

# Bregman Divergences for Infinite Dimensional Covariance Matrices

Mehrtash Harandi<sup>1,2</sup>, Mathieu Salzmann<sup>2,1</sup>, and Fatih Porikli<sup>1,2</sup>

<sup>1</sup>College of Engineering and Computer Science, Australian National University, Australia

<sup>2</sup>NICTA, Canberra Research Laboratory, Australia \*

{mehrtash.harandi, mathieu.salzmann, fatih.porikli}@nicta.com.au

## Abstract

We introduce an approach to computing and comparing Covariance Descriptors (CovDs) in infinite-dimensional spaces. CovDs have become increasingly popular to address classification problems in computer vision. While CovDs offer some robustness to measurement variations, they also throw away part of the information contained in the original data by only retaining the second-order statistics over the measurements. Here, we propose to overcome this limitation by first mapping the original data to a high-dimensional Hilbert space, and only then compute the CovDs. We show that several Bregman divergences can be computed between the resulting CovDs in Hilbert space via the use of kernels. We then exploit these divergences for classification purpose. Our experiments demonstrate the benefits of our approach on several tasks, such as material and texture recognition, person re-identification, and action recognition from motion capture data.

## 1. Introduction

In this paper, we tackle the problem of employing infinite-dimensional Covariance Descriptors (CovDs) for classification. CovDs are becoming increasingly popular in many computer vision tasks due to their robustness to measurement variations [24]. Such descriptors take the form of, e.g., region covariance matrices for pedestrian detection [24] and texture categorization [8], human joint covariances for activity recognition [10], and covariance matrices of the local Brownian motion of water molecules in diffusion tensor imaging (DTI) [20].

As the name implies, CovDs are obtained by computing the second order statistics of feature vectors extracted

at a finite number of observation points, such as the pixels of an image. The resulting descriptors are Symmetric Positive Definite (SPD) matrices and naturally lie on non-linear manifolds known as tensor, or SPD manifolds. As a consequence, Euclidean geometry is often not appropriate to analyze CovDs [20]. To overcome the drawbacks of Euclidean geometry and better account for the Riemannian structure of CovDs, state-of-the-art methods make use of non-Euclidean metrics (e.g., [20, 11]). In particular, Bregman divergences have recently been successfully employed in a number of CovD-based applications [26, 22, 8, 5, 21].

Nevertheless, all previous studies work with relatively small CovDs (i.e., at most  $50 \times 50$ , to the best of our knowledge) built from feature vectors whose dimension is typically much smaller than the number of observations. While this could be thought of as a filtering operation, it also implies that the information encoded in such a CovD is inherently poorer than the information jointly contained in all the observations. Recently, it was shown that CovDs could be mapped to Reproducing Kernel Hilbert Space (RKHS) via the use of SPD-specific kernels [8, 11]. While this may, to some degree, enhance the discriminative power of the low-dimensional CovDs, it is unlikely to be sufficient to entirely recover the information lost when constructing them.

In this paper, we overcome this issue by introducing an approach to building and analyzing infinite-dimensional CovDs from a finite number of observations. To this end, we map the original features to RKHS and compute CovDs in the resulting space. Since the dimensionality of the RKHS is much larger than the dimensionality of the observations, the resulting descriptor will encode more information than a CovD constructed in the original lower-dimensional space, and is therefore better suited for classification.

In practice, of course, the mapping to RKHS is unknown and the CovDs cannot be explicitly computed. However, here, we show that several Bregman divergences can be derived in Hilbert space via the use of kernels, thus alleviating

\*NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the ARC through the ICT Centre of Excellence program. Mehrtash Harandi is partially supported by the ARC discovery grant DP130104567.

the need for the explicit mapping. In particular, we consider the Burg [22], Jeffreys [26] and Stein [22] divergences, that have proven powerful to analyze SPD matrices. These divergences allow us to perform classification in Hilbert space via a simple nearest-neighbor (NN) classifier, or by making use of more sophisticated distance-based classifiers, such as support vector machines (SVM) with a Gaussian kernel.

We evaluated the resulting descriptors on the tasks of image-based material, texture and virus recognition, person re-identification, and action recognition from motion capture data. Our experimental evaluation clearly evidences the importance of keeping all the data information by mapping to Hilbert space before computing the CovDs. Furthermore, our empirical results show that, with this new representation, a simple NN classifier can achieve accuracies comparable to those of much more sophisticated methods, and that these accuracies can even be boosted beyond the state-of-the-art when using more powerful classifiers.

## 2. Theory of Bregman Divergences

In this section, we review several Bregman divergences and discuss the properties that motivated our decision to use them to compare CovDs in RKHS.

Throughout the paper, we use bold upper-case letters to denote matrices (e.g.,  $\mathbf{C}$ ) and bold lower-case letters for column vectors (e.g.,  $\mathbf{x}$ ). The  $n \times n$  identity matrix is written as  $\mathbf{I}_n$ .  $GL(n)$  denotes the general linear group, i.e., the group of real invertible  $n \times n$  matrices.  $\mathcal{S}_{++}^n$  is the space of  $n \times n$  symmetric positive definite matrices, i.e.,  $\mathbf{C} \in \mathcal{S}_{++}^n$  iff  $\mathbf{a}^T \mathbf{C} \mathbf{a} > 0, \forall \mathbf{a} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ .

**Definition 2.1.** Let  $\zeta : \mathcal{S}_{++}^n \rightarrow \mathbb{R}$  be a strictly convex and differentiable function defined on the symmetric positive cone  $\mathcal{S}_{++}^n$ . The Bregman matrix divergence  $d_\zeta : \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n \rightarrow [0, \infty)$  is defined as

$$d_\zeta(\mathbf{C}_1, \mathbf{C}_2) = \zeta(\mathbf{C}_1) - \zeta(\mathbf{C}_2) - \langle \nabla_{\mathbf{C}_2} \zeta, \mathbf{C}_1 - \mathbf{C}_2 \rangle, \quad (1)$$

where  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{Tr}(\mathbf{A}^T \mathbf{B})$ , and  $\nabla_{\mathbf{C}_2} \zeta$  is the gradient of  $\zeta$  evaluated at  $\mathbf{C}_2$ . The Bregman divergence is non-negative and definite (i.e.,  $d_\zeta(\mathbf{C}_1, \mathbf{C}_2) = 0$  iff  $\mathbf{C}_1 = \mathbf{C}_2$ ).

**Definition 2.2.** The Euclidean (Frobenius) distance is obtained by using  $\zeta(\mathbf{C}) = \text{Tr}(\mathbf{C}^T \mathbf{C})$  as seed function in the Bregman divergence of Eq. 1.

**Definition 2.3.** The Burg, or  $B$ -, divergence is obtained by using  $\zeta(\mathbf{C}) = -\log \det(\mathbf{C})$  as seed function in the Bregman divergence of Eq. 1, where  $\det(\cdot)$  denotes the determinant of a matrix. The  $B$ -divergence can be expressed as

$$B(\mathbf{C}_1, \mathbf{C}_2) = \text{Tr}(\mathbf{C}_1 \mathbf{C}_2^{-1}) - \log \det(\mathbf{C}_1 \mathbf{C}_2^{-1}) - n. \quad (2)$$

While Bregman divergences exhibit a number of useful properties [12], their general asymmetric behavior is often counter-intuitive and undesirable in practical applications. Therefore, here, we also consider two symmetrized Bregman divergences, namely the *Jeffreys* and the *Stein* divergences.

**Definition 2.4.** The *Jeffreys*, or  $J$ -, divergence is obtained from the Burg divergence, and can be expressed as

$$\begin{aligned} J(\mathbf{C}_1, \mathbf{C}_2) &= \frac{1}{2} B(\mathbf{C}_1, \mathbf{C}_2) + \frac{1}{2} B(\mathbf{C}_2, \mathbf{C}_1) \\ &= \frac{1}{2} \text{Tr}(\mathbf{C}_1 \mathbf{C}_2^{-1}) - \frac{1}{2} \log \det(\mathbf{C}_1 \mathbf{C}_2^{-1}) \\ &\quad + \frac{1}{2} \text{Tr}(\mathbf{C}_2 \mathbf{C}_1^{-1}) - \frac{1}{2} \log \det(\mathbf{C}_2 \mathbf{C}_1^{-1}) - n \\ &= \frac{1}{2} \text{Tr}(\mathbf{C}_1 \mathbf{C}_2^{-1}) + \frac{1}{2} \text{Tr}(\mathbf{C}_2 \mathbf{C}_1^{-1}) - n. \quad (3) \end{aligned}$$

**Definition 2.5.** The *Stein*, or  $S$ -, divergence (also known as the *Jensen-Bregman LogDet divergence* [5]) is also obtained from the Burg divergence, but through Jensen-Shannon symmetrization. It can be written as

$$\begin{aligned} S(\mathbf{C}_1, \mathbf{C}_2) &= \frac{1}{2} B\left(\mathbf{C}_1, \frac{\mathbf{C}_1 + \mathbf{C}_2}{2}\right) + \frac{1}{2} B\left(\mathbf{C}_2, \frac{\mathbf{C}_1 + \mathbf{C}_2}{2}\right) \\ &= \log \det\left(\frac{\mathbf{C}_1 + \mathbf{C}_2}{2}\right) - \frac{1}{2} \log \det(\mathbf{C}_1 \mathbf{C}_2). \quad (4) \end{aligned}$$

### 2.1. Properties of Bregman divergences

Here, we present the properties of Bregman divergences that make them a natural choice as a measure of dissimilarity between two CovDs. In particular, we discuss these properties in comparison to the popular Affine Invariant Riemannian Metric (AIRM) on  $\mathcal{S}_{++}^n$  [20], which was introduced as a geometrically-motivated way to analyze CovDs.

#### Invariance to affine transformations:

As indicated by the name, the AIRM was designed to be invariant to affine transformations, which often is an attractive property in computer vision algorithms. In our case, the  $B$ -divergence exhibits the same invariance property. More specifically, given  $\mathbf{A} \in GL(n)$ ,  $B(\mathbf{C}_1, \mathbf{C}_2) = B(\mathbf{A} \mathbf{C}_1 \mathbf{A}^T, \mathbf{A} \mathbf{C}_2 \mathbf{A}^T)$ . This can easily be shown from the definition of the  $B$ -divergence. Since the  $J$ - and  $S$ -divergences are obtained from the  $B$ -divergence, it can easily be verified that they inherit this affine invariance property. Furthermore, these two divergences are also invariant to inversion, i.e.,

$$\begin{aligned} J(\mathbf{C}_1, \mathbf{C}_2) &= J(\mathbf{C}_1^{-1}, \mathbf{C}_2^{-1}) \\ S(\mathbf{C}_1, \mathbf{C}_2) &= S(\mathbf{C}_1^{-1}, \mathbf{C}_2^{-1}). \end{aligned}$$

Finally, we also note that  $B(\mathbf{C}_1, \mathbf{C}_2) = B(\mathbf{C}_2^{-1}, \mathbf{C}_1^{-1})$ .

Divergence Name	Formula	Invariance	P.D. Gaussian Kernel
<b>Frobenius</b>	$\ \mathbf{C}_1 - \mathbf{C}_2\ _F^2$	Rotation	Yes
<b>Burg</b>	$\text{Tr}(\mathbf{C}_1 \mathbf{C}_2^{-1}) - \log \det(\mathbf{C}_1 \mathbf{C}_2^{-1}) - n$	Affine	No
<b>Jeffreys</b>	$\frac{1}{2} \text{Tr}(\mathbf{C}_1 \mathbf{C}_2^{-1} + \mathbf{C}_2 \mathbf{C}_1^{-1}) - n$	Affine	Yes
<b>Stein</b>	$\log \det(\frac{1}{2} \mathbf{C}_1 + \frac{1}{2} \mathbf{C}_2) - \frac{1}{2} \log \det(\mathbf{C}_1 \mathbf{C}_2)$	Affine	Partial

**Table 1:** Properties of several Bregman divergences on  $\mathcal{S}_{++}^n$ .

### Positive definite Gaussian kernel:

Recently, kernel methods have been successfully employed on Riemannian manifolds [8, 11]. In particular, an attractive solution is to form a kernel by replacing the Euclidean distance in the popular Gaussian kernel with a more accurate metric on the manifold. However, the resulting kernel is not necessarily positive definite for any metric. In particular, the AIRM does not yield a positive definite Gaussian kernel in general. In contrast, both the  $J$ - and the  $S$ -divergences admit a Hilbert space embedding via a Gaussian kernel.

More specifically, for the  $J$ -divergence, it was shown in [9] that the kernel

$$k_J(\mathbf{C}_1, \mathbf{C}_2) = \exp\{-\beta J(\mathbf{C}_1, \mathbf{C}_2)\}, \quad (5)$$

is Conditionally Positive Definite (CPD). CPD kernels correspond to Hilbertian metrics and can be exploited in a wide range of machine learning algorithms. An example of this is kernel SVM, whose optimal solution was shown to only depend on the Hilbertian property of the metric [9]. Note that while the kernel  $k_J(\cdot, \cdot)$  was claimed to be positive definite [17], we are not aware of any formal proof of this claim.

For the  $S$ -divergence, the kernel

$$k_S(\mathbf{C}_1, \mathbf{C}_2) = \exp\{-\beta S(\mathbf{C}_1, \mathbf{C}_2)\}, \quad (6)$$

is not positive definite for all  $\beta > 0$ . However, as was shown in [22],  $k_S(\cdot, \cdot)$  is positive definite iff

$$\beta \in \left\{ \frac{1}{2}, \frac{2}{2}, \dots, \frac{n-1}{2} \right\} \cup \left\{ \tau \in \mathbb{R} : \tau > \frac{1}{2}(n-1) \right\}.$$

Note that, here, we are not directly interested in positive definite Gaussian kernels on  $\mathcal{S}_{++}^n$  to derive our infinite-dimensional CovDs, but only to learn a kernel-based classifier with the divergences between our infinite-dimensional CovDs as input. The properties of the Bregman divergences that we use in the remainder of this paper are summarized in Table 1.

### 3. Covariance Descriptors in RKHS

In this section, we show how CovDs can be computed in infinite-dimensional spaces. To this end, we first review some basics on Hilbert spaces.

**Definition 3.1.** A Hilbert space is a (possibly infinite-dimensional) inner product space which is complete with respect to the norm induced by the inner product.

An RKHS is a special type of Hilbert space with the additional property that the inner product can be defined by a bivariate function known as the *reproducing kernel*. For an RKHS  $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$  on a non-empty set  $\mathcal{X}$  with  $\phi : \mathcal{X} \rightarrow \mathcal{H}$  there exists a kernel function  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  such that  $k(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle_{\mathcal{H}}, \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$ . The concept of reproducing kernel is typically employed to recast algorithms that only exploit inner products to high-dimensional spaces (e.g., SVM).

Given these definitions, we now turn to the problem of computing a covariance matrix in an RKHS. Let  $\mathbf{X} = [\mathbf{x}_1 | \mathbf{x}_2 | \dots | \mathbf{x}_m]$  be an  $n \times m$  matrix, obtained by stacking  $m$  independent observations  $\mathbf{x}_i \in \mathbb{R}^n$  from an image or a video. The covariance descriptor  $\mathbf{C} \in \mathcal{S}_{++}^n$  is defined as

$$\mathbf{C} = \frac{1}{m} \sum_{i=1}^m (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T = \mathbf{X} \mathbf{J} \mathbf{J}^T \mathbf{X}^T, \quad (7)$$

where  $\boldsymbol{\mu} = \frac{1}{m} \sum_{i=1}^m \mathbf{x}_i$  is the mean of the observations,  $\mathbf{J} = m^{-3/2}(\mathbf{m} \mathbf{I}_m - \mathbf{1}_{m \times m})$  is a centering matrix, and  $\mathbf{1}_{m \times m}$  is a square matrix with all elements equal to 1.

Let  $\phi : \mathbb{R}^n \rightarrow \mathcal{H}$  be a mapping to an RKHS whose corresponding Hilbert space  $\mathcal{H}$  has dimensionality  $|\mathcal{H}|$  ( $|\mathcal{H}|$  could go to  $\infty$ ). Following Eq. 7, a CovD in this RKHS can be written as

$$\mathbf{C}_{\mathcal{X}} = \Phi_{\mathcal{X}} \mathbf{J} \mathbf{J}^T \Phi_{\mathcal{X}}^T, \quad (8)$$

where  $\Phi_{\mathcal{X}} = [\phi(\mathbf{x}_1) | \phi(\mathbf{x}_2) | \dots | \phi(\mathbf{x}_m)]$ . If  $|\mathcal{H}| > m$ , then  $\mathbf{C}_{\mathcal{X}}$  is rank-deficient, which would make any divergence derived from the Burg divergence indefinite. More precisely, the resulting matrix would be on the boundary of the positive cone, which would make it at an infinite distance from any positive definite matrix, not only for Burg-based divergences, but also according to the AIRM.

Here, we address this issue by exploiting ideas developed in the context of covariance matrix estimation from a limited number of observations [2, 27]. More specifically, we seek to keep the positive eigenvalues of  $\mathbf{C}_{\mathcal{X}}$  intact and replace the zero ones with a very small positive

number  $\rho$ , thus making the CovD positive definite. First, using a standard result [23], we note that the positive eigenvalues of  $\mathbf{C}_X$ , denoted by  $\Lambda_X$ , can be computed from  $\mathbf{J}^T \Phi_X^T \Phi_X \mathbf{J} = \mathbf{J}^T \mathbf{K}_{X,X} \mathbf{J}$ , where  $\mathbf{K}_{X,X}$  is the  $m \times m$  kernel matrix whose elements are defined by the kernel function  $k(\mathbf{x}_i, \mathbf{x}_j)$ . By eigenvalue decomposition, we can write

$$\mathbf{J}^T \mathbf{K}_{X,X} \mathbf{J} = \mathbf{V}_X \Lambda_X \mathbf{V}_X^T. \quad (9)$$

This lets us write a (regularized) estimate of  $\mathbf{C}_X$  as

$$\widehat{\mathbf{C}}_X = \Phi_X \mathbf{W}_X \mathbf{W}_X^T \Phi_X^T + \rho \mathbf{I}_{|\mathcal{H}|}, \quad (10)$$

where

$$\mathbf{W}_X = \mathbf{J} \mathbf{V}_X (\mathbf{I}_X - \rho \Lambda_X^{-1})^{\frac{1}{2}}, \quad (11)$$

with  $\mathbf{I}_X$  the identity matrix whose dimension is the number of positive eigenvalues of  $\mathbf{C}_X$  [27]. Note that this derivation can also be employed to model points in  $\mathcal{H}$  with lower-dimensional latent variables by retaining only the top  $r$  eigenvalues and eigenvectors of  $\mathbf{J}^T \mathbf{K}_{X,X} \mathbf{J}$  to form  $\mathbf{W}_X$  [3].

#### 4. Bregman Divergences in RKHS

In this section, we derive different Bregman divergences for the infinite-dimensional CovDs introduced in Section 3. In these derivations, we will make use of the equivalence

$$\mathbf{W}_X^T \Phi_X^T \Phi_X \mathbf{W}_X = \Lambda_X - \rho \mathbf{I}_X, \quad (12)$$

whose derivation is provided in supplementary material.

##### Euclidean Metric:

The Frobenius norm can easily be computed as

$$\begin{aligned} \delta_e^2(\widehat{\mathbf{C}}_X, \widehat{\mathbf{C}}_Y) &= \|\Phi_X \mathbf{W}_X \mathbf{W}_X^T \Phi_X^T - \Phi_Y \mathbf{W}_Y \mathbf{W}_Y^T \Phi_Y^T\|_F^2 \\ &= \|\Lambda_X - \rho \mathbf{I}_X\|_F^2 + \|\Lambda_Y - \rho \mathbf{I}_Y\|_F^2 - 2\|\mathbf{W}_Y^T \mathbf{K}_{Y,X} \mathbf{W}_X\|_F^2. \end{aligned}$$

Note that, although not a desirable property [20], the Euclidean metric is definite for positive semi-definite matrices, which makes it possible to set  $\rho$  to zero.

##### Burg Divergence:

Using the Sylvester determinant theorem [7], we first note that

$$\begin{aligned} \det(\widehat{\mathbf{C}}_X) &= \det(\Phi_X \mathbf{W}_X \mathbf{W}_X^T \Phi_X^T + \rho \mathbf{I}_{|\mathcal{H}|}) \\ &= \rho^{|\mathcal{H}|} \det\left(\mathbf{I}_X + \frac{1}{\rho} \mathbf{W}_X^T \Phi_X^T \Phi_X \mathbf{W}_X\right) \\ &= \rho^{|\mathcal{H}|} \det\left(\mathbf{I}_X + \frac{1}{\rho} (\Lambda_X - \rho \mathbf{I}_X)\right) \\ &= \rho^{|\mathcal{H}|} \det(\rho^{-1} \Lambda_X). \end{aligned} \quad (13)$$

Then, from the Woodbury matrix identity [7], we have

$$\begin{aligned} \widehat{\mathbf{C}}_Y^{-1} &= \left(\Phi_Y \mathbf{W}_Y \mathbf{W}_Y^T \Phi_Y^T + \rho \mathbf{I}_{|\mathcal{H}|}\right)^{-1} \\ &= \frac{1}{\rho} \mathbf{I}_{|\mathcal{H}|} - \frac{1}{\rho} \Phi_Y \mathbf{W}_Y \Lambda_Y^{-1} \mathbf{W}_Y^T \Phi_Y^T. \end{aligned} \quad (14)$$

This lets us write,

$$\begin{aligned} \text{Tr}(\widehat{\mathbf{C}}_X \widehat{\mathbf{C}}_Y^{-1}) &= |\mathcal{H}| + \text{Tr}\left(\frac{1}{\rho} \Lambda_X - \mathbf{I}_X\right) - \text{Tr}(\mathbf{I}_Y - \rho \Lambda_Y^{-1}) \\ &\quad - \frac{1}{\rho} \text{Tr}\left(\mathbf{W}_X^T \mathbf{K}_{X,Y} \mathbf{W}_Y \Lambda_Y^{-1} \mathbf{W}_Y^T \mathbf{K}_{Y,X} \mathbf{W}_X\right). \end{aligned} \quad (15)$$

By combining Eqs. 13 and 15, we then obtain

$$\begin{aligned} B_{\mathcal{H}}(\widehat{\mathbf{C}}_X, \widehat{\mathbf{C}}_Y) &= \text{Tr}\left(\frac{1}{\rho} \Lambda_X - \mathbf{I}_X\right) - \text{Tr}(\mathbf{I}_Y - \rho \Lambda_Y^{-1}) \\ &\quad - \frac{1}{\rho} \text{Tr}\left(\mathbf{W}_X^T \mathbf{K}_{X,Y} \mathbf{W}_Y \Lambda_Y^{-1} \mathbf{W}_Y^T \mathbf{K}_{Y,X} \mathbf{W}_X\right) \\ &\quad + \log \det(\rho^{-1} \Lambda_Y) - \log \det(\rho^{-1} \Lambda_X). \end{aligned} \quad (16)$$

Note that the Burg divergence is independent of  $|\mathcal{H}|$ . This property is inherited by the Jeffreys and Stein divergences derived below.

##### Jeffreys Divergence:

From the definition in Section 2, the Jeffreys divergence can be obtained directly from the Burg divergence. This yields

$$\begin{aligned} J_{\mathcal{H}}(\widehat{\mathbf{C}}_X, \widehat{\mathbf{C}}_Y) &= \frac{1}{2\rho} \text{Tr}(\Lambda_X - \rho \mathbf{I}_X) + \frac{1}{2\rho} \text{Tr}(\Lambda_Y - \rho \mathbf{I}_Y) \\ &\quad - \frac{1}{2\rho} \text{Tr}\left(\mathbf{W}_X^T \mathbf{K}_{X,Y} \mathbf{W}_Y \Lambda_Y^{-1} \mathbf{W}_Y^T \mathbf{K}_{Y,X} \mathbf{W}_X\right) \\ &\quad - \frac{1}{2\rho} \text{Tr}\left(\mathbf{W}_Y^T \mathbf{K}_{Y,X} \mathbf{W}_X \Lambda_X^{-1} \mathbf{W}_X^T \mathbf{K}_{X,Y} \mathbf{W}_Y\right) \\ &\quad - \frac{1}{2} \text{Tr}(\mathbf{I}_X - \rho \Lambda_X^{-1}) - \frac{1}{2} \text{Tr}(\mathbf{I}_Y - \rho \Lambda_Y^{-1}). \end{aligned} \quad (17)$$

##### Stein Divergence:

To compute the Stein divergence in  $\mathcal{H}$ , let us first define

$$\mathbf{Q} = \begin{bmatrix} \mathbf{W}_X & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_Y \end{bmatrix}. \quad (18)$$

This lets us write

$$\frac{\widehat{\mathbf{C}}_X + \widehat{\mathbf{C}}_Y}{2} = \rho \mathbf{I}_{|\mathcal{H}|} + \frac{1}{2} [\Phi_X \ \Phi_Y] \mathbf{Q} \mathbf{Q}^T \begin{bmatrix} \Phi_X^T \\ \Phi_Y^T \end{bmatrix}. \quad (19)$$

Similarly as in Eq. 13,  $\det((\widehat{\mathbf{C}}_{\mathbf{X}} + \widehat{\mathbf{C}}_{\mathbf{Y}})/2)$  becomes

$$\begin{aligned} & \rho^{|\mathcal{H}|} \det \left( \mathbf{I}_{|\mathcal{H}|} + \frac{1}{2\rho} [\Phi_{\mathbf{X}} \ \Phi_{\mathbf{Y}}] \mathbf{Q} \mathbf{Q}^T \begin{bmatrix} \Phi_{\mathbf{X}}^T \\ \Phi_{\mathbf{Y}}^T \end{bmatrix} \right) \\ &= \rho^{|\mathcal{H}|} \det \left( \mathbf{I}_{\mathbf{X}+\mathbf{Y}} + \frac{1}{2\rho} \mathbf{Q}^T \begin{bmatrix} \Phi_{\mathbf{X}}^T \\ \Phi_{\mathbf{Y}}^T \end{bmatrix} [\Phi_{\mathbf{X}} \ \Phi_{\mathbf{Y}}] \mathbf{Q} \right) \\ &= \rho^{|\mathcal{H}|} \det \left( \mathbf{I}_{\mathbf{X}+\mathbf{Y}} + \frac{1}{2\rho} \mathbf{Q}^T \mathbb{K}_{\mathbf{X},\mathbf{Y}} \mathbf{Q} \right), \end{aligned}$$

where

$$\mathbb{K}_{\mathbf{X},\mathbf{Y}} = \begin{bmatrix} \mathbf{K}_{\mathbf{X},\mathbf{X}} & \mathbf{K}_{\mathbf{X},\mathbf{Y}} \\ \mathbf{K}_{\mathbf{Y},\mathbf{X}} & \mathbf{K}_{\mathbf{Y},\mathbf{Y}} \end{bmatrix}. \quad (20)$$

Therefore, we have

$$\begin{aligned} S_{\mathcal{H}}(\widehat{\mathbf{C}}_{\mathbf{X}}, \widehat{\mathbf{C}}_{\mathbf{Y}}) &= \log \det \left( \mathbf{I}_{\mathbf{X}+\mathbf{Y}} + \frac{1}{2\rho} \mathbf{Q}^T \mathbb{K}_{\mathbf{X},\mathbf{Y}} \mathbf{Q} \right) \\ &\quad - \frac{1}{2} \log \det (\rho^{-1} \Lambda_{\mathbf{X}}) - \frac{1}{2} \log \det (\rho^{-1} \Lambda_{\mathbf{Y}}). \end{aligned} \quad (21)$$

#### 4.1. Practical Considerations

When computing divergences in RKHS, it is desirable to minimize the effect of the parameter  $\rho$ , and thus have divergences that do not depend on its inverse. To this end, let us assume that the same number of eigenvectors were kept to build  $\mathbf{C}_{\mathbf{X}}$  and  $\mathbf{C}_{\mathbf{Y}}$ . In this case, the Stein divergence can be written as

$$\begin{aligned} \widehat{S}_{\mathcal{H}}(\widehat{\mathbf{C}}_{\mathbf{X}}, \widehat{\mathbf{C}}_{\mathbf{Y}}) &= \log \det \left( \rho \mathbf{I}_{\mathbf{X}+\mathbf{Y}} + \frac{1}{2} \mathbf{Q}^T \mathbb{K}_{\mathbf{X},\mathbf{Y}} \mathbf{Q} \right) \\ &\quad - \frac{1}{2} \log \det (\Lambda_{\mathbf{X}}) - \frac{1}{2} \log \det (\Lambda_{\mathbf{Y}}), \end{aligned} \quad (22)$$

where the term  $\rho \mathbf{I}_{\mathbf{X}+\mathbf{Y}}$  can be thought of as a regularizer for  $\frac{1}{2} \mathbf{Q}^T \mathbb{K}_{\mathbf{X},\mathbf{Y}} \mathbf{Q}$ . For the Jeffreys divergence, we can define

$$\begin{aligned} \widehat{J}_{\mathcal{H}}(\widehat{\mathbf{C}}_{\mathbf{X}}, \widehat{\mathbf{C}}_{\mathbf{Y}}) &= \lim_{\rho \rightarrow 0} 2\rho J_{\mathcal{H}}(\widehat{\mathbf{C}}_{\mathbf{X}}, \widehat{\mathbf{C}}_{\mathbf{Y}}) = \\ &\quad - \text{Tr} \left( \mathbf{W}_{\mathbf{X}}^T \mathbf{K}_{\mathbf{X},\mathbf{Y}} \mathbf{W}_{\mathbf{Y}} \Lambda_{\mathbf{Y}}^{-1} \mathbf{W}_{\mathbf{Y}}^T \mathbf{K}_{\mathbf{Y},\mathbf{X}} \mathbf{W}_{\mathbf{X}} \right) \\ &\quad - \text{Tr} \left( \mathbf{W}_{\mathbf{Y}}^T \mathbf{K}_{\mathbf{Y},\mathbf{X}} \mathbf{W}_{\mathbf{X}} \Lambda_{\mathbf{X}}^{-1} \mathbf{W}_{\mathbf{X}}^T \mathbf{K}_{\mathbf{X},\mathbf{Y}} \mathbf{W}_{\mathbf{Y}} \right) \\ &\quad + \text{Tr}(\Lambda_{\mathbf{X}}) + \text{Tr}(\Lambda_{\mathbf{Y}}). \end{aligned} \quad (23)$$

In our experiments, we used the definitions of Eqs. 23 and 22.

#### 4.2. Computational Complexity

Here we compare the complexity of computing  $J_{\mathcal{H}}(\cdot, \cdot)$  and  $S_{\mathcal{H}}(\cdot, \cdot)$  against that of  $J(\cdot, \cdot)$  and  $S(\cdot, \cdot)$ . Let  $\mathbf{X} \in \mathbb{R}^{n \times m}$  and  $\mathbf{Y} \in \mathbb{R}^{n \times m}$  be two given sets of observation, with  $m \gg n$ .

Computing the  $n \times n$  CovDs based on Eq. 7 requires  $O(n^2 m)$ . The inverse of an  $n \times n$  SPD matrix can be computed by Cholesky decomposition in  $\frac{1}{2}n^3$  flops. Therefore,

computing the  $J$ -divergence requires  $2n^2 m + 2n^{2.3} + n^3$  flops, which is dominated by  $2n^2 m$ . The complexity of computing the determinant of an  $n \times n$  matrix by Cholesky decomposition is  $O(\frac{1}{3}n^3)$ . Therefore, computing the  $S$ -divergence requires  $2n^2 m + n^{2.3} + \frac{2}{3}n^3$  flops, which is again dominated by  $2n^2 m$ .

In RKHS, computing  $K_{\mathbf{X},\mathbf{X}}$ ,  $K_{\mathbf{Y},\mathbf{X}}$  and  $K_{\mathbf{X},\mathbf{Y}}$  requires  $m^2$  flops for each matrix. Therefore, evaluating Eq. 9 requires for  $m^3$  flops. Assuming that  $r$ ,  $r < m$ , eigenvectors are used to create  $\mathbf{W}_{\mathbf{X}}$  in Eq. 11, computing  $J_{\mathcal{H}}$  according to Eq. 17 requires  $2m^3 + 3m^2 + 4m^2 r + 2mr^2$ . For  $S_{\mathcal{H}}$ , evaluating Eq. 21 takes  $6m^3 + 3m^2 + 8m^2 r + 8mr^2$  flops.

Generally speaking, the complexity of computing the Jeffreys and Stein divergences in the observation space is linear in  $m$  while it is cubic when working in RKHS. Our experimental evaluation shows, however, that working in RKHS remains practical. To illustrate this, we compare the runtimes required to compute the Stein divergence between 500,000 pairs of CovDs on  $\mathcal{S}_{++}^{10}$  using Eq. 4 and Eq. 22. Each CovD on  $\mathcal{S}_{++}^{10}$  was obtained from  $m = 100$  observations. In the observation space, computing the Stein divergence on an i7 machine using Matlab took 53s. For  $S_{\mathcal{H}}$ , it took 452s, 566s and 868s when keeping 10, 20 and 50 eigenvectors to estimate the covariances, respectively. While slower, these runtimes remain perfectly acceptable, especially when considering the large accuracy gain that working in RKHS entails, as evidenced by our experiments.

### 5. Experimental Evaluation

We now present our empirical results obtained with the infinite-dimensional CovDs and their Bregman divergences defined in Sections 3 and 4. In particular, due to their symmetry and the fact that they yield valid Gaussian kernels, we utilized the Jeffreys and Stein divergences, and relied on two different classifiers for each divergence: A simple nearest neighbor classifier, which clearly evidences the benefits of using infinite-dimensional CovDs, and an SVM classifier with a Gaussian kernel, which further boosts the performance of our infinite-dimensional CovDs.

The different algorithms evaluated in our experiments are referred to as:

**$J/S$ -NN:** Jeffreys/Stein based Nearest Neighbor classifier on CovDs in the observation space.

**$J/S$ -SVM:** Jeffreys/Stein based kernel SVM on CovDs in the observation space.

**$J_{\mathcal{H}}/S_{\mathcal{H}}$ -NN:** Jeffreys/Stein based Nearest Neighbor classifier on infinite-dimensional CovDs.

**$J_{\mathcal{H}}/S_{\mathcal{H}}$ -SVM:** Jeffreys/Stein based kernel SVM on infinite-dimensional CovDs.

We also provide the results of the PLS-based Covariance Discriminant Learning (CDL) technique of [25], which can

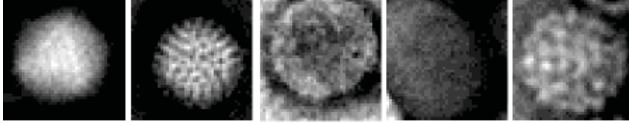


Figure 1: Sample images from the virus dataset [14].

Table 2: Recognition accuracies for the virus dataset [14].

Method	Recognition Accuracy
$B$ -NN	56.6% $\pm$ 2.7
$J$ -NN	60.3% $\pm$ 5.3
$S$ -NN	60.7% $\pm$ 5.4
CDL [25]	69.5% $\pm$ 3.1
$J$ -SVM	73.9% $\pm$ 4.0
$S$ -SVM	76.5% $\pm$ 3.3
$B_{\mathcal{H}}$ -NN	63.5% $\pm$ 3.4
$J_{\mathcal{H}}$ -NN	66.7% $\pm$ 4.2
$S_{\mathcal{H}}$ -NN	67.1% $\pm$ 4.3
$J_{\mathcal{H}}$ -SVM	81.1% $\pm$ 3.4
$S_{\mathcal{H}}$ -SVM	<b>81.2% <math>\pm</math> 2.9</b>

be considered as the state-of-the-art for CovD-based classification. In all our experiments, we used the RBF kernel to create infinite-dimensional CovDs. The parameters of our algorithm, *i.e.*, the RBF bandwidth and the number of eigenvectors  $r$ , were determined by cross-validation.

## 5.1. Virus Classification

As a first experiment, we used the virus dataset [14] which contains 15 different virus classes. Each class has 100 images of size  $41 \times 41$  that were segmented automatically [14]. Samples from the virus dataset are shown in Fig. 1. We used the 10 splits provided with the dataset in a leave-one-out manner, *i.e.*, 10 experiments with 9 splits for training and 1 split as query.

At each pixel  $(u, v)$  of an image, we computed the 25-dimensional feature vector

$$\mathbf{x}_{u,v} = \left[ I_{u,v}, \left| \frac{\partial I}{\partial u} \right|, \left| \frac{\partial I}{\partial v} \right|, \left| \frac{\partial^2 I}{\partial u^2} \right|, \left| \frac{\partial^2 I}{\partial v^2} \right|, |G_{u,v}^{0,0}|, \dots, |G_{u,v}^{4,5}| \right]^T,$$

where  $I_{u,v}$  is the intensity value,  $G_{u,v}^{o,s}$  is the response of a 2D Gabor wavelet [15] with orientation  $o$  and scale  $s$ , and  $|\cdot|$  denotes the magnitude of a complex value. Here, we generated 20 Gabor filters at 4 orientations and 5 scales.

We report the mean recognition accuracies over the 10 runs in Table 2. The NN results clearly show that the CovDs computed in RKHS are more discriminative than the ones built directly from the original features. Note that applying kernel SVM boosts the performance of all the CovDs. Note also that our simple NN scheme in RKHS achieves comparable performance to the more involved CDL. Our

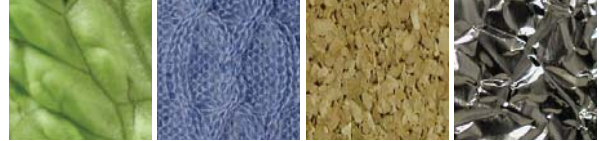


Figure 2: Samples from the KTH-TIPS2b material dataset [4].

Table 3: Recognition accuracies for the KTH-TIPS2b material dataset [4].

Method	Split#1	Split#2	Split#3	Split#4	Average
$J$ -NN	72.6%	72.8%	64.8%	64.3%	68.6%
$S$ -NN	72.5%	73.4%	64.6%	64.6%	68.8%
CDL [25]	83.5%	75.6%	71.5%	74.5%	76.3%
$J$ -SVM	77.4%	76.6%	71.4%	73.3%	74.7%
$S$ -SVM	83.6%	<b>80.9%</b>	73.1%	75.4%	78.3%
$J_{\mathcal{H}}$ -NN	79.1%	75.7%	69.9%	67.7%	73.1%
$S_{\mathcal{H}}$ -NN	78.1%	76.3%	69.2%	67.8%	72.9%
$J_{\mathcal{H}}$ -SVM	<b>85.2%</b>	78.5%	<b>76.4%</b>	79.7%	79.9%
$S_{\mathcal{H}}$ -SVM	85.1%	79.8%	74.0%	<b>81.6%</b>	<b>80.1%</b>

$J_{\mathcal{H}}/S_{\mathcal{H}}$ -SVM methods outperform all the baselines. Here, for each split, the runtimes were on average 130s for the Stein divergence in observation space and 1180s for  $S_{\mathcal{H}}$ , which remains perfectly practical.

In addition to the baselines in Table 2, we evaluated the performance of Local Binary Patterns (LBP) [19] and Gabor filters [15], which are popular methods to analyze textures. With an NN classifier, we obtained accuracies of  $36.8\% \pm 3.9$  and  $33.7\% \pm 4.0$  for LBP and Gabor filters, respectively. This clearly shows the difficulty of this task and the notable improvement achieved by using CovDs.

With this dataset, we also evaluated the performance of the Euclidean metric and asymmetric Burg divergence in RKHS. We obtained 52.4% and 63.5% accuracy for the Euclidean metric and the Burg divergence, respectively. This indicates that a simple Euclidean metric is poorly-suited to handle CovDs. In the remainder of this section, we focus on the Stein and Jeffreys divergences.

## 5.2. Material Categorization

We then used the KTH-TIPS2b dataset [4] to perform material categorization. KTH-TIPS2b contains images of 11 materials captured under 4 different illuminations, in 3 poses and at 9 scales. This yields a total of  $3 \times 4 \times 9 = 108$  images for each sample in a category, with 4 samples per material. We resized the original images to  $128 \times 128$  pixels and generated CovDs from 1024 observations computed on a coarse grid (*i.e.*, every 4 pixels horizontally and vertically). At each point on the grid, we extracted the 23-dimensional feature vector

$$\mathbf{x}_{u,v} = \left[ r_{u,v}, g_{u,v}, b(u,v), |G_{u,v}^{0,0}|, \dots, |G_{u,v}^{4,5}| \right]^T,$$

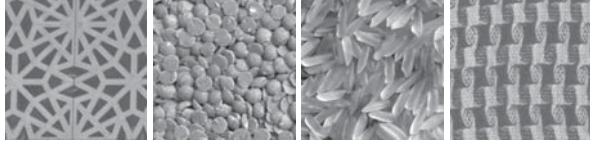


Figure 3: Samples from the Kylberg texture dataset [13].

Table 4: Recognition accuracies for the Kylberg dataset [13].

Method	Recognition Accuracy
$J$ -NN	77.00% $\pm$ 1.40
$S$ -NN	77.02% $\pm$ 1.37
CDL [25]	79.87% $\pm$ 1.06
$J$ -SVM	82.19% $\pm$ 1.30
$S$ -SVM	81.27% $\pm$ 1.07
$J_{\mathcal{H}}$ -NN	84.89% $\pm$ 1.06
$S_{\mathcal{H}}$ -NN	84.91% $\pm$ 1.08
$J_{\mathcal{H}}$ -SVM	91.25% $\pm$ 1.33
$S_{\mathcal{H}}$ -SVM	<b>91.36% <math>\pm</math> 1.27</b>

where  $r_{u,v}$ ,  $g_{u,v}$  and  $b_{u,v}$  are the color intensities, and  $G_{u,v}^{o,s}$  are the same Gabor filter responses as before.

In Table 3, we report the recognition accuracies computed by training on 3 samples per category and testing on the remaining sample. On average, with an NN classifier, our infinite-dimensional CovDs outperform the  $23 \times 23$ -dimensional ones by more than 4%. As before, kernel SVM further improves the performance of all CovDs. This yields a maximum average accuracy of 80.1% for our  $S_{\mathcal{H}}$ -SVM, which, to the best of our knowledge, is state-of-the-art on this dataset [16].

### 5.3. Texture Classification

For texture classification, we used the Kylberg dataset [13] that contains 28 texture classes of different natural and man-made surfaces. Each class has 160 unique samples imaged with and without rotation. Samples from this dataset are shown in Fig. 3.

As in Section 5.2, we resized the images to  $128 \times 128$  pixels and generated CovDs from 1024 observations obtained on a coarse grid. The feature vector at each pixel on this grid was taken as

$$\mathbf{x}_{u,v} = \left[ I_{u,v}, \left| \frac{\partial I}{\partial u} \right|, \left| \frac{\partial I}{\partial v} \right|, \left| \frac{\partial^2 I}{\partial u^2} \right|, \left| \frac{\partial^2 I}{\partial v^2} \right| \right]^T.$$

We randomly selected 5 images in each class for training and used the remaining ones as test data.

In Table 4, we report recognition accuracies averaged over 10 such random partitions. As before, NN on infinite-dimensional CovDs clearly outperforms NN on  $5 \times 5$ -dimensional CovDs. Interestingly, it even outperforms the

Table 5: Recognition accuracies for the ETHZ dataset [6].

Method	Seq#1	Seq#2
$J$ -NN	80.7% $\pm$ 1.5	77.2% $\pm$ 1.2
$S$ -NN	81.3% $\pm$ 1.5	77.9% $\pm$ 1.1
SDALF [1]	83.4% $\pm$ N/A	83.4% $\pm$ N/A
$J$ -SVM	83.4% $\pm$ 1.0	83.1% $\pm$ 1.2
$S$ -SVM	84.4% $\pm$ 1.0	84.2% $\pm$ 1.3
$J_{\mathcal{H}}$ -NN	85.7% $\pm$ 1.7	84.3% $\pm$ 1.8
$S_{\mathcal{H}}$ -NN	85.9% $\pm$ 1.7	84.5% $\pm$ 1.8
$J_{\mathcal{H}}$ -SVM	89.1% $\pm$ 1.1	90.9% $\pm$ 1.1
$S_{\mathcal{H}}$ -SVM	<b>90.2% <math>\pm</math> 1.0</b>	<b>91.4% <math>\pm</math> 0.8</b>

more involved CDL method. With kernel SVM, the accuracies of our  $J_{\mathcal{H}}$  and  $S_{\mathcal{H}}$  divergences are improved to 91%.

### 5.4. Person Re-identification

For person re-identification, we used two sequences from the ETHZ dataset [6]. Sequence 1 contains 83 pedestrians in 4,857 images, and Sequence 2 contains 35 pedestrians in 1,936 images. We resized all images to  $48 \times 24$  pixels, and, at each pixel  $\mathbf{u} = (u, v)$ , computed the 17-dimensional feature vector

$$\mathbf{x}_{\mathbf{u}} = [\mathbf{u}, r_{\mathbf{u}}, g_{\mathbf{u}}, b_{\mathbf{u}}, \dot{r}_{\mathbf{u}}, \dot{g}_{\mathbf{u}}, \dot{b}_{\mathbf{u}}, \ddot{r}_{\mathbf{u}}, \ddot{g}_{\mathbf{u}}, \ddot{b}_{\mathbf{u}}]^T,$$

where  $r_{\mathbf{u}}$ ,  $g_{\mathbf{u}}$  and  $b_{\mathbf{u}}$  are the color intensities, and, e.g., for the  $r$  channel,  $\dot{r}_{\mathbf{u}} = (|\partial r / \partial u|, |\partial r / \partial v|)$  and  $\ddot{r}_{\mathbf{u}} = (|\partial^2 r / \partial u^2|, |\partial^2 r / \partial v^2|)$ . Following [1], we randomly selected 10 images from each subject for training and used the rest for testing.

In Table 5, we report the accuracies averaged over 10 random partitions. In addition to the usual baselines, we report the state-of-the-art results obtained with the Symmetry-Driven Accumulation of Local Features (SDALF) of [1]. Once again, both  $J_{\mathcal{H}}$ -NN and  $S_{\mathcal{H}}$ -NN outperform  $J$ -NN and  $S$ -NN, and similarly for SVM. More importantly,  $J_{\mathcal{H}}$ -NN and  $S_{\mathcal{H}}$ -NN outperform SDALF, and even more so with kernel SVM. In supplementary material, we provide the Cumulative Matching Characteristic (CMC) curves that are commonly used for person re-identification.

### 5.5. Action Recognition from Motion Capture Data

Finally, we performed an experiment on human action recognition from motion capture sequences using the HDM05 database [18], which contains 14 different actions. Each action is represented by the 3D locations of 31 joints over time. In our experiments, we only used the 4 joints corresponding to arms and legs. This let us compute a 12-dimensional feature vector per frame by concatenating the 3D locations of these 4 joints in that frame. The CovDs

**Table 6:** Recognition accuracies for the HDM05 database [18].

Method	Recognition Accuracy
$J$ -NN	47.3% $\pm$ 7.0
$S$ -NN	47.8% $\pm$ 7.4
CDL [25]	65.3% $\pm$ 8.9
$J$ -SVM	50.8% $\pm$ 8.4
$S$ -SVM	56.8% $\pm$ 11.5
<hr/>	
$J_{\mathcal{H}}$ -NN	63.3% $\pm$ 9.4
$S_{\mathcal{H}}$ -NN	65.9% $\pm$ 12.8
<hr/>	
$J_{\mathcal{H}}$ -SVM	70.8% $\pm$ 8.1
$S_{\mathcal{H}}$ -SVM	<b>73.3% <math>\pm</math> 11.4</b>

are then computed over the frames. We used a leave-one-subject-out setup, where 4 out of the 5 available subjects were used for training and the remaining one for testing.

In Table 6, we report the average accuracies over the 5 runs. Again, infinite-dimensional CovDs outperform the ones computed from the original observations and yield the best results when used in conjunction with kernel SVM.

## 6. Conclusions and Future Work

We have introduced an approach to computing infinite-dimensional CovDs, as well as several Bregman divergences to compare them. Our experimental evaluation has demonstrated that the resulting infinite-dimensional CovDs lead to state-of-the-art recognition accuracies on several challenging datasets. In the future, we intend to explore how other types of similarity measures, such as the AIRM, can be computed over infinite-dimensional CovDs. Furthermore, we are interested in studying how the Fréchet mean of a set of infinite-dimensional CovDs can be evaluated. This would allow us to perform clustering, and would therefore pave the way to extending well-known methods, such as bag of words, to infinite dimensional CovDs.

## References

- [1] L. Bazzani, M. Cristani, and V. Murino. Symmetry-driven accumulation of local features for human characterization and re-identification. *Computer Vision and Image Understanding (CVIU)*, 117(2):130–144, 2013.
- [2] P. J. Bickel and E. Levina. Regularized estimation of large covariance matrices. *The Annals of Statistics*, pages 199–227, 2008.
- [3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [4] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *Proc. Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 1597–1604, 2005.
- [5] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos. Jensen-Bregman logdet divergence with application to efficient similarity search for covariance matrices. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(9):2161–2174, 2013.
- [6] A. Ess, B. Leibe, and L. Van Gool. Depth and appearance for mobile scene analysis. In *Proc. Int. Conf. on Computer Vision (ICCV)*, pages 1–8, 2007.
- [7] G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.
- [8] 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 *Proc. European Conf. on Computer Vision (ECCV)*, pages 216–229, 2012.
- [9] M. Hein and O. Bousquet. Hilbertian metrics and positive definite kernels on probability measures. In *Proc. Int. Conf. Artificial Intelligence & Statistics*, pages 136–143, 2005.
- [10] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations. In *Proc. Int. Joint Conference on Artificial Intelligence (IJCAI)*, 2013.
- [11] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi. Kernel methods on the Riemannian manifold of symmetric positive definite matrices. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2013.
- [12] B. Kulis, M. A. Sustik, and I. S. Dhillon. Low-rank kernel learning with Bregman matrix divergences. *Journal of Machine Learning Research (JMLR)*, 10:341–376, 2009.
- [13] G. Kylberg. The kylberg texture dataset v. 1.0. External report (Blue series) 35, Centre for Image Analysis, Swedish University of Agricultural Sciences and Uppsala University, Uppsala, Sweden, September 2011.
- [14] G. Kylberg, M. Uppstroem, K.-O. Hedlund, G. Borgefors, and I.-M. Sintorn. Segmentation of virus particle candidates in transmission electron microscopy images. *Journal of Microscopy*, 245(2):140–147, 2012.
- [15] T. S. Lee. Image representation using 2d Gabor wavelets. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(10):959–971, 1996.
- [16] L. Liu, L. Zhao, Y. Long, G. Kuang, and P. Fieguth. Extended local binary patterns for texture classification. *Image and Vision Computing (IVC)*, 30(2):86–99, 2012.
- [17] P. J. Moreno, P. P. Ho, and N. Vasconcelos. A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2003.
- [18] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation: Mopac database HDM05. Technical Report CG-2007-2, Universität Bonn, 2007.
- [19] T. Ojala, M. Pietikainen, and T. Maenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
- [20] X. Pennec, P. Fillard, and N. Ayache. A Riemannian framework for tensor computing. *Int. Journal of Computer Vision (IJCV)*, 66(1):41–66, 2006.
- [21] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos. Tensor sparse coding for positive definite matrices. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2013.
- [22] S. Sra. A new metric on the manifold of kernel matrices with application to matrix geometric means. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 144–152, 2012.
- [23] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.
- [24] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on Riemannian manifolds. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(10):1713–1727, 2008.
- [25] R. Wang, H. Guo, L. S. Davis, and Q. Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2496–2503, 2012.
- [26] Z. Wang and B. C. Vemuri. An affine invariant tensor dissimilarity measure and its applications to tensor-valued image segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages I–228, 2004.
- [27] S. Zhou and R. Chellappa. From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel hilbert space. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(6):917–929, 2006.