A Robust Human Activity Recognition Approach Using OpenPose, Motion Features, and Deep Recurrent Neural Network

Farzan Majeed Noori[1](✉), Benedikte Wallace[1,2], Md. Zia Uddin[1], and Jim Torresen[1,2]

[1] Department of Informatics, University of Oslo, Oslo, Norway
{farzanmn,benediwa,mdzu,jimtoer}@ifi.uio.no
[2] RITMO Centre for Interdisciplinary Studies in Rhythm, Time and Motion, Oslo, Norway

Abstract. With the emerging advancements in computer vision and pattern recognition, methods for human activity recognition have become increasingly accessible. In this paper, we present a robust approach for human activity recognition which uses the open source library *OpenPose* to extract anatomical key points from RGB images. We further use these key points to extract robust motion features considering their movements in consecutive frames'. Then, a Recurrent Neural Network (RNN) with Long Short-term Memory cells (LSTM) is used to recognize the activities associated with these features. To make the approach person-independent, different subjects from different camera angles are used. The proposed method shows promising performance, with the best result reaching an overall accuracy of 92.4% on a publicly available activity data set, which outperforms the conventional approaches (i.e. support vector machines, decision trees, and random forests) which achieve maximum accuracy of 78.5%. The proposed activity recognition system can contribute in prominent research fields such as image processing and computer vision with practical applications such as caregiving for older people to help them live more independently.

Keywords: Human activity recognition · LSTM · OpenPose · RNNs

1 Introduction

Nowadays, perceiving human activity is one of the vital areas of computer vision research. The goal of human activity recognition (HAR) is to distinguish and analyze activities based on data extracted from sensors such as wearables [17]

First and second authors contributed equally to this work. This work is partially supported by The Research Council of Norway (RCN) as a part of the Multimodal Elderly Care systems (MECS) project, under grant agreement 247697 and the RCN Centres of Excellence scheme, project number 262762.

M. Felsberg et al. (Eds.): SCIA 2019, LNCS 11482, pp. 299–310, 2019.
https://doi.org/10.1007/978-3-030-20205-7_25

or external sensing modalities. HAR can be used in smart-homes [18,22], sports [16], as well as in health monitoring [19], assistance for the elderly [13] and in mental health care [8].

Recently, there has been an increase in the popularity of external sensors such as Kinect and RGB sensors over wearable sensors as they can be seen as less intrusive. In various computer vision applications, depth images have been used widely among researches. In [24], temporal motion energies were extracted from human activity depth images and a Hidden Markov Model (HMM) was applied to model the activities using depth image features [14]. In computer vision, 2D pose estimation is widely used and several algorithms have been proposed to localize body joints. Fujimori et al. developed a wearable suit to capture body motion with tactile sensors, using motion sensors to estimate the user's orientation [10]. Liu et al. obtained static gestures from individual pictures by using skeletal tracking with a Kinect camera [15].

These conventional approaches of activity recognition, though suitable for a variety of tasks, are often slow and lack reliability and performance in complex environments. *OpenPose* [6,23], an open source library developed at Carnegie Mellon University in 2017 achieved significant interest among researchers due to its computational performance for extraction of body joints. OpenPose can operate in real-time detecting facial expressions, body and hand joints by feeding RGB images through deep convolutional neural networks (CNNs) [20,21].

In this work, we propose deep learning approach using *OpenPose* and recurrent neural nets (RNN) to facilitate activity recognition. The *OpenPose* library was used to detect 14 body joints. Using this data, we extract the changes in magnitude and angle between joints in consecutive frames. These robust features are then used as the input for a sequence classifier which uses an RNN with Long Short-term Memory cells (LSTM). As LSTMs have the ability to retain salient information over a sequence of time steps, it lends itself well to sequence classification tasks such as activity recognition. In this work, the LSTM is trained to recognize which activities are performed by learning the sequence of motion features associated with each activity. The main contributions of this work, therefore, are the extraction of robust motion features from the body joints acquired using *OpenPose*, combined with the use of LSTM-RNNs to recognize the activities and boost the performance compared to other conventional approaches such as SVM, Decision Trees, and Random Forests.

This paper is organized as follows: Sect. 2 describes our methodology, where we discuss the data set and the extraction of motion features, as well as give an overview of the LSTM architecture and the other classifiers compared with our experiments. Section 3 presents the experiments and results. The paper is concluded in Sect. 4 where we also discuss the limitations of our approach and possibilities for future work.

2 Methodology

In this section, we present the steps included in our proposed method of activity recognition. Specifically, Sect. 2.1 describes the extraction of key points using

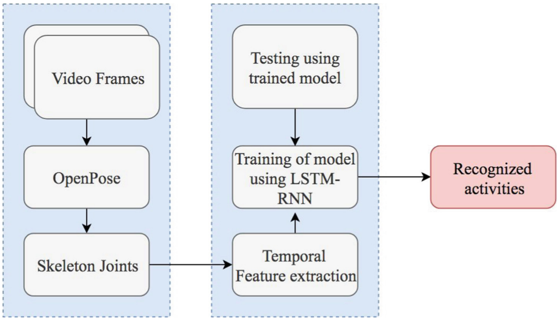

Video Frames
OpenPose
Skeleton Joints
Testing using trained model
Training of model using LSTM-RNN
Temporal Feature extraction
Recognized activities

Fig. 1. Flow of work

OpenPose and Sect. 2.2 defines the motion features. The data set and classifiers are described in Sects. 2.3 and 2.4, respectively. Figure 1 gives an overview of the general flow of our process.

2.1 Extracting Joints

OpenPose takes RGB images as input and generates 2-dimensional anatomical key-points for individual bodies detected in the image. The first stage in two-branched CNN predicts confidence maps, and the second stage predicts part affinity fields - a 2D vector which encodes the position and orientation of each limb [5].

Furthermore, both the confidence maps and affinity fields are parsed by greedy inference to visualize the 2D key-points of all individuals in the image [7,23]. *OpenPose* generates the location of 18 body joints. These body joints can then be exported and used for applications such as gesture and activity recognition.

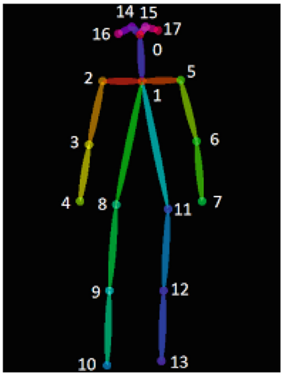


Fig. 2. OpenPose skeleton joints

2.2 Motion Features

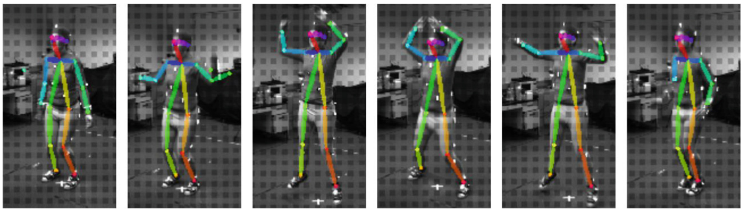


Fig. 3. Extracted body joints from several frames while performing jumping jacks

In our work, we compute the two temporal features magnitude and angle from 14 body joints (L) between consecutive frames. Joints 14, 15, 16 and 17 pertain to the face and head as shown in Fig. 2. These key points are excluded from the data as they are not necessarily useful for activity recognition. Formally, we derive the magnitude M at a joint number N at time frame t as follows:

$$M_t^N = \sqrt{(L_{Nx(t+1)} - L_{Nx(t)})^2 + (L_{Ny(t+1)} - L_{Ny(t)})^2} \tag{1}$$

The angles of a body joint N for frame t^{th} are computed as follows:

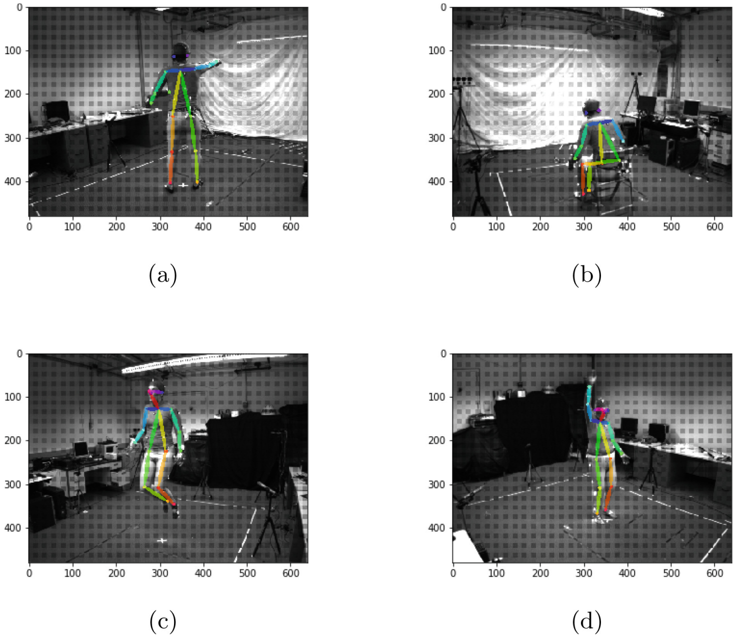
$$A_{N(x,y)t} = tan^{-1}(\frac{L_{Ny(t+1)} - L_{Ny(t)}}{L_{Nx(t+1)} - L_{Nx(t)}}) \tag{2}$$

where L_{Nx} and L_{Ny} are the distances between the two-consecutive frames in x-axis and y-axis. For each example, we extract the body joints from a sequence of consecutive frames. Figure 3 shows the several frames of jumping jacks, while the dotted lines represent the joint-to-joint connection for motion features. Thus, the motion features (T) at time step t can be represented as:

$$T_t = [M_t^N, A_{N(x,y)t}] \tag{3}$$

2.3 Data Set

Our data set is a subset of the Berkeley Multimodal Human Action Database [1]. MHAD contains 11 actions, as listed in Table 1. These actions are performed by 7 male and 5 female subjects recorded using audio, video, accelerometers, motion capture, and kinect. For current work, we have chosen to use a subset containing the image sequences captured by 12 RGB cameras placed in clusters surrounding the participants, achieving views from the front and back as shown in Fig. 4. We use the images produced by the video recordings of all 12 subjects performing each action for approximately 5 s. The image sequences are captured by each of the four camera clusters as shown in Fig. 5. Cluster C1 and C2 contain



(a)

(b)

(c)

(d)

Fig. 4. Examples of images from the MHAD [1] database.

four cameras. The remaining two clusters, C3 and C4 contain two cameras each. We include the images from all four clusters in an effort to make our system view-invariant, as each camera captures the video at a different angle.

The number of images in each recording varies from around 40 (approx. 3 s of video at 22 Hz) to 130 frames (approx. 10 s of video). Activities which involved less complex movements, such as standing up or sitting down, consist of fewer than 40 frames. To mediate this difference in sequence lengths and ensure consistency in the data-set, the longer sequences were clipped after 85 frames and the shorter sequences were extended by stacking them. Each camera captures 132 sequences. This results in a data-set of 1584 sequences of 85 images each consisting of 12 participants performing 11 actions captured by 12 cameras. Each sequence of images thereby represents the view captured by a single camera in one of the clusters and a single data point in training and test data. Finally, z-score standardization was applied before applying the classifiers to the data.

2.4 Classification

Recurrent Neural Networks (RNN). An RNN with LSTM cells was implemented in order to tackle the long term dependencies found in our data. RNNs and LSTMs have previously been shown to be effective in modeling temporal sequences such as those found in speech [11] and handwriting recognition [12] and also in music [9]. This is due to their ability to retain 'memory' over

Table 1. Activity labels: 11 classes

Class label	Activity
1	Jumping
2	Jumping Jacks
3	Bending (Hands up)
4	Punching
5	Waving (Both hands)
6	Waving (One hand)
7	Clapping
8	Throwing a ball
9	Sit down then stand up
10	Sit down
11	Stand up

several time steps by allowing the states from preceding time steps to affect the RNNs current state. While this makes the architecture a good choice for modeling time series data, some limitations exist when dealing with longer time series. If the sequence length becomes too long, the RNN may suffer from vanishing or exploding gradients during back-propagation through time (BPTT). LSTMs mitigates this problem by adding multiple learnable parameters, or gates, which affect weight updates during BPTT enabling more control over what is retained in the internal state of the LSTM cell and what it 'forgets' at each time step. As illustrated in Fig. 6, the features which describe the variations of magnitude and angle of each joint between consecutive frames are fed to the network at each time step.

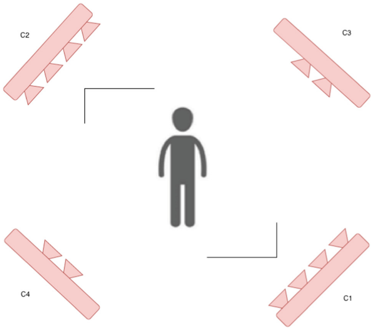


Fig. 5. Setup of cameras from Berkeley MHAD acquisition system [1]

The model used in this work consist of two layers containing 256 LSTM cells with hyperbolic tangent activation function. The output layer is a dense layer with softmax activation and 11 units representing the different activities. Categorical cross entropy is used as the loss function during batch gradient descent and RMSprop with a learning rate of 0.001, is used as the optimizer. The model is trained for 300 epochs with a batch size of 256 samples and a 0.4 dropout rate. These hyperparameters were chosen empirically according to which values yielded the best results for the task.

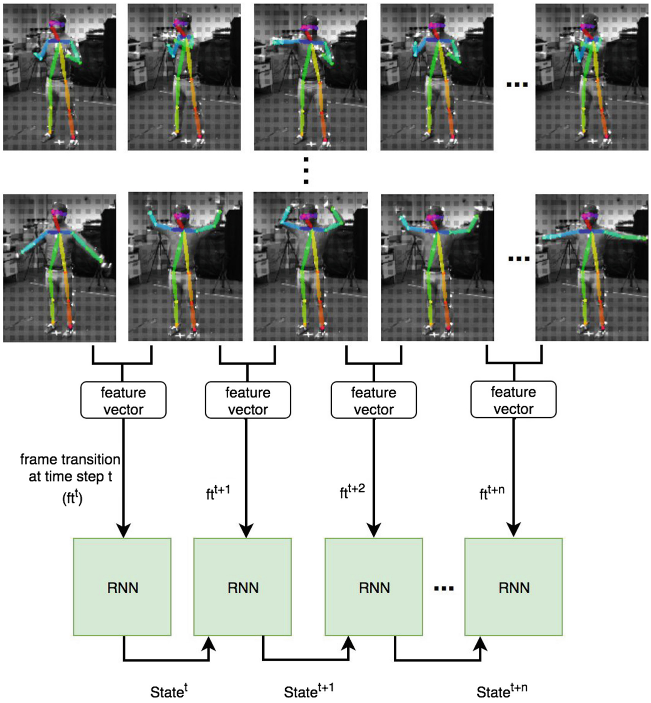


Fig. 6. LSTM-RNN implementation for all activities

Support Vector Machine (SVM). SVM has been widely used in HAR systems due to its high classification performance [2,3]. SVM creates hyper-planes that maximize the margins between several classes, which enables maximum classification accuracy. The vectors are used to represent hyper-planes are called support vectors. By minimizing the cost function, an optimal solution can be obtained i.e. maximize the distance between hyper-plane and the nearest training point. Herein, a non-linear multi-class SVM with *sigmoid* kernel was used.

Decision Trees. A decision support tool that utilizes a model of decisions or tree-like graph and their possible consequences including utility, and chance event outcomes, called a decision tree. A decision tree is a well-known classifier in machine learning. Its structure is similar to the flowchart in which each internal node represents a test on an attribute; for instance, whether it would be heads or tails by flipping a coin. Each branch is responsible for the test's outcome, and the class label would be represented by each leaf node. The decision would be taken after applying all features. The classification rules would be based on the paths from the root to the leaf [4].

Random Forests. Random Forests method used in both classification and regression problems. It generates multiple decision trees based on the random selection of variables and data, and recognizes the classes of dependent variables based on decision trees. In this work, 10 decision trees were used to explore the classes.

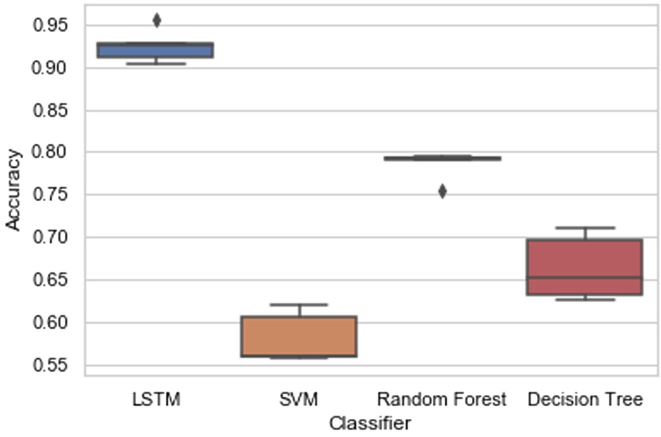


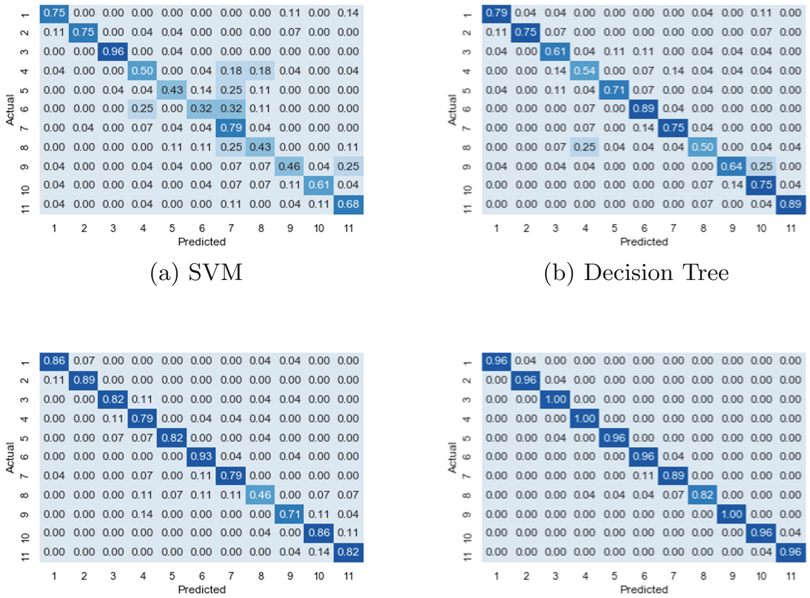
Fig. 7. Box plot showing the accuracies achieved by each model at each of the 5 folds

3 Experiments and Results

In this section, the experiments performed to validate the performance of the proposed method are explained. The accuracy of each classifier is evaluated by performing stratified 5-fold cross-validation. The average accuracy of each classifier is listed in Table 2. Figure 7 shows the distribution of the results achieved by the different classifiers. Figure 8 and Table 3 displays confusion matrices and the precision and recall values generated from the *predictions* of each classifier from a single fold of the data. Moreover, Fig. 9 shows the confusion matrices of the *number of samples* classified as belong to each class.

Table 2. Average accuracy from 5-fold cross validation of each classifier

Classifier	Accuracy (%)
SVM	58.1
Decision Trees	66.3
Random Forests	78.5
LSTM	**92.4**



	1	2	3	4	5	6	7	8	9	10	11
1	0.75	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.00	0.14
2	0.11	0.75	0.00	0.04	0.04	0.00	0.00	0.00	0.07	0.00	0.00
3	0.00	0.00	0.96	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00
4	0.04	0.00	0.00	0.50	0.00	0.04	0.18	0.18	0.04	0.00	0.04
5	0.00	0.00	0.04	0.04	0.43	0.14	0.25	0.11	0.00	0.00	0.00
6	0.00	0.00	0.00	0.25	0.00	0.32	0.32	0.11	0.00	0.00	0.00
7	0.00	0.04	0.00	0.07	0.04	0.04	0.79	0.04	0.00	0.00	0.00
8	0.00	0.00	0.00	0.00	0.11	0.11	0.25	0.43	0.00	0.00	0.11
9	0.04	0.00	0.00	0.04	0.04	0.00	0.07	0.07	0.46	0.04	0.25
10	0.04	0.00	0.00	0.04	0.00	0.04	0.07	0.07	0.11	0.61	0.04
11	0.04	0.00	0.00	0.04	0.00	0.00	0.11	0.00	0.04	0.11	0.68

(a) SVM

	1	2	3	4	5	6	7	8	9	10	11
1	0.79	0.04	0.04	0.00	0.00	0.00	0.00	0.04	0.00	0.11	0.00
2	0.11	0.75	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00
3	0.04	0.00	0.61	0.04	0.11	0.11	0.00	0.04	0.04	0.04	0.00
4	0.00	0.00	0.14	0.54	0.00	0.07	0.14	0.04	0.04	0.04	0.00
5	0.04	0.00	0.11	0.04	0.71	0.07	0.00	0.04	0.00	0.00	0.00
6	0.00	0.00	0.00	0.07	0.00	0.89	0.04	0.00	0.00	0.00	0.00
7	0.00	0.00	0.00	0.07	0.00	0.14	0.75	0.04	0.00	0.00	0.00
8	0.00	0.00	0.07	0.25	0.04	0.04	0.04	0.50	0.00	0.04	0.04
9	0.04	0.00	0.04	0.04	0.00	0.00	0.00	0.00	0.64	0.25	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.14	0.75	0.04
11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.04	0.89

(b) Decision Tree

	1	2	3	4	5	6	7	8	9	10	11
1	0.86	0.07	0.00	0.00	0.00	0.00	0.00	0.04	0.04	0.00	0.00
2	0.11	0.89	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.00	0.00	0.82	0.11	0.00	0.00	0.00	0.04	0.04	0.00	0.00
4	0.00	0.00	0.11	0.79	0.00	0.04	0.04	0.04	0.00	0.00	0.00
5	0.00	0.00	0.07	0.07	0.82	0.00	0.00	0.04	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00	0.93	0.04	0.00	0.04	0.00	0.00
7	0.04	0.00	0.00	0.07	0.00	0.11	0.79	0.00	0.00	0.00	0.00
8	0.00	0.00	0.00	0.11	0.07	0.11	0.11	0.46	0.00	0.07	0.07
9	0.00	0.00	0.00	0.14	0.00	0.00	0.00	0.00	0.71	0.11	0.04
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.86	0.11
11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.14	0.82

(c) Random Forest

	1	2	3	4	5	6	7	8	9	10	11
1	0.96	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.96	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.04	0.00	0.96	0.00	0.00	0.00	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00	0.96	0.04	0.00	0.00	0.00	0.00
7	0.00	0.00	0.00	0.00	0.00	0.11	0.89	0.00	0.00	0.00	0.00
8	0.00	0.00	0.00	0.04	0.04	0.04	0.07	0.82	0.00	0.00	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.96	0.04
11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.96

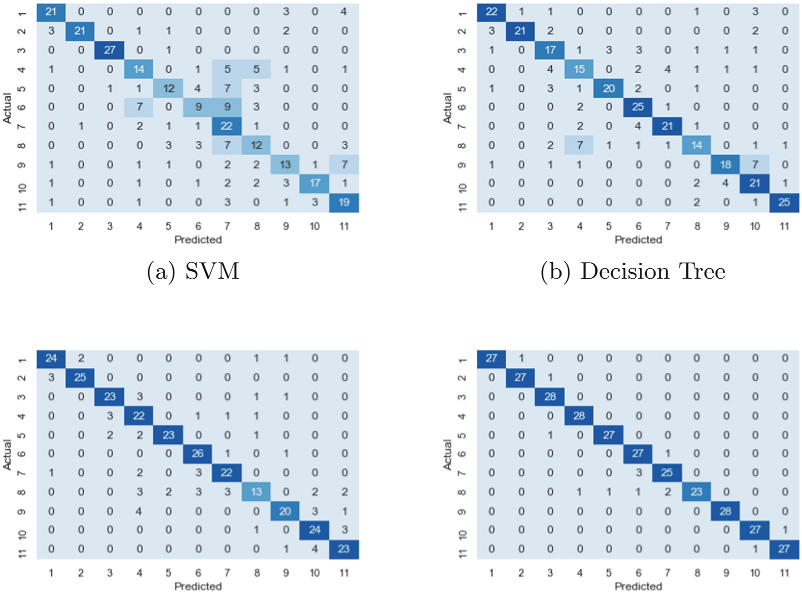
(d) LSTM

Fig. 8. Confusion matrices for (a) SVM, (b) Decision Trees, (c) Random Forests and (d) LSTM calculated on a test set of 308 sequences. These show the percentage of predictions for each class

As shown in Fig. 8(a), SVM was unable to distinguish the difference among waving one hand, waving both hands and clapping activities. Furthermore, Decision trees and Random forests showed better performance i.e. 0.66 and 0.78 as shown in Table 2. Moreover, both classifiers outperformed SVM in precision and recall, respectively. Still, both were lacking in their ability to differentiate all activities properly. The LSTM model outperformed the conventional approaches, achieving the highest average accuracy at 92,4%. Student's t-test was applied to validate the statistically discriminant. The p values obtained LSTM versus other approaches were less than 0.05, thus proofing the statistical significance of the LSTM approach. It can be seen clearly from the accuracy achieved at each fold shown in the box plot in Fig. 8 that LSTM showed superior results compared to the other approaches, with a higher median and higher accuracy for all folds.

Table 3. Precision and recall of Decision tree, SVM, Random Forest, and LSTM on a test set of 308 sequences

Activity	SVM		Decision Tree		Random Forest		LSTM	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
1	0.75	0.75	0.79	0.79	0.86	0.86	1.00	0.96
2	0.95	0.75	0.95	0.75	0.93	0.89	0.96	0.96
3	0.96	0.96	0.57	0.61	0.82	0.82	0.93	1.00
4	0.50	0.50	0.52	0.54	0.61	0.79	0.97	1.00
5	0.63	0.43	0.83	0.71	0.92	0.82	0.96	0.96
6	0.47	0.32	0.68	0.89	0.79	0.93	0.87	0.96
7	0.39	0.79	0.78	0.75	0.81	0.79	0.89	0.89
8	0.43	0.43	0.61	0.50	0.72	0.46	1.00	0.82
9	0.57	0.46	0.75	0.64	0.83	0.71	1.00	1.00
10	0.81	0.61	0.57	0.75	0.73	0.86	0.96	0.96
11	0.54	0.68	0.93	0.89	0.79	0.82	0.96	0.96
Avg	**0.64**	**0.61**	**0.72**	**0.71**	**0.80**	**0.80**	**0.96**	**0.95**



(a) SVM

(b) Decision Tree

(c) Random Forest

(d) LSTM

Fig. 9. Confusion matrices showing the number of samples classified as belonging to each class

4 Conclusions

In this work, person independent and view-invariant activity recognition approach has been proposed. The OpenPose library was used to detect anatomical key points in the image sequences collected from the MHAD database. Afterward, temporal motion features were extracted from consecutive frames in each sequence. Lastly, different classifiers were applied to detect human activities. The classification accuracy of these models was compared to the proposed approach of using an LSTM-RNN. Our approach shows improved results compared to the conventional approaches, it is able to correctly classify activities performed by several different subjects and from various camera angles. Although OpenPose is able to detect several persons in a frame, our work will not be able to correctly classify the activity of several people at once. In future work, the proposed network would be implemented in a real-time system to detect different activities and gestures.

References

1. Berkeley multimodal human action database. http://tele-immersion.citris-uc.org/berkeley_mhad
2. Abidine, B.M., Fergani, L., Fergani, B., Oussalah, M.: The joint use of sequence features combination and modified weighted SVM for improving daily activity recognition. Pattern Anal. Appl. **21**(1), 119–138 (2018)
3. Adama, D.A., Lotfi, A., Langensiepen, C., Lee, K., Trindade, P.: Human activity learning for assistive robotics using a classifier ensemble. Soft Comput. **22**(21), 7027–7039 (2018)
4. Altun, K., Barshan, B., Tunçel, O.: Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recognit. **43**(10), 3605–3620 (2010)
5. Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., Sheikh, Y.: OpenPose: realtime multi-person 2D pose estimation using part affinity fields. arXiv preprint arXiv:1812.08008 (2018)
6. Cao, Z., Simon, T., Wei, S.-E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. arXiv preprint arXiv:1611.08050 (2016)
7. Cao, Z., Simon, T., Wei, S.-E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: CVPR (2017)
8. Ceron, J.D., Lopez, D.M., Ramirez, G.A.: A mobile system for sedentary behaviors classification based on accelerometer and location data. Comput. Ind. **92**, 25–31 (2017)
9. Eck, D., Schmidhuber, J.: Finding temporal structure in music: blues improvisation with LSTM recurrent networks. In: Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing, pp. 747–756. IEEE (2002)
10. Fujimori, Y., Ohmura, Y., Harada, T., Kuniyoshi, Y.: Wearable motion capture suit with full-body tactile sensors. In: IEEE International Conference on Robotics and Automation, ICRA 2009, pp. 3186–3193. IEEE (2009)
11. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: International Conference on Machine Learning, pp. 1764–1772 (2014)

12. Graves, A., Liwicki, M., Bunke, H., Schmidhuber, J., Fernández, S.: Unconstrained on-line handwriting recognition with recurrent neural networks. In: Advances in Neural Information Processing Systems, pp. 577–584 (2008)
13. Hassan, M.M., Huda, S., Uddin, M.Z., Almogren, A., Alrubaian, M.: Human activity recognition from body sensor data using deep learning. J. Med. Syst. **42**(6), 99 (2018)
14. Jalal, A., Uddin, M.Z., Kim, J.T., Kim, T.-S.: Recognition of human home activities via depth silhouettes and transformation for smart homes. Indoor Built Environ. **21**(1), 184–190 (2012)
15. Liu, L., Wu, X., Wu, L., Guo, T.: Static human gesture grading based on kinect. In: 2012 5th International Congress on Image and Signal Processing (CISP), pp. 1390–1393. IEEE (2012)
16. Margarito, J., Helaoui, R., Bianchi, A.M., Sartor, F., Bonomi, A.G., et al.: User-independent recognition of sports activities from a single wrist-worn accelerometer: a template-matching-based approach. IEEE Trans. Biomed. Eng. **63**(4), 788–796 (2016)
17. Noori, F.M., Garcia-Ceja, E., Uddin, M.Z., Riegler, M.: Fusion of multiple representations extracted from a single sensor's data for activity recognition using CNNs. In: International Joint Conference on Neural Networks (IJCNN). IEEE (2019)
18. Nweke, H.F., Teh, Y.W., Al-Garadi, M.A., Alo, U.R.: Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges. Expert Syst. Appl. **105**, 233–261 (2018)
19. Nweke, H.F., Teh, Y.W., Mujtaba, G., Al-garadi, M.A.: Data fusion and multiple classifier systems for human activity detection and health monitoring: review and open research directions. Inf. Fusion **46**, 147–170 (2019)
20. Qiao, S., Wang, Y., Li, J.: Real-time human gesture grading based on OpenPose. In: 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), pp. 1–6. IEEE (2017)
21. Simon, T., Joo, H., Matthews, I.A., Sheikh, Y.: Hand keypoint detection in single images using multiview bootstrapping. In: CVPR, vol. 1, p. 2 (2017)
22. Uddin, M.Z., Kim, D.-H., Kim, J.T., Kim, T.-S.: An indoor human activity recognition system for smart home using local binary pattern features with hidden markov models. Indoor Built Environ. **22**(1), 289–298 (2013)
23. Wei, S.-E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. In: CVPR (2016)
24. Yang, X., Zhang, C., Tian, Y.: Recognizing actions using depth motion maps-based histograms of oriented gradients. In: Proceedings of the 20th ACM International Conference on Multimedia, pp. 1057–1060. ACM (2012)