Elastic Functional Coding of Riemannian Trajectories

Rushil Anirudh, Pavan Turaga, Jingyong Su, Anuj Srivastava

Abstract—Visual observations of dynamic phenomena, such as human actions, are often represented as sequences of smoothly-varying features . In cases where the feature spaces can be structured as Riemannian manifolds, the corresponding representations become trajectories on manifolds. Analysis of these trajectories is challenging due to non-linearity of underlying spaces and high-dimensionality of trajectories. In vision problems, given the nature of physical systems involved, these phenomena are better characterized on a low-dimensional manifold compared to the space of Riemannian trajectories. For instance, if one does not impose physical constraints of the human body, in data involving human action analysis, the resulting representation space will have highly redundant features. Learning an effective, low-dimensional embedding for action representations will have a huge impact in the areas of search and retrieval, visualization, learning, and recognition. Traditional manifold learning addresses this problem for static points in the Euclidean space, but its extension to Riemannian trajectories is non-trivial and remains unexplored. The difficulty lies in inherent non-linearity of the domain and temporal variability of actions that can distort any traditional metric between trajectories. To overcome these issues, we use the framework based on transported square-root velocity fields (TSRVF); this framework has several desirable properties, including a rate-invariant metric and vector space representations. We propose to learn an embedding such that each action trajectory is mapped to a single point in a low-dimensional Euclidean space, and the trajectories that differ only in temporal rates map to the same point. We utilize the TSRVF representation, and accompanying statistical summaries of Riemannian trajectories, to extend existing coding methods such as PCA, KSVD and Label Consistent KSVD to Riemannian trajectories or more generally to Riemannian functions. We show that such coding efficiently captures trajectories in applications such as action recognition, stroke rehabilitation, visual speech recognition, clustering and diverse sequence sampling. Using this framework, we obtain state-of-the-art recognition results, while reducing the dimensionality/complexity by a factor of $100 - 250\times$. Since these mappings and codes are invertible, they can also be used to interactively visualize Riemannian trajectories and synthesize actions.

Index Terms—Riemannian Geometry, Activity Recognition, Dimensionality Reduction, Visualization.

1 Introduction

There have been significant advances in understanding differential geometric properties of image and video features in vision and robotics. Examples include activity recognition [39], [9], [44], medical image analysis [15], and shape analysis [33]. Some of the popular non-Euclidean features used for activity analysis include shape silhouettes on the Kendall's shape space [43], pairwise transformations of skeletal joints on $SE(3) \times SE(3) \cdots \times SE(3)$ [44], representing the parameters of a linear dynamical system as points on the Grassmann manifold [39], and histogram of oriented optical flow (HOOF) on a hyper-sphere [9]. A commonly occurring theme in many applications is the need to *represent, compare, and manipulate* such representations in a manner that respects certain constraints.

One such constraint is the geometry of such features, since they do not obey conventional Euclidean properties. Another constraint for temporal data such as human actions is the need for speed invariance or *warping*, which causes two sequences to be mis-aligned in time inducing unwanted distortions in the distance metric. Figure 1 shows the effects of ignoring warping, in the context of human actions. Accounting for warping reduces the intra-class distance and improves the inter-class distance. Consequently, statistical quantities such as the *mean sequence* are distorted as seen in figure 1 for two actions S_1 and S_2. Such effects can cause significant performance losses when using building class templates, without accounting for the changes in speed. The most common way to solve for the mis-alignment problem is to use dynamic time warping (DTW) which originally found its use in speech processing [6]. For human actions, [42], [51] address this problem using different strategies for features in the Euclidean space. However, DTW behaves as

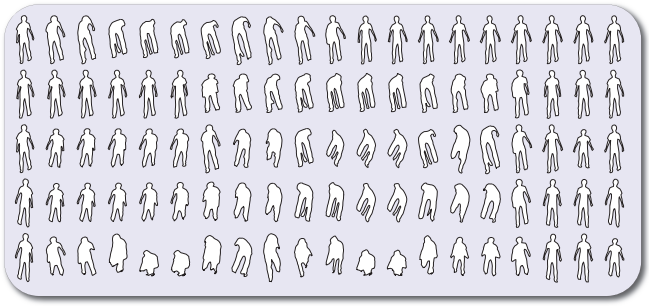


Fig. 1: Row wise from top – S_1, S_2, Warped action $\tilde{S}_2$, Warped mean, Unwarped mean. The TSRVF can enable more accurate estimation of statistical quantities such as average of two actions S_1, S_2.

- *R. Anirudh and P. Turaga are with the School of Arts, Media, & Engineering and Department of Electrical, Computer, and Energy Engineering at Arizona State University, Tempe, AZ. E-mail: {ranirudh,pturaga}@asu.edu*
- *Jingyong Su is with the Department of Mathematics & Statistics at Texas tech University, Lubbock, TX. E-mail: jingyong.su@ttu.edu*
- *Anuj Srivastava is with the Department of Statistics at Florida State University, Tallahasse, FL. E-mail:anuj@stat.fsu.edu*

Manuscript received XX, 201X; revised XX, 201X.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2564409, IEEE Transactions on Pattern Analysis and Machine Intelligence

a similarity measure instead of a true distance metric in that it does not naturally allow the estimation of statistical measures such as mean and variance of action trajectories. We seek a representation that is highly discriminative of different classes while factoring out temporal warping to reduce the variability within classes, while also enabling low dimensional coding at the sequence level.

Learning such a representation is complicated when the features extracted are non-Euclidean (i.e. they do not obey conventional properties of the Euclidean space). Finally, typical representations for action recognition tend to be extremely high dimensional in part because the features are extracted per-frame and stacked. Any computation on such non-Euclidean trajectories can become very easily involved. For example, a recently proposed skeletal representation [44] results in a 38220 dimensional vector for a 15 joint skeletal system when observed for 35 frames. Such features do not take into account, the physical constraints of the human body, which translates to giving varying degrees of freedom to different joints. It is therefore a reasonable assumption to make that the *true* space of actions is much lower dimensional. This is similar to the argument that motivated manifold learning for image data, where the number of observed image pixels maybe extremely high dimensional, but the object or scene is often considered to lie on a lower dimensional manifold. A lower dimensional embedding will provide a robust, computationally efficient, and intuitive framework for analysis of actions. In this paper, we address these issues by studying the statistical properties of trajectories on Riemannian manifolds to extract lower dimensional representations or codes. We propose a general framework to *code* Riemannian trajectories in a speed invariant fashion that generalizes to many manifolds, the general idea is presented in figure 2. We validate our work on three different manifolds - the Grassmann manifold, the product space $SE(3) \times \cdots \times SE(3)$, and the space of SPD matrices.

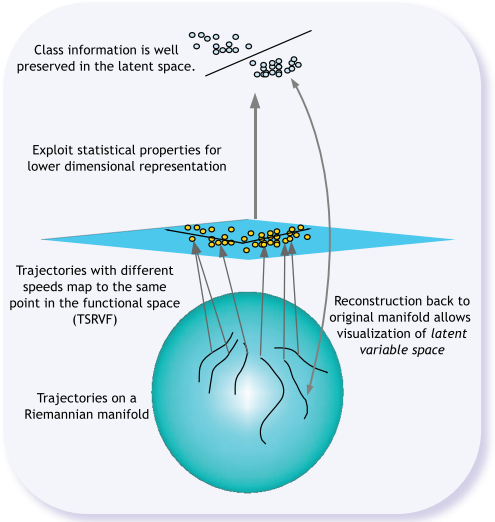


Fig. 2: Dimensionality Reduction for Riemannian Trajectories

Elastic representations for Riemannian trajectories is relatively new and the lower dimensional embedding of such sequences has remained unexplored. We employ the transport square-root velocity function (TSRVF) representation – a recent development in statistics [35], to provide a warp invariant representation to the Riemannian trajectories. The TSRVF is also advantageous as it provides a functional representation that is Euclidean. Exploiting this we propose to learn the low dimensional embedding with a Riemannian functional variant of popular coding techniques. In other words, we are interested in parameterization of Riemannian trajectories, i.e. for N actions $A_i(t), i = 1 \dots N$, our goal is to learn $\mathcal{F}$ such that $\mathcal{F}(x) = A_i$ where $x \in \mathbb{R}^k$ is the set of parameters. Such a model will allow us to compare actions by simply comparing them in their parametric space with respect to $\mathcal{F}$, with significantly faster distance computations, while being able to reconstruct the original actions. In this work, we learn two different kinds of functions using PCA and dictionary learning, which have attractive properties for recognition and visualization.

Broader impact: While one advantage of embedding Riemannian trajectories into a lower dimensional space is the low cost of storage and transmission, perhaps the biggest advantage is the reduction in complexity of search and retrieval in the latent spaces. Although this work concerns itself primarily with recognition and reconstruction, it is easy to see the opportunities these embeddings present in search applications given that the search space dimension is now $\sim 250\times$ smaller. We conclusively demonstrate that the embeddings are as discriminative as their original features, therefore guaranteeing an accurate and fast search. The proposed coding scheme also enables visualization of highly abstract properties of human movement in an intuitive manner. We show results on a stroke rehabilitation project which allows us to visualize the *quality* of movement for stroke survivors. These ideas present a lot of opportunity towards building applications that provide users with feedback, while facilitating rehabilitation. We summarize our contributions next.

Contributions

1) An elastic vector-field representation for Riemannian trajectories by modeling the TSRVF on the Grassmann manifold, the product space of $SE(3) \times .. \times SE(3)$ and the space of symmetric positive definite matrices (SPD).
2) Dimensionality reduction for Riemannian trajectories in a speed invariant manner, such that each trajectory is mapped to a single point in the low dimensional space.
3) We present results on three coding techniques that have been generalized for Riemannian Functionals (RF) - PCA, KSVD [2] and Label Consistent KSVD [21].
4) We show the application of such embedded features or codes in three applications - action recognition, visual speech recognition, and stroke rehabilitation outperforming all comparable baselines, while being nearly $100 - 250\times$ more compressed. Their effectiveness is also demonstrated in action clustering and diverse action sampling.
5) The low dimensional codes can be used for visualization of Riemannian trajectories to explore the

latent space of human movement. We show that these present interesting opportunities for stroke rehabilitation.

6) We perform a thorough analysis of the TSRVF representation testing its stability under different conditions such as noise, length of trajectories and its impact on convergence.

1.1 Organization

A preliminary version of this work appeared in [4], with application to human activity analysis. In this work we generalize the idea significantly, by considering new applications, new coding methods, and an additional manifold. We also provide a detailed discussion on design choices, parameters and potential weaknesses. We begin in section 2 with a review of related techniques and ideas that provide more context to our contributions. In section 3 we describe the mathematical tools and geometric properties of the various manifolds considered in this work. Section 4 introduces the TSRVF and discusses its speed invariance properties. In section 5, we propose the algorithm to perform functional coding, specifically for PCA, K-SVD [2], and Label Consistent K-SVD [21]. We experimentally validate our low dimensional codes in section 6 on three applications - human activity recognition, visual speech recognition and stroke rehabilitation. We also show applications in visualization, clustering, and diverse sampling for Riemannian trajectories. Section 7 contains experiments that test the stability and robustness of the TSRVF and the coded representations of Riemannian trajectories under conditions such as noise, and different sampling rates. We also report its convergence rates. We conclude our work and present future directions of research in section 8.

2 Related Work

2.1 Elastic metrics for trajectories

The TSRVF is a recent development in statistics [35] that provides a way to represent trajectories on Riemannian manifolds such that the distance between two trajectories is invariant to identical time-warpings. The representation itself lies on a tangent space and is therefore Euclidean, this is discussed further in section 4. The representation was then applied to the problem of visual speech recognition by warping trajectories on the space of SPD matrices [36]. A more recent work [49] has addressed the arbitrariness of the reference point in the TSRVF representation, by developing a purely intrinsic approach that redefines the TSRVF at the starting point of each trajectory. A version of the representation for Euclidean trajectories - known as the Square-Root Velocity Function (SRVF), was recently applied to skeletal action recognition using joint locations in $\mathbb{R}^3$ with promising results [12]. We differentiate our contribution as the first to use the TSRVF representation by representing actions as trajectories in high dimensional non-linear spaces. We use the skeletal feature recently proposed in [44], which models each skeleton as a point on the space of $SE(3) \times \cdots \times SE(3)$. Rate invariance for activities has been addressed before [42], [51], for example [42] models the space of all possible warpings of an action sequence. Such techniques can align sequences correctly, even when features are multi-modal [51]. However, most of the techniques are used for recognition which can be achieved with a similarity measure, but we are interested in a representation which serves a more general purpose to 1) provide an effective metric for comparison, recognition, retrieval, etc. and 2) provide a framework for efficient lower dimensional coding which also enables recovery back to the original feature space.

2.2 Low dimensional data embedding

Principal component analysis has been used extensively in statistics for dimensionality reduction of linear data. It has also been extended to model a wide variety of data types. For high dimensional data in $\mathbb{R}^n$, manifold learning (or non-linear dimensionality reduction) techniques [37], [29] attempt to identify the underlying low dimensional manifold while preserving specific properties of the original space. Using a robust metric, one could theoretically use such techniques for coding, but the algorithms have impractical memory requirements for very high dimensional data of the order of $\sim 10^4 - 10^5$, they also do not provide a way of reconstructing the original manifold data. For data already lying on a known manifold, geometry aware mapping of SPD matrices [17] constructs a lower-dimensional SPD manifold, and principal geodesic analysis (PGA) [15] identifies the primary geodesics along which there is maximum variability of data points. We are interested in identifying the variability of sequences instead. Recently, dictionary learning methods for data lying on Riemannian manifolds have been proposed [20], [18] and could potentially be used to code sequential data but they can be expected to be computationally more intensive. Coding data on Riemannian manifolds is still a new idea with some progress in the past few years, for example recently the Vector of Locally Aggregated Descriptors (VLAD) has also been extended recently to Riemannian manifolds [14]. However, to the best of our knowledge, coding Riemannian trajectories has received little or no attention, but has several attractive advantages.

Manifold learning of Trajectories: Dimensionality reduction for high dimensional time series is still a relatively new area of research, some recent works have addressed the issue of defining spatial and temporal neighborhoods. For example, [24] recently proposed a generalization of Laplacian eigenmaps to incorporate temporal information. Here, the neighborhoods are also a function of time, but the final reduction step still involves mapping a single point in the high dimensional space to a single point in the lower dimensional space. Next, the Gaussian process latent variable model (GPLVM) [23] and its variants, are a set of techniques that perform non-linear dimensionality reduction for data in $\mathbb{R}^N$, while allowing for reconstruction back to the original space. However, its generalization to non-linear Riemannian trajectories is unclear, which is the primary concern of this work. Quantization of Riemannian trajectories has been addressed in [3], which reduces dimensionality but does not enable visualization. Further, there is loss of information which can cause reduction in recognition performance, whereas we propose to reduce

dimensionality by exploiting the latent variable structure of the data. Comparing actions in the latent variable space is similar in concept to learning a linear dynamical system [39] for Euclidean data, where different actions can be compared in the parametric space of the model.

2.3 Visualization in Biomedical Applications

A promising application for the ideas proposed here, is in systems for rehabilitation of patients suffering from impairment of their motor function. Typically visual sensors are used to record and analyze the movement, which drives feedback. An essential aspect of the feedback is the idea of decomposing human motion into its individual components. For example, they can be used to understand abstract ideas such as movement quality [11], gender styles [38] etc. Troje [38] proposed to use PCA on individual body joints in $\mathbb{R}^3$, to model different styles of the walking motion. However, they work with data in the Euclidean space, and explicitly model the temporality of movement using a combination of sinusoids at different frequencies. More recently, a study in neuroscience [11] showed that the perceived space of movement in the brain is inherently non-linear and that visualization of different movement attributes can help achieve the *most efficient* movement between two poses. This efficient movement is known to be the geodesic in the *pose space* [7]. The study was validated on finger tapping, which is a much simpler motion than most human actions. In this work, we generalize these ideas by visualizing entire trajectories of much more complicated systems such as human skeletons and show results on the movement data of stroke-patients obtained from a motion-capture based hospital system [10].

3 Mathematical Preliminaries

In this section we will briefly introduce the properties of manifolds of interest namely – the product space $SE(3) \times \cdots \times SE(3)$, the Grassmann manifold, and the space of symmetric positive definite (SPD) matrices. For an introduction to Riemannian manifolds, we refer the reader to [1], [8].

3.1 Product Space of the Special Euclidean Group

For action recognition, we represent a stick figure as a combination of relative transformations between joints, as proposed in [44]. The resulting feature for each skeleton is interpreted as a point on the product space of $SE(3) \times \cdots \times SE(3)$. The skeletal representation explicitly models the 3D geometric relationships between various body parts using rotations and translations in 3D space [44]. These transformation matrices lie on the curved space known as the Special Euclidean group $SE(3)$. Therefore the set of all transformations lies on the product space of $SE(3) \times \cdots \times SE(3)$.

The special Euclidean group, denoted by $SE(3)$ is a Lie group, containing the set of all 4×4 matrices of the form

$$P(R, \overrightarrow{v}) = \begin{bmatrix} R & \overrightarrow{v} \\ 0 & 1 \end{bmatrix}, \tag{1}$$

where R denotes the rotation matrix, which is a point on the special orthogonal group $SO(3)$ and $\overrightarrow{v}$ denotes the translation vector, which lies in $\mathbb{R}^3$. The 4×4 identity matrix I_4 is an element of $SE(3)$ and is the identity element of the group. The tangent space of $SE(3)$ at I_4 is called its Lie algebra – denoted here as $\mathfrak{se}(3)$. It can be identified with 4×4 matrices of the form[1]

$$\widehat{\xi} = \begin{bmatrix} \widehat{\omega} & \overrightarrow{v} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega 1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{2}$$

where $\widehat{\omega}$ is a 3×3 skew-symmetric matrix and $\overrightarrow{v} \in \mathbb{R}^3$. An equivalent representation is $\xi = [\omega_1, \omega_2, \omega_3, v_1, v_2, v_3]^T \in \mathbb{R}^6$. For the exponential and inverse exponential maps, we use the expressions provided on p. 413-414 in [26], we reproduce them for completeness here.
The exponential map is given by

$$\exp \widehat{\xi} = \begin{bmatrix} I & \overrightarrow{v} \\ 0 & 1 \end{bmatrix} \quad \omega = 0 \text{ and } \exp \widehat{\xi} = \begin{bmatrix} e^{\widehat{\omega}} & A\overrightarrow{v} \\ 0 & 1 \end{bmatrix} \quad \omega \neq 0, \tag{3}$$

where $e^{\widehat{\omega}}$ is given explicitly by the Rodrigues's formula – $= I + \frac{\widehat{\omega}}{\|\omega\|}\sin\|\omega\| + \frac{\widehat{\omega}^2}{\|\omega\|^2}(1 - \cos\|\omega\|)$, and $A = I + \frac{\widehat{\omega}}{\|\omega\|^2}(1 - \cos\|\omega\|) + \frac{\widehat{\omega}^2}{\|\omega\|^3}(\|\omega\| - \sin\|\omega\|)$.
The inverse exponential map is given by

$$\widehat{\xi} = \log \begin{bmatrix} R & \overrightarrow{v} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \widehat{\omega} & A^{-1}\overrightarrow{v} \\ 0 & 0 \end{bmatrix}, \tag{4}$$

where $\widehat{\omega} = \log R$, and

$$A^{-1} = I - \frac{1}{2}\widehat{\omega} + \frac{2\sin\|\omega\| - \|\omega\|(1 + \cos\|\omega\|)}{2\|\omega\|^2\sin\|\omega\|}\widehat{\omega}^2 \quad \omega \neq 0,$$

when $\omega = 0$, then $A = I$.

Parallel transport on the product space is the parallel transport of the point on component spaces. Let $T_O(SO(3))$ denote the tangent space at $O \in SO(3)$, then the parallel transport of a $W \in T_O(SO(3))$ from O to $I_{3\times 3}$ is given by $O^T W$. For more details on the properties of the special Euclidean group, we refer the interested reader to [26].

3.2 The space of Symmetric Positive Definite (SPD) matrices

We utilize the covariance features for the problem of Visual Speech Recognition (VSR). These features first introduced in [41] have become very popular recently due to their ability to model unstructured data from images such as textures and scenes. A covariance matrix of image features such as pixel locations, intensity and their first and second derivatives is constructed to represent the image. As described in [28], for a rectangular region R, let $\{z_k\}_{k=1\ldots n}$ be the d-dimensional feature vector of the points inside R. The covariance matrix for R is given by $C_R = \frac{1}{n-1}\sum_{k=1}^{n}(z_k - \mu)(z_k - \mu)^T$. The Riemannian structure of the space of covariance matrices is studied as the space of non-singular, symmetric positive definite matrices [28]. Let $\tilde{\mathcal{P}}(d)$ be the space of $d \times d$ SPD matrices and $\mathcal{P}(d) = \{P | P \in \tilde{\mathcal{P}}(d) \text{ and } \det(P) = 1\}$. The space $\mathcal{P}(d)$ is a well known symmetric Riemannian manifold,

1. We are following the notation to denote the vector space ($\xi \in \mathbb{R}^6$) and the equivalent Lie algebra representation ($\widehat{\xi} \in \mathfrak{se}(3)$) as described in p. 411 of [26].

it is the quotient of the special linear group $SL(d) = \{G \in GL(d) | \det(G) = 1\}$ by its closed subgroup $SO(d)$ acting on the right and with an $SL(d)$ invariant metric [22]. Although several metrics have been proposed for this space, few qualify as Riemannian metrics, we use the metrics defined in [34] since the expression for parallel transport is readily available. The Lie algebra of $\mathcal{P}(d)$ is $\mathcal{T}_I(\mathcal{P}(d)) = \{A | A^T = A \text{ and } \text{trace}(A) = 0\}$, where I denotes the $d \times d$ identity matrix and the inner product on $\mathcal{T}_I(\mathcal{P}(d))$ is $\langle A, B \rangle = \text{trace}(AB^T)$. The tangent space at $P \in \mathcal{P}(d)$ is $\mathcal{T}_P(\mathcal{P}(d)) = \{PA | A \in \mathcal{T}_I(\mathcal{P}(d))\}$ and $\langle PA, PB \rangle = \text{trace}(AB^T)$. The exponential map is given as $P \in \mathcal{P}(d)$ and $V \in \mathcal{T}_P(\mathcal{P}(d)), \exp_P(V) = \sqrt{P e^{2(P^{-1})V} P}$. The inverse exponential map: For any $P_1, P_2 \in \mathcal{P}(d), \exp_{P_1}(P_2) = P_1 log\left(\sqrt{P_1{}^{-1} P_2^2 P_1{}^{-1}}\right)$. Finally, for any $P_1, P_2 \in \mathcal{P}(d)$, the parallel transport of $V \in \mathcal{T}_P(\mathcal{P}(d))$ from $P_1 \rightarrow P_2$ is $P_2 T_{12}^T B T_{12}$, where $B = P_1^{-1} V, T_{12} = P_{12}^{-1} P_1^{-1} P_2$ and $P_{12} = \sqrt{P_1^{-1} P_2^2 P_1^{-1}}$.

3.3 Grassmann Manifold as a Shape Space

To visualize the action space, we also use shape silhouettes of an actor for different activities. These are interpreted as points on the Grassmann manifold. To obtain a shape as a point, we first obtain a landmark representation of the silhouette by uniformly sampling the shape. Let $L = [(x_1, y_1), (x_2, y_2) \ldots, (x_m, y_m)]$ be an $m \times 2$ matrix that defines m points on the silhouette whose centroid has been translated to zero. The affine shape space [16] is useful to remove small variations in camera locations or the pose of the subject. Affine transforms of a base shape L_{base} can be expressed as $L_{affine}(A) = L_{base} A^T$, and this multiplication by a full rank matrix on the right preserves the column-space of the matrix, L_{base}. Thus the 2D subspace of $\mathbb{R}^m$ spanned by the columns of the shape, i.e. $span(L_{base})$ is invariant to affine transforms of the shape. Subspaces such as these can be identified as points on a Grassmann manifold [5], [40].

Denoted by, $\mathcal{G}_{k,m-k}$, the Grassmann manifold is the space whose points are k-dimensional hyperplanes (containing the origin) in $\mathbb{R}^m$. An equivalent definition of the Grassmann manifold is as follows: To each k-plane, ν in $\mathcal{G}_{k,m-k}$ corresponds a unique $m \times m$ orthogonal projection matrix, P which is idempotent and of rank k. If the columns of a tall $m \times k$ matrix Y spans ν then $YY^T = P$. Then the set of all possible projection matrices $\mathbb{P}$, is diffeomorphic to $\mathcal{G}$. The identity element of $\mathbb{P}$ is defined as $Q = diag(I_k, 0_{m-k,m-k})$, where $0_{a,b}$ is an $a \times b$ matrix of zeros and I_k is the $k \times k$ identity matrix. The Grassmann manifold $\mathcal{G}$(or $\mathbb{P}$) is a quotient space of the orthogonal group, $O(m)$. Therefore, the geodesic on this manifold can be made explicit by lifting it to a particular geodesic in $O(m)$ [32]. Then the tangent, X, to the lifted geodesic curve in $O(m)$ defines the velocity associated with the curve in $\mathbb{P}$. The tangent space of $O(m)$ at identity is $o(m)$, the space of $m \times m$ skew-symmetric matrices, X. Moreover in $o(m)$, the Riemannian metric is just the inner product of $\langle X_1, X_2 \rangle$ = trace($X_1 X_2^T$) which is inherited by $\mathbb{P}$ as well.

The geodesics in $\mathbb{P}$ passing through the point Q (at time t = 0) are of the type $\alpha : (-\epsilon, \epsilon) \mapsto \mathbb{P}, \alpha(t) = \exp(tX) Q \exp(-tX)$, where X is a skew-symmetric matrix belonging to the set M where

$$M = \left\{ \begin{bmatrix} 0 & A \\ -A^T & 0 \end{bmatrix} : A \in \mathbb{R}^{k,n-k} \right\} \subset o(m) \tag{5}$$

Therefore the geodesic between Q and any point P is completely specified by an $X \in M$ such that exp(X)Qexp(-X) = P. We can construct a geodesic between any two points $P_1, P_2 \in \mathbb{P}$ by rotating them to Q and some $P \in \mathbb{P}$. Readers are referred to [32] for more details on the exponential and logarithmic maps of $\mathcal{G}_{k,m-k}$.

4 Rate Invariant Sequence Comparison

In this section we describe the Transport Square Root Velocity Function (TSRVF), recently proposed in [35] as a representation to perform warp invariant comparison between multiple Riemannian trajectories. Using the TSRVF representation for human actions, we propose to learn the latent function space of these Riemannian trajectories in a much lower dimensional space. As we demonstrate in our experiments, such a mapping also provides some robustness to noise which is essential when dealing with noisy sensors.

Let α denote a smooth trajectory on $\mathcal{M}$ and let $\mathbb{M}$ denote the set of all such trajectories: $\mathbb{M} = \{\alpha : [0,1] \mapsto \mathcal{M} | , \alpha \text{ is smooth}\}$. Also define Γ to be the set of all orientation preserving diffeomorphisms of [0,1]: $\Gamma = \{\gamma \mapsto [0,1] | \gamma(0) = 0, \gamma(1) = 1, \gamma \text{ is a diffeomorphism}\}$. It is important to note that γ forms a group under the composition operation. If α is a trajectory on $\mathcal{M}$, then $\alpha \circ \gamma$ is a trajectory that follows the same sequence of points as α but at the evolution rate governed by γ. The group Γ acts on $\mathbb{M}$, $\mathbb{M} \times \Gamma \rightarrow \mathbb{M}$, according to $(\alpha, \gamma) = \alpha \circ \gamma$. To construct the TSRVF representation, we require a formulation for parallel transporting a vector between two points $p, q \in \mathcal{M}$, denoted by $(v)_{p \rightarrow q}$. For cases where p and q do not fall in the cut loci of each other, the geodesic remains unique, and therefore the parallel transport is well defined.

The TSRVF [35] for a smooth trajectory $\alpha \in \mathbb{M}$ is the parallel transport of a scaled velocity vector field of α to a reference point $c \in M$ according to:

$$h_\alpha(t) = \begin{cases} \frac{\dot{\alpha}(t)_{\alpha(t) \mapsto c}}{\sqrt{|\dot{\alpha}(t)|}} \in T_c(\mathcal{M}), & |\dot{\alpha}(t)| \neq 0 \\ 0 \in T_c(\mathcal{M}) & |\dot{\alpha}(t)| = 0 \end{cases} \tag{6}$$

where $|\ .\ |$ denotes the norm related to the Riemannian metric on $\mathcal{M}$ and $T_c(\mathcal{M})$ denotes the tangent space of $\mathcal{M}$ at c. Since α is smooth, so is the vector field h_α. Let $\mathcal{H} \subset T_c(\mathcal{M})^{[0,1]}$ be the set of smooth curves in $T_c(\mathcal{M})$ obtained as TSRVFs of trajectories in $\mathcal{M}$, $\mathcal{H} = \{h_\alpha | \alpha \in \mathcal{M}\}$.

Distance between TSRVFs: Since the TSRVFs lie on $T_c(\mathcal{M})$, the distance is measured by the standard $\mathbb{L}^2$ norm given by

$$d_h(h_{\alpha_1}, h_{\alpha_2}) = \left(\int_0^1 |h_{\alpha_1}(t) - h_{\alpha_2}(t)|^2 \right)^{\frac{1}{2}}. \tag{7}$$

If a trajectory α is warped by γ, to result in $\alpha \circ \gamma$, the TSRVF of the warped trajectory is given by:

$$h_{\alpha \circ \gamma}(t) = h_\alpha(\gamma(t)) \sqrt{\dot{\gamma}(t)} \tag{8}$$

The distance between TSRVFs remains unchanged to warping, i.e.

$$d_h(h_{\alpha_1}, h_{\alpha_2}) = d_h(h_{\alpha_1 \circ \gamma}, h_{\alpha_2 \circ \gamma}). \quad (9)$$

The invariance to group action is important as it allows us to compare two trajectories using the optimization problem stated next.

Metric invariant to temporal variability: Next, we will use d_h to define a metric between trajectories that is invariant to their time warpings. The basic idea is to partition $\mathbb{M}$ using an equivalence relation using the action of Γ and then to inherit d_h on to the quotient space of this equivalence relation. Any two trajectories α_1, α_2 are set to be equivalent if there is a warping function $\gamma \in \Gamma$ such that $\alpha_1 = \alpha_2 \circ \gamma$. The distance d_h can be inherited as a metric between the orbits if two conditions are satisfied: (1) the action of Γ on $\mathbb{M}$ is by isometries, and (2) the equivalence classes are closed sets. While the first condition has already been verified (see Eqn. 9), the second condition needs more consideration. In fact, since Γ is an open set (under the standard norm), its equivalence classes are also consequently open. This issue is resolved in [35] using a larger, *closed* set of time-warping functions as follows. Define $\tilde{\Gamma}$ to the set of all non-decreasing, absolutely continuous functions, $\gamma : [0,1] \to [0,1]$ such that $\gamma(0) = 0$ and $\gamma(1) = 1$. This $\tilde{\Gamma}$ is a semi-group with the composition operation. More importantly, the original warping group Γ is a *dense* subset of $\tilde{\Gamma}$ and the elements of $\tilde{\Gamma}$ warp the trajectories in the same way as Γ, except that they allow for singularities [35]. If we define the equivalence relation using $\tilde{\Gamma}$, instead of Γ, then orbits are closed and the second condition is satisfied as well. This equivalence relation takes the following form. Any two trajectories α_1, α_2 are said to be equivalent, if there exists a $\gamma \in \tilde{\Gamma}$ such that $\alpha_1 = \alpha_2 \circ \gamma$. Since Γ is dense in $\tilde{\Gamma}$, and since the mapping $\alpha \mapsto (\alpha(0), h_\alpha)$ is bijective, we can rewrite this equivalence relation in terms of TSRVF as $\alpha_1 \sim \alpha_2$, if **(a.)** $\alpha_1(0) = \alpha_1(0)$, and **(b.)** there exists a sequence $\{\gamma_k\} \in \Gamma$ such that $\lim_{k \mapsto \infty} h_{\alpha_1 \circ \gamma_k} = h_{\alpha_2}$, this convergence is measured under the $\mathbb{L}^2$ metric. In other words two trajectories are said to be equivalent if they have the same starting point, and the TSRVF of one can be time-warped into the TSRVF of the other using a sequence of warpings. We will use the notation $[\alpha]$ to denote the set of all trajectories that are equivalent to a given $\alpha \in \mathbb{M}$. Now, the distance d_h can be inherited on the quotient space, with the result d_s on $\mathbb{M}/\sim$ (or equivalently $\mathcal{H}/\sim$) given by:

$$d_s([\alpha_1], [\alpha_2]) \equiv \inf_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} d_h((h_{\alpha_1}, \gamma_1), (h_{\alpha_2, \gamma_2}))$$

$$= \inf_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} \left(\int_0^1 \left| h_{\alpha_1}(\gamma_1(t)) \sqrt{\dot{\gamma_1}(t)} - h_{\alpha_2}(\gamma_2(t)) \sqrt{\dot{\gamma_2}(t)} \right|^2 dt \right)^{\frac{1}{2}} \quad (10)$$

The interesting part is that we do not have to solve for the optimizers in $\tilde{\Gamma}$ since Γ is dense in $\tilde{\Gamma}$ and, for any $\delta > 0$, there exists a γ^* such that

$$|d_h(h_{\alpha_1}, h_{\alpha_2} o \gamma^*) - d_s([h_{\alpha_1}], [h_{\alpha_2}])| < \delta. \quad (11)$$

This γ^* may not be unique but any such γ^* is sufficient for our purpose. Further, since $\gamma^* \in \Gamma$, it has an inverse that can be used in further analysis. The minimization over Γ is solved for using dynamic programming. Here one samples the interval $[0,1]$ using T discrete points and then restricts to only piecewise linear γ that passes through the $T \times T$ grid. Further properties of the metric d_s are provided in [35].

Warping human actions: In the original formulation of the TSRVF [35], a set of trajectories were all warped together to produce the mean trajectory. In the context of analyzing skeletal human actions, several design choices are available to warp different actions and maybe chosen to potentially improve performance. For example, warping actions per class may work better for certain kinds of actions that have a very different speed profile, this can be achieved by modifying (10), to use class information. Next, since the work here is concerned with skeletal representations of humans, different joints have varying degrees of freedom for all actions. Therefore, in the context of skeletal representations, it is reasonable to assume that different joints require different constraints on warping functions. While it may be harder to explicitly impose different constraints to solve for γ, it can be easily achieved by solving for γ per joint trajectory instead of the entire skeleton.

5 Riemannian Functional Coding

A state of the art feature for skeletal action recognition – the Lie Algebra Relative Pairs (LARP) features [44] uses the relative configurations of every joint to every other joints, which provides a very robust representation, but also ends up being extremely high dimensional. For example, for a 15 joint skeletal system, the LARP representation lies in a 182×6 dimensional space, therefore an action sequence with 35 frames has a final representation that has 38220 dimensions. Such features do not encode the physical constraints on the human body while performing different movements because explicitly encoding such constraints may require hand tuning specific configurations for different applications, which may not always be obvious, and is labor intensive. Therefore, for a given set of human actions, if one can identify a lower dimensional *latent variable* space, which automatically encodes the physical constraints, while removing the redundancy in the original feature representation - one can theoretically represent entire actions as lower dimensional points. This is an extension to existing manifold learning techniques to Riemannian trajectories. It is useful to distinguish the lower dimensional manifold of sequences that is being learned from the Riemannian manifold that represents the individual features such as LARP on $SE(3) \times .. \times SE(3)$ etc. Our goal is to exploit the redundancy in these high dimensional features to learn a lower dimensional embedding without significant information loss. Further, the TSRVF representation, provides us speed invariance which is essential for human actions, this results in an embedding where trajectories that only differ in their rates of execution will map to the same point or to points that are very close in the lower dimensional space.

We study two main applications of coding - 1) visualization of high dimensional Riemannian trajectories, and 2) classification. For visualization, one key property is to be able to reconstruct back from the low dimensional space, which is easily done using principal component analysis (PCA). For classification, we show results on discriminative coding methods such as K-SVD, LC-KSVD, in additional to PCA, that learn a dictionary where each atom is a trajectory.

More generally, common manifold learning techniques such as Isomap [37], and LLE [29] can also be used to perform coding, while keeping in mind that it is not easy to obtain the original feature from the low dimensional code. Further, the trajectories tend to be extremely high dimensional (of the order of $10^4 - 10^5$), therefore most manifold learning techniques require massive memory requirements.

Next we describe the algorithm to obtain low dimensional codes using PCA and dictionary learning algorithms.

Algorithm 1 Riemannian Functional Coding

Input: $\alpha_1(t), \alpha_2(t) \dots \alpha_N(t) \in \mathbb{M}$
Output: Codes $C \in \mathbb{R}^{d \times N}$, in a basis $B \in \mathbb{R}^{D \times d}, d << D$
Compute the Riemannian center of mass $\mu(t)$, which also aligns $\tilde{\alpha}_1(t), \tilde{\alpha}_2(t) \dots \tilde{\alpha}_N(t)$ [35].
for $i \leftarrow [1 \dots N]$ **do**
  for $t \leftarrow [1 \dots T]$ **do**
    Compute shooting vectors $v(i,t) \in T_{\mu(t)}(M)$ as $v(i,t) = \mathbf{exp}^{-1}_{\mu(t)}(\tilde{\alpha}_i(t))$
  end for
  Define $V(i) = [v(i,1)^T \; v(i,2)^T \; \dots \; v(i,T)^T]^T$
end for
$[C, B] = \mathcal{F}(V)$. // $\mathcal{F}$ can be any Euclidean coding scheme

5.1 Representing an elastic trajectory as a vector field

The TSRVF representation allows the evaluation of first and second order statistics on *entire sequences of actions* and define quantities such as the variability of actions, which we can use to estimate the redundancy in the data similar to the Euclidean space. We utilize the TSRVF to obtain the ideal warping between sequences, such that the warped sequence is equivalent to its TSRVF. To obtain a low dimensional embedding, first we represent the sequences as deviations from a reference sequence using tangent vectors. For manifolds such as $SE(3)$ the natural "origin" I_4 can be used, in other cases the sequence mean [35] by definition lies equi-distant from all the points and therefore is a suitable candidate. In all our experiments, we found the tangent vectors obtained from the mean sequence to be much more robust and discriminative. Next, we obtain the *shooting vectors*, which are the tangent vectors one would travel along, starting from the average sequence $\mu(t)$ at $\tau = 0$ to reach the i^{th} action $\tilde{\alpha}_i(t)$ at time $\tau = 1$, this is depicted in figure 3. Note here that τ is the time in the sequence space which is different from t, which is time in the original manifold space. The combined shooting vectors can be interpreted as a *sequence tangent* that takes us from one point to another in sequence space, in unit time. Since we are representing each trajectory as a vector field, we can use existing algorithms to perform coding treating the sequences as points, because we have accounted for the temporal information. The algorithm 1 describes the process to perform coding using a generic coding function represented as $\mathcal{F} : \mathbb{R}^D \rightarrow \mathbb{R}^d$, where $d << D$. In the algorithm, C represents the low dimensional representation in the basis/dictionary B that is learned using $\mathcal{F}$.

Complexity: Computing the mean trajectory and simultaneously warping N trajectories for a single iteration can be done in $\mathcal{O}(N(T^2 + \nu))$, where the cost to compute the TSRVF is $\mathcal{O}(\nu)$. If we assume the cost of computing the exponential map is $\mathcal{O}(m)$, algorithm 1 has a time complexity of $\mathcal{O}(mNT)$. This can be a computational bottle neck for manifolds that do not have a closed form solution for the exponential and logarithmic maps. However, the warping needs to be done once offline, as test trajectories can be warped to the computed mean sequence in $\mathcal{O}(T^2 + \nu)$. Further, both the mean and shooting vector computation can be parallelized to improve speed.

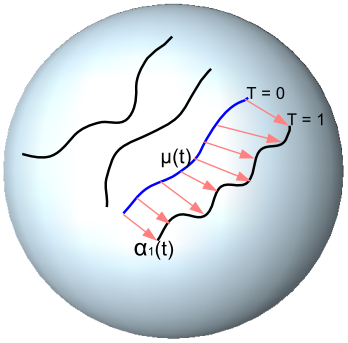


Fig. 3: Representing the warped trajectories on a manifold as a vector field, allows us to use existing algorithms to perform dimensionality reduction efficiently, while also respecting the geometric and temporal constraints.

Reconstructing trajectories from codes: If the $\mathcal{F}$ is chosen such that it can be easily inverted, i.e. we can find an appropriate $\mathcal{F}^{-1} : \mathbb{R}^d \rightarrow \mathbb{R}^D$, then the lower dimensional embedding can be used to reconstruct a trajectory on the manifold, $\mathcal{M}$, by traveling along the reconstructed tangents from the mean, $\mu(t)$. This is described in algorithm 2.

Algorithm 2 Reconstructing Non Euclidean Trajectories

Input: $C \in \mathbb{R}^{d \times N}, d << D, B \in \mathbb{R}^{D \times d}, \mu(t)$.
Output: $\widehat{\alpha}(t) \in \mathbb{M}$
for $i \leftarrow [1 \dots N]$ **do**
  $\widehat{V}_i = \mathcal{F}^{-1}(B, C)$
  Rearrange $\widehat{V}_i$ as an $m \times T$ matrix, where T is the length of each sequence.
  for $t \leftarrow [1 \dots T]$ **do**
    $\widehat{\alpha}_i(t) = \mathbf{exp}_{\mu(t)}(\widehat{V}_i(t), 1)$
  end for
end for

5.2 Choices of coding techniques

Since the final representation before dimensionality reduction lies in a vector space, any Euclidean coding scheme can be chosen depending on the application. We focus on two main techniques to demonstrate the ideas. First we perform principal component analysis (PCA) since it can be computed efficiently for extremely high dimensional data, it allows reconstruction by which we can obtain the original features, and it also provides an intuitive interpretation to visualize the high dimensional data in $2D$ or $3D$. This version of Riemannian Functional PCA (RF-PCA, previously referred to as mfPCA in [4]), generalizes functional PCA to Riemannian manifolds, and also generalizes principal geodesic analysis (PGA)[15] to sequential data. Next, we use dictionary learning algorithms, allowing us to exploit sparsity. K-SVD [2] is one of the most popular dictionary learning algorithms that has been influential in a wide variety of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2016.2564409, IEEE Transactions on Pattern Analysis and Machine Intelligence

problems. Recently, label consistent - KSVD (LCKSVD) [21] improved the results for recognition. K-Hyperline clustering [19] is a special case of K-SVD where the sparsity is forced to be 1, i.e. each point is approximated by a single dictionary atom. It is expected that since K-SVD relaxes the need for the bases to be orthogonal, it achieves much more efficient codes, that are much more compact, have the additional desirable property of sparsity and perform nearly as well as the original features themselves.

Eigenvalue decay using RF-PCA: To first corroborate our hypothesis that Riemannian trajectories are often far lower dimensional than the feature spaces in which they are observed, we show the eigenvalue decay in figure 4, after performing RF-PCA on three commonly used datasets in skeletal action recognition. It is evident that most of the variation in the datasets is captured by 10-20 eigenvectors of the covariance matrix. It is also interesting to note that that RF-PCA does a good job of approximating the different classes in the product space of $SE(3) \times \cdots \times SE(3)$. The MSRActions dataset [25] contains 20 classes and correspondingly the eigenvalue decay flattens around 20. In comparison the UTKinect [48] and Florence3D [30] datasets contain 10 and 9 classes of actions respectively, which is reflected in the eigenvalue decay that flattens closer to around 10. Features in the RF-PCA tend to be lower dimensional and more robust to noise, which is helpful in reducing the amount of pre/post processing required for optimal performance.

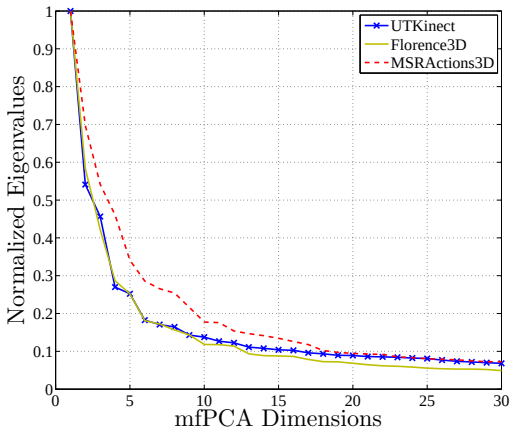


Fig. 4: Eigenvalue decay for MSRActions3D [25], UTKinect [48], and Florence3D [30] datasets obtained with RF-PCA. UTKinect and Florence3D have 10 and 9 different classes respectively, as a result the corresponding eigenvalue decay saturates at around 10 dimensions. MSRActions consists of 20 classes and the decay saturates later at around 20.

6 Experimental Evaluation

We evaluate our low dimensional Riemannian coding approach in several applications and show their advantages over conventional techniques that take geometry into account as well as other Euclidean approaches. First we address the problem of activity recognition from depth sensors such as the Microsoft Kinect. We show that a low dimensional embedding can perform as well or better than the high dimensional features on benchmark datasets. Next we evaluate our framework on the problem of visual speech recognition (VSR), or also known as lip-reading from videos. We show that, all other factors remaining the same, our low dimensional codes outperform many baselines. Finally, we also address the problem of movement quality assessment in the context of stroke rehabilitation with state-of-the-art results. We also show that low dimensional mapping provides an intuitive visual interpretation to understand quality of movement in stroke rehabilitation.

6.1 Action recognition

We use a recently proposed feature called Lie algebra relative pairs (LARP) [44] for skeleton action recognition. This feature maps each skeleton to a point on the product space of $SE(3) \times SE(3) \cdots \times SE(3)$, where it is modeled using transformations between joint pairs. It was shown to be very effective on three benchmark datasets - UTKinect [48], Florence3D [30], and MSR Actions3D [25]. We show that using geometry aware warping results in significant improvement in recognition. Further, we show that it is possible to do so with a representational feature dimension that is $250\times$ smaller than state-of-the-art.

Florence3D actions dataset [30] contains 9 actions performed by 10 different subjects repeated two or three times by each actor. There are 15 joints on the skeleton data collected using the Kinect sensor. There are a total of 182 relative joint interactions which are encoded in the features.

UTKinect actions dataset [48] contains 10 actions performed by 10 subjects, each action is repeated twice by the actor. Totally, there are 199 action sequences. Information regarding 20 different joints is provided. There are a total of 342 relative joint interactions.

MSRActions3D dataset [25] contains a total of 557 labeled action sequences consisting of 20 actions performed by 10 subjects. There are 20 joint locations provided on the skeletal data, which gives 342 relative joint interactions.

6.1.1 Alternative Representations

We compare the performance of our representation with various other recently proposed related methods:

Lie Algebra Relative Pairs (LARP): Recently proposed in [44], this feature is shown to model skeletons effectively. We will compare our results to those obtained using the LARP feature with warping obtained from DTW and unwarped sequences as baselines.

Body Parts + SquareRoot Velocity Function (BP + SRVF) : A skeleton is a collection of body parts where each skeletal sequence is represented as a combination of multiple body part sequences, proposed in [12]. It is also relevant to our work because the authors use the SRVF for ideal warping, which is the vector space version of the representation used in this paper. The representational dimension is calculated assuming the number of body parts $N_{jp} = 10$, per skeleton[12].

Principal Geodesic Analysis (PGA)[15]: Performs PCA on the tangent space of static points on a manifold. We code individual points using this technique and concatenate the final feature vector before classification.

6.1.2 Evaluation Settings

The skeletal joint data obtained from low cost sensors are often noisy, as a result of which post-processing methods

Feature	Representational Dimension	Accuracy
BP+SRVF [12]	30000	87.04
LARP [44]	38220	86.27
DTW [44]	38220	86.74
PGA [15]	6370	79.01
TSRVF	38200	89.50
RF-KSVD	**45** (sparse)	**88.55**
RF- LCKSVD	**60** (sparse)	**89.02**
RF-PCA	**110**	**89.67**

TABLE 1: Recognition performance on the Florence3D actions dataset [30] for different feature spaces.

Feature	Representational Dimension	Accuracy
BP+SRVF [12]	60000	91.10
HOJ3D [48]	N/A	90.92
LARP [44]	151,848	93.57
DTW [44]	151,848	92.17
PGA [15]	25308	91.26
TSRVF	151,848	94.47
RF-KSVD	**50** (sparse)	92.67
RF-LCKSVD	**50** (sparse)	**94.87**
RF-PCA	**105**	**94.87**

TABLE 2: Recognition performance on the UTKinect actions dataset [48].

Feature	Representational Dimension	Accuracy
BP + SRVF [12]	60000	$\mathbf{87.28 \pm 2.99}$
HON4D [27]	N/A	82.15 ± 4.18
LARP[44]	155,952	75.57 ± 3.43
DTW[44]	155,952	78.75 ± 3.08
PGA [15]	25,992	72.06 ± 3.12
TSRVF	155,952	84.62 ± 3.08
RF-KSVD	**120** (sparse)	84.45 ± 3.15
RF-LCKSVD	**50** (sparse)	83.60 ± 3.14
RF-PCA	**250**	$\mathbf{85.16 \pm 3.13}$

TABLE 3: Recognition performance on the MSRActions3D dataset [25] following the protocol of [27] by testing on 20 classes, with all possible combinations of test train subjects.

such as Fourier Temporal Pyramid (FTP) [46] have been shown to be very effective for recognition in the presence of noise. FTP is also a powerful tool to work around alignment issues, as it transforms a time series into the Fourier domain and discards the high frequency components. By the nature of FTP, the final feature is invariant to any form of warping. One of the contributions of this work is to demonstrate the effectiveness of geometry aware warping over conventional methods, and then explore the space of these warped sequences, which is not easily possible with FTP. Therefore, we perform our recognition experiments on the non-Euclidean features sequences without FTP. We computed the mean on $SE(3)$ extrinsically for the sake of computation, since the Riemannian center of mass for the manifold is iterative. In general this can lead to errors since the log map for $SE(3)$ is not unique, however we found this to work well enough to model skeletal movement in our experiments. This can easily be replaced with the more stable intrinsic version, for details on implementations we refer the reader to [13]. For classification, we use a one-vs-all SVM classifier following the protocol of [44], and set the C parameter to 1 in all our experiments. For the Florence3D and UTKinect datasets we use five different combinations of test-train scenarios and average the results. For the MSRActions dataset, we follow the train-test protocol of [27] by performing recognition on all 242 scenarios of 10 subjects of which half are used for training, and the rest for testing.

6.1.3 Recognition results

The recognition rates for Florence 3D, UTKinect, and MSRActions3D are shown in tables 1, 2 and 3 respectively. It is clear from the results that using TSRVF on a Riemannian feature, leads to significant improvement in performance. Further, using RF-PCA improves the results slightly, perhaps due to robustness to noise, but more importantly, reduces the representational dimension of each action by a factor of nearly 250. Sparse codes obtained by K-SVD, and LC-KSVD further reduce the data requirement on the features, where LC-KSVD performs as well as RF-PCA while also inducing sparsity in the codes. The improvements are significant compared to using DTW as a baseline; the performance is around 3% better on Florence3D, 2% on UTKinect, and 7% averaged over all test train variations on MSR Actions 3D. Although BP+SRVF [12] has higher recognition numbers on the MSRActions3D, our contribution lies in the significant advantage obtained using the LARP features with RF-PCA (over 7% on average). We observed that simple features in $\mathbb{R}^N$ performed exceedingly well on MSRActions3D, for example using relative joint positions (given by $\overrightarrow{v} = J_1 - J_2$, where J_1 and J_2 are 3D coordinates joints 1 and 2.) on the MSRActions3D with SRVF and PCA we obtain $87.17 \pm 3.08\%$ by embedding every action into $\mathbb{R}^{250}\times$, which is similar to [12], but in a much lower dimensional space. The performance of LCKSVD on MSRActions3D is lower than state-of-the-art because it requires a large number of samples per action class to learn a robust dictionary. There are $\sim$ 20 action classes in the dataset, but only 557 actions, therefore we are restricted to learn a much smaller dictionary. In other datasets with enough samples per class, LCKSVD performs as well as RF-PCA while also generating sparse codes.

We also show that performing PCA on the shooting vectors is significantly better than performing PCA on individual time samples using Principal Geodesic Analysis. The dimensions for LARP features are calculated as $6 \times J \times T$, where J is the number of relative joint pairs per skeleton, and T is the number of frames per video. We learn the RF-PCA basis using the training data for each dataset, and project the test data onto the orthogonal basis.

6.2 Visual speech recognition

Next we evaluate our method Visual Speech Recognition (VSR) on the OuluVS database [50] and show that the proposed coding framework outperforms comparable techniques at a significantly reduced dimensionality. VSR is the problem of understanding speech as observed from videos. The dataset contains audio and video clues, but we will use only the videos to perform recognition, this problem is also known as automatic lipreading. Speech is a dynamic process, and very much like human movement. It is also

subject to significant variation in speed, as a result of which accounting for speed becomes important before choosing a metric between two samples of speech [36].

OuluVS database [50]: This includes 20 speakers uttering 10 phrases: *Hello, Excuse me, I am sorry, Thank you, Good bye, See you, Nice to meet you, You are welcome, How are you, Have a good time.* Each phrase is repeated 5 times. All the videos in the database are segmented, with the mouth regions determined by the manually labeled eye positions in each frame. We compare our results to those reported in [36], who used covariance descriptors on the space of SPD matrices to model the visual speech using TSRVF. There are two protocols of evaluation for VSR typically, speaker independent test and speaker dependent test (SDT). We report results on the latter following [36].

6.2.1 Feature descriptor and evaluation settings

We use the covariance descriptor [41] which has proven to be very effective in modeling unstructured data such as textures, materials etc. We follow the feature extraction process as described in [36], to show the effectiveness of our framework. For the covariance descriptor, seven features are extracted including $\{x, y, I(x,y), |\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial x}|, |\frac{\partial^2 I}{\partial x}|, |\frac{\partial^2 I}{\partial x}|\}$, where x, y are the pixel locations, $I(x,y)$ is the intensity of the pixel, and the remaining terms are the first & second partial derivatives of the image with respect to x, y. This is extracted at each pixel, within a bounded region around the mouth. These covariance matrices are summed up to obtain a single 7×7 region covariance descriptor per frame. These form a trajectory of such matrices per video, which we use to calculate its TSRVF and subsequently the low dimensional codes.

We show improved results are achieved while also providing highly compressed feature representations as shown in Table 4. We train a one-vs-all SVM similar to the previous experiment, on the shooting vectors directly, by training on 60% of the subjects for each spoken phrase, this is repeated for all train/test combinations. We obtain an accuracy of 74.05% on uncompressed shooting vectors, as compared to 66.0% using a 1-NN classifier on all the 1000 videos proposed in [36]. The functional codes using different coding schemes outperform even the SVM results by around 1.5%. While the improvement is not significant, it is important to note that there is a reduction in the feature representation by a factor of nearly $100\times$.

Feature	Representational Dimension	Accuracy
Cov SPD [41]	2450	31.9
TSRVF + NN [36]	2450	66.0
Spatio-temporal[50]	N/A	70.2 (800 videos)
PGA [15]	1000	72.42 ± 3.14
TSRVF + SVM	2450	74.05 ± 4.14
RF - LCKSVD	**20** (sparse)	74.04 ± 3.5
RF - KSVD	**20** (sparse)	$\mathbf{75.63 \pm 4.45}$
RF - PCA	30	$\mathbf{75.3 \pm 5.41}$

TABLE 4: Visual speech recognition performance on the OuluVS database [50] on 1000 videos using the subject dependent testing (SDT). Results show that the functional coding representation outperforms previous state-of-the-art with similar features, while significantly reducing dimensionality.

6.3 Movement quality for stroke rehabilitation

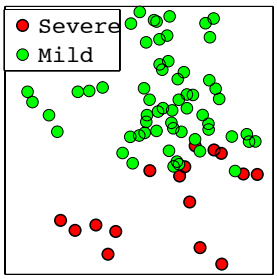
Fig. 5: The stroke rehabilitation system [10], that uses a 14 marker configuration to provide feedback on motor function for stroke patients. A typical evaluation protocol requires a therapist to observe a specified movement to give a score indicating the quality of movement.

Each year stroke leaves millions of patients disabled with reduced motor function, which severely restricts a person's ability to perform activities of daily living. Fortunately, the recent decade has seen the development of rehabilitation systems with varying degrees of automated guidance, to help the patients regain a part of their motor function. A typical system is shown in figure 5, which was developed by Chen et al. [10]. The system uses 14 markers to analyze and study the patient's movement (eg. reach and grasp), usually in the presence of a therapist who then provides a *movement quality score*, such as the Wolf Motor Function Test (WMFT) [47].

Our goal in this experiment is to predict the quality of the stroke survivor's movement as well as the therapist, so that such systems can be home-based with fewer therapist interventions. There are 14 markers on the right hand, arm and torso in a hospital setting. A total of 19 impaired subjects perform multiple repetitions of reach and grasp movements, both on-table and elevated (with the additional force of gravity acting against their movement). Each subject performs 4 sets of reach and grasp movements to different target locations, with each set having 10 repetitions.

6.3.1 Feature description and evaluation settings

We choose 4 joints – back, shoulder, elbow, and wrist. This is used to represent them in relative configurations to each other as is done in LARP [44] resulting in each *hand skeleton* that lies in $SE(3) \times \cdots \times SE(3)$ as earlier. The problem now reduces to performing logistic regression on trajectories that lie in $SE(3) \times \cdots \times SE(3)$. The stroke survivors were also evaluated by the WMFT [47] on the day of recording, where a therapist evaluates the subject's ability on a scale of 1 - 5 (with 5 being least impaired to 1 being most impaired). We use these scores as the ground truth, and predict the quality scores using the LARP features extracted from the hand markers. The dataset is small in size due to the difficulty in obtaining data from stroke survivors, therefore we use the evaluation protocol of [45], where we train on all but one test sample for regression. We compare our results to Shape of Phase Space (SoPS) [45], who perform a reconstruction of the phase space from individual motion trajectories in each dimension of each joint.



(a) Easily visualizing quality of movement in RFPCA space

Feature	Dimension	Score
SoPS* [45]	2100	88.6
KIM* [10]	NA	85.2
LARP [44]	79200	92.27
LARP + TSRVF [35]	79200	93.53
RF-PCA	**70**	**97.84**
RF-KSVD	**25** (sparse)	75.76

(b) Predicting the quality of movement in the rehabilitation of stroke survivors.

Fig. 6: The RF-PCA is able to accurately predict movement quality as compared to an expert therapist which can improve home-based systems for stroke rehabilitation.

Table 6b shows the results for different features. The baseline, using the features as it is, gives a correlation score of 92.27 to the therapist's WMFT evaluation. Adding elasticity to the curves in the $SE(3)$ product space improves the correlation score to 93.53. The functional codes improves the score significantly to 97.84, while using only 70 dimensions giving state of the art performance. We also compare our score to the kinematic based features proposed by [45]. **Visualizing quality:** Next, figure 6a shows the different movements in the lower dimensional space. Visualizing the movements in RF-PCA space, it is evident that even in $\mathbb{R}^2$, information about the quality of movement is captured. Movements which are indicative of high impairment in the motor function appear to be physically separated from the movements which indicate mild or less impairment. It is easy to see the opportunities such visualizations present for rehabilitation, for example a recent study in neuroscience [11] showed that real-time visual feedback can help *learn* the most efficient control strategies for certain movements.

6.4 Reconstruction and visualization of actions

We also show results on visualization and exploration of human actions as Riemannian trajectories. Since shapes are easy to visualize, we use the silhouette feature as a point on the Grassmann manifold.

UMD actions dataset [42]: This is a relatively constrained dataset, which has a static background allowing us to easily extract shape silhouettes. It contains 100 sequences consisting of 10 different actions repeated 10 times by the same actor. For this dataset, we use the shape silhouette of the actor as our feature, because of its easy visualization as compared to other non-linear features.

6.4.1 Reconstruction results

Once we have mapped the actions onto their lower dimensional space using RF-PCA, we can reconstruct them back easily using algorithm 2. We show that high dimensional action sequences that lie in non-Euclidean spaces can be effectively embedded into a lower dimensional latent variable space. Figure 7c shows the sampling of one axis at different points. As expected, the "origin" of the dataset contains no information about any action, but moving in the positive or negative direction of the axis results in different *styles* as shown. Note, that since we are only using 2 dimensions, there is a loss of information, but the variations are still visually discernible.

6.5 Diverse sequence sampling

Next, we show that applications such as clustering can also benefit from a robust distance metric that the TSRVF provides. Further, performing clustering is significantly faster in the lower dimensional vector space, such as the one obtained with RF-PCA. We perform these experiments on the UMD Actions data with actions as trajectories on the Grassmann manifold. K-means for data on manifolds involves generalizing the notion of distance to the geodesic distance and the mean to the Riemannian center of mass. We can further generalize this to sequences on manifolds by replacing the geodesic distance with the TSRVF distance and the mean by the RCM of sequences as defined in [35]. A variant of this problem is to identify the different kinds of groups within a population, i.e. clustering *diversly*, which is a harder problem in general and cannot be optimally solved using K-means. Instead we use manifold Precis which is a *diverse sampling* method [31]. Precis is an unsupervised exemplar selection algorithm for points on a Riemannian manifold, i.e. it picks a set of K most representative points S from a data set X. The algorithm works by jointly optimizing between approximation error and diversity of the exemplars, i.e. forcing the exemplars to be as different as possible while covering all the points.

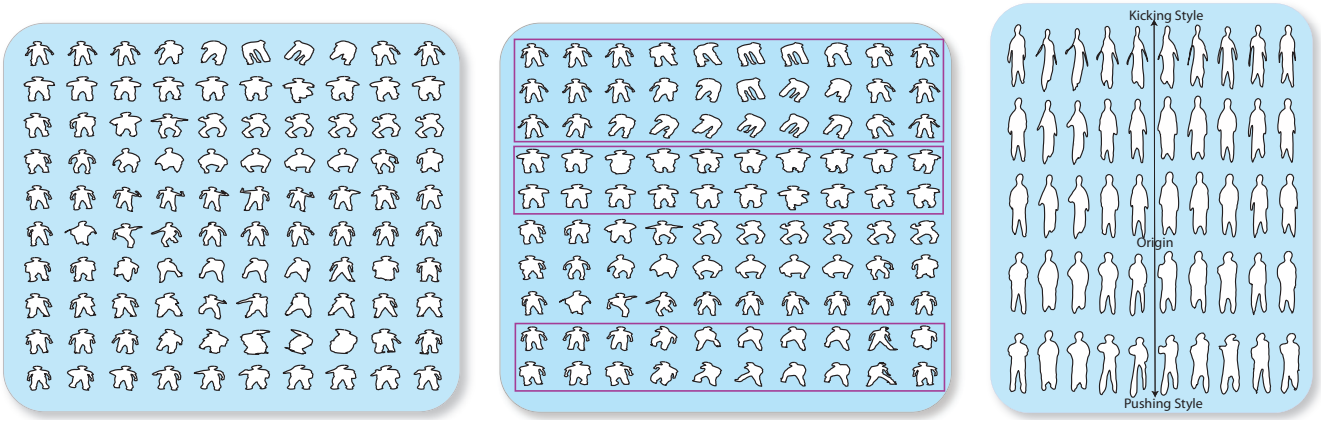
To demonstrate the generalizability of our functional codes, we perform an experiment to perform K-means clustering and diverse clustering of *entire sequences*. In the experiment on the UMD actions dataset, we constructed a collection of actions that were chosen such that different classes had significantly different populations in the collection. Action centers obtained with K-medoids is shown in figure 7b and as expected classes which have a higher population are over represented in the chosen samples as compared to Precis (figure 7a) which is invariant to the distribution. Due to the low dimensional Euclidean representation, these techniques can be easily extended to suit sequential data in a speed invariant fashion due to the TSRVF and at speeds $\sim 500\times$ faster due to RF-PCA.

7 Analysis of the TSRVF Representation

In this section, we consider different factors that influence the stability and robustness of the TSRVF representation, thereby affecting its coding performance. Factors such as (a) it's stability for different choices of the reference point, (b) the effect of noise on functional coding, and (c) arbitrary length of a trajectory, are realistic scenarios that occur in many applications.

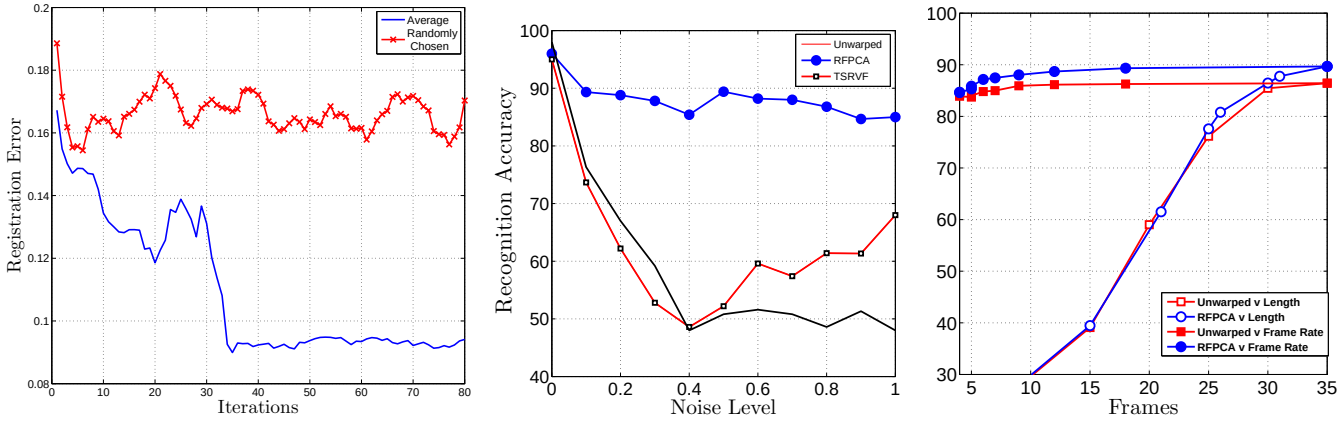
7.1 Stability to the choice of reference point

A potential weakness in the present TSRVF framework is in the choice of the reference point c, which may introduce unwanted distortions if chosen incorrectly. In manifolds such as the $SE(3)$ and SPD, a natural candidate for c is I_4, however for other manifolds such as the Grassmann, the reference must be chosen experimentally. In such cases,



(a) Row-wise: functional-Precis (b) Row-wise: functional K-medoids (c) Exploring the 2D latent action space

Fig. 7: Diverse action sampling using Precis[31] by sampling in RF-PCA space $\in \mathbb{R}^{10}$ on a highly skewed dataset. K-medoids picks more samples (marked) from classes that have a higher representation, while Precis remains invariant to it. The K-medoids and diverse clustering operations are performed $\sim 500\times$ faster in the RF-PCA space. Figure 7c shows a 2D axis sampled in the latent space. It's clearly seen that even in only 2 dimensions, some action information ("style") is discernible.



(a) Convergence & choice of reference point **C** (b) Presence of Noise (c) Trajectory length and sampling rate

Fig. 8: Robustness experiments for different factors as measured by their effect on recognition accuracy. Experiments in table (8a) and figure (8b) are performed on the Grassmann manifold, & figure (8c) shows results on the $SE(3) \times SE(3) \ldots SE(3)$ manifold. It can be clearly seen that the RFPCA representation is robust in the presence of noise, and remains more robust to different sampling rates than unwarped trajectories.

a common solution is to choose the Riemannian center of mass (RCM), since it is equally distant from all the points thereby minimizing the possible distortions. In our experiments we show that choosing an arbitrarily bad reference point can lead to poor convergence when warping multiple trajectories. We test the stability of the TSRVF representation to the choice of reference point by studying the convergence rate. We chose a set of 10 similar actions from the UMD actions dataset and measured registration error over time. The registration error is measured as $\Sigma_j d(\mu(t) - \alpha_j(t))^2$, where $\mu(t)$ is the current estimate of the mean as described in algorithm 1. When c is chosen as the mean, the convergence occurs in about 35 iterations as seen in 8a. To generate an arbitrary reference point, we chose a point at random from the dataset and travel along an arbitrary direction from that point. The resulting point is chosen as the new reference point and the unwarped trajectories are now aligned by transporting the TSRVFs to the new c. In order to account for the randomness, we repeat this experiment 10 times and take the average convergence error. The distortion is clearly visible in figure 8a, where there is no sign of convergence even after 80 iterations.

7.2 Effect of Noise

In the Euclidean setting, the robustness of PCA to noisy data is well known. We examine the consequences of performing PCA on noisy trajectories for activity recognition here. There are many different stages of adding noise to a trajectory in this context - a) sensor noise which is obtained due to poor background segmentation or sensor defect that causes the resulting shape feature to be distorted, b) warping noise that is caused by a poor warping algorithm and c) TSRVF noise, which is obtained due to a poor choice of the reference point, or obtained as a consequence to parallel transport. We have studied the effect of the reference point previously, and the effect of poor warping is unlikely in realistic scenarios. We consider the noise at the sensor level which is most likely, by inducing noise in the shape feature. We perform this by perturbing each shape point on the Grassmann manifold along a random direction, $v_r \in \mathcal{T}_{\alpha(i)}(\mathcal{G})$, for a random

distance, k drawn from a uniform distribution: $k \in \mathcal{U}(0,1)$. We generate the random tangent and the random distance to be traversed uniformly. Therefore, the i^{th} point in a trajectory is transformed as : $\hat{\alpha}(i) = \mathbf{exp}(\alpha(i), k\ v_r)$. We then perform a recognition experiment on the noisy datasets using the RFPCA, TSRVF and unwarped representations. Figure 8b shows the results of the experiment on the UMD actions dataset, with k on the X-axis. As expected, the RFPCA representation is least affected, while the TSRVF representation performs slightly better than the unwarped trajectories. The different levels of noise indicate how far along the random vector one traverses to obtain the new *noisy* shape.

7.3 Arbitrary length & sampling rates

The choice of parameter T in algorithm 1, directly affects the resulting dimensionality of the the trajectory before performing coding. Here we investigate its effect on coding and recognition. We can generate different trajectory lengths by considering two factors a)frame-rate, where $\widehat{\alpha}(t) = \alpha(\mathbf{m}t)$ where the factor is governed by $\mathbf{m}$, and b) arbitrary end point, where $\widehat{\alpha}(t) = \alpha(1 : T')$, such that $T' < T$. The TSRVF is invariant to frame rate or sampling rate, therefore for a wide range of sampling rates, the recognition accuracy remains unchanged. To observe this, we perform a recognition experiment on the Florence3D skeleton actions dataset. The results for both factors are shown in figure 8c, and it is seen that in both cases the TSRVF warped actions are recognized better than the unwarped actions with an average of 5% better accuracy.

Canonical length: Using the coding framework proposed in this paper, it is conceivable that there is a close relationship between the *true* length of a trajectory and its intrinsic dimensionality. For example - a more complex trajectory contains more information which naturally requires a higher dimensional RFPCA space to truly capture its variability. However, determining the explicit relationship between the RF-PCA dimension and the canonical length of a trajectory is out of the scope of this work.

8 Conclusion & Future Work

In this paper we introduced techniques to explore and analyze sequential data on Riemannian manifolds, applied to human activities, visual speech recognition, and stroke rehabilitation. We employ the TSRVF space [35], which provides an elastic metric between two trajectories on a manifold, to learn the latent variable space of actions, which is a generalization of manifold learning to Riemannian trajectories. We demonstrate these ideas on the curved product space $SE(3) \times \cdots \times SE(3)$ for skeletal actions, the Grassmann manifold, and the SPD matrix manifold. We propose a framework that allows for the parameterization of Riemannian trajectories using popular coding methods – RF-PCA which generalizes functional PCA to manifolds and PGA to sequences, sparsity inducing coding RF-KSVD and discriminative RF-LCKSVD. The learned codes not only provide a compact and robust representation that outperforms many state of the art methods, but also the visualization of actions due to its ability to reconstruct original non-linear features. We also show applications for intuitive visualization of abstract properties such as quality of movement, which has a proven benefit in rehabilitation. The proposed representation also opens up several opportunities to understand various properties of Riemannian trajectories, including their canonical lengths, their intrinsic dimensionality, ideal sampling rates, and other inverse problems which are sure to benefit several problems involving the analysis of temporal data.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008.

[2] M. Aharon, M. Elad, and A. Bruckstein. k -svd: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, Nov 2006.

[3] R. Anirudh and P. Turaga. Geometry-based symbolic approximation for fast sequence matching on manifolds. *International Journal of Computer Vision*, pages 1–13, 2015.

[4] R. Anirudh, P. Turaga, J. Su, and A. Srivastava. Elasting functional coding of human actions: from vector fields to latent variables. In *CVPR*, pages 3147–3155, 2015.

[5] E. Begelfor and M. Werman. Affine invariance revisited. *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[6] D. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, 1994.

[7] A. Biess, D. G. Liebermann, and T. Flash. A computational model for redundant human three-dimensional pointing movements: integration of independent spatial and temporal motor plans simplifies movement dynamics. *The Journal of Neuroscience*, 27(48):13045–13064, 2007.

[8] W. M. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry. Revised 2nd Ed.* Academic, New York, 2003.

[9] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *CVPR*, June 2009.

[10] Y. Chen, M. Duff, N. Lehrer, H. Sundaram, J. He, S. L. Wolf, T. Rikakis, T. D. Pham, X. Zhou, H. Tanaka, et al. A computational framework for quantitative evaluation of movement during rehabilitation. In *AIP Conference Proceedings-American Institute of Physics*, volume 1371, page 317, 2011.

[11] Z. Danziger and F. A. Mussa-Ivaldi. The influence of visual motion on motor learning. *The Journal of Neuroscience*, 32(29):9859–9869, 2012.

[12] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo. 3-d human action recognition by shape analysis of motion trajectories on riemannian manifold. *Cybernetics, IEEE Transactions on*, PP(99):1–1, September 2014.

[13] X. Duan, H. Sun, and L. Peng. Riemannian means on special euclidean group and unipotent matrices group. *The Scientific World Journal*, 2013, 2013.

[14] M. Faraki, M. Harandi, and F. Porikli. More about vlad: A leap from euclidean to riemannian manifolds. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 4951–4960, June 2015.

[15] P. T. Fletcher, C. Lu, S. M. Pizer, and S. C. Joshi. Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging*, 23(8):995–1005, August 2004.

[16] C. R. Goodall and K. V. Mardia. Projective shape analysis. *Journal of Computational and Graphical Statistics*, 8(2), 1999.

[17] M. T. Harandi, M. Salzmann, and R. Hartley. From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices. In *ECCV 2014*, pages 17–32, 2014.

[18] M. T. Harandi, C. Sanderson, C. Shen, and B. C. Lovell. Dictionary learning and sparse coding on grassmann manifolds: An extrinsic solution. In *ICCV*, pages 3120–3127, 2013.

[19] Z. He, A. Cichocki, Y. Li, S. Xie, and S. Sanei. K-hyperline clustering learning for sparse component analysis. *Signal Processing*, 89(6):1011–1022, 2009.

[20] J. Ho, Y. Xie, and B. C. Vemuri. On a nonlinear generalization of sparse coding and dictionary learning. In *ICML (3)*, pages 1480–1488, 2013.

[21] Z. Jiang, Z. Lin, and L. Davis. Label consistent k-svd: Learning a discriminative dictionary for recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2651–2664, Nov 2013.
[22] J. Jost. *Riemannian Geometry and Geometric Analysis (6. ed.)*. Springer, 2011.
[23] N. D. Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. *Advances in neural information processing systems*, 16:329–336, 2004.
[24] M. Lewandowski, D. Makris, S. Velastin, and J.-C. Nebel. Structural laplacian eigenmaps for modeling sets of multivariate sequences. *Cybernetics, IEEE Transactions on*, 44(6):936–949, June 2014.
[25] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010*, pages 9–14. IEEE, 2010.
[26] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.
[27] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *(CVPR), 2013*, pages 716–723. IEEE, 2013.
[28] X. Pennec, P. Fillard, and N. Ayache. A Riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.
[29] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *SCIENCE*, 290:2323–2326, 2000.
[30] L. Seidenari, V. Varano, S. Berretti, A. D. Bimbo, and P. Pala. Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2013*, pages 479–485, 2013.
[31] N. Shroff, P. K. Turaga, and R. Chellappa. Manifold precis: An annealing technique for diverse sampling of manifolds. In *NIPS*, 2011.
[32] A. Srivasatava and E. Klassen. Bayesian geometric subspace tracking. *Advances in Applied Probability*, 36(1):43–56, March 2004.
[33] A. Srivastava, S. H. Joshi, W. Mio, and X. Liu. Statistical shape analysis: Clustering, learning, and testing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(4), 2005.
[34] J. Su, I. L. Dryden, E. Klassen, H. Le, and A. Srivastava. Fitting smoothing splines to time-indexed, noisy points on nonlinear manifolds. *Image Vision Comput.*, 30(6-7):428–442, 2012.
[35] J. Su, S. Kurtek, E. Klassen, and A. Srivastava. Statistical analysis of trajectories on Riemannian manifolds: Bird migration, hurricane tracking, and video surveillance. *Annals of Applied Statistics*, 8(1), 2014.
[36] J. Su, A. Srivastava, F. D. M. de Souza, and S. Sarkar. Rate-invariant analysis of trajectories on riemannian manifolds with application in visual speech recognition. In *CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 620–627, 2014.
[37] J. B. Tenenbaum, V. d. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.
[38] N. F. Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision*, 2(5):2, 2002.
[39] P. K. Turaga and R. Chellappa. Locally time-invariant models of human activities using trajectories on the Grassmannian. In *CVPR*, pages 2435–2441, 2009.
[40] P. K. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa. Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(11):2273–2286, 2011.
[41] O. Tuzel, F. M. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. *European Conference on Computer Vision*, pages II: 589–600, 2006.
[42] A. Veeraraghavan, R. Chellappa, and A. K. Roy-Chowdhury. The function space of an activity. *IEEE CVPR*, pages 959–968, 2006.
[43] A. Veeraraghavan, A. K. R. Chowdhury, and R. Chellappa. Matching shape sequences in video with applications in human movement analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(12):1896–1909, 2005.
[44] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *(CVPR), 2014*, pages 588–595, June 2014.
[45] V. Venkataraman, P. Turaga, N. Lehrer, M. Baran, T. Rikakis, and S. L. Wolf. Attractor-shape for dynamical analysis of human movement: Applications in stroke rehabilitation and action recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 514–520. IEEE, 2013.
[46] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *(CVPR), 2012*, pages 1290–1297, June 2012.
[47] S. L. Wolf, P. A. Catlin, M. Ellis, A. L. Archer, B. Morgan, and A. Piacentino. Assessing wolf motor function test as outcome measure for research in patients after stroke. *Stroke*, 32(7):1635–1639, 2001.
[48] L. Xia, C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW)2012*, pages 20–27. IEEE, 2012.
[49] Z. Zhang, J. Su, E. Klassen, H. Le, and A. Srivastava. Video-based action recognition using rate-invariant analysis of covariance trajectories. *CoRR*, abs/1503.06699, 2015.
[50] G. Zhao, M. Barnard, and M. Pietikäinen. Lipreading with local spatiotemporal descriptors. *IEEE Transactions on Multimedia*, 11(7):1254–1265, 2009.
[51] F. Zhou and F. De la Torre. Generalized time warping for multi-modal alignment of human motion. In *(CVPR), 2012*, pages 1282–1289. IEEE, 2012.

Rushil Anirudh is a PhD candidate working with the School of Electrical Engineering and the School of Arts, Media, and Engineering at Arizona State University. He received his M.S. at ASU in 2012, and his B.Tech in 2010 from National Institute of Technology Karnataka in India. He is currently interested in solving problems in computer vision using machine learning and Riemannian geometry.

Pavan Turaga (S05, M09, SM14) is an Assistant Professor in the School of Arts, Media, Engineering, and Electrical Engineering at Arizona State University. He received the B.Tech. degree in electronics and communication engineering from the Indian Institute of Technology Guwahati, India, in 2004, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park in 2008 and 2009 respectively. He then spent two years as a research associate at the Center for Automation Research, University of Maryland, College Park. His research interests are in computer vision and computational imaging with applications in activity analysis, and dynamic scene analysis, with a focus on non-Euclidean techniques for these applications. He was awarded the Distinguished Dissertation Fellowship in 2009. He was selected to participate in the Emerging Leaders in Multimedia Workshop by IBM, New York, in 2008. He received the National Science Foundation CAREER award in 2015.

Jingyong Su received the BE and MS degrees in Electrical Engineering in 2006 and 2008 from Harbin Institute of Technology in China, and a PhD degree in the Department of Statistics at Florida State Universityin May, 2013. He is currently an assistant professor in the Department of Mathematics and Statistics at Texas Tech University in Lubbock. His main research interests include statistical image analysis, computer visionand computational statistics.

Anuj Srivastava is a Professor of Statistics and a Distinguished Research Professor at the Florida State University. He obtained his PhD degree in Electrical Engineering from Washington University in St. Louis in 1996 and was a visiting research associate at Division of Applied Mathematics at Brown University during 1996-1997. He joined the Department of Statistics at the Florida State University in 1997 as an Assistant Professor. His areas of research include statistics on nonlinear manifolds, statistical image understanding, functional analysis, and statistical shape theory. He has published more than 200 papers in refereed journals and proceedings of refereed international conferences. He has been the associate editor for the Journal of Statistical Planning and Inference, and the IEEE Transactions on Signal Processing and the IEEE Transactions on Pattern Analysis and Machine Intelligence. He is a fellow of IAPR and a senior member of IEEE.