2014 22nd International Conference on Pattern Recognition

Skeletal Quads: Human Action Recognition Using Joint Quadruples

Georgios Evangelidis, Gurkirt Singh and Radu Horaud
INRIA Grenoble Rhône-Alpes
655, avenue de l'Europe
38330 Montbonnot Saint-Martin, FRANCE
email: firstname.lastname@inria.fr

***Abstract*—Recent advances on human motion analysis have made the extraction of human skeleton structure feasible, even from single depth images. This structure has been proven quite informative for discriminating actions in a recognition scenario. In this context, we propose a local skeleton descriptor that encodes the relative position of joint quadruples. Such a coding implies a similarity normalisation transform that leads to a compact (6D) view-invariant skeletal feature, referred to as *skeletal quad*. Further, the use of a Fisher kernel representation is suggested to describe the skeletal quads contained in a (sub)action. A Gaussian mixture model is learnt from training data, so that the generation of any set of quads is encoded by its Fisher vector. Finally, a multi-level representation of Fisher vectors leads to an action description that roughly carries the order of sub-action within each action sequence. Efficient classification is here achieved by linear SVMs. The proposed action representation is tested on widely used datasets, MSRAction3D and HDM05. The experimental evaluation shows that the proposed method outperforms state-of-the-art algorithms that rely only on joints, while it competes with methods that combine joints with extra cues.**

I. Introduction

Action recognition is an active topic in computer vision and pattern recognition, with many potential applications in human-computer interaction. Despite the advances in recent years, however, recognising human actions remains a challenging problem, mainly because of the articulated nature of human motion. Therefore, the discrimination of human postures and actions benefits from the segmentation of the body into parts. While this kind of segmentation remains a quite difficult task using monocular visual sensors, the release of depth sensors (e.g. Kinect) has simplified the pose estimation by means of 3D body joints [15].

Typically, an action recognition method employs an L-class classifier that performs a 1-of-L assignment to input vectors. By putting aside the classifier itself, what mostly differentiates the majority of the methods is the way of building the input of the classifier, i.e., the vector representation of a video segment. Commonly, multiple descriptors of raw data are summarised into a vector, e.g. through a Bag-of-Words (BoW) paradigm, in order to encode an action sequence.

Thanks to recent achievements [15], the detection of the human pose by means of skeleton joints is feasible even from single depth images. This implies a powerful representation for action recognition, since actions can be seen as a set of poses. However, when joints are pooled into global features, e.g., using all pairwise joint differences [21], the articulated nature of the human pose is not well encoded in the action descriptor. Therefore, local skeleton features are more meaningful.

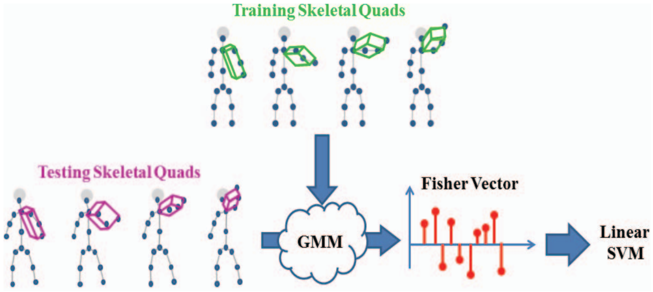

Fig. 1. A Gaussian Mixture Model (GMM), learnt on training data, is supposed to generate skeletal quads. Based on the GMM parameters, the skeletal quads of any action example are encoded into a Fisher vector, thus building the action descriptor which is led to a multi-class linear SVM.

In this paper, we propose a compact yet effective local skeleton descriptor that makes the pose representation invariant to any similarity transformation, hence view-invariant. The novel skeletal feature, referred here to as *skeletal quad*, locally encodes the relation of joint quadruples, so that 3D similarity invariance is guaranteed. Unlike the common BoW paradigm, we adopt a Fisher kernel representation [5]. Inspired by [13], we consider a Gaussian mixture model that generates the skeletal quads, while we enable a power normalisation for the Fisher vectors. Further, a multi-level splitting of sequences into segments is invoked to integrate the performing order of sub-actions into the vector representation. Such vectors constitute the input of a multi-class linear SVM. Fig. 1 illustrates our action recognition pipeline.

The remainder of the paper is organised as follows. We summarise the related work in Sec. II and we propose our video representation in Sec. III. Sec. IV in short discusses the used classifier, while our method is tested on public datasets in Sec. V. Finally, Sec. VI concludes this work.

Support from the European Research Council (ERC) through the Advanced Grant VHIA (#340113) is greatly acknowledged.

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.772

II. Related work

Shotton *et al.* presented in [15] an articulated pose recognition algorithm that makes the extraction of skeleton structure from single depth images possible. This result inspired many researchers either to rely on skeleton information only or to combine joints with other depth and/or color cues in order to recognise actions. In what follows, we first describe methods that only use skeleton data, hence more related to our approach, and we proceed with the state-of-the-art methods that use multiple features. For a recent detailed survey on human motion analysis from depth data, we refer the reader to [1], [22].

Xia *et al.* [20] suggest a compact posture representation through a histogram of 3D joint locations. Linear discriminant analysis along with a BoW model translates each action into a series of symbols (quantized postures). Then, a generative classifier (discrete HMM), that deals with the temporal nature of data, classifies each input. Yang and Tian [21] rely on position differences of in-frame and cross-frame joints. The resulting long vectors are compressed by PCA so that each frame is described by an EigenJoint descriptor. As for the classification, a naive Bayes nearest-neighbor classifier assigns to unknown inputs the label with the minimum video-to-class distance. Ofli *et al.* [10] represent an action instance as a sequence of the most informative joints (SMIJ) per action. This selection is based on joint related measures such as the moments of the joint angles. Several encoding methods are suggested for the vector representation of SMIJ, while two classifiers are tested, nearest-neighbor classifier (NNC) and SMVs. Chaundry *et al.* [3] are inspired by a neural encoding method and describe skeletal features based on a medial-axis template. Global linear dynamical systems (LDS) model the temporal dynamics of such features, while the LDS parameters describe the sequences and define the input of the classifier (either SVMs or NNC).

Regardless of the skeleton structure, the raw depth map itself provides a source for extracting discriminative features. Li *et al.* [8] propose a BoW model to describe the points close to the silhouette, after their projection to the three Cartesian planes. Local occupancy patterns (LOP) of depth sequences have been also used as features. Random and spatio-temporal LOPs are proposed in [17] and [16] respectively. Wang *et al.* in [18] combine LOPs around joints with joint differences to build joint-based time series, whose Fourier coefficients are used to describe the action. Moreover, a mining step provides a pool of informative actionlets (subset of joints) per action, that are taken into account by the classifier being used. Instead of the raw depth data, the 4D normals are used by Oreifej *et al.* [12], while their distribution is encoded in a histogram whose bins are irregularly spaced in 4D; this binning results from a learning process. Xia and Aggarwal [19] recently presented a depth-based spatio-termporal detector along with a self-similarity feature that describes local depth areas of adaptive size. Finally, Ohn-Bar and Trivedi [11] track the joint angles and build a descriptor based on similarities between angle trajectories. This feature is further combined with a double-HOG descriptor that accounts for the spatio-temporal distribution of depth values around the joints. Both [19] and [11] illustrate the benefits of combining different features.

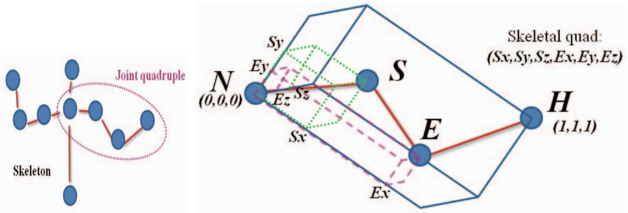

Fig. 2. An example of coding the marked joint quadruple {Neck, Shoulder, Elbow, Hand}. The Neck and Hand joints are identified with the points $(0, 0, 0)$ and $(1, 1, 1)$ in the new local coordinate system. The local 3D coordinates of the Shoulder and Elbow joints describe the structure of the quadruple and the quad descriptor is $\mathbf{q} = [S_x, S_y, S_z, E_x, E_y, E_z]^T$.

III. Skeleton-based video representation

We propose new representation based on skeleton joints, namely a local skeletal descriptor and an associated representation that. This skeletal descriptor, which is referred to as *skeletal quad*, was inspired by a geometry hashing method that describes the positions of nearby stars in night sky images [6]. It was also used to describe keypoint constellations in video frames [4]. Our action descriptor is a Fisher kernel representation which encodes the Jacobian of the probability function that generates the skeletal quads contained in a depth-image sequence.

A. Skeletal quads

Let $\mathbf{x} \in \mathbb{R}^3$ denote the coordinates of a skeleton joint in some world centered frame. Suppose also that we are given a quadruple of (nearby) joints $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ \mathbf{x}_3\ \mathbf{x}_4]$, where $(\mathbf{x}_1, \mathbf{x}_2)$ is the most widely separated pair of points within the quadruple. We consider a local coordinate system, such that $\mathbf{x}_1$ becomes the origin and $\mathbf{x}_2$ is mapped onto $[1, 1, 1]^\top$. This constrains a similarity transformation (a 3×3 rotation matrix, a translation vector and a scale factor) whose parameters can be easily computed from $\mathbf{x}_1$ and $\mathbf{x}_2$. Once these parameters are estimated, the quadruple is mapped onto its new coordinates:

$$\mathcal{S}(\mathbf{x}_i) = s\mathbf{R}[\mathbf{x}_i - \mathbf{x}_1], \quad i = 1 \dots 4, \tag{1}$$

with $\mathcal{S}(\mathbf{x}_1) = [0, 0, 0]^\top$ and $\mathcal{S}(\mathbf{x}_2) = [1, 1, 1]^\top$. Hence the quadruple is encoded by six parameters, namely $\mathbf{q} = [\mathcal{S}(\mathbf{x}_3); \mathcal{S}(\mathbf{x}_4)]$, where the notation $[\cdot; \cdot]$ denotes vertical vector concatenation. We refer to this descriptor as skeletal quad. Fig. 2 shows an upper-body skeleton example and how the skeletal quad is formed for a joint quadruple.

Unlike translation-invariant representations of skeleton joints that rely on joint differences (e.g. [18]), our descriptor is scale, viewpoint and body-orientation invariant. Moreover, this coding scheme leads to well distributed points in $\mathbb{R}^6$ [6]. Note that there is a kind of symmetry between descriptors owing to the different order in the pairs $\mathbf{x}_1, \mathbf{x}_2$, and $\mathbf{x}_3, \mathbf{x}_4$, which can be easily broken or taken into account. Here, we consider both orders for the first pair, while the third point of $\mathbf{X}$ is always the closest to the local origin between the remaining points.

B. A Fisher Kernel representation

The superiority of Fisher vectors (FV) against the popular BoW representation has been analyzed in the image classification context [13]. We follow a similar approach in order to describe an action sequence. It is important to note that the low dimension of the proposed descriptor compensates for the large inherent dimensionality associated with Fisher vectors.

Let $Q = \{\mathbf{q}_i, 1 \leq i \leq M\}$ be a set of M skeletal quads in an action example. By assuming statistical independence, Q may be modeled by a K-component Gaussian mixture model (GMM):

$$p(Q|\theta) = \prod_{i=1}^{M} \sum_{k=1}^{K} w_k \mathcal{N}(\mathbf{q}_i|\boldsymbol{\mu}_k, \boldsymbol{\sigma}_k), \tag{2}$$

where $\theta = \{w_k, \boldsymbol{\mu}_k, \boldsymbol{\sigma}_k\}$, $k = 1, ..., K$ is the set of the mixture parameters with mixing parameters w_k, means $\boldsymbol{\mu}_k \in \mathbb{R}^6$ and diagonal covariance matrices $\boldsymbol{\sigma}_k \in \mathbb{R}^{6\times 6}$. These parameters can be easily estimated via the standard EM algorithm based on a training dataset. Once the GMM parameters are estimated, any set Q may be described by its Fisher score [5], namely the Jacobian of the log-probability with respect to the GMM parameters:

$$J_\theta^Q = \nabla_\theta \log p(Q|\theta) \ . \tag{3}$$

The Fisher kernel, however, relates any two such vectors through a bilinear form based on the inverse of the Fisher information matrix. Since the decomposition of this matrix is possible, one can write the Fisher kernel as a linear kernel of the so called Fisher vectors, denoted here as $\mathcal{J}$. The reader is referred to [5] for a detailed analysis.

As in [13], we consider the Jacobians with respect to non-scalar parameters only, so that the FV consists of the concatenation of two vectors $\mathcal{J}_{\boldsymbol{\mu}_k}^Q$ and $\mathcal{J}_{\boldsymbol{\sigma}_k}^Q$. One can easily show (see [14]) that the $((k-1)6+j)$-th element of the above vectors ($1 \leq j \leq 6$, $1 \leq k \leq K$), i.e. the j-th entry for the k-th mixture component, is given by:

$$\begin{aligned} \mathcal{J}_{\boldsymbol{\mu}_k}^Q(j) &= \frac{1}{M\sqrt{\pi_k}} \sum_{i=1}^{M} \gamma_{k,i} \frac{q_i^j - \mu_k^j}{\sigma_k^j}, \\ \mathcal{J}_{\boldsymbol{\sigma}_k}^Q(j) &= \frac{1}{M\sqrt{2\pi_k}} \sum_{i=1}^{M} \gamma_{k,i} \left(\left(\frac{q_i^j - \mu_k^j}{\sigma_k^j} \right)^2 - 1 \right) , \end{aligned} \tag{4}$$

where $\gamma_{k,i}$ is the posterior probability that $\mathbf{q}_i$ belongs to kth cluster conditioned by Q. The normalization by M is added to avoid dependence on the Q's cardinality. Since quads live in $\mathbb{R}^6$, our Fisher vectors are reasonably long, i.e., of dimension $12K$. Typically, d-dimensional descriptors imply $2Kd$-dimensional Fisher vectors in a similar framework, a strong disadvantage when long descriptors are employed.

Once the two vectors of (4) are computed and appended in a single vector, a power-normalisation step is applied, i.e., each element undergoes the transformation [13]

$$T(x) = sgn(x) * |x|^\alpha \ . \tag{5}$$

This normalisation step eliminates the sparseness of the Fisher vector, thus increasing its discriminability. Note that the FVs tend to be sparse since the majority of the quads are assigned with high posterior probability to a couple of components. The impact of such a normalization is evident in Fig. 3 which depicts the distribution of a Fisher vector's elements before and after the power normalisation. We refer the reader to [13], [14] for a detailed discussion about power normalization.

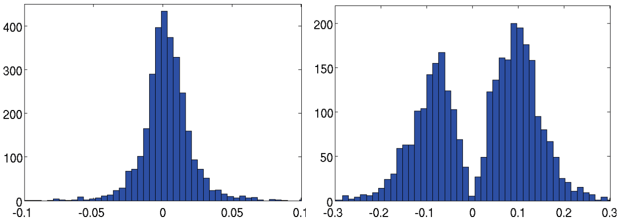

Fig. 3. The distribution of entries (left) before and (right) after the power normalisation ($\alpha = 0.5$) of Fisher vector.

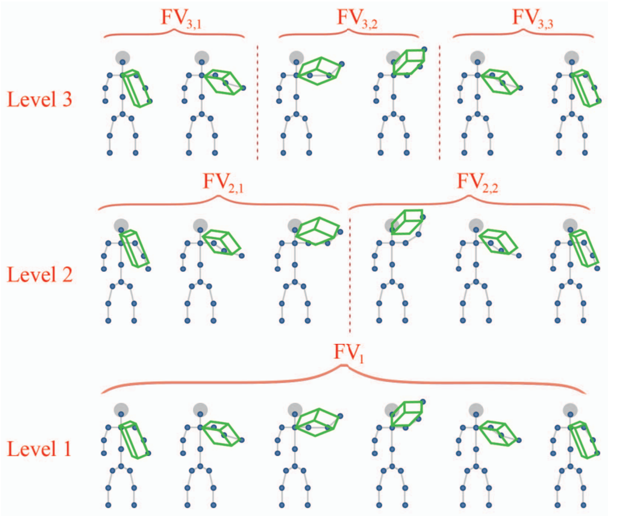

Fig. 4. A 3-level representation of an action example. The complete action descriptor is the concatenation of the six FVs.

Since any permutation of Q's elements would lead to the same FV, the order of different postures within an action is not taken into account. Therefore, and similar to spatial pyramids in scene recognition [7], we further enable a multi-level splitting of sequences, so that n video non-overlapping segments are present in the n-th level. The concatenation of FVs of all segments somewhat makes the representation temporally- and order-dependent as long as the order of sub-actions is encoded in the action descriptor. Three pyramid levels are used in this paper as shown in Fig. 4.

IV. Action Classification

It is beyond the scope of this paper to focus on the classification step. We simply employ linear SVMs trained in an one-versus-all fashion in order to build a multi-class classifier, while a cross-validation on training sets provides the best offset per classifier. Notice that a Fisher kernel classifier is equivalent to a linear classifier on FVs [5]. Moreover, linear SVMs easily deal with the high-dimensional representations that result from our mutli-level FVs. However, a more in-depth analysis on the classification step will possibly lead to higher performance. To implement the classifier, we make use of the LIBSVM library [2].

V. Experiments

In this section, we test our action recognition method on widely used datasets and compare it with the state-of-the-art.

TABLE II. RECOGNITION ACCURACY ON MSRACTION3D DATASET USING SKELETON JOINTS.

	AS1	AS2	AS3	Overall
Histogram of 3D Joints (Xia *et al.* [20])	87.98%	85.48%	63.46%	78.97%
EigenJoints (Yang & Tian [21])	74.50%	76.10%	96.40%	82.33%
Joint Angles (Ohn-Bar & Trivedi [11])	N/A	N/A	N/A	83.53%
Joint angles + SIMJ* (Ofli *et al.* [10])	N/A	N/A	N/A	47.06%
Joint angles + LDS* (Chaundry *et al.* [3])	N/A	N/A	N/A	83.89%
FV of Skeletal Quads	88.39%	86.61%	94.59%	**89.86**%

**results on a reduced dataset with less actions*

Two publicly available datasets are used: MSR-Action3D [8] and HDM05 [9]. The datasets are captured from different modalities and regard different human activities.

A. MSR-Action3D Dataset

MSR-Action3D dataset [8] is a set of depth videos, captured by a Kinect device, that contains 20 actions: *high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, sideboxing, bend, forward kick, side kick, jogging, tennis swing, tennis serve, golf swing, pick up & throw*. Ten actors perform each action three or two times. Commonly, this dataset is divided into three action subsets as shown in Table I, while the average performance over these subsets is reported as the recognition accuracy [8], [20]. Notice that AS1 and AS2 were intended to group similar actions, while AS3 defines a subset with high cross-action variability.

We consider here the challenging case of cross-subject splitting into training and testing sets [20], [18], [19], [8], i.e. half of the subjects are used for training and the rest for testing. In particular, the common splitting with odd and even subject-IDs defining the training and testing sets respectively is adopted [8], [18]. As in [18], we exclude ten sequences where the skeleton data is missing or corrupted, namely, 557 sequences are used in total. A 20-joint skeleton is provided that implies 4845 skeletal quads per frame. The number of GMM components is $K = 128$, thus leading to 1560-element FVs that are power normalised with $a = 0.3$. As mentioned, three levels are considered, hence $(1 + 2 + 3 =)6$ FVs are obtained per sequence.

A fair comparison suggests using competitors that exploit the skeleton structure only. For the sake of completeness, however, we present in a separate table the recognition accuracy of methods that use multiple features and combine joints with other depth cues. Table II shows the recognition accuracy per action subset along with the corresponding results of methods that rely on skeleton joints. Notice that SMIJ [10] and LDS [3] methods use a reduced dataset of 17 actions. Unlike most of the competitors, our method provides equally good results in all action subsets, while it provides the best overall accuracy. Fig. 5 depicts the confusion matrices we obtain per action subset. Actions with similar poses, such as the drawing actions in AS2, are more easily confused owing the similarity of the resulting quads. Instead, actions in AS3 are better discriminated, as expected. For comparison reasons, we also show the confusion matrices obtained by the EigenJoint-based method [21].

TABLE I. THE THREE ACTION SUBSETS (AS) OF MSRACTION3D DATASET AS DEFINED IN [8]

AS1	AS2	AS2
Horizontal arm wave	High arm wave	High throw
Hammer	Hand catch	Forward kick
Forward punch	Draw x	Side kick
High throw	Draw tick	Jogging
Hand clap	Draw circle	Tennis swing
Bend	Two hand wave	Tennis serve
Tennis serve	Forward kick	Golf swing
Pickup & throw	side boxing	Pickup & throw

TABLE III. RECOGNITION ACCURACY ON MSRACTION3D DATASETS. THE METHODS EITHER COMBINE SKELETON JOINTS WITH OTHER DEPTH CUES OR USE VARIOUS DEPTH FEATURES

Methods	Accuracy
Bag of 3D points (Li *et al.* [8])	74.70%
Radnom Occupancy Patterns (Wang *et al.* [17])	86.50%
Space-time Occupancy Patterns (Vieira *et al.* [16])	87.50%
Joints + Actionlets (Wang *et al.* [18])	88.20%
HON4D (Oreifej and Liu [12])	88.89%*
Joints + Depth Cuboids (Xia and Aggarwal [19])	89.30%
Joint Angles + MaxMin + HOG^2 (Ohn-Bar and Trivedi [11])	94.84%*

*performance obtained with different cross-subject splitting (first five actors for training, last five actors for testing).

Table III shows the recognition accuracy of methods that use the full-body depth map or combine skeleton information with other features. To the best of our knowledge, the method by Ohn-Bar and Trivedi [11] performs best when it combines the joints angles with two other features. However, when only joint information is employed, our method performs better (see Table II). It is important to note that our algorithm competes with the majority of these methods, despite the fact that they employ multiple features. As a consequence, the use of skeletal quads in conjunction with other features suggests an even promising approach.

B. Mocap databse HDM05

In this subsection, we present results when applying our method on skeleton sequences of a Motion Capture dataset, the HDM05 database [9]. Unlike MSRAction3D, this dataset is captured by motion-capture sensors that acquire more precise data. Moreover, the frame rate is much higher (120 fps), while 31 joints are provided per pose instance. However, we consider here only 15 joints: *root, L/Rhip, L/Rknee, L/Rankle, neck, head, L/Rshoulder, L/Relbow, L/Rwrist*. As a result, each skeleton pose implies 1365 quads.

We adopt the experimental setup of [10] that suggests a set of 11 actions: *deposit floor, elbow to knee, grab high, hop both legs, jog, kick forward, lie down floor, rotate both arms*

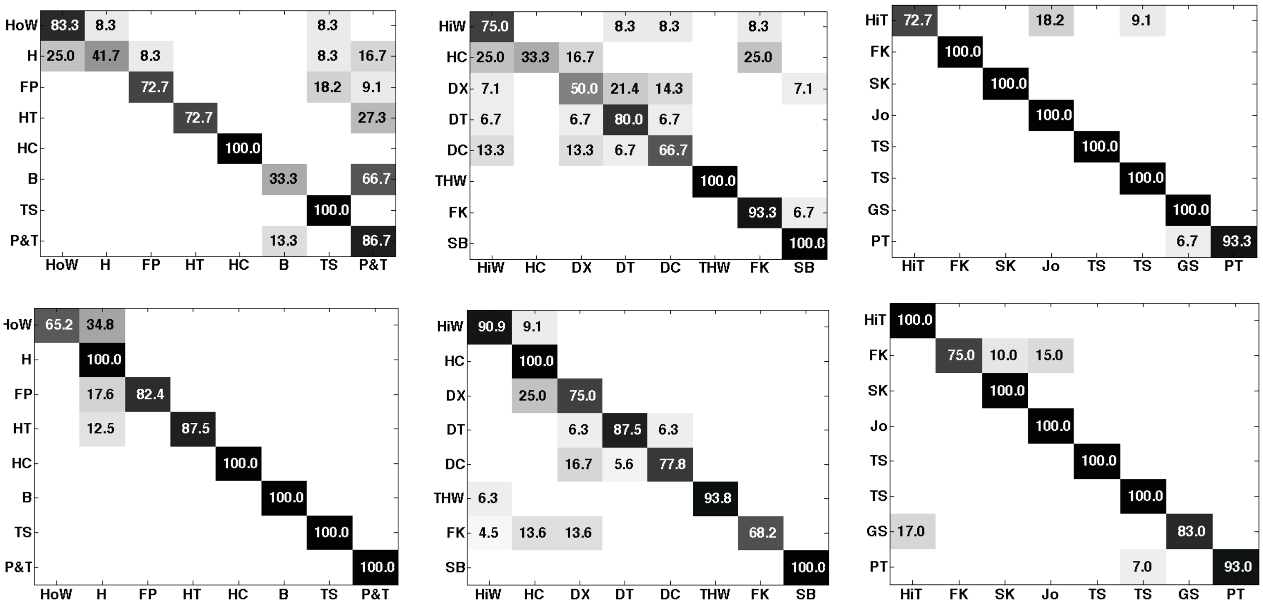

Fig. 5. The confusion matrices obtained by (top) [21] and (bottom) our method for each action subset of MSRAction3D dataset (left: AS1, middle: AS2, right: AS3).

backward, sneak,squat, and throw basketball. The actions are performed by 5 subjects, while each subject performs each action a couple of times (not fixed); this suggests a set of 249 sequences. As with [10], we use a cross-subject splitting with 3 and 2 subjects in training and testing sets respectively, thus having 139 training and 110 testing examples at our disposal. The same parameters with MSR-Action3D are used.

Table IV shows the recognition accuracy of our method along with several counterparts of the algorithms presented in [10] and [3]. Notice that all methods in [3] regard global linear dynamical systems (LDS) that describe the dynamics of several joint features across the whole sequence. Then, the LDS parameters encode each action sequence and are classified by SVMs. Moreover, both methods enable a mining step that provides the most discriminative features. Even so, our simple method is more robust and provides higher recognition accuracy. The confusion matrix of our results is shown in Fig. 6.[1] Despite the more accurate joints in this dataset, there is still confusion between some actions, such as *deposit floor* and *kick forward*. Note that *deposit floor* means “deposit an item on the floor with knees bent” [9]. As a consequence, there are similar leg poses between these two actions, hence similar quads.

TABLE IV. RECOGNITION ACCURACY ON HDM05 DATASET USING SKELETON JOINTS

Methods	Accuracy
Joint angles + LDS (Ofli *et al.* [10])	76.15%
Joint angles + HMW* (Ofli *et al.* [10])	78.90%
Joint angles + HMIJ** (Ofli *et al.* [10])	82.57%
Joint angles + SMIJ (Ofli *et al.* [10])	84.47%
Joint Shape + LDS (Chaudhry *et al.* [3])	82.57%
Joint Tangents + LDS (Chaudhry *et al.* [3])	88.07%
Joint positions + LDS (Chaudhry *et al.* [3])	91.74%
FV of Skeletal Quads	**93.89**%

**histogram of motion words*, ***histogram of most informative joints*

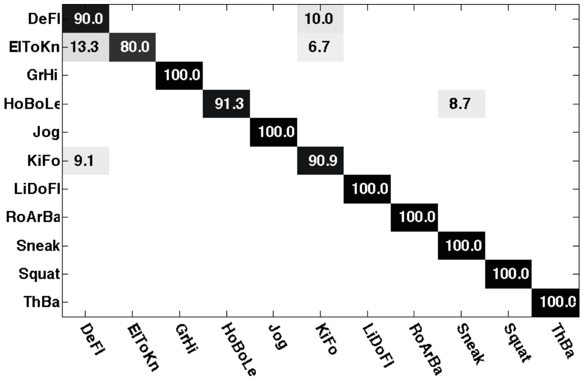

Fig. 6. The confusion matrix of our method on HDM05 dataset.

VI. CONCLUSIONS

A local skeletal representation was proposed in this paper. This representation implies a short, view-invariant descriptor of joint quadruples. Furthermore, a Fisher kernel representation was devised that encodes the generation of such a representation from a Gaussian mixture model. The final action descriptor results from a multi-level representation of Fisher vectors that encodes the temporal nature of action examples. Experimental validation of the proposed method verified its state-of-the-art performance in human action recognition from depth data.

Future work regards the combination of skeletal quads with other cues. As well, the dimensionality reduction of the final

[1] confusion matrices are not provided by [10] and [3]

action descriptor will lead to more efficient classification and will allow the use of non-linear SVMs.

References

[1] J. Aggarwal and L. Xia. Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 2014.

[2] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[3] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal. Bio-inspired dynamic 3d discriminative skeletal features for human action recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2013.

[4] G. Evangelidis and C. Bauckhage. Efficient subframe video alignment using short descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35:2371–2386, 2013.

[5] T. Jaakola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1999.

[6] D. Lang, D. W. Hogg, K. Mierle, M. Blanton, and S. Roweis. Astrometry.net: Blind astrometric calibration of arbitrary astronomical images. *The astronomical journal*, 137:1782–2800, 2010.

[7] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[8] 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.

[9] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation of mocap database hdm05. Technical Report CG-2007-2, University of Bonn, June 2007.

[10] 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. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2013.

[11] E. Ohn-Bar and M. M. Trivedi. Joint angles similiarities and hog^2 for action recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2013.

[12] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *CVPR*, 2013.

[13] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[14] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International Journal of Computer Vision*, 105(3):222–245, 2013.

[15] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[16] A. W. Vieira, E. R. Nascimento, G. L. Oliveira, Z. Liu, and M. F. Campos. On the improvement of human action recognition from depth map sequences using spacetime occupancy patterns. *Pattern Recognition Letters*, 36:221–227, 2014.

[17] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu. Robust 3d action recognition with random occupancy patterns. In *ECCV*, 2012.

[18] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, 2012.

[19] L. Xia and J. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *CVPR*, 2013.

[20] L. Xia, C.-C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012.

[21] X. Yang and Y. Tian. Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012.

[22] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall. A survey on Human Motion Analysis from Depth Data. *Time-of-Flight and Depth Imaging. Sensors, Algorithm and Applications*, 8200:149–187, 2013.