Product Grassmann Manifold Representation and Its LRR Models

Boyue Wang[1], Yongli Hu[1], Junbin Gao[2], Yanfeng Sun[1] and Baocai Yin[1,3]

[1]Beijing Key Laboratory of Multimedia and Intelligent Software Technology
College of Metropolitan Transportation, Beijing University of Technology Beijing, 100124, China
boyue.wang@emails.bjut.cn, {huyongli,yfsun,ybc}@bjut.edu.cn
[2]School of Computing and Mathematics, Charles Sturt University Bathurst, NSW 2795, Australia
jbgao@csu.edu.au
[3]School of Software Technology at Dalian University of Technology, Dalian 116620, China

Abstract

It is a challenging problem to cluster multi- and high-dimensional data with complex intrinsic properties and non-linear manifold structure. The recently proposed subspace clustering method, Low Rank Representation (LRR), shows attractive performance on data clustering, but it generally does with data in Euclidean spaces. In this paper, we intend to cluster complex high dimensional data with multiple varying factors. We propose a novel representation, namely Product Grassmann Manifold (PGM), to represent these data. Additionally, we discuss the geometry metric of the manifold and expand the conventional LRR model in Euclidean space onto PGM and thus construct a new LRR model. Several clustering experimental results show that the proposed method obtains superior accuracy compared with the clustering methods on manifolds or conventional Euclidean spaces.

1 Introduction

In many practical applications, one often faces the problem of clustering image sets into proper classes. For example, in action classification or recognition applications, action video clips are assigned with actual action labels. Another typical example is for facial image sets. Generally, the facial images of different subjects vary according to multiple factors, such as illumination, pose and expression. So it is a challenging problem for facial image clustering or recognition, particularly there are a set of facial images available for each person. However, the image set clustering problem is different from the traditional image clustering problem, in which each image is assigned to a class label. In the image set clustering, a clustering label is given to an image set which generally consists of several images from the same subject. Thus a critical problem is to effectively represent image sets and design a proper clustering method.

The subspace method and its clustering method have attracted great interest in computer vision, pattern recognition and signal processing (Elhamifar and Vidal 2013; Vidal 2011; Xu and Wunsch-II 2005). The basic idea of subspace clustering relies on that most data often have intrinsic subspace structures and can be regarded as the samples of a mixture of multiple subspaces. Thus the main goal of subspace clustering is to group data into different clusters, data points in each of which just come from one subspace. To investigate and represent the underlying subspace structure, many subspace methods have been proposed, such as the conventional iterative methods (Ho et al. 2003; Tseng 2000), the statistical methods (Gruber and Weiss 2004; Tipping and Bishop 1999), the factorization-based algebraic approaches (Kanatani 2001; Ma et al. 2008; Hong et al. 2006), and the spectral clustering-based methods (Chen and Lerman 2009; Favaro, Vidal, and Ravichandran 2011; Lang et al. 2012; Liu and Yan 2011; von Luxburg 2007).

The spectral clustering methods based on affinity matrix are considered having good prospects, in which an affinity matrix is firstly learned from the given data and the final

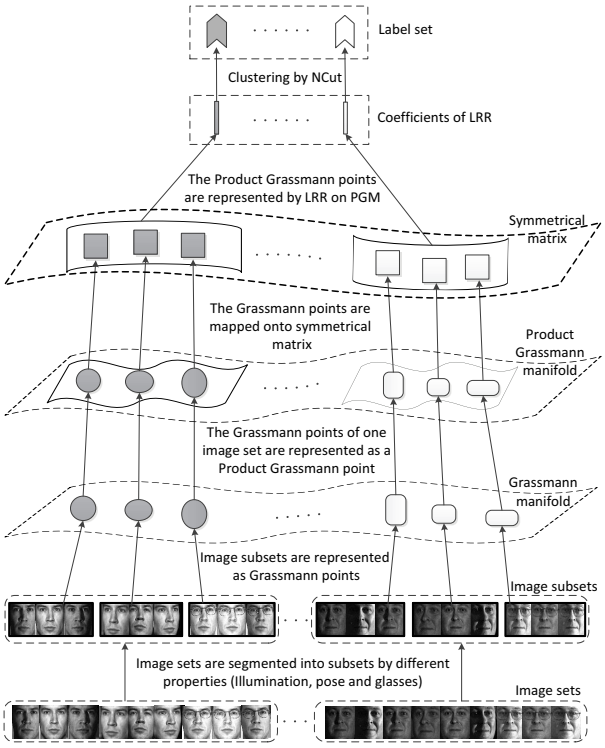

Figure 1: An overview of our proposed LRR on Product Grassmann manifolds.

clustering results are obtained by spectral clustering algorithms such as K-means or Normalized Cuts (NCut) (Shi and Malik 2000). The main component of the spectral clustering methods is to construct a proper affinity matrix for different data. In Sparse Subspace Clustering (SSC) (Elhamifar and Vidal 2013), one assumes that the data of subspaces are independent and are sparsely represented under the so-called ℓ_1 Subspace Detection Property (Donoho 2004). It has been proved that under certain conditions the multiple subspace structures can be exactly recovered via $\ell_p (p \leq 1)$ minimization (Lerman and Zhang 2011). However, the current sparse subspace methods mainly focus on independent sparse representation for data objects. The low rank representation (LRR) (Liu, Lin, and Yu 2010) further introduces a holistic constraint, i.e., the low rank or nuclear norm $\|\cdot\|_*$ to reveal the latent structural sparse property embedded in the data set. When a high-dimensional data set is actually from a union of several low dimension subspaces, the LRR model can reveal this structure through subspace clustering.

Although the subspace clustering methods have good performance in many applications, the current methods assume that data objects come from linear space and the similarity among data is measured in Euclidean-alike distance. However, this hypothesis may not be always true in practice since data may reside in a "curved" nonlinear manifold. In fact, many high-dimensional data are embedded in low dimensional manifolds. So it is desired to reveal the nonlinear manifold structure underlying these high-dimensional data and obtain proper representation and clustering method for the data derived from non-linear space.

To explore the non-linear structure underlying the data, many manifold related methods are proposed. The classic manifold learning methods, such as Locally Linear Embedding (Roweis and Saul 2000), ISOMAP (Tenenbaum, Silva, and Langford 2000) try to learn manifold structures from data by exploring data local geometry, and ultimately to complete other learning tasks, e.g., Sparse Manifold Clustering and Embedding (Elhamifar and Vidal 2011) and the kernel LRR method (Wang, .Saligrama, and nón 2011).

On the other hand, in many scenarios, data are generated from a known manifold. For example, covariance matrices are used to describe the region feature (Tuzel, Porikli, and Meer 2006). In fact, the covariance matrix descriptor is a point on the manifold of symmetrical positive definite matrices. Similarly an image set can be represented as a point on the so-called Grassmann manifold (Harandi et al. 2013). It is beneficial to use manifold properties in designing new learning algorithms. Shirazi et al. (2012) embeds the Grassmann manifolds into reproducing kernel Hilbert spaces. Turaga et al. (2011) presents statistical modeling methods that are derived from the Riemannian geometry of the manifold. A Low-Rank Representation on Grassmann Manifold was explored in our recent paper (Wang et al. 2014).

Although using points on Grassmann manifold is a natural way to represent image sets, the current single space representation method is still limited for the image sets with multiple variations. For example, the human face images are generally captured under different views and illuminations with various expressions, poses and accessorizing. Another problem is that there usually exist noises or outliers in a dataset. For example, some non-face images or another person's face images are often mixed into one's face image set. Moreover, many new types of signals are composed of heterogeneous data with different modalities, such as the RGB-D data and other multi-sources data. In these cases, it is difficult to represent data in a uniform space or construct a proper transformation between different subspaces. So how to properly represent these data with multi-factors and obtain good clustering results is still a challenging problem for Grassmann manifold based clustering method.

In this paper, we concentrate on the image set clustering problem and propose a novel image set representation using Product Grassmann manifold to describe the intrinsic complexity of image sets. The motivation of using product space representation is that product space is a good mathematical tool to represent multi-factors with multi-subspaces. To further use the Product Grassmann manifold in image sets clustering, we explore the geometry property of the Product Grassmann Manifold (PGM) and expand conventional LRR model onto PGM. Also the proposed LRR model on PGM is kernelized. The pipeline of our method is illustrated in Figure 1. Our main contributions are

- Proposing a new data representation based on PGM for image sets with multiple varying factors;
- Formulating the LRR model on PGM;
- Presenting a new general kernelized LRR model on PGM.

The major difference from the previous work (Wang et al. 2014) lies in the new representation of multiple varying factors in image sets and its LRR.

2 Data representation by Product Grassmann Manifold

2.1 Grassmann Manifold

Grassmann manifold $\mathcal{G}(p, d)$ (Absil, Mahony, and Sepulchre 2008) is the space of all p-dimensional linear subspaces of $\mathbb{R}^d$ for $0 \leq p \leq d$. A point on Grassmann manifold is a p-dimensional subspace of $\mathbb{R}^d$ which can be represented by any orthonormal basis $X = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_p] \in \mathbb{R}^{d \times p}$. The chosen orthonormal basis is called a representative of its subspace span(X). There are many ways to represent Grassmann manifold. In this paper, We take the way of embedding Grassmann manifold into the space of symmetric matrices Sym(d). Here the embedding mapping is defined as, see (Harandi et al. 2013),

$$\Pi : \mathcal{G}(p, d) \to \text{Sym}(d), \quad \Pi(X) = XX^T. \tag{1}$$

The embedding $\Pi(X)$ is diffeomorphism (Helmke and Hüper 2007), hence it is reasonable to replace the distance on Grassmann manifold with the following distance defined on the symmetric matrix space under this mapping,

$$d_g^2(X, Y) = \frac{1}{2}\|\Pi(X) - \Pi(Y)\|_F^2. \tag{2}$$

2.2 Product Grassmann Manifold (PGM)

PGM is considered as a space of product of multiple Grassmann spaces. Given a set of natural number $\{p_1, ..., p_M\}$, we define the PGM $\mathcal{PG}_{d:p_1,...,p_M}$ as the space of $\mathcal{G}(p_1, d) \times \cdots \times \mathcal{G}(p_M, d)$. So a PGM point can be represented as an assembled Grassmann point, denoted by $[X] = \{X^1, ..., X^M\}$ such that $X^m \in \mathcal{G}(p_m, d), m = 1, ..., M$.

For our purpose, we adopt a weighted sum of Grassmann distances as the distance on PGM,

$$d_{\mathcal{PG}}([X], [Y])^2 = \sum_{m=1}^{M} w_m d_g^2(X^m, Y^m). \tag{3}$$

where w_m is the weight to represent the importance of each Grassmann space. In practice, it can be determined by a data driven manner or according to prior knowledge. In this paper, we simply set all $w_m = 1$. So from (2), we obtain the following distance on PGM,

$$d_{\mathcal{PG}}([X], [Y])^2 = \sum_{m=1}^{M} \frac{1}{2} \|X^m (X^m)^T - Y^m (Y^m)^T\|_F^2. \tag{4}$$

2.3 Data representation by PGM (the New Idea)

Given a set of data, e.g., a facial image set from one subject, denoted by $\mathbf{I} = \{\mathbf{I}_1, ..., \mathbf{I}_P\}$, which is generated/taken under M varying factors, we can construct a disjoint union of subsets $\mathbf{S} = \{\mathbf{S}_1, ..., \mathbf{S}_M\}$ such that $\mathbf{S}_m \subset \mathbf{I}, m = 1, ..., M$ is the subset corresponding to the mth factor, for example, the subset of facial images with various illuminations. For each subset $\mathbf{S}_m \in \mathbf{S}$, we first represent it as a Grassmann point. Then we construct a PGM point by combining these Grassmann points. Here we adopt SVD to construct an orthogonal basis to represent the subset of images as a Grassmann point. Finally, we combine M Grassmann points to obtain the aforementioned Product Grassmann point $[X] = \{X^1, ..., X^M\}$. This is a new way to jointly describe image sets by using factor-related subspaces, rather than a single subspace as done in subspace analysis.

3 LRR on Product Grassmann Manifold

3.1 The Classic LRR

Given a set of data drawn from an unknown union of subspaces $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ where D is the data dimension, the objective of subspace clustering is to assign each data sample to its underlying subspace. The basic assumption is that the data in $\mathbf{X}$ are drawn from the union of K subspaces $\{\mathcal{S}_k\}_{k=1}^K$ of dimensions $\{d_k\}_{k=1}^K$.

Under the data self representation principle, each data point in data can be written as a linear combination of other data points, i.e., $\mathbf{X} = \mathbf{XZ}$, where $\mathbf{Z} \in \mathbb{R}^{N \times N}$ is a matrix of similarity coefficients. The LRR model (Liu, Lin, and Yu 2010) is formulated as

$$\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{E}\|_F^2 + \lambda \|\mathbf{Z}\|_*, \text{ s.t. } \mathbf{X} = \mathbf{XZ} + \mathbf{E}, \tag{5}$$

where $\mathbf{E}$ is the error resulting from the self representation. F-norm can be changed to other norms e.g. $\ell_{2,1}$ as done in the original LRR model. LRR takes a holistic view in favor of a coefficient matrix in the lowest rank, measured by the nuclear norm $\|\cdot\|_*$.

3.2 LRR on PGM

Let $\mathcal{X}^0 = \{[X_1], [X_2], ..., [X_N]\}$ be a set of given PGM points from $\mathcal{F}_{n:p_1,..,p_M}$ and $[X_i]$ can be represented by a set of orthogonal bases $\{X_i^1, X_i^2, ..., X_i^M\}$ such that the basis matrix $X_i^m \in \mathcal{G}(p_m, d)$. To generalize the LRR model (5) for the dataset $\mathcal{X}^0$, we first note that in (5)

$$\|\mathbf{E}\|_F^2 = \|\mathbf{X} - \mathbf{XZ}\|_F^2 = \sum_{i=1}^{N} \|\mathbf{x}_i - \sum_{j=1}^{N} Z_{ij} \mathbf{x}_j\|^2,$$

where the measure $\|\mathbf{x}_i - \sum_{j=1}^N Z_{ij}\mathbf{x}_j\|$ is the Euclidean distance between the point $\mathbf{x}_i$ and its linear combination of all the other data points including $\mathbf{x}_i$. So on PGM we propose the following form of LLR,

$$\min_{Z} \sum_{i=1}^{N} \left\| [X_i] \ominus (\biguplus_{j=1}^{N} Z_{ij} \odot [X_j]) \right\|_{\mathcal{PG}} + \lambda \|Z\|_*, \tag{6}$$

where $\left\| [X_i] \ominus (\biguplus_{j=1}^N Z_{ij} \odot [X_j]) \right\|_{\mathcal{PG}}$ with the operator $\ominus$ representing the manifold distance between $[X_i]$ and its reconstruction $\biguplus_{j=1}^N Z_{ij} \odot [X_j]$. So to get LRR model on PGM, one should define a proper distance and a combination operation in the manifold.

From the geometric property of Grassmann manifold, we can use the metric of Grassmann manifold and the PGM in (2) and (3) to replace the manifold distance in (6), i.e. $\left\| [X_i] \ominus (\biguplus_{j=1}^N Z_{ij} \odot [X_j]) \right\|_{\mathcal{PG}} = d_{\mathcal{PG}}([X_i], \biguplus_{j=1}^N Z_{ij} \odot [X_j])$. Additionally, from the mapping in (1), the mapped points in $Sym(d)$ are positive definite matrices, so they have the linear combination operation like that in Euclidean space if the coefficients are positive. So it is intuitive to replace the Grassmann points with its mapped points to implement the combination in (6), i.e.

$$\biguplus_{j=1}^{N} Z_{ij} \odot [X_j] = \mathcal{X} \times_4 Z_i,$$

where Z_i is a vector of $(Z_{i1}, ..., Z_{iN})^T$ and $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, ..., \mathcal{X}_N\}$ is a 4-order tensor, including a set of 3-order tensors $\mathcal{X}_i$ which stacks all mapped symmetric matrices along the 3rd mode, i.e. $\mathcal{X}_i = \{X_i^1 (X_i^1)^T, X_i^2 (X_i^2)^T, ..., X_i^M (X_i^M)^T\} \subset \text{Sym}(d)$. Up to now, we can construct the LRR model on PGM as follows,

$$\min_{\mathbf{E},Z} \|\mathbf{E}\|_F^2 + \lambda \|Z\|_* \quad \text{s.t.} \quad \mathcal{X} = \mathcal{X} \times_4 Z + \mathbf{E}. \tag{7}$$

We name the above model PGLRR.

3.3 Algorithms for LRR on PGM

To avoid any complex calculation between the 4-order tensor and a matrix in (7), we carefully analyze the reconstruction tensor error $\mathbf{E}$ and translate the optimization problem into an equivalent and solvable optimization model.

We consider the slice E_i of $\mathbf{E}$ in (7). $\|E_i\|_F^2$ is re-written as the following form:

$$\|E_i\|_F^2 = \sum_{m=1}^{M} \|(X_i^m X_i^{mT} - \sum_{j=1}^{N} Z_{ij}(X_j^m X_j^{mT}))\|_F^2.$$

To simplify the expression for $\|E_i\|_F^2$, note that the matrix property $\|A\|_F^2 = \text{tr}(A^T A)$ and denote

$$\Delta_{ij}^m = \text{tr}[(X_j^{mT} X_i^m)(X_i^{mT} X_j^m)]. \quad (8)$$

Clearly $\Delta_{ij}^m = \Delta_{ji}^m$ and we define M $N \times N$ symmetric matrices

$$\Delta^m = (\Delta_{ij}^m)_{i=1,j=1}^N, \quad m = 1, 2, ..., M. \quad (9)$$

With some algebraic manipulation, it is not hard to prove

$$\|\mathbf{E}\|_F^2 = -2\text{tr}(Z\Delta) + \text{tr}(Z\Delta Z^T) + const, \quad (10)$$

where $\Delta = \sum_{m=1}^{M} \Delta^m$ and $const$ collects all the terms irrelevant to the variable Z. Similar to (Wang et al. 2014), we can prove that Δ is semi-definite positive in the latter Appendix. As a result, we have the eigenvector decomposition for Δ defined by $\Delta = UDU^T$, where $U^T U = I$ and $D = \text{diag}(\sigma_i)$ with non-negative eigenvalues σ_i. Thus, (10) can be converted to its equivalent form $\|\mathbf{E}\|_F^2 = \|Z\Delta^{\frac{1}{2}} - \Delta^{\frac{1}{2}}\|_F^2 + \text{const}$.

After variable elimination, (7) can be converted to

$$\min_Z \|Z\Delta^{\frac{1}{2}} - \Delta^{\frac{1}{2}}\|_F^2 + \lambda\|Z\|_*. \quad (11)$$

Problem (11) has a closed form solution given in the following theorem (Favaro, Vidal, and Ravichandran 2011).

Theorem 1 *Given that $\Delta = UDU^T$ as defined above, the solution to* (11) *is given by*

$$Z^* = UD_\lambda U^T,$$

where D_λ is a diagonal matrix with its i-th element defined by

$$D_\lambda(i,i) = \begin{cases} 1 - \frac{\lambda}{\sigma_i} & \text{if } \sigma_i > \lambda, \\ 0 & \text{otherwise}. \end{cases}$$

We briefly conclude the main procedures of our proposed algorithm in Algorithm 1.

3.4 Complexity Analysis of PGLRR

If we denote the rank of coefficient matrix Z by R and the number of iterations by s, for the N PGM samples generated from M Grassmann manifolds, the complexity of the proposed PGLRR algorithm (Algorithm 1) can be mainly divided into two parts: the data representation part (step 2-11) and the solution to the algorithm part (step 12-13). In the formal part, the trace norm should be calculated to get the new coefficient matrix Δ. The complexity of Δ computation is $\mathcal{O}(MN^2)$; In the second part, we perform a partial SVD method to solve the final coefficient matrix Z, whose computation complexity is $\mathcal{O}(RN^2)$. Overall, for the s iterations, the computation complexity of our proposed method is $\mathcal{O}(MN^2) + \mathcal{O}(sRN^2)$.

Algorithm 1 The whole procedures about Problem (7).

Input: The Product Grassmann sample set $\{[X_i]\}_{i=1}^N$, $[X_i] \in \mathcal{PG}_{n:p_1,..,p_M}$ and the balancing penalty parameter λ.

Output: The Low-Rank Representation Z

1: Initialize:$J = Z = 0, A = B = 0, \mu = 10^{-6}, \mu_{max} = 10^{10}$ and $\varepsilon = 10^{-8}$
2: **for** m=1:M **do**
3: **for** i=1:N **do**
4: **for** j=1:N **do**
5: $\Delta_{ij}^m \leftarrow \text{tr}[(X_j^{mT} X_i^m)(X_i^{mT} X_j^m)]$;
6: **end for**
7: **end for**
8: **end for**
9: **for** m=1:M **do**
10: $\Delta \leftarrow \Delta + \Delta_{::}^m$;
11: **end for**
12: Performing SVD on Δ
 $\Delta \leftarrow UDU^T$
13: Calculating the coefficient matrix Z by
 $Z \leftarrow UD_\lambda U^T$

4 Kernelized LRR on Product Grassmann Manifold

4.1 Kernels on PGM

The LRR model on PGM (7) can be regarded as a kernelized LRR with a kernel feature mapping $\prod$ defined by (1). It is not surprised that Δ is semi-definite positive as it serves as a kernel matrix. It is natural to further generalize the PGLRR based on kernel functions on PGM.

A straightforward way to define a kernel function on PGM is to use the kernel functions on Grassmann manifolds such as Canonical Correlation Kernel and Projection Kernel (Harandi et al. 2011). Consider any two Product Grassmann points $[X_i] = \{X_i^1, ..., X_i^M\}$ and $[X_j] = \{X_j^1, ..., X_j^M\}$ where X_i^m and X_j^m $(m = 1, 2, ..., M)$ are Grassmann points respectively. We define a kernel $K([X_i], [X_j])$ as follows

$$K([X_i], [X_j]) = \sum_{m=1}^{M} k(X_i^m, X_j^m), \quad (12)$$

where k is any kernel on Grassmann manifold. For the simplicity of expression, we denote $\mathcal{K}_{ij}^m = k(X_i^m, X_j^m)$.

4.2 Kernelized LRR on PGM

Let k be any kernel function on Grassmann manifold. For an example, here we use the largest canonical correlation kernel (Yamaguchi, Fukui, and Maeda 1998). According to the kernel theory (Schölkopf and Smola 2002), there exists a feature mapping $\phi : \mathcal{G}(p,n) \to \mathcal{F}$, where $\mathcal{F}$ is the relevant feature space under the given kernel k.

Given a set of points $\{[X_1], [X_2], ..., [X_N]\}$ on PGM $\mathcal{F}_{n:p_1...p_M}$, we define the following LRR model

$$\min_Z \|\mathcal{E}\|_F^2 + \lambda\|Z\|_* \quad \text{s.t.} \quad \phi(\mathcal{X}) = \phi(\mathcal{X})_{\mathcal{X}_\oplus} Z + \mathcal{E}, \quad (13)$$

where $\phi(\mathcal{X}) = \{\phi([X_1]), \phi([X_2]), ..., \phi([X_N])\}$ denotes the "tensor" on feature spaces $\mathcal{F}$s and $\mathcal{X}_{\bigoplus}$ denotes the tensor mode multiplication in the last mode (or data mode). We call the above model KPGLRR.

4.3 Algorithm for KPGLRR

By using the similar derivation in PGLRR algorithm, we can prove that the model (13) is equivalent to

$$\min_{Z} -2\text{tr}(Z\mathcal{K}) + \text{tr}(Z\mathcal{K}Z^T) + \lambda\|Z\|_*, \tag{14}$$

where $\mathcal{K}$ is an $N \times N$ kernel matrix over all the data points $[X_i]$s, $\mathcal{K} = \sum_{m=1}^{M} \mathcal{K}^m$ and $\mathcal{K}^m = (\mathcal{K}^m_{ij})^N_{i=1,j=1}$. Clearly the symmetric matrix $\mathcal{K}$ is positive semi-definite. Finally (14) can be re-written as

$$\min_{Z} \|Z\mathcal{K}^{\frac{1}{2}} - \mathcal{K}^{\frac{1}{2}}\|^2_F + \lambda\|Z\|_*, \tag{15}$$

where $\mathcal{K}^{\frac{1}{2}}$ is the square root matrix of kernel matrix $\mathcal{K}$. A closed solution to (15) is formed as the way for (11).

5 Experiments

We evaluate our proposed PGLRR and KPGLRR methods on the following public datasets: MNIST Handwritten dataset[1], CMU-PIE dataset[2], ALOI dataset[3], SKIG dataset[4], Highway Traffic dataset[5].

Our methods are assessed against the following clustering methods:

- **Sparse Subspace Clustering (SSC) (Elhamifar and Vidal 2013)** finding the sparsest representation for the data set using l_1 approximation.
- **Low Rank Representation (LRR) (Liu et al. 2013)** revealing the global intrinsic structure of the data by its lowest rank representation.
- **Low Rank Representation on Grassmann Manifold (FGLRR) (Wang et al. 2014)** representing the image sets as Grassmann manifold pionts and constructing LRR model on Grassmann manifold.
- **Statistical computations on Grassmann and Stiefel manifolds (SCGSM) (Turaga et al. 2011)** computing the Riemannian geometry of the Grassmann and Stiefel manifold by statistical methods.
- **Sparse Manifold Clustering and Embedding (SMCE) (Elhamifar and Vidal 2011)** constructing proper metric by considering the local geometry of manifold.
- **Clustering on Grassmann Manifold via Kernel Embedding (CGMKE) (Shirazi et al. 2012)** embedding the Grassmann manifold into a Hilbert space where a measure of clustering distortion is minimised.

[1]http://yann.lecun.com/exdb/mnist/.

[2]http://vasc.ri.cmu.edu/idb/html/face/.

[3]http://aloi.science.uva.nl/.

[4]http://lshao.staff.shef.ac.uk/data/SheffieldKinectGesture.htm.

[5]http://www.svcl.ucsd.edu/projects/.

Among them, FGLRR, SCGSM, SMCE and CGMKE are related to clustering on manifolds. As the conventional SSC and LRR methods implement clustering in linear space, the Grassmann points cannot be used as inputs for SSC and LRR. To construct a fair comparison, we "vectorize" them into a long vector with all the raw data in each image set, in a carefully chosen order, e.g., in the frame order. As the dimension of such vectors is usually too high, we apply PCA to reduce the raw vectors to a low dimension which equals to the number of PCA components retaining 95% of its variance energy.

Our experiments are coded in Matlab 2014a and implemented on a machine with Intel Core i7-4770K 3.5GHz CPU. All color images are converted into gray images and normalized with mean zero and unit variance.

5.1 MNIST Handwritten digits Clustering

The MNIST dataset consists of about 70,000 digit images written by 250 volunteers. All the digit images have been size-normalized and centered in a fixed size of 28×28. This dataset can be regarded as synthetic as the data are clean.

This dataset has 10 classes. To test the robustness of the proposed methods, we construct the image set of one digit contains other digits images. For each digit class, we randomly select 9 images from its samples and randomly select 1 image from other 9 digit classes to construct an image set, i.e. $\mathbf{S} = \{S_1, ..., S_9\}$ and $S_m = \{I^m_1, I^m_2\}, m = 1, ..., 9$, where the first image I^m_1 is selected from the same digit class, and the second I^m_2 is from the other digit classes which are noise. Then the subset S_m is represented as a Grassmann point, i.e. $X^m \in \mathcal{G}(2, 748)$ ($p_m = 2$, $d = 28 \times 28 = 748$). Therefore, we construct a Product Grassmann point of the digit class as $[X] = \{X^1, ..., X^9\} \in \mathcal{PG}_{748:2,2,2,2,2,2,2,2,2}$. Thus 9 underlying varying factors are simulated in this case. Here the number of samples of each cluster are set to 20, 30, 40, 50 to construct test datasets.

For FGLRR, SCGSM, SMCE and CGMKE methods, Grassmann points are directly used as the input. For SSC and LRR methods, the original vectors with dimension $28 \times 28 \times 18 = 14112$ are reduced to dimension of {162, 234, 302, 366} for the different scales by PCA, respectively.

The experiment results are reported in Figure 2. It is shown that the accuracy of our proposed algorithms outperform other methods almost 20 percents for different scales of test sets. We conclude that PGM representation has capacity in extracting the common features crossing a number of varying factors as shown in Figure 3. Thus the combination of Product Grassmann geometry and LRR model brings better accuracy for NCut clustering.

5.2 ALOI Object Clustering

The ALOI dataset collects 1000 objects with simple background and each object has over 100 images captured under four different conditions: 72 views, 24 light directions, 12 illuminations and 4 stereos. Figure 4(a) shows some samples. We down-sampling image size to 48×32.

In this experiment, we use the images of $C(= 5, 10, 15, 20, 25, 30)$ objects. We select 4 images with different light directions and 14 images with different views

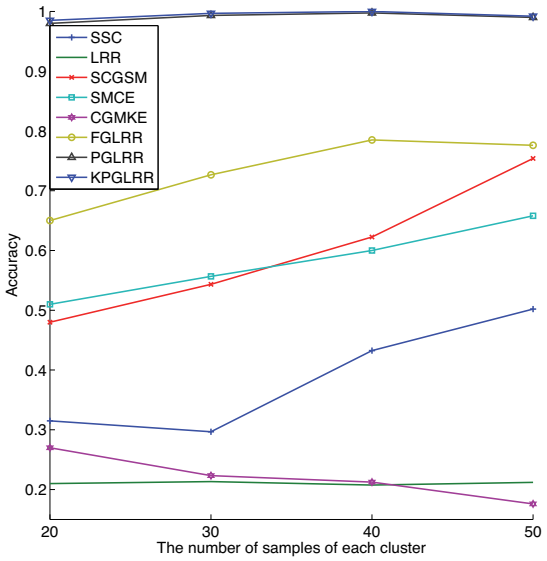

Figure 2: The experimental results on MNIST Datasets.

Figure 3: Demonstration of a point of Product Grassmann manifold. The first row is the first dimension of the Grassmann points, the second row is the second dimension.

(a) (b)

Figure 4: (a) Some samples of ALOI dataset and (b) the Learned Similarity Matrix.

to construct an image set, i.e. $\mathbf{S} = \{S_1, S_2\}$ ($M = 2$). Then S_1, S_2 are represented as Grassmann points as $X^1 \in \mathcal{G}(2, 1536)$ and $X^2 \in \mathcal{G}(3, 1536)(d = 1536, p_1 = 2, p_2 = 3)$. Therefore, we could create a PGM point $[X] = \{X^1, X^2\} \in \mathcal{PG}_{1536:2,3}$ to represent an image set. For each object, we generate 5 image sets. For SSC and LRR methods, the original vectors with dimension $48 \times 32 \times (2+3) = 7680$ reduced to {14,30,46,58,73,87} for the different C by PCA, respectively.

The experimental results are shown in Figure 5. It indicates that SMCE, FGLRR and our methods perform excellently, but our methods are more stable when the cluster number is increasing. Figure 4(b) shows an affinity matrix generated from PGLRR for C=10, which is an obvious block diagonal matrix.

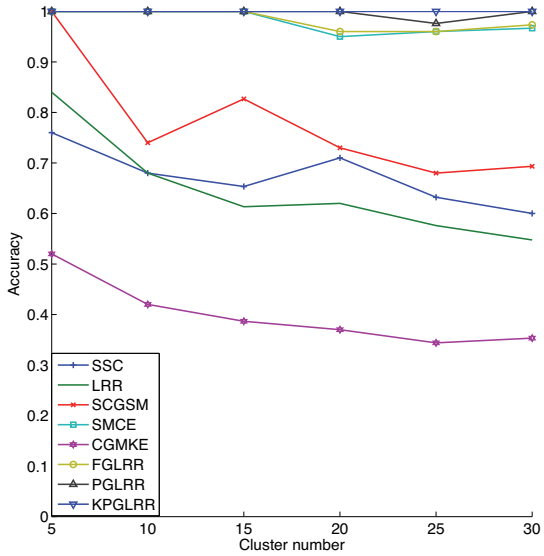

Figure 5: The experimental results on ALOI Datasets.

5.3 CMU-PIE Face Clustering

The CMU-PIE database contains facial images of 68 persons captured under 13 poses, 43 illuminations and with 4 different expressions. Here the images were cropped to attain the face region. The cropped images have been down-sampled to 32×32 pixels. Some samples of this dataset are shown in Figure 6.

Figure 6: The CMU PIE face samples. The three rows are facial images with glasses, different illuminations and poses respectively.

Since the expression variation is not very obvious in this dataset, we use the images with glasses instead of expression variation to implement clustering. We select images of 7 persons who have glasses. For the images of one person, we select 4 images with different illuminations, 8 images with

different poses and 5 images with glasses to construct an image set, i.e. $\mathbf{S} = \{S_1, S_2, S_3\}$ ($M = 3$). Then S_1, S_2, S_3 are represented as Grassmann points as $X^1 \in \mathcal{G}(3, 1024)$, $X^2 \in \mathcal{G}(3, 1024)$ and $X^3 \in \mathcal{G}(4, 1024)$. Therefore, we could create a PGM point $[X] = \{X^1, X^2, X^3\} \in \mathcal{PG}_{1024:3,3,4}$ to represent an image set. Each person generates 5 image sets. For SSC and LRR, the original vectors with dimension $32 \times 32 \times 17 = 17408$ are reduced to 22 by PCA. Table 1 presents the clustering result of all algorithms.

Methods \ Datasets	CMU-E	TRAFFIC
SSC	0.4286	0.6285
LRR	0.6000	0.6838
SCGSM	**1**	0.7787
SMCE	**1**	0.5573
CGMKE	0.4857	0.5652
FGLRR	**1**	0.7984
PGLRR	**1**	0.8379
KPGLRR	**1**	**0.8458**

Table 1: The clustering results on the CMU-PIE dataset and TRAFFIC dataset.

5.4 SKIG Action Video Clustering

The SKIG dataset (Liu and Shao 2013) contains 1080 RGB-D sequences and this dataset stores ten kinds of gestures of six persons. All the gestures are performed by fist, finger and elbow respectively under three backgrounds (wooden board, white plain paper and news paper) and two illuminations (strong and poor light). Each RGB-D sequence contains 63 to 605 frames. These images are normalized to 24×32. Figure 7 shows some RGB images and its DEPTH images of ten actions.

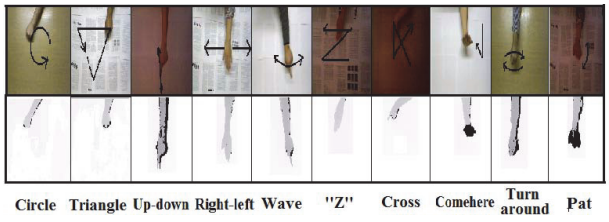

Figure 7: The SKIG samples.First row is the RGB images of ten actions. Second row is the DEPTH images of ten actions.

We design different types of PGM points with different combinations of factors, including: illumination + depth sequences ($[X] = \{X^1, X^2\} \in \mathcal{PG}_{1024:20,20}$); illumination + dark background sequences ($[X] = \{X^1, X^2\} \in \mathcal{PG}_{1024:20,20}$); fist + finger + elbow sequences ($[X] = \{X^1, X^2, X^3\} \in \mathcal{PG}_{1024:20,20,20}$). For each PGM type, we select 54 samples for one of the 10 clusters.

Since there is a big gap between 63 to 405 frames among SKIG sequences and both SSC and LRR require input data in the same dimension, we give up comparing our methods with SSC and LRR. Table 2 shows our methods have better performance.

Methods	light+depth	light+dark	fist+index+flat
SCGSM	0.4093	0.4667	0.3806
SMCE	0.4481	0.4130	0.4639
CGMKE	0.1796	0.1648	0.1778
FGLRR	0.5648	0.5185	0.4944
PGLRR	0.5833	0.5963	0.5056
KPGLRR	**0.5907**	**0.6000**	**0.5194**

Table 2: The clustering results on the SKIG dataset.

5.5 UCSD Traffic video Clustering

The Traffic dataset contains 253 video sequences of highway with three traffic levels: light, medium and heavy, in various weather scenes. Each video sequence has 42 to 52 frames. Each image is normalized to size 24×24.

Each video containing frames at different traffic levels may be assigned one particular traffic level. In other word, there are outliers in an image set. So we split each video into a set of M short clips, some clips may contains a few outliers. We represent each clip as a Grassmann point, thus a video can be regarded as a PGM point. In our experiments we set $M = 3$ with roughly equal number of frames for each clip. The constructed PGM point ($[X] = \{X^1, X^2, X^3\} \in \mathcal{PG}_{1024:6,6,6}$).

In SSC and LRR methods, the dimension $24192 (= 24 \times 24 \times 42)$ of the raw data is reduced to 147 by PCA. Table 1 presents the clustering results. The accuracy of our methods is obviously at least 4% higher than the other methods.

6 Conclusion

In this paper, we proposed a data representation method based on PGM. By exploiting the metric on the manifold, the LRR based subspace clustering method is extended to the PGLRR model. An efficient algorithm is also proposed for PGLRR. Additionally, the LRR model on PGM is generalized in a kernel framework. The high performance in the clustering experiments on different image sets and video databases indicates that PGLLR is well suitable for representing non-linear high dimensional data with multiple varying factors and revealing their intrinsic multiple subspaces structures underlying the data. In the future work, we will focus on investigating different metrics of PGM and test these methods on large scale complex image sets.

Acknowledgements

The research project is supported by the Australian Research Council (ARC) through the grant DP140102270 and also partially supported by National Natural Science Foundation of China under Grant No.61390510, 61133003, 61370119, 61171169, 61227004, 61300065, Beijing Natural Science Foundation No.4132013, 4142010 and Funding Project for Academic Human Resources Development in Institutions of Higher Learning Under the Jurisdiction of Beijing Municipality(PHR(IHLB)).

References

Absil, P.; Mahony, R.; and Sepulchre, R. 2008. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.

Chen, G., and Lerman, G. 2009. Spectral curvature clustering. *IJCV* 81(3):317–330.

Donoho, D. 2004. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. *Comm. Pure and Applied Math.* 59:797–829.

Elhamifar, E., and Vidal, R. 2011. Sparse manifold clustering and embedding. NIPS.

Elhamifar, E., and Vidal, R. 2013. Sparse subspace clustering: Algorithm, Theory, and Applications. *IEEE TPAMI* 35(1):2765–2781.

Favaro, P.; Vidal, R.; and Ravichandran, A. 2011. A closed form solution to robust subspace estimation and clustering. In *CVPR*, 1801–1807.

Gruber, A., and Weiss, Y. 2004. Multibody factorization with uncertainty and missing data using the EM algorithm. In *CVPR*, volume I, 707–714.

Harandi, M. T.; Sanderson, C.; Shirazi, S. A.; and Lovell, B. C. 2011. Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching. In *CVPR*, 2705–2712.

Harandi, M. T.; Sanderson, C.; Shen, C.; and Lovell, B. 2013. Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution. In *ICCV*, 3120–3127.

Helmke, J. T., and Hüper, K. 2007. Newton's method on Grassmann manifolds. Technical report, Preprint: [arXiv:0709.2205].

Ho, J.; Yang, M. H.; Lim, J.; Lee, K.; and Kriegman, D. 2003. Clustering appearances of objects under varying illumination conditions. In *CVPR*, volume 1, 11–18.

Hong, W.; Wright, J.; Huang, K.; and Ma, Y. 2006. Multi-scale hybrid linear models for lossy image representation. *IEEE TIP* 15(12):3655–3671.

Kanatani, K. 2001. Motion segmentation by subspace separation and model selection. In *ICCV*, volume 2, 586–591.

Lang, C.; Liu, G.; Yu, J.; and Yan, S. 2012. Saliency detection by multitask sparsity pursuit. *IEEE TPAMI* 21(1):1327–1338.

Lerman, G., and Zhang, T. 2011. Robust recovery of multiple subspaces by geometric l_p minimization. *The Annuals of Statistics* 39(5):2686–2715.

Liu, L., and Shao, L. 2013. Learning discriminative representations from rgb-d video data. In *IJCAI*.

Liu, G., and Yan, S. 2011. Latent low-rank representation for subspace segmentation and feature extraction. In *ICCV*, 1615–1622.

Liu, G.; Lin, Z.; Sun, J.; Yu, Y.; and Ma, Y. 2013. Robust recovery of subspace structures by low-rank representation. *IEEE TPAMI* 35(1):171–184.

Liu, G.; Lin, Z.; and Yu, Y. 2010. Robust subspace segmentation by low-rank representation. In *ICML*, 663–670.

Ma, Y.; Yang, A.; Derksen, H.; and Fossum, R. 2008. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM Review* 50(3):413–458.

Roweis, S., and Saul, L. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290(1):2323–2326.

Schölkopf, B., and Smola, A. 2002. *Learning with Kernels*. Cambridge, Massachusetts: The MIT Press.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE TPAMI* 22(1):888–905.

Shirazi, S.; Harandi, M.; Sanderson, C.; Alavi, A.; and Lovell, B. 2012. Clustering on grassmann manifolds via kernel embedding with application to action analysis. In *ICIP*. 781–784.

Tenenbaum, J.; Silva, V.; and Langford, J. 2000. A global geometric framework for nonlinear dimensionality reduction. *Optimization Methods and Software* 290(1):2319–2323.

Tipping, M., and Bishop, C. 1999. Mixtures of probabilistic principal component analyzers. *Neural Computation* 11(2):443–482.

Tseng, P. 2000. Nearest q-flat to m points. *Journal of Optimization Theory and Applications* 105(1):249–252.

Turaga, P.; Veeraraghavan, A.; Srivastava, A.; and Chellappa, R. 2011. Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. *IEEE TPAMI* 33(11):2273–2286.

Tuzel, O.; Porikli, F.; and Meer, P. 2006. Region covariance: A fast descriptor for detection and classification. *In ECCV* 3952:589–600.

Vidal, R. 2011. Subspace clustering. *IEEE Signal Processing Magazine* 28(2):52–68.

von Luxburg, U. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17(4):395–416.

Wang, B.; Hu, Y.; Gao, J.; Sun, Y.; and Yin, B. 2014. Low rank representation on Grassmann manifolds. In *ACCV*.

Wang, J.; .Saligrama, V.; and nón, D. 2011. Structural similarity and distance in learning. *http://arxiv.org/pdf/1110.5847.pdf*.

Xu, R., and Wunsch-II, D. 2005. Survey of clustering algorithms. *IEEE TNN* 16(2):645–678.

Yamaguchi, O.; Fukui, K.; and Maeda, K. 1998. Face recognition using temporal image sequence. In *Automatic Face and Gesture Recognition*, 318–323.

Appendix

Lemma 1 *Given a set of 3-order tensors* $\{\mathcal{X}_1, \mathcal{X}_2, ..., \mathcal{X}_N\}$ *and each tensor contains* M *matrices,* $\mathcal{X}_i = \{X_i^1, X_i^2, ..., X_i^M\}$ *where* $X_i^{m^T} X_i^m = I_d$, *if* $\Delta = \sum_{m=1}^{M} \Delta^m = [\sum_{m=1}^{M} \Delta_{ij}^m]_{i,j=1}^N \in R^{N\times N}$ *with element* $\Delta_{ij}^m = tr\left[(X_j^{m^T} X_i^m)(X_i^{m^T} X_j^m)\right]$, *then the matrix* Δ *is semi-positive definite.*

Proof: Denote by $B_i^m = X_i^m X_i^{m^T}$. Then B_i^m is a symmetric matrix of size $d \times d$. Then

$$\begin{aligned}\Delta_{ij}^m &= \text{tr}\left[(X_j^{m^T} X_i^m)(X_i^{m^T} X_j^m)\right] = \text{tr}\left[(X_j^m X_j^{m^T})(X_i^m X_i^{m^T})\right] \\ &= \text{tr}[B_j^m B_i^m] = \text{tr}[B_j^m B_i^{m^T}] = \text{tr}[B_i^{m^T} B_j^m] \\ &= \text{vec}(B_i^m)^T \text{vec}(B_j^m)\end{aligned}$$

where $\text{vec}(\cdot)$ is the vectorization of a matrix.

Define a matrix $B^m = [\text{vec}(B_1^m), \text{vec}(B_2^m), ..., \text{vec}(B_N^m)]$. Then it is easy to show that

$$\Delta^m = [\Delta_{ij}^m]_{i,j=1}^N = [\text{vec}(B_i^m)^T \text{vec}(B_j^m)]_{i,j=1}^N = B^{m^T} B^m.$$

So Δ^m is a semi-positive definite matrix. Obviously,

$$\Delta = \sum_{m=1}^{M} \Delta^m = \sum_{m=1}^{M} B^{m^T} B^m$$

is also a semi-positive definite matrix.