Statistical Spatial Color Information Modeling in Images and Applications

Walid Elguebaly

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Quality Systems Engineering) at

Concordia University

Montréal, Québec, Canada

June 2008

Your file Votre référence
ISBN: 978-0-494-42527-5
Our file Notre référence
ISBN: 978-0-494-42527-5

Abstract

Statistical Spatial Color Information Modeling in Images and Applications

Walid Elguebaly

Image processing, among its vast applications, has proven particular efficiency in quality control systems. Quality control systems such as the ones in the food industry, fruits and meat industries, pharmaceutic, and hardness testing are highly dependent on the accuracy of the algorithms used to extract image feature vectors and process them. Thus, the need to build better quality systems is tied to the progress in the field of image processing.

Color histograms have been widely and successfully used in many computer vision and image processing applications. However, they do not include any spatial information. We propose statistical models to integrate both color and spatial information. Our first model is based on finite mixture models which have been applied to different computer vision, image processing and pattern recognition tasks. The majority of the work done concerning finite mixture models has focused on mixtures for continuous data. However, many applications involve and generate discrete data for which discrete mixtures are better suited. In this thesis, we investigate the problem of discrete data modeling using finite mixture models. We propose a novel, well motivated mixture that we call a multinomial generalized Dirichlet mixture.

Our second model is based on finite multiple-Bernoulli mixtures. For the estimation of the model's parameters, we use a maximum a posteriori (MAP) approach through deterministic annealing expectation maximization (DAEM). Smoothing priors to the components parameters are introduced to stabilize the estimation. The selection of the number of clusters is based on stochastic complexity.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Dr. Nizar Bouguila, for his continuous support and encouragement throughout my graduate studies. When I started my masters I knew very little about image processing and yet he supported and guided me until this day. His mentorship was essential to the completion of this thesis and my graduation.

Finally, I would also like to thank my family members for their unconditional support during this process.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Human vision is the primary source of information about the outside world, and it plays a major role in human perception. With all the advances in technology, our dependence on vision and images has become increasingly stronger. Today, digital image processing is in the heart of most areas of science since image collections exist to serve many purposes including: medicine, art, geography,...etc. Although image processing has been applied in different areas, it has proven particular efficiency in quality control systems.

In the food industry, for instance, machine vision methods were used to obtain 3D measurements of the different types and sizes of loaves for bread production [1]. For the tea process, image processing techniques were used to monitor the tea leaf color for fermentation, to inspect the tea taster, and to estimate the quality of the tea by different tea planters. In the fruits and meat industries, image processing methods were implemented to detect numerous defects in shape, color, size, and texture of the fruits [2, 3], and to grade the quality of the meat [4].

Image processing was also applied in other industries like in pharmaceutic to find out missing tablets in the inspection of tablet strips [5], in hardness testing by PATNI computer systems to measure metal diameter, and in fish food to repair coating quality in can ends of metal containers [6]. Thus, the significance of the ability to classify, search, and retrieve images from an image collection can't be overemphasized.

Unfortunately, indexing and classifying images has many difficulties in comparison to text. For instance, images do not satisfy the requirements of a language; they do not have a clear purpose or meaning that might help in classifying them, nor can they be searched by keywords to extract relevant needed information and this is due to the fact that keywords are subjective and can mean different things to different people. Researchers around the world tried to overcome these problems and proposed many

methods and algorithms to better index and classify the images.

One of the very first attempts to index color images was the color histogram [7, 8]. The histogram is an effective yet simple method of indexing based on the color feature of the images; it counts the number of occurrence of each color in the image. The major drawback in the color histogram is its lack of spatial information. In an attempt to overcome this problem, Pass et al. introduced the histogram refinements [9–12] approach which uses the color coherence vectors. Because of its high computational cost and to further improve the indexing of images, Huang et al. introduced the color correlogram [13, 14]. The correlogram expresses the spatial correlation of color changes with distances in the image. Another approach introduced by Lee et al. is the spatial color descriptor for image retrieval and video segmentation [15] which outperforms the histogram, its refinements, and the autocorrelogram.

This thesis addresses statistical modeling of spatial color information in images. The following background material is presented to provide context for this work.

1.1 Histograms

As an image feature vector, color histograms can be used in many image related applications including content-based image retrieval [16–18], object indexing and localization [7], image subregion querying [7], video cut detection [19], as well as some other video processing applications. Color histograms have several advantages like easiness of computation, insensitivity to small changes and partial occlusions, and storage efficiency by only taking an $O(n^2)$ space [7, 8, 13].

Let I be an image of pixels $p(x, y)$, each pixel has a certain color defined using a color space. Different color spaces have been proposed. In the following, we give some examples:

1.1.1 Color Spaces

❖RGB color space

RGB is the most commonly used color space, and it is composed of primary colors: Red, Green, and Blue which are considered additive primaries. The representation of the RGB color space is a cube in which the diagonal from (0,0,0) "black" to (1,1,1) "white" represents the grayscale.

❖HSV color space

The HSV stands for: Hue, saturation, and value (intensity). The hue and saturation components are very related to the way human eye perceives colors. The representation of the HSV color space is a hexacone where hue is defined as an angle in the range $[0, 2\Pi]$ relative to the red axis. The saturation is the depth

or purity of the color and is measured as the radical distance from the central axis with values between 0 and 1. The intensity value $[0, 1]$ determines the particular gray shade to which this transformation converges.

❖L* A* B* color space

The LAB color space with coordinates l*, a*, and b* is based on nonlinearly-compressed CIE XYZ color space coordinates but perceptually more linear. It is designed to approximate the human vision as its L component closely matches the human perception of lightness, the A component's position lies between the red and green, while the B component lies between the yellow and blue. Many of the colors within the LAB color space fall outside the gamut of human vision, and are therefore purely imaginary.

❖XYZ color space

The CIE XYZ system is at the root of all colorimetry. It is defined such that all visible colors can be defined using only positive values and the Y value is luminance. Consequently, the colors of the XYZ primaries themselves are not visible. The chromaticity diagram is highly non-linear, in that a vector of unit magnitude representing the difference between two chromaticities is not uniformly visible. A color defined in this system is referred to as Yxy. A third coordinate, z, can also be defined but is redundant since $x + y + z = 1$ for all colors.

1.1.2 Color Histograms

For a pixel p in an image I, let $I(p)$ corresponds to a pixel p, and let I_g denotes its color g, for which $I(p) = g$. Then the histogram for the color g_i is defined as:

$$h_{g_i}(I) \equiv P_{p \in \mathcal{I}}(p \in \mathcal{I}_{g_i}) \tag{1}$$

When dealing with large databases, the histograms of two different images can be the same, this is due to the fact that they do not include any spatial information. The color histograms are susceptible to false positives and are not robust to large appearance changes [13]. To decrease the computation time and memory storage, the sum and difference histograms are computed [20]. The sum and difference histograms with parameters (d_1, d_2) over the domain D (where D is a subset of indexes specifying the texture region) are

$$h_s(i; d_1, d_2) = Card\{(k, l) \in D, S_{k,l} = i\} \tag{2}$$

$$h_d(j; d_1, d_2) = Card\{(k, l) \in D, D_{k,l} = j\} \tag{3}$$

where d_1 and d_2 are the relative positions of the two pictures.

To overcome several problems including the distractions in the background, the viewing from different viewpoints, the occlusion, and the different images resolution; Swain et al. introduced the histogram intersection [8]. It is defined as the intersection between a model histogram and an image histogram. In other words, it is the number of pixels from the model that have corresponding pixels with similar color in the image. Given two histograms, Q and I, each containing m colors, the intersection is defined as:

$$\sum_{i=1}^{m} \min(Q_i, I_i) \tag{4}$$

They also proposed the incremental intersection used for indexing into a large database efficiently. The idea behind it is to compare only the largest bins of the image and construct a partial histogram intersection.

The main disadvantage of the histogram is its lack of spatial information. This missing information may result that two images with difference appearances (See Fig.1.1) can have the same histogram (See Fig.1.2). Researches have come up with other image feature vectors where the spatial information is

Figure 1.1: Two different images having same histogram

included which improves the images indexing and retrieval tasks.

1.2 Approaches to Introduce the Spatial Information

1.2.1 Histogram Refinements

Pass et al. introduced the histogram refinement [9, 10] by dividing the pixels in the image into classes based upon some local property (texture, orientation, relative brightness,...etc.), and then comparing the pixels of the classes using any standard method. The color coherence vector is an example of histogram refinement in which the buckets are divided based on whether they are part of a large group of pixels

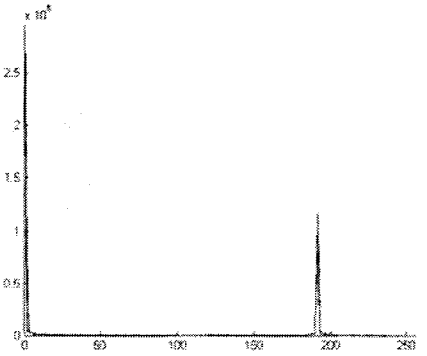

Figure 1.2: Color histogram of both pictures

with the same color (coherent) or not. τ is the value which if the size of the connected components exceeds, the pixels are defined as coherent. Two images I and I' can be compared using their CCV's by using the $L1$ distance the following equation:

$$\Delta ccv = |(\alpha_j - \alpha'_j)| + |(\beta_j - \beta'_j)| \tag{5}$$

where α_j and β_j are the number of coherent and incoherent pixels of the color j respectively. The histogram refinement incorporates spatial information to overcome the problem of the color histogram, it is efficient and gives better results than the histogram.

Another approach introduced by Pass et al. is the joint histogram [11] . The idea behind joint histograms is to add extra information to the image summary to overcome the histogram problem. One or more local features (edge density, texture, gradient magnitude, rank) are selected to be used in parallel with the color histogram. Some of the advantages of the joint histogram is the fact that any improvement in the histogram can be applied directly, as well as the fact that it preserves the robustness of the color histogram.

Joint histograms support parallel computation which can decrease the computation time if multiple processors were used simultaneously. On the other hand, the joint histogram takes a significantly larger space and longer computational time than the color histogram. To overcome the limitations of the joint histogram, Zabih and Pass proposed the reduced intersection [11] as an extension to their work. In the reduced intersection only the largest entries in the color histogram are computed, leading to a decrease in the indexing and retrieval times for large databases.

1.2.2 Color Correlogram

The color correlograms express the spatial correlation of color changes with distance in the images. Correlograms have been used to replace color histograms in different applications such as content-based image retrieval, image subregion querying and localization, video cut detection,and image indexing and classification [13] [14].

For a distance $d \in [n]$, the correlogram of the image I is defined for (g_i, g_j) at a distance d.

$$\gamma^{(d)}_{g_i,g_j}(I) \equiv P_{p_1 \in \mathcal{I}_{g_i}, p_2 \in \mathcal{I}}(p_2 \in \mathcal{I}_{g_j} || p_1 - p_2 = d|) \tag{6}$$

which gives the probability that given any pixel p_1 of level g_i, a pixel p_2 at a distance d in certain direction from the given pixel p_1 is of level g_i.

Color correlograms are stable and scalable in contrary to the histograms. The correlogram is easy to compute with a fairly small feature size. When compared to the histogram and its refinements from the performance point of view, the correlogram showed better results in images indexing and retrieval, and video cut detection [14]. The correlogram is robust in tolerating large changes in appearance of the same scene as well as partial occlusions. The Autocorrelogram only considers the correlation between the identical colors and takes only $O(md)$ space.

$$\alpha^{(d)}_{g}(I) \triangleq \gamma^{(d)}_{g,g}(I) \tag{7}$$

It gives the probability that pixels p_1 and p_2, d away from each other, are of the same level g_i.

Because the correlogram takes $O(m^2 d)$, and to overcome its storage efficiency and computation time problem, the banded correlogram [14] is in general used.

$$\gamma^{(d)}_{g_i,g_j}(I) \triangleq \sum_{d'=db}^{(d+1)b-1} \gamma^{(d')}_{g_i,g_j}(I) \qquad 1 \leq d \leq b \tag{8}$$

If only local information is needed, a small set of distances is enough to capture the spatial correlation of pixels, and reducing the dimensions of the feature. The banded correlogram takes only $m^2 d/b$ space, but it is more susceptible to false matches than the correlogram since

$$|I - I'|_{\gamma', L_1} \leq |I - I'|_{\gamma, L_1} \tag{9}$$

which follows the triangle inequality.

Another correlogram related method to facilitate image subregion querying by avoiding exhaustive searching in an image, is to define the correlogram intersection [14].

$$\gamma^{(d)}_{g_i,g_j}(Q \cap I) = \frac{\Gamma^{(d)}_{g_i,g_j}(Q \cap I)}{H_{g_i}(Q \cap I) \cdot 8k} \tag{10}$$

We measure the presence of Q in I by the distance $|Q - Q \cap I|_{\gamma,L_1}$

The Edge correlogram [13] [14] is obtained by adding the edge information to the color correlogram, it augments the discriminative power of the correlogram. Suppose $\xi : I \rightarrow \{0,1\}$; 1 for an edge and 0 otherwise. Then the edge correlogram can be obtained by:

$$I'(P) = \begin{cases} C+ & \text{if } I(P) = C \text{ and } \xi(P) = 1, \\ C- & \text{if } I(P) = C \text{ and } \xi(P) = 0 \end{cases} \tag{11}$$

The main disadvantage of the edge correlogram is that it takes double the space required by the correlogram which is $(4m^2d)$ space.

1.2.3 Spatial Color Descriptor

The spatial color descriptor [15], proposed by Lee et al. to enhance the performance of image and video analysis, overcomes the two main problems faced by the conventional color descriptors. The algorithm uses the color edge information as well as the augmented histogram based on colors quantized in the HLS color space. The augmented histogram consists of the color adjacency histogram and the color vector angle histogram. The color adjacency histogram includes the spatial information of edge color pairs, while the color vector angle histogram shows the distribution of the smooth image pixels. The proposed algorithm is divided into four parts:

→Color edge detection

Color vector angles are used to identify color edges because they lessen the effect of illumination, and they are sensitive to differences in hue and saturation. The angle separating the color pair (P1) and (P2) represents the perceptual color difference between them (See Fig.1.3). In a 3 × 3 mask, the maximum color vector angle between the center pixel and all eight neighbor pixels is calculated using [21]:

$$\sin(\theta)_{v1,v2} = (1 - \frac{(V_1^T V_2)^2}{V_1^T V_1 V_2^T V_2})^{\frac{1}{2}} \tag{12}$$

and then

$$\sin(\theta)_{max} = \max[\sin(\theta)_{V_c V_1}, \sin(\theta)_{V_c V_2}, ..., \sin(\theta)_{V_c V_8}] \tag{13}$$

If the maximum angle θ is bigger than 0.09 the center pixel is classified as an edge pixel, else it is classified as a smooth pixel.

→Color space conversion and nonuniform quantization

The image is conversed from the RGB to the HLS color space to facilitate the computation and automated image segmentation. Very good results can be achieved when using the HLS color space. After

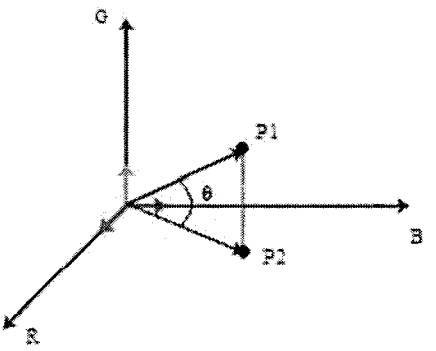

Figure 1.3: Color vector angle

Table 1.1: Quantization levels for H, L, and S.

Number of colors	Achromatic			Low chromatic			High chromatic		
	H	L	S	H	L	S	H	L	S
32	1	4	1	7	3	1	7	1	1

the conversion of the image to the new color space, the color space is divided into three subregions: achromatic, low chromatic, and high chromatic, depending on saturation related values. Because some components are more important that others for a specific subregion, the image is non-uniformly quantized into 32 colors (See Table: 1.1).

→Color adjacency histogram

After the detection of the color edges, a color adjacency histogram of the each edge pixel and the neighboring pixel with the maximum angle is constructed. It is a 32 × 32 matrix representing the count of each color pair. The histogram will normally include many empty bins and only a few peaks. In order to reduce the computational cost and avoid fine comparisons, when the pixel count of a certain bin in the histogram exceeds a certain threshold, it will be classified as effective with a "1", else a noneffective "0". To further refine the histogram, the storage cost is reduced by performing a decimal conversion to the binary stream of each row in the binary matrix.

→Color vector angle histogram

For all smooth pixels, a color vector angle histogram is constructed. The number of frequencies of each color over the entire image is inserted in the corresponding bin of the histogram. For better color classification and because of the large number of smooth pixels, a classification of the smooth pixels is

performed. Depending on the maximum color vector angle between the pixel and the eight-connectivity neighbors, the pixel is classified either into two bins A or Bin B. If the angle is bigger than 0.045 the pixel is counted in Bin A, else it is counted in Bin B.

The augmented histogram, which consists of the color adjacency histogram and the color vector angle histogram, is then used as the color descriptor for image retrieval and video segmentation. Experimental results show that the spatial color descriptor achieve better results than the color histogram and the autocorrelogram in terms of recall and precision.

1.3 Contributions

The contributions of this thesis are as follows:

☞ **Discrete Data Clustering Using Finite Mixture Models:** We investigate the problem of discrete data modeling using finite mixture models. We propose a novel, well motivated mixture that we call the multinomial generalized Dirichlet mixture. The novel model is compared with other discrete mixtures. We designed experiments involving spatial color image databases modeling and summarization to show the robustness, flexibility and merits of our approach.

☞ **Integrating Spatial and Color Information in Images Using A Generative Statistical Framework:** We propose a statistical model to integrate both color and spatial information. Our model is based on finite multiple-Bernoulli mixtures. For the estimation of the model's parameters, we use a maximum a posteriori (MAP) approach through deterministic annealing expectation maximization (DAEM). Smoothing priors on the components parameters are introduced to stabilize the estimation. The selection of the number of clusters is based on stochastic complexity. The results show that our model achieves good performance in a specific image classification problem (city vs. landscape).

1.4 Thesis Overview

The organization of this thesis is as follows:

❑ The first Chapter contains an introduction to image histogram and spatial information, a brief review of some well known approaches found in the literature.

- In Chapter 2, we introduce a finite mixture model as an effective method for different computer vision, image processing and pattern recognition tasks. We propose a multinomial generalized Dirichlet mixture to model discrete data. The novel model is compared with other discrete mixtures and a conclusion is drawn. Two shorter versions of this work have been published in IEEE ICASSP [22] and PAKDD [23]. A long version has been submitted to pattern recognition and revised [24].

- In Chapter 3, we propose a statistical model to integrate both color and spatial information. Our model is based on finite multiple-Bernoulli mixtures. A detailed description of the algorithm and its applications will be presented. A shorter version of this work has been accepted at ICANN [25]. A long version has been submitted to pattern recognition letters [26].

- In the **Conclusions** Chapter, we summarize the various methodologies and contributions that were presented , and we propose several future research directions that are directly or indirectly related to the work performed in this thesis.

CHAPTER 2

Discrete Data Clustering Using Finite Mixture Models

Finite mixture models have been applied in different computer vision, image processing and pattern recognition tasks. The majority of the work done concerning finite mixture models has focused on mixtures for continuous data. However, many applications involve and generate discrete data for which discrete mixtures are better suited.

In this chapter, we investigate the problem of discrete data modeling using finite mixture models. We propose a novel, well motivated mixture which we call a multinomial generalized Dirichlet mixture. The novel model is compared with other discrete mixtures. We designed experiments involving spatial color image databases modeling and summarization to show the robustness, flexibility and merits of our approach.

2.1 Introduction

Discrete data appear in many pattern recognition, machine learning, computer vision and image processing applications. In computer vision, for example, discrete data are present in several applications such as texture modeling, narrowing the semantic gap for content-based image summarization and retrieval [27], histogram clustering and image segmentation [28]. In this work, we are motivated by the need to construct powerful statistical approaches to model, analyze and cluster this type of data. Different statistical models have been proposed and were generally dedicated to text classification and language processing. The majority of these models make the naive Bayes assumption and are

based on multi-variate Bernoulli distributions [29], Poissons mixtures [30, 31] or multinomial distributions [32]. It is well-known that the multinomial distribution performs well in the case of discrete data modeling [33]. However, recent works have shown that even this distribution has some drawbacks such as considering that the events to model are independent [27, 34–36]. Different smoothing techniques have been proposed to overcome these problems. The most successful approach is the use of the Dirichlet distribution as a prior to the multinomial which results in a completely formal statistical model [27, 35, 37, 38]. This is due to the fact that the Dirichlet is a conjugate prior to the multinomial distribution [39]. Despite this conjugacy property, the consistency of its estimates as a prior, its flexibility and its ease of use (See [40, 41] for many other interesting properties of the Dirichlet), the Dirichlet distribution has a very restrictive negative covariance structure which makes its use as a prior in the case of positively correlated data inappropriate [42, 43] as we will show in the next section.

In this chapter, we present a discrete finite mixture model [44] based on both a generalization of the Dirichlet distribution and the multinomial. We stress the fact that the parent distribution in the proposed model is the multinomial and that the generalized Dirichlet is the prior distribution. The key contribution in this chapter lies on the introduction of the generalized Dirichlet mixture as a smoothing technique to deal with the modeling problems that arise when using the multinomial. The choice of the generalized Dirichlet is motivated by the excellent results obtained when we have used it as a parent distribution in different pattern recognition and computer vision tasks [43, 45, 46]. The estimation of the parameters of our mixture model is based on the maximum likelihood estimation by invoking the expectation maximization (EM) approach. In order to accelerate the EM convergence, we have also introduced a Newton-Raphson step. The proposed mixture model is applied to an important problem in computer vision which is the introduction of spatial constraints in color histograms. Indeed, we propose a generative model for this task. Our generative model is used for image databases categorization using both labeled and unlabeled images.

2.2 The Multinomial Generalized Dirichlet Mixture

2.2.1 The Dirichlet Assumption

Let $\vec{X} = (X_1, \ldots, X_D)$ be a discrete vector which means that each element X_d, $d = 1, \ldots, D$ in $\vec{X}$ is discrete and takes on values $1, 2, \ldots, V$. Then, the joint probability of $\vec{X}$ is given by

$$p(\vec{X}|\vec{\pi}) = \prod_{d=1}^{D} \prod_{v=1}^{V} \pi_v^{\delta(X_d=v)} = \prod_{v=1}^{V} \pi_v^{f_v} \quad (1)$$

where $\delta(X_d = v)$ is an indicator function, $\vec{\pi} = (\pi_1, \ldots, \pi_V)$ is the parameter vector, $\pi_v = p(X_d = v)$, $\sum_{v=1}^{V} \pi_v = 1$, and $f_v = \sum_{d=1}^{D} \delta(X_d = v)$. The distribution given by Eq.(1) is a multinomial distribution based on the samples and is different from the well-known following multinomial distribution which model the counts of the samples [1]

$$p(\vec{f}|\vec{\pi}) = \frac{(\sum_{v=1}^{V} f_v)!}{f_1! f_2! \ldots f_V!} \prod_{v=1}^{V} \pi_v^{f_v} \tag{2}$$

where $\vec{f} = (f_1, \ldots, f_V)$. In this chapter, we will use the distribution given by Eq.(1) as we would like to model the samples and not their counts. Using Eq.(1), the samples will be used to set the probabilities, obtaining

$$\hat{\pi}_w = \frac{f_w}{\sum_{v=1}^{V} f_v} \tag{3}$$

which gives poor estimate [27, 48]. Indeed, by using the multinomial, we suppose that the occurrence of a given event is independent of other events which is generally incorrect. Then, the majority of the researchers assign a single Dirichlet or a Dirichlet mixture prior to the parameter vector of multinomial distribution to moderate the extreme estimates given by Eq.(3) [27]. The Dirichlet distribution with V parameters $\vec{\alpha} = (\alpha_1, \ldots, \alpha_V)$ is defined by

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

The Dirichlet distribution depends on V parameters $\alpha_1, \ldots, \alpha_V$, which are all real and positive. In spite of its flexibility and the fact that it is conjugate to the multinomial, the Dirichlet has a very restrictive covariance matrix. Indeed, the covariance between π_v and π_w is [41, 42]

$$Cov(\pi_v, \pi_w) = -\frac{\alpha_v \alpha_w}{(\sum_{l=1}^{V} \alpha_l)^2 (\sum_{l=1}^{V} \alpha_l + 1)}$$

Thus, any two random variables in $\vec{\pi} = (\pi_1, \ldots, \pi_V)$ have to be negatively correlated which is not always the case. In general, this negative correlation violates experimental observations [49] as shown in [50] and [51], in the case of biological data and data describing microelectronic chips, respectively. For instance, if $\vec{\pi}$ describes the normalized histogram of an image or the normalized frequencies of some words in a document, two entries π_v and π_w may be positively correlated. Another restriction of the Dirichlet distribution is that the variables with the same mean must have the same variance as shown in [52]. Then, if we wish, for instance, that different π_v have the same expectation *a priori*, then they must also have the same variance which can make our model unrealistic. All these disadvantages can be handled by using the generalized Dirichlet distribution.

[1] See [47] for a discussion about the difference between samples and counts modeling

2.2.2 The Multinomial Generalized Dirichlet Distribution

The generalized Dirichlet pdf is defined by [50]

$$p(\vec{\pi}|\vec{\xi}) = \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v, \beta_v)} \pi_v^{\alpha_v - 1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_v}$$

where $B(\alpha_v, \beta_v) = \frac{\Gamma(\alpha_v)\Gamma(\beta_v)}{\Gamma(\alpha_v+\beta_v)}$, $\vec{\xi} = (\alpha_1, \beta_1, \ldots, \alpha_{V-1}, \beta_{V-1})$, $\alpha_v > 0, \beta_v > 0, \gamma_v = \beta_v - \alpha_{v+1} - \beta_{v+1}$ for $v = 1 \ldots V-2$ and $\gamma_{V-1} = \beta_{V-1} - 1$. Note that the generalized Dirichlet distribution is reduced to a Dirichlet distribution with parameters $(\alpha_1, \ldots, \alpha_{V-1}, \alpha_V = \beta_{V-1})$ when $\beta_v = \alpha_{v+1} + \beta_{v+1}$. Thus, the generalized Dirichlet includes the Dirichlet as a special case. Compared to the Dirichlet, the generalized Dirichlet has $V-2$ extra parameters which is a very important advantage. Indeed, as the Dirichlet has V parameters, when constructing a Dirichlet prior and if the mean probabilities of the variables have been fixed, it remains only one degree of freedom (by fixing the value of $\sum_{v=1}^{V} \alpha_v$) to adjust the distribution [53]. For the generalized Dirichlet, however, it remains $V-1$ degrees of freedom which makes it more flexible for several applications [43]. The general moment function of the generalized Dirichlet distribution is [54]

$$E(\pi_1^{r_1}, \pi_2^{r_2}, \ldots, \pi_{V-1}^{r_{V-1}}) = \prod_{v=1}^{V-1} \frac{B(\alpha_v, \beta_v)}{B(\alpha_v + r_v, \beta_v + \delta_v)}$$

where $\delta_v = r_{v+1} + r_{v+2} + \ldots + r_{V-1}$ for $v = 1, 2, \ldots, V-2$ and $\delta_{V-1} = 0$. Then, we can show that the mean and the variance of the generalized Dirichlet distribution satisfy the following conditions [50, 54]

$$E(\pi_v) = \frac{\alpha_v}{\alpha_v + \beta_v} \prod_{k=1}^{v-1} \frac{\beta_k}{\alpha_k + \beta_k} \tag{4}$$

$$Var(\pi_v) = E(\pi_v)\left(\frac{\alpha_v + 1}{\alpha_v + \beta_v + 1} \prod_{k=1}^{v-1} \frac{\beta_k + 1}{\alpha_k + \beta_k + 1} - E(\pi_v) \right) \tag{5}$$

and the covariance between π_v and π_w is

$$Cov(\pi_v, \pi_w) = E(\pi_w)\left(\frac{\alpha_v}{\alpha_v + \beta_v + 1} \prod_{k=1}^{v-1} \frac{\beta_k + 1}{\alpha_k + \beta_k + 1} - E(\pi_v) \right)$$

Note that the generalized Dirichlet distribution has a more general covariance structure than the Dirichlet distribution and that variables with the same mean do not need to have the same variance. In addition to these properties, the generalized Dirichlet is conjugate to the multinomial distribution and the joint distribution of $\vec{X}$ and $\vec{\pi}$ is (See appendix A)

$$p(\vec{X}, \vec{\pi}|\vec{\xi}) = \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v, \beta_v)} \pi_v^{\alpha_v' - 1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_v'} \tag{6}$$

where $\alpha_v^{'} = \alpha_v + f_v$ and $\beta_v^{'} = \beta_v + f_{v+1} + \ldots + f_V$ for $v = 1, \ldots, V-1$, $\gamma_v^{'} = \beta_v^{'} - \alpha_{v+1}^{'} - \beta_{v+1}^{'}$ for $v = 1, \ldots, V-2$ and $\gamma_{V-1}^{'} = \beta_{V-1}^{'} - 1$. Integrating over $\vec{\pi}$, we obtain the marginal distribution of $\vec{X}$

$$\begin{aligned} p(\vec{X}|\vec{\xi}) &= \int_{\vec{\pi}} p(\vec{X}, \vec{\pi}|\vec{\xi}) d\vec{\pi} = \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v, \beta_v)} \int_{\vec{\pi}} \pi_v^{\alpha_v^{'}-1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_v^{'}} d\vec{\pi} \\ &= \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v, \beta_v)} \prod_{v=1}^{V-1} B(\alpha_v^{'}, \beta_v^{'}) \end{aligned}$$

We call this density the multinomial generalized Dirichlet distribution (MGD). Then, the posterior is given by

$$p(\vec{\pi}|\vec{X}, \vec{\xi}) = \frac{p(\vec{X}, \vec{\pi}|\vec{\xi})}{p(\vec{X}|\vec{\xi})} = \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v^{'}, \beta_v^{'})} \pi_v^{\alpha_v^{'}-1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_v^{'}} \tag{7}$$

which is a generalized Dirichlet with parameters $(\alpha_1^{'}, \beta_1^{'}, \ldots, \alpha_{V-1}^{'}, \beta_{V-1}^{'})$. Then, by taking the generalized Dirichlet as a prior to the multinomial and according to Eq.(4) and Eq.(7), we obtain

$$\hat{\pi}_w = E[\pi_w|\vec{X}; \vec{\xi}] = \frac{\alpha_w + f_w}{\alpha_w + \beta_w + n_w} \prod_{l=1}^{w-1} \frac{\beta_l + n_{l+1}}{\alpha_l + \beta_l + n_l} \tag{8}$$

where $n_l = f_l + f_{l+1} + \ldots + f_V$. When $\beta_v = \alpha_{v+1} + \beta_{v+1}$, it is straightforward to verify that this equation is reduced to

$$\hat{\pi}_w = E[\pi_w|\vec{X}; \vec{\xi}] = \frac{\alpha_w + f_w}{\sum_{v=1}^{V} (\alpha_v + f_v)} \tag{9}$$

where $\alpha_V = \beta_{V-1}$, which represents the expectation when we consider a Dirichlet distribution, with parameters $(\alpha_1, \ldots, \alpha_V)$, as a prior. Note that Eq.(9) is reduced to Eq.(3) when $\alpha_v = 0, v = 1, \ldots, V$, and to the well-known Laplace smoothing equation

$$\hat{\pi}_w = \frac{1 + f_w}{V + \sum_{v=1}^{V} f_v}$$

when $\alpha_v = 1, v = 1, \ldots, V$.

2.2.3 The Multinomial Generalized Dirichlet Mixture

Suppose now that we select a generalized Dirichlet mixture as a prior to the multinomial. A generalized Dirichlet mixture with M components is defined as

$$p(\vec{\pi}|\Theta) = \sum_{j=1}^{M} p(\vec{\pi}|\vec{\xi_j}) p_j$$

where p_j ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$) are the mixing proportions and $p(\vec{\pi}|\vec{\xi_j})$ is the generalized Dirichlet. The symbol $\Theta = (\vec{\xi}_1, \ldots, \vec{\xi}_M, p_1, \ldots, p_M)$ refers to the entire set of parameters to be estimated. With a mixture prior, the joint distribution of $\vec{X}$ and $\vec{\pi}$ is

$$p(\vec{X}, \vec{\pi}|\Theta) = \sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{jv}, \beta_{jv})} \pi_v^{\alpha'_{jv}-1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma'_{jv}}$$

and the marginal distribution of $\vec{X}$ is

$$p(\vec{X}|\Theta) = \int_{\vec{\pi}} p(\vec{X}, \vec{\pi}|\Theta) d\vec{\pi} = \sum_{j=1}^{M} p_j p(\vec{X}|\vec{\xi_j}) = \sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{B(\alpha'_{jv}, \beta'_{jv})}{B(\alpha_{jv}, \beta_{jv})} \tag{10}$$

We call this density the multinomial generalized Dirichlet mixture (MGDM). Then, the posterior is given by

$$p(\vec{\pi}|\vec{X}, \Theta) = \frac{p(\vec{X}, \vec{\pi}|\Theta)}{p(\vec{X}|\Theta)} = \frac{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{jv}, \beta_{jv})} \pi_v^{\alpha'_{jv}-1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma'_{jv}}}{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{B(\alpha'_{jv}, \beta'_{jv})}{B(\alpha_{jv}, \beta_{jv})}}$$

And (See Appendix B)

$$\hat{\pi}_w = E[\pi_w|\vec{X}; \Theta] = \sum_{j=1}^{M} p(j|\vec{X}; \xi_j) \frac{\alpha'_{jw}}{\alpha'_{jw} + \beta'_{jw}} \prod_{k=1}^{w-1} \frac{\beta'_{jk}}{\alpha'_{jk} + \beta'_{jk}} \tag{11}$$

where

$$p(j|\vec{X}; \xi_j) = \frac{p_j p(\vec{X}|\xi_j)}{\sum_{j=1}^{M} p_j p(\vec{X}|\xi_j)}$$

and represents the posterior probability. Note that Eq.(11) is reduced to Eq.(8) when $j = 1$.

2.3 Maximum Likelihood Estimation

Given a set of independent vectors $\mathcal{X} = \{\vec{X}_1, \ldots, \vec{X}_N\}$, the log-likelihood corresponding to an M-component MGDM is given by

$$L(\mathcal{X}, \Theta) = \sum_{i=1}^{N} \log \left(\sum_{j=1}^{M} p_j p(\vec{X}_i|\xi_j) \right) \tag{12}$$

It's well-known that the maximum likelihood (ML) estimate [2]

$$\hat{\Theta}_{ML} = \arg\max_{\Theta} \{L(\Theta, \mathcal{X})\} \tag{13}$$

[2] We are actually performing an empirical Bayes estimation, by maximizing over the generalized Dirichlet prior distribution parameters as opposed to the parameters of the multinomials. See [55] for more details about empirical Bayes approaches.

cannot be found analytically. The maximization defining the ML estimates is subject to the constraints $0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$. Obtaining ML estimates of the mixture parameters is possible through EM and related techniques [56]. The EM algorithm is a general approach to maximum likelihood in the presence of incomplete data. In EM, the "complete" data are considered to be $Y_i = \{\vec{X}_i, \vec{Z}_i\}$, where $\vec{Z}_i = (Z_{i1}, \ldots, Z_{iM})$ with

$$Z_{ij} = \begin{cases} 1 & \text{if } \vec{X}_i \text{ belongs to class } j \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

constituting the "missing" data. The relevant assumption is that the density of an observation $\vec{X}_i$ given $\vec{Z}_i$ is given by $\prod_{j=1}^{M} p(\vec{X}_i|\xi_j)^{Z_{ij}}$. The resulting *complete-data log-likelihood* is

$$L(\mathcal{X}, \Theta, \mathcal{Z}) = \sum_{i=1}^{N}\sum_{j=1}^{M} Z_{ij} \log\Big(p_j p(\vec{X}_i|\xi_j)\Big) \tag{15}$$

The Q-function (the conditional expectation) of the complete-data log-likelihood in the above equation is

$$Q(\Theta; \Theta^{(t)}) = \sum_{i=1}^{N}\sum_{j=1}^{M} \hat{Z}_{ij} \log(p_j) + \sum_{i=1}^{N}\sum_{j=1}^{M} \hat{Z}_{ij} \log\Bigg(\prod_{v=1}^{V-1} \frac{B(\alpha_{jv} + f_{iv}, \beta_{jv} + n_{iv+1})}{B(\alpha_{jv}, \beta_{jv})}\Bigg) \tag{16}$$

where $\Theta^{(t)}$ is the value of Θ at iteration t and

$$\hat{Z}_{ij} = p(Z_{ij} = 1|\vec{X}_i; \Theta^{(t)}) = \frac{p_j^{(t)} p(\vec{X}_i|\xi_j^{(t)})}{\sum_{j=1}^{M} p_j^{(t)} p(\vec{X}_i|\xi_j^{(t)})} \tag{17}$$

The first term in Eq.(16) can be maximized by updating p_j as following

$$p_j^{(t+1)} = \frac{\sum_{i=1}^{N} \hat{Z}_{ij}^{(t)}}{N} \tag{18}$$

The maximization of the second term, however, does not yield to a closed form solution. Thus, we will use Newton-Raphson method which is based on the computation of the first and second derivatives.

$$\begin{aligned} \frac{\partial Q(\Theta; \Theta^{(t)})}{\partial \alpha_{jv}} &= \sum_{i=1}^{N}\sum_{j=1}^{M} \hat{Z}_{ij}\Big(\Psi(\alpha_{jv} + \beta_{jv}) - \Psi(\alpha_{jv}) \\ &+ \Psi(\alpha_{jv} + f_{iv}) - \Psi(\alpha_{jv} + \beta_{jv} + n_{iv})\Big) \end{aligned} \tag{19}$$

$$\begin{aligned} \frac{\partial Q(\Theta; \Theta^{(t)})}{\partial \beta_{jv}} &= \sum_{i=1}^{N}\sum_{j=1}^{M} \hat{Z}_{ij}\Big(\Psi(\alpha_{jv} + \beta_{jv}) - \Psi(\beta_{jv}) \\ &+ \Psi(\beta_{jv} + n_{iv+1}) - \Psi(\alpha_{jv} + \beta_{jv} + n_{iv})\Big) \end{aligned} \tag{20}$$

By computing the second and mixed derivatives of $Q(\Theta;\Theta^{(t)})$ we obtain

$$\frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\alpha_{jv_1}\partial\alpha_{jv_2}} = \begin{cases} \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi^{'}(\alpha_{jv}+\beta_{jv})-\Psi^{'}(\alpha_{jv}) \\ +\Psi^{'}(\alpha_{jv}+f_{iv})-\Psi^{'}(\alpha_{jv}+\beta_{jv}+n_{iv})\Big) & \text{if } v_1=v_2=v \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

$$\frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\beta_{jv_1}\partial\beta_{jv_2}} = \begin{cases} \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi^{'}(\alpha_{jv}+\beta_{jv})-\Psi^{'}(\beta_{jv}) \\ +\Psi^{'}(\beta_{jv}+n_{iv+1})-\Psi^{'}(\alpha_{jv}+\beta_{jv}+n_{iv})\Big) & \text{if } v_1=v_2=v \\ 0 & \text{otherwise} \end{cases} \tag{22}$$

$$\frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\beta_{jv_1}\partial\alpha_{jv_2}} = \frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\alpha_{jv_1}\partial\beta_{jv_2}} = \begin{cases} \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi^{'}(\alpha_{jv}+\beta_{jv}) \\ -\Psi^{'}(\alpha_{jv}+\beta_{jv}+n_{iv})\Big) & \text{if } v_1=v_2=v \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

where Ψ and $\Psi^{'}$ are the digamma and trigamma functions. Then, the Hessian matrix has a block-diagonal structure

$$H(\xi_j) = \text{block-diag}\big\{H_1(\alpha_{j1},\beta_{j1}),\ldots,H_{V-1}(\alpha_{jV-1},\beta_{jV-1})\big\} \tag{24}$$

where

$$H_v(\alpha_{jv},\beta_{jv}) = \begin{pmatrix} \frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial^2\alpha_{jv}} & \frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\alpha_{jv}\partial\beta_{jv}} \\ \frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial\beta_{jv}\partial\alpha_{jv}} & \frac{\partial^2 Q(\Theta;\Theta^{(t)})}{\partial^2\beta_{jv}} \end{pmatrix} \tag{25}$$

and we have [57, Theorem 8.8.16]

$$H(\xi_j)^{-1} = \text{block-diag}\big\{H_1(\alpha_{j1},\beta_{j1})^{-1},\ldots,H_v(\alpha_{jv},\beta_{jv})^{-1}\big\} \tag{26}$$

We remark that $H_v(\alpha_{jv},\beta_{jv})$ can be written as:

$$H_v(\alpha_{jv},\beta_{jv}) = S+\gamma\vec{a}\vec{a}^T \tag{27}$$

where
$S = \text{diag}\Big[\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi^{'}(\alpha_{jv})+\Psi^{'}(\alpha_{jv}+f_{iv})\Big),\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi^{'}(\beta_{jv})+\Psi^{'}(\beta_{jv}+n_{iv+1})\Big)\Big], \gamma = \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi^{'}(\alpha_{jv}+\beta_{jv})-\Psi^{'}(\alpha_{jv}+\beta_{jv}+n_{iv})\Big), \vec{a}^T = \mathbf{1}$ and $\gamma \neq (\sum_{k=1}^{2}\frac{a_k^2}{S_{kk}})^{-1}$.
Then, the inverse of the matrix $H_l(\alpha_{jl},\beta_{jl})$ is given by [57, Theorem 8.3.3]:

$$H_v(\alpha_{jv},\beta_{jv})^{-1} = S^{-1}+\delta^* a^* a^{*T} \tag{28}$$

where:

$$S^{-1} = \text{diag}\Bigg[\frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\alpha_{jv})+\Psi'(\alpha_{jv}+f_{iv})\Big)}, \frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\beta_{jv})+\Psi'(\beta_{jv}+n_{iv+1})\Big)}\Bigg] \tag{29}$$

$$a^{*T} = (a_1/S_1, a_2/S_2) \tag{30}$$
$$= \Bigg(\frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\alpha_{jv})+\Psi'(\alpha_{jv}+f_{iv})\Big)}, \frac{1}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\beta_{jv})+\Psi'(\beta_{jv}+n_{iv+1})\Big)}\Bigg)$$

$$\delta^* = -\gamma\Big(1+\gamma(1/S_1+1/S_2)\Big)^{-1} \tag{31}$$
$$= \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi'(\alpha_{jv}+\beta_{jv})-\Psi'(\alpha_{jv}+\beta_{jv}+n_{iv})\Big)$$
$$\times \Bigg(1+\frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi'(\alpha_{jv}+\beta_{jv})-\Psi'(\alpha_{jv}+\beta_{jv}+n_{iv})\Big)}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\alpha_{jv})+\Psi'(\alpha_{jv}+f_{iv})\Big)}$$
$$+ \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(\Psi'(\alpha_{jv}+\beta_{jv})-\Psi'(\alpha_{jv}+\beta_{jv}+n_{iv})\Big)}{\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}\Big(-\Psi'(\beta_{jv})+\Psi'(\beta_{jv}+n_{iv+1})\Big)}\Bigg)^{-1}$$

Given a set of initial estimates, Newton-Raphson method can now be used. The iterative scheme of the Newton-Raphson method is given by the following equation:

$$\vec{\xi}_j^{(t)} = \vec{\xi}_j^{(t-1)} - H(\vec{\xi}_j^{(t-1)})^{-1}\frac{\partial Q(\Theta;\Theta^{(t-1)})}{\partial\vec{\xi}_j^{(t-1)}} \tag{32}$$

where $-H(\vec{\xi}_j^{(t-1)})^{-1}\frac{\partial Q(\Theta;\Theta^{(t-1)})}{\partial\vec{\xi}_j^{(t-1)}}$ is often referred to as the Newton direction. The introduction of the Hessian matrix helps to accelerate the convergence of our algorithm which can be viewed as a hybrid or generalized expectation maximization algorithm [58]. Note, however, that in order to converge to a local maximum, the hessian $H(\vec{\xi}_j^{(t-1)})$ should be negative definite [56]. Consequently, all the submatrices $H_v(\alpha_{jv},\beta_{jv})$ should be negative definite which is not always the case. The problem now is how to

approximate $H_v(\alpha_{jv}, \beta_{jv})$ to be negative definite. Since the inverse of a negative definite matrix is negative definite, it follows that $S^{-1} + \delta^* a^* a^{*T}$ should be negative definite. Note that the two diagonal entries in S^{-1} are negative because the trigamma function is decreasing [59]. Then, since $\gamma > 0$, a sufficient condition for $H_v(\alpha_{jv}, \beta_{jv})$ to be negative definite is: $\Big(1 + \gamma(1/S_1 + 1/S_2)\Big)^{-1} > 0$ which suggests that γ has to be replaced by $\min\{\gamma, \frac{\epsilon - 1}{1/S_1 + 1/S_2}\}$, where ϵ is a small positive real constant, during the iterations [60].

2.4 Complete Algorithm of Estimation and Selection

The initialization step is very important when we deal with mixture models and should take into account the nature of the data; i.e continuous or discrete. The majority of effective initialization algorithms such K-Means and Fuzzy C-Means [61] are dedicated to continuous data and cannot be applied to discrete data [62]. For instance, the classical K-means algorithm uses Euclidean distance which is inappropriate to cluster discrete data [63] and gives poor results [64]. In order to make our estimation algorithm less sensitive to local maxima, we have used an initialization scheme using both the spherical K-Means algorithm [63] and the method of moment (MM). The spherical K-Means algorithm was proposed in [63] as an extension to the classical K-Means algorithm and uses cosine similarity [65], rather than Euclidean distance, and thus is better suited for us. In order to apply this algorithm, we have computed the frequency vectors $\vec{f_i}$ associated with each $\vec{X_i}, i = 1, \ldots, N$. Then, we have applied a preprocessing step, called normalized term frequency scheme, in order to form new vectors $\vec{Y_i} = (Y_{i1}, \ldots, Y_{iV})$ that will be used as input to the spherical K-Means algorithm. The normalized term frequency scheme is based on the following equation [63]

$$Y_{iv} = f_{iv} \Big(\sum_{l=1}^{V} f_{il}^2 \Big)^{-1/2} \tag{33}$$

More details about this normalization scheme and the spherical K-Means algorithm can be found in [63].

INITIALIZATION Algorithm

1. INPUT: Discrete vectors $\vec{X_i}, i = 1, \ldots, N$, V, and number of clusters M.
2. Compute the frequencies vectors $\vec{f_i} = (f_{i1}, \ldots, f_{iV}), i = 1, \ldots, N$.
3. Apply the normalized term frequency scheme using Eq.(33), $v = 1, \ldots, V$, $i = 1, \ldots, N$.
4. Apply the spherical K-Means [63] algorithm to obtain the elements of each component.

5. Apply the MM, based on Eq.(4) and Eq.(5), for each component j.

6. Assign the data to clusters, assuming that the current model is correct.

7. If the current model and the new model are sufficiently close to each other, terminate, else go to step 4.

The MM is applied by using the first moment given by Eq.(4) and the second moment given by Eq.(5) of each $\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}$, $v = 1, \ldots, V$, which form a system of linear equations to solve for α_{jv} and β_{jv}. Straightforward manipulations lead to the following equations for the initialization:

$$\alpha_{jv} = \frac{E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right) C}{\prod_{k=1}^{v-1} \frac{\beta_{jk}}{\alpha_{jk}+\beta_{jk}} - E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right) C - C} \tag{34}$$

$$\beta_{jv} = \frac{C \prod_{k=1}^{v-1} \frac{\beta_{jk}}{\alpha_{jk}+\beta_{jk}} - C^2}{\prod_{k=1}^{v-1} \frac{\beta_{jk}}{\alpha_{jk}+\beta_{jk}} - E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right) C - C} \tag{35}$$

$$C = \frac{E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right) \prod_{k=1}^{v-1} \frac{\beta_{jk}+1}{\alpha_{jk}+\beta_{jk}+1} - E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right)^2 + Var\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right)}{E\left(\frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}}\right)^2} \tag{36}$$

An important part of the modeling problem is about determining the number of consistent clusters which best describe the data. For this purpose, many approaches have been suggested [44, 66]. In our case, we have used the Bayesian information criterion (BIC) proposed by Schwarz [67]

$$BIC(M) = \log(L(\mathcal{X}), \Theta) - \frac{1}{2} N_p \log(N) \tag{37}$$

where N_p is the number of free parameters in the mixture model and is equal to $M(2d+1)$ in our case. Having the BIC criterion and the initialization algorithm in hand, the complete algorithm for estimation and selection is as 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 $\hat{Z}_{ij}^{(t)}$ using Eq.(17).

3. M-Step:

(a) Update the $p_j^{(t)}$ using Eq.(18).

(b) Update the $\xi_j^{(t)}$ using Eq.(32).

4. Calculate the associated criterion BIC(M) using Eq.(37).

5. Select the optimal model M^* such that:

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

2.5 Experimental Results

Our intent, in this section, is to compare our model with previously proposed approaches for discrete data classification and modeling. As we have mentioned in the introduction, discrete data are an important component in a wide range of domains and applications such as pattern recognition, image and text processing. In image processing and computer vision, for instance, features extraction is an important step for several applications such as content-based image database categorization and retrieval. The extracted features may be discrete (ex. color histogram) and then should be represented by a discrete model. We show the merits of our proposed approach by discussing the following two applications. In the first application, our approach is used to develop a generative model for spatial color image databases summarization. The second application concerns text document classification.

2.5.1 A Generative Model for Spatial Color Image Databases Summarization

In recent years, there has been a tremendous increase in the generation of digital images. As this content grows, the need for tools to summarize, filter and retrieve image databases becomes more important. A variety of techniques have been proposed to retrieve this content [68, 69]. Although different, all these techniques agree on the fact that an efficient summarization scheme plays an important role. Summarizing an image database is very important because it simplifies the task of retrieval by restricting the search for similar images to a smaller domain of the database [70]. Summarization is also very efficient for browsing. Color histograms are widely used as features vectors for image summarization and retrieval [7, 8] and are used in different systems [68, 69]. This can be explained by the fact that histograms provide a stable object recognition in the presence of occlusions and over views change [7]. However, histograms do not include any spatial information which is an important issue in the human visual perception. Indeed, images with different appearance may have similar histograms which is a

critical problem in large image databases [11]. Different approaches have been proposed to integrate spatial information with color histograms [13, 71]. In the next subsection, we propose a statistical model based on the multinomial generalized Dirichlet mixture (Eq.(10)) to introduce the spatial information into color histograms. The proposed model is then applied to images databases summarization using both labeled and unlabeled images.

The Generative Model

In this first application, we present a probabilistic framework for images summarization using both color and spatial information. The problem of image summarization is of great importance given the huge number of images generated every day. Our summarization problem is the following: given a single unlabeled (its class is unknown) image, classify it into one of a set of learned classes. The learned models are generated from the training labeled images for each class. Let us introduce our generative model. Suppose that we have N labeled images $\mathcal{I}_i$, $i = 1, \ldots, N$ classified in R classes and that the number of labeled images in each class r is equal to n_r where $\sum_{r=1}^{R} n_r = N$. By associating a distribution and a weight to each class in the training set, we can suppose that each image $\mathcal{I}_i$ is generated by a mixture of R distributions with parameters $\vec{\pi} = (\vec{\pi}_1, \ldots, \vec{\pi}_R)$

$$p(\mathcal{I}_i|\vec{\pi}) = \sum_{r=1}^{R} p_r p(\mathcal{I}_i|\vec{\pi}_r) \tag{38}$$

The problem now is the determination of $p(\mathcal{I}_i|\vec{\pi}_r)$. For this, let us introduce some notations. An $L \times K$ image $\mathcal{I}_i$ is considered to be a set of pixels $\{X_{i_{lk}}, l = 1, \ldots, L; k = 1, \ldots, K\}$, where $X_{i_{lk}}$ is the pixel in position (l, k) of image $\mathcal{I}_i$. The colors in $\mathcal{I}_i$ are quantized into C colors $c_1, \ldots, c_C$. The distribution $p(\mathcal{I}_i|\vec{\pi}_r)$ can be described in terms of the features of the image. In our case, the features are the pixels. In order to introduce the spatial information, the probability of a pixel should be conditioned on its neighborhood. By taking the neighborhood consisting of the pixels at a distance $\Delta \in \{\Delta_1, \ldots, \Delta_{DIS}\}$ measured using the L_∞ norm, $p(\mathcal{I}_i|\vec{\pi}_r)$ will be given by

$$p(\mathcal{I}_i|\vec{\pi}_r) = \prod_{d=1}^{DIS} \prod_{l=1}^{L} \prod_{k=1}^{K} p(X_{i_{lk}}|\vec{\pi}_r; X_{i_{l'k'}}, \Delta_d) \tag{39}$$

where $|(l, k) - (l', k')| = \max\{|l - l'|, |k - k'|\} = \Delta_d$. Note that Eq.(39) will represent the classic image histogram, if we suppose that each pixel $X_{i_{lk}}$ is independent of its neighborhood, which is actually the standard naive Bayes assumption [32]. According to Eq.(39) the parameters of an individual mixture component are a multinomial distribution over the $C \times C$ possible color pairs and

can be written as $\pi_{c_{t_1},c_{t_2},\Delta_d|r}$, where $t_1, t_2 = 1,\ldots,C$ and $\pi_{c_{t_1},c_{t_2},\Delta_d|r} = p\Big(X_{i_{lk}} = c_{t_1}, X_{i_{l'k'}} = c_{t_2} \big| |(l,k)-(l',k')| = \Delta_d\Big)$, $l,l' = 1,\ldots,L$, $k,k' = 1,\ldots,K$, which is the probability that a pixel of color c_{t_1} has at a distance Δ_d a pixel of color c_{t_2}. Then, Eq.(39) could be written as follows

$$p(\mathcal{I}_i|\vec{\pi}_r) = \prod_{d=1}^{DIS}\prod_{c_{t_1}=1}^{C}\prod_{c_{t_2}=1}^{C} \pi_{c_{t_1},c_{t_2},\Delta_d|r}^{f_{c_{t_1},c_{t_2},\Delta_d}} \tag{40}$$

$$f_{c_{t_1},c_{t_2},\Delta_d} \equiv Card\{(X_{i_{lk}}, X_{i_{l'k'}}) = (c_{t_1},c_{t_2}) \big| |(l,k)-(l',k')| = \Delta_d\} \tag{41}$$

where $Card\{\}$ refers to the number of elements of a set. Learning our model consists of estimating the parameters $\pi_{c_{t_1},c_{t_2},\Delta_d|r}$ using the n_r labeled images in class r. By noting that we can associate a C^2-dimensional vector of frequencies $\vec{f}_{i,\Delta_d} = (f_{i,c_1,c_1,\Delta_d},\ldots,f_{i,c_1,c_C,\Delta_d},\ldots,f_{i,c_C,c_1,\Delta_d},\ldots,f_{i,c_C,c_C,\Delta_d})$ to each image $\mathcal{I}_i$ for each distance Δ_d, the parameters are estimated using Eq.(11)

$$\pi_{c_{t_1},c_{t_2},\Delta_d|r} = \sum_{j=1}^{M} p(j|\mathcal{I}_i;\xi_j) \frac{\alpha'_{j,c_{t_1},c_{t_2},\Delta_d}}{\alpha'_{j,c_{t_1},c_{t_2},\Delta_d} + \beta'_{j,c_{t_1},c_{t_2},\Delta_d}} \prod_{c_{t'_1}=1}^{c_{t_1}}\prod_{c_{t'_2}=1}^{c_{t_2}} \frac{\beta'_{j,c_{t'_1},c_{t'_2},\Delta_d}}{\alpha'_{j,c_{t'_1},c_{t'_2},\Delta_d} + \beta'_{j,c_{t'_1},c_{t'_2},\Delta_d}}$$

where $p(j|\mathcal{I}_i;\xi_j)$, $(\alpha'_{j,c_1,c_1,\Delta_d},\ldots,\alpha'_{j,c_1,c_C,\Delta_d},\ldots,\alpha'_{j,c_C,c_1,\Delta_d},\ldots,\alpha'_{j,c_C,c_C,\Delta_d})$ and M are determined for each distance Δ_d using the algorithm in Section 2.4 applied to the data set composed of the images in class r. In our experiments, we take $M_{min} = 1$ and $M_{max} = 10$. Having all the parameters describing Eq.(38) in hand, we can now assign a given test image $\mathcal{I}_t$ to a particular mixture component in Eq.(38) by using the Bayes' rule: $\mathcal{I}_t \longmapsto \arg\max_r p_r p(\mathcal{I}_t|\vec{\pi}_r)$, where $p_r = \frac{n_r}{N}$ [61].

Spatial Color Image Databases Summarization

In our experiments, we used a database containing 45100 images and composed of 10 homogeneous classes (See Figure 2.1). We divided the database into two sets. A data set containing 22550 images used for training. The remaining images were used for testing. The repartition of the different classes in the training and test sets is given in table 2.1. We considered the RGB space with color quantization into 512 colors ($8 \times 8 \times 8$) and the set of distances $\Delta = \{1,3,5,7,9,11\}$ used to introduce the spatial information (See Section 2.5.1). Further, we have considered only probabilities of pixels having same colors in order to reduce zero frequencies, which is a common approach and used, for instance, in the case of the autocorrelogram proposed by Huang et al. [13]. Then, each image was represented by a 512-dimensional vector of frequencies. Table 2.2 shows the number of clusters found to represent each image class in the training set for each distance used.

In order to measure the classification accuracy produced by our classifier, we have counted the number

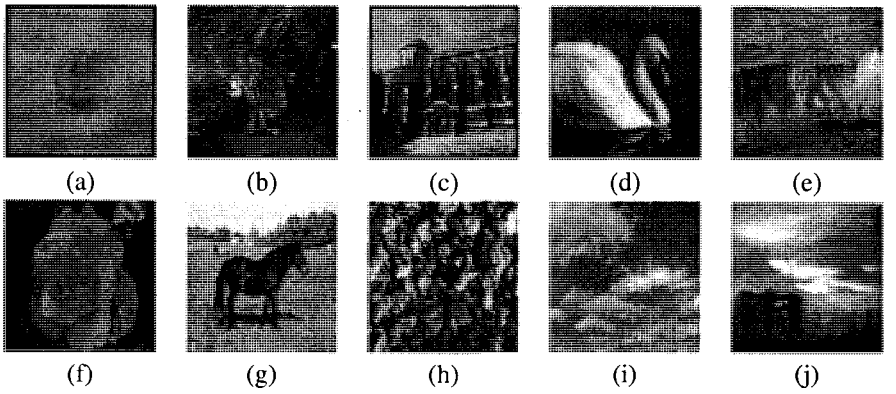

Figure 2.1: Sample images from each group. (a) Class1, (b) Class2, (c) Class3, (d) Class4, (e) Class5, (f) Class6, (g) Class7, (h) Class8, (i) Class9, (j) Class10,

Table 2.1: Repartition of the different classes in the training and test sets.

class	Training set	Testing set
Class1	2250	2250
Class2	2500	2500
Class3	3000	3000
Class4	1900	1900
Class5	2000	2000
Class6	2100	2100
Class7	2250	2250
Class8	2200	2200
Class9	2050	2050
Class10	2300	2300

of misclassified images (a misclassified image is an image that should be in class j_1, but is classified in j where $j \neq j_1$) in each class. The number of images misclassified when we used multinomial generalized Dirichlet mixture (MGDM), was 2189, which represents an accuracy of 90.29 percent where the accuracy is measured by $(\frac{\text{Number of images in the testing set} - \text{number of misclassified images}}{\text{Number of images in the testing set}} \times 100)$. Table 2.3 represents the classification accuracies when using the multinomial generalized Dirichlet mixture, multinomial Dirichlet mixture, single multinomial generalized Dirichlet (MGD), single mulinomial Dirichlet (MD), multinomial mixtures without smoothing (MM), multinomial mixtures with Laplace smoothing (MMLS), INITIALIZATION algorithm and the spherical K-Means. We have also compared our results with the FCA-MPL algorithm [62] which is an extension to the classical fuzzy c-means taking into account the discrete nature of the data. Note that the improvement achieved by the MGDM, comparing to the other approaches, is highly statistically significant. We have also measured the training time of

Table 2.2: Number of clusters found to represent each images class in the training set for each distance used.

	Δ=1	Δ=3	Δ=5	Δ=7	Δ=9	Δ=11
Class1	2	1	3	1	2	2
Class2	3	2	4	2	2	1
Class3	4	3	5	3	2	2
Class4	2	2	2	2	1	2
Class5	1	2	2	1	2	1
Class6	2	2	1	2	2	2
Class7	3	3	2	1	2	2
Class8	2	3	3	3	2	2
Class9	2	1	2	3	2	2
Class10	4	2	2	3	1	2

Table 2.3: Classification accuracies using different methods.

Method	Classification accuracy	Training time (seconds)
MGDM	90.29%	350.48
MDM	83.54%	320.48
MGD	85.10%	290.32
MD	82.15%	270.19
MM	80.35%	210.08
MMLS	81.07%	210.08
INITIALIZATION	82.11%	240.12
Spherical K-Means	78.40%	180.45
FCA-MPL	78.34%	190.13

the different methods. For instance, the training times when using the INITIALIZATION algorithm and MGDM were 280.12 and 350.48 seconds, respectively, which show that the initialization helps to reach convergence rapidly. The training using the MGDM is slower than the MDM (320.54 seconds), which is acceptable since the gain in accuracy is significant.

Figure 2.2 shows the accuracy of the classification, when using the three mixtures, as a function of the number of images in the training set. It is clear that the classification correctness increases as the number of training images increase. We have also tested the representation of the image colors in the HSV space and we did not remark much changes in the results. From the results, we can conclude that the MDM and the MGDM perform better than the multinomial. This can be explained by the sparseness problem; i.e the zero frequency problem [72, 73]. Indeed, many frequencies are actually zero or have very small probabilities. Then, when the multinomial is used for modeling, prediction and classification, a large number of observations will be judged to be impossible based on the training data. The introduction of the Dirichlet and the generalized Dirichlet as priors can be viewed as smoothing technique [74] to deal with this problem.

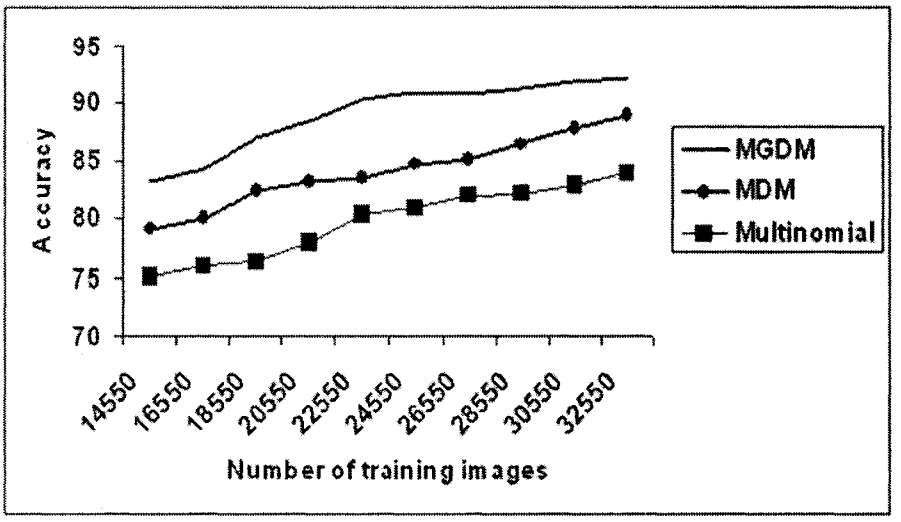

Figure 2.2: Accuracy, using spatial color information, as a function of the number of images in the training set

Spatial Color Image Database Summarization Using Labeled and Unlabeled Images

From the previous application, we can see clearly that the accuracy of classification is improved by increasing the number of images in the training set (labeled images). However, obtaining these labeled images is very expensive in terms of time, since the labeling has to be done by a person. In this second application, we will try to improve the accuracy of our classification by using unlabeled images from the test set. Combining labeled and unlabeled images is an old approach in the statistics community (for instance see [75, 76]). The basic idea is to use the available labeled images to train a classifier to label the unlabeled images, using the Bayes' decision rule, which will be added to the training set to build a new classifier. Roughly speaking the classifier is determined by estimating the vectors $\vec{\pi}_r, r = 1, \ldots, R$ from the labeled and formerly unlabeled images. Indeed, each training class r is composed now of n_r^l labeled images and n_r^u formerly unlabeled images which will be used to estimate $\vec{\pi}_r$. The *complete-data log-likelihood* function associated to each class r is then:

$$L(\mathcal{I}^r, \Theta, \mathcal{Z}) = \sum_{i \in \mathcal{I}^l} \sum_{j=1}^{M} Z_{ij} \log \Big(p_j p(\mathcal{I}_i | \xi_j) \Big) + \lambda \sum_{i \in \mathcal{I}^u} \sum_{j=1}^{M} Z_{ij} \log \Big(p_j p(\mathcal{I}_i | \xi_j) \Big) \qquad (42)$$

where $\mathcal{I}^r$ is the set of images in class r and we have $\mathcal{I}^r = \mathcal{I}^l \cup \mathcal{I}^u$, where $\mathcal{I}^l$ and $\mathcal{I}^u$ represents respectively the labeled and formerly unlabeled images in class r. $\lambda \in [0, 1]$ is a parameter introduced in order

to decrease the contribution of the unlabeled images to parameter estimation [32]. The maximization of Eq.(42) is performed exactly as Eq.(7) and the algorithm in Section 2.4 is used to estimate the parameters. Figure 2.3 shows the effect of introducing 1000 unlabeled images, by varying the amount of labeled training images and by taking $\lambda = 0.3$, on the classification accuracy achieved by the MGDM. From the experimental results, we can see clearly that the introduction of the unlabeled images can improve the classification accuracy especially when the number of labeled images is small. Indeed, according to our experiments, the classification accuracy improvement was statistically significant just when the number of labeled images was smaller than 18550. Figure 2.4 shows the influence of λ on the classification accuracy when we use 6425 labeled images and 1000 unlabeled images. Note that when λ is close to zero, the unlabeled images will have little influence and the classification accuracy is close to that obtained in the previous section. When $\lambda = 1$, the labeled and unlabeled images will have the same weight.

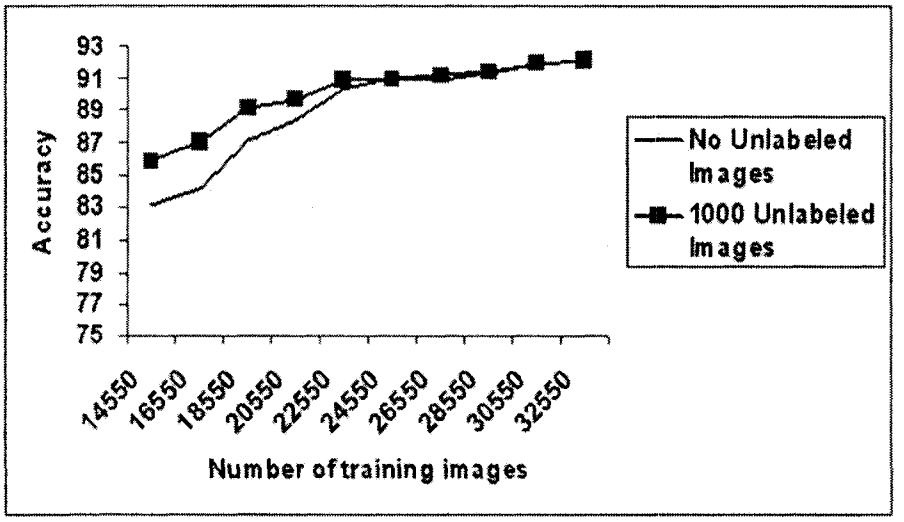

Figure 2.3: Classification accuracy with and without 1000 unlabeled images.

Image Retrieval

As we have mentioned at the beginning of this section, categorizing images databases facilitates image retrieval by searching only the cluster or the category that is closest to a given image query. As in [70],

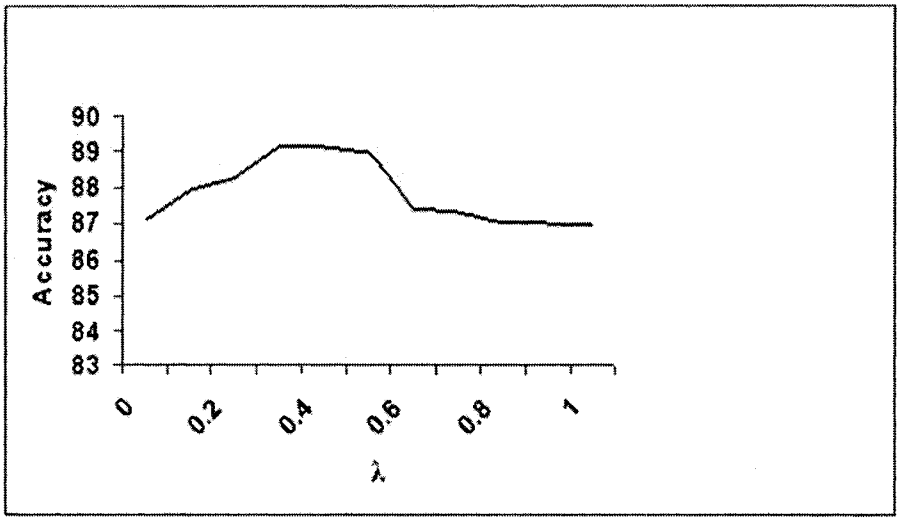

Figure 2.4: The effect of varying λ on the classification accuracy when the number of labeled and unlabeled images is 18550 and 1000, respectively.

we follow two steps to retrieve images that are similar to our query. First, we use the *a posteriori* probabilities to decide the nearest category to the image query. Second, we use the cosine similarity to find the most similar images to our query within the closest category. Of course, this similarity measure was applied after applying the preprocessing step described by Eq.(33). To compute the retrieval accuracies, we have used the following measures

$$\text{precision} = \frac{\text{number of relevant retrieved images}}{\text{total number of retrieved images}} \tag{43}$$

$$\text{recall} = \frac{\text{number of relevant retrieved images}}{\text{total number of relevant images}} \tag{44}$$

We took 1000 images from each class and each image was used as a query. Then, the measures in Eqs.43 and 44 were averaged over all the queries. Tables 2.4 and 2.5 present the retrieval rates, when 25, 50, 75 and 100 images were retrieved from the database in response to a query, in term of precision and recall obtained when the MGDM, MDM and multinomial mixture were used. The results shown in these tables clearly indicate that the MGDM offers better modeling capabilities.

Model	No. of retrieved images			
	25	50	75	100
MGDM	0.92	0.90	0.85	0.75
MDM	0.88	0.86	0.82	0.70
Multinomial mixture	0.80	0.76	0.66	0.61

Table 2.4: Precision obtained for the images database.

Model	No. of retrieved images			
	25	50	75	100
MGDM	0.23	0.43	0.61	0.74
MDD mixture	0.22	0.41	0.58	0.67
Multinomial mixture	0.20	0.35	0.47	0.56

Table 2.5: Recall obtained for the images database.

CHAPTER 3

Integrating Spatial and Color Information in Images Using A Generative Statistical Framework

Color histograms have been widely and successfully used in many computer vision and image processing applications. However, they do not include any spatial information. In this chapter, we propose a statistical model to integrate both color and spatial information. Our model is based on finite multiple-Bernoulli mixtures. For the estimation of the model's parameters, we use a maximum a posteriori (MAP) approach through deterministic annealing expectation maximization (DAEM). Smoothing priors on the components parameters are introduced to stabilize the estimation. The selection of the number of clusters is based on stochastic complexity. The results show that our model achieves good performance in a specific image classification problem (city vs. landscape).

3.1 Introduction

Multimedia data are undergoing an important expansion in terms of volume and variety [77]. Color histograms are one of the most important and used techniques for image representation [7]. Because of their efficiency and effectiveness, histograms have been used as features vectors to represent images in many applications such as content-based image and video retrieval [78] and indexing [79], object localization [80] and identification [7], image segmentation [28], and video cut detection [81]. This can be explained by their trivial computation and the fact that histograms are invariant to translation

and rotation about the viewing axis, and provide a stable object recognition in the presence of occlusions and over views change [7]. For instance, a small number of histograms can represent adequately a three-dimensional object [7]. However, histograms do not include any spatial information which is an important issue in the human visual perception [82, 83]. Indeed, images with completely different appearance and spatial colors organization may have similar histograms. Many approaches have been proposed to consider the spatial information [12, 13, 84]. A successful approach to tackle this problem and introduce the spatial information was proposed by Pass and Zabih [9]. Their technique is called histogram refinement and imposes additional constraints on histograms. These constraints are introduced by subdividing the set of pixels having a given color into classes based on local features. A possible subdivision can be based on pixels spatial coherence where a pixel is considered as coherent if it is a part of a large group of connected pixels of the same color, and incoherent if not [9, 10]. In this chapter, we propose a statistical framework to model the histogram refinement technique. Our framework is based on finite multiple-Bernoulli mixture models widely used in the case of document clustering [85] and recently introduced for image and video annotation [86]. We give an analytical solution for computing the evidence of these mixtures which is used to select the optimal number of the components.

3.2 Model Learning

3.2.1 The Model

Let I a $K \times L$ image considered to be a set of pixels $\{X_{lk}, l = 1, \ldots, L; k = 1, \ldots, K\}$, where X_{lk} is the pixel in position (l, k) of image I. The colors in I are quantized into C colors $c_1, \ldots, c_C$. The color histogram $H = (h_{c_1}, \ldots, h_{c_C})$ is a discrete vector of counts in which each h_c represents the number of pixels of color c. According to its histogram, a given image can be represented by the following probability distribution

$$p(I|\vec{\pi}) = \prod_{c=c_1}^{c_C} \prod_{l=1}^{L} \prod_{k=1}^{K} \pi_c^{\delta(X_{lk}=c)} = \prod_{c=c_1}^{c_C} \pi_c^{h_c} \tag{1}$$

which is a multinomial distribution with parameters $\vec{\pi} = \{\pi_{c_1}, \ldots, \pi_{c_C}\}$ and $\delta(X_{lk} = c)$ is an indicator function.

By considering the histogram refinement technique, a given image can be represented by a color coherence vector (CCV) [9, 10] $\langle (f_{c_1}, h_{c_1} - f_{c_1}), \ldots, (f_{c_C}, h_{c_C} - f_{c_C}) \rangle$ which is a vector of pairs, one for each color. In each pair $(f_c, h_c - f_c)$, f_c represents the number of pixels having color c and coherent (i.e the pixel is a part of a large group of connected pixels of the same color). Thus, an image can be

viewed as a collection of samples from a multiple-Bernoulli distribution. Indeed, we can assume that we sample from a multiple-Bernoulli distribution once for each pixel in the image, where each binary trial corresponds to the event that a pixel with a given color is coherent or not. Modeling the image in this manner gives us the following

$$p(I|\vec{\pi}) = \prod_{k=1}^{K}\prod_{l=1}^{L}\prod_{c=c_1}^{c_C}\left[\left(\pi_c\right)^{r_{kl}}\left(1-\pi_c\right)^{1-r_{kl}}\right]^{\delta(X_{lk}=c)} = \prod_{c=c_1}^{c_C}\pi_c^{f_c}(1-\pi_c)^{h_c-f_c} \quad (2)$$

where $r_{kl} = 1$ if pixel X_{lk} is coherent and 0 if not. Note that this distribution allows fine distinction between images that cannot be made when images are modeled using the multinomial distribution in Eq.(1). Indeed, consider two images I_1 and I_2 containing the same number of pixels having histograms $H^1 = (h^1_{c_1}, \ldots, h^1_{c_C})$, $H^2 = (h^2_{c_1}, \ldots, h^2_{c_C})$, respectively; and coherence vectors $CCV^1 = \langle(f^1_{c_1}, h^1_{c_1} - f^1_{c_1}), \ldots, (f^1_{c_C}, h^1_{c_C} - f^1_{c_C})\rangle$ and $CCV^2 = \langle(f^2_{c_1}, h^2_{c_1} - f^2_{c_1}), \ldots, (f^2_{c_C}, h^2_{c_C} - f^2_{c_C})\rangle$, respectively. Suppose that we would like to compare these images. Typically discrete distributions are compared using the Kullback−Leibler (KL) distance. Using Eq.(1) to model images, we have:

$$KL(p(I_1), p(I_2)) = \sum_{c=c_1}^{c_C}\frac{h^1_c}{H}\log(\frac{h^1_c}{h^2_c}) \quad (3)$$

where H is the summation of H^1 and H^2. Using Eq.(2) to model images, we have:

$$KL(p(I_1), p(I_2)) = \sum_{c=c_1}^{c_C}\left[\frac{f^1_c}{H}\log(\frac{f^1_c}{f^2_c}) + \frac{h^1_c - f^1_c}{H}\log(\frac{h^1_c - f^1_c}{h^2_c - f^2_c})\right] \quad (4)$$

Then, if the two images have the same histogram there will not be any distinction between them when using Eq.(3) which will be equal to zero. On the other hand, it is clear that Eq.(4) will be greater than 0 and then creates a finer distinction. For more flexibility and in order to improve the distinction between the image pixels, it is better to consider a finite mixture model of multiple-Bernoulli distributions to represent the image which gives us the following

$$p(I|\Theta) = \sum_{j=1}^{M} p_j \prod_{c=c_1}^{c_C}\pi_{jc}^{f_c}(1-\pi_{jc})^{h_c-f_c} \quad (5)$$

where p_j ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$) are the mixing proportions and $\Theta = (p_1, \ldots, p_M, \vec{\pi}_1, \ldots, \vec{\pi}_M)$.

3.2.2 ML and MAP estimation

Given a set of N images $\mathcal{I} = \{I_1, \ldots, I_N\}$, where each image is represented by a coherence vector $\langle(f_{ic_1}, h_{ic_1} - f_{ic_1}), \ldots, (f_{ic_C}, h_{ic_C} - f_{ic_C})\rangle$, the loglikelihood corresponding to our mixture model

given by Eq.(5) is

$$\log p(\mathcal{I}|\Theta) = \log \prod_{i=1}^{N} p(I_i|\Theta) = \sum_{i=1}^{N} \log \sum_{j=1}^{M} p_j p(I_i|\vec{\pi}_j) \tag{6}$$

A known method to estimate the parameters is the ML approach given by $\hat{\Theta}_{ML} = \arg\max_{\Theta}\{\log p(\mathcal{I}|\Theta)\}$. Another widely used approach is based on the Bayesian MAP criterion and it is given by $\hat{\Theta}_{MAP} = \arg\max_{\Theta}\{\log p(\mathcal{I}|\Theta) + \log p(\Theta)\}$, where $p(\Theta)$ is a prior for the mixture parameters. Note that the maximization in both cases is done with respect to the constraints over the mixing parameters. The usual choice for obtaining ML and MAP estimation of mixture parameters is the EM algorithm which is a general approach in the presence of incomplete data [56]. In EM, the "complete" data are considered to be $Y_i = \{I_i, \vec{Z}_i\}$, where $\vec{Z}_i = (Z_{i1}, \ldots, Z_{iM})$ with $Z_{ij} = 1$ if I_i belongs to class j and 0 otherwise, constituting the "missing" data. The relevant assumption is that the density of an image I_i given $\vec{Z}_i$ is given by $\prod_{j=1}^{M} p(I_i|\vec{\pi}_j)^{Z_{ij}}$. The resulting *complete-data log-likelihood* is

$$L(\mathcal{I}, \Theta, \mathcal{Z}) = \sum_{i=1}^{N} \sum_{j=1}^{M} Z_{ij} \log \Big(p_j p(I_i|\vec{\pi}_j) \Big) \tag{7}$$

The EM algorithm produces a sequence of estimates $\{\Theta^t, t = 0, 1, 2 \ldots\}$ by applying two steps in alternation

1. **E-step:** Compute $\hat{Z}_{ij}^{(t)} = \frac{p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}}{\sum_{j=1}^{M} p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}}$ given the parameter estimates from the initialization.
2. **M-step:** Update the parameter estimates according to
   $\Theta^{(t+1)} = \arg\max_{\Theta} \left\{ \sum_{i=1}^{N} \sum_{j=1}^{M} \hat{Z}_{ij}^{(t)} \log(p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}) \right\}$
   in the case of ML estimation, or
   $\Theta^{(t+1)} = \arg\max_{\Theta} \left\{ L_{MAP}(\mathcal{I}, \Theta, \mathcal{Z}) = \sum_{i=1}^{N} \sum_{j=1}^{M} \hat{Z}_{ij}^{(t)} \log(p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}) + \log(p(\Theta)) \right\}$
   for the MAP criterion.

The quantity $\hat{Z}_{ij}$ represents the conditional expectation of Z_{ij} given the image I_i and parameter vector Θ. The value $Z_{ij}^{\star}$ of $\hat{Z}_{ij}$ at a maximum of Eq.(7) is the conditional probability that observation i belongs to class j (the *posterior* probability). The EM algorithm is widely used in the case of finite mixture models estimation. However, it highly depends on initialization and it suffers from a local maxima problem because of the multimodal nature of the likelihood when we deal with mixture models [56]. Different extensions were proposed to overcome this problem [56] and one of the most successful extension of the EM was the deterministic annealing method [87] which has been used to avoid the initialization dependence and poor local maxima. Deterministic annealing is obtained by modifying the

E-step in which we compute the following parameterized variant of the original posterior probability and is given by [87]

$$\hat{w}_{ij}^{(t)} = \frac{\left(p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}\right)^{\tau}}{\sum_{j=1}^{M}\left(p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}\right)^{\tau}} \tag{8}$$

where $\tau = \frac{1}{T}$ and T corresponds to the *computational temperature*. The algorithm starts at high temperature which is lowered during the iterations. In the M-step we update the parameter estimates according to

$$\Theta^{(t+1)} = \arg\max_{\Theta}\left\{L(\mathcal{I},\Theta,\mathcal{Z},T) = \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{w}_{ij}^{(t)}\log(p(\vec{X}_i|\vec{\alpha}_j)p(j))\right\} \tag{9}$$

for the ML criterion and according to

$$\Theta^{(t+1)} = \arg\max_{\Theta}\left\{L_{MAP}(\mathcal{I},\Theta,\mathcal{Z},T) = \sum_{i=1}^{N}\sum_{j=1}^{M}\hat{Z}_{ij}^{(t)}\log(p(I_i|\vec{\pi}_j^{(t)})p_j^{(t)}) + \log(p(\Theta))\right\} \tag{10}$$

for the MAP criterion. Using the ML approach, we obtain the following estimates for the mixture parameters (See appendix C):

$$p_j = \frac{1}{N}\sum_{i=1}^{N}\hat{w}_{ij} \tag{11}$$

$$\pi_{jc} = \frac{\sum_{i=1}^{N}\hat{w}_{ij}f_{ic}}{\sum_{i=1}^{N}\hat{w}_{ij}h_{ic}} \tag{12}$$

Note that the estimation of π_{jc} is based only on the frequencies f_{ic} and h_{ic}. It is well known, however, that estimation based only on the frequencies give "poor" estimates [27]. With the MAP, we can smooth these estimates. As a smoothing prior to the $\vec{\pi}_j = (\pi_{j1},\ldots,\pi_{jC})$, we choose a multiple-Beta distribution which is a conjugate prior [85] $p(\vec{\pi}_j) = \prod_{c=c_1}^{c_C}\frac{\Gamma(\alpha_{jc}+\beta_{jc})}{\Gamma(\alpha_{jc})\Gamma(\beta_{jc})}\pi_{jc}^{\alpha_{jc}-1}(1-\pi_{jc})^{\beta_{jc}-1}$, where α_{jc} and β_{jc} are the hyperparameters. A conjugate prior for the mixing parameters is the Dirichlet distribution $p(p_1,\ldots,p_M) = \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)}\prod_{j=1}^{M}p_j^{\eta_j-1}$, where η_j is the hyperparameters. Consequently the prior of the overall mixture model is the following product

$$p(\Theta) = \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)}\prod_{j=1}^{M}\left[p_j^{\eta_j-1}\prod_{c=c_1}^{c_C}\frac{\Gamma(\alpha_{jc}+\beta_{jc})}{\Gamma(\alpha_{jc})\Gamma(\beta_{jc})}\pi_{jc}^{\alpha_{jc}-1}(1-\pi_{jc})^{\beta_{jc}-1}\right] \tag{13}$$

Having this prior, the goal is to find the MAP parameter estimate which maximize Eq.(10). We obtain (See appendix D)

$$p_j = \frac{\sum_{i=1}^{N}\hat{w}_{ij} + (\eta-1)}{N + M(\eta-1)} \tag{14}$$

$$\pi_{jc} = \frac{\sum_{i=1}^{N}\hat{w}_{ij}f_{ic} + \alpha_{jc} - 1}{\sum_{i=1}^{N}\hat{w}_{ij}h_{ic} + \alpha_{jc} + \beta_{jc} - 2} \tag{15}$$

Note that these equations are reduced to Eqs.11 and 12 when $\eta = 1$, $\alpha_{jc} = 1$ and $\beta_{jc} = 1$ (i.e $p(\Theta)$ is uniform and then the MAP model is reduced to the ML one).

3.3 Selection of the Number of Clusters and Complete Algorithm

The choice of the number of components M may affect the flexibility of the model. For the selection of M we use integrated (or marginal) likelihood, which has been shown to be robust and efficient for models selection in the case of discrete data [88], and defined by

$$p(\mathcal{I}|M) = \int \pi(\Theta|\mathcal{I}, M)d\Theta = \int p(\mathcal{I}|\Theta, M)\pi(\Theta)d\Theta \tag{16}$$

$p(\mathcal{I}|M)$ is called also the evidence and can be viewed as an information theory measure and its minus logarithm is called stochastic complexity, which is the shortest possible code length for coding the data with respect to the chosen model form (the number of clusters in our case), by Rissanen [89] (See [90, 91] for more details). Computing the integrated likelihood is analytically intractable in practice and different approximations have been proposed [92]. Two well known approximations are the Bayesian Information Criterion (BIC) [67], which is equivalent to the first version of the minimum description length (MDL) proposed by Rissanen [93], given by

$$\log p(\mathcal{I}) \approx \log p(\mathcal{I}|\hat{\Theta}) - \frac{N_p \log N}{2} \tag{17}$$

where N_p is the number of parameters and $\hat{\Theta}$ is the posterior mode. and the Akaike Information Criterion (AIC) [94] given by

$$\log p(\mathcal{I}) \approx \log p(\mathcal{I}|\hat{\Theta}) - N_p \tag{18}$$

Another approach is the Cheeseman-Stutz approximation used in the AutoClass system [95] which suggests the use of the complete data evidence $p(\mathcal{I}, \mathcal{Z}) = \int p(\mathcal{I}, \mathcal{Z}|\Theta)\pi(\Theta)d\Theta$, where

$$\begin{aligned} p(\mathcal{I}, \mathcal{Z}|\Theta) &= \prod_{i=1}^{N}\prod_{j=1}^{M}\left[p_j \prod_{c=c_1}^{c_C} \pi_{jc}^{f_{ic}}(1-\pi_{jc})^{h_{ic}-f_{ic}}\right]^{Z_{ij}} \\ &= \prod_{j=1}^{M}\left[p_j^{n_j} \prod_{c=c_1}^{c_C} \pi_{jc}^{\sum_{i=1}^{N} Z_{ij} f_{ic}}(1-\pi_{jc})^{\sum_{i=1}^{N} Z_{ij}(h_{ic}-f_{ic})}\right] \end{aligned} \tag{19}$$

where n_j is the number of vectors in cluster j. Then, we have

$$\begin{aligned} p(\mathcal{I}, \mathcal{Z}|\Theta)\pi(\Theta) &= Const \prod_{j=1}^{M}\left[p_j^{n_j+\eta_j-1} \prod_{c=c_1}^{c_C} \pi_{jc}^{\alpha_{jc}-1+\sum_{i=1}^{N} Z_{ij} f_{ic}}\right. \\ &\times \left.(1-\pi_{jc})^{\beta_{jc}-1+\sum_{i=1}^{N} Z_{ij}(h_{ic}-f_{ic})}\right] \end{aligned} \tag{20}$$

Where $Const = \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)}\prod_{j=1}^{M}\prod_{c=c_1}^{c_C}\frac{\Gamma(\alpha_{jc}+\beta_{jc})}{\Gamma(\alpha_{jc})\Gamma(\beta_{jc})}$. The integral in Eq.(16) has the form of a Dirichlet integral, then

$$p(\mathcal{I},\mathcal{Z}) = Const\frac{\prod_{j=1}^{M}\Gamma(\eta_j+n_j)}{\Gamma\left(N+\sum_{j=1}^{M}\eta_j\right)}\prod_{j=1}^{M}\prod_{c=c_1}^{c_C}\frac{\Gamma\left(\alpha_{jc}+\sum_{i=1}^{N}Z_{ij}f_{ic}\right)\Gamma\left(\beta_{jc}+\sum_{i=1}^{N}Z_{ij}(h_{ic}-f_{ic})\right)}{\Gamma\left(\alpha_{jc}+\beta_{jc}+\sum_{i=1}^{N}Z_{ij}h_{ic}\right)} \tag{21}$$

Interesting discussion about estimating the evidence in the case of finite mixture models can be found in [88]. Having Eq.(21) in hand, the complete algorithm is as follows:

Algorithm

For each candidate value of M:

1. Set $\tau \leftarrow \tau_{min}$ ($\tau_{min} \ll 1$), choose an initial estimate $\Theta^{(0)}$ and set $t \leftarrow 0$
2. Iterate the two following steps until convergence:
   (a) E-Step: Compute $\hat{w}_{ij}^{(t)}$ using Eq.(8)
   (b) M-Step: Update the $p_j^{(t)}$ using Eq.(14) and the $\pi_{jc}^{(t)}$ using Eq.(15)
3. Increase τ ($\tau \leftarrow \tau \times \Delta$)
4. If $\tau \leq 1$, set $t \leftarrow t+1$, go to step 3.
5. Calculate the associated evidence using Eq.(21).
6. Select the optimal model M^* such that: $M^* = \arg\max_M \log p(\mathcal{I},\mathcal{Z})$

Experimentally, we have concluded that τ_{min} set to 0.04 is enough which is the same confirmation reached in [96]. For the temperature, we used $\Delta = 5$ which is a choice that gives good results and corresponds to 3 phases as in [96].

3.4 Experimental Results: Image Classification (city vs. Landscape)

The main goal of this section is to show the importance of the introduction of spatial information through the comparisons of mixture of multinomials [1], used to model the color information, and our model based

[1] See [27] for details about ML and MAP estimation in the case of multinomial mixtures.

on mixtures of multiple-Bernoulli used to model both the color and spatial information. We also designed experiments to compare ML and MAP estimation and the different selection criteria discussed in Section 3.3. Images are represented in the RGB [2] color space. The color space is discretized in 64 distinct colors in the image. A pixel is considered coherent if the size of its connected components exceeds the fixed value of 300. Following [9], we have scaled all images to contain a total number of pixels equal to 38,978, so a region is considered as coherent if its area is about 1% of the image.

With the amount of digital information growing rapidly, the need for efficient automatic images categorization techniques has increased. Images categorization facilitates navigation and content-based images retrieval, and at the same time provides tools for continual maintenance as images categories grow in size. In this section, we propose a categorization approach based on our statistical model. We mainly focus on a specific interesting organization problem: city images vs. landscape images for which coherence vectors were shown to give good results [97]. Our experimental study is conducted on an image database consisting of 30,000 images (15,000 city images and 15,000 landscape images) collected from various sources. Figure 3.1 shows examples of images from both classes. Note that compared to city images, landcape images have relatively constant colors.

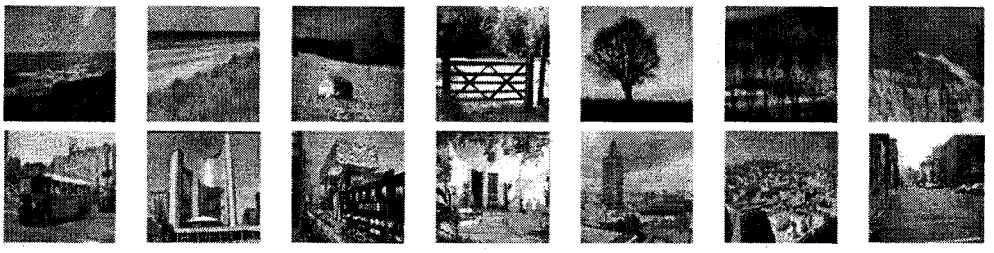

Figure 3.1: Sample images from each group. Row 1: Landscape images, Row 2: City images.

Our main problem in this section can be defined as follows: Given an input image, assign it to either the city or landscape class. The assignment is based on a set of training images which are already labeled. All training images are passed through the color coherence vectors computation stage, and then through the mixture's parameters estimation stage, in which the color coherence vectors are modeled as multiple-Bernoulli mixtures. After this stage, city and landscape classes are represented by M_{city}- and $M_{landscape}$-components multiple-Bernoulli mixtures $p(city|\Theta_{city})$ and $p(landscape|\Theta_{landcape})$, respectively. Finally, the classification stage uses the Bayesian decision rule, to determine to which class

[2] We have also tested the representation of the image colors in the HSV and LAB spaces and we did not remark much changes in the results.

Table 3.1: Classification accuracies for training set.

	ML+BIC	ML+C-S	ML+AIC	MAP+BIC	MAP+C-S	MAP+AIC
Multiple-Bernoulli mixture	93.49	94.92	94.33	95.86	97.93	95.77
Multinomial mixture	85.87	86.83	85.12	88.54	89.11	88.24

each image will be assigned, given as following: image I is assigned to class "City" if $p(I|\Theta_{city}) > p(I|\Theta_{landscape})$ and to class "landscape" otherwise. In our experiments we have used 10,000 images for training (5,000 city images and 5,000 landscape images). Figure 3.2 shows the number of clusters found for each training class when we apply our algorithm with the different selection criteria (BIC, Cheeseman-Stutz approximation (C-S) and AIC). Tables 3.1 and 3.2 shows the classification results for

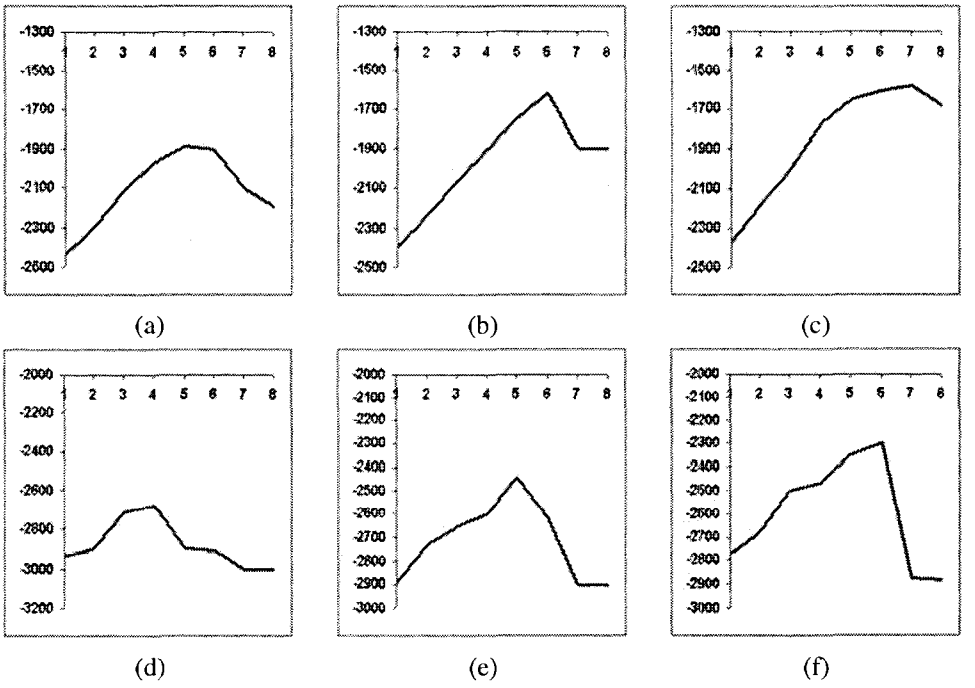

Figure 3.2: Number of clusters found for the two training classes. (a) Landscape with BIC, (b) Landscape with Cheeseman-Stutz approximation, (c) Landscape with AIC, (d) City with BIC, (e) City with C-S approximation, (f) City with AIC.

the training and test sets, respectively, when applying the multinomial and multiple-Bernoulli mixtures using ML and MAP estimation with different selection criteria. The best accuracies of 97.93% and 90.33% (which corresponds to 207 and 1934 misclassified images for the training and test sets, respectively) were obtained with the multiple-Bernoulli mixture with MAP estimation and selection based on

Table 3.2: Classification accuracies for test set.

	ML+BIC	ML+C-S	ML+AIC	MAP+BIC	MAP+C-S	MAP+AIC
Multiple-Bernoulli mixture	85.48	86.67	85.98	89.45	90.33	88.87
Multinomial mixture	77.87	78.54	77.32	80.07	81.97	79.32

Table 3.3: Confusion matrices for multiple-Bernoulli mixture for test set with. (a) MAP + C-S, (b) MAP + BIC, (c) MAP + AIC.

	City	Landscape
City	8879	1121
Landscape	813	9187

(a)

	City	Landscape
City	8638	1262
Landscape	848	9152

(b)

	City	Landscape
City	8701	1299
Landscape	927	9073

(c)

C-S criterion. It is clear also that the best results (over 95% and 88% for the training and test sets, respectively) were obtained using multiple-Bernoulli mixtures, with MAP estimation, which show that the spatial color distribution can be effective to discriminate between landscape and city images. Tables 3.3 and 3.4 show the confusion matrices for both the training and test sets using multiple-Bernoulli mixtures with MAP estimation and different selection criterion. From these tables, we can conclude that the selected number of clusters can affect the performance and that the C-S criterion is the best one in our case. Figure 3.3 shows the classification accuracies for the test set as a function of the number of images in the training set. According to this figure, increasing the number of images in the training set improves the classification accuracy which is actually an expected result showing that our model has good capacities to learn when additional images are introduced.

Table 3.4: Confusion matrices for multiple-Bernoulli mixture for training set. (a) MAP + C-S, (b) MAP + BIC, (c) MAP + AIC.

	City	Landscape
City	4858	142
Landscape	65	4935

(a)

	City	Landscape
City	4707	293
Landscape	121	4879

(b)

	City	Landscape
City	4702	298
Landscape	125	4875

(c)

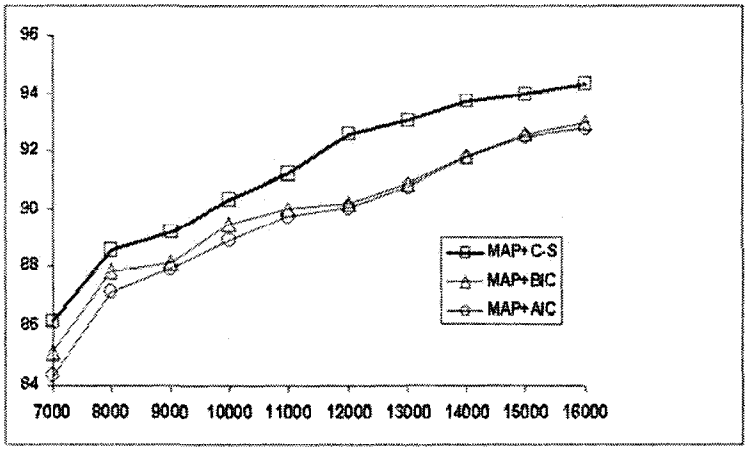

Figure 3.3: Accuracy, using Multiple-Bernoulli mixture with MAP estimation and different selection criteria, as a function of the number of images in the training set

CHAPTER 4

Conclusions

This thesis has presented a finite mixture to model discrete data, and a statistical model to add spatial information to the well known image color histogram. Our algorithms are a follow up to some previous work and they achieve significantly better results.

In chapter two, we have proposed, discussed and evaluated a novel finite mixture to model discrete data. This mixture model is based on both the generalized Dirichlet and the multinomial distributions. The recently proposed multinomial Dirichlet mixture has turned out to be a special case. We have also addressed the problem of the mixture parameters estimation. We have developed an EM algorithm accelerated by a Newton-Raphson step. The results obtained are very promising and show the merit of our model. Our proposed model is powerful and flexible enough to be adapted to a broad variety of applications where discrete data plays an important role such as information retrieval and filtering, natural language processing and bioinformatics. We intend to use the proposed MGDM for text classification.

In chapter three, a statistical model to add spatial constraints to the image histogram has been proposed. Each image pixel with a given color is considered to be coherent or not and then modeled as a multiple-Bernoulli mixture. All the model parameters are estimated using the MAP approach through the DAEM algorithm. To assess the capabilities of this model, experiments have been carried out in a specific image classification problem. Indeed, as was noted by Pass and Zabih [9], the pixels of a given bucket can be subdivided also into more than two classes. The subdivision could be based on many possible features such as texture, orientation, distance from the nearest edge, relative brightness and intensity gradients.

APPENDIX A

$$
\begin{aligned}
p(\vec{X},\vec{\pi}|\vec{\xi}) &= \prod_{v=1}^{V} p(\vec{X}|\vec{\pi})p(\vec{\pi}|\vec{\xi}) = \pi_v^{f_v} \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v,\beta_v)} \pi_v^{\alpha_v-1}(1-\sum_{l=1}^{v}\pi_l)^{\gamma_v} \\
&= \pi_1^{\alpha_1+f_1-1}(1-\pi_1)^{\beta_1-\alpha_2-\beta_2}\pi_2^{\alpha_2+f_2-1}(1-\pi_1-\pi_2)^{\beta_2-\alpha_3-\beta_3}\ldots \\
&\times \pi_{V-1}^{\alpha_{V-1}+f_{V-1}-1}(1-\pi_1-\ldots-\pi_{V-1})^{\beta_{V-1}-1}(1-\pi_1-\ldots-\pi_{V-1})^{f_V} \\
&\times \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v,\beta_v)} \\
&= \pi_1^{\alpha_1+f_1-1}(1-\pi_1)^{(\beta_1+f_2+\ldots+f_V)-(\alpha_2+f_2)-(\beta_2+f_3+\ldots+f_V)} \\
&\times \pi_2^{\alpha_2+f_2-1}(1-\pi_1-\pi_2)^{(\beta_1+f_3+\ldots+f_V)-(\alpha_3+f_3)-(\beta_3+f_4+\ldots+f_V)}\ldots \\
&\times \pi_{V-1}^{\alpha_{V-1}+f_{V-1}-1}(1-\pi_1-\ldots-\pi_{V-1})^{\beta_{V-1}+f_V-1} \\
&\times \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v,\beta_v)} \\
&= \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v,\beta_v)} \pi_v^{\alpha_v'-1}(1-\sum_{l=1}^{v}\pi_l)^{\gamma_v'}
\end{aligned}
$$

where $\alpha_v' = \alpha_v + f_v$ and $\beta_v' = \beta_v + f_{v+1} + \ldots + f_V$ for $v = 1,\ldots,V-1$, $\gamma_v' = \beta_v' - \alpha_{v+1}' - \beta_{v+1}'$ for $v = 1,\ldots,V-2$ and $\gamma_{V-1}' = \beta_{V-1}' - 1$.

APPENDIX B

$$
\begin{aligned}
\hat{\pi}_w &= E[\pi_w|\vec{X};\Theta] = \int_{\vec{\pi}} p_w \frac{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{jv},\beta_{jv})} \pi_v^{\alpha'_{jv}-1}(1-\sum_{l=1}^{v}\pi_l)^{\gamma'_{jv}}}{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{B(\alpha'_{jv},\beta'_{jv})}{B(\alpha_{jv},\beta_{jv})}} \\
&= \frac{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{jv},\beta_{jv})} \int_{\vec{\pi}} \prod_{v=1}^{V-1} \pi_v^{\alpha'_{jv}+\delta(v=w)-1}(1-\sum_{l=1}^{v}\pi_l)^{\gamma'_v}}{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{B(\alpha'_{jv},\beta'_{jv})}{B(\alpha_{jv},\beta_{jv})}} \\
&= \frac{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{jv},\beta_{jv})} \prod_{v=1}^{w-1} B(\alpha'_{jv},\beta'_{jv}+1) B(\alpha'_{jv}+1,\beta'_{jv}) \prod_{v=w+1}^{V-1} B(\alpha'_{jv},\beta'_{jv})}{\sum_{j=1}^{M} p_j \prod_{v=1}^{V-1} \frac{B(\alpha'_{jv},\beta'_{jv})}{B(\alpha_{jv},\beta_{jv})}} \\
&= \frac{\sum_{j=1}^{M} p_j p(\vec{X}|\xi_j) \frac{\alpha'_{jw}}{\alpha'_{jw}+\beta'_{jw}} \prod_{k=1}^{w-1} \frac{\beta'_{jk}}{\alpha'_{jk}+\beta'_{jk}}}{\sum_{j=1}^{M} p_j p(\vec{X}|\xi_j)} = \sum_{j=1}^{M} p(j|\vec{X};\xi_j) \frac{\alpha'_{jw}}{\alpha'_{jw}+\beta'_{jw}} \prod_{k=1}^{w-1} \frac{\beta'_{jk}}{\alpha'_{jk}+\beta'_{jk}}
\end{aligned}
$$

APPENDIX C

With ML, we have to maximize $L(\mathcal{I}, \Theta, \mathcal{Z}, T)$. Computing its derivative w.r.t π_{jc}, we obtain

$$\begin{aligned}
\frac{\partial L(\mathcal{I}, \Theta, \mathcal{Z}, T)}{\partial \pi_{jc}} &= \sum_{i=1}^{N} \frac{\partial}{\partial \pi_{jc}} \left[\hat{w}_{ij} \log(\prod_{c=c_1}^{c_C} \pi_{jc}^{f_{ic}} (1 - \pi_{jc})^{h_{ic} - f_{ic}}) \right] \\
&= \sum_{i=1}^{N} \frac{\partial}{\partial \pi_{jc}} \left[\hat{w}_{ij} \left(f_{ic} \log(\pi_{jc}) + (h_{ic} - f_{ic}) \log(1 - \pi_{jc}) \right) \right] \\
&= \sum_{i=1}^{N} \left[\hat{w}_{ij} \left(\frac{f_{ic} - h_{ic} \pi_{jc}}{\pi_{jc}(1 - \pi_{jc})} \right) \right]
\end{aligned}$$

which gives us:

$$\pi_{jc} = \frac{\sum_{i=1}^{N} \hat{w}_{ij} f_{ic}}{\sum_{i=1}^{N} \hat{w}_{ij} h_{ic}}$$

APPENDIX D

With MAP, we have to maximize $L_{MAP}(\mathcal{I}, \Theta, \mathcal{Z}, T)$. Note that we have to introduce a langrange multiplier Λ to incorporate the constraint $\sum_{j=1}^{M} p_j = 1$. Computing its derivative w.r.t p_j, we obtain

$$\begin{aligned}
& \frac{\partial\left[L_{MAP}(\mathcal{I}, \Theta, \mathcal{Z}, T) + \Lambda(1 - \sum_{j=1}^{M} p_j)\right]}{\partial p_j} \\
= & \sum_{i=1}^{N} \frac{\partial}{\partial p_j}\left[\hat{w}_{ij} \log p_j + (\eta - 1) \log p_j + \Lambda(1 - p_j)\right] \\
= & \sum_{i=1}^{N} \left[\frac{\hat{w}_{ij} + (\eta - 1)}{p_j} - \Lambda\right] = 0
\end{aligned}$$

which gives us:

$$p_j = \frac{\sum_{i=1}^{N} \hat{w}_{ij} + (\eta - 1)}{\Lambda} \tag{1}$$

Taking the derivative w.r.t Λ, we obtain

$$\frac{\partial\left[L_{MAP}(\mathcal{I}, \Theta, \mathcal{Z}, T) + \Lambda(1 - \sum_{j=1}^{M} p_j)\right]}{\partial \Lambda} = 1 - \sum_{j=1}^{M} p_j = 0$$

The previous two equations gives us

$$\sum_{j=1}^{M} \frac{\sum_{i=1}^{N} \hat{w}_{ij} + (\eta - 1)}{\Lambda} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{M} \hat{w}_{ij} + \sum_{j=1}^{M} (\eta - 1)}{\Lambda} = 1 \tag{2}$$

since $\sum_{j=1}^{M} \hat{w}_{ij} = 1$, we obtain

$$\Lambda = N + M(\eta - 1) \tag{3}$$

Then

$$p_j = \frac{\sum_{i=1}^{N} \hat{w}_{ij} + (\eta - 1)}{N + M(\eta - 1)} \tag{4}$$

Appendix D.

Computing its derivative w.r.t π_{jc}, we obtain

$$\frac{\partial\left[L_{MAP}(\mathcal{I},\Theta,\mathcal{Z},T))\right]}{\partial\pi_{jc}} = \sum_{i=1}^{N}\left[\hat{w}_{ij}\left(\frac{f_{ic}-h_{ic}\pi_{jc}}{\pi_{jc}(1-\pi_{jc})}\right)\right] + \frac{\alpha_{jc}-1}{\pi_{jc}} - \frac{\beta_{jc}-1}{1-\pi_{jc}}$$

$$= \frac{1}{\pi_{jc}(1-\pi_{jc})}\left[\alpha_{jc}-1-\pi_{jc}(\alpha_{jc}+\beta_{jc}-2)+\sum_{i=1}^{N}\left[\hat{w}_{ij}\left(f_{ic}-h_{ic}\pi_{jc}\right)\right]\right] = 0$$

Which gives us:

$$\pi_{jc} = \frac{\sum_{i=1}^{N}\hat{w}_{ij}f_{ic}+\alpha_{jc}-1}{\sum_{i=1}^{N}\hat{w}_{ij}h_{ic}+\alpha_{jc}+\beta_{jc}-2}$$

List of References

[1] R. Davies, P. Heleno, B. Correia, and J. Dinis. Vip3d - an application of image processing technology for quality control in the food industry. *Proceedings of the International Conference on Image Processing*, 1:293–296, 2001.

[2] Y. Tao. Spherical transform of fruit images for online defect extraction of mass objects. *Optical Engineering*, 35(2):344–350, February 1996.

[3] P.H. Heinemann, N.P. Pathare, and C.T. Morrow. An automated inspection station for machine-vision grading of potatoes. *Machine Vision Application*, 9(1):14–19, 1996.

[4] M. Barni, V. Cappellini, and A. Mecocci. A vision system for automatic inspection of meat quality. In *Image Analysis and Processing*, pages 748–753, 1995.

[5] J. Derganc, B. Likar, R. Bernard, D. Tomazevic, and F. Pernus. Real-time automated visual inspection of color tablets in pharmaceutical blisters. *Real-Time Imaging*, 9(2):113–124, April 2003.

[6] P. Marino, V. Pastoriza, M. Santamarfa, and E. Martinez. Fuzzy image processing in quality control application. *Sixth International Conference on Computational Intelligence and Multimedia Applications*, pages 55–60, 16-18 August 2005.

[7] M. Swain and D. Ballard. Color Indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[8] M. Stricker and M. Swain. The Capacity of Color Histogram Indexing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 704–708, 1995.

[9] G. Pass and R. Zabih. Histogram refinement for content-based image retrieval. *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision*, pages 96–102, 2-4 December 1996.

[10] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *Proceedings of the fourth ACM international conference on Multimedia*, pages 65–73, New York, NY, USA, 1996. ACM.

[11] G. Pass and R. Zabih. Comparing Images Using Joint Histograms. *Multimedia Systems*, 7:234–240, 1999.

[12] P. Chang and J. Krumm. Object recognition with color cooccurrence histogram. In *Proceedings of Computer Vision and Pattern Recognition*, 1999.

[13] J. Huang, S. Ravi Kumar, M. Mitra, W. Zhu, and R. Zabih. Spatial color indexing and applications. *International Journal of Computer Vision*, 35(3):245–268, 1999.

[14] J. Huang. Color Spatial Image Indexing and Applications, Cornell University, Department of Computer Science Dissertation, 1998.

[15] H. Y. Lee, H. K. Lee, and Y. H. Ha. Spatial color descriptor for image retrieval and video segmentation. *IEEE Transactions on Multimedia*, 5(3):358–367, September 2003.

[16] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: the qbic system. pages 255–264, 2001.

[17] V. E. Ogle and M. Stonebraker. Chabot: retrieval from a relational database of images. *Computer*, 28(9):40 – 48, 1995.

[18] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233 – 254, 1996.

[19] J. S. Boreczky and L. A. Rowe. Comparison of video shot boundary detection techniques. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 2670, pages 170 – 179, San Jose, CA, USA, 1996.

[20] M. Unser. Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):118–125, 1986.

[21] R.D. Dony and S. Wesolkowski. Edge detection on color images using rgb vector angles. *IEEE Canadian Conference on Electrical and Computer Engineering*, 2:687–692 vol.2, 1999.

[22] N. Bouguila and W. Elguebaly. A generative model for spatial color image databases categorization. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 821–824, 2008.

[23] N. Bouguila and W. Elguebaly. On discrete data clustering. In *The Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 503–510, 2008.

[24] N. Bouguila and W. Elguebaly. Discrete data clustering using finite mixture models. In *Pattern Recognition. Accepted.*

[25] N. Bouguila and W. Elguebaly. A statistical model for histogram refinement. In *International Conference on Artificial Neural Networks. Accepted.*

[26] N. Bouguila and W. Elguebaly. Integrating spatial and color information in images using a generative statistical framework. In *Pattern Recognition Letters. Submitted.*

[27] 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.

[28] J. Puzicha, J. M. Buhmann and T. Hofmann. Histogram Clustering for Unsupervised Image Segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 2602–2608, 1999.

[29] A. Juan and E. Vidal. On the use of Bernoulli mixture models for text classification. *Pattern Recognition*, 35:2705–2710, 2002.

[30] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Special Interest Group on Imformation Retrieval*, pages 232–241. Springer-Verlag New York, Inc., 1994.

[31] K. W. Churchand and W. A. Gale. Poisson Mixtures. *Natural Language Engineering*, 1:163–190, 1995.

[32] K. Nigam, A. McCallum, S. Thrun and T. Mitchell. Text Classification from Labeled and Unlabeled Documents using EM. *Machine Learning*, 39(2):103–134, 2000.

[33] A. McCallum and K. Nigam. A Comparison of Event Models for Naive Bayes Text Classification. In *AAAI-98 Workshop on Learning for Text Categorization*, pages 41–48. AAAI Press, 1998.

[34] J. D. M. Rennie, L. Shih, J. Teevan, and D. R. Karger. Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 616–623. Morgan Kauffmann, 2003.

[35] R. E. Madsen, D. Kauchak and C. Elkan. Modeling Word Buristness Using the Dirichlet Distribution. In *Proceedings of the 22^{nd} International Conference on Machine Learning*, pages 545–552. ACM Press, 2005.

[36] D. Lewis. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In *Proceedings of the 10^{th} European Conference on Machine Learning*, pages 4–15. Springer-Verlag, 1998.

[37] 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*, pages 172–181. Springer, LNAI2734, 2003.

[38] N. Bouguila and D. Ziou. Improving Content Based Image Retrieval Systems Using Finite Multinomial Dirichlet Mixture. In *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing*, pages 23–32, Sao Luis, Brazil, 2004.

[39] W. M. Bolstad. *Introduction to Bayesian Statistics*. John Wiley and Sons, 2004.

[40] S. Kotz, K. W. Ng and K. Fang. *Symmetric Multivariate and Related Distributions*. London/New York: Chapman and Hall, 1990.

[41] M. H. DeGroot. *Optimal Statistical Decisions*. Wiley-Interscience, 2004.

[42] 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.

[43] N. Bouguila and D. Ziou. A Hybrid SEM Algorithm for High-Dimensional Unsupervised Learning Using a Finite Generalized Dirichlet Mixture. *IEEE Transactions on Image Processing*, 15(9):2657– 2668, 2006.

[44] G.J. McLachlan and D. Peel. *Finite Mixture Models*. New York: Wiley, 2000.

[45] N. Bouguila and D. Ziou. A Powerful Finite Mixture Model Based on the Generalized Dirichlet Distribution: Unsupervised Learning and Applications. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 280–283, 2004.

[46] 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.

[47] T. P. Minka. Bayesian Inference, Entropy, and the Multinomial Distribution. Technical Report , CMU, 2003.

[48] S. M. Katz. Distribution of Content Words and Phrases in Text and Language Modelling. *Natural Language Engineering*, 2:15–59, 1996.

[49] J. Aitchison. *The Statistical Analysis of Compositional Data.* Chapman And Hall, second edition.

[50] R. J. Connor and J. E. Mosimann. Concepts of Independence for Proportions with a Generalization of the Dirichlet Distribution. *Journal of the American Statistical Association*, 64:194–206, 1969.

[51] T. Wong. A Bayesian Approach Employing Generalized Dirichlet Priors in Predicting Microchip Yields. *Journal of the Chinese Institute of Industrial Engineering*, 22(3):210–217, 2005.

[52] R. H. Lochner. A Generalized Dirichlet Distribution in Bayesian Life Testing. *Journal of the Royal Statistical Society, B*, 37:103–113, 1975.

[53] P. F. Thall and H. G. Sung. Some Extensions and Applications of a Bayesian Startegy for Monitoring Multiple Outcomes in Clinical Trials. *Statistics in Medicine*, 17:1563–1580, 1998.

[54] T. Wong. Generalized Dirichlet Distribution in Bayesian Analysis. *Applied Mathematics and Computation*, 97:165–181, 1998.

[55] B. P. Carlin and T. A. Louis. *Bayes and Empirical Bayes Methods for Data Analysis.* Chapman & Hall/CRC, second edition, 2000.

[56] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* New York: Wiley-Interscience, 1997.

[57] F. A. Graybill. *Matrices with Applications in Statistics.* Wadsworth, California, 1983.

[58] M. Jamshidian and R.I. Jennrich. Acceleration of the EM Algorithm by using Quasi-Newton Methods. *Journal of the Royal Statistical Society (B)*, 59(3):569–587, 1997.

[59] E. Hille. *Analytic Function Theory*, volume I. Ginn & Company, 1959.

[60] K. Lange. Applications of the Dirichlet Distribution to Forensic Match Probabilities. *Genetica*, 96:107–117, 1995.

[61] R. O. Duda, P. E. Hart and D. G. Stork. *Pattern Classification*. Wiley, New York, 2001.

[62] C-T. Lin, C-B. Chen and W-H. Wu. Fuzzy Clustering Algorithm for Latent Class Model. *Statistics and Computing*, 14(4):299–310, 2004.

[63] I. S. Dhillon and D. S. Modha. Concept Decompositions for Large Sparse Text Data Using Clustering. *Machine Learning*, 42(1-2):143–175, 2001.

[64] A. Strehl, J. Ghosh and R. Mooney. Impact of Similarity Measures on Web-Pages Clustering. In *Proceedings of the AAAI-2000 Workshop of Artificial Intelligence for Web Search*, pages 58–64. AAAI/MIT Press, 2000.

[65] G. Salton and M. J. McGill. *Introduction to Modern Retrieval*. McGraw-Hill, 1983.

[66] N. Bouguila and D. Ziou. Unsupervised Selection of a Finite Dirichlet Mixture Model: An MML-Based Approach. *IEEE Transactions on Knowledge and Data Engineering*, 18(8):993–1009, 2006.

[67] G. Schwarz. Estimating the Dimension of a Model. *Annals of Statistics*, 6:461–464, 1978.

[68] Y. Rui, T. S. Huang and S-F. Chang. Image Retrieval: Current Techniques, Promising Directions and Open Issues. *Journal of Visual Communication and Image Representation*, 10:39–62, 1999.

[69] A. W. M. Smeulders, M. Worring, S. Santini, A. Guptaand and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[70] S. Medasani and R. Krishnapuram. Categorization of Image Databases for Efficient Retrieval Using Robust Mixture Decomposition. *Computer Vision and Image Understanding*, 83:216–235, 2001.

[71] N. Bouguila. Spatial Color Image Databases Summarization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–953–I–956, Honolulu, HI, USA, 2007.

[72] S. Katz. Estimation of Probabilities From Sparse Data For The Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400–401, 1987.

[73] 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.

[74] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In A. Joshi and M. Palmer, editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 310–318. Morgan Kaufmann Publishers, 1996.

[75] H. O. Hartley and J. N. K. Rao. Classification and Estimation in Analysis of Variance Problems. *Review of International Statistical Institute*, 36:141–147, 1968.

[76] N. E. Day. Estimating the Components of a Mixture of Normal Distributions. *Biometrika*, 56(3):463–474, 1969.

[77] S. Basu, A. Del Bimbo, A. H. Tewfik and H. Zhang. Introduction to the Special Issue on Multimedia Database. *IEEE Transactions on Multimedia*, 4(2):141–145, 2002.

[78] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta and R. Jain. Content-Based Image Retrieval at the End of the Early Days. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[79] J. Hafner, H. Sawhney, W. Equitz, M. Flickner and W. Niblack. Efficient Color Histogram Indexing for Quadratic Form Distance Functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7):729–736, 1995.

[80] F. Ennesser and G. Medioni. Finding Waldo, or Focus of Attention Using Local Color Information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):805–809, 1995.

[81] U. Gargi, R. Kasturi and S. H. Strayer. Performance Characterization of Video-Shot-Change Detection Methods. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(1):1–13, 2000.

[82] J. Beck. Perceptual Grouping Produced by Line Figures. *Percept Psychophys*, 2:491–495, 1967.

[83] A. Treisman and R. Paterson. A Feature Integration Theory of Attention. *Cognitive Psychology*, 12(1):97–136, 1980.

[84] H. Y. Lee, H. K. Lee and H. H. Yeong. Spatial Color Descriptor for Image Retrieval and Video Segmentation. *IEEE Transactions on Multimedia*, 5(3):358–367, 2003.

[85] D. Metzler, V. Lavrenko and W. B. Croft. Formal Multiple-Bernoulli Models for Language Modeling. In *Proceedings of the 27th Annual International ACM Conference on Research and Development in Information Retrieval*, pages 540–541, New York, NY, USA, 2004. ACM Press.

[86] S. L. Feng, R. Manmatha and V. Lavrenko. Multiple Bernoulli Relevance Models for Image and Video Annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1002–1009, 2004.

[87] N. Ueda and R. Nakano. Deterministic Annealing EM Algorithm. *Neural Networks*, 11:271–282, 1998.

[88] P. Kontkanen, P. Myllymaki, T. Silander, H. Tirri and P. Grunwald. On Predective Distributions and Bayesian Networks. *Statistics and Computing*, 10:39–54, 2000.

[89] J. J. Rissanen. Fisher Information and Stochastic Complexity. *IEEE Transactions on Information Theory*, 42(1):40–47, 1996.

[90] M. Gyllenberg, T. Koski and M. Verlaan. Classification of Binary Vectors by Stochastic Complexity. *Journal of Multivariate Analysis*, 63:47–72, 1997.

[91] B. E. Dom. MDL Estimation for Small Samples Sizes and its Application to Segmentation Binary Strings. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 280–287, 1997.

[92] D. M. Chickering and D. Heckerman. Efficient Approximations for the Marginal Likelihood of Bayesian Networks With Hidden Variables. *Machine Learning*, 29:181–212, 1997.

[93] J. J. Rissanen. Modeling by Shortest Data Description. *Automatica*, 14:445–471, 1978.

[94] A. Akaike. Information Theory and an Extension of the Maximum Likelihood Principle. In B. N. Petrox and F. Caski, editors, *Proceedings of the 2nd International Symposium on Information Theory*, pages 267–281, 1973.

[95] P. Cheeseman and J. Stutz. Bayesian Classification (AutoClass): Theory and Results. In *Advances in Knowledge Discovery and Data Mining, chapter 6*, pages 153–180. AAAI Press, 1995.

[96] C. Elkan. Clustering Documents with an Exponential-Family Approximation of the Dirichlet Compound Multinomial Distribution. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 289–296. ACM Press, 2006.

[97] A. Vailaya, A. K. Jain and H. J. Zhang. On Image Classification: City vs. Landscape. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 3–8, 1998.