# Hierarchical long short-term memory for action recognition based on 3D skeleton joints from Kinect sensor

Nur Awal Hidayanto [a,1,*], Adhi Prahara [b,2], Riky Dwi Puriyanto [c,3]

[a,b] Informatics Department, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
[c] Electrical Engineering Department, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
[1] nur1500018089@webmail. uad.ac.id *; [2] adhi.prahara@tif.uad.ac.id; [3] rikydp@ee.uad.ac.id
* Corresponding Author

Received 11 August 2020; accepted 23 February 2021; published 24 February 2021

## ABSTRACT

Action recognition has been used in a wide range of applications such as human-computer interaction, intelligent video surveillance systems, video summarization, and robotics. Recognizing action is important for intelligent agents to understand, learn and interact with the environment. The recent technology that allows the acquisition of RGB+D and 3D skeleton data and a deep learning model's development significantly increases the action recognition model's performance. In this research, hierarchical Long Sort-Term Memory is proposed to recognize action based on 3D skeleton joints from Kinect sensor. The model uses the 3D axis of skeleton joints and groups each joint in the axis into parts, namely, spine, left and right arm, left and right hand, and left and right leg. To fit the hierarchically structured layers of LSTM, the parts are concatenated into spine, arms, hands, and legs and then concatenated into the body. The model crosses the body in each axis into a single final body and fed to the final layer to classify the action. The performance is measured using cross-view and cross-subject evaluation and achieves accuracy 0.854 and 0.837, respectively, from the 10 action classes of the NTU RGB+D dataset.

KEYWORDS
Action recognition,
hierarchical LSTM
Skeleton joints
Kinect

## 1. Introduction

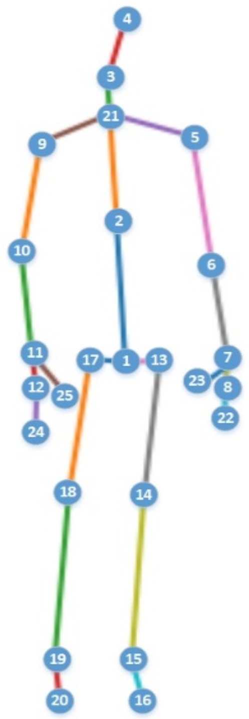
Computer vision is a study to model biological vision into artificial vision. Computer vision has two aims: to propose the computational models of human visual system (HVS) and build an autonomous system that performs the same tasks or even surpasses the human visual system [1]. One of the challenging tasks in the field of computer vision is action recognition. Action recognition has been used in a wide variety of applications such as human-computer interaction, an intelligent system in video surveillance, health care, robotics, video summarization and so on. Action recognition from video can be interpreted as recognizing human action using pattern recognition system automatically [2]. The system analyzes and learns the pattern of human action in the training phase and model the knowledge to classify similar action in the testing phase. Action recognition from video is not only analyzing the pattern of motion of the body but also describing the subject intention, emotion and thought [2]. It is important, especially in the development of intelligent robot/agent to perform actions by recognizing the movements and action of observed agent [3] then followed by more complex understanding such as observing the effects of action on the environment and to perform an action in order to change the environment. The human action can be observed based on full-body, part of the body, grammars, and scene approaches [3].

The challenges in action recognition such as the variety of movement to perform one action (every human executes slightly different movement to perform the same action), the appearance of action is different when observed in a different viewpoint (the camera viewpoint), occlusion, background clutter, and the variety of sequence movement speed when performing the same action [4]. Continuous action (performing one action after another action continuously) also increases the difficulty because the action did not start from the beginning but from the end of the previous action. With the recent development of technology, action recognition can be performed on RGB frames, RGB+D frames, skeleton joints, or

10.26555/jifo.v15i1.20106

their combination [4]. Recent review on action recognition classifies the dataset into single viewpoint which uses a single camera to record human action from a certain invariant angle without camera movement, multiple viewpoints which use multiple camera to observe the action from multiple view of the same human, and RGB+D dataset which uses RGB and Depth camera to observe the action and sometimes combined with the 3D skeleton data [2]. The three types of frame data can be acquired using the Kinect sensor that plays a significant role in developing action recognition models. Kinect able to recognize human body pattern in the form of skeleton joints. The skeleton joints are normally projected on the 3D space, but they can also be projected on the RGB frame or Depth frame coordinates. Kinect able to record up to six skeletons at a time, and each of the skeleton has 25 joints. The joints group arrangement used in this research is shown in Fig. 1.

Action recognition model has been proposed by many researchers [5], [6], [15]–[24], [7], [25]–[34], [8]–[14]. In the early action recognition research, the model uses motion and texture descriptors from the video frames to compute the Spatio-temporal interest point as features. The hand-crafted features then used in the training phase to classify the action. Although the model achieves good performance but it is difficult to be applied in the real-world problem [2]. Recently, deep learning-based approaches have become popular in action recognition tasks [2]. The model can generate a high-level representation of features automatically using Convolutional Neural Network (CNN), learn the pattern of the sequence of movement using Long Sort-Term Memory (LSTM), or use both of them. An excellent review of action recognition model using CNN has been made by [35]. Some of the action recognition researches will be explained in the next paragraph.



**The index of skeleton joints:**

1. Base of the spine
2. Middle of the spine
3. Neck
4. Head
5. Left shoulder
6. Left elbow
7. Left wrist
8. Left hand
9. Right shoulder
10. Right elbow
11. Right wrist
12. Right hand
13. Left hip
14. Left knee
15. Left ankle
16. Left foot
17. Right hip
18. Right knee
19. Right ankle
20. Right foot
21. Spine
22. Tip of the left hand
23. Left thumb
24. Tip of the right hand
25. Right thumb

**The joints group arrangement:**

| | |
|---|---|
| Spine | 1,2,3,4,21 |
| Right arm | 9,10,11 |
| Left-arm | 5,6,7 |
| Right hand | 12,24,25 |
| Left hand | 8,22,23 |
| Right leg | 17,18,19,20 |
| Left leg | 13,14,15,16 |

**Fig. 1.** The joints group arrangement of the Kinect sensor (The index arrangement taken from [36] and [37]).

Zhu *et al.* [13] state that skeleton joints serve a good representation for describing actions. They propose action recognition using end-to-end fully connected deep Long Short-Term Memory (LSTM) based on skeleton joints. The skeleton joints are taken as input at each time slot with a novel regularization and dropout scheme. The experiment on three datasets shows that the proposed method is effective in recognizing action. Zhu *et al.* [18] use a multimodal fusion from RGB frames and depth frames. The proposed method utilizes 3D Convolutional Neural Network followed by Convolutional LSTM on both of the frames then performs multimodal fusion to achieve the finals score. The fine-tuning is performed in the multimodal data to avoid overfitting. The result shows high accuracy on the SKIG dataset and 51.02% on IsoGD dataset. Du *et al.* [6] propose an end-to-end hierarchical recurrent neural network (RNN) on skeleton joints to perform action recognition. The skeleton joints are divided hierarchically into five parts according to the human physical structure and separated them into five subnets. The subnets hierarchically fused at each layer to the upper layer and fed into a single-layer perceptron to generate the decision. The result compared to the five others deep RNN architectures derived from the model and several other models on three public datasets shows promising state-of-the-art performance with high computational efficiency.

Li *et al.* [20] also use 3D skeleton data for action recognition because of its robustness, conciseness, and view-independent representation. They utilize SPF (spatial-domain-feature) as the input of the LSTM network and TPF (temporal-domain-feature) data as the CNN network's input. The score fusion from LSTM and CNN is used to perform effective recognition. The result on the NTU RGB+D dataset achieved 87.40% accuracy and ranked first in the Large Scale 3D Human Activity Analysis Challenge in Depth Video. Chai *et al.* [10] propose two streams RNN (2S-RNN) to handle the continuous gesture recognition problems on RGB-D data. The continuous gestures are segmented into separated gestures and each isolated gesture is recognized by 2S-RNN. The method is designed to fuse multimodal features such as RGB and depth channels efficiently. The result shows promising performance on the Continuous Gesture Dataset (ConGD) dataset and ranked first in the ChaLearn LAP Large-scale Continuous Gesture Recognition Challenge. In this research, hierarchical LSTM is proposed to recognize action and will be explained in detail in the next section, organized as follows. Section 2 presents the proposed hierarchical LSTM model. Section 3 presents the result and discussion and the conclusion of this work is described in Section 4.

## 2. Method

LSTM is a special type of Recurrent Neural Network (RNN). LSTM consists of cell, input gate, output gate, and forget gate. Cell is used to remember the information at each time and the other three gates have function to control the flow of information from the input, cell and output. LSTM is effective to save information based on the data in the sequence of time. LSTM model which uses memory cells to store information is better at finding and exploiting long range context. LSTM for action recognition based on 3D skeleton data have been proposed by researchers [6], [8], [23], [25], [28], [11]–[13], [16], [19]–[22]. In this research, 3D skeleton joints are used as the input data for hierarchical LSTM. Hierarchical LSTM [38] is proposed to capture the temporal dependencies by discovering the latent hierarchical structure of the sequence of 3D skeleton joints. This work inspired by [6] that utilizes hierarchical structure of 3D skeleton data to represent the human action.

### 2.1. The Dataset

In this research, we use the 3D skeletons data from NTU RGB+D dataset [36]. The dataset consists of 40 subjects performing 60 actions and taken from 3 cameras with 45°, 0°, and -45° arrangement. For a simple interaction purpose we only take 10 of 60 actions namely drink water (A1), stand up (A9), play with phone/tablet (A29), eat meal (A2), sit down (A8), hand waving (A23), kicking something (A24), phone call (A28), taking a selfie (A32), and cross hands in front action (A40).

The 3D skeletons data in each action consists of 25 joints per time step and every joint has 11 features namely color (x, y), depth (x, y), camera (x, y, z) and orientation (x, y, z, w). The color features are associated with the RGB frame, the depth features are associated with the Depth frame, the camera features are related with the 3D projection coordinates and the orientation features are the direction of the joints. From the 11 features, we only take the first three features namely joints position in the 3D space (x, y, z). From the joints position in the 3D space, the sequence of joints movement will form an action as illustrated in Fig. 2.

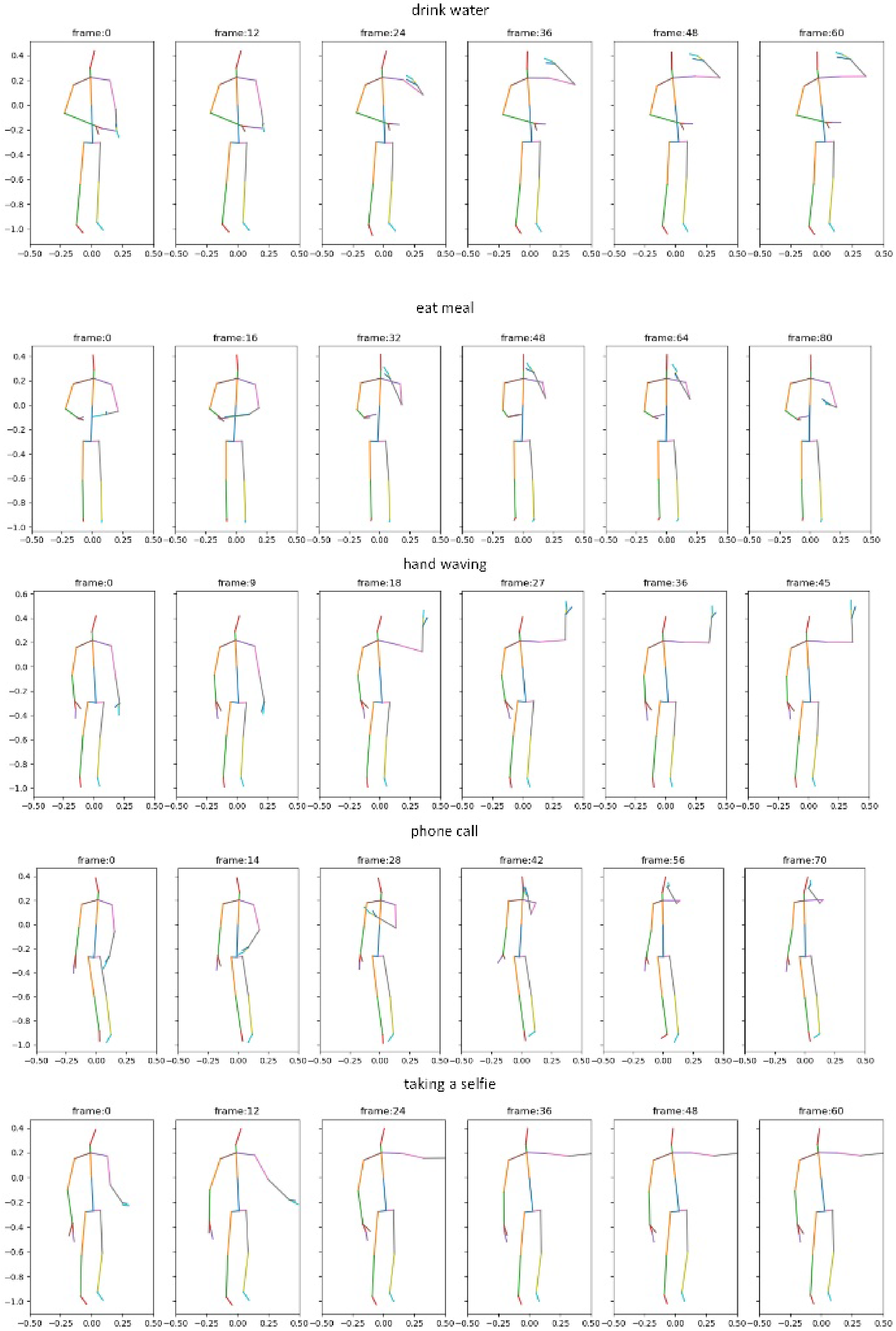


**Fig. 2.** Illustration of actions based on 3D skeleton.

### 2.2. Pre-Processing Data

The Kinect skeleton joints need to undergo some pre-processing procedures before passed into the hierarchical LSTM network. The pre-processing procedures proposed in this research such as:

- Feature dimension. The input of LSTM is composed in the form of data dimension (batch), the sequence of time (time steps) and the feature dimension at each time (feature). The features divided further into axis and joints because they still in the form of 25 skeleton joints in 3D field (x, y, and z axis). The final skeleton joints data in the format of batch, time, axis, and joints.
- Position normalization. The 3D skeleton joints from sensor Kinect are taken relative to the camera. Therefore, the position need to be normalized to the origin by translating the "base of the spine" to the origin (0,0,0) then followed by other joints using the same translation.
- Padding. The data length may be varied at each time. In order to fit the input, padding is used to fill the empty time slot before the starting point of real data sequence.

### 2.3. Hierarchical LSTM Architecture

The skeleton data have (*n*, 3, 25) dimension where the first dimension *n* is the time step that has been fitted through padding in the pre-processing step, the second and the third dimension are the three axes (x, y, z) and 25 skeleton joints respectively. Since LSTM receives input in the format of (samples, time steps, features), the skeleton data need to be reshaped to fit the input layer of LSTM. The axis dimension is split into three independent axes hence one axis will be represented by its 25 joints directly as features. The joints then combined into group according to the joints group arrangement in Fig. 1. The general architecture of the proposed hierarchical LSTM layers is shown in Fig. 3. From Fig. 3, per time step, the skeleton data are split into three independent axes and the 25 skeleton joints on each axis (x, y, z) are split into joints group arrangement that shown in Fig. 1 namely spine, right arm, left arm, right hand, left hand, right leg, and left leg. Every group will become the input of LSTM (32) which is LSTM with 32 units. LSTM (32) output is concatenated to form some pairings between the right and the left parts, namely right and left arm concatenated into arms, right and left hand concatenated into hands right and left leg concatenated into legs. The result of concatenation will be fed into LSTM (64). The output from the spine group and the outputs of LSTM (64) are concatenated together into body of each respective axis.

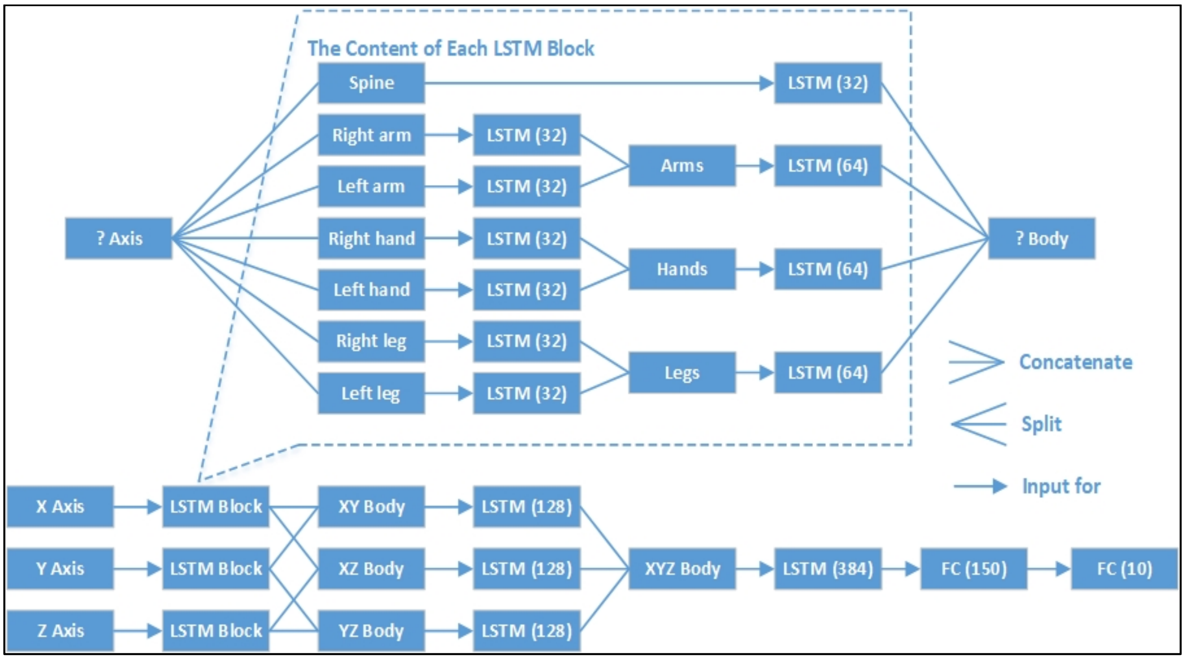


**Fig. 3.** The proposed hierarchical LSTM architecture for action recognition.

The body of each respective axis are crossed each other to form three pairs of concatenated body namely XY, XZ and YZ body. Each of them is fed into LSTM (128). The outputs of LSTM (128) are concatenated into single body namely XYZ body and fed into LSTM (384). Each concatenated output of the LSTM layers followed by dropout layers. Finally, the output of LSTM (384) become the input of the dense/fully connected layer with 150 neurons with *tanh* activation function and goes to the output layer with sigmoid activation function to classify the 10 categories of action.

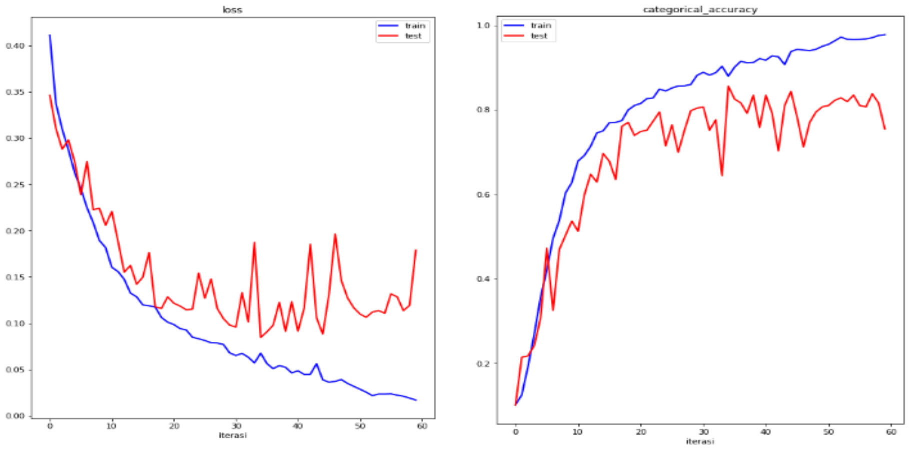
### 2.4. Evaluation Metrics

The performance is measured based on the accuracy of cross-subject and cross-view evaluation [36]. Cross-subject evaluation considers the difference of movement sequence between subjects while performing the same action. Cross-view evaluation considers the difference between viewpoint of the camera when capturing the action.

- Cross Subject. The training and test data are split from the 40 subjects evenly. The subject's ID for the training data are 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38 and the other subject's ID used as the test data. The performance evaluation used accuracy score calculated from the test data.
- Cross View. From the 3-views setup for the camera which is 45°, 0°, and -45°, the test data are taken from the camera 1 (45°) and the training data consist samples from the other cameras. The performance evaluation also used accuracy score calculated from the test data.

## 3. Results and Discussion

The proposed hierarchical LSTM for action recognition built using Python with Keras library and Kinect SDK 2.0. The method runs on Intel Xeon CPU @2.30GHz processor, 11GB of RAM, and NVidia Tesla T4/Tesla K80 from Google Colabs for training and testing. The proposed method is trained using Adagrad optimizer with learning rate 0.01, the batch size is 256, epoch is 1000, early stopping mode to monitor the loss, and using 0.05% of the training data as validation data. The proposed hierarchical LSTM model's performance is evaluated using cross-view and cross-subject evaluation as proposed in [36]. The evaluation is used to test the challenges in action recognition which are the same action can be performed differently by different human and the same action can be different when observed from different camera viewpoint. The cross-view training result is shown in Fig. 4 where Fig. 4(a) is the result of monitoring loss per epoch and Fig. 4(b) is result of monitoring accuracy per epoch. From the cross-view training result shown in Fig. 4, there is an indication of overfitting in the 20th epoch where the training loss is steadily decreasing, but the validation loss is continuously rising and the accuracy of training is steadily increasing the validation accuracy is falling.



(a) (b)

**Fig. 4.** The training loss (a) and accuracy (b) of the cross-view evaluation.

The early stopping mode caught this sign and stopped the training to return the best model before overfitting. The evaluation result is shown in Fig. 5 with accuracy score 0.854. The false prediction mainly caused by the similar action such as eat meal predicted as drink water, phone call predicted as drink water, and also taking a selfie predicted as hand waving. The good result with categorical accuracy score above 0.90 shown by stand up, sit down, and kicking action because the actions provide less similarity with the other action.

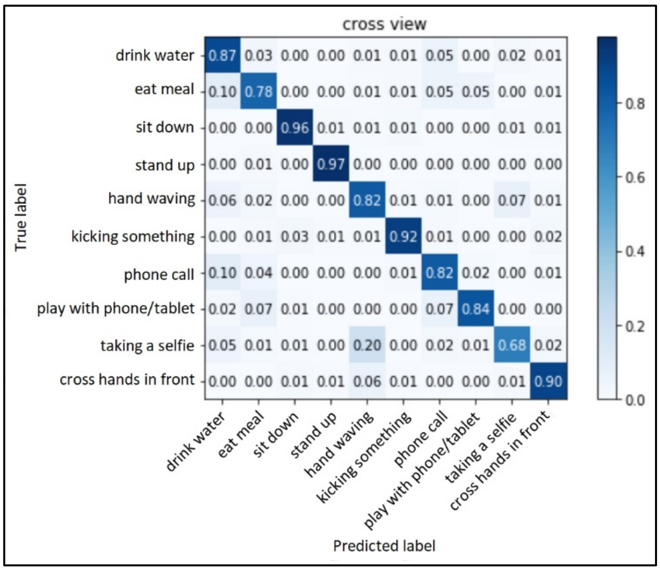
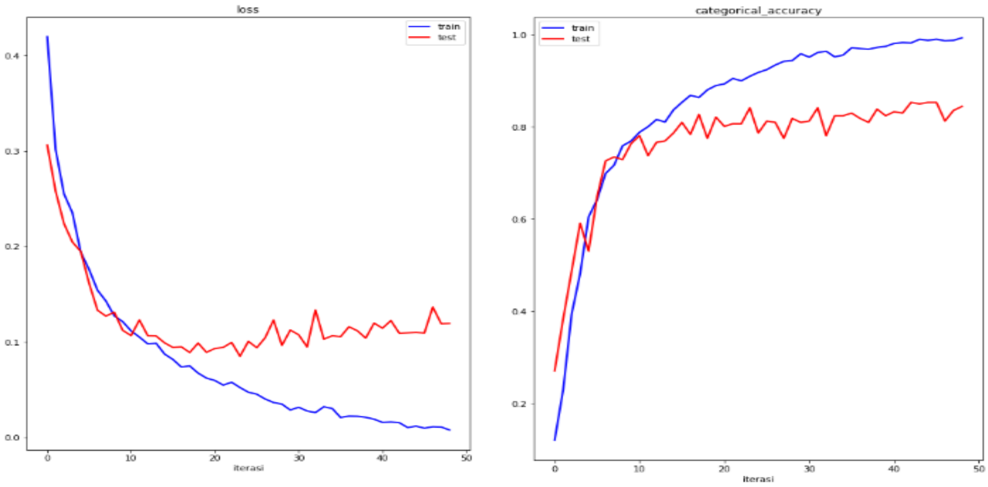


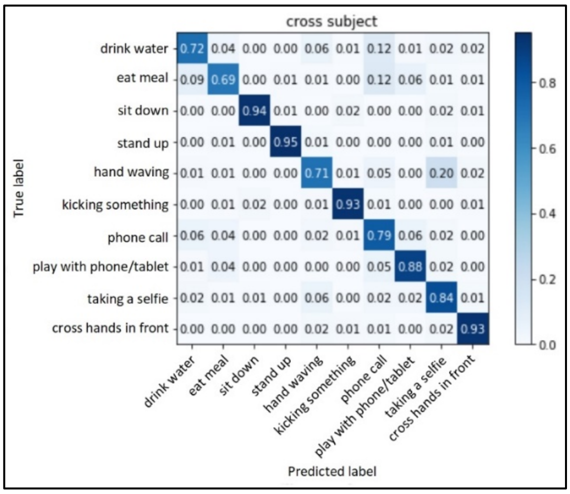
**Fig. 5.** The cross-view evaluation result.

The second evaluation is cross-subject evaluation. The cross-subject training result is shown in Fig. 6 where Fig. 6(a) is the result of monitoring loss per epoch and Fig. 6(b) is result of monitoring accuracy per epoch. From the cross-subject training result shown in Fig. 6, there is an indication of overfitting in the 20th epoch. The situation similar to the previous cross-view evaluation where the loss of training is steadily decrease but the validation loss is continuously rising and the accuracy of training is steadily increase but the validation accuracy is falling hence the situation triggers early stopping condition. The evaluation result is shown in Fig. 7 with accuracy score 0.837. The false prediction also similar to the cross-view evaluation result that mainly caused by the similar action such as eat meal and drink water predicted as phone call and also hand waving predicted as taking a selfie. The model also has confidence with accuracy score above 0.90 when recognizing category such as stand up, sit down, kicking something, and cross hands in front action because the actions provide less similarity with the other action.



(a) (b)

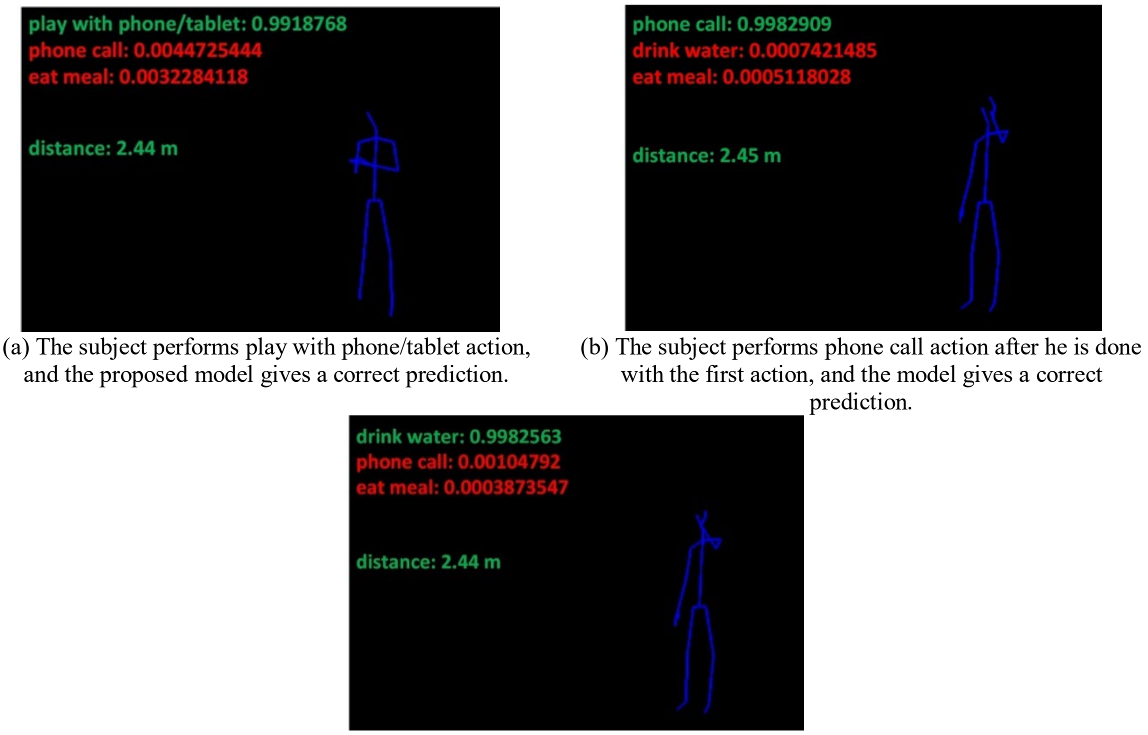
**Fig. 6.** The training loss (a) and accuracy (b) of the cross-subject evaluation.



| | drink water | eat meal | sit down | stand up | hand waving | kicking something | phone call | play with phone/tablet | taking a selfie | cross hands in front |
|---|---|---|---|---|---|---|---|---|---|---|
| drink water | 0.72 | 0.04 | 0.00 | 0.00 | 0.06 | 0.01 | 0.12 | 0.01 | 0.02 | 0.02 |
| eat meal | 0.09 | 0.69 | 0.00 | 0.01 | 0.01 | 0.00 | 0.12 | 0.06 | 0.01 | 0.01 |
| sit down | 0.00 | 0.00 | 0.94 | 0.01 | 0.00 | 0.02 | 0.00 | 0.00 | 0.02 | 0.01 |
| stand up | 0.00 | 0.01 | 0.00 | 0.95 | 0.01 | 0.00 | 0.00 | 0.00 | 0.01 | 0.00 |
| hand waving | 0.01 | 0.01 | 0.00 | 0.00 | 0.71 | 0.01 | 0.05 | 0.00 | 0.20 | 0.02 |
| kicking something | 0.00 | 0.01 | 0.02 | 0.00 | 0.01 | 0.93 | 0.01 | 0.00 | 0.00 | 0.01 |
| phone call | 0.06 | 0.04 | 0.00 | 0.00 | 0.02 | 0.01 | 0.79 | 0.06 | 0.02 | 0.00 |
| play with phone/tablet | 0.01 | 0.04 | 0.00 | 0.00 | 0.00 | 0.00 | 0.05 | 0.88 | 0.02 | 0.00 |
| taking a selfie | 0.02 | 0.01 | 0.01 | 0.00 | 0.06 | 0.00 | 0.02 | 0.02 | 0.84 | 0.01 |
| cross hands in front | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.01 | 0.01 | 0.00 | 0.02 | 0.93 |

**Fig. 7.** The cross-subject evaluation result.

To test the model in real-time action, we conduct a scenario where the subject plays with his phone then after sometime the subject begin receiving a phone call; hence the action change from play with phone/tablet to phone call. Fig. 8 shows the result of action recognition using the above scenario where the prediction result of play with phone/tablet and phone call action is shown in Fig. 8(a) and Fig. 8(b), and the wrong prediction of phone call action is shown in Fig. 8(c). The top three confidence scores and the distance between the subject and the camera are displayed to observe the prediction result. From Fig. 8(a), the model predicted that the subject is playing with his/her phone/tablet with a confidence score of 0.9918768 then recognized that the subject is doing a phone call with a confidence score of 0.9982909 as shown in Fig. 8(b). However, the model gets the wrong prediction after recognizing a phone call. The confidence score of phone call decrease and drink water prediction score raises. This is wrong because the subject still performing a phone call action.



play with phone/tablet: 0.9918768
phone call: 0.0044725444
eat meal: 0.0032284118

distance: 2.44 m

(a) The subject performs play with phone/tablet action, and the proposed model gives a correct prediction.

phone call: 0.9982909
drink water: 0.0007421485
eat meal: 0.0005118028

distance: 2.45 m

(b) The subject performs phone call action after he is done with the first action, and the model gives a correct prediction.

drink water: 0.9982563
phone call: 0.00104792
eat meal: 0.0003873547

distance: 2.44 m

(c) The subject still performing phone call action for a while but the model confidence score of phone call dropped and the drink water confidence score raised thus gives wrong prediction.

**Fig. 8.** A scenario to test the proposed model.

## 4. Conclusion

Action recognition is an important task in computer vision that triggers intelligent agents or robots to perform interaction with humans. In this work, hierarchical LSTM is proposed to recognize action based on 3D skeleton joints. The evaluation result shows that the proposed model achieves 0.854 in the cross-view evaluation and achieves 0.837 in cross-subject evaluation. The good result indicates that the model can capture the hierarchical structure of motion from 3D skeleton data through a hierarchical LSTM model. For future work, the model can be tuned to handle continuous action efficiently and combined with the RGB+D data to increase the performance.

## Acknowledgment

The portions of this paper's research used the NTU RGB+D Action Recognition Dataset made available by the ROSE Lab at the Nanyang Technological University, Singapore. N. A. Hidayanto thanks the ROSE Lab at the Nanyang Technological University that permit access to the dataset.

## Declarations

**Author contribution.** All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

**Funding statement.** This work was supported by LPPM Universitas Ahmad Dahlan under Grant PF-135/SP3/LPPM-UAD/IV/2019.

**Conflict of interest.** The authors declare no conflict of interest.

**Additional information.** No additional information is available for this paper.

## References

[1] T. Huang, "Computer vision: Evolution and promise," 1996. Available at: Google Scholar

[2] D. Wu, N. Sharma, and M. Blumenstein, "Recent advances in video-based human action recognition using deep learning: A review," in *Proceedings of the International Joint Conference on Neural Networks*, 2017, vol. 2017-May, pp. 2865–2872, doi: 10.1109/IJCNN.2017.7966210.

[3] V. Krüger, D. Kragic, A. Ude, and C. Geib, "The meaning of action: A review on action recognition and mapping," *Adv. Robot.*, vol. 21, no. 13, pp. 1473–1501, 2007, doi: 10.1163/156855307782148578.

[4] C. Chen, R. Jafari, and N. Kehtarnavaz, "A survey of depth and inertial sensor fusion for human action recognition," *Multimed. Tools Appl.*, vol. 76, no. 3, pp. 4405–4425, Feb. 2017, doi: 10.1007/s11042-015-3177-1.

[5] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, pp. 588–595, doi: 10.1109/CVPR.2014.82.

[6] Yong Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1110–1118, doi: 10.1109/CVPR.2015.7298714.

[7] H. Wang, W. Wang, and L. Wang, "Hierarchical motion evolution for action recognition," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015, pp. 574–578, doi: 10.1109/ACPR.2015.7486568.

[8] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition," Springer, Cham, 2016, pp. 816–833. doi: 10.1007/978-3-319-46487-9_50

[9] Y. Wang, S. Wang, J. Tang, N. O'Hare, Y. Chang, and B. Li, "Hierarchical Attention Network for Action Recognition in Videos," Jul. 2016. Available at: Google Scholar

[10] X. Chai, Z. Liu, F. Yin, Z. Liu, and X. Chen, "Two streams Recurrent Neural Networks for Large-Scale Continuous Gesture Recognition," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016, pp. 31–36, doi: 10.1109/ICPR.2016.7899603.

[11] Y. Du, Y. Fu, and L. Wang, "Representation Learning of Temporal Dynamics for Skeleton-Based Action Recognition," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3010–3022, Jul. 2016, doi: 10.1109/TIP.2016.2552404.

[12] B. Mahasseni and S. Todorovic, "Regularizing Long Short Term Memory with 3D Human-Skeleton Sequences for Action Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3054–3062, doi: 10.1109/CVPR.2016.333.

[13] W. Zhu *et al.*, "Co-Occurrence Feature Learning for Skeleton Based Action Recognition Using Regularized Deep LSTM Networks." 2016. Available at: Google Scholar

[14] R. Vemulapalli and R. Chellappa, "Rolling rotations for recognizing human actions from 3D skeletal data," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 4471–4479, doi: 10.1109/CVPR.2016.484.

[15] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A New Representation of Skeleton Sequences for 3D Action Recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4570–4579, doi: 10.1109/CVPR.2017.486.

[16] S. Zhang, X. Liu, and J. Xiao, "On Geometric Features for Skeleton-Based Action Recognition Using Multilayer LSTM Networks," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 148–157, doi: 10.1109/WACV.2017.24.

[17] T. Tsunoda, Y. Komori, M. Matsugu, and T. Harada, "Football Action Recognition Using Hierarchical LSTM," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 155–163, doi: 10.1109/CVPRW.2017.25.

[18] G. Zhu, L. Zhang, P. Shen, and J. Song, "Multimodal Gesture Recognition Using 3-D Convolution and Convolutional LSTM," *IEEE Access*, vol. 5, pp. 4517–4524, 2017, doi: 10.1109/ACCESS.2017.2684186.

[19] I. Lee, D. Kim, S. Kang, and S. Lee, "Ensemble Deep Learning for Skeleton-Based Action Recognition Using Temporal Sliding LSTM Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1012–1020, doi: 10.1109/ICCV.2017.115.

[20] Chuankun Li, Pichao Wang, Shuang Wang, Yonghong Hou, and Wanqing Li, "Skeleton-based action recognition using LSTM and CNN," in *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2017, pp. 585–590, doi: 10.1109/ICMEW.2017.8026287.

[21] S. Wei, Y. Song, and Y. Zhang, "Human skeleton tree recurrent neural network with joint relative motion feature for skeleton based action recognition," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 91–95, doi: 10.1109/ICIP.2017.8296249.

[22] P. Shukla, K. K. Biswas, and P. K. Kalra, "Recurrent Neural Network Based Action Recognition from 3D Skeleton Data," in *2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2017, pp. 339–345, doi: 10.1109/SITIS.2017.63.

[23] W. Li, L. Wen, M.-C. Chang, S. N. Lim, and S. Lyu, "Adaptive RNN Tree for Large-Scale Human Action Recognition," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1453–1461, doi: 10.1109/ICCV.2017.161.

[24] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Two Stream LSTM: A Deep Fusion Framework for Human Action Recognition," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 177–186, doi: 10.1109/WACV.2017.27.

[25] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global Context-Aware Attention LSTM Networks for 3D Action Recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3671–3680, doi: 10.1109/CVPR.2017.391.

[26] D. Xu, X. Xiao, X. Wang, and J. Wang, "Human action recognition based on Kinect and PSO-SVM by representing 3D skeletons as points in lie group," in *ICALIP 2016 - 2016 International Conference on Audio, Language and Image Processing - Proceedings*, 2017, pp. 568–573, doi: 10.1109/ICALIP.2016.7846646.

[27] W. Ding, K. Liu, B. Xu, and F. Cheng, "Skeleton-based human action recognition via screw matrices," *Chinese J. Electron.*, vol. 26, no. 4, pp. 790–796, Jul. 2017, doi: 10.1049/cje.2017.06.012.

[28] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, "Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 3007–3021, Dec. 2018, doi: 10.1109/TPAMI.2017.2771306.

[29] S. Yan, J. S. Smith, W. Lu, and B. Zhang, "Hierarchical Multi-scale Attention Networks for action recognition," *Signal Process. Image Commun.*, vol. 61, pp. 73–84, Feb. 2018, doi: 10.1016/J.IMAGE.2017.11.005.

[30] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action Recognition in Video Sequences using Deep Bi-Directional LSTM With CNN Features," *IEEE Access*, vol. 6, pp. 1155–1166, 2018, doi: 10.1109/ACCESS.2017.2778011.

[31] X. Liu, Y. Li, and Q. Wang, "Multi-View Hierarchical Bidirectional Recurrent Neural Network for Depth Video Sequence Based Action Recognition," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 10, p. 1850033, Oct. 2018, doi: 10.1142/S0218001418500337.

[32] C. Li, Q. Zhong, D. Xie, and S. Pu, "Co-occurrence Feature Learning from Skeleton Data for Action Recognition and Detection with Hierarchical Aggregation," Apr. 2018. doi: 10.24963/ijcai.2018/109

[33] L. Wang, Y. Xu, J. Cheng, H. Xia, J. Yin, and J. Wu, "Human Action Recognition by Learning Spatio-Temporal Features With Deep Neural Networks," *IEEE Access*, vol. 6, pp. 17913–17922, 2018, doi: 10.1109/ACCESS.2018.2817253.

[34] Y. Han, S. L. Chung, A. Ambikapathi, J. S. Chan, W. Y. Lin, and S. F. Su, "Robust Human Action Recognition Using Global Spatial-Temporal Attention for Human Skeleton Data," in *Proceedings of the International Joint Conference on Neural Networks*, 2018, vol. 2018-July, doi: 10.1109/IJCNN.2018.8489386.

[35] G. Yao, T. Lei, and J. Zhong, "A review of Convolutional-Neural-Network-based action recognition," *Pattern Recognit. Lett.*, vol. 118, pp. 14–22, Feb. 2019, doi: 10.1016/j.patrec.2018.05.018.

[36] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. doi: 10.1109/CVPR.2016.115

[37] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019, doi: 10.1109/TPAMI.2019.2916873.

[38] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical Multiscale Recurrent Neural Networks," Sep. 2016. Available at: Google Scholar