Noname manuscript No.
(will be inserted by the editor)

Nonparametric Bayesian Image Segmentation

Peter Orbanz, Joachim M. Buhmann

Institute of Computational Science, ETH Zürich

The date of receipt and acceptance will be inserted by the editor

Abstract Image segmentation algorithms partition the set of pixels of an image into a specific number of different, spatially homogeneous groups. We propose a nonparametric Bayesian model for histogram clustering which automatically determines the number of segments when spatial smoothness constraints on the class assignments are enforced by a Markov Random Field. A Dirichlet process prior controls the *level of resolution* which corresponds to the number of clusters in data with a unique cluster structure. The resulting posterior is efficiently sampled by a variant of a conjugate-case sampling algorithm for Dirichlet process mixture models. Experimental results are provided for real-world gray value images, synthetic aperture radar images and magnetic resonance imaging data.

Key words Markov random fields – nonparametric Bayesian methods – Dirichlet process mixtures – image segmentation – clustering – spatial statistics – Markov chain Monte Carlo

1 Introduction

Statistical approaches to image segmentation usually differ in two difficult design decisions, i. e. the statistical model for an individual segment and the number of segments which are found in the image. k-means clustering of gray or color values [33], histogram clustering [29] or mixtures of Gaussians [15] are a few examples of different model choices. Graph theoretic methods like normalized cut or pairwise clustering in principle also belong to this class of methods, since these techniques implement versions of weighted or unweighted k-means in kernel space [2,32]. The number of clusters poses a model order selection problem with various solutions available in the literature. Most clustering algorithms require the data analyst to specify the number of classes, based either on a priori knowledge or educated guessing. More advanced methods include strategies for automatic model order selection, i. e. the number of classes is estimated from the input data. Available model selection methods for data clustering include approaches based on coding theory and minimum description length [30], and cross-validation approaches, such as stability [18].

We consider a nonparametric Bayesian approach based on Dirichlet process mixture (MDP) models [11,1]. MDP models provide a Bayesian framework for clustering problems with an unknown number of groups. They support a range of prior choices for the number of classes; the different resulting models are then scored by the likelihood according to the observed data. The number of clusters as an input constant is substituted by a random variable with a control parameter. Instead of specifying a constant number of clusters, the user specifies a level of cluster resolution by adjusting the parameter. These models have been applied to a variety of problems in statistics and language processing; to the best of our knowledge, their application to image segmentation has not yet been studied. Using MDP models for segmentation seems appealing, since a MDP may be interpreted as a mixture model with a varying number of classes, and therefore as a generalization of one of the standard modeling tools used in data clustering. Feature extraction and grouping approaches used with finite mixture models can be transferred to the MDP framework in a straightforward way.

A possible weakness of clustering approaches for image segmentation is related to their lack of spatial structure, i. e. these models neglect the spatial smoothness of natural images: Image segmentation is performed by grouping local features (such as local intensity histograms), and only information implicit in these features is exploited in the search for satisfactory segmentations. Noisy images with unreliable features may result in incoherent segments and jagged boundaries. This drawback can be addressed by introducing spatial coupling between adjacent image features. The classic Bayesian approach to

spatial statistical modeling is based on Markov random field (MRF) priors [14]. It is widely applied in image processing to problems like image restauration and texture segmentation. As will be shown below, MRF priors which model spatial smoothness constraints on cluster labels can be combined with MDP clustering models in a natural manner. Both models are Bayesian by nature, and inference of the combined model may be conducted by Gibbs sampling.

1.1 Previous work

Dirichlet process mixture models have been studied in nonparametric Bayesian statistics for more than three decades. Originally introduced by Ferguson [11] and Antoniak [1], interest in these models has increased since efficient sampling algorithms became available [10,21,28]. MDP models have been applied in statistics to problems such as regression, density estimation, contingency tables or survival analysis (cf. [23] for an overview). More recently, they have been introduced in machine learning and language processing by Jordan et al. [5,25]; see also [38].

Markov random fields define one of the standard model classes of spatial statistics and computer vision [4,37]. In computer vision they have originally been advocated for restauration of noisy images [14,3]. Their application to image segmentation has been studied in [13].

1.2 Contribution of this article

We discuss the application of Dirichlet process mixture models to image segmentation. Our central contribution is a Dirichlet process mixture model with spatial constraints, which combines the MDP approach to clustering and model selection with the Markov random field approach to spatial modeling. Applied to image processing, the model performs image segmentation with automatic model selection under smoothness constraints. A Gibbs sampling algorithm for the combined model is derived. For the application to image segmentation, a suitable MDP model for histogram clustering is defined, and the general Gibbs sampling approach is adapted to this model.

1.3 Outline

Sec. 2 provides an introduction to Dirichlet process methods and their application to clustering problems. Since these models have only recently been introduced into machine learning, we will explain them in some detail. Sec. 3 briefly reviews Markov random field models with pairwise interactions. In Sec. 4, we show how Markov random fields may be combined with Dirichlet process mixture models and derive a Gibbs sampling inference algorithm for the resulting model in Sec. 5. Sec. 6 describes a combined Markov/Dirichlet model for histogram clustering, and Sec. 7 extensions of the basic model. Experimental results obtained by application of the model to image segmentation are summarized in Sec. 8.

2 Dirichlet process methods

Dirichlet process models belong to the family of *nonparametric Bayesian models* [36]. In the Bayesian context, the term nonparametric[1] indicates that these models specify a likelihood function by other means than the random generation of a parameter value. Consider first the standard, parametric Bayesian setting, where the posterior is a parametric model of the form

$$p(\theta|\mathbf{x}) \propto F(\mathbf{x}|\theta)\, G(\theta) \ . \tag{1}$$

Data generation according to this model can be regarded as a two-step process: (i) Choose a distribution function F at random by drawing a parameter value θ from the prior, (ii) then draw a data value $\mathbf{x}$ from this distribution. For a given parameter value θ, the likelihood F is one specific element of a parametric family of distributions. By multiplication with the prior probability G, each possible choice of F is endowed with a probability of occurrence. A Bayesian generative model always requires the random generation of an element F of a function space, and thus of an infinite-dimensional vector space. Parametric Bayesian approaches restrict this space to a finite-dimensional parametric family such that only a finite, constant number of degrees of freedom remains for the resulting problem. Bayesian nonparametric methods generalize this approach by drawing a random function F, but without the restriction to a parametric family. Such a generalization requires a method capable of generating a non-parametric distribution function, an object with an infinite number of degrees of freedom. Dirichlet processes provide one possible solution to this problem.

2.1 Dirichlet process priors

The *Dirichlet process* [11] approach obtains a random probability distribution by drawing its cumulative distribution function (CDF) as the sample path of a suitable stochastic process. The sample path of a stochastic process may be interpreted as the graph of a function.

[1] In the non-Bayesian context, "nonparametric" refers to methods such as Parzen window density estimates, which, for example, require one location parameter per observation. These methods are not of parametric type because the number of model parameters depends on the number of data points; parametric models require the number of parameters (and therefore the complexity of the model) to be fixed.

For a process on an interval $[a,b] \subset \mathbb{R}$, the function defines a CDF if it assumes the value 0 at a, the value 1 at b, and if it is non-decreasing on the interval (a,b). The Dirichlet process (DP) is a stochastic process which can be shown to generate trajectories with the above properties almost surely (i. e. with probability 1). It can also be generalized beyond the case of a real interval, to generate proper CDFs on an arbitrary sample space. The DP is parameterized by a scalar $\alpha \in \mathbb{R}_+$ and a probability measure G_0, called the *base measure* of the process, and it is denoted as $\mathrm{DP}(\alpha G_0)$.

Since a random draw $F \sim \mathrm{DP}(\alpha G_0)$ defines a probability distribution almost surely, the DP may be regarded as a distribution on distributions (or a measure on measures), and is hence suitable to serve as a prior in a Bayesian setting. Instead of drawing a parameter defining a likelihood F at random, the likelihood function itself is drawn at random from the DP:

$$\begin{aligned} \mathbf{x}_1, \ldots, \mathbf{x}_n &\sim F \\ F &\sim \mathrm{DP}(\alpha G_0) \ . \end{aligned} \tag{2}$$

Explicit sampling from this model would require the representation of an infinite-dimensional object (the function F). The DP avoids the problem by marginalization: Given a set of observations, the random function is marginalized out by integrating against the random measure defined by the process.[2] The DP owes its actual applicability to the following properties [11]:

1. *Existence*: The process $\mathrm{DP}(\alpha G_0)$ exists for an arbitrary measurable space $(\Omega, \mathcal{A})$ and an additive probability measure G_0 on Ω.
2. *Posterior estimation*: Assume that observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are generated by a distribution drawn at random from a Dirichlet process, according to (2). Denote by $\hat{F}_n$ the empirical distribution of the observed data,

$$\hat{F}_n(\mathbf{x}) := \frac{1}{n} \sum_{i=1}^{n} \delta_{\mathbf{x}_i}(\mathbf{x}) \ , \tag{3}$$

   where $\delta_{\mathbf{x}_i}$ is the Dirac measure centered at $\mathbf{x}_i$. Then the posterior probability of the random distribution F conditional on the data is also a Dirichlet process:

$$F|\mathbf{x}_1, \ldots, \mathbf{x}_n \sim \mathrm{DP}\left(\alpha G_0 + n\hat{F}_n\right) \ . \tag{4}$$

3. *Sampling*: If a sequence $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is drawn from a random measure $F \sim \mathrm{DP}(\alpha G_0)$, the first data value is drawn according to

$$\mathbf{x}_1 \sim G_0 \ . \tag{5}$$

All subsequent data values are drawn according to

$$\mathbf{x}_{n+1}|\mathbf{x}_1, \ldots, \mathbf{x}_n \sim \frac{n}{\alpha+n}\hat{F}_n(\mathbf{x}_{n+1}) + \frac{\alpha}{\alpha+n}G_0(\mathbf{x}_{n+1}) \ . \tag{6}$$

In a Bayesian framework, where F is considered to be the likelihood of the data, property 2 corresponds to conjugacy in the parametric case: Prior and posterior belong to the same model class, with the posterior parameters updated according to the observations. Eq. (6) states that the posterior puts mass $\frac{n}{\alpha+n}$ on the set of values already observed. This weighting is sometimes referred to as the *clustering property* of the Dirichlet process. Property 3 renders the DP computationally accessible; if the base measure G_0 can be sampled, then $\mathrm{DP}(\alpha G_0)$ can be sampled as well. The combination of properties 2 and 3 facilitates sampling from the posterior, and thereby Bayesian estimation using DP models.

2.2 Dirichlet process mixture models

Dirichlet processes are most commonly used in the form of so-called *Dirichlet process mixture* models, introduced in [1]. These models employ a Dirichlet process to choose a prior at random (rather than a likelihood, as the standard DP model does). The initial motivation for considering MDP models is an inherent restriction of the DP: Regarded as a measure on the set of Borel probability measures, the DP can be shown to be degenerate on the set of discrete measures [11]. In other words, a distribution drawn from a DP is discrete with probability 1, even when the base measure G_0 is continuous. MDP models avoid this restriction by drawing a random prior G from a DP and by combining it with a parametric likelihood $F(\mathbf{x}|\theta)$. Sampling from the model is conducted by sampling

$$\begin{aligned} \mathbf{x}_i &\sim F(\,.\,|\theta_i) \\ \theta_i &\sim G \\ G &\sim \mathrm{DP}(\alpha G_0) \ . \end{aligned} \tag{7}$$

If the parametric distribution F is continuous, the model can be regarded as a convolution of the degenerate density G with a continuous function, resulting in a continuous distribution of the data $\mathbf{x}_i$. Sampling the posterior of a MDP model is more complicated than sampling the standard DP posterior: The sampling formula (6) is obtained by conditioning on the values drawn from the random measure. In the MDP case these are the parameters θ_i. Since the parameters are not observed (but rather the data $\mathbf{x}_i$), they have to be integrated out to obtain a sampling formula conditional on the actual data. This is possible in principle, but may be difficult in practice unless a benign combination of likelihood F and base measure G_0 is chosen.

[2] Other nonparametric methods represent the random function F explicitly, by means of finite-dimensional approximations using e. g. wavelets or Bernstein polynomials [27]. In contrast to such approximation-based techniques, the marginalization approach of the DP may be regarded as a purely statistical solution to the problem of dealing with F.

2.3 MDP models and data clustering

The principal motivation for the application of MDP models in machine learning is their connection with both clustering problems and model selection: MDP models may be interpreted as mixture models. The number of mixture components of these mixtures is a random variable, and may be estimated from the data.

The term *clustering algorithm* is used here to describe an unsupervised learning algorithm which groups a set $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of input data into distinct classes. The number of classes will be denoted by N_C, and the class assignment of each data value $\mathbf{x}_i$ is stored by means of an indicator variable $S_i \in \{1, \ldots, N_\mathrm{C}\}$.

A standard modeling class in data clustering are finite mixture models with distributions of the form

$$p(\mathbf{x}) = \sum_{k=1}^{N_\mathrm{C}} c_k p_k(\mathbf{x}) \ , \tag{8}$$

subject to $c_k \in \mathbb{R}_{\geq 0}$ and $\sum_k c_k = 1$. The vector $(c_1, \ldots, c_{N_\mathrm{C}})$ defines a finite probability distribution, with $c_k = \Pr\{S = k\}$ for a data value drawn at random from the model. Each individual cluster is represented by a single probability distribution p_k. The model assumes a two-stage generative process for $\mathbf{x}$:

$$\begin{aligned} \mathbf{x} &\sim p_S \\ S &\sim (c_1, \ldots, c_{N_\mathrm{C}}) \ . \end{aligned} \tag{9}$$

If the component distributions are parametric models $p_k(\mathbf{x}) = p_k(\mathbf{x}|\theta_k)$, the distribution (8) is a parametric model as well, with parameters $c_1, \ldots, c_{N_\mathrm{C}}$ and $\theta_1, \ldots, \theta_{N_\mathrm{C}}$.

Now we consider the MDP model (7). For a given set of values $\theta_1, \ldots, \theta_n$, which are already drawn from the random measure G, the measure can be integrated out to obtain a conditional prior:

$$\begin{aligned} p(\theta_{n+1}|\theta_1, \ldots, \theta_n) = &\frac{1}{n+\alpha} \sum_{i=1}^{n} \delta_{\theta_i}(\theta_{n+1}) \\ &+ \frac{\alpha}{n+\alpha} G_0(\theta_{n+1}) \ . \end{aligned} \tag{10}$$

This equation specifies the MDP analogue of formula (6). Due to the clustering property of the DP, $\theta_1, \ldots, \theta_n$ will accumulate in $N_\mathrm{C} \leq n$ groups of identical values (cf. Sec. 2.1). Each of these classes is represented by its associated parameter value, denoted θ_k^* for class $k \in \{1, \ldots, N_\mathrm{C}\}$. (That is, $\theta_i = \theta_k^*$ for all parameters θ_i in class k). The sum over sites in the first term of (10) may be expressed as a sum over classes:

$$\sum_{i=1}^{n} \delta_{\theta_i}(\theta_{n+1}) = \sum_{k=1}^{N_\mathrm{C}} n_k \delta_{\theta_k^*}(\theta_{n+1}) \ , \tag{11}$$

where n_k denotes the number of values accumulated in group k. The conditional distribution (10) may be rewritten as

$$\begin{aligned} p(\theta_{n+1}|\theta_1, \ldots, \theta_n) = &\sum_{k=1}^{N_\mathrm{C}} \frac{n_k}{n+\alpha} \delta_{\theta_k^*}(\theta_{n+1}) \\ &+ \frac{\alpha}{n+\alpha} G_0(\theta_{n+1}) \ . \end{aligned} \tag{12}$$

Combination with a parametric likelihood F as in (7) results in a single, fixed likelihood function $F(\,.\,|\theta_k^*)$ for each class k. Therefore, the model may be regarded as a mixture model consisting of N_C parametric components $F(\,.\,|\theta_k^*)$ and a "zero" component (the base measure term), responsible for the creation of new classes.

For a parametric mixture model applied to a clustering problem, the number of clusters is determined by the (fixed) number of parameters. Changing the number of clusters therefore requires substitution of one parametric mixture model by another one. MDP models provide a description of clustering problems that is capable of adjusting the number of classes without switching models. This property is, in particular, a necessary prerequisite for Bayesian inference of the number of classes, which requires N_C to be a random variable within the model framework, rather than a constant of the (possibly changing) model. For a MDP model, a conditional draw from $\mathrm{DP}(\alpha G_0)$ will be a draw from the base measure with probability $\frac{\alpha}{\alpha+n}$. Even for large n, a sufficient number of additional draws will eventually result in a draw from G_0, which may generate a new class. [3] Hence, N_C is a random variable, and an estimate can be obtained by Bayesian inference, along with estimates of the class parameters.

The data to be grouped by a MDP clustering procedure are the observations $\mathbf{x}_i$ in (7). The classes defined by the parameter values $\theta_1^*, \ldots, \theta_k^*$ are identified with clusters. Similar to the finite mixture model, each cluster is described by a generative distribution $F(\,.\,|\theta_k^*)$. In contrast to the finite case, new classes can be generated when the model is sampled. Conditioned on the observations $\mathbf{x}_i$, the probability of creating a new class depends on the supporting evidence observed.

3 Markov random fields

This work combines nonparametric Dirichlet process mixture models with Markov random field (MRF) models to enforce spatial constraints. MRF models have been widely used in computer vision [4,37]; the following brief exposition is intended to define notation and specify the type of models considered.

Markov random fields provide an approach to the difficult problem of modeling systems of dependent random variables. To reduce the complexity of the problem, interactions are restricted to occur only within small

[3] In fact, a new class is generated a. s. unless G_0 is finite.

groups of variables. Dependence structure can conveniently be represented by a graph, with vertices representing random variables and an edge between two vertices indicating statistical dependence.

More formally, a MRF is a collection of random variables defined on an undirected, weighted graph $\mathcal{N} = (V_\mathcal{N}, E_\mathcal{N}, W_\mathcal{N})$, the *neighborhood graph*. The vertices in the vertex set $V_\mathcal{N} = \{v_1, \ldots, v_n\}$ are referred to as *sites*. $E_\mathcal{N}$ is the set of graph edges, and $W_\mathcal{N}$ denotes a set of constant edge weights. Since the graph is undirected, the edge weights $w_{ij} \in W_\mathcal{N}$ are symmetric ($w_{ij} = w_{ji}$). Each site v_i is associated with an observation $\mathbf{x}_i$ and a random variable θ_i. When dealing with subsets of parameters, we will use the notation $\theta_A := \{\theta_i | i \in A\}$ for all parameters with indices in the set A. In particular, $\partial(i) := \{j | (i,j) \in E_\mathcal{N}\}$ denotes the index set of neighbors of v_i in $\mathcal{N}$, and $\theta_{-i} := \{\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n\}$ is a shorthand notation for the parameter set with θ_i removed.

Markov random fields model constraints and dependencies in Bayesian spatial statistics. A joint distribution Π on the parameters $\theta_1, \ldots, \theta_n$ is called a Markov random field w. r. t. $\mathcal{N}$ if

$$\Pi(\theta_i | \theta_{-i}) = \Pi(\theta_i | \theta_{\partial(i)}) \tag{13}$$

for all $v_i \in V_\mathcal{N}$. This *Markov property* states that the random variables θ_i are dependent, but dependencies are local, i. e. restricted to variables adjacent in the graph $\mathcal{N}$. The MRF distribution $\Pi(\theta_1, \ldots, \theta_n)$ plays the role of a prior in a Bayesian model. The random variable θ_i describes a parameter for the generation of the observation $\mathbf{x}_i$. Parameter and observation at each site are linked by a parametric likelihood F, i. e. each $\mathbf{x}_i$ is assumed to be drawn $\mathbf{x}_i \sim F(\,.\,|\theta_i)$.

For the image processing application discussed in Sec. 6, each site corresponds to a location in the image; two sites are connected by an edge in $\mathcal{N}$ if their locations in the image are adjacent. The observations $\mathbf{x}_i$ are local image features extracted at each site.

Defining a MRF distribution to model a given problem requires verification of the Markov property (13) for all conditionals of the distribution, a tedious task even for a small number of random variables and often infeasible for large systems. The Hammersley-Clifford theorem [37] provides an equivalent property which is easier to verify. The property is formulated as a condition on the MRF *cost function*, and is particularly well-suited for modeling. A cost function is a function $H : \Omega_\theta^n \to \mathbb{R}_{\geq 0}$ of the form

$$H(\theta_1, \ldots, \theta_n) := \sum_{A \subset V_\mathcal{N}} H_A(\theta_A) \,. \tag{14}$$

The sum ranges over all possible subsets A of nodes in the graph $\mathcal{N}$. On each of these sets, costs are defined by a local cost function H_A, and θ_A denotes the parameter subset $\{\theta_i | v_i \in A\}$. The cost function H defines a distribution by means of

$$\Pi(\theta_1, \ldots, \theta_n) := \frac{1}{Z_H} \exp(-H(\theta_1, \ldots, \theta_n)) \,, \tag{15}$$

with a normalization term Z_H (the *partition function*). Without further requirements, this distribution does not in general satisfy (13). By the Hammersley-Clifford theorem, the cost function (14) will define a MRF if and only if

$$H(\theta_1, \ldots, \theta_n) = \sum_{C \subset \mathcal{C}} H_C(\theta_C) \,, \tag{16}$$

where $\mathcal{C}$ denotes the set of all cliques, or completely connected subsets, of $V_\mathcal{N}$. In other words, the distribution defined by H will be a MRF if the local cost contributions H_A vanish for every subset A of nodes which are not completely connected. Defining MRF distributions therefore comes down to defining a proper cost function of the form (16).

Inference algorithms for MRF distributions rely on the full conditional distributions

$$\Pi(\theta_i | \theta_{-i}) = \frac{\Pi(\theta_1, \ldots, \theta_n)}{\int \Pi(\theta_1, \ldots, \theta_n) d\theta_i} \,. \tag{17}$$

For sampling or optimization algorithms, it is typically sufficient to evaluate distributions up to a constant coefficient. Since the integral in the denominator is constant with respect to θ_i, it may be neglected, and the full conditional can be evaluated for algorithmic purposes by substituting given values for all parameters in θ_{-i} into the functional form of the joint distribution $\Pi(\theta_1, \ldots, \theta_n)$. Due to the Markov property (13), the full conditional for θ_i is completely defined by those components H_C of the cost function for which $i \in C$. This restricted cost function will be denoted $H(\theta_i | \theta_{-i})$. A simple example of a MRF cost function with continuously-valued parameters θ_i is

$$H(\theta_i | \theta_{-i}) := \sum_{l \in \partial(i)} \|\theta_i - \theta_l\|^2 \,. \tag{18}$$

The resulting conditional prior contribution $M(\theta_i | \theta_{-i}) \propto \prod_{l \in \partial(i)} \exp(-\|\theta_i - \theta_l\|^2)$ will favor similar parameter values at sites which are neighbors.

In the case of clustering problems, the constraints are modeled on the discrete set $\{S_1, \ldots, S_n\}$ of class label indicators. Cost functions such as (18) are inadequate for this type of problem, because they depend on the magnitude of a distance between parameter values. If the numerical difference between two parameters is small, the resulting costs are small as well. Cluster labels are not usually associated with such a notion of proximity: Most clustering problems do not define an order on class labels, and two class labels are either identical or different. This binary concept of similarity is expressed by cost functions such as

$$H(\theta_i | \theta_{-i}) = -\lambda \sum_{l \in \partial(i)} w_{il} \delta_{S_i, S_l} \,, \tag{19}$$

where δ is the Kronecker symbol, λ a positive constant and w_{il} are edge weights. The class indicators S_i, S_l specify the classes defined by the parameters θ_i and θ_l. Hence, if θ_i defines a class different from the classes of all neighbors, $\exp(-H) = 1$, whereas $\exp(-H)$ will increase if at least one neighbor is assigned to the same class. More generally, we consider cost functions satisfying

$$\begin{aligned} H(\theta_i|\theta_{-i}) &= 0 \quad \text{if } S_i \notin S_{\partial(i)} \\ H(\theta_i|\theta_{-i}) &< 0 \quad \text{if } S_i \in S_{\partial(i)} \ . \end{aligned} \tag{20}$$

The function will usually be defined to assume a larger negative value the more neighbors are assigned to the class defined by θ_i. Such a cost function may be used, for example, to express smoothness constraints on the cluster labels, as they encourage smooth assignments of adjacent sites. In Bayesian image processing, label constraints may be used to smooth the results of segmentation algorithms, as first proposed by Geman et al. [13].

4 Dirichlet process mixtures constrained by Markov random fields

Spatially constrained Dirichlet process mixture models are composed of a MRF term for spatial smoothness and a site specific data term. This local data term is drawn from a DP, whereas the interaction term may be modeled by a cost function. We will demonstrate that the resulting model defines a valid MRF. Provided that the full conditionals of the MRF interaction term can be efficiently evaluated, the full conditionals of the constrained MDP/MRF model can be efficiently evaluated as well.

The MRF distribution Π may be decomposed into a sitewise term P and the remaining interaction term M. In the cost function (16), sitewise terms correspond to singleton cliques $C = \{i\}$, and interaction terms to cliques of size two or larger. We denote the latter by $\mathcal{C}_2 := \{C \in \mathcal{C} | |C| \geq 2\}$. The MRF distribution is rewritten as

$$\begin{aligned} \Pi(\theta_1, \ldots, \theta_n) &\propto P(\theta_1, \ldots, \theta_n) M(\theta_1, \ldots, \theta_n) \quad \text{with} \\ P(\theta_1, \ldots, \theta_n) &:= \frac{1}{Z_P} \exp\left(- \sum_i H_i(\theta_i) \right) \\ M(\theta_1, \ldots, \theta_n) &:= \frac{1}{Z_M} \exp\left(- \sum_{C \in \mathcal{C}_2} H_C(\theta_C) \right) . \end{aligned} \tag{21}$$

To construct a MRF-constrained Dirichlet process prior, the marginal distribution $G(\theta_i)$ of θ_i at each site is defined by a single random draw from a DP. The generative representation of the resulting model is

$$\begin{aligned} (\theta_1, \ldots, \theta_n) &\sim M(\theta_1, \ldots, \theta_n) \prod_{i=1}^{n} G(\theta_i) \\ G &\sim \mathrm{DP}\left(\alpha G_0\right) . \end{aligned} \tag{22}$$

The component P in (21), defined in terms of the cost function $H_i(\theta_i)$, has thus been replaced by a random $G \sim \mathrm{DP}\left(\alpha G_0\right)$. To formally justify this step, we may assume a draw G to be given and define a cost function for individual sites in terms of G:

$$H_i(\theta_i) := -\log G(\theta_i) \tag{23}$$

$$Z_G := \int \prod_{i=1}^{n} \exp(-\log G(\theta_i)) d\theta_1 \cdots d\theta_n \tag{24}$$

Since the term acts only on individual random variables, substitution into the MRF will not violate the conditions of the Hammersley-Clifford theorem. When the parameters $(\theta_1, \ldots, \theta_n)$ are drawn from $G \sim \mathrm{DP}\left(\alpha G_0\right)$ as in (7), the θ_i are conditionally independent given G and their joint distribution assumes the product form

$$P(\theta_1, \ldots, \theta_n | G) = \prod_{i=1}^{n} G(\theta_i) \ . \tag{25}$$

This conditional independence of θ_i justifies the product representation (22). The model is combined with a parametric likelihood $F(\,.\,|\theta)$ by assuming the observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$ to be generated according to

$$\begin{aligned} (\mathbf{x}_1, \ldots, \mathbf{x}_n) &\sim \prod_{i=1}^{n} F(\mathbf{x}_i|\theta_i) \\ (\theta_1, \ldots, \theta_n) &\sim M(\theta_1, \ldots, \theta_n) \prod_{i=1}^{n} G(\theta_i) \\ G &\sim \mathrm{DP}\left(\alpha G_0\right) . \end{aligned} \tag{26}$$

Full conditionals $\Pi(\theta_i|\theta_{-i})$ of the model can be obtained up to a constant as a product of the full conditionals of the components:

$$\Pi(\theta_i|\theta_{-i}) \propto P(\theta_i|\theta_{-i}) M(\theta_i|\theta_{-i}) \tag{27}$$

For DP models, $P(\theta_i|\theta_{-i})$ is computed from (25) as described in Sec. 2, by conditioning on θ_{-i} and integrating out the randomly drawn distribution G. The resulting conditional prior is

$$P\left(\theta_i|\theta_{-i}\right) = \sum_{k=1}^{N_C} \frac{n_k^{-i}}{n-1+\alpha} \delta_{\theta_k^*}\left(\theta_i\right) + \frac{\alpha}{n-1+\alpha} G_0\left(\theta_i\right) \ . \tag{28}$$

n_k^{-i} again denotes the number of samples in group k, with the additional superscript indicating the exclusion of θ_i. The representation is the analogue of the sequential representation (12) for a sample of fixed size. The θ_i are now statistically dependent after G is integrated out of the model.

The constrained model exhibits the key property that the MRF interaction term does not affect the base measure term G_0 of the DP prior. More formally, $M(\theta_i|\theta_{-i})G_0$ is equivalent to G_0 almost everywhere, i.e. everywhere on the infinite domain except for a finite set of points. The properties of G_0 are not changed by its values on

a finite set of points for operations such as sampling or integration against non-degenerate functions. Since sampling and integration are the two modes in which priors are applied in Bayesian inference, all computations involving the base measure are significantly simplified. Sec. 5 will introduce a sampling algorithm based on this property.

Assume that $M(\theta_i|\theta_{-i})$ is the full conditional of an MRF interaction term, with a cost function satisfying (20). Combining P with M yields

$$M(\theta_i|\theta_{-i})P(\theta_i|\theta_{-i}) \propto M(\theta_i|\theta_{-i}) \sum_k n_k^{-i} \delta_{\theta_k^*}(\theta_i) + \alpha M(\theta_i|\theta_{-i}) G_0(\theta_i) \; . \tag{29}$$

As an immediate consequence of the cost function property (20), the support of H is at most the set of the cluster parameters $\Theta^* := \{\theta_1^*, \ldots, \theta_{N_C}^*\}$,

$$\operatorname{supp}(H(\theta_i|\theta_{-i})) \subset \theta_{-i} \subset \Theta^* \; . \tag{30}$$

Since Θ^* is a finite subset of the infinite domain Ω_θ of the base measure, $G_0(\Theta^*) = 0$. A random draw from G_0 will not be in Θ^* with probability 1, and hence $\exp(-H(\theta_i|\theta_{-i})) = 1$ almost surely for $\theta_i \sim G_0(\theta_i)$. With $M(\theta_i|\theta_{-i}) = \frac{1}{Z_H}$ almost surely,

$$M(\theta_i|\theta_{-i})G_0(\theta_i) = \frac{1}{Z_H} G_0(\theta_i) \tag{31}$$

almost everywhere. Sampling $M(\theta_i|\theta_{-i})G_0(\theta_i)$ is therefore equivalent to sampling G_0. Integration of $M(\theta_i|\theta_{-i})G_0(\theta_i)$ against a non-degenerate function f yields

$$\begin{aligned} &\int_{\Omega_\theta} f(\theta_i) M(\theta_i|\theta_{-i}) G(\theta_i) d\theta_i \\ &= \int_{\Omega_\theta} f(\theta_i) \frac{1}{Z_H} \exp(-H(\theta_i|\theta_{-i})) G_0(\theta_i) d\theta_i \\ &= \int_{\Omega_\theta \setminus \Theta^*} f(\theta_i) \frac{1}{Z_H} \exp(-H(\theta_i|\theta_{-i})) G_0(\theta_i) d\theta_i \\ &= \frac{1}{Z_H} \int_{\Omega_\theta} f(\theta_i) G_0(\theta_i) d\theta_i \; . \end{aligned} \tag{32}$$

The MRF constraints change only the finite component of the MDP model (the weighted sum of Dirac measures), and the full conditional of Π almost everywhere assumes the form

$$\Pi(\theta_i|\theta_{-i}) \propto \sum_{k=1}^{N_C} M(\theta_i|\theta_{-i}) n_k^{-i} \delta_{\theta_k^*}(\theta_i) + \frac{\alpha}{Z_H} G_0(\theta_i) \tag{33}$$

The formal argument above permits an intuitive interpretation: The finite component represents clusters already created by the model. The smoothness constraints on cluster assignments model a local consistency requirement: consistent assignments are encouraged within neighborhoods. Therefore, the MRF term favors two adjacent sites to be assigned to the same cluster. Unless the base measure G_0 is finite, the class parameter drawn from G_0 will differ from the parameters of all existing classes with probability one. In other words, a draw from the base measure always defines a new class, and the corresponding site will not be affected by the smoothness constraint, as indicated by equation (31).

5 Sampling

Application of the constrained MDP model requires a method to estimate a state of the model from data. Inference for MDP and MRF models is usually handled by Markov chain Monte Carlo sampling. Since full conditionals of sufficiently simple form are available for both models, Gibbs sampling in particular is applicable. We propose a Gibbs sampler for estimation of the combined MDP/MRF model, based on the full conditionals derived in the previous section.

The combined MDP/MRF model (26) can be sampled by a modified version of MacEachern's algorithm [21,22], a Gibbs sampler for MDP models. The original algorithm executes two alternating steps, an *assignment step* and a *parameter update step*. For each data value $\mathbf{x}_i$, the assignment step computes a vector of probabilities q_{ik} for $\mathbf{x}_i$ being assigned to cluster k, according to the current state of the model. An additional probability q_{i0} is estimated for the creation of a new cluster to explain $\mathbf{x}_i$. The assignment indicators S_i are sampled from these assignment probabilities. Given a complete set of assignment indicators S_i, a posterior is computed for each cluster based on the data it currently contains, and the model parameters θ_k^* are updated by sampling from the posterior. More concisely, the algorithm repeats:

1. For each observation $\mathbf{x}_i$, compute the probability q_{ik} of each class $k = 0, \ldots, N_C$ and sample a class label accordingly. Class $k = 0$ indicates the base measure term in (28).
2. Given the class assignments of all sites, compute the resulting likelihoods and sample values for the class parameters.

The sampler notably resembles the expectation-maximization (EM) algorithm for finite mixture models. A MAP-EM algorithm applied to a finite mixture with a prior on the cluster parameters computes a set of assignment probabilities in its E-step, and maximum a posteriori point estimates in the M-step. In the sampler, both steps are randomized, the former by sampling assignment indicators from the assignment probabilities, and the latter by substituting a posterior sampling step for the point estimate.

A sampler for the MDP/MRF model can be obtained by adapting MacEachern's algorithm to the full conditionals of the constrained model, which were computed in the previous section. We define the algorithm before detailing its derivation. Let G_0 be an infinite probability measure, i.e. a non-degenerate measure on an infinite

domain Ω_θ. Let F be a likelihood function such that F, G_0 form a conjugate pair. Assume that G_0 can be sampled by an efficient algorithm. Let H be a cost function of the form (20), and $\mathbf{x}_1, \ldots, \mathbf{x}_n$ a set of observations drawn from the nodes of the MRF. Then the MDP/MRF model (26) can be sampled by the following procedure:

Algorithm 1 (MDP/MRF Sampling)
Initialize: Generate a single cluster containing all points:

$$\theta_1^* \sim G_0\left(\theta_1^*\right) \prod_{i=1}^{n} F\left(\mathbf{x}_i|\theta_1^*\right) \ . \tag{34}$$

Repeat:

1. Generate a random permutation σ of the data indices.
2. *Assignment step.* For $i = \sigma(1), \ldots, \sigma(n)$:
   (a) If $\mathbf{x}_i$ is the only observation assigned to its cluster $k = S_i$, remove this cluster.
   (b) Compute the cluster probabilities

$$\begin{aligned} q_{i0} &\propto \alpha \int_{\Omega_\theta} F\left(\mathbf{x}_i|\theta\right) G_0\left(\theta\right) d\theta \\ q_{ik} &\propto n_k^{-i} \exp\left(-H(\theta_k^*|\theta_{-i})\right) F\left(\mathbf{x}_i|\theta_k^*\right) \end{aligned} \tag{35}$$

   for $k = 1, \ldots, N_\mathrm{C}$.
   (c) Draw a random index k according to the finite distribution $(q_{i0}, \ldots, q_{iN_\mathrm{C}})$.
   (d) Assignment:
   – If $k \in \{1, \ldots, N_\mathrm{C}\}$, assign $\mathbf{x}_i$ to cluster k.
   – If $k = 0$, create a new cluster for $\mathbf{x}_i$.
3. *Parameter update step.* For each cluster $k = 1, \ldots, N_\mathrm{C}$: Update the cluster parameters θ_k^* given the class assignments $S_1, \ldots, S_n$ by sampling

$$\theta_k^* \sim G_0\left(\theta_k^*\right) \prod_{i|S_i=k} F\left(\mathbf{x}_i|\theta_k^*\right) \ . \tag{36}$$

Estimate assignment mode: For each point, choose the cluster it was assigned to most frequently during a given final number of iterations.

The sampler is implemented as a random scan Gibbs sampler, a design decision motivated by the Markov random field. Since adjacent sites couple, the data should not be scanned by index order. Initialization collects all data in a single cluster, which will result in comparatively stable results, since the initial cluster is estimated from a large amount of data. Alternatively, one may start with an empty set of clusters, such that the first cluster will be created during the first assignment step. The initial state of the model is then sampled from the single-point posterior of a randomly chosen observation, resulting in more variable estimates unless the sampler is run for a large number of iterations to ensure proper mixing of the Markov chain. The final assignment by maximization is a rather primitive form of mode estimate, but experiments show that class assignment probabilities tend to be pronounced after a sufficient number of iterations. The estimates are therefore unambiguous. If strong variations in cluster assignments during consecutive iterations are observed, maximization should be substituted by a more sophisticated approach.

The algorithm is derived by computing the assignment probabilities q_{ik} and the cluster posterior (36) based on the parametric likelihood F and the full conditional probabilities (33) of the MDP/MRF model. The posterior for a single observation $\mathbf{x}_i$ is

$$p\left(\theta_i|\theta_{-i}, \mathbf{x}_i\right) = \frac{F(\mathbf{x}_i|\theta_i)\Pi(\theta_i|\theta_{-i})}{\int_{\Omega_\theta} F(\mathbf{x}_i|\theta)\Pi(\theta|\theta_{-i})d\theta} \ . \tag{37}$$

Substituting (33) for $\Pi(\theta_i|\theta_{-i})$ gives

$$\begin{aligned} p\left(\theta_i|\theta_{-i}, \mathbf{x}_i\right) \propto\ & F(\mathbf{x}_i|\theta_i) M(\theta_i|\theta_{-i}) \sum_{k=1}^{N_\mathrm{C}} n_k^{-i} \delta_{\theta_k^*}\left(\theta_i\right) \\ & + F(\mathbf{x}_i|\theta_i) \cdot \frac{\alpha}{Z_H} G_0(\theta_i) \ . \end{aligned} \tag{38}$$

The probabilities of the individual components can be computed as their relative contributions to the mass of the overall model, i. e. by integrating each class component of the conditional (38) over Ω_θ. For each cluster $k \in \{1, \ldots, N_\mathrm{C}\}$ of parameters, the relevant integral measure is degenerate at θ_k^*. Integrating an arbitrary function f against the degenerate measure δ_{θ_j} "selects" the function value $f(\theta_j)$. Hence,

$$\begin{aligned} & \int_{\Omega_\theta} \delta_{\theta_k^*}\left(\theta_i\right) F(\mathbf{x}_i|\theta_i) \frac{1}{Z_H} \exp(-H(\theta_i|\theta_{-i})) d\theta_i \\ & \qquad = \quad \frac{1}{Z_H} F(\mathbf{x}_i|\theta_k^*) \exp(-H(\theta_k^*|\theta_{-i})) \ . \end{aligned} \tag{39}$$

The MRF normalization constant Z_H appears in all components and may be neglected. Combined with the coefficients of the conditional posterior (37), the class probabilities q_{i0} and q_{ij} are thus given by (35).

The class posterior for sampling each cluster parameter θ_k^* is

$$\theta_k^* \sim G_0(\theta_k^*) \prod_{i|S_i=k} F(\mathbf{x}_i|\theta_k^*) M(\theta_k^*|\theta_{-i}) \ . \tag{40}$$

Once again, a random draw $\theta \sim G_0$ from the base measure will not be an element of Θ^* a. s., and

$$F(\mathbf{x}_i|\theta_k^*) M(\theta_k^*|\theta_{-i}) = F(\mathbf{x}_i|\theta_k^*) \frac{1}{Z_H} \tag{41}$$

almost everywhere for a non-degenerate likelihood. Therefore, θ_k^* may equivalently be sampled as

$$\theta_k^* \sim G_0(\theta_k^*) \prod_{i|S_i=k} F(\mathbf{x}_i|\theta_k^*) \ , \tag{42}$$

which accounts for the second step of the algorithm.

If F and G_0 form a conjugate pair, the integral in (35) has an analytical solution, and the class posterior (42) is an element of the same model class as G_0. If G_0 can be sampled, then the class posterior can be sampled as well. Consequently, just as MacEachern's algorithm, Alg. 1 is feasible in the conjugate case. The fact that the clustering cost function gives a uniform contribution a. e. is therefore crucial. With the inclusion of the MRF contribution, the model is no longer conjugate. Due to the finite support of the cost function, however, it reduces to the conjugate case for both steps of the algorithm relying on a conjugate pair.

MacEachern's algorithm is not the only possible approach to MDP sampling. More straightforward algorithms draw samples from the posterior (37) directly, rather than employing the two-stage sampling scheme described above [10]. For the MDP/MRF model, the two-stage approach is chosen because of its explicit use of class labels. The choice is motivated by two reasons: First, the MRF constraints act on class assignments, which makes an algorithm operating on class labels more suitable than one operating on the parameters θ_i. The second reason similarly applies in the unconstrained case, and makes MacEachern's algorithm the method of choice for many MDP sampling problems. If a large class exists at some point during the sampling process, changing the class parameter of the complete class to a different value is possible only by pointwise updates, for each θ_i in turn. The class is temporarily separated into at least two classes during the process. Such a separation is improbable, because for similar observations, assignment to a single class is more probable then assignment to several different classes. Thus, changes in parameter values are less likely, which slows down the convergence of the Markov chain. Additionally, if a separation into different classes occurs, the resulting classes are smaller and the corresponding posteriors less concentrated, causing additional scatter. The two-stage algorithm is not affected by the problem, since parameters are sampled once for each class (rather than for each site). Given the current class assignments, the posterior does not depend on any current parameter values θ_i. The difference between the two algorithms becomes more pronounced when MRF smoothness constraints are applied. For a direct, sequential parameter sampling algorithm, constraints favoring assignment of neighbors to the same class will make separation into different classes even less probable. A two-stage sampling approach therefore seems more suited for sampling the MRF-constrained model.

6 Application to image processing

We will now discuss how the previously described and developed methods can be applied to image segmentation, both with a standard MDP approach and with a MDP/MRF model. The results derived in the previous section have not assumed any restriction on the choice of base measure G_0 and likelihood F (except for the assumption that the base measure is infinitely supported). In the following, we specialize the model by choosing specific distributions for G_0, F and the MRF term M, to define a suitable histogram clustering model for use with the MDP/MRF method.

6.1 A histogram clustering model

Our approach to image segmentation is based on histogram clustering. Given a grayscale image, local histograms are extracted as features. This feature extraction is performed on a rectangular, equidistant grid, placed within the input image. Pixels coinciding with nodes of the grid are identified with sites, indexed by $i = 1, \ldots, n$. A square histogram window is placed around each site, and a histogram $\mathbf{h}_i$ is drawn from the intensity values of all pixels within the window. The size of the window (and therefore the number N_{counts} of data values recorded in each histogram) is kept constant for the whole image, as is the number N_{bins} of histogram bins. Each histogram is described by a vector $\mathbf{h}_i = (h_{i1}, \ldots, h_{iN_{\text{bins}}})$ of non-negative integers. The histograms $\mathbf{h}_1, \ldots, \mathbf{h}_n$ are the input features of the histogram clustering algorithm. They replace the observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in the previous discussion.

The parameters θ_i drawn from the DP in the MDP model are, in this context, the probabilities of the histogram bins (i. e. θ_{ij} is the probability for a value to occur in bin j of a histogram at site i). Given the probabilities of the individual bins, histograms are multinomially distributed, and the likelihood is chosen according to

$$F(\mathbf{h}_i|\theta_i) = N_{\text{counts}}! \prod_{j=1}^{N_{\text{bins}}} \frac{\theta_{ij}^{h_{ij}}}{h_{ij}!} \tag{43}$$
$$= \frac{1}{Z_M(\mathbf{h}_i)} \exp\left(\sum_{j=1}^{N_{\text{bins}}} h_{ij} \log(\theta_{ij}) \right) .$$

The normalization function $Z_M(\mathbf{h}_i)$ does not depend on the value of θ_i.

The prior distribution of the parameter vectors is assumed to be conjugate, and therefore a Dirichlet distribution of dimension N_{bins}. The Dirichlet distribution [17] has two parameters β, π, where β is a positive scalar and π is a N_{bins}-dimensional probability vector. It is defined by the density

$$G_0(\theta_i|\beta\pi) = \frac{\Gamma(\beta)}{\prod_{j=1}^{N_{\text{bins}}} \Gamma(\beta\pi_j)} \prod_{j=1}^{N_{\text{bins}}} \theta_{ij}^{\beta\pi_j - 1} \tag{44}$$
$$= \frac{1}{Z_D(\beta\pi_j)} \exp\left(\sum_{j=1}^{N_{\text{bins}}} (\beta\pi_j - 1) \log(\theta_{ij}) \right) .$$

Sampling of this model will be discussed below, along with sampling of the MRF-enhanced model.

6.2 Histogram clustering with MRF constraints

Combining the histogram clustering model with a MRF constraint requires the choice of a cost function for local smoothness. We have used the simple function

$$H(\theta_i|\theta_{-i}) = -\lambda \sum_{l\in\partial(i)} \delta_{\theta_i,\theta_l} \ . \tag{45}$$

The resulting MRF will make a larger local contribution if more neighbors of site i are assigned to the same class, thereby encouraging spatial smoothness of cluster assignments.

To sample the MRF-constrained histogram clustering model, the sampler (Alg. 1) has to be derived for the particular choice of distributions (43) and (44), which requires computation of the class probabilities q_{i0} and q_{ik} in (35) and the respective posterior (36).

Since F, G_0 form a conjugate pair, their product is (up to normalization) a Dirichlet density:

$$\begin{aligned} F(\mathbf{x}_i|\theta_i)G_0(\theta_i) &\propto \exp\Big(\sum_j (h_{ij} + \beta\pi_j - 1)\log(\theta_{ij})\Big) \\ &= G_0(\theta_i|\mathbf{h}_i + \beta\pi) \ . \end{aligned} \tag{46}$$

Therefore, q_{i0} has an analytic solution in terms of partition functions:

$$\begin{aligned} &\int_{\Omega_\theta} F(\mathbf{h}_i|\theta_i)G_0(\theta_i)d\theta_i \\ &\quad = \int_{\Omega_\theta} \frac{\exp\left(\sum_j (h_{ij} + \beta\pi_j - 1)\log(\theta_{ij})\right)}{Z_M(\mathbf{h}_i)Z_D(\beta\pi)} d\theta_i \\ &\quad = \frac{Z_D(\mathbf{h}_i + \beta\pi)}{Z_M(\mathbf{h}_i)Z_D(\beta\pi)} \ . \end{aligned} \tag{47}$$

For $k = 1, \ldots, N_C$,

$$\begin{aligned} q_{ik} &\propto n_k^{-i} \exp(-H(\theta_k^*|\theta_{-i}))F(\mathbf{h}_i|\theta_k^*) \\ &= \frac{n_k^{-i}}{Z_M(\mathbf{h}_i)} \exp\Big(\lambda \sum_{l\in\partial(i)} \delta_{\theta_i,\theta_l} + \sum_j h_{ij}\log(\theta_{kj}^*)\Big) \ . \end{aligned} \tag{48}$$

Since the multinomial partition function $Z_M(\mathbf{h}_i)$ appears in all equations, the cluster probabilities may be computed for each i by computing preliminary values

$$\begin{aligned} \tilde{q}_{i0} &:= \frac{Z_D(\mathbf{h}_i + \beta\pi)}{Z_D(\beta\pi)} \\ \tilde{q}_{ik} &:= n_k^{-i} \exp\Big(\lambda \sum_{l\in\partial(i)} \delta_{\theta_i,\theta_j} + \sum_j h_{ij}\log(\theta_{kj}^*)\Big) \end{aligned} \tag{49}$$

From these, cluster probabilities are obtained by normalization:

$$q_{ik} := \frac{\tilde{q}_{ik}}{\sum_{l=0}^{N_C} \tilde{q}_{il}} \ . \tag{50}$$

The posterior to be sampled in (42) is Dirichlet as well:

$$\begin{aligned} G_0(\theta_k^* \mid \beta\pi) \prod_{i|S_i=k} F(\mathbf{x}_i|\theta_k^*) \\ &\propto \exp\Big(\sum_j (\beta\pi_j + \sum_{i|S_i=k} h_{ij} - 1)\log(\theta_{kj}^*)\Big) \\ &\propto G_0\Big(\theta_k^* \Big| \beta\pi + \sum_{i|S_i=k} \mathbf{h}_i\Big) \end{aligned} \tag{51}$$

Dirichlet distributions can be sampled efficiently by means of Gamma sampling; cf. for example [8]. Sampling of the unconstrained model may be conducted by choosing $\lambda = 0$ in the MRF cost function.

6.3 Behavior of the segmentation model

Since both the base measure and the posterior sampled in the algorithm are Dirichlet, the properties of this distribution have a strong influence on the behavior of the clustering model. Dirichlet densities are delicate to work with, since they involve a product over exponentials, and because their domain covers a multidimensional real simplex, which renders them difficult to plot or illustrate. The clustering model, however, which has been obtained by combining the Dirichlet base measure and the multinomial likelihood behaves in a manner that is intuitive to understand: Each observed histogram $\mathbf{h}_i$ is assumed to be generated by the likelihood F, which is determined at each site by the parameter θ_i. The vector θ_i lies in the N_{bins}-dimensional simplex $\mathbb{S}^{N_{\text{bins}}}$, and it can be regarded as a finite probability distribution on the histogram bins. Its distribution $G_0(\theta_i|\beta\pi)$ is parameterized by another vector $\pi \in \mathbb{S}^{N_{\text{bins}}}$, which defines the expected value of G_0. The scalar parameter β controls the scatter of the distribution: The larger the value of β, the more tightly G_0 will concentrate around π. For $\beta\pi = (1, \ldots, 1)^t$, G_0 is the uniform distribution on the simplex. Consider the posterior (51), which is a Dirichlet distribution with the scaled vector $\beta\pi$ replaced by $\beta\pi + \sum_{i|S_i=k} \mathbf{h}_i$. By setting

$$\begin{aligned} \tilde{\beta}_k &:= \Big\|\beta\pi + \sum_{i|S_i=k} \mathbf{h}_i\Big\|_1 \\ \tilde{\pi}_k &:= \frac{1}{\tilde{\beta}}\Big(\beta\pi + \sum_{i|S_i=k} \mathbf{h}_i\Big) \ , \end{aligned} \tag{52}$$

the posterior assumes the form $G_0(\,.\,|\tilde{\beta}_k\tilde{\pi}_k)$. For each cluster k, the expected value of the posterior is $\tilde{\pi}_k$, and its scatter is determined by $\tilde{\beta}_k$. The expected value $\tilde{\pi}_k$ is the (normalized) average of the histograms assigned to the cluster, with an additive distortion caused by the base measure parameters. The larger β, the more influence the prior will have, but generally, it has less influence if the number of histograms assigned to the cluster is large. Since $\tilde{\beta}_k$ controls the scatter and grows with the number of histograms assigned, the posterior of a large

cluster will be tightly concentrated around its mean. In other words, for a very large cluster, drawing from the posterior will reproduce the cluster's normalized average with high accuracy. Therefore, larger clusters are more stable. For a smaller cluster, draws from the posterior scatter, and the additive offset $\beta\pi$ has a stronger influence.

Assignments to clusters are determined by sampling from the finite distributions $(q_{i0}, \ldots, q_{iN_C})$, which are based on the multinomial likelihood F. For illustration, consider a non-Bayesian maximum likelihood approach for F. Such an approach would assign each histogram to the class which achieves the highest likelihood score. Multinomial likelihood maximization can be shown to be equivalent to the minimization of the Kullback-Leibler divergence between the distribution represented by the histogram and that defined by the parameter. Each histogram would thus be assigned to the "nearest" cluster, in the sense of the Kullback-Leibler divergence. The behavior of our histogram clustering model is similar, with two notable differences: The greedy assignment is replaced by a sampling approach, and the MDP model may create a new class for a given histogram, instead of assigning it to a currently existing one. The key properties of the model are not affected or altered by adding or removing the MRF constraints, except for the assignment step: The assignment probabilities computed from the basic, unconstrained model are modified by the constraint to increase the probability of a smooth assignment.

7 Extensions of the constrained model

The histogram clustering model introduced in Sec. 6 characterizes image patches by a set of intensity histograms. We will extend this concept to include additional features by modeling multiple channels and side information not contained in the features. The segmentation algorithm becomes directly applicable to multi-channel data, such as color images, multiple frequency bands in radar images, or image filter response data. For color images or multi-band radar data, the segmentation algorithm can draw on marginal intensity histograms extracted from each channel. Filter responses of local image filters can be represented as an image, and be included as additional channels. For example, texture information may be included in color image segmentation by including histograms of Gabor responses.

Including multiple channels increases the amount of data. The information provided by the different channels affects the behavior of the model by means of the likelihood F. The MDP/MRF model provides a second generic way of including additional information, by using side information to adjust the edge weights w_{ij} of the MRF neighborhood graph. The w_{ij} must not depend on the current state of the model (i. e. the values of the model variables θ_i), but they may depend on the data. The coupling strength between adjacent sites may thus be modeled conditional on the local properties of the input image.

7.1 Multiple channels

The MDP/MRF histogram clustering model introduced above represents each site by a single histogram. To model multiple channels, we again assume the marginal histograms to be generated by a multinomial likelihood F, with parameter vectors drawn from a Dirichlet distribution prior G_0. For N_{ch} separate channels, a local histogram $\mathbf{h}_i^c = (h_{i1}^c, \ldots, h_{iN_{\text{bins}}}^c)$ is assumed to be drawn from each channel c at each site i. The channels are parameterized individually, so each histogram $\mathbf{h}_i^c$ is associated with a bin probability vector θ_i^c with prior probability $G_0(\theta_i^c|\beta^c\pi^c)$. The joint likelihood is assumed to factorize over channels. The resulting posterior for site i has the form

$$(\theta_i^1, \ldots, \theta_i^{N_{\text{ch}}}) \sim \prod_{c=1}^{N_{\text{ch}}} F(\mathbf{h}_i^c|\theta_i^c) G_0(\theta_i^c|\beta^c\boldsymbol{\pi}^c) \,. \tag{53}$$

This generalization of the MDP/MRF clustering model (Sec. 4) only affects the base measure G_0 and the random function G in (25), it does not alter the MRF interaction term M. Both the MDP/MRF model and the sampling algorithm remain applicable. In the sampler, only the computation of the cluster probabilities q_{ik} and the cluster posterior in (36) have to be modified. Substituting the multi-channel likelihood F into (49) yields

$$\begin{aligned} \tilde{q}_{i0} &:= \prod_{l=1}^{N_{\text{ch}}} \frac{Z_D(\mathbf{h}_i^c + \beta^l\pi^l)}{Z_D(\beta^c\boldsymbol{\pi}^l) Z_M(\mathbf{h}_i^c)} \\ \tilde{q}_{ik} &:= n_{-i}^k \exp\left(-H(\theta_k^*|\theta_{-i})\right) \prod_{c=1}^{N_{\text{ch}}} F(\mathbf{h}_i^c|\theta_k^{*c}) \,. \end{aligned} \tag{54}$$

Each site remains associated with a single assignment variable S_i (the clustering model groups sites, rather than individual histograms). The cluster posterior (36) is

$$(\theta_i^{*1}, \ldots, \theta_i^{*N_{\text{ch}}}) \sim \prod_{c=1}^{N_{\text{ch}}} G_0\left(\theta_i^{*c} \middle| \beta^c\boldsymbol{\pi}^c + \sum_{i|S_i=k} \mathbf{h}_i^c\right) . \tag{55}$$

This model with multiple channels assumes that local marginal histograms are obtained individually from each channel. It is not applicable to joint histograms. The advantage of marginal histograms is that, unlike joint histograms, they are not affected by the curse of dimensionality. At a constant level of discretization, the number of bins in a joint histogram grows exponentially with the number of dimensions, as opposed to linear growth for a set of marginal histograms. Marginal histograms therefore provide more robust estimates and require less

complex models for their representation. Their disadvantages are (i) the loss of co-occurrence information, and (ii) the independence assumption in (53) required to obtain a feasible model. Choosing marginal histograms can be justified by observing that both problems are limited by the use of local features.

Marginalization of histograms can incur a substantial loss of image information. The global marginal histograms of an RGB image, for example, are informative about the amount of red and blue occurring in the image, but not about the amount of purple. The latter requires a joint histogram. Since the segmentation algorithm relies on local features, the loss of co-occurrence information is limited: If the local marginals show the occurrence of both red and blue within a small local window, a joint histogram will not provide much additional information. Joint histograms measure co-occurrence at pixels, whereas local marginal histograms coarsen the resolution from pixels to local windows.

A similar argument applies for independence: The product in (53) constitutes a local independence assumption, i. e. the marginal histograms $\mathbf{h}_i^1, \mathbf{h}_i^2, \ldots$ are assumed to be independent at site i. Histograms of two different channels at two different sites (e. g. $\mathbf{h}_i^1$ and $\mathbf{h}_l^2$) are not independent, since they interact through the cluster parameters and MRF constraints. Local independence of channels is a more accurate assumption than global independence. Loosely speaking, given a single channel of an entire color image, guessing the image structure (and therefore significant information about the remaining channels) is usually easy. This is not the case for local image patches containing only a few pixels, since their resolution is below the scale of image structures.

7.2 Side information: Image edges

Smoothing constraints may result in unsolicited coupling effects at segment boundaries. Two sites may belong to different segments and still be caused by the smoothing term to be assigned to the same cluster.

Side information on image edges can be used to improve the resolution of segment boundaries, in addition to the input features of the algorithm. Edge information is particularly useful for segmentation, since segment boundaries can be expected to coincide with an image edge. A priori we assume that two sites should not be coupled by a smoothing constraint if they are separated by an image edge. Therefore, edge information may be taken into account in the MDP/MRF model by modifying the neighborhood graph $\mathcal{N}$ of the MRF:

1. Generate an edge map using a standard edge detector.
2. If two sites i and j are separated by an image edge, set $w_{ij} = w_{ji}$ to zero.

Since the MRF constraints act only along edges of the neighborhood graph, this will eliminate coupling between the features $\mathbf{h}_i$ and $\mathbf{h}_j$. Neighborhoods in the MRF graph are usually of small, constantly bounded size ($|\partial(i)| \leq 8$ for the examples provided in the following section), such that the computational expense of this preprocessing step will be linear in the number of sites (rather than quadratic, despite the pairwise comparison).

Given an edge map, i. e. a binary matrix indicating pixels which are classified as edges by the edge detector, the algorithm has to determine whether or not a given pair of sites is separated by an edge. The method used in the experiments presented in Sec. 8 is to remove sites containing an edge pixel in their local image neighborhood from all neighborhoods in $\mathcal{N}$. A single edge is then reinserted (by setting the corresponding weight to 1), such that each site links with at least one of its neighbors. The reinserted edge is chosen in the same direction for all sites (e. g. the edge connecting the site with its left neighbor in the image). This may cause an inaccuracy of edge positions, but only on the scale of the subsampling grid. Simply removing sites completely from the graph neighborhood turns out to interact unfavorably with the model selection property of the MDP algorithm: The histogram windows of sites close to segment boundaries contain mixture distributions from two segments, which typically differ significantly from other local distributions. If coupling constraints with their neighbors are removed, these edge sites tend to be assigned clusters of their own. Edges become visible in the segmentation solution as individual segments. In other words, the approach is prone to remove constraints in regions where they are particularly relevant.

7.3 Side information: Local data disparity

Alternatively, the coupling weights w_{il} may be set according to local data disparity, an approach originally introduced in [13]. The idea is to define a similarity measure $d(\mathbf{x}_i, \mathbf{x}_l)$ between local data vectors and set $w_{il} := d(\mathbf{x}_i, \mathbf{x}_l)$. Substitution into the cost function (19) yields

$$M(\theta_i|\theta_{-i}) \propto \frac{1}{Z_H} \exp\left(-\lambda \sum_{l\in\partial(i)} d(\mathbf{x}_i, \mathbf{x}_l)\delta_{S_i,S_l}\right) \quad (56)$$

for the MRF interaction term. The point-wise contribution P is not affected. Computing the weights from data makes the partition function $Z_H = Z_H(\lambda, \mathbf{x}_i, \mathbf{x}_{\partial(i)})$ data-dependent, but the dependence is uniform over clusters at any given site, and the partition function still cancels from the computation of assignment probabilities in the same manner as described in Sec. 5. The similarity function has to be symmetric, i. e. satisfy $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$, to ensure symmetry of the edge weights. In the case of Euclidean data $\mathbf{x}_i \in \mathbb{R}^m$, for example, d may be chosen as a regularized inverse of the Euclidean distance:

$$d(\mathbf{x}_i, \mathbf{x}_l) := \frac{1}{1 + \|\mathbf{x}_i - \mathbf{x}_l\|_2} \quad (57)$$

The corresponding edge weight w_{il} will be 1 (maximum coupling) for identical data, and decay hyperbolically as the distance between data values increases. Histograms represent finite probability distributions (up to normalization). Hence, for histogram data, norms may be substituted by distribution divergence measures, such as the Kolmogorov-Smirnov statistic [19] or the Jensen-Shannon divergence [20]. The Kullback-Leibler divergence and chi-square statistic [19] are not directly applicable, since neither is symmetric. The dissimilarity measure should be carefully chosen for robustness, since local dissimilarities are measured between individual data values, with no averages to temper the effect of outliers. High-order statistics, such as the Jensen-Shannon divergence, are notoriously hard to estimate from data. For practical purposes, Euclidean norms or the Kolmogorov-Smirnov distance seem a more advisable choice.

In the Bayesian setting, the MRF interaction term is part of the prior. If the MRF cost function depends on the data, the prior function depends on both the parameter variables and the data, a role usually reserved for the likelihood. The data-dependent prior may be justified as a joint distribution on data and parameters, where the data is "fixed by observation", as outlined in [13]. We note the formal difference: A likelihood is a function of both data and parameter, but constitutes a density only with respect to the data. The MRF prior defined above is a density with respect to the parameter variables.

The cost term in (56) measures local differences between the data vectors associated with adjacent sites. The overall model can be interpreted in terms of distances: Many parametric distributions may be regarded as exponentials of average divergences between data and parameters. Multinomial and Dirichlet distributions measure divergence between data and parameters in a Kullback-Leibler sense, and the Gaussian by an average Euclidean distance. Suppose the multinomial/Dirichlet model described in Sec. 6 is combined with the cost function (19) and edge weights $w_{il} = d(\mathbf{h}_i, \mathbf{h}_l)$. The log-posterior of each cluster is a weighted sum of divergence measures, between data and parameter variables (contributed by the likelihood F), hyperparameters and parameter variables (base measure G_0) and data at adjacent sites (MRF interaction term M). The DP hyperparameter α adjusts the sensitivity with which the DP will react to the differences measured by the parametric model by creating new classes.

8 Experimental results

The experiments presented below implement both the unconstrained MDP model and the MDP/MRF model for image segmentation. The unconstrained model is applied to natural images (from the Corel database), which are sufficiently smooth not to require spatial constraints. The MDP/MRF model is applied to synthetic aperture

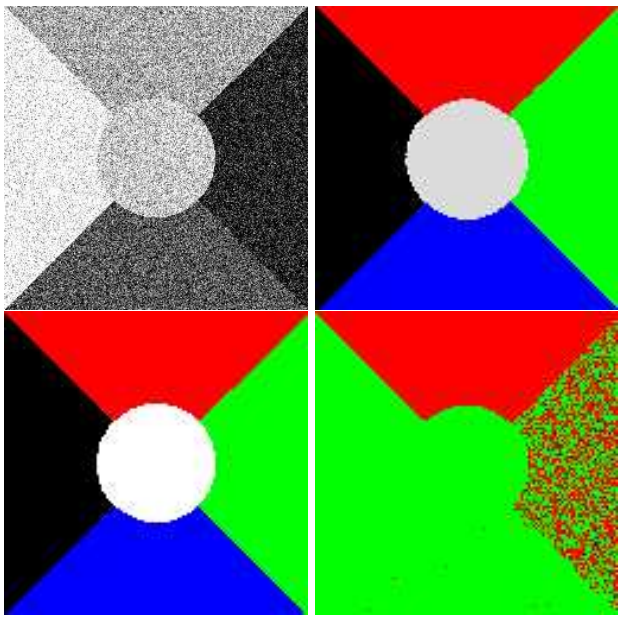

Fig. 1 Behavior of the unconstrained MDP sampler on an image with clearly defined segments. Upper row: Input image (left) and segmentation result for $\alpha = 10$ (right). Bottom row: Segmentation results for $\alpha = 10^{-4}$ (left) and $\alpha = 10^{-10}$.

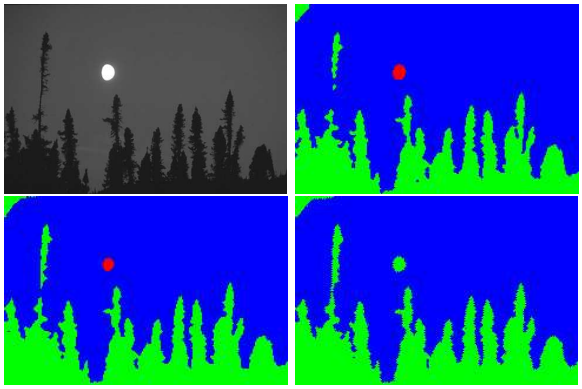

Fig. 2 Unconstrained MDP results on a simple natural image (Corel database): Original image (upper left), MDP results with $\alpha = 10^{-2}$ (upper right), $\alpha = 10^{-7}$ (bottom left), $\alpha = 10^{-9}$ (bottom right).

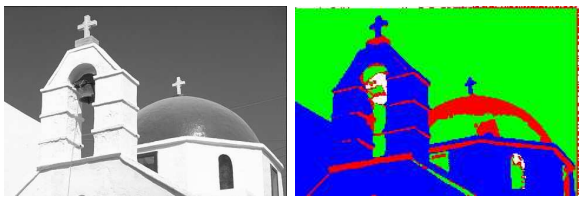

Fig. 3 Natural image (Corel database, left) and unconstrained MDP segmentation result (right).

radar (SAR) images and magnetic resonance imaging (MRI) data, chosen for their high noise level.

8.1 Model parameters

In addition to the parameter α and the input data, the estimate of the number of clusters N_C depends on the parametric model used with the MDP/MRF approach. The Dirichlet process estimates the number of clusters based on disparities in the data. The disparities are measured by the parametric model, which consists of the likelihood F and the base measure G_0. The parameters θ of F are random variables estimated during inference, but any parameters of G_0 are hyperparameters of the overall model. Adjusting the parameters of G_0 changes the parametric model, and thereby influences the model order selection results of the DP prior. In general, increasing the scatter of the distribution G_0 will increase the number of clusters in the MDP solution: The parameters θ_k^* representing the clusters are effectively sampled from a posterior with prior G_0. A concentrated distribution G_0 biases the cluster parameters towards its expected value, and restricts the adaptation of each cluster to the data it contains.

Our strategy is to set the expectation of the base measure to a generic value. The bias incurred from the base measure can then be regarded as data regularization: When a new cluster is created by the algorithm, its initial parameter is based on a single observation. The biasing effect of the hyperparameters will prevent the cluster parameter from adapting to individual outliers. As more observations are collected in the cluster, the influence of the bias decreases. The relative magnitude of the bias is determined by the scatter of the base measure.

The histogram clustering model described in Sec. 6 uses a Dirichlet distribution as its base measure, $G_0 = G_0(\,.\,|\beta\pi)$. The expected value is the parameter π and the scatter is controlled by β (increasing the value decreases scatter). Since $\pi \in \mathbb{S}^d$ represents a finite probability distribution, the obvious choice for a generic value is the uniform vector $\pi = (1/N_{\text{bins}}, \ldots, 1/N_{\text{bins}})$, which was used for most experiments in this section. For some cases, π was chosen as the normalized average histogram of the input image, which adapts the method somewhat better to the input data than a uniform parameter vector, but also tends to result in an algorithm which neglects small segments, as will be discussed below. We propose to choose a β of the same order of magnitude as the mass of a histogram ($\beta = 2N_{\text{counts}}$ was used for the experiments in this section). The regularization effect will be substantial for the creation of new clusters containing only a single histogram, and prevent overfitting of cluster representations to outliers. As soon as the cluster contains a significant number of observations (in particular when it is large enough to be visible in an image segmentation solution), the effect of the bias becomes negligible.

8.2 Image segmentation by a MDP model

As a first test of the model selection property of the MDP clustering algorithm, the (unconstrained) algorithm was applied to an image with unambiguously defined segments (the noisy Mondrian in Fig. 1); the classes are accurately recovered for a wide range of hyperparameter values (α ranging from 10^{-5} to 10^{1}). For a very small value of the hyperparameter ($\alpha = 10^{-10}$), the estimated number of clusters is too small, and image segments are joined erroneously.

Figs. 2 and 3 show images from the Corel database. The three classes in Fig. 2 are clearly discernible, and are once again correctly estimated by the process for $\alpha = 10^{-2}$ and $\alpha = 10^{-7}$. For $\alpha = 10^{-9}$, the process underestimates the number of segments. Note that this results in the deletion of the smallest segment (in this case, the moon): The scatter of the Dirichlet posterior distribution (36) is controlled by the total mass of its parameter vector $(\beta\pi + \sum_{i|S_i=k} \mathbf{h}_i)$. Since large clusters contribute more histogram mass to the parameter vector than small clusters, they are more stable (cf. Sec. 6.3). A small cluster defines a less concentrated posterior, and is less stable. The effect is more pronounced if π is chosen to be the average normalized histogram of the input image, since small segments will be underrepresented. If π is chosen uniform, the offset $\beta\pi$ acts as a regularization term on the average histogram.

The segmentation result in Fig. 3 exhibits a typical weakness of segmentation based exclusively on local histograms: The chapel roof is split into two classes, since it contains significantly different types of intensity histograms due to shading effects. Otherwise, the segmentation is precise, because the local histograms carry sufficient information about the segments.

8.3 Segmentation with smoothness constraints

The results discussed so far do not require smoothing: The presented images (Figs. 2 and 3) are sufficiently smooth, and the noise in Fig. 1 is additive Gaussian, which averages out well even for histograms of small image blocks.

Synthetic aperture radar (SAR) images and MRI data are more noisy than the Corel images. The images shown in Figs. 4 and 5 are SAR images of agricultural areas. In both cases, the unconstrained MDP clustering result are inhomogeneous. Results are visibly improved by the MRF smoothing constraint. Fig. 6 shows results for an image which is hard to segment by histogram clustering, with several smaller classes that are not well-separated and a high noise level. In this case, the improvement achievable by smoothing is limited. Results for a second common type of noisy image, MRI data, are shown in Fig. 8.

The Dirichlet process approach does not eliminate the class number parameter. Like any Bayesian method,

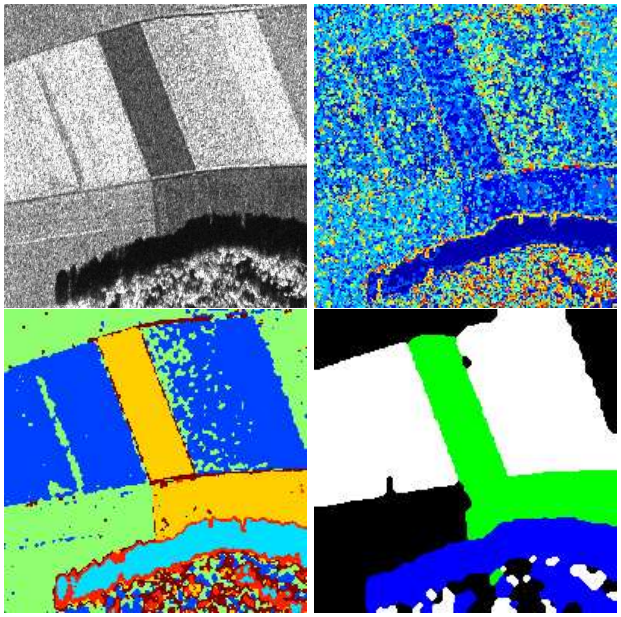

Fig. 4 Segmentation results on real-world radar data. Original image (upper left), unconstrained MDP segmentation (upper right), constrained MDP segmentation at two different levels of smoothing, $\lambda = 1$ (lower left) and $\lambda = 5$ (lower right).

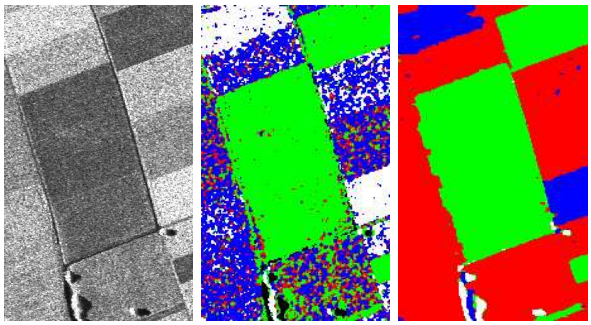

Fig. 5 Original SAR image (left), unconstrained MDP segmentation (middle), smoothed MDP segmentation (right).

it effectively replaces the parameter by a random variable, which is equipped with a prior probability. The prior is controlled by means of the hyperparameter α. The number of classes depends on α, but the influence of the hyperparameter can be overruled by observed evidence. A question of particular interest is therefore the influence of the hyperparameter α on the number of clusters. Table 1 shows the average number of clusters selected by the model for a wide range of hyperparameter values, ranging over several orders of magnitude. Averages are taken over ten randomly initialized experiments each. In general, the number of clusters increases monotonically with an increasing value of the DP scatter parameter α. With smoothing activated, the average estimate becomes more conservative, and more stable with respect to a changing α. The behavior of the estimate depends on the class structure of the data. If the

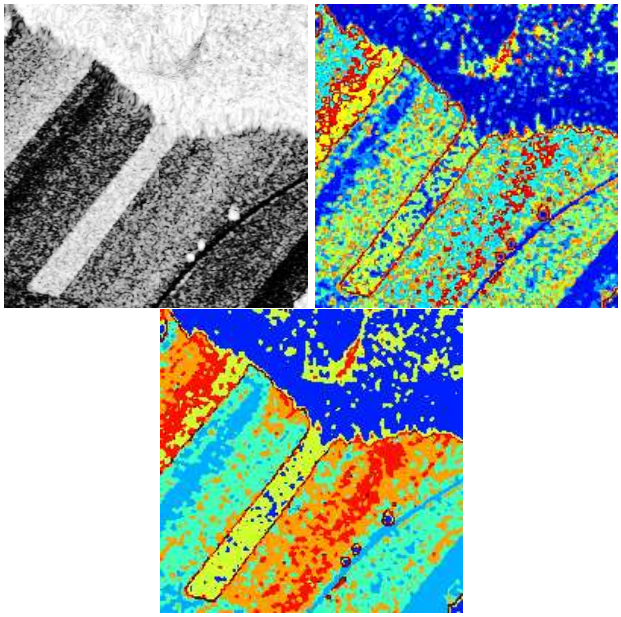

Fig. 6 A SAR image with a high noise level and ambiguous segments (upper left). Solutions without (upper right) and with smoothing.

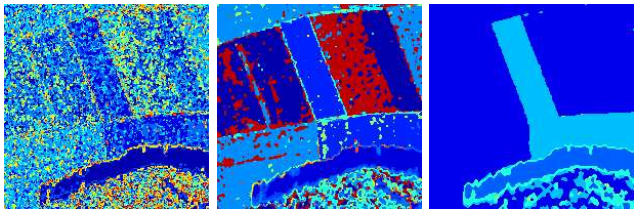

Fig. 7 Segmentation results for $\alpha = 10$, at different levels of smoothing: Unconstrained (left), standard smoothing ($\lambda = 1$, middle) and strong smoothing ($\lambda = 5$, right).

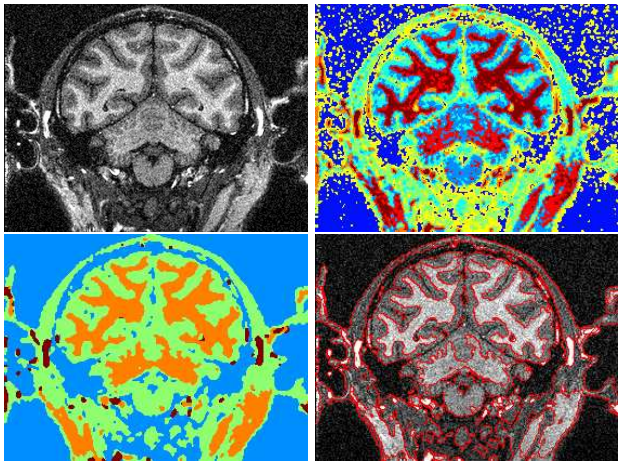

Fig. 8 MR frontal view image of a monkey's head. Original image (upper left), unsmoothed MDP segmentation (upper right), smoothed MDP segmentation (lower left), original image overlaid with segment boundaries (smoothed result, lower right).

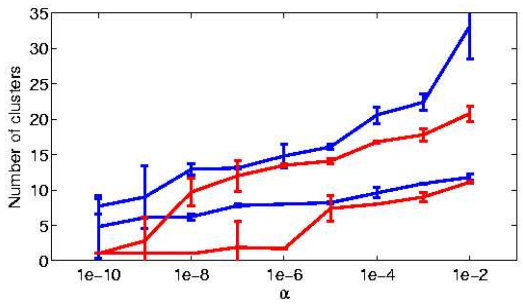

Fig. 9 Influence of the base measure choice: Average number of clusters plotted against α, for two different values of base measure scatter. Blue curves represent $\beta = 50$, red curves $\beta = 200$. In either case, the upper curve corresponds to the unsmoothed and the lower curve to the smoothed model.

	Image Fig. 4		Image Fig. 8	
α	MDP	smoothed	MDP	smoothed
1e-10	7.7 ± 1.1	4.8 ± 1.4	6.3 ± 0.2	2.0 ± 0.0
1e-8	12.9 ± 0.8	6.2 ± 0.4	6.5 ± 0.3	2.6 ± 0.9
1e-6	14.8 ± 1.7	8.0 ± 0.0	8.6 ± 0.9	4.0 ± 0.0
1e-4	20.6 ± 1.2	9.6 ± 0.7	12.5 ± 0.3	4.0 ± 0.0
1e-2	33.2 ± 4.6	11.8 ± 0.4	22.4 ± 1.8	4.0 ± 0.0

Table 1 Average number of clusters (with standard deviations), chosen by the algorithm on two images for different values of the hyperparameter. When smoothing is activated ($\lambda = 5$, right column), the number of clusters tends to be more stable with respect to a changing α.

data is well-separated, estimation results become more stable, as is the case for the MRI image (Fig. 8). With smoothing activated, the estimated number of clusters stabilizes at $N_C = 4$. In contrast, the data in Fig. 4 does not provide sufficient evidence for a particular number of classes, and no stabilization effect is observed. We thus conclude that, maybe not surprisingly, the reliability of MDP and MDP/MRF model selection results depends on how well the parametric clustering model used with the DP is able to separate the input features into different classes. The effect of the base measure scatter, defined here by the parameter β, is demonstrated in Fig. 9. The number of clusters selected is plotted over α at two different values of $\beta = 50$ and $\beta = 200$, each with and without smoothing. The number of clusters is consistently decreased by increasing β and activating the smoothing constraint.

The stabilizing effect of smoothing is particularly pronounced for large values of α, resulting in a large number of clusters selected by the standard MDP model. Results in Fig. 7 were obtained with $\alpha = 10$, which results in an over-segmentation by the MDP model ($\bar{N}_C = 87.1$). With smoothing, the estimated number of clusters decreases ($\bar{N}_C = 29.1$). The level of smoothing can be increased by scaling the cost function. By setting $\lambda = 5$, the number of clusters is decreased further, to $\bar{N}_C = 8.2$.

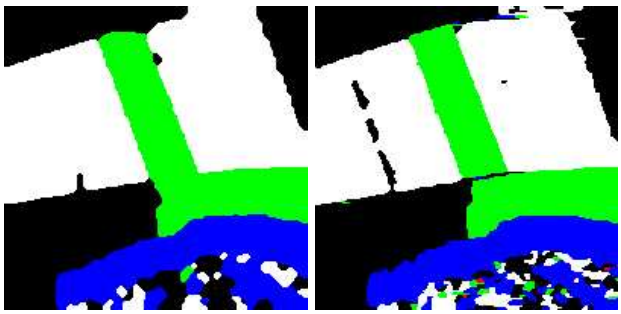

Fig. 10 Stabilization of segmentation results by edge information for a strong smoothing constraint: Smoothed segmentation (left), and the same experiment repeated using edge information (right), both conducted on the image in Fig. 4.

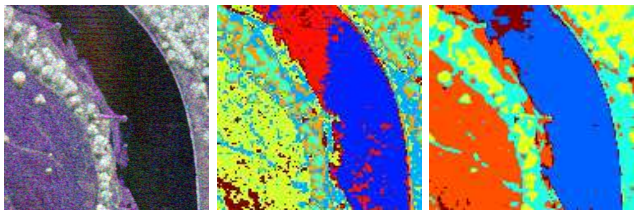

Fig. 11 Multi-channel information: A SAR image consisting of three frequency bands (left), segmentation solutions obtained from the averaged single channel by the standard MRF/MDP model (middle) and by the multi-channel model (right).

8.4 Extensions: Edges and multiple channels

Long runs of the sampler with a large value of λ, which may be necessary on noisy images to obtain satisfactory solutions, can result in unsolicited smoothing effects. Comparing the two smoothed solutions in Fig. 4 (lower left and right), for example, shows that a stronger smoothing constraint leads to a deterioration of some segment boundaries. The segment boundaries can be stabilized by including edge information as described in Sec. 7.2. An example result is shown in Fig. 10.

For SAR images consisting of multiple frequency bands, the multi-channel version of the MDP/MRF model (Sec. 7.1) can be applied. A segmentation result is shown in Fig. 11. Both solutions were obtained with smoothing. To demonstrate the potential value of multiple channel information, only a moderate amount of smoothing was applied. One solution (middle) was obtained by converting the multi-channel input image into a single-channel grayscale image before applying the MDP/MRF model. The second solution (right) draws explicitly on all three frequency bands by the multi-channel model. Parameter values for the single-channel and multi-channel approach are not directly comparable. When computing the cluster assignment probabilities q_{ik}, the multi-channel model multiplies probabilities over channels. Hence, the computed values are generally smaller than in the single-channel case. This increases the relative influence of α,

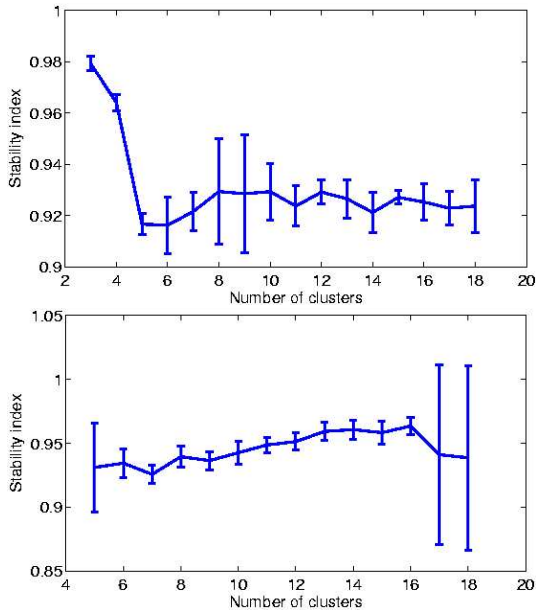

Fig. 12 Stability index results over number of clusters, plotted for images in Fig. 8 (top) and Fig. 4 (bottom).

and the multi-channel approach tends to select more clusters for the same parameter values than the single-channel model. To make the result comparable, we have chosen examples with similar number of clusters ($N_C = 7$ and $N_C = 5$, respectively). The segmentation result is visibly improved by drawing on multi-channel features.

8.5 Comparison: Stability

Relating the approach to other methods is not straightforward, since model order selection methods typically try to estimate a unique, "correct" number of clusters. We use the *stability method* to devise a comparison that may offer some insight into the behavior of the MDP model.

Stability-based model selection for clustering [9,7,18] is a frequentist model selection approach for grouping algorithms, based on cross-validation. It has been demonstrated to perform competitively compared to a wide range of published cluster validation procedures [18]. The stability algorithm is a wrapper method for a clustering algorithm specified by the user. It is applicable to any clustering algorithm which computes a unique assignment of an object to a cluster, e. g. it can be applied to a density estimate (such as mixture model algorithms) with maximum a posteriori assignments. The validation procedure works as follows: The set of input data is split into two subsets at random, and the clustering algorithm is run on both subsets. The model computed by the clustering algorithm on the first set (training data) is then used to *predict* a solution on the second set (test data). The two solutions on the second set, one obtained by clustering and one by prediction, are compared to compute a "stability index". The index measures how well the predicted solution matches the computed one; the mismatch probability is estimated by averaging over a series of random split experiments. Finally, the number of clusters is selected by choosing the solution most stable according to the index.

The MDP model is built around a Bayesian mixture model, consisting of the multinomial likelihood F and the Dirichlet prior distribution G_0. The Bayesian mixture without the DP prior can be used as a clustering model for a fixed number of segments. Inference of this model may be conducted by a MCMC sampling algorithm closely related to MacEachern's algorithm for MDP inference. The only substantial difference between the algorithms is the additional assignment probability term corresponding to the base measure, as observed in [31]. A wrapper method like stability allows us to compare the behavior of the MDP approach to a method using exactly the same parametric model, including the base measure and its scatter parameter β. Only the parameter α is removed from the overall model, and the random sampling of the model order replaced by a search over different numbers of clusters.

Stability index results are shown in Fig. 12 for two images, the monkey image in Fig. 8 and the SAR image in Fig. 4. Results are not smoothed, because the subsampling strategy will break neighborhoods. In both cases, model order selection results for these noisy images are ambiguous. For the monkey image (upper graph), results for $N_C \geq 5$ are mostly within error bars of each other. A smaller number of clusters is ruled out, which is consistent with the unsmoothed MDP results (Tab. 1). For the SAR image, stability results are also ambiguous, but exhibit a significant, monotonous growth with the number of clusters, which is consistent with the monotonous behavior or the MDP results as α increases.

In general, stability has been reported to produce somewhat conservative estimates, since only the stability index of a solution is taken into account [18]. This observation is apparently reflected by the behavior of both methods on the monkey image, where the MDP approach settles at 6 clusters (with very small standard deviation), whereas stability advocates solutions with $N_C \geq 5$.

8.6 Convergence behavior

Gibbs sampling algorithms are notoriously slow, and it is often difficult to determine whether or not the algorithm has converged to the distribution of interest. Gibbs sampling results reported in the MDP literature are typically based on several thousand iterations.

To the advantage of our algorithm, we are interested in segmentation results rather than parameter estimates. The cluster labels are discrete and tend to stabilize after

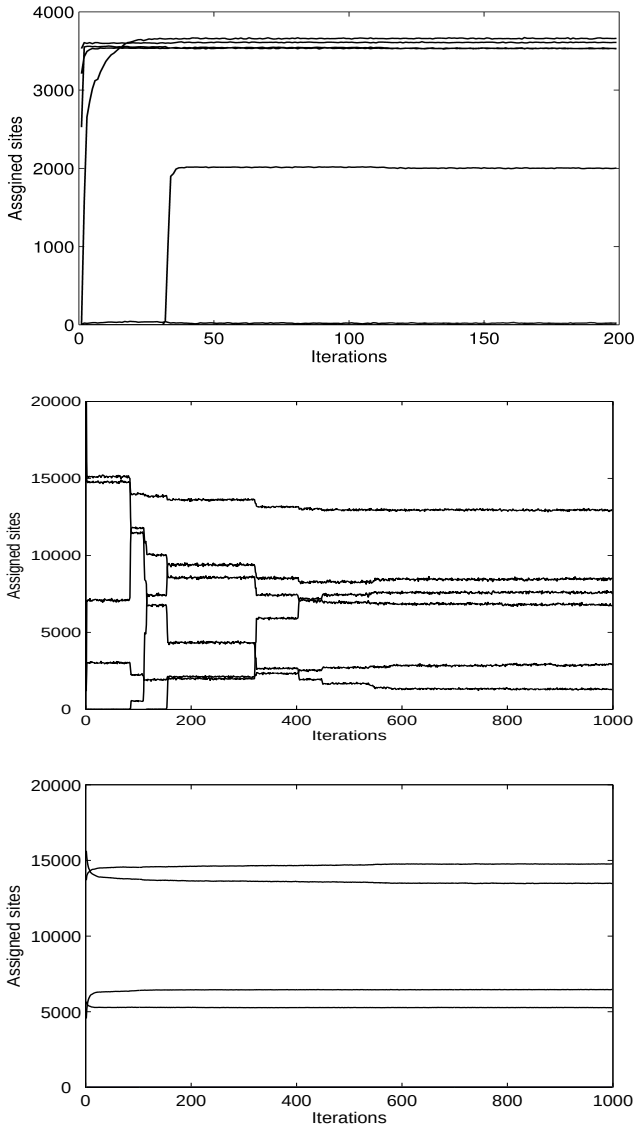

Fig. 13 Split-merge behavior of the sampler for different images and parameters. The number n_k of sites assigned to each cluster (vertical) are drawn against the number of iterations (horizontal), with each graph representing a cluster. Top: Mondrian image (Fig. 1, lower left), no smoothing. Middle: Radar image (Fig. 4), no smoothing. Bottom: Radar image (Fig. 4), with smoothing.

the initial burn-in. Therefore, after discarding the burn-in, class assignments can be estimated reliably from a small number of samples. The indicator for convergence used in the experiments is the relative fluctuation of class labels per iteration. The burn-in phase is assumed to be over once the number of assignments changed per iteration remains stable below 1% of the total number of sites. For the non-smoothing MDP sampler, this condition is usually met after no more than 500-1000 iterations – details depending on the input data and the scatter of the DP. These figures are comparable to those reported in the MDP literature. For example, [21] discards 1000 iterations as burn-in (and estimates are then obtained from 30000 subsequent iterations).

Fig. 13 shows the behavior of class assignment during the sampling process, for the noisy Mondrian and one radar image. For the Mondrian image with its well-separated segments, 40 iterations suffice for the clustering solution to stabilize (the cluster graph turns constant). On the radar image, both the non-smoothing and the smoothing version of the algorithm take about 600 iterations to stabilize, but their splitting behavior differs significantly: The standard MDP algorithm creates the last new significant cluster after about 150 iterations, while the MDP/MRF algorithm creates its classes during the first few iterations and slowly adjusts assignments throughout the sampling process. Without smoothing, large batches of sites are suddenly reassigned from one cluster to another (visible as jumps in the diagram). With smoothness constraints, clusters change gradually. Since the curves represent cluster sizes, they do not indicate the explorative behavior of the sampler. Even if the curve is smooth, the sampler may still explore a large number of possible states in parameter space, depending on the posterior.

9 Conclusions

Segmentation models for mid-level vision have to address the two core issues of what a suitable model for individual segments should capture and how many segments should be inferred from an image. The last decade has seen significant progress in segmentation algorithms ranging from graph-based methods like partitioning models [13], pairwise clustering [16,12] and Normalized Cut [34] to variational [26] and statistical [35] approaches. The specific nature of the images and the intended computer vision task most often determine the appropriateness of a model and the success of its related algorithm. The comparison is still subjective to a large degree, although the Berkeley data base of hand segmented natural color images [24] allows us to benchmark new algorithms against human performance.

The principal focus of this paper is on the second question, i. e. choosing the number of segments. Dirichlet process mixture models have been applied to image segmentation when the statistical model of an individual segment is defined by a parametric likelihood, such as a multinomial distribution. We have theoretically and experimentally demonstrated how MDP models can be combined with Markov random fields to model spatial constraints. A suitable histogram clustering model has been defined, and its properties have been discussed for a number of different images. A Gibbs sampling algorithm for the Dirichlet process mixture combined with the Markov random field constraint has been derived, which can be executed as efficiently as a conjugate-case algorithm for standard MDP models.

The applicability of the MDP/MRF model is not restricted to either image segmentation or histogram clustering. Any kind of parametric mixture model may be used, by choosing the likelihood function F appropriately, and defining a suitable base measure to generate the parameter values. One might, for example, consider a k-means model with variable number of clusters and smoothness constraints, by defining F to be a Gaussian of fixed scale. The mean parameters are drawn from the base measure. If the base measure is also defined as a Gaussian (and therefore conjugate to F), the sampling algorithm proposed in Sec. 5 remains applicable as well. Similar models without spatial constraints (a conjugate pair of normal distributions in the MDP framework) have already been studied in statistics [23]. Furthermore, we expect that our model covers a large part of the landscape of segmentation algorithms since normalized cut and pairwise clustering can be written as weighted and unweighted versions of k-means in feature space [32].

MDP methods do not "solve" the model order selection problem, because the number of clusters is replaced rather than removed as an input parameter. The utility of DP priors is not a decrease in the number of parameters, but the substitution of the constant model order by a random variable. The behavior of the random variable is parameter-controlled, and its eventual value estimated from data. Rather than specifying a number of image segments, the user can specify a "*level of resolution*" for the resulting segmentation. Part of the appeal of MDP-based models is their simplicity. Despite lengthy theoretical derivations, the final form of the model relevant for application is essentially a parametric mixture model with an additional term defined by the base measure. Familiar intuition for parametric mixture models remains applicable, and inference can be conducted by a sampling algorithm with a structure reminiscent of the expectation-maximization algorithm.

Since MDP and MDP/MRF models are built around a parametric model, careful parametric modeling is crucial for their successful application. The DP parameter α specifies a sensitivity with which the DP prior reacts to disparities in the data by creating additional clusters. The disparities are measured by the parametric model. As discussed in Sec. 8, modification of the parametric model will directly influence the MDP results. Hence, an MDP model can only be expected to work well if the class structure in the features is properly resolved by the parametric model. A clearly discernible cluster structure results in stable model order selection. Smoothing constraints can serve to emphasize cluster structure and stabilize results.

The present work is focused on the segmentation of individual images. We expect, however, that MDP-based models will develop their full potential when applied to multiple images. Possible examples include video sequences or collections of radar images obtained by a satellite. In both cases, the number of segments may vary from image to image, but the images are drawn from the same source or very similar sources. If the number of segments is an input parameter, it has to be reset manually for each instance. Nonparametric Bayesian models treat the number of segments as a random variable, with a distribution depending on the image instance. Since the distribution is controlled by parameters, they enable the data analyst to specify a segment resolution, possibly by calibrating the model parameters on a small subset of the data. Applied to new image instances with similar statistical properties, the model will automatically adapt the number of segments to variations in the data. Application to large numbers of image instances will require efficient inference algorithms for MDP models. Considering the progress made in inference for auxiliary variable problems in recent years, both by MCMC sampling and variational inference, feasible methods for large-scale inference can be expected to become available in the near future. The first results on efficient approximations for MDP models are already available [6]. We believe that, as the focus of computer vision shifts towards large collections of images and video sequences, random modeling of the number of clusters may emerge as a valuable and competitive alternative to existing heuristics for selecting the model complexity.

Acknowledgements We thank the anonymous reviewers for providing us with insightful comments and suggestions. We are grateful to C. Sigg, Y. Moh and T. Lange for a careful proof-reading of the manuscript.

Code and data for the work presented in this article are available from the first author's website, currently at http://www.inf.ethz.ch/~porbanz/ijcv06/.

References

1. C. E. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric estimation. *Annals of Statistics*, 2(6):1152–1174, 1974.
2. F. Bach and M. I. Jordan. Learning spectral clustering. In Sebastian Thrun, Lawrence K. Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing System 16*, pages 305–312. MIT Press, 2004.
3. J. Besag. On the statistical analysis of dirty pictures (with discussion). *J. R. Statist. Soc. B*, 48(3):259–302, 1986.
4. J. Besag, P. Green, D. Higdon, and K. Mengersen. Bayesian computation and stochastic systems. *Statistical Science*, 10(1):3–66, 1995.
5. D. M. Blei. *Probabilistic models for text and images*. PhD thesis, U. C. Berkeley, 2004.
6. D. M. Blei and M. I. Jordan. Variational methods for the Dirichlet process. *Bayesian Analysis*, 1(1):121–144, 2006.
7. J. Breckenridge. Replicating cluster analysis: Method, consistency and validity. *Multivariate Behavioral Research*, 24:147–161, 1989.

8. L. Devroye. *Non-uniform random variate generation*. Springer, 1986.
9. S. Dudoit and J. Fridyland. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3(7), 2002.
10. M. D. Escobar. Estimating normal means with a Dirichlet process prior. *Journal of the American Statistical Association*, 89(425):268–277, 1994.
11. T. S. Ferguson. A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2), 1973.
12. B. Fischer and J. M. Buhmann. Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):1411–1415, November 2003.
13. D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary detection by constrained optimization. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):609–628, 1990.
14. S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.
15. L. Hermes, T. Zöller, and J. M. Buhmann. Parametric distributional clustering for image segmentation. In *Computer Vision - ECCV '02*, volume 2352 of *LNCS*, pages 577–591. Springer, 2002.
16. T. Hofmann, J. Puzicha, and J. M. Buhmann. Unsupervised texture segmentation in a deterministic annealing framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):803–818, 1998.
17. S. Kotz, N. Balakrishnan, and N. L. Johnson. *Continuous Multivariate Distributions, Vol. 1*. John Wiley & Sons, second edition, 2000.
18. T. Lange, V. Roth, M. Braun, and J. M. Buhmann. Stability-based validation of clustering solutions. *Neural Computation*, 16(6):1299–1323, 2004.
19. E. L. Lehmann and J. P. Romano. *Testing statistical hypotheses*. Springer, third edition, 2005.
20. J. Lin. Divergence measures based on the shannon entropy. *IEEE Trans. Infor. Theory*, 37:145–151, 1991.
21. S. N. MacEachern. Estimating normal means with a conjugate style Dirichlet process prior. *Communications in Statistics: Simulation and Computation*, 23:727–741, 1994.
22. S. N. MacEachern. Computational methods for mixture of Dirichlet process models. In D. Dey, P. Müller, and D. Sinha, editors, *Practical nonparametric and semiparametric Bayesian statistics*, number 133 in Lecture Notes in Statistics, pages 23–43. Springer, 1998.
23. S. N. MacEachern and P. Müller. Efficient MCMC schemes for robust model extensions using encompassing Dirichlet process mixture models. In D. Rios Insua and F. Ruggeri, editors, *Robust Bayesian Analysis*, pages 295–315. Springer, 2000.
24. D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):530–549, May 2004.
25. J. D. McAuliffe, D. M. Blei, and M. I. Jordan. Nonparametric empirical Bayes for the Dirichlet process mixture model. Technical report, UC Berkeley, 2004.
26. J.-M. Morel and S. Solimini. *Variational Methods for Image Segmentation*. Birkhäuser, 1995.
27. P. Müller and F. A. Quintana. Nonparametric Bayesian data analysis. *Statistical Science*, 19(1):95–111, 2004.
28. R. M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9:249–265, 2000.
29. J. Puzicha, T. Hofmann, and J. M. Buhmann. Histogram clustering for unsupervised segmentation and image retrieval. *Pattern Recognition Letters*, 20:899–909, 1999.
30. J. Rissanen. A universal prior for integers and estimation by minimum description lenght. *Annals of Statistics*, 11:416–431, 1983.
31. C. P. Roberts. Mixtures of distributions: inference and estimation. In W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*, pages 441–464. Chapman & Hall, 1996.
32. V. Roth, J. Laub, K. Motoaki, and J. M. Buhmann. Optimal cluster preserving embedding of non-metric proximity data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1540–1551, December 2003.
33. C. Samson, L. Blanc-Féraud, G. Aubert, and J. Zerubia. A variational model for image classification and restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(5):460–472, May 2000.
34. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
35. Z. Tu and S.-C. Zhu. Image segmentation by data-driven Markov chain Monte Carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):657–673, May 2002.
36. S. G. Walker, P. Damien, P. W. Laud, and A. F. M. Smith. Bayesian nonparametric inference for random distributions and related functions. *Journal of the Royal Statistical Society B*, 61(3):485–527, 1999.
37. G. Winkler. *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. Springer, 2003.
38. H. Zaragoza, D. Hiemstra, D. Tipping, and S. Robertson. Bayesian extension to the language model for ad hoc information retrieval. In *Proc. SIGIR 2003*, 2003.