Low Rank Representation on Riemannian Manifold of Symmetric Positive Definite Matrices

Yifan Fu* Junbin Gao* Xia Hong† David Tien*

Abstract

Sparse coding aims to find a more compact representation based on a set of dictionary atoms. A well-known technique looking at 2D sparsity is the low rank representation (LRR). However, in many computer vision applications, data often originate from a manifold, which is equipped with some Riemannian geometry. In this case, the existing LRR becomes inappropriate for modeling and incorporating the intrinsic geometry of the manifold that is potentially important and critical to applications. In this paper, we generalize the LRR over the Euclidean space to the LRR model over a specific Rimannian manifold—the manifold of symmetric positive matrices (SPD). Experiments on several computer vision datasets showcase its noise robustness and superior performance on classification and segmentation compared with state-of-the-art approaches.

Keywords: Riemannian manifold, Symmetric Positive Definite, LRR, Feature Extraction

1 Introduction

Low rank representation (LRR) [1] is an effective method to explore the intrinsic low rank structures embedded in a data set in high dimensional space, which has been successfully used in many applications, such as motion segmentation [2], image segmentation [3] and salient object detection [4]. Generally, given a collection of data points $\mathcal{X} = \{X_1, \cdots, X_n\}$, LRR seeks a joint low rank representation of $\mathcal{X}$ using data points themselves as the dictionary, which can be formulated as,

$$\min_{W} \|\mathcal{X} - \mathcal{X}W\|^2 + \lambda\|W\|_\star \tag{1.1}$$

where $W = \begin{pmatrix} \mathbf{w}_1 \\ \vdots \\ \mathbf{w}_n \end{pmatrix}$ is the low rank representation matrix under the dictionary $\mathcal{X}$, each row $\mathbf{w}_i$ $(i = 1, \ldots, n)$ is the corresponding representation of X_i, $\|\cdot\|_\star$ is the nuclear norm, which is defined as the sum of all the singular values of a matrix, and $\lambda > 0$ is a penalty parameter to balance the nuclear norm term and the reconstruction error.

However, for many applications, such as those in machine learning [5], computer vision [6] and medical image analysis [7], data are not characterized by simple vector features. Taking image data as an example, the raw pixel features , such as color, gradient and filter responses are not robust in the presence of illumination changes and non-rigid motion. To mitigate the variances in raw pixel features, a natural way is to gather statistical information of the raw pixel features. Among them, the covariance of a set of raw features inside a region of interest is one of the successful feature descriptors, which is called '*covariance matrix*' [8]. Covariance matrices as feature descriptors offer a convenient platform for fusing multiple features into a compact form independent of the number of data points. By choosing appropriate features, this fusion can be made invariant to image affine distortion [32], or robust or static image noise and illumination variations, while generating these matrices remains efficient using integral image transforms [5].

Covariance matrices are symmetric positive definite (SPD). It is well-known that all the SPD matrices form the so-called curved Riemannian manifold with negative sectional curvature [9], thus each covariance feature descriptor can be regarded as a point on the manifold. The distance between two points along this manifold is shorter than the corresponding Euclidean distance. As a result, for any algorithms that do not exploit the manifold geometry, the performance may be compromised in practical applications. Although the SPD manifold is embedded in its ambient Euclidean space, direct application of the LRR model in this ambient space may fail to capture intrinsic property of the covariance feature descriptors.

It is well-known that the SPD manifold is a Lie group for which the Log-Euclidean metrics are well established and particular simple [11], since they correspond to the Euclidean metrics in the logarithmic domain. Therefore, we use the matrix logarithm mapping from the SPD manifold to the normal symmetric matrix

*School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia; Email:{yfu, jbgao, dtien}@csu.edu.au.

†School of Systems Engineering, University of Reading, Reading, RG6 6AY, UK; Email:x.hong@reading.ac.uk.

space where we employ the Euclidean metrics. Then the SPD manifold has been flattened in its corresponding logarithmic space.

To this end, we study low rank representation of SPD matrices in the "flattened" space by using a dictionary comprised of original SPD matrices as atoms. The contributions of this work are listed as follows:

- We propose a novel LRR model on the manifold of SPD matrices. The approximation quality is measured by the extrinsic distance defined by the metric on tangent spaces.
- We describe a simple and effective approach for optimizing our objective function as the objective function is a composition of a quadratic term and the nuclear norm term.
- We present results on a few computer vision tasks on several image sets to demonstrate superior performance obtained by using our new LRR model.

2 Related Work

While significant steps have been taken to develop the theory of sparse coding and LRR in spaces, a handful of solutions have been devised to address the similar problems under non-Euclidean geometry.

There are two fundamental issues associated with sparse coding. First, a datum X_i should be approximately represented by a linear combination of dictionary elements as $\sum_{k=1}^{n} w_{ik}X_k$; Second, we need a measure for the error between X_i and $\sum_{k=1}^{n} w_{ik}X_k$. There is no problem at all for the traditional sparse coding in space, where the linear combination is an element in the same space, and the measure is explicitly defined by the Euclidean metric. However this is no longer the case on a manifold where we have lost the concept of linear operation.

The problem of sparse coding (SC)[37] of SPD matrices has recently received significant attention in the computer vision community. For this problem, the first issue arises because, for the given dictionary atoms $\{X_k\}$, there is no guarantee for the linear combination $\sum_{k=1}^{n} w_{ik}X_k$ to be a SPD matrix. This issue can be easily resolved by assuming combination coefficients w_{ik} be non-negative. This strategy has been seen in recent works [33], [15] and [10]. For the second issue, authors of [33] and [15] take different strategies. As a covariance matrix defines a Gaussian distribution, thus the log-determinant divergence is proposed to measure errors in [33], while the geodesic distance on the SPD manifold, a.k.a. Affine Invariant Riemannian Measures (AIRM), is proposed in [15] for the same purpose. Both strategies result in a highly nonlinear optimization problem requiring a sophisticated solver, which leads to its performance deterioration for moderately large covariances. In [10], a data matrix is approximated by a sparse linear combination of rank-one matrices under a Frobenius norm based loss. Although it is computationally efficient, it discards the manifold geometry. Kernel functions [12, 13, 14, 38] have also been explored to map the manifold features to a linear space, so that the linear combination issue is thus automatically resolved. Unfortunately, it is well-known that a kernelized sparse coding scheme suffers significantly when the number of dictionary atoms is high.

Most existing sparse coding methods for SPD matrices mainly focus on independent sparse representation for each data point. As a result, the relation among data points, or the underlying structure of subspaces that the subsets of group data generate, is usually not well considered. However, this intrinsic property is very important in some computer vision tasks, especially in classification and clustering applications.

In contrast, low rank representation uses holistic constraints as its sparse representation condition, which can reveal the latent sparse structure property embedded in a data set in high dimensional space. Low rank representations on non-Euclidean geometry have received comparatively little attention. Recently, a low rank representation on Grassmann manifolds is proposed in [16] by mapping the Grassmann manifold onto the Euclidean space of symmetric matrices. Some researchers exploited the local manifold structure of the data by adding a manifold regularization term characterized by a Laplacian graph [26, 39] or manifold matrix factorization [27]. However, the loss functions are still formulated in the Euclidean space. Others generalized Riemannian geometry to the matrix factorization with fixed-rank cases [29, 28, 30].

To our best knowledge, none of existing work is specialized for the low rank representation for SPD matrices measured simultaneously by the Riemmanian distance and nuclear norm, which motivates our study.

3 Preliminaries

A manifold $\mathcal{M}$ of dimensional d [34] is a topological space that locally resembles a Euclidean space $\mathbb{R}^d$ in a neighbourhood of each point $X \in \mathcal{M}$. For example, lines and circles are $1D$ manifolds, and surfaces, such as a plane, a sphere, and a torus, are $2D$ manifolds. Geometrically, a tangent vector is a vector that is tangent to a manifold at a given point X. Abstractly, a tangent vector is a functional, defined on the set of all the smooth functions over the manifold $\mathcal{M}$, which satisfies the Leibniz differential rule. All the possible tangent vectors at X constitutes a Euclidean space, named

the *tangent space* of $\mathcal{M}$ at X and denoted by $T_X\mathcal{M}$. If we have a smoothly defined metric (inner-product) across all the tangent spaces $\langle \cdot, \cdot \rangle_X : T_X\mathcal{M} \times T_X\mathcal{M} \to \mathbb{R}$ on every point $X \in \mathcal{M}$, then we call $\mathcal{M}$ *Riemannian manifold*. With a globally defined differential structure, manifold $\mathcal{M}$ becomes a differentiable manifold. SPD matrices form a differentiable Riemannian manifold. A *geodesic* between two points $X_i, X_j \in \mathcal{M}$ is the shortest smooth path connecting them.

There are predominantly two operations for computations on the Riemannian manifold, namely (1) the exponential map at point X, denoted by $\exp_X : T_X\mathcal{M} \to \mathcal{M}$, and (2) the logarithmic map, at point X, $\log_X : \mathcal{M} \to T_X\mathcal{M}$. The former projects a tangent vector in the tangent space onto the manifold, the latter does the reverse. Locally both mappings are diffeomorphic. Note that these maps depend on the manifold point X at which the tangent spaces are computed. Given two points $X_i, X_j \in \mathcal{M}$ that are close to each other, the distance between them can be calculated through the following formula as the norm in tangent space.

$$\text{dist}_{\mathcal{M}}(X_i, X_j) = \| \log_{X_i}(X_j) \|_{X_i} \tag{3.2}$$

The squared distance function $\text{dist}^2_{\mathcal{M}}(X, \cdot)$ is a smooth function for all $X \in \mathcal{M}$.

4 Problem Formulation

The Euclidean LRR model uses the Frobenius norm based on metric to model the reconstruction error, as shown in Eq. (1.1). However, in many real world applications, high dimension data have a Riemannian manifold structure, such as face recognition [17] and object detection [6]. In ideal scenarios, error should be measured according to the manifold's geometry. Inspired by the sparse coding formulation for data on the Riemannian manifold in [7], the LRR model in Eq. (1.1) can be changed to the following manifold form:

$$\begin{aligned} \min_W \sum_{i=1}^{n} \left\| \sum_{j=1}^{n} w_{ij} \log_{X_i}(X_j) \right\|^2_{X_i} + \|W\|_\star \\ \text{s.t. } \sum_{j=1}^{n} w_{ij} = 1, i = 1, 2, \ldots, n \end{aligned} \tag{4.3}$$

where $\mathbf{w}_i = (w_{i1}, \cdots, w_{in})$ denotes the low rank representation for X_i, and the i-th row of matrix W is $\mathbf{w}_i$.

There is an intuitive explanation for the formulation in Eq. (4.3). For each point X_i on the manifold, X_i can be projected onto 0 tangent vector of tangent space at X_i, other points X_j are projected as $\log_{X_i}(X_j)$. The norm in the first term of (4.3) means the distance between 0 tangent vector and the linearly combined tangent vector from all the other projected tangent vectors. Minimization aims at finding an appropriate linear combination for the "best" approximation in terms of tangent vectors.

The difference between the Euclidean LRR and our proposed method is illustrated in Figure 1. Given a toy people face dataset consisting of 32-by-32 face images of 2 people, each one performing 3 different pose variations, the goal is to recognize the people in each image. To start with, we extract a feature covariance matrix for each image, which lies on the SPD manifold. Traditional LRR directly vectorizes the covariance matrices to form their Euclidean features, and measure data distance using Euclidean geometry as shown in Figure 1(b). In contrast, our proposed method uses the covariance matrices on the Riemannian manifold as input features, and the distance between input SPD matrices on the manifold is approximated by the Euclidean distance between their corresponding mapped points on the tangent space, as shown in Figure 1(a). In Figure 1, we note that two points on the manifold having a large geodesic distance separating them may be near each other in the Euclidean space, which demonstrates that the important intrinsic properties of the data manifold cannot be captured using the Euclidean geometry.

When the underlying Riemannian manifold is SPD matrices, the Riemannian distance $\| \log_{X_i}(X_j) \|_{X_i}$ can be computed as follows:

Let $S_+(d)$ denote the space of $d \times d$ SPD matrices, the tangent space $T_X S_+(d)$ at every point $X \in S_+(d)$ is the space of $d \times d$ symmetric matrices. Let $Y, Z \in T_X S_+(d)$ be two tangent vectors at X, and define an inner product in $T_X S_+(d)$ using the formula

$$\langle Y, Z \rangle_X = \text{tr}(YX^{-1}ZX^{-1}) \tag{4.4}$$

where tr is the matrix trace. In information geometry, the above formula is the Fisher Information metric for $S_+(d)$ considered as the domain for parameterizing zero-mean normal distribution on $\mathbb{R}^d$ [18].

The Riemannian exponential map at X, which maps $Y \in T_X S_+(d)$ to a point in $S_+(d)$ is given by the formula

$$\exp_X(Y) = G\text{Exp}(G^{-1}G^{-T})G^T \tag{4.5}$$

where Exp denotes the matrix exponential.

For a given positive definite matrix $X \in S_+(d)$, denote G the square root matrix of X, i.e., $G = X^{\frac{1}{2}}$. Then the Riemannian logarithmic map $\log_X : S_+(d) \to T_X S_+(d)$ is given by

$$\log_X(Z) = G\text{Log}(G^{-1}ZG^{-T})G^T \tag{4.6}$$

where Log denotes the matrix logarithm.

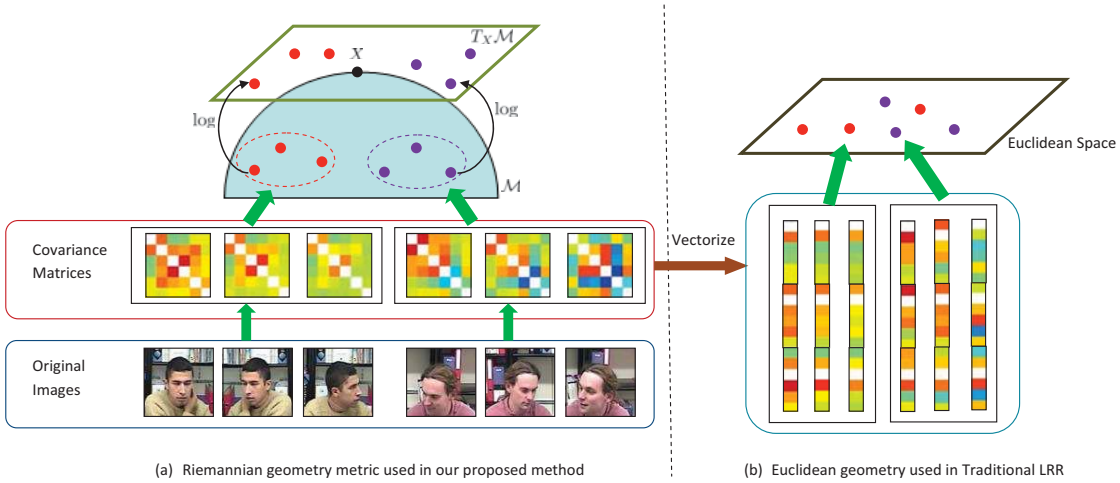


Figure 1: The illustration of distance metrics used in our proposed method and Euclidean LRR methods

Hence, under the Riemannian metric defined by (4.4), we have

$$\|\log_X(Z)\|_X^2 = \mathrm{tr}(\mathrm{Log}^2(G^{-1}ZG^{-T})) \tag{4.7}$$

Eq. (4.7) is useful for specializing Eq. (4.3) to $S_+(d)$: let $X_1, \ldots, X_n \in S_+(d)$ denote a collection of $d \times d$ SPD matrices, and $\mathcal{D}$ denotes the dictionary whose atoms consist of these input SPD matrices. Then the LRR on the manifold of SPD matrices can be defined as

$$\begin{aligned} \min_W \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1,k=1}^{n} w_{ij}w_{ik}\mathrm{tr}(L_{ij}L_{ik}) + \lambda||W||_\star \\ \text{s.t.} \quad \sum_{j=1}^{n} w_{ij} = 1, i = 1, 2, \ldots, n \end{aligned} \tag{4.8}$$

where $L_{ij} = \mathrm{Log}(G_i^{-1}X_jG_i^{-T})$. Eq.(4.8) can be written into a matrix form as follows:

$$\begin{aligned} \min_W \frac{1}{2}\sum_{i=1}^{n} \mathbf{w}_i Q_i \mathbf{w}_i^T + \lambda\|W\|_\star \\ \text{s.t.} \quad \sum_{j=1}^{n} w_{ij} = 1, i = 1, 2, \ldots, n \end{aligned} \tag{4.9}$$

where $\mathbf{w}_i$ is the i-th row of W and $Q_i = [\mathrm{tr}(L_{ij}L_{ik})]$ are $n \times n$ matrices.

5 Solution to LRR on SPD matrices

In this section, we consider an algorithm to solve the constrained optimization problem in Eq. (4.9). We propose to use the Augmented Lagrange Multiplier (ALM) method [19] to solve it. The reason we choose the ALM to solve this optimization problem is threefold: (1) superior convergence property of ALM makes it very attractive; (2) parameter tuning is much easier than the iterative thresholding algorithm; and (3) it converges to an exact optimal solution.

First of all, the augmented Lagrange problem of (4.9) can be written as

$$\begin{aligned} L = \sum_{i=1}^{n}\Bigg(\frac{1}{2}\sum_{i=1}^{n}\mathbf{w}_i Q_i \mathbf{w}_i^T + y_i(\sum_{j=1}^{n} w_{ij} - 1) \\ + \frac{\beta}{2}(\sum_{j=1}^{n} w_{ij} - 1)^2\Bigg) + \lambda||W||_\star \end{aligned} \tag{5.10}$$

where y_i are Lagrangian multipliers, and β is a weight to tune the error term of $(\sum_{j=1}^{n} w_{ij} - 1)^2$.

In fact, the above problem can be solved by updating one variable at a time with all the other variables fixed. More specifically, the iterations of ALM go as follows:

1) Fix all others to update W: We define a function $F(W)$ by

$$\begin{aligned} F(W) = \sum_{i=1}^{n}\Bigg(\frac{1}{2}\sum_{i=1}^{n}\mathbf{w}_i Q_i \mathbf{w}_i^T + y_i(\sum_{j=1}^{n} w_{ij} - 1) \\ + \frac{\beta}{2}(\sum_{j=1}^{n} w_{ij} - 1)^2\Bigg) \end{aligned} \tag{5.11}$$

and it is easy to prove that

$$\frac{\partial F}{\partial \mathbf{w}_i} = \mathbf{w}_i Q_i + y_i\mathbf{1} + \beta(\sum_{j=1}^{n} w_{ij} - 1)\mathbf{1} \tag{5.12}$$

where $\mathbf{1} \in \mathbb{R}^n$ is a row vector of all ones. Now at the

current location $W^{(k)}$, we take a linearization of $F(W)$,

$$\begin{aligned} F(W) \approx & F(W^{(k)}) + \langle \partial F(W^{(k)}), W - W^{(k)} \rangle \\ & + \frac{\mu_k}{2} ||W - W^{(k)}||_F^2 \end{aligned} \tag{5.13}$$

where $\partial F(W^{(k)})$ is a gradient matrix whose i-th row is given by

$$\mathbf{w}_i^{(k)} Q_i + y_i^{(k)} \mathbf{1} + \beta (\sum_{j=1}^{n} w_{ij}^{(k)} - 1)\mathbf{1} \tag{5.14}$$

Taking Eq.(5.14) into Eq.(5.10), we have

$$\begin{aligned} W^{(k+1)} = & \arg\min_W F(W^{(k)}) + \langle \partial F(W^{(k)}), W - W^{(k)} \rangle \\ & + \frac{\mu_k}{2} ||W - W^{(k)}||_F^2 + \lambda ||W||_\star \\ = & \arg\min_W \frac{\mu_k}{2} ||W - \big(W^{(k)} - \frac{1}{\mu_k} \partial F(W^{(k)})\big)|| \\ & + \lambda ||W||_\star \end{aligned} \tag{5.15}$$

The above problem admits a closed form solution by using SVD thresholding operator to $M = W^{(k)} - \frac{1}{\mu_k} \partial F(W^{(k)})$. Taking SVD for $M = U \Sigma V^T$, then the new iteration is given by

$$W^{(k+1)} = U \text{soft}(\Sigma, \frac{\mu_k}{\lambda}) V^T \tag{5.16}$$

where $\text{soft}(\Sigma, \sigma) = \max\{0, (\Sigma_{ii} - \frac{1}{\sigma})\}$ is the soft thresholding operator for a diagonal matrix, see [31].

2) Fix all others to update y_i by

$$y_i^{k+1} \leftarrow y_i^k + \beta_k (\sum_{j=1}^{n} w_{ij}^{k+1} - 1) \tag{5.17}$$

After a number of iterations by alternatively updating W and y_i respectively, we achieved a complete solution to LRR on SPD matrices. The whole procedure of LRR on SPD matrices is summarized in Algorithm 1.

5.1 Convergence Analysis The convergence of ALM algorithm has been guaranteed by Theorem 3 in [35]. To ensure the convergence of the Algorithm 1, we only need prove that problem (4.9) is a convex optimization problem.

Theorem 1 Each Q_i is a semi-positive definite matrix, thus problem (4.9) is a convex optimization.

Proof. The jk-th element q_{jk}^i of Q_i is the trace of the matrix product of symmetrical matrices L_{ij} and L_{ik}, thus it can be rewritten as

$$q_{jk}^i = \text{tr}(L_{ij} L_{ik}) = \text{tr}(L_{ij}^T L_{ik}) = \text{vec}(L_{ij})^T \text{vec}(L_{ik})$$

If we define a new matrix V_i by

$$V_i = [\text{vec}(L_{i1}), \text{vec}(L_{i2}), \ldots, \text{vec}(L_{in})] \tag{5.18}$$

then we can see $Q_i = V_i^T V_i$, which means Q_i has a decomposition as the product of a matrix with its transpose, hence Q_i is semi-positive definite.

5.2 Algorithm Complexity For ease of analysis, we assume that the size of X_i is $d \times d$, and the iteration number of ALM algorithm is t. The complexity of Algorithm 1 can be decomposed into two parts: the data preparation part (Step 1-4) and the ALM solution part (Step 5-9).

In the data preparation part, G_i and L_{ij} for $(i, j = 1, \ldots, n)$ can be computed with a complexity of $O(nd^3 + n^2 d^3)$. The bottleneck of data preparation procedure is the computation of all Q_i with total cost of $O(n^3 d^2)$. However as $Q_i = V_i^T V_i$, to break this complexity bound, we use a parallel matrix multiplication scheme [36] to reduce the complexity to $O(d^2 \text{log} n)$. Therefore, the complexity of data preparation is $O(nd^3 + n^2 d^3) + O(d^2 \text{log} n)$.

In the ALM, the major computation cost is for SVD of an $n \times n$ matrix in Step 6 with a complexity of $O(n^3)$. Fortunately, we can utilize the accelerated method in [35] to reduce the complexity of partial SVD computation to $O(rn^2)$, where r is predicted rank of $W^{(k+1)}$. For t iterations, the complexity of ALM solution is $O(trn^2)$.

With the above analysis, the overall complexity of Algorithm 1 is given as

$$O(nd^3 + n^2 d^3) + O(d^2 \text{log} n) + O(trn^2) \tag{5.19}$$

As $d \ll n$, the complexity of Algorithm 1 mainly depends on the size of data set n. It can be approximated as $O(n^2 d^3) + O(trn^2)$, which is similar to the complexity of the original LRR algorithm $O(trn^2)$.

Algorithm 1 Solving Problem (4.9) by ALM

Require: The SPD matrices sample set$\{X_i\}_{i=1}^n, X_i \in S_+(d)$, a penalty parameter λ.

Ensure: : The low rank representation W

1: initialize:$W = 0, y_i = 0 \ (i = 1, \ldots, n), \beta = 10^{-6}, \beta_{\max} = 10^{10}, \rho = 1.1, \varepsilon = 10^{-8}$.
2: $G_i \leftarrow$ a square root of X_i for $i = 1, 2, \ldots, n$;
3: $L_{ij} \leftarrow \text{Log}(G_i^{-1} X_j G_i^{-T})$ for $i, j = 1$ to n;
4: $q_{jk}^i \leftarrow \text{tr}(L_{ij} L_{ik})$ for $i, j, k = 1$ to n;
5: **while** $|\sum_{j=1}^n w_{ij} - 1| \geq \varepsilon$ **do**
6: W is computed by solving the problem (5.15) using SVD thresholding operator;
7: y_i is updated by (5.17).
8: $\beta \leftarrow \min(\rho\beta, \beta_{max})$
9: **end while**

6 Experiments

To evaluate the proposed LRR model on the manifold of SPD matrices, we apply it to both clean and corrupted image datasets for image classification and image segmentation.

To apply our method for image classification, we firstly employ our proposed LRR model to obtain the low rank features, and the classification accuracy is computed by training a SVM classifier on the low rank features. In order to compare performances with the state-of-the art methods, we use the following four baseline methods:

- SVM on vectorized data: we directly vectorize the manifold data to form their Euclidean features and train an SVM using these Euclidean features.
- LRR +SVM: we firstly apply the Euclidean LRR model in [1] on the Euclidean features (vectorized the manifold data) to obtain the low rank features, and then train an SVM classifier on the low rank features.
- GKNN [7]: is a K-nearest neighbour classifier that uses Riemannian distance on the manifold for determining neighbours, and solves the classification problem on the manifold directly without LRR transforms.
- SC+SVM : is a variant of our proposed method within the same framework. The only difference between SC+SVM and our proposed model is that l_1 regularization on W is used in the objective function Eq.(4.9).

All SVMs used in the experiments are trained using the LIBSVM package [20].

To apply our method for image segmentation, we compute the low rank features using the proposed method, and then use these low rank features as the input of the Ncut clustering algorithm. The baselines used in the experiments are listed as follows:

- Ncut [21]: we simply conduct the Ncut clustering algorithm over the vectorized manifold data.
- LRR+Ncut: we use the Euclidean LRR method on the Euclidean features to obtain the low rank features, and then apply Ncut to the low rank features.
- Riemannian Distance based Ncut (RNcut): is a variant of the traditional Ncut method, based on the Riemannian distance introduced in this paper as the node similarity measure.

6.1 Performance for Image Classification

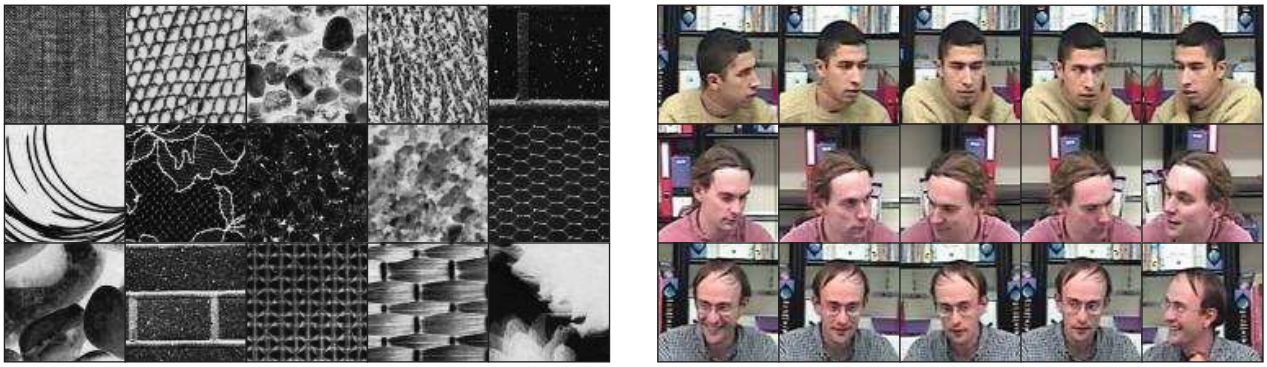
In this experiment, we evaluate the performance of our proposed method in terms of image classification using Brodatz texture [22] and IDIAP [24] datasets. Sample images from both data sets are shown in Figure 2.

Brodatz texture dataset contains two subsets with 16 textures and 32 textures respectively. Each pixel of the image is characterized by a 5-dimensional feature vector $C = \left(I, |\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial y}|, |\frac{\partial^2 I}{\partial x^2}|, |\frac{\partial^2 I}{\partial y^2}|\right)^T$, where I is the intensity of the pixel, $|\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial y}|, |\frac{\partial^2 I}{\partial x^2}|, |\frac{\partial^2 I}{\partial y^2}|$ denote the norm of the first and second order derivatives of the intensities with respect to x and y respectively. We divide each 256×256 texture image into 64 non-overlapping blocks of size 32×32. Inside each block, we compute a 5×5 covariance matrix CC^T summing over the block . To ensure that each covariance matrix is positive-definite, we use $CC^T + \sigma E$ as our final input features, where σ is a small positive constant and E is the identity matrix.

The IDIAP data set comes from 8 meeting sequences of 360×288 frame resolution, where two individuals were captured while discussing various topics in a 4-person dialogue scenario. The total number of different people captured is 15, each person having ten images with different head poses. A face detector was used to extract the bounding box of each face in every video frame. All the acquired image regions were resized to 75×75 pixels. Using the similar experimental setup as in [25], we randomly extract a subset of 5288 images from the training data in [25] as our experimental database. We involve 3 subsets with 5 people, 10 people and 15 people respectively.

For people recognition, each pixel is mapped to a 9-dimensional feature: $C = \big(x, R(x,y), G(x,y), B(x,y), Y_i, |G_x|, |G_y|, |\sin(\theta) + \cos(\theta)|, |H_y|\big)^T$, where x is the x-coordinate, $R(x,y), G(x,y), B(x,y)$ are the color (RGB) values, Y_i is the pixel intensity in the YCbCr color space, $|G_x|$ and $|G_y|$ are the absolute magnitudes of pixel gradients in each orientation, the angle of orientation of the edge (relative to the pixel grid) giving rise to the spatial gradient is given by $\theta = \tan^{-1}(\frac{G_y}{G_x})$ and the y-gradient of pixel hue is denoted by H_y. For each image, we divide it into upper and lower parts and compute the covariance matrix CC^T for each part. Then we create a block diagonal matrix of size 18×18 as image feature descriptor, where the diagonal elements consist of the above two 9×9 covariance matrices.

The classification results on the above two datasets are reported in Tabel 1. Among the four comparative methods, LRR+SVM outperforms SVM, which demonstrates that low rank features are more discriminative for classification than the original high-dimensional



(a) Brodatz (b) IDIAP

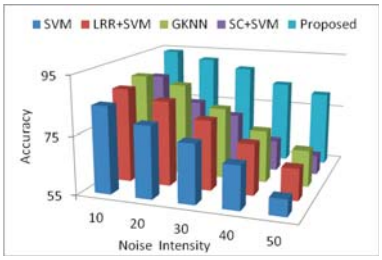
Figure 2: Sample Images for classification

features. Another interesting observation is that the method considering the intrinsic property of the data manifold like GKNN performs better than ones using extrinsic Euclidean metric (SVM and LRR+SVM), which confirms the importance of intrinsic geometry. Not surprisingly, our method and SC+SVM utilizing both intrinsic geometry and sparse feature transform (i.e. sparse coding and low rank representation) outperform all the other baselines. The performance of our proposed method outperforms its variant method SC+SVM under the same framework, which suggests that the lowest-rank criterion is accurate for modeling the embedding manifold structures of high dimensional data.

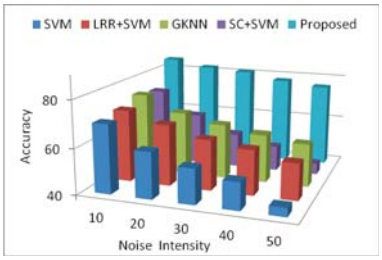
Table 1: Classification Accuracy comparisons on Clean data sets

	Brodatz		IDIAP		
Class	16	32	5	10	15
SVM	93.36	88.67	76.54	72.75	68.56
LRR+SVM	95.08	90.99	80.17	75.81	72.89
GKNN	95.7	92.11	82.69	77.32	80.36
SC+SVM	99.37	95.43	87.71	82.69	83.78
Proposed	99.89	97.12	90.38	87.78	87.33

We further examine the noise robustness of the proposed model. We add Guassian white noises with different intensity values ranging from 10 to 50 to both datasets. Figure 3 illustrates the classification results for all the methods. The pattern among methods of SVM, LRR+SVM and GKNN in the previous experiment is also observed in this experiment, confirming the importance of intrinsic geometry and sparse feature transform. However, the performance of SC+SVM is inferior to all LRR based methods (LRR+SVM and our proposed method), which shows that sparse representation is not good at handling corrupted data like LRR. Our model performs best among all the methods on the corrupted data set. This is because it explores data spatial correlation information with a low rank representation in the intrinsic Riemannian geometry, which guarantees an accurate classification of data into different categories.



(a) 16-class Brodatz



(b) 15-class IDIAP

Figure 3: Classification Accuracy comparisons on Noisy data sets

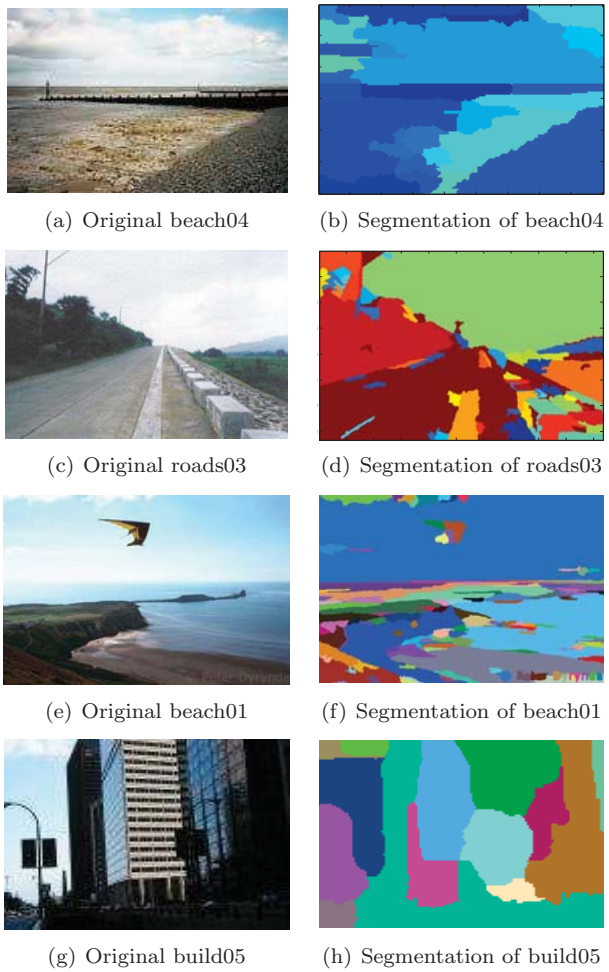
6.2 Performance for Image Segmentation In this section, our proposed method is used to solve the image segmentation problem. The test images include 4 images in the Automatic Photo Pop-up data set [23], named as "beach01", "beach04", "roads03", and "build05", which are resized uniformly into 128×128. We start with extracting several features from each pixel in the image. For the image segmentation problem, we use pixel locations (x, y), color (RGB) values and the norm of the first and second order derivatives of the intensities with

Table 2: Image segmentation accuracy on the Automatic Photo Pop-up dataset

Dataset	Ncut	LRR+Ncut	RNcut	Proposed
beach04	76.75	80.55	81.67	84.96
roads03	78.66	82.51	83.68	87.99
beach01	80.56	84.33	86.56	89.34
build05	75.44	78.34	80.54	83.55

respect to x and y. Each pixel of the image is converted to a nine-dimensional feature vector $C = \left(x, y, R(x,y), G(x,y), B(x,y), |\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial y}|, |\frac{\partial^2 I}{\partial x^2}|, |\frac{\partial^2 I}{\partial y^2}|\right)^T$. The covariance matrix CC^T is a 9×9 matrix.

The segmentation results are detailed in Table 2. It has shown that the performance of Ncut is inferior to that of LRR+Ncut, which indicates that low rank features are more discriminative and hence useful than the data themselves for clustering problems. As RNcut and our proposed model use the Riemannian distance measurement, their clustering results are better than that of Ncut and LRR+Ncut, in which the Euclidean distance is used to measure the similarities between points on the manifold. This is mainly because manifold mapping extracts more useful information about the difference among the sample data. Besides, our proposed method has the highest accuracy among all the baseline methods, which suggests that the combination of Riemannian distance and the LRR model brings good accuracy for Ncut clustering. To visualize our proposed model's effectiveness in image segmentation, we show the segmentation results of above 6 images in Figure 4.



(a) Original beach04 (b) Segmentation of beach04

(c) Original roads03 (d) Segmentation of roads03

(e) Original beach01 (f) Segmentation of beach01

(g) Original build05 (h) Segmentation of build05

Figure 4: The image segmentation results of beach04, roads03 , beach01 and build05 by our proposed algorithm, the original images are shown in the first column, and the segmentation results are shown in the second column.

7 Conclusions

In this paper, we propose a novel LRR model on the manifold of SPD matrices, in which we exploit the intrinsic property of SPD matrices manifold in the Riemannian geometric context. Compared with the existing Euclidean LRR algorithms, the loss of the global linear structure is compensated by the local linear structures given by the tangent spaces of the manifold. Further, we derive a easily solvable optimization problem, which incorporates the structured embedding mapping into the LRR model. Our experiments demonstrate that our proposed method is efficient and robust to the noise, and produces superior results compared to other state-of-art methods for classification and segmentation applications on several computer vision datasets.

8 Acknowledgements

This research was supported under Australian Research Council Discovery Projects funding scheme (project DP130100364).

References

[1] G. Liu, Z. Liu and Y. Yu, *Robust subspace segmentation by low-rank representation*, Proceedings of ICML,2010.

[2] G. Liu, Z. Liu, J. Sun, Y. Yu and Y. Ma, *Robust recovery of subspace structures by low-rank representation*, Pattern Analysis and Machine Intelligence, IEEE Transaction on, 35 (2013), pp. 171–184.

[3] J. Wright, A. Ganesh, S. Rao, Y. Peng and Y. Ma, *Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization*, Proceedings of NIPS,2009.

[4] C. Lang, G. Liu, J. Yu and S. Yan, *Saliency detection by multitask sparsity pursuit*, Image Processing, IEEE Transaction on, 21 (2012), pp. 1327–1338.

[5] F. Porikli and O. Tuzel, *Tuzel: Covariance tracker*, Proceedings of CVPR, 2006.

[6] B. Ma, Y. Su, F. Jurie, et al, *Bicov: a novel image representation for person re-identification and face verification*, Proceedings of BMVC, 2012.

[7] Y. Xie, J. Ho and B. Vemuri, *On a nonlinear generalization of sparse coding and dictionary learning*, Proceedings of ICML, 2013.

[8] O. Tuzel, F. Porikli and P. Meer, *Region covariance: a fast descriptor for detection and classification*, Proceedings of ECCV, 2006.

[9] W. Ziller, *Examples of Riemannian manifolds with non-negative sectional curvature*, Metric and Comparison Geometry, Surv. Diff. Gem, International Press, (11) 2007, pp. 63–102.

[10] S. Sra and A. Cherian, *Generalized dictionary learning for symmetric postive definite matrices with application to nearest neigbour retrieval*, Proceedings of ECML, Springer, 2011.

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

[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*, Proceedings of ECCV, Springer, 2012.

[13] S. Jayasumana, R. Hartley, M. Salzmann, H. Li and M. Harandi, *Kernel methods on the Riemannian manifold of symmetric positive definite matrices*, Proceedings of CVPR, 2013.

[14] P. Li, Q. Wang, W. Zuo and L. Zhang, *Log-Euclidean kernels for sparse representation and dictionary learning*, Proceedings of ICCV, 2013.

[15] A. Cherian and S. Sra, *Riemannian sparse coding for positive definite matrices*, Proceedings of ECCV, 2014.

[16] B. Wang, Y. Hu, J. Gao, Y. Sun and B. Yin *Low rank representation on Grassmann manifold*, accepted by ACCV, 2014.

[17] Y. Pang, Y. Yuan and X. Li, *Gabor-based region covariance matrices for face recognition*, Circuits and Systems for Video Technology, IEEE Transaction on, 18(7)(2008), pp. 989–993.

[18] S. Amari and H. Nagaoka, *Methods of Information Geometry*, American Mathematical Society, 2007.

[19] Y. Shen, Z. Wen and Y. Zhang, *Augmented Lagrangian alternating direction method for matrix separation based on low-rank factorization*, Optimization Methods and Software, 29(2014), pp. 239–263.

[20] C. C. Chang and C. J. Lin, *LIBSVM: A library for support vector machines*, ACM Transactions on Intelligent Systems and Technology, 2(2011), PP. 27:1–27:27.

[21] J. Shi and J. Malik, *Normalized cuts and image segmentation*, Pattern Analysis and Machine Intelligence, IEEE Transaction on, 22(2000), pp. 888–905.

[22] P. Brodatz, *Textures: a photographic album for artists and designers*, vol. 66, Dover New York, 1966.

[23] D. Hoiem, A. A. Efros and M. Hebert *Automatic photo pop-up*, Proceedings of SIGGRAPH, 2005, pp. 577–584.

[24] J. M. Odobez, *Idiap head pose database*, http://www.idiap.ch/dataset/headpose.

[25] S. Ba and J. Odobez, *Evaluation of multiple cue head pose estimation algorithms in natural environements*, Proceedings of ICME, 2005, pp. 1330-1333.

[26] J. Liu, Y. Chen J. Zhang and Z. Xu, *Enhancing low-rank subspace clustering by manifold regularization*, Image Processing , IEEE Transaction on, 23(9)(2014), pp. 4022–4030.

[27] Z. Zhang and K. Zhao, *Low-rank matrix approximation with manifold regularization*, Pattern Analysis and Machine Intelligence, IEEE Transaction on, 35(7)(2013), pp. 1717–1729.

[28] U. Shalit, D. Weinshall and G. Chechik, *Online learning in the embedded manifold of low-rank matrices.* Journal of Machine Learning Research, 13(2012), pp. 429–458.

[29] Y. C. Eldar, D. Needell and Y. Plan, *Uniqueness conditions for low-rank matrix recovery*, Proceedings of SPIE, 2011.

[30] B. Vandereycken, P. A. Asil and S. Vandewalle, *A Riemannian geometry with complete geodesics for the set of positive semidefinite matrices of fixed rank*, IMA Journal of Numerical Analysis, 33(2013), pp. 481–514.

[31] J. Cai, E. J. Candes and Z. Shen, *A singular value thresholding algorithm for matrix completion*, SIAM J. on Optimization, 20(4)(2010), pp. 1956–1982.

[32] B. Ma, Y. Wu, F. Sun, *Affine object tracking using kernel-based region covariance descriptors*, Foundations of Intelligent Systems, Springer, (2012), pp. 613–623.

[33] R. Sivalingam, D. Boley, V. Morellas and N. Papanikolopoulos, *Tensor sparse coding for region covariances*, Proceedings of ECCV, Springer, 2010.

[34] J. M. Lee, *Introduction to Smooth Manifolds*, Graduate Texts in Mathematics, 218(2002), Spring.

[35] Z. Lin, R. Liu and Z. Su, *Linearized alternating direction method with adaptive penalty for low rank representation*, Proceedings of NIPS, 2011.

[36] G. H. Chen, B. F. Wang and C. J. Lu, *On the Parallel Computation of the Algebraic Path Problem*, Parallel and Distributed Systems, IEEE Transactions on, 3(2)(1992), pp. 251–256.

[37] J. Gui, D. Tao, Z. Sun, Y. Luo, X. You and Y. Tang, *Group sparse multiview patch alignment framework with view consistency for image classification*, Image Processing, IEEE Transactions on, 23(7) (2014), pp. 3126–3137.

[38] J. Gui, Z. Sun, J. Cheng, S. Ji and X. Wu, *How to estimate the regularization parameter for spectral regression discriminant analysis and its kernel version?*, Circuits and Systems for Video Technology, IEEE Transactions on, 24(2)(2014), pp. 211–223.

[39] B. Geng, D. Tao, C. Xu, L. Yang and X. Hua, *Ensemble manifold regularization*, Pattern Analysis and Machine Intelligence, IEEE Transactions on, 3(6)(2012), pp. 1227–1233.