

LETTER 

---

 Communicated by Daniel Lee

## Supervised Dimensionality Reduction on Grassmannian for Image Set Recognition

**Tianci Liu**

*liutianci@sia.cn*

*Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China; University of Chinese Academy of Sciences, Beijing 100049, China; Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences, Shenyang 110016, China; and Key Lab of Image Understanding and Computer Vision, Liaoning Province, Shenyang 110016, China*

**Zelin Shi**

*582507583@qq.com*

**Yunpeng Liu**

*1099149944@qq.com*

*Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China; Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences, Shenyang 110016, China; and Key Lab of Image Understanding and Computer Vision, Liaoning Province, Shenyang 110016, China*

Modeling videos and image sets by linear subspaces has achieved great success in various visual recognition tasks. However, subspaces constructed from visual data are always notoriously embedded in a high-dimensional ambient space, which limits the applicability of existing techniques. This letter explores the possibility of proposing a geometry-aware framework for constructing lower-dimensional subspaces with maximum discriminative power from high-dimensional subspaces in the supervised scenario. In particular, we make use of Riemannian geometry and optimization techniques on matrix manifolds to learn an orthogonal projection, which shows that the learning process can be formulated as an unconstrained optimization problem on a Grassmann manifold. With this natural geometry, any metric on the Grassmann manifold can theoretically be used in our model. Experimental evaluations on several data sets show that our approach results in significantly higher accuracy than other state-of-the-art algorithms.

## 1 Introduction

---

Representing the set of images of objects as linear subspaces has remained a subject of interest because of variabilities arising from changes in pose, lighting, expression, and other physical parameters. Although the dimension of the subspaces is typically not large, those subspaces constructed from visual data always exist in notoriously high-dimensional Euclidean space. The computational complexity of this high-dimensional ambient space limits the applicability of existing techniques. Moreover, linear subspaces with the same dimensionality reside on a special type of Riemannian manifold, the Grassmann manifold, which has a nonlinear structure. A Grassmann manifold (denoted by  $G(n, D)$ , where  $n < D$ ) is the set of  $n$ -dimensional linear subspaces in the  $D$ -dimensional Euclidean space  $\mathbb{R}^D$ , which is the compact Riemannian manifold with  $n(D - n)$  dimensionality. However, conventional methods of dimensionality reduction (DR), such as principal component analysis (PCA; Holland, 2008) and linear discriminant analysis (LDA; Izenman, 2013), are devised for vectors in a flat Euclidean space instead of a curved Riemannian space. On account of the high dimensionality of the Grassmann manifold derived from visual data, simply applying the conventional algorithms designed for data in vector spaces to subspaces for dimensionality reduction may lead to a distortion in the geometry. In response to this issue, this letter proposes a method of dimensionality reduction on Grassmannian to learn a low-dimensional and more discriminative Grassmann manifold for higher computational efficiency and better classification performance. Moreover, the class labels are used to encode a more discriminative structure in the low-dimensional manifold from the pairwise relationship of the original data in the dimensionality reduction.

Recently, some related work concerning the Grassmann manifold has appeared. Grassmann discriminant analysis (GDA) (Hamm & Lee, 2008) was the first to propose a Grassmann framework that embeds the Grassmann manifold into a reproducing kernel Hilbert space by learning a projection kernel and then performs classification via linear discriminant analysis (LDA). Based on GDA, graph-embedding Grassmann discriminant analysis (GGDA) (Harandi, Sanderson, Shirazi, & Lovell, 2011) proposed a graph-embedding framework and a new Grassmannian kernel to learn a more discriminatory mapping on Grassmannian manifolds. By combining sparse coding and dictionary learning on Grassmann manifolds, Grassmann dictionary learning (GDL) (Harandi, Sanderson, Shen, & Lovell, 2013) updated a Grassmann dictionary under the projection embedding and proposed a kernelized version to solve the nonlinearity in the data.

Nevertheless, the limitations of these methods are obvious. First, the most important part is to find a desirable kernel function that is positive definite to satisfy Mercer's theorem allowing the valid reproducing kernel Hilbert space to be generated. Second, the embedded data in

higher-dimensional Hilbert space will cause distortions by flattening the Grassmann manifold. Furthermore, the kernel function measures only similarity, not distance, and the computational cost is excessive, with large numbers of data samples.

Several studies have investigated the mapping from manifold to manifold directly, an approach that has attracted increasing attention. Harandi, Salzmann, and Hartley (2017) first learned a mapping with an orthonormal projection from a high-dimensional symmetric positive-definite (SPD) manifold to a lower-dimensional and more discriminative SPD manifold. Projection metric learning on a Grassmann manifold (PML) in Huang, Wang, Shan, and Chen (2015) learned a Mahalanobis-like matrix on a symmetric positive-semidefinite manifold to seek a lower-dimensional and more discriminative Grassmann manifold under the projection framework by embedding Grassmann manifolds onto the space of symmetric matrices.

However, to the best of our knowledge, there is no general framework of a dimensional-reduction model for the Grassmann manifold that can be combined with other Grassmannian-based recognition algorithms. Based on this research gap, we propose a generalized supervised dimensionality-reduction method on Grassmannian with various metrics. Our algorithm can also be regarded as an enhanced preprocessing algorithm that learns a lower-dimensional and more discriminative Grassmann manifold. Note that our framework is suitable for any metric on the Grassmann manifold instead of being limited to the typical projection framework. This letter makes three contributions:

- We explore the possibility of proposing a Riemannian geometry-based framework to construct lower-dimensional subspaces with maximum discriminative power from high-dimensional subspaces in the supervised scenario.
- For the essential metrics used in our method, we introduce five metrics on the Grassmann manifold and derive the corresponding formulas that are required to calculate the gradient in our model. In certain complicated cases where computing the gradient is difficult for some metrics, we use the matrix chain rule to resolve this issue.
- We propose a more general, more extended, and more complete Grassmannian framework for various metrics on the Grassmannian.

The rest of this letter is organized as follows. In section 2, we briefly introduce the notions of the Grassmann manifold and Riemannian metrics on it. Then the proposed method and the formulations for calculating the gradients are derived in section 3. In section 4, we describe several experiments conducted to demonstrate the competitive performance of our approach compared with those of the state-of-the-art algorithms and provide a detailed discussion of our algorithm. We conclude in section 5.

Table 1: Different Measures on the Grassmann Manifold.

Measure Name	Mathematical Expression	Metric/ Distance	Kernel
Projection F-norm	$d_{pro}(X_1, X_2) = 2^{-\frac{1}{2}} \ X_1 X_1^T - X_2 X_2^T\ _F$	✓	×
Fubini-Study	$d_{FS}(X_1, X_2) = \arccos \det(X_1^T X_2) $	✓	×
Binet-Cauchy distance	$d_{BC}^2(X_1, X_2) = 2 - 2 \det(X_1^T X_2) $	✓	×
Projection kernel distance	$d_{pk}^2(X_1, X_2) = 2n - 2\ X_1^T X_2\ _F^2$	✓	×
Binet-Cauchy kernel	$d_{BCK}^2(X_1, X_2) = \det(X_1^T X_2 X_2^T X_1)$	×	✓

Note:  $X_1, X_2$  are two points on the Grassmannian  $G(n, D)$ .

## 2 Grassmann Manifolds

The Grassmann manifold  $G(n, D)$  is a compact Riemannian manifold with  $n(D - n)$  dimensionality and its space is a non-Euclidean space. Any point on it can be represented as an orthonormal matrix  $X$  of size  $D \times n$  that spans a linear subspace  $span(X)$  by its orthonormal basis such that  $X^T X = I_n$ , where  $I_n$  is the  $n \times n$  identity matrix. From a mathematical perspective,  $G(n, D)$  can be represented by the quotient space of all the  $D \times n$  matrices with  $n$  orthogonal columns under the  $n$ -order orthogonal group  $O(n)$ :

$$G(n, D) = \{X \in \mathbb{R}^{D \times n} : X^T X = I_p\} / O(n). \quad (2.1)$$

In particular, we note that the element of  $G(n, D)$  is one linear subspace represented by  $span(X)$ . However, this type of matrix representation is not unique because the orthonormal basis of subspaces is right invariant to orthonormal matrices  $R \in O(n)$ . Consequently,  $span(X_1)$  and  $span(X_2)$  are the same if and only if  $X_1 R_1 = X_2 R_2$  for  $R_1, R_2 \in O(n)$ . Those matrices like  $X_1$  and  $X_2$  are actually the same to some extent because the same subspace  $span(X)$  is spanned by their columns. In the remainder of this letter, we denote  $X$  to represent the equivalence class  $span(X)$  for a point on the Grassmannian. Next, we introduce in Table 1 several prevalent distances on the Grassmann manifold that are widely used in the literature (Hamm & Lee, 2008; Edelman, Arias, & Smith, 1998; Harandi, Salzmann, Jayasumana, Hartley, & Li, 2014).

**Proposition 1.** *All the metrics between two subspaces are invariant to the group of orthonormal transformations. For any distance  $d(X_1, X_2)$  on the Grassmannian manifold,*

$$d(X_1, X_2) = d(X_1 R_1, X_2 R_2), \quad \forall R_1, R_2 \in O(n).$$

### 3 The Proposed Method

**3.1 Optimization on the Riemannian Manifold.** To obtain greater discriminative capacity, the initial purpose of our model is to learn a projection  $W$ , which can construct a lower Grassmann manifold  $G(n, d)$  to handle the high dimensionality of  $G(n, D)$  ( $D \gg d$ ). More specifically, assuming  $X \in G(n, D)$ ,  $Y \in G(n, d)$ , we intend to obtain reduced subspaces by a general mapping  $f : G(n, D) \rightarrow G(n, d)$  via the formula

$$f(X) = W^T X = Y, \quad (3.1)$$

where  $W \in \mathbb{R}^{D \times d}$  a column full-rank matrix.

In practice, we impose orthonormality constraints on  $W$  such that  $W^T W = I_d$ , which can avoid possible degeneracies of optimization when minimizing the cost function with regard to  $W$  and is more practical for computation. However,  $W^T X$  is not guaranteed to be on the Grassmann manifold despite  $W$  representing generally an orthogonal matrix. Thus, the QR decomposition is used to obtain the orthonormal components of  $W^T X$  s.t.  $W^T X = QR$ , where  $Q$  is the orthonormal matrix that includes the first  $d$  columns and  $R$  is the invertible upper-triangular matrix. Then we normalize  $W^T X$  by  $Q$  to guarantee the orthonormality,  $Q = W^T (XR^{-1})$ .

The purpose of conventional methods of dimensionality reduction (DR) is to obtain the information of the original data as much as possible in the space of reduced dimensionality. Without loss of generality, the dimensionality reduction of high-dimensional visual data for cheaper computational complexity will not always improve classification accuracy. In fact, conventional dimensionality-reduction algorithms always discard original information to some extent, which leads recognition accuracies to decrease. Furthermore, the geometry precondition of DR methods is based on the data residing in the vector space. However, for non-Euclidean data such as subspaces, which are widely used in the image set recognition task, how can novel DR algorithm that can improve recognition accuracy be designed?

Our motivation toward this goal has two aspects: to reduce the computational complexity through the DR framework and to simultaneously improve the discriminative power of the data. Regarding this goal, equation 3.1 is simply a DR projection that meets only the first aspect of our motivation. To achieve the latter aspect, we hope that the transformation  $W$  will encode more effective information from the original data. Based on these two aspects, we introduce the objective function in our model designed for the classification of image sets  $\Gamma = \{X_1, X_2, \dots, X_N\}$ ,

$$L(W) = \sum_{i,j} G(i, j) \cdot d(W^T X_1, W^T X_2), \quad (3.2)$$

where  $d : M \times M \rightarrow R^+$  represents  $d_{pro}$ ,  $d_{FS}$ ,  $d_{BC}^2$ ,  $d_{BCK}$  or  $d_{pk}$  and  $G(i, j)$  is computed based on class similarity,  $G(n, D) \times G(n, D) \rightarrow R^+$ , which encodes information on the classification from the category labels in the original data.<sup>1</sup>

From a mathematical point of view, the optimization problem with orthonormality constraints is actually an unconstrained optimization problem on the Stiefel manifold. Concretely, the search space of  $W$  is on the Stiefel manifold if the minimization problem  $L(W)$  has the orthonormality constraints:  $W^T W = I_d$ . Moreover, when the objective function is invariant to the orthogonal group,  $L(W) = L(WR)$  for any  $R \in O(n)$ , the search space of  $W$  is on the Grassmann manifold. In this case, equation 3.2 is identified as an unconstrained minimization problem on  $G(d, D)$ . In this case, combined with proposition 1, it can be guaranteed that our objective function is invariant to the choice of basis of the subspace spanned by  $W$ .

For the optimization problem on the Riemannian manifold, we seek a solution through the Riemannian gradient descent (RGD) method (Absil, Mahony, & Sepulchre, 2009). We briefly introduce the RGD method next.

**3.2 Riemannian Gradient Descent.** Consider a nonconvex and constrained optimization problem that can be expressed as

$$\min_{W \in M} L(W). \quad (3.3)$$

The set of constraints,  $M$ , forms the geometry of a Riemannian manifold. By capitalizing on the RGD algorithm, the solution to the optimization in equation 3.2 is iteratively enhanced as

$$W^{(t+1)} = \Upsilon(-\eta R_W L(W^{(t)})), \quad (3.4)$$

where  $R_W L(\cdot)$  is the Riemannian gradient of the cost function and  $\eta > 0$  is the step size. Furthermore,  $\Upsilon(\cdot) : T_M \rightarrow M$  is called a retraction, which is a mapping from the tangent space of  $M$  at  $W$  to the manifold.

As shown in equation 3.3, the Riemannian gradient  $R_W L(\cdot)$  is in the tangent space  $T_M$ , while  $W$  is the point on the manifold. Thus, we need to obtain the component on the manifold from the tangent space through a Riemannian operator, which is called the Riemannian exponential map. However, computing the exponential maps is computationally expensive in most cases. In practice, retractions are selected as approximations to the Riemannian exponential maps. In Riemannian optimization, a retraction plays a significant role that serves to jointly move in the descent direction and guarantees the new solution to be on the manifold. As a matter of course,

<sup>1</sup>The detailed notion of  $G$  is described in section 3.4.

the forms of  $R_W L(\cdot)$  and  $\Upsilon(\cdot)$  are manifold specific. (For further details and rigorous treatments of these formulas, refer to Absil et al., 2009.)

This nonlinear method essentially requires the gradient on the Riemannian manifold. More specifically, the Riemannian gradient on  $G(d, D)$  can be computed as

$$R_W L(W) = (I_D - WW^T) \nabla_W L(W), \quad (3.5)$$

where  $\nabla_W L(W)$  is the Euclidean gradient of  $L(W)$  with respect to  $W$ , which is the Jacobian matrix of size  $D$  by  $d$ .

The retraction on  $G(d, D)$  can be formed as

$$\Upsilon_W(\zeta) = qf(W + \zeta), \quad (3.6)$$

where  $qf(\cdot)$  is the adjusted Q factor of the QR decomposition (Golub & Van Loan, 2012).

Next, we describe the detailed derivations of  $\nabla_W L(W)$  under different metrics.

**3.3 The Derivation of Gradient.** Here, we derive the components that are required to perform Riemannian optimization on the Grassmannian. Note that  $X_1, X_2 \in G(n, D)$  are two arbitrary Grassmannian points, and  $Y_1 = W^T X_1, Y_2 = W^T X_2$  are two resulting Grassmannian points on the low-dimensional manifold. We also denote  $X_{sym} = \frac{1}{2}(X^T + X)$ .

**Remark 1.** For the projection metric, the partial derivatives  $\nabla_W L(W)$  are

$$\nabla_W d_{pro}^2(W^T X_1, W^T X_2) = 2AWW^T AW,$$

where  $A = X_1 X_1^T - X_2 X_2^T$ .

For the other four metrics, due to their complex formulas, we cannot directly compute the gradient of the objective function with respect to  $W$  that we want. To tackle this issue, the matrix chain rule and Taylor's theorem are under consideration.

**3.3.1 The Matrix Chain Rule.** In this section, we explain how the chain rule of matrix (Ionescu, Vantzou, & Sminchisescu, 2015) and the Taylor expansion of matrix functions (Magnus & Neudecker, 1999) can be used to calculate the partial derivative of a matrix function. For two arbitrary matrix functions,  $f(X) = Y$  and  $L = L(Y)$ , we have

$$L = L(Y) = L \circ f(X).$$

According to the matrix calculus theorem (Giles, 2008; Bodewig, 2014), the Taylor expansions of  $L(Y)$  and  $L \circ f(X)$  are

$$L(Y + dY) - L(Y) = \frac{\partial L}{\partial Y} : dY + O(\|dY\|^2), \quad (3.7)$$

$$L \circ f(X + dX) - L \circ f(X) = \frac{\partial L \circ f}{\partial X} : dX + O(\|dX\|^2), \quad (3.8)$$

where  $X : Y = \text{tr}(X^T Y)$  is an inner product of the matrices.

Referring to Ionescu et al. (2015), when  $dY = df(X; dX)$ , equations 3.7 and 3.8 are equal. Consequently, the first-order terms of the Taylor expansions in equations 3.7 and 3.8 are also equal, which leads to the matrix chain rule,

$$\begin{aligned} \frac{\partial L}{\partial Y} : dY &= \frac{\partial L}{\partial Y} : \phi(dX) \triangleq \phi^* \left( \frac{\partial L}{\partial Y} \right) : dX = \frac{\partial L \circ f}{\partial X} : dX \Rightarrow \frac{\partial L \circ f}{\partial X} \\ &= \phi^* \left( \frac{\partial L}{\partial Y} \right), \end{aligned}$$

where  $dY = \phi(dX) \triangleq df(X; dX)$  and  $\phi^*(\cdot)$  is a nonlinear adjoint operator (Ionescu et al., 2015) of  $\phi$ .

**Remark 2.** For the Fubini-Study metric, the partial derivatives  $\nabla_W L(W)$  are

$$\nabla_W d_{FS}^2(W^T X_1, W^T X_2) = \left( 2X_1 \frac{\partial L(Y)}{\partial Y} X_2^T \right)_{sym} W,$$

where  $\frac{\partial L(Y)}{\partial Y} = \frac{-1}{\sqrt{1+|\det(Y)|^2}} |\det(Y)| (Y^{-1})^T$  and  $Y = X_1^T W W^T X_2$ .

**Remark 3.** For the Binet-Cauchy distance, the partial derivatives  $\nabla_W L(W)$  are

$$\nabla_W d_{BC}^2(W^T X_1, W^T X_2) = \left( 2X_1 \frac{\partial L(Y)}{\partial Y} X_2^T \right)_{sym} W,$$

where  $\frac{\partial L(Y)}{\partial Y} = -|\det(Y)| (Y^{-1})^T$  and  $Y = X_1^T W W^T X_2$ .

**Remark 4.** For the Binet-Cauchy kernel metric, the partial derivatives  $\nabla_W L(W)$  are

$$\nabla_W d_{BCK}^2(W^T X_1, W^T X_2) = \left( 2 \frac{\partial L(Y)}{\partial Y} \right)_{sym} W,$$

where  $\frac{\partial L(Y)}{\partial Y} = (X_1 \frac{\partial L(A)}{\partial A} X_2 X_2^T Y X_1)^T + (X_1^T Y X_2 X_2^T)^T \frac{\partial L(A)}{\partial A} X_1^T$ .

**Remark 5.** For the projection kernel metric, the partial derivatives  $\nabla_W L(W)$  are

$$\nabla_W d_{\text{pk}}^2(W^T X_1, W^T X_2) = \frac{\partial L(Y)}{\partial Y} W + \left( \frac{\partial L(Y)}{\partial Y} \right)^T W,$$

where  $\frac{\partial L(Y)}{\partial Y} = X_1 X_1^T W W^T X_2 X_2^T + X_2 X_2^T W W^T X_1 X_1^T$ .

**3.4 Defining the Graph Matrix.** Given a training data set of image sets, the graph matrix can be constructed according to the supervised samples, which will be used in equation 3.2. For unsupervised data, the method is also available provided that the pairwise similarity can be measured (i.e., the distance measurement). In this letter, our task is related to classification so we naturally consider taking advantage of class labels. We aim to maximize the between-class distances while minimizing the within-class distances to achieve better classification on the new manifold. Based on this criterion, for the image sets with  $C$  classes,  $y_i$  is the class label of the image set  $X_i$ , and each element of the graph matrix used to measure the similarity of  $X_i$  and  $X_j$  can be expressed as

$$G(i, j) = G_w(i, j) - G_b(i, j),$$

where  $G_w$  is the within-class similarity graph and  $G_b$  is the graph to measure the between-class similarity.

$G_w$  and  $G_b$  are defined as

$$G_w(i, j) = \begin{cases} 1, & \text{if } X_i \in N_w(X_j) \text{ or } X_j \in N_w(X_i) \\ 0, & \text{otherwise} \end{cases},$$

$$G_b(i, j) = \begin{cases} 1, & \text{if } X_i \in N_b(X_j) \text{ or } X_j \in N_b(X_i) \\ 0, & \text{otherwise} \end{cases},$$

where  $N_w(X_i)$  consists of  $k_w$  neighbors with the same label as  $X_i$  and  $N_b(X_i)$  is the set of  $k_b$  neighbors whose labels are different from  $X_i$ . In practice,  $k_w$  is defined as the minimum number of points in each class, and the value of  $k_b \leq k_w$  is set by cross-validation to balance the relationship of  $G_w$  and  $G_b$ .

## 4 Experiments

In this section, we conduct extensive experiments to evaluate our proposed method on image set recognition tasks. First, we use the validation data set consisting of labeled Grassmannian points to validate the effectiveness of our algorithm. Second, we evaluate our method on the Cambridge hand

gesture data set. Then, one challenging data set for activity recognition, the ballet data set, is chosen to evaluate the performance of our method.

In our experiments, each image set is represented in the matrix form as  $X_i = (x_1, x_2, x_3, \dots, x_n)$ , where  $x_i \in R^D$  corresponds to the vectorized feature of the  $i$ th frame in the video. Because our method is devised on the Grassmann manifold, we represent each image set as a point on the Grassmannian referred to (Liu, Shi, & Liu, 2018, and Liu, Shi, Liu, & Zhang, 2018). It is known that a subspace is generally represented by orthonormal bases, so  $X_i$  can be constructed as a linear subspace by using the singular value decomposition (SVD) method. More specifically, we preserve the first  $n$  singular vectors to model the linear subspace of  $X_i$  as an element on the  $G(n, D)$ . In all our experiments, the dimensionality of the low-dimensional Grassmann manifold and the value of  $n$  are determined by cross-validation. All the operations associated with conjugate gradient descent required on the manifold are implemented within the `manopt` Riemannian optimization toolbox (Boumal, Mishra, Absil, & Sepulchre, 2014). Next, we provide a brief overview of the experimental data sets and then present the analysis of our experimental results.

To evaluate the performance of our method, we first adopt the simple nearest neighbor (NN) classifier based on different Grassmannian metrics to intuitively evaluate the effectiveness of our proposed algorithm. This simple classifier clearly and directly reflects the advantages of learning the lower-dimensional manifold from the original manifold. Second, we compare our method with three state-of-the-art algorithms: GGDA, GDL, and PML. Moreover, we make use of different Grassmann-based algorithms to show that the lower-dimensional manifold learned by our algorithm can lead to more state-of-the-art algorithms. Because both GGDA and GDL employ a kernel derived from the projection metric, we apply them only to the projection metric-based version of our method. For GGDA, the parameter  $\beta$  is tuned within the range of  $\{e^1, e^2, \dots, e^{10}\}$ . For GDL, the parameter  $\lambda$  is tuned within the range of  $\{e^{-1}, e^{-2}, \dots, e^{-10}\}$ . For PML, we use the code offered by the author and adopt parameter settings suggested in his paper. For a fair comparison, the key parameters of each method are empirically tuned according to the recommendations in the original works. All the algorithms used in our experiments are referenced as follows:

- NN-P/FS/PK/BC/BCK: NN classifier on the original Grassmannian based on the Projection/ Fubini-Study/ Projection kernel/ Binet-Cauchy/ Binet-Cauchy kernel metric.
- P/FS/PK/BC/BCK-DR: NN classifier with different metrics on the low-dimensional Grassmann manifold obtained with our approach
- GGDA (Harandi et al., 2011)/GGDA-DR: Graph-embedding Grassmann discriminant analysis on the original Grassmannian and the low-dimensional Grassmann manifold obtained with our approach

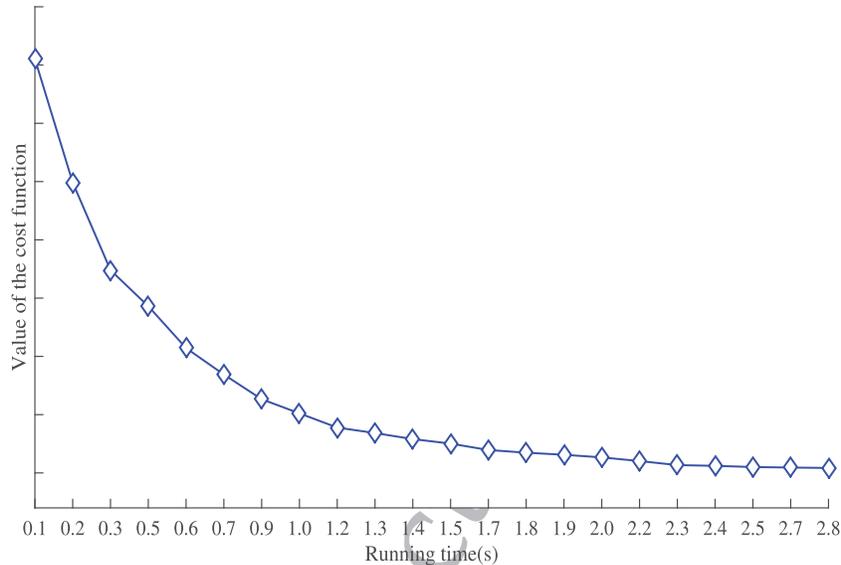


Figure 1: Convergence behavior of our algorithm.

- GDL (Harandi et al., 2013)/GDL-DR: Grassmann Dictionary Learning on the original Grassmannian and the low-dimensional Grassmann manifold obtained with our approach
- PML (Huang et al., 2015): Projection metric learning based on Grassmannian

**4.1 Validation Experiment.** In this section, we use the validation data set from Huang et al. (2015) to provide a systematic study of the effects of various parameters and metrics on the performance of our algorithm. This data set consists of 80 samples with eight classes. Each class includes five training samples and five test samples. Each sample is a  $37 \times 41$  matrix that can be represented as a point on the Grassmann manifold with a linear subspace by SVD. In this case, we obtain 80 Grassmannian points with labels. This data set was selected to validate the correctness of our method because it has the advantages of low-dimensionality of the data and small temporal expense. Figure 1 illustrates the typical convergence behavior of our method. In practice, the algorithm generally converges rapidly in fewer than 25 iterations.

Next, we analyze the effects of various parameters on the performance of the proposed method. The parameters we focus on are the order of the linear subspace,  $n$ , and the reduced dimensionality learned by our algorithm,

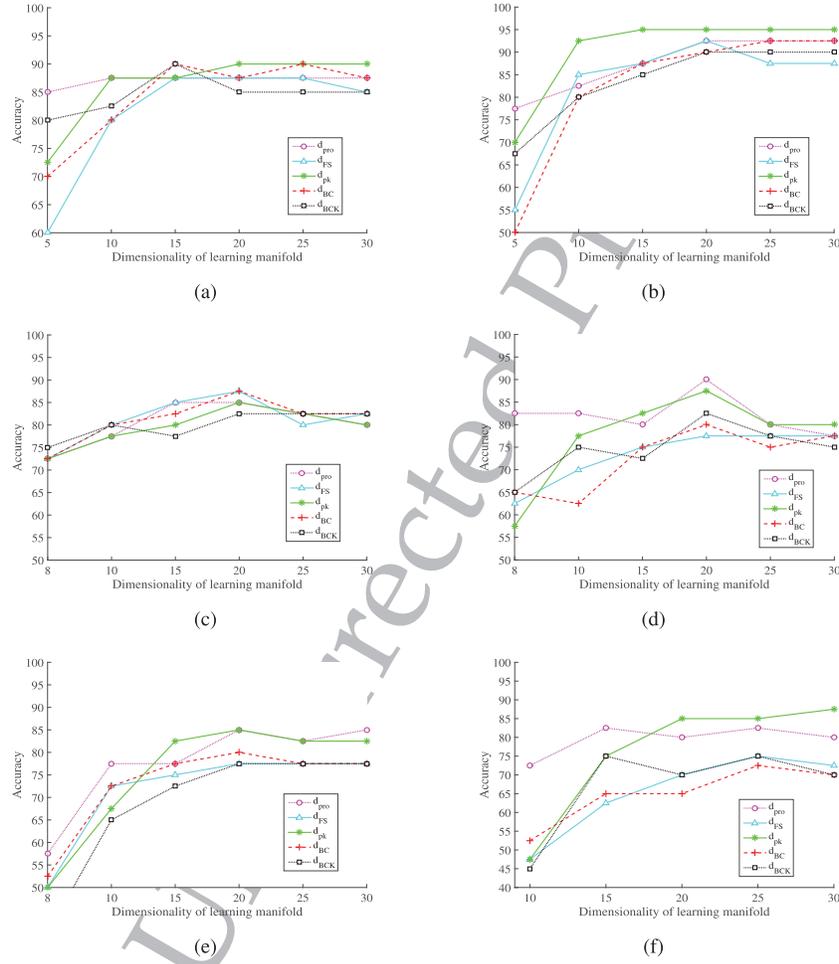


Figure 2: Averaged accuracies of the proposed method with different dimensionalities of the learning manifold: (a)  $n = 2$ ; (b)  $n = 3$ ; (c)  $n = 4$ ; (d)  $n = 5$ ; (e)  $n = 6$ ; (f)  $n = 7$ .

*d.* Figure 2 illustrates the performance of our algorithm with varying levels of dimensionality reduction under five metrics. The images in Figures 2a to 2f, respectively, show the classification accuracies obtained from the NN classifier while varying the subspace order from  $n = 2$  to  $n = 7$ . These comparisons show that the curves always reach their peaks at the dimensionality of 20 except when  $n = 7$ . These results may be due to the intrinsic

increase in the dimensionality of the ambient space as the subspace order becomes larger. From Figure 2b, it is obvious that  $n = 3$  is a good candidate for this data set.

Next, we report the average classification accuracy. All experiments are repeated 10 times to obtain the average results. Table 2 shows the performance of the different methods. Comparing the results of NN-P method and the NN-P-DR method, the classification accuracy is improved after our mapping from the original Grassmannian to a lower-dimensional space. Similarly, for the other metrics, we also obtain a better classification result on the newly learned Grassmann manifold, which can be observed by the results of the NN and NN-DR methods. For the FS metric and the BC distance, the results are improved significantly. All of these results demonstrate that our method generates a better Riemannian geometry for classification purposes (i.e., the low-dimensional Grassmann manifold).

Note that our method is different from that of Huang et al. (2015), which focuses on the projection metric learning on the Grassmann manifold and is actually an optimization problem on the SPD manifold. Although Huang performs dimensionality reduction on the Grassmann manifold, he embeds the points on the Grassmannian into the points on the SPD manifold through projection mapping. However, in fact, Huang’s work is a special type of dimensionality reduction for the SPD manifold. From this viewpoint, his work is limited to the projection mapping on the Grassmannian, and the projection metric is the unique metric specific to his methodology. In our method, inspired by Harandi et al. (2017), we directly approve a geometry-aware dimensionality reduction for the Grassmann manifold to obtain a lower-dimensional manifold where better classification can be achieved. We directly use the geometry-aware property of the Grassmann manifold instead of transforming it to the SPD manifold. Furthermore, because our method does not depend on any additional intermediary, in theory, any distance metric can be used directly. In this letter, five metrics are derived for our purposes.

**4.2 Hand Gesture Recognition.** In this experiment, we used the Cambridge hand gesture data set (Kim & Cipolla, 2009) to test our method on hand gesture recognition. This data set contains 900 image sequences in nine classes. All sequences are divided into five sets according to varying illuminations. Each set consists of 180 image sequences of 10 arbitrary motions performed by two subjects. We compute histogram of oriented gradient (HOG) (Dalal & Triggs, 2005) features to construct linear subspaces of the image sequences. In our protocol, we select the first 10 sequences as test data and the last 3 sequences as training data in each class. Hence, we generate 117 Grassmannian points from the 90 test samples and 27 training samples.

Table 2 describes the performances of our method under different metrics and those of the state-of-the-art methods on this data set. The

Table 2: Average Recognition Rates on Different Datasets.

Method	NN-P	NN-FS	NN-PK	NN-BC	NN-BK	GGDA	GDI	PML	P-DR	FS-DR	PK-DR	BC-DR	BCK-DR	GGDA-DR	GDL-DR
Data set: Validation: 80 samples (40 test samples and 40 training samples)															
Results	85	90	85	85	85	90	92.5	92.5	95	92.5	95	92.5	90	97.5	97.5
Data set: Hand gesture: 117 samples (90 test samples and 27 training samples)															
Results	71.11	65.56	67.78	65.56	65.56	52.24	74.39	55.57	73.33	68.89	70.0	66.67	66.82	73.49	76.85
Data set: Ballet: 1328 samples (1168 test samples and 160 training samples)															
Results	48.63	29.62	48.63	29.62	29.62	37.84	50.86	51.71	51.88	31.42	50.60	31.25	31.34	39.38	56.68

Note: The numbers in bold indicate the best results. The numbers in italic indicated the best enhanced accuracies by our method.

NN classifier's performance under all metrics is enhanced by the dimensionality-reduced data and reaches competitive results that are minimally 10% higher than PML. Specifically, P-DR is approximately 19% higher than PML. Both GGDA and GDL improve when learning the lower-dimensional Grassmann manifold (GGDA-DR and GDL-DR). Furthermore, our method boosts the accuracy of GGDA to a large extent of more than 21% (from 52.24% to 73.49%). In addition, GDL-DR improves on the original method, GDL, and has the best performance. These improvements occur because our algorithm respects the Riemannian structure of Grassmannian while simultaneously learning a lower-dimensional and more discriminative manifold. Consequently, the competing methods are also improved due to the reduced data our method obtains.

**4.3 Recognition on the Ballet Data Set.** The ballet data set includes 440 videos derived from an instructional ballet DVD (Wang & Mori, 2009). All of these videos can be classified into eight complicated motion patterns performed by three people. This data set is highly challenging because the large intraclass variations exist with respect to spatial and temporal scales, clothing, speed, and movement.

We generate 1328 image sets from the data set by treating every 12 frames derived from the same action as a subspace. Each image set is represented as a subspace based on the HOG features. We select 20 image sets from each action (160 samples in total) as training data and 1168 samples for testing. For the projection metric and projection kernel metric, each image set is represented as a linear subspace of order 6. For the other metrics, the dimension of each subspace is set to 3.

Table 2 shows the experimental evaluation on this data set. As these results show, the accuracies of the NN classifier on the learning dimensionality-reduced Grassmann manifold are always improved compared with those on the original manifold. After applying our learning algorithm, P-DR not only outperforms GGDA by approximately 15% but also surpasses PML and GDL. The best result of GDL-DR exceeds the accuracy of GDL on the learning Grassmann manifold by about 6% and achieves 56.68%.

#### 4.4 Experiments in the Euclidean Geometry.

*4.4.1 Derivations in the Euclidean Space.* To validate the effectiveness of the Riemannian geometry of Grassmann manifolds, we use the Euclidean structure to perform comparisons. Because the Grassmann manifold  $G(n, d)$  can be regarded as a subspace in the Euclidean space, we embed the Grassmannian into the Euclidean space and use the Frobenius norm as the distance metric. In this case, the distance between any two points on

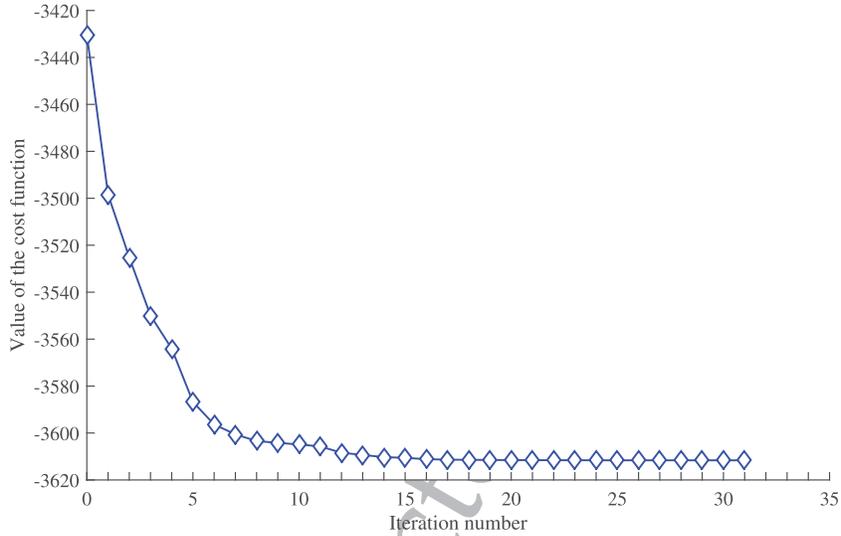


Figure 3: Convergence behavior in Euclidean geometry.

Grassmannian is  $d(X_1, X_2) = \|X_1 - X_2\|_F$ , and then we have

$$\begin{aligned} d^2(Y_1, Y_2) &= \|Y_1 - Y_2\|_F^2 = \|W^T X_1 - W^T X_2\|_F^2 = \|W^T (X_1 - X_2)\|_F^2 \\ &= \text{tr} \left\{ (X_1 - X_2)^T W W^T (X_1 - X_2) \right\}. \end{aligned}$$

Based on this formulation, we compute the corresponding Euclidean gradient of  $d^2(Y_1, Y_2)$  as

$$\begin{aligned} d^2(Y_1, Y_2) &= \nabla_W \text{tr} \left\{ (X_1 - X_2)^T W W^T (X_1 - X_2) \right\} \\ &= \nabla_W \text{tr} \left\{ W W^T (X_1 - X_2) (X_1 - X_2)^T \right\} \\ &= 2(X_1 - X_2)(X_1 - X_2)^T W. \end{aligned}$$

*4.4.2 Experimental Evaluation.* We use the demonstration data in the validation experiment to evaluate the performance of our algorithm in the Euclidean space. Figure 3 illustrates the convergence behavior of our algorithm in the Euclidean geometry. The classification accuracy is 57.5%, which is considerably lower than the result of the Grassmannian model in this letter. However, this result is reasonable due to the utilization of Riemannian geometry. All the above validate that the constraint of the output space being a Grassmannian adds value to the dimensionality-reduction process.

**4.5 Further Discussion.** Here, we provide a detailed discussion of the proposed method from three aspects: computational complexity, how it scales with the number of examples, and various dimensionalities.

*4.5.1 Computational Complexity.* As mentioned in section 3.2, the computational complexity of each iteration of SGD on a Grassmann manifold depends on the computational cost of the following major steps:

- Riemannian gradient: Projecting the Euclidean gradient to the tangent space of  $G(d, D)$ , as in equation 3.4, involves matrix multiplications between matrices of size  $1-d \times D$  and  $D \times d$  and  $2-D \times d$  and  $d \times d$ , which sums to  $2Dd^2$  flops.
- Retraction: The retraction involves computing and adjusting the QR decomposition of a  $D \times d$  matrix. The complexity of the QR decomposition using the Householder algorithm is  $2d^2(D - \frac{d}{3})$  (Golub & Van Loan, 2012). The adjustments change the sign of elements of a column only if the corresponding diagonal element of  $R$  is negative, which does not incur much cost. Consequently, the retraction has a total complexity of  $2d^2(D - \frac{d}{3})$ .

Overall, an update of our method mainly demands  $2d^2(2D - \frac{d}{3})$  extra flops, which is linear in  $D$  and causes all the steps to have affordable computational complexity.

*4.5.2 Performance with Different Factors.*

*Various dimensionalities.* Because the proposed method is a supervised learning algorithm for dimensionality reduction, the key parameter in our model is the reduced dimensionality in the low-dimensional space. Therefore, we use the ETH-80 data set (i.e., the validation data set) to provide a more systematic study on the effects of various dimensionalities under different metrics on the performance of our algorithm. In the experiments, the dimensionalities of the original data on the Grassmann manifold  $G(3, 400)$  are reduced to the various dimensionalities as shown in Figure 2. From an overall perspective, the classification accuracies are always improved by our supervised dimensionality-reduction method. The classification not only benefits from the intrinsic dimensionality obtained from the high-dimensional data but also can be attributed to the utility of class labels during the dimensionality reduction, thereby encoding a more discriminative structure in the low-dimensional manifold from the pairwise relationship of the original data. When the reduced dimensionality adopts an initially small value, the accuracies are far away from the baseline results. This result occurs because when the reduced dimensionality  $d$  is smaller than the intrinsic dimensionality, the reduced data can lose some useful discriminative information. From the observation, the reduced data obtained from our

method can reach peak values when the dimensionality ranges from 20 to 30, allowing a powerful discriminatory capacity from the original data.

*Number of examples.* We select different number of examples from three data sets to conduct the experiments. As reported in Table 2, we generate 80 samples and 117 samples in the ETH-80 data set and Cambridge hand gesture data set, respectively. In this case, the reduced data learned from our algorithm with a small quantity of training samples can also lead to better classification results when they are treated as inputs for the Grassmann-based algorithms. Specifically, the accuracies of the NN classifier increase by 7.5% under the FS and BCK metrics on the ETH-80 data set. The classification ability of GGDA obviously improves by more than 21% in the hand gesture data set, as indicated by the italic numbers in Table 2. To explore the results when the number of examples is large, we generated 1328 samples from the ballet data set. Although this data set is challenging, considering the complexity of its data, through our framework, the capacity of GDL generates a substantial promotion of approximately 6%.

Overall, the dimensionality reduced by our method not only leads to the reduction of one order of magnitude but also gains a significant enhancement on the classification accuracies, which demonstrates the effectiveness and robustness of the proposed algorithm.

## 5 Conclusion

To the best of our knowledge, this work is the first effort to provide a general framework for the Grassmann manifold without certain metric limitations, and it shows the importance of respecting the Riemannian geometry when performing dimensionality reduction.

We proposed a novel supervised algorithm that inherently learns a lower-dimensional and more discriminative Grassmann manifold from the original one while simultaneously accounting for different metrics. The learning process to find an orthogonal transformation can be modeled as an optimization problem on a Grassmann manifold. Our experimental evaluations on several challenging data sets have demonstrated that the resulting low-dimensional Grassmann manifold consistently improves classification accuracy compared to using the low-dimensional Grassmannians directly. In the future, we plan to study additional types of cost functions and metrics within our framework to improve the discriminative capability further. Moreover, we intend to extend our method to both unsupervised and semisupervised scenarios.

## Acknowledgments

This work is supported by the Innovation Fund of the Chinese Academy of Sciences (grant Y8K4160401). We especially appreciate the discussion and help provided by Mehrtash Harandi and Chenxi Li. We are also very

grateful to the efficient editors and anonymous reviewers for their constructive comments and suggestions that improved this letter.

## References

- Absil, P.-A., Mahony, R., & Sepulchre, R. (2009). *Optimization algorithms on matrix manifolds*. Princeton, NJ: Princeton University Press.
- Bodewig, E. (2014). *Matrix calculus*. Amsterdam: Elsevier.
- Boumal, N., Mishra, B., Absil, P.-A., & Sepulchre, R. (2014). Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15(1), 1455–1459.
- Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (vol. 1, pp. 886–893). Piscataway, NJ: IEEE.
- Edelman, A., Arias, T. A., & Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2), 303–353.
- Giles, M. B. (2008). Collected matrix derivative results for forward and reverse mode algorithmic differentiation. In C. H. Bischof, H. M. Bücker, P. Hovland, U. Naumann, & J. Utke (Eds.), *Advances in automatic differentiation* (pp. 35–44). Berlin: Springer.
- Golub, G. H., & Van Loan, C. F. (2012). *Matrix computations*, vol. 3. Baltimore: Johns Hopkins University Press.
- Hamm, J., & Lee, D. D. (2008). Grassmann discriminant analysis: A unifying view on subspace-based learning. In *Proceedings of the 25th international Conference on Machine Learning* (pp. 376–383). New York: ACM.
- Harandi, M., Salzmann, M., & Hartley, R. (2017). Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, 48–62.
- Harandi, M. T., Salzmann, M., Jayasumana, S., Hartley, R., & Li, H. (2014). Expanding the family of Grassmannian kernels: An embedding perspective. In *Proceedings of the European Conference on Computer Vision* (pp. 408–423). Berlin: Springer.
- Harandi, M., Sanderson, C., Shen, C., & Lovell, B. C. (2013). Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3120–3127). Piscataway, NJ: IEEE.
- Harandi, M. T., Sanderson, C., Shirazi, S., & Lovell, B. C. (2011). Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2705–2712). Piscataway, NJ: IEEE.
- Holland, S. M. (2008). *Principal components analysis (PCA)*. Athens: Department of Geology, University of Georgia.
- Huang, Z., Wang, R., Shan, S., & Chen, X. (2015). Projection metric learning on Grassmann manifold with application to video based face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 140–149). Piscataway, NJ: IEEE.

- Ionescu, C., Vantzos, O., & Sminchisescu, C. (2015). Matrix backpropagation for deep networks with structured layers. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2965–2973). Piscataway, NJ: IEEE.
- Izenman, A. J. (2013). *Modern multivariate statistical techniques*. Berlin: Springer.
- Kim, T.-K., & Cipolla, R. (2009). Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8), 1415–1428.
- Liu, T., Shi, Z., & Liu, Y. (2018). Kernel sparse representation on Grassmann manifolds for visual clustering. *Optical Engineering*, 57(5), 053104.
- Liu, T., Shi, Z., Liu, Y., & Zhang, Y. (2018). Geometry deep network image-set recognition method based on Grassmann manifolds. *Infrared and Laser Engineering*, 47(7), 0703002.
- Magnus, J. R., & Neudecker, H. (1999). *Matrix differential calculus with applications in statistics and econometrics*. New York: Wiley.
- Wang, Y., & Mori, G. (2009). Human action recognition by semilattent topic models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10), 1762–1774.

---

Received November 28, 2017; accepted August 29, 2018.