

# From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel Hilbert space

Shaohua Kevin Zhou<sup>1</sup> and Rama Chellappa<sup>2</sup>

<sup>1</sup>Integrated Data Systems Department, Siemens Corporate Research

755 College Road East, Princeton, NJ 08540

Email: {kzhou}@scr.siemens.com

<sup>2</sup>Center for Automation Research and

Department of Electrical and Computer Engineering

University of Maryland, College Park, MD 20742

Email: {rama}@cfar.umd.edu

## Abstract

This paper attacks the problem of characterizing ensemble similarity from sample similarity in a principle manner. Using reproducing kernel as the sample similarity, we propose to use probabilistic distance measure in the so-called reproduced kernel Hilbert space (RKHS) as the ensemble similarity. Assuming normality in the RKHS, we derive analytic expressions for probabilistic distance measures that are commonly used in many applications, such as Chernoff distance (or Bhattacharyya distance as its special case), Kullback-Leibler divergence, etc. Since the reproducing kernel implicitly embeds a nonlinear mapping, we achieve a new approach to study these distances whose feasibility and efficiency is demonstrated using experiments with synthetic and real examples. Further we extend the ensemble similarity to the reproducing kernel for ensemble and study the ensemble similarity for data representations other than vector.

## Index Terms

Ensemble similarity, kernel methods, Chernoff distance, Bhattacharyya distance, Kullback-Leibler (KL) divergence/relative entropy, Patrick-Fisher distance, Mahalonobis distance, reproducing kernel Hilbert space, Gaussian process.

## I. INTRODUCTION

### A. Problem definition

This paper attacks the problem of characterizing ensemble similarity from sample similarity. An ensemble is a collection of entities or samples. Based on the knowledge of the similarity function between any two entities or samples, defined as the *sample similarity*, we are interested in defining the *ensemble similarity* function that calibrates the proximity between two ensembles.

The target problem has a wide range of applications. For example, video retrieval lies on the similarity function between two videos. If we treat an video sequence as an ensemble that consists of multiple video frames (or samples), designing the ensemble similarity function is essential to any video retrieval algorithm. In face recognition from more than one image, the ensemble is a collection of face images that are prepared beforehand from a video sequence or multiple data collections. Often we are able to compare two face images, e.g., using the similarity function arising from a still image based face recognition module. Given an video input, we wish to directly compare two ensembles that requires defining the ensemble similarity.

Let  $\Omega$  denote the space of interest. A *sample* is an element in the space  $\Omega$ . Suppose that  $\alpha \in \Omega$  and  $\beta \in \Omega$  are two samples, the *sample similarity* function is a two-input function  $k(\alpha, \beta)$  that measures the closeness between  $\alpha$  and  $\beta$ . An *ensemble* is an subset of  $\Omega$  that contains multiple samples. Suppose that  $\mathcal{A} = \{\alpha_1, \dots, \alpha_M\}$ , with  $\alpha_i \in \Omega$ , and  $\mathcal{B} = \{\beta_1, \dots, \beta_N\}$ , with  $\beta_j \in \Omega$ , are two ensembles, where  $M$  and  $N$  are not necessarily the same, the ensemble similarity is a two-input function  $k(\mathcal{A}, \mathcal{B})$  that measures the closeness between  $\mathcal{A}$  and  $\mathcal{B}$ .

Starting from the sample similarity  $k(\alpha, \beta)$ , the ideal ensemble similarity  $k(\mathcal{A}, \mathcal{B})$  should utilize all possible pairwise similarity functions between all elements in  $\mathcal{A}$  and  $\mathcal{B}$ . All these similarity functions are encoded in the so-called Gram matrix:

$$\left[ \begin{array}{ccc|ccc} k(\alpha_1, \alpha_1) & \dots & k(\alpha_1, \alpha_M) & k(\alpha_1, \beta_1) & \dots & k(\alpha_1, \beta_N) \\ & & \ddots & & & \ddots \\ k(\alpha_M, \alpha_1) & \dots & k(\alpha_M, \alpha_M) & k(\alpha_M, \beta_1) & \dots & k(\alpha_M, \beta_N) \\ \hline k(\beta_1, \alpha_1) & \dots & k(\beta_1, \alpha_M) & k(\beta_1, \beta_1) & \dots & k(\beta_1, \beta_N) \\ & & \ddots & & & \ddots \\ k(\beta_N, \alpha_1) & \dots & k(\beta_N, \alpha_M) & k(\beta_N, \beta_1) & \dots & k(\beta_N, \beta_N) \end{array} \right].$$

Examples of ad hoc construction of the ensemble similarity function  $k(\mathcal{A}, \mathcal{B})$  include taking the

mean or median of the cross dot product, i.e., the upper right corner of the above Gram matrix. We are interested in proposing a principled solution. The proposed ensemble similarity is related to the spectral analysis of the Gram matrix, i.e., the eigen-decomposition of the Gram matrix.

### B. Probabilistic distance measures

We propose to use probabilistic distance measure (or probabilistic distance in short) as the ensemble similarity. This is from the following interpretation: *An ensemble  $\mathcal{A}$  is thought of as an set of i.i.d. realizations from an underlying probability distribution  $p_{\mathcal{A}}(\alpha)$ .* Therefore, the ensemble similarity is an equivalent description of the distance between two probability distributions, i.e., the probabilistic distance measure. By denoting the probabilistic distance measure by  $J(\mathcal{A}, \mathcal{B})$ , we have

$$k(\mathcal{A}, \mathcal{B}) = J(\mathcal{A}, \mathcal{B}).$$

In this paper, we interchange the use of the two quantities  $k(\mathcal{A}, \mathcal{B})$  and  $J(\mathcal{A}, \mathcal{B})$ . Obviously,  $J(\mathcal{A}, \mathcal{B})$  is a function of  $p_{\mathcal{A}}(\alpha)$  and  $p_{\mathcal{B}}(\beta)$ .

Probabilistic distance measures are important quantities and find their uses in many research areas such as probability and statistics, pattern recognition, information theory, communication and so on. In statistics, the probabilistic distances are often used in asymptotic analysis. In pattern recognition, pattern separability is usually evaluated using probabilistic distance measures [1], [2] such as Chernoff distance or Bhattacharyya distance because they provide bounds for probability of error. In information theory, mutual information, a special example of Kullback-Leibler (KL) distance or relative entropy [3] is a fundamental quantity related to channel capacity. In communication, the KL divergence and Bhattacharyya distance measures are used for signal selection [4].

However, there is a gap between the sample similarity function  $k(\alpha, \beta)$  and the probabilistic distance measure  $J(\mathcal{A}, \mathcal{B})$ . Only when the space  $\Omega$  is a vector space say  $\Omega = \mathcal{R}^d$  and the similarity function is the regular inner product  $k(\alpha, \beta) = \alpha^T \beta$ , the probabilistic distance measures  $J$  coincide with those defined on  $\mathcal{R}^d$ . This is due to the equivalence between the inner product and the distance metric.

$$\|\alpha - \beta\|^2 = \alpha^T \alpha - 2\alpha^T \beta + \beta^T \beta = k(\alpha, \alpha) - 2k(\alpha, \beta) + k(\beta, \beta).$$

This leads to the line of research called kernel methods. In the kernel methods, the sample similarity function  $k(\alpha, \beta)$  evaluates the inner product in a nonlinear feature space  $\mathcal{R}^f$ :

$$k(\alpha, \beta) = \phi(\alpha)^\top \phi(\beta), \quad (1)$$

where  $\phi : \Omega \rightarrow \mathcal{R}^f$  is a nonlinear mapping, where  $f$  is the dimension of the feature space. This is the so-called “kernel trick”. The function  $k(\alpha, \beta)$  in Eq. (1) is referred to as a reproducing kernel function. The nonlinear feature space is referred to as reproducing kernel Hilbert space (RKHS)  $\mathcal{H}_k$  induced by the kernel function  $k$ . For a function to be a reproducing kernel, it must be positive definite, i.e., satisfying the Mercer’s theorem [5]. Refer to [6] for a good review of the properties of RKHS. Obviously, the distance metric in the RKHS can be evaluated

$$\|\phi(\alpha) - \phi(\beta)\|^2 = \phi(\alpha)^\top \phi(\alpha) - 2\phi(\alpha)^\top \phi(\beta) + \phi(\beta)^\top \phi(\beta) = k(\alpha, \alpha) - 2k(\alpha, \beta) + k(\beta, \beta). \quad (2)$$

In this paper, we investigate the use of the reproducing kernel as the sample similarity function and derive the probabilistic distance measures in the RKHS as the ensemble similarity function. In particular, assuming normality in the RKHS, we are able to derive analytic expressions for the Chernoff distance, the Bhattacharyya distance, the (symmetric) KL divergence, etc.

### C. Insights

Our analysis provides additional insights to the following issues.

- *Nonlinear data structure.*

By nonlinear data structure, we mean that if conventional linear modeling techniques, such as fitting the Gaussian density, are used, the responses are badly approximated. High-order statistical information plays an essential role in modeling the nonlinearity. Direct evaluation of probabilistic distances between nonlinear data structures in the original data space is nontrivial since they involve integrals. Only within certain parametric families, say the widely-used normal density, we have analytic expressions for probability distances. However, the normal density employs only up to second-order statistics and is hence rather limited when confronted with a nonlinear data structure. To absorb the nonlinearity, mixture models or non-parametric densities are used in practice. For such cases, one has to resort to numerical methods for computing the probabilistic distances. Such computation is not robust since two approximations are invoked: one in estimating the density and the other one in evaluating the numerical integral.

In this paper, we model the nonlinearity through a different approach: kernel methods. Since a nonlinear function is used, albeit in an implicit fashion, The derived probabilistic distances account for nonlinearity or high-order statistical characteristics of the data. Thus we achieve a new approach to study these distances and investigate their uses in a different space.

- *Normality in RKHS.*

Our computation depends on the artificial assumption that the data is normal in the RKHS. This assumption has been implicitly used in many kernel methods such as [7], [8]. In [7], principal component analysis (PCA) is operated in the RKHS. Even though it seems that PCA needs only the covariance matrix without the normal assumption, it is the deviation of the data from normality in the original space that drives us to search for principal components in the nonlinear feature space. In [8], discriminant analysis is performed in the feature space. We know that discriminant analysis had its origins in a two-class problem by assuming that each class is distributed as Gaussian with a common covariance matrix. Recently, the normal assumption has been directly adopted in the literature [9], [10], [11]. In [9], [10], it is used to compute the mutual information between two Gaussian random vectors in the RKHS. In [11], it is used to define the so-called Bhattacharyya kernel. In principle, the normal assumption in the RKHS is connected to a Gaussian process argument [11]. In [12], the normality is justified through a Wishart process.

The induced RKHS is certainly limited by the number of available samples. Therefore a regularized covariance matrix is needed in [9], [10], [11]. In this paper, we propose a novel way to regularize the covariance matrix that enables us to study certain limiting behaviors.

- *Reproducing kernel for ensemble.*

If the ensemble similarity function satisfies positive definiteness that characterizes the reproducing kernel function, then it becomes a *kernel function for ensemble* (or *ensemble kernel*). The kernel function for ensemble can be readily used in a classification scheme such as support vector machine (SVM) [13] to classify data that is represented by ensemble.

- *Data representation*

There is no restriction on the space  $\Omega$ , i.e., it is not necessarily a vector space. Real applications call for different data representations. While a vector is a very conventional way to represent data, it is a recent trend to define data-dependent kernel function. For example, alternative representations include strings [14], graphs [15], lattices [16], statistical manifolds

[17], [18], [19], and so on [20], [21]. If there are means to define the reproducing kernel for these representation, the proposed probabilistic distances are universally applicable to these representations too. For example, we can calibrate the similarity between two collections of graphs by using the kernel function defined in [15] and the probabilistic distance measures proposed in this paper. In other words, we implicitly define a distribution for data of arbitrary representation through the reproducing kernel function.

#### D. Paper organization

This paper is organized as follows. Section II introduces several probabilistic distances often used in the literature. Section III elaborates the derivations of the probabilistic distances in the RKHS and their characteristics. Section IV demonstrates the feasibility and efficiency of the proposed approach using experiments with synthetic and real examples. Section V concludes the paper.

## II. PROBABILISTIC DISTANCES IN $\mathcal{R}^d$

Consider a two-class problem and suppose that class 1 has density  $\mathfrak{p}_1(\mathbf{x})$  and class 2  $\mathfrak{p}_2(\mathbf{x})$ , both defined on  $\mathcal{R}^d$ . Table I defines a list of probabilistic distance measures often found in the literature [1]. It is obvious that

- 1) The Bhattacharyya distance is a special case of the Chernoff distance with  $\alpha_1 = \alpha_2 = 1/2$ .
- 2) The Matusita distance, also known as the Hellinger distance, is related to the Bhattacharyya distance as follows:

$$J_T = \{2[1 - \exp(-J_B)]\}^{1/2}.$$

- 3) The relationship between the Kullback-Leibler (KL) divergence or relative entropy and the symmetric KL divergence distance is that

$$J_D(\mathfrak{p}_1, \mathfrak{p}_2) = J_R(\mathfrak{p}_1||\mathfrak{p}_2) + J_R(\mathfrak{p}_2||\mathfrak{p}_1).$$

- 4) The Kolmogorov distance is a special case of the Lissack-Fu distance with  $\alpha_1 = 1$ .

Other interesting properties of these distances can be found in [1], [4].

As mentioned earlier, computing the above probabilistic distance measures is nontrivial. Only within certain parametric families, say the Gaussian density, we know how to analytically

Distance Type	Definition
Chernoff distance [22]	$J_C(\mathfrak{p}_1, \mathfrak{p}_2) = -\log\{\int_{\mathbf{x}} \mathfrak{p}_1^{\alpha_2}(\mathbf{x})\mathfrak{p}_2^{\alpha_1}(\mathbf{x})d\mathbf{x}\}$
Bhattacharyya distance [23]	$J_B(\mathfrak{p}_1, \mathfrak{p}_2) = -\log\{\int_{\mathbf{x}} [\mathfrak{p}_1(\mathbf{x})\mathfrak{p}_2(\mathbf{x})]^{1/2}d\mathbf{x}\}$
Matusita distance [24]	$J_T(\mathfrak{p}_1, \mathfrak{p}_2) = \{\int_{\mathbf{x}} [\sqrt{\mathfrak{p}_1(\mathbf{x})} - \sqrt{\mathfrak{p}_2(\mathbf{x})}]^2 d\mathbf{x}\}^{1/2}$
KL divergence [3]	$J_R(\mathfrak{p}_1  \mathfrak{p}_2) = \int_{\mathbf{x}} \mathfrak{p}_1(\mathbf{x}) \log\{\frac{\mathfrak{p}_1(\mathbf{x})}{\mathfrak{p}_2(\mathbf{x})}\}d\mathbf{x}$
Symmetric KL divergence [3]	$J_D(\mathfrak{p}_1, \mathfrak{p}_2) = \int_{\mathbf{x}} [\mathfrak{p}_1(\mathbf{x}) - \mathfrak{p}_2(\mathbf{x})] \log \frac{\mathfrak{p}_1(\mathbf{x})}{\mathfrak{p}_2(\mathbf{x})} d\mathbf{x}$
Patrick-Fisher distance [25]	$J_P(\mathfrak{p}_1, \mathfrak{p}_2) = \{\int_{\mathbf{x}} [\mathfrak{p}_1(\mathbf{x})\pi_1 - \mathfrak{p}_2(\mathbf{x})\pi_2]^2 d\mathbf{x}\}^{1/2}$
Lissack-Fu distance [26]	$J_L(\mathfrak{p}_1, \mathfrak{p}_2) = \int_{\mathbf{x}}  \mathfrak{p}_1(\mathbf{x})\pi_1 - \mathfrak{p}_2(\mathbf{x})\pi_2 ^{\alpha_1} [\mathfrak{p}_1(\mathbf{x})\pi_1 + \mathfrak{p}_2(\mathbf{x})\pi_2]^{\alpha_2} d\mathbf{x}$
Kolmogorov distance [27]	$J_K(\mathfrak{p}_1, \mathfrak{p}_2) = \int_{\mathbf{x}}  \mathfrak{p}_1(\mathbf{x})\pi_1 - \mathfrak{p}_2(\mathbf{x})\pi_2  d\mathbf{x}$

TABLE I

A list of probabilistic distances and their definitions, where  $0 < \alpha_1, \alpha_2 < 1$  and  $\alpha_1 + \alpha_2 = 1$ .

compute some of the above defined distance measures. Suppose that  $\mathbb{N}(\mathbf{x}; \mu, \Sigma)$  with  $\mathbf{x} \in \mathcal{R}^d$  is a multivariate Gaussian density defined as

$$\mathbb{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu)\right\},$$

where  $\mathbf{x} \in \mathcal{R}^d$  and  $|\cdot|$  is matrix determinant. With  $\mathfrak{p}_1(\mathbf{x}) = \mathbb{N}(\mathbf{x}; \mu_1, \Sigma_1)$  and  $\mathfrak{p}_2(\mathbf{x}) = \mathbb{N}(\mathbf{x}; \mu_2, \Sigma_2)$ , Table II lists analytic expressions of some probabilistic distances between two Gaussian densities. When the covariance matrices for two densities are the same, i.e.,  $\Sigma_1 = \Sigma_2 = \Sigma$ , the Bhattacharyya distance and the symmetric divergence reduce to the Mahalanobis distance [28]:

$$J_M = J_D = 8J_B.$$

### III. PROBABILISTIC DISTANCES IN RKHS

In this section, we first illustrate the computational details of the probabilistic distances in the RKHS. We then study the limiting behaviors of the probabilistic distances in the RKHS presented when the variance of the isotropic noise component  $\rho$  approaches zero. Finally, we highlight some extensions related to these distances.

Distance Type	Analytic Expression
Chernoff distance	$J_C(\mathcal{P}_1, \mathcal{P}_2) = \frac{1}{2}\alpha_1\alpha_2(\mu_1 - \mu_2)^\top[\alpha_1\Sigma_1 + \alpha_2\Sigma_2]^{-1}(\mu_1 - \mu_2) + \frac{1}{2}\log\frac{ \alpha_1\Sigma_1 + \alpha_2\Sigma_2 }{ \Sigma_1 ^{\alpha_1} \Sigma_2 ^{\alpha_2}}$
Bhattacharyya distance	$J_B(\mathcal{P}_1, \mathcal{P}_2) = \frac{1}{8}(\mu_1 - \mu_2)^\top[\frac{1}{2}(\Sigma_1 + \Sigma_2)]^{-1}(\mu_1 - \mu_2) + \frac{1}{2}\log\frac{ \frac{1}{2}(\Sigma_1 + \Sigma_2) }{ \Sigma_1 ^{1/2} \Sigma_2 ^{1/2}}$
KL divergence	$J_R(\mathcal{P}_1  \mathcal{P}_2) = \frac{1}{2}(\mu_1 - \mu_2)^\top\Sigma_2^{-1}(\mu_1 - \mu_2) + \frac{1}{2}\log\frac{ \Sigma_2 }{ \Sigma_1 } + \frac{1}{2}\text{tr}[\Sigma_1\Sigma_2^{-1} - \mathbf{I}_d]$
Symmetric KL divergence	$J_D(\mathcal{P}_1, \mathcal{P}_2) = \frac{1}{2}(\mu_1 - \mu_2)^\top(\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) + \frac{1}{2}\text{tr}[\Sigma_1^{-1}\Sigma_2 + \Sigma_2^{-1}\Sigma_1 - 2\mathbf{I}_d]$
Patrick-Fisher distance	$J_P(\mathcal{P}_1, \mathcal{P}_2) = [(2\pi)^d 2\Sigma_1 ]^{-1/2} + [(2\pi)^d 2\Sigma_2 ]^{-1/2} - 2[(2\pi)^d \Sigma_1 + \Sigma_2 ]^{-1/2} \exp\{-\frac{1}{2}(\mu_1 - \mu_2)^\top(\Sigma_1 + \Sigma_2)^{-1}(\mu_1 - \mu_2)\}$
Mahalanobis distance	$J_M(\mathcal{P}_1, \mathcal{P}_2) = (\mu_1 - \mu_2)^\top\Sigma^{-1}(\mu_1 - \mu_2)$

TABLE II

Analytic expressions of probabilistic distances between two normal densities.

### A. Mean and covariance matrix in RKHS

Computing the probabilistic distance measures requires first- and second-order statistics in the RKHS, as shown in Section II. In practice, we have to estimate these statistics from a set of training samples.

Suppose that  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$  are given observations in the original data space  $\Omega$ . We operate in the RKHS  $\mathcal{R}^f$  induced by a nonlinear mapping function  $\phi : \Omega \rightarrow \mathcal{R}^f$ , where  $f$  is unknown and could even be infinite. The training samples in  $\mathcal{R}^f$  are denoted by

$$\Phi_{f \times N} = [\phi_1, \phi_2, \dots, \phi_N],$$

where  $\phi_n = \phi(\mathbf{x}_n) \in \mathcal{R}^f$ . The quantity  $\Phi$  is a hypothesized one in the sense that we cannot evaluate it in practice. As in any kernel method, all computations are conducted through the Gram matrix

$$\mathbf{K} = \Phi^\top \Phi$$

that can be evaluated using the ‘kernel trick’ [13], [7], i.e., the  $ij^{th}$  entry of the Gram matrix is  $\phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$  that can be easily computed as  $k(\mathbf{x}_i, \mathbf{x}_j)$ , where  $k(\cdot, \cdot)$  is a pre-specified kernel function. Two widely-used examples of  $k(\mathbf{x}, \mathbf{y})$  for vector inputs are the polynomial kernel and the radial basis function (RBF) kernel:

$$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^\top \mathbf{y} + \theta)^p; \quad k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{y}\|^2\right) \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{R}^d, \quad (3)$$

where  $\sigma$  controls the kernel width. The RKHS corresponding to the RBF kernel is infinite-dimensional, i.e.,  $f = \infty$ .

Following the maximum likelihood estimate (MLE) theory, the mean  $\mu$  and the covariance matrix  $\Sigma$  are estimated as

$$\begin{aligned}\hat{\mu} &= N^{-1} \sum_{n=1}^N \phi(\mathbf{x}_n) = \Phi \mathbf{s}, \\ \hat{\Sigma} &= N^{-1} \sum_{n=1}^N (\phi_n - \mu)(\phi_n - \mu)^\top = \Phi \mathbf{J} \mathbf{J}^\top \Phi^\top,\end{aligned}\quad (4)$$

where the weight vector  $\mathbf{s}_{N \times 1} = N^{-1} \mathbf{1}$  with  $\mathbf{1}$  being a vector of 1's and  $\mathbf{J}$  is an  $N \times N$  centering matrix given as  $\mathbf{J} = N^{-1/2}(\mathbf{I}_N - \mathbf{s} \mathbf{1}^\top)$ .

*1) Covariance matrix approximation:* The covariance matrix  $\hat{\Sigma}$  in (4) is rank-deficient since often  $f \gg N$ . Thus, inverting such a matrix is impossible and an approximation to the covariance matrix is necessary. Later we show that this approximation can be exact by studying its limiting behavior.

Such an approximation  $\mathbf{C}$  should ideally possess the following features: (i) It keeps the principal structure of the covariance matrix  $\hat{\Sigma}$ . In other words, the dominant eigenvalues and eigenvectors of  $\hat{\Sigma}$  and  $\mathbf{C}$  should be same. (ii) It is compact and regularized. The compactness is inspired by the fact that the smallest eigenvalues of the covariance matrix are very close to zero. The regularity is always desirable in the approximation theory. (iii) It is easy to invert.

We propose to use the following approximation form:

$$\mathbf{C} = \Phi \mathbf{J} \mathbf{Q} \mathbf{Q}^\top \mathbf{J}^\top \Phi^\top + \rho \mathbf{I}_f = \mathbf{W} \mathbf{W}^\top + \rho \mathbf{I}_f = \Phi \mathbf{A} \Phi^\top + \rho \mathbf{I}_f, \quad (5)$$

where  $\mathbf{Q}$  is an  $N \times r$  matrix,

$$\mathbf{W}_{f \times r} \equiv \Phi \mathbf{J} \mathbf{Q}, \quad \mathbf{A}_{N \times N} \equiv \mathbf{J} \mathbf{Q} \mathbf{Q}^\top \mathbf{J}^\top,$$

and  $\rho > 0$  is a pre-specified constant. Typically,  $r \ll N \ll f$ . First, as shown in Appendix-I, an appropriate  $\mathbf{Q}$  can be derived from the Gram matrix  $\mathbf{k}$  so that the top  $r$  eigenpairs of  $\Sigma$  are maintained. Hence, if  $\rho = 0$ , we exactly maintain the subspace containing the top  $r$  eigenpairs. Secondly,  $\mathbf{C}$  is regularized and its compactness is achieved through the  $\mathbf{Q}$  matrix. Finally, inverting  $\mathbf{C}$  is also easy by using the Woodbury formula,

$$\mathbf{C}^{-1} = (\rho \mathbf{I}_f + \mathbf{W} \mathbf{W}^\top)^{-1} = \rho^{-1} (\mathbf{I}_f - \mathbf{W} \mathbf{M}^{-1} \mathbf{W}^\top) = \rho^{-1} (\mathbf{I}_f - \Phi \mathbf{B} \Phi^\top),$$

where

$$\mathbf{B}_{N \times N} \equiv \mathbf{J} \mathbf{Q} \mathbf{M}^{-1} \mathbf{Q}^T \mathbf{J}^T$$

and the matrix  $\mathbf{M}_{r \times r}$  can be thought as a “reciprocal” matrix for  $\mathbf{C}$ ,

$$\mathbf{M}_{r \times r} \equiv \rho \mathbf{I}_r + \mathbf{W}^T \mathbf{W} = \rho \mathbf{I}_r + \mathbf{L},$$

with

$$\mathbf{L}_{r \times r} \equiv \mathbf{W}^T \mathbf{W} = \mathbf{Q}^T \mathbf{J}^T \Phi^T \Phi \mathbf{J} \mathbf{Q}.$$

In ridge regression [29], the form of  $\mathbf{C}_1 = \Phi \mathbf{J} \mathbf{J}^T \Phi^T + \rho \mathbf{I}_f$  is used to provide a regularized approximation. This has a smoothness interpretation of the regression parameters. However, the eigenvalues of  $\mathbf{C}_1$  always increase those of  $\Sigma$  by an amount of  $\rho$  but the eigenvectors of the  $\mathbf{C}_1$  are the same as those of  $\Sigma$ . Although  $\mathbf{C}$  is in a compact form and also regularized, inversion of the  $\mathbf{C}_1$  matrix involves inverting an  $N \times N$  matrix, which is still prohibitive in real applications with a large  $N$ , whereas  $\mathbf{C}^{-1}$  involves inverting only a  $r \times r$   $\mathbf{M}$  matrix. This form of  $\mathbf{C}_1$  is also used in [9] and in [11]. In [30] the covariance matrix  $\Sigma$  is approximated as  $\mathbf{C}_2 = \Phi \mathbf{J} \mathbf{D} \mathbf{J}^T \Phi^T + \rho \mathbf{I}_f$ , where  $\mathbf{D}$  is a diagonal matrix whose many diagonal entries empirically shown to be zero. However, we do not enforce  $\mathbf{D}$  to be diagonal.

### B. Computations of probabilistic distances in RKHS

Since the probabilistic distances involve two densities  $\mathfrak{p}_1$  and  $\mathfrak{p}_2$ , we need two sets of training samples:  $\Phi_1$  for  $\mathfrak{p}_1$  and  $\Phi_2$  for  $\mathfrak{p}_2$ . For each density  $\mathfrak{p}_i$ , we can find its corresponding  $\mathbf{s}_i$ ,  $\mathbf{J}_i$ ,  $\mu_i$ ,  $\Sigma_i$ ,  $\mathbf{K}_i$ ,  $\mathbf{C}_i$ ,  $\mathbf{V}_{r_i, i}$ ,  $\Lambda_{r_i, i} = \text{Diag}[\lambda_{1, i}, \lambda_{2, i}, \dots, \lambda_{r_i, i}]$ ,  $\mathbf{Q}_i$ ,  $\mathbf{A}_i$ ,  $\mathbf{B}_i$ , etc., by keeping the top  $r_i$  principal components. In general, we can have  $r_1 \neq r_2$  and  $N_1 \neq N_2$  with  $N_i$  being the number of samples for the  $i^{\text{th}}$  density. In addition, we define the following dot product matrix:

$$\begin{bmatrix} \Phi_1^T \\ \Phi_2^T \end{bmatrix} [\Phi_1 \ \Phi_2] = \begin{bmatrix} \Phi_1^T \Phi_1 & \Phi_1^T \Phi_2 \\ \Phi_2^T \Phi_1 & \Phi_2^T \Phi_2 \end{bmatrix} \equiv \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix}, \quad (6)$$

where  $\mathbf{K}_{ij} \equiv \Phi_i^T \Phi_j$  and  $\mathbf{K}_{21} = \mathbf{K}_{12}^T$ .

1) *The Chernoff and Bhattacharyya distances:* As mentioned before, the Bhattacharyya distance is a special case of the Chernoff distance with  $\alpha_1 = \alpha_2 = 1/2$ . Hence, we focus on only the Chernoff distance.

The key quantity in computing the Chernoff distance is  $\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2$  with  $\alpha_1 + \alpha_2 = 1$ . Appendix-II presents the detailed computation.

$$\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2 = \alpha_1 \{\rho \mathbf{I}_f + \Phi_1 \mathbf{A}_1 \Phi_1^\top\} + \alpha_2 \{\rho \mathbf{I}_f + \Phi_2 \mathbf{A}_2 \Phi_2^\top\} = \rho \mathbf{I}_f + [\Phi_1 \ \Phi_2] \mathbf{A}_{ch} \begin{bmatrix} \Phi_1^\top \\ \Phi_2^\top \end{bmatrix},$$

where the matrix  $\mathbf{A}_{ch}$  is rank-deficient since  $\mathbf{A}_{ch} = \mathbf{P} \mathbf{P}^\top$  with

$$\mathbf{P}_{(N_1+N_2) \times (r_1+r_2)} \equiv \begin{bmatrix} \sqrt{\alpha_1} \mathbf{J}_1 \mathbf{Q}_1 & \mathbf{0} \\ \mathbf{0} & \sqrt{\alpha_2} \mathbf{J}_2 \mathbf{Q}_2 \end{bmatrix}.$$

Therefore, the  $\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2$  matrix is of such a form that we can easily find its determinant and inverse. The determinant  $|\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2|$  is given by

$$|\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2| = \rho^{f-(r_1+r_2)} |\rho \mathbf{I}_{r_1+r_2} + \mathbf{L}_{ch}| = \rho^{f-(r_1+r_2)} \prod_{i=1}^{r_1+r_2} (\tau_i + \rho),$$

where  $\{\tau_i; i = 1, \dots, r_1 + r_2\}$  are eigenvalues of the  $\mathbf{L}_{ch}$  matrix of size  $(r_1 + r_2) \times (r_1 + r_2)$  that is given by

$$\mathbf{L}_{ch} = \mathbf{P}^\top \begin{bmatrix} \Phi_1^\top \\ \Phi_2^\top \end{bmatrix} [\Phi_1 \ \Phi_2] \mathbf{P} = \mathbf{P}^\top \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \mathbf{P}. \quad (7)$$

The inverse  $\{\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2\}^{-1}$  is given by

$$\{\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2\}^{-1} = \rho^{-1} \{ \mathbf{I}_f - [\Phi_1 \ \Phi_2] \mathbf{B}_{ch} \begin{bmatrix} \Phi_1^\top \\ \Phi_2^\top \end{bmatrix} \}, \quad \mathbf{B}_{ch} = \mathbf{P} (\rho \mathbf{I}_{r_1+r_2} + \mathbf{L}_{ch})^{-1} \mathbf{P}^\top. \quad (8)$$

It is now easy to compute the following two quantities involved in computing the Chernoff distance:  $\mu_i^\top \{\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2\}^{-1}$  and  $\log \frac{|\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2|}{|\mathbf{C}_1|^{\alpha_1} |\mathbf{C}_2|^{\alpha_2}}$ .

$$\mu_i^\top \{\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2\}^{-1} \mu_j = \mathbf{s}_i^\top \Phi_i^\top \rho^{-1} \{ \mathbf{I}_f - [\Phi_1 \ \Phi_2] \mathbf{B}_{ch} \begin{bmatrix} \Phi_1^\top \\ \Phi_2^\top \end{bmatrix} \} \Phi_j \mathbf{s}_j \equiv \rho^{-1} \xi_{ij} \quad (9)$$

where  $\xi_{ij}$  is defined in Appendix-II.

$$\log \frac{|\alpha_1 \mathbf{C}_1 + \alpha_2 \mathbf{C}_2|}{|\mathbf{C}_1|^{\alpha_1} |\mathbf{C}_2|^{\alpha_2}} = \alpha_1 \sum_{i=1}^{r_1+r_2} \log \frac{\rho + \tau_i}{\lambda_{i,1}} + \alpha_2 \sum_{i=1}^{r_1+r_2} \log \frac{\rho + \tau_i}{\lambda_{i,2}},$$

- 1) For each class  $i$ , since we know the number of training data points, i.e.  $N_i$ , we compute the weight vector  $\mathbf{s}_i$  and the centering matrix  $\mathbf{J}_i$ .
- 2) As in any kernel method, the key quantity is the Gram matrix that is defined in (6) and can be pre-computed. Starting from the Gram matrix that contains  $\mathbf{K}_1$ ,  $\mathbf{K}_2$ , and  $\mathbf{K}_{12}$ , we compute the eigenvalues and eigenvectors of  $\mathbf{K}_1$  and  $\mathbf{K}_2$ , that are encoded in  $\mathbf{V}_{r_1,1}$ ,  $\mathbf{\Lambda}_{r_1,1}$ ,  $\mathbf{V}_{r_2,2}$  and  $\mathbf{\Lambda}_{r_2,2}$ . The rest of computations just follows.
- 3) Compute (i) the  $\mathbf{Q}_1$  and  $\mathbf{Q}_2$  matrices using the method in Appendix-I, i.e., Eq. (18); (ii) the  $\mathbf{L}_{ch}$  matrix using (7) and its eigenvalues  $\{\tau_i; i = 1, 2, \dots, r_1 + r_2\}$ ; (iii) the  $\mathbf{B}_{ch}$  matrix using (8); and (iv) the values of  $\xi_{11}$ ,  $\xi_{22}$  and  $\xi_{12}$  using (9).
- 4) Finally, compute the Chernoff distance in the RKHS  $J_C$  using (10).

Fig. 1. Summary of computing the Chernoff distance in RKHS.

where

$$\lambda_{i,j} = \begin{cases} \lambda_{i,j} & \text{when } i = 1, \dots, r_j; \\ \rho & \text{when } i = r_j + 1, \dots, r_1 + r_2. \end{cases}$$

with  $\{\lambda_{i,j}; i = 1, \dots, r_j\}$  being the eigenvalues for  $\mathbf{C}_j$ .

Finally, we compute the Chernoff distance as follows:

$$2J_C(\mathbf{p}_1, \mathbf{p}_2) = \rho^{-1} \alpha_1 \alpha_2 \{\xi_{11} + \xi_{22} - 2\xi_{12}\} + \alpha_1 \sum_{i=1}^{r_1+r_2} \log \frac{\rho + \tau_i}{\lambda_{i,1}} + \alpha_2 \sum_{i=1}^{r_1+r_2} \log \frac{\rho + \tau_i}{\lambda_{i,2}} \quad (10)$$

Note that the dimensionality  $f$  disappears in our computation. This is needed since  $f$  is an unknown quantity and could be infinite. Figure 1 summarizes the computation of the Chernoff distance in RKHS, assuming that the values of  $r_1$ ,  $r_2$  and  $\rho$  are pre-specified. Clearly, the computation originates from the Gram matrix and its eigen-analysis, convincing our claim made in Section I.

2) *The Mahalanobis distance:* In order to compute the Mahalanobis distance, we assume that the covariance matrices for two classes are the same. In practice, we estimate the common covariance matrix  $\Sigma$  from the data, Suppose that the class-specific covariance matrices  $\Sigma_1$  and  $\Sigma_2$  estimated from the training data, the MLE for the common covariance matrix is

$$\Sigma = \frac{N_1}{N} \Sigma_1 + \frac{N_2}{N} \Sigma_2.$$

Again, we need to approximate  $\Sigma$  to avoid singularity. The approximation  $\mathbf{C}$  is given by  $\mathbf{C} = \frac{N_1}{N} \mathbf{C}_1 + \frac{N_2}{N} \mathbf{C}_2$ .

Therefore, the Mahalanobis distance is proportional to the first term in the Chernoff distance with  $\alpha_1 = \frac{N_1}{N}$  and  $\alpha_2 = \frac{N_2}{N}$ , e.g.,

$$J_M(\mathbf{p}_1, \mathbf{p}_2) = \rho^{-1} \{\xi_{11} + \xi_{22} - 2\xi_{12}\}.$$

3) *The KL divergence:* Computing the KL divergence in the RKHS is just to collect terms like  $\mu_i^\top \mathbf{C}_j^{-1} \mu_k$  and  $\text{tr}\{\mathbf{C}_i \mathbf{C}_j^{-1}\}$ . Detailed computation is shown in Appendix-II.

$$\mu_i^\top \mathbf{C}_j^{-1} \mu_k = \mathbf{s}_i^\top \Phi_i^\top \rho^{-1} (\mathbb{I}_f - \Phi_j \mathbf{B}_j \Phi_j^\top) \Phi_k \equiv \rho^{-1} \theta_{ijk}, \quad (11)$$

$$\text{tr}[\mathbf{C}_i \mathbf{C}_j^{-1}] = \rho^{-1} \{\text{tr}[\Lambda_{r_i, i}] - \eta_{ij}\} + \rho \text{tr}[\Lambda_{r_j, j}^{-1}] + f - (r_i + r_j), \quad (12)$$

where  $\theta_{ijk}$  and  $\eta_{ij}$  are defined in Appendix-II.

Finally, we obtain the KL divergence and its symmetric version in the RKHS by substituting (11) and (12) into those in Table II with  $d$  replaced by  $f$ ,

$$\begin{aligned} 2J_R(\mathbf{p}_1 || \mathbf{p}_2) &= \rho^{-1} \{\theta_{121} + \theta_{222} - \theta_{122} - \theta_{221}\} + \{\log |\Lambda_{r_2, 2}| - \log |\Lambda_{r_1, 1}|\} \\ &+ (r_1 - r_2) \log \rho + \rho^{-1} \{\text{tr}[\Lambda_{r_1, 1}] - \eta_{12}\} + \rho \{\text{tr}[\Lambda_{r_2, 2}^{-1}]\} - (r_1 + r_2). \end{aligned}$$

$$2J_D(\mathbf{p}_1, \mathbf{p}_2) = 2J_R(\mathbf{p}_1 || \mathbf{p}_2) + 2J_R(\mathbf{p}_2 || \mathbf{p}_1).$$

4) *The Patrick-Fisher distance:* Given the derivations in the above subsections, computing the Patrick-Fisher distance  $J_P(\mathbf{p}_1, \mathbf{p}_2)$  can be easily done by combining related terms.

$$\begin{aligned} J_P(\mathbf{p}_1, \mathbf{p}_2) &= [2(2\pi)^f \rho^{f-r_1} \prod_{i=1}^{r_1} \lambda_{i,1}]^{-1/2} + [2(2\pi)^f \rho^{f-r_2} \prod_{i=1}^{r_2} \lambda_{i,2}]^{-1/2} \\ &- 2[2(2\pi)^f \rho^{f-r_1-r_2} \prod_{i=1}^{r_1+r_2} (\rho + \tau_i)]^{-1/2} \exp\{-\rho^{-1}(\xi_{11} + \xi_{22} - 2\xi_{12})\}. \end{aligned}$$

where  $\{\tau_i; i = 1, 2, \dots, r_1 + r_2\}$  are eigenvalues of the  $\mathbf{L}_{ch}$  matrix defined in (7) with  $\alpha_1 = \alpha_2 = 1/2$ .

### C. Characteristics of probabilistic distances in RKHS

1) *Limiting behaviors:* It is interesting to study the behavior of the distances when  $\rho$  approaches to zero. When  $\rho = 0$ , the RKHS reduces to two different kernel principal subspaces, one for each class. The derived limiting distances measure the ‘growth’ rate of the distances (before limiting) between two Gaussian densities with full-rank covariance matrices defined in the RKHS when the full-rank covariance matrix of the Gaussian density degenerates to a lower-rank. However, the limiting distances still calibrate the pattern separability and carry many optimal properties their original counterparts possess, additionally equipped with nonlinear embedding. In addition, they free us from specifying the  $\rho$  parameter.

As shown in Appendix-II, we have

$$\lim_{\rho \rightarrow 0} \rho J_C(\mathbf{p}_1, \mathbf{p}_2) = \hat{J}_C(\mathbf{p}_1, \mathbf{p}_2), \quad \lim_{\rho \rightarrow 0} \rho J_R(\mathbf{p}_1 || \mathbf{p}_2) = \hat{J}_R(\mathbf{p}_1 || \mathbf{p}_2), \quad \lim_{\rho \rightarrow 0} \rho J_D(\mathbf{p}_1, \mathbf{p}_2) = \hat{J}_D(\mathbf{p}_1, \mathbf{p}_2),$$

where

$$\begin{aligned} 2\hat{J}_C(\mathbf{p}_1, \mathbf{p}_2) &= \alpha_1 \alpha_2 \{ \hat{\xi}_{11} + \hat{\xi}_{22} - 2\hat{\xi}_{12} \}, \\ 2\hat{J}_R(\mathbf{p}_1 || \mathbf{p}_2) &= \hat{\theta}_{121} + \hat{\theta}_{222} - \hat{\theta}_{122} - \hat{\theta}_{221} + \text{tr}[\Lambda_{r_1,1}] - \hat{\eta}_{12}, \\ 2\hat{J}_D(\mathbf{p}_1, \mathbf{p}_2) &= 2\hat{J}_R(\mathbf{p}_1 || \mathbf{p}_2) + 2\hat{J}_R(\mathbf{p}_2 || \mathbf{p}_1). \end{aligned}$$

When  $\alpha_1 = \alpha_2 = 1/2$ , we obtain the limiting distance for the Bhattacharyya distance

$$2\hat{J}_B(\mathbf{p}_1, \mathbf{p}_2) = \frac{1}{4} \{ \hat{\xi}_{11} + \hat{\xi}_{22} - 2\hat{\xi}_{12} \}.$$

When  $\alpha_1 = \frac{N_1}{N}$  and  $\alpha_2 = \frac{N_2}{N}$ , we obtain the limiting distance for the Mahalonobis distance

$$\hat{J}_M(\mathbf{p}_1, \mathbf{p}_2) = \hat{\xi}_{11} + \hat{\xi}_{22} - 2\hat{\xi}_{12}.$$

Especially if  $N_1 = N_2$ , the limiting Bhattacharyya and Mahalonobis distances are identical up to a fixed constant. The limiting behavior of the Patrick-Fisher distance  $J_P(\mathbf{p}_1, \mathbf{p}_2)$  is not interesting since it involves  $f$ , thus we omit its discussion.

It should be noted that the limiting distances are significantly different from the distances directly computed from the  $r$ -dimensional kernel principal subspace that disregards the remaining dimensions. The above statement implies the assumption that  $r_1 = r_2 = r$ . First of all, such an assumption is not necessary for computing the limiting distances. Even with this assumption, as mentioned earlier, the limiting distances measure the ‘‘growth’’ speed of the corresponding distances that are defined on the full space when the full space is reduced to the  $r$ -dimensional kernel principal subspace, i.e.  $\rho$  approaches zero. The only common thing about the limiting distances and the distances directly from the  $r$ -dimensional kernel principal subspace is that they are both related to the eigenvalues and eigenvectors of the  $r$ -dimensional kernel principal subspace.

The proposed probabilistic distance measures can be extended in many ways. Here we emphasize two important extensions. The first extension is to convert probabilistic distances into kernel functions for ensemble. The second extension is to generalize the observational vector data to arbitrary data representation.

2) *Kernel for ensemble*: A kernel for ensemble is a two-input kernel function that takes the two ensembles as inputs and satisfies the requirement of positive definiteness. Several kernels for ensemble have emerged in the literature. We review some related kernels.

Wolf and Shashua [31] proposed kernel principal angle. The principal angle is defined as the angle between the principal subspaces of the two matrices and then “kernelized”. However, this is only for the ensemble that is in a matrix form.

Jebara and Kondor [32] showed that the Bhattacharyya coefficient [4] that operates the probability distribution defined in the original data space is a reproducing kernel.

$$k(\mathbb{P}_1, \mathbb{P}_2) = \int_{\mathbf{x}} \mathbb{P}_1(\mathbf{x})^{1/2} \mathbb{P}_2(\mathbf{x})^{1/2} d\mathbf{x}. \quad (13)$$

In [11], they extended the Bhattacharyya kernel to operate the probability distribution defined in the RKHS. However, there are several differences between our approach and that in [11]. Firstly, they only computed the Bhattacharyya coefficient that differs from the Bhattacharyya distance by  $-\log(\cdot)$ . Secondly, we also compute other distances such as the Chernoff distance, the KL divergence and its symmetric version, etc. For example, we find in the experiment that the KL divergence can be used in a retrieval problem to replace the need of building a discriminant model. Finally, different regularizations are used to approximate the covariance matrix in the feature space. Our approximation allows us to study the limiting behavior.

In [33], [34], Vasconcelos *et al.* proposed a kernel function based on the Kullback-Leibler divergence distance in the original data space. This is done in the following fashion<sup>1</sup>:

$$k_J = \exp\{-aJ + b\}; \quad a, b > 0. \quad (14)$$

In this paper, we also adopt the same strategy to convert a probabilistic distance  $J$  to a kernel function. However, this is the only commonality between [33] and our paper. They bear many differences, highlighting the contributions of our paper. First, the probabilistic distance  $J$  in our work can be in various forms such as Chernoff distance, Bhattacharyya distance, etc., while in [33] only the KL divergence is used. Second, we focus on computing the probabilistic distance based on the sample similarity function, while in [33] the KL divergence is computed in the original data domain. This means that data representation other than the vector can not be handled in [33]. Thirdly, the concept of ensemble similarity is never introduced in [33]. This

<sup>1</sup>It seems that there is no proof that  $k_J$  is a kernel function. However, this is still a useful quantity for SVM.

is the founding concept of the paper. We emphasized this from the beginning of the paper, derived its computations in RKHS, addressed its extensions, and confirmed its effectiveness using experiments. Finally, even the KL divergence is computed very differently. The paper [33] investigated the KL divergence only evaluated in the original data space and also addressed how to compute the KL divergence for different families of densities. In our work, we investigated the KL divergence between two Gaussian densities in the RKHS.

3) *Probabilistic distances for different data representations:* So far, we focused on the vector data type and derived probabilistic distances in the RKHS that is mapped from a vector space. However, because our derivation only relies on the knowledge of the reproducing kernel function, we are able to compute the probabilistic distances between ensembles of data points in various representations as long as we have kernel function based on these representations. Examples of such representation include strings [14], graphs [15], lattices [16], statistical manifolds [17], [18], [19], and so on [20], [21]. For instance, a graph ensemble is a collection of graphs. Since we are able to compute the probabilistic distances between two graph ensembles, we implicitly define the probabilistic distribution for the graph population.

The computation of the probabilistic distances can be seen from the computational details presented in Section III since all the computations are derived from the Gram matrix that needs kernel function only. From a theoretical perspective, this can be justified by the equivalence between the kernel function and the distance metric (i.e., equation (2)): the inner product defines the geometry of the space containing the data points with specified representations.

#### IV. EXPERIMENTAL RESULTS

In our experiments, we used only the limiting distances, namely the limiting Chernoff distance  $\hat{J}_C(\mathbf{p}_1, \mathbf{p}_2)$  (or the limiting Bhattacharyya distance  $\hat{J}_B(\mathbf{p}_1, \mathbf{p}_2)$ ), the limiting KL divergence  $\hat{J}_R(\mathbf{p}_1 || \mathbf{p}_2)$ , and the limiting symmetric KL divergence  $\hat{J}_D(\mathbf{p}_1, \mathbf{p}_2)$ , since they do not depend on the choice  $\rho$ , which frees us from the burden of choosing  $\rho$ . Since  $N_1 = N_2$  in the experiments, the limiting Mahalanobis distance is identical to the limiting Bhattacharyya distance. Also, we always set  $r_1 = r_2 = r$  for simplicity, even though the general case of  $r_1 \neq r_2$  is legitimate.

We performed the following three experiments. The first experiment tested on synthetically generated ensembles that share the same mean and covariance matrix and demonstrated that the probabilistic distances in the RKHS is able to capture high-order statistical information for

nonlinear data structure. The second experiment on recognizing digits evaluated the viewpoint of treating the probabilistic distances as the kernel functions for ensemble using the SVM framework. The third experiment computed the probabilistic distances on a data representation other than vector using face recognition from video. Here we represented each face image in the video frame as a matrix (rather than vector) and employed the kernel between matrix as the sample similarity function.

### A. Synthetic examples

To fail the probabilistic distances between two Gaussian densities in the original space, we designed four different 2-D densities sharing the same mean (zero mean) and covariance matrix (identity matrix). As shown in Fig. 2, the four densities are 2-D Gaussian, and ‘O’-, ‘D’-, and ‘X’-shaped uniform densities, where say the ‘O’-shaped uniform density means that it is uniform in the ‘O’-shaped region and zero outside the region. Fig. 2 actually shows their 300 i.i.d. realizations sampled from these four densities. Due to the same first- and second-order statistics, the probabilistic distance between any of two densities in the original space is simply zero. This highlights the virtue of a nonlinear mapping that provides us information embedded in higher-order statistics.

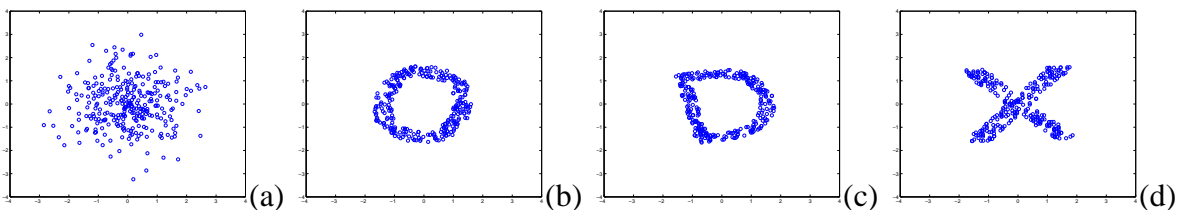


Fig. 2. 300 i.i.d. realizations of four different densities with the same mean (zero mean) and covariance matrix (identity matrix). (a) 2-D Gaussian. (b) ‘O’-shaped uniform. (c) ‘D’-shaped uniform. (d) ‘X’-shaped uniform.

Obviously, the probabilistic distances depend on the number of eigenpairs  $r$  and the RBF kernel width  $\sigma$ . Fig. 3 displays  $\hat{J}_D$  and  $\hat{J}_B$  as a function of  $r$  and  $\sigma$ . (i) The effect of  $\sigma$  is biased: It always disfavors a large  $\sigma$  since a large  $\sigma$  tends to pool the data together. For example, when  $\sigma$  is infinite, all data points collapse to one single point in the RKHS and become inseparable. (ii) Generally speaking, it is not necessary that a large  $r$  (or equivalently using a nonlinear subspace

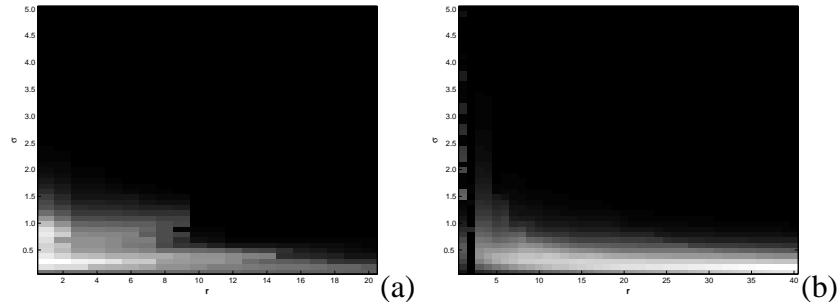


Fig. 3. (a) The Bhattacharyya distance  $\hat{J}_B(\sigma, r)$  and (a) the divergence distance  $\hat{J}_D(\sigma, r)$  between the 2-D Gaussian and the ‘O’-shaped uniform as a function of  $\sigma$  and  $r$ .

$\hat{J}_R(p_1    p_2)$	Gau	‘O’	‘D’	‘X’
Gau	-	.0740	.0782	.0808
‘O’	.0584	-	.0281	.0523
‘D’	.0670	.0295	-	.0436
‘X’	.0944	.0505	.0417	-

$\hat{J}_B(p_1, p_2)$	Gau	‘O’	‘D’	‘X’
Gau	-	.0033	.0037	.0048
‘O’	.0033	-	.0021	.0099
‘D’	.0037	.0021	-	.0086
‘X’	.0048	.0099	.0086	-

TABLE III

(a) The symmetric KL divergence in the RKHS with  $\sigma = 1$  and  $r = 3$ . (b) The Bhattacharyya distance in the RKHS with  $\sigma = 0.5$  and  $r = 1$ .  $p_1$  is listed in the first column and  $p_2$  in the first row.

with a large dimension) yields a large distance. A typical subspace yielding the maximum distances is of low-dimensional.

Table III lists some computed values of the probabilistic distances. It is interesting to observe that when the shapes of two densities are close, their distance is small. For example, ‘O’ is closest to ‘D’ among all possible pairs. The closest density to the 2-D Gaussian is the ‘O’-shaped uniform. It seems that the proximity of shape determines the closeness of probabilistic distances. We further evaluate this using the digit recognition experiment reported below.

### B. Digit recognition

We used the USPS digit database [13] in the experiments. It is a 10-class problem. Rather than using the binary images of digits as inputs, we used a sample representation (using 50 data points) for each image. Furthermore, to make the problem more difficult, we normalized

these 50 data points so that they have zero mean and unit variances along horizontal and vertical axes. Such a normalization attempts to remove the difference in lower-order (1st and 2nd order) statistical information and leaves only higher-order statistical information to be characterized. Figure 4 shows some original binary images and their normalized sample representations. Note the stretching effect due to normalization. In addition, such normalization makes inapplicable the regular kernel methods that are typically used in digit recognition. These kernel methods take vectors input coming from “vectorizing” the original images. However, the normalization step ruins the integer pixel grid by producing float coordinate values that have different ranges for different digits.

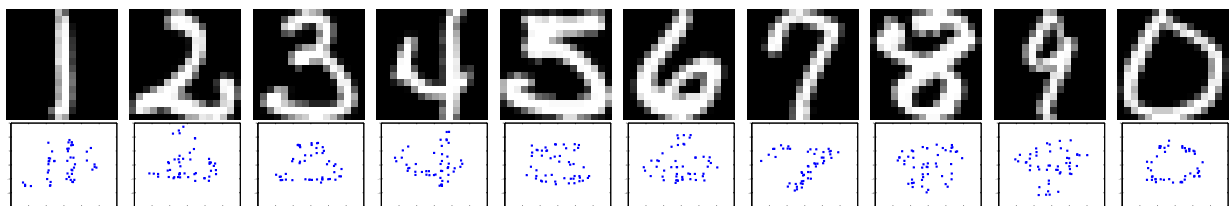


Fig. 4. Original binary images of digits and their normalized sample representations.

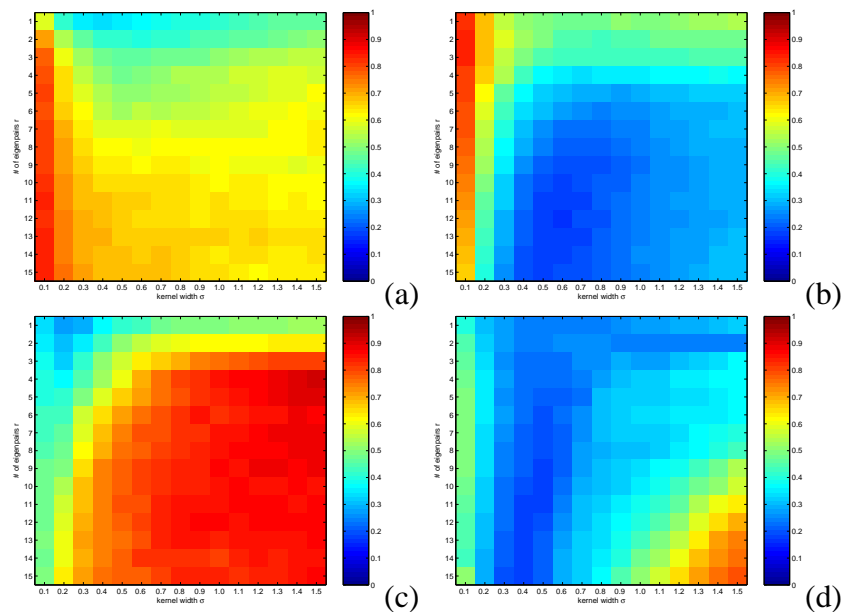


Fig. 5. Recognition error rates as a function of  $r$  and  $\sigma$  obtained using the 1-NN rule based on (a)  $\hat{J}_B(p_1, p_2)$  and (b)  $\hat{J}_D(p_1, p_2)$  and the SVM based on (c)  $\hat{J}_B(p_1, p_2)$  and (d)  $\hat{J}_D(p_1, p_2)$ .

For each digit, we randomly selected 60 images to generate training data points and another 40 images to generate testing data points. For each testing data point, we computed its probabilistic distances to every training data point and determined its class label using the one nearest neighbor (1-NN) classifier. We repeated such random selection ten times and took the average classification error rate for reporting.

We used the RBF kernel function  $k(\mathbf{x}, \mathbf{y}) = \exp\{|\mathbf{x} - \mathbf{y}|^2 / (2\sigma^2)\}$  in the experiments. There are two free parameters: the kernel width  $\sigma$  and the number of eigenpairs  $r$ . Figs. 5(a) and (b) show the 1-NN classification error rates with different choices of  $\sigma$  and  $r$ . When  $r$  is very small, a smaller classification error is obtained by using the Bhattacharyya distance instead of the divergence distance. As  $r$  becomes large, the error rate corresponding to the Bhattacharyya distance actually increases, while that corresponding to the divergence distance consistently becomes smaller. In general, the divergence distance is more discriminative than the Bhattacharyya distance in the sense that using the divergence distance (using appropriate parameters) can generate far smaller classification error. As  $\sigma$  varies, the classification error varies too. When  $\sigma$  is around 0.4 – 0.6, the best performances are achieved by using 1-NN classifier with the divergence distance.

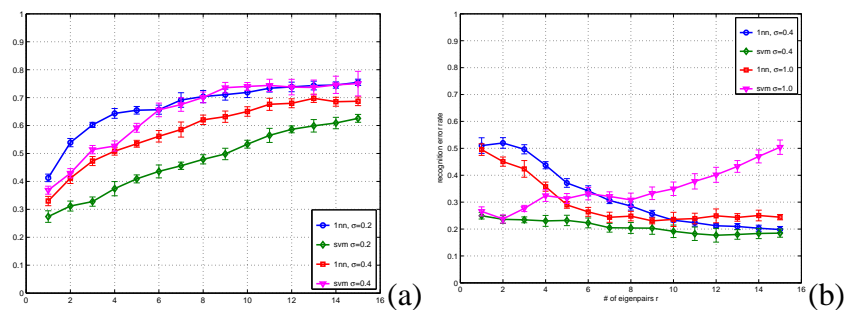


Fig. 6. The mean and covariance curves of recognition error rate as a function of  $r$  with different  $\sigma$ 's using (a) the Bhattacharyya distance  $\hat{J}_B(\mathcal{P}_1, \mathcal{P}_2)$  and (b) the symmetric KL divergence  $\hat{J}_D(\mathcal{P}_1, \mathcal{P}_2)$ .

We further tested the kernel for ensemble. We used  $a = 1$  and  $b = 0$  in (14). We plugged in  $k_J$  in the support vector machine (SVM) for classification. We followed a one-versus-all strategy and trained separate SVM for each class. In testing, the class label goes to the SVM with the highest score. Figs. 5(c) and (d) show that the SVM classification error rates with different choices of  $\sigma$  and  $r$ . Fig. 6 highlights the comparison of the recognition performances obtained

by 1-NN and SVM classifiers.

Using the SVM classifier, the recognition performance can be significantly improved. For instance, when the kernel derived from the Bhattacharyya distance with  $\sigma = 0.2$  is used, the performance improvement is consistently about 15%-30% regardless of the value of  $r$ . When the kernel derived from the symmetric KL divergence with  $\sigma = 0.4$  is used, the performance improvement can be as large as about 30% for small  $r$ 's. However, using the SVM classifier does not necessarily guarantee improvement of the recognition performance. For instance, when the kernel derived from the Bhattacharyya distance with  $\sigma = 0.4$  is used, the performance is degraded consistently about 5% regardless of the value of  $r$ . When the kernel derived from the symmetric KL divergence with  $\sigma = 1.0$  is used, the performance improves for small  $r$ 's but degrades for large  $r$ 's. Therefore, in practice, cross-validation should be invoked to arrive at best performance. In addition, the standard deviation of the recognition error rate is rather consistent.

Incidentally, we performed digit recognition using regular kernel method based on vectors raster-scanned from sampled representation before normalization, which makes the recognition problem simpler. The best performance using the RBF kernel for vector input is around 50%. The proposed probabilistic distance (with proper choice) can outperform it by a large margin, even after normalization.

The final observation is that (i) using the kernel derived from the symmetric KL divergence produces smaller recognition error rate than using the Bhattacharyya distance and (ii) the best performance is obtained (the last point of the green curve in Fig. 6(b)) using the SVM classifier. However, directly using the divergence distance in the 1-NN classifier yields the performance very close the best one. This means that utilizing the SVM does not gain additional discrimination and further proves the discriminative power possessed by the KL distance. Therefore, in cases when training the SVM is inconvenient, we can directly utilize the KL divergence.

### *C. Face recognition from video*

The gallery set consists of 15 sets (one per person) while the probe set consists of 30 new sets of the same people (1-4 videos per person). In these sets, the people move their heads freely so that pose and illumination variations abound. The existence of these variations violates the normal assumption of the original data space used in [35]. Fig. 7 shows some example faces of the 4<sup>th</sup> gallery person, the 9<sup>th</sup> gallery person, and the 4<sup>th</sup> probe person (whose identity is the

same as the 4<sup>th</sup> gallery person). The face images of size 16 by 16 are obtained by automatically cropping from video sequences (courtesy of [36]) using an in-house flow tracking algorithm. A zero-mean-unit-variance normalization is adopted to partially compensate illumination variation.

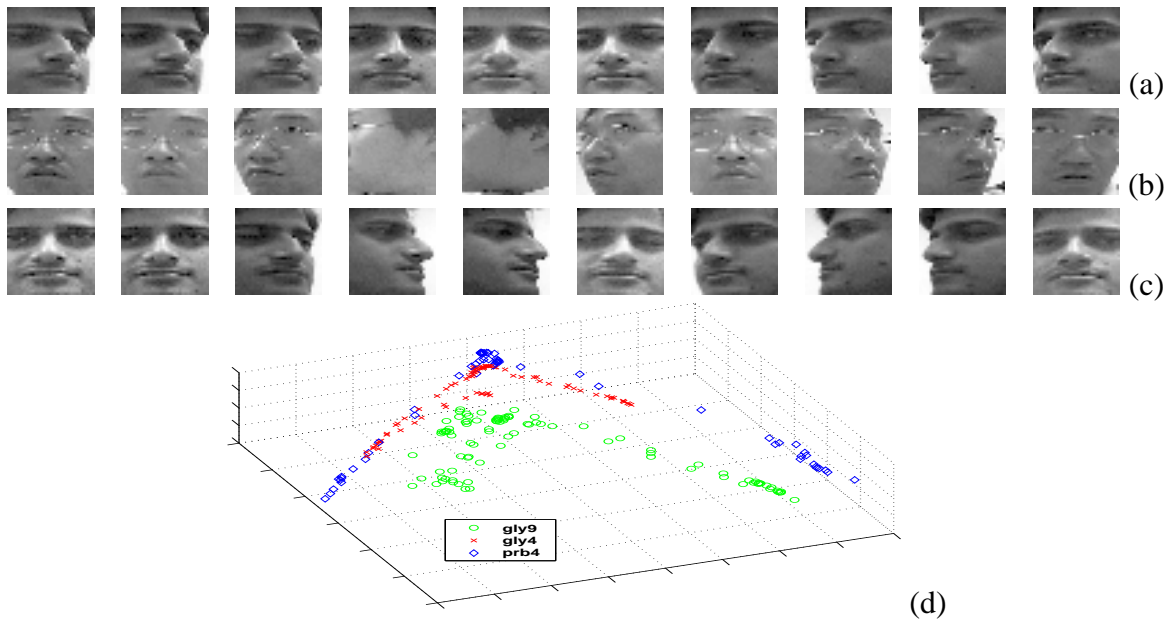


Fig. 7. Examples of face images in the gallery and probe set. (a) The 4<sup>th</sup> gallery person in 10 frames (every 8 frames) of a 80-frame sequence. (b) The 9<sup>th</sup> gallery person in 10 frames (every 10 frames) of a 105-frame sequence. (c) The 4<sup>th</sup> probe person in 10 frames (every 6 frames) of a 60-frame sequence. (d) The plot of first three PCA coefficients of the above three sets.

A generic principal component analysis is performed to visualize the data. Fig. 7 also plots the first three PCA coefficients of the 4<sup>th</sup> gallery person, the 9<sup>th</sup> gallery person, and the 4<sup>th</sup> probe person. Clearly, the manifolds are highly nonlinear, which indicates a need for nonlinear modeling. The nonlinearity mainly arises from the pose/illumination variations available in the video sequences as evidenced in Figs. 7(a), 7(b), and 7(c).

We studied three different representations of a face image: (i) a vector, (ii) a matrix, and (iii) a bag of pixels [37]. The vector representation is commonly used in the literature as in subspace analysis. The image is converted to a vector by raster scanning the pixels. The matrix representation is a natural representation of the image. The ‘bag’ representation treats an image as a collection of triples  $\{(x, y, i(x, y))\}$ , with each triple containing the pixel location and intensity.

We need the sample similarity for the three representations. For the vector representation, we used the vector RBF kernel as in (3) with  $\sigma = 16$ . For the matrix representation, it is easy to show that the following function  $k(\mathbf{X}, \mathbf{Y})$  between two  $p \times q$  matrices  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_q]$  and  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_q]$  (here  $p = q = 16$ ) is a reproducing kernel for matrix.

$$k(\mathbf{X}, \mathbf{Y}) = \exp\left\{\frac{\text{tr}[\mathbf{K}_\psi(\mathbf{X}, \mathbf{X})] - 2\text{tr}[\mathbf{K}_\psi(\mathbf{X}, \mathbf{Y})] + \text{tr}[\mathbf{K}_\psi(\mathbf{Y}, \mathbf{Y})]}{2\sigma^2}\right\}, \quad (15)$$

where  $\mathbf{K}_\psi(\mathbf{X}, \mathbf{Y})$  is the Gram matrix between  $\mathbf{X}$  and  $\mathbf{Y}$ , whose  $ij^{\text{th}}$  entry  $\psi(\mathbf{x}_i)^\top \psi(\mathbf{y}_j)$  is evaluated by another (vector) kernel function  $l(\mathbf{x}_i, \mathbf{y}_j)$ . We set the  $l$  function as the vector RBF kernel, defined in Eq. (3), with its  $\sigma = 16$ . We set  $\sigma$  in (15) to be  $\sigma = 1$ . We call this as the RBF matrix kernel since it has a similar form to the RBF kernel for vector. For the ‘bag’ representation, we use the Bhattacharyya kernel as defined in 13. Note here both the sample similarity function and the ensemble similarity can be Bhattacharyya kernels.

For purpose of comparison, we implemented two ad hoc ensemble similarity functions. The first one is the mean value of the cross dot product matrix, i.e., the part  $\mathbf{K}_{12} = \Phi_1^\top \Phi_2$  in (6). The second one is the median value of the cross dot product matrix.

Table IV reports the recognition rates. The top match with the smallest distance is claimed to be the winner. For a comparison, we also implemented the divergence distance and the Bhattacharyya distance in the original vector space [35] (the last row of Table IV). From Table IV, we observe the following:

- Using the proposed ensemble similarity always outperform the ad hoc functions;
- Using ensemble similarity in RKHS induced by the vector RBF kernel is better than that in the original vector space;
- The ‘bag’ representation has advantage over the matrix and vector representations.
- Comparing the divergence distance and the Bhattacharyya distance, the divergence distance is better.

The best performance is achieved using and the KL divergence in the RKHS induced by the Bhattacharyya kernel defined on the ‘bag’ representation. Out of 30 probe sets, we successfully classified 29 of them. In fact, Fig. 7 shows a misclassification example in [35], where the 4<sup>th</sup> probe person is misclassified as the 9<sup>th</sup> gallery person, while one of our approaches (the KL divergence in the RKHS induced by the matrix RBF kernel) corrected this error.

Ensemble similarity	Divergence distance $\hat{J}_R$ in RKHS	Bhattacharyya distance $\hat{J}_B$ in RKHS	mean of sample similarity	median of sample similarity
Sample similarity				
Bhattacharyya kernel	28/30	26/30	20/30	23/30
RBF matrix kernel	27/30	25/30	19/30	23/30
RBF vector kernel	26/30	25/30	17/30	22/30
Vector space $\mathcal{R}^{d=16 \times 16}$	24/30	24/30	NA	NA

TABLE IV

*The recognition scores obtained by using the probabilistic distance measures in different spaces.*

## V. CONCLUSIONS AND DISCUSSIONS

In this paper, we studied pattern separability in the RKHS. This separability was measured by the probabilistic distance measures in the RKHS. The probabilistic distance measure can be universally regarded as ensemble similarity functions based on the sample similarity function that is the reproducing kernel corresponding to the RKHS. Since the RKHS might be infinite-dimensional, we derived “limiting” distances which can be easily computed. These distances retain their original properties while taking into account the data nonlinearity. We conducted a series of experiments using synthetic and real data sets to demonstrate the properties and efficiency of the proposed distances.

## REFERENCES

- [1] P. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*. Prentice Hall International, 1982.
- [2] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2001.
- [3] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 1991.
- [4] T. Kailath, “The divergence and Bhattacharyya distance measures in signal selection,” *IEEE Trans. on Communication Technology*, vol. COM-15, no. 1, pp. 52–60, 1967.
- [5] J. Mercer, “Functions of positive and negative type and their connection with the theory of integral equations,” *Philos. Trans. Roy. Soc. London*, vol. A 209, pp. 415–446, 1909.
- [6] N. Aronszajn, “Theory of reproducing kernels,” *Transactions of the American Mathematics Society*, vol. 68, no. 3, pp. 337–404, 1950.
- [7] B. Schölkopf, A. Smola, and K.-R. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.
- [8] G. Baudat and F. Anouar, “Generalized discriminant analysis using a kernel approach,” *Neural Computation*, vol. 12, no. 10, pp. 2385–2404, 2000.

- [9] F. Bach and M. I. Jordan, “Kernel independent component analysis,” *Journal of Machine Learning Research*, vol. 3, pp. 1–48, 2002.
- [10] —, “Learning graphical models with Mercer kernels,” *Neural Information Processing Systems*, 2002.
- [11] R. Kondon and T. Jebara, “A kernel between sets of vectors,” *International Conference on Machine Learning (ICML)*, 2003.
- [12] Z. Zhang, D. Yeung, and J. Kwok, “Wishart processes: a statistical view of reproducing kernels,” *Technical Report KHUST-CS401-01*, 2004.
- [13] V. N. Vapnik, *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, ISBN 0-387-94559-8, 1995.
- [14] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, “Text classification using string kernels,” *Journal of Machine Learning Research*, vol. 2, pp. 419–444, 2002.
- [15] R. Kondor and J. Lafferty, “Diffusion kernels on graphs and other discrete input spaces,” *ICML*, 2002.
- [16] C. Cortes, P. Haffner, and M. Mohri, “Lattice kernels for spoken-dialog classification,” *ICASSP*, 2003.
- [17] T. Jaakkola and D. Haussler, “Exploiting generative models in discriminative classifiers,” *NIPS*, vol. 11, 1999.
- [18] K. Tsuda, M. Kawanabe, G. Rätsch, S. Sonnenburg, and K. Müller, “A new discriminative kernel from probabilistic models,” *NIPS*, vol. 14, 2002.
- [19] M. Seeger, “Covariances kernel from Bayesian generative models,” *NIPS*, vol. 14, pp. 905–912, 2002.
- [20] M. Collins and N. Duffy, “Convolution kernels for natural language,” *NIPS*, vol. 14, pp. 625–632, 2002.
- [21] L. Wolf and A. Shashua, “Learning over sets using kernel principal angles,” *Journal of Machine Learning Research*, vol. 4, pp. 895–911, 2003.
- [22] H. Chernoff, “A measure of asymptotic efficiency of tests for a hypothesis based on a sum of observations,” *Annals of Mathematical Statistics*, vol. 23, pp. 493–507, 1952.
- [23] A. Bhattacharyya, “On a measure of divergence between two statistical populations defined by their probability distributions,” *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.
- [24] K. Matusita, “Decision rules based on the distance for problems of fit, two samples and estimation,” *Ann. Math. Stat.*, vol. 26, pp. 631–640, 1955.
- [25] E. Patrick and F. Fisher, “Nonparametric feature selection,” *IEEE Trans. Information Theory*, vol. 15, pp. 577–584, 1969.
- [26] T. Lissack and K. Fu, “Error estimation in pattern recognition via L-distance between posterior density functions,” *IEEE Trans. Information Theory*, vol. 22, pp. 34–45, 1976.
- [27] B. Adhikara and D. Joshi, “Distance discrimination et resume exhaustif,” *Publs. Inst. Statis.*, vol. 5, pp. 57–74, 1956.
- [28] P. Mahalanobis, “On the generalized distance in statistics,” *Proc. National Inst. Sci. (India)*, vol. 12, pp. 49–55, 1936.
- [29] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, New York, 2001.
- [30] M. Tipping, “Sparse kernel principal component analysis,” *Neural Information Processing Systems*, 2001.
- [31] L. Wolf and A. Shashua, “Kernel principal angles for classification machines with applications to image sequence interpretation,” *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003.
- [32] T. Jebara and R. Kondon, “Bhattacharyya and expected likelihood kernels,” *Conference on Learning Theory (COLT)*, 2003.
- [33] N. Vasconcelos, P. Ho, and P. Moreno, “The Kullback-Leibler kernel as a framework for discriminant and localized representations for visual recognition,” *European Conference on Computer Vision*, 2004.
- [34] P. Moreno, P. Ho, and N. Vasconcelos, “A Kullback-Leibler divergence based kernel for svm classification in multimedia applications,” *Neural Information Processing Systems*, 2003.

- [35] G. Shakhnarovich, J. Fisher, and T. Darrell, "Face recognition from long-term observations," *European Conference on Computer Vision*, 2002.
- [36] K. Lee, M. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003.
- [37] T. Jebara, "Images as bags of pixels," *Proc. of IEEE International Conference on Computer Vision*, 2003.
- [38] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, pp. 72–86, 1991.
- [39] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis*. Academic Press, 1979.
- [40] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society, Series B*, vol. 61, no. 3, pp. 611–622, 1999.

#### APPENDIX-I: COMPUTATIONS RELATED TO COVARIANCE MATRIX APPROXIMATION

Table V lists important quantities related to the covariance matrix approximation in Section III. Their computations are then detailed next.

##### *Computation related to the Q matrix*

As mentioned earlier, the key quantity is the Gram matrix  $\mathbf{K} = \Phi^T \Phi$ , whose every element can be evaluated using the 'kernel trick'. Furthermore, we define the centered Gram matrix  $\bar{\mathbf{K}}$  as

$$\bar{\mathbf{K}} \equiv \mathbf{J}^T \Phi^T \Phi \mathbf{J} = \mathbf{J}^T \mathbf{K} \mathbf{J},$$

where  $\mathbf{J}$  is the centering matrix defined in Section III-A (also in Table V).

The top  $r$  eigenpairs for the covariance matrix  $\Sigma$  can be easily derived from  $\bar{\mathbf{K}}$  using the standard trick in [38]. Suppose that the top  $r$  eigenpairs for  $\bar{\mathbf{K}}$  are  $\{(\lambda_n, \mathbf{v}_n)\}_{n=1}^r$ , where  $\lambda_n$ 's are sorted in a non-increasing order, and the  $r$  top eigenpairs for  $\Sigma$  are  $\{(\lambda_n, \mathbf{u}_n)\}_{n=1}^r$ . We compute  $\mathbf{u}_n$  as

$$\mathbf{u}_n = (\lambda_n)^{-1/2} \Phi \mathbf{J} \mathbf{v}_n.$$

In a matrix form (if only the top  $r$  eigenvectors are retained),

$$\mathbf{U}_r \equiv [\mathbf{u}_1, \dots, \mathbf{u}_r] = \Phi \mathbf{J} \mathbf{V}_r \Lambda_r^{-1/2}, \quad (16)$$

where  $\mathbf{V}_r \equiv [\mathbf{v}_1, \dots, \mathbf{v}_r]$  and  $\Lambda_r \equiv \text{Diag}[\lambda_1, \dots, \lambda_r]$ , a diagonal matrix whose diagonal elements are  $\{\lambda_1, \dots, \lambda_r\}$ .

RKHS:	$\mathcal{H} = \mathcal{R}^f.$
Original observations:	$\mathbf{X}_{d \times N} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$
Nonlinear mapping:	$\phi(x) : \mathcal{R}^d \rightarrow \mathcal{R}^f$
Observations in RKHS:	$\Phi_{f \times N} = [\phi_1, \phi_2, \dots, \phi_N].$
Weight vector:	$\mathbf{s}_{N \times 1} = N^{-1} \mathbf{1}.$
Mean:	$\mu_{f \times 1} = \Phi \mathbf{s}$
Centering matrix:	$\mathbf{J}_{N \times N} = N^{-1/2} (\mathbf{I}_N - \mathbf{s} \mathbf{1}^T).$
Covariance matrix (c.m.):	$\Sigma_{f \times f} = \Phi \mathbf{J} \mathbf{J}^T \Phi^T.$
Gram matrix:	$\mathbf{K}_{N \times N} = \Phi^T \Phi.$
Centered Gram matrix:	$\bar{\mathbf{K}}_{N \times N} = \mathbf{J}^T \mathbf{K} \mathbf{J}.$
Eigenvalues of $\bar{\mathbf{K}}$ :	$\Lambda_r = \mathcal{D}[\lambda_1, \dots, \lambda_r]_{r \times r}.$
Eigenvectors of $\bar{\mathbf{K}}$ :	$\mathbf{V}_r = [\mathbf{v}_1, \dots, \mathbf{v}_r]_{N \times r}.$
Approximate centered Gram matrix:	$\mathbf{C}_{f \times f} = \Phi \mathbf{A} \Phi^T + \rho \mathbf{I}_f.$
A matrix:	$\mathbf{A}_{N \times N} = \mathbf{J} \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1}) \mathbf{V}_r^T \mathbf{J}^T.$
Inverse of $\mathbf{C}$ :	$\mathbf{C}_{N \times N}^{-1} = \rho^{-1} (\mathbf{I}_f - \Phi \mathbf{B} \Phi^T).$
B matrix:	$\mathbf{B}_{N \times N} = \mathbf{J} \mathbf{V}_r (\Lambda_r^{-1} - \rho \Lambda_r^{-2}) \mathbf{V}_r^T \mathbf{J}^T.$
Q matrix:	$\mathbf{Q}_{N \times r} = \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2}$
M matrix:	$\mathbf{M}_{r \times r} = \rho \mathbf{I}_r + \mathbf{Q}^T \bar{\mathbf{K}} \mathbf{Q}.$
L matrix:	$\mathbf{L}_{r \times r} = \mathbf{Q}^T \bar{\mathbf{K}} \mathbf{Q}.$

TABLE V

*A list of important quantities used in the paper.*

It leaves to show how to find the  $\mathbf{Q}$  matrix. To do this, we notice that the data in the feature space follow a factor analysis model [39] which relates an  $f$ -dimensional data  $\phi(\mathbf{x})$  to a latent  $r$ -dimensional variable  $\mathbf{z}$  as

$$\phi(\mathbf{x}) = \mu + \mathbf{W} \mathbf{z} + \epsilon,$$

where  $\mathbf{z} \sim \mathbf{N}(0, \mathbf{I}_r)$ ,  $\epsilon \sim \mathbf{N}(0, \rho \mathbf{I}_f)$ , and  $\mathbf{W}$  is a  $f \times r$  loading matrix. Therefore,  $\phi(\mathbf{x}) \sim \mathbf{N}(\mu, \mathbf{C})$ , where  $\mathbf{C} = \mathbf{W} \mathbf{W}^T + \rho \mathbf{I}_f$ . Note that this  $\mathbf{C}$  is exactly in the same form as in (5).

As shown in [40], the MLE's for  $\mu$  and  $\mathbf{W}$  are given by

$$\hat{\mu} = \frac{1}{N} \sum_{n=1}^N \phi(\mathbf{x}_n) = \Phi \mathbf{s}, \quad \hat{\mathbf{W}} = \mathbf{U}_r (\Lambda_r - \rho \mathbf{I}_r)^{1/2} \mathbf{R}, \quad (17)$$

where  $\mathbf{R}$  is any  $r \times r$  orthogonal matrix, and  $\mathbf{U}_r$  and  $\Lambda_r$  contain the top  $r$  eigenvectors and eigenvalues of the  $\Sigma$  matrix. Without loss of generality, we assume that  $\mathbf{R} = \mathbf{I}_r$  from now on.

Substituting (16) into (17), we obtain the following:

$$\hat{\mathbf{W}} = \Phi \mathbf{J} \mathbf{V}_r \Lambda_r^{-1/2} (\Lambda_r - \rho \mathbf{I}_r)^{1/2} = \Phi \mathbf{J} \mathbf{Q},$$

where

$$\mathbf{Q}_{N \times r} \equiv \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2}. \quad (18)$$

Since the matrix  $(\mathbf{I}_r - \rho \Lambda_r^{-1})$  in (18) is diagonal, additional saving in computing its square root is achieved.

#### *Computation related to the M matrix*

We compute first  $\mathbf{L} = \mathbf{Q}^T \bar{\mathbf{K}} \mathbf{Q}$  and then  $\mathbf{M}$ .

$$\begin{aligned} \mathbf{L} = \mathbf{Q}^T \bar{\mathbf{K}} \mathbf{Q} &= (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \mathbf{V}_r^T \bar{\mathbf{K}} \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \\ &= (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \Lambda_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} = \Lambda_r - \rho \mathbf{I}_r, \end{aligned}$$

where the fact that  $\mathbf{V}_r^T \bar{\mathbf{K}} \mathbf{V}_r = \mathbf{V}_r^T \mathbf{J}^T \mathbf{K} \mathbf{J} \mathbf{V}_r = \Lambda_r$  is used. Therefore,

$$\mathbf{M} = \rho \mathbf{I}_r + \mathbf{Q}^T \bar{\mathbf{K}} \mathbf{Q} = \rho \mathbf{I}_r + (\Lambda_r - \rho \mathbf{I}_r) = \Lambda_r, \quad |\mathbf{M}| = |\Lambda_r| = \prod_{i=1}^q \lambda_i, \quad \mathbf{M}^{-1} = \Lambda_r^{-1}.$$

#### *Computation related to the approximate covariance matrix C*

$$|\mathbf{C}| = \rho^{f-r} |\mathbf{M}| = \rho^{f-r} |\Lambda_r| = \rho^{f-r} \prod_{i=1}^r \lambda_i.$$

$$\mathbf{C}^{-1} = (\rho \mathbf{I}_f + \mathbf{W} \mathbf{W}^T)^{-1} = \rho^{-1} (\mathbf{I}_f - \mathbf{W} \mathbf{M}^{-1} \mathbf{W}^T) = \rho^{-1} (\mathbf{I}_f - \Phi \mathbf{B} \Phi^T),$$

#### *Computation related to the A and B matrices*

$$\begin{aligned} \mathbf{A} &= \mathbf{J} \mathbf{Q} \mathbf{Q}^T \mathbf{J}^T = \mathbf{J} \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \mathbf{V}_r^T \mathbf{J}^T \\ &= \mathbf{J} \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1}) \mathbf{V}_r^T \mathbf{J}^T \end{aligned}$$

$$\begin{aligned} \mathbf{B} &= \mathbf{J} \mathbf{Q} \mathbf{M}^{-1} \mathbf{Q}^T \mathbf{J}^T = \mathbf{J} \mathbf{V}_r (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \Lambda_r^{-1} (\mathbf{I}_r - \rho \Lambda_r^{-1})^{1/2} \mathbf{V}_r^T \mathbf{J}^T \\ &= \mathbf{J} \mathbf{V}_r (\Lambda_r^{-1} - \rho \Lambda_r^{-2}) \mathbf{V}_r^T \mathbf{J}^T \end{aligned}$$

$$\begin{aligned}\text{tr}[\mathbf{AK}] &= \text{tr}[\mathbf{J}\mathbf{V}_r(\mathbf{I}_r - \rho\Lambda_r^{-1})\mathbf{V}_r^{\mathbf{T}}\mathbf{J}^{\mathbf{T}}\mathbf{K}] = \text{tr}[(\mathbf{I}_r - \rho\Lambda_r^{-1})\mathbf{V}_r^{\mathbf{T}}\mathbf{J}^{\mathbf{T}}\mathbf{K}\mathbf{J}\mathbf{V}_r] \\ &= \text{tr}[(\mathbf{I}_r - \rho\Lambda_r^{-1})\Lambda_r] = \text{tr}[\Lambda_r] - \rho r = \sum_{i=1}^r \lambda_i - \rho r.\end{aligned}\quad (19)$$

$$\begin{aligned}\text{tr}[\mathbf{BK}] &= \text{tr}[\mathbf{J}\mathbf{V}_r(\Lambda_r^{-1} - \rho\Lambda_r^{-2})\mathbf{V}_r^{\mathbf{T}}\mathbf{J}^{\mathbf{T}}\mathbf{K}] = \text{tr}[(\Lambda_r^{-1} - \rho\Lambda_r^{-2})\mathbf{V}_r^{\mathbf{T}}\mathbf{J}^{\mathbf{T}}\mathbf{K}\mathbf{J}\mathbf{V}_r] \\ &= \text{tr}[(\Lambda_r^{-1} - \rho\Lambda_r^{-2})\Lambda_r] = r - \rho\text{tr}[\Lambda_r^{-1}] = r - \rho\sum_{i=1}^r \lambda_i^{-1}.\end{aligned}\quad (20)$$

## APPENDIX-II: COMPUTATIONS RELATED TO PROBABILISTIC DISTANCES IN RKHS

This part presents the detail of computing probabilistic distances in RKHS in Section III.

*Computations related to the Chernoff distance*

$$\begin{aligned}\mathbf{A}_{ch} &= \begin{bmatrix} \alpha_1\mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \alpha_2\mathbf{A}_2 \end{bmatrix} = \begin{bmatrix} \alpha_1\mathbf{J}_1\mathbf{Q}_1\mathbf{Q}_1^{\mathbf{T}}\mathbf{J}_1^{\mathbf{T}} & \mathbf{0} \\ \mathbf{0} & \alpha_2\mathbf{J}_2\mathbf{Q}_2\mathbf{Q}_2^{\mathbf{T}}\mathbf{J}_2^{\mathbf{T}} \end{bmatrix} = \mathbf{P}\mathbf{P}^{\mathbf{T}}. \\ \mathbf{P}_{(N_1+N_2)\times(r_1+r_2)} &\equiv \begin{bmatrix} \sqrt{\alpha_1}\mathbf{J}_1\mathbf{Q}_1 & \mathbf{0} \\ \mathbf{0} & \sqrt{\alpha_2}\mathbf{J}_2\mathbf{Q}_2 \end{bmatrix}. \\ \mathbf{L}_{ch} &= \mathbf{P}^{\mathbf{T}} \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \mathbf{P} = \begin{bmatrix} \alpha_1\mathbf{Q}_1^{\mathbf{T}}\mathbf{J}_1^{\mathbf{T}}\mathbf{K}_{11}\mathbf{J}_1\mathbf{Q}_1 & \sqrt{\alpha_1\alpha_2}\mathbf{Q}_1^{\mathbf{T}}\mathbf{J}_1^{\mathbf{T}}\mathbf{K}_{12}\mathbf{J}_2\mathbf{Q}_2 \\ \sqrt{\alpha_1\alpha_2}\mathbf{Q}_2^{\mathbf{T}}\mathbf{J}_2^{\mathbf{T}}\mathbf{K}_{21}\mathbf{J}_1\mathbf{Q}_1 & \alpha_2\mathbf{Q}_2^{\mathbf{T}}\mathbf{J}_2^{\mathbf{T}}\mathbf{K}_{22}\mathbf{J}_2\mathbf{Q}_2 \end{bmatrix} \\ &= \begin{bmatrix} \alpha_1\{\Lambda_{r_1,1} - \rho\mathbf{I}_{r_1}\} & \sqrt{\alpha_1\alpha_2}\mathbf{L}_{12} \\ \sqrt{\alpha_1\alpha_2}\mathbf{L}_{12}^{\mathbf{T}} & \alpha_2\{\Lambda_{r_2,2} - \rho\mathbf{I}_{r_2}\} \end{bmatrix},\end{aligned}\quad (21)$$

with  $\mathbf{L}_{12} \equiv \mathbf{Q}_1^{\mathbf{T}}\mathbf{J}_1^{\mathbf{T}}\mathbf{K}_{12}\mathbf{J}_2\mathbf{Q}_2$ . The last equality in the above is obtained by using the derivations detailed in Appendix-I.

$$\xi_{ij} \equiv \mathbf{s}_i^{\mathbf{T}}\Phi_i^{\mathbf{T}}\{\mathbf{I}_f - [\Phi_1 \ \Phi_2]\mathbf{B}_{ch} \begin{bmatrix} \Phi_1^{\mathbf{T}} \\ \Phi_2^{\mathbf{T}} \end{bmatrix}\}\Phi_j\mathbf{s}_j = \{\mathbf{s}_i^{\mathbf{T}}\mathbf{K}_{ij}\mathbf{s}_j - \mathbf{s}_i^{\mathbf{T}}[\mathbf{K}_{i1} \ \mathbf{K}_{i2}]\mathbf{B}_{ch} \begin{bmatrix} \mathbf{K}_{1j} \\ \mathbf{K}_{2j} \end{bmatrix} \mathbf{s}_j\}.$$

*Computations related to the KL divergence*

$$\theta_{ijk} \equiv \mathbf{s}_i^{\mathbf{T}}\Phi_i^{\mathbf{T}}(\mathbf{I}_f - \Phi_j\mathbf{B}_j\Phi_j^{\mathbf{T}})\Phi_k\mathbf{s}_k = (\mathbf{s}_i^{\mathbf{T}}\mathbf{K}_{ik}\mathbf{s}_k - \mathbf{s}_i^{\mathbf{T}}\mathbf{K}_{ij}\mathbf{B}_j\mathbf{K}_{jk}\mathbf{s}_k).$$

$$\begin{aligned}
\text{tr}[\mathbf{C}_i \mathbf{C}_j^{-1}] &= \text{tr}[(\Phi_i \mathbf{A}_i \Phi_i^\top + \rho \mathbf{I}_f) \rho^{-1} (\mathbf{I}_f - \Phi_j \mathbf{B}_j \Phi_j^\top)] \\
&= \rho^{-1} \text{tr}[\Phi_i \mathbf{A}_i \Phi_i^\top] - \rho^{-1} \text{tr}[\Phi_i \mathbf{A}_i \Phi_i^\top \Phi_j \mathbf{B}_j \Phi_j^\top] + f - \text{tr}[\Phi_j \mathbf{B}_j \Phi_j^\top] \\
&= \rho^{-1} \text{tr}[\mathbf{A}_i \mathbf{K}_{ii}] - \rho^{-1} \text{tr}[\mathbf{A}_i \mathbf{K}_{ij} \mathbf{B}_j \mathbf{K}_{ji}] + f - \text{tr}[\mathbf{B}_j \mathbf{K}_{jj}] \\
&= \rho^{-1} \text{tr}[\Lambda_{r_i, i}] - r_i - \rho^{-1} \text{tr}[\mathbf{A}_i \mathbf{K}_{ij} \mathbf{B}_j \mathbf{K}_{ji}] + f + \rho \text{tr}[\Lambda_{r_j, j}^{-1}] - r_j \\
&= \rho^{-1} \{ \text{tr}[\Lambda_{r_i, i}] - \eta_{ij} \} + \rho \text{tr}[\Lambda_{r_j, j}^{-1}] + f - (r_i + r_j),
\end{aligned}$$

where  $\eta_{ij} \equiv \text{tr}[\mathbf{A}_i \mathbf{K}_{ij} \mathbf{B}_j \mathbf{K}_{ji}]$ . To get the next last equality in the above, we use (19) and (20) detailed in Appendix-I.

### *Computation related to limiting distances*

First,

$$\lim_{\rho \rightarrow 0} \mathbf{A} = \hat{\mathbf{A}} \equiv \mathbf{J} \mathbf{V}_r \mathbf{V}_r^\top \mathbf{J}^\top, \quad \lim_{\rho \rightarrow 0} \mathbf{B} = \hat{\mathbf{B}} \equiv \mathbf{J} \mathbf{V}_r \Lambda_r^{-1} \mathbf{V}_r^\top \mathbf{J}^\top.$$

Then,

$$\lim_{\rho \rightarrow 0} \theta_{ijk} = \hat{\theta}_{ijk} \equiv \mathbf{s}_i^\top \mathbf{K}_{ik} \mathbf{s}_k - \mathbf{s}_i^\top \mathbf{K}_{ij} \hat{\mathbf{B}}_j \mathbf{K}_{jk} \mathbf{s}_k, \quad \lim_{\rho \rightarrow 0} \eta_{ij} = \hat{\eta}_{ij} \equiv \text{tr}[\hat{\mathbf{B}}_i \mathbf{K}_{ij} \hat{\mathbf{A}}_j \mathbf{K}_{ji}].$$

Similarly,

$$\lim_{\rho \rightarrow 0} \xi_{ij} = \hat{\xi}_{ij} \equiv \mathbf{s}_i^\top \mathbf{K}_{ij} \mathbf{s}_j - \mathbf{s}_i^\top [\mathbf{K}_{i1} \ \mathbf{K}_{i2}] \hat{\mathbf{B}}_{ch} \begin{bmatrix} \mathbf{K}_{1j} \\ \mathbf{K}_{2j} \end{bmatrix} \mathbf{s}_j,$$

where  $\hat{\mathbf{B}}_{ch} = \lim_{\rho \rightarrow 0} \mathbf{B}_{ch}$ .

**Shaohua Kevin Zhou** (S'01–M'04) received his B.E. degree in Electronic Engineering from the University of Science and Technology of China, Hefei, China, in 1994, M.E. degree in Computer Engineering from the National University of Singapore in 2000, and Ph.D. degree in Electrical Engineering from the University of Maryland at College Park in 2004. He is currently a research scientist at Siemens Corporate Research, Princeton, New Jersey.

Dr. Zhou has general research interests in signal/image/video processing, computer vision, pattern recognition, machine learning, and statistical inference and computing, with applications to biometrics recognition, medical imaging, surveillance, etc. Over the past four years, he has written two research monographs on *Unconstrained Face Recognition* (co-authored by R. Chellappa and W. Zhao) and *Recognition of Humans and Their Activities Using Videos* (co-authored by A. Roy-Chowdhury and R. Chellappa), published over 40 book chapters and peer-reviewed journal and conference papers on various topics including face recognition, database-guided echocardiographic image analysis, visual tracking and motion analysis, illumination and pose modeling, kernel machine, and boosting method, and reviewed many papers for over 15 top-rated journals and conferences.

**Rama Chellappa** (S'78–M'79–SM'83–F'92) received the B.E. (Hons.) degree from the University of Madras, Madras, India, in 1975 and the M.E.(Distinction) degree from the Indian Institute of Science, Bangalore, in 1977. He received the M.S.E.E. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1978 and 1981, respectively. Since 1991, he has been a Professor of electrical engineering and an Affiliate Professor of computer science with the University of Maryland, College Park. He is the Director of the Center for Automation Research and a Permanent Member of the Institute for Advanced Computer Studies. Prior to joining the University of Maryland, he was an Associate Professor and Director of the Signal and Image Processing Institute with the University of Southern California, Los Angeles. During the last 22 years, he has published numerous book chapters and peer-reviewed journal and conference papers. Several of his journal papers have been reproduced in collected works published by IEEE Press, IEEE Computer Society Press, and MIT Press. He has edited a collection of papers on *Digital Image Processing* (Santa Clara, CA: IEEE Computer Society Press), co-authored a research monograph on *Artificial Neural Networks for Computer Vision* (with Y. T. Zhou) (Berlin, Germany: Springer-Verlag), and co-edited a book on *Markov Random Fields* (with A. K. Jain) (New York Academic). His current research interests are image compression, automatic target recognition from stationary and moving platforms, surveillance and monitoring, biometrics, human activity modeling, hyper spectral image understanding, and commercial applications of image processing and understanding.

Dr. Chellappa has served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, and IEEE TRANSACTIONS ON NEURAL NETWORKS. He also served as Co-Editor-in-Chief of *Graphical Models and Image Processing*; and a member of the IEEE Signal Processing Society Board of Governors from 1996 to 1999. He is currently serving as the Editor-in-Chief of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and as the vice-president of the IEEE Signal Processing Society for Awards and Membership. He has received several awards, including the 1985 NSF Presidential Young Investigator Award, the 1985 IBM Faculty Development Award, the 1991 Excellence in Teaching Award from the School of Engineering, University of Southern California, the 1992 Best Industry Related Paper Award from the International Association of Pattern Recognition (with Q. Zheng), and the IEEE Signal Processing Society Technical Achievement Award in 2001. He was elected as a Distinguished Faculty Research Fellow (1996–1998) and recently elected as a distinguished Scholar-Teacher for 2003 at the University of Maryland. He is a Fellow of the International Association for Pattern Recognition. He has served as a General and Technical Program Chair for several IEEE international and national conferences and workshops.

## LIST OF FIGURE/TABLE CAPTIONS

Figure 1: Summary of computing the Chernoff distance in RKHS.

Figure 2: 300 i.i.d. realizations of four different densities with the same mean (zero mean) and covariance matrix (identity matrix). (a) 2-D Gaussian. (b) ‘O’-shaped uniform. (c) ‘D’-shaped uniform. (d) ‘X’-shaped uniform.

Figure 3: (a) The Bhattacharyya distance  $\hat{J}_B(\sigma, r)$  and (a) the divergence distance  $\hat{J}_D(\sigma, r)$  between the 2-D Gaussian and the ‘O’-shaped uniform as a function of  $\sigma$  and  $r$ .

Figure 4: Original binary images of digits and their normalized sample representations.

Figure 5: Recognition error rates as a function of  $r$  and  $\sigma$  obtained using the 1-NN rule based on (a)  $\hat{J}_B(\mathfrak{p}_1, \mathfrak{p}_2)$  and (b)  $\hat{J}_D(\mathfrak{p}_1, \mathfrak{p}_2)$  and the SVM based on (c)  $\hat{J}_B(\mathfrak{p}_1, \mathfrak{p}_2)$  and (d)  $\hat{J}_D(\mathfrak{p}_1, \mathfrak{p}_2)$ .

Figure 6: The mean and covariance curves of recognition error rate as a function of  $r$  with different  $\sigma$ 's using (a) the Bhattacharyya distance  $\hat{J}_B(\mathfrak{p}_1, \mathfrak{p}_2)$  and (b) the symmetric KL divergence  $\hat{J}_D(\mathfrak{p}_1, \mathfrak{p}_2)$ .

Figure 7: Examples of face images in the gallery and probe set. (a) The 4<sup>th</sup> gallery person in 10 frames (every 8 frames) of a 80-frame sequence. (b) The 9<sup>th</sup> gallery person in 10 frames (every 10 frames) of a 105-frame sequence. (c) The 4<sup>th</sup> probe person in 10 frames (every 6 frames) of a 60-frame sequence. (d) The plot of first three PCA coefficients of the above three sets.

Table I: A list of probabilistic distances and their definitions, where  $0 < \alpha_1, \alpha_2 < 1$  and  $\alpha_1 + \alpha_2 = 1$ .

Table II: Analytic expressions of probabilistic distances between two normal densities.

Table III: (a) The symmetric KL divergence in the RKHS with  $\sigma = 1$  and  $r = 3$ . (b) The Bhattacharyya distance in the RKHS with  $\sigma = 0.5$  and  $r = 1$ .  $\mathfrak{p}_1$  is listed in the first column and  $\mathfrak{p}_2$  in the first row.

Table IV: The recognition scores obtained by using the probabilistic distance measures in different spaces.

Table V: A list of important quantities used in the paper.

## COMPLETE CONTACT INFORMATION

- Shaohua Kevin Zhou (corresponding author)  
Integrated Data Systems Department  
Siemens Corporate Research  
755 College Road East, Princeton, NJ 08540  
Email: {kzhou}@scr.siemens.com  
Phone: 609-734-3324  
Fax: 609-734-6565
  
- Rama Chellappa  
Center for Automation Research and  
Department of Electrical and Computer Engineering  
University of Maryland, College Park, MD 20742  
Email: {rama}@cfar.umd.edu  
Phone: 301-405-3656  
Fax: 301-314-9115