Autonomous gesture recognition using multi-layer LSTM networks and laban movement analysis

Zahra Ramezanpanah, Malik Mallem, Frédéric Davesne

Autonomous Gesture Recognition using Multi-layer LSTM Networks and Laban Movement Analysis

March 30, 2023

Zahra Ramezanpanah zahra.ramezanpanah@myteam.ai
Malik Mallem malik.mallem@univ-evry.fr
Frédéric Davesne frederic.davesne@univ-evry.fr

University of Paris-Saclay, University of Évry, IBISC, 91000, Évry-Courcouronnes, France.

Abstract

In recent years, due to the reasonable price of RGB-D devices, the use of skeletal-based data in the field of human-computer interaction has attracted a lot of attention. Being free from problems such as complex backgrounds as well as changes in light is another reason for the popularity of this type of data. In the existing methods, the use of joint and bone information has had significant results in improving the recognition of human movements and even emotions. However, how to combine these two types of information in the best possible way to define the relationship between joints and bones is a problem that has not yet been solved. In this article, we used the Laban Movement Analysis (LMA) to build a robust descriptor and present a precise description of the connection of the different parts of the body to itself and its surrounding environment while performing a gesture. To do this, in addition to the distances between the hip center and other joints of the body and the changes of the quaternion angles in time, we define the triangles formed by the different parts of the body and calculate their area. We also calculate the area of the single conforming 3-D boundary around all the joints of the body. We use a long short-term memory (LSTM) network to evaluate this descriptor. The proposed algorithm is implemented on five public datasets: NTU RGB+D 120, SYSU 3D HOI, FLORENCE 3D ACTIONS, MSR Action3D and UTKinect-Action3D datasets, and the results are compared with those available in the literature.

1 Introduction

Bilateral interaction between humans and computers has been one of the most interesting and popular topics in the field of computer vision and artificial intelligence. It is used in many areas such as education, health, the medical industry, video surveillance, and the gaming industry [3], [30], [8]. The display of human movements using three-dimensional skeletons in which the human body is represented by joints has received much attention in recent years. As a superior feature, the skeletons have the advantage of being robust against a clustered background, distractions, and changing views and lights. Gesture detection methods, based on a sequence of skeletal data, have high diagnostic accuracy and less computational complexity. That is why these methods are popular among researchers. For example, in [25], according to the skeletal sequence data, the authors first extracted bone information according to the 2D or 3D joint coordinates. The joints and bones (spatial information) in each frame were then displayed as vertices and edges in a circular directional graph, which is fed into the directed graph neural network (DGNN) to extract features for gesture detection. They implemented their proposed algorithm on two large public datasets, NTURGB+D [19] and Skeleton-Kinetics [9], and achieved significant results. In [26], the authors considered three main factors that contribute to the complexity of the pattern in a movement, including spatial dependence between joints, temporal body dependence, and changes in performance such as velocity and accelerations. Their proposed solution offers

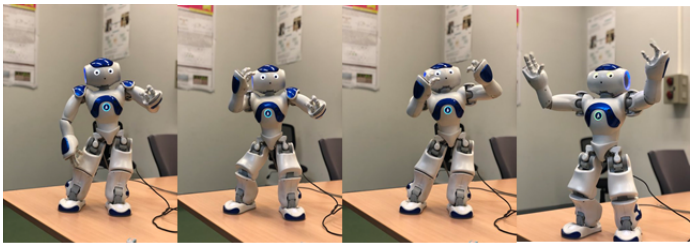
a combination of graph diagrams and Long-Short term memory (LSTM) for space-time dynamics modeling. The whole problem then extends to be a probabilistic model following the Bayesian framework with a novel adversarial prior. To improve robustness and increase detection accuracy, a Bayesian inference problem has been designed for the classification phase. Since generally, the movements of the body might have a different attribute due to the orientation misalignment, the authors, in [14], transformed the original raw data which is the 3D coordinate of the joints into a human cognitive coordinate system. Afterward, by the use of the multi-term temporal sliding LSTM networks, they fed a different type of dependencies including spatial and temporal data into the network and at the end by implementing their proposed method on MSR Action3D, UTKinect-Action [11], NTURGB+D [24], Northwestern-UCLA [32], and UWA3DII [22] datasets, they evaluate their proposed algorithm. Laban Movement Analysis (LMA) [13], is another method that uses temporal and spatial information to formally describe human movement. This algorithm observes the gestures based on four aspects of the movement, namely body, effort, space, and shape. This method, which can accurately describe the movements due to considering all aspects of a gesture, has been considered by many researchers. This method, which can accurately describe the movements due to considering all aspects of a gesture, has been considered by many researchers. For example, in [34], this method is used to construct a descriptor for dance analysis. They used spatial orientation, limb structure, and force effect to calculate the components of LMA. In the training and classification phase, they proposed the CNN-LSTM hybrid deep learning model, which is a combination of two network structures of LSTM and CNN and verifies the effectiveness of the method through contrast experiments. Their results show that the CNN-LSTM model has the highest accuracy rate. In [1], the authors used LMA to classify human emotions based on body movements. They used Random Decision Forest to classify the emotions. In [23], the authors calculated the curve of Dynamic Time Warping of the descriptor they constructed by the use of LMA. They trained and classified their data according to the $argmax(WX + b)$ by the use of a Support Vector Machine. Their proposed method outperforms many results in the literature.

In this paper, we used spatio-temporal characteristics of the skeletal body to calculate LMA components and construct a robust descriptor. Several contributions are made to the descriptor to improve gesture recognition:

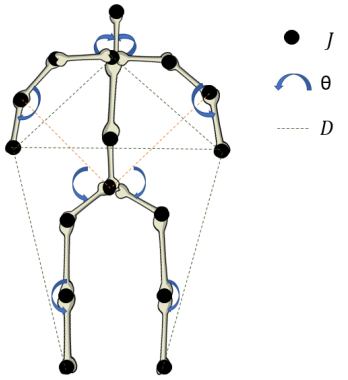
- On the spatial aspect, the areas of the triangles encompassing several parts of the body as well as the sphere encompassing the body are used as discriminating characteristics.
- On the temporal aspect, the variations of the quaternions representing the rotation vectors of the joints of the body are also used as discriminating characteristics.
- A broad evaluation of our descriptor and the proposed algorithm on five public datasets is applied: NTURGB+D 120, SYSU HOI, FLORENCE 3D ACTIONS, MSR Action3D and UTKinect-Action3D.

2 Proposed Feature Descriptor

The construction of a precise and robust descriptor is one of the most critical steps in the process of pattern recognition. When dealing with the three-dimensional data of the joints and the information obtained from the skeleton of the working body, we must pay attention to the fact that this data can change according to different conditions. For example, the three-dimensional coordinates of the joints vary with the changes in the initial position and rotation of the person performing the gestures, according to the global frame. Therefore, without the normalization of data, the 3D coordinates of the joints can not be reliable data for recognizing a gesture. To this end, we have selected factors that are immutable to these factors. Factors that are variable with these factors were also normalized. In the following of this section, we will first define LMA, and then we will detail the construction of a robust descriptor by the use of this algorithm.
Laban Movement Analysis (LMA) is an algorithm to analyze, visualize and describe human gestures and emotions using its four components, namely body, shape, space, and effort which are composed of Spatio-temporal features.
Since in this article we are dealing with gestures without emotions, then in this study, we investigate the first three components that mean Body, Shape, and Space. The **body** component describes the structural and physical characteristics of the human body during movement. In this paper, to express the connectivity



(a) The various poses of NAO while dancing.



(b) Representation of the selected characters for the body component.

Figure 1: NAO

of the body and find the relationship between the parts of the body, several characters are defined for this component. To do this, using Choregraphe we prepared the blue NAO robot to perform several gestures. The purpose is to find out what components a humanoid robot uses more to perform various gestures such as dancing (Fig 1a).

The first character defined for the body element is the 3D coordinates of all the joints. Let consider $j_i = (x_i, y_i, z_i)$ is the i_th joint of the skeletal representation of the body, so for all joints we will have $J_n = \{j_1, ..., j_i, ..., j_n\}$, where n is the number of joints and can vary depending on the type of used sensor. The second subcomponents of the body are the angles of body parts to the descriptor. So the angles around elbows (θ_r^1, θ_l^1), neck (θ_r^2, θ_l^2), hips (θ_r^3, θ_l^3) and knees (θ_r^4, θ_l^4) are computed by:

$$\theta^{j_j} = \arccos \frac{\overrightarrow{j_j - j_i} \cdot \overrightarrow{j_k - j_j}}{\left\|\overrightarrow{j_j - j_i}\right\| \left\|\overrightarrow{j_k - j_j}\right\|}. \tag{1}$$

Where $j_i = (x_i, y_i, z_i)$, $j_j = (x_j, y_j, z_j)$, and $j_k = (x_k, y_k, z_k)$ are the $3D$ coordinates of three consecutive joints. We also calculated the distances between the joints that move the most during a gesture according to the following formula.

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}. \tag{2}$$

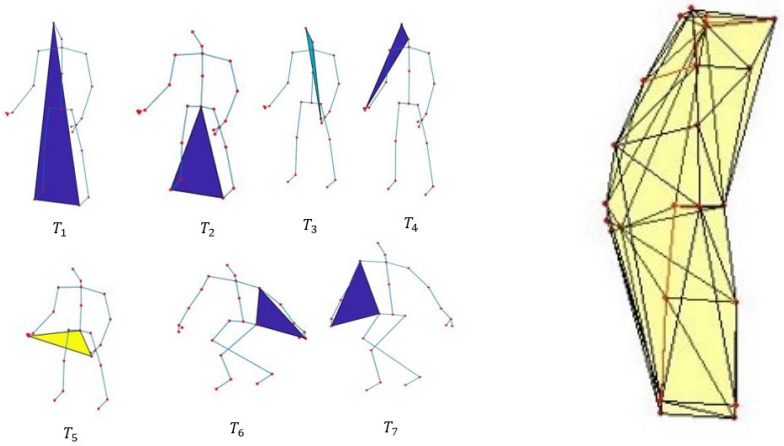
Where i and j belong to the index of the joints of the head, left hand, right hand, left foot and right foot. Figure 1b shows all the selected elements for the body component.
The **Space** component describes the location, directions and paths of a movement. This component answers the question of where the body is going and in which space it fits. In this part, we chose the changes of Yaw, Roll and pitch of the body as an element of the descriptor. With this choice, we can find out which axis the body rotates during a gesture.

The **Shape** component consists of three distinct qualities to describe the changes in the form of the body: shape flow, directional movement, and shaping. Shape flow reflects the relationship of the body with itself. The changes can be seen as the increasing or decreasing volume of the shape of the body. At this point, we've selected a few geometric factors as follows:

- The area of $Tr = \{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$, in which Tr is the set of triangles formed by the three joints that have the most changes in different parts of the body (Fig 2a):
  - Triangle consisting of head, right foot and left foot, T_1.
  - Triangle consisting of hip center, right foot and left foot, T_2.

– Triangle consisting of head, neck and right hand, T_3.
– Triangle consisting of head, neck and left hand, T_4.
– Triangle consisting of right hand, left hand and hip center T_5.
– Triangle consisting of left hand, left shoulder and left hip T_6.
– Triangle consisting of right hand, right shoulder and right hip T_7.

The shaping subcomponent characterizes the change in the shape of the body in relation to its space and defines the connection between where the body interacts continuously and three-dimensionally and the volume of the surrounding environment. For this subcomponent, we calculate the volume of B, in which B is the 3D boundary around all the joints of the skeleton data. In order to obtain the boundary that encapsulates all the joints of the skeleton body, we used the algorithm suggested in [31] (Fig 2b).



(a) The defined triangles in the descriptor.

(b) The 3D boundary around all the joints of the skeleton body.

Figure 2: Saptio features.

The last subcomponent of shape is called Directional Movement, which predicts where the body is directed toward some part of its environment. It is divided further into Straight-like and Arc-like trajectories. To do this end, we calculated the curvature measurement of the left and right hand according to the following formula:

$$C(j_h) = \frac{\|v(j_h) \times a(j_h)\|}{(\sqrt{v(j_h)_x^2 + v(j_h)_y^2 + v(j_h)_z^2})}. \tag{3}$$

Where, the h index indicates the left and right hand, the V indicates the speed, and the a indicates the acceleration. If C tends to zero, it means that the path taken by the hands is a straight path, and a large curvature indicates a curvy trajectory.

3 Classification Via Deep Learning

The following is a brief overview of how LSTM works and then the network used in this article is detailed.

3.1 Long Short Term Memory (LSTM)

LSTM is an advanced model of recurrent neural networks (RNN). In general, the function of RNN is to use short-term memory, meaning that they continuously use the previous results for use in the current neural network layers. In principle, previous information is used in the present work. This means that we do not have a list of all the previous information available for the neural network. LSTM presents long term

memory to recurrent neural networks. This feature of LSTM reduces the problem of vanishing gradient [6], a problem that stops RNN because updates related to different weights in a given neural network gradually shrink. This is done in LSTM using a series of gates. Figure 3 shows the general structure of RNN and LSTM, which we will explain in the following.

As can be seen, in RNN we have a simple structure consisting of only one layer, $tanh$, which has the respon-

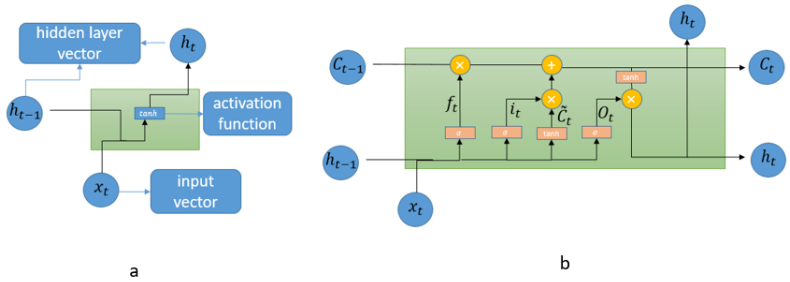


Figure 3: The General Structure For a a) Recurrent Neural Network and b) Long Short Term Memory.

sibility of squashing values which means it converts them between -1 and 1. But in LSTM the structure of each block is a little more complicated. The key in LSTM is the cell state, which is the path between C_{t-1} and C_T in figure 3. The cell state acts like a conveyor belt and runs straight down the entire chain. That means the network can remove or add information and update itself by passing to the cell state (algorithm 1).

Algorithm 1 Cell State Algorithm in LSTM.

```
IF The value of input gate > 0 Then
    Open gate: Transmission of external information inside the block
        IF The value of forget gate > 0 Then
            The previous state of the cell state, C_{t-1} impacts the calculation of the present state.
                IF The value of output gate > 0 Then
                    Open gate: Transmission of internal information to the outside of the unit
                ELSE
                        gate closed: No information provided on exit
                END
        ELSE
        The cell forgets its previous value
        END
ELSE
    gate closed: Information blocked
END
```

Therefore, as can be seen in Algorithm 1, the update of a cell state is done through three gates called

Input and Output and Forget gates. The gates are a way to optionally let information through and they are composed out of a sigmoid neural layer and a point-wise multiplication operation.

3.2 Learning the Classifier

Since there are tags for each video, we feed them as neural network output at each step. To train the data, an LSTM network is terminated using a soft-max layer. This layer is responsible for predicting the action class $\hat{y}$ among the given class sets, y. To this end, it calculated the probability of an input x belonging to class C_k:

$$p(C_k|x) = \frac{exp(z_k)}{\Sigma_{i=1}^{C} exp(z_i)}. \tag{4}$$

Where

$$z = W_z h_n^L + b_z. \tag{5}$$

And h_n^L denotes the hidden output layer at the last time step n which is placed on the highest layer of LSTM, L.

3.3 Used Network

Using the LSTM multilayer network (stacked/deep LSTM), we construct our pipeline. The first LSTM layer takes LMA features as the input $x_t = (f_1 f_2 \dots f_T)$, in which T is the total number of frames (i.e $T = 3.n + 8 + 5 + 3(n-1) + 7 + 8 + 2$). The last LSTM layer takes the output ht from the lower LSTM layer as the input x_t. This type of structure gives the upper LSTM layers the ability to achieve longer-term dependencies on the input sequence.

4 Experiments

The proposed pipeline is evaluated and analyzed on five public datasets. These datasets are NTU RGB+D 120 [19], UT-Kinect [35], SYSU-3D [7], MSR Action3D [16], and Florence 3D [2].

4.1 Implementation Details

In this experiment, except the three-dimensional coordinates of the joints, the other factors used to construct the descriptor are invariant with respect to the factors such as initial position, orientation and size of the participants. Thus, the three-dimensional coordinates of the joints can be as good an element in describing a movement as they can be in reducing the accuracy of recognition. In order to solve this problem, we normalized this data. Data normalization is done in the following three steps:

- Resizing: all distances between two consecutive joints are divided by its magnitude.
- Translation: displacement of the skeleton body in the center of the Kinect.
- Rotation: align the coordinate of the skeleton boy with the coordinates of the Kinect.

In the training phase, the network splits the data into small batches and pads the sequence of frames so that they are all the same length. If the number of these patches is high, the network performance will decrease. In order to prevent the network from having a large number of patches, by the use of Spline function we fixed the number of frames in each dataset. Figure 4 shows the hand changes in y-axis when performing the "Waving" gesture in two modes: the number of main frames, and the number of fixed frames. As can be seen, the shape of the path traveled by hand did not change after fixing the number of frames, which means that fixing the frames did not change the nature of the data. The implementation of our approach is based on the MATLAB 2018. We implemented our proposed descriptor before the first LSTM layer of the designed network. By the use of "Holdout" validation with a fraction of 0.3 for SYSU HOI, MSRACTION, Florence3D, UTKinect and 0.4 for NTU 120, the data are divided into two categories, training and validation. We set the number of hidden layer to 50 for SYSU HOI, MSRACTION, Florence3D, UTKinect and 100 for

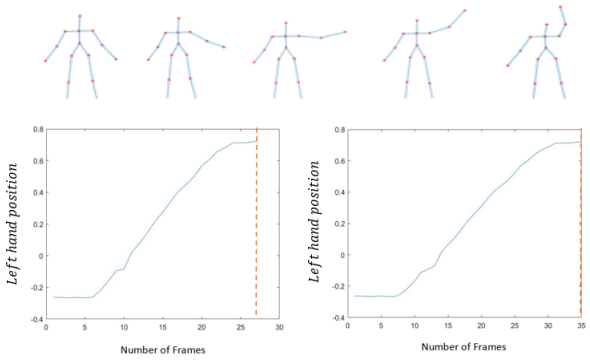


Figure 4: Display of y changes of the left hand in "Waving" gesture a) in 27 frames (number of main frames) and 35 frames (number of selected frames).

NTU 120. The learning rate is set to 0.01. The hardware used in this work is a single GPU. Our proposed framework achieves promising results on all the public datasets, which shows the robustness of our process.

4.2 Results

In this section, we will compare the results obtained from the proposed pipeline with the results in the literature. We evaluate these five datasets by setting up a Holdout training and testing validation setup. Two types of datasets are used in this article. The first type are small datasets, which are MSR Action, UTKinect, Florence3D and SYSU HOI and a very large dataset called NTU RGB+D 120. Depending on the size of the data -set, the number of epochs used in the network varies. Figure 5 demonstrates the diagram of the relationship between the epoch and the accuracy obtained in the three UTKinect, Florence 3D and SYSU HOI datasets. According to the trial and error method, the best results were obtained for MSRAction in epoch 120, for SYSU HOI in epoch 110, For Florence and UTKinect in Epoch 65 and for NTU RGB+D 120 in epoch 170. The results obtained from the implementation of the pipeline proposed above after setting

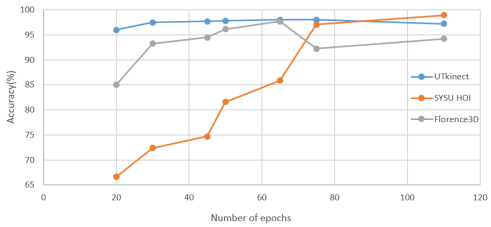


Figure 5: Display the obtained accuracy in different epochs.

the necessary parameters in the network are 98.9% for SYSU HOI, 97.7% for Florence3D, 98% for UTKinect, 92.2% for MSRAction and 72.6% for NTU RGB+D 120. In [23], the features of the first four datasets, which are small datasets, same as this paper, have been extracted using the Laban Movement Analysis, but for their training and classification, the classical machine learning method, Support Vector Machine, has been used. By comparing the results obtained in [23] with the results obtained in this work, we can conclude that deep learning methods are more accurate even for small datasets. Figure 6 show the confusion matrices of SYSU HOI, Florence3D and UTKinect.

Tables 1, 2 and 3 compare the results obtained in this article with other results in the literature.

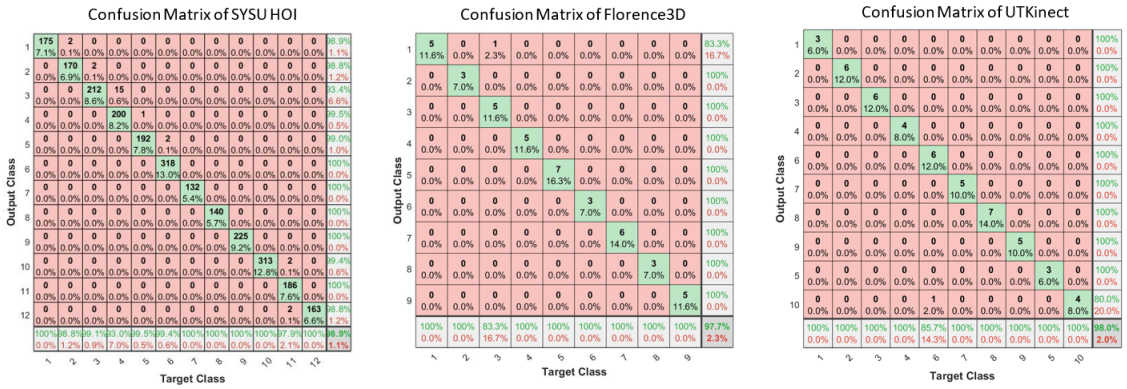


Figure 6: Confusion matrices of SYSU HOI, Florence3D and UTKinect.

Method	SYSU HOI	NTU 120
Cooperative Training of Deep Aggregation Networks [33]	98.33	-
Dynamic Time Warping [23]	92.32	-
Convolutional Neural Networks with Joint Supervision [18]	97.08	-
GVFE + ST-GCN w/ DH-TCN[21]	-	**74.2**
Body Pose Evolution Map[20]	-	66.9
Multi-Task Learning Network[10]	-	57.9
Our Method	**98.94**	72.6

Table 1: Comparison with the state-of-the-art results SYSU HOI and NTU 120

Method	Florence 3D Acc(%)
Cooperative Warp [27]	88.38
Intrinsic SCDL (Bi-LSTM) [28]	93.04
Transition-Forest [5]	94.16
SCK+DCK [12]	95.23
Our method	97.7

Table 2: Comparison with the state-of-the-art results Florence 3D.

Method	MSRACtion	UTKinect Action
Active Joint [29]	84.72	95.9
Mining Key Skeleton Poses with Latent SVM[17]	90.94	91.5
Motion Trajectories [4]	92.1	91.50
Cooperative Warp [27]	90.90	95.38
Group Sparse Regression[15]	-	95.1
Our method	92.2	98

Table 3: Comparison with the state-of-the-art results UTKinect and MSRAction.

5 Conclusions and perspectives

In this paper, we focused on recognizing human movements using LMA. A multi-layer LSTM is used to train and classify data. In general, data is used as input for deep networks without processing. Therefore, problems

such as automatic padding as well as non-normalized data may reduce system performance and accuracy. In this paper, we solved these problems as two strategies. We tried to use elements to construct descriptors that were invariant with respect to the factors such as initial position and rotation of the participants. Only one element in this descriptor is variable in relation to these factors, which was solved by normalizing it. The problem of automatic data padding during network training was also solved by equalizing the number of frames in each dataset. By implementing the above methods and also selecting the appropriate elements to build the descriptor, the pipeline proposed in this paper was evaluated on five public datasets. The results obtained in this article were in many cases better than the results in the literature. In future work, we intend to use deep learning methods to identify human emotions through a combination of factors such as body movements, face, voice and large datasets.

References

[1] Ajili, Insaf et al. "Expressive motions recognition and analysis with learning and statistical methods". In: *Multimedia Tools and Applications* 78.12 (2019), pp. 16575–16600.

[2] Bagdanov, Andrew D, Del Bimbo, Alberto, and Masi, Iacopo. "The florence 2d/3d hybrid face dataset". In: *Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding*. 2011, pp. 79–80.

[3] Chopin, Adrien, Bediou, Benoit, and Bavelier, Daphne. "Altering perception: the case of action video gaming". In: *Current Opinion in Psychology* 29 (2019), pp. 168–173.

[4] Devanne, Maxime et al. "3-d human action recognition by shape analysis of motion trajectories on riemannian manifold". In: *IEEE transactions on cybernetics* 45.7 (2014), pp. 1340–1352.

[5] Garcia-Hernando, Guillermo and Kim, Tae-Kyun. "Transition forests: Learning discriminative temporal transitions for action recognition and detection". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 432–440.

[6] Hochreiter, Sepp. "The vanishing gradient problem during learning recurrent neural nets and problem solutions". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), pp. 107–116.

[7] Hu, Jian-Fang et al. "Jointly learning heterogeneous features for RGB-D activity recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 5344–5352.

[8] Islam, Naveed et al. "A blockchain-based fog computing framework for activity recognition as an application to e-Healthcare services". In: *Future Generation Computer Systems* 100 (2019), pp. 569–578.

[9] Kay, Will et al. "The kinetics human action video dataset". In: *arXiv preprint arXiv:1705.06950* (2017).

[10] Ke, Qiuhong et al. "A new representation of skeleton sequences for 3d action recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 3288–3297.

[11] Kisku, DR, Tistarelli, M, and Sing, JK. "Computer Vision and Pattern Recognition Workshops". In: *Miami, Florida, USA* (2009), p. 60.

[12] Koniusz, Piotr, Cherian, Anoop, and Porikli, Fatih. "Tensor representations via kernel linearization for action recognition from 3d skeletons". In: *European Conference on Computer Vision*. Springer. 2016, pp. 37–53.

[13] Laban, Rudolf and Ullmann, Lisa. "The mastery of movement." In: (1971).

[14] Lee, Inwoong et al. "Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 1012–1020.

[15] Li, Meng, Yan, Liang, and Wang, Qianying. "Group sparse regression-based learning model for real-time depth-based human action prediction". In: *Mathematical Problems in Engineering* 2018 (2018).

[16] Li, Wanqing, Zhang, Zhengyou, and Liu, Zicheng. "Action recognition based on a bag of 3d points". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE. 2010, pp. 9–14.

[17] Li, Xiaoqiang, Zhang, Yi, and Liao, Dong. "Mining key skeleton poses with latent svm for action recognition". In: *Applied Computational Intelligence and Soft Computing* 2017 (2017).

[18] Li, Yupeng et al. "Action Recognition using Convolutional Neural Networks with Joint Supervision". In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE. 2019, pp. 2015–2020.

[19] Liu, Jun et al. "Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding". In: *IEEE transactions on pattern analysis and machine intelligence* (2019).

[20] Liu, Mengyuan and Yuan, Junsong. "Recognizing human actions as the evolution of pose estimation maps". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1159–1168.

[21] Papadopoulos, Konstantinos et al. "Vertex feature encoding and hierarchical temporal modeling in a spatial-temporal graph convolutional network for action recognition". In: *arXiv preprint arXiv:1912.09745* (2019).

[22] Rahmani, Hossein et al. "Histogram of oriented principal components for cross-view action recognition". In: *IEEE transactions on pattern analysis and machine intelligence* 38.12 (2016), pp. 2430–2443.

[23] Ramezanpanah, Zahra, Mallem, Malik, and Davesne, Frédéric. "Human Action Recognition Using Laban Movement Analysis and Dynamic Time Warping". In: *24th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2020)*. 2020.

[24] Shahroudy, Amir et al. "Ntu rgb+ d: A large scale dataset for 3d human activity analysis". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1010–1019.

[25] Shi, Lei et al. "Skeleton-based action recognition with directed graph neural networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7912–7921.

[26] Si, Chenyang et al. "An attention enhanced graph convolutional lstm network for skeleton-based action recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2019, pp. 1227–1236.

[27] Sun, Zheng et al. "Cooperative Warp of Two Discriminative Features for Skeleton Based Action Recognition". In: *Journal of Physics: Conference Series*. Vol. 1187. 4. IOP Publishing. 2019, p. 042027.

[28] Tanfous, Amor Ben, Drira, Hassen, and Amor, Boulbaba Ben. "Sparse Coding of Shape Trajectories for Facial Expression and Action Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[29] Tehrani, Ahmad KN, Aghbolaghi, Maryam Asadi, and Kasaei, Shohreh. "Skeleton-based Human Action Recognition". In: (2017).

[30] Ullah, Amin et al. "Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments". In: *Future Generation Computer Systems* 96 (2019), pp. 386–397.

[31] Veltkamp, Remco C. "Boundaries through scattered points of unknown density". In: *Graphical Models and Image Processing* 57.6 (1995), pp. 441–452.

[32] Wang, Jiang et al. "Cross-view action modeling, learning and recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 2649–2656.

[33] Wang, Pichao et al. "Cooperative training of deep aggregation networks for RGB-D action recognition". In: *arXiv preprint arXiv:1801.01080* (2017).

[34] Wang, Simin et al. "Dance Emotion Recognition Based on Laban Motion Analysis Using Convolutional Neural Network and Long Short-Term Memory". In: *IEEE Access* 8 (2020), pp. 124928–124938.

[35] Xia, Lu, Chen, Chia-Chih, and Aggarwal, Jake K. "View invariant human action recognition using histograms of 3d joints". In: *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE. 2012, pp. 20–27.