Tensor Kernel Supervised Dictionary Learning for Face Recognition

Yeong Khang Lee, Cheng Yaw Low and Andrew Beng Jin Teoh
School of Electrical and Electronic Engineering, Yonsei University
E-mail: leeyeongkhang@yonsei.ac.kr, chengyawlow@yonsei.ac.kr and bjteoh@yonsei.ac.kr

***Abstract*— Sparse-representation is well-known for its promising performance in face recognition task. Recently, researchers have focused on optimizing the dictionary by learning the discriminative sparse model. On the other hand, symmetric positive definite (SPD) matrix descriptor has spurred great interest among computer vision community due to its inherent merits that enables features fusion. However SPD descriptors form a curved-geometry known as Tensor Manifold, which is incompatible to traditional vector-based dictionary learning methods. In order to close the gap between dictionary learning and SPD matrices, this paper proposes Tensor kernel supervised dictionary learning (TKSDL) for face recognition. TKSDL works in such a way by embedding the Tensor manifold into reproducing kernel Hilbert spaces by means of Tensor kernel functions. The discriminative dictionary is then learned by maximizing the Hilbert Schmidt independence criterion (HSIC) that leverages the class labels from the training data. Sparse coefficients are solved independently from the dictionary learning via a soft thresholding mechanism. Extensive experiments on the ORL, AR and FERET datasets are conducted to verify the efficiency of the proposed methods.**

I. INTRODUCTION

Regional covariance matrix (RCM) [1][2] as a feature descriptor is receiving heighten attention for applications such as object tracking and texture classification since it was introduced by Tuzel et al. [1]. RCM is recognized as a powerful descriptor that enables fusion of diverse features. However, direct adoption of RCM for face recognition did not work as anticipated [3]. In order to capture the local spatiality, scale and orientation information of facial image, Pang et al [3] have proposed Gabor-based region covariance matrices (GRCM), which have demonstrated comparable accuracy performance with state of the arts.

GRCM as a Symmetric and Positive Definite (SPD) matrix does not conform to the Euclidean setting but instead characterized by Tensor[1] manifold geometry [4]. GRCM can be seen as a point on the Tensor manifold, in which the shortest distance of arbitrary two points is not connected by a straight line but a geodesic along the curvature of manifold [5]. Therefore, geodesic distances of Tensor manifold should be applied instead of common distance metrics that used in vector-based feature such as L_p norm and inner product.

Sparse representation classifier (SRC) has been shown as a highly promising method for face recognition since the seminal work of Wright et al. [6]. This approach works in such a way by collecting a set of training face images (known as atoms) from different subjects to form a dictionary. The method is capable to represent an unseen face image by means of a weighted combination of a few atoms in the dictionary due to the sparseness of weighting coefficients. The classification is then conducted by labeling the test datum with the atom class label of the least reconstruction error.

In SRC, the dictionary is merely constructed by the training face images. The immediate drawbacks that can be anticipated are the dictionary size will grow indefinitely large when more users enrolled into the system and it is difficult to ensure such a primitive dictionary can contribute to the accuracy performance. Therefore, it is preferable if a discriminative and scalable dictionary can be learned from the training data. Many works have been done for dictionary learning for vector-based data. For example K-SVD, online learning [8], cyclic coordinate descent [9] etc. Generally, there are two limitations for these approaches for the problem of interest in this paper – face recognition on the Tensor manifold via dictionary learning. Firstly, the special structure of GRCM as a SPD matrix discourages direct vectorization of GRCM for dictionary learning. Secondly, in the context of face recognition, the label information should be incorporated into the dictionary but not late until to the error reconstruction stage as in [6].

In order to address the first issue, prior works [10] followed the K-SVD idea by updating the sparse coefficients and dictionary alternatively and iteratively. The authors formulated a dictionary learning problem on the Tensor manifold as a means of LogDet divergence minimization problem. On the other hand, Generalized Dictionary Learning (GDL) [11] formulates the problem to allow online learning. In [12], the authors introduced stein kernel function, which is meant to embed the SPD matrix into the reproducible kernel Hilbert space (RKHS). This approach learns and updates the sparse coefficients and dictionary alternately until the solution converges. However, those methods are generic and do not take the unique challenges of face recognition into account. A closer work to this paper is [13] whereby the authors integrated GRCM and vector-based collaboration representation classifier (CRC) via a Tensor kernel function for face recognition. However, the dictionary is constructed by a collection of plain training facial images as in SRC and

[1] Tensor in this paper is referred to a signal descriptor that characterized by SPD matrices but not the tensors, which is commonly understood as a mathematical object that generalized scalars and vectors.

hence no attempt to optimize them. As for second issue, none of the existing works attempt to incorporate class labels for dictionary learning on the Tensor manifold to the best of our knowledge.

Recently, Gangeh et al [14] introduced a supervised dictionary learning (SDL) scheme for vector-based data by leveraging the Hilbert Schmidt independence criterion (HSIC) to incorporate class labels of the training data into dictionary learning. SDL is free from iterative and alternate sparse coefficients and dictionary update paradigm that commonly adopted presently. Yet, both tasks can be solved deterministically and separately with SDL.

In this paper, we bridge the gap of GRCM and SDL as an attempt to incorporate GRCM, a rich facial descriptor, into dictionary learning based classifier. Our approach leverages kernel learning method to embed the manifold-valued data, ie GRCM onto the Reproducing Kernel Hilbert Space (RKHS). The embedding relies on using kernel functions that compliant with Mercer's theorem in which the kernel functions should be symmetric and positive definite (SPD). Therefore, we adapt a Gaussian radial basis function (RBF), which is inherently SPD [15]. However, Euclidean distance in the native Gaussian RBF is far from adequate to account the inherent manifold geometry and hence it is replaced with Tensor manifold geodesic distances or its re-parameterized dissimilarity measures. Several geodesic distances are examined including Affine Invariant Riemannian Metric (AIRM), Log-Euclidean Riemann Metric (LERM), Cholesky distance and Point Restricted Modified Hausdorff Distance (PRMHD).

In a nutshell, the contribution of this paper is three-fold: (1) we link the GRCM and SDL by embedding the GRCM into RKHS. Unlike current efforts of integrating SPD matrix/GRCM and representation classifier/dictionary learning, our method is of supervised by incorporating class labels for dictionary learning and hence render better accuracy performance. The learning is free from conventional alternate update of dictionary and sparse coefficients but both can be solved separately and deterministically thanks to SDL. (2) We study and compare the four distances measurement on Tensor manifold for Tensor based kernel function construction in terms of the accuracy performance and computation efficiency. (3) The proposed method demonstrated a generic means for how to integrate the manifold-valued data such as SPD matrix with vector-based learning models that can be kernelized.

II. PRELIMINARY

In this section, we provide some background knowledge on GRCM and its associate geodesic and re-parameterized distance measures. In addition, supervised dictionary learning (SDL) with Hilbert-Schmidt independence criterion (HSIC) as (in)dependency measure is also given.

A. Gabor-based Region Covariance Matrix (GRCM)

Gabor kernel is the product of elliptical Gaussian and a complex plane wave as defined in [16]:

$$\Psi_{u,v}(z) = \frac{||k_{u,v}||^2}{\sigma^2} e^{\frac{||k_{u,v}||^2||z||^2}{2\sigma^2}} \left(e^{ik_{u,v}z} - e^{-\sigma^2/2}\right) \tag{1}$$

such that $z = (x, y)$ denote pixel location, u and v are the scale and orientation of the Gabor kernel. The wave vector $k_{u,v}$ is defined as $k_{u,v} = k_v e^{i\phi_u}$ where $k_v = k_{max}/f_v$ and $\phi_u = \pi u/8$. In this paper we use 40 Gabor filters that tune with 5 scales and 8 orientations to form a filter bank.

Consider each pixel on image $\boldsymbol{F}$ as a node, an image with size $h \times w$ has a total number of nodes $N = hw$. Gabor features for each node can be sought by convoluting face image with the filter bank.

$$g_{uv}(x, y) = \left|\boldsymbol{F}(x, y) * \Psi_{u,v}(x, y)\right| \tag{2}$$

Consecutively, the node is constructed using feature mapping function as follow:

$$\boldsymbol{z}_i = [x\ y\ g_{00}(x, y)\ g_{01}(x, y) \ldots g_{74}(x, y)]^{\mathrm{T}} \tag{3}$$

where $\boldsymbol{z}_i \in \mathbb{R}^{42}$.Substituting Eq.(3) into Eq.(4), an image can be presented as a covariance matrix $\boldsymbol{G} \in \mathbb{R}^{42\times42}$:

$$\boldsymbol{G} = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{z}_i - \boldsymbol{u})(\boldsymbol{z}_i - \boldsymbol{u})^{\mathrm{T}} \tag{4}$$

Where $\boldsymbol{u}$ is the mean of $\boldsymbol{z_i}$

Similar with [3], each face image is segmented into 5 different parts – whole face, left, right, half up and half bottom, and thus correspond to five Gabor-based covariance matrices $\{\boldsymbol{G}^k \in \mathbb{R}^{42\times42} | k = 1, \ldots, 5\}$, which is collectively named as *Gabor-based region covariance matrix (GRCM) set*. The dissimilarity measure of two GRCM sets can be measured using:

$$d_T(G, P) = \sum_{i=1}^{5} p\left(\boldsymbol{G}_G^i, \boldsymbol{G}_P^i\right) - \max_j\left(p\left(\boldsymbol{G}_G^j, \boldsymbol{G}_P^j\right)\right) \tag{5}$$

Here, $p(\cdot,\cdot)$ is a distance measure for SPD matrices that will be discussed in next section.

B. Metric on Riemannian Manifold

The positive definiteness structure of GRCM allows it to be perceived as a point on differentiable Tensor manifold, $\boldsymbol{G} \in \mathcal{M}$ [17]. Differentiable manifold, $\mathcal{M}$ is a topological space that is locally resembled Euclidean space and has a globally defined differential structure. Under such circumstances, geodesic distance induced by Riemannian metric is a better measure for distance between two points.

There are several geodesic distances[18][19] and re-parameterized distance [20][21] measures can be used to measure the similarity of two SPD matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathcal{M}$. We have chosen four distances as follows:

1. *Affine Invariant Riemannian Metric* (AIRM)

$$p_A(\boldsymbol{A}, \boldsymbol{B}) = \sqrt{\sum_{i=1}^{d} \ln^2 \lambda_i(\boldsymbol{A}, \mathbf{B})} \tag{6}$$

where λ_i are the eigenvalues that can be solved through $\lambda_i \boldsymbol{Au} = \boldsymbol{Bu}$, $i = 1, \ldots, d$.

2. *Log-Euclidean Riemann Metric* (LERM)

$$p_L(\boldsymbol{A}, \mathbf{B}) = ||\,\mathrm{Log}(\boldsymbol{A}) - \mathrm{Log}(\boldsymbol{B})\,||_F \tag{7}$$

where $\mathrm{Log}(\cdot)$ is the matrix logarithm. LERM is closely related to AIRM and it is also a true geodesic distance.

3. *Cholesky distance* (CHOL)

$$p_C(\boldsymbol{A}, \boldsymbol{B}) = ||L_A - L_B||_F \tag{8}$$

CHOL is a re-parameterization measure that decomposes the matrices as $\boldsymbol{A} = L_A L_A^{\mathrm{T}}$ and $\boldsymbol{B} = L_B L_B^{\mathrm{T}}$.

4. *Point Restricted Modified Hausdorff Distance* (PRMHD)

Similar to CHOL, one can re-parameterize Tensor data into sigma set [20] by using Cholesky factorization $\boldsymbol{A} = L_A L_A^T$. Consider $\boldsymbol{l}_i^A$ as the i^{th} column of factorized matrix L_A, each column in factorized matrix is called sigma point, hence they are known as sigma set collectively. Then the measurement between two sigma sets is measured as:

$$p_M(\boldsymbol{A}, \boldsymbol{B}) = \frac{1}{d}\sum_{k=1}^{d} d_E(\boldsymbol{l}_k^A, \boldsymbol{l}_k^B) \tag{9}$$

where $d_E(\cdot,\cdot)$ is any common distance metric defined in Euclidean space such as Euclidean distance and d is the size of GRCM where d = 42 for this paper.

C. Supervised Dictionary Learning

Consider a reproducing kernel Hilbert space $\mathcal{F}$, there exist a mapping function $\phi: \mathcal{X} \to \mathcal{F}$, thus each point $x \in \mathcal{X}$ is mapped into Hilbert space such that $k(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{F}}$, $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Similarly, another RKHS, $\mathcal{H}$ is defined for space $\mathcal{Y}$ with a mapping function $\psi: \mathcal{Y} \to \mathcal{H}$ such that $l(\cdot,\cdot) = \langle \psi(\cdot), \psi(\cdot) \rangle_{\mathcal{H}}$.

When measuring the dependency between n random variables, $\boldsymbol{z} = \{(\boldsymbol{x}_1, y_1) \ldots (\boldsymbol{x}_n, y_n)\} \subseteq \mathcal{X} \times \mathcal{Y}$ via HSIC, the estimation is defined as follows:

$$HSIC(\boldsymbol{z}) = (n-1)^{-2} tr(\boldsymbol{KHLH}) \tag{9}$$

where $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ with entries $k_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\boldsymbol{L} \in \mathbb{R}^{n \times n}$ with $l_{ij} = l(y_i, y_j)$ and centering matrix, $\boldsymbol{H} = \boldsymbol{I} - n^{-1}\boldsymbol{e}\boldsymbol{e}^T \in \mathbb{R}^{n \times n}$ with $\boldsymbol{e}$ is a n-dimension ones vector. In the context of face recognition, $\boldsymbol{x}$ represents a face vector while y is its corresponding class label. In this case, $tr(\boldsymbol{KHLH})$ should be maximized to increase the dependency of x and y.

When formulating SDL, given n d-dimension training vectors that packed into a matrix, $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$, the dictionary learning problem can be formulated based on minimum error reconstruction principle as:

$$\min_{\boldsymbol{D},\boldsymbol{A}} \|\boldsymbol{X} - \boldsymbol{DA}\|_F^2 \tag{10}$$

where $\boldsymbol{D} \in \mathbb{R}^{d \times b}$ is the dictionary, $\boldsymbol{A} \in \mathbb{R}^{b \times n}$ is a reconstruction coefficients matrix and b is the number of atoms in $\boldsymbol{D}$. Since the problem in Eq.(10) is ill-posed, a constraint can be imposed such that $\boldsymbol{D}$ is orthonormal. With manipulations and transformation shown in [14], Eq. (10) can be reformulated to:

$$\max_{\boldsymbol{U}} tr(\boldsymbol{D}^{\mathrm{T}} \boldsymbol{XHLHX}^{\mathrm{T}} \boldsymbol{D}) \quad \text{s.t } \boldsymbol{D}^{\mathrm{T}}\boldsymbol{D} = \boldsymbol{I} \tag{11}$$

where $\boldsymbol{L} = \boldsymbol{Y}^{\mathrm{T}}\boldsymbol{Y} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{Y} \in \{0,1\}^{c \times n}$ is a label matrix of c-class training data in which each vector $\boldsymbol{y}_i$ in $\boldsymbol{Y}$ contains none-zero elements on the position corresponds to its label. Under such transformation, the correlation of the data in the same class is maximized and zero correlation for other class. Let

$$\boldsymbol{\Phi} = \boldsymbol{XHLHX}^T \tag{12}$$

Eq.(11) can be solved according to Rayleigh-Ritz theorem. The optimized $\boldsymbol{D}$ can be obtained by solving the $\boldsymbol{\Phi v}=\boldsymbol{v}\boldsymbol{\lambda}$ and $\boldsymbol{D} \in \mathbb{R}^{d \times b}$ is equivalent to the top b eigenvectors $\boldsymbol{v}$ of $\boldsymbol{\Phi}$.

III. TENSOR KERNEL SUPERVISED DICTIONARY LEARNING (TKSDL)

Reproducing Kernel Hilbert Space (RKHS) embedding has been widely studied and proved versatile in machine learning and computer vision to explore structure data beyond Hilbert space [22]. RKHS embedding allows us to generalize the vector-based learning models to differentiable manifolds such as Tensor manifold [17] and Grassmann manifold [23]. The procedure involves embedding $\mathcal{X} \in \mathcal{M}$ into Hilbert space as a point through a non-linear function, $\phi(\cdot)$. A kernel function $k: \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is used to define the inner product on the Hilbert space, thus forming a RKHS. However Mercer's theorem dictates that only symmetric and positive definite (SPD) kernels delineate valid RKHS.

Gaussian RBF kernel is always a popular kernel function for RKHS embedding attributed to its inherent SPD property. However, Euclidean distance in the native Gaussian RBF is inadequate to account the Tensor manifold geometry and hence it should be replaced with geodesic distances or re-parameterized dissimilarity measures of Tensor manifold described in Section IIB. It is shown by [17] that such a substitution would not alter SPD property of the Gaussian RBF. In other words, a valid (SPD) Tensor kernel function can be induced through:

$$k(\mathcal{X}_i, \mathcal{X}_j) = exp(-\gamma^{-1} d_T(\mathcal{X}_i, \mathcal{X}_j)^2) \tag{13}$$

where $d_T(\cdot)$ is the GRCMs set dissimilarity measure given in Eq.(5).

For SDL kernelization, a manifold data, $\mathcal{X}$ (GRCM in this paper) is first mapped into a feature space, $\mathcal{F}$ via a non-linear mapping function $\phi(\cdot)$ and an objective function following the idea depicted in Eq (11) can be formed as follows:

$$tr(\boldsymbol{P}^{\mathrm{T}}\phi(\mathcal{X})\boldsymbol{HLH}\phi(\mathcal{X})^{\mathrm{T}}\boldsymbol{P}) \tag{14}$$

With representation theory [24], $\boldsymbol{P}$ is perceived as a linear combination of $\phi(\mathcal{X})$ or $\boldsymbol{P} = \boldsymbol{Q}\phi(\mathcal{X})$ where $\boldsymbol{Q}$ is a representation of $\boldsymbol{P}$ in $\mathcal{F}$. By replacing the projection matrix $\boldsymbol{P}$ = $\boldsymbol{Q}\phi(\mathcal{X})$ to Eq.(14), we obtain:

$$\begin{aligned} &tr(\boldsymbol{Q}^{\mathrm{T}}\phi(\mathcal{X})^{\mathrm{T}}\phi(\mathcal{X})\boldsymbol{HLH}\phi(\mathcal{X})^{\mathrm{T}}\phi(\mathcal{X})\boldsymbol{Q}) \\ &\quad = tr(\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{KHLHKQ}) \end{aligned} \tag{15}$$

Along with constraint in Eq. (11), we obtain

$$\begin{aligned} \boldsymbol{P}^{\mathrm{T}}\boldsymbol{P} &= \boldsymbol{Q}^{\mathrm{T}}\phi(\mathcal{X})^{\mathrm{T}}\phi(\mathcal{X})\boldsymbol{Q} \\ &= \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{KQ} \end{aligned} \tag{16}$$

where $\boldsymbol{K} \in \mathbb{R}^{n \times n}$ is a kernel matrix with entries k_{ij} = $\phi(\mathcal{X})^{\mathrm{T}}\phi(\mathcal{X})$. Note that k_{ij} can be explicitly computed with Eq.(13). The optimization problem for Tensor kernel SDL is hence given as:

$$\max_{\boldsymbol{Q}} tr(\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{KHLHKQ}) \quad \text{s.t } \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{KQ} = \boldsymbol{I} \tag{17}$$

Similar to Eq.(11), optimized $\boldsymbol{Q} \in \mathbb{R}^{n \times b}$ can be acquired by solving the eigenvalue decomposition problem, $\boldsymbol{\Phi v} = \lambda \boldsymbol{K v}$ where

$$\boldsymbol{\Phi} = \boldsymbol{KHLHK} \tag{18}$$

and $\boldsymbol{Q} \in \mathbb{R}^{n\times b}$ is the top b eigenvectors $\boldsymbol{v}$ of $\boldsymbol{\Phi}$. Note that the size of $\boldsymbol{Q}$ is limited by the number of training data, n and b that need to be empirically determined from the experiments.

IV. CLASSIFICATION ON LEARNED DICTIONARY

In this section, we first introduce how sparse coefficients can be solved given optimized dictionary $\boldsymbol{D}$ described in Section II-C and the extension to TKSDL. Finally, computed coefficients can be used as inputs to a classifier.

Recall Eq.(10), $\boldsymbol{A}$ is the coefficient matrix and substitute the optimized dictionary $\boldsymbol{D}$ to Eq.(10), along with the constraint in Eq.(11), the coefficients can be computed as:

$$\boldsymbol{A} = \boldsymbol{D}^{\mathrm{T}}\boldsymbol{X} \tag{19}$$

In order to impose sparse constraint on the coefficients, the problem can be formulated as :

$$\min_{\boldsymbol{a}_i}\left(\frac{1}{2}\|\boldsymbol{D}^T\boldsymbol{x_i} - \boldsymbol{a}_i\|_2^2 + \lambda\|\boldsymbol{a}_i\|_1\right) \tag{20}$$

where $\boldsymbol{x}_i$ is the i^{th} training data sample and $a \in \mathbb{R}^m$ where m is the total number of atoms in $\boldsymbol{D}$. Interestingly, Eq.(20) admits a closed form solution by using a soft-thesholding operator [25]:

$$\boldsymbol{a}_i = S_\lambda(\boldsymbol{D}^{\mathrm{T}}\boldsymbol{x}_i) \tag{21}$$

where $S_\lambda(\cdot)$ is defined as:

$$S_\lambda(\boldsymbol{m})\begin{cases}\boldsymbol{m} - 0.5\lambda & \text{if } \boldsymbol{m} > \ \ 0.5\lambda \\ \boldsymbol{m} + 0.5\lambda & \text{if } \boldsymbol{m} < -0.5\lambda \\ 0 & \text{otherwise}\end{cases} \tag{22}$$

For TKSDL, the coefficient can be computed as follows:

$$\begin{aligned}\boldsymbol{A}' &= \boldsymbol{Q}^{\mathrm{T}}\,\phi(\mathcal{X})^{\mathrm{T}}\phi(\mathcal{X}) \\ &= \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{K}\end{aligned} \tag{24}$$

Following same transformation from Eq.(19) and Eq.(20)

$$\boldsymbol{a}_i = S_\lambda(\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{k}_i) \tag{25}$$

where $\boldsymbol{k}_i \in \mathbb{R}^n$, n is the number of training sample. Algorithm 1 detailed the complete progression of entire TKSDL.

Finally, sparse coefficients $\boldsymbol{a}_{tr_i}$ and $\boldsymbol{a}_{tt}$ computed from training data and testing data, respectively can be as input to any classifier such as Support Vector Machine or Extreme Learning Machine (ELM) [26] for classification.

Algorithm 1 : Tensor Kernel SDL (TKSDL)

Input:
Test data: GRCM set, $\boldsymbol{G}_t$
Training data: $\boldsymbol{G} = \{\boldsymbol{G}_1 \ ... \ \boldsymbol{G}_n\}$ with each $\boldsymbol{G}_i$ as GRCM set.
Soft threshold value :λ

Output:
$\boldsymbol{Q}$
Coefficient for train data : $\boldsymbol{a}_{tr}$
Coefficient for test data : $\boldsymbol{a}_{tt}$

Start:
1. Compute a kernel matrix based on training GRCM set, $\boldsymbol{K}_{tr}$ with matrix entry, $k_{ij} = k(\boldsymbol{G}_i, \boldsymbol{G}_j)$
2. Compute a kernel vector based on testing GRCM set, $\boldsymbol{k}_{tt} = [k(\boldsymbol{G}_1, \boldsymbol{G}_t), \ldots, k(\boldsymbol{G}_{Nn}, \boldsymbol{G}_t)]$.
3. Compute
$\boldsymbol{H} = \boldsymbol{I} - n^{-1}\boldsymbol{e}\boldsymbol{e}^T$,
$\boldsymbol{L}$,
$\boldsymbol{\Phi} = \boldsymbol{K}_{tr}\boldsymbol{H}\boldsymbol{L}\boldsymbol{H}\boldsymbol{K}_{tr}^T$.
3. Perform eigenvalue decomposition on $\boldsymbol{\Phi}$ for $\boldsymbol{Q}$
4. Compute train coefficient using Eq.(21)
$$\boldsymbol{a}_{tr_i} = S_\lambda(\boldsymbol{Q}^T\boldsymbol{k}_{tr_i})$$
5. Compute test coefficient using Eq.(21)
$$\boldsymbol{a}_{tt} = S_\lambda(\boldsymbol{Q}^T\boldsymbol{k}_{tt})$$

V. EXPERIMENT AND DISCUSSION

The experiments are conducted on three face databases: ORL[27], FERRET [28] and AR [29]. The ORL database contain 40 subjects and 10 image for each subject and FERET database is a subset of original FERET that consists of 292 subjects with 11 images each. AR dataset contain only 100 subjects and 14 non-occluded images per subject.

Table 1 Parameters used in each database

Database	Distance	Parameter			
		γ	b	h	λ
ORL	AIRM	490	120	100	0
	LERM	8000			
	CHOL	410			
	PRMHD	2000			
	FROB	20000			
FERET	AIRM	50	846	200	0
	LERM	40000			
	CHOL	20			
	PRMHD	350			
	FROB	15000			
AR	AIRM	4000	300	700	0
	LERM	6000			
	CHOL	600			
	PRMHD	400			
	FROB	20000			

In the experiment, the databases are partitioned into 2 non-overlapping training and testing sets. Training image is randomly selected from size of 1 to 5 and the multiple-fold cross-validation is subsequently performed. The classification is performed with ELM using h number of Gaussian RBF hidden neurons. Classification with ELM is performed 5 times and average correct recognition rate (CRR) is taken. The total computation time for training and testing is also recorded. The proposed method is compared with SRC [6], KSRC [30], CRC [31], KCRC[32] and Joint Tensor Kernel Collaborative Representation-based classification (TKCRC) [13]. Kernel based on Frobenius distance (FROB), $p_F(\boldsymbol{A}, \mathbf{B}) = ||\boldsymbol{A} - \boldsymbol{B}||_F$ is included as a baseline. It should be noted that FROB disregards curvature in Tensor manifold instead just calculate the distance in a straight line. Table 1 tabulates various parameters used in this work where γ is from Eq.(13), b from Eq.(18) and λ is from Eq. (22).

The parameters of Gabor kernel are set as $k_{max} = \pi/2$, $f_v = \sqrt{2}$ and $\sigma = 2\pi$ [16]. On the other hand, the Gabor kernel size is prefixed to 21 x 21, 25 x 24, 26 x 26 for ORL, FERET and AR databases, respectively.

Table 2 Recognition Rate on ORL

Methods	**Training Size 1**	**2**	**3**	**4**	**5**
SRC[6]	65.83	79.67	87.06	91.12	93.68
KSRC[30]	69.28	80.24	85.19	88.39	90.76
CRC[31]	64.66	78.75	87.31	91.76	94.01
KCRC[32]	74.25	86.19	91.12	93.87	95.44
TKCRC (Strategy 1)[13]					
AIRM	84.57	93.13	96.36	97.90	98.64
LERM	84.75	93.12	96.21	97.72	98.48
CHOL	83.27	92.52	96.24	98.02	98.09
FROB	69.13	83.25	89.72	93.15	95.11
PRMHD	85.86	93.84	97.01	98.59	99.32
TKCRC (Strategy 2)[13]					
AIRM	85.61	93.26	96.38	97.77	98.52
LERM	84.72	92.93	96.00	97.60	98.40
CHOL	85.19	93.09	96.35	97.93	98.40
FROB	69.22	82.25	88.94	92.67	95.00
PRMHD	84.64	92.17	94.71	95.62	95.58
TKSDL					
AIRM	80.61	88.23	93.55	97.31	98.29
LERM	81.00	91.91	94.87	96.90	97.64
CHOL	86.43	93.88	97.00	98.40	99.16
FROB	70.89	83.88	89.55	91.67	94.07
PRMHD	**88.52**	**95.62**	**98.01**	**98.94**	**99.48**

Table 2 shows that TKSDL performs consistently better than all the vector-based representation based methods across all training sizes especially when the training size is low in ORL database. TKSDL has consistently outperformed every other methods especially when using PRMHD distance measure. Performance on every distance measure does not have huge differences with the exception of FROB. Since FROB does not consider the curvature of Tensor manifold on distance calculation, the poor performance is expected.

Table 3 Recognition Rate on FERET

Methods	**Training Size 1**	**2**	**3**	**4**	**5**
SRC[6]	33.59	45.27	52.75	58.10	62.33
KSRC[30]	33.86	45.52	51.52	54.15	55.29
CRC[31]	39.37	61.99	74.74	82.15	87.17
KCRC[32]	45.00	63.34	73.43	79.99	84.55
TKCRC (Strategy 1)[13]					
AIRM	77.26	88.18	92.69	95.00	96.39
LERM	79.28	90.53	93.86	**96.43**	97.22
CHOL	76.46	88.18	92.88	95.24	96.62
FROB	65.23	79.23	**96.06**	89.94	92.43
PRMHD	79.78	90.15	94.21	96.17	97.30
TKCRC (Strategy 2)[13]					
AIRM	78.49	89.36	93.35	95.55	96.80
LERM	78.40	90.40	93.80	96.30	97.00
CHOL	79.15	90.28	94.30	96.26	**97.39**
FROB	63.97	78.65	85.58	89.55	92.21
PRMHD	73.99	78.43	82.85	87.08	90.19
TKSDL					
AIRM	26.64	85.18	93.78	90.83	77.23
LERM	76.01	90.06	89.04	90.76	95.98
CHOL	74.51	87.61	92.23	94.49	95.20
FROB	66.36	81.00	86.47	80.96	85.61
PRMHD	**81.48**	**90.88**	93.85	95.64	86.34

In Table 3 and Table 4 that correspond to the experiment results for FERET and AR databases, TKSDL recorded the best performance in small training size using the PRMHD distance measure while TKCRC performs better when training size is larger than two. It is interesting to note that the performance of TKSDL with five training samples has dropped severely on AIRM and PRMHD. This might be due to ELM failing to classify the data.

Overall, TKSDL has better performance when the training sample size is small, ie. 1 and 2. However, when training size exceeds three, TKSDL does not have clear advantage compared to KCRC. In the experiments, we disclose that FROB performs poorly over other distance measure. This further solidifies the idea that geometry of Tensor manifold is the key contributor on determining good recognition performance.

Table 4 Recognition Rate on AR

Methods	**Training Size 1**	**2**	**3**	**4**	**5**
SRC[6]	54.15	66.20	64.00	72.52	75.04
KSRC[30]	50.12	59.91	67.38	72.75	76.72
CRC[31]	58.23	76.21	79.99	90.01	91.40
KCRC[32]	57.09	77.22	85.21	91.37	93.07
TKCRC (Strategy 1)[13]					

AIRM	61.95	79.47	87.70	92.07	94.65
LERM	60.21	75.73	83.77	88.51	91.53
CHOL	62.58	80.54	88.36	92.54	94.98
FROB	41.32	55.01	63.46	69.32	73.62
PRMHD	66.38	83.20	90.30	93.83	95.84
TKCRC (Strategy 2)[13]					
AIRM	66.51	83.53	90.33	93.65	95.52
LERM	63.46	81.31	88.91	92.79	95.00
CHOL	66.56	84.33	91.15	94.40	96.20
FROB	40.79	58.95	69.40	75.99	80.53
PRMHD	68.48	85.57	92.11	**95.07**	**96.64**
TKSDL					
AIRM	68.13	87.31	90.98	92.49	93.18
LERM	60.28	79.17	83.08	88.22	91.08
CHOL	69.28	88.00	91.23	92.14	91.30
FROB	41.58	62.38	66.25	68.42	70.51
PRMHD	**71.19**	**89.14**	**92.48**	94.07	95.03

Figure 1 summarizes the average computation time in seconds, necessitated for each distance metrics tested on entire selected FERET database. Ranking computation speed from the fastest to the slowest is; FROB, CHOL, PRMHD, AIRM and LERM. Since FROB is a naïve distance measure implementation, speed performance can be disregarded. Note that AIRM and LERM are true geodesic distances which take the geometry of manifold into account and hence resulting in slower computation time. LERM is especially slow due to the requirement for performing matrix logarithm operation. PRMHD and CHOL re-parameterize GRCM into another form and this allows faster distance computation. The simple calculation of coefficients with soft-thresholding allows TKSDL to have better computation time.

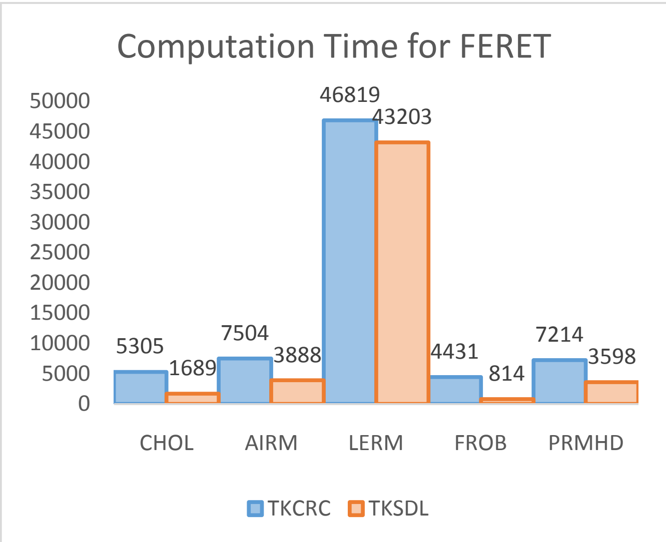

Figure 1 Computation time (seconds) on FERET

VI. Conclusions

In this paper, we introduced an efficacious way to remedy the gap between GRCM descriptor and vector-based supervised dictionary by means of Reproducible Kernel Hilbert Space (RKHS) embedding approach. The subject label information is utilized to obtain a discriminative dictionary through HISC. The experimental results show that our proposed method remarkably outperforms the unsupervised state of the arts representation classifiers on ORL, FERET and AR databases. In addition to that, our proposed method is also computationally inexpensive. We noted that TKSDL yields the best performance in most of the empirical settings when it is paired with the PRMHD distance.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Science, ICT, and Future Planning (2013006574).

References

[1] O. Tuzel, F. Porikli, and P. Meer, “Region Covariance: A Fast Descriptor for Detection and Classification,” in *Proceedings of the 9th European Conference on Computer Vision - Volume Part II*, Berlin, Heidelberg, 2006, pp. 589–600.

[2] D. Tosato, M. Spera, M. Cristani, and V. Murino, “Characterizing Humans on Riemannian Manifolds,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1972–1984, 2013.

[3] Y. Pang, Y. Yuan, and X. Li, “Gabor-Based Region Covariance Matrices for Face Recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 7, pp. 989–993, 2008.

[4] S. Lang, *Fundamentals of Differential Geometry*, vol. 191. New York, NY: Springer New York, 1999.

[5] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos, “Jensen-Bregman LogDet divergence with application to efficient similarity search for covariance matrices,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2161–2174, Sep. 2013.

[6] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust Face Recognition via Sparse Representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[7] M. Aharon, M. Elad, and A. Bruckstein, “-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation,” *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[8] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Online Learning for Matrix Factorization and Sparse Coding,” *J Mach Learn Res*, vol. 11, pp. 19–60, Mar. 2010.

[9] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization Paths for Generalized Linear Models via Coordinate Descent," *J. Stat. Softw.*, vol. 33, no. 1, pp. 1–22, 2010.

[10] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos, "Positive definite dictionary learning for region covariances," in *2011 IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 1013–1019.

[11] S. Sra and A. Cherian, "Generalized Dictionary Learning for Symmetric Positive Definite Matrices with Application to Nearest Neighbor Retrieval," in *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III*, Berlin, Heidelberg, 2011, pp. 318–332.

[12] M. T. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell, "Sparse Coding and Dictionary Learning for Symmetric Positive Definite Matrices: A Kernel Approach," in *Computer Vision – ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Springer Berlin Heidelberg, 2012, pp. 216–229.

[13] Y. K. Lee, A. B. J. Teoh, and K.-A. Toh, "Joint kernel collaborative representation on Tensor manifold for face recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 6245–6249.

[14] M. J. Gangeh, A. Ghodsi, and M. S. Kamel, "Kernelized Supervised Dictionary Learning," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4753–4767, Oct. 2013.

[15] G. E. Fasshauer, *Positive Definite Kernels: Past, Present and Future*. .

[16] T. S. Lee, "Image Representation Using 2D Gabor Wavelets," *IEEE Trans Pattern Anal Mach Intell*, vol. 18, no. 10, pp. 959–971, Oct. 1996.

[17] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, "Kernel Methods on the Riemannian Manifold of Symmetric Positive Definite Matrices," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 73–80.

[18] 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.

[19] X. Pennec, "Statistical Computing on Manifolds: From Riemannian Geometry to Computational Anatomy," in *Emerging Trends in Visual Computing*, F. Nielsen, Ed. Springer Berlin Heidelberg, 2009, pp. 347–386.

[20] X. Hong, H. Chang, S. Shan, X. Chen, and W. Gao, "Sigma Set: A small second order statistical region descriptor," in *IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009*, 2009, pp. 1802–1809.

[21] A. K. Ian L. Dryden, "Non-Euclidean statistics for covariance matrices, with applications to diffusion tensor imaging," 2009.

[22] "Kernel Methods for Pattern Analysis," *Cambridge University Press*. [Online]. Available: http://www.cambridge.org/ve/academic/subjects/computer-science/pattern-recognition-and-machine-learning/kernel-methods-pattern-analysis. [Accessed: 26-May-2015].

[23] C. Tee, M. K. O. Goh, and A. B. J. Teoh, "Gait recognition using Sparse Grassmannian Locality Preserving Discriminant Analysis," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 2989–2993.

[24] J. L. Alperin, *Local Representation Theory: Modular Representations as an Introduction to the Local Representation Theory of Finite Groups*. Cambridge: Cambridge University Press, 1986.

[25] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Am. Stat. Assoc.*, pp. 1200–1224, 1995.

[26] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *2004 IEEE International Joint Conference on Neural Networks, 2004. Proceedings*, 2004, vol. 2, pp. 985–990 vol.2.

[27] F. S. Samaria and A. C. Harter, "Parameterisation of a stochastic model for human face identification," in *, Proceedings of the Second IEEE Workshop on Applications of Computer Vision, 1994*, 1994, pp. 138–142.

[28] P. J. Phillips, H. Moon, S. . Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[29] A. Martinez and R. Benavente, "The AR face database," 1998, p. 24.

[30] L. Zhang, W.-D. Zhou, P.-C. Chang, J. Liu, Z. Yan, T. Wang, and F.-Z. Li, "Kernel Sparse Representation-Based Classifier," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1684–1695, Apr. 2012.

[31] L. Zhang, M. Yang, and X. Feng, "Sparse Representation or Collaborative Representation: Which Helps Face Recognition?," in *Proceedings of the 2011 International Conference on Computer Vision*, Washington, DC, USA, 2011, pp. 471–478.

[32] B. Wang, W. Li, N. Poh, and Q. Liao, "Kernel collaborative representation-based classifier for face recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 2877–2881.