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/TMM.2020.3003783, IEEE Transactions on Multimedia

3D Skeletal Gesture Recognition via Discriminative Coding on Time-Warping Invariant Riemannian Trajectories

Xin Liu, *Member, IEEE*, and Guoying Zhao, *Senior Member, IEEE*

***Abstract*—Learning 3D skeleton-based representation for gesture recognition has progressively stood out because of its invariance to the viewpoint and background dynamics of video. Typically, existing techniques use absolute coordinates to determine human motion features. The recognition of gestures, however, is irrespective of the position of the performer, and the extracted features should be invariant to body size. In addition, when comparing and classifying gestures, the problem of temporal dynamics can greatly distort the distance metric. In this paper, we represent a 3D skeleton as a point in the special orthogonal group $SO(3)$ product space that expressly models the 3D geometric relationships between body parts. As such, a gesture skeletal sequence can be described by a trajectory on a Riemannian manifold. Following that, we propose to generalize the transported square-root vector field to obtain a time-warping invariant metric for comparing these trajectories (identifying these gestures). Moreover, by specifically considering the labeling information with encoding, a sparse coding scheme of skeletal trajectories is presented to enforce the discriminant validity of atoms in the dictionary. Experimental results indicate that the proposed approach has achieved state-of-the-art performance on many challenging gesture recognition benchmarks.**

***Index Terms*—3D skeleton representation, gesture recognition, Riemannian geometry, sparse coding, time-warping invariant feature**

I. Introduction

HUMAN body gesture recognition is emerging as a key area of computer vision research and has been widely utilized in applications such as automated sign language translation, gaming, human-computer interfaces, and artificial companions. Three-dimensional skeleton data is rapidly gaining popularity as it simplifies the mission by replacing monocular RGB cameras with more advanced sensors like the Kinect [1], which can localize gesture performers directly and produce human skeleton joint trajectories in the manner of real-time processing. Compared to RGB input, skeletal data are invariant to camera viewpoints and robust to a changing environment.

This work was supported by the Academy of Finland for postdoctoral researcher project (grant 331146), project MiGA (grant 316765), and ICT 2023 project (grant 328115), the strategic Funds of the University of Oulu, the Infotech Oulu, Finland. As well, the authors wish to acknowledge CSC - IT Center for Science, Finland, for computational resources. *(Corresponding author: Guoying Zhao.)*

X. Liu is with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China, and also with the Center for Machine Vision and Signal Analysis, University of Oulu, FI-90014, Finland. (E-mail: linuxsino@gmail.com)

G. Zhao is with the Center for Machine Vision and Signal Analysis, University of Oulu, FI-90014, Finland, and also with the School of Information and Technology, Northwest University, 710069, China. (E-mail: guoying.zhao@oulu.fi)

A great number of 3D skeleton-based models [2] [3] for gesture recognition have been developed over the past decade, ranging from hand-crafted features to numerous deep learning approaches. While considerable progress has been made in this area, it remains difficult to correctly recognize the human gesture in unconstrained settings. Two issues need to be addressed thoroughly:

- One concern in the identification of human gesture is the depiction of features to reflect the variation of the human body (skeleton) and its dynamics. Existing methods commonly use the absolute coordinate (real world) to derive the characteristics of human motion. Nonetheless, behaviors are independent of the position of the performer and the feature should be invariant to the length of the body part (performer's size).
- Another problem in gesture recognition is temporal dynamics. For instance, even the same gesture executed by the same person may happen at varying execution speeds and have different start/end points, and it is yet more complex when one considers different performers. Therefore, the variation of a category of human behavior can be very high, and if temporal dynamics are overlooked, it will certainly deteriorate the accuracy of recognition.

A common way of dealing with the first problem is to convert all 3D joint coordinates from the world coordinate system into a performer-centered coordinate system, for example, by positioning the hip center at the origin. Nevertheless, its success depends heavily on the exact location of this particular point. Another route is to consider the relative geometry between different body parts (bones) like the Lie Group [4] that uses rotations and translations (rigid-body transformation) to represent the body parts' 3D geometric relationships. The translation, however, is not a scale-invariant representation as the size of the skeleton varies from subject to subject. The researchers picked one of the skeletons from the training sets as a reference in [4], but in this empiric procedure it is difficult to normalize the skeletal data to specifically accommodate scale variations.

A typical treatment for the second issue uses a graphic model to describe the presence of sub-states (events), where time series are reorganized by a sequential prototype, and the temporal dynamics of gestures are learned as a set of transitions in these prototypes [5]. The hidden Markov model (HMM) [6] is the representative model. Nonetheless, a HMM's input sequences must be segmented in advance, according

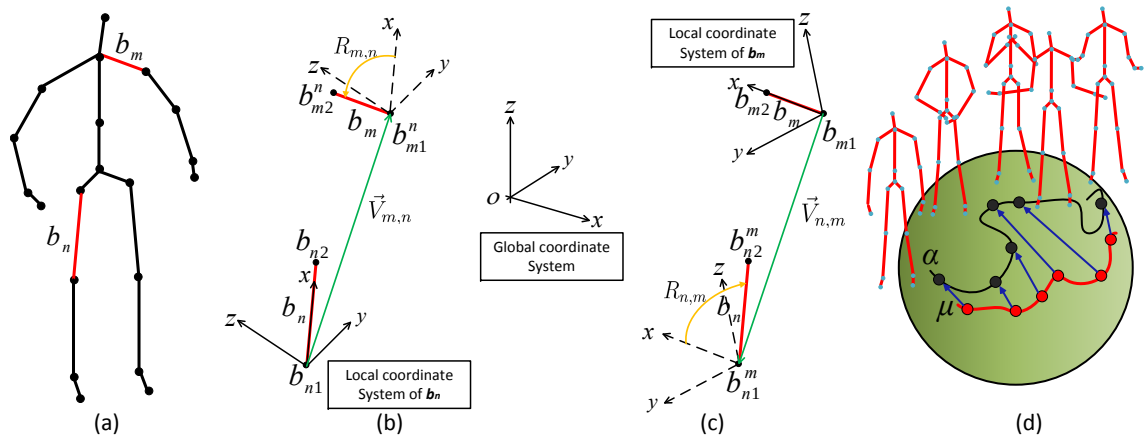

Fig. 1. The figure depicts: (a) an illustration of a 3D skeleton, (b) a representation of bone b_m in the local coordinate system of b_n, (c) a representation of b_n in the local coordinate system of b_m, (d) a pictorial of the warped trajectory α on a manifold according to a reference μ.

to certain unique clustering metrics or discriminative states, which is a challenging problem in itself. Recently, with the advancement of deep learning, a lot of researches [7]–[9] have addressed the issue of temporal dynamics by using a recurrent neural network (RNN), such as long short-term memory (LSTM). While LSTM is a strong tool for sequential data modeling, it is still difficult to learn the details of the entire sequence with many sub-events (states). In fact, the most common solution to temporal dynamics is dynamic time warping (DTW) [4] [10], which needs to choose a nominal temporal alignment, and then all the sequences of a class are warped to that alignment. Nonetheless, DTW's performance depends heavily on the choice of the reference sequence, and such a reference is commonly obtained by experience.

In this paper, a novel approach to gesture recognition is proposed to tackle the above problems. The key contributions are summed up as follows:

1) We represent a human skeleton as a point in the special orthogonal group product space, which is a Riemannian manifold. This representation is independent of the position of the performer and can use rotations to explicitly model the 3D geometric relationships between body parts. A gesture (a skeletal sequence) can then be represented by a trajectory of these points (see Fig. 1 (d)). The mission of gesture recognition is formulated as a question of measuring the similarities between the shapes of trajectories.

2) We expand the representation of the transported square-root vector field (TSRVF) to compare trajectories in the $SO(3) \times \cdots \times SO(3)$ product space. So, this time-warping invariant feature can overcome the temporal dynamic issue of gesture recognition.

3) We present a sparse coding of skeletal trajectories by explicitly considering the labeling information for each atom in order to enforce the discriminant validity of the dictionary. The comparison of experimental results from many challenging datasets shows that the proposed method has achieved state-of-the-art performance.

The remainder of this article is organized as follows. Section II reviews related methods. The 3D skeleton representation of the product space of $SO(3) \times \cdots \times SO(3)$ is presented in Section III. The TSRVF's representation for trajectory comparison is defined in Section IV. The sparse coding of skeletal trajectories is given in Section V. In Section VI experiments and discussions are presented, and conclusions are drawn in Section VII.

The preliminary work has appeared in [11].

II. Related Methods

Over the past few years, several 3D skeletal human gesture recognition models have been studied. We provide a categorized review of the relevant literature that is principally on handcrafted features, deep neural networks, and manifold-based models.

A. Approaches with handcrafted features

Conventional approaches for skeleton-based action recognition usually develop handcrafted features to model the motions of humans, representative research includes: the histogram of 3D joints (HOJ3D) [12], EigenJoints by principal component analysis (PCA) [13], discriminative key-frames [14], histogram of oriented 4D normals (HON4D) [15], sequence of most informative joints (SMIJ) [16], and rotation and relative velocity (RRV) [10]. Further research attempted to develop a robust gesture recognition model, such as the actionlets ensemble [17], the maximum entropy Markov model (MEMM) [18], latent structural support vector machine (SVM) (pose-based) [19], HMMs [20] [21], conditional random field (CRF) [22], latent Dirichlet allocation (LDA) [23], Markov random field (MRF) [24], and the naive Bayes nearest neighbor (NBNN) [25]. Due to the space limitation, we only list some representative methods for using handcrafted features, for more details, please see the surveys [2] [3] [26].

B. Approaches with deep neural networks

Recently, with the development of multi-cores and throughput GPU devices, much research has concentrated on using a large amount of data to train deep neural networks [6]–[9], [27]–[38], where the convolutional neural network (CNN) and RNN are the most widely used schemes. Specifically, the RNN

with LSTM carefully designs a series of schemes to memorize the contextual information obtained from previous sequential inputs, allowing the long-term temporal dependency to be monitored. In [7], Du $et\ al.$ first introduced a bi-directional LSTMs for action recognition, dividing the entire skeleton into five groups of joints and feeding each group into a group-specific LSTM subnetwork. The system then hierarchically fuses the outputs of these subnetworks and eventually feeds them into another set of higher-level LSTMs to represent the global body motions. In [30], a scheme for encoding/decoding LSTMs was proposed for action recognition. The encoder is trained on 3D skeleton sequences in an unsupervised manner. The manifold is then used to regularize the supervised learning of decoding LSTM for RGB data-based recognition. In [31], Li $et\ al.$ employed a Gaussian-like curve to measure the confidences of the start/end frame of action and used a combined classification regression LSTM to solve an online action detection and recognition problem. Zhu $et\ al.$ [32] added a group sparse regularization term to an LSTM's cost function, which enables the network to automatically learn the co-occurrence of discriminative skeleton joints. In [8], Liu $et\ al.$ implemented a trust gate into the LSTM in order to learn the reliability of the inputs and accordingly adjust their confidence in updating the context information. Following that, in [9], Liu $et\ al.$ also proposed a global context-aware attention LSTM that aimed at handling the limitation of LSTM in perceiving global contextual information. While LSTM is powerful in modeling sequential data, it still suffers from remembering the information of an entire sequence with many time steps (states) [29]. In addition, RNN based models lack the ability to efficiently learn the spatial relations between the skeleton joints [39].

CNN-based methods model the skeleton data as a 2D/3D grid (pseudo-image) with manually designed transformation rules. For instance, in [29], three clips corresponding to the three channels of cylindrical coordinates were extracted in order to represent the skeleton sequence. Based on those clips, Ke $et\ al.$ introduced a CNN to learn temporal information and then used a multi-task learning network (MTLN) to jointly learn the feature vectors at all the time steps in a parallel way. Kim $et\ al.$ [40] proposed a temporal convolutional network with residual units (Res-TCN) that learns to pay different levels of attention both spatially and temporally. In [39], a novel skeletal pseudo-image was proposed by computing the magnitude and orientation values of the joints. In [35], Li $et\ al.$ proposed a hierarchical co-occurrence network (HCN) that transforms a skeleton sequence into a pseudo-image by treating the joint coordinates (x, y, z) as the channels (R, G, B) of a pixel. Besides the independent convolution operation on each channel of input, an element-wise summation across channels is used for globally aggregating co-occurrence features. Instead of taking the skeleton data as a pseudo-image or grids, the skeletal sequence in [41] [42] was represented as a graph in a non-Euclidean space with the joints as vertexes and their connections in the human body as edges. Then, a generalized CNN method called spatial-temporal graph convolutional network (ST-GCN) [42] was proposed to model the arbitrary structures of graphs. In this framework, the graph embedding is computed by a graph convolutional network (GCN) layer aggregating node information from its neighbors by differentiable aggregation functions. Also, in spatio-temporal graph convolution (STGC) [41], Li $et\ al.$ proposed a graph-based skeleton representation, which is then fed into the GCN to learn the spatial and temporal patterns automatically. Papadopoulos $et\ al.$ [43] extended ST-GCN by introducing the graph vertex feature encoder (GVFE) and the dilated hierarchical temporal convolutional network (DH-TCN). The GVFE is used to generate graph vertex features, and the DH-TCN is designed for modeling long-term and short-term dependencies simultaneously. Compared to ST-GCN [42], this method can achieve almost the same accuracy on benchmarks but with a small number of layers and parameters. Due to the success of GCN, in the attention-enhanced graph convolutional LSTM network (AGC-LSTM) [37] the graph convolution operation is introduced to the RNN (LSTM) for better recognition of skeleton-based actions. AGC-LSTM can learn discriminative features in spatial configuration and temporal dynamics as well as model the co-occurrence relationship between spatial and temporal domains [37].

C. Approaches with manifolds

In this subsection, we focus on the relevant manifold-based solutions. A representative work is the Lie group in [4], which used the special Euclidean (Lie) group $SE(3)$ to represent the 3D geometric relationships between body parts. A convenient way to analyze a Lie group is to embed them into Euclidean spaces, with the embedding typically achieved by flattening the manifold via tangent spaces, namely the Lie algebra $\mathfrak{se}(3)$ at the tangent space identity I_4. In this way, previous recognition tasks in a curved manifold space are converted into the classification problems in a common vector space. The researchers of [4] used DTW and a Fourier temporal pyramid (FTP) to address the temporal dynamics problem. As discussed in Section I, DTW's success depends heavily on the nominal temporal alignment selection. Also, an FTP is limited by the time window length and can only use restricted contextual information [7]. Following the same representation, Anirudh $et\ al.$ [44] introduced a system of transported square-root velocity fields [45] to encode the trajectories lying in Lie groups. Therefore, the distance between two trajectories is invariant to identical time warping. Because the final feature is a vector with a high dimension, PCA is used to reduce the dimension. However, PCA is an unsupervised model, it cannot be boosted by a labeled learning. According to the square root velocity (SRV) model [46], in [47], trajectories were transported to a reference tangent space attached to the Kendall's shape space at a fixed point. Nevertheless, in the case where points are not close to the reference point, this procedure can introduce distortions. In [48], Ho $et\ al.$ introduced a general framework for sparse coding and dictionary learning on Riemannian manifolds. In comparison to [46], which used the fixed point for embedding, [48] operated on the tangent bundle, meaning that each point of the manifold was coded on its attached tangent space where the atoms are mapped. Following [48], Tanfous $et\ al.$ [49] explored sparse coding and dictionary

learning in the Kendall shape spaces (Kendall-SCDL), aiming to study the time-varying shapes of 3D skeleton trajectories for action recognition. However, Kendall-SCDL [49] has a mandatory step of dictionary initialization that heavily relies on principal geodesic analysis (PGA) [50] to generate atoms. Also, Kendall-SCDL is still an unsupervised model. In LieNet [51], Huang $et\ al.$ incorporated the Lie group structure into a deep network architecture to learn more appropriate Lie group features for 3D action recognition.

Another branch of manifold-based methods uses kernel functions to embed the Riemannian manifolds into a reproducing kernel Hilbert space (RKHS) [52] [53]. However, the input manifold of symmetric positive definite (SPD) matrices is from a covariance descriptor, which is calculated by just a few skeleton joints due to the computationally intensive kernel functions. Recently, Kacem $et\ al.$ [54] proposed modelling the temporal evolution of the human skeleton as parametrized trajectories on a Riemannian manifold with Gramian matrices. And the geometry of the manifold is assumed to be positive-semidefinite matrices with fixed rank. This method relies on DTW for sequence alignment, while the resulting distance may not lead to a positive-definite kernel for classification [55].

III. The Product Space of SO(3) for 3D Skeleton Representation

Inspired by rigid body kinematics [56], the relative geometry between different body parts [4] is introduced, as this feature has a view-invariant property that can guarantee the uniqueness of motion representation and can thus provide a better reflection of human gestures than that gained from the use of absolute positions.

Mathematically, any rigid body displacement can be realized by a rotation about an axis, paired with a translation parallel to that axis. This 3D rigid body displacement forms an $SE(3)$, the special Euclidean group in three dimensions [56]. An $SE(3)$ can be identified with the space of 4×4 matrices in the form

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

where $R \in SO(3)$ is a point in the special orthogonal group $SO(3)$, it denotes the rotation matrix, and $\vec{v} \in \mathbb{R}^3$ denotes the translation vector.

The human skeleton can be modeled by an articulated system of rigid segments connected by joints. As such, let $S = (J, B)$ be a skeleton, where $J = \{j_1, \cdots, j_N\}$ indicates the set of body joints, and $B = \{b_1, \cdots, b_M\}$ indicates the set of body bones (oriented edges). The relative geometry between a pair of body parts (bones) can be expressed as a point in $SE(3)$, as studied in [4]. More specifically, given a pair of bones b_m and b_n, their relative geometry can be featured in a local coordinate system attached to another bone [4]. Let $b_{i1} \in \mathbb{R}^3$ and $b_{i2} \in \mathbb{R}^3$ represent the starting and end points of bone b_i, respectively. The local coordinate system of bone b_n is determined by rotating with minimal rotation and translating the global coordinate system so that b_{n1} serves as the origin and b_n coincides with the $x-$axis (Fig. 1 provides an example for illustration). Then, at time t, the representation of bone b_m in the local coordinate system of b_n (see Fig. 1 (b)), the starting point $b^n_{m1}(t) \in \mathbb{R}^3$ and end point $b^n_{m2}(t) \in \mathbb{R}^3$ are given by

$$\begin{bmatrix} b^n_{m1}(t) & b^n_{m2}(t) \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} R_{m,n}(t) & \vec{v}_{m,n}(t) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & l_m \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}, \tag{2}$$

where $R_{m,n}(t)$ and $\vec{v}_{m,n}(t)$ denote the rotation and translation measured in the local coordinate system attached to b_n, and l_m is the length of b_m. Likewise, $R_{n,m}(t)$, $\vec{v}_{n,m}(t)$, and l_n (see Fig. 1 (c)) can be used to represent bone b_n in the local coordinate system of b_m. The sizes of bones (body parts) can be assumed to not vary with time according to the theory of rigid body kinematics. Therefore, the relative geometry of b_m and b_n at time t can be represented by

$$\begin{aligned} P_{m,n}(t) &= \begin{bmatrix} R_{m,n}(t) & \vec{v}_{m,n}(t) \\ 0 & 1 \end{bmatrix} \in SE(3), \\ P_{n,m}(t) &= \begin{bmatrix} R_{n,m}(t) & \vec{v}_{n,m}(t) \\ 0 & 1 \end{bmatrix} \in SE(3). \end{aligned} \tag{3}$$

There is natural stability and consistency in this relative geometry. For instance, if a pair of bones undergo the same rotation, their relative geometry matrix would not be altered. However, one restriction of this motion feature is that the translation $\vec{v}$ is relative to the size of the performer (subject). As we know, in an unconstrained setting, achieving a scale-invariant skeletal representation is very necessary for the recognition mission. We delete the translation from motion representation to eliminate the skeleton scaling variations, then the relative geometry of b_m and b_n at time t can be represented by rotations $R_{m,n}(t)$ and $R_{n,m}(t)$, and expressed as elements of $SO(3)$. Now, let M denote the number of bones, the corresponding feature for the entire human skeleton is interpreted by the relative geometry between all pairs of bones, as a point $C(t) = (R_{1,2}(t), R_{2,1}(t), \ldots, R_{M-1,M}(t), R_{M,M-1}(t))$ on the curved product space of $SO(3) \times \cdots \times SO(3)$, and the number of $SO(3)$ is $2C^2_M$, where C^2_M is the combination formula.

IV. The TSRVF for The Product Space of SO(3)-Featured Trajectories

As shown in the last paragraph, on a Riemannian manifold, a gesture skeleton sequence can be characterized as a trajectory. Consequently, the mission of gesture recognition is to measure the similarities of trajectory shapes, and a distance function on the Riemannian manifold is the basis for these comparability determinations. To address this problem, a few Riemannian metrics [57] have been proposed, but it is still difficult to accurately model the temporal dynamics of gesture trajectories.

A. Representation of trajectories on a Riemannian manifold

Specifically, let α denote a smooth oriented curve (trajectory) on a Riemannian manifold M, and let $\mathcal{M}$ denote the set of all such trajectories: $\mathcal{M} = \{\alpha : [0, 1] \to M | \alpha \text{ is smooth}\}$.

Reparameterizations are operated by increasing diffeomorphisms $\gamma : [0,1] \to [0,1]$, and the set of all these orientation preserving diffeomorphisms is denoted by $\Gamma = \{\gamma \to [0,1]\}$. Actually, γ plays the role of a time-warping operation where $\gamma(0) = 0, \gamma(1) = 1$ is used to maintain the end points of the curve. So, if α in the form of time observations $\alpha(t_1), ..., \alpha(t_n)$, is a trajectory on M, the composition $\alpha \circ \gamma$ in the form of the time-warped trajectory $\alpha(\gamma(t_1)), ..., \alpha(\gamma(t_n))$, is also a trajectory that goes through the same sequence of points as α but at the rate of evolution governed by γ [45].

In order to identify trajectories, a metric is required to characterize the variance of a class of trajectories and to quantify the information contained within a trajectory. Calculating a point-wise discrepancy is a straightforward and simple solution. As M is a Riemannian manifold, it is possible to use a natural distance of d_m between points on M. Then, for any two trajectories: $\alpha_1, \alpha_2 : [0,1] \to M$, the distance d_x between them can be calculated by

$$d_x(\alpha_1, \alpha_2) = \int_0^1 d_m\left(\alpha_1(t), \alpha_2(t)\right) dt. \tag{4}$$

This quantity gives a natural extension of d_m from M to $M^{[0,1]}$. Nonetheless, it suffers from the issue that $d_x(\alpha_1, \alpha_2) \neq d_x(\alpha_1 \circ \gamma_1, \alpha_2 \circ \gamma_2)$. For the mission of gesture recognition, as discussed in Section I, temporal dynamics is a central problem that needs to be resolved when a trajectory (gesture) α is observed as $\alpha \circ \gamma$, at a random temporal evolution γ. In other words, for arbitrary temporal re-parameterizations γ_1, γ_2 and arbitrary trajectories α_1, α_2, a distance $d(\cdot, \cdot)$ is needed that enables

$$d(\alpha_1, \alpha_2) = d(\alpha_1 \circ \gamma_1, \alpha_2 \circ \gamma_2). \tag{5}$$

Thanks to the square root velocity (SRV) framework [46], the theory of elastic trajectories is especially well adapted to our target. Inspired by [45], we expand the original Euclidean metric-based SRV to the manifold space-based TSRVF. In general, for a smooth trajectory $\alpha \in \mathcal{M}$, the TSRVF is a parallel propagation of a scaled velocity vector field of α to a reference point $c \in M$ according to

$$h_\alpha(t) = \frac{\dot{\alpha}(t)_{\alpha(t) \to c}}{\sqrt{|\dot{\alpha}(t)|}} \in T_c(M), \tag{6}$$

where $\dot{\alpha}(t)$ is the velocity vector along the trajectory at time t, and $\dot{\alpha}(t)_{\alpha(t) \to c}$ is its transport from the point $\alpha(t)$ to c along a geodesic path, and $|\cdot|$ means the norm related to the Riemannian metric on M and $T_c(M)$ denotes the tangent space of M at c. In particular, if $|\dot{\alpha}(t)| = 0$, $h_\alpha(t) = 0 \in T_c(M)$. Let $\mathcal{H} \subset T_c(M)^{[0,1]}$ be the set of smooth curves in $T_c(M)$ obtained as the TSRVFs of trajectories in M, $\mathcal{H} = \{h_\alpha | \alpha \in \mathcal{M}\}$ [45]. Two trajectories such as α_1 and α_2, can be mapped into the tangent space $T_c(M)$, as two equivalent TSRVFs, h_{α_1} and h_{α_2}. The distance between them can be determined in the standard vector space by the ℓ_2-norm,

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

The main motivation for the representation of a TSRVF actually comes from the following fact. If a trajectory α is warped by γ, to result in $\alpha \circ \gamma$, the TSRVF of $\alpha \circ \gamma$ is given by

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

As such, for any $\alpha_1, \alpha_2 \in \mathcal{M}$ and $\gamma \in \Gamma$, the distance d_h satisfies

$$\begin{aligned} d_h(h_{\alpha_1 \circ \gamma}, h_{\alpha_2 \circ \gamma}) &= \sqrt{\int_0^1 |h_{\alpha_1}(s) - h_{\alpha_2}(s)|^2 ds} \\ &= d_h(h_{\alpha_1}, h_{\alpha_2}), \end{aligned} \tag{9}$$

where $s = \gamma(t)$. The concerned reader is redirected to [45] [46] for the proof of the equality. From the geometric point of view, this equality means that the action of Γ on $\mathcal{H}$ under the ℓ_2 metric is by isometries. This helps us to establish a fully invariant distance to time-warping and use it to properly register trajectories [45]. Furthermore, for statistical analyses such as sample means and covariances, this invariability of execution rates is important. So we define the equivalence class $[h_\alpha]$ (or the notation $[\alpha]$) to denote the set of all trajectories equal to a given $h_\alpha \in \mathcal{H}$ (or $\alpha \in \mathcal{M}$), as

$$[h_\alpha] = \{h_{\alpha \circ \gamma} | \gamma \in \Gamma\}. \tag{10}$$

Definitely, such an equivalent class $[h_\alpha]$ (or $[\alpha]$) is associated with a category of gesture. Under this scheme, the process of the comparison of two trajectories is done by comparing their equivalence classes. In other words, an optimal reparameterization γ^* needs to be obtained to minimize the cost function $d_h(h_{\alpha_1}, h_{\alpha_2 \circ \gamma})$. Let $\mathcal{H}/\sim$ be the corresponding quotient space that can be bijectively identified with the set $\mathcal{M}/\sim$ using $[h_\alpha] \mapsto [\alpha]$. The distance d_s on $\mathcal{H}/\sim$ (or $\mathcal{M}/\sim$) is the shortest d_h distance between equivalence classes in $\mathcal{H}$ [45], given by

$$\begin{aligned} &d_s([\alpha_1], [\alpha_2]) \equiv d_s([h_{\alpha_1}], [h_{\alpha_2}]) = \inf_{\gamma \in \Gamma} d_h(h_{\alpha_1}, h_{\alpha_2 \circ \gamma}) \\ &= \inf_{\gamma \in \Gamma} \left(\int_0^1 |h_{\alpha_1}(t) - h_{\alpha_2}(\gamma(t)) \sqrt{\dot{\gamma}(t)}|^2 dt \right)^{1/2}. \end{aligned} \tag{11}$$

In practice, the minimization over Γ is solved for using dynamic programming. In this paper, we give a brief overview of SRV and TSRVF. Interested readers are refer to the papers [45] [46].

B. The TSRVF on product space of rotation group

One may find that the reference point c is an important parameter of a TSRVF, which should remain unchanged throughout the whole cycle of computation. Because the choice of c could potentially affect the outcome, a point is usually a good candidate for c if most trajectories pass near it. In this paper, the Karcher mean [58] as the Riemannian center of mass is employed, as it is equally distant from all points, thereby minimizing the possible distortions.

Given a set of $\{\alpha_i(t)_{t=1,..,n}\}_{i=1}^m$ of sequences (trajectories) of gestures (or actions), its Karcher mean $\mu(t)$ is computed using TSRVF representation with respect to d_s in $\mathcal{H}/\sim$, defined as

$$h_\mu = \arg \min_{[h_\alpha] \in \mathcal{H}/\sim} \sum_{i=1}^m d_s([h_\alpha], [h_{\alpha_i}])^2. \tag{12}$$

As a consequence, each trajectory is recursively aligned to the mean $\mu(t)$, so another product of the Karcher mean computation is the set of aligned trajectories $\{\tilde{\alpha}_i(t)_{t=1,...,n}\}_{i=1}^m$. Then, the shooting vector $v_i(t) \in T_{\mu(t)}(M)$ is calculated for each aligned trajectory $\tilde{\alpha}_i(t)$ at time t so that a geodesic goes from $\mu(t)$ to $\tilde{\alpha}_i(t)$ in unit time with the initial velocity $v_i(t)$, as

$$v_i(t) = \exp_{\mu(t)}^{-1}(\tilde{\alpha}_i(t)). \tag{13}$$

Finally, we combine the shooting vectors as $V(i) = [v_i(1)^{\mathrm{T}}\ v_i(2)^{\mathrm{T}}\ ...\ v_i(n)^{\mathrm{T}}]^{\mathrm{T}}$, which is the feature representation of a trajectory α_i.

V. Sparse Coding of 3D Skeletal Trajectories

From the above we can conclude that the feature of a trajectory (gesture sequence) is situated in a high dimensional space. The PCA, such as that of the applied methods of [45] [46], is a common solution to reduce dimensions. Nonetheless, PCA is an unsupervised learning model without knowing label information. Compared to component analysis techniques, a sparse coding representation with labeled training is more capable of capturing underlying associations between the input data and their labels. To the best of our knowledge, few manifold representation-based models considered the connection between labels and dictionary training. In this paper, we aim to associate label information with each dictionary atom to enforce the discriminability in sparse codes during the dictionary learning.

Specifically, given a set of observations (feature vectors of gestures) $\mathcal{Y} = \{y_i\}_{i=1}^N$, where $y_i \in \mathbb{R}^n$, let $\mathcal{D} = \{d_i\}_{i=1}^K$ be a set of vectors in $\mathbb{R}^n$ denoting a dictionary of K atoms, the learning of dictionary $\mathcal{D}$ for sparse representation of $\mathcal{Y}$ can be described as

$$< \mathcal{D}, X >= \arg\min_{\mathcal{D},X} \|\mathcal{Y} - \mathcal{D}X\|_2^2 \qquad s.t.\ \forall i,\ \|x_i\|_0 \leq S, \tag{14}$$

where $X = [x_1, ..., x_N] \in \mathbb{R}^{K \times N}$ means the sparse codes of observation $\mathcal{Y}$, and S is a sparsity constraint factor. The building of D is accomplished by minimizing the reconstruction error $\|\mathcal{Y} - \mathcal{D}X\|_2^2$, and satisfying the sparsity constraints. The K-SVD [59] algorithm is a widely used approach to (14).

In this paper, the classification error and regularization of label consistency are introduced into the objective function

$$\begin{aligned} &< \mathcal{D}, W, A, X >= \arg\min_{\mathcal{D},W,A,X} \|\mathcal{Y} - \mathcal{D}X\|_2^2 \\ &+ \beta\|L - WX\|_2^2 + \tau\|Q - AX\|_2^2 \qquad s.t.\ \forall i,\ \|x_i\|_0 \leq S, \end{aligned} \tag{15}$$

where $W \in \mathbb{R}^{C \times K}$ represents the parameters of classifier, and C corresponds to the number of categories. $L = [l_1, ..., l_N] \in \mathbb{R}^{C \times N}$ denotes the class labels of observation $\mathcal{Y}$, and $l_i = [0, ..., 1, ..., 0]^{\mathrm{T}} \in \mathbb{R}^C$ is a label vector corresponding to an observation y_i, where the nonzero position (index) shows the category of y_i. The additional term $\|L - WX\|_2^2$ is then used to denote the classification error for label information.

The final term is $\|Q - AX\|_2^2$, where $Q = [q_1, ..., q_N] \in \mathbb{R}^{K \times N}$ and $q_i = [0, ..., 1, ..., 1, ..., 0]^{\mathrm{T}} \in \mathbb{R}^K$ is a sparse code referring to an observation y_i for classification. The aim of setting nonzero elements is to enforce the "discriminative" of sparse codes. It is noted that the nonzero elements of q_i occur at those indices where the corresponding dictionary atom d_n shares the same label with the observation y_i. The A denotes a $K \times K$ transformation matrix, which is used to convert the original sparse code x into a discriminative one. Thereby, the term $\|Q - AX\|_2^2$ reflects the discriminative sparse code error, which enforces that the transformed sparse codes AX approximates the discriminative sparse codes Q. This operation forces the signals from the same category to have similar sparse representations. The regularization parameters β and τ govern the relative contributions of the corresponding terms. Equation (15) can be rewritten as

$$\begin{aligned} &< \mathcal{D}, W, A, X >= \\ &\arg\min_{\mathcal{D},W,A,X} \left\| \begin{pmatrix} \mathcal{Y} \\ \sqrt{\beta}L \\ \sqrt{\tau}Q \end{pmatrix} - \begin{pmatrix} \mathcal{D} \\ \sqrt{\beta}W \\ \sqrt{\tau}A \end{pmatrix} X \right\|_2^2 s.t.\ \forall i,\ \|x_i\|_0 \leq S. \end{aligned} \tag{16}$$

Here, we set $\mathcal{Y}' = (\mathcal{Y}^{\mathrm{T}}, \sqrt{\beta}L^{\mathrm{T}}, \sqrt{\tau}Q^{\mathrm{T}})^{\mathrm{T}}$, $\mathcal{D}' = (\mathcal{Y}^{\mathrm{T}}, \sqrt{\beta}W^{\mathrm{T}}, \sqrt{\tau}A^{\mathrm{T}})^{\mathrm{T}}$. Then, the optimization of Equation (16) is equivalent to solving (14) (replace $\mathcal{Y}$ and $\mathcal{D}$ with $\mathcal{Y}'$ and $\mathcal{D}'$ respectively). This is just the problem that K-SVD [59] handles. In this paper, a similar initialization and optimization solution of K-SVD to that described in [64] is adopted. In our experiments, the maximum iteration is set to 60, and the sparsity factor $S = 50$ is used. The β and τ are both set to 1.0.

VI. Experiments

In this section, the proposed 3D skeletal gesture recognition model is evaluated in comparison to the state-of-the-art methods using six public benchmarks, including sign language gestures: ChaLearn 2014 gesture [60]; controlled activities: MSR Action3D [61], UTKinect-Action3D [12], and Florence 3D Action [63]; more natural daily activities: MSR-DailyActivity3D [62]; and a large-scale dataset with different view-variations: NTU RGB+D [33]. The basic information of these datasets is summarized in Table I.

A. Experimental settings

In order to testify the effectiveness of the proposed method, 30 state-of-the-art algorithms, simply categorized into three groups, are compared.

The first group is a group of the methods most related to ours, including five Lie group representation-based algorithms: Lie group using DTW [4] (Lie group-DTW), Lie group with TSRVF [45] (Lie group-TSRVF), Lie group with TSRVF and using PCA for dimensionality reduction [44] (Lie group-TSRVF-PCA), Lie group with TSRVF and K-SVD for sparse coding [59] (Lie group-TSRVF-KSVD), and the Lie group with deep learning (LieNet) [51], as well as two TSRVF-related methods, the body part features with SRV and k-nearest neighbors clustering [47] (SRV-KNN), and TSRVF on Kendall's shape [5] (Kendall-TSRVF). In addition, two recent manifold-based methods, namely the Kendall-SCDL [49] and Gramian matrices [54], are compared.

TABLE I
THE BASIC INFORMATION OF THE SIX BENCHMARK DATASETS.

Dataset	Instances #	Classes #	Subjects #	Protocols
ChaLearn 2014 gesture [60]	13 585	20	27	Training: 7 754; validation: 3 362; testing: 2 742.
MSR Action3D [61]	567	20	10	Cross-Subject.
UTKinect-Action3D [12]	200	10	10	Leave-One-Sequence-Out.
MSR-DailyActivity3D [62]	320	16	10	Cross-Subject.
Florence 3D Action [63]	215	9	10	Leave-One-Subject-Out.
NTU RGB+D [33]	56 880	60	40	Cross-Subject and Cross-View.

Fig. 2. 20 gesture frames (with meanings in Italian and English) sampled from the ChaLearn 2014 [60] dataset.

The methods in the second group are based on classic feature representations, like HOJ3D [12], EigenJoints [13], actionlet ensemble (Actionlet) [17], HON4D [15], discriminative key-frames (Key-frames) [14], RVV with DTW (RVV-DTW) [10], and spatio-temporal naive Bayes nearest-neighbor (ST-NBNN) [25].

The last group includes fourteen deep learning methods, the HMM with a deep belief network (HMM-DBN) [6] and its extension (HMM-DBN-ext) [27], and four RNN-based approaches, namely LSTM [65], Deep LSTM [33], hierarchical RNN (HBRNN) [7], and spatio-temporal LSTM with trust gates (ST-LSTM-TG) [8], as well as eight CNN-based models: ModDrop (CNN) [28], Res-TCN [40], SkeleMotion [39], Clips-CNN-MTLN [29], STGC [41], GVFE and DH-TCN modules [43] incorporated with an ST-GCN [42] (GVFE + ST-GCN w/ DH-TCN), HCN [35], and AGC-LSTM [37] (please note that AGC-LSTM is a hybrid CNN-RNN architecture). These baseline results are collected from their original reports. Please note that several of the compared methods, like HMM-DBN-ext utilize both RGB-D and skeletal data, while the proposed method is based only on the 3D skeleton.

To analyze the effectiveness of the TSRVF on a product space of $SO(3) \times \cdots \times SO(3)$ (SO3-TSRVF), we report its discriminative output without any further steps (such as PCA or sparse coding) on six datasets. We aim to compare the ability of dictionary learning. We also present the results of the classical coding, such as K-SVD [59] (SO3-TSRVF-KSVD) and the proposed sparse coding scheme (SO3-TSRVF-SC). For a fair comparison, we follow the same identification system as in [4] [5] [44] [45] [49] [59], in other words, an one-vs-all linear SVM classifier (with the parameter C set to 1.0) is employed.

TABLE II
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON CHALEARN 2014 [60] DATASET (BEST: BOLD, SECOND BEST: UNDERLINED). * THE METHODS USE SKELETON AND RGB-D DATA.

Methods	Accuracy
Lie group-DTW [4]	79.2
Lie group-TSRVF [45]	91.8
Lie group-TSRVF-PCA [44]	90.4
Lie group-TSRVF-KSVD [59]	91.5
EigenJoints [13]	59.3
ModDrop (CNN) [28]*	<u>93.1</u>
HMM-DBN [6]	83.6
HMM-DBN-ext [27]*	86.4
LSTM [65]	82.0
Ours (SO3-TSRVF)	92.1
Ours (SO3-TSRVF-KSVD)	92.8
Ours (SO3-TSRVF-SC)	**93.2**

B. ChaLearn 2014 gesture dataset

The ChaLearn 2014 [60] is a gesture dataset with multi-modality data, including RGB, depth, human body masks, and 3D skeletal joints from 27 subjects. This dataset collects 13 585 gesture video segments (Italian cultural gestures) from 20 classes. Fig. 2 displays frames sampled from each category of gesture. We adopt the evaluation protocol provided by the dataset which assigns 7 754 gesture sequences for training, 3 362 sequences for validation, and 2 742 sequences for testing. To the best of our knowledge, ChaLearn 2014 is one of

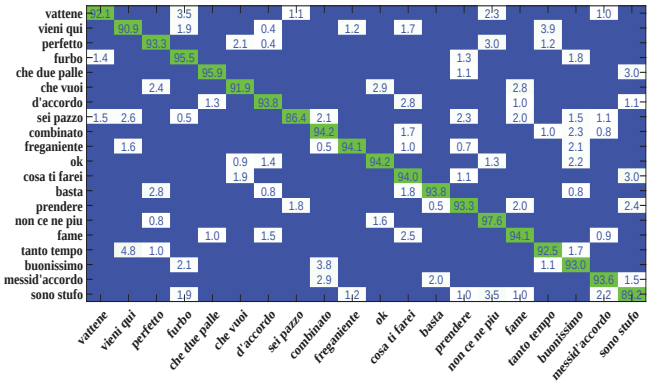

Fig. 3. A confusion matrix of the proposed method on ChaLearn 2014 [60] gesture dataset.

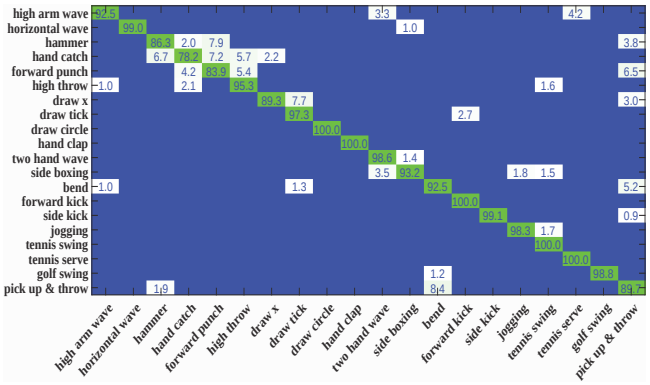

Fig. 4. A confusion matrix of the proposed method on MSR Action3D [61] dataset.

TABLE III
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON MSR ACTION3D [61] DATASET (BEST: BOLD, SECOND BEST: UNDERLINED). * THE METHODS USE SKELETON AND RGB-D DATA.

Methods	Accuracy
Lie group-DTW [4]	92.5
Lie group-TSRVF [45]	87.7
Lie group-TSRVF-PCA [44]	88.3
Lie group-TSRVF-KSVD [59]	87.6
SRV-KNN [47]	92.1
Kendall-TSRVF [5]	89.9
Kendall-SCDL [49]	94.2
EigenJoints [13]	82.3
Actionlet [17]*	88.2
HOJ3D [12]	78.9
HON4D [15]*	88.9
Key-frames [14]	91.7
RVV-DTW [10]	93.4
ST-NBNN [25]	**94.8**
HMM-DBN [6]	82.0
LSTM [65]	88.9
HBRNN [7]	94.5
ST-LSTM-TG [8]	**94.8**
Ours (SO3-TSRVF)	93.4
Ours (SO3-TSRVF-KSVD)	93.7
Ours (SO3-TSRVF-SC)	<u>94.6</u>

the largest gesture datasets, so the aim of this challenge is to evaluate the proposed method for sign gestures based on the given test and training sequences. Table II demonstrates the detailed comparison with other methods. It can be seen that the proposed method achieves the highest recognition accuracy as 93.2%. The experimental results show the efficacy of SO3-TSRVF compared to Lie group-based approaches. It is noted that Lie group-DTW [4] only gets 79.2%, which is due to the performance of DTW being highly dependent on the reference sequences for each category and that empirical choice becomes complicated as the size of the dataset grows bigger. It can also be observed that the accuracy of LSTM [65] is 11 percentage points lower than the proposed method. Despite the fact that LSTM is designed for perceiving the contextual information, modeling the sequence with temporal dynamics is still difficult, particularly when the size of the training data is small. It is important to mention that ModDrop [28] was placed first in the Looking at People Challenge [60], which uses the ChaLearn 2014 dataset as the benchmark. Please note that, without using RGB-D and audio data, our system could achieve a higher score than ModDrop.

The confusion matrix is shown in Fig. 3 to present the accuracy of the proposed method for individual gestures. As can be seen, for most categories, the proposed method achieves high precision. There is some confusion between similar gestures with very small values, like the *tanto tempo* (a long time a go) and *vieni qui* (come here) gestures, as well as the *furbo* (clever) and *buonissimo* (very good) gestures.

C. MSR Action3D dataset

The MSR Action3D [61] is a dataset that is widely used to evaluate action recognition efficiency. MSR Action3D is very challenging where actions are highly similar to each other (e.g., *hammer* and *hand catch*) and have a typical large temporal misalignment. This dataset consists of 567 pre-segmented action instances, and 10 individuals executing 20 action classes. The MSR Action3D dataset is so popular that many researchers have reported their results using it. The same evaluation protocol is adopted for a fair comparison, namely the *Cross-Subject* testing as defined in [61], where half of the subjects are used for training (subjects numbers 1, 3, 5, 7, 9) and the rest are used for testing (2, 4, 6, 8, 10). We compare the proposed method with the state-of-the-art methods, the recognition accuracies on the MSR Action3D dataset are recorded in Table III. We can see that the proposed method achieves better performance than both Lie group-based and classical feature representation approaches. Again, the performance of the proposed sparse coding is superior to K-SVD coding-based methods. The accuracy of the proposed method is only 0.2% lower than the ST-LSTM-TG [8]. This shows that our approach performs a bit worse than the deep learning model with an ample size of data for training network parameters, while the score of ST-LSTM-TG on the UTKinect-Action3D [12] dataset is lower than ours (see Table IV). Also, the performance of the proposed method is slightly lower than the ST-NBNN [25]. In fact, the ST-NBNN is based on naive-Bayes nearest-neighbor (NBNN) distance matrices, and a specifically-designed tensor SVM is introduced to improve classification accuracy. However,

TABLE IV
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON UTKINECT-ACTION3D [12] DATASET (BEST: BOLD, SECOND BEST: UNDERLINED). * THE METHODS USE SKELETON AND RGB-D DATA.

Methods	Accuracy
Lie group-DTW [4]	97.1
Lie group-TSRVF [45]	94.5
Lie group-TSRVF-PCA [44]	94.9
Lie group-TSRVF-KSVD [59]	92.7
SRV-KNN [47]	91.5
Kendall-TSRVF [5]	89.8
Gramian matrix [54]	96.5
Kendall-SCDL [49]	<u>97.5</u>
EigenJoints [13]	92.4
HOJ3D [12]	90.9
HON4D [15]*	90.9
ST-NBNN [25]	**98.0**
ST-LSTM-TG [8]	97.0
Ours (SO3-TSRVF)	96.8
Ours (SO3-TSRVF-KSVD)	97.2
Ours (SO3-TSRVF-SC)	<u>97.5</u>

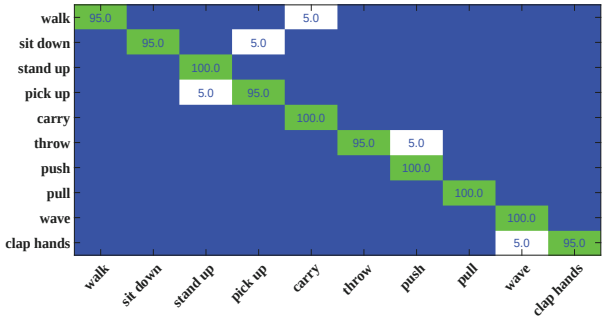

Fig. 5. A confusion matrix of the proposed method on UTKinect-Action3D [12] dataset.

as mentioned when discussing the experimental settings, we only employ the linear SVM for classification, as with many comparative methods [4] [5] [44] [45] [59]. Actually, as reported in [25], with the same setting namely the NBNN with weight learning using linear SVM (NBNN+SVM) yields a worse score (92.4%) than ours. It is worth mentioning that the authors of [49] also offered the accuracy of Kendall-SCDL with the Bi-LSTM classifier (for both temporal modeling and classification). For a fair comparison, we only report the results of Kendall-SCDL with a linear SVM classifier in Table III (and in Tables IV and V). In Fig. 4, the accuracy of each action in the form of a confusion matrix is given. It can be found that the proposed method works very well on the MSR Action3D dataset.

D. UTKinect-Action3D dataset

The multi-modal human action dataset UTKinect-Action3D [12] is a difficult benchmark due to its high intra-class variations. Another challenge of this dataset is the variations in the viewpoint, and there are occlusions caused by the absence of some body-parts in the sensor's field of view. This dataset uses Kinect to capture 10 classes of actions. Each action is carried out twice by 10 subjects. As a consequence, a total of 200 action instances are collected in 20 videos. The available modalities are RGB images, depth maps, and skeletal joints. We obey the *Leave-One-Sequence-Out Cross Validation* setting of [12] that selects each sequence as the testing sample; in turn, regards other sequences as training samples and computes the average recognition rate (20 rounds of testing). Table IV reports the comparisons of the proposed method with some representative state-of-the-art methods. It is noticeable that the proposed method can yield superior performance over the deep learning model ST-LSTM-TG [8]. This is not surprising because a large number of sequences are required for training such an RNN-based network, but compared to a dataset like MSR Action3D [61], the size of UTKinect-Action3D is rather small. It can be seen that our solution outperforms all approaches except the ST-NBNN [25]. This indicates that our method, using relative geometry (rotations), can handle viewpoint variations very well. In the last subsection, we discussed that the ST-NBNN may benefit from the tensor SVM classifier. As reported in [25], the recognition accuracy of NBNN+SVM on UTKinect-Action3D is only 94%. It is noted that the Gramian matrix [54] reported its scores not only with full body but also with the body parts fusion (BP fusion). Obviously, using a late fusion of classifiers (based on the body parts) would be beneficial and boost performance. For a fair comparison, the result of the Gramian matrix with the full-body (the whole skeleton) is given in Table IV (and in Table V). In addition, the confusion matrix of the proposed method is shown in Fig. 5. Clearly, in all the cases, a good accuracy score is achieved for all activities.

TABLE V
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON FLORENCE 3D ACTION [63] DATASET (BEST: BOLD, SECOND BEST: UNDERLINED).

Methods	Accuracy
Lie group-DTW [4]	90.9
Lie group-TSRVF [45]	89.5
Lie group-TSRVF-PCA [44]	89.7
Lie group-TSRVF-KSVD [59]	89.6
SRV-KNN [47]	87.0
Gramian matrix [54]	88.1
Kendall-SCDL [49]	92.3
DBN-HMM [6]	87.5
LSTM [65]	86.2
Ours (SO3-TSRVF)	90.8
Ours (SO3-TSRVF-KSVD)	<u>91.9</u>
Ours (SO3-TSRVF-SC)	**93.5**

E. Florence 3D Action dataset

We also test the proposed method on the popular Florence 3D Action dataset [63]. The Florence 3D Action dataset collected nine classes of action. Each action is carried out for two or three times by 10 subjects. As a result, a total of 215 action sequences are captured. This dataset includes two inputs: RGB frames and the 3D coordinates of skeleton

TABLE VI
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON MSR-DAILYACTIVITY3D [62] DATASET (BEST: BOLD, SECOND BEST: UNDERLINE). * THE METHODS USE SKELETON AND RGB-D DATA.

Methods	Accuracy
Lie group-DTW [4]	90.3
Lie group-TSRVF [45]	88.9
Lie group-TSRVF-PCA [44]	90.3
Lie group-TSRVF-KSVD [59]	89.5
Kendall-TSRVF [5]	70.0
Actionlet [17]*	85.5
HON4D [15]*	80.0
Key-frames [14]	73.8
Ours (SO3-TSRVF)	89.7
Ours (SO3-TSRVF-KSVD)	<u>90.5</u>
Ours (SO3-TSRVF-SC)	**91.2**

TABLE VII
A COMPARISON OF RECOGNITION ACCURACY (%) WITH EXISTING SKELETON-BASED METHODS ON NTU RGB+D [33] DATASET (BEST: BOLD, SECOND BEST: UNDERLINED).

Methods	Accuracy	
	Cross-Subject	Cross-View
Lie group-DTW [4]	50.1	52.8
LieNet [51]	61.4	67.0
Kendall-SCDL [49]	73.9	83.0
HOJ3D [12]	32.4	22.3
HON4D [15]	30.6	7.3
HBRNN [7]	59.1	64.0
Deep LSTM [33]	60.7	67.3
ST-LSTM-TG [8]	69.2	77.7
Res-TCN [40]	74.3	83.1
SkeleMotion [39]	76.5	84.7
Clips-CNN-MTLN [29]	79.6	84.8
STGC [41]	74.9	86.3
GVFE + ST-GCN [42] w/ DH-TCN [43]	79.1	88.2
HCN [35]	<u>86.5</u>	<u>91.1</u>
AGC-LSTM [37]	**87.5**	**93.5**
Ours (SO3-TSRVF)	61.5	69.2
Ours (SO3-TSRVF-KSVD)	69.0	78.4
Ours (SO3-TSRVF-SC)	74.7	86.3

joints. The challenges of this dataset are the similarity between action classes and the high intra-class variations as the same action can be done by different hands. Due to a few skeletal joints (the skeletons are composed of 15 joints), some types of action are hard to distinguish, such as *drink from a bottle*, *answer phone* and *read watch*. To test the proposed method, we follow [63] and employ the same *Leave One Subject Out* experimental protocol. A complete comparison with previous studies is reported in Table V. Our approach achieved a classification accuracy of 93.5%, which is the highest score of all the comparative algorithms.

F. MSR-DailyActivity3D dataset

The MSR-DailyActivity3D dataset [63] is designed to cover daily activities that are recorded using a Kinect V1. The captured skeletons of MSR-DailyActivity3D are noisier than other datasets. In particular, many activities involve human-object interactions, such as the *use laptop* and *play guitar*, where joints are occurred by objects, thus the resulting estimations of joints are almost random. Another challenging part here is each subject performs an activity twice, once in a standing position and once in a sitting position. To evaluate the performance of the proposed method on such natural daily activities, we report the results on MSR-DailyActivity3D. This dataset collects 16 types of activity from 10 subjects. Each of the subjects performs an activity twice, so there are 320 sequences in total. We follow the dataset's evaluation protocol and apply the *Cross-Subject* setting to evaluate the proposed method. Namely, half of the subjects (IDs 1, 2, 3, 4, 5) are used for gathering training data, while the other half are used for gathering testing data. We summarize the classification accuracy results in Table VI. It can be seen that the proposed method achieved the highest score in all comparative methods. In fact, the existing approaches use either joint positions or the angles between bones to represent a human skeleton. In our work, we model each skeleton as the relative 3D rotations between all pairs of bones. Compared to the absolute coordinates or the angles of connected bones, our method considers any possible or latency relationships between bones (they may or may not be directly connected). For example, the relationship between the two hands is important for recognizing classes such as *read book* and *use laptop*. However, it is difficult for conventional methods to capture the dependency between the two hands since they are not connected physically. In particular, using only the rotations makes the skeletal representation scale-invariant, this is an important advantage when dealing with the noisy skeletons extracted with an RGB-D senor. Besides, to further limit the effect of noises, we follow [4] [44] and use the interpolation algorithm to perform the temporal smoothing on trajectories (please refer to [4] for the details).

G. NTU RGB+D dataset

The NTU RGB+D [33] (in total, 56 880 video clips) is currently one of the largest datasets with 3D skeletons, and it is the most widely used for testing indoor-captured action recognition. In order to provide a fair comparison with recent action/gesture recognition methods, we report the scores of the proposed approach on this dataset since most of the recent work has been evaluated on it. NTU RGB+D contains 60 action categories, and 40 subjects (performers) have attended the data collection. The skeletons were detected by the Kinect V2, and there are 25 joints for each subject. Each video has no more than two subjects. More specifically, NTU RGB+D includes single-actor actions, which are from class 1 to 49, and two-actor actions, which are from class 50 to 60. In video capturing, each action is recorded simultaneously by three cameras at the same height but with different horizontal angles: -45°, 0°, and +45°. As such, as something not merely provided the common *Cross-Subject* protocol, the authors of NTU RGB+D also recommended the *Cross-View* evaluation. Namely, in the *Cross-View* protocol, the training set contains 37 920 clips that were captured by cameras #2 and #3, and the validation set contains 18 960 clips from camera #1. In the *Cross-Subject* protocol, the training set contains 40 320 clips performed by 20 subjects, and the remaining 16 560 clips from the remaining 20 subjects are used for validation. We follow

TABLE VIII
THE COMPUTATIONAL EFFICIENCY OF EACH STEP IN THE PROPOSED METHOD (PROCESSING A SEQUENCE OF THE FLORENCE 3D ACTION DATASET [63]).

Pipeline Steps	Implementation	Time (sec)
Feature extraction	MATLAB	0.116
Karcher mean (Eq. (12))	MATLAB, C++	1.564
Shooting vector (Eq. (13))	MATLAB	0.037
Dictionary learning (Eq. (15))	MATLAB	0.249
Classification	MATLAB	0.001
In total		1.967

TABLE IX
A COMPARISON OF REPRESENTATIONAL DIMENSION WITH STATE-OF-THE-ART MANIFOLD-BASED METHODS ON MSR ACTION3D [61] DATASET.

Methods	Accuracy (%)	Dimension
Lie group-DTW [4]	92.5	155 952
Lie group-TSRVF [45]	87.7	155 952
Lie group-TSRVF-PCA [44]	88.3	250
SRV-KNN [47]	92.1	60 000
Ours (SO3-TSRVF)	93.4	77 976
Ours (SO3-TSRVF-SC)	94.6	50 (sparsity)

this convention and report the recognition accuracy of the two protocols.

The results for this dataset are presented in Table VII. Obviously, our approach (SO3-TSRVF-SC) outperforms other manifold-based models (Lie group-DTW [4], LieNet [51], and Kendall-SCDL [49]) for both protocols *Cross-View* and *Cross-Subject*. This could prove our method's ability to handle large-scale datasets compared to conventional manifold approaches. It is noted that our method obtained better scores than approaches based on the RNN. Although the proposed method is not based on deep learning, our performance is competitive with the *Cross-View* protocol, even when compared to the dominant CNN-based models. More specifically, our method is superior (or equal) to the Res-TCN, SkeleMotion, Clips-CNN-MTLN, and STGC, with the exception of [43] [35] [37]. The authors of [37] also provided the score by the late fusion of joint-based (full-body) and part-based AGC-LSTM. However, most of the compared methods only use full-body joints. For a fair comparison, the result of the joint-based AGC-LSTM is reported in Table VII. In our method, we explicitly model the 3D geometric relationships between bones using the relative rotations-based $SO(3)$. This feature has natural stability and consistency. For example, if a pair of bones undergo the same rotation, their relative geometry matrix is not altered. Also, this feature can be invariant to the subject's orientation in the scene. The evolution results of the *Cross-View* challenge have demonstrated the efficiency of our method (SO3-TSRVF), which outperforms all other manifold-related models. For instance, it exceeds LieNet [51]. As reported in [49], using the raw data of Kendall's shape space representations (without sparse coding SCDL), the classification accuracy is 56.5%, obviously inferior to ours. Also, the performance of SO3-TSRVF is better than an RNN-based HBRNN and Deep LSTM. Compared to the absolute locations of the skeleton, the relative rotations provide a more meaningful description, and the *Cross-View* testing could clearly benefit.

In addition, with the *Cross-View* and *Cross-Subject* protocols, the SO3-TSRVF after a sparse coding with labeled learning, is remarkably boosted (by 13% and 17%). This demonstrates the efficiency of our model on the large-scale dataset that the resulting codes are more discriminative than the original data.

H. Computational efficiency

The computational efficiency of the proposed method was evaluated on a PC with Intel Core i7 CPU and 16 GB RAM. Our method has been implemented in MATLAB and C++. In Table. VIII, we report the average processing time of each step performed on the Florence 3D Action dataset, where the average length of sequences is 35 frames. It is noted that only the temporal alignment (Eq. (11)) is a C++ implementation of the dynamic programming algorithm, while other steps are implemented in MATLAB. As presented in Table VIII, during the training, the average time of the proposed method required to process a (skeletal) sequence is 1.967 seconds. In the same experimental setting, Lie group-DTW [4] spends more than six seconds classifying an action. This is not surprising since Lie group-DTW relies on the time-consuming steps of DTW and FTP, which sum up to costing around five seconds per sequence. There is no dictionary learning procedure in the Lie group-TSRVF [45], its representational dimension is 38 220, which is 764 times over the dimension of our SO3-TSRVF-SC (with sparsity $S = 50$), please see the following analysis of the models' representational dimension for details. The computational complexity of the (training) linear SVM is $O(dn)$, where n and d are the number and dimension of samples, respectively. Moreover, if the samples have extremely sparse feature vectors with the sparsity S, the computational complexity turns to $O(Sn)$ [66]. SO3-TSRVF-SC only spends 0.001 seconds for the classification, while the same step of the Lie group-TSRVF costs about 0.8 seconds. As a result, the whole computational cost of Lie group-TSRVF is higher than that of our method. The computational time of Lie group-TSRVF-PCA [44] and Lie group-TSRVF-KSVD [59] are similar to ours. However, their performances are obviously inferior to the SO3-TSRVF-SC, as shown in Table V where their accuracies are nearly 4 percentage points lower than our method. As noted in SRV-KNN [47], its processing speed is faster than the proposed method, but the classification performance of SO3-TSRVF-SC surpasses SRV-KNN by 6.5%. Because the implementations of the Gramian matrix [54] and Kendall-SCDL [49] are not publicly available, we cannot present their computational costs. Nevertheless, as reported for the Kendall-TSRVF [5], where the model is also based on the Kendall's shape space, its average time of processing a skeletal sequence is 2.85 seconds (on a 3.1 GHz CPU machine). As we know, deep learning methods can reduce the need for feature engineering. However, they require a long time to train a huge amount of network parameters. For example, LSTM [65] needs more than two hours to accomplish the training on the Florence 3D Action dataset (215 action sequences).

Obviously, in the proposed method, the dominant cost comes from the computation of the Karcher mean. The multi-kernel GPU can be utilized to speed up our model via parallelized processing on sequences since the computational time of the Karcher mean depends on the number of sequences. After having obtained the Karcher mean, in the testing stage, the processing time required to classify a sequence is less than half a second, which points to a real-time system potential. To evaluate the observational latency of the proposed method, we report how the accuracy depends on the duration of observation. Here, this latency is analyzed on the MSR Action3D dataset [61], where the accuracy is computed by processing only a fraction of the sequence, such as observing 25%, 50%, and 75% of the frames (skeletons). Correspondingly, the classification accuracies are 59.7%, 86.4%, and 92.6%. The results have shown that an accuracy close to the best is obtained by processing just 75% of the sequences. We can also notice that even only half of the sequence is sufficient to guarantee an accuracy over 85%.

As we know, the key target of coding is to reduce the complexity of search and retrieval in the latent spaces. In this paper, we propose a sparse coding scheme to learn a dictionary with the discriminative and representative atoms. Here, to further verify the efficiency of our coding method, we compare the representational dimension with the state-of-the-art methods. As mentioned at the end of Section IV, in our model, the feature representation of a video sequence is the combined shooting vectors (SO3-TSRVF) of a skeletal trajectory. Because a shooting vector of $SO(3)$ is in a tangent space $\mathbb{R}^3$, the Euclidean representation of a human skeleton is in the space $\mathbb{R}^{6C_M^2}$ (please see the end of Section III). If a video sequence has $\mathcal{N}$ frames (a trajectory with $\mathcal{N}$ points), then the eventual dimension of our feature vector is $6C_M^2 \times \mathcal{N}$. Let us take the MSR Action3D [61] dataset for example, each skeleton has 19 bones and 20 joints, so $M = 19$. Following the setting of [4] [44], all the (action) sequences in a dataset are interpolated to have the same length $\mathcal{N}$, which has been set to 76 for the MSR Action3D dataset. Therefore, the representational dimension of SO3-TSRVF is 77 976. Because the Lie algebra $\mathfrak{se}(3)$ is in the space $\mathbb{R}^6$, the dimension of Lie group-based models is 155 952, which is twice that of SO3-TSRVF. As noted in Section V, the sparsity constraint factor of the proposed sparse coding is set to 50, which means the representational dimension of our SO3-TSRVF-SC is much smaller than the SO3-TSRVF and Lie group-based methods. However, as reported on all the above datasets, the classification results of SO3-TSRVF-SC are always better than others. It can be concluded that the redundancy data can be reduced by our method to a discriminative sparse code with a low computational cost. Also, on MSR Action3D [61] dataset, we summarize the representational dimension (with classification accuracy) of state-of-the-art manifold-based methods in Table IX. Clearly, our method archived the highest score with the smallest representational dimension. These results proved that our method can learn meaningful relations between bones and discard the redundancies efficiently.

VII. Conclusion

A new human gesture recognition method is proposed in this paper. We represented a 3D human skeleton as a point in the product space of the special orthogonal group $SO(3)$, due to this, a human gesture can be described as a trajectory in the Riemannian manifold space. In order to consider re-parametrization invariance properties for trajectory analysis, we generalize the TSRVF to obtain a time-warping invariant metric for comparing trajectories. Furthermore, a sparse coding scheme of skeletal trajectories is proposed by carefully considering the labeling information with each atom to enforce the discriminant validity of the dictionary. Experiments show that the proposed method has achieved state-of-the-art performances. Possible directions for future work include researching end-to-end deep network architecture in the manifold space in order to address the issues of 3D skeletal gesture recognition.

References

[1] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time human pose recognition in parts from single depth images," *Commun. ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[2] L. L. Presti and M. La Cascia, "3D skeleton-based human action classification: A survey," *Pattern Recognit.*, vol. 53, pp. 130–147, 2016.

[3] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3D skeletal data: A review," *Comput. Vis. Image Underst.*, vol. 158, pp. 85–105, 2017.

[4] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a Lie group," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, 2014, pp. 588–595.

[5] B. B. Amor, J. Su, and A. Srivastava, "Action recognition using rate-invariant analysis of skeletal shape trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 1–13, 2016.

[6] D. Wu and L. Shao, "Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, 2014, pp. 724–731.

[7] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, 2015, pp. 1110–1118.

[8] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal LSTM with trust gates for 3D human action recognition," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 816–833.

[9] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention LSTM networks for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1647–1656.

[10] Y. Guo, Y. Li, and Z. Shao, "RRV: A spatiotemporal descriptor for rigid body motion recognition," *IEEE Trans. Cybern.*, 2017.

[11] X. Liu and G. Zhao, "3D skeletal gesture recognition via sparse coding of time-warping invariant Riemannian trajectories," *Int. Conf. Multimed. Modeling*, pp. 678–690, 2019.

[12] L. Xia, C. C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*. IEEE, 2012, pp. 20–27.

[13] X. Yang and Y. Tian, "Eigenjoints-based action recognition using naive-bayes-nearest-neighbor," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*. IEEE, 2012, pp. 14–19.

[14] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3D kinematics descriptor for low-latency action recognition and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2752–2759.

[15] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 716–723.

[16] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition," *J Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 24–38, 2014.

[17] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, 2014.

[18] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RBGD images," in *Proc. IEEE Conf. Robot. Autom.* IEEE, 2012, pp. 842–849.

[19] B. Packer, K. Saenko, and D. Koller, "A combined pose, object, and feature model for action understanding." in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1378–1385.

[20] F. Lv and R. Nevatia, "Recognition and segmentation of 3D human action using HMM and multi-class adaboost," *Proc. Eur. Conf. Comput. Vis.*, pp. 359–372, 2006.

[21] L. Piyathilaka and S. Kodagoda, "Gaussian mixture based HMM for human daily activity recognition using 3D skeleton features," in *Proc. IEEE Conf. Ind. Electron. Appl.* IEEE, 2013, pp. 567–572.

[22] H. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 14–29, 2016.

[23] C. Wu, J. Zhang, S. Savarese, and A. Saxena, "Watch-n-patch: Unsupervised understanding of actions and relations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4362–4370.

[24] X. Cai, W. Zhou, L. Wu, J. Luo, and H. Li, "Effective active skeleton representation for low latency human action recognition," *IEEE Trans. Multimed.*, vol. 18, no. 2, pp. 141–154, 2016.

[25] J. Weng, C. Weng, and J. Yuan, "Spatio-temporal naive-bayes nearest-neighbor (ST-NBNN) for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.

[26] L. Wang, D. Q. Huynh, and P. Koniusz, "A comparative review of recent Kinect-based action recognition algorithms," *IEEE Trans. Image Process.*, vol. 29, pp. 15–28, 2019.

[27] D. Wu, L. Pigou, P. J. Kindermans, N. Le, L. Shao, J. Dambre, and J. M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1583–1597, 2016.

[28] N. Neverova, C. Wolf, G. Taylor, and F. Nebout, "ModDrop: Adaptive multi-modal gesture recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1692–1706, 2016.

[29] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.

[30] B. Mahasseni and S. Todorovic, "Regularizing long short term memory with 3D human-skeleton sequences for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, June 2016, pp. 3054–3062.

[31] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, and J. Liu, "Online human action detection using joint classification-regression recurrent neural networks," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 203–220.

[32] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, X. Xie *et al.*, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks." in *Proc. AAAI Conf. Artif. Intell.*, vol. 2, 2016, p. 8.

[33] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1010–1019.

[34] S. Zhang, Y. Yang, J. Xiao, X. Liu, Y. Yang, D. Xie, and Y. Zhuang, "Fusing geometric features for skeleton-based action recognition using multilayer LSTM networks," *IEEE Trans. Multimed.*, vol. 20, no. 9, pp. 2330–2343, 2018.

[35] C. Li, Q. Zhong, D. Xie, and S. Pu, "Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 786–792.

[36] Z. Fan, X. Zhao, T. Lin, and H. Su, "Attention-based multiview re-observation fusion network for skeletal action recognition," *IEEE Trans. Multimed.*, vol. 21, no. 2, pp. 363–374, 2019.

[37] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1227–1236.

[38] X. Liu, H. Shi, X. Hong, H. Chen, D. Tao, and G. Zhao, "3D skeletal gesture recognition via hidden states exploration," *IEEE Trans. Image Process.*, vol. 29, pp. 4583–4597, 2020.

[39] C. Caetano, J. Sena, F. Brémond, J. A. Dos Santos, and W. R. Schwartz, "SkeleMotion: A new representation of skeleton joint sequences based on motion information for 3D action recognition," in *IEEE Int. Conf. Advan. Vid. Signal Surv.* IEEE, 2019, pp. 1–8.

[40] T. S. Kim and A. Reiter, "Interpretable 3D human action analysis with temporal convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*. IEEE, 2017, pp. 1623–1631.

[41] C. Li, Z. Cui, W. Zheng, C. Xu, and J. Yang, "Spatio-temporal graph convolution for skeleton based action recognition," in *AAAI Conf. Artif. Intell.*, 2018.

[42] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI Conf. Artif. Intell.*, 2018.

[43] K. Papadopoulos, E. Ghorbel, D. Aouada, and B. Ottersten, "Vertex feature encoding and hierarchical temporal modeling in a spatial-temporal graph convolutional network for action recognition," *arXiv preprint arXiv:1912.09745*, 2019.

[44] R. Anirudh, P. Turaga, J. Su, and A. Srivastava, "Elastic functional coding of human actions: From vector-fields to latent variables," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3147–3155.

[45] J. Su, S. Kurtek, E. Klassen, and A. Srivastava, "Statistical analysis of trajectories on Riemannian manifolds: bird migration, hurricane tracking and video surveillance," *Ann. Appl. Stat.*, pp. 530–552, 2014.

[46] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn, "Shape analysis of elastic curves in Euclidean spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 7, pp. 1415–1428, 2011.

[47] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo, "3D human action recognition by shape analysis of motion trajectories on Riemannian manifold," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1340–1352, 2015.

[48] J. Ho, Y. Xie, and B. Vemuri, "On a nonlinear generalization of sparse coding and dictionary learning," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1480–1488.

[49] A. B. Tanfous, H. Drira, and B. B. Amor, "Sparse coding of shape trajectories for facial expression and action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[50] P. T. Fletcher, C. Lu, S. M. Pizer, and S. Joshi, "Principal geodesic analysis for the study of nonlinear statistics of shape," *IEEE Trans. Med. Imaging*, vol. 23, no. 8, pp. 995–1005, 2004.

[51] Z. Huang, C. Wan, T. Probst, and L. Van Gool, "Deep learning on Lie groups for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017.

[52] M. T. Harandi, M. Salzmann, and R. Hartley, "From manifold to manifold: Geometry-aware dimensionality reduction for spd matrices," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 17–32.

[53] M. Harandi and M. Salzmann, "Riemannian coding and dictionary learning: Kernels to the rescue," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3926–3935.

[54] A. Kacem, M. Daoudi, B. B. Amor, S. Berretti, and J. C. Alvarez-Paiva, "A novel geometric framework on gram matrix trajectories for human behavior understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 1–14, 2020.

[55] B. Szczapa, M. Daoudi, S. Berretti, A. Del Bimbo, P. Pala, and E. Massart, "Fitting, comparison, and alignment of trajectories on positive semi-definite matrices with application to action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019.

[56] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[57] M. Bauer, M. Bruveris, and P. W. Michor, "Overview of the geometries of shape spaces and diffeomorphism groups," *J. Math. Imaging Vis.*, vol. 50, no. 1-2, pp. 60–97, 2014.

[58] H. Karcher, "Riemannian center of mass and mollifier smoothing," *Commun. Pure Appl. Math.*, vol. 30, no. 5, pp. 509–541, 1977.

[59] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[60] S. Escalera, X. Baró, J. Gonzalez, M. A. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H. J. Escalante, J. Shotton, and I. Guyon, "Chalearn looking at people challenge 2014: Dataset and results," in *Proc. Eur. Conf. Comput. Vis. Workshops*. Springer, 2014, pp. 459–473.

[61] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*. IEEE, 2010, pp. 9–14.

[62] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, 2012, pp. 1290–1297.

[63] L. Seidenari, V. Varano, S. Berretti, A. Bimbo, and P. Pala, "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2013, pp. 479–485.

[64] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent K-SVD: Learning a discriminative dictionary for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2651–2664, 2013.

[65] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[66] T. Joachims, "Training linear SVMs in linear time," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Disc Data Min.*, 2006, pp. 217–226.

Xin Liu (Member, IEEE) received the Ph.D. degree in information and communication engineering from Xi'an Jiaotong University, China, in 2016, and the Ph.D. degree in computer science from the University of Oulu, Finland, in 2019. He is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University, China, and a Senior Researcher with the Center for Machine Vision and Signal Analysis, University of Oulu. His research interests include human behavior recognition and analysis, emotion understanding, and video background subtraction. He has authored or coauthored more than 30 papers in prominent journals and conferences, and has served for prestigious conferences and journals, including the IEEE CVPR, ICCV, T-PAMI, T-IP, T-CSVT, T-NNLS, T-CI, IJCV, ACM TOMM, and PR.

Guoying Zhao (SM'12) is currently a Professor with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland, where she has been a senior researcher since 2005 and an Associate Professor since 2014. She received the Ph.D. degree in computer science from the Chinese Academy of Sciences, Beijing, China, in 2005. She has authored or co-authored more than 210 papers in journals and conferences. Her papers have currently over 12520 citations in Google Scholar (h-index 52). She was co-publicity chair for FG2018, General chair of 3rd International Conference on Biometric Engineering and Applications (ICBEA 2019), and Late Breaking Results Co-Chairs of 21st ACM International Conference on Multimodal Interaction (ICMI 2019), has served as area chairs for several conferences and is associate editor for Pattern Recognition, IEEE Transactions on Circuits and Systems for Video Technology, and Image and Vision Computing Journals. She has lectured tutorials at ICPR 2006, ICCV 2009, SCIA 2013 and FG 2018, authored/edited three books and eight special issues in journals. Dr. Zhao was a Co-Chair of many International Workshops at ICCV, CVPR, ECCV, ACCV and BMVC. Her current research interests include image and video descriptors, facial-expression and micro-expression recognition, gait analysis, dynamic-texture recognition, human motion analysis, and person identification. Her research has been reported by Finnish TV programs, newspapers and MIT Technology Review.