DISTRIBUTIONAL FEATURE MAPPING IN DATA CLASSIFICATION

Md. Hafizur Rahman

A thesis
in
The Department
of
Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Masters of Applied Science
(Quality Systems Engineering)
Concordia University
Montréal, Québec, Canada

March 2020

Concordia University

School of Graduate Studies

This is to certify that the thesis prepared

By: **Md. Hafizur Rahman**

Entitled: **Distributional Feature Mapping in Data Classification**

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

Masters of Applied Science
(Quality Systems Engineering)

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

Signed by the final examining committee:

Dr. Walter Lucia ______________________ Chair
Dr. Nizar Bouguila ______________________ Supervisor
Dr. Jamal Bentahar ______________________ CIISE Examiner
Dr. Zahangir Kabir ______________________ External Examiner

Approved ______________________________
Dr. Mohammad Mannan, Graduate Program Director

__________ 20 _____ ______________________________
Dr. Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

Distributional Feature Mapping in Data Classification

Md. Hafizur Rahman

Performance of a machine learning algorithm depends on the representation of the input data. In computer vision problems, histogram based feature representation has significantly improved the classification tasks. *L1* normalized histograms can be modelled by Dirichlet and related distributions to transform input space to feature space. We propose a mapping technique that contains prior knowledge about the distribution of the data and increases the discriminative power of the classifiers in supervised learning such as Support Vector Machine (SVM). The mapping technique for proportional data which is based on Dirichlet, Generalized Dirichlet, Beta Liouville, scaled Dirichlet and shifted scaled Dirichlet distributions can be incorporated with traditional kernels to improve the base kernels accuracy. Experimental results show that the proposed technique for proportional data increases accuracy for machine vision tasks such as natural scene recognition, satellite image classification, gender classification, facial expression recognition and human action recognition in videos. In addition, in object tracking, learning parametric features of the target object using Dirichlet and related distributions may help to capture representations invariant to noise. This further motivated our study of such distributions in object tracking. We propose a framework for feature representation on probability simplex for proportional data utilizing the histogram representation of the target object at initial frame. A set of parameter vectors determine the appearance features of the target object in the subsequent frames.

Motivated by the success of distribution based feature mapping for proportional data, we extend this technique for semi-bounded data utilizing inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions. Similar approach is taken into account for count data where Dirichlet multinomial and generalized Dirichlet multinomial distributions are used to map density features with input features.

Acknowledgments

First of all, I am grateful to Almighty for bestowing His blessings on me to finish this thesis on time. I would like to express my gratitude to my supervisor Prof. Nizar Bouguila, who has given me the opportunity to come abroad and work in his research lab. I will always be grateful for his relentless support and guidance. I am grateful to my parents for their continuous support in pursuing my higher studies abroad.

I am thankful to my sister Sharmin, who helped me to get adapted here in Canada. My gratitude to my other elder sister and brother who encouraged me all the way along during this time. My love to my cute nephew Raaida, who brought smile on my face all the time.

I appreciate a lot all my colleagues in the lab specially Samr, Fatma, Dr. Nuha, Ornella and Xavier for their insightful discussion about machine learning research. I would like to extend my gratitude to my roommate Mithun, who is like a brother to me for his continuous help and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Appropriate and accurate representation of the data for classification models is one of the existing problems in machine learning. Several classification and hybrid models have been developed, yet a little attention has been given to a get a proper representation of the data through distribution based feature mapping in discriminative approaches [103]. In this thesis, we address this issue in supervised learning problems for proportional, semi-bounded and count data. A popular image representation is the Bag of Visual Words (BoVW) approach which is essentially quantizing similar patches of an image to the corresponding cluster center which is known as codebook [8], [105]. Modelling such data after normalization in a probabilistic manner needs to satisfy the constraints of non-negativity and unit sum. Examples of such data includes *L1* normalized histogram for images and normalized bag of words representation of texts (or images) data. In particular, we are motivated by the problem of modelling features in images and videos where each feature represents a portion of the total features considered. For example, an image can be represented by a normalized histogram of bag of vectors where each vector element represents a sub-region of the image. Knowledge about statistical characteristics of such representations has to be used effectively in order to get better classification accuracy. Dirichlet and related distributions can model this type of data to get the prior information which can be used as a feature. The advantages of such distributions are that they can capture the nature of the data and provide flexibility. For support vector machine (SVM), traditional kernels do not take into account the nature of the data. Utilizing our proposed feature mapping technique increases the classification accuracy of these kernels.

Incorporating invariances in the representation using prior knowledge is a common technique to make the learning task more efficient and in general form, prior information makes it possible to generalize training examples to novel test examples [103]. In supervised learning, hyperparameters of the classifiers work as prior information. Another approach is to select features that convey most relevant information regarding the data or the task. Such features are automatically incorporated in some kernels such polynomial kernel for SVM [35]. On a different note, distribution based flexible feature mapping can be efficient in different classification tasks [82]. For SVM, input data are represented as points in high dimensional space. This representation needs to be linearly separable to make the model work properly. Therefore, for non-linear data, performance of SVM model lacks accuracy. However, kernel trick or feature mapping technique has made it possible to model non-linear data which is essentially taking the data space to higher dimension where the data become linearly separable. It is a common idea to extract new features from the input variables through a feature mapping function which increases the separability between the data classes. On the contrary, feature mapping without statistical measure about the data does not guarantee the improvement in model's performance. Selecting the most informative attributes from the set of redundant attributes is suboptimal for a classifier and on the contrary, it may keep out some relevant features as well [56]. Hence, extracting or creating new features from the data with prior information using a parameterized feature mapping function can be incorporated in classification model with certain degree of confidence. Histogram representation of the extracted data can be modelled in a probabilistic way by performing *L1*-normalization and Dirichlet or Liouville type distributions is the choice to estimate the density of such data. Therefore, a parametric distribution based mapping function can be developed to increase the quantization capability of the classification model.

1.2 Contributions

The main objective of this thesis is to study the effectiveness of combining different distributions based features with the input features to improve performance of discriminative classifiers accuracy such as SVM.

- **Efficient feature mapping using Dirichlet, generalized Dirichlet and Beta Liouville, scaled Dirichlet and shifted scaled Dirichlet distributions**
  We propose a new feature mapping technique for proportional data to improve the classification accuracy. *L1* normalized histograms can be modelled by Dirichlet and related distributions to transform input space to feature space. A paper based on Dirichlet, generalized Dirichlet and Beta Liouville based feature mapping technique has been submitted to *Neurocomputing* [93] and is under review. An extension of this paper based on scaled Dirichlet and shifted scaled Dirichlet distributions has been submitted to *IEEE International Symposium on Networks, Computers and Communications 2020* [95].

- **Feature mapping for semi-bounded data using inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions**
  We extend our previous contribution for inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions. This contribution has been submitted to *IEEE International Conference on Systems, Man, and Cybernetics, 2020* [94].

- **Feature mapping for count data using Dirichlet multinomial and generalized Dirichlet multinomial distributions**
  A statistically flexible feature mapping technique for count data is proposed using Dirichlet multinomial and generalized Dirichlet multinomial distributions. This contribution has been published at *IEEE Symposium on Computational Intelligence and Data Mining, 2019* [92].

- **Parametric Features on Simplex Manifold for Online Object Tracking**
  We propose a framework to concatenate density based features in simplex manifold with raw features to improve object tracking performance. We also discuss how to approximate non-linear kernel with the proposed approach. This

contribution has been submitted to *IEEE International Conference on Image Processing, 2020* [91].

1.3 Thesis Overview

□ Chapter 1 introduces the concepts of feature mapping and various related works on Support Vector Machine (SVM) and kernel functions. We also explain the motivation behind this work.

□ In chapter 2, we present efficient feature mapping approach for proportional data using Dirichlet, generalized Dirichlet, Beta Liouville, scaled Dirichlet and shifted scaled Dirichlet distributions. We demonstrate the effectiveness of the proposed method in computer vision problems by solving several recognition tasks such as natural scene recognition, satellite image classification, human action recognition in videos, gender classification and facial expression recognition.

□ In chapter 3, we extend the idea proposed in chapter 2 for inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions. Experiments with various applications such as image classification, action recognition and texture classification are described in details.

□ In chapter 4, we present Dirichlet multinomial and generalized Dirichlet multinomial distributions based feature mapping for count data.

□ In chapter 5, we extend our ideas in online object tracking. Also, we present a framework to approximate non-linear kernels with our proposed feature mapping approach.

□ We summarize our overall contributions in chapter 6 with concluding remarks.

Chapter 2

Feature Mapping for Classifying Proportional Data

2.1 Distributions for Proportional Data

In this chapter, we detail our proposed feature mapping technique for classifying proportional data. First, we present our selected distributions for proportional data which are bounded to unit simplex. Next, we discuss SVM learning algorithm along with feature mapping function. Finally, we show the experimental results of the proposed technique in details.

2.1.1 Dirichlet Distribution

Dirichlet distribution is the generalization of Beta distribution and most appropriate candidate in probability and statistics when modelling proportional data [31, 83]. It is a distribution over the multinomials in a simplex with support $[0, 1]$. If a vector $\mathbf{p} = (p_1, p_2, \ldots, p_D)$ of length D resides in a D dimensional closed simplex of $\mathbb{R}^D$, then it can be defined as,

$$\mathbb{C}(1) = \{\mathbf{p} \in \mathbb{R}^D : p_1 + p_2 + \ldots + p_D = 1, p_d \geq 0, 1 \leq d \leq D \text{ for all d}\} \tag{1}$$

If the proportional vector $\mathbf{p} \in \mathbb{C}(1)$ [1], then the joint probability density function

[1] $\mathbb{C}(n) = \mathbb{C}(1)$; n = sum of the multinomials

of $\mathbf{p} = (p_1, p_2, \ldots, p_D)$ is defined as,

$$\begin{aligned} P(\mathbf{p}|\boldsymbol{\alpha}) &= \frac{\Gamma\left(\sum_d \alpha_d\right)}{\prod_{d=1}^{D}\Gamma(\alpha_d)} \prod_{d=1}^{D} p_d^{\alpha_d - 1} \\ \sum_{d=1}^{D} p_d &= 1 \, , p_d \geq 0 \end{aligned} \tag{2}$$

Here, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_D)$ is a positive parameter vector which defines the shape of the distribution in D dimensional space. Total mass, $\alpha_0 = \sum_d \alpha_d$ is the concentration or scale parameter and the base measure is $(\alpha'_1, \alpha'_2, \ldots, \alpha'_D) = \frac{\alpha_d}{\alpha_0}$. In case of symmetric distribution, the mean of the distribution is determined by the base measure. In addition, altering the measurements in $\boldsymbol{\alpha}$ affects the variance of the distribution.

$$\begin{aligned} E(p_d) &= \frac{\alpha_d}{\alpha_0} = \alpha'_d \\ Var(p_d) &= \frac{\alpha_d(\alpha_0 - \alpha_d)}{\alpha_0^2(\alpha_0 + 1)} = \frac{\alpha'_d(1 - \alpha'_d)}{\alpha_0 + 1} \\ Cov(p_d, p_f) &= \frac{-\alpha_d \alpha_f}{\alpha_0^2(\alpha_0 + 1)} \end{aligned} \tag{3}$$

It should be noted that, small values of the concentration parameter α_0 favors the extreme values of the density function and as a result, data are distributed all over the simplex and it is more compact at the corner of the simplex. The shape parameter $\boldsymbol{\alpha}$ makes it possible to model data in linear, convex and concave hulls [82]. Figure 1 shows the flexibility of the distribution by changing the parameters. α_0 controls the peak of the distribution and α_d determines the location of the peak. If the expected values of the parameters are equal then data are distributed uniformly over the simplex. The higher the parameter value, more confident we are about that parameter and hence density values are more peaked on that side.

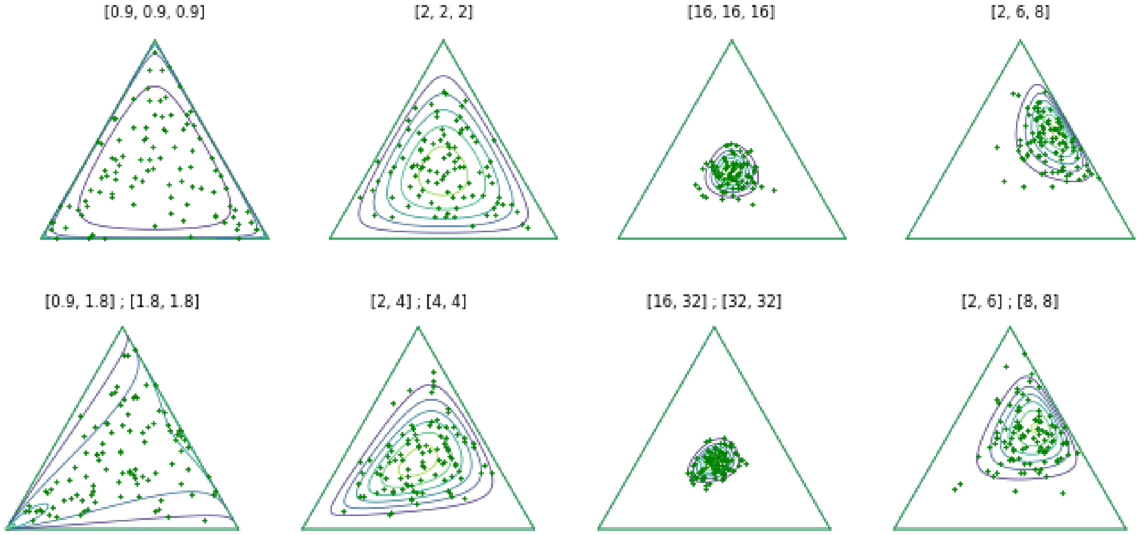


Figure 1: Peak of the Dirichlet and generalized Dirichlet distributions generated using four different sets of parameters.

2.1.2 Generalized Dirichlet Distribution

From Eq.(3), we see that any two random variables following Dirichlet distribution are negatively correlated. If the variables are positively correlated, then Dirichlet prior is not a proper choice. A modification in such cases is the generalized Dirichlet (GD) distribution which entertains both negatively and positively correlated random variables [32,115]. In dimension D, generalized Dirichlet distribution with parameter vector $\boldsymbol{\theta} = (\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_D, \beta_D)$ is defined as,

$$P(\mathbf{p}|\boldsymbol{\theta}) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} p_d^{\alpha_d - 1} \Big(1 - \sum_{l=1}^{d} p_l\Big)^{\gamma_d} \tag{4}$$

Here, $\sum_{d=1}^{D} p_d < 1$, and $0 < p_d < 1$ for $d = 1, 2, \ldots, D$ where $\alpha_d > 0, \beta_d > 0$ and $\gamma_d = \beta_d - \alpha_{d+1} - \beta_{d+1}$, $\gamma_D = \beta_D - 1$ for $d = 1, 2, \ldots, D$. GD becomes Dirichlet distribution when $\beta_d = \alpha_{d+1} + \beta_{d+1}$. If a vector $\mathbf{p} \sim GD(\alpha_1, \beta_1, \ldots, \alpha_D, \beta_D)$, then it can be transformed to follow independent Beta distributions for each dimension using the following transformation proposed by [44]-

$$
\begin{aligned}
&z1 = p1 \\
&z_d = \frac{p_d}{1 - \sum_{j=1}^{d-1} p_j} \\
&p_d = z_d(1 - p_1 - p_2 -, \dots, p_{d-1}) = z_d \prod_{j=1}^{d-1}(1 - z_j)
\end{aligned}
\tag{5}
$$

It is evident that generalized Dirichlet distribution has $2D$ number of parameters. Unlike Dirichlet distribution where the expectation is fixed, in GD distribution, the expectation of each dimension d continues to evolve over the dimension $d - 1$.

$$
E[p_d] = \frac{\alpha_d}{\alpha_d + \beta_d} \prod_{j=1}^{d-1} \frac{\beta_j}{\alpha_j + \beta_j} \qquad d = 1, 2, \dots, D \tag{6}
$$

$$
Cov(p_d, p_f) = E(p_f)\Big(\frac{\alpha_d}{\alpha_d + \beta_d + 1} \prod_{j=1}^{d-1} \frac{\beta_j + 1}{\alpha_j + \beta_j + 1}\Big) \qquad d, f = 1, 2, \dots, D \tag{7}
$$

Flexible covariance structure of GD distribution enables it to have different degrees of belief on random variables while keeping the same expectation [115]. From Fig. 1, it is evident that for Dirichlet distribution, symmetrically distributed random variables are less concentrated at the center (for example, $\alpha = [2, 2, 2]$) than the random variables following generalized Dirichlet distribution which are more concentrated at the center asymmetrically ($\alpha = [2, 4]; \beta = [4, 4]$). It can be shown that generalized Dirichlet distribution reduces to Dirichlet distribution when $\beta_d = \alpha_{d+1} + \beta_{d+1}$ (see [26] for details). If the expectation is varied and for example when $\alpha = [2, 6]; \beta = [6, 8]$ in Fig. 1, generalized Dirichlet distribution captures the variation of the data more flexibly.

2.1.3 Beta-Liouville distribution

While generalized Dirichlet distribution is more flexible than Dirichlet distribution, it requires twice the number of parameters. An efficient replacement for Dirichlet and generalized Dirichlet distributions is the Beta-Liouville distribution which overcomes the limitations of Dirichlet distribution and requires less parameters to estimate than generalized Dirichlet distribution [23]. Vector, $\mathbf{p} = (p_1, p_2, \ldots, p_D)$ will follow a Liouville distribution if and only if $\mathbf{p} \overset{d}{=} u\mathbf{q}$ where $\mathbf{q} = (q_1, q_2, \ldots, q_D) = (\frac{p_1}{\sum p}, \frac{p_2}{\sum p}, \ldots, \frac{p_D}{\sum p})$ $\sim Dir(\alpha_1, \alpha_2, \ldots, \alpha_D)$ and $u = \sum_{d=1}^{D} p_d$ is an independent random variable with density function $f(\cdot)$. The joint distribution is the density function of the Liouville distribution.

$$\begin{aligned} P(p_1, p_2, \ldots, p_D | \alpha_1, \alpha_2, \ldots, \alpha_D) &= f(u)\ Dir(\mathbf{q}|\boldsymbol{\alpha}) \\ &= f(u) \frac{\Gamma(\sum \alpha_d)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \prod_{d=1}^{D} q_d^{\alpha_d - 1} \end{aligned} \tag{8}$$

The mean, variance and covariance are given by [25],

$$E[p_d] = E[u] \frac{\alpha_d}{\alpha_0} \tag{9}$$

$$Var(p_d) = E[u^2] \frac{\alpha_d(\alpha_d + 1)}{\alpha_0(\alpha_0 + 1)} - E[u]^2 \frac{\alpha_d^2}{(\alpha_0)^2} \tag{10}$$

$$Cov(p_l, p_k) = \frac{\alpha_l \alpha_k}{\alpha_0} \Big[\frac{E(u^2)}{\alpha_0 + 1} - \frac{E(u)^2}{\alpha_0} \Big]\ ;\ l \neq k \tag{11}$$

Development of Beta-Liouville density function

In the joint density function if the generating variate $u = 1$, then it becomes Dirichlet distribution. If it follows $B(\alpha, \beta)$ that is u is defined over $[0, 1]$ then distribution with generating density,

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

$$E[u] = \frac{\alpha}{\alpha + \beta} \tag{13}$$

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

$$Var(u) = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)} \tag{15}$$

The density generator $g(\cdot)$ becomes,

$$\begin{aligned} g(u) &= \frac{\alpha_0}{B(\alpha,\beta)u^{\alpha_0-1}}u^{\alpha-1}(1-u)^{\beta-1} \\ &= \frac{\alpha_0}{B(\alpha,\beta)}u^{\alpha-\alpha_0}(1-u)^{\beta-1} \end{aligned} \tag{16}$$

Finally, the joint probability density function of $\mathbf{p}$ becomes,

$$P(\mathbf{p}|\alpha_1,\ldots,\alpha_D;\alpha,\beta) = \frac{\Gamma(\alpha_0)}{B(\alpha,\beta)}\prod_{d=1}^{D}\frac{p_d^{\alpha_d-1}}{\Gamma(\alpha_d)}(\sum_{d=1}^{D}p_d)^{\alpha_d-\alpha_0}(1-\sum_{d=1}^{D}p_d)^{\beta-1} \tag{17}$$

This is called the Beta-Liouville distribution having $D+2$ parameters. Using Eq. (9), Eq.(10) and Eq.(11), we get the mean, variance and covariance of the Beta-Liouville distribution.

$$E[X_d] = \frac{\alpha}{\alpha+\beta}\frac{\alpha_d}{\alpha_0} \tag{18}$$

$$Var(X_d) = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}\frac{\alpha_d(\alpha_d+1)}{\alpha_0(\alpha_0+1)} - \frac{\alpha^2}{(\alpha+\beta)^2}\frac{\alpha_d^2}{\alpha_0^2} \tag{19}$$

$$Cov(X_l,X_k) = \frac{\alpha_l\alpha_k}{\alpha_0}\Big[\frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)(\alpha_0+1)} - \frac{\alpha^2}{(\alpha+\beta)^2\alpha_0}\Big]\ ;\ l \neq k \tag{20}$$

2.1.4 Scaled Dirichlet Distribution

Scaled Dirichlet distribution is a generalization of Dirichlet distribution after applying perturbation operation. Similar to Dirichlet, it is a distribution over multinomials in a simplex with support [0, 1]. Given a D dimensional vector of scaled gamma random variables of $\mathbf{p}$ such that $p_d \sim Ga(\alpha_d, \beta_d)$, scaled Dirichlet distribution can be obtained by performing normalization operation which transforms $\mathbf{p}$ into a proportional vector. If the proportional vector is $\mathbf{p} \in \mathbb{C}(1)$ where $\mathbb{C}(n) = \mathbb{C}(1)$ and n = sum of the multinomials, then the joint probability density function of $\mathbf{p} = (p_1, p_2, \ldots, p_D)$ is defined by, [2,3,81,86].

$$p(P|\theta) = \frac{\Gamma(\alpha_+)}{\prod_{d=1}^{D}\Gamma(\alpha_d)}\frac{\prod_{d=1}^{D}\beta_d^{\alpha_d}p_d^{\alpha_d-1}}{\left(\sum_{d=1}^{D}\beta_d p_d\right)^{\alpha_+}} \tag{21}$$

Here, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_D)$ and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_D)$ are shape and scale parameter vectors, respectively. Since, elements of $\boldsymbol{\beta}$ vector sum to unity, the number of free parameters is $2D - 1$. It can be easily shown that Eq.(21) turns into Dirichlet distribution by setting a constant value for the relaxed $\boldsymbol{\beta}$ variables.

2.1.5 Shifted Scaled Dirichlet Distribution

Shifted Scaled Dirichlet (SSD) distribution is obtained after applying the powering operation to the proportion vector $\mathbf{p} \sim SD(\mathbf{p}|\boldsymbol{\alpha}, \boldsymbol{\beta})$. A new scale parameter τ is introduced that scales the translated density values of Scaled Dirichlet distribution. This distribution has $2D$ free parameters. Given, $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{R}_+^{\mathbb{D}}$ as shape and location parameters and $\tau \in \mathbb{R}_+$ as scale parameter, Shifted Scaled Dirichlet density function can be computed as follows,

$$p(P|\theta) = \frac{\Gamma(\alpha_+)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \frac{1}{\tau^{D-1}} \frac{\prod_{d=1}^{D} \beta_d^{-\frac{\alpha_d}{\tau}} p_d^{\frac{\alpha_d}{\tau}-1}}{\left(\sum_{d=1}^{D} \left(\frac{p_d}{\beta_d} \right)^{\frac{1}{\tau}} \right)^{\alpha_+}} \tag{22}$$

2.1.6 Parameter Estimation

The concentration parameter $\boldsymbol{\alpha}$ can be determined from the observed proportional data $D_{obs} = \{p_1, p_2, \ldots, p_D\}$. If the dataset contains N observation with D dimensions, then the joint probability function of the whole dataset will be-

$$\begin{aligned} p(D_{obs}|\boldsymbol{\alpha}) &= \prod_{i=1}^{N} p(P_i|\boldsymbol{\alpha}) \\ &= \prod_{i=1}^{N} \frac{\Gamma(\sum_d \alpha_d)}{\prod_d \Gamma \alpha_d} \prod_d p_{i,d}^{\alpha_d - 1} \end{aligned} \tag{23}$$

In order to maximize Eq.(23), we need to take the gradient and set it to zero. It is cumbersome to apply chain rule with the product terms in Eq.(23). Therefore, we take maximum likelihood estimation (MLE) approach. Since the distributions discussed above are from exponential family, taking the logarithm will turn it into a convex optimization problem [62] and thus a line search algorithm such as Newton-Raphson method or fixed point iteration method can be applied [79], [114], [106].

$$\log(p(D_{obs}|\boldsymbol{\alpha})) = N \log\Gamma \sum_d \alpha_d - N\sum_d \log\Gamma\alpha_d + N\sum_d (\alpha_d - 1)\log\bar{p_d} \tag{24}$$

The derivative for one α_d is,

$$g_d = N\,\psi(\sum_d \alpha_d) - N\,\psi(\alpha_d) + N\log\bar{p_d} \tag{25}$$

Here, $\psi(x) = \frac{d\log\Gamma(x)}{dx}$ is the digamma function. The gradient for the dataset is $D \times 1$ and can be written as follows,

$$\mathbf{g} = \nabla\log(p(D_{obs}|\boldsymbol{\alpha})) = N\begin{pmatrix} \psi(\sum_d \alpha_d) - \psi(\alpha_1) + \log\bar{p}_1 \\ \psi(\sum_d \alpha_d) - \psi(\alpha_2) + \log\bar{p}_2 \\ \vdots \\ \psi(\sum_d \alpha_d) - \psi(\alpha_D) + \log\bar{p}_D \end{pmatrix} \tag{26}$$

In exponential family of distribution, when the gradient is set to zero, the observed and sufficient statistics becomes equal and as since Dirichlet distribution is from the exponential family, it is possible to formulate an iterative equation and solve it as a fixed point iteration problem to determine the concentration parameters α (see [79] for details). For a vector, this can be expressed as follows-

$$\begin{aligned} \mathbb{E}[\log p_d] &= \psi(\alpha_d) - \psi(\sum_d \alpha_k) \\ \psi(\alpha_d^{new}) &= \psi(\sum_d \alpha_d^{old}) + \log\bar{p}_k \end{aligned} \tag{27}$$

Fixed point iteration method converges only when $|g| < 1$ and is linearly convergent meaning that decreasing error in each step is roughly proportional to previous step. In contrast, Newton-Raphson method has quadratic convergence rate and guarantees to converge given that the initial guess is close to final estimate. The Hessian of the log-likelihood function is,

$$\mathbf{H} = \nabla^2 \log(p(D_{obs}|\boldsymbol{\alpha})) = \begin{pmatrix} \frac{\partial l^2}{\partial {\alpha_1}^2} \frac{\partial l^2}{\partial \alpha_1 \alpha_2} \cdots \frac{\partial l^2}{\partial \alpha_1 \alpha_d} \\ \frac{\partial l^2}{\partial \alpha_2 \alpha_1} \frac{\partial l^2}{\partial {\alpha_2}^2} \cdots \frac{\partial l^2}{\partial \alpha_2 \alpha_d} \\ \vdots \ddots \\ \frac{\partial l^2}{\partial \alpha_d \alpha_1} \frac{\partial l^2}{\partial \alpha_d \alpha_2} \cdots \frac{\partial l^2}{\partial {\alpha_d}^2} \end{pmatrix} = \mathbf{B} + 1_{\mathrm{d}} 1_{\mathrm{d}}^{\mathrm{T}} \mathbf{b} \tag{28}$$

where, $\mathbf{B} \hat{=}$ diag: $\mathbb{R}^D \rightarrow \mathbb{R}^{DxD} : -N$ diag$(\psi^{'}(\alpha_1), \ldots, \psi^{'}(\alpha_D))$ and $\mathbf{b} = N\psi^{'}(\sum_d \alpha_d)$; $\psi^{'}(x) = \frac{\mathrm{d}\psi(x)}{\mathrm{dx}}$ is the trigamma function. For Newton-Raphson algorithm, the Hessian needs to be inverted and [78] provided the following inversion technique using Sherman-Liberman formula-

$$\mathbf{H}^{-1} = \mathbf{B}^{-1} - \frac{\mathbf{B}^{-1} 1_{\mathrm{D}} 1_{\mathrm{D}}^{\mathbf{T}} \mathbf{B}^{-1}}{\mathbf{b}^{-1} + 1_{\mathrm{D}}^{\mathrm{T}} \mathbf{B}^{-1} 1_{\mathrm{D}}} \tag{29}$$

Therefore, update for the Newton's algorithm becomes,

$$\boldsymbol{\alpha}^{new} = \boldsymbol{\alpha}^{old} - \mathbf{H}^{-1}\mathbf{g} \tag{30}$$

As discussed, it is important to estimate the initial values of the parameters more accurately rather than taking random initial guess so that Eq.(30) converges to global optima. There are some propositions for the initial estimation of these parameters. Method of moments technique provides good estimate of the initial guess of the parameters. The first and second moments of the data can be calculated from the moment generating function. The moment generating function of a vector X of random variable x is given by $\mathbb{E}(\mathrm{e}^{tX})$ and is defined by $\mathbb{M}_X(t)$.

$$\begin{aligned} \mathbb{M}_X(t) &= \mathbb{E}(\mathrm{e}^{tX}) \\ &= \int_{-\infty}^{\infty} \mathrm{e}^{tx} f(x) dx \end{aligned}$$

With the utilization of Taylor series expansion solving the above equation for Dirichlet distribution results in the first and second moments can be presented as follows-

$$
\begin{aligned}
\mathbb{E}(X) &= \frac{\alpha_d}{\sum_d \alpha_d} \\
\mathbb{E}(X^2) &= \frac{\alpha_d(\alpha_d + 1)}{\sum_d \alpha_d(\sum_d \alpha_d + 1)}
\end{aligned}
$$

Solving the above equations, we get the values of the parameters $\boldsymbol{\alpha}$ which can be used as an initial guess for the Newton's algorithm.

$$\alpha_d = \mathbb{E}[p_d]\frac{\mathbb{E}[p_d] - \mathbb{E}[p_d^2]}{\mathbb{E}[p_d^2] - \mathbb{E}[p_d]^2}$$

2.2 Support Vector Machine

SVM is a well known and common choice for supervised machine learning. Empirically it has shown good generalization performance in different fields of research and applications [108], [70], [18]. The aim of using this classifier is to find the support vectors that maximize the margin between class labels where number of support vectors is proportional to generalization error [109]. Considering the primal representation of the optimization problem, we have

$$\min_{w,b,\xi} \ \frac{1}{2}||w||^2 + C\sum_{i}^{N}\xi_i \tag{31}$$

$$\text{subject to, } y^{(i)}(w^{\mathrm{T}}\phi(p_i) + b) \geq 1 - \xi_i, \ \ i = 1, \ldots, N \tag{32}$$

$$\xi_i \geq 0, \ \ i = 1, \ldots, N \tag{33}$$

Assume the dataset $D = \{(p_i, y_i)\}_i^N$ where N is the number of images and each image is represented by a *L1*-normalized histogram, p_i and the corresponding label y_i. The objective is to determine the infinite number of linear classifiers that maximizes the geometric margin between the classes with the lowest generalization error. In case of non-seperable data, we look into higher dimensional space to find the appropriate hyperplane that maximizes the geometric margin and minimizes the misclassification error through some feature mapping technique. To control the trade off between the large margin and error rate, the hyperparameter C is incorporated.

The above is a convex quadratic optimization problem with linear constraints. Solving this problem will result in the maximum geometric margin between classes.

Here, $\phi(p_i)$ is the embedding or feature mapping function from the input space, χ to the feature space, $\mathcal{H}$. If no extra features are extracted from the data then this function represents the original input data known as the attributes and the kernel, K which is the inner product between two datapoints become $\langle p_i, p_j \rangle$ instead of $\langle \phi(p_i), \phi(p_j) \rangle$. For non-linearly separated data, slack variables ξ_i are introduced in the objective function and the constraints are modified accordingly. C is a hyper-parameter that regularizes our objective function for misclassification. $\sum_i^N \xi_i$ is the upper bound of the generalization error. In hard margin classifier, higher values of hyperparameter C allows lower the misclassification error and in soft margin classifier, C is set to low values to provide flexibility at boundary region for some datapoints to be miss-classified.

Solving the dual problem is computationally convenient for large datasets. Relaxing the constraints with the help of Lagrange multipliers, dual solution becomes,

$$\begin{aligned} &\underset{\gamma}{\text{maximize}} \sum_i^N \gamma_i - \frac{1}{2} \sum_i^N \sum_j^N \gamma_i \gamma_j y^{(i)} y^{(j)} \langle \phi(p_i), \phi(p_j) \rangle \\ &\text{subject to:} \quad 0 \leq \gamma_i \leq C, \quad \sum_i \gamma_i y^{(i)} = 0 \quad \text{where} \quad i = 1, \dots N \qquad \forall \ \alpha_i, y^{(i)} \end{aligned} \tag{34}$$

Only the support vectors have $\boldsymbol{\gamma}$ values elsewhere it is zero. Getting the support vectors, the decision function classifies the data by comparing the kernel with the support vectors. The decision function of the support vector machine becomes,

$$f(p) = \sum_i^n \gamma_i y^{(i)} \langle \phi(p_i), \phi(p) \rangle \tag{35}$$

2.3 Feature Mapping: Dirichlet SVM, Generalized Dirichlet SVM, Beta-Liouville SVM

In this section, we focus on the primal and dual forms of the optimization problem in Eq(33) and Eq.(82) to modify the feature mapping function $\phi(p)$. As discussed, optimum performance of SVM depends on the choice of the kernel function and there is no structured procedure to select the kernel function or feature mapping function [14]. One of the advantages of embedding input vectors into the feature space is providing flexibility in choosing the mapping function $\phi(p)$ depending on the structure of the data. Taking the advantage of Dirichlet, generalized Dirichlet, Beta-Liouville, scaled Dirichlet and shifted scaled Dirichlet distributions for proportional data modelling, a new feature map can be constructed as follows,

$$\phi_j(p_i) = \begin{cases} p_{ij} & j = 1, \dots, D \\ \frac{\Gamma\left(\sum_d \alpha_d\right)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \prod_{d=1}^{D} p_{id}^{\alpha_d - 1} & j = D+1 \\ \prod_{d=1}^{D} \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} p_{id}^{\alpha_d - 1} \left(1 - \sum_{l=1}^{d} p_{il}\right)^{\gamma_d} & j = D+1 \\ \frac{\Gamma(\alpha_0)}{B(\alpha,\beta)} \prod_{d=1}^{D} \frac{p_{id}^{\alpha_d - 1}}{\Gamma(\alpha_d)} (\sum_{d=1}^{D} p_{id})^{\alpha_d - \alpha_0} (1 - \sum_{d=1}^{D} p_{id})^{\beta - 1} & j = D+1 \\ \frac{\Gamma(\alpha_+)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \frac{\prod_{d=1}^{D} \beta_d^{\alpha_d} p_d^{\alpha_d - 1}}{\left(\sum_{d=1}^{D} \beta_d p_d\right)^{\alpha_+}} & j = D+1 \\ \frac{\Gamma(\alpha_+)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \frac{1}{\tau^{D-1}} \frac{\prod_{d=1}^{D} \beta_d^{-\frac{\alpha_d}{\tau}} p_d^{\frac{\alpha_d}{\tau} - 1}}{\left(\sum_{d=1}^{D} \left(\frac{p_d}{\beta_d}\right)^{\frac{1}{\tau}}\right)^{\alpha_+}} & j = D+1 \end{cases} \tag{36}$$

To estimate the parameters in Eq.(36), a similar technique is followed as described by [82]. Using the kernel trick, the proposed feature mapping technique can be used with the traditional non-linear kernels to map input space into feature space implicitly without knowing about the feature space. The dimension of the input space is increased by 1 by doing the feature mapping mentioned in Eq.(36). We can formulate the Dirichlet SVM (DSVM) as follows,

$$\min_{w,b,\xi} \ \frac{1}{2}\sum_{d=1}^{D+1} w_d^2 + C\sum_{d=1}^{D+1}\xi_i \tag{37}$$

$$\text{subject to, } y^{(i)}\Big(\sum_{d=1}^{D} w_d p_{id} + w_{D+1}\frac{\Gamma\left(\sum_d \alpha_d\right)}{\prod_{d=1}^{D}\Gamma(\alpha_d)}\prod_{d=1}^{D} p_i d^{\alpha_d - 1} + b\Big) \geq 1 - \xi_i, \ \ i = 1, \dots, n \tag{38}$$

$$p_{iD} = 1 - \sum_{d=1}^{D-1} p_d \tag{39}$$

$$\xi_i \geq 0, \ \ i = 1, \dots, n \tag{40}$$

In a similar fashion, generalized Dirichlet SVM (GDSVM) and Beta-Liouville SVM (BLSVM) can be formulated. For a new data p', the trained Dirichlet parameter α is used to determine the feature mapping according to Eq.(36). The decision function for this new data becomes,

$$f(p') = \sum_{i}^{N}\Big(\gamma_i \sum_{d=1}^{D+1} p_{id}p_d^{'}\Big) \tag{41}$$

Applying the flexible mapping function $\phi(p)$ in Eq.(36) changes the similarity measure and thus enables us to modify the base kernel. Apart from the regular kernels such as RBF, polynomial, sigmoid, χ^2 which are discussed vastly in the literature, we combine our proposed feature mapping technique with other kernels as well.

- **Linear**
  Linear kernel is the simplest kernel which takes the dot products of the features to capture the similarity amongst them.

$$K(\mathbf{x}, \mathbf{y}) = \langle x, y \rangle \tag{42}$$

- **RBF-unnormalized Gaussian**
  Radial basis function characterizes the features by considering the distance from the center irrespective of the direction. Taking the Euclidean distance and take scaling that by the hyperparameter γ results in unnormlized Gaussian kernel. γ controls the width of the distance. [39] presented the following generalized

representation of the RBF kernels

$$K(\mathbf{x}, \mathbf{y}) = e^{-\gamma d(x,y)} \tag{43}$$

Considering the Euclidean distance results in the Gaussian kernel.

$$K_{i,j} = e^{-\gamma ||x_i - x_j||^2} \tag{44}$$

- **Exponential**

  $\gamma = \frac{1}{2\sigma^2}$ results in exponential kernel. Nomalizing the RBF-unnormalized kernel with the feature variance σ gives the the Gaussian kernel also known as the exponential kernel. The similarity decreases if the the parameter value is too large.

  $$K_{i,j} = e^{\dfrac{-||x_i - x_j||}{\sqrt{2}\sigma}} \tag{45}$$

  Doing the Taylor series expansion, this kernel presents an infinite dimensional feature space.

- **Polynomial**

  This is a popular kernel for non-linear data modelling. The basic idea is to take the dot product of two vectors to higher dimension d. It is preferable to add an additional shifting parameter c so that the Hessian does not become zero [55].

  $$K_{i,j} = (\langle x_i, x_j \rangle + c)^d \tag{46}$$

- **Bhattacharya Measure**

  Bhattacharya coefficient is a divergence type measure between distributions and defined as [43],

  $$B = \sum_{i=1}^{N} \sqrt{p_i q_i} \tag{47}$$

  Considering a $D + 1$ dimensional vector, it can be geometrically interpreted that the Bhattacharya coefficient measures the cosine of the angle between the vector elements. Since, p_i and q_i represent probability distributions and if they have the similar density function then the coefficient is 1. However, this coefficient can not be used as a metric distance since it violates the axioms of being a distance metric [52]. To make a proper representation of the distance metric, [43] modified the coefficient as $D_{p_i,q_i} = \sqrt{1-B}$. The kernel for this distance with hyperparameter γ,

$$K(\mathbf{p}, \mathbf{q}) = e^{-\gamma\sqrt{1-B}} \tag{48}$$

- **Generalized Histogram Intersection**

  Histogram intersection kernel is a positive definite kernel and satisfies Mercer's condition to be used in SVM [8], [19]. Global or low-level features are commonly used for this, however, use of local features works well with this kernel as well. Given two vectors namely p_i and p_j containing the elements of two normalized histograms, histogram intersection measures the similarity between the them by using Eq.(49) [76].

$$K(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{N} min[(p_i)^\alpha, (q_i)^\alpha] \tag{49}$$

  Setting $\alpha = 1$ results in histogram intersection kernel.

- **Jeffrey Divergence**

  KL-divergence is non-symmetric and sensitive to histogram binning [101]. In addition, it is not robust and does not qualify to be used as a metric of the spread since it violates the triangle inequality. In response to this, Jeffrey divergence is empirically derived and it is mostly invariant to noise and histogram binning [61].

$$K(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^{N} (p_i \log \frac{p_i}{\mu_i} + q_i \log \frac{q_i}{\mu_i});\ \mu_i = \frac{p_i + q_i}{2} \tag{50}$$

- **Rational Quadratic**

  From the probabilistic graphical point of view, several squared error kernels are derived and rational quadratic is one of them. This kernel is a scale mixture of different characteristic length scales [97]. This kernel is useful for modelling data which varies in multiple scales.

$$K(\mathbf{p},\mathbf{q}) = \Big(1 + \frac{\sum_i^N ||p_i - q_i||^2}{2\alpha l^2}\Big)^{-\alpha} \tag{51}$$

Here, α is scale mixture parameter and l is the scale length.

- **Inverse Multiquadratic**

  Inverse multiquadratic function is a member of generalized multiquadratic (GMQ) family of radial basis functions defined by $K(\mathbf{p},\mathbf{q}) = (c^2 + (\epsilon r)^2)^\beta$ [80] where ϵ is the shape parameter and parameter β determines the positive definiteness of the function [41]. Unlike multiquadratic kernel, inverse multiquadratic is a positive definite [64]. Setting $\beta = \frac{1}{2}$, we get the following expression for this kernel-

  $$K(\mathbf{p},\mathbf{q}) = \frac{1}{\sqrt{\sum_i^N |p_i - q_i|^2 + c^2}} \tag{52}$$

- **ANOVA**

  ANOVA kernel is one of the examples of convolution kernels [111]. This kernel uses factor d to get higher order interactions of the features that we are interested in and then sum over the terms to get the similarity score.

  $$K(\mathbf{p},\mathbf{q}) = \sum_i^N e^{-(\sigma(p_i - q_i)^2)^d} \tag{53}$$

- **Generalized T-student Kernel**

  This is a positive semi definite kernel and satisfies the condition of Mercer's theorem [20]. It has similar form to Inverse Multiquadratic kernel.

  $$K(\mathbf{p},\mathbf{q}) = \sum_i^N \frac{1}{1 + (p_i - q_i)^d} \tag{54}$$

- **MinMax**

  MinMax is a graph kernel proposed by [96] which is similar to Tanimoto kernel when applied to binary dataset. MinMax kernel models count data and thus takes into account the values between 0 and 1. Therefore, this kernel is suitable for proportional data modelling.

$$K(\mathbf{p}, \mathbf{q}) = \frac{\sum_{i}^{N} \min(p_i, q_i)}{\sum_{i}^{N} \max(p_i, q_i)} \tag{55}$$

- **Cauchy**

  Derived from the long tail Cauchy distribution, Cauchy kernel puts more weight on interaction of distant non-zero values [9]. [87] applied the Cauchy kernel for sparse coding of natural scenes data.

  $$K(\mathbf{p}, \mathbf{q}) = \sum_{i}^{N} \frac{1}{1 + \frac{(p_i - q_i)^2}{s^2}} \tag{56}$$

  Unlike Gaussian kernel, in this kernel moving from the center gives more weight to the features. A combination of these two kernels showed good classification performance on some datasets [9].

- **Cosine Similarity**

  In an inner product space, cosine similarity measures the similarity between two vectors by calculating the direction of each vector [57]. This is a non-metric measure since it does not satisfy all the conditions to be a metric.

  $$K(\mathbf{p}, \mathbf{q}) = \frac{\langle p_i, q_i \rangle}{||p_i||||q_i||} \tag{57}$$

- **Tanimoto or Extended Jaccard Similarity**

  A modification in the cosine similarity function results in Tanimoto similarity index [96]. It represents the number of attributes shared by the vectors.

  $$K(\mathbf{p}, \mathbf{q}) = \frac{\langle p_i, q_i \rangle}{\langle p_i, p_i \rangle + \langle q_i, q_i \rangle - \langle p_i, q_i \rangle} \tag{58}$$

  Here, $\langle p_i, q_i \rangle = \sum_{d=1}^{D} p_{id} \times q_{id}$ and the term $\langle p_i, p_i \rangle = ||p_i||^2$ and $\langle q_i, q_i \rangle = ||q_i||^2$ is the Euclidean norm or the length of the vector. [4] derived the modified

Tanimoto coefficient in relation with Cosine similarity as,

$$K(\mathbf{p},\mathbf{q}) = \frac{\text{cossim}(p_i, q_j)}{\frac{||p_i||^2+||q_i||^2}{||p_i||||q_i||} - \text{cossim}(p_i, q_i)} \tag{59}$$

Here, $\text{cossim}(x_i, y_j)$ is calculated from Eq.(57).

- **Sorensen Similarity**

  Similar to cosine similarity Sorensen similarity index is a non-metric measure as it does not satisfy all the axioms of being a metric. This measure is more appropriate in retaining the sensitivity of the heterogenous data than Euclidean distance and in image segmentation and lexical association [50], [102] .

$$K(\mathbf{p},\mathbf{q}) = \sum_{i}^{N} \frac{2p_iq_i}{p_i^2 + q_i^2} \tag{60}$$

Algorithm 1 shows the steps for the Dirichlet SVM, generalized Dirichlet SVM and Beta Liouville SVM using Eq.(36).

```
Algorithm 1: Algorithm for DSVM, GDSVM and BLSVM
1. Input: Training data, $D = \{(p_1, y_1), (p_2, y_2), \ldots, (p_N, y_N)\}$.
2. Estimate: Initial parameters using Method of Moments (MoM) [82].
3. Update: Apply Newton Raphson's method until convergence [82].
4. Compute kernel:
    • Base kernel: Compute K($\mathbf{p}$, $\mathbf{q}$) froms Eq.(48) to Eq.(60) for $\phi_j(p_i)$ in Eq.(36) only when $j = 1, 2, \ldots, D$.
    • DSVM: Use first and second forms of Eq.(36) for $\phi_j(p_i)$ and apply Eq.(48) to Eq.(60) to compute DSVM kernel, K($\mathbf{p}$, $\mathbf{q}$).
    • GDSVM: Use first and third forms of Eq.(36) for $\phi_j(p_i)$ and apply Eq.(48) to Eq.(60) to compute GDSVM kernel, K($\mathbf{p}$, $\mathbf{q}$).
    • BLSVM: Use first and fourth forms of Eq.(36) for $\phi_j(p_i)$ and apply Eq.(48) to Eq.(60) to compute BLSVM kernel, K($\mathbf{p}$, $\mathbf{q}$).
    • SDSVM: Use first and fifth forms of Eq.(36) for $\phi_j(p_i)$ and apply Eq.(48) to Eq.(60) to compute SDSVM kernel, K($\mathbf{p}$, $\mathbf{q}$).
    • SSDSVM: Use first and sixth forms of Eq.(36) for $\phi_j(p_i)$ and apply Eq.(48) to Eq.(60) to compute SSDSVM kernel, K($\mathbf{p}$, $\mathbf{q}$).
5. Optimization: Solve the primal problem in Eq.(33) or dual problem in Eq.(82) to get the support vectors.
```

2.3.1 Feature Encoding

In several computer vision applications, Bag of Visual Words (BoVW) approach is considered to represent image features. In this framework, features such as SIFT [73] or HOG [47] of each images are extracted in the prepossessing stage. The collection of local features known as the descriptors are clustered into a specified bin size using K-means clustering algorithm so that similar patches are grouped together. This is known as quantization and after this each cluster center represents codeword [40].

The feature distribution of each image over the vocabulary, V is computed by,

$$f_{hist} = \frac{1}{N}\sum_{j=1}^{N}\begin{cases}1 & \text{if s= } \underset{w\in V}{\operatorname{argmin}}(||w - r_j||_2^2)\\ 0 & \text{Otherwise}\end{cases} \tag{61}$$

Here, w is the codebook or cluster center and r_j is interest region extracted by SIFT descriptors. Therefore, each image is represented by visual word frequencies of codewords in the vocabulary. Satisfying the non-negative constraint and unit sum, this representation can be modelled with our proposed method as described in Eq.(84).

2.4 Experimental Results: DSVM, GDSVM and BLSVM

In this section, we evaluate the proposed feature mapping technique for natural scene classification, satellite image classification and human action recognition in videos for DSVM, GDSVM and BLSVM. The dual form of the SVM optimization problem is solved using sklearn API [88]. For multiclass classification, one-vs-all technique is applied and the tolerance value 10^{-3} is used as stopping criterion and a hard limit on the solver is imposed by setting maximum iterations to 5000. All the models are evaluated using 10 fold cross validation. 9 folds are used for training and the remaining fold for testing the model. Similar to [82], for image classification best score is reported for each kernel and for action recognition, average scores with standard deviations are reported for all kernels. For misclassification, the hyperparameter C in the objective function is varied from 1 to 15 in 10 base logarithm scale and best models are found by doing a simple grid search and are reported thereby. For polynomial kernel, degree 3 is considered and for RBF kernel, the similarity measurements are scaled down by dividing the length of vocabulary size.

2.4.1 15 scenes dataset classification

Scene recognition is very essential for reasoning in navigation and recognition tasks. Specially in terms of robotics and automation it is significant to enhance machine's visual understandings [117]. 15 scene dataset consists of 15 different scene categories.

First 13 categories were collected combinedly by [49] and [68]. For our experiment, from each category 100 images were selected totalling to 1500 images. Local features are extracted using Scale Invariant Feature Transform (SIFT) [73] algorithm as it is invariant to scale and rotation. In our experiment, we calculate dense SIFT [49] for speed using [110] . Descriptors are computed for densely sampled keypoints with similar size and orientation. Each image is converted to grayscale and for each pixel descriptors are computed over a patch of 16 × 16 pixels. The extracted features are quantized into a vocabulary size of 200. Table 7 shows the best results for the baseline SVM, DSVM, GDSVM and BLSVM.

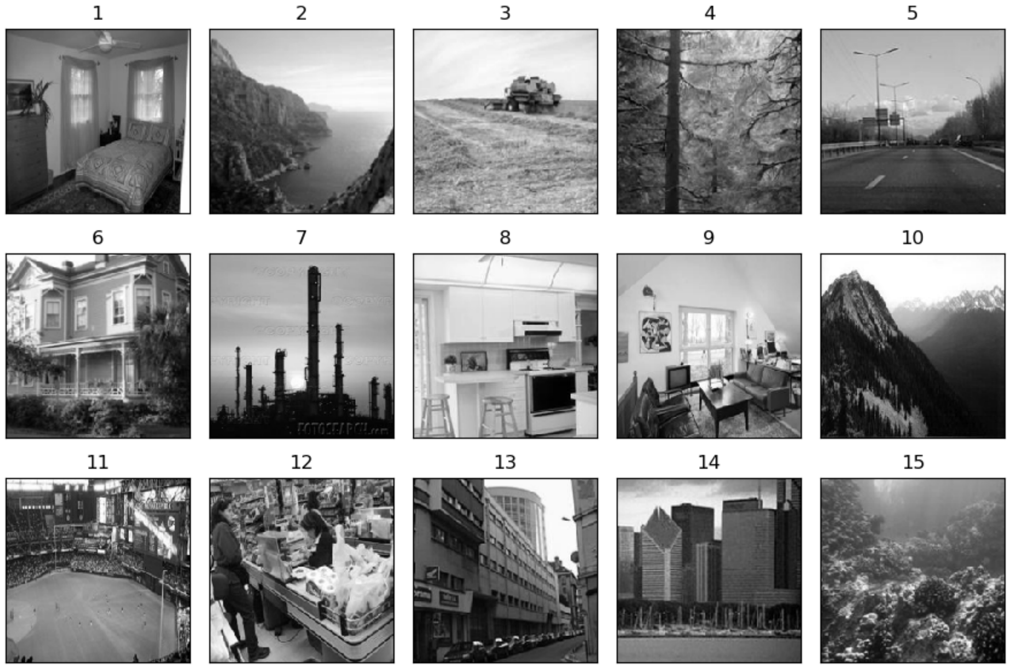


Figure 2: Sample image from 15 different categories: 1. bedroom, 2. sea coast, 3. field, 4. forest, 5. highways, 6. house, 7. industrial, 8. kitchen, 9. living room, 10. mountain, 11. stadium, 12. store, 13. street, 14. sky scrappers, 15. ocean underwater

Kernel	Baseline SVM	DSVM	GDSVM	BLSVM
Linear	0.72000	0.72000	0.70667	**0.74000**
Polynomial	0.76000	**0.77333**	0.76000	0.74677
Sigmoid	0.70677	0.72000	0.72000	**0.73333**
RBF	0.70677	0.72000	0.71333	**0.74677**
Exponential	0.74667	0.74667	0.74667	**0.79333**
Tanimoto	0.74000	0.74000	0.74000	**0.76000**
MinMax	0.76667	0.76000	0.76000	0.76667
Bhattacharya	0.74000	0.74667	0.73333	**0.76000**
Cosine Similarity	0.72667	0.71333	0.71333	**0.73333**
Rational Quadratic	0.75333	**0.76000**	**0.76000**	0.72667
Inverse Multiquadratic	0.77333	0.77333	**0.78000**	0.74000
Cauchy	0.71333	0.72000	0.72000	**0.75333**
Tstudent	0.75333	**0.76000**	**0.76000**	0.72667
ANOVA	0.72667	0.71333	0.72000	**0.74667**
Sorensen Similarity	0.72667	0.72667	0.72667	0.71333
Additive χ^2	0.76667	**0.77333**	0.76667	0.76000
Histogram Intersection	0.76000	0.76000	0.76000	0.74667

Table 1: Accuracy score of natural scene recognition for baseline kernels and our proposed kernels

In core form, BLSVM shows the best accuracy of 74.00% compared to the baseline SVM which is 72.00%. Combining with other kernels, either of the three proposed SVM shows better accuracy than corresponding baseline kernel accuracy. The enhanced performance is made possible due to the flexible feature mapping technique discussed in section 2.3. Combining exponential kernel with Beta-Liouville SVM gives the highest accuracy of 79.33% whereas its baseline, DSVM and GDSVM accuracy is 74.67%.

2.4.2 Satellite Image Classification

This dataset has 19 categories of google satellie images collected from `http://www.escience.cn/people/yangwen/WHU-RS19.html`. Each category has 50 images and the resolution of each image is 600×600. The challenges in classifying high resolution satellite image data is that the dominance of structures and objects in the image leads to misclassification [46]. For feature extraction, we use the same configuration as described in previous section.

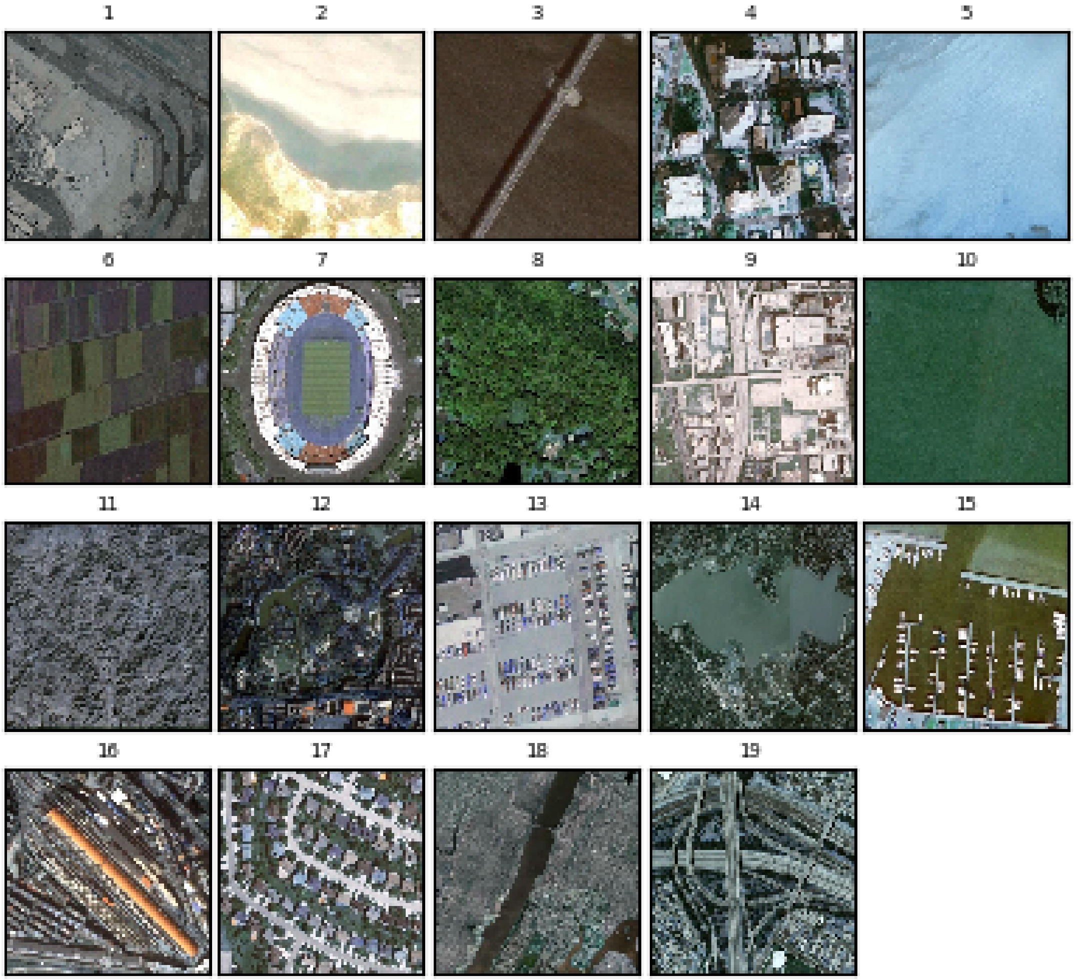


Figure 3: Sample satellite image from 19 different categories: 1. airport, 2. sea beach, 3. bridge, 4. commercial area, 5. desert, 6. farmland, 7. stadium, 8. forest, 9. industrial area, 10. meadow, 11. mountain, 12. park, 13. parking, 14. pond, 15. port, 16. railway station, 17. residential area, 18. river, 19. viaduct

Kernel	Baseline SVM	DSVM	GDSVM	BLSVM
Linear	0.86364	0.85577	0.87000	**0.90196**
Polynomial	0.85454	0.86486	0.85454	**0.89189**
Sigmoid	0.86274	0.86274	0.86364	**0.89215**
RBF	0.87272	0.87272	0.87272	**0.89215**
Exponential	0.88073	0.88073	**0.88991**	0.88182
Tanimoto	0.90566	0.90566	0.90566	**0.90999**
MinMax	0.88462	0.89423	0.88462	**0.90197**
Sorensen Similarity	0.87273	0.86363	0.88182	**0.89216**
Bhattacharya	0.90196	0.89215	0.90196	**0.88991**
Cosine Similarity	0.86363	0.87000	0.87273	**0.90196**
Rational Quadratic	0.88181	0.87272	0.86363	**0.88235**
Inverse Multiquadratic	0.88000	0.88000	0.88000	**0.88235**
Cauchy	0.88000	0.89000	0.89000	**0.89215**
Tstudent	0.86000	0.87000	0.86000	**0.88235**
ANOVA	0.85294	0.85000	0.86000	**0.87129**
Additive χ^2	0.89215	0.89215	0.89215	0.89215
Histogram Intersection	0.90384	0.90384	0.89423	**0.91176**
Jfd	0.89215	0.89215	0.88679	**0.90196**

Table 2: Accuracy score of satellite image classification for baseline kernels and our proposed kernels

For all the kernel, BLSVM outperforms baseline SVM, DSVM and GDSVM except for the exponential kernel where generalized Dirichlet SVM achieves higher accuracy of 88.991%(Table 2). Considering the core form SVM, BLSVM gives highest accuracy of 90.196% whereas linear SVM achieves 86.364% accuracy.

2.4.3 Human action recognition

Recognizing human action in videos is an interesting learning task for surveillance and navigation tasks. For the purpose of evaluation of our model for videos, we choose KTH-human action recognition data introduced by [67]. This dataset contains 6 categories each having 100 videos with 4 different scenarios and each action is performed by 25 different persons with different variations like different color of clothing, different motion of the person, camera angle, zooming, zittering, etc. In total, there are 2391 sequences in this dataset. We are interested in dense features as it is more accurate than sparse features, we use dense optical flow algorithm proposed by [48]. Open source computer vision library such as [34] is used with default values to extract features with the codebook size of 500. Each frame is resized to 160 × 120 and further downsampled to 16 × 12 by taking the pixel values of the positions which are divisible by 10. L_2 normalization is used for feature invariance. To create Dirichlet, generalized Dirichlet and Beta-Liouville SVM, the whole dataset is normalized as proposed in [82]. For 10 fold cross validation, mean accuracies with standard deviation are reported in Table 3. In total 384 videos are used for training and 216 videos are used for testing. In the test data, each class has 36 videos.

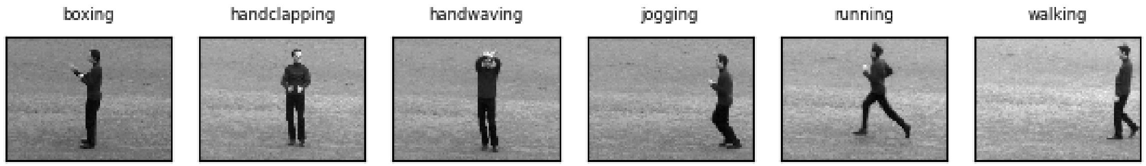


Figure 4: Sample frame from each categories performed by one person. Each frame is resized to 160 × 120.

Kernel	Baseline SVM	DSVM	GDSVM	BLSVM
Linear	0.90401 ± 0.047	0.89886 ± 0.048	0.90401 ± 0.047	**0.91167 ± 0.056**
Polynomial	0.82869± 0.045	0.92323± 0.042	0.84132± 0.045	**0.93185± 0.043**
Sigmoid	0.89171± 0.050	0.92046± 0.047	0.90045± 0.059	**0.93185 ± 0.049**
RBF	0.90197 ± 0.050	0.92319 ± 0.042	0.89449 ± 0.056	**0.93185 ± 0.043**
Exponential	0.91161 ± 0.054	0.91399 ± 0.051	0.90962 ± 0.050	**0.91677 ± 0.047**
Tanimoto	0.90923 ± 0.048	0.92040 ± 0.042	0.90923 ± 0.048	**0.92868 ± 0.046**
MinMax	0.92034± 0.051	**0.94104 ± 0.045**	0.93933 ± 0.059	0.93661 ± 0.031
Sorensen Similarity	0.89934 ± 0.064	0.92041 ± 0.042	0.90214 ± 0.064	**0.92590 ± 0.045**
Bhattacharya	0.90634± 0.040	0.90634± 0.040	0.90395± 0.044	**0.92403± 0.044**
Cosine Similarity	0.88701± 0.064	**0.89528± 0.053**	0.88939± 0.063	0.89296± 0.049
Rational Quadratic	0.89897± 0.066	0.92046± 0.047	0.89858± 0.061	**0.92629± 0.040**
Inverse Multiquadratic	0.90969± 0.044	0.92046± 0.047	0.90969± 0.044	**0.92392± 0.047**
Cauchy	0.89212± 0.053	0.91008± 0.048	0.89212± 0.053	**0.91473± 0.056**
ANOVA	0.89767± 0.064	0.89767± 0.064	**0.90481± 0.062**	0.90124± 0.056
Additive χ^2	0.91989± 0.052	**0.92312± 0.051**	0.92267± 0.049	0.92205 ± 0.048
Histogram Intersection	0.92001 ± 0.065	**0.93826± 0.045**	0.92278± 0.061	0.93383 ± 0.031
Jfd	0.90963± 0.061	**0.92312± 0.051**	0.90481± 0.057	0.91689± 0.051

Table 3: 10 fold cross validation results (mean score with standard deviation) of action recognition from videos

From Table 3, highest average accuracy of baseline SVM is 92.034% for MinMax kernel which is increased to 94.104% when we combine MinMax kernel with Dirichlet feature mapping function.

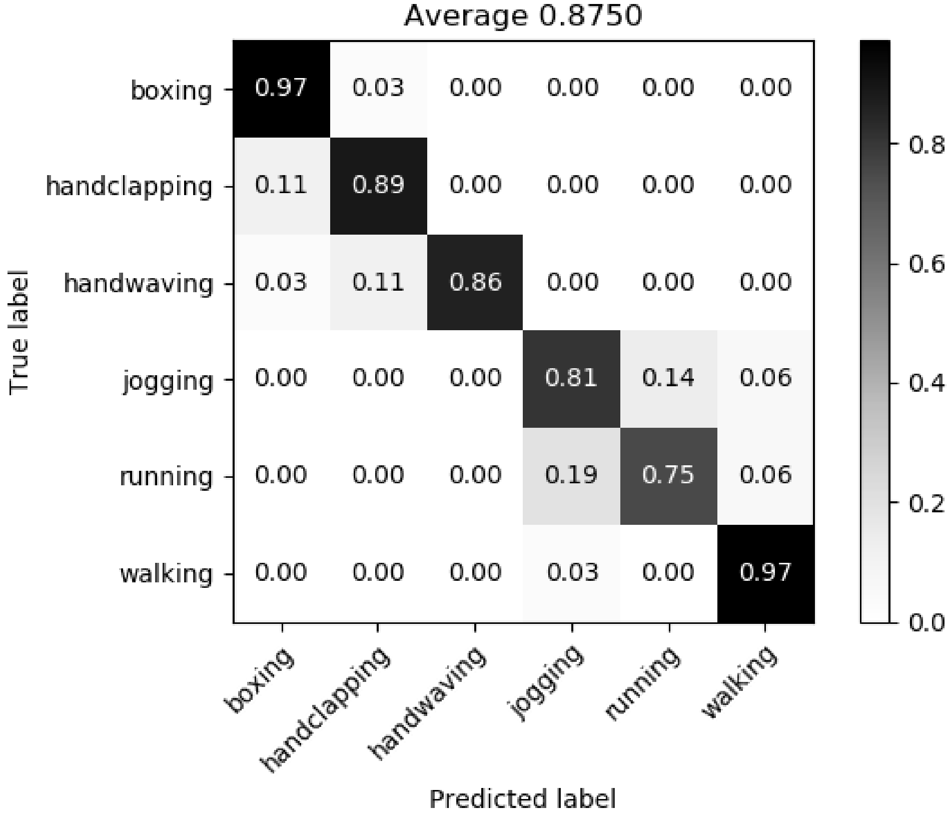


Figure 5: Confusion matrix for human action recognition in videos

Fig. 5 shows the confusion matrix for MinMax kernel with Dirichlet feature mapping SVM (DSVM) which achieves 87.50% accuracy for the test data compared to base MinMax kernel which achieves 86.11% accuracy for the test data.

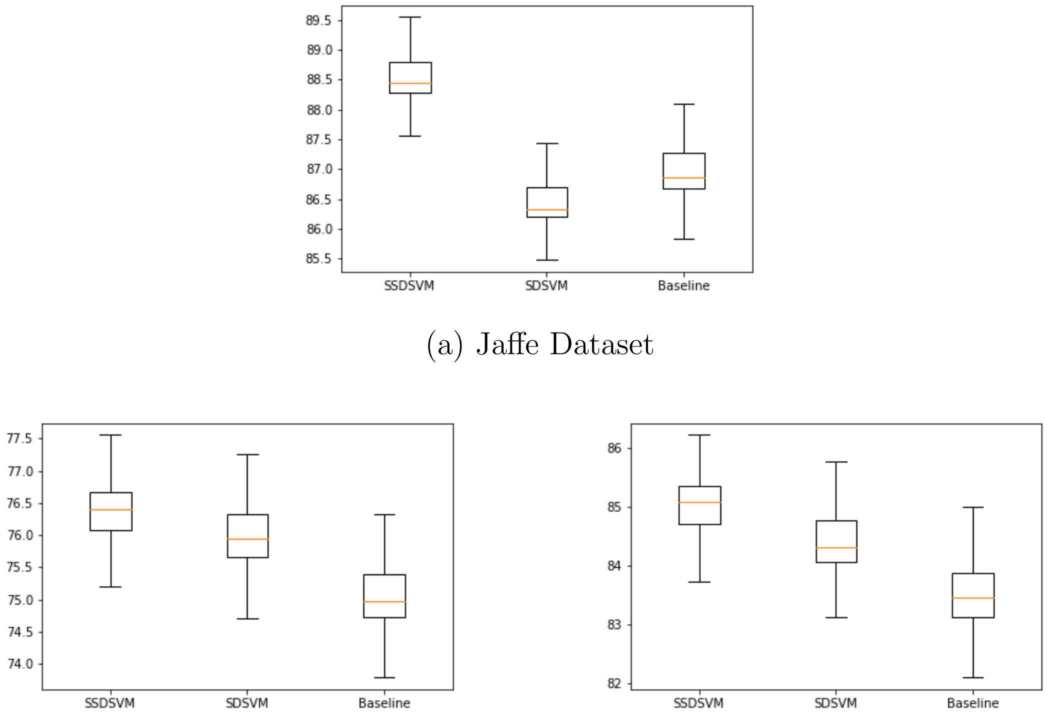
2.5 Experimental Results: SDSVM and SSDSVM

2.5.1 Dataset

In order to verify the effectiveness of proposed methods, we conduct a set of experiments regarding gender classification and emotion recognition from images of faces. For gender classification we choose, two different public datasets which are computationally challenging with different variations in the images. We have used Caltech face dataset [54], IMDB-Wiki [99] for gender classification and Jaffe emotion dataset [74] for emotion recognition. A fixed cluster size 256 is used for gender classification task. For fair evaluation, a fixed set of parameter values are used in all experiments. Degree and width of polynomial and RBF kernel are set to 3 and 1, respectively. Miss-classification penalty term C is set to $[0.0001, 0.01, 0.1, 1.0, 10, 100]$. All datasets are split into 80:20 ratio where 80% of a dataset are used for training and 20% for validation. For robustness, we perform 10-fold cross validation in all experiments and average accuracy score (except IMDB-Wiki dataset) same as [82] are reported in Table 4 and Table 5.

Dataset	Caltech Dataset			IMDB-Wiki Dataset		
Kernel	Baseline	SDSVM	SSDSVM	Baseline	SDSVM	SSDSVM
Linear	0.75542	0.76644	**0.77997**	0.83132	0.84189	**0.85098**
Polynomial	0.71975	0.74401	**0.74841**	0.80889	0.82403	**0.82833**
RBF	0.75759	**0.76199**	0.75750	0.85825	0.86653	**0.87356**
Sorensen Similarity	0.76901	0.76678	**0.77113**	0.83499	0.84189	**0.85207**
Log	0.74001	0.75987	**0.76462**	0.83132	0.84313	**0.84812**
Cauchy	0.75567	0.75779)	**0.76215**	0.84251	0.84086	**0.85098**

Table 4: Average accuracy score of gender classification for Caltech and IMDB-Wiki dataset



(a) Jaffe Dataset

(b) Caltech Dataset

(c) IMDB-Wiki Dataset

Figure 6: Confidence interval with mode of proposed methods

2.5.2 Gender Classification

Caltech Dataset:

This dataset includes 450 face images of males and females. There are 27 different subjects. There are 277 male and 173 female images in this dataset [100]. Each image has 896 × 592 pixels resolution. Classifying this dataset is challenging since there is variation in lighting, face expressions and change in background. SIFT [73] descriptors are extracted from each images and quantized into 256 bins. Next, we take the proportion of each features by normalizing the data using the same technique as [82]. Kernels mentioned in Table 4 improves the baseline kernels accuracy by 1% $\sim$ 2%. Linear kernel for SSDSVM achieves the highest accuracy of 77.99% whereas baseline linear kernel accuracy is about 75.54%.

IMDB-Wiki Dataset:

We choose a total of 912 images for training and validation of which 457 are females and 455 are males. This dataset is challenging since it contains images of different persons with different age labels with random poses. Similar to [16], we applied Linear Discriminant Analysis (LDA) technique to reduce feature dimension to 128 from 256. After cross-validation, the best models are tested on a test dataset. For testing 500 images are chosen of which 250 are males and 250 are females. Parameters learned from the training data are used to extract features for test data. SSDSVM outperforms all baseline and SDSVM kernels as reported in Table 4. Best result is achieved as 87.36% for RBF kernel with shifted scaled Dirichlet feature mapping.

2.5.3 Facial Expression Recognition

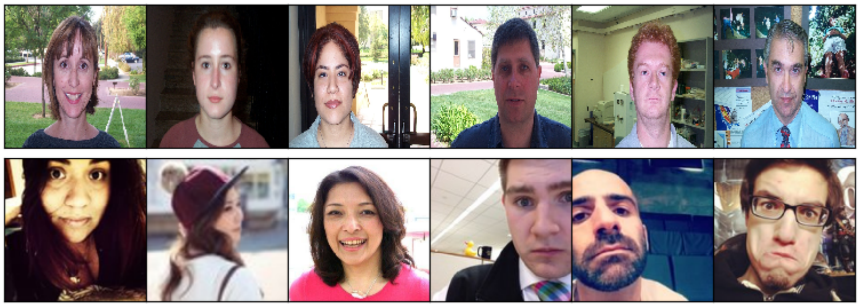


Figure 7: Sample images for gender classification. Top row shows images from Caltech dataset and bottom row shows images from IMDB-Wiki dataset.

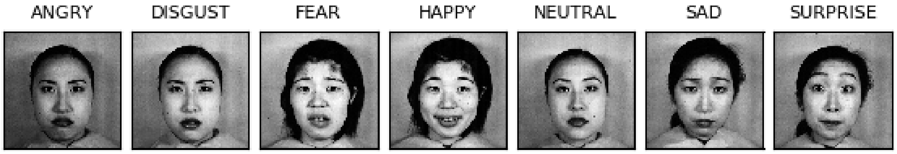


Figure 8: Sample images Jaffe emotion recognition dataset with 7 different classes.

For facial expression recognition, we choose Jaffe emotion recognition dataset. This is a laboratory controlled dataset and it contains 213 images of 10 Japanese females with 7 different facial expressions such as angry, disgust, fear, happy, neutral, sad and surprise. For emotion recognition, curated features such as Discrete Wavelength

Transform [120], [104], wavelet based features [90], Gaussian features [65] are commonly used. To show the effectiveness of our propose method in general, we consider pixel values after resizing each images to 60x60 which contains noise and feature overlapping. 8 different kernel functions are tested on this dataset.

Kernel	Baseline	SDSVM	SSDSVM
Linear	0.87713	0.82681	**0.88110**
Polynomial	0.85901	0.87260	**0.88331**
Cosine	0.86427	0.86784	**0.88728**
MinMax	0.87713	**0.87974**	0.86886
Tanimoto	0.87189	0.86196	**0.89331**
Sorensen Similarity	0.86059	0.86784	**0.88110**
Cauchy	0.86773	0.86784	**0.88776**
Additive χ^2	0.87252	0.86159	**0.89332**

Table 5: Accuracy score of Jaffe emotion recognition dataset.

As the results shown in Table 2, SSDSVM outperforms baseline and SDSVM except for min-max kernel in which SDSVM achieves better accuracy. SSDSVM with additive χ^2 kernel achieves highest accuracy of 89.332%. Both SDSVM and SSDSVM with polynomial kernels outperform baseline score. For kernels such as, linear tanimoto and additive χ^2, baseline score is better than scaled Dirichlet feature mapping. Hence, average accuracy for SDSVM is lower than the baseline as shown in Fig.6(a). However, for both datasets in gender classification, SDSVM and SSDSVM overall average accuracy is higher than baseline score in Fig.6(b) and Fig.6(c).

Chapter 3

Inverted Dirichlet and related distributions based feature for data classification

In this chapter, we extend the idea of feature mapping function in SVM classifier for inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions. These distributions are relaxed from unit sum constraint and can model semi bounded positive vectors [12], [11], [15].

3.1 Distributions for Positive Vectors

3.1.1 Inverted Dirichlet distribution

Inverted Dirichlet distribution is a generalization of multivariate Beta-Prime distribution. Let $\mathbf{X} = (X_1, X_2 \dots X_N)$, a collection of images with labels $Y = (Y_1, Y_2 \dots Y_N)$. Each image is represented by a D dimensional positive vector, $\mathbf{p} = (p_1, p_2, \dots p_D)$ where $p \geq 0$. The inferred parameters from the data can be represented by $\Theta = \{\alpha\}$ where $\alpha = (\alpha_1, \alpha_2, \dots \alpha_{D+1})$ is a parameter vector of inverted Dirichlet distribution. Then, probability density function is defined as [13],

$$p(X_i|\alpha) = \frac{|\alpha_+|}{\prod_{d=1}^{D+1} \Gamma\alpha_d} \prod_{d=1}^{D} p_{id}^{\alpha_d - 1}(1 + \sum_{d=1}^{D} p_{id})^{-|\alpha_+|} \tag{62}$$

where, $p_{id} \geq 0$, $d = 1, 2, \ldots D$ and $\alpha_+ = \sum_{d=1}^{D+1} \alpha_d$ with $d = 1, 2, \ldots, D+1$. The mean, variance and covariance of inverted Dirichlet distribution are as follows,

$$E(p_d) = \frac{\alpha_d}{\alpha_{D+1} - 1} \tag{63}$$

$$Var(p_d) = \frac{\alpha_d(\alpha_d +_{D+1} -1)}{(\alpha_{D+1} - 1)^2(\alpha_{D+2} - 1)} \tag{64}$$

$$Cov(p_l, p_d) = \frac{\alpha_d \alpha_l}{(\alpha_{D+1} - 1)^2(\alpha_{D+2} - 1)} \tag{65}$$

To generate random positive vector $\mathbf{p} = (p_1, p_2, \ldots ,_D)$, a method has been proposed by [118] that considers independent variables $\mathbf{q} = (q_1, q_2, \ldots, q_{D+1})$ that follows Gamma distribution with constant scale parameter and varying shape parameter $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_{D+1})$. Let, $p_d = \frac{q_d}{q_{D+1}}$, $d = 1, 2, \ldots, D$, then vector $\mathbf{p}$ follows inverted Dirichlet distribution, $\mathbf{p} \sim ID(\alpha)$. Initial parameters for ID distribution can be estimated as,

$$\alpha_{D+1} = \frac{E(p_d)^2 + E(p_d)}{Var(p_d)} + 2 \tag{66}$$

$$\alpha_d = E(p_d)(\alpha_{D+1} - 1); d = 1, 2, \ldots, D+1 \tag{67}$$

3.1.2 Generalized Inverted Dirichlet Distribution

In inverted Dirichlet, any two random variables are positively correlated when $\alpha_{D+1} \geq 2$. This is a limitation for inverted Dirichlet distribution. In practice, variables can be positively and negatively correlated and inverted Dirichlet distribution is not an appropriate choice to model such data. A solution to this problem is generalized inverted Dirichlet (GID) distribution that can model both positively and negatively correlated data [1].

$$p(X|\boldsymbol{\Theta}) = \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma\alpha_d \Gamma\beta_d} \prod_{d=1}^{D} \frac{p_d^{\alpha_d - 1}}{(1 + \sum_{l=1}^{d} p_l)^{\gamma_d}} \tag{68}$$

where, $\Theta = \{\alpha_1, \alpha_2, \ldots, \alpha_D; \beta_1, \beta_2, \ldots, \beta_D\}$ and $\gamma_d = \beta_d + \alpha_d - \beta_{d+1}$ and $\beta_{D+1} = 0$. Generalized inverted Dirichlet distribution has twice the number of parameters than the inverted Dirichlet distribution which makes it computationally more expensive. Also, it can be easily shown that this distribution has a more generalized covariance structure and overcomes the limitations of inverted Dirichlet distribution. Generalized inverted Dirichlet distribution can be transformed to inverted Dirichlet distribution by setting $\gamma_1 = \gamma_2 = \ldots = \gamma_{D-1} = 0$ [71]. [71] modified generalized inverted Dirichlet distribution by breaking it into a factor representation of multiple Beta-Prime distributions.

$$p(X_i|\boldsymbol{\Theta}) = \prod_{d=1}^{D} p_{IBeta}(\hat{p}_{id}|\alpha_d, \beta_d) \tag{69}$$

where, $\hat{p}_{i1} = p_{i1}$ and $\hat{p}_{id} = \frac{p_{id}}{1+\sum_{k=1}^{d-1} p_{il}}$ for $l > 1$. Probability distribution of inverted Beta distribution with parameter vectors α and β is defined as,

$$p_{IBeta}(\hat{p}_{id}|\alpha_d, \beta_d) = \frac{\Gamma\alpha_d + \beta_d}{\Gamma\alpha_d \Gamma\beta_d} \hat{p}_{id}^{\alpha_d - 1}(1 + \hat{p}_{id})^{-(\alpha_d + \beta_d)} \tag{70}$$

3.1.3 Inverted Beta Liouville Distribution

Another variation of inverted Dirichlet distribution is inverted Beta-Liouville (IBL) distribution that has less number of parameters than generalized inverted Dirichlet distribution. Given the mentioned D dimensional vector, $\mathbf{p}$, then density function for this distribution can be defined as,

$$p(X_i|\Theta) = \frac{\Gamma(\sum_{d=1}^{D} \alpha_d)\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \prod_{d=1}^{D} \frac{p_{id}^{\alpha_d - 1}}{\Gamma\alpha_d} \frac{\lambda^{\beta}(\sum_{d=1}^{D} p_{id})^{\alpha_d - \sum_{d=1}^{D}}}{(\lambda + \sum_{d=1}^{D} p_{id})^{(\alpha+\beta)}} \tag{71}$$

Where, $\Theta = (\alpha, \beta, \lambda)$ and $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_D)$, $\beta_1, \beta_2, \ldots, \beta_D$ and $\alpha > 0$, $\beta > 0$ and $\lambda > 0$. Inverted Beta-Liouville distribution can be seen as generalization of inverted Dirichlet distribution that supports multiple symmetric and asymmetric modes. The mean, variance and covariance of IBL can be formulated as follows,

$$E(p_{id}) = \frac{\lambda\alpha}{\beta - 1} \frac{\alpha_d}{\sum_{d=1}^{D} \alpha_d} \tag{72}$$

$$Var(p_{id}) = \frac{\lambda^2\alpha(\alpha + 1)}{(\beta - 1)(\beta - 2)} \frac{\alpha_d(\alpha + 1)}{\sum_{d=1}^{D} \alpha_d(\sum_{d=1}^{D} \alpha_d + 1)} - \frac{\lambda^2\alpha^2}{(\beta - 1)^2} \frac{\alpha_d^4}{(\sum_{d=1}^{D} \alpha_d)^4} \tag{73}$$

$$Cov(p_{il}, p_{id}) = \frac{\alpha_l \alpha_d}{\sum_{d=1}^{D} \alpha_d} \Big[\frac{\lambda^2\alpha(\alpha + 1)}{(\beta - 1)(\beta - 2)(\sum_{d=1}^{D} \alpha_d + 1)} - \frac{\lambda^2\alpha^2}{(\beta - 1)^2(\sum_{d=1}^{D} \alpha_d)} \Big] \tag{74}$$

3.2 Parameter Estimation

An important step in our proposed feature mapping approach is to learn the parameters of the distributions. To learn the optimal values of the parameters, we take maximum likelihood approach. More specifically, we take the log-likelihood of the density function as loss function. For example, the log-likelihood of the inverted Beta Liouville distribution can be written as,

$$\begin{aligned}
\mathcal{L}(\mathcal{X}|\mathbf{\Theta}) &= \log(\Gamma\sum_{d=1}^{D}\alpha_d) + \log\Gamma(\alpha+\beta) - \log\Gamma(\alpha) - \log\Gamma(\beta) \\
&+ \sum_{d=1}^{D}\Big((\alpha_d - 1)\log p_{id} - \log\Gamma\alpha_d\Big) + \beta\log\lambda \qquad (75) \\
&+ (\alpha - \sum_{d=1}^{D}\alpha_d)\log(\sum_{d=1}^{D}p_{id}) - (\alpha+\beta)\log(\lambda + \sum_{d=1}^{D}p_{id})
\end{aligned}$$

$$(76)$$

The expectation of the complete log-likelihood in Eq.(75) is utilized to compute the partial derivatives with respect to all the parameters,

$$\frac{\partial\mathcal{L}(\mathcal{X}|\mathbf{\Theta})}{\partial\alpha} = \sum_{i=1}^{N}\Big[\log\sum_{d=1}^{D}p_{id} - \log(\lambda + \sum_{d=1}^{D}p_{id})\Big] + \psi(\alpha+\beta) - \psi(\alpha) \qquad (77)$$

$$\frac{\partial(\mathcal{L}(\mathcal{X}|\mathbf{\Theta}))}{\partial\beta} = \sum_{i=1}^{N}\Big[\log\lambda - \log(\lambda + \sum_{d=1}^{D}p_{id})\Big] + \psi(\alpha+\beta) - \psi(\beta) \qquad (78)$$

$$\frac{\partial\mathcal{L}(\mathcal{X}|\mathbf{\Theta})}{\partial\alpha_d} = \sum_{i=1}^{N}\Big[\log p_{id} - \log\sum_{d=1}^{D}p_{id}\Big] \qquad (79)$$

$$\frac{\partial(\mathcal{L}(\mathcal{X}|\mathbf{\Theta}))}{\partial\lambda} = \sum_{i=1}^{N}\Big[\frac{\beta}{\lambda} - \frac{\alpha+\beta}{\lambda + \sum_{d=1}^{D}p_{id}}\Big] \qquad (80)$$

$\psi(\cdot)$ denotes digamma function. From Eq.(77) to Eq.(80) it is evident that a close-form solution to update each parameter vector does not exist. Therefore, we take Newton-Raphson method to update the parameters iteratively.

$$\theta^{(t+1)} = \theta^{t} - \mathrm{H}(\theta^{(t)})^{-1}\frac{\partial\mathcal{L}(\mathcal{X}|\mathbf{\Theta})}{\partial\theta^{t}} \qquad (81)$$

where $\mathrm{H}(\theta^{(t)})$ is the Hessian of the log-likelihood function. In order to be invertible, the Hessian matrix needs to be positive definite or positive semi-definite. However, due to different combinations of some datasets, it can be negative definite which results in singularity and the Hessian becomes non-invertible [53]. To avoid noninvertible Hessians, we the similar approach as described in [79]. The complete Hessian can be expressed as invertible block-diagonal matrix, $\mathrm{H}(\Theta)^{-1} = \mathrm{BlockDiag}\{\mathrm{H}(\alpha, \beta, \lambda)^{-1}, \mathrm{H}(\alpha_1, \alpha_2, \ldots, \alpha_D)^{-1}\}$. A detailed derivation of this can be found in [60]. To estimate the initial parameters, method of moments technique is applied which is derived from low order statistics of each distribution namely first and second moments equations from where we get mean and variance of that distribution as in Eq.(63) and Eq.(64) for inverted Dirichlet and Eq.(72) and Eq.(73) for inverted Beta Liouville distribution.

3.3 Support Vector Machines with Proposed Feature Mapping

Solving the dual problem is computationally convenient for large datasets. Relaxing the constraints with the help of Lagrange multipliers, dual solution becomes,

$$\begin{aligned}&\underset{\gamma}{\text{maximize}} \sum_{i}^{N} \gamma_i - \frac{1}{2} \sum_{i}^{N} \sum_{j}^{N} \gamma_i \gamma_j y^{(i)} y^{(j)} \langle \phi(p_i), \phi(p_j) \rangle \\ &\text{subject to: } 0 \le \gamma_i \le C,\ \sum_{i} \gamma_i y^{(i)} = 0\ ; \text{ where } i = 1, \ldots N\ \forall\ \alpha_i, y^{(i)}\end{aligned} \tag{82}$$

Only the support vectors have $\boldsymbol{\gamma}$ values elsewhere it is zero. Getting the support vectors, the decision function classifies the data by comparing the kernel with the support vectors. The decision function of the support vector machine becomes,

$$f(p) = \sum_{i}^{n} \gamma_i y^{(i)} \langle \phi(p_i), \phi(p) \rangle \tag{83}$$

Optimum performance of SVM depends on the choice of kernel or feature mapping function $\phi(p_i)$ and since embedding input vector to feature space gives flexibility to modify the kernel function based on the distribution of the data, we take the advantage of the proposed distributions to modify the kernel function as follows,

$$\phi_j(p_i) = \begin{cases} p_{ij} \\ \\ \frac{|\alpha_+|}{\prod_{d=1}^{D+1}\Gamma\alpha_d}\prod_{d=1}^{D} p_{id}^{\alpha_d-1}(1+\sum_{d=1}^{D} p_{id})^{-|\alpha_+|} \\ \\ \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma\alpha_d\Gamma\beta_d}\prod_{d=1}^{D} p_d^{\alpha_d-1}(1+\sum_{l=1}^{d} p_l)^{-\gamma_d} \\ \\ \frac{\Gamma(\sum_{d=1}^{D}\alpha_d)\Gamma\alpha+\beta}{\Gamma\alpha\Gamma\beta}\prod_{d=1}^{D}\frac{p_{id}^{\alpha_d-1}}{\Gamma\alpha_d}\frac{\lambda^\beta(\sum_{d=1}^{D} p_{id})^{\alpha_d-\sum_{d=1}^{D}}}{(\lambda+\sum_{d=1}^{D} p_{id})^{(\alpha+\beta)}} \end{cases} \tag{84}$$

where, $j = 1$ to D for p_{ij} and $j = (D+1)$ elsewhere. It is evident from Eq.(84) that our proposed feature mapping increases the data dimensionality by one. Such technique changes the similarity measurements of the datapoints and we get a new kernel matrix representation for the base kernels such linear, RBF, polynomial, χ^2 etc. [92]. Algorithm 2 shows a high level interpretation of our proposed feature mapping technique.

Algorithm 2: : IDSVM, GIDSVM and IBLSVM

```
1. Input: Training data, $D = \{(X_1, X_2, \ldots, X_N)\}$.
2. Parameter Estimation
    • Initialization: Use Eq.(67) to estimate initial parameter value
    • Optimize: Apply Eq.(81) to optimize the initial parameters.
        repeat
            $\theta^{(t+1)} = \theta^t - \mathrm{H}(\theta^{(t)})^{-1} \frac{\partial \mathcal{L}(\mathcal{X}|\mathbf{\Theta})}{\partial \theta^t}$
        until convergence
3. Compute kernel:
    • Baseline SVM: Compute $\phi_j = p_{ij}$ for $j = 1, \ldots, D$
    • IDSVM: Concatenate $p_{ij}$ and inverted Dirichlet feature from Eq.(84).
        – Raw features: $\phi_j = p_{ij}$        $j = 1, \ldots, D$
        – Inverted Dirichlet feature: $\phi_{D+1}$ =ID($\mathbf{p}_i$, $\alpha$)
        – Concatenate, $\phi_j \oplus \phi_{D+1}$        $j = 1, \ldots, D$
    • GIDSVM: Repeat IDSVM process with generalized inverted Dirichlet
      distribution.
    • IBLSVM: Repeat IDSVM process with inverted Beta Liouville distribution.
4. Learning SVM: Apply algorithm 3 to find the support vectors and
   decision function.
```

Given a set of functions for the given data, the learning algorithm finds the function that minimizes the empirical loss function. By this, the algorithm finds the support vectors that maximizes the margin between the class labels in feature space where number of support vectors is proportional to empirical risk [109].

Algorithm 3: : Find Support Vectors

```
1. Input: $D = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_N, Y_N)\}$.
2. Compute kernel matrix, $\mathbf{S}$
       $S_{i,j} = y^{(i)} y^{(j)} \langle \phi(p_i), \phi(p_j) \rangle$
3. Choose miss-classification parameter, $C$
4. Solve quadratic optimization in Eq.(82). to find $\gamma$
5. Get decision for query vector, $p$ in Eq.(83).
```

3.4 Experimental Results

In this section, we evaluate our proposed method on texture recognition, natural scene recognition and human action recognition in videos. We extract SIFT descriptors from each image and compute a histogram of quantized local descriptors which is described in next section. For all experiments, we use 80% as training and the rest as test set with fixed random seeds. The tolerance limit is set to 10^{-3} and miss-classification penalty parameter, C is varied from 0.003 to 30. The dual SVM form is solved using [37]. We find that allowing a soft margin with lower values of C gives better results. We report f1 score with three different averaging methods such as micro, macro and weighted average. Micro average computes the average f1 score by cumulative contributions of all classes whereas in macro average f1 score is computed independently for each class and then we take the average. We attain weighted average by normalizing macro average with the number of supports for each class. Micro average is preferable in multi-class classification since it takes class imbalance into account.

3.4.1 Texture recognition-KTH TIPS dataset

A classic problem in pattern recognition is texture classification which plays an important role in generalizing image segmentation task, medical image analysis and understanding, image retrieval, industrial inspection, etc. [72]. To evaluate our proposed method in texture classification, we choose KTH-TIPS dataset [77]. This dataset consists of 10 different textures with varying poses and scales under different illumination conditions [51]. Local SIFT descriptors are extracted and quantized as discussed above. Table 6 shows three different scoring value for different models. It is evident that for linear kernel with proposed feature mapping for ID, GID and IBL distributions achieve accuracy of 95.062%, 95.679% and 95.062%, respectively, whereas the baseline result is close to 91.975%. We observe that for this dataset, combining other kernels with our proposed method improves the baseline accuracy with a good margin. RBF, Tanimoto and Bhattacharyya kernel with inverted Beta Liouville distribution achieves better results than inverted Dirichlet and generalized inverted Dirichlet based feature mapping. For cosine and additive χ^2 kernel, highest accuracy is achieved for generalized inverted Dirichlet based feature mapping.

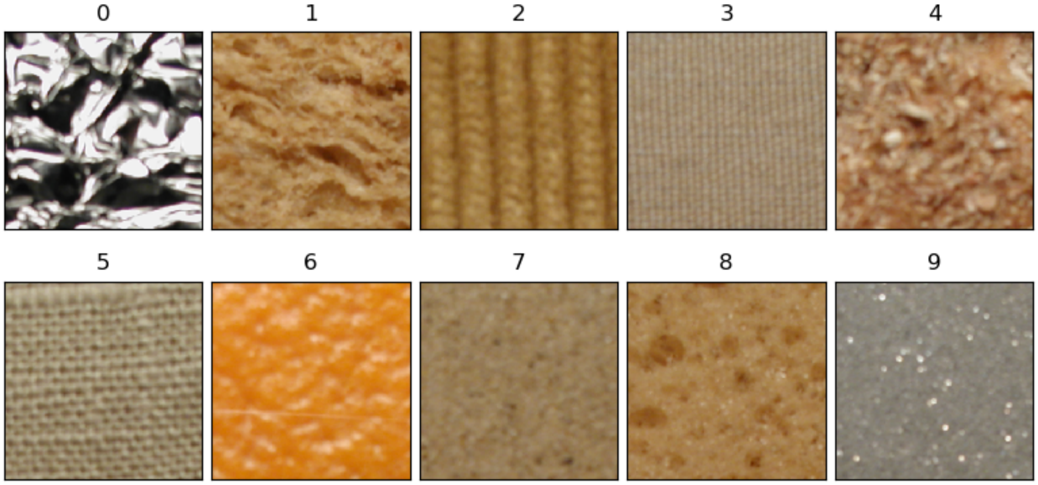
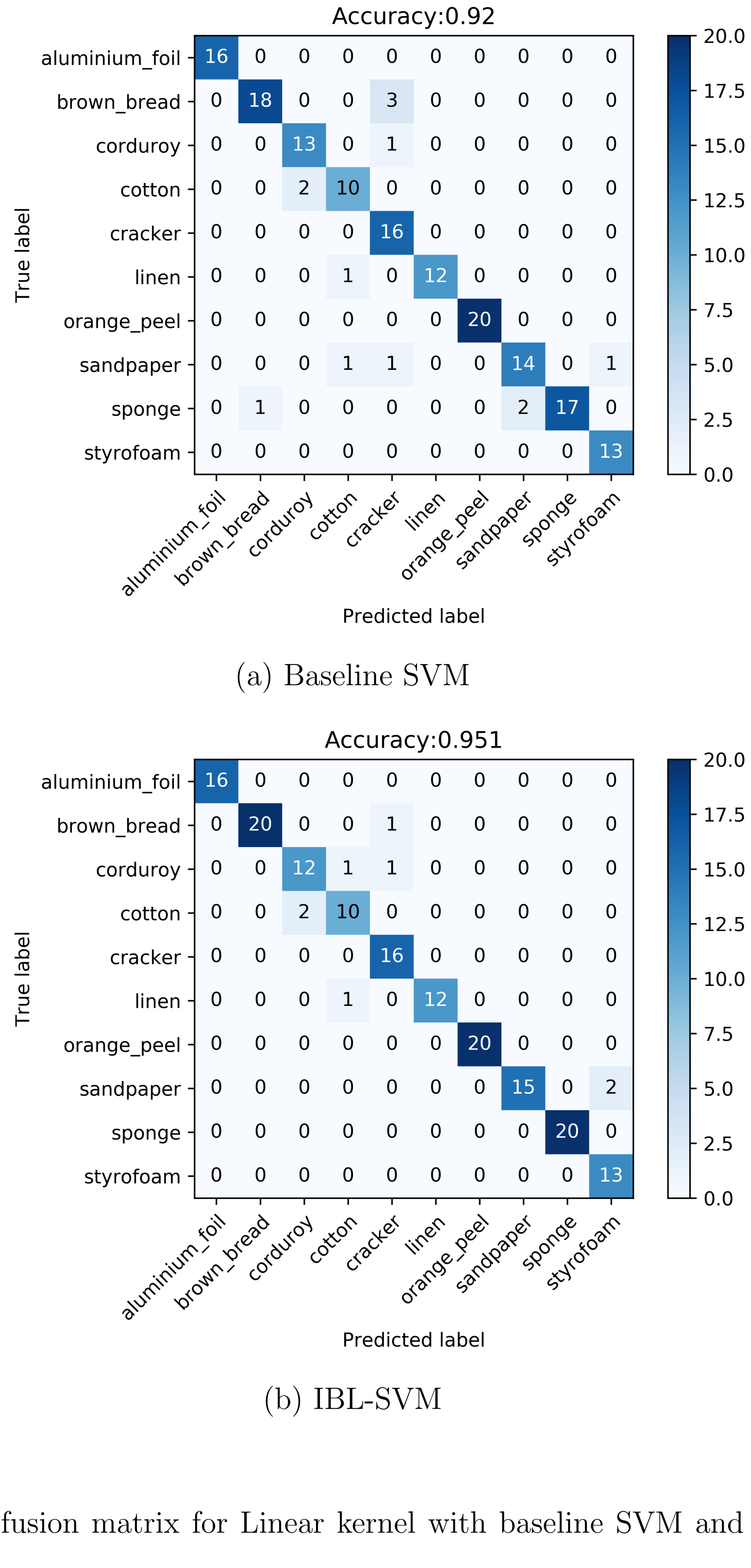


Figure 9: Sample image from KTH-TIPS dataset: 0. aluminium foil, 1. brown bread, 2. corduroy, 3. cotton, 4. cracker, 5. linen, 6. orange peel, 7. sand paper, 8. sponge , 9. styrofoam

Kernel	Map. Func.	f1-score(micro)	f1-score(weighted)	f1 socre(macro)
Linear	Baseline	0.91975	0.92038	0.91828
RBF		0.81481	0.81829	0.81738
Additive χ^2		0.95062	0.94947	0.94443
Bhattacharya		0.96296	0.96279	0.95972
Cosine		0.87654	0.87830	0.87618
Tanimoto		0.91358	0.91385	0.91257
Linear	ID	0.95062	0.95116	0.94544
RBF		0.85802	0.85779	0.85049
Additive χ^2		0.98148	0.98147	0.98257
Bhattacharyya		0.97531	0.97585	0.97141
Cosine		0.91975	0.91975	0.91549
Tanimoto		0.93827	0.93910	0.93622
Linear	GID	**0.95679**	**0.95728**	**0.95479**
RBF		0.86419	0.86632	0.85049
Additive χ^2		**0.98765**	**0.98781**	**0.98659**
Bhattacharyya		0.97531	0.97585	0.97141
Cosine		**0.95679**	**0.95695**	**0.95356**
Tanimoto		0.93209	0.939208	0.93261
Linear	IBL	0.950617	0.950836	0.94333
RBF		**0.87037**	**0.87036**	**0.86486**
Additive χ^2		0.97531	0.97573	0.97138
Bhattacharyya		**0.98765**	**0.98761**	**0.98397**
Cosine		0.89506	0.89661	0.89292
Tanimoto		**0.95062**	**0.95059**	**0.94608**

Table 6: KTH-TIPS texture classification performance results with baseline and proposed feature mapping functions.



	aluminium_foil	brown_bread	corduroy	cotton	cracker	linen	orange_peel	sandpaper	sponge	styrofoam
aluminium_foil	16	0	0	0	0	0	0	0	0	0
brown_bread	0	18	0	0	3	0	0	0	0	0
corduroy	0	0	13	0	1	0	0	0	0	0
cotton	0	0	2	10	0	0	0	0	0	0
cracker	0	0	0	0	16	0	0	0	0	0
linen	0	0	0	1	0	12	0	0	0	0
orange_peel	0	0	0	0	0	0	20	0	0	0
sandpaper	0	0	0	1	1	0	0	14	0	1
sponge	0	1	0	0	0	0	0	2	17	0
styrofoam	0	0	0	0	0	0	0	0	0	13

(a) Baseline SVM

	aluminium_foil	brown_bread	corduroy	cotton	cracker	linen	orange_peel	sandpaper	sponge	styrofoam
aluminium_foil	16	0	0	0	0	0	0	0	0	0
brown_bread	0	20	0	0	1	0	0	0	0	0
corduroy	0	0	12	1	1	0	0	0	0	0
cotton	0	0	2	10	0	0	0	0	0	0
cracker	0	0	0	0	16	0	0	0	0	0
linen	0	0	0	1	0	12	0	0	0	0
orange_peel	0	0	0	0	0	0	20	0	0	0
sandpaper	0	0	0	0	0	0	0	15	0	2
sponge	0	0	0	0	0	0	0	0	20	0
styrofoam	0	0	0	0	0	0	0	0	0	13

(b) IBL-SVM

Figure 10: Confusion matrix for Linear kernel with baseline SVM and IBL feature mapped SVM

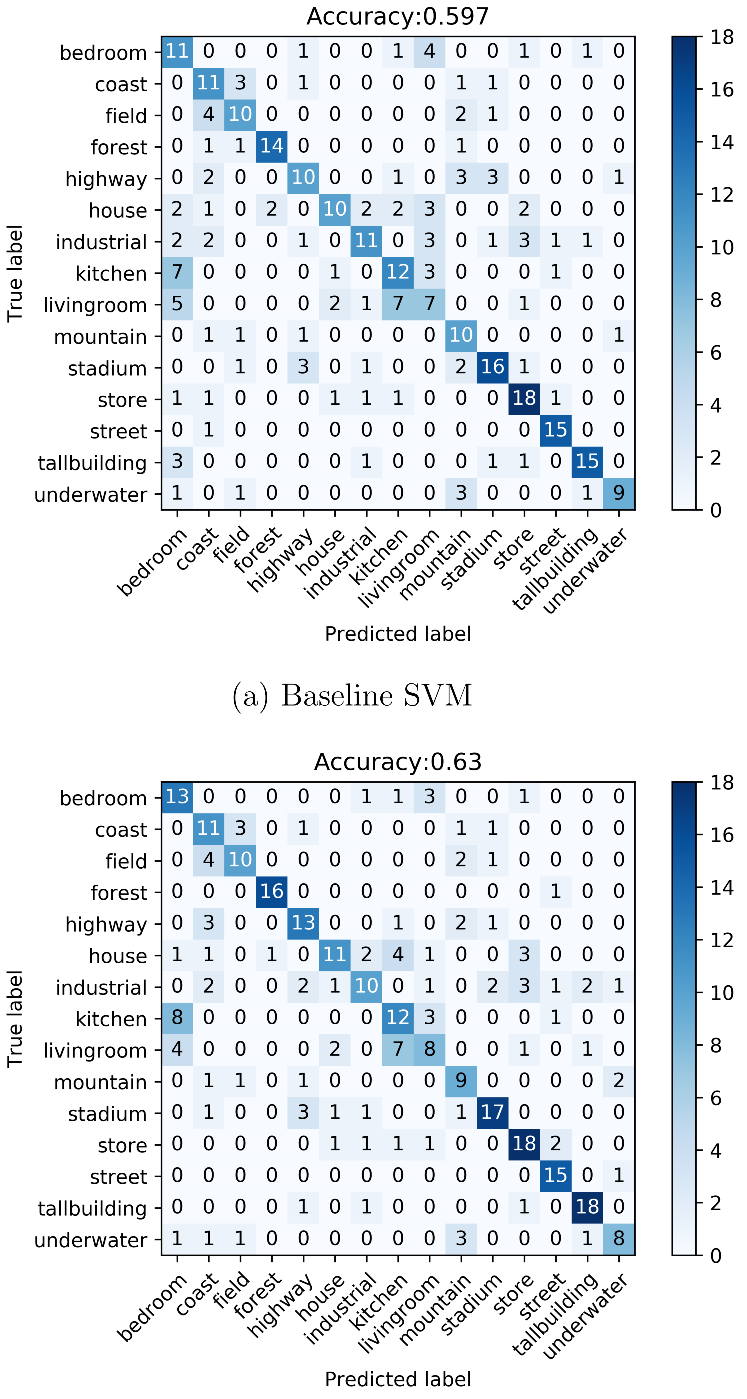
Fig 10 shows the confusion matrix for linear kernel SVM classifier with its core form and our proposed IBL feature mapped SVM. IBL-SVM improves the recognition rate for sandpaper, sponge and brown bread. However, it miss classifies 2 instances of sandpaper and cotton as styrofoam and corduroy, respectively.

3.4.2 Natural Scene Recognition

15 scene dataset consists of 15 different scene categories. First 13 categories were collected combinedly by [49] and [68]. For our experiment, from each category 100 images were selected totalling to 1500 images. We observe the highest improvement in accuracy using inverted Beta Liouville distribution based feature mapping. In core form of SVM, linear kernel with 128 dimensional feature vectors has an accuracy of almost 60.000% while ID-SVM, GID-SVM and IBL-SVM with linear kernel achieve close to 63.000% of accuracy. Highest accuracy of 74.000% is achieved for IBL-SVM with MinMax kernel. For Bhattacharyya kernel, GID-SVM performs better achieving 65.000% accuracy than ID-SVM and IBL-SVM. However, for additive χ^2 kernel, neither of the proposed feature mapping functions based SVM models perform better than core form baseline SVM.

Kernel	Mapping Func.	f1-score(micro)	f1-score(weighted)	f1-score(macro)
Linear	Baseline	0.59667	0.59827	0.60714
Additive χ^2		**0.70667**	**0.70078**	**0.70097**
Bhattacharya		0.64333	0.64066	0.63939
Cosine		0.63000	0.62363	0.62512
Sorensen		0.64000	0.63292	0.63807
Tanimoto-128		0.62333	0.61847	0.62234
MinMax		0.71000	0.70614	0.70352
Linear	ID	0.62667	0.62894	0.62299
Additive χ^2		0.66333	0.66127	0.66204
Bhattacharyya		0.64667	0.64125	0.64473
Cosine		0.63000	0.62363	0.62512
Tanimoto		0.66667	0.66459	0.66145
Sorenson		0.64000	0.63292	0.63807
MinMax		0.71667	0.71469	0.71445
Linear	GID	0.63000	0.62375	0.62680
Additive χ^2		0.66000	0.65712	0.65798
Bhattacharyya		**0.65000**	**0.64426**	**0.64774**
Cosine		0.64667	0.64989	0.65001
Tanimoto		0.66333	0.66146	0.65896
MinMax		0.71333	0.71157	0.71204
Linear	IBL	**0.63000**	**0.62547**	**0.63271**
Additive χ^2		0.69667	0.69315	0.70148
Bhattacharyya		0.64667	0.64371	0.65027
Cosine		**0.65333**	**0.64789**	**0.65404**
Tanimoto		**0.69667**	**0.69481**	**0.69844**
Sorensen		**0.68000**	**0.67619**	**0.67960**
MinMax		**0.74000**	**0.73641**	**0.74249**

Table 7: Natural scene recognition performance results with baseline and proposed feature mapping functions.



(a) Baseline SVM

(b) IBL-SVM

Figure 11: 15 SCENE: Confusion matrix for Linear kernel with baseline SVM and IBL feature mapped SVM

Fig 11 shows unnormalized confusion matrix for linear kernel. Combining inverted Beta Liouville distribution based feature mapping with linear kernel improves the classification accuracy of building, forest, highway, house, living room and tall building.

3.4.3 Human Action Recognition in Videos

An important application in surveillance is activity recognition. To evaluate our proposed method on videos, we use KTH-human action recognition [67] dataset. This dataset has 100 different videos with 6 different categories. To construct the BOVW, we choose to extract optical flow from each video frame using Farneback optical flow algorithm [48]. First, RGB images are converted to HSV colorspace. Next, each frame is resized to 160×120 and further downsampled to 16×12 by taking the pixel position that are divisible by 10. We use open source computer vision library [34] to extract optical flow. Then extracted features are clustered and quantized into a 128 codebooks. Top row in Fig. 12 shows examples of video frames for each video category in the database and the bottom row presents extracted optical flows. Each video frame is subsampled 3 times repeatedly to generate pyramid with averaging window size of 15.

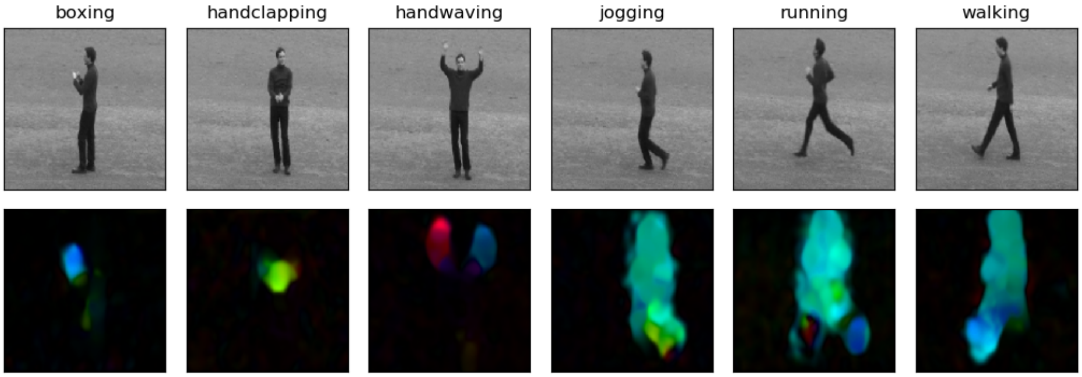
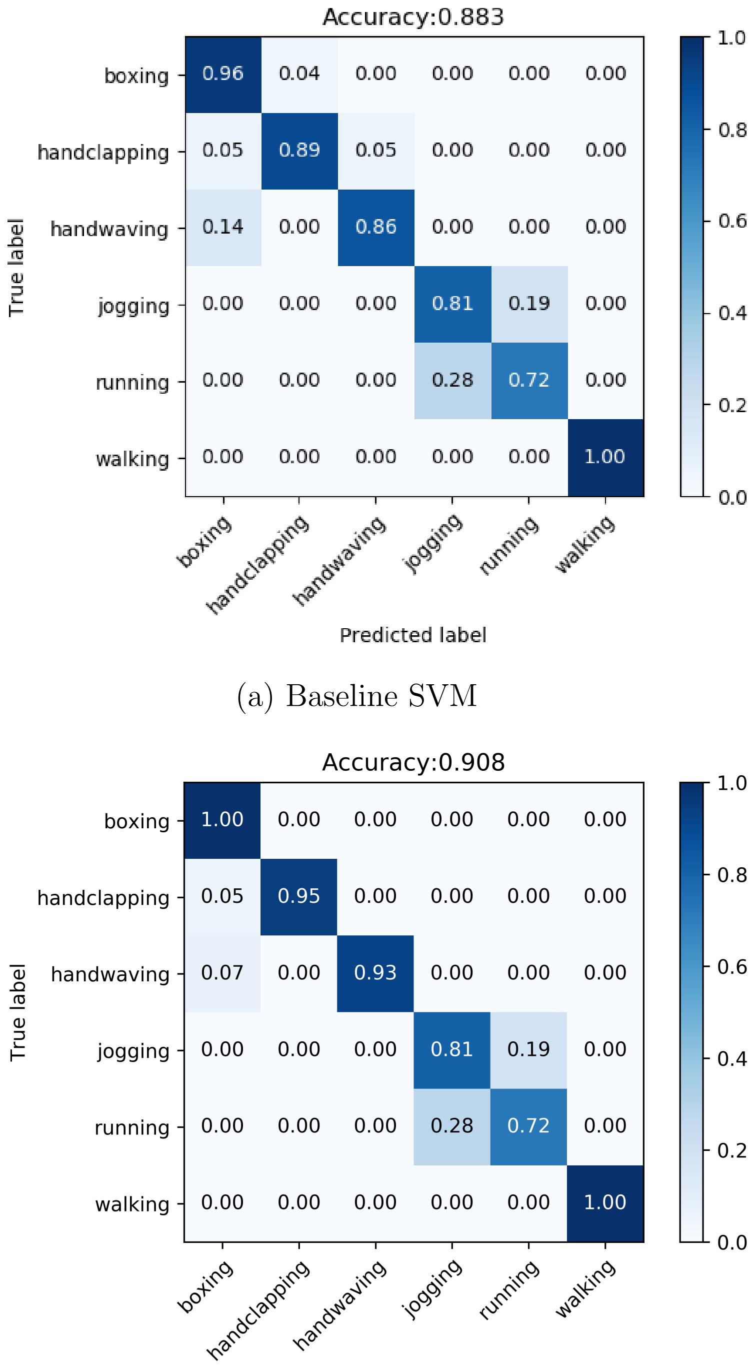


Figure 12: Top row: sample frames from KTH-human action recognition dataset for each categories. Bottom row: optical flow extracted from the corresponding frame.

Kernel	Map. Func.	f1-score(micro)	f1-score(weighted)	f1-score(macro)
Linear	Baseline	0.88333	0.84165	0.87741
Additive χ^2		0.91667	0.91561	0.91662
Bhattacharya		0.90833	0.90637	0.90574
Cosine		0.81667	0.80389	0.79857
Sorensen		0.88333	0.88190	0.88004
Tanimoto		0.91667	0.91556	0.91322
MinMax		**0.94167**	**0.94091**	**0.94017**
Linear	ID	0.90000	0.90180	0.89914
Additive χ^2		**0.93333**	**0.93232**	**0.93325**
Bhattacharyya		**0.93333**	**0.93197**	**0.93185**
Cosine		**0.85833**	**0.85767**	**0.85577**
Tanimoto		**0.92500**	**0.92471**	**0.92224**
Sorenson		**0.90833**	**0.90775**	**0.90769**
MinMax		0.93333	0.93232	0.93099
Linear	GID	**0.90833**	**0.91006**	**0.90672**
Additive χ^2		0.92500	0.92406	0.92566
Bhattacharyya		0.93333	0.93197	0.93185
Cosine		0.85833	0.85767	0.85577
Tanimoto		0.92500	0.92471	0.92224
Sorensen		0.91667	0.91700	0.91237
MinMax		0.93333	0.93232	0.93099
Linear	IBL	**0.90833**	**0.90819**	**0.90517**
Additive χ^2		0.91667	0.91562	0.91661
Bhattacharyya		0.90833	0.90687	0.90573
Cosine		0.84167	0.82726	0.82363
Tanimoto		0.91667	0.91607	0.91361
Sorensen		0.90000	0.89837	0.89689
MinMax		0.93333	0.93219	0.93126

Table 8: KTH-human action recognition performance results with baseline and proposed feature mapping functions.



(a) Baseline SVM

(b) IBL-SVM

Figure 13: KTH-ACTION: Confusion matrix for Linear kernel with baseline SVM and IBL feature mapped SVM

From Table 8, accuracy for all kernels has improved except MinMax kernel. Action recognition rate for boxing, handclapping and handwaving has improved by 4%, 6% and 7% respectively for IBL-SVM than baseline SVM as shown in Fig 13. For action recognition, inverted Dirichlet based feature mapping performs better than other proposed distributions. Additive χ^2, Sorensen, Tanimoto, Cosine and Bhattacharyya kernel with ID feature mapping increases the baseline f1-score. Both linear GID-SVM and IBL-SVM achieve accuracy of 90.833% while the baseline is 88.333%. This 2.5% improvement in performance is due to the fact that the proposed feature mapping technique allows us to model the data more flexibly [82].

Chapter 4

Distribution Based Feature Mapping for Classifying Count Data

4.0.1 Multinomial Distribution

Multinomial distribution is the generalized form of Binomial distribution. In Binomial distribution, the probability of two mutually exclusive features is computed for N independent trials. However, if the features are more than two such that, there can be D possible features for each observation, $\{x_1, x_2, \ldots, x_D\}$ with probabilities $\{p_1, \ldots p_D\}$, then multinomial distribution can be employed to model the dataset. Multinomial models the distribution of the count data (histogram) [21, 24, 27, 28] vector indicating how many times any specific outcome was observed with m trials of experiments.

$$\begin{aligned} P(x_1, \ldots, x_D | p_1, \ldots, p_D) &= \frac{m!}{\prod_{d=1}^{D} x_d!} \prod_{d=1}^{D} p_d^{x_d} \\ &= \binom{m}{\mathbf{x}} \prod_{d=1}^{D} p_d^{x_d} \end{aligned} \tag{85}$$

Here, $m = \sum_{d=1}^{D} x_d$ is the number of trials for each observation. Each count for an observation holds the non-negativity constraint and the probability must satisfy the unit sum constraint such that $x_d \geq 0$ and $p_D = 1 - \sum_{d=1}^{D-1} p_d$.

4.0.2 Dirichlet Distribution

Feature mapping only with multinomial distribution gives us a naive estimation about the likelihood of each observation. In such cases, the unobserved or relatively very small counts gets zero probability which makes the mapping function inefficient. To smooth the estimation of p_d, prior belief for each feature added to each count. The prior follows Dirichlet distribution with parameter $\alpha = (\alpha_1, \ldots, \alpha_D)$.

$$\begin{gathered} p(p_1, \ldots, p_D | \alpha_1, \ldots, \alpha_d) = \frac{\Gamma\left(\sum_d \alpha_d\right)}{\prod_d \Gamma(\alpha_D)} \prod_d^D p_d^{\alpha_d - 1} \\ \sum_d p_d = 1 \quad \text{and} \quad p_d \geq 0 \end{gathered} \tag{86}$$

Positive parameter vector α determines the shape of the distribution. From the moment generating function of the distribution, we get the following properties,

$$\begin{aligned} E(p_d) &= \frac{\alpha_d}{\sum_d \alpha_d} \\ Var(p_i) &= \frac{\alpha_i(\sum \alpha_d - \alpha)}{\sum \alpha_d^2(\sum \alpha_d + 1)} \\ Cov(p_i, p_j) &= \frac{-\alpha_i \alpha_j}{\sum \alpha_d^2(\sum \alpha_d + 1)} \end{aligned} \tag{87}$$

Using Dirichlet prior with multinomial distribution, it is easy to show that the mean probability estimate i.e $E(p_d)$ of each features can be updated as follows-

$$\hat{E}[p_d] = \frac{\alpha_d + x_d}{\sum_d \alpha_d + \sum_d x_d} \tag{88}$$

Eq.(88) shows that unobserved values also gets initial probabilities and gets updated whenever new data arrives.

4.0.3 Combining Dirichlet and Multinomial Distribution

The joint distribution of the Dirichlet and Multinomial distributions results in a compound distribution called Dirichlet Multinomial distribution which is preferred over Multinomial distribution to counter over-dispersion in count data. Consider that we have a count vector $\mathbf{x} = (x_1, x_2, \ldots, x_D)$ with the probability of each element to be drawn with the probability vector $\mathbf{p} = (p_1, p_2, \ldots, p_D)$. Variable x_d represents the number of times the features is observed $i.e$ $x_d = \sum_{\hat{d}} \delta(x_{\hat{d}} - d)$ and the number of trial $m = \sum_d x_d$.The resulting distribution can be expressed as follows for a vector,

$$\begin{aligned} P(\mathbf{x}|\alpha) &= \int_{\mathbf{p}} P(\mathbf{x}|\mathbf{p})P(\mathbf{p}|\alpha) \\ &= \frac{(\sum_d x_d)!}{\prod_d x_d!} \prod_d \frac{\Gamma(\sum_d \alpha_d)}{\Gamma(\alpha_d)} \frac{\Gamma(\alpha_d + x_d)}{\Gamma(\sum_d \alpha_d + \sum_d x_d)} \\ &= \frac{m!}{\prod_d x_d!} \prod_d \frac{\Gamma(\sum_d \alpha_d)}{\Gamma(\alpha_d)} \frac{\Gamma(\alpha_d^{'})}{\Gamma(\sum_d \alpha_d + m)} \\ &= \binom{m}{x} \frac{B(\alpha_d^{'})}{B(\alpha_d)} \end{aligned} \tag{89}$$

Eq.(89) can be simplified further using the properties of gamma (Γ) function [121],

$$\begin{aligned} P(\mathbf{x}|\alpha) &= \binom{m}{x} \frac{B(\alpha_d^{'})}{B(\alpha_d)} \\ &= \binom{m}{x} \prod_d \frac{\Gamma(\sum_d \alpha_d)}{\Gamma(\alpha_d)} \frac{\Gamma(\alpha_d + x_d)}{\Gamma(\sum_d \alpha_d + m)} \\ &= \binom{m}{x} \prod_d \frac{\alpha_d(\alpha_d + 1) \ldots (\alpha_d + x_d - 1)}{\sum_d \alpha_d (\sum_d \alpha_d + 1) \ldots (\sum_d \alpha_d + m - 1)} \end{aligned} \tag{90}$$

Collecting similar terms in the overall samples, the log-likelihood of the entire sample can be simplified further and leads to the foundation of another optimization algorithm namely minorization- maximization (MM) [121]. We use the same representation to compute the first and second derivatives to update the parameters using Newton's method. Eq.(89) can be expressed in factorial form since $\Gamma(n) = (n-1)!$. Using this property, similar terms can be collected efficiently and the log-likelihood can be expressed as follows,

$$\begin{aligned}\mathcal{L}(\alpha) &= \sum_{i}^{N} ln\binom{m_i}{x} + \sum_{i}\sum_{d}\sum_{k=0}^{x_{id}-1} ln(\alpha_d + k) - \\ &\qquad \sum_{i}\sum_{k=0}^{m_i-1} ln(\sum_{d} \alpha_d + k) \\ &= \sum_{i}^{N} ln\binom{m_i}{x} + \sum_{d}\sum_{k=0}^{\max, x_{id}-1} s_{dk} ln(\alpha_d + k) - \\ &\qquad \sum_{k=0}^{\max, m_i-1} r_k ln(\sum_{d} \alpha_d + k)\end{aligned} \tag{91}$$

where

$$r_k = \sum_{i} 1_{\{m_i > k+1\}} \text{ and } s_{dk} = \sum_{i} 1_{\{x_{ij} > k+1\}}$$

From Eq.(87), it is visible that if we model our data using Dirichlet distribution, the features are negatively correlated which is indicated by the negative covariance of the distribution. In real cases, any two features can be positively correlated too. In those cases, Dirichlet distribution fails to capture the relation between the features. However, there's a remedy for this which is discussed in the next section.

4.0.4 Generalized Dirichlet Distribution

Dirichlet distribution is a special case of Generalized Dirichlet distribution. In dimension D, the probability density function of Dirichlet distribution in more generalized form with the parameter vectors $\alpha = (\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \ldots, \alpha_D, \beta_D)$ is defined by,

$$p(P) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} p_d^{\alpha_d - 1}\Big(1 - \sum_{j=1}^{d} p_j\Big)^{\gamma_d} \tag{92}$$

For $\sum_{d=1}^{D-1} p_d < 1$ and $0 < p_d < 1$ for $d = 1 \ldots, D$ and $\alpha_d > 0, \beta_d > 0, \gamma_d = \beta_d - \alpha_{d+1} - \beta_{d+1}$ for $d = 1 \ldots D-1$ and $\gamma_D = \beta_D - 1$ when $d = D$. Generalized Dirichlet distribution reduces to Dirichlet distribution when $\beta_d = \alpha_{d+1} + \beta_{d+1}$. GD gives more flexibility compared to Dirichlet distribution by incorporating d degrees of freedom to the mean probability distribution where in Dirichlet distribution the degrees of freedom is fixed. The mean, variance and covariance of the GD are as follows:-

$$\begin{aligned} E(p_d) &= \frac{\alpha_d}{\alpha_d + \beta_d} \prod_{k=1}^{d-1} \frac{\beta_k}{\alpha_k + \beta_k} \\ Var(p_d) &= E(p_d)\left(\frac{\alpha_d + 1}{\alpha_d + \beta_d + 1} \prod_{k=1}^{d-1} \frac{\beta_k + 1}{\alpha_k + \beta_k + 1} - E(p_l) \right) \\ Cov(p_d, p_k) &= E(p_j)\left(\frac{\alpha_d}{\alpha_d + \beta_d + 1} \prod_{k=1}^{d-1} \frac{\beta_k + 1}{\alpha_k + \beta_k + 1} - E(p_d) \right) \end{aligned} \tag{93}$$

In Dirichlet distribution, the variables are negatively correlated whereas GD distribution has more general covariance structure allowing the variables to have different variance with the same mean. In addition, similar to Dirichlet distribution, Generalized Dirichlet distribution is also a conjugate prior to Multinomial distribution.

4.0.5 Combining Multinomial and Generalized Dirichlet Distribution

Multinomial distribution in combination with generalized Dirichlet distribution gives a flexible covariance structure which gives flexibility for both negatively and positively correlated data.

$$\begin{aligned} &= P(\mathbf{x}|\alpha) \\ &= \int_{\mathbf{p}} P(\mathbf{x}|\mathbf{p}) P(\mathbf{p}|\alpha) \\ &= \frac{(\sum_{d=1}^{D} x_d)!}{\prod_{d=1}^{D} x_d!} \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma\alpha_d \Gamma\beta_d} \frac{\Gamma(\alpha_d^{'})\Gamma(\beta_d^{'})}{\Gamma(\alpha_d^{'} + \beta_d^{'})} \\ &= \binom{m}{x} \prod_{d=1}^{D} \frac{B(\alpha_d^{'}, \beta_d^{'})}{B(\alpha_d, \beta_d)} \end{aligned} \tag{94}$$

4.1 Dirichlet Multinomial and Generalized Dirichlet Multinomial Feature mapping and Parameters Learning

4.1.1 Proposed Feature Mapping

A D dimensional count vector $x = (x_1, x_2, \ldots, x_D) \in \mathbb{R}^{\mathbb{D}}$ can be mapped into the feature space $\phi_i(x_d)$ using the following transformation-

$$\phi_i(x_d) = \begin{cases} x_{id} & d = 1, 2, \ldots, D \\ \\ \binom{m}{x} \frac{B(\alpha_d^{'})}{B(\alpha_d)} & d = D+1 \\ \text{or} & \\ \binom{m}{x} \prod_{d=1}^{D} \frac{B(\alpha_d^{'},\beta_d^{'})}{B(\alpha_d,\beta_d)} & d = D+1 \end{cases} \tag{95}$$

By doing this, dimension of each vector is increased by one and the new dimension is $\phi_i(x_{D+1})$ for Dirichlet Multinomial SVM and Generalized Dirichlet Multinomial SVM.

$$\begin{gathered} \min_{w,b,\xi} \ \frac{1}{2}||w||^2 + C \sum_{i=1}^{D+1} \xi_i \\ \text{s.t., } y^{(i)}(w^{\mathrm{T}} x_d) + w_{D+1} \frac{(\sum_{d=1}^{D} x_d)!}{\prod_{d=1}^{D} x_d!} \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma \alpha_d \Gamma \beta_d} \frac{\Gamma(\alpha_d^{'})\Gamma(\beta_d^{'})}{\Gamma(\alpha_d^{'} + \beta_d^{'})} + b) \geq 1 - \xi_i, \\ \xi_i \geq 0, \ \ i = 1, \ldots, n \end{gathered} \tag{96}$$

As stated earlier, it is possible to derive the dual form of the above primal problem which results in the same optimum value except that the Lagrange multipliers γ determines the optimum value of the dual form. Replacing Eq.(95) in the primal form, we get the dual formulation of the optimization problem which we call Dirichlet Multinomial SVM or Generalized Dirichlet Multinomial SVM depending on the selection of the feature mapping function in Eq.(95).

$$\begin{aligned}\mathcal{Q}(\gamma) &= \sum_{i=1}^{N}\gamma_i - \sum_{i=1}^{N}\sum_{j=1}^{N}\gamma_i\gamma_j y_i y_j \langle\phi_i(x_d)\phi_j(x_d)\rangle \\ \text{s.t.} &\sum_{i=1}^{N}\gamma_i y_i = 0 \\ &0 \leq \gamma_i \leq C\end{aligned} \tag{97}$$

In the above equation, $K(\mathbf{x}, \mathbf{y}) = \langle\phi_i(x_d)\phi_j(x_d)\rangle$ represents linear kernel function. Apart from the common kernels such as polynomial, RBF, χ^2 [39] histogram intersection (HI) and min max [8], other measurements such as Cosine similarity, Tanimoto coefficient and Sorensen similarity are proved to be efficient for histogram based representation of the data. Since, the data dimension is increased by one, the training time complexity of solving the dual svm optimization problem with the proposed feature mapping technique becomes $\mathbf{O}(max(N, D+1)min(N, D+1)^2)$ [38].

4.1.2 Parameter Estimation

The probability of observing each outcome can be determined from the frequencies i.e the observed proportions of each dimension represents its probability. For a single count vector, the proportion vector can be calculated as follows,

$$p_d = \frac{x_d}{\sum_{d=1}^{D} x_d} \tag{98}$$

In order to get the optimized parameters of the DM and GDM based feature mapping function, we take maximum likelihood estimation approach since there exists no close form solution of the log-likelihood function described in Eq.(91) and use Newton Raphson method as the optimization algorithm. For any optimization algorithm, it is required to estimate the initial values of the parameters and calculate the gradient or an auxiliary function of the objective function. Choosing random initial values of the parameters is troublesome since it increases the possibility to converge at the local maxima. Therefore, the initial values of the parameters for this algorithm are estimated by a low order method of statistics such as method of moments (MoM) [29,79].

Using gradient and Hessian of the log-likelihood of Eq.(91), we can update the parameters until convergence using the equation (see [22] for details). Here, g is a $D \times 1$ matrix that represents the gradient of the log-likelihood function with respect to each parameter α_d

$$\begin{aligned} g(\alpha_d) &= \frac{\partial \mathcal{L}(\alpha)}{\partial \alpha_d} \\ &= \sum_{k=0}^{\max, x_{id}-1} \frac{s_{dk}}{(\alpha_d + k)} - \sum_{k=0}^{\max, m_i-1} \frac{r_k}{(\sum_{d=1}^{D} \alpha_d + k)} \end{aligned} \tag{99}$$

Taking the second derivative of the log-likelihood function results in the Hessian,

$$\begin{aligned} H(\alpha_d) &= \frac{\partial \mathcal{L}^2(\alpha)}{\partial \alpha_d^2} \\ &= \sum_{k=0}^{\max, m_i-1} \frac{r_k}{(\sum_{d=1}^{D} + k)^2} - \sum_{k=0}^{\max, x_{id}-1} \frac{s_{dk}}{(\alpha_d + k)^2} 1_{(d=d')} \end{aligned} \tag{100}$$

The negative of the Hessian matrix is to be taken to calculate the Newton update which is essentially the observed information matrix:

$$\begin{aligned} -H(\alpha) &= \sum_{k=0}^{\max, x_{id}-1} \frac{s_{dk}}{(\alpha_d + k)^2} 1_{(d=d')} - \sum_{k=0}^{\max, m_i-1} \frac{r_k}{(\sum_{d=1}^{D} + k)^2} \\ &= b - a 1_d 1_d^T \end{aligned} \tag{101}$$

[78] provided the inversion of Hessian using Sherman-Liberman formula,

$$(b - a 1_D 1_D^T)^{-1} = b^{-1} + \frac{a}{1 - a 1_D^T b^{-1} 1_D} b^{-1} 1_D 1_D^T b^{-1} \tag{102}$$

The Newton update becomes,

$$\alpha^{new} = \alpha^{old} + (-\mathbf{H}^{-1})\mathbf{g} \tag{103}$$

When, matrix -H is positive definite, inverting the Hessian i.e. -H^{-1} works fine except the time complexity. For each iteration it requires to be calculated in a linear system and inverting the Hessian matrix has complexity of $O(t^3)$. However, if the Hessian matrix is singular or negative definite, the inversion becomes impossible or violates the constraint of the parameters. For negative definite Hessian *i.e.* the negative eigenvalues of the matrix $(-H)$ leads to the parameters α being negative which is against the constraint $\alpha_d > 0$. In addition, since the function $\psi(x)$ is convex, the gradient function, g is concave and Newton's method may fail to converge to the maxima. To overcome this situation, a more generic algorithm called MM (minimization-maximization) is preferred [121]. Replacing the Gamma functions in the original log-likelihood function with the rising polynomials makes it possible to derive the algorithm. A derived surrogate or auxiliary function carries out the parameters of the objective function to an optimum value. At each iteration, the update of the parameters satisfies the condition $\mathcal{L}(\alpha^{\text{new}}) \geq g(\alpha^{\text{new}}|\alpha^{\text{old}}) \geq g(\alpha^{\text{old}}|\alpha^{\text{old}}) = \mathcal{L}(\alpha^{\text{old}})$. Thus, since the value of log-likelihood function increases at each iteration, therefore, MM algorithm guarantees an ascent algorithm. In Eq.(91), applying Jensen's inequality to the convex term $ln(\sum_d \alpha_d + k)$ and concave term $ln(\alpha_d + k)$ we get the following surrogate function at current iteration,

$$g(\alpha|\alpha^{\text{old}}) = \sum_{d=1}^{D} \sum_{k=0}^{\max,x_{id}-1} \frac{s_{dk}\alpha_d^{\text{old}}}{(\alpha_d^{\text{old}} + k)} ln\alpha_d - \sum_{k=0}^{\max,m_i-1} \frac{r_k}{(\sum_{d=1}^{D} \alpha_d^{\text{old}} + k)} \sum_{d=1}^{D} \alpha_d \tag{104}$$

In MM algorithm, a surrogate function of the original function is defined and iteratively update the parameters until it reaches the maximum value of the surrogate function,

$$\alpha_d^{new} = \alpha_d^{old} \frac{\sum_k \frac{s_d k}{\alpha_d^{old}+k}}{\sum_k \frac{r_k}{\sum_d \alpha_d^{old}+k}} \tag{105}$$

Updating the parameter vector using Eq.(105) is simpler than Newton-Raphson method and non-negativity constraint of updated parameters which is $\alpha^{\text{new}} > 0$ is always satisfied given that $\alpha^{\text{old}} > 0$. In our experiment, we use both optimization technique and the parameters with the maximum log-likelihood values are chosen. To update the parameters of Generalized Dirichlet Multinomial, we use the concept of complete neutrality [45] and transform each variable to a Beta Binomial distribution which is parameterized by (α_d, β_d). Thus, use can update the parameters for Generalized Dirichlet Multinomial using the same technique as Dirichlet Multinomial.

4.2 Experimental Results

In this section, we investigate the effectiveness our proposed feature mapping technique by applying it to two different classification tasks namely natural scene recognition from images and human action recognition in videos. For each task, the dual problem of the SVM model is solved using [88]. All the parameters are kept as default except the misclassification parameter C. For each model, 15 different values of C are taken in log scale varying from 0.0001 to 15. To measure the performance of the model, 10-fold cross validation technique is considered where 9 folds are used to train and the remaining fold is used to test the model. Mean classification accuracy and standard deviation are reported for each kernel and are compared with the baseline SVM. In baseline SVM, only the original count data are taken according to the mapping function in Eq.12. For polynomial kernel degree 3 is considered and for Rational Quadratic and Inverse Multiquadratic kernel the hyperparamter c is set to 1. Obtained results using the proposed feature mapping technique has proved to be statistically significant [82].

4.2.1 Natural Scene Classification

Kernels	Baseline SVM	DM SVM	GDM SVM
Linear	0.67933 ± 0.020	**0.69000 ± 0.019**	0.67933 ± 0.019
Polynomial	0.69000 ±0.029	0.69333 ± 0.026	**0.69533 ± 0.027**
RBF	0.7113 ± 0.036	**0.71466 ± 0.035**	0.70933 ± 0.032
Cosine Similarity	0.680666± 0.023	**0.69466 ± 0.026**	0.69400 ± 0.023
Exponential	0.70866 ± 0.030	0.70800 ± 0.029	**0.71000 ± 0.033**
Rational Quadratic	0.70000 ± 0.033	0.70066 ± 0.037	**0.70533 ± 0.038**
Inverse Multiquadratic	0.71066 ± 0.031	0.71466 ± 0.035	**0.71733 ± 0.031**
Sorensen	0.69200 ± 0.019	**0.69466 ± 0.030**	**0.69466 ± 0.030**
Tanimoto	0.71400 ± 0.031	**0.72400 ± 0.029**	0.71800 ± 0.030
GHIK	0.71933 ± 0.034	**0.72333 ± 0.039**	0.72000 ± 0.038
Min Max	0.72666 ± 0.036	**0.73333 ± 0.026**	0.72733 ± 0.038

Table 9: Scene classification results

Scene recognition is crucial for reasoning in navigation and recognition tasks. Specially in terms of robotics and automation it is significant to enhance machine's visual understandings [117], [30]. 15 scene dataset consists of 15 different scene categories.

First 13 categories were collected jointly by [49] and [68]. For our experiment, 1200 images are randomly selected ranging from 90-100 images from each category. Scale invariant feature transform (SIFT) [73] descriptors are extracted from each image. For our experiment, dense SIFT descriptors are drawn out from each images with 16 pixels interval. Extracted keypoints are quantized into a 200 vocabulary size. Finally each image is represented by a 200 dimensional count vector. To prevent over-fitting and reducing the variance, the dataset is randomly shuffled and normalized. Mean accuracy and the standard deviation for the kernels that give comparable results are reported in Table 12. It is obvious that, the DM SVM improves the performance of linear kernel.

4.2.2 Human Action Recognition

The dataset contains 6 categories. Each category has 100 videos with 4 different scenarios and each action is performed by 25 different peoples with different variations. Each video is on an average of 4s and 2391 frames. Each frame is down sampled to 160×120 as indicated by the original paper [67]. In video analysis, optical flow is used as measurement of the apparent motion of the brightness patterns between the consecutive frames. To show the novelty of our proposed method, we adopt a simple feature extraction pipeline. In our experiment, dense optical flow is calculated for each frame using Farneback's algorithm [48]. For faster calculation, we down sample each frame to 16×12 $[width \times height]$ and calculate the optical flow using [34]. Thus, we get 384 descriptors for each frame and all the frames are quantized into 500 cluster centers. We follow the data split mentioned in [67] and then we do the concatenation of training and validation set. 384 videos are used for this experiment. Incorporating our proposed feature mapping technique with other kernels improves the performance of the action classification as indicated in Table 10.

Kernels	**Baseline SVM**	**DM SVM**	**GDM SVM**
Linear	0.89256 ± 0.044	0.89256 ± 0.044	**0.89778 ± 0.040**
Polynomial	0.89262 ± 0.028	0.89466 ± 0.038	**0.89942 ± 0.032**
RBF	0.90532 ± 0.026	**0.90764 ± 0.043**	0.90572 ± 0.032
Cosine Similarity	0.91093 ± 0.025	0.91093 ± 0.025	**0.91609 ± 0.030**
Exponential	0.91813 ± 0.040	**0.92289 ± 0.041**	0.91813 ±0.040
Rational Quadratic	0.89568 ± 0.049	0.89290 ± 0.047	**0.89846 ± 0.053**
Inverse Multiquadratic	0.91093 ± 0.044	0.91371 ± 0.038	**0.91609 ± 0.048**
Sorensen	0.91325 ± 0.027	**0.92085 ± 0.033**	0.91603 ± 0.032
Tanimoto	0.91564 ± 0.028	**0.92119 ± 0.034**	0.91881 ± 0.033
GHIK	0.91569 ± 0.029	0.91365 ± 0.029	**0.91853 ± 0.031**
MinMax	0.92567± 0.034	**0.92845 ± 0.034**	0.92607 ± 0.033

Table 10: Human Action Recognition results

It is to be noted here that we run the experiments with and without normalization and the best results are reported here. GDM SVM improves the accuracy of base linear SVM by 0.52% and gives less standard deviation. Incorporating DM SVM and GDM SVM with other kernels improves the baseline accuracy for respective kernels as well.

Chapter 5

Parametric features for online object tracking

Online object tracking is of great importance given its many applications in several areas of computer vision such as surveillance, human-robot interaction, motion analysis, traffic safety, and robotics. Traditional tracking algorithms are classified into two categories: generative and discriminative, where the main learning task in the latter is to learn a mapping from target's visual features to a structured output form, i.e., bounding box. In generative classifiers, the appearance of the object is learned to search for it in subsequent frames with the tracker following the patches where the reconstruction error is the lowest. However, in discriminative classifiers ([5], [6], [113], [7], [85]), the problem is solved as a classification problem by finding the boundary between the target and background instead of minimizing the differences of feature values between successive frames. In general, the performance of the generative process is less than that of the discriminative classifiers [25].

Either approaches requires four modules of tracking: object initialization, appearance modelling, motion estimation, and object localization [69]. In appearance modelling, either global features (raw pixels, optical flow etc.) or local features (SIFT [73], [10] etc.) are used to compute statistical descriptors. Detection-based tracking uses such features to localize the target. In image classification, hybrid methods are developed to take advantage of both generative and discriminative approaches while learning from the features. For example, [25], [23] develop a technique to learn proportional data on a simplex manifold exploiting Dirichlet, Generalized

Dirichlet (GD) and Beta-Liouville (BL) distributions which improves the classification accuracy. In a similar manner, [33], [63] uses Scaled Dirichlet (SD) distribution to improve the classification accuracy of retinal images. Such distributions embed data on simplex manifold with different degrees of freedom and allow more efficient data modeling. In tracking, object appearance varies due to noise disturbance, occlusion, pose variation, etc. [69]. A key factor in object tracking is to carefully design and merge different features to get a robust representation to increase the discriminative power of the tracking model [63, 107]. A parametric feature learning of the target object using Dirichlet and related distributions may help to capture representations invariant to noise [81]. This further motivated our study of such distributions in object tracking.

5.1 Proposed Framework

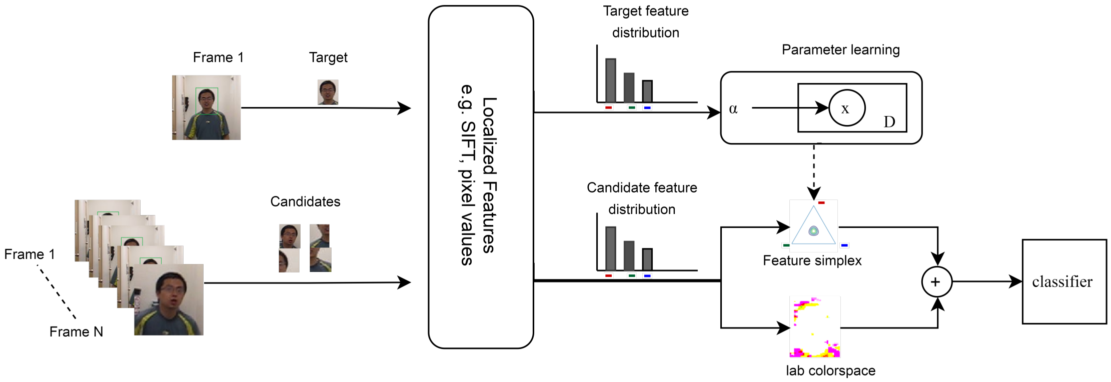


Figure 16: Architecture of our proposed generative feature mapping for online object tracking.

A given set of video frames, $\mathcal{F} = \{F_n\}_{n=1}^{N}$, and the initial state, S_1, of the target at the initial frame, $F_1 \in \mathcal{F}$, represents the standard setup for detection-based object tracking [85], [58]. Our goal is to predict the state of the target object $S_2, S_3, \ldots, S_N$ in frames $F_2, F_3, \ldots, F_N$. Let, f_n be the central location of the object at frame n, and d is the relative displacement of the location of the object in the next frame according to f_n. Hence, a new position is attained on the next frame denoted by $f_n \circ d$. A search space, Ω, is defined within radius r of this new position. We then extract the image patches, $\{I\}_{i=1}^{K}$, from this search space and each patch is a defined as a target candidate. For a Bag of Visual Words (BOVW) or a color space histogram representation, each image is considered $I_i : \mathbb{R}^2 \rightarrow \{1, 2, \ldots, D\}$ as a collection of codebooks (visual words) in some feature space, $\chi \in I_i = \{p_j\}_{j=1}^{D}$ where $\sum_{j=1}^{D} p_j = 1$. The codebooks are generated from each localized image patches with a fixed number of bins. The histogram representation of the image is a probability distribution and is proportional. Following the notations of [42], we can write the distributions of the target (h_t) and each of the potential candidates (h_c) as:

$$\begin{cases} \text{target distribution: } h_t = \{\hat{p}_{jt}\}_{j=1}^{D}; \ \ \sum_{j=1}^{D} p_{jt} = 1 \\ \text{candidate distribution: } \ \ h_c = \{\hat{p}_{jc}\}_{j=1}^{D}; \ \ \ \sum_{j=1}^{D} p_{jc} = 1 \end{cases} \tag{106}$$

Adhering to Eq.((106)), we can exploit proportional data distributions to compute meaningful features of the candidates in the probability simplex. The parameters of the distributions are learned from the BOVW representation or color histogram representation of the localized target in the initial frame. The learned parameters are then used to embed features for the target candidates of later frames in the probability simplex. In contrast to fully Bayesian approaches where posterior probability density is computed from likelihood and prior density up-to the current frame to update the tracker [112], our method uses the prior density which is then utilized as a feature in a discriminative classifier. The architecture of our proposed method is presented in Fig. 16. It is to be noted that, in our experiments, we use the RGB color space histogram representation to learn the parameters of the target distribution and combine CIE lab color space with the features in the probability simplex. For discriminative classifier, we choose online dual linear structured support vector machine (DLSSVM) [85]. DLSSVM has superior tracking performance on benchmark dataset [116] than traditional models for structured prediction such as kernelized object tracking [42, 58] and sub-gradient based tracking [98].

5.2 Distributions for proportional data

Dirichlet and related distributions are most natural to model compositional data or measure data proportionately [84]. It is a distribution over multinomials in a simplex. If a vector $\mathbf{p} = (p_1, p_2, \ldots, p_D)$ of length D resides in a D dimensional closed simplex of $\mathbb{R}^D$ then the data composition is defined as $\mathbb{C}(1) = \{\mathbf{p} \in \mathbb{R}^D : p_1 + \ldots + p_D = 1,$

$p_d \geq 0, 1 \leq i \leq D\}$. Here, $\mathbb{C}(n) = \mathbb{C}(1)$ and n is sum of the multinomials, which is 1 in this case. If the proportional vector $\mathbf{p}$ is parameterized by a positive shape parameter vector $\Theta = (\alpha_1, \alpha_2, \ldots, \alpha_D)$, then Dirichlet probability density function is defined as,

$$p(\mathbf{p}|\Theta) = \frac{\Gamma\left(\sum_d \alpha_d\right)}{\prod_d \Gamma(\alpha_d)} \prod_{d=1}^{D} p_d^{\alpha_d - 1} \tag{107}$$

One of the shortcomings of Eq.(107) is that any two random variables are negatively correlated. To model $\mathbf{p}$ with a flexible covariance structure, the GD distribution with parameter vector $\Theta = (\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_D, \beta_D,)$ can be employed,

$$p(\mathbf{p}|\Theta) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma\alpha_d \Gamma\beta_d} p_d^{\alpha_d - 1} \Big(1 - \sum_{l=1}^{d} p_l\Big)^{\gamma_d} \tag{108}$$

From Eq.(108), we note that the GD has twice the number parameters than Dirichlet distribution. It has no closed form solution to optimize the parameters which renders it computationally expensive [25]. Nonetheless, a closed form solution for the parameter β can be attained by satisfying unit sum constraint to make a computationally cheaper SD distribution [33],

$$p(\mathbf{p}|\Theta) = \frac{\Gamma \sum_{d=1}^{D} \alpha_d}{\prod_{d=1}^{D} \Gamma\alpha_d} \frac{\prod_{d=1}^{D} \beta_d^{\alpha_d} p_d^{\alpha_d - 1}}{\Big(\sum_{d=1}^{D} \beta_d p_d\Big)^{\sum_{d=1}^{D} \alpha_d}} \tag{109}$$

The BL distribution is another extension of the Dirichlet distribution with an additional two parameters that characterize the sum of the vector elements [23]. The BL probability density function with $\Theta = (\alpha_1, \ldots, \alpha_D; \alpha, \beta)$ and $\alpha_+ = \sum_{d=1}^{D} \alpha_d$ is defined as,

$$p(\mathbf{p}|\Theta) = \frac{\Gamma\alpha_+}{B(\alpha, \beta)} \prod_{d=1}^{D} \frac{p_d^{\alpha_d - 1}}{\Gamma\alpha_d} (\sum_{d=1}^{D} p_d)^{\alpha_d - \alpha_+} (1 - \sum_{d=1}^{D} p_d)^{\beta - 1} \tag{110}$$

Another variation in the Dirichlet family of distributions is Inverted Dirichlet distribution which is a generalization of Beta Prime distribution and is relaxed from unit sum constraint. The joint probability function for this distribution with $\Theta =$

$(\alpha_1, \ldots, \alpha_D; \sum_{d=1}^{D+1} \alpha_d)$ is defined as [14],

$$p(\mathbf{p}|\Theta) = \frac{\sum_{d=1}^{D+1} \alpha_d)}{\sum_{d=1}^{D+1} \Gamma \alpha_d} \prod_{d}^{D} p_d^{\alpha_d - 1} (1 + \sum_{d=1}^{D} p_d)^{-\sum_{d=1}^{D+1} \alpha_d} \tag{111}$$

5.3 Parameter Learning and kernel approximation

5.3.1 General framework to update the parameters

For efficient learning, it is important to estimate the parameters of the distributions optimally. In this paper, we adopt the maximum likelihood estimation (MLE) with Newton-Raphson algorithm to maximize the probability of the sampled vectors of the target frame. The maximum likelihood (ML) estimate results in the optimum values of the latent parameters to compute density features on the simplex and is given as $\Theta = \text{argmax}_{\theta} \log p(\mathbf{p}|\Theta)$, where Θ denotes the set of parameters and $p(\mathbf{p}|\Theta)$ is the likelihood function. Given the initial or current estimates of the parameter vector Θ, the log likelihood is computed as $\mathcal{L} = \sum_i^N \sum_{d=1}^{D} \log p(\mathrm{p}_{id}|\theta_d)$. The gradient, $\mathbf{g}$ is computed with respect to the parameter being updated [25], [23]. To describe the local curvature, the Hessian (a second-order partial derivative of a block-diagonal matrix) is computed as, $\mathbf{H} = \nabla^2 \log(p(\mathbf{p}|\Theta) = \mathbf{B} + 1_{\mathrm{D}} 1_{\mathrm{D}}^{\mathrm{T}} \mathbf{b}$. Where, $\mathbf{B} \hat{=}$ diag: $\mathbb{R}^D \rightarrow \mathbb{R}^{\mathrm{DxD}}$ is the diagonal elements of Hessian matrix and $\mathbf{b}$ is a constant value of that matrix elsewhere [25], [23]. In the Newton's method defined as $\Theta^{new} = \Theta^{old} - \mathbf{H}^{-1}\mathbf{g}$, the Hessian needs to be inverted numerically to avoid singularity [79] and it can be easily shown that:

$$\mathbf{H}^{-1} = \mathbf{B}^{-1} - \frac{\mathbf{B}^{-1} 1_{\mathrm{D}} 1_{\mathrm{D}}^{\mathbf{T}} \mathbf{B}^{-1}}{\mathbf{b}^{-1} + 1_{\mathrm{D}}^{\mathrm{T}} \mathbf{B}^{-1} 1_{\mathrm{D}}} \tag{112}$$

The method of moments technique is a common choice for initial guess of the parameters [79].

5.3.2 Approximate non-linear kernels

Linear classifiers are faster to train while non-linear classifiers are computationally expensive although provide better classification results [89]. Linear classifiers based on additive models can be trained on max margin frameworks to approximate non-linear kernels with faster training time [59,76]. Grayscale pixels are transformed to a unary representation based on feature transformation technique described in [75, 85, 119]. In a similar manner, we concatenate density based features. Pixel based features embedded in the probability simplex can be approximated by an additive kernel such as histogram intersection kernel or min kernel. Assume that ρ_i and ρ_j are density representation of two image patches in the probability simplex and p_i and p_j are feature vectors as described in Eq.(106) of the corresponding image patches, then the min kernel can be represented as follows:

$$K_{int} = \text{CONCAT}\Big(\sum_{d=1}^{D} \min(p_i^d, p_j^d), \min(\rho_i, \rho_j)\Big) \tag{113}$$

where d is the d-th element of the feature vector. Using explicit feature mapping technique [119], we can approximate Eq. (113). Let N be the number of discrete levels and $U(n)$ the encoded representation where $n \in D$, then the feature mapping is defined as follows:

$$\phi(p^d, \rho) = \text{CONCAT}(U(R(Np^d)), U(R(N\rho))) \tag{114}$$

where $R(.)$ is a rounding function and $U(.)$ is a unary transformation function. For example, if $N = 6$, then $\phi(0.6, 0.3) = \text{CONCAT}(U(3), U(3)) = [1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0]$. Therefore, the intersection kernel can be approximated as,

$$\sum_d \min(p_i^d, p_j^d) \approx \sum_d < \phi(p_i^d, \rho_i), \phi(p_j^d, \rho_j) > \tag{115}$$

Similar to [85], we set quantization number, N = 4 for both color channel and density based features. This allows us to get a feature vector proportional to 8 for each image patches.

5.4 Experimental Results

5.4.1 Experimental Setup, Dataset and Evaluation Metric

For fair evaluation of different models, we set the same classifier configuration as [85]. Each of the image patches are reduced to 50% of the original size for faster evaluation and the number of support vectors are fixed to 100. The experiment is conducted on a PC configured with Intel i5-2400 CPU (2.3GHz) and 8.0 GB of RAM. Our proposed method is tested on 8 challenging video sequences [116] of mixed attributes of illumination variation (IV), occlusion (OCC), deformation (DEF), out of plane rotation (OPR), scale variation (SV), fast motion (FM), in-plane-rotation (IPR), and background clutter (BC). We choose region overlap based on Jaccard Index as evaluation metric over center error or precision metric since it is sensitive to subjective annotation as it ignores the target size and misleads to incorrect tracking result [36]. Given the area of ground truth bounding box as Λ_G and area of predicted bounding box as Λ_T, overlap becomes,

$$\Delta(\Lambda_G, \Lambda_T) = \frac{\Lambda_G \cap \Lambda_T}{\Lambda_G \cup \Lambda_T} \tag{116}$$

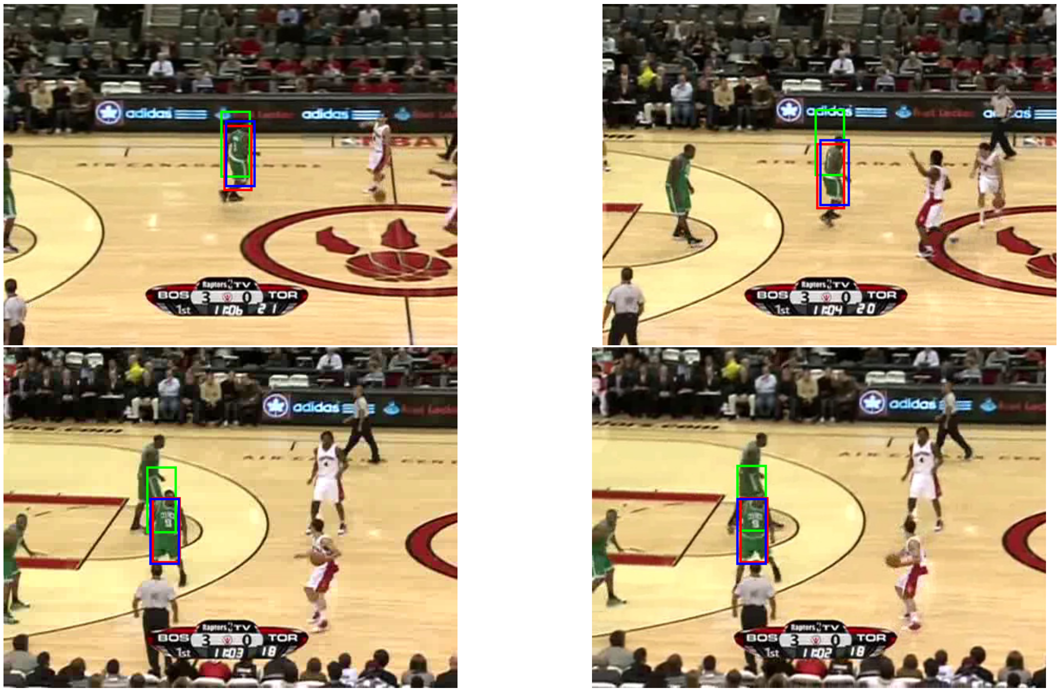


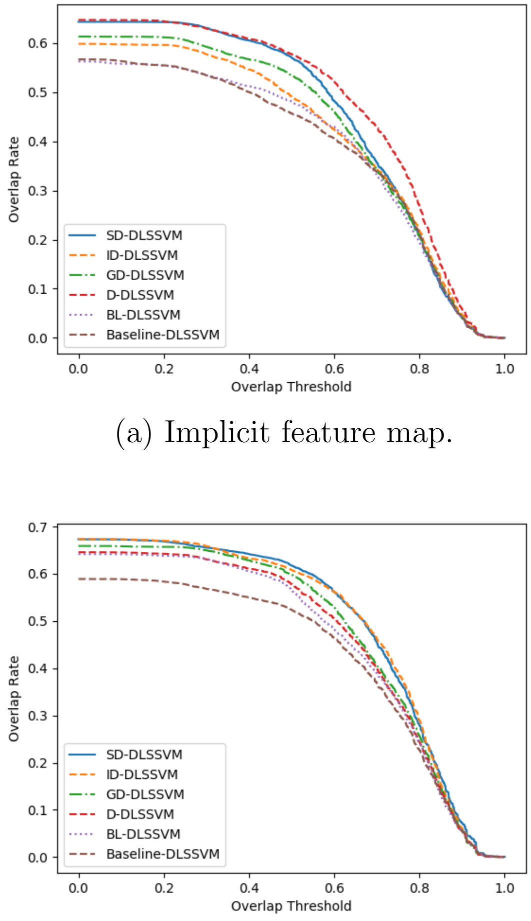
Figure 17: Example result frames with background clutters. Ground truth is presented in a red box, DLSSVM in a green box and SD-DLSSVM in a blue box.

Models	Implicit Map		Explicit Map	
	succ. (AUC)	prec. (20 px)	succ. (AUC)	prec. (20 px)
DLSSVM	0.56623	0.84339	0.58851	0.80889
D-DLSSVM	**0.64652**	**0.87154**	0.64516	0.88334
GD-DLSSVM	0.61264	0.86109	0.65845	0.89332
SD-DLSSVM	**0.65360**	**0.89787**	**0.67261**	**0.89877**
BL-DLSSVM	0.53262	0.76305	0.64107	0.89332
ID-DLSSVM	0.59787	0.85974	**0.67283**	**0.89787**

Table 11: Overall success rate for different models.

Video Sequence	**DLSSVM**	**D-DLSSVM**	**GD-DLSSVM**	**SD-DLSSVM**	**BL-DLSSVM**	**ID-DLSSVM**
Boy	0.79485	**0.80310**	**0.80310**	0.80188	0.79308	0.80169
Coke	0.17414	0.49583	0.42922	**0.59307**	0.14004	0.36894
MountainBike	0.71090	0.72449	**0.73177**	0.72548	0.70412	0.70288
Basketball	0.54973	**0.77277**	0.64843	0.74607	0.61806	0.59926
Crossing	0.63019	**0.64392**	0.62155	0.62512	0.62550	0.62322
Couple	0.61497	0.48930	0.48635	0.50662	0.50667	**0.62224**
Football1	**0.80952**	0.77635	0.69756	0.69756	0.78999	0.77714
Walking2	0.44763	0.43537	**0.45698**	0.44716	0.42436	0.42916
Mean FPS	26.82	21.43	10.35	23.96	23.97	21.68
Boy-U	**0.80983**	0.80293	0.80088	0.80421	0.79573	0.80279
Coke-U	0.60939	0.53439	0.52456	0.61775	0.53914	**0.65905**
MountainBike-U	0.72169	0.73133	0.72690	**0.73288**	0.73005	0.72549
Basketball-U	0.39316	0.72946	0.74408	**0.78793**	0.70853	0.76099
Crossing-U	0.67833	0.68711	0.67771	**0.69345**	0.61829	0.67257
Couple-U	**0.63369**	0.50555	0.51079	0.62394	0.50256	0.59659
Football1-U	0.59162	0.74941	**0.82721**	0.63219	0.80647	0.78024
Walking2-U	0.42850	0.43165	**0.48082**	0.43234	0.44724	0.42857
Mean FPS	29.08	18.35	8.67	13.64	19.33	18.19

Table 12: Individual success rate of eight video sequences for different models.



(a) Implicit feature map.

(b) Explicit feature map.

Figure 18: Average success plot for five different video sequences.

5.4.2 Analysis of the proposed feature mapping

Visually, the performance of one of our proposed method SD-DLSSVM compared to DLSSVM is presented in Fig. 18. The top row shows example frames in which DLSSVM is affected by background noise, while SD-DLSSVM is invariant to it. In the bottom row, our proposed method is still invariant in the presence of similar objects in the background. The different characteristics of each of the distributions discussed in Section 5.2 improves the tracking performance in different situations where the traditional method fails to track target object. Since there is peculiarity in each video, the tracker performance will vary. Hence, the overall success rate is measured to get the best model instead of comparing performance for each video sequences [113, 119]. Table 1 presents the overall success score at threshold of 50% overlap rate and precision score at threshold of 20 pixels of our proposed method in comparison to DLSSVM in our selected video sequences. For density based or implicit feature mapping, SD distribution based mapping achieves highest (in blue) overall success rate is 65.36% with a precision rate of 89.78%. Second best model (in red) is Dirichlet DLSSVM (D-DLSSVM) having a success rate of 64.65. As shown in Table 1, using explicit feature mapping, overall performances of all the models have increased. Inverted Dirichlet (ID) has the highest score of 67.28% by a close margin to SD based model. For better comparison of different models, success rate for each video sequences with corresponding models are presented in Table 2.

Significant gain is observed in the Basketball and Coke video sequences with density based feature mapping. With our experimental setup, tracking performance of DLSSVM for Coke video sequences is very low due to fast motion and illumination variation. SD distribution based feature mapping enables us to improve the performance by almost 42% with the same setup. For explicit feature mapping with unary representation, Baketball-U, Coke-U and Football-U sequences show major improvement in success rate score. Fig.18 shows the Area Under Curve (AUC) ranking scores of baseline DLSSVM and our proposed generative feature mapping with different distributions. It is evident that for explicit feature representations, our proposed method with all distributions outperforms baseline tracker. Average frame rate per second (FPS) is higher than the baseline DLSSVM for all models. This is because new features are inferred from probability density equations using the learned parameters from initial frame for each frame and for each of the images patches. For explicit feature mapping, the mean FPS of our proposed models are significantly lower than the baseline since we have twice the number of features for each patch according to Eq.(114). Compared to state of the art deep learning based trackers such as [17, 66] where convolutional features are learned by training the model offline on a large dataset, our approach is computationally cheaper and the features are generated online in later sequences. Such flexible representation is desirable compared to deep learning approaches where the model tries to find static features that has been learned in the training phase.

Chapter 6

Conclusion

In this thesis, we have developed several feature mapping functions in order to improve the accuracy of SVM learning algorithm.

Chapter 2 shows a novel feature mapping technique for proportional data based on Dirichlet, generalized Dirichlet and Beta-Liouville distributions which shows good accuracy in classifying images and videos. Such data types are prevalent in data mining, image processing and pattern recognition problems which motivated us to exploit the statistical representation of the data in order to enhance the discriminative power of the traditional SVM kernels. In particular, we have introduced five feature mapping functions for proportional data to be used in SVM learning algorithm. Our experiments on DSVM, GDSVM and BLSVM show good performance of the proposed technique in classifying natural and satellite images and also in classifying human action recognition in videos. The results also show that either of the proposed distribution based feature mapping function increases the accuracy of the corresponding SVM kernel. By using scaled Dirichlet and shifted scaled Distributions with SVM classifiers which we name as SDSVM and SSDSVM, we can improve the accuracy of gender classification and facial expression recognition. These distributions have the same computational complexity to update the free parameters as Dirichlet distribution since a closed form solution to new parameters exist. Experimental results show that the proposed method performs favorably for gender classification and emotion recognition against the baseline SVM kernels.

Inc chapter 3, we have introduced a distribution based feature mapping for semi bounded positive vectors. By taking the advantage of inverted Dirichlet, generalized inverted Dirichlet and inverted Beta Liouville distributions, we arrived at three different feature mapping functions that can be combined with other traditional kernels. In addition, we have presented a framework to update parameters using Newton-Raphson method with an initialization of the parameters utilizing method of moments. We have shown empirically that the combination of our proposed approach with traditional kernel functions favors the improvement of baseline kernel accuracy. According to our results, inverted Beta Liouville distribution has better feature modelling capabilities for classification in general. Howerver, these experiments are highly dependent on similarity matrix and inverted Dirichlet and generalized inverted Dirichlet distributions have proved to be efficient for some kernels. Indeed, the performance of the models can be improved by considering feature weights such TF-IDF. Since the accuracy of the proposed technique is dependent on the learned parameters, convergence to local maxima in Newton-Raphson method may result in less accurate model.

In chapter 4, we presented a new feature mapping technique for count data exploiting the characteristics of Dirichlet Multinomial and Generalized Dirichlet Multinomial distributions. In addition, mentioned parameter estimation technique guarantees optimal value of the parameters and hence can confidently be used in the mapping function. Our experimental results show that the proposed method is capable to increase the accuracy of the classifier. It is noteworthy that, this mapping technique can be used with any count data modelling approach for classification task and depending on the kernel function and nature of the data, either DM SVM or GDM SVM beats the baseline SVM classifier.

In chapter 5, we have proposed an efficient feature mapping technique for object tracking. By utilizing five different distributions for proportional data, we show that features based on probability simplex is an effective feature for efficient object tracking in severe noise conditions. We further improve the baseline tracker performance by approximating the non-linear intersection kernel. Experimental results show that the combination of color pixels and their corresponding simplex based features favors efficient object tracking.

Since histogram based encoding approach quantizes all feature information into a fixed number of bins, some discriminative information might be lost. An important future research direction can be to consider soft encoding method such mixture models. Also, instead of using a fixed number of bins, variational approaches can be taken to determine the number of clusters that describes the data most. Also, as indicated by [82] piecewise non-linear mapping function based on all the distributions discussed in this thesis can be developed. From application point of view, our proposed method can be applied for node classification in graph for histogram based representation of adjacency matrix.

Bibliography

[1] Al Mashrgy, M., Bdiri, T., and Bouguila, N. Robust simultaneous positive data clustering and unsupervised feature selection using generalized inverted dirichlet mixture models. *Knowledge-Based Systems 59* (2014), 182–195.

[2] Alsuroji, R., Bouguila, N., and Zamzami, N. Predicting defect-prone software modules using shifted-scaled dirichlet distribution. In *2018 First International Conference on Artificial Intelligence for Industries (AI4I)* (2018), IEEE, pp. 15–18.

[3] Alsuroji, R., Zamzami, N., and Bouguila, N. Model selection and estimation of a finite shifted-scaled dirichlet mixture model. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2018), IEEE, pp. 707–713.

[4] Anastasiu, D. C., and Karypis, G. Efficient identification of tanimoto nearest neighbors. *International Journal of Data Science and Analytics 4*, 3 (2017), 153–172.

[5] Avidan, S. Support vector tracking. *IEEE transactions on pattern analysis and machine intelligence 26*, 8 (2004), 1064–1072.

[6] Avidan, S. Ensemble tracking. *IEEE transactions on pattern analysis and machine intelligence 29*, 2 (2007), 261–271.

[7] Babenko, B., Yang, M.-H., and Belongie, S. Robust object tracking with online multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence 33*, 8 (2010), 1619–1632.

[8] Barla, A., Odone, F., and Verri, A. Histogram intersection kernel for image classification. In *Proceedings 2003 international conference on image processing (Cat. No. 03CH37429)* (2003), vol. 3, IEEE, pp. III–513.

[9] Basak, J. A least square kernel machine with box constraints. *Journal of Pattern Recognition Research 5*, 1 (2010), 38–51.

[10] Bay, H., Tuytelaars, T., and Van Gool, L. Surf: Speeded up robust features. In *European conference on computer vision* (2006), Springer, pp. 404–417.

[11] Bdiri, T., and Bouguila, N. An infinite mixture of inverted dirichlet distributions. In *International Conference on Neural Information Processing* (2011), Springer, pp. 71–78.

[12] Bdiri, T., and Bouguila, N. Learning inverted dirichlet mixtures for positive data clustering. In *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing* (2011), Springer, pp. 265–272.

[13] Bdiri, T., and Bouguila, N. Positive vectors clustering using inverted dirichlet finite mixture models. *Expert Systems with Applications 39*, 2 (2012), 1869–1882.

[14] Bdiri, T., and Bouguila, N. Bayesian learning of inverted dirichlet mixtures for svm kernels generation. *Neural Computing and Applications 23*, 5 (2013), 1443–1458.

[15] Bdiri, T., Bouguila, N., and Ziou, D. Variational bayesian inference for infinite generalized inverted dirichlet mixtures with feature selection and its application to clustering. *Applied Intelligence 44*, 3 (2016), 507–525.

[16] Bekios-Calfa, J., Buenaposada, J. M., and Baumela, L. Revisiting linear discriminant techniques in gender recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 33*, 4 (2010), 858–864.

[17] Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., and Torr, P. H. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision* (2016), Springer, pp. 850–865.

[18] Boiy, E., and Moens, M.-F. A machine learning approach to sentiment analysis in multilingual web texts. *Information retrieval 12*, 5 (2009), 526–558.

[19] Boughorbel, S., Tarel, J.-P., and Boujemaa, N. Generalized histogram intersection kernel for image recognition. In *IEEE International Conference on Image Processing 2005* (2005), vol. 3, IEEE, pp. III–161.

[20] Boughorbel, S., Tarel, J.-P., and Fleuret, F. Non-mercer kernels for svm object recognition. In *BMVC* (2004), pp. 1–10.

[21] Bouguila, N. A model-based approach for discrete data clustering and feature weighting using MAP and stochastic complexity. *IEEE Trans. Knowl. Data Eng. 21*, 12 (2009), 1649–1664.

[22] Bouguila, N. Count data modeling and classification using finite mixtures of distributions. *IEEE Transactions on Neural Networks 22*, 2 (2010), 186–198.

[23] Bouguila, N. Bayesian hybrid generative discriminative learning based on finite liouville mixture models. *Pattern Recognition 44*, 6 (2011), 1183–1200.

[24] Bouguila, N. Count data clustering using unsupervised localized feature selection and outliers rejection. In *IEEE 23rd International Conference on Tools with Artificial Intelligence, ICTAI 2011, Boca Raton, FL, USA, November 7-9, 2011* (2011), IEEE Computer Society, pp. 1020–1027.

[25] Bouguila, N. Hybrid generative/discriminative approaches for proportional data modeling and classification. *IEEE Transactions on Knowledge and Data Engineering 24*, 12 (2011), 2184–2202.

[26] Bouguila, N. Hybrid generative/discriminative approaches for proportional data modeling and classification. *IEEE Transactions on Knowledge and Data Engineering 24*, 12 (Dec 2012), 2184–2202.

[27] Bouguila, N., and Amayri, O. A discrete mixture-based kernel for svms: Application to spam and image categorization. *Inf. Process. Manag. 45*, 6 (2009), 631–642.

[28] Bouguila, N., and ElGuebaly, W. On discrete data clustering. In *Advances in Knowledge Discovery and Data Mining, 12th Pacific-Asia Conference,*

PAKDD 2008, Osaka, Japan, May 20-23, 2008 Proceedings (2008), vol. 5012 of *Lecture Notes in Computer Science*, Springer, pp. 503–510.

[29] Bouguila, N., and ElGuebaly, W. Discrete data clustering using finite mixture models. *Pattern Recognit. 42*, 1 (2009), 33–42.

[30] Bouguila, N., and Ghimire, M. N. Discrete visual features modeling via leave-one-out likelihood estimation and applications. *J. Visual Communication and Image Representation 21*, 7 (2010), 613–626.

[31] Bouguila, N., and Ziou, D. Mml-based approach for finite dirichlet mixture estimation and selection. In *Machine Learning and Data Mining in Pattern Recognition, 4th International Conference, MLDM 2005, Leipzig, Germany, July 9-11, 2005, Proceedings* (2005), vol. 3587 of *Lecture Notes in Computer Science*, Springer, pp. 42–51.

[32] Bouguila, N., and Ziou, D. Mml-based approach for high-dimensional unsupervised learning using the generalized dirichlet mixture. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2005, San Diego, CA, USA, 21-23 September, 2005* (2005), IEEE Computer Society, p. 53.

[33] Bourouis, S., Zaguia, A., and Bouguila, N. Hybrid statistical framework for diabetic retinopathy detection. In *International Conference Image Analysis and Recognition* (2018), Springer, pp. 687–694.

[34] Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[35] Burges, C. J. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery 2*, 2 (Jun 1998), 121–167.

[36] Čehovin, L., Leonardis, A., and Kristan, M. Visual object tracking performance measures revisited. *IEEE Transactions on Image Processing 25*, 3 (2016), 1261–1274.

[37] Chang, C.-C., and Lin, C.-J. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST) 2*, 3 (2011), 1–27.

[38] Chapelle, O. Training a support vector machine in the primal. *Neural computation 19*, 5 (2007), 1155–1178.

[39] Chapelle, O., Haffner, P., and Vapnik, V. N. Support vector machines for histogram-based image classification. *Trans. Neur. Netw. 10*, 5 (Sept. 1999), 1055–1064.

[40] Chatfield, K., Lempitsky, V. S., Vedaldi, A., and Zisserman, A. The devil is in the details: an evaluation of recent feature encoding methods.

[41] Chenoweth, M. E., and Sarra, S. A. A numerical study of generalized multiquadric radial basis function interpolation.

[42] Comaniciu, D., Ramesh, V., and Meer, P. Kernel-based object tracking. *IEEE Transactions on pattern analysis and machine intelligence 25*, 5 (2003), 564–577.

[43] Comaniciu, D., Ramesh, V., Meer, P., Member, S., and Member, S. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence 25* (2003), 564–577.

[44] Connor, R. J., and Mosimann, J. E. Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association 64*, 325 (1969), 194–206.

[45] Connor, R. J., and Mosimann, J. E. Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association 64*, 325 (1969), 194–206.

[46] Dai, D., and Yang, W. Satellite image classification via two-layer sparse coding with biased image representation. *IEEE Geoscience and Remote Sensing Letters 8*, 1 (2010), 173–176.

[47] Dalal, N., and Triggs, B. Histograms of oriented gradients for human detection. In *international Conference on computer vision & Pattern Recognition (CVPR'05)* (2005), vol. 1, IEEE Computer Society, pp. 886–893.

[48] Farnebäck, G. Two-frame motion estimation based on polynomial expansion. In *Scandinavian conference on Image analysis* (2003), Springer, pp. 363–370.

[49] Fei-Fei, L., and Perona, P. A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (2005), vol. 2, IEEE, pp. 524–531.

[50] Fligner, M. A., Verducci, J. S., and Blower, P. E. A modification of the jaccard–tanimoto similarity index for diverse selection of chemical compounds using binary strings. *Technometrics 44*, 2 (2002), 110–119.

[51] Fritz, M., Hayman, E., Caputo, B., and Eklundh, J.-O. The kth-tips database.

[52] Fukunaga, K. *Introduction to Statistical Pattern Recognition (Second Edition)*, second edition ed. Academic Press, Boston, 1990.

[53] Gill, J., and King, G. *Numerical Issues Involved in Inverting Hessian Matrices.* John Wiley and Sons, Inc., Hoboken, NJ, 2003, ch. 6, pp. 143–176.

[54] Group, C. V. http://www.vision.caltech.edu/html-files/archive.html.

[55] Gunn, S. R., et al. Support vector machines for classification and regression. *ISIS technical report 14*, 1 (1998), 5–16.

[56] Guyon, I., and Elisseeff, A. An introduction to variable and feature selection. *Journal of machine learning research 3*, Mar (2003), 1157–1182.

[57] Han, J., Pei, J., and Kamber, M. *Data mining: concepts and techniques.* Elsevier, 2011.

[58] Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.-M., Hicks, S. L., and Torr, P. H. Struck: Structured output tracking with kernels. *IEEE transactions on pattern analysis and machine intelligence 38*, 10 (2015), 2096–2109.

[59] Herbster, M. Learning additive models online with fast evaluating kernels. In *International Conference on Computational Learning Theory* (2001), Springer, pp. 444–460.

[60] Hu, C., Fan, W., Du, J.-X., and Bouguila, N. A novel statistical approach for clustering positive data based on finite inverted beta-liouville mixture models. *Neurocomputing 333* (2019), 110–123.

[61] Hu, W., Xie, N., Hu, R., Ling, H., Chen, Q., Yan, S., and Maybank, S. Bin ratio-based histogram distances and their application to image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence 36*, 12 (Dec 2014), 2338–2352.

[62] Huang, J. Maximum likelihood estimation of dirichlet distribution parameters.

[63] Huang, S., Jiang, S., and Zhu, X. Multi-object tracking via discriminative appearance modeling. *Computer Vision and Image Understanding 153* (2016), 77 – 87. Special issue on Visual Tracking.

[64] Javaran, H., and Khaji, N. Inverse multiquadric (imq) function as radial basis function for plane dynamic analysis using dual reciprocity boundary element method.

[65] Krestinskaya, O., and James, A. P. Facial emotion recognition using min-max similarity classifier. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (2017), IEEE, pp. 752–758.

[66] Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Pflugfelder, R., Kamarainen, J.-K., Cehovin Zajc, L., Drbohlav, O., Lukezic, A., Berg, A., et al. The seventh visual object tracking vot2019 challenge results. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (2019), pp. 0–0.

[67] Laptev, I., Caputo, B., et al. Recognizing human actions: a local svm approach. In *null* (2004), IEEE, pp. 32–36.

[68] Lazebnik, S., Schmid, C., and Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (2006), vol. 2, IEEE, pp. 2169–2178.

[69] Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., and Hengel, A. V. D. A survey of appearance models in visual object tracking. *ACM transactions on Intelligent Systems and Technology (TIST) 4*, 4 (2013), 1–48.

[70] Li, X., Wang, L., and Sung, E. Adaboost with svm-based component classifiers. *Engineering Applications of Artificial Intelligence 21*, 5 (2008), 785–795.

[71] Lingappaiah, G. On the generalised inverted dirichlet distribution. *Demostratio Mathematica 9*, 3 (1976), 423–433.

[72] Liu, J., Luo, J., and Shah, M. Recognizing realistic actions from videos in the wild. Citeseer.

[73] Lowe, D. G. Distinctive image features from scale-invariant keypoints. *International journal of computer vision 60*, 2 (2004), 91–110.

[74] Lyons, M. J., Akamatsu, S., Kamachi, M., Gyoba, J., and Budynek, J. The japanese female facial expression (jaffe) database.

[75] Maji, S., and Berg, A. C. Max-margin additive classifiers for detection. In *2009 IEEE 12th International Conference on Computer Vision*, IEEE, pp. 40–47.

[76] Maji, S., Berg, A. C., and Malik, J. Classification using intersection kernel support vector machines is efficient. In *2008 IEEE conference on computer vision and pattern recognition* (2008), IEEE, pp. 1–8.

[77] Mallikarjuna, P., Targhi, A. T., Fritz, M., Hayman, E., Caputo, B., and Eklundh, J.-O. The kth-tips2 database.

[78] Miller, K. S. *Some eclectic matrix theory.* Krieger Publishing Company, 1987.

[79] Minka, T. Estimating a dirichlet distribution, 2000.

[80] Mongillo, M. Choosing basis functions and shape parameters for radial basis function methods.

[81] Monti, G. S., Mateu-Figueras, G., and Pawlowsky-Glahn, V. Notes on the scaled dirichlet distribution. *Compositional Data Analysis* (2011), 128–138.

[82] Nedaie, A., and Najafi, A. A. Support vector machine with dirichlet feature mapping. *Neural Networks 98* (2018), 87–101.

[83] Ng, K. W., Tian, G.-L., and Tang, M.-L. *Dirichlet and related distributions: Theory, methods and applications*, vol. 888. John Wiley & Sons, 2011.

[84] Ng, K. W., Tian, G.-L., and Tang, M.-L. *Dirichlet and Related Distributions: Theory, Methods and Applications.* Wiley Series in Probability and Statistics. John Wiley Sons, Hoboken, NJ, 2011.

[85] Ning, J., Yang, J., Jiang, S., Zhang, L., and Yang, M.-H. Object tracking via dual linear structured svm and explicit feature map. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 4266–4274.

[86] Oboh, B. S., and Bouguila, N. Unsupervised learning of finite mixtures using scaled dirichlet distribution and its application to software modules categorization. In *IEEE International Conference on Industrial Technology, ICIT 2017, Toronto, ON, Canada, March 22-25, 2017* (2017), IEEE, pp. 1085–1090.

[87] Olshausen, B. A., and Field, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature 381*, 6583 (1996), 607.

[88] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[89] Pele, O., Taskar, B., Globerson, A., and Werman, M. The pairwise piecewise-linear embedding for efficient non-linear classification. In *International Conference on Machine Learning* (2013), pp. 205–213.

[90] Poursaberi, A., Noubari, H. A., Gavrilova, M., and Yanushkevich, S. N. Gauss–laguerre wavelet textural feature fusion with geometrical information for facial expression identification. *EURASIP Journal on Image and Video Processing 2012*, 1 (2012), 17.

[91] Rahman, M. H., Ali, S., and Bouguila, N. Parametric features on simplex manifold for online object tracking. In *IEEE International Conference on Image Processing* (2020). Manuscript submitted.

[92] Rahman, M. H., and Bouguila, N. Distribution based feature mapping for classifying count data. In *IEEE International Conference on Computational Intelligence and Data Mining (CIDM)* (December 2019).

[93] Rahman, M. H., and Bouguila, N. Efficient feature mapping for classifying proportional data. *Neurocomputing* (2019). Manuscript submitted for publication.

[94] Rahman, M. H., and Bouguila, N. Inverted dirichlet and related distributions based feature mapping for data classification. In *IEEE International Conference on Systems, Man and, Cybernetics* (2020). Manuscript submitted.

[95] Rahman, M. H., and Bouguila, N. Probabilistic features on simplex manifold in predictive data modelling. In *IEEE International Symposium on Networks, Computers and Communications* (2020). Manuscript submitted.

[96] Ralaivola, L., Swamidass, S. J., Saigo, H., and Baldi, P. Graph kernels for chemical informatics. *Neural networks 18*, 8 (2005), 1093–1110.

[97] Rasmussen, C. E. Gaussian processes in machine learning. In *Summer School on Machine Learning* (2003), Springer, pp. 63–71.

[98] Ratliff, N. D., Bagnell, J. A., and Zinkevich, M. A. (approximate) subgradient methods for structured prediction. In *Artificial Intelligence and Statistics* (2007), pp. 380–387.

[99] Rothe, R., Timofte, R., and Gool, L. V. Dex: Deep expectation of apparent age from a single image. In *IEEE International Conference on Computer Vision Workshops (ICCVW)* (December 2015).

[100] Roychowdhury, S., and Emmons, M. A survey of the trends in facial and expression recognition databases and methods. *arXiv preprint arXiv:1511.02407* (2015).

[101] Rubner, Y., Tomasi, C., and Guibas, L. J. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision 40*, 2 (Nov. 2000), 99–121.

[102] Rychlý, P. A lexicographer-friendly association score. In *RASLAN* (2008), pp. 6–9.

[103] Scholkopf, B., and Smola, A. J. *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT press, 2001.

[104] Shih, F. Y., Chuang, C.-F., and Wang, P. S. Performance comparisons of facial expression recognition in jaffe database. *International Journal of Pattern Recognition and Artificial Intelligence 22*, 03 (2008), 445–459.

[105] Singh, J. P., and Bouguila, N. Proportional data clustering using k-means algorithm: A comparison of different distances. In *2017 IEEE International Conference on Industrial Technology (ICIT)* (2017), IEEE, pp. 1048–1052.

[106] Sklar, M. Fast mle computation for the dirichlet multinomial. *arXiv preprint arXiv:1405.0099* (2014).

[107] Su, Z., Li, J., Chang, J. W., Du, B., and Xiao, Y. Real-time visual tracking using complementary kernel support correlation filters. *Frontiers of Computer Science 14* (2019), 417–429.

[108] Tong, S., and Koller, D. Support vector machine active learning with applications to text classification. *Journal of machine learning research 2*, Nov (2001), 45–66.

[109] Vapnik, V. *The nature of statistical learning theory.* Springer science & business media, 2013.

[110] Vedaldi, A., and Zisserman, A. Efficient additive kernels via explicit feature maps. *IEEE transactions on pattern analysis and machine intelligence 34*, 3 (2012), 480–492.

[111] Wahba, G. *Spline models for observational data*, vol. 59. Siam, 1990.

[112] Welch, G., Bishop, G., et al. An introduction to the kalman filter.

[113] Wen, L., Cai, Z., Lei, Z., Yi, D., and Li, S. Z. Online spatio-temporal structural context learning for visual tracking. In *European conference on computer vision* (2012), Springer, pp. 716–729.

[114] Wicker, N., Muller, J., Kalathur, R. K. R., and Poch, O. A maximum likelihood approximation method for dirichlet's parameter estimation. *Computational statistics & data analysis 52*, 3 (2008), 1315–1322.

[115] Wong, T.-T. Generalized dirichlet distribution in bayesian analysis. *Applied Mathematics and Computation 97*, 2-3 (1998), 165–181.

[116] Wu, Y., Lim, J., and Yang, M.-H. Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2013), pp. 2411–2418.

[117] Xiao, J., Ehinger, K. A., Hays, J., Torralba, A., and Oliva, A. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision 119*, 1 (2016), 3–22.

[118] Yassaee, H. Inverted dirichlet distribution and multivariate logistic distribution. *Canadian Journal of Statistics 2*, 1-2 (1974), 99–105.

[119] Zhang, J., Ma, S., and Sclaroff, S. Meem: robust tracking via multiple experts using entropy minimization. In *European conference on computer vision* (2014), Springer, pp. 188–203.

[120] Zhang, Y.-D., Yang, Z.-J., Lu, H.-M., Zhou, X.-X., Phillips, P., Liu, Q.-M., and Wang, S.-H. Facial emotion recognition based on biorthogonal wavelet entropy, fuzzy support vector machine, and stratified cross validation. *IEEE Access 4* (2016), 8375–8385.

[121] Zhou, H., and Zhang, Y. Em vs mm: A case study. *Computational Statistics and Data Analysis 56*, 12 (2012), 3909 – 3920.