The Visual Computer (2023) 39:1335–1350
https://doi.org/10.1007/s00371-022-02409-1

ORIGINAL ARTICLE

Spatio-temporal analysis and comparison of 3D videos

Simone Cammarasana[1] · Giuseppe Patanè[1]

Accepted: 8 January 2022 / Published online: 1 April 2022

Abstract
Depth sensors of low-cost acquisition devices (e.g. Microsoft Kinect, Asus Xtion) are coming into widespread use; however, 3D acquired data are generally large, heterogeneous, and complex to analyse and interpret. In this context, our overall goal is the analysis of the action of a subject in a 3D video, e.g. the action of a human or the movement of its subparts. To this end, the action classification is achieved through the analysis of the temporal variation of geometric (e.g. centroid path, volume variation, activated voxels) and kinematic (e.g. speed) properties in consecutive frames. Then, these descriptors and the corresponding histograms are used to search a frame in a 3D video and to compare 3D videos. Our approach is applied to 3D videos represented as triangle meshes or point sets, and eventually to an underlying skeleton or to markers (if available). Our tests on the MIT, Berkley, i3DPost, NTU, and DUTH data sets confirm the usefulness of the proposed approach for the analysis and comparison of 3D videos, as well as for action classification.

Keywords Shape analysis · 3D videos · 3D Video understanding · 3D Video classification and spatio-temporal reasoning · 3D Video action recognition

1 Introduction

In Computer Vision, the analysis of 2D images and videos has been studied for decades to address several problems, such as pose [4] and action [34,52] recognition, reconstruction of 3D human motion from 2D images [24] or 2D videos [11]. These methods have been specialised to indoor physical security [44], human–robot [19] and human–objects [51] interactions. In this context, depth sensors of low-cost acquisition devices (e.g. Microsoft Kinect, Asus Xtion) are coming into widespread use and offer unprecedented opportunities for a deeper understanding of the world around us and for expanding the way of interactions with digital environments. The richness of geometric data in 3D videos allows us to discriminate among different kinds of actions through geometric and kinematic descriptors. Since 3D acquired data are generally large, heterogeneous, and complex to analyse and interpret, there is an increasing need to integrate experimental observations acquired by sensors with high-level tasks, e.g. scene understanding [41] and interpretation [53].

This paper addresses the analysis, comparison, and classification of the action of a subject in a 3D video, e.g. the action of a human or the movement of its subparts. In our setting, a 3D video is represented as a sequence of consecutive frames; each frame contains a 3D subject, defined as a rigid or an articulated object. This subject is represented as a point set or a triangle mesh and is eventually associated with a skeleton, whose nodes locate relevant features and are connected with rigid joints.

Our method requires that the input subject is segmented from the background. In case of raw data, learning-based segmentation [35,55] can be applied to automatically segment the input subject and to recognise the subject of interest with respect to the background, through a semantic segmentation of the scene. Under these assumptions, the analysis of the action of a subject is achieved through the study of the *temporal variation of its kinematic properties*, which are computed in linear time with respect to the number of input points and video frames.

As a contribution with respect to previous work (Sect. 2), our focus is on the study of the changes among frames rather than on the analysis of a single frame. The proposed approach (Fig. 1) is based on descriptors that are easily and efficiently computable, are applicable to 3D point sets and markers (if any), and are enough flexible to address different tasks, such as the segmentation of subparts in terms of

✉ Simone Cammarasana
simone.cammarasana@ge.imati.cnr.it

Giuseppe Patanè
patane@ge.imati.cnr.it

[1] CNR-IMATI, Via de Marini 6, Genova, Italy

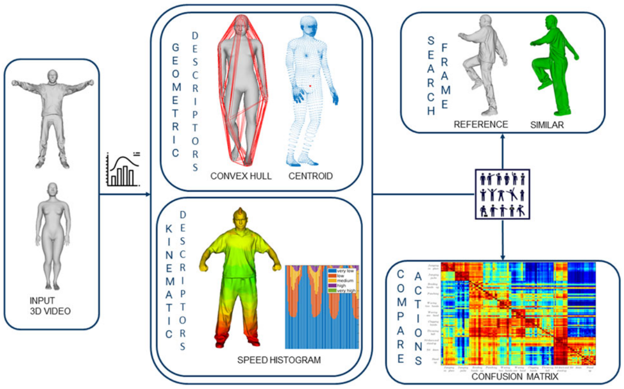


Fig. 1 Main steps of the proposed spatio-temporal analysis and comparison of 3D videos, where a set of descriptors (e.g. the histograms of the speed of the points) are used to search a frame in an action, to compare and classify actions performed by different subjects

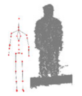
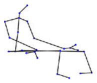
Name	Description	3D Data
MIT	Segmented point cloud	
Berkeley	Point cloud and markers	
i3DPost	Segmented point cloud	
NTU	Skeleton	
DUTH	Segmented point cloud	

Fig. 2 Data sets used for the experimental validation

homogeneous speed and the analysis of motion patterns. As *kinematic descriptors*, we introduce the pointwise, average, and angular velocities for the classification of subparts of the input subject with an analogous speed, and for the classification of an action. To improve the analysis of 3D videos, we introduce a set of *geometric descriptors*, which characterise specific behaviours of the subject (Sect. 3). These descriptors and the corresponding histograms are used to search a frame in a 3D video (Sect. 4), to compare (Sect. 5) and classify (Sect. 6) 3D actions. The combination of the proposed descriptors with markers, associated with the semantics of the underlying sub-parts (if available), allows us to accurately discriminate different actions. For our experiments, we have selected the MIT [50], Berkley [31], i3DPost [42], NTU [38], and DUTH [48] data sets, which include 3D videos of humans represented as point clouds or triangle meshes, with or without markers (Fig. 2). Finally, a video on the main tests discussed in this paper is available at the URL: https://www.dropbox.com/s/t3s1k7jk3ozrjn7/Video_PAPER.mp4?dl=0.

2 Previous work

Analysis of 3D poses In [20], the pose of a 3D human is classified by studying the space occupancy of the point cloud with respect to a cylindrical voxel grid. The identification and comparison of the postures are addressed by analysing each frame, without considering the temporal variation. In [2], *poselets* are defined to characterise human poses, considering only static data and classifying single poses through the analysis of key points. In [3], human action is classified by using silhouette-based poses as features, and the prediction of 3D human poses is based on the 2D joints' location [28] and neural networks.

Analysis of 3D actions The analysis of 3D actions is typically achieved by decomposing a complex action into atomic actions, which are then analysed through a set of descriptors that encode the geometry of the input data and their temporal evolution. In [23], the position of the body joints and a dictionary of atomic actions are applied to recognise actions in 3D videos, through the human skeleton and the geometrical and motion information of key points as descriptors. In [49], an action, represented as a sequence of 3D points, is analysed through the time-varying occupancy of a voxel grid of the scene. Human actions are detected by comparing features extracted from a time-sequence of silhouettes with examples in a training data set [36], or by extracting spatial and temporal features from skeleton sequences (e.g. displacement vector of joints), and aggregating them in a local descriptor [26]. In [40], shape descriptors are applied to compare both human shapes and clips; in [48], speed and trajectory descriptors of key points (e.g. centroid, protrusion ends) are combined to classify human actions. In [56], the combination of physical and geometric properties (e.g. acceleration, reciprocal joint orientation) of skeleton joints with a CNN is applied to characterise and classify 3D actions. In [58], a set of geometric features (e.g. joints' distance, angles between joints) extracted from skeletons are used to train a multilayer LSTM network to perform action recognition tasks. In [18], the skeleton sequences are transformed into a collection of clips (e.g. multiple images) to incorporate spatial relationships between the joints; then, the clips are used to train a deep CNN and to recognise 3D actions.

Analysis of human interaction The shape and functionality of an unknown object (e.g. a cup) are extrapolated by analysing the human interaction with the object itself, through sensors placed in the interacting surfaces. These studies (e.g. [33]) typically focus on a specific application (e.g. analysis of hand-object interaction) and are limited by the sensor position. In [21], the action recognition is performed through a mutual analysis of object detection and activity labelling;

then, the information on the specific use of an object becomes the input for the action recognition. In [60], the interaction between human characters and objects is analysed through the detection of similarities in dynamic body-object reciprocal and is based on the contact interaction bisector and surface descriptors.

Comparison of 3D actions and analysis of 3D videos For the human gait analysis [29], kinematic and spatial-temporal descriptors (e.g. joints angular speed) of the human skeleton are applied to the analysis of the walk cadence and of the stride length. To recognise 3D actions, convolutional neural networks [45] or motion context descriptors (i.e. motion and harmonic motion context) [10] are applied to a set of 3D skeletons. The optical flow [37], the 3D CNN [14], and the histogram of oriented gradients for 2D videos have been specialised to the analysis of 3D spatio-temporal data [39] and to the segmentation of 3D videos [59]. The analysis of different metrics for shape similarity in 3D video sequences [12] shows that temporal shape histograms give the best performance for different people and motions comparison. Histograms of 3D joint locations [54] are used as a compact representation of postures, which are clustered into prototypical poses of actions, and compared to recognise human actions. In [57], a semantic representation of human behaviour is based on the features extracted from the 3D video and auxiliary data. For a survey on 3D video analysis and characterisation, we refer the reader to [7,13].

3 Kinematic and geometric descriptors

We introduce the kinematic descriptors and the speed histograms, enriched with pose/volume information and geometric descriptors.

Kinematic descriptors The *speed* of the input points characterises the movement of the subject and provides a descriptor of the underlying action. Given two consecutive frames, the speed of a point $\mathbf{p}_i^s$ of a frame s is equal to the ratio between the variation of $\mathbf{p}_i^s$ from its corresponding point $\mathbf{p}_i^{s-1}$ in the previous frame and the elapsed time Δt, i.e. $\mathbf{v}_i^s = (\mathbf{p}_i^s - \mathbf{p}_i^{s-1})/\Delta t$, where Δt^{-1} is the acquisition frequency. The module v_i^s of the speed is the *speed* of a point between consecutive frames. If there is no correspondence among the points of consecutive frames, then for each point of the frame s we identify the closest point in the previous frame with a kd-tree search in $\mathcal{O}(N \log N)$ time, where N is the number of input points. This method generally works well if the acquisition frequency is high, as commonly satisfied by current acquisition systems, and guarantees the consistency of the subject geometry in consecutive frames. The *average speed* $\overline{v} = \frac{1}{T-1}\sum_{j=1}^{T-1} v_j^s$ of a point is the average of the speed at this point with respect to all the T frames and allows us to classify the subparts of the subject with a persistent speed, e.g. a static behaviour, a slow or a fast movement, and to specialise the movement analysis on his/her subparts in order to classify a persistent or an inconsistent behaviour during the actions. The subparts' segmentation of our method provides results that are analogous to learning-based methods (e.g. [27]), with the main advantage that our segmentation is performed on a single 3D video, without the need of a training data set. To further characterise a specific behaviour or movement (e.g. jumping, walking), we evaluate the *directional* and *angular speed*, by projecting the speed $\mathbf{v}_i^s$ along a given direction (e.g. the positive z-direction) or on a plane (e.g. xy plane); the projection is computed automatically, by considering the related component on the desired axis/plane. We also analyse the *variation of the angles between the velocities* of corresponding centroids, points, or markers (Fig. 3) between consecutive frames or with respect to the initial position.

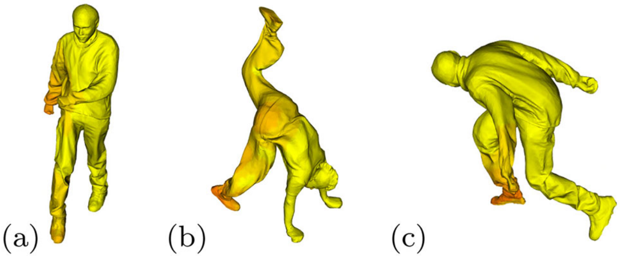


Fig. 3 Variation of the angular speed at each point for a subject performing: **a** march, **b** handstand, and **c** jumping actions. A greater angle variation is depicted as red, and the yellow colour identifies the subparts that have null angle variation between consecutive frames

Speed histogram The speed of the input points is converted into a speed histogram, which is used to segment the subject into subparts with an analogous speed, to search a frame in a 3D video (Sect. 4), and to compare (Sect. 5) and classify (Sect. 6) 3D videos. For each frame of the input 3D video, we subdivide the interval of the points' speed into n uniform intervals and group those points whose speed belongs to the same interval in the same cluster. Then, we compute the *frequencies*, i.e. the percentage of points belonging to each cluster and the average of the *speed* of the points belonging to each cluster.

Indeed, each frame i has n clusters and each cluster has a couple of descriptors $[\mathbf{frq}_i, \mathbf{spd}_i]$, where $\mathbf{frq}_i$ and $\mathbf{spd}_i$ are n-dimensional arrays of frequencies and speeds. If m markers are available (e.g. Berkley's data set), then we compute the speed of each marker, whose frequency is $1/m$. In this case, $\mathbf{frq}_i$ and $\mathbf{spd}_i$ are m-dimensional vectors and the frequencies are uniformly distributed. The speed histogram is computed in linear time with respect to the number of points or markers.

Figure 4a shows the histogram of the human performing a squat, as a slow movement with a periodic behaviour (blue curve); for each frame (x-axis), all the points are grouped in n ($n = 5$, in our experiments) clusters and we measure the number of points that are static or that are moving at a different speed. Figure 4b reports the average speed of the points belonging to each cluster (y-axis), computed for each frame (x-axis). Figure 4c shows the speed descriptor of a human throwing a ball, evaluated at 10 markers, which allows us to identify the correctness of the movement in terms of the speed of the symmetric parts (e.g. left/right arms/legs). In this case, the number of clusters of the histogram is equal to the number of physical markers that have been placed on the subject. Discretizing the bounding box of the frames' sequence with an octree, the ratio between the number of points in a voxel and the total number of points also provides the *significance of a voxel*, which is zero if the voxel is empty and low if the voxel contains outliers (Fig. 5a,b,d). The *voxel speed*, which is defined as the average of the speed of the points belonging to each voxel, is robust to noise and outliers (Fig. 5b), is useful to compute a speed distribution that is independent of the number of points.

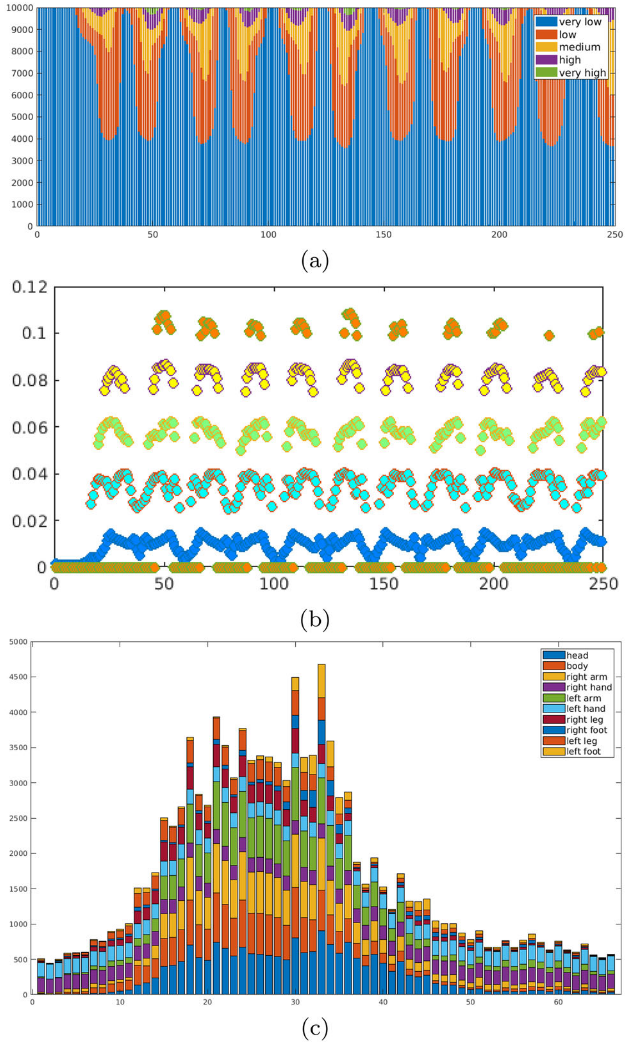


Fig. 4 For each frame (**a**, **b**, x-axis), we compute (**a**, y-axis) the histograms of the frequencies in $n = 5$ clusters (i.e. very slow, slow, medium, fast, very fast speed) and (**b**, y-axis) with the average speed of each cluster. **c** (y-axis) Speed of 10 markers in a 3D video with 66 frames (x-axis)

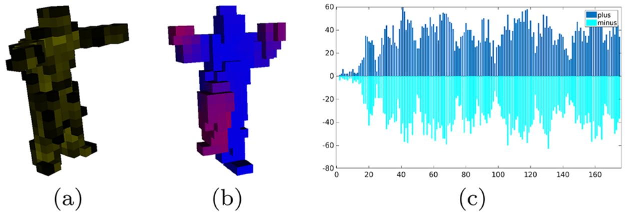


Fig. 5 **a** Activated voxel coloured according to the number of points it contains (a darker colour corresponds to a higher number of points inside the voxel) and **b** speed (blue/red=low/fast). **c** The variation of the number (y-axis) of activated/deactivated (blue/cyan) voxels identifies posture changes in each frame (x-axis)

Robust speed and geometric descriptor To address the potential impact of the quality of the input 3D videos (e.g. segmentation of objects in frames, point cloud sampling) on the action classification, we apply different strategies. In the case of clean data, already segmented from the background (e.g. MIT data set), our method correctly identifies the speed of the subparts (Fig. 6); in fact, it provides a subparts' decomposition that is visually comparable to learning-based methods [1,27]. As the main advantage, our subparts segmentation is performed on a single 3D video, without a training phase and data set. In case of noisy data and outliers (e.g. Berkeley data set) or of the irregular distribution of the veloc-

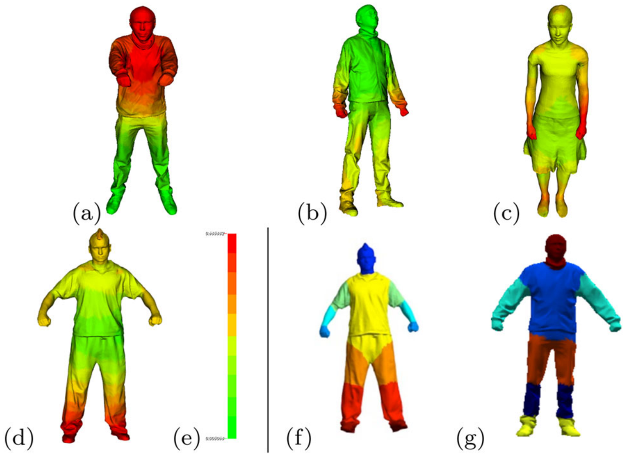


Fig. 6 Colour map of the average speed, from green (low speed) to red (high speed). In the squat action **a**, legs are static while the torso and the arms are moving faster; in the march action **b**, the hands are moving faster, the legs and the arms are moving at medium speed, while the head and the torso are basically static; in the dance action **c**, the hands are moving faster, while all the rest of the body is moving at average speed; in the handstand action **d**, the feet are moving faster and the torso is static; **e** shows the colours legend of our results. Sub-parts segmentation of state-of-the-art methods for the squat action subject: **f** [27], **g** [1]

ities (e.g. i3DPost data set), we compute the voxel speed instead of the pointwise speed, apply the voxel persistence to further reduce the noise influence on the evaluation of the descriptors, and prefer the use of markers (if available) with respect to points. Markers are generally robust to noise and particularly useful when the segmentation of the subject from the background is not available or have low quality; in this last case, we also evaluate the voxel speed and significance for a more reliable identification of the outliers and the static elements in the scene outliers.

Analysing the velocity of the voxel, we distinguish the subject from the background and/or static elements; furthermore, the significance of voxels allows us to disregard outliers and to increase the robustness of our approach to noise, through the computation of the frequency and velocities of the bins. The computation of the voxel speed allows us to reduce the impact of noisy data since the voxel speed is less affected by small shifts of the points. In contrast, a small displacement of a point affects the speed descriptor in a more significant way. To further discuss this aspect and show the higher robustness of the proposed approach on noisy 3D videos, we apply the Gaussian noise as a random Gaussian displacement along the normal direction, at each point in the frames (Fig. 7, **a–c**). Then, we compute the voxel speed (Fig. 7, **d–f**) and the histogram of the pointwise speed (Fig. 7g), for the ground-truth, and the noisy frame. The error between the ground-truth and the noisy data is computed as the norm between the speed descriptor of each element (i.e. the voxel or the bin), and normalised with the number of elements (i.e. 256 voxels or 25 bins). The error of the voxel speed is $7.4 \cdot 10^{-4}$ and $9.6 \cdot 10^{-4}$ for the low and high noise data, respectively; the error of the histogram speed is $1.3 \cdot 10^{-3}$ and $2.0 \cdot 10^{-3}$ for low and high noise data. As expected, both the errors grow as we increase the noise, and the error of the voxel speed descriptor is an order of magnitude lower than the histogram descriptor.

The analysis of 3D videos is enriched through *geometric descriptors* that characterise specific behaviours of the subject and provide a fast and approximated check of the interaction between subjects. For the *analysis of the path* of a subject, we compute the curve that best fits the path of the centroid; then, a low least-squares error between the fitted and centroid curves indicates that the subject is moving along a certain trajectory (Fig. 8a). The speed and angle variation of the centroid enriches the analysis of the behaviour of the subject (Fig. 8b). The analysis of the *temporal variation of the volume* between consecutive frames identifies changes in the space occupancy and allows us to provide basic considerations about the type of movement that the subject is performing. For instance, the analysis of the temporal variation of the size and volume of the convex hull of a man performing a squat (Fig. 8c,d) identifies that the subject is moving and that his/her movement has a periodic variation.

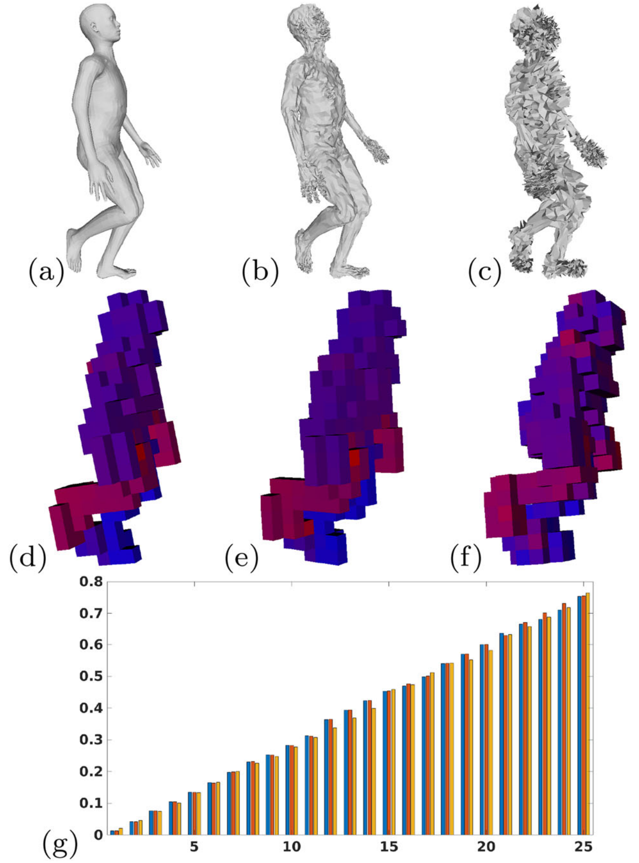


Fig. 7 **a** Ground-truth frame, **b** low-noise, **c** high-noise frame; voxel speed (256 voxels) of **d** ground-truth, **e** low, and **f** high noisy data; histogram with 25 bins ($x-$axis) of the pointwise speed descriptor ($y-$axis) **g**: ground-truth (blue), low noise (red), and high-noise (yellow) data

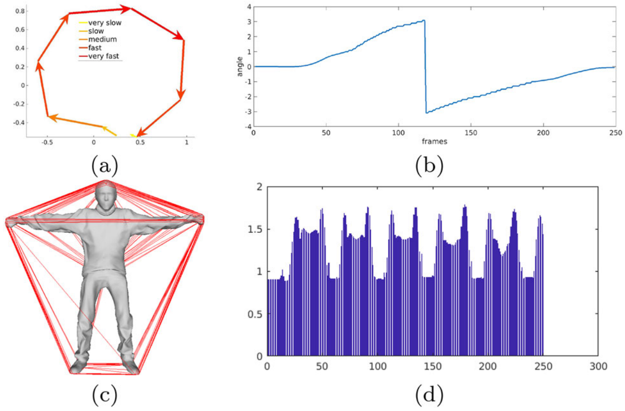


Fig. 8 For a marching subject, the temporal variation of the (x, y) coordinates of the centroid allows us to identify **a** a circular movement of the subject and his/her speed. **b** The temporal variation (x-axis) of the angle of the centroid (y-axis) with respect to the initial frame identifies a circular movement. **c** Convex hull and **d** its variation (y-axis) with respect to frames (x-axis). The bar graph shows a periodic movement every 50 frames (i.e. ≈ 2 s.)

The proposed descriptors can be enriched with the information encoded by the extremal human curve [40], physical [56] and geometric [58] properties of joints, space-time interest points detectors [22], spherical harmonics [17]. More complex descriptors generally have a higher computation time: for example, a fast algorithm for the computation of spherical harmonics [43] has a computational cost of $O(n^2 \log n)$, while our method is linear in the number of points n. For further uses of these descriptors, we refer to the following experimental tests.

4 Searching frames in 3D videos

We introduce the search of a frame in a 3D video (Sect. 4.1), the experimental results (Sect. 4.2), and the analysis of the computational cost (Sect. 4.3).

4.1 Method: frame search

We introduce temporal geometric and kinematic descriptors of a point set $\mathcal{P} := \{\mathbf{p}_i\}_{i=1}^N$ in a 3D video. Firstly, we perform a frame-video co-registration and compare their speed histograms. For *frame-video co-registration*, let us consider a frame i of a 3D video $\mathcal{Q}$ and another 3D video $\mathcal{S}$. If the frame i in $\mathcal{Q}$ does not refer to the same subject in $\mathcal{S}$, then we "align" the subject in i with the subject in $\mathcal{S}$. An initial alignment of the subjects in $\mathcal{Q}$ and $\mathcal{S}$ is performed through the principal component analysis [15], and for the co-registration we apply the RANdom SAmple Consensus (RANSAC) [5] algorithm, which computes the new sampling (i.e. the co-registered one) through an iterative approach: after an initial selection of the new samples and the computation of the fitting between the proposed sampling and the target points, the inliers are separated from the outliers, thus providing an output sampling that considers only the meaningful points (i.e. the inliers). For *frames' comparison*, searching a frame i in the video $\mathcal{S}$ is equivalent to comparing this frame with all the frames of $\mathcal{S}$. To compare a frame $i \in \mathcal{Q}$, with a frame $j \in \mathcal{S}$, we compute the speed histograms $(\mathbf{frq}_i, \mathbf{spd}_i)$ and $(\mathbf{frq}_j, \mathbf{spd}_j)$ of i and j, respectively. Then, the *distance between the histograms* of two input frames is defined as $d(i, j) = \|\mathbf{frq}_i \odot \mathbf{spd}_i - \mathbf{frq}_j \odot \mathbf{spd}_j\|_2$, i.e. the Euclidean distance of the difference of the weighted speed histograms of the frames (i, j). Here, $\mathbf{a} \odot \mathbf{b}$ is the vector whose entries are the pointwise product between the corresponding entries of $\mathbf{a} := (a(j))_j$ and $\mathbf{b} := (b(j))_j$, i.e. $(\mathbf{a} \odot \mathbf{b})(j) = a(j)b(j)$.

Given the frame i in $\mathcal{Q}$, we evaluate the distances for each frame in S, i.e. $\mathbf{d} := (d(i, j))_{j \in S}$; similar frames are computed as the minimum values of $\mathbf{d}$ with a local search algorithm. The lower is the distance, the greater is the similarity between the two frames. The speed histogram is independent of the geometry of the poses and identifies the speed similarity of the movement of two subjects, which can be performed in a different posture and/or by subjects with a different shape. Indeed, a static shape descriptor is applied to identify a geometric similarity (if any) between static positions of the input subject; in our experiments, we select the 3D shape context [6], which is invariant with respect to rescaling, rotation, translation and is robust with respect to geometric sampling. In particular, the kinematic descriptors (e.g. speed histogram) compare all the frames of the 3D video and detect the most similar frames to the reference one; when two or more frames are both very close to the reference one, we apply the shape context descriptor, which allows us to compare the static poses, and improve the discrimination between different behaviours of the subject. Since the shape context is computed at each point, a small set of *key-points* is selected in order to reduce its computational cost, without affecting its discriminative power. Key points are not necessarily the input markers (if available) and must be stable to surface acquisition and discretisation, distinctive, and easily detectable. Among the methods for the selection of key points, we have selected the *intrinsic shape signature* (ISS) [61], which provides good results in terms of points' stability and distribution.

4.2 Experimental results

MIT data set Figure 9a shows the detected poses for a human performing a squat when he/she is in the starting pose, i.e. we are searching the pose on the left (belonging to a subject performing the squat action) in the video on the right (where the squat action is performed by a different subject). The poses in green and light green are those ones detected as more similar to the reference pose through the comparison of the corresponding speed histograms. Both poses are correct in terms of the speed of their subparts; in fact, the input subjects are moving the arms and the knees at a similar speed, while the feet are static. Furthermore, Table 1 (row 1–2) shows the histogram-based distances between the reference frame and the compared frames (Fig. 9). The green pose (i.e. frame 1) is the most similar to the reference one; then, the light green pose (i.e. frame 50) is more similar to the reference one than the pose referring to frame 8 (which is 7 frames distant from the green pose). We underline that the histogram-based distance between the reference frame and the first frame has a low value; then, the distance increases as the movement of the subject becomes more rapid and finally decreases to a low value, as soon as the final position that is similar to the initial one, in terms of speed of the subparts.

We enrich this analysis with the 3D shape context [6], which is able to disambiguate the subject in light green (not geometrically similar to the reference one) from the subject in green. In fact, the 3D shape context distance between the light green one and the reference pose is 1.00, and the distance

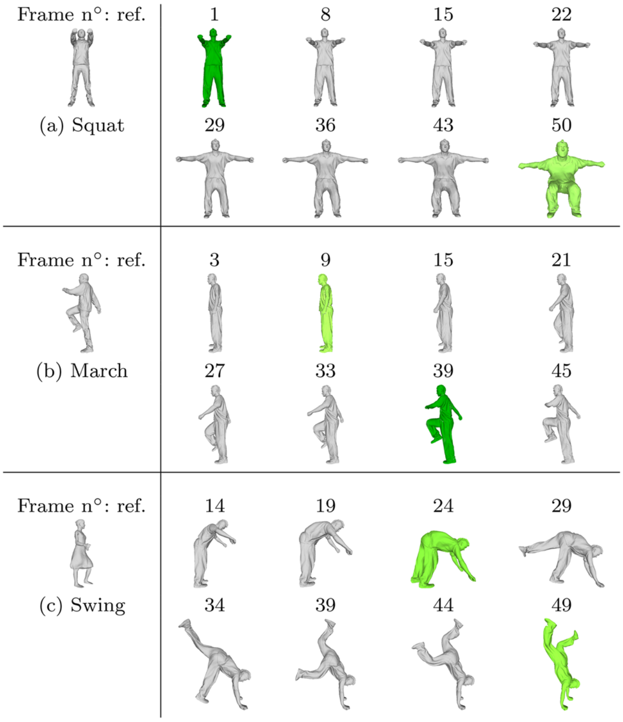


Fig. 9 The actions on the left (ref.) are searched in a 3D video (i.e. **a** squat, **b** march, **c** swing step) performed by a different subject. In **a**,**b**, the speed histogram detects two poses (green, light green) similar to the reference one in terms of speed. Among these two poses, the 3D shape context identifies the more similar pose in terms of static geometry (green). **c** The swing step movement (left) is searched in the 3D video of the handstand action; the speed histogram identifies two poses similar in terms of speed; and the shape context is not useful as the subjects have a high shape difference. We also report the frame number in the video above each 3D model

Table 1 With reference to Fig. 9, we report the histogram-based distance between the reference action and the reported frames of the three videos: squat, march, and swing

Frame	1	8	15	22	29	36	43	50
Squat	**0.11**	0.21	0.32	0.40	0.39	0.36	0.24	**0.12**
Frame	3	9	15	21	27	33	39	45
March	0.24	**0.16**	0.22	0.26	0.26	0.19	**0.15**	0.19
Frame	14	19	24	29	34	39	44	49
Swing	0.36	0.29	**0.20**	0.27	0.39	0.32	0.26	**0.21**

of the pose in green with respect to the reference one is 0.78. Among all the frames of the input 3D video, the object in green is the most similar to the reference frame (a) in terms of speed and geometry.

Figure 9b shows an analogous analysis for the human marching pose; the reference one (left) has been searched in the video on the right, where the marching action is performed by two different subjects. The proposed method identifies two similar poses (green and light green) in terms of speed; in fact, these two poses are moving the left arm and the right leg at a similar speed. The speed histogram of the march action is more complex than the previous (e.g. squat) action; and in fact, more body sub-parts (i.e. points of the point cloud) are moving simultaneously. Table 1 (row 3–4) shows the histogram-based distances for the march action. In this case, the green pose is also the most similar to the reference one (i.e. frame 39, 0.15 distance), and the light green pose has a slightly higher distance (i.e. frame 9, 0.16 distance). Through the 3D shape context, we compare the poses in green to the reference one in terms of geometry; the corresponding distance between the pose in light green and the reference pose is 1.00, and the distance of the pose in green from the reference pose is 0.84. Indeed, the pose in green is the most similar to the reference pose in terms of speed and geometry. In Fig. 9c, two poses (green) of the handstand action are classified as similar to the swing step frame (on the left) through the speed histogram method. Table 1 (rows 5–6) shows that the histogram-based distance between the two green poses and the reference one is higher with respect to squat and march actions; in fact, these two poses are different from the reference one, both in terms of shape and speed descriptors. In this case, the shape of the subjects is different, and the 3D shape context is not able to identify similar poses; in fact, the shape context distance of the first pose in green from the reference one is 1.00, and the distance of the second pose in green from the reference one is 0.96.

The nonlinear behaviour of the results (Table 1) mainly depends on the periodicity of the movement, and on the acceleration/deceleration of the human body parts, which may be nonlinear. We also underline that the histogram distance descriptor is a measure of dynamic similarity, without any assumption on the shape/pose of the subject. For this reason, two poses can result different in terms of geometric shapes, yet result similar in terms of velocities. Finally, we note that the most similar frame of the squat action has a distance value of 0.11, while the most similar frame of the swing action has a distance value of 0.20. Despite the histogram descriptor does not use any knowledge on the geometry, it is able to correctly identify the high similarity of the squat action with the reference frame.

i3DPost data set We perform an analogous test on the i3DPost data set (Fig. 10). In (a), our method correctly identifies two poses of the walking woman, which are similar to the reference pose of the walking man. In (b), we recognise two poses as similar to the bent person reference, in terms of speed descriptor. Table 2 shows the histogram-based distances, between the reference frame and the compared frames (Fig. 10). Also in these examples, the green actions are more similar to the reference one; for example, frame 50 of the bend action has a distance of 16 with respect to the reference pose, and this value is the lowest among the poses of the bend video. Then, the 3D shape context identifies the exact pose; in fact, the distance between the pose in light green and the ref-

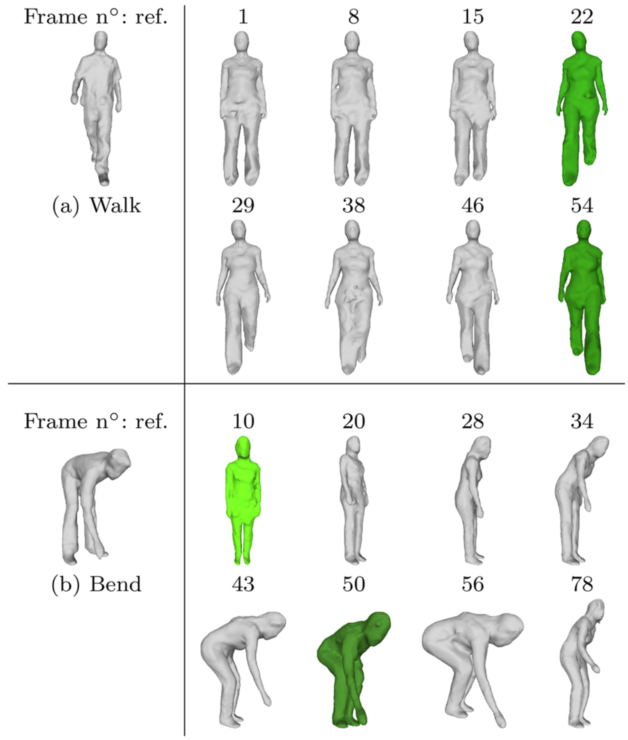


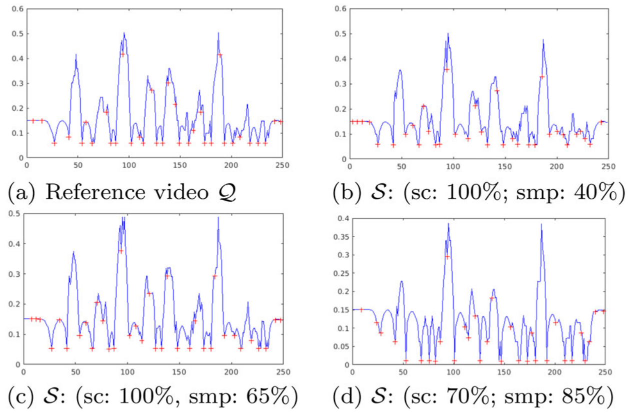
Fig. 10 Each action on the left is searched in a 3D video. The speed histogram **a** identifies two poses similar in terms of speed and **b** detects two poses (green, light green) similar to the reference one. Then, the 3D shape context identifies the more similar pose in terms of geometry (green). We also report the frame number in the video above each 3D model

Table 2 With reference to Fig. 10, we report the histogram-based distance between the reference action and the reported frames of the two videos: walk and bend

Frame	1	8	15	22	29	38	46	54
Walk	0.31	0.26	0.20	**0.15**	0.22	0.24	0.19	**0.15**
Frame	10	20	28	34	43	50	56	78
Bend	0.19	0.24	0.28	0.23	0.19	**0.16**	0.19	0.24

erence one is 1.00, while the distance of the green pose with respect to the reference one is 0.72. Our method achieves good results both with intra-person and inter-person frame search, and the results are comparable with state-of-the-art methods [9,12] that are not based on machine learning.

The distance between the speed histograms of two frames (Sect. 4) is *invariant with respect to rotation and translation* of the input data. Scale changes have a partial influence; in fact, the frequencies remain unchanged but the speed of the points varies. Indeed, the scale independence of the speed descriptor is achieved by uniformly rescaling the input data. Data downsampling affects the correspondences among points; in this case, the method remains the same and the



(a) Reference video $\mathcal{Q}$ (b) $\mathcal{S}$: (sc: 100%; smp: 40%) (c) $\mathcal{S}$: (sc: 100%, smp: 65%) (d) $\mathcal{S}$: (sc: 70%; smp: 85%)

Fig. 11 Robustness of the speed histogram of the march action $\mathcal{Q}$ searched in a 3D video $\mathcal{S}$ with a different marching subject and its minima to *scaling*-sc., *downsampling*-smp

closest points are detected with a kd-tree search. To test the *robustness of the speed histogram*, we evaluate its accuracy on several videos, which are achieved by applying a roto-translation, downsampling, and rescaling to a reference 3D video. According to the results on the i3DPost data set (Fig. 11), the distance functions have similar behaviour for all the transformed 3D videos and almost the same minima are correctly identified.

4.3 Computational cost and execution time

We compute the velocity descriptor for each point of the two videos, then we compute the histograms, and finally, we compare the histograms and find the requested frame. The computational cost is $O(n \cdot fr_2)$, where n is the number of points of each frame, and fr_2 is the number of frames of the video where we search the reference frame. Here, the number of points affects the computational cost for computing the histogram of each frame. For example, given a reference frame of $n_1 = 10K$ points, and 3D video of 200 frames (each of $n_2 = 10K$ points), and given $t \approx 10^{-7}s$ as the time for computing the velocity descriptor for a single point, and $h = 0.02s$ as the time for computing the histogram of a point cloud, with 25 bins, the execution time is computed as:

- Pointwise velocity: $t \cdot n_1 + t \cdot n_2 \cdot fr_2 = 0.2s$;
- Histogram of the reference frame: h;
- Histogram of the 3D video: $h \cdot fr_2 = 0.4$s;
- Histogram comparison for frame search: 0.007s;

where the total execution time for the frame search task is about 0.6 s. Tests have been performed with MATLAB R2020a, on a workstation with 2 Intel i9-9900KF CPUs (3.60 GHz), and 32 GB RAM.

5 Comparing 3D actions

We discuss the comparison of 3D actions (Sect. 5.1), the experimental results (Sect. 5.2), and the analysis of the computational cost (Sect. 5.3). For these tests, our method does not require any preliminary assumption on the type of action; the labels of the data (e.g. jumping action) are used only to verify that similar actions have similar histograms, and the comparison is performed only through the proposed descriptors.

5.1 Method: actions comparison

To compare actions in 3D videos, we extend the previous method (Sect. 4.1) to a block of consecutive frames instead of considering only two frames. We notice that we can have a certain variability in the action that we are searching; if the two actions have a different length, then the frames' blocks and the corresponding speed histograms have a different size. Furthermore, the subjects can perform the same action at the same time length, but with a certain discrepancy in the two 3D videos. Indeed, we first align the speed histograms of frames' blocks, which are then compared through a similarity distance.

Alignment of speed histograms Given two histograms represented as vectors $\mathbf{x}_1 \in \mathbb{R}^m$, $\mathbf{x}_2 \in \mathbb{R}^n, m < n$ of different lengths, we find a set of interpolated and evenly distributed values that complete the smaller vector with $(n - m)$ entries. Firstly, we compute the unique couple $[(m_1, a), m_1)]$ such that $m_1 \leq m$ and $a \cdot m_1 + (a + 1) \cdot (m - m_1) = n$. Then, we fulfil m_1 intervals with a values and the remaining intervals with $a + 1$ values, so that the size of the two vectors is the same. The parameter a represents the number of new elements that are inserted in the vector of lower size $\mathbf{x}_1$, and that are linearly distributed between the boundary values of each subinterval of this vector.

Frames' blocks comparison To compare two blocks of T frames, we define their distance as a matrix $\mathbf{D} \in \mathbb{R}^{T \times T}$, where the entry (i, j) is the distance between the frame i and the frame j. Firstly, we consider the vector $\mathbf{d} := (d(i))_i$, whose i-th entry is the minimum of the i-th row of $\mathbf{D}$ in a neighbourhood of the entry (i, i); i.e. the minimum distance $d_{\min}(i) = \min_k(D(i, i + k))$, $-h < k < h$, between the frame i and a set of $2h$ frames temporally close to i, with $h = 5$ in our experiments. Then, the distance between these two sequences is computed as $d_F = \|\mathbf{d}\|_2/(T - 1)$. Our approach computes the distance of two actions by considering a window that takes into account the shifts between them; our approach has comparable results with respect to the dynamic time-warping method [30], while the Procrustes' analysis [8] method gives worse results, as it does not involve the temporal consistency among frames.

To compare two arbitrary frames' blocks, we identify frames with a static subject (i.e. null speed), which are considered as the instants when the action starts and ends. Then, we select the portions of each histogram that correspond to the selected frames' blocks, represented as two matrices $\mathbf{A} \in \mathbb{R}^{f_A \times n}$ and $\mathbf{B} \in \mathbb{R}^{f_B \times n}$ with $f_A < f_B$, where f is the number of frames and n is the number of clusters. To compute the distance between $\mathbf{A}$ and $\mathbf{B}$, we transform each column $A(:, j)$ in a new vector $A^*(:, j)$ with the same length as $B(:, j)$, by applying the vector alignment introduced in Sect. 5.1. Finally, we compute the distance $d_F(\mathbf{A}^*, \mathbf{B})$ between $\mathbf{A}^*$ and $\mathbf{B}$, and we expect that the distance of the same actions, even performed by different subjects, is lower than the distance between different actions.

5.2 Experimental results

MIT data set Table 3 reports the results of the comparison of four movements, i.e. a squat movement performed by five different persons, a squat movement from a stand-up position to a crouched position, a squat movement from the crouched to the stand-up position, a walking step, and a swing dance step. Our method has good performances in terms of the identification of analogous actions (even if performed by different persons). For example, the squat actions from up to down performed by two different subjects (ID 1 and ID 3, respectively) have a distance of 0.30; the same squat action compared with a *march step* (ID 6) has a distance of 1.00. The *march step* action of subject C (ID 5) has a distance of 0.21 (i.e. it is very similar) with respect to the same action performed by a different subject (ID 6); the *march step* has a higher distance with respect to the *swing step* (ID 7, distance value = 1.00).

Table 3 Comparison among 7 actions (*ID*) and corresponding distances (Sect. 4.1)

ID	Subject	Movement
1	A	squat–up to down
2	A	squat–down to up
3	B	squat–up to down
4	B	squat–down to up
5	C	march–step
6	D	march–step
7	E	swing–dance step

ID	1	2	3	4	5	6	7
1	–	0.41	**0.30**	0.51	0.92	1.00	0.57
2	0.49	–	0.36	**0.26**	0.81	0.69	1.00
5	0.59	0.39	0.42	0.46	–	**0.21**	1.00

Similar actions have a lower distance (bold), even if performed by different subjects. Parameters: $h = 10$ in $d_{\min}$, 25 clusters

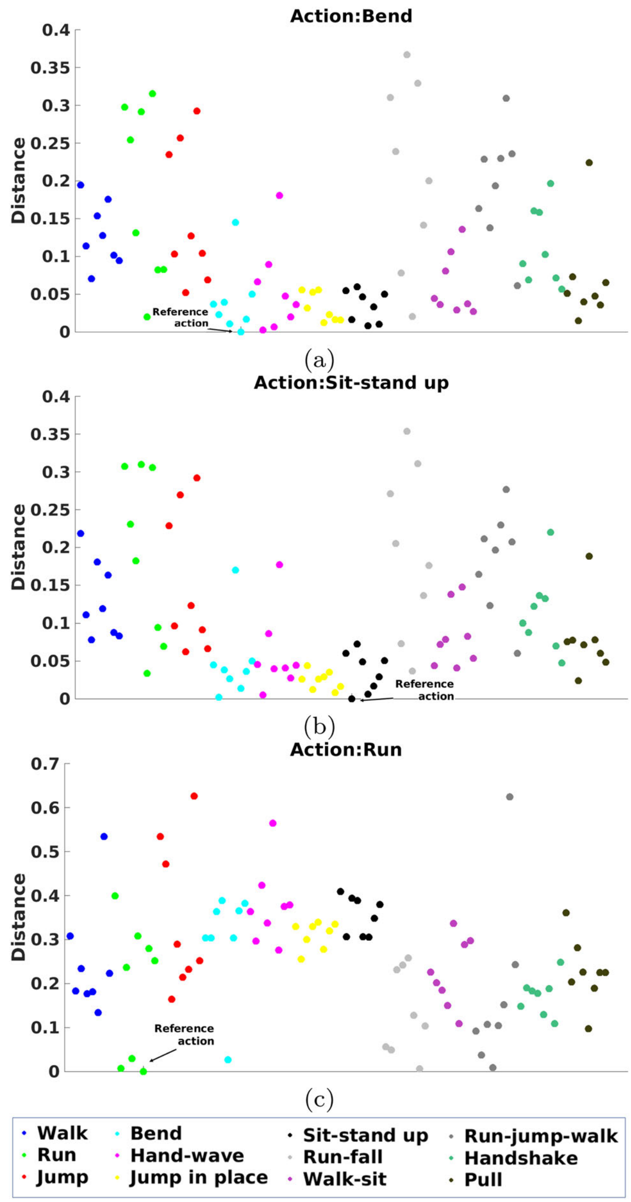


Fig. 12 Distance between a reference action (i.e. an instance of **a** bend, **b** stand-up, **c** run) and all the other actions of different subjects

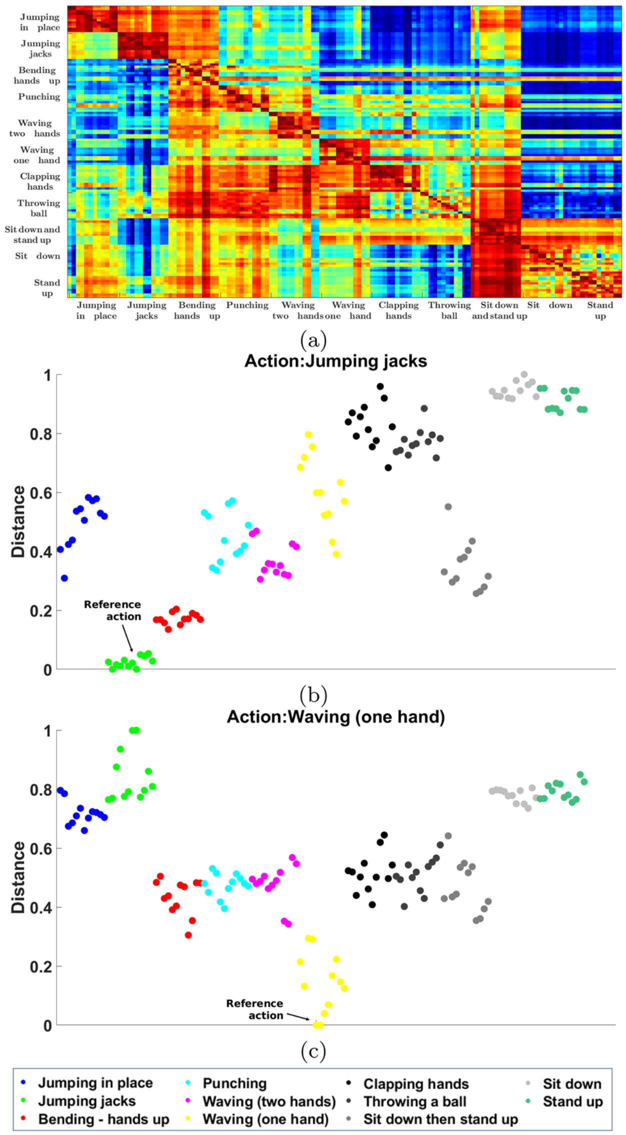


Fig. 13 **a** Confusion matrix of Berkeley's actions: similarity between each pair of actions varies from 0 (blue colour) to 1 (red colour). **b**,**c** Distance between a reference action (i.e. an instance of **b** jumping jacks, **c** waving one-hand) and all the other actions of different subjects

i3DPost data set We compute the distances between pairs of actions and cluster the results according to their action' classes. Figure 12a–c shows the distance between a reference action (i.e. (a) bend, (b) stand-up, (c) run) and all the other actions; the distances are well clustered, and the actions of the same cluster have a low distance with respect to the reference action, thus showing us that the method is able to recognise similar actions of the same class. However, a few false-positive results are present in the actions comparison, i.e. different actions with a low distance with the reference one. For example, in (c) our method correctly classifies the *run* actions performed by different subjects as similar to the reference (run) action; however, a low number of false positives (e.g. *run-fall*, *run-jump-walk*) are detected, due to the similarity of these activities to the reference one. There is only one case where the bend action is similar to the run action (i.e. the reference one): this result depends on the way this instance of the bend action is performed, which is quite different with respect to all the other instances of the same action. In fact, analysing the results of Fig. 12a, b, this instance of the bend action is typically out of the cluster of the same action performed by different persons.

Berkeley's data set We compute the distances between pairs of actions and cluster the results according to their action' classes. Figure 13a shows the confusion matrix for the similarity (i.e. the inverse of the computed distance) between

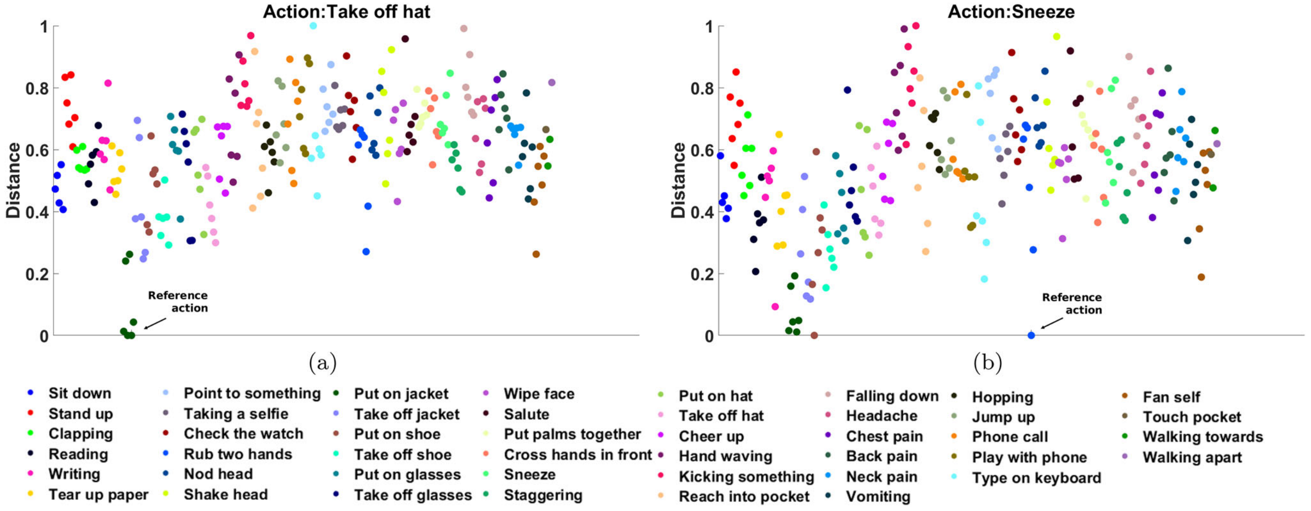


Fig. 14 Distance between a reference action (i.e. an instance of **a** taking off hat, **b** sneezing) and all the other actions of different persons

each pair of actions. Our method correctly identifies similar actions; the similarity between actions belonging to the same class (i.e. the diagonal blocks of the matrix) is always very high, and the similarity among different classes is generally low. Some blocks that are out of the diagonal have high similarity (e.g. *sit-down*, *stand-up*, *sit-down, and stand-up* actions), mainly due to the similarity of the performed action among different classes. The *throwing-ball* action is the only case where the block with the highest values is detected in a different class (i.e. *bending-hands-up*), as the two actions are performed in a similar way in terms of kinematic parameters. In Fig. 13b,c, the actions with lower distance are those ones belonging to the same class. The distance between the reference action and different classes is well clustered, thus showing us that actions of the same class are performed in a similar way and allowing us to better discriminate between different types of actions.

NTU's data set We perform the same tests on the NTU's data set, where the speed of the body joints is more irregular, as the skeleton is extracted by the Kinect, and the markers are not present. Depending on the type of action, we get a higher variability of the results. In Fig. 14a, the reference action (i.e. *put hat*) has a low distance with respect to actions of the same class. In Fig. 14b, the reference action (i.e. *sneeze*) is similar to actions belonging to other classes, due to the irregularity of the speed of some body parts, thus leading to a wrong comparison among different actions. This result shows that our algorithm generally requires a smooth motion of the skeleton. The reduced clustering effect in Fig. 14 depends on the higher irregularity of the 3D videos of the NTU data set, with respect to the Berkeley data set (i.e. Fig. 12); since our descriptors are sensitive to irregular variation of the speed of the points, the characterisation of the actions is less precise.

DUTH's data set According to the confusion matrix of DUTH's 3D videos (Fig. 15), the similarity between actions of the same class (i.e. the diagonal blocks of the matrix) is always very high, except for the *run* action that has high variability in terms of the behaviour of the subjects. A high similarity of some non-diagonal blocks (e.g. *walk-left* and *walk-right*, or *jump* and *jump-forward* actions) is due mainly to similar actions among different classes. In Fig. 15b, c, *walk-right* and *jump-forward* actions are correctly classified as similar to the same actions performed by different subjects and to other analogous actions. Figure 16 shows a detail of the actions' comparison between two different actions (i.e. jumping forward and washing window) performed by three different persons: tall and thin man, small woman, tall and robust woman. Our method correctly compares the actions, independently of the subjects' characteristics; in fact, the similarity between the same actions is high, even if performed by different persons. Furthermore, the "washing window" actions have smaller distances among themselves, with respect to the "jumping forward" actions; indeed, this action is performed more similar among the different subjects.

DUTH/i3DPost cross-comparison Cross-comparing the DUTH and i3DPost actions (Fig. 17), most of the actions are correctly recognised as similar (e.g. DUTH *walk* and i3DPost *walk*) with a few false positives (e.g. DUTH *jump* and the i3DPost *run*).

5.3 Computational cost and execution time

We evaluate the velocity of each point of the two videos, then we compute the histograms, and finally, we compare the

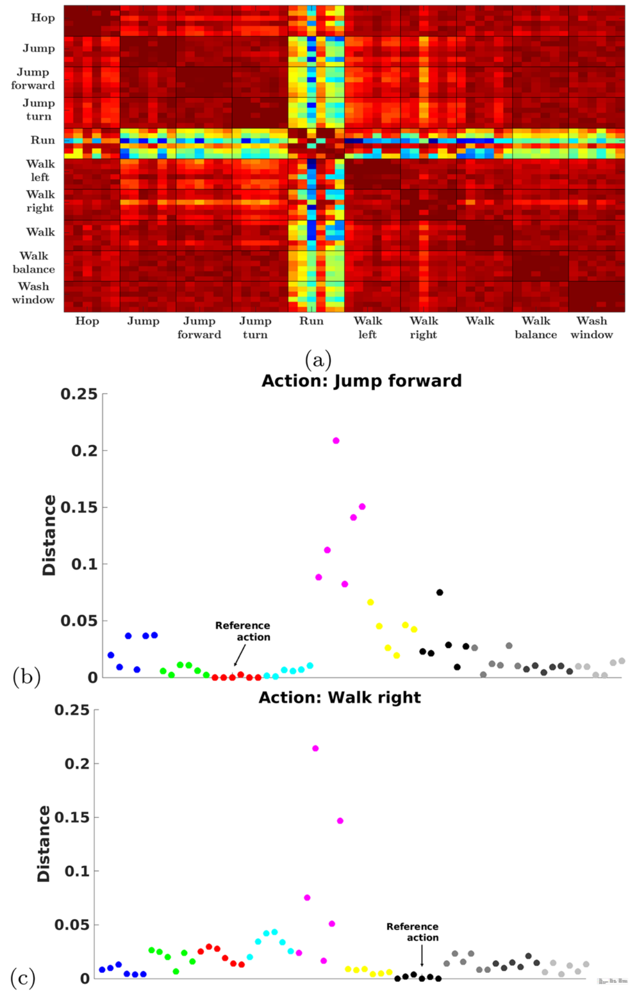


Fig. 15 **a** Confusion matrix of DUTH's actions: similarity between each pair of actions varies from 0 (blue colour) to 1 (red colour). **b**, **c** Distance between a reference action (i.e. an instance of **b** jump forward, **c** walk right) and all the other actions of different subjects

histograms to compare two actions. The computational cost is $O(n \cdot (fr_1 + fr_2))$, where n is the number of points of the point cloud of the video, and $fr_{1,2}$ is the number of frames of the two videos, respectively. As a matter of example, given two videos of 150 and 130 frames of 8K and 10K points, respectively, the execution time is:

- Velocity computation: $t \cdot n_1 \cdot fr_1 + t \cdot n_2 \cdot fr_2 = 0.3s$;
- Histogram of the first video: $h_1 \cdot fr_1 = 0.24s$;
- Histogram of the second video: $h_2 \cdot fr_2 = 0.26s$;
- Compare histograms and actions: $0.09s$;

where h is the time for computing the histogram of a single point cloud. Our method takes less than 1 second to compare two different videos.

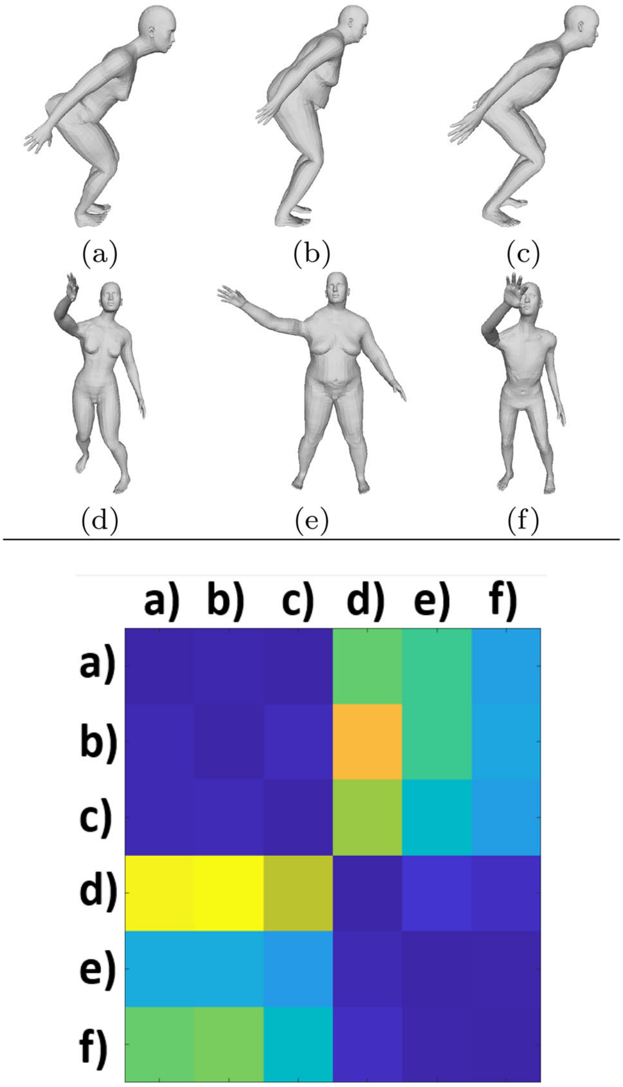


Fig. 16 With reference to Fig. 15, we report the comparison between the jumping forward action **a**, **b**, **c** and the washing action **d**, **e**, **f**, performed by three persons of different body sizes

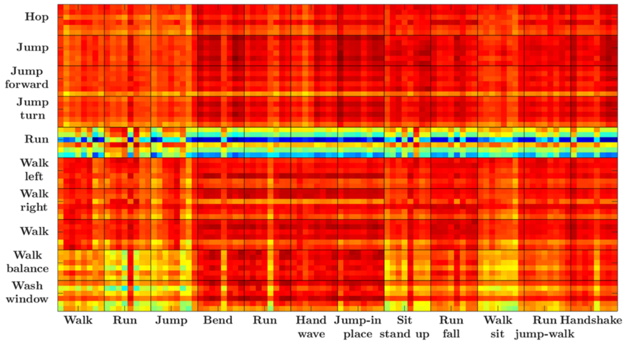


Fig. 17 Confusion matrix of the actions of two different data sets: DUTH (y-axis) and i3DPost (x-axis)

6 Classifying 3D actions

We introduce the comparison of 3D actions (Sect. 6.1) and the experimental results (Sect. 6.2).

6.1 Method: actions classification

Even though our focus is the comparison of 3D videos, the proposed descriptors and the corresponding distances are general enough to classify complex actions. Given a set of actions $\mathcal{C} = \{C_i\}_{i=1}^{n}$ (e.g. walk, run, move hands) classified into n classes, for each action $i \in \mathcal{C}$ we compute its average distance with the actions in each class of $\mathcal{C}$, i.e. $d_{AC}(i, \mathcal{C}_k) = \sum_{j \in \mathcal{C}_k} d_{ij}/\#\mathcal{C}_k, k = 1, \dots, n$, where d_{ij} is the distance between actions i and j, computed as described in Sect. 4.1. Then, each action is classified according to the category with the lower distance, i.e. $c = \arg\min_k d_{AC}(i, \mathcal{C}_k)$. We define the accuracy of a class as the average of the accuracies of the actions that belong to it.

6.2 Experimental results

Berkeley's data set For each action, we identify its category as the corresponding label of the minimum distances d_{AC}. Table 4 shows a comparison of our method with state-of-the-art methods (both learning and non-learning based), in terms of classification accuracy. Our method correctly classifies 94.2% of the actions; this result is comparable with state-of-the-art methods, although they have a slightly better accuracy; in particular, the learning-based method (i.e. [25]) reaches a 100% accuracy. Considering the classification accuracy of each category (Table 5), some actions (e.g. jumping, waving) are recognised with a 100% accuracy or higher than 90% (e.g. bending, clapping hands), while others (e.g. punching, stand-up) have an accuracy around 83%.

When markers are available, our method has performance in line with previous work, without requiring any learning phase; this result is a significant advantage when big data are not available for a training phase. Furthermore, the proposed descriptors are very simple yet effective, easy to compute, and computationally cost-effective. As an additional advantage, our method allows us to break down the descriptors at the human sub-parts level, e.g. to analyse the similarity of the movements of the upper or lower limbs. In contrast, learning-based methods classify the action, without any additional detail in terms of granularity of human body parts.

Table 4 Accuracy (acc.) computation on Berkeley's data set

Method	Acc. (%)	Method	Acc. (%)
Ofli et. al [32]	95.4	Vantigodi et. al [46]	96.1
Kapsouras et. al [16]	98.2	Liu et. al [25]	100

Our method: 94.2%

Table 5 Accuracy (acc.) of our descriptors on Berkeley's data set.

Category	Acc.	Category	Acc.
Jumping in place	100%	Waving one hand	100%
Jumping jacks	100%	Clapping hands	91.7%
Bending hands	91.7%	Sit-down and stand-up	100%
Punching	83.3%	Sit-down	91.7%
Waving two hands	100%	Stand-up	83.3%

Average 94.2%

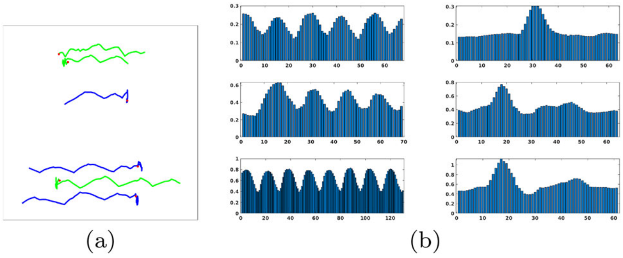


Fig. 18 DUTH data set: analysis of *walk-left* (blue) and *walk-right* (green) actions; **b** bounding box analysis of *walk* (left) and jump *right* actions, with frames on the x-axis and descriptor on the y-axis

NTU's & i3DPost data sets To classify each action of the NTU data set, we compute the minimum of d_{AC} and compare the classification results with state-of-the-art methods, i.e. the Lie group method [47] with an accuracy of 50.1% and the Deep LSTM method [38] with an accuracy of 60.7%. Our method generally provides good results for the classification of smooth actions, which are in line with state-of-the-art methods; for instance, the *take-off a shoe* action has a classification accuracy of 60%, and the *jump-up* action has an accuracy of 50%. In contrast, some actions (e.g. take a selfie, type on the keyboard) of the NTU data set have an irregular profile of joint velocities, with a negative impact on our descriptors and on the resulting classification, as we assume smooth movements. The higher number of distinct classes of similar actions (e.g. writing, typing on the keyboard) also reduces the capability of our method in differentiating them. For these classes with flickered 3D videos, the classification accuracy is lower than state-of-the-art methods (i.e. around 20%). Finally, our method has some limits in the classification accuracy with the *i3DPost data set*, since the absence of the labelled markers does not allow us to correctly identify the right class of action, with an average classification accuracy around 40%.

DUTH data set Even though geometric and kinematic descriptors are simple and intuitive, they provide a preliminary information on the subject behaviour; for instance, the centroid analysis (Fig. 18a) discriminates between *walk-*

left and *walk-right* actions and the volume/speed variation (Fig. 18b) is useful to roughly discriminate between two different actions, which involve a strong posture change (e.g. *jump*) or have a regular behaviour (e.g. *walk*).

7 Conclusions and future work

We have presented a set of geometric and kinematic descriptors for the characterisation of 3D videos. These descriptors allow us to analyse and describe the behaviour of a subject, segment its subparts, and compare poses and actions, without assuming similarity in the geometry of the subjects. Our underlying assumption is that the input subject is segmented from the background; this condition requires a pre-processing of the raw point cloud (e.g. identification of static elements through voxel analysis), while it is guaranteed with markers data.

As main pros, the proposed descriptors are easy to compute and manipulate, and they allow us to compare different actions without the need of a large data set and a training phase, through a fully unsupervised method; in addition, they achieve good actions' classification and recognition results, with a low computational cost. Furthermore, our method can be applied both to point clouds and markers, and it allows us to compare actions of the different data sets, e.g. for the monitoring of rehabilitation activities. As main cons, learning-based methods have better performance than ours, in terms of classification accuracy, due to the possibility to manage more complex descriptors. Furthermore, our method shows some limits in the classification of actions with an irregular speed.

As future work, we plan to apply our approach to different applications, such as the analysis of multiple subjects and their mutual interactions, e.g. humans in a station or human–robot interaction in an industrial environment in order to evaluate how much it is deviating from a predefined path. Another application is the rehabilitation by evaluating the quality of a postural re-education exercise, compared with a reference one that has been precomputed in a training session.

Acknowledgements We thank the Reviewers for their comments, which helped us to improve the paper content, presentation, and experiments.

References

1. Aubry, M., Schlickewei, U., Cremers, D.: Pose-consistent 3d shape segmentation based on a quantum mechanical feature descriptor. In: Joint Pattern Recognition Symposium, pp. 122–131. Springer (2011)
2. Bourdev, L., Malik, J.: Poselets: Body part detectors trained using 3D human pose annotations. In: Conference on Computer Vision, pp. 1365–1372. IEEE (2009)
3. Chaaraoui, A.A., Climent-Pérez, P., Flórez-Revuelta, F.: An efficient approach for multi-view human action recognition based on bag-of-key-poses. In: Workshop on Human Behavior Understanding, pp. 29–40. Springer (2012)
4. Eichner, M., Marin-Jimenez, M., Zisserman, A., Ferrari, V.: 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. Int. J. Computer Vis. **99**(2), 190–214 (2012)
5. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
6. Frome, A., Huber, D., Kolluri, R., Bülow, T., Malik, J.: Recognizing objects in range data using regional point descriptors. In: European Conference on computer vision, pp. 224–237. Springer (2004)
7. Georgiou, T., Liu, Y., Chen, W., Lew, M.: A survey of traditional and deep learning-based feature descriptors for high dimensional data in computer vision. Int. J. Multimed. Inf. Retrieval **9**(3), 135–170 (2020)
8. Gower, J.C.: Generalized procrustes analysis. Psychometrika **40**(1), 33–51 (1975)
9. Holte, M.B., Chakraborty, B., Gonzalez, J., Moeslund, T.B.: A local 3-d motion descriptor for multi-view human action recognition from 4-d spatio-temporal interest points. J. Selected Topics Signal Process. **6**(5), 553–565 (2012)
10. Holte, M.B., Moeslund, T.B., Nikolaidis, N., Pitas, I.: Conference on 3D human action recognition for multi-view camera systems. In: 3D Imaging, Modeling, Processing, Visualization and Transmission, pp. 342–349. IEEE (2011)
11. Howe, N.R., Leventon, M.E., Freeman, W.T.: Bayesian reconstruction of 3D human motion from single-camera video. In: Solla S, Leen T, Müller K (eds) Advances in Neural Information Processing Systems, vol 12. MIT Press pp. 820–826 (2000)
12. Huang, P., Hilton, A., Starck, J.: Shape similarity for 3d video sequences of people. Int. J. Computer Vis. **89**(2–3), 362–381 (2010)
13. Ioannidou, A., Chatzilari, E., Nikolopoulos, S., Kompatsiaris, I.: Deep learning advances in computer vision with 3d data: a survey. ACM Comput. Surv. (CSUR) **50**(2), 1–38 (2017)
14. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. **35**(1), 221–231 (2013)
15. Jolliffe, I.: Principal component analysis. In: International Encyclopedia of Statistical Science, pp. 1094–1096. Springer (2011)
16. Kapsouras, I., Nikolaidis, N.: Action recognition on motion capture data using a dynemes and forward differences representation. J. Visual Commun. Image Represent. **25**(6), 1432–1445 (2014)
17. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3 d shape descriptors. In: Symposium on Geometry Processing, vol. 6, pp. 156–164 (2003)
18. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: A new representation of skeleton sequences for 3d action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3288–3297 (2017)

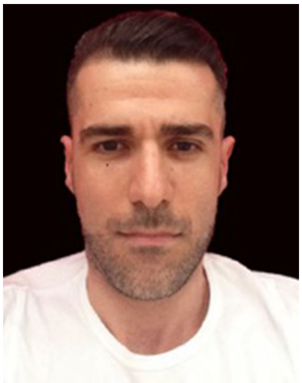
19. Kelley, R., Tavakkoli, A., King, C., Nicolescu, M., Nicolescu, M.: Understanding activities and intentions for human-robot interaction. In: Human-Robot Interaction (2010)
20. Khokhlova, M., Migniot, C., Dipanda, A.: 3D point cloud descriptor for posture recognition. In: Conference on Computer Vision Theory and Applications. Science and Technology Publications (2018)
21. Koppula, H.S., Gupta, R., Saxena, A.: Learning human activities and object affordances from RGD-D videos. Int. J. Robot. Res. **32**(8), 951–970 (2013)
22. Laptev, I.: On space-time interest points. Int. J. Comput. Vis. **64**(2–3), 107–123 (2005)
23. Lillo, I., Carlos Niebles, J., Soto, A.: A hierarchical pose-based approach to complex action understanding using dictionaries of actionlets and motion poselets. In: Conference on Computer Vision and Pattern Recognition, pp. 1981–1990 (2016)
24. Liu, F., Shen, C., Lin, G.: Deep convolutional neural fields for depth estimation from a single image. In: Conference on Computer Vision and Pattern Recognition, pp. 5162–5170 (2015)
25. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal lstm with trust gates for 3D human action recognition. In: European Conference on Computer Vision, pp. 816–833. Springer (2016)
26. Luvizon, D.C., Picard, D., Tabia, H.: 2D/3D pose estimation and action recognition using multitask deep learning. CoRR (2018)
27. Maron, H., Galun, M., Aigerman, N., Trope, M., Dym, N., Yumer, E., Kim, V.G., Lipman, Y.: Convolutional neural networks on surfaces via seamless toric covers. ACM Trans. Graph. **36**(4), 71–1 (2017)
28. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: Conference on Computer Vision and Pattern Recognition, pp. 4674–4683. IEEE (2017)
29. Mihradi, S., Henda, A., Dirgantara, T., Mahyuddin, A.: 3D kinematics of human walking based on segment orientation. In: Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering, pp. 386–390. IEEE (2011)
30. Müller, M.: Dynamic time warping. Information retrieval for music and motion. Springer Berlin Heidelberg pp. 69–84 (2007)
31. Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., Bajcsy, R.: Berkeley mhad: a comprehensive multimodal human action database. In: Workshop on Applications of Computer Vision, pp. 53–60. IEEE (2013)
32. Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., Bajcsy, R.: Sequence of the most informative joint: a new representation for human skeletal action recognition. J. Visual Commun. Image Represent. **25**(1), 24–38 (2014)
33. Pirk, S., Krs, V., Hu, K., Rajasekaran, S.D., Kang, H., Yoshiyasu, Y., Benes, B., Guibas, L.J.: Understanding and exploiting object interaction landscapes. ACM Trans. Graphics **36**(3), 31 (2017)
34. Poppe, R.: A survey on vision-based human action recognition. Image Vis. Comput. **28**(6), 976–990 (2010)
35. Rao, Y., Zhang, M., Cheng, Z., Xue, J., Pu, J., Wang, Z.: Semantic point cloud segmentation using fast deep neural network and dcrf. Sensors **21**(8), 2731 (2021)
36. Rusu, R.B., Bandouch, J., Marton, Z.C., Blodow, N., Beetz, M.: Action recognition in intelligent environments using point cloud features extracted from silhouette sequences. In: RO-MAN, pp. 267–272 (2008)
37. Sevilla-Lara, L., Liao, Y., Güney, F., Jampani, V., Geiger, A., Black, M.J.: On the integration of optical flow and action recognition. In: Brox, T., Bruhn, A., Fritz, M. (eds.) Pattern Recognition, pp. 281–297. Springer Publishing, Berlin (2019)
38. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: a large scale dataset for 3d human activity analysis. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1010–1019 (2016)
39. Silambarasi, R., Sahoo, S.P., Ari, S.: 3D spatial-temporal view based motion tracing in human action recognition. In: Conference on Communication and Signal Processing, pp. 1833–1837. IEEE (2017)
40. Slama, R., Wannous, H., Daoudi, M.: 3d human motion analysis framework for shape similarity and retrieval. Image Vis. Comput. **32**(2), 131–154 (2014)
41. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: Conference on Computer Vision and Pattern Recognition, pp. 567–576 (2015)
42. Starck, J., Hilton, A.: Surface capture for performance-based animation. Comput. Graphics Appl. **27**(3), 21–31 (2007)
43. Suda, R., Takami, M.: A fast spherical harmonics transform algorithm. Math. Comput. **71**(238), 703–715 (2002)
44. Tsai, D.M., Lai, S.C.: Independent component analysis-based background subtraction for indoor surveillance. IEEE Trans. Image Process. **18**(1), 158–167 (2008)
45. Tu, J., Liu, M., Liu, H.: Skeleton-based human action recognition using spatial temporal 3D convolutional neural networks. In: Conference on Multimedia and Expo, pp. 1–6. IEEE (2018)
46. Vantigodi, S., Babu, R.V.: Real-time human action recognition from motion capture data. In: IEEE Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, pp. 1–4 (2013)
47. Veeriah, V., Zhuang, N., Qi, G.J.: Differential recurrent neural networks for action recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4041–4049 (2015)
48. Veinidis, C., Pratikakis, I., Theoharis, T.: Unsupervised human action retrieval using salient points in 3d mesh sequences. Multimed. Tools Appl. **78**(3), 2789–2814 (2019)
49. Vieira, A.W., Nascimento, E.R., Oliveira, G.L., Liu, Z., Campos, M.F.: Stop: Space-time occupancy patterns for 3D action recognition from depth map sequences. In: Iberoamerican Congress on Pattern Recognition, pp. 252–259. Springer (2012)
50. Vlasic, D., Baran, I., Matusik, W., Popović, J.: Articulated mesh animation from multi-view silhouettes. ACM Trans. Graphics **27**(3), 97 (2008)
51. Wang, H., Pirk, S., Yumer, E., Kim, V.G., Sener, O., Sridhar, S., Guibas, L.J.: Learning a generative model for multi-step human-object interactions from videos. Computer Graphics Forum **38**(2), 367–378 (2019)
52. Weinland, D., Ronfard, R., Boyer, E.: A survey of vision-based methods for action representation, segmentation and recognition. Comput. Vis. Image Understand. **115**(2), 224–241 (2011)
53. Weinmann, M., Jutzi, B., Mallet, C.: Feature relevance assessment for the semantic interpretation of 3D point cloud data. ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci. **5**, W2 (2013)
54. Xia, L., Chen, C.C., Aggarwal, J.K.: View invariant human action recognition using histograms of 3d joints. In: 2012 Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 20–27. IEEE (2012)
55. Xie, Z., Chen, J., Peng, B.: Point clouds learning with attention-based graph convolution networks. Neurocomputing **402**, 245–255 (2020)
56. Yan, G., Hua, M., Zhong, Z.: Multi-derivative physical and geometric convolutional embedding networks for skeleton-based action recognition. Comput. Aided Geom. Design **86**, 101964 (2021)
57. Zhang, J., Hu, H.: Domain learning joint with semantic adaptation for human action recognition. Pattern Recogn. **90**, 196–209 (2019)
58. Zhang, S., Liu, X., Xiao, J.: On geometric features for skeleton-based action recognition using multilayer lstm networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 148–157. IEEE (2017)
59. Zhang, Y., Tang, S., Sun, H., Neumann, H.: Human motion parsing by hierarchical dynamic clustering. In: BMVC (2018)
60. Zhao, X., Choi, M.G., Komura, T.: Character-object interaction retrieval using the interaction bisector surface. In: Computer

Graphics Forum, vol. 36, pp. 119–129. Wiley Online Library (2017)

61. Zhong, Y.: Intrinsic shape signatures: A shape descriptor for 3D object recognition. In: Conference on Computer Vision Workshops, ICCV Workshops, pp. 689–696. IEEE (2009)

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

Simone Cammarasana is a PhD student in Computer Science at the University of Genova-DIBRIS. He obtained a post-lauream master in Scientific Computing at the University of Sapienza-Roma (2018) and a master degree in Engineering at the University of Pisa. His research interests include signals analysis, optimisation problems, and linear systems.



Giuseppe Patanè is senior researcher at CNR-IMATI. Since 2001, his research is mainly focused on computer graphics and shape modelling. He is the author of scientific publications on international journals and conference proceedings and the tutor of PhD and Post.Doc students. He is responsible for R&D activities in national and European projects.