Signal Processing 235 (2025) 110028

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

# Image-set classification using Discriminant Neighborhood Preserving Embedding on Grassmann manifold

Benchao Li [a], Yuanyuan Zheng [a], Ruisheng Ran [a,*], Bin Fang [b]

[a] College of Computer and Information Science, Chongqing Normal University, Chongqing, 401331, China
[b] College of Computer Science, Chongqing University, Chongqing, 400044, China

ARTICLE INFO

*Keywords:*
Grassmann manifold
Image-set classification
Neighborhood Preserving Embedding
Discriminant learning
Semi-supervised learning

ABSTRACT

Existing supervised dimensionality reduction techniques on the Grassmann manifold fail to accurately preserve the manifold and local structures of samples. To address this issue, this study introduces Discriminant Neighborhood Preserving Embedding on Grassmann manifold (GDNPE). Furthermore, to tackle the common problem of insufficient data labels, this paper proposes Semi-Supervised Neighborhood Preserving Embedding on Grassmann manifold (GSNPE). The proposed GDNPE and GSNPE methods are applied to the task of image-set classification in this work. Experimental evaluations conducted on a diverse array of benchmark image-set datasets have conclusively demonstrated the superiority of both GDNPE and GSNPE over existing image-set classification methods. These advanced techniques exhibit remarkable performance in classification and feature extraction endeavors, highlighting their efficacy and potential as formidable tools for image-set analysis.

## 1. Introduction

An image-set refers to a collection of images that share some inherent relationships. These image-sets can be unordered collections, such as images of an object captured under different lighting conditions, or time-ordered sequences, like video frames of a person walking captured by a surveillance camera. In image-set applications, classification is a key task. Recent studies have shown that image-sets often lie on a low-dimensional manifold with nonlinear geometric structures, making traditional Euclidean representation models less effective. Nonlinear Riemannian manifolds [1,2], such as the Symmetric Positive Definite (SPD) manifold, Grassmann manifold, and Gaussian manifold, have gained significant attention in image-set research. They can more accurately capture the geometric relationships between different feature regions within image-sets. Moreover, compared to Euclidean metrics, Riemannian metrics can more precisely reflect the geodesic similarities between points, making Riemannian manifold-based approaches a hotspot in image-set analysis.

The Grassmann manifold [1] is an effective subspace structure for capturing the intrinsic structure of image-sets and is widely used in image-set classification [3–6], and clustering [7–9] tasks. However, image-sets are typically high-dimensional, leading to significant computational costs for Grassmann manifold-based methods. Additionally, high-dimensional data often contains a substantial amount of redundant information, which can severely impact the performance of manifold models [10]. Moreover, when image-sets are large, manifold models face increased time and space complexity. Therefore, developing dimensionality reduction methods for manifold data that preserve the underlying manifold structure is of great research significance and practical value. Such methods can reduce computational complexity while extracting meaningful features.

Dimensionality reduction techniques [11,12] are widely used in machine learning to remove noise and redundant information from data. They also effectively reduce computational costs, making them essential in many machine learning tasks. Over the past few decades, numerous powerful dimensionality reduction methods have been developed, with Principal Component Analysis (PCA) [13] and Linear Discriminant Analysis (LDA) [14] being prominent examples. However, PCA, LDA, and their variants can only capture linear mappings between high-dimensional and low-dimensional spaces. To uncover the nonlinear relationships between these spaces, researchers have developed a range of manifold learning algorithms, such as Locally Linear Embedding (LLE) [15], Laplacian Eigenmaps (LE) [16], and t-distributed Stochastic Neighbor Embedding (t-SNE) [17]. However, the nonlinear mappings learned by these algorithms are implicit, making it difficult to embed new samples into the low-dimensional space. This issue is known as the Out-of-Sample Extension Problem [18,19]. Neighborhood Preserving Embedding (NPE) [20] addresses this limitation by resolving

* Corresponding author.
*E-mail addresses:* 2022110516061@stu.cqnu.edu.cn (B. Li), 2023110516071@stu.cqnu.edu.cn (Y. Zheng), rshran@cqnu.edu.cn (R. Ran), fb@cqu.edu.cn (B. Fang).

https://doi.org/10.1016/j.sigpro.2025.110028
Received 30 September 2024; Received in revised form 6 March 2025; Accepted 31 March 2025
Available online 12 April 2025

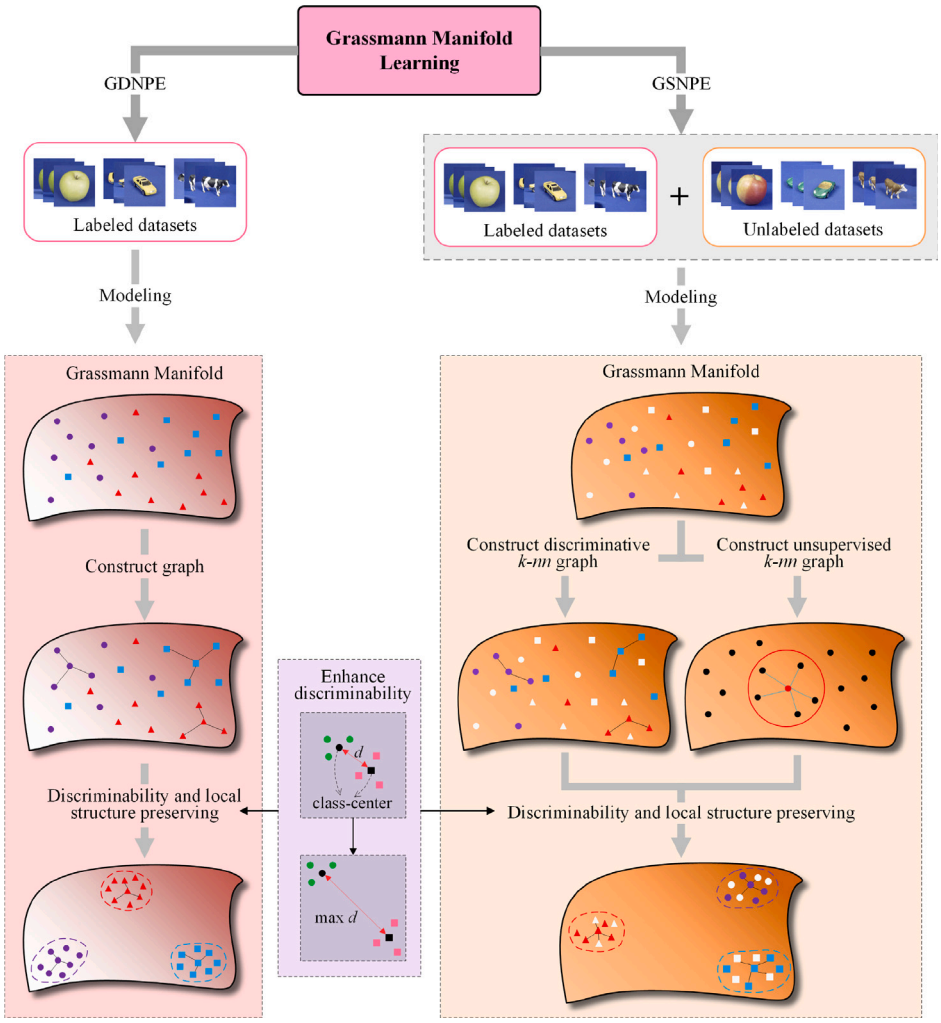



**Fig. 1.** Conceptual illustration of the proposed discriminant neighborhood preserving embedding on Grassmann Manifold and semi-supervised neighborhood preserving embedding on Grassmann Manifold.

the Out-of-Sample Extension Problem and learning a linear mapping of the nonlinear structure, making it broadly applicable in machine learning tasks.

However, the aforementioned algorithms are designed for vector data and are not suitable for dimensionality reduction tasks involving image-set data. As a result, researchers have extended some popular dimensionality reduction techniques from Euclidean space to the Grassmann manifold. Early approaches to dimensionality reduction on the Grassmann manifold involved embedding points from the Grassmann manifold into a Reproducing Kernel Hilbert Space (RKHS). These approaches then applied Euclidean methods for dimensionality reduction and classification tasks. Representative algorithms of this approach include Grassmann Dictionary Learning (KGDL) [21], Grassmann Discriminant Analysis (GKDA) [22], and Graph Embedded Discriminant Analysis on the Grassmann manifold (GEDA) [23]. While these algorithms effectively leverage discriminative information, they fall short in accurately preserving the manifold structure of the samples.

Another approach to dimensionality reduction on the Grassmann manifold is based on the "From manifold to manifold" concept. Representative algorithms embodying this approach include Locality Preserving Projection on Grassmann manifold (GLPP) [24], Neighborhood Preserving Embedding on Grassmann manifold (GNPE) [25], and Grassmann Adaptive Local Learning (GALL) [26]. These methods excel at preserving the intricate manifold structure inherent in the data, ensuring a more faithful representation of the geometric properties. An outstanding classification framework necessitates accurately reflecting the inter-class and intra-class structures of data through a mapping matrix. Nevertheless, the unsupervised nature of these algorithms confines their discriminative capabilities, overlooking the essential label information of the data. Overall, it is necessary to develop a dimensionality reduction algorithm based on the "From manifold to manifold" concept that effectively utilizes label information.

Discriminant learning [27,28] is a machine learning task that enhances algorithm performance by utilizing label information, which significantly improves the feature extraction capabilities of dimensionality reduction algorithms. Various forms of Discriminant Neighborhood Preserving Embedding (DNPE) have been widely applied in fields such as face recognition and hyperspectral image recognition, achieving significant improvements. Wen et al. introduced Preserving Neighborhood Discriminant Embedding (PNDE) [29] for feature extraction in hyperspectral images. PNDE efficiently captures the local geometric structures of data and fully utilizes label information, effectively reducing the dimension and achieving high classification accuracy. Building on the successful experiences of PNDE and the strong performance of GNPE in image-set classification tasks, this paper proposes a Discriminant Neighborhood Preserving Embedding on Grassmann manifold (GDNPE) based on the "From manifold to manifold" concept. By maximizing inter-class distances and minimizing intra-class distances, GDNPE enhances the discriminability of the projection matrix and increases the distinguishability of the low-dimensional data, thereby achieving better classification accuracy.

In discriminant learning, acquiring label information can be challenging, and semi-supervised learning addresses the issue of insufficient labels. In recent years, dimensionality reduction techniques based on semi-supervised learning [30,31] have seen significant development. Mehdizadeh et al. [32] proposed a Semi-Supervised Neighborhood Preserving Embedding (SNPE) based on LDA and NPE. They integrated the objective function of NPE as a regularization term into the LDA objective function, resulting in the SNPE objective function. Compared to traditional NPE, SNPE can better preserve the geometric structure of data, enhancing its discriminability and achieving higher classification accuracy. Drawing on the successful experience of SNPE and the practical issue of limited image-set label availability, we propose a Semi-Supervised Neighborhood Preserving Embedding on Grassmann manifold (GSNPE). GSNPE extends the concept of GDNPE by enhancing the discriminability of the mapping matrix using labeled data and constructing a neighborhood graph with all samples to learn the local structure of the data. The fundamental concepts of GDNPE and GSNPE are schematically illustrated in Fig. 1.

The contributions of this paper are as follows:

- We propose a supervised dimensionality reduction method on the Grassmann manifold. By employing Discriminant Neighborhood Preserving Embedding based on the Grassmannian metric, this method enhances feature extraction for image-set data, thus achieving improved classification performance.
- We introduce a semi-supervised dimensionality reduction method on the Grassmann manifold. This method addresses the issue of insufficient labels in discriminant learning tasks by effectively leveraging a small amount of labeled data, thereby significantly improving the performance of unsupervised dimensionality reduction techniques on the Grassmann manifold.
- This paper pioneers the integration of semi-supervised dimensionality reduction within the context of the Grassmann manifold, thereby introducing a novel framework for semi-supervised dimensionality reduction in this geometric domain. Our contribution is poised to serve as a valuable reference for the prospective evolution of semi-supervised dimensionality reduction techniques on the Grassmann manifold.

The rest of this paper is organized as follows: Section 2 introduces the basic concepts of the Grassmann manifold and NPE; Section 3 provides details of the proposed GDNPE and GSNPE algorithms and their implementation; Section 4 discusses the experimental results of the algorithms and compares them with similar methods; and Section 5 provides a summary and future directions.

## 2. Preliminary knowledge

### 2.1. Grassmann manifold

The Grassmann manifold, an abstract concept denoted as $\mathcal{G}(p,d)$, $(0 \le p \le d)$, encompasses all $p$-dimensional subspaces within a $d$-dimensional Euclidean space. A concise explanation would be that the Grassmann manifold comprises the equivalence classes of all tall-and-thin orthogonal matrices under the action of the orthogonal group $\mathcal{O}(p)$ of order $p$, formulated as follows:

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

#### 2.1.1. Projection metric on Grassmann manifold

Embedding the Grassmann manifold points into the space of symmetric matrices enables the acquisition of Euclidean metrics, a common approach in its study [21]. This method has been adopted in various research efforts, including GDL, GLPP, and GNPE. For points $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathcal{G}(p,d)$ on the Grassmann manifold:

$$dist_g\left(\boldsymbol{X}_1, \boldsymbol{X}_2\right) = \frac{1}{\sqrt{2}} \left\| \Omega(\boldsymbol{X}_1) - \Omega(\boldsymbol{X}_2) \right\|_F \tag{2}$$

where,

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

#### 2.1.2. Extrinsic mean on Grassmann manifold

Yu et al. proposed the Fréchet Mean-based Grassmann Discriminant Analysis (FM-GDA) [33] by leveraging the extrinsic mean on the Grassmann manifold, achieving promising results in classification on image-set datasets. FM-GDA provides a valuable reference for the development of supervised dimensionality reduction techniques on the Grassmann manifold. By embedding the Grassmann manifold into a vector space, the extrinsic mean $\boldsymbol{M}^*$ on the Grassmann manifold can be computed [34]. In this study, the projection metric on the Grassmann manifold is employed to describe the distances between samples. Correspondingly, the extrinsic mean on the Grassmann manifold is utilized to characterize the centroid of a class of samples. Given a sample set $\left\{\boldsymbol{X}_i\right\}_{i=1}^N, \boldsymbol{X}_i \in \mathcal{G}(p,D)$, we define:

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

the extrinsic mean of samples can be defined as the $p$ largest eigenvectors of $\boldsymbol{T}$.

### 2.2. GNPE

Neighborhood Preserving Embedding on Grassmann manifold (GNPE) is a newly proposed dimensionality reduction technique, with broad applications in image-set classification and clustering tasks. Given a sample set $\left\{\boldsymbol{X}_i\right\}_{i=1}^N, \boldsymbol{X}_i \in \mathcal{G}(p,D)$ on the high-dimensional Grassmann manifold, GNPE maps it to the low-dimensional Grassmann manifold $\mathcal{G}(p,d)$ using a projection matrix $\boldsymbol{A}$. To more accurately preserve the local relationships between samples on the high-dimensional manifold within the low-dimensional space, GNPE defines its objective function as follows:

$$\mathbb{O}_{GNPE} = \min_{\boldsymbol{A}} \sum_{i=1}^{N} \left\| \boldsymbol{Q}_i\boldsymbol{Q}_i^T - \sum_{j=1}^{N} s_{ij}\boldsymbol{Q}_j\boldsymbol{Q}_j^T \right\|_F^2 \tag{5}$$

where $\boldsymbol{Q}_i$ represents the projection of a point $\boldsymbol{X}_i$ from the high-dimensional manifold into the low-dimensional manifold, and $s_{ij}$ denotes the connection coefficient between samples. GNPE offers two strategies for calculating $s_{ij}$, resulting in two variations of the method: GNPE-I and GNPE-II, depending on the strategies used.

In GNPE-I, given a sample set $\{\boldsymbol{X}_i\}_{i=1}^N$, $s_{ij}$ can be optimized by minimizing the objective function:

$$\min_{\boldsymbol{S}} \sum_i \|\boldsymbol{X}_i\boldsymbol{X}_i^T - \sum_{j\in N_k(\boldsymbol{X}_i)} s_{ij}\boldsymbol{X}_j\boldsymbol{X}_j^T\|_F^2 \quad \text{s.t.} \quad \boldsymbol{s}^T\boldsymbol{1} = \boldsymbol{1} \tag{6}$$

where $N_k(\boldsymbol{X}_i)$ denotes the index list of the neighboring points of sample $\boldsymbol{X}_i$. In GNPE-II, given a sample set $\left\{\boldsymbol{X}_i\right\}_{i=1}^N$, $s_{ij}$ is calculated by minimizing the objective function:

$$\min_{\boldsymbol{S}} \sum_{i=1}^{N} \left(\frac{1}{2}\|\boldsymbol{X}_i\boldsymbol{X}_i^T - \sum_{j=1}^{N} s_{ij}\boldsymbol{X}_j\boldsymbol{X}_j^T\|_F^2 + \frac{\mu}{2}\|\boldsymbol{s}_i\|_2^2\right) \quad \text{s.t.} \quad \boldsymbol{e}_i^T\boldsymbol{s}_i = 0, \quad i = 1,2,\ldots,N \tag{7}$$

where $\boldsymbol{e}_i$ is a zero vector with the $i$th element set to 1, and $\mu > 0$ is the ridge regression coefficient.

After computing the connection coefficients between samples, the objective function Eq. (5) can be transformed into a generalized eigenvalue problem. The projection matrix $\boldsymbol{A}$ is then optimized iteratively.

## 3. The proposed methods

To further enhance the performance of GNPE and improve the discriminability of the projection matrix, this paper introduces label information of the data and proposes GDNPE and GSNPE.

### 3.1. GDNPE

For a point $\boldsymbol{X}$ on a high-dimensional Grassmann manifold, we aim to learn a mapping matrix $\boldsymbol{A}$ that projects it to a point $\boldsymbol{Y}$ on a low-dimensional Grassmann manifold. This mapping relationship follows the general form of a linear mapping:

$$\boldsymbol{Y}_i = \boldsymbol{A}^T \boldsymbol{X}_i \tag{8}$$

It is evident that we cannot guarantee that the mapping matrix $\boldsymbol{A}$ will always be column-orthogonal during the iteration process, thus failing to ensure $\boldsymbol{A}^T\boldsymbol{X}$ remains on the Grassmann manifold. To ensure that $\boldsymbol{A}^T\boldsymbol{X}$ lies on the Grassmann manifold, a QR decomposition is applied:

$$\boldsymbol{Y}_i = \boldsymbol{A}^T \boldsymbol{X}_i = \boldsymbol{Q}_i \boldsymbol{R}_i \tag{9}$$

where $\boldsymbol{Q}_i$ is a column-orthogonal tall-and-thin matrix, representing the projection of point $\boldsymbol{X}_i$ on the high-dimensional Grassmann manifold to the low-dimensional Grassmann manifold. Specifically,

$$\boldsymbol{Q}_i = \boldsymbol{A}^T \widetilde{\boldsymbol{X}}_i = \boldsymbol{A}^T \boldsymbol{X}_i \boldsymbol{R}_i^{-1} \tag{10}$$

The objective function of GDNPE can be defined as:

$$\frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c}\left\|\Omega(\boldsymbol{Q}_i^c) - \sum_{j=1}^{N_c}\omega_{ij}^c\Omega(\boldsymbol{Q}_j^c)\right\|_F^2}{\sum_{i=1}^{C}\left\|\Omega(\boldsymbol{F}_i) - \sum_{j=1}^{C}b_{ij}\Omega(\boldsymbol{F}_j)\right\|_F^2} \tag{11}$$

where $C$ denotes the number of classes in the data, $N_c$ represents the number of samples in the $c$th class, $\boldsymbol{Q}_i^c$ and $\boldsymbol{Q}_j^c$ signify the embeddings of the $i$th and $j$th sample points from the $c$th class onto the low-dimensional Grassmann manifold, respectively. $\omega_{ij}^c$ is the intra-class connection weight between $\boldsymbol{Q}_i^c$ and $\boldsymbol{Q}_j^c$, while $\boldsymbol{F}_i$ and $\boldsymbol{F}_j$ represent the embeddings of the centroid points of the $i$th and $j$th classes onto the low-dimensional Grassmann manifold. $b_{ij}$ denotes the inter-class connection weight between the $i$th and $j$th classes.

In the GNPE framework, the connection weights between samples can be computed using either Eq. (6) or Eq. (7). The weight matrices derived from these equations are subsequently applied in GNPE-I and GNPE-II, respectively. Similarly, in the GDNPE framework, the weights $\omega_{ij}^c$ between sample $\boldsymbol{X}_i^c$ and sample $\boldsymbol{X}_j^c$ within $c$th class are calculated using Eq. (6) or Eq. (7), and these weights are then incorporated into GDNPE-I and GDNPE-II, respectively. Furthermore, in GDNPE, the centroids of samples in $c$th class can be determined by computing the extrinsic mean on the Grassmann manifold as specified in Eq. (4). The connection weights $b_{ij}$ between the centroids of $i$th class and $j$th class in GDNPE-I and GDNPE-II are subsequently calculated using Eqs. (6) and (7), respectively.

By combining Eqs. (10) and (11), the objective function of GDNPE can be consolidated:

$$\min_{\boldsymbol{A}} \frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c}\left\|\Omega\left(\boldsymbol{A}^T\widetilde{\boldsymbol{X}}_i^c\right) - \sum_{j=1}^{N_c}\omega_{ij}^c\Omega\left(\boldsymbol{A}^T\widetilde{\boldsymbol{X}}_j^c\right)\right\|_F^2}{\sum_{i=1}^{C}\left\|\Omega\left(\boldsymbol{A}^T\boldsymbol{Z}_i\right) - \sum_{j=1}^{C}b_{ij}\Omega\left(\boldsymbol{A}^T\boldsymbol{Z}_j\right)\right\|_F^2} \tag{12}$$

The form of the objective function can be further simplified as:

$$\min_{\boldsymbol{A}} \frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c}\left\|\boldsymbol{A}^T\boldsymbol{G}_i\boldsymbol{A}\right\|_F^2}{\sum_{i=1}^{C}\left\|\boldsymbol{A}^T\boldsymbol{S}_i\boldsymbol{A}\right\|_F^2} \tag{13}$$

where $\boldsymbol{G}_i = \widetilde{\boldsymbol{X}}_i^c\widetilde{\boldsymbol{X}}_i^{cT} - \sum_{j=1}^{N_c}\omega_{ij}^c\widetilde{\boldsymbol{X}}_j^c\widetilde{\boldsymbol{X}}_j^{cT}$ and $\boldsymbol{S}_i = \boldsymbol{Z}_i\boldsymbol{Z}_i^T - \sum_{j=1}^{C}b_{ij}\boldsymbol{Z}_j\boldsymbol{Z}_j^T$.

The works in [24,25] offer iterative approaches to solving the objective function, which differs from optimizations on the Riemannian manifold to some extent. Nevertheless, this straightforward optimization form has achieved remarkable results. By expanding the Frobenius norm term, the objective function of GDNPE can be further elaborated as:

$$\mathbb{f}_{obj}(\boldsymbol{A}) = \frac{\sum_{c=1}^{C}\sum_{i=1}^{N_c} tr\left(\boldsymbol{A}^T\boldsymbol{G}_i\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{G}_i\boldsymbol{A}\right)}{\sum_{i=1}^{C} tr\left(\boldsymbol{A}^T\boldsymbol{S}_i\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{S}_i\boldsymbol{A}\right)} \tag{14}$$

Lastly, the iterative objective function of GDNPE can be described as:

$$\mathbb{f}_{obj}^t(\boldsymbol{A}) = \frac{tr\left(\boldsymbol{A}^T\left(\sum_{c=1}^{C}\sum_{i=1}^{N_c}\boldsymbol{G}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{G}_i\right)\boldsymbol{A}\right)}{tr\left(\boldsymbol{A}^T\left(\sum_{i=1}^{C}\boldsymbol{S}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{S}_i\right)\boldsymbol{A}\right)} \tag{15}$$

where $\boldsymbol{A}$ denotes the mapping matrix obtained through optimization in the current epoch, while $\boldsymbol{A}^{(t-1)}$ represents the mapping matrix resulting from the optimization in the preceding epoch. $\boldsymbol{G}_i$ is calculated from the normalized high-dimensional Grassmann points, and $\boldsymbol{S}_i$ is derived from the mean values of the high-dimensional Grassmann points.

Given $\boldsymbol{J} = \sum_{c=1}^{C}\sum_{i=1}^{N_c}\boldsymbol{G}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{G}_i$ and $\boldsymbol{H} = \sum_{c=1}^{C}\boldsymbol{S}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{S}_i$, the objective function of GDNPE can be simplified as:

$$\min_{\boldsymbol{A}} \frac{tr(\boldsymbol{A}^T\boldsymbol{J}\boldsymbol{A})}{tr(\boldsymbol{A}^T\boldsymbol{H}\boldsymbol{A})} \tag{16}$$

In Eq. (16), $\boldsymbol{J}$ and $\boldsymbol{H}$ are real symmetric positive semi-definite matrices, and the mapping matrix $\boldsymbol{A}$ in each iteration can be obtained from the smallest eigenvalue of the generalized eigenvalue problem. The dimensionality reduction process of GDNPE can be summarized in Algorithm 1.

**Algorithm 1** Discriminant Neighborhood Preserving Embedding on Grassmann Manifold

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

1: Initialize the mapping matrix $\boldsymbol{A}^{(0)} = \begin{bmatrix}\boldsymbol{I}_d \\ \boldsymbol{R}\end{bmatrix}$, where $\boldsymbol{I}_d$ is a $d$ order identity matrix, and $\boldsymbol{R}$ is randomly initialized;
2: Compute the extrinsic mean $\{\boldsymbol{Z}_c\}_{c=1}^C$ for each class using Eq. (4);
3: Compute the intra-class weight matrix $\boldsymbol{W}$ and the inter-class weight matrix $\boldsymbol{B}$ from the samples $\{\boldsymbol{X}_i\}_{i=1}^N$ and centroids $\{\boldsymbol{Z}_c\}_{c=1}^C$ using Eq. (6) (or Eq. (7)) for GDNPE-I (or GDNPE-II);
4: **while** not converged and not max epoch **do**:
5: Normalize $\widetilde{\boldsymbol{X}}_i^t = \boldsymbol{X}_i^{t-1}\boldsymbol{R}_i^{-1}$ by $\boldsymbol{Q}_i\boldsymbol{R}_i = \boldsymbol{A}^T\boldsymbol{X}_i^{t-1}$;
6: Compute $\boldsymbol{J}$ and $\boldsymbol{H}$;
7: Solve the generalized eigenvalue problem associated with $\boldsymbol{A}$.
8: **end while**

### 3.2. GSNPE

The fundamental concept of the GSNPE algorithm lies in incorporating the objective function of GNPE as a regularization term within the objective function of GDNPE, thereby constructing the objective function of GSNPE. By integrating the objective function of GNPE in Eq. (5) with the objective function of GDNPE in Eq. (11), the objective function of GSNPE can be defined as:

$$\mathbb{O}_{GSNPE} = \alpha\cdot\mathbb{O}_{GDNPE} \oplus (1-\alpha)\cdot\mathbb{O}_{GNPE} \tag{17}$$

where $\oplus$ denotes the integration of the objective functions of GDNPE and GNPE, representing a generalized form of addition. The specific form of $\mathbb{O}_{GSNPE}$ can be described as:

$$\min_{\boldsymbol{A}} \frac{\alpha\cdot\sum_{c=1}^{C}\sum_{i=1}^{N_c}\left\|\Omega\left(\boldsymbol{Q}_i^c\right) - \sum_{j=1}^{N_c}\omega_{ij}^c\Omega\left(\boldsymbol{Q}_j^c\right)\right\|_F^2 + (1-\alpha)\cdot\sum_{i=1}^{N}\left\|\Omega\left(\boldsymbol{Q}_i\right) - \sum_{j=1}^{N}\lambda_{ij}\Omega\left(\boldsymbol{Q}_j\right)\right\|_F^2}{\sum_{i=1}^{C}\left\|\Omega\left(\boldsymbol{F}_i\right) - \sum_{j=1}^{C}b_{ij}\Omega\left(\boldsymbol{F}_j\right)\right\|_F^2} \tag{18}$$

where, $N$ represents the number of all samples, $\lambda_{ij}$ denotes the connection weights of the $k$-nearest neighbor graph based on all samples, and $\alpha$ signifies the regularization coefficient. The computation method for $\lambda_{ij}$ is consistent with that of $s_{ij}$ in GNPE. In GSNPE-I and GSNPE-II, the

connection weights $\lambda_{ij}$ between samples are calculated using Eqs. (6) and (7), respectively.

Assuming the sample set comprises $M$ 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 intra-class weight matrix $\boldsymbol{W}_{N\times N}$ can be normalized to the following form, where $M=\sum_{c=1}^{C} N_c$:

$$\boldsymbol{W}=\begin{bmatrix} \boldsymbol{W}^1 & & & & \\ & \boldsymbol{W}^2 & & & \\ & & \ddots & & \\ & & & \boldsymbol{W}^C & \\ & & & & \boldsymbol{0} \end{bmatrix} \tag{19}$$

In Eq. (19), $\boldsymbol{W}^i, i=1,2,\ldots,C$ is a symmetric matrix of order $N_i$, representing the intra-class weight matrix for the $i$th class. $\boldsymbol{0}$ is a symmetric matrix of order $N-M$, denoting the weight matrix for the unlabeled samples. Then, the objective function of GSNPE can be rewritten as:

$$\min_{\boldsymbol{A}} \frac{\alpha\cdot\sum_{i=1}^{N}\left\|\Omega(\boldsymbol{Q}_i)-\sum_{j=1}^{N}\omega_{ij}\Omega(\boldsymbol{Q}_j)\right\|_F^2+(1-\alpha)\cdot\sum_{i=1}^{N}\left\|\Omega(\boldsymbol{Q}_i)-\sum_{j=1}^{N}\lambda_{ij}\Omega(\boldsymbol{Q}_j)\right\|_F^2}{\sum_{i=1}^{C}\left\|\Omega(\boldsymbol{F}_i)-\sum_{j=1}^{C}b_{ij}\Omega(\boldsymbol{F}_j)\right\|_F^2} \tag{20}$$

Analogous to the derivation of GDNPE, the form of $\boldsymbol{G}_i$ is redefined as $\boldsymbol{G}_i=\widetilde{\boldsymbol{X}}_i\widetilde{\boldsymbol{X}}_i^T-\sum_{j=1}^{N}\omega_{ij}\widetilde{\boldsymbol{X}}_j\widetilde{\boldsymbol{X}}_j^T$, and let $\boldsymbol{U}_i=\widetilde{\boldsymbol{X}}_i\widetilde{\boldsymbol{X}}_i^T-\sum_{j=1}^{N}\lambda_{ij}\widetilde{\boldsymbol{X}}_j\widetilde{\boldsymbol{X}}_j^T$. Thus, the iterative objective function of GSNPE is presented as:

$$\mathrm{g}_{obj}^{t}(\boldsymbol{A})=\frac{tr\left(\boldsymbol{A}^T\left(\alpha\cdot\sum_{i=1}^{N}\boldsymbol{G}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{G}_i+(1-\alpha)\cdot\boldsymbol{U}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{U}_i\right)\boldsymbol{A}\right)}{tr\left(\boldsymbol{A}^T\left(\sum_{i=1}^{C}\boldsymbol{S}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{S}_i\right)\boldsymbol{A}\right)} \tag{21}$$

By letting $\boldsymbol{J}=\sum_{i=1}^{N}\alpha\cdot\boldsymbol{G}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{G}_i+(1-\alpha)\cdot\boldsymbol{U}_i\boldsymbol{A}^{(t-1)}\boldsymbol{A}^{(t-1)T}\boldsymbol{U}_i$, the objective function of GSNPE can be rearranged into the form of Eq. (16). Consequently, the optimization process of GSNPE is generally similar to that of GDNPE, with the main distinction lying in the specific calculations performed during optimization. The optimization process of GSNPE can be summarized as Algorithm 2:

**Algorithm 2** Semi-Supervised Neighborhood Preserving Embedding on Grassmann Manifold

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

**Ensure:** Optimal mapping matrix $\boldsymbol{A}\in\mathbb{R}^{D\times d}$.

1: Initialize the mapping matrix $\boldsymbol{A}^{(0)}=\begin{bmatrix}\boldsymbol{I}_d\\ \boldsymbol{R}\end{bmatrix}$, where $\boldsymbol{I}_d$ is a $d$ order identity matrix, and $\boldsymbol{R}$ is randomly initialized;
2: Compute the extrinsic mean $\{\boldsymbol{Z}_c\}_{c=1}^{C}$ of each class from labeled data $\{\boldsymbol{X}_i\}_{i=1}^{M}$ using Eq. (4);
3: Compute the intra-class weight matrix $\boldsymbol{W}$ and inter-class weight matrix $\boldsymbol{B}$ from the labeled data $\{\boldsymbol{X}_i\}_{i=1}^{M}$ and centroids $\{\boldsymbol{Z}_c\}_{c=1}^{C}$ using Eq. (6) (or Eq. (7)) for GSNPE-I (or GSNPE-II);
4: Compute the weight matrix $\Lambda$ based on the $k$-nearest neighbor graph of data $\{\boldsymbol{X}_i\}_{i=1}^{N}$ using Eq. (6) (or Eq. (7)) for GSNPE-I (or GSNPE-II);
5: **while** not converged and not max epoch **do**:
6: Normalize $\widetilde{\boldsymbol{X}}_i^t=\boldsymbol{X}_i^{t-1}\boldsymbol{R}_i^{-1}$ by $\boldsymbol{Q}_i\boldsymbol{R}_i=\boldsymbol{A}^T\boldsymbol{X}_i^{t-1}$;
7: Compute $\boldsymbol{J}$ and $\boldsymbol{H}$;
8: Solve the generalized eigenvalue problem associated with $\boldsymbol{A}$.
9: **end while**

**Table 1**
Explanations of Basic Symbols.

| Symbols | Explanations |
|---|---|
| $\boldsymbol{X}_i\in\mathcal{G}(p,D)$ | Samples on the high-dimensional Grassmann manifold |
| $\boldsymbol{Q}_i\in\mathcal{G}(p,d)$ | Projections on the low-dimensional Grassmann manifold |
| $\boldsymbol{Z}_i\in\mathcal{G}(p,D)$ | Centroids of $i$th class samples on the high-dimensional Grassmann manifold |
| $\boldsymbol{F}_i\in\mathcal{G}(p,d)$ | Centroids of $i$th class samples on the low-dimensional Grassmann manifold |
| $\boldsymbol{A}$ | Mapping matrix |
| $\omega_{ij}$ | The intra-class connection weight between $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ of labeled samples in GDNPE and GSNPE |
| $b_{ij}$ | The inter-class connection weight between $\boldsymbol{Z}_i$ and $\boldsymbol{Z}_j$ in GDNPE and GSNPE |
| $\lambda_{ij}$ | The connection weight between $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ of unlabeled samples in GSNPE |
| $\alpha$ | Regularization coefficient |

This section introduces numerous symbols. To enhance the clarity of the paper, we have summarized some of the fundamental symbols in Table 1.

### 3.3. Relationship with the previous works

Existing research on Grassmann manifolds has addressed the label information and the local structure of the data. In this section, we discuss the distinctions between the proposed GDNPE and prior related work.

Projection Metric Learning on Grassmann Manifold (GPML) [35] is a supervised dimensionality reduction technique based on the "From manifold to manifold" concept. It serves as an extension of Linear Discriminant Analysis on the Grassmann manifold. GPML differs from the proposed GDNPE in three primary aspects: (1) Neighborhood Relationship: GPML does not utilize the neighborhood structure of the data, whereas GDNPE incorporates this essential information. (2) Objective Function: GDNPE optimizes the mapping matrix by minimizing the ratio of the within-class scatter matrix to the between-class scatter matrix, which more closely aligns with the discriminative learning paradigm and potentially yields better results. In contrast, GPML optimizes the mapping matrix by minimizing the difference between the within-class and between-class scatter matrices. (3) Optimization Strategy: GPML utilizes the Riemannian Conjugate Gradient (RCG) method to optimize the mapping matrix, while GDNPE leverages projection metrics for the same purpose.

The study in [36] proposes A-based Metric Learning for Subspace Representation (AMLS), which considers both local relationships among samples and class information to learn a metric. This learned metric is then applied to dimensionality reduction methods. There are notable differences between this study and the proposed GDNPE, primarily in the following two aspects: (1) Metric Learning and Dimensionality Reduction: AMLS trains a metric based on the class and local information of the sample set and performs dimensionality reduction within this learned space. In contrast, the philosophy and goals of GDNPE are more intuitive and direct. (2) Reconstruction of Projection Points: GDNPE reconstructs the projection points on the low-dimensional manifold based on the local structure of samples in the high-dimensional manifold. This approach is fundamentally different from that of AMLS, which does not explicitly focus on such reconstruction.

The study in [37] introduces Cascaded Feature Extraction Architecture on Grassmann Manifold (CasArct) and Graph Embedding Multi-Kernel Metric Learning (GEMKML), enabling feature extraction on the Grassmann manifold. CasArct learns multiple manifold features

on the high-dimensional Grassmann manifold, and these features are then embedded into the kernel space for classification within the kernel space using GEMKML. There are fundamental differences between the proposed GDNPE and GEMKML, primarily in the following three primary aspects: (1) Feature Embedding: GEMKML embeds the multiple manifold features extracted by CasArct into the kernel space, whereas GDNPE operates on the low-dimensional Grassmann manifold. (2) Kernel Metrics: GEMKML integrates multiple kernel metrics on the Grassmann manifold, while GDNPE utilizes a single projection metric. (3) Methodological Approach: GEMKML is essentially a deep learning-based method, contrasting with GDNPE.

## 4. Experiments

To demonstrate the advantages of GDNPE and GSNPE in image-set classification tasks, we compared them with a variety of algorithms, including Grassmann KNN (GKNN), Grassmann SVM (GSVM) [38], GKDA, GLPP, GNPE, GALL, GPML [35], GrNet [39], Nested Grassmann (NG) and Supervised Nested Grassmann (SNG) for dimensionality reduction [40], and Generalized Relevance Learning on Grassmann manifold (GRLGQ) [41].

### 4.1. Experimental setting

We evaluated the classification performance of the algorithms using 4 metrics: accuracy, precision, recall, and $F1$ score. The better an algorithm performs, the closer these metrics approach 1; poorer performance results in values closer to 0.

The target dimension $d$ of the Grassmann manifold samples is determined by the cumulative contribution rate of the eigenvectors. These eigenvectors are derived from the matrix $\sum_{i=1}^{N} \boldsymbol{X}\boldsymbol{X}^T$. For a given cumulative contribution rate $\eta$ $(0 < \eta < 1)$ and a maximum dimension $D$, $d$ can be defined as:

$$d = \arg\min_{d^*} \left\{ 0 < d^* < D \mid \sum_{i=1}^{d^*} \sigma_i \geq \eta \cdot \sum_{i=1}^{D} \sigma_i \right\} \tag{22}$$

where $\sigma_i$ represents the contribution rate of the $i$th eigenvector.

The code for algorithms such as GrNet,[1] GPML,[2] NG,[3] SNG, GRLGQ,[4] and GSVM[5] is publicly available, and we used their default parameter settings. GKDA is implemented by applying the Grassmann Projection Kernel to KDA[6] and employing a nearest neighbor classifier in Euclidean space. For methods like GLPP, GNPE, and GALL, for which the code is not available, we reproduced the algorithms based on the pseudocode provided in the respective papers.

GKNN, GSVM, and GRLGQ are three classifiers on the Grassmann manifold. They are directly trained and tested for performance using high-dimensional Grassmann samples. For unsupervised dimensionality reduction methods (e.g., GALL, GLPP, GNPE-I, and GNPE-II) and supervised methods (e.g., GKDA, GDNPE-I, and GDNPE-II), we train their mapping matrices using the training set on the high-dimensional manifold. For GSNPE-I and GSNPE-II, we divide the training samples on the high-dimensional Grassmann manifold into labeled and unlabeled subsets, and use these subsets to train the mapping matrices. For these methods, we train a GKNN classifier using the training samples on the low-dimensional manifold. During the testing phase, the trained mapping matrices reduce the dimensions of the test samples, and the trained classifier classifies them on the low-dimensional manifold.

Since NG and SNG lack an explicit mapping between the high-dimensional and low-dimensional Grassmann manifolds, we use NG and SNG to reduce the dimensions of all high-dimensional samples, respectively. The reduced samples are then divided into training and test sets on the low-dimensional manifold.

To ensure the reliability and reproducibility of our results, we conducted repeated experiments on various hardware platforms. We consistently arrived at similar conclusions.

### 4.2. Datasets

We conducted experiments on eight datasets: Ballet, CASIA-B, ETH-80, Extended Yale B (EYB), RGB-D, Traffic, UCF-Sport, and UT-Kinect. Example images from these datasets are shown in Fig. 2, and details are summarized in Table 2.

The Ballet [42] dataset consists of 44 videos collected from ballet instructional DVDs, featuring 8 complex ballet movements performed by 3 subjects. This dataset presents significant challenges for video classification tasks due to substantial intra-class variations in speed, spatial and temporal scales, attire, and movement patterns. Each video frame is resized to a $20 \times 20$ grayscale image. Since some videos contain 2−3 different action classes, all videos are modeled as 59 image-set samples. In this study, we randomly selected one sample from each category to serve as the test data, while the remaining samples were used as the training data.

The CASIA-B [43] dataset consists of gait data from 124 individuals (93 males and 31 females). In this dataset, gait sequences were captured from 11 angles for each subject under 3 walking conditions. Each subject has 6 motion sequences under the normal walking condition, and the video captured at 0° is used to model the image-set samples. Each video frame is resized to a $20 \times 20$ grayscale image. In this study, the subjects' motion data are used for gender classification. In this study, we randomly selected 10 samples from each gender to serve as the test data, while the remaining samples were used as the training data.

The ETH-80 [44] dataset contains 8 different object categories: cow, cup, horse, dog, potato, car, pear, and apple. Each category includes 10 distinct objects, each captured from 41 different viewpoints, forming an image-set sample for each object. In this study, 9 objects from each category are selected as training samples, while the remaining object are used as test samples.

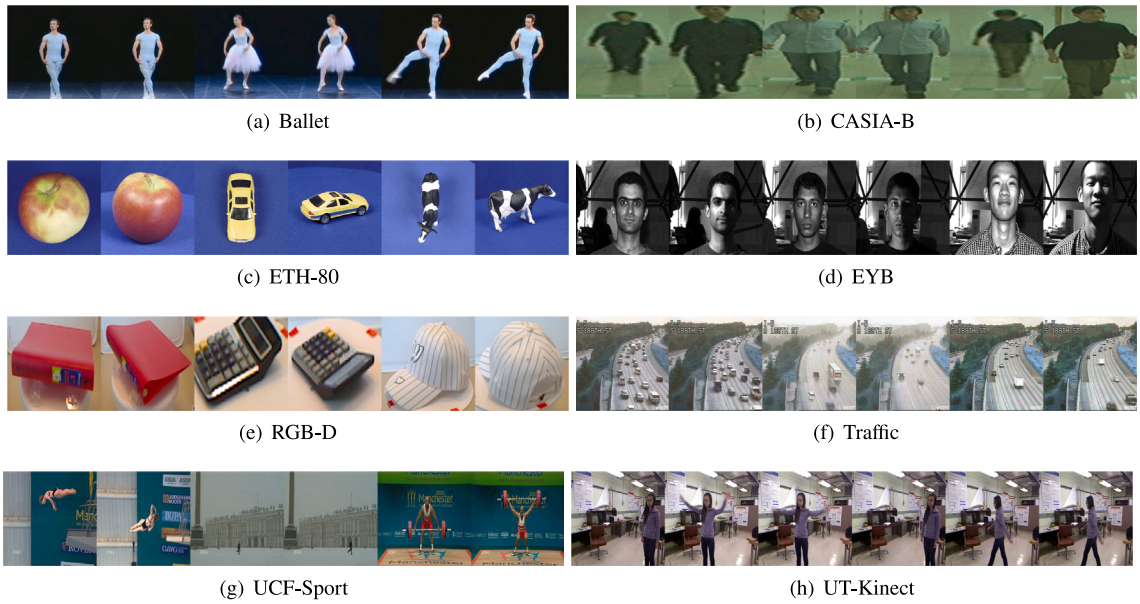
The EYB [45] dataset includes 16128 images of 28 subjects under 9 different poses and 64 lighting conditions, thus forming a large-scale facial dataset. Each subject has 64 images for each pose, are organized into image-set samples. In this study, samples from 8 poses for each subject are selected as training data, while samples from the remaining pose are used as test data.

The RGB-D [46] dataset contains RGB images of 300 objects across 51 categories captured under various conditions. Each object has hundreds of images, from which 40 images are randomly selected, converted to $20 \times 20$ grayscale, and organized into an image-set sample. In this study, 1 sample from each category are selected as test data, while the remaining samples are used as training data.

The Traffic [47] video database consists of 254 highway traffic sequences captured by a stationary traffic camera. This database includes a variety of traffic conditions (heavy, medium, light) and weather scenarios (overcast, sunny, rainy). Each video sequence is recorded at 10 frames per second with a resolution of $320 \times 240$ and lasts approximately 5 seconds. Each frame is converted to a $20 \times 20$ grayscale image, and each video is modeled as an image-set sample. In this study, 5 samples from each category are selected as test data, while the remaining samples are used as the training data.

The UCF-Sport [48] dataset is an action recognition dataset consisting of 150 video sequences across 13 action categories. In this dataset, each video contains 22 – 144 frames. Each frame is converted to a grayscale image and resized to $20 \times 20$ pixels. Each video is modeled as an image-set sample. In this study, we randomly selected one sample

[1] https://github.com/zhiwu-huang/GrNet
[2] https://github.com/GitWR/TDL
[3] https://github.com/cvgmi/NestedGrassmann
[4] https://github.com/mohammadimathstar/GRLGQ
[5] https://github.com/adamguos/arma-grassmann-classifier
[6] https://github.com/concavegit/kfda/



(a) Ballet (b) CASIA-B

(c) ETH-80 (d) EYB

(e) RGB-D (f) Traffic

(g) UCF-Sport (h) UT-Kinect

**Fig. 2.** Examples of some images used in the experiments.

**Table 2**
Summary of Datasets.

| Datasets | Samples | Train | Test | Classes | $\mathcal{G}(p, D)$ |
|---|---|---|---|---|---|
| Ballet | 59 | 51 | 8 | 8 | $\mathcal{G}(12, 400)$ |
| CASIA-B | 744 | 724 | 20 | 2 | $\mathcal{G}(10, 400)$ |
| ETH-80 | 80 | 72 | 8 | 8 | $\mathcal{G}(10, 400)$ |
| EYB | 252 | 224 | 28 | 28 | $\mathcal{G}(5, 400)$ |
| RGB-D | 300 | 249 | 51 | 51 | $\mathcal{G}(10, 400)$ |
| Traffic | 254 | 239 | 15 | 3 | $\mathcal{G}(15, 400)$ |
| UCF-Sport | 150 | 137 | 13 | 13 | $\mathcal{G}(15, 400)$ |
| UT-Kinect | 199 | 189 | 10 | 10 | $\mathcal{G}(10, 400)$ |

**Table 3**
Basic Parameter Values of GSNPE across Different Datasets.

| Datasets | $\mathcal{G}(p, d)$ | $\eta$ | $\alpha$ of GSNPE-I | $\alpha$ of GSNPE-II |
|---|---|---|---|---|
| Ballet | $\mathcal{G}(12, 131)$ | 0.95 | 0.90 | 0.95 |
| CASIA-B | $\mathcal{G}(10, 35)$ | 0.75 | 0.50 | 0.50 |
| ETH-80 | $\mathcal{G}(10, 106)$ | 0.90 | 0.70 | 0.70 |
| EYB | $\mathcal{G}(5, 71)$ | 0.95 | 0.95 | 0.95 |
| RGB-D | $\mathcal{G}(10, 263)$ | 0.95 | 0.50 | 0.50 |
| Traffic | $\mathcal{G}(15, 54)$ | 0.75 | 0.50 | 0.50 |
| UCF-Sport | $\mathcal{G}(15, 256)$ | 0.95 | 0.40 | 0.40 |
| UT-Kinect | $\mathcal{G}(10, 184)$ | 0.95 | 0.50 | 0.50 |

from each category to serve as the test data, while the remaining samples were used as the training data.

The UT-Kinect [49] dataset contains videos, depth sequences, and skeleton data for 10 action categories, performed by 10 subjects in front of a Kinect device. Each subject performs each action twice, resulting in a total of 200 action videos, with each video modeled as an image-set sample. In this study, we randomly selected one sample from each category to serve as the test data, while the remaining samples were used as the training data.

### 4.3. Classification performance of GDNPE and GSNPE

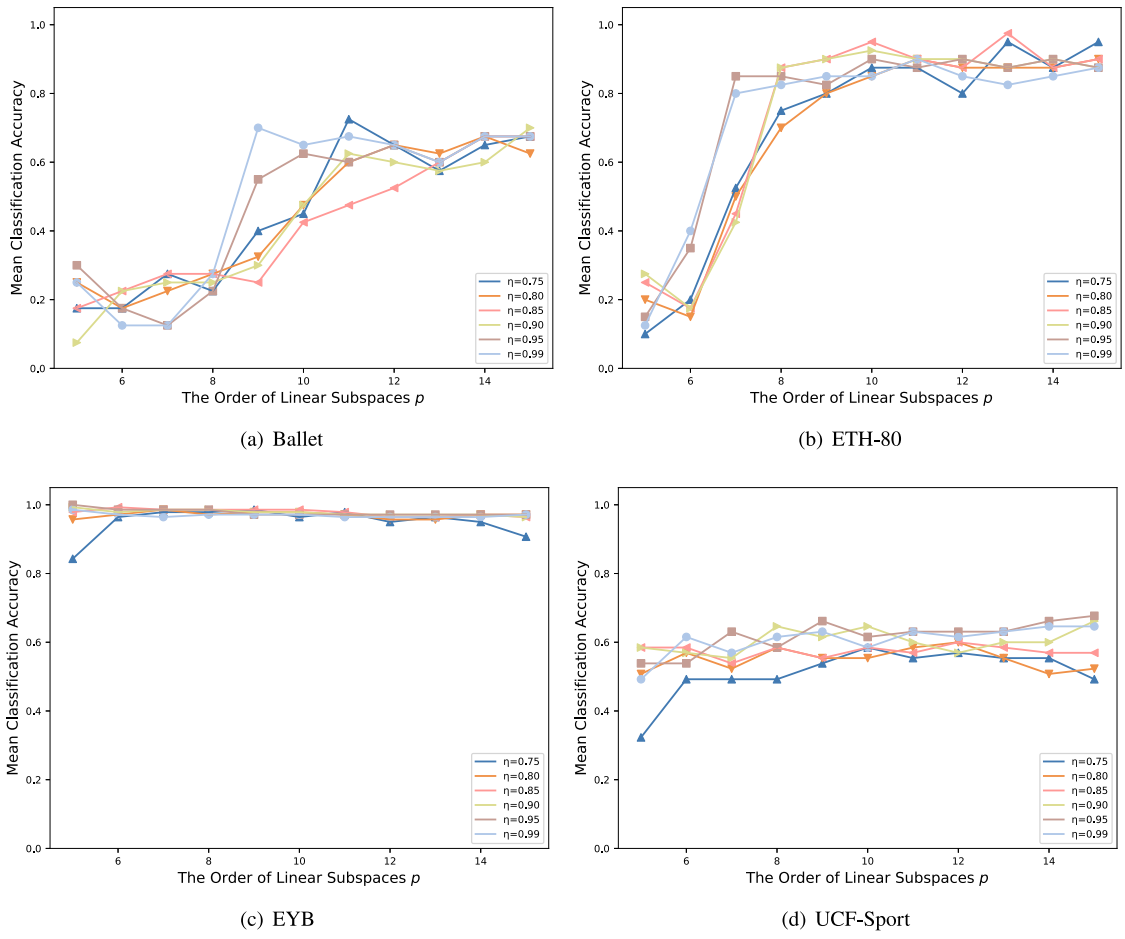
We compared the classification performance of the proposed GDNPE and GSNPE with other algorithms on the Grassmann manifold, such as GKNN, GSVM, GKDA, GLPP, GNPE, GALL, GPML, GrNet, NG, SNG, and GRLGQ. Tables 4–7 report the accuracy, precision, recall, and $F1$ scores of GDNPE-I, GDNPE-II, GSNPE-I, GSNPE-II, and all baseline methods across the eight datasets.

In our experiments, the specific parameter settings for GSNPE-I and GSNPE-II across different datasets are summarized in Table 3. The process of analyzing these parameters is detailed in Section 4.4.

The results indicate that across the majority of datasets, the performance of GDNPE-I and GDNPE-II consistently surpasses that of GNPE-I and GNPE-II respectively. This observation underscores the efficacy of integrating discriminative information into the dimensionality reduction process. GDNPE also surpasses GKDA in classification performance, indicating that it better preserves the manifold and geometric structure of the data, highlighting its advantage in dimensionality reduction and classification tasks. Moreover, GDNPE outperforms the Grassmann manifold classifiers such as GKNN, GSVM, and GRLGQ, validating its superior feature extraction capability. Compared to dimensionality reduction algorithms like GLPP, GALL, NG, and SNG, GDNPE also exhibits a clear advantage in both dimensionality reduction and classification tasks. GPML fails to converge on several datasets, including EYB, UCF-Sport, and UT-Kinect. In addition, GDNPE outperforms GPML on the majority of datasets, thereby demonstrating its superior performance. GSNPE outperforms GNPE on most datasets, further proving the importance of discriminative information in preserving local structure. GrNet is a classic deep learning method on the Grassmann manifold. In this study, GrNet was iteratively optimized for 50 epochs across all datasets.

**Table 4**
Comparison of the Average Classification Accuracy across Different Datasets.

| Methods | Ballet | CASIA-B | ETH-80 | EYB | RGB-D | Traffic | UCF-Sport | UT-Kinect |
|---|---|---|---|---|---|---|---|---|
| GKNN | 0.708 | 0.800 | 0.927 | 0.952 | 0.408 | 0.789 | 0.417 | 0.458 |
| GSVM[38] | 0.240 | 0.763 | 0.906 | 0.973 | 0.031 | 0.789 | 0.109 | 0.458 |
| GKDA[22] | 0.625 | 0.896 | 0.979 | 0.994 | 0.283 | 0.878 | 0.269 | 0.467 |
| GLPP[24] | 0.396 | 0.688 | 0.719 | 0.887 | 0.552 | 0.783 | 0.532 | 0.367 |
| GALL[26] | 0.771 | 0.683 | 0.823 | 0.726 | 0.493 | 0.767 | 0.481 | 0.358 |
| GNPE-I[25] | 0.542 | 0.788 | 0.823 | 0.943 | 0.605 | 0.844 | 0.513 | 0.500 |
| GNPE-II[25] | 0.375 | 0.783 | 0.781 | 0.926 | 0.649 | 0.811 | 0.500 | 0.467 |
| GPML[35] | 0.750 | 0.567 | 0.958 | – | 0.647 | 0.556 | – | – |
| GrNet[39] | 0.375 | 0.750 | 1.000 | 1.000 | 0.182 | 0.800 | 0.231 | 0.400 |
| NG[40] | 0.635 | 0.842 | 0.875 | 0.938 | 0.639 | 0.822 | 0.515 | 0.425 |
| SNG[40] | 0.719 | 0.725 | 0.917 | 0.955 | 0.605 | 0.922 | 0.538 | 0.420 |
| GRLGQ[41] | 0.594 | 0.600 | 0.948 | 0.964 | 0.461 | 0.711 | 0.455 | 0.508 |
| GSNPE-I | **0.563** | **0.804** | **0.927** | **0.985** | **0.717** | **0.767** | **0.532** | **0.483** |
| GSNPE-II | **0.521** | **0.804** | **0.938** | **0.979** | **0.729** | **0.817** | **0.603** | **0.442** |
| GDNPE-I | **0.635** | **0.825** | **0.979** | **0.991** | **0.685** | **0.817** | **0.628** | **0.475** |
| GDNPE-II | **0.615** | **0.771** | **0.948** | **0.970** | **0.672** | **0.883** | **0.551** | **0.517** |




(a) Ballet (b) ETH-80

(c) EYB (d) UCF-Sport

**Fig. 3.** Mean classification accuracy of GDNPE across different subspace orders $p(p \in [5, 15])$ and cumulative contribution rates $\eta(\eta \in [0.75, 0.99])$.

**Table 5**
Comparison of the Average Classification Precision across Different Datasets.

| Methods | Ballet | CASIA-B | ETH-80 | EYB | RGB-D | Traffic | UCF-Sport | UT-Kinect |
|---|---|---|---|---|---|---|---|---|
| GKNN | 0.599 | 0.829 | 0.896 | 0.932 | 0.319 | 0.809 | 0.321 | 0.344 |
| GSVM[38] | 0.123 | 0.837 | 0.859 | 0.960 | 0.006 | 0.811 | 0.038 | 0.345 |
| GKDA[22] | 0.506 | 0.916 | 0.969 | 0.991 | 0.204 | 0.892 | 0.176 | 0.380 |
| GLPP[24] | 0.275 | 0.729 | 0.594 | 0.843 | 0.464 | 0.793 | 0.432 | 0.290 |
| GALL[26] | 0.701 | 0.714 | 0.767 | 0.648 | 0.397 | 0.785 | 0.387 | 0.255 |
| GNPE-I[25] | 0.432 | 0.814 | 0.741 | 0.920 | 0.509 | 0.858 | 0.418 | 0.404 |
| GNPE-II[25] | 0.264 | 0.798 | 0.702 | 0.892 | 0.545 | 0.825 | 0.403 | 0.368 |
| GPML[35] | 0.646 | 0.567 | 0.938 | – | 0.562 | 0.633 | – | – |
| GrNet[39] | 0.213 | 0.833 | 1.000 | 1.000 | 0.115 | 0.822 | 0.103 | 0.350 |
| NG[40] | 0.513 | 0.862 | 0.833 | 0.911 | 0.545 | 0.831 | 0.404 | 0.334 |
| SNG[40] | 0.613 | 0.746 | 0.875 | 0.936 | 0.515 | 0.930 | 0.420 | 0.333 |
| GRLGQ[41] | 0.456 | 0.737 | 0.922 | 0.946 | 0.359 | 0.747 | 0.354 | 0.410 |
| GSNPE-I | **0.436** | **0.823** | **0.891** | **0.978** | **0.638** | **0.774** | **0.459** | **0.403** |
| GSNPE-II | **0.415** | **0.829** | **0.906** | **0.969** | **0.656** | **0.829** | **0.517** | **0.375** |
| GDNPE-I | **0.528** | **0.849** | **0.969** | **0.987** | **0.600** | **0.841** | **0.545** | **0.386** |
| GDNPE-II | **0.507** | **0.802** | **0.922** | **0.957** | **0.575** | **0.893** | **0.446** | **0.422** |

**Table 6**
Comparison of the Average Classification Recall across Different Datasets.

| Methods | Ballet | CASIA-B | ETH-80 | EYB | RGB-D | Traffic | UCF-Sport | UT-Kinect |
|---|---|---|---|---|---|---|---|---|
| GKNN | 0.708 | 0.800 | 0.927 | 0.952 | 0.408 | 0.789 | 0.417 | 0.458 |
| GSVM[38] | 0.240 | 0.763 | 0.906 | 0.973 | 0.031 | 0.789 | 0.109 | 0.458 |
| GKDA[22] | 0.625 | 0.896 | 0.979 | 0.994 | 0.283 | 0.878 | 0.269 | 0.467 |
| GLPP[24] | 0.396 | 0.688 | 0.719 | 0.887 | 0.552 | 0.783 | 0.532 | 0.367 |
| GALL[26] | 0.771 | 0.683 | 0.823 | 0.726 | 0.493 | 0.767 | 0.481 | 0.358 |
| GNPE-I[25] | 0.542 | 0.788 | 0.823 | 0.943 | 0.605 | 0.844 | 0.513 | 0.500 |
| GNPE-II[25] | 0.375 | 0.783 | 0.781 | 0.926 | 0.649 | 0.811 | 0.500 | 0.467 |
| GPML[35] | 0.750 | 0.567 | 0.958 | – | 0.647 | 0.556 | – | – |
| GrNet[39] | 0.375 | 0.750 | 1.000 | 1.000 | 0.154 | 0.800 | 0.231 | 0.400 |
| NG[40] | 0.635 | 0.842 | 0.875 | 0.938 | 0.639 | 0.822 | 0.515 | 0.425 |
| SNG[40] | 0.719 | 0.725 | 0.917 | 0.955 | 0.605 | 0.922 | 0.538 | 0.420 |
| GRLGQ[41] | 0.594 | 0.600 | 0.948 | 0.964 | 0.461 | 0.711 | 0.455 | 0.508 |
| GSNPE-I | **0.563** | **0.804** | **0.927** | **0.985** | **0.717** | **0.767** | **0.532** | **0.483** |
| GSNPE-II | **0.521** | **0.804** | **0.938** | **0.979** | **0.729** | **0.817** | **0.603** | **0.442** |
| GDNPE-I | **0.635** | **0.825** | **0.979** | **0.991** | **0.685** | **0.817** | **0.628** | **0.475** |
| GDNPE-II | **0.615** | **0.771** | **0.948** | **0.970** | **0.672** | **0.883** | **0.551** | **0.517** |

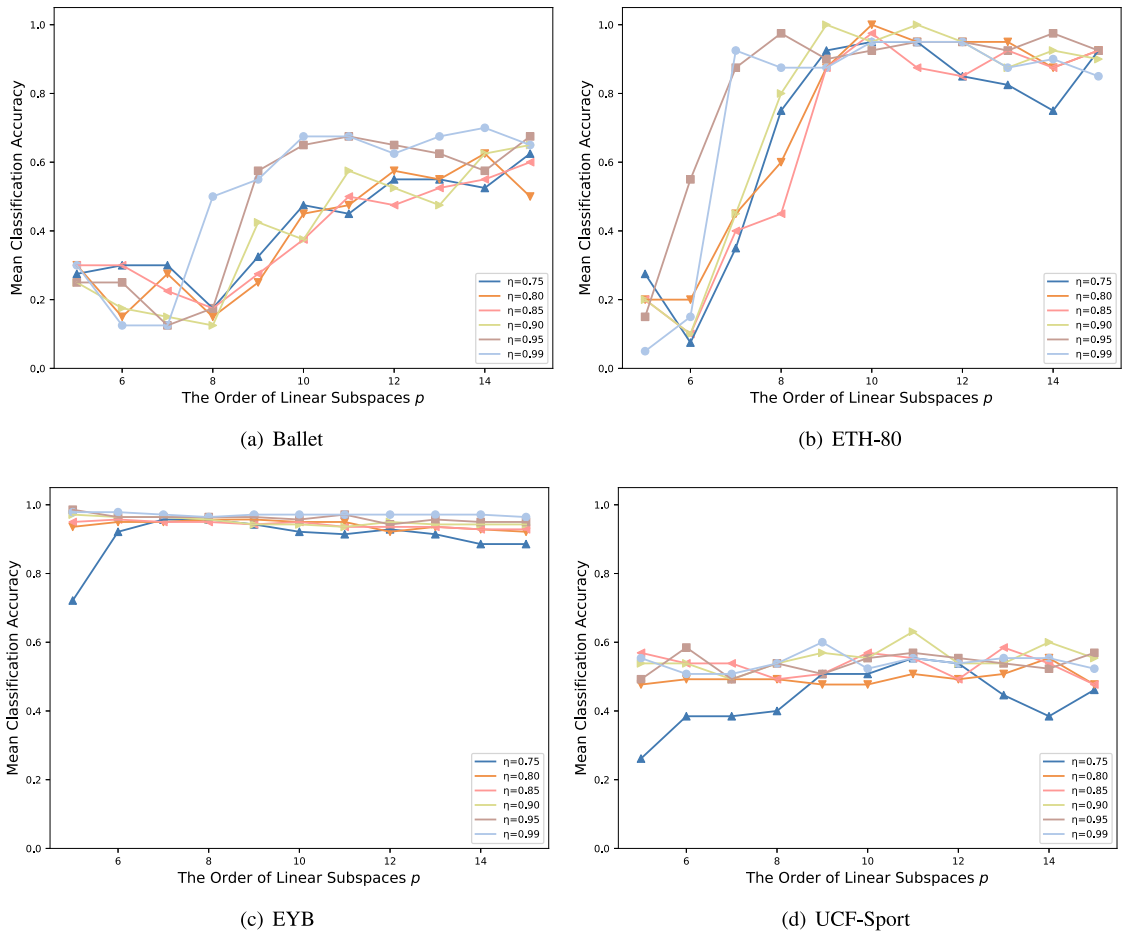
Under these conditions, GDNPE demonstrated superior classification performance compared to GrNet.

### 4.4. Parameters analysis

The GDNPE model has 2 key parameters: the subspace order $p$ of the Grassmann manifold and the target dimension $d$ of the low-dimensional Grassmann manifold, which are determined by the cumulative contribution rate $\eta$. To select the optimal $p$ and $\eta$, we perform classification experiments on samples from subspaces of different orders. In addition, the regularization coefficient $\alpha$ and the number of neighbors $k$ are also critical for GSNPE. Once $p$ and $\eta$ are determined, the optimal values of $\alpha$ and $k$ can be found using a search method.

In this section, to avoid redundant computation, we determine the hyperparameters of the algorithms on the Grassmann manifold using GNPE-I, GDNPE-I, and GSNPE-I as representatives. Since parameter optimization requires multiple repetitions of algorithms such as GNPE, GDNPE, and GSNPE, we conduct parameter experiments on four relatively small datasets: Ballet, ETH-80, EYB, and UCF-Sport to save computational time.

Based on our previous experience with dimensionality reduction algorithms on the Grassmann manifold, we perform a grid search over




(a) Ballet (b) ETH-80

(c) EYB (d) UCF-Sport

**Fig. 4.** Mean classification accuracy of GSNPE across different subspace orders $p(p \in [5, 15])$ and cumulative contribution rates $\eta(\eta \in [0.75, 0.99])$.

**Table 7**
Comparison of the Average Classification $F1$ Score across Different Datasets.

| Methods | Ballet | CASIA-B | ETH-80 | EYB | RGB-D | Traffic | UCF-Sport | UT-Kinect |
|---|---|---|---|---|---|---|---|---|
| GKNN | 0.634 | 0.795 | 0.906 | 0.938 | 0.346 | 0.783 | 0.350 | 0.381 |
| GSVM[38] | 0.141 | 0.746 | 0.875 | 0.964 | 0.008 | 0.776 | 0.043 | 0.381 |
| GKDA[22] | 0.538 | 0.894 | 0.972 | 0.992 | 0.216 | 0.876 | 0.191 | 0.408 |
| GLPP[24] | 0.307 | 0.673 | 0.634 | 0.857 | 0.490 | 0.774 | 0.460 | 0.315 |
| GALL[26] | 0.724 | 0.670 | 0.785 | 0.673 | 0.427 | 0.764 | 0.413 | 0.287 |
| GNPE-I[25] | 0.460 | 0.779 | 0.767 | 0.927 | 0.538 | 0.839 | 0.446 | 0.436 |
| GNPE-II[25] | 0.287 | 0.779 | 0.726 | 0.903 | 0.578 | 0.806 | 0.431 | 0.400 |
| GPML[35] | 0.681 | 0.534 | 0.944 | – | 0.587 | 0.513 | – | – |
| GrNet[39] | 0.250 | 0.733 | 1.000 | 1.000 | 0.128 | 0.805 | 0.122 | 0.367 |
| NG[40] | 0.551 | 0.838 | 0.847 | 0.920 | 0.574 | 0.814 | 0.439 | 0.363 |
| SNG[40] | 0.646 | 0.714 | 0.889 | 0.942 | 0.541 | 0.922 | 0.454 | 0.360 |
| GRLGQ[41] | 0.497 | 0.516 | 0.931 | 0.952 | 0.389 | 0.679 | 0.381 | 0.441 |
| GSNPE-I | **0.473** | **0.801** | **0.903** | **0.980** | **0.662** | **0.757** | **0.482** | **0.429** |
| GSNPE-II | **0.446** | **0.798** | **0.917** | **0.972** | **0.678** | **0.815** | **0.543** | **0.397** |
| GDNPE-I | **0.559** | **0.821** | **0.972** | **0.988** | **0.626** | **0.806** | **0.571** | **0.415** |
| GDNPE-II | **0.540** | **0.760** | **0.931** | **0.961** | **0.605** | **0.883** | **0.477** | **0.453** |

the subspace order $p$ and cumulative contribution rate $\eta$ for GDNPE and GSNPE within the ranges of $p \in [5, 15]$ and $\eta \in [0.75, 0.99]$. The results are shown in Figs. 3 and 4. As illustrated, the effective ranges for $p$ and $\eta$ vary across different datasets.

For the Ballet and ETH-80 datasets, both GDNPE and GSNPE achieve satisfactory classification results when the subspace order $p$ is larger. However, on the EYB dataset, both GDNPE and GSNPE require a lower value of the subspace order to achieve optimal performance. The UCF-Sport dataset is insensitivity to the subspace order. The specific values of subspace order $p$, cumulative contribution rate $\eta$, and target dimension $d$ for each dataset are presented in Table 3.

After selecting the subspace order $p$ and cumulative contribution rate $\eta$ on the Grassmann manifold, we determine the optimal regularization coefficient $\alpha$ within the range $\alpha \in [0.05, 0.99]$ for both GSNPE-I and GSNPE-II. As shown in Fig. 5, the regularization coefficient has a similar impact on GSNPE-I and GSNPE-II. For the ETH-80 dataset, GSNPE exhibits poor classification performance when the regularization coefficient $\alpha$ is too large. In contrast, for the EYB dataset, GSNPE typically requires a considerably larger regularization coefficient. On the Ballet and UCF-Sport datasets, the impact of the $\alpha$ value on GSNPE's performance is negligible. Both label information and local structure information play crucial roles in feature extraction and classification tasks.

We searched for the optimal number of neighbors $k$ for GSNPE within a given range $k \in [5, 15]$, with the results shown in Fig. 6. We found that the impact of $k$ on dimensionality reduction performance is minimal. Therefore, in subsequent experiments, the number of neighbors $k$ for both GSNPE-I and GSNPE-II is set to 5.

### 4.5. Discussion

We visualize the distributions of the CASIA-B, ETH-80, Traffic, and UT-Kinect datasets using the projection metrics on Grassmann manifolds and Uniform Manifold Approximation and Projection (UMAP) [50]. The results are documented in Fig. 7. In this section, we further discuss the reasons why GDNPE and GSNPE perform poorly with the CASIA-B, Traffic, and UT-Kinect datasets.

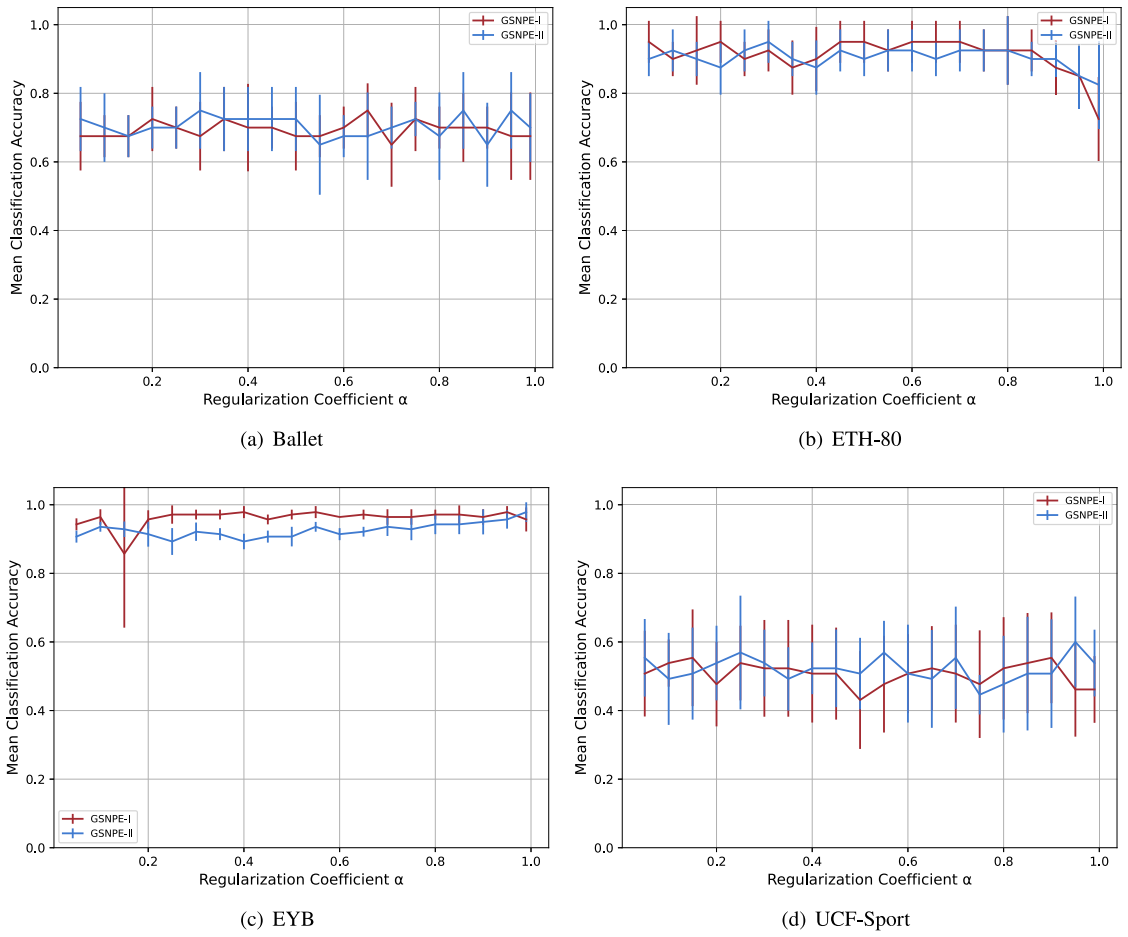



**Fig. 5.** Mean classification accuracy and standard deviation of GSNPE under different regularization coefficients $\alpha(\aleph \in [0.05, 0.99])$.

On one hand, both the CASIA-B and Traffic datasets are imbalanced across classes. For example, in the CASIA-B dataset, the number of male samples is 3 times that of female samples, and in the Traffic dataset, the number of heavy samples is 3.67 times that of light samples. In GDNPE and GSNPE, it is necessary to construct within-class scatter matrices to reflect the distribution of samples within a cluster. For clusters with a small number of samples, the constructed within-class scatter matrices may not accurately capture the distribution of the samples. This is one reason why GDNPE and GSNPE perform poorly on these datasets.

On the other hand, the distributions of samples in the CASIA-B, Traffic, and UT-Kinect datasets are disordered, which is significantly different from that of the ETH-80 dataset. Methods such as GDNPE and GSNPE largely depend on the distribution of data. This leads to their better performance on datasets with clear boundaries such as ETH-80 and poorer performance on datasets with disordered distributions such as CASIA-B.

## 5. Conclusion

This paper introduces the Discriminant Neighborhood Preserving Embedding on Grassmann manifold. This innovative approach adeptly maintains the local geometric structure of the data while optimally harnessing discriminative information, thereby exhibiting superior performance in feature extraction and classification tasks related to image-set data.

Furthermore, we present the GSNPE method, which constitutes an initial step in applying semi-supervised dimensionality reduction on the Grassmann manifold. This method provides valuable insights and serves as a reference point for researchers in related fields, fostering further exploration and advancements in this rapidly growing field.

Although our work has achieved promising results, there are still aspects worthy of further exploration. Our future research will focus on enhancing the performance of dimensionality reduction techniques by integrating clustering algorithms on the Grassmann manifold using a self-supervised approach.

## CRediT authorship contribution statement

**Benchao Li:** Writing – review & editing, Writing – original draft, Software. **Yuanyuan Zheng:** Visualization. **Ruisheng Ran:** Supervision, Methodology, Conceptualization. **Bin Fang:** Supervision.

## Declaration of competing 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.

## Acknowledgments

This work was supported by Chongqing Municipal Education Commission under grant KJZD-K202100505, Chongqing Science and Technology Bureau under grant cstc2020jscx-msxmX0190, the Humanities and Social Science Fund of Ministry of Education of the People's Republic of China under grant 20YJAZH084.

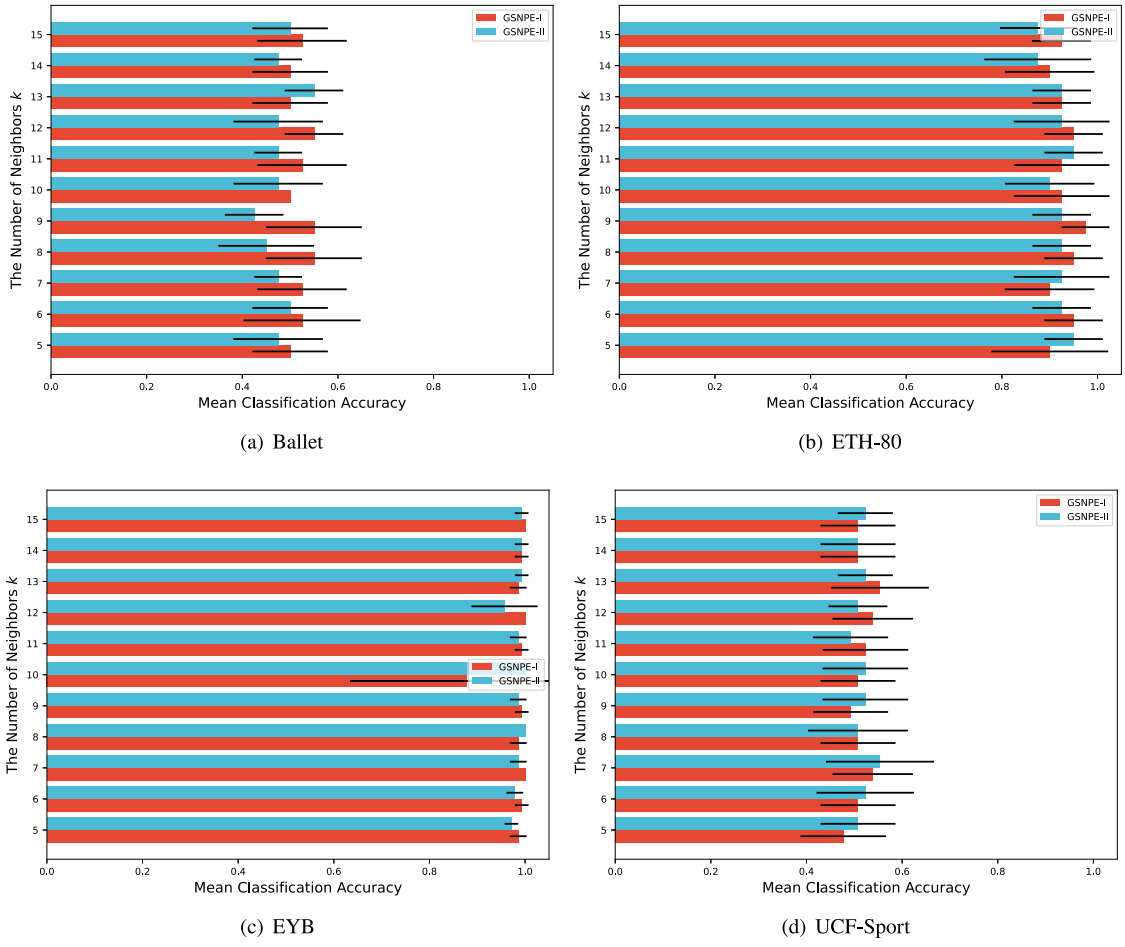
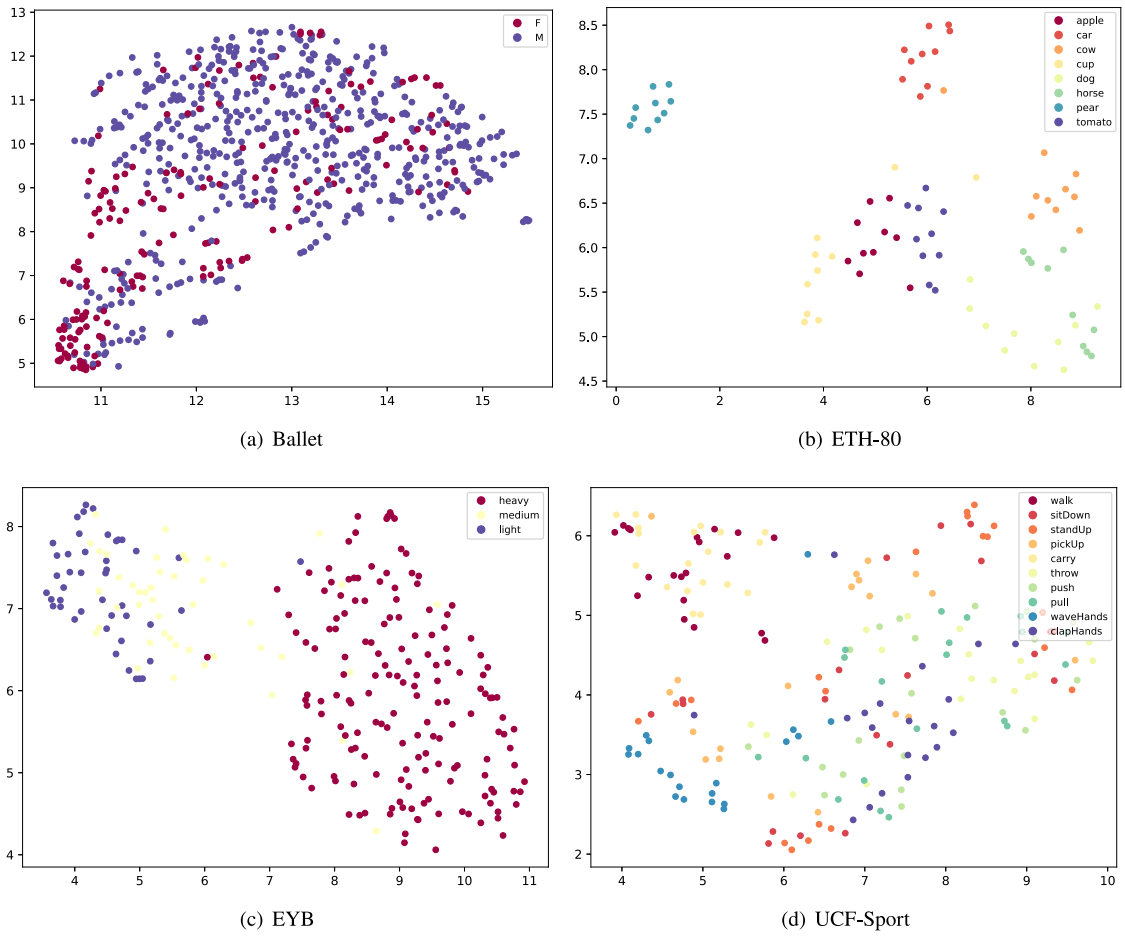



**Fig. 6.** Mean classification accuracy and standard deviation of GSNPE under different number of neighbors $k(k \in [5, 15])$.




(a) Ballet (b) ETH-80

(c) EYB (d) UCF-Sport

**Fig. 7.** Visualization of the distributions of CASIA-B, ETH-80, Traffic, and UT-Kinect datasets using the UMAP toolbox.

## Data availability

Data will be made available on request.

## References


[1] A. Edelman, T.A. Arias, S.T. Smith, The geometry of algorithms with orthogonality constraints, SIAM J. Matrix Anal. Appl. 20 (2) (1998) 303–353.

[2] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, M. Harandi, Kernel methods on the Riemannian manifold of symmetric positive definite matrices, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 73–80.

[3] K. Fukui, O. Yamaguchi, Face recognition using multi-viewpoint patterns for robot vision, in: Robotics Research. the Eleventh International Symposium: With 303 Figures, Springer, 2005, pp. 192–201.

[4] P. Jing, Y. Su, Z. Li, J. Liu, L. Nie, Low-rank regularized tensor discriminant representation for image set classification, Signal Process. 156 (2019) 62–70.

[5] T.-K. Kim, J. Kittler, R. Cipolla, Discriminative learning and recognition of image set classes using canonical correlations, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 1005–1018.

[6] O. Yamaguchi, K. Fukui, K.-i. Maeda, Face recognition using temporal image sequence, in: IEEE International Conference on Automatic Face and Gesture Recognition, IEEE, 1998, pp. 318–323.

[7] S. Shirazi, M.T. Harandi, C. Sanderson, A. Alavi, B.C. Lovell, Clustering on grassmann manifolds via kernel embedding with application to action analysis, in: IEEE International Conference on Image Processing, IEEE, 2012, pp. 781–784.

[8] P. Turaga, A. Veeraraghavan, A. Srivastava, R. Chellappa, Statistical computations on grassmann and stiefel manifolds for image and video-based recognition, IEEE Trans. Pattern Anal. Mach. Intell. 33 (11) (2011) 2273–2286.

[9] C. Ye, K. Slavakis, P.V. Patil, J. Nakuci, S.F. Muldoon, J. Medaglia, Network clustering via kernel-ARMA modeling and the Grassmannian: The brain-network case, Signal Process. 179 (2021) 107834.

[10] S. Karygianni, P. Frossard, Tangent-based manifold approximation with locally linear models, Signal Process. 104 (2014) 232–247.

[11] F. Anowar, S. Sadaoui, B. Selim, Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE), Comput. Sci. Rev. 40 (2021) 100378.

[12] W. Jia, M. Sun, J. Lian, S. Hou, Feature dimensionality reduction: a review, Complex & Intell. Syst. 8 (3) (2022) 2663–2693.

[13] K. Pearson, LIII. On lines and planes of closest fit to systems of points in space, Lond. Edinb. Dublin Philos. Mag. J. Sci. 2 (11) (1901) 559–572.

[14] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugen. 7 (2) (1936) 179–188.

[15] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[16] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.

[17] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.

[18] A.W. Long, A.L. Ferguson, Landmark diffusion maps (L-dMaps): Accelerated manifold learning out-of-sample extension, Appl. Comput. Harmon. Anal. 47 (1) (2019) 190–211.

[19] H. Strange, R. Zwiggelaar, A generalised solution to the out-of-sample extension problem in manifold learning, in: AAAI Conference on Artificial Intelligence, Vol. 25, (1) 2011, pp. 471–476.

[20] X. He, D. Cai, S. Yan, H.-J. Zhang, Neighborhood preserving embedding, in: IEEE International Conference on Computer Vision Volume 1, Vol. 2, IEEE, 2005, pp. 1208–1213.

[21] M. Harandi, C. Sanderson, C. Shen, B.C. Lovell, Dictionary learning and sparse coding on grassmann manifolds: An extrinsic solution, in: IEEE International Conference on Computer Vision, 2013, pp. 3120–3127.

[22] J. Hamm, D.D. Lee, Grassmann discriminant analysis: a unifying view on subspace-based learning, in: International Conference on Machine Learning, 2008, pp. 376–383.

[23] M.T. Harandi, C. Sanderson, S. Shirazi, B.C. Lovell, Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching, in: IEEE/CVF Computer Vision and Pattern Recognition Conference, IEEE, 2011, pp. 2705–2712.

[24] B. Wang, Y. Hu, J. Gao, Y. Sun, M. Ali, H. Chen, B. Yin, Locality preserving projections for Grassmann manifold, in: International Joint Conference on Artificial Intelligence, 2017, pp. 2893–2900.
[25] D. Wei, X. Shen, Q. Sun, X. Gao, Z. Ren, Neighborhood preserving embedding on Grassmann manifold for image-set analysis, Pattern Recognit. 122 (2022) 108335.
[26] D. Wei, X. Shen, Q. Sun, X. Gao, Z. Ren, Learning adaptive Grassmann neighbors for image-set analysis, Expert Syst. Appl. 247 (2024) 123316.
[27] B. Raducanu, F. Dornaika, A supervised non-linear dimensionality reduction approach for manifold learning, Pattern Recognit. 45 (6) (2012) 2432–2444.
[28] J.T. Vogelstein, E.W. Bridgeford, M. Tang, D. Zheng, C. Douville, R. Burns, M. Maggioni, Supervised dimensionality reduction for big data, Nat. Commun. 12 (1) (2021) 2872.
[29] J. Wen, Z. Tian, H. She, W. Yan, Feature extraction of hyperspectral images based on preserving neighborhood discriminant embedding, in: International Conference on Image Analysis and Signal Processing, IEEE, 2010, pp. 257–262.
[30] Y. Song, F. Nie, C. Zhang, S. Xiang, A unified framework for semi-supervised dimensionality reduction, Pattern Recognit. 41 (9) (2008) 2789–2799.
[31] D. Zhang, Z.-H. Zhou, S. Chen, Semi-supervised dimensionality reduction, in: SIAM International Conference on Data Mining, SIAM, 2007, pp. 629–634.
[32] M. Mehdizadeh, C. MacNish, R.N. Khan, M. Bennamoun, Semi-supervised neighborhood preserving discriminant embedding: a semi-supervised subspace learning algorithm, in: Asian Conference on Computer Vision, Springer, 2011, pp. 199–212.
[33] H. Yu, K. Xia, Y. Jiang, P. Qian, Fréchet mean-based Grassmann discriminant analysis, Multimedia Syst. 26 (1) (2020) 63–73.
[34] A. Srivastava, E. Klassen, Bayesian and geometric subspace tracking, Adv. in Appl. Probab. 36 (1) (2004) 43–56.
[35] Z. Huang, R. Wang, S. Shan, X. Chen, Projection metric learning on grassmann manifold with application to video based face recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 140–149.
[36] N. Sogi, L.S. Souza, B.B. Gatto, K. Fukui, Metric learning with a-based scalar product for image-set recognition, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 850–851.
[37] R. Wang, X.-J. Wu, J. Kittler, Graph embedding multi-kernel metric learning for image set classification with Grassmannian manifold-valued features, IEEE Trans. Multimed. 23 (2020) 228–242.
[38] W. Al-Samhi, M. Al-Soswa, Y. Al-Dhabi, Time series data classification on Grassmann manifold, J. Phys.: Conf. Ser. 1848 (1) (2021) 012037.
[39] Z. Huang, J. Wu, L. Van Gool, Building deep networks on grassmann manifolds, in: AAAI Conference on Artificial Intelligence, Vol. 32, (1) 2018.
[40] C.-H. Yang, B.C. Vemuri, Nested Grassmanns for dimensionality reduction with applications to shape analysis, in: International Conference on Information Processing in Medical Imaging, Springer, 2021, pp. 136–149.
[41] M. Mohammadi, M. Babai, M. Wilkinson, Generalized relevance learning Grassmann quantization, IEEE Trans. Pattern Anal. Mach. Intell. (2024).
[42] Y. Wang, G. Mori, Human action recognition by semilatent topic models, IEEE Trans. Pattern Anal. Mach. Intell. 31 (10) (2009) 1762–1774.
[43] S. Yu, T. Tan, K. Huang, K. Jia, X. Wu, A study on gait-based gender classification, IEEE Trans. Image Process. 18 (8) (2009) 1905–1910.
[44] B. Leibe, B. Schiele, Analyzing appearance and contour based methods for object categorization, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2, IEEE, 2003, pp. II–409.
[45] A.S. Georghiades, P.N. Belhumeur, D.J. Kriegman, From few to many: Illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[46] K. Lai, L. Bo, X. Ren, D. Fox, Detection-based object labeling in 3d scenes, in: IEEE International Conference on Robotics and Automation, IEEE, 2012, pp. 1330–1337.
[47] A.B. Chan, N. Vasconcelos, Probabilistic kernels for the classification of auto-regressive visual processes, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, IEEE, 2005, pp. 846–851.
[48] M.D. Rodriguez, J. Ahmed, M. Shah, Action mach a spatio-temporal maximum average correlation height filter for action recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.
[49] L. Xia, C. Chen, J. Aggarwal, View invariant human action recognition using histograms of 3D joints, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, IEEE, 2012, pp. 20–27.
[50] M. Leland, H. John, S. Nathaniel, G. Lukas, UMAP: uniform manifold approximation and projection, J. Open Source Softw. 3 (29) (2018) 861.