Pattern Recognition 53 (2016) 130–147

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

3D skeleton-based human action classification: A survey

Liliana Lo Presti *, Marco La Cascia

V. le delle Scienze, Ed. 6, University of Palermo, 90128 Palermo, Italy

ARTICLE INFO

Article history:
Received 1 April 2015
Received in revised form
27 October 2015
Accepted 24 November 2015
Available online 2 December 2015

Keywords:
Action recognition
Skeleton
Body joint
Body pose representation
Action classification

ABSTRACT

In recent years, there has been a proliferation of works on human action classification from depth sequences. These works generally present methods and/or feature representations for the classification of actions from sequences of 3D locations of human body joints and/or other sources of data, such as depth maps and RGB videos.

This survey highlights motivations and challenges of this very recent research area by presenting technologies and approaches for 3D skeleton-based action classification. The work focuses on aspects such as data pre-processing, publicly available benchmarks and commonly used accuracy measurements. Furthermore, this survey introduces a categorization of the most recent works in 3D skeleton-based action classification according to the adopted feature representation.

This paper aims at being a starting point for practitioners who wish to approach the study of 3D action classification and gather insights on the main challenges to solve in this emerging field.

1. Introduction

In several application domains, such as surveillance [1–4], human–computer interaction [5], assistive technologies [6], sign language [7–9], computational behavioral science [10,11] and consumer behavior analysis [12], the focus is the detection, recognition and analysis of human actions and behaviors. These applications have motivated a large part of the computer vision community to conduct research on action recognition and modeling. Therefore, there is a vast literature, including interesting surveys [13–19], on human body pose estimation [20–24] and activity/action classification [25–30].

As often happens in computer vision, there are psychological studies that motivate current approaches. In the case of human body pose estimation for action classification, Johansson's moving light-spots experiment [31] for motion perception[1] is certainly the most notable. This experiment was conducted in the 1970s to study 3D human motion perception from 2D patterns. The study has the goal of analyzing the visual information from typical motion patterns when some pictorial form aspect of the patterns is known. To these purposes, several bright spots distributed on the human body against a homogeneous, contrasting background are used in the experiment (see Fig. 1). The experiment demonstrates that the number of light-spots and their distribution on the human body may affect motion perception. In particular, an increasing number of light-spots may decrease ambiguity in motion understanding.

Johansson's study demonstrates that human vision not only detects motion directions but can also detect different types of limb motion patterns, including recognition of the activity and of the velocity of the different motion patterns. As reported in [31], "The geometric structures of body motion patterns in man […] are determined by the construction of their skeletons. […] From a mechanical point of view, the joints of the human body are end points of bones with constant length […] ".

This study has inspired most of the literature about human body pose estimation and action recognition [32–34] as, by knowing the position of multiple body parts, we want the machine to learn to discriminate among action-classes.

In particular, works on human body pose estimation try to estimate the configuration of the body (pose) typically from a single, monocular, image [35]. Part detectors [36,37] and/or pictorial structure (PS) models [38–40] are used to model the appearance of body parts and infer the pose based on constraints among body parts; in general, such constraints are meant to represent the actual human body articulations.

The major difficulty in body-pose estimation is that human body is capable of an enormous range of poses, which are also difficult to simulate or to account for. Technologies such as motion capture (Mo-Cap) have been used to collect accurate data and the corresponding ground-truth. Due to the difficulty in reliably estimating the body pose, several approaches have tried to use holistic representation of the body pose. Methods such as [27,41,42] have proved to be successful in performing recognition of simple

* Corresponding author. Tel.: +39 09123899526.
E-mail address: liliana.lopresti@unipa.it (L. Lo Presti).

[1] See for example https://www.youtube.com/watch?v=1F5ICP9SYLU.

http://dx.doi.org/10.1016/j.patcog.2015.11.019
0031-3203/

actions while skipping the human body pose inference step and adopting features that are correlated with the body pose. Emerging trends are action recognition "in the wild" [43–48], and action "localization" [49–53].

Technologies are evolving fast, and the very recent wide diffusion of cheap depth camera, and the seminal work by Shotton et al. [56] for estimating the joint locations of a human body from depth map have provided new stimulus to the research in action recognition given the locations of body joints. Depth map proved to be extremely useful in providing data for an easy and fast human body estimation. As first introduced in [31], the computer vision community defines a *skeleton* as a schematic model of the locations of torso, head and limbs of the human body. Parameters and motion of such a skeleton can be used as a representation of gestures/actions and, therefore, the human body pose is defined by means of the relative locations of the joints in the skeleton. Fig. 2 shows a graphical representation of possible skeletons (estimated by the methods [56,57]) where the red dots represent the estimated human body joints.

Gaming applications and human–computer interfaces are benefiting of the introduction of these new technologies. In the very recent years, we have assisted to a proliferation of works introducing novel body pose representations for *3D action classification* given depth maps and/or skeleton sequences [58–60]. These works have shown how, even with a reliable skeleton estimation, 3D skeleton-based action classification is not that simple as it may appear. In this sense, "one of the biggest challenges of using posed-based features is that semantically similar motions may not necessarily be numerically similar" [61]. Moreover, motion is ambiguous and action classes can share movements [62]. Most of the works in 3D action recognition attempt to introduce novel body pose representations for action recognition from 3D skeletal data [55,34,58] with the goal of capturing the correlation among different body joints across time. Other works [63,64] attempt to mine the most discriminative joints or group of joints for each class. Promising works [65,62] deal with the dynamics regulating the 3D trajectories of the body joints. Across all these works, various different classification frameworks have been applied to classify the actions.

In contrast to very recent surveys such as [66,67] which focus on the use of depths cameras in several application domains or present a review of the literature on activity recognition in depth maps, in this survey we focus mostly on action classification from skeletal data. To this purpose, we present an overview of the technologies used for collecting depth maps (see Section 2) and review the most used methods at the state of the art for estimating skeletons/body pose in Section 3.

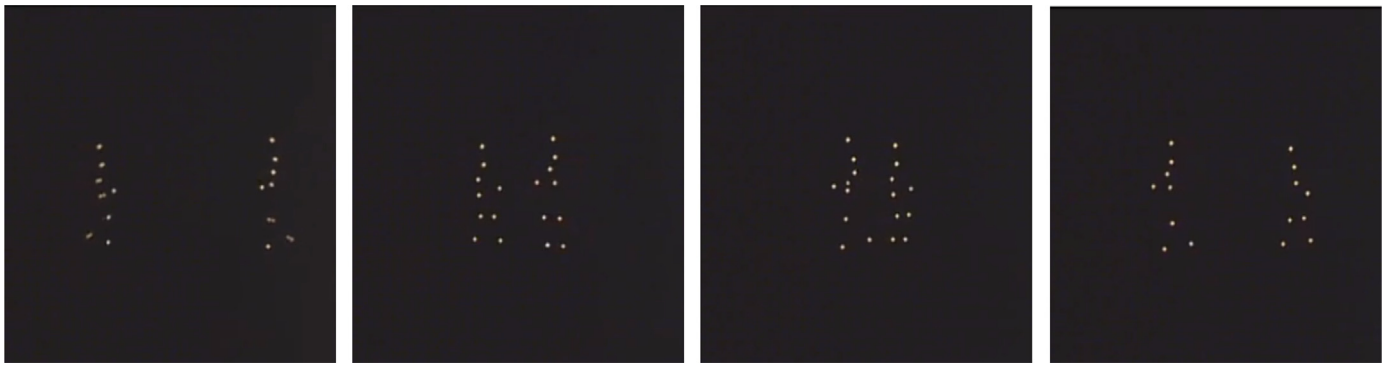

Fig. 1. Four frames of the video (see footnote 1) showing Johansson's moving light-spots experiment: two persons cross each other.

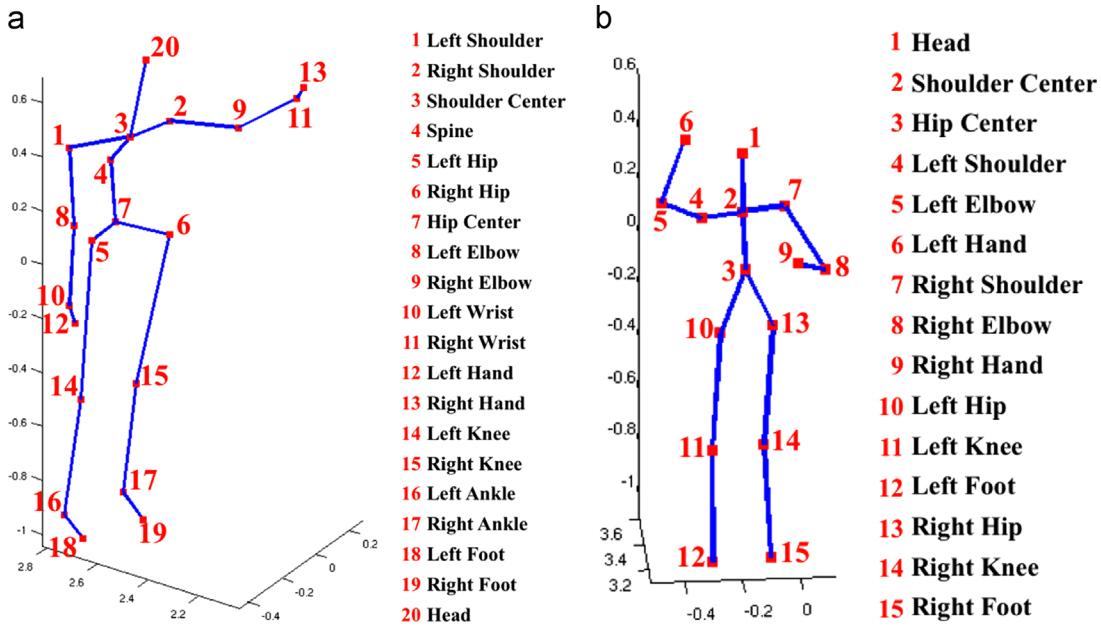

Fig. 2. Graphical representation of skeletal data with 20 and 15 joints. (a) Skeleton of 20 joints (MSRA-3D dataset [54]) and (b) skeleton of 15 joints (UCF Kinect dataset [55]). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Section 4 is devoted to describe the most common data pre-processing practices to account for biometric differences, and various techniques to deal with the classification of sequences of varying temporal length. In Section 5, we introduce a categorization for skeleton-based action representation approaches based on the adopted pose descriptors. We distinguish among *joint-based representations*, *mined joint-based descriptors*, and *dynamics-based descriptors*. In the *joint-based representations*, the skeleton is used to extract a feature representation that captures the correlation of the body joints; the *mined joint-based descriptors* try to learn the subsets of body joints that can help to discriminate among action classes; and the *dynamics-based descriptors* treat the skeleton sequence as a set of time series and use the parameters of the dynamical model regulating the time series to discriminate among action classes.

In Section 6, we briefly describe multi-modal methods where skeletons are used jointly with data acquired by inertial and/or RGB sensors and/or further descriptors extracted from depth maps.

We describe the most adopted publicly available benchmarks and validation protocols for skeleton-based action classification in Section 7. In Section 8, we report the state of the art performance of the reviewed methods on the largest and commonly adopted datasets. We also discuss performance measurements and latency in action recognition. Finally, in Section 9 we present conclusions of our analysis and outline potential future research directions.

2. Depth maps and related technologies

A range image or depth map is an image that stores in each pixel the distance to the camera of the corresponding point in the scene. Range images have been studied and used for long time, especially in robotic applications [68–70] and scene/object reconstruction [71–73]. Whilst several techniques have been devised to estimate depth maps of the scene, only in the last few years depth imaging technology has advanced dramatically, reaching a consumer price point with the launch of Kinect [74].

In comparison with traditional intensity sensors, depth cameras offer many advantages such as they provide a calibrated scale estimate, measurements are color and texture invariant, they permit to solve silhouette ambiguities in pose, and they simplify the task of background subtraction [56].

We briefly describe the three most popular technologies used to estimate a depth map. Technical details are out of the scope of this work, and we refer the interested reader to the corresponding literature.

- *Stereo cameras* consist of a set of calibrated cameras (at least two) for which a common 3D reference system has been estimated. Depth map is estimated based on stereo triangulation [75]. Even if several works [76,77] allowed us to achieve considerable progress in this field, still the depth estimated by stereo cameras can be unreliable, especially for points in the scene with homogeneous intensities/colors.
- 3*D Time-of-flight* (*ToF*) *cameras* illuminate the scene with a modulated light source, and observe the reflected light. Some cameras measure the time-of-flight of a light pulse illuminating the scene, which means the delay experienced by the reflected light. The elapsed time, that is the delay of the reflected light or simply “time of flight”, is correlated with the distance of the object in the scene. Other cameras measure phase differences between emitted and received signals. The estimated distance can be unreliable in case of low spatial resolution, as well as because of errors caused by radiometric, geometric and illumination variations, motion blur, background light, and multiple reflections [78]. Kinect sensors (version 2) belong to this category of cameras.
- *Structured-light* 3*D scanners* project an infrared structured-light pattern onto the scene. When projecting a pattern onto a three-dimensionally shaped surface, the observed pattern is geometrically distorted [79]. By comparing the expected projected pattern (if no object is in the scene) and the deformed observed pattern, exact geometric reconstruction of the surface shape can be recovered. Various patterns can be projected onto the scene, such as light stripes or arbitrary fringes. Depth estimation can be unreliable especially in case of reflective or transparent surfaces. Kinect sensors (version 1) belong to this category of cameras.

3. Body pose estimation

In this section we briefly review the most commonly adopted techniques for estimating the skeletons from a variety of data (intensity or depth maps), then we present some of the available off-the-shelf solutions that could be used for data collection.

3.1. Methods for skeleton estimation

Various attempts have been reported in literature to estimate body parts in RGB data [35,80,21,81] or in depth map [56,82]. We limit our attention to methods or technologies to obtain skeletal data.

- *Motion capture*: Motion capture (Mo-Cap) sensor permits to record actions of human actors. There are several kinds of Mo-Cap technologies. Optical systems use stereo vision to triangulate the 3D position of a subject. Data acquisition takes advantage of special markers attached onto the actor. Some systems use retroreflective material. The camera's threshold can be adjusted to ignore skin and fabric and only image the bright reflective markers. Other systems make use of one or multiple LEDs with software to identify them by their relative positions [13]. This technique is primarily meant to collect precise annotation of the collected intensity data.
- *Intensity data*: Pose estimation from images and RGB video is still an open and very challenging problem. The main difficulties arise due to variability of human visual appearance, inter-subjects biometric differences, lack of depth information and partial self-occlusions [35]. Nonetheless, very interesting and promising approaches have been presented in literature. As pose estimation is not the main topic of this survey, we refer the reader to the ones we believe are the must-know papers for computer vision scientists [83–86].
  In recent years, the most adopted model for pose has probably been pictorial structure (PS) [38]. This model decomposes the appearance of the whole body into local body-parts, each one with its own position and orientation in space. Statistical pairwise constraints on body parts enforce skeletal consistency and allow to represent body articulations.
  Further evolutions of this model have focused on modeling of the appearance of the parts by stronger detectors [36,80] or mixture of templates [21]. Other methods enhance the model by introducing further constraints among body parts [86,87]. While the original PS does use a tree model [38,21], which permits exact inference of the body pose, other methods adopt non-tree models. In these cases, the computational complexity increases so much that pruning strategies [24,39,88] or approximated inference [87,89] are in general adopted. Discriminative learning methods, such as conditional random field [40] and

Table 1
Hardware and software technologies for skeletal/depth data collection.

Name of product	Description	Link to Manufacturer's Website
ZED Stereo Camera	Lightweight depth sensor based on passive stereo vision (pair of RGB cameras)	https://www.stereolabs.com/
The Captury	Marker-less motion capture system that uses standard video cameras	http://www.thecaptury.com/
OptiTrack Mo-Cap	Multi-camera system for motion tracking	http://www.optitrack.com/
Vicon	Passive optical motion capture with different available configurations	http://www.vicon.com/
DepthSense 325	Short range depth sensor, built-in color camera and microphones	http://www.softkinetic.com/
Kinect sensor v1	Depth sensor, built-in color camera, infrared (IR) emitter, and a microphone array	http://www.microsoft.com/
Kinect sensor v2	Built-in color camera, time-of-flight (ToF) camera, and a microphone array	http://www.microsoft.com/
Intel RealSense 3D Camera	Built-in color camera, infrared camera, and infrared laser projector	http://www.intel.com/
Asus Xtion PRO Live	Infrared sensor, color camera and microphones	https://www.asus.com/
Creative Senz3D	Infrared sensor, color camera and microphones	http://us.creative.com/
Infineon 3D Image Sensor	Time-of-flight (ToF) camera	http://www.infineon.com/
Heptagon-Mesa Imaging Sensor	Time-of-flight (ToF) camera	http://enterprise.hptg.com/
Brekel Pro Pointcloud	Kinect-based motion capture system	http://brekel.com//
iiwSoft	Kinect-based motion capture system	http://ipisoft.com/

max-margin learning [90,21,89], are often used to learn the part-based model parameters.

- *Depth maps*: The most renowned method to estimate skeletons from depth images is probably the recent work in [56], which is applied and widely used to infer skeletons composed of 20 joints (25 joints with the Kinect for Windows version 2) on a frame-by-frame basis from depth images collected by the Kinect sensor. The work in [56] proposes to segment a single input depth image into a dense probabilistic body-part labeling that covers the whole body. The segmented body parts are used to estimate 3D body joint proposals without exploiting temporal or kinematic constraints.
  In particular, human body is represented by 31 body parts (L=Left, R=Right, U=Upper, W=loWer): LU/RU/LW/RW head, neck, L/R shoulder, LU/RU/LW/RW arm, L/R elbow, L/R wrist, L/R hand, LU/RU/LW/RW torso, LU/RU/LW/RW leg, L/R knee, L/R ankle, and L/R foot.
  In a nutshell, from a single input depth image, a per-pixel body part distribution is inferred by classification. Pixel classification is performed by a deep randomized decision forest. A forest is an ensemble of decision trees, each consisting of split and leaf nodes. Each split node divides the training samples based on a feature and a threshold (that are parameters of the classifier). Leaf nodes store class-conditional distributions (where classes indicate body parts). To classify a pixel in a depth image, each tree is traversed branching left or right according to the comparison of the feature and threshold stored in the split node. Once a leaf node is reached, the pixel is assigned with the class-conditional distribution stored in the leaf node. The estimated distributions are averaged over all the trees in the forest. The per-pixel information is pooled across pixels to generate proposals for the positions of 3D skeletal joints. To this purpose, mean shift with a weighted Gaussian kernel is adopted.
  One of the most notable aspect in [56] is the size and composition of the training data. A huge amount of realistic synthetic depth images of humans of many shapes and sizes in highly varied poses together with a large motion capture database and depth images acquired with the Kinect sensor has been used to train the pixel-classifier. The large size of the adopted training set permits to avoid over-fitting and account for varying human body poses.
  Another widely used method for skeleton estimation is the one offered by the OpenNI library [91]. The method [57] estimates a skeleton of 15 body joints by means of local descriptors representing statistics in spatial bins of patches of the depth map. The local descriptors provide depth-edge counts of a depth patch in a set of bins that are arrayed radially around the center point of the patch. Depth edges are extracted from the depth map by means of an edge detector, such as a Canny edge detector. In each radial slice of the patch, statistics about the number of pixels in each bin that are classified as edge pixels, the direction of the edge, the mean or median direction in each bin, the mean or median depth value, the variance of the depth are computed and stored. The bin values may be weighted and normalized to compensate for differences in bin areas.
  The approximate nearest neighbors (ANN) algorithm is then used to search the patch descriptors into a descriptor database. This database stores descriptors extracted at known locations of a humanoid form and are associated with the locations of body joints. Thus, after matching the test patch with a patch in the database (by means of the Euclidean distance for example), it is possible to compute an estimated location for the joints in the test depth map. After comparison of all the test patches, the collection of estimated joint locations from the matched patches is used within a weighted voting process to recover the location of the joints in the skeleton one-by-one. The method also applies normalization and scale estimation to make the skeleton reconstruction invariant to the distance of the subjects to the camera and to account for biometric differences.

3.2. Off-the-shelf solutions

The term passive stereo vision refers to systems that use two (or more) calibrated cameras in distinct locations to determine which points in both images correspond to the same 3D scene point. In contrast, active stereo vision refers to methods where the scene illumination is controlled, for example by projecting known patterns onto the objects.

Table 1 summarizes some of the currently available solutions that may be adopted for data collection. This list is not exhaustive and many other reliable solutions are available on the market.

The first part of the table reports passive stereo vision technologies, while the second part of the table reports active stereo vision systems. For each product we report the name, a short description and the link to the manufacturer's website. We note that most of the technologies, such as the Kinect sensor, the Intel RealSense 3D camera, and the Mo-Cap solutions come with their own software development kit (SDK) and/or proprietary software applications for the recording of skeletal data.

4. Pre-processing of skeletal data

Pre-processing of skeletal data is often used to cope with biometric differences among subjects, and varying temporal duration

of the sequences that may be due to different velocity of the actions and inter-subjects style variations.

In particular, accounting for biometric differences may require a transformation of each skeleton into a normalized pose. The problem is closely related to that of motion retargeting [92,93], which raises in computer graphics whenever it is required to adapt the motion from one animated character to another. Motion retargeting generates the parameters of the motion for the target character by satisfying constraints about the resulting poses (such as specific configurations of the joints' angles [94] or specific properties of the generated motion [95,96]). In the field of 3D action representation, such techniques might be useful in transferring the motion of each skeleton sequence to a generic reference pose, resulting in skeleton sequences that are invariant to biometric differences. In this sense, the techniques that are applied in literature and reviewed in Section 4.1 are related to the broader motion retargeting problem.

Varying length of the action sequences may limit the applicability of some classification frameworks. For example, the widely used support vector machine classifier requires that the feature representations of the samples have the same size. As explained in detail in Section 4.2, most of the works in literature attempt to extract histograms from the whole sequence, or apply some pyramidal approach, or adopt dynamic time warping to align each sequence to a reference one.

4.1. Data normalization and biometric differences

In 3D skeleton-based action classification, an action is described as a collection of time series of 3D positions (i.e., 3D trajectories) of the joints in the skeleton. However, this representation depends on the choice of the reference coordinate system, which is different in every recording environment, and on biometric differences. To account for such issues, a variety of coordinate system changes is used. Works such as [64], consider the joint angles between any two connected limbs and represent an action as a time series of joint angles. In [97], human poses are normalized by aligning torsos and shoulders. In [34], data are initially registered into a common coordinate system to make the joint coordinates comparable to each other. In [98,99], all of the 3D joint coordinates are transformed from world coordinate system to person centric coordinate system by placing the hip center (see Fig. 2) at the origin. In [63], skeletons are aligned based on the head location. Next, all the other coordinates in the skeleton are normalized by the head length. In [100], the coordinates of the 3D joints are normalized so to range in [0, 1] in all the dimensions over the sequence. In [101], each coordinate of the normalized pose vector is smoothed across time with a 5 by 1 Gaussian filter ($\sigma = 1$). Poses are normalized in order to compensate for biometric differences. A reference skeleton is learned from training data. The lengths of skeleton limbs in the reference skeleton are adjusted to unit norm. All the skeletons in the test and training data are transformed in order to impose same limb segment length as in the reference skeleton while preserving the direction vector. A similar normalization is applied also in [102], where inter-frame linear interpolation is also applied to account for missing values in the skeletal data.

4.2. Dealing with varying temporal durations

One of the main issues in action classification is that the sequences may have varying length. The length of the action sequence may depend on the velocity and style with which the action is performed. In general, there may be a temporal warping that affects the action sequence or, in case of repetitive movements, a different number of repetitions.

To account for this issue, works such as [58,63] adopt global feature representation of the entire sequence sacrificing, in general, the information about the temporal structure of the sequence. The most common approach is the bag-of-word schema, which represents a sequence in terms of codewords in a dictionary. The representation can include temporal information by adopting a pyramidal approach [103,100,102].

To account for the temporal warping and ensure equal length of the sequences, in [34] the sequences are aligned to a nominal curve. Approaches adopting some distance among the trajectories of 3D joints, such as [104], apply dynamic time warping and *K*-nearest neighbor. In [105], a dynamic programming based elastic distance is used. In [97], the sequences are resampled in order to obtain representations of comparable length.

Another approach used in [64] to obtain representations of the same length is that of dividing the sequence in a prefixed number of temporal segments. However, the problem of finding the most appropriate number of segments to use across all the classes and sequences is not trivial.

Therefore, [64,62] propose to divide the action sequence in a varying number of temporal segments each of the same temporal duration. In particular, [62] adopts a sliding window approach where the temporal segments are all partially overlapping. The sliding window approach makes the method more robust to the temporal warping of the sequence. Such kind of representation does not allow the adoption of a standard SVM and in [62] a set of hidden Markov models trained with a discriminative approach is used instead. Also [101] adopts a sliding window-based representation. The feature representation of each temporal window is classified via *K*-NN. The entire sequence is classified by a voting schema based on the labels predicted for each temporal window. In [102], a pooling strategy is adopted to build a representation of the sequence of skeletons in several temporal windows reducing all the sequences to have a same length representation.

Approaches like [99,65] overcome the issue of the length of the sequences by assuming the same order of the linear dynamic system for all the action classes and performing system identification to compute the parameters of the model.

5. Action representation and classification

As depicted in Fig. 3, we categorize the methods for 3D action representation into three categories: *joint-based representations*, that is methods that extract alternative feature representations from the skeletons in order to capture the correlation of the body joints; *mined joint based descriptors*, that is methods trying to learn what body parts are involved and/or are useful to discriminate among actions; and *dynamics-based descriptors*, that is methods that treat the skeleton sequence as 3D trajectories and model the dynamics of such time series. In turn, the *joint-based representations* can be categorized into *spatial descriptors*, *geometric descriptors*, and *key-pose based descriptors* according to the kind of characteristics in the skeleton sequence used to extract the corresponding descriptor.

5.1. Joint-based representation

The methods belonging to this category attempt to capture the relative body joint locations. In the following, we review works at the state of the art that fall in this category and organize the discussion by distinguishing among three sub-categories: *spatial descriptors*, *geometric descriptors*, and *key-pose based descriptors*. The first sub-category includes works that try to correlate the 3D body joints by measuring all the possible pairwise distances (at a given time or across time) or their covariance matrices. The second

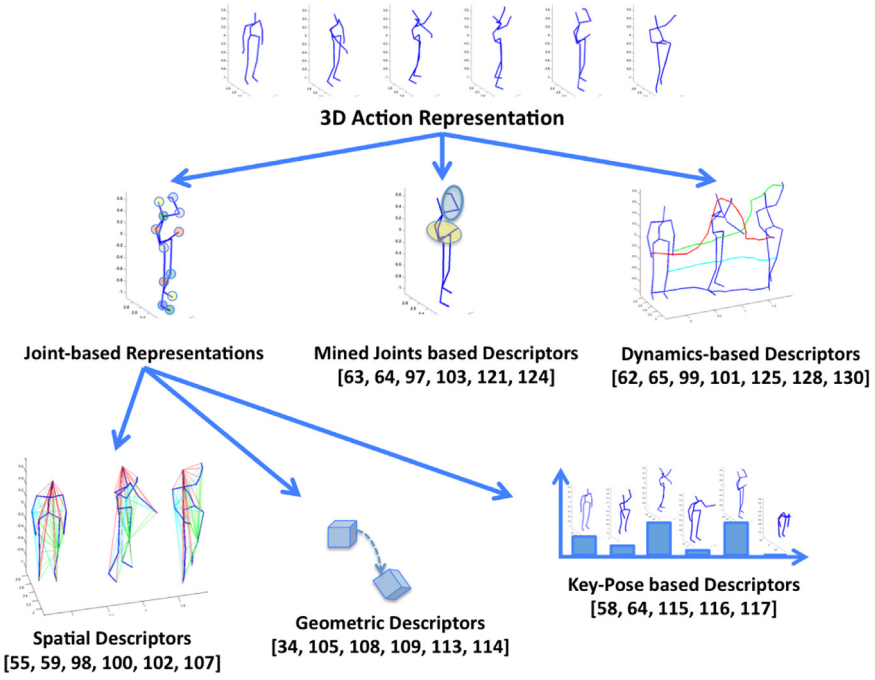

Fig. 3. Graphical representation of our categorization of the methods for 3D action representation.

category, that is the geometric descriptors, includes methods that try to estimate the sequence of geometric transformations required to move skeleton across time or to represent the relative geometry of sub-sets of joints. In the latter category, that is key-pose based descriptors, a set of key-poses is computed and a skeleton sequence is represented in terms of the closest key-poses.

5.1.1. Spatial descriptors

The simplest attempt to correlate body joints' locations is probably to consider all the pairwise distances of the 3D joints. Such kind of representation lacks of any temporal information and may result in ambiguous descriptions of the action sequence. Therefore, in [55], body pose is represented by concatenating the distances between all the possible pairs of joints in the current frame, the distances between the joints in the current frame and the ones in the previous frame, the distances between the joints in the current frame and in a neutral pose (computed by averaging the initial skeletons of all the action sequences). Each individual feature value was clustered into one of 5 groups via *K*-means and replaced with a binary vector to represent each cluster index. Canonical pose descriptors for each action are discovered via a multiple instance learning approach within the logistic regression framework.

In a similar way, [59] employs 3D position differences (and not distances) of joints in the same skeleton, between the 3D joints in the current frame and in the preceding frame, and between the current frame and the initial one. Principal component analysis (PCA) is applied for dimensionality reduction providing a descriptor called EigenJoints. Naive-Bayes-nearest-neighbor classifier is used for action classification.

In [102], a skeleton is interpreted as a graph whose edge weights are computed based on pair-wise distances. Further edges are included to link joints of temporally consecutive skeletons. Hence, the whole skeleton sequence is represented as a spatio-temporal graph. A pyramidal representation based on spectral graph wavelet transform (SGWT) [106] is used to capture information about the joints' trajectories at different scales (in space and time). To deal with the high dimensionality representation, PCA is applied followed by a l2-normalization. Classification is performed by standard SVM.

A further attempt to capture the relations among the joints in a skeleton is that of representing a skeleton sequence by means of its covariance matrix. In this sense, in [100], a sequence of skeletons in a fixed length temporal window is represented by means of a covariance matrix, which encodes information about the shape of the joint probability distribution of the set of random variables. To consider temporal dependences of the 3D joints, a hierarchical representation is adopted: the root node represents the covariance matrix of the entire sequence of skeletons, and lower levels represent the covariance matrices in overlapping fixed length temporal window of decreasing lengths. Computation of the descriptor can be done by dynamic programming while action classification is performed by linear SVM.

We include in this section also methods that try to capture the correlation among joints' locations through convolutional neural networks. In [98], a HMM is used to model an action. Each action is represented in a similar way as done in [59]. The main difference between this approach and other methods adopting HMM is that in [98] the emission probabilities are replaced by deep neural networks that contain many layers of features. In [107], only the joints corresponding to the right-hand, the left-hand and the pelvis are used to extract features relevant for the actions to recognize. A convolutional neural network classifier, consisting of an alternating sequence of convolution and subsampling layers followed by a neural network, is adopted for classification.

5.1.2. Geometrical descriptors

The methods in this category attempt to represent a skeleton by means of the geometric relations among different body parts. The geometric features presented in [108] consist of a set of boolean features each one associated with a quadruple of joints. Given a set of four points, three of them are used to identify a plane; the boolean feature corresponding to the given quadruple assumes value 1 if the fourth point is in front of the plane, otherwise it assumes value 0. This kind of feature allows representing the geometric relations among sets of joints and is robust to spatial variations, global orientation changes and size of the skeleton. In

[108], only 31 boolean features are manually identified and used for content-based motion retrieval.

Also in [109], joints are considered in quadruples. In this case, two out of the four joints in the quadruple are used to set a coordinate system where one of the points is used as origin, while the most distant point in the quadruple is represented in the new coordinate system as [1,1,1]. The resulting similarity transformation is applied on the remaining two joints, which form a skeletal quad. A skeleton is then represented as a set of skeletal quads. For each class, a Gaussian mixture model is trained by means of expectation maximization. The parameters of the model are then used to extract Fisher scores [110]. The concatenated scores are used to train a multi-class linear SVM and perform action classification.

More complex is the representation introduced in [34], where the relative 3D geometry between different body parts is explicitly estimated. Given two rigid body parts, their relative geometry can be described by considering a rigid-body transformation (rotation and translation) to align one body part to the other. Such rigid-body geometrical transformation is in the matrix Lie group $SE(3)$ [111]. The skeleton representation is a set of points in $SE(3)$ where each point represents the relative geometric transformation between a pair of body parts. A sequence of skeletons is a curve in the Lie group $SE(3) \times \cdots \times SE(3)$, which is a curved manifold. For classification purposes, each action curve is mapped from $SE(3) \times \cdots \times SE(3)$ to its Lie algebra, which is the tangent space at the identity element of the group. To account for varying duration of the action sequences and temporal warping, dynamic time warping (DTW) is used to align each sequence to a nominal curve. The warped curves are then represented by means of the Fourier temporal pyramid (FTP) proposed in [112], that is by removing the high frequency coefficients. Action classification is performed using FTP and linear one-vs-all SVM classifiers.

Methods such as [105,113] attempt to consider the geometric relations between skeleton sequences rather than body parts in the same skeleton. In [105,114], each action is represented by spatio-temporal motion trajectories of the joints. Trajectories are curves in the Riemannian manifold of open curve shape space, and a dynamic programming-based elastic distance is used to compare them. Classification is performed by KNN on the Riemannian manifold.

The work in [113] proposes a class of integral invariants to describe motion trajectories, and achieves an effective and robust motion trajectory matching. The representation is computed by calculating the line integral of a class of kernel functions at multiple scales along the motion trajectory. The integral invariants have a smoothing effect and are therefore less sensitive to noise without preprocessing the motion trajectory. Two motion trajectories are considered equivalent if and only if there exists a group transformation to map one trajectory onto the other one. Dynamic time warping and nearest neighbor are used for classification purposes.

5.1.3. Key-poses based descriptors

This category includes methods that learn a dictionary/codebook of key-poses and represent an action sequence in terms of these key-poses.

A histogram of motion words has been used as baseline method in [64] where the concatenated 3D joint locations (or their corresponding features) are clustered into K distinctive poses (i.e., motion words) by using K-means. Each action sequence is represented by counting the number of detected motion words. Motion words are detected by assigning each skeleton representation to its closest distinctive pose. The resulting histogram is then normalized by its l1-norm.

The method detailed in [115] groups all poses composing an action into a set of clusters based on the Hausdorff distance. Each of these clusters is defined by a representative element, which is the median element of the cluster of skeletons. The sequence of skeletons is then represented by means of the cumulative frequency occurrences (integral histogram) of the set of representative elements in the sequence. Because integral histograms lack of temporal information about poses, an action is decomposed into time-ordered sub-actions by considering all the possible substrings of poses. Given two action sequences, comparison of the two sequences is done by finding the optimal sub-action decompositions that yield the minimum score. The decomposition is found by dynamic programming and each sub-action is represented by the corresponding integral histogram. The method uses the Bhattacharyya distance to compare different histograms.

The method in [116] divides the body into 4 spatial regions: *right arm*, *right leg*, *left arm*, and *left leg*. Each body region is represented by a feature vector of 21 dimensions comprising the line-to-line angles between body joints and the 6 line-to-plane angles. A dictionary of body poses is obtained by standard K-means clustering algorithm. A hierarchical model is trained to represent complex activities as a mixture of simpler actions.

Also the method in [117] employs the idea of using key-poses to represent an action. In particular, a descriptor for the skeletal data similar to that proposed in [118] is adopted in order to obtain a representation robust to orientation changes. The representation consists of the spherical coordinates of joints corresponding to head, elbows and knees with respect to the torso basis. Extremities such as hands and feet are represented by considering the rotations of these body parts with respect to the connected joints. Then, a multi-class SVM is used to detect key poses while a decision forest permits to recognize human gestures from sequences of key-poses.

We also include in this category the work in [58] where a histogram of the locations of 11 manually selected 3D skeleton joints is computed to get a compact body pose representation that is invariant to the use of left and right limbs (histogram of 3D joints – HOJ3D). Linear discriminant analysis (LDA) is used to project the histograms and compute the K discrete states of the HMM classifier. These K discrete states may be considered as elements of a key-poses dictionary.

5.2. Mined joint-based descriptors

Even if each individual may perform the same action with a different style, in general the required movements involve similar subsets of joints. Detection of the activated subsets of joints can help to discriminate among different action classes. Methods such as [64,63,97] focus on mining the subsets of most discriminative joints or consider the correlation of subsets of joints.

The method in [103] models each joint by its location and velocity in a spherical coordinate system, and by the correlation between location and velocity represented as the orthogonal vector to the joint location and velocity. An action is modeled as a set of histograms, each computed over the sequence on a specific feature and joint. A temporal pyramid is used to capture the temporal structure of the action. Partial least squares (PLS) [119] is used to weight the importance of the joints and kernel-PLS SVM [120] is adopted for classification purposes.

The approach in [121] employs a genetic algorithm to select the joints to represent an action class. In [64], the most informative joints in a temporal window are found by exploiting the relative informativeness of all the joint angles based on their entropy. Under the assumption of Gaussian random variables, the entropy is proportional to the logarithm of the variance. The joint that has the highest variance of angle changes can be defined as the most

informative, assuming the joint angle data are independent and identically distributed (i.i.d.) samples from a one-dimensional Gaussian. Therefore, the ordered sequence of the most informative joints (SMIJ) in a full skeletal motion implicitly encodes the temporal dynamics of the motion and is used to represent an action.

The work in [63] groups the estimated joints into five body parts. Contrast mining algorithm [122] in the spatial domain is employed to detect distinctive co-occurring spatial and/or temporal configurations (poses) of body parts. Such co-occurring body-parts (part-sets) form a dictionary. By adopting a bag-of-words approach, an action sequence is represented as an histogram of the detected part-sets and 1-vs-1 intersection kernel SVMs are employed to classify the sequences.

In contrast to the former works, where the mining of interesting parts is performed as a step integrated into the feature representation, in [97] the mining of the most informative body parts is performed by adopting a multiple kernel learning (MKL) [123] method. Body parts are represented as trajectories of difference vectors between the 3D joints. In practice, for each 3D joint, all the difference vectors to the remaining joints are considered. Symlet wavelet transform is applied, and only the first V wavelet coefficients are retained. The effect of applying the wavelet transformation is that of noise and dimensionality reduction of the data. To account for different duration of the action sequences, the time series are interpolated in order to obtain 128 samples. The final descriptor consists of the concatenation of the first V coefficients of all the 3D joint time series in the skeleton.

The method in [124] adopts a multi-part modeling of the body. The coordinates of each joint are expressed in a local reference system, which is defined at the preceding joint in the chain. Body sub-parts are aligned separately and a modified nearest-neighbor classifier is used to perform action classification by learning the most informative body parts.

5.3. Dynamics-based descriptors

Methods for 3D skeleton-based action recognition in this category focus on modeling the dynamics of either subsets or all the joints in the skeleton. This can be accomplished by considering linear dynamical systems (LDS) [99,65] or hidden Markov models (HMM) or mixed approaches [62].

The parameters obtained from the LDS modeling of the skeletal joint trajectories likely describe positions and velocities of the individual joints. In [99], the skeleton is divided in several body parts represented by subsets of joints such to represent the full body, the upper body, the lower body, the left/right arms, the left/right legs, etc. For each of these body parts, a shape context feature representation is computed by considering the directions of a set of equidistant points sub-sampled over the segments in each body part. The histogram of such direction is then normalized. The skeleton sequence is represented as a set of time series (one for each body part) of features such as position, tangents and shape context features. The time series are further divided into several temporal scales starting from the entire time-series data in a particular sequence to smaller and smaller equal-sized temporal parts. Each individual feature time series is modeled using an LDS and the method learns the corresponding system parameters by performing system identification. The estimated parameters are used to represent the action sequence. Multiple kernel learning (MKL) [123] is used to learn a set of optimal weights for each part configuration and temporal extent.

The LTBSVM in [65] describes an action as a collection of time series of 3D locations of the joints. Each action sequence is represented as a linear dynamical system that has produced the 3D joint trajectories. In particular, an autoregressive moving average model is adopted to represent the sequence. The dynamics captured by the ARMA model during an action sequence can be represented by means of the observability matrix, which embeds the parameters of the model. Therefore, two ARMA models can be compared in terms of their finite observability matrix. The subspace spanned by columns of this finite observability matrix corresponds to a point on a Grassmann manifold. Comparison of different models can be performed on the Grassmann manifold by considering the principle angles among the subspaces. Then a control tangent (CT) space representing the mean of each class is learned. Each observed sequence is projected on all CTs to form a local tangent bundle (LTB) representation and linear SVM is adopted to perform classification.

In [62,125], autoregressive models are used to represent the 3D joint trajectories but, in contrast to [65], no system identification is performed to recover the parameters of the AR model. Instead, a set of trajectories of the 3D joints is represented by an Hankelet [126], which embeds the observability matrix of the linear time invariant (LTI) system. A subspace distance to compare Hankel matrices is approximated through a dissimilarity score [127]. By means of a sliding window approach, an action is represented as a sequence of Hankelets. An HMM allows modeling the transition from one LTI system to another, yielding to a model for switching dynamical systems. Being an Hankelet invariant to affine transformation [126], no pre-processing of the data is required.

Another related work is the dynamic forest model (DFM) [128], where a set of autoregressive trees [129] is used. An autoregressive tree is a probabilistic autoregressive tree for time-series data in which each leaf node stores a multivariate normal distribution with a fixed covariance matrix, and the predictive filtering distribution depends on the visited nodes on the tree, representing each one a past observation. Each autoregressive tree is trained separately using bagging. The set of Gaussian posteriors estimated by the forest are used to compute the forest posterior, which can be represented as a multimodal mixture of Gaussians.

We also include in this section the hierarchical Dirichlet process-hidden Markov model (HDP-HMM) method [130]. This is a non-parametric variant of the HMM where each state has state specific transition probabilities and the model has an unbounded number of states. The model is trained in a discriminative way and allows action classes to share training examples.

Another related representation is the one proposed in [101], where the body pose is viewed as a continuous and differentiable function of the body joint locations over time. Under this assumption, it is possible to locally approximate such body pose function by means of its second-order Taylor approximation in a window around the current time step. In this way, the local 3D body pose can be completely characterized by means of the current joint locations and differential properties like speed and acceleration of human body joints. An informative descriptor is obtained by concatenating the normalized 3D pose, the first and the second order derivatives estimated in a temporal window of 5 frames. Here, the velocity describes the direction and speed of the 3D joints while the acceleration captures the change in velocity over time. A non-parametric action classification scheme based on K-nearest-neighbors (KNN) and a voting scheme is adopted.

6. Fusing skeletal data with other sensors' data

Several other approaches in literature [54,112,131] fuse information extracted from several streams: skeletal data, RGB videos, and depth maps. Here, we briefly describe these approaches as they are not directly comparable to methods that use only skeletal data.

In general, works using only depth maps tend to describe the human body surface based on statistics of the estimated depth. Li et al. [54] propose to use an action graph where each node is a bag of 3D points that encodes the body pose. Wang et al. [132] treats a 3D action sequence as a 4D shape and a random occupancy patterns (ROP) feature is extracted. Sparse coding is used to encode only the features that contain information useful for classification purposes. In [133], space and time axes are divided in cells and space–time occupancy patterns are computed to represent depth sequences. Oreifej et al. [60] described the depth sequence as histograms of oriented surface normals (HON4D) captured in the 4D volume, based on depth and spatial coordinates.

Rahmani et al. [134] propose to use the histogram of oriented principal components (HOPC) to capture the local geometric characteristics around each point in the depth map by applying PCA to a volume of data around each point. HOPC is formed by concatenating the projected eigenvectors in decreasing order of their eigenvalues. A pruning process allows retaining only the keypoints with the highest quality. Classification is performed by a histogram intersection kernel SVM.

There is also the trend to re-interpret methods that proved to be successful for action recognition from RGB video and that may prove to be useful on depth data. For example, in [135], a spatio-temporal feature for depth maps based on a modified histogram of oriented gradients (HOG) is adopted. Skeletal data are used to detect body parts in the depth map and extract local histogram descriptors. The HOG-based descriptor is combined with the joint angles similarity (JAS), which measures the pairwise similarity of joint angles along the gesture. In [136], an algorithm for extracting spatio-temporal interest points from depth videos (DSTIPs) is presented. The local 3D depth cuboid similarity feature (DCSF) is extracted around each DSTIPs. Such descriptor encodes the spatio-temporal shape of the 3D cuboid by measuring the similarity of sub-blocks composing the 3D cuboids.

Other works use the skeletal data to detect body parts and extract local features from the depth map or RGB video. In [112], depth data and the estimated 3D joint positions are used to compute the local occupancy pattern (LOP) feature, that is depth statistics around joints in the skeleton. Fourier temporal pyramid is used to capture the temporal structure of the action. The set of features computed for a skeleton is called actionlet. Data mining techniques are used to discover the most discriminative actionlets. Finally, a multiple kernel learning approach is used to weight the actionlets. Sung et al. [131] combined RGB, depth and hand positions, body pose and motion features extracted from skeleton joints. HOG is used to describe both RGB and depth images. Then, a two-layer maximum-entropy Markov model is adopted for classification. In [137] the authors propose to fuse skeleton information and STIPS based features [138] by a random forest. A random forest is trained to perform feature fusion, selection, and action classification altogether.

In [104], both skeletal and silhouette-based features are combined to account for body pose estimation errors. The silhouette is obtained by background suppression and its shape by considering the boundary of the silhouette itself. Then a histogram of the shape is computed by adopting a radial schema centered on the silhouette. The final descriptor is obtained by combining the two features. A bag of words approach is adopted to represent the human body pose in terms of a codebook of key poses. Dynamic time warping and the nearest neighbor classifier are adopted for classification purposes.

The method in [139] focuses on human activity recognition in sequences of RGB-D data. The method fuses shape and motion information. Shape features are used to describe the 3D silhouette structure and are extracted from the depth map using spherical harmonics representation. Motion features are used to describe the movement of the human body and are extracted from the estimated 3D joint positions. In particular, the method uses only four distal limb segments of the human body to describe the motion: left and right lower arm parts and left and right lower leg parts. Each distal limb segment is described by the orientation and translation distance with respect to the initial frame. Multiple kernel learning (MKL) technique is then used to produce an optimal kernel matrix within the SVM classification framework.

In [140] action alignment and classification are solved together by means of maximum margin temporal warping (MMTW), which learns temporal action alignment for max-margin classification. The problem is formulated as a latent support vector machine where a phantom action template for representing an action class is learnt. Multiple features are used to represent an action: skeletal data are represented as in [112], depth data are represented as in [60] while RGB data are represented by means of HOG and HOF. All the descriptors are concatenated together.

Finally, we consider works that attempt to obtain a reliable action/activity recognition by fusing features extracted from depth/skeletal data and accelerometer measurements from wearable inertial sensors. The method in [141] adopts a motion descriptor (similar to the motion history image in [142]) extracted from depth maps and acceleration statistics such as mean, variance, standard deviation and root mean square over a temporal window. Dempster–Shafer theory is used to combine the classification outcomes from two classifiers, each corresponding to one sensor typology. In [143], an inertial sensor and a depth camera are used to monitor food intake gestures (such as fine cutting, loading food, and maneuvering the food to the mouth). Position and displacement of body parts (wrist, elbow, and shoulder) are estimated from depth data and combined with the accelerations measured by wearable inertial sensors. In a similar way, in [144,145] measurements from accelerometers and skeletal data are concatenated and used to classify various activities. In particular, in [144] the concatenated features are used as the input to an ensemble of binary neural network classifiers; in [145], a hidden Markov model (HMM) classifier over the observed concatenated features was used for hand gesture recognition.

The joint use of skeletal data and accelerometer measurements may prove very useful for assistive and/or health monitoring technologies. However, even if in principle methods adopting also inertial sensors may boost the accuracy values in action classification, wearable sensors may be unavailable or unreliable in other domains. We note that, among all the datasets described in Section 7, only the Berkeley multimodal human action database (MHAD) [148] provides accelerometer measurements.

7. Datasets and validation protocol

Table 2 summarizes the main characteristics of the most commonly used datasets in skeleton-based action classification. The table reports: reference to the paper introducing the dataset,

Table 2
The most commonly used datasets for skeleton-based action recognition.

Dataset	Ref.	Classes	Seq.	NS	RGB	Depth	Skel.	EP
UCF	[55]	16	1280	16	N	N	Y(15)	4-folds CV
MHAD	[148]	11	660	12	Y	Y	Y(30)	CS-V
MSRA 3D	[112]	20	557[a]	10	N	Y	Y(20)	HS-V(P1, P4), 3F-1:2 (P2), 3F-2:1 (P3)
MSRDA	[112]	16	320	16	Y	Y	Y(20)	HS-V
UTKA	[58]	10	195	10	Y	Y	Y(20)	LOOCV

[a] After removing 10 noisy sequences as suggested in [112].

number of action classes, number of sequences in the dataset, number of subjects performing the actions, the presence (N=No and Y=Yes) of RGB, depth and skeletal data, suggested validation protocol. The validation protocols are described in Table 3. The number in the skeleton column indicates the number of joints in the skeleton. These benchmarks are designed for segmented action classification and, in general, each action starts and ends in a neutral pose. In the following, we briefly describe each of these datasets. As this survey is about 3D skeleton-based action classification, we mainly focus on datasets providing skeletal data. Therefore, the table does not include datasets such as Hollywood 3D [146], which provides only depth data, or Human3.6M [147],[2] which is providing an impressive mole of data with the main goal of testing body pose estimation methods. These data are not meant to test action classification approaches.

As Table 2 shows, there are huge variations in the datasets in terms of number of actions, number of subjects involved in the data collection and number of sequences. Some datasets may be inappropriate for methods requiring learning of the parameters of complex models because of the small number of available samples.

7.1. UCF

The UCF dataset [55],[3] provides only skeletal data. It provides the skeleton (15 joints) data for 16 actions performed 5 times by 16 individuals (13 males and 3 females, all ranging between ages 20 and 35). The action samples are in total 1280 with a temporal duration that ranges in [27, 229], and an average length of 66 ± 34 frames. The actions in this dataset are: *balance*, *climbladder*, *climbup*, *duck*, *hop*, *kick*, *leap*, *punch*, *run*, *stepback*, *stepfront*, *stepleft*, *stepright*, *twistleft*, *twistright*, and *vault*. The suggested validation protocol in [55] is based on a 4-folds cross-validation (4F-CV) that is 1/4 of the sequences is used for test and the remaining for training. However, other protocols have been adopted in literature.

7.2. MHAD

The Berkeley multimodal human action database (MHAD) [148],[4] contains data from a motion capture system, stereo cameras, depth sensors, accelerometers, and microphones. It provides about 660 motion sequences of 11 actions performed 5 times by 12 actors. The actions are: *jumping in place*, *jumping jacks*, *bending*, *punching*, *waving two hands*, *waving right hand*, *clapping throwing a ball*, *sit down and stand up*, *sit down*, and *stand up*. The subjects were instructed on the action to perform; however no specific details were given on how the action should be executed (i.e., performance style or speed). The Mo-Cap data include 3D position of 43 LED markers, which have been processed to get skeletal data of 30 joints. In [148], cross-subjects validation (CS-V) is adopted, where the action sequences of the first 7 subjects are used for training, and the sequences of the last 5 subjects are used in test.

7.3. MSRA3D

The MSRA3D dataset [54],[5] provides both skeletal and depth data. In particular, the dataset provides the skeleton (20 joints) for 20 actions performed 2–3 times by 10 subjects. It provides both 3D world coordinates and screen coordinates plus depth of the detected skeleton joints. The dataset contains skeleton sequences of the following actions: *high arm wave*, *horizontal arm wave*, *hammer*, *hand catch*, *forward punch*, *high throw*, *draw x*, *draw tick*, *draw circle*, *hand clap*, *two hand wave*, *side-boxing*, *bend*, *forward kick*, *side kick*, *jogging*, *tennis swing*, *tennis serve*, *golf swing*, *pickup* and *throw*. The subjects face the camera, and the actions cover various movements of arms, legs, torso and their combinations. If an action is performed by a single arm or leg, the subjects were advised to use their right arm or leg.

The lengths of the 567 available sequences ranges in [13, 76] with an average value of 39.6 ± 10. There are several validation protocols adopted with this dataset. In *Protocol* 1, a cross-subject validation over the entire set of classes is performed. This is an interesting kind of evaluation, due to the high number of classes in the dataset, but presents several challenges due to the small number of available sequences for training and the high inter-subjects variability. The validation protocol is described on the authors' website: 10 sequences have been filtered out because of the excessive noise on the skeletons; the subject splitting of the data in training and test set is as follows: subjects 1, 3, 5, 7, and 9 for training, the others in test (HS-V protocol).

The second kind of validation protocol splits the actions into 3 overlapping sub-sets of 8 classes each one. The first action set (AS1) includes the actions *horizontal arm wave*, *hammer*, *forward punch*, *high throw*, *hand clap*, *bend*, *tennis serve*, *pickup & throw*. The second action set (AS2) includes *high arm wave*, *hand catch*, *draw x*, *draw tick*, *draw circle*, *two hand wave*, *forward kick*, *side boxing*. The third action set (AS3) includes *high throw*, *forward kick*, *side kick*, *jogging*, *tennis swing*, *tennis serve*, *golf swing*, *pickup & throw*. AS1 and AS2 group actions that require similar movements, while AS3 groups more complex actions. Test on these subsets is done with different kinds of data splitting. In *Protocol* 2, a 3-fold cross-validation is adopted with 1/3 of the data used for testing the models and 2/3 for training purposes (3F-1:2). In *Protocol* 3, a 3-fold cross-validation is adopted with 2/3 of the data used for testing the models and 1/3 for training the models (3F-2:1). In *Protocol* 4, the most widely adopted, cross-subject validation with subjects 1, 3, 5, 7, and 9 for training, the others for test (HS-V protocol) is adopted.

The main challenge of this dataset is data corruption. Some sequences are very noisy and in some papers [100], corrupted sequences have been removed from the dataset. In other works, [64,99], some classes are totally ignored. Some works, such as [102,65], adopt a different splitting of the subjects. Because of all these different expedients, comparison over this dataset is difficult and may be confusing. We refer to [149] for further discussion of these issues.

7.4. UTKA

The UTKA dataset [58],[6] provides RGB, depth and skeletal data. The dataset provides the skeleton (20 joints) for 10 actions

Table 3
The most commonly adopted validation protocols.

Abbreviation	Description
LOOCV	Leave-one-out cross-validation (one sequence to test, the rest for training)
CS-LOOCV	Cross-subjects leave-one-out cross-validation (one subject to test, the rest for training)
4F-CV	4 folds cross-validation (1/4 of the data to test, 3/4 for training)
3F-1:2	3 folds cross-validation (1/3 of the data to test, 2/3 for training)
3F-2:1	3 folds cross-validation (2/3 of the data to test, 1/3 for training)
HS-V	Half subjects to test, the rest for training
CS-V	Some subjects to test, the rest for training

[2] http://vision.imar.ro/human3.6m/description.php
[3] http://www.cs.ucf.edu/smasood/research.html
[4] http://tele-immersion.citris-uc.org/berkeley_mhad
[5] http://research.microsoft.com/en-us/um/people/zliu/actionrecorsrc/
[6] http://cvrc.ece.utexas.edu/KinectDatasets/HOJ3D.html

performed twice by 10 subjects. The actions are: *walk, sit down, stand up, pick up, carry, throw, push, pull, wave and clap hands*. The dataset comprises 195 sequences. The lengths of the skeleton sequence ranges in $[5, 170]$ with an average value of 30.5 ± 20 frames. Experiments in [58] are performed in leave-one-out cross-validation (LOOCV) on a subset of manually selected joints *head, L/ R elbow, L/ R hands, L/ R knee, L/ R feet, hip center and L/ R hip*. In this dataset, one of the subject is left-handed and there is a significant variation among different realizations of the same action: some actors pick up objects with one hand, while others pick up the objects with both hands. The individuals can toss an object with either their right or left arm, producing different trajectories. Finally, actions have been taken from different views and, therefore, the body orientation varies.

7.5. MSR daily activity 3D

The MSR daily activity 3D dataset [112],[7] comprises 16 activities: *drink, eat, read book, call cellphone, write on a paper, use laptop, use vacuum cleaner, cheer up, sit still, toss paper, play game, lie down on sofa, walk, play guitar, stand up*, and *sit down*. Actions are performed twice by 10 different subjects: once in standing position, and once in sitting position. The dataset provides depth maps, skeleton joint positions (20 joints), and RGB video of 320 gesture samples. In some cases, in each frame more than a skeleton is detected. The authors of [112] suggest to use the first detected skeleton. Joint locations are represented both in terms of real world coordinate and normalized screen coordinates plus depth. The suggested protocol is cross-subject validation (HS-V) where subjects 1, 3, 5, 7, and 9 are used for training, the others in test.

7.6. Further benchmarks

Other datasets that have been sometimes used in literature are the Cornell activity dataset (CAD)-120 dataset and the Cornell activity dataset (CAD)-60 dataset, which are composed of a very limited number of sequences.

The *Cornell activity dataset* (*CAD*)-120 *dataset* [150],[8] provides 120 RGB-D videos of 10 long daily activities: *making cereal, taking medicine, stacking objects, unstacking objects, microwaving food, picking objects, cleaning objects, taking food, arranging objects*, and *having a meal*. Each activity may be composed of 10 different (labeled) sub-activities: *reaching, moving, pouring, eating, drinking, opening, placing, closing, scrubbing*, and *null*. The activities have been performed by 4 subjects (one left-handed). The subjects were only given a high-level description of the task, and were asked to perform the activities multiple times with different objects. They performed the activities through a long sequence of sub-activities, which varied from subject to subject significantly in terms of length of the sub-activities, order of the sub-activities as well as in the way they executed the task.

The camera was mounted so that the subject was in view (although the subject may not be facing the camera), but often there were significant occlusions of the body parts.

The *Cornell activity dataset* (*CAD*)-60 *dataset* [150] (see footnote 8) contains 60 RGB-D videos representing 12 activities *rinsing mouth*, *brushing teeth*, *wearing contact lens*, *talking on the phone*, *drinking water*, *opening pill container*, *cooking* (*chopping*), *cooking* (*stirring*), *talking on couch*, *relaxing on couch*, *writing on whiteboard*, and *working on computer*. The activities were performed by 4 different subjects in 5 different environments: office, kitchen, bedroom, bathroom, and living room. One of the subject is left-handed. When collecting the data, the subjects were given basic instructions on how to carry out the activity.

Validation protocols comprise cross subject analysis (that means leave-one-subject-out cross-validation) and in 2-fold cross validation where half of the data were used to train and the other half to test independently on the subject performing the activity. Data have been mirrored to train action models for both right and left handed subjects.

Another publicly available dataset is the *composable activities dataset* [116].[9] This dataset consists of 693 videos of activities in 16 classes performed by 14 actors. Each activity in the dataset is composed by a number of atomic actions. The total number of actions in the videos is 26. The number of actions in each activity ranges between 3 and 11. Up to date, we did not find other works adopting this dataset.

For completeness, we also report about the *Microsoft Research Cambridge*-12 (*MSRC*-12) *Kinect gesture* [151],[10] dataset. This dataset provides only the skeletons (20 joints) for 12 different actions performed by 30 people. The actions were performed based on different sets of instructions (descriptive text, images, and videos or their combinations) Each of the 594 sequences contains multiple repetitions of the same gestures for a total of 6243 samples.

The gestures can be categorized into two categories: Iconic gestures, which imbue a correspondence between the gesture and the reference, and Metaphoric gestures, which represent an abstract concept. The six iconic gestures in the dataset are *duck* (crouch/hide), *goggles* (put on night vision goggles), *shoot*, *throw*, *change weapon*, and *kick*. The six metaphoric gestures in this dataset are: *lift outstretched arms* (start system/music/raise volume), *push right* (navigate to next menu/move arm right), *wind it up*, *bow* (take a bow to end music session), *had enough* (protest the music), *beat both* (move up the tempo of the song/beat both arms).

The dataset has been conceived for action localization in the videos rather than for segmented action classification. Therefore the annotation consists of the timestamp when the action can be detected, but no information about the duration of the action is provided. However, in [100], further annotation for classification purposes (the segmented actions) is provided and made publicly available.[11] Based on such annotation, the length of each sequence ranges in $[14, 493]$ with an average length of 85 ± 31 frames. The validation protocols in [151] are leave-one-out cross-subject validation (CS-LOOCV), 50% subject split (HS-V), 3 folds cross-validation (1/3 to test and 2/3 to train, but also 2/3 to test and 1/3 to train). We believe that the annotation provided along with the dataset might have been used in different ways by works focusing on segmented action datasets, and it is unclear if the performance of such papers are comparable.

8. Comparison of methods at the state of the art

In this section we discuss the performance of the reviewed methods in 3D skeleton-based action classification. First, we make clear how the performance of each method is evaluated; later, we focus on methods that report experimental results on the most adopted datasets described in Section 7. We compare and discuss the results reported in literature from the point of view of the categorization introduced in Section 5. We also highlight the kind of classification framework used together with the proposed action representation to make sense of the informativeness of the

[7] http://research.microsoft.com/en-us/um/people/zliu/actionrecorsrc/
[8] http://pr.cs.cornell.edu/humanactivities/data.php
[9] http://web.ing.puc.cl/ialillo/ActionsCVPR2014
[10] http://research.microsoft.com/en-us/um/cambridge/projects/msrc12/
[11] http://www.eng.alexu.edu.eg/mehussein/msrc12_annot4rec/index.html

descriptors. Finally, we discuss latency in action recognition as introduced in [55].

8.1. Performance evaluation – accuracy in classification

The most commonly used approach for measuring the performance of a classification method is the average accuracy value, which is defined as the number of correctly classified sequences over the total number of sequences to classify. This accuracy value is averaged over several runs, especially in case of cross-validation or multiple runs with random initialization of the model parameters. The average accuracy value is equal to the average per-class accuracy value in case of balanced datasets, that is in dataset where the same number of samples is given for each class. Several insights can be gathered by the average per-class accuracy value that measures the proportion of correctly classified sequence within a class. Common practice is to present a confusion matrix that reports the per-class accuracy values along the main diagonal, and the per-class proportion of misclassified sequences in the extra-diagonal elements. Here, we only report the average accuracy value attained by each method on equal terms of validation protocol unless differently specified.

8.2. Datasets included in the comparison

Most of the methods at the state of the art reports accuracy values on no more than 2 datasets. The most used benchmarks are the UTKA, UCF, MSRA-3D, MHAD and MSRDA datasets. Among all these benchmarks, we note that MHAD is the only one providing Mo-Cap data, while UCF provides skeletons of 15 joints estimated from depth maps. The rest of the datasets provide skeletons of 20 joints estimated from depth maps (see Table 2 for details on the main characteristics of each dataset).

We are not reporting results on CAD-60 and CAD-120, which have a smaller number of sequences compared to the other datasets and have been less commonly adopted. We do not report results on the MSRC-12 dataset as well. Indeed different validation protocols have been applied when testing on the MSRC-12 dataset, and it is unclear how comparable these accuracy values are.

We also note that different validation protocols have been used when testing on the MSRA-3D datasets. It is already known that the choice of the validation protocol affects the reported accuracy values on the MSRA-3D dataset [149]. Hence, when reporting accuracy values on this dataset, we have highlighted whether a validation protocol different than the one proposed by the authors of [54] has been adopted.

8.3. Discussion

We have collected in Table 4 the accuracy values reported by methods at the state of the art on UCF, MHAD, MSRDA, UTKA and MSRA-3D. In this table, results on the MSRA-3D dataset are obtained by applying the protocol P1, that is, all action classes are used in the experiments. In Tables 5 and 6 we report the accuracy values on the MSRA-3D dataset when using the protocols described in Section 7.3 on different subsets of the action classes (protocols P2, P3 and P4).

The structure of the tables reflects the categorization we have proposed in Section 5. In particular, the first column of each table indicates the category to which the method belongs to: spatial descriptors (*S*), geometric descriptors (G), key-pose based descriptors (K), mined joints based descriptors (M), and finally dynamics-based descriptors (D).

Some methods such as [137,112,104] propose to use hybrid features where skeletal data and information extracted from depth maps and/or RGB video are jointly used for classifying the action sequence.

Table 4
Average accuracy value reported for skeleton-based action recognition. (S) indicates that the performance refers to skeletal data adoption but the cited work reports also performance on hybrid representation.

Cat.	Methods	Classifier	UCF	MHAD	MSRDA	UTKA	MSRA-3D All
S	[55]	Log. reg.	95.94%	–	–	–	65.7%
S	[98]	sHMM + CNN	–	–	–	–	82%
S	[107]	CNN	–	98.38%[a]	–	–	–
S	[102]	SVM	98.8%	–	–	–	91.4%[a]
S	[101]	KNN + Vote	98.50%	–	73.8%[b]	–	91.7%[b]
S	[135] (S)	SVM	97.07%	–	–	–	83.53%
S	[148]	K-SVM	–	79.93%	–	–	–
S	[137] (S)	Rnd forest	–	–	–	87.9%[b]	–
G	[34]	DTW+SVM	–	–	–	97.08%	89.48%
G	[105]	DP+KNN	–	–	–	91.5%	–
G	[114]	DP+KNN	99.2%	–	–	91.5%	–
K	[116]	Gen. model	–	–	–	–	89.5%[b]
K	[58]	sHMM	–	–	–	90.92%	–
K	[115]	SVM	–	–	–	–	90.56%[c]
M	[64]	SVM	–	95.37%	–	–	33.99%[d]
M	[124]	NBNN	–	–	70%	–	–
M	[103]	PLS-SVM	–	–	70%	–	91.5%
M	[103]	PLS-SVM+TP	–	–	73.1%	–	90.1%
M	[112] (S)	SVM	–	–	68%	–	88.2%
D	[99]	MKL-SVM	–	100%	–	–	90%[d]
D	[65]	SVM	97.91%[b]	–	–	88.5%	91.21%[d]
D	[62]	dHMM	–	–	–	86.7%	89.1%
D	[125]	dHMM	97.66%	–	–	–	89.23%

[a] Different splitting of the subjects.
[b] Different splitting of the data.
[c] Splitting of data undeclared.
[d] 17 classes.

Table 5
Performance on the MSRA3D using protocols P2 and P3.

Cat.	Methods	Classifier	AS 1 P2 (%)	AS 2 P2 (%)	AS 3 P2 (%)	AS 1 P3 (%)	AS 2 P3 (%)	AS 3 P3 (%)
S	[59]	NBNN	94.7	95.4	97.3	97.3	98.7	97.3
S	[102][a]	SVM	96.6	90.8	98	98.6	96	98.6
G	[105]	DP+KNN	90.3	91	98	93.4	93.9	98.6
K	[58]	sHMM	98.47	96.67	93.47	98.61	97.92	94.93

[a] Different splitting of the subjects.

Table 6
Performance on the MSRA3D using protocol P4. (S) indicates that the performance refers to skeletal data adoption but the cited work reports also performance on hybrid representation.

Cat.	Methods	Classifier	AS 1 P4 (%)	AS 2 P4 (%)	AS 3 P4 (%)
S	[59]	NBNN	74.5	76.1	96.4
S	[102][a]	SVM	88.4	91.6	100
S	[104] (S)	DTW+NN	88.83	85.01	94.22
G	[105]	DP+KNN	90.1	90.6	97.6
G	[34]	DTW+SVM	95.29	83.87	98.22
G	[100]	SVM	88.04	89.29	94.29
G	[117]	SVM+RF	96	57.1	97.3
G	[109]	SVM	88.39	86.61	94.59
K	[58]	sHMM	87.98	85.48	63.46
D	[125]	dHMM	90.29	95.15	93.29

[a] Different splitting of the subjects.

Whenever available, we report the performance of these methods when adopting only skeletal data. We highlight this eventuality by adding (S) near the reference in the second column. In the third column we indicate the adopted classification framework.

Among the works reported in this survey, the highest accuracy values are obtained on the MHAD dataset (up to 100% by [99]). We

believe that this is ascribable to the fact that Mo-Cap-based skeleton sequences are less noisy than the one estimated from depth maps and provided with the other datasets.

The second dataset where the highest accuracy values are reported is the UCF dataset (up to 99.2% by [114]), which provides an higher number of sequences for training the models and only moderately corrupted skeleton sequences.

On the MSRA-3D dataset, performance of most of the reported results is comparable (up to 91.5% by [103]). The comparison of results in Tables 5 and 6 highlights how the validation protocol can impact on the reported performance. Moreover, P4 is a cross-subjects validation protocol, which is the most suitable kind of validation for the problem at hand.

We note that MSRA-3D and MSRDA datasets have the most corrupted skeleton sequences. Fig. 4 shows some few examples of corrupted skeletons in the MSRA-3D dataset. The image is meant to give an idea of the level of corruption of the skeletal data. Due to failures of the skeleton tracking algorithm, in some sequences of the class *pickup & throw* the skeleton is missing (or it reduces to a point) as shown in the last image of Fig. 4.

Similar failures/corrupted data are also present in the MSRDA dataset, where the best accuracy value is of about 73.1% and is achieved by [103]. The excessive noise may have contributed to the low accuracy values achieved with skeleton-based action descriptors on the MSRDA dataset. We also note that the actions in the MSRDA are more complex than the ones in the other datasets and in general they require interactions with objects in the scene. Such kind of interactions are difficult to model/detect from skeletal data.

Unfortunately, the use of different benchmarks and protocols in experiments makes challenging to state which are the best methods. The datasets employed in all the reviewed works present also different characteristics and therefore each method has its own merits in dealing with the specific challenges in a dataset rather than in the other. Looking at the categories of methods, among the works adopting spatial descriptors (S), the most competitive seems to be the work in [102], where a skeleton sequence is represented as a spatio-temporal graph and wavelet-based features allow to account for different spatio-temporal scales. Even if the method achieves the highest accuracy value among the works adopting spatial descriptors on the UCF dataset, experiments on the MSRA-3D dataset have been performed with a different validation protocol and hence the comparison remains unclear.

Among the methods in Table 4 that adopt geometric descriptors, the best accuracy value on the UTKA dataset is of 97.08% and has been reported by [34] (this is also the best accuracy value obtained on the UTKA dataset in general). We recall from our discussion in Section 5 that in [34], the relative 3D geometry between different body parts is explicitly estimated and a sequence of skeletons is a curve in the Lie group manifold. The curves are further represented by means of the Fourier temporal pyramid (FTP). This suggests that the method might have a high computational complexity.

We believe that, in general, geometric descriptors look very promising even if more validation on other benchmarks are necessary to confirm this intuition.

Most of the mined-joint based descriptors have been tested on the MSRDA dataset. This is probably due to the nature of the sequences in the dataset: by learning the subsets of joints more relevant to an action it might be easier to discriminate among different interactions with the objects in the scene. Among these works, the most promising is probably [103] where an action is modeled as a set of histograms, each computed over the sequence on a specific feature and body joint; partial least squares is used to weight the importance of the joints.

As for the dynamics-based descriptors, all the reviewed methods have comparable performance. These methods look promising in that they are very competitive with the methods in the other categories across different datasets. Moreover, even if tested only on the 3D trajectories of the body joints, all these methods are general and might be applied on more complex 3D action representation, such as geometric descriptors of the skeleton sequences.

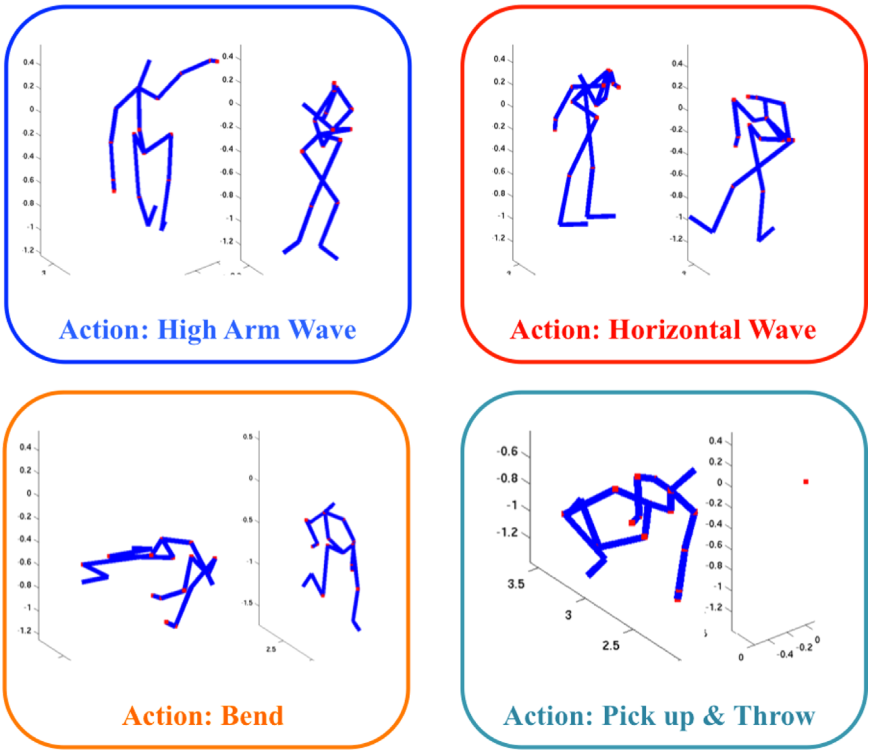

Fig. 4. Some samples from the MSRA-3D dataset where the skeleton is corrupted.

8.4. Action classification framework

As shown in Tables 4–6, most of the methods at the state of the art adopt discriminative approaches such as SVM or some variants to solve the 3D skeleton-based action classification [34,103,99]. Only few methods adopt generative models such as [116,58]. In particular, standard hidden Markov model (sHMM) is applied in [58,98].

Discriminative learning of generative models is presented in [130] (hierarchical Dirichlet process-hidden Markov model – HDP-HMM) and [62] (discriminative HMM – dHMM). Logistic regression is applied in [55], while a random forest is used in [137].

Several other methods apply non-parametric methods such as nearest neighbor (NN) [104], *K*-NN [105], or simple variants to improve the classification accuracy value [101,124,59].

Dynamic time warping (DTW) is adopted in [34,104] while a dynamic programming (DP) based distance is applied in [105].

Overall, the tables seem to confirm that on this kind of data and for the problem of 3D skeleton-based action classification, discriminative approaches outperform the generative ones.

Furthermore, we note that, in contrast to most of the above cited works, [97] tackles with a different and original problem, which is the detection of concurrent actions.

8.5. Latency in action recognition

Finally, we note that [55] proposes an alternative evaluation protocol for skeleton-based action classification based on recognition latency. This validation protocol attempts to measure how many frames are required to recognize an action, and it consists in measuring the accuracy in classification when partially observed action sequences are available. In particular, the accuracy values are measured considering temporal windows of increasing duration.

A few other works [135,114,65] have adopted this other validation protocol. Table 7 reports and compares the analyses conducted in these papers on the UCF dataset. In the table, columns indicate the length of the observed temporal window. We note that it is unclear if this evaluation can be comparable across datasets because of the following reasons: (1) there may be huge variations in the length of the action sequences, especially across classes. Hence, sequences in some classes may be fully observable after *N* frames, while sequences in the remaining classes are not; (2) it is unclear how the frame rate in data acquisition can impact on such kind of analysis across different benchmarks. A solution to the first issue might be to consider percentages of observed frames rather than absolute values of the number of observed frames. We further note that such kind of evaluation may pose some challenges for works such as [102,99,103,100], which use a pyramidal approach and need to see the whole sequence for recognizing the action.

9. Conclusions and future directions

In the past few years, thanks to the diffusion of cheap depth cameras and the recent success of works on skeleton estimation and tracking from depth maps, we have assisted to a proliferation of works on human action recognition from skeleton sequences. This survey has summarized the main technologies (hardware and software) necessary to put in place a system for 3D skeleton-based action classification that solve the problem of inferring the kind of action performed by a subject given in input a time series of 3D skeletons. From the analysis of a large portion of the relevant literature, we have highlighted the common practice of adopting data normalization techniques to account for different camera setting and biometric differences. We have also pointed out how some of the methods for skeleton normalization are related to a broader set of techniques for motion retargeting.

In this survey, we have reviewed several publicly available benchmarks for 3D action classification. Based on our analyses, we noted that the most adopted datasets for skeleton-based action classification are the UCF, MHAD, MSRDA, UTKA and MSRA-3D datasets. Such benchmarks show very different characteristics not only in terms of data acquisition technology (i.e., Mo-Cap vs. depth cameras), but also in terms of kind of collected data (for example, different number of body joints in the skeletons, availability of depth/RGB data, etc.) and action classes (i.e., complexity of the actions and number of classes). We have also observed that the skeletal data are corrupted by different levels of noise across the datasets and we show examples of corrupted skeletons.

Based on our analysis, the overall impression is that most of the benchmarks in literature are too small for appropriate training of complex models, and this aspect may affect the results reported in literature. We believe that, despite the number of publicly available datasets, there is still a lack of consolidated benchmarks and best practices in the adoption of suitable evaluation protocols, and a more exhaustive and wider dataset of 3D skeletal data for action classification is still missing. In fact, most of the datasets we have reviewed deals with simple actions, with the only exception of the MSRDA dataset. Moving towards more natural action recognition as well as spontaneous action recognition given skeletal data might be an interesting future direction. Considering the purposes of 3D action classification, we feel that the most suitable validation protocols should be based on cross-subject validation, especially when non-parametric approaches such as KNN are used. This would allow for testing how much an action descriptor can be robust to inter-subjects variations.

This survey also proposes a categorization of the skeleton-based action representation techniques according to the kind of information extracted from the skeleton sequences. In our categorization, the methods at the state of the art are classified in: *joint-based representations*, that include methods that try to capture the correlation among body joint locations in an action; *mined-joint based descriptors*, that try to learn which subsets of body joints is more informative to discriminate a given action from the other ones; and *dynamics-based descriptors*, that take advantage of the way 3D body joint locations move across time, and model an action as a set of 3D trajectories.

We have further grouped the *joint-based representations* into: *spatial descriptors*, that extract spatial features (such as distances, angles or covariance matrices) from the set of joints in a skeleton, in some cases considering also the temporal extent of the action; *geometric descriptors*, that model – more or less explicitly – the relative position of different body parts in the skeleton; and *key-pose*

Table 7
Latency-based analysis on the UCF dataset (average classification accuracy value in %). The number on each column indicates the number of observed frames.

Cat.	Methods	Classifier	10	15	20	25	30	40	45	50	55	60
S	[55]	Log. regr.	13.91	36.95	64.77	81.56	90.55	95.16	95.78	96.1	96.84	95.94
S	[135]	SVM	–	–	47.63	72.75	86.54	–	96.05	96.05	96.84	97.07
G	[114]	DP+KNN	30.5	60.9	79.9	91.1	95.1	97.8	–	–	–	99.2
D	[65]	SVM	21.87	49.37	74.37	86.87	92.08	97.29	–	–	–	97.91

based descriptors, that learn a dictionary of informative pose representations and describe an action in terms of the estimated key-poses.

By the analysis of the reviewed work, we have found that very different (sometimes even arbitrary) validation protocols have been used when testing the methods making more difficult the comparison of different approaches. Nonetheless the variations in the adopted protocols, benchmarks and classification frameworks have made difficult the comparison among different approaches, overall the impression is that the most convenient and informative approaches fall in the categories of geometric and dynamics-based descriptors. Therefore, methods mixing the two approaches may help to obtain even higher accuracy values in classification in future works.

Despite the significant efforts and recent progress in skeleton-based action classification, there are still some issues that have not been completely addressed in the general case. For example, many methods do not explicitly account for commonalities and movement sharing among action classes. Instead it is natural to think about an action as a sequence of atomic movements that might be shared among different classes. As an example, in the MSRA-3D dataset, the action *bend* is mostly a sub-action of the action *pickup & throw*. This example also reflects the lack of a generally accepted definition of action in comparison to activity (that might be possibly defined as a composition of actions).

Another issue that seems to be ignored by most of the reviewed methods, especially the one using pyramidal approaches and holistic descriptors, is that actions can be performed with a different number of repetitions of atomic movements. For example, in the action *waving hands*, the hands can be waved several times; considering this issue in the action representation might possibly mitigate the problem.

Another direction to investigate might be the recognition of concurrent actions as proposed in [97], which could help human behavior understanding in collaborative tasks. The problem here is not only that of recognizing which actions are performed by a group of subjects, but also which actions are *socially* correlated, by recognizing groups of socially engaged subjects.

Finally, in this survey we focused only on segmented action classification. However, it is unclear on what extent some of the reviewed action representations might be adopted in online action recognition and action detection frameworks. As an example, methods involving wavelets decompositions, pyramidal approaches or holistic descriptors may result in higher computational complexity when adopted in a framework for action localization.

In conclusion, we believe that reasoning on skeletal data offers a very convenient opportunity to study well-established frameworks and to extend our understanding about human action recognition. Such kind of studies could greatly enhance both gaming applications and human–computer interaction [152]. In this sense, online learning of subject-specialized models and/or adaptation of prior models to subjects could be extremely helpful to improve specific visual human–computer interface.

Conflict of interest

None declared.

Acknowledgments

We thank the editor and the anonymous reviewers for their insightful and sharp comments, and for their suggestions for improvements of this paper. This work was partially supported by Italian MIUR SINTESYS – Security and INTElligence SYStem Grant PON0101687.

References

[1] S. Kwak, B. Han, J. Han, Scenario-based video event recognition by constraint flow, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 3345–3352, http://dx.doi.org/10.1109/CVPR.2011.5995435.

[2] U. Gaur, Y. Zhu, B. Song, A. Roy-Chowdhury, A string of feature graphs model for recognition of complex activities in natural videos, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Barcelona, Spain, 2011, pp. 2595–2602, http://dx.doi.org/10.1109/ICCV.2011.6126548.

[3] S. Park, J. Aggarwal, Recognition of two-person interactions using a hierarchical Bayesian network, in: First ACM SIGMM International Workshop on Video surveillance, ACM, Berkeley, California, 2003, pp. 65–76, http://dx.doi.org/10.1145/982452.982461.

[4] I. Junejo, E. Dexter, I. Laptev, P. Pérez, View-independent action recognition from temporal self-similarities, IEEE Trans. Pattern Anal. Mach. Intell. 33 (1) (2011) 172–185, http://dx.doi.org/10.1109/TPAMI.2010.68.

[5] Z. Duric, W. Gray, R. Heishman, F. Li, A. Rosenfeld, M. Schoelles, C. Schunn, H. Wechsler, Integrating perceptual and cognitive modeling for adaptive and intelligent human–computer interaction, Proc. IEEE 90 (2002) 1272–1289, http://dx.doi.org/10.1109/JPROC.2002.801449.

[6] Y.-J. Chang, S.-F. Chen, J.-D. Huang, A Kinect-based system for physical rehabilitation: a pilot study for young adults with motor disabilities, Res. Dev. Disabil. 32 (6) (2011) 2566–2570, http://dx.doi.org/10.1016/j.ridd.2011.07.002.

[7] A. Thangali, J.P. Nash, S. Sclaroff, C. Neidle, Exploiting phonological constraints for handshape inference in ASL video, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 521–528, http://dx.doi.org/10.1109/CVPR.2011.5995718.

[8] A. Thangali Varadaraju, Exploiting phonological constraints for handshape recognition in sign language video (Ph.D. thesis), Boston University, MA, USA, 2013.

[9] H. Cooper, R. Bowden, Large lexicon detection of sign language, in: Proceedings of International Workshop on Human–Computer Interaction (HCI), Springer, Berlin, Heidelberg, Beijing, P.R. China, 2007, pp. 88–97.

[10] J.M. Rehg, G.D. Abowd, A. Rozga, M. Romero, M.A. Clements, S. Sclaroff, I. Essa, O.Y. Ousley, Y. Li, C. Kim, et al., Decoding children's social behavior, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, Oregon, 2013, pp. 3414–3421, http://dx.doi.org/10.1109/CVPR.2013.438.

[11] L. Lo Presti, S. Sclaroff, A. Rozga, Joint alignment and modeling of correlated behavior streams, in: Proceedings of International Conference on Computer Vision-Workshops (ICCVW), Sydney, Australia, 2013, pp. 730–737, http://dx.doi.org/10.1109/ICCVW.2013.100.

[12] H. Moon, R. Sharma, N. Jung, Method and system for measuring shopper response to products based on behavior and facial expression, US Patent 8,219,438, July 10, 2012 ⟨http://www.google.com/patents/US8219438⟩.

[13] T.B. Moeslund, E. Granum, A survey of computer vision-based human motion capture, Comput. Vis. Image Underst. 81 (3) (2001) 231–268, http://dx.doi.org/10.1006/cviu.2000.0897.

[14] S. Mitra, T. Acharya, Gesture recognition, a survey, IEEE Trans. Syst. Man Cybern. Part C: Appl. Rev. 37 (3) (2007) 311–324, http://dx.doi.org/10.1109/TSMCC.2007.893280.

[15] R. Poppe, A survey on vision-based human action recognition, Image Vis. Comput. 28 (6) (2010) 976–990, http://dx.doi.org/10.1016/j.imavis.2009.11.014.

[16] D. Weinland, R. Ronfard, E. Boyer, A survey of vision-based methods for action representation, segmentation and recognition, Comput. Vis. Image Underst. 115 (2) (2011) 224–241, http://dx.doi.org/10.1016/j.cviu.2010.10.002.

[17] M. Ziaeefar, R. Bergevin, Semantic human activity recognition: a literature review, Pattern Recognit. 8 (48) (2015) 2329–2345, http://dx.doi.org/10.1016/j.patcog.2015.03.006.

[18] G. Guo, A. Lai, A survey on still image based human action recognition, Pattern Recognit. 47 (10) (2014) 3343–3361, http://dx.doi.org/10.1016/j.patcog.2014.04.018.

[19] C.H. Lim, E. Vats, C.S. Chan, Fuzzy human motion analysis: a review, Pattern Recognit. 48 (5) (2015) 1773–1796, http://dx.doi.org/10.1016/j.patcog.2014.11.016.

[20] M. Andriluka, S. Roth, B. Schiele, Pictorial structures revisited: people detection and articulated pose estimation, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Miami Beach, Florida, 2009, pp. 1014–1021, http://dx.doi.org/10.1109/CVPRW.2009.5206754.

[21] Y. Yang, D. Ramanan, Articulated pose estimation with flexible mixtures-of-parts, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 1385–1392, http://dx.doi.org/10.1109/CVPR.2011.5995741.

[22] D. Ramanan, D.A. Forsyth, A. Zisserman, Strike a pose: tracking people by finding stylized poses, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, IEEE, San Diego, CA, USA, 2005, pp. 271–278, http://dx.doi.org/10.1109/CVPR.2005.335.

[23] L. Bourdev, J. Malik, Poselets: body part detectors trained using 3D human pose annotations, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Kyoto, Japan, 2009, pp. 1365–1372, http://dx.doi.org/10.1109/ICCV.2009.5459303.

[24] D. Tran, D. Forsyth, Improved human parsing with a full relational model, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Crete, Greece, 2010, pp. 227–240.

[25] N. Ikizler, D. Forsyth, Searching video for complex activities with finite state models, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Minneapolis, Minnesota, 2007, pp. 1–8, http://dx.doi.org/10.1109/CVPR.2007.383168.

[26] F. Lv, R. Nevatia, Single view human action recognition using key pose matching and Viterbi path searching, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Minneapolis, Minnesota, 2007, pp. 1–8.

[27] N. Ikizler, P. Duygulu, Human action recognition using distribution of oriented rectangular patches, in: Proceedings of Workshop on Human Motion Understanding, Modeling, Capture and Animation, Springer, Rio de Janeiro, Brazil, 2007, pp. 271–284.

[28] M. Brand, N. Oliver, A. Pentland, Coupled hidden Markov models for complex action recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, San Juan, Puerto Rico, 1997, pp. 994–999.

[29] H. Wang, A. Kläser, C. Schmid, C.-L. Liu, Dense trajectories and motion boundary descriptors for action recognition, Int. J. Comput. Vis. 103 (1) (2013) 60–79, http://dx.doi.org/10.1007/s11263-012-0594-8.

[30] J.C. Niebles, H. Wang, L. Fei-Fei, Unsupervised learning of human action categories using spatial–temporal words, Int. J. Comput. Vis. 79 (3) (2008) 299–318, http://dx.doi.org/10.1007/s11263-007-0122-4.

[31] G. Johansson, Visual perception of biological motion and a model for its analysis, Percept. Psychophys. 14 (2) (1973) 201–211.

[32] S. Sadanand, J.J. Corso, Action bank: a high-level representation of activity in video, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Providence, Rhode Island, 2012, pp. 1234–1241, http://dx.doi.org/10.1109/CVPR.2012.6247806.

[33] A. Ciptadi, M.S. Goodwin, J.M. Rehg, Movement pattern histogram for action recognition and retrieval, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Zurich, 2014, pp. 695–710, http://dx.doi.org/10.1007/978-3-319-10605-2_45.

[34] R. Vemulapalli, F. Arrate, R. Chellappa, Human action recognition by representing 3D skeletons as points in a Lie Group, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, Ohio. 2014, pp. 588–595, http://dx.doi.org/10.1109/CVPR.2014.82.

[35] L. Sigal, Human pose estimation, Comput. Vis.: A Ref. Guide (2014) 362–370.

[36] K. Mikolajczyk, B. Leibe, B. Schiele, Multiple object class detection with a generative model, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, IEEE, New York, 2006, pp. 26–36.

[37] P. Viola, M.J. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Nice, France, 2003, pp. 734–741.

[38] P.F. Felzenszwalb, D.P. Huttenlocher, Pictorial structures for object recognition, Int. J. Comput. Vis. 61 (1) (2005) 55–79, http://dx.doi.org/10.1023/B:VISI.0000042934.15159.49.

[39] V. Ferrari, M. Marin-Jimenez, A. Zisserman, Progressive search space reduction for human pose estimation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Anchorage, Alaska, 2008, pp. 1–8, http://dx.doi.org/10.1109/CVPR.2008.4587468.

[40] D. Ramanan, Learning to parse images of articulated objects, in: Advances in Neural Information Processing Systems 134 (2006).

[41] A. Klaser, M. Marszałek, C. Schmid, A spatio-temporal descriptor based on 3d-gradients, in: Proceedings of British Machine Vision Conference (BMVC), BMVA Press, Leeds, UK. 2008, p. 275:1.

[42] L. Wang, Y. Wang, T. Jiang, D. Zhao, W. Gao, Learning discriminative features for fast frame-based action recognition, Pattern Recognit. 46 (7) (2013) 1832–1840, http://dx.doi.org/10.1016/j.patcog.2012.08.016.

[43] A. Gilbert, J. Illingworth, R. Bowden, Fast realistic multi-action recognition using mined dense spatio-temporal features, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Kyoto, Japan, 2009, pp. 925–931, http://dx.doi.org/10.1109/ICCV.2009.5459335.

[44] J. Liu, J. Luo, M. Shah, Recognizing realistic actions from videos in the wild, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Miami Beach, Florida, 2009, pp. 1996–2003.

[45] K. Soomro, A.R. Zamir, M. Shah, Ucf101: a dataset of 101 human actions classes from videos in the wild, arXiv preprint arXiv:1212.0402.

[46] K.K. Reddy, M. Shah, Recognizing 50 human action categories of web videos, Mach. Vis. Appl. 24 (5) (2013) 971–981.

[47] J. Cho, M. Lee, H.J. Chang, S. Oh, Robust action recognition using local motion and group sparsity, Pattern Recognit. 47 (5) (2014) 1813–1825, http://dx.doi.org/10.1016/j.patcog.2013.12.004.

[48] L. Liu, L. Shao, F. Zheng, X. Li, Realistic action recognition via sparsely-constructed gaussian processes, Pattern Recognit. 47 (12) (2014) 3819–3827, http://dx.doi.org/10.1016/j.patcog.2014.07.006.

[49] M. Hoai, Z.-Z. Lan, F. De la Torre, Joint segmentation and classification of human actions in video, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 3265–3272, http://dx.doi.org/10.1109/CVPR.2011.5995470.

[50] C.-Y. Chen, K. Grauman, Efficient activity detection with max-subgraph search, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Providence, Rhode Island, 2012, pp. 1274–1281, http://dx.doi.org/10.1109/CVPR.2012.6247811.

[51] A. Gaidon, Z. Harchaoui, C. Schmid, Temporal localization of actions with actoms, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2782–2795, http://dx.doi.org/10.1109/TPAMI.2013.65.

[52] D. Gong, G. Medioni, X. Zhao, Structured time series analysis for human action segmentation and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 36 (7) (2014) 1414–1427, http://dx.doi.org/10.1109/TPAMI.2013.244.

[53] K.N. Tran, I.A. Kakadiaris, S.K. Shah, Part-based motion descriptor image for human action recognition, Pattern Recognit. 45 (7) (2012) 2562–2572, http://dx.doi.org/10.1016/j.patcog.2011.12.028.

[54] W. Li, Z. Zhang, Z. Liu, Action recognition based on a bag of 3D points, in: Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, San Francisco, CA, USA, 2010, pp. 9–14, http://dx.doi.org/10.1109/CVPRW.2010.5543273.

[55] S.Z. Masood, C. Ellis, M.F. Tappen, J.J. LaViola, R. Sukthankar, Exploring the trade-off between accuracy and observational latency in action recognition, Int. J. Comput. Vis. 101 (3) (2013) 420–436, http://dx.doi.org/10.1007/s11263-012-0550-7.

[56] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, R. Moore, Real-time human pose recognition in parts from single depth images, Commun. ACM 56 (1) (2013) 116–124, http://dx.doi.org/10.1145/2398356.2398381.

[57] S. Litvak, Learning-based pose estimation from depth maps, US Patent 8,582,867, November 12, 2013.

[58] L. Xia, C.-C. Chen, J. Aggarwal, View invariant human action recognition using histograms of 3D joints, in: Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Providence, Rhode Island, 2012, pp. 20–27, http://dx.doi.org/10.1109/CVPRW.2012.6239233.

[59] X. Yang, Y. Tian, Eigenjoints-based action recognition using Naive-Bayes-Nearest-Neighbor, in: Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Providence, Rhode Island, 2012, pp. 14–19, http://dx.doi.org/10.1109/CVPRW.2012.6239232.

[60] O. Oreifej, Z. Liu, W. Redmond, HON4D: histogram of oriented 4D normals for activity recognition from depth sequences, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), Portland, Oregon, 2013, pp. 716–723, http://dx.doi.org/10.1109/CVPR.2013.98.

[61] A. Yao, J. Gall, G. Fanelli, L.J. Van Gool, Does human action recognition benefit from pose estimation? in: Proceedings of the British Machine Vision Conference (BMVC), vol. 3, BMVA Press, Dundee, UK, 2011, pp. 67.1–67.11, http://dx.doi.org/10.5244/C.25.67.

[62] L. Lo Presti, M. La Cascia, S. Sclaroff, O. Camps, Gesture modeling by Hanklet-based hidden Markov model, in: D. Cremers, I. Reid, H. Saito, M.-H. Yang (Eds.), Proceedings of Asian Conference on Computer Vision (ACCV 2014), Lecture Notes in Computer Science, Springer International Publishing, Singapore, 2015, pp. 529–546, http://dx.doi.org/10.1007/978-3-319-16811-1_35.

[63] C. Wang, Y. Wang, A.L. Yuille, An approach to pose-based action recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, Oregon, 2013, pp. 915–922, http://dx.doi.org/10.1109/CVPR.2013.123.

[64] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, R. Bajcsy, Sequence of the most informative joints (SMIJ): a new representation for human skeletal action recognition, J. Vis. Commun. Image Represent. 25 (1) (2014) 24–38, http://dx.doi.org/10.1016/j.jvcir.2013.04.007.

[65] R. Slama, H. Wannous, M. Daoudi, A. Srivastava, Accurate 3D action recognition using learning on the Grassmann manifold, Pattern Recognit. 48 (2) (2015) 556–567, http://dx.doi.org/10.1016/j.patcog.2014.08.011.

[66] L. Chen, H. Wei, J. Ferryman, A survey of human motion analysis using depth imagery, Pattern Recognit. Lett. 34 (15) (2013) 1995–2006, http://dx.doi.org/10.1016/j.patrec.2013.02.006.

[67] J. Aggarwal, L. Xia, Human activity recognition from 3D data: a review, Pattern Recognit. Lett. 48 (2014) 70–80, http://dx.doi.org/10.1016/j.patrec.2014.04.011.

[68] D. Murray, J.J. Little, Using real-time stereo vision for mobile robot navigation, Auton. Robots 8 (2) (2000) 161–171.

[69] I. Infantino, A. Chella, H. Dindo, I. Macaluso, Visual control of a robotic hand, in: Proceedings of International Conference on Intelligent Robots and Systems (IROS), vol. 2, IEEE, Las Vegas, CA, USA, 2003, pp. 1266–1271, http://dx.doi.org/10.1109/IROS.2003.1248819.

[70] A. Chella, H. Dindo, I. Infantino, I. Macaluso, A posture sequence learning system for an anthropomorphic robotic hand, Robot. Auton. Syst. 47 (2) (2004) 143–152, http://dx.doi.org/10.1016/j.robot.2004.03.008.

[71] P. Henry, M. Krainin, E. Herbst, X. Ren, D. Fox, RGB-D mapping: using depth cameras for dense 3D modeling of indoor environments, in: Experimental Robotics, Springer Tracts in Advanced Robotics, vol. 79, Citeseer, Springer, Berlin, Heidelberg, 2014, pp. 477–491, http://dx.doi.org/10.1007/978-3-642-28572-1_33.

[72] J.C. Carr, R.K. Beatson, J.B. Cherrie, T.J. Mitchell, W.R. Fright, B.C. McCallum, T. R. Evans, Reconstruction and representation of 3D objects with radial basis functions, in: Proceedings of Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), ACM, Los Angeles, CA, USA, 2001, pp. 67–76, http://dx.doi.org/10.1145/383259.383266.

[73] V. Kolmogorov, R. Zabih, Multi-camera scene reconstruction via graph cuts, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Copenhagen, Denmark, 2002, pp. 82–96.
[74] Microsoft kinect sensor ⟨http://www.microsoft.com/en-us/kinectforwindows/⟩.
[75] E. Trucco, A. Verri, Introductory Techniques for 3-D Computer Vision, vol. 201, Prentice Hall, Englewood Cliffs, 1998.
[76] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, Int. J. Comput. Vis. 74 (1–3) (2002) 7–42.
[77] P. Fua, A parallel stereo algorithm that produces dense depth maps and preserves image features, Mach. Vis. Appl. 6 (1) (1993) 35–49, http://dx.doi.org/10.1007/BF01212430.
[78] S. Foix, G. Alenya, C. Torras, Lock-in time-of-flight (tof) cameras: a survey, IEEE Sens. J. 11 (9) (2011) 1917–1926.
[79] D. Scharstein, R. Szeliski, High-accuracy stereo depth maps using structured light, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, IEEE, Madison, Wisconsin, 2003, p. I-195.
[80] P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained, multiscale, deformable part model, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Anchorage, Alaska, 2008, pp. 1–8, http://dx.doi.org/10.1109/CVPR.2008.4587597.
[81] J. Shen, W. Yang, Q. Liao, Part template: 3D representation for multiview human pose estimation, Pattern Recognit. 46 (7) (2013) 1920–1932, http://dx.doi.org/10.1016/j.patcog.2013.01.001.
[82] M. Ye, X. Wang, R. Yang, L. Ren, M. Pollefeys, Accurate 3d pose estimation from a single depth image, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Barcelona, Spain, 2011, pp. 731–738.
[83] M.A. Fischler, R.A. Elschlager, The representation and matching of pictorial structures, IEEE Trans. Comput. 22 (1) (1973) 67–92, http://dx.doi.org/10.1109/T-C.1973.223602.
[84] M. W. Lee, I. Cohen, Proposal maps driven MCMC for estimating human body pose in static images, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, IEEE, Washington, DC, 2004, p. II-334.
[85] G. Mori, X. Ren, A.A. Efros, J. Malik, Recovering human body configurations: combining segmentation and recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, IEEE, Washington, DC, 2004, p. II-326.
[86] X. Ren, A. C. Berg, J. Malik, Recovering human body configurations using pairwise constraints between parts, in: Proceedings of International Conference on Computer Vision (ICCV), vol. 1, IEEE, Beijing, P.R. China, 2005, pp. 824–831.
[87] T.-P. Tian, S. Sclaroff, Fast globally optimal 2d human detection with loopy graph models, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, San Francisco, CA, USA, 2010, pp. 81–88.
[88] B. Sapp, A. Toshev, B. Taskar, Cascaded models for articulated pose estimation, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Crete, Greece, 2010, pp. 406–420.
[89] Y. Wang, D. Tran, Z. Liao, Learning hierarchical poselets for human parsing, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 1705–1712.
[90] M.P. Kumar, A. Zisserman, P.H. Torr, Efficient discriminative learning of parts-based models, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Kyoto, Japan, 2009, pp. 552–559.
[91] S.S. SDK, Openni 2, openNI 2 SDK Binaries ⟨http://structure.io/openni⟩, 2014.
[92] M. Gleicher, Retargetting motion to new characters, in: Proceedings of Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), ACM, Orlando, Florida, USA, 1998, pp. 33–42, http://dx.doi.org/10.1145/280814.280820.
[93] C. Hecker, B. Raabe, R.W. Enslow, J. DeWeese, J. Maynard, K. van Prooijen, Real-time motion retargeting to highly varied user-created morphologies, ACM Trans. Graph. 27 (3) (2008) 27, http://dx.doi.org/10.1145/1399504.1360626.
[94] M. Gleicher, Comparing constraint-based motion editing methods, Graph. Models 63 (2) (2001) 107–134, http://dx.doi.org/10.1006/gmod.2001.0549.
[95] R. Kulpa, F. Multon, B. Arnaldi, Morphology-independent representation of motions for interactive human-like animation, Comput. Graph. Forum 24 (3) (2005) 343–351, http://dx.doi.org/10.1111/j.1467-8659.2005.00859.x.
[96] P. Baerlocher, R. Boulic, An inverse kinematics architecture enforcing an arbitrary number of strict priority levels, Vis. Comput. 20 (6) (2004) 402–417, http://dx.doi.org/10.1007/s00371-004-0244-4.
[97] P. Wei, N. Zheng, Y. Zhao, S.-C. Zhu, Concurrent action detection with structural prediction, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Sydney, Australia, 2013, pp. 3136–3143.
[98] D. Wu, L. Shao, Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, Ohio. 2014, pp. 724–731.
[99] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, R. Vidal, Bio-inspired dynamic 3D discriminative skeletal features for human action recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshop (CVPRW), IEEE, Portland, Oregon, 2013, pp. 471–478, http://dx.doi.org/10.1109/CVPRW.2013.153.
[100] M.E. Hussein, M. Torki, M.A. Gowayyed, M. El-Saban, Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations, in: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), AAAI Press, Beijing, P.R. China, 2013, pp. 2466–2472.
[101] M. Zanfir, M. Leordeanu, C. Sminchisescu, The moving pose: an efficient 3d kinematics descriptor for low-latency action recognition and detection, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Sydney, Australia, 2013, pp. 2752–2759.
[102] T. Kerola, N. Inoue, K. Shinoda, Spectral graph skeletons for 3D action recognition, in: Proceedings of Asian Conference on Computer Vision (ACCV), Springer, Singapore, 2014, pp. 1–16.
[103] A. Eweiwi, M.S. Cheema, C. Bauckhage, J. Gall, Efficient pose-based action recognition, in: Proceedings of Asian Conference on Computer Vision (ACCV), Springer, Singapore, 2014, pp. 1–16.
[104] A.A. Chaaraoui, J.R. Padilla-López, F. Flórez-Revuelta, Fusion of skeletal and silhouette-based features for human action recognition with RGB-D devices, in: Proceedings of International Conference on Computer Vision Workshops (ICCVW), IEEE, Sydney, Australia, 2013, pp. 91–97, http://dx.doi.org/10.1109/ICCVW.2013.19.
[105] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, A. Del Bimbo, Space–time pose representation for 3D human action recognition, in: Proceedings of the International Conference on Image Analysis and Processing (ICIAP), Springer, Naples, Italy, 2013, pp. 456–464, http://dx.doi.org/10.1007/978-3-642-41190-849.
[106] D.K. Hammond, P. Vandergheynst, R. Gribonval, Wavelets on graphs via spectral graph theory, Appl. Comput. Harmon. Anal. 30 (2) (2011) 129–150.
[107] E.P. Ijjina, C.K. Mohan, Human action recognition based on MOCAP information using convolution neural networks, in: Proceedings of International Conference on Machine Learning and Applications (ICMLA), IEEE, Detroit Michigan, 2014, pp. 159–164, http://dx.doi.org/10.1109/ICMLA.2014.30.
[108] M. Müller, T. Röder, M. Clausen, Efficient content-based retrieval of motion capture data, ACM Trans. Graph. 24 (3) (2005) 677–685, http://dx.doi.org/10.1145/1186822.1073247.
[109] G. Evangelidis, G. Singh, R. Horaud, et al., Skeletal quads: human action recognition using joint quadruples, in: Proceedings of International Conference on Pattern Recognition (ICPR), IEEE, Stockholm, Sweden, 2014, pp. 4513–4518, http://dx.doi.org/10.1109/ICPR.2014.772.
[110] T. Jaakkola, D. Haussler, et al., Exploiting generative models in discriminative classifiers, in: Advances in Neural Information Processing Systems, 1999, pp. 487–493.
[111] J.E. Humphreys, Introduction to Lie Algebras and Representation Theory, vol. 9, Springer Science & Business Media, New York, 1972.
[112] J. Wang, Z. Liu, Y. Wu, J. Yuan, Mining actionlet ensemble for action recognition with depth cameras, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Providence, Rhode Island, 2012, pp. 1290–1297, http://dx.doi.org/10.1109/CVPR.2012.6247813.
[113] Z. Shao, Y. Li, Integral invariants for space motion trajectory matching and recognition, Pattern Recognit. 48 (8) (2015) 2418–2432, http://dx.doi.org/10.1016/j.patcog.2015.02.029.
[114] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, A. Del Bimbo, 3-D human action recognition by shape analysis of motion trajectories on Riemannian manifold, IEEE Trans. Cybern. 45 (7) (2015) 1340–1352.
[115] M. Barnachon, S. Bouakaz, B. Boufama, E. Guillou, Ongoing human action recognition with motion capture, Pattern Recognit. 47 (1) (2014) 238–247, http://dx.doi.org/10.1016/j.patcog.2013.06.020.
[116] I. Lillo, A. Soto, J.C. Niebles, Discriminative hierarchical modeling of spatio-temporally composable human activities, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, Ohio. 2014, pp. 812–819.
[117] L. Miranda, T. Vieira, D. Martínez, T. Lewiner, A.W. Vieira, M.F. Campos, Online gesture recognition from pose kernel learning and decision forests, Pattern Recognit. Lett. 39 (2014) 65–73.
[118] M. Raptis, D. Kirovski, H. Hoppe, Real-time classification of dance gestures from skeleton animation, in: Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, ACM, Hong Kong, 2011, pp. 147–156.
[119] M. Barker, W. Rayens, Partial least squares for discrimination, J. Chemom. 17 (3) (2003) 166–173.
[120] R. Rosipal, L.J. Trejo, Kernel partial least squares regression in reproducing kernel Hilbert space, J. Mach. Learn. Res. 2 (2002) 97–123.
[121] P. Climent-Pérez, A.A. Chaaraoui, J.R. Padilla-López, F. Flórez-Revuelta, Optimal joint selection for skeletal data from rgb-d devices using a genetic algorithm, in: Advances in Computational Intelligence, Springer, Tenerife - Puerto de la Cruz, Spain, 2013, pp. 163–174, http://dx.doi.org/10.1007/978-3-642-37798-3_15.
[122] G. Dong, J. Li, Efficient mining of emerging patterns: discovering trends and differences, in: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Diego, CA, USA, 1999, pp. 43–52.
[123] F.R. Bach, G.R. Lanckriet, M.I. Jordan, Multiple kernel learning, conic duality, and the SMO algorithm, in: Proceedings of International Conference on Machine Learning (ICML), ACM, Alberta, Canada, 2004, p. 6.
[124] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, P. Pala, Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Portland, Oregon, 2013, pp. 479–485.
[125] L. Lo Presti, M. La Cascia, S. Sclaroff, O. Camps, Hankelet-based dynamical systems modeling for 3D action recognition, in: Image and Vision Computing, Elsevier, 44 (2015), 29–43, http://dx.doi.org/10.1016/j.imavis.2015.09.007 ⟨http://www.sciencedirect.com/science/article/pii/S02628%85615001134⟩.

[126] B. Li, O.I. Camps, M. Sznaier, Cross-view activity recognition using Hankelets, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Providence, Rhode Island, 2012, pp. 1362–1369, http://dx.doi.org/10.1109/CVPR.2012.6247822.
[127] B. Li, M. Ayazoglu, T. Mao, O.I. Camps, M. Sznaier, Activity recognition using dynamic subspace angles, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, 2011, pp. 3193–3200, http://dx.doi.org/10.1109/CVPR.2011.5995672.
[128] A.M. Lehrmann, P.V. Gehler, S. Nowozin, Efficient nonlinear Markov models for human motion, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, Ohio. 2014, pp. 1314–1321.
[129] C. Meek, D.M. Chickering, D. Heckerman, Autoregressive tree models for time-series analysis, in: Proceedings of the Second International SIAM Conference on Data Mining, SIAM, Toronto, Canada, 2002, pp. 229–244.
[130] N. Raman, S.J. Maybank, Action classification using a discriminative multilevel HDP-HMM, Neurocomputing 154 (2015): 149-161.
[131] J. Sung, C. Ponce, B. Selman, A. Saxena, Unstructured human activity detection from RGBD images, in: Proceedings of International Conference on Robotics and Automation (ICRA), IEEE, St. Paul, Minnesota, 2012, pp. 842–849, http://dx.doi.org/10.1109/ICRA.2012.6224591.
[132] J. Wang, Z. Liu, J. Chorowski, Z. Chen, Y. Wu, Robust 3D action recognition with Random Occupancy Patterns, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Florence, Italy, 2012, pp. 872–885, http://dx.doi.org/10.1007/978-3-642-33709-362.
[133] A.W. Vieira, E.R. Nascimento, G.L. Oliveira, Z. Liu, M.F. Campos, STOP: space–time occupancy patterns for 3D action recognition from depth map sequences, Prog. Pattern Recognit. Image Anal. Comput. Vis. Appl. (2012) 252–259, http://dx.doi.org/10.1007/978-3-642-33275-331.
[134] H. Rahmani, A. Mahmood, D.Q. Huynh, A. Mian, Hopc: histogram of oriented principal components of 3d pointclouds for action recognition, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Zurich, 2014, pp. 742–757.
[135] E. Ohn-Bar, M.M. Trivedi, Joint angles similarities and HOG2 for action recognition, in: Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Portland, Oregon, 2013, pp. 465–470, http://dx.doi.org/10.1109/CVPRW.2013.76.
[136] L. Xia, J. Aggarwal, Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, Oregon, 2013, pp. 2834–2841.
[137] Y. Zhu, W. Chen, G. Guo, Fusing spatiotemporal features and joints for 3D action recognition, in: Proceedings of Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Portland, Oregon, 2013, pp. 486–491, http://dx.doi.org/10.1109/CVPRW.2013.78.
[138] I. Laptev, On space–time interest points, Int. J. Comput. Vis. 64 (2) (2005) 107–123, http://dx.doi.org/10.1007/s11263-005-1838-7.
[139] S. Althloothi, M.H. Mahoor, X. Zhang, R.M. Voyles, Human activity recognition using multi-features and multiple kernel learning, Pattern Recognit. 47 (5) (2014) 1800–1812.
[140] J. Wang, Y. Wu, Learning maximum margin temporal warping for action recognition, in: 2013 IEEE International Conference on Computer Vision (ICCV), IEEE, Sydney, Australia, 2013, pp. 2688–2695.
[141] C. Chen, R. Jafari, N. Kehtarnavaz, Improving human action recognition using fusion of depth camera and inertial sensors, IEEE Trans. Hum.-Mach. Syst. 45 (1) (2015) 51–61, http://dx.doi.org/10.1109/THMS.2014.2362520.
[142] A.F. Bobick, J.W. Davis, The recognition of human movement using temporal templates, IEEE Trans. Pattern Anal. Mach. Intell. 23 (3) (2001) 257–267.
[143] H.M. Hondori, M. Khademi, C.V. Lopes, Monitoring intake gestures using sensor fusion (microsoft kinect and inertial sensors) for smart home tele-rehab setting, in: 1st Annual IEEE Healthcare Innovation Conference, IEEE, Houston, TX, 2012, pp. 1–4.
[144] B. Delachaux, J. Rebetez, A. Perez-Uribe, H.F.S. Mejia, Indoor activity recognition by combining one-vs.-all neural network classifiers exploiting wearable and depth sensors, in: Advances in Computational Intelligence. Lecture Notes in Computer Science, Springer, Tenerife - Puerto de la Cruz, Spain, 7903 (2013), pp. 216–223.
[145] K. Liu, C. Chen, R. Jafari, N. Kehtarnavaz, Fusion of inertial and depth sensor data for robust hand gesture recognition, IEEE Sens. J. 14 (6) (2014) 1898–1903.
[146] S. Hadfield, R. Bowden, Hollywood 3d: recognizing actions in 3d natural scenes, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, Oregon, 2013, pp. 3398–3405.
[147] C. Ionescu, D. Papava, V. Olaru, C. Sminchisescu, Human3. 6m: large scale datasets and predictive methods for 3d human sensing in natural environments, IEEE Trans. Pattern Anal. Mach. Intell. 36 (7) (2014) 1325–1339.
[148] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, R. Bajcsy, Berkeley MHAD: a comprehensive multimodal human action database, in: Proceedings of Workshop on Applications of Computer Vision (WACV), IEEE, Clearwater Beach Florida, 2013, pp. 53–60.
[149] J.R. Padilla-López, A.A. Chaaraoui, F. Flórez-Revuelta, A discussion on the validation tests employed to compare human action recognition methods using the MSR Action 3D dataset, CoRR abs/1407.7390.arXiv:1407.7390.
[150] J. Sung, C. Ponce, B. Selman, A. Saxena, Human activity detection from RGBD images, in: AAAI Workshops on Plan, Activity, and Intent Recognition, San Francisco, CA, USA, vol. 64, 2011, pp. 1–8.
[151] S. Fothergill, H.M. Mentis, P. Kohli, S. Nowozin, Instructing people for training gestural interactive systems, in: J.A. Konstan, E.H. Chi, K. Höök (Eds.), Proceedings of ACM Conference on Human Factors in Computing Systems (CHI), ACM, Austin Texas, 2012, pp. 1737–1746, http://dx.doi.org/10.1145/2207676.2208303.
[152] A. Malizia, A. Bellucci, The artificiality of natural user interfaces, Commun. ACM 55 (3) (2012) 36–38.

Liliana Lo Presti is post-doc in the Computer Vision and Image Processing (CVIP) Group, Università degli Studi di Palermo (UNIPA), Italy, since 2013. She got her Ph.D. in Computer Engineering from UNIPA in 2010. In 2011–2013, she worked as post-doc in the CS Department of Boston University, Massachusetts, USA. She is interested in video-surveillance, multimedia stream alignment, action recognition and behavior understanding. She is member of IAPR and has served as reviewer for many international journals and conferences.

Marco La Cascia received the Electrical Engineering degree (summa cum laude) and the Ph.D. degree from the Università degli Studi di Palermo, Italy, in 1994 and 1998, respectively. From 1998 to 1999, he was a post-doctoral fellow with the Image and Video Computing Group, in the Computer Science Department at Boston University, Massachusetts and was visiting student with the same group from 1996 to 1998. From 1999 to 2000 he was at Offnet S.p.A., (Rome) as senior software engineer. In 2000 he joined the Faculty of Engineering of Università degli Studi di Palermo as assistant professor. Presently, he is Associate Professor at the same institution where he is also coordinator of the Computer and Telecommunication Engineering degrees. His research interests include low and mid-level computer vision with applications to image and video databases and distributed video-surveillance systems. He is co-author of about 80 scientific papers. Marco La Cascia is member of IEEE and IAPR and has served as reviewer for many international journals and conferences. He was also local coordinator in several research projects and participant in many others.