappellons la notion de modèles markoviens ainsi qu'un exemple simple suffisant pour comprendre le premier article. Finalement, nous passons en revue les principales méthodes d'estimation de paramètres et de segmentation d'images utilisées en traitement d'images dans le contexte de modèles de Markov. Le premier article est présenté au chapitre 4 dans sa version publiée dans *IEEE Transactions on Image Processing*. La méthode ESE de calcul du MAP y est présentée dans le cadre de couples de champs de Markov.

Au chapitre 5, nous présentons la notion de textures au sens de Julesz, ainsi que le modèle FRAME. Nous expliquons également quelques notions sur les modèles génériques et structurels d'images naturelles. Le deuxième article est présenté au chapitre 6 dans sa version acceptée dans *IEEE Transactions on Image Processing*. La méthode ESE de calcul du MAP y est présentée dans le cadre de triplets de champs de Markov. Ce modèle permet de formaliser la fusion de données au sens des ensembles de Julesz.

Nous présentons au chapitre 7, une description de trois modèles de localisation de formes, reprise de mon mémoire de maîtrise. Le troisième article est présenté au chapitre 8 dans sa version soumise. Nous y présentons le nouveau modèle de mélange de PPCA, ainsi que sa méthode d'estimation. Nous introduisons également deux types de contraintes globales basées sur une segmentation couleur de l'image.

Finalement, nous concluons par une discussion des résultats au chapitre 9. Nous

y mentionnons également quelques avenues de recherche.

Chapitre 2

OPTIMISATION STOCHASTIQUE

2.1 Introduction

Ce chapitre présente les notions et résultats nécessaires à la bonne compréhension des algorithmes stochastiques utilisés dans cette thèse pour le calcul du MAP (chapitres 4 et 6) et pour la localisation de formes (chapitre 8).

Le recuit simulé généralisé (RSG) [76, 137, 138] est une famille d'algorithmes d'optimisation qui inclue l'algorithme évolutionnaire d'Exploration/Sélection (E/S) de O. François [59], le recuit simulé classique (RS) [72], une version parallèle du RS [136], ainsi que l'algorithme génétique de R. Cerf [21, 22]. Les paramètres internes de l'algorithme E/S ne dépendent (à toute fin pratique) que du diamètre d'un graphe d'exploration, et non de la fonction à minimiser. Cet avantage majeur sur les autres algorithmes connus de type RSG en fait un candidat de choix pour nos algorithmes.

2.2 Recuit simulé généralisé

Nous présentons tout d'abord la définition d'un RSG [59, 137].

Définition 1

Soit E un ensemble fini et π un noyau markovien sur E. On dit que π est irréductible *si pour tout $\mathbf{x}, \mathbf{y} \in E$, il existe une suite $(\mathbf{x}_k)_{0 \leq k \leq n}$ avec $\mathbf{x}_0 = \mathbf{x}$ et $\mathbf{x}_n = \mathbf{y}$ qui satisfait la condition $\pi(\mathbf{x}_k, \mathbf{x}_{k+1}) > 0$ pour $k \leq n-1$.*

Définition 2

Soit $(q_T)_{T>0}$ une famille de noyaux markoviens sur E et $\kappa \geq 1$. On dit que $(q_T)_{T>0}$ est admissible *pour π et κ si il existe une fonction $V_1 : E \times E \rightarrow [0, \infty]$ (appelée* fonction de coût de communication*) telle que*

$$V_1(\mathbf{x},\mathbf{y}) < \infty \text{ ssi } \pi(\mathbf{x},\mathbf{y}) > 0; \tag{2.1}$$

pour tout $T > 0$ *et tout* $\mathbf{x},\mathbf{y} \in E$,

$$\frac{1}{\kappa}\pi(\mathbf{x},\mathbf{y})e^{-V_1(\mathbf{x},\mathbf{y})/T} \leq q_T(\mathbf{x},\mathbf{y}) \leq \kappa\pi(\mathbf{x},\mathbf{y})e^{-V_1(\mathbf{x},\mathbf{y})/T}. \tag{2.2}$$

Définition 3

Soit $(X_t)_{t\in\mathbb{N}}$ *une chaîne de Markov sur* E. *On dit que cette chaîne est un* recuit simulé généralisé *(RSG) avec paramètres* $(E,\pi,\kappa,V_1,(q_T)_{T>0},(T(t))_{t\in\mathbb{N}})$, *où* $(T(t))_{t\in\mathbb{N}}$ *est une suite décroissante de nombres réels positifs, si la famille* $(q_T)_{T>0}$ *de noyaux markoviens sur* E *est admissible pour* π *et* κ, *avec fonction de coût de communication* V_1, *et si:*

$$P(X_{t+1} = \mathbf{y} \mid X_t = \mathbf{x}) = q_{T(t)}(\mathbf{x},\mathbf{y}), \qquad \mathbf{x},\mathbf{y} \in E. \tag{2.3}$$

Le paramètre T est appelé *la température*. Nous utiliserons aussi la notation X_t^T pour insister sur le fait qu'à l'itération t la température est $T = T(t)$.

Avant d'aller plus loin dans les définitions relatives au RSG, nous allons présenter l'exemple principal considéré dans cette thèse.

2.3 L'algorithme d'optimisation Exploration/Sélection (E/S)

Le but de l'algorithme E/S [58, 59] est de minimiser une fonction f sur un espace d'états fini A. Nous considérons une structure de graphe $\mathcal{G}$ sur A, appelé *graphe d'exploration*, qui est supposé connexe et symétrique. Pour chaque élément $x \in A$, $N(x)$ dénote le voisinage de x dans le graphe $\mathcal{G}$. Pour chaque $x \in A$, une distribution *positive* $a(x,\cdot)$ est définie sur le voisinage $N(x)$ de x. Étant donné $m \geq 2$, un élément $\mathbf{x} = (x_1,...,x_m)$ du produit cartésien $E = A^m$ est appelé une *population* (de solutions). Étant donné une population $\mathbf{x} = (x_1,...,x_m)$, $\alpha(\mathbf{x})$ dénotera la meilleure solution

d'indice minimal: $\alpha(\mathbf{x}) = x_l$ tel que $f(x_k) > f(x_l)$, pour $1 \leq k < l$, et $f(x_k) \geq f(x_l)$, pour $l < k \leq m$.

L'algorithme E/S s'énonce ainsi.

Initialisation: Choisir au hasard la population initiale $\mathbf{x} = (x_1, x_2, ..., x_m)$.

Répéter jusqu'à un critère d'arrêt:

Mise à jour de la meilleure solution courante: déterminer $\alpha(\mathbf{x})$ à partir de la population courante $\mathbf{x}$, selon les valeurs de la fonction f.

Exploration/sélection: pour chaque $l = 1, ..., m$, remplacer avec probabilité p, x_l par $y_l \in N(x_l)$ selon la distribution $a(x_l, y_l)$; sinon, remplacer x_l par $\alpha(\mathbf{x})$ (avec probabilité $1 - p$). Décroître p.

La probabilité p est appelée *probabilité d'exploration*. Dans [59], à l'étape de l'exploration, l'élément y_l est pris dans $N(x_l) \setminus \{\alpha(\mathbf{x})\}$. De plus, les auteurs supposent que la distribution d'exploration est uniforme. Plus exactement, les auteurs définissent comme suit un noyau d'exploration $\hat{a}(\mathbf{x}, \cdot)$ avec $\mathbf{x}= (x_l) \in A^m$, plutôt que $a(x, \cdot)$ avec $x \in A$:

$$\hat{a}(\mathbf{x}, y_l) = \begin{cases} 1/\deg x_l, & \text{si } y_l \in N(x_l) \setminus \{\alpha(\mathbf{x}), x_l\}; \\ 0, & \text{si } y_l \in N(x_l)^C \cup \{\alpha(\mathbf{x})\}; \\ 1 - \sum_{y'_l \neq x_l} \hat{a}(\mathbf{x}, y'_l) & \text{si } y_l = x_l. \end{cases} \tag{2.4}$$

Notons que par définition $0 \leq \sum_{y'_l \neq x_l} \hat{a}(\mathbf{x}, y'_l) \leq 1$. Il en découle que $\hat{a}(\mathbf{x}, \cdot)$ définit une mesure de probabilité sur A. Nous obtenons le noyau de transition suivant

$$P(X_{t+1} = \mathbf{y} \mid X_t = \mathbf{x}) = \prod_{l=1}^{m} (p\hat{a}(\mathbf{x}, y_l) + (1 - p)\delta(y_l = \alpha(\mathbf{x})), \tag{2.5}$$

où δ est le symbole de Kronecker.

Soit maintenant $I(\mathbf{x}, \mathbf{y}) = \{l \ : \ 1 \leq l \leq m, y_l \neq \alpha(\mathbf{x})\}$. Posons

$$\pi(\mathbf{x},\mathbf{y}) = \prod_{l\in I(\mathbf{x},\mathbf{y})} \hat{a}(\mathbf{x},y_l); \tag{2.6}$$

$$V_1(\mathbf{x},\mathbf{y}) = \begin{cases} |I(\mathbf{x},\mathbf{y})|, \text{ si } \pi(\mathbf{x},\mathbf{y})>0; \\ \infty, \text{ sinon.} \end{cases} \tag{2.7}$$

En posant $p_T = \exp(-1/T)$, où $T > 0$ est la température, nous pouvons voir que l'algorithme E/S est un recuit simulé généralisé avec paramètres (partiels) (E, π, V_1).

2.4 Théorème de convergence du RSG

Nous nous replaçons maintenant dans le cadre d'un RSG $(X_t)_{t\in\mathbb{N}}$ avec paramètres $(E, \pi, \kappa, V_1, (q_T)_{T>0}, (T(t))_{t\in\mathbb{N}})$. Deux notions sont fondamentales pour comprendre l'énoncé du théorème de convergence du RSG : l'énergie virtuelle du RSG, et sa hauteur critique. En résumé, la hauteur critique permet d'ajuster le taux de décroissance de la température qui assure la convergence asymptotique du RSG vers un minimum (global) de son énergie virtuelle. La hauteur critique peut être considérée comme l'effort à fournir au RSG pour sortir d'un minimum local. Voici tout d'abord deux définitions techniques.

Définition 4

Un $\mathbf{x}$*-graphe* g *sur* E *est un ensemble d'arcs* $g = \{\mathbf{y}\to\mathbf{z}\}$ *(*$\mathbf{y},\mathbf{z}\in E$, $\mathbf{y}\neq\mathbf{x},\mathbf{z}$*) tels que chaque élément de* $E\setminus\{\mathbf{x}\}$ *est le point initial d'un et un seul arc, et puisse mener à* $\mathbf{x}$ *par une suite d'arcs. Si* $\mathbf{x}\in E$*, l'ensemble des* $\mathbf{x}$*-graphes est noté* $G(\mathbf{x})$*.*

Ne pas confondre un $\mathbf{x}$-graphe g avec le graphe d'exploration $\mathcal{G}$ dans le cas de l'algorithme E/S.

Définition 5

Le coût de communication *de* $\mathbf{x}$ *à* $\mathbf{y}$ *est défini par*

$$V(\mathbf{x},\mathbf{y}) = \min_{(\mathbf{x}_k)_{k=0}^{k=r} : \mathbf{x}_0=\mathbf{x},\, \mathbf{x}_r=\mathbf{y}} \sum_{k=0}^{r-1} V_1(\mathbf{x}_k, \mathbf{x}_{k+1}). \tag{2.8}$$

Notons que $V(\mathbf{x},\mathbf{y})$ *est le coût du plus court chemin de* $\mathbf{x}$ *à* $\mathbf{y}$.

Nous pouvons maintenant donner la définition formelle de l'énergie virtuelle d'un RSG.

Définition 6

L'énergie virtuelle *de* $\mathbf{x} \in E$ *est définie par* $W(\mathbf{x}) = \min_{g\in G(\mathbf{x})} V(g)$, *où* $V(g) = \sum_{\mathbf{y}\to\mathbf{z}\in g} V(\mathbf{y},\mathbf{z})$. *La valeur minimale de* W *est notée* W_* *et l'ensemble des points de* E *pour lesquels* W *est minimale est noté* $\mathcal{W}_*$.

Comme il est mentionné dans [59], on obtient de [60], p.186, l'interprétation suivante pour l'énergie virtuelle d'un RSG. Soit μ_T la distribution stationnaire du noyau q_T. Alors, quelque soit $\mathbf{x} \in E$, $\lim_{T\to 0} -T \log \mu_T(\mathbf{x}) = W(\mathbf{x}) - W_*$.

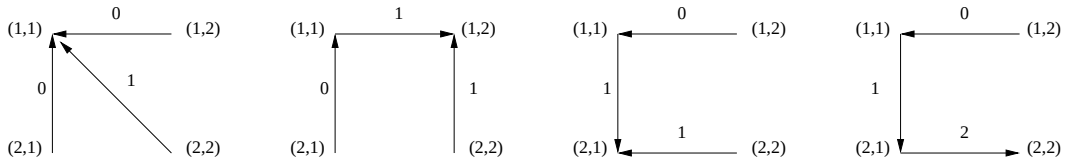
Afin d'illustrer les concepts présentés ci-dessus, considérons l'algorithme E/S sur l'ensemble $A = \{1,2\}$ muni de la fonction à minimiser $f(1) = 1$ et $f(2) = 2$. En posant $m = 2$, nous obtenons $E = \{(1,1),(1,2),(2,1),(2,2)\}$. En utilisant la définition (2.7), nous calculons les coûts de communication V_1 suivants :

V_1	(1,1)	(1,2)	(2,1)	(2,2)
$(1,1)$	0	1	1	2
$(1,2)$	0	1	1	2
$(2,1)$	0	1	1	2
$(2,2)$	2	1	1	0

En utilisant la définition (2.8), nous obtenons les coûts de communication V suivants :

V	(1,1)	(1,2)	(2,1)	(2,2)
$(1,1)$	0	1	1	2
$(1,2)$	0	1	1	2
$(2,1)$	0	1	1	2
$(2,2)$	1	1	1	0

Pour chacun des éléments $\mathbf{x}$ de E, nous obtenons les $\mathbf{x}$-graphes de coût minimal suivants :



Nous déduisons les valeurs suivantes pour l'énergie virtuelle : $W((1,1)) = 1$, $W((1,2)) = W((2,1)) = 2$, et $W((2,2)) = 3$.

Nous présentons maintenant la définition d'altitude de communication entre deux éléments distincts de E.

Définition 7

Soient $\mathbf{x} \neq \mathbf{y} \in E$, *et* $\gamma_{\mathbf{x},\mathbf{y}} = (\mathbf{x}_0 = \mathbf{x}, \mathbf{x}_1, ..., \mathbf{x}_r = \mathbf{y})$ *une trajectoire entre* $\mathbf{x}$ *et* $\mathbf{y}$ *qui ne se recoupe pas. On définit l'élévation par*

$$H(\gamma_{\mathbf{x},\mathbf{y}}) = \max_{0 \leq k < r} \{W(\mathbf{x}_k) + V(\mathbf{x}_k, \mathbf{x}_{k+1})\}. \tag{2.9}$$

L'altitude de communication $H(\mathbf{x}, \mathbf{y})$ *entre* $\mathbf{x}$ *et* $\mathbf{y}$ *est le minimum de* $H(\gamma_{\mathbf{x},\mathbf{y}})$ *sur l'ensemble des trajectoires* $\gamma_{\mathbf{x},\mathbf{y}}$ *entre* $\mathbf{x}$ *et* $\mathbf{y}$ *qui ne se recoupent pas. Cette fonction s'avère être symétrique en* $\mathbf{x}$ *et* $\mathbf{y}$ *[137].*

Définition 8

Un cycle π *est un sous-ensemble non vide de* E *tel que pour une certaine valeur de* $\lambda \geq 0$, *on a:*

$$W(\mathbf{x}) \leq \lambda \text{ pour tout } \mathbf{x} \in \pi; \tag{2.10}$$

$$H(\mathbf{x}, \mathbf{y}) \leq \lambda \text{ pour tout } \mathbf{x} \neq \mathbf{y} \in \pi. \tag{2.11}$$

De plus, un singleton quelconque de E est également appelé un cycle.

Il peut être en fait vérifié qu'un cycle est une classe d'équivalence pour une certaine relation d'équivalence définie sur l'ensemble $W_\lambda = \{\mathbf{x} \ : \ W(\mathbf{x}) \leq \lambda\}$, mais nous n'aurons pas besoin de cette technicalité.

Définition 9

Soit π un cycle et $\mathbf{x} \in \pi$. On appelle hauteur de sortie *du cycle la quantité*

$$H_e(\pi) = \lim_{T \to 0} T \log E[\tau_\pi \mid X_0 = \mathbf{x}], \tag{2.12}$$

où τ_π est le temps de sortie de π. Il s'avère que $H_e(\pi)$ est indépendante de $\mathbf{x} \in \pi$.

Dans [137], une définition algorithmique des hauteurs de sortie est présentée. Fort heureusement, nous n'aurons pas à manipuler directement cette définition fastidieuse dans ce qui suit. Nous donnons finalement la définition de la hauteur critique d'un RSG.

Définition 10

La hauteur critique *est la quantité géométrique*

$$H_1 = \max\{H_e(\pi) \ : \ \pi \cap \mathcal{W}_* = \emptyset\}. \tag{2.13}$$

Le théorème suivant met en lumière l'importance de la hauteur critique pour ajuster la température afin d'assurer la convergence de l'énergie virtuelle vers un minimum global.

Théorème 1 (Trouvé [137])

Pour toute suite $(T(t))_{t \geq 0}$ convergente vers 0, on a

$$\lim_{t\to\infty} \sup_{\mathbf{x}} P(X_t \notin \mathcal{W}_* \mid X_0 = \mathbf{x}) = 0$$

si et seulement si $\sum_{t=0}^{\infty} \exp(-H_1/T(t)) = \infty$, *où* H_1 *est la hauteur critique.*

Typiquement, nous choisissons $T(t) = H_1/\log(t+2)$, pour $t \geq 0$, comme dans le cas du RS classique. En pratique, on se contente souvent de calculer une majorante de la hauteur critique. En effet, nous avons le corollaire immédiat suivant.

Corollaire 1

Supposons que $H_1 \leq \tau < \infty$. *Soit la suite* $T(t) = \tau/\log(t+2)$. *Alors, on a*

$$\lim_{t\to\infty} \sup_{\mathbf{x}} P(X_t \notin \mathcal{W}_* \mid X_0 = \mathbf{x}) = 0.$$

2.5 Théorème de convergence de l'E/S

Dans le cas de l'algorithme E/S, nous pouvons appliquer directement le théorème 1. Mais encore faut-il connaître une majorante de la hauteur critique qui soit exploitable en pratique. De plus, il faut connaître l'ensemble des minima de l'énergie virtuelle. Les propositions suivantes répondent à ces deux questions dans le cas de l'algorithme E/S.

Soit U l'ensemble $\{\mathbf{x} \in E = A^m \ : \ x_1 = x_2 = ... = x_m\}$. Nous identifions l'ensemble des minima A_* de f sur A avec son injection naturelle dans U.

Proposition 1 (François [59])

Soit D *le diamètre du graphe d'exploration. Si* $m > D$, *alors* $\mathcal{W}_* \subset A_*$.

Proposition 2 (François [59])

Si $m > D$, *alors* $H_1 \leq D$.

Nous obtenons dans le cas de l'exemple simple présenté à la section 2.4, la valeur $H_1 = 1$ en utilisant le théorème 8 de [59].

Nous obtenons le corollaire suivant.

Corollaire 2

Soit D le diamètre d'exploration. Soient $m > D$ et $\tau \geq D$. Soit la suite $T(t) = \frac{\tau}{\log(t+2)}$. Alors, on a

$$\lim_{t \to \infty} \sup_{\mathbf{x}} P(X_t \notin A_* \mid X_0 = \mathbf{x}) = 0.$$

2.6 Contribution de cette thèse sur l'E/S

Dans cette thèse, il est utile de se donner une version plus souple et générale de l'algorithme E/S, c'est-à-dire que nous voulons :

1. un graphe d'exploration connexe, mais pas nécessairement symétrique ;
2. une distribution d'exploration positive $a(x, \cdot)$, avec $x \in A$, quelconque (et donc pas nécessairement uniforme sur le voisinage $N(x)$) ;
3. qu'à l'étape d'exploration, il ne soit pas nécessaire de prendre $y_l \neq \alpha(\mathbf{x})$.

Nous allons montrer dans le premier article (section 4.6) que les deux propositions de la section 2.5 sont encore valides dans ce cadre plus souple. En fait, nous allons considérer pour nos besoins la version légèrement plus générale de la section 4.2.3. La preuve présentée est relativement simple, car elle fait surtout appel à des propriétés "haut-niveau" du RSG, plutôt que de manier directement les définitions techniques présentées ci-dessus. Dans le cas particulier de [59], les preuves sont sensiblement plus difficiles, car les auteurs y calculent la valeur exacte de H_1 et ils décrivent complètement l'ensemble $\mathcal{W}_*$. Notez que pour l'informaticien, ces deux précisions ne sont guère exploitables. Dès lors, notre cadre formel suffit amplement, à toute fin pratique.

De plus, nous allons montrer dans le deuxième article (section 6.3.2) comment ramener un graphe d'exploration quelconque au cas d'un graphe complètement connexe ($D = 1$). Dans le cadre d'un diamètre de 1, mais (bien sûr) d'une distribution

d'exploration quelconque, nous donnerons à la section 6.7 des bornes en temps fini pour la convergence de l'E/S. La preuve présentée est élémentaire, en ce sens qu'elle ne nécessite aucune notion sur le RSG.

Chapitre 3

ESTIMATION DE PARAMÈTRES ET SEGMENTATION D'IMAGES

3.1 Introduction

Ce chapitre présente les notions et résultats concernant l'estimation de paramètres nécessaires à la bonne compréhension de la méthode proposée dans cette thèse pour le calcul du MAP (chapitres 4 et 6). Nous y présentons également un modèle simple d'images comme préalable pour la compréhension du premier article. Des modèles plus élaborés seront présentés au chapitre 5.

3.2 Estimateurs bayésiens

En ce qui a trait à l'estimation de modèles, nous adoptons le paradigme de l'analyse bayésienne. Nous présentons d'abord très brièvement la notion d'estimateur bayésien généralisé.

Soit y un échantillon (indépendant ou non) de variables aléatoires. Nous considérons les notions suivantes :

1. la distribution de vraisemblance jointe $P(y \mid \theta)$ paramétrée par un vecteur θ ;

2. une distribution *a priori* $P(\theta)$ des paramètres à estimer ;

3. une fonction de perte $E(\Theta, \theta)$.

La distribution *a posteriori* des paramètres étant donné l'échantillon s'exprime selon le théorème de Bayes par

$$P(\theta \mid y) = \frac{P(y \mid \theta)P(\theta)}{P(y)} \propto P(y \mid \theta)P(\theta). \tag{3.1}$$

Nous obtenons l'estimateur bayésien

$$\begin{aligned} \hat{\theta} &= \arg\min_{\theta} \int E(\Theta, \theta) P(\Theta \mid y)\, d\Theta \\ &= \arg\min_{\theta} \int E(\Theta, \theta) P(y \mid \Theta) P(\Theta)\, d\Theta. \end{aligned} \tag{3.2}$$

Voici des exemples d'estimateurs bayésiens ([75]) :

1. Maximum de vraisemblance (MV). Nous posons $P(\theta) \propto 1$ et $E(\Theta, \theta) = 1 - \delta(\Theta, \theta)$, où δ est la fonction delta de Dirac. Nous obtenons $\hat{\theta} = \arg\max_{\theta} P(y|\theta)$.

2. Maximum *a posteriori* (MAP). En posant $E(\Theta, \theta) = 1 - \delta(\Theta, \theta)$, nous obtenons $\hat{\theta} =$ mode de $P(\theta \mid y)$.

3. Espérance. En posant $E(\Theta, \theta) = ||\Theta - \theta||_2^2$ (la norme L_2 au carré), nous obtenons $\hat{\theta} =$ espérance mathématique de $P(\theta \mid y)$.

4. Médiane. En posant $E(\Theta, \theta) = ||\Theta - \theta||_1$ (la norme L_1), nous obtenons $\hat{\theta} =$ médiane de $P(\theta \mid y)$.

5. Minimum d'une fonction de coût. Soit $\rho(\theta, y)$ une fonction de coût globale. Nous posons artificiellement $P(\theta \mid y) \propto e^{-\rho(\theta, y)}$ et $E(\Theta, \theta) = 1 - \delta(\Theta, \theta)$. Nous obtenons $\hat{\theta} = \arg\min_{\theta} \rho(\theta, y)$.

Le critère 5 n'est présenté ici que pour établir un lien avec la méthode de segmentation [100].

3.3 Méthodes de simulation

Soit une distribution *a posteriori* $P(\theta \mid y)$. Nous nous intéressons aux algorithmes stochastiques décrits par une chaîne de Markov de type Monte Carlo (MCMC), c'est-à-dire, pour laquelle le noyau de transition $P(\theta^{[t+1]} \mid \theta^{[t]}, y)$ admet le comportement asymptotique $\lim_{t\to\infty} P(\theta^{[t]} = \theta \mid \theta^{[0]}, y) = P(\theta \mid y)$. Rappelons que $P(\theta^{[t]} \mid \theta^{[0]}, y) = \int_{\theta^{[1]}} \cdots \int_{\theta^{[t-1]}} \prod_{k=0}^{t-1} P(\theta^{[k+1]} \mid \theta^{[k]}, y)\, d\,\theta^{[1]} \cdots d\,\theta^{[t-1]}$.

Le MCMC est un choix naturel pour le calcul des estimateurs de la moyenne ou de la médiane. Dans le cas du calcul d'un mode (MV ou MAP), il est également standard d'utiliser le MCMC, en retenant la meilleure solution courante.

Parmi les algorithmes de type MCMC, on retrouve principalement l'échantillonneur de Gibbs [63], ainsi que l'algorithme de Metropolis-Hastings [73, 107] et ses nombreuses variantes [20, 66, 67, 85, 123, 131, 133, 139].

3.3.1 Échantillonneur de Gibbs

L'échantillonneur de Gibbs peut s'énoncer ainsi. Soit $\theta = (\theta_1, ..., \theta_r)$.

1. **Initialisation:** Initialisation quelconque de $\theta^{[0]}$.

2. **Échantillonneur:** Simuler chacune des variables $\theta_1, ..., \theta_r$:

$$\begin{aligned}
\theta_1^{[t+1]} &\sim P(\theta_1 \mid \theta_2^{[t]}, \theta_3^{[t]}, ..., \theta_r^{[t]}, y);\\
\theta_2^{[t+1]} &\sim P(\theta_2 \mid \theta_1^{[t+1]}, \theta_3^{[t]}, ..., \theta_r^{[t]}, y);\\
\theta_3^{[t+1]} &\sim P(\theta_3 \mid \theta_1^{[t+1]}, \theta_2^{[t+1]}, ..., \theta_r^{[t]}, y);\\
&\cdots\\
\theta_r^{[t+1]} &\sim P(\theta_r \mid \theta_1^{[t+1]}, \theta_2^{[t+1]}, ..., \theta_{r-1}^{[t+1]}, y).
\end{aligned}$$

3. Augmenter t de 1 et retour en 2.

3.3.2 Algorithme de Metropolis-Hastings

L'algorithme de Metropolis-Hastings peut s'énoncer ainsi dans sa forme la plus simple.

1. **Initialisation:** Initialisation quelconque de $\theta^{[0]}$.

2. **Proposition:** Engendrer θ' selon une distribution de proposition $q(\theta' \mid \theta^{[t]}, y)$.

3. **Disposition:** Poser $\theta^{[t+1]} = \theta'$ avec probabilité

$$\alpha = \min\Big\{1, \frac{P(\theta' \mid y)q(\theta^{[t]} \mid \theta', y)}{P(\theta^{[t]} \mid y)q(\theta' \mid \theta^{[t]}, y)}\Big\} \tag{3.3}$$

   (*acceptation* de la proposition). Sinon, poser $\theta^{[t+1]} = \theta^{[t]}$ (*rejet* de la proposition). Augmenter t de 1 et retour en 2.

3.3.3 MCMC avec saut réversible (RJMCMC)

Récemment, l'algorithme RJMCMC ("Reversible Jump Markov Chain Monte Carlo") [66, 123] a été proposé dans un cadre formel où la dimension de θ peut varier. L'algorithme ressemble à l'algorithme de Metropolis-Hastings, mais avec un mécanisme pour tenir compte du passage d'une dimension à une autre.

Formellement, les auteurs considèrent un ensemble énumérable de modèles $\{\mathcal{M}_k\}$. Chaque modèle est paramétrisé par un vecteur $\theta^{(k)} \in \mathbb{R}^{d_k}$, dont la dimension d_k peut varier d'un modèle à l'autre (ne pas confondre $\theta^{(k)}$ avec la valeur $\theta^{[k]}$ des paramètres à l'itération k). Pour chacun des modèles, il y a une distribution de vraisemblance $P(y \mid k, \theta^{(k)})$. De plus, il faut se donner la probabilité $P(k)$ de chaque modèle, ainsi que la densité *a priori* $P(\theta^{(k)} \mid k)$ des paramètres de chaque modèle.

Par exemple, pour chaque valeur de $k \in \{1, 2, ..., K\}$, un modèle peut consister en un mélange de k noyaux gaussiens de dimension d, $P(y|k, \theta^{(k)}) = \sum_{i=1}^{k} \pi_i \mathcal{N}(y; \mu_i, \Sigma_i)$. Le vecteur de paramètres $\theta^{(k)}$ consiste en $(\pi_1, ..., \pi_k, \mu_1, ..., \mu_k, \Sigma_1, ..., \Sigma_k)$, et il est de dimension variable $d_k = k(d+1)(d+2)/2 - 1$.

Dans ce cadre formel, les auteurs proposent de simuler $(k, \theta^{(k)})$ selon la distribution *a posteriori* $P(k, \theta^{(k)} \mid y)$. Pour ce faire, les auteurs présentent la technique suivante de saut en dimension [66].

Fixons k_1 et k_2. Les auteurs considèrent deux vecteurs aléatoires auxilliaires $u^{(1)} \in \mathbb{R}^{m_1}$ et $u^{(2)} \in \mathbb{R}^{m_2}$, tels que $d_{k_1} + m_1 = d_{k_2} + m_2$, ainsi qu'un difféomorphisme $\varphi = (\varphi_\theta, \varphi_u)$ entre $(\theta^{(k_1)}, u^{(1)})$ et $(\theta^{(k_2)}, u^{(2)})$. Ils considèrent deux densités de proposition $q_1(u^{(1)} \mid \theta^{(k_1)}, y)$ et $q_2(u^{(2)} \mid \theta^{(k_2)}, y)$. Lorsque $u^{(1)}$ est simulé selon la distribution q_1, nous pouvons donc calculer de façon déterministe la valeur de $\theta^{(k_2)}$, en utilisant la fonction φ_θ. Inversement, en simulant $u^{(2)}$ selon la distribution q_2, nous retrouvons de façon déterministe la valeur de $\theta^{(k_1)}$. Finalement, $j(k_1, \theta^{(k_1)})$ dénote la probabilité d'effectuer le premier saut, et $j(k_2, \theta^{(k_2)})$ la probabilité d'effectuer le saut inverse.

Une étape du RJMCMC peut s'énoncer ainsi.

1. Proposer le saut de type $(k_1, \theta^{(k_1)})$ à $(k_2, \theta^{(k_2)})$ avec probabilité $j(k_1, \theta^{(k_1)})$.

2. Simuler $u^{(1)}$ selon la densité de proposition $q_1(u^{(1)} \mid \theta^{(k_1)}, y)$.

3. Calculer la valeur $(\theta^{(k_2)}, u^{(2)}) = \varphi(\theta^{(k_1)}, u^{(1)})$.

4. **Disposition:** Accepter le saut de $(k_1, \theta^{(k_1)})$ à $(k_2, \theta^{(k_2)})$ avec probabilité

$$\alpha = \min\Big\{1, \frac{P(k_2, \theta^{(k_2)} \mid y) q_2(u^{(2)} \mid \theta^{(k_2)}, y) j(k_2, \theta^{(k_2)})}{P(k_1, \theta^{(k_1)} \mid y) q_1(u^{(1)} \mid \theta^{(k_1)}, y) j(k_1, \theta^{(k_1)})} \Big| \frac{\partial(\varphi)}{\partial(\theta^{(k_1)}, u^{(1)})} \Big| \Big\} \qquad (3.4)$$

(*acceptation* de la proposition). Sinon, conserver $(k_1, \theta^{(k_1)})$.

Mentionnons également le cas particulier où $d_{k_1} + m_1 = d_{k_2}$, c'est-à-dire qu'il y a un difféomorphisme φ entre $(\theta^{(k_1)}, u^{(1)})$ et $\theta^{(k_2)}$. Il n'est alors plus nécessaire de considérer la densité de proposition $q_2(u^{(2)} \mid \theta^{(k_2)}, y)$. Lorsque $u^{(1)}$ est simulé selon la distribution q_1, nous pouvons donc calculer de façon déterministe la valeur de $\theta^{(k_2)}$,

en utilisant directement la fonction φ. Inversement, nous retrouvons directement la valeur de $\theta^{(k_1)}$ à partir de $\theta^{(k_2)}$. Dans ce cas-ci, la probabilité d'acceptation s'exprime par :

$$\alpha = \min\Big\{1, \frac{P(k_2, \theta^{(k_2)} \mid y) j(k_2, \theta^{(k_2)})}{P(k_1, \theta^{(k_1)} \mid y) q_1(u^{(1)} \mid \theta^{(k_1)}, y) j(k_1, \theta^{(k_1)})} \Big| \frac{\partial(\varphi)}{\partial(\theta^{(k_1)}, u^{(1)})} \Big| \Big\}. \tag{3.5}$$

3.3.4 Autres variantes du MCMC

Parmi les algorithmes pour simuler la distribution *a posteriori* des paramètres à dimension variable, on retrouve également le DDMCMC ("Data Driven MCMC") [139], le DRMCMC ("Delayed Rejection MCMC") [67], le BDMCMC ("Birth-and-Death MCMC") [131], le CTMCMC ("Continuous Time MCMC") [20], ainsi qu'une généralisation [5] de l'algorithme de Swendsen-Wang [133]. Ces algorithmes sont des variantes assez sophistiquées de l'algorithme de Metropolis-Hastings.

3.4 Méthode pour calculer le MAP

Dans [3], les auteurs proposent une variante du recuit simulé afin de calculer le MAP de modèles markoviens cachés. En voici une description dans sa forme la plus simple.

Soit $q(\theta^{[t+1]} \mid \theta^{[t]}, y)$ un noyau de transition irréductible.

1. **Initialisation:** Initialisation quelconque de $\theta^{[0]}$.

2. **Proposition:** Engendrer θ' selon le noyau de transition $q(\theta' \mid \theta^{[t]}, y)$.

3. **Disposition:** Poser $\theta^{[t+1]} = \theta'$ avec probabilité

$$\alpha = \min\Big\{1, \frac{P(\theta' \mid y)}{P(\theta^{[t]} \mid y)}\Big\}^{1/T} \tag{3.6}$$

(*acceptation* de la proposition). Sinon, poser $\theta^{[t+1]} = \theta^{[t]}$ (*rejet* de la proposition). Augmenter t de 1 et retour en 2.

Pour assurer la convergence asymptotique vers le mode de $P(\theta \mid y)$, il suffit que la température T soit de la forme $\tau/\log(t+2)$, où $\tau > 0$ est la température initiale. Toutefois, contrairement à l'algorithme E/S, il n'y a pas de majorante explicite pratique pour τ, à notre connaissance. Toutefois, lorsque le noyau $q(\theta' \mid \theta, y)$ ne s'annule pas, il suffit de supposer que $\lim_{t\to\infty} T(t) = 0$. Il s'agit là d'un cas exceptionnel démontré à l'annexe A. Finalement, mentionnons que dans [3], les auteurs considèrent α de la forme $\min\left\{1, \frac{P(\theta' \mid y)}{P(\theta^{[t]} \mid y)}\right\}^{(1/T)-1}$. Cette formulation est équivalente à celle que nous avons adoptée. En effet, en posant $(1/T'(t)) - 1 = 1/T(t)$, on obtient $\lim_{t\to\infty} T(t)/T'(t) = 1$, car $\lim_{t\to\infty} T(t) = 0$.

3.5 Modèles markoviens

Soit G l'ensemble $\mathbb{Z}^2$ muni de l'addition (vectorielle). Nous considérons en chaque site $s \in G$, un vecteur aléatoire $I_s \in E$, où E est un ensemble d'états, fini ou non. La configuration des vecteurs aléatoires $I = (I_s)$ prend ses valeurs dans l'espace $\Omega = \{I = (I_s) \,:\, I_s \in E \text{ pour tout } s \in G\}$.

Définition 11

Un champs aléatoire est une mesure de probabilité sur Ω. Un champs aléatoire est dit stationnaire si il est invariant sous translation, c'est-à-dire $P(t \cdot I) = P(I)$, où $(t \cdot I)_s = I_{s-t}$ définit l'opérateur de translation dans G.

Si A est un sous-ensemble de G, I_A dénotera la restriction de I à A, c'est-à-dire $(I_s)_{s\in A}$. La restriction de I au complément de A est notée $I^A = I_{A^C}$.

Nous nous intéressons à un type particulier de champs aléatoire, appelé champs de Gibbs [29, 90].

Définition 12

Une famille de potentiels est une famille de fonctions $U = \{U_A\}_{A\subset G, |A|<\infty}$, telle que

1. $U_A \,:\, \Omega_A \to \mathbb{R}$;

2. $U_{A+t} \circ t = U_A$;

3. $\sum_{A\,:\,0\in A} \sup_I |U_A(I_A)| < \infty$.

Les deux premières conditions permettent de définir un champ aléatoire stationnaire à partir d'une famille de potentiels, comme dans la définition suivante. La troisième condition assure techniquement que ce champ aléatoire est strictement positif (car $e^{-\infty} = 0$).

Définition 13

Un champs de Gibbs avec famille de potentiels U est une mesure de probabilité P sur Ω telle que pour tout $A \subset G$ fini et $I \in \Omega$:

$$P(I_A \mid I^A) = \frac{1}{Z(I^A)} \exp\{-\sum_{W \subset G, W \cap A \neq \emptyset} U_W(I_W)\}, \tag{3.7}$$

où $Z(I^A)$ est une constante de normalisation.

Soit maintenant le graphe ayant G comme ensemble de sommets et dont les voisinages sont définis par la relation $t \in N(s)$ si et seulement si il existe un sous-ensemble fini W de G contenant s et t, pour lequel U_W n'est pas identiquement nulle.

Définition 14

Un champs de Markov sur le graphe G est une mesure de probabilité positive satisfaisant

$$P(I_A \mid I^A) = P(I_A \mid I_{\partial A}) \tag{3.8}$$

pour tout sous-ensemble fini A de G, où $\partial A = \{t :$ il existe $s \in A$ tel que $t \in N(s)\}$.

Il s'avère qu'un champs de Gibbs est nécessairement un champs de Markov sur le graphe G. Si A est un sous-ensemble fini de G, nous obtenons alors l'expression

$$P(I_A \mid I^A) = \frac{1}{Z(I^A)} \exp\{-\sum_{\mathcal{C}\,:\,\mathcal{C}\cap A \neq \emptyset} U_{\mathcal{C}}(I_{\mathcal{C}})\}, \tag{3.9}$$

où $\mathcal{C}$ parcourt l'ensemble des cliques du graphe G, et $Z(I^A)$ est une constante de normalisation appelée fonction de partition. Rappelons qu'une clique $\mathcal{C}$ d'un graphe G est un sous-ensemble fini de G, tel que deux éléments quelconques de $\mathcal{C}$ sont voisins dans G. Le théorème de Hammersley-Clifford (voir [12]) stipule que tout champs de Markov est une mesure de Gibbs. Nous appellerons la fonction $-\log P(I)$ énergie de Gibbs du modèle.

Une clique binaire de la forme $< (k,l),(k+1,l) >$ ou $< (k,l),(k,l+1) >$ est dite d'ordre 1. Une clique binaire de la forme $< (k,l),(k+1,l+1) >$ ou $< (k,l),(k-1,l+1) >$ est dite d'ordre 2. Tout champs aléatoire stationnaire peut être approché par un champs de Gibbs avec potentiels à support borné, c'est-à-dire telles que $U_A = 0$ si la cardinalité de A est supérieure à une une borne b arbitrairement grande [90]. Dans une autre direction, tout champs aléatoire stationnaire peut être approché par un champs de Gibbs caché avec potentiels d'ordre 1, mais avec un nombre d'étiquettes cachées arbitrairement grand [90].

3.6 Exemple de modèle markovien pour une image

Pour fixer les idées, nous considérons un modèle markovien caché sur un graphe G, muni d'un champs aléatoire observable Y et d'un champs caché discret X (qui décrit le processus des régions), dont la distribution *a priori* est un modèle markovien.

Plus précisément, étant donné une image, nous notons G le graphe dont les noeuds sont les pixels de l'image, avec pour voisinage les 8 voisins habituels d'un pixel. Nous considérons le champs observable $Y = \{Y_s\}$ des composantes couleurs basées aux pixels s de G, ainsi que le champs d'étiquettes caché discret $X = \{X_s\}$. Dans un problème de segmentation typique, nous cherchons à estimer X étant donné une réalisation de Y. La variable X_s prend ses valeurs dans un ensemble fini d'étiquettes $\Lambda = \{e_1, e_2, ..., e_K\}$, chaque étiquette représentant une classe d'équivalence de pixels ayant une couleur semblable.

Pour chacune des classes e_k, $1 \leq k \leq K$, la vraisemblance de Y_s est modélisée par une distribution de la forme $P(y_s \mid x_s = e_k, \Phi_{(k)})$ qui est paramétrisée par un vecteur de paramètres $\Phi_{(k)}$. Nous posons $\Phi = (\Phi_{(1)}, ..., \Phi_{(K)})$, le vecteur (complet) des paramètres. Par exemple, dans le cas d'un modèle gaussien $P(y_s \,|\, x_s = e_k, \Phi_{(k)}) = \mathcal{N}(y_s \mid \mu_{(k)}, \Sigma_{(k)})$, nous posons $\Phi_{(k)} = (\mu_{(k)}, \Sigma_{(k)})$.

La vraisemblance $P(y \mid x, \Phi)$ considérée est de la forme

$$\prod_s P(y_s \mid x_s, \Phi). \tag{3.10}$$

Nous modélisons la distribution *a priori* du processus de régions X par un modèle de Potts isotrope avec voisinages d'ordre 2. Cet *a priori* dépend d'un hyper-paramètre $\beta > 0$. Ainsi, $P(x \mid \beta)$ est définit par

$$\frac{1}{Z(\beta)} \exp\Big\{-\beta \sum_{<s,t>} \big(1 - \delta(x_s, x_t)\big)\Big\}, \tag{3.11}$$

où la somme est effectuée sur les paires de sites voisins, $\delta(\cdot)$ est le symbole de Kronecker, et $Z(\beta)$ est une constante de normalisation (appelée fonction de partition) égale à

$$\sum_x \exp\Big\{-\beta \sum_{<s,t>} \big(1 - \delta(x_s, x_t)\big)\Big\}. \tag{3.12}$$

3.7 Modèles markoviens hiérarchiques cachés

Nous tirons cette section de notre article [49].

Depuis les travaux inauguraux de [14, 63], plusieurs modèles markoviens hiérarchiques cachés ont été développés en traitement d'images. Dans ce qui suit, nous en passons quelques uns en revue. Les données observées sont tantôt la luminance de l'image, tantôt les réponses à des filtres (linéaires ou non) (c.f. section 5.2.2). Voir la figure 3.1 pour une illustration du graphe sous-jacent. La relation entre les étiquettes

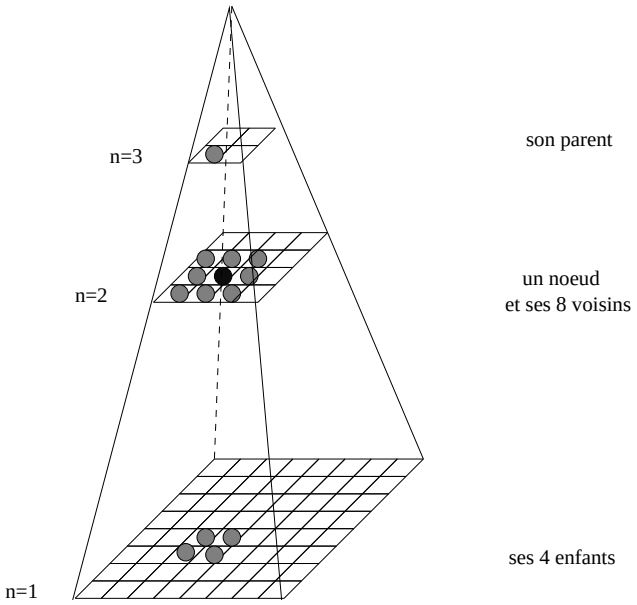


Figure 3.1. Graphe pyramidale d'un modèle markovien hiérarchique caché. Les noeuds en gris sont voisins du noeud en noir.

cachées d'un site et de son parent est appelée lien causal en échelle (il y a en fait une distinction entre un lien causal en échelle et une clique hiérarchique, mais nous omettrons ce détail dans le présent document). La relation entre les étiquettes cachées d'un site et d'un de ses voisins situés au même niveau est appelée relation spatiale.

Les modèles multi-résolution (MR) (par exemple [7]) ne considèrent qu'un seul modèle markovien mais à divers niveaux de résolution, tandis que dans le cas d'une approche multi-modèle (MM) [122], le modèle markovien varie d'un niveau de résolution à un autre.

On peut également considérer qu'un seul niveau de résolution, mais analysé à différentes échelles, comme dans le modèle multi-grille (MG) de [118] (où les liens causals en échelle ne sont pas modélisés). Dans [15], un modèle hiérarchique multi-échelle (HME) est introduit afin de tenir en compte les liens causals en échelle. Dans les modèles multi-échelle de [86] et [109], les auteurs y ajoutent des relations spatiales.

On peut également considérer à chaque niveau du modèle des données de nature différente, comme dans le modèle HME de [93]. Dans le modèle HME [26], les relations spatiales ainsi que les liens causals en échelle s'expriment dans un cadre contextuel, et les données varient d'un niveau à l'autre selon le niveau de résolution de coefficients d'ondelettes. Dans [54], ce modèle est élaboré en un modèle multi-contexte et multi-échelle (MCME).

3.8 Méthodes de segmentation et d'estimation en traitement d'images

Plusieurs méthodes ont été développées pour segmenter une image selon un modèle comme celui présenté à la section 3.6. Il s'agit d'estimer la réalisation x du champs caché, en supposant connus les paramètres Φ. Le recuit simulé (RS) [72] calcule asymptotiquement [63] une segmentation optimale au sens du MAP. L'algorithme ICM ("Iterative Conditional Mode") [14], produit souvent une bonne solution sous-optimale à partir d'une stratégie vorace. Le critère MPM ("Modes of Posterior Marginals") a été proposé comme alternative au critère du MAP, avec l'avantage d'être facilement calculable par un algorithme de type Monte Carlo (MC) [103]. Dans le contexte de modèles multi-échelle, le critère SMAP ("Sequential Maximum A Posteriori") a été introduit afin de tenir en compte les relations inter-échelle dans des modèles hiérarchiques [15]; un algorithme récursif en calcule une solution approchée [15]. Une variante multi-température du RS a été appliquée à des modèles multi-échelle [86]. D'autres méthodes sont adaptées à des modèles multi-résolution [7] ou multi-grille [118].

Lorsque la segmentation x de l'image est connue, l'estimation des paramètres Φ de vraisemblance est grandement simplifiée, puisqu'il s'agit alors d'un problème classique en statistique (les données observables sont indépendantes conditionnellement à x). Le problème est plus ardu dans le cas non supervisé, c'est-à-dire, sans connaissance de la segmentation. Il s'agit alors d'estimer conjointement x ainsi que Φ, ou

encore seulement Φ. Le recuit simulé adaptatif (RSA) [94] calcule conjointement une estimation des paramètres de vraisemblance et la segmentation de l'image au sens du MAP. Cependant, la solution (asymptotique) peut être sous-optimale [94]. Une solution (sous-optimale) des paramètres du modèle et une segmentation de l'image au sens du MV peuvent se calculer conjointement en utilisant une généralisation [24] de l'algorithme EM ("Expectation Maximization") [41]. À notre connaissance, le critère MPM n'est pas appliqué dans le cas d'une estimation et segmentation conjointe.

Une autre approche consiste à estimer d'abord les paramètres, puis de segmenter ensuite l'image. À cet effet, la procédure ECI ("Estimation Conditionnelle Itérative") s'est avérée utile pour divers modèles de segmentation d'images [17, 19, 39, 64, 117, 120, 125].

Dans chacune des méthodes mentionnées ci-dessus, le nombre de classes de régions est supposé connu. Récemment, l'algorithme RJMCMC [66] a été proposé pour l'estimation et la segmentation conjointe de modèles markoviens [6, 123, 139], dans le cas où le nombre de régions est inconnu. Dans [85], une décroissance en température est imposée afin de calculer une solution optimale au sens du MAP. Dans [96], une stratégie de type "split-and-merge" est incorporée dans un algorithme génétique, mais sans qu'il y ait nécessairement convergence.

Mentionnons également la méthode de segmentation [100], qui consiste à minimiser une fonction d'énergie à l'aide de l'algorithme [128]. En bref, le critère utilisé est l'homogénéité de la texture à l'intérieur d'une région et l'hétérogénéité de la texture entre régions.

3.9 Contribution de cette thèse sur le calcul du MAP

Dans le premier article (section 4.2.1), nous considérons un modèle simple comme celui de la section 3.6, mais avec un nombre inconnu de classes de régions. De plus, nous considérons des distributions de vraisemblance définies à partir de la distribution beta,

plutôt que la distribution gaussienne. Nous nous proposons de calculer conjointement une segmentation et une estimation des paramètres dans le sens du MAP.

Pour ce faire, nous proposons une variante originale de l'algorithme E/S de la section 2.3. La nouveauté consiste à utiliser comme noyau d'exploration, un noyau approché de l'échantillonneur de Gibbs. Nous évitons ainsi la difficulté majeure de concevoir un algorithme de type RJMCMC pour lequel le taux d'acceptation soit suffisamment élevé. La méthode proposée porte l'acronyme ESE pour Exploration/Sélection/Estimation.

Chapitre 4

A STOCHASTIC METHOD FOR BAYESIAN ESTIMATION OF HIDDEN MARKOV RANDOM FIELD MODELS WITH APPLICATION TO A COLOR MODEL

Cet article [51] a été publié comme l'indique la référence bibliographique.

François Destrempes, Max Mignotte, et Jean-François Angers, "A Stochastic Method for Bayesian Estimation of Hidden Markov Random Field Models with Application to a Color Model", *IEEE Trans. on Image Processing*, vol. 14, no. 8, pages 1096-1124, Août 2005.

Abstract

We propose a new stochastic algorithm for computing useful Bayesian estimators of Hidden Markov Random Field models, that we call Exploration/Selection/Estimation procedure. The algorithm is based on an optimization algorithm of O. François, called the Exploration/Selection algorithm. The novelty consists in using the A Posteriori distribution of the HMRF, as exploration distribution in the E/S algorithm. The ESE procedure computes the estimation of the likelihood parameters and the optimal number of region classes according to global constraints, as well as the segmentation of the image. In our formulation, the total number of region classes is fixed, but classes are allowed or disallowed dynamically. This framework replaces the mechanism of split-and-merge of regions, that can be used in the context of image segmentation. The procedure is applied to the estimation of a HMRF color model for images, whose likelihood is based on multivariate distributions, with each component following a

Beta distribution. Meanwhile, a method for computing the Maximum Likelihood estimators of Beta distributions is presented. Experimental results performed on 100 natural images are reported. We also include a proof of convergence of the E/S algorithm in the case of non-symmetric exploration graphs.

4.1 Introduction

Estimation of an image model is an important problem in Image Processing, with applications to higher-level tasks (such as object recognition, or 3D-reconstruction), and is closely related to image segmentation [25]. Since the pioneer work of [14, 63], Hidden Markov random field (HMRF) models have shown to be useful, if not fundamental, in understanding that problem. HMRF models are sufficiently simple to be algorithmically amenable, although that simplicity might be considered as an over-restrictive hypothesis. However, it is known [90] that (first-order) "HMRF models are dense among essentially all finite-state discrete-time stationary processes and finite-state lattice-based stationary random fields", so that they actually offer a nearly universal structure. The Bayesian paradigm has been widely used in the context of estimation of HMRF models and its richness deserves further study.

Various methods have been developed for segmenting an image based on HMRF models. The Simulated Annealing (SA) algorithm [72] computes asymptotically [63] the optimal segmentation in the sense of the Maximum A Posteriori (MAP) criterion. However, the temperature-cooling schedule depends on the function to minimize (i.e., the image treated). The Iterative Conditional Mode (ICM) algorithm [14], based on a greedy strategy, usually produces a good sub-optimal solution. The Modes of Posterior Marginals (MPM) criterion has been proposed as an alternative to the MAP criterion, with the advantage of being easily computed by a Monte Carlo (MC) algorithm [103]. In the context of hierarchical multi-scale (HMS) models, the Sequential Maximum A Posteriori (SMAP) criterion has been introduced in order to take into

account the inter-scale relations [15]; a recursive algorithm computes an approximate solution [15]. A multi-temperature variant of the SA has been extended to the case of HMS models [86]. Other segmentation methods are based on multi-resolution (MR) [7] or multi-grid (MG) [118] models.

One fundamental aspect of HMRF models, is the unsupervised estimation of the model parameters (i.e., without knowing the segmentation) [25]. The Adaptive Simulated Annealing (ASA) algorithm [94] computes a joint estimation of the likelihood parameters of the HMRF model and segmentation of the image, in the sense of the MAP. However, the solution might be sub-optimal [94]. A (sub-optimal) estimation of the model parameters and segmentation of the image, in the sense of the Maximum Likelihood (ML), can be computed jointly using a generalization [24] of the Expectation Maximization (EM) algorithm [41]. Another approach consists in estimating the HMRF model parameters, and then perform the segmentation of the image. Under that point of view, the Iterated Conditional Estimation (ICE) procedure has shown to be relevant in estimating a wide variety of HMRF models [17, 19, 39, 64, 117, 120, 125], although the statistical estimator that it computes is not fully understood as of now.

In all the methods mentioned above, the number of region classes is assumed to be known. More recently, the Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm [66] has been used to perform a joint estimation and segmentation of the HMRF model [6, 123, 139], in the case where the number of region classes is unknown. In [85], a cooling-temperature schedule is imposed on the RJMCMC stochastic process, in order to compute an optimal solution in the sense of the MAP. In [96], the split-and-merge strategy that is exploited in the RJMCMC, is incorporated into a hybrid genetic algorithm.

In this paper, we consider a useful family of Bayesian estimators for HMRF models, that take into account global constraints in the loss function. We propose a method for computing these estimators, that we call Exploration/Selection/Estimation (ESE) procedure. This procedure is an instance of the Exploration/Selection

(ES) algorithm of O. François [59], with the novelty that the A Posteriori distribution of the HMRF model is used as exploration distribution. The E/S algorithm is an evolutionary optimization algorithm that belongs to the family of the Generalized Simulated Annealing (GSA) algorithm [76, 137, 138]. Other GSA algorithms include the Simulated Annealing (SA) itself [72], a parallel version of the SA [136], and the genetic algorithm of R. Cerf [21, 22]. The internal parameters of the E/S algorithm depend (for all practical purposes) on the diameter of an exploration graph, and *not* on the fitness function itself. This appears to be a major advantage over other GSA algorithms[1]. It follows from O. François' Theorem [59], that the ESE procedure converges to an optimal solution independently of the initial solution. The ESE procedure computes not only the estimation of the HMRF likelihood parameters and the segmentation of the image, but also the optimal number of region classes, based on global constraints. In our framework, these tasks can be performed jointly, or in two steps (estimation of the likelihood parameters, followed by a segmentation and an estimation of the number of region classes). We view the total number of classes as fixed, but with the possibility of dynamically allowing or disallowing classes; in contrast, one would usually consider a total number of classes that varies [6, 85, 96, 123, 139]. Our formulation allows the ESE procedure to find the optimal number of (allowed) classes without resorting (explicitly) to the more sophisticated split-and-merge operators.

To keep this paper to its simplest form, we do not consider hierarchical HMRF models. Rather, we apply the ESE procedure to a new statistical HMRF (mono-scale) model for colors, whose likelihood is modeled on multivariate distributions, with each component following a Beta distribution. Incidentally, we note that the log-likelihood function of a Beta distribution is strictly concave, which justifies the

[1] More precisely, the *critical height* H_1 of the E/S algorithm depends on the fitness function, but the exploration diameter D is a good upper bound. For other GSA algorithms, the known upper bounds are impractical.

use of the Fletcher-Reeves algorithm in the computation of its ML estimators. This observation can be useful in SAR imagery [39, 40, 99], where Beta distributions are commonly used[2]. Other HMRF color models include: a probabilistic model [36] for various color features, which is segmented in the sense of the MAP by Hopfield neural network optimization; a heuristic probabilistic multi-resolution model [97] for dissimilarities of color features (based on thresholds), which is segmented in the sense of the MAP by a multi-resolution SA; a Gaussian model [116] for spatial interactions of RGB color features, which is estimated in the sense of the ML, and then segmented by a split-and-merge strategy; a Gaussian model [88] for the Luv features, which is estimated in the sense of the ML, and then segmented in the sense of the MAP by the SA; a Gaussian model [85] for the Luv features, with variable number of classes, which is jointly estimated and segmented in the sense of the MAP by a RJMCMC with temperature-cooling schedule.

The remaining part of this paper is organized as follows. In Section 4.2, we present the HRMF models considered in this paper, and the Bayesian estimators that we study. Also, the E/S algorithm is described in details, as well as its application to Bayesian estimation (i.e., the ESE procedure). Section 4.2 ends with a description of the two-step estimation and segmentation variant of the ESE procedure. In Section 4.3, we apply those concepts to the proposed HMRF color model, with a discussion on the computation of the ML estimators. Experimental results are briefly presented in Section 4.4.

[2] Gamma distributions are equally used in SAR imagery. However, our color features are *bounded*, and hence, we chose a family of distributions with bounded support. In particular, the simpler Gaussian distribution hypothesis would not be sound in our context.

4.2 Bayesian estimation of constrained HMRF models

4.2.1 Constrained HMRF models considered in this paper

Given an image, G will denote the graph whose nodes s are the pixels of the image with neighborhoods given by the usual 8-neighbors. We consider a couple of random fields (X, Y), where $Y = \{Y_s\}$ represents a random field of (continuous) observations located at the sites s of G, and $X = \{X_s\}$ represents the labeling field (i.e., a hidden discrete random field). Typically in a standard segmentation formulation, we seek an optimal realization (in the sense of some statistical criterion) of X given an observed realization of Y. For the color model presented in Section 4.3, X_s represents a class of regions in the image with "similar color" and takes its values in a finite set of labels $\Lambda = \{e_1, e_2, ..., e_K\}$, whereas Y_s is the YIQ color channels based at the pixel s.

In our context, K represents the maximal number of region classes allowed in the image. In our opinion, it is reasonable to set this upper bound according to the image size; indeed, an exceedingly large number of region classes will result in a poor estimation of the model parameters (to be discussed below), due to too few elements in the sample sets. The problem of estimating the exact number of classes will be handled below.

We consider as usual a likelihood $P(y \mid x)$ defined by a site-wise product

$$\prod_s P(y_s \mid x_s); \tag{4.1}$$

i.e., the components of Y are mutually independent given X, and furthermore $P(y_s|x)$ $= P(y_s \mid x_s)$. In a typical application, the local distributions $P(y_s \mid x_s = e_k)$ belong to a specified family of distributions parametrized by a vector $\Phi_{(k)}$ (for instance, a multivariate Gaussian model). The likelihood of the HMRF is then described completely by the parameter vector $\Phi = (\Phi_{(k)})$, $1 \leq k \leq K$. The dependence of the likelihood distribution on the particular values of the parameters is made explicit

by using the notations $P(y_s \mid x_s = e_k, \Phi_{(k)})$ and $P(y \mid x, \Phi)$. We assume that the distributions $P(y_s \mid x_s = e_k, \Phi_{(k)})$ are strictly positive and continuous functions of y_s and $\Phi_{(k)}$.

Now, it might be desirable to have actually less classes than the maximal number allowed. We view this option (for reasons that will be clear later) as omitting certain classes, rather than decreasing the actual number of classes. Thus, we introduce a vector v of K bits, that indicates which classes are allowed, with the obvious constraint that at least one of them is allowed (i.e., $\sum_{k=1}^{K} v_k \geq 1$). In particular, the vector of parameters Φ has a fixed size ($\dim(\Phi) = \sum_{k=1}^{K} \dim(\Phi_{(k)}) \geq K$) in our framework.

We model the prior distribution by a two-dimensional isotropic Potts model with a second-order neighborhood in order to favor homogeneous regions with no privileged orientation; more complex models are available in the literature. We also consider a constraint imposed by the vector v of allowed classes; namely, we say that a segmentation x is *allowed* by v, if all labels e_k appearing in x (i.e. $e_k = x_s$ for some pixel s) satisfy $v_k = 1$. Setting $\chi(e_k, v) = v_k$, it follows that $\prod_s \chi(x_s, v) = 1$ if x is allowed by v, and $\prod_s \chi(x_s, v) = 0$, otherwise. Thus, $P(x)$ is modeled by

$$\frac{1}{Z(\beta, v)} \exp\Big\{-\beta \sum_{<s,t>} \big(1 - \delta(x_s, x_t)\big)\Big\} \prod_s \chi(x_s, v), \tag{4.2}$$

where summation is taken over all pairs of neighboring sites, $\delta(\cdot)$ is the Kronecker delta function, $\beta > 0$ is a parameter, and $Z(\beta, v)$ is a normalizing constant equal to

$$\sum_x \exp\Big\{-\beta \sum_{<s,t>} \big(1 - \delta(x_s, x_t)\big)\Big\} \prod_s \chi(x_s, v). \tag{4.3}$$

So, the prior model depends on the parameter vector $\Psi = (\beta, v)$, and again, the dependence of the *prior* on Ψ is made explicit by the notation $P(x \mid \Psi)$.

Altogether, the joint distribution of the couple of random fields (X, Y) is given by $P(x, y \mid \Phi, \Psi) = P(y \mid x, \Phi) P(x \mid \Psi)$. Note that the exact number L of region classes appears implicitly in Ψ; namely, $L = \sum_{k=1}^{K} v_k$. If ever $L < K$, it is understood that

the parameter vectors $\Phi_{(k)}$ corresponding to disallowed classes (i.e, $v_k = 0$) become obsolete in subsequent higher-level tasks (such as indexing images, or localizing objects). However, they turn out to be useful at the intermediate step of estimation (to be discussed below).

4.2.2 Bayesian estimators of HMRF models

As mentioned in Section 4.2.1, the joint distribution of the HMRF (X, Y) is completely specified by the vector (Φ, Ψ). We want to estimate jointly X, Φ, and Ψ, according to some statistical criterion.

We formulate the estimation of the parameters in a Bayesian framework. We view $\Theta = (X, \Phi, \Psi)$ as the parameters to be estimated. Would it be only for numerical reasons, we find convenient to assume in the sequel that the parameters Φ and Ψ belong to *bounded* domains. The prior distribution on the parameters is defined by

$$P(\Theta) = P(X \mid \Psi)\,\mathcal{U}(\Phi)\,\mathcal{U}(\Psi), \tag{4.4}$$

where $\mathcal{U}$ denotes the uniform distribution, and $P(X \mid \Psi)$ is provided by the HMRF model.

Now, image segmentation is not a well-posed problem; it depends on some criteria that favor an over-segmentation, or on the contrary, a region merging. Thus, we consider an energy function $\rho(x)$ that sets a particular global constraint on the segmentation process. In general, that function might depend on meta-parameters, based on the particular application one has in mind (for instance, a probabilistic model of the real scene). In this paper, we consider an energy function based on the "cubic law" for region sizes [95]. Namely, assuming a Poisson model for the objects of the real scene that is captured by the image under orthographic projection [95], the area A of disk-like objects has a density proportional to $1/A^2$ (because the radius r has a density proportional to $1/r^3$). We also want to restrict directly the number

of regions in the image. So, we consider the energy function

$$\rho(x) = \omega n |G|^{1/2} + \sum_{i=1}^{n} (2 \log(|R_i(x)|) + \log(1 - 1/|G|)), \tag{4.5}$$

where $|R_1(x)|, ..., |R_n(x)|$ are the sizes of the n connected regions induced by x, $|G|$ is the size of the image, and ω is a meta-parameter ($= 0$ or 1 in our tests). More precisely, $R_1(x), ..., R_n(x)$ are the connected components of the graph H, whose vertices are the pixels of the image, and whose edges consist of pairs of 8-neighbors with *same* region label. Now, it is crucial to realize that the value of the partition function $Z(\psi)$ increases at an exponential rate with respect to the number of allowed classes (for a fixed value of β). This combinatorial fact makes obsolete the comparison of the prior $P(x \mid \Psi)$ for different number of classes: allowing just one class would be optimal. So, the constraint function ρ has to counter-balance the term $\log Z(\psi)$ appearing in the Gibbs energy of the prior model[3]. We thus consider a loss function defined by

$$E(\Theta, \theta) = 1 - e^{-\rho(X,\Psi)} \delta(X, x) \delta(\Phi, \phi) \delta(\Psi, \psi), \tag{4.6}$$

where $\rho(X, \Psi) = \rho(X) - \log Z(\Psi)$, δ denotes the Dirac distribution for continuous variables, or the Kronecker symbol for discrete variables, with $\Theta = (X, \Phi, \Psi)$ and $\theta = (x, \phi, \psi)$[4].

Finally, the likelihood $P(y \mid \Theta) = P(y \mid X, \Phi)$ is provided by the HRMF model. We are then interested in the generalized Bayesian estimator defined pointwise by

[3] In statistical mechanical Physics, the quantity $\lim_{|G| \to \infty} \frac{\log Z(\psi)}{|G|}$ corresponds to the *pressure* of the image lattice under the prior distribution $P(x \mid \psi)$. See [29, 71], for instance.

[4] One could also include in the constraint ρ a term corresponding to the Bayesian Information Criterion (BIC) [127] in order to encourage simpler models (i.e., a smaller number of allowed classes).

$$\hat{\theta}(y) = \arg\min_{\theta} \int_{\Theta} E(\Theta, \theta) P(\Theta \mid y)\, d\,\Theta \tag{4.7}$$

$$= \arg\min_{\theta} \int_{\Theta} E(\Theta, \theta) P(\Theta) P(y \mid \Theta)\, d\,\Theta, \tag{4.8}$$

since $P(y) = \int_{\Theta} P(\Theta) P(y \mid \Theta)\, d\,\Theta$ does not depend on θ; henceforth

$$\hat{\theta}(y) = (x_*, \Phi_*, \Psi_*) = \arg\max_{(x,\phi,\psi)} e^{-\rho(x,\psi)} P(x \mid \psi) P(y \mid x, \phi), \tag{4.9}$$

as is readily seen. Note that one could include ρ in the *prior* of the HMRF model and obtain the MAP estimator. However, we prefer not to do so, because this would make the Markovian blanket of each pixel extend to the whole image lattice[5]. At any rate, the proposed loss function yields the *weighted mode* of the posterior distribution of θ. The squared error loss function and the absolute error loss function would give respectively the *weighted mean* and the *weighted median* of the posterior distribution of θ (see Chapter 3 of [75]).

Now, let $\hat{\Phi}(x, y)$ be the Maximum Likelihood (ML) estimator for the complete data (x, y). That is, given a realization x and the observed realization y, let $\hat{\Phi}_{(k)}(x, y)$ be the ML estimator of $\Phi_{(k)}$ on the sample set $\{y_s \ : \ x_s = e_k\}$, so that $\hat{\Phi}(x, y) = (\hat{\Phi}_{(k)}(x, y))$. Here, it is understood that $\hat{\Phi}_{(k)}(x, y)$ can have *any* value whenever the class e_k is empty in the segmentation x (i.e., $x_s \neq e_k$ for all pixels s). Following [94], we obtain

$$(x_*, \Psi_*) = \operatorname*{argmax}_{x,\psi} e^{-\rho(x,\psi)} P(x \mid \psi) P(y \mid x, \hat{\Phi}(x, y)) \tag{4.10}$$

$$\Phi_* = \hat{\Phi}(x_*, y), \tag{4.11}$$

since for given values of x and ψ, we have $e^{-\rho(x,\psi)} P(x \mid \psi) P(y \mid x, \phi) \leq e^{-\rho(x,\psi)} P(x \mid \psi)$ $P(y \mid x, \hat{\Phi}(x, y))$, upon using the independence of the variables Y_s conditional to X.

[5] But the two formulations are perfectly equivalent, and it is just a matter of taste which one is preferred.

For simplicity, the prior parameter β is fixed to 1 throughout[6], so that Ψ reduces to the vector of allowed classes v. Thus, in that case, the estimation problem is reduced to the minimization of the fitness function

$$f(x, v) = \rho(x) - \log(P(y \mid x, \hat{\Phi}(x, y))) + \beta \sum_{<s,t>} (1 - \delta(x_s, x_t)) \qquad (4.12)$$

on the set A of all realizations (x, v) for which x is allowed by v[7]. In this context, the Simulated Annealing (SA) algorithm [63] is intractable. Also, the Adaptive Simulated Annealing (ASA) algorithm [94] might converge to sub-optimal solutions. In this paper, we propose a new variant of the Exploration/Selection (E/S) algorithm [59] in order to find an *optimal* solution, that we now present.

4.2.3 The Exploration/Selection (E/S) optimization algorithm

The aim of the E/S is to minimize a fitness function f on a finite search space A. It relies on a graph structure $\mathcal{G}$ on A, called the *exploration graph*, which is assumed connected and symmetric. For each element $x \in A$, $N(x)$ denotes the neighborhood of x in the graph $\mathcal{G}$. For each $x \in A$, a *positive* distribution $a(x, \cdot)$ is defined on the neighborhood $N(x)$ of x in the graph $\mathcal{G}$. Given $m \geq 2$, an element $\mathbf{x} = (x_1, ..., x_m)$ of the Cartesian product A^m is called a *population* (of solutions). Given a population $\mathbf{x} = (x_1, ..., x_m)$, $\alpha(\mathbf{x})$ will denote the current best solution with minimal index: $\alpha(\mathbf{x}) = x_l$ such that $f(x_k) > f(x_l)$, for $1 \leq k < l$, and $f(x_k) \geq f(x_l)$, for $l < k \leq m$. The algorithm can be stated as follows.

---

[6] See, for instance, [13, 14, 24, 42, 43, 142] for estimation methods of the *prior* model. Technically, the estimation of the parameters of the *prior* distribution can be included in our framework, but we choose not to discuss this aspect in the present paper.

[7] The term $-\log Z(\psi)$ of the constraint function cancels out with the term $\log Z(\psi)$ of the prior, so that only $\rho(x)$ appears explicitly. In particular, the fitness function does not depend on v, once restricted to the case where x is allowed by v. But note that $f(x, v) = \infty$, whenever x is not allowed by v.

1. **Initialization**: Choose randomly the initial population $\mathbf{x} = (x_1, x_2, ..., x_m)$.

2. Repeat until a stopping criterion is met:

   (a) **Updating the current best**: determine $\alpha(\mathbf{x})$ from the current population $\mathbf{x}$, according to the fitness function f.

   (b) **Exploration/selection**: for each $l = 1, ..., m$, replace with probability p, x_l by $x'_l \in N(x_l)$ according to the distribution $a(x_l, x'_l)$; otherwise, replace x_l by $\alpha(\mathbf{x})$ (with probability $1 - p$). Decrease p.

In [59], at the exploration step, the element x'_l is taken in $N(x_l) \setminus \{\alpha(\mathbf{x})\}$. But this is unnecessary, as is explained in details in Appendix A[8].

The probability p is called the *probability of exploration* and depends on a parameter $T > 0$, called the *temperature*. Taking $p_T = \exp(-1/T)$, one has to decrease T to 0 sufficiently slowly and assume that the size m of the population is sufficiently large. Let A_* be the set of global minima of the fitness function f. The following result follows directly from Theorem 2 of [59] and will suffice for our purposes.

Corollary 1 (Corollary to O. François' Theorem 2 [59])
Let D be the diameter of the exploration graph. Then, for any $m > D$, and any $\tau \geq D$

$$\lim_{t\to\infty} \max_{\mathbf{x}} P(X(t) \notin A_* \mid X(0) = \mathbf{x}) = 0$$

whenever $p(t) = (t+2)^{-1/\tau}$ (i.e., $T(t) = \frac{\tau}{\log(t+2)}$), where $t \geq 0$ is the iteration.

Proof: See [44], p.43. □

Now, we will actually need a slightly modified version of the E/S algorithm. Let B be an auxiliary finite set. We assume that the exploration distribution depends on

[8] However, for the variant [58] of the E/S algorithm, one *has* to take $x'_l \neq \alpha(\mathbf{x})$.

an element ϕ of B. So, given $x \in A$ and $\phi \in B$, $a_\phi(x, \cdot)$ is a positive distribution on the neighborhood $N(x)$. The modified E/S algorithm can be stated as follows.

1. **Initialization**: Choose randomly the initial population $\mathbf{x} = (x_1, x_2, ..., x_m)$, and choose by some deterministic rule the initial vector of auxiliary elements $\boldsymbol{\phi} = (\phi_1, ..., \phi_m)$.

2. Repeat until a stopping criterion is met:

   (a) **Updating the current best**: determine $\alpha(\mathbf{x})$ from the current population $\mathbf{x}$, according to the fitness function f.

   (b) **Exploration/selection**: for each $l = 1, ..., m$, replace with probability p, x_l by $x'_l \in N(x_l)$ according to the distribution $a_{\phi_l}(x_l, \cdot)$; otherwise, replace x_l by $\alpha(\mathbf{x})$ (with probability $1 - p$).

   (c) **Updating**: Modify the auxiliary elements $\phi_1, ..., \phi_m$ according to some deterministic rule, based on the current values of $x_1, ..., x_m$. Decrease the probability of exploration p.

In Appendix A, we show that all the results of [59] also hold for this modified version of the E/S algorithm. An example of "deterministic rule" for modifying the auxiliary elements, is presented in Section 4.2.4.

4.2.4 The Exploration/Selection/Estimation (ESE) procedure

We now present a particular instance of the E/S algorithm in the context of Section 4.2.2. We let $\mathcal{G}$ be the complete graph structure on the search space A of all pairs (x, v) for which x is allowed by v. Thus, $D = 1$, and this would yield a very poor algorithm if the exploration distribution were the uniform distribution. So, one has to design carefully an exploration distribution.

Let the auxiliary set B consists of all elements $\phi = (\phi_{(k)})$ of the form $\phi_{(k)} = \hat{\Phi}_{(k)}(x, y)$ for some x (*depending* on k). A simple possibility for the exploration distribution is the A Posteriori distribution of the HMRF model itself $a_\phi((x, v), (x', v')) = P(x' \mid y, \phi, v')\mathcal{U}(v')$, which can be simulated (approximatively) using a few sweeps of the Gibbs sampler. Thus, roughly speaking, the new allowed classes are chosen randomly according to a uniform distribution, and the exploration is concentrated around the modes of the posterior distribution $P(x' \mid y, \phi, v')$. However, for algorithmic reasons, it seems to us more interesting to replace the uniform distribution by a distribution $\lambda(v' \mid v)$ that modifies only 1/2 bit on average, and to simulate x' according to the classes allowed by v' for only one sweep:

$$a_\phi((x, v), (x', v')) \approx P(x' \mid y, \phi, v')\, \lambda(v' \mid v). \tag{4.13}$$

Note that we do not mind whether x is allowed by v', as long as x' is. In our implementation, the dependence of $a_\phi((x, v), \cdot)$ on x, holds in the fact that x serves as initialization for one sweep of the Gibbs sampler. Also, the deterministic rule for *modifying* ϕ given x, consists in setting $\phi_{(k)} = \hat{\Phi}_{(k)}(x, y)$, whenever a class k appears in x (i.e., $x_s = e_k$ for some s), and keeping the current value of $\phi_{(k)}$, otherwise. Hence, ϕ is not completely determined by x; this prevents us from dropping the dependence of the distribution $a_\phi((x, v), \cdot)$ on ϕ. In other words, writing $\phi = \hat{\Phi}(x, y)$ might leave out some classes, which would be problematic since we want to simulate *any* currently allowed class. This is the whole point in using the auxiliary set B in the E/S algorithm. Note that the exploration distribution is strictly positive because of the assumption made in Section 4.2.1 on $P(x \mid \Psi)$ and $P(y \mid x, \Phi)$.

In order to speed up convergence, one can use the K-means algorithm described in [4], rather than a random initialization. Altogether, the E/S algorithm can be outlined as follows in our context. Let $m \geq 2$ and $\tau \geq 1$.

1. **Parameter initialization**: use the K-means algorithm to obtain a raw seg-

mentation $x^{[0]}$ based on the set of color features $\{y_s\}$ into K classes. Set $x_l^{[0]} = x^{[0]}$, for $1 \leq l \leq m$; set $v_l^{[0]} = \mathbf{1}$, for $1 \leq l \leq m$, where $\mathbf{1}$ denotes the vector with all bits equal to 1. The estimate $\phi^{[0]}$ is equal to the ML estimator on the complete data $(x^{[0]}, y)$. Set $\phi_l^{[0]} = \phi^{[0]}$, for $1 \leq l \leq m$.

2. Then, $(\mathbf{x}^{[t+1]}, \mathbf{v}^{[t+1]}, \boldsymbol{\phi}^{[t+1]})$ is computed recursively from $(\mathbf{x}^{[t]}, \mathbf{v}^{[t]}, \boldsymbol{\phi}^{[t]})$ until a stopping criterion is met, as follows.

   (a) **Updating the current best**: determine $\alpha(\mathbf{x}^{[t]}, \mathbf{v}^{[t]})$ from the current population $(\mathbf{x}^{[t]}, \mathbf{v}^{[t]})$, using the values of the fitness function $f(x_l^{[t]}, v_l^{[t]})$, $1 \leq l \leq m$.

   (b) For $l = 1, 2, ..., m$, explore a solution with probability $p = (t+2)^{-1/\tau}$, or else select the current best:

   Exploration: Modify each bit of $v_l^{[t]}$ with probability $\frac{1}{2K}$; if all bits become equal to 0, set one of them (randomly) equal to 1. Let $v_l^{[t+1]}$ be the resulting vector of allowed classes. For one sweep, visit the sites of the image lattice G sequentially. At each site s, draw $x_s = e$ according to the weights

$$\begin{aligned} & P(y_s \mid x_s = e, \phi_l^{[t]})\, \chi(e, v_l^{[t+1]}) \\ \times \quad & \exp\Big\{-\beta \sum_{s' \in N(s)} \big(1 - \delta(e, x_{s'})\big)\Big\}, \end{aligned} \tag{4.14}$$

   where $N(s)$ denotes the set of 8-neighbors of s. Let $x_l^{[t+1]}$ be the resulting segmentation.

   or Selection: Let $(x_l^{[t+1]}, v_l^{[t+1]}) = \alpha(\mathbf{x}^{[t]}, \mathbf{v}^{[t]})$.

   (c) **Estimation**: Set $\phi_l^{[t+1]} = \hat{\Phi}(x_l^{[t+1]}, y)$. It is understood that for each class not appearing in $x_l^{[t+1]}$, the former estimation is kept.

From O. François' Theorem, we obtain $\lim_{t\to\infty}(x_l^{[t]}, v_l^{[t]}, \phi_l^{[t]}) = (x_*, v_*, \Phi_*)$, for $1 \leq l \leq m$, with probability 1. The Bayesian estimator sought (x_*, v_*, Φ_*) might not

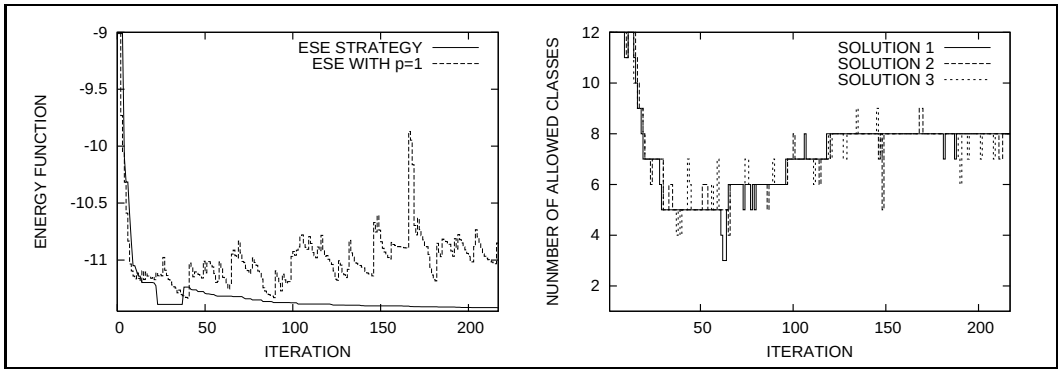


Figure 4.1. Left: example showing the current best value of the fitness function f as a function of the iteration t (the value of the function is normalized by the size of the image); the ESE strategy converges surely to the optimal solution, whereas a simulation-like strategy might take a lot longer before it reaches the optimal solution. Right: example showing the actual number of allowed classes as a function of the iteration t, for a population of 3 solutions.

be uniquely defined, but the algorithm will compute one of the optimal solutions. We call this algorithm Exploration/Selection/Estimation (ESE) procedure.

It remains to determine a sensible stopping criterion. The best result known to date in that direction is given by Theorem 3 of [59]. However, the constants R_1 and R_2 appearing there are not known explicitly. Moreover, achieving this task is way beyond the scope of this paper. So, we have decided to fix the final exploration probability empirically. In our tests, we take $m = 3$ and $\tau = 3$, and the final exploration probability is set equal to 1/6. Thus, the procedure is stopped after 217 iterations and an average of 158 explorations are performed. See Figure 4.1 for an example showing the current best value of the fitness function f as a function of the iteration t. In that figure, we compare the ESE strategy with a simulation-like strategy, upon setting $p \equiv 1$. Clearly, the ESE procedure seems more promising. Figure 4.1 also presents an example of the actual number of allowed classes that were explored, in the case of an upper bound of 12 allowed classes. We note that only 3

to 12 allowed classes were actually explored within the 217 iterations; the minimal Gibbs energy obtained was -9.79657 and the estimated number of allowed classes was 8. We also performed the estimation procedure with a fixed number of allowed classes varying from 1 to 12. The respective Gibbs energy obtained were: -4.31726, -5.6363, -9.62538, -9.80042, -9.79146, -9.75114, -9.74272, -9.72639, -9.73738, -9.74145, -9.72946, -9.707. The main point is that an exhaustive search on the number of classes yield a relative improvement of only 0.039% on the Gibbs energy (see Section 4.4 for further discussion).

As is seen from above, the exploration distribution can be easily simulated using the Gibbs sampler. If ρ were included in the exploration distribution, one would need the MCMC algorithm to simulate the exploration distribution (because, in that case, the Markovian blanket of a pixel would be too large). In that case, the Gibbs sampler could be used to simulate the proposal function. But this is unnecessary in our framework: the acceptance/rejection mechanism of the MCMC is replaced by the exploration/selection mechanism of the E/S.

One could choose the model of variable size for the vector of parameters Φ, corresponding to a variable number of region classes. But then, one would need the RJMCMC for the simulation of the exploration distribution. In contrast, our framework based on omission of classes and auxiliary set in the E/S algorithm, allows the use of the Gibbs sampler.

The ESE procedure presents some resemblance with particle filtering (PF) algorithms [35]. One can consider the iteration t as the time, the sequence of estimated parameters (θ_t) as the *signal process*, and the constant sequence $(y_t) = (y)$ as the *observation process*. The exploration distribution $a_\phi((x, v), (x', v'))$ would correspond to the transition kernel of the signal process at consecutive time, and the model likelihood to the marginal distribution of the observation conditional to the signal process. The selection step of the ESE would be replaced by the *updating step* (or *re-sampling*) of the PF. Finally, the estimation step would be replaced by a simulation

of the parameters and included in the *prediction step*, together with the exploration step. The main point is that the ESE procedure converges with a fixed number of particles (i.e., solutions) as small as 2, whereas the known convergence results [35] for the PF require that the number of particles tend to infinity.

4.2.5 Variants of the ESE procedure

In the case where the model is very complex, it might be preferable to perform the estimation and the segmentation of the model in two steps. In a first step, the estimation is performed without omitting any class, nor considering any global constraint. In a second step, the segmentation is performed according to the full model, but using the parameters of the likelihood previously estimated. We now give the details.

Estimation with no class omitted

In order to omit no class, it suffices to consider for the prior model the distribution $P(v) = \delta_{\mathbf{1}}(v)$, where δ is the Kronecker delta symbol. Moreover, the global constraint is not considered, upon setting $\rho(x) = 0$. This approach amounts to computing the Bayesian estimator

$$(x_*, \Phi_*) = \arg\max_{(x,\phi)} P(x, y \mid \phi, v = \mathbf{1}). \tag{4.15}$$

The ESE procedure is modified accordingly upon letting the search space A consists of all realizations x of the hidden random field X.

Segmentation based on the likelihood parameters

Once the vector of parameters Φ_* of the model is estimated, one can estimate once again x_* itself, but using this time the global constraint $\rho(x)$ and permitting the omission of classes. This amounts to computing the Bayesian estimator

$$(x_*, v_*) = \arg\max_{(x,v)} e^{-\rho(x,(\beta,v))} P(x, y \mid \Phi_*, v). \tag{4.16}$$

Thus, $P(\Phi) = \delta_{\Phi_*}(\Phi)$. Accordingly, one can modify the ESE procedure upon letting the auxiliary set B consists of the only element Φ_*.

Note that the resulting estimated parameters x_*, Φ_*, v_* are *not* equal to the ones computed in Section 4.2.4. Nevertheless, they also constitute reasonable and (hopefully) useful estimators of the model.

4.3 A statistical model for colors

We now apply the general concepts presented in Section 4.2 to an original statistical model for colors. We adopt the same formalism as in Section 4.2.1. Namely, G denotes the image lattice, Y is the observable random field of YIQ color channels on G, and X is the hidden random field of color labels that belong to a finite set Λ of K region classes.

4.3.1 Description of the color features

The raw data I_s represents the RGB channels at the pixel located at the site s. We compute the YIQ coordinates $y'_s = MI_s$ using the transition matrix [57]

$$M = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{pmatrix}. \tag{4.17}$$

With the convention that each component of I_s takes its values in $[0, 255] \subset (-1, 256)$, we deduce that $-1 < y'_{s,1} < 256$, $-0.596 \times 257 < y'_{s,2} < 0.596 \times 257$, and $-0.523 \times 257 < y'_{s,3} < 0.523 \times 257$. Based on these bounds, each component of y'_s is normalized

between 0 and 1. This yields the transformed data y_s[9].

4.3.2 Statistical model for the color features

For each site s of G, and each color class $e_k \in \Lambda$, we model the distribution $P(y_s | x_s = e_k)$ by a multivariate Beta model, that we now describe. First of all, we consider the diffeomorphism $\xi : (0,1)^d \to \mathbb{R}^d$ defined by $\tanh^{-1}(2x-1)$ on each component $x \in (0,1)$, where $d = 3$. A few examples convinced us that the variable $\xi(y_s)$ does not quite follow a Gaussian distribution. We chose to model y_s by considering the random vector of dimension d equal to

$$u_s = \pi(y_s) = \xi^{-1}(W_{(k)}^t(\xi(y_s) - \nu_{(k)}) + \nu_{(k)}), \tag{4.18}$$

where $\nu_{(k)}$ is the average d-dimensional vector of the transformed features $\xi(y_s)$, and $W_{(k)}$ is a $d \times d$ orthogonal (de-correlation) matrix for $\xi(y_s)$. Thus, after a suitable rotation, the components of the variable $W_{(k)}^t(\xi(y_s) - \nu_{(k)}) + \nu_{(k)}$ are assumed independent, and the same holds true for the components of u_s.

We model independently each variable $u_{s,r}$ by a Beta distribution $\mathcal{B}(u; a_r, b_r)$, where

$$\mathcal{B}(u; a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\, u^{a-1}(1-u)^{b-1}, \quad u \in (0,1), \tag{4.19}$$

with $a, b > 0$. Here, $\Gamma(x) = \int_0^1 (-\log u)^{x-1}\, du = \int_0^\infty t^{x-1} e^{-t}\, dt$ is the Euler (1729) gamma function[10]. Now, it does not seem suitable to allow an arbitrarily small value for the standard deviation of the Beta distribution, since one might end-up with arbitrarily large values for the shape parameters a, b. Indeed, we have $\sigma^2(\mathcal{B}(a,b)) = \frac{ab}{(a+b)^2(1+a+b)}$, and hence, $\lim_{a\to\infty} \sigma^2(\mathcal{B}(a,a)) = 0$. So, we impose the condition that σ

[9] One could also consider non-linear transformations of the RGB channels [74], such as the Luv coordinates.

[10] By Theorem, the Euler Beta function $B(a,b) = \int_0^1 u^{a-1}(1-u)^{b-1}\, du$ is equal to $\Gamma(a)\Gamma(b)/\Gamma(a+b)$.

be no less than a fixed value σ_-. This condition implies that a, b are bounded. Thus, our requirement that the likelihood vector of parameters Φ be defined over a bounded domain is fulfilled (see Section 4.2.2).

The values of $\sigma_{-,r}$, for $r = 1, .., d$, are established as follows. We compute $\hat{\mu} = (\hat{\mu}_1, ..., \hat{\mu}_d)$, where $\hat{\mu}_r$ is the estimated mean of $u_{s,r}$ over the sample set. We consider the derivative D of the map π evaluated at the point $y_s = \pi^{-1}(\hat{\mu})$, and we set $\sigma_{-,r} = \varepsilon \sum_{r'=1}^{d} |D_{r,r'}|$, for some fixed value $\varepsilon > 0$. With that choice, the image of the box $\prod_{r=1}^{d}[\hat{\mu}_r - \sigma_{-,r}, \hat{\mu}_r + \sigma_{-,r}]$ under the map π^{-1}, *covers* the box centered at $\pi^{-1}(\hat{\mu})$ of radius ε (with respect to the norm $||\cdot||_\infty$). Thus, roughly speaking, at least 99% of the distribution of y_s covers for each r an interval of length no less than 6ε. In our tests, we chose $\varepsilon = \frac{1}{6\times 257}$ in order to cover one unit of the RGB channels (on a scale of 0 to 255). Since, the RGB channels actually vary between 0 and 255, rather than -1 and 256 (see Section 4.3.1), the variances obtained are indeed bounded.

Altogether, $P(y_s \mid x_s = e)$ is modeled by $\prod_{r=1}^{d} \mathcal{P}_r(u_{s,r})$, where $u_s = \pi(y_s)$ and $\mathcal{P}_r(u_{s,r})$ stands for $\mathcal{B}(u_{s,r}; a_r, b_r)$. See Figure 4.2 for an example of empirical distributions for the de-correlated color features.

4.3.3 Maximum Likelihood estimators

Let $y_1, y_2, ..., y_n$ be a sample of i.i.d. observations drawn according to the multivariate Beta model $(\nu, W, \mathcal{P}_1, ..., \mathcal{P}_d)$. The first step in computing the Maximum Likelihood (ML) estimators of the model is the estimation of the de-correlation operator (ν, W). Here, we use the Principal Component Analysis (PCA) estimators

$$\hat{\nu} = \frac{1}{n}\sum_{l=1}^{n} \xi(y_l), \qquad \hat{W} = U_d, \tag{4.20}$$

where the columns of U_d span the principal subspace of the sample covariance matrix of the sample $\xi(y_l)$ (with corresponding eigenvalues in decreasing order).

Next, the pseudo-de-correlated features $u_l = \pi(y_l)$ are computed. For each fixed

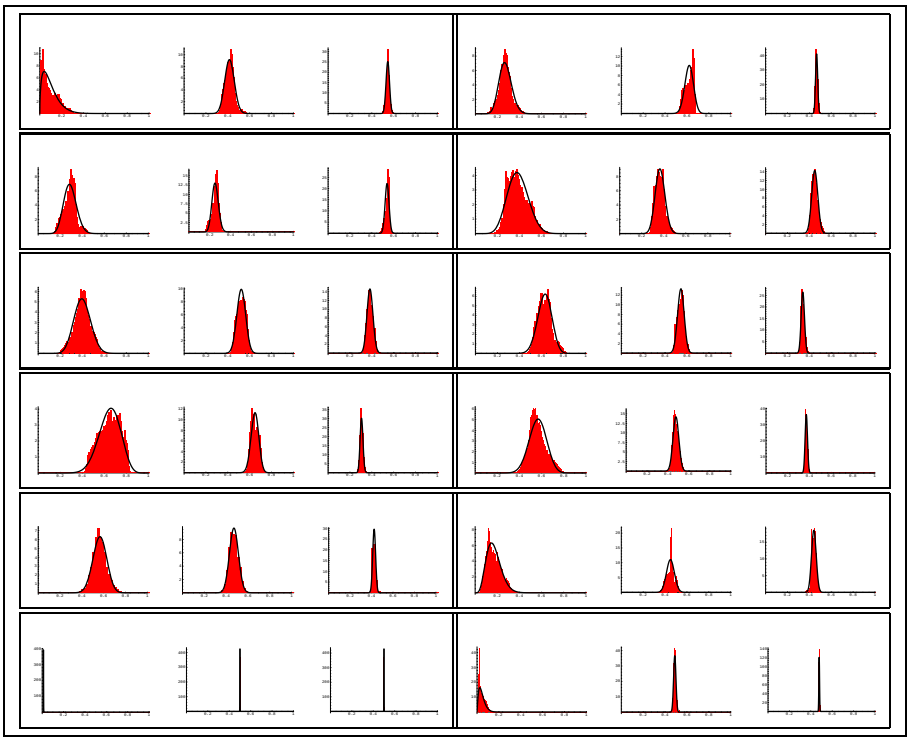


Figure 4.2. Example of empirical distributions for the de-correlated color features, based on the segmentation and the parameters estimated by the ESE procedure on the 1st image of Figure 4.4, with a maximum of 12 allowed color classes. Here, we show two classes per line. The histogram of each normalized de-correlated color feature is compared with the corresponding estimated Beta distribution.

index r, we estimate the corresponding Beta distribution, using the method explained in Appendix B.

4.3.4 Estimation and segmentation based on the color model

Given an image, the statistical model for colors is described completely by the parameter vectors

$$\Phi = (\Phi_{(k)}) = (\nu_{(k)}, W_{(k)}, \mathcal{P}_{(k),1}, ..., \mathcal{P}_{(k),d}), \quad \Psi = (\beta, v), \tag{4.21}$$

where $1 \leq k \leq K$. As in Section 4.2.2, we fix $\beta = 1$ throughout, so that Ψ reduces to v. The ESE procedure described in Section 4.2.4 is used in order to perform a joint estimation and segmentation. Alternatively, one can use the two-step variant of Section 4.2.5.

4.3.5 Simulation of the color model

Given a color class e_k, and a statistical parameter vector $\Phi_{(k)} = (\nu_{(k)}, W_{(k)}, \mathcal{P}_{(k),1}, ..., \mathcal{P}_{(k),d})$, we proceed as follows to simulate a color region of that class. For each pixel s with label k, simulate each component $u_{s,r}$ according to the given distribution $\mathcal{P}_{(k),r}$, and set $y_s = \pi^{-1}(u_s)$. Then, compute the vector y'_s corresponding to y_s before normalization and set $I_s = M^{-1}y'_s$. This process is repeated until $0 \leq I_{s,r} \leq 255$, for $r = 1, 2, 3$.

4.4 Experimental Results

We have tested the proposed method of estimation and segmentation on 100 natural images taken from the database The Big Box of Art™. We think that all of them are optical images obtained by electronic acquisition, though we do not have that information at hand. The typical size of an image was 511×768. We have performed two series of tests, with the cubic law of sizes as global constraint.

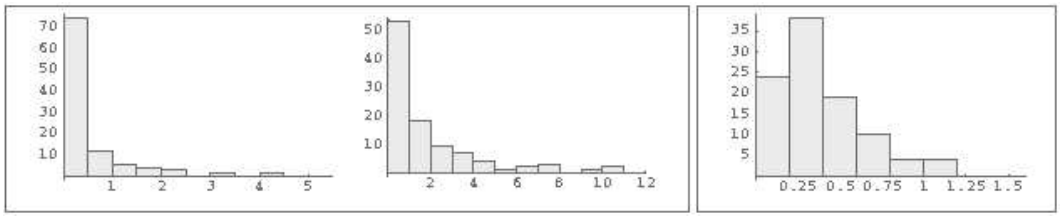


Figure 4.3. Histograms of evaluation measures over the dataset. In the usual order: Δ_1 for $K = 12$ and $\omega = 0$; mean: 0.47%. Δ_1 for $K = 12$ and $\omega = 1$; mean: 1.73%. Δ_2 for $K = 4$ and $\omega = 0$; mean: 0.50%.

In the first series of tests, we performed for each natural image I, a joint estimation and segmentation (x_*, Φ_*, v_*), with a maximal number of $K = 12$ allowed classes, and $\omega = 0$ or 1. We then simulated a synthetic image I' based on that estimation. Thus, I' and (x_*, Φ_*, v_*) were considered as ground-truth. The RGB channels of that image were saved in floats, rather than in the format ppm, in order to preserve the distributions. Next, we performed a joint estimation and segmentation (x'_*, Φ'_*, v'_*) for the synthetic image, with a maximal number of $K' = 12$ allowed classes. We evaluated the estimation error with the measure

$$\Delta_1 = \frac{|g(x_*, \Phi_*) - g(x'_*, \Phi'_*)|}{|g(x_*, \Phi_*)|} \times 100\%, \tag{4.22}$$

where $g(x, \phi) = \rho(x) - \log(P(y' \mid x, \phi)) + \beta \sum_{<s,t>} \delta(x_s \neq x_t)$, and y' is the observed random field for the synthetic image. See Figure 4.3 for a histogram of Δ_1 over the dataset, and Figure 4.4 for examples of simulated images.

The average number of allowed classes was 11.91 with $\omega = 0$, and 7.18 with $\omega = 1$. This does not necessarily mean that the algorithm failed in finding an optimal reduced number of classes. It could just mean that the optimal number of classes, according to the color model and the global constraint, is not so low. In order to clarify that important point, we performed a second series of tests, with $K = 4$, $K' = 12$, and $\omega = 0$. We compared the two segmentations with the following measure:

$$\Delta_2 = \arg\min_h \frac{1}{|G|} \sum_s \delta(x'_*(s) \neq h(x_*(s))) \times 100\%, \tag{4.23}$$

where h ranges over all one-to-one maps from Λ into Λ'. Thus, that measure represents the classification error, after an optimal match of classes. Δ_2 indicates whether the ESE procedure is capable of estimating the right number of classes, in the difficult situation where the algorithm has to reach 4 classes, starting with 12 of them. The average number of classes was 5.57, but note that Δ_2 takes into account the proportion occupied by extra classes in the image and had an average value of 0.5%. See Figure 4.3 for a histogram of Δ_2 over the dataset.

In the case of synthetic images produced with $K = 4$, we estimated each image with a *fixed* number of 4 classes. We then compared the optimal Gibbs energy with the one obtained when $K' = 12$. The relative error was only 0.20% on average. Thus, one would not gain much by performing an exhaustive search on the number of classes. The point is that, as in [62], all that matters for higher-level tasks, is the Gibbs energy of the model.

Note that specifying the value of ω (i.e., the global constraint) does not amount to fixing the number of allowed classes. Indeed, once the synthetic images are obtained upon setting $K = 12$ or $K = 3$, one obtains an average of 11.91 classes, and 4.89 classes, respectively, with a fixed value of $\omega = 0$. The point is that once the global constraint is fixed, the number of classes found by the proposed model depends on the constraint *and* the observed data. That being said, modifying the global constraint (e.g., taking $\omega = 1$ instead of $\omega = 0$) does affect the number of allowed classes. As in [139], the choice of a global constraint could be guided by a generic model of the image acquisition (e.g., [95]), a statistical criterion (e.g., [127]), or a learning phase performed on a database of images. It would remain to test the robustness of the proposed method with respect to a calibration or estimation of the global constraint parameters.

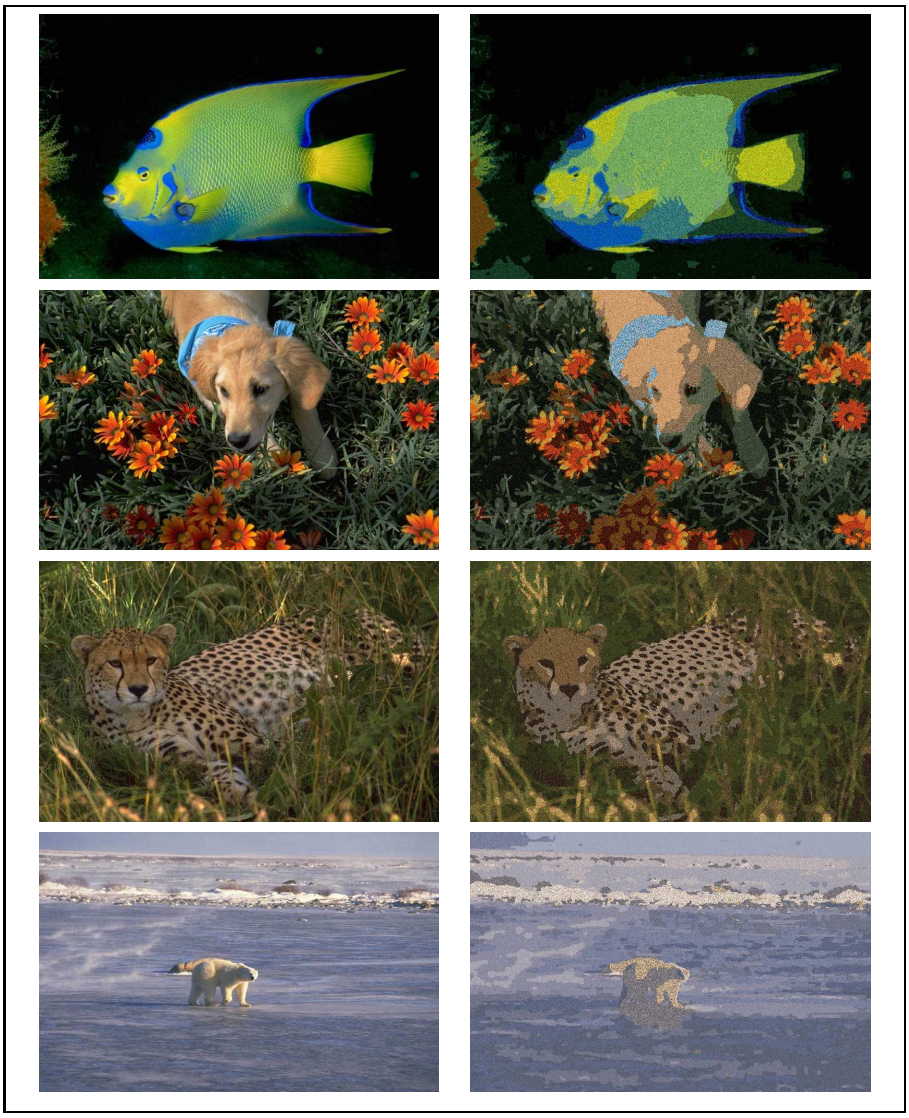


Figure 4.4. Unsupervised estimation and segmentation of images using the ESE procedure, according to the multivariate Beta color model. Left: image; right: simulation based on the resulting Bayesian estimators, using the cubic law of region sizes, with $\omega = 0$ and $K = 12$.

The ESE procedure is stopped when the exploration probability reaches 1/6 (i.e., after 217 iterations) and takes about 48 min. on a Workstation 1.2GHz when $K = 12$. This represents an average of no more than 18.5 sec. per exploration, for a total of 158 explorations. In [139], it is reported that an image of size 350×250 takes from 10 to 30 min. *after* some pre-processing step, on a Pentium-III PC. In our case, an image of size 366×250 takes about 10 min. The CPU time is not available for [85], but we know that 200 explorations were performed. Thus, we are inclined to think that the computational complexity of the ESE procedure is equivalent to that of the RJMCMC [85, 139]. Also, the ASA [94] would take the same time per iteration (and might yield a sub-optimal solution). The main point is that the known internal parameters [11] of the ESE procedure that ensure convergence are *practical*, whereas for other state-of-the-art algorithms (ASA, RJMCMC with stochastic relaxation), the known bounds are *impractical* (e.g., what should be the initial temperature that would *ensure* convergence?). Furthermore, as mentioned in [94], a joint estimation and segmentation with a plain SA is out of the question, since this would require at each iteration one estimation of the model parameters per pixel for each color label. In our case, this represents about 44 days and 6 hours of CPU time per exploration step (i.e., an increment by a factor of about 2.06×10^5). Thus, it seems to us that the computational load of the ESE procedure compares favorably to state-of-the-art algorithms for joint segmentation and estimation, with a clear advantage of having practical optimal internal parameters.

4.5 Conclusion

The ESE procedure is a general method for estimating the weighted modes of HMRF models, with global constraints taken into account. The optimal internal parameters of the algorithm (i.e., that insure asymptotic convergence to an optimal solution)

[11] i.e., $m \geq 2$ and $\tau \geq 1$ (added footnote).

are known explicitly and are practical. The split-and-merge mechanism is handled implicitly by the procedure, thus yielding a relatively easy implementation. The tests reported in this paper indicate that the ESE procedure succeeds in finding the optimal solution of the proposed color model, within a relative error bound of less than 1.73% on average.

As for the color model itself, it remains to be tested in various higher-level tasks, such as indexing or localization of shapes, in combination with models for additional aspects of image analysis. For instance, it is agreed that image segmentation should also include texture analysis and edge-detection. See [139] and [128] for instance. But in this paper, we wanted to test the estimation method on a simple model. Future work will include an extension of the ESE procedure to a hierarchical HMRF model [49], in view of texture segmentation.

4.6 Appendix A

The E/S algorithm simulates a in-homogeneous Markov chain on the set A^m, since the temperature depends on the iteration $t \geq 0$. X_t^T will denote the state of the vector $\mathbf{x}$ at iteration t, where $T = T(t)$. We let q_T be the Markov transition matrix associated with the chain (X_t^T); i.e., $q_T(\mathbf{x}, \mathbf{x}') = P(X_{t+1}^T = \mathbf{x}' \mid X_t^T = \mathbf{x})$. We then have

$$q_T(\mathbf{x}, \mathbf{x}') = \prod_{l=1}^{m} (p_T a_{\phi_l(T)}(x_l, x_l') + (1 - p_T)\delta(\alpha(\mathbf{x}), x_l'))$$

where δ denotes the Kronecker symbol. Let $I(\mathbf{x}, \mathbf{x}') = \{l \ : \ 1 \leq l \leq m, \ x_l' \neq \alpha(\mathbf{x})\}$. We obtain

$$\prod_{l \in I(\mathbf{x}, \mathbf{x}')} a_{\phi_l(T)}(x_l, x_l')\, p_T^{|I(\mathbf{x}, \mathbf{x}')|} (1 - p_T)^m \leq q_T(\mathbf{x}, \mathbf{x}') \leq \prod_{l \in I(\mathbf{x}, \mathbf{x}')} a_{\phi_l(T)}(x_l, x_l')\, p_T^{|I(\mathbf{x}, \mathbf{x}')|}.$$

Let $C > 1$ be a constant such that $\frac{1}{C}\mathcal{U}_{N(x)}(x') \leq a_\phi(x,x') \leq C\mathcal{U}_{N(x)}(x')$, for any $x \in A$, $\phi \in B$, and $x' \in N(x)$, where $\mathcal{U}_{N(x)}$ denotes the uniform distribution on $N(x)$. Such a constant exists because all sets involved are *finite*, and the distributions are *positive* on $N(x)$. Then, we have

$$\begin{aligned}\frac{1}{\kappa}\,\pi(\mathbf{x},\mathbf{x}')\exp(-V_1(\mathbf{x},\mathbf{x}')/T) &\leq q_T(\mathbf{x},\mathbf{x}') \\ &\leq \kappa\,\pi(\mathbf{x},\mathbf{x}')\exp(-V_1(\mathbf{x},\mathbf{x}')/T)\end{aligned}$$

where

$$\begin{aligned}\pi(\mathbf{x},\mathbf{x}') &= \prod_{l\in I(\mathbf{x},\mathbf{x}')} \mathcal{U}_{N(x_l)}(x'_l) \\ V_1(\mathbf{x},\mathbf{x}') &= \begin{cases} |I(\mathbf{x},\mathbf{x}'))| & \text{if } \pi(\mathbf{x},\mathbf{x}')>0 \\ \infty & \text{otherwise}\end{cases} \\ \kappa &= (1-p_{T(0)})^{-m}C^m.\end{aligned}$$

In particular, π is irreducible (since $\mathcal{G}$ is connected), and the function V_1 is exactly as in [59]. Hence, we are exactly in the same relevant setting as [59], and all the results there apply directly.

We now turn to the case where the exploration graph $\mathcal{G}$ is not necessarily symmetric. We recall from [60] that an $\mathbf{x}$-graph on A^m consists in a set of arrows $\mathbf{y} \to \mathbf{z}$ ($\mathbf{y},\mathbf{z} \in A^m$, $\mathbf{y} \neq \mathbf{x},\mathbf{z}$) such that every point of $A^m \setminus \{\mathbf{x}\}$ is the initial point of exactly one arrow, and leads to $\mathbf{x}$ through a sequence of arrows. If $\mathbf{x} \in A^m$, the set of all $\mathbf{x}$-graphs is denoted by $G(\mathbf{x})$. Also, the communication cost from $\mathbf{x}$ to $\mathbf{y}$ is defined by

$$V(\mathbf{x},\mathbf{y}) = \min\{\sum_{k=0}^{r-1} V_1(\mathbf{x}_k,\mathbf{x}_{k+1}) \;:\; \mathbf{x}_0 = \mathbf{x}, \mathbf{x}_r = \mathbf{y}, r \geq 1\}.$$

The virtual energy of $\mathbf{x}$ is then defined by

$$W(\mathbf{x}) = \min_{g \in G(\mathbf{x})} V(g)$$

where

$$V(g) = \sum_{\mathbf{y} \to \mathbf{z} \in g} V(\mathbf{y}, \mathbf{z}).$$

The set of minima of W on A^m is denoted by $\mathcal{W}_*$ and the minimal value by W_*. Let U denote the set $\{\mathbf{x} \in A^m \ : \ x_1 = x_2 = ... = x_m\}$. We identify A_* with its natural embedding into U.

The asymptotic behavior of the algorithm is determined by the critical height H_1. We refer the reader to [137] for a detailed definition of this concept, as well as the notion of cycles and exit height of a cycle. If π is a cycle, $H_e(\pi)$ denotes its exit height. H_1 is then defined as $\max_{\pi \cap \mathcal{W}_* = \emptyset} H_e(\pi)$. The importance of the critical height is expressed by the following theorem valid for any GSA.

Theorem 1 (Trouvé [137])

(a) For all decreasing cooling schedules $(T(t))_{t \geq 0}$ converging to 0 we have

$$\lim_{t \to \infty} \sup_x Prob(X(t) \notin \mathcal{W}_* \mid X(0) = \mathbf{x}) = 0$$

if and only if $\sum_{t=0}^{\infty} \exp(-H_1/T(t)) = \infty$.

The Corollary of Section 4.2.3 in the case of not necessarily symmetric graphs follows from Trouvé's Theorem, upon proving the following two propositions. See [59] for similar results and proofs in the symmetric case.

Proposition 1

If $m > D$, then $\mathcal{W}_* \subset A_*$.

Proposition 2

If $m > D$, then $H_1 \leq D$.

Lemma 1

Let $\mathbf{x} \in A^m$. *Then, there exists* $g \in G(\mathbf{x})$ *such that* $V(g) = W(\mathbf{x})$ *and for all* $\mathbf{y} \to \mathbf{z} \in g$ *either 1)* $\mathbf{y} \in A^m \setminus U$, $\mathbf{z} \in U$, $V(\mathbf{y}, \mathbf{z}) = 0$ *or 2)* $\mathbf{y} \in U$, $\mathbf{z} \in U \cup \{\mathbf{x}\}$.

Proof: This is a special case of [21] Lemma 5.9 with $H = U$, since $V(\mathbf{x}, \alpha(\mathbf{x})) = 0$ for any $\mathbf{x} \in A^m$. □

Lemma 2

Let $a_* \in A_*$ *and* $\mathbf{x} \in A^m$. *Then,* $V(\mathbf{x}, a_*) \leq D$.

Proof: This follows from [58] Lemma 6.1 (with identical proof in the non-symmetric case), since $V(\mathbf{x}, \alpha(\mathbf{x})) = 0$. □

Lemma 3

Let $\mathbf{x} \in A^m \setminus U$. *Then,* $W(\mathbf{x}) > W(\alpha(\mathbf{x}))$.

Proof: Let g be an $\mathbf{x}$-graph as in Lemma 1. There exists $\mathbf{y} \neq \alpha(\mathbf{x}) \in U \cup \{\mathbf{x}\}$ such that $\alpha(\mathbf{x}) \to \mathbf{y} \in g$. Remove that edge and introduce the edge $\mathbf{x} \to \alpha(\mathbf{x})$. This gives a $\alpha(\mathbf{x})$-graph so that

$$\begin{aligned} W(\alpha(\mathbf{x})) &\leq V(g) - V(\alpha(\mathbf{x}), \mathbf{y}) + V(\mathbf{x}, \alpha(\mathbf{x})) \\ &= W(\mathbf{x}) - V(\alpha(\mathbf{x}), \mathbf{y}). \end{aligned}$$

But $V(\alpha(\mathbf{x}), \mathbf{y}) > 0$, since $\mathbf{y} \neq \alpha(\mathbf{x})$. □

Proof: (*Proposition 1*) First consider any $\mathbf{x}$ with $\alpha(\mathbf{x}) \notin A_*$. Let g be an $\mathbf{x}$-graph as in Lemma 1. There exists $a_* \in A_*$ and $\mathbf{y} \in (U \setminus A_*) \cup \{\mathbf{x}\}$ such that $a_* \to \mathbf{y} \in g$. Remove that edge and introduce the edge $\mathbf{x} \to a_*$. This gives a a_*-graph.

Now, $V(a_*, \mathbf{y}) \geq m$, since $\alpha(\mathbf{y}) \notin A_*$. Moreover, from Lemma 2, $V(\mathbf{x}, a_*) < m$. Hence,

$$W(a_*) \leq V(g) - V(a_*, \mathbf{y}) + V(\mathbf{x}, a_*) < W(\mathbf{x}).$$

Now use Lemma 3. □

Lemma 4

For any cycle $\pi \neq A^m$, $\mathbf{x} \in \pi$, $\mathbf{y} \notin \pi$, we have

$$H_e(\pi) + W(\pi) \leq W(\mathbf{x}) + V(\mathbf{x}, \mathbf{y})$$

where $W(\pi) = \min_{\mathbf{z} \in \pi} W(\mathbf{z})$.

Proof: This is established within the proof of [137] Proposition 2.16. □

Proof: (*Proposition 2*) Let $\pi \cap \mathcal{W}_* = \emptyset$ and pick $a_* \in \mathcal{W}_* \subset A_*$ as well as $\mathbf{x} \in \pi$ such that $W(\pi) = W(\mathbf{x})$. By Lemma 4, $H_e(\pi) \leq V(\mathbf{x}, a_*)$. Now use Lemma 2. □

4.7 Appendix B

Lemma 5

Consider the Euler Beta function

$$B(a, b) = \int_0^1 u^{a-1}(1-u)^{b-1}\, du.$$

Then, $\log B$ is strictly convex on its domain $D = (0, \infty) \times (0, \infty)$; i.e.,

$$\log B(t(a_1, b_1) + (1-t)(a_2, b_2)) <$$
$$t \log B(a_1, b_1) + (1-t) \log B(a_2, b_2),$$

for any $(a_1, b_1) \neq (a_2, b_2)$ in D, and $t \in (0, 1)$.

Proof: Fixing $(a_1, b_1) \neq (a_2, b_2)$, we have to show that

$$\begin{aligned} & \log \int_0^1 u^{ta_1 + (1-t)a_2 - 1}(1-u)^{tb_1 + (1-t)b_2 - 1}\, du \\ < \;& t \log \int_0^1 u^{a_1 - 1}(1-u)^{b_1 - 1}\, du \\ + \;& (1-t) \log \int_0^1 u^{a_2 - 1}(1-u)^{b_2 - 1}\, du, \quad t \in (0, 1). \end{aligned}$$

This inequality amounts to

$$\begin{aligned} & \int_0^1 \Big(u^{a_1-1}(1-u)^{b_1-1}\Big)^t \Big(u^{a_2-1}(1-u)^{b_2-1}\Big)^{1-t} du \\ < \quad & \Big(\int_0^1 u^{a_1-1}(1-u)^{b_1-1}du\Big)^t \Big(\int_0^1 u^{a_2-1}(1-u)^{b_2-1}du\Big)^{1-t}, \end{aligned}$$

$t \in (0,1)$. Now, write $g(u) = \sqrt{u^{a_1-1}(1-u)^{b_1-1}}$ and $h(u) = \sqrt{u^{a_2-1}(1-u)^{b_2-1}}$. Then, in the case where $t = 1/2$, the inequality reads as

$$\int_0^1 g(u)h(u)\,du < \Big(\int_0^1 g(u)^2\,du\Big)^{1/2}\Big(\int_0^1 h(u)^2\,du\Big)^{1/2}.$$

But this is just Cauchy-Schwartz inequality, since $h(u)$ is of the form $cg(u)$ only if $(a_1, b_1) = (a_2, b_2)$. Thus, $\log B$ is strictly convex in the Jensen sens (J-convex) on D:

$$\log B(\frac{a_1+a_2}{2}, \frac{b_1+b_2}{2}) < \frac{1}{2}\Big(\log B(a_1,b_1) + \log B(a_2,b_2)\Big),$$

for $(a_1, b_1) \neq (a_2, b_2)$. Since $\log B$ is continuous, we conclude that it is strictly convex. □

Let $u_1, u_2, ..., u_n$ be a sample of i.i.d. observations drawn according to a Beta distribution

$$\mathcal{B}(u; a, b) = \frac{1}{B(a,b)}\, u^{a-1}(1-u)^{b-1}, \quad u \in (0,1).$$

From [65], the log-likelihood function of the distribution $\mathcal{B}(a, b)$ is given by

$$\begin{aligned} \mathcal{L}(a,b) &= \log\Gamma(a+b) - \log\Gamma(a) - \log\Gamma(b) \\ &+ (a-1)\lambda_1 + (b-1)\lambda_2, \end{aligned}$$

where $\lambda_1 = \frac{1}{n}\sum_{l=1}^n \log(u_l)$, $\lambda_2 = \frac{1}{n}\sum_{l=1}^n \log(1-u_l)$.

Corollary 2

The log-likelihood function of the Beta distribution is strictly concave and has a unique global maximum, which is its unique critical point.

Proof: Since the function $(a-1)\lambda_1+(b-1)\lambda_2$ is affine, we conclude from the lemma, that $\mathcal{L}(a,b)$ is strictly concave. Furthermore, setting $\mathcal{L}_l(a,b) = -\log(B(a,b)) + (a-1)\log(u_l)+(b-1)\log(1-u_l)$, we have that $\lim_{a\to\infty}\lim_{b\to\infty}\mathcal{L}_l(a,b) = \lim_{a\to 0}\mathcal{L}_l(a,b) = \lim_{b\to 0}\mathcal{L}_l(a,b) = -\infty$. Thus, $\mathcal{L}$ has a global maximum on its domain. Furthermore, using strict concavity, this is the unique critical point of $\mathcal{L}$ on its domain. □

Following [65], we obtain

$$\frac{\partial\mathcal{L}}{\partial a} = \psi(a+b) - \psi(a) + \lambda_1, \quad \frac{\partial\mathcal{L}}{\partial b} = \psi(a+b) - \psi(b) + \lambda_2,$$

where $\psi(x)$ is the digamma function $\frac{\Gamma'(x)}{\Gamma(x)}$. For an initial approximation of the ML estimators, let

$$\hat{\mu} = \frac{1}{n}\sum_{l=1}^{n} u_l,\ \hat{\eta} = \frac{1}{n}\sum_{l=1}^{n} u_l(1-u_l),\ \hat{\sigma}^2 = \frac{1}{n}\sum_{l=1}^{n}(u_l-\hat{\mu})^2,$$

and set $a_0 = \hat{\mu}\,\frac{\hat{\eta}}{\hat{\sigma}^2}$, $b_0 = (1-\hat{\mu})\,\frac{\hat{\eta}}{\hat{\sigma}^2}$. If ever $\hat{\sigma} < \sigma_-$, replace the former by the latter. In [65], it is recommended to use Newton-Raphson's method in order to refine the solution. But, by the Corollary, it is more appropriate to use a method such as Fletcher-Reeves algorithm for the optimization of the log-likelihood function $\mathcal{L}$. Using *strict concavity*, this algorithm will converge to the optimal solution, even if the initial solution is somewhat far from the optimal one. This gives us the estimated Beta distribution $\mathcal{B}(\hat{a},\hat{b})$. In our implementation in C++, we use the GNU scientific library of functions for the log-gamma and digamma functions, as well as for the Fletcher-Reeves method (with a tolerance of 10^{-4} as stopping criterion). If ever $\frac{ab}{(a+b)^2(1+a+b)} < \sigma_-^2$, the procedure is stopped. We admit that this is rather *ad hoc*, but in this manner we avoid working directly with the constraint.

Acknowledgment

The authors are grateful to the associate-editor and all of the anonymous reviewers for their comments and questions that helped them improve both the technical content and the presentation quality of this paper. In particular, they acknowledge the contribution of the reviewer who pointed out the omission of the partition function in the loss function in the first draft of this paper, and who mentioned the relevance of other loss functions. They also thank the reviewer who made various suggestions to extend the experimental results section, and who asked to present the link of the proposed method with particle filtering algorithms. Finally, they thank the associate-editor for mentioning the reference [90].

Chapitre 5

MODÈLES DE TEXTURES

Dans ce chapitre, nous présentons les notions sur la modélisation de texture nécessaires à la compréhension du deuxième article.

5.1 Introduction

L'information brute d'une image optique en couleurs consiste en les composantes RVB ("rouge-vert-bleu") situés en chaque pixel de l'image. Cette information se retrouve dans un fichier dont le format dépend du mode de codage ou de compression des données (formats "ppm", "gif", "jpeg", etc.). Chaque composante est habituellement encodé avec 8 bits, ce qui représente 256 valeurs. Si le facteur de compression est omit, une image de taille $N \times M$ comporte donc en principe 2^{24NM} possibilités. Ainsi, il y a 2^{240000} images possible de taille 100×100 en format "ppm", ce qui constitue un nombre formidable.

Nous notons toutefois que relativement peu de ces fichiers correspondent à des images réelles d'un type particulier (imagerie optique, SONAR, SAR, SPECT, radiographique, MRI, etc.). Nous cherchons donc, implicitement ou explicitement, à concentrer l'analyse d'images sur un sous-ensemble spécifique de l'ensemble des images potentielles, en se donnant une distribution pertinente sur cet ensemble.

Nous relevons trois points de vue en modélisation statistique d'images ([95, 139]).

1. Le point de vue descriptif : la distribution de la luminance, ou encore, la distribution jointe des réponses du signal de la luminance à divers filtres, est modélisée directement.

2. Le point de vue générique : la genèse de l'image, c'est-à-dire les différentes étapes de sa formation, est modélisée statistiquement.

3. Le point de vue structurel : le modèle de l'image s'organise autour d'une structure hiérarchique de processus stochastiques correspondants à divers niveaux d'analyse.

Nous présentons dans les trois prochaines sections chacun de ces points de vue. Dans tout ce qui suit, nous ne nous intéresserons qu'aux images optiques de scènes naturelles obtenues par acquisition électronique.

5.2 Point de vue descriptif

5.2.1 Modèles markoviens de la luminance

Soit $G = \mathbb{Z}^2$ le réseau discret de dimension 2, vu comme support d'une image de taille infinie. Nous considèrons en chaque pixel $s \in \mathbb{Z}^2$, la luminance $I_s \in E = \{0, 1, ..., 255\}$ comme une variable aléatoire. Rappelons que la luminance (la composante Y des attributs de couleurs YIQ [57]) exprime l'intensité du signal. Nous considérons un champ de Markov $P(I)$ sur l'espace $\Omega = \{I = (I_s) \ : \ I_s \in E \text{ pour tout } s \in \mathbb{Z}^2\}$ des configurations $I = (I_s)$ de la luminance. Voir section 3.5.

Les deux résultats [90] présentés à la section 3.5 ont le mérite d'être rassurants quant à la pertinence de l'étude des modèles markoviens. Par contre, dans un cas comme dans l'autre, le nombre de paramètres peut être en principe astronomique, tant et si bien que ces deux résultats ne sont guère exploitables en pratique. D'ailleurs, le nombre de paramètres pourrait être plus grand qu'il n'en faut, lorsqu'on tient compte des limites de la perception visuelle humaine.

5.2.2 Filtres

Nous présentons maintenant la notion de filtre, essentielle à la compréhension de la définition de texture.

Définition 15

Un filtre de dimension d appliqué sur les images $I \in \Omega$ à support $\mathbb{Z}^2$, est une fonction $\tau : \mathbb{Z}^2 \times \Omega \to \mathbb{R}^d$, invariante en translation, c'est-à-dire telle que $\tau(s+t, t \cdot I) = \tau(s, I)$ pour tout $s, t \in \mathbb{Z}^2$.

La valeur $\tau(s, I)$ est appelée réponse du signal de la luminance au filtre en s et sera dénotée I_s^τ dans ce qui suit.

Définition 16

Un filtre est dit linéaire si $(I_1 + I_2)_s^\tau = (I_1)_s^\tau + (I_2)_s^\tau$.

Un filtre linéaire de dimension 1 peut toujours s'exprimer sous la forme d'un produit de convolution $f_\tau * I$, où f_τ est une fonction de $\mathbb{Z}^2$ dans $\mathbb{R}$. On a alors $I_s^\tau = (f_\tau * I)_s$. Nous identifions par abus de langage, le filtre τ avec la fonction f_τ.

Parmi les filtres utiles pour l'étude des textures, se retrouvent les filtres de Gabor [61, 129], motivés par des propriétés psycho-physiques, et les filtres basés sur les ondelettes [37, 101, 102, 140], motivés par des propriétés mathématiques.

Dans cette thèse, nous utiliserons les filtres de Gabor linéaires et les filtres gaussiens. Les filtres de Gabor modélisent les réponses de cellules simples du cortex, sensibles à la fréquence et à l'orientation.

5.2.3 Textons et textures

Dans les travaux du groupe de Malik [100, 128], le point de vue markovien est abandonné. Les auteurs cherchent plutôt à estimer la distribution jointe des réponses du signal de la luminance à une banque de filtres.

Les auteurs considèrent d'abord le champs aléatoire I de la luminance d'une image à support G. Ils considèrent une banque de filtres finie $\{\tau_1, ..., \tau_m\}$, avec $\tau_i : \mathbb{Z}^2 \to$

$\mathbb{R}^{d_i}$ pour $i = 1, ..., m$. Nous obtenons les champs aléatoires des réponses $I^1, ..., I^m$ du signal I aux divers filtres, ainsi que le champ aléatoire $Y = (Y_s = (I_s^1, ..., I_s^m))$ des attributs de textures. Dans ce cas, l'espace des attributs de textures est $\Upsilon = \mathbb{R}^{d_1} \times \cdots \mathbb{R}^{d_m}$.

Les vecteurs d'attributs y calculés sur une base d'apprentissage d'images, ou sur une seule image, sont regroupés dans un même ensemble-échantillon $\mathcal{E}$. Pour une valeur de K fixée, l'espace des attributs de textures est partitionné en K cellules de Voronoi à partir de l'échantillon $\mathcal{E}$. Pour ce faire, on cherche K vecteurs d'attributs $c_1, ..., c_K$ qui minimisent la fonction $\sum_{y \in \mathcal{E}} ||y - c_{pr(y)}||^2$, où $c_{pr(y)}$ est le vecteur d'attributs le plus près de y parmi les vecteurs $c_1, ..., c_K$. Une solution sous-optimale est obtenue à l'aide de l'algorithme des K-moyennes (la solution optimale est en général intractable). Les vecteurs $c_1, ..., c_K$ représentent donc les centres de K agrégats et sont appelés textons. Nous obtenons ainsi un vocabulaire de K éléments de textures atomiques. Une cellule de Voronoi consiste en l'ensemble des vecteurs d'attributs de Υ dont un même texton est le plus près parmi les textons du vocabulaire.

Étant donné une image, on peut associer à chaque pixel s le texton situé le plus près du vecteur d'attributs y_s (c'est-à-dire, la cellule de Voronoi du vocabulaire qui contient y_s). Nous obtenons ainsi le texton $T(s)$ situé en s.

Une fois la décomposition de l'image en textons calculée, la texture d'une région R de l'image peut être définie comme la distribution des textons dans la région; c'est-à-dire, pour chaque texton c_i du vocabulaire, $p(c_i)$ dénote sa fréquence dans la région R. Ainsi, nous obtenons rien d'autre que la distribution jointe des réponses de la luminance à une banque de filtres, telle qu'estimée par la méthode du plus près voisin. Il s'agit donc d'une méthode non paramétrique standard [52], dont le nombre de paramètres est fixé par le nombre de cellules et de filtres.

5.2.4 Ensembles de Julesz (notion mathématique de texture)

Dans son article inaugural, Julesz [83] a initié la recherche sur les textures en suscitant la question fondamentale suivante : quels attributs et statistiques constituent les éléments de base pour la perception humaine des textures? Pour répondre à cette question, il faut d'abord répondre à la question suivante [146] : étant donné un ensemble de statistiques cohérentes, comment engendrer des images texturales aléatoires admettant ces statistiques?

Dans [146], le groupe de Zhu définit une texture essentiellement comme une classe d'équivalence d'images partageant un même ensemble de statistiques. Plus précisément, ils considèrent l'ensemble Ω des images I ayant G comme graphe des pixels. Étant donné un ensemble h de m statistiques $h^1, ..., h^m$, on assigne à chaque image I le point $h(I) = (h^1(I), ..., h^m(I))$ dans l'espace des statistiques. Soit $\mathcal{H}(h_0)$ un voisinage d'une réalisation h_0 de h; $\Omega_G(\mathcal{H}(h_0))$ dénote la classe d'équivalence d'images $\{I : h(I) \in \mathcal{H}(h_0)\}$. L'ensemble de Julesz $\Omega(h_0)$ est alors définit par $\lim_{G\to\mathbb{Z}^2, \mathcal{H}(h_0)\to\{h_0\}} \Omega_G(\mathcal{H}(h_0))$. La notion d'ensemble de Julesz constitue la définition mathématique de texture. À première vue, la notion de texture de [100] en est un cas particulier (le cas $m = 1$). Toutefois, nous pouvons toujours nous ramener au cas d'une seule statistique en prenant la jointe $h(I^1, ..., I^m)$, ce qui permet de retrouver les marginales $(h^1(I), ..., h^m(I))$. Dès lors, les deux notions sont en fait équivalentes.

Un algorithme pour simuler un ensemble de Julesz quelconque est présenté dans [146]. Il s'agit en fait d'une application du recuit simulé avec une fonction de coût pertinente.

5.2.5 Modèle FRAME

Dans [147, 148], le groupe de Zhu considère comme statistiques texturales, les histogrammes $h(I^1), ..., h(I^m)$ des réponses du signal de la luminance I (définie sur un support fini G) à une banque de filtres de Gabor $\{\tau_1, ..., \tau_m\}$. Les auteurs justifient

ce choix tout d'abord par des expériences en psycho-physique [11, 27], dans le cas des textures homogènes. De plus, un résultat [148] stipule que, quelque soit le type de texture, la distribution jointe de la luminance (distribution de même dimension que la taille de l'image) est uniquement déterminée par les histogrammes des réponses de la luminance à tous les filtres linéaires (distributions de dimension 1). Bien entendu, il est hors de question de considérer tous les filtres linéaires, mais on peut espérer qu'un choix judicieux de certains filtres suffise pour différencier les textures perceptiblement.

Soient maintenant les histogrammes en question $h(I^1), ..., h(I^m)$. Les auteurs cherchent une distribution $P(I)$ de la luminance qui maximise l'entropie définie par $-\int \log\{P(I)\}P(I)dI$, sous la contrainte que les fréquences des réponses aux filtres discrétisées $\int \delta(z - I^k)P(I)dI$ coïncident avec les distributions $h(I^k)(z)$, pour $k = 1, ..., m$. C'est le principe de l'entropie maximale, bien connu en statistiques [80]. La solution à ce problème [147] s'avère être un modèle markovien de la forme $\frac{1}{Z_G(\beta)} \exp\{- < \beta, h(I) >\}$, où $h(I) = (h(I^1), ..., h(I^m))$ est discrétisé en un certain nombre de classes. Il s'agit du modèle FRAME.

Ce modèle est particulièrement ardu à estimer, à cause de la fonction de partition $Z_G(\beta)$. Toutefois, des théorèmes d'équivalence entre le modèle FRAME et les ensembles de Julesz correspondant sont démontrés dans [149]. En outre, il n'est pas nécessaire d'estimer le modèle FRAME pour le simuler.

5.3 Point de vue générique

5.3.1 Généralités

En infographie [57], il est entendu que la synthèse d'images est effectuée selon un modèle complet de la genèse de l'image, depuis la scène réelle jusqu'à l'acquisition. Plus précisément, nous pouvons distinguer les éléments suivants d'un modèle générique de l'image :

1. scène réelle (aspect géométrique) : taille, forme et disposition des objets ;
2. scène réelle (aspect physique) : textures des objets et mode d'éclairage (sources lumineuses, matériau des objets, opacité, transparence) ;
3. mode de propagation des rayons (phénomènes de réfraction, réflection) ;
4. modèle de caméra (type de projection géométrique) ;
5. mode d'acquisition (capteurs sensoriels, digitalisation) ;
6. image (encodage, compression).

Ce point de vue a été adopté en analyse d'images avec les travaux de Mumford *et al.* [95, 115] et de Grenander *et al.* [69, 70]. Même dans le cas de modèles simples (mais pertinents), certaines statistiques de l'image qui s'en déduisent, correspondent bien aux distributions observées dans le cas d'images naturelles.

5.3.2 Un exemple de modèle générique pour les images naturelles

Nous présentons maintenant un modèle d'occlusion pour les images naturelles [95] qui a le mérite d'être relativement simple, tout en permettant de déduire des résultats significatifs sur certaines statistiques de l'image.

Un objet $T \subseteq \mathbb{R}^2 \times \{0\}$ est un disque parallèle au plan de projection $z = 0$, de niveau de gris a variant entre $a_{\min}$ et $a_{\max}$, et de rayon r variant entre $r_{\min}$ et $r_{\max}$. Chaque objet est translaté par un vecteur $v = (x, y, z)$, avec $z > 0$. L'éclairage est supposé constant de valeur nulle et les objets sont opaques. La projection est orthographique et on suppose aucun bruit d'acquisition. L'image obtenue est en niveaux de gris encodés sur 8 bits.

La scène réelle est donc de la forme $\bigcup_i (T_i \oplus v_i)$, où chaque objet T_i est de niveau de gris a_i, de rayon r_i et est translaté par le vecteur $v_i = (x_i, y_i, z_i)$ ($T_i \oplus v_i$ dénote

la translation de T_i par v_i). Les niveaux de gris a_i sont considérés comme variables indépendantes identiquement distribuées selon une loi $p(a)$. De même, les rayons r_i sont supposés indépendants identiquement distribués selon $f(r)$. Les vecteurs de translation v_i sont disposés selon un modèle de Poisson [92]. L'image obtenue est de la forme $I(x,y) = a_{i(x,y)}$, où $i(x,y) = \arg\min_i \{z_i \ : \ (x - x_i, y - y_i) \in T_i\}$ (le niveau de gris est hérité de l'objet le plus près du plan de projection, car les objets sont supposés opaques). Sous ces hypothèses, on génère des images qui ne semblent pas naturelles, mais voir la remarque à la fin de la présente section.

Un processus de Poisson (Π, Ω, μ) est un sous-ensemble dénombrable aléatoire Π d'un espace mesurable Ω, qui satisfait aux propriétés suivantes :

1. pour tout sous-ensemble mesurable A de Ω, le nombre (aléatoire) $N(A)$ de points de Π contenus dans A suit une loi de Poisson de moyenne $\mu(A)$; c'est-à-dire, $P(|A \cap \Pi| = r) = e^{-\mu(A)} \frac{\mu(A)^r}{r!}$;

2. pour tout sous-ensembles mesurables disjoints A_1, A_2, ..., A_n de Ω, les variables aléatoires $N(A_1)$, $N(A_2)$, ..., $N(A_n)$ sont indépendantes.

Techniquement, il faut supposer que la diagonale $\{(x,x) : x \in \Omega\}$ est mesurable dans l'espace produit $\Omega \times \Omega$ (en outre, les singletons de Ω sont mesurables). Il s'avère que la fonction μ définit une mesure sur Ω, appelée mesure moyenne du processus de Poisson. Cette mesure est nécessairement non atomique (c'est-à-dire, $\mu(\{x\}) = 0$, pour tout $x \in \Omega$) [92], p. 13. De plus, si μ est une mesure non atomique sur un ensemble mesurable Ω, telle que μ se décompose en une somme dénombrable de mesures finies, alors il existe un processus de Poisson sur Ω ayant précisément μ comme mesure moyenne [92], p. 23. Les auteurs s'intéressent surtout à l'espace euclidien $\Omega = \mathbb{R}^d$ muni des ensembles de Borel et de la mesure de Lebesgue dv. La mesure définie par $\mu(A) = \lambda \int_A dv$ étant non atomique (où $\lambda > 0$ est appelée l'intensité), il existe un processus de Poisson Π sur $\mathbb{R}^d$ ayant μ comme mesure moyenne. Ce processus est

appelé processus de Poisson uniforme, et est le seul dont les propriétés stochastiques sont invariantes sous rotations et translations de $\mathbb{R}^d$ [92], p. 13.

Dans le cas du modèle [95], l'ensemble dénombrable aléatoire $\{v_i\}$ des vecteurs de translation forme un processus de Poisson uniforme sur l'espace $\mathbb{R}^3$ d'intensité $\lambda > 0$. Le couple aléatoire (r_i, a_i) est considéré comme une caractéristique (*marking*) de v_i à valeurs dans l'espace $M = [r_{\min}, r_{\max}] \times [a_{\min}, a_{\max}]$. Par un théorème de Kingman [92] (*Marking's Theorem*), nous obtenons un processus de Poisson (non uniforme) $\Pi^* = \{(v_i, r_i, a_i)\}$ sur l'espace $\mathbb{R}^3 \times M$ de mesure moyenne ayant comme dérivée $d\,\mu = \lambda f(r)p(a)dv\,dr\,da$. Pour les image de scènes naturelles, les auteurs supposent l'invariance du processus Π^* sous homothéties $\sigma : (x, y, r) \mapsto (\sigma x, \sigma y, \sigma r)$ (à ceci près que l'on ignore pour l'instant que le rayon r est définit sur un intervalle compact). Ils déduisent que la distribution du rayon suit une loi $f(r) \propto r^{-3}$. Plus précisément, la distribution de rayon est définie par $f(r) = \frac{2}{(r_{\min}^{-2} - r_{\max}^{-2})r^3}$. En guise d'exemple, nous pouvons déduire immédiatement que le nombre d'objets situés dans un volume V, ayant niveau de gris $a \pm \varepsilon$ et de rayon $r \pm \delta$ suit une loi de Poisson de moyenne $\lambda \frac{((r-\delta)^{-2} - (r+\delta)^{-2})}{(r_{\min}^{-2} - r_{\max}^{-2})} \int_V dv \int_{a-\varepsilon}^{a+\varepsilon} p(a)\,da$.

Un résultat fondamental [95] porte sur la probabilité $P(o_s = o_t \mid d(s,t) = d)$ que deux pixels s et t de l'image, situés à une distance d, appartiennent à un même objet de la scène (après projection). Soit la fonction définit par :

$$B(d) \;=\; \begin{cases} \frac{a_3}{3}(s^3 - u^3) + \frac{a_2}{2}(s^2 - u^2) + a_1(s - u) \\ + \log(\frac{r_{\max}}{r_{\min}}), \text{ si } 0 \le d < 2r_{\min}; \\ \frac{a_3}{3}(8 - u^3) + \frac{a_2}{2}(4 - u^2) + a_1(2 - u) \\ + \log(\frac{2}{u}), \text{ si } 2r_{\min} \le d < 2r_{\max}; \\ = 0, \qquad \text{si } d > 2r_{\max}, \end{cases} \tag{5.1}$$

où $a_3 = 0.052$, $a_2 = -0.051$, $a_1 = -0.61$, $s = \frac{d}{r_{\min}}$ et $u = \frac{d}{r_{\max}}$. Alors [95],

$$P(o_s = o_t \mid d(s,t) = d) \approx \frac{B(d)}{2\log\frac{r_{\max}}{r_{\min}} - B(d)}. \quad (5.2)$$

Une fois que la distribution $p(a)$ des niveaux de gris des objets est spécifiée, les auteurs peuvent déduire de ce théorème diverses statistiques de l'image. À titre d'exemple, nous obtenons immédiatement que la probabilité que deux pixels s et t, situés à une distance d l'un de l'autre, aient un niveau de gris $a_s = a$ et $a_t = b$, respectivement, est égale à

$$\begin{aligned} P(a_s = a, a_t = b \mid d(s,t) = d) &= (1 - P(o_s = o_t \mid d(s,t) = d))p(a)p(b) \quad (5.3) \\ &+ P(o_s = o_t \mid d(s,t) = d)p(a)\delta(a = b). \quad (5.4) \end{aligned}$$

Dans [95], les auteurs proposent une loi exponentielle tronquée pour $p(a)$ et obtiennent des statistiques semblables à celles observées dans les images naturelles, telles :

1. la différence en niveaux de gris entre deux pixels situés à une même distance ;
2. la distribution jointe des coefficients de l'ondelette de Haar ;
3. la matrice de co-occurrence des niveaux de gris d'ordre 2.

5.4 Point de vue structurel

5.4.1 Généralités

Dans [139], les auteurs rappellent le point de vue inaugural de Marr [104]: "les images naturelles consistent en des couches multiples de processus stochastiques avec organisations spatiales, tels que les processus des textures, des textons, des points (stochastiques), des droites, des courbes, des graphes, des régions, et des objets." [1]

[1] "real world images consist of multiple layers of stochastic processes with hierarchic spatial organizations, such as texture, texton, stochastic point, line, curve, graph, region, and object processes".

C'est ce que nous appelons le point de vue structurel. Les auteurs retiennent trois avantages [139] de ce point de vue :

1. il inclut comme sous-problème, la segmentation d'images en tant que calcul du processus des régions ;

2. il intègre la reconnaissance d'objets et l'organisation perceptive ;

3. il intègre implicitement la notion de modèles génériques.

5.4.2 Un exemple de modèle structurel pour les images naturelles

Nous présentons maintenant un exemple de modèle structurel [139].

Soit une image I de support G. Les auteurs s'intéressent aux décompositions de I de la forme $W = (K, \{(R_i, l_i, \Theta_i); i = 1, ..., K\})$, où la collection $(R_i)_{i=1}^{K}$ constitue une partition de G en régions non vides, l_i est un modèle générique de la région R_i (voir section 5.3), et Θ_i est le vecteur de paramètres de ce modèle.

Les auteurs proposent 4 modèles pour la luminance :

1. un modèle gaussien à faible variance pour les régions uniformes (modèle g_1) ;

2. un modèle non paramétrique (diagramme de batôns) pour les textures non-homogènes (modèle g_2) ;

3. le modèle FRAME pour les textures homogènes (modèle g_3) ;

4. un modèle de surface de Bézier pour les ombrages ("shadings") (modèle g_4).

Les auteurs proposent également 2 modèles pour les couleurs, que nous omettrons.

Les auteurs proposent une distribution jointe du couple (W, I) de la manière suivante. La vraisemblance est de la forme

$$P(I \mid W) \;=\; \prod_{i=1}^{K} P(I_{R_i} \mid l_i, \Theta_i),$$

où les distributions pour chaque modèle sont définies par

$$\begin{aligned}
&P(I_R \mid l = g_1, \Theta_1 = (\mu, \sigma)) = \prod_{s \in R} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(I_s - \mu)^2}{2\sigma^2}\};\\
&P(I_R \mid l = g_2, \Theta_2 = (\pi_0, ..., \pi_{255})) = \prod_{s \in R} \pi_{I_s};\\
&P(I_R \mid l = g_3, \Theta_3 = \beta = \text{FRAME}) = \prod_{s \in R} \frac{1}{Z_s(\beta)} \exp\{- < \beta, H(I_s \mid I_{\partial s}) >\};\\
&P(I_R \mid l = g_4, \Theta_4 = (A = \text{surface de Bézier}, \sigma)) = \prod_{s \in R} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(I_s - A(s))^2}{2\sigma^2}\}.
\end{aligned}$$

La densité *a priori* est de la forme

$$\begin{aligned}
P(W) \;&\propto\; P(K) \prod_{i=1}^{K} P(R_i) P(l_i) P(\Theta_i \mid l_i)\\
&\propto\; \exp\Big\{-\lambda_0 K - \sum_{i=1}^{K} \oint_{\partial R_i} ds + \gamma |R_i|^c + \nu |\Theta_i|\Big\},
\end{aligned}$$

où $\lambda_0, \gamma, c, \nu$ sont des paramètres du modèle. Le terme $-\lambda_0 K$ définit une loi *a priori* de Poisson sur le nombre de régions; le terme $\sum_{i=1}^{K} \oint_{\partial R_i} ds$ favorise des contours lisses; le terme $\gamma |R_i|^c$ contrôle la taille des régions; le terme $\nu |\Theta_i|$ pénalise un trop grand nombre de paramètres du modèle.

5.5 Point de vue adopté dans cette thèse

Dans une perspective structurelle, nous voulons nous donner un modèle markovien caché de l'image qui intègre à un premier niveau d'abstraction, un modèle descriptif global pour expliquer les textons et les couleurs, et à un deuxième niveau, un modèle

markovien du processus des régions. Contrairement à [139], les régions partagent donc un même modèle descriptif des textures, mais avec des paramètres estimés distincts. Tout comme dans [100] et contrairement à [139], nous choisissons de modéliser la distribution des attributs de textures plutôt que la distribution de la luminance (les deux formulations sont équivalentes). Contrairement à [100], nous nous intéressons à un modèle paramétrique des distributions, plutôt que non paramétrique. L'approche proposée réduit considérablement le nombre de paramètres en évitant les cellules de Voronoi ou le modèle FRAME. Nous nous proposons d'estimer ce type de modèle sans phase d'apprentissage supervisé, la segmentation en régions n'étant qu'une partie des paramètres estimés. Notons que nous aurions pu nous fixer comme objectif d'utiliser une phase d'apprentissage comme il est fait dans [100]. Dans le chapitre 6, nous généralisons la procédure ESE dans le cadre de ce modèle plus élaboré que celui du chapitre 4. À l'annexe C, nous comparons la méthode proposée avec un recuit simulé dans le cadre du modèle du chapitre 6.

Chapitre 6

FUSION OF HIDDEN MARKOV RANDOM FIELD MODELS AND ITS BAYESIAN ESTIMATION

Cet article [45] a été accepté pour publication comme l'indique la référence bibliographique.

François Destrempes, Jean-François Angers, et Max Mignotte, "Fusion of Hidden Markov Random Field Models and its Bayesian Estimation", *IEEE Trans. on Image Processing*, accepté, 10 février 2006.

Abstract

In this paper, we present a Hidden Markov Random Field (HMRF) data-fusion model. The proposed model is applied to the segmentation of natural images based on the fusion of colors and textons into Julesz ensembles. The corresponding Exploration/Selection/Estimation (ESE) procedure for the estimation of the parameters is presented. This method achieves the estimation of the parameters of the Gaussian kernels, the mixture proportions, the region labels, the number of regions, and the Markov hyper-parameter. Meanwhile, we present a new proof of the asymptotic convergence of the ESE procedure, based on original finite time bounds for the rate of convergence.

6.1 Introduction

Data fusion of image channels provided by various sensors is an important problem in Image Processing, with applications to image segmentation of natural images, or

in areas such as geophysical imaging, medical imaging, radio-astronomy (see [111]). In this paper, we focus on image segmentation based on data fusion.

One of the goals of image segmentation is to decompose an image into meaningful regions, such as consistent parts of objects or of the background, based on the fusion of various types of features. Our point of view is to characterize each meaningful region in the image by the distribution of the features on the region. This characterization of a region is called a Julesz ensemble [146].

When working with Julesz ensembles, it is customary to combine the various features assuming the independence property conditional to the region process. An example can be found in [141], in the case of univariate Gaussian kernels and uniform priors on the estimated Gaussian parameters. In [55], a Gaussian and an inverse gamma priors are set on the mean and the variance, respectively, of each Gaussian kernel. One can also assume a correlation between the channels conditional to the region process. For instance, the channels can be combined into a single multi-channel vector [87], and the joint likelihood is then defined directly on that vector. In [120], a correlation is also set on the channels. In this paper, we set a correlation between the features of a same type, but consider features of different types as independent conditional to the region process.

In order to perform data fusion, a stationary distribution can also be put directly on the image features knowing the marginals of the features, based on the maximum entropy principle [80, 111, 147]. The solution belongs to the generalized exponential family (c.f. the Filters, Random fields And Maximum Entropy (FRAME) model [148]). A fundamental result states the equivalence [149] between FRAME models and Julesz ensembles. In this paper, we consider directly Julesz ensembles as a mean of data fusion, due to the computational load of estimating FRAME models. In [139], possibly different generic (FRAME) models are set on the various regions. But since the likelihoods of distinct models might not be comparable, appropriate weights need to be assigned on each type of generic model [139]. In contrast, we adopt a same

model for all the regions, that is flexible enough to represent *any* type of region.

In the models mentioned above, the fusion process is based on the data itself. One can also base the fusion decision on the individual channel decisions. In [89], the fusion of the decisions is based on an *ad hoc* Markov model. In [10, 126], the various channel decisions are combined together according to Dempster-Shafer theory.

As it stands, the image itself could form a single region, or on the contrary, each region could be formed of only a few pixels. Thus, in order to obtain a meaningful decomposition of the image, one needs a prior probability on the region process that sets a constraint on the spatial organization of the region labels. In this paper, the spatial prior is defined by a Markov model of order 2, as well as a global constraint based on the size of connected regions.

Once a model is established, an equally important problem is the estimation of the model parameters. We adopt the Bayesian paradigm for the estimation of the model. We propose to estimate not only the Gaussian parameters of the kernels and the region labels, but also the mixture proportions, the number of regions, and the Markov hyper-parameter. Various Bayesian priors are set on the parameters. We choose to compute the MAP of the proposed model, weighted by a global constraint on the region process.

The algorithm presented in this paper in order to compute the MAP is an extension of the Exploration/Selection/Estimation (ESE) procedure [51] to the proposed fusion model. It is a variant of the Exploration/Selection (ES) algorithm [59], that integrates an (approximate) Markov Chain Monte Carlo (MCMC) transition kernel into the exploration scheme. Meanwhile, we present a new proof of the asymptotic convergence of the ES algorithm, based on original finite time bounds for the rate of convergence. The ES algorithm can be viewed as a mix between genetic algorithms and simulated annealing. See [112, 113] for closely related algorithms.

Among the main algorithms for simulating the posterior distribution of models of variable dimension, are the Reversible Jump Markov Chain Monte Carlo (RJM-

CMC) [66, 85, 123], the Data Driven MCMC (DDMCMC) [139], the Birth-and-Death MCMC (BDMCMC) [131], the Delayed Rejection MCMC (DRMCMC)[67], the general Continuous Time MCMC (CTMCMC) [20], and a generalization [5] of Swendsen-Wang algorithm [133]. In our case, we avoid the (major) difficulty of engineering a Metropolis-Hastings dynamics with a sufficiently high rate of acceptance, upon using the ES algorithm. The point is that in order to compute the MAP of the model, it is not required to simulate precisely the posterior distribution of its parameters [1].

In this paper, we illustrate the proposed data fusion model and the estimation method with color and texture features. The color features are the *Luv* components, and the texture features are the responses to Gabor and Gaussian filters. As mentioned in [145], textons refer to micro-structures in natural images. In [100], a texton is modeled by a Voronoi cell in the space of texture features. In this paper, we model a texton by a unimodal (Gaussian) distribution on the space of texture features. The distribution of the texture features on the region class is then a mixture of the unimodal kernels, each one appearing according to a certain proportion within the region. That distribution describes a Julesz texture.

This work develops on our previous paper [51] in the following aspects: 1) the ESE procedure is applied to a triplet of Markov random fields, rather than a pair of random fields, in the context of data fusion; 2) the local likelihoods are mixtures of distributions rather than unimodal distributions; 3) the hyper-parameter of the Markov prior on the region process is estimated rather than being fixed; 4) the proposed model is shown to be identifiable; 5) finite time bounds for the proposed algorithm are given rather than just a proof of asymptotic convergence.

The remaining part of this paper is organized as follows. In Section 6.2, we present the Hidden Markov Random Field (HMRF) color and texture models considered in this paper, as well as their fusion model. In Section 6.3, we present the Bayesian

[1] as is done in this paper (added footnote).

estimator and its algorithmic computation. Experimental results are briefly presented in Section 6.4.

6.2 Fusion of colors and textures

6.2.1 Random Fields considered

The lattice of the image pixels is viewed as a graph G, with set of nodes V. We consider a hidden discrete random field $X = (X_s)$ on G with random variable X_s taking its values in a finite set of labels $\Lambda = \{e_1, e_2, ..., e_K\}$. Our intention is to consider Λ as the set of region classes, and we call X the region process.

We consider M levels of analysis of the image, such as colors and textures. At each level of analysis $n = 1, ..., M$, an observable random field Y_n is defined on the graph G. Accordingly, for each level of analysis n, an observable (continuous) random variable $Y_{s,n}$ is defined at each site $s \in V$. The variables $Y_{s,n}$ take their values in a space of image features Υ_n of dimension d_n, depending only on the level n. Our intention is to consider each set Υ_n as a space of image features, such as color or texture features. We collect the various levels of analysis together, upon considering the random field $\hat{Y} = Y_1 \times \cdots \times Y_M$ on the graph G.

Next, for each level of analysis $n = 1, ..., M$, we consider discrete random variables $C_{s,n}$ that take their values in a finite set of K_n cue labels $\Omega_n = \{g_{1,n}, ..., g_{K_n,n}\}$. Each label represents an equivalence class of similar image features. The discrete random field $C_n = (C_{s,n})$ is called a cue process. Examples of cue processes are the color process and the texton process (Section 6.2.4). We collect the various cue processes together, upon considering the random field $\hat{C} = C_1 \times \cdots \times C_M$ on the graph G. See Fig. 6.1 for an illustration of the region process X, and the cue processes C_1 and C_2 in the case of color and texture features.

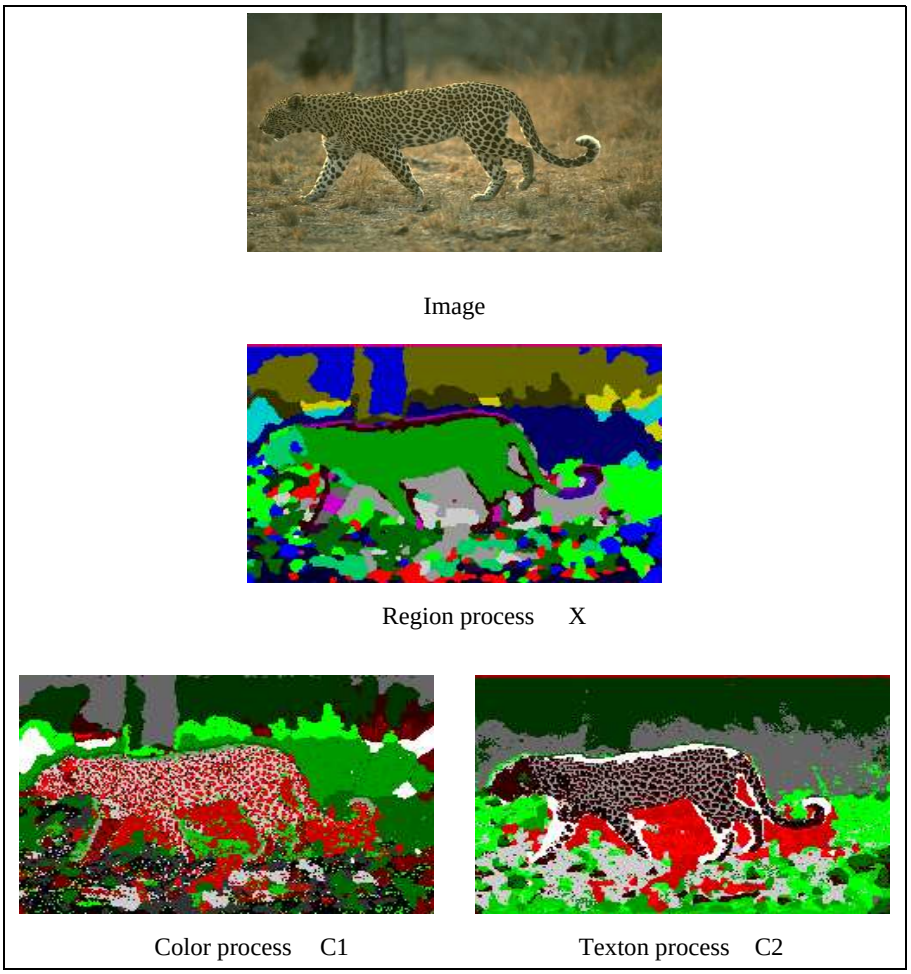


Figure 6.1. A natural image; the estimated region process X; and the simulated cue processes: the color process C^1 and the texton process C^2 (c.f. Section 6.2.1). See Section 6.2.4 for a description of the color features Y^1 and the texture features Y^2.

6.2.2 Likelihood

We now present a model for the likelihood of the observable image features $\hat{y} = (y_1, ..., y_M)$ conditional to the hidden field of region labels x, and the hidden field of cue labels $\hat{c} = (c_1, ..., c_M)$ (such as color labels, texton labels). The main point is to use a unimodal distribution for the local likelihood of the observable image features $y_{s,n}$ conditional to a cue label, and to use a mixture of these unimodal distributions for the local likelihood of the observable image features conditional to a region label.

For each site s at level n and each cue class $g_{i,n} \in \Omega_n$, the likelihood of the observable features $y_{s,n}$ conditional to the cue label $c_{s,n}$ is modeled essentially by a Gaussian kernel. More precisely, we consider the diffeomorphism $h : (0,1)^{d_n} \to \mathbb{R}^{d_n}$ defined by $\tanh^{-1}(2x-1)$ on each component $x \in (0,1)$, where d_n is the dimension of the feature space Υ_n. We define $P(y_{s,n} \mid c_{s,n} = g_{i,n}, \mu_{i,n}, \Sigma_{i,n})$ by

$$\mathcal{N}(h(y_{s,n}); \mu_{i,n}, \Sigma_{i,n})\, J(h)(y_{s,n}), \tag{6.1}$$

where $J(h)$ denotes the Jacobian of the map h.

We collect the local likelihoods together by setting

$$P(y_n \mid c_n, \mu_n, \Sigma_n) = \prod_s P(y_{s,n} \mid c_{s,n}, \mu_n, \Sigma_n), \tag{6.2}$$

where $\mu_n = (\mu_{i,n})$ and $\Sigma_n = (\Sigma_{i,n})$. We view the M levels as independent, conditional to $\hat{C}$; more precisely, the joint distribution of $\hat{Y}$ conditional to $\hat{C}$ is modeled by

$$P(\hat{y} \mid \hat{c}, \mu, \Sigma) = \prod_{n=1}^{M} P(y_n \mid c_n, \mu_n, \Sigma_n). \tag{6.3}$$

where $\mu = (\mu_n)$ and $\Sigma = (\Sigma_n)$.

Next, at each level of analysis $n = 1, ..., M$, we consider each cue label $g_{i,n}$ to appear in some proportion within the region class e_k. Namely, let $\pi_n = (\pi_{(i,k),n})$ satisfy

$$\pi_{(i,k),n} \geq 0, \quad \sum_{i=1}^{K_n} \pi_{(i,k),n} = 1, \quad 1 \leq i \leq K_n, 1 \leq k \leq K. \tag{6.4}$$

Then, we model the probability of $c_{s,n}$ conditional to x_s by

$$P(c_{s,n} = g_{i,n} \mid x_s = e_k, \pi_n) = \pi_{(i,k),n}. \tag{6.5}$$

We collect these local likelihoods together by setting

$$P(c_n \mid x, \pi_n) = \prod_s P(c_{s,n} \mid x_s, \pi_n). \tag{6.6}$$

Again, we consider the M level of analysis to be independent. So, we set

$$P(\hat{c} \mid x, \pi) = \prod_{n=1}^{M} P(c_n \mid x, \pi_n), \tag{6.7}$$

where $\pi = (\pi_n)$.

The joint distribution of $(\hat{y}, \hat{c})$ conditional to the region process x is expressed as

$$\begin{aligned} P(\hat{y}, \hat{c} \mid x, \mu, \Sigma, \pi) &= P(\hat{y} \mid \hat{c}, \mu, \Sigma) P(\hat{c} \mid x, \pi) \\ &= \prod_{n=1}^{M} P(y_n \mid c_n, \mu_n, \Sigma_n) P(c_n \mid x, \pi_n) \\ &= \prod_{n=1}^{M} \prod_s P(y_{s,n} \mid c_{s,n}, \mu_n, \Sigma_n) P(c_{s,n} \mid x_s, \pi_n), \end{aligned} \tag{6.8}$$

using equations (6.2), (6.3), (6.6), and (6.7). We deduce that the marginal of $\hat{y}$ conditional to x is equal to

$$P(\hat{y} \mid x, \mu, \Sigma, \pi) = \sum_{\hat{c}} P(\hat{y}, \hat{c} \mid x, \mu, \Sigma, \pi) \tag{6.9}$$

$$= \sum_{\hat{c}} \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid c_{s,n}, \mu_n, \Sigma_n) P(c_{s,n} \mid x_s, \pi_n)$$

$$= \prod_{n=1}^{M} \prod_{s} \{ \sum_{c_{s,n}} P(y_{s,n} \mid c_{s,n}, \mu_n, \Sigma_n) P(c_{s,n} \mid x_s, \pi_n) \}$$

$$= \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid x_s, \mu_n, \Sigma_n, \pi_n),$$

where each factor $P(y_{s,n} \mid x_s = e_k, \mu_n, \Sigma_n, \pi_n)$ is equal to

$$\sum_{i=1}^{K_n} \pi_{(i,k),n} \mathcal{N}(h(y_{s,n}) \mid \mu_{i,n}, \Sigma_{i,n}) J(h(y_{s,n})). \tag{6.10}$$

Thus, for each region label x_s, the likelihood $P(y_{s,n} \mid x_s, \mu_n, \Sigma_n, \pi_n)$ is a mixture of the K_n distributions $P(y_{s,n} \mid c_{s,n} = g_{i,n}, \mu_{i,n}, \Sigma_{i,n})$, and only the mixture proportions $\pi_{(i,k),n}$ vary from one region class to another. In particular, the Gaussian kernels $P(y_{s,n} \mid c_{s,n} = g_{i,n}, \mu_{i,n}, \Sigma_{i,n})$ are independent of the region label e_k. The proposed family of distributions is quite flexible, since any continuous distribution can be approximated by a mixture of a sufficiently large number of Gaussian kernels. See Fig. 6.2.

The marginal distributions of the features $P(y_{s,n} | x_s = e_k, \mu_n, \Sigma_n, \pi_n)$, $n = 1, ..., M$, define uniquely a Julesz ensemble [146] for each region e_k; namely, the set of stationary fields with the distributions $P(y_{s,n} \mid x_s = e_k, \mu_n, \Sigma_n, \pi_n)$, $n = 1, ..., M$, as marginals. In the case of texture features, the Julesz ensemble is referred to as a Julesz texture. Furthermore, we then call the micro-texture corresponding to a single Gaussian kernel $P(y_{s,n} \mid c_{s,n} = g_{i,n}, \mu_{i,n}, \Sigma_{i,n})$ a texton. Thus, a texture is a mixture of textons.

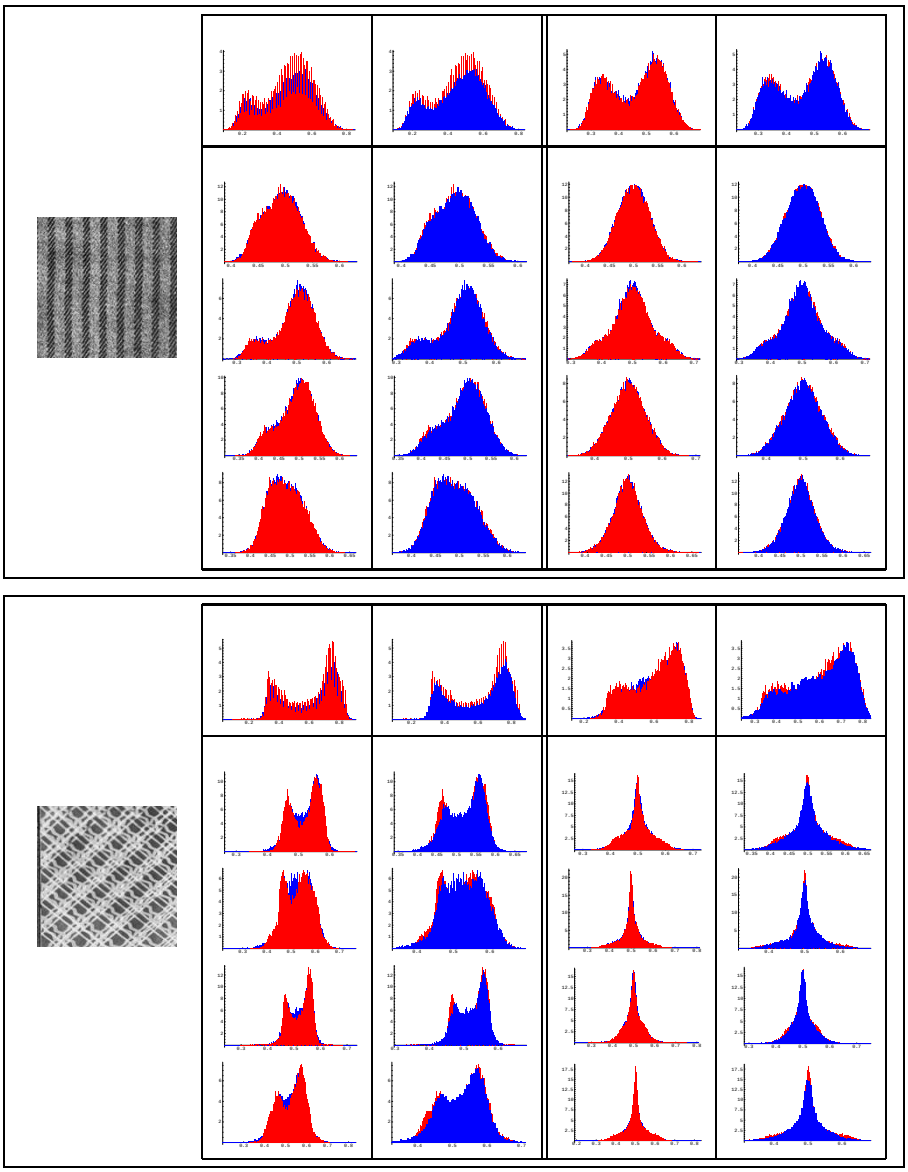


Figure 6.2. Examples of empirical and simulated distributions for the gray-level and the Gabor filter responses, based on the parameters estimated by the ESE procedure. Mixtures of Gaussian kernels are quite flexible in modeling various continuous distributions.

6.2.3 Prior on the region process

In this paper, we consider a Potts model of order 2 on G. Namely, we consider the family of potential functions

$$\begin{aligned} U_{\langle s,t\rangle}(x) &= \sqrt{2}\delta(x_s \neq x_t), \\ &\quad \text{if } \langle s,t\rangle \text{ is horizontal or vertical;} \\ U_{\langle s,t\rangle}(x) &= \delta(x_s \neq x_t), \\ &\quad \text{if } \langle s,t\rangle \text{ is diagonal;} \\ U(x) &= \sum_{\langle s,t\rangle} U_{\langle s,t\rangle}(x), \end{aligned} \tag{6.11}$$

where $\langle s,t\rangle$ ranges over the set of all binary cliques. The diagonal cliques have less weight than horizontal or vertical cliques as in [16].

Now, as in [51], K is considered as the maximal number of regions labels allowed in the region process X. In order to handle the case of possibly less classes than K, we consider a vector v of K bits, with the constraint that $|v| = \sum_{k=1}^{K} v_k \geq 1$ (c.f. [51]). The vector v indicates which regions labels are allocated (i.e., e_k is allocated if $v_k = 1$).

Let $\beta > 0$ be a hyper-parameter. The prior distribution $P(x \,|\, \beta, v)$ is then defined by

$$\frac{1}{Z(\beta, v)}\, \chi(x, v)\, e^{-\beta\, U(x)}, \tag{6.12}$$

where $\chi(x, v) = 1$ if the labels appearing in x are precisely the ones allocated by the vector v (i.e., $v_k = 1$ if and only if $x_s = e_k$ for some pixel s), and $\chi(x, v) = 0$, otherwise. Here, $Z(\beta, v)$ is a normalizing constant called the *partition function*

$$Z(\beta, v) = \sum_{x\,:\,\chi(x,v)=1} e^{-\beta\, U(x)}. \tag{6.13}$$

An important uniqueness property of this model will be discussed in Section 6.3.1.

6.2.4 Image features

We now present the color features at level $n = 1$. At each pixel of the image, the raw color RGB components yields the CIE XYZ components under the hypothesis that the NTSC phosphors standard [57] was used. The features $y_{s,1}$ are then the Luv color components [1] at the pixel s, computed from its XYZ components. In particular, $d_1 = 3$ is the dimension of the space of color features Υ_1. The purpose of the Luv components is to provide a perceptually uniform color space. Note however, that we could have used any color system since they all differ by a (possibly non-linear) change of variables.

Next, we present the texture features at level $n = 2$. A fundamental result [148] states the perfect reconstruction of the luminance density (a stationary field) from the marginals of *all linear* filter responses (one dimensional random variables). In practice, only a few filters suffice to distinguish textures within a given image. In that case, the joint distribution of the chosen filter responses defines uniquely a Julesz ensemble [146] that describes the texture.

Let $\mathcal{M}$ be a given filter bank. We consider the observable random vector $y_{s,2}$ defined by the filter responses $(f * Y)(s)$, where Y_s is the image luminance Y of the XYZ components at pixel s (and *not* the L component of the Luv coordinates). In particular, $d_2 = |\mathcal{M}|$. An important issue is the design of a filter bank [77, 147]. In this paper, we choose a filter bank as follows.

Recall that the linear 2D-Gabor filters [38] *are optimal* for a joint spatial and spectral resolution (c.f. the *uncertainty principle* of [38]). Such a filter $f(x, y; \phi, \mu_x, \mu_y, \sigma_x, \sigma_y)$ is defined by

$$\exp\{-2\pi^2((\sigma_x x')^2 + (\sigma_y y')^2) - 2\pi j(\mu_x x + \mu_y y)\}, \tag{6.14}$$

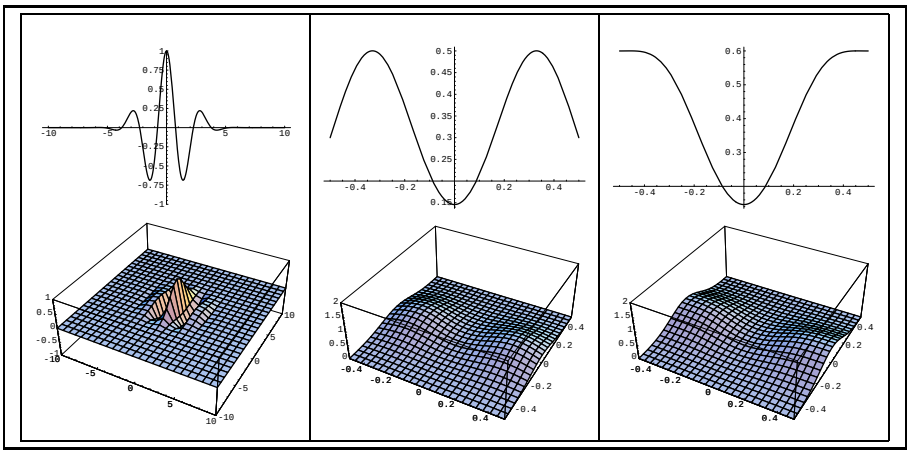


Figure 6.3. Left: real part of a Gabor filter. Center: its Fourier transform. Right: spectral aliasing due to spatial digitization. Parameters: $\mu_x = 0.33$**,** $\mu_y = 0$**,** $\phi = 0$**,** $\sigma_x = \sigma_y = 0.168166$**.**

where $x' = \cos(\phi)x + \sin(\phi)y$ and $y' = -\sin(\phi)x + \cos(\phi)y$, and $j = \sqrt{-1}$. The first term defines a Gaussian kernel with mean (μ_x, μ_y) (of dimension 2) and covariance matrix $\Sigma_{\phi,\sigma_x,\sigma_y}$

$$\begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix} \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix} \begin{pmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{pmatrix}^t . \tag{6.15}$$

The second term produces a harmonic modulation with mean frequency (μ_x, μ_y). The angle ϕ is called the orientation, and σ_x, σ_y the standard-deviations. Accordingly, its Fourier transform F is a Gaussian kernel of mean (μ_x, μ_y) and covariance matrix $\Sigma_{\phi,\sigma_x,\sigma_y}$.

Note that for a Gabor filter $f = f(\phi, \mu_x, \mu_y, \sigma_x, \sigma_y)$, the spatial and the spectral uncertainties [38] are respectively equal to

$$\Delta(f) = \frac{1}{8\pi^2}\, \sigma_x^{-1}\sigma_y^{-1}, \quad \Delta(F) = \frac{1}{2}\, \sigma_x\sigma_y. \tag{6.16}$$

We establish the architecture of the filter bank as follows. We set in what follows

$\sigma_x = \sigma_y = \sigma$, and $\mu_y = 0$. We choose a *bandwidth* of 2 octaves; i.e., $\frac{\mu_x + \sqrt{2\log(2)}\sigma}{\mu_x - \sqrt{2\log(2)}\sigma} = 4$. Thus, the mean frequency is equal to $\mu_x = \frac{5}{3}\sqrt{2\log(2)}\sigma$. We take 4 equally-spaced rotations $\phi = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$. Taking the real and imaginary parts of each Gabor filter, we obtain 8 high-pass filters (see Fig. 6.3). We also consider a low-pass Gaussian filter with *same spectral resolution* as the Gabor filters, i.e. $\frac{1}{\sqrt{2}}\sigma$, and hence with same variance. The corresponding spatial resolution of those 9 filters is $\frac{1}{\sqrt{8}\pi\sigma}$. In our tests, we chose $\mu_x = 0.33$, which yields $\sigma \approx 0.168166$, $3\sqrt{\Delta(f)} \approx 2.00765$ and $3\sqrt{\Delta(F)} \approx 0.356733$.

6.3 Estimation of the model

6.3.1 Bayesian estimator considered

The fusion model $(X, \hat{Y})$ presented in Section 6.2 is completely described by the vectors of parameters

$$\begin{aligned} &\mu = (\mu_{i,n}),\ \Sigma = (\Sigma_{i,n}),\ \pi = (\pi_{(i,k),n}); \\ &\beta, v; \end{aligned} \tag{6.17}$$

where $1 \leq k \leq K$, $1 \leq i \leq K_n$, $1 \leq n \leq M$, as presented in equations (6.1), (6.4), and (6.12). We estimate the vector of parameters $\theta = (x, \mu, \Sigma, \pi, \beta, v)$ in a Bayesian framework. Under that paradigm, it is essential to specify carefully the prior on the parameters to be estimated.

Prior on the mixture proportions

It is now a standard practice in Bayesian statistics to set a Dirichlet process prior [53] on the mixture proportions. Namely, the prior on the mixture proportions π (conditional to the allocation vector v) is defined independently at each level $n =$

$1, ..., M$ and for each allowed region class $k = 1, ..., K$ (i.e., such that $v_k = 1$) by a Dirichlet distribution $\mathcal{D}(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n}; A_0, \alpha_1, ..., \alpha_{K_n})$ equal to

$$\frac{\Gamma(A_0)}{\prod_i \Gamma(A_0\alpha_i)} \prod_{i=1}^{K_n} (\pi_{(i,k),n})^{A_0\alpha_i - 1}, \tag{6.18}$$
$$\sum_{i=1}^{K_n} \alpha_i = 1, \quad \alpha_i > 0,$$

where Γ is the Euler gamma function. The constant $A_0 > 0$ is called the dispersion parameter, and the constants $\alpha_1, ..., \alpha_{K_n}$ represent the prior information of the latent variable $C_{s,n}$.

In our case, the initial guess on the mixture proportions π_n is that the cue label $c_{s,n}$ conditional to each region label $x_s = e_k$ is distributed uniformly on the set Ω_n of cue labels. So, we set the prior proportions $\alpha_1, ..., \alpha_{K_n}$ equal to $\frac{1}{K_n}$. This is called a Dirichlet process prior with base measure the uniform distribution on Ω_n, and with dispersion parameter $A_0 > 0$. In our setting, we want a non-informative uniform distribution for the mixture proportions, so that we take $A_0 = K_n$. Thus, we obtain the uniform prior $P(\pi \mid v)$

$$\prod_{k\,:\,v_k=1} \prod_{n=1}^{M} \mathcal{D}(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n}; K_n, \frac{1}{K_n}, ..., \frac{1}{K_n}). \tag{6.19}$$

In order to sample $(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n})$ from a Dirichlet distribution $\mathcal{D}(A_0, \alpha_1, ..., \alpha_{K_n})$, we use the algorithm of Table 6.1. An interesting advantage of the Dirichlet prior on the mixture parameters $(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n})$ is that the posterior distribution of the parameters conditional to the cue process c_n and the region process x is also a Dirichlet distribution as detailed in Table 6.2. A prior with that property is called a conjugate prior.

Simulation of $(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n})$ according to a Dirichlet distribution $\mathcal{D}(A_0, \alpha_1, ..., \alpha_{K_n})$:

for $i = 1, ..., K_n$ **do**

Sample Z_i from a Gamma distribution $\mathcal{G}(A_0\alpha_i, 1)$.

end for

for $i = 1, ..., K_n$ **do**

Set $\pi_{(i,k),n} = \frac{Z_i}{\sum_{i=1}^{K_n} Z_i}$.

end for

Table 6.1. Simulation of a Dirichlet distribution.

Let n be a fixed level of analysis. Let c_n be a realization of the cue process at level n, and x be a realization of the region process. Let the prior on $(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n})$ be as in equation (6.18).

Compute $N_{(i,k)} = |\{s \ : \ c_{s,n} = g_{i,n}, x_s = e_k\}|$ and $N_k = |\{s \ : \ x_s = e_k\}|$, for $i = 1, ..., K_n$, and $k = 1, ..., K$.

Then, the posterior distribution of $(\pi_{(1,k),n}, ..., \pi_{(K_n,k),n})$ given c_n and x is the Dirichlet distribution $\mathcal{D}(N_k + A_0, \frac{N_{(i,k)}+A_0\alpha_1}{N_k+A_0}, ..., \frac{N_{(K_n,k)}+A_0\alpha_{K_n}}{N_k+A_0})$.

Table 6.2. Expression of the posterior Dirichlet distribution.

Prior on the cue likelihood parameters

For the cue Gaussian likelihood parameters $(\mu_{i,n})$ and $(\Sigma_{i,n})$ of equation (6.1), we consider independently for each level of analysis $n = 1, ..., M$ and for each cue class $i = 1, ..., K_n$, the usual conjugate prior for multivariate Gaussian distributions defined by Theorem 7.7.3 of [2]

$$\begin{aligned} \mu_{i,n} \mid \Sigma_{i,n} &\sim \mathcal{N}(\mu_0, \frac{1}{k_0}\Sigma_{i,n}); \\ \Sigma_{i,n} &\sim \mathcal{IW}(\Lambda_0, \nu_0), \end{aligned} \tag{6.20}$$

where $\mathcal{N}$ is the Normal distribution and $\mathcal{IW}$ is the *inverted Wishart* distribution with ν_0 degrees of freedom. Here, μ_0 is a vector of dimension d_n, the dimension of the feature space Υ_n, k_0 is a positive constant, and Λ_0 is a positive-definite symmetric matrix of dimension $d_n \times d_n$. The inverted Wishart distribution is defined by:

$$\mathcal{IW}(\Sigma_{i,n}; \Lambda_0, \nu_0) = \frac{|\Lambda_0|^{\frac{1}{2}\nu_0} |\Sigma_{i,n}|^{-\frac{1}{2}(\nu_0+d_n+1)} e^{-\frac{1}{2}tr\Lambda_0(\Sigma_{i,n})^{-1}}}{2^{\frac{1}{2}\nu_0 d_n} \Gamma_{d_n}(\frac{1}{2}\nu_0)}, \tag{6.21}$$

where $|A|$ denotes the determinant of a square matrix $A = (a_{jk})$, $tr(A) = \sum_{j=1}^{d_n} a_{jj}$ denotes its trace, and

$$\Gamma_{d_n}\left(\frac{1}{2}\nu_0\right) = \pi^{d_n(d_n-1)/4} \prod_{j=1}^{d_n} \Gamma\left(\frac{1}{2}\nu_0 - \frac{1}{2}(j-1)\right), \tag{6.22}$$

with Γ the Euler gamma function. The expectation of $\Sigma_{i,n}$ is $\frac{1}{\nu_0-(d_n+1)}\Lambda_0$, for $\nu_0 > d_n + 1$. See [2], Lemma 7.7.1.

In our tests, we fix $k_0 = 0.01$ and $\nu_0 = d_n + 2$, and the values of Λ_0 and μ_0 are estimated, once and for all on a given image, at each level of analysis according to the method presented in Table 6.3. Thus, we obtain the prior defined by $P(\mu, \Sigma)$ equal to

Let n be a fixed level of analysis.

Compute the empirical mean $\bar{\mu}_n$ of the set $\mathcal{E}_n = \{h(y_{s,n}) : s \in V\}$. Compute its ML covariance matrix S_n.

Set $k_0 = 0.01$, $\nu_0 = d_n + 2$, $\mu_0 = \bar{\mu}_n$ and $\Lambda_0 = \nu_0 S_n$ in equation (6.20).

Table 6.3. Computation of the prior Gaussian/Inverted Wishart parameters.

$$\prod_{n=1}^{M} \prod_{i=1}^{K_n} \mathcal{N}(\mu_{i,n}; \bar{\mu}_n, \frac{1}{k_0}\Sigma_{i,n})\, \mathcal{IW}(\Sigma_{i,n}; (d_n+2)S_n, d_n+2), \tag{6.23}$$

where $\bar{\mu}_n$ and S_n are as in Table 6.3.

In order to simulate $\Sigma_{i,n}$ according to an inverted Wishart distribution $\mathcal{IW}(\tilde{\Lambda}, \tilde{\nu})$, we use the algorithm of Table 6.4, which is a variant of Jones' algorithm [82]. In order to simulate $\mu_{i,n}$ according to a Gaussian distribution $\mathcal{N}(\tilde{\mu}, \tilde{\Sigma})$, we use the algorithm of Table 6.5. Again, an interesting advantage of the Gaussian/Inverted Wishart prior on the likelihood parameters $\mu_{i,n}$ and $\Sigma_{i,n}$ is that the posterior distribution of the parameters conditional to the cue process c_n is also a Gaussian/Inverted Wishart distribution as described in Table 6.6. See [2], Theorem 7.7.3. Furthermore, Table 6.6 shows that the matrix $\tilde{\Lambda}_{i,n}$ will be positive-definite even if the empirical matrix $S_{i,n}$ is singular, as long as the prior matrix Λ_0 is non-singular. This property is useful when analyzing an image in which some regions have a constant feature vector (because then, $S_{i,n} = 0$).

Prior on the region process

The prior on the region process x is given by the model $P(x \,|\, \beta, v)$ of equation (6.12).

Simulation of $\Sigma_{i,n}$ according to an inverted Wishart distribution $\mathcal{IW}(\tilde{\Lambda}, \tilde{\nu})$, with $\nu > d_n + 1$:

for $j = 1, ...d_n$ **do**

sample R_{jj} from a chi distribution with $\tilde{\nu} - j + 1$ degrees of freedom.

end for

for $1 \leq j < k \leq d_n$ **do**

sample R_{jk} from a Gaussian distribution $\mathcal{N}(0, 1)$.

end for

for $1 \leq k < j \leq d_n$ **do**

set $R_{jk} = 0$.

end for

Decompose $\tilde{\Lambda} + I$ in the form WDW^t, where W is an orthogonal matrix and D is a diagonal matrix.

Set $A = R^{-1}(D - I)^{1/2}W^t$.

Set $\Sigma_{i,n} = A^t A$.

Table 6.4. Simulation of an inverted Wishart distribution.

Simulation of $\mu_{i,n}$ according to a Gaussian distribution $\mathcal{N}(\tilde{\mu}, \tilde{\Sigma})$:

for $j = 1, ...d_n$ **do**

sample Z_j from a Gaussian distribution $\mathcal{N}(0, 1)$.

end for

Set $Z = (Z_j)$.

Decompose $\tilde{\Sigma} + I$ in the form WDW^t, where W is an orthogonal matrix and D is a diagonal matrix.

Set $\mu_{i,n} = \tilde{\mu} + W(D - I)^{1/2}Z$.

Table 6.5. Simulation of a Gaussian distribution.

Prior on the spatial hyper-parameter

We adopt the following non-informative improper prior distribution on the parameter β of the Markov prior model $P(x \mid \beta, v)$:

$$P(\beta \mid v) \quad \propto \quad 1. \tag{6.24}$$

Prior on the number of allocated regions

We consider the number of allocated region labels $|v|$ as a random variable of the form $1 + p$, where p is a Poisson variable with mean λ. So, $\mathcal{P}(p \mid \lambda) = \frac{\lambda^p e^{-\lambda}}{p!}$. We set Jeffrey's prior on the mean: $P(\lambda) = \frac{1}{\sqrt{\lambda}}$. With that model, we obtain $P(p) = \int P(p \mid \lambda)P(\lambda)\, d\lambda = \frac{1}{p!}\int \lambda^{p-1/2}e^{-\lambda}\, d\lambda = \frac{\Gamma(p+1/2)}{p!} = \frac{\Gamma(|v|-1/2)}{\Gamma(|v|)}$.

Taking also the number of permutations of the region labels into account, we obtain the following prior distribution on the vector of allowed region classes v defined by

Let n be a fixed level of analysis. Let c_n be a realization of the cue process at level n. Let $1 \leq i \leq K_n$ be fixed. Let $\mu_{i,n}, \Sigma_{i,n}$ have the prior of equation (6.20).

Compute $N_{i,n} = |\{s \ : \ c_{s,n} = g_{i,n}\}|$.

Compute the empirical mean $\bar{\mu}_{i,n}$ of the set $\mathcal{E}_{i,n} = \{h(y_{s,n}) \ : \ c_{s,n} = g_{i,n}\}$.

Compute its ML covariance matrix $S_{i,n}$.

Set

$$\begin{aligned} \tilde{\mu}_{i,n} &= \bar{\mu}_{i,n} - \frac{k_0}{N_{i,n} + k_0}(\bar{\mu}_{i,n} - \mu_0), \\ \tilde{\Lambda}_{i,n} &= \Lambda_0 + N_{i,n} S_{i,n} \\ &\quad + \frac{N_{i,n} k_0}{N_{i,n} + k_0}(\bar{\mu}_{i,n} - \mu_0)(\bar{\mu}_{i,n} - \mu_0)^t. \end{aligned}$$

Then, the posterior distribution of $(\mu_{i,n}, \Sigma_{i,n})$ given c_n is the Gaussian/Inverted-Wishart distribution $\mathcal{N}(\mu_{i,n}; \tilde{\mu}_{i,n}, \frac{1}{N_{i,n}+k_0}\Sigma_{i,n})\mathcal{IW}(\Sigma_{i,n}; \tilde{\Lambda}_{i,n}, N_{i,n} + \nu_0)$.

Table 6.6. Expression of the posterior Gaussian/inverted Wishart distribution.

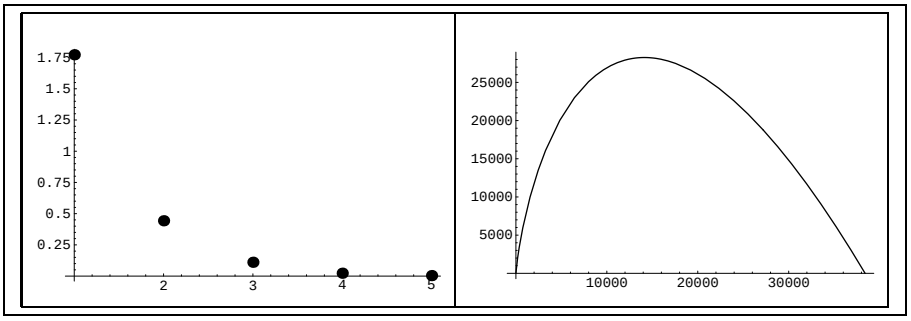


Figure 6.4. Left: Prior distribution on the number $|v|$ of allocated region labels (see Section 6.3.1). Right: Global constraint on the number of connected regions in the case of 38400 pixels and equal size components (see Section 6.3.1).

$$P(v) \propto \frac{1}{|v|!}\frac{\Gamma(|v|-1/2)}{\Gamma(|v|)} = \frac{\Gamma(|v|-1/2)}{\Gamma(|v|+1)\Gamma(|v|)}. \tag{6.25}$$

See Fig. 6.4 for an illustration of the shape of the proposed distribution.

Prior distribution on the parameters

Altogether, we obtain the following prior on the parameters θ:

$$P(\theta) \quad \propto \quad P(\pi \mid v)\, P(\mu, \Sigma)\, P(x \mid \beta, v)\, P(\beta \mid v) P(v), \tag{6.26}$$

as defined in equations (6.19), (6.23), (6.12), (6.24), and (6.25).

Likelihood

We find convenient to consider the augmented data $(\hat{y}, \hat{c})$. The joint distribution of the augmented model is described by equation (6.8), whereas the marginal distribution of $\hat{y}$ given x is described by equation (6.9). Thus, the corresponding likelihoods can be expressed as

$$P(\hat{y}, \hat{c} \mid \theta) = P(\hat{y} \mid \hat{c}, \mu, \Sigma) P(\hat{c} \mid x, \pi); \tag{6.27}$$

$$P(\hat{y} \mid \theta) = P(\hat{y} \mid x, \mu, \Sigma, \pi). \tag{6.28}$$

Posterior distribution on the parameters

Finally, the corresponding posterior distributions on the parameters θ are expressed as

$$P(\theta, \hat{c} \mid \hat{y}) \quad \propto \quad P(\hat{y}, \hat{c} \mid \theta) P(\theta); \tag{6.29}$$

$$P(\theta \mid \hat{y}) \quad \propto \quad P(\hat{y} \mid \theta) P(\theta). \tag{6.30}$$

The Directed Acyclic Graph (DAG) presented in Figure 6.5 summarizes the proposed Bayesian model.

Global constraint

Given a segmentation x, let $R_1(x), ..., R_\ell(x)$ be the ℓ connected regions induced by x. That is, let G' be the graph with nodes the pixels of the image, in which two pixels s and t are connected if they are 8-neighbors and if they have the same label ($x_s = x_t$). Then, $R_1(x), ..., R_\ell(x)$ are the connected components of the graph G'. Under the "cubic law" of 3D-object sizes [95], the probability of observing a connected component of size $|R|$ is proportional to $\frac{1}{|R|^2}$. If we let the size $|R|$ vary from 1 to the size $|G|$ of the image, the constant of proportionality is $\frac{1}{(1-|G|^{-1})}$.

Also, we consider the number of connected regions ℓ to be of the form $1 + p$ where p is a Poisson variable of unknown mean λ. As in Section 6.3.1, the marginal distribution of ℓ is equal to $\frac{\Gamma(\ell-\frac{1}{2})}{\Gamma(\ell)}$.

Taking $-\log$ of the combined probabilities, we obtain a global constraint on the region process x of the form

$$\begin{aligned} \rho(x) &= \sum_{i=1}^{\ell} \big\{ 2 \log |R_i(x)| + \log(1 - 1/|G|) \big\} \\ &+ \log \Gamma(\ell) - \log \Gamma(\ell - \frac{1}{2}). \end{aligned} \tag{6.31}$$

Figure 6.4 illustrates the shape of the proposed global constraint, in the special case of equal size components (i.e., $|R_1(x)| = ... = |R_\ell(x)| = |G|/\ell$). In this case, we took $|G| = 38400$ pixels. In practice, only the increasing part of the curve is relevant, since the function starts to decrease after as much as 12000 connected regions. The likelihood of a natural image prevents this case to occur. In our tests, the average number of connected regions was 521.89.

Weighted MAP estimator

Due to the intractable computation of the partition function $Z(\beta, v)$, the ML estimator of β cannot be computed. So, we replace the likelihood (as is often done) by the pseudo-likelihood [13]. Since the pseudo-likelihood estimator of an MRF is *consistent* [29], nothing is lost in the estimation of the hyper-parameter β, at least for sufficiently large images. The pseudo-likelihood estimator $\hat{\beta}(x)$ is the maximum of the function

$$\begin{aligned} & -\frac{1}{2}\sum_{s}\sum_{t\in N(s)}\beta U_{\langle s,t\rangle}(x) \\ - \quad & \frac{1}{2}\sum_{s}\log\left(\sum_{k\,:\,v_k=1} e^{-\sum_{t\in N(s)}\beta U_{\langle s,t\rangle}(x\,|\,x_s=e_k)}\right), \end{aligned} \tag{6.32}$$

where $N(s)$ is the set of neighbors of the pixel s. The factor $1/2$ takes into account the fact that each binary clique is counted twice in the pseudo-likelihood term.

Therefore, we propose the following *weighted* Maximum A Posteriori (MAP) estimator $\theta_* = (x_*, \mu_*, \Sigma_*, \pi_*, \beta_*, v_*)$ of the fusion model of equation (6.30): the values of the parameters that maximize the function

$$P(x, \mu, \Sigma, \pi, \beta, v \,|\, \hat{y})\, \frac{Z(\beta, v)}{Z_p(x, \beta, v)}\, e^{-\rho(x)}, \tag{6.33}$$

where the pseudo-partition function $Z_p(x, \beta, v)$ is equal to

$$\prod_{s}\Big(\sum_{k\,:\,v_k=1} e^{-\sum_{t\in N(s)}\beta U_{\langle s,t\rangle}(x\,|\,x_s=e_k)}\Big)^{1/2}. \tag{6.34}$$

Equivalently, $(x_*, \mu_*, \Sigma_*, \pi_*, \beta_*, v_*)$ is a global minimum of the energy function $\bar{f}(x, \mu, \Sigma, \pi, \beta, v \,|\, \hat{y})$ defined by

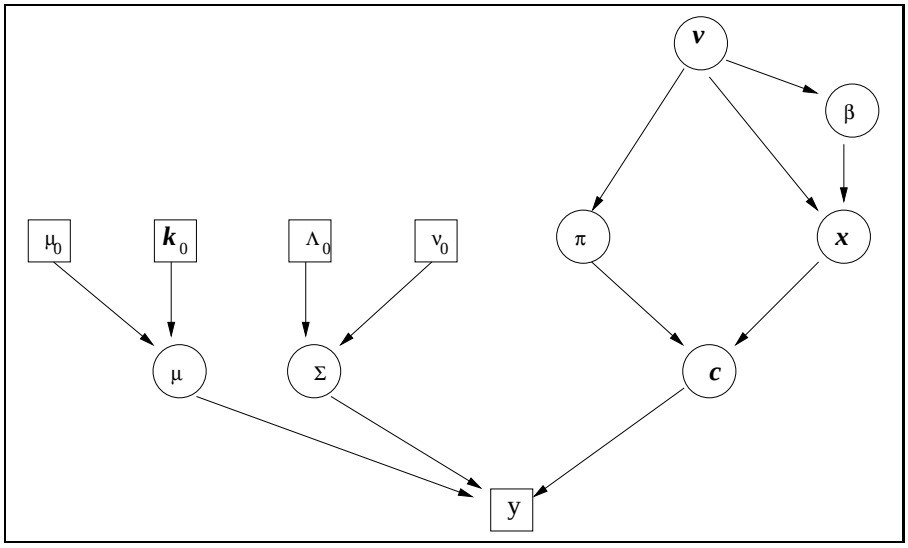


Figure 6.5. DAG of the proposed fusion model.

$$
\begin{aligned}
& -\sum_{n=1}^{M} \log P(y_n \mid x, \mu_n, \Sigma_n, \pi_n) \\
+ & \sum_{n=1}^{M} \log P(\pi_n \mid v) + \log P(\mu_n, \Sigma_n) \\
+ & \beta U(x) + \log Z_p(x, \beta, v) - \log P(v) + \rho(x). \qquad (6.35)
\end{aligned}
$$

Note that β_* must be equal to the pseudo-likelihood estimator $\hat{\beta}(x_*)$ on the optimal segmentation x_*. Also, the dependence of $\bar{f}$ on v is only implicit. Namely, v is the set of labels appearing in x (otherwise, $\bar{f}(x, \mu, \Sigma, \pi, \beta, v \mid \hat{y}) = \infty$). See Table 6.7 for an algorithm that computes the pseudo-likelihood estimator. This algorithm works because the function defined by equation (6.32) is a concave function.

Identifiability

An important property in statistics is the *identifiability* of the model (c.f. [135]). In the context of the proposed model, this property can be stated as follows.

Let x be a realization of the region process. Let $|v|$ be the number of region classes appearing in x. Let $p'(\beta)$ be the derivative of equation (6.32) with respect to β.

if $|v| = 1$ **then**
  Return $\beta = 0$.
else
  Set $\beta_1 = 0$, and $\beta_2 = 1$.
  while $p'(\beta_2) > 0$ **do**
    Multiply β_2 by 2.
  end while
  while $\beta_2 - \beta_1 > 0.01$ **do**
    Set $\beta = (\beta_1 + \beta_2)/2$.
    if $p'(\beta) > 0$ **then**
      Set $\beta_1 = \beta$.
    else
      Set $\beta_2 = \beta$.
    end if
  end while
  Return β.
end if

Table 6.7. Computation of the maximum pseudo-likelihood parameter.

Theorem 1

Let $\theta_i = (x_i, \mu_i, \Sigma_i, \pi_i, \beta_i, v_i)$ with $\beta_i = \hat{\beta}(x_i)$, for $i = 1, 2$, be two vectors of parameters that induce the same values of the energy function $\bar{f}(\theta \mid \hat{y})$ for any observable data $\hat{y}$. Then, $\theta_1 = \theta_2$ (up to permutation of the indices).

The proof of the Theorem is postponed until Appendix 6.6. A practical consequence is that, for large images, the parameters are uniquely determined by the observed data. In particular, the weighted MAP is practically unique.

6.3.2 Stochastic algorithm

We find convenient to use the augmented data $\hat{c}$ in the calculation of $(x_*, \mu_*, \Sigma_*, \pi_*, \beta_*, v_*)$. Also, we consider an auxiliary integer $1 \leq L \leq K$ that represents the number of allocated region labels so far. The role of L in the algorithm will appear clearer later. Thus, we consider the augmented vector $\psi = (x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$ and we define a function $f(x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L) = \bar{f}(x, \mu, \Sigma, \pi, \beta, v)$. Clearly, for any $\hat{c}$ and L, the vector $(x_*, \hat{c}, \mu_*, \Sigma_*, \pi_*, \beta_*, v_*, L_*)$ is optimal for f if and only if $(x_*, \mu_*, \Sigma_*, \pi_*, \beta_*, v_*)$ is optimal for $\bar{f}$. Thus, we want to optimize the augmented function f on the (augmented) search space A consisting of all admissible 8-tuples $(x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$; i.e., $v_k = 1$ if and only if $x_s = e_k$ for some pixel s, $v_k = 0$ for $k > L$, and $\beta = \hat{\beta}(x)$. Note that we can view the space A as finite, upon using the ε-machine of the computer. That is, in reality, we are working with a finite set!

In order to solve this optimization problem, we resort to O. François' Exploration/Selection (E/S) algorithm [59]. The goal of the algorithm is to minimize a function f on a finite set A. The algorithm relies on an exploration kernel $a(\psi, \psi')$, with $\psi, \psi' \in A$, which gives the probability of reaching ψ' from ψ under an exploration operator.

Given $m > 1$, a vector of m solutions in the Cartesian product A^m is denoted $\boldsymbol{\psi}$. Such a vector is called a population of solutions. Given a population of solutions

$\boldsymbol{\psi} = (\psi_1, ..., \psi_m)$, $\alpha(\boldsymbol{\psi})$ denotes the best solution (with minimal index in case of a tie) among $\psi_1, ..., \psi_m$; i.e., $\alpha(\boldsymbol{\psi}) = \psi_l$, where $f(\psi_k) > f(\psi_l)$ for $k < l$, and $f(\psi_k) \geq f(\psi_l)$ for $k \geq l$.

At each step, two operators are available. An exploration operator that draws a new solution ψ'_l according the kernel $a(\psi_l, \cdot)$; a selection operator that replaces ψ_l by the best current solution $\alpha(\boldsymbol{\psi})$. The exploration operator is performed with probability p_t at iteration t. The probability of exploration p_t is set equal to $t^{-1/\tau}$, where $\tau \geq 1$ is a parameter of the algorithm. The random state of the vector $\boldsymbol{\psi}$ at iteration t is denoted $\psi^{[t]}$. The E/S algorithm is summarized in Table 6.8.

In the original version of the algorithm [59], the exploration kernel $a(\psi, \psi')$ is a uniform distribution on a neighborhood $N(\psi)$ of ψ with $\alpha(\boldsymbol{\psi})$ deleted. In this paper, the exploration kernel can be any distribution that satisfies the following hypothesis:

$$\text{For any } \psi, \psi' \in A,\ a(\psi, \psi') > 0. \tag{6.36}$$

Theorem 2

Let hypothesis (6.36) hold. For any $t \geq 1$ and any $\varepsilon > 0$,

$$\begin{aligned} & P\{f(\alpha(\boldsymbol{\psi}^{[t+1]})) \geq f_* + \varepsilon\} \\ & \leq t^{-\frac{m}{\tau}}(1 - P_\varepsilon)^m(1 - P\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq f_* + \varepsilon\}) \\ + \quad & (1 - P_\varepsilon t^{-\frac{1}{\tau}})^m P\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq f_* + \varepsilon\}, \end{aligned}$$

where f_* is the global minimum of f on A, and $P_\varepsilon = \min_\psi P_{a(\psi,\cdot)}(f < f_* + \varepsilon)$.

Theorem 2 is useful because we obtain a sequence converging to 0, as shows the following Lemma.

Lemma 1

Let b_t be the sequence defined recursively by

$$\begin{aligned} b_1 &= 1; \\ b_{t+1} &= t^{-\frac{m}{\tau}}(1-P)^m(1-b_t) + (1-Pt^{-\frac{1}{\tau}})^m b_t, \end{aligned}$$

where $0 < P \leq 1$, $m \geq 2$, *and* $\tau \geq 1$. *Then,* $\lim_{t\to\infty} b_t = 0$.

Corollary 1

For any $\varepsilon > 0$, $\lim_{t\to\infty} P\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq f_* + \varepsilon\} = 0$.

Proof: This follows directly from Theorem 2 and Lemma 1, upon observing that hypothesis (6.36) implies that $P_\varepsilon > 0$. □

Thus, we recover a new proof of the asymptotic convergence of the ES algorithm under hypothesis (6.36). But Theorem 2 actually gives unilateral confidence intervals for $f(\alpha(\boldsymbol{\psi}^{[t]}))$. For instance, with $m = 6$, $\tau = 3$, $P_\varepsilon = 0.01$, we obtain $P\left\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq f_* + \varepsilon\right\} \leq 1.225 \times 10^{-4}$ if $t \geq 1465$. If $t \geq 8352$, we obtain $P\left\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq f_* + \varepsilon\right\} \leq 4.912 \times 10^{-6}$.

In practice, condition (6.36) can be met as follows. Let $a(\psi, \psi')$ be any exploration kernel that satisfies the hypothesis:

$$\begin{aligned} &\text{For any } \psi, \psi' \in A, \text{ there exists } \psi_0 = \psi, \psi_1, ..., \psi_D = \psi' \\ &\text{such that } a(\psi_i, \psi_{i+1}) > 0, \end{aligned} \tag{6.37}$$

where $D \geq 1$ is called the diameter. If $D = 1$, the kernel a satisfies itself condition (6.36). If $D > 1$, consider the modified kernel $\tilde{a}(\psi, \psi')$ defined by the algorithm of Table 6.9. The idea is to simply repeat the kernel a a random number of times between 1 and D. Then, clearly, the modified kernel satisfies condition (6.36), because there is a positive probability of performing consecutively D times the exploration according to the kernel a.

There is a closed connection between the ES algorithm and the simulated annealing. To see this, let $p_t = \exp(-\frac{1}{T})$, where $T > 0$ is called the temperature. Then, if

$T = T(t)$ is given by the usual simulated annealing temperature schedule $T = \frac{\tau}{\log t}$, we recover $p_t = t^{-1/\tau}$ as in Table 6.8. In fact, it is shown in [59] that the E/S algorithm converges to an optimal solution (under hypothesis (6.37)) if and only if T is of the form $\frac{\tau}{\log t}$, with an appropriate value of τ (in particular, it is sufficient that $\tau \geq D$).

The ESE procedure [51] is a variant of the E/S algorithm designed in the case where A is a space of parameters and f is $-\log$ the posterior distribution of the parameters conditional to the observed data. Again, after digitization of the space, A can be viewed as finite. The main idea is to use an MCMC kernel of the posterior distribution as exploration kernel. In practice, this crucial idea helps the algorithm perform efficiently; in particular, using a uniform distribution would yield a very poor algorithm in our case.

We can build systematically the MCMC kernel upon using the Gibbs sampler. Namely, one transition consists in performing the following sampling steps:

1. $v^{[t+1]} \sim v \mid x^{[t]}, c^{[t]}, \mu^{[t]}, \Sigma^{[t]}, \pi^{[t]}, \beta^{[t]}$.

2. $x^{[t+1]} \sim x \mid \hat{c}^{[t]}, \mu^{[t]}, \Sigma^{[t]}, \pi^{[t]}, \beta^{[t]}, v^{[t+1]}$.

3. $\hat{c}^{[t+1]} \sim \hat{c} \mid x^{[t+1]}, \mu^{[t]}, \Sigma^{[t]}, \pi^{[t]}, \beta^{[t]}, v^{[t+1]}$.

4. $\pi^{[t+1]} \sim \pi \mid x^{[t+1]}, \hat{c}^{[t+1]}, \mu^{[t]}, \Sigma^{[t]}, \beta^{[t]}, v^{[t+1]}$.

5. $\mu^{[t+1]}, \Sigma^{[t+1]} \sim \mu, \Sigma \mid x^{[t+1]}, \hat{c}^{[t+1]}, \pi^{[t+1]}, \beta^{[t]}, v^{[t+1]}$.

6. $\beta^{[t+1]} \sim \beta \mid x^{[t+1]}, \hat{c}^{[t+1]}, \mu^{[t+1]}, \Sigma^{[t+1]}, \pi^{[t+1]}, v^{[t+1]}$.

However, we need to bring some modifications to this general scheme.

- In step (2), one needs to perform many sweeps of the image, and moreover take into account the global constraint $\rho(x)$ in a Metropolis-Hastings (M-H) strategy. But this is unnecessary in our case, because the proposal/disposal mechanism

of the M-H algorithm is replaced by the exploration/selection mechanism of the ES algorithm. The point is that our goal in this paper is not to simulate the posterior distribution of ψ, but rather to compute the MAP estimator.

- Step (1) should be combined with step (2) in a Reversible Jump Monte Carlo Markov Chain (RJMCMC) strategy in order to simulate the posterior distribution of (x, v) with jumps in dimension. Note that engineering a RJMCMC kernel that offers sufficiently high rates of acceptance is a hard task in practice. But again, all we need, is an exploration kernel that satisfies hypothesis (6.37) (for some value of D). So, we replace step (1) by an *ad hoc* exploration $v^{[t+1]} \sim v \mid v^{[t]}$, as described in Table X. The point is that in the sampling of the region labels $x^{[t+1]}$ in step (2), only those labels allowed by $v^{[t+1]}$ are used. In this manner, an artificial jump in dimension is performed.

- In step (4), whenever a region class e_k is empty, we do not simulate the mixture proportions according to the prior distribution. Rather, we keep the former values of the mixture proportions for this region class intact for a subsequent iteration. Similarly for step (5).

- As explained in Section 6.3.1, in step (6), we are actually interested in taking $\beta^{[t+1]} = \hat{\beta}(x^{[t+1]})$ instead of simulating β.

The resulting modified Gibbs sampler is presented in Table 6.10.

Now, we want to start with one region (i.e., $L = 1$) and let the number of allowed regions grow gradually until it reaches the maximal value K (i.e., $L = K$). In order to do so, we consider an operator of birth of a region explained in Table 6.12. The idea of using such an operator can be found in [131]. Altogether, the exploration kernel $a(\cdot, \cdot)$ used in this paper consists of Table 6.12 followed by Table 6.10. The resulting kernel satisfies hypothesis (6.37) with $D = K$. Then, use Table 6.9 with $D = K$ to

Let $m > 1$ and $\tau \geq 1$. Goal: to minimize a function f on a finite set A.

Parameter initialization: Initialize randomly $\psi_l^{[0]}$, for $l = 1, ..., m$. Set $t = 1$.

while a stopping criterion is not met **do**

Update $t \leftarrow t + 1$.

Determine the best current solution $\alpha(\boldsymbol{\psi}^{[t]}) = \psi_l^{[t]}$, where $f(\psi_k^{[t]}) > f(\psi_l^{[t]})$ for $k < l$, and $f(\psi_k^{[t]}) \geq f(\psi_l^{[t]})$ for $k > l$.

for $l = 1, 2, ..., m$ **do**

Let u be a random number between 0 and 1.

if $u \leq p_t = t^{-\frac{1}{\tau}}$ **then**

Exploration: Replace $\psi_l^{[t]}$ by $\psi_l^{[t+1]}$ drawn according to an exploration kernel $a(\psi_l^{[t]}, \cdot)$.

else

Selection: Replace $\psi_l^{[t]}$ by $\alpha(\boldsymbol{\psi}^{[t]})$.

end if

end for

end while

Table 6.8. ES algorithm in its general form.

obtain a kernel $\tilde{a}$ that satisfies hypothesis (6.36). Note that at the intermediate steps, the vector ψ might not be admissible, but that at the output ψ *is* admissible.

Finally, we present the initialization steps in Table 6.13. The results above imply that the whole procedure converges asymptotically to the weighted MAP $(x_*, \mu_*, \Sigma_*, \pi_*, \beta_*, \upsilon_*)$, with probability 1. In our tests, we took $m = 6$ and $\tau = 3$ in Table 6.8. Furthermore, we waited for the first 10 iterations before increasing the number of allowed regions (c.f. Table 6.12).

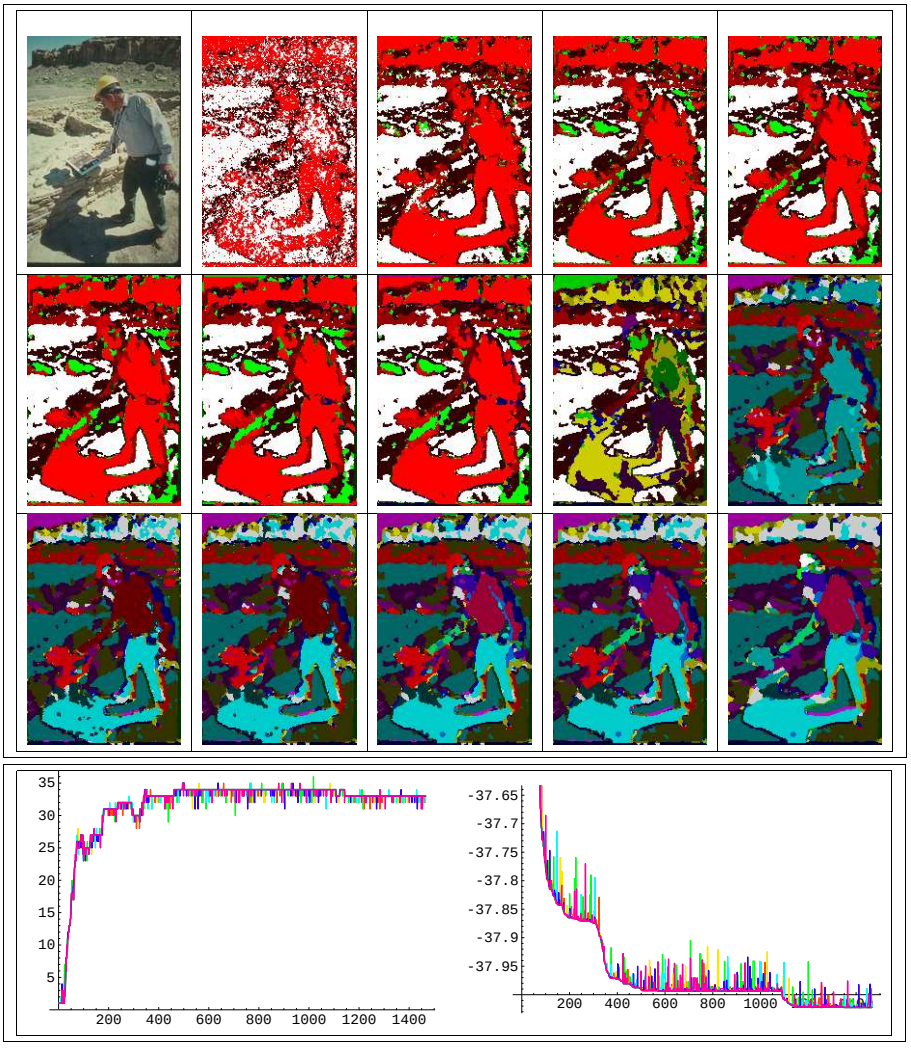


Figure 6.6. Top: A natural image and the formation of its region process at iterations 25, 30, 35, ..., 85, and 1465. Bottom: evolution of the number of region classes and the Gibbs energy (i.e., the value of the function f) of 6 solutions as a function of the iteration.

Modified exploration kernel $\tilde{a}(\psi, \psi')$ from a kernel $a(\psi, \psi')$ satisfying hypothesis (6.37).

Draw N according to a Binomial distribution $\mathcal{B}(p, n)$ with $p = 0.1/(D-1)$ and $n = D - 1$. Set $\psi_0 = \psi$.

for $i = 0, ..., N$ **do**

Draw ψ_{i+1} according to the kernel $a(\psi_i, \cdot)$.

end for

Set $\psi' = \psi_{N+1}$.

Table 6.9. Modification of an ES exploration kernel a satifying hypothesis (6.37) into a kernel $\tilde{a}$ that satisfies hypothesis (6.36).

6.4 Experimental Results

We have tested the proposed method of estimation and segmentation on the University of California at Berkeley (UCB) test dataset [105] of 100 natural images in ".jpg" format. We think that all of them are optical images obtained by electronic acquisition, though we do not have that information at hand. The typical size of an image was 481×321. Each image was reduced by a factor of 50%. In our implementation in C++, we use the GNU scientific library of functions.

We performed for each natural image I, a joint estimation and segmentation $(x_*, \phi_*, \pi_*, \beta_*, v_*)$ based on the observed channels data $\hat{y}(I)$, with a maximal number of $K = 40$ allowed classes, and a fixed number of $K_1 = 16$ color classes and $K_2 = 16$ texton classes. This represents a task of estimating 38400 color labels, 38400 texton labels, 38400 region labels, 144 Gaussian color parameters, 864 Gaussian texton parameters, 30 mixture parameters per region class, and one Markovian

Let $\psi = (x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$. Modified Gibbs sampler: $\psi^{[t]} \to \psi^{[t+1]}$.

1. $(v^{[t+1]} \sim v \,|\, v^{[t]})$ Let $L^{[t]}$ be the number of labels allocated so far. Modify each bit of $v^{[t]}$, with index $k \leq L^{[t]}$, with probability $\frac{1}{2L^{[t]}}$. If they are all equal to 0, set one of them equal to 1 randomly. Let $v^{[t+1]}$ be the resulting vector of allowed classes, and set $L^{[t+1]} = L^{[t]}$.

2. $(x^{[t+1]} \sim x \mid \hat{c}^{[t]}, \mu^{[t]}, \Sigma^{[t]}, \pi^{[t]}, \beta^{[t]}, v^{[t+1]})$ For two sweeps, set $x_s = e_k$ at each site s of the image graph V according to the weights

$$\exp\left\{-\beta^{[t]} \sum_{s' \in N(s)} U_{\langle s,s' \rangle}(x \mid x_s = e_k)\right\} \prod_{n=1}^{M} \pi^{[t]}_{(i_n,k),n},$$

   where $N(s)$ denotes the set of 8-neighbors of s in the image graph G, $c_{s,n} = g_{i_n,n}$, and e_k ranges over the set of allocated classes by $v^{[t+1]}$. Let $x^{[t+1]}$ be the updated segmentation. Readjust $v^{[t+1]}$ so that $v_k^{[t+1]} = 1$ if and only if $x_s^{[t+1]} = e_k$ for some pixel s.

3. $(\hat{c}^{[t+1]} \sim \hat{c} \,|\, x^{[t+1]}, \mu^{[t]}, \Sigma^{[t]}, \pi^{[t]}, \beta^{[t]}, v^{[t+1]})$ At each level n, for each pixel s, draw $c_{s,n}$ according to the weights

$$P(y_{s,n} \,|\, c_{s,n} = g_{i,n}, \mu^{[t]}_{i,n}, \Sigma^{[t]}_{i,n}) \pi^{[t]}_{(i,k),n},$$

   where $x_s = e_k$. Let $c^{[t+1]}$ be the updated segmentations.

Table 6.10. Modified Gibbs sampler for the ESE procedure (part 1).

4. $(\pi^{[t+1]} \sim \pi \mid x^{[t+1]}, \hat{c}^{[t+1]}, \mu^{[t]}, \Sigma^{[t]}, \beta^{[t]}, v^{[t+1]} = \pi \mid x^{[t+1]}, \hat{c}^{[t+1]}, v^{[t+1]})$ For each level n and for each allowed region class e_k, simulate $\pi_{(i,k),n}$ according to the posterior distribution of Table 6.2. But, if ever the class is empty, the former value of $\pi_{(i,k),n}$ is kept. Let $\pi^{[t+1]}$ be the resulting mixture proportions.

5. $(\mu^{[t+1]}, \Sigma^{[t+1]} \sim \mu, \Sigma \mid x^{[t+1]}, \hat{c}^{[t+1]}, \pi^{[t+1]}, \beta^{[t]}, v^{[t+1]} = \mu, \Sigma \mid \hat{c}^{[t+1]})$ For each level n and each non-empty class $g_{i,n}$, simulate $\mu_{i,n}, \Sigma_{i,n}$ according to the posterior distribution of Table 6.6. But, if ever $c_{s,n} \neq g_{i,n}$ for all $s \in V$, the former values of $\mu_{i,n}, \Sigma_{i,n}$ are kept. Let $\mu^{[t+1]}, \Sigma^{[t+1]}$ be the resulting likelihood parameters.

6. Compute the pseudo-likelihood estimator $\beta^{[t+1]} = \hat{\beta}(x^{[t+1]})$ as in Table 6.7.

Table 6.11. Modified Gibbs sampler for the ESE procedure (part 2).

Let $\psi = (x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$. Birth of a region: $\psi^{[t]} \rightarrow \psi^{[t+1]}$.

Let $L^{[t]}$ be the number of classes allocated in $v^{[t]}$ so far. Set $L^{[t+1]} = L^{[t]} + 1$, provided $L^{[t]} < K$. Otherwise, set $L^{[t+1]} = L^{[t]}$ and return.

Set $v_k^{[t+1]} = v_k^{[t]}$, for $k \leq L^{[t]}$, and $v_{L^{[t+1]}}^{[t+1]} = 1$.

for each level of analysis n **do**

Sample $(\pi_{(1,L^{[t+1]}),n}, ..., \pi_{(K_n,L^{[t+1]}),n})$ according to the Dirichlet prior $\mathcal{D}(A_0, \alpha_1, ..., \alpha_{K_n})$.

end for

Let $\pi_{L^{[t+1]}}^{[t+1]}$ be the resulting proportions and set $\pi_k^{[t+1]} = \pi_k^{[t]}$ for $k \leq L^{[t]}$.

Table 6.12. Operator of birth of a region for the ESE procedure.

Let $\psi = (x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$. Initialization of $\psi^{[0]}$.

1. Set $v_1^{[0]} = 1$, and $v_k^{[0]} = 0$ for $k \geq 2$. Set $L^{[0]} = 1$.

2. Set $x_s^{[0]} = e_1$, for all pixels s.

3. Random initialization of $\hat{c}^{[0]}$.

4. Perform steps 4 to 6 of the exploration kernel of Table 6.11.

Table 6.13. Initialization of the ESE procedure.

hyper-parameter. We then simulated the image channels $\hat{y}'$ based on that estimation. Thus, $\hat{y}'$ and $(x_*, \phi_*, \pi_*, \beta_*, v_*)$ were considered as ground-truth. Note that the image I' itself was not simulated. Next, we performed a joint estimation and segmentation $(x'_*, \phi'_*, \pi'_*, \beta'_*, v'_*)$ for the synthetic image, with again $K = 40$ and $K_1 = K_2 = 16$. We took $m = 6$ and $\tau = 3$ as internal parameters of the ESE procedure. The procedure was stopped after $t = 1465$ iterations (see Section 6.3.2).

We evaluated the estimation error with the measure

$$\Delta_0 = |\bar{f}(x_*, \phi_*, \pi_*, \beta_*, v_* \mid \hat{y}') - \bar{f}(x'_*, \phi'_*, \pi'_*, \beta'_*, v'_* \mid \hat{y}')|,$$

where $\bar{f}$ is defined by equation (6.35), as well as the relative measure proposed in [51]

$$\begin{aligned} \Delta_1 &= \frac{|\bar{f}(x_*, \phi_*, \pi_*, \beta_*, v_* \mid \hat{y}') - \bar{f}(x'_*, \phi'_*, \pi'_*, \beta'_*, v'_* \mid \hat{y}')|}{|\bar{f}(x_*, \phi_*, \pi_*, \beta_*, v_* \mid \hat{y}')|} \\ &\times 100\%. \end{aligned}$$

For the first estimation, the average number of region classes was $31.08 \pm 4.208 SD$ (the maximum 40 was reached for only one image out of 100), while the average number of connected regions was $489.79 \pm 418.18 SD$ (or $349.12 \pm 234.34 SD$ if singletons are omitted). For the second estimation, the average number of region classes was $32.03 \pm 6.509 SD$ (the maximum 40 was never reached), while the average number of connected regions was $467.83 \pm 408.16 SD$ (or $236.12 \pm 159.14 SD$ if singletons are omitted). The ESE procedure took on average 3 hrs and 26 min. on a Workstation 2.4GHz for an average of 1046.44 explorations. This represents roughly 11.81 sec. per exploration. The complexity of each exploration is actually linear in the size of the image times the number of region classes.

See Fig. 6.7 for a histogram of Δ_0 and Δ_1 over the dataset, and Fig. 6.1, 6.6 and 6.8 for examples of segmentations. The three images 175043.jpg, 38082.jpg and 69040.jpg were totally missed ($\Delta_0 = 2.145700, 2.371300, 1.792500$ and $\Delta_1 =$

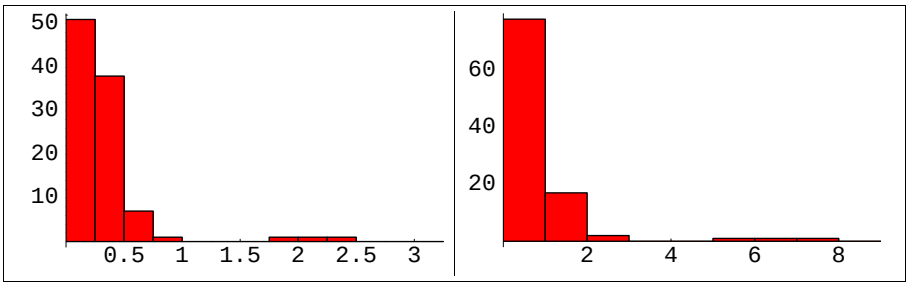


Figure 6.7. Histograms of the evaluation measures Δ_0 and Δ_1 over the dataset. Mean of Δ_0: 0.308919; mean of Δ_1: 0.84%.

$6.954818\%, 7.005151\%, 5.494504\%$, respectively). In fact, the current number of allowed region classes was only 1 at iteration 1465. We increased the number of iterations to 8352 and obtained successfully $\Delta_0 = 0.499901, 0.152901, 0.212399$ and $\Delta_1 = 1.538261\%, 0.423909\%, 0.620983\%$, respectively.

6.5 Conclusion

We have presented an HMRF data-fusion model based on Julesz ensembles and applied it to the segmentation of natural images. The ESE procedure [51] is a general method for estimating the weighted modes of HMRF models, with global constraints taken into account. We have shown how to adapt it to the proposed data fusion model. Not only the parameters of the Gaussian kernels and the region labels were estimated, but also the mixture proportions, the number of regions, and the Markov hyper-parameter. The internal parameters of the algorithm that insure asymptotic convergence to an optimal solution are known explicitly and are practical [51]. Furthermore, we have presented new finite time bounds for the rate of convergence. The tests reported in this paper indicate that the ESE procedure succeeds in finding the optimal solution of the proposed fusion model, within a relative error bound of less than 0.87% on average.

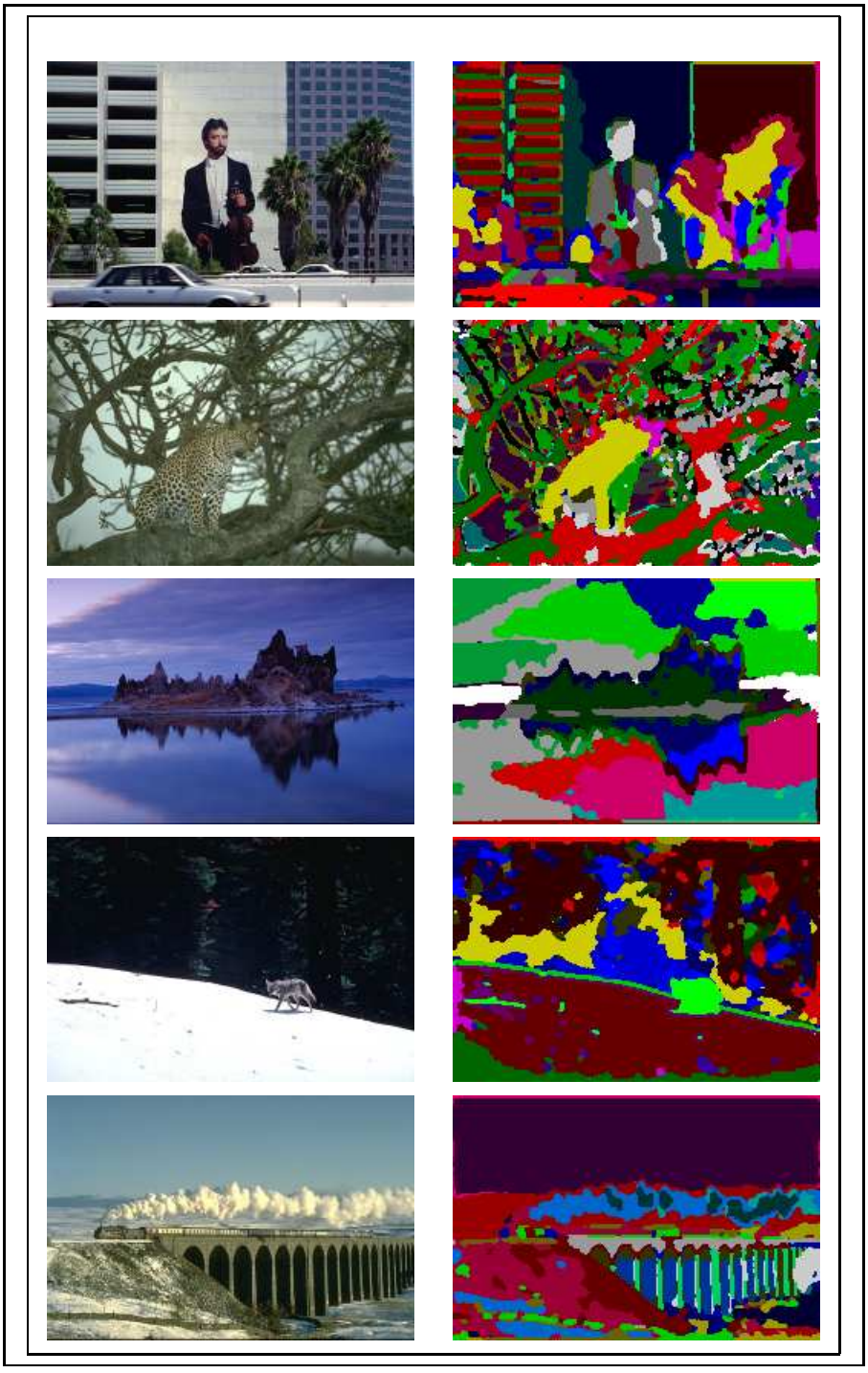


Figure 6.8. Examples of segmentations based on a fusion of colors and textons.

It remains to test the fusion model in various higher-level tasks, such as image indexing, 3D-reconstruction, motion detection, or localization of shapes, in combination with prior knowledge on the particular problem.

6.6 Appendix I

We present in this appendix a proof of Theorem 1 of Section 6.3.1.

Let $\theta_i = (x_i, \mu_i, \Sigma_i, \pi_i, \beta_i, v_i)$, $i = 1, 2$, be two vectors of parameters, with $\beta_i = \hat{\beta}(x_i)$. Assume that $\bar{f}(\theta_1 \mid \hat{y}) = \bar{f}(\theta_2 \mid \hat{y})$ for all $\hat{y}$. This means that

$$\begin{aligned}
& \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid x_1, \mu_{1,n}, \Sigma_{1,n}, \pi_{1,n}) \\
= & \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid x_2, \mu_{2,n}, \Sigma_{2,n}, \pi_{2,n}) \\
\times & \frac{P(\pi_{2,n} \mid v_2) P(\mu_{2,n}, \Sigma_{2,n})}{P(\pi_{1,n} \mid v_1) P(\mu_{1,n}, \Sigma_{1,n})} \\
\times & \frac{P(\beta_2 \mid v_2) e^{-\beta_2 U(x_2)} Z_p(x_2, \beta_2, v_2)^{-1} e^{-\rho(x_2)} P(v_2)}{P(\beta_1 \mid v_1) e^{-\beta_1 U(x_1)} Z_p(x_1, \beta_1, v_1)^{-1} e^{-\rho(x_1)} P(v_1)}.
\end{aligned}$$

In particular, we obtain an equality of distributions

$$\begin{aligned}
& \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid x_1, \mu_{1,n}, \Sigma_{1,n}, \pi_{1,n}) \\
& = \prod_{n=1}^{M} \prod_{s} P(y_{s,n} \mid x_2, \mu_{2,n}, \Sigma_{2,n}, \pi_{2,n}),
\end{aligned}$$

as well as the equality

$$\begin{aligned} 1 &= \frac{P(\pi_{2,n} \mid v_2) P(\mu_{2,n}, \Sigma_{2,n})}{P(\pi_{1,n} \mid v_1) P(\mu_{1,n}, \Sigma_{1,n})} \\ &\times \frac{P(\beta_2 \mid v_2) e^{-\beta_2 U(x_2)} Z_p(x_2, \beta_2, v_2)^{-1} e^{-\rho(x_2)} P(v_2)}{P(\beta_1 \mid v_1) e^{-\beta_1 U(x_1)} Z_p(x_1, \beta_1, v_1)^{-1} e^{-\rho(x_1)} P(v_1)}. \end{aligned}$$

Considering the marginals, we deduce

$$P(y_{s,n} \mid x_1, \mu_{1,n}, \Sigma_{1,n}, \pi_{1,n}) = P(y_{s,n} \mid x_2, \mu_{2,n}, \Sigma_{2,n}, \pi_{2,n})$$

for each pixel s and each model n. Indeed, the variables $y_{s,n}$ are mutually independent. It follows at once from the identifiability property [135] of mixtures of Gaussian distributions, that after relabeling of indices, $\pi_1 = \pi_2$ on allowed region classes, and $\mu_1 = \mu_2$, $\Sigma_1 = \Sigma_2$ on cue classes. We deduce immediately that $x_1 = x_2$ w. p. 1, since distinct regions have distinct mixture proportions w. p. 1. Furthermore $v_1 = v_2$, since v indicates which region labels are present in the segmentation x.

Finally, we conclude that $\beta_1 = \beta_2$ because β_i is the pseudo-likelihood estimator of x_i.

6.7 Appendix II

The purpose of this appendix is to give an upper bound on the rate of convergence of the ES algorithm of Table 6.8. The hypothesis is given in equation (6.36) and the notation is as in Section 6.3.2. Our approach is inspired by previous work on genetic algorithms [121] and [143].

We now present the proof of the main result. After normalization, we may assume without loss of generality that the global minimum of the function f is equal to $f_* = 0$.

Proof: (of Theorem 2) Let q_t be the Markov transition matrix associated with the chain $(\boldsymbol{\psi}^{[t]})$; i.e., $q_t(\boldsymbol{\psi}, \boldsymbol{\psi'}) = P(\boldsymbol{\psi}^{[t+1]} = \boldsymbol{\psi'} \mid \boldsymbol{\psi}^{[t]} = \boldsymbol{\psi})$. From Table 6.8, we have for any $\boldsymbol{\psi}, \boldsymbol{\psi'} \in A^m$

$$q_t(\boldsymbol{\psi}, \boldsymbol{\psi'}) = \prod_{l=1}^{m} (p_t a(\psi_{[l]}, \psi'_{[l]}) + (1 - p_t)\delta(\alpha(\boldsymbol{\psi}), \psi'_{[l]})) \tag{6.38}$$

where δ denotes the Kronecker symbol.

Given $\varepsilon > 0$, let χ_ε be the characteristic function of the event $\{f(\psi) \geq \varepsilon\}$; i.e., $\chi(\psi) = 1$ if $f(\psi) \geq \varepsilon$, and $\chi(\psi) = 0$ otherwise. Then, $P\{f(\alpha(\boldsymbol{\psi}^{[t+1]})) \geq \varepsilon\} = E[\chi_\varepsilon(\alpha(\boldsymbol{\psi}^{[t+1]}))]$. Let $\pi^{[t]}$ denote the distribution of $\boldsymbol{\psi}^{[t]}$. Since $\pi^{[t+1]}(\boldsymbol{\psi'}) = \int_{\boldsymbol{\psi}} q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi}$, we compute

$$\begin{aligned}
& E[\chi_\varepsilon(\alpha(\boldsymbol{\psi}^{[t+1]}))] = \int_{\boldsymbol{\psi'}} \chi_\varepsilon(\alpha(\boldsymbol{\psi'}))\pi^{[t+1]}(\boldsymbol{\psi'})\, d\boldsymbol{\psi'} \\
= & \int_{\boldsymbol{\psi'}} \chi_\varepsilon(\alpha(\boldsymbol{\psi'})) \int_{\boldsymbol{\psi}} q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi}\, d\boldsymbol{\psi'} \\
= & \int_{\boldsymbol{\psi}} \Big\{ \int_{\boldsymbol{\psi'}} \chi_\varepsilon(\alpha(\boldsymbol{\psi'}))q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\, d\boldsymbol{\psi'} \Big\} \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \\
= & \int_{\boldsymbol{\psi} \in E_0} \Big\{ \int_{\boldsymbol{\psi'} \in E_1} \chi_\varepsilon(\alpha(\boldsymbol{\psi'}))q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\, d\boldsymbol{\psi'} \Big\} \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \qquad (6.39) \\
+ & \int_{\boldsymbol{\psi} \in E_1} \Big\{ \int_{\boldsymbol{\psi'} \in E_1} \chi_\varepsilon(\alpha(\boldsymbol{\psi'}))q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\, d\boldsymbol{\psi'} \Big\} \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi}, \qquad (6.40)
\end{aligned}$$

where $E_0 = \{\boldsymbol{\psi} \,|\, \chi(\alpha(\boldsymbol{\psi})) = 0\}$ and $E_1 = \{\boldsymbol{\psi} \,|\, \chi(\alpha(\boldsymbol{\psi})) = 1\}$. In the first term (6.39), we have $f(\alpha(\boldsymbol{\psi})) < \varepsilon$, whereas $f(\psi'_l) \geq \varepsilon$, for any l. Thus, the equality $\psi'_l = \alpha(\boldsymbol{\psi})$ never occurs. It follows from equation (6.38) that $q_t(\boldsymbol{\psi}, \boldsymbol{\psi'}) = \prod_{l=1}^{m} p_t a(\psi_{[l]}, \psi'_{[l]})$. Thus, we obtain

$$\begin{aligned}
& \int_{\boldsymbol{\psi} \in E_0} \Big\{ \int_{\boldsymbol{\psi'} \in E_1} \chi_\varepsilon(\alpha(\boldsymbol{\psi'}))q_t(\boldsymbol{\psi}, \boldsymbol{\psi'})\, d\boldsymbol{\psi'} \Big\} \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \\
\leq & \int_{\boldsymbol{\psi} \in E_0} p_t^m (1 - P_\varepsilon)^m \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \\
\leq & \; p_t^m (1 - P_\varepsilon)^m (1 - P\{f(\alpha(\boldsymbol{\psi}^{[t]})) \geq \varepsilon\}). \qquad (6.41)
\end{aligned}$$

In the second term (6.40), we have $f(\psi'_l) \geq \varepsilon$, for any l. Thus, we obtain

$$\begin{aligned}
& \int_{\boldsymbol{\psi}\in E_1} \Big\{ \int_{\boldsymbol{\psi}'\in E_1} \chi_\varepsilon(\alpha(\boldsymbol{\psi}'))q_t(\boldsymbol{\psi},\boldsymbol{\psi}')\, d\boldsymbol{\psi}' \Big\} \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \\
\leq & \int_{\boldsymbol{\psi}\in E_1} \{p_t(1-P_\varepsilon)+(1-p_t)\}^m \pi^{[t]}(\boldsymbol{\psi})\, d\boldsymbol{\psi} \\
= & (1-P_\varepsilon p_t)^m P\{f(\alpha(\boldsymbol{\psi}^{[t]}))\geq\varepsilon\}. \qquad (6.42)
\end{aligned}$$

This completes the proof of the Theorem, upon setting $p_t = t^{-1/\tau}$. □

Finally, we prove the lemma.

Proof: (of Lemma 1) We rewrite the recursion for b_t as follows:

$$\begin{aligned}
b_t &= 1, \text{ for } t=1; \\
b_{t+1} &= \alpha(t)(1-b_t)+\beta(t)b_t, \text{ for } t\geq 1,
\end{aligned}$$

where $\alpha(t) = t^{-\frac{m}{\tau}}(1-P)^m$ and $\beta(t) = (1-Pt^{-\frac{1}{\tau}})^m$.

First of all, we claim that $0 \leq b_t \leq 1$, for all $t \geq 1$. For $t = 1$, the property holds by definition. Assume that the property holds for t; i.e., $b_t \in [0,1]$. Then, b_{t+1} is located *between* the numbers $\alpha(t)$ and $\beta(t)$. Since both of them are in the interval $[0,1]$, the same holds true for b_{t+1}.

Next, we claim that it is sufficient that $\lim_{t\to\infty} b_{2t} = 0$. Indeed, we have $b_{2t+1} \leq (2t)^{-\frac{m}{\tau}} + b_{2t}$. Thus, it follows that $\lim_{t\to\infty} b_{2t+1} = 0$. We now show that $\lim_{t\to\infty} b_{2t} = 0$.

Case 1: $\tau > 1$.

Since $\lim_{t\to\infty}\alpha(t) = 0$, and $\lim_{t\to\infty}\beta(2t) = 1$, we can take t sufficiently large so that $\alpha(t) < \beta(2t)$. Note also that the sequence $\alpha(t)$ is decreasing, whereas the sequence $\beta(t)$ is increasing.

Fix t, and consider the sequence

$$\begin{aligned}
c_k &= b_t, \text{ for } k=t; \\
c_{k+1} &= \alpha(t)(1-c_k)+\beta(2t)c_k, \text{ for } k\geq t.
\end{aligned}$$

We claim that $b_k \leq c_k$ for $k \in [t, 2t]$. For $k = t$, the property is immediate. Assume

the property true for some $k \in [t, 2t)$. We compute

$$\begin{aligned} b_{k+1} &= \alpha(k)(1-b_k) + \beta(k)b_k \le \alpha(t)(1-b_k) + \beta(2t)b_k \\ &= \alpha(t) + (\beta(2t) - \alpha(t))b_k \le \alpha(t) + (\beta(2t) - \alpha(t))c_k \\ &= c_{k+1}, \end{aligned}$$

which proves the claim. In particular, $b_{2t} \le c_{2t}$.

Now, the sequence c_k can be solved explicitly. Namely, we have

$$\begin{aligned} c_k &= \left(b_t - \frac{\alpha(t)}{1-\beta(2t)+\alpha(t)}\right)(\beta(2t)-\alpha(t))^{(k-t)} \\ &+ \frac{\alpha(t)}{1-\beta(2t)+\alpha(t)}, \text{ for } k \ge t. \end{aligned}$$

Indeed, this sequence satisfies the recursive definition of c_k. In particular, we obtain

$$b_{2t} \le \beta(2t)^t + \frac{\alpha(t)}{1-\beta(2t)+\alpha(t)},$$

upon making use of the facts that $b_t \le 1$ and $\alpha(t) \le \beta(2t)$.

One can check that $\tau > 1$ implies that $\lim_{t\to\infty} t \log \beta(2t) = -\infty$; also, $m > 1$ implies that $\lim_{t\to\infty} \frac{\alpha(t)}{1-\beta(2t)+\alpha(t)} = 0$. Therefore, the R. H. side converges to 0, and we are done. One can actually show more: the R. H. side is of the same order as $t^{-\frac{m-1}{\tau}}$, where the constant of proportionality depends on P_ε. We skip the details here.

Case 2: $\tau = 1$.

We then have $b_{k+1} \le k^{-m} + (1 - Pk^{-1})^m b_k$. Fixing t, we thus obtain $b_{t+N} \le \sum_{k=t}^{N-1} k^{-m} + \prod_{k=t}^{N-1}(1-Pk^{-1})^m b_t$. But now, $\prod_{k=t}^{\infty}(1-Pk^{-1}) = 0$ since $\sum_{k=t}^{\infty} \log(1 - Pk^{-1}) = -\infty$. Thus, $\limsup b_t \le \sum_{k=t}^{\infty} k^{-m}$. Since $m > 1$, the series $\sum_{k=1}^{\infty} k^{-m}$ converges. Thus, $\lim_{t\to\infty} \sum_{k=t}^{\infty} k^{-m} = 0$, and we are done. □

Acknowledgment

The authors are grateful to the anonymous reviewers for their comments and questions that helped them improve both the technical content and the presentation quality of this paper.

Chapitre 7

LOCALISATION DE FORMES

Les quatre premières sections de ce chapitre sont reprises en bonne partie de mon mémoire de maîtrise [44].

7.1 Introduction

L'approche suivante pour la localisation de formes dans une image a été formulée par Jain *et al.* [78]. Elle a été reprise par la suite avec quelques modifications par Jain *et al.* [79], Dubuisson *et al.* [81], Mignotte *et al.* [108, 110], Cootes *et al.* [30], et Destrempes *et al.* [44, 47, 48].

Une courbe est représentée par une suite

$$\gamma = (x_0, y_0, ..., x_m, y_m), \tag{7.1}$$

où les $m + 1$ points $(x_0, y_0), ..., (x_m, y_m)$ sont équidistants. Soit γ_0 un patron de forme, c'est-à-dire une courbe. Les auteurs considèrent des déformations γ_θ de γ_0, où θ prend ses valeurs dans un espace compact. À cet espace de déformations est associé la variable aléatoire Θ avec fonction de densité $P(\theta)$.

Ils considèrent également une variable aléatoire Y dont les valeurs sont des données observables y reliées à l'image. Il n'est pas nécessaire de spécifier la fonction de densité $P(y)$ puisque l'image est fixée. Ils se donnent une fonction de vraisemblance $P(y \mid \theta) \propto \exp(-\epsilon_l(\theta, y))$ qui tient compte des contours, textures, etc. Ils obtiennent alors la fonction de densité *a posteriori*

$$P(\theta \mid y) \propto P(y \mid \theta)P(\theta) \tag{7.2}$$

qui peut s'écrire sous la forme

$$\frac{1}{Z}\exp(-\epsilon(\theta, y)), \tag{7.3}$$

où Z est une constante de normalisation et $\epsilon(\theta, y)$. (Ne pas confondre $\epsilon(\theta, y)$ avec $\epsilon_l(\theta, y)$.) Il s'agit alors de maximiser cette fonction de densité, ce qui revient à minimiser la fonction d'énergie correspondante.

Nous présentons maintenant en détails cet approche dans le cadre des modèles de Jain *et al.* [78], Mignotte *et al.* [110] et Cootes *et al.* [30].

7.2 Modèle de Jain *et al.* (1996)

7.2.1 Déformations et distribution a priori

Les auteurs posent $\theta = (\tau_x, \tau_y, s, \psi, \xi)$ avec (τ_x, τ_y) le vecteur de translation, s le facteur d'homothétie, ψ l'angle de rotation, et $\xi = (\xi^x_{mn}, \xi^y_{mn};\ 1 \leq m \leq M,\ 1 \leq n \leq N)$, le vecteur des déformations non linéaires.

Les auteurs définissent un opérateur de déformation

$$D_\xi(x, y) = \sum_{m=1}^{M}\sum_{m=1}^{N} \frac{\xi^x_{mn} e^x_{mn}(x, y) + \xi^y_{mn} e^y_{mn}(x, y)}{\lambda_{mn}} \tag{7.4}$$

où $\lambda_{mn} = \pi^2(m^2 + n^2)$, avec

$$\begin{aligned} e^x_{mn}(x, y) &= (2\sin(\pi n x)\cos(\pi m y), 0) \\ e^y_{mn}(x, y) &= (0, 2\cos(\pi n x)\sin(\pi m y)). \end{aligned} \tag{7.5}$$

Notez que l'espace des transformations $D(x, y)$ telles que $(x, y) \mapsto (x, y) + D(x, y)$ est une transformation lisse de $[0, 1]^2$ dans lui-même, a pour base orthogonale $\{e^x_{mn}, e^y_{mn};\ m, n \geq 1\}$. Voir la figure 7.1 pour un exemple de déformation d'une forme selon ce modèle.

Si γ_0 est un patron de forme, sa déformation γ_θ est définie par

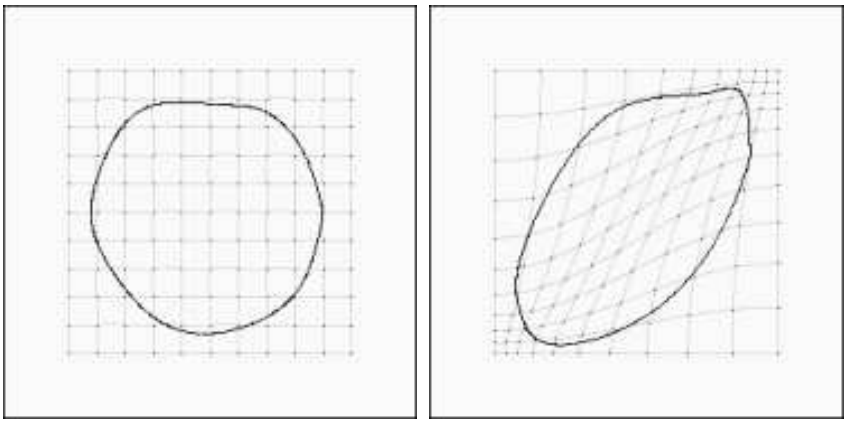


Figure 7.1. Une forme ainsi que sa déformation pour les valeurs $M = N = 1$, $\xi^x_{1,1} = \xi^y_{1,1} = 2$.

$$\gamma_\theta = T_{\tau_x,\tau_y,s,\psi}((Id + D_\xi)(\gamma_0)) \tag{7.6}$$

où

$$T_{\tau_x,\tau_y,s,\psi}(x, y) = s(x\cos(\psi) - y\sin(\psi) + \tau_x, x\sin(\psi) + y\cos(\psi)y + \tau_y). \tag{7.7}$$

Les auteurs donnent en fait une formulation légèrement différente pour γ_θ, mais qui est équivalente à toute fin pratique.

Les variables $\tau_x, \tau_y, s, \psi, \xi^x_{mn}, \xi^y_{mn}$ sont supposées indépendantes, avec τ_x, τ_y, s, ψ uniformes, et ξ^x_{mn}, ξ^y_{mn} gaussiens. D'où

$$P(\theta) = \frac{1}{Z}\exp(\frac{- \parallel \xi \parallel^2}{2\sigma^2}), \tag{7.8}$$

où Z est une constante de normalisation et σ^2 est à déterminer.

7.2.2 Vraisemblance

La fonction de vraisemblance

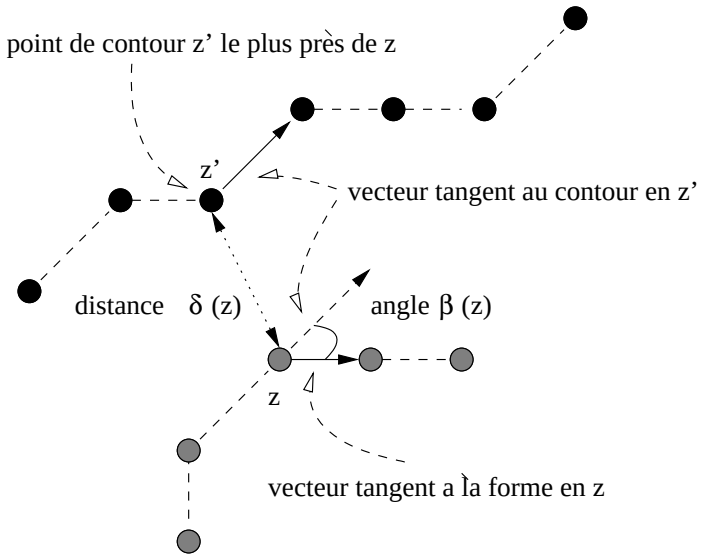


Figure 7.2. Illustration de la distance $\delta(z)$ et de l'angle $\beta(z)$.

$$P(y \mid \theta) = \frac{1}{Z} \exp(-\epsilon_l(\theta, y)) \tag{7.9}$$

est définie directement à partir de la fonction d'énergie

$$\epsilon_l(\theta, y) = \frac{1}{N_{\gamma_\theta}} \sum_{z \in \gamma_\theta} (1 - \exp(-c \parallel \delta(z) \parallel) |\cos \beta(z)|) \tag{7.10}$$

où $\delta(z)$ est la distance de z au point de contour de l'image y le plus près, et $\beta(z)$ est l'angle entre le vecteur tangent de γ_θ en z et le vecteur tangent au point de contour de y le plus près (voir Figure 7.2), et c un facteur de pondération ajusté manuellement pour chaque image traitée. Ici, le contour est obtenu à l'aide du détecteur de Canny, ce qui requiert l'ajustement manuel de seuils.

7.2.3 Optimisation

Les auteurs utilisent une procédure de multi-résolution en trois étapes avec un algorithme de descente du gradient déterministe à chaque étape. En pratique, il faut fournir une initialisation de la solution suffisamment près de la solution cherchée.

7.3 *Mignotte et al.* (2001)

Des modèles pour la localisation de formes basés sur les textures plutôt que les contours ont été proposés par Jain *et al.* [144], Liu *et al.* [98] et Mignotte *et al.* [110]. Nous ne présentons que celui de Mignotte *et al.*, car il est le seul à comporter une estimation des paramètres statistiques. Il s'agit là d'un avantage majeur, du moins en principe, puisque l'estimation de paramètres statistiques permet d'éviter l'ajustement manuel de paramètres heuristiques et donne lieu à des méthodes adaptatives aux images.

7.3.1 Déformations et distribution a priori

Il s'agit du même modèle que celui présenté à la section 7.2.1.

7.3.2 Vraisemblance

Dans le cas de la recherche du contour endocardial dans une séquence d'images échocardiographique, les auteurs considèrent deux classes e_1 et e_2 correspondant aux textures "sang" et "muscle du coeur". Étant donné un patron de forme γ_θ, la région intérieure est dénotée par $\gamma_\theta^\bullet$ et la région extérieure par γ_θ°. Les auteurs cherchent le contour pour lequel la région intérieure coincide avec la texture "sang" et la région extérieure avec la texture "muscle".

Les auteurs considèrent la fonction de vraisemblance

$$P(y \mid \theta) = \frac{1}{Z} \exp(-\epsilon_l(\theta, y)), \tag{7.11}$$

où

$$\epsilon_l(\theta, y) = -\frac{1}{N_{\gamma_\theta^\bullet}} \sum_{s \in \gamma_\theta^\bullet} \ln P(y_s \mid x_s = e_1) - \frac{1}{N_{\gamma_\theta^\circ}} \sum_{s \in \gamma_\theta^\circ} \ln P(y_s \mid x_s = e_2) \tag{7.12}$$

où s est un site de l'image, y_s est le niveau de gris du pixel au site s, et x_s est la classe correspondante.

Dans ce cas-ci, le modèle de Rayleigh est utilisé

$$P(y_s \mid x_s = e_i) = \mathcal{R}(y_s; \min_i, \alpha_i) = \frac{(y_s - \min_i)}{\alpha_i^2} \exp\left(-\frac{(y_s - \min_i)^2}{2\alpha_i^2}\right) \qquad (7.13)$$

avec $y_s > \min_i$ et $\alpha_i > 0$, où le vecteur de paramètres $(\min_i, \alpha_i)$ dépend de la classe e_i. Pour chaque classe, le vecteur des paramètres $(\min_i, \alpha_i)$ est estimé grâce à une procédure de type ECI [17, 119, 124].

7.3.3 Optimisation

Les auteurs proposent un algorithme génétique avec stratégie de préservation de l'élite ou avec stratégie hybride.

7.4 ***Modèle de Cootes** et al. **(2000)***

Le modèle de Cootes *et al.* [30] constitue une première tentative de modélisation statistique des déformations d'une forme pouvant être utilisée pour définir la distribution *a priori*.

7.4.1 Déformations et distribution a priori

Soient (τ_x, τ_y) un vecteur de translation, s un facteur d'homothétie, et ψ un angle de rotation. Les auteurs définissent

$$T_{\tau_x,\tau_y,s,\psi}(x, y) = s(x\cos(\psi) - y\sin(\psi) + \tau_x, x\sin(\psi) + y\cos(\psi)y + \tau_y). \qquad (7.14)$$

Nous supposons donné un échantillon de formes (c'est-à-dire de courbes) $\{\gamma_1, ..., \gamma_l\}$ obtenu de façon semi-automatique. Ces formes sont alignées sur un patron de forme moyen γ_0 selon l'algorithme présenté au tableau 7.1.

L'analyse en composantes principales est ensuite appliquée à l'échantillon des formes alignées dans l'espace vectoriel de dimension $d = 2(m+1)$. Soit U_q la matrice des vecteurs propres de la matrice de covariance correspondants aux q plus grandes valeurs propres. La déformation du patron de forme est définit par

$$T_\theta(\gamma_0) = T_{\tau_x,\tau_y,s,\psi}(U_q\xi + \gamma_0), \tag{7.15}$$

où $\theta = (\tau_x, \tau_y, s, \phi, \xi)$, ξ étant un vecteur dans l'espace vectoriel de dimension q. Les variables $\tau_x, \tau_y, s, \phi, \xi$ sont supposées indépendantes et τ_x, τ_y, s, ϕ uniformes. Donc, $P(\theta) \propto P(\xi)$. La fonction de densité de Ξ est estimée à l'aide d'un mélange de noyaux gaussiens. Cette méthode n'est pas optimale au sens du maximum de vraisemblance. Il faut plutôt utiliser le modèle probabiliste de réduction de la dimension de Tipping *et al.* (PPCA) [134].

7.4.2 Vraisemblance

La vraisemblance est définie par la fonction heuristique

$$\epsilon_l(\theta, y) = \sum_{z\in\gamma_\theta} \delta(z)^2 \tag{7.16}$$

où $\delta(z)$ est la distance euclédienne de z au point de contour de l'image le plus près.

7.4.3 Optimisation

Les auteurs ne travaillent pas directement avec $P(\theta|y)$. Leur approche est équivalente à considérer la fonction

$$\epsilon(\theta, y) = \epsilon_l(\theta, y) + \epsilon_a(\theta) \tag{7.17}$$

où

$$\epsilon_a(\theta) = \begin{cases} \infty & \text{si } P(\theta) < P_0 \\ 0 & \text{sinon} \end{cases} \tag{7.18}$$

avec P_0 un seuil fixé.

L'algorithme proposé (ASM) nécessite une bonne initialisation de θ. Nous en donnons la description dans [44].

7.5 Point de vue adopté dans cette thèse

Dans mon mémoire de maîtrise [44], nous avions présenté un modèle qui s'inscrit dans le cadre formel de Jain *et al.* [78]. La distribution *a priori* sur les paramètres de déformation du patron de forme était basée sur la PPCA [134]. La distribution de vraisemblance utilisait un modèle statistique des contours dans une image, maintenant publié dans [50]. L'optimisation stochastique de la distribution *a posteriori* des paramètres de la forme était effectuée à l'aide de l'algorithme E/S.

Dans cette thèse, nous ajoutons au modèle de contours, une contrainte globale basée sur une segmentation préalable de l'image au sens des couleurs selon la méthode du deuxième article (chapitre 6). Cette contrainte facilite grandement la recherche de la forme dans l'image, comme le montre nos tests. Nous avons également modifié le choix des paramètres internes de l'algorithme E/S, en utilisant le résultat théorique de la section 6.7. Finalement, nous présentons une généralisation de la PPCA à un mélange de telles distributions. Contrairement au mélange introduit dans [134], notre modèle ne comporte qu'un opérateur de reconstruction de la forme à partir des paramètres réduits. De plus, nous présentons un algorithme de type MCMC pour l'estimation du modèle proposé.

La tâche haut-niveau de localisation de formes nous permet de tester notre méthode de segmentation avec deux autres méthodes : une classification sommaire à l'aide de l'algorithme des K-moyennes, et l'algorithme du "Mean shift" [28]. Notons que nous

$\gamma_1, \gamma_2, ...\gamma_l$	échantillon de courbes comportant un même nombre de points 2D
γ_0	forme moyenne à l'itération précédente
$\bar{\gamma}$	forme moyenne à l'itération courante

But de l'algorithme: Aligner les courbes $\gamma_1, ...\gamma_l$ sur une forme moyenne $\bar{\gamma}$.

1. **Initialisation**: Ajuster chaque forme γ_i pour que le centre géométrique soit $(0, 0)$. Poser $\bar{\gamma} = \gamma_1/|\gamma_1|$, et $\gamma_0 = \bar{\gamma}$.

2. **Calcul récursif**: répéter jusqu'à convergence de $\bar{\gamma}$.

   (a) Aligner chaque γ_i avec γ_0; c'est-à-dire, trouver les valeurs de s et ψ qui minimisent $|T_{0,0,s,\psi}(\gamma_i) - \gamma_0|$ (voir l'algorithme d'alignement au tableau 7.2), et remplacer γ_i par $T_{0,0,s,\psi}(\gamma_i)$. Normaliser chaque γ_i; c'est-à-dire, remplacer γ_i par $\gamma_i/|\gamma_i|$.

   (b) Calculer le patron de forme moyen $\bar{\gamma}$ des γ_i obtenus.

   (c) Aligner $\bar{\gamma}$ avec γ_0. Normaliser $\bar{\gamma}$. Poser $\gamma_0 = \bar{\gamma}$.

Table 7.1. Algorithme pour obtenir un patron de forme moyen

aurions pu également tester des méthodes de segmentation telles que celle proposée dans [128].

x, x'	deux ensembles comportant un même nombre de points 2D dont le centre géométrique est en $(0, 0)$
s	facteur d'homothétie
ψ	angle de rotation (en radians)

But de l'algorithme: Trouver les valeurs de s et ψ qui minimisent $|T_{0,0,s,\psi}(x) - x'|$.

1. Calculer $a = (\sum_{i=0}^{m}(x_i x'_i + y_i y'_i))/|x|^2$ et $b = (\sum_{i=0}^{m}(x_i y'_i - y_i x'_i))/|x|^2$, où $x = (x_0, y_0, ..., x_m, y_m)$ et $x' = (x'_0, y'_0, ..., x'_m, y'_m)$.

2. Poser $s = \sqrt{a^2 + b^2}$ et $\psi = \tan^{-1}(b/a)$.

Table 7.2. Algorithme d'alignement

Chapitre 8

LOCALIZATION OF SHAPES USING STATISTICAL MODELS AND STOCHASTIC OPTIMIZATION

Nous avons soumis la première version de cet article comme l'indique la référence bibliographique

> François Destrempes, Max Mignotte, et Jean-François Angers, "Localization of Shapes using Statistical Models and Stochastic Optimization", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, soumis, mars 2006.

Nous présentons dans la version finale de cette thèse une première version corrigée de l'article.

Abstract

In this paper, we present a new Bayesian model for deformations of shapes. The likelihood is based on the statistical distribution of the gradient vector field of the gray level. The prior distribution is based on the Probabilistic Principal Component Analysis (PPCA). We also propose a new model based on mixtures of PPCA that is useful in the case of greater variability in the shape. A criterion of global or local object specificity based on a preliminary color segmentation of the image, is included into the model. The localization of a shape in an image is then viewed as minimizing the corresponding Gibbs field. We use the Exploration/Selection (E/S) stochastic algorithm in order to find the optimal deformation. This yields a new unsupervised statistical method for localization of shapes. In order to estimate the statistical parameters for the gradient vector field of the gray level, we use an Iterative

Conditional Estimation (ICE) procedure. The color segmentation of the image can be computed with an Exploration/Selection/Estimation (ESE) procedure.

8.1 Introduction

Localization of shapes is an important problem in Image Processing with applications to segmentation and classification [81, 108], pattern recognition [79], motion tracking [91], and 3-D reconstruction [8, 9].

This paragraph is a short summary of the literature review presented in [78]. Localization of a shape consists of matching a deformable template to the data in the image. In the free-form formulation, the shape of the template is constrained only by local geometric criteria, such as smoothness and continuity. The elastic deformable model [18, 114], the active contour model [84], and the models presented in [56] and [132] are examples of that approach. In the parametric formulation, the prior global information of the shape is used to model the deformations. In that approach, one can represent the template either by 1) a family of parametrized curves or by 2) a prototype template together with its parametric space of deformations, as formulated in the pattern theory of [68]. The models of [23, 130] follow the first scheme, whereas [31–34, 78, 79, 81, 108, 110] are examples of the second scheme.

An important idea exploited in [56, 78, 110, 132], is the formulation of the localization problem in a Bayesian framework. Namely, one searches for the optimal deformation of the shape in the sense of the Maximum A Posteriori (MAP). Nevertheless, in the probabilistic model presented in [78], the likelihood is defined in an *ad hoc* manner: it depends on a smoothing parameter ρ, which is not estimated from the image. Moreover, it is based on a binary detection of contours, so that the localization step might be penalized by flaws occurring at the edge-detection step. In contrast, the likelihood functions presented in [56, 110, 132] are based on the distribution of the gray level. However in those models, the *prior* distribution is not learned

from a sample of shapes. In the active shape model [31], the prior distribution of the deformations of the shape is learned from a sample of curves representing the same shape. The classical Principal Component Analysis (PCA) is used to reduce the dimensionality of the deformation space and the distribution of the reduced parameters is modeled by a mixture of Gaussian distributions. In [33, 34], a statistical shape prior is learned form a sample set, and is incorporated into a Mumford-Shah segmentation process. Models based on textures have been introduced in [98, 144].

In [47], we presented a Bayesian model for localization of shapes based solely on contours. A shape is viewed as a deformable template and the distribution of the deformation parameters is modeled with the Probabilistic Principal Component Analysis (PPCA) [134] obtained from a training set aligned as in [31]. This gives the prior distribution of deformations of a shape. The likelihood of deformations of a given shape is based on the statistical distribution of the gradient vector field of the gray level [50] (similar to [44, 46]). We use an Iterative Conditional Estimation (ICE) procedure [119, 124] for the estimation of the parameters, as explained in [50]. Since the likelihood and the *prior* distribution are endowed with a true statistical meaning, this model allows proper Bayesian inference. Localization of a shape can then be viewed as minimizing the Gibbs field corresponding to the posterior distribution. The E/S optimization algorithm [59] is then used to find the optimal solution. Note that this method is based on a probabilistic detection of contours, and hence, should be more robust to errors occurring at the contour detection step.

Yet, due to its complexity, the Gibbs field of the posterior distribution is rather hard to optimize directly, even if the optimization algorithm converges asymptotically. Thus, one has to use an initialization procedure to explore plausible regions of the deformation space in order to speed up convergence to a solution. In [47], we optimized a heuristic potential function similar to [108] for the initialization procedure. In [48], we used a clustering of contour segments in order to find initial positions of the shape. In [44], we used for the initialization procedure a clustering of the shapes.

None of those methods turned out to be convincing. In fact, the main problem lies in fitting a 1-dimensional object (the shape) with a 1-dimensional subset of the image; i.e. the contours.

In this paper, we present an extension of the above model that takes into account a segmentation of the image into regions. Namely, we require that the region labels inside and outside of the shape be distinct. We call this property the *object specificity*. This simple idea turns out to make the model a lot easier to estimate, as the tests reported in this paper indicate. Indeed, the task now consists of fitting a 2-dimensional object (the interior of the shape) with a 2-dimensional subset of the image; i.e. the color regions forming the object. In particular, we do not need anymore the (complicated and not so successful) initialization procedures of [44, 47, 48]. We base the image segmentation on the model presented in [45]. A procedure called the ESE procedure [51] is used to compute the segmentation, as explained with full details in [45]. We also make comparison with a few other segmentation models. We also propose a *local object specificity* property that is useful in the case of multiple occurence of the object in the image. Furthermore, we include a generalization of the PPCA that uses mixture of Gaussian kernels. This is a new model for reduction of dimensionality, that offers more flexibility in the case of a shape with greater variability.

Altogether, the method proposed here for localizing shapes is organized in four steps:

1. The first step consists of learning off-line the deformations of the shape from a sample of curves. This means learning the mean shape, the non-linear deformations of the shape, and the prior distribution of deformations of the shape. This training step is based on the PPCA [134] in its simplest form, or on the new model for reduction of dimensionality presented in Section 8.5.

2. The second step consists of estimating on-line the statistical distribution of the

gradient vector field of the gray level from the observed data in a given image, using an ICE procedure.

3. The third step consists of segmenting on-line the image based on colors using the ESE procedure.

4. The fourth step consists of localizing the shape in the image, by minimizing with the E/S algorithm [59] the Gibbs field of the posterior distribution of deformations of the shape, with the global constraint of (local) object specificity taken into account.

In our opinion, the main contribution of this paper is to bring together interesting ideas of previous work with the novelty of a new model for deformations of shapes and new global constraints for the localization.

The remaining part of this paper is organized as follows. In Section 8.2, we recall the statistical model [50] for the gradient vector field of the gray level, as well as the statistical model of the colors [45]. Section 8.3 presents the statistical model for deformations of a shape, with a brief review of the PPCA. The stochastic method for localization of a shape is explained in Section 8.4. The new model for reduction of dimensionality is presented in Section 8.5. The local object specificity property is presented in Section 8.6. In Section 8.7, we discuss experimental results. We conclude in Section 8.8.

8.2 Statistical models for the image

8.2.1 Statistical model for the gradient vector field

In this section, we recall the model introduced in [50] (similar to what we have proposed in [44, 46]) for the distribution of the gradient of the gray level in an image. It gives much more than a binary detection of contours (i.e., a classification of pixels

as being on or off contours): it also gives for each pixel its likelihood of being on contours and its likelihood of being off contours. The likelihood of deformations of a shape presented in Section 8.3.4 is based on this model.

Given an image of size $M \times N$, $G = (V_G, E_G)$ will denote the non-oriented graph consisting of the $M \times N$ pixels of the image together with the edges given by the usual 8-neighbors. If s and t are adjacent sites of G, we denote the edge joining s and t by (s,t) (so, $(s,t) \in E_G$).

For each $s \in V_G$, the random variable Y_s represents the norm of the gradient of the gray level at the corresponding pixel. If $(s,t) \in E_G$, the random variable Y_{st} represents the absolute value of the angle between the mean of the gradient at s and t, and the normal to the vector from s to t. This angle is normalized between $-\pi/2$ and $\pi/2$ before taking its absolute value. Finally, for each $s \in V_G$ and $(s,t) \in E_G$, the random variables Z_s and Z_{st} take their values in the set $\{e_1, e_2\}$, where e_1 denotes the class "off contours" and e_2 the class "on contours".

As in [50], we adopt the following distributions for the marginals of Y_s and Y_{st} on the classes e_1 and e_2:

- $P(y_s \mid z_s = e_1, C, \alpha)$ is a shifted Weibull law [106] $\mathcal{W}(y_s; \min, C, \alpha)$, where $y_s > \min = -0.001$.

- $P(y_s \mid z_s = e_2, w_j, \mu_j, \sigma_j)$ is a mixture $\mathcal{M}(y_s; w_j, \mu_j, \sigma_j) = \sum_{j=1}^{3} w_j \mathcal{N}(y_s; \mu_j, \sigma_j)$ of three Gaussian kernels truncated on the interval $(0, \infty)$.

- $P(y_{st} \mid z_{st} = e_1)$ is a uniform distribution $\mathcal{U}(y_{st}; 0, \frac{\pi}{2})$ on $[0, \frac{\pi}{2}]$.

- $P(y_{st} \mid z_{st} = e_2, \alpha_0)$ is a truncated exponential law $k_0 \mathcal{E}(y_{st}; \alpha_0)$ on the interval $[0, \frac{\pi}{2}]$, with $k_0 = \{1 - \exp(-\frac{\pi}{2\alpha_0})\}^{-1}$.

See Fig. 8.1 for an example of estimated distributions and empirical distributions. We use an ICE procedure [119], [124], as explained in [50] in order to estimate the

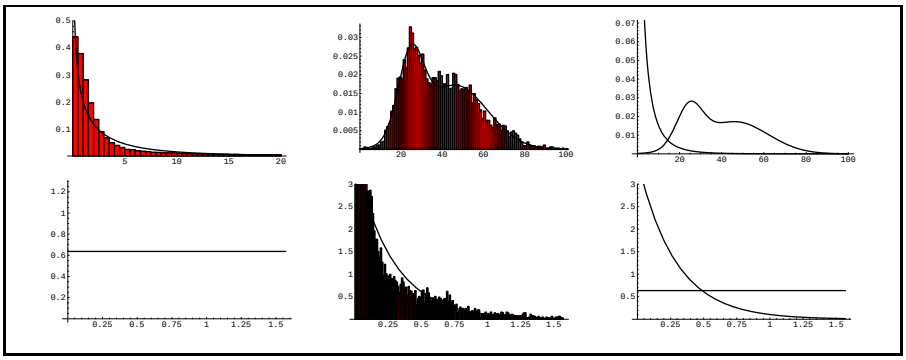


Figure 8.1. Top: Example of distributions for the norm Y_s of the gradient of the gray level for image (11) of Fig. 8.8. From left to right: norm of the gradient for points off contours; norm of the gradient for points on contours; comparison between the two distributions. Bottom: Example of distributions for the angle Y_{st} of the gradient of the gray level for image (11) of Figure 8.8. From left to right: angle for edges off contours; angle for edges on contours; comparison between the two distributions.

hyper-parameters C, α, w_j, μ_j, σ_j, and α_0 for a given image.

8.2.2 Statistical model for the field of colors

In this section, we recall the model introduced in [45] for the distribution of the colors in an image.

For each $s \in V_G$, the random variable W_s represents the Luv components at the corresponding pixel. For each $s \in V_G$, the random variable X_s takes its values in a finite set of K region labels $\{f_1, f_2, ..., f_K\}$.

We adopt the following distributions for the marginals of W_s on a region class f_k:

- $P(w_s \mid x_s = f_k, \pi_{(i,k)}, \tilde{\mu}_i, \tilde{\Sigma}_i)$ is a mixture of K_1 Gaussian kernels $\sum_{i=1}^{K_1} \pi_{(i,k)}$ $\mathcal{N}(w_s; \tilde{\mu}_i, \tilde{\Sigma}_i)$, where $\sum_i^{K_1} \pi_{(i,k)} = 1$ and $\pi_{(i,k)} \geq 0$.

We use the ESE procedure [51], as explained in [45], in order to estimate the parameters and obtain a segmentation x for a given image. Here, we take $K = 30$

region labels, and $K_1 = 30$ Gaussian kernels. The initial number of labels is K and the ESE procedure allows this number to be decreased. See Fig. 8.2 for an example of segmentation. Note that this is an unsupervised method.

8.3 Statistical model for deformations of a shape

In this section, we present a statistical model for deformations of a shape. The training phase and the prior distribution are based on the PPCA [134]. The likelihood is based on the statistical model of the gradient vector field of the gray level presented in Section 8.2.1. Thus, the prior distribution can be learned from a sample of curves representing the shape, and the likelihood can be estimated from the observable data given by the image. The global constraint is based on the image segmentation method presented in Section 8.2.2. The material presented in Sections 8.3.1, 8.3.2, 8.3.3, and 8.3.4 first appeared in [47]. Sections 8.3.1, 8.3.2, 8.3.3 were generalized to the 3-dimensional case int the context of object reconstruction in [8]. Sections 8.3.1 and 8.3.3 are extended to a more flexible model in Section 8.5.

8.3.1 Training phase

A curve is represented as a template, that is a sequence of points

$$\gamma = (a_0, b_0, a_1, b_1, ..., a_m, b_m) \tag{8.1}$$

where the $m + 1$ points $(a_0, b_0), (a_1, b_1), ..., (a_m, b_m)$ are equally spaced on the curve between certain key-points (see Fig. 8.3 for an example). Those key-points are only used at the training phase. Although not essential, they are useful in the semi-automatic edition of the templates corresponding to a database of the shape. All steps of the localization procedure itself are automatic and do not use those key-points.

Given a sample $\gamma_1, ..., \gamma_N$ of curves with same number of points and representing the deformations of a same shape, we resort to the procedure [31] to align this training

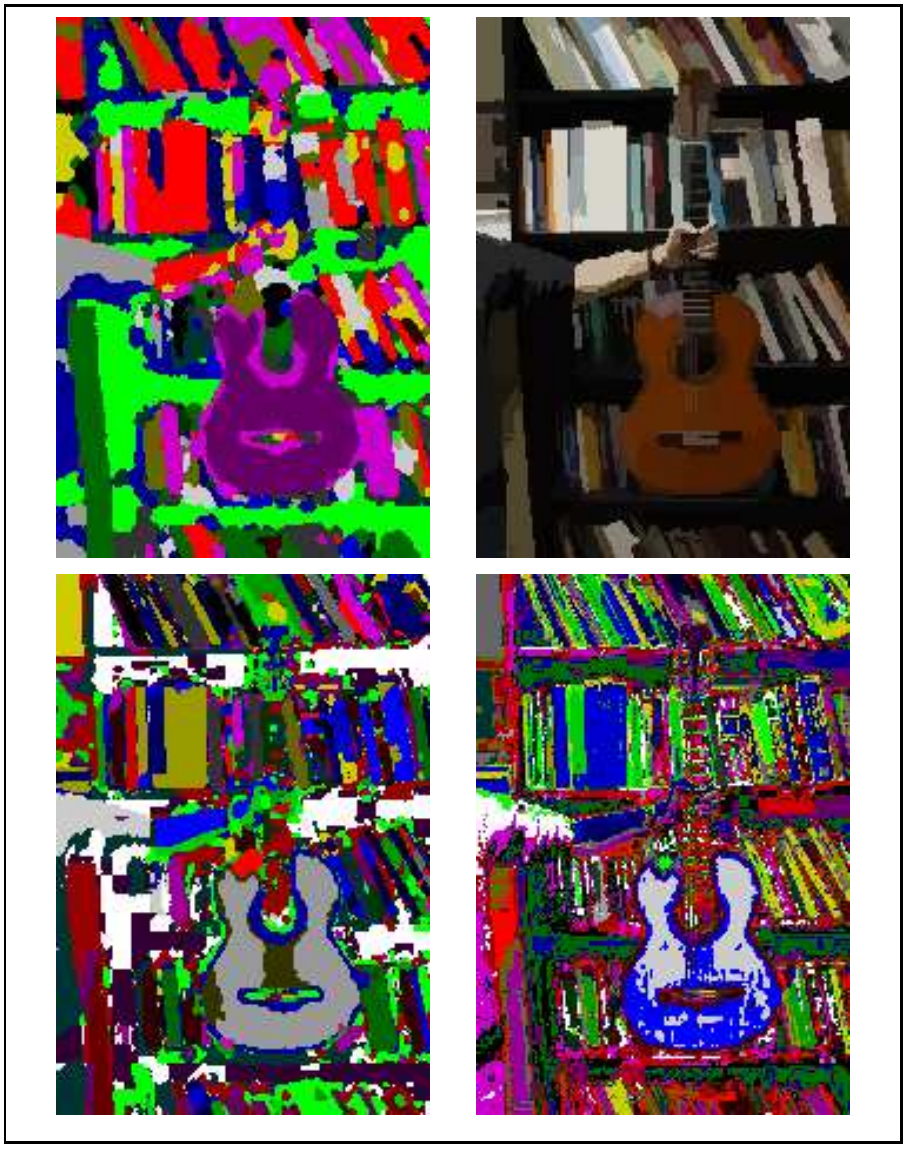


Figure 8.2. Segmentations of image (25) of Fig. 8.8, based on color models. From top to bottom and left to right: (1) the color model of Section 8.2.2; (2) the Mean Shift algorithm; (3) the color model of [51]; (4) the K-means algorithm (with $K = 30$).

Figure 8.3. Key-points on the curve.

set on a mean shape. Viewing $T = (A_0, B_0, ..., A_m, B_m)$ as a random vector, and setting $d = 2m + 2$, we consider the Probabilistic Principal Component Analysis (PPCA) model [134]: $T = W\Xi + \nu + \varepsilon$, with W a $d \times q$ matrix, ν a vector of dimension 2, $\Xi \sim \mathcal{N}(0, I_q)$, and $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_{2d})$. If t_1,,...,t_N is an i.i.d. sample of the variable T, we obtain an optimal model [134] in the sense of the ML upon taking

$$\begin{aligned} \sigma^2 &= \frac{1}{2d-q}\sum_{i=q+1}^{d}\lambda_i, \\ \nu &= \bar{t} = \frac{1}{N}\sum_{i=1}^{N} t_i, \\ W &= U_q(\Lambda_q - \sigma^2 I_q)^{1/2}, \end{aligned} \tag{8.2}$$

where $\lambda_1, ..., \lambda_d$ are the eigenvalues of the sample covariance matrix, in decreasing order, Λ_q is the diagonal matrix with entries $\lambda_1, ..., \lambda_q$, and U_q is the $d \times q$ matrix with columns equal to the corresponding eigenvectors, normalized so that they have euclidean norm equal to 1 (i.e., the columns of U_q span the principal subspace of the sample covariance matrix). Note that T has distribution function

$$p_T(t) = \mathcal{N}(t; \nu, \sigma^2 I_d + WW^t). \tag{8.3}$$

The corresponding reconstruction operator is

$$\gamma_{ppca}(\xi) = U_q(\Lambda_q - \sigma^2 I_d)^{1/2}\xi + \nu. \tag{8.4}$$

8.3.2 Deformation parameters

From the training phase, we obtain the mean shape $\nu = \bar{t}$, and its non-linear deformations $\gamma_{ppca}(\xi)$, where ξ is a vector in the parameter space Ξ of dimension q. In addition, we consider projective deformations of the form $(x', y') = (\cos(\psi_y)x + \sin(\psi_y)\sin(\psi_x)y, \cos(\psi_x)y)$ applied point-wise to a curve. Such a transformation is obtained by applying to a point $(x, y, 0)$ a rotation by an angle ψ_x around the x-axis, then a rotation by an angle ψ_y around the y-axis, and finally projecting the resulting point onto the xy-plane. We also consider rigid deformations given by scaling s, rotation ψ, and translation (τ_x, τ_y) applied point-wise to a curve lying in the xy-plane. This yields a vector of deformation

$$\theta = (\tau_x, \tau_y, s, \psi, \psi_y, \psi_x, \xi_1, ..., \xi_q) \tag{8.5}$$

of dimension $q + 6$. The transformation θ is computed in the following order: first calculate the non-linear deformation $\gamma_{ppca}(\xi)$; next, apply the projective transformation (ψ_y, ψ_x); then, apply s and ψ; finally, translate by τ. The resulting template is denoted γ_θ.

8.3.3 Prior distribution

Let Θ be the random variable corresponding to the vector of deformations. We model the distribution of Θ by

$$P(\theta) = \mathcal{U}(\tau_x, \tau_y, s, \psi, \psi_y, \psi_x)p_T(\gamma_{ppca}(\xi)) \tag{8.6}$$

where $\mathcal{U}$ denotes the uniform distribution and T represents the full data. From the Lemma of [47], we obtain

$$p_T(\gamma_{ppca}(\xi)) \propto \exp\big(-\frac{1}{2}\xi^t(I_q - \sigma^2\Lambda_q^{-1})\xi\big). \tag{8.7}$$

8.3.4 Likelihood

Let Y be the random field $\{Y_s, Y_{st}\}$, where the random variables Y_s, Y_{st} are as in Section 8.2.1. Given an image, y is the observed realization of Y. Given a path $c = (s_0, ..., s_m)$ in the graph G of Section 8.2.1, let $E(c)$ denote the set of edges $\{(s_{i-1}, s_i) \,:\, i = 1, 2, ..., m\}$. We consider as we have proposed in [44] the likelihood $P(y \mid c)$ given by

$$\begin{aligned}
& \prod_{s \notin c} P(y_s \mid z_s = e_1) \prod_{s \in c} P(y_s \mid z_s = e_2) \\
\times \quad & \prod_{(s,t) \notin E(c)} P(y_{st} \mid z_{st} = e_1) \prod_{(s,t) \in E(c)} P(y_{st} \mid z_{st} = e_2) \\
= \quad & k(y) \prod_{s \in c} \frac{P(y_s \mid z_s = e_2)}{P(y_s \mid z_s = e_1)} \prod_{(s,t) \in E(c)} \frac{P(y_{st} \mid z_{st} = e_2)}{P(y_{st} \mid z_{st} = e_1)} \\
= \quad & k(y) \prod_{i=0}^{m} \frac{\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j)}{\mathcal{W}(y_{s_i}; \min, C, \alpha)} \prod_{i=1}^{m} \frac{k_0 \mathcal{E}(y_{s_{i-1}, s_i}; \alpha_0)}{\mathcal{U}(y_{s_{i-1}, s_i}; 0, \frac{\pi}{2})}
\end{aligned} \tag{8.8}$$

where

$$k(y) = \prod_{s \in V_G} P(y_s \mid z_s = e_1) \prod_{(s,t) \in E_G} P(y_{st} \mid z_{st} = e_1). \tag{8.9}$$

The distributions $\mathcal{W}(y_s; \min, C, \alpha)$, $\mathcal{M}(y_s; w_j, \mu_j, \sigma_j)$, $\mathcal{U}(y_{st}; 0, \frac{\pi}{2})$, and $k_0\mathcal{E}(y_{st}; \alpha_0)$ are as in Section 8.2.1. The map of $\log(P(y_s \mid z_s = e_1)/P(y_s \mid z_s = e_2))$ is presented in Fig. 8.4.

Note that the scaling factor $k(y)$ depends only on the observed data (and not on the curve), and that the principal factor

$$\prod_{i=0}^{n} \frac{\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j)}{\mathcal{W}(y_{s_i}; \min, C, \alpha)} \prod_{i=1}^{n} \frac{k_0 \mathcal{E}(y_{s_{i-1}, s_i}; \alpha_0)}{\mathcal{U}(y_{s_{i-1}, s_i}; 0, \frac{\pi}{2})} \tag{8.10}$$

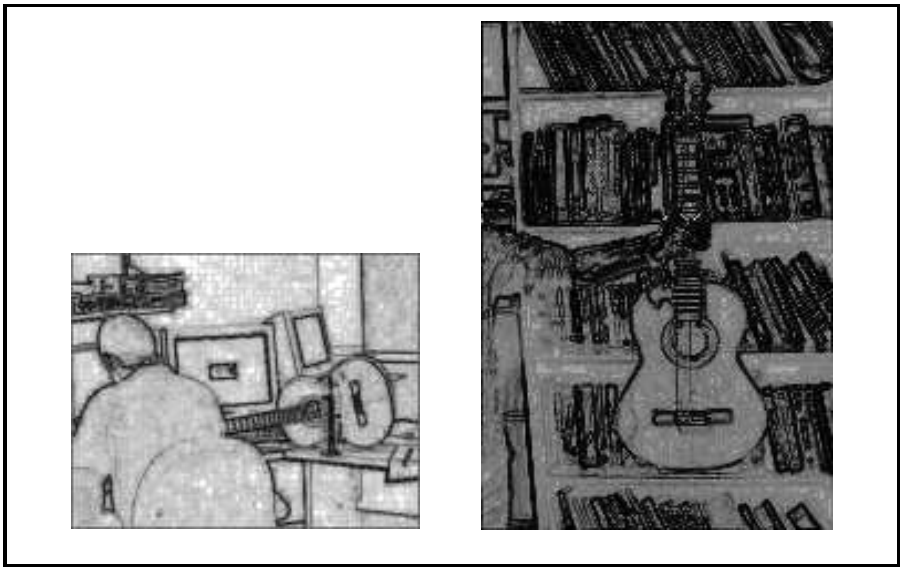


Figure 8.4. Representation of the function $\log(P(y_s|z_s = e_1)/P(y_s|z_s = e_2))$ **for images (11) and (25) of Fig. 8.8.**

is invariant under affine transformations of the gray levels.

We can then define the likelihood of a deformation θ by

$$P(y \mid \theta) = P(y \mid c_\theta) \tag{8.11}$$

where c_θ is obtained by interpolation of the polygonal curve with vertices given by the points of γ_θ.

8.3.5 Likelihood of the augmented data

Let $x = (x_s)$ be a classification of the pixels in the image into K equivalence classes according to the colors as in section 8.2.2, with $x_s \in \Lambda = \{f_1, ..., f_K\}$. We consider the global constraint

$$V(x,\theta) = \sum_{k=1}^{K} p_k(\theta) |\{s \,:\, x_s = f_k, s \notin c_\theta^{int}\}|, \tag{8.12}$$

where $p_k(\theta)$ is the proportion of the points in the interior c_θ^{int} of the curve having label f_k. Note that the second factor is equal to $(1 - p_k(\theta))|\{s \,:\, x_s = f_k\}|$. So, only $p_k(\theta)$ needs to be computed dynamically, whereas $|\{s \,:\, x_s = f_k\}|$ can be pre-computed. Thus, $V(x,\theta)$ is minimal whenever the region labels inside an object are specific to that object; i.e., the region labels inside an object do not occur outside that object, and vice-versa. Thus, we are working under the following hypothesis:

Object specificity: the color labels inside and outside the object are distinct. (8.13)

In Section 8.6, a variant of this global constraint is presented in the context of multiple occurence of the object.

We consider the couple of random fields (Y, X). The hidden discrete random field X is not observable, but it can nevertheless be deduced from the observable random field of colors W, for instance as in Section 8.2.2. We define the likelihood of the augmented data by

$$P(y, x \mid \theta) \propto P(y \mid \theta) e^{-V(x,\theta)}. \tag{8.14}$$

8.3.6 Posterior distribution

We deduce by Bayesian inference the posterior distribution of a deformation conditional to (y, x)

$$P(\theta \mid y, x) \propto P(y \mid \theta) e^{-V(x,\theta)} P(\theta). \tag{8.15}$$

Writing $c_\theta = (s_0, s_1, ..., s_m)$, let $V_1(\theta, y, x)$ be the Gibbs field defined by

$$
\begin{aligned}
& \sum_{i=0}^{m} \big(\log(\mathcal{W}(y_{s_i}; \min, C, \alpha) - \log(\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j))\big) \\
+ & \sum_{i=1}^{m} \big(\log(\mathcal{U}(y_{s_{i-1},s_i}; 0, \frac{\pi}{2}) - \log(k_0\, \mathcal{E}(y_{s_{i-1},s_i}; \alpha_0))\big) \\
+ & \sum_{k=1}^{K} p_k(\theta) |\{s \,:\, x_s = f_k, s \notin c_\theta^{int}\}| \\
+ & \frac{1}{2}\xi^t (I_q - \sigma^2 \Lambda_q^{-1})\xi. \qquad (8.16)
\end{aligned}
$$

Then, the posterior distribution is given by

$$P(\theta \mid y, x) \propto \exp(-V_1(\theta, y, x)) \qquad (8.17)$$

where the factor depends only on the image, and not on the deformation of the shape.

8.4 Stochastic localization of a shape

We view similarly to [56, 78, 110, 132], the localization of a shape c_0 in an image as finding its deformation θ that maximizes the posterior distribution $P(\theta \mid y, x)$. So, in order to localize a shape in an image, we want to minimize the Gibbs field $V_1(\theta, y, x)$ of Section 8.3.6, as a function of θ. Our assumption is that an optimal solution for the function V_1 is the desired deformation of the shape. This will be the case, as show our tests, under the hypothesis of object specificity (c.f. (8.13) of Section 8.3.5).

In order to solve the optimization problem, we resort to the stochastic algorithm [59] for which the asymptotic convergence has been proved. In this paper, we use the version presented in Table 8.1. Its asymptotic convergence follows from [51]. We had already used the ES algorithm in the context of localization of shapes in [44, 47, 48], and in the context of 3D-reconstruction in [8]. The main point here, is that the use of the global constraint of Section 8.3.5 makes the algorithm converge a lot more rapidly

E	a finite discrete subset (using the ε-machine) of the Cartesian product of Section 8.4: $[0, M-1] \times [0, N-1] \times [\rho_1 d, \rho_2 d] \times [0, 2\pi] \times [-\psi_0, \psi_0]^2 \times [-1, 1]^q$;
l	number of coordinates (here $l = q + 6$);
V_1	the Gibbs field of equation (8.16) defined on E;
r	a real number in the interval $(0, 1)$ called the radius of exploration with r greater than the ε-machine (here, $r = \frac{1}{8}$);
D	the diameter of exploration ($D = \frac{1}{2r}$);
τ	the initial temperature (here, $\tau = 15$);
n	an integer greater than 1 (here, $n = 30$);
$\boldsymbol{\theta}$	an element $(\theta_1, ..., \theta_n)$ of E^n, called a population;
k	the current iteration;
p_k	the probability of exploration.

Goal: to minimize V_1 on E.

Table 8.1. Version of the E/S algorithm used in Section 8.4 (part 1)

than before. In particular, we do not need anymore the (complicated) initialization procedures proposed in [44, 47, 48].

Given an image of dimension $M \times N$, we consider the function V_1 of equation (8.16) on a domain of the form

$$\begin{aligned} (\tau_x, \tau_y, s, \psi, \psi_x, \psi_y) &\in [0, M-1] \times [0, N-1] \times [\rho_1 d, \rho_2 d] \\ &\quad \times [0, 2\pi] \times [-\psi_0, \psi_0]^2 \\ \xi = (\xi_1, ..., \xi_q) &\in [-1, 1]^q, \end{aligned} \tag{8.18}$$

Initialization: Initialize randomly $\theta_i^{[0]}$, for $i = 1, ..., n$. Set $k = 1$.

while $k \leq 200$ **do**

Update $k \leftarrow k + 1$.

Determine the best current solution $\alpha(\boldsymbol{\theta}^{[k]}) = \theta_i^{[k]}$, where i is defined by $V_1(\theta_j^{[k]}) > V_1(\theta_i^{[k]})$ for $j < i$, and $V_1(\theta_j^{[k]}) \geq V_1(\theta_i^{[k]})$ for $j > i$.

for $i = 1, 2, ..., n$ **do**

Let u be a random number between 0 and 1.

if $u \leq p_k = k^{-\frac{1}{\tau}}$ **then**

Exploration: draw m according to the binomial distribution $b(D - 1, \frac{1}{D-1})$. Set $\theta = \theta_i^{[k+1]} = \theta_i^{[k]}$.

for $m + 1$ times **do**

Set $\theta = \theta_i^{[k+1]}$.

Replace with probability $\frac{1}{l}$ each of the l coordinates of θ by a random number within a distance of r to the coordinate.

With probability $\frac{1}{2}$, add to the fourth coordinate (representing the angle of rotation) a random multiple of $\frac{\pi}{4}$. Let $\theta_i^{[k+1]}$ be the resulting solution.

end for

else

Selection: Set $\theta_i^{[k+1]} = \alpha(\boldsymbol{\theta}^{[k]})$.

end if

end for

end while

Table 8.2. Version of the E/S algorithm used in Section 8.4 (part 2)

where $d = \sqrt{M^2 + N^2}$ is the diameter of the image, $0 < \rho_1 < \rho_2 \leq 0.5$, and $0 \leq \psi_0 < \frac{\pi}{2}$. The choice for the interval associated to the scalar factor s assumes implicitly an *a priori* on the dimension of the shape. This is reasonable since in applications of localization of shapes, one usually knows the relative size of the shape (for instance, an anatomic object in a medical image). We experimented with $\rho_1 = 0.25$ and $\rho_2 = 0.5$ (see Figure 8.7). This amounts to assuming that the diameter of the object is at least half the diameter of the image. Also, the choice for the interval of ψ_x and ψ_y is guided by the restriction on projective deformations that is allowed for the shape. In our tests, $\psi_0 = \frac{\pi}{8}$.

Note that it seems preferable in practice to do a few trials with different seeds, rather than increase the number of iterations with a same seed. In our tests, 20 stochastic optimizations are performed with different seeds. Each search is limited to 436 iterations. The parameters of the E/S algorithm are set equal to $r = \frac{1}{8}$, $\tau = 15$ and $n = 30$ in Table 8.1.

8.5 General model for deformations of a shape

In this section, we present a statistical model for deformations of a shape that extends the PPCA of Section 8.3. The main point is to replace a single Gaussian distribution by a more flexible mixture of Gaussian kernels.

8.5.1 Modified training phase

Let $T = (T_1, ..., T_d) = (A_0, B_0, ..., A_m, B_m)$ be the random vector of a template as in equation (8.1). We consider the model: $T = W\Xi + \nu + \varepsilon$, with W a $d \times q$ orthonormal matrix, ν a vector of dimension d, $\Xi \sim \sum_{i=1}^{\ell} \pi_i \mathcal{N}(\zeta_i, \Sigma_i)$, and $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_d)$. Note that, unlike the PPCA, we require that W be orthonormal (i.e., $W^t W = I_q$). But then, the distribution of Ξ is allowed to be any mixture of Gaussian kernels. One can show that T has distribution

$$p_T(t) = \sum_{i=1}^{\ell} \pi_i \mathcal{N}(t; W\zeta_i + \nu, \sigma^2 I_d + W\Sigma_i W^t). \tag{8.19}$$

Now, let $t_1, ..., t_N$ be an i.i.d. sample of the variable T. We consider the Principal Component Analysis (PCA) reconstruction operator

$$\gamma_{pca}(\xi) = U_q \xi + \nu, \tag{8.20}$$

where U_q and $\nu = \bar{t}$ are as in equation (8.2). Thus, we now take $W = U_q$ instead of $U_q(\Lambda_q - \sigma^2 I_d)^{1/2}$. That choice minimizes the reconstruction error. The reduced dimension q is chosen so that the minimal reconstruction error is smaller than a certain value.

Our goal is then to estimate the vector of parameters $\chi = (\pi_i, \zeta_i, \Sigma_i, \sigma^2)$. For instance, in the case of $\ell = 1$, the estimation of the parameters in the sense of the ML is given by $\zeta_1 = 0$, $\Sigma_1 = \Lambda_q - \sigma^2 I_q$, and $\sigma^2 = \frac{1}{(d-q)} \sum_{i=q+1}^{d} \lambda_i$, as follows from equation (8.2). Thus, in that case, we recover the PPCA. Note that the proposed mixture model differs from the mixture of PPCA [134] used in [8]. In our case, there is only one reconstruction operator (W, ν), whereas in [134], the reconstruction operator varies with the kernel.

Under the Bayesian paradigm, we set the following usual priors [2] on the parameters π_i, ζ_i, Σ_i, and σ^2.

1. A Dirichlet prior on the mixture parameters π_i:

$$(\pi_1, ..., \pi_\ell) \sim \mathcal{D}(A_0; \alpha_1, ..., \alpha_\ell) \tag{8.21}$$

   where $A_0 = \ell$, and $\alpha_1 = ... = \alpha_\ell = \frac{1}{\ell}$.

2. An inverted Wishart prior on the covariance matrix Σ_i:

$$\Sigma_i \sim \mathcal{IW}(\Lambda_0, d_0) \tag{8.22}$$

where Λ_0 and d_0 are as in Table 8.5.

3. A Gaussian conditional prior on the mean ζ_i:

$$\zeta_i \mid \Sigma_i \sim \mathcal{N}(\zeta_0, \frac{1}{k_0}\Sigma_i) \tag{8.23}$$

where ζ_0 and $k_0 = 0.01$ are as in Table 8.5.

4. A non-informative improper prior on σ^{-2}:

$$\sigma^{-2} \sim 1. \tag{8.24}$$

We are then interested in the mean estimator of χ:

$$\hat{\chi} = \int_{\chi} \chi \, P(\chi \mid t_1, ..., t_N) \, d\chi. \tag{8.25}$$

We compute an approximate value of the estimator using a Monte Carlo Markov Chain (MCMC) algorithm. In order to do so, we consider a latent discrete variable C taking its values in the set of hidden labels $\{g_1, ..., g_\ell\}$ indicating the Gaussian kernel. Furthermore, we consider Ξ as a latent continuous variable. This is equivalent to the following distributions:

$$\begin{aligned} t \mid \xi, \sigma^2 &\sim \mathcal{N}(W\xi + \nu; \sigma^2 I_d) \\ \xi \mid c = g_i &\sim \mathcal{N}(\zeta_i, \Sigma_i) \\ c &\sim \mathcal{M}(\pi_1, ..., \pi_\ell), \end{aligned} \tag{8.26}$$

where $\mathcal{M}$ denotes the multinomial distribution. We deduce the distributions:

$$
\begin{aligned}
&\xi \mid t, c = g_i, \sigma^2 \sim \mathcal{N}(\tilde{\zeta}, \tilde{\Sigma}), \\
&\tilde{\zeta} = (\sigma^{-2} I_q + \Sigma_i^{-1})^{-1} (\Sigma_i^{-1} \zeta_i + \sigma^{-2} W^t (t - \nu)), \\
&\tilde{\Sigma} = (\sigma^{-2} I_q + \Sigma_i^{-1})^{-1}; \\
&\sigma^{-2} \mid t, \xi \sim \mathcal{G}(d/2 + 1, \frac{1}{2} ||t - W\xi - \nu||^2),
\end{aligned}
\tag{8.27}
$$

where $\mathcal{G}(x; a, b)$ denotes the Gamma distribution $\frac{b^a}{\Gamma(a)} x^{a-1} e^{-bx}$. We can now simulate from the *augmented* posterior distribution of $(\xi_j, c_j, \pi_i, \zeta_i, \Sigma_i, \sigma^2)$ conditional to $t_1, ..., t_N$ by the straightforward Gibbs sampler of Table 8.3. The model parameters are learned from a sample set upon iterating the Gibbs sampler for a few hundreds of iterations and then taking the average value of each parameter. The initialization of the simulation is presented in Table 8.5.

8.5.2 Modified Gibbs energy

Let Θ be the random variable corresponding to the vector of deformations of Section 8.3.2. We model the distribution of Θ by

$$
P(\theta) = \mathcal{U}(\tau_x, \tau_y, s, \psi, \psi_y, \psi_x) p_T(\gamma_{pca}(\xi)) \tag{8.28}
$$

where the first factor is as in equation (8.6), whereas p_T is now given by equation (8.19).

We have the following Lemma.

Lemma 6

The distribution $p_T(\gamma_{pca}(\xi))$ is equal to

$$
\begin{aligned}
&\sum_{i=1}^{\ell} \pi_i \frac{1}{(2\pi)^d |\sigma^2 I_d + U_q \Sigma_i U_q^t|^{1/2}} \\
\times \quad &\exp\big(-\frac{1}{2\sigma^2} (\xi - \zeta_i)^t (I_q - (I_q + \sigma^2 \Sigma_i^{-1})^{-1}) (\xi - \zeta_i)\big).
\end{aligned}
$$

$t_1, ..., t_N$	an i.i.d. sample of the random vector T of equation (8.19);
W, ν	the PCA on $t_1, ..., t_N$;
ℓ	number of Gaussian kernels;
ζ_i	mean of the ith kernel;
Σ_i	covariance matrix of the ith kernel;
π_i	proportion of the ith kernel;
σ^2	variance of the reduction noise;
$A_0, \alpha_1, ..., \alpha_\ell$	parameters for the prior on $\pi_1, ..., \pi_\ell$;
Λ_0, d_0	parameters for the prior on Σ_i;
ζ_0, k_0	parameters for the prior on ζ_i.

Goal: to sample from the posterior distribution of $(\xi_j, c_j, \pi_i, \zeta_i, \Sigma_i, \sigma^2)$ conditional to $t_1, ..., t_N$.

Table 8.3. Gibbs sampler used in Section 8.5.1 (part 1)

Simulate σ^{-2} according to the posterior distribution $\mathcal{G}(Nd/2+1, \frac{1}{2}\sum_{j=1}^{N}||t_j - W\xi_j - \nu||^2)$.

for $j = 1, ..., N$ **do**

sample ξ_j according to the distribution $\xi_j \sim \mathcal{N}((\sigma^{-2}I_q + \Sigma_i^{-1})^{-1}(\Sigma_i^{-1}\zeta_i + \sigma^{-2}W^t(t_j - \nu)), (\sigma^{-2}I_q + \Sigma_i^{-1})^{-1})$.

end for

for $j = 1, ..., N$ **do**

draw $c_j = g_i$ with probability $\omega_i = \frac{\mathcal{N}(\xi_j;\zeta_i,\Sigma_i)\pi_i}{\sum_{i=1}^{\ell}\mathcal{N}(\xi_j;\zeta_i,\Sigma_i)\pi_i}$, where $i = 1, ..., \ell$.

end for

Simulate the mixture parameters according to the posterior distribution $(\pi_1, ..., \pi_\ell) \sim \mathcal{D}\left(N + A_0; \frac{N_i + A_0\alpha_i}{N + A_0}\right)$, where N_i is the number of labels c_j equal to g_i.

for $i = 1, ..., \ell$ **do**

simulate the covariance matrix Σ_i according to the posterior distribution $\Sigma_i \sim \mathcal{IW}(\tilde{\Lambda}, \tilde{d})$, where $\tilde{\Lambda} = \Lambda_0 + N_i S + \frac{N_i k_0}{N_i + k_0}(\bar{\xi} - \zeta_0)(\bar{\xi} - \zeta_0)^t$ and $\tilde{d} = N_i + d_0$, with $\bar{\xi}$ and S the empirical mean and covariance matrix, respectively, of the set of elements ξ_j with hidden label $c_j = g_i$.

simulate the mean ζ_i according to the posterior distribution $\zeta_i \mid \Sigma_i \sim \mathcal{N}\left(\tilde{\zeta}, \frac{1}{\tilde{k}}\Sigma_i\right)$, where $\tilde{\zeta} = \bar{\xi} - \frac{k_0}{N_i + k_0}(\bar{\xi} - \zeta_0)$ and $\tilde{k} = N_i + k_0$.

end for

Table 8.4. Gibbs sampler used in Section 8.5.1 (part 2)

Given a reduced dimension q, compute the PCA W, ν on $t_1, ..., t_N$.

Let $\lambda_1, ..., \lambda_d$ be the corresponding eigenvalues.

Set $\sigma^{-2} = \frac{1}{2d-q} \sum_{i=q+1}^{d} \lambda_i$.

for $j = 1, ..., N$ **do**

Set $\xi_j = W^t(t_j - \nu)$.

end for

Set $d_0 = q + 2$, $k_0 = 0.01$. Let ζ_0 be the empirical mean of the sample set $\xi_1, ..., \xi_N$, and Λ_0 be d_0 times its empirical covariance matrix.

for $j = 1, ..., N$ **do**

draw $c_j = g_i$ with probability $\frac{1}{\ell}$, where $i = 1, ..., \ell$.

end for

Set $A_0 = \ell$ and $\alpha_1 = ... = \alpha_\ell = \frac{1}{\ell}$.

Simulate the mixture parameters according to the posterior distribution $(\pi_1, ..., \pi_\ell) \sim \mathcal{D}\left(N + A_0; \frac{N_i + A_0\alpha_i}{N + A_0}\right)$, where N_i is the number of labels c_j equal to g_i.

for $i = 1, ..., \ell$ **do**

simulate the covariance matrix Σ_i according to the posterior distribution $\Sigma_i \sim \mathcal{IW}(\tilde{\Lambda}, \tilde{d})$, where $\tilde{\Lambda} = \Lambda_0 + N_i S + \frac{N_i k_0}{N_i + k_0}(\bar{\xi} - \zeta_0)(\bar{\xi} - \zeta_0)^t$ and $\tilde{d} = N_i + d_0$, with $\bar{\xi}$ and S the empirical mean and covariance matrix, respectively, of the set of elements ξ_j with hidden label $c_j = g_i$.

simulate the mean ζ_i according to the posterior distribution $\zeta_i \mid \Sigma_i \sim \mathcal{N}\left(\tilde{\zeta}, \frac{1}{\tilde{k}}\Sigma_i\right)$, where $\tilde{\zeta} = \bar{\xi} - \frac{k_0}{N_i + k_0}(\bar{\xi} - \zeta_0)$ and $\tilde{k} = N_i + k_0$.

end for

Table 8.5. Initialization of the Gibbs sampler used in Section 8.5.1

Proof: First, observe that the inverse matrix of $\sigma^2 I_d + W\Sigma_i W^t$ is $\sigma^{-2}\Big(I_d - W(W^tW + \sigma^2\Sigma_i^{-1})^{-1}W^t\Big)$. Now, substitute in equation (8.19), taking $W = U_q$ and using the fact that $U_q^t U_q = I_q$. $\square$

Accordingly, the Gibbs energy of equation (8.16) is replaced by $V_2(\theta, y, x)$ equal to

$$\begin{aligned}
& \sum_{i=0}^{m} \big(\log(\mathcal{W}(y_{s_i}; \min, C, \alpha) - \log(\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j))\big) \qquad (8.29)\\
+ \quad & \sum_{i=1}^{m} \big(\log(\mathcal{U}(y_{s_{i-1},s_i}; 0, \frac{\pi}{2}) - \log(k_0\, \mathcal{E}(y_{s_{i-1},s_i}; \alpha_0))\big)\\
+ \quad & \sum_{k=1}^{K} p_k(\theta) |\{s \,:\, x_s = f_k, s \notin c_\theta^{int}\}|\\
- \quad & \log\Big\{ \sum_{i=1}^{\ell} \pi_i \frac{1}{(2\pi)^d |\sigma^2 I_d + U_q \Sigma_i U_q^t|^{1/2}}\\
& \qquad \times \exp\big(-\frac{1}{2\sigma^2}(\xi - \zeta_i)^t (I_q - (I_q + \sigma^2\Sigma_i^{-1})^{-1})(\xi - \zeta_i)\big)\Big\}.
\end{aligned}$$

Now, we consider ℓ auxiliary change of variables, one for each Gaussian kernel:

$$\xi = \varphi_i(\tilde{\xi}) = A_i\tilde{\xi} + \zeta_i, \qquad (8.30)$$

for $i = 1, ..., \ell$, where $\Sigma_i = A_i A_i^t$. Instead of equation (8.18), we now consider the domain

$$\begin{aligned}
(\tau_x, \tau_y, s, \psi, \psi_x, \psi_y) \quad &\in \quad [0, M-1] \times [0, N-1] \times [\rho_1 d, \rho_2 d]\\
&\quad \times [0, 2\pi] \times [-\psi_0, \psi_0]^2\\
\tilde{\xi} \quad &\in \quad [-1, 1]^q. \qquad (8.31)
\end{aligned}$$

This is equivalent to a parametrization of the domain for ξ. In particular, it would not make sense to require that $\xi \in [-1, 1]^q$. In the E/S algorithm of Table 8.1, the choice of the auxiliary change of variables is modified randomly with probability 1/2.

8.6 Global constraint in the case of multiple occurence of the object

When an object is expected to appear in multiple instances in the image, the specificity hypothesis (8.13) is replaced by the following hypothesis:

Local object specificity: the color labels inside and outside the object are distinct within a neighborhood of the object. (8.32)

Formally, let $x = (x_s)$ be a classification of the pixels in the image into K equivalence classes according to the colors as in section 8.2.2, with $x_s \in \Lambda = \{f_1, ..., f_K\}$. Consider the global constraint

$$V_{loc}(x, \theta) = \sum_{k=1}^{K} p_k(\theta) |\{s \,:\, x_s = f_k, s \in c_\theta^{nbhd} \setminus c_\theta^{int}\}|, \tag{8.33}$$

where $p_k(\theta_k)$ is the proportion of the points inside the curve γ_θ having label f_k, c_θ^{int} is the interior of the curve, and c_θ^{nbhd} is a neighborhood of the curve. Thus, $V_{loc}(x, \theta)$ is minimal whenever the region labels inside an object are *locally* specific to that object. In our tests, we took a neighborhood of radius of 25 pixels. Accordingly, equation (8.29) is replaced by $V_3(\theta, y, x)$

$$\begin{aligned}
& \sum_{i=0}^{m} \big(\log(\mathcal{W}(y_{s_i}; \min, C, \alpha) - \log(\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j))\big) \qquad (8.34)\\
+ \;& \sum_{i=1}^{m} \big(\log(\mathcal{U}(y_{s_{i-1},s_i}; 0, \frac{\pi}{2}) - \log(k_0\, \mathcal{E}(y_{s_{i-1},s_i}; \alpha_0))\big)\\
+ \;& \sum_{k=1}^{K} p_k(\theta) |\{s \,:\, x_s = f_k, s \in c_\theta^{nbhd} \setminus c_\theta^{int}\}|\\
- \;& \log\Big\{ \sum_{i=1}^{\ell} \pi_i \frac{1}{(2\pi)^d |\sigma^2 I_d + U_q \Sigma_i U_q^t|^{1/2}}\\
& \qquad \times \exp\big(-\frac{1}{2\sigma^2}(\xi - \zeta_i)^t (I_q - (I_q + \sigma^2 \Sigma_i^{-1})^{-1})(\xi - \zeta_i)\big)\Big\}.
\end{aligned}$$

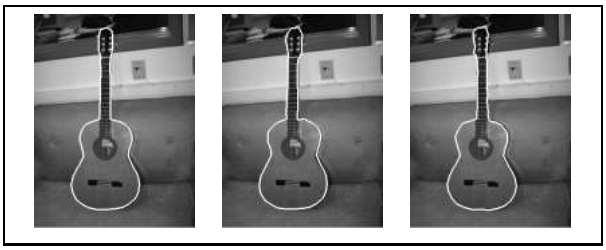


Figure 8.5. Example of non-linear deformations of a shape. Results obtained for three different sets of values, fixing the rigid transformation parameters. Left: $\psi_x = \psi_y = \xi_1 = 0$. **Center:** $\psi_x = \psi_y = 0$ **and** $\xi_1 = 3$. **Right:** $\psi_x = \psi_y = 0$ **and** $\xi_1 = -3$.

8.7 Experimental results

8.7.1 Comparing a few segmentation models

A first experiment uses a 20th century classical guitar as shape. The data base consists of 28 pictures. Each curve is represented by a template of 70 points obtained as follows. We took 7 key-points on the curve and split each of the 7 portions of the curve with 10 equally spaced points (see Fig. 8.3). The training phase of Section 8.3.1 yields a reduced dimension of $q = 1$ for the non-linear deformations with less than 1% for the relative average reconstruction error. We present in Fig. 8.5 three deformations of the mean shape and the corresponding value of the non-linear parameter. Note that the deformations do not vary significantly from the mean shape, since 20th century classical guitars have almost identical proportions. But the statistical model of Section 8.5.1 allows for more variable shapes. In Fig. 8.6, projective deformations are presented.

The localization procedure depends crucially on the color segmentation of the image. We have experimented the procedure with the following 4 segmentation models:

1. The color model of Section 8.2.2 (with $K = K_1 = 30$).

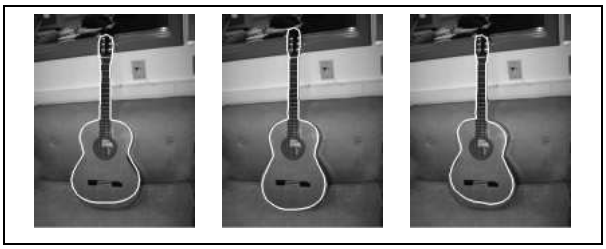


Figure 8.6. Example of projective deformations of a shape. Results obtained for three different sets of values, fixing the rigid transformation parameters. Left: $\psi_x = 0$, $\psi_y = \frac{\pi}{8}$, **and** $\xi_1 = 0$. **Center:** $\psi_x = \frac{\pi}{8}$, $\psi_y = 0$, **and** $\xi_1 = 0$. **Right:** $\psi_x = \psi_y = \frac{\pi}{8}$, **and** $\xi_1 = 0$.

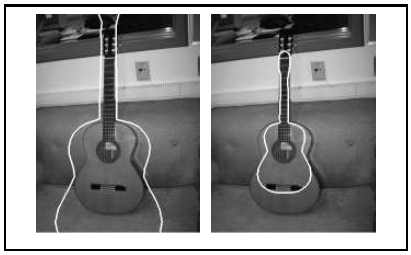


Figure 8.7. Example of scaling deformations of a shape. Results obtained for three different values of ρ, **fixing all other parameters. Left:** $\rho = 0.5$. **Right:** $\rho = 0.25$.

2. The Mean Shift algorithm [28], with a spatial bandwidth of 7 and a range bandwidth of 6.5.

3. The color model of [51] (with $K = 30$).

4. The K-means algorithm (with $K = 30$).

See Fig. 8.2 for examples of segmentation according to each of the 4 models. For the Mean Shift algorithm, we have used the code available at the following URL: http://www.caip.rutgers.edu/riul/research/code.html.

We have tested the localization procedure using equation (8.16) with the 4 segmentation models on 30 images with 5 different initial seeds. See Fig. 8.8. The number of seeds yielding to a failure for each image is indicated in Table 8.6. For image (1) of Fig. 8.8, the whole procedure takes about 50 min. The segmentation method takes 90% of that time. When using the K-means algorithm, the whole procedure takes only 5 minutes.

As can be seen, the proposed color model outperforms the other models, except for time considerations. In our opinion, this is due to the fact that for models (2)-(4), the distribution of each color region is unimodal, whereas in the case of the proposed model, the region distributions are multi-modal. It follows that methods (2)-(4) yield to an over-segmentation of the image, which in turn makes the optimization task too difficult. However, quite surprisingly, the K-means algorithm ranks relatively well for that data set.

We have also experimented with the following schemes:

5) The color model of Section 8.2.2 without the contour model of Section 8.2.1.

6) The contour model of Section 8.2.1 and no segmentation model.

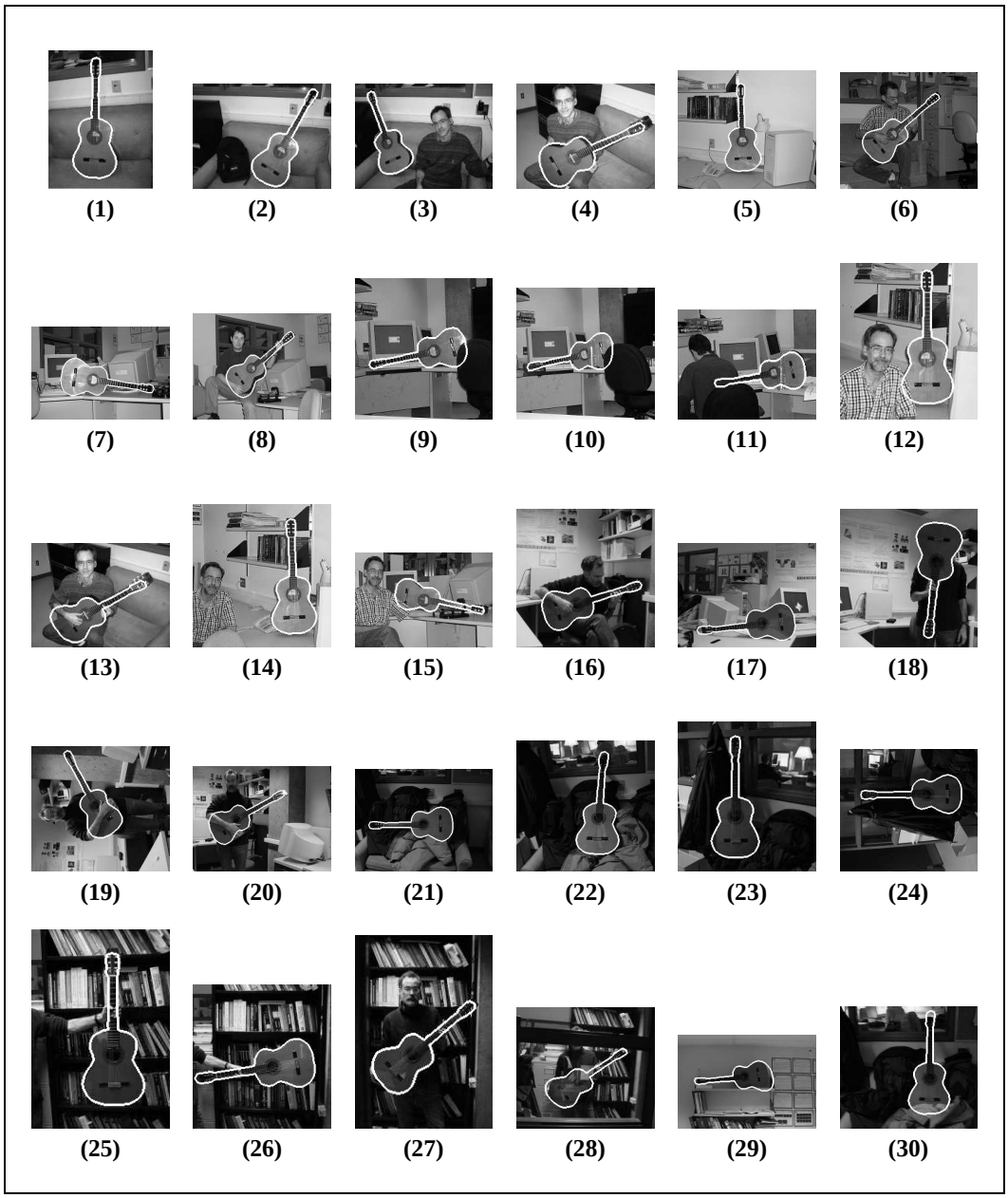


Figure 8.8. Examples of localization of a shape obtained by stochastic optimization of the Gibbs field based on the contour parameters estimated by the ICE procedure and the segmentation computed with the ESE procedure, for the guitar shape (images (1) to (30)).

Image no	(1)	(2)	(3)	(4)	(5)	(6)
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	1
4	0	3	2	1	0	3
5	0	0	0	0	0	0
6	0	1	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	1	0
10	0	0	0	0	0	0
11	0	3	0	0	0	2
12	0	5	3	0	0	5
13	0	0	0	0	1	1
14	0	1	0	0	0	2
15	0	5	0	0	5	4

Table 8.6. Number of seeds leading to a wrong solution for the images of Fig. 8.8. Segmentation models: (1) the color model of Section 8.2.2; (2) the Mean Shift algorithm; (3) the color model of [51]; (4) the K-means algorithm (with $K = 30$); (5) the color model of Section 8.2.2 but no contour model; (6) only the contour model (part 1).

Image no	(1)	(2)	(3)	(4)	(5)	(6)
16	0	5	2	0	0	5
17	0	0	0	0	1	3
18	0	0	3	0	0	5
19	0	0	0	0	3	1
20	0	0	0	0	2	1
21	0	1	0	0	2	3
22	0	1	1	0	1	2
23	0	5	1	2	0	5
24	0	4	4	0	0	4
25	0	5	3	2	0	5
26	0	5	1	1	0	5
27	1	5	4	4	0	5
28	0	4	1	0	0	4
29	2	5	4	2	1	5
30	4	2	0	0	5	1
error rate	4.7%	40%	19.3%	8%	14.7%	48%

Table 8.7. Number of seeds leading to a wrong solution for the images of Fig. 8.8. Segmentation models: (1) the color model of Section 8.2.2; (2) the Mean Shift algorithm; (3) the color model of [51]; (4) the K-means algorithm (with $K = 30$); (5) the color model of Section 8.2.2 but no contour model; (6) only the contour model (part2).

See Table 8.6 for the results. As can be seen, contours or regions alone are not sufficient to localize efficiently the shape. In [44, 47, 48], the success rates were much higher for most images because initialization procedures were used. In this paper, we have dropped those initialization procedures, in order to show that a localization procedure based solely on contours is a much harder optimization problem. Furthermore, despite various initialization procedures, there were still a few images that presented 0% success rate, whereas now each image has at least a 20% success rate. So, we are inclined to think that the proposed model is more adequate than one that is based solely on contours.

In the case of the segmentation model of Section 8.2.2, out of the 5 initial seeds, the lowest value of the Gibbs energy always corresponds to a good localization. Thus, the actual error rate for 5 initial seeds is actually 0%. In the case of the K-means algorithm, the same conclusion holds.

8.7.2 Experimenting the new deformation model

A second experiment uses a van as shape. The data base consists of 27 pictures. Each curve is represented by a template of 120 points. The training phase of Section 8.5.1 yields a reduced dimension of $q = 3$ for the non-linear deformations with less than 0.07% for the relative reconstruction error. We fixed the number ℓ of gaussian kernels to 2. Note that there is more variation in the shape of a van than for a classical guitar. This point justifies the use of a mixture of Gaussian kernels, rather than a single one. In principle, one could use Bayes factors in order to determine the number of gaussian kernels, but we omit this technical aspect in this paper.

We have tested the localization procedure with the deformation model of equation (8.29) on 24 images with 5 different initial seeds. See Fig. 8.9 for examples of localization. The error rate was 7.5% with the segmentation method of Section 8.2.2, whereas it was 15% with the K-means algorithm.

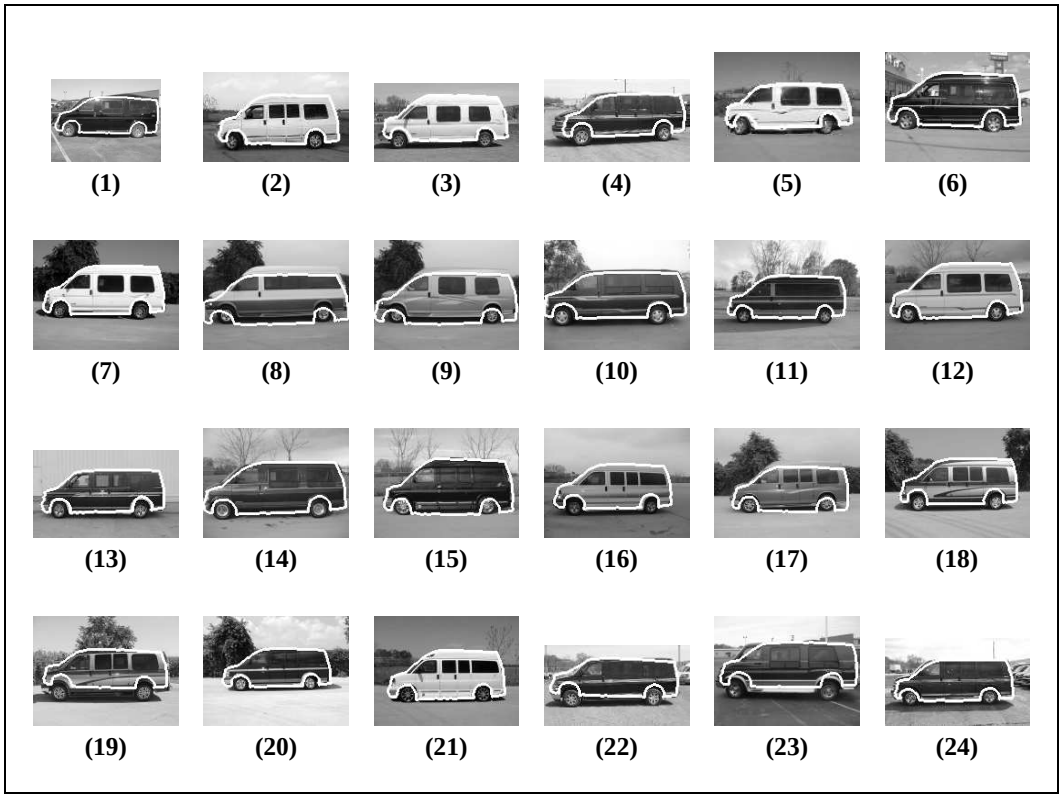


Figure 8.9. Examples of localization of a shape obtained by stochastic optimization of the Gibbs field based on the contour parameters estimated by the ICE procedure and the segmentation computed with the ESE procedure, for the van shape (images (1) to (24)).

Image no	(1)	(4)
1	0	0
2	0	0
3	0	0
4	1	0
5	0	0
6	0	1
7	0	0
8	3	4
9	1	3
10	0	1
11	0	0
12	0	0

Table 8.8. Number of seeds leading to a wrong solution for the images of Fig. 8.9. Segmentation models: (1) the color model of Section 8.2.2; (4) the K-means algorithm (with $K = 30$) (part 1).

Image no	(1)	(4)
13	0	1
14	0	0
15	0	0
16	0	1
17	0	0
18	0	0
19	1	4
20	0	0
21	0	0
22	2	1
23	1	1
24	0	1
error rate	7.5%	15%

Table 8.9. Number of seeds leading to a wrong solution for the images of Fig. 8.9. Segmentation models: (1) the color model of Section 8.2.2; (4) the K-means algorithm (with $K = 30$) (part 2).

Again, in the case of the segmentation model of Section 8.2.2, out of the 5 initial seeds, the lowest value of the Gibbs energy corresponded to a good localization for all images. The same fact holds for the K-means algorithm. Thus, in both cases, the actual error rate for 5 initial seeds is actually 0%.

8.7.3 Multiple objects

A third experiment uses a saxophone as shape. The data base consists of 20 pictures. Each curve is represented by a template of 60 points. The training phase of Section 8.5.1 yields a reduced dimension of $q = 1$ for the non-linear deformations with less than 0.006% for the relative reconstruction error. We fixed the number ℓ of gaussian kernels to 1. We experimented with an image where the object appears more than once. Thus, we consider the local specificity hypothesis (8.32).

We have tested the localization procedure with the deformation model of equation (8.34) on 1 image (c.f. Fig. 8.10) with 30 different initial seeds. The error rate was 6.7% with the segmentation method of Section 8.2.2. The lowest value of the Gibbs energy of equation (8.34) corresponded to a good localization. When using the equation (8.29) corresponding to the global specificity hypothesis, we obtain an error rate of 66.7%. Thus, in the case of multiple occurence of an object one has to replace the global specificity hypothesis by the local one. Note that different seeds yield to possibly different solutions, such as can be seen in Fig. 8.10.

8.8 Conclusion

In this paper, we have presented a coherent statistical Bayesian model for deformations of shapes. We have brought together the following ideas. The prior distribution of deformations can be learned using the PPCA, or the proposed mixture of PPCA. The likelihood distribution of deformations is based on a statistical model for the gradient vector field of the gray level in the image, and can be estimated using an

Figure 8.10. Examples of localization of a shape obtained by stochastic optimization of the Gibbs field based on the contour parameters estimated by the ICE procedure and the segmentation computed with the ESE procedure, for the saxophone shape. The image on the left is counted as wrong, and the two others are counted as right.

ICE procedure. A criterion of global or local object specificity makes the localization of the shape a lot easier. This criterion is based on a color segmentation of the image that can be computed with an ESE procedure. The optimization E/S algorithm converges asymptotically to an optimal solution, in the sense of the MAP in our context. The error rates with our method were 4.7%, 7.5%, and 6.7% for three sets of experiments. In future work, we intend to develop a (much) more efficient version of the segmentation method used in this paper, in terms of the computational time.

Acknowledgments

The authors thank FCAR (Fonds formation chercheurs & aide recherche, Qc., Canada) for financial support of this work. We thank the music stores Archambault and Steve's Music in Montreal for kindly allowing us to take pictures of classical guitars.

Chapitre 9

CONCLUSION

9.1 Contribution de cette thèse en traitement d'images

Dans cette thèse, nous avons présenté trois articles qui couvrent les trois volets suivants : 1) l'optimisation stochastique ; 2) la modélisation markovienne d'images en vue de la segmentation et l'estimation des paramètres ce ces modèles au sens du MAP ; 3) la localisation de formes.

Optimisation stochastique

Dans le premier article, nous avons démontré que l'algorithme d'Exploration/Selection de O. François est valide dans le cadre général d'un graphe d'exploration connexe, mais pas nécessairement symétrique, et d'une distribution d'exploration positive quelconque. Cette version plus souple nous a permis d'utiliser l'E/S avec un noyau d'exploration basé sur l'échantillonneur de Gibbs pour le calcul du MAP de modèles markoviens dans les deux premiers articles. Nous avons également utilisé notre version de l'E/S dans le calcul de la localisation de formes dans le troisième article.

Dans le deuxième article, nous avons présenté des bornes en temps fini dans le cas d'un graphe d'exploration complet, mais d'une distribution d'exploration positive quelconque. Ces bornes nous permettent d'ajuster de façon heuristique les paramètres internes de l'algorithme E/S.

Nous espérons que la version générale de l'algorithme E/S présentée dans cette thèse s'avèrera utile dans d'autres domaines de l'informatique. L'avantage théorique de cet algorithme est que les paramétres internes qui en assurent la convergence asymptotique sont connus explicitement et qu'ils sont exploitables en pratique. En

outre, ils ne dépendent que du diamètre du graphe d'exploration.

Modélisation d'images en vue de la segmentation et estimation des paramètres ce ces modèles au sens du MAP

Dans le premier article, nous avons proposé un couple de champs markoviens, formé du champ observable des couleurs et du champ discret caché des régions. Nous avons considèré des distributions de vraisemblance définies à partir de distributions unimodales, mais non gaussiennes. Nous avons supposé le nombre de régions inconnu mais borné. De plus, nous avons ajouté des contraintes globales sur la taille et le nombre de régions.

Dans le deuxième article, nous avons considéré un triplet de champs markoviens : le champs observable joint des attributs de textures et de couleurs ; le champs discret caché des régions ; et le champs discret caché des classes de couleurs et de textons. En plus de considérer un nombre de régions inconnu et une contrainte globale sur la taille et le nombre de régions, nous avons supposé un hyper-paramètre inconnu pour le modèle markovien du processus des régions. Le deuxième modèle présente l'avantage d'offrir une paramétrisation des ensembles de Julesz plus simple que celle du modèle FRAME. Le modèle proposé s'applique au cadre plus général de la fusion de données.

Pour l'estimation des paramètres, notre cadre de travail a été celui du paradigme bayésien. Dans les deux premiers articles, nous avons proposé une variante originale de l'algorithme E/S qui consiste à utiliser comme noyau d'exploration, un noyau approché de l'échantillonneur de Gibbs. Cette méthode permet de calculer le MAP des modèles proposés plus efficacement qu'un RJMCMC ou qu'un RS, comme le montrent les résultats présentés en annexe B et C. De plus, nous avons testé la méthode proposée dans le cadre du calcul du mode d'un mélange de deux noyaux gaussiens (voir annexe D). Nos tests montrent que notre méthode est plus efficace que la version originale de l'algorithme E/S, sauf dans le cas où le domaine de définition

englobe de près la solution cherchée.

Localisation de formes

Dans mon mémoire de maîtrise, nous avions présentés un modèle qui comportait une densité *a priori* basée sur les résultats de la PPCA [134], et une distribution de vraisemblance qui utilise un modèle statistique des contours dans une image. Dans cette thèse, nous avons proposé un modèle des déformations dont la densité *a priori* peut être un mélange de PPCA, ainsi qu'une méthode de type MCMC pour en estimer les paramètres. De plus, outre le modèle statistique des contours, nous avons proposé une contrainte globale qui exploite une segmentation préalable de l'image en régions basées sur les couleurs. Cette contrainte présente l'avantage majeur de faciliter grandement la recherche stochastique de la forme dans l'image, effectuée à l'aide de l'algorithme E/S.

Selon notre point de vue, le problème de la segmentation d'images n'a de sens qu'en vue d'une tâche de haut-niveau particulière. Dans cette thèse, nous avons comparé notre méthode de segmentation avec deux autres méthodes dans le cadre de la localisation de formes. Les tests effectués indiquent un net avantage de la méthode proposée au point de vue de la fiabilité et précision de la localisation. Par contre, le temps de calcul requis est plus important.

9.2 Avenues de recherche

La recherche effectuée dans cette thèse admet plusieurs extensions.

En optimisation stochastique, il serait intéressant de développer des bornes en temps fini pour la convergence de l'algorithme E/S dans le cas d'un diamètre d'exploration quelconque, bien qu'il est toujours possible de se ramener au cas particulier d'un graphe complet.

Au niveau de la modélisation d'images, nous avons l'intention d'intégrer les as-

pects de l'illumination (réflection, ombrage, transparence) dans le cadre du modèle de fusion couleurs/textures déjà élaboré. En particulier, nous sommes intéressé au phénomène de transparence en imagerie radiologique. Un problème connexe concerne la restauration d'images qui tient compte du flou et du bruit appliqué aux données brutes de l'image. Dans ces deux exemples, il s'agit d'ajouter un niveau hiérarchique au modèle présenté au chapitre 6. Une autre direction de recherche consiste à considérer un modèle *a priori* spatial autre que celui de Potts. À ce sujet, le modèle générique de Mumford présenté à la section 5.3.2 semble pouvoir s'adapter à un modèle générique du processus des régions.

Le modèle de segmentation proposé est utile pour la localisation d'images, sauf en ce qui a trait au temps de calcul. Nous avons donc l'intention de développer une version beaucoup plus rapide de notre méthode de segmentation. Pour ce faire, il suffit de considérer un graphe de petites régions obtenues à l'aide d'une segmentation préliminaire, comme il est fait dans l'article [5]. En travaillant sur un graphe de taille beaucoup plus petite, nous espèrons ainsi diminuer le temps alloué à chaque itération de la procédure ESE.

Finalement, il nous semble intéressant de modéliser la réduction de la dimensionalité de données par un mélange de PPCA, mais basée sur des distributions de Student, plutôt que gaussiennes. Encore là, il suffit d'ajouter un niveau hiérarchique au modèle présenté au chapitre 8. Il reste à savoir si un tel modèle permet une plus grande flexibilité sur les déformations des formes.

RÉFÉRENCES

[1] *Coloremetry, 2nd Edition*. Publication CIE 15.2-1986.

[2] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis, second edition*. John Wiley & Sons, 1971.

[3] C. Andrieu et A. Doucet. Simulated Annealing for Maximum *a posteriori* Parameter Estimation of Hidden Markov Models. *IEEE Trans. Information Theory*, 46(3):994–1004, 2000.

[4] S. Banks. *Signal processing, image processing and pattern recognition*. Prentice Hall, 1990.

[5] A. Barbu et S.-C. Zhu. Generalizing Swendsen-Wang to Sampling Arbitrary Posterior Probabilities. *IEEE Trans. Pattern Anal. Machine Intell.*, 27(8):1239–1253, 2005.

[6] S. A. Barker et P. J. W. Rayner. Unsupervised image segmentation using Markov random field models. Dans *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 165–178. Springer-Verlag, 1997.

[7] M. G. Bello. A combined Markov random field and wave-packet transform-based approach for image segmentation. *IEEE Trans. Geosci. Remote Sensing*, 31(3):618–633, 1993.

[8] S. Benameur, M. Mignotte, F. Destrempes, et J. A. de Guise. 3D Biplanar Reconstruction of Scoliotic Rib Cage using the Estimation of a Mixture of Probabilistic Prior Models. *IEEE Trans. Biomed. Eng.*, 52(10):1713–1728, 2005.

[9] S. Benameur, M. Mignotte, H. Labelle, et J. A. de Guise. A Hierarchical Statistical Modeling Approach for the Unsupervised 3D Biplanar Reconstruction of the Scoliotic Spine. *IEEE Trans. Biomed. Eng.*, 52(12):2041–2057, 2005.

[10] A. Bendjebbour, Y. Delignon, L. Fouque, V. Samson, et W. Pieczynski. Multisensor Image Segmentation Using Dempster-Shafer Fusion in Markov Fields Context. *IEEE Trans. Geosci. Remote Sensing*, 39(8):25–32, 2001.

[11] J. R. Bergen et E. H. Adelson. Theories of visual texture perception. Dans *Spatial Vision*. D. Regan (ed.), CRC press, 1991.

[12] J. Besag. Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Society, Series B*, 36:192–236, 1974.

[13] J. Besag. Statistical analysis of non-lattice data. *The Statistician*, 24:179–195, 1977.

[14] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, 48:259–302, 1986.

[15] C. Bouman et M. Shapiro. A multiscale image model for Bayesian image segmentation. *IEEE Trans. Image Processing*, 3(2):162–177, 1994.

[16] C. A. Bouman et K. Sauer. A Unified Approach to Statistical Tomography Using Coordinate Descent Optimization. *IEEE Trans. Image Processing*, 5(3):480–492, 1996.

[17] B. Braathen, P. Masson, et W. Pieczynski. Global and local methods of unsupervised Bayesian segmentation of images. *Machine Graphics and Vision*, 2(1):39–52, 1993.

[18] D. J. Burr. Elastic matching of line drawings. *IEEE Trans. Pattern Anal. Machine Intell.*, 3(6):708–713, 1981.

[19] H. Caillol, W. Pieczynski, et A. Hillon. Estimation of fuzzy gaussian mixture and unsupervised statistical image segmentation. *IEEE Trans. Image Processing*, 6(3):425–440, 1997.

[20] O. Cappé, C. P. Robert, et T. Rydén. Reversible jump, birth-and-death and more general continuous time Markov chain Monte Carlo samplers. *Journal of the Royal Statistical Society, Series B*, 65:679–700, 2003.

[21] R. Cerf. The dynamics of mutation-selection algorithms with large population sizes. *Ann. Inst. Henri Poincaré Probab. Statist.*, 32(4):455–508, 1996.

[22] R. Cerf. A new genetic algorithm. *Ann. Appl. Probab.*, 6(3):778–817, 1996.

[23] A. Chakraborty, L. H. Staib, et J. S. Duncan. Deformable boundary finding influenced by region homogeneity. Dans *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Seattle*, pages 624–627, June 1994.

[24] B. Chalmond. An iterative Gibbsian technique for reconstruction of m-ary images. *Pattern Recognition*, 22(6):747–761, 1989.

[25] P. C. Chen et T. Pavlidis. Image segmentation as an estimation problem. *Computer Graphics and Image Understanding*, 12:153–172, 1980.

[26] H. Choi et R. Baraniuk. Multiscale image segmentation using wavelet-domain hidden markov models. *IEEE Trans. Image Processing*, 10(9):1309–1321, 2001.

[27] C. Chubb et M. S. Landy. Orthogonal distributin analysis: a new approach to the study of texture perception. Dans *Comp. Models of Visual Proc.* Landy *et al.* (eds.), MIT press, 1991.

[28] D. Comaniciu et P. Meer. A Robust Approach Toward Feature Space Analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(5):603–619, 2002.

[29] F. Comets. On consistency of a Class of Estimators for Exponential Families of Marlov Random Fields on the Lattice. *Annals of Statistics*, 20(1):455–468, 1992.

[30] T. F. Cootes et C. J. Taylor. A mixture model for representing shape variation. *Image and Vision Computing*, 17(8):567–574, 1999.

[31] T. F. Cootes, C. J. Taylor, D. H. Cooper, et J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[32] T. F. Cootes, C. J. Taylor, et J. Haslam. The use of active shape models for locating structures in medical images. *Image and Vision Computing*, 12(6):355–366, 1994.

[33] D. Cremers, T. Kohlberger, et C. Schnorr. Shape statistics in kernel space for variational image segmentation. *Pattern Recognition*, 36(9):1929–1943, 2003.

[34] D. Cremers et C. Schnorr. Statistical shape knowledge in variational motion segmentation. *Image and Vision Computing*, 21(1):77–86, 2003.

[35] D. Crisan et A. Doucet. A Survey of Convergence Results on Particle Filtering Methods for Practitioners. *IEEE Trans. Signal Processing*, 50(3):736–746, 2002.

[36] M. J. Daily. Color image segmentation using Markov random fields. Dans *Proc. DARPA Image Understanding*, 1989.

[37] I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[38] J. G. Daughman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America*, 2(7):1160–1169, 1985.

[39] Y. Delignon, A. Marzouki, et W. Pieczynski. Estimation of generalized mixture and its application in image segmentation. *IEEE Trans. Image Processing*, 6(10):1364–1375, 1997.

[40] Y. Delignon et W. Pieczynski. Modeling Non-Rayleigh Speckle Distribution in SAR images. *IEEE Trans. Geosci. Remote Sensing*, 40(6):1430–1435, 2002.

[41] A.P. Dempster, N.M. Laird, et D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Royal Statistical Society, Series B*, pages 1–38, 1976.

[42] H. Derin et H. Elliott. Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(1):39–55, 1987.

[43] X. Descombes, R. D. Morris, J. Zerubia, et M. Berthod. Estimation of Markov Random Field Prior Parameters Using Markov Chain Monte Carlo Maximum Likelihood. *IEEE Trans. Image Processing*, 8(7):954–962, 1999.

[44] F. Destrempes. Détection non-supervisée de contours et localisation de formes à l'aide de modèles statistiques. Mémoire de maîtrise, Université de Montréal, April 2002.

[45] F. Destrempes, J.-F. Angers, et M. Mignotte. Fusion of Hidden Markov Random Field Models and its Bayesian Estimation. *IEEE Trans. Image Processing*, submitted.

[46] F. Destrempes et M. Mignotte. Unsupervised detection and semi-automatic extraction of contours using a statistical model and dynamic programming. Dans *4th IASTED International Conference on Signal and Image Processing, Kaua'i Marriott, Hawaii, USA*, pages 60–65, August 2002.

[47] F. Destrempes et M. Mignotte. Unsupervised localization of shapes using statistical models. Dans *4th IASTED International Conference on Signal and Image Processing, Kaua'i Marriott, Hawaii, USA*, pages 66–71, August 2002.

[48] F. Destrempes et M. Mignotte. Unsupervised statistical method for edgel clustering with application to shape localization. Dans *3rd Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP'02, Space Application Centre (ISRO), Ahmedabad, India*, pages 411–416, December 2002.

[49] F. Destrempes et M. Mignotte. Unsupervised texture segmentation using a statistical wavelet-based hierarchical multi data model. Dans *10th IEEE International Conference on Image Processing (ICIP'2003)*, volume II, pages 1053–1056, Barcelona, Spain, Septembre 2003.

[50] F. Destrempes et M. Mignotte. A statistical model for contours in images. *IEEE Trans. Pattern Anal. Machine Intell.*, 26(5):626–638, 2004.

[51] F. Destrempes, M. Mignotte, et J.-F. Angers. A Stochastic Method for Bayesian Estimation of Hidden Markov Random Field Models with Application to a Color Model. *IEEE Trans. Image Processing*, 14(8):1096–1124, 2005.

[52] R. O. Duda, P. E. Hart, et D. G. Stork. *Pattern Classification*. John Wiley & Sons, Inc., 2001.

[53] M. Escobar. Estimating normal means with a Dirichlet process prior. *Journal of the American Statistical Association*, 89(425):268–277, 1993.

[54] G. Fan et X.-G. Xia. A joint multicontext and multiscale approach to Bayesian segmentation. *IEEE Trans. Geosci. Remote Sensing*, 39(12):2680–2688, 2001.

[55] O. Féron et A. Mohammad-Djafari. A Hidden Markov model for Bayesian fusion of multivariate signals. Dans *Fifth Int. Triennial Calcutta Symposium on Probability and Statistics, 28-31 December 2003, Dept. of Statistics, Calcutta University, Kolkata, India*, December 2003.

[56] M. Figueiredo et J. Leitao. Bayesian estimation of ventricular contours in angiographic images. *IEEE Trans. Med. Imag.*, 11(3):416–429, 1992.

[57] J. D. Foley, A. van Dam, S. K. Feiner, et J. F. Hughes. *Computer Graphics, Principles and Practice, second edition in C*. Addison-Wesley, New-York, 1996.

[58] O. François. An evolutionary strategy for global minimization and its Markov chain analysis. *IEEE Trans. Evol. Comput.*, 2(3):77–90, 1998.

[59] O. François. Global Optimization with Exploration/Selection Algorithms and Simulated Annealing. *Ann. Appl. Probab.*, 12(1):248–271, 2002.

[60] M. I. Freidlin et A. D. Wentzell. *Random Perturbations of Dynamical Systems*. Springer-Verlag, New-York, 1984.

[61] D. Gabor. Theory of communication. *IEEE Proc.*, 93(26), 1946.

[62] D. Geman et B. Jedynak. An active testing model for tracking roads in satellite images. *IEEE Trans. Pattern Anal. Machine Intell.*, 18(1):1–14, 1996.

[63] S. Geman et D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, 6(6):721–741, 1984.

[64] N. Giordana et W. Pieczynski. Estimation of generalized multisensor hidden Markov chains and unsupervised image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 19(5):465–475, 1997.

[65] R. Gnanadesikan, R. S. Pinkhan, et L. P. Hughes. Maximum Likelihood Estimation of the Parameters of the Beta Distribution from Smallest Order Statistics. *Technometrics*, 9(4):607–620, 1967.

[66] P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–731, 1995.

[67] P. J. Green et A. Mira. Delayed rejection in reversible jump Metropolis-Hastings. *Journal of the Royal Statistical Society, Series B*, 88(4):1035–1053, 2001.

[68] U. Grenander. *Pattern synthesis: Lectures in pattern theory*, volume 18. Springer-Verlag, 1976.

[69] U. Grenander et A. Srivastava. Probability models for clutter in natural images. *IEEE Trans. Pattern Anal. Machine Intell.*, 23(4):424–429, 2001.

[70] U. Grenander et A. Srivastava. Universal Analytical Forms for Modeling Image Probabilities. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(9):1201–1214, 2002.

[71] R. Griffiths et D. Ruelle. Strict Convexity (Continuity) of the Pressure in Lattice Systems. *Commun. Math. Physics*, 23:169–175, 1971.

[72] B. Hajek. Cooling schedule for optimal annealing. *Mathematics of Operations Research*, 13:311–329, 1988.

[73] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their application. *Biometrika*, 57:97–109, 1970.

[74] G. Healy et T. O. Binford. A color metric for computer vision. Dans *Proc. DARPA Image Understanding*, pages 854–861, 1988.

[75] T. N. Herzog. *Introduction to Credibility Theory.* ACTEX Publications, Inc., USA, 1994.

[76] C.-R. Hwang et S.-J. Sheu. Singular perturbed Markov chains and exact behaviors of simulated annealing process. *J. Theoret. Probab.*, 5:223–249, 1992.

[77] A. K. Jain et F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, 24(12):1167–1186, 1992.

[78] A. K. Jain, Y. Zhong, et S. Lakshmanan. Object matching using deformable templates. *IEEE Trans. Pattern Anal. Machine Intell.*, 18(3):267–278, 1996.

[79] A. K. Jain et D. Zongker. Representation and recognition of handwritten digits using deformable templates. *IEEE Trans. Pattern Anal. Machine Intell.*, 19(12):1386–1391, 1997.

[80] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106:620–630, 1957.

[81] M.-P. Dubuisson Jolly, S. Lakshmanan, et A. K. Jain. Vehicle segmentation and classification using deformable templates. *IEEE Trans. Pattern Anal. Machine Intell.*, 18(3):293–308, 1996.

[82] M. C. Jones. Generating Inverse Wishart Matrices. *Communications in Statistics - Simulation and Computation*, 14(2):511–514, 1985.

[83] B. Julesz. Visual Pattern Discrimination. *IEEE Trans. on Information Theory*, 8(2):84–92, 1962.

[84] M. Kass, A. Witkin, et D. Terzopoulos. Snakes: active contour models. *International Journal Computer Vision*, 1(4):321–331, 1988.

[85] Z. Kato. Bayesian color image segmentation using reversible jump Markov chain Monte Carlo, 1999. Research Report01/99-R055, ERCIM.

[86] Z. Kato, M. Berthod, et Z. Zerubia. A hierarchical Markov random field model and multi-temperature annealing for parallel image classification. *Graph. Mod. Image Process.*, 58(1):18–37, 1996.

[87] Z. Kato et T. C. Pong. A Markov Random Field Image Segmentation Model Using Combined Color and Texture Features. Dans *Proceedings of International Conference on Computer Analysis of Images and Patterns, W. Skarbek Ed., Springer; Warsaw, Poland, Sept. 2001*, pages 547–554, 2001.

[88] Z. Kato, T. C. Pong, et J. C. M. Lee. Motion compensated color video classification using Markov random fields. Dans R. Chin et T. C. Pong, editeurs, *Proc. ACCV*, volume I, pages 738–745, Hong Kong, China, Janvier 1998.

[89] Z. Kato, T.-C. Pong, et S. G. Qiang. Unsupervised Segmentation of Color Textures Images Using a Multi-Layer MRF Model. Dans *10th IEEE International Conference on Image Processing ICIP' 2003, Barcelona, Spain*, September 2003.

[90] H. Künsch, S. Geman, et A. Kehagias. Hidden Markov Random Fields. *Annals of Applied Probability*, 5:577–602, 1995.

[91] C. Kervrann et F. Heitz. A hierarchical markov modeling approach for the segmentation and tracking of deformable shapes. *Graphical Models and Image Processing*, 60(3):173 – 195, 1998.

[92] J. F. C. Kingman. *Poisson Processes*. Oxford Studies in Probability, Clarendon Press, Oxford, 1993.

[93] J.-M. Laferté, P. Pérez, et F. Heitz. Discrete markov image modeling and inference on the quadtree. *IEEE Trans. Image Processing*, 9(3):390–404, 2000.

[94] S. Lakshmanan et H. Derin. Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealings. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(8):799–813, 1989.

[95] A. B. Lee, J. G. Huang, et D. B. Mumford. Occlusion Models for Natural Images. *International Journal of Computer Vision*, 41:33–59, 2001.

[96] C.-T. Li et R. Chiao. Unsupervised Texture Segmentation Using Multiresolution Hybrid Genetic Algorithm. Dans *10th IEEE International Conference on Image Processing (ICIP'2003)*, Barcelona, Spain, Septembre 2003.

[97] J. Liu et Y. H. Yang. Multiresolution Color Image Segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(7):689–700, 1994.

[98] L. Liu et S. Sclaroff. Deformable shape detection and description via model-based region grouping. *IEEE Trans. Pattern Anal. Machine Intell.*, 23(5):475–489, 2001.

[99] A. L. Maffett et C. C. Wackerman. The Modified Beta Density Function as a Model For Synthetic Aperture Radar Clutter Statistics. *IEEE Trans. Geosci. Remote Sensing*, 29(2):277–283, 1991.

[100] J. Malik, S. Belongie, T. Leung, et J. Shi. Contour and Texture Analysis for Image Segmentation. *International Journal of Computer Vision*, 43(1):7–27, 2001.

[101] S. G. Mallat. A theory of multiresolution signal decomposition. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(7):674–693, 1989.

[102] S. G. Mallat. Multiresolution approximations and wavelet orthogonal bases of $L^2(\mathbb{R}^2)$. *Trans. Americ. Math. Soc.*, 315(1):69–87, 1989.

[103] J. Maroquin, S. Mitter, et T. Poggio. Probabilistic solution of ill-posed problems in computation vision. *Journal of the American Statistical Association*, 82(397):76–89, 1987.

[104] D. Marr. *Vision*. W. H. Freeman and Company, 1982.

[105] D. Martin, C. Fowlkes, D. Tal, et J. Malik. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. Dans *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[106] N. S. Matloff. *Probability Modeling and Computer Simulation*. PWS-KENT Publishing Company, 1988.

[107] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, et E. Teller. Equations of state calculations by fast computing machines. *J. Chem. Phys.*, 21:335–341, 1953.

[108] M. Mignotte, C. Collet, P. Pérez, et P. Bouthemy. Hybrid genetic optimization and statistical model-based approach for the classification of shadow shapes

in sonar imagery. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(2):129–141, 2000.

[109] M. Mignotte, C. Collet, P. Pérez, et P. Bouthemy. Sonar image segmentation using an unsupervised hierarchical MRF model. *IEEE Trans. Image Processing*, 9(7):1216–1231, 2000.

[110] M. Mignotte, J. Meunier, et J.-C. Tardif. Endocardial boundary estimation and tracking in echocardiographic images using deformable templates and markov random fields. *Pattern Analysis and Applications*, 4(4):256–271, 2001.

[111] A. Mohammad-Djafari. Probabilistic methods for data fusion. Dans *Maximum entropy and Bayesian methods (Boise, ID, 1997)*, volume 98, pages 57–69, 1998.

[112] P. Del Moral. *Genealogical and interacting particle approximations*. Springer New York, Series: Probability and Applications, 2004.

[113] P. Del Moral et L. Miclo. On the Convergence and the Applications of the Generalized Simulated Annealing. *SIAM Journal on Control and Optimization*, 37(4):1222–1250, 1999.

[114] M. Moshfeghi, S. Ranganath, et K. Nawyn. Three-dimensional elastic matching of volumes. *IEEE Trans. Pattern Anal. Machine Intell.*, 3(2):128–138, 1994.

[115] D. B. Mumford et B. Gidas. Stochastic models for generic images. *Quarterly Journal of Applied Mathematics*, LIX(1):85–111, 2000.

[116] D. K. Panjwani et G. Healey. Markov Random Field Models for Unsupervised Segmentation of Textured Color Images. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(10):939–954, 1995.

[117] A. Peng et W. Pieczynski. Adaptive mixture estimation and unsupervised local Bayesian image segmentation. *CVGIP: Graphical Models and Image Processing*, 57(5):389–399, 1995.

[118] P. Pérez. *Champs markoviens et anaylse multirésolution de l'image: application à l'analyse du mouvement*. PhD thesis, Université de Rennes 1, IRISA, 1993.

[119] W. Pieczynski. Champs de Markov cachés et estimation conditionnelle itérative. *Revue Traitement Du Signal*, 11(2):141–153, 1994.

[120] W. Pieczynski, J. Bouvrais, et C. Michel. Estimation of generalized mixture in the case of correlated sensors. *IEEE Trans. Image Processing*, 9(2):308–311, 2000.

[121] Y. Rabinovich et A. Wigderson. Techniques for bounding the convergence rate of genetic algorithms. *Random Structures Algorithms*, 14(2):111–138, 1999.

[122] C. Regazzoni, F. Arduini, et G. Vernazza. A multilevel HMRF-based approach to image segmentation and restoration. *Signal Processing*, 34(1):43–67, 1993.

[123] S. Richardson et P. J. Green. On Bayesian analysis of mixtures with an unkown number of components. *Journal of the Royal Statistical Society*, 59(4):731–792, 1997.

[124] F. Salzenstein et W. Pieczynski. Unsupervised Bayesian segmentation using hidden markovian fields. Dans *Proceedings ICASSP'95 International Conference on Acoustics, Speech, and Signal Processing, Detroit, MI*, volume 4, pages 2411–2414, May 1995.

[125] F. Salzenstein et W. Pieczynski. Parameter Estimation in hidden fuzzy Markov

random fields and image segmentation. *CVGIP: Graphical Models and Image Processing*, 59(4):205–220, 1997.

[126] A. Sarkar, A. Banerjee, N. Banerjee, S. Brahma, B. Kartkeyan, M. Chakraborty, et K. L. Majumder. Landcover Classification in MRF Context Using Dempster-Shafer Fusion for Multisensor Imagery. *IEEE Trans. Image Processing*, 14(5):634–645, 2005.

[127] G. Schwartz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

[128] J. Shi et J. Malik. Normalized Cuts and Image Segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(8), Août 2000.

[129] M. S. Silverman, D. H. Grosof, R. L. De Valois, et S. D. Elfar. Spatial-frequency organization in primate striate cortex. *Proc. Natl. Acad. Sci. U.S.A.*, 86, 1989.

[130] L. H. Staib et J. S. Duncan. Boundary finding with parametric deformable models. *IEEE Trans. Pattern Anal. Machine Intell.*, 14(11):1061–1075, 1992.

[131] M. Stephens. Bayesian Analysis of Mixture Models with an Unknown Number of Components - An Alternative to Reversible Jump Methods. *The Annals of Statistics*, 28(1):40–74, 2000.

[132] G. Storvik. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(10):976–986, 1994.

[133] R. H. Swendsen et J. S. Wang. Nonuniversal critical dynamics in Monte Carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[134] M. E. Tipping et C. M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2):443–482, 1999.

[135] D. M. Titterington, A. F. Smith, et U. E. Makov. *Statistical Analysis of Finite Mixture Distributions*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, 1992.

[136] A. Trouvé. Partially parallel simulated annealing: Low and high temperature approach to the invariant measure. Dans *Proceedings Volume of the US-French Workshop on Applied Stochastic Analysis (Rutgers University, 29 April-2 May 1991)*, volume 177 de *Lecture Notes in Control and Infor. Sci.* Springer-Verlag, New-York, 1992.

[137] A. Trouvé. Cycle decomposition and Simulated Annealing. *SIAM J. Control Optim.*, 34(3):966–986, 1996.

[138] A. Trouvé. Rough large deviation estimates for the optimal convergence speed exponent of generalized simulated annealing algorithm. *Ann. Inst. Henri Poincaré Probab. Statist.*, 32, 1996.

[139] Z. W. Tu et S. C. Zhu. Image Segmentation by Data-Driven Markov Chain Monte Carlo. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(5):657–673, 2002.

[140] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Trans. Image Processing*, 4(11):1549–1560, 1995.

[141] O. Viveros-Cancino, X. Descombes, J. Zerubia, et N. Baghdadi. Fusion of Radiometry and Textural Information for SIR-C Image Classification. Dans *9th IEEE International Conference on Image Processing ICIP' 2002, Rochester, USA*, September 2002.

[142] L. Younes. Parametric inference for imperfectly observed Gibbsian fields. volume 82 de *Probability Theory and Related Fields*, pages 625–645. Springer-Verlag, 1989.

[143] S. Y. Yuen et B. K. S. Cheung. Bounds for Probability of Success of Classical Genetic Algorithm based on Vose-Liepens Model. *Les Cahiers du GERAD*, ISSN: 0711-2440, 2003.

[144] Y. Zhong et A. K. Jain. Object localization using color, texture and shape. *Pattern Recognition*, 33:671–684, 2000.

[145] S. C. Zhu, C. Guo, Y. Wang, et Z. Xu. What are Textons. *International Journal on Computer Vision*, 62(1):121–143, 2005.

[146] S. C. Zhu, X. W. Liu, et Y. N. Wu. Exploring Texture Ensembles by Efficient Markov Chain Monte Carlo - Toward a Trichromacy Theory of Texture. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(6):554–569, 2000.

[147] S. C. Zhu, Y. N. Wu, et D. Mumford. Minimax Entropy Principle and Its Application to Texture Modeling. *Neural Computation*, 9(8), 1997.

[148] S. C. Zhu, Y. N. Wu, et D. Mumford. Filters, Random fields And Maximum Entropy (FRAME). *International Journal of Computer Vision*, 27(2), 1998.

[149] S. C. Zhu, Y. N. Wu, et D. Mumford. Equivalence of Julesz Ensembles and FRAME Models. *International Journal on Computer Vision*, 38(3):247–265, 2000.

Annexe A

PREUVE DE CONVERGENCE DU RECUIT SIMULÉ DANS UN CAS PARTICULIER

Nous nous replaçons dans le contexte de la section 3.4 et du chapitre 2. Soit donc $q(\theta^{[t+1]} \mid \theta^{[t]}, y)$ un noyau de transition irréductible. Nous faisons l'hypothèse que

$$q(\theta' \mid \theta, y) > 0; \tag{A.1}$$

$$P(\theta \mid y) > 0, \tag{A.2}$$

pout tout θ, θ'.

Nous considérons l'ensemble E des valeurs de θ. L'ensemble est fini après discrétisation. Nous posons $\pi(\theta, \theta') = q(\theta' \mid \theta, y)$. Nous définissons une fonction de coût de communication $V_1 : E \times E \to [0, \infty]$ ainsi :

$$V_1(\theta, \theta') = \begin{cases} 0, & \text{si } \frac{P(\theta' \mid y)}{P(\theta \mid y)} \geq 1; \\ -\log\left\{\frac{P(\theta' \mid y)}{P(\theta \mid y)}\right\}, & \text{sinon.} \end{cases} \tag{A.3}$$

De plus, prenons κ suffisement grand pour que $\kappa \geq q(\theta \mid \theta, y)^{-1}$ pour tout θ. Finalement, posons

$$q_T(\theta, \theta') = \begin{cases} q(\theta' \mid \theta, y) \min\left\{1, \frac{P(\theta' \mid y)}{P(\theta \mid y)}\right\}^{1/T}, & \text{si } \theta' \neq \theta; \\ q(\theta \mid \theta, y) + \sum_{\theta' \neq \theta} q(\theta' \mid \theta, y)\Big(1 - \min\left\{1, \frac{P(\theta' \mid y)}{P(\theta \mid y)}\right\}^{1/T}\Big), & \text{si } \theta' = \theta. \end{cases} \tag{A.4}$$

Lemme 1

L'algorithme de la section 3.4 est un recuit simulé généralisé avec paramètres partiels $(E, \pi, \kappa, V_1, (q_T)_{T>0})$.

Preuve: Tout d'abord, il peut être facilement vu que $q_{T(t)}(\theta, \theta') = P(X_{t+1} = \theta' \,|\, X_t = \theta)$ en utilisant la description de l'algorithme à la section 3.4.

Soit maintenant $\theta' \neq \theta$ tel que $\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)} \geq 1$. Alors, $q_T(\theta, \theta') = q(\theta' \mid \theta, y)$ et $V_1(\theta, \theta') = 0$. Nous obtenons donc $q_T(\theta, \theta') = \pi(\theta, \theta') e^{-V_1(\theta, \theta')/T}$.

Soit $\theta' = \theta$. Alors, $V_1(\theta, \theta) = 0$. De plus, on a $q(\theta|\theta, y) \leq q_T(\theta, \theta) \leq 1 \leq \kappa q(\theta|\theta, y)$. Donc, $\frac{1}{\kappa}\pi(\theta, \theta) e^{-V_1(\theta,\theta)/T} \leq q_T(\theta, \theta) \leq \kappa \pi(\theta, \theta) e^{-V_1(\theta,\theta)/T}$.

Finalement, soit $\theta' \neq \theta$ tel que $\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)} < 1$. Alors, $V_1(\theta, \theta') = -\log\{\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)}\}$. Il en découle que $q_T(\theta, \theta') = q(\theta' \mid \theta, y)\Big\{\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)}\Big\}^{1/T} = q(\theta' \mid \theta, y) e^{-V_1(\theta, \theta')/T}$. □

Proposition 1

L'ensemble $\mathcal{W}_*$ des minima de la fonction d'énergie virtuelle $W(\theta)$ coincide avec l'ensemble des maxima de la distribution $P(\theta \,|\, y)$.

Preuve: Soit θ_{MAP} un mode de $P(\theta \mid y)$. Nous considérons le θ_{MAP}-graphe formé des arcs $\theta \to \theta_{\text{MAP}}$ pour $\theta \neq \theta_{\text{MAP}}$. Si $\theta \to \theta_{\text{MAP}}$ est un de ces arcs, nous avons: $V(\theta, \theta_{\text{MAP}}) \leq V_1(\theta, \theta_{\text{MAP}}) = 0$, car nous avons forcément que $\frac{P(\theta_{\text{MAP}} \,|\, y)}{P(\theta \,|\, y)} \geq 1$. Dès lors, $W(\theta_{\text{MAP}}) \leq \sum_{\theta \to \theta_{\text{MAP}}} V(\theta, \theta_{\text{MAP}}) \leq 0$. Par ailleurs, il est toujours vrai que $W(\theta) \geq 0$ pour un RSG quelconque, car la fonction V_1 prend ses valeurs dans $[0, \infty]$. Dès lors, $W(\theta_{\text{MAP}}) = 0$, et donc $\theta_{\text{MAP}} \in \mathcal{W}_*$.

Soit maintenant, θ_* qui n'est pas un mode de $P(\theta \mid y)$. Soit un θ_*-graphe g tel que $W(\theta_*) = \sum_{\theta \to \theta' \in g} V(\theta, \theta')$. Comme $\frac{P(\theta_* \,|\, y)}{P(\theta_{\text{MAP}} \,|\, y)} < 1$, il existe au moins un arc $\theta \to \theta'$ le long du chemin qui mène de θ_{MAP} à θ_* dans g, tel que $\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)} < 1$. Soit maintenant $\theta_0 = \theta, \theta_1, ..., \theta_r = \theta'$ tel que $V(\theta, \theta') = \sum_{k=0}^{r-1} V_1(\theta_k, \theta_{k+1})$. Il y a au moins un indice k tel que $\frac{P(\theta_{k+1} \,|\, y)}{P(\theta_k \,|\, y)} < 1$, puisque $\frac{P(\theta' \,|\, y)}{P(\theta \,|\, y)} < 1$. Mais alors, $V_1(\theta_k, \theta_{k+1}) = -\log\{\frac{P(\theta_{k+1} \,|\, y)}{P(\theta_k \,|\, y)}\} > 0$. D'où, $V(\theta, \theta') > 0$ et par suite, $W(\theta_*) > 0$. Il en découle que $\theta_* \notin \mathcal{W}_*$. □

Proposition 2

La hauteur critique H_1 s'annule.

Preuve: Nous suivons le même argument qu'à la section 4.6. Soit π un cycle tel que

$\pi \cap \mathcal{W}_* = \emptyset$. Prenons $\theta_{\text{MAP}} \in \mathcal{W}_*$ et $\theta \in \pi$ tel que $W(\pi) = W(\theta)$. Du Lemme 4 de la section 4.6, nous déduisons que $H_e(\pi) \leq V(\theta, \theta_{\text{MAP}})$. Or, comme $q(\theta_{\text{MAP}} \mid \theta, y) > 0$ par hypothèse, nous obtenons que $V(\theta, \theta_{\text{MAP}}) = 0$. Dès lors, $H_e(\pi) = 0$. Il en découle que $H_1 = \max_{\pi \cap \mathcal{W}_* = \emptyset} H_e(\pi) = 0$. $\square$

En utilisant le théorème de Trouvé (section 2.4), les deux propositions ci-dessus montrent que l'algorithme converge asympotiquement vers un des modes de la distribution *a posteriori* $P(\theta \mid y)$, avec probabilité 1, pourvu que $\lim_{t \to \infty} T(t) = 0$. En outre, il n'est pas nécessaire de prendre $T(t)$ de la forme $\tau / \log(t + 1)$. C'est le cas exceptionnel d'une hauteur critique nulle. Lorsque $q(\theta' \mid \theta, y)$ s'annule, il n'est plus vrai que $H_1 = 0$. Dans ce cas, il faut supposer la condition de réversibilité faible de Hajek pour obtenir la convergence du RS vers un MAP; voir [137] section 2.5.2.

Annexe B

MÉLANGE DE NOYAUX GAUSSIENS À NOMBRE VARIABLE

B.1 Introduction

Dans cet annexe, nous comparons notre algorithme ESE avec le RJMCMC dans le cadre d'un modèle de mélange de noyaux gaussiens en nombre variable. Nous supposerons les données indépendantes, ce qui permet de concevoir facilement un algorithme de type RJMCMC. Par contre, dans le cadre du modèle du chapitre 6, il est beaucoup plus difficile de mettre en oeuvre le RJMCMC, dû au modèle *a priori* markovien sur le processus des régions.

B.2 Modèle considéré

Nous nous replaçons dans le contexte de la section 3.3.3. Nous considèrons pour chaque valeur de $k \in \{1, 2, ..., K\}$, un mélange de k noyaux gaussiens de dimension d:

$$P(y \mid k, \theta^{(k)}) = \sum_{i=1}^{k} \pi_i \mathcal{N}(y; \mu_i, \Sigma_i), \tag{B.1}$$

où $\sum_{i=1}^{k} \pi_i = 1$ et $\pi_i \geq 0$, et $\theta^{(k)} = (\pi_1, ..., \pi_k, \mu_1, ..., \mu_k, \Sigma_1, ..., \Sigma_k)$. Nous considérons une densité *a priori* uniforme sur le nombre de noyaux gaussiens k:

$$P(k) = \frac{1}{K}. \tag{B.2}$$

Nous posons une densité *a priori* de Dirichlet sur les proportions du mélange :

$$P(\pi_1, ..., \pi_k \mid k) = \mathcal{D}(\pi_1, ..., \pi_k; A_0, \alpha_1, ..., \alpha_k), \tag{B.3}$$

où $A_0 = k$ et $\alpha_1 = ... = \alpha_k = 1/k$. Nous considérons une densité *a priori* Wishart inversée pour chaque matrice de covariance :

$$P(\Sigma_i \mid k) = \mathcal{IW}(\Sigma_i; \Lambda_0, \nu_0), \tag{B.4}$$

où $\nu_0 = d + 2$ et Λ_0 est une matrice symétrique positive-définie. La loi *a priori* sur chaque moyenne est alors donnée par une gaussienne :

$$P(\mu_i \mid k, \Sigma_i) = \mathcal{N}(\mu_i; \mu_0, \frac{1}{k_0}\Sigma_i), \tag{B.5}$$

où $k_0 = 0,01$ et μ_0 est un vecteur de dimension d. Nous obtenons l'*a priori* des paramètres conditionnelle au nombre de noyaux :

$$P(\theta^{(k)} \mid k) = k!P(\pi_1, ..., \pi_k \mid k) \prod_{i=1}^{k} P(\mu_i \mid k, \Sigma_i)P(\Sigma_i \mid k). \tag{B.6}$$

(La factorielle tient compte du nombre de permutations des k noyaux.)

Nous considérons également une variable aléatoire cachée c qui prend ses valeurs dans l'ensemble fini $\{e_1, ..., e_k\}$ afin d'indiquer quel noyau gaussien est choisi. Nous obtenons le système hiérarchique suivant :

$$P(y \mid c = e_i, k, \theta^{(k)}) = \mathcal{N}(y; \mu_i, \Sigma_i) \tag{B.7}$$

$$P(c = e_i \mid k, \theta^{(k)}) = \pi_i. \tag{B.8}$$

B.3 Simulation selon le modèle a posteriori

Soit maintenant $y_1, ..., y_N$ un échantillon indépendant identiquement distribué du vecteur aléatoire y. La distribution jointe de $(y_1, ..., y_N, c_1, ..., c_N, k, \theta^{(k)})$ est donnée par :

$$\prod_{l=1}^{N} P(y_l \mid c_l, k, \theta^{(k)}) P(c_l \mid k, \theta^{(k)}) \times P(\theta^{(k)} \mid k) P(k). \tag{B.9}$$

Notre objectif est de simuler $(c_1, ..., c_N, k, \theta^{(k)})$ selon la distribution a posteriori conditionnelle à l'échantillon.

Fixons k pour l'instant. Nous considérons l'échantillonneur de Gibbs suivant :

1. Pour $l = 1, ..., N$, tirer $c_l = e_i$ avec probabilité $\omega_i = \frac{\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi_i}{\sum_{i=1}^{k}\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi_i}$, où $i = 1, ..., k$.

2. Simuler les proportions $(\pi_1, ..., \pi_k)$ selon la distribution *a posteriori* $\mathcal{D}\left(N + k; \frac{N_i+1}{N+k}\right)$, où N_i est le nombre d'étiquettes c_l égales à e_i.

3. Pour $i = 1, ..., k$, simuler la matrice de covariance Σ_i selon la distribution *a posteriori* $\mathcal{IW}(\tilde{\Lambda}, \tilde{\nu})$, où $\tilde{\Lambda} = \Lambda_0 + N_i S + \frac{N_i k_0}{N_i + k_0}(\bar{y} - \mu_0)(\bar{y} - \mu_0)^t$ et $\tilde{\nu} = N_i + \nu_0$, avec $\bar{y}$ et S la moyenne et la matrice de covariance empiriques, respectivement, sur l'ensemble $\{y_l : c_l = e_i\}$. Simuler ensuite la moyenne μ_i selon la distribution *a posteriori* $\mathcal{N}\left(\tilde{\mu}, \frac{1}{\tilde{k}}\Sigma_i\right)$, où $\tilde{\mu} = \bar{y} - \frac{k_0}{N_i + k_0}(\bar{y} - \mu_0)$ et $\tilde{k} = N_i + k_0$. Si l'ensemble $\{y_l : c_l = e_i\}$ est vide, simuler Σ_i selon la distribution *a priori* $\mathcal{IW}(\Lambda_0, \nu_0)$; puis, simuler μ_i selon la distribution *a priori* $\mathcal{N}(\mu_0, \frac{1}{k_0}\Sigma_i)$.

Cet échantillonneur de Gibbs permet d'explorer l'espace des paramètres à l'intérieur du sous-espace de dimension fixe $d_k = k(d + 1)(d + 2)/2 - 1$. Il s'agit maintenant de construire deux opérateurs supplémentaires : un pour passer de k noyaux à $k + 1$ noyaux, et l'autre pour faire le trajet inverse.

Pour le premier opérateur (augmentation de la dimension), soit k le nombre de noyaux et $\theta^{(k)} = (\pi_1, ..., \pi_k, \mu_1, ..., \mu_k, \Sigma_1, ..., \Sigma_k)$. Nous posons $u^{(1)} = (u, \mu_{k+1}, \Sigma_{k+1})$. Pour un $i \in \{1, ..., k\}$ fixé, nous considérons le difféomorphisme φ qui envoie $(\theta^{(k)}, u^{(1)})$ au vecteur de paramètres $\theta^{(k+1)} = (\pi_1, ..., (1 -$

$u)\pi_i, ..., \pi_k, u\pi_i, \mu_1, ..., \mu_{k+1}, \Sigma_1, ..., \Sigma_{k+1})$. Le jacobien de φ est égal à π_i. Nous proposons donc ce qui suit :

1. Simuler u selon une distribution beta $\mathcal{B}e(1, k)$.

2. Simuler Σ_{k+1} selon la densité *a priori* $\mathcal{IW}(\Lambda_0, \nu_0)$.

3. Simuler μ_{k+1} selon la densité *a priori* $\mathcal{N}(\mu_0, \frac{1}{k_0}\Sigma_{k+1})$.

4. Choisir $i_1 \in \{1, ..., k\}$ selon une loi uniforme. Poser $i_2 = k + 1$.

5. Poser $\pi'_{i_1} = (1 - u)\pi_{i_1}$, $\pi'_{i_2} = u\pi_{i_1}$, et $\pi'_j = \pi_j$ pour $j \neq i_1, i_2$.

6. Pour $l = 1, ..., N$, tirer $c'_l = e_i$ avec probabilité $\omega_i = \frac{\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i}{\sum_{i=1}^{k+1}\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i}$, où $i = 1, ..., k + 1$.

7. Accepter le saut en dimension donné par le passage de $(c_1, ..., c_N, k, \theta^{(k)})$ au vecteur $(c'_1, ..., c'_N, k + 1, \theta^{(k+1)})$ avec probabilité $\alpha = \min(1, \rho)$ où

$$\rho = \frac{\prod_{l=1}^{N} P(y_l \mid k + 1, \theta^{(k+1)})}{\prod_{l=1}^{N} P(y_l \mid k, \theta^{(k)})} \frac{\pi_{i_1}}{(1 - u)^{k-1}}. \tag{B.10}$$

Avant d'expliquer la valeur de ρ ci-dessus, nous présentons le deuxième opérateur (diminution de la dimension) :

1. Choisir (i_1, i_2) selon une loi uniforme sur l'ensemble des paires ordonnées d'éléments distincts de $\{1, ..., k\}$.

2. Poser $\pi'_{i_1} = \pi_{i_1} + \pi_{i_2}$, $\pi'_{i_2} = 0$, et $\pi'_j = \pi_j$ pour $j \neq i_1, i_2$. Poser $u = \pi_{i_2}/\pi'_{i_1}$.

3. Pour $j = i_2, ..., k - 1$, poser $\pi'_j \leftarrow \pi'_{j+1}$.

4. Pour $l = 1, ..., N$, tirer $c'_l = e_i$ avec probabilité $\omega_i = \frac{\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i}{\sum_{i=1}^{k}\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i}$, où $i = 1, ..., k-1$.

5. Accepter le saut en dimension donné par le passage de $(c_1, ..., c_N, k, \theta^{(k)})$ au vecteur $(c'_1, ..., c'_N, k-1, \theta^{(k-1)})$ avec probabilité $\alpha = \min(1, \rho)$ où

$$\rho = \frac{\prod_{l=1}^{N} P(y_l \mid k-1, \theta^{(k-1)})}{\prod_{l=1}^{N} P(y_l \mid k, \theta^{(k)})} \frac{(1-u)^{k-2}}{(\pi_{i_1} + \pi_{i_2})}. \tag{B.11}$$

Nous proposons chacun des deux opérateurs avec une même probabilité $p/2$. L'échantillonneur de Gibbs est alors proposé avec probabilité $1 - p$. Dans nos tests, nous avons pris $p = 1/5$. Le noyau de transition du RJMCMC $q(k', \theta^{(k')} \mid k, \theta^{(k)}, y_1, ..., y_N)$ se résume ainsi :

1. Avec probabilité p faire:

   (a) Si $k = 1$, tenter l'augmentation de dimension.

   (b) Si $k = K$, tenter la diminution de dimension.

   (c) Si $1 < k < K$, tenter l'augmentation de dimension avec probabilité $1/2$, ou sinon, tenter la diminution de dimension.

2. Sinon (avec probabilité $1 - p$), faire l'échantillonneur de Gibbs.

Nous présentons maintenant le calcul menant à la valeur de ρ du premier opérateur. Nous avons d'abord

$$\begin{aligned}
&\frac{P(c'_1,...,c'_N,k+1,\theta^{(k+1)} \mid y_1,...,y_N)}{P(c_1,...,c_N,k,\theta^{(k)} \mid y_1,...,y_N)} \\
&= \frac{P(y_1,...,y_N,c'_1,...,c'_N,k+1,\theta^{(k+1)})}{P(y_1,...,y_N,c_1,...,c_N,k,\theta^{(k)})} \\
&= \frac{\prod_{l=1}^N P(y_l,c'_l \mid k+1,\theta^{(k+1)})}{\prod_{l=1}^N P(y_l,c_l \mid k,\theta^{(k)})} \frac{P(\theta^{(k+1)} \mid k+1)}{P(\theta^{(k)} \mid k)} \frac{P(k+1)}{P(k)} \\
&= \frac{\prod_{l=1}^N P(y_l,c'_l \mid k+1,\theta^{(k+1)})}{\prod_{l=1}^N P(y_l,c_l \mid k,\theta^{(k)})} \frac{(k+1)!P(\pi'_1,...,\pi'_{k+1} \mid k+1)}{k!P(\pi_1,...,\pi_k \mid k)} \\
&\times P(\mu_{k+1},\Sigma_{k+1} \mid k+1)\frac{P(k+1)}{P(k)}.
\end{aligned} \tag{B.12}$$

Ensuite, nous observons que la distribution de proposition est donnée par

$$\begin{aligned}
&q_1(c'_1,...,c'_N,u^{(1)} \mid \theta^{(k)}) \\
&= \prod_{l=1}^N \frac{P(y_l,c'_l \mid k+1,\theta^{(k+1)})}{P(y_l \mid k+1,\theta^{(k+1)})} P(\mu_{k+1},\Sigma_{k+1} \mid k+1)\mathcal{B}e(u;1,k),
\end{aligned} \tag{B.13}$$

car

$$\omega_i = \frac{\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i}{\sum_{i=1}^{k+1}\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi'_i} = \frac{P(y_l,c'_l \mid k+1,\theta^{(k+1)})}{P(y_l \mid k+1,\theta^{(k+1)})}. \tag{B.14}$$

De même, nous avons

$$q_2(c_1,...,c_N \mid \theta^{(k+1)}) = \prod_{l=1}^N \frac{P(y_l,c_l \mid k,\theta^{(k)})}{P(y_l \mid k,\theta^{(k)})}. \tag{B.15}$$

Notons que $j(k+1,\theta^{(k+1)})/j(k,\theta^{(k)}) = \frac{1}{k+1}$. Nous déduisons donc pour valeur de ρ :

$$\frac{\prod_{l=1}^N P(y_l \mid k+1,\theta^{(k+1)})}{\prod_{l=1}^N P(y_l \mid k,\theta^{(k)})} \frac{(k+1)!P(\pi'_1,...,\pi'_{k+1} \mid k+1)P(k+1)}{k!P(\pi_1,...,\pi_k \mid k)P(k)(k+1)} \frac{\pi_{i_1}}{\mathcal{B}e(u;1,k)}. \tag{B.16}$$

Après simplifications, nous obtenons la valeur de l'équation (B.10). La valeur de l'équation (B.11) s'obtient semblablement.

Finalement, l'algorithme est initialisé de la façon suivante :

1. Poser $k = K$.

2. Pour $l = 1, ..., N$, tirer $c_l = e_i$ avec probabilité $1/K$, où $i = 1, ..., K$.

3. Simuler les proportions $(\pi_1, ..., \pi_K)$ selon la distribution *a posteriori* $\mathcal{D}\left(N + K; \frac{N_i+1}{N+K}\right)$, où N_i est le nombre d'étiquettes c_l égales à e_i.

4. Pour $i = 1, ..., K$, simuler la matrice de covariance Σ_i selon la distribution *a posteriori* $\mathcal{IW}(\tilde{\Lambda}, \tilde{\nu})$, où $\tilde{\Lambda} = \Lambda_0 + N_i S + \frac{N_i k_0}{N_i+k_0}(\bar{y} - \mu_0)(\bar{y} - \mu_0)^t$ et $\tilde{\nu} = N_i + \nu_0$, avec $\bar{y}$ et S la moyenne et la matrice de covariance empiriques, respectivement, sur l'ensemble $\{y_l : c_l = e_i\}$. Simuler ensuite la moyenne μ_i selon la distribution *a posteriori* $\mathcal{N}\left(\tilde{\mu}, \frac{1}{\tilde{k}}\Sigma_i\right)$, où $\tilde{\mu} = \bar{y} - \frac{k_0}{N_i+k_0}(\bar{y} - \mu_0)$ et $\tilde{k} = N_i + k_0$.

B.4 Algorithme ESE

Voici le noyau d'exploration de l'ESE dans le cadre du mélange de distributions gaussiennes. Nous considérons un vecteur de K bits $v = (v_i)$ qui indique quels noyaux sont considérés : $v_i = 1$ si le noyau i est présent dans le mélange ; $v_i = 0$ sinon.

Engendrer n selon une distribution binomiale $b(0.1, 1)$. Répéter $n+1$ fois le noyau de transition suivant:

1. Modifier chaque bit v_i avec probabilité $1/(2K)$. Si tous les bits valent 0, en mettre un au hasard égal à 1.

2. Pour $l = 1, ..., N$, tirer $c_l = e_i$ avec probabilité $\omega_i = \frac{\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi_i v_i}{\sum_{i=1}^{k}\mathcal{N}(y_l;\mu_i,\Sigma_i)\pi_i v_i}$, où $i = 1, ..., K$. Notez que $\omega_i = 0$ si $v_i = 0$.

3. Soit $k = |v|$. Simuler les proportions $(\pi_i; v_i = 1)$ selon la distribution *a posteriori* $\mathcal{D}\left(N + k; \frac{N_i+1}{N+k}\right)$, où N_i est le nombre d'étiquettes c_l égales à e_i.

4. Pour chaque i tel que $v_i = 1$, simuler la matrice de covariance Σ_i selon la distribution *a posteriori* $\mathcal{IW}(\tilde{\Lambda}, \tilde{\nu})$, où $\tilde{\Lambda} = \Lambda_0 + N_i S + \frac{N_i k_0}{N_i + k_0}(\bar{y} - \mu_0)(\bar{y} - \mu_0)^t$ et $\tilde{\nu} = N_i + \nu_0$, avec $\bar{y}$ et S la moyenne et la matrice de covariance empiriques, respectivement, sur l'ensemble $\{y_l \ : \ c_l = e_i\}$. Simuler ensuite la moyenne μ_i selon la distribution a posteriori $\mathcal{N}\left(\tilde{\mu}, \frac{1}{\tilde{k}}\Sigma_i\right)$, où $\tilde{\mu} = \bar{y} - \frac{k_0}{N_i + k_0}(\bar{y} - \mu_0)$ et $\tilde{k} = N_i + k_0$. Si l'ensemble $\{y_l : c_l = e_i\}$ est vide, simuler Σ_i selon la distribution *a priori* $\mathcal{IW}(\Lambda_0, \nu_0)$; puis, simuler μ_i selon la distribution *a priori* $\mathcal{N}(\mu_0, \frac{1}{k_0}\Sigma_i)$.

L'initialisation est comme à la fin de la section B.3. Se référer à la section 2.3 pour la description de l'algorithme E/S. Nous avons choisit une population de $m = 6$ solutions et une température initale de $\tau = 3$.

B.5 Algorithme RS

Une étape du recuit simulé de la section 3.4 pour le calcul du MAP s'exprime ainsi :

1. **Proposition :** Engendrer n selon une distribution binomiale $b\left(\frac{0.1}{(K-1)}, K-1\right)$. Répéter $n+1$ fois le noyau de transition du RJMCMC. Soit $(k', \theta^{(k')})$ la valeur résultante.

2. **Disposition :** Accepter la proposition avec probabilité

$$\alpha = \min\left\{1, \frac{P(k', \theta^{(k')} \mid y_1, ..., y_N)}{P(k, \theta^{(k)} \mid y_1, ..., y_N)}\right\}^{1/T}. \tag{B.17}$$

   Sinon, conserver la valeur $(k, \theta^{(k)})$.

Notez que le noyau de transition est positif ; c'est-à-dire, $q(k', \theta' \mid k, \theta, y) > 0$ pour tout (k, θ) et (k', θ'), grâce à la répétition du noyau de transition du RJMCMC. Dans ce cas, les résultats présentés à l'annexe A s'appliquent. En outre, la convergence est assurée dès que $\lim_{t\to\infty} T(t) = 0$. Dans nos tests, nous avons choisit pour température $T(t) = 3/\log(t+2)$.

Nous pouvons également construire le RS à l'aide du noyau de transition de l'ESE

1. **Proposition:** Engendrer $(k', \theta^{(k')})$ selon le noyau de transition de l'ESE.

2. **Disposition:** Accepter la proposition avec probabilité

$$\alpha = \min\Big\{1, \frac{P(k', \theta^{(k')} \mid y_1, ..., y_N)}{P(k, \theta^{(k)} \mid y_1, ..., y_N)}\Big\}^{1/T}. \tag{B.18}$$

   Sinon, conserver la valeur $(k, \theta^{(k)})$.

Encore une fois, les résultats présentés à l'annexe A s'appliquent ici.

B.6 Tests de comparaison

Soit d la dimension du vecteur des attributs (c'est-à-dire, la dimension de chaque noyau gaussien). Nous simulons un modèle de mélange de noyaux gaussiens par la méthode suivante :

1. Simuler k entre 1 et $K = 20$ selon une loi uniforme.

2. Simuler les proportions $\pi_1, ..., \pi_k$ selon la loi *a priori* de Dirichlet $\mathcal{D}\left(k; \frac{1}{k}, ..., \frac{1}{k}\right)$

3. Pour $i = 1, ..., k$, simuler la matrice de covariance Σ_i selon la distribution *a priori* $\mathcal{IW}(I_d, d+2)$. Simuler ensuite la moyenne μ_i selon la distribution *a priori* $\mathcal{N}\left(0, \frac{1}{K_0}\Sigma_i\right)$.

Une fois le modèle simulé, nous nous donnons un échantillon indépendant identiquement distribué de taille $N = 10000$ ainsi :

- Pour $l = 1, ..., N$, tirer $c_l = e_i$ avec probabilité π_i, où $i = 1, ..., k$. Ensuite, simuler y_l selon la distribution $\mathcal{N}(\mu_i, \Sigma_i)$.

d	(1): ESE vs RJMCMC	(2): ESE vs RS	(2): ESE vs RS (version 2)
1	$-0{,}029521$	$-0{,}027289$	$-0{,}013069$
5	$0{,}058135$	$0{,}111948$	$-0{,}094704$
10	$0{,}134349$	$0{,}055028$	$0{,}224760$
15	$0{,}013515$	$0{,}087585$	$0{,}118346$

Table B.1. (1): $\log\{P((k,\theta^{(k)})_{\text{ESE}} \,|\, y_1,...,y_N)/P((k,\theta^{(k)})_{\text{MCMC}} \,|\, y_1,...,y_N)\}$.
(2): $\log\{P((k,\theta^{(k)})_{\text{ESE}} \mid y_1,...,y_N)/P((k,\theta^{(k)})_{\text{RS}} \mid y_1,...,y_N)\}$.
(3): $\log\{P((k,\theta^{(k)})_{\text{ESE}} \mid y_1,...,y_N)/P((k,\theta^{(k)})_{\text{RS}} \mid y_1,...,y_N)\}$ **(version 2). Une valeur positive indique que l'ESE performe mieux que la méthode comparée.**

Nous estimons ensuite les paramètres $(k,\theta^{(k)})_{\text{ESE}}$ avec la méthode ESE de la section B.4. Le nombre d'itérations est fixé à 1465 (ce nombre d'itérations provient des bornes en temps fini de la section 6.7). Nous procédons ensuite à une nouvelle estimation des paramètres $(k,\theta^{(k)})_{\text{MCMC}}$, à l'aide du RJMCMC de la section B.3 pour le même nombre d'explorations. Nous comparons les solutions obtenues à l'aide de la cote de Bayes : $\log\{P((k,\theta^{(k)})_{\text{ESE}} \,|\, y_1,...,y_N)/P((k,\theta^{(k)})_{\text{MCMC}} \,|\, y_1,...,y_N)\}$. Nous répétons l'expérience pour 40 seeds différents. De plus, nous faisons de même avec le recuit simulé de la section 3.4, ce qui donne la cote de Bayes $\log\{P((k,\theta^{(k)})_{\text{ESE}} \mid y_1,...,y_N)/P((k,\theta^{(k)})_{\text{RS}} \mid y_1,...,y_N)\}$.

Nous présentons les résultats obtenus lorsque $d = 1, 5, 10, 15$ au tableau B.1. Nous remarquons que l'ESE s'avère en moyenne plus efficace que le RJMCMC ou le RS. Il s'agit là d'un avantage en traitement d'images, car nous nous intéressons à des estimations satisfaisantes obtenues dans le moins de temps de calcul possible.

Annexe C

COMPARAISON DE L'ESE AVEC LE RS

Dans cet annexe, nous comparons notre algorithme ESE avec le RS dans le cadre du modèle présenté au chapitre 6.

Nous nous replaçons dans le contexte de la section 6.3. Rappelons qu'aux tableaux 6.9, 6.10 et 6.11 est présenté le noyau d'exploration de la procédure ESE. Le vecteur de paramètres θ est égal à $(x, \mu, \Sigma, \pi, \beta, v)$, tandis que le vecteur augmenté ψ est de la forme $(x, \hat{c}, \mu, \Sigma, \pi, \beta, v, L)$. Nous cherchons le minimum de la fonction $\bar{f}(\theta)$ de l'équation (6.35), ou encore le minimum de la fonction $f(\psi) = \bar{f}(\theta)$ définie au début de la section 6.3.2. Une étape du recuit simulé de la section 3.4 pour le calcul du MAP s'exprime ainsi :

1. **Proposition:** Engendrer ψ' à partir de la solution ψ selon le noyau de transition de l'ESE.

2. **Disposition:** Accepter la proposition avec probabilité

$$\alpha = \min\Big\{1, e^{-\{f(\psi')-f(\psi)\}/T}\Big\}. \tag{C.1}$$

   Sinon, conserver la valeur ψ.

Notez que le noyau d'exploration ne s'annule en aucun point, grâce à la répétition de l'échantillonneur de Gibbs tel qu'indiqué au tableau 6.9. Dès lors, les résultats présentés à l'annexe A s'appliquent. En outre, la convergence est assurée dès que $\lim_{t\to\infty} T(t) = 0$. Dans nos tests, nous avons choisit pour température $T(t) = 3/\log(t+2)$, c'est-à-dire la même que pour la procédure ESE du chapitre 6.

Nous estimons les paramètres ψ_{ESE} avec la méthode ESE de la section 6.3 sur les 100 images utilisées à la section 6.4. Le nombre d'itérations est fixé à 1465 (ce nombre d'itérations provient des bornes en temps fini de la section 6.7). Nous procédons ensuite à une nouvelle estimation des paramètres ψ_{RS}, à l'aide du RS pour le même nombre d'explorations. Nous comparons les solutions obtenues à l'aide de la cote de Bayes (en échelle logarithmique) : $-f(\psi_{\text{ESE}}) + f(\psi_{\text{RS}})$. Nous obtenons en moyenne une valeur de 0,199248. Nous concluons que l'ESE s'avère en moyenne plus efficace que le RS, du moins dans le cadre de nos tests.

Annexe D

COMPARAISON DE L'E/S AVEC L'ESE DANS UN CAS SIMPLE

D.1 Introduction

Dans cet annexe, nous comparons notre algorithme ESE avec l'algorithme E/S dans le cadre du calcul du mode d'une distribution qui est facile à simuler.

D.2 Fonction considérée

Nous considèrons une distribution définie par un mélange de 2 noyaux gaussiens de dimension d :

$$p(y) = \sum_{i=1}^{2} \pi_i \mathcal{N}(y; \mu_i, \Sigma_i), \tag{D.1}$$

où $\pi_1 = 1/3$, $\pi_2 = 2/3$, $\mu_{1,j} = -1/2$ pour $j = 1, ..., d$, $\mu_{2,j} = 1/2$ pour $j = 1, ..., d$, et $\Sigma_1 = \Sigma_2 = 1/10\, I$. Nous nous proposons d'évaluer le mode de cette distribution, c'est-à-dire de calculer le minimum (global) de la fonction à d variables définie par

$$f(y) = -\log p(y). \tag{D.2}$$

Nous considérons comme domaine de définition pour la fonction f un hyper-cube de la forme $[L_1, L_2]^d$. Dans nos test, nous avons considéré trois cas : l'hyper-cube $[0, 1]^d$, l'hyper-cube $[-1, 1]^d$, et l'hyper-cube $[-2, 2]^d$. Dans chacun de ces trois cas, le minimum global de la fonction est atteint lorsque $y = \mu_2$.

D.3 Algorithmes testés

Se référer à la section 2.3 pour la description de l'algorithme E/S. Voici le noyau d'exploration de l'algorithme E/S dans le cadre de nos tests. Soit $r = 1/10$, le rayon d'exploration.

1. Soit y_l la solution courante.

2. Pour $j = 1, ..., d$, proposer $y'_{l,j}$ selon une loi uniforme sur l'intervalle $[y_{l,j} - r(L_2 - L_1), y_{l,j} + r(L_2 - L_1)]$ jusqu'à ce que y'_l soit dans le domaine de définition.

Nous observons que le diamètre du graphe d'exploration est égal à $D = d/r$. Par le corollaire de la section 2.5, nous pouvons prendre une taille de population égale à $m = D + 1$ et une température initale égale à $\tau = D$.

Voici maintenant le noyau d'exploration de l'algorithme ESE dans le cadre de nos tests.

1. On ignore la solution courante y_l.

2. Échantillonner u selon une loi uniforme sur l'intervalle $[0, 1]$. Si $u \leq 1/3$, poser $i = 1$; sinon, poser $i = 2$ (choix du noyau gaussien).

3. Pour $j = 1, ..., d$, proposer $y'_{l,j}$ selon une loi gaussienne $\mathcal{N}(\mu_{i,j}, 1/10)$ jusqu'à ce que y'_l soit dans le domaine de définition.

Dans ce cas-ci, le diamètre du graphe d'exploration est égal à $D = 1$. Nous pouvons donc prendre la même taille de population et la même température initale que pour l'E/S, bien que $m = 2$ et $\tau = 1$ auraient suffit.

d	(1): ESE vs E/S	(2): ESE vs E/S	(3): ESE vs E/S
1	0	0	0
5	−0,0121458	0,13185	0,155788
10	−0,184299	0,267314	0,430125
15	−0,494015	0,296456	0,74845

Table D.1. (1) : $f(y_{\text{ES}}) - f(y_{\text{ESE}})$ **sur l'hyper-cube** $[0,1]^d$**. (2) :** $f(y_{\text{ES}}) - f(y_{\text{ESE}})$ **sur l'hyper-cube** $[-1,1]^d$**. (3) :** $f(y_{\text{ES}}) - f(y_{\text{ESE}})$ **sur l'hyper-cube** $[-2,2]^d$**. Une valeur positive indique que l'ESE performe mieux que la méthode comparée.**

D.4 Tests de comparaison

Nous estimons le mode y_{ES} de $p(y)$ avec l'algorithme E/S de la section D.3. Le nombre d'itérations est fixé à 1000. Nous procédons ensuite à une nouvelle estimation du mode y_{ESE} à l'aide de l'algorithme ESE de la section D.3 pour le même nombre d'explorations. Nous comparons les solutions obtenues en évaluant la moyenne de $f(y_{\text{ES}}) - f(y_{\text{ESE}})$ pour 30 essais différents.

Soit d la dimension du vecteur des attributs (c'est-à-dire, la dimension de chaque noyau gaussien). Nous présentons les résultats obtenus lorsque $d = 1, 5, 10, 15$ au tableau D.1. Nous remarquons que l'E/S s'avère en moyenne plus efficace que l'ESE dans le cas du domaine restreint $[0,1]^d$. Par contre, dans le cas des domaines plus étendus $[-1,1]^d$ et $[-2,2]^d$, l'ESE s'avère en moyenne plus efficace que l'E/S, et ce d'autant plus que la dimension du problème est élevée.