The Journal of Supercomputing (2025) 81:397
https://doi.org/10.1007/s11227-024-06904-1

Discriminant locality preserving projection on Grassmann Manifold for image-set classification

Benchao Li[1] · Ting Wang[1] · Ruisheng Ran[1]

Accepted: 27 December 2024

Abstract
In the domain of image-set analysis, the Grassmann Manifold serves as an efficacious instrument, though the computational expenses it incurs are notably substantial. Locality Preserving Projection on Grassmann Manifold (GLPP), a classical algorithm for dimensionality reduction, effectively alleviates these computational burdens. However, it is important to note that the construction of the neighborhood graph in GLPP overlooks the categorical information of the data, thereby limiting its discriminative capacity. To address this issue, we propose the Discriminant Locality Preserving Projection on Grassmann Manifold (GDLPP) and Semi-Supervised Locality Preserving Projection on Grassmann Manifold (GSLPP). Classification outcomes across 6 image-set datasets reveal that the classification capabilities of GDLPP and GSLPP surpass those of existing image-set classification algorithms. By effectively preserving the local manifold structure of the data and comprehensively leveraging the label information, GDLPP and GSLPP notably augment the feature extraction and classification prowess of GLPP.

Keywords Grassmann Manifold · Image-set classification · Locality preserving projection · Discriminant learning · Semi-supervised learning

1 Introduction

An image-set refers to a collection of images with inherent correlations, which can either be an unordered set of photographs taken from different perspectives or an ordered sequence of video frames. Recent studies have demonstrated that image-set

Benchao Li and Ting Wang have contributed equally to this work.

✉ Ruisheng Ran
rshran@cqnu.edu.cn

[1] College of Computer and Information Science, Chongqing Normal University, Chongqing 401331, China

Published online: 17 January 2025

data often reside on low-dimensional manifolds with nonlinear metrics, where traditional Euclidean metrics perform poorly in tasks such as image-set classification and clustering. In contrast, Riemannian manifolds, such as the Grassmann Manifold [1, 2], SPD Manifold [3–5], and Stiefel Manifold [6, 7], possess nonlinear metrics that more accurately capture the internal geometric relationships within an image-set and the geodesic similarities between samples. Consequently, image-set analysis based on Riemannian manifolds has emerged as a significant area of research.

The Grassmann Manifold is an excellent subspace structure for capturing the intrinsic geometry of image-set data and is widely applied in image-set classification [8, 9] and clustering [10, 11] tasks. However, methods based on the Grassmann Manifold typically require substantial computational resources. Additionally, image-set data often contain significant amounts of noise, which can severely impact the performance of algorithms on the Grassmann Manifold. Therefore, from the perspective of computational efficiency, developing an effective dimensionality reduction method that preserves neighborhood and manifold structures to extract meaningful features from the data is of great research significance and practical value.

Dimensionality reduction [12, 13] techniques play a crucial role in machine learning tasks and are a vital step in data processing. These techniques help eliminate redundant information in the data, thereby significantly enhancing the performance of classification and clustering algorithms. Classical dimensionality reduction methods, such as principal component analysis (PCA) [14] and linear discriminant analysis (LDA) [15], are representative algorithms for unsupervised and supervised dimensionality reduction, respectively. However, PCA and LDA typically preserve only the global structure of the data and fail to maintain the manifold structure. Manifold learning-based dimensionality reduction techniques, such as Locally Linear Embedding (LLE) [16], Laplacian Eigenmaps (LE) [17], and Uniform Manifold Approximation and Projection (UMAP) [18], address this limitation by preserving the neighborhood structure of the data while reducing its dimensionality. To resolve the Out-of-Sample Extension Problem [19, 20] inherent in Manifold Learning algorithms like LE, He et al. proposed the Locality Preserving Projection (LPP) [21].

LPP is a linear Manifold Learning algorithm that establishes the embedding relationship between high-dimensional and low-dimensional manifolds through a mapping matrix. These algorithms are capable of preserving the local relationships within data while extracting its features. To further enhance the performance of LPP, a series of improved algorithms have been developed. Yu et al. proposed the Improved LPP with L1-norm [22], which effectively addresses LPP's sensitivity to outliers and Small Sample Size Problem, significantly improving its feature extraction capability. The Exponential LPP [23] overcomes the singular matrix problem, resolving both the Small Sample Size Problem and neighborhood sensitivity, thereby markedly enhancing LPP's performance. Wang et al. introduced the Local Adaptive Preserving Projections [24], which iteratively selects the optimal

neighbors for samples, mitigating the impact of noise on neighborhood selection and further improving the robustness of LPP.

There are numerous improvements to the LPP algorithm; however, these vector-based dimensionality reduction techniques typically require flattening images into vectors, which demands substantial computational resources. To address dimensionality reduction directly on images, Chen [25] and Hu [26] et al. independently proposed the Two-Dimensional Locality Preserving Projection (2DLPP). 2DLPP significantly reduces the time complexity and enhances classification performance. To tackle the issue of 2DLPP's sensitivity to outliers, Chen et al. introduced the Robust Two-Dimensional Locality Preserving Projection with Regularization (2DRLPP) [27]. This method employs the L1-norm to measure the similarity between samples and uses a regularization term to control the computational complexity of the algorithm.

The aforementioned improvements to LPP are designed for samples in Euclidean space and are not suitable for dimensionality reduction on Grassmann Manifold. Wang et al. were the first to extend LPP to the Grassmann Manifold, achieving Locality Preserving Projections on Grassmann Manifold (GLPP) [28] by utilizing embedding metrics of Grassmann Manifold. GLPP significantly enhances the performance of classification and clustering algorithms on the Grassmann Manifold and has been widely applied in image-set analysis tasks. Wei et al. further proposed an F-norm-based OLPP algorithm on Grassmann Manifold (O-GLPP) [29], which improves the robustness of the GLPP algorithm.

However, both GLPP and O-GLPP are unsupervised dimensionality reduction methods that do not fully utilize the label information in the data. Existing supervised dimensionality reduction techniques on Grassmann Manifold are primarily based on kernel methods, including Grassmann Discriminant Analysis (GKDA) [30] and Grassmann Graph Embedding Discriminant Analysis (GEDA) [31]. The general approach of these methods involves embedding samples from the Grassmann Manifold into a Reproducing Kernel Hilbert Space (RKHS) and then applying kernel-based techniques in Euclidean space for dimensionality reduction. This represents an early approach to dimensionality reduction on Grassmann Manifold. However, experimental results from GLPP, Neighborhood Preserving Embedding on Grassmann Manifold (GNPE) [32], and similar methods indicate that GKDA and GEDA do not perform well in classification tasks, as kernel-based dimensionality reduction techniques fail to faithfully preserve the manifold structure.

In Euclidean space, various forms of Discriminant LPP algorithms have been successfully applied in fields such as face recognition, palmprint recognition, and fault diagnosis, underscoring the importance of discriminative information in preserving neighborhood structures. Building on the success of Discriminant Locality Preserving Projection (DLPP) in classification tasks within Euclidean space and addressing the issue of inadequate utilization of label information in LPP on Grassmann Manifold, we propose a Discriminant Locality Preserving Projection on Grassmann Manifold (GDLPP). This method enhances the discriminability of the projection matrix

by maximizing the between-class distance and minimizing the within-class distance, thereby increasing the distinguishability of the low-dimensional data and achieving superior classification accuracy.

Although advancements in information technology have made it easier to acquire large amounts of data, the cost of data labeling remains substantial. Semi-Supervised Learning [33, 34] offers a way to effectively enhance the performance of unsupervised algorithms by leveraging a small amount of labeled data.

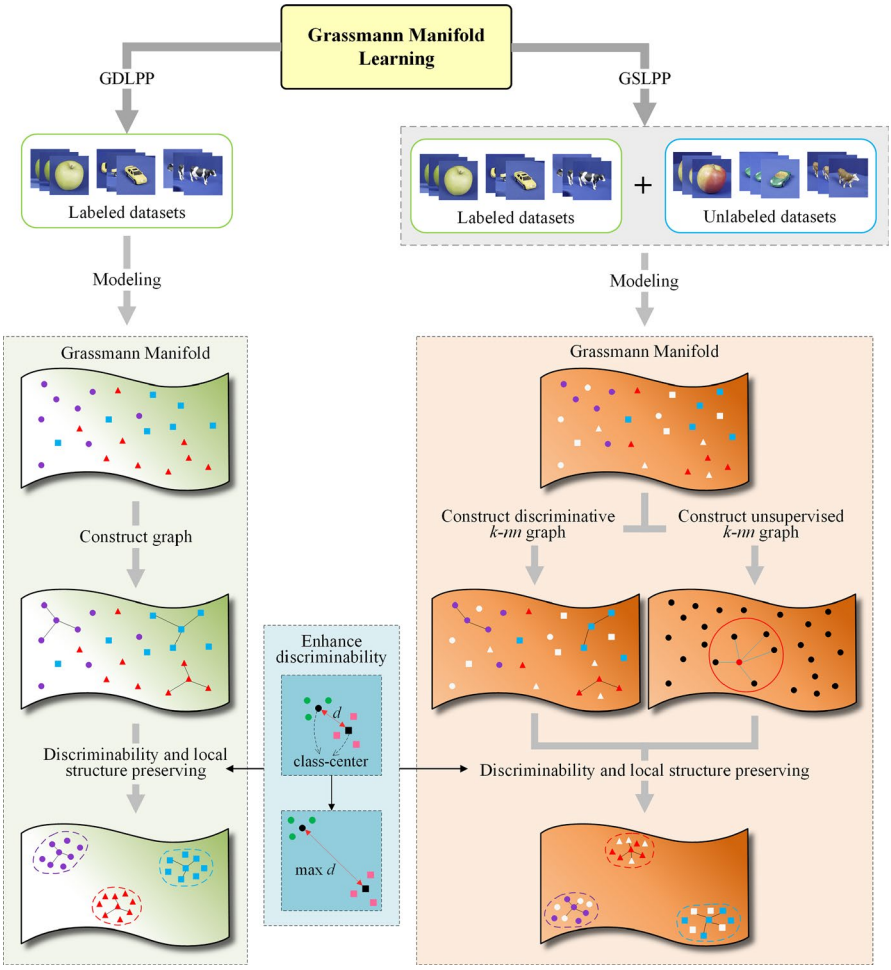


Fig. 1 Diagram of the Dimensionality Reduction Process for Discriminant Locality Preserving Projection on Grassmann Manifold (GDLPP) and Semi-Supervised Locality Preserving Projection on Grassmann Manifold (GSLPP). The left side illustrates the dimensionality reduction process of GDLPP, while the right side shows the process for GSLPP. The box in the middle represents the maximization of between-class distances among samples, which is key to enhancing the discriminability of the mapping matrix

In recent years, dimensionality reduction techniques based on Semi-Supervised Learning have seen significant development. Drawing on the success of Semi-Supervised LPP in Euclidean space and addressing the issue of insufficient data labels in real-world scenarios, we propose the Semi-Supervised Locality Preserving Projection on Grassmann Manifold (GSLPP) to improve the performance of GLPP when labels are scarce. The dimensionality reduction processes of GDLPP and GSLPP are illustrated in Fig. 1.

The contributions of this paper include:

- This paper proposes a supervised dimensionality reduction method on Grassmann Manifold, which employs Discriminant Locality Preserving Projections based on embedding metrics of Grassmann Manifold to extract features from image-set data, thereby achieving improved classification performance.
- This study proposes a semi-supervised dimensionality reduction method on Grassmann Manifold, addressing the issue of insufficient data labels in Discriminant Learning. By leveraging a limited amount of labeled data, this method effectively enhances the performance of unsupervised dimensionality reduction techniques on Grassmann Manifold.
- This paper introduces, for the first time, the integration of Semi-Supervised Learning with dimensionality reduction techniques on the Grassmann Manifold. This represents a novel exploration in the field of image-set analysis and provides a valuable reference for the future advancement of semi-supervised dimensionality reduction techniques on the Grassmann Manifold.

The rest of this paper is organized as follows: Sect. 2 introduces the Grassmann Manifold and DLPP; Sect. 3 provides a detailed explanation of the proposed GDLPP and GSLPP algorithms, along with implementation details; Sect. 4 presents the experimental results and a comparison with similar algorithms; finally, Sect. 5 offers conclusions and future research directions.

2 Related works

2.1 Grassmann manifold

The Grassmann Manifold, denoted as $\mathscr{G}(p,d)$, is an abstract concept representing the set of all p-dimensional subspaces within a d-dimensional Euclidean space. Simply put, the Grassmann Manifold consists of the equivalence classes of tall-and-thin orthogonal matrices of rank p under the action of the orthogonal group $\mathcal{O}(p)$. This can be expressed as:

$$\mathscr{G}(p,d) = \left(\boldsymbol{X} \in \mathbb{R}^{d\times p} \mid \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}_p\right)/\mathcal{O}(p) \tag{1}$$

2.1.1 Embedding distance on grassmann manifold

Embedding the Grassmann Manifold into the space of symmetric matrices allows for the adoption of a Euclidean metric, a technique widely utilized in the study of Grassmann Manifold. [35] This approach has been employed in various studies, such as GDL, GLPP, and GNPE. For a point $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathscr{G}(p, d)$ on Grassmann Manifold:

$$\operatorname{dist}_{\mathscr{G}}\left(\boldsymbol{X}_1, \boldsymbol{X}_2\right) = \frac{1}{\sqrt{2}}\left\|\Pi(\boldsymbol{X}_1) - \Pi(\boldsymbol{X}_2)\right\|_F \tag{2}$$

where

$$\Pi(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{X}^T \tag{3}$$

2.1.2 Average on grassmann manifold

Yu et al. [36] proposed a Grassmann Discriminant Analysis method based on the Fréchet mean, achieving a "From Manifold to Manifold" dimensionality reduction approach, similar in concept to GLPP and GNPE. For a given sample set $\left\{\boldsymbol{X}_i\right\}_{i=1}^N, \boldsymbol{X}_i \in \mathscr{G}(p, D)$, let:

$$\boldsymbol{C} = \sum_{i=1}^{N} \boldsymbol{X}_i \boldsymbol{X}_i^T \tag{4}$$

the Fréchet mean of the sample set can then be defined as the first p largest eigenvectors of the matrix $\boldsymbol{C}$.

2.2 LPP

In Euclidean space, LPP is a linear dimensionality reduction technique that projects high-dimensional data points $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ...\boldsymbol{x}_n\}$ onto low-dimensional counterparts $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, ...\boldsymbol{y}_n\}$ through a mapping matrix $\boldsymbol{A}$. The objective function of LPP is formally defined as:

$$\mathscr{O}_{\mathrm{LPP}} = \min_{\boldsymbol{A}} \frac{1}{2} \sum_{ij} \left(\boldsymbol{y}_i - \boldsymbol{y}_j\right) \cdot \omega_{ij} = \min_{\boldsymbol{A}} \frac{1}{2} \sum_{ij} \left(\boldsymbol{A}^T \boldsymbol{x}_i - \boldsymbol{A}^T \boldsymbol{x}_j\right) \cdot \omega_{ij} \tag{5}$$

When the distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is large, the operator $\mathscr{O}_{\mathrm{LPP}}$ assigns a small weight ω_{ij}. Conversely, when $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are close, a larger weight ω_{ij} is used. By minimizing the objective function, this approach ensures that if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are close in the high-dimensional space, their corresponding projections $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ will also be close in the low-dimensional space.

The process of LPP can be summarized in 3 steps:

1. A neighborhood graph is constructed to capture the relationships between points. LPP offers two methods for constructing a neighborhood graph: the first is based on an ϵ-threshold, and the second is the k-nearest neighbor graph.
2. Calculate the similarity between neighboring points. LPP provides two approaches for calculating weights: the first method uses the heat kernel function to compute the connection weight ω_{ij} between samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$; the second method assigns $\omega_{ij} = 1$ when samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are neighbors.
3. After simplification, the objective function of LPP can be expressed as:

$$\mathcal{O}_{LPP} = \boldsymbol{A}^T\boldsymbol{X}(\boldsymbol{D} - \boldsymbol{W})\boldsymbol{X}^T\boldsymbol{A} = \boldsymbol{A}^T\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{A} \tag{6}$$

where $\boldsymbol{D}_{ii} = \sum_i \omega_{ij}, \boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. Additionally, LPP must satisfy the following constraint:

$$\boldsymbol{Y}^T\boldsymbol{D}\boldsymbol{Y} = \boldsymbol{A}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{A} = 1 \tag{7}$$

The objective function of LPP can be reformulated as:

$$\arg\min_{\boldsymbol{A}} \boldsymbol{A}^T\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^T\boldsymbol{A} \quad s.t. \quad \boldsymbol{A}^T\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\boldsymbol{A} = 1 \tag{8}$$

2.3 Semi-supervised LPP

Zhang et al. proposed a Semi-Supervised Locality Preserving Projection (SSLPP) [37] algorithm that defines within-class compactness and smoothness terms to describe the similarity between labeled samples and the entire dataset. SSLPP uses the smoothness term as a regularization term for the within-class compactness and solves the mapping matrix in the same manner as LPP. SSLPP significantly enhances the performance of LPP, achieving noticeably higher classification accuracy across various facial datasets.

2.4 Discriminant LPP

Discriminative information is crucial for classification tasks. In existing research, there are numerous LPP based on the concept of Discriminative Learning [38, 39]. Yu et al. [40] enhanced the discriminability of LPP by minimizing the within-class scatter matrix and maximizing the between-class scatter matrix, following the principles of Linear Discriminant Analysis (LDA). This approach resulted in improved classification performance. However, both DLPP and LPP suffer from the Small Sample Size Problem. To address the problem, Lu et al. proposed DLPP based on the Maximum Margin Criterion (MMC), known as DLPP/MMC [41]. DLPP/MMC

mitigates the Small Sample Size Problem by maximizing the difference between the between-class+ scatter matrix and the within-class scatter matrix, while also improving the classification accuracy of DLPP to some extent. Traditional DLPP and DLPP/MMC employ the $L2$-norm to measure the similarity between samples, which results in poor robustness to outliers and noise. To further enhance the performance and robustness of DLPP/MMC, Lu et al. proposed an $L1$-norm-based DLPP/MMC [42], which measures sample similarity using the $L1$-norm.

The objective function of DLPP is defined as:

$$\frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c}\sum_{j=1}^{N_c}\left(\boldsymbol{y}_i^c-\boldsymbol{y}_j^c\right)\omega_{ij}^c}{\sum_{i=1}^{C}\sum_{j=1}^{C}\left(\boldsymbol{m}_i-\boldsymbol{m}_j\right)b_{ij}} \tag{9}$$

where $\boldsymbol{y}_i^c$ represents the i-th sample in the c-th class, N_c denotes the number of samples in the c-th class, and ω_{ij}^c indicates the connection weight between samples $\boldsymbol{x}_i^c$ and $\boldsymbol{x}_j^c$ within the c-th class. Additionally, $\boldsymbol{m}_i^c$ represents the centroid of the i-th class, and b_ij denotes the connection weight between the centroids $\boldsymbol{m}_i^c$ and $\boldsymbol{m}_j^c$.

The process of DLPP can be summarized in 3 steps:

(1) Construct the within-class weight matrix $\boldsymbol{W^c}$ and the between-class weight matrix $\boldsymbol{B}$:

$$\boldsymbol{W}_{ij}^c=\exp\left(-\frac{\left\|\boldsymbol{x}_i-\boldsymbol{x}_j\right\|^2}{t}\right),B_{ij}=\exp\left(-\frac{\left\|\boldsymbol{m}_i-\boldsymbol{m}_j\right\|^2}{t}\right) \tag{10}$$

(2) Compute the Laplacian matrices $\boldsymbol{L}$ and $\boldsymbol{H}$ for the within-class weight matrix $\boldsymbol{W}^c$ and the between-class weight matrix $\boldsymbol{B}$. We first define $\boldsymbol{D}$ and $\boldsymbol{E}$:

$$\boldsymbol{D}_{ii}^c=\sum_j\boldsymbol{W}_{ij}^c,\boldsymbol{E}_{ii}=\sum_j\boldsymbol{B}_{ij} \tag{11}$$

The specific forms of the Laplacian matrices $\boldsymbol{L}$ and $\boldsymbol{H}$ are then given by:

$$\boldsymbol{L}=\begin{bmatrix}\boldsymbol{D}^1-\boldsymbol{W}^1 & & & \\ & \boldsymbol{D}^2-\boldsymbol{W}^2 & & \\ & & \ddots & \\ & & & \boldsymbol{D}^C-\boldsymbol{W}^C\end{bmatrix},\boldsymbol{H}=\boldsymbol{E}-\boldsymbol{B} \tag{12}$$

The mapping matrix $\boldsymbol{A}$ in DLPP can be obtained by solving the following generalized eigenvalue problem:

$$\boldsymbol{XLX}^Ta=\lambda\boldsymbol{XHX}^Ta \tag{13}$$

3 Methods

To further enhance the performance of GLPP and improve the discriminability of the projection matrix, this study introduces label information from the data. Building upon GLPP, we propose two advanced methods: GDLPP and GSLPP.

3.1 GDLPP

For a point $\boldsymbol{X}_i \in \mathscr{G}(p, D)$ on a high-dimensional Grassmann Manifold, we aim to learn a mapping matrix $\boldsymbol{V} \in \mathbb{R}^{D\times d}$ that projects it onto a point $\boldsymbol{Y}_i \in \mathscr{G}(p, d)$ on a low-dimensional Grassmann Manifold. This mapping adheres to the general form of a linear transformation:

$$\boldsymbol{Y}_i = \boldsymbol{V}^T \boldsymbol{X}_i \tag{14}$$

Clearly, we cannot guarantee that the mapping matrix $\boldsymbol{V}$ remains column orthogonal throughout the iterative process, which in turn means that $\boldsymbol{V}^T\boldsymbol{X}_i \in \mathscr{G}(p, d)$ cannot be assured. To ensure that $\boldsymbol{V}^T\boldsymbol{X}_i$ lies on the Grassmann Manifold, it is necessary to perform a QR decomposition for $\boldsymbol{V}^T\boldsymbol{X}_i$.

$$\boldsymbol{Y}_i = \boldsymbol{V}^T\boldsymbol{X}_i = \boldsymbol{Q}_i\boldsymbol{R}_i \tag{15}$$

$\boldsymbol{Q}_i \in \mathscr{G}(p, d)$ is a column-orthogonal tall-and-thin matrix that represents the projection of the point $\boldsymbol{X}_i \in \mathscr{G}(p, D)$ from the high-dimensional Grassmann Manifold onto the low-dimensional Grassmann Manifold. Specifically:

$$\boldsymbol{Q}_i = \boldsymbol{V}^T\widetilde{\boldsymbol{X}}_i = \boldsymbol{V}^T\boldsymbol{X}_i\boldsymbol{R}_i^{-1} \tag{16}$$

Based on the general form of the objective function Eq. (9) in Euclidean space for DLPP, the objective function of GDLPP can be defined as follows:

$$\frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c}\sum_{j=1}^{N_c}\omega_{ij}^{c}\cdot\left\|\Pi\left(\boldsymbol{Q}_i^c\right)-\Pi\left(\boldsymbol{Q}_j^c\right)\right\|_F^2}{\sum_{i=1}^{C}\sum_{j=1}^{C}b_{ij}\cdot\left\|\Pi\left(\boldsymbol{F}_i\right)-\Pi\left(\boldsymbol{F}_j\right)\right\|_F^2} \tag{17}$$

where C denotes the number of classes in the data, and N_c represents the number of samples in class c. $\boldsymbol{Q}_i^c$ and $\boldsymbol{Q}_j^c$ refer to the embeddings of the i-th and j-th sample points within class c on the low-dimensional Grassmann Manifold, respectively. The variable ω_{ij}^c represents the within-class connection weight between $\boldsymbol{Q}_i^c$ and $\boldsymbol{Q}_j^c$. $\boldsymbol{F}_i$ and $\boldsymbol{F}_j$ denote the embeddings of the centroid points of class i and class j on the low-dimensional Grassmann Manifold, respectively, and b_{ij} indicates the between-class connection weight between class i and class j.

Combining Eqs. (16) and (17), the objective function of GDLPP can be reformulated as:

$$\min_{V} \frac{\sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \omega_{ij}^{c} \cdot \left\| \Pi\left(V^T \widetilde{X}_i^c\right) - \Pi\left(V^T \widetilde{X}_j^c\right) \right\|_F^2}{\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot \left\| \Pi\left(V^T Z_i\right) - \Pi\left(V^T Z_j\right) \right\|_F^2} \tag{18}$$

The form of Eq. (18) can be further simplified to:

$$\min_{V} \frac{\sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \omega_{ij}^{c} \cdot \left\| V^T G_{ij} V \right\|_F^2}{\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot \left\| V^T S_{ij} V \right\|_F^2} \tag{19}$$

where $G_{ij} = \widetilde{X}_i^c \widetilde{X}_i^{cT} - \widetilde{X}_j^c \widetilde{X}_j^{cT}$ and $S_{ij} = Z_i Z_i^T - Z_j Z_j^T$.

[28] and [32] provide iterative methods for solving the objective function, which differ somewhat from optimization on Riemannian manifolds. However, this simplified optimization approach has yielded effective results. By expanding the Frobenius norm term, the objective function of GDLPP can be further expressed as:

$$\mathscr{F}_{obj}(V) = \frac{\sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \omega_{ij}^{c} \cdot tr\left(V^T G_{ij} V \cdot V^T G_{ij} V\right)}{\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot tr\left(V^T S_{ij} V \cdot V^T S_{ij} V\right)} \tag{20}$$

Finally, the objective function of GDLPP in its iterative form can be described as:

$$\mathscr{F}_{obj}^{(t)}(V) = \frac{tr\left(V^T \left(\sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \omega_{ij}^{c} \cdot G_{ij} V^{(t-1)} \cdot V^{(t-1)T} G_{ij}\right) V\right)}{tr\left(V^T \left(\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot S_{ij} V^{(t-1)} \cdot V^{(t-1)T} S_{ij}\right) V\right)} \tag{21}$$

where $V^{(t-1)}$ represents the mapping matrix optimized at time $t-1$. The matrix G_{ij} can be computed using the normalized high-dimensional Grassmann points, while S_{ij} is calculated based on the mean of the high-dimensional Grassmann points across different classes.

Let

$$J = \sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{i=1}^{N_c} \omega_{ij}^{c} \cdot G_{ij} V^{(t-1)} \cdot V^{(t-1)T} G_{ij} \tag{22}$$

and

$$\boldsymbol{H} = \sum_{c=1}^{C} \sum_{i=1}^{C} b_{ij} \cdot \boldsymbol{S}_{ij} \boldsymbol{V}^{(t-1)} \cdot \boldsymbol{V}^{(t-1)T} \boldsymbol{S}_{ij} \tag{23}$$

the form of the objective function for GDLPP can be simplified to:

$$\min_{\boldsymbol{V}} \frac{tr\left(\boldsymbol{V}^T \boldsymbol{J} \boldsymbol{V}\right)}{tr\left(\boldsymbol{V}^T \boldsymbol{H} \boldsymbol{V}\right)} \tag{24}$$

In Eq. (24), $\boldsymbol{J}$ and $\boldsymbol{H}$ are real symmetric positive semidefinite matrices. The mapping matrix $\boldsymbol{V}$ in each iteration can be determined by the smallest eigenvector of the generalized eigenvalue problem.

The dimensionality reduction process of GDLPP can be summarized in Algorithm 1:

Algorithm 1 Discriminant locality preserving projection on grassmann manifold

Require: A set of Grassmann points $\mathcal{X} = \{\boldsymbol{X}_i\}_{i=1}^{N}, \boldsymbol{X}_i \in \mathcal{G}(p, D)$ with labels $\mathcal{L} = \{\boldsymbol{l}_i\}_{i=1}^{N}$ and target dimension d.
Ensure: Optimal Mapping Matrix $\boldsymbol{V} \in \mathbb{R}^{D \times d}$.

1: Initialize the mapping matrix $\boldsymbol{V}^{(0)} = \begin{bmatrix} \boldsymbol{I}_d \\ \boldsymbol{R} \end{bmatrix}$, where $\boldsymbol{I}_d$ is a d order identity matrix, and $\boldsymbol{R}$ is randomly initialized.
2: Compute the mean $\{\boldsymbol{F}_c\}_{c=1}^{C}$ for each class.
3: Compute the within-class weight matrix $\boldsymbol{W}$ and the between-class weight matrix $\boldsymbol{B}$, refining their respective roles and significance.
4: **while** not converged and not max epoch **do**
5: Normalize $\boldsymbol{X}_i^t = \boldsymbol{X}_i^{t-1} \boldsymbol{R}_i^{-1}$ by $\boldsymbol{Q}_i \boldsymbol{R}_i = \boldsymbol{V}^T \boldsymbol{X}_i^{t-1}$
6: Compute $\boldsymbol{J}$ and $\boldsymbol{H}$.
7: Resolve the generalized eigenvalue problem pertaining to $\boldsymbol{V}$.
8: **end while**

3.2 GSLPP

The fundamental idea behind the GSLPP algorithm is to incorporate the objective function of the GLPP algorithm as a regularization term in the GDLPP objective function, thereby constructing the objective function for GSLPP. The specific form of the GLPP objective function is provided in [28].

$$\mathcal{O}_{\text{GLPP}} = \min_{\boldsymbol{V}} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_{ij} \cdot \left\| \boldsymbol{Q}_i \boldsymbol{Q}_i^T - \boldsymbol{Q}_j \boldsymbol{Q}_j^T \right\|_F^2 \tag{25}$$

By combining the objective function of GLPP as presented in Eq. (25) with the objective function of GDLPP as given in Eq. (17), the objective function for GSLPP can be defined as:

$$\mathscr{O}_{\mathrm{GSLPP}} = \xi \cdot \mathscr{O}_{\mathrm{GDLPP}} \oplus (1-\xi) \cdot \mathscr{O}_{\mathrm{GLPP}} \tag{26}$$

where $\oplus$ represents the fusion of $\mathscr{O}_{\mathrm{GLPP}}$ and $\mathscr{O}_{\mathrm{GDLPP}}$, which is a generalized form of addition. The specific form of $\mathscr{O}_{\mathrm{GSLPP}}$ can be described as follows:

$$\mathscr{O}_{\mathrm{GSLPP}} = \min_{V} \frac{\xi \cdot \sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \omega_{ij}^{c} \cdot \left\| \Pi(\boldsymbol{Q}_i^c) - \Pi\left(\boldsymbol{Q}_j^c\right) \right\|_F^2 + (1-\xi) \cdot \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_{ij} \cdot \left\| \Pi(\boldsymbol{Q}_i) - \Pi(\boldsymbol{Q}_j) \right\|_F^2}{\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot \left\| \Pi(\boldsymbol{F}_i) - \Pi(\boldsymbol{F}_j) \right\|_F^2} \tag{27}$$

where N denotes the total number of samples, λ_{ij} represents the connection weights of the k-nearest neighbor graph constructed from all samples, and $\aleph$ signifies the regularization coefficient.

Assuming the sample set comprises of M labeled points denoted as $\{\boldsymbol{X}_i\}_{i=1}^{M}$ and $N-M$ unlabeled points denoted as $\{\boldsymbol{X}_i\}_{i=M+1}^{N}$, the within-class weight matrix $\boldsymbol{W}$ can be normalized into the following form, given $M = \sum_{c=1}^{C} N_c$:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}^1 & & & & \\ & \boldsymbol{W}^2 & & & \\ & & \ddots & & \\ & & & \boldsymbol{W}^C & \\ & & & & \boldsymbol{0} \end{bmatrix} \tag{28}$$

Then, the objective function of GSLPP can be rewritten as:

$$\mathscr{O}_{\mathrm{GSLPP}} = \min_{V} \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \left[\xi\omega_{ij} + (1-\xi)\lambda_{ij}\right] \cdot \left\| \Pi\left(\boldsymbol{Q}_i\right) - \Pi\left(\boldsymbol{Q}_j\right) \right\|_F^2}{\sum_{i=1}^{C} \sum_{j=1}^{C} b_{ij} \cdot \left\| \Pi\left(\boldsymbol{F}_i\right) - \Pi\left(\boldsymbol{F}_j\right) \right\|_F^2} \tag{29}$$

The form of the objective function for GSLPP in its iterative form is:

$$\mathscr{H}_{\mathrm{obj}}^{(t)}(\boldsymbol{V}) = \frac{tr\left(\boldsymbol{V}^T \left(\sum_{i=1}^{N} \sum_{j=1}^{N} \left[\xi\omega_{ij} + (1-\xi)\lambda_{ij}\right] \cdot \boldsymbol{G}_{ij} \boldsymbol{V}^{(t-1)} \boldsymbol{V}^{(t-1)T} \boldsymbol{G}_{ij} \right) \boldsymbol{V} \right)}{tr\left(\boldsymbol{V}^T \left(\sum_{i=1}^{C} \sum_{j=i}^{C} b_{ij} \cdot \boldsymbol{S}_{ij} \boldsymbol{V}^{(t-1)} \boldsymbol{V}^{(t-1)T} \boldsymbol{S}_{ij} \right) \boldsymbol{V} \right)} \tag{30}$$

If we set $\boldsymbol{J} = \sum_{i=1}^{N} \sum_{j=1}^{N} \left[\xi\omega_{ij} + (1-\xi)\lambda_{ij}\right] \cdot \boldsymbol{G}_{ij} \boldsymbol{V}^{(t-1)} \boldsymbol{V}^{(t-1)T} \boldsymbol{G}_{ij}$, the objective function of GSLPP can be simplified to the form given in Eq. (24). Thus, the optimization process of GSLPP is broadly similar to that of GDLPP, with differences primarily in the specific calculations involved. The optimization process for GSLPP can be summarized in Algorithm 2:

Algorithm 2 Semi-supervised locality preserving projection on grassmann manifold

Require: A set of Grassmann points $\mathcal{X} = \{\boldsymbol{X}_i\}_{i=1}^{M}, \boldsymbol{X}_i \in \mathcal{G}(p, D)$ with labels $\mathcal{L} = \{l_i\}_{i=1}^{M}$, a set of Grassmann unlabeled points $\{\boldsymbol{X}_i\}_{i=M+1}^{N}, \boldsymbol{X}_i \in \mathcal{G}(p, D)$ and target dimension d.

Ensure: Optimal Mapping Matrix $\boldsymbol{V} \in \mathbb{R}^{D\times d}$.

1: Initialize the mapping matrix $\boldsymbol{V}^{(0)} = \begin{bmatrix} \boldsymbol{I}_d \\ \boldsymbol{R} \end{bmatrix}$, where $\boldsymbol{I}_d$ is a d order identity matrix, and $\boldsymbol{R}$ is randomly initialized.
2: Compute the center $\{\boldsymbol{F}_c\}_{c=1}^{C}$ of each class of data through labeled data $\{\boldsymbol{X}_i\}_{i=1}^{M}$
3: Compute the within-class weight matrix $\boldsymbol{W}$ and between-class weight matrix $\boldsymbol{B}$ through labeled data $\{\boldsymbol{X}_i\}_{i=M+1}^{N}$;
4: Compute the weight matrix $\boldsymbol{\Lambda}$ based on the k-nearest neighbor graph using all data $\{\boldsymbol{X}_i\}_{i=1}^{N}$.
5: **while** not converged and not max epoch **do**
6: Normalize $\boldsymbol{X}_i^t = \boldsymbol{X}_i^{t-1}\boldsymbol{R}_i^{-1}$ by $\boldsymbol{Q}_i\boldsymbol{R}_i = \boldsymbol{V}^T\boldsymbol{X}_i^{t-1}$
7: Compute $\boldsymbol{J}$ and $\boldsymbol{H}$
8: Resolve the generalized eigenvalue problem pertaining to $\boldsymbol{V}$.
9: **end while**

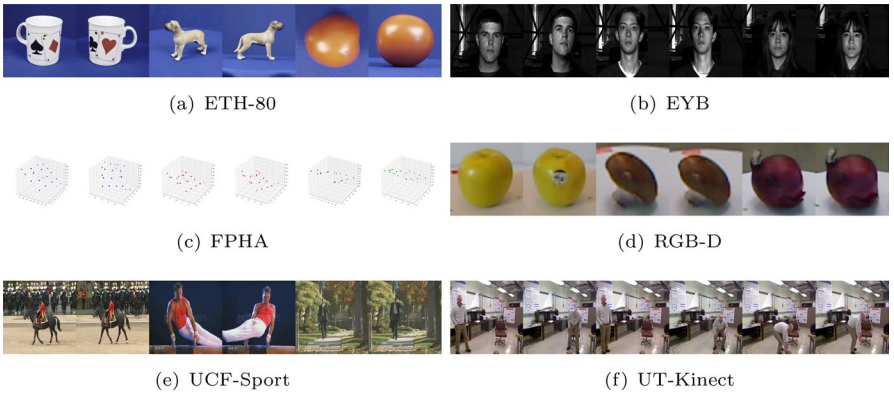
4 Experiments

To validate the advantages of GDLPP and GSLPP in image-set classification tasks, we conducted a comparative analysis with existing methods, including Grassmann KNN (GKNN), Grassmann SVM (GSVM) [43], GKDA, GLPP, GNPE, Grassmannian Adaptive Local Learn (GALL) [29], Nested Grassmann (NG) and Supervised Nested Grassmann (SNG) for Dimensionality Reduction [44], and Generalized Relevance Learning Grassmann Quantization (GRLGQ) [45]. It is noteworthy that 2 distinct weight calculation methods are proposed in [32] and applied to the GNPE, termed GNPE-I and GNPE-II, respectively. Furthermore, the Grassmann Network (GrNet) [8], proposed by Huang et al., is a feature extraction network grounded in the computational properties of the Grassmann Manifold. It has been extensively applied in various domains such as action recognition, skeleton recognition, and image-set recognition. GrNet stands as a paradigmatic representative of deep learning models on the Grassmann manifold.

The source codes for methods such as GLPP, GNPE, GALL, and GKDA are not available. We rigorously reproduced these methods based on the pseudocode provided in the original papers and obtained results that closely match those reported. It is important to note that we implemented GKDA by applying the Grassmann Projection Kernel to the Kernel Discriminant Analysis (KDA) [46] code provided in the original work. For methods with publicly available codes, such as NG, SNG, GRLGQ, and GSVM, we utilized the default parameter settings provided by the authors.

Table 1 Crucial parameters information of image-set datasets

Datasets	Samples	Train	Test	Classes	$\mathscr{G}(p, D)$
ETH-80	80	72	8	8	$\mathscr{G}(15, 400)$
EYB	252	224	28	28	$\mathscr{G}(5, 400)$
FPHA	1175	1130	45	45	$\mathscr{G}(10, 63)$
RGB-D	300	249	51	51	$\mathscr{G}(10, 400)$
UCF-Sport	150	137	13	13	$\mathscr{G}(11, 400)$
UT-Kinect	199	189	10	10	$\mathscr{G}(14, 400)$



(a) ETH-80 (b) EYB

(c) FPHA (d) RGB-D

(e) UCF-Sport (f) UT-Kinect

Fig. 2 Visualization of some images used in the study

To eliminate the potential impact of randomness on experimental outcomes, we conducted 10 repeated trials for both GDLPP, GSLPP, and the comparison methods. The classification performance was evaluated using the average results from these trials. Additionally, we replicated the experiments across various hardware configurations and consistently obtained similar conclusions.

4.1 Datasets

We analyzed the advantages of GDLPP and GSLPP in image-set classification across several datasets, including ETH-80 [47], Extended Yale B (EYB) [48], First-Person Hand Action (FPHA) [49], RGB-D [50], UCF-Sport [51], and UT-Kinect [52]. Examples of images from these datasets are displayed in Fig. 2, and the crucial parameters of each dataset are summarized in Table 1.

In the GDLPP experiments, one sample from each class in each dataset was randomly selected as the test set, while the remaining samples were used as the training set. For the GSLPP experiments, the division of the training and test sets followed the same approach. Additionally, half of the training data from each dataset was

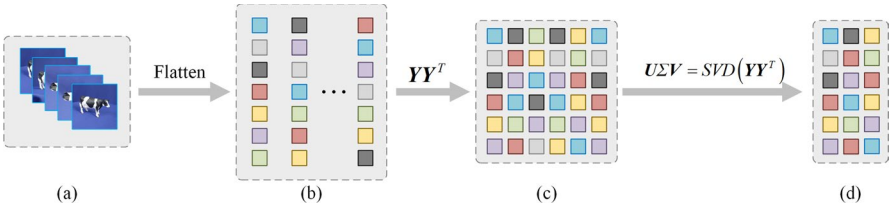


Fig. 3 Illustration of the sample modeling process on the grassmann manifold: **a** Images of an object captured from various viewpoints or consecutive frames from a video; **b** A set of vectors flattened from a single image; **c** A positive semi-definite feature matrix; **d** A sample point on the Grassmann Manifold

randomly selected as labeled data, while the other half was treated as unlabeled data. The specific numbers of training and test samples are detailed in Table 1.

The ETH-80 dataset comprises RGB images of 80 objects belonging to 8 different categories, captured from various angles. In this study, each RGB image was resized to a 20×20 grayscale image, with 41 images of a object modeled as an image-set sample.

The EYB dataset contains 16128 images of 28 individuals, each captured in 9 different poses and under 64 distinct lighting conditions. In this study, each image was resized to a 20×20 grayscale image. For each individual, the 64 images corresponding to each pose were grouped into an image-set sample.

FPHA is a large-scale first-person hand action benchmark, comprising 1175 action sequences across 45 gesture categories, collected from 6 subjects in 3 different visual environments. In this study, we utilized the provided 3D coordinates of 21 hand skeletal points to convert each frame into a 63-dimensional vector; each action sequence was modeled as an image-set sample.

The RGB-D dataset consists of 300 objects across 51 categories, with each object represented by hundreds of RGB images. In this study, each image was resized to a 20×20 grayscale image, and 40 images of each object were selected to construct an image-set sample.

UCF-Sport is an action recognition dataset comprising 150 videos across 13 action categories. Each video contains $< spanclass =' convertEndash' > 22 - 144 < /span >$ frames, which were converted into 20×20 grayscale images. Each video was modeled as an image-set sample.

The UT-Kinect dataset comprises videos, depth sequences, and skeletal data for 10 action categories, performed by 10 individuals using a Kinect device. Each individual executed each action twice, resulting in a total of 200 video. In this study, each frame of the videos was resized to a grayscale image, and each video was modeled as an image-set sample.

ETH-80 and RGB-D are image-set datasets, while EYB, UCF-Sport, and UT-Kinect are video datasets. FPHA consists of sequences of 3D coordinate vectors representing 21 key points of the hand. Samples from these datasets are illustrated in Fig. 2. Figure 3 depicts the process of modeling image-set samples on the Grassmann Manifold $\mathscr{G}(p, D)$.

Table 2 Comparison of average classification accuracy among image-set classification algorithms

Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
GKNN	0.8875	0.9464	0.4022	0.4000	0.4615	0.4900
GSVM	0.9125	0.9893	0.5556	0.0294	0.1077	0.3600
GKDA	0.9625	0.9893	0.4156	0.2784	0.3154	0.4300
GNPE-I	0.8875	0.9321	0.4711	0.6039	0.4769	0.4500
GNPE-II	0.8625	0.9750	0.4467	0.6510	0.5077	0.4300
GALL	0.9000	0.7607	0.2933	0.5059	0.3923	0.2300
GRLGQ	0.9000	0.9607	0.5267	0.4765	0.4462	0.5500
NG	0.8472	0.9444	0.6716	0.7124	0.5385	0.4000
SNG	0.8929	0.9439	0.5370	0.6275	0.5385	0.3167
GrNet	1.0000	1.0000	0.2000	0.3636	0.6667	0.4000
GLPP	0.8500	0.9107	0.4356	0.5882	0.4923	0.4300
GSLPP	**0.8875**	**0.9714**	**0.6311**	**0.7294**	**0.6538**	**0.4800**
GDLPP	**0.9750**	**0.9714**	**0.5622**	**0.7098**	**0.6308**	**0.4200**

The experimental results of the proposed methods are highlighted in bold

Figure 3 illustrates the process of modeling sample points on the Grassmann Manifold $\mathscr{G}(p, D)$. Taking the UCF-Sport dataset as an example, each frame of a video is first resized to a 20×20 grayscale image and then flattened into a vector $\boldsymbol{y}_i$. The flattened vectors from all frames of the video are combined into a matrix $\boldsymbol{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N\}$, and the feature matrix of the video is represented as the positive semi-definite real symmetric matrix $\boldsymbol{Y}\boldsymbol{Y}^T$. If $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V} = \text{SVD}(\boldsymbol{Y}\boldsymbol{Y}^T)$, the first p columns of U define a point on the Grassmann manifold $\mathscr{G}(p, D)$. For the FPHA dataset, which is not an image or video dataset, its modeling process begins from step (b) in Fig. 3.

4.2 Experimental setting

In the subsequent experiments, we evaluated the classification performance of GDLPP, GSLPP, and the comparative algorithms using 5 metrics: accuracy, precision, recall, $F1$ score, and Matthews correlation coefficient (MCC). For accuracy, precision, recall, and $F1$ score, a value closer to 1 indicates better classification performance, while a value closer to 0 reflects poorer performance. The value range of MCC is $[-1, 1]$, where a higher MCC value indicates better classification performance of the algorithm.

The dimension d of the Grassmann Manifold can be determined by the cumulative contribution rate of the eigenvectors of $\sum_{i=1}^{N} \boldsymbol{X}\boldsymbol{X}^T$. For a given cumulative contribution rate $\mu (0 < \mu < 1)$, d can be defined as:

$$d = \arg\min \left\{ 0 < d^* < D \middle| \sum_{i=1}^{d^*} \sigma_i \geq \mu \cdot \sum_{i=1}^{D} \sigma_i \right\} \tag{31}$$

Table 3 Comparison of average classification precision among image-set classification algorithms

Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
GKNN	0.8354	0.9274	0.3008	0.3113	0.3756	0.3867
GSVM	0.8688	0.9839	0.4454	0.0058	0.0331	0.2783
GKDA	0.9438	0.9839	0.3189	0.2006	0.2250	0.3650
GNPE-I	0.8375	0.9048	0.3713	0.4953	0.4077	0.3383
GNPE-II	0.7938	0.9625	0.3604	0.5570	0.4481	0.3250
GALL	0.8500	0.6801	0.2107	0.4181	0.3109	0.1375
GRLGQ	0.8500	0.9423	0.4262	0.3628	0.3385	0.4450
NG	0.7963	0.9226	0.5681	0.6242	0.4231	0.3241
SNG	0.8571	0.9175	0.4318	0.5349	0.4487	0.2500
GrNet	1.0000	1.0000	0.0625	0.2692	0.5556	0.1833
GLPP	0.7750	0.8696	0.3270	0.4994	0.4128	0.3300
GSLPP	**0.8313**	**0.9571**	**0.5178**	**0.6511**	**0.5849**	**0.4267**
GDLPP	**0.9625**	**0.9571**	**0.4673**	**0.6415**	**0.5205**	**0.3233**

The experimental results of the proposed methods are highlighted in bold

Table 4 Comparison of average classification recall among image-set classification algorithms

Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
GKNN	0.8875	0.9464	0.4022	0.4000	0.4615	0.4900
GSVM	0.9125	0.9893	0.5556	0.0294	0.1077	0.3600
GKDA	0.9625	0.9893	0.4156	0.2784	0.3154	0.4300
GNPE-I	0.8875	0.9321	0.4711	0.6039	0.4769	0.4500
GNPE-II	0.8625	0.9750	0.4467	0.6510	0.5077	0.4300
GALL	0.9000	0.7607	0.2933	0.5059	0.3923	0.2300
GRLGQ	0.9000	0.9607	0.5267	0.4765	0.4462	0.5500
NG	0.8472	0.9444	0.6716	0.7124	0.5385	0.4000
SNG	0.8929	0.9439	0.5370	0.6275	0.5385	0.3167
GrNet	1.0000	1.0000	0.1250	0.3077	0.6667	0.4000
GLPP	0.8500	0.9107	0.4356	0.5882	0.4923	0.4300
GSLPP	**0.8875**	**0.9714**	**0.6311**	**0.7294**	**0.6538**	**0.4800**
GDLPP	**0.9750**	**0.9714**	**0.5622**	**0.7098**	**0.6308**	**0.4200**

The experimental results of the proposed methods are highlighted in bold

4.3 Experimental results

We compared the classification performance of the proposed GDLPP and GSLPP methods with several image-set classification methods on the Grassmann Manifold, including GKNN, GSVM, GKDA, GLPP, GNPE, GALL, NG, SNG, and GRLGQ. The average classification accuracy, precision, recall, $F1$ score, and MCC for these algorithms across 6 datasets are recorded in Tables 2, 3, 4, 5 and 6, respectively.

Table 5 Comparison of average classification $F1$ score among image-set classification algorithms

Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
GKNN	0.8521	0.9333	0.3310	0.3376	0.4038	0.4183
GSVM	0.8833	0.9857	0.4781	0.0074	0.0396	0.3050
GKDA	0.9500	0.9857	0.3471	0.2115	0.2389	0.3867
GNPE-I	0.8542	0.9137	0.4024	0.5288	0.4282	0.3750
GNPE-II	0.8167	0.9667	0.3874	0.5863	0.4659	0.3600
GALL	0.8667	0.7056	0.2353	0.4444	0.3364	0.1657
GRLGQ	0.8667	0.9482	0.4568	0.3956	0.3690	0.4767
NG	0.8125	0.9299	0.6014	0.6521	0.4584	0.3481
SNG	0.8690	0.9260	0.4637	0.5635	0.4765	0.2694
GrNet	1.0000	1.0000	0.0833	0.2821	0.5833	0.2500
GLPP	0.8000	0.8827	0.3599	0.5271	0.4369	0.3617
GSLPP	**0.8500**	**0.9619**	**0.5537**	**0.6759**	**0.6049**	**0.4433**
GDLPP	**0.9667**	**0.9619**	**0.4968**	**0.6627**	**0.5551**	**0.3550**

The experimental results of the proposed methods are highlighted in bold

Table 6 Comparison of average classification MCC among image-set classification algorithms

Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
GKNN	0.8862	0.9462	0.3926	0.3916	0.4254	0.4495
GSVM	0.9101	0.9893	0.5506	0.0369	0.0794	0.3000
GKDA	0.9619	0.9893	0.4064	0.2867	0.3287	0.3749
GNPE-I	0.8833	0.9317	0.4627	0.6002	0.4451	0.4018
GNPE-II	0.8588	0.9750	0.4374	0.6477	0.4796	0.3775
GALL	0.8983	0.7577	0.2803	0.4999	0.3526	0.1513
GRLGQ	0.8974	0.9608	0.5200	0.4710	0.4191	0.5189
NG	0.8396	0.9439	0.6682	0.7101	0.5173	0.3435
SNG	0.8851	0.9439	0.5311	0.6243	0.5135	0.2497
GrNet	1.0000	1.0000	0.1581	0.3964	0.6708	0.3581
GLPP	0.8456	0.9107	0.4268	0.5841	0.4659	0.3788
GSLPP	**0.8847**	**0.9714**	**0.6271**	**0.7273**	**0.6408**	**0.4330**
GDLPP	**0.9746**	**0.9714**	**0.5560**	**0.7071**	**0.6153**	**0.3668**

The experimental results of the proposed methods are highlighted in bold

The specific values of the parameters in the experiment are shown in Table 7, and the detailed process of parameter analysis is presented in Sect. 4.4.

Based on the results presented in the tables, it is evident that GDLPP outperforms GLPP in both dimensionality reduction and classification across various datasets, demonstrating the effectiveness of incorporating discriminative information during the dimensionality reduction process. The superior classification performance of GDLPP compared to GKDA indicates that GDLPP better preserves the manifold and geometric structure of the data, highlighting its advantages in both dimensionality reduction and classification tasks. Furthermore,

Table 7 The target dimension d for dimensionality reduction on each dataset and the hyperparameter settings for GSLPP on each dataset

Datasets	ETH-80	EYB	FPHA	RGB-D	UCF-Sport	UT-Kinect
$\mathscr{G}(p,d)$	$\mathscr{G}(15,192)$	$\mathscr{G}(5,71)$	$\mathscr{G}(10,49)$	$\mathscr{G}(10,263)$	$\mathscr{G}(11,345)$	$\mathscr{G}(14,58)$
μ	0.95	0.95	0.95	0.95	0.99	0.75
ξ	0.45	0.50	0.50	0.50	0.65	0.45

GDLPP surpasses classifiers such as GKNN, GSVM, and GRLGQ on Grassmann Manifold, underscoring its robust feature extraction capabilities. When compared with dimensionality reduction algorithms such as GLPP, GALL, NG, and SNG, GDLPP also demonstrates superior performance in both dimensionality reduction and classification. GSLPP, by leveraging a small amount of labeled data, effectively preserves the discriminative structure of the data, thereby enhancing the classification capability of GLPP. The classification performance across 8 datasets reveals that GSLPP outperforms GLPP.

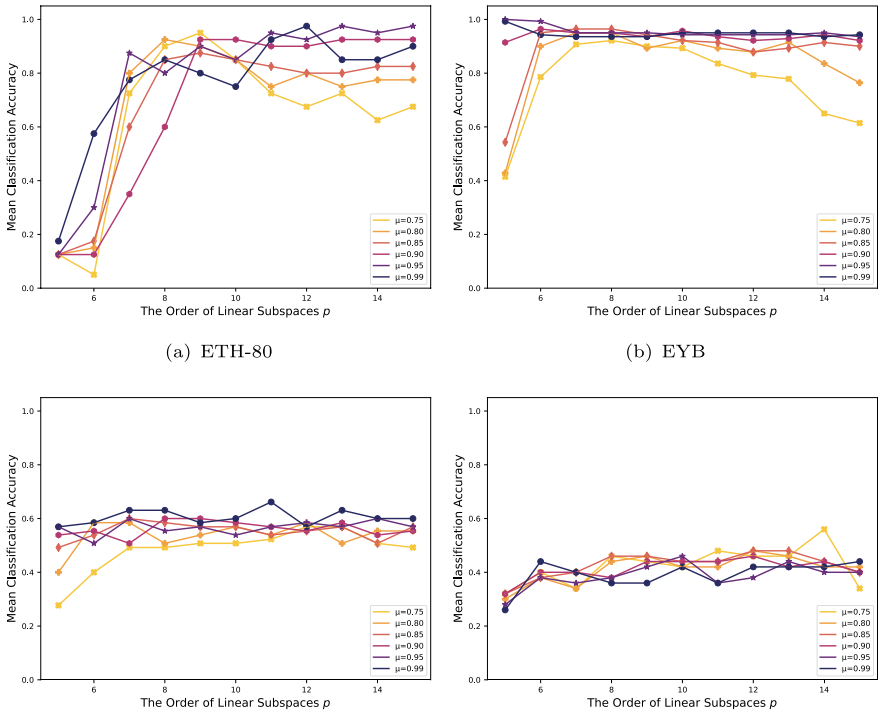
4.4 Parameters analysis

To analyze the sensitivity of the model to hyperparameters, this study evaluates the algorithm's performance by examining its average classification accuracy and conducts an analysis of the relevant hyperparameters. The GDLPP and GSLPP algorithms involve two critical parameters: the subspace order p of the Grassmann Manifold and the target dimension d of the low-dimensional Grassmann Manifold. In addition, the regularization coefficient ξ and the number of neighbors k are also important parameters for GSLPP. Given that multiple iterations of GDLPP and GSLPP are required during parameter analysis, we conducted this analysis on 4 relatively small datasets: ETH-80, EYB, UCF-Sport, and UT-Kinect to minimize computational costs.

Since the target dimension for dimensionality reduction in image-set datasets is determined by the cumulative contribution rate, we first analyze the classification performance of GDLPP and GSLPP under varying subspace dimensions and cumulative contribution rates. Drawing on previous research on dimensionality reduction methods on Grassmann Manifold, we examined the classification performance of GDLPP and GSLPP within a specific range of $p \in [5, 15]$ and $\mu \in [0.75, 0.99]$. The results are presented in Figs. 4 and 5.

To achieve optimal classification performance, GDLPP and GSLPP typically require retaining less energy in UT-Kinect datasets. In contrast, for the ETH-80, EYB, and UCF-Sport dataset, a higher cumulative contribution rate is generally necessary. The specific values of subspace order p and cumulative contribution rate μ for each dataset are detailed in Table 7.

In Fig. 6, we report the average classification accuracy of GSLPP under different regularization coefficients $\xi \in (0.05, 0.99)$. For the EYB datasets, the performance of GSLPP is relatively insensitive to the regularization coefficient ξ.



(a) ETH-80 (b) EYB

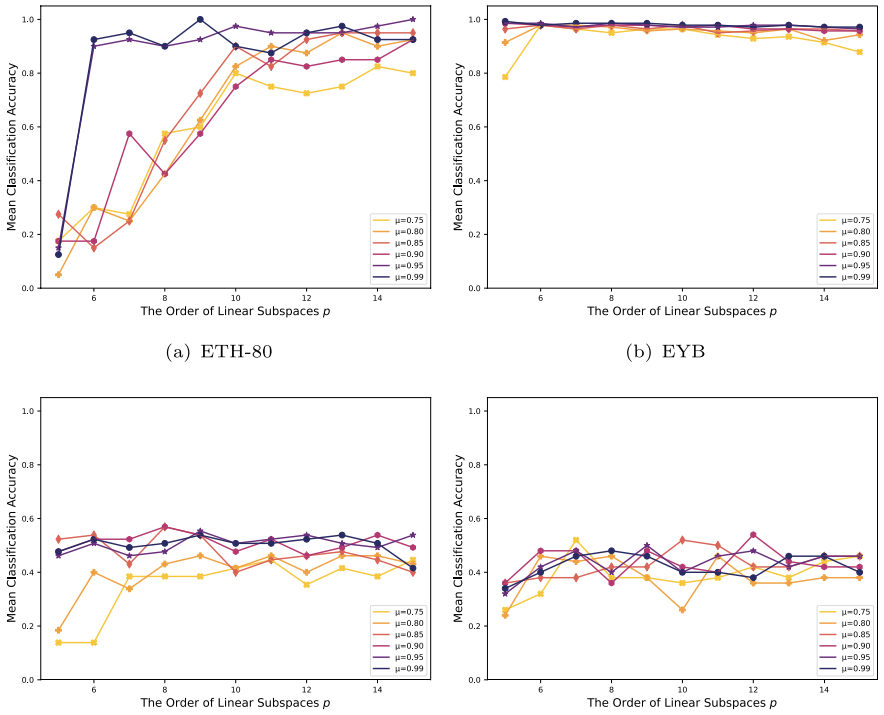
(c) UCF-Sport (d) UT-Kinect

Fig. 4 Analysis of subspace order p and cumulative contribution rate μ based on discriminant locality preserving projection on grassmann manifold (GDLPP). The figure shows the average classification accuracy of GDLPP on various datasets with different subspace orders $p \in [5, 15]$ and cumulative contribution rates $\mu \in [0.75, 0.99]$

However, for the ETH-80, UCF-Sport, and UT-Kinect datasets, the regularization coefficient typically has a significant impact on the classification performance of GSLPP.

We analyzed the impact of the number of neighbors k on the classification performance of GSLPP within a specified range $k \in [5, 15]$, with the results presented in Fig. 7. Our findings indicate that the effect of the neighbor count k on GSLPP's classification accuracy is minimal. Consequently, in experiments, the number of neighbors k for GSLPP was consistently set to 8.

Following the analysis of the hyperparameters for GDLPP and GSLPP, we recorded the specific settings for the subspace order p of the Grassmann Manifold, cumulative contribution rate μ, target dimension d, and the regularization coefficient ξ for GSLPP in Table 7.



(a) ETH-80 (b) EYB

(c) UCF-Sport (d) UT-Kinect

Fig. 5 Analysis of subspace order p and cumulative contribution rate μ based on semi-supervised locality preserving projection on grassmann manifold (GSLPP). The figure shows the average classification accuracy of GSLPP on various datasets with different subspace orders $p \in [5, 15]$ and cumulative contribution rates $\mu \in [0.75, 0.99]$

4.5 Time complexity analysis

For GLPP and GDLPP, the training set $\mathcal{X}_{train}$ consists of N samples belonging to C classes, with n samples in each class. For GSLPP, each class contains m labeled samples. In the experiments presented in this paper, each sample is projected from the high-dimensional Grassmann Manifold $\mathcal{G}(p, D)$ to the low-dimensional Grassmann Manifold $\mathcal{G}(p, d)$ using the mapping matrix $\boldsymbol{V}$.

Based on the aforementioned experimental conditions, we analyzed the time complexity of the GLPP, GDLPP, and GSLPP methods and evaluated their computational time on 6 datasets. In GDLPP and GSLPP, the computation of the within-class weight matrix $\boldsymbol{W}$ and the between-class weight matrix $\boldsymbol{B}$ is computationally expensive. Additionally, significant computational costs are associated with the calculation of the Fréchet means $\{\boldsymbol{F}\}_{c=1}^{C}$ of the samples, the normalization of the training samples $\mathcal{X}_{train} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_N\}$, and the computation of the within-class scatter matrix $\boldsymbol{J}$ and the between-class scatter matrix $\boldsymbol{S}$. The time complexity and

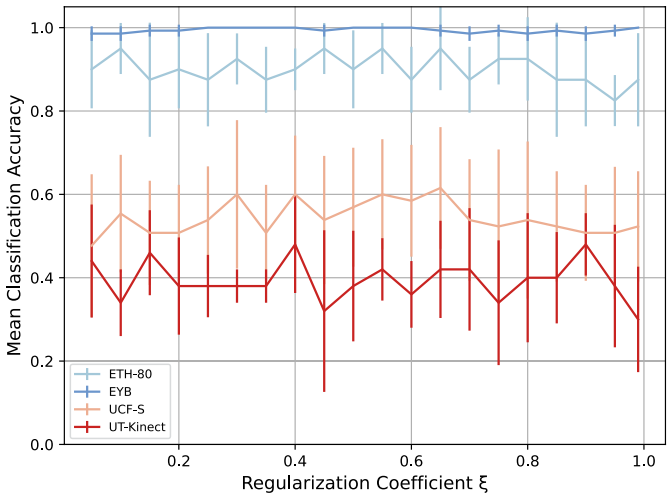


Fig. 6 Analysis of the regularization coefficient ξ in semi-supervised locality preserving projection on grassmann manifold (GSLPP). The figure shows the average classification accuracy of GSLPP under varying regularization coefficients $\xi \in [0.05, 0.99]$, with error bars visualizing the standard deviation of the average classification accuracy

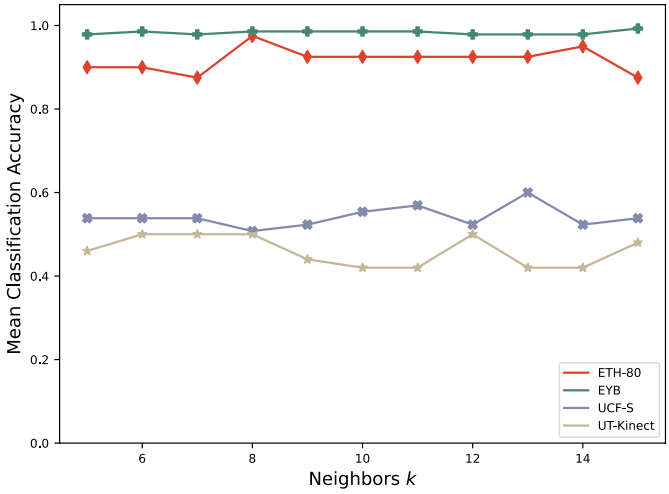


Fig. 7 Analysis of the number k of neighbors in semi-supervised locality preserving projection on grassmann manifold (GSLPP). The figure shows the average classification accuracy of GSLPP on different datasets with varying numbers of neighbors $k \in [5, 15]$

Table 8 Analysis of the time complexity for the main computational steps in GLPP, GDLPP, and GSLPP

Computation	Method	Complexity
$\boldsymbol{W}$	GLPP	$O(N^2D^2p)$
	GSLPP	$O(Cm^2D^2p)$+$O(N^2D^2p)$
	GDLPP	$O(Cn^2D^2p)$
$\boldsymbol{F}$	GSLPP	$O(CmD^2p) + O(CD^3)$
	GDLPP	$O(CnD^2p) + O(CD^3)$
$\boldsymbol{B}$	GSLPP	$O(C^2D^2p)$
	GDLPP	
Normalized	GLPP	$O(Ndp^2) + O(NDdp)$
	GSLPP	
	GDLPP	
$\boldsymbol{J}$	GLPP	$O(kND^2p)$+$O(kND^2d) + O(kND^3)$
	GSLPP	$O(kND^2p)$+$O(kND^2d) + O(kND^3)$ $+O(cm^2D^2p)$ $+O(cm^2D^2d) + O(cm^2D^3)$
	GDLPP	$O(cn^2D^2p)$+$O(cn^2D^2d) + O(cn^2D^3)$
$\boldsymbol{S}$	GSLPP	$O(C^2D^2p) + O(C^2D^2d) + O(C^2D^3)$
	GDLPP	

Table 9 Time costs of the main computational steps in GLPP, GDLPP, and GSLPP across different Datasets

Computation	Method	ETH-80	EYB	FPHA	RGBD	UCF-S	UT-Kinect
$\boldsymbol{W}$	GLPP	5.5140	52.9010	24.1211	67.5915	20.6692	39.4293
	GSLPP	5.6909	53.8274	23.6111	69.6479	20.0752	38.3911
	GDLPP	0.6715	1.7807	0.6315	1.4203	1.7608	3.6507
$\boldsymbol{F}$	GSLPP	0.5014	1.6060	0.1791	2.8258	0.7290	0.5810
	GDLPP	0.5270	1.7288	0.1992	3.0302	0.8644	0.7937
$\boldsymbol{B}$	GSLPP	0.0626	0.7726	0.0416	2.8242	0.1802	0.0984
	GDLPP	0.0647	0.7823	0.0455	2.7706	0.1709	0.1047
Normalized	GLPP	0.0347	0.0492	0.2121	0.1086	0.0882	0.0639
	GSLPP	0.0375	0.0462	0.2085	0.1208	0.0672	0.0722
	GDLPP	0.0387	0.0626	0.2074	0.0993	0.0750	0.0698
$\boldsymbol{J}$	GLPP	3.4348	9.1416	0.7515	12.6679	7.9937	7.9230
	GSLPP	3.3998	9.6861	0.7830	14.1471	8.0471	8.7512
	GDLPP	3.2898	12.9647	2.1863	7.5190	11.1457	17.9747
$\boldsymbol{S}$	GSLPP	0.3218	3.9437	0.1503	15.4448	1.0750	0.4431
	GDLPP	0.3048	6.4889	0.1507	15.4033	1.0344	0.4783

Table 10 Computational time costs of GLPP, GDLPP, and GSLPP across different datasets

Method	ETH-80	EYB	FPHA	RGB-D	UCF-Sport	UT-Kinect
GLPP	13.6382	76.5157	26.6414	94.1661	37.0132	56.8281
GSLPP	14.4236	91.5385	27.2488	131.6520	41.6818	59.2968
GDLPP	9.2156	29.0903	6.1292	55.7781	28.0304	39.6199

runtime of GLPP, GDLPP, and GSLPP on each dataset are summarized in Tables 8 and 9, respectively.

As shown in the results presented in Table 9, the construction of the weight matrix $\boldsymbol{W}$ in GLPP and GSLPP incurs significant computational costs. In contrast, the cost of constructing the weight matrix in GDLPP is considerably lower compared to that in GLPP and GSLPP.

As indicated in the results from Table 8, the construction costs of the within-class scatter matrix $\boldsymbol{J}$ and the between-class scatter matrix $\boldsymbol{S}$ in GDLPP and GSLPP are directly related to the number of samples within each class. The greater the number of samples per class, the higher the time complexity of GDLPP and GSLPP.

Table 10 presents the total computation time of GLPP, GDLPP, and GSLPP across 6 datasets. As shown in the results, the computational cost of GDLPP is significantly lower than that of GLPP and GSLPP, while the cost of GSLPP is slightly higher than that of GLPP. Overall, the results reported in Table 10 align with the conclusions drawn from the analyses in Tables 8 and 9. GDLPP not only significantly enhances the feature extraction and classification capabilities of GLPP but also substantially reduces its computational cost.

5 Conclusion

This study proposes Discriminant Locality Preserving Projection and Semi-Supervised Locality Preserving Projection on Grassmann Manifold based on label information from the data. Both GDLPP and GSLPP make full use of label information while preserving the manifold structure of the data. Compared to the unsupervised GLPP, the mapping matrices learned through GDLPP and GSLPP offer higher discriminability. GDLPP and GSLPP enhance the classification ability and feature extraction performance of GLPP.

The proposed GSLPP is an attempt to integrate Semi-Supervised Learning with dimensionality reduction techniques on the Grassmann manifold. Experimental results demonstrate that Semi-Supervised Learning on Grassmann Manifold is effective, with GSLPP outperforming the unsupervised GLPP in classification performance. GSLPP provides a valuable reference for the future development of semi-supervised dimensionality reduction techniques on Grassmann Manifold.

While our work has enhanced the performance of GLPP, there remain areas for further improvement. Obtaining data labels is a challenging task, and in future

work, we aim to enhance the classification capability of GLPP through self-supervised methods, without relying on data labels.

Acknowledgements This work was supported by Chongqing Municipal Education Commission under grant KJZD-K202100505, Chongqing Science and Technology Bureau under grant cstc2020jscx-msxmX0190, the Ministry of Education of China under grant 20YJAZH084.

Author contributions Benchao Li contributed to software and writing—original draft. Ting Wang contributed to methodology, writing—review & editing, and visualization. Ruisheng Ran contributed to conceptualization and supervision.

Data availability Data will be made available on request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Wong Y-C (1967) Differential geometry of grassmann manifolds. Proc Natl Acad Sci 57(3):589–594
2. Absil P-A, Mahony R, Sepulchre R (2004) Riemannian geometry of grassmann manifolds with a view on algorithmic computation. Acta Appl Math 80:199–220
3. Pennec X, Fillard P, Ayache N (2006) A riemannian framework for tensor computing. Int J Comput Vision 66:41–66
4. Harandi M, Salzmann M, Hartley R (2017) Dimensionality reduction on spd manifolds: The emergence of geometry-aware methods. IEEE Trans Pattern Anal Mach Intell 40(1):48–62
5. Gao W, Ma Z, Xiong C, Gao T (2023) Dimensionality reduction of spd data based on riemannian manifold tangent spaces and local affinity. Appl Intell 53(2):1887–1911
6. Rapcsák T (2002) On minimization on stiefel manifolds. Eur J Oper Res 143(2):365–376
7. Gao B, Son NT, Absil P-A, Stykel T (2021) Riemannian optimization on the symplectic stiefel manifold. SIAM J Optim 31(2):1546–1575
8. Huang Z, Wu J, Van Gool L (2018) Building deep networks on grassmann manifolds. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32
9. Shigenaka R, Raytchev B, Tamaki T, Kaneda K (2012) Face sequence recognition using grassmann distances and grassmann kernels. In: The 2012 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. IEEE
10. Turaga P, Veeraraghavan A, Srivastava A, Chellappa R (2011) Statistical computations on grassmann and stiefel manifolds for image and video-based recognition. IEEE Trans Pattern Anal Mach Intell 33(11):2273–2286
11. Dong X, Frossard P, Vandergheynst P, Nefedov N (2013) Clustering on multi-layer graphs via subspace analysis on grassmann manifolds. IEEE Trans Signal Process 62(4):905–918
12. Anowar F, Sadaoui S, Selim B (2021) Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). Comput Sci Rev 40:100378
13. Cunningham JP, Ghahramani Z (2015) Linear dimensionality reduction: survey, insights, and generalizations. J Machine Learn Res 16(1):2859–2900
14. Pearson K (1901) Liii. on lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin philosophical magazine and journal of science 2(11), 559–572
15. Fisher RA (1936) The use of multiple measurements in taxonomic problems. Ann Eugen 7(2):179–188
16. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326

17. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6):1373–1396
18. McInnes L, Healy J, Melville J (2018) Umap: Uniform manifold approximation and projection for dimension reduction. ArXiv e-prints arXiv:1802.03426 [stat.ML]
19. Vural E, Guillemot C (2016) Out-of-sample generalizations for supervised manifold learning for classification. IEEE Trans Image Process 25(3):1410–1424
20. Taşkin G, Crawford MM (2019) An out-of-sample extension to manifold learning via meta-modeling. IEEE Trans Image Process 28(10):5227–5237
21. He X, Niyogi P (2003) Locality preserving projections. Advances in neural information processing systems **16**
22. Yu W, Wang R, Nie F, Wang F, Yu Q, Yang X (2018) An improved locality preserving projection with l1-norm minimization for dimensionality reduction. Neurocomputing 316:322–331
23. Wang S-J, Chen H-L, Peng X-J, Zhou C-G (2011) Exponential locality preserving projections for small sample size problem. Neurocomputing 74(17):3654–3662
24. Wang A, Zhao S, Liu J, Yang J, Liu L, Chen G (2020) Locality adaptive preserving projections for linear dimensionality reduction. Expert Syst Appl 151:113352
25. Chen S, Zhao H, Kong M, Luo B (2007) 2d-lpp: A two-dimensional extension of locality preserving projections. Neurocomputing 70(4–6):912–921
26. Hu D, Feng G, Zhou Z (2007) Two-dimensional locality preserving projections (2dlpp) with its application to palmprint recognition. Pattern Recogn 40(1):339–342
27. Chen W-J, Li C-N, Shao Y-H, Zhang J, Deng N-Y (2019) 2drlpp: Robust two-dimensional locality preserving projection with regularization. Knowl-Based Syst 169:53–66
28. Wang B, Hu Y, Gao J, Sun Y, Ali M, Chen H, Yin B (2017) Locality preserving projections for grassmann manifold. In: International Joint Conference on Artificial Intelligence, pp. 2893–2900
29. Wei D, Shen X, Sun Q, Gao X, Ren Z (2024) Learning adaptive grassmann neighbors for image-set analysis. Expert Syst Appl 247:123316
30. Hamm J, Lee DD (2008) Grassmann discriminant analysis: a unifying view on subspace-based learning. In: Proceedings of the 25th International Conference on Machine Learning, pp. 376–383
31. Harandi MT, Sanderson C, Shirazi S, Lovell BC (2011) Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In: CVPR 2011, pp. 2705–2712. IEEE
32. Wei D, Shen X, Sun Q, Gao X, Ren Z (2022) Neighborhood preserving embedding on grassmann manifold for image-set analysis. Pattern Recogn 122:108335
33. Zhang D, Zhou Z-H, Chen S (2007) Semi-supervised dimensionality reduction. In: Proceedings of the 2007 SIAM International Conference on Data Mining, pp. 629–634. SIAM
34. Song Y, Nie F, Zhang C, Xiang S (2008) A unified framework for semi-supervised dimensionality reduction. Pattern Recogn 41(9):2789–2799
35. Harandi M, Sanderson C, Shen C, Lovell BC (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
36. Yu H, Xia K, Jiang Y, Qian P (2020) Fréchet mean-based grassmann discriminant analysis. Multimedia Syst 26(1):63–73
37. Zhang S, Yu G (2010) Semi-supervised locality preserving projections with compactness enhancement. In: 2010 International Conference on Educational and Information Technology, vol. 2, pp. 2–460. IEEE
38. Raducanu B, Dornaika F (2012) A supervised non-linear dimensionality reduction approach for manifold learning. Pattern Recogn 45(6):2432–2444
39. Vogelstein JT, Bridgeford EW, Tang M, Zheng D, Douville C, Burns R, Maggioni M (2021) Supervised dimensionality reduction for big data. Nat Commun 12(1):2872
40. Yu W, Teng X, Liu C (2006) Face recognition using discriminant locality preserving projections. Image Vis Comput 24(3):239–248
41. Lu G-F, Lin Z, Jin Z (2010) Face recognition using discriminant locality preserving projections based on maximum margin criterion. Pattern Recogn 43(10):3572–3579
42. Lu G-F, Zou J, Wang Y (2016) L1-norm and maximum margin criterion based discriminant locality preserving projections via trace lasso. Pattern Recogn 55:207–214
43. Al-Samhi W, Al-Soswa M, Al-Dhabi Y (2021) Time series data classification on grassmann manifold. In: Journal of Physics: Conference Series, vol. 1848, p. 012037. IOP Publishing

44. Yang C-H, Vemur, BC (2021) Nested grassmanns for dimensionality reduction with applications to shape analysis. In: International Conference on Information Processing in Medical Imaging, pp. 136–149. Springer
45. Mohammadi M, Babai M, Wilkinson M (2024) Generalized relevance learning grassmann quantization. arXiv preprint arXiv:2403.09183
46. Liu Q, Lu H, Ma S (2004) Improving kernel fisher discriminant analysis for face recognition. IEEE Trans Circuits Syst Video Technol 14(1):42–49
47. Leibe B, Schiele B (2003) Analyzing appearance and contour based methods for object categorization. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., vol. 2, p. 409. IEEE
48. Georghiades AS, Belhumeur PN, Kriegman DJ (2001) From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans Pattern Anal Mach Intell 23(6):643–660
49. Garcia-Hernando G, Yuan S, Baek S, Kim T-K (2018) First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 409–419
50. Lai K, Bo L, Ren X, Fox D (2012) Detection-based object labeling in 3d scenes. In: 2012 Ieee International Conference on Robotics and Automation, pp. 1330–1337. IEEE
51. Rodriguez MD, Ahmed J, Shah M (2008) Action mach a spatio-temporal maximum average correlation height filter for action recognition. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE
52. Xia L, Chen CC, Aggarwal J (2012) View invariant human action recognition using histograms of 3d joints. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference On, pp. 20–27. IEEE