Deep-BEJT: A New Human Activity Recognition System based on Beta Elliptical Joint Trajectory (BEJT) and Long Short Term Memory (LSTM)

Sawsen Ghabri*, Wael Ouarda, Houcine Boubaker, Adel M. Alimi

University of Sfax, National Engineering School of Sfax (ENIS), REGIM-Lab.: REsearch Groups in Intelligent Machines, LR11ES48, 3038, Sfax, Tunisia

Abstract

This research article aims to introduce a novel approach for Human Activity Recognition (HAR) from 3D skeletal sequences captured by Kinect sensor that can operate in low light and solve the problem of body detection and human tracking. Latter approaches in 3D human activity recognition outstanding performance and gain much interest thanks to the effectiveness of skeleton data such as joint trajectory that is chosen as source of different HAR models. 3D Skeleton model removes the factors of the invariance of view, scale and background that both the original and depth image have. Beta elliptic model can be employed in HAR system thanks to its power to characterize the joint trajectory by merging the dynamic and the geometric description. Therefore, we propose a deep learning approach for action recognition problem based on 3D skeletal representation, Beta Elliptic model and a special kind of Recurrent Neural Networks called Long Short-Term Memory (LSTM). Beta Elliptical Joint Trajectory (BEJT) is a novel feature that describes both geometric and dynamic characteristics of skeletal sequences with elliptical arcs. Additionally, to enhance the success of our proposal, the classification of human activities is carried out according to two layers LSTM framework since its capability to model the long-term temporal dependencies automatically. Relying on an experimental evaluation of the proposed methods for human activity recognition based on BEJT features and LSTM classifier on four data sets; UT-Kinect, MSR-Action 3D, CAD60 and CAD120, we have validated the performance of our proposed method compared to the existing ones.

Keywords: 3D human activity recognition, Joint trajectory, Beta Elliptic model, BEJT, LSTM

*Corresponding author.

Email addresses: sawsen.ghabri.s@ieee.org (Sawsen Ghabri), wael.ouarda@ieee.org (Wael Ouarda), houcine-boubaker@ieee.org (Houcine Boubaker), adel.alimi@ieee.org (Adel M. Alimi)

1. Introduction

Human action recognition has stilled one of major challenging and important tasks in computer vision and machine learning. It promotes an extensive range of applications such as intelligent video surveillance, video understanding and human computer interaction. Recognizing human actions aims to determine the actions from input sensor streams, where RGB [1, 2], depth [3, 4] and skeleton [5, 6] are three input types. The most popular input is the RGB videos; it has been widely studied. Withal, information that represent human activities and captured in the 3D space is richer. There are many sensors that are used to recollect information of a person performing activities, sensors like accelerometers [7], microphones [8], video cameras [9] and other means. Generally, depth cameras produce better quality 3D depth data than those estimated from monocular video sensors. Skeleton information recollected from Microsoft Kinect are a good source of data because they are not affected by environment light variations. they can provide body shape and simplify the problem of body detection and human tracking [1]. It is easier to use the skeletal data than RGB information in HAR systems because skeleton model could remove the factors of the invariance of view, scale and background that the original image and the depth image have. Also, it eliminates problems related to clothing textures, color and hair shapes, which greatly simplify the learning of action recognition itself.

In this work, a human action recognition system, exploiting the 3D skeletal joint positions provided by Microsoft Kinect, is proposed.

Our algorithm starts from a 3D skeletal representation as an input. Based on the skeleton joints positions in each sequence frames, we extract the joint trajectory that can represent the skeleton motion to construct the new feature BEJT using the Beta Elliptic model [10]. The motivation for the use of Beta Elliptic model is the richest output in terms of kinematic, graphical and biometrical data. Indeed, Beta-elliptic model is characterizing by a description combining the dynamic and the geometric profiles modeling [11]. In the current skeleton-based action recognition, literature (RNN) [12, 13] has a great influence on processing video sequence. Several works [14, 15, 16] successfully built well-designed multi-layer RNNs for recognizing action based on skeletons [17]. However, there are often strong dependency relations among the skeletal joints in spatial domain. The spatial dependency structure is also usually discriminative for action classification. In this paper, Long Short-Term Memory (LSTM) network is adopted to take advantage of its powerful ability to model the long-term contextual information in the temporal domain. It has been successfully applied to language modeling [18], RGB based video analysis [19 .. 27], and skeleton based action recognition [28], [29], [30]. One of the problems in recognition is the availability of data. Accurate data like MoCap, CMU MoCap and HDM05 [31], is expensive to acquire. Recently, Microsoft Kinect and other low-cost sensors have provided the depth data with acceptable accuracy, [32] evolving a real time process to get 3D joint position from a single depth image. Despite extensive training on synthetic data extracting the joint locations in real-time has become a doable task. With all these available data, human activity recognition based joint trajectory reconstructed with 3D joints positions has become tractable. Over this paper, we use the 3D skeletal joint locations to expand an efficient action recognition approach. We

introduce a novel descriptor named Beta Elliptic Joint Trajectory (BEJT) based on beta elliptic modelization. The motivation for the use of Beta Elliptic model [10, 11] is the rich output in terms of kinematic, graphical and biometrical data. Indeed, Beta-elliptic model is characterizing by a description combining the dynamic and the geometric profiles modeling [33]. We will show how the proposed descriptor is efficient and discriminative on popular 3D action recognition data sets: MSR-Action3D, UT-Kinect, CAD 60, ... then, we will evaluate the performance on LSTM because of his capability to learn the long- term dependencies defined by the limitation of processing long term information where the Beta Elliptic joints Trajectories are the input. Finally, we will sum up the main contributions of this document as follows: Firstly, BEJT is a novel local descriptor that represent the skeleton movement based on beta elliptic model and joint trajectories. Secondly, it is an integrated system combing the advantages of new dynamic features and stacked LSTM model for skeleton-based action recognition. Last but not least, The suggested method perform state-of-the art achievements on four benchmark data sets, the rest of this paper is orderly as following: In section 2, we introduce the related works on skeleton-based activity recognition using recurrent neural networks (LSTM)to model the temporal dynamics. Our approach is depicted in Section 3. Section 4 presents the data sets used with the experimental results. Finally, the paper is concluded in section 5.

2. Related work

Human activity recognition is an active area of research, with many existing algorithms. Surveys by Weinland *et al.* [14] and Poppe [15] exploring the vast literature in activity recognition. Here, we will focus on recent technologies and approaches for 3D skeleton-based action recognition and recent related advances in deep learning.

2.1. Skeleton-Based 3D Action Representation

The geometric structures of body motion patterns 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 [34], [35] [36] as, by knowing the position of multiple body parts, we want the machine to learn to discriminate among action-classes. Technologies are evolving fast, and the very recent wide diffusion of cheap depth camera, and the seminal work by Shotton, *et al.* [37] 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.

3D action representation based on human skeleton can be generally splitted into three categories [38]: joints, joints group, and joint dynamics. To capture the correlation of body's joints in the joint representation, it is necessary to extract spatial descriptor [39], [40], geometric descriptor [41], [42], [43], [44], [45] or key poses [46], [47], [48]. Joints group detect the discriminative subsets of joints to distinguish actions. Joint dynamics focused on modeling the dynamics of either subsets or all skeleton joints. In [49], 3D joint trajectories are projected into three 2D trajectories, and the oriented displacement histogram is calculated to

depict the three trajectories. Each trajectory displacement polling its length in the histogram of angles orientation. Chaudhry et al. [50] proposed to divide the fully human skeleton into various body parts depicted by joints, comprising the upper body, lower body, left/right arms and left/right legs. The computing of shape context feature started by regarding the directions of a set of equidistant points sub-sampled beyond the segments of each body part. Finally, the skeletal sequence is described as a set of time series features (position, tangent and shape context feature). Then, they divided these time series into different temporal scales modeled by a linear dynamic system. So, to describe the skeleton dynamics in the skeletal sequence all estimated parameters series are used. In [51], the authors characterized the 3D joint coordinates and their changing over time as a trajectory in the Riemannian Manifold, and the human action recognition is considered as the problem of computing the similarity between the trajectories shape. Wang et al. [52] used the color to encode the trajectory's dynamics, and model the spatial-temporal data carried in a skeleton sequence across shape and textures. They depicted the spatio-temporal information on three 2D images through coding the joint trajectories and their dynamics into color distribution. This method called Joint Trajectory Maps where they adopted ConvNets to learn the discriminative features to recognize the human. Hai *et al.* [53] 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 auto regressive 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.

2.2. Direct Acquisition of 3D Skeletal Data

Different commercial devices, including motion capture systems, time-of-flight sensors, and structured-light cameras, enabled the direct retrieval of 3D skeleton information. The human body models obtained by these devices are shown in Figure 1.

Fig. 1. Examples of skeletal human body models obtained from different devices. The OpenNI library tracks 15 joints; Kinect v1 SDK tracks 20 joints and MoCap systems can track several numbers of joints.

Motion Capture Systems (MoCap) [54] obtained the 3D skeleton information by identifying and tracking markers attaching to a human joints or body parts. It exists two important categories of Mo-Cap systems based on inertia sensors or visual cameras. In inertial sensors based MoCap systems, the rotation of a body part is estimated by 3-axis inertial sensor respecting a fixed point. The collection of this information aims to obtain the skeleton data without any optical devices. Commercial MoCap systems provided a software collecting skeletal data but such systems are so expensive and unable to be used outside a greatly-controlled indoor environments. Compared to other sensors, the Microsoft Kinect camera affords an affordable alternative to obtain depth information [55]. Moreover, a color camera is incorporated into the sensor to obtain the color data registered. The color depth information can be reached by the Kinect-SDK 2.0 or the OpenKinect library[56]. The Kinect camera offers an important resolution of depth images (512 424) at 30 Hz. In addition, this camera can estimate the positions of 25 human body joints to provide 3D skeleton information, with better tracking accuracy. Also, using Kinect SDK gives many advantages. It provides 3D human skeletal data in real time using the estimation method described by Shotton et.al [57]. This sensor is also inexpensive and eliminates the body tracking and background extraction problems.

2.3. Skeleton Based Action Recognition with RNN and LSTM Models

Du et al. [8] propounded an end-to-end RNN with handcrafted subnets. According to human structure, the raw positions of body joints are divided into to five parts singly fed into five bidirectional RNNs. When the layers number increases, the network hierarchically fused the representations extracted by the subnets to a higher-level representation.

The classical RNN has a problem to learn the long-term representation of video sequences because of the exponential decrease in retaining the context information of video frames [58]. To surpass this limitation, Long Short-Term Memory (LSTM) [59], a kind variant of RNN created to learn the long-range dependency between the input frame and the output label. LSTM networks achieved competitive performance on human activity recognition task [60][61]. Zhu et al. [67] used LSTM to represent a skeletal sequence as a feature vector. At each step, the LSTM input consisting of the 3D concatenated locations of skeletal joints in a frame. They modeled a feature manifold due to a set of encoded feature vectors. Finally, the manifold was utilized to aid and regularize the supervised learning of LSTM for action recognition based on RGB video. In [68] and[69], a 2D Spatio-Temporal LSTM framework is created to explore concurrently the hidden sources of activity related context information in the temporal and spatial domains. They also proposed a trust gate mechanism aims to deal with the inaccurate 3D joints coordinates provided by the depth sensors.

3. A Beta Elliptical Joint Trajectory (BEJT) System for HAR

The aim of building a human representation based on 3D skeleton is to extract compact, discriminative descriptions to characterize human impute from 3D human skeletal data that can be acquired, comprising devices that directly assist the skeletal information and computational methods to build the skeleton. Thus, the human body is encoded as an

articulated system of rigid segments attached by joints. In our method, informations in the raw joint position are used to form a joint trajectory, and then we extract features from this trajectory, which are often labeled the trajectory-based representation. Indeed, Beta-elliptic model is characterizing by a description combining the dynamic and the geometric profiles modeling. The proposed technique comprises three main stages including 3D skeletal data preprocessing, Beta Elliptic Joint Trajectory extraction and classification step based on multi-layer LSTM.

Fig. 2. Human Activity Recognition System based on Beta Elliptical Joint Trajectory (BEJT) and Long Short Term Memory (LSTM)

3.1. Joint Position Estimation and 3D skeletal data preprocessing

The human body can be defined as a set of rigid segments connected by skeletal joints to form an articulated system and the human action is considered as a continuous evolution of the spatial configuration of these segments (i.e. body postures) [70].

Fig. 3. (a) represent a Depth image and (b) represent a Skeletal joint locations by Shotton et al.'s method

Here, we use joint locations to build a compact representation of postures. The launch of Kinect offers a low-cost and real-time solution for the estimation of the 3D locations of objects or persons in the scene. The launch of Kinect offers a low-cost and real-time solution for the estimation of the 3D locations of objects or persons in the scene. Shotton *et al.* [57] propose to extract 3D body joint locations from a depth image using an object recognition scheme. The human body is labeled as body parts based on the per-pixel classification results. The parts include 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 (Left, Right, Upper, Lower). They compute the confidence-scored 3D position estimation of body joints by employing a local mode finding approach based on the mean shift with a weighted Gaussian kernel. Their gigantic and diverse training set allows the classifier to estimate body parts invariant of pose, body shape, clothing, and so on. Using their algorithm, we acquire the 3D locations of 20 skeletal joints which comprise hip center, spine, shoulder center, head, Left/ Right shoulder, Left/ Right elbow, Left/ Right wrist, Left/ Right hand, Left/ Right hip, Left/ Right knee, Left/ Right angle and Left/ Right foot. Figure.3 shows an example result of 3D skeletal joints and the corresponding depth map.

3.2. Feature Extraction

Using the Microsoft Kinect SDK, we can easily obtain in real-time a 3D skeleton by following the approach of Shotton *et al.* [57]. This skeleton contains the 3D position of a certain number of joints representing different parts of the human body. The number of estimated joints that depends on the SDK used in combination with the device extracted in our approach is 20 joints. We explore the joint positions as gesture representation, and we model the dynamics of the full skeleton as a trajectory.

3.2.1. 3D Joint Trajectory Construction

For each frame of a sequence, the real-world 3D position of each joint of the skeleton is represented by three coordinates expressed in the camera reference system.

$p_i(t) = (x_i(t), y_i(t), z_i(t))$ (1) The trajectory is defined by the motion over time of the feature point encoding the 3D coordinates of all the joints of the skeleton (or by all the feature points coding the body parts separately).

The 3D joints trajectories constructed by the position of joint J in frame 1 to frame N where N was the last frame in the skeletal sequence is defined by the equation below:

$$J - t = \sum_{t=0}^N J - c(2)$$

J-c: the 3D coordinates (x, y, z) of joint J defined by:

$$J - c(J) = (J - x, J - y, J - z) \quad (3)$$

To construct the 3D Skeleton Trajectory, we focus on modeling the dynamics of all skeletal joints utilizing the raw joint position information (the coordinates of joint in each frame of the 3D skeletal sequence). Figure 3 show the dynamic of some skeletal joints (Head, Right Hand and Spine).

Fig. 4. Examples of 3D Joint Trajectory: (a) show the trajectory of the Hed joint; (b) show the trajectory of the Right-hand joint and (c) show the trajectory of the Spine joint.

By concatenating the 3D coordinates of skeleton joints, data representation encodes the shape of the human posture at each frame. The dynamics of human motion was captured by modeling the sequence of frame features along the action as a trajectory. Let N_j be the number of joints composing the skeleton, the posture of the skeleton at frame t is represented by a $3N_j$ dimensional tuple:

$$c(t) = [x1(t) \ y1(t) \ z1(t) \ ...xNj(t) \ yNj(t) \ zNj(t)]^T \quad (4)$$

For an action sequence composed of N_f frames, N_f joint position vectors are extracted and arranged in columns to build a skeleton joints coordinates matrix M_c ($3N_j * N_f$) describing the whole sequence.

$$M_c = (c(1) \ c(2) \ ...c(N_f)) \quad (5)$$

Thus, the skeleton joints coordinates matrix M_c represents the evolution of the skeleton joints pose over time. Each column vector is regarded as a sample of a continuous trajectory of a single body joint.

3.2.2. Trajectory Filtering and Resampling

First of all, the resampling concerns the increase in the number of points defining the skeletal trajectory extracted in the case where their number is insufficient to apply the Beta elliptical extraction module. We respect the velocity of the original trajectory because resampling depends on this velocity.

Fig. 5. (a) Resampling and filtering Trajectory (b): Velocity profile modeling by overlapped Beta impulses

Physiological tremor is present in all normal and healthy subjects and is exhibited in different conditions, such as various task execution (motion or isometric contraction), posture maintenance and even at rest. It can be found in spatial and temporal domains. In the frequency domain, it can be divided into three ranges: low frequency (less than 4 Hz), intermediate frequency(8 ~ 12 Hz) and high frequency (greater than 12 Hz), according to the tremble frequency. The most common physiological tremor has a tremor frequency of 8 ~ 12 Hz [1]; This is the reason of our frequency choice: $f_{cut} = 8\text{Hz}$.

3.2.3. Beta Elliptical Joint Trajectory Reconstruction

In this paper a new formalism is proposed to describe the trajectory of the skeleton by resorting to an extension in 3D of an efficient 2D representation called beta elliptic model [33]. The beta -elliptic approach for 3D skeletal movement representation consists in modeling the trajectory by a combination of two aspects of profiles: dynamic and static. In the dynamic profile, overlapped Beta signals model the curvilinear velocity, whereas in the static profile named also geometric profile, elliptic arcs model trajectory segments. The combination of both types of features characterizing the extracted Beta impulse and its corresponding arc of ellipse allows decomposing a complex trajectory in elementary segments called ‘strokes’.

3D trajectory modeling based on beta-elliptical representation. The 3D joint trajectory modeling based on beta-elliptic approach goes through three steps. The first consists of the segmentation of the trajectory into strokes according to velocity extreme, i.e. local minimum and maximum, and double-inflection point of curvilinear pulse velocity defined by:

$$V_{\sigma}(t) = \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2 + \left(\frac{dz}{dt}\right)^2} \quad (6)$$

Fig. 6. (a) show the 3D trajectory segmentation in Beta stroke and (b) represent an example of beta elliptical joint trajectory reconstruction by two types of elliptical arcs

The monotonic variation of the curvilinear velocity during an elliptical beta segment also guarantees a monotonic variation of the radius of curvature on the corresponding trajectory line. Depending on the level of approximation, each segmented trajectory is then approximated by a planar trajectory.

We assume that the trajectory does not change planes appreciably until moments of velocity extremes, ie when passing from one segment to another. Thus, this hypothesis suggests that each segment of the trajectory is substantially approximate to a plane shape. On the other hand, studies carried out in the literature [70, 71, 72] on the flat trajectories of rapid movements of human bio-motor systems (arm-hand system, locomotion system, eyes, etc.) show that their trajectories are approximate to elliptical arcs.

A given segment is defined by all points of its sampled trajectory:

$$S_k : \{M_i, i = n, \dots, m\} \quad (7)$$

The plane P_k that supports approximately the trajectory segment S_k could be estimated precisely by a plane regression using the least square error algorithm, however, to avoid any discontinuity in the inter-strokes of the reconstructed hand trajectory path and to reduce the computation time a direct estimation method is adopted that considers as elements belonging to P_k the following three-most distant points in the S set:

The beginning point $M_n(x_n, y_n, z_n)$

The last point $M_m(x_m, y_m, z_m)$

The most distant point $M_j(x_j, y_j, z_j)$ of the trajectory segment S_k with respect to the straight line checking thus:

$$M_j \in S_k \text{ maximizing the orthogonal distance between } M_j (M_n; M_m).$$

The plane P_k can be represented by the Cartesian equation below:

$$(a_{P_k} \cdot x) + (b_{P_k} \cdot y) + (c_{P_k} \cdot z) + 1 = 0 \quad (8)$$

The parameters a_{P_k} , b_{P_k} and c_{P_k} can be calculated from the following equations:

$$\begin{cases} (a_{P_k} \cdot x_n) + (b_{P_k} \cdot y_n) + (c_{P_k} \cdot z_n) + 1 = 0 \\ (a_{P_k} \cdot x_m) + (b_{P_k} \cdot y_m) + (c_{P_k} \cdot z_m) + 1 = 0 \\ (a_{P_k} \cdot x_j) + (b_{P_k} \cdot y_j) + (c_{P_k} \cdot z_j) + 1 = 0 \end{cases} \quad (9)$$

The second step consists in projecting the segment into the corresponding plane P_k . In reality, the trajectory of a segment S_k is not perfectly flat. To be approximated by the arc of an ellipse, it must first be projected orthogonally on the plane P_k . To have the whole information about the trajectory tangential direction at the starting and ending points M_m and M_n , the front and rear neighborhoods' trajectory of the segment S_k are also projected on the plane P_k . The Cartesian coordinates of the direction vector of the orthogonal projection on the plane P_k are given by:

$$\vec{v} = \begin{pmatrix} a_{pk} \\ b_{pk} \\ c_{pk} \end{pmatrix} \quad (10)$$

The set of point $S'_k : \{M'_i, i = n, \dots, m\}$ representing the orthogonal projection of the trajectory segment S_k on the plane P_k is composed of the points $M'_i (x'_i, y'_i, z'_i), i = n, \dots, m$ verifying the following condition:

$$\begin{cases} M'_i \in P_k \\ (M_i M'_i // \vec{v}) \end{cases} \quad (11)$$

Hence, from the system of linear equations, it is possible to calculate the coordinates of the point $M'_i (x'_i, y'_i, z'_i)$ according to those of the points $M_i (x_i, y_i, z_i)$, the parameters a_{P_k} , b_{P_k} and c_{P_k} of the P_k plane defined by:

$$\begin{cases} (a_{P_k} \cdot x'_i) + (b_{P_k} \cdot y'_i) + (c_{P_k} \cdot z'_i) + 1 = 0 \\ x'_n - x_n = k c_i \cdot a_{P_k} \\ y'_n - y_n = k c_i \cdot b_{P_k} \\ z'_n - z_n = k c_i \cdot c_{P_k} \end{cases} \quad (12)$$

The third step of the 3D trajectory modeling process is the estimation of the optimal elliptic arc that approximates the obtained set of point S'_k . Since the objective is to estimate the elliptic arc that approximates the projection of the trajectory segment S_k on the plane P_k , we proceed to change its marker analytical description from the original frame of the 3D domain $Rep_{3D}(O, \vec{ox}, \vec{oy}, \vec{oz})$ to the 2D description on the P_k plane.

$$Rep_{2D-K}(M'_n, \frac{\overrightarrow{M_n M'_m}}{\|M'_n M'_m\|}, \frac{\overrightarrow{M_n M'_m}}{\|M'_n M'_m\|}^T) \quad (13)$$

The new orthonormal frame is defined by its origin M'_n , and the direction of its two axes $M'_n M'_m$ and its perpendicular direction. The unit of measurement on the two axes of the new 2D representation is equal to that used for the distance measurements in 3D space. Thus, the points of the projected trajectory segment S'_k pass from a 3D description $M'_i (x'_i, y'_i, z'_i)$ in the original frame Rep_{3D} to a 2D description in the Rep_{2D-k} frame associated to the plane P_k : $M'_i(x_{2D-i}, y_{2D-i})$; where

$$\begin{cases} x_{2D-i} = \text{sign}[\max(\|M'_n M'_m\|, \|M'_n M'_i\|) - \|M'_m M'_i\|] \cdot \|M'_n M''_i\| \\ y_{2D-i} = \text{sign}[z'_i - z''_i] \cdot \|M'_i M''_i\| \end{cases} \quad (14)$$

And $M''_i(x''_i, y''_i, z''_i)$ represent the orthogonal projection of the point M'_i on the straight line $(M'_n M'_m)$.

Then, to estimate the arc of ellipse that approximates the projected trajectory segment S'_k described in the 2D frame Rep_{2D_k} , 2 tangent endpoints' method is used. In what follows we present this method modeled the obtained plot.

Joint Trajectory modeling based onelliptical path limited by two tangents method

. In this method, we consider the limit point M_1 of the line trajectory which coincides with the local maximum velocity is the top of the small axis of the ellipse modeling the trajectory segment $M_1 M_2$. Thus, the direction of the ellipse's major axis is parallel to the tangent to the trajectory at M_1 . Then, contrary to the previous approach, the endpoint of the long axis of the ellipse ϵ is not fixed to the other limit point M_2 of the trajectory segment. Indeed, the ellipse parameters calculation is based on the position of the point M_2 and the angle α as explained below.

Let $M(X_M, Y_M)$ be a point on the ϵ ellipse defined in an orthogonal reference frame (O, X, Y) where one axis is parallel to the ϵ major axis. It verifies:

$$\left(\frac{x-x_0}{a}\right)^2 + \left(\frac{y-y_0}{b}\right)^2 = 1 \quad (15)$$

The angle β between the ellipse main axis and the tangent to the ellipse at a point $M(X_M, Y_M)$ verifies:

$$\text{tg}(\beta) = \text{tg}\left(\frac{\pi}{2} - \alpha\right) = -\left(\frac{b}{a}\right)^2 \cdot \frac{(X_M - X_0)}{(Y_M - Y_0)} \quad (16)$$

Therefore,

$$\text{tg}(\alpha) = -\left(\frac{a}{b}\right)^2 \cdot \frac{(Y_M - Y_0)}{(X_M - X_0)} \quad (17)$$

Assuming that the trajectory segment endpoint $M_2(X_{M2}, Y_{M2})$ belongs to the ellipse ϵ and that the tangent to the ellipse at this point forms an angle α with respect to direction of the small axis, the following relation is obtained:

$$\left(\frac{a_0}{b}\right)^2 + \left(\frac{b-b_0}{b}\right)^2 = 1 \quad (18)$$

$$\text{tg}(\alpha) = -\left(\frac{a}{b}\right)^2 + \left(\frac{b-b_0}{a_0}\right) \quad (19)$$

$a_0 = X_{M2} - X_0$ and $b_0 = Y_{M1} - Y_{M2}$ represent the long axis a_0 and small axis b_0 of the quarter ellipse ϵ_0 , respectively. The latter, defined by the points M_1 and M_2 , is tangent to the modeled trajectory at M_1 .

The evaluation of a and b parameters of the ellipse ϵ according to a_0 , b_0 and α is obtained from equations (18) and (19) as follows:

$$a = \frac{a_0 \cdot b_0}{\sqrt{2 - \left(\frac{2a_0 \cdot b_0}{\text{tg}(\alpha)}\right)}} \text{ and } b = \frac{a_0 \cdot b_0}{\text{tg}(\alpha)} \quad (20) \text{ and } (21)$$

The coordinates of the center $C(x_0, y_0)$ of this ellipse in the original coordinate system $O(\vec{i}, \vec{j})$ are given by:

$$\begin{cases} x_0 = X_0 + ((b - b_0) \cdot \sin \theta_0) \\ y_0 = Y_0 + ((b - b_0) \cdot \cos \theta_0) \end{cases} \quad (22)$$

a_0 and b_0 are calculated according to the coordinates of $M_1 (x_1, y_1)$ and $M_2 (x_2, y_2)$ in the original coordinate system $O(\vec{i}, \vec{j})$ as well as the inclination angle of the tangent θ_0 at the point M_1 .

$$a_0 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \cdot \cos(\theta_0 - \text{Arctg}(\frac{y_2 - y_1}{x_2 - x_1})) \quad (23)$$

$$b_0 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \cdot \sin(\theta_0 - \text{Arctg}(\frac{y_2 - y_1}{x_2 - x_1})) \quad (24)$$

It appears that the calculation of the ellipse parameters using this approach takes into account the position of the two endpoints of the arc $M_1(x_1, y_1)$ and $M_2(x_2, y_2)$ and the inclination angles of the tangent at these points $\theta_1 = \theta_0$ and $\theta_2 = \theta_0 + \beta = \theta_0 + \frac{\pi}{2}$ hence the name of the presented method.

Finally, the 2D analytical description of the elliptic arc approximating the path segment S_k on the plane P_k is mapped to a 3D reference frame. The referential changes that regard the coordinates of the ellipse center $C(x_{C_2D}, y_{C_2D})$ are computed by the reciprocate transformation of the 2D planar conversion. In the 2D reference frame, θ_{2D} is the angle between the ellipse major axis and the first coordinate axis of Rep_{2D_k} , where in the 3D reference frame, θ_{xy} and θ_{xz} are respectively the inclination angles of the elliptic arc major

Table 1: Completed description of the beta elliptical joint trajectory model parameters

Parameter	Description
DELTA_T	Duration of the Beta pulse (Delta_T = t1 - t0)
RAP_Tc	Symmetry ratio of the Beta pulse, defining its peak time tc (RAP_Tc = (tc - t0) / Delta_T)
P	Parameter defining the width of the Beta pulse
K	Beta pulse amplitude
a	Half major axis of the ellipse carrying the arc modeling the trajectory segment
b	Half minor axis of the ellipse carrying the arc modeling the trajectory segment [M ₁ , M ₂]
TETA	Inclination of the axis of the ellipse with respect to the direction [M ₁ , M ₂]
a _{PK}	Parameter defining the plane carrying the arc of the ellipse
b _{PK}	Parameter defining the plane carrying the arc of the ellipse
c _{PK}	Parameter defining the plane carrying the arc of the ellipse
X_Origin	X coordinate of the starting point of the arc
Y_Origin	Y coordinate of the starting point of the arc
Z_Origin	Z coordinate of the starting point of the arc
θ_{xy}	Inclination relative to the axis OX of the projected axis of the ellipse on the plane [OX, OY]
θ_{xz}	Inclination relative to the axis OY of the projected axis of the ellipse on the plane [OY, OZ]

axis projection on the planes (oxy) and (oxz). On the other hand, the measures of the major

axis a and the minor axis b remain unchanged. In the spatial domain, each trajectory stroke fitted by an elliptic arc is defined by a set of parameters. The table1 contains a complete description of the model parameters. Each joint trajectory stroke is modelled by a feature vector V combining geometric and dynamic characteristics:

$$V: \{\text{ellipse } \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \theta_{xy}, \theta_{xz}\}, \text{ Normal Vector to plan } P_k \{ \mathbf{a}_{Pk}, \mathbf{b}_{Pk}, \mathbf{c}_{Pk} \}\}$$

3.3. HAR based on LSTM

In order to put our proposal for human action recognition into context, we first review a special kind type of RNN called Long-Short Term Memory neuron (LSTM).

The choice of LSTM is due to the famous problem of simple RNN is the vanishing gradient problem [39]. Actually, LSTMs are improved to obtain long-distance dependencies within sequence data. The contextual semantics of information are retained and stored for the purpose of obtaining long dependencies between data with the LSTM approach. For this purpose, special memory units named cells are exploited to stock information for dependencies in a long-range context in LSTM. Also, in each LSTM unit composed by a data-driven gates; input, forget, and output gates; to check which fragments of information to remember, forget and pass on to the next step. In this way, LSTM gains the capability to make decisions about what to store, and when to permit reads, writes and deletions by via gates that pass or block information through the LSTM unit.

The formulas (excluding the bias terms) are as follows:

$$\begin{aligned} i_t &= \sigma_i (W_{xi} x_t + W_{hi} h_{t-1} + W_{ci} c_{t-1} + b_i) \\ f_t &= \sigma_f (W_{xf} x_t + W_{hf} h_{t-1} + W_{cf} c_{t-1} + b_f) \\ c_t &= f_t c_{t-1} + i_t \sigma_c (W_{xc} x_t + W_{hc} h_{t-1} + b_c) \\ o_t &= \sigma_o (W_{xo} x_t + W_{ho} h_{t-1} + W_{co} c_t + b_o) \\ h_t &= o_t \sigma_h (c_t) \end{aligned}$$

Where i , f , o , and c represent the outputs of input gate, forget gate, output gate, and cell activation vectors, respectively; all of them have the same vector size h_t therefore defining the hidden value (i.e., the memory state of the block). σ_i , σ_f and σ_o are, respectively, the non-linear functions of input, forget, and output gates. W_{xi} , W_{hi} , W_{ci} , W_{xf} , W_{hf} , W_{cf} , W_{xc} , W_{hc} , W_{xo} , W_{ho} and W_{co} are weight matrices of the respective gates, where x and h are the input and the hidden value of LSTM block respectively. b_i , b_f , b_c and b_o are the bias vectors of input gate, forget gate, cell, and output gate, respectively [74].

Fig.7. Proposed LSTM network. The input data are the BEJT feature of each skeletal joint

Traditional RNNs only mines the short-term dynamics and are unable to discover relations among long-terms of inputs and in order to alleviate this drawback, a typical LSTM model is employed to classify the human actions. LSTM is powerful deep model for time series classification for its capability of interpreting the dynamics in time series and its ability to overcomes the RNN's vanishing gradient problem.

As shown in Figure 7, we build a multi-layer LSTM framework. For the first LSTM layer, we use the beta elliptical joint trajectories as an input x_n where n is the number of joint points in the human skeleton and the output h_t of the lower LSTM layer is as the input of the upper LSTM layer. The output of the highest LSTM layer is fed to a softmax layer to transform the output codes to probability values of class labels. The probability that a sequence V belongs to the class C_k is:

$$p(C_k \setminus V) = \frac{e^{o_k}}{\sum_{i=0}^C e^{o_i}}, \quad k = 1, \dots, C \quad (25)$$

$$o = \sum_{i=1}^T W_{ho} h_t + b_o \quad (26)$$

where C denotes the number of classes, T represents the length of the skeleton joint sequence, $o = [o_1, o_2, \dots, o_C]^T$; h_t is the output response of the last LSTM layer and b_o .

4. Experimental results and discussion

To evaluate the performance of the proposed HAR system, we used four data sets: UT-Kinect data set, MSR-Action3D data set, CAD60 data set and CAD120 data set.

Based on skeleton information for action recognition recorded with Microsoft Kinect and after extracting the BEJT features and constructing the multi-layer LSTM network, we will train this network for human activity recognition. Accomplishing training, according to equation (25), calculating the highest probability as action class for test data set. For these experiments, learning rate $\alpha = 0.6$, both batch size and the number of hidden LSTM are set to 100, forget bias is set to 1.0 and dropout rate is set to 0.5.

All descriptions of each data set, the experimental results and comparison of our proposal system with methods identified in the state of the art are given below.

4.1. Results on UT-Kinect data set

The UT-Kinect [75] data set is captured by a single stationary Kinect that holds 200 sequences of 10 classes produced by 10 subjects in varied views. Each activity is recorded twice for each one and each frame contains 20 skeleton joints. To divide the data set, we pursue the half-vs-half protocol where half of the subjects for training and the remaining for testing [51].

Table 2: Comparison of recognition accuracy on the UT-Kinect dataset. Bold figure means the best performance.

Method	Accuracy
Sub-Trajectories + HMM [53]	86.70%
Joint trajectory + KNN [43]	91.50%
*Joint Coordinates	90.91%
*Joint-Joint Distance	87.88%
*Joint-Joint Orientation + LSTM [2]	84.85%
*Joint-Line Distance	95.96%
STIP Trajectory + LSTM [62]	95.0%
Proposed Method (BEJT + LSTM)	96.4%

Table 2 compares the recognition accuracy of the proposed BEJT method with that of the state-of-the-art methods using UT-Kinect data set. And it is observed that our system based

on beta elliptical joint features and two-layer LSTM shows significantly better recognition accuracy than other methods.

Fig.8. Confusion matrix for Ut-kinect data set

The confusion matrix of classification results affirms the performance of our proposed system. Most of the human action classes obtained higher than 90% classification rates with BEJT-LSTM method. Also, some skeletal sequences are misclassified such as sit down class: 16.70% are classed as stand up class. This misclassification can be explained by the use of the same interest joints that can affect the same BEJT feature vector or the movement of Kinect sensor during recording sequences.

4.2. Results on MSR Action 3D data set

The MSR-Action3D data set [76] is generated by a Microsoft Kinect which is widely used in action recognition. This data set consists of 20 actions performed by 10 subjects for two or three times. 557 samples are valid and each sequence frame contains 20 skeleton joints. Following the protocol A, we used the subjects 1, 3, 5, 7, 9 for training and 2, 4, 6, 8, 10 for testing.

Table 3 collects the accuracy obtained by the proposed BEJT+LSTM architecture over the MSR Action 3D data set as well as other works. As can be seen, the accuracy results

Table 3: Comparison of recognition accuracy on the MSR-Action 3D dataset. Bold figure means the best performance.

Method	Recognition Accuracy
3D positional pairwise differences of joints + HMM [22]	82.0%
Joint Location + HMM [45]	89.23%
Body part + SVM [47]	89.48%
STIP Trajectory + LSTM [62]	90.36%
Joint Trajectory + LSTM [5]	91.21%
Multi-fused-feature + HMM [41]	93.3%
BEJT + LSTM	95.20%

obtained by our proposal outperforms the previous works found in the literature. As can be seen, the accuracy results obtained by our proposal outperforms the previous works found in the literature. We obtained 95.20% as accuracy rate of our proposed system. Confusion matrix in figure 9 reported that thirteen classes 100% correctly classed; four classes obtained higher than 90% classification rates and only three classes obtained an accuracy rate less than 85%.

Fig.9. Confusion matrix for MSR Action data set

4.3. Results on CAD 60 data set

The CAD-60 [77] data set is a publicly available data set captured by the Kinect sensor. Three types of data are available: the 3-D locations of the 15 tracked skeleton joints, the RGB and depth map modalities. It consists of 12 human daily life activities captured in five different environments, performed by four subjects.

Table 4: Comparison of recognition accuracy on the CAD60 data set. Bold figure means the best performance.

Method	Accuracy Recognition
Joint Coordinates + Naïve-Bayes-Nearest-Neighbor [3]	71.9%
Key pose + RF [46]	76.6%
Key pose + RF [45]	80.02%
STIP + SVM[62]	87.5%
BEJT + LSTM	90.5%

Table 4 collects the accuracy obtained by the proposed BEJT+LSTM architecture over the CAD60 data set as well as other works. As can be seen, the accuracy results obtained by our proposal outperforms the previous works found in the literature.

Fig.10. Confusion matrix for CAD 60 data set

In addition, in order to compare our algorithm better, figure 10 depict the confusion matrix obtained for the testing of action recognition algorithm based on BEJT features.

Most of the human action classes obtained higher than 90% classification rates such as rinsing mouth, working in computer, cooking, Our system confused between a number of classes like drinking water and brushing teeth. This confusion due to the convergence between entry data: the same movement of right hand.

4.4. Results on CAD 120 data set

The CAD-120[78] data set contains 120 RGB-D video sequences representing daily living activities of four human subjects (are male and two are female) recorded using a Microsoft Kinect camera. Each video is labeled with a single high-level activity name. The data set provides skeleton tracks of people in the scene. The 3-D locations of tracked skeleton joints are available.

When we compared our accuracy results based on BEJT+LSTM architecture with works of the state-of-the-art in table 5, we found that our results are as good as the best one of the considered works.

Table 5: Comparison of recognition accuracy on the CAD120 data set. Bold figure means the best performance.

Method	Accuracy Recognition
Joints Coordinates + HMM [53]	94.4%
Geometric feature+ SVM [42]	94.2%
Joint trajectories + RDF [44]	90.1%
Lop + CNN [5]	81.2%
BEJT + LSTM	95.0%

5. Conclusion

In this paper, we presented a deep learning approach for human action recognition based on 3D skeletal representation. This effective approach achieves high accuracy on the relevant benchmark data sets. Two factors are considered as the key of our performance: BEJT feature that describes both geometric and dynamic characteristics of skeletal sequences with elliptical arcs that decompose a complex skeleton trajectory in strokes. We also, takes the advantages of LSTM for final recognition. This special kind of recurrent neural networks has the capability to model the long short temporal dependencies automatically.

We validated our model on four public benchmark data sets: MSR Action 3D data set, UT Kinect data set, CAD60 data set and CAD120 data set. Experimental results demonstrated that our approach leads to a significant improvement of classification performance. In the future, we will present a more refined level of 3D joint trajectory approximation that define not only an ellipse to support the trajectory but also an ellipsoidal spindle.

Acknowledgments

The research leading to these results has received funding from the Ministry of Higher Education and Scientific Research of Tunisia under the grant agreement number LR11ES48.

References

- [2] Y. Zhang, X. Liu, M.-C. Chang, W. Ge, and T. Chen. Spatiotemporal phrases for activity recognition. In Proc. European Conference on Computer Vision, Firenze, Italy, October 2012.
- [3] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 804–811, 2014.
- [4] P. Wang, W. Li, Z. Gao, J. Zhang, C. Tang, and P. O. Ogunbona. Action recognition from depth maps using deep convolutional neural networks. IEEE Transactions on Human-Machine Systems, 46(4):498–509, Aug 2016.
- [5] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
- [6] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In Thirtieth AAAI Conference on Artificial Intelligence, 2016.
- [7] GARCIA-CEJA, E., AND BRENA, R. Long-term activity recognition from accelerometer data. Procedia Technology 7 (2013), 248–256
- [8] STAGER, M., LUKOWICZ, P., AND TROSTER, G. Implementation and evaluation of a low-power sound-based user activity recognition system. In Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on (2004), vol. 1, IEEE, pp. 138–141
- [9] NIU, W., LONG, J., HAN, D., AND WANG, Y.-F. Human activity detection and recognition for video surveillance. In Multimedia and Expo, 2004. ICME’04. 2004
- [10] H. Boubaker, A. Chaabouni, N. Tagougui, M. Kherallah and A.M. Alimi, “Handwriting and Hand drawing Velocity Modeling by Superposing Beta Impulses and Continuous Training Component,” in the International Journal of Computer Science Issues (IJCSI), pp. 57-63, September 2013.
- [11] Boubaker H, El-Baati A, Kherallah M, Elabed H, Alimi AM. 2010. Online Arabic handwriting modeling system based on the grapheme segmentation. In: 20th International Conference on Pattern Recognition. (ICPR). Istanbul, Turkey: IEEE. p. 2016 –2064
- [12] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4694–4702, 2015.
- [13] H. Zhang, M. Wang, R. Hong, and T.-S. Chua. Play and rewind: Optimizing binary representations of videos by self-supervised temporal hashing. In Proceedings of the 2016 ACM on Multimedia Conference, pages 781–790. ACM, 2016.
- [14] Y. Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In the IEEE Conference on Computer Vision and Pattern

Recognition (CVPR), June 2015.

[15] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[16] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. NTU RGB+D: A large scale dataset for 3D human activity analysis. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[17] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Spatio-temporal LSTM with trust gates for 3D human action recognition. In European Conference on Computer Vision, pages 816–833. Springer, 2016.

[18] M. Sundermeyer, R. Schlüter, and H. Ney, “Lstm neural networks for language modeling,” in INTERSPEECH, 2012.

[19] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, “A hierarchical deep temporal model for group activity recognition,” in CVPR, 2016.

[20] S. Yeung, O. Russakovsky, G. Mori, and L. Fei-Fei, “End-to-end learning of action detection from frame glimpses in videos,” in CVPR, 2016.

[21] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in CVPR, 2015.

[22] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, “Modeling spatialtemporal clues in a hybrid deep learning framework for video classification,” in ACM MM, 2015.

[23] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” in CVPR, 2015.

[24] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, “Leveraging structural context models and ranking score fusion for human interaction prediction,” arXiv, 2017.

[25] Q. Ke, M. Bennamoun, S. An, F. Bossaid, and F. Sohel, “Spatial, structural and temporal feature learning for human interaction prediction,” arXiv, 2016.

[26] N. Srivastava, E. Mansimov, and R. Salakhutdinov, “Unsupervised learning of video representations using lstms,” in ICML, 2015.

[27] S. Ma, L. Sigal, and S. Sclaroff, “Learning activity progression in lstms for activity detection and early detection,” in CVPR, 2016.

[28] Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural network for skeleton based action recognition,” in CVPR, 2015.

[29] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, “Cooccurrence feature learning for skeleton based action recognition using regularized deep lstm networks,” in AAAI, 2016.

[30] J. Liu, A. Shahroudy, D. Xu, and G. Wang, “Spatio-temporal lstm with trust gates for 3d human action recognition,” in ECCV, 2016.

[31] M. Muller, T. Roder, M. Clausen, B. Eberhardt, B. Kruger, and A. Weber. Documentation of mocap database hdm05. 2007.

[32] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images.

In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1297–1304. IEEE, 2011.

[33] H. Boubaker, A. Chaabouni, N. Tagougui, M. Kherallah and A.M. Alimi, “Hand-writing and Hand drawing Velocity Modeling by Superposing Beta Impulses and Continuous Training Component,” in the International Journal of Computer Science Issues (IJCSI), pp. 57-63, September 2013.

[34] S. Sadanand, J. J. Corso, Action bank: A high-level representation of activity in video, in: Proc. of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, pp. 1234{1241. doi:10.1109/CVPR.2012.6247806.

[35] A. Ciptadi, M. S. Goodwin, J. M. Rehg, Movement pattern histogram for action recognition and retrieval, in: Proc. Of European Conference on Computer Vision (ECCV), Springer, 2014, pp. 695{710. doi:10.1007/978-3-319-10605-2_45.

[36] R. Vemulapalli, F. Arrate, R. Chellappa, Human action recognition by representing 3D skeletons as points in a Lie Group, in: Proc. Of Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 588{595. doi: 10.1109/CVPR.2014.82.

[37] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, “Skeleton-based action recognition using spatio-temporal lstm network with trust gates,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

[38] L. L. Presti and M. La Cascia, “3d skeleton-based human action classification: A survey,” Pattern Recognition, vol. 53, pp. 130–147, 2016.

[39] X. Yang and Y. Tian, “Eigenjoints-based action recognition using Naive- Bayes-Nearest-Neighbor,” in CVPRW, 2012, pp. 14–19.

[40] D. Wu and L. Shao, “Leveraging hierarchical parametric networks for skeletal joints-based action segmentation and recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 724–731.

[41] Jalal, A., Kim, Y. H., Kim, Y. J., Kamal, S., & Kim, D. (2017). Robust human activity recognition from depth video using spatiotemporal multi-fused features. Pattern recognition, 61, 295-308.

[42] Tayyub, J., Tavanai, A., Gatsoulis, Y., Cohn, A. G., & Hogg, D. C. (2014, November). Qualitative and quantitative spatio-temporal relations in daily living activity recognition. In Asian Conference on Computer Vision (pp. 115-130). Springer, Cham.

[43] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo, “3-D human action recognition by shape analysis of motion trajectories on riemannian manifold,” IEEE transactions on cybernetics, vol. 45, no. 7, pp. 1340–1352, 2015.

[44] R. Vemulapalli and R. Chellappa, “Rolling rotations for recognizing human actions from 3D skeletal data,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[45] Slama, R., Wannous, H., Daoudi, M., & Srivastava, A. (2015). Accurate 3D action recognition using learning on the Grassmann manifold. Pattern Recognition, 48(2), 556-567.

[46] E. Cippitelli, S. Gasparrini, E. Gambi, S. Spinsante, A human activity recognition system using skeleton data from RGBD sensors, Computational Intelligence and Neuroscience 2016.

- [47] Evangelidis, G.Singh, G. Horaud, Skeletal quads: Human action recognition using joint quadruples. In 2014 22nd International Conference on Pattern Recognition (pp. 4513-4518). IEEE.
- [48] I. Lillo, A. Soto, and J. Carlos Niebles, “Discriminative hierarchical modeling of spatio-temporally composable human activities,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 812–819
- [49] M. A. Gawayyed, M. Torki, M. E. Hussein, and M. El-Saban, “Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition,” in IJCAI, 2013, pp. 1351–1357.
- [50] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal, “Bioinspired dynamic 3D discriminative skeletal features for human action recognition,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2013, pp. 471–478.
- [51] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo, “3-D human action recognition by shape analysis of motion trajectories on riemannian manifold,” IEEE transactions on cybernetics, vol. 45, no. 7, pp. 1340–1352, 2015.
- [52] P.Wang, W. Li, C. Li, and Y. Hou, “Action Recognition Based on Joint Trajectory Maps with Convolutional Neural Networks,” in IEEE TRANSACTIONS ON CYBERNETICS, VOL. X, NO. X, X 2016.
- [53] Hai, P. T., & Kha, H. H. (2016, July). An efficient star skeleton extraction for human action recognition using hidden Markov models. In 2016 IEEE Sixth International Conference on Communications and Electronics (ICCE) (pp. 351-356). IEEE.
- [54] M. Muller, T. Roder, M. Clausen, B. Eberhardt, B. Kruger, and A. Weber. Documentation “mocap database hdm05. 2007.
- [55] Z. Zhang. “Microsoft kinect sensor and its effect. IEEE MultiMedia, 2012.
- [56] Z. Cai, J. Han, Li Liu, L. Shao, “RGB-D datasets using microsoft kinect or similar sensors: a survey, 2017
- [57] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, Real-Time Human Pose Recognition in Parts from a Single Depth Image, in CVPR, IEEE, June 2011
- [58] Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural network for skeleton based action recognition,” in CVPR, 2015.
- [59] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997. 1, 2, 3, 5
- [60] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Action classification in soccer videos with long short-term memory recurrent neural networks. In Artificial Neural Networks–ICANN 2010, pages 154–159. Springer, 2010. 1, 2, 3
- [61] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In Human Behavior Understanding, pages 29– 39. Springer, 2011. 2, 3, 6, 7
- [62] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In Thirtieth AAAI Conference on Artificial Intelligence, 2016.

- [63] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. NTU RGB+D: A large scale dataset for 3D human activity analysis. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.
- [64] W. Zhu, C. Lan, J. Xing, Y. Li, L. Shen, W. Zeng, X. Xie, Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM network, in: AAAI Conference on Artificial Intelligence, to appear, 2016.
- [65] J. Liu, A. Shahroudy, D. Xu, and G. Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In ECCV, pages 816–833, 2016
- [66] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, G. Mori, “Hierarchical Deep Temporal Models for Group Activity Recognition “. In IEEE transactions on Pattern Analysis and Machines Intelligence, 2016
- [67] B. Mahasseni and S. Todorovic, “Regularizing long short term memory with 3d human-skeleton sequences for action recognition,” in CVPR, 2016.
- [68] J. Liu, A. Shahroudy, D. Xu, and G. Wang, “Spatio-temporal lstm with trust gates for 3d human action recognition,” in ECCV, 2016.
- [69] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, “Skeleton-based action recognition using spatio-temporal lstm network with trust gates,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.
- [70] V. M. Zatsiorsky. Kinematics of Human Motion. Human Kinetics Publishers, 1997.
- [71] Viviani P, Schneider R. 1991. A developmental study of the relationship between geometry and kinematics in drawing move
- [72] Bezine H, Alimi AM, Sherkat N. 2007. Generation and analysis of handwriting script with the beta-elliptic model. Int J Simul. 8(2):45 – 65.
- [73] Boubaker H, Kherallah M, Alimi AM. 2013. Optimization of the beta – elliptic model features estimation. In: 16th International Conference of Graphonomix Society (IGS’13); 2013 June. Nara, Osaka, Japan: International Graphonomix Society. p. 151– 154
- [74] Ordóñez, F., Roggen, D. : Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. Sensors 16(1), 115 (2016)
- [75] Xia, L., Chen, C.C., Aggarwal, J.: View invariant human action recognition using histograms of 3D joints. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, IEEE (2012) 2027
- [76] Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 1290-1297
- [77] Sung J, Ponce C, Selman B, Saxena A (2012) Unstructured human activity detection from rgbd images. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. pp. 842 849.
- [78] Sung, Jaeyong, et al. "Unstructured human activity detection from rgbd images." 2012 IEEE international conference on robotics and automation. IEEE, 2012.