Bayesian Nonparametric Clustering for Positive Definite Matrices

Anoop Cherian Vassilios Morellas Nikolaos Papanikolopoulos

Abstract—Symmetric Positive Definite (SPD) matrices emerge as data descriptors in several applications of computer vision such as object tracking, texture recognition, and diffusion tensor imaging. Clustering these data matrices forms an integral part of these applications, for which soft-clustering algorithms (K-Means, Expectation Maximization, etc.) are generally used. As is well-known, these algorithms need the number of clusters to be specified, which is difficult when the dataset scales. To address this issue, we resort to the classical nonparametric Bayesian framework by modeling the data as a mixture model using the Dirichlet process (DP) prior. Since these matrices do not conform to the Euclidean geometry, rather belongs to a curved Riemannian manifold,existing DP models cannot be directly applied. Thus, in this paper, we propose a novel DP mixture model framework for SPD matrices. Using the log-determinant divergence as the underlying dissimilarity measure to compare these matrices, and further using the connection between this measure and the Wishart distribution, we derive a novel DPM model based on the Wishart-Inverse-Wishart conjugate pair. We apply this model to several applications in computer vision. Our experiments demonstrate that our model is scalable to the dataset size and at the same time achieves superior accuracy compared to several state-of-the-art parametric and nonparametric clustering algorithms.

Index Terms—Region covariances, Dirichlet process, nonparametric methods, positive definite matrices

1 Introduction

Recent years have witnessed an increasing trend in computer vision and machine learning applications that use rich data representations such as strings, graphs, matrices, etc. instead of the traditional vectorial data types. Among these one important class of structured data descriptors that is gaining popularity in computer vision is the class of symmetric positive definite (SPD) matrices, especially in the form of feature covariances dubbed Region Covariance Descriptors (RCD) [52].

Compared to traditional vector based descriptors (such as histograms, feature vectors, etc.), the structure of the covariance matrix offers several useful properties. The dimensionality of the RCD is independent of the number of data points used in its construction. As a result, RCDs offer a convenient platform for fusing multiple features into a compact form when the data dimensionality is less compared to the number of data points. This fusion offers several advantages. For example, RCDs can be made invariant to image affine distortions by choosing appropriate feature representations [46]. In addition, since the feature mean is subtracted off when computing RCDs, they are relatively robust to static noise and illumination variations in the image. From a practical standpoint, RCDs can be computed very efficiently using integral images. Due to these advantages, they are finding an increasing number of applications in computer vision, such as in people appearance tracking for visual surveillance [45], [51], [66], face recognition [53], object recognition [57], and action recognition [28], to name a few. SPD matrices also play an important role as data descriptors in several other applications, such as image set classification [68], diffusion tensor imaging [1], sound compression [60], polarimetric image modeling [24], and as quantum density matrices [21].

Clustering data is an important algorithmic ingredient in several applications. Unfortunately, clustering RCDs is not straightforward. This is because, although covariance matrices form a sub-manifold of the Euclidean space, it is generally found that assuming a curved manifold structure on them is advantageous [55]. As a result, covariances are generally assumed to be elements of the open cone of symmetric positive definite matrices, adhering to a Riemannian geometry dictated by an appropriate Riemannian metric [25]. Thus, clustering algorithms developed for covariances are expected to adhere to this geometry. The centroid of RCDs can be computed using the Karcher mean algorithm [55], thus suggesting the viability of a K-means type clustering [51] scheme on the Riemannian manifold.

There have been also been clustering approaches proposed for these matrices from statistical viewpoint. A soft-clustering algorithm via expectation maximization is proposed in [32] which models the clusters as a mixture of Wishart distributions. Since the number of components in the clustering model needs to be specified in these soft-clustering algorithms, they are not scalable to real-world scenarios where this number might change over time (for example, clustering appearances of people in a camera surveillance network). This motivates us to investigate unsupervised models in which the number of

• *A. Cherian is with the Australian Center for Robotic Vision at the Australian National University, Canberra. His emailid is anoop.cherian@anu.edu.au*
• *V. Morellas and N. Papanikolopoulos are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis. Their emailids are {morellas, npapas}@cs.umn.edu.*

clusters is dynamically updated monotonically according to the complexity of an ever increasing volume of data.

We approach the unsupervised clustering problem from a nonparametric Bayesian perspective and resort to the classical Dirichlet Process Mixture Model (DPMM) framework [3], [20]. Existing DPM models are designed for vector valued data, thus posing a major difficulty when working with matrix valued objects that have their own specific geometry; this difficulty we circumvent using our novel model. The first step in developing Bayesian framework is to define the probability measure on the underlying data that captures its structure effectively. Some of the well-known statistical measures on SPD matrices are the matrix Frobenius norm, log-Euclidean metric and the log-determinant divergence. The first two measures can be used to embed the data matrices into the Euclidean space, while the third measure operates directly in the matrix space. Since a Euclidean embedding may distort the structure of the data, we primarily focus on the log-determinant divergence measure for developing our DPM model.

The next step is to find an appropriate density function that models the distribution of the data points. Fortunately, the logdet (short for log-determinant) divergence belongs to a special class of information divergences called Bregman divergences [16] which are convex functions and have strong connections with the exponential family distributions. Utilizing the so called Bregman-Exponential bijection, we derive a probabilistic density function on covariances, which turns out to be the Wishart distribution. Using this distribution and its conjugate pair (which is the Inverse-Wishart distribution), we derive the necessary expressions for sampling from a DP mixture model. To accommodate the difficult inference problem when using the DP prior, we use a collapsed Gibbs sampler. Experiments are presented on a variety of applications from computer vision and demonstrate the superior performance of our method against the state-of-the-art methods.[1]

Before we proceed, we will briefly introduce our notations. We use $|X|$ to denote the determinant of a matrix X and $\mathrm{Tr}(X)$ denotes the matrix trace. We use $abs(x)$ to denote the absolute value of a vector x and $\|\ \|_{\mathrm{F}}$ the Frobenius norm. We use upper case for matrices and lower case for vectors.

2 Related Work

To set the stage for our discussion on nonparametric clustering of positive definite matrices, we will review some relevant literature in this section. First, let us formally define an RCD.

Definition 1. *Let $\mathbf{x}_i \in \mathbb{R}^d$, for $i = 1, 2, \cdots, m$, be the feature vectors from the region of interest of an image, then the Covariance Descriptor of this region $X_c \in \mathcal{S}_{++}^d$ is defined as:*

$$X_c = \frac{1}{m-1} \sum_{i=1}^{m} (\mathbf{x}_i - \mu_{\mathbf{x}})(\mathbf{x}_i - \mu_{\mathbf{x}})^T, \tag{1}$$

where $\mu_{\mathbf{x}} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i$ is the mean feature vector and $\mathcal{S}_{++}^d$ denotes the space of all $d \times d$ SPD matrices.

Matrix valued data appears in several contexts in machine learning, such as Grassmannian manifolds, Homogeneous subspaces, fundamental matrices, etc. of which we will restrict our literature review to algorithms designed for SPD matrices. As we mentioned in the last section, the standard vectorial clustering algorithms such as K-Means, Expectation Maximization (EM), Spectral Clustering, etc. have been extended to deal with SPD matrices in the past. The main challenge in these algorithms is to make sure that the data similarity is computed adhering to the manifold topology and the cluster means lie on the manifold. Towards this end, the classical K-Means clustering has been modified to deal with SPD matrices using the Karcher mean algorithm [8]. This algorithm minimizes the sum of the squared geodesic distances of the data points from the cluster centroids. Usually, the affine invariant Riemannian metric [55] or the Log-Euclidean Riemannian metric [4] are used for computing the distances between SPD matrices. K-Means using the former metric is known to converge when all the points in a cluster lie within the injectvity radius[2] of the manifold. On the other hand, the Log-Euclidean metric maps the SPD matrices to a flat Euclidean space which is isomorphic and diffeomorphic to the positive definite cone of SPD matrices. Spectral clustering algorithms can be extended to SPD objects by embedding the data points into a Euclidean space using a similarity matrix computed using the Riemannian metric. A major difficulty with finite mixture models is the selection of the number of mixture components such that the final model does not result in overfitting or underfitting the data.

There have been several unsupervised approaches proposed for clustering SPD valued data that extend the standard kernel density estimation (KDE) [31], [54]. The classical mean-shift algorithm for vector valued data has been extended in [62] using one of the above Riemannian metrics. An issue with KDE and mean-shift extensions is their sensitivity to bandwidth selection. Nonparametric clustering of SPD objects via embedding them in the Euclidean space is considered in [26]. These algorithms extend methods such as Laplacian Eigenmaps [7], Locally Linear Embedding [56], etc. to SPD objects. The main idea of these approaches is to use the connection between matrix exponential and logarithmic maps to project data points onto the manifold tangent space, followed by data embedding. Often these matrix operations are computationally expensive.

1. The current paper is an extension of the conference paper [14] and differs in the following ways: (i) we show detailed derivations of the model and (ii) apply it to several real-world computer vision problems.

2. That is, the largest distance for which the exponential map of a point on the Riemannian manifold is diffeomorphic.

A Bayesian treatment to nonparametric problems predominantly uses the Dirichlet Process (DP) prior, introduced in [22], and later extended to mixture models in [3], [20], [23]. The main advantage of the Bayesian formalism is its ability to model the underlying stochastic process that might have generated the data. Analogous to finite mixture models, DPMM allows for each cluster to have its own size and parameters. In addition to this advantage, DPMMs allow for the possibility of an infinite number of clusters, thus circumventing the problem of choosing the number of clusters that may increase or decrease when more data is available. The main idea of a Dirichlet process is often explained using the Chinese restaurant process metaphor [27], which considers the clustering process as that of a seating arrangement of customers in a Chinese restaurant. Assume that an infinite number of tables is possible in a Chinese restaurant. A new customer entering the restaurant has two choices; he may choose to sit on a table where other customers are sitting with a probability proportional to the existing number of customers at that table, or he may choose to sit on a new table according to a predefined probability. As new customers enter, the seating arrangement evolving in the Chinese restaurant follows a draw from a Dirichlet process prior. Other interesting interpretations of the DP prior idea can be seen in the stick-breaking constructions [33], and the Polya urn scheme [9]. Customers seated around one table in the restaurant is analogous to a data cluster. Although exact inference of the posterior distribution in the DP framework is challenging, efficient approximate inference algorithms have been suggested [10], [33], [38], [48].

Over the past decade, DPMMs have found immense applications in a variety of domains such as Genomics [71], computer vision [35], [41], [64], computational biology [71], data modeling [11], Diffusion tensor MRI [69], and linguistics [42], to name a few. These applications mainly deal with vector valued data, following normal likelihood distribution. For computational ease, the DP mixture prior is chosen as a conjugate to the data density function, which for these applications is usually the normal-inverse-Wishart or normal-inverse-gamma distributions [65], [72]. Extensions of DPMM for non-Gaussian data has been explored in the past such as to histograms [11], spatio-temporal data [36], and linear dynamical models [13].

In this paper, we consider the problem of clustering SPD matrices in a nonparametric Bayesian framework using DPMMs. As our data points lie on a non-Euclidean manifold, the traditional vectorial approaches for clustering them might not be adequate. To the best of our knowledge, the Dirichlet process framework has never been applied to clustering SPD matrices before.

3 Dirichlet Process Mixture Model

Nonparametric Bayesian techniques seek a predictive model for the data such that the complexity and accuracy of the model grows with the data size. The existence of such a statistical model is invariably dependent on the property of exchangeability [17] of observations, leading to the De Finetti's theorem, which states that if a sequence of observations $y_1, y_2, \cdots, y_n$ is infinitely exchangeable (that is, their joint distribution is invariant to a permutation of their order), then there exists a mixture representation for the joint distribution of these observations. That is,

$$p(y_1, y_2, \cdots, y_n) = \int_{\Theta} p(\theta) \prod_{i=1}^{n} p(y_i|\theta) d\theta \tag{2}$$

where Θ is an infinite-dimensional space of probability measures, $p(\theta)$ denotes the density distribution of θ, and $d\theta$ defines a probability measure over distributions.

A Dirichlet Process Mixture Model, $DP(\alpha, H)$, parameterized by a concentration α and a prior H, is a stochastic process that defines a distribution over probability distributions adhering to Eq. (2) such that if $A_1, A_2, \cdots, A_r$ represent any finite measurable partition of Θ, and if $G \sim DP(\alpha, H)$, then the vector of joint distributions of samples from G over these partitions follow a Dirichlet distribution, $Dir(.)$ [3], [23]. That is,

$$(G(A_1), \cdots, G(A_r)) \sim Dir\left(\alpha H(A_1), \cdots, \alpha H(A_r)\right) \tag{3}$$

In our pertinent problem of finding the number of clusters in the given dataset, we would like to find the distribution G over each of the clusters automatically, by computing the posterior distribution of G given the observations and the prior model H. Fortunately, as is shown in [3], the posterior distribution has the following simple form:

$$\begin{aligned} p\left(G|y_1, \cdots, y_n\right) &\sim Dir\left(\alpha H(A_1) + n_1, \cdots, \alpha H(A_n) + n_r\right) \\ &\sim DP\left(\alpha + n, \frac{1}{\alpha + n}\left(\alpha H + \sum_{i=1}^{n} \delta_{y_i}\right)\right) \end{aligned} \tag{4}$$

where $n_1, n_2, \cdots, n_r$ represent the number of observations falling in each of the partitions $A_1, A_2, \cdots, A_r$ respectively, n is the total number of observations, and δ_{y_i} represents the delta function at the sample point y_i. There are some subtle points to be noted from (4), namely (i) DP acts as a conjugate prior to the distribution over distributions, and (ii) each observation only updates the corresponding component in the Dirichlet distribution. The latter property implies that the underlying distribution is essentially discrete with probability one and is agnostic over the topology of the underlying space in which the data lie. It was shown in [59] that this property leads to a concentration of the posterior distribution towards the mean, leading to a clustering effect.

For the above model to be practically useful and tractable, it is a general practice to model the dependency of the observations y_j to G through a parameterized family $F(\theta_i)$ (where F is the likelihood function

1) Define a likelihood distribution F on data using a suitable distance measure.
2) Find a prior H that is conjugate to F.
3) Model a collapsed Gibbs sampler over the posterior distribution from (1) and (2) as follows:
   - 3a) Remove a data point y from a cluster C and update the sufficient statistics of C.
   - 3b) Compute the predictive distribution $p(y|y_{C_i})$ for each of the existing clusters; y_{C_i} represents data in cluster C_i.
   - 3c) Assuming there are n clusters at this step, create an $n+1$ dimensional vector p_v where n dimensions of p_v correspond to $p(y|y_{C_i})$ for $i = 1, 2, \cdots, n$, while the extra dimension corresponds to the concentration parameter α. Normalize p_v so that it sums to one and thus forms a probability distribution over the clusters, while also incorporating a probability to create a new cluster.
   - 3d) Sample a cluster from p_v and assign y to that cluster, while also updating its sufficient statistics.
   - 3e) Repeat the steps (3a), (3b) (3c), and (3d) until convergence.

Algorithm 1
Overview of DPMM algorithm.

with parameters θ_i) [20], [23] leading to a mixture model characterization of the DP as follows:

$$\begin{aligned} y_j|\theta_i &\sim F(\theta_i) \\ \theta_i|G &\sim G \\ G &\sim DP(\alpha, H) \end{aligned} \tag{5}$$

Since the exact computation of the posterior is infeasible when the data size is large, we resort to a variant of MCMC algorithms, namely, the collapsed Gibbs sampler [43] for faster approximate inference. The discussion in this section and the formulas we seek in the context of covariance matrices are summarized in Algorithm 1.

4 Mathematical Preliminaries

In this section, we review a few important topics that we will be using in the subsequent sections for deriving a suitable DPM model for clustering SPD matrices.

4.1 Bregman Divergence

Bregman divergence [12] is a generalized distance measure over convex functions and has the following general form: Let $\psi : Q \rightarrow \mathbb{R}, (Q \subseteq \mathbb{R}^d)$ be a function of Legendre type in the $relint(Q)$. The Bregman divergence $d_\psi : Q \times relint(Q) \rightarrow [0, \infty)$ is defined as:

$$d_\psi(x, y) = \psi(x) - \psi(y) - \langle x - y, \nabla\psi(y)\rangle \tag{6}$$

where $\nabla\psi(y)$ is the gradient vector of ψ evaluated at y. The squared Euclidean distance, $d_\psi(a, b) = \|a - b\|^2$, is an example of a Bregman divergence corresponding to the convex function $\psi(x) = \frac{1}{2}\|x\|^2$. See [5] for details.

Bregman divergences can be extended to matrices as follows. If X and Y are matrices, and if Ψ is a real-valued, strictly convex function over matrices, then the Bregman matrix divergence can be defined as:

$$D_\Psi(X, Y) = \Psi(X) - \Psi(Y) - \mathrm{Tr}\left(\nabla\Psi(Y)^T(X - Y)\right). \tag{7}$$

For example, when $\Psi(X) = \|X\|_F^2$, then the corresponding Bregman matrix divergence for two matrices A and B is the squared Frobenius norm $\|A - B\|_F^2$. See [15], [18] for other examples.

4.2 Exponential Family

Let Ω be a sample space (discrete or continuous) and let $\Theta \subseteq \mathbb{R}^d$ be an open set of parameters. Let $x \in \Omega$ be a random variable. If $\theta \in \Theta$ and if $p_0 : \Omega \rightarrow \Theta$ is a probability base measure, then a *regular exponential family* is defined as the family of probability distributions of the form:

$$p(x|\theta) = p_0(\phi(x))exp\left\{\langle\theta, \phi(x)\rangle - \eta(\theta)\right\}, \tag{8}$$

where $\phi : \Omega \rightarrow \mathcal{T}$ for $\mathcal{T} \subseteq \mathbb{R}^d$. Here $\phi(x)$ is the *natural statistic* and θ is the *natural parameter*; ϕ is said to be a *sufficient statistic* for the exponential family due to the Fisher-Neyman factorization theorem [49]. $\eta(\theta)$ is the *cumulant function* and normalizes $p(x|\theta)$ so that it integrates to one [5], [63].

4.3 Bregman Divergence-Exponential Family Bijection

A useful property of Bregman divergences is their connection to the exponential family distributions. It has been shown in [5] that there exists a unique Bregman divergence corresponding to every regular exponential family. Thus, if D_ψ is a Bregman divergence associated with the convex function ψ and if ϕ is a conjugate function to ψ, then the regular exponential family, $p_\psi(x|\theta)$, parameterized by θ, can be written as:

$$p_\psi(x|\theta) \propto exp\left\{(-d_\psi(x, \mu(\theta)))\, g_\phi(x)\right\} \tag{9}$$

where $\mu(\theta)$ is the mean of the distribution and $g_\phi(x)$ is a function uniquely determined by ϕ. See [5] for details.

4.4 Wishart Distribution

Let $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m$, $(\mathbf{x}_i \in R^d)$, be independent and identically distributed random vectors such that $\mathbf{x}_i \sim \mathcal{N}(0, \Sigma)$, for $i = 1, 2, \cdots, m$ and let $X \in \mathcal{S}_{++}^d$ such that $X = \sum_{i=1}^m \mathbf{x}_i\mathbf{x}_i^T$. If we define $n = m - 1$, then X is said to follow a non-singular d-dimensional Wishart distribution $W(n, d, \Sigma)$, with n degrees of freedom ($n > d - 1$) and scale matrix Σ if it has a probability density defined by:

$$W(X; n, d, \Sigma) = \frac{1}{\omega(n, d)} \frac{|X|^{(n-d-1)/2}}{|\Sigma|^{n/2}} exp\left\{-\frac{1}{2}\,\mathrm{Tr}(\Sigma^{-1}X)\right\}, \tag{10}$$

where $\omega(n, d)$ is a normalizing constant [19] and has the following form:

$$\omega(n, d) = \int_{\mathcal{S}_{++}^d} |Y|^{(n-d-1)/2} exp\left\{-\frac{1}{2}\,\mathrm{Tr}\,Y\right\} dY \tag{11}$$

$$= \pi^{d(d-1)/4} 2^{nd/2} \prod_{k=1}^{d} \Gamma\left(\frac{n - k + 1}{2}\right), \tag{12}$$

with $\Gamma(.)$ representing the Gamma function.

5 DPMM ON SPD MATRICES

The first step in designing a clustering framework on any datatype is to define the measure that captures the similarity between data objects. There have been several such measures available for SPD objects, such as the affine invariant Riemannian metric [55], the log-Euclidean Riemannian metric [4], etc. of which we will be interested in a statistically motivated similarity measure on covariances, viz. the *log-determinant divergence*. In the following subsections, we will exposit this measure and develop the required density distributions for developing a DPM model using this measure.

5.1 LogDet Divergence

The LogDet divergence D_{ld} (also known as *Stein's loss*) was introduced in [34]. It defines the Kullback-Leibler divergence between two equal-mean Gaussian distributions and has the following form (for $C_1, C_2 \in \mathcal{S}_{++}^d$):

$$D_{ld}(C_1, C_2) = \operatorname{Tr} C_1 C_2^{-1} - \log \left| C_1 C_2^{-1} \right| - d. \tag{13}$$

Unlike other metrics on SPD matrices that we pointed to above, the log-det divergence is not a metric. This is because it is not symmetric (as implied by (13)), and does not satisfy the triangle inequality. Nevertheless, it is a Bregman matrix divergence for the convex function $-log\,|.|$, and has been utilized in several soft clustering algorithms in the recent past [16], [61].

5.2 Log-det Wishart Connection

Utilizing the Bregman-exponential family bijection that we introduced earlier, we can derive the likelihood distribution associated with the log-det divergence. It turns out that this exponential family is the Wishart distribution as stated in the following theorem:

Theorem 1. *Let X_c be the covariance matrix of m iid zero-mean Gaussian-distributed random vectors $\mathbf{x}_i$, i.e. $X_c = \frac{1}{n}\sum_{i=1}^{m} \mathbf{x}_i \mathbf{x}_i{}^T$ where $\mathbf{x}_i \in \mathbb{R}^d$, $\mathbf{x}_i \sim \mathcal{N}(0, \Sigma)$ and $n = m-1$. Then the probability density of X_c follows:*

$$\begin{aligned} p(X_c|\Sigma) &= W(X_c; n, d, \Sigma) \\ &\propto exp\left\{-\frac{1}{2} D_\Psi(\Sigma, X_c)\right\} p_0(X_c), \end{aligned}$$

where D_Ψ is the Bregman matrix divergence function for the convex function $\psi(.) = -n\ log\,|.|$ and p_0 is the base measure.

Proof: Let $X \in \mathcal{S}_{++}^d$ and assume $X = \sum_{i=1}^{m} \mathbf{x}_i \mathbf{x}_i^T$. Thus, $X = nX_c$. From the definition of the Wishart distribution, we have $X \sim W(n, d, \Sigma)$. Substituting for X and rearranging the terms, we get:

$$\begin{aligned} p(X|\Sigma) \propto |X_c|^{-\frac{(d+1)}{2}} exp\Big\{-\frac{n}{2}\big[\operatorname{Tr}\left(\Sigma^{-1} X_c\right) \\ - log|\Sigma^{-1} X_c|\big]\Big\} \end{aligned} \tag{14}$$

$$\propto exp\left\{-D_\Psi(\Sigma, X_c)\right\} |X_c|^{-(d+1)/2}. \tag{15}$$

□

It is well-known in multivariate statistics that for the Wishart distribution $W(n, d, \Sigma)$, the conjugate prior is the Inverse-Wishart distribution parametrized as $IW(n, d, S)$ where $S \in \mathcal{S}_{++}^d$ is the inverse scale matrix, and has the following form:

$$IW(\Sigma; S, n, d) \;=\; \frac{|S|^{\frac{n}{2}} |\Sigma|^{-\frac{(n+d+1)}{2}}}{\omega(n, d)} exp\left(-\frac{1}{2}\operatorname{Tr}(\Sigma^{-1} S)\right). \tag{16}$$

Utilizing this observation, we derive the posterior and the predictive distributions for the Wishart-Inverse-Wishart (WIW) conjugate pair next.

5.3 Posterior and Predictive Distributions

For deriving the posterior distribution $p(\Sigma|X, S, n)$, we will need the marginal distribution $p(X|S, n)$. The following Lemma will come useful in our derivations to follow.

Lemma 2. *Let A be a $d \times d$ nonsingular matrix and define the function g on the linear space of $d \times d$ real symmetric matrices by*

$$g(Z) \;=\; AZA^T = (A \otimes A)Z. \tag{17}$$

Then the Jacobian $J_g(Z) \;=\; abs(|A|)^{d+1}$.

Proof: See [19], Proposition 5.12. □

The marginal distribution can be derived by integrating out the cluster parameter Σ (of each Wishart distribution) from the Wishart-Inverse-Wishart conjugate pair. The following theorem formalizes the expression for the marginal.

Theorem 3 (Marginal)**.** *Given $X \sim W(n, d, \Sigma)$ and $\Sigma \sim IW(S, n, d)$,*

$$p(X|S, n) = \frac{1}{\omega(n, d)^2} |S|^{\frac{n}{2}} |X|^{\frac{(n-d-1)}{2}} \frac{\omega(2n, d)}{|X+S|^n}. \tag{18}$$

Proof: We have

$$\begin{aligned} p(X|S, n) \;&=\; \int_{\mathcal{S}_{++}^d} p(X|\Sigma, n) p(\Sigma|S, n) d\Sigma \qquad (19) \\ &=\; \frac{1}{\omega(n, d)^2} |X|^{\frac{n-d-1}{2}} |S|^{\frac{n}{2}} \ldots \\ &\quad \int_{\mathcal{S}_{++}^d} |\Sigma|^{-\frac{2n+d+1}{2}} exp\left\{-\frac{1}{2}\operatorname{Tr}\Sigma^{-1}(X+S)\right\} d\Sigma. \qquad (20) \end{aligned}$$

Using the transformation $\Sigma^{-1} \Rightarrow \Sigma$, let $\Sigma^{-1} = R$. Since we deal with symmetric matrices, the Jacobian of this transformation is $J_\Sigma(\Sigma^{-1}) = |\Sigma|^{-(d+1)}$. See the Appendix of [2] for details. This Jacobian can be written as $d\Sigma = \frac{dR}{|R|^{d+1}}$, using which we can write (20) as:

$$\begin{aligned} \Rightarrow \frac{1}{\omega(n, d)^2} |S|^{\frac{n}{2}} |X|^{\frac{n-d-1}{2}} \ldots \qquad (21) \\ \int_{\mathcal{S}_{++}^d} |R|^{\frac{2n-d-1}{2}} exp\left\{-\frac{1}{2}\operatorname{Tr} R(X+S)\right\} dR. \end{aligned}$$

Now, let

$$Y = R(X + S). \tag{22}$$

Using Lemma 2 and assuming $A = (X + S)^{\frac{1}{2}}$, we get the Jacobian of the transformation (22) as $|X + S|^{\frac{d+1}{2}}$, such that we have $|X + S|^{\frac{d+1}{2}} dR = dY$. Substituting this in (21) and using the definition of $\omega(n, d)$, we have the required marginal. □

Using the above marginal, the posterior distribution $p(\Sigma|X, S, n)$ can be derived as below:

$$p(\Sigma|X, S, n) = \frac{p(X|\Sigma, n)\ p(\Sigma|S, n)}{p(X|n)} \tag{23}$$

$$= \frac{|S + X|^n\ |\Sigma|^{-\frac{(2n+d+1)}{2}}}{\omega(2n, d)} exp\left\{-\frac{1}{2}\operatorname{Tr}\Sigma^{-1}(X + S)\right\}. \tag{24}$$

Assuming conditional independence of N data matrices $X_1, X_2, \cdots, X_N$ belonging to the same cluster, if Σ is the parameter associated with this cluster (the scale matrix of the Wishart mixture component), the joint distribution of these N covariances is given by:

$$p(X_1, \cdots, X_N) = \int_{\mathcal{S}_{++}^d} p(X_1, \cdots, X_N|\Sigma, n)\ p(\Sigma|S, n)\ d\Sigma$$

$$= \frac{\omega\left((N + 1)n, d\right)}{\omega(n, d)^{N+1}} |S|^{\frac{n}{2}} \frac{\prod_{i=1}^N |X_i|^{\frac{(n-d-1)}{2}}}{\left|\sum_{i=1}^N X_i + S\right|^{\frac{(N+1)n}{2}}}. \tag{25}$$

Referring back to Algorithm 1, our last step is to derive the expression for the predictive distribution, required by the collapsed Gibbs sampler for inference.

Theorem 4. *Let $X_i \in \mathcal{S}_{++}^d$, $i = 1, 2, \cdots, N - 1$, belong to a cluster C such that each $X_i \sim W(n, d, \Sigma)$, where Σ is the Wishart scale matrix and n, the degrees of freedom. Let $\Sigma \sim IW(n, d, S)$ with inverse scale matrix S. Then the predictive distribution of a data matrix X_N to belong to the cluster $C = \{X_1, X_2, \cdots, X_{N-1}\}$ will be:*

$$p(X_N|X_1, ..., X_{N-1}) = \int_{\mathcal{S}_{++}^d} p(X_N|\Sigma, n)\ p(\Sigma|X_1 ... X_{N-1}) d\Sigma$$

$$= \frac{\omega\left((N + 1)n, d\right)}{\omega(n, d)\ \omega(Nn, d)} \frac{|X_N|^{\frac{(n-d-1)}{2}}\ |\sum_{i=1}^{N-1} X_i + S|^{\frac{Nn}{2}}}{|\sum_{i=1}^N X_i\ +\ S|^{\frac{(N+1)n}{2}}}.$$

Proof: This can be proved directly from Bayes theorem followed by substitutions using (24) and (25). □

6 Alternative DPM Models

As we alluded to above, there are alternative similarity measures, both statistical and differential geometric, that we could potentially use instead of the log-det divergence, leading to other DPM models. We will consider two such alternatives, namely (i) using the squared matrix Frobenius norm as the base measure, and (ii) the squared log-Euclidean distance. Both these distances vectorize the SPD matrix, suggesting the traditional Gaussian-Inverse-Wishart DPMM for nonparametric clustering. We detail this below.

6.1 Frobenius Norm based DPMM

Using the base measure as the squared Frobenius norm, it can be shown that the exponential family for the associated Bregman divergence is the multivariate normal distribution. That is, given $X,\ \mu_X \in \mathcal{S}_{++}^d$, and a variance σ^2,

$$p(X|\mu_X, \sigma^2) \propto exp\left\{-\frac{\|X - \mu_X\|_{\mathrm{F}}^2}{2\sigma^2}\right\} \tag{26}$$

$$\propto exp\left\{-\frac{\|\mathcal{V}(X) - \mathcal{V}(\mu_X)\|_2^2}{2\sigma^2}\right\}, \tag{27}$$

where $\mathcal{V} : \mathcal{S}_{++}^d \to \mathbb{R}^{d(d+1)/2}$ is the half-vectorization operator. The variance σ^2 in (27) can be generalized using a covariance matrix Λ, leading to a standard GIW DPMM, where the $\mathcal{V}(\mu_X) \sim \mathcal{N}(\mu, S_1)$ and $\Lambda \sim IW(n, d, S_2)$; S_1, S_2 are the hyper-scale matrices and μ is the hyper-mean [6].

6.2 Log-Euclidean based DPMM

Similar to the approach above, we can derive the associated density function for the log-Euclidean distance (which computes the Euclidean distance between data matrices via the matrix-logarithm operator). Using the Euclidean embedding suggested in [55], the density function takes the form:

$$p(X|\mu_X, \sigma^2) \propto exp\left\{-\frac{1}{2}\frac{\|\mathcal{V}(\log(X)) - \mu_x\|_2^2}{\sigma^2}\right\} \tag{28}$$

where $\log(.)$ is the matrix logarithm, $\mu_X = 1/N \sum_{i=1}^N \mathcal{V}(\log(X_i))$ and σ^2 is the assumed variance. We can approximate μ_X to follow a multivariate normal distribution, in which case the DPMM follows the standard GIW model as mentioned earlier.

7 Experiments and Results

In this section, we provide experimental results on simulated and real-world data demonstrating the effectiveness of our algorithm compared to state of the art in supervised and unsupervised clustering on covariance valued data. Before going into the details of our experiments, we will review our performance metrics in the next subsection.

7.1 Performance Metrics

Cluster analysis is a fundamental problem in statistics and machine learning for which several standard metrics exist [47]. Of these various metrics, we chose the *pair counting F-measure* and *cluster purity* as our performance metrics. We detail these metrics below.

7.1.1 F1-score

Considering the generality of our proposed algorithm to work with large numbers of clusters of various sizes, we decided to use the *pair counting F-measure* as our primary performance metric. See [67] for a review of the various methods along with their pros and cons. Assuming the ground-truth cluster labels are known for every data point, the F-measure computes the accuracy of clustering in a precision-recall framework. Considering all pairs of data points in every cluster, we can define the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for the clustering depending on the ground truth labels of the points in the pair and whether they belong to the same cluster. Under this notation, we can define the precision and recall of clustering as:

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN}. \tag{29}$$

Then, the F-measure defines the harmonic mean between $Precision$ and $Recall$, that is:

$$F(\beta) = (\beta^2 + 1)PR/(\beta^2 P + R), \tag{30}$$

for a weighting factor β. When $\beta = 1$, we have the *F1-score*, which we will be using in our performance analysis.

To explore the adequacy of covariance matrices as a means of data representation and to evaluate the ability of the DPMM to automatically cluster the data, we further explore variants of the F1-score for performance evaluation; the new metrics we call *purity*. Performance results of our methodology are analyzed and presented in the light of two such purity measures: (i) cluster purity and (ii) class purity.

7.1.2 Cluster Purity

This measure captures the ability of the proposed DPMM methodology (and the associated measure employed) to partition the symmetric positive definite matrix data in the multi-dimensional space they exist. It is defined for every cluster automatically created by the DPMM process as the fraction of class instances that dominate the respective cluster. For example, in Figure 1, cluster 1 is dominated by class instances denoted by the triangles although instances belonging to another class (denoted by the stars) are also included in the same cluster.

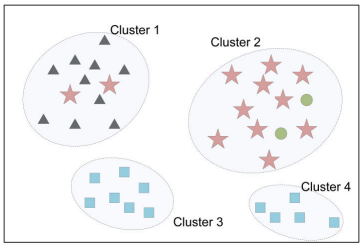


Fig. 1. Illustration of the measure of purity.

Mathematically, we can write this measure as follows: for a cluster C_i, if $label(C_i)$ represents the set of labels of all the data points in C_i, then we define the *cluster purity* as:

$$P_{cluster}(C_i) = \frac{\#\{label(C_i) = \ell^*\}}{\#C_i} \text{ subject to}$$
$$\ell^* = \max_{\ell} \#\{label(C_i) = \ell\}, \tag{31}$$

where $\#\{.\}$ defines the cardinality of the set. An issue with this measure is that it does not take into account the cardinality of each ground truth cluster. For example, in the cluster 2 in Figure 1, there are circle labels which have a low cardinality as compared to star labels and are ignored in $P_{cluster}$.

7.1.3 Class Purity

This measure helps better understand if the feature vectors that we chose for building the covariances adequately capture the differentiating properties of the classes. For a label ℓ, we define the class purity, P_{class} as:

$$P_{class}(\ell) = \frac{\#\{label(C_{k^*}) = \ell\}}{\#C_{k*}} \text{ subject to}$$
$$k^* = \max_{k} \#\{label(C_k) = \ell\}, \tag{32}$$

that is, class purity measures the cluster purity with respect to a ground truth class label; the underlying assumption being that if the features used for covariances are adequate, the clustering algorithm must be able to spawn separate clusters for distinct class labels. This measure also circumvents the issues associated with evaluating unbalanced clusters as mentioned above.

We remark that neither $P_{cluster}$ nor P_{class} can be used as absolute measures of clustering performance. For example, when data with the same label gets split into multiple clusters, each cluster might be pure in itself and thus will have high $P_{cluster}$ and P_{class} scores (for example, clusters 3 and 4 in Figure 1). Thus, evaluation should use a combination of $P_{cluster}, P_{class}$, and the estimated number of clusters; which is what is done using the $F1$ measure implicitly.

7.2 Comparison Methods

We compare our method WIW-DPMM against five standard clustering algorithms, namely (i) K-Means clustering with the affine invariant Riemannian metric (GeoKM) [55] (using the Karcher mean algorithm to compute the cluster centroids), (ii) K-Means clustering using the Log-Euclidean metric (LEKM), (iii) Expectation Maximization (EM) algorithm [32] using a mixture of Wishart distributions, (iv) Spectral Clustering (SP) [50] (using the affine invariant Riemannian metric as the underlying distance), and (v) K-Means using the matrix Frobenius distance (FKM). We also report results comparing our method to other unsupervised clustering algorithms, namely (i) the two alternative DPMM models

on covariances that use the GIW prior on the squared Frobenius norm and the squared log-Euclidean distance, and (ii) the non-linear MeanShift algorithm [62] using the log-Euclidean distance.

Our main evaluation strategy is as follows: for all the supervised clustering algorithms, we increase the preferred number of clusters (K) from a low value to a high value (generally the lowest value is chosen to be half of the true number of clusters and the highest value is chosen as double the true number), and we report the accuracy on the three metrics defined above. As GeoKM convergence is often found to be very slow (using the Karcher mean algorithm), we stop the iterations when the number of cluster changes is less than 1% of the data size. The spectral clustering algorithm requires computing a similarity matrix for every pair of data points, later an eigendecomposition of this matrix is used. We found that computation of this similarity matrix is difficult when the data sizes are beyond 10K and the matrix dimensions are large. Thus, we skip experiments for these methods that are difficult.

7.3 Simulated Experiments

We first evaluate the scalability of our algorithm in a simulated environment via three experiments demonstrating DPMM performance (i) for increasing matrix dimensions, and (ii) the scalability to increasing number of true data clusters, and (iii) increasing data size. All simulation results are averaged over 100 repetitions with different data. We will briefly review our simulation setup next before introducing our experiments in detail.

7.3.1 Simulation Setup

We used the following baseline setup for all our simulation experiments. We used data covariances of size 5×5 in our baseline setup. We fixed the number of true clusters at 50; the positive definite scale matrices associated with each Wishart distribution (from which the data clusters are generated) were generated from 100 uniformly distributed vectors. Using these scale matrices and a fixed number for the degrees of freedom $n = 100$, we sampled 100 data covariances from Wishart distributions using the Bartlett decomposition [37] for a total dataset size of 50K.

7.3.2 Increasing Matrix Dimensionality

Computer vision algorithms using region covariances work with diverse data dimensionalities. For example, texture recognition generally uses covariances of size 5×5, while face recognition uses 40×40 covariances of Gabor filter outputs. In the absence of public covariance datasets for these different applications, we decided to test the effectiveness of our algorithm in the simulated setup for varying dimensionality. Towards this end, we used the baseline simulation setup, but changed the matrix dimensionality to vary from 5 to 100. In Figure 2, we compare the performance of WIW-DPMM against the standard clustering algorithms on the F1 score. As is clear from the figure, WIW-DPMM scales well with the dimensionality and achieves perfect clustering accuracy.

7.3.3 Increasing Number of True Clusters

Recall that our main motivation for investigating a Bayesian formalism for clustering covariances is due to its ability to cluster covariances of increasing complexity (that is, the number of ground truth clusters). Towards this end, we changed the baseline setup using 5×5 covariances, with the true number of clusters increasing from 10 to 1000, with each cluster containing 100 covariances each. In Figure 2(b), we plot the accuracy of WIW-DPMM against other methods. We assumed K is known for the supervised clustering algorithms. As is clear from the figure, our DPMM model (which does not assume K) achieves similar performance as other algorithms with known K, demonstrating the correctness and effectiveness of our approach. The decreasing performance, we suspect, is due to the increased similarity between the hyper-prior scale matrices used to sample the data covariances, when the number of true clusters increased. Note that this experiment also evaluates the performance of DPMM for an increasing dataset size.

7.3.4 Computational Performance

In this experiment, we benchmark the computational performance of our DPM model against other algorithms on SPD matrices. Figure 2(d) shows the average clustering time when the true number of clusters increases from 50 to 500 with 100 covariances in each cluster. Each algorithm was implemented in Matlab, and we used a single core 3GHz Pentium machine for the experiment. The algorithms were run until convergence or for 100 iterations whichever happened earlier. It is to be noted that the Geodesic K-Means is computationally very expensive. Figure 2(c) shows a similar comparison for increasing matrix dimensions. These two experiments piggybacked the setup that we used above. The figures show that the computational expenditure incurred by WIW-DPMM is comparable to that of spectral clustering or EM algorithms.

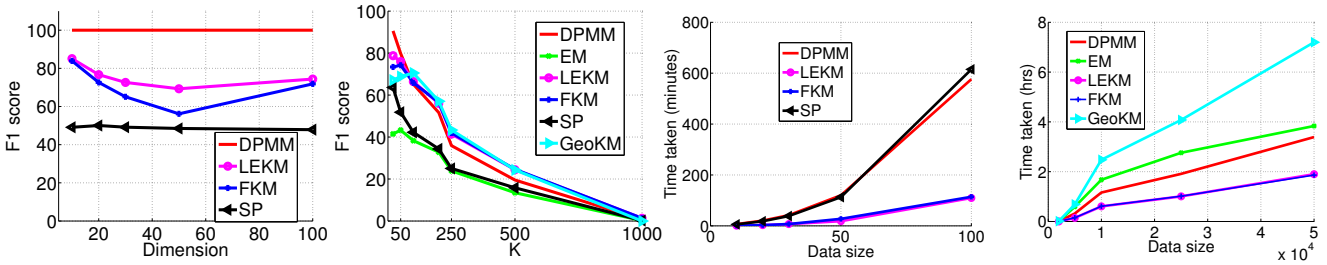
7.4 Real World Experiments

Let us move on to the real-world applications of our clustering framework. We experimented on four applications, namely (i) texture recognition, (ii) appearance clustering, (iii) object recognition, and (iv) face recognition. Below we detail the datasets used in these experiments, along with describing the features used for generating the covariance objects.

7.4.1 Brodatz Texture

Region covariances have demonstrated superior performances in texture recognition applications [44], [52]. We used the Brodatz texture dataset[3], which consists of 110

3. http://www.ux.uis.no/~tranden/brodatz.html



(a) Dimensions (b) #Clusters (c) Time taken (dims) (d) Time taken (datasize)

Fig. 2. Simulation results: 2(a) shows accuracy for increasing matrix dimensions keeping the true number of clusters at 50 and each cluster consisting of 100 covariances. Figure 2(b) shows accuracy for increasing number of true clusters keeping the matrix dimensionality fixed at 5×5 and 100 covariances in each cluster. Figure 2(c) plots the time taken for 100 iterations of each algorithm using the experiment shown in Figure 2(a). We do not show results for GeoKM and EM for this plot as they are not found to converge in reasonable time, especially for large covariance dimensions. In Figure 2(d), we plot the time taken for clustering until convergence when the dataset size increases from 500 to 50K at the same time the number of true clusters increasing number from 5 to 500, using 5×5 covariances.

gray scale texture images, each of dimension 512×512. We sampled 300 patches of dimensions 25×25 from random locations of each image. Those patches, without any useful textures (low entropy), were removed. This resulted in approximately 10K patches. We used a 5D feature descriptor to generate the covariances, with features defined by: $F_{textures} = [x, y, I, abs(I_x), abs(I_y)]^T$, where the first two dimensions are the coordinates of a pixel from the top-left corner of a patch, the last three dimensions are the image intensity, and image gradients in the x and y directions respectively. Region covariances of size 5×5 were computed from all features in a patch.

7.4.2 ETHZ Person Re-identification Dataset

Tracking and recognition of people appearances is an essential component of visual surveillance applications. Compared to other areas of computer vision, the visual data in these applications are often of very low resolution, are corrupted by illumination changes, pose variations due to multiple camera views, and suffer from occlusions. Region covariances have been shown to provide a robust platform for person re-identification in surveillance tasks [30], [45]. In this experiment, we evaluate the performance of clustering people appearances on the benchmark ETHZ dataset [58]. This dataset consists of real world surveillance images of resolutions ranging between 78×30 to 400×200. The images are from 146 different individuals, and the cluster cardinalities range from 5 to 356. Sample images from this dataset are shown in Figure 3. There are a total of 8580 images in this dataset.

Clustering on this dataset expects the covariance descriptors from the same individual to be grouped together. Several types of features have been suggested in literature that have shown varying degrees of success; examples include Gabor wavelet based features as in [45], color gradient based features as in [30], etc. Rather than detailing the results on several feature combinations, we describe here the feature combination that worked the best in our experiments.[4] We used a combination of nine features for each image as described below:

$$F_{ETHZ} = [x\ I_r\ I_g\ I_b\ Y_i\ abs(I_x)\ abs(I_y) \\ abs(sin(\theta) + cos(\theta))\ abs(H_y)]\,, \tag{33}$$

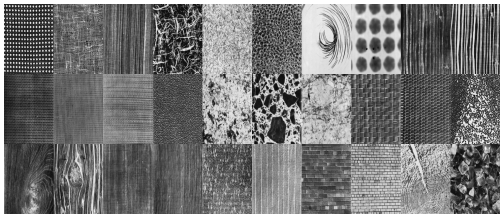
where x is the x-coordinate of a pixel location, I_r, I_g, I_b are the RGB color of a pixel, Y_i is the pixel intensity in the YCbCr color space, I_x, I_y are the gray scale pixel gradients, and H_y is the y-gradient of pixel hue. Further, we also use the gradient angle $\theta = tan^{-1}(I_y/I_x)$ in our feature set. Each image is resized to a fixed size 300×100, and is divided into upper and lower parts. We compute two different region covariances for each part, which are combined as two block diagonal matrices to form a single covariance descriptor of size 18×18 for each appearance image.

7.4.3 ETH80 Object Recognition

Region covariances have shown promising results for the task of object recognition [57]. In this experiment, we demonstrate the performance of unsupervised clustering of object appearances on the ETH80 dataset, which consists of 8 ground truth categories. Each category has ten different instances of an object and each instance has 41 different views. Although the images are clean, we cannot rely on a single image cue to recognize each category due to the high intra-class diversity. Further, the instances undergo severe view point change. Sample images demonstrating the difficulty of this dataset are provided in Figure 3. The dataset has 3,198 images.

Inspired by the experiments in [40], we use a combination of texture and color features for generating the region covariances. First, the objects are segmented out from the images using the given ground truth masks. To preserve the object boundaries, we used a dilation of

4. We used a validation set of 500 covariances and 10 true clusters from this dataset. The performance was evaluated over LEKM.



(a) Brodatz textures

(b) ETHZ appearances

(c) ETH80 objects

(d) FERET faces

Fig. 3. Sample images from the various datasets that we use in our experiments.

these masks. Next, texture features were computed by a bank of three separable Laws filter masks [39]. Suppose $H_1 = [1\ 2\ 1]^T$, $H_2 = [-1\ 0\ 1]^T$, and $H_3 = [-1\ 2\ -1]^T$ are these filters, then the filter bank is defined as:

$$L_{bank} = \big[H_1H_1^T,\ H_1H_2^T\ H_1H_3^T\ H_2H_1^T \\ H_2H_2^T\ H_2H_3^T\ H_3H_1^T\ H_3H_2^T\ H_3H_3^T\big]\,. \tag{34}$$

Let F_{Laws} be a 9D feature vector obtained from a pixel after applying L_{bank}. We also append other texture and color features as provided by the pixel color, and gradients. The complete feature vector has the following form:

$$F_{ETH80} = [F_{Laws}, x, y, I_r, I_g, I_b, abs(I_x), \\ abs(I_y), abs(I_{LoG}), \sqrt{I_x^2 + I_y^2}\big]\,, \tag{35}$$

where I_{LoG} stands for the Laplacian of Gaussian filter, useful for edge detection. With this feature set, we generate covariances of size 18×18 for each segmented image.

7.4.4 FERET Face Recognition

Our main motivation for this experiment is to analyze the performance of our clustering framework in real-world applications that use large region covariances. To this end, we selected the feature set in [53] for face recognition. We used the benchmark FERET face dataset, which consists of face images of multiple people, with each face undergoing pose variations to varying degrees. Sample images from this dataset are shown in Figure 3. We used 800 images from this dataset from 110 different faces. As suggested in [53], we used a filter bank of 40 Gabor filters with 5 scales and 8 different orientations to extract the facial features.

7.4.5 Experimental Setup: Hyperparameters

One important challenge in the DPM model is to choose the hyperparameters of the model, such as the concentration parameter α and the inverse-Wishart scale matrix S. We sampled α from a Gamma prior ($G(1, 0.5)$) for every iteration of the collapsed Gibbs sampler as suggested in [70]. The hyperparameter scale matrix S was estimated by taking the Karcher mean of all the covariances in the dataset. The DPMMs in all the experiments start with an initial set of 1000 clusters, with data points assigned to each cluster randomly. The number of degrees of freedom (n) was chosen to be twice the data dimensionality, which seemed to work well in all our experiments.

7.4.6 Results

The collapsed Gibbs sampler was found to converge in approximately 200 iterations in all our experiments. Our results are averaged over 10 different repetitions of the algorithms with different initializations. In Figures 4(a), 4(b), 4(c), and 4(d) we show the F1-score, the cluster purity score, and the class purity score of our WIW-DPMM model against standard soft-clustering algorithms. As is clear from the figures, the Frobenius norm based K-Means show the worst performance in all our experiments, validating our assumption that treating the covariances as Euclidean vectors is not appropriate. The accuracies of the K-Means based algorithms (such as LEKM, GeoKM, SP, and FKM) showed similar trends in the various performance metrics, while EM is seen to showcase a slightly different trend. This is perhaps due to a better cluster initialization. Note that, in EM, we initialize the clusters from the output of a K-Means algorithm (using the Log-Euclidean distance).

From Figure 4, we see that the performance of the various comparison methods are similar to ours when the covariance dimensionality is low and the cluster cardinality is high (such as the Brodatz textures), while differs significantly when the dimensionality is high, while each cluster has very few instances (such as the FERET dataset). The latter phenomenon is perhaps due to three important reasons, namely (i) large covariances are often found to be close to being semi-definite; as a result leads to numerical instability when computing the geodesic distances (such as the GeoKM and SP methods), (ii) large covariances are more vulnerable to the curse of dimensionality (FKM and LEKM), and (iii) small cluster cardinalities lead to poor estimation of the cluster parameters (as in the EM). In contrast, in the DPMM setup, the predictive distribution is computed in the log domain leading to better numerical stability, while the DPMM procedure (via the concentration parameter) brings in more flexibility in generating new clusters leading to better accuracy. On the other extreme, when the number of instances in a cluster are not sufficient to estimate the parameters, the DPMM framework over-clusters the data, which can be seen for the estimated number of clusters (K) in Figures 4(b) and 4(c); the former has 146 true clusters, but DPMM finds more than 250 clusters as some of the true clusters in this dataset have only a few (about 10 or less) instances. On the other hand, for the ETH80 dataset, there are only 8 true clusters, but each has a large number of instances, resulting in better clustering.

The performance of GeoKM is sometimes seen inferior to LEKM (in the texture and face datasets), which is not unexpected and may be attributed to the early stopping when convergence is slow. Further, we found that the FERET face dataset often produced highly ill-conditioned covariances. As a result, LEKM which uses the matrix logarithm, shows improved results. Compared to all the standard algorithms, our WIW-DPMM model showed excellent performance, especially in the F1-score. We also find that the cluster purity is generally high for all the approaches, implying that large clusters are mostly pure. The class purity is also very high near to the true number of clusters, suggesting the choice of our features is adequate for discriminating the classes. A high class purity around the ground truth K indicates that classes with low cardinalities often fall into their own clusters (note that it increases with more clusters).

7.4.7 Comparisons Against Unsupervised Methods

In this subsection, we show results comparing WIW-DPMM against alternative unsupervised methods, specifically the GIW DPM models and the Log-Euclidean based mean-shift algorithm. For the GIW DPMMs, the vectorized covariances are of high dimensionality (e.g., an 18×18 matrix results in a half-vector of 171D); the dataset size might not be sufficient to estimate useful hyper-parameters. We found that such a vectorization often results in GIW DPM models performing poorly. Thus, we used an intermediate PCA step to reduce the dimensionality to 10D for the ETHZ, ETH80, and FERET datasets. We experimented with several other dimensionalities (on a small validation set) and found inferior results. The bandwidth of the mean-shift algorithm was decided as the mean of a k-NN search on the datasets using the log-Euclidean distance. In Figure 5, we show the results of these experiments. As is clear from the bar plots, our WIW-DPMM method demonstrates superior clustering performance to other unsupervised methods. This is especially evident when the covariance dimensionality is high, especially for the FERET dataset. This is perhaps because the FERET dataset has a large number of clusters (110) while each cluster has only 7 face instances; as a result, the PCA step could have failed to capture the appropriate principal components that was essential for distinguishing the cluster boundaries. An method that might have been useful in this context is the manifold based dimensionality reduction strategy proposed in [29], which we leave as interesting future work.

In Figure 6, we show a few qualitative results when applying our algorithm to two problems, namely (i) texture recognition, and (ii) texture based aerial image[5] segmentation. For the former case, we show a comparison against the mean-shift algorithm. We used covariances computed using the 5D texture features we defined in Subsection 7.4.1 for this experiment. For the aerial image, we used a texture recognition approach as before, but used gradients of each color channel as well, leading to each covariance of size 11×11. For this experiment, we used a patch size of 16×16, leading to approximately 39K covariances. Each segment found using our algorithm is shown in Figure 6 using a unique color.

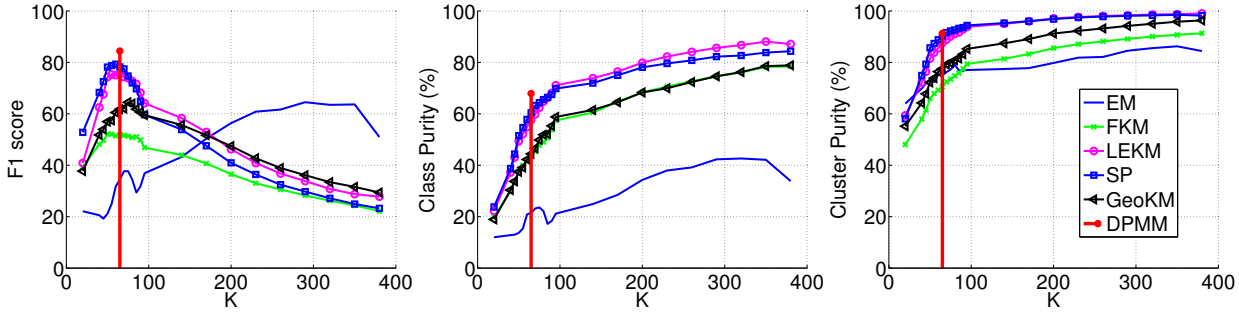
8 Conclusions and Future Work

In this paper, a nonparametric Bayesian framework was introduced for clustering SPD matrices by extending the Dirichlet process Mixture Models. Using the log-det divergence as the underlying similarity measure for comparing SPD matrices, we derived a collapsed Gibbs sampler using the Wishart Inverse Wishart conjugate prior. Our experiments demonstrated the superiority of our scheme against other supervised and unsupervised clustering algorithms, especially substantiating that applying vector based DPMMs to covariances is not useful.

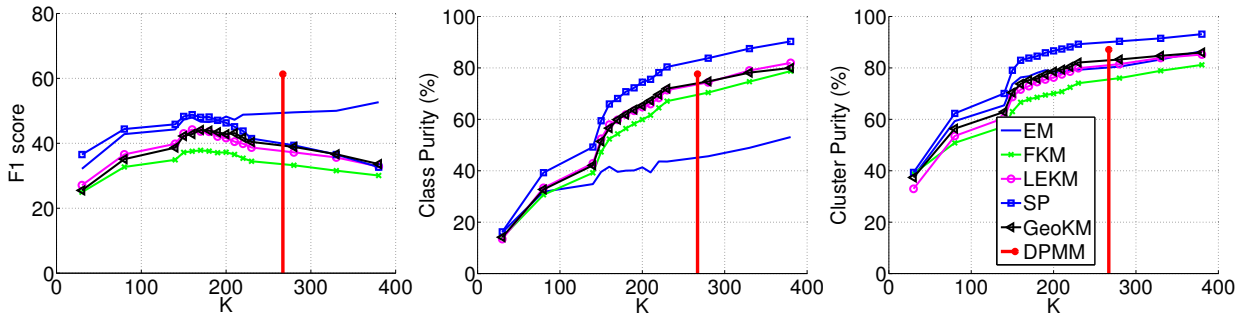
Acknowledgments

The authors would like to thank Prof. Arindam Banerjee (University of Minnesota) for many helpful discussions. This material is based upon work supported in part by Honeywell, and the National Science Foundation through grants #CNS-0821474, #IIP-0934327, #CNS-1039741, #SMA-1028076, and #CNS-1338042.

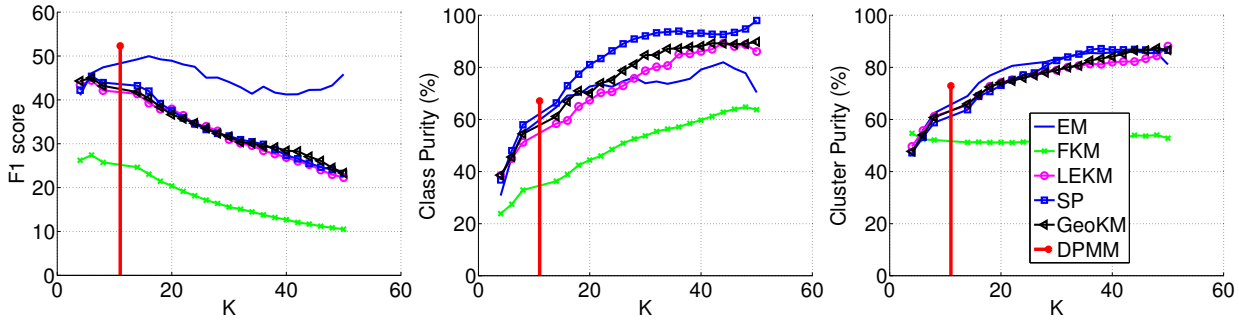
5. Downloaded from http://www.aerialperspectives.net/aerials/



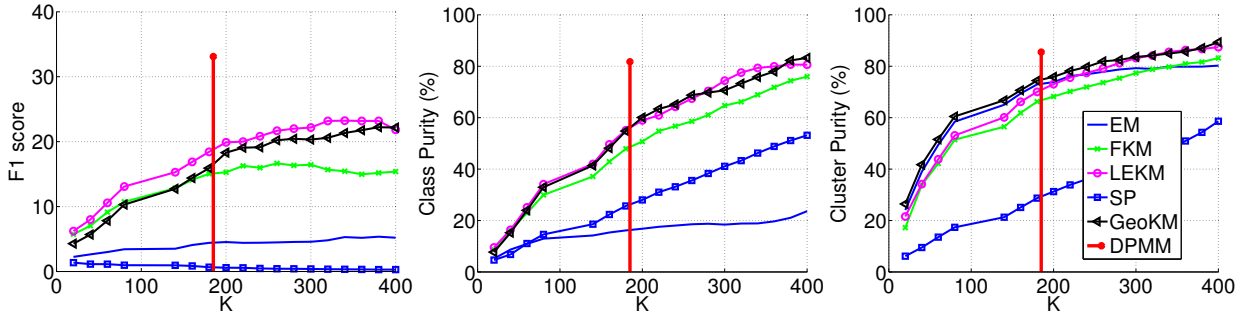
(a) Brodatz textures - 10K covariances of dimension 5×5 from 110 classes.



(b) ETHZ person re-identification dataset - 8580 covariances of dimension 18×18 from 146 classes.

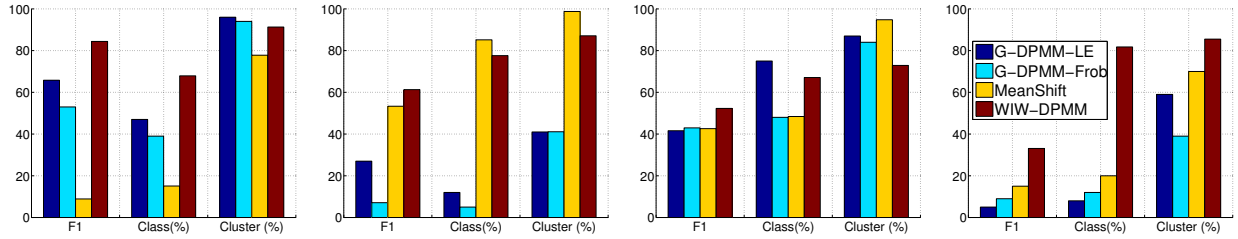


(c) ETH80 object recognition dataset - 3198 covariances of dimension 18×18 from 8 classes.



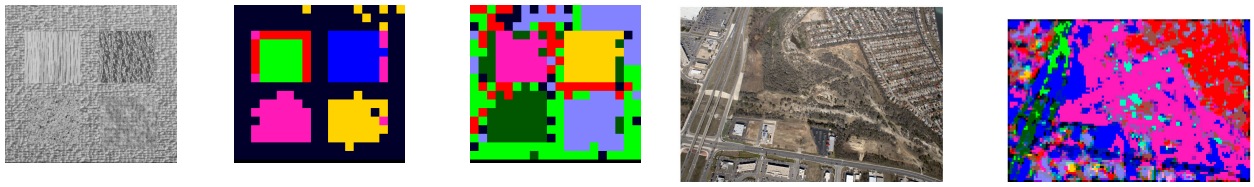
(d) FERET face dataset - 800 covariances of dimension 40×40 from 110 classes.

Fig. 4. The red vertical lines in the plots show the result from our DPMM framework for which we do not need to provide the number of clusters K.



(a) Brodatz texture (b) ETHZ appearance (c) ETH-80 objects (d) FERET faces

Fig. 5. Comparison of unsupervised clustering algorithms on the four datasets.



(a) Texture group (b) WIW-DPMM (c) Mean-shift (d) Aerial Image (e) WIW-DPMM

Fig. 6. Figures 6(a), 6(b), 6(c) show segmentation results for Brodatz textures. Figures 6(d), 6(e) show results from an aerial image segmentation using our WIW-DPMM method.

REFERENCES

[1] D. Alexander, C. Pierpaoli, P. Basser, and J. Gee. Spatial transformations of diffusion tensor magnetic resonance images. *IEEE Transactions on Medical Imaging*, 20(11):1131–1139, 2002.

[2] T. Anderson. *An Introduction to Multivariate Statistical Analysis*. 1958.

[3] C. E. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Annals of Statistics*, pages 1152–1174, 1974.

[4] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, 2006.

[5] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh. Clustering with Bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005.

[6] M. Beal, Z. Ghahramani, and C. Rasmussen. The infinite hidden Markov model. *Advances in Neural Information Processing Systems*, 2002.

[7] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, 2001.

[8] D. A. Bini and B. Iannazzo. Computing the Karcher mean of symmetric positive definite matrices. *Linear Algebra and its Applications*, 2011.

[9] D. Blackwell and J. MacQueen. Ferguson distributions via Polya urn schemes. *Annals of Statistics*, 1(2):353–355, 1973.

[10] D. M. Blei and M. I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–143, 2006.

[11] N. Bouguila and D. Ziou. A Dirichlet process mixture of generalized Dirichlet distributions for proportional data modeling. *Neural Networks*, 21(1):107–122, 2010.

[12] L. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

[13] F. Caron, M. Davy, A. Doucet, E. Duflos, and P. Vanheeghe. Bayesian inference for linear dynamic models with Dirichlet process mixtures. *IEEE Transactions on Signal Processing*, 56(1):71–84, 2008.

[14] A. Cherian, V. Morellas, N. Papanikolopoulos, and S. J. Bedros. Dirichlet process mixture models on symmetric positive definite matrices for appearance clustering in video surveillance applications. In *Computer Vision and Pattern Recognition*. IEEE, 2011.

[15] J. Davis and I. Dhillon. Differential entropic clustering of multivariate Gaussians. *Advances in Neural Information Processing Systems*, 2007.

[16] J. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon. Information-theoretic metric learning. In *International Conference on Machine Learning*, 2007.

[17] B. De Finetti. *Probability, induction and statistics: The art of guessing*. J. Wiley & Sons, 1972.

[18] I. Dhillon and J. Tropp. Matrix nearness problems with Bregman divergences. *SIAM Journal on Matrix Analysis and Applications*, 29(4):1120–1146, 2007.

[19] M. Eaton. *Multivariate Statistics: A Vector Space Approach*. Wiley New York, 1983.

[20] M. D. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90(430):577–588, 1995.

[21] U. Fano. Description of states in quantum mechanics by density matrix and operator techniques. *Reviews of Modern Physics*, 29(1):74–93, 1957.

[22] T. Ferguson. A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2):209–230, 1973.

[23] T. S. Ferguson. Bayesian density estimation by mixtures of normal distributions. *Recent Advances in Statistics*, 24:287–302, 1983.

[24] L. Ferro-Famil, E. Pottier, and J.-S. Lee. Unsupervised classification of multifrequency and fully polarimetric SAR images based on the H/A/Alpha-Wishart classifier. *IEEE Transactions on Geoscience and Remote Sensing*, 39(11):2332–2342, 2001.

[25] W. Forstner and B. Moonen. A metric for covariance matrices. *Qua vadis geodesia*, pages 113–128, 1999.

[26] A. Goh and R. Vidal. Clustering and dimensionality reduction on Riemannian manifolds. In *Computer Vision and Pattern Recognition*. IEEE, 2008.

[27] T. Griffiths, M. Jordan, J. Tenenbaum, and D. M. Blei. Hierarchical topic models and the nested chinese restaurant process. *Advances in Neural Information Processing Systems*, 2004.

[28] K. Guo, P. Ishwar, and J. Konrad. Action recognition using sparse representation on covariance manifolds of optical flow. In *Advanced Video and Signal Based Surveillance*. IEEE, 2010.

[29] M. T. Harandi, M. Salzmann, and R. Hartley. From manifold to manifold: geometry-aware dimensionality reduction on spd matrices. In *European Conference on Computer Vision*. Springer, 2014.

[30] M. T. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell. Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach. In *European Conference on Computer Vision*. Springer, 2012.

[31] G. Henry and D. Rodriguez. Kernel density estimation on Riemannian manifolds: asymptotic results. *Journal of Mathematical Imaging and Vision*, 34(3):235–239, 2009.

[32] S. Hidot and C. Jean. An expectation-maximization algorithm for

the Wishart mixture model: Application to movement clustering. *Pattern Recognition Letters*, 31(14):2318–2324, 2010.
[33] H. Ishwaran and L. F. James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453), 2001.
[34] W. James and C. Stein. Estimation with quadratic loss. *Breakthroughs in Statistics: Foundations and Basic Theory*, page 443, 1992.
[35] J. J. Kivinen, E. B. Sudderth, and M. I. Jordan. Learning multiscale representations of natural scenes using Dirichlet processes. In *International Conference on Computer Vision*. IEEE, 2007.
[36] A. Kottas, J. A. Duan, and A. E. Gelfand. Modeling disease incidence data with spatial and spatio temporal Dirichlet process mixtures. *Biometrical Journal*, 50(1):29–42, 2008.
[37] A. Kshirsagar. Bartlett decomposition and Wishart distribution. *The Annals of Mathematical Statistics*, 30(1):239–241, 1959.
[38] K. Kurihara, M. Welling, and Y. W. Teh. Collapsed variational Dirichlet process mixture models. In *International Joint Conferences on Artificial Intelligence*, 2007.
[39] K. I. Laws. Rapid texture identification. In *24th Annual Technical Symposium*, pages 376–381. International Society for Optics and Photonics, 1980.
[40] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *Computer Vision and Pattern Recognition*. IEEE, 2003.
[41] L.-J. Li and L. Fei-Fei. Optimol: automatic online picture collection via incremental model learning. *International Journal of Computer Vision*, 88(2):147–168, 2010.
[42] P. Liang, M. I. Jordan, and D. Klein. Probabilistic grammars and hierarchical Dirichlet processes. *The Handbook of Applied Bayesian Analysis*, 2009.
[43] J. S. Liu. The collapsed Gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 89(427):958–966, 1994.
[44] R. Luis-García, R. Deriche, and C. Alberola-López. Texture and color segmentation based on the combined use of the structure tensor and the image components. *Signal Processing*, 88(4):776–795, 2008.
[45] B. Ma, Y. Su, F. Jurie, et al. Bicov: a novel image representation for person re-identification and face verification. In *British Machine Vision Conference*, 2012.
[46] B. Ma, Y. Wu, and F. Sun. Affine object tracking using kernel-based region covariance descriptors. In *Foundations of Intelligent Systems*, pages 613–623. Springer, 2012.
[47] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.
[48] N. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, pages 249–265, 2000.
[49] J. Neyman and E. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London*, pages 289–337, 1933.
[50] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2002.
[51] O. Tuzel, F. Porikli, and P. Meer. Covariance Tracking using Model Update Based on Lie Algebra. *Computer Vision and Pattern Recognition*, 2006.
[52] O. Tuzel, F. Porikli, and P. Meer. Region Covariance: A Fast Descriptor for Detection and Classification. In *European Conference on Computer Vision*, 2006.
[53] Y. Pang, Y. Yuan, and X. Li. Gabor-based region covariance matrices for face recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(7):989–993, 2008.
[54] B. Pelletier. Kernel density estimation on Riemannian manifolds. *Statistics & Probability Letters*, 73(3):297–304, 2005.
[55] X. Pennec, P. Fillard, and N. Ayache. A Riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.
[56] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
[57] J. Sadeep, H. Richard, S. Mathieu, L. H., and M. Harandi. Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *Computer Vision and Pattern Recognition*, 2013.
[58] W. Schwartz and L. Davis. Learning Discriminative Appearance-Based Models Using Partial Least Squares. In *Proceedings of the XXII Brazilian Symposium on Computer Graphics and Image Processing*, 2009.
[59] J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4(2):639–650, 1994.
[60] Y. Shinohara, T. Masuko, and M. Akamine. Covariance clustering on Riemannian manifolds for acoustic model compression. In *International Conference on Acoustics, Speech and Signal Processing*, 2010.
[61] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos. Tensor sparse coding for region covariances. In *European Conference onComputer Vision*. Springer, 2010.
[62] R. Subbarao and P. Meer. Nonlinear mean shift over Riemannian manifolds. *International Journal of Computer Vision*, 84(1):1–20, 2009.
[63] E. Sudderth. *Graphical models for visual object recognition and tracking*. PhD thesis, 2006.
[64] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky. Describing visual scenes using transformed Dirichlet processes. *Advances in Neural Information Processing Systems*, 2006.
[65] Y. W. Teh and M. I. Jordan. Hierarchical bayesian nonparametric models with applications. *Bayesian Nonparametrics: Principles and Practice*, pages 158–207, 2010.
[66] D. Tosato, M. Farenzena, M. Spera, V. Murino, and M. Cristani. Multi-class classification on Riemannian manifolds for video surveillance. In *European Conference on Computer Vision*, 2010.
[67] S. Wagner and D. Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik, 2007.
[68] R. Wang, H. Guo, L. S. Davis, and Q. Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *Computer Vision and Pattern Recognition*. IEEE, 2012.
[69] X. Wang, W. E. L. Grimson, and C.-F. Westin. Tractography segmentation using a hierarchical Dirichlet processes mixture model. *NeuroImage*, 54(1):290–302, 2011.
[70] M. West. *Hyperparameter estimation in Dirichlet process mixture models*. Duke University, 1992.
[71] E. Xing, M. Jordan, and R. Sharan. Bayesian haplotype inference via the Dirichlet process. *Journal of Computational Biology*, 14(3):267–284, 2007.
[72] Z. Zhang, G. Dai, and M. I. Jordan. Matrix-variate Dirichlet process mixture models. In *International Conference on Artificial Intelligence and Statistics*, 2010.

Anoop Cherian is a Research Fellow at the Australian National University. Previously, he was a Postdoctoral Researcher in the LEAR team at INRIA at Grenoble. He received his B.Tech (honors) degree in computer science and Engineering from the National Institute of Technology, Calicut, India in 2002, his M.S. and Ph.D. degrees in computer science from the University of Minnesota, Minneapolis in 2010 and 2013 respectively. From 2002–2007, he worked as a software design engineer at Microsoft. His research interests lie in the areas of computer vision and machine learning.

Vassilios Morellas received his diploma degree in Mechanical Engineering from the National Technical University of Athens, Greece, his MSME degree from Columbia University, NY and his PhD degree from the department of Mechanical Engineering at the University of Minnesota. He is Program Director in the department of Computer Science & Engineering and Executive Director of the NSF Center for Safety Security and Rescue. His research interests are in the area of geometric image processing, machine learning, robotics and sensor integration.

Nikolaos Papanikolopoulos received his Diploma of Engineering in Electrical and Computer Engineering, from the National Technical University of Athens in 1987. He received his M.S. in 1988 and PhD in 1992 in Electrical and Computer Engineering from Carnegie Mellon University. His research interests include robotics, sensors for transportation applications, computer vision, and control systems. As the director of the Center for Distributed Robotics, his research has included projects involving vision-based sensing and classification of vehicles, and the recognition of human activity in public areas.