Research Article

Extrinsic Least Squares Regression with Closed-Form Solution on Product Grassmann Manifold for Video-Based Recognition

Yuping Wang,¹ Lichun Wang¹,¹ Dehui Kong¹,¹ and Baocai Yin^{1,2}

¹Beijing Key Lab of Multimedia and Intelligent Software Technology, Faculty of Information Technology, Beijing University of Technology, 100 Pingleyuan, Chaoyang District, Beijing 100124, China

²School of Software Technology, Dalian University of Technology, No. 2 Linggong Road, Ganjingzi District, Dalian 116024, China

Correspondence should be addressed to Lichun Wang; wanglc@bjut.edu.cn

Received 21 August 2017; Accepted 30 January 2018; Published 1 March 2018

Academic Editor: Simone Bianco

Copyright © 2018 Yuping Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Least squares regression is a fundamental tool in statistical analysis and is more effective than some complicated models with small number of training samples. Representing multidimensional data with product Grassmann manifold has recently led to notable results in various visual recognition tasks. This paper proposes extrinsic least squares regression with Projection Metric on product Grassmann manifold by embedding Grassmann manifold into the space of symmetric matrices via an isometric mapping. The proposed regression has closed-form solution which is more accurate compared with numerical solution of previous least squares regression using geodesic distance. Experiments on several recognition tasks show that the proposed method achieves considerable accuracy in comparison with some state-of-the-art methods.

1. Introduction

As an important application of computer vision, video-based recognition such as action recognition [1] attracts more and more attention. For inferring the correct label of a query in a given database of examples, there are mainly two kinds of methods. One kind approach is based on representations with the handcrafted features and the other kind is based on deep learning architectures such as Convolutional Neural Networks (CNN) [2]. Generally speaking, deep learning algorithms have been shown to be successful when large amount of data is available [3, 4]. However, the size of database for many recognition tasks in daily life is small. In this case, deep learning algorithms lose efficacy and it becomes important to analyze the structure of data and represent it with discriminant features.

Nowadays, Grassmann manifold has proven a powerful representation for video-based applications like activity classification [5], action recognition [6], age estimation [7], face recognition [8, 9], and so on. In the above applications, Grassmann manifold is used to characterize the intrinsic geometry of data. Taking one representative work as an

example, Lui [10] factorized a data tensor using Higher Order Singular Value Decomposition (HOSVD) and imposed each factorized element on a Grassmann manifold. This representation yields a very discriminating structure for action recognition.

Inference on manifold spaces can be achieved extrinsically by embedding manifold into Euclidean space, which can be considered as flattening the manifold. In the literature, the most popular choice for embedding manifold is through considering tangent spaces [11, 12]. For example, Lui [10] presented a least squares regression on product Grassmann manifold, in which the weighted average from the training samples was computed in tangent space and was projected back to Grassmann manifold by standard logarithmic and exponential map. The distance between points to the tangent pole is equal to geodesic distance, which is restrictive and may lead to inaccurate modeling. An alternate method considers embedding Grassmann manifold into space of symmetric matrices by a diffeomorphism [13] and uses Projection Metric [14] which is equal to the true Grassmann geodesic distance up to a scale of $\sqrt{2}$.

In this paper, by representing multidimensional data on product Grassmann manifold with same form as Lui [10], we propose an extrinsic least squares regression on product Grassmann manifold using Projection Metric and give a closed-form solution which is more accurate. Least squares regression as a simple statistical model has many advantages such as simple calculation and being more effective than some complicated models with small number of training samples [15]. We experiment with the proposed method on three kinds of small-scale datasets including hand gesture, Ballet, and traffic; the higher recognition rates reveal that our method is competitive to some state-of-the-art methods.

The rest of this paper is organized as follows: Section 2 introduces mathematical background; Section 3 gives product Grassmann manifold representation for video; Section 4 presents distance on product Grassmann manifold; Section 5 proposes extrinsic least squares regression on product Grassmann manifold; Section 6 gives classification based on extrinsic least squares regression; Section 7 shows experiments on different datasets, and experiment results show that the proposed method achieves considerable accuracy; Section 8 analyzes the time complexity of proposed method and Section 9 gives a conclusion.

2. Mathematical Background

In this section, we introduce the mathematical background used in this paper.

2.1. Grassmann Manifold. Stiefel manifold $\mathcal{S}(p, d)$ is the set of all $d \times p$ matrices with orthonormal columns; that is,

$$\mathcal{S}(p, d) \triangleq \{\mathbf{X} \in \mathbb{R}^{d \times p} : \mathbf{X}^T \mathbf{X} = \mathbf{I}_p\}, \quad (1)$$

where $\mathbf{I}_p$ is the $p \times p$ identity matrix. Grassmann manifold $\mathcal{G}(p, d)$ can be defined as a quotient manifold of $\mathcal{S}(p, d)$ with an equivalence relation $\sim$. In fact, for any $\mathbf{X}, \mathbf{Y} \in \mathcal{S}(p, d)$,

$$\mathbf{X} \sim \mathbf{Y} \iff \text{span}(\mathbf{X}) = \text{span}(\mathbf{Y}), \quad (2)$$

where $\text{span}(\mathbf{X})$ is the subspace spanned by columns of $\mathbf{X} \in \mathcal{S}(p, d)$. In other words, Grassmann manifold $\mathcal{G}(p, d)$ is the space of p -dimensional linear subspaces of $\mathbb{R}^d$ for $0 < p < d$ [16], which may be specified by arbitrary orthogonal matrix with dimension $d \times p$. Notice it is not unique for the choice of matrix $\mathbf{X}$ for a point $\text{span}(\mathbf{X})$ on Grassmann manifold; that is, the same point on Grassmann manifold can be spanned by different matrix $\mathbf{X}$ and $\mathbf{Y}$.

2.2. Higher Order Singular Value Decomposition (HOSVD). HOSVD is a multilinear SVD operating on tensor. Let $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_M}$ be a tensor with order M . The process of reordering the elements of an M -mode tensor into a matrix is called matricization. The mode- k ($k = 1, \dots, M$) matricization of a tensor $\mathcal{A}$ is denoted by $\mathbf{A}_{(k)}$ (see details in [17]). Then each $\mathbf{A}_{(k)}$ is factored using SVD as follows:

$$\mathbf{A}_{(k)} = \mathbf{U}^{(k)} \mathbf{\Sigma}^{(k)} \mathbf{V}^{(k)T}, \quad (3)$$

where $\mathbf{\Sigma}^{(k)}$ is a diagonal matrix, $\mathbf{U}^{(k)}$ is an orthogonal matrix which spanned the column space of $\mathbf{A}_{(k)}$, and $\mathbf{V}^{(k)}$ is an orthogonal matrix which spanned the row space of $\mathbf{A}_{(k)}$. By using HOSVD method, an M order tensor $\mathcal{A}$ can be decomposed as follows:

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \dots \times_M \mathbf{U}^{(M)}, \quad (4)$$

where $\mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_M}$ is core tensor, $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(M)}$ are orthogonal matrices given in (3), and $\times_k$ denotes mode- k multiplication.

2.3. Product Manifold. Let $\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_M$ be manifolds; the product manifold of the manifolds $\mathcal{U}_1, \mathcal{U}_2, \dots, \mathcal{U}_M$ is defined as

$$\begin{aligned} \mathcal{U} = \mathcal{U}_1 \times \mathcal{U}_2 \times \dots \times \mathcal{U}_M = \{(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M) : \mathbf{x}_1 \\ \in \mathcal{U}_1, \mathbf{x}_2 \in \mathcal{U}_2, \dots, \mathbf{x}_M \in \mathcal{U}_M\}, \end{aligned} \quad (5)$$

where $\times$ denotes Cartesian product and $\mathcal{U}_i$ ($i = 1, 2, \dots, M$) is called factor manifold.

3. Product Grassmann Manifold Representation for Video

Video is a kind of multidimensional data and 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. The variation of each mode can be captured by HOSVD. Lui et al. [18] found that traditional HOSVD is not appropriate for forming product manifold, so they redefined the traditional definition of HOSVD to factorize tensor using the orthogonal matrices $\mathbf{V}^{(1)}, \mathbf{V}^{(2)}$, and $\mathbf{V}^{(3)}$ described in (3). That is,

$$\mathcal{A} = \widehat{\mathcal{S}} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \mathbf{V}^{(3)}, \quad (6)$$

where $\widehat{\mathcal{S}} \in \mathbb{R}^{(J_2 \times J_3) \times (J_1 \times J_3) \times (J_1 \times J_2)}$ is core tensor.

Since $\mathbf{V}^{(k)}$ ($k = 1, 2, 3$) is a tall orthogonal matrix, hence it is a point on Stiefel manifold. Then $\text{span}(\mathbf{V}^{(k)})$ is a point on Grassmann manifold. Hence, $(\text{span}(\mathbf{V}^{(1)}), \text{span}(\mathbf{V}^{(2)}), \text{span}(\mathbf{V}^{(3)}))$ is a point on product Grassmann manifold. Then $(\text{span}(\mathbf{V}^{(1)}), \text{span}(\mathbf{V}^{(2)}), \text{span}(\mathbf{V}^{(3)}))$ is a representation for videos on product Grassmann manifold.

4. Distance on Product Grassmann Manifold

The metric on Grassmann manifold is geodesic distance which is the shortest curve between two p -dimensional subspaces $\mathbf{X}$ and $\mathbf{Y}$, that is, $d_g(\mathbf{X}, \mathbf{Y}) = \sqrt{\sum_{i=1}^p \sin^2 \theta_i}$ with θ_i representing the principal angles [16]. Recently, Chikuse [13] introduced a projection embedding $\Pi : \mathcal{G}(p, d) \rightarrow \text{sym}(d)$, $\Pi(\mathbf{X}) = \mathbf{X}\mathbf{X}^T$, where $\text{sym}(d)$ denotes space of symmetric matrices. And Hamm and Lee [19] defined a distance called *Projection Metric* on Grassmann manifold as follows.

Definition 1. Given two points $\mathbf{X}$ and $\mathbf{Y}$ on Grassmann manifold $\mathcal{G}(p, d)$, the distance between $\mathbf{X}$ and $\mathbf{Y}$ is defined as

$$d_s(\mathbf{X}, \mathbf{Y}) = \frac{\sqrt{2}}{2} \|\mathbf{X}\mathbf{X}^T - \mathbf{Y}\mathbf{Y}^T\|_F. \quad (7)$$

Remark 2. In fact, for any matrix $\mathbf{X} \neq \mathbf{Y}$, there exists a $p \times p$ orthogonal matrix $\mathbf{Q}_p$ such that $\mathbf{X} = \mathbf{Y}\mathbf{Q}_p$, then element $\text{span}(\mathbf{X}) \in \mathcal{G}(p, d)$ is equal to element $\text{span}(\mathbf{Y}) \in \mathcal{G}(p, d)$. In this case, $d_s(\mathbf{X}, \mathbf{Y}) = (\sqrt{2}/2)\|\mathbf{Y}\mathbf{Q}_p\mathbf{Q}_p^T\mathbf{Y}^T - \mathbf{Y}\mathbf{Y}^T\|_F = 0$. Hence it is feasible to use the matrix $\mathbf{X}$ representing $\text{span}(\mathbf{X})$. And $d_s(\mathbf{X}, \mathbf{Y})$ is equal to geodesic distance of two points on Grassmann manifold [14].

Based on Definition 1, we give a kind of definition of distance on product Grassmann manifold which sums distance of each factor Grassmann manifold.

Definition 3. Given two points $(\mathbf{X}_1, \mathbf{Y}_1)$ and $(\mathbf{X}_2, \mathbf{Y}_2)$ on product Grassmann manifold $\mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2)$, the distance between $(\mathbf{X}_1, \mathbf{Y}_1)$ and $(\mathbf{X}_2, \mathbf{Y}_2)$ is defined as

$$D_s((\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2)) = d_s(\mathbf{X}_1, \mathbf{X}_2) + d_s(\mathbf{Y}_1, \mathbf{Y}_2). \quad (8)$$

5. Extrinsic Least Squares Regression on Product Grassmann Manifold

Least squares regression is a simple and efficient technique in statistical analysis. In Euclidean space, parameter $\boldsymbol{\beta} \in \mathbb{R}^{n \times 1}$ is estimated by minimizing the residual sum-of-square error

$$R(\boldsymbol{\beta}) = \|\mathbf{y} - \mathbf{A}\boldsymbol{\beta}\|^2, \quad (9)$$

where $\mathbf{A} \in \mathbb{R}^{n \times k}$ is training set and $\mathbf{y} \in \mathbb{R}^{n \times 1}$ is regression value. The estimated parameter has closed solution as

$$\hat{\boldsymbol{\beta}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}. \quad (10)$$

Hence the corresponding error is

$$\|\mathbf{y} - \mathbf{A}(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}\|^2. \quad (11)$$

In Grassmann manifold space, Lui [10] extended the linear least squares regression to a nonlinear form. In detail, the estimated parameter is equal to

$$(\mathbf{A} \star \mathbf{A})^{-1} (\mathbf{A} \star \mathbf{y}), \quad (12)$$

where $\star$ is a nonlinear similarity operator, $\mathbf{A}$ is a set of training samples on manifold, and $\mathbf{y}$ is an element on manifold. So the corresponding error is

$$d_g(\mathbf{y}, \mathbf{A} \circ (\mathbf{A} \star \mathbf{A})^{-1} (\mathbf{A} \star \mathbf{y})), \quad (13)$$

where $\circ$ is an operator mapping points from vector space back to manifold. While Grassmann manifold is not closed under normal matrix subtraction and addition, the mapping $\circ$ is realized by employing exponential mapping and its inverse

without closed-form solution. To realize the composition map $\circ$, an improved Karcher Mean Computation algorithm is employed. To avoid loss of the above iterative algorithm, we introduce an extrinsic least squares regression on Grassmann manifold by embedding its elements to space of symmetric matrices. Due to the distance on product Grassmann manifold in (8) being additive for each factor, the extrinsic least squares regression on product Grassmann manifold equals three independent subregression problems on each factor. Taking one factor as example, we show the details in the following.

Let $\{\mathbf{D}_j \in \mathcal{G}(p, d)\}_{j=1,2,\dots,N}$ be training set where N is number of samples, and $\mathbf{y} = (\mathbf{y}_j) \in \mathbb{R}^{N \times 1}$ is fitting parameter. $\mathbf{X} \in \mathcal{G}(p, d)$ is regression value. Similar to the idea of least squares regression in Euclidean space, we give a regression on Grassmann manifold, which is defined in the embedded space of symmetric matrices. The residual is measured as follows:

$$R(\mathbf{y}) \triangleq \min_{\mathbf{y}} \left\| \mathbf{X}\mathbf{X}^T - \sum_{j=1}^N \mathbf{D}_j \mathbf{D}_j^T \mathbf{y}_j \right\|_F^2, \quad (14)$$

where $\mathbf{y}_j$ is the j th element in vector $\mathbf{y}$. Next we show how to solve the optimization. We have

$$\begin{aligned} & \left\| \mathbf{X}\mathbf{X}^T - \sum_{j=1}^N \mathbf{D}_j \mathbf{D}_j^T \mathbf{y}_j \right\|_F^2 \\ &= \text{Tr}(\mathbf{X}^T \mathbf{X} \mathbf{X}^T \mathbf{X}) + \sum_{i,j=1}^N \mathbf{y}_i \mathbf{y}_j \text{Tr}(\mathbf{D}_i^T \mathbf{D}_j \mathbf{D}_j^T \mathbf{D}_i) \\ & \quad - 2 \sum_{j=1}^N \mathbf{y}_j \text{Tr}(\mathbf{D}_j^T \mathbf{X} \mathbf{X}^T \mathbf{D}_j) \end{aligned} \quad (15)$$

and we define

$$\begin{aligned} [K(\mathbf{D})]_{i,j} &= \text{Tr}(\mathbf{D}_i^T \mathbf{D}_j \mathbf{D}_j^T \mathbf{D}_i) = \|\mathbf{D}_i^T \mathbf{D}_j\|_F^2, \\ [K(\mathbf{X}, \mathbf{D})]_j &= \text{Tr}(\mathbf{D}_j^T \mathbf{X} \mathbf{X}^T \mathbf{D}_j) = \|\mathbf{X}^T \mathbf{D}_j\|_F^2. \end{aligned} \quad (16)$$

Hence model (14) becomes

$$\min_{\mathbf{y}} \{ \mathbf{y}^T K(\mathbf{D}) \mathbf{y} - 2 \mathbf{y}^T K(\mathbf{X}, \mathbf{D}) \}. \quad (17)$$

Let derivation of (17) with respect to $\mathbf{y}$ equal to 0; we have

$$(\mathbf{K}(\mathbf{D}) + \mathbf{K}(\mathbf{D})^T) \mathbf{y} - 2 \mathbf{K}(\mathbf{X}, \mathbf{D}) = 0. \quad (18)$$

So the solution of optimization (14) is

$$\mathbf{y}^* = 2 (\mathbf{K}(\mathbf{D}) + \mathbf{K}(\mathbf{D})^T)^{-1} \mathbf{K}(\mathbf{X}, \mathbf{D}). \quad (19)$$

Hence the corresponding error becomes

$$\left\| \mathbf{X}\mathbf{X}^T - \sum_{j=1}^N \mathbf{D}_j \mathbf{D}_j^T \mathbf{y}_j^* \right\|_F^2. \quad (20)$$

FIGURE 1: Cambridge hand gesture samples: (a) from left to right (flat-leftward; flat-rightward; flat-contract; spread-leftward; spread-rightward); (b) from left to right (spread-contract; V-shape-leftward; V-shape-rightward; V-shape-contract).

6. Recognition Based on Extrinsic Least Squares Regression

In this subsection, we consider 3-order product Grassmann manifold

$$\mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2) \times \mathcal{G}(p_3, d_3) \quad (21)$$

for videos, while the situation for higher order is similar. Suppose M classes are defined for the data. We denote training set corresponding with the k th class as $\{(\mathbf{U}_j^k, \mathbf{V}_j^k, \mathbf{W}_j^k)\}_{j=1,2,\dots,N_k}$, where N_k is number of samples. Our objective is inferring to which class the test sample $(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \in \mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2) \times \mathcal{G}(p_3, d_3)$ belongs.

The residual error of query sample $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ for class k is defined as

$$\begin{aligned} \varepsilon^k = & \left\| \mathbf{X}\mathbf{X}^T - \sum_{j=1}^{N_k} \mathbf{U}_j^k \mathbf{U}_j^{kT} \boldsymbol{\alpha}_j^{k*} \right\|_F \\ & + \left\| \mathbf{Y}\mathbf{Y}^T - \sum_{j=1}^{N_k} \mathbf{V}_j^k \mathbf{V}_j^{kT} \boldsymbol{\beta}_j^{k*} \right\|_F \\ & + \left\| \mathbf{Z}\mathbf{Z}^T - \sum_{j=1}^{N_k} \mathbf{W}_j^k \mathbf{W}_j^{kT} \boldsymbol{\gamma}_j^{k*} \right\|_F, \end{aligned} \quad (22)$$

where $\boldsymbol{\alpha}_j^{k*}, \boldsymbol{\beta}_j^{k*}, \boldsymbol{\gamma}_j^{k*}$ ($j = 1, \dots, N_k$) are solutions of subregression on each factor Grassmann manifold, respectively. The category of the query sample is determined by

$$k^* = \arg \min_k \varepsilon^k. \quad (23)$$

7. Experiments on Different Datasets

In this section, we show performance of the proposed method against some state-of-the-art methods on two kinds of datasets.

7.1. Action Recognition

7.1.1. Cambridge Hand Gesture Dataset. The Cambridge hand gesture dataset [20] contains 900 video sequences with nine kinds of hand gestures, which is divided into 5 sets according to different illuminations. Figure 1 shows some hand gesture samples. Set 5 (normal illumination) is considered for training while the remaining sequences (with different illumination characteristics) are used for testing. The original sequences are converted to grayscale and resized to $24 \times 32 \times 23$. We denote our method as ELSR and report the correct recognition rate (CRR) for the four illumination sets in Table 1. Compared with product manifold (PM) [10], Grassmann Sparse Coding (gSC) [14], Grassmann Locality-Constrained Coding (gLC) [14], kernel Grassmann Sparse Coding (kgSC) [14], and kernel Grassmann Locality-Constrained Coding (kgLC) [14], we find that our method is competitive to these state-of-the-art methods.

7.1.2. Ballet Dataset. 44 videos are collected from a Ballet instruction DVD as the Ballet dataset [21]. In fact, 8 complex motion patterns from 3 persons are included in the dataset. In detail, the actions are “right-to-left hand opening,” “left-to-right hand opening,” “standing hand opening,” “jumping,” “leg swinging,” “hopping,” “turning,” and “standing still”. The main challenge of this dataset is large variations among classes such as speed, clothing, and motion paths. Figure 2 shows some examples of the dataset. Table 2 shows ELSR has superior performance compared with gSC-dic, gLC-dic, kgSC-dic, and kgLC-dic [14].

7.2. Scene Analysis. For scene analysis, we use the UCSD traffic dataset [22] which contains 254 videos of highway traffic under different weather conditions. Resolution is 320×240 and number of frames ranges from 42 to 52. The dataset is divided into three classes (“heavy,” “medium,” and “light”) according to traffic congestion level. In total, there are 44 sequences defined as heavy traffic, 45 sequences labeled as medium traffic, and 165 sequences are light traffic. Figure 3

FIGURE 2: Examples from the Ballet dataset.

TABLE 1: Recognition results on the Cambridge hand gesture dataset.

Method	Set 1	Set 2	Set 3	Set 4	Overall
PM [10]	93	89	91	94	$91.7 \pm 2.3\%$
gSC [14]	93	92	93	94	$93.3 \pm 0.9\%$
gLC [14]	96	94	96	97	$95.4 \pm 1.3\%$
kgSC [14]	96	92	93	97	$94.4 \pm 2.0\%$
kgLC [14]	96	94	96	98	$95.7 \pm 1.6\%$
ELSR	98	94	96	96	$95.7 \pm 1.6\%$

TABLE 2: Correct recognition rate on the Ballet dataset.

Method	CRR
gSC-dic [14]	$79.64 \pm 1.1\%$
gLC-dic [14]	$81.42 \pm 0.8\%$
kgSC-dic [14]	$83.53 \pm 0.8\%$
kgLC-dic [14]	$86.94 \pm 1.1\%$
ELSR	$96.31 \pm 1.7\%$

shows some typical examples. In experiment, we use the first 40 frames of each video and they are normalized as grayscale with resolution 48×48 . We adopt the four pairs of training and testing sets provided in paper [23]. The classification results are shown in Table 3; the average correct recognition rate of ELSR is higher than that of gSC and gLC but lower than kgSC and kgLC.

7.3. Discussion. Through above experiments, we conclude that the proposed method is more effective for action recognition than scene analysis. In fact, the product Grassmann manifold could capture the appearance, horizontal motion, and vertical motion through three factor manifolds. To visualize the product manifold representation, the overlay appearance, horizontal motion, and vertical motion of examples from three dataset are given in Figure 4. Note that there are obvious variation features along horizontal motion for hand gesture examples, both horizontal and vertical motion for Ballet examples. These curves in last two columns characterize the motion and are the key factors for recognizing. This can be seen as an explanation of the higher CRR result of

ELSR on Ballet dataset. Meanwhile, for samples from UCSD, horizontal and vertical motion features are not clear because of all cars running along the same path, and the critical factor is appearance, characterizing the number of cars. Hence for UCSD dataset, the CRR of ELSR is just little higher than gSC and gLC, but lower than kgSC and kgLC which maps to higher-dimensional manifolds using kernel function to diminish nonlinearity.

8. Performance Analysis

We analyze time complexity of inferring the label for a query $(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \in \mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2) \times \mathcal{G}(p_3, d_3)$ with given N training samples $\{(\mathbf{U}_j, \mathbf{V}_j, \mathbf{W}_j)\}_{j=1}^N \in \mathcal{G}(p_1, d_1) \times \mathcal{G}(p_2, d_2) \times \mathcal{G}(p_3, d_3)$. The main computing steps include (19) and (22). We take one factor manifold for example. Some terms only related to $\{\mathbf{U}_j\}_{j=1}^N$ such as $K(\mathbf{U})$ can be computed offline. For computing y^* , the time complexity of computing one element of $K(\mathbf{X}, \mathbf{U})$ is $O(p_1^2 d_1) + O(d_1 p_1) + O(p_1^2) = O(p_1^2 d_1)$; then the complexity of vector $K(\mathbf{X}, \mathbf{U})$ is $NO(p_1^2 d_1)$, and hence the complexity of solving solution $\mathbf{y}^*$ is $O(N^2) + NO(p_1^2 d_1)$. Computing the error ϵ needs $O(d_1 p_1) + O(d_1^2 p_1) + NO(d_1^2) + O(d_1^2) = NO(d_1^2 p_1)$. Therefore the whole time complexity of our approach is $\sum_{i=1}^3 (O(N^2) + NO(p_i^2 d_i) + NO(d_i^2 p_i))$. For small-scale dataset, our proposed method is effective and the time complexity will not be too large. For example in experiment of Cambridge hand gesture dataset: $N = 180$, $d_1 = 32 \times 23 = 736$, $d_2 = 23 \times 24 = 552$, $d_3 = 24 \times 32 = 768$, $p_1 = 24$, $p_2 = 32$, and $p_3 = 23$.

TABLE 3: Average correct recognition rate on the UCSD video traffic dataset.

Method	Exp 1	Exp 2	Exp 3	Exp 4	Overall
gSC [14]	93.7	87.5	95.3	95.2	$92.9 \pm 3.7\%$
gLC [14]	96.8	85.9	92.2	93.7	$92.2 \pm 4.5\%$
kgSC [14]	96.8	89.1	95.3	98.4	$94.9 \pm 4.1\%$
kgLC [14]	95.2	92.2	96.9	96.8	$95.3 \pm 2.2\%$
ELSR	92.1	92.2	92.2	96.8	$93.3 \pm 2.3\%$

FIGURE 3: Examples from the UCSD dataset: “light”; “medium”; “heavy.”

Dataset	Class	Frames from videos	Appearance	Vertical	Horizontal
Cambridge hand gesture	1				
	9				
Ballet	1				
	7				
UCSD	Heavy				
	Medium				

FIGURE 4: Examples of the overlay appearance, vertical and horizontal motion on three datasets.

9. Conclusion

In this paper, we propose extrinsic least squares regression on product Grassmann manifold. Video can be viewed as third order tensor and then transformed to point on product Grassmann manifold factorized through HOSVD. One advantage of this method is the regression has closed-form solution which guides to a more accurate ratio of correct recognition. And when number of training samples is small, the proposed method is efficient. Several experiments on different recognition tasks (hand gesture recognition, action recognition, and scene analysis) show that our method performs very well on three small-scale public datasets.

In future work, we would like to devise kernel version of extrinsic least squares regression on product manifold.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (nos. 61390510, 61632006, and 61772049), the Beijing Natural Science Foundation (no. 4162009), Funding Project for Academic Human Resources Development in Institutions of Higher Learning under the Jurisdiction of Beijing Municipality and Jing-Hua Talents Project of Beijing University of Technology, and Funding Project of Beijing Municipal Human Resources and Social Security Bureau (no. 2017-ZZ-031).

References

- [1] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and Vision Computing*, vol. 60, pp. 4–21, 2017.
- [2] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, 2017.
- [3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 770–778, Las Vegas, Nev, USA, June 2016.
- [4] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Proceedings of the 29th Annual Conference on Neural Information Processing Systems, NIPS 2015*, pp. 2377–2385, can, December 2015.
- [5] P. Turaga and R. Chellappa, "Locally time-invariant models of human activities using trajectories on the grassmannian," in *Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, pp. 2435–2441, usa, June 2009.
- [6] Y. M. Lui, "Tangent bundles on special manifolds for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 6, pp. 930–942, 2012.
- [7] P. Turaga, S. Biswas, and R. Chellappa, "The role of geometry in age estimation," in *Proceedings of the 2010 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010*, pp. 946–949, USA, March 2010.
- [8] Y. M. Lui and J. R. Beveridge, "Grassmann Registration Manifolds for Face Recognition," in *Computer Vision – ECCV 2008*, vol. 5303 of *Lecture Notes in Computer Science*, pp. 44–57, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [9] Z. Huang, R. Wang, S. Shan, and X. Chen, "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, CVPR 2015*, pp. 140–149, USA, June 2015.
- [10] Y. M. Lui, "Human gesture recognition on product manifolds," *Journal of Machine Learning Research (JMLR)*, vol. 13, pp. 3297–3321, 2012.
- [11] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on Riemannian manifolds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1713–1727, 2008.
- [12] M. Faraki, M. T. Harandi, A. Wiliem, and B. C. Lovell, "Fisher tensors for classifying human epithelial cells," *Pattern Recognition*, vol. 47, no. 7, pp. 2348–2359, 2014.
- [13] Y. Chikuse, *Statistics on special manifolds*, vol. 174 of *Lecture Notes in Statistics*, Springer-Verlag, New York, 2003.
- [14] M. Harandi, R. Hartley, C. Shen, B. Lovell, and C. Sander-son, "Extrinsic methods for coding and dictionary learning on Grassmann manifolds," *International Journal of Computer Vision*, vol. 114, no. 2-3, pp. 113–136, 2015.
- [15] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, New York, NY, USA, 2001.
- [16] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*, Princeton University Press, Princeton, NJ, 2008.
- [17] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [18] Y. M. Lui, J. R. Beveridge, and M. Kirby, "Action classification on product manifolds," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 833–839, San Francisco, Calif, USA, June 2010.
- [19] J. Hamm and D. D. Lee, "Grassmann discriminant analysis," in *Proceedings of the the 25th international conference*, pp. 376–383, Helsinki, Finland, July 2008.
- [20] T.-K. Kim and R. Cipolla, "Canonical correlation analysis of video volume tensors for action categorization and detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 8, pp. 1415–1428, 2009.
- [21] Y. Wang and G. Mori, "Human action recognition by semilattent topic models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 10, pp. 1762–1774, 2009.
- [22] A. B. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, pp. 846–851, USA, June 2005.
- [23] A. B. Chan and N. Vasconcelos, "Classification and retrieval of traffic video using auto-regressive stochastic processes," in *Proceedings of the 2005 IEEE Intelligent Vehicles Symposium*, pp. 771–776, USA, June 2005.