Multimed Tools Appl (2019) 78:1971–1998
https://doi.org/10.1007/s11042-018-6209-9

Action recognition by fusing depth video and skeletal data information

Ioannis Kapsouras[1] · Nikos Nikolaidis[1]

Received: 29 August 2017 / Revised: 31 March 2018 / Accepted: 23 May 2018 /
Published online: 4 July 2018

Abstract Two action recognition approaches that utilize depth videos and skeletal information are proposed in this paper. Dense trajectories are used to represent the depth video data. Skeletal data are represented by vectors of skeleton joints positions and their forward differences in various temporal scales. The extracted features are encoded using either Bag of Words (BoW) or Vector of Locally Aggregated Descriptors (VLAD) approaches. Finally, a Support Vector Machine (SVM) is used for classification. Experiments were performed on three datasets, namely MSR Action3D, MSR Action Pairs and Florence3D in order to measure the performance of the methods. The proposed approaches outperform all state of the art action recognition methods that operate on depth video/skeletal data in the most challenging and fair experimental setup of the MSR Action3D dataset. Moreover, they achieve 100% correct recognition in the MSR Action Pairs dataset and the highest classification rate among all compared methods on the Florence3D dataset.

Keywords Kinect · Bag of Words · Vector of Locally Aggregated Descriptors · Action recognition · Fusion · Depth video · Motion capture data · MSR Action3D

1 Introduction

Depth videos have been lately used very often in computer vision and video analysis and understanding research, especially since the release of the Microsoft Kinect RGBD device in 2010 [45]. Kinect is able to record depth video data and can also track the skeletons of the humans depicted in the videos (Fig. 1), transforming motion capture (mocap), a rather expensive procedure until recently, to a common and affordable operation. Thus, many

✉ Ioannis Kapsouras
jkapsouras@aiia.csd.auth.gr

[1] Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

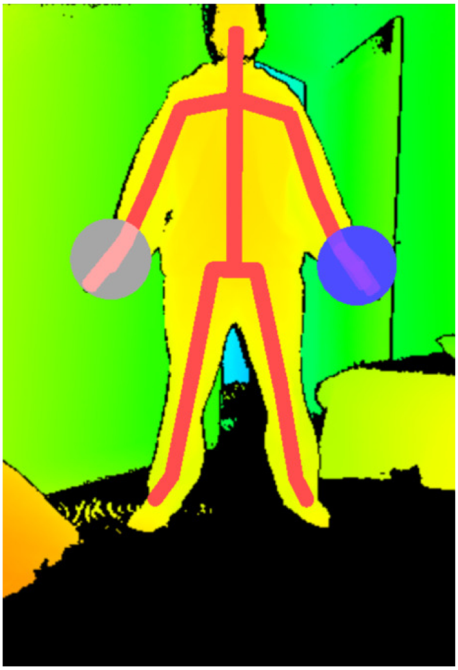


Fig. 1 A depth frame acquired by a Kinect device, alongside with the tracked skeleton of the depicted subject

algorithms that utilize skeletal data and/or depth video data have been introduced through the past few years.

One important research topic related to data acquired from depth cameras is action recognition. Action recognition is the process of labeling a motion sequence with respect to the human actions depicted in them. Action recognition has numerous applications including human computer interaction, video surveillance, multimedia annotation and retrieval, etc.

This paper presents two multimodal action recognition methods that operate on features derived from both depth and skeletal data. The proposed methods fuse information from both modalities and manage to achieve better action classification rates compared to those obtained by using information from only one modality.

The first approach is an extension of the method presented in [19]. In the proposed extension, depth videos are combined with skeletal information used in [19]. Dense trajectories [38] (actually their improved version) are used as features for the depth video since they have been shown to be very efficient for action recognition. Since their introduction, dense trajectories have dominated the area of event/action detection/recognition due to their superior performance over other video features. Similar to [19], posture vectors, containing the 3D positions of skeletal joints, and their forward differences in different temporal scales are extracted from skeletal data. These features can model complex human actions and take into account both the pose of the skeleton and its changes through time and have been shown to achieve state of the art action recognition results on skeletal data. The bag of words framework [7] is used to encode the sequences and Support Vector Machines (SVM) are used for classification. In more detail, a fuzzy scheme is used to encode dense trajectory features instead of the hard encoding scheme of the classic Bag of Words (BoW) framework. A voting scheme is used with the skeletal features. Information derived from both skeletal and depth video data is fused before the classification through kernel addition.

In the second approach, the same types of features are used to represent the skeletal and the depth video data. Vector of Locally Aggregated Descriptors (VLAD) [17] is used in this approach for encoding the features of each sequence. VLAD is similar to BoW and is applied in a similar way to both modalities (depth video and skeletal data). K-means is applied on the extracted features and then the association of each feature vector with each cluster center is computed. The main difference from BoW is that, instead of encoding each feature vector to its closest centers (centroids), the residuals of the feature vectors from the closest centers are encoded. The same fuzzy and voting schemes utilized in the first approach are also used here for video and skeletal features respectively, whereas a Radial Basis Function (RBF) kernel SVM is used for classification. The two modalities are fused with kernel addition as in the BoW framework. A flowchart of the two approaches is shown in Fig. 2.

The proposed approaches were tested in three action datasets and in various experimental setups. The methods outperformed all state of the art methods for action recognition in the most credible and challenging experimental setup of the MSR3D dataset. One of the proposed variants achieved perfect recognition in MSRAction Pairs dataset and two of the variants achieved the highest classification rate among all compared methods on the Florence3D dataset. Results show that the combined use of the two modalities can significantly increase the correct classification rate (Section 4). These findings verify the fact that approaches fusing information from more than one modality provide in most cases better results than their single-modality counterparts, since they can take advantage of the richer information contained in the combination of modalities. Of course, fusion is effective only if the used modalities bear complementary information i.e., they are uncorrelated. This seems to be indeed the case in skeletal and depth video data, when using the selected features. Due to the nature of depth video data, dense trajectories features on these data concentrate in the human silhouette. Thus the information they carry is, to a significant extent, complementary to the information carried by the skeletons. Moreover, skeletal data, especially those from Kinect, are often noisy. In this case, one can expect that information from the depth modality will be of significant help in the action recognition task.

The remaining of this paper is organized as follows. In Section 2, we present a review of previous work on this topic. Section 3 presents the proposed methods. Experimental performance evaluation and comparison with other approaches is presented in Section 4. Conclusions follow in Section 5.

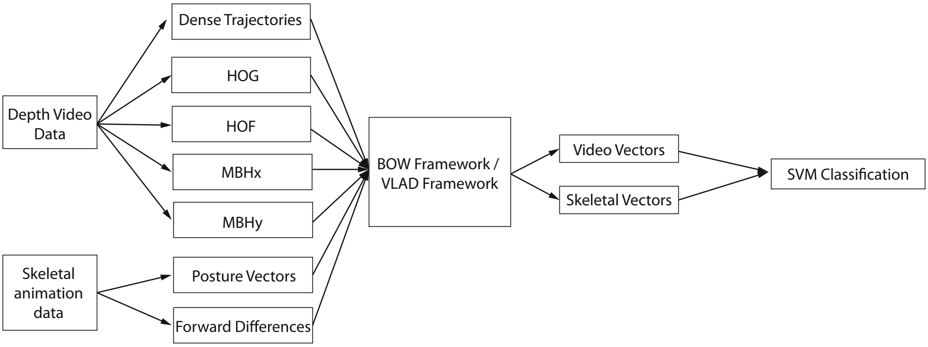


Fig. 2 Flowchart of the two proposed multimodal action recognition approaches

2 Related work

Action recognition from video data has been for many years a very active research field. Surveys and reviews of action recognition methods on such data can be found in [1, 14, 37]. A review of public datasets used for the experimental evaluation of such methods can be found in [4]. However, motion capture technology became widely available only during the last years. Hence the body of research for movement recognition on mocap/skeletal data is not as extensive as for video data. A review of spacetime representations of skeletal data for action recognition or related tasks is presented in [13]. In [18], the human poses were represented by codewords adopting the Tree-Structured Vector Quantization. Two approaches were followed for the classification: a spatial approach, based on the histogram of codewords, and a spatiotemporal one, based on codeword sequence matching. Shariat and Pavlovic performed activity classification by using a 1-NN classifier in [30]. They used as distance metric the alignment cost between sequences computed by a sequence alignment algorithm called IsoCCA. IsoCCA extends the Canonical Correlation Analysis (CCA) algorithm, by introducing a number of alternative monotonicity constraints. Their method achieved improved classification rates compared to other alignment algorithms, such as Canonical Time Wrapping (CTW), Dynamic Time Wrapping (DTW), Hungarian and CCA. In [27] a method for the classification of dance gestures represented by skeletal animation data is proposed. An angular skeleton representation that maps the motion data to a smaller set of features is applied. The full torso is fitted with a single reference frame that is used to parametrize the orientation estimates of joints. A cascaded correlation-based maximum-likelihood multivariate classifier is applied to build statistical models for the classes. The classifier compares the input data with the model of each class and generates a maximum-likelihood score. An input gesture is finally compared with a prototype one using a distance metric that involves DTW. The method proposed by Deng et al. in [9] applies the K-means algorithm in five partitions of a human model, namely, torso, left upper limb, right upper limb, left lower limb and right lower limb. Then, a generalized model is used to represent each K-means class. For continuous motion recognition, body partition index maps are constructed and applied, whereas for isolated motion recognition the authors propose a voting scheme that can be used with common dynamic programming techniques. They also present a new penalty-based similarity measure for DTW. The use of the most informative joints in order to represent skeletal sequences for action recognition was proposed by Ofli et al. in [22]. 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 neighbor and SVM are used for classification. Han et al. used a hierarchical discriminative approach in [12] for human action recognition. The human motion is represented in a hierarchical manifold space by performing a hierarchical latent variable space analysis. Conditional random fields are used to extract mutual invariant features from each manifold subscpace, and the classification is performed by an SVM classifier.

Action recognition in depth video data became more popular with the release of devices such as Microsoft Kinect. Indeed, a number of methods proposed for action recognition on Kinect data often use the depth video data. Li et al. in [20] proposed a method for action recognition in depth video data without the use of the corresponding tracked skeleton. They construct an action graph to encode human actions and propose a bag of 3D points approach to characterize a set of salient postures that correspond to the nodes in the action graph. They also propose a projection method to sample the 3D points from the depth maps. The MSR dataset, widely used in the action recognition community, is also introduced in this paper. Chen et al. proposed the use of the Local Binary Pattern (LBP) operator for action recognition in depth video data in [5]. In more detail, the LBP operator is applied in three Depth Motions Maps, calculated by projecting the frames of a depth video onto three orthogonal planes (front, side, and top). The authors used Kernel Extreme Learning Machine for the classification. Depth motion maps are also used by Wang et al. in [42]. At first, 3D point clouds are computed from the depth video data and weighted depth motion maps are generated. Three orthogonal projections and different temporal scales are used to calculate the depth motion maps. Deep convolutional neural networks (ConvNets) are trained for the classification. Another method applied on depth video data is proposed in [40]. The authors extract occupancy patterns features in 4D volumes. A weighted sampling approach is proposed for excluding subvolumes that do not contain any useful information. Sparse encoding is performed in order to encode the extracted features and SVM is used for classification. Rahmani et al. in [26] propose a method for action recognition applied on point clouds derived from depth video data. The method involves a detector and a descriptor for such point clouds. The descriptor is called Histogram of Oriented Principal Components (HOPC). PCA is performed in a volume around a point of the cloud and the resulting eigenvectors are projected onto different directions. The projections, scaled by the eigenvalues are concatenated to form the descriptor. The proposed spatio temporal detector is used to find keypoints in the point cloud, where the descriptors will be computed. SVM is used for classification. Wang and Wu proposed a method [39], called Maximum Margin Temporal Warping (MMTW), that learns to align action sequences and measure their matching score. A model is learned for each action class so as to achieve the maximum margin separation from the other classes. The learning process is developed as a latent structural SVM. The cutting plane algorithm is used to solve the SVM. Oreifej and Liu in [24] represent a depth action sequence by forming a histogram of the surface normal orientation in time, depth and space. Moreover, they use a novel discriminative density measure to refine the quantization and SVM for classification. Neural networks were used by Veeriah and Zhuang in [33] for action recognition. In more detail, the authors proposed the differential Recurrent Neural Network (dRNN), a variation of the long short-term memory (LSTM) neural network. The LSTM does not consider the impact of spatio-temporal dynamics in human actions. To address this problem, the authors proposed a differential gating scheme that captures the information gain between the frames. dRNN quantifies this information using Derivative of States (DoS).

Skeletal data, e.g. those provided by Kinect, were also used for action recognition. Xia et al. [43] used Hidden Markov Models to perform action recognition on 3D skeletal joint locations, extracted from Kinect depth data. The data were represented by histograms of 3D joint locations and action sequences were encoded using Linear Discriminant Analysis, clustering and vector quantization. Eweiei et al. proposed a method for action recognition on skeletal data in [10]. The authors used joints position, joints velocity and the correlation between location and velocity as features. Partial Least Squares (PLS) is used to learn a representation from these features and an SVM is used for classification. Luo et al. in [21] proposed the use of sparse coding to address the problem of action recognition in skeletal data. The differences of all joints from a reference frame are used as features. The extracted features are utilized from a dictionary learning algorithm that learns a different dictionary for each action. Features are quantized to the dictionaries and group sparsity, alongside with geometry constraints, are used in order to aid the proper reconstruction of features. A new set of features (local occupancy patterns) and a new temporal patterns representation (Fourier temporal pyramid) was proposed in [41] in order to represent 3D joint positions.

The authors defined 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. Chen et al. [6] proposed a two-level hierarchical framework for action recognition that operates on skeletal data. In the first level, the most important joints for each action are used to form a five dimensional vector. These vectors are used to cluster the action sequences. In the second level, motion feature extraction is performed by using only the relevant joints of the first level. Pairwise differences in different temporal scales are used as features and standard deviation is used to determine the time scale of these differences. Finally, action graphs applied to motion features are used for classification. The authors in [35] propose a new skeletal representation for action recognition. Instead of using the joint locations, they model the relative 3D geometry between different body parts as a point in the Lie group $SE_3 \times \cdots \times SE_3$, where × denotes the direct product between Lie groups. Hence, human actions are modeled as curves in the Lie group $SE_3 \times \cdots \times SE_3$. In order to perform classification, the authors map the action curves from the Lie group to its Lie algebra, which is the tangent space at the identity element of the group. Moreover, they use DTW and the Fourier temporal pyramid to cope with rate variations, temporal misalignment and noise. Finally, the use linear SVM for classification. Anirudh et al. in [3] use shape silhouettes on the Grassmann manifold [32] and skeletal joints as points on the product space $SE_3 \times \cdots \times SE_3$ [35] as features that lie on different manifolds. They embed the features in a lower dimensional manifold by using a manifold functional variant of PCA (mfPCA). Classification is done through SVM. Vemulapalli and Chellappa in [34] 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 [35] to model the human actions as curves in a Lie group. In order to classify the modeled 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. Gowayyed et al. in [11] 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. Yang and Tian in [44] proposed a new type of features for action recognition on skeletal data. The authors compute the pairwise joint differences within the current frame, between the current frame and the preceding frame and between the current frame and the initial frame. The final feature vector is formed by concatenating these three vectors. PCA is used to reduce noise and the final vectors are called EigenJoints. Naive-Bayes-Nearest-Neighbor (NBNN) is used for classification. The covariance matrix for skeleton joints locations over time is used as a descriptor by Hussein et al. in [15] to address the action recognition problem. To use the temporal dependency of joint locations, multiple covariance matrices are computed in a hierarchical fashion. Linear SVM is used for classification. Amor et al. in [2] 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 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.

The main difference of the methods presented above with the method proposed in this paper is the use of only one modality. In this paper, both depth and skeletal data are used to achieve high classification rates for action recognition.

A limited number of methods, i.e. those proposed in [23, 25, 36, 47] and [29] also use both video and skeletal data. Vieira et al. [36] introduced a new feature representation in order to combine spatial and temporal information and to encounter for the intra-actions variations. The proposed features are applied on depth maps. The authors divide space and time in small segments in order to create 4D cells and use a function to determine the number of space-time points that fall into these cells. The points of all cells form a high dimensional feature vector that represents an action. Skeletal information is also used in order to obtain view invariance. In [47], spatiotemporal features are extracted from video data. In more detail, Harris3D is used as a detector for keypoints and Histogram of Oriented Gradient (HOG), Histogram of Optical Flow (HOF), Histogram of Oriented Gradient 3D (HOG3D) and ESURF are used as descriptors. Pairwise distances of joints, joints differences between current and previous frame as well as joint differences between the current and the first frame are used to represent the skeletal data. A bag of words approach is used in both video and skeletal data and a random forest technique is applied to fuse and classify the histograms from the two modalities. Ohn-Bar and Trivedi [23] proposed two sets of features for action recognition on both depth video and skeletal data. For the depth data, the authors use a modified Histogram of Oriented Gradients approach called HOG^2. They compute histograms at each frame in box regions around each joint. The resulting histograms are concatenated and the algorithm is reapplied on this array to capture temporal dynamics. Affinities within sequences of joint angles are used for the skeletal data. Depth video and skeletal data features are used to represent an action in a bag of words approach. In [29] a multimodal method for action recognition that combines different features in the learning process is proposed. The authors use skeleton features, local occupancy patterns (LOP) and histogram of oriented 4D normals (HON4D). They use a joint sparsity regression based learning method to select the most discriminative joints for different action classes and use these joints to train classifiers. They also propose a hierarchical mixed norm which includes three levels of regularization over learning weights in order to model the hierarchy of the different types of features and build an integrated learning and selection framework. Rahmani et al. [25] also used information from both video and skeletal data. The depth video data are divided into subvolumes and depth alongside with depth gradient variations are encoded to histograms so as to form video features. The differences of each joint from a fixed joint position are encoded to histograms to form skeletal features. Moreover, a 3D space-time motion volume is computed for each joint, to encode the space-time area of the joint. These volumes, along with their differences from a fixed joint position are also used as skeletal features. Random Decision Forests (RDF) are used to fuse the different types of features and for classification.

The proposed methods are multimodal, like those reviewed in the previous paragraph i.e. [23, 25, 29, 36, 47]. The method proposed in [36] does not use skeletal data for feature extraction, as in the methods proposed in this paper, but only for alignment purposes. The differences between the proposed method and the method in [47] are the fusion technique and the features used to represent the sequences. Random forests are used in [47] to fuse histograms created by the bag of words framework while kernel addition is used for fusion before the SVM classification in the proposed method. Moreover, dense trajectories are used for feature representation of the video data in the proposed method, instead of spatiotemporal interest points (STIPs) used in [47]. Regarding the skeletal data, forward differences in different temporal scales are used instead of pairwise differences in [47]. The method in [23] and the proposed method both use the bag of words framework, but the proposed methods use a voting scheme instead of hard encoding. Moreover, this paper also investigates the use of a different encoding scheme (i.e. the VLAD framework). Also, different features were used both in video and skeletal data (Dense Trajectories instead of HOG^2 for video data and posture vectors and forward differences instead of affinities within sequences of joint angles). The method proposed in [29] also uses different features than the proposed methods. In the skeletal data modality, the authors perform a joint selection to select the most discriminative joints before feature extraction while the proposed methods use information from all the joints of the skeleton. The method in [25] uses also different types of features both for depth and skeletal data and a different classification framework (RDF instead of SVM).

In general, although the constituent parts (BOW, VLAD, dense trajectories etc) of the introduced methods were proposed elsewhere, the proposed combination is novel and leads to above the state-of-the-art or state-of-the-art results. It should be also stressed that, as mentioned above, there is a rather limited number of activity recognition methods that combine skeletal and depth data, a fact that gives our approaches an additional element of novelty. Finally, the methods have some individual novelty elements, the most important being the use of soft encoding techniques in the VLAD framework (Section 3.2.2) and the fact that, as far as we know, this is the first time that VLAD encoding is used with multimodal data.

3 Method description

3.1 Feature extraction

As already mentioned, the proposed approaches use information derived from both depth video and skeletal data. Different features are used in order to capture information from each modality. In more detail, dense trajectories are extracted from the video modality while posture vectors and forward differences are used to represent information from skeletal data. Both depth and skeletal features are computed in different temporal or spatiotemporal scales, hence the resulting representations are able to encode the high variety and dynamics of human motion.

3.1.1 Depth video data

Dense Trajectories features [38] are extracted from depth video data in the proposed approaches, as summarized below. At first, dense sampling is performed on a grid spaced by W pixels. Sampling is performed in a number of spatial scales and the sampled points are tracked through the video. In order to avoid samples in homogeneous image areas, the criterion presented in [31] is used to remove points from these areas.

Feature points are tracked on each spatial scale separately by computing the optical flow field $\boldsymbol{\omega}_t = (u_t, g_t)$ for each frame $\mathbf{I}_t$, where u_t and g_t are the horizontal and vertical components of the optical flow. Given a point $\mathbf{P}_t = (x_t, y_t)$ in frame $\mathbf{I}_t$, its tracked position in frame $\mathbf{I}_{t+1}$ is smoothed by applying a median filter on ω_t. Points of subsequent frames are concatenated to form trajectories $(\mathbf{P}_t, \mathbf{P}_{t+1}, \mathbf{P}_{t+2}, \ldots)$, whose length is limited to L frames. The shape of a trajectory is described by a sequence $(\varDelta\mathbf{P}_t, \ldots, \varDelta\mathbf{P}_{t+L-1})$ of displacement vectors, where $\Delta \mathbf{P}_t = (\mathbf{P}_{t+1} - \mathbf{P}_t)$. The resulting vector is normalized by the sum of displacement vector magnitudes.

A space-time volume aligned with a trajectory is also used to encode motion information. The size of the volume is $N \times N$ pixels and L frames long and is subdivided into a spatio-temporal grid. In each cell of this grid, various descriptors are computed. These include Histograms of Oriented Gradients (HOG) and Histograms of Optical Flow (HOF) descriptors and Motion Boundary Histograms (MBH) descriptors along the two dimensions x and y [8]. Summing up, five types of features are calculated for each depth sequence (Trajectories, HOG, HOF, MBHx and MBHy) resulting to 5 sets of feature vectors: f_i^{traj}, f_i^{HOG}, f_i^{HOF}, f_i^{MBHx}, and f_i^{MBHy}, $i = 1 \ldots Q$ where Q is the number of feature vectors extracted from a certain video. Dense trajectories extracted from a depth video sequence can be seen in Fig. 3. These descriptors are computed in different spatiotemporal scales, hence they can form a rich representation of human motion.

3.1.2 Skeletal data

In the proposed approaches, 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 [19]. However, the method proposed in [19] was unable to distinguish similar motions that have different directions (e.g., stand up and sit down). This is because the forward differences calculated in [19] were not able to distinguish the direction of the human motion. In this paper, the forward differences are computed in a slightly different manner (2) so as to encode directional information for the human motion. As will be shown in Section 4.2, the proposed methods can now distinguish similar actions with different directions.

Skeletal data are 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 posture 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.

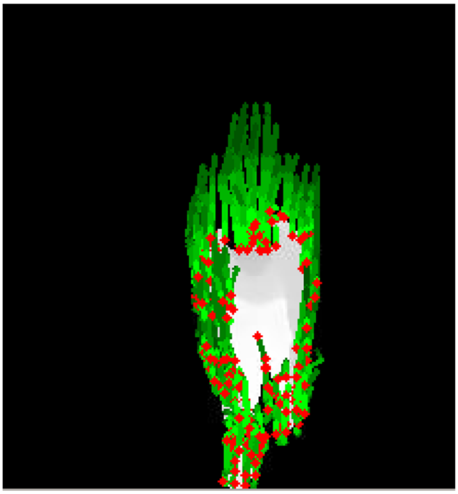


Fig. 3 Dense trajectories extracted from a depth video sequence. The red color indicates feature points, while the green color indicates the tracking of the feature points (i.e., the trajectory)

Skeletal 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 skeletal joints in frame i. In the proposed approaches, the joints forward differences are computed in different temporal scales, more specifically for $t = 1$, $t = 5$ and $t = 10$, in order to capture the joints dynamics. Posture vectors, alongside with forward differences can be seen in Fig. 4.

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: $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3, \mathbf{T}_4$:

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

3.2 Feature vector encoding

Two different approaches are used to encode the derived feature vectors, leading to two different variants. These approaches are described below.

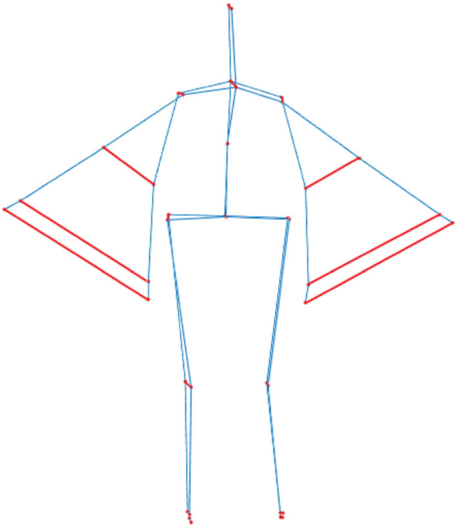


Fig. 4 The skeletal features. The postures vectors refer to the joint positions in the two skeletons ($t = 0$, $t = 10$), while the forward differences encode the displacement of the joints. The displacements of 6 joints are shown for $t = 10$ (red lines)

3.2.1 Bag of Words framework (BoW)

The bag of words framework with soft encoding can be summarized as follows. Feature vectors extracted from the training data (dense trajectories for video data and posture vectors along with forward differences for skeletal data) are clustered using K-means. It should be noted here that K-means is applied separately on each feature type, i.e. 5 times for the video features (trajectories, HOG, HOF, MBHx, MBHy) and 4 times for the skeletal features (posture vectors and forward differences for $t = 1$, $t = 5$ and $t = 10$). The centroids $\mathbf{c}_k, k = 1, \ldots, C$, where C is the number of K-means clusters in each of the feature spaces, form a discriminative representation of the feature vectors. Next, the feature vectors of each sequence are mapped to the corresponding centroids. A different mapping is used for video features and for skeletal features. Regarding video features, a fuzzy vector quantization is used as in [16]. In more detail, let $\mathbf{f}_i^j$ be a feature vector of the j-th sequence belonging to one of the five feature spaces and $i = 1 \ldots K_j$ where K_j is the number of feature vectors of j-th sequence. The fuzzy distances of $\mathbf{f}_i^j$ from centroids $\mathbf{c}_k$ in this feature space are calculated as follows:

$$e_{ik}^j = \left(\|\mathbf{f}_i^j - \mathbf{c}_k\|_2\right)^{\frac{2}{q-1}}, \ k = 1, \ldots, C \tag{4}$$

where q is the fuzzification parameter ($q > 1$). Thus, the $\mathbf{e}_i^j = \left[e_{i1}^j, e_{i2}^j, \ldots, e_{ik}^j\right]$ distance vector is formed. $\mathbf{e}_i^j$ is l^2 normalized and the final vector that represents the j-th sequence is formed as the mean of $\mathbf{e}_i^j$ vectors:

$$\mathbf{v}_j = \frac{\sum_{i=1}^{K_j} \mathbf{e}_i^j}{K_j} \tag{5}$$

This procedure is repeated for each video feature type. Thus 5 vectors are formed to represent the j-th sequence, namely $\mathbf{v}_j^{traj}$ for trajectories, $\mathbf{v}_j^{hog}$ for HOG, $\mathbf{v}_j^{hof}$ for HOF, $\mathbf{v}_j^{mbhx}$ for MBHx and $\mathbf{v}_j^{mbhy}$ for MBHy features.

For the skeletal features, a voting scheme is used for mapping [19]. In more detail, the similarities of each feature vector (belonging to one of the four feature spaces) of the j-th sequence with each centroid of this feature space are computed:

$$s_k^j = sim(\mathbf{c}_k, \mathbf{p}^j) = exp\left(-\left(\frac{\sum_{i=1}^{l}(\| c_{ki} - p_i^j \|_2)}{0.5 * \max\limits_k(\sum_{i=1}^{l}(\| c_{ki} - p_i^j \|_2))}\right)^2\right) \tag{6}$$

where s_k^j is the similarity between centroid $\mathbf{c}_k$ and a feature vector $\mathbf{p}^j$ of the j-th sequence. Then, a vector of ordered similarities: $S = [s_{(1)}^j, \ldots, s_{(C)}^j]$, where C is the number of clusters, is formed for each feature vector in sequence j. Finally, a C-dimensional vector $\mathbf{v}_j$ that characterizes the sequence (for this feature type) is formed by adding to the bin that corresponds to a cluster center the similarity of the feature vector with this center, starting from the most similar until the sum of the R largest similarities surpasses the $Y\%$ of the sum of all similarities. In other words, R is found as the value that satisfies the following inequalities:

$$\frac{\sum_{k=1}^{R-1} s_{(k)}^j}{\sum_{k=1}^{C} s_{(k)}^j} < Y < \frac{\sum_{k=1}^{R} s_{(k)}^j}{\sum_{k=1}^{C} s_{(k)}^j}, R \geq 2 \tag{7}$$

In the special case where $s_{(1)}^{j} > Y\sum_{k=1}^{C} s_{(k)}^{j}$, R is set to 1. This happens, for example, when $Y = 0$, a case that corresponds to hard encoding. This procedure is repeated for the 4 skeletal feature types, resulting to 4 vectors that represent the j-th sequence, namely $\mathbf{v}_j^p$ for posture vectors, and $\mathbf{v}_j^{\upsilon^1}$, $\mathbf{v}_j^{\upsilon^5}$, $\mathbf{v}_j^{\upsilon^{10}}$ for forward differences with $t = 1, 5, 10$ respectively.

In total, 9 vectors are used to represent the depth video and skeletal data of each action sequence.

3.2.2 Vector of Locally Aggregated Descriptors approach

In the second proposed approach, the features are encoded using the Vector of Locally Aggregated Descriptors (VLAD) framework [17]. The VLAD framework is similar to the BoW framework but also has important differences.

First, feature vectors are clustered in each feature space by using the K-means algorithm. As in the BoW framework, K-means was applied separately for each feature type, resulting to $9 * C$ centroids, where C is the number of clusters in each feature space. The main difference between VLAD and BoW lies in the calculation of the vector that represents a sequence. Instead of forming vectors that encode the distance or similarity of the feature vectors from the cluster centers as described in Section 3.2.1, the differences of the feature vectors from cluster centers are used to form vectors that represent the sequences.

In more detail, let $\mathbf{T}_{kj}$ $(k = 1\ldots 9)$ be the sets of features that have been extracted from the j-th sequence, in the 9 different feature spaces. K-means is applied separately in each feature type, resulting to C clusters in each feature space. Next, each feature vector is mapped to the cluster centers. As in the BoW framework, a fuzzy vector quantization is used for the video features and a voting scheme is used for the skeletal features. To the best of our knowledge, this is the first time that a voting scheme is used for encoding the features in VLAD framework. Let $\mathbf{b}_{ij}$ be the C-dimensional quantization/voting vector that encodes the association of the i-th feature vector of the j-th sequence with each cluster center. Thus, for the fuzzy vector quantization, the elements b_{ijm} of $\mathbf{b}_{ij} = \left[b_{ij1}, \ldots, b_{ijC}\right]$ $(m = 1, \ldots, C)$ are the fuzzy distances of feature vector i from each cluster center, while for 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 (see Section 3.2.1) of this feature vector or 0 otherwise. In the next step, vectors $\mathbf{v}'_z$ are formed as follows:

$$\mathbf{v}'^{j}_{z} = \sum_{i=1}^{M} b_{ijz}\left(\mathbf{t}_i^j - \mathbf{c}_z\right),\ z = 1, \ldots, C \tag{8}$$

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 $\mathbf{v}'^{j}_{z}$ is the same as that of the feature vectors. Then, square root normalization is applied to each v'^{j}_{zq} element of $\mathbf{v}'^{j}_{z}$ to obtain $\mathbf{v}''^{j}_{z}$:

$$v''^{j}_{zq} = sgn(v'^{j}_{zq})\sqrt{|v'^{j}_{zq}|},\ q = 1, \ldots, O \tag{9}$$

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

$$\mathbf{V}'_j = \begin{bmatrix} \mathbf{v}_1^j \\ \mathbf{v}_2^j \\ \vdots \\ \mathbf{v}_C^j \end{bmatrix} \tag{10}$$

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

$$\mathbf{V}_j = \frac{\mathbf{V}'_j}{\|\mathbf{V}'_j\|_2} \tag{11}$$

This procedure is repeated for each feature type, and, finally, each sequence is represented by 9 vectors, namely $\mathbf{V}_j^{traj}$ for trajectories, $\mathbf{V}_j^{hog}$ for HOG, $\mathbf{V}_j^{hof}$ for HOF, $\mathbf{V}_j^{mbhx}$ for MBHx, $\mathbf{V}_j^{mbhy}$ for MBHy, $\mathbf{V}_j^{p}$ for posture vectors and $\mathbf{V}_j^{\upsilon^1}$, $\mathbf{V}_j^{\upsilon^5}$, $\mathbf{V}_j^{\upsilon^{10}}$ for forward differences with $t = 1, 5, 10$ respectively.

3.3 Classification

SVM is used for classification in both approaches (BoW and VLAD). In more detail, χ^2 kernels are used for the BoW framework:

$$K(\mathbf{s}_j, \mathbf{s}_k) = exp\left(-\frac{1}{2A}\sum_{i=1}^{C}\frac{(s_{j,i} - s_{k,i})^2}{s_{j,i} + s_{k,i}}\right) \tag{12}$$

where A is the mean value of distances between all training samples. RBF kernels are used for the VLAD framework:

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = exp\{-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|_2^2\} \tag{13}$$

Since 9 vectors have been formed to represent each sequence, 9 kernels are computed, one for each feature type. The kernels are fused (i.e., information from both video and skeletal data is combined) by computing the mean kernel:

$$\begin{aligned}\mathbf{K}_f &= (\mathbf{K}_{traj} + \mathbf{K}_{hog} + \mathbf{K}_{hof} + \mathbf{K}_{mbhx} \\ &\quad +\mathbf{K}_{mbhy} + \mathbf{K}_{pos} + \mathbf{K}_{\upsilon^1} + \mathbf{K}_{\upsilon^5} + \mathbf{K}_{\upsilon^{10}})/9\end{aligned} \tag{14}$$

where $\mathbf{K}_{traj}$ the kernel formed from the dense trajectories features, $\mathbf{K}_{hog}$, $\mathbf{K}_{hof}$, $\mathbf{K}_{mbhx}$, $\mathbf{K}_{mbhy}$ the kernels formed from the HOG, HOF, MBHx, and MBHy features respectively (the video features) and $\mathbf{K}_{pos}, \mathbf{K}_{\upsilon^1}, \mathbf{K}_{\upsilon^5}$ and $\mathbf{K}_{\upsilon^{10}}$ the kernels formed from the skeletal features, namely the posture vectors and the forward differences for $t = 1, t = 5$ and $t = 10$.

4 Experimental results

The proposed method has been tested on three datasets, namely *MSR Action3D (MSR3D)* [20], *MSR Action Pairs (MSRPairs)* [24], and *Florence3D* [28]. MSR3D and MSRPairs datasets contain both depth video and skeletal animation data for each action sequence while Florence3D contains only skeletal animation data. Skeletal data from all three datasets were obtained using depth cameras, therefore these data are more noisy than those obtained from "traditional" (and more expensive) motion capture systems.

4.1 MSR3D dataset

The MSR3D dataset consists of 10 subjects performing 20 actions with 2 or 3 repetitions of each action: *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)*. In total there are 567 action sequences and, as stated in [41], 10 of these sequences are very noisy. In this paper, experiments using both the 567 and the 557 sequences were conducted. It should be also noted that PCA was applied to the positions of the joints to de-correlate the data. Both SVM classifiers (RBF and χ^2) were trained with values of the soft margin parameter in the range $2^{-20}, 2^{-19}, \ldots, 2^{19}, 2^{20}$ and the best results are presented. Moreover, q (4) was set to 1.2 and Y (7) was set to 0.05 for all experiments unless stated otherwise.

4.1.1 First experimental setup

The first experimental setup used to asses the performance of the proposed methods in MSR3D was initially introduced in [20]. All action sequences of the dataset were used in this setup. Odd subjects (1,3,5,7) were used for training and even subjects (2,4,6,8) were used for testing. A depth frame of the dataset alongside with the corresponding skeleton are shown in Fig. 5. The results of the proposed method alongside with those of a number of methods that use the same experimental setup are shown in Table 1. As can be observed, the multimodal (depth and skeletal) representation acquired from VLAD framework achieved the best results among the proposed variants when both 557 and all 567 sequences were used. Moreover, as expected, the classification rates achieved by the proposed approaches when features from both modalities were used are higher than the single modal approaches. One can also observe that, in all variants, the features acquired from the skeletal data achieve

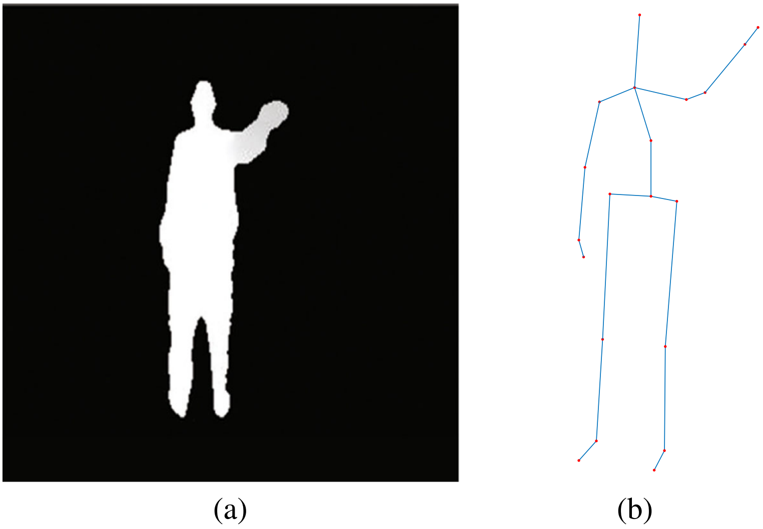


Fig. 5 A frame from an MSR Action3D sequence: **a** depth data, **b** skeletal data

Table 1 Correct classification rates in the experimental setup proposed in [20] on the MSR Action3D dataset

	557 sequences	Unknown number of sequences	567 sequences
skeletal BoW	92.67		92
depth BoW	91.21		90.55
skeletal + depth BoW	95.6		94.18
skeletal VLAD	92.31		91.27
depth VLAD	91.58		91.27
skeletal + depth VLAD	97.8		96.36
Wang et al. [42] (depth)	–	**100**	–
Eweiwi et al. [10] (skeletal)	92.3	–	–
Chen et al. [5] (depth)	91.94	–	–
Rahmani et al. [26] (depth)	–	–	91.64
Wang and Wu [39] (depth)	–	92.7	–
Luo et al. [21] (skeletal)	–	–	96.7
Wang et al. [41] (skeletal)	88.2	–	–
Oreifej and Liu [24] (depth)	–	88.89	–
Vieira et al. [36] (depth + skeletal)	–	81.5	–
Rahmani et al. [25] (depth +skeletal)	–	90.3	–
Ohn-Bar and Trivedi al [23] (depth + skeletal)	–	94.84	–
Vemulapalli et al. [35] (skeletal)	–	89.48	–
Veeriah and Zhuang [33] (depth)	–	–	92.03
Amor et al. [2] (skeletal)	–	89	–
Shahroudy et al. [29] (depth + skeletal)	–	–	93.1
Hussein et al. [15] (skeletal)	–	90.53 (544 sequences)	–
Gowayyed et al. [11] (skeletal)	–	–	91.26

When the number of sequences used in the experiments was not stated in the corresponding paper, a single result is provided

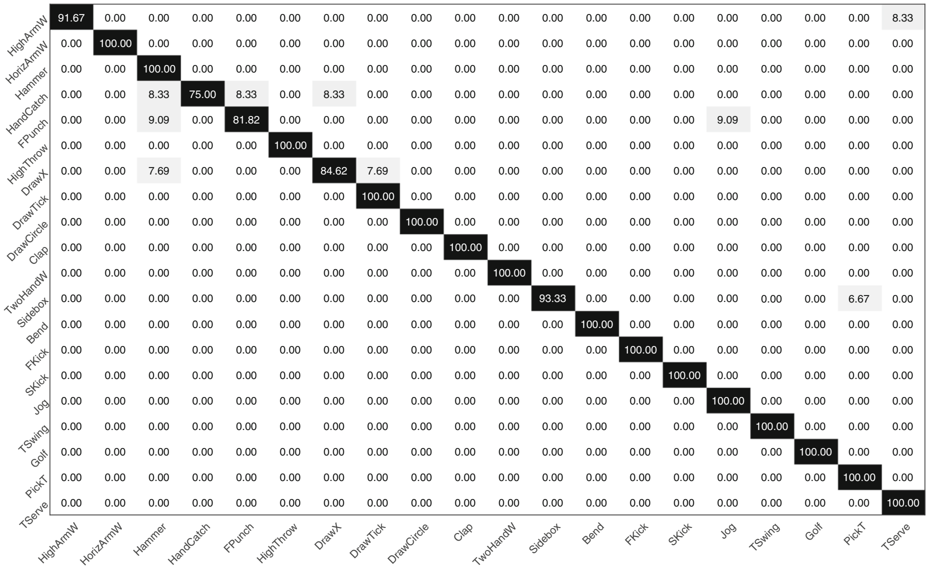
Values in underline correspond to the best score achieved by the proposed variants

Values in bold correspond to the best overall score achieved

slightly better classification results than the depth video features. Furthermore, the proposed multimodal variants outperform all other methods except for two. The first one, proposed in [42] achieves 100% classification rate, outperforming the best proposed variant by 3.64%, while the method in [21] achieves better performance only by about 0.34% in the case of 567 sequences. The confusion matrix for the proposed depth + skeletal VLAD approach, for the case of 557 sequences, is presented in Fig. 6. As can be seen in this figure, 16 out of 20 actions are recognized with 100% classification rate. For the remaining experiments on the MSR dataset, presented in Sections 4.1.2 and 4.1.3, the more challenging set of the 567 sequences was used.

4.1.2 Second experimental setup

A second experimental setup [24] was used to assess the performance of the proposed method. In this setup, all possible combinations of using 5 persons for training and the rest for testing were used. These combinations construct a 252-fold cross validation setup.



	HighArmW	HorizArmW	Hammer	HandCatch	FPunch	HighThrow	DrawX	DrawTick	DrawCircle	Clap	TwoHandW	Sidebox	Bend	FKick	SKick	Jog	TSwing	Golf	PickT	TServe
HighArmW	91.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	8.33
HorizArmW	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Hammer	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
HandCatch	0.00	0.00	8.33	75.00	8.33	0.00	8.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FPunch	0.00	0.00	9.09	0.00	81.82	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	9.09	0.00	0.00	0.00	0.00
HighThrow	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DrawX	0.00	0.00	7.69	0.00	0.00	0.00	84.62	7.69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DrawTick	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
DrawCircle	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Clap	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TwoHandW	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Sidebox	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	93.33	0.00	0.00	0.00	0.00	0.00	0.00	6.67	0.00
Bend	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FKick	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
SKick	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00
Jog	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00
TSwing	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00
Golf	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00
PickT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00
TServe	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00

Fig. 6 Confusion matrix (20 classes/actions) for fusion + VLAD approach with 96.36% overall correct classification rate (MSR3D datset)

This setup is obviously more fair and credible. The corresponding results are shown in Table 2. The best, worst and the mean ± std classification rates are presented. As can be observed, when both skeletal and depth data are used, both the proposed approaches (BoW and VLAD) outperform all state-of-the-art methods that have been tested with the same experimental setup. It can be also seen that, similar to the first setup, the fusion of depth with skeletal features can increase the classification rates obtained with the use of only one modality. It should be noted that the VLAD variant achieves slightly better results than BOW and has more consistent performance across folds, as indicated by its smaller standard deviation.

4.1.3 Third experimental setup

In the third experimental setup, proposed in [20], the dataset sequences were divided into three subsets (AS1, AS2 and AS3), each containing 8 actions and recognition was performed separately within each subset. The actions that form each subset are shown in Table 3. The AS1 and AS2 subsets group actions with similar movements, while AS3 groups complex actions. Three different tests, proposed in [20], were performed using subsets AS1, AS2 and AS3 in order to evaluate the performance of the proposed methods and for comparison with the state-of-the-art. In the first test (TEST 1), 1/3 of the sequences were used for training and the remaining ones for testing, while in the second test (TEST 2), 2/3 of the sequences were used to form the training set and the remaining sequences were used for testing. Finally, in the third test (TEST 3), action sequences of the odd subjects were used for training and those of the even subjects for testing. TEST 1 and TEST 2 test performance in small/large training sets respectively and TEST 3 tests performance when the training and test sets consist of different subjects. The overall classification results (over all three

Table 2 Correct classification rates in the experimental setup of 252-fold cross validation on the MSR Action3D dataset (567 sequences)

	min	max	mean ± std
skeletal BoW	78.35	91.27	85.10 ± 2.55
depth BoW	70	90.15	80.93 ± 4.26
skeletal + depth BoW	79.04	96.01	89.4 ± 2.97
skeletal VLAD	72.6	92.73	83.33 ± 0.0364
depth VLAD	73.10	92.39	84.13 ± 0.0384
skeletal + depth VLAD	80	96.01	**89.66** ± **0.0314**
Eweiwi et al. [10] (skeletal)	–	–	88.38 ± 0.027
Oreifej and Liu [24] (depth)	–	–	82.15 ± 4.18
Rahmani et al. [26] (depth)	74.36	92.39	86.46 ± 2.28
Rahmani et al. [25] (depth skeletal)	70.9	90.3	82.7 ± 3.3
Anirudh et al. [3]	82.03	88.29	85.16 ± 3.13
Chen et al. [6] (depth + skeletal)	77.39	95.56	87.05 ± 3.75

Values in bold correspond to the best overall score achieved

subsets AS1 − 3) for the three tests are shown in Table 4 alongside with results of methods that were tested in the same experimental setup.

As can be seen in this table, the proposed methods achieve very high classification rates in all three tests. The lowest classification rates were obtained in TEST 1, where only the $1/3$ of the sequences were used for training. Very high classification rates were achieved in TEST 2, which is the easiest one, since $2/3$ of the sequences were used for training. The proposed methods also achieved very good classification rates in TEST 3 which is the fairest one, since sequences of the same subject cannot coexist in both the training and test sets. The multimodal variants (depth + skeletal BOW and VLAD) achieved higher performance than their single modality counterparts in all three tests.

The proposed methods were surpassed by one method from the competition and outperformed all others in each test. In more detail, the method proposed in [21] achieved slightly better classification rates (0.86%) from the multimodal BOW variant in TEST 1, while the method proposed in [5] achieved perfect recognition in TEST 2, surpassing the multimodal VLAD variant by 0.43%. The multimodal VLAD variant was also surpassed by the method proposed in [29] in TEST 3 by 0.6%. However the method in [29] was not able to achieve as high classification rates as the multimodal VLAD variant in the more difficult first experimental setup (Section 4.1.1, Table 1).

Table 3 The three subsets of actions from the MSR database used in the experiments as proposed in [20]

Action Set 1 (AS1)	Action Set 2 (AS2)	Action Set 3 (AS3)
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 4 Correct classification rates in the third experimental setup (TEST 1, TEST 2 and TEST 3) proposed in [20] on the MSR Action3D dataset

	TEST 1	TEST 2	TEST 3
skeletal BoW	94.98	94.85	96.17
depth BoW	87.02	96.96	94.84
skeletal + depth BoW	98.04	99.56	96.34
skeletal VLAD	95.85	98.7	94.26
depth VLAD	90.99	97.35	94.25
skeletal + depth VLAD	95.62	99.57	97.6
Gowayyed et al. [11] (skeletal)	–	–	91.26
Yang and Tian [44] (skeletal)	95.8	97.78	82.33
Shahroudy et al.[29] (depth + skeletal)	–	–	**98.2**
Chen et al. [5] (557 seq.) (depth)	98.7	**100**	94.9
Luo et al. [21] (557 seq.) (skeletal)	**98.9**	98.9	96.7
Vieira et al. [33] (557 seq.) (depth)	96.8	98.27	87.5
Vemulapalli et al. [35] (skeletal)	95.29	83.87	92.46

The results are the overall classification rates achieved in the three subsets (AS1, AS2 and AS3)

Values in underline correspond to the best score achieved by the proposed variants

Values in bold correspond to the best overall score achieved

4.2 MSRPairs dataset

The MSRPairs dataset was introduced in [24]. The main characteristic of this dataset is that it consists of 6 pairs of actions: *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 rates can be seen in Table 5.

One can observe that all the variants of the proposed methods achieve high classification rates. Again, mulitmodal variants achieve, as expected, better classification rates than the single modal ones. The skeletal + depth VLAD variant alongside with the method proposed in [29] achieve the highest classification rate for this dataset (100%). All other methods tested on this dataset yield lower rates.

Table 5 Correct classification rates in the experimental setup proposed in [24] on the MSRPairs dataset

	Classification rate
skeletal BoW	92.13
depth BoW	92.22
skeletal + depth BoW	97.8
skeletal VLAD	95.51
depth VLAD	95.56
skeletal + depth VLAD	**100**
Shahroudy et al. [29] (depth + skeletal)	**100**
Vemulapalli and Chellappa [34] (skeletal)	94.09
Amor et al. [2] (skeletal)	93
Eweiwi et al. [10] (skeletal)	99.4
Rahmani et al. [26] (depth)	98.33
Wang and Wu [39] (depth)	97.22
Oreifej and Liu [24] (depth)	96.67

Values in bold correspond to the best overall score achieved

Table 6 Correct classification rates in the Florence3D dataset

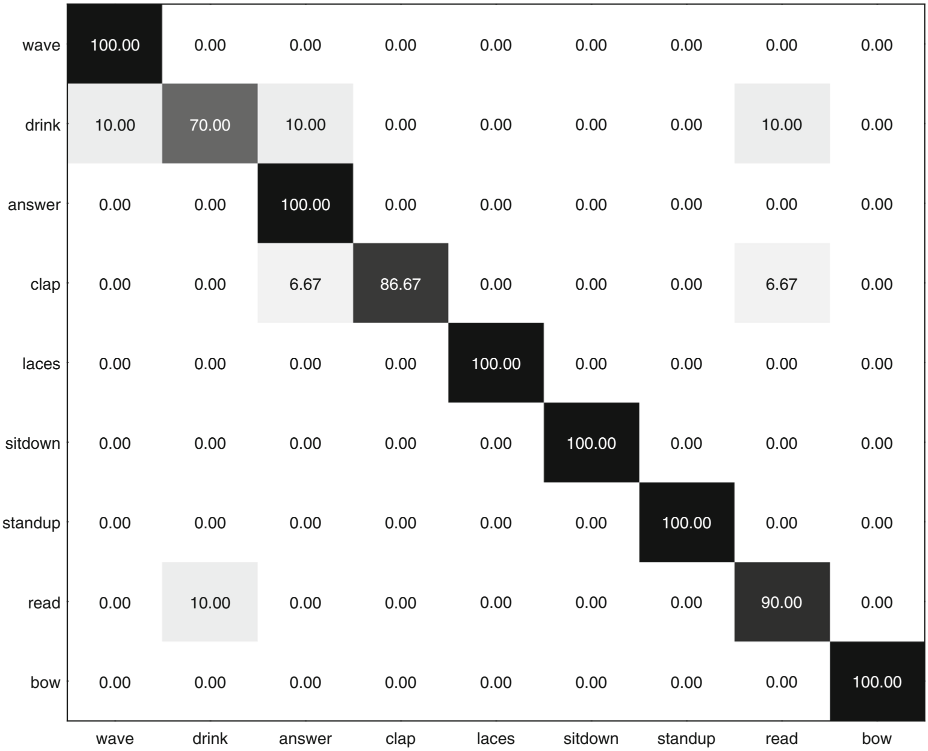
	Classification rate
skeletal BoW	**94.34**
skeletal VLAD	91.51
Vemulapalli and Chellappa [34]	89.82
Vemulapalli et al. [35]	90.88
Anirudh et al. [3]	89.67

Values in bold correspond to the best overall score achieved

4.3 Florence3D dataset

Florence3D [28] is a dataset collected at the University of Florence and consists of 9 actions, namely *wave (wave)*, *drink from a bottle (drink)*, *answer phone (answer)*, *clap (clap)*, *tight laces (laces)*, *sit down (sitdown)*, *stand up (standup)*, *read watch (read)*, *bow (bow)*. These actions are performed by 10 subjects and each subject performs each action 2 or 3 times. There are 215 sequences in total in this dataset. Sequences of half of the subjects were used for training and the other half for testing. Classification rates for Florence3D can be seen in Table 6. It should be noted that depth videos are not provided for this dataset, hence classification rates refer only to the skeletal data.

As can be seen in this table, the proposed methods achieve higher classification rates compared to those achieved by other methods that report results in this dataset. The BoW



	wave	drink	answer	clap	laces	sitdown	standup	read	bow
wave	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
drink	10.00	70.00	10.00	0.00	0.00	0.00	0.00	10.00	0.00
answer	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00	0.00
clap	0.00	0.00	6.67	86.67	0.00	0.00	0.00	6.67	0.00
laces	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00
sitdown	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00	0.00
standup	0.00	0.00	0.00	0.00	0.00	0.00	100.00	0.00	0.00
read	0.00	10.00	0.00	0.00	0.00	0.00	0.00	90.00	0.00
bow	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00

Fig. 7 Confusion matrix (9 classes / actions) for skeletal BoW approach with 94.34% overall correct classification rate (Florence3D dataset)

variant achieved the highest rates. The confusion matrix for this variant can be seen in Fig. 7. 6 out of 9 classes are classified with 100% classification rate. The action with the lowest classification rate is *drink* and is confused with *wave*, *answer* and *read*.

4.4 Discussion

4.4.1 Effect of the number of clusters C in K-means

K-means is used in both frameworks (BOW and VLAD) for the evaluation of the codewords that will be used for the representation of skeletal and depth features. An obvious question is how the number of clusters (C) affects the performance of the proposed frameworks. Classification rates for various values of C for the MSR3D dataset and for the first experimental setup can be seen in Fig. 8. As can be observed, with the exception of depth + BOW variant, C has not strong impact to the classification rates achieved by the proposed methods.

4.4.2 Effect of parameters Y and q

A common step in both approaches (BOW and VLAD) is the soft encoding of the features. A fuzzy vector quantization is used for the video features (4) and a voting scheme is used for the skeletal features (7). Two parameters that can affect the classification rate of the proposed methods are the fuzzification parameter q for the video features in (4) and parameter Y in (7). The classification rate for various values of Y for the VLAD variant

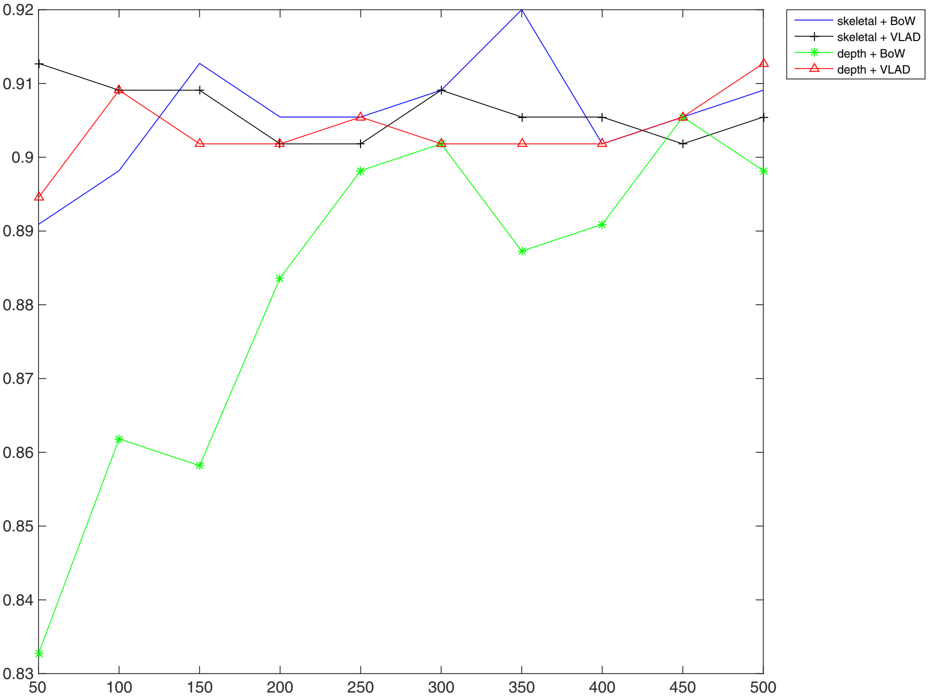


Fig. 8 Classification rates for various numbers of cluster centers C

(when only skeletal features are used) can be seen in Fig. 9. Hard encoding ($Y = 0$) leads to inferior results than soft encoding ($Y > 0$) for Y values up to 0.75. However the performance increases only in the range $0 < Y \leq 0.15$ and then decreases, remaining above the hard encoding performance up to $Y = 0.75$. Soft encoding utilizes information from more cluster centers than the closest one, hence the increase of the classification rate is expected since feature vectors can have high similarity with more than one cluster centers. But as Y increases, information from less similar cluster centers is used in the encoding, thus the representations are less distinct and the classification rate decreases. As a rule of thumb, values in the range $0.05 \leq Y \leq 0.15$ shall be used.

The effect of parameter q in the encoding of video features (4) for the VLAD variant (when only depth video features are used) can be seen in Fig. 10. It is obvious that the classification rate increases for $1.1 \leq q \leq 1.3$ and then decreases. The explanation for this behavior is similar to that given for parameter Y above. A small increase of q result to increased classification rate since information from the more similar cluster centers is used. However, by further increasing q, information from more distant cluster centers is used, resulting to more noisy representations.

4.4.3 *Effect of features combinations and SVM kernel*

The two proposed variants, combine a number of different features through kernel addition. An interesting question is whether specific combinations of features lead to good classification rates. Results for such combinations can be seen in Table 7.

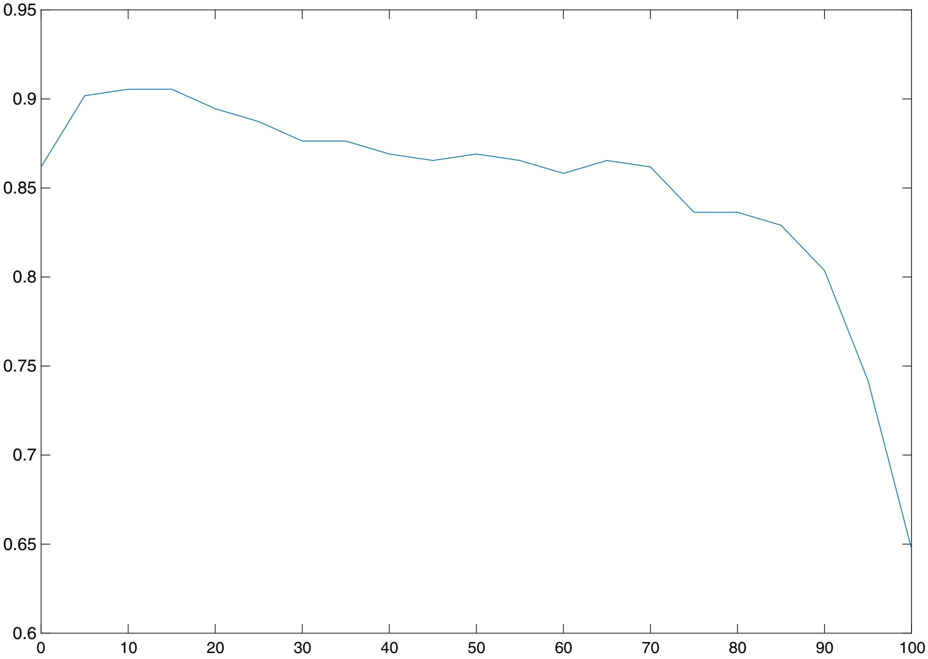


Fig. 9 Classification rates for various values of parameter Y, for the VLAD variant, when only skeletal features are used (MSR3D dataset)

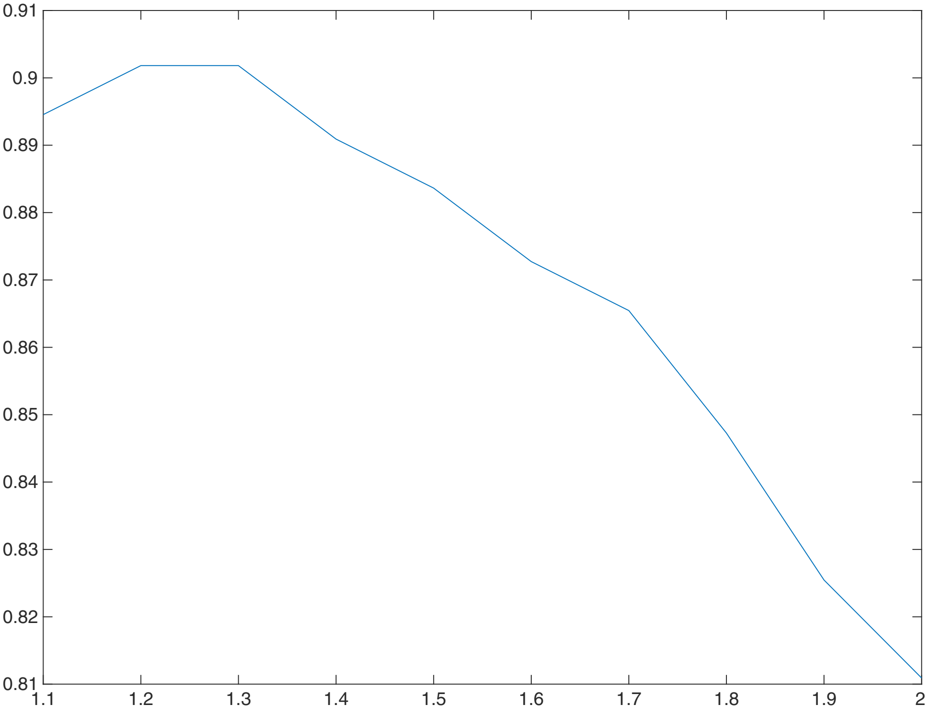


Fig. 10 Classification rates for various values of parameter q, for the VLAD variant, when only video features are used (MSR3D dataset)

One can observe that, the best classification rates are achieved when all depth and skeletal features are taken into account, both for VLAD and BOW variants, thus verifying our decision to use all these features in the proposed approaches. Good results were also obtained when combining the forward differences skeletal features with all or most of the depth features. As a matter of fact, in the case of the VLAD variant, the combination of forward differences with all depth features or just with MBHxy, HOG and HOF features provides the same results as those obtained when using the full set of features. This can be perhaps explained by the fact that forward differences capture the dynamics of human motion. On the

Table 7 Classification rates for various combinations of features (MSR3D dataset)

Features fused	BOW	VLAD
postures + trajectories	79.64%	80.36%
differences + HOG + HOF + MBHxy	93.45%	**96.36**%
postures + all depth features	90.91%	94.55%
differences + all depth features	93.54%	**96.36**%
MBHxy + all skeletal features	92.73%	94.55%
all features (depth and skeletal)	**95.60**%	**96.36**%

Values in bold correspond to the best overall score achieved

Table 8 Classification rates for various kernels (MSR3D dataset)

Variant	Kernel	Classification rate
BOW	Linear	90.54%
	RBF	93.09%
	χ^2	**95.60**%
VLAD	Linear	96%
	RBF	**96.36**%

Values in bold correspond to the best overall score achieved

contrary, combinations that involve posture vectors of skeletal data provide inferior results, which can be attributed to the lack of temporal /dynamic information in these features.

The SVM kernels used by the BOW and VLAD variants also affect the classification rate. Classification rates observed when using χ^2, RBF and Linear kernels can be seen in Table 8. χ^2 kernel is not favorably applicable in the case of VLAD due to the large size of final feature vectors (having $O \times C$ dimensions).

As can be seen in this table, the best result for the BOW variant is achieved using χ^2 kernels. This result is expected since this kernel provides very good results in the case of codebook representations [46]. In the VLAD variant, RBF and Linear kernels achieve very similar results, RBF being slightly better.

4.4.4 Computational complexity considerations

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 s) can be seen in Table 9. The framework that was used for classification was trained with 200 and 250 cluster centers for skeletal and depth video data respectively (i.e., those that achieved the best classification rates). The experiment ran on a PC with a quad-core processor and 8 GB of RAM. Dense Trajectories were computed using C++ under Linux (with

Table 9 Computational time (in seconds) of the proposed frameworks

Data	Method component	BoW	VLAD
Skeletal	Feature extraction	0.014	
	Feature encoding	0.087	0.144
	Classification	0.333	1.927
Depth	Feature extraction	15.41	
	Feature encoding	0.42	0.504
	Classification	1.22	3.245
Skeletal + Depth	Feature extraction	15.424	
	Feature encoding	0.507	0.648
	Classification	1.55	5.172
Overall skeletal		0.434	2.084
Overall depth		17.05	19.16
Overall skeletal + Depth		17.48	21.24

the code provided by [38]) and the rest of the computations were made using unoptimized MatLab code under Windows.

As can be seen in this table, the BoW framework is faster than the VLAD one both in the feature encoding and the classification step. This is easily explained since the dimensionality of the encoded features is $O \times C$ in VLAD where O is the dimensionality of the feature vector and C the number of the cluster centers and only C in BoW. It can also be seen that the most time consuming step is the computation of the depth features (dense trajectories). The single modality variant that uses skeletal data and BoW is the fastest one (0.42 s for a sequence of 2 s duration) and, considering a time window for continuous classification, this variant can be used in real time classification scenarios. The skeletal + VLAD variant (2.08 s for a sequence of 2 s duration) is also suitable for real-time operation. The other variants involving depth or multimodal data are significantly slower but given appropriate hardware and optimized implementation, can also operate in real time.

A critical parameter for the computational complexity of the proposed methods is the number of clusters C. C affects the classification and feature encoding steps, but, obviously, not the feature extraction step. The effect of C in the computational complexity of the classification step is shown in Fig. 11 since, as can be seen in Table 9, this step is more time consuming than feature encoding. As can be observed, the time needed for classification for the proposed methods is almost linear to the number of clusters and is larger and grows faster for VLAD. However, according to Fig. 8, good classification results can be achieved even with a small C. Hence, with a small sacrifice in classification rates, the overall classification time can be kept fairly low.

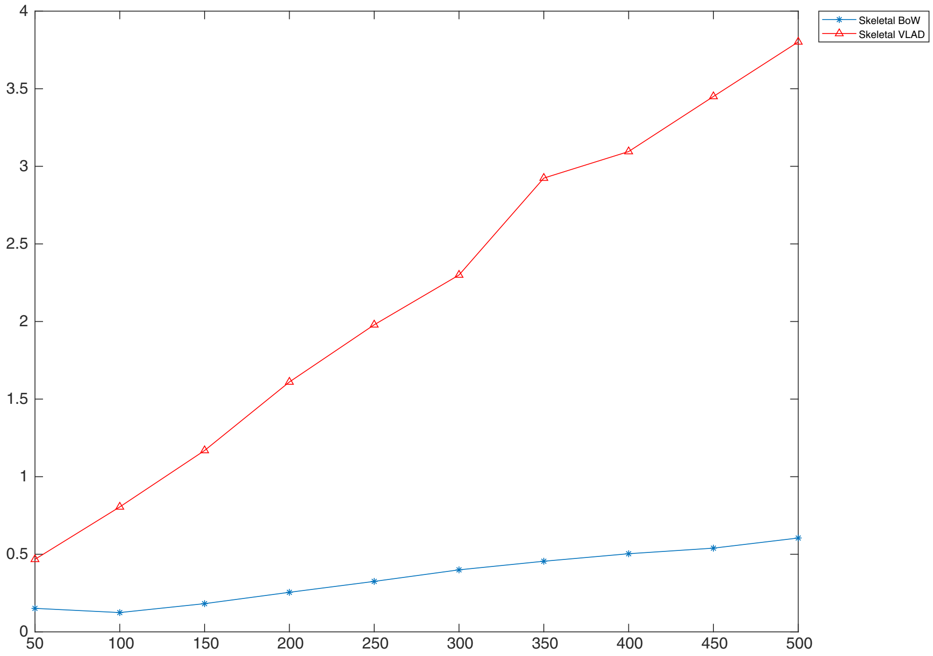


Fig. 11 Computational time (in seconds) for the classification step of the skeletal variants for different numbers of clusters

5 Conclusions

Two approaches for human action recognition that exploit both depth video data and skeletal data are proposed in this paper. Different types of features are extracted from each modality and two frameworks (BoW and VLAD) are used to encode these features. A fuzzy vector quantization is used for the encoding of the video features while a voting scheme is used for the skeletal features. SVM is used for classification and the various representations extracted from the features are fused using kernel addition. Experiments showed that the use of both depth and skeletal data leads to enhanced action recognition performance compared to that achieved when data of only one type are used. The variants of the method that fuse information from both depth and skeletal data outperform all but two existing methods in the cross subject test of the MSR Action3D dataset. Moreover, they achieve the best classification rate among competitors in the more fair and challenging 252-fold cross validation test of the same dataset. The correct recognition rate in the MSRPairs dataset is 100%. Finally, the variants that involve only skeletal data achieve the best classification rates among the compared methods in Florence3D skeletal dataset. In the future, extension toward motion clustering, segmentation, indexing and retrieval will be considered. Moreover, the combination of both frameworks with weighted kernel addition will be investigated.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Aggarwal J, Ryoo M (2011) Human activity analysis: a review. ACM Comput Surv 43(3):16:1–16:43
2. Amor BB, Su J, Srivastava A (2016) Action recognition using rate-invariant analysis of skeletal shape trajectories. IEEE Trans Pattern Anal Mach Intell 38(1):1–13
3. Anirudh R, Turaga P, Su J, Srivastava A (2015) Elastic functional coding of human actions: from vector-fields to latent variables. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 3147–3155
4. Chaquet JM, Carmona EJ, Fernández-Caballero A (2013) A survey of video datasets for human action and activity recognition. Comput Vis Image Underst 117(6):633–659
5. Chen C, Jafari R, Kehtarnavaz N (2015) Action recognition from depth sequences using depth motion maps-based local binary patterns. In: Proceedings of 2015 IEEE winter conference on applications of computer vision (WACV), pp 1092–1099
6. Chen H, Wang G, Xue JH, He L (2016) A novel hierarchical framework for human action recognition. Pattern Recognit
7. Csurka G, Dance CR, Fan L, Willamowski J, Bray C (2004) Visual categorization with bags of keypoints. In: Proceedings of workshop on statistical learning in computer vision (ECCV '04), pp 1–22
8. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: Proceedings of the 9th European conference on computer vision—volume part II, ECCV'06. Springer, Berlin, pp 428–441
9. Deng L, Leung H, Gu N, Yang Y (2012) Generalized model-based human motion recognition with body partition index maps. Comput Graphics Forum 31(1):202–215
10. Eweiwi A, Cheema MS, Bauckhage C, Gall J (2014) Efficient pose-based action recognition. In: Cremers D, Reid I, Saito H, Yang MH (eds) Proceedings of the Asian conference on computer vision (ACCV 14). Springer International Publishing
11. Gowayyed MA, Torki M, Hussein ME, El-Saban M (2013) 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, IJCAI '13. AAAI Press, pp 1351–1357
12. Han L, Wu X, Liang W, Hou G, Jia Y (2010) Discriminative human action recognition in the learned hierarchical manifold space. Image Vis Comput 28(5):836–849

13. Han F, Reily B, Hoff W, Zhang H (2017) Space-time representation of people based on 3D skeletal data. Comput Vis Image Underst 158(C):85–105
14. Holte MB, Tran C, Trivedi MM, Moeslund TB (2011) 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, J-HGBU '11. ACM, New York, pp 47–52
15. Hussein ME, Torki M, Gowayyed MA, El-Saban M (2013) 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, IJCAI '13. AAAI Press, pp 2466–2472
16. Iosifidis A, Tefas A, Nikolaidis N, Pitas I (2012) Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis. Comput Vis Image Underst 116(3):347–360. Special issue on Semantic Understanding of Human Behaviors in Image Sequences
17. Jegou H, Douze M, Schmid C, Perez P (2010) Aggregating local descriptors into a compact image representation. In: Proceedings of 2010 IEEE conference on computer vision and pattern recognition (CVPR), pp 3304–3311
18. Kadu H, Kuo M, Kuo CCJ (2011) Human motion classification and management based on mocap data analysis. In: Proceedings of the 2011 joint ACM workshop on human gesture and behaviour understanding. ACM, New York, pp 73–74
19. Kapsouras I, Nikolaidis N (2014) Action recognition on motion capture data using a dynemes and forward differences representation. J Vis Commun Image Represent 25(6):1432–1445
20. Li W, Zhang Z, Liu Z (2010) Action recognition based on a bag of 3D points. In: Proceedings of 2010 IEEE computer society conference on computer vision and pattern recognition workshops, pp 9–14
21. Luo J, Wang W, Qi H (2013) Group sparsity and geometry constrained dictionary learning for action recognition from depth maps. In: Proceedings of the 2013 IEEE international conference on computer vision (ICCV), pp 1809–1816
22. Ofli F, Chaudhry R, Kurillo G, Vidal R, Bajcsy R (2014) Sequence of the most informative joints (smij): a new representation for human skeletal action recognition. J Vis Commun Image Represent 25(1):24–38
23. Ohn-Bar E, Trivedi MM (2013) Joint angles similiarities and HOG^2 for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops: human activity understanding from 3D data, CVPR '13. IEEE Press
24. Oreifej O, Liu Z (2013) 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, CVPR '13. IEEE Computer Society, Washington, DC, pp 716–723
25. Rahmani H, Mahmood A, Huynh D, Mian A (2014) Real time action recognition using histograms of depth gradients and random decision forests. In: Proceedings of the 2014 IEEE winter conference on applications of computer vision (WACV), pp 626–633
26. Rahmani H, Mahmood A, Q Huynh D, Mian A (2014) HOPC: histogram of oriented principal components of 3D pointclouds for action recognition. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Proceedings of the 13th European conference on computer vision (ECCV 14), Zurich, Switzerland, September 6–12, 2014, Proceedings, Part II. Springer International Publishing, Cham, pp 742–757
27. Raptis M, Kirovski D, Hoppe H (2011) Real-time classification of dance gestures from skeleton animation. In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation. ACM, New York, pp 147–156
28. Seidenari L, Varano V, Berretti S, Bimbo AD, Pala P (2013) Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In: IEEE conference on computer vision and pattern recognition workshops, pp 479–485
29. Shahroudy A, Ng TT, Yang Q, Wang G (2016) Multimodal multipart learning for action recognition in depth videos. IEEE Trans Pattern Anal Mach Intell 38(10):2123–2129
30. Shariat S, Pavlovic V (2011) Isotonic cca for sequence alignment and activity recognition. In: Proceedings of the international conference on computer vision
31. Shi J, Tomasi C (1994) Good features to track. In: Proceedings of the 1994 IEEE computer society conference on computer vision and pattern recognition, 1994 (CVPR '94), pp 593–600
32. Turaga P, Chellappa R (2009) Locally time-invariant models of human activities using trajectories on the grassmannian. In: IEEE conference on computer vision and pattern recognition, pp 2435–2441
33. Veeriah V, Zhuang N, Qi G (2015) Differential recurrent neural networks for action recognition. In: IEEE international conference on computer vision, ICCV 2015, Santiago, Chile, December 7–13, 2015, pp 4041–4049

34. Vemulapalli R, Chellappa R (2016) Rolling rotations for recognizing human actions from 3d skeletal data. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 4471–4479
35. Vemulapalli R, Arrate F, Chellappa R (2014) Human action recognition by representing 3d skeletons as points in a lie group. In: IEEE conference on computer vision and pattern recognition, pp 588–595
36. Vieira AW, Nascimento ER, Oliveira GL, Liu Z, Campos MF (2014) On the improvement of human action recognition from depth map sequences using spacetime occupancy patterns. Pattern Recognit Lett 36:221–227
37. Vishwakarma S, Agrawal A (2013) A survey on activity recognition and behavior understanding in video surveillance. Vis Comput 29(10):983–1009
38. Wang H, Schmid C (2013) Action recognition with improved trajectories. In: Proceedings of the IEEE international conference on computer vision. Sydney
39. Wang J, Wu Y (2013) Learning maximum margin temporal warping for action recognition. In: Proceedings of the 2013 IEEE international conference on computer vision (ICCV), pp 2688–2695
40. Wang J, Liu Z, Chorowski J, Chen Z, Wu Y (2012) Robust 3d action recognition with random occupancy patterns. In: Proceedings of the 12th European conference on computer vision—volume part II, ECCV'12. Springer, Berlin, pp 872–885
41. Wang J, Liu Z, Wu Y, Yuan J (2012) Mining actionlet ensemble for action recognition with depth cameras. In: Proceedings of the 2012 IEEE conference on computer vision and pattern recognition, pp 1290–1297
42. Wang P, Li W, Gao Z, Tang C, Zhang J, Ogunbona P (2015) Convnets-based action recognition from depth maps through virtual cameras and pseudocoloring. In: Proceedings of the 23rd ACM international conference on multimedia, MM '15. ACM, New York, pp 1119–1122
43. Xia L, Chen CC, Aggarwal JK (2012) View invariant human action recognition using histograms of 3D joints. In: Proceedings of the CVPR workshops. IEEE, pp 20–27
44. Yang X, Tian Y (2012) Eigenjoints-based action recognition using naïve-bayes-nearest-neighbor. In: CVPR workshops. IEEE, pp 14–19
45. Zhang Z (2012) Microsoft kinect sensor and its effect. IEEE MultiMed 19(2):4–10
46. Zhang J, Marszałek M, Lazebnik S, Schmid C (2007) Local features and kernels for classification of texture and object categories: a comprehensive study. Int J Comput Vis 73:213–238
47. Zhu Y, Chen W, Guo G (2013) Fusing spatiotemporal features and joints for 3d action recognition. In: Proceedings of the 2013 IEEE conference on computer vision and pattern recognition workshops (CVPRW), pp 486–491

Ioannis Kapsouras was born in Larisa, Greece, received the Degree in Informatics and MSc in Digital Media, both from the Aristotle University of Thessaloniki. He is currently a PHD candidate at the Artificial Intelligence and Information Analysis (AIIA) laboratory of the Department of Informatics in Aristotle University of Thessaloniki. He has participated in one research project financed by European funds. Ioannis Kapsouras has co-authored 3 journal pa- pers and 5 conference papers. His current research interests include computer graphics, video processing, computer vision and pattern recognition.

Nikos Nikolaidis is Associate Professor at the Department of Informatics, Aristotle University of Thessaloniki. He has co-authored 1 book, 15 book chapters, 59 journal papers and 177 conference papers and co-edited one book and two special issues in journals. The number of citations to his work by third authors exceeds 5300 (h-index 31). He has participated into 24 research projects funded by the EU and national funds. His areas of interest/expertise include anthropocentric video analysis (human detection and tracking, activity recognition), computer vision, digital image/video processing, stereoscopic/multiview video processing/analysis,computer graphics and visualization. Dr. Nikolaidis is currently serving as associate editor for the EURASIP Journal on Image and Video Processing, Signal Processing: Image Communication and IET Image Processing. He served as Exhibits chair of IEEE ICIP 2001, Technical Program chair of IEEE IVMSP 2013 workshop and Publicity co-chair of EUSIPCO 2015. He is publicity chair of IEEE ICIP 2018. Dr Nikolaidis is an IEEE Senior Member.