Received: 17 March 2017 | Revised: 19 March 2017 | Accepted: 20 March 2017

DOI: 10.1002/cav.1782

SPECIAL ISSUE PAPER

WILEY

3D skeleton-based action recognition by representing motion capture sequences as 2D-RGB images

Sohaib Laraba[1] | Mohammed Brahimi[2,3] | Joëlle Tilmanne[1] | Thierry Dutoit[1]

[1]TCTS Lab, Numediart Institute, University of Mons, Mons, Belgium

[2]Department of Computer Science, USTHB University, Algiers, Algeria

[3]Department of Computer Science, Mohamed El Bachir El Ibrahimi University, Bordj Bou Arreridj, Algeria

Correspondence
Sohaib Laraba, TCTS Lab, Numediart Institute, University of Mons, Mons, Belgium.
Email: sohaib.laraba@umons.ac.be

Funding information
Numediart and Infortech of the University of Mons (UMONS); EU; Wallonia, Grant/Award Number: n 600676

Abstract

In recent years, 3D skeleton-based action recognition has become a popular technique of action classification, thanks to development and availability of cheaper depth sensors. State-of-the-art methods generally represent motion sequences as high dimensional trajectories followed by a time-warping technique. These trajectories are used to train a classification model to predict the classes of new sequences. Despite the success of these techniques in some fields, particularly when the data used are captured by a high-precision motion capture system, action classification is still less successful than the field of image classification, especially with the advance of deep learning. In this paper, we present a new representation of motion sequences (Seq2Im—for sequence to image), which projects motion sequences onto the RGB domain. The 3D coordinates of joints are mapped to red, green, and blue values, and therefore, action classification becomes an image classification problem and algorithms for this field can be applied. This representation was tested with basic image classification algorithms (namely, support vector machine, *k*-nearest neighbor, and random forests) in addition to convolutional neural networks. Evaluation of the proposed method on standard 3D human action recognition datasets shows its potential for action recognition and outperforms most of the state-of-the-art results.

KEYWORDS

action recognition, convolutional neural networks, 3D data representation

1 | INTRODUCTION

Engineers have long dreamed of machines that understand human actions. In the early days of artificial intelligence, researchers analyzed human actions from video sequences. Despite significant efforts to tackle this problem,[1] it is still an unresolved challenge. Recently, this field has rapidly advanced due to the development of depth sensors. 3D human activity analysis[2] has then attracted more interest than ever before, as the articulation of a human body skeleton can be estimated in real time with cheap cameras. Late skeletal-based approaches to recognize human activities were quite successful, thanks to their view- and illumination-invariant representations. Typical examples include a pairwise representation of 3D joints in a lie group,[3] a histogram of oriented displacements to describe 2D trajectories from 3D trajectories,[4] and a histogram of 3D joint locations as representation postures.[5]

Despite the interesting results of such methods, compared to action recognition from video sequences, this field still needs to be improved. Recently, deep neural networks and particularly convolutional neural networks (CNNs) have shown their great power in learning patterns from images and videos.[6–10] Unfortunately, CNNs only capture local spatial patterns in data. In this paper, we

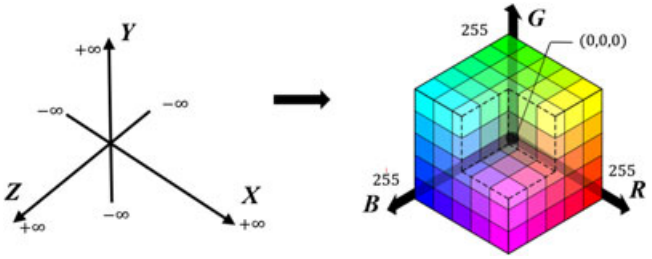


FIGURE 1 Similarity between the *XYZ* and RGB spaces

present a new, yet simple, representation of 3D skeletal sequences for human action recognition. To the best of our knowledge, this representation has not been used before and can advance this field because efficient image classification methods can be applied. To transform a sequence into an image, 3D data are normalized and then mapped into the RGB space. As a result, the high dimensionality of motion capture (mocap) sequences is reduced to 2D color images. In this paper, we use these images to train regular image classifiers, such as support vector machine (SVM), *k*-nearest neighbor (KNN), and random forest (RF), to see the efficiency of this representation. We also try our representation with CNNs. Most of the available datasets are small and not suited for deep learning. However, we try fine-tuning existing models trained on ImageNet[7,8,11] to exploit learned features.

The rest of this paper is organized as follows. A quick overview of the related works will be presented in Section 2. Section 3 presents the details of the proposed method, and the evaluation of this method on three datasets with analysis of the results are presented in Section 4. Section 5 concludes this paper with future works.

2 | RELATED WORKS

Many works have exploited RGB-D (Red, Green, Blue and Depth) data for human action recognition. The review of Aggarwal et al.[2] summarizes the major techniques. In this section, we briefly review works related to our method, including particularly skeleton-based 3D action representations.

Zhang et al.[12] used the Kinect sensor to track a human body skeleton and detect falling events. They used angles between every pair of selected joints in addition to head–floor distance (the distance between the head joint and the floor plane) as features to train a set of SVM-based classifiers. Sempena et al.[13] built a feature vector from joint orientations along time series and applied dynamic time warping (DTW) to recognize some daily human actions. Joint orientation is a good feature because it is body invariant. However, it is less useful in the case of noisy data like the one provided by depth sensors. Bloom et al.[14] extracted a set of pose-based features from 3D joint positions, such as position difference between different joints, velocity, velocity magnitude, angular velocity, and joint angles. These features are used to recognize human gaming actions. Laraba et al.[15] followed a similar representation. They extracted a set of geometric features from different 3D joint positions, such as the distance between the right ankle and the plane defined by the pelvis, left hip and ankle joints, and so forth. These features are used to train hidden Markov models (HMMs) for recognition of traditional dance steps. These last works focused on 3D joint trajectories to recognize human actions. Vemulapalli et al.[3] proposed a different representation that explicitly models the 3D geometric relationships between different body parts. A 3D skeleton was represented as a point in a lie group. A human action was represented then as a curve in a lie group. The classification was then performed using a combination of DTW, SVM, and Fourier temporal pyramid representation.

With the late advances in the field of deep learning, some researchers attempted to apply these methods to recognize human actions. Huang et al.[16] incorporated the lie group structure into a deep network architecture to learn more lie group features for 3D action recognition. Bao[17] proposed an action recognition framework based on conceptors of skeleton joint trajectories. Conceptors are neurodynamical organizations based on recurrent neural networks (RNNs) proposed by Jaeger.[18] Softmax regression is then used to recognize trajectory codes.

3 | PROPOSED METHOD

The RGB space is an alternative domain to explore data by mapping *XYZ* coordinates into RGB components (Figure 1). Figure 2 illustrates the different steps to transform a 3D skeleton sequence into an RGB image.

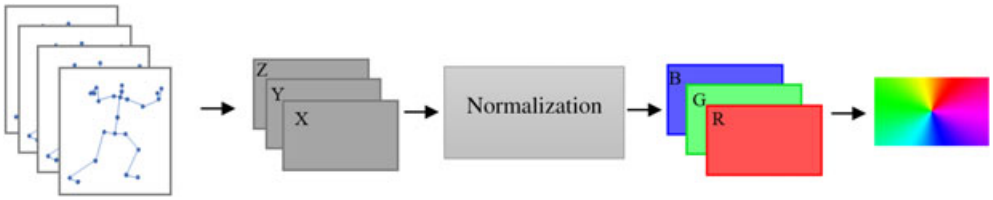


FIGURE 2 Illustration of the proposed RGB representation of a 3D skeleton sequence. From a 3D skeletal sequence, we extract X, Y, and Z matrices, and we normalize them between 0 and 255 to extract red, green, and blue channels of a color image

3.1 | 3D sequence to RGB image transformation

Let $S(V(f))$ be a sequence of body skeletons, where $V(f) = v_1(f), \ldots, v_N(f)$ denotes a set of body joint locations, N is the number of joints, and f is the index of the frame. For each body joint $i = 1, \ldots, N$, $v_i = (x_i, y_i, z_i), \forall (x_i, y_i, z_i) \in \mathbb{R}^3$. The sequence $S(V(f))$ can be represented in a matrix form as follows:

$$S(V(f)) = \begin{pmatrix} x_1(1)y_1(1)z_1(1) & \cdots & x_1(F)y_1(F)z_1(F) \\ \vdots & \ddots & \vdots \\ x_N(1)y_N(1)z_N(1) & \cdots & x_N(F)y_N(F)z_N(F) \end{pmatrix},$$

where F is the number of frames.

In this work, we map the values of $S(V(f))$ onto the RGB domain by normalizing all values between 0 and 255. First, we extract the X, Y, and Z matrices from $S(V(f))$ and process each one separately. $S(V(f)) = (X, Y, Z)$, where

$$X = \begin{pmatrix} x_1(1) & \cdots & x_1(F) \\ \vdots & \ddots & \vdots \\ x_N(1) & \cdots & x_N(F) \end{pmatrix},$$
$$Y = \begin{pmatrix} y_1(1) & \cdots & y_1(F) \\ \vdots & \ddots & \vdots \\ y_N(1) & \cdots & y_N(F) \end{pmatrix},$$
$$Z = \begin{pmatrix} z_1(1) & \cdots & z_1(F) \\ \vdots & \ddots & \vdots \\ z_N(1) & \cdots & z_N(F) \end{pmatrix}.$$

Then, for each $x_i(f), y_i(f), z_i(f), i = 1, \ldots, N, f = 1, \ldots, F$, we compute $r_i(f), g_i(f), b_i(f)$, respectively the red, green, and blue values as follows:

$$\begin{cases} r_i(f) = 255 * \frac{x_i(f) - \min(X)}{\min(X) - \max(X)} \\ g_i(f) = 255 * \frac{y_i(f) - \min(Y)}{\min(Y) - \max(Y)} \\ b_i(f) = 255 * \frac{z_i(f) - \min(Z)}{\min(Z) - \max(Z)}. \end{cases} \quad (1)$$

The minimum and maximum values of each matrix (X, Y, and Z) are $\min(X)$, $\min(Y)$, $\min(Z)$ and $\max(X)$, $\max(Y)$, $\max(Z)$, respectively. We obtain the new matrices as follows:

$$R = \begin{pmatrix} r_1(1) & \cdots & r_1(F) \\ \vdots & \ddots & \vdots \\ r_N(1) & \cdots & r_N(F) \end{pmatrix},$$
$$G = \begin{pmatrix} g_1(1) & \cdots & g_1(F) \\ \vdots & \ddots & \vdots \\ g_N(1) & \cdots & g_N(F) \end{pmatrix},$$
$$B = \begin{pmatrix} b_1(1) & \cdots & b_1(F) \\ \vdots & \ddots & \vdots \\ b_N(1) & \cdots & b_N(F) \end{pmatrix},$$

where $(r_i, g_i, b_i) \in [0, 255]^3$.

From these matrices, we create a single RGB image, where each matrix represents a channel in the final image. Figure 3 shows a transformation of a random mocap sequence of one joint and 20 frames (Figure 3a) into a $1 * 20$ pixel RGB image (Figure 3c).

The length of a mocap sequence can be, in most cases, very large compared to the number of joints (or markers). The resulting image, in this case, is very narrow. For example, a Kinect V2 sequence of 5 s long will result an image of $25 * 150$ pixels

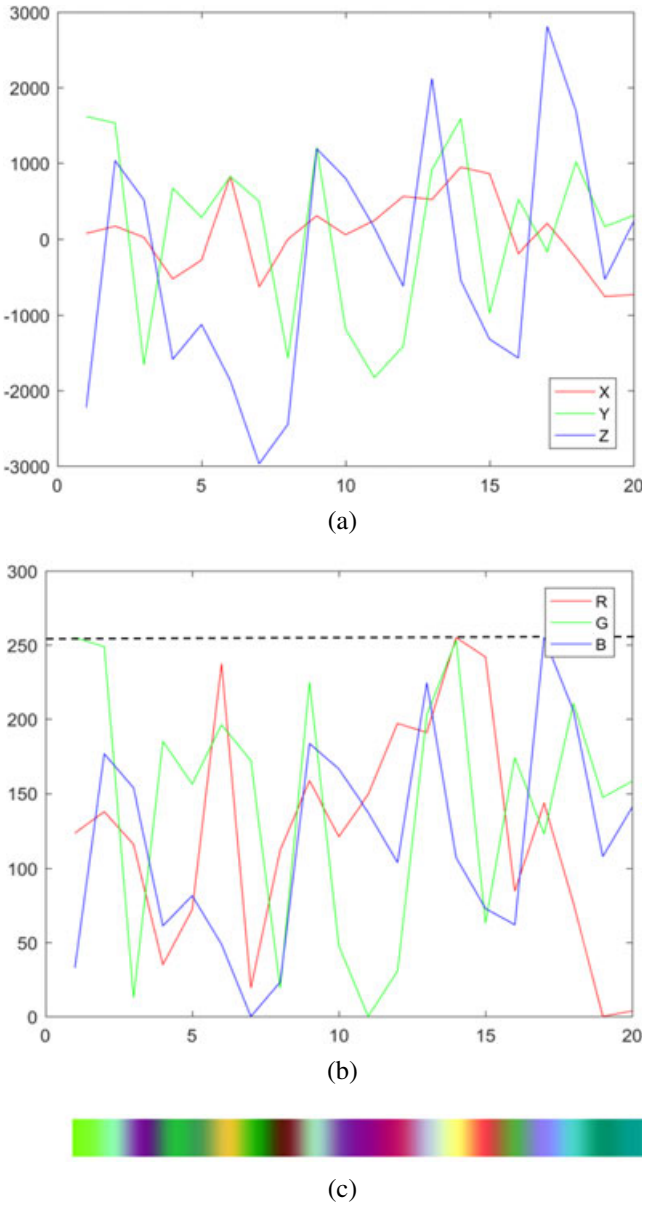


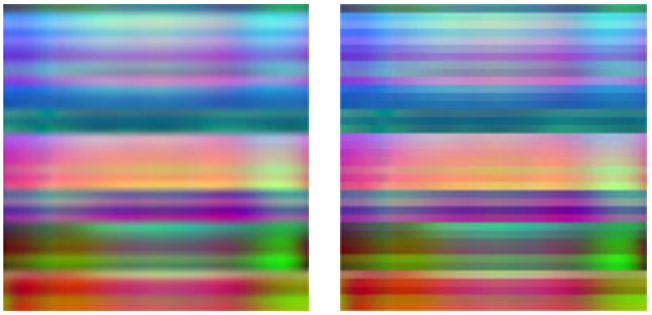
FIGURE 3 Transformation of motion capture sequence into an RGB image: (a) X, Y, and Z coordinates of random sequence of one joint; (b) normalized of the sequence between 0 and 255; (c) reconstructed $1 * 20$ color image

approximately. In the case of a high-precision mocap system such as Vicon* or Qualisys,† which runs at a frame rate of 180 fps for instance, the width of the resulting image will be 900 pixels. Moreover, the sequences that will be used for classification have different dimensions, depending on the number of frames. This is hard for the classifier to handle because the lengths of the feature vectors will be different. We resize the images using a bicubic interpolation[19] to have a fixed size of $256 * 256$ for all sequences. Figure 4a shows a result of transforming a sequence of 380 frames long and 38 markers. The resulting image is very blurry. Enlarging an image makes more loss than shrinking it.[19] To avoid this loss, we first create a square image by repeating each row (relative to joints) m times, where m is calculated as follows:

$$m = floor\left(\frac{F}{N}\right). \tag{2}$$

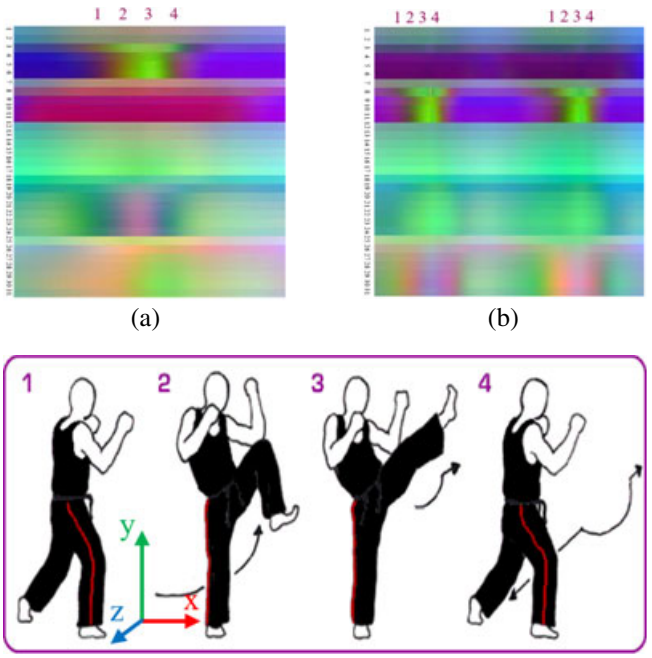
In this example, $m = 10$. Each row is repeated 10 times, which gives a $380 * 380$ square image; then, the interpolation is applied to fix the size to $255 * 255$. Figure 4b shows the new constructed image, which has higher quality and smoother edges.

*https://www.vicon.com/
†https://www.qualisys.com/



(a) (b)

FIGURE 4 Reconstructed image after a bicubic interpolation: (a) without preprocessing; (b) with preprocessing



(a) (b)

(c)

FIGURE 5 Reconstructed images from kicking sequences: (a) front kick with the left leg; (b) front kicks with the right leg; (c) illustration of the left kick

Figure 5 shows two examples of transformed sequences from the HDM05 dataset.[20] The images represent two kicking mocap sequences: a front kick using the left leg and two front kicks using the right leg. Original mocap data, recorded using a Vicon mocap system, contains 3D positions of 31 markers. Each distinctive row represents a sequence of one marker, where the order of markers is shown in Figure 6.

First, to perform a front kick using the left foot in martial arts, the person starts from standing in a combative stance with the left leg behind the other leg and hands are in a guard position (step 1 in Figure 5c). The left knee is then raised so that the thigh is parallel to the ground, or higher, in some cases (step 2 in Figure 5c). The third step is to kick the leg, snapping it forward quickly (step 3 in Figure 5c), and the last step is to unsnap the leg so that the thigh is once again parallel, or higher, to the ground and then setting it back on the ground. It is a symmetric gesture, where the step 3 in Figure 5c is the gesture of symmetry.
The two images in Figure 5a and 5b are self-explanatory. The first one is a front kick with the left leg.

When the person raises the left leg, the red, and particularly the green, components of rows 3–6, relative to the left leg markers, increase. Furthermore, the red and green components of rows 18–24, relative to the left arm markers, increase, although they decrease for the right arm markers (rows 26–31). This happens because the person raises the left arm and moves forward at the same time while he/she does the opposite for the right arm. We can also clearly see the symmetry of the gesture in the image. The second image represents two consecutive front kicks with the right leg. The previous description also applies to this image. This representation is powerful to have an overview of a mocap sequence without complicated processing techniques.

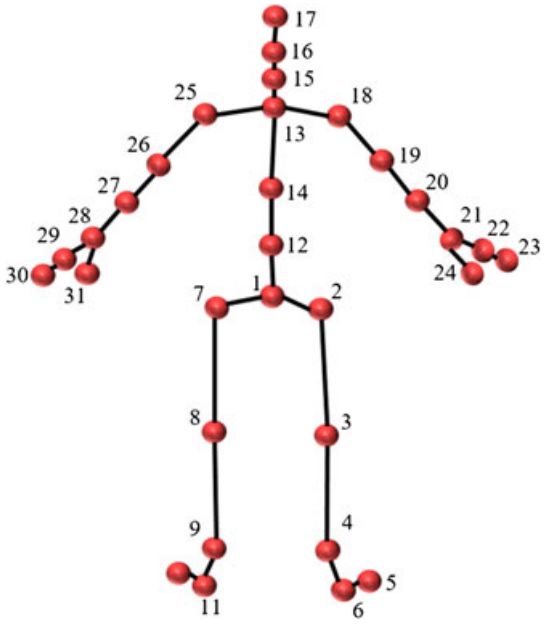


FIGURE 6 Schematic representation of 31-joint skeleton from HDM05 dataset

The generated images are used to train different classifiers to see the efficiency of this representation. In the next section, we compare our representation, applied on three public datasets and trained with four classifiers, to some results obtained from related works.

4 | EXPERIMENTS

The proposed method was evaluated on three public benchmark datasets: the Microsoft Research (MSR) Action 3D dataset,[21] the Hochschule der Medien (HDM05) dataset,[20] and the large Nanyang Technological University (NTU) RGB+D dataset.[22] Note that these datasets are composed of everyday human actions (e.g., drinking, jumping, and calling) in addition to sports actions (e.g., kicking, punching, and dancing). Moreover, the datasets were recorded with different mocap systems of different precisions and frame rates. The proposed representation was tested with four classification algorithms: multiclass SVM, KNN, RF, and CNN.

Raw 256×256 images were used to train each classifier. In the first three classifiers, a matrix was generated for the entire training dataset. Each image was represented by a row in the matrix. The number of columns was $256 * 256 = 65,536$. Thus, the dimension of the training matrix was $N * 65,536$ (N is the number of images in the training dataset). MATLAB software[‡] was used to run different experiments without changing default parameters.

In the case of CNN classifier, we focus on a popular architecture, namely, GoogleNet,[8] which was designed in the context of "Large Scale Visual Challenge" for the ImageNet.[11] This architecture is very deep and wide with 22 layers. We analyze the performances of this architecture on the three datasets by training the models from scratch in one case, and then by fine-tuning already trained models (i.e., trained in the ImageNet dataset) using transfer learning. Each experiment runs for a total of 50 epochs, where one epoch is defined as the number of training iterations in which the particular neural network has completed a full pass of the whole training set. The choice of 50 epochs was made based on the empirical observation that, in all of these experiments, the learning always converged within 50 epochs. All the experiments were conducted using the open-source NVIDIA Digits,[§] which is an interactive deep learning development tool that integrates Berkeley's Caffe framework[¶] with a friendly web-based graphical user interface.

[‡]MATLAB, the Language of Technical Computing: https://nl.mathworks.com/products/matlab.html
[§]The NVIDIA Deep Learning GPU Training System (DIGITS): https://github.com/NVIDIA/DIGITS
[¶]Caffe Deep Learning Framework: http://caffe.berkeleyvision.org/

TABLE 1 Recognition accuracy with and without fine-tuning for three datasets

Dataset	Without fine-tuning	With fine-tuning
MSR Action 3D	63.97%	92.18%
HDM05	53.30%	83.33%
NTU RGB+D (cross-subject)	66.83%	74.27%

We evaluate our results by computing the accuracy (Acc), where we compare the predicted classes ($y^i_{\text{predicted}}$) with the ground truth labels (y^i_{labels}). It is the number of correct predictions divided by the total number of testing samples (M).

$$E\left(y^i_{\text{predicted}}, y^i_{\text{labels}}\right) = \begin{cases} 1, & y^i_{\text{predicted}} = y^i_{\text{labels}} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

$$Acc = \frac{100}{M} \sum_{i=1}^{M} E\left(y^i_{\text{predicted}}, y^i_{\text{labels}}\right) \tag{4}$$

From Table 1, we see that the recognition accuracy increased by almost 30% for MSR Action 3D and HDM05 after fine-tuning. Even though the size of the training set of the NTU RGB+D is large enough to train a deep network (around 500 samples per class), accuracy increased by more than 7% after fine-tuning.

4.1 | MSR Action 3D dataset

MSR Action 3D[21] is one of the earliest datasets recorded using a depth sensor. The samples of this dataset were limited to depth sequences. Later, body joint information was added. There are 20 action classes performed by 10 subjects. Each action was performed twice to three times by each subject. In our experiment, we focused only on the skeletal data. There are in total 557 skeletal sequences, where each sequence has 20 joint positions. In this dataset, we use the cross-subject experiment, where the sequences of five subjects are used in training and the rest are used for testing. Table 2 compares our results with some of the state-of-the-art skeleton-based action recognition approaches. The proposed representation used with CNN shows its superiority over other techniques with an accuracy of 92.18%. Besides, our RGB representation used with KNN and RF achieves acceptable results compared to other techniques.

TABLE 2 Comparison of the proposed method with existing methods on the MSR Action 2D dataset

Method	Accuracy
Sequence of Most Informative Joints[23]	29.41%
Recurrent neural network[24]	42.50%
Dynamic time warping[25]	54.00%
Hidden Markov models[26]	63.00%
Multiple instance learning[27]	65.70%
EigenJoints + NBNN[28]	72.00%
Structured Streaming Skeletons[29]	81.70%
DBN + HMM[30]	82.00%
Conceptors of Skeleton Joint Trajectories[17]	83.40%
Seq2Im + SVM	57.44%
Seq2Im + KNN	72.55%
Seq2Im + RF	77.94%
Seq2Im + CNN (fine-tuning)	**92.18%**

Note. NBNN = Naïve-Bayes-nearest-neighbor; DBN = deep belief network; HMM = hidden Markov model; Seq2Im = sequence to image; SVM = support vector machine; KNN = k-nearest neighbor; RF = random forest; CNN = convolutional neural network.

TABLE 3 Comparison of the proposed method with existing methods on the HDM05 dataset

Method	Accuracy
SPDNet[31]	61.45%
SE[3]	70.26%
SO[32]	71.31%
LieNet[16]	75.78%
Seq2Im + SVM	70.70%
Seq2Im + KNN	66.82%
Seq2Im + RF	**80.62%**
Seq2Im + CNN (fine-tuning)	**83.33%**

Note. SPDNet = symmetric positive definite network; SE = special Euclidean group; SO = special Orthogonal group; Seq2Im = sequence to image; SVM = support vector machine; KNN = *k*-nearest neighbor; RF = random forest; CNN = convolutional neural network.

4.2 | HDM05 dataset

HDM05[20] contains 2,343 sequences of 130 classes executed by various actors. Each action was performed 10 to 50 times by each actor. The dataset was recorded using a Vicon mocap system, where 31 reflective markers were placed on the actors' bodies. The 3D positions of these markers are provided.

Following Huang et al.,[31] we conducted 10 evaluations, each of which selects randomly half of the sequences for training and the other half for testing. However, due to the long time it takes to train CNNs, we only ran one experiment for this case. Table 3 lists the average accuracy of the proposed method and the results obtained in the previous works. Our representation achieved the highest results in the case of RF classifier and CNNs.

4.3 | NTU RGB+D dataset

To the best of our knowledge, the NTU RGB+D dataset[22] is currently the largest action recognition dataset. It was collected using three Kinect V2 sensors at the same time covering three views (i.e., $-45°, 0°, 45°$) and contains more than 56,000 action sequences. A total of 60 different action classes are performed by 40 subjects aging between 10 and 35 years. In addition to depth maps, RGB frames, and infrared (IR) sequences, information of 25 3D joints is available. This dataset is challenging because of the large intraclass and viewpoint variations; however, due to its large scale, it is highly suitable for deep learning. For evaluation, this dataset has two standard testing protocols. One is cross-subject, for which half of the subjects are used for training and the rest are used for testing. The other one is a cross-view test, for which two views are used for training and the other one is used for testing. Due to the large size of this dataset, the tool we used (MATLAB) was not able to process the data. We focused instead on classification using CNN. We compared our method, with and without fine-tuning, to the state-of-the-art

TABLE 4 Comparison of the proposed method with existing methods on the NTU RGB+D dataset

Method	Cross-subject	Cross-view
HBRNN[33]	59.07%	63.97%
Deep RNN[22]	56.29%	64.09%
Deep LSTM[22]	60.69%	67.29%
PA-LSTM[22]	62.93%	70.27%
LieNet[16]	61.37%	66.95%
ST-LSTM[34]	69.20%	**77.70%**
Seq2Im + CNN (no fine-tuning)	66.83%	66.31%
Seq2Im + CNN (fine-tuning)	**74.27%**	75.74%

Note. HBRNN = hierarchically bidirectional recurrent neural networks; RNN = recurrent neural network; LSTM = long short-term memory; PA-LSTM = part-aware long short-term memory; ST-LSTM = spatio-temporal LSTM; Seq2Im = sequence to image; CNN = convolutional neural network.

methods. Results are summarized in Table 4. Our method has high accuracy results and overcame other techniques in the case of cross-subject evaluation. In the case of cross-view evaluation, our method did not obtain the highest accuracy, but the results were competitive. Most of the confusions were related to the pairs of actions "reading" and "writing," "playing with phone/tablet" and "typing on keyboard," "pat on back of other person" and "point finger at the other person," and "giving something to other person" and "touch other person's pocket." These pairs are very similar and could not be handled correctly by our algorithm.

5 | CONCLUSION AND FUTURE WORKS

This paper addresses the problem of skeleton-based human action recognition. An effective, yet simple, method is proposed to represent a skeleton sequence into a 2D-RGB image. Such a representation allows us to use powerful image classifiers to recognize human actions, particularly CNNs. Furthermore, this imagelike representation of 3D skeleton sequences allows fine-tuning the existing CNN models without training a whole deep network from scratch. The experimental results on three public datasets have shown the efficiency of this representation even without extracting complex features. It should be noted that these three datasets were recorded with different mocap systems, from depth sensors (such as the two versions of Microsoft Kinect) to a high-precision mocap system (such as Vicon), which makes our method independent of the quality of data and the number of joints (makers). This results from the fact that image-processing techniques deal generally with noise. Moreover, they are scale and translation invariant, particularly in the case of CNNs. In future works, we aim to improve our representation and extend the method to online action recognition.

ACKNOWLEDGEMENTS

This work was partially financed by the institutes Numediart and Infortech of the University of Mons (UMONS). This work is also partially funded by the EU and Wallonia in the framework of the ERDF/DIGISTORM project in addition to the European Union (FP7-IC7-2011-9) under grant agreement n 600676 (i-Treasures project).

REFERENCES

1. Poppe R. A survey on vision-based human action recognition. Image Vis Comput. 2010;28(6):976–990.
2. Aggarwal JK, Xia L. Human activity recognition from 3D data: A review. Pattern Recogn Lett. 2014;48:70–80.
3. Vemulapalli R, Arrate F, Chellappa R. Human action recognition by representing 3D skeletons as points in a lie group. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2014 June 23–28; Columbus, OH, USA; p. 588–595.
4. Gowayyed MA, Torki M, Hussein ME, El-Saban M. Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition. Paper presented at: IJCAI; 2013.
5. Xia L, Chen C-C, Aggarwal JK. View invariant human action recognition using histograms of 3D joints. Paper presented at: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE; 2012 June 16–21; Providence, RI, USA; p. 20–27.
6. Taigman Y, Yang M, Ranzato MA, Wolf L. DeepFace: Closing the gap to human-level performance in face verification. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2014 June 23–28; Columbus, OH, USA; p. 1701–1708.
7. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Paper presented at: Advances in Neural Information Processing Systems; 2012. p. 1097–1105.
8. Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015 June 7–12; Boston, MA, USA; p. 1–9.
9. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–444.
10. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016 June 27–30; Las Vegas, NV, USA; p. 770–778.
11. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. Paper presented at: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2009 June 20–25; Miami, FL, USA; p. 248–255.
12. Zhang C, Tian Y. RGB-D camera-based daily living activity recognition. J Comput Vis Image Process. 2012;2(4):12.
13. Sempena S, Maulidevi NU, Aryan PR. Human action recognition using dynamic time warping. Paper presented at: 2011 International Conference on Electrical Engineering and Informatics (ICEEI); 2011 July 17–19; Bandung, Indonesia; p. 1–5.
14. Bloom V, Makris D, Argyriou V. G3D: A gaming action dataset and real time action recognition evaluation framework. Paper presented at: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); 2012 June 16–21; Providence, RI, USA; p. 7–12.
15. Laraba S, Tilmanne J. Dance performance evaluation using hidden Markov models. Comput Anim Virtual Worlds. 2016;27(3-4):321–329.
16. Huang Z, Wan C, Probst T, Van Gool L. Deep learning on lie groups for skeleton-based action recognition. arXiv preprint arXiv:1612.05877; 2016.

17. Bao J. Action recognition based on conceptors of skeleton joint trajectories. Rev Fac Ing. 2016;31(4):11–22.
18. Jaeger H. Controlling recurrent neural networks by conceptors. arXiv preprint arXiv:1403.3369; 2014.
19. Prashanth HS, Shashidhara HL, Balasubramanya Murthy KN. Image scaling comparison using universal image quality index. Paper presented at: 2009 International Conference on Advances in Computing, Control, & Telecommunication Technologies (ACT'09). IEEE; 2009 December 28–29; Bangalore, India, India; p. 859–863.
20. Müller M, Röder T, Clausen M, Eberhardt B, Krüger B, Weber A. Documentation mocap database HDM05. Technical Report CG-2007-2. Universität Bonn; June 2007.
21. Li W, Zhang Z, Liu Z. Action recognition based on a bag of 3D points. Paper presented at: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE; 2010 June 13–18; San Francisco, CA, USA; p. 9–14.
22. Shahroudy A, Liu J, Ng T-T, Wang G. NTU RGB+D: A large scale dataset for 3D human activity analysis. Paper presented at: IEEE Conference on Computer Vision and Pattern Recognition (CVPR); June 2016.
23. Ofli F, Chaudhry R, Kurillo G, Vidal R, Bajcsy R. Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition. J Vis Commun Image Represent. 2014;25(1):24–38.
24. Martens J, Sutskever I. Learning recurrent neural networks with Hessian-free optimization. Paper presented at: Proceedings of the 28th International Conference on Machine Learning (ICML-11); 2011 June 28–July 2; Bellevue, Washington, USA; p. 1033–1040.
25. Müller M, Röder T. Motion templates for automatic classification and retrieval of motion capture data. Paper presented at: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Eurographics Association; 2006 September 2–4; Vienna, Austria; p. 137–146.
26. Lv F, Nevatia R. Recognition and segmentation of 3-D human action using HMM and multi-class AdaBoost. Paper presented at: European Conference on Computer Vision. Springer; 2006. p. 359–372.
27. Ellis C, Masood SZ, Tappen MF, LaViola JJ, Sukthankar R. Exploring the trade-off between accuracy and observational latency in action recognition. Int J Comput Vis. 2013;101(3):420–436.
28. Yang X, Tian YL. EigenJoints-based action recognition using naive-Bayes-nearest-neighbor. Paper presented at: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE; 2012 June 16–21; Providence, RI, USA; p. 14–19.
29. Zhao X, Li X, Pang C, Zhu X, Sheng QZ. Online human gesture recognition from motion data streams. Paper presented at: Proceedings of the 21st ACM International Conference on Multimedia. ACM; 2013 October 21–25; Barcelona, Spain; p. 23–32.
30. Wu D, Shao L. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2014 June 23–28; Columbus, OH, USA; p. 724–731.
31. Huang Z, Van Gool L. A Riemannian network for SPD matrix learning. arXiv preprint arXiv:1608.04233; 2016.
32. Vemulapalli R, Chellapa R. Rolling rotations for recognizing human actions from 3D skeletal data. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2016 June 27–30; Las Vegas, NV, USA; p. 4471–4479.
33. Du Y, Wang W, Wang L. Hierarchical recurrent neural network for skeleton based action recognition. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2015 June 7–12; Boston, MA, USA; p. 1110–1118.
34. Liu J, Shahroudy A, Xu D, Wang G. Spatio-temporal LSTM with trust gates for 3D human action recognition. Paper presented at: European Conference on Computer Vision. Springer; 2016 October 11–14; Amsterdam, The Netherlands; p. 816–833.

Sohaib Laraba is a PhD student at the University of Mons (UMONS) in Belgium. His research includes analysis, recognition, and assessment of stylistic gestures captured by different motion capture systems, and particularly low-cost ones, using machine learning techniques.

Mohamed Brahimi is PhD student at the University of Sciences and Technology Houari Boumediene (USTHB) of Algiers in Algeria and computer science assistant professor at University of Mohamed El Bachir El Ibrahimi of Bordj Bou Arreridj in Algeria. His research includes machine learning and particularly deep learning applications, images processing, and the extraction of knowledge from deep architectures.

Joëlle Tilmanne is a postdoctoral researcher at UMONS and is the head of the motion capture and analysis research group at the numediart Institute. She holds a PhD in Applied Sciences from UMONS Faculty of Engineering since 2012 in the field of motion capture data analysis and hidden Markov model-based motion synthesis. She is the co-founder of Hovertone, a young startup active in the domain of creative experience design.

Thierry Dutoit teaches Circuit Theory, Signal Processing, Applied Signal Processing, and Biomedical Signal processing. His research interests are in speech and audio processing, biomedical signal processing, as well as in real-time signal processing. He heads the numediart Institute for New Media Art Technology.

How to cite this article: Laraba S, Brahimi M, Tilmanne J, Dutoit T. 3D skeleton-based action recognition by representing motion capture sequences as 2D-RGB images. *Comput Anim Virtual Worlds*. 2017;28:e1782. https://doi.org/10.1002/cav.1782