PeerJ Computer Science

A weighted sparse coding model on product Grassmann manifold for video-based human gesture recognition

Yuping Wang[1] and Junfei Zhang[2]

[1] School of Statistics, Capital University of Economics and Business, Beijing, China
[2] School of Statistics and Mathematics, Central University of Finance and Economics, Beijing, China

ABSTRACT

It is a challenging problem to classify multi-dimensional data with complex intrinsic geometry inherent, such as human gesture recognition based on videos. In particular, manifold structure is a good way to characterize intrinsic geometry of multi-dimensional data. The recently proposed sparse coding on Grassmann manifold shows high discriminative power in many visual classification tasks. It represents videos on Grassmann manifold using Singular Value Decomposition (SVD) of the data matrix by vectorizing each image in videos, while vectorization destroys the spatial structure of videos. To keep the spatial structure of videos, they can be represented as the form of data tensor. In this paper, we firstly represent human gesture videos on product Grassmann manifold (PGM) by Higher Order Singular Value Decomposition (HOSVD) of data tensor. Each factor manifold characterizes features of human gesture video from different perspectives and can be understood as appearance, horizontal motion and vertical motion of human gesture video respectively. We then propose a weighted sparse coding model on PGM, where weights can be understood as modeling the importance of factor manifolds. Furthermore, we propose an optimization algorithm for learning coding coefficients by embedding each factor Grassmann manifold into symmetric matrices space. Finally, we give a classification algorithm, and experimental results on three public datasets show that our method is competitive to some relevant excellent methods.

Subjects Computer Vision, Data Mining and Machine Learning, Multimedia

Keywords Product Grassmann manifold, Sparse coding, Video classification, Human gesture recognition

Submitted 10 November 2021
Accepted 18 February 2022
Published 16 March 2022

Corresponding author
Junfei Zhang,
junfeizhang_cufe@163.com

Academic editor
Yilun Shang

Additional Information and Declarations can be found on page 17

DOI 10.7717/peerj-cs.923

INTRODUCTION

Human action/gesture recognition (*Pareek & Thakkar, 2021*) is a hot research area due to its wide applications such as human–computer interaction, robot control, security and survillance, sign language assistance, education, medical, etc. Roughly speaking, human actions /gestures convey intentional information by physical movement of body parts. Usually, the term "action" is considered with a higher complexity level comparing to the term "gesture" (*Zhu et al., 2016*). Researches for human gesture recognition are mainly divided into two categories: wearable device based techniques (*Jung et al., 2015*) and vision-based techniques (*Ji et al., 2012*). However, wearing devices requires users to carry special designed wearable sensors and sensors are usually quite expensive. For vision-based

How to cite this article Wang Y, Zhang J. 2022. A weighted sparse coding model on product Grassmann manifold for video-based human gesture recognition. *PeerJ Comput. Sci.* **8**:e923 http://doi.org/10.7717/peerj-cs.923

approaches, videos carry more information for gesture recognition than still images. Moreover, the number of available videos on the Internet significantly increased with the development of acquisition and storage device. Hence, video-based human gesture recognition (*Ji et al., 2012*; *Chakraborty et al., 2018*; *Patil & Subbaraman, 2019*) attracts more and more attentions.

For video-based human gesture recognition, each video is assigned to a class label and videos of the same class maybe acted by different person in different environment. It becomes more difficult for gesture recognition due to large variations, such as illumination, appearance, pose and scale. There exist variations even though for the same person. Therefore it is a challenging problem for video-based human gesture recognition. Basically, the key problems of video-based human gesture recognition are learning discriminative feature representations for a gesture video and designing an effective recognition method.

For feature representation, some researches focused on handcrafted approaches, such as HOG-3D (*Klaser, Marszałek & Schmid, 2008*), space–time interest point (*Laptev, 2005*), pose-based techniques (*Carreira et al., 2016*), motion-based techniques (*Paul, Haque & Chakraborty, 2013*), shape-based techniques (*Vishwakarma & Kapoor, 2015*). Some researches focused on learning-based approaches which can be roughly divided into non-neural network and neural network learning approaches. The latter approaches received good recognition performances because it is designed to mimic human nervous system biologically, such as 3D ConvNets (*Baccouche et al., 2011*; *Tran et al., 2015*; *Feichtenhofer, Pinz & Wildes, 2016*) and variational autoencoder(VAE) (*Spurr et al., 2018*; *Chen et al., 2019*). Millions of parameters need to be learned by training networks and large amounts of data are often required. For non-neural network learning approaches, subspace is a robust representation and had received good performance for many problems in computer vision field (*Le et al., 2011*; *Sheng et al., 2019*). The reason is that most data often have intrinsic subspace structure and can be regarded as samples of subspace. Moreover, subspace-based feature representation method can learn features directly from image or video data without hand-designed local feature. For investigating and representing the underlying intrinsic subspace structure, many subspace methods were proposed, such as linear subspace learning (PCA (*Wold, Esbensen & Geladi, 1987*), FLDA (*Belhumeur, Hespanha & Kriegman, 1997*; *Mohammadzade, Sayyafan & Ghojogh, 2018*)) and non-linear manifold learning (Isomap (*Pless, 2003*), LLE (*Ge, Yang & Lee, 2008*), LE (*Luo, 2011*)). As an excellent representative, Grassmann manifold received widely applications such as activity classification (*Turaga & Chellappa, 2009*), action recognition (*Rahimi, Aghagolzadeh & Ezoji, 2019*), face recognition (*Huang et al., 2015*) and so on.

For recognition methods, sparsity representation classification (SRC) had been shown to deliver notable results for various visual-based tasks, such as face recognition (*Wright et al., 2008*; *Wright et al., 2010*), subspace clustering (*Elhamifar & Vidal, 2013*). Furthermore, some weighted forms for sparse coding were proposed for various applications, such as image denoising (*Xu, Zhang & Zhang, 2018*), visual tracking (*Yan & Tong, 2011*) and saliency detection (*Li, Sun & Yu, 2015*). Although the SRC method and its extended models had good performance in many applications, they assumed data come from linear space. However, many multi-dimensional data may reside in a non-linear manifold space. So it is desire to explore the latent non-linear manifold structure of data. Recently, for Grassmann manifold representation of videos/image sets, many researches had been proposed for kinds of applications and received good performance. For instance, *Harandi et al. (2015)* proposed a sparse coding algorithm on Grassmann manifold for classification tasks such as gesture classification, scene analysis and dynamic texture classification; *Wang et al. (2020)* proposed a self-expression learning framework on Grassmann manifolds for video/image-set subspace clustering; *Verma & Choudhary (2020)* did Grassmann manifold discriminant analysis for hand gesture recognition from depth data; *Souza et al. (2020a)* proposed an enhanced Grassmann discriminant analysis framework for classifying motion sequences.

Although the Grassmann manifold can well reflect the non-linear structure of data, the single space representation methods lose some important information by vectorizing each image in videos. Naturally, video and image set can be represented in the form of data tensor. Tensor computing had been successfully applied to many visual-based application (*Kim & Cipolla, 2008*). *Lui (2012)* factorized a data tensor using Higher Order Singular Value Decomposition (HOSVD) and imposed each factorized element on a Grassmann manifold, then a video can be represented as a point on product Grassmann manifold (PGM). This representation yielded a very discriminating structure for action recognition. *Wang et al. (2016)* proposed a low rank representation model on PGM, which received good performance for clustering of videos or image sets. *Wang et al. (2018)* proposed an extrinsic least square regression on PGM for video-based recognition.

In this paper, we represent a human gesture video as a point on PGM. In brief, there are three factor Grassmann manifolds which can reflect appearance, horizontal motion and vertical motion of human gesture video respectively. In addition, the importance of these three aspects should be considered. Hence, we explore a weighted sparse coding method on PGM for video-based human gesture recognition. It is solved by minimizing the reconstruction error with a l_1-norm regularizer.

Our main contributions lie in the following three aspects:

(1) Extending SRC model on Grassmann manifold into product Grassmann manifold to deal with multi-dimensional data such as videos and image-sets.

(2) Discussing the different importance of three factor manifolds and proposing a weighted sparse coding model.

(3) Comparing with several classification methods on three datasets to show the effectiveness of our proposed method.

The rest of this paper is organized as follows: 'Product Grassmann Manifold Representation for Data' introduces product Grassmann manifold representation for data; 'Weighted Sparse Coding on Product Grassmann Manifold' gives a weighted sparse coding model on PGM; 'Experiments' shows experiments on different datasets, and experiment results show that the proposed method achieves considerable accuracy; 'Computational Complexity' analyzes the computational complexity of our proposed method; 'Main Findings and Future Directions' gives main findings and future directions.

PRODUCT GRASSMANN MANIFOLD REPRESENTATION FOR DATA

In the following paper, we use the mathematical symbols in Table 1 which are commonly used.

Product Grassmann manifold

A point on Grassmann manifold $\mathcal{G}(p,d)$ is a p-dimensional subspace of $\mathbb{R}^d$ (*Absil, Mahony & Sepulchre, 2009*). That means it can be spanned by any orthonormal basis $\mathbf{X}=[\mathbf{x}_1|\mathbf{x}_2|\cdots|\mathbf{x}_p]\in\mathbb{R}^{d\times p}$ and it is denoted as $\mathrm{span}(\mathbf{X})$. For the sake of convenience, we use the same symbol $\mathbf{X}$ to represent $\mathrm{span}(\mathbf{X})$. The distance of two points $\mathbf{X}$ and $\mathbf{Y}$ on Grassmann manifold can be defined as

$$d_g(\mathbf{X},\mathbf{Y})=\|\Pi(\mathbf{X})-\Pi(\mathbf{Y})\|_F=\|\mathbf{X}\mathbf{X}^T-\mathbf{Y}\mathbf{Y}^T\|_F$$

where embedding mapping $\Pi:\mathcal{G}(p,d)\rightarrow\mathrm{Sym}(d)$ is defined as $\Pi(\mathbf{X})=\mathbf{X}\mathbf{X}^T$, and $\mathrm{Sym}(d)$ is the symmetric matrices space with order d (refer to *Harandi et al., 2015*). Product Grassmann manifold (PGM) $\mathcal{PG}(p_1,\ldots,p_M|\,d_1,\ldots,d_M)$ is defined as

$$\mathcal{PG}(p_1,\ldots,p_M|d_1,\ldots,d_M)=\mathcal{G}(p_1,d_1)\times\cdots\times\mathcal{G}(p_M,d_M)$$

where the symbol × denotes Cartesian product, $\mathcal{G}(p_i,d_i)$ $(i=1,\ldots,M)$ is called factor manifold and $p_i(i=1,\cdots,M)$ is called dimension of each factor manifold. A point on PGM is denoted as $[\mathbf{X}]=(\mathbf{X}^1,\ldots,\mathbf{X}^M)$. The distance between two points $[\mathbf{X}]=(\mathbf{X}^1,\ldots,\mathbf{X}^M)$ and $[\mathbf{Y}]=(\mathbf{Y}^1,\ldots,\mathbf{Y}^M)$ on PGM is defined as weighted average distance of each factor Grassmann manifold

$$d_{\mathcal{PG}}([\mathbf{X}],[\mathbf{Y}])=\sqrt{\sum_{m=1}^{M}\omega_m d_g^2(\mathbf{X}^m,\mathbf{Y}^m)}$$

where each weight $\omega_m(\geq 0)$ represents the importance of factor manifold $\mathcal{G}(p_m,d_m)$ and $\sum\limits_{m=1}^{M}\omega_m=1$.

Data representation on PGM

In the real world, there exists many data with multi-dimensional structure. For example, video can be represented as tensor $\mathcal{A}\in\mathbb{R}^{J_1\times J_2\times J_3}$, where J_1, J_2 and J_3 represent height, width and length of video respectively; Image set can be represented as tensor $\mathcal{A}\in\mathbb{R}^{J_1\times J_2\times J_3}$, where J_1, J_2 and J_3 represent height, width and number of image set respectively; Light field can be represented as tensor $\mathcal{A}\in\mathbb{R}^{J_1\times J_2\times J_3\times J_4}$ (*Wang & Zhang, 2020*), where J_1 and J_2 represent angular resolution of light field, J_3 and J_4 represent spatial resolution of light field.

Before introducing data representation on PGM, we give a schematic of matrix unfolding for a third tensor in Fig. 1. The reader can refer to *Kolda & Bader (2009)* for more theory on tensor operation. For ease of understanding we give a corresponding example of two videos described by tensor in Fig. 2. We find that the corresponding unfolding

Table 1 Mathematical symbols in this paper.

Symbol	Description
$\mathbf{X},\mathbf{Y},\ldots$	a matrix
$\mathbf{x},\mathbf{y},\ldots$	a vector
$\mathcal{X},\mathcal{Y},\ldots$	a tensor
$N,M,d,p,\ldots$	scalar
$\mathbf{x}_i,\ldots$	the ith column of matrix $\mathbf{X}$
$x_{ij},\ldots$	the (i,j)-th element of matrix $\mathbf{X}$
$\mathbf{X}^T$	the transpose of matrix $\mathbf{X}$
$\mathrm{Tr}(\cdot)$	sum of the diagonal elements of a matrix
$\|\cdot\|_F$	$\|\mathbf{X}\|_F=\sqrt{\mathrm{Tr}(\mathbf{X}^T\mathbf{X})}$
$\|\cdot\|_1$	$\|\mathbf{X}\|_1=\sum_{i,j}\lvert x_{ij}\rvert$

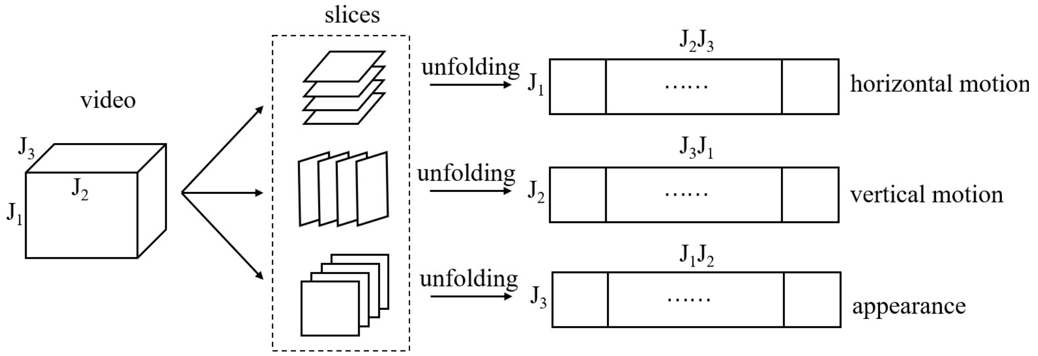


Figure 1 A schematic of matrix unfolding for a video tensor. J_1, J_2 and J_3 represent height, width and length of video respectively.

Full-size DOI: 10.7717/peerjcs.923/fig-1

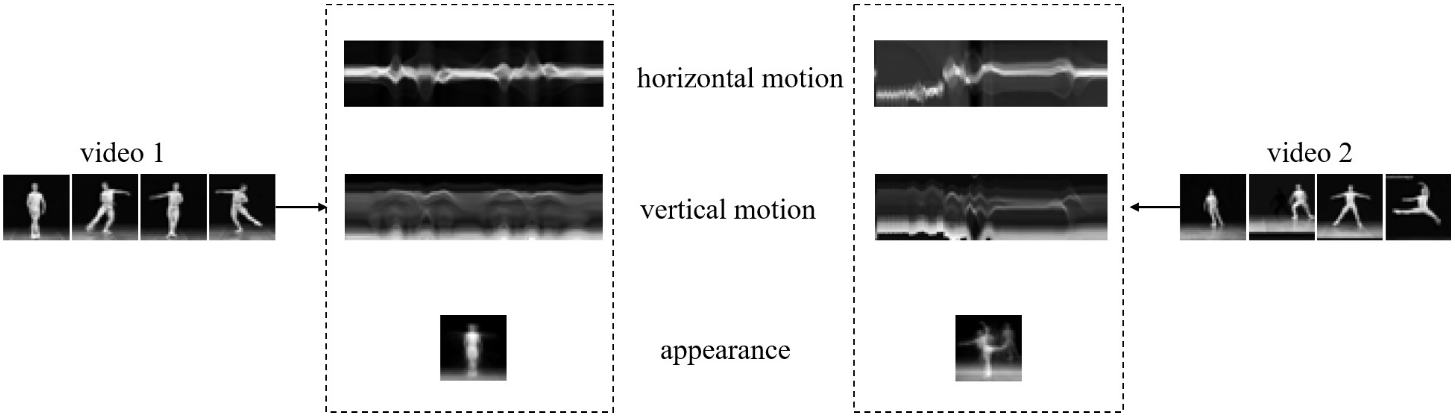


Figure 2 A visual example of matrix unfolding. Two videos with different labels are shown for comparison, which come from Ballet datasets (it will be discussed in 'Experiments'). The two dashed frames show overlay horizontal motion, vertical motion and appearance of video 1 and video 2 respectively.

Full-size DOI: 10.7717/peerjcs.923/fig-2 matrix is discriminative for two videos with different labels, hence the multi-dimensional information of video tensor is worth mining for classification task.

In the following, we discuss the way to represent multi-dimensional data on PGM. The variation for each mode of a tensor $\mathcal{A} \in \mathbb{R}^{J_1 \times \cdots \times J_M}$ can be captured by HOSVD (followed as *Lui, 2012*), which factorize tensor $\mathcal{A}$ using the orthogonal matrices in the following equation:

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{V}^{(1)} \times_2 \cdots \times_M \mathbf{V}^{(M)}$$

where $\mathbf{V}^{(m)} \in \mathbb{R}^{J_m \times d_m}$ $(m = 1\ ,\ldots, M)$ are orthogonal matrices spanning the row space with the first J_m rows associated with non-zero singular values from the unfolded matrices respectively, $\mathcal{S} \in \mathbb{R}^{d_1 \times \cdots \times d_M}$ is a core tensor, $d_m = \prod_{i \neq m} J_i$, and $\times_m (m = 1\ ,\ldots, M)$ denotes mode- m multiplication. Each $\mathbf{V}^{(m)^T} \in \mathbb{R}^{d_m \times J_m}$ is a tall orthogonal matrix. We take the first p_m $(p_m \leq J_m)$ columns of $\mathbf{V}^{(m)^T}$ and denote it as $\mathbf{U}^{(m)} \in \mathbb{R}^{d_m \times p_m}$. Hence, $\mathbf{U}^{(m)}$ is a point on Grassmann manifold $\mathcal{G}(p_m, d_m)$. And then $(\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(M)})$ is a point on PGM $\mathcal{G}(p_1, d_1) \times \cdots \times \mathcal{G}(p_M, d_M)$.

Remark: The value of parameter $p_m (m = 1\ ,\ldots, M)$ reflects the principal information of data. In brief, the information of data may be redundant if the value of p_m is too large and the information of data may be insufficient if the value of p_m is too small. Hence it is important to select the parameters $p_m (m = 1\ ,\ldots, M)$ and we will discuss this problem in details in our experiments.

WEIGHTED SPARSE CODING ON PRODUCT GRASSMANN MANIFOLD

Weighted sparse coding model on PGM

Let $\{[\mathbf{X}_1], \ldots, [\mathbf{X}_N]\}$ be the training set which includes N samples, where $[\mathbf{X}_i] = (\mathbf{X}_i^1, \ldots, \mathbf{X}_i^M) \in \mathcal{PG}(p_1, \ldots, p_M | d_1, \ldots, d_M)$ is a point on product Grassmann manifold. Let $[\mathbf{Y}] = (\mathbf{Y}^1, \ldots, \mathbf{Y}^M) \in \mathcal{PG}(p_1,\ldots,p_M | d_1,\ldots,d_M)$ be a query sample on product Grassmann manifold. The sparse coding model on PGM is formulated as follows:

$$\min_{\boldsymbol{\alpha}} d^2_{\mathcal{PG}}([\mathbf{Y}], \biguplus_{i=1}^{N} \alpha_i \odot [\mathbf{X}_i]) + \lambda \|\boldsymbol{\alpha}\|_1$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)^T$ is the sparse representation coefficient, the abstract symbols $\biguplus_{i=1}^{N}$ and $\odot$ are used to simulate "linear" combination defined on PGM, *i.e.*, addition and scalar-mulitplication. $d_{\mathcal{PG}}([\mathbf{Y}], \biguplus_{i=1}^{N} \alpha_i \odot [\mathbf{X}_i])$ measures the distance between reconstruction $\biguplus_{i=i}^{N} \alpha_i \odot [\mathbf{X}_i]$ and the query sample $[\mathbf{Y}]$. To get the sparse coding model on PGM, proper definitions of distance and combination operator should be specified. According to the geometric property of Grassmann manifold, we use the embedded distance and linear combination on the space of symmetric matrices. Hence, we construct the weighted sparse coding model on PGM as follows,

$$\min_{\boldsymbol{\alpha}} \sum_{m=1}^{M} \omega_m \|\mathbf{Y}^m \mathbf{Y}^{mT} - \sum_{i=1}^{N} \alpha_i (\mathbf{X}_i^m \mathbf{X}_i^{mT})\|_F^2 + \lambda \|\boldsymbol{\alpha}\|_1. \tag{1}$$

Algorithm for the weighted sparse coding on PGM

In this subsection, we show how to solve the optimization Eq. (1). We have

$$\begin{aligned}\min_{\boldsymbol{\alpha}}\sum_{m=1}^{M}\omega_m\|\mathbf{Y}^m\mathbf{Y}^{m^T}-\sum_{i=1}^{N}\alpha_i(\mathbf{X}_i^m\mathbf{X}_i^{m^T})\|_F^2\\
=\min_{\boldsymbol{\alpha}}\sum_{m=1}^{M}\omega_m\big\{\mathrm{Tr}(\mathbf{Y}^{m^T}\mathbf{Y}^m\mathbf{Y}^{m^T}\mathbf{Y}^m)+\sum_{i,j=1}^{N}\alpha_i\alpha_j\mathrm{Tr}(\mathbf{X}_i^{m^T}\mathbf{X}_j^m\mathbf{X}_j^{m^T}\mathbf{X}_i^m)\\
-2\sum_{i=1}^{N}\alpha_i\mathrm{Tr}(\mathbf{X}_i^{m^T}\mathbf{Y}^m\mathbf{Y}^{m^T}\mathbf{X}_i^m)\big\}.\end{aligned}$$

For simplicity, we define a matrix $K^m(\mathbf{X})$ and a vector $K^m(\mathbf{X},\mathbf{Y})$ as following, *i.e.*, their elements are

$[K^m(\mathbf{X})]_{ij}=\omega_m\mathrm{Tr}(\mathbf{X}_i^{m^T}\mathbf{X}_j^m\mathbf{X}_j^{m^T}\mathbf{X}_i^m),\;\; i{,}j{=}1\;,\ldots,N$

$[K^m(\mathbf{X},\mathbf{Y})]_i=\omega_m\mathrm{Tr}(\mathbf{X}_i^{m^T}\mathbf{Y}^m\mathbf{Y}^{m^T}\mathbf{X}_i^m),\;\; i=1,\ldots,N$

Hence the model Eq. (1) becomes

$$\begin{aligned}\min_{\boldsymbol{\alpha}}\sum_{m=1}^{M}\big\{\boldsymbol{\alpha}^TK^m(\mathbf{X})\boldsymbol{\alpha}-2\boldsymbol{\alpha}^TK^m(\mathbf{X},\mathbf{Y})\big\}+\lambda\|\boldsymbol{\alpha}\|_1\\
=\min_{\boldsymbol{\alpha}}\boldsymbol{\alpha}^T\left(\sum_{m=1}^{M}K^m(\mathbf{X})\right)\boldsymbol{\alpha}-2\boldsymbol{\alpha}^T\left(\sum_{m=1}^{M}K^m(\mathbf{X},\mathbf{Y})\right)+\lambda\|\boldsymbol{\alpha}\|_1.\end{aligned}$$

The symemetric matrix $\sum_{m=1}^{M}K^m(\mathbf{X})$ is positive semidefinite since for all $\mathbf{v}=(v_1,v_2,\ldots,v_N)^T\in\mathbb{R}^N$:

$$\begin{aligned}&\mathbf{v}^T\left(\sum_{m=1}^{M}K^m(\mathbf{X})\right)\mathbf{v}\\
&=\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\omega_mv_iv_j\mathrm{Tr}(\mathbf{X}_i^{m^T}\mathbf{X}_j^m\mathbf{X}_j^{m^T}\mathbf{X}_i^m)\\
&=\sum_{m=1}^{M}\omega_m\mathrm{Tr}\left(\sum_{i=1}^{N}\sum_{j=1}^{N}v_iv_j\mathbf{X}_i^{m^T}\mathbf{X}_j^m\mathbf{X}_j^{m^T}\mathbf{X}_i^m\right)\\
&=\sum_{m=1}^{M}\omega_m\|\sum_{i=1}^{N}v_i\mathbf{X}_i^m\mathbf{X}_i^{m^T}\|_F^2\geq 0.\end{aligned}$$

Therefore, the problem is convex and can be solved by a vectorized sparse coding problem. In detail, let $\mathbf{U}\Sigma\mathbf{U}^T$ be the SVD of $\sum_{m=1}^{M}K^m(\mathbf{X})$, then the problem is equal to

$$\min_{\boldsymbol{\alpha}}\|\mathbf{Y}^*-\mathbf{A}\boldsymbol{\alpha}\|^2+\lambda\|\boldsymbol{\alpha}\|_1 \tag{2}$$

where $\mathbf{A}=\Sigma^{1/2}\mathbf{U}^T$ and $\mathbf{Y}^*=\Sigma^{-1/2}\mathbf{U}^T\left(\sum_{m=1}^{M}K^m(\mathbf{X},\mathbf{Y})\right)$. The pseudo-code for performing the proposed weighted sparse coding on PGM is summarized in Algorithm 1, which is simply called WSC-PGM.

Algorithm 1 Weighted sparse coding on product Grassmann manifold (WSC-PGM)

```
Require:
  Training data includes $N$ samples on PGM: $[\mathbf{X}_i] = (\mathbf{X}_i^1, \mathbf{X}_i^2, \ldots, \mathbf{X}_i^M)$, $i = 1, 2, \ldots, N$ and
  $\mathbf{X}_i^m \in \mathcal{G}(p_m, d_m)$, $m = 1, 2, \ldots, M$; the query sample on PGM: $[\mathbf{Y}] = (\mathbf{Y}^1, \mathbf{Y}^2, \ldots, \mathbf{Y}^M)$ and
  $\mathbf{Y}^m \in \mathcal{G}(p_m, d_m)$,m=$1, 2, \ldots, M$.
Ensure:
  The sparse code $\alpha^*$
  for $m = 1 : M$ do
    for $i = 1 : N$ do
      for $j = 1 : N$ do
        $[K^m(\mathbf{X})]_{ij} = \omega_m \mathrm{Tr}(\mathbf{X}_i^{mT}\mathbf{X}_j^m\mathbf{X}_j^{mT}\mathbf{X}_i^m)$        /* compute matrix $K^m(\mathbf{X})$
      end for
      $[K^m(\mathbf{X},\mathbf{Y})]_i = \omega_m \mathrm{Tr}(\mathbf{X}_i^{mT}\mathbf{Y}^m\mathbf{Y}^{mT}\mathbf{X}_i^m)$        /* compute vector $K^m(\mathbf{X},\mathbf{Y})$
    end for
  end for
  $\sum_{m=1}^{M} K^m(\mathbf{X}) = \mathbf{U}\Sigma\mathbf{U}^T$        /* compute SVD of $\sum_{m=1}^{M} K^m(\mathbf{X})$
  $\mathbf{A} \leftarrow \Sigma^{1/2}\mathbf{U}^T$
  $\mathbf{Y}^* \leftarrow \Sigma^{-1/2}\mathbf{U}^T \left(\sum_{m=1}^{M} K^m(\mathbf{X},\mathbf{Y})\right)$
  $\alpha^* = \underset{\alpha}{\mathrm{argmin}} \|\mathbf{Y}^* - \mathbf{A}\alpha\|^2 + \lambda\|\alpha\|_1$        /* the solution of model (2)
  return $\alpha^*$
```

Classification rule and algorithm

When model Eq. (1) is minimized, the optimal coefficient $\boldsymbol{\alpha}^*$ can be used for classification. Following the idea of the Sparse Representation Classification (SRC) (*Wright et al., 2008*), the query sample can be classified by it's codes α^* of these labeled training samples $[\mathbf{X}_i]$i =(1 ,2,...,N).

In details, let $(\alpha_1^*\delta(l_1 - k), \alpha_2^*\delta(l_2 - k), \ldots, \alpha_N^*\delta(l_N - k))^T$ be the class- k sparse codes, where l_i(i =1 ,2,...,N) is the class label of training sample $[\mathbf{X}_i]$ and $\delta(x)$ is the discrete Dirac function.

$$\delta(x) = \begin{cases} 1 & x = 0 \\ 0 & else \end{cases}.$$

The residual error of a query sample $[\mathbf{Y}]$ =$(\mathbf{Y}^1, \mathbf{Y}^2, \ldots, \mathbf{Y}^M)$ by using the samples associated to class k is defined as

$$\varepsilon_k([\mathbf{Y}]) = \sum_{m=1}^{M} \omega_m \|\mathbf{Y}^m\mathbf{Y}^{mT} - \sum_{i=1}^{N} \alpha_i^*(\mathbf{X}_i^m\mathbf{X}_i^{mT})\delta(l_i - k)\|_F^2. \tag{3}$$

Then the estimated class of the query $\mathbf{Y}$ is determined by

$$\mathrm{Label}([\mathbf{Y}]) = \underset{k}{\mathrm{argmin}}\, \varepsilon_k([\mathbf{Y}]). \tag{4}$$

The procedure of sparse representation classification on product Grassmann manifold is summarized in Algorithm 2.

Algorithm 2 Weighted sparse representation classification on product Grassmann manifold (WSRC-PGM)

```
Require:
    Training data $[\mathbf{X}_i] = (\mathbf{X}_i^1, \mathbf{X}_i^2, \ldots, \mathbf{X}_i^M)$, i=1, 2, ..., $N$ belonging to $c$ classes; the query $[\mathbf{Y}] = (\mathbf{Y}^1, \mathbf{Y}^2, \ldots, \mathbf{Y}^M)$
Ensure:
    The class label Label($[\mathbf{Y}]$) of the given test sample $[\mathbf{Y}]$
    Compute $\alpha^*$ as Algorithm 1
    Compute residual $\varepsilon_k([\mathbf{Y}])$ by using equation (3)
    Compute the class label by using equation (4)
    return Label($[\mathbf{Y}]$)
```

EXPERIMENTS

In this section, we show performance of the proposed method against some state-of-the-art methods on three kinds of datasets. In the following experiments, all video data can be regarded as points on PGM $\mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2) \times \mathcal{G}(p_3, d_3)$ and the parameter λ is all chosen as 0.1 by experience.

Cambridge hand gesture datasets

The Cambridge hand gesture datasets (*Kim & Cipolla, 2008*) contains 900 video sequences with 9 classes and it is divided into 5 sets according to different illuminations. The 9 classes are flat-leftward (FL), flat-rightward (FR), flat-contract (FC), spread-leftward (SL), spread-rightward (SR), spread-contract (SC), V-shape-leftward (VL), V-shape-rightward (VR) and V-shape-contract (VC) respectively. We follow the experimental protocol in paper (*Kim & Cipolla, 2008*), set 5 (normal illumination) is considered for training while the remaining sequences (with different illumination characteristics) are used for testing. In this experiment, the original sequences are converted to grayscale and resized to $20 \times 20 \times 20$. Obviously, experiment results depend on the selection of parameters, so we firstly discuss the parameter setting in the following.

Parameter setting

In this subsection, we discuss the parameter setting including dimensions (p_1, p_2, p_3) of three factor Grassmann manifolds and their weights $(\omega_1, \omega_2, \omega_3)$. In fact, we have $\omega_1 + \omega_2 + \omega_3 = 1$ in model Eq. (1). Hence, we jointly determine the parameters $(p_1, p_2, p_3, \omega_1, \omega_2)$. For this datasets, p_1, p_2, p_3 are optimized all in the range of 2 to 20 by step 2, and ω_1, ω_2 are optimized in the range as Table 2. We perform 5-fold cross validation on Set5 and find the optimal $(p_1^*, p_2^*, p_3^*, \omega_1^*, \omega_2^*)$ to obtain the best experimental results. Each time we leave one cross validation set as testing and the other four folds for training. Recursively, we perform experiments and record the correct recognition rate (CRR) of each fold.

Maximizing the average CRRs of five results to have good discrimination, there exist 33 optional parameter combinations. Meanwhile, we expect the data representation carrying more information to better fit the testing data. Hence, among the 33 combinations we choose the top 5 % combinations making $p_1 + p_2 + p_3$ larger. We list the selected

Table 2 The range of ω_1, ω_2.

$\omega_1 \setminus \omega_2$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.1	(0.1, 0.1)	(0.1, 0.2)	(0.1, 0.3)	(0.1, 0.4)	(0.1, 0.5)	(0.1, 0.6)	(0.1, 0.7)	(0.1, 0.8)
0.2	(0.2, 0.1)	(0.2, 0.2)	(0.2, 0.3)	(0.2, 0.4)	(0.2, 0.5)	(0.2, 0.6)	(0.2, 0.7)	
0.3	(0.3, 0.1)	(0.3, 0.2)	(0.3, 0.3)	(0.3, 0.4)	(0.3, 0.5)	(0.3, 0.6)		
0.4	(0.4, 0.1)	(0.4, 0.2)	(0.4, 0.3)	(0.4, 0.4)	(0.4, 0.5)			
0.5	(0.5, 0.1)	(0.5, 0.2)	(0.5, 0.3)	(0.5, 0.4)				
0.6	(0.6, 0.1)	(0.6, 0.2)	(0.6, 0.3)					
0.7	(0.7, 0.1)	(0.7, 0.2)						
0.8	(0.8, 0.1)							

Table 3 The top 5% combinations of parameter on Cambridge hand gesture datasets.

Parameter	p_1^*	p_2^*	p_3^*	ω_1^*	ω_2^*	ω_3^*
combination 1	8	18	12	0.3	0.3	0.4
combination 2	20	10	12	0.2	0.4	0.4
combination 3	14	12	12	0.2	0.4	0.4

combinations of parameters $(p_1, p_2, p_3, \omega_1, \omega_2, \omega_3)$ in Table 3. Table 4 shows the CRRs of the five folds of Set5 with the combinations of parameter in Table 3. In order to illustrate the above parameter selection process, Figs. 3–5 show the slice of CRR's variation with each dimension of parameter corresponding to the optimal combinations listed in Table 3, respectively.

Experiment result on testing sets

In this experiment, the parameter λ is set as 0.1. With the three combinations of parameters $(p_1, p_2, p_3, \omega_1, \omega_2)$, the samples of Set1-Set4 are represented as points on $\mathcal{PG}(8, 18, 12|400, 400, 400)$, $\mathcal{PG}(20, 10, 12|400, 400, 400)$ and $\mathcal{PG}(14, 12, 12|400, 400, 400)$ respectively. Table 5 summarizes the correct recognition rate for Set1-Set4 and the average correct recognition rate which followed by the standard deviation. As Table 5 shows, WSRC-PGM has superior performance compared with TCCA (*Kim & Cipolla, 2008*), PM (*Lui, 2012*), gSC and kgSC (*Harandi et al., 2015*), DMD+SC(SCCD2) (*Singh et al., 2021*).

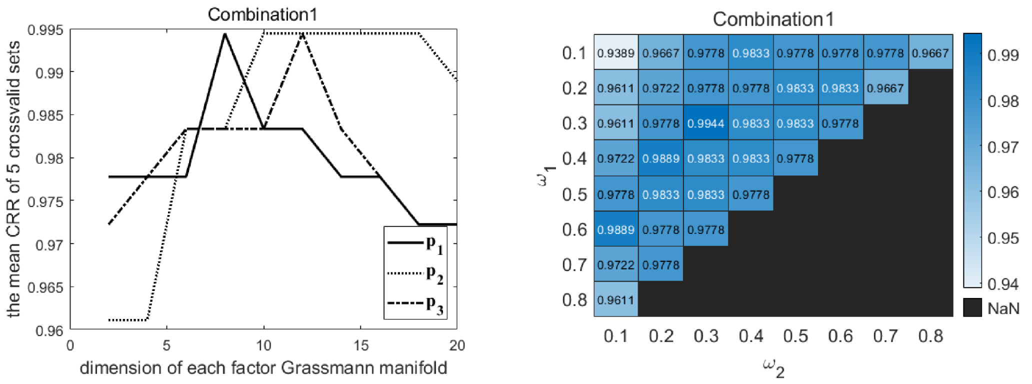
The confusion matrix of our proposed approach on the four testing sets under parameter combination 1 are given in Fig. 6. Naturally, confusion matrices for combination 2 and 3 can be discussed similarly and they are omitted here. From Fig. 6 see, the most misclassified class is SL and most of the misclassified samples with lable SL were misassigned to the SC class. The second most misclassified class is SC and most of the misclassified samples with lable SC were misassigned to the VC class.

Ballet datasets

The Ballet dataset contains 44 videos including 8 complex motion patterns from 3 persons (*Fathi & Mori, 2008*). In detail, the actions are "left-to-right hand opening", "right-to-left hand opening", "standing hand opening", "leg swinging", "jumping", "turning" , "hopping" and "standing still" . Main challenge of this dataset is large variations among

Table 4 **CRR of the five cross validation sets on Cambridge hand gesture datasets.**

Cross validation sets	1	2	3	4	5
CRR of combination 1	100%	100%	100%	97.22%	100%
CRR of combination 2	100%	100%	100%	97.22%	100%
CRR of combination 3	100%	100%	100%	97.22%	100%



(a) The slices with fixed $(\omega_1,\omega_2,\omega_3)$=(0.3,0.3,0.4)

(b) The slice with fixed (p_1,p_2,p_3)=(8,18,12)

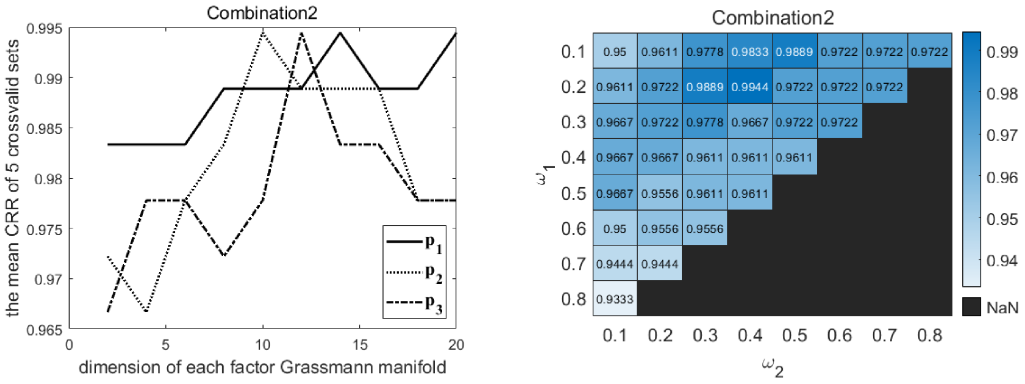
Figure 3 **The two graphs show the slice of CRR's variation with each parameter with combination 1 on the Cambridge Hand Gesture Datasets.** (A) The solid line shows the variation of CRR with varying p_1 while (p_2,p_3) are fixed as $(18,12)$, and the optimal p_1 is 8 in this slice. The dotted line shows the variation of CRR with varying p_2 while (p_1,p_3) are fixed as $(8,12)$, and the optimal p_2 is 18 in this slice. The dash-dot line shows the variation of CRR with varying p_3 while (p_1,p_2) are fixed as $(8,18)$, and the optimal p_3 is 12 in this slice. (B) The heatmap reflects the variation of CRR with different (ω_1,ω_2) and the optimal (ω_1,ω_2) is $(0.3,0.3)$.

Full-size DOI: 10.7717/peerjcs.923/fig-3

Table 5 **Recognition results on the Cambridge hand-gesture dataset.**

Method	Set1	Set2	Set3	Set4	Overall
TCCA (*Kim & Cipolla, 2008*)	81	81	78	86	82 ± 3.5%
PM (*Lui, 2012*)	93	89	91	94	91.7 ± 2.3%
gSC (*Harandi et al., 2015*)	93	92	93	94	93.3 ± 0.9%
kgSC (*Harandi et al., 2015*)	96	92	93	97	94.4 ± 2.0%
HOG3DVV+GGDA (*Verma & Choudhary, 2018*)	86	93	87	93	89.7
WSRC-PGM (combination 1)	98	92	96	97	95.6 ± 2.8%
WSRC-PGM (combination 2)	99	91	94	96	95.0 ± 3.5%
WSRC-PGM (combination 3)	99	89	94	96	94.3 ± 4.2%

classes such as speed, clothing and motion paths. The frame images are normalized and centered in a fixed size of 20 × 20. We extract total 2400 sub-videos by randomly sampling 6 frames from original video that exhibited the same action and then images are converted to grayscale. We randomly select 1200 samples as training set and the remainder as testing set.



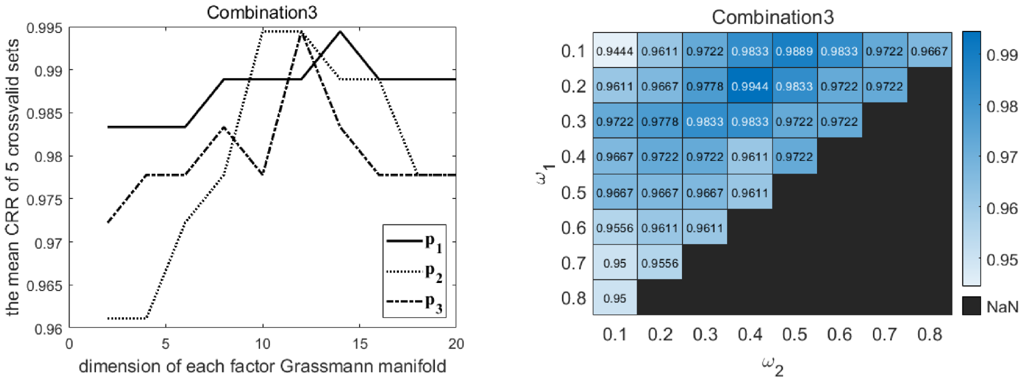
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.1	0.95	0.9611	0.9778	0.9833	0.9889	0.9722	0.9722	0.9722
0.2	0.9611	0.9722	0.9889	0.9944	0.9722	0.9722	0.9722	
0.3	0.9667	0.9722	0.9778	0.9667	0.9722	0.9722		
0.4	0.9667	0.9667	0.9611	0.9611	0.9611			
0.5	0.9667	0.9556	0.9611	0.9611				
0.6	0.95	0.9556	0.9556					
0.7	0.9444	0.9444						
0.8	0.9333							

(a) The slices with fixed $(\omega_1,\omega_2,\omega_3)$=(0.2,0.4,0.4)

(b) The slice with fixed (p_1,p_2,p_3)=(20,10,12)

Figure 4 The two graphs show the slice of CRR's variation with each parameter with combination 2 on the Cambridge Hand Gesture Datasets. (A) The solid line shows the variation of CRR with varying p_1 while (p_2,p_3) are fixed as $(10,12)$, and the optimal p_1 is 20 in this slice. The dotted line shows the variation of CRR with varying p_2 while (p_1,p_3) are fixed as $(20,12)$, and the optimal p_2 is 10 in this slice. The dash-dot line shows the variation of CRR with varying p_3 while (p_1,p_2) are fixed as $(20,10)$, and the optimal p_3 is 12 in this slice. (B) The heatmap reflects the variation of CRR with different (ω_1,ω_2), and the optimal (ω_1,ω_2) is $(0.2,0.4)$.

Full-size DOI: 10.7717/peerjcs.923/fig-4



	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.1	0.9444	0.9611	0.9722	0.9833	0.9889	0.9833	0.9722	0.9667
0.2	0.9611	0.9667	0.9778	0.9944	0.9833	0.9722	0.9722	
0.3	0.9722	0.9778	0.9833	0.9833	0.9722	0.9722		
0.4	0.9667	0.9722	0.9722	0.9611	0.9722			
0.5	0.9667	0.9667	0.9667	0.9611				
0.6	0.9556	0.9611	0.9611					
0.7	0.95	0.9556						
0.8	0.95							

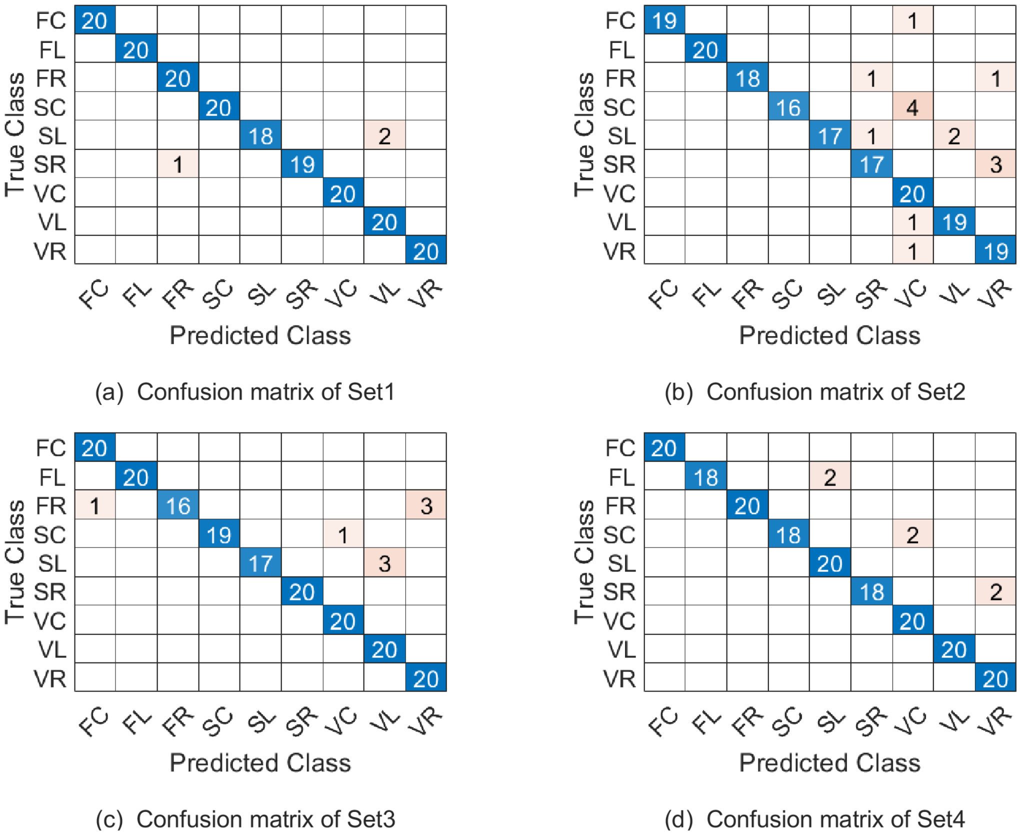
(a) The slices with fixed $(\omega_1,\omega_2,\omega_3)$=(0.2,0.4,0.4)

(b) The slice with fixed (p_1,p_2,p_3)=(14,12,12)

Figure 5 The two graphs show the slice of CRR's variation with each parameter with combination 3 on the Cambridge Hand Gesture Datasets. (A) The solid line shows the variation of CRR with varying p_1 while (p_2,p_3) are fixed as $(12,12)$, and the optimal p_1 is 14 in this slice. The dotted line shows the variation of CRR with varying p_2 while (p_1,p_3) are fixed as $(14,12)$, and the optimal p_2 is 12 in this slice. The dash-dot line shows the variation of CRR with varying p_3 while (p_1,p_2) are fixed as $(14,12)$, and the optimal p_3 is 12 in this slice. (B) The heatmap reflects the variation of CRR with different (ω_1,ω_2) and the optimal (ω_1,ω_2) is $(0.2,0.4)$ in this slice.

Full-size DOI: 10.7717/peerjcs.923/fig-5

Similar to the discussion for parameter setting of experiment on Cambridge hand gesture dataset, we jointly determine the parameters $(p_1,p_2,p_3,\omega_1,\omega_2)$ by 5-fold cross validation on training set, where p_1,p_2 are all in the range of $\{2:2:20\}$, p_3 is in the range of $\{1:1:6\}$ and ω_1,ω_2 are in the range as Table 2. The top 5 % optional parameter combinations of



(a) Confusion matrix of Set1

(b) Confusion matrix of Set2

(c) Confusion matrix of Set3

(d) Confusion matrix of Set4

Figure 6 **The confusion matrix of combination 1 on the Cambridge hand-gesture dataset.**
Full-size DOI: 10.7717/peerjcs.923/fig-6

Table 6 **The top 5% combinations of parameter on Ballet dataset.**

Parameter	p_1^*	p_2^*	p_3^*	ω_1^*	ω_2^*	ω_3^*
combination 1	10	6	2	0.2	0.2	0.6
combination 2	10	4	4	0.2	0.2	0.6
combination 3	10	2	6	0.2	0.2	0.6

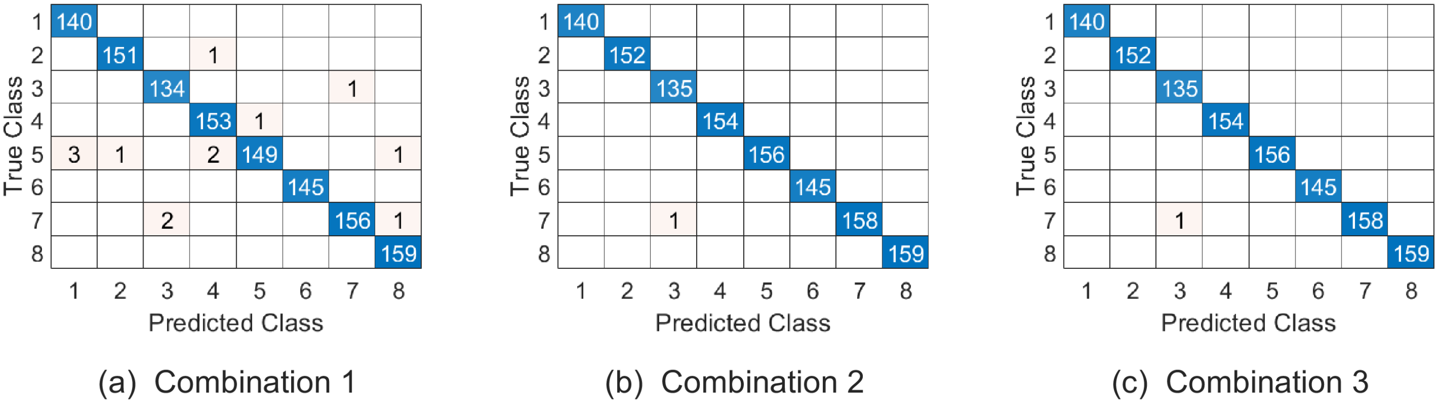
$(p_1,p_2,p_3,\omega_1,\omega_2,\omega_3)$ are listed in Table 6. And the samples on testing set are represented on $\mathcal{PG}(10,6,2|120,120,400)$, $\mathcal{PG}(10,4,4|120,120,400)$ and $\mathcal{PG}(10,2,6|120,120,400)$ respectively in experiments. Table 7 summarizes the average correct recognition rate. The results show that our algorithm has superior performance compared with some state-of-the-art methods. And the confusion matrix of our proposed approach on the testing set under the three parameter combinations are given in Fig. 7.

UMD Keck body-gesture datasets

The UMD Keck Body-Gesture Datasets contains 14 naval body gestures acquired from both static and dynamic backgrounds. The subjects and the camera remain stationary in the static backgrounds, the subjects and the camera are moving in the dynamic backgrounds. 126 videos and 168 videos are collected from the static scene and the dynamic environment

Table 7 Correct recognition rate on the Ballet dataset.

Method	CRR
(*Fathi & Mori, 2008*)	51%
DBoWs (*Iosifidis, Tefas & Pitas, 2014*)	91.1%
S-CTM (*Wang & Mori, 2009*)	91.36%
kgSC-dic (*Harandi et al., 2015*)	83.53 ± 0.8%
kgLC-dic (*Harandi et al., 2015*)	86.94 ± 1.1%
DMD+SC (SCCD2) (*Singh et al., 2021*)	96.25
WSRC-PGM (Combination 1)	98.9
WSRC-PGM (Combination 2)	99.9
WSRC-PGM (Combination 3)	99.9



(a) Combination 1

	1	2	3	4	5	6	7	8
1	140							
2		151		1				
3			134				1	
4				153	1			
5	3	1		2	149			1
6						145		
7			2				156	1
8								159

(b) Combination 2

	1	2	3	4	5	6	7	8
1	140							
2		152						
3			135					
4				154				
5					156			
6						145		
7			1				158	
8								159

(c) Combination 3

	1	2	3	4	5	6	7	8
1	140							
2		152						
3			135					
4				154				
5					156			
6						145		
7			1				158	
8								159

Figure 7 The confusion matrix of three combinations on Ballet datasets. The class labels "1-8" represent actions "left-to-right hand opening", "right-to-left hand opening", "standing hand opening", "leg swinging", "jumping", "turning" , "hopping" and "standing still" respectively.
Full-size DOI: 10.7717/peerjcs.923/fig-7

respectively. The 14 body gestures are turn left, turn right, attention left, attention right, flap, stop left, stop right, stop both, attention both, start, go back, close distance, speed up and come near respectively.

We follow the experimental setting proposed in paper (*Lin, Jiang & Davis, 2009*). In the static background, we adopt Leave One Out Cross Validation (LOOCV). For dynamic background, the gestures acquired from the static background are used for training, while the gestures in dynamic background are used for testing.

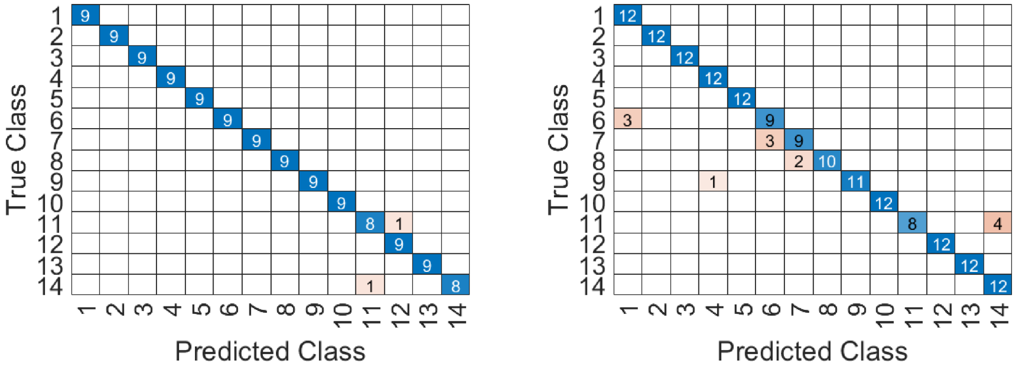
In our experiment, videos are firstly cropped by tracking the region of interest through a simple correlation filter, and then all videos are resized to $32 \times 24 \times 45$. The videos whose frames are less than 45 are appended with the last frame added some Gaussian noise. Similar to the previous discussion, we jointly determine the parameters $(p_1, p_2, p_3, \omega_1, \omega_2)$ by 5-fold cross validation on training set, where p_1 is in the range of $\{2:4:32\}$, p_2 is in the range of $\{2:4:24\}$, p_3 is in the range of $\{10:4:45\}$ and (ω_1, ω_2) are in the range as Table 2. The top 5 % optional parameter combinations of $(p_1, p_2, p_3, \omega_1, \omega_2, \omega_3)$ are listed in Table 8. And the samples on testing set are represented on $\mathcal{PG}(6, 22, 14|1080, 1440, 768)$. Table 9 shows that WSRC-PGM has higher performance compared with TB (*Lui, 2011*),

Table 8 The top 5% combinations of parameter on UMD Keck body-gesture dataset.

Parameter	p_1^*	p_2^*	p_3^*	ω_1^*	ω_2^*	ω_3^*
combination 1	6	22	14	0.2	0.4	0.4

Table 9 Correct recognition rate on the UMD Keck Body-Gesture datasets.

Method	CRR of static	CRR of dynamic
TB (Lui, 2011)	92.1%	91.1%
Prototype-Tree (Lin, Jiang & Davis, 2009)	95.2%	91.1%
PM (Lui, 2012)	94.4%	92.3%
WSRC-PGM	98.4%	92.3%



(a) Confusion matrix of static set

(b) Confusion matrix of dynamic set

Figure 8 The confusion matrix of combination 1 on UMD Keck Body-Gesture datasets.
Full-size DOI: 10.7717/peerjcs.923/fig-8

Prototype-Tree (Lin, Jiang & Davis, 2009) and PM (Lui, 2012). The confusion matrix of our proposed approach with parameter combination 1 are given in Fig. 8.

Discussion

Through above experiments, we conclude that the proposed method is effective for video-based human gesture recognition. In experiments, the selection of parameters is a key step. We jointly selected optional parameters $(p_1^*, p_2^*, p_3^*, \omega_1^*, \omega_2^*, \omega_3^*)$ on grid parameter set, through maximizing the average CRRs of 5-fold cross validation on training set. The parameter selection process is time-consuming because of the high dimension of parameter. This limitation may be solved by alternative iterations of optimization, through setting rational initial values based on prior information of data distribution. The reason is that the dimensions of parameter for each iteration can be reduced.

COMPUTATIONAL COMPLEXITY

We analyze the time complexity of WSC-PGM algorithm in this section. The algorithm focus on improving the correct recognition rate by sparse coding on product Grassmann manifold. Compared with sparse coding on single Grassmann manifold named as gSC

Table 10 **Time complexity(in seconds) for classifying testing sets on three datasets respectively.**

	Cambridge hand gesture	Ballet	UMD Keck
PGM size	$\mathcal{PG}(8,18,12\|400,400,400)$	$\mathcal{PG}(8,18,12\|120,120,400)$	$\mathcal{PG}(6,22,14\|1080,1440,768)$
Train size	180	1200	126
Test size	720	1200	168
Time	4.85	19.08	2.31

(*Harandi et al., 2015*), we discuss the computation efficiency of WSC-PGM algorithm in the following.

Same as the notations of algorithm WSC-PGM, the WSC-PGM algorithm requires $O(N(d_1p_1^2+d_2p_2^2+d_3p_3^2))$ flops for computing $K^m(\mathbf{X},\mathbf{Y})$. The gSC algorithm (*Harandi et al., 2015*) requires $O(Ndp^2)$ flops for computing $\|\mathbf{Z}^T\mathbf{D}_j\|_F^2$(j =1 ,…,$N$), where span($\mathbf{Z}$), span($\mathbf{D}_j$) $\in \mathcal{G}(p,d)$ while other steps of the two algorithms have the same computational complexity. To make it easier for the readers to understand, we take the Cambridge Hand Gesture Dataset as an example, we set $d_1=d_2=d_3=400, p_1=8, p_2=18, p_3=12$ of combination 1 in our experiment and $d=400, p=50$ are chosen in gSC (*Harandi et al., 2015*). We can see that $d_1p_1^2+d_2p_2^2+d_3p_3^2=212800 \ll dp^2=1000000$. However, the CRR of WSC-PGM algorithm is higher than that in gSC (*Harandi et al., 2015*).

We further evaluate the execution time of our WSC-PGM for classification in Table 10. And all experiments are executed on Intel(R) Core(TM) i7-10700 CPU with 32GB RAM.

MAIN FINDINGS AND FUTURE DIRECTIONS

Subject to video-based human gesture recognition, we proposed a novel weighted sparse coding model on product Grassmann manifold. A video can be viewed as a third order tensor and then represented as a point on product Grassmann manifold by factorizing the tensor through HOSVD. This representation can characterize the multi-dimensional information including appearance, horizontal motion, vertical motion from video data and also can efficiently take advantage of the nonlinear manifold structure of video data. Based on PGM representation of videos, we proposed a sparse coding method by embedding the product Grassmann manifold to the product space of symmetric matrices. Meanwhile, an efficient algorithm WSC-PGM and the corresponding classification algorithm WSRC-PGM are proposed. The method of this paper improves the correct recognition rate and meanwhile it reduces the time complexity comparing with sparse coding on single Grassmann manifold. Experiments on three kinds of public datasets show that our method performs very well.

In future work, we would like to study the product Grassmann manifold representation method combing with time series model in tensor form, in order to enhance the discriminant performance of videos.

ACKNOWLEDGEMENTS

We would like to thank two reviewers and editors for their helpful comments and suggestions. Junfei Zhang would like to thank the School of Statistics and Mathematics of Central University of Finance and Economics for supplying working space and other necessary supplies.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This work was supported by the Natural Science Foundation of China (NSFC NO. 11871488, 11971504), the disciplinary funding of Central University of Finance and Economics, the Emerging Interdisciplinary Project of CUFE. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Natural Science Foundation of China (NSFC): NO. 11871488, 11971504.
The disciplinary funding of Central University of Finance and Economics.
The Emerging Interdisciplinary Project of CUFE.

Competing Interests

The authors declare there are no competing interests.

Author Contributions

- Yuping Wang Junfei Zhang conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The code is available at GitHub: https://github.com/wangyp010/SRC_PGM.

The data is available at:

- Cambridge hand gesture database: https://labicvl.github.io/ges_db.htm
- Ballet dataset: https://www2.cs.sfu.ca/research/groups/VML/semilatent/
- UMD Keck body-gesture: http://www.zhuolin.umiacs.io/Keckgesturedataset.html.

REFERENCES

Absil P-A, Mahony R, Sepulchre R. 2009. *Optimization algorithms on matrix manifolds*. Princeton: Princeton University Press.

Baccouche M, Mamalet F, Wolf C, Garcia C, Baskurt A. 2011. Sequential deep learning for human action recognition. In: *International workshop on human behavior understanding*. Springer, 29–39.

Belhumeur PN, Hespanha JP, Kriegman DJ. 1997. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19(7)**:711–720 DOI 10.1109/34.598228.

Carreira J, Agrawal P, Fragkiadaki K, Malik J. 2016. Human pose estimation with iterative error feedback. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4733–4742.

Chakraborty BK, Sarma D, Bhuyan MK, MacDorman KF. 2018. Review of constraints on vision-based gesture recognition for human–computer interaction. *IET Computer Vision* **12(1)**:3–15 DOI 10.1049/iet-cvi.2017.0052.

Chen X, Wang G, Guo H, Zhang C, Wang H, Zhang L. 2019. Mfa-net: motion feature augmented network for dynamic hand gesture recognition from skeletal data. *Sensors* **19(2)**:239 DOI 10.3390/s19020239.

Elhamifar E, Vidal R. 2013. Sparse subspace clustering: algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **35(11)**:2765–2781 DOI 10.1109/TPAMI.2013.57.

Fathi A, Mori G. 2008. Action recognition by learning mid-level motion features. In: *2008 IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 1–8.

Feichtenhofer C, Pinz A, Wildes RP. 2016. Spatiotemporal residual networks for video action recognition. ArXiv preprint. arXiv:1611.02155.

Ge SS, Yang Y, Lee TH. 2008. Hand gesture recognition and tracking based on distributed locally linear embedding. *Image and Vision Computing* **26(12)**:1607–1620 DOI 10.1016/j.imavis.2008.03.004.

Harandi M, Hartley R, Shen C, Lovell B, Sanderson C. 2015. Extrinsic methods for coding and dictionary learning on Grassmann manifolds. *International Journal of Computer Vision* **114(2)**:113–136 DOI 10.1007/s11263-015-0833-x.

Huang Z, Wang R, Shan S, Chen X. 2015. Projection metric learning on Grassmann manifold with application to video based face recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 140–149.

Iosifidis A, Tefas A, Pitas I. 2014. Discriminant bag of words based representation for human action recognition. *Pattern Recognition Letters* **49**:185–192 DOI 10.1016/j.patrec.2014.07.011.

Ji S, Xu W, Yang M, Yu K. 2012. 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **35(1)**:221–231.

Jung P-G, Lim G, Kim S, Kong K. 2015. A wearable gesture recognition device for detecting muscular activities based on air-pressure sensors. *IEEE Transactions on Industrial Informatics* **11(2)**:485–494.

Kim T-K, Cipolla R. 2008. Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31(8)**:1415–1428.

Klaser A, Marszałek M, Schmid C. 2008. A spatio-temporal descriptor based on 3d-gradients. In: *BMVC 2008-19th British machine vision conference*. British Machine Vision Association, 275–271.

Kolda TG, Bader BW. 2009. Tensor decompositions and applications. *SIAM Review* **51(3)**:455–500 DOI 10.1137/07070111X.

Laptev I. 2005. On space-time interest points. *International Journal of Computer Vision* **64(2)**:107–123 DOI 10.1007/s11263-005-1838-7.

Le QV, Zou WY, Yeung SY, Ng AY. 2011. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In: *CVPR 2011*. Piscataway: IEEE, 3361–3368.

Li N, Sun B, Yu J. 2015. A weighted sparse coding framework for saliency detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 5216–5223.

Lin Z, Jiang Z, Davis LS. 2009. Recognizing actions by shape-motion prototype trees. In: *2009 IEEE 12th international conference on computer vision*. Piscataway: IEEE, 444–451.

Lui YM. 2011. Tangent bundles on special manifolds for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology* **22(6)**:930–942.

Lui YM. 2012. Human gesture recognition on product manifolds. *The Journal of Machine Learning Research* **13(1)**:3297–3321.

Luo W. 2011. Face recognition based on laplacian eigenmaps. In: *2011 International conference on computer science and service system (CSSS)*. Piscataway: IEEE, 416–419.

Mohammadzade H, Sayyafan A, Ghojogh B. 2018. Pixel-level alignment of facial images for high accuracy recognition using ensemble of patches. *JOSA A* **35(7)**:1149–1159.

Pareek P, Thakkar A. 2021. A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. *Artificial Intelligence Review* **54(3)**:2259–2322 DOI 10.1007/s10462-020-09904-8.

Patil AR, Subbaraman S. 2019. A spatiotemporal approach for vision-based hand gesture recognition using Hough transform and neural network. *Signal, Image and Video Processing* **13(2)**:413–421 DOI 10.1007/s11760-018-1370-1.

Paul M, Haque SM, Chakraborty S. 2013. Human detection in surveillance videos and its applications-a review. *EURASIP Journal on Advances in Signal Processing* **2013(1)**:1–16 DOI 10.1186/1687-6180-2013-1.

Pless R. 2003. Image spaces and video trajectories: using isomap to explore video sequences. In: *ICCV, vol. 3*. 1433–1440.

Rahimi S, Aghagolzadeh A, Ezoji M. 2019. Human action recognition based on the Grassmann multi-graph embedding. *Signal, Image and Video Processing* **13(2)**:271–279 DOI 10.1007/s11760-018-1354-1.

Sheng B, Li J, Xiaoc F, Li Q, Yang W, Han J. 2019. Discriminative multi-view subspace feature learning for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology* **30(12)**:4591–4600.

Singh K, Dhiman C, Vishwakarma DK, Makhija H, Walia GS. 2021. Sparse coded composite descriptor for human activity recognition. *Expert Systems* **39(1)**:e12805.

Souza LS, Gatto BB, Xue J-H, Fukui K. 2020a. Enhanced Grassmann discriminant analysis with randomized time warping for motion recognition. *Pattern Recognition* **97**:107028 DOI 10.1016/j.patcog.2019.107028.

Spurr A, Song J, Park S, Hilliges O. 2018. Cross-modal deep variational hand pose estimation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 89–98.

Tran D, Bourdev L, Fergus R, Torresani L, Paluri M. 2015. Learning spatiotemporal features with 3d convolutional networks. In: *Proceedings of the IEEE international conference on computer vision*. Piscataway: IEEE, 4489–4497.

Turaga P, Chellappa R. 2009. Locally time-invariant models of human activities using trajectories on the grassmannian. In: *2009 IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 2435–2441.

Verma B, Choudhary A. 2018. Framework for dynamic hand gesture recognition using Grassmann manifold for intelligent vehicles. *IET Intelligent Transport Systems* **12(7)**:721–729 DOI 10.1049/iet-its.2017.0331.

Verma B, Choudhary A. 2020. Grassmann manifold based dynamic hand gesture recognition using depth data. *Multimedia Tools and Applications* **79(3)**:2213–2237 DOI 10.1007/s11042-019-08266-w.

Vishwakarma DK, Kapoor R. 2015. Hybrid classifier based human activity recognition using the silhouette and cells. *Expert Systems with Applications* **42(20)**:6957–6965 DOI 10.1016/j.eswa.2015.04.039.

Wang B, Hu Y, Gao J, Sun Y, Ju F, Yin B. 2020. Learning adaptive neighborhood graph on Grassmann manifolds for video/image-set subspace clustering. *IEEE Transactions on Multimedia* **23**:216–227.

Wang B, Hu Y, Gao J, Sun Y, Yin B. 2016. Product Grassmann manifold representation and its LRR models. In: *Thirtieth AAAI conference on artificial intelligence.*.

Wang Y, Mori G. 2009. Human action recognition by semilatent topic models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31(10)**:1762–1774 DOI 10.1109/TPAMI.2009.43.

Wang Y, Wang L, Kong D, Yin B. 2018. Extrinsic least squares regression with closed-form solution on product Grassmann manifold for video-based recognition. *Mathematical Problems in Engineering* **2018**:6598025.

Wang Y, Zhang J. 2020. Reconstruction of compressively sampled light field by using tensor dictionaries. *Multimedia Tools and Applications* **79(27)**:20449–20460 DOI 10.1007/s11042-020-08903-9.

Wold S, Esbensen K, Geladi P. 1987. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems* **2(1–3)**:37–52 DOI 10.1016/0169-7439(87)80084-9.

Wright J, Ma Y, Mairal J, Sapiro G, Huang TS, Yan S. 2010. Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE* **98(6)**:1031–1044 DOI 10.1109/JPROC.2010.2044470.

Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y. 2008. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31(2)**:210–227.

Xu J, Zhang L, Zhang D. 2018. A trilateral weighted sparse coding scheme for real-world image denoising. In: *Proceedings of the European conference on computer vision (ECCV)*. 20–36.

Yan J, Tong M. 2011. Weighted sparse coding residual minimization for visual tracking. In: *2011 visual communications and image processing (VCIP)*. Piscataway: IEEE, 1–4.

Zhu F, Shao L, Xie J, Fang Y. 2016. From handcrafted to learned representations for human action recognition: a survey. *Image and Vision Computing* **55**:42–52 DOI 10.1016/j.imavis.2016.06.007.