LINEARIZED KERNEL REPRESENTATION LEARNING FROM VIDEO TENSORS BY EXPLOITING MANIFOLD GEOMETRY FOR GESTURE RECOGNITION

Krishan Sharma *Renu Rameshan*

School of Computing and Electrical Engineering, Indian Institute of Technology, Mandi

ABSTRACT

A video tensor is an organized multidimensional array of numerical values. In this paper, we explore the underlying manifold geometry of a video tensor by factorizing it using modified higher order singular value decomposition (HOSVD). Each factor (mode matrix) of a video tensor obtained after modified HOSVD can be thought of as a subspace and hence represents a point in Grassmann manifold ($\mathcal{M}_{GM}$). These factors cumulatively represent a point in product Grassmann manifold ($\mathcal{M}_{PGM}$). We propose a novel kernel for $\mathcal{M}_{PGM}$ that measures the similarity between two points in $\mathcal{M}_{PGM}$ and generates a kernel-gram matrix. For representation learning, we diagonalize the obtained kernel-gram matrix and generate a small fixed length representation corresponding to each point in $\mathcal{M}_{PGM}$. Classification is performed in sparse framework with minimum residual error as classifier. Experimentation is carried out over Cambridge hand gesture and UMD Keck body gesture databases for both static and dynamic settings. Experimental study shows that even with small length feature representation, there is a significant improvement in classification results as compared to state-of the-art techniques.

Index Terms— Grassmann manifold, product Grassmann manifold, video tensor, sparse representation, matrix diagonalization, kernel methods.

1. INTRODUCTION

Human gestures convey meaningful information through physical movement of various body parts such as arm, finger, leg, eye etc. However, recognizing gestures from videos has been a challenging problem for vision community from several years. Different human body structures, way of performing the gestures, clothing, pose variations and various dynamic conditions such as light variation or occlusion make the recognition task difficult. Several works [1–9] exist in literature with various classification strategies for gesture/action recognition task.

The first step in gesture recognition is to learn descriptive and discriminative features which correspond to a gesture video. Holistic features based representations [1] like Motion History Image and Motion Energy Image encode the gesture dynamics into a single image but are very sensitive to change in view point. A global feature representation by extracting the space-time shape properties is proposed in [2]. Space-time interest point based approaches such as 3D-SIFT [3], HOG-3D [10] etc. extract the local representations but capture only short temporal information. To capture features for long duration, [4] and [5] use trajectory based features like HOF, MBH respectively. Introduction of deep networks in action recognition unified the representation learning and classification task into a single framework. Though deep networks [6, 7] show a good recognition performance, it comes at the cost of millions of parameters learned by training networks. Despite having a fair recognition performance, all of the above methods do not explore the true underlying geometry of videos data points. By underlying geometry we mean either the subspace structure of data or the manifold structure. Being sensitive to such structures is important, since it enables usage of distance measures suitable for such structures. In the methods listed above, features are points in Hilbert space & Euclidean geometry is followed for measuring similarity and hence classification. However true geometry of data points can be captured by modeling them in Riemannian manifold and by considering Riemannian geometry.

Mathematical modeling of data as points in product manifold has been explored by many researchers. A product manifold is a Cartesian product of simple factor manifolds with dimensionality equal to the sum of all factor dimensions [11]. The use of product manifold in vision applications are by Ma et al. [12] for 3-D motion, Datta et al. [13] in local linear motion models and by Shaji et al. [14] in structure from motion.

A notable work in modeling the geometry of action videos is done by Lui et al. [8, 9]. They modeled action videos as points in product space of three Grassmann manifolds [15], $\mathcal{M}_{PGM}$, and used geodesic distance as a similarity measure. Though the method has the advantage of simplicity, it suffers from the following limitations: 1) There is a no unique representation for a point in $\mathcal{M}_{PGM}$ as each factor manifold represents a subspace and hence state-of-the-art Euclidean space based classifiers cannot be directly used; the only classifier that can be used is K-nearest neighbor (KNN) as it uses only the distance between points. 2) Space complexity of each point in $\mathcal{M}_{PGM}$ is very high as it is represented by three matrices and their size is directly proportional to the video size. 3) Each $\mathcal{M}_{PGM}$ point is represented by its own descriptive features which are not good for classification task requiring discriminative features.

We address these issues in our work by proposing a kernel that essentially maps the points in $\mathcal{M}_{PGM}$ to Hilbert space $\mathbb{R}^d$ without compromising the true manifold geometry. The obtained features are of small length and carry the discriminative information and hence suited for the classification task. The main contribution of this work is the novel product space based kernel that finds the non-negative similarity score between two points in $\mathcal{M}_{PGM}$. Overall we create a classification pipeline starting from modified HOSVD representation of videos, followed by a kernel, which is linearized to obtained features which are descriptive as well as discriminative. Experimentation shows a marked improvement in performance over the state-of-the-art.

The rest of the paper is organized as follows: Section 2 describes the tensor decomposition followed by the proposed kernel and representation learning in Section 3 and 4 respectively. Sparse representation (SR) based classifier is explained in Section 5. Detailed experimental analysis is given in Section 6 followed by conclusion in Section 7.

2. VIDEO TENSOR DECOMPOSITION

In order to make this paper self contained, a brief review of tensor decomposition [16] using HOSVD is given in this section. A video tensor is a 3-dimensional array (order 3 data tensor) and hence can be represented as a point in $\mathbb{R}^{(n_1 \times n_2 \times n_3)}$, where n_1, n_2, n_3 denote the image height, image width and video depth (number of frames) respectively. In general, a K-order data tensor, $\mathcal{T} \in \mathbb{R}^{(n_1 \times \cdots \times n_i \times \cdots \times n_K)}$, can be factorized using HOSVD [17] into K different modes as

$$\mathcal{T} = \boldsymbol{C} \times_1 \boldsymbol{U}^{(1)} \cdots \times_i \boldsymbol{U}^{(i)} \cdots \times_K \boldsymbol{U}^{(K)}. \quad (1)$$

$\boldsymbol{U}^{(i)} \in \mathbb{R}^{n_i \times n_i}$ is an orthogonal matrix representing i^{th} mode of $\mathcal{T}$. $\boldsymbol{C} \in \mathbb{R}^{(n_1 \times \cdots \times n_i \times \cdots \times n_K)}$ is the core tensor that denotes the inter mode interaction and generally non diagonal for higher order tensors. $\times_i$ denotes the i^{th} mode multiplication. Matrix $\boldsymbol{U}^{(i)}$ is obtained via matrix unfolding of tensor $\mathcal{T}$ for i^{th} mode as

$$\boldsymbol{T}^{(i)} = \boldsymbol{U}^{(i)} \boldsymbol{\Sigma}^{(i)} \boldsymbol{V}^{(i)^T}, \quad (2)$$

where $\boldsymbol{T}^{(i)} \in \mathbb{R}^{n_i \times (n_1 \ldots n_{i-1} n_{i+1} \ldots n_K)}$ is i^{th} mode unfolded matrix represented by single order row vectors and $K-1$ order column vectors. $\boldsymbol{U}^{(i)}$ and $\boldsymbol{V}^{(i)}$ are the matrices corresponding to left and right singular vectors of $T^{(i)}$ obtained by SVD which span the column and row space of $\boldsymbol{T}^{(i)}$ respectively. $\boldsymbol{\Sigma}^{(i)}$ is a diagonal matrix with singular values.

However, representing a tensor by $\boldsymbol{U}^{(i)}$'s as point in product manifold is not a good idea. The main reasons being: 1) $\boldsymbol{U}^{(i)} \in \mathbb{R}^{n_i \times n_i}$ is an orthogonal matrix that represents a point in special orthogonal group, $SO(n_i)$, and no closed form solution for distance between two points exists in $SO(n_i)$ [15]. 2) Rotating $\boldsymbol{U}^{(i)}$ by a rotation matrix $R^{(i)} \in \mathbb{R}^{n_i \times n_i}$ results in a different point in $SO(n_i)$ though both span the same subspace and this is highly undesirable.

So a better strategy is to represent the tensor as a point in product of $\mathcal{M}_{GM}$'s [8] than the product of SO's. This can be easily achieved by modifying the HOSVD as

$$\mathcal{T} = \hat{\boldsymbol{C}} \times_1 \boldsymbol{V}^{(1)} \cdots \times_i \boldsymbol{V}^{(i)} \cdots \times_K \boldsymbol{V}^{(K)},$$

where $\boldsymbol{V}^{(i)} \in \mathbb{R}^{(n_1 \ldots n_{i-1} n_{i+1} \ldots n_K) \times n_i}$ denotes the column span of $\boldsymbol{T}^{(i)^T}$ and $\hat{\boldsymbol{C}} \in \mathbb{R}^{(n_2 \ldots n_K) \times \cdots \times (n_1 \ldots n_{i-1} n_{i+1} \ldots n_K) \times \cdots \times (n_1 \ldots n_{K-1})}$ is the core tensor. In this work we use this representation. In order to overcome the disadvantages of this representation mentioned in the introduction, we define a kernel in $\mathcal{M}_{PGM}$ which is explained in the next section.

3. THE PROPOSED KERNEL

We propose a novel kernel that measures the similarity between two points in $\mathcal{M}_{PGM}$. Let $\mathcal{V}_l = \{\boldsymbol{V}_l^{(1)}, \ldots, \boldsymbol{V}_l^{(i)}, \ldots, \boldsymbol{V}_l^{(K)}\}$ represent a point in $\mathcal{M}_{PGM} = \mathcal{M}_{GM}^{(1)} \times \cdots \times \mathcal{M}_{GM}^{(i)} \times \cdots \times \mathcal{M}_{GM}^{(K)}$, where $\boldsymbol{V}_l^{(i)} \in \mathcal{M}_{GM}^{(i)}$. We generate the kernel function by measuring similarities between the corresponding factor manifolds of $\mathcal{M}_{PGM}$ as

$$k(\mathcal{V}_l, \mathcal{V}_m) = \sum_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)}||_F^2, \quad (3)$$

$k : \mathcal{M}_{PGM} \times \mathcal{M}_{PGM} \to \mathbb{R}^+$. For each of the i^{th} factor manifold, projection distance [18] between two points can be expressed as

$$d_{proj}^2(\boldsymbol{V}_l^{(i)}, \boldsymbol{V}_m^{(i)}) = n_i - ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)}||_F^2. \quad (4)$$

Projection distance is ℓ_2 norm of the sine of principal angles between subspaces while geodesic distance [15] is only ℓ_2 norm of the same. Both projection and geodesic distances are same for small values of angles while distances are proportional for high value of angles. The kernel proposed in equation 3 is derived from the projection distance which is an approximation to the geodesic distance and hence preserves the geometry.

w_i ($w_i > 0$) represents the weight corresponding to individual kernel for i^{th} factor manifold, $\mathcal{M}_{GM}^{(i)}$. w_i is a controlling parameter that decides how much weightage should be given to different modes. Low weightage can be given to those modes which are not discriminative for different classes. Selection of weights is data specific and can be fixed empirically. Since points on $\mathcal{M}_{GM}^{(i)}$ represent the subspaces, column vectors of $\boldsymbol{V}_l^{(i)}$ denotes the basis for subspaces which obviously can not be unique. To make the kernel function $k(\mathcal{V}_l, \mathcal{V}_m)$ as rotation invariant to different representations of $\boldsymbol{V}_l^{(i)}$'s and $\boldsymbol{V}_m^{(i)}$'s, we need to satisfy an extra well-definedness condition along with Mercer's condition [19] of positive definiteness.

Positive definiteness - A kernel is positive definite kernel function if

$$\sum_{l,m} c_l c_m \; k(\mathcal{V}_l, \mathcal{V}_m) \geq 0 \quad \forall c_l, c_m \in \mathbb{R}$$

Proof.

$$\sum_{l,m} c_l c_m \; k(\mathcal{V}_l, \mathcal{V}_m) = \sum_{l,m} c_l c_m \sum_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)}||_F^2$$

$$= \sum_{l,m} c_l c_m \sum_{i=1}^{K} w_i \; tr(\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)} \boldsymbol{V}_m^{(i)^T} \boldsymbol{V}_l^{(i)})$$

$$= \sum_{i=1}^{K} w_i \sum_{l,m} c_l c_m \; tr(\boldsymbol{V}_l^{(i)} \boldsymbol{V}_l^{(i)^T})(\boldsymbol{V}_m^{(i)} \boldsymbol{V}_m^{(i)^T})$$

$$= \sum_{i=1}^{K} w_i \; tr(\sum_{l} c_l \boldsymbol{V}_l^{(i)} \boldsymbol{V}_l^{(i)^T})(\sum_{m} c_m \boldsymbol{V}_m^{(i)} \boldsymbol{V}_m^{(i)^T})$$

$$= \sum_{i=1}^{K} w_i ||\sum_{l} c_l \boldsymbol{V}_l^{(i)} \boldsymbol{V}_l^{(i)^T}||_F^2 \geq 0, \quad \because w_i > 0.$$

□

Well-definedness- A kernel is well defined if it is invariant to different subspace representations, i.e,

$$k(\mathcal{V}_l \mathcal{R}_l, \mathcal{V}_m \mathcal{R}_m) = k(\mathcal{V}_l, \mathcal{V}_m). \quad (5)$$

Here $\mathcal{R}_l = \{\boldsymbol{R}_l^{(1)}, \ldots, \boldsymbol{R}_l^{(i)}, \ldots, \boldsymbol{R}_l^{(K)}\}$, and $\boldsymbol{R}_l^{(i)} \in SO(n_i)$ denotes $n_i \times n_i$ dimensional orthogonal rotation matrix.

Proof.

$$k(\mathcal{V}_l \mathcal{R}_l, \mathcal{V}_m \mathcal{R}_m) = \Sigma_{i=1}^{K} w_i ||(\boldsymbol{V}_l^{(i)} \boldsymbol{R}_l^{(i)})^T \boldsymbol{V}_m^{(i)} \boldsymbol{R}_m^{(i)}||_F^2,$$
$$= \Sigma_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)} (\boldsymbol{R}_l^{(i)^T} \boldsymbol{R}_m^{(i)})||_F^2 = \Sigma_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)} (\boldsymbol{R}^{(i)})||_F^2$$
$$= \Sigma_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)}||_F^2 = k(\mathcal{V}_l, \mathcal{V}_m)$$

□

Fig 1. shows the pictorial illustration of proposed kernel. Gesture video tensors $\mathcal{T}_l$ and $\mathcal{T}_m$ are decomposed into respective core tenors and three mode matrices. Each mode matrix represents a subspace and hence all the matrices for that mode are modeled as points in a factor manifold, $\mathcal{M}_{GM}$. Cartesian product of these three factor manifolds forms the $\mathcal{M}_{PGM}$. Kernel $k(\mathcal{V}_l, \mathcal{V}_m)$ finds the similarity measure by weighted sum of similarity measures of points in the factor manifolds.

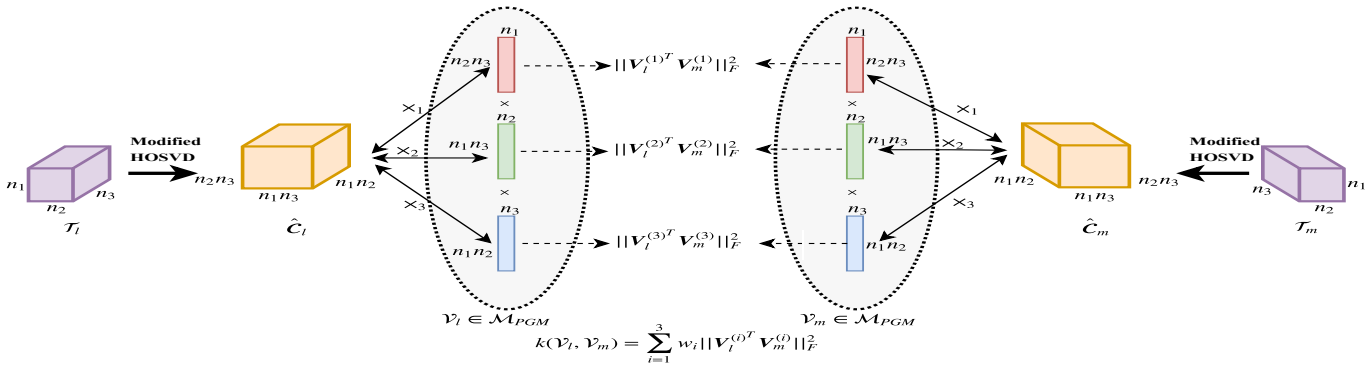

Fig. 1: Pictorial illustration of the proposed kernel

Algorithm 1 The proposed method

Inputs:

(i) Training video tensor database $\mathcal{D} = \{\mathcal{T}_l, t_l\}_{l=1}^{N}$, where $t_l \in \{1,2,...,c\}$ denotes gesture class label.

(ii) Test video tensor $\mathcal{T}_{test}$.

1: **Procedure:**

2: Decompose $\mathcal{T}_l$ into $\mathcal{V}_l = \{\boldsymbol{V}_l^{(1)}, \ldots, \boldsymbol{V}_l^{(i)}, \ldots, \boldsymbol{V}_l^{(K)}\}$ using modified HOSVD
where $\mathcal{V}_l \in \mathcal{M}_{PGM}, \boldsymbol{V}_l^{(i)} \in \mathcal{M}_{GM}^{(i)}$ such that
$\mathcal{M}_{PGM} = \mathcal{M}_{GM}^{(1)} \times \cdots \times \mathcal{M}_{GM}^{(i)} \times \cdots \times \mathcal{M}_{GM}^{(K)}$

3: Compute $\boldsymbol{\Psi}_{train}$ and $\boldsymbol{\psi}(.,\mathcal{V}_{test})$ using kernel function

$$k(\mathcal{V}_l, \mathcal{V}_m) = \Sigma_{i=1}^{K} w_i ||\boldsymbol{V}_l^{(i)^T} \boldsymbol{V}_m^{(i)}||_F^2,$$

4: Factorize $\boldsymbol{\Psi}_{train}$ using SVD

$$\boldsymbol{\Psi}_{train} = \boldsymbol{P}\boldsymbol{\Lambda}_N\boldsymbol{P}^T.$$

5: Generating the d-dimensional feature representation $(d \leq N)$ corresponding to each tensor by linearizing the kernel

$$\boldsymbol{X}_{train}^{d} = {\boldsymbol{\Lambda}_d}^{-\frac{1}{2}} \boldsymbol{P}^T \boldsymbol{\Psi}_{train},$$

$$\boldsymbol{x}_{test}^{d} = \boldsymbol{\Lambda}_d^{-\frac{1}{2}} \boldsymbol{P}^T \boldsymbol{\psi}(.,\mathcal{V}_{test}),$$

where $\boldsymbol{\Lambda}_d = \boldsymbol{\Lambda}_N(1:d, 1:N)$.

6: Obtain sparse coefficients $\hat{\boldsymbol{y}}$ by solving,

$$\hat{\boldsymbol{y}} = \arg\min_{\boldsymbol{y}} ||\boldsymbol{x}_{test}^{d} - \boldsymbol{X}_{train}^{d}\boldsymbol{y}||_2^2 + \alpha||\boldsymbol{y}||_1.$$

7: Obtain label by minimizing residual error,

$$label(\boldsymbol{x}_{test}^{d}) = \arg\min_{j=1,2,...,c} ||\boldsymbol{x}_{test}^{d} - \boldsymbol{X}_{train}^{d}\Gamma_j||_2^2.$$

8: $label(\mathcal{T}_{test}) = label(\boldsymbol{x}_{test}^{d})$.

Outputs:

(i) $label(\mathcal{T}_{test})$.

4. REPRESENTATION LEARNING USING KERNEL LINEARIZATION

We generate a kernel-gram matrix $\boldsymbol{\Psi}_{train} \in \mathbb{R}^{N\times N}$ using kernel function $k(\mathcal{V}_l, \mathcal{V}_m)$ from equation 3. Kernel linearization [19, 20] is the process of obtaining the virtual feature representations in mapped space by diagonalizing the kernel-gram matrix. Being a symmetric matrix, Ψ_{train} can be decomposed as

$$\boldsymbol{\Psi}_{train} = \boldsymbol{P}\boldsymbol{\Lambda}_N\boldsymbol{P}^T = (\boldsymbol{\Lambda}_N^{\frac{1}{2}}\boldsymbol{P}^T)^T(\boldsymbol{\Lambda}_N^{\frac{1}{2}}\boldsymbol{P}^T), \tag{6}$$

where $\boldsymbol{P}$ is an $N \times N$ matrix with orthonormal columns and $\boldsymbol{\Lambda}_N$ is a diagonal matrix of singular values. Each entry of $\boldsymbol{\Psi}_{train}$ can be thought of as an inner product between the mapped signals in Hilbert space, $\mathcal{H}$, i.e. : $\boldsymbol{\Psi}_{train}(l,m) = k(\mathcal{V}_l, \mathcal{V}_m) = \langle \phi(\mathcal{V}_l), \phi(\mathcal{V}_m) \rangle$ with mapping $\phi : \mathcal{M}_{PGM} \rightarrow \mathcal{H}$. Therefore $\boldsymbol{\Psi}_{train}$ can also be decomposed as

$$\boldsymbol{\Psi}_{train} = \Phi(\mathcal{V})^T\Phi(\mathcal{V}) = (\boldsymbol{X}_{train}^{N})^T(\boldsymbol{X}_{train}^{N}), \tag{7}$$

where $\mathcal{V} = \{\mathcal{V}_l\}_{l=1}^{N}$ denotes the complete set of training points in $\mathcal{M}_{PGM}$ and $\Phi(\mathcal{V}) = [\phi(\mathcal{V}_1)\ldots\phi(\mathcal{V}_l)\ldots\phi(\mathcal{V}_N)]$. l^{th} column of $\boldsymbol{X}_{train}^{N} \in \mathbb{R}^{N\times N}$ represents an N-dimensional virtual feature corresponding to $\mathcal{V}_l$ and hence can be represented as

$$\boldsymbol{X}_{train}^{N} = \boldsymbol{\Lambda}_N^{\frac{1}{2}}\boldsymbol{P}^T = \boldsymbol{\Lambda}_N^{-\frac{1}{2}}\boldsymbol{P}^T\boldsymbol{\Psi}_{train}. \tag{8}$$

Here inverse is corresponding to non-zero diagonal entries only. Since all the singular values are not significant, a small length representation can be generated by considering $d(d < N)$ singular values as

$$\boldsymbol{X}_{train}^{d} = \boldsymbol{\Lambda}_d^{-\frac{1}{2}}\boldsymbol{P}^T\boldsymbol{\Psi}_{train}. \tag{9}$$

$\boldsymbol{\Lambda}_d$ is obtained by selecting first d rows of $\boldsymbol{\Lambda}_N$ and each column of $\boldsymbol{X}_{train}^{d}$ represents a d- length vector. Similarly feature representation for a test point $\mathcal{V}_{test}$ can be written as $\boldsymbol{x}_{test}^{d} = \boldsymbol{\Lambda}_d^{-\frac{1}{2}}\boldsymbol{P}^T\boldsymbol{\psi}(.,\mathcal{V}_{test})$, where $\boldsymbol{\psi}(.,\mathcal{V}_{test}) = [k(\mathcal{V}_1,\mathcal{V}_{test})\ldots k(\mathcal{V}_l,\mathcal{V}_{test})\ldots k(\mathcal{V}_N,\mathcal{V}_{test})]^T$.

5. SPARSE REPRESENTATION BASED CLASSIFIER

We employ SR based classifier for classification task. Each $\boldsymbol{x}_{test}^{d}$ is expressed as a combination of few training examples from $\boldsymbol{X}_{train}^{d}$ and sparse coefficient vector $\hat{\boldsymbol{y}}$ is obtained by solving

$$\hat{\boldsymbol{y}} = \arg\min_{\boldsymbol{y}} ||\boldsymbol{x}_{test}^{d} - \boldsymbol{X}_{train}^{d}\boldsymbol{y}||_2^2 + \alpha||\boldsymbol{y}||_1, \tag{10}$$

where $\hat{\boldsymbol{y}} = [\hat{\boldsymbol{y}}_1 \ldots \hat{\boldsymbol{y}}_j \ldots \hat{\boldsymbol{y}}_c]^T$ denotes the N length sparse representation corresponding to $\boldsymbol{x}_{test}^{d}$ and $\hat{\boldsymbol{y}}_j$ as j^{th} class coefficients. α is the

Table 1: Recognition results on Cambridge hand gesture dataset

Method	Set1	Set2	Set3	Set4	Accuracy
TCCA [21]	81%	81%	78%	86%	82±3.5 %
RLPP [23]	86%	86%	85%	88%	86.3±1.3 %
TB [24]	88%	84%	85%	87%	86 ± 3.0%
PM +1-NN [8]	89%	86 %	89%	87%	88±2.1 %
PM + Regression [9]	93%	89%	91%	94%	91.7±2.3%
gSC [25]	93%	92%	93%	94%	93.3±0.9 %
Proposed approach	**95%**	**94%**	**97%**	**95%**	**95.3 ± 1.3%**

Table 2: Recognition results on UMD Keck body gesture dataset

Method	Static Setting	Dynamic Setting
HOG3D [26]	-	53.6 %
Shape Manifold [27]	82%	-
MMI +SIFT [28]	95%	-
TB [24]	92.1 %	91.1 %
Prototype-Tree [29]	95.2 %	91.1 %
PM + 1-NN [8]	92.9%	92.3 %
PM + Regression [9]	94.4 %	92.3 %
Proposed approach	**100** %	**93.5** %

sparsity parameter. Class label is obtained by minimizing the residual error as

$$label(\boldsymbol{x}^d_{test}) = \underset{j=1,2,\ldots,c}{\arg\min} ||\boldsymbol{x}^d_{test} - \boldsymbol{X}^d_{train}\Gamma_j||^2_2. \quad (11)$$

Here Γ_j is a characteristic function that picks coefficients corresponding to j^{th} class only. Algorithm 1 shows the pseudo-code of the proposed approach.

6. EXPERIMENTAL ANALYSIS

In order to validate the proposed approach, we have performed experimentations over two datasets for gesture recognition task namely Cambridge hand gesture database [21] and UMD Keck body gestures database [22]. We have confined our experimentation to only these two datasets since papers related to gesture recognition have given their results only for these datasets. The detailed description of these databases is given below:

(i) *Cambridge hand gesture database* [21]- consists of 900 video sequences of 9 hand movements gesture classes with 100 images per class. These hand movements are defined by 3 primitive shapes and 3 primitive motions. Each gesture is performed by 2 persons in 10 arbitrary motions under 5 different illumination conditions. These 5 illumination conditions are labeled as SET1 to SET5.

(ii) *UMD Keck body gestures database* [22]- consists of 14 different gesture class videos of military signals like turn left, turn right, stop etc. These gestures are performed by 3 subjects and videos are recorded in both static and dynamic conditions. In static condition, both camera and subject remain stationary while performing gesture and total of 126 videos are recorded. In dynamic condition, 168 videos are recorded for both camera and subject moving.

For Cambridge hand gesture database, we have used the standard protocol followed by Kim and Cipolla [21] for recognition task

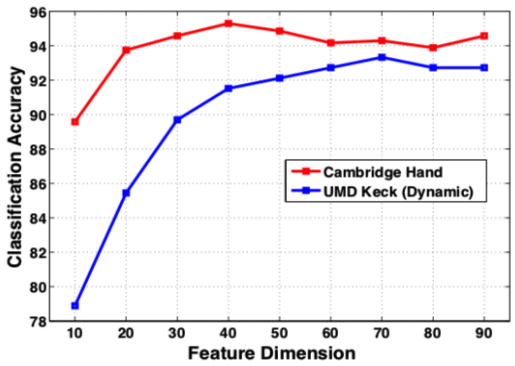

Fig. 2: Classification accuracy as a function of feature dimension for Cambridge hand and UMD keck body dataset (dynamic settings).

in which SET5 (videos under normal illumination) is used for training and remaining for testing. Each video is resized to a fixed size of $20 \times 20 \times 32$ by collecting the middle 32 frames. No space-time alignment on videos is performed. We have fixed all the w_i's as 1 in all of the experiments. Sparsity parameter α is tuned over a range of values. Recognition results on different illumination sets are shown in Table 1. It is evident from the results that our approach shows a significant improvement in classification accuracy. Compared to other product manifold based approaches [8,9], the proposed method gives 4% to 7% increment in classification performance.

For classification on UMD Keck body gestures database, first we have cropped the videos by tracking the region of interest by a correlation filter. We have resized all the videos to $20 \times 20 \times 40$ without performing any space-time alignment. Videos with lesser frames are appended with initial frames to make them of fixed size. Standard protocol followed by Lin et al. [29] is used in both static and dynamic settings. In static settings, we have used leave one subject out protocol(LOSO). In dynamic settings, we have used videos captured under the static environment for training and dynamic environment for testing. Table 2 shows the performance comparison of our approach with other manifold based techniques under same protocol. 100% accuracy is achieved in static settings even with small feature representation of length 20. This is due to the reason that videos are recorded in ideal settings and variation due to pose and background clutter are not present. However for dynamic settings where change in background and pose variations are present, 1% improvement in result can be seen. Fig. 2 shows the change in classification accuracy with varying feature length. It can be seen in the figure that maximum performance is achieved for Cambridge hand and UMD keck body datasets at feature dimension of 40 and 70 respectively after which performance gets saturated.

7. CONCLUSION

We have shown that a discriminative as well as descriptive feature representation for videos can be obtained by defining a kernel in product of Grassmann manifolds, $\mathcal{M}_{PGM}$. This is evident from the significant improvement in classification results in all except one set of data, where we have a marginal improvement. Comparing to deep neural networks, advantages of the proposed approach can be summarized in terms of its simplicity and no prior training. Though the approach is entirely pixel based i.e no higher level features are extracted, noteworthy classification performance is achieved.

8. REFERENCES

[1] Aaron F. Bobick and James W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[2] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri, "Actions as space-time shapes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 12, pp. 2247–2253, 2007.

[3] Paul Scovanner, Saad Ali, and Mubarak Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007, pp. 357–360.

[4] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu, "Action recognition by dense trajectories," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 3169–3176.

[5] Mihir Jain, Herve Jegou, and Patrick Bouthemy, "Better exploiting motion for better action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2555–2562.

[6] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[7] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes, "Spatiotemporal multiplier networks for video action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 7445–7454.

[8] Yui Man Lui, J Ross Beveridge, and Michael Kirby, "Action classification on product manifolds," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 833–839.

[9] Yui Man Lui, "Human gesture recognition on product manifolds," *Journal of Machine Learning Research*, vol. 13, no. Nov, pp. 3297–3321, 2012.

[10] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *BMVC 2008-19th British Machine Vision Conference*. British Machine Vision Association, 2008, pp. 275–1.

[11] John M Lee, "Smooth manifolds," in *Introduction to Smooth Manifolds*, pp. 1–29. Springer, 2003.

[12] Yi Ma, Jana Kosecka, and Shankar Sastry, "Optimal motion from image sequences: A riemannian viewpoint," in *In Proceeding of the Conference on Mathematical Theory of Networks and Systems*. Citeseer, 1998.

[13] Ankur Datta, Yaser Sheikh, and Takeo Kanade, "Modeling the product manifold of posture and motion," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1034–1041.

[14] Appu Shaji, Sharat Chandran, and David Suter, "Manifold optimisation for motion factorisation," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[15] Alan Edelman, Tomás A Arias, and Steven T Smith, "The geometry of algorithms with orthogonality constraints," *SIAM journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303–353, 1998.

[16] Tamara G Kolda and Brett W Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[17] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[18] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin 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*, 2015, pp. 140–149.

[19] Alex J Smola and Bernhard Schölkopf, *Learning with kernels*, vol. 4, Citeseer, 1998.

[20] Krishan Sharma, Shikha Gupta, AD Dileep, and Renu Rameshan, "Scene image classification using reduced virtual feature representation in sparse framework," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2701–2705.

[21] Tae-Kyun Kim, Shu-Fai Wong, and Roberto Cipolla, "Tensor canonical correlation analysis for action classification," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[22] Zhuolin Jiang, Zhe Lin, and Larry Davis, "Recognizing human actions by learning and matching shape-motion prototype trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 533–547, 2012.

[23] Mehrtash T Harandi, Conrad Sanderson, Arnold Wiliem, and Brian C Lovell, "Kernel analysis over riemannian manifolds for visual recognition of actions, pedestrians and textures," in *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*. IEEE, 2012, pp. 433–439.

[24] Yui Man 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.

[25] Mehrtash Harandi, Richard Hartley, Chunhua Shen, Brian Lovell, and Conrad Sanderson, "Extrinsic methods for coding and dictionary learning on grassmann manifolds," *International Journal of Computer Vision*, vol. 114, no. 2-3, pp. 113–136, 2015.

[26] Piotr Bilinski and Francois Bremond, "Evaluation of local descriptors for action recognition in videos," in *International Conference on Computer Vision Systems*. Springer, 2011, pp. 61–70.

[27] Mohamed F Abdelkader, Wael Abd-Almageed, Anuj Srivastava, and Rama Chellappa, "Silhouette-based gesture and action recognition via modeling trajectories on riemannian shape manifolds," *Computer Vision and Image Understanding*, vol. 115, no. 3, pp. 439–455, 2011.

[28] Qiang Qiu, Zhuolin Jiang, and Rama Chellappa, "Sparse dictionary-based representation and recognition of action attributes," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 707–714.

[29] Zhe Lin, Zhuolin Jiang, and Larry S Davis, "Recognizing actions by shape-motion prototype trees," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 444–451.