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

On the Smoothing of Multinomial Estimates using Liouville Mixture Models and Applications

Nizar Bouguila

Concordia Institute for Information Systems Engineering (CIISE), Concordia University, Montreal, QC H3G 1T7, Canada
bouguila@ciise.concordia.ca

Abstract There has been major progress in recent years in statistical model-based pattern recognition, data mining and knowledge discovery. In particular, generative models are widely used and are very reliable in terms of overall performance. Success of these models hinges on their ability to construct a representation which captures the underlying statistical distribution of data. In this article we focus on count data modeling. Indeed, this kind of data is naturally generated in many contexts and in different application domains. Usually, models based on the multinomial assumption are used in this case which may have several shortcomings especially in the case of high-dimensional sparse data. We propose then a principled approach to smooth multinomials using a mixture of Beta-Liouville distributions which is learned to reflect and model prior beliefs about multinomial parameters. Via both theoretical interpretations and experimental validations, we argue that the proposed smoothing model is general and flexible enough to allow accurate representation of count data.

Key words Liouville family of distributions, mixture models, smoothing, count data, generative discriminative learning, SVM, texture classification, object recognition.

1 Introduction

Many pattern recognition, computer vision and data mining problems can be formalized as data classification problems. Given a set of observations composed of input (for instance, the features representing a given image or text) and output variables (i.e. the class label), the main goal is to learn the relationship between the inputs and outputs in order to assign new observations into one of the data classes. Many classification approaches have been proposed in the past. Be it pattern recognition, image processing, computer vision or any other area, approaches to classification depend heavily on the type (ex. discrete, continuous, mixed, sequence, etc.) of the generated data that we would like to analyze [1–3]. Compared to count (or frequency) data, continuous data have received more attention by the pattern recognition community. Count data appear naturally, however, in many applications such as statistical natural language processing where the goal is generally to determine the likelihood of word combination from its frequency in a given training corpus [4], text classification which is mainly based on the frequency of words (i.e. bag of words representation) [5], images representation via visual words [6], texture classification using textons [7] or cooccurrence matrices [8], and protein classification [9]. The dominant approaches in these cases have been based on the multinomial assumption which may cause severe practical problems and unreliable model's parameters estimates especially when the data is sparse [1].

The main approach generally used to resolve these problems is to smooth (i.e. adjust) the multinomial estimates. Different authors have proposed different smoothing approaches in the past (see, for instance, [11–13]). The most popular approach, widely used by pattern recognition and machine learning researchers, consists of considering the Dirichlet distribution as prior to the multinomial parameters in order to exploit the conjugateness of the family of Dirichlet distributions to the multinomial. However, this approach has also its own drawbacks. Indeed, effective use of this smoothing method requires the choice of the smoothing parameters. When insufficient smoothing is done, the resulting parameters estimate can be too rough. On the other hand, excessive smoothing can compromise the modeling of the data. The choice of the smoothing parameters is generally left to the expert experience or prior opinion. A better approach is to be able to choose the amount of smoothing

[1] The problem of data sparseness is also known as the zero-frequency problem [10].

automatically from the data by letting the data speak for itself as shown in our previous works [14] where the prior knowledge, modeled via Dirichlet mixtures, and the statistical data are combined to estimate the smoothing parameters. Despite the fact that the consideration of the Dirichlet as a prior to the multinomial has dominated the research literature, recent studies have shown that this choice is inappropriate in several applications. Indeed, the Dirichlet has the unfortunate property that its covariance matrix is always negative [15–17] which may compromise the modeling capabilities in practical situations.

In this paper, we propose and discuss a new method to overcome the Dirichlet assumption's shortcomings. Our approach is based on the consideration of the Liouville family of distributions, which includes the Dirichlet as a special case, and from which we select the Beta-Liouville distribution. Like the Dirichlet, the Beta-Liouville is conjugate to the multinomial. But, it has a more general covariance structure than the Dirichlet which makes it more useful in real life applications. A mixture of Beta-Liouville distributions is taken then to describe our prior beliefs about the multinomial parameters with the ultimate goal to smooth the final estimates and to achieve good generalization. The choice of mixtures is justified by the fact that these models are a powerful probabilistic representation and their merits have been firmly established via intense research activity [18]. However, there are a certain number of problems to resolve when using them namely the accurate estimation of the parameters and the selection of the appropriate number of mixture components. Given our multinomial model and the Beta-Liouville mixture prior, we learn the model following an empirical Bayes approach by integrating out the multinomial parameters. Then, the prior hyperparameters are computed via a generalized expectation maximization (EM) algorithm which includes a gradient descent step. The selection of the optimal number of components is performed using the minimum description length (MDL) criterion.

We had several goals in carrying out this research. The first was to investigate the Liouville family of distributions as a prior to the multinomial family as explained above. In this work we additionally investigate the problem of count vectors classification via support vector machines (SVM). In this case classic kernels cannot be applied and the common recent practice is to consider the so-called hybrid generative discriminative approaches by generating SVM kernels from the generative model at hand. These approaches are attractive for allowing for integrating problem-specific background knowledge about the particularity of count data feature space and for efficiently combining the advantages of both discriminative and generative approaches and then getting the best of both worlds.

The rest of this paper is organized as follows. Section 2 describes the background for this work, briefly surveying the most widely used smoothing approaches and presents in sufficient details a new smoothing technique based on Beta-Liouville mixture models. In Section 3 we propose an approach to learn the smoothing parameters related to our mixture model. Section 4 is devoted to the experimental evaluation through a set of challenging applications namely texture classification and object recognition. Section 5 draws some conclusions and discusses issues for further research.

2 The Model

In this section, we shall discuss the problem of multinomial estimates smoothing within a unified framework and we shall propose a new smoothing approach based on Liouville mixture models.

2.1 Background

Let $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$ be a set of of frequency (or count) vectors representing N textual (or visual) documents where $\mathbf{X}_n = (X_{n1}, \ldots, X_{nV})$, X_{nv} denotes the frequency of feature (ex. word, visual word, etc) w_v occurrence in document n among the set of features (ex. vocabulary) $\mathcal{V} =< w_1, \ldots, w_V >$. V denotes the total number of features (ex. total number of words in the vocabulary). A given vector $\mathbf{X} \in \mathcal{X}$ is generally considered to have a multinomial distribution with parameters $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_{V-1})$:

$$p(\mathbf{X}|\boldsymbol{\pi}) \propto \prod_{v=1}^{V} \pi_v^{X_v} \tag{1}$$

where $\pi_v > 0$ denotes the probability of observing the particular v^{th} feature w_v in the document represented by $\mathbf{X}$, and $\pi_V = 1 - \sum_{v=1}^{V-1} \pi_v$. It is noteworthy that Eq. 1 is based actually on the well-known naive Bayes assumption for which a lot of work has been done in the past (see, for instance, [19,20]). The usual estimator of $(\pi_1, \ldots, \pi_V)$, commonly called the vector of observed proportions, is given by

$$\hat{\pi}_v = \frac{X_v}{\sum_{v=1}^{V} X_v} \quad v = 1, \ldots, V \tag{2}$$

Many studies, however, have shown that this estimator is "poor" especially in the case of large sparse data where the number of features is high. In this case the frequencies can be small [2] and then the observed proportions will tend to zero. The usual approach to tackle this problem is to smooth the estimates by adding a certain value to the different frequencies in the vector $\mathbf{X}$. For instance, the author in [21,22] suggests adding a $\frac{1}{2}$ count to every frequency. In an earlier work, he suggested adding

[2] It is easy to note from Eq. 1 that the presence of zero counts creates serious numerical problems.

a count of one to every frequency [23]. The same suggestions can be found in [24]. The authors in [25] have increased the counts by $\frac{1}{V}$, where V is the dimensionality of the vector. These heuristics can be viewed as special cases of a widely used general approach which consists on using prior information by assuming that $\boldsymbol{\pi}$ follows the conjugate Dirichlet distribution:

$$p(\boldsymbol{\pi}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{v=1}^{V}\alpha_v)}{\prod_{v=1}^{V}\Gamma(\alpha_v)}\prod_{v=1}^{V}\pi_v^{\alpha_v-1} \tag{3}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_V)$ is the vector of hyperparameters. Using this prior it is reasonably straightforward to show that [26,27,14]

$$\hat{\pi}_v = \frac{X_v + \alpha_v}{\sum_{v=1}^{V}(X_v+\alpha_v)} = r\frac{X_v}{\sum_{v=1}^{V}X_v} + (1-r)\frac{\alpha_v}{\sum_{v=1}^{V}\alpha_v} \tag{4}$$

where $r = \frac{\sum_{v=1}^{V}X_v}{\sum_{v=1}^{V}(X_v+\alpha_v)}$ and $\sum_{v=1}^{V}\alpha_v$ is generally called the "flattening constant" [28] [3]. We can see that Eq. 4 is reduced to Eq. 2 if $\alpha_v = 0$. Generally the Dirichlet is chosen to be symmetric (i.e. $\alpha_1 = \ldots = \alpha_V = \alpha$) (see, for instance, [28,29]) and several choices of α have been proposed and used. Examples of choices include $\alpha = 1$, called Laplace prior [30], $\alpha = \frac{1}{2}$, called Jeffreys prior [31], and $\alpha = \frac{1}{V}$ proposed by Perks in [32]. It is noteworthy that these priors coincide with the heuristics used in [23], [22] and [25], respectively.

Smoothing approaches based on Dirichlet priors have several main weaknesses. First, in spite of its flexibility and the fact that it is conjugate to the multinomial, the Dirichlet has a very restrictive negative covariance matrix which violates generally experimental observations [33]. Another restriction of the Dirichlet is that the variables with the same mean must have the same variance as shown in [34,35]. Third, generally the hyperparameters are taken independently from the sample according to a certain expert's knowledge. Finally, in most of the cases only one distribution is taken as a prior which may not be flexible enough for statistical modeling purposes. Hence, one would expect to be able to improve the smoothing of multinomial estimates by overcoming these shortcomings. In the following, we propose a novel smoothing approach that depends on the sample itself which is obviously more appropriate. The new approach is based on finite Liouville mixture model that is shown to be an appropriate choice as prior to the multinomial.

2.2 Liouville Mixture-Based Smoothing

2.2.1 Liouville Family of Distributions If a vector $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_{V-1})$ has a $(V-1)$-variate Liouville distribution with positive parameters $(\alpha_1, \ldots, \alpha_{V-1})$ and density generator $g(\cdot)$, then [36]

$$p(\boldsymbol{\pi}|\alpha_1, \ldots, \alpha_{V-1}) = g(u)\prod_{v=1}^{V-1}\frac{\pi_v^{\alpha_v-1}}{\Gamma(\alpha_v)} \tag{5}$$

where $u = \sum_{v=1}^{V-1}\pi_v < 1$, $\pi_v > 0, v = 1, \ldots, V-1$. The general moment function of a Liouville distribution is given by [36]

$$E(\pi_1^{r_1}\pi_2^{r_2}\ldots\pi_{V-1}^{r_{V-1}}) = E(U^r)\frac{\prod_{v=1}^{V-1}\Gamma(\alpha_v+r_v)\Gamma(\sum_{v=1}^{V-1}\alpha_v)}{\prod_{v=1}^{V-1}\Gamma(\alpha_v)\Gamma(\sum_{v=1}^{V-1}\alpha_v+r)} \tag{6}$$

And the mean, the variance and the covariance are given by

$$E(\pi_v) = E(U)\frac{\alpha_v}{\sum_{v=1}^{V-1}\alpha_v} \tag{7}$$

$$\begin{aligned} Var(\pi_v) = {} & E(U^2)\frac{\alpha_v(\alpha_v+1)}{\sum_{v=1}^{V-1}\alpha_v(\sum_{v=1}^{V-1}\alpha_v+1)} \\ & - E(\pi_v)^2\frac{\alpha_v^2}{(\sum_{v=1}^{V-1}\alpha_v)^2} \end{aligned} \tag{8}$$

$$Cov(\pi_l, \pi_k) = \frac{\alpha_l\alpha_k}{\sum_{v=1}^{V-1}\alpha_v}\left(\frac{E(U^2)}{\sum_{v=1}^{V-1}\alpha_v+1} - \frac{E(U)^2}{\sum_{v=1}^{V-1}\alpha_v}\right) \tag{9}$$

where $r = r_1 + \ldots + r_d$ and $E(U^r)$ is the r^{th} moment of a random variable $U \in [0,1]$ which follows a probability density function $f(\cdot)$ generally called generating density and related to the density generator $g(\cdot)$ by the following

$$g(u) = \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_v)}{u^{\sum_{v=1}^{V-1}\alpha_v-1}}f(u) \tag{10}$$

using this previous relation, the Liouville distribution of the second kind can be written also as follows

$$p(\boldsymbol{\pi}|\alpha_1, \ldots, \alpha_{V-1}) = \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_v)}{u^{\sum_{v=1}^{V-1}\alpha_v-1}}f(u)\prod_{v=1}^{V-1}\frac{\pi_v^{\alpha_v-1}}{\Gamma(\alpha_v)} \tag{11}$$

Note that, in contrast to the Dirichlet distribution [33, 37], the covariance of the Liouville can be positive or negative. A convenient choice as a distribution for u is the Beta distribution, which shapes are variable enough to allow for an approximation of almost any arbitrary distribution [38], with parameters α and β

$$f(u|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}u^{\alpha-1}(1-u)^{\beta-1} \tag{12}$$

and then

$$E(u) = \frac{\alpha}{\alpha+\beta} \quad E(u^2) = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)} \tag{13}$$

$$Var(u) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{14}$$

[3] Geometric interpretation of $\sum_{v=1}^{V}\alpha_v$ has been proposed in [29].

replacing Eq. 12 into Eq. 11, gives us the following

$$p(\boldsymbol{\pi}|\alpha_1,\ldots,\alpha_{V-1},\alpha,\beta) = \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_v)\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \times \prod_{v=1}^{V-1}\frac{\pi_v^{\alpha_v-1}}{\Gamma(\alpha_v)}(\sum_{v=1}^{V-1}\pi_v)^{\alpha-\sum_{v=1}^{V-1}\alpha_v}(1-\sum_{v=1}^{V-1}\pi_v)^{\beta-1} \quad (15)$$

which is called the Beta-Liouville distribution [36]. Using Eq. 13 and Eqs. 7, 8, and 9, we obtain the mean, the variance and the covariance of the Beta-Liouville distribution

$$E(\pi_v) = \frac{\alpha}{\alpha+\beta}\frac{\alpha_v}{\sum_{v=1}^{V-1}\alpha_v} \quad (16)$$

$$Var(\pi_v) = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}\frac{\alpha_v(\alpha_v+1)}{\sum_{v=1}^{V-1}\alpha_v(\sum_{v=1}^{V-1}\alpha_v+1)} - \frac{\alpha^2}{(\alpha+\beta)^2}\frac{\alpha_v^4}{(\sum_{v=1}^{V-1}\alpha_v)^4} \quad (17)$$

$$Cov(\pi_l,\pi_k) = \frac{\alpha_l\alpha_k}{\sum_{v=1}^{V-1}\alpha_v}\Bigg(-\frac{\alpha^2}{(\alpha+\beta)^2\sum_{v=1}^{V-1}\alpha_v} + \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)(\sum_{v=1}^{V-1}\alpha_v+1)}\Bigg) \quad (18)$$

Note that when the density generator has a Beta distribution with parameters $\sum_{v=1}^{V-1}\alpha_v$ and α_V:

$$f(u) = \frac{\Gamma(\alpha_V+\sum_{v=1}^{V-1}\alpha_v)}{\Gamma(\alpha_V)\Gamma(\sum_{v=1}^{V-1}\alpha_v)}u^{\sum_{v=1}^{V-1}\alpha_v-1}(1-u)^{\alpha_V} \quad (19)$$

Eq. 11 is reduced to the Dirichlet distribution with parameters $\alpha_1,\ldots,\alpha_V$. Thus, Liouville distribution includes the Dirichlet distribution as a special case.

2.2.2 Finite Beta-Liouville Mixture Model as a Multinomial Prior Let us assume that $\boldsymbol{\pi}$ follows a finite Beta-Liouville mixture:

$$p(\boldsymbol{\pi}|\Theta) = \sum_{j=1}^{M}p_j p(\boldsymbol{\pi}|\theta_j) \quad (20)$$

where $p(\boldsymbol{\pi}|\theta_j)$ is a Beta-Liouville distribution with parameters $\theta_j = (\alpha_{j1},\ldots,\alpha_{jV-1},\alpha_j,\beta_j)$, $\{p_j\}$ is the set of mixing parameters which are positive and sum to one, and $\Theta = \{\{p_j\},\{\theta_j\}\}$. Having this mixture as a prior, the joint distribution of $\mathbf{X}$ and $\boldsymbol{\pi}$ is

$$p(\mathbf{X},\boldsymbol{\pi}|\Theta) \propto \sum_{j=1}^{M}p_j\frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \times \prod_{v=1}^{V-1}\frac{\pi_v^{\alpha_{jv}+X_v-1}}{\Gamma(\alpha_{jv})}(\sum_{v=1}^{V-1}\pi_v)^{\alpha_j-\sum_{v=1}^{V-1}\alpha_{jv}} \times (1-\sum_{v=1}^{V-1}\pi_v)^{\beta_j+X_V-1} \quad (21)$$

then, it is easy to show that the marginal is

$$p(\mathbf{X}|\Theta) \propto \sum_{j=1}^{M}p_j\frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \quad (22)$$
$$\times \int_{\boldsymbol{\pi}}\Bigg[\prod_{v=1}^{V-1}\pi_v^{\alpha_{jv}+X_v-1}(\sum_{v=1}^{V-1}\pi_v)^{\alpha'_j-\sum_{v=1}^{V-1}\alpha'_{jv}} (1-\sum_{v=1}^{V-1}\pi_v)^{\beta'_j}\Bigg]d\boldsymbol{\pi}$$
$$= \sum_{j=1}^{M}p_j\Bigg[\frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \times \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}\Bigg]$$

where $\alpha'_{jv} = \alpha_{jv} + X_v$, $\alpha'_j = \alpha_j + \sum_{v=1}^{V-1}X_v$ and $\beta'_j = \beta_j + X_V$. Having the joint and marginal distributions in hand, we can show that π_v can be estimated as follows (See Appendix 1):

$$\hat{\pi}_v = \sum_{j=1}^{M}p(j|\mathbf{X})\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}} \quad v=1,\ldots,V-1 \quad (23)$$
$$\hat{\pi}_V = 1-\sum_{v=1}^{V-1}\hat{\pi}_v$$

where

$$p(j|\mathbf{X}) = \frac{p_j\frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})}\frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}{\sum_{j=1}^{M}p_j\frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})}\frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}} \quad (24)$$

and can be viewed as the posterior probability that the vector $\mathbf{X}$ will be assigned to cluster j when the marginal distribution $p(\mathbf{X}|\Theta)$ in Eq. 22 is taken as the parent distribution to model the data. Note that when $M=1$, Eq. 23 is reduced to

$$\hat{\pi}_v = \frac{\alpha'}{\alpha'+\beta'}\frac{\alpha'_v}{\sum_{v=1}^{V-1}\alpha'_v} \quad (25)$$

which is actually the mean of a Beta-Liouville distribution with parameters $(\alpha_1,\ldots,\alpha_{V-1},\alpha,\beta)$ according to Eq. 16. Finally, it is noteworthy that Eq. 25 is itself reduced to Eq. 4 if we take $\alpha = \sum_{v=1}^{V-1}\alpha_v$ and $\beta = \alpha_V$.

3 Model Learning

3.1 Parameters Estimation

According to Eq. 23 the smoothing of the multinomial parameters requires the estimation of $p(j|\mathbf{X})$, α_j, β_j and α_{jv}. In this section, we propose an approach to estimate

these quantities via the learning of the marginal distribution in Eq. 22 which is actually a mixture of distributions. It is noteworthy that we are making here an inferential statement (i.e. point estimation) about the hyperparameters of our mixture prior on the basis of data which is it is actually an empirical Bayes approach so named and developed in [39] and [40], respectively. The reader interested in the general empirical Bayesian theory is referred to accessible expositions in [41,42][4]. Traditionally, the estimation of finite mixture models has been based on the maximum likelihood approach:

$$\max_{\Theta} \left\{ p(\mathcal{X}|\Theta) = \prod_{n=1}^{N} p(\mathbf{X}_i|\Theta) \right\} \tag{26}$$

In some situations, however, maximizing the likelihood is not straightforward or appropriate. In our case, for instance, the maximization of the likelihood leads to the following estimate for the p_j parameters:

$$p_j = \frac{1}{N} \sum_{n=1}^{N} p(j|\mathbf{X}_N) \tag{27}$$

The maximization with respect to the α_j, β_j and α_{jv} parameters, however, involves the Gamma special function, $\Gamma(.)$, and by computing its derivatives other special functions such as the digamma (or the psi function) $\Psi(\alpha) = \frac{\partial \log \Gamma(\alpha)}{\partial \alpha}$ and trigamma $\Psi'(\alpha) = \frac{\partial \Psi(\alpha)}{\partial \alpha}$ occur which makes the parameters estimation intractable. A possible approach to overcome this problem is to optimize the following function

$$f(\mathcal{X}|\Theta) = \prod_{n=1}^{N} \prod_{v=1}^{V} \hat{\pi}_v^{X_{nv}} \tag{28}$$

$$= \prod_{n=1}^{N} \left[\left(\prod_{v=1}^{V-1} \left(\sum_{j=1}^{M} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j + \beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}} \right)^{X_{nv}} \right) \times \left(1 - \sum_{v=1}^{V-1} \sum_{j=1}^{M} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j + \beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}} \right)^{X_{nV}} \right]$$

that we obtain by substituting the estimates in Eq. 23 in Eq. 1 for all the $\mathbf{X}_n$. In order to estimate the α_j, β_j and α_{jv} parameters, we use a gradient descent method based on the first derivatives of $\log f(\mathcal{X}|\Theta)$ since the logarithm is a monotonic function. We will therefore compute these derivatives. By computing the first derivatives, we obtain (see Appendix 2)

$$\frac{\partial \log f(\mathcal{X}|\Theta)}{\partial \alpha_j} = \sum_{n=1}^{N} \left[X_{nv} \left(\sum_{v=1}^{V-1} F_{njv} \left(\frac{1}{\alpha'_j} - \frac{1}{\alpha'_j + \beta'_j} \right) \right) - X_{nV} F_{njV} \frac{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\beta'_j}{(\alpha'_j+\beta'_j)^2} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}}{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}} \right] \tag{29}$$

[4] In particular the authors in [41] provide interesting discussions about the difference between Bayesian and empirical Bayesian approaches.

$$\frac{\partial \log f(\mathcal{X}|\Theta)}{\partial \beta_j} = \sum_{n=1}^{N} \left[X_{nv} \left(\sum_{v=1}^{V-1} F_{njv} \left(-\frac{1}{\alpha'_j + \beta'_j} \right) \right) + X_{nV} F_{njV} \frac{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\alpha'_j}{(\alpha'_j+\beta'_j)^2} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}}{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}} \right] \tag{30}$$

$$\frac{\partial \log f(\mathcal{X}|\Theta)}{\partial \alpha_{jv}} = \sum_{n=1}^{N} \left[X_{nv} \left(F_{njv} \left(\frac{1}{\alpha'_{jv}} - \frac{1}{\sum_{v=1}^{V-1} \alpha'_{jv}} \right) \right) - X_{nV} F_{njV} \frac{p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{(\sum_{v=1}^{V-1} \alpha'_{jv}) - \alpha'_{jv}}{(\sum_{v=1}^{V-1} \alpha'_{jv})^2}}{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}} \right] \tag{31}$$

where

$$F_{njv} = \frac{p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}}{\sum_{j=1}^{M} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}} \quad v = 1, \ldots, V-1 \tag{32}$$

$$F_{njV} = \frac{\sum_{v=1}^{V-1} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}}{1 - \sum_{v=1}^{V-1} \sum_{j=1}^{M} p(j|\mathbf{X}_n) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jv}}{\sum_{v=1}^{V-1} \alpha'_{jv}}} \tag{33}$$

Having our derivatives in hand, the parameters are updated as follows

$$\theta_j^{new} = \theta_j^{old} - \gamma \frac{\partial \log f(\mathcal{X}|\Theta)}{\partial \theta_j} \tag{34}$$

where γ is a small number.

3.2 Complete Algorithm

The EM algorithm plays a uniquely important role in the estimation of mixture's parameters and has been widely studied in the past [43,44]. Thus, we shall consider it here. Two important problems when applying the EM framework for mixture models learning are the determination of the number of mixture components and the initialization of the parameters. Many proposals for the automatic selection of the number of clusters have been made over the years. Some of them are widely discussed in [18,45]. Here, we use the MDL criterion given by [46]

$$MDL(M) = -\log(p(\mathcal{X}|\Theta)) + \frac{1}{2} N_p \log(N) \tag{35}$$

where $N_p = M(D+3) - 1$ is the number of free parameters in the mixture model. Concerning the initialization, we use of the spherical K-means [47], rather than the well-known K-means with Euclidean distance. This choice is justified by the fact that count data lack a Euclidean structure since they are represented in terms of multinomial models for which the associated geometry is well-known to be spherical [48]. The spherical K-means is applied in conjunction with the method of moments based on the first and second moments of the Beta-Liouville distribution given by Eqs. 16 and 17:

Initialization Algorithm

1. INPUT: Count vectors $\mathbf{X}_n, n = 1, \ldots, N$ and number of clusters M.
2. Apply the spherical K-Means [47] algorithm to obtain the elements of each component.
3. Apply the method of moments based on Eqs. 16 and 17 for each component j.
4. Assign the data to clusters, assuming that the current model is correct.
5. If the current model and the new model are sufficiently close to each other, terminate, else go to step 2.

Having the initialization algorithm and the MDL criterion in hand, the complete smoothing parameters learning algorithm can be summarized as the following

Algorithm

For each candidate value of $M \in [M_{min}, M_{max}]$:

1. Apply the initialization algorithm.
2. E-Step: Compute the *posterior* probabilities $p(j|\mathbf{X}_n)$ using Eq. 24.
3. M-Step:
   (a) Update the p_j using Eq. 27.
   (b) Update the θ_j using Eq. 34.
4. Calculate the associated criterion MDL(M) using Eq. 35.
5. Select the optimal model M^* such that:

$$M^* = \arg\max_M MDL(M)$$

4 Experimental Results

4.1 Design of Experiments

The primary purpose of this section is to compare the proposed smoothing approach and outline its effectiveness when compared to previously proposed techniques namely Laplace smoothing, Jefferys smoothing, Perks smoothing and smoothing with Dirichlet mixtures. We empirically test our approach on several applications in order to show its general capability and to test its effectiveness in different situations. Our experiments are conducted within hybrid generative discriminative frameworks which have emerged as an efficient data representation and classification engine and have recently been studied by many researchers with great interest mainly as a way of exploiting the main advantages of both generative and discriminative approaches. In our frameworks the generative part consists of the multinomial model and the discriminative one is conducted via SVM by handling the count vectors as points on the multinomial manifold. Details of the SVM are well documented in [49], for instance, and will be omitted in the interests of brevity. The Achilles' heel of SVM is the need to choose an efficient kernel function to introduce non-linearity. Classic widely used SVM standard kernels, such as linear, polynomial, Gaussian and sigmoid, do not make use of explicit representations of domain-specific prior knowledge and ignore the geometric structure, defined by the Riemannian multinomial manifold of count data [50]. The main desire of hybrid models is to overcome some of the disadvantages associated with purely generative and discriminative methods. A natural way is to use the generative models (i.e. the multinomial in our case) to generate kernels. The main idea is to replace the kernel computation in the original data space by computation in the probability density functions space (i.e. the kernel becomes a measure of similarity between probability distributions) as the following $\mathcal{K}(\mathbf{X}, \mathbf{X}') \Rightarrow \mathcal{K}\big(p(\mathbf{X}|\boldsymbol{\pi}), p'(\mathbf{X}'|\boldsymbol{\pi}')\big)$. Examples of generated kernels that have been proposed to take into account the intrinsic geometric structure of count data include the negative Geodesic distance kernel (NGD) defined by [50]:

$$\mathcal{K}_{NGD}(\boldsymbol{\pi}, \boldsymbol{\pi}') = -2 \arccos\big(\sum_{v=1}^{V} \sqrt{\pi_v \pi'_v}\big) \quad (36)$$

Another approach is the Bhattacharyya kernel given by the following in the case of the multinomial [51]:

$$\mathcal{K}_B(\boldsymbol{\pi}, \boldsymbol{\pi}') = \sum_{v=1}^{V} \sqrt{\pi_v \pi'_v} \quad (37)$$

It is noteworthy that $K_B(\boldsymbol{\pi}, \boldsymbol{\pi}') = \cos(-\frac{1}{2} K_{NGD}(\boldsymbol{\pi}, \boldsymbol{\pi}'))$. Another notable work is the divergence kernel which has been proposed in [52] and given by the following in the case of the multinomial:

$$\mathcal{K}_D(\boldsymbol{\pi}, \boldsymbol{\pi}') = \exp\Big[-aJ\big(\boldsymbol{\pi}, \boldsymbol{\pi}'\big)\Big] \quad (38)$$

where and $a > 0$ is a kernel parameter included for numerical stability, and

$$\begin{aligned} J\big(\boldsymbol{\pi}, \boldsymbol{\pi}'\big) &= KL(\boldsymbol{\pi}, \boldsymbol{\pi}') + KL(\boldsymbol{\pi}', \boldsymbol{\pi}) \quad (39)\\ &= \sum_{v=1}^{V} \big(\pi_v \log(\frac{\pi_v}{\pi'_v}) + \pi'_v \log(\frac{\pi'_v}{\pi_v})\big) \end{aligned}$$

is the symmetric Kullback-Leibler (KL) divergence between the two multinomials $p(\mathbf{X}|\pi)$ and $p(\mathbf{X}|\pi')$. An alternative distance to the KL divergence, called the capacitory discriminant, has been proposed in [53] and is given by:

$$C(\boldsymbol{\pi}, \boldsymbol{\pi}') = KL(\boldsymbol{\pi}, \frac{1}{2}(\boldsymbol{\pi}+\boldsymbol{\pi}')) + KL(\boldsymbol{\pi}', \frac{1}{2}(\boldsymbol{\pi}+\boldsymbol{\pi}')) \quad (40)$$

Thus, we will investigate its use as a kernel in the same way as the KL divergence was used in [52]

$$\mathcal{K}_C(\boldsymbol{\pi}, \boldsymbol{\pi}') = \exp\Big[-aC(\boldsymbol{\pi}, \boldsymbol{\pi}')\Big] \quad (41)$$

The capacitory discriminant is related to the χ^2 distance by the following [53]

$$\frac{1}{2}\chi^2(\boldsymbol{\pi}, \boldsymbol{\pi}') \leq C(\boldsymbol{\pi}, \boldsymbol{\pi}') \leq \log 2\chi^2(\boldsymbol{\pi}, \boldsymbol{\pi}') \quad (42)$$

and $\lim_{\boldsymbol{\pi}\to\boldsymbol{\pi}'} C(\boldsymbol{\pi},\boldsymbol{\pi}') = \frac{1}{2}\chi^2(\boldsymbol{\pi},\boldsymbol{\pi}')$. It is noteworthy that the χ^2 has been used itself as an SVM kernel in [54]:

$$\mathcal{K}_{\chi^2}(\boldsymbol{\pi},\boldsymbol{\pi}') = \sum_{v=1}^{V} \frac{(\boldsymbol{\pi}-\boldsymbol{\pi}')^2}{\boldsymbol{\pi}+\boldsymbol{\pi}'} \tag{43}$$

In our experiments, we have employed the "one-vs-one" method for multi-class classification via the LIBSVM [5] implementation of SVM. In each of the applications that we will discuss in this section the used data sets were split into two groups: one for training and the other for testing. Then, the kernel parameters were selected by performing 10-fold cross-validation. After finding the best parameters, the SVM was trained using all the training data. For the smoothing parameters estimation using the algorithm in section 3.2, we set γ to 10^{-3}. In the following, we present our experimental results which concern statistical texture modeling and classification and object recognition and which main goal is to validate the proposed smoothing approach.

4.2 Statistical Texture Modeling and Classification

Texture modeling and classification plays an important role in remote sensing, computer vision, graphics and image processing and is a challenging task especially when the images representing textured materials are obtained under unknown viewing, camera pose and illumination conditions [55]. Several approaches have been proposed in the past to deal with this problem (see, for instance, [56]). Some techniques have been based on the characterization of the statistical nature of textures by the distribution of filter responses [57]. Recent studies have shown, however, that the use of filter banks is not necessary and that textures can be classified more accurately using only the joint distribution of intensity values over very compact neighborhoods [58]. For instance, the authors in [58] have used $n \times n$ pixel compact neighborhoods as image descriptors. Using this approach each texture pixel is described by an n^2-dimensional vector which represents the pixel intensities of its $n \times n$ square neighborhood. Then, a global vocabulary of V *textons* is constructed via the clustering of these n^2-dimensional descriptors, extracted from a texture training set, into V clusters (i.e. each cluster center is treated as a *texton*). Having this vocabulary in hand, each texture image can be represented as a V-dimensional vector of counts (i.e. the signature of the texture) containing the frequency of each *texton* in that image and then can be modeled by a multinomial distribution which parameters can be estimated and smoothed using our proposed approach.
In our experiments, we use the Columbia-Utrecht [59] data set previously considered in [58] and which is composed of 61 classes, with 205 images per class, which capture variation in illumination and pose of 61 different materials. We use a subset from this data set, as considered in [55], containing all the 61 classes with 92 images for each class. Figure 1 shows examples of images from the different classes in this data set. We consider also a second data set, called UIUCTex [60], which is composed of 25 texture classes with 40 images per class. The images in this data set are viewed under significant scale and viewpoint changes and include illumination changes, viewpoint-dependent appearance variations and non-rigid deformations. Figure 2 shows examples of images from the different classes in this data set. For each texture class, we select randomly 10 images from which we extract 7×7 pixel compact neighborhoods used as image descriptors (i.e. 49-dimensional vectors) and then clustered using the K-Means algorithm by considering 10 clusters (i.e. each class provides 10 textons). For the two data sets, we randomly select, 50 times, part of the images (20 and 46 images per class for the UIUCTex and CUReT data sets, respectively) for training and the rest for testing.
We perform our experiments using different kernels and smoothing approaches. Tables 1 and 2 show the average classification results for the CUReT and UIUCTex data sets, respectively. According to these tables we can see clearly that Beta-Liouville-based smoothing outperforms significantly (the differences are statistically significant according to a paired Student's t-test with 95% confidence; p-values between 0.001 and 0.024) the other approaches. We can see also that the classification accuracies when using different kernels are very close and that $\mathcal{K}_{NGD}$ performs slightly better. Figures 3 displays the average classification results as a function of the size of the neighborhood n and the size of the *textons* dictionary V for the CUReT set. According to this figure, we can see clearly that the best results were obtained for V ranging from 549 (i.e. 9 *textons* per class) to 671 (i.e. 11 *textons* per class) and for n ranging from 5 to 9.

4.3 Object Recognition Using Image Patches

A major goal in computer vision is the recognition of objects based on their visual appearance. This problem is challenging since the appearance of objects may change from one image to another due to many factors such as occlusion, noise and lighting conditions. Several approaches have been proposed in the past using both global and local visual features [61]. Recognition based on global features (e.g. texture, color) tends to suffer from partial occlusions or object deformation. Recognition based on local descriptors is known to be robust to appearance changes caused by imaging conditions and viewpoints. Examples of approaches based on local descriptors include [62–66]. In particular, an interesting ap-

[5] C-C. Chang and C-J. Lin, LIBSVM: A Library for Support Vector Machines. Available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

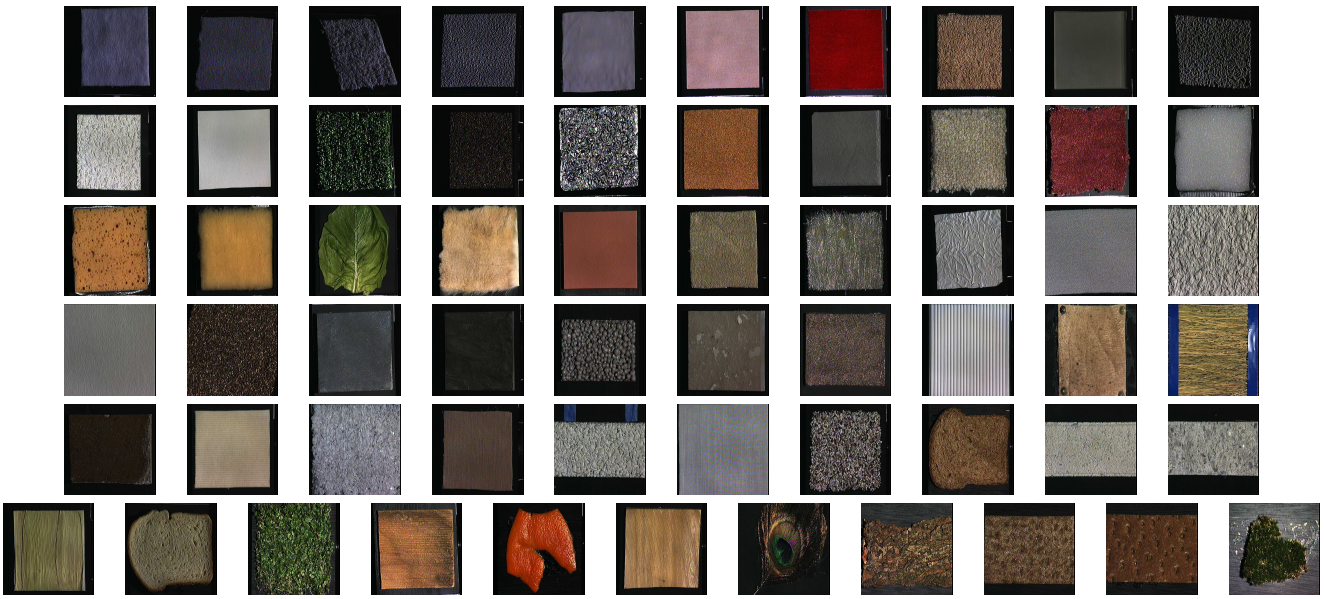

Fig. 1 Examples of images from the 61 different classes in the CUReT data set. Note that all images have been converted to monochrome in our experiments.

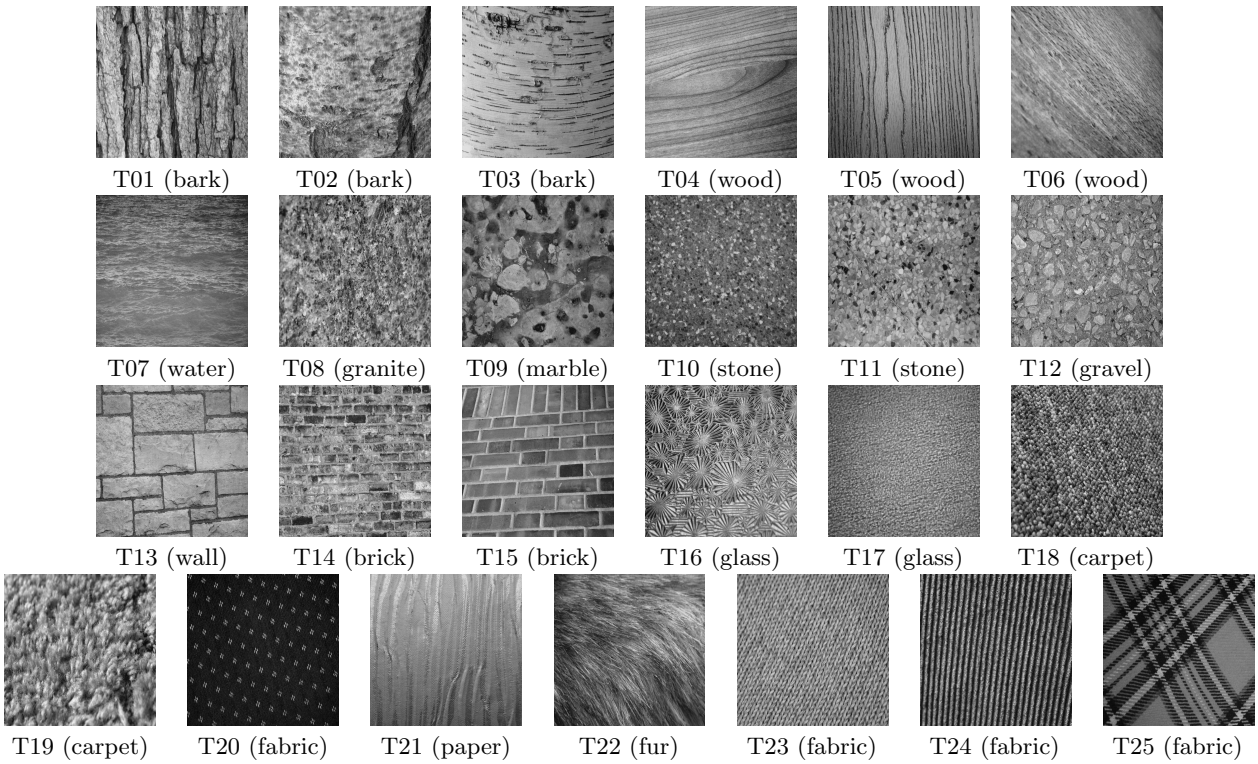

Fig. 2 Examples of images from the 25 different classes in the UIUCTex data set.

proach based on image patches, extracted at points of interest, has been proposed in [66]. This approach that we will consider here can be summarized as follows. First, up to 1000 square image patches are taken as image features and are extracted around interest points obtained using the approach described in [67]. The main idea of [67] is to extract salient points, where variations occur, regardless if they are corner-like or not. The extraction is based on Haar wavelet transform which is able to detect both local and global variations (i.e. a high wavelet coefficient in absolute value corresponds to a high variations). Moreover, 300 patches are added from a uniform grid of 15×20 cells that is projected onto the image. The main goal of these added patches is to take into

Table 1 Classification accuracies (%) for the CUReT data set using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	92.78 ± 0.61	92.71 ± 0.63	92.75 ± 0.72	92.76 ± 0.71	92.45 ± 0.74
Jefferys	92.66 ± 0.63	92.57 ± 0.60	92.59 ± 0.63	92.61 ± 0.55	92.58 ± 0.70
Perks	91.88 ± 0.81	91.79 ± 0.79	91.81 ± 0.75	91.84 ± 0.71	91.15 ± 0.67
Dirichlet	95.09 ± 0.91	94.98 ± 0.84	94.88 ± 0.80	94.90 ± 0.79	94.78 ± 0.64
Beta-Liouville	96.13 ± 0.87	96.07 ± 0.79	96.01 ± 0.81	96.10 ± 0.78	95.93 ± 0.64

Table 2 Classification accuracies (%) for the UIUCTex data set using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	93.05 ± 0.75	93.11 ± 0.77	93.09 ± 0.79	93.13 ± 0.80	92.95 ± 0.69
Jefferys	92.93 ± 0.72	92.94 ± 0.78	92.96 ± 0.80	92.98 ± 0.92	92.67 ± 0.62
Perks	91.90 ± 0.91	91.89 ± 0.88	91.74± 0.82	91.78 ± 0.87	91.59 ± 0.71
Dirichlet	95.31 ± 0.89	95.06 ± 0.71	95.10 ± 0.69	95.13 ± 0.70	94.91 ± 0.68
Beta-Liouville	96.27 ± 0.87	96.08 ± 0.82	96.11 ± 0.79	96.14 ± 0.78	95.97 ± 0.77

Table 3 Recognition rates (%) for the COIL-20 database using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	93.11 ± 0.74	93.06 ± 0.69	92.89 ± 0.61	92.73 ± 0.60	91.98 ± 0.57
Jefferys	93.03 ± 0.76	92.99 ± 0.88	92.90 ± 0.87	92.78 ± 0.79	91.77 ± 0.64
Perks	93.79 ± 0.82	93.68 ± 0.74	93.54± 0.91	93.37 ± 0.77	91.97 ± 0.86
Dirichlet	96.83 ± 0.77	96.58 ± 0.83	96.42 ± 0.86	96.17 ± 0.78	95.99 ± 0.71
Beta-Liouville	98.94± 1.03	98.38 ± 0.98	98.11 ± 0.95	97.94 ± 0.89	97.78 ± 0.87

Table 4 Recognition rates (%) for the COIL-100 database using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	89.90 ± 0.85	89.80 ± 0.87	89.71 ± 0.73	89.43 ± 0.86	89.05 ± 0.78
Jefferys	89.85 ± 0.82	89.74 ± 0.81	89.66 ± 0.78	89.45 ± 0.95	89.14 ± 0.72
Perks	90.99 ± 1.01	90.83 ± 0.98	90.60± 0.77	90.58 ± 0.72	90.17 ± 0.80
Dirichlet	94.91 ± 0.90	94.86 ± 0.89	94.77 ± 0.79	94.83 ± 0.76	93.69 ± 0.88
Beta-Liouville	97.73± 0.66	97.68 ± 0.68	97.51 ± 0.66	97.04 ± 0.81	96.76 ± 0.67

Table 5 Recognition rates (%) for cars category using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	92.55 ± 0.70	92.51 ± 0.67	92.49 ± 0.69	92.43 ± 0.78	91.98 ± 0.79
Jefferys	92.13 ± 0.62	92.04 ± 0.68	92.06 ± 0.68	92.08 ± 0.71	91.89 ± 0.72
Perks	92.16 ± 0.66	92.13 ± 0.56	92.14± 0.71	92.09 ± 0.77	91.95 ± 0.74
Dirichlet	95.11 ± 0.87	94.96 ± 0.80	95.01 ± 0.79	95.03 ± 0.72	94.81 ± 0.78
Beta-Liouville	97.19± 0.91	97.18 ± 0.92	96.98 ± 0.85	97.01 ± 0.88	96.89 ± 0.81

Table 6 Recognition rates (%) for leaves category using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	86.85 ± 0.79	86.61 ± 0.75	86.49 ± 0.68	86.53 ± 0.78	86.05 ± 0.76
Jefferys	86.80 ± 0.81	86.74 ± 0.79	86.76 ± 0.74	86.68 ± 0.72	86.15 ± 0.70
Perks	86.95 ± 0.77	86.88 ± 0.68	86.81± 0.78	86.77 ± 0.87	86.27 ± 0.80
Dirichlet	91.02 ± 0.58	89.98 ± 0.67	89.91 ± 0.69	89.93 ± 0.68	89.02 ± 0.70
Beta-Liouville	93.59± 0.50	93.48 ± 0.52	93.51 ± 0.56	93.45 ± 0.58	93.08 ± 0.67

account the homogeneity of objects. Having the patches in hand, a PCA dimensionality reduction is applied by keeping only 40 coefficients. The resulting data are then clustered with a Linde-Buzo-Gray algorithm [68] by considering the Euclidean distance. Thus, each image patch is assigned to a cluster which allows to represent each

Table 7 Recognition rates (%) for motorbikes category using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	89.96 ± 0.84	89.91 ± 0.86	89.89 ± 0.81	89.83 ± 0.82	98.99 ± 0.79
Jefferys	89.97 ± 0.87	89.86 ± 0.84	89.79 ± 0.81	89.55 ± 0.69	89.06 ± 0.76
Perks	90.01 ± 0.85	89.98 ± 0.83	89.84± 0.81	89.71 ± 0.68	89.05 ± 0.75
Dirichlet	93.03 ± 0.77	93.00 ± 0.72	93.02 ± 0.78	93.06 ± 0.79	92.89 ± 0.80
Beta-Liouville	95.44± 0.90	95.37 ± 0.84	95.31 ± 0.87	95.25 ± 0.83	95.09 ± 0.87

Table 8 Recognition rates (%) for faces category using different methods.

Method	$\mathcal{K}_{NGD}$	$\mathcal{K}_{B}$	$\mathcal{K}_{D}$	$\mathcal{K}_{C}$	$\mathcal{K}_{\chi^2}$
Laplace	89.95 ± 0.78	89.91 ± 0.67	89.90 ± 0.69	89.81 ± 0.58	89.09 ± 0.63
Jefferys	89.99 ± 0.79	89.95 ± 0.77	89.86 ± 0.82	89.88 ± 0.81	89.56 ± 0.71
Perks	90.09 ± 0.71	90.08 ± 0.78	90.03± 0.72	90.05 ± 0.78	89.90 ± 0.70
Dirichlet	93.88 ± 0.69	93.86 ± 0.61	93.81 ± 0.66	93.83 ± 0.71	93.51 ± 0.76
Beta-Liouville	96.03± 0.70	95.96 ± 0.72	95.91 ± 0.77	95.94 ± 0.74	95.29 ± 0.71

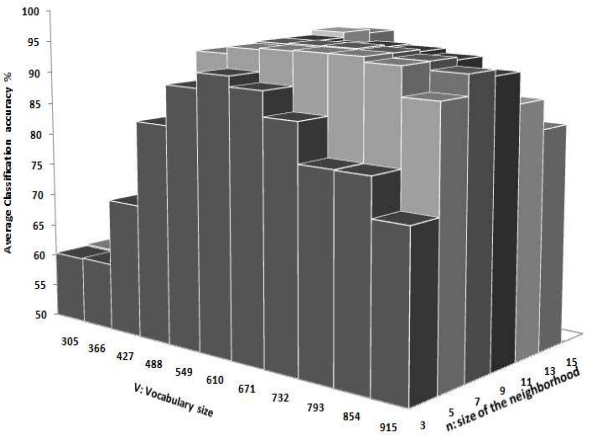

Fig. 3 Average classification accuracy, using Beta-Liouville smoothing and $\mathcal{K}_{NGD}$, as a function of the size of the neighborhood n and the size of the *textons* dictionary V for the CUReT data set.

image by a histogram of cluster frequencies (i.e. each entry in the histogram is created by counting how many patches belong to its associated cluster). As each image is now represented by a vector of counts, we can obviously assume that it is generated by a multinomial distribution which parameters can be estimated using our developed algorithm. In the following experiments we set the number of clusters to 512 (i.e. we use 512-dimensional count vectors to represent the images) and the results are averaged over 10 runs of the algorithm.
Two image databases are selected to evaluate our approach and are the Columbia Object libraries (COIL-20 and COIL-100). COIL-20 contains 1440 images of 20 objects (72 images per object) [69]. Each object is represented in the database by 72 images obtained by the rotation of the object through 360° in 5° steps. COIL-100 complete the COIL-20 with additional 80 objects (72 images per object) and consists then of 7200 images [70]. Figure 4 shows some of the 20 objects in the COIL-20 and figure 5 shows examples of images from the additional 80 objects. Both databases have been divided into disjuncts sets of 50% training and 50% test images. Tables 3 and 4 show the recognition rates for

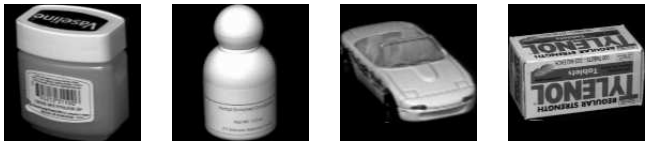

Fig. 4 Examples of images from the COIL-20 data set.

Fig. 5 Examples of images from the COIL-100 data set.

the COIL-20 and COIL-100 databases, respectively.
In another set of experiments we use 4 object categories which are cars, leaves, motorbikes and faces (see figure 6) from the Caltech database [71]. We use half of the images for testing and the rest for training. Moreover, following [71,64], we train the recognition against the background class in the Caltech database. Tables 5, 6, 7 and 8 report the recognition results for the cars, leaves, motorbikes and faces categories, respectively. According to our experimental results we can see clearly again

Fig. 6 Examples of images from the cars, faces, motorbikes and leaves categories in the Caltech database.

that Beta-Liouville-based smoothing outperforms significantly (the differences are statistically significant according to a paired Student's t-test with 95% confidence; p-values between 0.011 and 0.023) the other smoothing approaches.

5 Conclusion

Statistical analysis of count data plays a major role in several pattern recognition, computer vision and data mining applications. Many such approaches rely on the multinomial distribution which parameters are estimated and smoothed using ad hoc parameters or according to the consideration of Dirichlet priors. In many applications, Dirichlet priors are not realistic because they have a very restrictive negative covariance. In this paper, we have introduced and investigated a new prior to smooth multinomial estimates that is based on Liouville mixture models which include the Dirichlet as a special case. The advantage of our prior over the well-established and widely used technique is that it can be viewed as a more general smoothing technique. We have illustrated our results with many concrete examples and challenging applications namely texture classification and object recognition where the proposed smoothing technique is shown to offer improvement over widely used other approaches. In particular, we have shown that the use of the smoothed parameters to generate data-based SVM kernels provides excellent classification results. The methods proposed in this article can be applied to other problems besides image processing and computer vision since count data are naturally generated in many other research areas such as Bioinformatics, natural language processing, text mining and information retrieval. Promising future works could be devoted to the consideration of a nonparametric Bayesian approach, similar to the one proposed in [72], for the learning of the proposed model or the integration of a feature selection component, like the one proposed in [73], within the proposed smoothing framework to improve further the smoothing quality.

Acknowledgment

The completion of this research was made possible thanks to the Natural Sciences and Engineering Research Council of Canada (NSERC).

Appendix 1: Proof of Equation 23

We start by computing the posterior distribution:

$$p(\boldsymbol{\pi}|\mathbf{X},\Theta) = \frac{p(\mathbf{X},\boldsymbol{\pi}|\Theta)}{p(\mathbf{X}|\Theta)} \tag{44}$$

$$= \frac{1}{\sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)} \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}$$
$$\times \sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)}$$
$$\times \prod_{v=1}^{V-1} \frac{\pi_v^{\alpha_{jv}+X_v-1}}{\Gamma(\alpha_{jv})} (\sum_{v=1}^{V-1}\pi_v)^{\alpha_j-\sum_{v=1}^{V-1}\alpha_{jv}}$$
$$\times (1-\sum_{v=1}^{V-1}\pi_v)^{\beta_j+X_V-1}$$

In order to find the estimate of a certain parameter the $\pi_l, l = 1,\ldots,V$ when a Beta-Liouville mixture is taken as a prior, we have to compute the expectation π_v according to the previous posterior distribution:

$$\hat{\pi}_l = \int_{\pi_l} \pi_l p(\boldsymbol{\pi}|\mathbf{X},\Theta) d\pi_l$$
$$= \frac{1}{\sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}$$
$$\times \sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})}$$
$$\times \int_{\pi_l} \Bigg[\prod_{v=1}^{V-1} \pi_v^{\alpha_{jv}+\delta(v=l)+X_v-1}$$
$$(\sum_{v=1}^{V-1}\pi_v)^{\alpha_j-\sum_{v=1}^{V-1}\alpha_{jv}} (1-\sum_{v=1}^{V-1}\pi_v)^{\beta_j+X_V-1} d\pi_l \Bigg]$$

$$= \frac{1}{\sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}$$
$$\times \sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})}$$
$$\times \frac{\Gamma(\alpha'_j+1)\Gamma(\beta'_j)\Gamma(\alpha'_{jl}+1)\prod_{v=1,v\neq l}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv}+1)\Gamma(\alpha'_j+\beta'_j+1)}$$

where $\delta(v=l)=1$ if $v=1$ and 0, otherwise. Since $\Gamma(x+1)=x\Gamma(x)$, we obtain

$$\hat{\pi}_l = \sum_{j=1}^{M} p(j|\mathbf{X}) \frac{\alpha'_j}{\alpha'_j+\beta'_j} \frac{\alpha'_{jl}}{\sum_{v=1}^{V-1}\alpha'_{jv}}$$

where

$$p(j|\mathbf{X}) = \frac{p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}{\sum_{j=1}^{M} p_j \frac{\Gamma(\sum_{v=1}^{V-1}\alpha_{jv})\Gamma(\alpha_j+\beta_j)}{\Gamma(\alpha_j)\Gamma(\beta_j)\prod_{v=1}^{V-1}\Gamma(\alpha_{jv})} \frac{\Gamma(\alpha'_j)\Gamma(\beta'_j)\prod_{v=1}^{V-1}\Gamma(\alpha'_{jv})}{\Gamma(\sum_{v=1}^{V-1}\alpha'_{jv})\Gamma(\alpha'_j+\beta'_j)}}$$

Appendix 2: Proof of Equations 29, 30 and 31

We have

$$\begin{aligned}
&\log f(\mathcal{X}|\Theta)\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}\log\left(\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)\right)\right.\\
&\left.+\ X_{nV}\log\left(1-\sum_{v=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)\right]
\end{aligned}$$

Thus,

$$\begin{aligned}
&\frac{\partial\log f(\mathcal{X}|\Theta)}{\partial\alpha_j}\\
&= \sum_{n=1}^{N}\left[X_{nv}\frac{\partial}{\partial\alpha_j}\left(\sum_{v=1}^{V-1}\log\left(\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)\right)\right.\\
&\left.+\ X_{nV}\frac{\partial}{\partial\alpha_j}\log\left(1-\sum_{v=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}\frac{\frac{\partial}{\partial\alpha_j}\left(\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right)\right.\\
&\left.+\ X_{nV}\frac{\frac{\partial}{\partial\alpha_j}\left(1-\sum_{v=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{1-\sum_{l=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}F_{njv}\frac{\partial}{\partial\alpha_j}\log(p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right)\right.\\
&\left.-\ X_{nV}\frac{\frac{\partial}{\partial\alpha_j}\left(\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{1-\sum_{l=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}F_{njv}\frac{\partial}{\partial\alpha_j}\log(\frac{\alpha'_j}{\alpha'_j+\beta'_j})\right)\right.\\
&\left.-\ X_{nV}F_{njV}\frac{\partial}{\partial\alpha_j}\log(\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}F_{njv}(\frac{1}{\alpha'_j}-\frac{1}{\alpha'_j+\beta'_j})\right)\right.\\
&\left.-\ X_{nV}F_{njV}\frac{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\beta'_j}{(\alpha'_j+\beta'_j)^2}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]
\end{aligned}$$

where

$$F_{njv} = \frac{p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}{\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}$$

$$F_{njV} = \frac{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}{1-\sum_{v=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}$$

Using the same development we can easily show that

$$\begin{aligned}
\frac{\partial\log f(\mathcal{X}|\Theta)}{\partial\beta_j} &= \sum_{n=1}^{N}\left[X_{nv}\left(\sum_{v=1}^{V-1}F_{njv}(-\frac{1}{\alpha'_j+\beta'_j})\right)\right.\\
&\left.+\ X_{nV}F_{njV}\frac{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{(\alpha'_j+\beta'_j)^2}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]
\end{aligned}$$

and that

$$\begin{aligned}
&\frac{\partial\log f(\mathcal{X}|\Theta)}{\partial\alpha_{jv}}\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(\frac{\frac{\partial}{\partial\alpha_{jv}}\left(\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right)\right.\\
&\left.+\ X_{nV}\frac{\frac{\partial}{\partial\alpha_{jv}}\left(1-\sum_{v=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{1-\sum_{l=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(F_{njv}\frac{\partial}{\partial\alpha_{jv}}\log(p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right)\right.\\
&\left.-\ X_{nV}\frac{\frac{\partial}{\partial\alpha_{jv}}\left(\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}\right)}{1-\sum_{l=1}^{V-1}\sum_{j=1}^{M}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(F_{njv}\frac{\partial}{\partial\alpha_{jv}}\log(\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right)\right.\\
&\left.-\ X_{nV}F_{njV}\frac{\partial}{\partial\alpha_{jv}}\log(\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right]\\
&= \sum_{n=1}^{N}\left[X_{nv}\left(F_{njv}(\frac{1}{\alpha'_{jv}}-\frac{1}{\sum_{v=1}^{V-1}\alpha'_{jv}})\right)\right.\\
&\left.-\ X_{nV}F_{njV}\frac{p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{(\sum_{v=1}^{V-1}\alpha'_{jv})-\alpha'_{jv}}{(\sum_{v=1}^{V-1}\alpha'_{jv})^2}}{\sum_{v=1}^{V-1}p(j|\mathbf{X}_n)\frac{\alpha'_j}{\alpha'_j+\beta'_j}\frac{\alpha'_{jv}}{\sum_{v=1}^{V-1}\alpha'_{jv}}}\right]
\end{aligned}$$

References

1. C. E. Brodley and P. Smyth. Applying Classification Algorithms in Practice. *Statistics and Computing*, 7(1):45–56, 1997.
2. N. Bouguila, D. Ziou and J. Vaillancourt. Novel Mixtures Based on the Dirichlet Distribution: Application to Data and Image Classification. In Petra Perner and Azriel Rosenfeld, editors, *Machine Learning and Data Mining in Pattern Recognition (MLDM)*, pages 172–181. Springer, LNAI2734, 2003.
3. P. A. Vijaya, M. N. Murty and D. K. Subramanian. Efficient Median Based Clustering and Classification Techniques for Protein Sequences. *Pattern Analysis and Applications*, 9(2-3):243–255, 2006.

4. I. Dagan, L. Lee and F. C. N. Perrira. Similarity-Based Nodels of Word Cooccurrence Probabilities. *Machine Learning*, 34(1-3):43–69, 1999.
5. S. Scott and S. Matwin. Feature Engineering for Text Classification. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 379–388, 1999.
6. G. Csurka, C. R. Dance, L. Fan, J. Willamowski and C. Bray. Visual Categorization with Bags of Keypoints. In *Workshop on Statistical Learning in Computer Vision, 8th European Conference on Computer Vision (ECCV)*, 2004.
7. T. Leung and J. Malik. Representing and Recognizing the Visual Appearance of Materials Using Three-Dimensional Textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.
8. N. Bouguila and W. ElGuebaly. Discrete Data Clustering Using Finite Mixture Models. *Pattern Recognition*, 42(1):33–42, 2009.
9. B. Y. M. Cheng, J. G. Carbonell and J. Klein-Seetharaman. Protein Classification Based on Text Document Classification Techniques. *Proteins: Structure, Function, and Bioinformatics*, 58:955–970, 2005.
10. I. H. Witten and T. C. Bell. The Zero-Frequency Problem: Estimating The Probabilities Of Novel Events In Adaptive Text Compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094, 1991.
11. S. E. Fienberg and P. W. Holland. Simultaneous Estimation of Multinomial Cell Probabilities. *Journal of the American Statistical Association*, 68(343):683–691, 1973.
12. P. Hall and D. M. Titterington. On Smoothing Sparse Multinomial Data. *Australian Journal of Statistics*, 29(1):19–37, 1987.
13. J. S. Simonoff. Smoothing Categorical Data. *Journal of Statistical Planning and Inference*, 47:41–69, 1995.
14. N. Bouguila and D. Ziou. Unsupervised Learning of a Finite Discrete Mixture: Applications to Texture Modeling and Image Databases Summarization. *Journal of Visual Communication and Image Representation*, 18(4):295–309, 2007.
15. N. Bouguila and D. Ziou. A Powerful Finite Mixture Model Based on the Generalized Dirichlet Distribution: Unsupervised Learning and Applications. In *Proc. of the 17th International Conference on Pattern Recognition (ICPR)*, pages 280–283, 2004.
16. N. Bouguila and D. Ziou. Dirichlet-Based Probability Model Applied to Human Skin Detection. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 521–524, 2004.
17. N. Bouguila, D. Ziou and R. I. Hammoud. On Bayesian Analysis of a Finite Generalized Dirichlet Mixture Via a Metropolis-within-Gibbs Sampling. *Pattern Analysis and Applications*, 12(2):151–166, 2009.
18. G.J. McLachlan and D. Peel. *Finite Mixture Models*. New York: Wiley, 2000.
19. Z. Hoare. Landscapes of Naive Bayes Classifiers. *Pattern Analysis and Applications*, 11(1):59–72, 2008.
20. J. Andrés-Ferrer and A. Juan. Constrained Domain Maximum Likelihood Estimation for Naive Bayes Text Classification. *Pattern Analysis and Applications*, 13(2):189–196, 2010.
21. L. A. Goodman. The Multivariate Analysis of Qualitative Data: Interactions among Multiple Classifications. *Journal of the American Statistical Association*, 65(329):226–256, 1970.
22. L. A. Goodman. The Analysis of Multidimensional Contingency Tables: Stepwise Procedures and Direct Estimation Methods for Building Models for Multiple Classifications. *Technometrics*, 13(1):33–61, 1971.
23. L. A. Goodman. Interactions in Multidimensional Contingency Tables. *The Annals of Mathematical Statistics*, 35(2):632–646, 1964.
24. J. J. Gart and J. R. Zweifel. On the Bias of Various Estimators of the Logit and its Variance with Application to Quantal Biossay. *Biometrika*, 54(1/2):181–187, 1967.
25. J. E. Grizzle, C. F. Starmer and G. G. Koch. Analysis of Categorical Data by Linear Models. *Biometrics*, 25(3):489–504, 1969.
26. N. Bouguila and D. Ziou. Improving Content Based Image Retrieval Systems Using Finite Multinomial Dirichlet Mixture. In *Proc. of the IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pages 23–32, 2004.
27. N. Bouguila. Spatial Color Image Databases Summarization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Volume 1*, pages 953–956, Honolulu, HI, USA, 2007.
28. I. J. Good. A Bayesian Significance Test for Multinomial Distribution (with Discussion). *Journal of the Royal Statistical Society B*, 29(3):399–431, 1967.
29. S. E. Fienberg. On the Choice of Flattening Constants for Estimating Multinomial Probabilities. *Journal of Multivariate Analysis*, 2(1):127–134, 1972.
30. G. J. Lidstone. Note on the General Case of the Bayes-Laplace Formula for Inductive or a posteriori probabilities. *Trans. Fac. Actuar.*, 8:182–192, 1920.
31. J. Jeffreys. *Theory of Probability*. Clarendon Press, Oxford, third edition, 1961.
32. W. Perks. Some Observations on Inverse Probability Including a New Indifference Rule (with discussion). *J. Inst. Actuar.*, 73:285–334, 1947.
33. N. Bouguila, D. Ziou and J. Vaillancourt. Unsupervised Learning of a Finite Mixture Model Based on the Dirichlet Distribution and its Application. *IEEE Transactions on Image Processing*, 13(11):1533–1543, 2004.
34. R. H. Lochner. A Generalized Dirichlet Distribution in Bayesian Life Testing. *Journal of the Royal Statistical Society, B*, 37:103–113, 1975.
35. N. Bouguila and W. ElGuebaly. On Discrete Data Clustering. In *Proc. of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), LNCS 5012*, pages 503–510, Osaka, Japan, 2008. Springer.
36. K. T. Fang, S. Kotz and K. W. Ng. *Symmetric Multivariate and Related Distributions*. Chapman and Hall, New York, 1990.
37. N. Bouguila and D. Ziou. Using unsupervised learning of a finite Dirichlet mixture model to improve pattern recognition applications. *Pattern Recognition Letters*, 26(12):1916–1925, 2005.
38. N. Bouguila, D. Ziou and E. Monga. Practical Bayesian Estimation of a Finite Beta Mixture Through Gibbs Sampling and its Applications. *Statistics and Computing*, 16(2):215–225, 2006.
39. H. E. Robbins. An Empirical Bayes Approach to Statistics. In J. Neyman, editor, *Proc. of the Third Berkeley*

Symposimum on Mathematical Statistics and Probability, Vol. 1, pages 157–163, 1956.
40. H. E. Robbins. The Empirical Bayes Approach to Statistics. *The Annals of Mathematical Statistics*, 35(1):1–20, 1964.
41. J. J. Deely and D. V. Lindley. Bayes Empirical Bayes. *Journal of the American Statistical Association*, 76(376):833–841, 1981.
42. B. P. Carlin and T. A. Louis. *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall/CRC, second edition, 2000.
43. J. G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley, 1997.
44. T. Hu and S. Y. Sung. Clustering Spatial Data with a Hybrid EM Approach. *Pattern Analysis and Applications*, 8(1-2):139–148, 2005.
45. N. Bouguila and D. Ziou. High-Dimensional Unsupervised Selection and Estimation of a Finite Generalized Dirichlet Mixture Model Based on Minimum Message Length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1716–1731, 2007.
46. J. Rissanen. Modeling by Shortest Data Description. *Automatica*, 14:465–471, 1978.
47. I. S. Dhillon and D. S. Modha. Concept Decompositions for Large Sparse Text Data Using Clustering. *Machine Learning*, 42(1-2):143–175, 2001.
48. G. Lebanon and J. Lafferty. Hyperplane Margin Classifiers on the Multinomial Manifold. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 66–73, 2004.
49. V.N. Vapnik. *Statistical Learning Theory*. New York: John Wiley and Sons, 1998.
50. D. Zhang, X. Chen and W. S. Lee. Text Classification with Kernels on the Multinomial Manifold. In *Proc. of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 266–273, 2005.
51. T. Jebara, R. Kondor and A. Howard. Probability Product Kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.
52. P. J. Moreno, P. P. Ho and N. Vasconcelos. A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2003.
53. F. Topsoe. Some Inequalities for Information Divergence and Related Measures of Discrimination. *IEEE Transactions on Information Theory*, 46(4):1602–1609, 2000.
54. O. Chapelle, P. Haffner and V. N. Vapnik. Support Vector Machines for Histogram-Based Image Classification. *IEEE Transactions on Neural Networks*, 10(5):1055–1064, 1999.
55. M. Varma and A. Zisserman. Classifying Images of Materials: Achieving Viewpoint and Illumination Independence. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 255–271, 2002.
56. P. M. Szczypiński, M. Strzelecki, A. Materka and A. Klepaczko. MaZdaA Software Package for Image Texture Analysis. *Computer Methods and Programs in Biomedicine*, 94(1):66–76, 2009.
57. S. C. Zhu, Y. Wu and D. Mumford. Filters, Random Fields and Maximum Entropy (FRAME): Towards a Unified Theory for Texture Modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.
58. M. Varma and A. Zisserman. A Statistical Approach to Material Classification Using Image Patch Exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(11):2032–2047, 2009.
59. K. J. Dana, B. van Ginneken, S. K. Nayar and J. J. Koenderink. Reflectance and Texture of Real-World Surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999.
60. S. Lazebnik, C. Schmid and J. Ponce. A Sparse Texture Representation Using Local Affine Regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, 2005.
61. M. Grzegorzek. A System for 3D Texture-Based Probabilistic Object Recognition and its Applications. *Pattern Analysis and Applications*, 13(3):333–348, 2010.
62. B. Schiele and A. Pentland. Probabilistic Object Recognition and Localization. In *Proc. of the IEEE International Conference on Computer Vision (ICCV)*, pages 177–182, 1999.
63. L. Amsaleg and P. Gros. Content-based Retrieval Using Local Descriptors: Problems and Issues from a Database Perspective. *Pattern Analysis and Applications*, 4(2-3):108–124, 2001.
64. B. Caputo, C. Wallraven and M-E. Nilsback. Object Categorization via Local Kernels. In *Proc. of the 17th International Conference on Pattern Recognition (ICPR)*, pages 132–135, 2004.
65. S. Lyu. Mercer Kernels for Object Recognition with Local Features. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 223–229, 2005.
66. T. Deselaers, D. Keysers and H. Ney. Discriminative Training for Object Recognition Using Image Patches. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 157–162, 2005.
67. E. Loupias, N. Sebe, S. Bres and J. Jolion. Wavelet-Based Salient Points for Image Retrieval. In *Proc. of the IEEE International Conference on Image Processing (ICIP)*, pages 518–521, 2000.
68. Y. Linde, A. Buzo,and R. M. Gray. An Algorithm for Vector Quantization Design. *IEEE Tranactions on Communications*, 28:84–95, 1980.
69. S. A. Nene, S. K. Nayar and H. Murase. Columbia Object Image Library (COIL-20). Technical Report CUCS-005-96, Columbia University, 1996.
70. S. A. Nene, S. K. Nayar and H. Murase. Columbia Object Image Library (COIL-100). Technical Report CUCS-006-96, Columbia University, 1996.
71. M. Weber, M. Welling and P. Perona. Unsupervised Learning of Object Models and Recognition. In *Proc. of the European Conference on Computer Vision (ECCV)*, pages 18–32, 2000.
72. N. Bouguila and D.Ziou. A Dirichlet Process Mixture of Generalized Dirichlet Distributions for Proportional Data Modeling. *IEEE Transactions on Neural Networks*, 21(1):107–122, 2010.
73. N. Bouguila. A Model-Based Approach for Discrete Data Clustering and Feature Weighting Using MAP and Stochastic Complexity. *IEEE Transactions on Knowledge and Data Engineering*, 21(12):1649–1664, 2009.