Modèle bayésien non paramétrique pour la segmentation jointe d'un ensemble d'images avec des classes partagées

Jessica Sodjo

UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

THÈSE

présentée pour obtenir le grade de

DOCTEUR

de l'Université de Bordeaux

Spécialité : Automatique, Productique, Signal et Image, Ingénierie cognitique

par

Jessica SODJO

Modèle bayésien non paramétrique pour la segmentation jointe d'un ensemble d'images avec des classes partagées

Encadrement :

François CARON
Nicolas DOBIGEON
Jean-François GIOVANNELLI
Audrey GIREMUS

Soutenue le

Membres du jury :

Faïcel CHAMROUKHI	Professeur, Université de Caen	*Rapporteur*
Xavier DESCOMBES	Directeur de recherche, INRIA Sophia Antipolis - Méditerranée	*Rapporteur*
Florence FORBES	Directrice de recherche, INRIA Grenoble Rhône-Alpes	*Présidente du jury*
François DESBOUVRIES	Professeur, Télécom SudParis	*Examinateur*
François CARON	Maître de conférences - HDR, Université d'Oxford	*Co-directeur*
Nicolas DOBIGEON	Professeur, INP-ENSEEIHT	*Co-directeur*
Audrey GIREMUS	Maître de conférences, Université de Bordeaux	*Co-encadrante*

Modèles bayésiens non paramétriques pour la segmentation jointe d'images

Ce travail porte sur la segmentation jointe d'un ensemble d'images dans un cadre bayésien. Le modèle proposé combine le processus de Dirichlet hiérarchique (HDP) et le champ de Potts. Ainsi, pour un groupe d'images, chacune est divisée en régions homogènes et les régions similaires entre images sont regroupées en classes. D'une part, grâce au HDP, il n'est pas nécessaire de définir *a priori* le nombre de régions par image et le nombre de classes, communes ou non. D'autre part, le champ de Potts assure une homogénéité spatiale. Les lois *a priori* et *a posteriori* en découlant sont complexes rendant impossible le calcul analytique d'estimateurs. Un algorithme de Gibbs est alors proposé pour générer des échantillons de la loi *a posteriori*. De plus, un algorithme de Swendsen-Wang généralisé est développé pour une meilleure exploration de la loi *a posteriori*. Enfin, un algorithme de Monte Carlo séquentiel a été défini pour l'estimation des hyperparamètres du modèle.

Ces méthodes ont été évaluées sur des images-test et sur des images naturelles. Le choix de la meilleure partition se fait par minimisation d'un critère indépendant de la numérotation. Les performances de l'algorithme sont évaluées via des métriques connues en statistiques mais peu utilisées en segmentation d'image.

Mots-clés : inférence bayésienne, méthodes de Monte Carlo par chaînes de Markov, Monte Carlo séquentiel, bayésien non paramétrique, processus de Dirichlet hiérarchique, champ de Potts, algorithme de Swendsen-Wang, segmentation, image.

⋆ ⋆ ⋆

Non parametric Bayesian model for joint image segmentation

This work concerns the joint segmentation of a set images in a Bayesian framework. The proposed model combines the hierarchical Dirichlet process (HDP) and the Potts random field. Hence, for a set of images, each is divided into homogeneous regions and similar regions between images are grouped into classes. On the one hand, thanks to the HDP, it is not necessary to define *a priori* the number of regions per image and the number of classes, common or not. On the other hand, the Potts field ensures a spatial consistency. The arising *a priori* and *a posteriori* distributions are complex and makes it impossible to compute analytically estimators. A Gibbs algorithm is then proposed to generate samples of the distribution *a posteriori*. Moreover, a generalized Swendsen-Wang algorithm is developed for a better exploration of the *a posteriori* distribution. Finally, a sequential Monte Carlo sampler is defined for the estimation of the hyperparameters of the model.

These methods have been evaluated on toy examples and natural images. The choice of the best partition is done by minimization of a numbering free criterion. The performance are assessed by metrics well-known in statistics but unused in image segmentation.

Key-words : Bayesian inference, Markov chain Monte Carlo, sequential Monte Carlo, non parametric Bayesian, hierarchical Dirichlet process, Potts field, Swendsen-Wang algorithm, segmenation, image.

Remerciements

Je remercie M. Faïcel CHAMROUKHI et M. Xavier DESCOMBES d'avoir accepté d'évaluer mon travail en tant que rapporteurs. Merci pour l'intérêt apporté à mon travail, la lecture minutieuse de mon manuscrit et les axes de réflexion proposés. Je remercie également M. François DESBOUVRIES et Mme Florence FORBES d'avoir accepté de faire partie du jury.

Merci à l'Agence Nationale de la Recherche d'avoir financé ce travail via le projet ANR BNPSI n°ANR-13-BS-03-0006.

Je remercie mes encadrants M. François CARON, M. Nicolas DOBIGEON, M. Jean-François GIOVANNELLI et Mme Audrey GIREMUS de m'avoir fait confiance pour ce travail de thèse. Je les remercie pour les idées novatrices et leur accompagnement personnalisé. Merci de m'avoir guidée dans la découverte du monde de la recherche. J'ai beaucoup appris à vos côtés, autant sur le plan scientifique que social. Grâce à vous j'ai pu avancer toujours un peu plus. Merci pour les différents déplacements que j'ai pu effectuer pendant ma thèse, notamment à l'université d'Oxford et à l'IRIT à Toulouse.

Je remercie également les différents services administratifs qui m'ont aidée au cours de ce travail.

Je remercie tous les stagiaires, doctorants, post-doctorants et chercheurs que j'ai pu rencontrer pendant ces années. Merci à Andrei spontanément disposé à m'aider pour des problèmes informatiques. Le trio que nous formions avec Mireille et Ouiame était vraiment spécial, merci les filles pour cette amitié qui s'est construite. Mireille, Ouiame, Mariem et Roxana ont changé le B2.48 en salle de conférences, sans oublier la chaise de la confession qui permettait à chacune de se livrer sans retenue. Merci pour tous ces beaux moments passés ensemble.

Je remercie mes anciens camarades de classes préparatoires qui ont suivi mon projet de loin et qui n'ont cessé d'y croire. Un merci spécial à Yawo. Merci à Alex pour son soutien et pour les nouvelles de l'autre côté de l'océan !

Merci à toutes les personnes que j'ai pu côtoyer à l'association à Toulouse. Merci aux Jeunes

qui m'ont gentiment accueillie et avec qui j'ai passé la plupart de mes week-ends pendant ces années. Un merci particulier à Loïc pour les discussions interminables que nous pouvions avoir, Maitena pour sa joie de vivre et Nicolas pour sa zénitude légendaire. Ce fut un plaisir !

Merci infiniment à ma famille qui n'a jamais cessé de m'encourager et de croire en moi. Je n'y serais pas arrivée sans vous. Vous avez encaissé tous les coups de colère, de doute, de désespoir. En retour, vous avez rempli mon cœur d'espoir, de joie et de détermination ! Je suis reconnaissante de vous avoir ! Merci papa, maman, Chéryl, Sandy et William.

Et, merci à Georges-Hervé, pour tout.

Koklo nɔ nu sin bo nɔ wɔn jixwe ă.
Proverbe fon
La poule a l'habitude de boire de l'eau et de ne pas oublier le ciel.
« Je me souviendrai toujours de toi »

Su may dee ci àll, gayndee may rey.
Proverbe wolof
Si je dois mourir en brousse,
que ce soit le lion qui me tue.

Table des matières

Nomenclature

Cas mono-données

N	nombre de pixels dans l'image
$\boldsymbol{x}$	ensemble d'observations associées aux pixels
z_n	classe affectée au pixel n
ϑ_n	paramètre associé au pixel n
m_k	nombre de pixels affectés à la classe k
$\boldsymbol{z}$	ensemble des classes assignées aux pixels

Cas multi-données

J	nombre d'images
N_j	nombre de pixels/sites dans l'image j
(j,n)	indique le pixel n dans l'image j
y_{jn}	observation au pixel (j,n)
θ_{jn}	paramètre associé au pixel (j,n)
c_{jn}	région du pixel (j,n)
ψ_{jt}	paramètre de la région t
ν_{jt}	nombre de pixels dans la région t de l'image j
$m_{j\cdot}$	nombre de régions dans l'image j
$r_{j_{nq}}$	lien entre les pixels (j,n) et (j,q)
C_{jl}	ensemble de pixels dans le spin-cluster l de l'image j
$\boldsymbol{c}_{jl}$	région affectée aux pixels dans C_{jl}
d_{jt}	classe affectée aux pixels dans la région t
ϕ_k	paramètre de la classe k
A_k	ensemble de pixels affectés à la classe k
m_{jk}	nombre de régions affectées à la classe k dans l'image j
$m_{\cdot k}$	nombre total de régions affectées à la classe k
$m_{\cdot\cdot}$	nombre total de régions
$\boldsymbol{y}$	ensemble d'observations associées aux pixels
$\boldsymbol{c}$	ensemble de régions associées aux pixels
$\boldsymbol{r}$	ensemble de tous les liens
$\boldsymbol{d}$	ensemble des classes assignées aux régions

Introduction

Classification non supervisée et segmentation d'images

La classification joue un rôle fondamental en analyse de données et ses applications sont très variées. En médecine préventive, par exemple, un enjeu est de déterminer si un facteur est à risque pour une pathologie donnée. Dans un autre contexte, les systèmes de recommandation utilisés par les sites Internet décident si un produit est susceptible de plaire à un utilisateur en fonction de ses choix précédents et des caractéristiques dudit produit. Dans ces deux cas, la classification est binaire car le choix est fait entre deux hypothèses « risqué » ou non, « adapté au goût » ou non. Un exemple plus compliqué est la classification de documents. Pour un ensemble de documents, une classification consiste à les regrouper automatiquement par thème ou de prédire la classe d'un nouveau document, dans une bibliothèque par exemple, connaissant la partition des documents déjà présents. Notons de plus que le critère utilisé pour la construction de la classification est essentiel. Dans le dernier cas par exemple, les classifications par langue et par thème ne sont pas identiques, ainsi, les critères « langue » ou « thème » permettent d'interpréter le résultat obtenu.

Classifier un ensemble de N observations $\boldsymbol{x} = \{x_1, \cdots, x_N\}$ consiste ainsi à les regrouper en K *classes* : classifier est donc partitionner. Ainsi, l'objectif est de réunir les observations similaires dans une même classe et séparer les observations différentes. Deux types de classification existent :

- la classification supervisée est étroitement liée au concept de *prédiction*. Un ensemble de données dites d'entraînement sont pré-classifiées pour extraire les caractéristiques de chaque classe permettant ainsi de prédire la classe d'une nouvelle donnée
- la classification non supervisée a pour but d'effectuer une partition des observations sans informations sur les classes telles que leur nombre ou traits distinctifs.

La segmentation d'une image de N pixels se définit comme la subdivision de cet ensemble en K classes. La segmentation peut donc être considérée comme une classification non supervisée

des N pixels. Un critère intuitif de regroupement de pixels est la similarité de leurs niveaux de gris lorsqu'à chaque pixel est associée une valeur traduisant son intensité lumineuse.

Pour des configurations complexes, il peut être intéressant d'utiliser une approche probabiliste bayésienne pour l'inférence de la classification. Dans ce cas, un paramètre caché ϑ_n est associé à chaque observation $x_n, n = 1, \cdots, N$. Sa valeur est caractéristique d'une classe, c'est-à-dire qu'elle est partagée par tous les pixels d'une même classe. De plus, la loi des paramètres sachant les données, $p(\boldsymbol{\vartheta}|\boldsymbol{x})$, est construite par le théorème de Bayes.

Un modèle bayésien couramment utilisé en segmentation est le champ de Markov aléatoire, en particulier le champ de Potts, qui assure une homogénéité spatiale : plus une classe est adoptée par ses voisins plus la probabilité qu'un pixel soit affecté à cette classe est élevée.

Inférence bayésienne non paramétrique

Les modèles précédents sont des modèles paramétriques, c'est-à-dire qu'ils dépendent d'un nombre fini de paramètres. En particulier, le nombre de classes K est supposé connu. Une solution pour lever cette condition est l'utilisation des modèles bayésiens non paramétriques.

Les modèles bayésiens non paramétriques sont définis sur un espace de paramètres de dimension infinie. Il s'agit donc de modèles inférant des paramètres qui « s'adaptent » aux observations présentes. Un de ces modèles est le processus de Dirichlet (DP) utilisé en particulier en classification. En effet, un processus descriptif du DP nommé *processus de restaurants chinois* permet d'observer que par construction les réalisations du DP se répètent et se regroupent ainsi en classes dont le nombre n'est *a priori* pas limité.

Bien que favorisant la création de classes, le processus de Dirichlet seul ne convient pas à la segmentation d'images. En effet, les interactions spatiales inhérentes à une image ne sont pas prises en compte. Il convient alors de le combiner à un modèle spatial, par exemple un champ de Markov aléatoire, comme proposé dans la littérature, pour allier l'apparition automatique de classes et l'homogénéité spatiale.

Nous nous intéressons dans ce travail à la segmentation jointe d'un ensemble d'images qui présentent des classes partagées. Elle peut être utile pour différentes applications. Par exemple, en télédétection : la segmentation conjointe peut être appliquée à des images correspondant à différents spectres pour notamment identifier les spectres donnant des informations redondantes.

Segmentation jointe d'un ensemble d'images

La segmentation conjointe d'un ensemble d'images est l'inférence de la partition non seulement de chacune des images mais aussi de la partition conjointe, ce qui revient à segmenter

chaque image et regrouper les classes identiques entre les images. Elle est particulièrement utile dans les cas où une classe est typique de certaines images et sous-représentée dans d'autres : l'inférence dans le deuxième lot d'images est facilitée par les caractéristiques obtenues dans le premier lot. Elle s'applique à différentes problématiques :

- analyse d'une collection d'images partageant des propriétés similaires
- traitement d'une image de grande taille découpée en imagettes analysées conjointement.

Nous proposons un modèle combinant le processus de Dirichlet hiérarchique (HDP), un modèle bayésien non paramétrique qui permet de ne pas prédéfinir le nombre de classes, et un champ de Potts pour l'homogénéité spatiale. Le HDP est une généralisation hiérarchique du processus de Dirichlet. Il est défini de telle sorte que pour un groupe d'ensembles de données différents et liés, les paramètres associés aux observations dans chaque groupe soient distribués *a priori* selon un processus de Dirichlet. Cette construction implique l'existence dans chaque groupe de *sous-classes* ayant des paramètres uniques. Ces paramètres uniques sont à leur tour supposés distribués suivant un processus de Dirichlet unique. Ainsi, plusieurs sous-classes dans différents groupes peuvent partager le même paramètre, elles sont alors regroupées en classes. Le HDP modélise alors *a priori* une partition conjointe.

Les lois *a priori* et *a posteriori* en résultant sont de grande dimension et avec des constantes de normalisation inaccessibles. Le calcul d'estimateurs requiert alors un échantillonnage.

Organisation du manuscrit

Les trois premiers chapitres de ce manuscrit sont dédiés à la présentation des concepts et des méthodes sur lesquels sont fondés le modèle bayésien non paramétrique proposé pour la segmentation et les algorithmes d'estimation associés.

Le **chapitre 1** est une présentation générale des algorithmes d'échantillonnage qui seront utiles pour l'exploration des différentes distributions *a posteriori* rencontrées dans ce manuscrit. Les bases des méthodes de Monte Carlo par chaînes de Markov sont présentées, deux algorithmes classiques sont aussi décrits : les algorithmes de Gibbs et de Metropolis-Hastings. De plus, l'algorithme de Monte Carlo séquentiel est présenté pour l'exploration d'une suite de distributions. Dans ce travail il est mis à profit pour explorer des lois de probabilité tout en estimant des hyperparamètres du modèle.

Le **chapitre 2** introduit les modèles bayésiens non paramétriques. Les processus de Dirichlet et de Dirichlet hiérarchique essentiels à la définition des modèles proposés y sont présentés. Des modèles génératifs fondés sur la construction en urne de Pólya et en *stick-breaking* sont également introduits. Suivent des algorithmes de Gibbs pour l'inférence des paramètres.

Le **chapitre 3** détaille les principes de pré-segmentation et de segmentation d'images. La

pré-segmentation consiste à sur-segmenter les images pour réduire la taille du problème de segmentation. Des modèles associés, en particulier le champ de Markov aléatoire sont décrits.

Dans le **chapitre 4**, le modèle DP-Potts est d'abord présenté pour la segmentation d'une image. Nous développons ensuite le modèle HDP-Potts de segmentation jointe que nous proposons. Les équations d'échantillonnage de Gibbs en résultant sont ensuite présentées. L'algorithme de Swendsen-Wang introduit pour une meilleure exploration de la loi *a posteriori* des paramètres est appliqué à notre modèle. Enfin, les hyperparamètres du modèle conditionnent la classification obtenue, il est alors essentiel de les fixer de manière appropriée ou de les estimer. Ce problème est difficile puisqu'ils apparaissent dans des constantes de normalisation de lois impossibles à calculer. Une contribution de ce travail est de proposer un algorithme de Monte Carlo séquentiel pour l'estimation des valeurs les plus vraisemblables.

Le **chapitre 5** décline le modèle générique proposé dans différentes applications, les classes étant caractérisées par un niveau de gris commun ou un histogramme de couleur ou de texture. Il en résulte des choix de lois de vraisemblance différentes et les équations de l'échantillonneur doivent être adaptées en conséquence. Il présente enfin une synthèse des résultats obtenus sur des images-test et des images naturelles. Des contributions méthodologiques sont aussi apportées dans cette partie notamment au travers de l'introduction de métriques peu exploitées en traitement d'images pour la définition de l'estimateur de la segmentation et des critères de performance. Premièrement, le choix de la meilleure partition se fait par la minimisation d'un coût indépendant de la numérotation des étiquettes contournant ainsi le problème de *label-switching*. Ce dernier se traduit par une permutation des numéros de classes attribuées aux étiquettes, rendant ainsi la comparaison délicate entre les chaînes. Deuxièmement, pour les images-test, des métriques sont introduites pour évaluer les performances des différents tests. Certaines sont dépendantes de la numérotation, entraînant donc une étape de renumérotation avant leur calcul et d'autres sont insensibles à la numérotation. Chaque métrique permet d'évaluer une propriété particulière de la segmentation, par exemple le pourcentage de pixels bien classés ou la proportion de couples de pixels correctement liés.

CHAPITRE 1

Algorithmes d'échantillonnage

La procédure d'échantillonnage consiste à simuler des échantillons représentatifs d'une distribution complexe. Ces échantillons peuvent autant servir à approcher la distribution de probabilité d'origine, à localiser son mode ou à estimer numériquement une intégrale non analytiquement calculable.

Dans ce chapitre, des algorithmes génériques sont détaillés et seront utilisés pour l'inférence dans les suivants. Une méthode simple d'approximation d'intégrale est d'abord introduite. Ensuite, deux algorithmes classiques d'échantillonnage par *méthodes de Monte Carlo par chaînes de Markov* sont présentés. Pour finir nous introduisons l'algorithme de Monte Carlo séquentiel proposé par Del Moral qui permet de générer des échantillons suivant une suite de lois [DdG01].

Les notions présentées dans ce chapitre s'inspirent des livres [LC98], [Bis06] et [GCS+13].

1.1 Intégration de Monte Carlo

Soit un ensemble de paramètres $\boldsymbol{\vartheta} = \{\vartheta_1, \cdots, \vartheta_N\}$, les problèmes d'estimation reposent souvent sur le calcul d'espérances :

$$\mathbb{E}_{p(\boldsymbol{\vartheta})}[g(\boldsymbol{\vartheta})] = \int g(\boldsymbol{\vartheta}) p(\boldsymbol{\vartheta}) \mathrm{d}\boldsymbol{\vartheta}. \tag{1.1}$$

où g est une fonction quelconque p-intégrable et p une densité de probabilité. Les ϑ_n, $n = 1, \cdots, N$ peuvent être scalaires ou vectoriels, en outre, ϑ_n prend ses valeurs dans un espace d'états E. $\mathbb{E}_*(**)$ est l'espérance de $**$ suivant la distribution $*$. De plus, $p(\boldsymbol{\vartheta})$ désigne ici une loi sur $\boldsymbol{\vartheta}$ pouvant dépendre d'autres variables ou d'observations $\boldsymbol{x} = \{x_1, \cdots, x_N\}$.

Si g est, par exemple, la fonction identité, l'intégrale (1.1) revient au calcul de la moyenne

du vecteur aléatoire de densité de probabilité p. Considérons le cas où g est la vraisemblance des données aux paramètres, on pose $g(\boldsymbol{\vartheta}) = f(\boldsymbol{x}|\boldsymbol{\vartheta})$ et p la distribution *a priori* sur les paramètres. L'expression analytique de (1.1) peut être déduite dans des cas spécifiques, par exemple si g et p sont conjuguées, c'est-à-dire que les lois *a priori* et *a posteriori* ont la même forme. Une présentation des distributions conjuguées est disponible dans [Bis06]. Un exemple est détaillé ci-dessous.

Exemple 1.1 (Calcul d'intégrales dans le cas de lois conjuguées)
Soit la distribution g définie comme $g(\boldsymbol{\vartheta}) \equiv f(\boldsymbol{x}|\boldsymbol{\vartheta})$ où $\boldsymbol{x} = \{x_1, \cdots, x_N\}$ avec les observations $x_n, n = 1, \cdots, N$ qui sont indépendantes les unes des autres conditionnellement à leurs paramètres et $f(x_n|\vartheta_n) = \mathcal{N}(x_n; \vartheta_n, \sigma^2)$ la loi normale prise en x_n de moyenne ϑ_n et de variance σ^2. Soient de plus les paramètres $\boldsymbol{\vartheta}$ distribués indépendamment suivant $\mathcal{N}(\vartheta_n; \mu_0, \sigma_0^2), n = 1, \cdots, N$. Par ailleurs, x_n et ϑ_n sont choisis scalaires. L'intégrale (1.1) *revient alors à calculer la densité marginale $p(\boldsymbol{x})$ et s'écrit :*

$$\begin{aligned} p(\boldsymbol{x}) &= \int \prod_{n=1}^{N} f(x_n|\vartheta_n) p(\vartheta_n) \mathrm{d}\vartheta_n = \prod_{n=1}^{N} \int \mathcal{N}(x_n; \vartheta_n, \sigma^2) \mathcal{N}(\vartheta_n; \mu_0, \sigma_0^2) \mathrm{d}\vartheta_n \\ &= \prod_{n=1}^{N} \mathcal{N}(x_n; \mu_0, \sigma^2 + \sigma_0^2) \end{aligned}$$

Ainsi, pour une observation x_n distribuée selon une loi normale de moyenne ϑ_n et de variance σ^2, et pour une loi a priori normale sur ϑ_n, sa distribution marginale est aussi une loi normale de moyenne μ_0 et de variance la somme $\sigma^2 + \sigma_0^2$

Lorsque g et p ne sont pas conjuguées, (1.1) peut être approchée à l'aide des *méthodes de Monte Carlo* dont l'idée générale est :

- d'échantillonner I réalisations indépendantes $(\boldsymbol{\vartheta}^{(1)}, \cdots, \boldsymbol{\vartheta}^{(I)})$ suivant la distribution $p(\boldsymbol{\vartheta})$
- de calculer ensuite l'estimateur de Monte Carlo défini par : $\hat{g} = I^{-1} \sum_{i=1}^{I} g(\boldsymbol{\vartheta}^{(i)})$ et consistant pour l'intégrale (1.1), c'est-à-dire :

$$\hat{g} \underset{I \to \infty}{\longrightarrow} \mathbb{E}_{p(\boldsymbol{\vartheta})}[g(\boldsymbol{\vartheta})] \quad \text{p.s.}$$

Ainsi, s'il est aisé d'obtenir des échantillons suivant la distribution p, il est possible d'estimer l'intégrale (1.1). Pour des distributions complexes qui ne sont pas faciles à échantillonner, différentes solutions ont été proposées dans la littérature. Une première méthode est proposée dans la suite, qui est basée sur l'introduction d'une seconde distribution plus simple.

1.2 Échantillonnage d'importance

L'*échantillonnage d'importance* se propose de contourner la difficulté de simuler la distribution p par l'introduction d'une loi q intermédiaire, facile à échantillonner. Lorsque le support

de q contient celui de p, l'équation (1.1) peut être réécrite comme suit [Gew89] :

$$\mathbb{E}_{p(\boldsymbol{\vartheta})}[g(\boldsymbol{\vartheta})] = \int g(\boldsymbol{\vartheta})p(\boldsymbol{\vartheta})\mathrm{d}\boldsymbol{\vartheta} = \int_S g(\boldsymbol{\vartheta})\frac{p(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta})}q(\boldsymbol{\vartheta})\mathrm{d}\boldsymbol{\vartheta}.$$

où S est le support de p. En suivant la méthode de Monte Carlo, la moyenne est approchée grâce aux $(\boldsymbol{\vartheta}^{(1)}, \cdots, \boldsymbol{\vartheta}^{(I)})$ échantillonnés suivant q [RC04] par :

$$\mathbb{E}_{q(\boldsymbol{\vartheta})}[g(\boldsymbol{\vartheta})w(\boldsymbol{\vartheta})] \simeq \frac{1}{I}\sum_{i=1}^{I} g(\boldsymbol{\vartheta}^{(i)})\frac{p(\boldsymbol{\vartheta}^{(i)})}{q(\boldsymbol{\vartheta}^{(i)})} = \sum_{i=1}^{I} g(\boldsymbol{\vartheta}^{(i)})w(\boldsymbol{\vartheta}^{(i)})$$

avec $w(\boldsymbol{\vartheta}) = \frac{1}{I}\frac{p(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta})}$ les *poids* non normalisés ; q est appelée *loi d'importance*. Dans la plupart des cas, les distributions q et p ne sont connues qu'à une constante de normalisation près, une renormalisation des poids peut alors être effectuée :

$$W(\boldsymbol{\vartheta}^{(i)}) = \frac{w(\boldsymbol{\vartheta}^{(i)})}{\sum_{i=1}^{I} w(\boldsymbol{\vartheta}^{(i)})} \tag{1.2}$$

L'estimation de l'espérance (1.1) s'écrit alors $\mathbb{E}_{p(\boldsymbol{\vartheta})}[g(\boldsymbol{\vartheta})] \simeq \sum_{i=1}^{I} g(\boldsymbol{\vartheta}^{(i)})W(\boldsymbol{\vartheta}^{(i)})$, elle permet entre autres d'avoir une approximation de la distribution p en choisissant g comme une mesure de Dirac centrée en $\boldsymbol{\vartheta}$.

Un exemple d'échantillonnage d'importance est présenté à la figure 1.1 où on observe que les échantillons obtenus sont représentatifs de la distribution cible.

Pour appliquer l'échantillonnage d'importance, il est nécessaire de choisir la distribution q de telle sorte qu'elle soit proche de p, que son domaine de définition inclue celui de p et que les queues de la distribution q soient plus lourdes que celles de p. Ces conditions sont difficiles à satisfaire pour des distributions complexes en grande dimension, il convient donc dans ces cas d'avoir recours à d'autres méthodes d'échantillonnage, par exemple les méthodes de Monte Carlo par chaînes de Markov.

1.3 Méthodes de Monte Carlo par chaînes de Markov (MCMC)

Cette partie présente une vue d'ensemble sur les méthodes de Monte Carlo par chaînes de Markov, puis, deux cas particuliers sont décrits : les algorithmes de Metropolis-Hastings et de Gibbs.

Les méthodes MCMC consistent à construire itérativement une séquence $(\boldsymbol{\vartheta}^{(1)}, \cdots, \boldsymbol{\vartheta}^{(I)})$ définissant une chaîne de Markov et de distribution asymptotique p, la loi cible.

Une chaîne de Markov satisfait entre autres une propriété d'indépendance conditionnelle :

$$p(\boldsymbol{\vartheta}^{(i)}|\boldsymbol{\vartheta}^{(i-1)}, \cdots, \boldsymbol{\vartheta}^{(1)}) = p(\boldsymbol{\vartheta}^{(i)}|\boldsymbol{\vartheta}^{(i-1)})$$

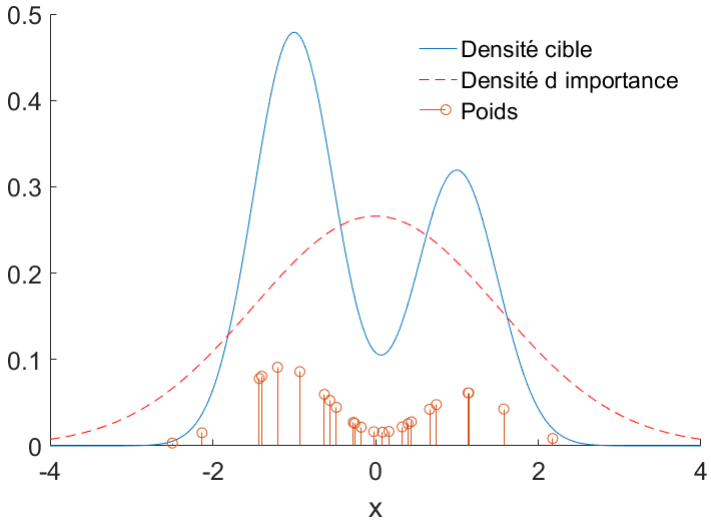


FIGURE 1.1 – Exemple d'échantillonnage d'importance avec la loi cible définie comme un mélange de deux lois gaussiennes de moyennes respectives -1 et 1, d'écart-type commun 0.5 et de proportions 0.6 et 0.4. La loi de proposition est la loi normale de moyenne 0 et d'écart-type 1.5. 25 échantillons ont été utilisés.

Une chaîne de Markov peut alors être définie à partir de deux composantes :

- la valeur initiale $\boldsymbol{\vartheta}^{(0)}$
- le noyau de transition $\mathcal{K}(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})$ qui est une loi de probabilité suivant $\boldsymbol{\vartheta}$ à $\boldsymbol{\vartheta}^{'}$ fixé qui permet de passer de l'état $\boldsymbol{\vartheta}^{'}$ à $\boldsymbol{\vartheta}$

Pour assurer que la séquence $(\boldsymbol{\vartheta}^{(i)})$ soit asymptotiquement distribuée suivant la distribution cible p, le noyau de transition doit vérifier certaines propriétés [Bis06]. En particulier, la distribution d'intérêt p est invariante par rapport à la chaîne de Markov, c'est-à-dire $p(\boldsymbol{\vartheta}) = \int p(\boldsymbol{\vartheta}^{'})\mathcal{K}(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})\mathrm{d}\boldsymbol{\vartheta}^{'}$, une condition nécessaire à satisfaire pour cela est que $\mathcal{K}$ soit réversible par rapport à p : $\mathcal{K}(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})p(\boldsymbol{\vartheta}^{'}) = \mathcal{K}(\boldsymbol{\vartheta}^{'}, \boldsymbol{\vartheta})p(\boldsymbol{\vartheta})$. La propriété d'ergodicité assure, en outre, sous certaines conditions [RC04], que la distribution $p(\boldsymbol{\vartheta}^{(I)})$ converge vers la distribution invariante p lorsque $I \to \infty$ indépendamment de l'état initial $\boldsymbol{\vartheta}^{(0)}$.

Critère de convergence

Pour éviter que l'initialisation de la chaîne influe sur l'estimation, un temps de chauffe est défini et les échantillons correspondants ne sont pas considérés ; il est souvent choisi comme la première moitié de la chaîne.

Par ailleurs, les échantillons successifs sont corrélés par construction, c'est-à-dire que l'échantillon à l'itération i dépend des précédents. Il convient alors de choisir un pas, c'est-à-dire utiliser tous les t-ièmes échantillons avec $t > 1$ pour avoir des échantillons le plus indépendants

possible. Ce pas peut par exemple être déterminé en calculant l'autocorrélation d'échantillons successifs. Pour un échantillon donné, il est alors possible de déterminer l'échantillon suivant qui lui est le plus proche tout en lui étant le moins corrélé.

Un problème majeur dans l'utilisation des méthodes de Monte Carlo par chaînes de Markov est la détection de la convergence de la chaîne : il n'est pas possible de déterminer avec exactitude si la chaîne de Markov a convergé ; mais, certains critères peuvent être utilisés suivant les caractéristiques à évaluer dans la chaîne obtenue. Un critère peut donc être la stabilité de la distribution des échantillons (ou de son logarithme) : étant donné que la distribution des échantillons tend asymptotiquement vers une distribution invariante, elle devrait se stabiliser autour d'une valeur finie lorsque la convergence est atteinte. Ce critère ne permet néanmoins pas de s'assurer que les échantillons varient suffisamment entre les itérations, preuve d'une exploration efficace de la distribution. Dans ce contexte, des tests ont été introduits pour évaluer la variance des réalisations obtenues ; en particulier une estimation des variances inter-chaînes et intra-chaînes. Elle a été proposée dans [BG98] et permet de décider si la convergence a eu lieu. Ces calculs sont proposés pour un ensemble de chaînes lancées en parallèle avec des états initiaux différents.

Pour I échantillons ne comprenant pas ceux du temps de chauffe et enregistrés avec le pas défini, la distribution p est approchée par une somme de mesures de Dirac centrées sur ces échantillons pondérées par $1/I$:

$$p(\mathrm{d}\boldsymbol{\vartheta}) \simeq \frac{1}{I} \sum_{i=1}^{I} \delta_{\boldsymbol{\vartheta}^{(i)}}(\mathrm{d}\boldsymbol{\vartheta})$$

Deux algorithmes MCMC standards sont présentés dans les sections 1.3.1 et 1.3.2.

1.3.1 Algorithme de Metropolis-Hastings

Considérons le cas où il n'est pas facile d'échantillonner suivant p. Une stratégie consiste alors à avoir recours à l'échantillonnage suivant une loi candidate q via l'algorithme de Metropolis-Hastings présenté dans l'algorithme 1.1.

La procédure est : à une itération i, une valeur candidate $\boldsymbol{\vartheta}^\star$ est générée suivant q, puis est acceptée, c'est-à-dire $\boldsymbol{\vartheta}^{(i)} = \boldsymbol{\vartheta}^\star$, avec la probabilité ϱ et est rejetée, c'est-à-dire $\boldsymbol{\vartheta}^{(i)} = \boldsymbol{\vartheta}^{(i-1)}$, avec la probabilité $1 - \varrho$ avec :

$$\varrho(\boldsymbol{\vartheta}^\star, \boldsymbol{\vartheta}^{(i-1)}) = \min\left(1, \frac{p(\boldsymbol{\vartheta}^\star)}{p(\boldsymbol{\vartheta}^{(i-1)})} \frac{q(\boldsymbol{\vartheta}^{(i-1)}|\boldsymbol{\vartheta}^\star)}{q(\boldsymbol{\vartheta}^\star|\boldsymbol{\vartheta}^{(i-1)})}\right) \tag{1.3}$$

Notons que la probabilité ϱ ne fait intervenir que les rapports des distributions, ce qui assure

que l'algorithme de Metropolis-Hastings soit aussi défini pour des lois p et q connues à des constantes de normalisation près [RC04].

Vérifions à présent si la distribution de la chaîne de Markov ainsi générée converge en loi vers la distribution cible p, c'est-à-dire l'invariance de p par rapport à la chaîne. Pour ce faire, nous montrerons que la relation de réversibilité est satisfaite.

Notons $\mathcal{K}(\mathrm{d}\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})$ le noyau de transition de $\boldsymbol{\vartheta}^{'}$ à $\mathrm{d}\boldsymbol{\vartheta}$ d'expression :

$$\mathcal{K}(\mathrm{d}\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'}) = q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'})\varrho(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})\mathrm{d}\boldsymbol{\vartheta} + r(\boldsymbol{\vartheta}^{'})\delta_{\boldsymbol{\vartheta}^{'}}(\mathrm{d}\boldsymbol{\vartheta})$$

avec

$$r(\boldsymbol{\vartheta}^{'}) = 1 - \int q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'})\varrho(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})\mathrm{d}\boldsymbol{\vartheta}$$

On a $r(\boldsymbol{\vartheta}^{'})p(\boldsymbol{\vartheta}^{'})\delta_{\boldsymbol{\vartheta}^{'}}(\boldsymbol{\vartheta}) = r(\boldsymbol{\vartheta})p(\boldsymbol{\vartheta})\delta_{\boldsymbol{\vartheta}}(\boldsymbol{\vartheta}^{'})$, alors, pour prouver que $\mathcal{K}(\mathrm{d}\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'})p(\boldsymbol{\vartheta}^{'}) = \mathcal{K}(\mathrm{d}\boldsymbol{\vartheta}^{'}, \boldsymbol{\vartheta})p(\boldsymbol{\vartheta})$ il suffit de montrer que $p(\boldsymbol{\vartheta}^{'})q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'})\varrho(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'}) = p(\boldsymbol{\vartheta})q(\boldsymbol{\vartheta}^{'}|\boldsymbol{\vartheta})\varrho(\boldsymbol{\vartheta}^{'}, \boldsymbol{\vartheta})$.

$$\begin{aligned} p(\boldsymbol{\vartheta}^{'})q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'})\varrho(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}^{'}) &= \min\Big(p(\boldsymbol{\vartheta}^{'})q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'}), p(\boldsymbol{\vartheta})q(\boldsymbol{\vartheta}^{'}|\boldsymbol{\vartheta})\Big) \\ &= \min\Big(p(\boldsymbol{\vartheta})q(\boldsymbol{\vartheta}^{'}|\boldsymbol{\vartheta}), p(\boldsymbol{\vartheta}^{'})q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'})\Big) = p(\boldsymbol{\vartheta})q(\boldsymbol{\vartheta}^{'}|\boldsymbol{\vartheta})\varrho(\boldsymbol{\vartheta}^{'}, \boldsymbol{\vartheta}). \end{aligned}$$

Algorithme 1.1 Algorithme de Metropolis-Hastings générique

Initialisation $\boldsymbol{\vartheta}^{(0)}$

pour $i = 1, \cdots, I$ **faire**

Échantillonner une valeur candidate $\boldsymbol{\vartheta}^{\star} \sim q(\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{(i-1)})$

Calculer la probabilité d'acceptation $\varrho(\boldsymbol{\vartheta}^{\star}, \boldsymbol{\vartheta}^{(i-1)}) = \min\left(1, \dfrac{p(\boldsymbol{\vartheta}^{\star})}{p(\boldsymbol{\vartheta}^{(i-1)})}\dfrac{q(\boldsymbol{\vartheta}^{(i-1)}|\boldsymbol{\vartheta}^{\star})}{q(\boldsymbol{\vartheta}^{\star}|\boldsymbol{\vartheta}^{(i-1)})}\right)$

Prendre $\boldsymbol{\vartheta}^{(i)} = \begin{cases} \boldsymbol{\vartheta}^{\star} & \text{avec la probabilité } \varrho(\boldsymbol{\vartheta}^{\star}, \boldsymbol{\vartheta}^{(i-1)}) \\ \boldsymbol{\vartheta}^{(i-1)} & \text{avec la probabilité } 1 - \varrho(\boldsymbol{\vartheta}^{\star}, \boldsymbol{\vartheta}^{(i-1)}) \end{cases}$

fin pour

En pratique, la loi de proposition q est difficile à choisir car elle doit être non seulement facile à échantillonner mais aussi cohérente avec la loi cible pour un faible taux de rejet tout en assurant une bonne exploration de l'espace de paramètres. Étudions à présent une autre méthode d'échantillonnage : l'algorithme de Gibbs qui peut en être une alternative.

1.3.2 Algorithme de Gibbs

Supposons possible l'échantillonnage suivant les densités conditionnelles $p(\vartheta_n|\vartheta_1, \cdots, \vartheta_{n-1}, \vartheta_{n+1}, \cdots, \vartheta_N)$, l'algorithme de Gibbs consiste alors à échantillonner itérativement chaque variable $\vartheta_n, n = 1, \cdots, N$ suivant sa distribution conditionnelle aux autres.

Cette procédure peut être effectuée de manière cyclique en mettant à jour toutes les variables ϑ_n à chaque itération ou en choisissant aléatoirement les variables à mettre à jour. Un algorithme de Gibbs générique est présenté en 1.2.

L'algorithme de Gibbs peut être vu comme un cas particulier des algorithmes de Metropolis-Hastings où les propositions sont toujours acceptées. Considérons pour cela la mise à jour du paramètre ϑ_n, et le cas particulier où q s'écrit :

$$q(\mathrm{d}\boldsymbol{\vartheta}|\boldsymbol{\vartheta}^{'}) = p(\vartheta_n|\boldsymbol{\vartheta}^{-n})\ \ \delta_{\boldsymbol{\vartheta}^{'-n}}(\mathrm{d}\boldsymbol{\vartheta}^{-n})$$

avec $\boldsymbol{\vartheta}^{-n} = \{\vartheta_1, \cdots, \vartheta_{n-1}, \vartheta_{n+1}, \cdots, \theta_N\}$. Il a été prouvé [Bis06] que la probabilité d'acceptation liée est toujours égale à 1. On en déduit que la distribution de la chaîne générée converge bien vers la distribution cible.

Algorithme 1.2 Algorithme de Gibbs générique

Initialisation $\vartheta_n^{(0)}, n = 1, \cdots, N$
pour $i = 1, \cdots, I$ **faire**
  Échantillonner $\vartheta_1^{(i)} \sim p(\vartheta_1|\vartheta_2^{(i-1)}, \vartheta_3^{(i-1)}, \cdots, \vartheta_N^{(i-1)})$
  Échantillonner $\vartheta_2^{(i)} \sim p(\vartheta_2|\vartheta_1^{(i)}, \vartheta_3^{(i-1)}, \cdots, \vartheta_N^{(i-1)})$
  $\cdots$
  Échantillonner $\vartheta_n^{(i)} \sim p(\vartheta_n|\vartheta_1^{(i)}, \vartheta_2^{(i)}, \cdots, \vartheta_{n-1}^{(i)}, \vartheta_{n+1}^{(i-1)}, \cdots, \vartheta_N^{(i-1)})$
  $\cdots$
  Échantillonner $\vartheta_N^{(i)} \sim p(\vartheta_N|\vartheta_1^{(i)}, \vartheta_2^{(i)}, \cdots, \vartheta_{N-1}^{(i)})$
fin pour

Il peut parfois être difficile d'explorer efficacement la loi avec cet algorithme. En outre, il existe des variantes de cette procédure plus efficaces pour explorer la loi. Citons par exemple le *partially collapsed Gibbs* ; cet échantillonneur remplace les distributions conditionnelles utilisées dans l'algorithme de Gibbs par des distributions conditionnelles marginalisées [VDP08].

Dans cette partie ont été présentés les algorithmes les plus classiquement utilisés pour obtenir des échantillons suivant une distribution $p(\boldsymbol{\vartheta})$. Nous présentons finalement un algorithme qui permet de simuler itérativement des échantillons selon une suite de distributions $p_m(\boldsymbol{\vartheta})_{m\geq 1}$ présentant la particularité d'être définies sur le même espace probabilisable. Il peut s'agir de distributions a posteriori obtenues en accumulant des données au fil du temps ou encore correspondant à différentes valeurs d'hyperparamètres. L'idée est généralement de considérer des distributions de plus en plus délicates à échantillonner, et convergeant vers la distribution d'intérêt. Ainsi la simulation de cette dernière est facilitée. Dans cette thèse, cette méthode nous permettra de gérer l'incertitude sur les hyperparamètres des modèles de segmentation proposés. L'algorithme de Monte Carlo séquentiel offre une solution élégante à ce problème.

1.4 Méthodes de Monte Carlo séquentielles (SMC)

Considérons une suite de distributions $p_m(\boldsymbol{\vartheta}_m)$ sur un vecteur de paramètres $\boldsymbol{\vartheta}$ avec $m \geq 1$. La notation abusive $\boldsymbol{\vartheta}_m$ a été ici adoptée pour des facilités d'écriture et ne signifie pas la m-ième composante de l'ensemble des paramètres mais plutôt l'ensemble des paramètres correspondant à l'étape m.

Il est récurrent, pour des distributions complexes, que l'expression des distributions p_m ne soit connue qu'à une constante de normalisation Z_m près ; nous considérons dans cette partie que tel est le cas. Les méthodes SMC restent non seulement définies dans ces conditions, mais elles permettent aussi d'avoir une approximation du rapport Z_m/Z_{m-1} pour deux itérations successives.

Nous nous plaçons en outre dans le cas où l'échantillonnage direct n'est pas aisé. Nous pouvons alors recourir à l'échantillonnage d'importance, c'est-à-dire qu'à l'étape m, l'idée est d'obtenir I échantillons pondérés $\left(\boldsymbol{\vartheta}_m^{(i)}, W_m^{(i)}\right), i = 1, \cdots, I$, appelés *particules*, dont la distribution empirique tend vers p_m [DDJ06] :

$$p_m(\mathrm{d}\boldsymbol{\vartheta}_m) \simeq \sum_{i=1}^{I} W_m^{(i)} \delta_{\boldsymbol{\vartheta}_m^{(i)}}(\mathrm{d}\boldsymbol{\vartheta}_m).$$

De plus, les échantillons sont obtenus suivant une loi dite d'importance q_m. Les poids servent à corriger l'écart entre q_m et p_m. Le choix de la distribution d'importance est crucial pour obtenir une bonne estimation avec un nombre raisonnable de particules. Si les distributions cibles p_m sont proches les unes des autres, une idée pour obtenir des échantillons pertinents à l'itération courante m est de faire simplement évoluer les particules de l'itération $m-1$ selon un noyau de transition $q_m(\boldsymbol{\vartheta}_m|\boldsymbol{\vartheta}_{m-1})$. La loi de proposition s'exprime alors comme suit :

$$q_m(\boldsymbol{\vartheta}_m) = \int q_1(\boldsymbol{\vartheta}_1) \prod_{k=2}^{m} q_k(\boldsymbol{\vartheta}_k|\boldsymbol{\vartheta}_{k-1}) \mathrm{d}\boldsymbol{\vartheta}_{1:m-1}. \tag{1.4}$$

où $1 : m-1$ désigne l'ensemble des paramètres échantillonnés des étapes 1 à $m-1$. Différentes solutions ont été proposées pour le choix des noyaux q_k. Une des plus appropriées au sens où elle garantit que les lois d'importance restent proches des lois d'intérêt est de prendre le noyau d'un algorithme MCMC, Gibbs ou Metropolis, ciblant la distribution p_m à l'itération m. Une discussion est menée à ce sujet dans [DDJ06]. Cette approche théoriquement élégante conduit à une difficulté. Pour des problèmes simples, il est aisé de calculer analytiquement (1.4), mais pour des cas complexes, ce n'est pas possible ; une solution proposée dans [DDJ06] est ainsi brièvement présentée dans cette partie.

A l'instar du filtrage particulaire, l'idée clé est d'éviter la marginalisation et donc le calcul d'intégrale présent dans (1.4) en échantillonnant directement la trajectoire $\boldsymbol{\vartheta}_{1:m}$. A cet effet, une

distribution cible artificielle sur $\boldsymbol{\vartheta}_{1:m}$ est introduite admettant $p_m(\boldsymbol{\vartheta}_m)$ comme loi marginale. Elle est construite à partir de noyaux rétrogrades sous la forme :

$$\tilde{p}_m(\boldsymbol{\vartheta}_{1:m}) \propto p_m(\boldsymbol{\vartheta}_m) \prod_{k=1}^{m-1} \varrho_k(\boldsymbol{\vartheta}_{k+1}, \boldsymbol{\vartheta}_k).$$

Ce noyau permet ainsi de ramener cet échantillonnage à un échantillonnage d'importance séquentiel classique où seuls les paramètres correspondants à l'état courant $\boldsymbol{\vartheta}_m$ sont échantillonnés. La procédure est présentée ci-après.

1.4.1 Échantillonnage d'importance séquentiel

Le principe de la méthode est de simuler des échantillons suivant une loi $q_m(\boldsymbol{\vartheta}_{1:m})$ et estimer la distribution cible à l'aide des poids non normalisés $w_m = \tilde{p}_m(\boldsymbol{\vartheta}_{1:m})/q_m(\boldsymbol{\vartheta}_{1:m})$.

La distribution d'importance jointe peut alors se réécrire sous la forme suivante :

$$q_m(\boldsymbol{\vartheta}_{1:m}) = q_{m-1}(\boldsymbol{\vartheta}_{1:m-1}) q_m(\boldsymbol{\vartheta}_m|\boldsymbol{\vartheta}_{m-1}) = q_1(\boldsymbol{\vartheta}_1) \prod_{k=2}^{m} q_k(\boldsymbol{\vartheta}_k|\boldsymbol{\vartheta}_{k-1}) \tag{1.5}$$

En pratique, cela signifie que pour obtenir des particules $\boldsymbol{\vartheta}_{1:m}^{(i)} \sim q_m(\boldsymbol{\vartheta}_{1:m})$ à chaque instant m, il faut échantillonner $\boldsymbol{\vartheta}_1^{(i)} \sim q_1(\boldsymbol{\vartheta}_1)$ puis $\boldsymbol{\vartheta}_k^{(i)} \sim q_k(\boldsymbol{\vartheta}_k|\boldsymbol{\vartheta}_{k-1}^{(i)})$ à l'instant k, $k = 2, \cdots, m$. Les poids non normalisés associés peuvent de même être récursivement calculés avec la décomposition :

$$\begin{aligned} w_m &= \frac{\tilde{p}_m(\boldsymbol{\vartheta}_{1:m})}{q_m(\boldsymbol{\vartheta}_{1:m})} \\ &= \frac{p_{m-1}(\boldsymbol{\vartheta}_{1:m-1})}{q_{m-1}(\boldsymbol{\vartheta}_{1:m-1})} \; \frac{\tilde{p}_m(\boldsymbol{\vartheta}_{1:m})}{\tilde{p}_{m-1}(\boldsymbol{\vartheta}_{1:m-1}) q_m(\boldsymbol{\vartheta}_m|\boldsymbol{\vartheta}_{m-1})} \end{aligned} \tag{1.6}$$

En procédant à la même subdivision que (1.5), l'équation (1.6) devient :

$$w_m = w_{m-1} \widetilde{w}_m = w_1 \prod_{k=2}^{m} \widetilde{w}_k$$

où les $\widetilde{w}_k$ sont les poids incrémentaux non normalisés et sont donnés par :

$$\widetilde{w}_k = \frac{\tilde{p}_k(\boldsymbol{\vartheta}_{1:k})}{\tilde{p}_{k-1}(\boldsymbol{\vartheta}_{1:k-1}) q_k(\boldsymbol{\vartheta}_k|\boldsymbol{\vartheta}_{k-1})} \tag{1.7}$$

Concernant le problème d'échantillonnage suivant $p_m(\boldsymbol{\vartheta}_m)$, rappelons que les échantillons sont artificiellement propagés selon un noyau rétrograde ϱ pour se ramener à un pseudo échantillonnage d'importance séquentiel.

Les poids incrémentaux non normalisés $\widetilde{w}_m$ s'expriment alors comme suit :

$$\widetilde{w}_m = \frac{p_m(\boldsymbol{\vartheta}_m)\ \ \varrho_{m-1}(\boldsymbol{\vartheta}_m, \boldsymbol{\vartheta}_{m-1})}{p_{m-1}(\boldsymbol{\vartheta}_{m-1})\ \ q_m(\boldsymbol{\vartheta}_m|\boldsymbol{\vartheta}_{m-1})} \tag{1.8}$$

Et la normalisation des poids se fait suivant :

$$W_m^{(i)} = \frac{W_{m-1}^{(i)} \widetilde{w}_m^{(i)}}{\sum_{\ell=1}^{I} W_{m-1}^{(\ell)} \widetilde{w}_m^{(\ell)}} \tag{1.9}$$

A l'itération m, l'approximation suivante est alors obtenue :

$$p_m(\mathrm{d}\boldsymbol{\vartheta}_{1:m}) \simeq \sum_{i=1}^{I} W_m^{(i)} \delta_{\boldsymbol{\vartheta}_{1:m}^{(i)}}(\mathrm{d}\boldsymbol{\vartheta}_{1:m}).$$

Finalement $p_m(\boldsymbol{\vartheta}_m)$ se déduit par simple marginalisation.

Une propriété importante est que pour des distributions connues à une constante de normalisation près, le SMC permet aussi d'estimer le rapport de constantes pour des instants successifs :

$$\frac{Z_m}{Z_{m-1}} \simeq \sum_{i=1}^{I} W_{m-1}^{(i)} \widetilde{w}_m^{(i)} \tag{1.10}$$

avec $Z_m = \int p_m(\boldsymbol{\vartheta}_m)\mathrm{d}\boldsymbol{\vartheta}_m$. Dans la suite du manuscrit, cette relation sera mise à profit pour calculer des constantes de normalisation et résoudre des problèmes d'estimation d'hyperparamètres par maximum de vraisemblance.

1.4.2 Rééchantillonnage

Une des limites de l'échantillonnage d'importance séquentiel est que les poids échantillonnés ont une variance croissante avec m [KLW94]. Après plusieurs itérations, les particules dégénèrent : un faible nombre de particules a un poids important et les autres un poids négligeable. Le rééchantillonnage a été proposé pour résoudre en partie ce problème. Il consiste à calculer un critère de dégénérescence, et, lorsqu'il est en dessous d'un seuil ϵ fixé, par exemple $I/2$, les particules de poids fort sont dupliquées et celles de poids faible supprimées. Les particules $\left(\overline{\boldsymbol{\vartheta}}_m^{(i)}, \overline{W}_m^{(i)}\right)$ ainsi rééchantillonnées sont définies équiprobables. Il s'ensuit qu'elles ont toutes le même poids $\overline{W}_m^{(i)} = 1/I$. Nous choisissons dans ce manuscrit de répliquer les particules selon une loi multinomiale, c'est-à-dire, que chaque particule i est copiée $\xi^{(i)}$ fois et $\xi^{(i)} \sim \mathrm{Mult}(W_m^{(i)}), \sum_{i=1}^{I} \xi^{(i)} = I$ et $\mathrm{Mult}(p)$ est la loi multinomiale de paramètre p.

Cette méthode de reconstruction est intéressante du fait que seules les particules les plus probables, à un instant donné, sont propagées dans le temps.

La dégénérescence est mesurée par l'*effective sample size (*ESS*)* [LC98] :

$$\mathrm{ESS} = \frac{1}{\sum_{i=1}^{I} (W_m^{(i)})^2}, \quad 1 \leq \mathrm{ESS} \leq I \tag{1.11}$$

Les méthodes de Monte Carlo séquentielles sont une combinaison de l'échantillonnage d'importance séquentiel et du rééchantillonnage ; un algorithme générique est présenté en 1.3 et la figure 1.2 en est un exemple.

Algorithme 1.3 Algorithme générique de Monte Carlo séquentiel

A l'instant 1, pour les I particules,
pour $i = 1, \cdots, I,$ **faire**
  Échantillonner $\boldsymbol{\vartheta}_1^{(i)} \sim q_1(\boldsymbol{\vartheta}_1)$
  Calculer les poids non normalisés $w_1^{(i)}$ et normalisés $W_1^{(i)}$
fin pour
Calculer l'ESS et pour les I particules,
  si ESS $< \epsilon$, rééchantillonner : $\left(\overline{\boldsymbol{\vartheta}}_1^{(i)}, \overline{W}_1^{(i)}\right) = \left(\overline{\boldsymbol{\vartheta}}_1^{(\ell)}, 1/I\right)$, $\ell = 1, \cdots, L$ indiquant les L particules à répliquer
  sinon conserver les échantillons $\left(\overline{\boldsymbol{\vartheta}}_1^{(i)}, \overline{W}_1^{(i)}\right) = \left(\boldsymbol{\vartheta}_1^{(i)}, W_1^{(i)}\right)$
pour $m = 2, \cdots, M,$ **faire**
  pour $i = 1, \cdots, I,$ **faire**
    Échantillonner $\boldsymbol{\vartheta}_m^{(i)} \sim q_m(\boldsymbol{\vartheta}_m | \boldsymbol{\vartheta}_{m-1})$
    Calculer les poids non normalisés $w_m^{(i)}$ et normalisés $W_m^{(i)}$
  fin pour
  Calculer l'ESS et pour les I particules,
    si ESS $< \epsilon$, rééchantillonner : $\left(\overline{\boldsymbol{\vartheta}}_m^{(i)}, \overline{W}_m^{(i)}\right) = \left(\overline{\boldsymbol{\vartheta}}_m^{(\ell)}, 1/I\right)$, $\ell = 1, \cdots, L$ indiquant les L particules à répliquer
    sinon conserver les échantillons $\left(\overline{\boldsymbol{\vartheta}}_m^{(i)}, \overline{W}_m^{i}\right) = \left(\boldsymbol{\vartheta}_m^{(i)}, W_m^{(i)}\right)$
fin pour

La loi jointe p_m peut être approchée par les réalisations obtenues après l'étape de rééchantillonnage par :

$$p_m(\mathrm{d}\boldsymbol{\vartheta}_{1:m}) \simeq \sum_{i=1}^{I} \overline{W}_m^{(i)} \delta_{\overline{\boldsymbol{\vartheta}}_{1:m}^{(i)}}(\mathrm{d}\boldsymbol{\vartheta}_{1:m}). \tag{1.12}$$

Les résultats de convergence pour les méthodes de Monte Carlo séquentielles sont présentés dans [CDL99].

Dans la suite du manuscrit, les particules $(\overline{\boldsymbol{\vartheta}}_m, \overline{W}_m)$ seront simplement notées $(\boldsymbol{\vartheta}_m, W_m)$.

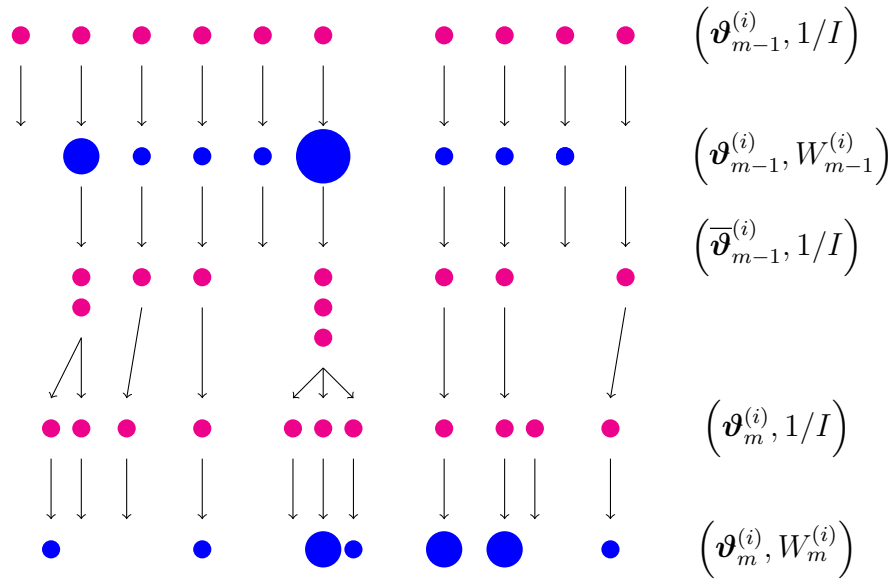


FIGURE 1.2 – Illustration du processus de rééchantillonnage pour l'algorithme de Monte Carlo séquentiel. Soit à l'état $m-1$ les paramètres $\boldsymbol{\vartheta}_m^{(i)}$ associés à chaque particule $i, i = 1, \cdots, I$ de même poids $1/I$. Les poids $W_m^{(i)}$ sont ensuite calculés. Plus un poids est élevé et plus le diamètre de son cercle représentatif est grand. La dégénérescence est illustrée par la disparition de certains poids, donc de certaines particules, par exemple la première et l'avant-dernière pour l'état $m-1$. A l'étape de rééchantillonnage, les particules sont redistribuées proportionnellement à leur poids associé, soit le passage de $\left(\boldsymbol{\vartheta}_{m-1}^{(i)}, W_m^{(i)}\right)$ à $\overline{\boldsymbol{\vartheta}}_m^{(i)}, 1/I$.

1.5 Conclusion

Ce chapitre donne dans un premier temps une présentation générale d'algorithmes classiques pour l'échantillonnage qui peut être réalisé selon les distributions *a priori* ou *a posteriori* des variables d'intérêt.

Concernant l'échantillonnage *a priori*, des algorithmes de Gibbs et de Metropolis-Hastings sont proposés pour la simulation selon un champ de Markov aléatoire dans le chapitre 3. Pour le second cas, un algorithme de Gibbs est présenté pour générer des variables selon leur distribution *a posteriori* dans le chapitre 2. De plus, le travail présenté dans ce manuscrit porte sur la segmentation non paramétrique d'images et donc un échantillonnage de partitions, proposé selon un algorithme de Gibbs dans le chapitre 4.

Dans un second temps, l'algorithme de Monte Carlo séquentiel générique est introduit. Il est adapté pour l'estimation d'hyperparamètres dans le chapitre 4.

CHAPITRE 2

Modèles bayésiens non paramétriques

En probabilité, l'approche bayésienne se fonde sur la définition de quatre termes pour un ensemble de données dépendant d'un jeu de paramètres :

- la vraisemblance des données aux paramètres qui quantifie l'adéquation des données aux paramètres
- la loi *a priori* sur les paramètres qui modélise des hypothèses sur les propriétés des paramètres, la forme de cette distribution est souvent choisie pour des facilités de calcul
- la loi marginale qui est la somme sur toutes les valeurs de paramètres possibles de la loi jointe des données et paramètres, cette loi jointe est le produit de la vraisemblance et de la loi *a priori*
- la loi *a posteriori* des paramètres qui donne la distribution des paramètres pour un ensemble d'observations donné.

Le théorème de Bayes s'écrit :

$$\text{Loi } a \textit{ posteriori} = \frac{\text{Vraisemblance} \times \text{Loi } a \textit{ priori}}{\text{Loi marginale}}$$

Les modèles bayésiens sont utilisés dans différents thèmes de recherche pour résoudre des problèmes d'estimation et d'inférence. Nous nous intéressons en particulier à l'apprentissage automatique qui consiste à développer des algorithmes pour « apprendre » des paramètres explicatifs uniques à partir des observations. Dans le cas paramétrique, le nombre de ces paramètres est fixé. Néanmoins, une version adaptative où le nombre et la valeur des paramètres évoluent avec les observations est préférable. Les modèles bayésiens non paramétriques ont souvent été utilisés dans ce but ; « non paramétrique » ici, ne signifie pas sans paramètre mais plutôt avec une infinité. Un modèle bayésien non paramétrique est ainsi défini sur un espace de paramètres de dimension infinie. Il est nécessaire de noter que les modèles bayésiens non paramétriques permettent à l'espace des paramètres de grandir avec le nombre d'observations.

Les modèles bayésiens sont fortement liés à la notion d'*échangeabilité* ; dans le cas des modèles bayésiens non paramétriques, présentons l'*échangeabilité infinie*. Soit une séquence infinie de variables aléatoires $\vartheta_1, \vartheta_2, \cdots$. Cette séquence est dite *infiniment échangeable* si la distribution jointe de n'importe quel sous-ensemble fini de ces variables aléatoires est invariant par rapport à n'importe quelle permutation de ces variables. La distribution de variables échangeables est donc indépendante de leur numérotation, même si elles sont dépendantes les unes des autres. En outre, le théorème de De Finetti stipule qu'une séquence est infiniment échangeable si et seulement si la distribution jointe de tout sous-ensemble peut s'écrire comme une probabilité marginale dont la définition fait intervenir une mesure aléatoire $\mathbb{G}$ [Dia88] :

$$p(\vartheta_1, \vartheta_2, \cdots, \vartheta_N) = \int \prod_{n=1}^{N} p(\vartheta_n|\mathbb{G}) P(\mathrm{d}\mathbb{G}) \tag{2.1}$$

avec $P(\mathbb{G})$ la loi *a priori* sur la mesure $\mathbb{G}$ elle-même loi *a priori* sur les variables $\vartheta_1, \vartheta_2, \cdots$. L'équation (2.1) est équivalente à dire que pour modéliser des variables aléatoires infiniment échangeables, il est nécessaire de considérer des distributions *a priori* sur des mesures de probabilité inconnues. Dans le reste du document, lorsque le terme échangeable est utilisé, il est sous-entendu, infiniment échangeable.

Citons à présent quelques exemples de modèles bayésiens non paramétriques et leurs domaines d'application.

La régression consiste à modéliser à partir d'un ensemble d'apprentissage une fonction capable de prédire la valeur en sortie pour n'importe quelle entrée ; un modèle bayésien non paramétrique adapté est une distribution sur l'espace des fonctions continues. Le modèle communément utilisé est le processus gaussien (GP) [RW06], une distribution sur un ensemble infini de variables aléatoires, telle que la loi jointe pour un ensemble fini est une gaussienne multivariée.

Un autre exemple est celui d'un modèle à variables latentes de K composantes où chaque observation résulte de l'influence de plusieurs composantes. Une donnée est alors associée à un ensemble de variables latentes décrivant sa dépendance par rapport à chaque composante du modèle. Les algorithmes de recommandation sont des applications de cette approche. Dans le cas de films, les utilisateurs sont considérés comme des données et les films proposés comme des variables latentes. La connexion utilisateur-film est ensuite représentée sous forme de matrice binaire. Les modèles bayésiens non paramétriques associés sont le processus Beta (BP) et le processus de buffet indien (IBP) [GG05].

Enfin, la classification consiste à effectuer une partition des observations et une représentation classique est le modèle de mélange. Pour les modèles paramétriques, le nombre de classes est supposé connu et fixé ; par opposition [Nea00], la version non paramétrique comporte une infinité possible de classes. Le processus de Dirichlet (DP) est un modèle souvent utilisé à cet effet.

Deux modèles bayésiens non paramétriques sont présentés dans ce chapitre : le processus de Dirichlet et une généralisation, le processus de Dirichlet hiérarchique introduit pour la classification conjointe de plusieurs groupes de données. Pour chacun un algorithme de Gibbs est aussi décrit pour l'inférence des paramètres suivant leur loi *a posteriori*.

2.1 Processus de Dirichlet (DP)

Considérons un ensemble infini d'observations $\boldsymbol{x} = \{x_1, x_2, \cdots, x_\infty\}$ de densité inconnue $p(\boldsymbol{x})$. Cette dernière peut être choisie comme une loi de mélange, c'est-à-dire qu'elle s'écrit pour chaque observation $x_n, n = 1, 2, \cdots$:

$$p(x_n) = \int f(x_n|\vartheta_n)\mathrm{d}\mathbb{G}(\vartheta_n) \tag{2.2}$$

avec f la distribution paramétrique des observations conditionnellement aux paramètres $\boldsymbol{\vartheta} = \{\vartheta_1, \vartheta_2, \cdots, \vartheta_\infty\}$ de fonction de distribution $\mathbb{G}$.

Dans un cadre bayésien, une loi *a priori* $p(\mathbb{G})$ est proposée pour la distribution $\mathbb{G}$; $p(\mathbb{G})$ est alors une distribution sur l'espace des distributions. Les modèles bayésiens non paramétriques, dont le processus de Dirichlet, font partie de cette catégorie de distributions.

Le modèle hiérarchique associé au processus de Dirichlet, le processus de Dirichlet à mélange, décrit dans la suite, a en particulier été souvent utilisé pour l'estimation de densité et en classification ; nous nous placerons ici dans ce contexte.

La classification consiste à rassembler les données similaires en *classes*. Cependant, la similarité ne se juge que par la définition d'une mesure associée. En particulier, dans le domaine probabiliste, les observations d'une même classe ont un paramètre commun caractérisant leur distribution de probabilité. Introduisons les variables catégorielles $\boldsymbol{z} = \{z_1, z_2, \cdots, z_\infty\}$ appelées *étiquettes* indiquant la classe de chaque observation. Ainsi, pour la classe k, $\{z_n = k, \vartheta_n = \vartheta_k^\star\}$ avec $\vartheta_k^\star$ le paramètre unique de classe.

Définition 2.1 (Processus de Dirichlet (DP))
Soient $(\Theta, \mathcal{A})$ un espace mesurable, Υ une mesure de probabilité définie sur cet espace et α un nombre réel strictement positif. Le processus de Dirichlet noté $\mathrm{DP}(\alpha, \Upsilon)$ est défini dans [Fer73] comme la distribution d'une mesure de probabilité aléatoire $\mathbb{G}$ sur $(\Theta, \mathcal{A})$ telle que, pour toute partition finie et mesurable $(A_1, A_2, \cdots, A_r)$ de Θ, le vecteur aléatoire $(\mathbb{G}(A_1), \cdots, \mathbb{G}(A_r))$ est distribué suivant une loi de Dirichlet à dimension finie de paramètres $(\alpha\Upsilon(A_1), \cdots, \alpha\Upsilon(A_r))$,

$$(\mathbb{G}(A_1), \cdots, \mathbb{G}(A_r)) \sim \mathrm{Dir}(\alpha\Upsilon(A_1), \cdots, \alpha\Upsilon(A_r)).$$

La notation adoptée pour la mesure de probabilité $\mathbb{G}$ est $\mathbb{G} \sim \mathrm{DP}(\alpha, \Upsilon)$.

Le processus de Dirichlet peut ainsi être défini comme une généralisation en dimension infinie d'une distribution de Dirichlet. Les propriétés de la distribution de Dirichlet se retrouvent par suite pour un processus de Dirichlet, par exemple la conjugaison. Les moments du premier et du second ordre s'expriment, pour tout ensemble $B \in \mathcal{A}$, par :

$$\mathbb{E}_{\mathrm{DP}(\alpha,\Upsilon)}[\mathbb{G}(B)] = \Upsilon(B) \tag{2.3}$$

$$V_{\mathrm{DP}(\alpha,\Upsilon)}[\mathbb{G}(B)] = \frac{\Upsilon(B)(1-\Upsilon(B))}{\alpha+1} \tag{2.4}$$

où $V_*[**]$ est la variance de $**$ suivant la distribution $*$. De ces équations on remarque que plus α est grand et plus les réalisations du DP sont différentes les unes des autres et on aura $\alpha \longrightarrow \infty, \mathbb{G}(B) \longrightarrow \Upsilon(B)$. Cependant, nous ne pouvons pas écrire $\mathbb{G} \longrightarrow \Upsilon$ car les réalisations du DP sont discrètes, comme présenté dans la suite.

2.1.1 Stick-breaking

Les réalisations d'un processus de Dirichlet sont discrètes avec une probabilité 1 [Fer73]. La construction stick-breaking introduite par [Set94] permet de mieux appréhender cette propriété.

Soient ω_k et $\vartheta_k^\star$ définis pour $k = 1, 2, \cdots$ par :

$$\omega_k^{'}|\alpha \sim \mathrm{Beta}(1,\alpha) \qquad \omega_k = \omega_k^{'}\prod_{\ell=1}^{k-1}(1-\omega_\ell^{'}) \qquad \vartheta_k^\star|\Upsilon \sim \Upsilon \tag{2.5}$$

La mesure $\mathbb{G}$ définie comme une somme infinie de mesures de Dirac centrées sur les paramètres de classe $\vartheta_k^\star$ et pondérées par les proportions ω_k est une réalisation du processus de Dirichlet de paramètres α et Υ :

$$\mathbb{G} = \sum_{k=1}^{\infty}\omega_k\delta_{\vartheta_k^\star} \tag{2.6}$$

La construction stick-breaking (2.5), comme son nom l'indique, revient à considérer le sectionnement d'un bâton de longueur 1 en morceaux successifs. Le premier morceau à couper est de longueur $\omega_1^{'} = \omega_1$. La longueur du morceau restant est donc $(1-\omega_1)$. De ce « nouveau » bâton est ôtée la fraction de longueur $\omega_2^{'}$. La longueur ω_2 du morceau coupé en considérant le bâton initial est donc $\omega_2^{'}(1-\omega_1)$. Les autres morceaux ω_k sont ainsi coupés successivement. Cela est illustré à la figure 2.1. Il est à noter que, par construction, $\sum_{k=1}^{\infty}\omega_k = 1$.

Reprenons le problème de classification présenté ci-dessus : de l'écriture (2.6), il résulte que les étiquettes sont distribuées suivant : $\boldsymbol{z} \sim \boldsymbol{\omega}$. Notons que la valeur prise par l'étiquette de classe z_n est sans importance, elle n'est utilisée que pour représenter une partition des données.

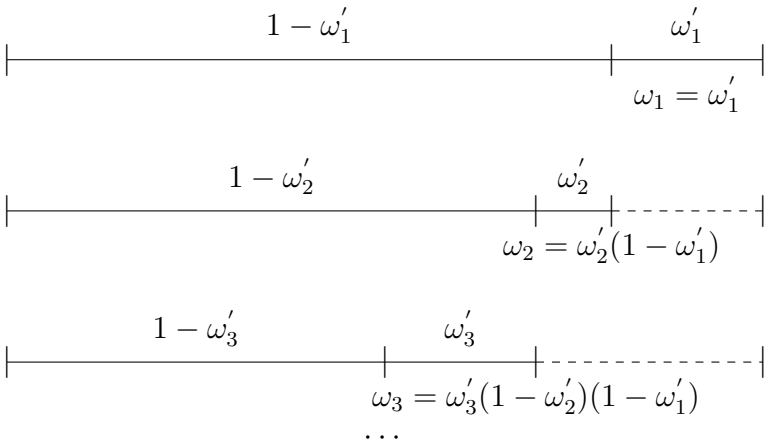


FIGURE 2.1 – Principe du stick-breaking

Des équations (2.5) et (2.6), un effet de *clustering*, partitionnement, s'observe : la mesure $\mathbb{G}$ étant discrète, différentes réalisations ϑ de $\mathbb{G}$ peuvent être égales. Cette propriété du processus de Dirichlet est illustrée par la métaphore du *processus de restaurants chinois (CRP)*.

2.1.2 Processus de restaurants chinois (CRP)

Soient $\vartheta_1, \cdots, \vartheta_N \overset{\text{iid}}{\sim} \mathbb{G}$ avec $\mathbb{G}$ une réalisation du processus de Dirichlet de paramètre scalaire α et de mesure de base Υ. De la conjugaison entre la loi de Dirichlet et la loi multinomiale, il est déduit que la distribution *a posteriori* de $\mathbb{G}$ est aussi un processus de Dirichlet [Fer73] :

$$\mathbb{G}|\boldsymbol{\vartheta}_{1:N} \sim \mathrm{DP}(\alpha', \Upsilon') \quad \text{avec} \quad \alpha' = \alpha + N \ \text{ et } \ \Upsilon' = \frac{\alpha}{\alpha+N}\Upsilon + \frac{1}{\alpha+N}\sum_{n=1}^{N}\delta_{\vartheta_n} \tag{2.7}$$

En se basant sur cette propriété, le processus de Dirichlet peut être représenté en urne de Pólya [BM73], ce qui permet d'observer un effet de partitionnement. En effet, selon (2.7) et en marginalisant $\mathbb{G}$, les paramètres $\boldsymbol{\vartheta}$ peuvent être récursivement échantillonnés suivant :

$$\vartheta_n|\vartheta_1, \cdots, \vartheta_{n-1}, \alpha, \Upsilon \sim \sum_{k=1}^{K}\frac{m_k}{\alpha+n-1}\delta_{\vartheta_k^\star} + \frac{\alpha}{\alpha+n-1}\Upsilon \tag{2.8}$$

Les paramètres $\vartheta_k^\star$ sont les valeurs uniques prises par les paramètres $\boldsymbol{\vartheta}$; ils ne correspondent pas forcément au paramètre de classe de même indice du stick-breaking, la notation $\vartheta_{i_k}^\star$ devrait être utilisée où i_k est l'indice correspondant à la position de cette réalisation dans la construction stick-breaking ; nous n'optons pas ici pour cette écriture par souci de simplicité. K est le nombre de classes existant jusqu'à l'instant $n-1$. m_k est le nombre paramètres ϑ_l ayant pris la valeur $\vartheta_k^\star$. Ce comportement du DP est décrit par la métaphore du restaurant chinois. Un restaurant

avec une infinité possible de clients, de tables et de repas à la carte est considéré. Lorsqu'un client ϑ_n entre dans le restaurant, il a une probabilité proportionnelle à m_k de s'asseoir à une table $k(\leq K)$ déjà occupée et proportionnelle à α de demander une nouvelle table. Si la table k est choisie, le client ϑ_n mangera le plat $\vartheta_k^\star$ partagé par ses occupants, $\vartheta_n = \vartheta_k^\star$. Pour une nouvelle table $K+1$, le plat associé est échantillonné suivant Υ : $\vartheta_{K+1}^\star \sim \Upsilon$ et $\vartheta_n = \vartheta_{K+1}^\star$. En introduisant les variables d'étiquette $\boldsymbol{z}$, (2.8) est équivalent à :

$$\begin{aligned} \Pr(z_n = k \leq K | \boldsymbol{z}_{1:n-1}) &= \frac{m_k}{\alpha + n - 1} \\ \Pr(z_n = k^{\text{new}} = K+1 | \boldsymbol{z}_{1:n-1}) &= \frac{\alpha}{\alpha + n - 1} \end{aligned} \tag{2.9}$$

La notation a^{new} est adoptée dans ce manuscrit pour désigner l'échantillonnage d'une nouvelle entité $a+1$; ceci pour une facilité d'écriture, en particulier dans les équations d'échantillonnage de Gibbs.

En classification, $\boldsymbol{z}$ correspond à une partition possible des données. La loi sur les partitions associée au modèle (2.9) est développée à l'annexe A et s'écrit :

$$\Pr(\boldsymbol{z}|\alpha, \boldsymbol{m}) = \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)} \alpha^K \prod_{k=1}^{K} \Gamma(m_k) \tag{2.10}$$

avec Γ la fonction gamma standard. En outre, on peut noter que cette probabilité est indépendante de la numérotation.

De (2.8) et (2.9), il est déduit que K peut augmenter avec le nombre de données et qu'il dépend en théorie exclusivement du nombre d'observations N et du paramètre scalaire du processus de Dirichlet α. En effet, la probabilité à un instant n donné que ϑ_n prenne une nouvelle valeur est proportionnelle à $\alpha/(\alpha + n - 1)$ impliquant que le nombre moyen de classes pour N et α fixés est [Ant74] :

$$\mathbb{E}[K|N, \alpha] = \sum_{n=1}^{N} \frac{\alpha}{\alpha + n - 1} \simeq \alpha \ln\left(\frac{N+\alpha}{\alpha}\right) \tag{2.11}$$

Il en résulte que pour $N \to \infty, K \to \infty$.

La figure 2.2, représente le nombre moyen de classes pour une grille de valeurs du paramètre α et du nombre d'observations N. On observe que le nombre moyen de classes croît de manière sous-linéaire (logarithmique) en fonction de α.

2.1.3 Extensions du Processus de Dirichlet

Différentes extensions du processus de Dirichlet ont été proposées [BF11], [PY97], [BGJ07] ; nous en présentons brièvement deux.

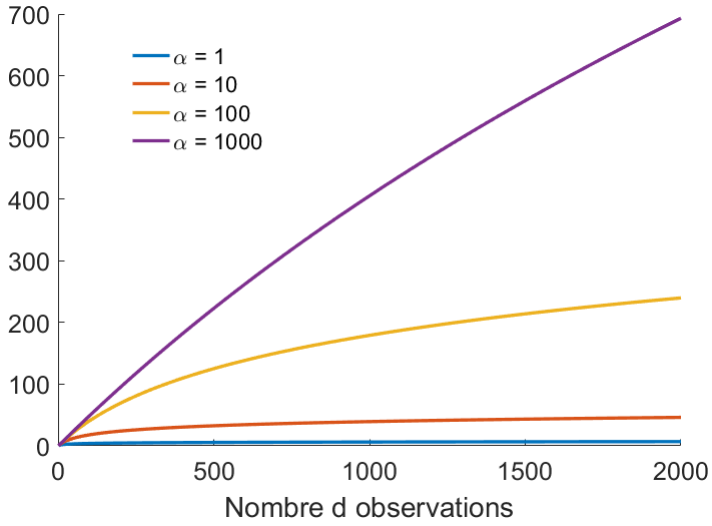


FIGURE 2.2 – Nombre moyen de classes pour différentes valeurs de α

Le *distance dependent Chinese restaurant process*

Contrairement au CRP classique, le processus de restaurant chinois dépendant de la distance (ddCRP) [BF11] propose d'effectuer des assignations de clients plutôt que des affectations de tables. Le processus génératif s'écrit :

$$\Pr(s_n = q|L, \varphi, \alpha) \propto \begin{cases} \varphi(\ell_{nq}) & \text{si } q \neq n \\ \alpha & \text{si } q = n \end{cases} \tag{2.12}$$

avec

- s_n l'étiquette d'assignation du client n et $\boldsymbol{s} = \{s_1, s_2, \cdots\}$
- ℓ_{nq} la distance entre les clients n et q
- L l'ensemble des distances
- φ une fonction de décroissance

De cette équation on peut déduire que l'assignation d'un client est indépendante de celle des autres. De plus, les clients que l'on peut joindre par leur assignation sont affectés à une même table et la partition en découlant est notée $\boldsymbol{z}(\boldsymbol{s})$, ce qui est illustré à la figure 2.3. Notons en particulier que la partition générée n'est pas échangeable.

Le processus de Pitman-Yor

Une autre généralisation du DP est le processus de Pitman-Yor [PY97] où les étiquettes sont échantillonnées suivant :

$$\Pr(z_n = k|\boldsymbol{z}_{1:n-1}, \alpha) = \frac{1}{n-1+\alpha} \begin{cases} n_k - d & \text{si } 1 \leq k \leq K \\ \alpha + dK & \text{si } k = k^{\text{new}} \end{cases} \tag{2.13}$$

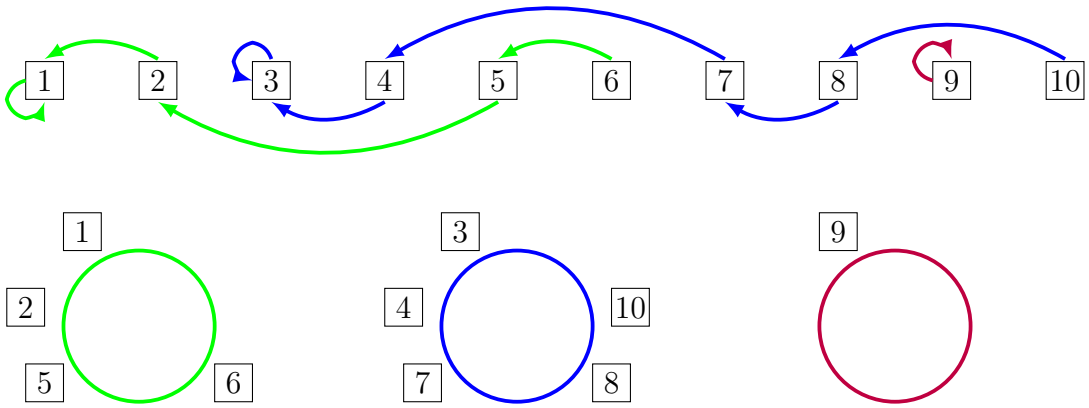


FIGURE 2.3 – Exemple de partition avec le ddCRP. La première ligne représente les assignations de clients, les couleurs donnent les clients qui ont la même assignation et la deuxième ligne donne la partition déduite.

où $0 \leq d < 1$ et le paramètre scalaire α respecte $\alpha > -d$. Le processus de restaurant chinois se retrouve lorsque $d = 0$. La représentation en stick-breaking correspondante est donnée par :

$$\begin{aligned} \omega_k^{'} &\sim \text{Beta}(1-d, \alpha + kd) & \omega_k &= \omega_k^{'} \prod_{i=1}^{k-1} (1-\omega_i^{'}) \\ \vartheta_k^{\star} &\sim \Upsilon & \mathbb{G} &= \sum_{k=1}^{\infty} \omega_k \delta_{\vartheta_k^{\star}} \end{aligned} \tag{2.14}$$

et $\sum_{k=1}^{\infty} \omega_k = 1$ presque sûrement si et seulement si $\sum_{k=1}^{\infty} \log(1 + a_k/b_k) = \infty$ (où $a_k = 1 - d$ et $b_k = \alpha + kd$) pour assurer que $\omega_k^{'}$ ne tend pas trop vite vers 0.

2.1.4 Processus de Dirichlet à mélanges

Dans cette partie, une interprétation alternative des processus de Dirichlet et de leur utilisation est proposée. Revenons tout d'abord à l'exemple de l'estimation de densité introduit à l'équation (2.2). Si un processus de Dirichlet est choisi comme distribution de $\mathbb{G}$, alors le modèle (2.2) peut se réécrire hiérarchiquement comme suit :

$$\begin{aligned} \mathbb{G}|\alpha, \Upsilon &\sim \text{DP}(\alpha, \Upsilon) \\ \vartheta_n|\mathbb{G} &\sim \mathbb{G} \\ x_n|\vartheta_n &\sim f(\cdot|\vartheta_n) \end{aligned} \tag{2.15}$$

et est appelé processus de Dirichlet à mélange.

Une représentation hiérarchique du processus de Dirichlet est donnée à la figure 2.4.

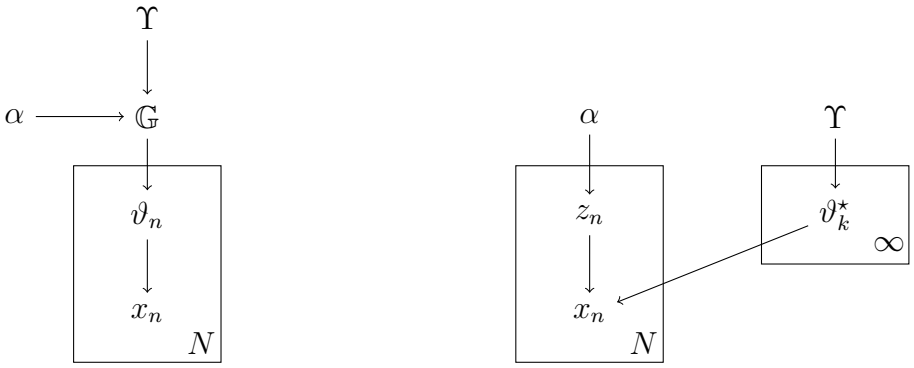


FIGURE 2.4 – Représentations hiérarchiques du processus de Dirichlet. A gauche est représenté le modèle hiérarchique du processus de Dirichlet à mélange et à droite une représentation équivalente du stick-breaking.

Limite infinie d'un mélange fini de distributions

Le processus de Dirichlet à mélange peut être défini comme la limite infinie d'un mélange fini de distributions où le nombre de composantes du mélange est à l'infini [Nea00], [Ras00], [IZ02].

Soit un mélange de K distributions avec $\boldsymbol{\omega} = \{\omega_1, \cdots, \omega_K\}$ l'ensemble des coefficients de proportion distribués selon une loi de Dirichlet de vecteur de paramètre symétrique de dimension K, $(\alpha/K, \cdots, \alpha/K)$. Soit $\vartheta_k^\star$ le paramètre associé à la composante k distribué selon Υ. L'ensemble des observations est noté $\boldsymbol{x} = \{x_1, \cdots, x_N\}$ et la distribution d'une observation est donnée conditionnellement au paramètre de sa composante associée. Soit de plus $\boldsymbol{z} = \{z_1, \cdots, z_N\}$ l'ensemble des étiquettes renseignant la composante associée à chaque observation. Le modèle obtenu est :

$$\begin{aligned} \vartheta_k|\Upsilon &\sim \Upsilon, \quad k = 1, \cdots, K \\ \boldsymbol{\omega}|\alpha &\sim \text{Dir}\left(\frac{\alpha}{K}, \cdots, \frac{\alpha}{K}\right) \\ z_n|\boldsymbol{\omega} &\sim \text{Mult}(\boldsymbol{\omega}), \quad n = 1, \cdots, N \\ x_n|z_n, \boldsymbol{\vartheta}^\star &\sim f(\cdot|\vartheta_{z_n}^\star), \quad n = 1, \cdots, N \end{aligned} \tag{2.16}$$

Il a été montré dans [Nea00] que ce modèle, lorsque $K \to \infty$ est équivalent au processus de Dirichlet à mélanges. Ce modèle a d'abord été proposé comme alternative aux mélanges finis où le nombre de composantes s'adapte aux observations [Ras00].

Vraisemblance marginale

Conditionnellement à la valeur de l'étiquette de classe $z_n = k$ et le paramètre unique associé $\vartheta_k^\star$, la distribution de l'observation x_n s'écrit : $f(x_n|\vartheta_k^\star)$. Par suite, la distribution jointe des observations peut se décomposer comme :

$$p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\vartheta}^\star) = \prod_{k=1}^{K} \left[\prod_{n \in A_k} f(x_n|\vartheta_k^\star) \right] \tag{2.17}$$

avec A_k les indices des observations affectées à la classe k. Dans le cas où seule la partition compte, une vraisemblance marginale peut ensuite être écrite en intégrant les paramètres de classe dans la densité jointe $p(\boldsymbol{x}, \boldsymbol{\vartheta}^\star|\boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\vartheta}^\star)\upsilon(\boldsymbol{\vartheta}^\star)$, avec υ la densité associée à Υ.

$$f(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} f(\boldsymbol{x}_{A_k}) = \prod_{k=1}^{K} \left[\int \prod_{n \in A_k} f(x_n|\vartheta_k^\star) \mathrm{d}\Upsilon(\vartheta_k^\star) \right] \tag{2.18}$$

Nous considérons ici que f et υ sont des lois conjuguées.

Le processus de Dirichlet à mélange peut être utilisé pour la classification d'un ensemble de données avec un nombre de classes inconnu. Dans d'autres problèmes de classification, les données à classifier peuvent plutôt appartenir des groupes de données. Ainsi chaque groupe a des observations propres. L'objectif dans ce cas est non seulement la partition de chaque groupe mais aussi l'identification des classes communes entre les groupes. Un exemple est la classification conjointe de différents types de documents, une approche paramétrique a été proposée dans [BNJ03]. Dans le cas non paramétrique, le *processus de Dirichlet hiérarchique* est une généralisation du processus de Dirichlet adaptée à la classification hiérarchique présentée dans [TJBB06].

2.2 Processus de Dirichlet hiérarchique (HDP)

Les notations ont été modifiées dans cette partie pour distinguer les cas mono-données et multi-données.

Soient J ensembles distincts de données $\boldsymbol{y}_j = \{y_{j1}, \cdots, y_{j\infty}\}$ et $\boldsymbol{y} = \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_J\}$. Les paramètres associés aux observation sont notés $\boldsymbol{\theta}_j = \{\theta_{j1}, \cdots, \theta_{j\infty}\}$ et $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J\}$. Les observations dans le groupe j notées $\boldsymbol{y}_j$ sont supposées être distribuées suivant un modèle de mélange du type (2.2). Les groupes sont de plus liés, c'est-à-dire qu'ils peuvent ou non avoir des caractéristiques communes.

Soit alors à définir un modèle permettant une classification partagée entre les groupes, c'est-à-dire diviser chaque groupe en sous-classes de paramètres uniques ψ_{jt} et regrouper les sous-

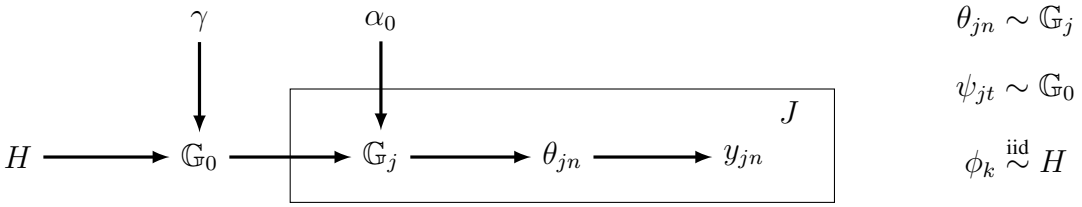


FIGURE 2.5 – Représentation du processus de Dirichlet hiérarchique

classes identiques entre les groupes en classes uniques de paramètres ϕ_k ; t est l'indice de la sous-classe dans le groupe j et k l'indice de classe pour l'ensemble. Pour une plus grande flexibilité du modèle, nous nous baserons sur le processus de Dirichlet précédemment introduit. Supposons que toutes les lois $\mathbb{G}_j$ sur les paramètres $\boldsymbol{\theta}_j$ soient modélisées par un processus de Dirichlet de même mesure de base continue $\mathbb{G}_0$. Conformémant à la définition du processus de Dirichlet, les valeurs uniques $\psi_{jt}, t = 1, \cdots, \infty$ prises par les θ_{jn} sont distribuées selon $\mathbb{G}_0$. Étant donné que la mesure de base est continue, la probabilité que des réalisations ψ_{jt} soient partagées entre les groupes est nulle, impliquant ainsi le non partage de classe. Il est donc nécessaire que la mesure de base $\mathbb{G}_0$ soit discrète. Pour à nouveau autoriser que le nombre de classes évolue avec les observations, la mesure $\mathbb{G}_0$ est elle aussi définie comme une réalisation d'un processus de Dirichlet. Ce modèle a été introduit comme le processus de Dirichlet hiérarchique [TJBB06]

$$\begin{aligned} \mathbb{G}_0|\gamma, H &\sim \mathrm{DP}(\gamma, H) \\ \mathbb{G}_j|\alpha_0, \mathbb{G}_0 &\overset{\text{iid}}{\sim} \mathrm{DP}(\alpha_0, \mathbb{G}_0) \quad j = 1, \cdots J \end{aligned} \tag{2.19}$$

avec $\mathbb{G}_j$ la mesure de probabilité aléatoire pour le groupe j définie comme un processus de Dirichlet de paramètre scalaire α_0 et de mesure de base $\mathbb{G}_0$ définie aussi comme un processus de Dirichlet de paramètre scalaire γ et de mesure de base H. De la définition (2.19), il suit que les mesures $\mathbb{G}_j$ sont indépendantes conditionnellement à $\mathbb{G}_0$, cela s'observe aussi sur la représentation du modèle à la figure 2.5.

Comme pour le DP, une version hiérarchique du HDP peut être écrite et se nomme processus de Dirichlet hiérarchique à mélange. Il est défini pour $j = 1, \cdots, J$ par :

$$\begin{aligned} \mathbb{G}_0|\gamma, H &\sim \mathrm{DP}(\gamma, H) \\ \mathbb{G}_j|\alpha_0, \mathbb{G}_0 &\overset{\text{iid}}{\sim} \mathrm{DP}(\alpha_0, \mathbb{G}_0) \\ \theta_{jn}|\mathbb{G}_j &\overset{\text{iid}}{\sim} \mathbb{G}_j \qquad\qquad n = 1, 2, \cdots, \infty \\ y_{jn}|\theta_{jn} &\overset{\text{iid}}{\sim} f(\cdot|\theta_{jn}) \qquad n = 1, 2, \cdots, \infty \end{aligned} \tag{2.20}$$

2.2.1 Stick-breaking

Conformément à la définition (2.19), $\mathbb{G}_0$ est distribué suivant un processus de Dirichlet et comme pour (2.6), une écriture en stick-breaking est possible :

$$\mathbb{G}_0 = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k} \tag{2.21}$$

avec $\phi_k \sim H$ et, π_k et ϕ_k sont respectivement le poids et le paramètre de la classe k. Comme vu à la section 2.1.1, les réalisations du DP sont discrètes, il en résulte que $\mathbb{G}_0$ est une mesure discrète de support les réalisations $\boldsymbol{\phi} = \{\phi_1, \cdots, \phi_\infty\}$. En outre, étant donné que les mesures aléatoires $\mathbb{G}_j$ partagent la même mesure de base $\mathbb{G}_0$, ils partagent le même support et s'écrivent donc :

$$\mathbb{G}_j = \sum_{k=1}^{\infty} \tau_{jk} \delta_{\phi_k} \tag{2.22}$$

L'échantillonnage des poids $\boldsymbol{\pi}$ et $\boldsymbol{\tau}$ se fait suivant [TJBB06] :

$$\begin{aligned} \pi_k^{'} &\sim \text{Beta}(1, \gamma) & \pi_k &= \pi_k^{'} \prod_{l=1}^{k-1} (1 - \pi_l^{'}) \\ \tau_{jk}^{'} &\sim \text{Beta}\left(\alpha_0 \pi_k, \alpha_0 \left(1 - \sum_{l=1}^{k} \pi_l\right)\right) & \tau_{jk} &= \tau_{jk}^{'} \prod_{i=1}^{k-1} (1 - \tau_{jt}^{'}) \end{aligned} \tag{2.23}$$

pour $j = 1, \cdots, J$ et $k = 1, \cdots, \infty$.

Les réalisations d'un processus de Dirichlet sont discrètes, ce qui implique que les paramètres ψ_{jt} distribués selon $\mathbb{G}_0$ se répètent, ainsi que les θ_{jn}. Un double partitionnement s'observe donc dans le cas du HDP qui est illustré par la métaphore de la *franchise de restaurants chinois (CRF)*.

2.2.2 Franchise de restaurants chinois (CRF)

Supposons une franchise comprenant J restaurants qui partagent le même menu $\boldsymbol{\phi}$. Chaque restaurant peut accueillir une infinité de clients et dresser une infinité de tables. Lorsqu'un individu θ_{jn} rentre dans un restaurant j, il a une probabilité proportionnelle à ν_{jt}, le nombre d'individus à la table t, de s'asseoir à cette table existante t et une probabilité proportionnelle à α_0 de s'asseoir à une nouvelle table :

$$\theta_{jn} | \alpha_0, \mathbb{G}_0 \sim \sum_{i=1}^{m_{j\cdot}} \frac{\nu_{jt}}{\alpha_0 + n - 1} \delta_{\psi_{jt}} + \frac{\alpha_0}{\alpha_0 + n - 1} \mathbb{G}_0 \tag{2.24}$$

avec $m_{j\cdot}$ le nombre de tables dans le restaurant j. Si le client s'assoit à une table existante $t \leq m_{j\cdot}$, il mangera le plat qui y est mangé ψ_{jt}. Dans le cas contraire, le plat associé doit être

échantillonné. A nouveau, ce plat peut être choisi dans la liste de plats uniques $\boldsymbol{\phi}_{1:K}$ existante dans la franchise ou un nouveau est échantillonné suivant H, K étant le nombre courant de plats dans le menu. $\psi_{jt} = \phi_k$ avec une probabilité proportionnelle à $m_{\cdot k}$, le nombre de tables dans la franchise ayant choisi le plat ϕ_k et $\psi_{jt} = \phi_{k^{\text{new}}} \sim H$ avec une probabilité proportionnelle au paramètre scalaire γ :

$$\psi_{jt}|\gamma, H, \boldsymbol{\psi}_{11:m_{1\cdot}}, \cdots, \boldsymbol{\psi}_{j1:t-1}, \cdots, \boldsymbol{\psi}_{J1:m_{J\cdot}} \boldsymbol{\phi}_{1:K} \sim \sum_{k=1}^{K} \frac{m_{\cdot k}}{\gamma + m_{\cdot\cdot}} \delta_{\phi_k} + \frac{\gamma}{\gamma + m_{\cdot\cdot}} H \tag{2.25}$$

où $m_{\cdot\cdot}$ est le nombre total de tables dans la franchise. Notons que comme pour le processus de Dirichlet, des indices différents devraient être introduits pour les équations (2.24) et (2.25). En effet, ψ_{jt} de l'équation (2.24) ne correspond pas au paramètre unique de sous-classe de même numéro échantillonné par stick-breaking, l'écriture devrait être $\psi_{j\ell_t}$ avec ℓ_t l'indice correspondant au paramètre de sous-classe t du stick-breaking ; de même, pour le paramètre de classe ϕ_{i_k}. Un exemple de réalisation de la franchise de restaurants chinois est donné à la figure 2.6.

Pour mieux appréhender le partitionnement effectué, deux jeux d'étiquettes sont ici introduits : c_{jn} l'étiquette qui renvoie le numéro de la table associée à l'individu n du restaurant j et d_{jt} qui donne le numéro du plat associé à la table t du restaurant j. La probabilité d'affectation de c_{jn} ne dépend que des affectations des $n-1$ premiers individus du restaurant j, elle est proportionnelle à ν_{jt} pour une table existante et à α_0 pour une nouvelle table. Ainsi, plus une table est populaire, plus la probabilité de s'y installer est forte. De même, la popularité d'un plat influence la probabilité que des tables le choisissent. Les équations (2.24) et (2.25) sont équivalentes à :

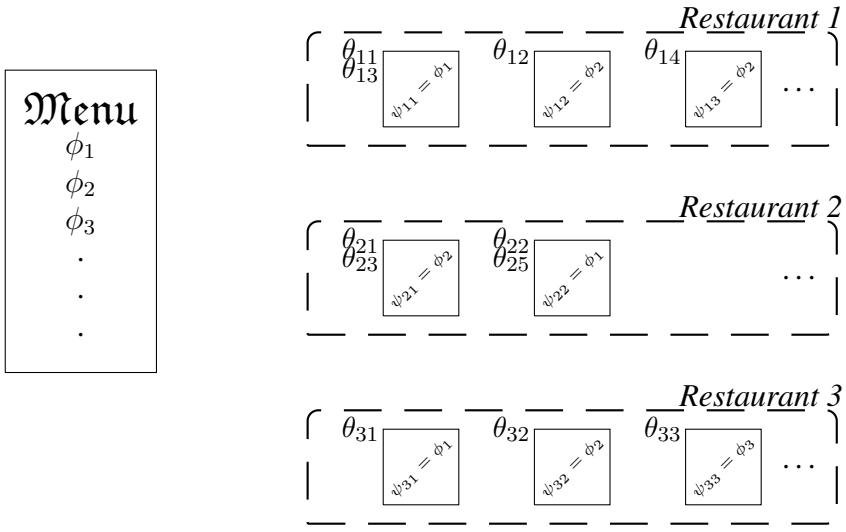
$$\begin{aligned} \Pr(c_{jn} = t|\alpha_0, \boldsymbol{c}_{1:n-1}) &\propto \begin{cases} \nu_{jt} & \text{si } t \leq m_{j\cdot} \\ \alpha_0 & \text{si } t = t^{\text{new}} = m_{j\cdot} + 1 \end{cases} \\ \Pr(d_{jt} = k|\gamma, \boldsymbol{d}_{11:m_{1\cdot}}, \cdots, \boldsymbol{d}_{j1:t-1}, \cdots, \boldsymbol{d}_{J1:m_{J\cdot}}) &\propto \begin{cases} m_{\cdot k} & \text{si } k \leq K \\ \gamma & \text{si } k = k^{\text{new}} = K + 1 \end{cases} \end{aligned} \tag{2.26}$$

La loi sur les partitions en résultant démontrée à l'annexe A s'écrit :

$$\varphi(\boldsymbol{c}, \boldsymbol{d}) = \prod_{j=1}^{J} \left\{ \left[\frac{\Gamma(\alpha_0)}{\Gamma(N_j + \alpha_0)} \right] \alpha_0^{m_{j\cdot}} \left[\prod_{i=1}^{m_{j\cdot}} \Gamma(\nu_{jt}) \right] \right\} \frac{\Gamma(\gamma)}{\Gamma(m_{\cdot\cdot} + \gamma)} \gamma^K \left[\prod_{k=1}^{K} \Gamma(m_{\cdot k}) \right]. \tag{2.27}$$

En faisant une analogie avec la classification, les tables désignent des *sous-classes* et les plats des classes. Ainsi c_{jn} indique la sous-classe de la donnée n dans le groupe j et d_{jt} la classe de la sous-classe t du groupe j. Le double partitionnement que nous avions comme objectif est ainsi observé dans le CRF. Chaque groupe est d'abord divisé en sous-classes puis les sous-classes statistiquement identiques entre les groupes sont affectées à la même classe.

Présentation de la franchise à un instant donné



Cas d'un nouveau client dans le restaurant 1

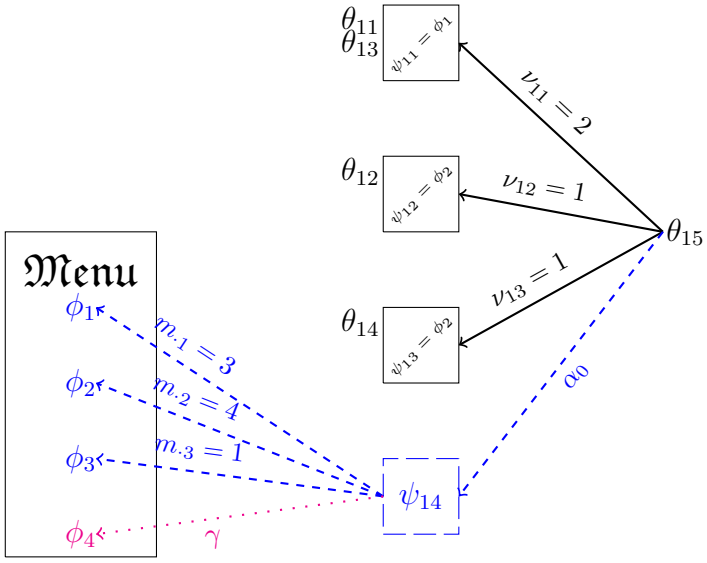


FIGURE 2.6 – Exemple d'une franchise de restaurants chinois (modèle *a priori*). En haut est présentée l'état de la franchise à un instant donné et en bas les possibilités d'affectation d'un nouveau client entrant dans le restaurant 1. La franchise comprend 3 restaurants et le menu au moment considéré 3 plats. Dans chaque restaurant, les θ accolés à une table représentent les clients assis à cette dernière, les ψ indiquent le plat du menu ayant été choisi. Dans le cas d'un nouveau client dans le restaurant 1, les flèches représentent les possibilités d'affectation et le chiffre au dessus la probabilité proportionnelle pour chaque cas. Ainsi, un nouveau client peut s'asseoir à une table existante ou à une nouvelle et pour cette nouvelle de choisir un plat existant ou un nouveau. On observe que plusieurs tables d'un même restaurant peuvent choisir le même plat : les tables 2 et 3 du restaurant 1 ont choisi le plat ϕ_2 et qu'à des tables dans différents restaurants peut être servi le même plat : $\psi_{11} = \psi_{22} = \psi_{31} = \phi_1$.

Loi *a posteriori*

Comme dans le cas mono-données (2.18), une vraisemblance marginale ne dépendant que de la partition $(\boldsymbol{c}, \boldsymbol{d})$ peut être écrite. Soit $f(y_{jn}|\phi_k)$ la densité de l'observation y_{jn} sachant sa classe affectée $d_{jc_{jn}} = k$ et le paramètre associé ϕ_k. La loi de l'ensemble des observations conditionnellement aux étiquettes et paramètres de classe s'écrit alors :

$$p(\boldsymbol{y}|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{\phi}) = \prod_{k=1}^{K} \left[\prod_{(j,n)\in A_k} f(y_{jn}|\phi_k) \right] \tag{2.28}$$

avec A_k les indices des observations dans tous les groupes dans la classe k. Intégrons à présent les paramètres de classe dans la distribution jointe $p(\boldsymbol{y}, \boldsymbol{\phi}|\boldsymbol{c}, \boldsymbol{d})$ pour obtenir une vraisemblance ne dépendant que de la partition $(\boldsymbol{c}, \boldsymbol{d})$:

$$f(\boldsymbol{y}|\boldsymbol{c}, \boldsymbol{d}) = \prod_{k=1}^{K} f(\boldsymbol{y}_{A_k}) = \prod_{k=1}^{K} \left\{ \left[\int \prod_{(j,n)\in A_k} f(y_{jn}|\phi_k) \right] \mathrm{d}H(\phi_k) \right\} \tag{2.29}$$

Il sera supposé dans la suite que les lois f et h, la densité de H, sont conjuguées.

La loi *a posteriori* est proportionnelle au produit de la loi *a priori* et de la vraisemblance des données. Pour des paramètres ayant comme loi *a priori* le processus de Dirichlet ou le processus de Dirichlet hiérarchique, la distribution *a posteriori* est difficile à explorer du fait de sa complexité. Les méthodes d'échantillonnage génériquement présentées au chapitre 1 sont alors utilisées pour simuler des échantillons suivant la loi d'intérêt.

2.3 Algorithmes d'inférence pour le DP et le HDP

Dans cette partie sont présentés deux algorithmes de Gibbs pour l'échantillonnage de la densité *a posteriori* sur les étiquettes et les paramètres, le premier pour le cas du processus de Dirichlet et le second pour le processus de Dirichlet hiérarchique. Une première approche est de les échantillonner conjointement, une autre méthode consiste à d'abord échantillonner les étiquettes puis l'échantillonnage des paramètres est facile à déduire. Notons aussi que, suite à l'échangeabilité, par définition, de la séquence échantillonnée, l'indice courant est considéré comme la dernière réalisation, ce qui permet d'avoir recours aux écritures en urnes de Pòlya (2.9) et (2.26).

2.3.1 DP : algorithme de Neal

Les variables inconnues du modèle présenté en (2.2) sont la mesure aléatoire $\mathbb{G}$ et les paramètres $\boldsymbol{\vartheta}$; la loi *a posteriori* à estimer est donc $p(\mathbb{G}, \boldsymbol{\vartheta}|\boldsymbol{x})$ ou de façon équivalente $p(\boldsymbol{\omega}, \boldsymbol{\vartheta}^\star, \boldsymbol{\vartheta}|\boldsymbol{x})$.

L'algorithme proposé dans [EW95] permet d'échantillonner, à l'aide des méthodes de Monte Carlo par chaînes de Markov, la densité *a posteriori* cible en marginalisant suivant la mesure aléatoire $\mathbb{G}$: $p(\boldsymbol{\vartheta}|\boldsymbol{x})$. Bien que les pré-requis soient assurés pour la convergence des chaînes générées, elle est assez lente. Un algorithme pour l'estimation conjointe de la partition et des paramètres uniques de classe a ensuite été proposé dans [Nea00], la densité estimée est ici $p(\boldsymbol{z}, \boldsymbol{\vartheta}^\star|\boldsymbol{x})$ via un algorithme de Gibbs. Le principe est d'accélérer la convergence en mettant à profit l'ensemble des observations attribuées à une même classe pour mettre à jour la partition. A une itération, étant donné la définition d'une vraisemblance marginale, les paramètres de classe peuvent être marginalisés pour l'inférence de la valeur des étiquettes $\boldsymbol{z}$. Sur la base de cette partition, les paramètres de classe sont ensuite déduits.

Puisque la séquence $\boldsymbol{z}$ est supposée échangeable, l'étiquette z_n peut être considérée comme la dernière à être échantillonnée. Conformément à l'écriture en urne de Pólya, (2.9), la distribution de z_n conditionnellement aux autres étiquettes s'écrit alors :

$$\Pr(z_n = k|\boldsymbol{z}^{-n}) = \frac{1}{\alpha + N - 1} \begin{cases} m_k^{-n} & \text{si } k \leq K \\ \alpha & \text{si } k = k^{\text{new}} \end{cases}$$

avec $\boldsymbol{z}^{-n} = z_1, \cdots, z_{n-1}, z_{n+1}, \cdots, z_N$ l'ensemble des étiquettes hormis la n-ième, m_k^{-n} le nombre d'observations dans la classe k en ôtant l'observation n et K le nombre courant de classes.

Ecrivons à présent la loi *a posteriori* conditionnelle de l'étiquette z_n sachant les autres variables du modèle. A l'aide du théorème de Bayes on a :

$$\Pr(z_n = k|\boldsymbol{z}^{-n}, \boldsymbol{x}) \propto p(x_n|z_n = k, \boldsymbol{z}^{-n}, \boldsymbol{x}^{-n}) \Pr(z_n = k|\boldsymbol{z}^{-n}).$$

En remplaçant chaque terme, il s'ensuit que :

$$\Pr(z_n = k|\boldsymbol{z}^{-n}, \boldsymbol{x}) \propto \begin{cases} m_k^{-n} f(x_n|x_{A_k^{-n}}) & \text{si } k \leq K \\ \alpha f(x_n) & \text{si } k = k^{\text{new}} \end{cases} \tag{2.30}$$

avec A_k^{-n} l'ensemble des indices $z_i = k, i \neq n$. $f(x_n|x_{A_k^{-n}})$ est la densité de l'observation x_n conditionnellement aux observations attribuées à la classe k sauf x_n qui peut s'écrire comme $f(x_n|x_{A_k^{-n}}) = f(x_n, x_{A_k^{-n}})/f(x_{A_k^{-n}})$ et $f(x_n) = \int f(x_n|\vartheta_{k^{\text{new}}}) \mathrm{d}\Upsilon(\vartheta_{k^{\text{new}}})$.

Lorsque la partition est obtenue, les paramètres uniques de classe sont mis à jour suivant la densité

$$p(\vartheta_k^\star|\boldsymbol{z}, \boldsymbol{x}) \propto \upsilon(\vartheta_k^\star) \prod_{n \in A_k} f(x_n|\vartheta_k^\star) \tag{2.31}$$

Algorithme 2.1 Algorithme de Neal

Initialisation
  Échantillonner $\boldsymbol{z}^{(0)}$
  Échantillonner $\boldsymbol{\vartheta}^{\star(0)} \sim \Upsilon$
pour $i = 1, \cdots, I$ **faire**
  pour $n = 1, \cdots, N$ **faire**
    Échantillonner $z_n^{(i)} \sim p(z_n^{(i)} | z_1^{(i)}, z_2^{(i)}, \cdots, z_{n-1}^{(i)}, z_{n+1}^{(i-1)}, \cdots, z_N^{(i-1)}, \boldsymbol{y})$ selon (2.30)
  fin pour
  Échantillonner $\boldsymbol{\vartheta}^{\star}$ selon (2.31)
fin pour

2.3.2 HDP : estimation de la partition

La densité *a posteriori* complète à estimer pour le processus de Dirichlet hiérarchique est $p(\mathbb{G}_0, \mathbb{G}_1, \cdots, \mathbb{G}_J, \boldsymbol{\theta} | \boldsymbol{y})$.

Dans [TJBB06], un algorithme semblable dans le principe à celui précédemment détaillé pour le DP a été proposé pour le HDP. L'algorithme de Gibbs proposé se base sur la représentation en franchise de restaurants chinois (2.26) ; les étiquettes de sous-classe $\boldsymbol{c}$ sont échantillonnées puis les étiquettes de classe $\boldsymbol{d}$, et ensuite les paramètres uniques.

Introduisons tout d'abord les notations utilisées par la suite :
- $\boldsymbol{c}^{-jn}$ l'ensemble des étiquettes de sous-classes sauf la n-ième du groupe j
- $\boldsymbol{y}^{-jn}$ l'ensemble des observations sauf y_{jn}
- ν_{jt}^{-jn} le nombre d'étiquettes $\boldsymbol{c}^{-jn}$ associées à la sous-classe t du groupe j
- $m_{j\cdot}^{-jn}$ le nombre de sous-classes dans le groupe j après avoir ôté c_{jn}
- $\boldsymbol{y}_{jt}$ l'ensemble des observations dans la sous-classe t de l'image j
- $\boldsymbol{y}^{-jt}$ l'ensemble des observations sauf $\boldsymbol{y}_{jt}$: les observations associées à tout une table sont retirées
- $\boldsymbol{d}^{-jt}$ l'ensemble des étiquettes de classe associées aux sous-classes excepté la t-ième du groupe j
- $m_{\cdot k}^{-jt}$ le nombre d'étiquettes $\boldsymbol{d}^{-jt}$ associées à la classe k

Les équations de l'algorithme de Gibbs pour $\boldsymbol{c}$ et $\boldsymbol{d}$ s'écrivent pour chaque observation et chaque sous-classe :

$$\Pr(c_{jn} = t | \boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}) \propto \Pr(c_{jn} = t | \boldsymbol{c}^{-jn}, \boldsymbol{d}) \; p(y_{jn} | c_{jn} = t, \boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}^{-jn}) \tag{2.32}$$

$$\Pr(d_{jt} = k | \boldsymbol{c}, \boldsymbol{d}^{-jt}, \boldsymbol{y}) \propto \Pr(d_{jt} = k | \boldsymbol{c}, \boldsymbol{d}^{-jt}) \; p(\boldsymbol{y}_{jt} | \boldsymbol{c}, d_{jt} = k, \boldsymbol{d}^{-jt}, \boldsymbol{y}^{-jt}) \tag{2.33}$$

De plus, les séquences $\boldsymbol{c}$ et $\boldsymbol{d}$ sont échangeables par définition, on a alors $\Pr(c_{jn} = t | \boldsymbol{c}^{-jn}, \boldsymbol{d}) = \Pr(c_{jn} = t | \boldsymbol{c}^{-jn})$. En outre, les distributions conditionnelles $\Pr(c_{jn} =$

$t|\boldsymbol{c}^{-jn})$ et $\Pr(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt})$ s'écrivent :

$$\Pr(c_{jn} = t|\boldsymbol{c}^{-jn}) = \frac{1}{\alpha_0 + N_j - 1} \begin{cases} \nu_{jt}^{-jn} & \text{si } t \leq m_{j\cdot} \\ \alpha_0 & \text{si } t = t^{\text{new}} \end{cases}$$

$$\Pr(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) = \frac{1}{\gamma + m_{\cdot\cdot} - 1} \begin{cases} m_{\cdot k}^{-jt} & \text{si } k \leq K \\ \gamma & \text{si } K = k^{\text{new}} \end{cases}$$

En réécrivant la loi de c_{jn} conditionnellement à toutes les autres variables à l'aide du théorème de Bayes, les équations d'échantillonnage des étiquettes $\boldsymbol{c}$ sont :

$$\Pr(c_{jn} = t|\boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}) \propto \begin{cases} \nu_{jt}^{-jn} f(y_{jn}|\boldsymbol{y}_{A_{d_{jt}}^{-jn}}) & \text{si } t \leq m_{j\cdot}^{-jn} \\ \alpha_0 p(y_{jn}|c_{jn} = t^{\text{new}}, \boldsymbol{c}^{-jn}, \boldsymbol{d}) & \text{si } t = t^{\text{new}} \end{cases} \tag{2.34}$$

avec

$$p(y_{jn}|c_{jn} = t^{\text{new}}, \boldsymbol{c}^{-jn}, \boldsymbol{d}) = \sum_{k=1}^{K} \frac{m_{\cdot k}}{m_{\cdot\cdot} + \gamma} f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) + \frac{\gamma}{m_{\cdot\cdot} + \gamma} f(y_{jn}) \tag{2.35}$$

la vraisemblance associée au choix d'une nouvelle sous-classe pour l'observation y_{jn} qui est obtenue en intégrant suivant toutes les valeurs $d_{jt^{\text{new}}}$ possibles et $f(y_{jn}) = \int f(y_{jn}|\theta)\mathrm{d}H(\theta)$. Dans le cas où $t = t^{\text{new}}$, la classe associée doit aussi être échantillonnée :

$$\Pr(d_{jt^{\text{new}}} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt^{\text{new}}}) \propto \begin{cases} m_{\cdot k} f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{si } k \leq K \\ \gamma f(y_{jn}) & \text{si } k = k^{\text{new}} \end{cases} \tag{2.36}$$

Ainsi, l'observation n du groupe j peut être affectée à une sous-classe existante ($t \leq m_{j\cdot}$) ou à une nouvelle avec une probabilité proportionnelle au nombre d'observations dans t et la distribution de l'observation y_{jn} conditionnellement aux autres observations dans la classe associée à la sous-classe t dans le cas où $t \leq m_{j\cdot}$ et proportionnelle au coefficient α_0 et la probabilité conditionnelle de y_{jn} sachant qu'elle est affectée à une nouvelle sous-classe.

Suivant le même procédé, les équations d'échantillonnage des étiquettes de classe sont :

$$\Pr(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}, \boldsymbol{y}) \propto \begin{cases} m_{\cdot k} f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) & \text{si } k \leq K \\ \gamma f(\boldsymbol{y}_{jt}) & \text{si } k = k^{\text{new}} \end{cases} \tag{2.37}$$

De même, la sous classe t de l'image j peut être affectée à une classe existante ou une nouvelle.

Les paramètres uniques de classe peuvent aussi, au besoin, être échantillonnés suivant :

$$p(\phi_k|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{y}) \propto h(\phi_k) \prod_{(j,n)\in A_k} f(y_{jn}|\phi_k) \tag{2.38}$$

Dans cette partie ont été présentés des algorithmes d'échantillonnage des paramètres selon leur distribution *a posteriori* dans le cas du DP et du HDP. Intéressons-nous à présent dans ce contexte à l'estimation de densité. Les résultats sont détaillés dans le cas du processus de Dirichlet puis généralisés pour le processus de Dirichlet hiérarchique.

La densité $p(\boldsymbol{x})$ a été présentée à la section 2.1 comme une loi de mélange inconnue de densité de mélange $f(x_n|\vartheta_n)$ et de distribution de mélange $\mathbb{G}$ qui a ensuite été choisie comme une réalisation d'un processus de Dirichlet. Après avoir obtenu des échantillons $\boldsymbol{\vartheta}$ selon leur distribution *a posteriori*, une approximation de la densité $p(\boldsymbol{x})$ est équivalente à celle de la densité prédictive $p(x_{N+1}|\boldsymbol{x})$. Cette méthode est détaillée à la section suivante.

2.3.3 Estimation de la densité

Dans le cas mono-données, modélisé par le DP, l'estimation de $p(\boldsymbol{x})$ revient à approcher la densité prédictive d'une nouvelle observation x_{N+1} sachant les N observations connues $p(x_{N+1}|\boldsymbol{x})$:

$$p(x_{N+1}|\boldsymbol{x}) = \int f(x_{N+1}|\vartheta_{N+1})p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{x}) \tag{2.39}$$

où

$$p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{x}) = \int p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{\vartheta})p(\boldsymbol{\vartheta}|\boldsymbol{x})\mathrm{d}\boldsymbol{\vartheta}$$

Les paramètres $\boldsymbol{\vartheta}$ correspondent à ceux échantillonnés suivant leur distribution *a posteriori* sachant les observations $\boldsymbol{x}$, par exemple par la méthode présentée à la partie 2.3.1. La distribution prédictive $p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{\vartheta})$ est donnée par la représentation en CRP, (2.8). Pour chaque échantillon $\boldsymbol{\vartheta}^{(i)}, i = 1, \cdots, I$ de la chaîne MCMC, une nouvelle valeur est échantillonnée suivant la distribution prédictive $\mathrm{d}\vartheta_{N+1}|\boldsymbol{\vartheta} \sim p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{\vartheta})$. Ainsi, I nouvelles particules $\vartheta_{N+1}^{(i)}$ sont obtenues et permettent d'approcher $p(\mathrm{d}\vartheta_{N+1}|\boldsymbol{x})$ par :

$$p_I(\mathrm{d}\vartheta_{N+1}|\boldsymbol{x}) = \frac{1}{I}\sum_{i=1}^{I} \delta_{\vartheta_{N+1}^{(i)}}(\mathrm{d}\vartheta_{N+1})$$

L'estimation de la densité prédictive sur la base de ces réalisations s'écrit :

$$p(x_{N+1}|\boldsymbol{x}) \simeq \frac{1}{I}\sum_{i=1}^{I} f(x_{N+1}|\vartheta_{N+1}^{(i)}) \tag{2.40}$$

Le même raisonnement est suivi pour le processus de Dirichlet hiérarchique. Pour chaque groupe j une estimation de la densité prédictive d'une nouvelle observation $y_{j(N_j+1)}$ sachant les N_j précédentes $\boldsymbol{y}_j$ s'écrit :

$$p(y_{j(N_j+1)}|\boldsymbol{y}_j) \simeq \frac{1}{I}\sum_{i=1}^{I} f(y_{j(N_j+1)}|\theta_{j(N_j+1)}^{(i)}) \tag{2.41}$$

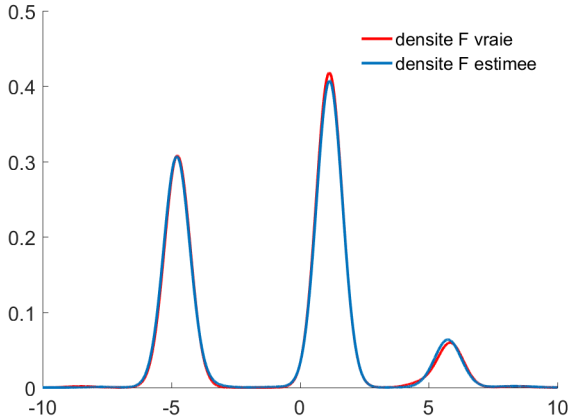


FIGURE 2.7 – Estimation de la densité dans le cas du DP avec $f(x_n|\vartheta_n) = \mathcal{N}(x_n; \vartheta_n, 0.5)$, $\Upsilon \equiv \mathcal{N}(0, 5)$, $N = 500$ et $\alpha = 1$

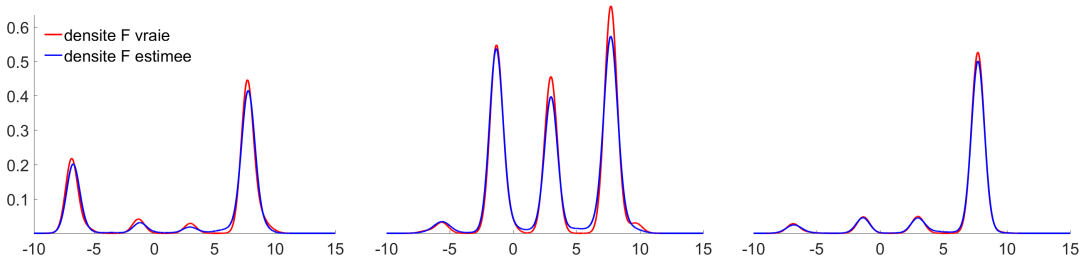


FIGURE 2.8 – Estimation de la densité dans le cas du HDP avec $f(y_{jn}|\theta_{jn}) = \mathcal{N}(y_{jn}; \theta_{jn}, 0.5)$, $H \equiv \mathcal{N}(0, 5)$, $J = 3$, $N = [500, 500, 500]$, $\alpha_0 = 10$ et $\gamma = 1$

où les $\theta^{(i)}_{j(N_j+1)}$ sont obtenus par construction CRF suite à l'obtention de réalisations $\boldsymbol{\theta}^{(i)}_j$ suivant leur distribution *a posteriori*.

Les figures 2.7 et 2.8 représentent les densités vraies et celles estimées d'un mélange de distributions gaussiennes généré suivant un CRP pour le cas mono-données et un CRF pour le cas multi-données. Les paramètres inconnus sont les moyennes des lois gaussiennes. On observe que les densités obtenues sont proches des densités vraies. On peut supposer que les valeurs estimées sont légèrement différentes des vraies, impliquant que les pics ne soient pas exactement superposés. A la figure 2.8, on observe de plus que certaines moyennes de lois gaussiennes sont partagées entre les groupes, conformément à la description du modèle génératif.

Pour les méthodes bayésiennes, l'efficacité de l'inférence est fortement liée au choix des valeurs des hyperpamètres des modèles ; ainsi, pour une complétion des méthodes proposées, il peut être utile d'estimer ces hyperparamètres. Par construction, des lois *a priori* conjuguées peuvent être trouvées pour α dans le cas du DP et α_0 et γ dans le cas du HDP. Un algorithme de Gibbs peut alors être dérivé pour obtenir des réalisations des hyperparamètres suivant leur distribution *a posteriori*. Cet algorithme peut être combiné à celui de l'échantillonnage des pa-

ramètres des modèles comme présenté dans [EW95] et [TJBB06]. Une méthode d'estimation des paramètres de concentration et ceux de la mesure de base est présentée dans la partie suivante.

2.3.4 Estimation des hyperparamètres

Paramètre de concentration α du processus de Dirichlet

Soient un ensemble d'observations $\boldsymbol{x}$, de paramètres associés $\boldsymbol{\vartheta}$ et le modèle hiérarchique défini par (2.15). Le nombre de classes K dépend du paramètre α et du nombre d'observations N, la probabilité associée est donnée par [Ant74] :

$$\Pr(K|\alpha, N) = \frac{s(N,K)\alpha^K}{\sum_{n=1}^{N} s(N,n)\alpha^n}.$$

$s(N,K)$ est la valeur absolue du nombre de Stirling de première espèce [AS65]. Étant donné que le paramètre de concentration α régule le nombre de paramètres uniques K et la probabilité de proposition de nouvelle classe, le choix de la meilleure valeur est donc primordial ; il peut être utile de l'estimer conjointement aux paramètres $\boldsymbol{\vartheta}$. Soit $p(\alpha)$ la loi *a priori* sur le paramètre scalaire α qui peut dépendre ou non du nombre d'observations N, sa loi *a posteriori* sachant tous les autres paramètres ne dépendrait donc que du nombre d'observations N et du nombre de paramètres uniques K,

$$p(\alpha|N, K, \boldsymbol{x}, \boldsymbol{\vartheta}) = p(\alpha|N, K) \propto \Pr(K|\alpha, N)p(\alpha) \tag{2.42}$$

En exploitant les conjugaisons de lois, il a été proposé une loi gamma pour $p(\alpha)$ [EW95], ce qui permet de dériver un algorithme de Gibbs combinant l'échantillonnage des paramètres $\boldsymbol{\vartheta}$ et α. D'autres lois *a priori* peuvent être choisies et la mise à jour du paramètre α s'intégrera à la procédure d'échantillonnage par un pas de Metropolis-Hastings.

En reprenant la démarche présentée dans [EW95], une loi *a priori* conjuguée a été proposée [TJBB06] pour les paramètres α_0 et γ du processus de Dirichlet hiérarchique.

Hyperparamètres de la mesure de base Υ

Soit $p(\xi)$ la loi *a priori* sur les hyperparamètres inconnus et à estimer ξ définissant la mesure de base Υ à estimer. La loi *a posteriori* des hyperparamètres ξ sachant les autres variables s'écrit [Nea00] :

$$p(\xi|\boldsymbol{x}, \boldsymbol{\vartheta}, \alpha) = p(\xi|\boldsymbol{\vartheta}^\star) \propto p(\xi) \prod_{k=1}^{K} \upsilon(\vartheta_k^\star|\xi) \tag{2.43}$$

avec $\boldsymbol{\vartheta}^\star$ l'ensemble des paramètres uniques. Si les distributions $\upsilon(\vartheta^\star|\xi)$ et $p(\xi)$ sont conjuguées, l'échantillonnage se fait par un algorithme de Gibbs, et par un algorithme de Metropolis-Hastings dans le cas contraire.

2.4 Conclusion

Ce chapitre présente les processus de Dirichlet et de Dirichlet hiérarchique. L'intérêt de ces modèles non paramétriques est leur capacité à s'adapter aux données fournies. Ils sont en particulier introduits pour la résolution de problèmes de classification et d'estimation de densité : le DP pour un ensemble de données et le HDP pour différents groupes de données. Quelques propriétés sont détaillées ainsi que des algorithmes de Gibbs pour l'échantillonnage des variables d'intérêt *a posteriori*. Ces modèles seront ensuite utilisés au chapitre 4 pour un cas particulier de classification : la segmentation.

CHAPITRE 3

Segmentation d'images

En traitement d'images, la segmentation est une étape préliminaire sur laquelle se fondent des questions de plus haut niveau comme l'analyse ou l'interprétation. Elle consiste en une partition des pixels de l'image en différents groupes appelés *classes* de sorte que les pixels d'une même classe partagent une même propriété, par exemple le niveau de gris, la couleur ou la texture. La segmentation est formalisée par l'attribution d'une variable d'étiquette à chaque pixel indiquant la classe à laquelle il est rattaché. Cette thèse est dédiée au développement de nouvelles méthodes de segmentation palliant certaines limitations de l'état de l'art. Ce chapitre permet d'introduire les modèles sur lesquels s'appuieront les approches proposées.

Il existe différentes méthodes de segmentation dont le choix est guidé par l'application. Nous citons ici les plus connues.

Une simple méthode est le seuillage, dans le cas où les classes sont caractérisées par des niveaux de gris différents. Comme son nom l'indique, elle consiste à appliquer un, voire plusieurs seuils aux niveaux de gris des pixels pour en déduire la segmentation. Néanmoins, une limitation est qu'aucune interaction spatiale n'est considérée. A noter que par extension, le même type de stratégie peut être appliquée non pas sur le niveau de gris mais sur des descripteurs caractérisant certaines propriétés comme la texture.

Les méthodes basées sur une approche région visent à regrouper les pixels présentant des propriétés communes. Les approches les plus utilisées peuvent être divisées en deux catégories désignées par les termes anglais *region-growing* et *split-and-merge* [Rob73]. *Region-growing* [AB94] qui peut être traduit par croissance de région est un processus itératif qui consiste à choisir un pixel source et à accumuler les voisins vérifiant une propriété, par exemple, la similarité de niveau de gris. *Split* ou segmentation par division consiste à d'abord définir un critère d'homogénéité, ensuite, le critère est évalué sur l'image, et, s'il n'est pas vérifié, elle est dé-

coupée en zones plus petites et la division est ainsi appliquée récursivement sur chaque zone. Puis, *merge* ou segmentation par fusion est l'exploration de l'image à partir de petites zones homogènes qui sont regroupées si elles vérifient un critère de similarité. Une autre famille de méthodes est la segmentation à partir de la détection de contours qui se caractérisent par un changement brutal de la fonction d'intensité dans l'image.

Plus récemment en vision, l'image a été assimilée à un graphe non orienté, la segmentation revenant ainsi à effectuer une partition de graphe. Différents algorithmes se fondent sur cette approche, en particulier l'algorithme *normalized-cuts* [SM00] que nous présenterons pour la division de nos images en super-pixels. Les champs de Markov reposent également sur l'assimilation de l'image à un graphe ; leur particularité est que la loi de l'étiquette de classe attachée à un pixel conditionnellement à toutes les autres étiquettes, ne dépend que de la valeur de l'étiquette des pixels voisins. En particulier, nous utilisons dans ce travail le champ de Potts.

Des algorithmes de classification sont aussi utilisés en segmentation. Par exemple, pour un nombre de classes donné, l'algorithme k-moyennes associe les pixels à la classe dont le barycentre est le plus proche [Mig08] alors que l'*Expectation-Maximization* propose la partition correspondant au maximum de vraisemblance [CFP03].

Le choix de la méthode dépend des caractéristiques de l'image et de l'objectif. Par exemple :

- les images sont-elles structurées, texturées ?
- les contours des objets sont-ils marqués ?
- les images sont-elles bruitées ?
- la segmentation doit-elle aboutir à un résultat visuellement acceptable ?
- la rapidité d'exécution peut-elle être sacrifiée au profit de la précision ?
- etc.

Notre étude s'inscrit dans un cadre bayésien et la segmentation est vue comme une classification des pixels ; nous choisissons alors le champ de Markov, comme loi des étiquettes. En outre, pour réduire la charge calculatoire, on applique une étape préliminaire de pré-segmentation qui consiste à subdiviser l'image en zones grossières. Cette approche permet entre autres de réduire la dimension du problème de segmentation. Selon la méthode, les zones obtenues sont cependant plus ou moins pertinentes. Les algorithmes de pré-segmentation utilisés dans notre travail sont aussi présentés par la suite.

3.1 Algorithmes de pré-segmentation

La pré-segmentation d'une image consiste à la subdiviser en zones cohérentes que nous appellerons *super-pixels* et qui seront à leur tour regroupées en classes par une méthode de segmentation plus fine. Il est nécessaire de noter que la méthode choisie influence la segmentation

finale de l'image. En effet, cette dernière sera une partition des super-pixels, et, remonter à la partition de pixels dépendra de la pré-segmentation.

Pour cela, il existe différents algorithmes souvent introduits dans le domaine de la vision par ordinateur par exemple pour la détection, la reconnaissance ou l'analyse.

Des approches basées sur la construction en graphe [SM00, FH04] ont été proposées où chaque pixel est considéré comme le nœud d'un graphe. Des liens sont de plus définis entre les nœuds et le poids associé à un lien est proportionnel à la similarité entre les pixels concernés. Les superpixels sont ensuite déduits par minimisation d'un coût défini sur le graphe.

Existent aussi des méthodes itératives pour la recherche des maxima reposant sur le calcul de gradient. Partant d'une segmentation initiale grossière, des algorithmes de gradient ascendant affinent la classification au cours des itérations pour une meilleure segmentation jusqu'à convergence [CM02, VS08].

Nous choisissons de présenter brièvement deux algorithmes de pré-segmentation que nous utiliserons dans la suite : le *normalized-cuts* [SM00] et le *Simple linear iterative clustering* (SLIC) [ASS+10].

Notons que les algorithmes de pré-segmentation pourraient aussi être utilisés pour la segmentation. Néanmoins, nous choisissons ici de les présenter indépendamment pour marquer la différence d'utilisation des méthodes dans le travail présenté.

3.1.1 *Normalized-cuts*

Définition 3.1 (Graphe non orienté)
Un graphe non orienté $G = (V, E)$ est un ensemble fini de nœuds V et d'arêtes E où chaque arête est une paire de nœuds.

L'image est considérée comme un graphe pondéré non dirigé $G = (V, E)$ et le poids du lien entre les nœuds n et q , $n, q = 1, \cdots, |V|; n \neq q$ est noté $w(n, q)$. Ce dernier est défini comme le « coût » associé à la liaison, par exemple la distance entre les nœuds considérés.

Dans cette configuration, effectuer une partition en deux groupes distincts A et B ($A \cup B = V$ et $A \cap B = \emptyset$) revient à supprimer les liens entre les nœuds de A et ceux de B. Le poids total des liens ainsi ôtés est défini comme le degré de dissimilarité. La quantité à minimiser en théorie des graphes s'écrit alors :

$$\mathrm{cut}(A, B) = \sum_{n \in A, q \in B} w(n, q)$$

Il a néanmoins été observé [WL93] que cette méthode favorise la création de groupes isolés composés d'un singleton. Pour contrer cela, une nouvelle mesure de dissociation entre groupes

est introduite [SM00] qui se calcule comme le ratio de la somme des fonctions de similarité par le nombre total de liens avec les autres nœuds du graphe. Ainsi, la nouvelle mesure appelée *normalized cut (*Ncut*)* s'écrit :

$$\mathrm{Ncut}(A,B) = \frac{\mathrm{cut}(A,B)}{\mathrm{assoc}(A,V)} + \frac{\mathrm{cut}(A,B)}{\mathrm{assoc}(B,V)} \tag{3.1}$$

avec $\mathrm{assoc}(P,V) = \sum_{n\in P, q\in V} w(n,q)$ le nombre total de connexions entre les nœuds dans P et ceux du graphe. En introduisant un autre jeu de notations, les auteurs de [SM00] montrent que minimiser cette mesure de dissociation revient à maximiser la mesure d'association des nœuds qui est un problème NP-complet. Ils proposent l'introduction d'une variable latente qui permet de trouver un minimum de manière itérative et une implémentation numériquement coûteuse. En pratique, nous utilisons les codes mis à disposition par les auteurs [SM00] qui prennent comme paramètre le nombre de super-pixels attendu.

Lorsque la partition désirée comprend plus de deux groupes, le graphe est d'abord divisé en deux puis chaque groupe est divisé en deux en suivant la même méthode. Le graphe est ainsi récursivement subdivisé jusqu'à l'obtention du nombre de groupes désiré.

3.1.2 *Simple linear iterative clustering* (SLIC)

Le *simple linear iterative clustering* (SLIC) [ASS+10] est un algorithme itératif s'inspirant de la méthode des k-moyennes [Mac67] avec une limitation de l'espace de recherche à une région de taille proportionnelle à la taille du super-pixel. La procédure consiste alors à trouver le centre de chaque super-pixel et les pixels appartenant à ce dernier. Le coût calculatoire associé est donc linéaire par rapport au nombre de pixels et indépendant du nombre de super-pixels. De plus, la distance utilisée combine les informations couleur et spatiale.

Notons N_{SP} le nombre de super-pixels désiré. Pour une image de N pixels, chaque super-pixel comprend alors en moyenne N/N_{SP} pixels. Définissons $S = \sqrt{N/N_{\mathrm{SP}}}$; on en déduit que la taille approximative d'un super-pixel est S^2.

Notons C_k le centre du super-pixel $k, k = 1, \cdots, N_{SP}$ de coordonnées $C_k = [l_k, a_k, b_k, x_k, y_k]$ avec (l_k, a_k, b_k) les coordonnées dans l'espace de couleurs CIELAB et (x_k, y_k) les coordonnées spatiales. A l'initialisation, les centres sont approximativement placés à une distance S. Puis, étant donné la surface moyenne occupée par un super-pixel, les pixels pouvant être associés à ce centre sont dans un espace maximal de $2S \times 2S$, qui représente l'espace de recherche. La

distance D de chaque pixel n dans l'aire de recherche à son centre associé C_k est :

$$\begin{aligned} d_{lab} &= \sqrt{(l_k - l_n)^2 + (a_k - a_n)^2 + (b_k - b_n)^2} \\ d_{xy} &= \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2} \\ D &= d_{lab} + \frac{m}{S} d_{xy} \end{aligned}$$

où m permet de contrôler la compacité du super-pixel, plus sa valeur est élevée, plus la proximité spatiale est importante et plus le super-pixel est petit.

Nous utilisons une version modifiée [ASS+12] où il n'est pas nécessaire de définir m ; la nouvelle distance à une itération i est :

$$D^{(i)} = \sqrt{\left(\frac{d_{lab}^{(i)}}{m_{lab}^{(i-1)}}\right)^2 + \left(\frac{d_{xy}^{(i)}}{m_{xy}^{(i-1)}}\right)^2}$$

avec $m_{lab}^{(i-1)}$ et $m_{xy}^{(i-1)}$ les mesures maximales de proximité en couleur et spatiale trouvées à l'itération précédente.

Nous utilisons la fonction *superpixels* disponible dans la *toolbox Image Processing* de Matlab et définie pour les deux approches SLIC présentées.

Un exemple comparatif de la pré-segmentation obtenue avec chacune des méthodes *normalized-cuts* et SLIC est donnée à la figure 3.1. On remarque que les résultats obtenus avec la première méthode fournit des super-pixels de forme plutôt arrondie contrairement à ceux obtenus avec la deuxième méthode.

3.2 Invariance aux rotations et à la luminance

Après un pré-découpage de l'image, on choisit un descripteur associé à chaque super-pixel : la moyenne ou l'histogramme des niveaux de gris sont des descripteurs classiques. Ces derniers constituent les entrées des algorithmes de classification/segmentation de haut niveau mis en œuvre par la suite.

La segmentation est sensible à la luminance et à l'orientation des objets dans l'image, un descripteur qui en assure une meilleure gestion est donc nécessaire. Différents descripteurs ont été proposés en vision par ordinateur, en particulier pour la détection d'objets [BM00, Low04, DT05]. La pertinence du choix d'un de ces descripteurs peut être évaluée dans notre problématique de segmentation. Nous nous intéressons en particulier au *Histogram of Oriented Gradient (HOG)* [DT05] qui est fondé sur la distribution de la norme du gradient et la direction des contours. Dans les développements présentés dans le chapitre suivant, nous le mettrons à profit comme descripteur de texture couplé à des descripteurs plus classiques de niveaux de gris.

FIGURE 3.1 – Exemple de découpage en environ 500 super-pixels de trois images issues de la base d'images Berkeley [1]. La première colonne représente les images originales, la seconde donne la pré-segmentation obtenue avec la méthode SLIC et la troisième avec *normalized-cuts*.

Concept

D'une manière générale dans la littérature, l'image est divisée en régions de petites tailles appelées *cellules*. Dans le travail présenté, les super-pixels obtenus après pré-segmentation sont considérés comme les cellules. Pour chaque cellule est ensuite calculé l'histogramme des directions du gradient ou des orientations des contours dans la cellule. La combinaison de ces histogrammes forme le descripteur HOG. De plus, pour de meilleurs résultats, les histogrammes locaux sont normalisés en contraste suivant la méthode expliquée ci-dessous. Des zones plus grandes que les cellules sont d'abord définies, les *blocs*, ici, un bloc est formé par la cellule d'intérêt et ses cellules directement voisines, c'est-à-dire avec lesquelles elle partagent une frontière. Puis, une mesure de l'intensité est calculée sur ces régions. On utilise ensuite cette valeur pour normaliser toutes les cellules du bloc. Cette normalisation permet une meilleure résistance aux changements d'illumination et aux effets d'ombres.

Construction du descripteur HOG

Etape 1 Calcul du gradient

Utilisation d'un filtre dérivateur $1-$D centré dans les directions horizontales et verticales, par exemple $[-1, 0, 1]$ et $[-1, 0, 1]^T$.

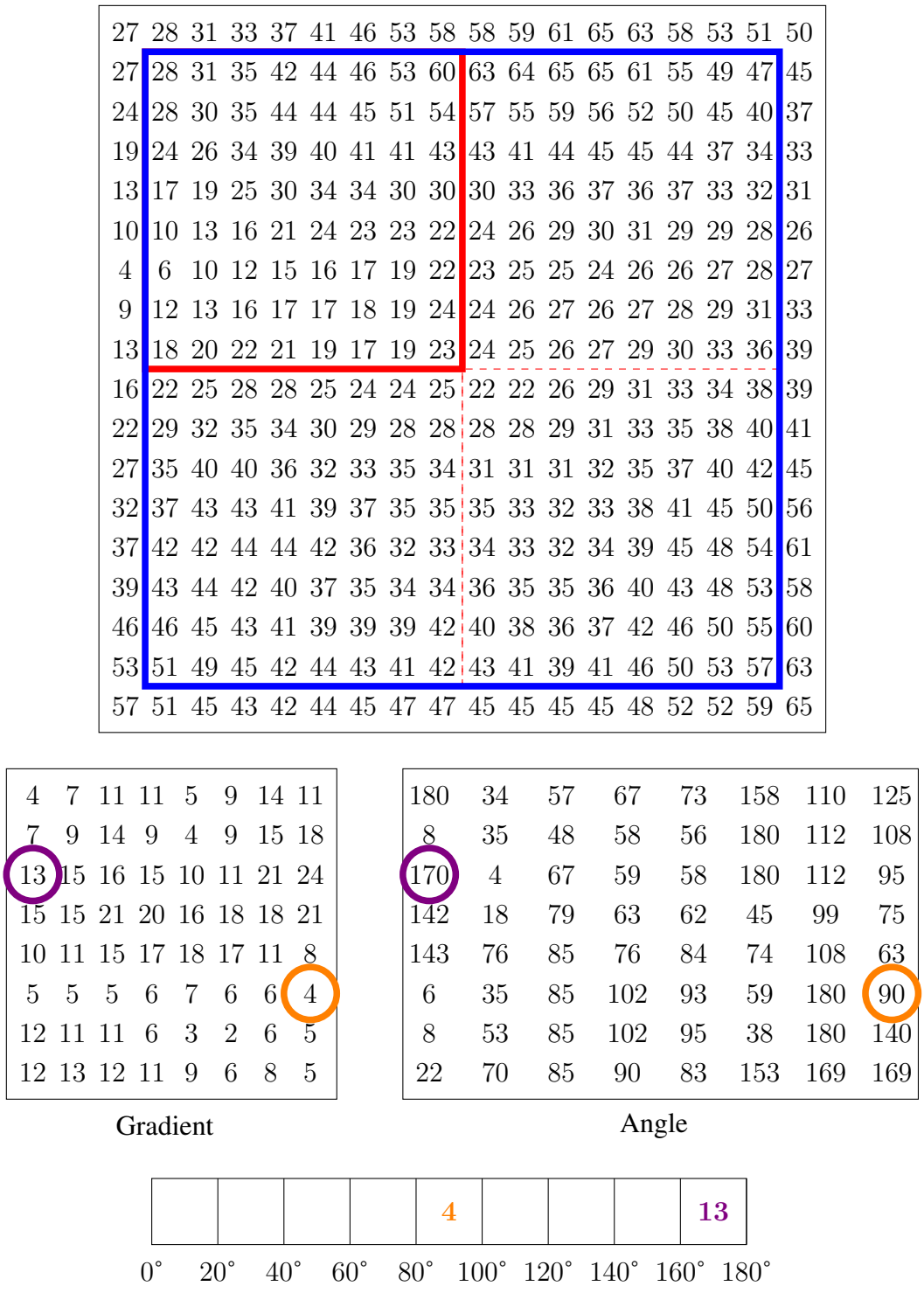


FIGURE 3.2 – Illustration de la construction des histogrammes de gradients orientés. Une cellule 8×8 est délimitée en rouge et un bloc 16×16 en bleu, ce bloc comprend 2×2 cellules. G_x et G_y sont obtenus en appliquant les filtres $[-1, 0, 1]$ et $[-1, 0, 1]^T$. Les gradient G et angle θ de la cellule en rouge sont calculés par : $G = \sqrt{G_x^2 + G_y^2}$ et $\theta = \arctan(G_y/G_x)$. Un histogramme de 9 bins est construit, il correspond à une subdivision de l'intervalle $[0°, 180°]$. Chaque gradient contribue au bin correspondant à son angle associé. Par exemple, le 4 entouré en orange contribue au bin $80 - 100°$centré en $90°$correspondant à son angle associé.

Etape 2 Construction de l'histogramme

Les histogrammes sont construits par vote : chaque pixel vote pour une classe de l'histogramme en fonction de l'orientation du gradient en ce point. Le vote du pixel est pondéré par l'intensité du gradient en ce point. Les histogrammes sont uniformes de 0°à 180°pour les angles non orientés sinon de 0°à 360°.

Etape 3 Normalisation des blocs

La normalisation consiste à concaténer les histogrammes présents dans le bloc puis de normaliser le vecteur.

Ce principe de construction est illustré à la figure 3.2 représentant un exemple sur un bloc de 16×16 comportant 4 cellules de taille 4×4.

Remarque 3.1
Pour un histogramme usuel, la valeur du vote serait 1 *alors que dans le cas du HOG, elle correspond à la valeur de la norme.*

Alors que les méthodes présentées dans cette partie serviront à un découpage grossier de l'image, nous détaillons dans la suite, le champ de Potts pour une segmentation plus fine. Nous nous plaçons en outre dans un cadre bayésien où l'image est décrite par un champ d'étiquettes.

3.3 Champ de Potts

Considérons une image de N pixels à segmenter en K classes et $\boldsymbol{z} = \{z_1, \cdots, z_N\}$ l'ensemble des étiquettes affectées aux pixels. Ainsi, à chaque pixel n est associée une étiquette z_n. Considérons de plus $\boldsymbol{z}$ comme la réalisation d'un champ aléatoire $\boldsymbol{Z}$. En outre, les pixels d'une classe partagent la même valeur d'étiquette, soit $Z_n = k$ pour les pixels de la classe k. Nous notons $S = \{1, \cdots, K\}$. Z_n prend alors ses valeurs dans S. Il s'ensuit que $\boldsymbol{Z}$ est à valeurs dans $\Omega = S^N$.

Le champ de Potts est largement utilisé en traitement d'images [GG84, KGK+07, AMD09, SWFU15] ; il s'agit d'un modèle stochastique utilisé comme loi *a priori* sur les étiquettes. Il a été introduit relativement à un graphe non orienté définissant ainsi des interactions locales dans l'image.

Définition 3.2 (Voisinage et clique [KSK76])
Notons $\mathbf{N} = \{1, 2, \cdots, N\}$ *l'ensemble des indices des pixels.*

Le système de voisinage est une collection $\mathcal{V} = \{\mathcal{V}_n, n \in \mathbf{N}\}$ *de sous-ensembles de* $\mathbf{N}$ *tels que*

- $n \notin \mathcal{V}_n$

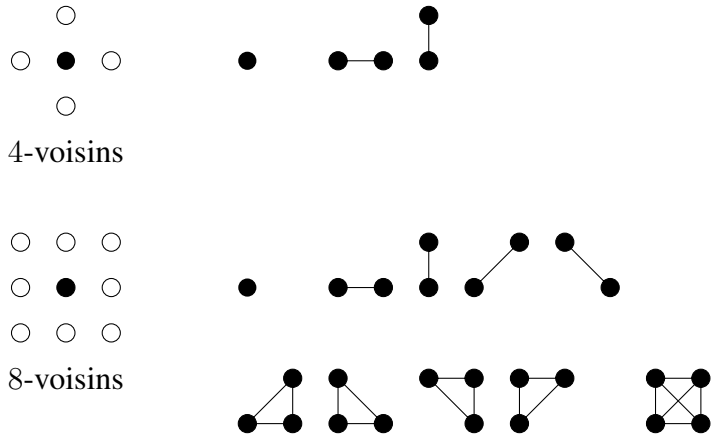


FIGURE 3.3 – Exemple de systèmes de voisinage. A gauche sont représentés les systèmes de voisinage : le pixel est un cercle plein et son voisinage par des cercles vides. A droite sont les cliques qui peuvent en être déduites : le nombre de cercles représente le nombre d'éléments de la clique et leur orientation la position spatiale des pixels concernés.

– $n \in \mathcal{V}_q$ *si et seulement si* $q \in \mathcal{V}_n, n, q \in \mathbf{N}$

Les pixels dans $\mathcal{V}_n$ *sont appelés voisins de* n*.*

Lorsque $\mathbf{N}$ *a un système de voisinage* $\mathcal{V}$*, un ensemble* c *est appelé une clique si* $\forall n, q \in c, n \neq q$ *alors* $q \in \mathcal{V}_n$*, c'est-à-dire si chaque couple de sites distincts dans* c *sont voisins. Notons* C *l'ensemble des cliques.*

Les cliques associées au système de voisinage définissent les relations spatiales nécessaires pour la définition du champ de Potts. Deux exemples de systèmes de voisinage ainsi que les cliques associées sont donnés à la figure 3.3.

Une caractéristique du champ de Potts est que la probabilité d'un pixel conditionnellement aux autres n'est fonction que des étiquettes associées aux pixels voisins :

$$\Pr(Z_n = z_n | \boldsymbol{Z}^{-n} = \boldsymbol{z}^{-n}) = \Pr(Z_n = z_n | \boldsymbol{Z}_{\mathcal{V}_n} = \boldsymbol{z}_{\mathcal{V}_n}) \tag{3.2}$$

avec $\boldsymbol{z}^{-n}$ l'ensemble des valeurs des étiquettes de tous les pixels de l'image excepté le pixel n et $\boldsymbol{z}_{\mathcal{V}_n}$ l'ensemble des étiquettes des pixels voisins de n.

Remarque 3.2

Le théorème de Hammersley-Clifford [Li09] établit une équivalence entre le champ de Markov aléatoire et le champ de Gibbs introduit à la section suivante. Il est détaillé car l'écriture du champ de Markov à partir de la mesure de Gibbs est plus aisée.

3.3.1 Champ de Potts et champ de Gibbs

Nous donnons dans cette partie les probabilités conditionnelles associées à un champ de Potts.

Les interactions locales dans l'image peuvent être décrites à l'aide de potentiels de cliques : à une clique c est attribuée un potentiel U_c qui dépend des étiquettes des pixels de la clique. Le potentiel global de l'image peut ensuite s'écrire : $U = \sum_{c \in C} U_c$ où C est l'ensemble des cliques.

La mesure de Gibbs de potentiel U est la probabilité définie par :

$$\Pr(\boldsymbol{Z} = \boldsymbol{z}) = \frac{1}{\Xi} \exp\left(-U(\boldsymbol{z})\right) = \frac{1}{\Xi} \exp\left(-\sum_{c \in C} U_c(\boldsymbol{z})\right) \tag{3.3}$$

Étant donné que le potentiel de cliques ne dépend pas de toute la configuration, mais uniquement de celle de la clique, on a, $U_c(\boldsymbol{z}) = U_c(z_q, q \in c)$. $\Xi = \sum_{\boldsymbol{z} \in \Omega} \exp\left(-U(\boldsymbol{z})\right)$ est la constante de normalisation appelée fonction de partition qui est souvent difficile à calculer. Pour des raisons de simplicité, $\Pr(\boldsymbol{Z} = \boldsymbol{z})$ sera noté dans la suite $\Pr(\boldsymbol{z})$.

Le potentiel U_c correspondant au champ de Potts est uniquement défini pour les cliques à deux éléments et s'écrit :

$$U_c(z_n, z_q) = \begin{cases} -\beta & \text{si } z_n = z_q \\ 0 & \text{si } z_n \neq z_q \end{cases} \tag{3.4}$$

La distribution associée s'écrit :

$$\Pr(\boldsymbol{z}) \propto \exp\left(\sum_{n=1}^{N} \sum_{q \in \mathcal{V}_n} \beta \mathbf{1}_{z_n, z_q}\right) \tag{3.5}$$

avec $\mathbf{1}_{z_n, z_q} = 1$ si $z_n = z_q$ et $\mathbf{1}_{z_n, z_q} = 0$ sinon. Le coefficient de granularité $\beta \in \mathbb{R}^+$ régule la taille des régions homogènes. La figure 3.4 représente des réalisations possibles de champs de Potts ; on observe que plus β est élevé et plus les régions créées sont grandes.

Ecrivons à présent les probabilités conditionnelles. Puisque le potentiel global d'un champ de Markov se décompose sous forme d'une somme de potentiels locaux, la probabilité conditionnelle locale $\Pr(z_n | \boldsymbol{z}^{-n})$ s'écrit :

$$\Pr(z_n | \boldsymbol{z}^{-n}) = \frac{\exp\left(-U_n(z_n | \boldsymbol{z}_{\mathcal{V}_n})\right)}{\sum_{z' \in S} \exp\left(-U_n(z' | \boldsymbol{z}_{\mathcal{V}})\right)} \tag{3.6}$$

où l'énergie locale U_n s'écrit :

$$U_n(z_n | z_q, q \in \mathcal{V}_n) = \sum_{c \in C | n \in C} U_c(z_n, z_q, q \in \mathcal{V}_n) = \sum_{c \in C | n \in C} U_c(z_n, \boldsymbol{z}_{\mathcal{V}_n})$$

Du fait de la fonction de partition, il n'est pas possible de calculer la probabilité (3.5) d'une configuration donnée $\boldsymbol{z}$. Néanmoins, grâce à l'expression (3.6), il est possible d'accéder aux probabilités conditionnelles locales qui seront utilisées pour l'échantillonnage du champ.

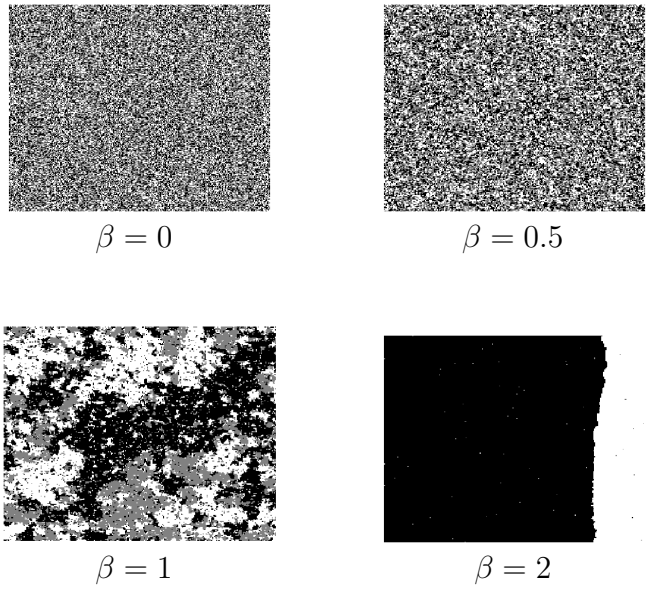


FIGURE 3.4 – Exemples de réalisations d'un champ de Potts pour une image 256×256, $K = 3$, pour un système de 4-voisins et pour différentes valeurs de β.

3.3.2 Échantillonnage *a priori* d'un champ de Potts

Après avoir défini le champ de Potts, nous proposons ici de tirer une réalisation de ce champ suivant trois méthodes de Monte Carlo : l'algorithme de Metropolis-Hastings et l'algorithme de Gibbs présentés dans le chapitre 1 et l'algorithme de Swendsen-Wang. Ces derniers peuvent aussi bien être utilisés pour simuler des données qu'adaptés ppour explorer des lois *a posteriori*.

L'échantillonnage consiste en la construction itérative d'une suite d'images qui, à convergence sont des réalisations tirées selon la probabilité $\Pr(\boldsymbol{z})$. La procédure consiste à mettre à jour les étiquettes au fil des itérations. Les deux premiers algorithmes ayant déjà été introduits, nous présentons ici uniquement les équations d'échantillonnage et détaillons le principe de l'algorithme de Swendsen-Wang qui permet d'accélérer la convergence de l'algorithme de Gibbs dans le cas où un champ de Potts est considéré comme loi *a priori*.

Algorithme de Gibbs

A l'itération i,

- choix d'un pixel n
- mise à jour de l'étiquette associée z_n suivant sa loi conditionnelle sachant toutes les autres étiquettes. L'écriture de la loi de simulation est donnée par l'équation (3.6) et ne

dépend que des étiquettes affectées aux pixels voisins :

$$\Pr(z_n|\boldsymbol{z}^{-n}) = \frac{\exp\left(-U_n(z_n|\boldsymbol{z}_{\mathcal{V}_n})\right)}{\sum_{z\in S}\exp\left(-U_n(z|\boldsymbol{z}_{\mathcal{V}_n})\right)}$$

Ainsi, pour chaque classe $k \in S$, on calcule $\exp(-U_n(z_n = k|\boldsymbol{z}_{\mathcal{V}_n})$ où $U_n(z_n = k|\boldsymbol{z} = \sum_{c\in C|n\in C} U_c(z_n = k, \boldsymbol{z}_{\mathcal{V}_n})$. La normalisation se fait ensuite en divisant par la somme des facteurs obtenus pour les différentes valeurs de k possibles.

Algorithme de Metropolis-Hastings

La procédure d'échantillonnage est classique alors que la mise à jour exploite le caractère markovien des étiquettes. Notons $z_n^{(i)}$ l'étiquette du pixel n à l'itération i.

A l'itération i,

- choix d'un pixel n
- tirage aléatoire d'une proposition $z^\star$ dans S selon une distribution q
- calcul de la probabilité d'acceptation, $\iota\left(z^\star, z_n^{(i-1)}\right)$:

$$\iota\left(z^\star, z_n^{(i-1)}\right) = \min\left(1, \frac{\Pr(z^\star|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)})}{\Pr(z_n^{(i-1)}|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)})}\frac{q(z_n^{(i-1)})}{q(z^\star)}\right)$$

Lorsque q est une loi uniforme, le ratio $q(z_n^{(i-1)})/q(z^\star) = 1$. La probabilité d'acceptation devient :

$$\iota\left(z^\star, z_n^{(i-1)}\right) = \min\left(1, \exp\left(-\left(U_n(z^\star|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)}) - U_n(z_n^{(i-1)}|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)})\right)\right)\right)$$

Ce qui revient à calculer la variation de potentiel $\Delta U = U_n(z^\star|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)}) - U_n(z_n^{(i-1)}|\boldsymbol{z}_{\mathcal{V}_n}^{(i-1)})$ et la mise à jour de $z_n^{(i)}$ se fait selon :

- si $\Delta U < 0$, $z_n^{(i)} = z^\star$
  si $\Delta U \geq 0$,

$$\begin{cases} z_n^{(i)} = z^\star & \text{avec la probabilité } \exp(-\Delta U) \\ z_n^{(i)} = z_n^{(i-1)} & \text{avec la probabilité } 1 - \exp(-\Delta U) \end{cases}$$

Algorithme de Swendsen-Wang

Les algorithmes de Gibbs et de Metropolis précédemment présentés peuvent converger lentement [BZ05]. Pour pallier ce problème, l'algorithme de Swendsen-Wang a été introduit [SW87]. Il introduit un jeu de variables latentes $\boldsymbol{r} = \{r_{nq}, n, q = 1, \cdots, N, n \neq q\}$ appelées *liens* définies comme $r_{nq} = 1$ si les pixels n et q sont liés et $r_{nq} = 0$ sinon. Ces liens binaires renseignent

sur la probabilité que deux pixels partagent la même étiquette. Ainsi, les pixels liés seront mis à jour conjointement lors de l'échantillonnage, ce qui permettra d'accélérer la convergence. L'introduction de $\boldsymbol{r}$ ne doit pas modifier la loi cible des étiquettes. A cet effet, les $\boldsymbol{r}$ sont définis au travers de leur loi conditionnelle $\Pr(\boldsymbol{r}|\boldsymbol{z})$. Ainsi par marginalisation de la loi jointe $\Pr(\boldsymbol{r},\boldsymbol{z}) = \Pr(\boldsymbol{r}|\boldsymbol{z})\Pr(\boldsymbol{z})$, on retrouve bien la loi marginale des étiquettes de classe.

Ecrivons la probabilité $\Pr(\boldsymbol{r}|\boldsymbol{z})$. Soit $\kappa = \beta \sum_{s\sim t} \mathbf{1}_{z_s,z_t}$ le potentiel du champ pour la réalisation $\boldsymbol{z}$, c'est-à-dire que la loi sur les étiquettes s'écrit $\Pr(\boldsymbol{z}) = \Xi^{-1}\exp(\kappa)$ et $\Xi = \sum_{\boldsymbol{z}} \exp(\kappa)$. Soient $\kappa_{n,q} = \beta \sum_{(s\sim t)\neq(n,q)} \mathbf{1}_{z_s,z_t}$ le Hamiltonien correspondant au retrait de l'interaction entre les pixels n et q, $\Xi^{\mathrm{s}}_{n,q} = \sum_{\boldsymbol{z}} \exp(\kappa_{n,q})\mathbf{1}_{z_n,z_q}$ et $\Xi^{\mathrm{i}}_{n,q} = \sum_{\boldsymbol{z}} \exp(\kappa_{n,q})$. Il suit que :

$$\Xi \propto (1-\exp(-\beta))\,\Xi^{\mathrm{s}}_{n,q} + \exp(-\beta)\,\Xi^{\mathrm{i}}_{n,q}$$

La formule précédente s'interprète ainsi : avec la probabilité $1-\exp(-\beta)$, les pixels n et q sont liés [SW87], c'est-à-dire $r_{nq}|z_n, z_q \sim \mathrm{Ber}(1-\exp(-\beta\mathbf{1}_{z_n,z_q}))$ où $\mathrm{Ber}(p)$ est la loi de Bernoulli de paramètre p.

Les pixels liés sont ensuite regroupés en *spin-clusters* ; pour un spin-cluster l, notons C_l l'ensemble de ses pixels. Une particularité de cet algorithme est que les étiquettes $\boldsymbol{z}_l$ des pixels dans C_l sont conjointement mises à jour selon une probabilité uniforme sur S, une illustration est donnée à la figure 3.5.

A chaque itération, l'algorithme comprend deux étapes : premièrement les liens sont échantillonnés et dans un deuxième temps, une configuration en classes est échantillonnée. La procédure d'échantillonnage consiste alors à l'itération i à :

- $\boldsymbol{r}^{(i)} \sim \Pr(\boldsymbol{r}|\boldsymbol{z}^{(i-1)})$
- $\boldsymbol{z}^{(i)} \sim \Pr(\boldsymbol{z}|\boldsymbol{r}^{(i)})$

Il a en outre été prouvé que la chaîne de Markov $\boldsymbol{z}^{(1)}, \cdots, \boldsymbol{z}^{(I)}$ est asymptotiquement distribuée selon $\Pr(\boldsymbol{z})$ [SW87].

Dans ce manuscrit, nous utilisons une généralisation proposée dans [Hig98] où un paramètre $\lambda > 0$ est introduit et la probabilité que deux pixels soient liés s'écrit :

$$r_{nq}|z_n, z_q \sim \mathrm{Ber}(1-\exp(-\beta\lambda\mathbf{1}_{z_n,z_q})) \tag{3.7}$$

Et les étiquettes $\boldsymbol{z}_l$ sont alors échantillonnées suivant :

$$\Pr(\boldsymbol{z}_l = k|\boldsymbol{r},\boldsymbol{z}^{-l}) \propto \exp\left(\sum_{\substack{q\in\mathcal{V}_n \\ n\in C_l, q\notin C_l}} \beta(1-\lambda)\mathbf{1}_{k,z_q}\right), \qquad k = 1,\cdots K \tag{3.8}$$

L'obtention de cette équation est donnée à l'annexe B.

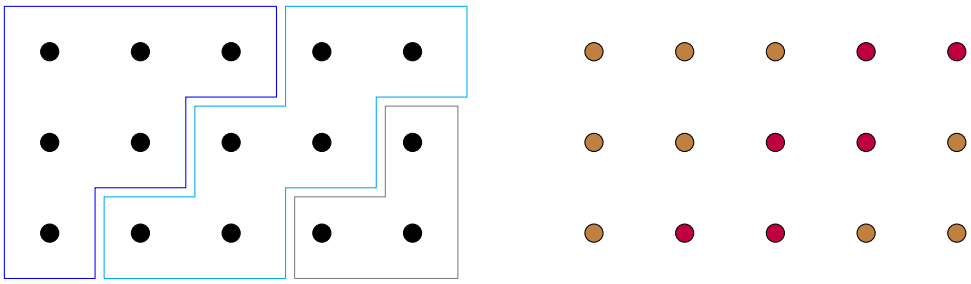


FIGURE 3.5 – Exemple de mise à jour d'étiquettes via l'algorithme de Swendsen-Wang. L'image initiale est représentée à gauche, les cadres représentent les *spin-clusters*. A droite la partition obtenue où chaque couleur représente une classe.

De (3.7), on remarque qu'à β fixé, plus λ augmente, plus les spin-clusters échantillonnés ont une taille importante. Et de (3.8), on a que plus la valeur de λ est élevée et moins il est probable que le spin-cluster change de classe. Il est alors important de choisir une valeur du paramètre λ qui assure la création de spin-clusters de taille raisonnable tout en permettant une exploration efficiente de la distribution des étiquettes. Un avantage de cet échantillonnage est que la chaîne d'étiquettes générée converge plus rapidement vers $\Pr(\boldsymbol{z})$ que celle générée par les algorithmes de Gibbs et de Metropolis. En outre, le paramètre λ ajoute de la flexibilité, ce qui est utile, en particulier pour pondérer suivant la valeur prise par β. Un inconvénient majeur souligné dans [Hig98], est que pour un choix inapproprié de λ, une infinité d'itérations est nécessaire pour que la chaîne converge. Ce comportement indésirable peut aussi être observé dans le cas $\lambda = 1$, c'est-à-dire pour le Swendsen-Wang classique.

Pour une complétion du modèle, l'estimation du paramètre β peut être envisagée ; pour ce faire, une loi *a priori* sur β est choisie. Cependant, une difficulté majeure est le calcul de la constante de normalisation Ξ : pour de petites images et un nombre de classes limité, son calcul numérique est possible, mais, dans la plupart des cas, cela est impossible. Une approche variationnelle avec une approximation de la fonction de partition est proposée dans [MTRP06]. Les méthodes de Monte Carlo sont de puissants outils pour l'inférence bayésienne lorsque les méthodes analytiques sont limitées. Il est alors proposé dans [MPRB06] d'échantillonner des variables aléatoires selon la densité non normalisée ; un algorithme de Metropolis-Hastings est ensuite défini de telle sorte que la constante de normalisation n'intervient pas dans le calcul de la probabilité d'acceptation. Sur le même principe, il est préconisé dans [PDBT13] de contourner le calcul de la fonction de partition en utilisant un algorithme d'approximation de la loi *a posteriori*.

3.4 Conclusion

Ce chapitre est une introduction à la segmentation d'images.

En premier lieu, la pré-segmentation qui est une sur-segmentation des images est détaillée, en particulier deux algorithmes sont présentés et utilisés pour les résultats présentés au chapitre 5 : le *normalized-cuts* et le *simple linear iterative clustering*.

Puis, le champ de Potts est présenté pour une segmentation bayésienne. La distribution des étiquettes associée est exprimée et des algorithmes d'échantillonnage sont proposés pour obtenir des étiquettes représentatives de la distribution. Dans le chapitre 4, le champ de Potts est utilisé pour construire le modèle proposé. Enfin, l'algorithme de Swendsen-Wang généralisé est présenté pour une exploration efficace de l'espace des partitions associé à un champ de Potts.

CHAPITRE 4

Segmentation bayésienne non paramétrique

Les méthodes de segmentation présentées au chapitre 3 présentent une restriction paramétrique : le nombre de classes doit être fixé avant la routine de segmentation. Ainsi, il est possible d'ajouter de la flexibilité au modèle, en considérant le nombre de classes inconnu et en l'estimant conjointement aux étiquettes de classes. Un algorithme de Monte Carlo à sauts réversibles [Gre95] peut être envisagé. Pour des données de grande dimension, il peut néanmoins rencontrer des difficultés à *sauter* efficacement dans les espaces de partitions possibles. Face à cette limitation, l'utilisation des méthodes bayésiennes non paramétriques en traitement d'images a suscité de l'intérêt ces dernières années [OB07, SJ08, GUSB11, ZCP+12].

Le principe adopté ici est de proposer une loi *a priori* sur les étiquettes qui est une combinaison de deux modèles. Le premier est un modèle bayésien non paramétrique : le processus de Dirichlet comme proposé dans [OB07, ACT+17] et le processus de Dirichlet hiérarchique dans notre cas. Ce modèle assure de ne pas fixer *a priori* le nombre de classes. Le second est un champ de Markov aléatoire, par exemple un champ de Potts, pour favoriser une homogénéité spatiale. Le modèle DP-Potts a été présenté dans [OB07] pour la segmentation d'une image et est détaillé à la partie 4.1. Nous proposons le modèle HDP-Potts pour la segmentation jointe d'un ensemble d'images. Ce type de segmentation est intéressant lorsque les images à étudier présentent des classes communes mais en proportions différentes.

La segmentation bayésienne non paramétrique connaît un intérêt grandissant. Par exemple, une version hiérarchique du *processus de restaurant chinois dépendant de la distance* a été introduit dans [GUSB11] pour la segmentation d'images naturelles. Le modèle est proposé pour la segmentation d'une image mais peut se généraliser pour un ensemble d'images. Dans le proces-

sus génératif, la première couche est une assignation de pixel suivant un processus de restaurants chinois dépendant de la distance, introduit au paragraphe 2.1.3, et définissant une partition en régions, puis les affectations des régions en classes se font selon un processus de restaurant chinois classique. Le modèle que nous proposons se démarque entre autres de celui-ci car nous étudions la pertinence de prendre en compte l'information spatiale pour le découpage en régions ou directement pour leur regroupement en classes. Pour la segmentation d'un ensemble d'images naturelles, [SJ08] propose un processus de Pitman-Yor hiérarchique (HPY) combiné à un processus gaussien ; ce modèle modifie l'écriture en stick-breaking où la nouvelle équation d'échantillonnage des poids non normalisés s'écrit à présent en fonction des lois normale et beta. De plus, la partition est induite par une approche variationnelle pour contourner le coût calculatoire induit par le recours à des méthodes d'échantillonnage.

La loi *a priori* sur les étiquettes est combinée à la fonction de vraisemblance des données pour écrire leur distribution *a posteriori*. L'exploration de cette loi est effectuée via un algorithme de Gibbs et les équations d'échantillonnage sont présentées. A l'instar de [XCD], nous proposons ensuite de modifier la routine d'exploration par un algorithme de Swendsen-Wang pour accélérer la convergence de la chaîne de Markov.

Enfin, le nombre de classes *a posteriori* dépend fortement du réglage des hyperparamètres du modèle ; nous proposons alors de les estimer de manière optimale par maximum de vraisemblance en mettant un œuvre un échantillonneur séquentiel de Monte Carlo.

Comme présenté dans le chapitre 3, il est assez courant en segmentation d'images d'effectuer au préalable une sur-segmentation des images en super-pixels. Les algorithmes définis ici restent valables qu'il y ait eu une étape de pré-segmentation ou pas. De plus, la démonstration des lois conditionnelles pour l'échantillonnage *a posteriori* des partitions selon notre modèle est détaillée en annexe.

4.1 Segmentation d'une image : algorithme DP-Potts

Dans cette partie est présentée une méthode de segmentation introduite dans [OB07] pour subdiviser une image en K classes, où K est inconnu. Considérons l'image à segmenter constituée de N pixels d'observations $\boldsymbol{x} = \{x_1, \cdots, x_N\}$ et de paramètres associés $\boldsymbol{\vartheta} = \{\vartheta_1, \cdots, \vartheta_N\}$. Nous introduisons de plus un ensemble d'étiquettes $\boldsymbol{z} = \{z_1, \cdots, z_N\}$ pour formaliser la notion de partition. L'étiquette z_n donne l'indice de la classe du pixel n (et de son observation x_n). En classification, chaque classe k, avec $k = 1, \cdots, K$ est associée à un paramètre $\vartheta_k^\star$ commun à toutes les observations dans la classe, $\{\vartheta_n = \vartheta_k^\star; z_n = k\}$. De plus, nous choisissons que les paramètres uniques soient distribués suivant Υ de densité υ.

Conditionnellement au paramètre ϑ_n, l'observation x_n est supposée distribuée selon $f(x_n|\vartheta_n)$.

Une propriété intéressante est qu'à partir de la vraisemblance $f(\cdot|\vartheta)$ et de la distribution *a priori* des paramètres de classe, une vraisemblance marginale des observations ne dépendant que de la partition peut être écrite :

$$f(\boldsymbol{x}|\boldsymbol{z}) = \prod_{k=1}^{K} f(\boldsymbol{x}_{A_k}) = \prod_{k=1}^{K} \left[\int \prod_{n \in A_k} f(x_n|\vartheta_k^\star)\, \upsilon(\vartheta_k^\star)\, \mathrm{d}\vartheta_k^\star \right]$$

où A_k est l'ensemble des pixels dans la classe k. Il est opportun de choisir υ et f conjugués pour obtenir une expression explicite de $f(\boldsymbol{x}|\boldsymbol{z})$.

Dans un cadre bayésien, la valeur des variables d'intérêt, ici les étiquettes, est déduite de leur distribution *a posteriori*. Ainsi, pour compléter le modèle, il reste à définir une loi *a priori* sur les étiquettes.

Le champ de Potts, défini à la partie 3.3, est un modèle bayésien couramment utilisé pour la segmentation d'images. Une de ses limitations, néanmoins, est que le nombre de classes K dans l'image doit être fixé, connu. Dans [OB07], il est proposé de le combiner à un modèle bayésien non paramétrique, en particulier le processus de restaurant chinois, pour contourner cette limite. L'idée est alors d'associer un processus favorisant une homogénéité spatiale à un second qui permet au nombre de classes d'évoluer avec les observations : le DP, introduit à la section 2.1 est alors défini comme une pénalité globale sur les valeurs des étiquettes et le champ de Potts comme une pénalité locale. Le terme « pénalité » renvoie ici à une notion de contrainte, poids. Ainsi, la pénalité globale est le poids affecté à chaque configuration d'étiquette dans le processus de Dirichlet et la pénalité spatiale la contrainte spatiale imposée par le champ de Potts. La loi *a priori* sur les étiquettes en découlant s'écrit alors :

$$\Pr(\boldsymbol{z}) \propto \underbrace{\Pr{}^{\mathrm{c}}(\boldsymbol{z})}_{\text{Pénalité globale}} \underbrace{\Pr{}^{\mathrm{p}}(\boldsymbol{z})}_{\text{Pénalité spatiale (locale)}} \propto \alpha^K \prod_{k=1}^{K} \Gamma(m_k) \prod_{n=1}^{N} \exp\left(\sum_{q \in \mathcal{V}_n} \beta \mathbf{1}_{z_q, z_n} \right) \quad (4.1)$$

avec $\mathbf{1}_{z_q,z_n} = 1$ si $z_q = z_n$ et $\mathbf{1}_{z_q,z_n} = 0$ sinon, $\mathcal{V}_n$ est l'ensemble des pixels voisins de n et β est le paramètre du champ de Potts. $\Pr{}^{\mathrm{p}}$ est la loi correspondant au champ de Potts et $\Pr{}^{\mathrm{c}}$ celle induite par le processus de restaurant chinois de paramètres α et Υ présenté à la partie 2.1.2. De plus, m_k est le nombre de pixels dans la classe k.

Ecrivons à présent la distribution *a posteriori* des étiquettes : $\Pr(\boldsymbol{z}|\boldsymbol{x}) \propto f(\boldsymbol{x}|\boldsymbol{z})\Pr(\boldsymbol{z})$. N'étant pas une distribution standard, un algorithme de Gibbs a été proposé pour son exploration. Il repose sur la distribution conditionnelle *a posteriori* de z_n qui s'écrit :

$$\Pr(z_n = k|\boldsymbol{z}^{-n}, \boldsymbol{x}) \propto \Pr{}^{\mathrm{c}}(z_n = k|\boldsymbol{z}^{-n})\, \Pr{}^{\mathrm{p}}(z_n = k|\boldsymbol{z}^{-n})\, p(x_n|\boldsymbol{x}^{-n}, \boldsymbol{z})$$

La distribution conditionnelle $\Pr{}^{\mathrm{c}}(z_n|\boldsymbol{z}^{-n})$ est donnée par la construction en urne de Pólya (2.9) :

$$\Pr{}^{\mathrm{c}}(z_n = k|\boldsymbol{z}^{-n}) \propto \begin{cases} m_k^{-n} & \text{si } k \leq K \\ \alpha & \text{si } k = k^{\mathrm{new}} \end{cases}$$

où m_k^{-n} est le nombre de pixels dans la classe k sans considérer le pixel n. Ainsi, il y a une probabilité non nulle de créer une nouvelle classe. Le terme correspondant au champ de Potts est :

$$\Pr{}^{\mathrm{p}}(z_n = k|\boldsymbol{z}^{-n}) \propto \exp\left(\sum_{q\in\mathcal{V}_n} \beta\, \mathbf{1}_{z_q,k}\right)$$

On remarque que dans le cas d'une nouvelle classe k^{new}, ce terme est proportionnel à 1, puisque par définition aucun pixel dans l'image ne peut être dans cette classe. En outre, on a

$$p(x_n|\boldsymbol{x}^{-n}, \boldsymbol{z}) = \begin{cases} f(x_n|\boldsymbol{x}_{A_k^{-n}}) & \text{si } k \leq K \\ f(x_n) & \text{si } k = k^{\mathrm{new}} \end{cases}$$

avec A_k^{-n} l'ensemble des pixels dans la classe k sauf le pixel n. $f(x_n|\boldsymbol{x}_{A_k^{-n}})$ est la distribution de l'observation x_n conditionnellement aux observations attachées à l'ensemble A_k^{-n} qui peut être écrite sous la forme : $f(x_n|\boldsymbol{x}_{A_k^{-n}}) = f(x_n, \boldsymbol{x}_{A_k^{-n}})/f(\boldsymbol{x}_{A_k^{-n}})$ et $f(x_n) = \int f(x_n|\varphi)\upsilon(\varphi)\mathrm{d}\varphi$ est la loi marginale de l'observation x_n. De ces équations, la distribution *a posteriori* conditionnelle de l'étiquette z_n est :

$$\Pr(z_n = k|\boldsymbol{z}^{-n}, \boldsymbol{x}) \propto \begin{cases} m_k^{-n} \exp\left(\sum_{q\in\mathcal{V}_n} \beta \mathbf{1}_{z_q,k}\right) f(x_n|\boldsymbol{x}_{A_k^{-n}}) & \text{si } k \leq K \\ \alpha f(x_n) & \text{si } k = k^{\mathrm{new}} \end{cases} \tag{4.2}$$

et l'algorithme de Gibbs associé est décrit à la table 4.1.

Algorithme 4.1 Algorithme de Gibbs pour l'échantillonnage de $\boldsymbol{z} \sim \Pr(\boldsymbol{z}|\boldsymbol{x})$

Initialisation $\boldsymbol{z}^{(0)}$
pour $i = 1, \ldots, I$ **faire**
  pour $n = 1, \ldots, N$ **faire**
    Échantillonner $z_n^{(i)} \sim \Pr(z_n|\boldsymbol{x}, \boldsymbol{z}_{1:n-1}^{(i)}, \boldsymbol{z}_{n+1:N}^{(i-1)})$ selon (4.2)
  fin pour
fin pour

Une flexibilité est observée dans la procédure d'échantillonnage présentée. En effet, d'une part, le pixel n est affecté à une classe k existante dans l'image proportionnellement au nombre de pixels lui étant affectés m_k^{-n} et, à travers le champ de Potts, au nombre de ses voisins ayant été affectés à cette classe. D'autre part, le pixel n peut aussi être dans une nouvelle classe proportionnellement à α, le paramètre scalaire du processus de Dirichlet.

Au besoin, les paramètres uniques de classe $\boldsymbol{\vartheta}^\star$ peuvent être échantillonnés directement à l'issue de l'algorithme MCMC suivant :

$$p(\vartheta_k^\star|\boldsymbol{x}, \boldsymbol{z}) \propto \upsilon(\vartheta_k^\star) \prod_{n\in A_k} f(x_n|\vartheta_k^\star) \tag{4.3}$$

Connaissant les paramètres de classe et la partition $\boldsymbol{z}$, la valeur des paramètres $\boldsymbol{\vartheta}$ peut ensuite être estimée.

Le modèle présenté dans cette partie a pour objectif la segmentation d'une seule image. Intéressons-nous à présent à la partition jointe d'un groupe d'images. Prenons l'exemple d'un ensemble d'images de ville et de campagne, les deux catégories sont composées des classes *verdure* et *habitation* en des proportions différentes ; une segmentation jointe permettrait de faciliter l'inférence de la classe *verdure* dans les images de ville où elle est moins représentée par *apprentissage* dans les images de campagne. Un autre exemple est une segmentation conjointe de coupes de cerveaux qui mettrait en évidence des zones problématiques d'intérêt dans une comparaison de cerveaux *sain* et *malade*. La segmentation d'un ensemble d'images s'avère alors intéressante. Il est cependant nécessaire de définir un modèle adéquat. Il doit permettre non seulement d'inférer la segmentation pour chaque image mais aussi la segmentation conjointe, c'est-à-dire identifier les classes partagées et leur proportion dans chaque image. Pour ce faire, nous proposons de nous baser sur le processus de Dirichlet hiérarchique. Une démarche comparable a également été considérée dans le papier de conférences [NHSM12] mais n'a pas donné lieu à des développements ultérieurs.

4.2 Segmentation jointe d'un ensemble d'images : algorithme HDP-Potts

Pour un ensemble J d'images avec des caractéristiques communes, la segmentation conjointe donne à la fois la partition de chaque image et la partition commune. Notons N_j le nombre de pixels dans l'image j ($j = 1, \cdots, J$). Au pixel n ($n = 1, \cdots, N_j$) de l'image j est associée l'observation y_{jn} de vraisemblance f paramétrée par θ_{jn} :

$$y_{jn}|\theta_{jn} \sim f(\cdot|\theta_{jn}) \tag{4.4}$$

Par la suite, les ensembles des observations et des paramètres sont respectivement notés $\boldsymbol{y}$ et $\boldsymbol{\theta}$.

Un travail précurseur [SJ08] a été effectué dans la segmentation d'un ensemble d'images naturelles. Le modèle bayésien nonparamétrique utilisé dans cet article est le Pitman-Yor, populaire pour sa modélisation des lois de puissance et les interactions spatiales sont représentées par des réalisations d'un processus gaussien.

La segmentation conjointe que nous proposons ici se fait à deux niveaux. D'abord, les pixels dont les observations sont statistiquement semblables sont groupés en *régions* de paramètres ψ_{jt}. Ainsi, dans l'image j, les paramètres des pixels attribués à la région t sont identiques $\{\theta_{jn} = \psi_{jt}|c_{jn} = t\}$ avec c_{jn} la variable indiquant la région allouée au pixel n de l'image j. On note $\boldsymbol{c} = \{c_{jn}; j = 1, \cdots, J; n = 1, \ldots, N_j\}$. De même, sont introduites des variables latentes

$\boldsymbol{d} = \{d_{jt}; j = 1, \cdots, J; t = 1, \cdots, m_{j\cdot}\}$ rattachées aux régions, où $m_{j\cdot}$ est le nombre de régions dans l'image j. Les régions statistiquement semblables sont ensuite groupées en *classes*. Par conséquent, les paramètres des régions assignées à la classe k partagent la même valeur ϕ_k ($k = 1, \cdots, K$), $\{\psi_{jt} = \phi_k | d_{jt} = k\}$. Par suite, les pixels dont les observations peuvent partager le même paramètre sont groupés en classes $\{\theta_{jn} = \phi_k | d_{jc_{jn}} = k\}$, $d_{jc_{jn}}$ indiquant la classe du pixel n de l'image j.

Dans un cadre bayésien, une loi *a priori* est proposée pour l'ensemble des étiquettes de région et de classe. Inspirée de la méthode précédente, cette loi s'écrit comme le produit de deux pénalités : une pénalité globale définie comme le *prior* induit par le processus de Dirichlet hiérarchique et une pénalité spatiale, le champ de Potts :

$$\begin{aligned}\Pr(\boldsymbol{c}, \boldsymbol{d}) \propto & \underbrace{\Pr{}^{\mathrm{c}}(\boldsymbol{c}, \boldsymbol{d})}_{\text{Pénalité globale}} \quad \underbrace{\Pr{}^{\mathrm{p}}(\boldsymbol{c}, \boldsymbol{d})}_{\text{Pénalité spatiale (locale)}} \\ & \propto \prod_{j=1}^{J} \left\{ \left[\prod_{n=1}^{N_j} \frac{1}{(\alpha_0 + n - 1)} \right] \alpha_0^{m_{j\cdot}} \left[\prod_{t=1}^{m_{j\cdot}} \Gamma(\nu_{jt}) \right] \right\} \left[\prod_{t=1}^{m_{\cdot\cdot}} \frac{1}{(\gamma + t - 1)} \right] \gamma^K \left[\prod_{k=1}^{K} \Gamma(m_{\cdot k}) \right] \\ & \times \prod_{j=1}^{J} \prod_{n=1}^{N_j} \exp\left(\sum_{q \in \mathcal{V}_n} \beta \, \mathbf{1}_{d_{jc_{jq}}, d_{jc_{jn}}} \right) \end{aligned} \tag{4.5}$$

avec ν_{jt} le nombre de pixels dans la région t de l'image j, $m_{\cdot k}$ le nombre de régions dans toutes les images de classe associée k et $m_{\cdot\cdot}$ le nombre total de régions. Le *prior* $\Pr^{\mathrm{c}}$ sur l'ensemble des étiquettes de région et de classe induit par le HDP est développé à la partie 2.2. En première intention, le champ de Potts est placé sur les étiquettes de classe associées directement aux pixels.

Ce modèle a fait l'objet d'une publication d'un article de conférence [SGC+16].

Ecrivons à présent la vraisemblance des données. Soit le processus de Dirichlet hiérarchique à mélange défini par l'équation (2.20). La loi de l'observation y_{jn} conditionnellement à son paramètre de classe affectée est donnée par $f(y_{jn}|\phi_k)$. Cependant, nous nous proposons pour notre étude de marginaliser les paramètres de classe. Il est donc nécessaire d'exprimer la distribution marginale des observations ne faisant pas intervenir les paramètres de classe, définie à la partie 2.2 par :

$$f(\boldsymbol{y}|\boldsymbol{c}, \boldsymbol{d}) = \prod_{k=1}^{K} f(\boldsymbol{y}_{A_k}) = \prod_{k=1}^{K} \left\{ \left[\int \prod_{(j,n) \in A_k} f(y_{jn}|\phi_k) \right] \mathrm{d}H(\phi_k) \right\}$$

L'estimation de la partition correspondant à l'ensemble d'observations $\boldsymbol{y}$ passe par la construction de la loi *a posteriori* des étiquettes sachant les observations : $\Pr(\boldsymbol{c}, \boldsymbol{d}|\boldsymbol{y}) \propto \Pr(\boldsymbol{c}, \boldsymbol{d}) f(\boldsymbol{y}|\boldsymbol{c}, \boldsymbol{d})$. Les modes de cette distribution ne peuvent pas être analytiquement calculés, un algorithme de Gibbs est alors proposé pour son exploration. Cette procédure consiste à échantillonner succes-

sivement les étiquettes de région assignées à chaque pixel de chaque image puis les étiquettes de classe associées aux régions. Ceci revient à effectuer à l'itération i :

- $\boldsymbol{c}^{(i)} \sim \Pr(\boldsymbol{c}|\boldsymbol{d}^{(i-1)}, \boldsymbol{y})$
- $\boldsymbol{d}^{(i)} \sim \Pr(\boldsymbol{d}|\boldsymbol{c}^{(i)}, \boldsymbol{y})$

Pour faciliter la compréhension des équations d'échantillonnage, nous présenterons d'abord les équations d'échantillonnage des étiquettes de classe puis les équations d'échantillonnage des étiquettes de région.

Échantillonnage des étiquettes de classe

La loi *a posteriori* conditionnelle d'une étiquette de classe sachant les autres variables s'écrit :

$$\Pr(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}, \boldsymbol{y}) \propto \Pr^{\mathrm{c}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \quad \Pr^{\mathrm{p}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \\ p(\boldsymbol{y}_{jt}|\boldsymbol{c}, d_{jt} = k, \boldsymbol{d}^{-jt}, \boldsymbol{y}^{-jt}) \tag{4.6}$$

avec $\boldsymbol{y}_{jt}$ l'ensemble des observations attachées aux pixels dans la région t de l'image j, $\boldsymbol{d}^{-jt}$ l'ensemble des étiquettes de classes sauf d_{jt} et A_k^{-jt} l'ensemble des pixels affectés à la classe k sauf ceux de la région t de l'image j.

Les différentes distributions conditionnelles intervenant dans l'équation (4.6) s'écrivent :

- distribution conditionnelle $\Pr^{\mathrm{c}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt})$:

$$\Pr^{\mathrm{c}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \propto \begin{cases} m_{\cdot k}^{-jt} & \text{si } k \leq K \\ \gamma & \text{si } k = k^{\mathrm{new}} = K + 1 \end{cases}$$

  où $m_{\cdot k}^{-jt}$ est le nombre total de régions affectées à la classe k sauf la région t de l'image j

- pour la distribution conditionnelle $\Pr^{\mathrm{p}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt})$, la région t est considérée supprimée de l'image, elle s'écrit donc :

$$\Pr^{\mathrm{p}}(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \propto \prod_{q \in \mathcal{V}_n | c_{jn} = t, c_{jq} \neq t} \exp\left(\beta \mathbf{1}_{d_{jc_{jq}}, k}\right)$$

- la distribution des observations $\boldsymbol{y}_{jt}$ conditionnellement aux autres paramètres est proportionnelle à la distribution de $\boldsymbol{y}_{jt}$ sachant les observations associées aux pixels dans la classe d'intérêt k sauf ceux de la région t, soit $\boldsymbol{y}_{A_k^{-jt}}$.

Le *prior* $\Pr^{\mathrm{c}}$ induit par le HDP assure donc qu'une région puisse être affectée à une classe existante ou à une nouvelle. La région t de l'image j est assignée à une classe k existante proportionnellement au nombre de régions affectées à cette classe hormis la région t et à la distribution $f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}})$ des observations attachées aux pixels de la région conditionnellement

aux observations attachées aux pixels dans la classe k en ôtant ceux de la région considérée ; le champ de Potts intervient via le nombre de voisins de la région alloués à cette classe. La probabilité que $d_{jt} = k^{\text{new}}$ conditionnellement au reste des variables est proportionnelle à γ et à la distribution marginale $f(\boldsymbol{y}_{jt})$ des observations attachées aux pixels dans la région. En remplaçant $\Pr^{\text{c}}$ et $\Pr^{\text{p}}$ par leurs expressions et en simplifiant :

$$\Pr(d_{jt} = k|\boldsymbol{c}, \boldsymbol{d}^{-jt}, \boldsymbol{y}) \propto \begin{cases} m_{\cdot k}^{-jt} \exp\left(\sum_{q \in \mathcal{V}^{[t]}} \beta \mathbf{1}_{d_{jc_{jq}}, k}\right) f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) & \text{si } k \leq K \\ \gamma f(\boldsymbol{y}_{jt}) & \text{si } k = k^{\text{new}} \end{cases} \tag{4.7}$$

où $q \in \mathcal{V}^{[t]}$ signifie ici l'ensemble des pixels q n'appartenant pas à la région t et voisins des pixels dans la région.

Échantillonnage des étiquettes de région

La loi *a posteriori* d'une étiquette de région s'écrit :

$$\begin{aligned} \Pr(c_{jn} = t|\boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}) \propto \Pr^{\text{p}}(d_{jc_{jn}}|c_{jn}, \boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}}) \quad \Pr^{\text{c}}(c_{jn} = t|\boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}}) \\ p(y_{jn}|c_{jn} = t, \boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}^{-jn}) \end{aligned} \tag{4.8}$$

avec $\boldsymbol{c}^{-jn}$ et $\boldsymbol{y}^{-jn}$ les ensembles d'étiquettes de région et d'observations en ôtant respectivement c_{jn} et y_{jn}.

Les distributions conditionnelles s'écrivent :

- la distribution conditionnelle $\Pr^{\text{p}}(d_{jc_{jn}}|c_{jn}, \boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}})$ est à prendre en compte car l'étiquette de région est implicitement indexée dans la distribution :

$$\Pr^{\text{p}}(d_{jc_{jn}}|c_{jn}, \boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}}) \propto \exp\left(\sum_{q \in \mathcal{V}_n} \beta \mathbf{1}_{d_{jc_{jq}}, d_{jc_{jn}}}\right)$$

- la distribution conditionnelle $\Pr^{\text{c}}(c_{jn} = t|\boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}})$, comme introduit au chapitre 2 s'écrit :

$$\Pr^{\text{c}}(c_{jn} = t|\boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jc_{jn}}) = \Pr^{\text{c}}(c_{jn} = t|\boldsymbol{c}^{-jn}) \propto \begin{cases} \nu_{jt}^{-jn} & \text{si } t \leq m_{j\cdot} \\ \alpha_0 & \text{si } t = t^{\text{new}} = m_{j\cdot} + 1 \end{cases}$$

  où ν_{jt}^{-jn} le nombre de pixels dans la région t de l'image j sauf le pixel n.

Le processus de Dirichlet hiérarchique induit donc que le pixel n de l'image j peut être dans une région existante $t \leq m_{j\cdot}$ proportionnellement au nombre de pixels dans cette région en omettant le n-ième, ν_{jt}^{-jn} ou dans une nouvelle proportionnellement à α_0. Le champ de Potts implique que cette probabilité d'être assigné à une région existante dépend aussi de la classe

associée à cette région et aux pixels voisins du pixel n. La probabilité pour le pixel d'être dans une nouvelle région est proportionnelle à α_0 et à la probabilité conditionnelle du pixel sachant qu'il appartient à une nouvelle région. Pour l'obtenir, il est nécessaire de sommer sur toutes les possibilités d'assignation de classe pour cette nouvelle région :

$$p(y_{jn}|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}^{-jn}) \propto \left\{\sum_{k=1}^{K} m_{\cdot k}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq},k}}\right)+\gamma\right\}^{-1} \left\{\sum_{k=1}^{K} m_{\cdot k}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq},k}}\right) f\left(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}\right)+\gamma f(y_{jn})\right\} \tag{4.9}$$

avec $f(y_{jn}) = \int f(y_{jn}|\vartheta^{\star}_{k^{\text{new}}})\mathrm{d}\Upsilon(\vartheta^{\star}_{k^{\text{new}}})$.

Les équations conditionnelles d'échantillonnage des étiquettes de région s'écrivent finalement :

$$\Pr(c_{jn}=t|\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}) \propto \begin{cases} \nu_{jt}^{-jn}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq}},d_{jt}}\right) f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{si } t\leq m_{j\cdot} \\ \alpha_0\, p(y_{jn}|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}^{-jn}) & \text{si } t=t^{\text{new}} \end{cases} \tag{4.10}$$

Dans le cas où une nouvelle région est choisie pour le pixel n de l'image j, la classe attribuée doit aussi être échantillonnée sur le même principe que (4.7) :

$$\Pr(d_{jt^{\text{new}}}=k|\boldsymbol{c},\boldsymbol{d}^{-jt^{\text{new}}}) \propto \begin{cases} m_{\cdot k}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq},k}}\right) f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{if } k\leq K \\ \gamma\, f(y_{jn}) & \text{if } k=k^{\text{new}} \end{cases} \tag{4.11}$$

Pour notre problématique, seules les étiquettes nous intéressent, néanmoins, notons que pour une partition donnée, il est possible d'échantillonner les paramètres de classe ϕ_k selon l'équation (2.38).

4.3 Algorithme de Swendsen-Wang

Une limitation d'un algorithme de Gibbs est que la convergence de la chaîne de Markov échantillonnée peut être lente en grande dimension. Nous proposons donc de recourir à l'algorithme de Swendsen-Wang pour pallier ce problème.

L'algorithme de Swendsen-Wang a été présenté au chapitre 3 pour une meilleure exploration de la distribution *a posteriori* des étiquettes lorsque leur loi *a priori* est le champ de Potts. Il est également utilisé dans [XCD] avec le modèle DP-Potts pour la segmentation d'une image. Nous nous proposons, sur le même principe, de modifier la routine d'exploration de notre algorithme,

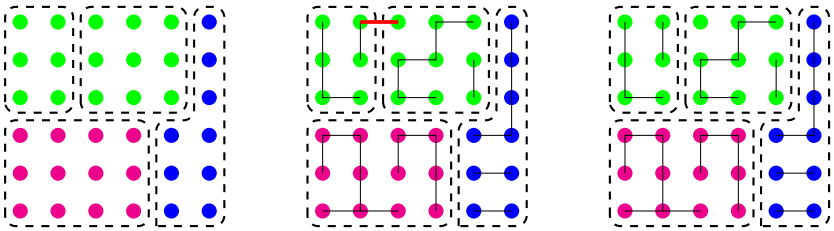


FIGURE 4.1 – La première figure est une image divisée en régions (pointillés) et la couleur des pixels représente leur classe. La seconde figure montre des liens possibles entre les pixels basés sur la classe, on observe que deux pixels de régions différentes peuvent être liés (rouge). La troisième figure est un exemple de type de liens que nous souhaitons obtenir.

le HDP-Potts, pour le cas de plusieurs images donc, avec l'algorithme de Swendsen-Wang. Cette approche a fait l'objet d'une publication dans une conférence [SGDG17].

Notons $\boldsymbol{r}$ le jeu d'étiquettes de liens avec $r_{j_{nq}}$ la valeur du lien entre les pixels n et q de l'image j, $r_{j_{nq}} = 1$ si les pixels sont liés et $r_{j_{nq}} = 0$ dans le cas contraire. Considérons de plus la probabilité $q_{j_{nq}} = \Pr(r_{j_{nq}} = 0|\boldsymbol{c}, \boldsymbol{d})$, à définir, que les pixels n et q de l'image j ne soient pas liés. Ecrivons la probabilité $q_{j_{nq}}$ sur la même base que le champ de Potts, c'est-à-dire $q_{j_{nq}} = \exp\left(-\lambda\beta \mathbf{1}_{d_{jc_{jn}}, d_{jc_{jq}}}\right)$. Cette probabilité pose néanmoins problème. En effet, puisque des régions différentes dans une même image peuvent être affectées à la même classe, des pixels de régions différentes pourraient être échantillonnés liés comme illustré dans la figure 4.1 ; l'échantillonnage des étiquettes de région liées serait alors faussé. Nous proposons pour empêcher cette configuration d'ajouter une contrainte d'appartenance à la région, soit $q_{j_{nq}} = \exp\left(-\lambda\beta \mathbf{1}_{d_{jc_{jn}}, d_{jc_{jq}}} \mathbf{1}_{c_{jn}, c_{jq}}\right)$. Étant donné que des pixels dans la même région sont forcément assignés à la même classe, $\mathbf{1}_{d_{jc_{jn}}, d_{jc_{jq}}} \mathbf{1}_{c_{jn}, c_{jq}} = 1$ si et seulement si $c_{jn} = c_{jq}$. Il en résulte que :

$$r_{j_{nq}}|c_{jn}, c_{jq} \sim \text{Ber}\left(1 - \exp\left(-\beta\lambda \mathbf{1}_{c_{jn}, c_{jq}}\right)\right) \tag{4.12}$$

avec $\text{Ber}(p)$ la loi de Bernoulli de paramètre p.

Remarque 4.1 (Distribution des variables de lien)
L'algorithme de Swendsen-Wang reste valide lorsque la distribution des variables de lien $\boldsymbol{r}$ est définie conditionnellement aux étiquettes de région $\mathbf{c}$ et de classe $\boldsymbol{d}$ et non juste sur les variables $\boldsymbol{d}$ définissant le champ de Potts. En effet, la seule contrainte à respecter est que la distribution des étiquettes de région et de classe reste inchangée lorsque les variables de lien sont marginalisées.

Equations d'échantillonnage

A chaque itération la procédure d'échantillonnage proposée est :

- $\boldsymbol{r}^{(i)} \sim \Pr(\boldsymbol{r}|\boldsymbol{c}^{(i-1)}, \boldsymbol{d}^{(i-1)}, \boldsymbol{y})$
- $\boldsymbol{c}^{(i)} \sim \Pr(\boldsymbol{c}|\boldsymbol{d}^{(i-1)}, \boldsymbol{r}^{(i)}, \boldsymbol{y})$
- $\boldsymbol{d}^{(i)} \sim \Pr(\boldsymbol{d}|\boldsymbol{c}^{(i)}, \boldsymbol{r}^{(i)}, \boldsymbol{y})$

L'échantillonnage des liens ne dépend que de la partition, en particulier des étiquettes de région $\boldsymbol{c}$. Conditionnellement à ces étiquettes, l'échantillonnage des liens est donc indépendant des observations, c'est-à-dire $\Pr(\boldsymbol{r}|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{y}) = \Pr(\boldsymbol{r}|\boldsymbol{c})$.

Ecrivons maintenant les équations d'échantillonnage des étiquettes de région. Une spécificité de l'algorithme de Swendsen-Wang est que les étiquettes des pixels d'un même spin-cluster sont simultanément mises à jour. Pour différencier l'étiquette d'un spin-cluster de l'étiquette d'un pixel, nous la notons en gras par la suite. La loi conditionnelle s'écrit :

$$\Pr(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{r}, \boldsymbol{y}) \propto \Pr(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d}) \quad \Pr(\boldsymbol{r}_{jl}|\boldsymbol{r}^{-jl}, \boldsymbol{c}_{jl} = t, \boldsymbol{c}^{-jl}) \\ p(\boldsymbol{y}_{jl}|\boldsymbol{c}_{jl} = t, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl})$$

avec $\boldsymbol{c}_{jl}$, $\boldsymbol{r}_{jl}$ et $\boldsymbol{y}_{jl}$ respectivement l'ensemble des étiquettes de région, de liens et d'observations associées aux pixels dans le spin-cluster l de l'image j, $\boldsymbol{c}^{-jl}$, $\boldsymbol{r}^{-jl}$ et $\boldsymbol{y}^{-jl}$ l'ensemble des étiquettes de région, de liens et l'ensemble des observations en omettant celles attachées aux pixels dans C_{jl}, l'ensemble des pixels du spin-cluster l de l'image j.

Nous avons $\Pr(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d}) \propto \Pr^{\mathrm{c}}(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d})\Pr^{\mathrm{p}}(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d})$. Le nombre de pixels voisins du spin-cluster affectés à la classe d_{jt} de la région t intervient dans le calcul de la contribution du champ de Potts. L'ensemble des pixels dits voisins du spin-cluster C_{jl} est l'ensemble des pixels voisins des pixels de C_{jl} et qui n'appartiennent pas au spin-cluster. La probabilité conditionnelle induite par le processus de Dirichlet hiérarchique est donnée par le produit de la probabilité $\Pr^{\mathrm{c}}(c_{jn_i} = t|\boldsymbol{c}^{-jn_i}, \boldsymbol{d}), (j, n_i) \in C_{jl}$ que le premier pixel du spin-cluster soit dans cette région, puis le second, jusqu'au dernier. Dans cette notation, $i \in [1, |C_{jl}|]$. Le calcul du terme $\Pr^{\mathrm{c}}(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d})$ est illustré dans le tableau suivant en notant $m_{j\cdot}^{-jl}$ le nombre de régions restant dans l'image j lorsque les pixels du spin-cluster l sont ôtés :

	$t \leq m_{j\cdot}^{-jl}$	$t = t^{\mathrm{new}} = m_{j\cdot}^{-jl} + 1$
$\Pr^{\mathrm{c}}(c_{jn_1} = t\|\boldsymbol{c}^{-jn_1}, \boldsymbol{d})$	$\propto \nu_{jt}^{-jl}$	$\propto \alpha_0$
$\Pr^{\mathrm{c}}(c_{jn_2} = t\|\boldsymbol{c}^{-jn_2}, \boldsymbol{d})$	$\times(\nu_{jt}^{-jl} + 1)$	$\times 1$
$\Pr^{\mathrm{c}}(c_{jn_3} = t\|\boldsymbol{c}^{-jn_3}, \boldsymbol{d})$	$\times(\nu_{jt}^{-jl} + 2)$	$\times 2$
$\vdots$	$\vdots$	$\vdots$
$\Pr^{\mathrm{c}}(c_{jn_{\lvert C_{jl}\rvert}} = t\|\boldsymbol{c}^{-jn_{\lvert C_{jl}\rvert}}, \boldsymbol{d})$	$\times(\nu_{jt}^{-jl} + \lvert C_{jl}\rvert - 1)$	$\times(\lvert C_{jl}\rvert - 1)$
$\Pr^{\mathrm{c}}(\boldsymbol{c}_{jl} = t\|\boldsymbol{c}^{-jl}, \boldsymbol{d})$	$\propto \Gamma(\nu_{jt}^{-jl} + \lvert C_{jl}\rvert)/\Gamma(\nu_{jt}^{-jl})$	$\propto \alpha_0 \Gamma(\lvert C_{jl}\rvert)$

La probabilité que les pixels dans C_{jl} soient affectés à une classe existante s'écrit donc :

$$\Pr(\boldsymbol{c}_{jl} = t \leq m_{j\cdot}^{-jl} | \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{r}, \boldsymbol{y}) \propto \frac{\Gamma(\nu_{jt}^{-jl} + |C_{jl}|)}{\Gamma(\nu_{jt}^{-jl})} \exp\left(\sum_{q \in \mathcal{V}_{C_{jl}}} \beta \left[\mathbf{1}_{d_{jc_{jq}}, d_{jt}} - \lambda \mathbf{1}_{c_{jq}, t} \right] \right) f(\boldsymbol{y}_{jl} | \boldsymbol{y}_{A_{d_{jt}}^{-jl}}) \tag{4.13}$$

avec $\mathcal{V}_{C_{jl}}$ l'ensemble des pixels voisins des pixels dans le spin-cluster C_{jl}. La probabilité que les pixels du spin-cluster C_{jl} soient dans une nouvelle région est :

$$\Pr(\boldsymbol{c}_{jl} = t^{\text{new}} | \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{r}, \boldsymbol{y}) \propto \alpha_0 \ \Gamma(|C_{jl}|) \ p(\boldsymbol{y}_{jl} | \boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl}) \tag{4.14}$$

où

$$p(\boldsymbol{y}_{jl} | \boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl}) \propto \left\{ \sum_{k=1}^{K} m_{\cdot k} \exp\left(\sum_{q \in \mathcal{V}_{C_l}} \beta \mathbf{1}_{d_{jc_{jq}}, k} \right) + \gamma \right\}^{-1} \left\{ \sum_{k=1}^{K} m_{\cdot k} \exp\left(\sum_{q \in \mathcal{V}_{C_l}} \beta \mathbf{1}_{d_{jc_{jq}}, k} \right) f\left(\boldsymbol{y}_{jl} \,|\, \boldsymbol{y}_{A_k^{-jl}} \right) + \gamma f(\boldsymbol{y}_{jl}) \right\} \tag{4.15}$$

avec $f(\boldsymbol{y}_{jl}) = \int [\prod_{n \in C_{jl}} f(y_{jn} | \phi_{k^{\text{new}}})] h(\phi_{k^{\text{new}}}) \mathrm{d}\phi_{k^{\text{new}}}$.

L'échantillonnage des étiquettes de lien affecte la mise à jour des étiquettes de région mais pas des étiquettes de classe ; les équations d'échantillonnage associées restent donc inchangées.

Deux algorithmes d'échantillonnage ont été proposés pour l'échantillonnage des étiquettes de région et de classe. Les hyperparamètres ont une place importante dans les équations développées, ils pilotent entre autres le nombre de classes générées. Il est alors primordial de bien les fixer. Il est donc nécessaire pour une segmentation optimale, de trouver le meilleur jeu d'hyperparamètres. Nous proposons un algorithme séquentiel Monte Carlo dans ce but.

4.3.1 Une alternative pour le champ de Potts

Un modèle différent peut être considéré permettant naturellement d'éviter la création de liens entre des pixels rattachés à des régions différentes. Il consiste à favoriser la cohérence spatiale non pas dans la définition des classes mais dans celle des régions. A cet effet, le champ de Potts est défini directement sur les étiquettes de région. Les équations de mise à jour des étiquettes de région ne font ainsi plus intervenir les étiquettes de classe. Nous présentons ci-après ce modèle succinctement en insistant sur les équations qui se trouvent modifiées par rapport à l'approche précédente. La pertinence de ces démarches respectives sera discutée dans le chapitre consacré aux résultats de simulation.

La loi *a priori* s'écrit toujours comme une pénalité globale associée à une pénalité spatiale où seule cette dernière est modifiée :

$$\Pr(\boldsymbol{c},\boldsymbol{d}) \propto \prod_{j=1}^{J}\left\{\left[\prod_{n=1}^{N_j}\frac{1}{(\alpha_0+n-1)}\right]\alpha_0^{m_{j\cdot}}\left[\prod_{t=1}^{m_{j\cdot}}\Gamma(\nu_{jt})\right]\right\}\left[\prod_{t=1}^{m_{\cdot\cdot}}\frac{1}{(\gamma+t-1)}\right]\gamma^{K}\left[\prod_{k=1}^{K}\Gamma(m_{\cdot k})\right] \times \prod_{j=1}^{J}\prod_{n=1}^{N_j}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\,\mathbf{1}_{c_{jq},c_{jn}}\right) \tag{4.16}$$

Algorithme de Gibbs

Le terme apporté par le champ de Potts dans l'équation d'échantillonnage des étiquettes de région ne dépend que des étiquettes de région. L'équation d'échantillonnage devient ainsi :

$$\Pr(c_{jn}=t|\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}) \propto \begin{cases}\nu_{jt}^{-jn}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{c_{jq},t}\right)f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{si } t\leq m_{j\cdot}\\ \alpha_0\,p(y_{jn}|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}^{-jn}) & \text{si } t=t^{\text{new}}\end{cases} \tag{4.17}$$

De plus, dans le cas où une nouvelle région est choisie pour le pixel n de l'image j, la classe attribuée doit aussi être échantillonnée sur le même principe que (4.7) :

$$\Pr(d_{jt^{\text{new}}}=k|\boldsymbol{c},\boldsymbol{d}^{-jt^{\text{new}}}) \propto \begin{cases} m_{\cdot k}f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{if } k\leq K\\ \gamma\,f(y_{jn}) & \text{if } k=k^{\text{new}}\end{cases} \tag{4.18}$$

Le champ de Potts n'intervient plus dans les équations d'échantillonnage des étiquettes de classe et elles s'écrivent :

$$\Pr(d_{jt}=k|\boldsymbol{c},\boldsymbol{d}^{-jt},\boldsymbol{y}) \propto \begin{cases} m_{\cdot k}^{-jt}f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) & \text{si } k\leq K\\ \gamma f(\boldsymbol{y}_{jt}) & \text{si } k=k^{\text{new}}\end{cases} \tag{4.19}$$

Algorithme de Swendsen-Wang

Les équations d'échantillonnage des étiquettes de région lorsque l'algorithme de Swendsen-Wang est appliqué s'écrivent dans ce cas :

$$\Pr(\boldsymbol{c}_{jl}=t\leq m_{j\cdot}^{-jl}|\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r},\boldsymbol{y}) \propto \frac{\Gamma(\nu_{jt}^{-jl}+|C_{jl}|)}{\Gamma(\nu_{jt}^{-jl})}\exp\left(\sum_{q\in\mathcal{V}_{C_{jl}}}\beta\,(1-\lambda)\,\mathbf{1}_{c_{jq},t}\right)f(\boldsymbol{y}_{jl}|\boldsymbol{y}_{A_{d_{jt}}^{-jl}}) \tag{4.20}$$

et

$$\Pr(\boldsymbol{c}_{jl} = t^{\text{new}} | \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{r}, \boldsymbol{y}) \propto \alpha_0 \ \Gamma(|C_{jl}|) \ p(\boldsymbol{y}_{jl} | \boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl})$$

avec

$$p(\boldsymbol{y}_{jl} | \boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl}) \propto \left\{ \sum_{k=1}^{K} m_{\cdot k} + \gamma \right\}^{-1} \left\{ \sum_{k=1}^{K} m_{\cdot k} f\left(\boldsymbol{y}_{jl} \,|\, \boldsymbol{y}_{A_k^{-jl}} \right) + \gamma f(\boldsymbol{y}_{jl}) \right\} \tag{4.21}$$

4.4 Estimation des hyperparamètres

Dans cette partie est présentée une méthode d'estimation des hyperparamètres du modèle HDP-Potts qui peut aussi être utilisée dans le cas du DP-Potts. Nous présentons en particulier le cas de l'estimation des paramètres scalaires du HDP, α_0 et γ. Le paramètre β du champ de Potts peut être estimé suivant la méthode présentée dans [PDBT13] via un algorithme *approximate Bayesian computation* qui repose sur l'introduction de variables latentes et d'une statistique à définir. Cette méthode a donné de bons résultats, néanmoins, elle fait intervenir une procédure d'acceptation rejet dépendant de la statistique à déterminer. Ainsi, il est nécessaire de choisir cette dernière de manière appropriée pour assurer des propositions pertinentes et un taux d'acceptation convenable. Notons qu'il n'y a pas de procédure d'acceptation rejet dans le SMC.

Notons $\chi = \{\alpha_0, \gamma\}$ l'ensemble des hyperparamètres. Dans un cadre bayésien, le meilleur χ peut être choisi comme maximisant $p(\chi|\boldsymbol{y})$ la loi *a posteriori* des hyperparamètres sachant les observations. Une loi *a priori* conjuguée pour ce couple de paramètres a été proposée dans [TJBB06]. Néanmoins, la constante de normalisation du champ de Potts qui est une somme sur toutes les configurations possibles du champ d'étiquettes pour un nombre de classes donné intervient dans notre modèle et rend impossible le recours à une loi *a priori* conjuguée. En effet, la fonction de partition est rarement analytiquement calculable. Un algorithme de Gibbs ne peut donc pas être développé pour l'inférence des hyperparamètres.

En outre, l'estimation de constantes de normalisation est une problématique d'actualité. Elle a été souvent abordée dans le cas de mélange de lois pour le choix du modèle correspondant [Bis06]. Elle intervient en particulier dans le calcul de la loi marginale et les calculs de ratios de constantes de normalisation pour la comparaison de modèles. Différentes méthodes existent, certaines permettent de contourner le calcul des constantes, comme l'introduction de variables auxiliaires [MPRB06] ou l'approximation directe de la loi *a posteriori* comme l'*approximate bayesian computation* [PSPLF99]. Il est néanmoins difficile d'appliquer la plupart de ces approches dans notre cas car une infinité de jeu d'étiquettes doit être considérée.

Un algorithme de Metropolis-Hastings within Gibbs peut également être envisagé. Il consiste à effectuer un pas de Metropolis-Hastings pour la proposition de valeurs pour les hyperparamètres et un algorithme de Gibbs pour l'échantillonnage des étiquettes. Soit $q(\cdot|\chi)$ (ou $q(\cdot|\chi, \boldsymbol{y})$) la loi de proposition associée. L'algorithme s'écrirait :

- Initialisation $\chi^{(0)}$
- Échantillonner une partition $\boldsymbol{c}^{(0)}, \boldsymbol{d}^{(0)} \sim \Pr(\boldsymbol{c}, \boldsymbol{d}|\boldsymbol{y}, \chi^{(0)})$
- A l'itération $i+1$,
  - Générer une valeur de proposition χ^*
  - Soit
$$\chi^{(i+1)} = \begin{cases} \chi^* & \text{avec la probabilité } \varrho(\chi^{(i)}, \chi^*) \\ \chi^{(i)} & \text{avec la probabilité } 1 - \varrho(\chi^{(i)}, \chi^*) \end{cases}$$
  - Échantillonner la partition $\boldsymbol{c}^{(i+1)}, \boldsymbol{d}^{(i+1)} \sim \Pr(\boldsymbol{c}, \boldsymbol{d}|\boldsymbol{y}, \chi^{(i+1)})$

où $\varrho(\chi^{(i)}, \chi^*) = \dfrac{p(\chi^*|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{y})\ q(\chi|\chi^*)}{p(\chi|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{y})\ q(\chi^*|\chi)}$.

Ici, $p(\chi|\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{c}, \boldsymbol{d}, \chi)\Pr(\boldsymbol{c}, \boldsymbol{d}|\chi)p(\chi)$ avec $p(\chi)$ la loi *a priori* choisie pour χ. De plus, la constante de normalisation Z pour $\Pr(\boldsymbol{c}, \boldsymbol{d}|\chi)$ est la somme sur toutes les partitions de (4.5). Deux problèmes se posent, premièrement, cette constante ne peut être calculée analytiquement dans la plupart des cas, et, deuxièmement, cette constante est différente pour chaque valeur de χ. Il suit que la probabilité d'acceptation ne peut être calculée simplement et qu'un algorithme de Metropolis-Hastings within Gibbs ne peut être implémenté.

Étudions à présent l'application d'un algorithme de Monte Carlo séquentiel introduit au chapitre 1 pour estimer χ au sens du maximum de vraisemblance. Pour ce faire, une grille décroissante de valeurs d'hyperparamètres $\chi_m, m = 1, \ldots, M$ est parcourue. En effet, des valeurs élevées d'hyperparamètres impliquent une meilleure exploration de la loi *a posteriori* mais favorisent la création de nouvelles classes, donc une sur-segmentation en opposition aux petites valeurs qui impliquent une probabilité faible de proposition de nouvelle classe comme présenté au chapitre 2. Cependant, l'échantillonnage doit être fait de manière adéquate pour assurer la convergence de la chaîne de Markov et sa convergence vers la distribution désirée. Si χ est changé à chaque itération du Gibbs alors la chaîne de Markov générée n'est pas homogène et la convergence n'est pas garantie. L'algorithme SMC [DDJ06] décrit au chapitre 1, consiste à simuler plusieurs chaînes de Markov en parallèle pour corriger l'écart entre la loi simulée et la loi cible par échantillonnage d'importance. Le principe est d'échantillonner pour chaque valeur χ_m, I particules en parallèle représentant des partitions candidates puis d'écrire la loi cible comme une combinaison de mesures de Dirac centrées sur ces particules :

$$\Pr(\boldsymbol{c}, \boldsymbol{d}|\boldsymbol{y}, \chi_m) \simeq \sum_{i=1}^{I} W_m^{(i)} \mathbf{1}_{\boldsymbol{c}_m^{(i)}, \boldsymbol{c}} \mathbf{1}_{\boldsymbol{d}_m^{(i)}, \boldsymbol{d}}$$

où $\boldsymbol{c}_m^{(i)}$ et $\boldsymbol{d}_m^{(i)}$ sont respectivement les ensembles d'étiquettes de région et de classe échantillonnés à l'itération i de l'étape m. Le poids pour la particule i à l'étape m s'écrit :

$$W_m^{(i)} = W_{m-1}^{(i)} \tilde{w}_m^{(i)} / \sum_{\ell=1}^{I} W_{m-1}^{(\ell)} \tilde{w}_m^{(\ell)} \tag{4.22}$$

$$\text{avec} \quad \tilde{w}_m^{(i)} = \frac{\Pr(\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)} | \chi_m)\, p(\boldsymbol{y} | \boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)}, \chi_m)\, \rho_m(\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)}, \boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)})}{\Pr(\boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)} | \chi_{m-1})\, p(\boldsymbol{y} | \boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)}, \chi_{m-1})\, q_m(\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)}, \boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)})}$$

où $\tilde{w}_m^{(i)}$ sont les poids incrémentaux non-normalisés, ρ_m est le noyau rétrograde et q_m est la loi de proposition.

L'intérêt de l'algorithme de Monte Carlo séquentiel est que par construction, il permet aussi de calculer une estimation des rapports de vraisemblance des données par rapport aux valeurs d'hyperparamètres. Néanmoins, dans notre cas, la constante de normalisation du modèle bayésien non paramétrique *a priori* évolue également en fonction des hyperparamètres, le ratio estimé est alors :

$$R(m) = \frac{p(\boldsymbol{y}|\chi_m) Z_m}{p(\boldsymbol{y}|\chi_{m-1}) Z_{m-1}} \simeq \sum_{i=1}^{I} W_{m-1}^{(i)} \tilde{w}_m^{(i)}. \tag{4.23}$$

avec Z_m la constante de normalisation de la loi *a priori* non normalisée.

Ainsi, autant pour le Metropolis-Hastings within Gibbs que pour l'échantillonneur de Monte Carlo séquentiel, apparaît le rapport de constantes de normalisation des lois bayésiennes non paramétriques *a priori*.

Nous proposons alors de contourner cette difficulté en implémentant deux algorithmes de SMC avec la même décroissance sur le vecteur de paramètres. Le premier, basé sur l'échantillonnage suivant les lois *a posteriori* donnera une approximation des rapports de vraisemblance non normalisés (4.23) et le second, basé sur l'échantillonnage uniquement suivant les lois *a priori* donnera une approximation des rapports de constantes de normalisation $R_p(m) = \dfrac{Z_m}{Z_{m-1}}$.

Ainsi, R peut être divisé par R_p pour compenser les constantes de normalisation des lois *a priori* et obtenir à chaque itération le rapport des vraisemblances $p(\boldsymbol{y}|\chi_m)/p(\boldsymbol{y}|\chi_{m-1})$. Le produit cumulé des ratios estimés R permet alors d'approcher $p(\boldsymbol{y}|\chi_m)/p(\boldsymbol{y}|\chi_0)$ à chaque itération. Finalement, le vecteur d'hyperparamètres conduisant à la valeur maximale de ce rapport est l'estimé de χ_m au sens du maximum de vraisemblance.

Algorithme 4.2 Échantillonneur de Monte Carlo séquentiel pour l'estimation des hyperparamètres

• SMC suivant la *a posteriori*

Initialisation
pour $i = 1, \ldots, I$ **faire**
  Initialisation $\boldsymbol{c}^{(i)}, \boldsymbol{d}^{(i)}$
fin pour
pour $m = 1, \ldots, M$ **faire**
  pour $i = 1, \ldots, I$ **faire**
    $\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)} \sim \Pr(\boldsymbol{c}, \boldsymbol{d} | \boldsymbol{y}, \chi_m, \boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)})$
    Calcul des poids non normalisés $\tilde{w}_m^{(i)}$
  fin pour
  Calcul des poids normalisés $W_m^{(i)}$
  Calcul du rapport $R(m)$ (avec $R(1) = 1$)
  si $\mathrm{ESS} < \varepsilon$ **alors**
    Rééchantillonnage des particules
  fin si
fin pour

• SMC suivant la loi *a priori*

Initialisation
pour $i = 1, \ldots, I$ **faire**
  Initialisation $\boldsymbol{c}^{(i)}, \boldsymbol{d}^{(i)}$
fin pour
pour $m = 1, \ldots, M$ **faire**
  pour $i = 1, \ldots, I$ **faire**
    $\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)} \sim \Pr(\boldsymbol{c}, \boldsymbol{d} | \chi_m, \boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)})$
    Calcul des poids non normalisés $\tilde{w}_m^{(i)}$
  fin pour
  Calcul des poids normalisés $W_m^{(i)}$
  Calcul du rapport $R_p(m)$ (avec $R_p(1) = 1$)
  si $\mathrm{ESS} < \varepsilon$ **alors**
    Rééchantillonnage des particules
  fin si
fin pour

• Calcul de la vraisemblance marginale

$p(\boldsymbol{y}|\chi_m) \simeq \prod_{\ell=1}^{m} R(\ell)/R_p(\ell)$

4.5 Conclusion

Ce chapitre présente des modèles bayésiens non paramétriques pour la segmentation d'images.

Premièrement, un modèle combinant le processus de Dirichlet et le champ de Potts est décrit pour la segmentation d'une image. Dans un second temps, le modèle que nous proposons est décrit. Il combine le processus de Dirichlet hiérarchique et le champ de Potts pour la segmentation jointe d'un ensemble d'images. Il permet la segmentation indépendante de chaque image en régions et le regroupement des régions identiques entre classes en classes pouvant ainsi être partagées entre les images ou non.

Un algorithme de Gibbs a par la suite été développé pour l'échantillonnage des étiquettes selon la loi *a posteriori* découlant. Concernant le champ de Potts, l'algorithme de Swendsen-Wang généralisé est réputé permettre une meilleure exploration de la loi, nous proposons alors un algorithme basé sur cette approche pour notre modèle. Enfin, les valeurs choisies pour les hyperparamètres du modèle sont importantes pour une bonne exploration. Nous proposons alors un algorithme de Monte Carlo séquentiel permettant de choisir la meilleure valeur d'hyperparamètres au sens du maximum de vraisemblance.

CHAPITRE 5

Présentation et analyse des résultats

Les algorithmes DP-Potts et HDP-Potts ont été testés sur deux types de données : des images simulées selon un champ de Potts, qui vont nous permettre de les valider avec une vérité terrain ainsi que des images de la base d'images LabelMe [1]. De plus, pour illustrer une application avec le découpage d'une « grande » image en imagettes, les algorithmes ont aussi été testés sur une image de montagnes d'Afrique de l'est [2]. Par souci de lisibilité, seule une partie des résultats est présentée dans ce chapitre, des résultats complémentaires sont donnés à l'annexe E.

Nous mettons en parallèle notre méthode et le DP-Potts car il s'agit du modèle bayésien non paramétrique le plus proche de celui que nous proposons, qui permet à la fois l'estimation de la partition et du nombre de classes. Puisqu'il ne s'applique qu'à une image, une concaténation des images à segmenter a été effectuée. La méthode a été appliquée en définissant un système de voisinage qui respecte les délimitations entre images. Ainsi, deux pixels dans des images différentes ne peuvent être voisins. Le DP, le HDP et le Swendsen-Wang appliqué au HDP-Potts ont aussi été programmés pour comparaison.

Nous présentons d'abord la méthode d'estimation de la meilleure partition et quelques métriques utilisées en classification qui nous permettront de comparer les performances des algorithmes dans le cas des images-test où la vraie partition est connue. Suivent les résultats obtenus pour les trois types de données.

1. labelme.csail.mit.edu/
2. http ://earth.imagico.de, The Rwenzori mountains and lake Edward

5.1 Estimation de la meilleure partition

Nous cherchons la meilleure partition au sens où elle minimise un certain critère d'erreur. Les différents critères sont décrits dans le cas d'une image pour faciliter leur compréhension ; ils sont cependant facilement généralisables à un ensemble d'images.

Comment choisir $\widehat{\boldsymbol{z}}$ correspondant à la partition la plus pertinente à partir de la loi *a posteriori* ? L'étiquetage d'une itération à une autre étant différent, il est nécessaire d'utiliser une métrique de comparaison entre la partition vraie et la partition estimée insensible à la numérotation des classes et ne prenant en compte que la partition.

La fonction de coût de Binder [Bin78] respecte ces propriétés et a été utilisée en classification [FS06], [CTM14]. Elle est liée à un risque bayésien et s'écrit :

$$\mathcal{C}(\boldsymbol{z}^\star) = \sum_{\boldsymbol{z}} L(\boldsymbol{z}, \boldsymbol{z}^\star)\Pr(\boldsymbol{z}|\boldsymbol{x}) \quad \text{avec} \quad L(\boldsymbol{z}, \boldsymbol{z}^\star) = \sum_{n=1}^{N}\sum_{q=1}^{N} a\mathbf{1}_{z_n=z_q, z_n^\star \neq z_q^\star} + b\mathbf{1}_{z_n\neq z_q, z_n^\star = z_q^\star} \quad (5.1)$$

avec C est le coût *a posteriori*, L la fonction de coût de Binder, a et b des nombres scalaires positifs, $\boldsymbol{z}$ le jeu d'étiquettes à comparer et $\boldsymbol{z}^\star$ un jeu d'étiquettes candidat pour l'estimation. La partition optimale dans ce cas est celle qui favorise une configuration souvent apparue. Le facteur a est l' « importance » liée au fait que deux pixels soient assignés à des groupes différents alors qu'ils devraient être dans le même. Le facteur b est la perte associée au fait d'assigner deux pixels au même groupe alors qu'ils appartiennent à des groupes différents. Le calcul de ce coût s'appuie sur la construction d'une matrice de similarité M_{sim} de dimension $N \times N$. Il s'agit d'une matrice binaire donnant l'appartenance aux classes par paire de pixels. Ainsi, si les pixels n et q sont dans la même classe, on a $M_{\text{sim}}(n, q) = M_{\text{sim}}(q, n) = 1$. Il s'agit alors de comparer les matrices de similarité associées à $\boldsymbol{z}$ et à $\boldsymbol{z}^\star$ respectivement.

Pour $a = b = 1$, L devient un coût associé à une distance quadratique et a été défini par Dahl dans [Dah06] :

$$L_D(\boldsymbol{z}, \boldsymbol{z}^\star) = \sum_{n=1}^{N}\sum_{q=1}^{N} \left[\mathbf{1}_{z_n=z_q} - \mathbf{1}_{z_n^\star=z_q^\star}\right]^2 \quad (5.2)$$

La meilleure partition $\widehat{\boldsymbol{z}}$, dans un cadre de décision bayésien, est celle qui minimise le coût a posteriori C défini à l'équation (5.1).

Néanmoins, minimiser L_D dans l'espace de toutes les partitions est combinatoirement impossible. Soient I échantillons obtenus au cours de la procédure d'exploration par MCMC notés $\boldsymbol{z}^{(i)}, i = 1, \cdots, I$. Il est alors proposé de choisir une partition $\widehat{\boldsymbol{z}}$ parmi celles déjà échantillonnées. Cette méthode a au moins deux avantages : le coût calculatoire et, les partitions obtenues sont par définition parmi les plus probables.

Dans ce cas, la section 3.1 de [FI09] présente les équations pour l'obtention de la meilleure partition qui s'écrivent :

$$\widehat{\boldsymbol{z}} = \underset{\boldsymbol{z}^\star \in \{\boldsymbol{z}^{(1)},\cdots,\boldsymbol{z}^{(I)}\}}{\operatorname{argmin}} \sum_{n=1}^{N}\sum_{q=1}^{N}\left(\mathbf{1}_{z_n^{\star(i)},z_q^{\star(i)}} - \varpi_{nq}\right)^2, \quad \varpi_{nq} = \frac{1}{I}\sum_{i=1}^{I}\mathbf{1}_{z_n^{\star(i)},z_q^{\star(i)}} \tag{5.3}$$

La meilleure partition est ici celle qui est la plus proche d'une partition moyenne donnée par $\boldsymbol{\varpi}$. Nous utiliserons ce critère dans notre cas.

Après avoir estimé la meilleure partition, il convient de définir des critères de comparaisons à la vérité terrain des configurations obtenues pour chaque algorithme dans le cas des images-test où la vraie partition est connue. Cette question est difficile car le nombre de classes estimées diffère généralement du nombre de classes réelles et l'étiquetage n'est pas le même. Nous présentons quelques métriques utilisées en classification ainsi que la méthode de ré-étiquetage utilisée. On pourrait comparer les matrices de similarité, mais nous cherchons à mener une analyse par classe et utiliser des métriques classiques en classification et faciles à interpréter.

5.1.1 Quelques métriques

Le modèle proposé dans ce manuscrit pose un problème de non identifiabilité lié au *label-switching* qui se traduit par une variabilité de la numérotation des classes d'une itération à l'autre [CMR05, Chapitre 13]. Une partie des métriques de classification sont basées sur la comparaison des étiquettes. Leur calcul passe par celui de la *matrice de confusion*. Il s'agit d'une matrice carrée de dimension la plus grande valeur $K^\star$ entre K_{vrai} et K_{est} avec K_{vrai} le nombre réel de classes et K_{est} le nombre estimé. Elle est construite comme suit :

- chaque ligne représente le nombre d'occurrences d'une classe réelle
- chaque colonne représente le nombre d'occurrences d'une classe estimée
- $M_{\text{conf}}(k,l)$ donne le nombre de pixels appartenant à la classe k mais classés en l
- les lignes/colonnes supplémentaires sont complétées par des 0.

Pour une comparaison étiquette à étiquette, il est nécessaire de ré-étiqueter les observations pour que la numérotation du jeu d'étiquettes estimé concorde avec celle correspondant à la vraie partition. Pour ce faire nous proposons une méthode de ré-étiquetage basée sur le nombre de pixels réel par classe. Les nombres de pixels par classe dans la partition estimée sont comparés aux nombres vrais et par un jeu de permutations, une renumérotation cohérente est proposée.

Il est à noter que cette méthode peut « corriger » certaines partitions et améliorer ainsi les performances données par les métriques. Elle peut aussi les fausser. Prenons le cas où $K_{\text{est}} < K_{\text{vrai}}$. La renumérotation effectue une concordance entre les classes estimées et les K_{est} classes les plus représentées dans la partition de référence. Lorsque $K_{\text{est}} \geq K_{\text{vrai}}$, étant donné que la renumérotation ne se base que sur le nombre de pixels par classe, deux classes ayant des effectifs

voisins peuvent voir leur numérotation estimée inversée. Pour s'affranchir de ces difficultés, une partie des métriques proposées est insensible à l'étiquetage.

Critère de Dahl

Le critère de Dahl L_D (5.2) permet d'évaluer la proportion de couples de pixels originellement partageant la même classe qui sont également dans la même classe dans la partition estimée. Cette procédure permet de s'affranchir de la numérotation, de K_{vrai} et de K_{est}. En pratique, nous utiliserons $\%L_D$ correspondant au pourcentage de couples de pixels étant dans la même classe dans la vraie configuration et dans des classes différentes dans la partition estimée. $\%L_D$ est calculé pour chaque image individuellement et $\%L_{D_g}$ donne le même coût pour l'ensemble des images jointes.

L'*overall accuracy*

La matrice de confusion permet d'évaluer le pourcentage de pixels bien classés (*overall accuracy* OA) :

$$\text{OA} = \frac{\sum_{k=1}^{K^\star} M_{\text{conf}}(k,k)}{N} \tag{5.4}$$

L'OA est une métrique intuitive et relativement facile à interpréter.

Le coefficient kappa de Cohen

Le coefficient *Kappa de Cohen* κ_{co}, permet de quantifier l'accord entre deux techniques pour des jugements catégoriels [Coh60] :

$$\kappa_{\text{co}} = \frac{P_0 - P_e}{1 - P_e} \tag{5.5}$$

avec P_0 la proportion de l'échantillon sur laquelle les deux techniques s'accordent (qui est donnée par OA) et P_e la probabilité d'un accord aléatoire entre les deux,

$$P_e = \frac{\sum_{k=1}^{K^\star} M_{\text{conf}}(k,\cdot) M_{\text{conf}}(\cdot,k)}{N^2} \tag{5.6}$$

avec $M_{\text{conf}}(k,\cdot) = \sum_{l=1}^{K^\star} M_{\text{conf}}(k,\ell)$ et $M_{\text{conf}}(\cdot,k) = \sum_{\ell=1}^{K^\star} M_{\text{conf}}(\ell,k)$.

κ_{co} prend ses valeurs entre -1 et 1. Plus il se rapproche de 1 et meilleur est l'accord entre les juges, c'est-à-dire dans notre cas que la partition estimée est proche de la vraie partition. Son interprétation est à moduler avec le nombre de classes comparées ; il a tendance à être faible pour K élevé. En effet, il est par exemple plus facile de choisir entre deux cas qu'entre dix.

La V-mesure

La V-mesure [RH07] ne nécessite pas un ré-étiquetage de la partition estimée car elle est indépendante de la numérotation, de K_{vrai} et de K_{est}. La matrice de confusion mentionnée dans cette partie peut alors être de dimension $K_{\text{vrai}} \times K_{\text{est}}$, d'où la spécification des bornes de sommation dans les équations données. Redéfinissons aussi les notations : $M_{\text{conf}}(\cdot, \ell) = \sum_{k=1}^{K_{\text{vrai}}} M_{\text{conf}}(k, \ell)$ et $M_{\text{conf}}(k, \cdot) = \sum_{\ell=1}^{K_{\text{est}}} M_{\text{conf}}(k, \ell)$.

La V-mesure est calculée à partir de :

- l'« homogénéité » : elle est maximale lorsque seuls les pixels assignés à une unique classe dans la vraie partition sont assignés à une unique classe dans la partition estimée

$$h_{\text{o}} = \begin{cases} 1 & \text{si } \Psi(\mathcal{K}) = 0 \\ 1 - \dfrac{\Psi(\mathcal{K}|\mathcal{K}^\star)}{\Psi(\mathcal{K})} & \text{sinon} \end{cases}$$

avec $\mathcal{K} = \{1, \ldots, K_{\text{vrai}}\}$, $\mathcal{K}^\star = \{1, \ldots, K_{\text{est}}\}$ et

$$\Psi(\mathcal{K}|\mathcal{K}^\star) = -\sum_{\ell=1}^{K_{\text{est}}} \sum_{k=1}^{K_{\text{vrai}}} \frac{M_{\text{conf}}(k, \ell)}{N} \log \left[\frac{M_{\text{conf}}(k, \ell)}{M_{\text{conf}}(\cdot, \ell)} \right]$$

$$\Psi(\mathcal{K}) = -\sum_{k=1}^{K_{\text{vrai}}} \frac{M_{\text{conf}}(k, \cdot)}{N} \log \left[\frac{M_{\text{conf}}(k, \cdot)}{N} \right]$$

- la « complétude » : elle est maximale lorsque tous les pixels assignés à une unique classe dans la vraie partition sont assignés à une unique classe dans la partition estimée

$$c_{\text{o}} = \begin{cases} 1 & \text{si } \Psi(\mathcal{K}^\star) = 0 \\ 1 - \dfrac{\Psi(\mathcal{K}^\star|\mathcal{K})}{\Psi(\mathcal{K}^\star)} & \text{sinon} \end{cases}$$

avec

$$\Psi(\mathcal{K}^\star|\mathcal{K}) = -\sum_{k=1}^{K_{\text{vrai}}} \sum_{\ell=1}^{K_{\text{est}}} \frac{M_{\text{conf}}(k, \ell)}{N} \log \left[\frac{M_{\text{conf}}(k, \ell)}{M_{\text{conf}}(k, \cdot)} \right]$$

$$\Psi(\mathcal{K}^\star) = -\sum_{\ell=1}^{K_{\text{est}}} \frac{M_{\text{conf}}(\cdot, \ell)}{N} \log \left[\frac{M_{\text{conf}}(\cdot, \ell)}{N} \right]$$

Un exemple illustratif de ces propriétés est donné à la figure 5.1. La comparaison des classes est ici décrite en indiquant les noms de classe, mais les propriétés estimées sont insensibles à la numérotation des classes. La classe « vert » a une forte homogénéité car les pixels affectés à cette classe dans la partition estimée sont affectés à une classe unique dans la partition référence, la complétude est néanmoins faible car tous les pixels de la classe dans la partition référence ne sont pas dans une unique classe dans l'image de droite ; en effet, deux pixels ont été assignés

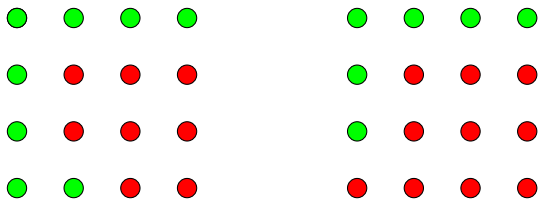


FIGURE 5.1 – Exemple visuel de la complétude et de l'homogénéité prises individuellement par classe. A gauche est représentée une partition référence et à droite une partition estimée ; les couleurs représentent les classes.

à la classe « rouge ». Pour la classe « rouge », on a l'inverse, l'homogénéité est faible car les pixels de cette classe dans la partition estimée comprennent aussi des pixels originellement dans la classe « verte » et la complétude est maximale parce que tous les pixels de la classe dans la partition référence y sont dans la partition estimée.

Finalement, l'expression de la V-mesure est :

$$V_b = \frac{(1+b) \times h_{\mathrm{o}} \times c_{\mathrm{o}}}{(b \times h_{\mathrm{o}}) + c_{\mathrm{o}}}$$

La valeur prise par V_b est comprise entre 0 et 1 ; 1 étant la meilleure. Pour $b > 1$, la complétude est plus fortement pondérée que l'homogénéité et l'inverse pour $b < 1$.

5.1.2 Critère de convergence

Dans ce travail, l'échantillonnage des partitions se fait via des algorithmes de Gibbs qui sont des méthodes MCMC, et, pour ce type d'approches, se pose la question de la convergence des chaînes générées. Nous nous intéressons alors à l'application de tests de convergence dans notre cas.

La plupart des critères de convergence ne sont définis que pour des variables continues ; ils ne sont donc pas adaptés dans le cas de partitions. Une solution souvent utilisée est le calcul de l'autocorrélation entre les réalisations pour déterminer leur degré de corrélation. Un autre problème émerge alors : le *label-switching*, il n'est donc pas non plus possible d'utiliser ce critère.

Nous nous baserons donc juste sur l'allure du logarithme de la probabilité *a posteriori* des étiquettes au cours des itérations. Cette approche n'est néanmoins pas satisfaisante puisque la chaîne peut être bloquée dans un minimum local ou « mal mélanger ».

5.2 Résultats : cas des images-test

Génération des données

Des images-test de dimension 64×64 ont été simulées *a priori* selon un champ de Potts de paramètre $\beta = 1.2$ avec un nombre de classes $K_{\text{vrai}} = 3$. A chaque classe est aléatoirement affecté un niveau de gris : -25, 0 ou 25. Les images-test sont donc constantes par morceau. Pour construire un modèle d'observation, un bruit gaussien a ensuite été ajouté.

Le modèle d'observation correspondant est alors défini comme :

$$\begin{aligned} \phi_k | H &\sim H \equiv \mathcal{N}(\mu_0 = 0, \sigma_0^2 = 75^2) \\ y_{jn} | \phi_k &\sim f(\cdot|\phi_k) \equiv \mathcal{N}(y_{jn}; \phi_k, \sigma_y^2) \end{aligned}$$

Les densités h, associée la distribution de base du processus de Dirichlet hiérarchique définie au paragraphe 2.2, et f sont conjuguées, il est alors possible d'écrire analytiquement la vraisemblance marginale des données associées aux pixels dans la classe k. Etant donné que le modèle de vraisemblance a été construit par ajout d'un bruit blanc gaussien, il peut être réécrit comme :

$$y_{jn} = \phi_k + \varepsilon_{jn}, \quad \varepsilon_{jn} \sim \mathcal{N}(0, \sigma_y^2) \tag{5.7}$$

L'ensemble des observations attachées aux pixels dans la classe k vérifie alors :

$$\boldsymbol{y}_{A_k} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \phi_k + \begin{bmatrix} \varepsilon_{\ell_1} \\ \vdots \\ \varepsilon_{\ell_{|A_k|}} \end{bmatrix}$$

avec $\ell_1, \cdots, \ell_{|A_k|}$ les indices des pixels dans la classe k ; ℓ comprenant le numéro d'image et de pixel dans le cas du HDP. Ce vecteur suit une loi normale de paramètres :

$$\boldsymbol{\mu}_k = \mathbb{E}[\boldsymbol{y}_{A_k}] = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \mu_0$$

$$\boldsymbol{\Sigma}_k = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \sigma_0^2 + I_{|A_k|} \times \sigma_y^2 = \begin{bmatrix} \sigma_0^2 + \sigma_y^2 & \sigma_0^2 & \cdots & \sigma_0^2 \\ \sigma_0^2 & \sigma_0^2 + \sigma_y^2 & \cdots & \sigma_0^2 \\ & & \ddots & \\ \sigma_0^2 & \sigma_0^2 & \cdots & \sigma_0^2 + \sigma_y^2 \end{bmatrix}$$

avec $\boldsymbol{\mu}$ la moyenne, $\boldsymbol{\Sigma}$ la matrice de covariance et la loi marginale $f(\boldsymbol{y}_{A_k}) \equiv \mathcal{N}(\boldsymbol{y}_{A_k}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. La matrice de covariance $\boldsymbol{\Sigma}$ est circulante, ce qui permet un calcul numérique du logarithme de

la vraisemblance par transformée de Fourier :

$$\log(f(\boldsymbol{y}_{A_k})) = -0.5 \times |A_k| \times \log(2\pi) - 0.5 \times \log(\det(\boldsymbol{\Sigma})) - 0.5 \times (\boldsymbol{y}_{A_k} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}_{A_k} - \boldsymbol{\mu})$$
$$\det(\boldsymbol{\Sigma}) = \prod V_p, \quad \boldsymbol{\Sigma}^{-1} = F_o^{-1} \times \mathrm{M}V_p^{-1} \times F_o \quad \text{et} \quad \boldsymbol{\Sigma}^{-1} \times (\boldsymbol{y}_{A_k} - \boldsymbol{\mu}) = F_o^{-1}(1/V_p \times F_o(\boldsymbol{y}_{A_k} - \boldsymbol{\mu}))$$

avec $\det(\boldsymbol{\Sigma})$ le déterminant de $\boldsymbol{\Sigma}$, V_p les valeurs propres de la matrice $\boldsymbol{\Sigma}$, $\mathrm{M}V_p$ la matrice diagonale des valeurs propres, F_o la matrice des coefficients de Fourier et F_o^{-1} son inverse.

Puisque la vraisemblance marginale admet une écriture analytique, il peut en être déduit celle des densités conditionnelles. Ainsi, $f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}})$ et $f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}})$ s'écrivent :

$$f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) = \mathcal{N}(y_{jn}; \mu_n, \sigma_y^2 + \sigma_n^2) \quad \text{et} \quad f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) = \mathcal{N}(\boldsymbol{y}_{jt}; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$$

avec

$$\sigma_n^2 = \mathrm{var}(\phi_k|\boldsymbol{y}_{A_k^{-jn}}) = \left(\frac{1}{\sigma_0^2} + \frac{|A_k| - 1}{\sigma_y^2}\right)^{-1} \quad ; \quad \mu_n = \mathbb{E}[\phi_k|\boldsymbol{y}_{A_k^{-jn}}] = \sigma_n^2 \left(\frac{\mu_0}{\sigma_0^2} + \frac{\sum\limits_{(j',n') \in A_k^{-jn}} y_{j'n'}}{\sigma_y^2}\right)$$

$$\boldsymbol{\Sigma}_t = \begin{bmatrix} \sigma_y^2 + \sigma_p^2 & \sigma_p^2 & \cdots & \sigma_p^2 \\ \sigma_p^2 & \sigma_y^2 + \sigma_p^2 & \cdots & \sigma_p^2 \\ & & \ddots & \\ \sigma_p^2 & \sigma_p^2 & \cdots & \sigma_y^2 + \sigma_p^2 \end{bmatrix} \quad ; \quad \boldsymbol{\mu}_t = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \mu_p$$

où $\sigma_p^2 = \left(\frac{1}{\sigma_0^2} + \frac{|A_k| - \nu_{jt}}{\sigma_y^2}\right)^{-1}$; $\mu_p = \sigma_p^2 \left(\frac{\mu_0}{\sigma_0^2} + \frac{\sum\limits_{(j',n') \in A_k^{-jt}} y_{j'n'}}{\sigma_y^2}\right)$

Les valeurs des paramètres des modèles DP, DP-Potts, HDP et HDP-Potts sont fixés comme suit :

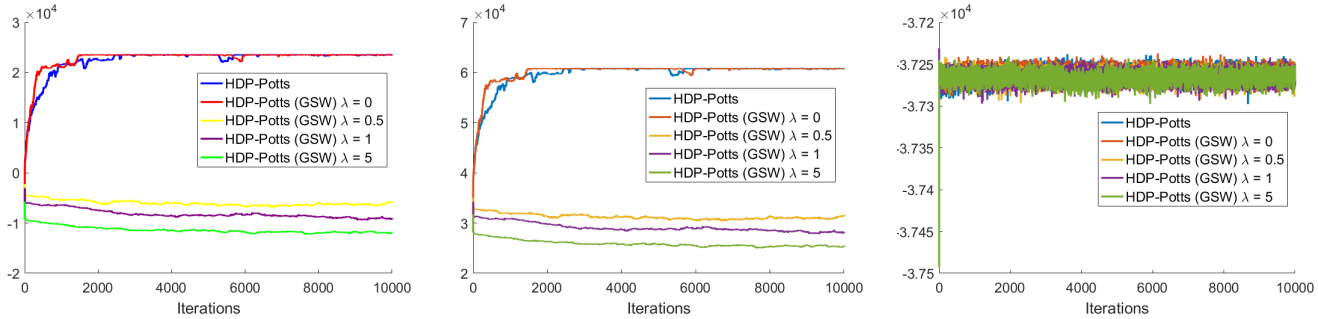
- $\alpha = 1$
- $\beta = 0.8$ (pour $\sigma_y = 5$) et $\beta = 1.2$ (pour $\sigma_y = 15$)
  Etant donné que pour un faible niveau de bruit, une faible régularisation spatiale suffit, nous choisissons de diminuer la valeur de β pour vérifier que les algorithmes restent cohérents à cette hypothèse.
- $\alpha_0 = 1, \gamma = 1$

Le nombre d'itérations total est choisi comme $T = 10000$, le temps de chauffe est fixé à $T/2$ et le pas à 10 c'est-à-dire qu'un échantillon sur dix est choisi.

Commentaires

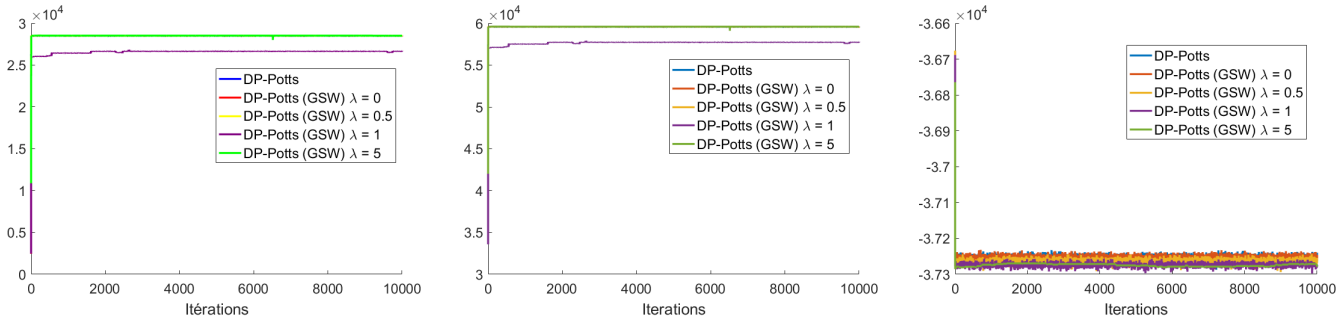
Etudions d'abord la procédure d'initialisation des différents algorithmes, notons K_{init} le nombre de classes initial. Pour le HDP et le HDP-Potts, le nombre de régions initial est choisi

FIGURE 5.2 – Images-test vraies.



(a) (b) (c)

FIGURE 5.3 – Tracé pour le modèle HDP-Potts (Gibbs et GSW) du logarithme de la loi *a posteriori* non normalisée (a), du logarithme de la loi *a priori* non normalisée (b) et du logarithme de la vraisemblance (c) avec $\sigma_y = 5$ avec $K_{\text{init}} = 15$.



(a) (b) (c)

FIGURE 5.4 – Tracé pour les modèles DP-Potts (Gibbs et GSW) du logarithme de la loi *a posteriori* non normalisée (a), du logarithme de la loi *a priori* non normalisée (b) et du logarithme de la vraisemblance (c) avec $\sigma_y = 5$ avec $K_{\text{init}} = 15$.

$\sigma_y = 5$

	DP					DP-Potts				
	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$
Image 1	0.98	0.96	0.91	2.57		0.99	0.99	0.99	0.018	
Image 2	0.97	0.96	0.9	2.87	2.25	0.99	0.99	0.99	0.062	0.079
Image 3	0.98	0.97	0.92	2.14		0.99	0.99	0.99	0.181	

	HDP					HDP-Potts				
	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$
Image 1	0.99	0.99	0.97	0.45		1	1	1	0	
Image 2	0.99	0.98	0.93	1.81	1.32	0.99	0.99	0.99	0.101	0.102
Image 3	0.98	0.98	0.92	2.09		0.99	0.99	0.99	0.248	

TABLE 5.1 – Comparaison des métriques dans le cas $\sigma_y = 5$.

$\sigma_y = 15$

	DP					DP-Potts				
	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$
Image 1	0.59	0.39	0.3	32.17		0.98	0.97	0.91	1.94	
Image 2	0.44	0.25	0.18	36.85	31.88	0.98	0.97	0.91	2.85	2.43
Image 3	0.48	0.3	0.2	34.18		0.98	0.96	0.89	3.11	

	HDP					HDP-Potts				
	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$	OA	κ_{co}	V_b	$\%L_D$	$\%L_{D_g}$
Image 1	0.83	0.67	0.45	18.09		0.62	0.33	0.79	9.84	
Image 2	0.59	0.38	0.2	37.45	29.18	0.44	0.13	0.58	33.67	27.88
Image 3	0.55	0.32	0.21	34.07		0.41	0.13	0.59	32.25	

TABLE 5.2 – Comparaison des métriques dans le cas $\sigma_y = 15$.

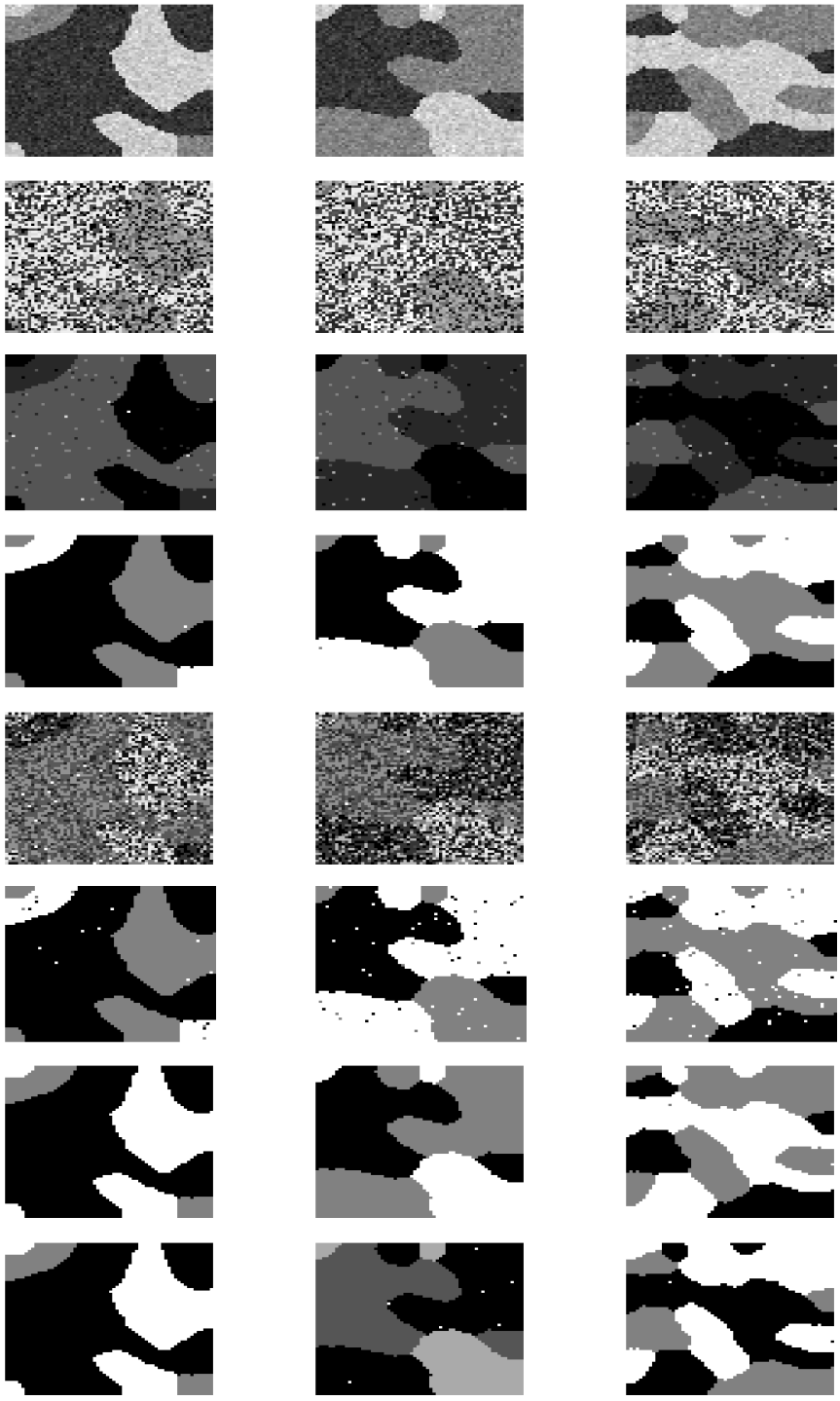


FIGURE 5.5 – Images bruitées avec un niveau de bruit $\sigma_y = 5$, initialisation et résultats de segmentation obtenus pour les DP, DP-Potts, HDP et HDP-Potts avec $K_{\text{init}} = 15$.

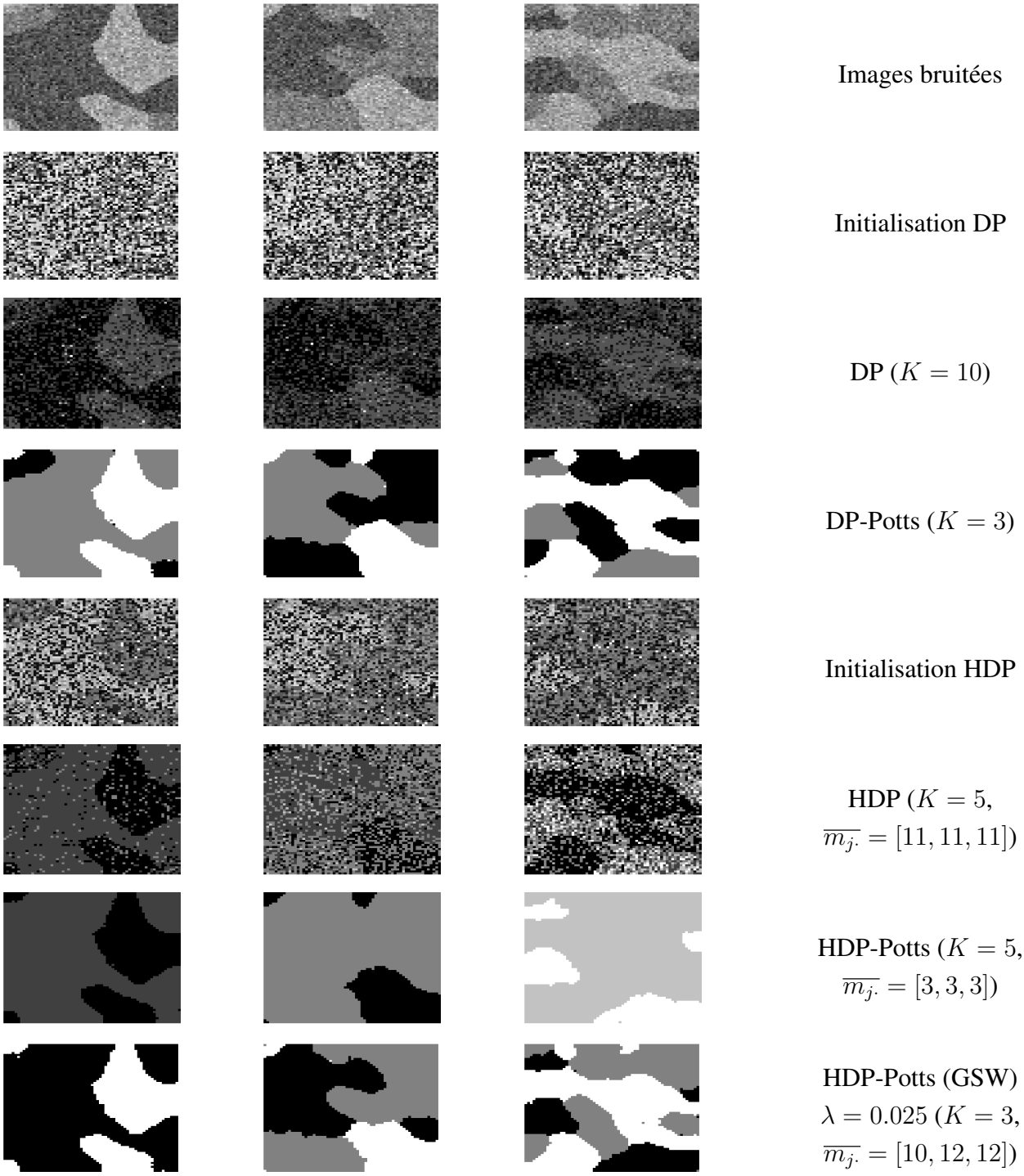


FIGURE 5.6 – Images bruitées avec un niveau de bruit $\sigma_y = 15$, initialisation et résultats de segmentation obtenus pour les DP, DP-Potts, HDP, HDP-Potts et HDP-Potts(GSW) avec $\lambda = 0.025$ avec $K_{\text{init}} = 15$.

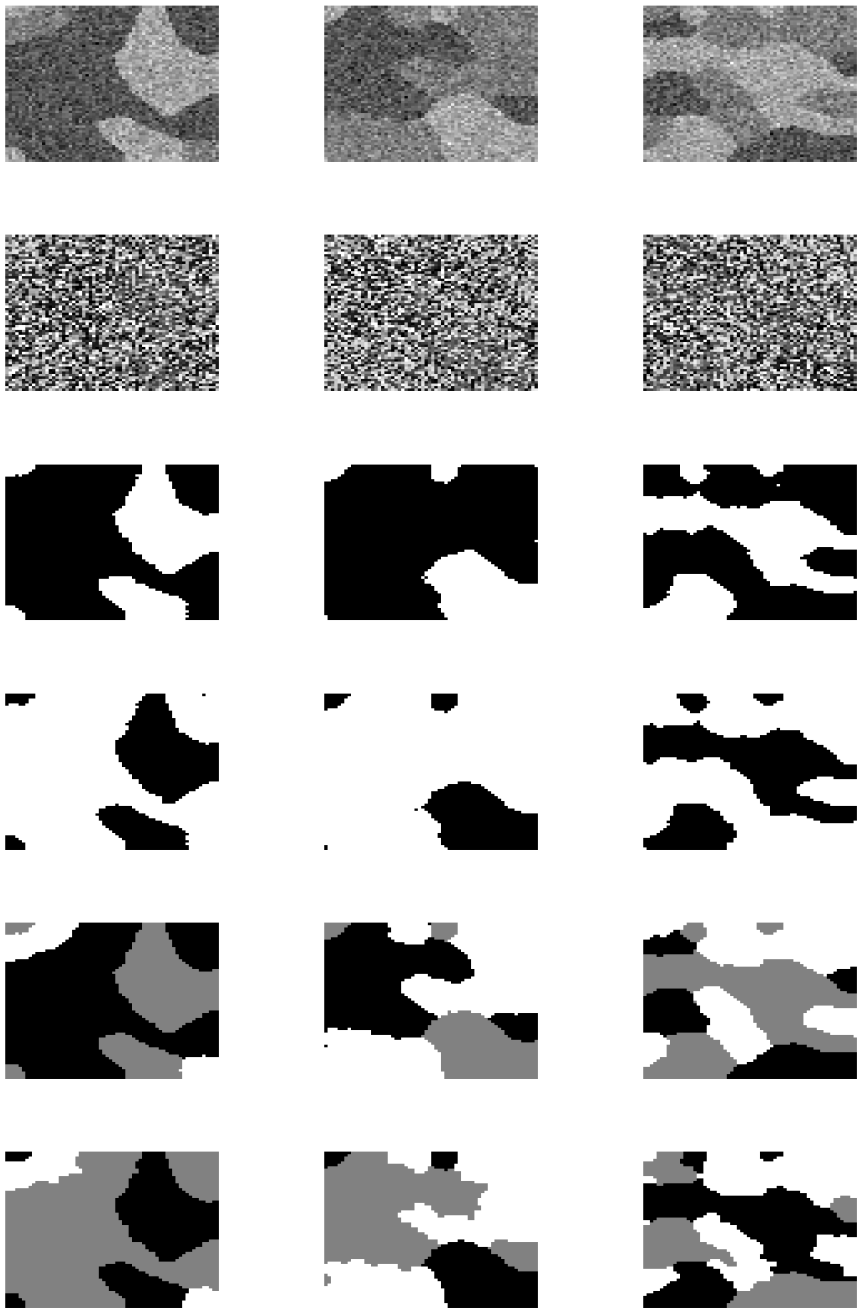
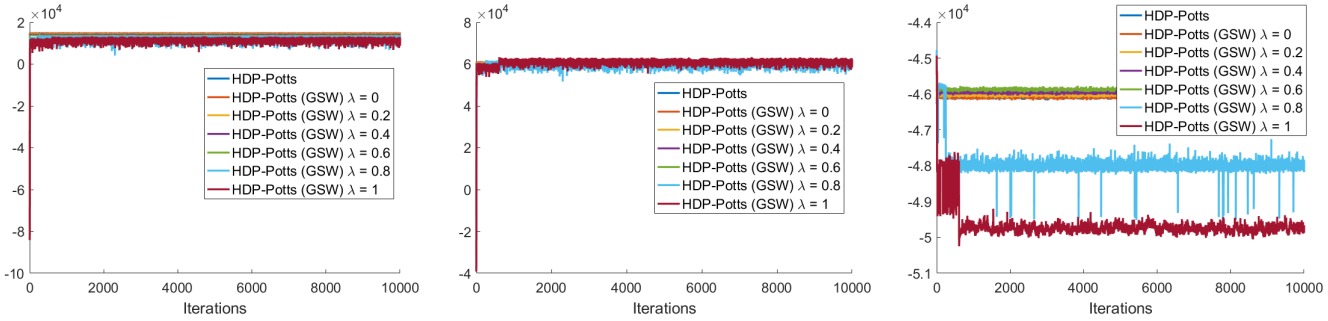


FIGURE 5.7 – Images bruitées avec un niveau de bruit $\sigma_y = 15$ et résultats de segmentation obtenus pour les DP-Potts et DP-Potts(GSW) avec $\lambda = 0.5$ et $\lambda = 5$ avec $K_{\text{init}} = 30$.



(a) (b) (c)

FIGURE 5.8 – Cas où le champ de Potts est placé sur les régions. Tracé pour le modèle HDP-Potts (Gibbs et GSW) du logarithme de la loi *a posteriori* non normalisée (a), du logarithme de la loi *a priori* non normalisée (b) et du logarithme de la vraisemblance (c) avec $\sigma_y = 10$ avec $K_{\text{init}} = 15$.

comme : $m_{j\cdot\text{init}} = 3 \times K_{\text{init}}; j = 1, \cdots, J$. Concernant la méthode d'initialisation, considérons d'abord un tirage aléatoire ; les figures E.1, E.2, E.3, et E.4, représentent des résultats obtenus par les algorithmes HDP-Potts et DP-Potts avec $K_{\text{init}} = 15$. Pour un niveau de bruit faible, $\sigma_y = 5$, la classification est retrouvée, néanmoins pour $\sigma_y = 15$, les résultats de segmentation sont insatisfaisants, il est alors nécessaire d'appliquer un procédé plus adapté. Du point de vue des observations, la classification attendue est une identification de lois gaussiennes composant un mélange. Ainsi, pour une initialisation plus cohérente, nous nous proposons dans la suite de la réaliser via le k-means.

Concernant l'affichage des résultats, K et $\overline{m_{j\cdot}}$ désignent respectivement le nombre de classes correspondant à la segmentation induite et le nombre moyen de régions au cours des itérations pour chaque image. Notons de plus que les couleurs attribuées aux classes sont arbitraires.

Les tableaux 5.1 et 5.2 présentent les valeurs des métriques obtenues pour $\sigma_y = 5$ et $\sigma_y = 15$ avec $K_{\text{init}} = 15$. On remarque que les résultats obtenus sont meilleurs avec le champ de Potts. On remarque en particulier dans le tableau 5.2 l'importance de métriques indépendantes de la numérotation. Ainsi, pour le HDP-Potts, la valeur de V_b est légèrement plus élevée que celle de l'OA. Inversement, pour le HDP, l'OA donne de meilleurs résultats que le V_b. Ces différences sont dues à la renumérotation, en effet, cette dernière ne dépend que du nombre de pixels affectés à chaque classe dans la vraie partition. La métrique V_b évalue le fait que seuls les pixels affectés dans une classe pour la partition vraie le soient dans la partition estimée, les résultats obtenus sont ainsi plus pertinents lorsque, par exemple, le nombre de classes estimé est différent du nombre de classes vrai. En outre, les configurations où la partition estimée n'est pas cohérente avec la vraie, du point de vue des binômes de pixels, ne sont pas pénalisées lors de la renumérotation et donc du calcul des métriques OA et κ_{co}, contrairement à la métrique $\%L_D$.

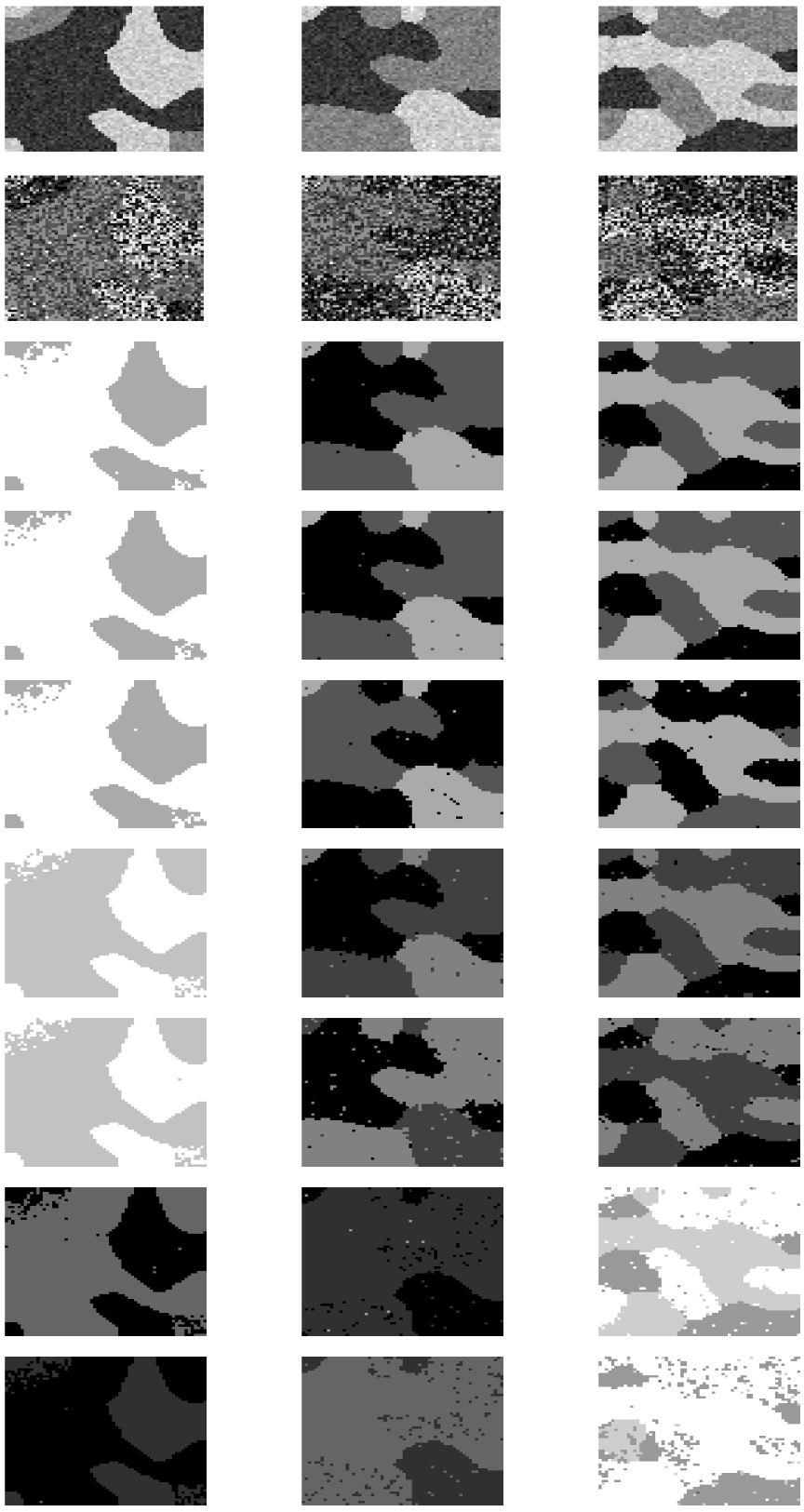


FIGURE 5.9 – Cas où le champ de Potts est placé sur les régions. Images bruitées avec un niveau de bruit $\sigma_y = 10$, initialisation et résultats de segmentation obtenus pour le HDP-Potts (Gibbs et GSW) avec $\beta = 0.8$ et $K_{\text{init}} = 15$.

De plus, considérant toujours les résultats obtenus avec le HDP et HDP-Potts, il est visible que les métriques V_b et $\%L_D$ évaluent différents critères ; en effet, les valeurs de $\%L_D$ sont assez faibles pour celles du V_b correspondant.

Les figures 5.3 et 5.4 donnent une comparaison des lois *a posteriori* non normalisées des modèles HDP-Potts et DP-Potts pour des échantillons obtenus par un algorithme de Gibbs et avec un algorithme de Swendsen-Wang généralisé où $\lambda = 0$, $\lambda = 0.5$, $\lambda = 1$ et $\lambda = 5$. Comme attendu, les lois obtenues pour le Gibbs et pour le Swendsen-Wang où $\lambda = 0$ se superposent. Lorsque λ varie, l'algorithme est bloqué dans des configurations locales, suivant l'évolution des lois *a priori* liées, en particulier, le nombre de régions augmente avec λ. Néanmoins, le choix de λ est délicat, en effet, les pixels sont liés avec la probabilité $1 - \exp(-\beta\lambda)$, à β fixé, plus λ est grand et plus grande est la probabilité pour deux pixels voisins de même classe (ou région selon le modèle) d'être liés. La figure 5.7 illustre les effets d'agrégation liés à une valeur de λ élevée ($\lambda = 5$).
Lorsque λ est grand, l'algorithme favorise la création de « gros » spin-clusters, dont la vraisemblance d'être liés à une région existante est plus faible que celle d'appartenir à une région propre. En revenant aux équations (4.12), (4.13) et (4.14), on observe que l'échantillonnage des étiquettes de lien dépend uniquement des étiquettes de région alors que l'échantillonnage des étiquettes de région dépend aussi des étiquettes de classe. Ainsi, la création des spin-clusters dépend de la segmentation en régions courante et l'affectation d'un spin-cluster à une nouvelle région n'est pas systématiquement discriminée. En effet, selon la valeur de λ et le nombre de pixels voisins, le terme dépendant des étiquettes de région est négligeable devant celui dépendant des étiquettes de classe. Ainsi, la sur-segmentation en régions reste vraisemblable tant que les classes affectées sont cohérentes avec la segmentation globale. En effet, lorsque le champ de Potts est placé sur les étiquettes de classe, en passant au Swendsen-Wang, en termes de vraisemblance, il n'est pas discriminatoire d'avoir plusieurs régions qui codent pour la même classe. Ce qui explique qu'à la figure 5.3, des configurations avec des poids *a priori* « faibles » soient échantillonnées. Une solution pour résoudre cette difficulté est d'opter pour le second modèle où l'algorithme de Swendsen-Wang est défini sur les étiquettes de région.

Les figures 5.8 et 5.9 présentent des résultats obtenus pour la segmentation jointe lorsque le champ de Potts est maintenant appliqué aux étiquettes de région. La figure 5.8 confirme les hypothèses faites quant au comportement observé à la figure 5.3. De plus à la figure 5.8, il est visible que la vraisemblance est ici discriminatoire et les poids *a priori* sont quasiment identiques. On observe en outre à la figure 5.9 que plus la valeur de λ augmente et moins le résultat de segmentation est satisfaisant, comme observé à la figure 5.6. Faut-il appliquer le champ de Potts sur les étiquettes de classe ou de région ? Dans le premier cas, les résultats de segmentation sont plus satisfaisants tandis que le deuxième cas permet de mieux interpréter les résultats obtenus avec l'algorithme de Swendsen-Wang.

Des résultats de segmentation sont présentés à la figure 5.6 pour $\sigma_y = 15$ et $K_{\text{init}} = 15$. Le DP-Potts donne de bons résultats ; cependant, en choisissant $K_{\text{init}} = 30$, le modèle ne détecte que deux classes, comme illustré à la figure 5.7. L'intérêt de l'algorithme Swendsen-Wang généralisé apparaît alors ; en effet, avec $\lambda = 0.5$ pour le DP-Potts et $\lambda = 0.025$ pour le HDP-Potts, on obtient la segmentation attendue.

Pour confirmer cette analyse, un problème de plus grande dimension a été considéré pour lesquels des résultats sont proposés en annexe. Il y apparaît que le DP-Potts et le HDP-Potts conduisent à une bonne segmentation mais échouent à retrouver les classes sous-représentées dans les images.

En conclusion, les segmentations induites par les DP-Potts et HDP-Potts sont bonnes, le HDP-Potts apporte néanmoins des informations supplémentaires. Il permet une gestion différenciée des images et la division en régions par image donne la segmentation de chacune, comme théoriquement attendu. Ainsi, le modèle hiérarchique peut être requis ou pas ; il apporte néanmoins une information inaccessible lorsqu'il n'est pas utilisé. Prenons l'exemple de l'analyse d'un espace agricole d'une surface donnée divisé en portions. Le modèle hiérarchique pourrait permettre de délimiter les zones agricoles dans chaque portion, indépendamment de l'espace entier considéré.

5.3 Résultats : cas d'une base de données

Génération des données

La base de données LabelMe est une base d'images gratuite qui a pour but de fournir un outil d'annotation d'images en ligne pour la recherche en vision par ordinateur. Les images issues de la base d'images ont été pré-segmentées en super-pixels. Le descripteur associé à chaque super-pixel est choisi ici comme étant un histogramme. Dans ce cas, les paramètres cachés de classe sont les probabilités d'une loi catégorielle et la vraisemblance des données est souvent définie comme la loi multinomiale. Pour pouvoir calculer explicitement la vraisemblance marginale, il est nécessaire que la distribution *a priori* sur les paramètres soit une loi conjuguée à la vraisemblance multinomiale. Cette loi de probabilité est la loi de Dirichlet qui sera la distribution de base de notre HDP. Notons ϕ_k le vecteur de paramètres, le modèle d'observation s'écrit alors :

$$\begin{aligned} \phi_k|\Upsilon &\sim \Upsilon \equiv \text{Dir}(\varphi_0) \\ y_{jn}|\phi_k &\sim f(\cdot|\phi_k) \equiv \text{Mult}(y_{jn};\phi_k) \end{aligned}$$

avec

$$\upsilon(\phi_k) = \frac{1}{\mathcal{D}(\varphi_0)} \prod_{i=1}^{N_{\text{bi}}} \phi_k[i]^{\varphi_0[i]-1} \qquad f(y_{jn}|\phi_k) = \frac{1}{\mathcal{M}(y_{jn})} \prod_{i=1}^{N_{\text{bi}}} \phi_k[i]^{y_{jn}[i]}$$

où N_{bi} est le nombre de *bin* des histogrammes.

$$\mathcal{D}(\varphi_0) = \frac{\prod_{i=1}^{N_{\text{bi}}} \Gamma(\varphi_0[i])}{\Gamma\left(\sum_{i=1}^{N_{\text{bi}}} \varphi_0[i]\right)} \qquad \mathcal{M}(y_{jn}) = \frac{\prod_{i=1}^{N_{\text{bi}}} y_{jn}[i]}{\left(\sum_{i=1}^{N_{\text{bi}}} y_{jn}[i]\right)!}$$

Les lois multinomiale et de Dirichlet étant conjuguées, on peut écrire analytiquement la loi marginale des observations associées aux pixels dans la classe k :

$$f(\boldsymbol{y}_{A_k}) = \left[\prod_{(j,n)\in A_k} \frac{1}{\mathcal{M}(y_{jn})}\right] \frac{\mathcal{D}\left(\varphi_0 + \sum_{(j,n)\in A_k} y_{jn}\right)}{\mathcal{D}\left(\varphi_0\right)}$$

Les lois conditionnelles $f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}})$ et $f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}})$ peuvent être déduites :

$$f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) = \frac{f(y_{jn}, \boldsymbol{y}_{A_k^{-jn}})}{f(\boldsymbol{y}_{A_k^{-jn}})} = \frac{1}{\mathcal{M}(y_{jn})} \frac{\mathcal{D}\left(\varphi_0 + y_{jn} + \sum_{(j',n')\in A_k^{-jn}} y_{j'n'}\right)}{\mathcal{D}\left(\varphi_0 + \sum_{(j',n')\in A_k^{-jn}} y_{j'n'}\right)}$$

$$f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) = \frac{f(\boldsymbol{y}_{jt}, \boldsymbol{y}_{A_k^{-jt}})}{f(\boldsymbol{y}_{A_k^{-jt}})} = \frac{1}{\prod_{q|c_{jq}=t} \mathcal{M}(y_{jq})} \frac{\mathcal{D}\left(\varphi_0 + \sum_{q|c_{jq}=t} y_{jq} + \sum_{(j',n')\in A_k^{-jt}} y_{j'n'}\right)}{\mathcal{D}\left(\varphi_0 + \sum_{(j',n')\in A_k^{-jt}} y_{j'n'}\right)}$$

Le modèle précédent peut être enrichi pour rendre la segmentation plus robuste à différents facteurs tels que la texture ou les changements d'illumination. Le HOG peut alors être associé à l'histogramme de niveaux de gris pour définir un nouveau descripteur. L'histogramme normalisé du HOG est construit comme décrit au chapitre 3 puis il est multiplié par le nombre de pixels dans le super-pixel considéré pour le ramener à l'échelle. Le nouveau descripteur s'écrit alors : $y_{jn} = (y_{jn}^{\text{c}}, y_{jn}^{\text{h}})$ où y_{jn}^{c} et y_{jn}^{h} représentent respectivement l'histogramme en niveaux de gris (ou en couleurs) et l'histogramme du HOG et sont supposés indépendants, c'est-à-dire

$$\begin{aligned} \phi_k = (\phi_k^{\text{c}}, \phi_k^{\text{h}}),\ \varphi_0 = (\varphi_0^{\text{c}}, \varphi_0^{\text{h}}) \\ \phi_k|\Upsilon \sim \text{Dir}(\varphi_0^{\text{c}})\text{Dir}(\varphi_0^{\text{h}}) \\ y_{jn}|\phi_k \sim \text{Mult}(\phi_k^{\text{c}})\text{Mult}(\phi_k^{\text{h}}) \end{aligned}$$

Par la suite, les valeurs des paramètres des modèles DP-Potts et HDP-Potts sont fixés comme :

- $\alpha = 1$
- $\beta = 1$
- $\alpha_0 = 1, \gamma = 1$
- pour les données en niveaux de gris, $\varphi_0^{\text{c}} = 1 \times 10^4 \times \overline{y}^{\text{c}}$, où $\overline{y}^{\text{c}}$ est la moyenne des données-histogrammes en niveaux de gris et pour les données en RGB, $\varphi_0^{\text{c}} = 3 \times 10^4 \times \overline{y}^{\text{c}}$, où $\overline{y}^{\text{c}}$ est la moyenne des données-histogrammes en RGB

Le nombre d'itérations total est choisi comme $T = 10000$, le temps de chauffe est fixé à $T/2$ et le pas à 10.

Commentaires

Les images segmentées provenant de la base d'images LabelMe sont de taille 256×256, elles ont été pré-segmentées à l'aide des méthodes SLIC et *normalized-cuts* en approximativement 1000 super-pixels et le nombre de *bins* est choisi comme $N_{\mathrm{bi}} = 16$.

Trois ensembles d'images ont été segmentés : des images de route, des images de bâtiments et un ensemble d'images mixtes comportant à la fois des routes et des bâtiments. Pour chaque ensemble et pour chaque pré-segmentation sont présentés les résultats obtenus avec le DP-Potts, le HDP-Potts (Gibbs) ainsi que ceux obtenus en prenant en compte l'histogramme de gradients orientés en niveaux de gris mais aussi en couleurs (RGB).

La figure 5.10 montre la segmentation obtenue avec une pré-segmentation par la méthode *normalized-cuts* pour des images de route. On observe à nouveau que les segmentations avec le DP-Potts et le HDP-Potts sont similaires, les classes *route* et *arbre* sont notamment bien définies sauf pour la première image où des effets d'illumination entraînent que l'herbe est confondue avec la route. Néanmoins, la classe *ciel* pose problème du fait non seulement des effets de lumière mais aussi de la présence de nuages. Les restrictions observées en raison des différences d'illumination sont en partie résolues par l'utilisation du HOG. En outre, lorsqu'il est appliqué au RGB, la segmentation est plus homogène, les classes sont mieux délimitées, sauf la *route* qui reste souvent confondue avec l'herbe.

Considérons à présent la pré-segmentation par la méthode SLIC, la segmentation obtenue en figure 5.11 laisse entrevoir plus de détails de l'image originale qu'avec la méthode *normalized-cuts* : les nuages dans le *ciel* sont beaucoup plus marqués, les poteaux présents dans les deux dernières images sont aussi détectés ainsi que les limites de la route. Néanmoins, la *route* n'est pas correctement segmentée dans la première image et cela n'est pas corrigé avec l'ajout du HOG, pour le DP-Potts, mais en partie pour le HDP-Potts. En effet, on remarque une identification de la *route* dans la première image mais elle est reconnue identique à l'herbe.

Le deuxième ensemble d'images est constitué d'images de bâtiments. La segmentation obtenue avec la pré-segmentation par *normalized-cuts* est assez mauvaise ; cela peut s'expliquer par le fait que cette méthode a tendance à fournir des super-pixels de forme arrondie qui ne sont pas adaptés pour les bâtiments. Les classes *ciel* et *verdure* sont néanmoins reconnues. Quel que soit le modèle de segmentation utilisé, les zones de verdure de la première image ne sont pas reconnues identiques à celles des autres. En outre, dans les deux dernières images, avec le DP-Potts et le HDP-Potts, toutes les zones de verdures sont identifiées mais confondues tandis qu'avec le HOG (appliqué au niveaux de gris), des végétations réellement différentes (herbe et arbres) sont détectées : celle qui borde les bâtiments reconnue identique dans les deux et celle qui est au sol. De même, la différence de luminance du ciel entraîne qu'il n'est pas reconnu comme étant le même dans les trois images et cela n'est pas « rattrapé » avec le HOG.

Les figures 5.12 et 5.13 présentent les résultats obtenus avec la pré-segmentation SLIC. On remarque qu'il y a beaucoup de détails, ce qui permet de bien délimiter les bâtiments ; néanmoins, les bâtiments sont trop différents pour être reconnus similaires.

Le dernier ensemble d'images à segmenter est un mélange d'images de route et de bâtiments 5.14 et 5.15. On remarque que la pré-segmentation *normalized-cuts* entraîne une plus mauvaise segmentation qu'avec SLIC. De plus, la seule classe détectée conjointement est la classe *arbre* dans la deuxième image de bâtiment et les deux images de route. Comme observé avec l'étude de segmentation des images de bâtiments, la différence entre les bâtiments est significative et ne permet pas de les identifier comme similaires, même dans la même image (la dernière) ; de même que les classes *route* et *ciel*. La meilleure segmentation est obtenue avec le HOG - RGB appliqué au HDP-Potts.

Les différentes figures montrent aussi, pour le HDP-Potts, le découpage en régions correspondant à la segmentation obtenue. Comme attendu, la segmentation en régions correspond à une segmentation individuelle de l'image.

La section E.4 présente des résultats de segmentation obtenus pour les ensembles d'images choisis dans la base d'images avec l'algorithme de Swendsen-Wang généralisé appliqué au HDP-Potts. On remarque qu'ils ne sont pas meilleurs que ceux obtenus avec le HDP-Potts (Gibbs).

De ces observations on peut déduire que la segmentation dépend fortement de la méthode de pré-segmentation adoptée : la forme des super-pixels et la caractéristique qui les réunit sont déterminantes. Les paramètres du modèle sont aussi primordiaux, les valeurs de N_{bi} et φ_0 influencent directement le nombre de classes proposé au travers de la vraisemblance, plus N_{bi} est faible et moins il y aura de classes, par opposition, pour limiter le nombre de classes, il faut fixer φ_0 assez grand.

Les imagettes ont été pré-segmentées à l'aide des méthodes SLIC et *normalized-cuts* en approximativement 3000 super-pixels et le nombre de *bins* est choisi comme $N_{\mathrm{bi}} = 16$. Les imagettes sont de taille 370×512 et le nombre de super-pixels a été choisi pour conserver le même rapport pixels/super-pixels que les images précédentes. La figure 5.16 représente l'image représentant les montagnes Rwenzori et le lac Edouard et les imagettes déduites. Les figures 5.17 et 5.18 présentent les résultats de segmentation obtenus pour cette image. La figure 5.17 montre les résultats obtenus avec une pré-segmentation avec la méthode *normalized-cuts*. On remarque que pour les premières images, la segmentation est assez représentative, en particulier concernant le *ciel*. La segmentation des deux dernières images est pertinente mais visuellement inexploitable. Cela est dû à l'utilisation du *normalized-cuts* qui fournit des super-pixels de formes arrondies inappropriées pour la représentation des *montagnes* de l'image. Sur la figure 5.18, on observe que la pré-segmentation avec la méthode SLIC est pertinente pour ce type d'images. Sur les dif-

férentes figures on observe que les classes partagées sont correctement identifiées, en particulier le *lac*, le *ciel* et le *sol* sur les deux dernières images. Pour le *lac*, sur l'image 3, il y a le lit d'un fleuve en décrue qui est détecté dans la classe *lac*, cela peut s'expliquer si la décrue est récente. Néanmoins, en utilisant le HOG-RGB, ce lit est bien identifié comme différent. Concernant le *ciel*, les méthodes utilisant les niveaux de couleur (RGB) en présentent plus de nuances, l'information de couleur a mis en évidence les différences d'illumination. La classe *montagne* a été identifiée, néanmoins, elle a été subdivisée en plusieurs classes dépendant des différences d'illumination. Le HOG a été partiellement pertinent pour gérer les différences d'illumination sur ce type d'images. Il est intéressant de remarquer que bien que pour les deux premières images, les résultats de segmentation aient l'air inexploitables, les différents reliefs correspondant à la *montagne* et les étendues d'eau sont correctement identifiés.

Le HOG a été joint au modèle pour gérer au mieux les différences d'illumination, mais on remarque qu'il n'est pertinent que pour de faibles variations dans les classes d'intérêt. Les images de bâtiments illustrent par exemple les limitations de cette approche.

Comme observé avec les images-test, de manière générale, les DP-Potts et HDP-Potts donnent une bonne segmentation. Le modèle proposé, le HDP-Potts apporte une segmentation propre des images en plus d'une segmentation globale.

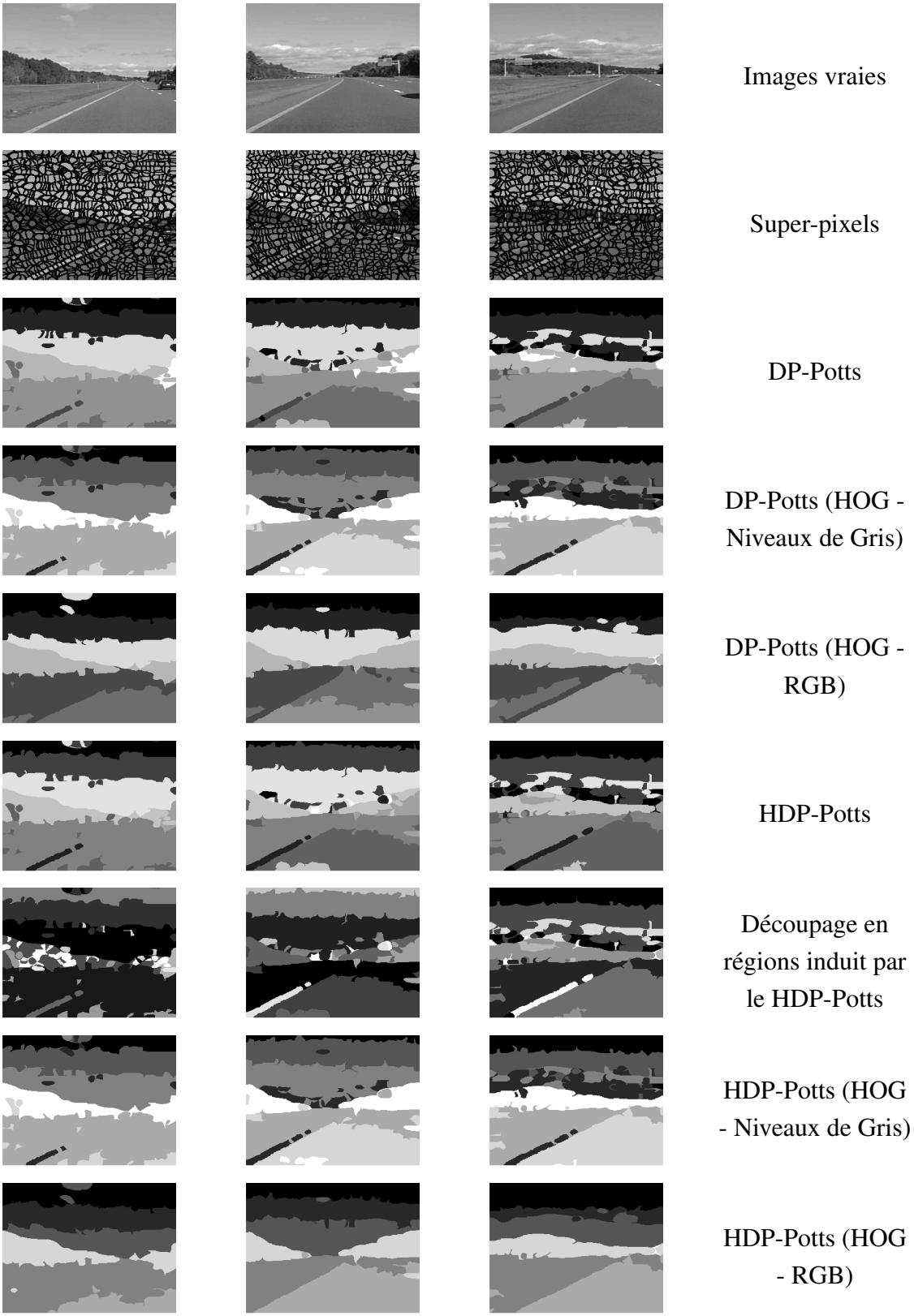


FIGURE 5.10 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

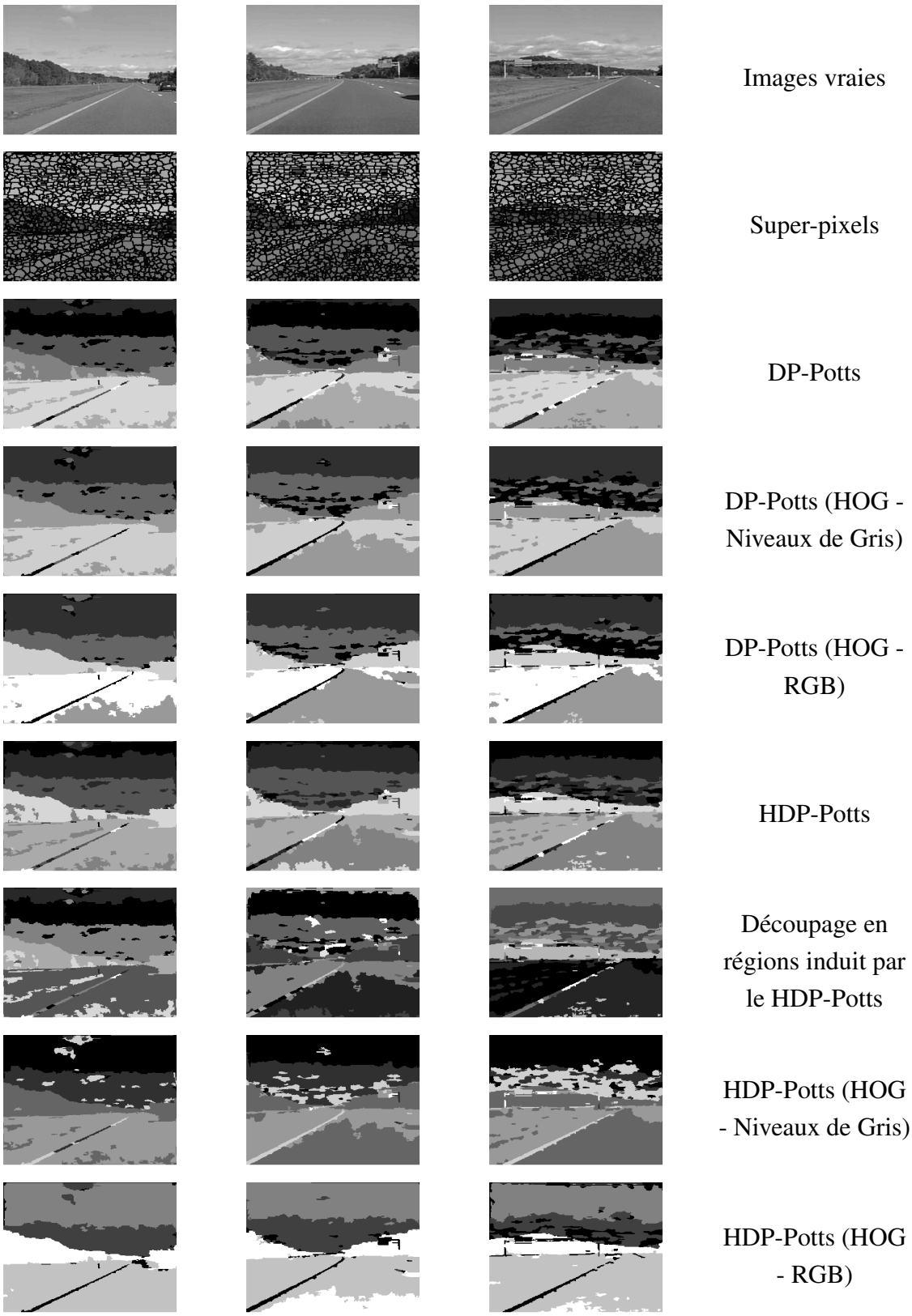


FIGURE 5.11 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

FIGURE 5.12 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

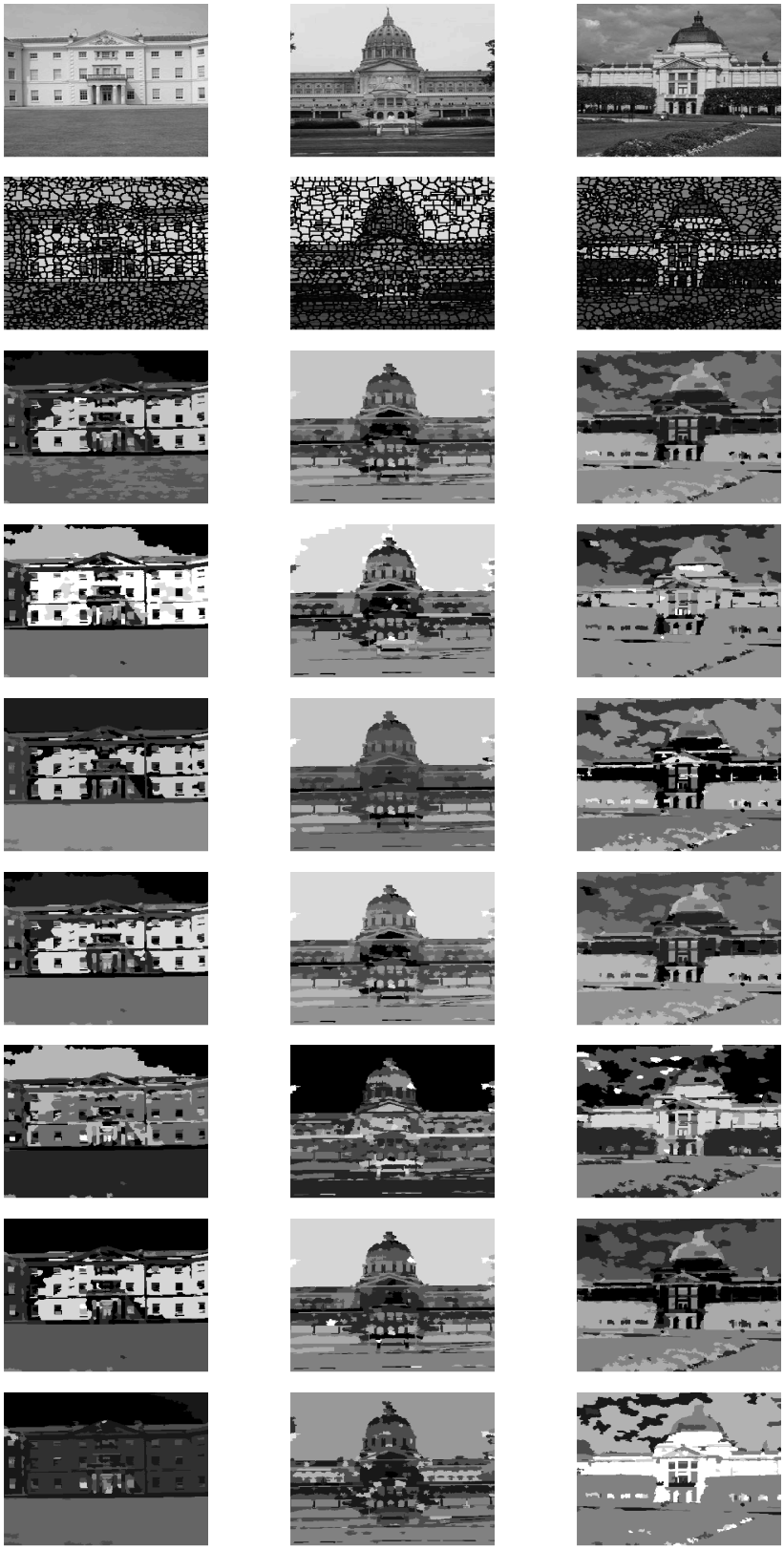


FIGURE 5.13 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

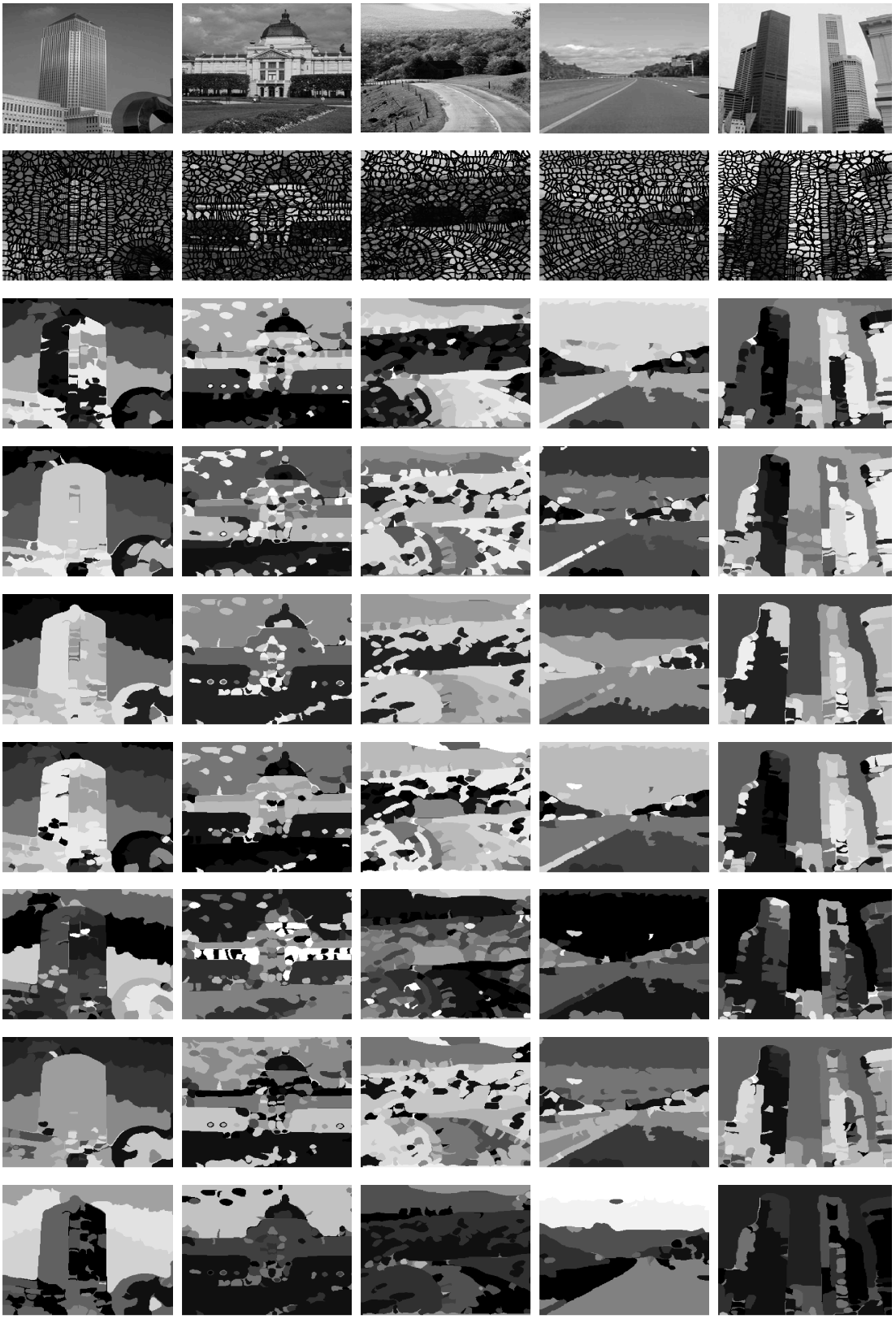


FIGURE 5.14 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

FIGURE 5.15 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

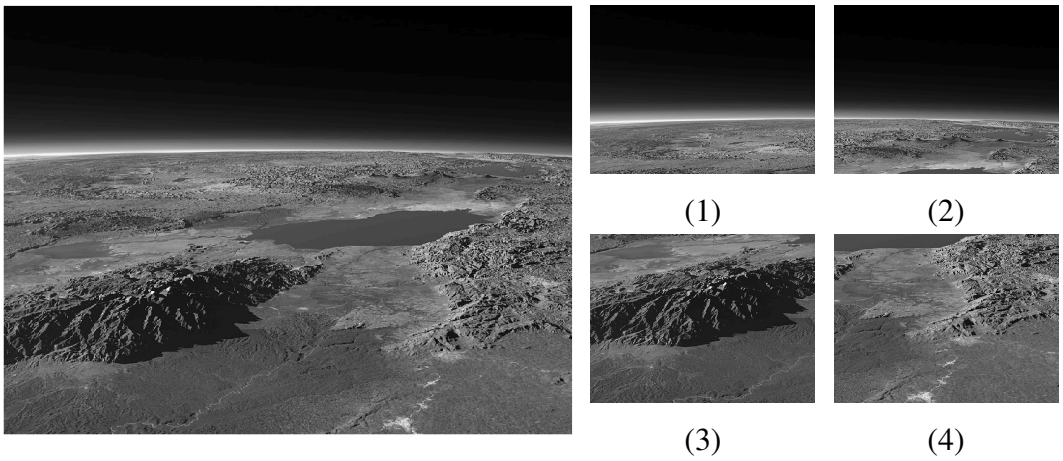


FIGURE 5.16 – A gauche l'image originale et à droite les imagettes issues de l'image des montagnes Rwenzori et du lac Edouard.

5.4 Choix des hyperparamètres

Bien choisir la valeur des hyperparamètres est primordial en classification bayésienne. Dans les parties précédentes, ces-derniers ont été fixés empiriquement.

Il a été montré au chapitre 2 que la valeur des paramètres α pour le processus de Dirichlet et α_0 et γ pour le processus de Dirichlet hiérarchique influe fortement sur le nombre de classes proposées. De plus, lorsque le champ de Potts est ajouté aux modèles, la probabilité d'affectation à une classe existante est plus élevée car multipliée par un terme exponentiel. Ainsi, fixer les hyperparamètres correspondants est d'autant plus difficile qu'il n'est pas aisé d'estimer le nombre de classes *a priori*.

Un algorithme de Monte Carlo séquentiel a été proposé dans le chapitre 4 pour l'estimation des paramètres dans le cas du modèle HDP-Potts.

Dans ce travail, le noyau rétrograde intervenant dans le SMC a été choisi de telle sorte que les poids incrémentaux non normalisés soient faciles à calculer [DDJ06] et s'écrivent comme le rapport de lois *a posteriori* non normalisées :

$$\tilde{w}_m^{(i)} = \frac{\varphi(\boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)} | \boldsymbol{y}, \chi_m)}{\varphi(\boldsymbol{c}_{m-1}^{(i)}, \boldsymbol{d}_{m-1}^{(i)} | \boldsymbol{y}, \chi_{m-1})} \tag{5.8}$$

où φ est la loi *a posteriori* non normalisée des étiquettes. Elle est ainsi définie telle que $\Pr(\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)} | \boldsymbol{y}, \chi_m) = \varphi(\boldsymbol{c}_m^{(i)}, \boldsymbol{d}_m^{(i)} | \boldsymbol{y}, \chi_m)/Z_m$ où Z_m est la constante de normalisation.

De plus, la loi de proposition q_m est le noyau de l'algorithme de Gibbs pour le modèle HDP-Potts.

Le SMC a été appliqué dans le cas d'un ensemble d'images de la base d'images LabelMe,

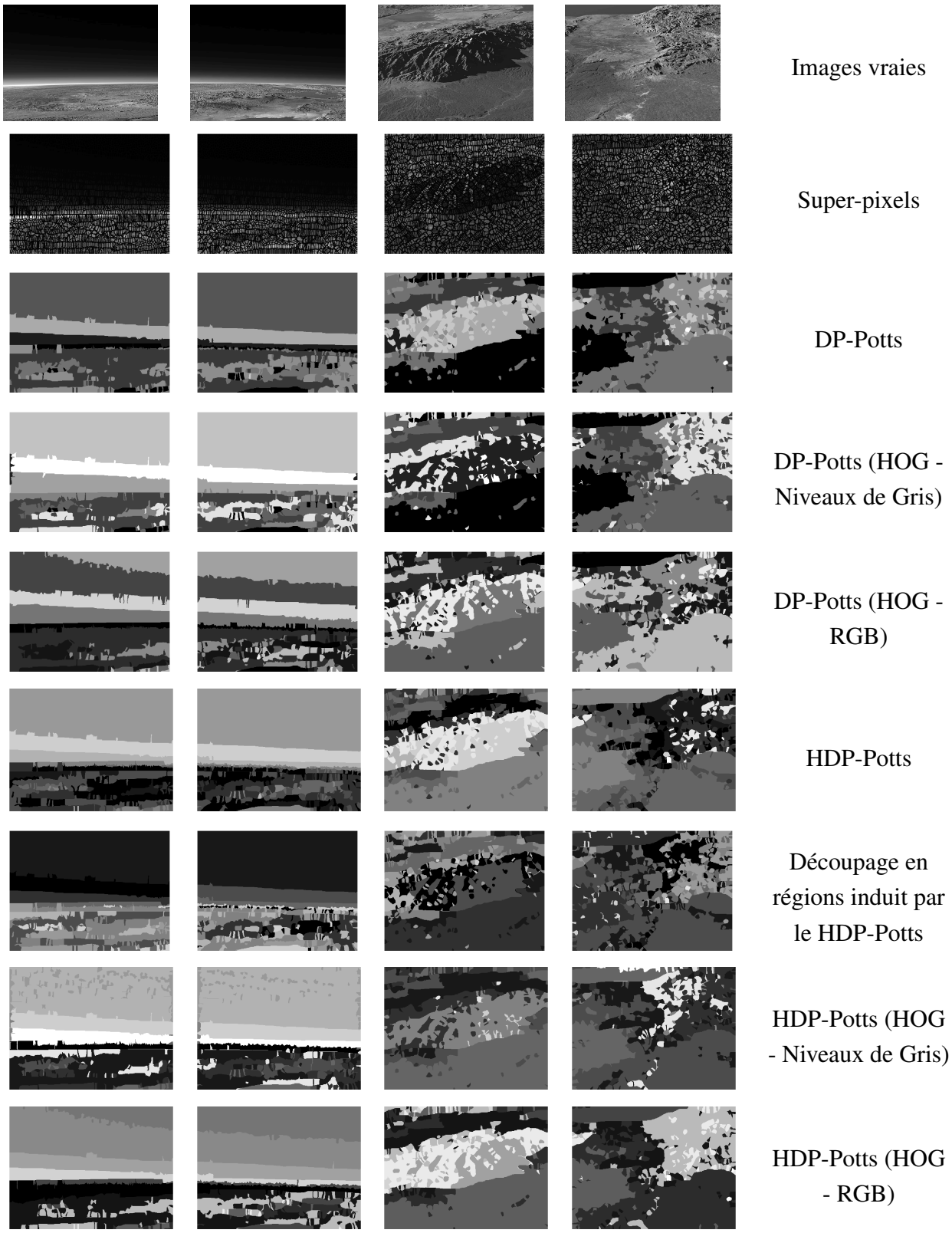


FIGURE 5.17 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

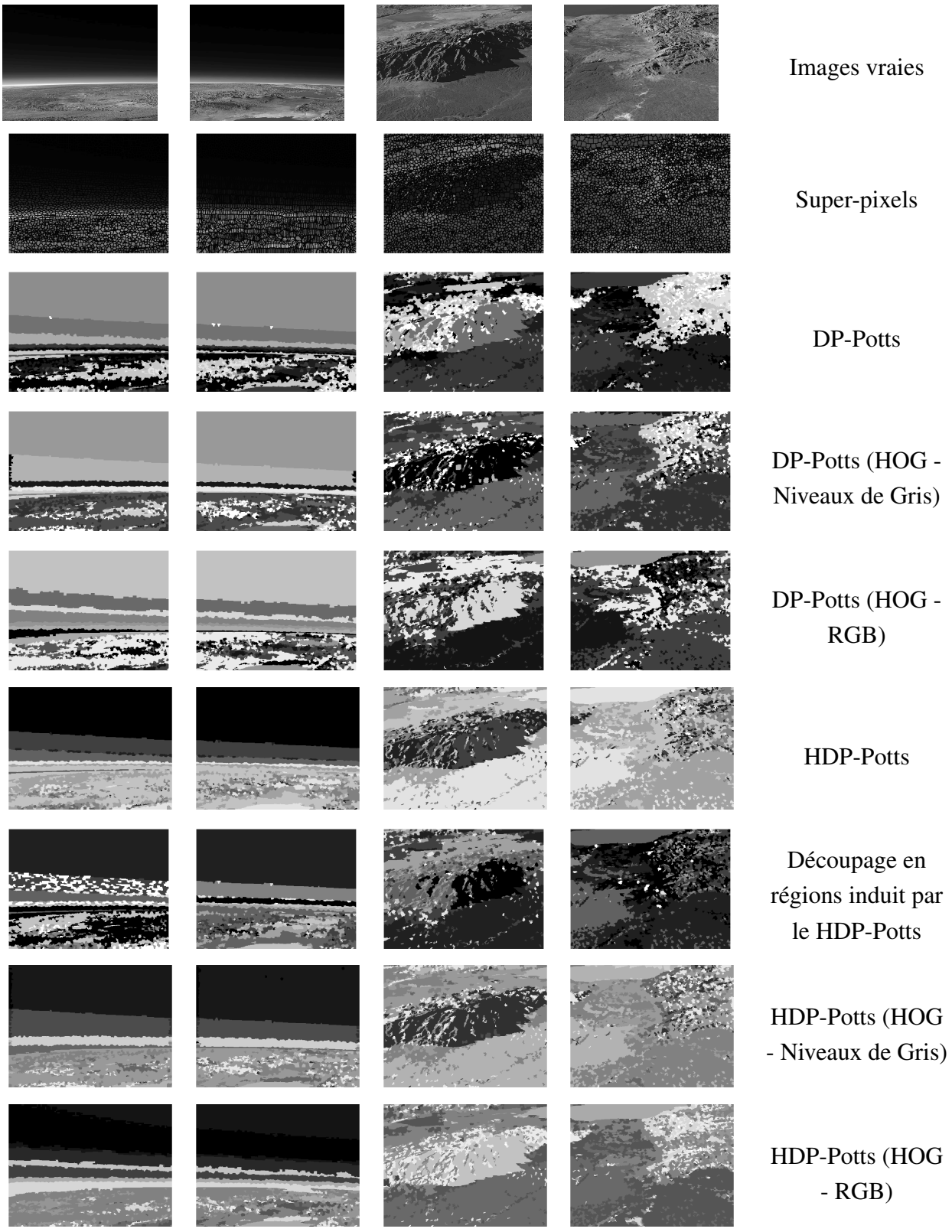


FIGURE 5.18 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC ; sur les suivantes les résultats de segmentation obtenus avec les modèles DP-Potts, DP-Potts combiné au HOG appliqué aux niveaux de gris, DP-Potts combiné au HOG appliqué aux couleurs RGB, HDP-Potts, HDP-Potts combiné au HOG appliqué aux niveaux de gris et HDP-Potts combiné au HOG appliqué aux couleurs RGB.

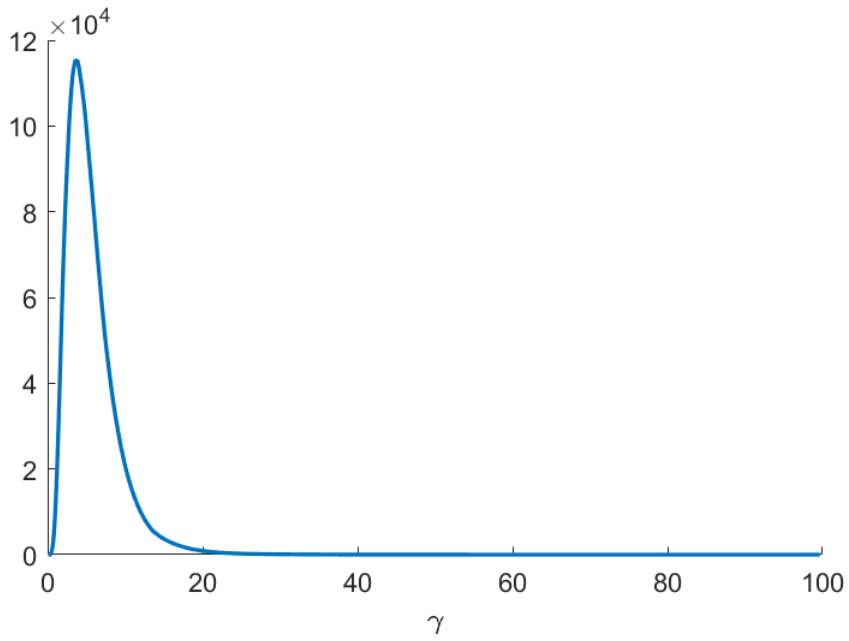


FIGURE 5.19 – Estimation de la vraisemblance marginale pour une grille de valeurs de γ pour l'ensemble d'images de route et une sur-segmentation par la méthode SLIC.

nous montrons ici les résultats obtenus pour l'ensemble d'images de route et avec une sur-segmentation par la méthode SLIC. Nous cherchons ici à estimer γ et les paramètres choisis sont :

- $\beta = 1$
- $\alpha_0 = 1$
- γ variant de 100 à 0.1 sur une grille de 2000 valeurs en décroissance logarithmique
- 100 particules en parallèle
- $\varphi_0^{\mathrm{c}} = 1 \times 10^4 \times \overline{y}^{\mathrm{c}}$, où $\overline{y}^{\mathrm{c}}$ est la moyenne des données-histogrammes en niveaux de gris

On remarque que le produit cumulé des rapports représenté à la figure 5.19 est maximal pour une valeur particulière de γ, ici, $\gamma = 3.65$. Selon la définition de l'algorithme de Monte Carlo séquentiel, la partition optimale peut directement être choisie parmi celles échantillonnées par les particules en parallèle pour le paramètre optimal. Ici, les résultats sont donnés à la figure 5.20.

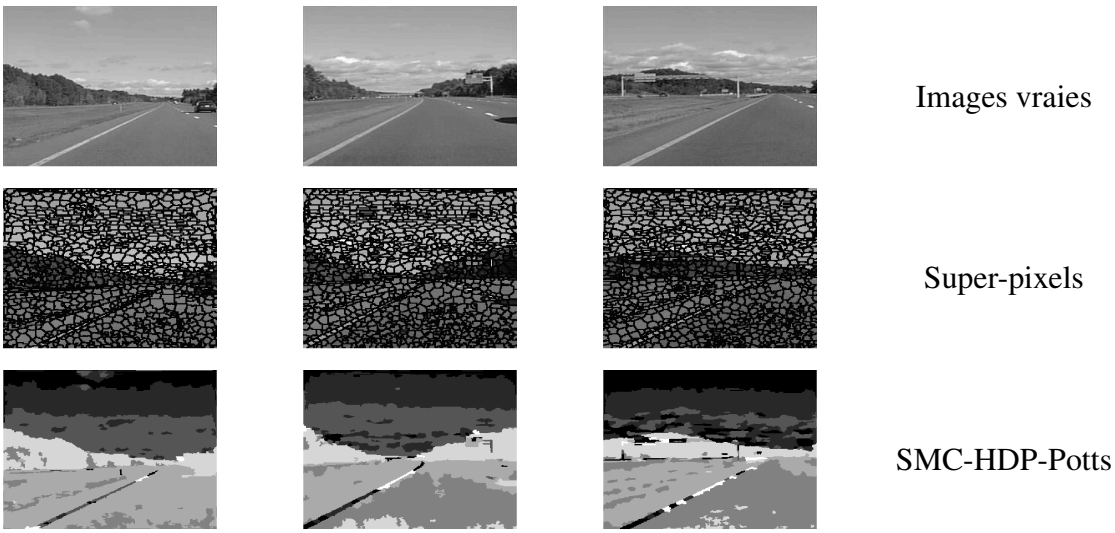


FIGURE 5.20 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur la suivante les résultats de segmentation obtenus avec le SMC appliqué au HDP-Potts pour l'estimation de γ avec $N_{\mathrm{bi}} = 16$.

Conclusion générale et perspectives

Cette thèse a porté sur l'apport des méthodes bayésiennes non paramétriques pour la segmentation d'images. Ces dernières introduisent de la flexibilité dans les modèles *a priori* car elles permettent notamment de s'affranchir de choix du nombre de classes. Cependant, elles ne prennent pas en compte les interactions entre pixels. Nous nous intéressons plus précisément à la segmentation jointe d'un ensemble d'images présentant des classes partagées. Le modèle que nous proposons fait intervenir un processus de Dirichlet hiérarchique qui met à profit la redondance d'informations pour améliorer la segmentation globale de l'ensemble des images et un champ de Potts.

La loi proposée est complexe et de grande dimension, il est ainsi difficile de trouver l'expression analytique de ses modes. Nous proposons alors une exploration via un algorithme de Monte Carlo par chaînes de Markov. Un algorithme de Swendsen-Wang a ensuite été présenté pour améliorer la convergence des chaînes générées par l'algorithme de Gibbs. La valeur des hyperparamètres est déterminante pour l'échantillonnage des étiquettes. Nous avons ainsi décrit un algorithme de Monte Carlo séquentiel pour leur estimation.

Concernant le modèle de vraisemblance associé aux données, il diffère selon le fait que les différentes classes soient représentées par un niveau de gris unique ou encore un histogramme de couleur. Les différentes équations associées ont aussi été écrites. Les images naturelles sont souvent soumises aux différences d'illumination, dépendant du moment de la prise, nous avons ainsi proposé d'utiliser l'histogramme de gradient orienté, proposé en vision par ordinateur, pour rendre notre modèle invariant aux rotations et à la luminance.

Sur le plan méthodologique, les métriques choisies pour évaluer les partitions correspondant aux étiquettes échantillonnées sont assez connues dans le domaine des statistiques mais moins en segmentation d'images. Ces métriques permettent principalement d'évaluer des partitions en s'affranchissant de la numérotation, et donc du problème du *label-switching*.

Des résultats de simulation sont ensuite présentés. Notons que pendant ce travail, des études

intensives ont été menées sur des images-test. De manière générale, les modèles DP-Potts et HDP-Potts donnent de bons résultats de segmentation. Néanmoins, concernant le modèle HDP-Potts, les résultats permettent de confirmer une hypothèse faite. En effet, avec l'utilisation du modèle hiérarchique, une information supplémentaire était attendue : la segmentation en régions pour chaque image du groupe. En outre, l'intérêt de l'algorithme de Swendsen-Wang a été illustré ainsi que les performances de l'algorithme de Monte Carlo séquentiel pour l'estimation des hyperparamètres.

Perspectives

Le nombre moyen de régions dépend de l'hyperparamètre α_0. Dans notre étude, il a été choisi identique pour chaque image. Cependant, pour la plupart des applications, le nombre de régions caractéristiques n'est pas le même dans toutes les images, il est alors intéressant de s'intéresser à une méthode pour pallier ce problème. Ainsi, pour autoriser plus de variabilité, il peut être défini différent pour chacune des images, c'est-à-dire, α_j avec $j = 1, \cdots, J$ et J le nombre d'images à segmenter. Ces hyperparamètres peuvent aussi être estimés conjointement aux étiquettes de régions et de classes.

Un intérêt de notre modèle est sa flexibilité. De nouvelles classes peuvent apparaître à mesure que de nouvelles images sont disponibles. Il serait intéressant d'envisager un traitement séquentiel d'un ensemble d'images récupérées au fil du temps. Le modèle associé devrait alors évoluer avec les observations. Un filtre particulaire pourrait alors être appliqué permettant non seulement de remplir cette condition mais aussi de s'assurer d'échantillonner selon la loi cible à chaque étape.

Une des limitations des méthodes de Monte Carlo par chaînes de Markov est qu'elles sont temporellement coûteuses pour l'échantillonnage et la convergence. Une alternative serait d'utiliser les méthodes variationnelles. Pour ce faire, le modèle génératif peut être réécrit à partir de la représentation en *stick-breaking* et une méthode variationnelle par espérance-maximisation déduite. Il peut être intéressant d'en étudier la faisabilité et la pertinence dans notre cas et comparer les résultats avec ceux obtenus par MCMC.

Enfin, les approches présentées ne sont pas adaptées aux images très structurées. L'utilisation du HOG ne suffit pas à caractériser pleinement les différentes classes. Il serait intéressant de s'appuyer sur un modèle bayésien de texture qui s'intégrera naturellement à notre modèle hiérarchique. Par exemple, un champ gaussien dont la matrice de covariance décrit la texture.

ANNEXE A

Preuve de l'écriture des lois sur les partitions dans le cas du DP et du HDP

A.1 Loi *a priori* sur les étiquettes z pour le processus de Dirichlet

Soit les paramètres $\boldsymbol{\vartheta} = \{\vartheta_1, \cdots, \vartheta_{\eta.}\}$ échantillonnés suivant le processus de Dirichlet $\mathbb{G} \sim \mathrm{DP}(\alpha, \Upsilon)$. Les réalisations d'un DP sont discrètes, les $\boldsymbol{\vartheta}$ ont donc des valeurs partagées nommées paramètres uniques $\boldsymbol{\vartheta}^\star = \{\vartheta_1^\star, \cdots, \vartheta_K^\star\}$ et les étiquettes $\boldsymbol{z}$ indiquant la valeur unique prise par chaque paramètre $\vartheta_n = \vartheta_{z_n}^\star, n = 1, \cdots, \eta.$.

Par définition du DP, la loi du paramètre ϑ_n s'écrit :

$$\vartheta_n | \vartheta_1, \cdots, \vartheta_{n-1}, \alpha, \Upsilon \sim \sum_{k=1}^{K} \frac{\eta_k}{\alpha + n - 1} \delta_{\vartheta_k^\star} + \frac{\alpha}{\alpha + n - 1} \Upsilon$$

et la loi de l'étiquette z_n sachant les valeurs prises par les étiquettes précédentes est :

$$\Pr(z_n | \boldsymbol{z}_{1:n-1}) = \frac{1}{\alpha + n - 1} \begin{cases} \eta_k & \text{si } k \leq K \\ \alpha & \text{si } k = k^{\mathrm{new}} = K + 1 \end{cases}$$

Soit η_1 le nombre de paramètres $\boldsymbol{\vartheta}$ prenant la première valeur unique $\vartheta_1^\star$, η_2 le nombre d'entre eux prenant la valeur $\vartheta_2^\star$, etc. Puisque la séquence $\boldsymbol{\vartheta}$ est échangeable, il peut être considéré que les η_1 premières valeurs correspondent à $\vartheta_1^\star$, les η_2 suivantes à $\vartheta_2^\star$, etc. Prenons le premier cas, la probabilité que $\vartheta_1 = \vartheta_1^\star$ est proportionnelle à α, la probabilité que la valeur suivante soit égale au même paramètre unique est proportionnelle au nombre de paramètres ayant déjà pris cette

valeur jusqu'à présent, donc 1, et ainsi de suite :

$$\begin{aligned}
p(\vartheta_1 = \vartheta_1^\star) &= \frac{\alpha}{\alpha + 1 - 1} \\
p(\vartheta_2 = \vartheta_1^\star | \vartheta_1) &= \frac{1}{\alpha + 2 - 1} \\
p(\vartheta_3 = \vartheta_1^\star | \vartheta_1, \vartheta_2) &= \frac{2}{\alpha + 3 - 1} \\
\cdots p(\vartheta_{\eta_1} = \vartheta_1^\star | \boldsymbol{\vartheta}_{1:\eta_1 - 1}) &= \frac{\eta_1 - 1}{\alpha + \eta_1 - 1} \\
\text{Il suit } p(\boldsymbol{z}_{1:\eta_1}) &= \alpha \prod_{n=1}^{\eta_1} \frac{n-1}{\alpha + n - 1}
\end{aligned}$$

Concernant la classe 2,

$$\begin{aligned}
p(\vartheta_{\eta_1+1} = \vartheta_2^\star | \boldsymbol{\vartheta}_{1:\eta_1}) &= \frac{\alpha}{\alpha + (\eta_1 + 1) - 1} \\
p(\vartheta_{\eta_1+2} = \vartheta_2^\star | \boldsymbol{\vartheta}_{1:\eta_1+1}) &= \frac{1}{\alpha + (\eta_1 + 2) - 1} \\
p(\vartheta_{\eta_1+3} = \vartheta_2^\star | \boldsymbol{\vartheta}_{1:\eta_1+2}) &= \frac{2}{\alpha + (\eta_1 + 3) - 1} \\
&\cdots \\
p(\vartheta_{\eta_1+\eta_2} = \vartheta_2^\star | \boldsymbol{\vartheta}_{1:\eta_1+\eta_2-1}) &= \frac{1}{\alpha + (\eta_1 + \eta_2) - 1} \\
\text{Il suit } p(\boldsymbol{z}_{\eta_1:\eta_1+\eta_2}) &= \alpha \prod_{n=\eta_1}^{\eta_1+\eta_2} \frac{n-1}{\alpha + n - 1}
\end{aligned}$$

Les lois sur les étiquettes associées aux paramètres prenant les valeurs uniques $\vartheta_3^\star$ à $\vartheta_K^\star$ sont construites sur le même principe, la loi jointe s'écrit alors :

$$\begin{aligned}
\Pr(\boldsymbol{z}) &= \alpha^K \left[\prod_{n=1}^{\eta_\cdot} \frac{1}{\alpha + n - 1} \right] \prod_{k=1}^{K} \underbrace{[1 \times 2 \times \cdots \times \eta_k - 1]}_{(\eta_k - 1)! = \Gamma(\eta_k)} \\
&= \alpha^K \left[\prod_{n=1}^{\eta_\cdot} \frac{1}{\alpha + n - 1} \right] \prod_{k=1}^{K} \Gamma(\eta_k)
\end{aligned}$$

De plus,

$$\prod_{n=1}^{\eta_\cdot} \frac{1}{\alpha + n - 1} = \frac{(\alpha - 1)!}{(\alpha + \eta_\cdot - 1)!} = \frac{\Gamma(\alpha)}{\Gamma(\alpha + \eta_\cdot)}$$

On retrouve donc la loi sur les étiquettes (2.10) :

$$\Pr(\boldsymbol{z} | \alpha, \boldsymbol{\eta}) = \frac{\Gamma(\alpha)}{\Gamma(\eta_\cdot + \alpha)} \alpha^K \prod_{k=1}^{K} \Gamma(\eta_k)$$

A.2 Loi *a priori* sur les étiquettes c et d pour le processus de Dirichlet hiérarchique

Soit le processus de Dirichlet hiérarchique :

$$\mathbb{G}_j \sim \mathrm{DP}(\alpha_0, \mathbb{G}_0) \qquad \mathbb{G}_0 \sim \mathrm{DP}(\gamma, H)$$
$$\theta_{jn} \sim \mathbb{G}_j,\ \psi_{jt} \sim \mathbb{G}_0\ \phi_k \sim H$$

Par construction, les paramètres $\boldsymbol{\theta}\{\theta_{jn}; j = 1, \cdots, J; n = 1, \cdots, N_j\}$ prennent leurs valeurs dans un ensemble de valeurs uniques caractéristiques du groupe j, $\boldsymbol{\psi} = \{\psi_{jt}; j = 1, \cdots, J, t = 1, \cdots, m_{j\cdot}\}$ avec $m_{j\cdot}$ le nombre de paramètres uniques dans le groupe j. $\mathbb{G}_0$ est une mesure discrète, donc les $\boldsymbol{\psi}$ sont aussi des valeurs uniques $\boldsymbol{\phi} = \{\phi_k, k = 1, \cdots, K\}$ qui se répètent, où K est le nombre de paramètres uniques parmi tous les groupes. On a :

$$\theta_{jn}|\alpha_0, \mathbb{G}_0 \sim \sum_{i=1}^{m_{j\cdot}} \frac{\nu_{jt}}{\alpha_0 + n - 1}\delta_{\psi_{jt}} + \frac{\alpha_0}{\alpha_0 + n - 1}\mathbb{G}_0$$
$$\psi_{jt}|\gamma, H, \boldsymbol{\psi}_{11:m_{1\cdot}}, \cdots, \boldsymbol{\psi}_{j1:t-1}, \cdots, \boldsymbol{\psi}_{J1:m_{J\cdot}} \sim \sum_{k=1}^{K} \frac{m_{\cdot k}}{\gamma + m_{\cdot\cdot} - 1}\delta_{\phi_k} + \frac{\gamma}{\gamma + m_{\cdot\cdot} - 1}H$$

avec ν_{jt} le nombre de $\theta_{jn} = \psi_{jt}$, $m_{\cdot k}$ le nombre de paramètres ψ_{jt} ayant pris la valeur ϕ_k.

Pour chaque groupe, la loi sur les étiquettes $\boldsymbol{c}_j$ s'écrit comme dans la partie précédente :

$$\Pr(\boldsymbol{c}_j|\alpha_0, \boldsymbol{\nu}_j) = \frac{\Gamma(\alpha_0)}{N_j + \alpha_0)}\alpha_0^{m_{j\cdot}} \prod_{t=1}^{m_{j\cdot}} \Gamma(\nu_{jt})$$

Basé sur le même principe, la loi des étiquettes $\boldsymbol{d}$ s'écrit :

$$\Pr(\boldsymbol{d}|\boldsymbol{c}, \gamma, \boldsymbol{m}) = \gamma^K \left[\prod_{k=1}^{K} \Gamma(m_{\cdot k})\right] \frac{\Gamma(\gamma)}{\Gamma(\gamma + m_{\cdot\cdot})}$$

Il s'ensuit que la loi sur l'ensemble des étiquettes (2.27) est :

$$\varphi(\boldsymbol{c}, \boldsymbol{d}) = \prod_{j=1}^{J} \left\{ \left[\frac{\Gamma(\alpha_0)}{\Gamma(N_j + \alpha_0)}\right] \alpha_0^{m_{j\cdot}} \left[\prod_{i=1}^{m_{j\cdot}} \Gamma(\nu_{jt})\right] \right\} \frac{\Gamma(\gamma)}{\Gamma(m_{\cdot\cdot} + \gamma)} \gamma^K \left[\prod_{k=1}^{K} \Gamma(m_{\cdot k})\right]$$

ANNEXE B

Démonstrations pour l'échantillonnage *a priori* de la partition pour le GSW dans le cas d'un champ de Potts

Soit à démontrer l'équation (3.8).

$$\begin{aligned}\Pr(\boldsymbol{z}_{C_l} = k|\boldsymbol{r}, \boldsymbol{z}^{-l}) &= \frac{p(\boldsymbol{z}_{C_l} = k, \boldsymbol{r}, \boldsymbol{x}, \boldsymbol{z}^{-l})}{p(\boldsymbol{r}, \boldsymbol{z}^{-l})} \\ &= \frac{\Pr(\boldsymbol{r}|\boldsymbol{z}_{C_l} = k, \boldsymbol{z}^{-l})}{\Pr(\boldsymbol{r}|\boldsymbol{z}^{-l})} \quad \frac{\Pr(\boldsymbol{z}_{C_l} = k, \boldsymbol{z}^{-l})}{\Pr(\boldsymbol{z}^{-l})}\end{aligned}$$

La distribution des liens est donnée à l'équation (3.7) et s'écrit :

$$\begin{aligned}\Pr(\boldsymbol{r}|\boldsymbol{z}_l = k, \boldsymbol{z}^{-jt}) &= \prod_{n=1, n\sim q}^{N} \exp\left(-\beta\lambda \mathbf{1}_{z_q, z_n}\right)^{1-r_{nq}} \left(1 - \exp\left(-\beta\lambda \mathbf{1}_{z_q, z_n}\right)\right)^{r_{nq}} \\ &= \prod_{n=1, n\sim q}^{N} \exp\left(-\beta\lambda \mathbf{1}_{z_q, z_n}\right)^{1-r_{nq}} \exp\left(-\beta\lambda \mathbf{1}_{z_q, z_n}\right)^{r_{nq}} \\ &\qquad \left(\exp\left(\beta\lambda \mathbf{1}_{z_q, z_n} - 1\right)\right)^{r_{nq}}\end{aligned}$$

$$\Pr(\boldsymbol{r}|\boldsymbol{z}_l = k, \boldsymbol{z}^{-jt}) = \prod_{n=1, q\in\mathcal{V}_n}^{N} \exp\left(-\beta\lambda \mathbf{1}_{z_q, z_n}\right) \left(\exp\left(\beta\lambda \mathbf{1}_{z_q, z_n} - 1\right)\right)^{r_{nq}}$$

On en déduit que le rapport des distributions de liens s'écrit :

$$\frac{\Pr(\boldsymbol{r}|\boldsymbol{z}_{C_l}=k,\boldsymbol{z}^{-l})}{\Pr(\boldsymbol{r}^{-l}|\boldsymbol{z}^{-l})} \propto \exp\left(-\sum_{\substack{q\in\mathcal{V}_n \\ n\in C_l, q\notin C_l}} \beta\lambda)\mathbf{1}_{z_q,z_n}\right)$$

Le rapport des distributions d'étiquettes donne :

$$\frac{\Pr(\boldsymbol{z}_l=k,\boldsymbol{z}^{-l})}{\Pr(\boldsymbol{z}^{-l})} \propto \exp\left(\sum_{\substack{q\in\mathcal{V}_n \\ n\in C_l, q\notin C_l}} \beta\mathbf{1}_{z_q,z_n}\right)$$

Il suit :

$$\Pr(\boldsymbol{z}_l=k|\boldsymbol{r},\boldsymbol{z}^{-l}) \propto \exp\left(\sum_{\substack{q\in\mathcal{V}_n \\ n\in C_l, q\notin C_l}} \beta(1-\lambda)\mathbf{1}_{z_q,z_n}\right)$$

ANNEXE C

Démonstrations pour l'échantillonnage des étiquettes pour le modèle HDP-Potts

C.1 Equations d'échantillonnage des étiquettes de classe

Soit à démontrer les équations (4.7) d'échantillonnage de l'étiquette de classe associée à la région t dans l'image j conditionnellement aux autres étiquettes de classe, aux étiquettes de région et aux observations. La règle de Bayes stipule que :

$$\begin{aligned}\Pr(d_{jt}=k|\boldsymbol{c},\boldsymbol{d}^{-jt},\boldsymbol{y}) &= \frac{p(\boldsymbol{c},d_{jt}=k,\boldsymbol{d}^{-jt},\boldsymbol{y})}{p(\boldsymbol{c},\boldsymbol{d}^{-jt},\boldsymbol{y})} \\ &= \frac{p(\boldsymbol{y}|\boldsymbol{c},d_{jt}=k,\boldsymbol{d}^{-jt})}{p(\boldsymbol{y}|\boldsymbol{c},\boldsymbol{d}^{-jt})} \quad \frac{\Pr(\boldsymbol{c},d_{jt}=k,\boldsymbol{d}^{-jt})}{\Pr(\boldsymbol{c},\boldsymbol{d}^{-jt})}\end{aligned}$$

avec $\boldsymbol{d}^{-jt}$ l'ensemble des étiquettes de classe sauf d_{jt}. Nous avons $p(\boldsymbol{y}|\boldsymbol{c},\boldsymbol{d}^{-jt}) = p(\boldsymbol{y}^{-jt}|\boldsymbol{c},\boldsymbol{d}^{-jt})$ avec $\boldsymbol{y}^{-jt}$ l'ensemble des observations en ôtant $\boldsymbol{y}_{jt}$ celles attachées aux pixels dans la région t. En effet, même si $\boldsymbol{y}_{jt}$ est connu, la classe associée est inexistante, ce qui ne permet donc pas d'avoir des informations statistiques sur l'ensemble. Le premier rapport s'écrit alors :

$$\frac{p(\boldsymbol{y}|\boldsymbol{c},d_{jt}=k,\boldsymbol{d}^{-jt})}{p(\boldsymbol{y}|\boldsymbol{c},\boldsymbol{d}^{-jt})} = \frac{p(\boldsymbol{y}|\boldsymbol{c},d_{jt}=k,\boldsymbol{d}^{-jt})}{p(\boldsymbol{y}^{-jt}|\boldsymbol{c},\boldsymbol{d}^{-jt})} = \left[\prod_{l=1,l\neq k}^{K}\frac{f(\boldsymbol{y}_{A_l}^{-jt})}{f(\boldsymbol{y}_{A_l}^{-jt})}\right]\frac{f(\boldsymbol{y}_{jt},\boldsymbol{y}_{A_k}^{-jn})}{f(\boldsymbol{y}_{A_k}^{-jn})} = \frac{f(\boldsymbol{y}_{jt},\boldsymbol{y}_{A_k}^{-jt})}{f(\boldsymbol{y}_{A_k}^{-jt})}$$

avec ${A_k}^{-jt}$ l'ensemble des pixels dans la classe k sauf ceux de la région t de l'image j. A partir de la loi jointe (4.5), la loi associée à l'ensemble des étiquettes sauf d_{jt} s'écrit :

$$\begin{aligned}\Pr(\boldsymbol{c}, \boldsymbol{d}^{-jt}) = &\prod_{\substack{i=1\\ i\neq j}}^{J}\left[\frac{\Gamma(\alpha_0)}{\Gamma(N_i+\alpha_0)}\alpha_0^{m_{i\cdot}}\prod_{t=1}^{m_{i\cdot}}\Gamma(\nu_{it})\right]\left[\frac{\Gamma(\alpha_0)}{\Gamma(N_j^{-jt}+\alpha_0)}\alpha_0^{m_{j\cdot}^{-jt}}\prod_{\ell=1}^{m_{j\cdot}^{-jt}}\Gamma(\nu_{jl})\right]\\ &\frac{\Gamma(\gamma)}{\Gamma(m_{\cdot\cdot}^{-jt}+\gamma)}\gamma^{K^{-jt}}\prod_{k=1}^{K^{-jt}}\Gamma(m_{\cdot k}^{-jt})\\ &\left[\prod_{i=1,i\neq j}^{J}\prod_{n=1}^{N_i}\exp\left(\sum_{q\in\mathcal{V}_\ell}\beta\mathbf{1}_{d_{ic_{iq}},d_{ic_{i\ell}}}\right)\right]\prod_{\ell=1,c_{jl}\neq t}^{N_j}\exp\left(\sum_{q\in\mathcal{V}_\ell}\beta\mathbf{1}_{d_{jc_{jq}},d_{jc_{j\ell}}}\right)\end{aligned} \tag{C.1}$$

où

- $N_j^{-jt} = N_j - \nu_{jt}$ le nombre de pixels dans l'image j en ne comptant pas celles de la région t
- $m_{j\cdot}^{-jt} = m_{j\cdot} - 1$ puisque la région t est retirée, de même
- $m_{\cdot\cdot}^{-jt} = m_{\cdot\cdot} - 1$
- K^{-jt} le nombre de classes en ôtant la région t de l'image j
- $m_{\cdot k}^{-jt}$ le nombre de régions affectées à la classe k sauf la région t de l'image j

Il suit que :

$$\begin{aligned}\frac{\Pr(\boldsymbol{c}, d_{jt}=k, \boldsymbol{d}^{-jt})}{\Pr(\boldsymbol{c}, \boldsymbol{d}^{-jt})} = &\left[\prod_{n=N_j^{-jt}+1}^{N_j}\frac{1}{\alpha_0+n-1}\right]\alpha_0^{m_{j\cdot}-m_{j\cdot}^{-jt}}\Gamma(\nu_{jt})\\ &\frac{1}{\gamma+m_{\cdot\cdot}-1}\,\gamma^{K-K^{-jt}}\frac{\Gamma(m_{\cdot k})}{\Gamma(m_{\cdot k}^{-jt})}\\ &\prod_{n,q|c_{jn}=t,c_{jq}\neq t}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq}},k}\right)\end{aligned}$$

Les cas possibles sont :

- $m_{\cdot k} = m_{\cdot k}^{-jt} + 1$, et sachant que $\Gamma(a) = (a-1)!$ avec $a!$ le factoriel du nombre a,la probabilité s'écrit :

$$\Pr(d_{jt}=k|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \propto m_{\cdot}^{-jt}\prod_{n,q|c_{jn}=t,c_{jq}\neq t}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq}},k}\right)$$

- $K = K^{-jt} + 1$, c'est-à-dire la région est dans une nouvelle classe, donc $m_{\cdot k^{\text{new}}} = 1$, il n'y a de plus aucun pixel déjà affecté à cette classe, la probabilité s'écrit alors :

$$\Pr(d_{jt}=k^{\text{new}}|\boldsymbol{c}, \boldsymbol{d}^{-jt}) \propto \gamma$$

On en déduit les équations conditionnelles *a posteriori* de l'étiquette de classe d_{jt} (4.7) :

$$\Pr(d_{jt}=k|\boldsymbol{c},\boldsymbol{d}^{-jt},\boldsymbol{y}) \propto \begin{cases} m_{\cdot k}^{-jt} \exp\left(\sum_{n\sim q}\beta \mathbf{1}_{d_{jc_{jq}},k}\right) f(\boldsymbol{y}_{jt}|\boldsymbol{y}_{A_k^{-jt}}) & \text{si } k \leq K \\ \gamma f(\boldsymbol{y}_{jt}) & \text{si } k = k^{\text{new}} \end{cases}$$

C.2 Equations d'échantillonnage des étiquettes de région

Soit à démontrer les équations (4.8) d'échantillonnage de l'étiquette de région affectée au pixel n dans l'image j conditionnellement aux autres étiquettes de région, aux étiquettes de classe et aux observations. Selon la règle de Bayes, il peut être écrit :

$$\begin{aligned}\Pr(c_{jn}=t|\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}) &= \frac{p(c_{jn}=t,\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y})}{p(\boldsymbol{y},\boldsymbol{c}^{-jn},\boldsymbol{d})} \\ &= \frac{p(\boldsymbol{y}|c_{jn}=t,\boldsymbol{c}^{-jn},\boldsymbol{d})}{p(\boldsymbol{y}|\boldsymbol{c}^{-jn},\boldsymbol{d})} \quad \frac{\Pr(c_{jn}=t,\boldsymbol{c}^{-jn},\boldsymbol{d})}{\Pr(\boldsymbol{c}^{-jn},\boldsymbol{d})}\end{aligned} \tag{C.2}$$

Comme précédemment, $p(\boldsymbol{y}|\boldsymbol{c}^{-jn},\boldsymbol{d}) = p(\boldsymbol{y}^{-jn}|\boldsymbol{c}^{-jn},\boldsymbol{d})$ car y_{jn} est certes observé mais aucune information relative n'est disponible. En outre, la distribution des données conditionnellement à la partition est donnée par l'équation (2.29). Il résulte que la première fraction s'écrit :

$$\frac{p(\boldsymbol{y}|c_{jn}=t,\boldsymbol{c}^{-jn},\boldsymbol{d})}{p(\boldsymbol{y}|\boldsymbol{c}^{-jn},\boldsymbol{d})} = \left[\prod_{k=1,k\neq d_{jt}}^{K} \frac{f(\boldsymbol{y}_{A_k}^{-jn})}{f(\boldsymbol{y}_{A_k}^{-jn})}\right] \frac{f(y_{jn},\boldsymbol{y}_{A_k}^{-jn})}{f(\boldsymbol{y}_{A_k}^{-jn})} = \frac{f(y_{jn},\boldsymbol{y}_{A_k}^{-jn})}{f(\boldsymbol{y}_{A_k}^{-jn})}$$

Soit la loi sur les étiquettes donnée par (4.5). Si l'étiquette c_{jn} en est ôtée, il reste :

$$\begin{aligned}\Pr(\boldsymbol{c}^{-jn},\boldsymbol{d}) = &\prod_{\substack{i=1\\i\neq j}}^{J}\left[\frac{\Gamma(\alpha_0)}{\Gamma(N_i+\alpha_0)}\alpha_0^{m_{i\cdot}}\prod_{t=1}^{m_{i\cdot}}\Gamma(\nu_{it})\right]\left[\frac{\Gamma(\alpha_0)}{\Gamma(N_j^{-jn}+\alpha_0)}\alpha_0^{m_{j\cdot}^{-jn}}\prod_{t=1}^{m_{j\cdot}^{-jn}}\Gamma(\nu_{jt}^{-jn})\right] \\ &\frac{\Gamma(\gamma)}{\Gamma(m_{\cdot\cdot}^{-jn}+\gamma)}\gamma^{K^{-jn}}\prod_{k=1}^{K^{-jn}}\Gamma(m_{\cdot k}^{-jn}) \\ &\left[\prod_{i=1}^{J}\prod_{\ell=1}^{N_i}\exp\left(\sum_{q\in\mathcal{V}_\ell}\beta\delta(d_{ic_{iq}},d_{ic_{i\ell}})\right)\right]\prod_{\ell=1,l\neq n}^{N_j}\exp\left(\sum_{q\in\mathcal{V}_\ell}\beta\mathbf{1}_{d_{jc_{jq}},d_{jc_{j\ell}}}\right)\end{aligned} \tag{C.3}$$

avec

- $N_j^{-jn} = N_j - 1$ le nombre de pixels dans l'image j en ôtant le n-ième
- $m_{j\cdot}^{-jn}$ le nombre de régions restant dans l'image j lorsque le pixel n est enlevé de la partition
- ν_{jt}^{-jn} le nombre de pixels dans la région t de l'image j lorsque le pixel n n'est pas pris en compte

– K^{-jn} le nombre de classes sans le pixel n de l'image j

Il suit que :

$$\begin{aligned}\frac{\Pr(c_{jn}=t,\boldsymbol{c}^{-jn},\boldsymbol{d})}{\Pr(\boldsymbol{c}^{-jn},\boldsymbol{d})} =& \frac{1}{\alpha_0+N_j-1}\alpha_0^{m_{j\cdot}-m_{j\cdot}^{-jn}}\frac{\Gamma(\nu_{jt})}{\Gamma(\nu_{jt}^{-jn})}\\ & \frac{1}{\gamma+m_{\cdot\cdot}-1}\gamma^{K-K^{-jn}}\frac{\Gamma(m_{\cdot k})}{\Gamma(m_{\cdot k}^{-jn})}\\ & \exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq}},d_{jc_{jn}}}\right)\end{aligned} \tag{C.4}$$

Les différents cas possibles sont :

– $m_{j\cdot}=m_{j\cdot}^{-jn}+1$, alors le pixel est affecté à une nouvelle région $t=t^{\text{new}}=m_{j\cdot}$, $\nu_{jt}=\nu_{jt^{\text{new}}}=1$, on a

$$\Pr(c_{jn}=t^{\text{new}}|\boldsymbol{c}^{-jn},\boldsymbol{d})\propto\alpha_0$$

et ces configurations sont possibles pour l'étiquette de classe $d_{jt^{\text{new}}}$ affectée à cette nouvelle région :

– $\nu_{jt}=\nu_{jt}^{-jn}+1$, alors le pixel n est dans une région existante t : $m_{j\cdot}=m_{j\cdot}^{-jn}$, $K=K^{-jn}$, $m_{\cdot k}=m_{\cdot k}^{-jn}$ et la probabilité conditionnelle s'écrit :

$$\Pr(c_{jn}=t|\boldsymbol{c}^{-jn},\boldsymbol{d})\propto\nu_{jt}^{-jn}\exp\left(\sum_{q\in\mathcal{V}_n}\mathbf{1}_{d_{jc_{jq}},d_{jt}}\right)$$

- $m_{\cdot k}=m_{\cdot k}^{-jn}+1$, la nouvelle région est alors associée à une classe existante :

$$\Pr(d_{jt^{\text{new}}}=k|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d})\propto m_{\cdot k}^{-jn}$$

- $K=K^{-jn}+1$, le pixel est donc affecté aussi à une nouvelle classe, $d_{jt^{\text{new}}}=k^{\text{new}}=K^{-jn}+1$, $m_{\cdot k^{\text{new}}}=1$, dans ce cas, aucun pixel n'a déjà été affecté à cette classe donc $\sum_{n\sim q}\mathbf{1}_{d_{jc_{jq}},d_{jt^{\text{new}}}}=0$ la probabilité conditionnelle s'écrit alors :

$$\Pr(d_{jt^{\text{new}}}=k^{\text{new}}|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d})\propto\gamma$$

La probabilité de l'étiquette c_{jn} conditionnellement aux autres étiquettes de région, aux étiquettes de classe et aux observations s'écrit alors :

$$\Pr(c_{jn}=t|\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y})\propto\begin{cases}\nu_{jt}^{-jn}\exp\left(\sum_{q\in\mathcal{V}_n}\beta\mathbf{1}_{d_{jc_{jq}},d_{jc_{jn}}}\right)f(y_{jn}|\boldsymbol{y}_{A_k^{-jn}}) & \text{si } t\leq m_{j\cdot}\\ \alpha_0\,p(y_{jn}|c_{jn}=t^{\text{new}},\boldsymbol{c}^{-jn},\boldsymbol{d},\boldsymbol{y}^{-jn}) & \text{si } t=t^{\text{new}}\end{cases}$$

La probabilité de l'observation y_{jn} sachant qu'elle est affectée à une nouvelle région (4.9) est développée est obtenue en sommant sur toutes les possibilités de classes :

$$\begin{aligned} p(y_{jn}|c_{jn} = t^{\text{new}}, \boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}^{-jn}) &\propto \Pr(d_{jt^{\text{new}}} = k|c_{jn} = t^{\text{new}}, \boldsymbol{c}^{-jn}, \boldsymbol{d}^{-jt^{\text{new}}}) \\ &\quad p(y_{jn}|c_{jn} = t^{\text{new}}, \boldsymbol{c}^{-jn}, \boldsymbol{d}, \boldsymbol{y}^{-jn}) \\ &\propto \left\{ \sum_{k=1}^{K} m_{\cdot k} \exp\left(\sum_{n \sim q} \beta \mathbf{1}_{d_{jc_{jq}}, k} \right) + \gamma \right\}^{-1} \\ &\quad \left\{ \sum_{k=1}^{K} m_{\cdot k} \exp\left(\sum_{n \sim q} \beta \mathbf{1}_{d_{jc_{jq}}, k} \right) f\left(y_{jn} | \boldsymbol{y}_{A_k^{-jn}} \right) + \gamma f(y_{jn}) \right\} \end{aligned} \tag{C.5}$$

ANNEXE D

Démonstrations pour l'échantillonnage avec le GSW pour le modèle HDP-Potts

Soit à démontrer les équations (4.13) et (4.14). Le théorème de Bayes permet d'écrire :

$$\begin{aligned}\Pr(\boldsymbol{c}_{jl}=t|\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r},\boldsymbol{y}) &= \frac{p(\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r},\boldsymbol{y})}{p(\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r},\boldsymbol{y})} \\ &= \frac{p(\boldsymbol{y}|\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r})}{p(\boldsymbol{y}|\boldsymbol{c}^{-jl},\boldsymbol{d},\boldsymbol{r})} \quad \frac{\Pr(\boldsymbol{r}|\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d})}{\Pr(\boldsymbol{r}|\boldsymbol{c}^{-jl},\boldsymbol{d})} \quad \frac{\Pr(\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d})}{\Pr(\boldsymbol{c}^{-jl},\boldsymbol{d})} \\ &= \frac{f(\boldsymbol{y}_{jl},\boldsymbol{y}_{A_k^{-jl}})}{f(\boldsymbol{y}_{A_k^{-jl}})} \quad \frac{\Pr(\boldsymbol{r}|\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d})}{\Pr(\boldsymbol{r}|\boldsymbol{c}^{-jl},\boldsymbol{d})} \quad \frac{\Pr(\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jl},\boldsymbol{d})}{\Pr(\boldsymbol{c}^{-jl},\boldsymbol{d})}\end{aligned}$$

avec $\boldsymbol{c}_{jl}$ l'ensemble des étiquettes de régions des pixels dans le spin-cluster l de l'image j, $\boldsymbol{y}_{jl}$ les observations associées, $\boldsymbol{c}^{-jl}$ l'ensemble des étiquettes de région sauf $\boldsymbol{c}_{jl}$, A_k^{-jl} l'ensemble des pixels dans la classe k en ne considérant pas les pixels du spin-cluster l de l'image j.

La distribution des liens est indépendante des étiquettes de classe et est donnée par (4.12) :

$$\begin{aligned}\Pr(\boldsymbol{r}|\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jt}) &= \prod_{j=1}^{J}\prod_{n=1,n\sim q}^{N_j} \exp\left(-\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}\right)^{1-r_{jnq}}\left(1-\exp\left(-\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}\right)\right)^{r_{jnq}} \\ &= \prod_{j=1}^{J}\prod_{n=1,n\sim q}^{N_j} \exp\left(-\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}\right)^{1-r_{jnq}} \exp\left(-\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}\right)^{r_{jnq}} \\ &\qquad \left(\exp\left(\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}-1\right)\right)^{r_{jnq}}\end{aligned}$$

$$\Pr(\boldsymbol{r}|\boldsymbol{c}_{jl}=t,\boldsymbol{c}^{-jt}) = \prod_{j=1}^{J}\prod_{n=1,n\sim q}^{N_j} \exp\left(-\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}\right)\left(\exp\left(\beta\lambda\mathbf{1}_{c_{jn},c_{jq}}-1\right)\right)^{r_{jnq}}$$

On a l'équivalence $\Pr(\boldsymbol{r}|\boldsymbol{c}^{-jl}, \boldsymbol{d}) = \Pr(\boldsymbol{r}^{-jl}|\boldsymbol{c}^{-jl}, \boldsymbol{d})$.

Et le rapport des probabilités de liens s'écrit :

$$\frac{\Pr(\boldsymbol{r}|\boldsymbol{c}_{jl} = t, \boldsymbol{c}^{-jl}, \boldsymbol{d})}{\Pr(\boldsymbol{r}^{-jl}|\boldsymbol{c}^{-jl}, \boldsymbol{d})} \propto \exp\left(-\sum_{q \in \mathcal{V}_{C_{jl}}} \beta\lambda \mathbf{1}_{c_{jq},t}\right)$$

où C_{jl} l'ensemble des pixels dans le spin-cluster l de l'image j et $\mathcal{V}_{C_{jl}} = \{q | n \in C_{jl}, q \in \mathcal{V}_n, r_{j_{nq}} = 0\}$ l'ensemble des pixels voisins des pixels dans C_{jl}.

En s'inspirant de la procédure à l'annexe C et le détail à la section 4.3,

$$\Pr(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d}) \propto \begin{cases} \dfrac{\Gamma(\nu_{jt}^{-jl} + |C_{jl}|)}{\Gamma(\nu_{jt}^{-jl})} \exp\left(-\sum\limits_{q \in \mathcal{V}_{C_{jl}}} \beta\lambda \mathbf{1}_{c_{jq},t}\right) & \text{si } t \leq m_{j\cdot}^{-jl} \\ \alpha_0 \Gamma(|C_{jl}|) & \text{si } t = t^{\text{new}} = m_{j\cdot}^{-jl} + 1 \end{cases}$$

Il s'ensuit que :

$$\Pr(\boldsymbol{c}_{jl} = t|\boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}) \propto \begin{cases} \dfrac{\Gamma(\nu_{jt}^{-jl} + |C_{jl}|)}{\Gamma(\nu_{jt}^{-jl})} \exp\left(\sum\limits_{q \in \mathcal{V}_{C_{jl}}} \beta \left[\mathbf{1}_{d_{jc_{jq}}, d_{jt}} - \lambda \mathbf{1}_{c_{jq},t}\right]\right) f(\boldsymbol{y}_{jl}|\boldsymbol{y}_{A_{d_{jt}}^{-jl}}) & \text{si } t \leq m_{j\cdot}^{-jl} \\ \alpha_0 \Gamma(|C_{jl}|) p(\boldsymbol{y}_{jl}|\boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl}) & \text{si } t = t^{\text{new}} \end{cases}$$

$p(\boldsymbol{y}_{jl}|\boldsymbol{c}_{jl} = t^{\text{new}}, \boldsymbol{c}^{-jl}, \boldsymbol{d}, \boldsymbol{y}^{-jl})$ est donnée comme à l'annexe C par la sommation sur toutes valeurs possibles de classe pour la nouvelle région t^{new}.

ANNEXE E

Résultats complémentaires

E.1 Résultats de segmentation d'un ensemble d'images-test avec une initialisation aléatoire

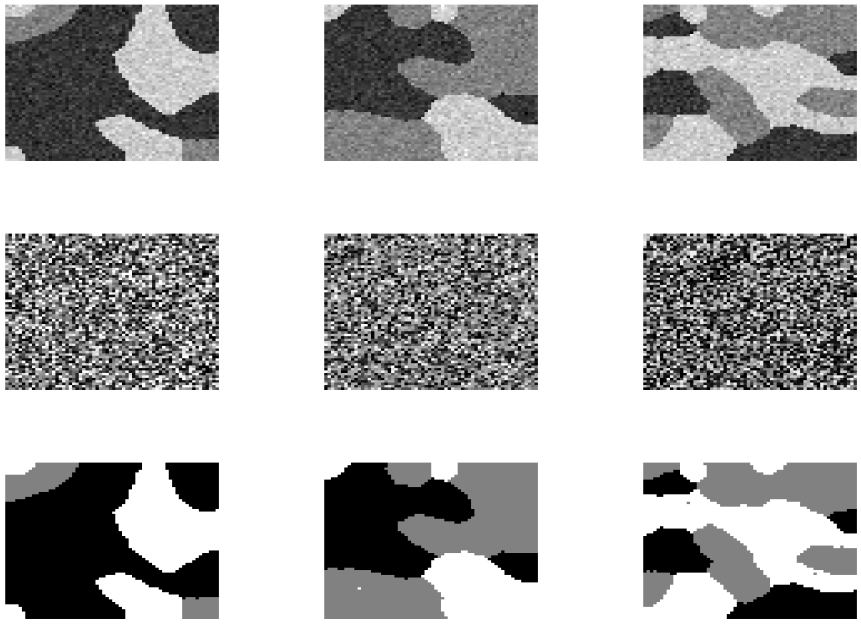


FIGURE E.1 – Images bruitées avec un niveau de bruit $\sigma_y = 5$, initialisation aléatoire et résultats de segmentation obtenus pour le HDP-Potts avec $K_{\text{init}} = 15$.

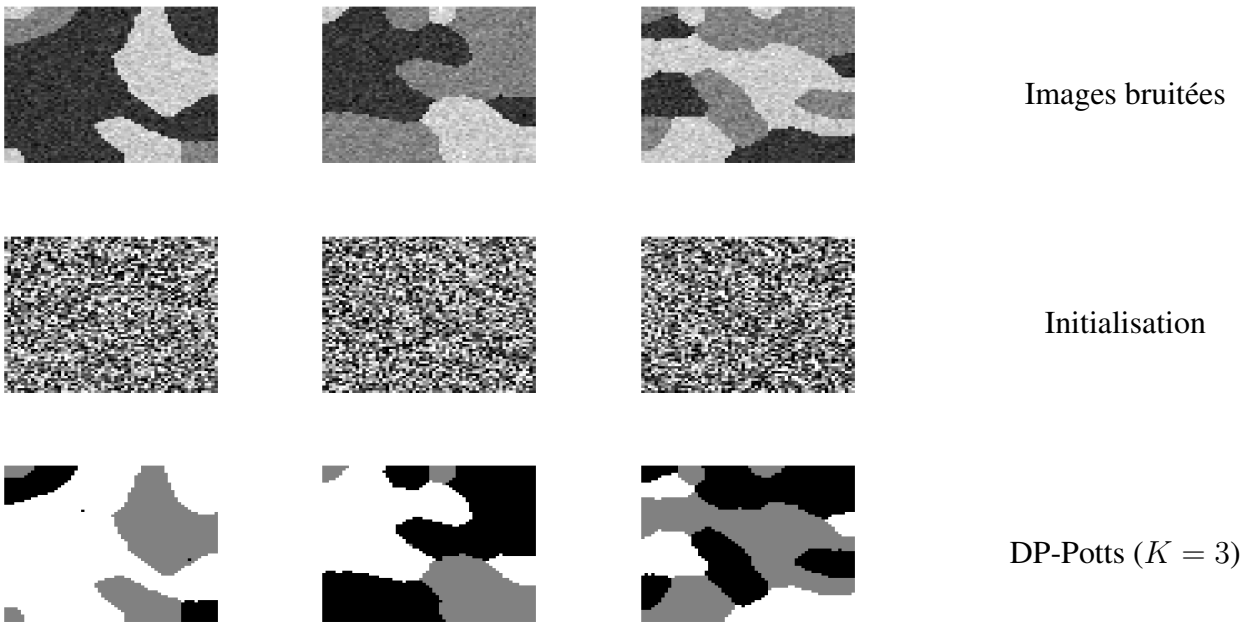


FIGURE E.2 – Images bruitées avec un niveau de bruit $\sigma_y = 5$, initialisation et résultats de segmentation obtenus pour le DP-Potts avec $K_{\text{init}} = 15$.

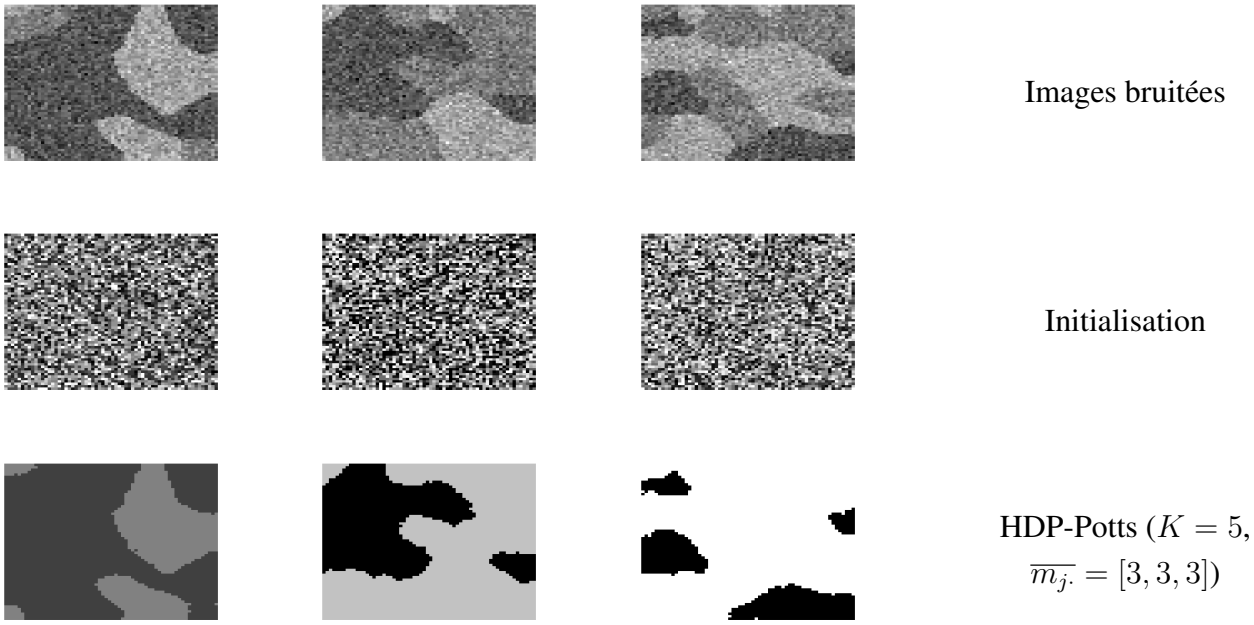


FIGURE E.3 – Images bruitées avec un niveau de bruit $\sigma_y = 15$, initialisation aléatoire et résultats de segmentation obtenus avec $\beta = 1.2$ pour le HDP-Potts avec $K_{\text{init}} = 15$.

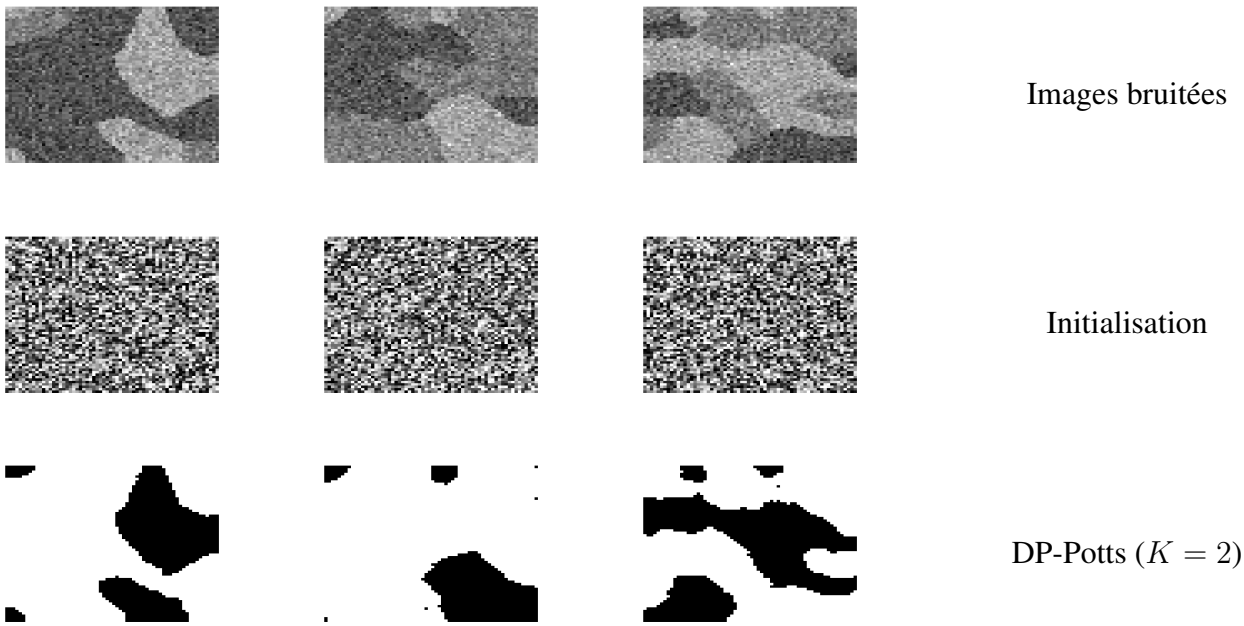


FIGURE E.4 – Images bruitées avec un niveau de bruit $\sigma_y = 15$, initialisation aléatoire et résultats de segmentation obtenus avec pour le DP-Potts avec $K_{\text{init}} = 15$.

E.2 Résultats de segmentation pour un ensemble de 20 images avec moyenne partagée entre régions de classes

Les sections E.2 et E.3 présentent des résultats obtenus pour un ensemble de 20 images de 64×64 pixels divisées en approximativement 1000 super-pixels chacune par la méthode SLIC. Ces images, constantes par morceaux, sont divisées en 3 classes dont une peu représentée. L'observation associée à chaque super-pixel est choisie comme la moyenne des observations des pixels composant le super-pixel. On considère par la suite que cette observation suit une loi normale, revenant au même modèle que précédemment. Deux jeux de données ont été définis : un premier où une unique moyenne est affectée aux régions composant une classe et un deuxième où les moyennes sont égales au paramètre de la classe avec une marge ± 0.5. Pour ces deux jeux de données les modèles DP-Potts et HDP-Potts donnent de bons résultats pour $\sigma_y = 5$ mais les deux modèles échouent à retrouver la classe sous-représentée lorsque le niveau de bruit augmente.

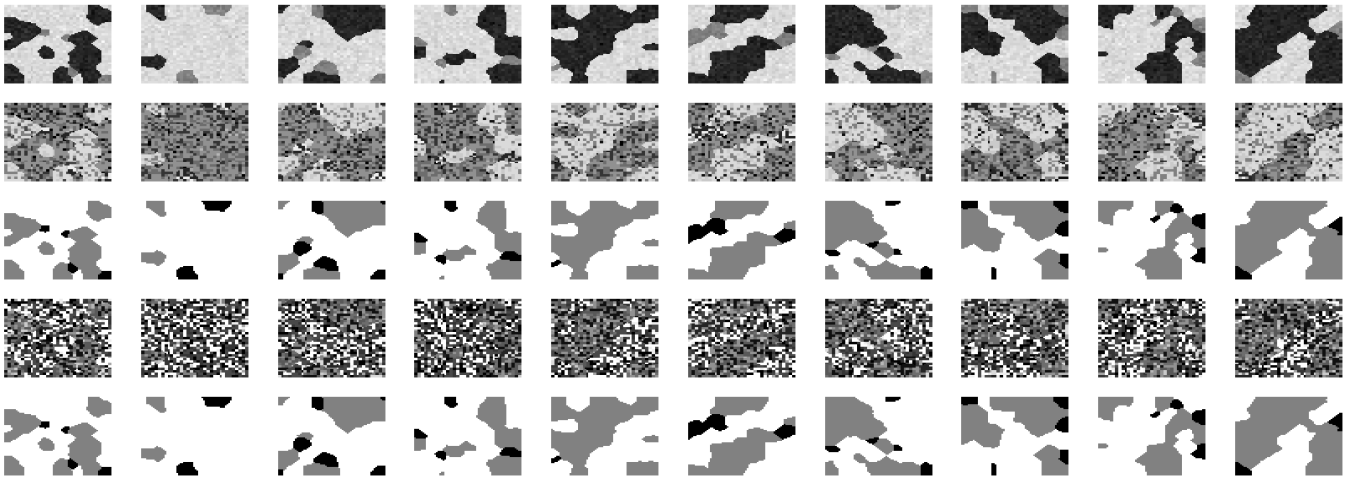


FIGURE E.5 – Images (1 à 10) bruitées avec un niveau de bruit $\sigma_y = 5$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

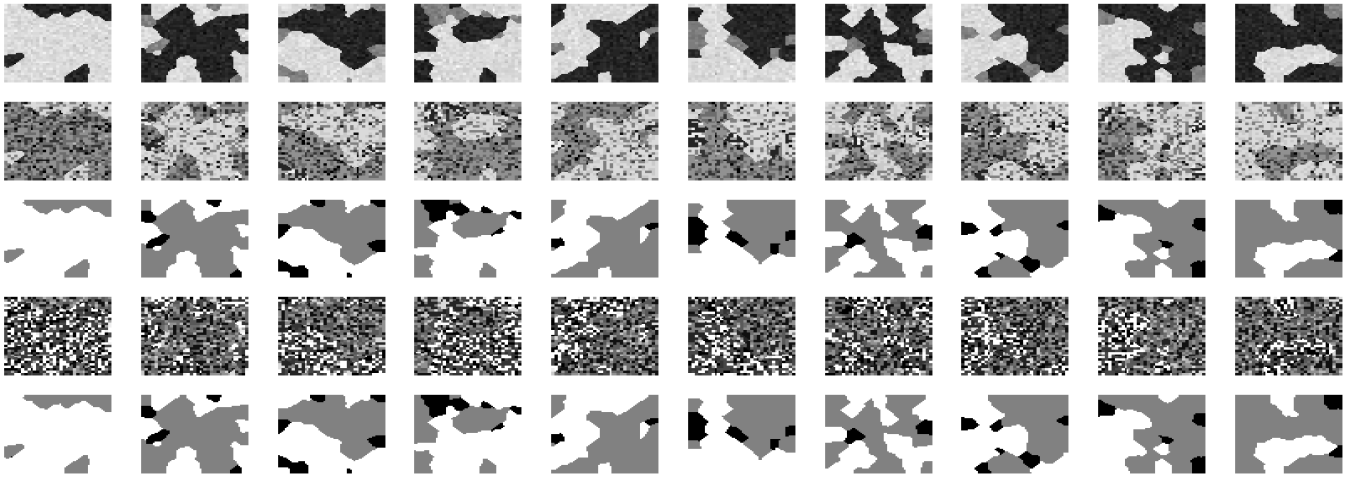


FIGURE E.6 – Images (11 à 20) bruitées avec un niveau de bruit $\sigma_y = 5$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

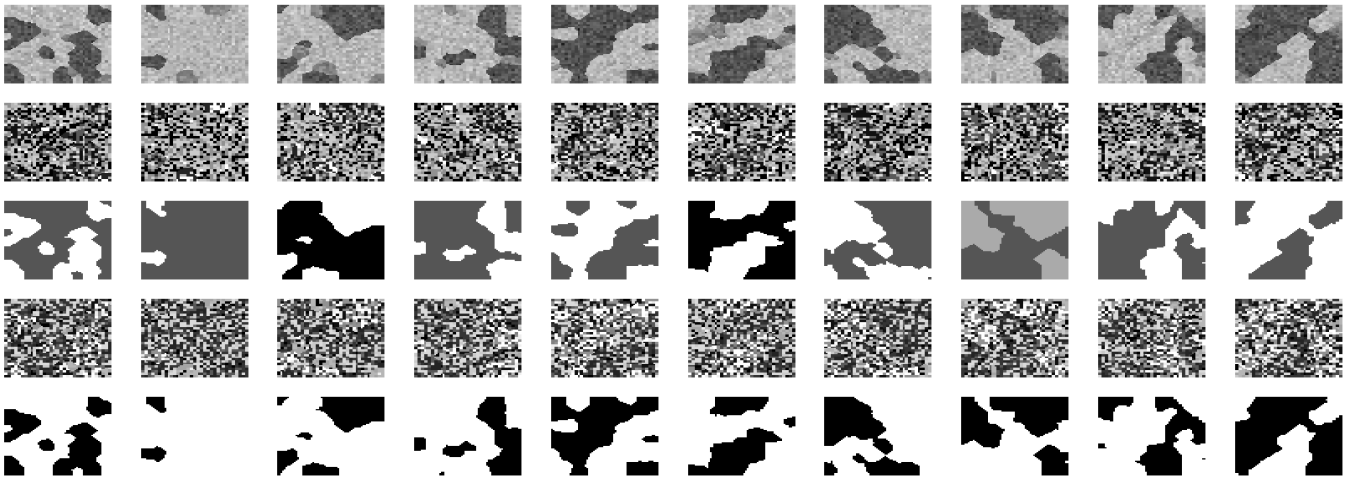


FIGURE E.7 – Images (1 à 10) bruitées avec un niveau de bruit $\sigma_y = 15$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

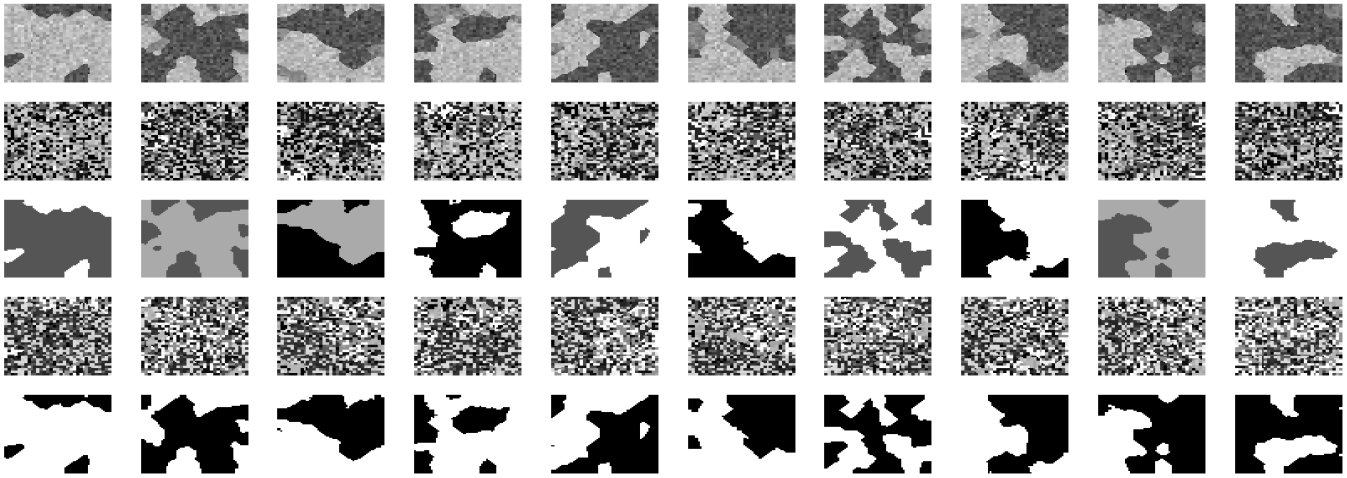


FIGURE E.8 – Images (11 à 20) bruitées avec un niveau de bruit $\sigma_y = 15$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

E.3 Résultats de segmentation pour un ensemble de 20 images avec moyennes différentes entre régions de classes

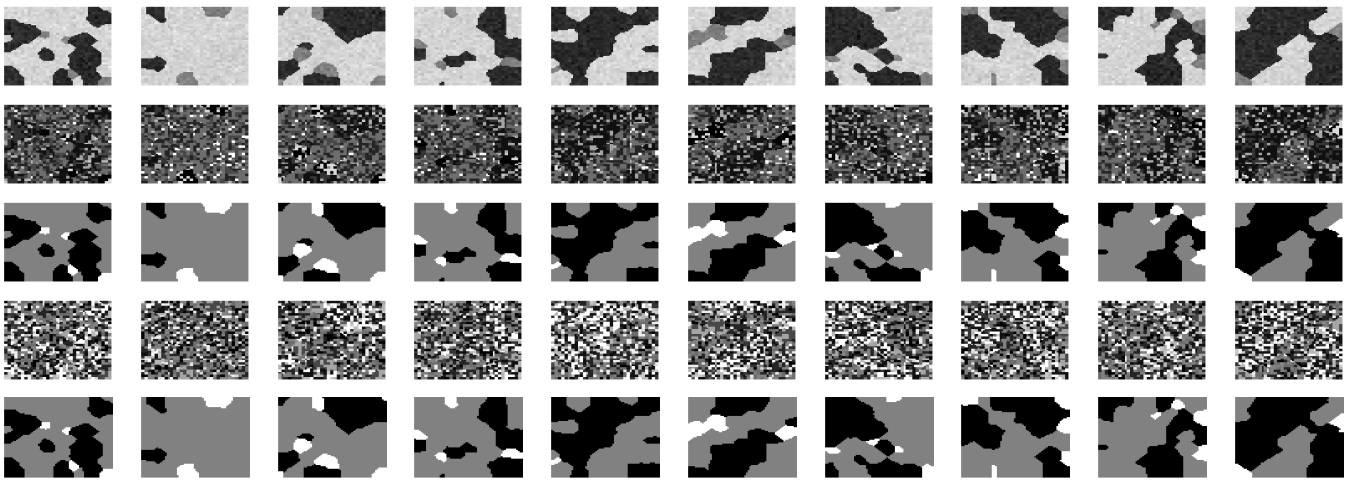


FIGURE E.9 – Images (1 à 10) bruitées avec un niveau de bruit $\sigma_y = 5$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

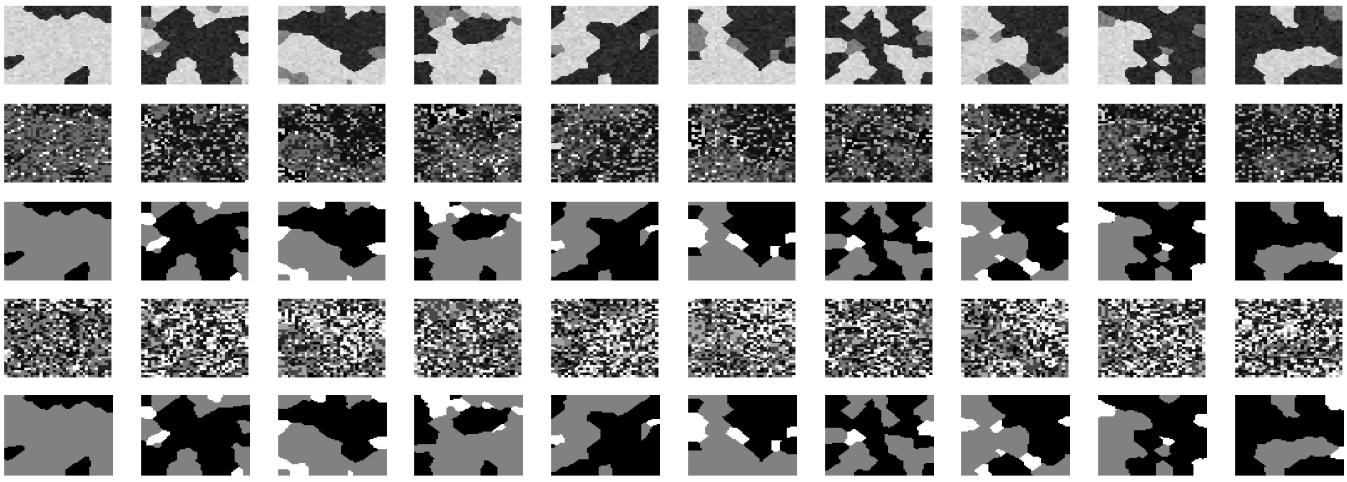


FIGURE E.10 – Images (11 à 20) bruitées avec un niveau de bruit $\sigma_y = 5$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

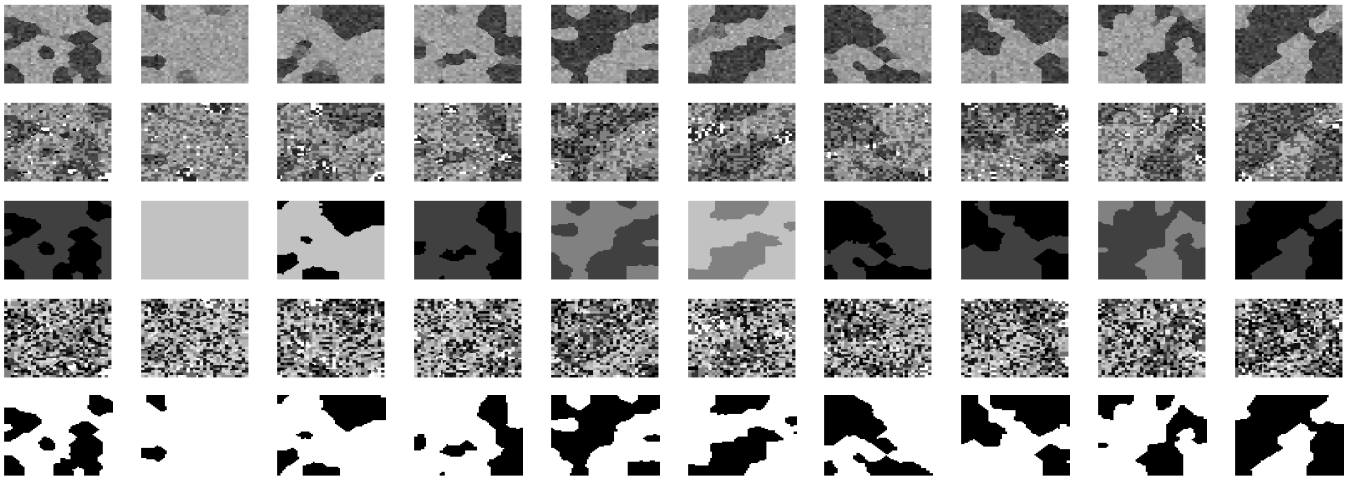


FIGURE E.11 – Images (1 à 10) bruitées avec un niveau de bruit $\sigma_y = 15$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

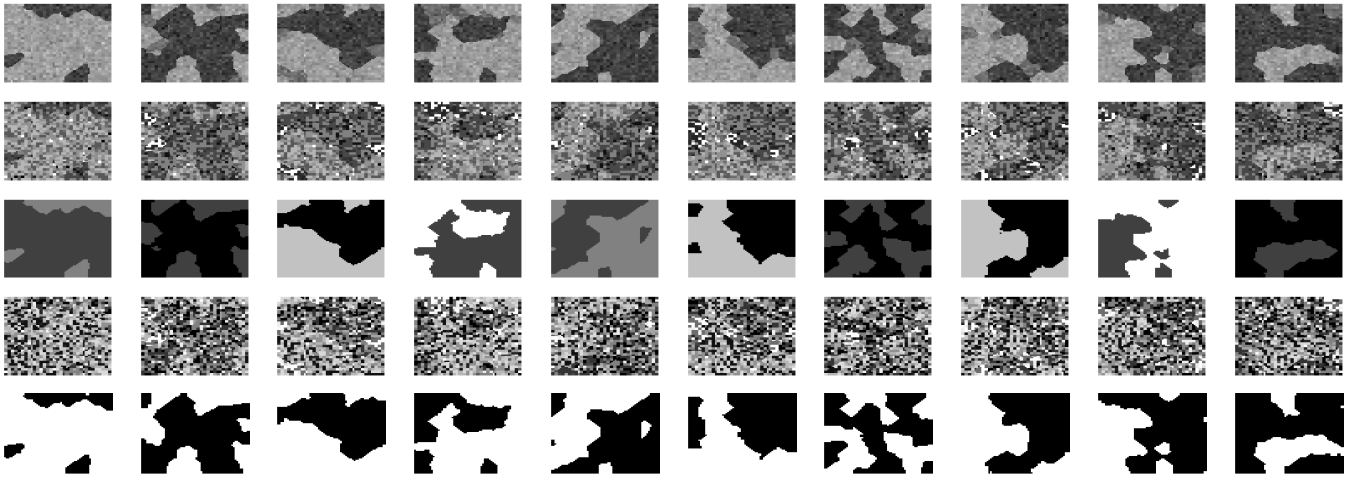


FIGURE E.12 – Images (11 à 20) bruitées avec un niveau de bruit $\sigma_y = 15$ (a), initialisation HDP (b), résultats de segmentation obtenus avec le HDP-Potts (c), initialisation DP (d) et résultats de segmentation obtenus avec le DP-Potts (e) avec $K_{\text{init}} = 15$.

E.4 Résultats complémentaires de segmentation des images de la base de données LabelMe

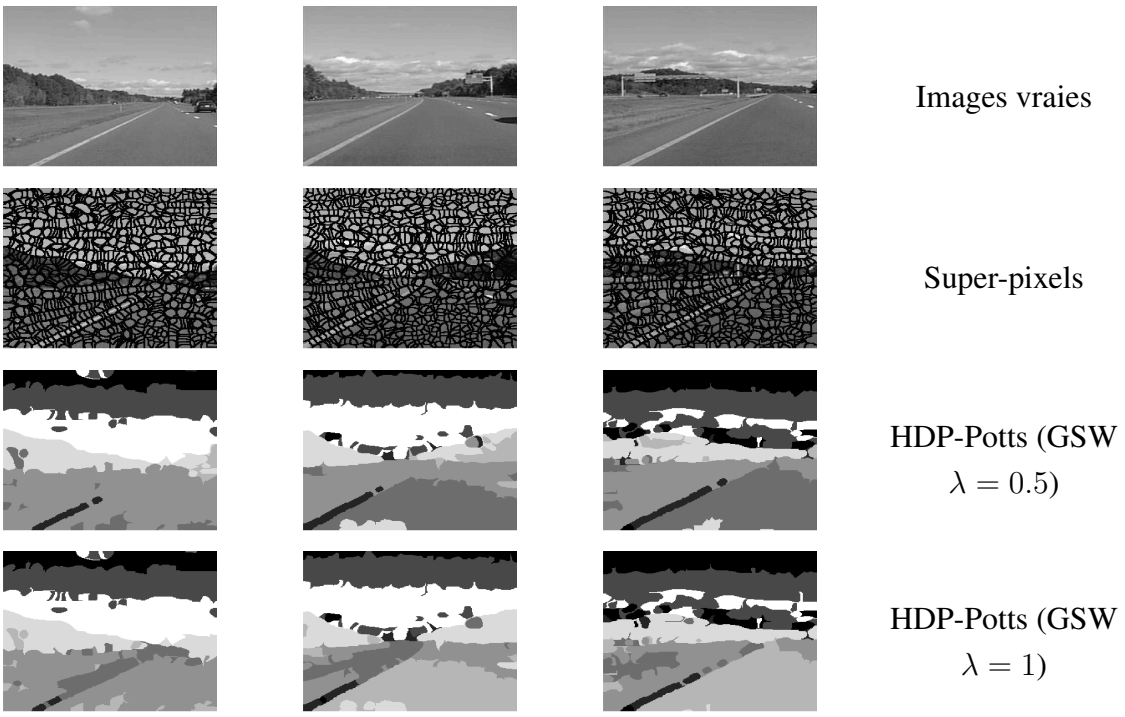


FIGURE E.13 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

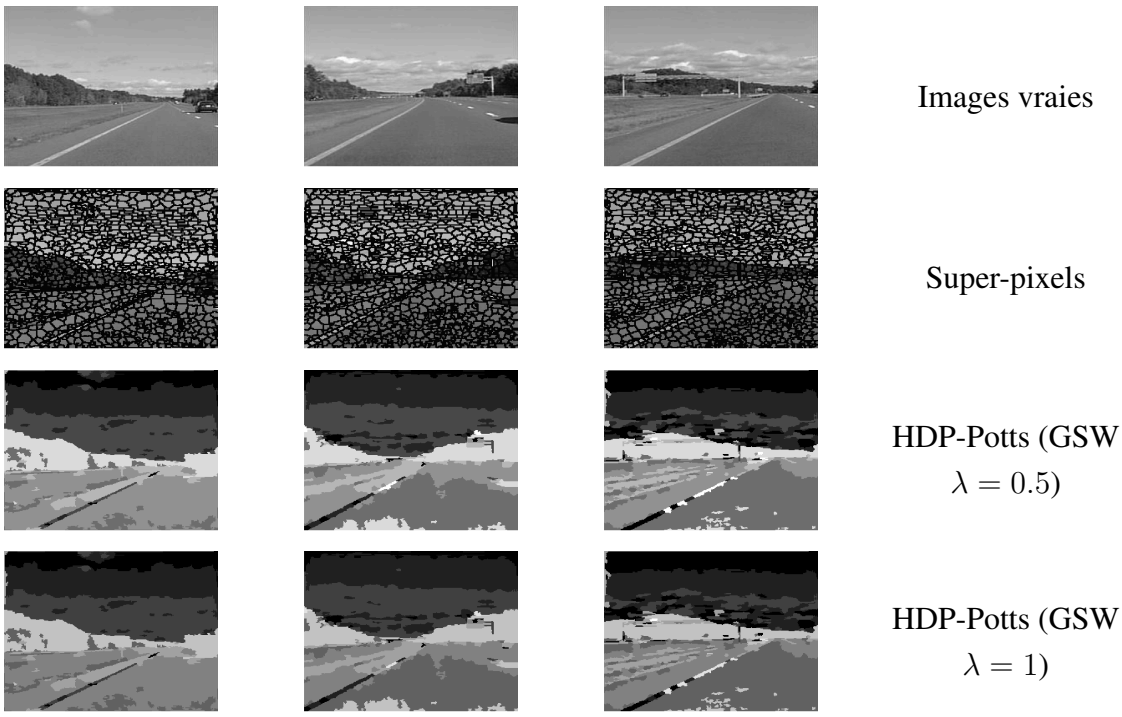


FIGURE E.14 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

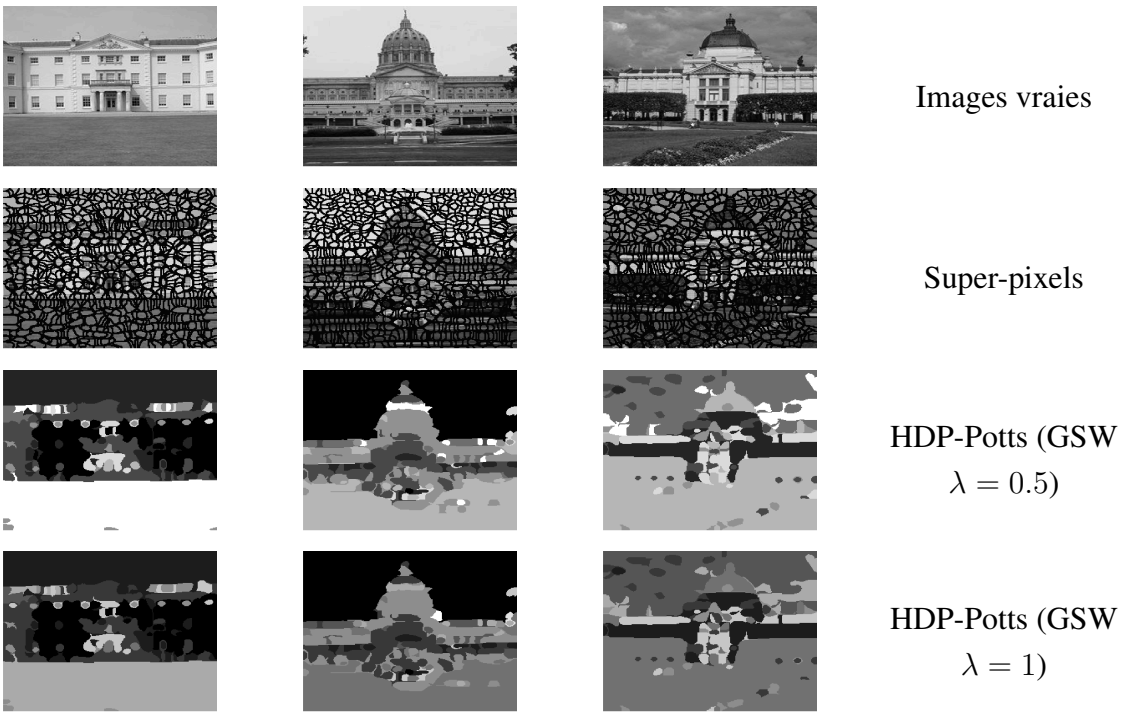


FIGURE E.15 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

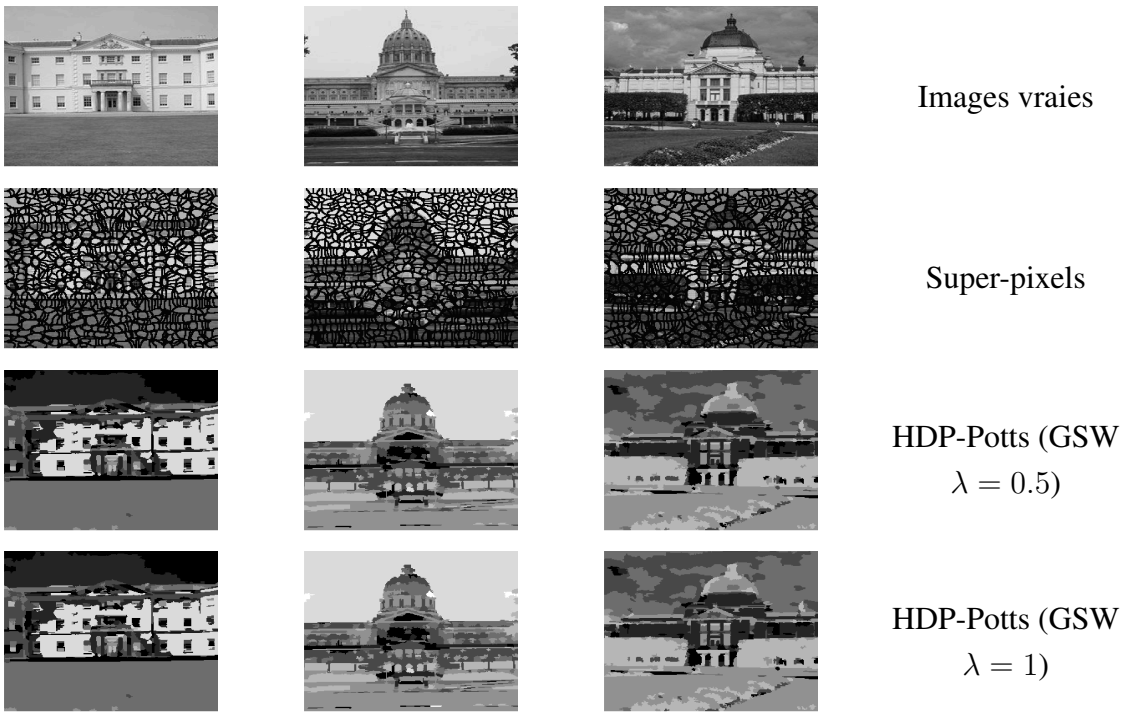


FIGURE E.16 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

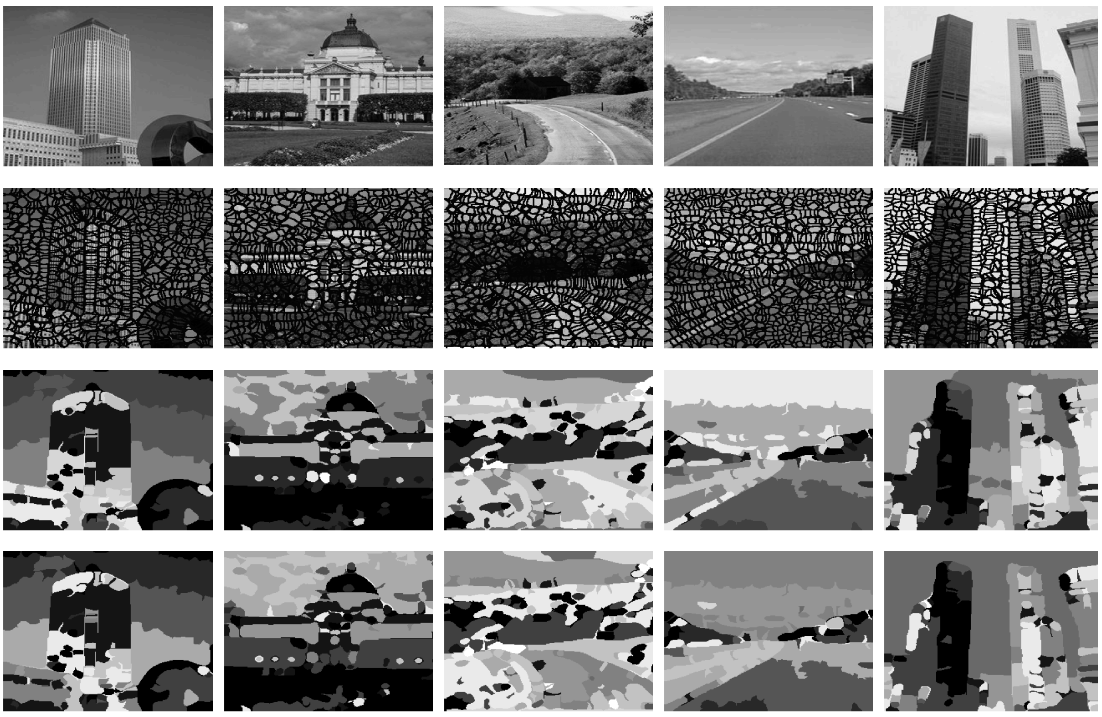


FIGURE E.17 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode *normalized-cuts* pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

FIGURE E.18 – Sur la première ligne, les images vraies, sur la seconde les images sur-segmentées avec la méthode SLIC pour un nombre de super-pixels approximatif de 1000 dans chaque image. Sur les suivantes les résultats de segmentation obtenus avec l'algorithme de Swendsen-Wang généralisé pour $\lambda = 0.5$ et $\lambda = 1$ appliqué au HDP-Potts.

Bibliographie

[AB94] Rolf ADAMS et Leanne BISCHOF : Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–647, June 1994.

[ACT+17] Mohanad ALBUGHDADI, Lotfi CHAARI, Jean-Yves TOURNERET, Florence FORBES, PHILIPPE et CIUCIU : A Bayesian non-parametric hidden Markov random model for hemodynamic brain parcellation. *Signal Processing*, 135:132–146, June 2017.

[AMD09] Hacheme AYASSO et Ali MOHAMMAD-DJAFARI : Joint image restoration and segmentation using gauss-markov-potts prior models and variational bayesian computation. *IEEE Internantional Conference on Image Processing*, pages 1297–1300, 2009.

[Ant74] Charles E. ANTONIAK : Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

[AS65] Milton ABRAMOWITZ et Irene STEGAN : *Handbook of mathematical functions, with formulas, graphs, and mathematical tables*. Dover Publications, New-York, 1965.

[ASS+10] Radhakrishna ACHANTA, Appu SHAJI, Kevin SMITH, Aurelien LUCCHI, Pascal FUA et Sabine SÜSSTRUNK : Slic superpixels. Rapport technique 149300, EPFL, June 2010.

[ASS+12] Radhakrishna ACHANTA, Appu SHAJI, Kevin SMITH, Aurelien LUCCHI, Pascal FUA et Sabine SÜSSTRUNK : Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:2274–2282, May 2012.

[BF11] David M. BLEI et Peter I. FRAZIER : Distance dependent Chinese restaurant processes. *Journal of Machine Learning Research*, 12:2461–2488, November 2011.

[BG98] Stephen BROOKS et Andrew GELMAN : General methods for monitoring convergence of iterative simulations. 7(4):434–455, 1998.

[BGJ07] David M. BLEI, Thomas L. GRIFFITHS et Michael JORDAN : The nested Chinese restaurant process and Bayesian inference of topic hierarchies. 57:7 :1–7 :30, 10 2007.

[Bin78] David BINDER : Bayesian cluster analysis. *Biometrika*, 65(1):31–38, 1978.

[Bis06] Christopher M BISHOP : *Pattern recognition and machine learning*. 2006.

[BM73] David BLACKWELL et James B. MACQUEEN : Ferguson distributions via Pólya urn schemes. *The Annals of Statistics*, 1(2):353–355, 1973.

[BM00] Serge BELONGIE et Jitendra MALIK : Matching with shape contexts. *IEEE Workshop on Contentbased Access of Image and Video Libraries*, 64:20–26, June 2000.

[BNJ03] David M. BLEI, Andrew Y NG et Michael I. JORDAN : Latent dirichlet allocation. *Journal of machine Learning Research*, 3:993–1022, 2003.

[BZ05] Adrian BARBU et Song-Chun ZHU : Generalizing Swendsen-Wang to sampling arbitrary posterior probabilities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1239–1253, August 2005.

[CDL99] Dan CRIŞAN, Pierre DEL MORAL et Terry LYONS : Interacting particle systems approximations of the kushner-stratonovitch equation. *Advances in Applied Probability*, 31(3):819–838, September 1999.

[CFP03] Gilles CELEUX, Florence FORBES et Nathalie PEYRARD : Em procedures using mean field-like approximations for markov model-based image segmentation. *Pattern Recognition*, 36:131–144, January 2003.

[CM02] Dorin COMANICIU et Peter MEER : Mean shift : A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[CMR05] Olivier CAPPÉ, Eric MOULINES et Tobias RYDEN : *Inference in Hidden Markov Models*. Springer Series in Statistics. Springer-Verlag, 2005.

[Coh60] Jacob COHEN : A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, XX(1):37–46, 1960.

[CTM14] François CARON, Yee W. TEH et Thomas B. MURPHY : Bayesian nonparametric plackett–luce models for the analysis of preferences for college degree programmes. *The Annals of Applied Statistics*, 8(2):1145–1181, June 2014.

[Dah06] David DAHL : *Model-based clustering for expression data via a Dirichlet process mixture model in Bayesian inference for gene expression and proteomics, Kim-Anh Do, Peter Müller, Marina Vannucci*. Cambridge University Press, 2006.

[DdG01] Arnaud DOUCET, Nando DE FREITAS et Neil GORDON : *Sequential Monte Carlo in Practice*. Springer-Verlag, New-York, 2001.

[DDJ06] Pierre DEL MORAL, Arnaud DOUCET et Ajay JASRA : Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society B*, 68:411–436, 2006.

[Dia88] Persi DIACONIS : Recent progress on de finetti's notions of exchangeability. *Bayesian statistics*, 3:111–125, 1988.

[DT05] Navneet DALAL et Bill TRIGGS : Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.

[EW95] Michael D. ESCOBAR et Mike WEST : Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430):577–588, June 1995.

[Fer73] Thomas S. FERGUSON : A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, March 1973.

[FH04] Pedro FELZENSZWALB et Daniel HUTTENLOCHER : Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, September 2004.

[FI09] Arno FRITSCH et Katja ICKSTADT : Improved criteria for clustering based on the posterior similarity matrix. *In Bayesian Analysis*, volume 4, pages 367–392, 2009.

[FS06] Sylvia FÜRWITH-SCHNATTER : *Finite mixture and Markov switching models*. Springer, 2006.

[GCS+13] Andrew GELMAN, John B. CARLIN, Hal S. STERN, David B. DUNSON, Aki VEHTARI et Donald B. RUBIN : *Bayesian Data Analysis*. Chapman and Hall/CRC, November 2013.

[Gew89] John GEWEKE : Bayesian inference in econometric models using monte carlo integration. *Econometrica*, 57(6):1317–1339, 1989.

[GG84] Stuart GEMAN et Donald GEMAN : Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:721–741, November 1984.

[GG05] Thomas L. GRIFFITHS et Zoubin GHAHRAMANI : Infinite latent feature models and the indian buffet process. *Advances in Neural Information Processing Systems 18*, pages 475–482, 2005.

[Gre95] Peter GREEN : Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82:711–732, December 1995.

[GUSB11] Soumya GHOSH, Andrei UNGUREANU, Erik SUDDERTH et David BLEI : Spatial distance dependent chinese restauant processes for image segmentation. *In Advances in Neural Information Processing Systems 24*, 2011.

[Hig98] David HIGDON : Auxiliary variable methods for Markov chain Monte Carlo with applications. *Journal of the American Statistical Association*, 93(442):585–595, 1998.

[IZ02] Hemant ISHWARAN et Mahmoud ZAREPOUR : Exact and approximate sum representations for the dirichlet process. *The Canadian Journal of Statistics*, 30:269–283, June 2002.

[KGK+07] Sunil KUMAR, Rajat GUPTA, Nitin KHANNA, Santanu CHAUDHURY et Shiv Dutt JOSHI : Text extraction and document image segmentation using matched wavelets and mrf model. *IEEE Trasactions on Image Processing*, 16:2117–2128, August 2007.

[KLW94] Augustine KONG, Jun S. LIU et Wing H. WONG : Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994.

[KSK76] John G. KEMENY, J. Laurie SNELL et Anthony W. KNAPP : *Denumerable Markov chains*. Springer-Verlag, second édition, 1976.

[LC98] Jun S. LIU et Rong CHEN : Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[Li09] Stan Z. LI : *Markov Random Field Modeling in Image Analysis*. Springer Publishing Company, Incorporated, third édition, 2009.

[Low04] David LOWE : Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[Mac67] James MACQUEEN : Some methods for classification and analysis of multivariate observations. *In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1 : Statistics*, pages 281–297, Berkeley, Calif., 1967. University of California Press.

[Mig08] Max MIGNOTTE : Segmentation by fusion of histogram-based k-means clusters in different color spaces. *IEEE Transactions on Image Processing*, 17:780–787, May 2008.

[MPRB06] Jesper MØLLER, Anthony PETTITT, Robert REEVES et Kasper BERTHELSEN : An efficient markov chain monte carlo method for distributions with intractable normalising constants. *Biometrika*, 93(2):451–458, 2006.

[MTRP06] Clare MCGRORY, D. M. TITTERINGTON, Robert REEVES et Anthony PETTITT : An efficient markov chain monte carlo method for distributions with intractable normalising constants. *Biometrika*, 93(2):451–458, 2006.

[Nea00] Radford NEAL : Markov chain sampling methods for dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2000.

[NHSM12] Takuma NAKAMURA, Tatsuhiro HARADA, Tomohiko SUZUKI et Takasi MATSUMOTO : HDP-MRF : a hierarchical nonparametric model for image segmentation. *International Conference on Pattern Recognition*, November 2012.

[OB07] Peter ORBANZ et Joachim M. BUHMANN : Nonparametric Bayesian image segmentation. *In Int. J. Computer Vision*, numéro 77, pages 25–45, 2007.

[PDBT13] Marcelo PEREYRA, Nicolas DOBIGEON, Hadj BATATIA et Jean-Yves TOURNERET : Estimating the granularity coefficient of a Potts-Markov random field within an MCMC algorithm. *IEEE Transactions on Image Processing*, 22(6):2385–2397, June 2013.

[PSPLF99] Jonathan PRITCHARD, Mark SEIELSTAD, Anna PEREZ-LEZAUN et Marcus FELDMAN : Population growth of human Y chromosomes : a study of Y chromosome microsatellites. *Society of Molecular Biology and Evolution*, pages 1791–1798, 1999.

[PY97] Jim PITMAN et Marc YOR : The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 25(2):855–900, 1997.

[Ras00] Carl RASMUSSEN : The infinite gaussian mixture model. *In In Advances in Neural Information Processing Systems 12*, pages 554–560. MIT Press, 2000.

[RC04] Christian ROBERT et George CASELLA : *Monte Carlo Statistical Methods*. Springer-Verlag, New-York, 2004.

[RH07] Andrew ROSENBERG et Julia HIRSCHBERG : V-measure : A conditional entropy-based external cluster evaluation measure. *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural La,guage Learning*, pages 410–420, 2007.

[Rob73] Thomas V. ROBERTSON : Extraction and classification of objects in multispectral images. Rapport technique, The Laboratory for Applications of Remote Sensing, East Lansing, Michigan, 1973.

[RW06] Carl RASMUSSEN et Christopher WILLIAMS : *Gaussian processes for Machine Learning*. MIT Press, 2006.

[Set94] Jayaram SETHURAMAN : A constructive definition of Dirichlet priors. *Statistica Sinica*, (4):639–650, 1994.

[SGC+16] Jessica SODJO, Audrey GIREMUS, François CARON, Jean-François GIOVANNELLI et Nicolas DOBIGEON : Joint segmentation of multiple images with shared classes : a Bayesian nonparametrics approach. *IEEE Workshop on Statistical Signal Processing (SSP)*, pages 1–5, 2016.

[SGDG17] Jessica SODJO, Audrey GIREMUS, Nicolas DOBIGEON et Jean-François GIOVANNELLI : A generalized swendsen-wang algorithm for bayesian nonparametric joint segmentation of multiple images. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1882–1886, 2017.

[SJ08] Erik B. SUDDERTH et Michael I. JORDAN : Shared segmentation of natural scenes using dependent Pitman-Yor processes. *In Advances in Neural Information Processing Systems 21*, volume 1, pages 1585–1592, 2008.

[SM00] Jianbo SHI et Jitendra MALIK : Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[SW87] Robert SWENDSEN et Jian-Sheng WANG : Nonuniversal critical dynamics in Monte Carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

[SWFU15] Martin STORATH, Andreas WEINMANN, Jürgen FRIKEL et Michael UNSER : Joint image reconstruction and segmentation using the potts model. *Inverse Problems*, 31(2):025003, 2015.

[TJBB06] Yee W. TEH, Michael I. JORDAN, Matthew J. BEAL et David M. BLEI : Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581, December 2006.

[VDP08] David VAN DYK et Taeyoung PARK : Partially collapsed Gibbs samplers : Theory and methods. *Journal of the American Statistical Association*, 103(482):790–796, June 2008.

[VS08] Andrea VEDALDI et Stefano SOATTO : Quick shift and kernel methods for mode seeking. *In In European Conference on Computer Vision*, volume IV, pages 705–718, 2008.

[WL93] Zhenyu WU et Richard LEAHY : An optimal graph theoretic approach to data clustering : Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, November 1993.

[XCD] Richard XU, François CARON et Arnaud DOUCET : Bayesian nonparametric image segmentation using a generalized Swendsen-Wang algorithm. *ArXiv*, (1602.03048).

[ZCP+12] Migyuan ZHOU, Haojun CHEN, Joh PAISLEY, Lu REN, Lingbo LI, Zhengming XING, David DUNSON, Guillermo SAPIRO et Lawrence CARIN : Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Transactions on Image Processing*, 21:130–144, January 2012.