

Action Classification on Product Manifolds

Yui Man Lui, J. Ross Beveridge

Department of Computer Science, Colorado State University, Fort Collins, CO 80523, USA

{lui}{ross}@cs.colostate.edu

Michael Kirby

Department of Mathematics, Colorado State University, Fort Collins, CO 80523, USA

kirby@math.colostate.edu

Abstract

Videos can be naturally represented as multidimensional arrays known as tensors. However, the geometry of the tensor space is often ignored. In this paper, we argue that the underlying geometry of the tensor space is an important property for action classification. We characterize a tensor as a point on a product manifold and perform classification on this space. First, we factorize a tensor relating to each order using a modified High Order Singular Value Decomposition (HOSVD). We recognize each factorized space as a Grassmann manifold. Consequently, a tensor is mapped to a point on a product manifold and the geodesic distance on a product manifold is computed for tensor classification. We assess the proposed method using two public video databases, namely Cambridge-Gesture gesture and KTH human action data sets. Experimental results reveal that the proposed method performs very well on these data sets. In addition, our method is generic in the sense that no prior training is needed.

1. Introduction

Human-computer interaction has attracted considerable attention in recent years [16]. Action classification is one key aspect of human-computer interaction, and a variety of methods have been proposed to construct action classifiers. Bissacco *et al.* [2] employed an ARMA model for human gait recognition. Schüldt *et al.* [19] combined local features and SVMs for human action classification. Turaga *et al.* [21] applied Procrustes distances on special manifolds for activity recognition. Recently, Laptev *et al.* [11] proposed a method using spatio-temporal bag-of-features combined with multi-channel SVMs for action classification. Despite these efforts, reliable action classification remains a hard problem because of the complexity of human motions. To address this concern, more powerful tools are needed,

and one such tool is multilinear algebra.

Multilinear algebra is a mathematical framework for high order tensors which capture multiple factor variations and interactions. Tensor computing has been successfully applied to many computer vision applications such as face recognition [23], visual tracking [13], and action classification [22, 8]. However, the advantages of representing a tensor on a product manifold in the context of classification have not been explored. In this paper, we represent a video as a 3rd order tensor and demonstrate that the geometric structure of the tensor space is discriminative for action classification.

It is known that multidimensional data can be considered as a high order tensor. Previous methods [23, 22, 8] often learn a projection to characterize a lower dimensional tensor subspace and apply discriminant analysis. Such techniques are usually complicated due to the nature of the learning algorithms. In addition, they require a large amount of training data and may suffer from a generalization problem. With so much effort paid to the learning algorithms, comparatively little work has investigated the underlying geometry of the tensor space.

The method proposed in this paper employs tensors to perform action classification from a different perspective. First, it is a non-trained method, and so avoids training data and generalization problems. Second, we focus attention on the geometric structure of the tensor space, and this in turn provides insight into the existing multilinear algebra, its underlying geometric interpretation, and how this geometry provides a robust basis for classification.

Our approach begins by abstracting an N order tensor as a point on a product manifold where the number of factors is given by the order of the tensor. Because each factor of the tensor can be characterized as an orthogonal matrix via a decomposition procedure, it is represented on a Grassmann manifold. However, traditional Higher Order Singular Value Decomposition (HOSVD) [12] does not factorize

a space which preserves the geodesic distance in the context of video classification. It is therefore helpful to modify the common definition of HOSVD, and with the modified HOSVD, each factor manifold is related to a single order of the tensor spanned by the column space. As such, an N order tensor yields N factor manifolds.

Our approach then draws upon the fact that the geodesic on a product manifold is equivalent to the Cartesian product of geodesics from multiple factor manifolds. In other words, elements of a product manifold are from the set of all elements on factor manifolds. Action classification is then performed on the basis of geodesic distance on a product manifold associated with an action video.

The most important contribution of this paper is the presentation of a new way of relating tensors on a product manifold for action video classification. Using geodesic distance on a product manifold to compare videos, we demonstrate that a simple nearest neighbor classifier can perform very well: comparable with the best highly trained algorithms in the literature.

The rest of this paper is organized as follows: Related works are summarized in Section 2. Tensor algebra and geodesic distances are reviewed in Section 3. The formulation of the proposed product manifold is presented in Section 4. Experimental results are reported in Section 5. Finally, discussion and conclusions are given in Section 6 and Section 7, respectively.

2. Related Works

Many researchers have investigated various practical applications of product spaces in recent years. We summarize some of this past work here.

Ma *et al.* [15] estimated 3D motion from a sequence of images. The motion parameters are represented on an essential manifold which is a set of 3×3 rotation matrices on a Lie group. The essential manifold is viewed as a product of Stiefel manifolds. As such, the parallel transport and geodesics are computed on each factor manifold. Newton's method is employed for optimization.

Shaji *et al.* [20] formulated the structure from motion problem on a product manifold. The underlying parameter space is constrained from the product manifold. The Gauss-Newton method is used for optimization over a N -fold product manifold of a special Euclidean group.

Schoenemann *et al.* [18] performed image segmentation in product spaces. The product space is spanned by the image and the prior contour. The contour of an image is considered as the shortest path found by minimizing the geodesic energy. Then, the joint space of an image and contour are used for segmentation.

Eldén and Savas [7] presented a method to approximate a 3rd order tensor. Newton optimization is exploited to search the orthogonal matrices on a product manifold represented

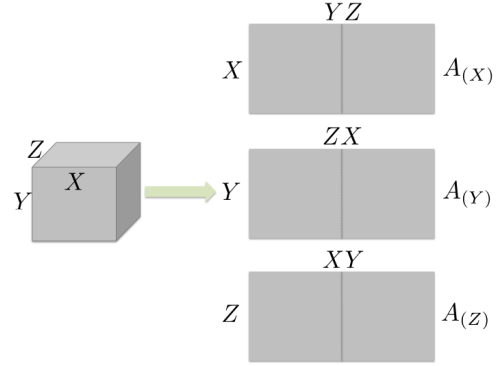


Figure 1. An example of matrix unfolding of a 3rd order tensor.

by three Grassmann manifolds. The authors compute the gradient directions on three tangent spaces of Grassmann manifolds.

Datta *et al.* [5] modeled the nonlinear motion manifold as a collection of local linear models. This method learns a selection of mappings that are used to encode the motion manifold. The mappings are collected on a product space as a motion manifold.

3. Mathematical Background

In this section, we briefly review the background mathematics used in this paper, i.e., elements of tensor algebra and geodesic distance on Grassmann manifolds.

3.1. Tensor Representation

A video can be naturally represented as a third order tensor $\in \mathbb{R}^{X \times Y \times Z}$ where X , Y , and Z are the image width, image height, and video length, respectively. Tensors can be regarded as a multilinear mapping over a set of vector spaces. Generally, useful information can be extracted using tensor decompositions. In particular, a Higher Order Singular Value Decomposition (HOSVD) [12] is considered in this paper. A recent review paper on tensor decompositions can be found in [10]. Before we describe HOSVD, we illustrate a building block operation called matrix unfolding.

3.1.1 Matrix Unfolding

Let \mathcal{A} be an order N tensor $\in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. \mathcal{A} can be converted to a set of matrices via a matrix unfolding operation. Matrix unfolding maps a tensor \mathcal{A} to a set of matrices $A_{(1)}, A_{(2)}, \dots, A_{(N)}$, where $A_{(k)} \in \mathbb{R}^{I_k \times (I_1 \times \dots \times I_{k-1} \times I_{k+1} \times \dots \times I_N)}$ is a mode- k matrix of \mathcal{A} . An example of matrix unfolding of a 3rd order tensor is given in Figure 1, where the rows are represented by a single order of the tensor and the columns are composed by two orders of the tensor.

3.1.2 HOSVD

Just as a matrix can be factorized using a Singular Value Decomposition (SVD), a tensor can also be factorized using HOSVD. HOSVD operates on the unfolded matrices $A_{(k)}$, and each is factored as follows:

$$A_{(k)} = U^{(k)} \Sigma^{(k)} V^{(k)T} \quad (1)$$

where $\Sigma^{(k)}$ is a diagonal matrix, $U^{(k)}$ is an orthogonal matrix spanning the column space of $A_{(k)}$ associated with nonzero singular values, and $V^{(k)}$ is an orthogonal matrix spanning the row space of $A_{(k)}$ associated with nonzero singular values. Then, an N order tensor can be factorized using HOSVD as follows:

$$\mathcal{A} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \dots \times_n U^{(N)} \quad (2)$$

where $\mathcal{S} \in \mathbb{R}^{(I_1 \times I_2 \times \dots \times I_N)}$ is a core tensor, $U^{(1)}, U^{(2)}, \dots, U^{(N)}$ are orthogonal matrices spanning the column space described in Equation (1), and \times_k denotes mode- k multiplication. The core tensor signifies the interaction of mode matrices and is generally not diagonal when the tensor order is greater than two.

3.2. Geodesic Distance on Grassmann Manifolds

A Grassmann manifold [6] $\mathcal{G}_{n,p}$ is a set of p -dimensional linear subspaces of \mathbb{R}^n (p -planes in \mathbb{R}^n). Every point on an $\mathcal{G}_{n,p}$ represents a subspace spanned by the column space of an $n \times p$ orthogonal matrix. In addition, points are viewed as being equivalent if there exists a $p \times p$ orthogonal matrix Q_p which maps one point into the other. i.e.,

$$[\mathcal{Y}] = \{\mathcal{Y}Q_p : Q_p \in \mathbb{O}_p\} \quad (3)$$

where $[\mathcal{Y}]$ is an element on a Grassmann manifold $\mathcal{G}_{n,p}$.

Geodesic distances on a $\mathcal{G}_{n,p}$, can be characterized in terms of canonical angles [6, 14] and can be recursively computed as follows [3]:

$$\theta_k = \min_{x \in [\mathcal{X}], y \in [\mathcal{Y}]} \cos^{-1}(x^T y) = \cos^{-1}(x_k^T y_k) \quad (4)$$

subject to

$$\begin{aligned} \|x\| &= \|y\| = 1 \\ x^T x_i &= 0, \quad y^T y_i = 0, \quad i = 1, \dots, k-1 \end{aligned}$$

The space of a Grassmann manifold is curved, and the shortest path between two points on a manifold is geodesic. In this paper, we employ chordal distance [4] as our measure of geodesic distance. It is defined as:

$$d_c([\mathcal{X}], [\mathcal{Y}]) = \|\sin \theta\|_2 \quad (5)$$

where $[\mathcal{X}]$ and $[\mathcal{Y}]$ are p -dimensional linear subspaces in \mathbb{R}^n . The chordal distance considers the curved nature of Grassmannian spaces, describing the geodesic distance between two spanning sets, and is differentiable everywhere.

4. Product Manifolds

A product manifold can be recognized as a complex compound object in a high dimensional space. For example, a line in $\mathbb{R}^1 \times$ a circle in \mathbb{R}^2 becomes an infinite cylinder in \mathbb{R}^3 . The product manifold may be viewed as the cross section of a lower dimensional object. Formally, let $\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_q$ be a set of manifolds. The set $\mathcal{M}_1 \times \mathcal{M}_2 \times \dots \times \mathcal{M}_q$ is called the product of the manifolds where the manifold topology is equivalent to the product topology. Thus, a product manifold is defined as:

$$\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2 \times \dots \times \mathcal{M}_q \quad (6)$$

where \times denotes the Cartesian product and \mathcal{M}_k represents a factor manifold.

4.1. Factorization in Product Spaces

As discussed in Section 3.1, HOSVD is built up from the unfolded matrices (modes) via matrix unfolding. The variation of each mode is captured by HOSVD. However, as we are about to make clear, the traditional formulation of HOSVD will cause difficulties for us as we attempt to use HOSVD to relate a tensor on a product manifold.

The column of every unfolded matrix $A_{(k)}$ is composed by multiple orders from the original tensor. This fact can also be observed in Figure 1. Let m be the dimension of the columns, $I_1 \times \dots \times I_{k-1} \times I_{k+1} \times \dots \times I_N$, and n be the dimension of the rows, I_k , for an unfolded matrix $A_{(k)}$. We can then assume that the dimension of the columns is greater than the dimension of the rows, i.e. $m > n$. This implies that the unfolded matrix $A_{(k)}$ only spans n dimensions.

According to the SVD Equation (1), $U^{(k)} \in \mathbb{R}^{n \times n}$ is the orthogonal matrix spanning the column space associated with nonzero singular values. Because $m > n$, $U^{(k)}$ is actually a point on a special orthogonal group $\mathbb{SO}(n)$ and there is no closed-form solution for computing the geodesic distance on $\mathbb{SO}(n)$. Furthermore, the geodesic distance would always be zero when we view points on $\mathbb{SO}(n)$ as Grassmannian. This is due to the fact that Grassmannian can be represented as the quotient space of $\mathbb{SO}(n)$. In other words, we can always find a rotation matrix (a mapping) to rotate a point to the other in $\mathbb{SO}(n)$. As such, traditional HOSVD employed $U^{(k)}$ for factorization is not the appropriate choice to form a product manifold.

On the other hand, $V^{(k)}$ in Equation (1) is the orthogonal matrix spanning the row space associated with nonzero singular values and it only spans n dimensions. Therefore, $V^{(k)}$ can be represented by an $m \times n$ orthogonal matrix and it is a point on a Grassmann manifold. This observation motivates us to form a product manifold by modifying the HOSVD.

The modification for the existing HOSVD is simple. Since the $V^{(k)}$ is the orthogonal matrix spanning the row

space, all we need is to make it span the column space because a point on a Grassmann manifold represents a subspace spanned by the column space of an orthogonal matrix. To do so, we can simply take the $V^{(k)}$ from Equation (1). Alternatively, we can change the matrix unfolding of $A^{(k)}$ from $\mathbb{R}^{I_k \times (I_1 \times \dots \times I_{k-1} \times I_{k+1} \times \dots \times I_N)}$ to $\mathbb{R}^{(I_1 \times \dots \times I_{k-1} \times I_{k+1} \times \dots \times I_N) \times I_k}$. Therefore, we need to transpose the unfolded matrix and the modified HOSVD can then be written as:

$$\mathcal{A} = \hat{S} \times_1 V^{(1)} \times_2 V^{(2)} \dots \times_N V^{(N)} \quad (7)$$

where the dimension of the core tensor \hat{S} is $\mathbb{R}^{(I_2 \times I_3 \times \dots \times I_N) \times (I_1 \times I_3 \times \dots \times I_N) \times \dots \times (I_1 \times I_2 \times \dots \times I_{(N-1)})}$.

One can easily verify that the core tensor \hat{S} along with $V^{(k)}$ would perfectly reconstruct the tensor \mathcal{A} .

Because $V^{(k)}$ is an orthogonal matrix, every such matrix has an associated point on a Grassmann manifold. Furthermore, a set of orthogonal matrices $V^{(1)}, V^{(2)}, \dots, V^{(N)}$ forms a set of Grassmann manifolds $\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_N$ where the dimension of each factor manifold is different. In other words, $V^{(k)}$ is the component for a product manifold.

Using the modified HOSVD, we factorize each order of the tensor into a factor manifold. Each factor manifold spans one order of a tensor in a column space whereas the traditional HOSVD spans multiple factors. Hence, the modified HOSVD can have a one-to-one factorization between the order of a tensor and a factor manifold.

4.2. Geodesic Distance on Product Manifolds

It is known that the geodesic in a product manifold \mathcal{M} is the product of geodesics in $\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_N$ [15, 1]. As such, for any differentiable curve γ parametrized by t , we have $\gamma(t) = (\gamma_i(t), \gamma_j(t))$ where γ_i and γ_j are the geodesics on \mathcal{M}_i and \mathcal{M}_j respectively. From this observation, a geodesic distance on a product manifold can be formulated as:

$$d_{\mathcal{M}}(\mathcal{A}, \mathcal{B}) = \|\sin \Theta\|_2 \quad (8)$$

where \mathcal{A} and \mathcal{B} are N order tensors, and $\Theta = (\theta_1, \theta_2, \dots, \theta_N)$ where the canonical angle $\theta_k \in \mathcal{M}_k$ is computed separately on each Grassmann (factor) manifold.

This development of geodesic distance on the product manifold can be related back to our cylinder example where a circle in \mathbb{R}^2 and a line in \mathbb{R}^1 form an open cylinder in \mathbb{R}^3 in a product space. Recall that a Grassmann manifold is a set of p -dimensional linear subspaces. In analogous fashion, the product of a set of p_1, p_2, \dots, p_N linear subspaces forms a set of product subspaces whose dimension is $(p_1 + p_2 + \dots + p_N)$. The product subspaces are the elements on a product manifold. This observation is consistent with the Θ in Equation (8) where the number of canonical angles agrees with the dimension of product subspaces on the product manifold.

Note that canonical angles θ_k are measured between $V_{\mathcal{A}}^{(k)}$ and $V_{\mathcal{B}}^{(k)}$ where each is an orthogonal matrix spanning the row space associated with nonzero singular values from a mode- k matrix. As such, an N order tensor in $\mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ would span N row spaces in I_1, I_2, \dots, I_N , respectively, and the dimension of product subspaces on a product manifold is the sum of each order of a tensor, i.e., $(\sum_{i=1}^N = I_1 + I_2 + \dots + I_N)$.

4.3. Classification on Product Manifolds

Putting all the components together for video classification, we employ a simple nearest neighbor classifier. Let \mathcal{A} and \mathcal{B}_j be 3rd order tensors where \mathcal{A} is a query video and \mathcal{B}_j is a target video. Then, the classification can be performed as follows:

$$j^* = \underset{j \in \text{target}}{\operatorname{argmin}} d_{\mathcal{M}}(\mathcal{A}, \mathcal{B}_j) \quad (9)$$

The tensor representation on a product manifold for a video models the variations in both space and time. Each video is explicitly formulated as multiple effects and the geometry of the space is properly considered. The geodesic distance on a product manifold is not only geometrically sound, but is also, as we will now demonstrate, very useful.

5. Experimental Results

We will test our method on the Cambridge-Gesture database [8]¹ and the KTH human action database [19]². The Cambridge-Gesture database includes nine gesture types. The video frame size is 320×240 and video lengths are diverse. The KTH human action database has six types of actions and is the largest action data set publicly available. Each video frame is scaled to 160×120 and the number of frames for each video sequence also varies.

5.1. Gesture Action Classification

The Cambridge-Gesture database includes 900 video sequences, 100 for each of nine gestures. Each of the 100 videos per gesture class is further broken down into five illuminations (Set1, Set2, Set3, Set4, and Set5) and ten motions from each of two subjects. Examples of these gestures are given in Figure 2.

All video sequences are resized to $20 \times 20 \times 32$. To standardize the video length, we collect the middle 32 frames from a video sequence. Furthermore, no space-time alignment is performed on this data set.

Following the experimental protocol of [8], the data set is partitioned into a number of illumination sets where Set1, Set2, Set3, and Set4 are the test sets, and Set5 is the training set. Furthermore, the training set is randomly divided

¹<http://mi.eng.cam.ac.uk/pub/CamGesData/>

²<http://www.nada.kth.se/cvpa/actions/>



Figure 2. Each row depicts a class of hand gesture actions. (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)).

| | Our Method | TCCA [8] | DCCA [9] |
|-------|------------|----------|----------|
| Set1 | 89% | 81% | 63% |
| Set2 | 86% | 81% | 61% |
| Set3 | 89% | 78% | 65% |
| Set4 | 87% | 86% | 69% |
| Total | 88% | 82% | 65% |

Table 1. Classification rates for gesture action classification on the Cambridge-Gesture database.

into training and validation sets (10 sequences for training and the other 10 sequences for validation). Since we do not perform prior training, we discard the validation set.

The classification results are reported in Table 1 and our method outperforms the current state-of-the-art methods, tensor CCA (TCCA) [8] and discriminative CCA (DCCA) [9], on all illumination data sets³. The classification results for our method and TCCA are further divided into categories and presented in confusion matrices in Figure 3. Each cell in the confusion matrix is the average classification rate from four illumination sets. The confusion matrices show that our method and TCCA handle individual gestures differently, with our method doing better on 5 actions and TCCA doing better on 4 actions. Furthermore, the worst TCCA performance for a gesture is 68%, compared to 78% for our method. The best performance of TCCA for a gesture is 98%, compared to 96% for our method.

5.2. Human Action Classification

The KTH human action data set [19] has six types of human actions including walking, running, jogging, boxing, handwaving, and handclapping. Examples are shown

³We do not include Wong and Cipolla’s results [24] here because their results were reported using the leave-one-out cross validation protocol.

| Our Method | | | | | | | | | | TCCA [8] | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|----|----|
| | FL | FR | FC | SL | SR | SC | VL | VR | VC | | FL | FR | FC | SL | SR | SC | VL | VR | VC |
| FL | 78 | 0 | 0 | 1 | 0 | 15 | 4 | 0 | 3 | FL | 94 | 0 | 0 | 4 | 0 | 0 | 1 | 0 | 0 |
| FR | 0 | 84 | 0 | 0 | 1 | 1 | 0 | 13 | 1 | FR | 0 | 98 | 0 | 2 | 0 | 0 | 0 | 0 | 0 |
| FC | 0 | 0 | 84 | 0 | 0 | 4 | 0 | 0 | 13 | FC | 1 | 0 | 81 | 0 | 0 | 13 | 0 | 0 | 5 |
| SL | 1 | 0 | 0 | 90 | 0 | 1 | 8 | 0 | 0 | SL | 3 | 0 | 0 | 95 | 0 | 0 | 2 | 0 | 0 |
| SR | 0 | 4 | 0 | 0 | 89 | 0 | 0 | 8 | 0 | SR | 0 | 14 | 0 | 0 | 84 | 0 | 0 | 2 | 0 |
| SC | 0 | 0 | 0 | 0 | 0 | 83 | 0 | 0 | 18 | SC | 5 | 0 | 0 | 2 | 0 | 93 | 0 | 0 | 0 |
| VL | 0 | 0 | 0 | 0 | 0 | 0 | 96 | 0 | 4 | VL | 6 | 0 | 0 | 14 | 0 | 0 | 81 | 0 | 0 |
| VR | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 95 | 4 | VR | 1 | 17 | 0 | 1 | 10 | 0 | 4 | 68 | 0 |
| VC | 0 | 0 | 0 | 0 | 0 | 9 | 0 | 0 | 91 | VC | 2 | 0 | 13 | 0 | 0 | 14 | 2 | 1 | 68 |

Figure 3. Confusion matrices for gesture action classification.



Figure 4. Rows from top to bottom are examples of walking, running, jogging, boxing, handwaving and handclapping.

in Figure 4. Each type of human actions is performed by 25 people with four different scenarios: outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3), and indoors (s4). Similar to Kim and Cipolla’s settings [8], we first perform space-time alignment (location and frame cropping) on the human action videos manually⁴. Next, all video sequences are resized to $20 \times 20 \times 32$. In order to standardize to a length of 32 frames, we take the middle 32 for longer sequences, and recycle frames for videos shorter than 32 frames.

To facilitate comparison with prior work, performance on human action classification is evaluated using two protocols. The first is proposed by Schüldt *et al.* [19]. The KTH human action data set is divided into three subsets with different people: training set (8 persons), validation set (8 persons), and test set (9 persons). Like the gesture experiment, we discard the validation set because no prior training is required for the proposed method. The training set is further divided into four groups of $\{s1\}$, $\{s1, s4\}$, $\{s1, s3, s4\}$, and $\{s1, s2, s3, s4\}$. The test set is always $\{s1, s2, s3, s4\}$.

⁴The action is repeated several times on each original video.

| Average Accuracy = 96.0% | | | | | | | Average Accuracy = 91.8% | | | | | | | Average Accuracy = 71.7% | | | | | | |
|--------------------------|------|-----|------|------|------|------|--------------------------|------|------|------|------|------|------|--------------------------|------|------|------|------|------|------|
| | Walk | Jog | Run | Box | Hclp | Hwav | | Walk | Jog | Run | Box | Hclp | Hwav | | Walk | Jog | Run | Box | Hclp | Hwav |
| Walk | 97.9 | 0 | 2.1 | 0 | 0 | 0 | Walk | 99.0 | 1.0 | 0 | 0 | 0 | 0 | Walk | 83.8 | 16.2 | 0 | 0 | 0 | 0 |
| Jog | 0 | 100 | 0 | 0 | 0 | 0 | Jog | 4.0 | 89.0 | 7.0 | 0 | 0 | 0 | Jog | 22.9 | 60.4 | 16.7 | 0 | 0 | 0 |
| Run | 4.8 | 2.1 | 91.0 | 0 | 2.1 | 0 | Run | 1.0 | 19 | 80.0 | 0 | 2.8 | 0 | Run | 6.3 | 38.9 | 54.9 | 0 | 0 | 0 |
| Box | 0 | 0 | 0 | 97.2 | 2.8 | 0 | Box | 0 | 0 | 0 | 97.0 | 0 | 3.0 | Box | 0.7 | 0 | 0 | 97.9 | 0.7 | 0.7 |
| Hclp | 0 | 0 | 0 | 0 | 100 | 0 | Hclp | 0 | 0 | 0 | 5.0 | 95.0 | 0.7 | Hclp | 1.4 | 0 | 0 | 35.4 | 59.7 | 3.5 |
| Hwav | 0 | 0 | 0 | 0.7 | 10.4 | 89.6 | Hwav | 0 | 0 | 0 | 0 | 9.0 | 91.0 | Hwav | 0.7 | 0 | 0 | 20.8 | 4.9 | 73.6 |

Figure 5. Confusion matrices for human action classification (Schüldt’s protocol): Left (Our Method), Middle (BOF + SVM) [11], Right (LF + SVM) [19].

| | Our Method | TCCA [8] | DCCA [9] | STIP [24] | STW [17] |
|--------------|------------|----------|----------|-----------|----------|
| Walking | 98% | 99% | 100% | 88% | 82% |
| Jogging | 99% | 90% | 80% | 75% | 53% |
| Running | 97% | 88% | 68% | 77% | 88% |
| Boxing | 97% | 98% | 97% | 92% | 98% |
| Handclapping | 98% | 100% | 99% | 100% | 86% |
| Handwaving | 95% | 97% | 99% | 88% | 93% |
| Total | 97% | 95% | 90% | 87% | 83% |

Table 2. Classification rates for human action classification on the KTH human action database (Leave-one-out cross validation).

The results are presented in the confusion matrices shown in Figure 5. Each cell in the confusion matrix is the average result from the four training groups. Our results are in the left confusion matrix, and the bag-of-features SVM (BOF + SVM) [11] and the local feature SVM (LF+SVM) [19] are presented in middle and on the right, respectively. As Figure 5 demonstrates, our method improves the classification accuracies in all categories except walking and boxing. Overall, our approach achieves 96% average classification rate whereas the BOF+SVM obtains 91.8% and the LF+SVM gets 71.7%.

The second experiment protocol, used by [8, 24, 17], is leave-one-out (LOO) cross validation. The classification results using LOO for the KTH human actions are reported in Table 2. The first thing to note in Table 2 is that no algorithm is universally best. In terms of top classification rates, DCCA, TCCA, and our method are each best for two of the six actions. However, when our method is better, it is typically by a larger amount, and this is reflected in the higher overall average classification rate of 97% versus 95% for TCCA and 90% for DCCA.

Looking at the results for both protocols on the KTH human action data set, our method achieves the highest overall classification rate. Interestingly, it is notable that the two actions most commonly confused by the other approaches [11, 19, 8, 9, 24, 17] are jogging and running, and our method is able to reduce this ambiguity greatly.

6. Discussion

The methods tested in this paper can be logically divided into two categories. They are feature-based methods [11, 19, 24, 17], and pixel-based methods [8, 9]. Our approach is pixel-based. Conceptually, pixel-based methods are simpler because they do not need additional human intervention and/or machine learning algorithms to identify the features, and a feature detector to locate the landmarks. Therefore, they are arguably easier to apply to new action classification problems.

All the algorithms that we compared in Section 5 require prior training. Note the LOO protocol in particular, as exemplified by the results in Table 2, works strongly in favor of highly trained methods by maximizing the available training data. Of course, in practice large amounts of training data are not always available. Whenever training is utilized, the opportunity arises for performance to degrade if there is a mismatch between training and operational data. Because our method depends upon the intrinsic geometry of the videos expressed through product manifolds, no prior training is involved.

Like many other pixel-based methods, our method may be sensitive to background clutter. There are algorithms for segmenting moving objects from cluttered background, and future work will evaluate our method in conjunction with video segmentation.

7. Conclusions

This paper demonstrates that the underlying geometry of a video is an important feature for action classification. We represent a video as a 3rd order tensor and map it to a product manifold where each component is a Grassmannian. The realization of points on these Grassmannians is achieved by applying the modified HOSVD to a tensor representation of the action video. A natural metric is inherited from the factor manifolds since the geodesic on the product manifold is given by the product of the geodesic on the Grassmann manifolds.

This composite geodesic distance is formulated and applied to the problem of action classification. Experimental results show that our method performs very well on two public video data sets and hence underscores the practical importance of the product manifold geometry. Finally, our method is generic insofar as no prior training is required, and no parameters need tuning. The matching time is also fast. With a non-optimized MATLAB implementation, each match takes about 8 milliseconds on a standard modern computer.

The proposed approach provides a basic metric for video classification and it is easy to combine with more advanced classifiers. Our future work will extend the product manifold representation to other video recognition problems such as gait recognition.

References

- [1] E. Begelfor and M. Werman. Affine invariance revisited. In *IEEE Conference on Computer Vision and Pattern Recognition, New York*, 2006.
- [2] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto. Recognition of human gaits. In *IEEE Conference on Computer Vision and Pattern Recognition, Hawaii*, 2001.
- [3] A. Björck and G. Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, pages 579–594, 1973.
- [4] J. Conway, R. Hardin, and N. Sloane. Packing lines, planes, etc.: Packings in grassmannian spaces. *Experimental Mathematics*, 5(2):139–159, 1996.
- [5] A. Datta, Y. Sheikh, and T. T. Kanade. Modeling the product manifold of posture and motion. In *Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (in conjunction with ICCV)*, 2009.
- [6] A. Edelman, R. Arias, and S. Smith. The geometry of algorithms with orthogonal constraints. *SIAM J. Matrix Anal. Appl.*, (2):303–353, 1999.
- [7] L. Eldén and B. Savas. A newton-grassmann method for computing the best multilinear rank-(r_1 , r_2 , r_3) approximation of a tensor. *SIAM J. Matrix Anal. Appl.*, (2):248–271, 2009.
- [8] 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*, 31(8):1415–1428, 2009.
- [9] T.-K. Kim, J. Kittler, and R. Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1–14, 2007.
- [10] T. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, (3):455–500, 2009.
- [11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition, Alaska*, 2008.
- [12] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21:1253–1278, 2000.
- [13] X. Li, W. Hu, Z. Zhang, X. Zhang, and G. Luo. Robust visual tracking based on incremental tensor subspace learning. In *IEEE International Conference on Computer Vision*, 2007.
- [14] Y. M. Lui, J. R. Beveridge, B. A. Draper, and M. Kirby. Image-set matching using a geodesic distance and cohort normalization. In *IEEE International Conference on Automatic Face and Gesture Recognition, Amsterdam, The Netherlands*, 2008.
- [15] Y. Ma, J. Košecák, and S. Sastry. Optimal motion from image sequences: A riemannian viewpoint, 1998. Technical Report No. UCB/ERL M98/37, EECS Department, University of California, Berkeley.
- [16] T. Moeslund and E. Granum. A survey of computer vision based human motion capture. *Computer Vision and Image Understanding*, 81:231–268, 2001.
- [17] J. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318, 2008.
- [18] T. Schoenemann, F. R. Schmidt, and D. Cremers. Image segmentation with elastic shape priors via global geodesics in product spaces. In *British Machine Vision Conference 2008, Leeds, U.K.*, 2008.
- [19] C. Schödl, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *International Conference on Pattern Recognition, Cambridge, UK*, 2004.
- [20] A. Shaji, S. Chandran, and D. Suter. Manifold optimisation for motion factorisation. In *IEEE International Conference on Pattern Recognition, Florida*, 2008.
- [21] P. Turaga, A. Veeraraghavan, and R. Chellappa. Statistical analysis on stiefel and grassmann manifolds with applications in computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
- [22] M. A. O. Vasilescu. Human motion signatures: Analysis, synthesis, recognition. In *International Conference on Pattern Recognition, Quebec City, Canada*, 2002.
- [23] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorface. In *European Conference on Computer Vision, Copenhagen, Denmark*, 2002.
- [24] S.-F. Wong and R. Cipolla. Extracting spatiotemporal interest points using global information. In *IEEE International Conference on Computer Vision*, 2007.