# A Statistical Framework for Discrete Visual Features Modeling and Classification

**Mukti Nath Ghimire**

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Electrical and Computer Engineering) at

Concordia University

Montréal, Québec, Canada

July 2011

**CONCORDIA UNIVERSITY**
**SCHOOL OF GRADUATE STUDIES**

This is to certify that the thesis prepared

By: Mukti Nath Ghimire

Entitled: "A Statistical Framework for Discrete Visual Features Modeling and Classification"

and submitted in partial fulfillment of the requirements for the degree of

**Master of Applied Science**

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

______________________________________ Chair
Dr. R. Raut

______________________________________ Examiner, External
Dr. W. F. Xie, MIE To the Program

______________________________________ Examiner
Dr. J. Bentahar, CIISE

______________________________________ Supervisor
Dr. N. Bouguila

Approved by: ______________________________________
Dr. W. E. Lynch, Chair
Department of Electrical and Computer Engineering

____________20_____

______________________________________
Dr. Robin A. L. Drew
Dean, Faculty of Engineering and
Computer Science

# Abstract

## A Statistical Framework for Discrete Visual Features Modeling and Classification

Mukti Nath Ghimire

Multimedia contents are mostly described in discrete forms, so analyzing discrete data becomes an important task in many image processing and computer vision applications. One of the most used approaches for discrete data modeling is the finite mixture of multinomial distributions, considering that the events to model are independent. It, however, fails to capture the true nature in the case of sparse data and leads generally to poor biased estimates. Different smoothing techniques that reflect prior background knowledge are proposed to overcome this issue. Generalized Dirichlet distribution has suitable covariance structure, so it offers flexibility in parameter estimation; therefore, it has become a favorable choice as a prior. This specific choice, however, has its problems mainly in the estimation of the parameters, which appears to be a laborious task and can deteriorate the estimates accuracy when we consider the maximum likelihood (ML) approach.

In this thesis, we propose an unsupervised statistical approach to learn structures of this kind of data. The central ingredient in our model is the introduction of the generalized Dirichlet distribution mixture as a prior to the multinomial. An estimation algorithm for the parameters based on leave-one-out (LOO) likelihood and empirical Bayesian inference is developed. This estimation algorithm can be viewed as a hybrid expectation-maximization (EM) which alternates EM iterations with Newton−Raphson iterations using the Hessian matrix. We also propose the use of our model as a parametric basis for support vector machines (SVM) within a hybrid generative/discriminative framework. Through a series of experiments involving scene modeling and classification using visual words and color texture modeling, we show the efficiency of the proposed approaches.

# Acknowledgements

This thesis would not have been possible without the constant support of my thesis advisor - Dr. Nizar Bouguila. I owe my deepest gratitude to him who constantly guided, suggested and insisted me to achieve the goal.

I would like to extend my gratitude by thanking to all my colleagues for their helpful suggestions, to my loving wife for her unconditional support throughout my study, and, last but not least, to the Concordia University which gave me a platform to extend my knowledge.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Due to the advent of digital technology, storing information in multimedia forms has become almost the first choice. These are in different forms such as audio, image, video, animation and graphics. Multimedia collections are tremendously huge and are increasing day-by-day. For instance, www.flickr.com, a popular photo-sharing service on the web, acknowledges that its database has crossed five billions by September 2010; furthermore, it claims that more than three thousands pictures are uploaded every minute [3] ! This is just a small portion in whole internet multimedia contents. It becomes even bigger if we think of personal and proprietary collections in local networks. Proliferation of such multimedia contents has created a need to develop approaches and models to process, manage and categorize them, so that they can be automatically located and retrieved when it is necessary. There is always such demand for image processing and computer vision tasks since visual contents serve necessary purposes in almost all areas of science and industries such as art, medicine, geography and forensic.

Intuitively, the categorization, which is a very common task in machine learning and data mining, involves creating a statistical model which helps to sort similar type of data into same category. This topic has been extensively studied and has been applied to several tasks in various areas such as image processing, pattern recognition, machine learning, remote sensing [4], automated text categorization [4], character [5] and face recognition [6], image categorization and retrieval [7], and autonomous vehicles [8].

Finite mixture models are one of the mostly used statistical approaches for categorization [9]. Like in all other generative models, the choice of an appropriate model structure to capture the characteristics of the features is the key concern here; otherwise, wrong choice may degrade the model performance. An important step in multimedia data categorization is the extraction of features which can be continuous or discrete. For continuous data, Gaussian mixture model (GMM) is largely adopted, but recent research has shown that it fails to discover the true structure when the partitions are non-Gaussian [10] such as the cases with discrete features (see, for instance, [11]).

Different assumptions have been made in the case of discrete data. The multinomial, however, represents the state-of-the-art distribution for discrete data modeling. In spite of its popularity, recent researches have shown that it has some drawbacks such as considering that the events to model are independent [11–14]. Another important problem is the parameters estimation in the case of sparse data[1] (i.e. the estimation of the probabilities of rarely observed or unobserved occurrences) [16, 17]. The severity of this problem, which leads generally to poor biased estimates, has been widely studied by the natural language processing community, but generally ignored by image processing and computer vision researchers[2]. Different smoothing techniques have been proposed to overcome these problems [19]. The most successful approach is the use of the Dirichlet distribution as a prior, reflecting a certain background knowledge, to the multinomial which results in a completely formal statistical model [11, 13].

Indeed, there is a previously proposed framework in which finite mixture of Dirichlet distributions was used as a prior to the multinomial and applied to different applications such as texture modeling and narrowing the semantic gap for content-based image summarization and retrieval [11, 20, 21]. Recently, it is noticed that even the Dirichlet has some problems such as its

[1] This is also known as the zero-frequency problem and arises when dealing with observations that never occurred in the training data [15].

[2] A main assumption generally considered in image processing and computer vision applications is the Gaussian mixture (GM) by considering continuous features. This assumption, however, is not realistic when dealing with discrete data. Moreover, it is well-known that the normal assumption limits the ability to analyze rare events [18].

very restrictive negative covariance structure which makes its use as a prior in the case of positively correlated data inappropriate (see [21, 22] for details and discussions). These problems can be overcame by the consideration of the generalized Dirichlet distribution which is more general in covariance structure and offers more flexibility [22, 23]. This specific choice, however, has its problems namely the estimation of the parameters, which appears to be a laborious task when we consider the maximum likelihood approach, as we will show in Section 2.3.

As this thesis presents a statistical framework to model discrete image features, following background information will be helpful to understand the context of the work.

## 1.1 Background

Image categorization incorporates mainly three tasks: features extraction, modeling and classification. First, it needs extracting the features that best describe the visual contents of the image. Second, the classifier, a set of decision rules or algorithm that classifies a query image into one of the predefined classes, has to be established to get the expected classification result. Finally, features representing query image are subjected to the classifier, and query image is affected to the corresponding category.

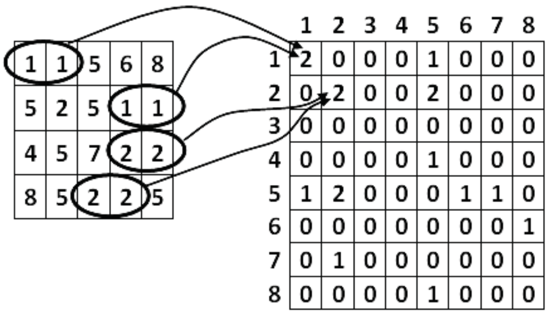
### 1.1.1 Visual Features

A very basic idea to classify an image would be to compare it directly against all categories and to find the category of the best match, but this method consumes a lot of resources and becomes an inefficient choice. So, there is always a need to represent the image by its features which still represents the image without the loss of information. Depending on its region of representation, visual features can be local or global.

#### 1.1.1.1 Global Features

Global image features such as color, shape and texture have been widely explored in the context of content-based image categorization and retrieval [24–27]. Unfortunately, they are not robust against occlusion, background clutter and other contents change. Moreover, their result is not only difficult to predict and control [28] but also, by intuition, computationally expensive when it needs handling a large image database. An example of such global feature is co-occurrence matrix.

**Co-occurrence Matrix:** Texture analysis is an important topic in image processing and computer vision field. Many approaches have been proposed to address this problem. These can be grouped under three methods: structural, statistical and signal theoretic [29]. Majority of these approaches deal with gray level images while a few of them incorporate color as well as texture information. The later approaches that combine color and texture information can be summarized into three groups: parallel, sequential and integrative [30]. Integrative approaches better combine color and texture information by taking into account the dependency between color and texture features [30, 31].



| | | | | |
|---|---|---|---|---|
| 1 | 1 | 5 | 6 | 8 |
| 5 | 2 | 5 | 1 | 1 |
| 4 | 5 | 7 | 2 | 2 |
| 8 | 5 | 2 | 2 | 5 |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 2 | 0 | 2 | 0 | 0 | 2 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5 | 1 | 2 | 0 | 0 | 0 | 1 | 1 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 7 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Figure 1.1: A typical grey scale image and its co-occurrence matrix for a displacement $\vec{d} = (1, 0)$.

A co-occurrence matrix, a second order texture measure, shows how frequently the set of pixels reoccurs in an image (see Figure 1.1). It is mostly used as an intermediate feature, and further dimension reduction is performed in computing features of the types described in [32, 33] such as energy, entropy, contrast, homogeneity and correlation.

#### 1.1.1.2 Local Features

Local interest points such as points, edges or small image patches are characteristic points where signal changes bidirectionally [34]. In other words, interest points are those which are invariant to some geometric and photometric transformations. Unlike global features, due to scale and affine invariant nature, local features-based methods have proven to be useful to solve for many problems in practical fields such as viewpoint-independent object recognition [35–37], wide baseline matching [1, 38, 39], image retrieval [40–42], video data mining [43], and texture recognition [44]. Their local nature inherits robustness to image clutter, occlusion and partial visibility, and their invariant nature provides stable representation to affine transform and lighting conditions change. All these properties make local features stable by producing a relatively repeatable representation of a particular object.

**Interest Point Detectors:** Before extracting them, the features have to be located first. Quite a few interest points detecting approaches have been developed in past few years. The earliest work can be traced back to Harris [45] who has developed a derivative-based edge and corner detector by measuring the trace and determinant of the gradient distribution matrix around interest points. Since Laplacian operator correctly extracts more candidate points [41], Mikolajczyk and Schmid [41, 46] extended it in generating scale space pyramid [47]. There are number of other approaches available[3], but Gaussian blur methods are mostly adopted. As Gaussian function[4] is considered as the best among available scale space kernels [49, 50], scale space pyramid [47] is usually

[3] An extensive survey on local invariant features and their extraction methods can be found in [48].

[4] $G(x, y, \sigma_k) = \frac{1}{2\pi(\sigma_k)^2} e^{-\frac{x^2+y^2}{2(\sigma_k)^2}}$ is a 2-D Gaussian function at scale $\sigma_k$ or with radius of blur $k\sigma$.

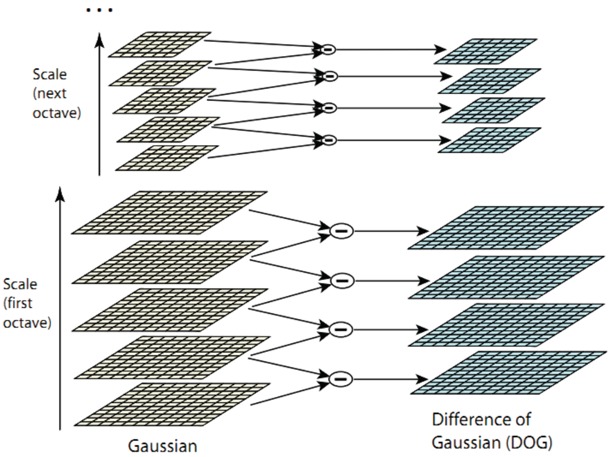


Figure 1.2: Steps in formation of DOG images. Repeated convolution of an initial image with Gaussian kernel, down sampling by factor of two after each octave $--$ doubling the sigma $\sigma$ finds next octave $--$ and then subtracting adjacent scales results pyramid of DOG images (shown in right) in scale space [1].

generated using it. After the application of scale normalized [51] $--$ scale normalization ensures average gray levels be same at all scales $--$ derivative-based operator[5] on each scale, keypoints are located according to cornerness measure (see in [48], for instance, for commonly used cornerness measures). To detect local keypoints, Lowe [1, 52] has used scale-normalized Laplacian-of-Gaussian (LOG) which is implemented by using difference-of-Gaussian (DOG) function[6].

[5] Harris and Hessian operators are typical examples and are are explained in [38, 45].

[6] DOG function can closely approximate scale-normalized LOG function, $\sigma^2\nabla^2 G$ [50]. To prove it, starting with diffusion equation $\frac{1}{2}\nabla^2 G = \frac{\partial G}{\partial t}$ where $G$ is Gaussian operator, let us replace $t = \sigma^2 \implies \partial t = 2\sigma\partial\sigma$ to parameterize in terms of $\sigma$:

$$\frac{1}{2}\nabla^2 G = \frac{\partial G}{2\sigma\partial\sigma}$$

$$\sigma\nabla^2 G = \frac{\partial G}{\partial\sigma}(\approx \frac{G(x,y,\sigma_k) - G(x,y,\sigma)}{\sigma_k - \sigma})$$

$$\sigma\nabla^2 G \approx \frac{G(x,y,\sigma_k) - G(x,y,\sigma)}{(k-1)\sigma}$$

$$(k-1)\sigma^2\nabla^2 G \approx G(x,y,\sigma_k) - G(x,y,\sigma)$$

$$(k-1)\sigma^2\nabla^2 G \approx \text{DOG}$$

The approximation error becomes zero when $k = 1$, but $k$ has no practical effect on stability of peak detection or localization; for practical purpose, therefore, $k$ is selected as $\sqrt{2}$ [1].

Pixel-by-pixel difference between two Gaussian blur images at scales $\sigma_k$ and $\sigma$ results DOG image at scale $\sigma_k$ (see Figure 1.2)[7]. Out of other derivative functions, DOG function is chosen not only because its maxima and minima incur the most stable image features but also because it is easy to compute [53]. Using similar idea of extrema location by Lindeberg [51], the local extrema among the points in neighboring scale space is located as keypoint [1].

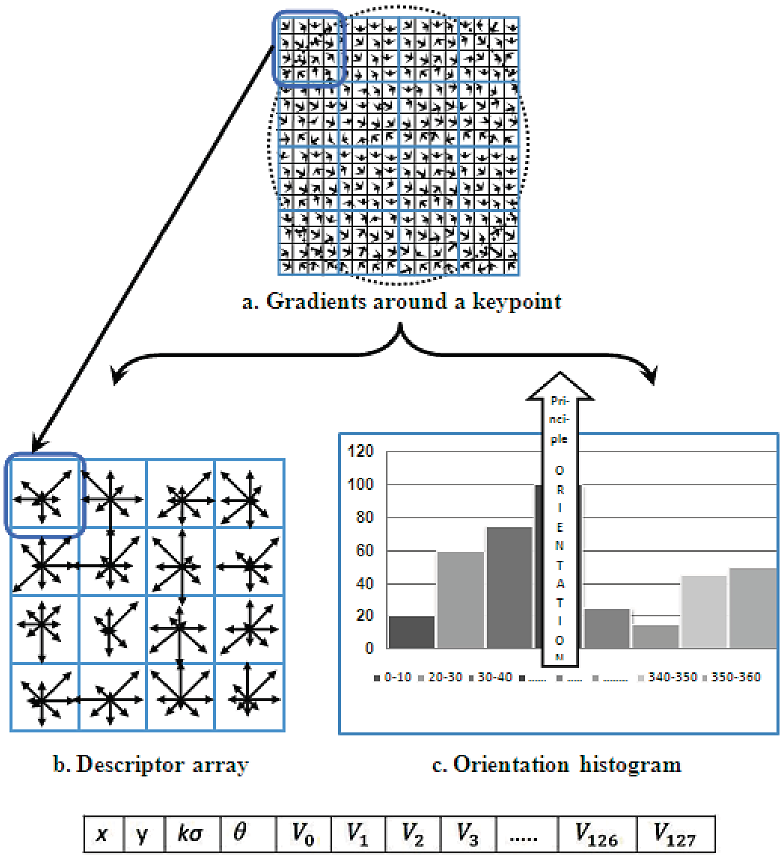
The methods discussed so far are scale invariant; they, however, are not robust against affine transformation. This problem has been addressed by developing an affine adaption process based on the second moment matrix [54]. Similar implementations can be found in [46, 55–57].

**Local Descriptors:** Assigning suitable descriptors to the local keypoints, which are invariant to class of transformations, is another necessary task. This adds the distinctiveness and robustness to the features [1]. Although numerous techniques such as gradient distribution [1, 58], Gabor wavelet [59], moment invariants [60], Harr wavelet filters [61], steerable filters [62], descriptors based on intensity [44, 63, 64] that are particularly used in texture images, and a technique inspired by biological vision [65] are suggested, scale-invariant feature transform (SIFT)-based descriptors [1] have performed the best among other available descriptors [53]. Furthermore, many approaches[8] have been suggested to improve the selectivity, robustness and cost of computation of SIFT descriptors, but Lowe's SIFT [1], which carries local gradient information of the patch around the keypoint, is extensively used, mostly cited and still regarded as de facto descriptor.

As this thesis extensively uses co-occurance matrix [69] and SIFT [1] as visual descriptors, an extended illustration on how a SIFT feature vector can be extracted from a keypoint is shown in Figure 1.3 on page 8.

[7] $D(x, y, \sigma_k)$, a DOG image at scale $\sigma_k$, equals $L(x, y, \sigma_k) - L(x, y, \sigma)$, where $L(x, y, \sigma_k) = I(x, y) * G(x, y, \sigma_k)$ is a linear $--$ Gaussian is also linear $--$ discrete scale space representation of an image $I(x, y)$: a family of signals defined for different scales $\sigma_k \forall k \in 1, 2, 3, \ldots$ and derived by convoluting the image with a Gaussian blur kernel. Note that $L(x, y, \sigma_k)$ is reduced to the image $I(x, y)$ itself for the scale zero (i.e. $k = 0$).

[8] Other variants of SIFT descriptors are also available such as principle component analysis SIFT (PCA-SIFT) [66], informative SIFT (i-SIFT) [67], and color SIFT (CSIFT) [68].



a. Gradients around a keypoint

b. Descriptor array

c. Orientation histogram

d. SIFT feature vector

Figure 1.3: An illustration showing SIFT descriptor extraction process. (a) A keypoint on the DOG image stacks and its $16 \times 16$ neighborhood image gradients are shown, and the dotted circular window signifies the scale normalization. (b) $4 \times 4$ descriptor array is rearranged from $16 \times 16$ sample array, also the length of each arrow stands for the sum of gradient magnitudes within corresponding $45°$ bin; as a result, $4 \times 4$ matrix with $8$ vectors on each cell results a $128(=8 \times 4 \times 4)$-dimensional SIFT descriptor. (c) Principle orientation, for example it is shown by a flat arrow pointing upward, is the mean of the dominant bin among 32 orientation bins (10°/bin). (d) A SIFT feature vector describes location, scale, dominant orientation and magnitude information of the keypoint.

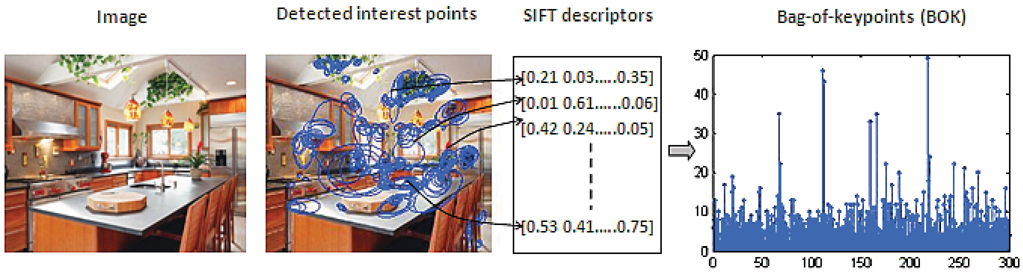


Figure 1.4: Representation of an image by BOK. (from left to right) A typical image; local keypoints are detected; a number of local SIFT descriptors are extracted on each of those keypoints; and the BOK vector representing the image is calculated.

**Bag-of-Visual Words:** Using analogy to learning methods using bag-of-words representation for text categorization [70–72] and motivated by the work of Zhu *et al.* [73], Csurka *et al.* [74] has used bag-of-keypoints (BOK) as visual words for visual categorization task. Visual vocabulary is represented by homogenous clusters that are obtained by a clustering or vector quantization, such as $K$-Means, of training features set. With all feature vectors of an image in hand, the BOK is formed by bin counts of each cluster (see Figure 1.4). This way, BOK shows the frequency of types of local image patches in the image; therefore, this approach reduces generic visual categorization problem into multi-class supervised learning.

### 1.1.2 Statistical Models

Machine learning involves the development of algorithms and techniques that help us to learn and to draw inferences on data. Creating a statistical model which captures class(es) information is a common task that all statistical machine learning methods involve. Depending on the way the model discriminates the class information, there are broadly two families of approaches for machine learning: generative and discriminative models.

#### 1.1.2.1 Generative Models

A statistical approach that explicitly models data using generative distribution $p(\mathbf{X}|\theta)$ is called generative model, where **X** is data variable and $\theta$ is the model parameter(s). Now, to classify a query datum $X_i$ into one of the several categories, a typical approach is to estimate a distribution $p(\mathbf{X}|\theta_j)$ for each of the categories $j = 1, 2, 3, ....M$, and then to classify the data to the category that has maximum posterior class probability given the data:

$$k^{th}_{category} \Leftarrow \underset{k}{\operatorname{argmax}}\, P(\theta_j|X_i) = \frac{p(X_i|\theta_j)P(\theta_j)}{P(X_i)} \tag{1.1}$$

Let's make it simple! $X_i$ falls on $k^{th}$ category if posterior probability $P(\theta_{j=k}|X_i)$ is the highest among all class posteriors $P(\theta_j|X_i)$; $j = 1, 2, 3, ......M$. We can see that prior assumption about the data is updated to posterior probability in the light of class-conditional likelihood. This is also called Bayes' rule. Examples of generative models include GMM and other types of mixture models, hidden Markov model (HMM), naïve Bayes' [75], averaged one-dependence estimators (AODE), latent Dirichlet allocation (LDA) [76], and restricted Boltzmann method [77]. In a situation where strong (naïve) independence can be assumed, the naïve Bayes' classification model is quite popular. It is often used in text categorization [78], and its classification accuracy is typically high [79].

#### 1.1.2.2 Discriminative Models

Unlike generative approaches, discriminative approaches do not model data explicitly; they, however, are concerned with defining the boundaries between the categories. The classifier is built by estimating a decision rule $f(j, X_i)$ straight from the training data. Examples of discriminative models include logistic regression, linear discriminant analysis (LDA), support vector machine (SVM), boosting, conditional random fields, linear regression, and neural networks. Discriminative approaches are implemented in wide range of application fields such as speech recognition [80], image segmentation [81], object recognition [82], and biomedical and life science [83] .

In the recent years, SVM is widely used and often known to produce state-of-the-art results for high-dimensional data [74], and finds its applications ranging from text categorization to pattern recognition [70, 84]. Data from practical problems may not always be linearly separable, so they are mapped to a space where the separation using hyperplane will be easier. The SVM classifier finds a hyperplane which separates two-class data with maximal margin [85]. The classifier's parameters are derived in such a way that margin from the closest training points to the separating hyperplane is maximized. For a given training instances $X_i,\ i = 1, 2, ...., N$, and corresponding labels or indicator vectors $Y_i$ that take values $\pm 1$, one finds a classification function as follows:

$$f(X) = Sgn(W^T X + b) \tag{1.2}$$

where $Sgn()$ is signum function. Whole SVM classifier design is the estimation of these hyperplane parameters: $W$ and $b$. To cope with this problem, kernels are in use [86].

Both generative and discriminative approaches have their own pros and cons. Generative models, for example, are easy to interpret, can be trained quickly, also can be easily extended to incorporate a new category by learning new class-conditional density [13]; it, however, may slow down the response time as these approaches often require iterative solution. Similarly, SVM, a typical discriminative approach, shows exciting results to high-dimensional data [74]; on the other side, most of the discriminative models are inherently supervised and can not be easily extended to unsupervised learning. Therefore, the choice of the approach is usually governed by the the constraints and requirements of the task in hand[9]. Current research trend is to blend good aspects from both of these approaches: the outcome is a hybrid generative/discriminative model. Some theoretical studies have shown its several advantages such as providing lower test error than both generative and discriminative techniques [88], also it has provided good solutions to various practical problems such as image classification [89], and object recognition in static images [90].

[9] Comparative study, in particular to object recognition, can be found in [87], and it indicates that both approaches have desirable properties under certain conditions.

## 1.2 Contributions

This thesis makes following contributions:

☞ **An Efficient Discrete Data Clustering Using Finite Mixture Model:** We look into the problem of discrete data modeling using finite mixture models. We propose a novel approach to enhance the parameters estimation and learning of the statistical framework, which uses a generalized Dirichlet mixture as a prior to the multinomial. During the estimation of model parameters, the iteration steps in expectation-maximization (EM) algorithm, which is based on leave-one-out (LOO) likelihood and empirical Bayesian inference, involve Newton-Raphson iteration using the Hessian matrix. With series of comparative experiments against other discrete mixtures, that involve image and texture databases modelling and classification, we verified the efficiency and merits of our proposed approach.

☞ **Integrating the Model as Parametric Basis for Hybrid Generative/Descriminative Framework:** Furthermore, we propose our model as parametric basis for SVM within a hybrid generative/discriminative framework, and we experimentally demonstrated the improvement in classification accuracy due to the new kernel.

It is noteworthy that these contributions have been published in the journal of visual communication and image representation [91].

## 1.3 Thesis Overview

The thesis is organized into four chapters:

- ❑ In Chapter 1, introduction to selected visual contents representation and a brief literature review of some contemporary approaches to model such data, which form the basis for subsequent chapters, are outlined.
- ❑ In Chapter 2, we review the multinomial assumption, and both Dirichlet and generalized Dirichlet distributions are used as priors for smoothing purposes. After suggesting a new approach for the estimation and selection of multinomial generalized Dirichlet mixture, we present a generative/disriminative framework based on our developed model and SVM.
- ❑ In Chapter 3, we discuss our experimental results in details.
- ❑ Finally in the last chapter, our proposed methodologies and contributions are summarized, and future directions are outlined.

CHAPTER 2

# Statistical Model

## 2.1 Introduction

Discrete features[1] appear in many application areas such as computer vision, image processing and pattern recognition [2, 11, 96] . As pointed in page 3, discrete data modeling with finite mixture models has some issues. To cope with that, we propose a statistical framework for discrete data modeling. We consider the use of generalized Dirichlet mixture as prior to the multinomial to model and cluster discrete visual feature vectors in the case of some interesting image representation applications.

We propose a novel approach to enhance the estimation and the learning of our statistical framework parameters. Our approach is based on the maximization of the LOO likelihood through a hybrid expectation maximization algorithm which alternates EM iterations with Newton-Raphson iterations using the Hessian matrix. The proposed model is also used for generating SVM kernel within a generative/dicriminative framework involving mixture model and SVM both in a way that it combines their respective advantages in order to take into account the discrete nature of the data. Indeed, mixing generative and discriminative approaches has attracted a lot of attention and some theoretical studies have shown its several advantages such as providing lower test error than both

[1] Examples of discrete features include color histograms [92], co-occurrence matrices [69], correlograms [93], color coherent vectors [94], and the recently proposed keyblocks (i.e. visual keywords) as an analogy to dictionaries in the case of text documents [73, 74, 95]

generative and discriminative techniques [88]. Moreover, generative/discriminate approaches have been found to be useful in many practical applications [89].

## 2.2 The Discrete Statistical Model

Let $\vec{X}_i = (X_{i1}, \ldots, X_{iD_i})$, $i = 1, \ldots, N$, be a discrete vector representing a given image, $D_i$ is the number of visual features in the image, and each variable $X_{id}, d = 1, \ldots, D_i$, takes on values on a $V$-sized visual corpus (or dictionary) that is a finite set of discrete values. Then, a classic assumption is that $\vec{X}_i$ is generated by the following model:

$$p(\vec{X}_i|\vec{\pi}) = \prod_{d=1}^{D_i}\prod_{v=1}^{V}\pi_v^{\delta(X_{id}=v)} = \prod_{v=1}^{V}\pi_v^{f_{iv}} \tag{2.1}$$

where $\delta(X_{id} = v)$ is an indicator function, $\{f_{iv}\}$ are the frequencies of values $v$ in $\vec{X}_i$ and represent the sufficient statistics, $\vec{\pi} = (\pi_1, \ldots, \pi_V)$ is the parameter vector of a multinomial, $\sum_{v=1}^{V} \pi_v = 1$.

Recent machine learning researches[2], however, have shown that the multinomial assumption as a naïve Bayes' approach has several drawbacks and suffers from the zero counts which create serious obstacles [11–14]. For instance, data sparseness problem makes the maximum likelihood (ML) approach to estimate the $\pi_v$ parameters unreliable [101]. Indeed, it is easy to show that the ML estimate is simply

$$\hat{\pi}_v = \frac{f_{iv}}{\sum_{v=1}^{V} f_{iv}} \tag{2.2}$$

Moreover, it is clear that $\hat{\pi}_v$ is zero for any feature that does not appear in $\vec{X}_i$, since the probabilities are estimated by the fraction of times the feature occurs over the total number of opportunities. The unreliability of ML estimates can be generalized for features which appear rarely (i.e. with small frequency). In order to adjust the ML estimates, a widely used approach is to modify the sample counts by augmenting them with some chosen values (i.e. pseudo-counts) and a common choice

[2] Note that the drawbacks underlying the multinomial assumption have been discussed a long time ago by statisticians (see [97–100], for instance).

is to add 1 to all frequencies[3]:

$$\hat{\pi}_v = \frac{1 + f_{iv}}{V + \sum_{v=1}^{V} f_{iv}} \tag{2.3}$$

This adjustment is actually a special case of another classic approach to prevent zero probabilities which is the consideration of a Dirichlet prior for $\vec{\pi}$:

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

where $\vec{\alpha} = (\alpha_1, \ldots, \alpha_V)$. The Dirichlet distribution depends on $V$ parameters $\alpha_1, \ldots, \alpha_V$, which are all real and positive. The choice of the Dirichlet distribution is motivated by the fact that it is closed under multinomial sampling (i.e. the Dirichlet is a conjugate prior for the multinomial) [104]. Using the Dirichlet as a prior, we can show that [11]:

$$\hat{\pi}_v = \frac{\alpha_v + f_{iv}}{\sum_{v=1}^{V} \alpha_v + \sum_{v=1}^{V} f_{iv}} \tag{2.4}$$

where $\sum_{v=1}^{V} \alpha_v$ is generally called *equivalent sample size*, since it can be interpreted as augmenting the actual frequencies by $\sum_{v=1}^{V} \alpha_v$ virtual ones [105]. Note that the last equation is reduced to Eq. 2.3 when we consider a symmetric Dirichlet, with unity concentration parameter, as a prior. In spite of its flexibility and the fact that it is conjugate to the multinomial which have led to its application in different learning approaches and techniques, the Dirichlet has restrictions: a very restrictive negative covariance matrix which violates generally experimental observations [106–108] and the variables with the same mean must have the same variance as shown in [109]. These problems can be handled by the consideration of a generalized Dirichlet as prior [2]:

$$p(\vec{\pi}|\xi) = \prod_{v=1}^{V-1} \frac{1}{B(\alpha_v, \beta_v)} \pi_v^{\alpha_v - 1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_v} \tag{2.5}$$

where the Beta function $B(\alpha_v, \beta_v) = \frac{\Gamma(\alpha_v)\Gamma(\beta_v)}{\Gamma(\alpha_v + \beta_v)}$. The generalized Dirichlet contains $2(V-1)$ parameters $\xi = (\alpha_1, \beta_1, \ldots, \alpha_{V-1}, \beta_{V-1})$, which are all real and positive, and $\gamma_v = \beta_v - (\alpha_{v+1} + \beta_{v+1})$ for $v < (V-1)$ and $\gamma_{V-1} = \beta_{V-1} - 1$. Note that the generalized Dirichlet is reduced to

[3] This choice is usually referred to as Jeffrey's estimate [102, 103, p. 293] or Laplace smoothing [19].

a Dirichlet with parameters $(\alpha_1, \ldots, \alpha_{V-1}, \alpha_V = \beta_{V-1})$ when $\beta_v = \alpha_{v+1} + \beta_{v+1}$. The particular choice of the generalized Dirichlet as a prior has several advantages which are widely discussed in [23] such as its general covariance matrix and the fact that it is also conjugate to the multinomial. Using this prior, we can show that [23]:

$$\hat{\pi}_v = \frac{\alpha_v + f_{iv}}{\alpha_v + \beta_v + n_{iv}} \prod_{l=1}^{v-1} \frac{\beta_l + n_{il+1}}{\alpha_l + \beta_l + n_{il}} \tag{2.6}$$

where $n_{il} = f_{il} + f_{il+1} + \ldots + f_{iV}$. For more flexibility we can even go further by considering a finite mixture of generalized Dirichlet distributions as a prior:

$$p(\vec{\pi}|\Theta) = \sum_{k=1}^{K} \omega_k \prod_{v=1}^{V-1} \frac{1}{B(\alpha_{kv}, \beta_{kv})} \pi_v^{\alpha_{kv}-1} (1 - \sum_{l=1}^{v} \pi_l)^{\gamma_{kv}}$$

where the parameter set $\Theta = (\vec{\omega}, \{\xi_k\})$ includes parameters from generalized Dirichlet mixtures $\xi_k = (\alpha_{k1}, \beta_{k1}, \ldots, \alpha_{kV-1}, \beta_{kV-1})$ and $\vec{\omega} = (\omega_1, \ldots, \omega_K)$ that represents the mixing parameters vector of our mixture model $\omega_k > 0$ and $\sum_{k=1}^{K} \omega_k = 1$. Using a generalized Dirichlet mixture as a prior, we can show that the marginal distribution of $\vec{X}_i$ is given by [2]

$$p(\vec{X}_i|\Theta) = \sum_{k=1}^{K} \omega_k \prod_{v=1}^{V-1} \frac{\Gamma(\alpha_{kv} + \beta_{kv})}{\Gamma(\alpha_{kv})\Gamma(\beta_{kv})} \prod_{v=1}^{V-1} \frac{\Gamma(\alpha'_{kv})\Gamma(\beta'_{kv})}{\Gamma(\alpha'_{kv} + \beta'_{kv})} \tag{2.7}$$

which we call the multinomial generalized Dirichlet mixture (MGDM), where $\alpha'_{kv} = \alpha_{kv} + f_{iv}$ and $\beta'_{kv} = \beta_{kv} + f_{iv+1} + \ldots + f_{iV}$ for $v = 1, \ldots, V-1$. Besides, it is straightforward to prove that [2]:

$$\hat{\pi}_v = \sum_{k=1}^{K} p(k|\vec{X}_i; \Theta) \frac{\alpha'_{kv}}{\alpha'_{kv} + \beta'_{kv}} \prod_{l=1}^{v-1} \frac{\beta'_{kl}}{\alpha'_{kl} + \beta'_{kl}} \tag{2.8}$$

where $p(k|\vec{X}_i; \Theta) = \frac{\omega_k p(\vec{X}_i|\xi_k)}{p(\vec{X}_i|\Theta)}$ and represents the posterior probabilities (i.e. the probability that a given $\vec{X}_i$ will be assigned to cluster $k$). Note that, when $K = 1$, Eq. 2.8 is reduced to Eq. 2.6 which is itself reduced to Eq. 2.4 when $\beta_v = \alpha_{v+1} + \beta_{v+1}$ (i.e. when the generalized Dirichlet is reduced to the Dirichlet).

## 2.3 Model Learning and Estimation of the Parameters

### 2.3.1 Leave-one-out (LOO) Likelihood Estimation

Let $\mathcal{X} = \{\vec{X}_i\}_{i=1}^N$ be a set of independent vectors represented by the mixture model in Eq. 2.7. An important problem is the estimation of the set of parameters $\Theta$ defining our model. The usual candidate for parameters estimation in the case of finite mixture models is the EM algorithm [110] where the E-step is devoted to compute the expected values of the class assignments (i.e. posterior probabilities $p(k|\vec{X}_i;\Theta)$) and the M-step updates the parameters estimates to refine the learned model by maximizing the following function $\sum_{i=1}^N \sum_{k=1}^K p(k|\vec{X}_i;\Theta)\log(w_k p(\vec{X}_i|\xi_k))$ which is actually the conditional expectation of the *complete-data log-likelihood*. By maximizing this function, it is easy to find the following estimate for the $w_k$ parameters:

$$w_k = \frac{1}{N}\sum_{i=1}^{N} p(k|\vec{X}_i;\Theta) \tag{2.9}$$

The maximization with respect to the $\xi_k$ parameters, however, involves the Gamma special function, $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}dt$, and by computing its derivatives other special functions such as the digamma (or the psi function) $\Psi(\alpha) = \frac{\partial \log(\Gamma(\alpha)}{\partial\alpha}$ and trigamma $\Psi'(\alpha) = \frac{\partial\Psi(\alpha)}{\partial\alpha}$ occur which makes the parameters estimation intractable [2]. In this thesis, we use another approach based on the maximization of the LOO likelihood[4] which has been shown to be an efficient approach when dealing with the estimation of small probabilities in the case of sparse data [112]. Given the set of independent vectors $\mathcal{X}$, the LOO likelihood corresponding to an $M$-component MGDM is obtained by replacing the estimates given by Eq. 2.8 in Eq. 2.1 for all the $\vec{X}_i$:

$$f_{LOO}(\mathcal{X}|\Theta) = \prod_{i=1}^{N}\prod_{v=1}^{V}\hat{\pi}_v^{f_{iv}} = \prod_{i=1}^{N}\prod_{v=1}^{V}\left(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha_{kv}^{'}}{\alpha_{kv}^{'}+\beta_{kv}^{'}}\prod_{l=1}^{v-1}\frac{\beta_{kl}^{'}}{\alpha_{kl}^{'}+\beta_{kl}^{'}}\right)^{f_{iv}} \tag{2.10}$$

That is the product of the probability of each sample, given the remaining data and parameters [113, 114]. Note that our approach can also be viewed as an empirical Bayes' technique[5], since we

[4] The leave-one-out estimator was proposed and applied originally by Mosteller and Wallace [111].

[5] This terminology was introduced by Robbins in [115] (See [104] for more details about empirical Bayes' approaches).

are using the data to help estimate the parameters by maximizing implicitly over the generalized Dirichlet prior mixture parameters $\Theta$ as opposed to the parameters of the multinomials $\vec{\pi}$. The LOO log-likelihood is given by

$$L_{LOO}(\mathcal{X}|\Theta) = \sum_{i=1}^{N}\sum_{v=1}^{V} f_{iv} \log\left(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}}\right) \tag{2.11}$$

In order to estimate the $\{\xi_k\}$, we use a second-order method which is Newton-Raphson approach based on the first, second and mixed derivatives of the LOO log-likelihood. We will therefore compute these derivatives. By computing the first derivatives of the LOO log-likelihood (see Appendix A), we obtain

$$\frac{\partial L_{LOO}(\mathcal{X}|\Theta)}{\partial \alpha_{kv}} = \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{\alpha'_{kv}} - \frac{1}{\alpha'_{kv}+\beta'_{kv}}\right] \tag{2.12}$$

$$\frac{\partial L_{LOO}(\mathcal{X}|\Theta)}{\partial \beta_{kv}} = \sum_{i=1}^{N} f_{iv} L_{ikv}\left[-\frac{1}{\alpha'_{kv}+\beta'_{kv}}\right] \tag{2.13}$$

where $L_{ikv} = \frac{p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}}}$ and can be interpreted as the posterior probability that a given feature $v$ will occur in a given vector $\vec{X}_i$ assigned to cluster $k$.

By computing the second and mixed derivatives of the LOO log-likelihood, we obtain (see Appendix B)

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial \alpha_{kv_1}\partial \alpha_{kv_2}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2} - \frac{\beta'_{kv}L_{ikv}+\alpha'_{kv}}{(\alpha'_{kv})^2(\alpha'_{kv}+\beta'_{kv})}\right] \\ \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} \end{cases} \quad .... (a)$$

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial \beta_{kv_1}\partial \beta_{kv_2}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(1-L_{ikv})}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})}\right] \\ \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} \end{cases} \quad .... (b)$$

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\alpha_{kv_1}\partial\beta_{kv_2}} = \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\beta_{kv_2}\partial\alpha_{kv_1}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2}\right] \\ \quad \text{if } v_1 = v_2 = v \\ 0 \quad \text{otherwise} \qquad .... (c) \end{cases} \tag{2.14}$$

Then, the Hessian matrix (i.e. the matrix of the second derivatives of the LOO log-likelihood) has a block-diagonal structure:

$$H(\xi_k) = \text{block-diag}\big\{H_1(\alpha_{k1}, \beta_{k1}), \ldots, H_V(\alpha_{kV}, \beta_{kV})\big\} \tag{2.15}$$

where

$$H_v(\alpha_{kv}, \beta_{kv}) = \begin{pmatrix} \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial^2\alpha_{kv}} & \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\alpha_{kv}\partial\beta_{kv}} \\ \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\beta_{kv}\partial\alpha_{kv}} & \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial^2\beta_{kv}} \end{pmatrix} \tag{2.16}$$

and we have [116, Theorem 8.8.16]

$$H(\xi_k)^{-1} = \text{block-diag}\big\{H_1(\alpha_{k1}, \beta_{k1})^{-1}, \ldots, H_V(\alpha_{kV}, \beta_{kV})^{-1}\big\} \tag{2.17}$$

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

$$H_v(\alpha_{kv}, \beta_{kv}) = D + \gamma\vec{a}\vec{a}^{tr} \tag{2.18}$$

where $D = \text{diag}[D_1, D_2] = \text{diag}\left[-\sum_{i=1}^{N} f_{iv} L_{ikv} \frac{\beta'_{kv} L_{ikv} + \alpha'_{kv}}{(\alpha'_{kv})^2(\alpha'_{kv}+\beta'_{kv})}, \sum_{i=1}^{N} f_{iv} L_{ikv} \frac{(1-L_{ikv})}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})}\right]$,
$\gamma = \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2}\right]$, $\vec{a}^{tr} = 1$, and $\gamma \neq (\sum_{k=1}^{2} \frac{a_k^2}{D_{kk}})^{-1}$.
Then, the inverse of the matrix $H_v(\alpha_{kv}, \beta_{kv})$ is another block diagonal matrix, composed of the inverse of each block [116, Theorem 8.3.3]:

$$H_v(\alpha_{kv}, \beta_{kv})^{-1} = D^* + \delta^* a^* a^{*tr} \tag{2.19}$$

where $D^* = D^{-1} = \text{diag}[1/D_1, 1/D_2]$, $a^{*tr} = (a_1/D_1, a_2/D_2)$, and $\delta^* = -\gamma(1 + \gamma(1/D_1 + 1/D_2))^{-1}$. 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:

$$\xi_k^{(t+1)} = \xi_k^{(t)} - H(\xi_k^{(t)})^{-1} \frac{\partial L_{LOO}(\mathcal{X}|\Theta)}{\partial\xi_k^{(t)}} \tag{2.20}$$

### 2.3.2 Complete Learning Algorithm

One of the major problems arising from mixture models application is how the optimum number of components is determined, and many methods have been proposed [9]. In this work, we have used a penalized likelihood approach based on the mixture minimum description length (MMDL) proposed initially in [117] and used successfully for the problem of images classification in [27], that is given by

$$MMDL(K) = -\log(p(\mathcal{X}|\Theta)) + \frac{N_K}{2}\log(N) + \frac{N_1}{2}\sum_{k=1}^{K}\log w_k \tag{2.21}$$

where $N_K = K(2V-1)$ is the number of parameters in our mixture model, $N_1 = 2(V-1)+1 = 2V-1$ is the number of parameters defining each component, and $(p(\mathcal{X}|\Theta)) = \prod_{i=1}^{N} p(\vec{X}_i|\Theta)$ is the likelihood function. As we can see from Eq. 2.21, the MMDL is actually a minimum description length (MDL) type criterion. Indeed, the MDL is given by

$$MDL(K) = -\log(p(\mathcal{X}|\Theta)) + \frac{N_M}{2}\log(N) \tag{2.22}$$

By comparing Eq. 2.21 and Eq. 2.22, we can see that $MMDL(K) = MDL(K) + \frac{N_1}{2}\sum_{k=1}^{K}\log w_k$. The extra negative term $\frac{N_1}{2}\sum_{k=1}^{K}\log w_k$ is introduced to overcome the fact that the MDL criterion considers that all the vectors to cluster have equal importance for each component. That is not true in the case of mixture models where each vector has its own importance (i.e. weight) in estimating the parameters. This fact can be shown through the Fisher information matrix of a mixture model (See [117] for more details). Having the MMDL criterion in hand, the complete algorithm for estimation and selection is as the following:

**Algorithm**

For each candidate value of $K$:

1. Initialize the parameters $\Theta^{(0)}$ using the initialization algorithm proposed in [2].

2. Iterate the two following steps until convergence:

(a) E-Step: Compute $p(k|\vec{X}_i;\Theta) = \frac{\omega_k p(\vec{X}_i|\xi_k)}{p(\vec{X}_i|\Theta)}$.

(b) M-Step:

i. Update the $w_k^{(t)}$ using Eq. 2.9.

ii. Update the $\xi_k^{(t)}$ using Eq. 2.20.

3. Calculate the associated criterion $MMDL(K)$ using Eq. 2.21.

4. Select the optimal model $M^*$ such that:

$$K^* = \arg\min_K MMDL(K)$$

## 2.4 A Generative/Discriminative Model

Different approaches have been proposed to manage, filter and retrieve visual information. Two main categories of approaches are: model-based approaches and discriminative classifiers. Model-based approaches are based on generative probabilistic models and discriminative classifiers allow the construction of flexible decision boundaries. Both models have achieved great successes in a variety of applications in terms of the improvement of data classification accuracies, and the modeling of complex data and concepts, respectively. SVM is a well known example of discriminative classifiers [118]. An important problem when considering SVM is the choice of the kernel. Choosing an appropriate kernel function for a given type of data in a particular application is a challenging and difficult problem and remains largely unresolved. One of the most successful approaches is the Fisher kernel proposed in [119] and which can be obtained from the generative model describing the data. In the following, we will investigate the derivation of a Fisher kernel from our statistical model (Eq. 2.7) and its application to SVMs. The Fisher kernel was proposed initially in [119] and is computed at the estimated $\Theta$ on the resulting statistical manifold as follows:

$$K(\vec{X},\vec{X}_i) = U_{\vec{X}}^{tr}(\Theta)I(\Theta)^{-1}U_{\vec{X}_i}(\Theta) \tag{2.23}$$

where $U_{\vec{X}}(\Theta)$ denotes the Fisher score (i.e. the gradient of log probability with respect to Θ), and $I(\Theta)$ is the Fisher information matrix given by

$$I(\Theta) = E_{\vec{X}}[U_{\vec{X}}(\Theta)U^{tr}_{\vec{X}}(\Theta)] \tag{2.24}$$

and the expectation is over $p(\vec{X}|\Theta)$. The role of the Fisher information matrix, however, is less significant as shown in [119] and then can be approximated by the identity matrix.

In the following, we shall derive the Fisher kernel for our generative $K$-component mixture model. By computing the gradient of log probability with respect to our model parameters: $w_k$, $\alpha_{kv}$ and $\beta_{kv}$, $k = 1, \ldots, K$, $v = 1, \ldots, V-1$, we obtain

$$\frac{\partial \log p(\vec{X}_i|\Theta)}{\partial \alpha_{kv}} = p(k|\vec{X}_i;\Theta)\Bigg(\Psi(\alpha_{kv}+\beta_{kv}) - \Psi(\alpha_{kv}) + \Psi(\alpha^{'}_{kv}) - \Psi(\alpha^{'}_{kv}+\beta^{'}_{kv}))\Bigg)$$

$$\frac{\partial \log p(\vec{X}_i|\Theta)}{\partial \beta_{kv}} = p(k|\vec{X}_i;\Theta)\Bigg(\Psi(\alpha_{kv}+\beta_{kv}) - \Psi(\beta_{kv}) + \Psi(\beta^{'}_{kv}) - \Psi(\alpha^{'}_{kv}+\beta^{'}_{kv}))\Bigg)$$

$$\frac{\partial \log p(\vec{X}_i|\Theta)}{\partial w_k} = \frac{p(k|\vec{X}_i;\Theta)}{w_k}$$

It is noteworthy that this Fisher kernel takes into account the posterior probabilities and then uses the totality of the data set as a background information.

CHAPTER 3

# Experiments

## 3.1 Experimental Results

In this section, we conduct some comprehensive experiments in order to investigate the effectiveness of the proposed approach. Our experiments involve the important problem of image databases categorization using low-level images contents. Texture and color are widely accepted as being two key low-level features in image representation. On the other hand, *bag-of-visterms (BOV)* (or bag-of-visual words) [95] that is based on local keypoint features has attracted a lot of research attention recently. Thus, our experiments take into account both of these approaches. Indeed, results will be first presented for an application involving scene modeling and classification using visual words. Second, we propose a novel model for color texture images modeling and categorization.

### 3.1.1 Scene Modeling and Classification using Visual Words

Our first application involves an important and difficult problem in computer vision which is visual scene modeling and classification[1] using the text-like BOV representation, which is actually the BOK [74] with quantized local descriptors, as an analogy to dictionaries in the case of text

[1] Many psychophysical and psychological studies have shown that humans may identify scenes independently of objects identification [120–122].

documents[2], and recently extensively studied in [95]. Note that the authors in [74] have used multinomial mixture and support vector machine with some classic kernels for classification. Visual scene modeling and classification may be used for different other applications such as image databases browsing, objects recognition and content-based retrieval or recommendation. In contrast to previous approaches based on global visual features, the BOV approach is based on features computed over local areas in the image (i.e. local descriptors) which have been shown to be efficient in many complex applications by providing stable representation and robustness to image clutter, occlusion and partial visibility [95]. After detecting local keypoints using one of the existing detectors (see in subSection 1.1.1.2 to recall), next important step in this approach is the extraction of local descriptors that should be invariant to images transformations, occlusions and lighting variations [74]. Keypoints are then grouped into a number of homogenous clusters $V$, using a clustering or vector quantization algorithm such as $K$-means, according to the similarity of their descriptors. Each cluster center is then treated as a visual word, and we obtain a vocabulary of $V$ visual words describing all possible local image patterns. Having this vocabulary in hand, each image can be represented as a $V$-dimensional vector each component of which contains the frequency of each visual word in that image. The resulting feature vector can be used then for the categorization task.

#### 3.1.1.1 Classification of Vacation Images

In the first experiment and following [27], we consider the particular problem of binary hierarchical classification of vacation images by performing multiple two class classifications. At the highest of the hierarchy level images are classified as indoor or outdoor. Then, we further classify outdoor images as city or landscape [123]. Finally, landscape images are classified into forest and mountain classes. To evaluate our model, we use a database of 5000 vacation images (3000 outdoor and 2000 indoor) collected from different sources. Among the 3000 outdoor images, 1200 are city images

[2] See [95] for an interesting discussion about this analogy.

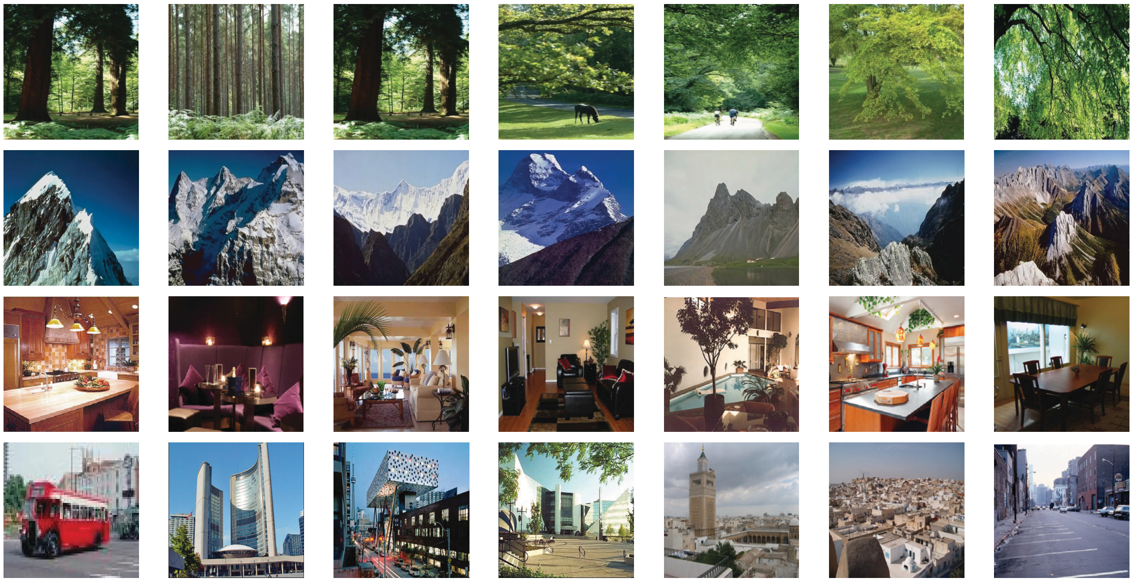


Figure 3.1: Sample images from each group. Row 1: Outdoor landscape images (forest), Row 2: Outdoor landscape images (mountain), Row 3: Indoor images, Row 4: City images.

and the rest represents the class landscape (1000 and 800 images are in the subclasses mountain and forest, respectively). Figure 3.1 shows some images from our database. From this database, 2000 images were taken, randomly, to construct the visual vocabulary. The interest points were detected using the DOG point detector since it has shown excellent performance [1, 95]. Then, we have used SIFT descriptors, based on the grayscale representation of images, which performs better than the majority of the existing descriptors [1, 53], computed on detected keypoints of all images and giving 128-dimensional vector for each keypoint. Moreover, extracted SIFT vectors were clustered using the $K$-means algorithm providing 300 visual-words. Each image in the database was then represented by a 300-dimensional vector of frequencies.

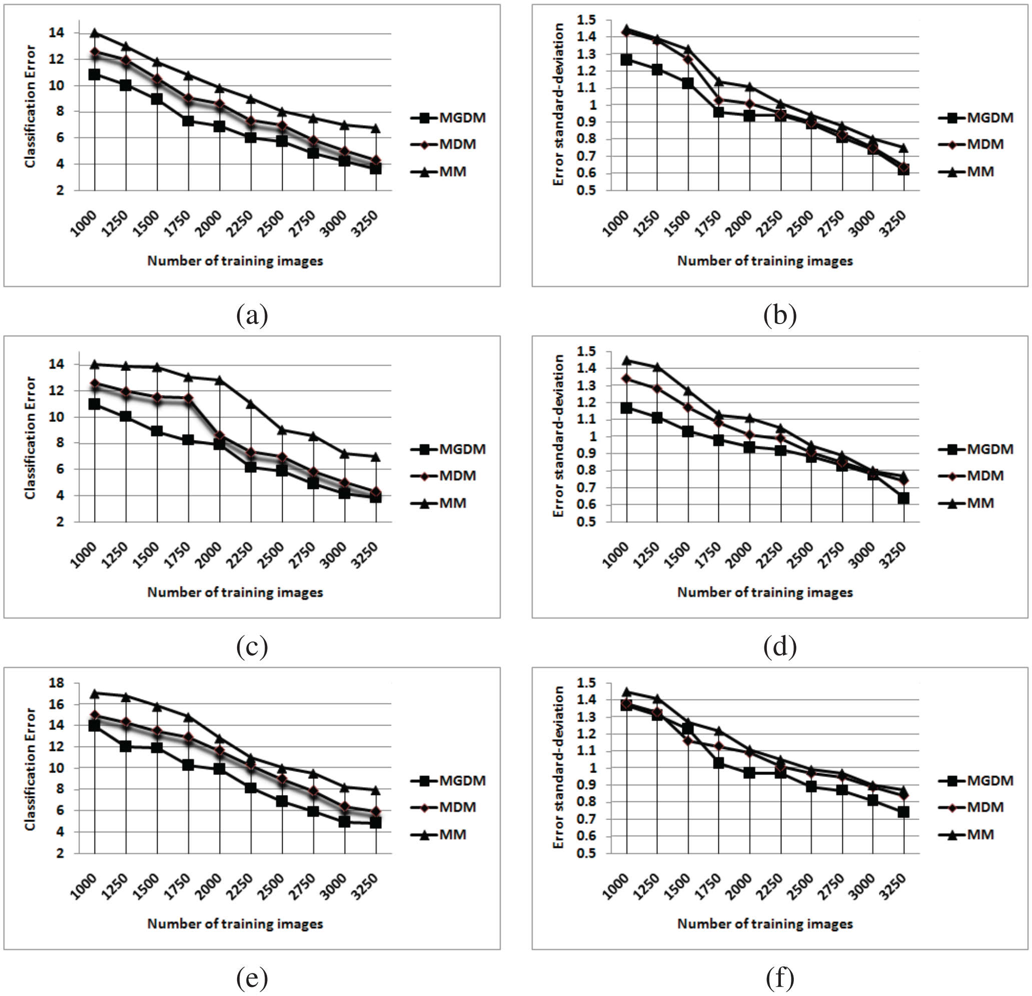
Table 3.1: Average rounded confusion matrices for the different classification problems using: (a-c) MGDM. (d-f) MDM. (g-i) MM.

| | Indoor | Outdoor | | City | Landscape | | Mountain | Forest |
|---|---|---|---|---|---|---|---|---|
| | (a) | | | (b) | | | (c) | |
| Indoor | **1889** | 111 | City | **1086** | 114 | Mountain | **892** | 108 |
| Outdoor | 254 | **2746** | Landscape | 159 | **1641** | Forest | 86 | **714** |
| | (d) | | | (e) | | | (f) | |
| Indoor | **1856** | 144 | City | **1043** | 157 | Mountain | **853** | 147 |
| Outdoor | 268 | **2732** | Landscape | 187 | **1613** | Forest | 88 | **712** |
| | (g) | | | (h) | | | (i) | |
| Indoor | **1798** | 202 | City | **1015** | 185 | Mountain | **832** | 168 |
| Outdoor | 313 | **2687** | Landscape | 203 | **1597** | Forest | 99 | **701** |

Table 3.2: Average per class errors (± standard deviation) for the different approaches.

| | Indoor vs. Outdoor | City vs. Landscape | Mountain vs. Forest |
|---|---|---|---|
| MGDM | **7.30**%±0.96 | **8.24**%±0.98 | **10.30**%±1.03 |
| MDM | 9.10%±1.03 | 11.47%±1.08 | 12.94%±1.13 |
| MM | 10.78% ± 1.14 | 13.05%±1.13 | 14.83%±1.22 |

For the indoor vs. outdoor classification, 1750 images were used for training (1000 outdoor and 750 indoor). For the city vs. landscape classification problem we have used 1000 images for training (500 images for each class). In addition 500 images were used as a training set for the mountain (300 images) vs. forest (200 images) classification problem. Each training set was modeled by an MGDM using the algorithm presented in subSection 2.3.2. Tables 3.1.a-i represent the rounded confusion matrices (we ran our algorithm 20 times with different training sets) for the different classification problems using the MGDM, multinomial Dirichlet mixture (MDM), and multinomial mixture (MM). Table 3.2 summarizes the average per class errors. Note that we have used laplace smoothing for the MM to avoid zero frequencies. According to the results, it is clear that the best results are obtained using the MGDM (the difference is statistically significant according to a T-test).



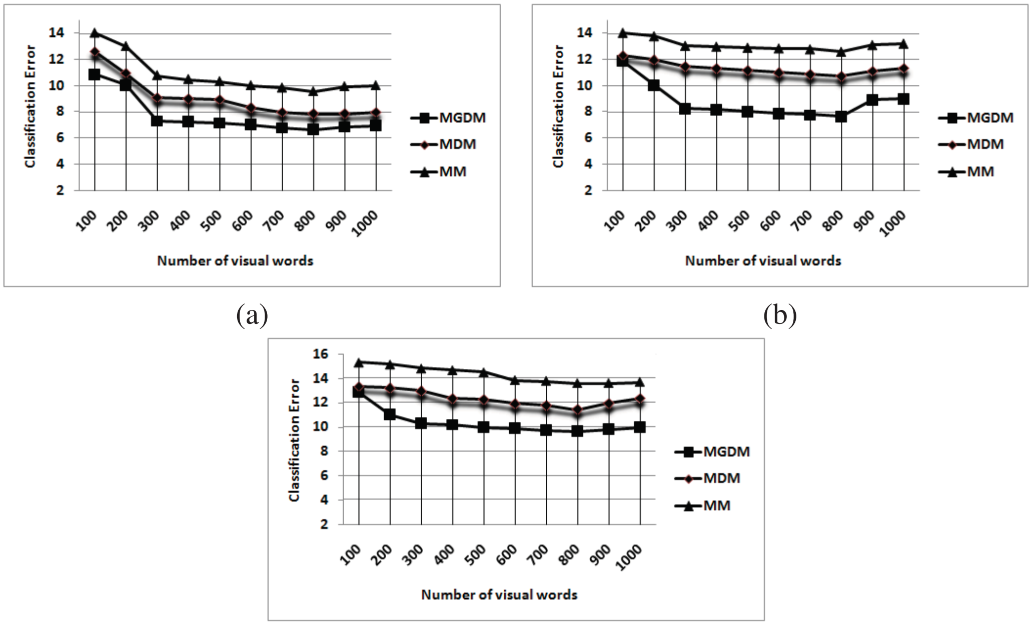
(a) (b)

(c) (d)

(e) (f)

Figure 3.2: Average classification error and standard deviation as a function of the number of images in the training set. (a)(b) Indoor vs. Outdoor, (c)(d) City vs. Landscape, (e)(f) Mountain vs. Forest.

Figure 3.2 shows the average classification errors and standard deviations as a function of the number of training images. These figures show that increasing the number of training images reduces both the average errors and standard deviations.



(a) (b)

(c)

Figure 3.3: Evolution of the classification error with the number of visual words. (a) Indoor vs. Outdoor, (b) City vs. Landscape, (c) Mountain vs. Forest.

We also conducted experiments to study the influence of the number of visual words on the classification performance. Figure 3.3.a-c show the evolution of the error with the number of visual words. According to these figures, we can see that the classification errors does not change much when the number of visual words is taken between 300 and 800.

In the second experiment, we consider the classification of the whole data set into 4 groups namely indoor, city, mountain, and forest. The first goal of this experiment is to compare the accuracy of mixture estimation and selection using the novel algorithm that we propose in this thesis and the approach that we previously introduced in [2]. The second goal is to compare the modeling capabilities of the MGDM against MDM and MM in a multiclass classification problem. We take the same number of training images and visual words used in the previous experiments.

Table 3.3: Loglikelihoods (average and standard deviation over 20 runs) of the training data in the different classes when using MGDM learned by both the approach in this thesis and the one in [2].

| | indoor | city | mountain | forest |
|---|---|---|---|---|
| New approach | -395.73 ± 0.97 | -400.09 ± 1.11 | -439.94 ± 1.21 | -417.28 ± 1.26 |
| Algorithm [2] | -397.73 ± 1.12 | -403.12 ± 1.56 | -443.51 ± 1.14 | -421.47 ± 1.41 |

Table 3.3 shows the loglikelihoods (measured in bits, i.e. base-two logarithm is used) of the training data in the different classes when using MGDM learned by both the approach in this thesis and the one in [2]. In the reported results, the values of the loglikelihoods are divided by the number of vectors in each training class. The table shows the clear dominance of the novel learning approach, looking at the increased likelihood, over the previously proposed one. Tables 3.4 and 3.5 show the confusion matrices using both approaches. The results show again that the performance is improved by the new learning and estimation algorithm.

Table 3.4: Confusion matrix for the 4 classes image categorization problem using MGDM learned by the proposed algorithm.

| | indoor | city | mountain | forest | class error |
|---|---|---|---|---|---|
| indoor | **1886** | 61 | 27 | 26 | 5.70% |
| city | 27 | **1033** | 67 | 73 | 13.91% |
| mountain | 9 | 28 | **852** | 111 | 14.80% |
| forest | 4 | 11 | 88 | **697** | 12.87% |

Table 3.5: Confusion matrix for the 4 classes image categorization problem using MGDM learned by the algorithm in [2].

| | indoor | city | mountain | forest | class error |
|---|---|---|---|---|---|
| indoor | **1879** | 65 | 29 | 27 | 6.05% |
| city | 29 | **1018** | 74 | 79 | 15.16% |
| mountain | 13 | 32 | **828** | 127 | 17.20% |
| forest | 6 | 15 | 103 | **676** | 15.50% |

Table 3.6: Confusion matrix for the 4 classes image categorization problem using MDM

| | indoor | city | mountain | forest | class error |
|---|---|---|---|---|---|
| indoor | **1866** | 71 | 35 | 28 | 6.70% |
| city | 32 | **1001** | 79 | 88 | 16.58% |
| mountain | 17 | 35 | **821** | 127 | 17.90% |
| forest | 12 | 19 | 117 | **652** | 18.50% |

Table 3.7: Confusion matrix for the 4 classes image categorization problem using MM

| | indoor | city | mountain | forest | class error |
|---|---|---|---|---|---|
| indoor | **1843** | 84 | 42 | 31 | 7.85% |
| city | 37 | **975** | 89 | 99 | 18.75% |
| mountain | 21 | 42 | **807** | 130 | 19.30% |
| forest | 17 | 24 | 127 | **632** | 21.00% |

Tables 3.6 and 3.7 represent the confusion matrices by applying MDM and MM. By analyzing these four tables, we can see that an important part of the misclassified images are made by mountain vs. forest which is caused by the fact that some mountain images contain forest, too. Moreover, we can conclude that MGDM reaches the best results in term of classification error reduction with an overall classification error of 10.64% (11.98% when using the algorithm in [2]) as compared to the 13.20% and 14.86% when we use the MDM and MM, respectively.

#### 3.1.1.2 Other Data Set

In the third experiment, we evaluate the performance of our model on a challenging database containing 13 categories of natural scenes [124]: highway (260 images), inside of cities (308 images), tall buildings (356 images), streets (292 images), suburb residence (241 images), forest (328 images), coast (360 images), mountain (374 images), open country (410 images), bedroom (174 images), kitchen (151 images), livingroom (289 images), and office (216 images). Figure 3.4 shows examples of these images which have an average size of approximately 250 × 300 pixels. Each extracted vector, representing frequencies of visual words, are separated into the unknown

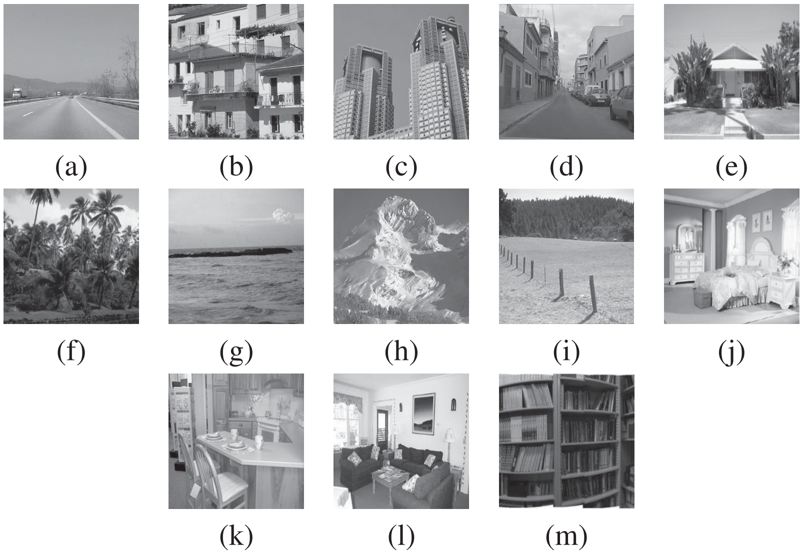


Figure 3.4: Sample images from each group. (a) Highway, (b) Inside of cities, (c) Tall buildings, (d) Streets, (e) Suburb residence, (f) Forest, (g) Coast, (h) Mountain, (i) Open country, (j) Bedroom, (k) Kitchen, (l) Livingroom, (m) Office.

or test set of vectors, whose class is unknown, and the training set of vectors (we take randomly 100 vectors for training from each class), whose class is known. Tables 3.8, 3.9 and 3.10 show the average confusion matrices reported by MGDM, MDM, and MM, respectively, by running the estimation algorithms 10 times with varying random selection of the training set. From these table, we can see that the average classification accuracies were 73.44% (653 misclassified images), 71.24% (707 misclassified images) and 67.22% (806 misclassified images), respectively.

### 3.1.2 Application to SVM

In this subsection, we investigate the performance of the hybrid model presented in Section 2.4 by applying it to previously introduced scenes classification problems described in subSection 3.1.1.1 and 3.1.1.2. Through this application, we compare the effectiveness of our MGDM kernel with other different kernels: a Fisher kernel based on MDM, a Fisher kernel based on MM, polynomial

Table 3.8: Average rounded confusion matrix for the 13 classes image categorization problem using MGDM.

| | (a) | (b) | (c) | (d) | (e) | (f) | (g) | (h) | (i) | (j) | (k) | (l) | (m) | accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Highway | **134** | 4 | 0 | 5 | 0 | 0 | 10 | 2 | 5 | 0 | 0 | 0 | 0 | 83.75% |
| Inside | 4 | **156** | 11 | 9 | 12 | 6 | 0 | 7 | 3 | 0 | 0 | 0 | 0 | 75.00% |
| Tall buildings | 0 | 25 | **183** | 14 | 0 | 11 | 0 | 23 | 0 | 0 | 0 | 0 | 0 | 71.48% |
| Streets | 4 | 5 | 9 | **149** | 11 | 10 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 77.60% |
| Suburb | 0 | 9 | 0 | 8 | **111** | 9 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 78.72% |
| Forest | 6 | 10 | 3 | 17 | 14 | **161** | 0 | 9 | 8 | 0 | 0 | 0 | 0 | 70.61% |
| Coast | 17 | 0 | 0 | 0 | 0 | 9 | **209** | 6 | 19 | 0 | 0 | 0 | 0 | 80.38% |
| Mountain | 0 | 4 | 14 | 5 | 3 | 16 | 2 | **206** | 24 | 0 | 0 | 0 | 0 | 75.18% |
| Open country | 7 | 9 | 3 | 16 | 11 | 17 | 5 | 39 | **203** | 0 | 0 | 0 | 0 | 65.48% |
| Bedroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **58** | 5 | 6 | 5 | 78.37% |
| Kitchen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | **39** | 5 | 4 | 76.47% |
| Livingroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 16 | 17 | **118** | 38 | 62.43% |
| Office | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 7 | 11 | 19 | **79** | 68.10% |

Table 3.9: Average rounded confusion matrix for the 13 classes image categorization problem using MDM.

| | (a) | (b) | (c) | (d) | (e) | (f) | (g) | (h) | (i) | (j) | (k) | (l) | (m) | accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Highway | **129** | 5 | 0 | 6 | 0 | 0 | 12 | 3 | 5 | 0 | 0 | 0 | 0 | 80.62% |
| Inside | 4 | **151** | 13 | 10 | 13 | 6 | 0 | 8 | 3 | 0 | 0 | 0 | 0 | 72.59% |
| Tall buildings | 0 | 26 | **176** | 16 | 0 | 13 | 0 | 25 | 0 | 0 | 0 | 0 | 0 | 68.75% |
| Streets | 4 | 5 | 9 | **146** | 14 | 10 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 76.04% |
| Suburb | 0 | 9 | 0 | 8 | **108** | 11 | 0 | 5 | 0 | 0 | 0 | 0 | 0 | 76.59% |
| Forest | 6 | 10 | 5 | 17 | 15 | **155** | 0 | 11 | 9 | 0 | 0 | 0 | 0 | 67.98% |
| Coast | 18 | 0 | 0 | 0 | 0 | 11 | **204** | 8 | 19 | 0 | 0 | 0 | 0 | 78.46% |
| Mountain | 0 | 4 | 14 | 7 | 3 | 16 | 2 | **204** | 24 | 0 | 0 | 0 | 0 | 74.45% |
| Open country | 8 | 9 | 5 | 16 | 11 | 17 | 6 | 39 | **199** | 0 | 0 | 0 | 0 | 64.19% |
| Bedroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **55** | 6 | 8 | 5 | 74.32% |
| Kitchen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | **38** | 6 | 4 | 74.50% |
| Livingroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 19 | 19 | **114** | 37 | 60.31% |
| Office | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10 | 14 | 19 | **73** | 62.93% |

kernel of degree 2, Gaussian kernel, and a generalized form of radial basis functions (RBF) kernels $K_{d-RBF}(\vec{X}_i, \vec{X}_j) = e^{-d(\vec{X}_i, \vec{X}_j)}$ [125], where $d(\vec{X}_i, \vec{X}_j)$ is a given distance. As we are dealing with discrete data, we have taken $\chi^2$ function as a distance $d_{\chi^2}(\vec{X}_i, \vec{X}_j) = \sum_{v=1}^{V} \frac{(f_{iv}-f_{jv})^2}{f_{iv}+f_{jv}}$ [125], also one-against-all approach is adopted to extend SVM to multi-class problems.

Table 3.10: Average rounded confusion matrix for the 13 classes image categorization problem using MM.

| | (a) | (b) | (c) | (d) | (e) | (f) | (g) | (h) | (i) | (j) | (k) | (l) | (m) | accuracy |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Highway | **121** | 4 | 0 | 4 | 0 | 0 | 17 | 6 | 8 | 0 | 0 | 0 | 0 | 75.62% |
| Inside | 5 | **141** | 9 | 9 | 21 | 9 | 0 | 8 | 6 | 0 | 0 | 0 | 0 | 67.78% |
| Tall buildings | 2 | 24 | **169** | 16 | 1 | 12 | 3 | 29 | 0 | 0 | 0 | 0 | 0 | 66.01% |
| Streets | 6 | 6 | 9 | **136** | 11 | 17 | 2 | 5 | 0 | 0 | 0 | 0 | 0 | 70.83% |
| Suburb | 1 | 15 | 2 | 9 | **97** | 10 | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 68.79% |
| Forest | 7 | 12 | 7 | 23 | 14 | **145** | 1 | 9 | 10 | 0 | 0 | 0 | 0 | 63.59% |
| Coast | 21 | 1 | 3 | 0 | 1 | 9 | **189** | 8 | 28 | 0 | 0 | 0 | 0 | 72.69% |
| Mountain | 3 | 3 | 15 | 8 | 5 | 16 | 4 | **191** | 29 | 0 | 0 | 0 | 0 | 69.70% |
| Open country | 9 | 14 | 9 | 19 | 12 | 17 | 5 | 35 | **190** | 0 | 0 | 0 | 0 | 61.29% |
| Bedroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **54** | 6 | 8 | 6 | 72.97% |
| Kitchen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | **33** | 6 | 7 | 64.70% |
| Livingroom | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 18 | 21 | **116** | 34 | 61.37% |
| Office | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 13 | 24 | **71** | 61.20% |

Table 3.11 represents the confusion matrices for the different binary classification problems described in the previous section using different kernels. The results for these binary classification problems and the 4 classes categorization one are summarized in table 3.12, which shows the average error rates using the different tested kernels. For the 13 classes categorization problem the average classification accuracies were 74.12%, 72.04%, 68.03%, 68.79%, 67.88% and 68.03% using MGDM, MDM, MM, $RBF_{\chi^2}$, polynomial and $RBF_{Gaussian}$ kernels, respectively. According to the results, it is clear that the MGDM-based kernel gives the best performances since it takes into account the discrete nature of the features. Moreover, it is noteworthy that introducing the different generative models as kernels for SVM gives better results than using them directly for classification. This is actually an expected result since it was shown in [119] that kernel classifiers employing the Fisher kernel would be at least as powerful as the generative model used to develop the kernel.

Table 3.11: Confusion matrices for the different binary classification problems using the different kernels. (a-c) MGDM. (d-f) MDM. (g-i) MM. (j-l) $RBF_{\chi^2}$. (m-o) Polynomial. (p-r) $RBF_{Gaussian}$ .

| | Indoor | Outdoor | | City | Landscape | | Mountain | Forest |
|---|---|---|---|---|---|---|---|---|
| | (a) | | | (b) | | | (c) | |
| Indoor | **1894** | 106 | City | **1106** | 94 | Mountain | **902** | 98 |
| Outdoor | 224 | **2776** | Landscape | 125 | **1675** | Forest | 73 | **727** |
| | (d) | | | (e) | | | (f) | |
| Indoor | **1837** | 163 | City | **1064** | 136 | Mountain | **875** | 125 |
| Outdoor | 262 | **2738** | Landscape | 160 | **1640** | Forest | 73 | **727** |
| | (g) | | | (h) | | | (i) | |
| Indoor | **1810** | 190 | City | **1025** | 175 | Mountain | **879** | 121 |
| Outdoor | 288 | **2712** | Landscape | 173 | **1627** | Forest | 102 | **698** |
| | (j) | | | (k) | | | (l) | |
| Indoor | **1815** | 185 | City | **1000** | 200 | Mountain | **860** | 140 |
| Outdoor | 304 | **2696** | Landscape | 162 | **1638** | Forest | 93 | **707** |
| | (m) | | | (n) | | | (o) | |
| Indoor | **1793** | 207 | City | **1017** | 183 | Mountain | **834** | 166 |
| Outdoor | 300 | **2700** | Landscape | 206 | **1594** | Forest | 97 | **703** |
| | (p) | | | (q) | | | (r) | |
| Indoor | **1791** | 209 | City | **1023** | 177 | Mountain | **840** | 160 |
| Outdoor | 374 | **2626** | Landscape | 215 | **1585** | Forest | 108 | **692** |

Table 3.12: Average error rates using the different tested kernels.

| | Outdoor vs. Indoor | City vs. Landscape | Mountain vs. Forest | 4 classes |
|---|---|---|---|---|
| SVM+MGDM | **6.60%** | **7.30%** | **9.50%** | **9.78%** |
| SVM+MDM | 8.50% | 9.86% | 11.01% | 12.09% |
| SVM+MM | 9.56% | 11.60% | 12.38% | 13.93% |
| SVM+$RBF_{\chi^2}$ | 9.78% | 12.06% | 12.94% | 13.21% |
| SVM+Polynomial | 10.14% | 12.96% | 14.61% | 14.03% |
| SVM+$RBF_{Gaussian}$ | 11.66% | 13.06% | 14.88% | 14.22% |

### 3.1.3 Color Texture Modeling and Classification

An important topic in the fields of image processing and computer vision is texture analysis. Many approaches have been proposed for this problem and can be classified into three classes: structural, statistical and signal theoretic methods [29]. The majority of these approaches, however, have been devoted to gray level images and their transfer to the color domain is still a challenging problem. According to [30], the techniques combining color and texture can be grouped into parallel, sequential and integrative. While parallel approaches separate the processing of color and texture, sequential approaches use color analysis as a preprocessing step to analyze texture. The most successful techniques are called integrative since they take into account the dependency between color and texture features [30, 126]. A well-known technique to analyze color texture is the use of co-occurrence matrices [69]. Co-occurrence matrices are mostly used as intermediate features and dimensionality reduction is performed in computing features of the types described in [32, 127] such as energy, entropy, contrast, homogeneity and correlation. Dimensionality reduction, however, causes the lost of important information contained in the distributions of co-occurrence matrices contents. In this subsection, we propose a statistical model, based on co-occurrence matrices and MGDM without dimensionality reduction, that integrates both color and texture to describe color texture images. The objectives we set in this subsection are three-fold: (1) propose a new approach for color texture modeling based on our discrete mixture; (2) determine the contribution of color information to the classification performance; and (3) investigate its performance when combined with visual words.

#### 3.1.3.1 The Model

In what follows, we will use $\{I(x, y), 0 \leq x \leq K-1, 0 \leq y \leq L-1\}$ to denote a $K \times L$ image with gray levels $\{c_1, \dots, c_G\}$, also resulting $G \times G$ co-occurrence matrix for a displacement vector $\vec{d} = (d_1, d_2)$ is denoted as $C_{\vec{d}}$. Therefore, an entry $C_{\vec{d}}(c_i, c_j)$ of the co-occurrence matrix $C_{\vec{d}}$ is the number of occurrences of the pair of pixels with gray levels $c_i$ and $c_j$ which are a distance $\vec{d}$ apart. Formally, it is given as:

$$C_{\vec{d}}(c_i, c_j) = f_{c_i,c_j} = Card\{(r, s) : I(r, s) = c_i, I(r + d_1, s + d_2) = c_j\} \tag{3.1}$$

where $Card\{\}$ refers to the number of elements of a set (see Figure 1.1). Generally, to get a good image texture representation, we need to assign co-occurrence matrices for each of the considered displacements $\{\vec{d_i}; i = 1, 2, ...T\}$. So, a color image $\mathcal{I} = (I_1, ...I_Z)$ defined in a $Z$-dimensional cartesian space[3], for a given displacement vector $\vec{d}$, retains $Z$ co-occurrence matrices. In this thesis, we propose to model the color texture information using discrete finite mixture models by observing that for each pair $(c_i, c_j)$ in each color channel $Z$, we can associate a $T$-dimensional vector of counts, by considering $T$ displacements $\{\vec{d_i}\}$, described as follows:

$$\vec{f}^{z}_{c_i,c_j} = (f^{z,d_1}_{c_i,c_j}, \ldots, f^{z,d_T}_{c_i,c_j}) \tag{3.2}$$

Then, the color texture information is represented by $Z \times G^2$ $T$-dimensional vectors of frequencies which can be modeled by our MGDM.

#### 3.1.3.2 Results

In the first experiment and in order to validate the proposed model, we used the *Vistex* color texture database obtained from the MIT Media Lab[4]. In our experimental framework, each of the 512 × 512 images from the *Vistex* database was divided into 64 × 64 images. Since each 512 × 512 *mother image* contributes 64 images to our database, ideally all 64 images should be classified in the same class. In the experiment, six homogeneous texture groups, "bark", "fabric", "food", "metal", "water" and "sand", were used to create a new database. A database with 1920 images of size 64 × 64 pixels was obtained. Four images from the bark, fabric and metal texture groups

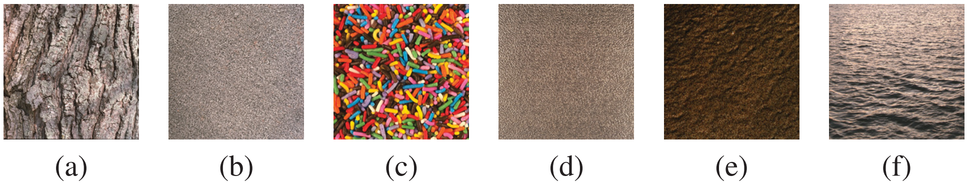


Figure 3.5: Sample images from each group. (a) Bark, (b) Fabric, (c) Food, (d) Metal, (e) Sand, (f) Water.

[3] For instance, an image defined in RGB spaces will have three co-occurrence metrics (one for each color channel). Besides RGB, we have also tested the algorithm using other color spaces (HSI, YIQ, CIE-XYZ and CIE-LAB) but we did not remark much changes in the results which is in agreement with the conclusions outlined in [126], for instance.

[4] http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html

were used to obtain 256 images for each of these categories, and 6 images from water, food and sand were used to obtain 384 images for those categories. Examples of images from each of the categories are shown in Fig. 3.5.

After randomly selecting 600 images, 100 from each category, as training data and computing the co-occurrence matrices (we consider 8 displacements) for each color channel and for each image, a MGDM is trained on each category of color texture. Finally, in the classification stage each image is assigned to the class increasing more its loglikelihood. The confusion matrices for the color texture images classification using MGDM, MDM and MM are given in tables 3.13, 3.14 and 3.15, respectively. These matrices show that the average numbers of misclassified images were 28 (i.e. average error rate of 1.45%), 44 (i.e. average error rate of 2.29%) and 55 (i.e. average error rate of 2.86%) using MGDM, MDM and MM, respectively. The average error rates, by running the estimation algorithms 10 times with varying random selection of the training images, using different classification approaches are summarized in table 3.16. According to these results, we can say that the incorporation of the color information enhances the performance of texture classification. It is noteworthy that this conclusion was previously suggested by psychological studies that have shown that color of textural elements helps in the discrimination of texture by the human visual system [128]. Results show also that MDM performs better than MM, and the MGDM performs even better.

Table 3.13: Average rounded confusion matrix for color texture image classification using MGDM.

| | Bark | Fabric | Food | Metal | Sand | Water |
|---|---|---|---|---|---|---|
| Bark | **252** | 0 | 0 | 0 | 4 | 0 |
| Fabric | 0 | **250** | 6 | 0 | 0 | 0 |
| Food | 0 | 6 | **378** | 0 | 0 | 0 |
| Metal | 0 | 0 | 0 | **256** | 0 | 0 |
| Sand | 2 | 0 | 0 | 0 | **382** | 0 |
| Water | 3 | 0 | 0 | 5 | 2 | **374** |

Table 3.14: Average rounded confusion matrix for color texture image classification using MDM.

| | Bark | Fabric | Food | Metal | Sand | Water |
|---|---|---|---|---|---|---|
| Bark | **248** | 0 | 0 | 3 | 3 | 2 |
| Fabric | 0 | **246** | 5 | 0 | 3 | 2 |
| Food | 0 | 4 | **376** | 4 | 0 | 0 |
| Metal | 0 | 0 | 0 | **252** | 1 | 3 |
| Sand | 4 | 0 | 0 | 0 | **377** | 3 |
| Water | 2 | 0 | 0 | 2 | 3 | **377** |

Table 3.15: Average rounded confusion matrix for color texture image classification using MM.

| | Bark | Fabric | Food | Metal | Sand | Water |
|---|---|---|---|---|---|---|
| Bark | **247** | 0 | 0 | 3 | 4 | 2 |
| Fabric | 0 | **246** | 5 | 0 | 3 | 2 |
| Food | 0 | 9 | **371** | 4 | 0 | 0 |
| Metal | 0 | 0 | 0 | **252** | 1 | 3 |
| Sand | 4 | 0 | 0 | 0 | **377** | 3 |
| Water | 2 | 0 | 0 | 5 | 5 | **372** |

Table 3.16: Average error rates (± standard deviation) for the color texture images classification problem using different approaches.

| | with color | without color |
|---|---|---|
| MGDM | 1.45% (±0.13) | 2.18% (±0.19) |
| MDM | 2.29% (±0.41) | 2.60 % (±0.43) |
| MM | 2.86% (±0.38) | 3.08% (±0.40) |
| SVM+MGDM | **1.04% (±0.17)** | **2.03% (±0.18)** |
| SVM+MDM | 1.56% (±0.35) | 2.34% (±0.39) |
| SVM+MM | 1.92% (±0.39) | 2.85% (±0.37) |
| SVM+$RBF_{\chi^2}$ | 1.87% (±0.24) | 2.68% (±0.27) |
| SVM+Polynomial | 2.93% (±0.22) | 3.41%(±0.26) |
| SVM+$RBF_{Gaussian}$ | 2.98% (±0.25) | 3.52% (±0.31) |

In the second experiment, we investigate the performance when color texture features are combined with visual words. Table 3.17 shows the classification results using different approaches. By comparing the results in this table and the performances presented in the previous sections (refer

Table 3.2 and 3.12 to made a comparison), we can see clearly that by combining color texture features and visual words the performance is further upgraded.

Table 3.17: Average error rates (± standard deviation) for different classification problems using different approaches by combining color texture and visual words.

| | Out vs. In | City vs. Landscape | Mountain vs. Forest | 4 classes |
|---|---|---|---|---|
| MGDM | 5.24% (±0.77) | 6.53% (±0.88) | 8.13% (±0.93) | 8.62% (±0.79) |
| MDM | 7.29% (±0.79) | 9.81% (±0.81) | 10.07% (±0.89) | 11.24% (±0.85) |
| MM | 9.16% (±0.80) | 11.12% (±0.83) | 12.54% (±0.85) | 12.22% (±0.88) |
| SVM+MGDM | **4.21% (±0.68)** | **5.13% (±0.69)** | **7.26% (±0.71)** | **7.41% (±0.73)** |
| SVM+MDM | 6.12% (±0.70) | 7.20% (±0.77) | 9.33% (±0.81) | 10.11% (±0.80) |
| SVM+MM | 7.98% (±0.91) | 9.88% (±0.92) | 10.84% (±0.92) | 11.19% (±0.85) |
| SVM+$RBF_{\chi^2}$ | 7.68% (±0.80) | 9.99% (±0.81) | 11.81% (±0.83) | 12.77% (±0.78) |
| SVM+Polynomial | 9.13% (±0.83) | 10.44%(±0.86) | 12.53%(±0.85) | 12.05%(±0.84) |
| SVM+$RBF_{Gaussian}$ | 9.93% (±0.95) | 10.90% (±0.94) | 12.93% (±0.97) | 12.49% (±0.89) |

We have also investigated the effect of feature selection in this experiment. The main goal is to study if feature selection can improve classification based on discrete visual features. Indeed, some of the features may be irrelevant and/or redundant and then may affect negatively the classification accuracy. In this application, we have studied the effect of some feature selection methods that have been previously used in the case of text categorization. The feature selection criteria that we have considered are document frequency (DF), $\chi^2$ statistics (CHI), mutual information (MI), and pointwise mutual information (PMI) (see [129], for instance, for more details about these criteria). In particular, for DF we have adopted two selection criteria: DF$_{max}$ which removes features above a certain threshold and DF$_{min}$ which removes features below a threshold. Figures 3.6, 3.7, 3.8 and 3.9 summarizes the classification results for the different categorization problems. These figures show that feature selection improves the classification accuracy which is consistent with the results obtained in the case of text categorization [129]. We can see also that $\chi^2$ statistics, MI, and PMI criteria reached the best results.

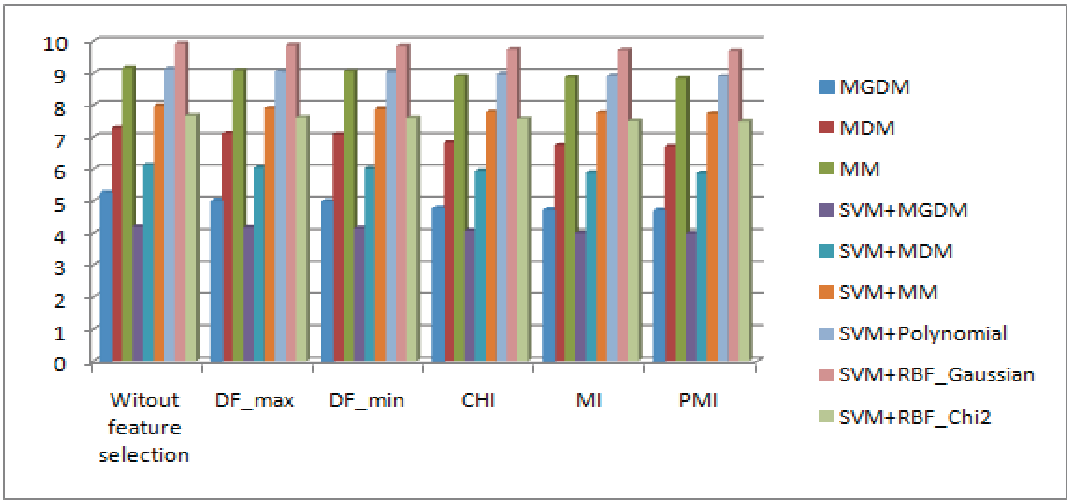


Figure 3.6: Feature Selection for the indoor vs. outdoor categorization problem.

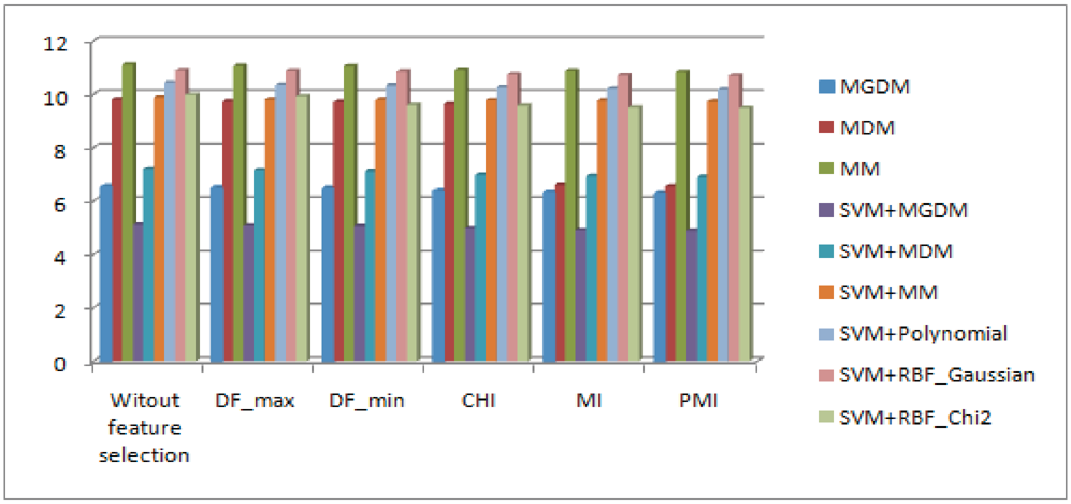


Figure 3.7: Feature Selection for the city vs. landscape categorization problem.

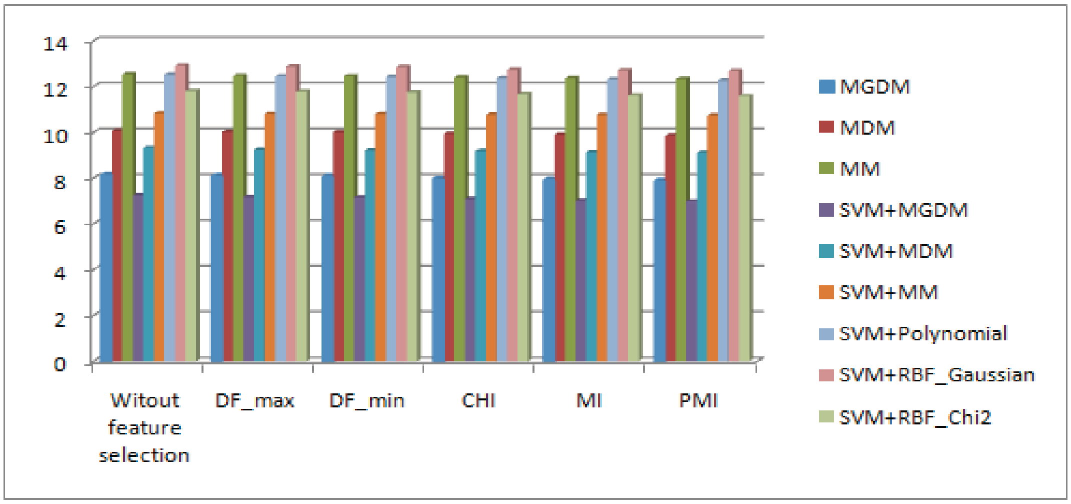


Figure 3.8: Feature Selection for the mountain vs. forest categorization problem.

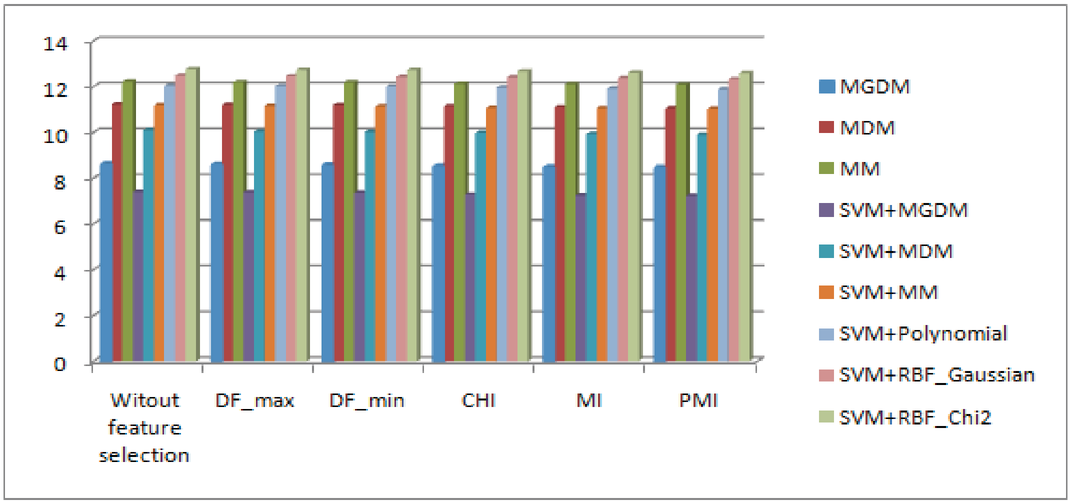


Figure 3.9: Feature Selection for the 4 classes categorization problem.

CHAPTER 4

# Conclusions

In this thesis, first we have introduced a novel statistical learning framework for discrete visual features modeling using a mixture model based on both multinomial and generalized Dirichlet mixture distributions as a prior. Second, the proposed model has been suggested as parametric basis for SVM within a hybrid generative/discriminative framework. Improvement in classification accuracy due to the new kernel is demonstrated through series of experiments.

Learning the parameters of this framework involves the computation of special functions which are intractable and can deteriorate the estimatation accuracy. This drawback motivated the development of a novel learning approach which attempts to use the LOO likelihood technique and shown to be more accurate. Effectiveness of the proposed method is demonstrated by applying it to several challenging problems such as scene classifications that involve discrete visual features modeling using visual words, also an accurate and stable statistical representation for color texture is developed. We have also proposed a hybrid generative/descriminative framework, that has shown encouraging results, by using fisher kernel from our generative model for SVM.

Our experiments were restricted to multimedia contents analysis for image processing and computer vision applications. As future work, the excellent performance of the proposed approach assures that it can also be extended to a broad range of other domains that handle features not only from discrete-valued variables but also from a mixture of continuous and discrete-valued variables which are very common in practice.

# Appendix A

## Proofs of Equations 2.12 and 2.13

By computing the first derivatives of the LOO log-likelihood, we obtain

$$
\begin{aligned}
\frac{\partial L_{LOO}(\mathcal{X}|\Theta)}{\partial \alpha_{kv}} &= \sum_{i=1}^{N} f_{iv} \frac{p(k|\vec{X}_i;\Theta)\frac{\beta^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \\
= \; & \sum_{i=1}^{N} f_{iv} \frac{p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \frac{\frac{\beta^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \\
= \; & \sum_{i=1}^{N} f_{iv} \frac{p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \frac{\partial}{\partial \alpha_{kv}} \log\left[\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}\right] \\
= \; & \sum_{i=1}^{N} f_{iv} L_{ikv} \frac{\partial}{\partial \alpha_{kv}}\left[\log \alpha^{'}_{kv} - \log(\alpha^{'}_{kv}+\beta^{'}_{kv}) + \sum_{l=1}^{v-1}\Big(\log \beta^{'}_{kl} - \log(\alpha^{'}_{kl}+\beta^{'}_{kl})\Big)\right] \\
= \; & \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{\alpha^{'}_{kv}} - \frac{1}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\right]
\end{aligned}
$$

where $L_{ikv} = \frac{p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}$

$$\frac{\partial L_{LOO}(\mathcal{X}|\Theta)}{\partial \beta_{kv}} = \sum_{i=1}^{N} f_{iv} \frac{p(k|\vec{X}_i;\Theta)\frac{-\alpha^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}$$

$$\begin{aligned}
&= \sum_{i=1}^{N} f_{iv} \frac{p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \frac{\frac{-\alpha^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\prod_{l=1}^{v-1}\frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}} \\
&= \sum_{i=1}^{N} f_{iv} L_{ikv} \frac{\partial}{\partial \beta_{kv}} \log\left[\frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}} \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}\right] \\
&= \sum_{i=1}^{N} f_{iv} L_{ikv} \frac{\partial}{\partial \beta_{kv}} \left[\log\alpha^{'}_{kv} - \log(\alpha^{'}_{kv}+\beta^{'}_{kv}) + \sum_{l=1}^{v-1}\Bigg(\log\beta^{'}_{kl} - \log(\alpha^{'}_{kl}+\beta^{'}_{kl})\Bigg)\right] \\
&= \sum_{i=1}^{N} f_{iv} L_{ikv} \left[-\frac{1}{\alpha^{'}_{kv}+\beta^{'}_{kv}}\right]
\end{aligned}$$

# APPENDIX B

## Proofs of Equations 2.14(a-c)

By observing that the first derivatives of $L_{ikv}$ w.r.t $\alpha_{kv'}$ and $\beta_{kv'}$, where $v \neq v'$ can be neglected (i.e $\frac{\partial L_{ikv}}{\partial \alpha_{kv'}} \simeq 0$ and $\frac{\partial L_{ikv}}{\partial \beta_{kv'}} \simeq 0$), we obtain the following second derivatives:

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial \alpha_{kv_1} \partial \alpha_{kv_2}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv} \Bigg[ \frac{-1}{(\alpha^{'}_{kv})^2} + \frac{1}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2} + \left( \frac{1}{\alpha^{'}_{kv}} - \frac{1}{\alpha^{'}_{kv}+\beta^{'}_{kv}} \right) \\ \times \left( \frac{\beta^{'}_{kv}}{\alpha^{'}_{kv}(\alpha^{'}_{kv}+\beta^{'}_{kv})} - L_{ikv} \frac{\beta^{'}_{kv}}{\alpha^{'}_{kv}(\alpha^{'}_{kv}+\beta^{'}_{kv})} \right) \Bigg] \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} \end{cases}$$

$$= \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv} \left[ \frac{1}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2} + \frac{(L_{ikv}-1)\beta^{'}_{kv}}{\alpha^{'}_{kv}(\alpha^{'}_{kv}+\beta^{'}_{kv})^2} - \frac{\beta^{'}_{kv} L_{ikv} + \alpha^{'}_{kv}}{(\alpha^{'}_{kv})^2(\alpha^{'}_{kv}+\beta^{'}_{kv})} \right] \\ \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} \end{cases}$$

By remarking that:

$$\begin{aligned} \frac{\partial L_{ikv}}{\partial \alpha_{kv}} &= \frac{p(k|\vec{X}_i;\Theta) \frac{\beta^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2} \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}} \sum_{k=1}^{K} p(k|\vec{X}_i;\Theta) \frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}} \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}}}{(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta) \frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}} \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}})^2} \\ &- \frac{(p(k|\vec{X}_i;\Theta) \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}})^2 \frac{\beta^{'}_{kv}}{(\alpha^{'}_{kv}+\beta^{'}_{kv})^2} \frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}}}{(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta) \frac{\alpha^{'}_{kv}}{\alpha^{'}_{kv}+\beta^{'}_{kv}} \prod_{l=1}^{v-1} \frac{\beta^{'}_{kl}}{\alpha^{'}_{kl}+\beta^{'}_{kl}})^2} \\ &= L_{ikv} \frac{\beta^{'}_{kv}}{\alpha^{'}_{kv}(\alpha^{'}_{kv}+\beta^{'}_{kv})} - L^2_{ikv} \frac{\beta^{'}_{kv}}{\alpha^{'}_{kv}(\alpha^{'}_{kv}+\beta^{'}_{kv})} \end{aligned}$$

And

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\beta_{kv_1}\partial\beta_{kv_2}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} - \frac{L_{ikv}}{(\alpha'_{kv}+\beta'_{kv})^2}\right] & \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} & \end{cases}$$

$$= \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{L_{ikv}\beta'_{kv}-\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2} + \frac{1-L_{ikv}}{(\alpha'_{kv}+\beta'_{kv})^2} - \frac{L_{ikv}\beta'_{kv}-\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2}\right] & \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} & \end{cases}$$

$$= \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(1-L_{ikv})}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})}\right] & \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} & \end{cases}$$

By remarking that:

$$\begin{aligned} \frac{\partial L_{ikv}}{\partial\beta_{kv}} &= \frac{p(k|\vec{X}_i;\Theta)\frac{-\alpha'_{kv}}{(\alpha'_{kv}+\beta'_{kv})^2}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}}\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}}}{(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}})^2} \\ &- \frac{(p(k|\vec{X}_i;\Theta)\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}})^2\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\frac{-\alpha'_{kv}}{(\alpha'_{kv}+\beta'_{kv})^2}}{(\sum_{k=1}^{K} p(k|\vec{X}_i;\Theta)\frac{\alpha'_{kv}}{\alpha'_{kv}+\beta'_{kv}}\prod_{l=1}^{v-1}\frac{\beta'_{kl}}{\alpha'_{kl}+\beta'_{kl}})^2} \\ &= L_{ikv}\frac{-1}{\alpha'_{kv}+\beta'_{kv}} + L_{ikv}^2\frac{1}{\alpha'_{kv}+\beta'_{kv}} \end{aligned}$$

And

$$\frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\alpha_{kv_1}\partial\beta_{kv_2}} = \frac{\partial^2 L_{LOO}(\mathcal{X}|\Theta)}{\partial\beta_{kv_2}\partial\alpha_{kv_1}} = \begin{cases} \sum_{i=1}^{N} f_{iv} L_{ikv}\left[\frac{1}{(\alpha'_{kv}+\beta'_{kv})^2} + \frac{(L_{ikv}-1)\beta'_{kv}}{\alpha'_{kv}(\alpha'_{kv}+\beta'_{kv})^2}\right] \\ \text{if } v_1 = v_2 = v \\ 0 \qquad \text{otherwise} \end{cases}$$

# List of References

[1] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[2] N. Bouguila and W. Elguebaly. Discrete Data Clustering Using Finite Mixture Models. *Pattern Recognition*, 42(1):33–42, 2009.

[3] www.flicker.com. *url link at http://blog.flickr.net/en/2010/09/19/5000000000/*, Sep 2010.

[4] A. K. Jain, R. P. W. Duin and J. Mao. Statistical Pattern Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000.

[5] V.K Govindan and A.P Shivaprasad. Character Recognition – A Review. *Pattern Recognition*, 23(7):671 – 683, 1990.

[6] J. Krizaj, V. Struc, and N. Pavesic. Adaptation of SIFT Features for Robust Face Recognition. In *Proc. of the International Conference on Image Analysis and Recognition*, pages 394–404, 2010.

[7] S. Antani. A survey on the Use of Pattern Recognition Methods for Abstraction, Indexing and Retrieval of Images and Video. *Pattern Recognition*, 35(4):945–965, Apr 2002.

[8] A.L. Meyrowitz, D.R. Blidberg, and R.C. Michelson. Autonomous Vehicles. In *Proc. of the IEEE*, volume 84, pages 1145–1164, Aug 1996.

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

[10] J. D. Banfield and A. E. Raftery. Model-based Gaussian and Non-Gaussian Clustering. *Biometrics*, 49(3):803–821, 1993.

[11] 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.

[12] J. D. M. Rennie, L. Shih, J. Teevan and D. R. Karger. Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In *Proc. of the 20th International Conference on Machine Learning (ICML)*, pages 616–623. Morgan Kauffmann, 2003.

[13] R. E. Madsen, D. Kauchak and C. Elkan. Modeling Word Buristness Using the Dirichlet Distribution. In *Proc. of the 22nd International Conference on Machine Learning (ICML)*, pages 545–552. ACM Press, 2005.

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

[15] 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.

[16] S. M. Katz. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech and Singal processing*, 35(3):400–401, Mar 1987.

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

[18] T. Dunning. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[19] C. Zhai and J. Lafferty. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.

[20] N. Bouguila, D. Ziou and J. Vaillancourt. Novel Mixtures Based on the Dirichlet Distribution: Application to Data and Image Classification. In *Machine Learning and Data Mining in Pattern Recognition (MLDM)*, pages 172–181. Springer, LNAI2734, 2003.

[21] N. Bouguila and D. Ziou. Improving Content Based Image Retrieval Systems Using Finite Multinomial Dirichlet Mixture. In *Proc. of the IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pages 23–32, 2004.

[22] N. Bouguila and W. Elguebaly. On Discrete Data Clustering. In *Proc. of the 12th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD)*, pages 503–510. Springer, LNAI 5012, 2008.

[23] N. Bouguila. Clustering of Count Data Using Generalized Dirichlet Multinomial Distributions. *IEEE Transactions on Knowledge and Data Engineering*, 20(4):462–474, 2008.

[24] K.S. Chou, K.C. Fan, and T.I. Fan. Peripheral and Global Features for Use in Coarse Classification of Chinese Characters. *Pattern Recognition*, 30(3):483–489, Mar 1997.

[25] T. M. Lehmann, M. O. Guld, T. Deselaers, D. Keysers, H. Schubert, K. Spitzer, H. Ney, and B. B. Wein. Automatic Categorization of Medical Images for Content-based Retrieval and Data Mining. *Computerized Medical Imaging and Graphics*, 29(2-3):143–155, 2005.

[26] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, 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.

[27] A. Vailaya, M. A. T. Figueiredo, A. K. Jain and H-J. Zhang. Image Classification for Content-Based Indexing. *IEEE Transactions on Image Processing*, 10(1):117–130, 2001.

[28] X. Shen M. R. Boutell, J. Luo and C. M. Brown. Learning Multi-label Scene Classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[29] M. Tuceryan and A. K. Jain. Texture Analysis. In *The Handbook of Pattern Recognition and Computer Vision*, pages 207–248, 1998.

[30] C. Palm. Color Texture Classification by Integrative Co-occurrence Matrices. *Pattern Recognition*, 37(5):965 – 976, 2004.

[31] A. Drimbarean and P. F. Whelan. Experiments in Colour Texture Analysis. *Pattern Recognition Letters*, 22(10):1161 – 1167, 2001.

[32] A. L. Vickers and J. W. Modestino. A Maximum Likelihood Approach to Texture Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(1):61–68, 1982.

[33] M. Unser. Sum and Difference Histograms for Texture Classification. *Pattern Recognition and Machine Intelligence*, 8(1):118–125, 1986.

[34] K. Mikolajczyk. *Detection of Local Features Invariant to Affine Transformations- Application to Matching and Recognition*. PhD thesis, Institut National Polytechnique De Grenoble, Jul 2002.

[35] Gy. Dorkó and C. Schmid. Selection of Scale-Invariant Parts for Object Class Recognition. In *Proc. of the IEEE Conference on Computer Vision*, volume 1, page 634, 2003.

[36] D. Grangier and S. Bengio. A Discriminative Kernel-based Approach to Rank Images from Text Queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1371–1384, 2008.

[37] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer. Generic Object Recognition with Boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):416–431, 2006.

[38] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. V. Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision*, 65(1):43–72, Nov 2005.

[39] T. Tuytelaars and L. V. Gool. Content-based Image Retrieval Based on Local Affinely Invariant Regions. In *Proc. of the International Conference on Visual Information Systems*, pages 493–500, 1999.

[40] R. Fergus, P. Perona, and A. Zisserman. Object Class Recognition by Unsupervised Scale-invariant Learning. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, volume 2, pages 264–271, 2003.

[41] K. Mikolajczyk and C. Schmid. Indexing Based on Scale Invariant Interest Points. In *Proc. of the 8th International Conference on Computer Vision*, pages 525–531, 2001.

[42] C. Schmid and R. Mohr. Local Grayvalue Invariants for Image Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–535, 1997.

[43] J. Sivic and A. Zisserman. Video Google: A Text Retrieval Approach to Object Matching in Videos. In *Proc. of the 9th IEEE International Conference on Computer Vision*, volume 2, pages 1470–1477, Apr 2003.

[44] S. Lazebnik, C. Schmid and J. Ponce. A Sparse Texture Representation Using Local Affine Regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, 2005.

[45] C. G. Harris and M. J. Stephens. A Combined Corner and Edge Detector. *Proc. of the Alvey Vision Conference*, pages 147–152, 1988.

[46] K. Mikolajczyk and C. Schmid. An Affine Invariant Interest Point Detector. In *Proc. of 7th European Conference on Computer Vision - Part I*, pages 128–142, 2002.

[47] P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, 31(4):532–540, Apr 1983.

[48] T. Tuytelaars and K. Mikolajczyk. Local Invariant Feature Detectors: A Survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2007.

[49] J. J. Koenderink. The Structure of Images. *Biological Cybernetics*, 50:363–396, 1984.

[50] T. Linderberg. Scale-space Theory: A Basic Tool for Analysing Structures at Different Scales. *Journal of Applied Statistics*, 21(2):224–270, 1994.

[51] T. Lindeberg. Feature Detection with Automatic Scale Selection. *International Journal of Computer Vision*, 30(2):79–116, Nov 1998.

[52] D. G. Lowe. Object Recognition from Local Scale-invariant Features. In *Proc. of the International Conference on Computer Vision*, volume 2, pages 1150–1157, Sep 1999.

[53] K. Mikolajczyk and C. Schmid. A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[54] T. Lindeberg and J. Garding. Shape-adapted Smoothing in Estimation of 3-D Depth Cues from Affine Distortions of Local 2-D Brightness Structure. *Image and Vision Computing*, 15(6):415–434, June 1997.

[55] K. Mikolajczyk and C. Schmid. Scale & Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[56] F. Schaffalitzky and A. Zisserman. Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?". In *Proc. of the 7th European Conference on Computer Vision-Part I*, pages 414–431. Springer-Verlag, 2002.

[57] A. Baumberg. Reliable Feature Matching Across Widely Separated Views. In *Proc. of the IEEE International conference on Computer Vision and Pattern Recognition*, volume 1, pages 774–781, 2000.

[58] J. J. Koenderink and A. J. van Doom. Representation of Local Geometry in the Visual System. *Biological Cybernetics*, 55(6):367–375, 1987.

[59] X. Wu and B. Bhanu. Gabor Wavelet Representation for 3-D Object Recognition. *IEEE Transactions on Image Processing*, 6(1):47–64, 1997.

[60] L. V. Gool, T. Moons, and D. Ungureanu. Affine/ Photometric Invariants for Planar Intensity Patterns. In *Proc. of the 4th European Conference on Computer Vision-Volume I*, pages 642–651, 1996.

[61] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. SURF: Speeded Up Robust Features. In *Proc. of the European Conference on Computer Vision*, pages 404–417, 2006.

[62] W. T. Freeman and E. H. Adelson. The Design and Use of Steerable Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

[63] A. Vardy and F. Oppacher. A Scale Invariant Local Image Descriptor for Visual Homing. In *MirrorBot Project*, pages 362–381, 2005.

[64] R. Zabih and J. Woodfill. Non-parametric Local Transforms for Computing Visual Correspondence. In *Proc. of the European Conference on Computer Vision*, volume 2, pages 151–158, 1994.

[65] S. Kim and I. S. Kweon. Biologically Motivated Perceptual Feature: Generalized Robust Invariant Feature. In *Proc. of the Asian Conference on Computer Vision*, volume 2, pages 305–314, 2006.

[66] Y. Ke and R. Sukthankar. PCA-SIFT: A more Distinctive Representation for Local Image Descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 506–513, 2004.

[67] G. Fritz, C. Seifert, M. Kumar, and L. Paletta. Building Detection from Mobile Imagery Using Informative SIFT Descriptors. In *Proc. of the Scandinavian Conference on Image Analysis*, pages 629–638, 2005.

[68] A. E. Abdel-Hakim and A. A. Farag. CSIFT: A SIFT Descriptor with Color Invariant Characteristics. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1978–1983, June 2006.

[69] R. M. Haralick, K. Shanmugan and I. Dinstein. Texture Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, 8:610–621, 1973.

[70] T. Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *Proc. of the 10th European Conference on Machine Learning*, number 1398, pages 137–142, 1998.

[71] S. Tong and D. Koller. Support Vector Machine Active Learning with Applications to Text Classification. In *Proc. of 17th International Conference on Machine Learning (ICML)*, pages 999–1006, 2000.

[72] H. Lodhi, C. Saunders, J. S.-Taylor, N. Cristianini, and C.J.C.H. Watkins. Text Classification Using String Kernels. *Machine Learning Research*, 2:419–444, 2002.

[73] L. Zhu, A. Rao and A. Zhang. Theory of Keyblock-based Image Retrieval. *ACM Transactions on Information Systems*, 20(2):224–257, 2002.

[74] G. Csurka, C. R. Dance, L. Fan, J. Willamowski and C. Bray. Visual Categorization with Bags of Keypoints. In *Workshop on Statistical Learning in Computer Vision, 8th European Conference on Computer Vision (ECCV)*, pages 1–22, 2004.

[75] T. Bayes. An Essay Towards Solving a Problem in the Doctrine of Chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418, 1763.

[76] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Machine Learning Research*, 3:993–1022, Jan 2003.

[77] G. E. Hinton. Boltzmann Machine. *Scholarpedia*, 2(5):1668, 2007.

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

[79] P. Domingos and M. Pazzani. On the Optimality of Simple Bayesian Classifier Under Zero-one Loss. *Machine Learning*, pages 103–130, Nov 1997.

[80] E. McDermott, T. J. Hazen, J. L. Roux, A. Nakamura, and S. Katagiri. Discriminative Training for Large-Vocabulary Speech Recognition Using Minimum Classification Error. *IEEE Transactions on Audio, Speech & Language Processing*, 15(1):203–223, 2007.

[81] X. Ren and J. Malik. Learning a Classification Model for Segmentation. In *Proc. of the 9th International Conference on Computer Vision*, volume 1, pages 10–17, 2003.

[82] T. Deselaers, D. Keysers, and H. Ney. Discriminative Training for Object Recognition Using Image Patches. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 157–162, June 2005.

[83] D. Johnson. Use of Discriminative Models of Pseudomonas Neuruginosa Bacteremia in Granulocytopenic Rats for Testing Antimicrobial Efficacy. *European Journal of Clinical Microbiology; Infectious Diseases*, 4:207–212, 1985. 10.1007/BF02013599.

[84] C. J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[85] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, Sep 1998.

[86] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An Introduction to Kernel-based Learning Algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.

[87] I. Ulusoy and C. M. Bishop. Generative Versus Discriminative Methods for Object Recognition. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2005.

[88] R. Raina, Y. Shen, A. Y. Ng and A. McCallum. Classification with Hybrid Generative/Discriminative Models. In *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2003.

[89] Y. Li, L. G. Shapiro and J. A. Bilmes. A Generative/Discriminative Learning Algorithm for Image Classification. In *Proc. of the 10th IEEE International Conference on Computer Vision (ICCV)*, pages 1605–1612, 2005.

[90] J. A. Lasserre. *Hybrid of Generative and Discriminative Methods for Machine Learning*. PhD thesis, Queens' College, University of Cambridge, Mar 2008.

[91] Nizar Bouguila and Mukti Nath Ghimire. Discrete Visual Features Modeling via Leave-one-out Likelihood Estimation and Applications. *Journal of Visual Communication and Image Representation (JVCIR)*, 21(7):613 – 626, 2010.

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

[93] J. Huang, S. R. Kumar, M. Mitra, W. Zhu and R. Zabih. Spatial Color Indexing and Applications. *International Journal of Computer Vision*, 35(3):245–268, 1999.

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

[95] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez and T. Tuytelaars. A Thousand Words in a Scene. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1575–1589, 2007.

[96] J. C. Stoffel. A Classifier Design Technique for Discrete Variable Pattern Recognition Problems. *IEEE Transactions on Computers*, 23(4):428–441, 1974.

[97] I. J. Good. The Population Frequencies of Species and the Estimation of Population Parameters. *Biometrika*, 40(3-4):237–264, 1953.

[98] I. J. Good. *The Estimation of Probabilities*. Cambridge, Mass: MIT Press, 1965.

[99] S. E. Fienberg and P. W. Holland. Simultaneous Estimation of Multinomial Cell Probabilities. *Journal of the American Statistical Association*, 68(343):683–691, 1973.

[100] M. Stone. Cross-validation and Multinomial Prediction. *Biometrika*, 61(3):509–545, 1974.

[101] I. Dagan, L. Lee and F. C. N. Pereira. Similarity-based Models of Words Co-occurrence Probabilities. *Machine Learning*, 34(1-3):43–69, 1999.

[102] H. Ney and U. Essen. On Smoothing Techniques for Bigram-based Natural Language Modelling. In *Proc. of the IEEE Intenational Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 825–828, 1991.

[103] E. L. Lehmann. *Theory of Point Estimates*. Wiley, New York, 1983.

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

[105] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[106] 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.

[107] 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.

[108] 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.

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

[110] A. P. Dempster, N. M. Laird and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, B*, 39:1–38, 1977.

[111] F. Mosteller and D. L. Wallace. Inference in an Authorship Problem. *Journal of the American Statistical Association*, 58(302):275–309, 1963.

[112] H. Ney, U. Essen and R. Kneser. On the Estimation of "Small" Probabilities by Leaving-One-Out. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1202–1212, 1995.

[113] B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.

[114] T. P. Minka. Estimating a Dirichlet Distribution. *Unpublished paper available at http://research.microsoft.com/∼minka/papers/dirichlet/.*

[115] H. Robbins. An Empirical Bayes Approach to Statistics. In *Proc. of the 3rd Berkley Symposium on Mathematical Statistics and Probability*, pages 131–148. University of California Press, 1956.

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

[117] M. A. T. Figueiredo, J. M. N. Leitão and A. K. Jain. On Fitting Mixture Models. In *Proc. of the 2nd International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, pages 54–69, 1999.

[118] C. Cortes and V. Vapnik. Support Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

[119] T. S. Jaakkola and D. Haussler. Exploiting Generative Models in Discriminative Classifiers. In *Advances in Neural Information Processing Systems (NIPS)*, pages 487–493. MIT Press, 1999.

[120] I. Biederman. On the Semantics of a Glance at a Scene. In M. Kubovy and J. R. Pomeranz, editor, *Perceptual Organizations*, pages 213–253. Hillsdale, NJ: Lawrence Erlbaum, 1981.

[121] I. Biederman. Aspects and Extensions of a Theory of Human Image Understanding. In Z. W. Pylyshyn, editor, *Computational Processes in Human Vision: An Interdisciplinary Perspective*, pages 370–428. Norwood, NJ: Ablex, 1988.

[122] P. G. Schyns and A. Oliva. From Blobs to Boundary Edges: Evidence for Time and Spatial Scale Dependent Scene Recognition. *Psychological Science*, 5:195–200, 1994.

[123] A. Vailaya, A. K. Jain and H-J. Zhang. On Image Classification: City Images Vs. Landscapes. *Pattern Recognition*, 31(12):1921–1935, 1998.

[124] L. Fei-Fei and P. Perona. A Bayesian Hierarchical Model for Learning Natural Scene Categories. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 524–531, 2005.

[125] O. Chapelle, P. Haffner and V. N. Vapnik. Support Vector Machines for Histogram-based Image Classification. *IEEE Transactions on Neural Networks*, 10(5):1055–1064, 1999.

[126] A. Drimbarean and P. F. Whelan. Experiments in Colour Texture Analysis. *Pattern Recognition Letters*, 22(10):1161–1167, 2001.

[127] M. Unser. Sum and Difference Histograms for Texture Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):118–125, 1986.

[128] J. Beck. Similarity Grouping and Peripheral Discriminability under Uncertainty. *American Journal of Psychology*, 85(1):1–19, 1972.

[129] Y. Yang and J. O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 412–420. Morgan Kauffmann, 1997.