Distribution Calibration in Riemannian Symmetric Space

Si Si, Wei Liu, Dacheng Tao, *Member, IEEE*, and Kwok-Ping Chan, *Member, IEEE*

***Abstract*—Distribution calibration plays an important role in cross-domain learning. However, existing distribution distance metrics are not geodesic; therefore, they cannot measure the intrinsic distance between two distributions. In this paper, we calibrate two distributions by using the geodesic distance in Riemannian symmetric space. Our method learns a latent subspace in the reproducing kernel Hilbert space, where the geodesic distance between the distribution of the source and the target domains is minimized. The corresponding geodesic distance is thus equivalent to the geodesic distance between two symmetric positive definite (SPD) matrices defined in the Riemannian symmetric space. These two SPD matrices parameterize the marginal distributions of the source and target domains in the latent subspace. We carefully design an evolutionary algorithm to find a local optimal solution that minimizes this geodesic distance. Empirical studies on face recognition, text categorization, and web image annotation suggest the effectiveness of the proposed scheme.**

***Index Terms*—Cross-domain learning, distribution calibration, Riemannian symmetric space, subspace learning.**

I. Introduction

CROSS-DOMAIN learning algorithms leverage knowledge learned from the source domain for use in the target domain, where both domains are different but related [10]. Cross-domain learning has widely been applied to machine learning [7], [18], pattern recognition [25], [31], [32], and image processing [8], [9], [17], [27], particularly for the case when it is relatively difficult or expensive to collect labeled training samples.

Recently, distribution calibration algorithms have been introduced to cross-domain learning. They minimize the mismatch between the distribution of the training samples and the test samples. Theoretically, samples from the source and target domains can be deemed to be drawn from a same distribution after the distribution calibration. Practically, sufficient applications have shown the effectiveness of distribution calibration algorithms for cross-domain learning. Dimension reduction [5], [30], [33] is often involved in distribution calibration algorithms for cross-domain learning. Basically, when all the samples from the source and target domains are projected into a low-dimensional subspace by some dimension reduction algorithms designed for cross-domain learning [28], their distribution bias can be calibrated.

One major concern for distribution calibration is measuring the distribution discrepancy between the source and target domains [29]. The maximum mean discrepancy (MMD) [3] is one of the most widely used nonparametric criteria in cross-domain learning for estimating the distance between different distributions. In MMD, the distance between distributions of two sets of samples can be estimated as the maximum Euclidean distance between the means of samples from the two domains in the reproducing kernel Hilbert space (RKHS). Many cross-domain learning algorithms have been developed based on this criterion. For example, Pan *et al.* (2008) proposed a transductive dimension reduction algorithm, i.e., maximum mean discrepancy embedding (MMDE), to minimize MMD in a latent subspace for cross-domain text categorization. Domain transfer support vector machine (DTSVM) [13] was proposed to cope with the change of feature distribution between different domains in video concept detection, and the change was calculated by MMD. Transfer component analysis (TCA) [22] was proposed to find a set of common transfer components for simultaneously matching distributions, also measured in MMD, a cross source, and target domains to adapt an indoor Wi-Fi localization model.

However, MMD applied to the aforementioned distribution calibrate techniques for cross-domain learning approaches suffers from two major disadvantages. First, MMD is not geodesic, i.e., it cannot discover the intrinsic distance between two probability densities. Many applications require the distribution distance measure, e.g., the geodesic distance, to reflect the underlying structure of the data. Second, MMD cannot handle the situation when covariances of probability densities are quite different, because it only considers the mean difference.

To solve the aforementioned two problems, in this paper, we calibrate distributions in the Riemannian symmetric space [1]. The proposed algorithm is referred to as the distribution calibration in Riemannian symmetric space (DC-RSS). It minimizes the distribution difference between different domains in a low-dimensional latent subspace. In particular, we first map all samples into RKHS and model the marginal distributions of the source and target domains in RKHS as two different Gaussians.

Manuscript received April 25, 2010; revised September 14, 2010 and November 30, 2010; accepted December 5, 2010. Date of publication January 6, 2011; date of current version July 20, 2011. This paper was recommended by Associate Editor G.-B. Huang.

S. Si was with the University of Hong Kong, Hong Kong, China. She is now with the Department of Computer Science, The University of Texas at Austin, Austin, TX 78701 USA (e-mail: ssi@cs.hku.hk).

W. Liu is with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA (e-mail: wliu@ee.columbia.edu).

D. Tao is with the Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW 2007, Australia (e-mail: dacheng.tao@uts.edu.au).

K.-P. Chan is with the Department of Computer Science, University of Hong Kong, Hong Kong, China (e-mail: kpchan@cs.hku.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2010.2100042

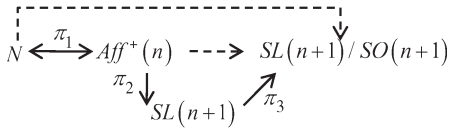


Fig. 1. Parameterizing the multivariate normal distribution space N as the Riemannian symmetric space $SL(n+1)/SO(n+1)$.

Information geometry [2] shows that parameters of Gaussian distributions are embedded in the Riemannian symmetric space. Therefore, after projecting samples from RKHS onto the subspace, these two Gaussians can be represented as two SPD matrices in Riemannian symmetric space through parameterization. In particular, three consecutive projections as shown in Fig. 1 are conducted to parameterize Gaussians in multivariate normal distribution space as SPDs in the Riemannian symmetric space. Based on the differential geometric structure upon SPD matrices in the Riemannian symmetric space [24], the geodesic distance between any two SPD matrices can be measured in a compact form, and thus, the distance function between the source and target domains is accordingly geodesic.

In addition, because there is no assumption on the means or covariances of Gaussians in our framework, differences on both means and covariances can be taken into account. Therefore, we can arrive at a cross-domain learning method by minimizing the geodesic distance in the Riemannian symmetric space. However, the gradient or Hessian of the objective function for DC-RSS is not of a compact form; therefore, it is difficult to find a suitable solution by using conventional optimization approaches, e.g., gradient descent or Newton's methods. In this paper, the evolutionary algorithm (EA) [12], a kind of global search heuristics, is carefully introduced to optimize DC-RSS.

The rest of this paper is organized as follows. Section II presents DC-RSS, analyzes DC-RSS's solution format, and develops an EA to solve DC-RSS. Section III illustrates the following three important applications for DC-RSS: 1) cross-domain face recognition; 2) text categorization; and 3) web image annotation on the machine learning database. Section IV concludes this paper.

II. DC-RSS

Measuring the distance between probability densities is of great importance in distribution calibration. To have a compact function for measuring the geodesic distance between two probability densities, we parameterize the multivariate normal distribution space as the Riemannian symmetric space, i.e., distributions of source and target domains can be represented by two symmetric positive definite (SPD) matrices. Then, the geodesic distance between two probability densities is equivalent to the geodesic distance between two corresponding SPD matrices in Riemannian symmetric space.

A. Problem Statement

In cross-domain learning, we have the following two sets of samples: 1) the l training samples $X_s = \{(x_i, y_i)\}_{i=1}^{l}$ from the source domain, where $x_i \in \mathcal{X}$ is the ith input feature, and $y_i \in \mathcal{Y}$ is the corresponding discrete label, and 2) the u test samples $X_t = \{(x_{i+l}, y_{i+l})\}_{i=1}^{u}$ drawn from the target domain, where the label y_i is unknown. The marginal distributions of the training and test samples are $P(X_s)$ and $Q(X_t)$, and $P(X_s) \neq Q(X_t)$ is the general assumption in cross-domain learning. In this paper, samples are transformed into RKHS by using a nonlinear transformation $\phi : \mathcal{X} \to \mathcal{H}$, wherein $\mathcal{H}$ is a universal RKHS. Let $X_s^\phi = \{\phi(x_i)\}_{i=1}^{l}$ and $X_t^\phi = \{\phi(x_i)\}_{i=l+1}^{l+u}$ denote the transformed input features from the source and target domains, respectively, and their corresponding marginal distributions are $P(X_s^\phi)$ and $Q(X_t^\phi)$. To have a compact distance function, we model both $P(X_s^\phi)$ and $Q(X_t^\phi)$ in RKHS as two Gaussians with different means (μ_s and μ_t) and covariance matrices (Σ_s and Σ_t). The proposed DC-RSS searches for a latent linear subspace W, and when all samples are projected into it, the geodesic distance between the marginal distributions of the source and target domains is minimized. Denote $Y_s = W^T X_s^\phi$ and $Y_t = W^T X_t^\phi$ as the training and test samples' low-dimensional representations, respectively. Their corresponding probability densities are $P(Y_s)$ and $Q(Y_t)$. As a consequence, DC-RSS is designed to find a W so that the geodesic distance between $P(Y_s)$ and $Q(Y_t)$ is minimized, i.e.,

$$W = \arg\min d_R\left(P(Y_s), Q(Y_t)\right). \tag{1}$$

Because $P(X_s^\phi)$ and $Q(X_t^\phi)$ are approximated by Gaussians in RKHS, the corresponding projections $P(Y_s)$ and $Q(Y_t)$ are also Gaussians in the multivariate normal distribution space. As a consequence, the means for the source and target domains in the subspace become $W^T\mu_s$ and $W^T\mu_t$, and their corresponding covariances are $W^T\Sigma_s W$ and $W^T\Sigma_t W$, respectively. Two well-known distance metrics for probability densities are the Kullback–Leibler (KL) divergence and the Euclidean measure. However, neither of these metrics is geodesic, i.e., they cannot measure the intrinsic distance between $P(Y_s)$ and $Q(Y_t)$. Therefore, it is necessary to introduce a geodesic distance to measure the distance between $P(Y_s)$ and $Q(Y_t)$. In this paper, we project $P(Y_s)$ and $Q(Y_t)$ into the Riemannian symmetric space as two SPD matrices and then measure the geodesic distance between two SPD matrices by the associated Riemannian distance metric.

B. Parameterization

In the rest of this paper, $SL(n+1)/SO(n+1)$ refers to the Riemannian symmetric space, wherein $SL(n+1)$ is the simple Lie group, and $N = \{\gamma|dx|\}$ is the multivariate normal distribution space associated with the Lebesgue measure dx on R^n.

According to [20], three consecutive projections, i.e., π_1, π_2, and π_3, are introduced to parameterize the multivariate normal distribution space to Riemannian symmetric space, i.e.,

$$N \to SL(n+1)/SO(n+1). \tag{2}$$

After these projections, two Gaussians $P(Y_s)$ and $Q(Y_t)$ can be identified by two SPD matrices (P and Q) in $SL(n+1)/SO(n+1)$, respectively, upon which the geodesic distance between P and Q, i.e., $d_R(P, Q)$, can accordingly be calculated

to measure the geodesic distance $d_R(P(Y_s), Q(Y_t))$ between $P(Y_s)$ and $Q(Y_t)$.

To parameterize normal distributions in a group structure, an affine group $Aff^+(n)$ is constructed. Let $GL(n)$ be the general linear group that contains all nonsingular matrices and R^n be the n-dimensional vector space, and thus, the affine group $Aff^+(n)$ is

$$Aff^+(n) = \left\{ \Phi_{\sigma,\mu} : x \mapsto \sigma x + \mu | \quad \begin{matrix} \sigma \in GL(n) \\ \mu \in R^n, \det\sigma > 0 \end{matrix} \right\}. \tag{3}$$

The affine group is a kind of Lie group and consists of all invertible affine transformations from the space to itself. It transitively acts on N by using $\pi_1 : Aff^+(n) \leftrightarrow N$, i.e.,

$$\pi_1 : \Phi_{\sigma,\mu} \mapsto \left(\Phi_{\sigma,\mu}^{-1}\right)(\gamma_0|dx|) \tag{4}$$

where $\gamma_0|dx| = (2\pi)^{-n/2} e^{-1/2x^T x}$ is the standard normal distribution on R^n, μ and Σ are the mean and covariance for the normal distribution, and σ can be obtained by the Cholesky decomposition of covariance Σ, i.e., $\Sigma = \sigma\sigma^t$. After the π_1 projection, each normal distribution in N can be represented by an element in $Aff^+(n)$.

Afterward, we project $Aff^+(n)$ into $SL(n+1)$, which is a simple Lie group, by $\pi_2 : Aff^+(n) \to SL(n+1)$, i.e.,

$$\pi_2 : \Phi_{\sigma,\mu} \mapsto (\det\sigma)^{-\frac{1}{n+1}} \begin{bmatrix} \sigma & \mu \\ 0 & 1 \end{bmatrix}. \tag{5}$$

Finally, we use π_3 to project $SL(n+1)$ into $SL(n+1)/SO(n+1)$ according to

$$\pi_3 : \sigma \mapsto \sigma\sigma^t \tag{6}$$

where $\sigma\sigma^t \in SL(n+1)/SO(n+1)$, with $\sigma \in SL(n+1)$.

We can arrive at $N \to SL(n+1)/SO(n+1)$ by combining these three consecutive projections π_1, π_2, and π_3, i.e.,

$$\left(\Phi_{\sigma,\mu}^{-1}\right)(\gamma_0|dx|) \mapsto (\det\sigma)^{-\frac{2}{n+1}} \begin{bmatrix} \sigma\sigma^t + \mu\mu^t & \mu \\ \mu^t & 1 \end{bmatrix} \tag{7}$$

where the matrix $(\det\sigma)^{-(2/n+1)} \begin{bmatrix} \sigma\sigma^t + \mu\mu^t & \mu \\ \mu^t & 1 \end{bmatrix}$ is an SPD matrix in $SL(n+1)/SO(n+1)$. Fig. 1 provides a canonical identification of N in $SL(n+1)/SO(n+1)$. In particular, this identification can be achieved by projecting N into $Aff^+(n)$ through π_1, then embedding $Aff^+(n)$ in $SL(n+1)$ through π_2, and finally transforming $SL(n+1)$ to $SL(n+1)/SO(n+1)$ through π_3.

According to (7), two Gaussians $P(Y_s)$ and $Q(Y_t)$ in N can thus be identified by two SPD matrices, and their corresponding identifications P and Q in $SL(n+1)/SO(n+1)$ are

$$\begin{aligned} P(Y_s) \mapsto P &= |\Sigma_s'|^{-\frac{1}{n+1}} \begin{bmatrix} \Sigma_s' + \mu_s'\mu_s'^T & \mu_s' \\ \mu_s'^T & 1 \end{bmatrix} \\ Q(Y_t) \mapsto Q &= |\Sigma_t'|^{-\frac{1}{n+1}} \begin{bmatrix} \Sigma_t' + \mu_t'\mu_t'^T & \mu_t' \\ \mu_t'^T & 1 \end{bmatrix} \end{aligned} \tag{8}$$

where μ_s' and μ_t' are the means of Y_s and Y_t, i.e., $\mu_s' = W^T\mu_s$ and $\mu_t' = W^T\mu_t$, and Σ_s' and Σ_t' are their corresponding covariance matrices, i.e., $\Sigma_s' = W^T\Sigma_s W$ and $\Sigma_t' = W^T\Sigma_t W$, respectively.

C. Geometry of the Riemannian Symmetric Space

Taking the simple Lie group $SL(n+1)$ as the bridge, through the three consecutive projections π_1, π_2, and π_3, the normal distributions in N can uniquely be identified by SPD matrices in $SL(n+1)/SO(n+1)$. Afterward, we can use the distance metric between SPD matrices in $SL(n+1)/SO(n+1)$ to measure the geodesic distance between two Gaussians in N. The differential geometric structure upon SPD matrices in $SL(n+1)/SO(n+1)$ is well defined, and the geodesic distance between any two SPD matrices can be measured in a compact form. Therefore, the distance between distributions in N is accordingly geodesic.

Let $P(n+1)$ consist of all $(n+1)\times(n+1)$ SPD matrices in $SL(n+1)/SO(n+1)$. According to [21], the geodesic distance between any two matrices P and Q in $P(n+1)$ is

$$d_R(P,Q) = \left\| Log(P^{-1}Q) \right\|_F \tag{9}$$

where $\|\cdot\|$ is the Frobenius matrix norm, and $Log(\cdot)$ is the principal matrix logarithm or, equivalently, the inversion of the matrix exponential.

Our objective is to find the projection subspace W, where the geodesic distance between the source and target domains distributions $P(Y_s)$ and $Q(Y_t)$ is minimized, i.e., $P(Y_s)$ and $Q(Y_t)$ are matched to each other. As a consequence, the objective of DC-RSS turns into

$$J(W) = \min_W \left\| Log\left(P(W)^{-1}Q(W)\right) \right\|_F. \tag{10}$$

However, it is not trivial to obtain W in (10). To obtain a suitable W, it is necessary to prove that the representer theorem holds for the optimization problem defined in (10).

Theorem 1: Representer Theorem: Let w_i be the projection vector in the projection matrix W, and then, each minimizer $W = [w_1, \ldots, w_d]$ of $J(W)$ has the following representation:

$$w_s = \sum_{i=1}^{l+u} \alpha_s^i \phi(x_i) \tag{11}$$

where, $\forall i \in \{1, \ldots, l+u\}$ and $\forall s \in \{1, \ldots, d\}$, $\alpha_s^i \in \mathrm{R}$, $\sum_{i=1}^{l+u} \alpha_s^i = 0$, and $\alpha_s = [\alpha_s^1, \ldots, \alpha_s^{l+u}]^T \in \mathrm{R}^{l+u}$; ϕ is a nonlinear transformation $\phi : \mathcal{X} \to \mathcal{H}$.

Proof: Let $\mathcal{H}_k$ be an RKHS associated with a kernel $k : x \times x \to \mathrm{R}$, which is a symmetric SPD function on the compact domain. Because we have assumed that k maps into R, we will use $\phi : \mathcal{X} \to \mathrm{R}^{\mathcal{X}}$, $x \mapsto k(.,x)$. Because k is a reproducing kernel, for all $x, x' \in \mathcal{X}$, the evaluation of the function on the point $\phi(x)$ yields

$$\phi(x)(x') = k(x', x) = \langle \phi(x'), \phi(x) \rangle \tag{12}$$

where $\langle .,. \rangle$ denotes the dot product defined on $\mathcal{H}_k$. Given all the samples in RKHS $\{\phi(x_i)\}_{i=1}^{l+u}$, any w_s can be decomposed

into a part that lies in the span of the $\phi(x_i)$, $\sum_{i=1}^{l+u}\alpha_s^i\phi(x_i)$ and a part v that is orthogonal to it, i.e.,

$$w_s = \sum_{i=1}^{l+u}\alpha_s^i\phi(x_i) + v, \langle v, \phi(x_i)\rangle = 0. \tag{13}$$

According to (13), the projection of an arbitrary sample $\phi(x_j)$ by w_s yields

$$\begin{aligned} w_s^T\phi(x_j) &= \left\langle \sum_i \alpha_s^i\phi(x_i) + v, \phi(x_j)\right\rangle \\ &= \sum_i \alpha_s^i \langle\phi(x_i), \phi(x_j)\rangle . \end{aligned} \tag{14}$$

It is obvious that (14) is independent of v. The calculation of covariance matrices, i.e., Σ_s', Σ_t', and means, i.e., μ_s', μ_t', in $J(W)$ are all based on the low-dimensional representation of samples, e.g., $w_s^T\phi(x_j)$. As a result, $J(W)$ is independent of v, and any solution w_s in W takes the form $w_s = \sum_{i=1}^{l+u}\alpha_s^i\phi(x_i)$. Furthermore,

$$W = \phi(X)A_{DC} \tag{15}$$

where $A_{DC} = [\alpha_1, \ldots, \alpha_d]$, and $\phi(X) = X_s^\phi \cup X_t^\phi = \{\phi(x_i)\}_{i=1}^{l+u}$. This expression completes the proof. ■

Based on $W = \phi(X)A_{DC}$ derived from the representer theorem, the representation of P in (8) and (10) in RKHS can thus be directly rewritten as

$$P = |\tilde{\Sigma}_s|^{-\frac{1}{n+1}}\begin{bmatrix} \left(\tilde{\Sigma}_s + \tilde{\mu}_s\tilde{\mu}_s^t\right) & \tilde{\mu}_s \\ \tilde{\mu}_s^t & 1 \end{bmatrix} \tag{16}$$

where K is an $(l+u)\times(l+u)$ kernel Gram matrix with entry $K_{i,j} = \phi^T(x_i)\phi(x_j)$, k_i is the ith column of K, and $\tilde{\mu}_s$ and $\tilde{\Sigma}_s$ are the mean and covariance of the training samples in RKHS, respectively, which can be calculated as

$$\begin{aligned} \tilde{\mu}_s &= \frac{1}{l}\sum_{i=1}^{l} W^T\phi(x_i) = \frac{1}{l}A_{DC}^T\sum_{i=\mathbf{1}}^{l} k_i \\ \tilde{\Sigma}_s &= \frac{1}{l}A_{DC}^T\left(\sum_{i=1}^{l}k_ik_i^T - \frac{1}{l}\sum_{i=1}^{l}k_i\sum_{j=1}^{l}k_j^T\right)A_{DC}. \end{aligned} \tag{17}$$

The kernel form of Q can similarly be obtained. As a consequence, the optimization problem (10) turns to learning the optimal linear combination coefficients matrix A_{DC}. However, the size of A_{DC} is directly proportional to the number of training and test samples, i.e., $l+u$, and thus does not scale up well when $l+u$ is relatively large. To solve this problem, we prove that learning DC-RSS in RKHS is equivalent to learning DC-RSS in the space spanned by the principal components of the kernel principle component analysis (KPCA) [26] in Theorem 2. As a result, we can dramatically reduce the time cost in DC-RSS.

Theorem 2: Learning A_{DC} in (10) is equivalent to applying DC-RSS in the space spanned by the principal components of KPCA, i.e., DC-RSS in RKHS is equal to KPCA, followed by DC-RSS with the linear kernel.

Proof: Denote the covariance matrix for all the samples in RKHS, i.e., $\phi(X)$, by $C = (1/l+u)\sum_{j=1}^{l+u}\phi(x_j)\phi(x_j)^T$. For KPCA, we aim at finding the eigenvector u and the eigenvalue λ that satisfy $Cu = \lambda u$. This problem is equivalent to solving

$$\phi(x_j)Cu = \lambda\phi(x_j)u \text{ for } j = 1, \ldots, l+u. \tag{18}$$

Based on the representer theorem, it is not difficult to prove that the ith eigenvector u_i in (18) is in the span of all the samples, i.e., $u_i \in span(\{\phi(x_j)\}_{j=1}^{l+u})$ or, more specifically, $u_i = \sum_{j=1}^{l+u}\beta_i^j\phi(x_j)$. Thus, (18) is equivalent to the following optimization problem:

$$K\beta_{KPCA} = \lambda\beta_{KPCA} \tag{19}$$

where K is a kernel function, with $K_{i,j} = \phi^T(x_i)\phi(x_j)$, $\beta_i = [\beta_i^1, \ldots, \beta_i^{l+u}]^T$, and $\beta_{KPCA} = [\beta_1, \ldots, \beta_{l+u}]$. The solution of the aforementioned eigendecomposition is the eigenvector β_i, and the corresponding eigenvalue is λ_i. Therefore, the projection matrix of KPCA, $U = [u_1, \ldots, u_{l+u}]$, is given by $U = \phi(X)\beta_{KPCA}$. Because of the constraint $U^TU = I$, we can arrive at

$$\left(\phi(X)\beta_{KPCA}\right)^T\left(\phi(X)\beta_{KPCA}\right) = \beta_{KPCA}^TK\beta_{KPCA} = I \tag{20}$$

which results in $\beta_{KPCA}^T\beta_{KPCA} = K^{-1}$ (because $\beta_{KPCA}^T\beta_{KPCA}$ is full rank).

Consequently, the projected x_i in KPCA space is given by

$$\hat{x}_i = U^T\phi(x_i) = \beta_{KPCA}^Tk_i \tag{21}$$

where k_i is the ith column of K. Therefore, all the samples preprocessed by KPCA become $\hat{X} = [\hat{x}_1, \ldots, \hat{x}_{l+u}] = \beta_{KPCA}^TK$.

Denote the mean and the covariance with linear kernels over training samples $\hat{X}_s = [\hat{x}_1, \ldots, \hat{x}_l]$ in the KPCA space by $\hat{\mu}_s$ and $\hat{\Sigma}_s$, respectively. Then, we have

$$\begin{aligned} \hat{\mu}_s &= \frac{1}{l}\sum_{i=1}^{l}\hat{x}_i = \frac{1}{l}\beta_{KPCA}^T\sum_{i=1}^{l}k_i, \text{ and} \\ \hat{\Sigma}_s &= \frac{1}{l}\sum_{i=1}^{l}(\hat{x}_i - \hat{\mu}_s)(\hat{x}_i - \hat{\mu}_s)^T \\ &= \frac{1}{l}\beta_{KPCA}^{\mathbf{T}}\left(\sum_{i=1}^{l}k_ik_i^T - \frac{1}{l}\sum_{i=1}^{l}\sum_{j=1}^{l}k_ik_j^T\right)\beta_{KPCA}. \end{aligned} \tag{22}$$

The mean $\hat{\mu}_t$ and covariance $\hat{\Sigma}_t$ with linear kernels over test samples $\hat{X}_t = [\hat{x}_{l+1}, \ldots, \hat{x}_{l+u}]$ in the KPCA space can similarly be derived. Next, we project all the samples in the KPCA space to a subspace by the projection matrix W. According to the representer theorem, the projection matrix $W = [w_1, \ldots, w_d]$ is in the span of all the samples projected in the KPCA space, i.e., $w_i = \sum_{j=1}^{l+u}\alpha_i^j\hat{x}_j$. As a result

$$w_i = \beta_{PCA}^TK\alpha_i^{KPCA} \tag{23}$$

where $\alpha_i^{KPCA} = [\alpha_i^1, \ldots, \alpha_i^{l+u}]^T$, and $W = \beta_{KPCA}^T K A_{DC}^{KPCA}$, with $A_{DC}^{KPCA} = [\alpha_1^{KPCA}, \ldots, \alpha_d^{KPCA}]$.

After the projection W, the mean and covariance for the training samples in the subspace become $W^T\hat{\mu}_s$ and $W^T\hat{\Sigma}_s W$, respectively. Based on (22) and $W = \beta_{KPCA}^T K A_{DC}^{KPCA}$, we have

$$\begin{aligned} W^T\hat{\mu}_s &= \frac{1}{l} W^T \sum_{i=1}^{l} \phi(x_i) = \frac{1}{l} \left(A_{DC}^{KPCA}\right)^T \sum_{i=\mathbf{1}}^{l} k_i \\ W^T\hat{\Sigma}_s W &= \frac{1}{l} W^T \sum_{i=1}^{l} (\hat{x}_i - \hat{\mu}_s)(\hat{x}_i - \hat{\mu}_s)^T W \\ &= \frac{1}{l} \left(A_{DC}^{KPCA}\right)^T \\ &\quad \times \left(\sum_{i=1}^{l} k_i k_i^T - \frac{1}{l} \sum_{i=1}^{l} k_i \sum_{j=1}^{l} k_j^T \right) A_{DC}^{KPCA}. \end{aligned} \tag{24}$$

As a consequence, $W^T\hat{\Sigma}_s W = \tilde{\Sigma}_s$, and $W^T\hat{\mu}_s = \tilde{\mu}_s$, and similarly, $W^T\hat{\Sigma}_t W = \tilde{\Sigma}_t$, and $W^T\hat{\mu}_t = \tilde{\mu}_t$. In other words, the mean and covariance do not change if we apply KPCA, followed by DC-RSS, instead of directly learning DC-RSS in RKHS with the linear kernel. Thus the optimization of (10) is equivalent to preprocessing data by KPCA and then applying DC-RSS to find the solution W. This expression completes the proof. ■

According to Theorem 2, we can make use of KPCA to preprocess the data and then conduct DC-RSS in the subspace spanned by KPCA's most important nonlinear principal components. This way, the time cost can significantly be reduced in DC-RSS.

D. EA for Optimization

However, neither the gradient nor the Hessian of the objective function defined in (10) are compact; therefore, it is difficult to obtain its solution by using conventional optimization algorithms, e.g., the gradient descent and Newton's method. Furthermore, (10) is not convex; therefore, it could be improper to apply the gradient descent method, which can only search a local solution. In this paper, EA is utilized to solve (10) so that we can obtain a better local solution of DC-RSS to suppress the local optimality of conventional optimization algorithms. EA is a generic population-based metaheuristic optimization strategy and analogy to the metaphor of natural biological evolution. It operates by searching on the population of potential solutions, applying the principal of survival of the fittest, and then iteratively generating new offspring according to their fitness values. EA will process for generations until the best solution is found. As a consequence, along with the EA process, the individual will become much more suitable for the optimization problems, e.g., the value of the objective in (10) will decrease.

First, a population of individuals that represent the projection matrices are randomly selected from the search space, where the search space Δ consists of the d-dimensional vectors, $\Delta = \{\alpha_i \in \mathcal{X}^d | i = 1, 2, \ldots, m\}$, where d is the dimension of the data preprocessed by KPCA. An individual or a new projection matrix W can be constructed by linearly combining the vectors from the basis vectors in Δ and then orthnormalizing the composed matrix. According to the aforementioned method of generating a projection matrix W, an individual can be represented as a vector, $v = [a_1, a_2, \ldots, a_m, b_1, \ldots b_m]$, where a_1 is a selection bit that indicates whether the ith basis vector α_i will be selected to construct W. If so, the corresponding combination coefficient can be b_i. Otherwise, b_i is not taken into account. Therefore, an individual under such definition can achieve a low space complexity.

After initialization, we calculate every individual's fitness value in this population. The larger the fitness value for individual v, the more likely that it will be the solution of the optimization problem. As a consequence, the fitness function is directly relative to the objective function and equal to the inverse of (10), i.e.,

$$Fitness(v) = -\left\| Log(P^{-1}Q) \right\|_F. \tag{25}$$

Algorithm 1: DC-RSS

Input: Preprocessed samples from source and target domains by KPCA; search space Δ; maximum population size n; the number of the individual m in one population; $\varepsilon > 0$.

Output: Projection matrix W.

Initialize: Randomly select a population of individuals from Δ.

repeat

$t \leftarrow t + 1$.

if $\mu_t - \mu_{t-1} > \varepsilon$ **then**

for $s = 1$ **to** m **do**

1. Decode the individual v_s in the tth population to construct W based on Δ.
2. Project all the samples from the source and target domains into a subspace by W.
3. Calculate the source and target domains' means, i.e., $W^T\hat{\mu}_s$ and $W^T\hat{\mu}_t$, and covariances, i.e., $W^T\hat{\Sigma}_s W$ and $W^T\hat{\Sigma}_t W$, in the subspace.
4. Parameterize $SL(n+1)/SO(n+1)$ to N by using (7).
5. Obtain the fitness value of v_s in (25).

end for

Calculate the mean of all the individuals' fitness value for tth population as μ_t based on (25).

end if

until $t > n$

After all individual's values are calculated, we can check whether the mean of all the individuals' fitness values in this population changes compared with the anterior population. If not, we output the individuals. Otherwise, we randomly select two individuals through tournament selection and undertake EA

Fig. 2. First row: images from the UMIST database for training. Second row: images from the YALE database for testing (U2Y setting).

operations under a certain probability to generate new individuals. Tournament selection [12] is a fitness-based process, i.e., the possibility of an individual of being a winner of the tournament selection is directly related to its fitness value. Thus, the larger the fitness value for individual v is, the more likely that the individual will be selected to produce offspring by the following two kinds of operations in EA: 1) mutation or 2) crossover.

For the crossover operation, after the tournament selection of two individuals $v_i = [a_1^i, a_2^i, \ldots, a_m^i, b_1^i, \ldots, b_m^i]$ and $v_j = [a_1^j, a_2^j, \ldots, a_m^j, b_1^j, \ldots, b_m^j]$ from the population, we randomly select two crossover points and implement an exchange procedure between these two individuals (e.g., if two crossover points are set as a_m and b_2, two segments a_m^i, b_1^i, b_2^i and a_m^j, b_1^j, b_2^j are exchanged in the crossover operation, and hereby, two new individuals are generated).

For mutation, after the tournament selection of an individual v from this population, every selection bit a_i and every bit in combination coefficient b_i in v is subject to mutation from 0 to 1, or vice versa, under a certain probability, and thus, a new individual will be generated.

The operations of crossover and mutation can help in keeping the diversity of the population and preventing premature convergence on poor solutions. The aforementioned generation process is repeated several times until the fitness value in (25) is unchanged or slightly changed. The overall procedure of the proposed DC-RSS is shown in Algorithm 1.

III. Experiments

In this section, we investigate the effectiveness of DC-RSS on the following three cross-domain learning tasks: 1) cross-domain face recognition; 2) text categorization; and 3) web image annotation. To demonstrate the superiority of DC-RSS, we will compare DC-RSS with three classical subspace learning algorithms, including principal component analysis (PCA) [16], Fisher's linear discriminative analysis (FLDA) [14] and semisupervised discriminate analysis (SDA) [6]. These three algorithms assume that the source- and target-domain samples are independent and identically distributed and thus are not cross-domain learning algorithms. Furthermore, to show the effectiveness of DC-RSS for distribution calibration under the cross-domain setting, we compare DC-RSS with two popular cross-domain learning algorithms, i.e., MMDE [23] and TCA [22], both of which apply MMD as the metric for calibrating the distribution between the source and target domains.

A. Cross-Domain Face Recognition

The first experiment is conducted for cross-domain face recognition. Because there is no public face database constructed under the cross-domain setting, we build two new data sets by combining the University of Manchester Institute of Science and Technology (UMIST) face database [15] and the YALE face database [4]. The UMIST database consists of 564 images from 20 people with different gender, races, and appearances, covering a range of poses from profile to front views. The YALE database includes 165 images from 15 individuals captured under different facial expressions and configurations. The images from both YALE and UMIST used for our experiments are of size 40×40 in raw pixel. Based on YALE and UMIST, we can construct the following two cross-domain face data sets: 1) Y2U, where the source domain is YALE, and the target domain is UMIST, and 2) U2Y, where the source domain is UMIST, and the target domain is YALE. Example face images from the U2Y database are shown in Fig. 2. Obviously, the source and target domains for both Y2U and U2Y belong to different domains and thus are suitable for cross-domain learning. To compare DC-RSS with other algorithms, first, each algorithm is applied to find the low-dimensional representation of the samples from the target domain. Then, we calculate the distance between a test sample and every reference sample, and using the nearest neighbor (NN) classifier to predict the label of the test sample. It is worth emphasizing that the label of reference samples is blind to all algorithms in the training stage.

Fig. 3 shows the detailed process of DC-RSS for cross-domain face recognition. In the U2Y data set, we take the UMIST face data set as the source domain and the YALE face database as the target domain. After preprocessing the data from UMIST and YALE by using KPCA, DC-RSS projects them into a subspace represented as generated by EA. Then, DC-RSS parameterizes the distribution of UMIST-Proj and YALE-Proj as two SPD matrices in the Riemannian symmetric space, where the distance between any two SPD matrices is geodesic. As a consequence, after the parameterization, the distance between the distributions of YALE and UMIST in RKHS is equivalent to the geodesic distance between their corresponding SPD matrices in RSS. Then, DC-RSS will continuously generate new projection matrices by the operations of crossover and mutation in EA until the geodesic distance is minimized.

The face recognition rates versus subspace dimensions on the databases of U2Y and Y2U are presented in Figs. 4 and 5,

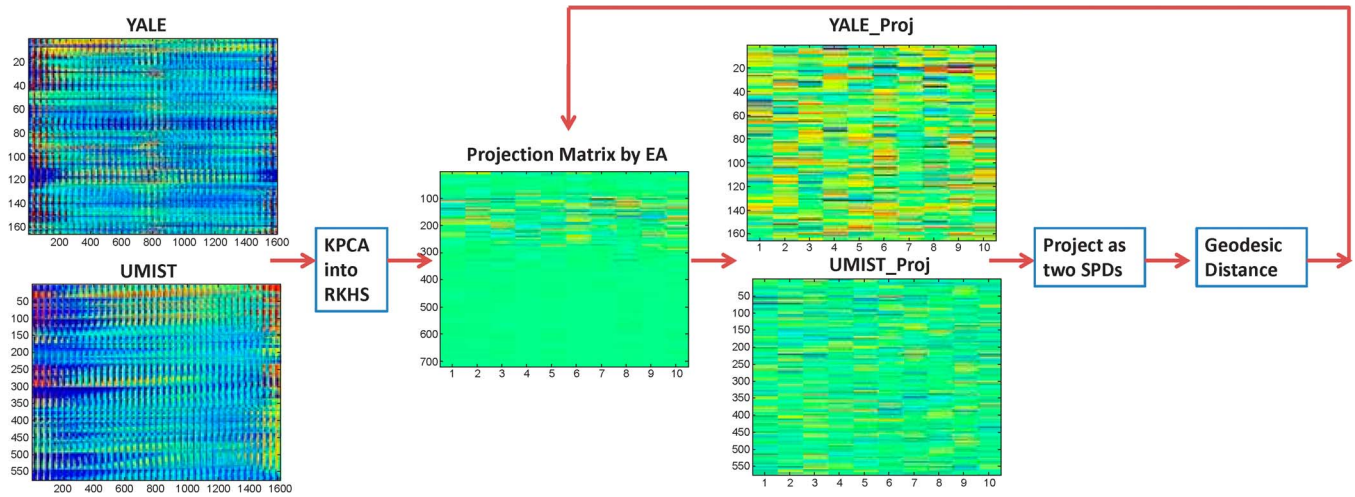


Fig. 3. Flowchart of DC-RSS.

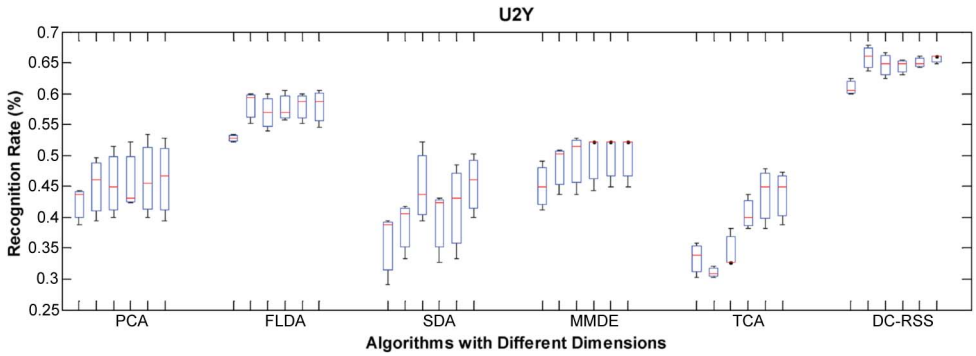


Fig. 4. Recognition rates versus different learning algorithms and subspace dimensions under the U2Y experimental setting.

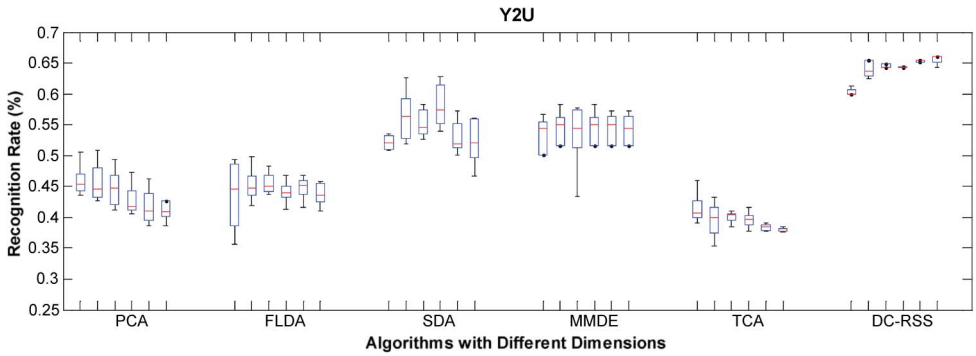


Fig. 5. Recognition rates versus different learning algorithms and subspace dimensions under the Y2U experimental setting.

respectively. Here, we utilize the boxplot to describe comparison results, where each boxplot produces a box and whisker plot for each method, and each box has lines at the lower quartile, median, and upper quartile values. In Figs. 4 and 5, we have six groups, each of which stands for a method, i.e., PCA, FLDA, SDA, TCA, MMDE, and DC-RSS. Each group contains six boxes, where boxes from left to right show the performances of 10, 20, 30, 40, 50, and 60 dimensions, respectively. It is shown that DC-RSS significantly outperforms subspace learning and existing cross-domain learning algorithms. Conventional subspace learning algorithms, e.g., PCA, FLDA, and SDA, cannot work well under the cross-domain setting, because they assume that both the source- and the target-domain samples are independent and identically distributed. MMDE and TCA cannot perform better than DC-RSS, because MMD used in MMDE, and TCA only considers the sample mean bias between the source and target domains, but it fails to measure the covariance difference between the two domains.

TABLE I
EXPERIMENT RESULTS IN THE RECALL RATE OF SIX LEARNING ALGORITHMS FOR TWO CROSS-DOMAIN LEARNING TASKS, I.E., CROSS-DOMAIN TEXT CATEGORIZATION AND CROSS-DOMAIN WEB IMAGE ANNOTATION. THE RESULTS ARE THE AVERAGES OF FIVE RANDOM REPEATS AND THEIR STANDARD DEVIATIONS. THE RESULT IN ITALICS MEANS NEGATIVE CROSS-DOMAIN LEARNING

Data Set	D	PCA	FLDA	SDA	TCA	MMDE	DC-RSS
TEXT	10	0.374±0.032	0.341±0.005	0.359±0.017	0.353±0.026	0.394±0.014	0.459±0.021
	20	0.399±0.029	0.428±0.017	0.411±0.030	0.355±0.024	0.449±0.032	0.497±0.012
	30	0.433±0.022	0.452±0.020	0.440±0.021	0.358±0.021	0.488±0.021	0.536±0.015
	40	0.446±0.028	0.483±0.017	0.494±0.018	0.359±0.022	0.532±0.018	0.555±0.002
	50	0.453±0.013	0.500±0.021	0.505±0.018	0.359±0.022	0.575±0.026	0.586±0.004
	60	0.479±0.036	0.541±0.020	0.535±0.021	0.360 ± 0.022	0.599±0.025	0.605±0.011
NUS-WIDE	10	0.373±0.028	0.292±0.025	0.324±0.025	0.301±0.022	0.327±0.014	0.386±0.021
	20	0.386±0.027	0.317±0.015	0.351±0.008	0.315±0.024	0.358±0.032	0.392±0.012
	30	0.393±0.024	0.344±0.013	0.367±0.007	0.332±0.020	0.404±0.021	*0.392±0.015*
	40	0.397±0.024	0.375±0.007	0.388±0.010	0.345±0.025	0.417±0.018	*0.414±0.002*
	50	0.403±0.030	0.405±0.016	0.406±0.011	0.351±0.029	0.427±0.026	0.446±0.004
	60	0.405±0.024	0.412±0.017	0.428±0.019	0.353±0.022	0.439±0.025	0.446±0.011

Fig. 6. Sample images under the scene concept (including 14 kinds of scenes) from the NUS-WIDE database.

DC-RSS performs consistently and significantly better than the other approaches, because it precisely calibrates the distribution bias and thus can well transfer the useful information from the source domain to the target domain.

B. Cross-Domain Text Categorization

To further examine the effectiveness of the proposed DC-RSS, we compare the proposed DC-RSS with the aforementioned five algorithms, i.e., PCA, FLDA, SDA, MMDE, and TCA, for text categorization on 20 Newsgroups [19]. The 20 Newsgroups data set is very popular for testing document classification algorithms. It contains 18846 documents with 26214 words from 20 topics (classes) of documents. Because some topics are closely related to each other, whereas other topics are not, these 20 topics can be grouped into six subjects. Because some subjects are not suitable for cross-domain learning, we only use four of these subjects (i.e., comp., rec., sci., and talk.) for subsequent experiments. Based on these four subjects, we use the following strategy to generate a new cross-domain learning data set. We randomly select one topic from each subject among four subjects and then select another topic from the remaining topics from each subject for test. For each topic, we randomly select 100 documents.

We apply the similar training and test strategy used in the cross-domain face recognition problem for cross-domain text categorization. Table I shows the experimental results with respect to six algorithms from 10 to 60 dimensions. This table shows that DC-RSS performs best among the six algorithms on the cross-domain text categorization task.

Cross-Domain Web Image Annotation: To demonstrate the effectiveness of DC-RSS for real-world applications, we evaluate the effectiveness of DC-RSS for cross-domain web image annotation on the real-world web image annotation database NUS-WIDE [11]. The NUS-WIDE database contains 269648 labeled web images with 81 categories (classes), and its example web images are shown in Fig. 6. The features used in the experiment for NUS-WIDE are 500-D bag of visual words. Because we require that samples from the source and target domains should share some common properties or nothing useful could be passed from the source domain to the target domain, the subject scene is selected as the main subject for cross-domain learning. In the subject of scene, there are 14 categories, including moon and frost. To test the effectiveness of DC-RSS for the scene data set, we randomly select six kinds of scene for training and use the remaining six kinds for testing (for five times). The test strategy is similar to the approach used in the cross-domain face recognition and text categorization tasks.

Table I compares DC-RSS with PCA, FLDA, SDA, TCA, and MMDE on the NUS-WIDE database under six different dimensions. As shown in this table, we conclude that conventional subspace learning algorithms, e.g., PCA, FLDA, and SDA, are not suitable for the tasks under the cross-domain setting, because they assume that samples for both the source and target domains are drawn from the same distribution. Although

both MMDE and TCA consider the distribution bias between the source and target domains, the metric involved, i.e., MMD, fails to discover the underlying distribution distance between these two domains. Therefore, they cannot work as well as DC-RSS. DC-RSS performs better than other approaches; in other words, useful information can better be transduced from the source domain to the target domain in DC-RSS, because it not only considers the distribution bias but also measures their geodesic distance that reflects the underlying bias.

IV. Conclusion

In this paper, we have studied the problem of distribution calibration for cross-domain setting tasks and proposed a novel cross-domain learning algorithm, termed DC-RSS. DC-RSS can calibrate the geodesic bias between the distributions of the source and target domains through subspace learning. In particular, DC-RSS parameterizes the distribution of the source and target domains in RKHS as two SPD matrices in the Riemannian symmetric space, where the distance between any two SPD matrices is geodesic. As a consequence, after the parameterization, the distance between two distributions in RKHS is equivalent to the geodesic distance between their corresponding SPD matrices in RSS. Then, we search for a subspace, and when all the samples are projected into it, the geodesic distance between the distribution of the source and target domains is minimized. Under this new feature representation, the knowledge from the source domain can be well shared to the target domain. Experiments on cross-domain face recognition, text categorization, and real-world web image annotation show that DC-RSS is effective in calibrating the distributions and outperforms subspace learning and popular cross-domain learning algorithms.

References

[1] A. Andai, "On the geometry of generalized Gaussian distributions," *J. Multivariate Anal.*, vol. 100, no. 4, pp. 777–793, Apr. 2009.

[2] S. Amari, *Differential-Geometrical Methods in Statistics (Lecture Notes in Statistics 28)*. Berlin, Germany: Springer-Verlag, 1990.

[3] K. M. Borgwardt, A. Gretton, M. J. Rasch, H. P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, Jul. 2006.

[4] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[5] W. Bian and D. Tao, "Max–Min distance analysis by using sequential SDP relaxation for dimension reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010, to be published.

[6] D. Cai, X. He, and J. Han, "Semisupervised discriminant analysis," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–7.

[7] J. H. Chen and C. S. Chen, "Reducing SVM classification time using multiple mirror classifiers," *IEEE Trans. Syst., Man, Cybern. B: Cybern.*, vol. 34, no. 2, pp. 1173–1183, Apr. 2004.

[8] L. Cao, J. B. Luo, and T. S. Huang, "Annotating photo collections by label propagation according to multiple similarity cues," in *Proc. ACM Int. Conf. Multimedia*, 2008, pp. 121–130.

[9] L. Cao, J. B. Luo, and T. S. Huang, "Image annotation within the context of personal photo collections using hierarchical event and scene models," *IEEE Trans. Multimedia*, vol. 11, no. 2, pp. 208–219, Feb. 2009.

[10] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997.

[11] T. S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. T. Zheng, "NUS-WIDE: A real-world web image data base from the National University of Singapore," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, pp. 1–9.

[12] L. D. Davis and M. Mitchell, Eds., *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.

[13] L. X. Duan, W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer SVM for video concept detection," in *Proc. Int Conf. Comput. Vision Pattern Recog.*, 2009, pp. 1375–1381.

[14] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, pp. 179–188, 1936.

[15] D. B. Graham and N. M. Allinson, "Characterizing virtual eigensignatures for general purpose face recognition," *Face Recognit.—From Theory Appl. NATO ASI Ser. F, Comput. Syst. Sci.*, vol. 163, pp. 446–456, 1998.

[16] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 7, pp. 498–520, Oct. 1933.

[17] X. He, "Laplacian regularized D-optimal design for active learning and its application to image retrieval," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 254–263, Jan. 2010.

[18] J. Liu, S. Ji, and J. Ye, *SLEP: Sparse Learning With Efficient Projections*. Tempe, AZ: Arizona State Univ., 2009. [Onine]. Available: http://www.public.asu.edu/~jye02/Software/SLEP/index.htm

[19] K. Lang, "Newsweeder: Learning to filter netnews," in *Proc. Int. Conf. Mach. Learn.*, 1995, pp. 331–339.

[20] M. Lovri, M. Min-Ooa, and E. A. Ruh, "Multivariate normal distributions parametrized as a Riemannian symmetric space," *J. Multivariate Anal.*, vol. 74, no. 1, pp. 36–48, Jul. 2000.

[21] S. Lang, Ed., *Fundamentals of Differential Geometry*. New York: Springer-Verlag, 1999.

[22] S. J. Pan, W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," in *Proc. Int. Joint Conf. Artif. Intell.*, 2009, pp. 1187–1192.

[23] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. 23th AAAI Conf. Artif. Intell.*, 2008, pp. 677–682.

[24] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *Int. J. Comput. Vis.*, vol. 66, no. 1, pp. 41–66, Jan. 2006.

[25] Y. Pang, D. Tao, Y. Yuan, and X. Li, "Binary two-dimensional PCA," *IEEE Trans. Syst., Man, Cybern. B*, vol. 38, no. 4, pp. 1176–1180, Aug. 2008.

[26] B. Schölkopf, J. C. Burges, and A. J. Smola, *Kernel Principal Component Analysis—Support Vector Learning*. Cambridge, MA: MIT Press, 1999.

[27] D. Song and D. Tao, "Biologically inspired feature manifold for scene classification," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 174–184, Jan. 2010.

[28] S. Si, D. Tao, and B. Geng, "Bregman divergence based regularization for transfer subspace learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 929–942, Jul. 2010.

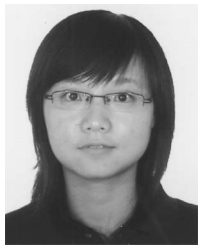
[29] S. Si, D. Tao, and K. P. Chan, "Evolutionary cross-domain discriminative Hessian eigenmaps," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1075–1086, Apr. 2010.

[30] D. Tao, X. Li, X. Wu, and S. J. Maybank, "Geometric mean for subspace selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 260–274, Feb. 2009.

[31] X. Tian, D. Tao, X.-S. Hua, and X. Wu, "Active reranking for web image search," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 805–820, Mar. 2010.

[32] J. P. Ye, "Least squares linear discriminant analysis," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 1087–1093.

[33] T. Zhou, D. Tao, and X. Wu, "Manifold Elastic Net: A unified framework for sparse dimension reduction," *Data Mining Knowl. Discov.*, 2010, DOI: 10.1007/s10618-010-0182-x.



Si Si received the B.Eng. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2008, and the M.Phil. degree from the University of Hong Kong (HKU), Hong Kong, in 2010. She was an exchange student in the School of Computer Engineering, Nanyang Technological University, Singapore, in 2009. She is currently working toward the Ph.D. degree in the Department of Computer Science, The University of Texas at Austin.

Her research interests include data mining and machine learning.

Wei Liu received the B.S. degree from Zhejiang University, Hangzhou, China, in 2001, and the M.E. degree from the Chinese Academy of Sciences, Beijing, China, in 2004. He is currently working toward the Ph.D. degree in the Department of Electrical Engineering, Columbia University, New York, NY.

He was a Research Assistant in the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong, China. He has published more than 30 scientific papers, including ones published in *ICML, KDD, CVPR, ECCV, CHI, ACM Multimedia, IJCAI, AAAI*, and *TOMCCAP*. His research interests include machine learning, computer vision, pattern recognition, data mining, and information retrieval.

Mr. Liu received several meritorious awards from the annual Mathematical Contest in Modeling (MCM) organized by the Consortium for Mathematics and Its Applications (COMAP) during his undergraduate years.

Dacheng Tao (M'07) is currently a Professor in the Centre for Quantum Computation and Information Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia. He mainly applies statistics and mathematics for data analysis problems in data mining, computer vision, machine learning, multimedia, and video surveillance. He has authored or coauthored more than 100 scientific articles at top venues including IEEE T-PAMI, T-KDE, T-IP, NIPS, AISTATS, AAAI, CVPR, ECCV, ICDM; ACM T-KDD, and KDD, with best paper awards.

Kwok-Ping Chan (M'95) received the B.Sc. (Eng.) and Ph.D. degrees from the University of Hong Kong, Hong Kong, China.

He is currently an Associate Professor in the Department of Computer Science, University of Hong Kong. His research interests include Chinese computing, pattern recognition, and machine learning.