A Vector of Locally Aggregated Descriptors Framework for Action Recognition on Motion Capture Data

Ioannis Kapsouras, Nikos Nikolaidis
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
Email: {jkapsouras,nikolaid}@aiia.csd.auth.gr

***Abstract*—In this paper we introduce an approach for action recognition in motion capture data. The data are represented by the joints positions of the skeleton in each frame (posture vectors) and the differences of these positions over time, in different temporal scales. The Vector of Locally Aggregated Descriptors (VLAD) framework is used to encode the extracted features whereas a Support Vector Machine (SVM) is used for classification. A voting scheme is used in the VLAD framework to achieve soft encoding. The effectiveness and robustness of the proposed approach is shown in experiments performed on three datasets (MSRAction3D, MSRActionPairs and HDM05).**

I. Introduction

Motion capture (skeleton animation) data have been lately used quite often in computer vision and video analysis and understanding research especially since the release of the Microsoft Kinect RGBD device in 2010 [1]. Kinect is able to record depth video data and can also provide, through algorithms included in the software that accompanies the sensor, information for the joints positions of tracked skeletons, transforming motion capture (mocap), a rather expensive technique until then, to a common and affordable operation. Markerless motion capture is a big advantage, besides the pricing, of RGBD devices such as the Kinect sensor. There is no need for sensors to be attached to a person's body in order to capture skeleton motion, thus the creation of datasets and their corresponding analysis is a much easier task with Kinect-like devices.

An important research topic related to motion capture data acquired from depth cameras or through other means is action recognition. Action recognition deals with the process of labeling a motion sequence with respect to the depicted motions. Technically, an action is a sequence generated by a human subject during the performance of a task. Action recognition has numerous applications including human computer interaction, video surveillance, multimedia annotation etc.

This paper presents an action recognition method that operates on features derived from skeletal data. Posture vectors containing the 3D position of skeletal joints and their forward differences in different temporal scales are extracted from motion capture data as in [2]. Vector of Locally Aggregated Descriptors (VLAD) [3] is used as a framework for encoding the features of each sequence. The proposed approach was tested in three action datasets. A flowchart of the approach is shown in Fig. 1.

The remaining of this paper is organized as follows. In Section II, we present a review of previous work on this topic. In Section III, the proposed method is described in detail. Experimental performance evaluation of the proposed method and comparison with other approaches is presented in Section IV. Conclusions follow in Section V.

II. Previous Work

Action recognition from video data has been a very active research field the last decades. Surveys and reviews of action recognition methods on such data can be found in [4], [5] and [6]. A review of public datasets used for the experimental evaluation of such methods can be found in [7]. However, motion capture technology became widely available only during the last years. Hence the body of research for movement recognition on mocap data is not as extensive as for video data. A review of spacetime representations of 3D skeletal data for movement recognition or related tasks is presented in [8]. The use of the most informative joints in order to represent skeletal sequences for action recognition was proposed by Ofli et al. in [9]. A sequence is segmented either by using a fixed number of segments or by using a fixed temporal window. Then the proposed features (the most informative joints) are computed in these segments and used to represent the sequence. Nearest neighbour and SVM are used for classification. Han et al. used a hierarchical discriminative approach in [10] for human action recognition. The human motion is represented in a hierarchical manifold space learned by the use of Hierarchical Gaussian Process Latent Variable Model (HGPLVM). Conditional random fields are used to extract mutual invariant features from each manifold subscpace, and the classification is performed by an SVM classifier. Amor et al. in [11] represent the skeletons as trajectories on Kendall's shape manifold. In order to make these representations suitable for statistical analysis, they use a combination of the transported square-root invariant vector fields (TSRVFs) of trajectories and the standard Euclidean norm. The authors used these representations for smoothing and denoising skeleton trajectories using median filtering, up and down sampling in time domain, simultaneous temporal

ISBN 978-90-827970-1-5 © EURASIP 2018

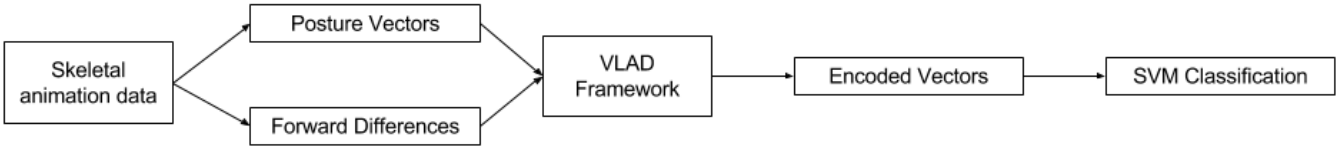

Fig. 1. A flowchart of the proposed approach.

registration of multiple actions and for extracting invertible Euclidean representations of actions. The latter were used to address the action recognition task with SVM classification. A new set of features (local occupancy patterns) and a new temporal patterns representation (Fourier Temporal Pyramid) was proposed in [12] in order to represent 3D joint positions. They define the so-called actionlets, each being a certain conjunction of the features for a joints subset. A sequence is represented as a linear combination of actionlets. SVM is used for classification. The covariance matrix for skeleton joints locations over time is used as a descriptor by Hussein et al. in [13] to address the action recognition problem. The authors compute the covariance matrix over time. To use the temporal dependency of joint locations, multiple covariance matrices are computed in a hierarchical fashion. Linear SVM is used for classification. Gowayyed et al. in [14] proposed a new descriptor to represent the 3D trajectories of body joints and perform action recognition. The descriptor is a histogram of oriented displacements in 2D space. Each displacement in the trajectory votes with its length in a histogram of orientation angles. The authors compute the descriptor for each joint in xy, xz and yz projections and then concatenate the histograms. In order to take into account the temporal information, they use the temporal pyramid approach to construct the final vector that represents the human action. Vemulapalli and Chellappa in [15] use 3D rotations between various body parts to represent each skeleton. In more detail, to obtain a scale-invariant representation, the authors use only the rotations to describe the relative 3D geometry between parts. The authors used a representation similar to that in [16] to model the human actions as curves in a Lie group. In order to classify the modelled actions, the authors unwrap the action curves onto the Lie algebra by combining the logarithm map with rolling maps (that describe how a manifold rolls over another, without slip and twist, along a smooth rolling curve). The mapped curves are classified using SVM.

III. Method Description

A. Feature extraction

In the proposed approach, skeletal data are represented by two types of features: the posture vectors and the forward differences vectors in a way similar to the approach used in [2]. However, the method proposed in [2] was unable to distinguish between similar motions that have different directions (e.g. stand up and sit down), because of the way the forward differences were calculated. The forward differences are computed in this paper in a slightly different manner (Eq. 2) so as to encode directional information of the human motion.

Skeletal data can be represented as a sequence of posture vectors $\mathbf{q}_i,\ i = 1, \ldots, N$ where N is the number of frames of the sequence. Each such vector carries information for the positions of the skeleton joints in the $3D$ space.

$$\mathbf{q}_i = [x_{i1}, y_{i1}, z_{i1}, x_{i2}, y_{i2}, z_{i2}, \ldots, x_{il}, y_{il}, z_{il}] \tag{1}$$

where l is the number of joints that form the posture vector. Motion capture sequences are also represented by vectors of forward differences evaluated over joint positions. Forward differences estimate the first derivative of a signal and thus, when applied on joint positions, carry information for the average velocities of the skeleton joints. More specifically, forward differences in terms of skeletal animation data can be defined as:

$$\boldsymbol{v}_i^t = \Delta_t[\mathbf{q}] = \mathbf{q}_{i+t} - \mathbf{q}_i \tag{2}$$

where $\mathbf{q}_i, \mathbf{q}_{i+t}$ are the posture vectors in frames i and $i + t$ respectively, $\boldsymbol{v}_i^t$ can be considered as a vector of the average velocities of the joints of a skeleton in frame i. In the proposed approach, the forward differences of the joints are computed in different temporal scales and more specifically for $t = 1$, $t = 5$ and $t = 10$ in order to capture the dynamics of the joints of a skeleton.

Summarizing, two types of features, forming 4 groups of vectors are used to represent a skeletal sequence: posture vectors and forward differences vectors in three different temporal scales. Thus each sequence is represented by four sets of feature vectors: $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3, \mathbf{T}_4$:

$$\begin{aligned} \mathbf{T}_1 &= \{\mathbf{q}_1, \ldots, \mathbf{q}_N\} \\ \mathbf{T}_2 &= \{\boldsymbol{v}_1^1, \ldots, \boldsymbol{v}_{N-1}^1\} \\ \mathbf{T}_3 &= \{\boldsymbol{v}_1^5, \ldots, \boldsymbol{v}_{N-5}^5\} \\ \mathbf{T}_4 &= \{\boldsymbol{v}_1^{10}, \ldots, \boldsymbol{v}_{N-10}^{10}\} \end{aligned} \tag{3}$$

ISBN 978-90-827970-1-5 © EURASIP 2018

B. Feature Vector encoding - Vector of Locally Aggregated Descriptors Framework

The features are encoded using the Vectors of Locally Aggregated Descriptors (VLAD) framework [3].

First, feature vectors are clustered in each feature space by using the K-means algorithm. In VLAD encoding the differences of the feature vectors from cluster centers are used to form vectors that represent the sequences.

In more detail, let $\mathbf{T}_{kj}$ where $k = 1 \dots 4$ be the sets of features that have been extracted from sequence j, in the 4 different feature spaces. K-means was applied separately for each feature type, resulting to $4 * C$ centroids, where C the number of clusters in each feature space. Next, each feature vector is mapped to the cluster centers. A voting scheme is used to encode the skeletal features. In more detail, the similarities of each feature vector (belonging to one of the four feature spaces) of the j-th sequence with each cluster center of this feature space are computed.

$$s_k^j = sim(\mathbf{c}_k, \mathbf{t}^j) = exp(-(\frac{\sum_{i=1}^{l}(\parallel c_{ki} - t_i^j \parallel_2)}{0.5 * \max_k(\sum_{i=1}^{l}(\parallel c_{ki} - t_i^j \parallel_2))})^2) \quad (4)$$

where s_k^j is the similarity between cluster center $\mathbf{c}_k$ and a feature vector $\mathbf{t}^j$ of the j-th sequence. Then a vector of ordered similarities: $S = [s_{(1)}^j, \dots, s_{(C)}^j]$, where C is the number of clusters is formed for each feature vector of sequence j. Let $\mathbf{b}_{ij}$ be the C-dimensional voting vector that encodes the association of the i-th feature vector of the j-th sequence with each cluster center. In the voting scheme, an element of $\mathbf{b}_{ij}$ is the similarity of the i-th feature with a cluster center if the corresponding cluster center is in the R most similar centers (as in [2]) of this feature vector or 0 otherwise, where R is is found as the value that satisfies the following inequalities:

$$\frac{\sum_{k=1}^{R-1} s_{(k)}^j}{\sum_{k=1}^{C} s_{(k)}^j} < 0.05 < \frac{\sum_{k=1}^{R} s_{(k)}^j}{\sum_{k=1}^{C} s_{(k)}^j} \quad (5)$$

In the special case where $s_{(1)}^j > 0.05 \sum_{k=1}^{C} s_{(k)}^j$, R is set to 1. In the next step, vectors $\mathbf{v}_z'$ are formed as follows:

$$\mathbf{v'}_z^j = \sum_{i=1}^{M} b_{ijz} \left(\mathbf{t}_i^j - \mathbf{c}_z \right), \; z = 1, \dots, C \quad (6)$$

where M is the number of features extracted from the j-th sequence, b_{ijz} is an element of the $\mathbf{b_{ij}}$ vector and represents the association of i feature with z cluster center, $\mathbf{t}_i^j$ the i-th feature vector in a certain feature space and $\mathbf{c}_z$ the z cluster center. Dimensionality of $\mathbf{v'}_z^j$ is the same as that of the feature vectors. Then, square root normalization is applied to each v'^j_{zq} element of $\mathbf{v'}_z^j$ to obtain $\mathbf{v''}_z^j$:

$$v''^j_{zq} = sgn(v'^j_{zq}) \sqrt{|v'^j_{zq}|}, \; q = 1, \dots, O \quad (7)$$

where O the dimensionality of the feature vector. Subsequently, $\mathbf{v''}_z^j = [v''^j_{z1}, v''^j_{z2}, \dots, v''^j_{zO}]$ is normalized using l^2 normalization, $\mathbf{v}_z^j = \mathbf{v''}_z^j / \|\mathbf{v''}_z^j\|_2$. The resulting $\mathbf{v}_z^j$ ($z = 1 \dots C$) vectors are concatenated to form a vector $\mathbf{V'}_j$ of $L = O \times C$ dimensionality that characterizes the j-th sequence:

$$\mathbf{V'}_j = \begin{bmatrix} \mathbf{v}_1^j \\ \mathbf{v}_2^j \\ \vdots \\ \mathbf{v}_C^j \end{bmatrix} \quad (8)$$

The final vector $\mathbf{V'}_j$ is also l^2 normalized to obtain $\mathbf{V}_j$:

$$\mathbf{V}_j = \frac{\mathbf{V'}_j}{\|\mathbf{V'}_j\|_2} \quad (9)$$

This procedure is repeated for each feature type, and finally, each sequence is represented by 4 vectors, namely $\mathbf{V}_j^p$ for posture vectors, $\mathbf{V}_j^{v^1}$ for forward differences when $t = 1$, $\mathbf{V}_j^{v^5}$ for forward differences when $t = 5$ and $\mathbf{V}_j^{v^{10}}$ for forward differences when $t = 10$.

C. Classification

An SVM with RBF kernels is used for classification. Since 4 vectors have been formed to represent each sequence, 4 kernels are computed, one for each feature type. The kernels are fused by computing the mean kernel:

$$\mathbf{K}_f = (\mathbf{K}_{pos} + \mathbf{K}_{v^1} + \mathbf{K}_{v^5} + \mathbf{K}_{v^{10}})/4 \quad (10)$$

where $\mathbf{K}_{pos}, \mathbf{K}_{v^1}, \mathbf{K}_{v^5}$ and $\mathbf{K}_{v^{10}}$ are the kernels formed from the posture vectors and the forward differences for $t = 1, t = 5$ and $t = 10$ respectively.

IV. Experimental Results

The proposed method has been tested on three datasets, namely *MSR Action3D (MSR)* [17], *MSR Action Pairs (MSR-Pairs)* [18], and *HDM05* [19]. SVM classifier was trained with values of the soft margin parameter in the range $2^{-20}, 2^{-19}, \dots, 2^{19}, 2^{20}$. The parameter value with the highest results was computed for each dataset and the best results are presented. It should also be noted that concatenation of the feature vectors and usage of a single kernel matrix was investigated for fusing the information from the 4 feature types but the achieved results were lower than those obtained by using the mean of 4 kernels, as described in III-C.

The MSR3D dataset consists of 10 subject performing 20 actions with 2 or 3 repetitions of each action. The actions performed in the MSR dataset are *high arm wave (HighArmW), horizontal arm wave (HorizArmW), hammer (Hammer), hand catch (HandCatch), forward punch (FPunch), high throw (HighThrow), draw x (DrawX), draw tick (DrawTick), draw circle (DrawCircle), hand clap (Clap), two hand wave (TwoHandW), side-boxing (Sidebox), Bend (Bend), forward kick (FKick), side kick (SKick), jogging (Jog), tennis swing (TSwing), Golf (Golf), pickup & throw (PickT)* and *tennis serve (TServe)*. Odd subjects (1,3,5,7) were used for training and even subjects (2,4,6,8) were used for testing. It should be noted that PCA was applied to the positions of the joints to decorrelate the data. The results of the proposed method alongside with four methods that use the same experimental

ISBN 978-90-827970-1-5 © EURASIP 2018

setup are shown in Table I. The proposed approach achieved very good results outperforming three methods and matching the results of the method proposed in [14].

TABLE I
CORRECT CLASSIFICATION RATE IN THE EXPERIMENTAL SETUP PROPOSED IN [17] ON THE MSR ACTION3D DATASET.

	Classification Rate
Proposed	**91.27**
Amor et al. [11]	89
Wang et al. [12]	88.2
Hussein et al. [13]	90.53
Gowayyed et al. [14]	91.26

The MSRPairs dataset was proposed in [18]. The main characteristic of this dataset is that is consists of pairs of actions. These action pairs have similar motion and shape cues but their correlations vary. In more detail, 6 pairs of actions exist in this dataset, namely *Pick up a box/Put down a box*, *Lift a box/Place a box*, *Push a chair/Pull a chair*, *Wear a hat/Take off a hat*, *Put on a backpack/Take of a backpack*, *Stick a poster/Remove a poster*. Ten subjects perform each action three times. Sequences of half of the subjects were used for training and the rest for testing. PCA was performed to the skeletal data as in MSR3D dataset. The classification rate achieved by the proposed method and by 2 other methods can be seen in Table II. The proposed method outpreformed both methods.

TABLE II
CORRECT CLASSIFICATION RATE IN THE EXPERIMENTAL SETUP PROPOSED IN [18] ON THE MSRPAIRS DATASET.

	Rate
Proposed	**95.51**
Vemulapalli and Chellappa [15]	94.09
Amor et al. [11]	93

The HDM05 database [20] consists of various movements performed by five subjects in the form of Amc files. The subset of this dataset that was used to assess the proposed method was proposed by Ofli et al. in [9] and includes 16 actions namely *deposit floor (DepositFloor), elbow to knee (ElbowKnee), grab high (GrabHigh), hop both legs (HopBoth), jog (Jog), kick forwards (KickFor), lie down floor (LieFloor), rotate both arms backward (RotateBArmsB), sneak (Sneak), squat (Squat), throw basketball (ThrowBasket), jump (Jump), jumping jacks (JumpJacks), throw (Throw), sit down (SitDown) and stand up (StandUp)*. The experimental setup proposed in [9] was used. In more detail the sequences of 3 subjects were used to form the training set (216 sequences) and the sequences from the other 2 subjects were used to form the test set (177 action sequences). The experimental setup proposed in [9] was used. The correct classification rate for the proposed method was 93.22%. This is 1.69% better than the result (91.53%) obtained on the same dataset by the method in [9].

K-means algorithm is used in VLAD framework for the evaluation of the codewords that will be used for the representation. An obvious question is how the number of clusters (C) affects the performance of the proposed framework. Classification rates for various values of C for the MSR3D dataset and for the first experimental setup can be seen in Figure 2. As can be seen in this figure, C has not strong impact to the classification rates achieved by the proposed method.

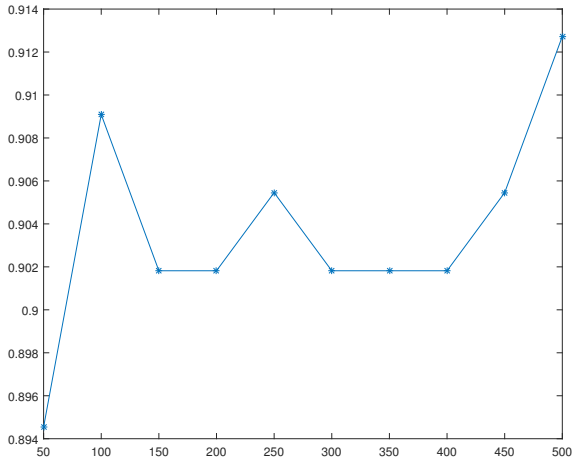

Fig. 2. Classification rates for various numbers of cluster centers (MSR3D dataset).

Another important characteristic of a classification method is the time needed for a sequence to be classified. The classification time for an unknown sequence of length 60 frames (2 seconds) can be seen in Table III. The framework that was used for classification was trained with 200 cluster centers (the parameters that achieved the best classification rate were used). The experiment ran on a PC with a quad-core processor and 8 GB of RAM and the computations were made using unoptimized MatLab code under Windows.

TABLE III
COMPUTATIONAL TIME (IN SECONDS) OF THE PROPOSED METHOD.

Method Component	Time (sec)
Feature Extraction	0.014
Feature Encoding	0.144
Classification	1.927
Overall	2.084

As can be seen in this Table, a sequence of 2 seconds duration was classified in 2.08 seconds, hence the proposed method can be used in real time classification scenarios, considering a time window for continuous classification.

A critical parameter for the computational complexity of the proposed method is the number of clusters C (Figure 3). C affects the classification and feature encoding steps, but, obviously, not the feature extraction step. The classification step execution time for various values of C is shown in Figure 3 since, as can be seen in Table III, it is more time consuming than feature encoding. As can be seen in this Figure, the time needed for classification for the proposed method is almost linear to the number of clusters. However, according to Figure 2, good classification results can be achieved even with a small C. Hence, with a small sacrifice in classification rate,

ISBN 978-90-827970-1-5 © EURASIP 2018

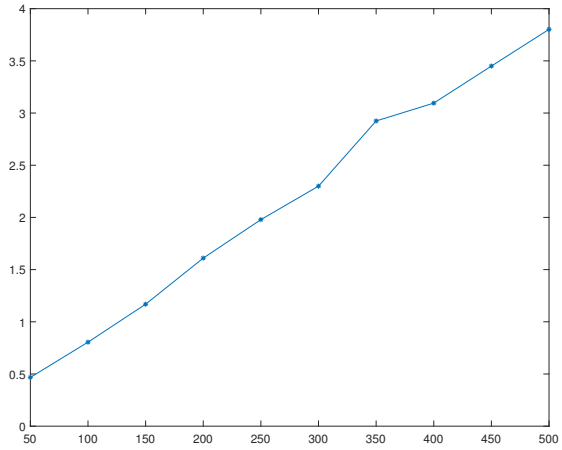

Fig. 3. Computational time (in seconds) for the classification step of the proposed method for different numbers of clusters.

the overall classification time can be kept fairly low.

V. Conclusion

In this paper, an approach for action recognition was proposed. Four types of features are extracted and VLAD framework is used to encode these features. A voting scheme is used for the encoding of the features. SVM is used for classification and the features are fused using kernel addition. Experiments showed that that the used features alongside with the VLAD framework achieve high classification results in three different datasets. In the future, extension towards motion clustering, segmentation and indexing will also be considered.

References

[1] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE MultiMedia*, vol. 19, no. 2, pp. 4–10, Apr. 2012.

[2] I. Kapsouras and N. Nikolaidis, "Action recognition on motion capture data using a dynemes and forward differences representation," *Journal of Visual Communication and Image Representation*, vol. 25, no. 6, pp. 1432 – 1445, 2014.

[3] H. Jegou, M. Douze, C. Schmid, and P. Perez, "Aggregating local descriptors into a compact image representation," in *Proceedings of 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010, pp. 3304–3311.

[4] S. Vishwakarma and A. Agrawal, "A survey on activity recognition and behavior understanding in video surveillance," *The Visual Computer*, vol. 29, no. 10, pp. 983–1009, 2013.

[5] M. B. Holte, C. Tran, M. M. Trivedi, and T. B. Moeslund, "Human action recognition using multiple views: A comparative perspective on recent developments," in *Proceedings of the 2011 Joint ACM Workshop on Human Gesture and Behavior Understanding*, ser. J-HGBU '11. New York, NY, USA: ACM, 2011, pp. 47–52.

[6] J. Aggarwal and M. Ryoo, "Human activity analysis: A review," *ACM Comput. Surv.*, vol. 43, no. 3, pp. 16:1–16:43, Apr. 2011.

[7] J. M. Chaquet, E. J. Carmona, and A. Fernndez-Caballero, "A survey of video datasets for human action and activity recognition," *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 633 – 659, 2013.

[8] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3D skeletal data," *Computer Vision and Image Understanding*, vol. 158, no. C, pp. 85–105, May 2017.

[9] 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," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 24 – 38, 2014.

[10] L. Han, X. Wu, W. Liang, G. Hou, and Y. Jia, "Discriminative human action recognition in the learned hierarchical manifold space," *Image and Vision Computing*, vol. 28, no. 5, pp. 836–849, May 2010.

[11] B. B. Amor, J. Su, and A. Srivastava, "Action recognition using rate-invariant analysis of skeletal shape trajectories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 1–13, January 2016.

[12] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1290–1297.

[13] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations," in *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, ser. IJCAI '13. AAAI Press, 2013, pp. 2466–2472.

[14] M. A. Gowayyed, M. Torki, M. E. Hussein, and M. El-Saban, "Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition," in *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, ser. IJCAI '13. AAAI Press, 2013, pp. 1351–1357.

[15] R. Vemulapalli and R. Chellappa, "Rolling rotations for recognizing human actions from 3d skeletal data," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 4471–4479.

[16] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 588–595.

[17] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2010, pp. 9–14.

[18] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 716–723.

[19] M. Müller, A. Baak, and H. Seidel, "Efficient and robust annotation of motion capture data," in *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. New York, NY, USA: ACM, 2009, pp. 17–26.

[20] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, "Documentation mocap database HDM05," Universität Bonn, Tech. Rep. CG-2007-2, Jun. 2007.

ISBN 978-90-827970-1-5 © EURASIP 2018