Motion Segments Decomposition of RGB-D Sequences for Human Behavior Understanding

Maxime Devanne, Stefano Berretti, Pietro Pala, Hazem Wannous, Mohamed Daoudi, Alberto Del Bimbo

Motion Segments Decomposition of RGB-D Sequences for Human Behavior Understanding

Maxime Devanne[a,b,*], Stefano Berretti[b], Pietro Pala[b], Hazem Wannous[a], Mohamed Daoudi[a], Alberto Del Bimbo[b]

[a] *Télécom Lille, Univ. Lille, CNRS, UMR 9189 - CRIStAL, F-59000 Lille, France*
[b] *MICC/University of Florence, Florence, Italy*

Abstract

In this paper, we propose a framework for analyzing and understanding human behavior from depth videos. The proposed solution first employs shape analysis of the human pose across time to decompose the full motion into short temporal segments representing elementary motions. Then, each segment is characterized by human motion and depth appearance around hand joints to describe the change in pose of the body and the interaction with objects. Finally, the sequence of temporal segments is modeled through a Dynamic Naive Bayes classifier, which captures the dynamics of elementary motions characterizing human behavior. Experiments on four challenging datasets evaluate the potential of the proposed approach in different contexts, including gesture or activity recognition and online activity detection. Competitive results in comparison with state of the art methods are reported.

Keywords: 3D human behavior understanding, temporal modeling, shape space analysis, online activity detection.

*Corresponding author
Email address: maxime.devanne@telecom-lille.fr (Maxime Devanne)

1. Introduction

Visual recognition and understanding of human activity and behavior represent a task of interest for many multimedia applications, including entertainment, medicine, sport, video surveillance, human-machine interfaces and active assisted living. This wide spectrum of potential applications encouraged computer vision community to address the issue of human behavior understanding from 2D videos taken from standard RGB cameras [1, 2, 3, 4, 5]. However, most of these methods suffer from some limitations, like the sensitivity to color and illumination changes, background clutter and occlusions. Since the recent release of RGB-D sensors, new opportunities have emerged in the field of human motion analysis and understanding. Hence, many research groups investigated data provided by such cameras in order to benefit from some advantages compared to RGB cameras [6, 7, 8, 9, 10]. Indeed, depth data allows a better understanding of the 3D structure of the scene and thus makes background subtraction and people detection easier. In addition, the technology behind such depth sensors provides robustness to light variations as well as the capability to work in complete darkness. Finally, the combination of such depth sensors and powerful pattern recognition algorithms [11] enables the representation of human pose at each frame as a set of 3D joints. In the past decades, human motion analysis from 3D data provided by motion capture systems has been widely investigated [12, 13, 14]. While these systems are very accurate, they present some disadvantages. First, the cost of such technology may limit its use. Second, it implies that the subject wears some physical markers so as to estimate the 3D pose. As a result, this technology is not convenient for the general public. All these considerations motivated us to focus our study of human behavior on RGB-D data. However, this task still faces some major challenges due to the temporal variability and complexity of human actions and the large number of

motion combinations that can characterize the human behavior. Motion analysis is further complicated by the fact that it should be invariant to geometric transformations, such as translation, rotation and global scaling of the scene. In addition, human behavior often involves interaction and manipulation of objects. While such information about the context may help the understanding of what the human is doing, it also involves possible occlusions of parts of the human body, resulting in missing or noisy data.

In order to face these challenges, we propose in this paper to locally investigate the sequence by detecting short temporal segments representing elementary motion, called *Motion Segments* (MS). Then, for each MS, we analyze human motion and depth appearance around human hands to characterize the interaction with objects. This provides a deeper analysis of the human behavior and allows the recognition of human *gestures*, *actions* and *activities*. In particular, in this paper, *gestures* indicate simple movements performed with only one part of the body, *actions* represent a combination of gestures with different parts of the body, and *activities* refer to more complex motion patterns possibly involving interaction with objects. The proposed solution can be adapted to realistic scenarios, where several actions or activities are performed subsequently in a continuous sequence. In that case, the sequence should be processed *online* in order to detect the starting and ending time of actions or activities. That is, the proposed approach can operate on the data stream directly, without assuming the availability of a segmentation module that identifies the first and last frame of each action/activity.

1.1. Previous Work

In recent years, recognition and understanding of human behavior by analyzing depth data has attracted the interest of several research groups [15, 16, 17, 18]. While some methods focus on the analysis of human motion in order to

recognize human *gestures* or *actions*, other approaches try to model more complex behaviors (*activities*) including object interaction. These solutions focus on the analysis of short sequences, where one single behavior is performed along the sequence. However, additional challenges appear when several different behaviors are executed one after another over a long sequence. In order to face these challenges, methods based on *online detection* have been proposed. Such methods can recognize behavior before the end of their execution by analyzing short parts of the observed sequence. Thus, these methods are able to recognize multiple behaviors within a long sequence, which may not be the case for methods analyzing the entire sequence directly. Existing methods for human behavior recognition using depth data are shortly reviewed below.

Methods analyzing human motion for the task of *gesture / action* recognition from RGB-D sensors can be grouped into three categories: *skeleton*-based, *depth map*-based and *hybrid* approaches. Skeleton based approaches have become popular thanks to the work of Shotton et al. [11]. This describes a real-time method to accurately predict the 3D positions of body joints in individual depth maps, without using any temporal information. In [19], Yang and Tian performed human action recognition by extracting three features for each joint, based on pair-wise differences of joint positions (initial, previous and current frames). PCA is then used to obtain a compact *EigenJoints* representation of each frame and a naïve-Bayes nearest-neighbor classifier is used for multi-class action classification. Similar features are used by Luo et al. [20], but pairwise differences are computed only in the current frame and with respect to only one reference joint (the hip joint). To better represent these features, they propose a dictionary learning method based on group sparsity and geometry constraints. The classification of sequences is performed using SVM. Zanfir et al. [15] propose the Moving Pose feature, capturing for each frame the human pose information

as well as the speed and acceleration of body joints within a short temporal window. A modified kNN classifier is employed to perform action recognition. Hongzhao et al. [21] introduce a part-based feature vector to identify the most relevant body parts in each action sequence. Other approaches use differential geometry to represent skeleton data. In [22], Vemulapalli and Chellappa represented each skeleton as one element on the Lie-group, and the sequence corresponds to a curve on this manifold. In [23], Slama et al. express the time series of skeletons as one point on a Grassmann manifold, where the classification is performed benefiting from Riemannian geometry of this manifold. In [24], Anirudh et al. regard actions as trajectories on a Riemannian manifold, and analysis of such trajectories using Transport Square-Root Velocity Function is employed for action recognition.

Methods based on depth maps extract meaningful descriptors from the entire set of points of depth images. In [25], Yang et al. described the action dynamics using Depth Motion Maps, which highlight areas where some motion takes place. Other methods, such as Spatio-Temporal Occupancy Pattern [26], Random Occupancy Pattern [27] and Depth Cuboid Similarity Feature [16], propose to work on the 4D space divided into spatio-temporal boxes to extract features representing the depth appearance in each box. Such features are extracted from Spatio-Temporal Interest Points. A similar method is proposed by Rahmani et al. [28], where keypoints are detected and the point cloud is described within a volume using the Histogram of Principal Components. In [29], Oreifej and Liu proposed a method to quantize the 4D space using vertices of a polychoron, and then model the distribution of the normal vectors for each cell. The idea of using surface normals to describe both local motion and shape information characterizing human action is also used by Yang and Tian [30]. Althloothi et al. [31] represent 3D shape features based on spherical harmonics

representation and 3D motion features using kinematic structure from skeleton. Both features are then merged using a multi-kernel learning method. A depth feature to describe shape geometry and motion, called Range-Sample, is proposed by Lu and Tang [32].

Analyzing human motion, however, may not be sufficient to understand more complex behaviors involving human interaction with the environment (i.e., what we call *activities*). Hybrid solutions are often proposed, which use depth maps for modeling scene objects and body skeleton for modeling human motion. For example, Wang et al. [33] used Local Occupancy Patterns to represent the observed depth values in correspondence to skeleton joints. Other methods propose to describe and model spatio-temporal interaction between human and objects characterizing the activities, using Markov Random Field [17]. A graphical model is also employed by Wei et al. [34] to hierarchically define activities as combination of sub-events including description of the human pose, the object and interaction between them. Yu and Liu [35] propose to capture meaningful skeleton and depth features using a middle level representation called *orderlet*.

Some of the works reviewed above have also *online* action recognition capabilities, as they compute their features within a short sliding window along the sequence [35]. This challenge has recently been investigated for continuous depth sequences, where several actions or activities are performed successively. For example, Huang et al. [18] proposed and applied the Sequential Max-Margin Event Detector algorithm on long sequences comprising many actions in order to perform online detection by successively discarding not corresponding action classes.

1.2. Overview of Our Approach

Human behavior is naturally characterized by the change of the human body across time. Thanks to depth sensors, we are able to capture skeleton data con-

taining the 3D position of different parts of the body. The skeleton and its changes across time provide valuable information. However, understanding the human behavior is still a difficult task due to the complexity of human motion and spatial/temporal variations in the way gestures, actions, or activities are performed. These challenges motivated us to analyze locally the motion sequences. First, we represent the skeleton of each frame by a 3D curve describing human pose. These curves are then interpreted in a Riemannian manifold, which defines a *shape space* where shapes of the curves can be modeled and compared using elastic registration and matching. Such shape analysis allows the identification and grouping of the human poses. As a result, a motion sequence is temporally segmented into a set of successive sub-sequences of elementary motions, called *Motion Segments* (MS). A MS is thus characterized by a sequence of skeletons, each of which is modeled as a multi-dimensional vector by concatenating the three-dimensional coordinates of its joints. Then, the trajectory described by this vector in the multi-dimensional space is regarded as a signature of the temporal dynamics of all the joints. Similarly to pose curves, the shape of such motion trajectories is studied in a Riemannian shape space. The elastic metric provided in this framework allows us to compare motion trajectories independently to their elasticity, i.e., the execution speed of motions. A statistical analysis on this manifold allows us to identify relevant shapes characterizing a set of MSs. However, skeleton data is not sufficient to describe human behavior in cases where objects are manipulated. This motivated us to describe, in each MS, the depth appearance around subject hands providing information about possible human-object interactions. Finally, the sequence of MSs is modeled through a Dynamic Naive Bayes classifier, which combines both skeleton and depth features and captures the dynamics of human behavior. Figure 1 summarizes the proposed solution. The main contributions

of the proposed approach are:

- A segmentation method based on the statistical shape analysis of human pose variation along the sequence;
- A temporal description of a sequence, which combines elastic shape analysis of motion trajectories on a Riemannian manifold, and description of depth appearance around subject hands.

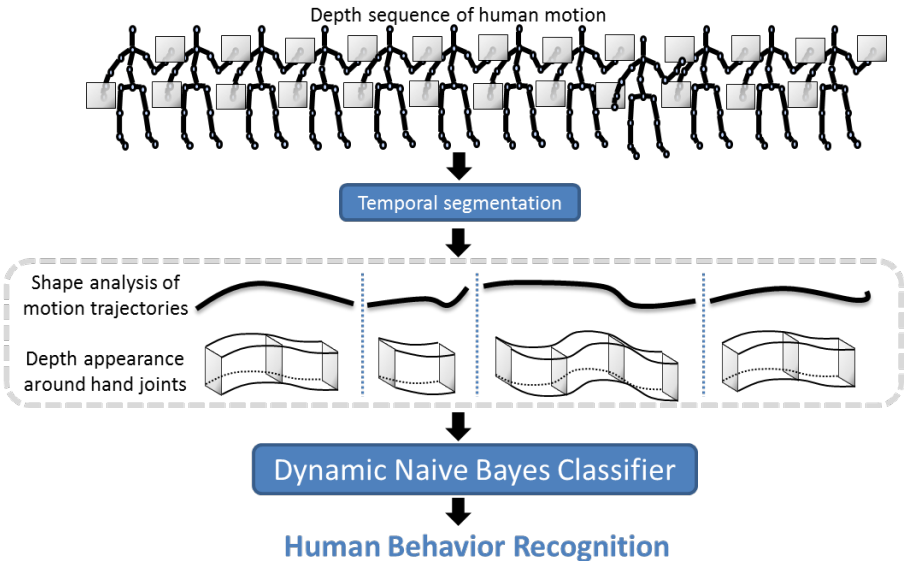


Figure 1: Overview of our approach. Shape analysis of human poses allows us to identify temporal segments of elementary motions (i.e., MS). Each MS is described using the trajectory of the joints of the skeleton regarded as a multidimensional vector, and the depth appearance around subject hands. A Dynamic Naive Bayes classifier is then used to model the sequence of temporal segments and recognize human behavior.

The rest of the paper is organized as follows: Sect. 2.1 discusses the Riemannian framework that we employ for shape analysis of both human pose and human motion; Sect. 2 presents our method for characterizing a motion sequence based on its segmentation into MSs, and their skeletal and depth description; In Sect. 3, we describe the Dynamic Naive Bayes classifier and show how we use it for classification and online detection; Sect. 4 describes the experimental

settings, the datasets used and also reports results in terms of accuracy of gesture, action and activity recognition in comparison to state of the art solutions; Finally, in Sect. 5 conclusions and future research directions are drawn.

2. Description of Activity Sequences

Our proposed approach is based on the analysis of both human pose and human motion. Using a shape analysis framework, an activity sequence is analyzed and described through two steps: First, we locally regard it at the level of human poses in order to segment the full human motion into a set of MSs; Then, the analysis of these segments allows us to describe the sequence as a combination of successive MSs.

2.1. Shape Analysis Framework

A pose of human body can be characterized by the spatial configuration of body parts. So we propose to analyze the shape of such spatial configuration. Human motion is characterized by the evolution of the human pose across time. In order to capture the geometric deformation of the pose as well as the dynamics of the motion, we propose to consider the motion as a trajectory of the human pose and analyze its shape. As a result, we recast the problem of human pose and human motion analysis to a problem of shape analysis by employing the *Shape Analysis* framework, presented in [36]. In the following, we recall the main idea of the framework and we refer the reader to [36] for more details. In this framework, the shape of a n-dimensional curve $\beta : I \to \mathbb{R}^n$, normalized in the interval $I = [0,1]$, is captured through the *Square-root Velocity Function* (SRVF) [37] defined as: $q(t) \doteq \dot{\beta}(t)/\sqrt{\|\dot{\beta}(t)\|}$. As a result, each q function can be viewed as an element of a Riemannian manifold $\mathcal{C}$ and the distance between two elements q_1 and q_2 is the length of the geodesic path connecting them on $\mathcal{C}$. Such geodesic path represents the elastic deformation of the shape q_2 to

correspond to the shape q_1. As $\mathcal{C}$ is a hyper-sphere, the geodesic length between two elements q_1 and q_2 is defined as $\theta = d_{\mathcal{C}}(q_1, q_2) = cos^{-1}(\langle q_1, q_2 \rangle)$.

The SRVF representation is invariant to translation and scaling, but it is not invariant to rotation and re-parametrization. To cope with this, we define the equivalence class of q as $[q]$ where elements of $[q]$ are equivalent up to rotation and re-parametrization. The set of all equivalence classes is called the *shape space* denoted as $\mathcal{S}$. To compute the geodesic distance between $[q_1]$ and $[q_2]$ on $\mathcal{S}$, we first need to find the optimal rotation and re-parametrization that register the element q_2 with respect to q_1 resulting in q_2^*. Then, the distance $d_{\mathcal{S}}([q_1], [q_2]) = d_{\mathcal{C}}(q_1, q_2^*)$ is invariant to translation, scale, rotation and re-parametrization of curves. In practice, SVD is used to find the optimal rotation, and Dynamic Programming is used to find the optimal re-parametrization.

2.2. Segmentation of Sequences

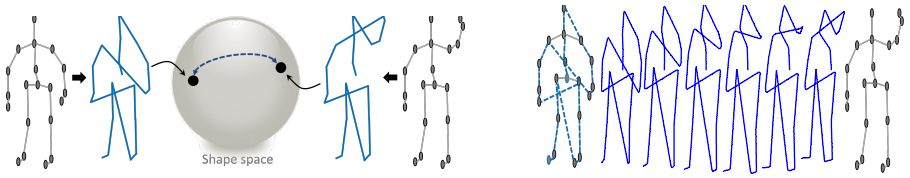
Due to the complexity of human motion characterizing activities, we propose to decompose the full motion into shorter MSs, which are easier to analyze. The idea of decomposing a motion sequence into a set of MSs has already been investigated in state-of-the-art. In [38] the 'movelet' is proposed on accelerometer data by concatenating features within overlapping temporal intervals with fixed length. However, as the length of each temporal interval is fixed, it may not represent a relevant MS. Another idea called 'dyneme' is employed in [14], where human poses are clustered to identify several temporal segments with similar poses represented by one centroid pose. However, the use of pose information only may lack of information about the dynamics of the MS. In addition, labeling successive poses independently may result in irrelevant intervals. In this paper, we propose to identify relevant MSs including continuous elementary motions. This process is based on the analysis of the human pose at each frame of the sequence.

2.2.1. Pose Representation and Analysis

Human body is represented by a set of 3D joints located in correspondence to different body parts. Thus, a human pose is characterized by a certain spatial configuration of these 3D joints. In order to describe human poses, we propose to analyze the shape of the spatial configuration of 3D joints. By connecting the 3D joints, human pose can be viewed as a 3D curve representing the shape of human body. As shown in Fig. 2, in order to keep the human shape information associated to the limbs, we keep a coherent structure linking together joints belonging to the same limb. Thus, a 3D curve representing the human pose connects successively the spine joints, then the arms joints (left/right) and finally the legs joints (left/right). In this way, a human pose is represented by a 3D curve instead of a 3D skeleton. Thus, We can perform shape analysis of curves using the shape analysis framework and the provided distance (see Sect. 2.1) for $n = 3$ as each joint is represented by the x, y, z coordinates. Note that, as we will explain later, we need to compare successive human poses from a same sequence (same subject). Hence, we can assume that the scale of skeletons as well as the orientation of the subject between two successive poses are unchanged during a short time interval. Likewise, as a 3D curve connects joints in a predefined order, the parametrization of curves remains the same along a single sequence. Since it is not necessary to find the optimal re-parametrization between two shapes, the analysis of the shape of the 3D curves is simplified. Figure 2 shows a geodesic path between two human poses represented by their 3D curve.

2.2.2. Motion Segmentation

Once a distance measuring the similarity between the shape of two poses is defined, we can use it to analyze the deformation of human body along an activity sequence. Hence, in order to divide the continuous sequence into MSs,



(a) (b)

Figure 2: Shape analysis of human poses in the shape space. (a) Shape of 3D curves representing human poses are interpreted in the shape space where the distance between two shapes is measured through the geodesic distance (length of the minimum path). (b) Visualization of the geodesic path representing a natural deformation between shape of poses.

we detect when the motion is changing. We identify MSs by breaking the sequence in correspondence to points where the speed of change of the 3D curve has a local minimum. To compute the speed of change, we take advantage of the shape analysis framework that enables the computation of statistics, like the mean and the standard deviation, on the manifold. Hence, given the poses $p_1, \ldots, p_n$ observed over a temporal window of predefined duration, the average pose shape μ is computed as the Riemannian center of mass [39] of the pose shapes $q_1, \ldots, q_n$ on the shape space. For this purpose, the distance $d_{\mathcal{S}}$ described in Sect. 2.1 is used according to the following expression:

$$\mu = \arg\min_{[q]} \sum_{i=1}^{n} d_{\mathcal{S}}([q], [q_i])^2 \ . \tag{1}$$

Once the mean pose shape is computed, the standard deviation σ between this mean shape and all the shapes within the window is estimated:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} d_{\mathcal{S}}([\mu], [q_i])^2} \ . \tag{2}$$

Higher values of σ correspond to faster motion, while lower values correspond to slower motion, i.e., transition intervals. By detecting local minima along the sequence, we are able to temporally localize the motion transition, and thus decompose the sequence into MSs. As an example, Fig. 3 shows the

variation of σ along a sequence and the MSs identified by breaking the sequence in correspondence to local minima of σ.

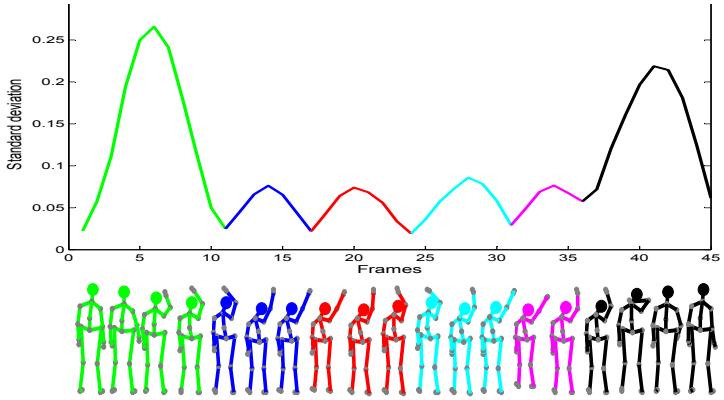


Figure 3: Segmentation of a sequence based on minima of the standard deviation σ. Different MSs and corresponding poses are displayed with different colors.

2.3. Segment Description

Once an activity sequence is segmented, we analyze the resulting MSs in order to describe the whole sequence.

2.3.1. Human Motion Analysis

Here, we interpret the pose changes across a time interval corresponding to a MS. For each frame included in a MS, we concatenate the x_i, y_i, z_i coordinates of each joint to build a feature vector. Let N_j be the number of joints of the skeleton, the posture of the skeleton at frame t is represented by a $3N_j$ dimensional tuple:

$$v(t) = [x_1(t)\ y_1(t)\ z_1(t), \ldots, x_{N_j}(t)\ y_{N_j}(t)\ z_{N_j}(t)]^T\ . \tag{3}$$

For a MS composed of N_f frames, N_f feature vectors are extracted and

arranged in columns to build a feature matrix M describing the whole segment:

$$M = \begin{pmatrix} v(1) & v(2) & \dots & v(N_f) \end{pmatrix} . \tag{4}$$

This matrix captures the changes of the skeleton pose across time. Hence, it can be viewed as a trajectory in R^{3N_j} representing the motion in a $3N_j$ dimensional space. The size of such feature matrix is $3N_j \times N_f$. Note that, in order to guarantee invariance to MSs translation and rotation, we normalize the position and the orientation of the subject before extracting the features. We use the spine and hips joints to form the base representing the position and orientation of the body. We align the initial pose of a segment with respect to a reference posture by finding the best rigid transformation between corresponding bases. The optimal transformation is then applied to all other poses of the segment. This makes the representation invariant to the position and orientation of the subject in the scene (see [36] for more details). With this representation, an activity sequence can be viewed as a set of short spatio-temporal trajectories in R^{3N_j} representing MSs, as illustrated in Fig. 4.

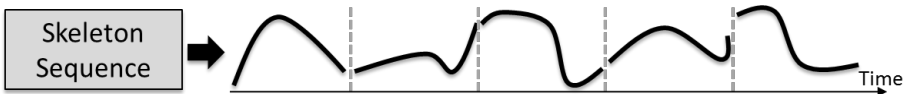


Figure 4: An activity sequence can be viewed as a set of successive spatio-temporal trajectories in R^{3N_j} representing MSs performed by the subject.

In this paper, we propose to use the *shape analysis* framework described in Sect. 2.1, with $n = 3N_j$, to capture and analyze the shape of trajectories of MSs. Shapes are represented as elements on the shape space and the similarity measure between two shapes is the elastic metric $d_{\mathcal{S}}$ on this shape space. Our idea here is to identify a *codebook* of exemplar shapes (*symbols*) to be used as a reference dictionary in the classification. To learn the codebook, we perform clustering of shapes using the k-means algorithm. First, k shapes are randomly

selected as mean shapes of k clusters, and each sample shape is assigned to the clusters using the $d_{\mathcal{S}}$ distance. Then, the mean shapes are repeatedly updated until convergence using the Riemannian center of mass (see Eq. (1)). Such clustering provides a mapping between trajectory shapes represented on the shape space and a finite set of symbols corresponding to clusters.

In order to describe each cluster by using its corresponding mean shape, we learn a density function for each cluster. These density functions capture the variability between shapes belonging to the same cluster and provide a deeper modeling of each cluster. In so doing, we assume the distribution of shapes within a cluster follows a multivariate normal model. Unfortunately, learning such density functions on the shape space is not straightforward, mainly due to the non-linearity and infinite-dimensionality of such manifold (i.e., shapes are represented by functions, so they have infinite dimension). Different methods have been proposed to deal with these two challenges [40, 41]. A common way to circumvent the non-linearity of the manifold is to consider a hyperplane tangent to the manifold at the mean shape (i.e., *tangent space*). Such tangent space is a linear vector space, where conventional statistics applies, like the computation of density functions. We denote $T_{\mu_k}\mathcal{S}$ the tangent space at the mean shape of the k-th cluster μ_k. For each shape $q_i \in \mathcal{S}$ within the k-th cluster, we compute its corresponding tangent vector $v_i \in T_{\mu_k}\mathcal{S}$ using the logarithm map. This approximation is valid because samples belong to the same cluster. Thus, we can assume that they lie in a small neighborhood around the mean shape μ_k. To deal with the problem of infinite-dimensionality, we assume the variations in tangent vectors are restricted to an m-dimensional subspace. Using tangent vectors of each cluster, we use PCA to learn a principal subspace for each cluster. We denote n the dimension of such principal subspace. Tangent vectors v_i are then projected into the learned subspace. Let $\tilde{v}_i$ be such projected

vectors, we compute the covariance matrix Σ between all projected vectors $\tilde{v}_i$ belonging to the same cluster. Finally, we use the resulting mean shape μ and covariance matrix Σ to learn a multivariate normal distribution for each cluster. Its corresponding probability density function is defined as:

$$f(\tilde{v}) = \frac{1}{(2\pi)^{n/2}\ |\Sigma|^{1/2}} e^{-\frac{1}{2}\tilde{v}^T\Sigma^{-1}\tilde{v}}\ . \tag{5}$$

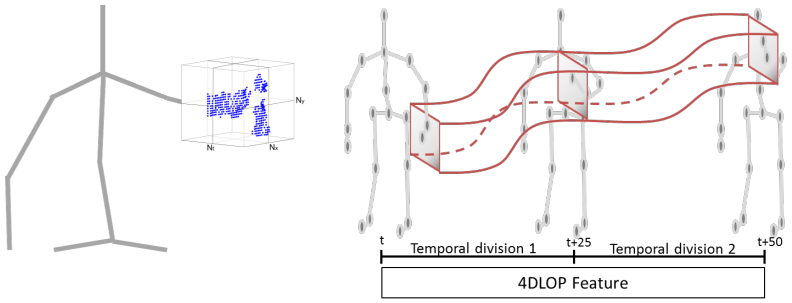
The codebook is learned from MSs of training sequences. Such codebook is then used to label MSs of the test sequence, characterized by its trajectory shape in the shape space. The test shape is first projected into the learned subspace of a cluster k. Then, using the corresponding covariance matrix, we can compute the probability that the test shape is generated by the learned density function corresponding to the cluster k. We do the same for each cluster and assign the test shape to the cluster giving the highest probability.

2.3.2. Depth Appearance

Descriptors of human motion are complemented with descriptors of the objects the user is interacting with, if any. Such combination of motion and object descriptors improves the robustness of the activity recognition, and is also necessary to discriminate between actions that would be almost identical in terms of motion patterns. For instance, discriminating between activities like *Drink* and *Phone call* based on the analysis of the sole motion patterns would require a description framework capable of accurately distinguishing whether the user hand is closer to the mouth than to the ear. This level of accuracy is generally beyond the capability of commercial low-res scanners, unless the user is very close to the sensor. Differently, two such actions can be much more easily discriminated by considering the objects the user is interacting with.

In order to describe the distribution of depth pixels within a local region

around subject hands, we adapt the Local Occupancy Pattern (LOP) [27] feature. In this approach, a depth image is viewed as a 3D point cloud, and the local regions are represented by 3D bounding boxes centered at the hand joints. As shown in Fig. 5a, each bounding box is partitioned into $N_c = N_x \times N_y \times N_z$ 3D cells, and the number of 3D points that fall in each cell is counted. In the experiments, we empirically select a local region of size $0.3m \times 0.3m \times 0.3m$ divided into $5 \times 5 \times 5$ cells.



(a) (b)

Figure 5: LOP feature computation. (a) A 3D cuboid divided into 3D cells is extracted from the depth image around the hand joint and the number of 3D points within each 3D cell is counted. (b) Schema of the 4DLOP feature representing depth appearance evolution along a MS in two time steps.

This local depth representation is combined with the motion description, which represents an activity as a sequence of successive MSs. For each frame of a MS, we compute the LOP feature for each hand joint (l_l and l_r) and concatenate them to form one global LOP feature vector $L_f = [l_l, l_r]$ for the frame f. The length of such feature vector is $2 \times N_c$. However, MSs can have different duration. As a consequence, they are described with a different number of LOP features, which is not convenient in the comparison. To deal with duration variability, we propose a compact representation of the depth appearance, which is independent from its duration. First, we assume the object held by the subject during the time interval corresponding to a MS does not change considerably, and we

compute the mean of the LOP features among frames of a MS. Thus, one single feature, that we call Mean LOP (MLOP) is used to describe the average depth appearance of a MS. Then, we consider changes of depth appearance around hand joints, which can be induced by object manipulation during a MS. For instance, for the activity *Drink* a MS would consist of bringing the container to the mouth. In that case, the support where the object is located may appear in the local region around the hand, in the first part of the MS, but the face of the subject may be present in this local region at the end of the MS. To represent this depth variation, we adopt an extension of LOP feature in four dimensions called 4DLOP. The spatio-temporal volume representing the change of the local region around hands along the MS is also partitioned in N_t divisions across temporal dimension. Note that, differently to [26], which analyzes depth variation in fixed 4D boxes, we consider depth variation in a moving spatio-temporal region following the motion of human hands. This idea is illustrated in Fig. 5b.

As a result, each MS is represented by a feature vector describing the depth appearance independently to its duration (either MLOP or 4DLOP). To cluster LOP features and build a codebook of exemplar LOP, we use the k-means algorithm with Euclidean distance. Such clustering provides a mapping between LOP feature vectors and a finite set of LOP symbols represented by the cluster centroids. Similarly to human motion, the codebook is learned from MSs of training sequences. For MSs of test sequences, we compute the distance between the corresponding LOP feature and all the exemplar LOP and labeling is done using the nearest rule.

3. Modeling of Activity Sequences

As discussed in Sect. 2, a sequence is decomposed in MSs, and each MS is described in terms of human motion and depth appearance around subject hands. Thus, the dynamics of a sequence can be viewed as combination of two sequences of successive symbols, one corresponding to human motion, and the other corresponding to depth appearance around hands. In so doing, we assume that sequences of the same class are represented by similar arrangements of MSs. Conversely, different sequences of symbols should represent different classes. Hence, we need a method to analyze the change of symbols across time, and recognize different arrangements of MSs. To this end, we propose to use the Dynamic Naive Bayes classifier (DNBC) [42] as statistical model.

3.1. Learning

In DNBC training, we only know the sequence of observations $X = \{X_t^a \mid t = 1, \ldots, T \ , \ 1 \leq a \leq A\}$, being A the number of attributes, while the states $S = \{S_t | t = 1, \ldots, T\}$ are not available. Thus, we need tools for estimating the model parameters, i.e., the *prior*, *transition* and *emission* probabilities. The prior probability represents the initial state of the process. The transition probability is the probability to transit from one state to another state of the process. The emission probability represents, for each state, the probability of generating each attribute. Similarly to HMM, a common way to learn such parameters from training sequences of observed symbols is to use the Baum-Welch algorithm [43]. In the case of DNBC, parameters estimation is slightly modified due to the model setting, which allows the emission of several attributes per state (more details on this can be found in [44]). For our task, we assume that each activity class is modeled with a different DNBC. Let the activity class $c \in \{1, \ldots, C\}$ with C being the number of activity classes, we learn one DNBC denoted λ_c for each class c using the training sequences of to the class c.

3.2. Classification

The classification process of an observed sequence X is the performed as follows. First, the sequence is presented to each of the trained λ_c DNBC modeling different activity classes. Then, the likelihood $P(X|\lambda_c)$ that the sequence X has been generated by the λ_c DNBC is computed using the *Forward* algorithm. Finally, the sequence is classified as the activity whose corresponding DNBC gives the highest log-likelihood: $activity(X) = \arg\max_c P(X|\lambda_c)$.

The classification process is then extended to work in an online manner, so that a classification decision can be taken before the end of a sequence. This is particularly convenient for real-time applications, permitting natural interaction with the system. In addition, it allows us to process a sequence as a continuous stream, where several activities can be performed successively, which is often the case in real-world contexts. As shown in Sect. 2.2, the segmentation process is based on a sliding window technique. Hence, it can also be applied in an online manner so as to detect MSs from a continuous stream. Each new frame of the sequence is given as input to the segmentation process. When a MS is detected, we compute the corresponding human motion and depth appearance features and assign a symbol to each, as described in Sect. 2. The resulted observation sequence of length-1 is then presented to each trained DNBC in order to compute the corresponding log-likelihoods. This process is performed for each new detected MS. Thus, the length of the observation sequence is increased by one, and the log-likelihoods are updated. If the log-likelihood of a class falls below a threshold, we discard the activity class. This allows us to gradually reduce the set of possible classes. The process is repeated until all classes are discarded. Among the remaining classes, we keep the class with the highest log-probability as the detected activity. However, transitions between activities are often smooth. Thus, when an activity is finished, its corresponding log-

probability may not considerably decrease and directly fall below the threshold. In order to consider this smooth transition, we select as the ending boundary of the activity the time step when its corresponding log-probability starts to decrease instead of the time step when it falls below the threshold. Finally, we restart the detection process from the successive time step using all the classes. This is repeated until the end of the sequence. As a result, we obtain the set of detected activities along the sequence with corresponding starting and ending boundaries. This online activity detection is illustrated in Fig. 6.

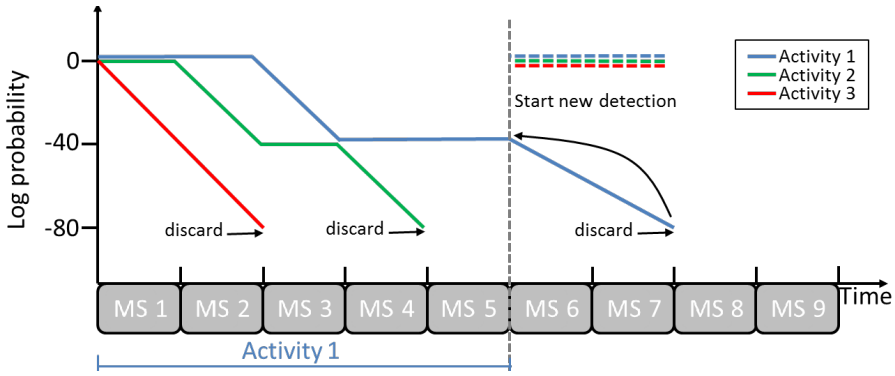


Figure 6: Online detection method. The *Activity-2* and *Activity-3* are discarded after the fourth and second time step, respectively, as their log-probability fall below -80. The remaining *Activity-1* is discarded after the seventh time interval. As a result, the five first time intervals are classified as *Activity-1*, and a new detection is started from the sixth time step.

4. Experimental Evaluation

We evaluate the proposed approach in comparison with state-of-the-art methods using four public benchmark datasets.

4.1. MSRC-12 dataset

First, we evaluate our method in the task of human gesture recognition. The main goal of this experiment is to show how the proposed method deals with actions characterized by repetitions of a single gesture. In particular, we

want to evidence the proposed decomposition of a sequence into a set of MSs is capable of managing such variability.

We perform this experiment on the Microsoft Research MSRC-12 dataset, which includes 12 gestures performed by 30 subjects for a total of about 50 sequences per class, where a single gesture is performed several times along a sequence (10 times in most of the cases, but this number may vary from 2 to 15). This variability is indeed important to show how it can affect the recognition accuracy. Only skeleton data is provided in this dataset, so we only use the motion features to describe each segment. Following the same protocol as in Lehrmann et al. [45], only six gestures are considered and a 5-fold cross validation protocol is applied. Results are reported in Table 1 as average accuracy across folds in comparison to [45] and [36].

Table 1: MSRC-12. Comparison of the proposed approach with DFM [45] and [36]. Accuracy is reported in percentage

Class	DFM [45]	Devanne et al. [36]	Our
Duck	96.0	**100**	**100**
Goggles	88.0	82.0	**91.6**
Shoot	**85.7**	73.5	83.0
Throw	**90.0**	88.0	**90.0**
Change weapon	87.5	**89.6**	**94.0**
Kick	**98.0**	**98.0**	**98.2**
Mean	90.9	88.5	**92.8**

From Table 1, we can notice the proposed approach outperforms [45] for all gesture classes except one (*Shoot*), with an overall accuracy of 92.8%, compared to 90.9% reported in [45]. In addition, the accuracy of the proposed approach increases of about 4% that reported in our previous work [36], where the decomposition into MSs is not considered. Moreover, we computed the standard deviation among the 5-folds and obtained a standard deviation of 0.9% for our method compared to 3.1% for [36]. This shows that our method is more robust to the variability of subjects used for training and test. Finally, by investi-

gating the failure cases, we notice that the different number of repetitions in the sequences affects the accuracy of [36] for the case of similar gestures, like *Shoot* and *Goggles*. To emphasize this latter aspect, we run an experiment on the sequences of these two classes only. In the training set, we include *Goggles* sequences with exactly 10 repetitions of the gesture, plus all *Shoot* sequences except those with exactly 10 repetitions of the gesture (these latter sequences are included in the test set). We observe that the recognition accuracy of the class *Shoot* is increased from 39.4% using [36] to 80.2% using the proposed approach. This shows that our method is able to handle various repetitions of a single gesture within a sequence. Indeed, as we use DNBC to model the sequences, repetitions of gestures are characterized by repetitions of the process without changing the structure of the model, thus allowing robustness to repetition variability.

4.2. Cornell Activity dataset 120

We use the Cornell Activity dataset 120 (CAD120) to test our approach in the context of human activity recognition. This dataset contains 120 RGB-D sequences of ten high-level activities involving manipulation with objects, performed by four different subjects three times each. The variability of performed activities, the variability of subject orientation in the scene and the body part occlusion caused by objects make this dataset quite challenging. For a fair comparison with state-of-the-art methods, the *leave-one-person-out* cross protocol is used, and the average accuracy and standard deviation among the four folds are finally computed. Table 2 reports results obtained by our method in comparison to state-of-the-art. Our best accuracy is obtained by using a codebook size of 100 for both features. In particular, methods are compared by separating the case in which only the human skeleton is used, from the case in which both skeleton and depth data are considered.

Table 2: Cornell Activity dataset 120. Comparison of our approach to state of the art methods

Method	Accuracy (%)
Skeleton Only	
Koppula et al. [17]	27.4
Devanne et al. [36]	48.3
Our	**69.4 ± 4.1**
Skeleton + Depth	
Koppula et al. [17]	80.6
Koppula and Saxena [46]	**83.1**
Rybok et al. [47]	78.2
Our (Skel + MLOP)	79.0
Our (Skel + LOP4D)	**82.3 ± 3.4**

From the results, we can first notice that our method significantly outperforms the other approaches when only skeleton data is used. More specifically, in comparison with [36], which represents each activity by spatio-temporal trajectory only, the recognition accuracy is improved by more than 20%. This shows that when activities involve complex motions, it is not sufficient to analyze the global motion. Indeed, local analysis and decomposition of the activity into MSs provides a better representation of activities, thus allowing a better understanding of the human behavior. In addition, the accuracy of 69.4% obtained by our method shows that the decomposition of the sequence allows us to quite well recognize activity sequences involving objects manipulation, even without describing any explicit information about objects held by the subject. However, results show that using only skeleton data is insufficient to be competitive with state-of-the-art methods. As we can see in Table 2, using depth appearance features in addition to skeleton in our DNBC allows us to improve the recognition by about 13%. As a result, we obtain competitive accuracy in comparison with other approaches. Indeed, only [46] is above by less than 1%. Note that methods in [17] and [46] use ground truth object bounding box in the training process. In our case, we do not need this information. Moreover, the small value of standard deviation among the folds shows that our method has

a low dependency on training data.

Finally, by comparing the results obtained with our two different depth appearance features, we can notice that the 4DLOP feature is more effective. This observation is strengthened by the confusion matrices in Fig. 7, and particularly by the confusion obtained for the pair of opposite activities *stacking* and *unstacking objects*. We can see that using the LOP4D feature results in less confusion between the two activities than using the MLOP feature. Indeed, in this particular case, the average depth appearance of *putting* and *taking* the object may be very similar and represented by the same symbol from the codebook. The 4DLOP feature capturing the variation of depth appearance is more suitable to discriminate the two elementary motions, and thus the two activities.

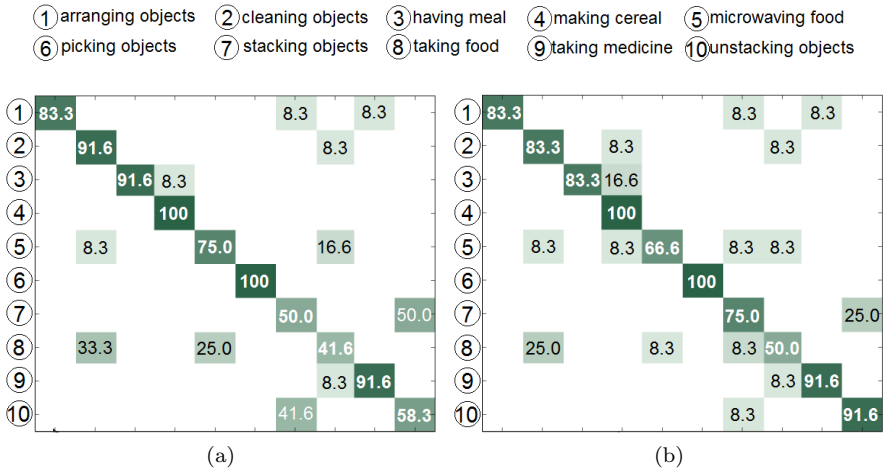
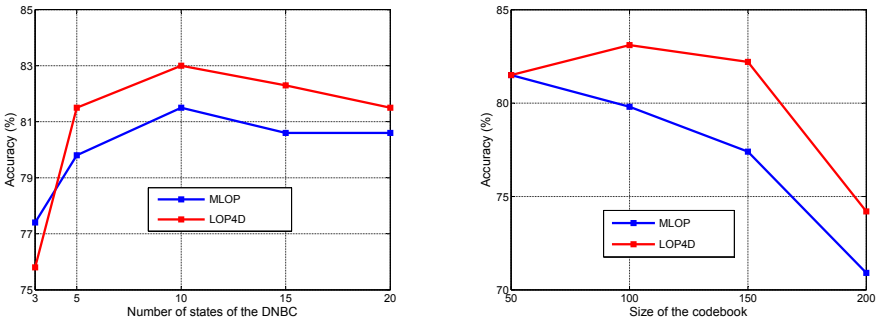


Figure 7: Confusion matrices obtained on CAD-120 using MLOP (a), and 4DLOP (b).

On this dataset, we also evaluate the effectiveness of our method when the value of parameters (size of the codebook and number of DNBC states) is changed. The evolution of the accuracy with respect to both parameters is displayed in Fig. 8 for both MLOP and LOP4D features. First, it can be observed that the proposed method obtains the best accuracy using both features, when a DNBC with 10 states is trained. It can be also observed that the ac-

curacy is relatively independent from the number of states (except when only three states are used). Second, we can notice the best accuracy is obtained with a codebook of size 50 for the MLOP feature, and a codebook of size 100 for the LOP4D feature. In addition, if too much exemplar features (i.e., 200) are used, the accuracy falls down. Indeed, learning a codebook with too much symbols may result in similar activities represented by different symbols. Hence, symbols represent more a particular sequence performed by one subject, than a generic template of one activity class.



(a) (b)

Figure 8: Accuracy evolution of our method with respect to varying parameters: the number of states of DNBC (a), and the size of the codebooks (b).

4.3. Multi-Modal Action database

The Multi-Modal Action Detection (MAD) database [18], has been used to evalaute our method in the online detection task. This RGB-D database has the advantage of including long sequences of 20 subjects performing successively 35 actions, like *Running*, *Throw* and *Kicking*. Since actions are performed without objects, and for a fair comparison with state-of-the-art-methods, we only use skeleton data in these experiments. A five-fold-cross-validation over the 20 subjects is used as evaluation protocol. In each iteration, the labeled sequences of four folds are used to build the vocabulary of MSs and train the DNBCs.

We used the ground truth segmentation in order to separate each action of the training sequences and learn one DNBC per action. One model corresponding to the null class is also learned from transition intervals when the human is standing.

Our method is run in an online way as described in Sect. 3.2. As a result, we obtain a segmented sequence with an action label for each AU corresponding to the action we detected. In order to evaluate the method and compare it with the state-of-the-art, we compute two measures: *Precision*, which corresponds to the percentage of correctly detected actions over all the detected actions; *Recall*, that is the percentage of correctly detected actions over all the ground truth actions. An action is considered as correctly detected if it overlaps with 50% of the segments of the ground truth action. The ground truth provided by the database authors is obtained by manual labeling of sequences. We compare these two measures with the SMMED and MSO-SVM methods, both proposed in [18]. The average and standard deviation values among the five folds are reported in Table 3. We can see that our method outperforms the state-of-the-art approaches for both the measures.

Table 3: MAD database. Comparison of the proposed online detection approach with SMMED [18] and MSO-SVM [18]. The precision and recall measures are computed

Measure (%)	MSO-SVM [18]	SMMED [18]	Our
Recall	51.4	57.4	**79.7 ± 6.4**
Precision	28.6	59.2	**72.1 ± 5.8**

Fig 9 also shows the detection results of one sequence in comparison with the ground truth and the best state of the art method, SMMED, proposed in [18]. We can see that while both our method and [18] are able to accurately detect actions along the time, our method detects more efficiently the end of actions, thus resulting in a duration of detected actions closer to the ground truth. As an overlap of 50% with ground truth is considered as the criterion of

good detection, our method obtains higher values of recall and precision.

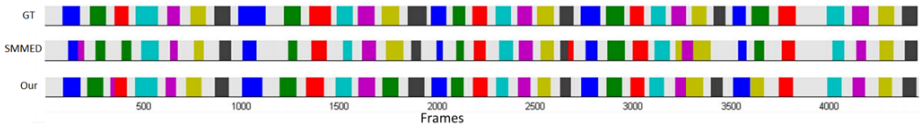


Figure 9: Action detection result, for the sequence-1 of subject-1 from the MAD database, of the SMMED method [18] (second row) and the proposed approach (third row) in comparison to the ground truth (first row). Our method provides segments whose duration is closer to ground truth compared to [18].

4.4. Online RGB-D dataset

The Online RGB-D dataset [35] proposes different types of sequences, which allow evaluation in different contexts, like activity recognition and online activity detection. The dataset contains RGB-D sequences of seven activities, like *drinking*, *eating* or *reading book*. On this dataset, we first evaluate the effectiveness of our method for activity recognition. To this end, we follow the same procedure as in [35] by employing a 2-fold cross validation. We compare our approach with state-of-the-art methods according to the type of features employed. When we use depth features in our method, we use the 4DLOP feature and learn codebooks of different sizes. The best accuracy is obtained for a codebook of size 100. Results are reported in Table 4.

Table 4: Online RGB-D dataset. Comparison of our approach with state of the art methods for the task of activity recognition

Method	Accuracy (%)		
	Depth	*Skeleton*	*Depth + Skeleton*
DCSF [16]	61.7	-	-
Moving Pose [15]	-	38.4	-
Actionlet [33]	-	-	66.0
DOM [35]	46.4	63.3	71.4
Our	**64.5 ± 0.7**	**71.8 ± 1.8**	**80.9 ± 1.1**

It can be noticed that the proposed approach outperforms the state-of-the-art methods for every combination of features. It should also be noted that if

only depth features are used, our method is not fairly comparable to the others. Indeed, even if we only use depth features to describe MSs, our method still needs skeleton data to identify MSs. Nevertheless, we can see that our segmentation approach allows a good recognition of activities when each segment is only described by depth appearance feature. Compared to skeleton-based methods, our approach significantly outperforms other solutions. This shows that our segmentation approach combined with shape analysis of human motion allows us to efficiently recognize activities involving manipulation of objects. Even without considering any information about objects held by the subject, we are able to recognize 71.8% of the activities. This result is higher than that scored by [33] and [35], which combine both skeleton and depth features. Finally, if we add depth features to the skeleton, the recognition accuracy is increased to 80.9%, which is almost 10% above the best state-of-the-art method [35].

We evaluate also the latency of our approach by measuring the ability to recognize the activity without observing the whole sequence. Hence, the average recognition accuracy is computed on different observed portions of the sequence, as reported in Fig. 10 in comparison to state-of-the-art. We can notice that the proposed approach outperforms the methods in [16] and [15] for every observation ratio. However, our method exceeds the method proposed in [35] from 40% of observation. Indeed, when we observe less than 40% of the sequence, it often results in activity sequences represented by one or two temporal segments. In these cases, the dynamics of the activity is null (one observation) or very small (two observations). Hence, the use of statistical models like DNBC is not appropriate and efficient for modeling short portions of the activity sequence. Finally, our method allows efficient recognition when half of the sequence is observed (accuracy of 75.6%). This shows that even if our method is not suitable for very early detection of activities (less than 30% of observation), we guarantee a good

recognition accuracy when only half of the sequence is observed.

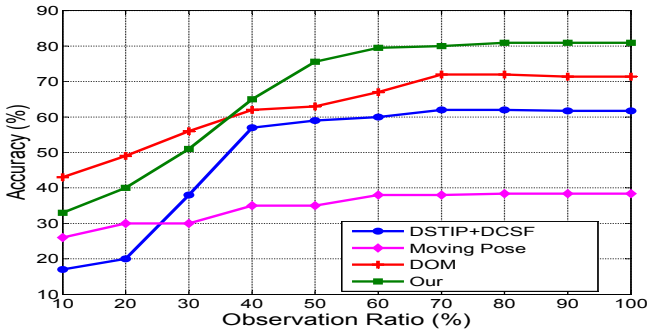


Figure 10: Latency analysis on Online RGB-D dataset. Accuracy obtained for different portion of the sequences is compared to and state-of-the-art methods.

Finally, we propose to evaluate our approach for online activity detection. The same set of activities as for activity recognition is used to train one DNBC for each activity class. In addition, we use a set of background activities provided by the dataset, so as to learn the null class. Finally, we run our detection method on a new set of sequences. It includes 36 long sequences from $30sec$ to two minutes, where 12 new subjects are successively performing different activities. Manual labeling provided by the dataset is used as ground truth. Detection is evaluated using a frame-level accuracy as in [35], computed by averaging the number of well classified frames out the all set of frames in the test sequences. Results are reported in Table 5. We can see that our method performs better than state-of-the-art approaches to detect activity in an online manner. Using an unoptimized Matlab implementation with an Intel Core i-5 2.6GHz CPU and a 8GB RAM, we run our detection method at $7fps$.

5. Conclusions

In this paper, we propose an effective method for modeling and understanding human behavior, like gestures, actions and activities. Thanks to a pose-based shape analysis, we decompose a sequence into relevant MSs. On the one

Table 5: Online RGB-D Dataset. Comparison of our approach with state of the art methods for the task of online activity detection

Method	Accuracy (%)
DSTIP + DCSF [16]	32.1
Moving Pose [15]	50.0
DOM [35]	56.4
Our	**60.9**

hand, such MSs are represented as motion trajectories and interpreted in the Riemannian shape space in order to capture the dynamics of human motion. On another hand, we add depth appearance information in order to characterize possible objects manipulation across MSs. The combination of skeleton and depth data, as well as the modeling of the dynamics of the sequence of MSs is done through a Dynamic Naive Bayes Classifier. Experiments on several datasets show the potential of our method for the task of human behavior recognition in comparison with state-of-the-art. Finally, we adapt our method to allow online behavior detection in long sequences, which is an important challenge in real-world contexts. Evaluation on two datasets demonstrate that the proposed approach outperforms state-of-the-art methods for online detection of human behavior. As future work, we plan to investigate more in detail the online detection problem and more specifically the early behavior detection.

Acknowledgment

This work is partially supported by the FUI project MAGNUM 2 and the Programme d'Investissements d'Avenir (PIA) and Agence Nationale pour la Recherche (grant ANR-11-EQPX-0023), and European Founds for the Regional Development (Grant FEDER- Presage 41779). A very preliminary version of this work appeared in [48].

References

[1] Y. Tian, L. Cao, Z. Liu, Z. Zhang, Hierarchical filtered motion for action recognition in crowded videos, IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews 42 (2012) 313–323.

[2] B. Solmaz, B. E. Moore, M. Shah, Identifying behaviors in crowd scenes using stability analysis for dynamical systems, IEEE Trans. on Pattern Analysis and Machine Intelligence 34 (2012) 2064–2070.

[3] W. Ge, R. T. Collins, R. B. Ruback, Vision-based analysis of small groups in pedestrian crowds, IEEE Trans. on Pattern and Analysis Machine Intelligence 34 (2012) 1003–1016.

[4] O. Arandjelović, Contextually learnt detection of unusual motion-based behaviour in crowded public spaces, in: Int. Symp. on Computer and Information Sciences, 2011, pp. 403–410.

[5] N. Buch, S. A. Velastin, J. Orwell, A review of computer vision techniques for the analysis of urban traffic, IEEE Trans. on Intelligent Transportation Systems 12 (2011) 920–939.

[6] S. Hadfield, R. Bowden, Kinecting the dots: Particle based scene flow from depth sensors, in: Int. Conf. on Computer Vision (ICCV), Barcelona, Spain, 2011, pp. 2290–2295.

[7] Z. Ren, J. Yuan, Z. Zhang, Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera, in: ACM Int. Conf. on Multimedia, Scottsdale, Arizona, USA, 2011, pp. 1093–1096.

[8] K. A. Funes Mora, J.-M. Odobez, Person independent 3D gaze estimation from remote RGB-D cameras, in: IEEE Int. Conf. on Image Processing, 2013, pp. 2787–2791.

[9] R. S. Ghiass, O. Arandjelović, D. Laurendeau, Highly accurate and fully automatic head pose estimation from a low quality consumer-level RGB-D sensor, in: Work. on Computational Models of Social Interactions: Human-Computer-Media Communication, 2015, pp. 25–34.

[10] L. Sun, Z. Liu, M.-T. Sun, Real time gaze estimation with a consumer depth camera, Information Sciences 320 (2015) 346–360.

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, Real-time human pose recognition in parts from single depth images, in: IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, Colorado, USA, 2011, pp. 1–8.

[12] J. Barbic, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, N. S. Pollard, Segmenting motion capture data into distinct behaviors, in: Graphics Interface, 2004.

[13] F. Zhou, F. De la Torre, J. K. Hodgins, Hierarchical aligned cluster analysis for temporal clustering of human motion, IEEE Trans. on Pattern Analysis and Machine Intelligence 35 (2014) 582–596.

[14] I. Kapsouras, N. Nikolaidis, Action recognition on motion capture data using a dynemes and forward differences representation, Journal of Visual Communication and Image Representation 25 (2014) 1432–1445.

[15] M. Zanfir, M. Leordeanu, C. Sminchisescu, The moving pose: An efficient 3D kinematics descriptor for low-latency action recognition and detection, in: IEEE Int. Conf. on Computer Vision (ICCV), Sydney, Australia, 2013, pp. 2752–2759.

[16] L. Xia, J. K. Aggarwal, Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera, in: CVPR Work. on Human Activity Understanding from 3D Data, Portland, Oregon, USA, 2013, pp. 2834–2841.

[17] H. S. Koppula, R. Gupta, A. Saxena, Learning human activities and object affordances from RGB-D videos, Int. Journal of Robotics Research 32 (2013) 951–970.

[18] D. Huang, Y. Wang, S. Yao, F. D. la Torre, Sequential max-margin event detectors, in: Eu. Conf. on Computer Vision (ECCV), Zurich, Switzerland, 2014, pp. 410–424.

[19] X. Yang, Y. Tian, Eigenjoints-based action recognition using naive-bayes-nearest-neighbor, in: Work. on Human Activity Understanding from 3D Data, Providence, Rhode Island, 2012, pp. 14–19.

[20] J. Luo, W. Wang, H. Qi, Group sparsity and geometry constrained dictionary learning for action recognition from depth maps, in: IEEE Int. Conf. on Computer Vision (ICCV), Sydney, Australia, 2013, pp. 1809–1816.

[21] H. Chen, G. Wang, J.-H. Xue, L. He, A novel hierarchical framework for human action recognition, Pattern Recognition (2016) Available online: http://dx.doi.org/10.1016/j.patcog.2016.01.020.

[22] R. Vemulapalli, F. Arrate, R. Chellappa, Human action recognition by representing 3D skeletons as points in a Lie group, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014, pp. 588–595.

[23] R. Slama, H. Wannous, M. Daoudi, A. Srivastava, Accurate 3D action recognition using learning on the Grassmann manifold, Pattern Recognition 48 (2015) 556–567.

[24] R. Anirudh, P. Turaga, J. Su, A. Srivastava, Elastic functional coding of human actions: From vector-fields to latent variables, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 3147–3155.

[25] X. Yang, C. Zhang, Y. Tian, Recognizing actions using depth motion maps-based histograms of oriented gradients, in: ACM Int. Conf. on Multimedia, Nara, Japan, 2012, pp. 1057–1060.

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

[27] J. Wang, Z. Liu, J. Chorowski, Z. Chen, Y. Wu, Robust 3D action recognition with random occupancy patterns, in: Eu. Conf. on Computer Vision (ECCV), Florence, Italy, 2012, pp. 1–8.

[28] H. Rahmani, A. Mahmood, D. Q. Huynh, A. Mian, Hopc: Histogram of oriented principal components of 3D pointclouds for action recognition, in: Eu. Conf. on Computer Vision (ECCV), Zurich. Switzerland, 2014, pp. 742–757.

[29] O. Oreifej, Z. Liu, HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Portland, Oregon, USA, 2013, pp. 716–723.

[30] X. Yang, Y. L. Tian, Super normal vector for activity recognition using depth sequences, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014, pp. 804–811.

[31] S. Althloothi, M. H. Mahoor, X. Zhang, R. M. Voyles, Human activity recognition using multi-features and multiple kernel learning, Pattern Recognition 47 (2014) 1800–1812.

[32] C. Lu, J. Jia, C.-K. Tang, Range-sample depth feature for action recognition, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014, pp. 772–779.

[33] J. Wang, Z. Liu, Y. Wu, J. Yuan, Mining actionlet ensemble for action recognition with depth cameras, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Providence, USA, 2012, pp. 1–8.

[34] P. Wei, Y. Zhao, N. Zheng, S.-C. Zhu, Modeling 4d human-object interactions for event and object recognition, in: Int. Conf. on Computer Vision (ICCV), Sydney, Australia, 2013, pp. 3272–3279.

[35] G. Yu, Z. Liu, J. Yuan, Discriminative orderlet mining for real-time recognition of human-object interaction, in: Asian Conf. on Computer Vision (ACCV), Singapore, 2014, pp. 50–65.

[36] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, A. Del Bimbo, 3D human action recognition by shape analysis of motion trajectories on Riemannian manifold, IEEE Trans. on Cybernetics 45 (2014) 1340–1352.

[37] S. H. Joshi, E. Klassen, A. Srivastava, I. Jermyn, A novel representation for Riemannian analysis of elastic curves in R^n, in: IEEE Int. Conf. on Com-

puter Vision and Pattern Recognition (CVPR), Minneapolis, MN, USA, 2007, pp. 1–7.

[38] J. Bai, J. Goldsmith, B. Caffo, Movelets: A dictionary of movement, Electronic Journal of Statistics 6 (2012) 559–578.

[39] H. Karcher, Riemannian center of mass and mollifier smoothing, Comm. on Pure and Applied Math. 30 (1977) 509–541.

[40] A. Srivastava, E. Klassen, S. H. Joshi, I. Jermyn, Shape analysis of elastic curves in euclidean spaces, IEEE Trans. on Pattern Analysis and Machine Intelligence 33 (2011) 1415–1428.

[41] A. Srivastava, S. Joshi, W. Mio, X. Liu, Statistical shape analysis: Clustering, learning, and testing, IEEE Trans. on Pattern Analysis and Machine Intelligence 27 (2005) 590–602.

[42] M. Martinez, L. E. Sucar, Learning dynamic naive bayesian classifiers, in: Int. FLAIRS Conf., 2008, pp. 655–659.

[43] L. Baum, T. Petrie, G. Soules, N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains, Ann. Math. Stat. 41 (1970) 164–171.

[44] H. Avils-Arriaga, L. Sucar-Succar, C. Mendoza-Durn, L. Pineda-Corts, A comparison of dynamic naive bayesian classifiers and hidden markov models for gesture recognition, Journal of Applied Research and Technology 9 (2011) 81–102.

[45] A. Lehrmann, P. Gehler, S. Nowozin, Efficient nonlinear markov models for human motion, in: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 2014, pp. 1314–1321.

[46] H. Koppula, A. Saxena, Learning spatio-temporal structure from RGB-D videos for human activity detection and anticipation, in: Int. Conf. on Machine Learning (ICML), Atlanta, Georgia, 2013, pp. 792–800.

[47] L. Rybok, B. Schauerte, Z. Al-Halah, R. Stiefelhagen, Important stuff, everywhere! activity recognition with salient proto-objects as context, in:

IEEE Winter Conf. on Applications of Computer Vision (WACV), Steamboat Springs, CO, 2014, pp. 646–651.

[48] M. Devanne, A. Wannous, S. Berretti, P. Pala, M. Daoudi, A. Del Bimbo, Combined shape analysis of human poses and motion units for action segmentation and recognition, in: IEEE Int. Conf. and Workshops on Automatic Face and Gesture Recognition, Ljubljana, Slovenia, 2015.