International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V70I2P103

Volume 70 Issue 2, 18-21, February 2022

Original Article

Skeleton Based Human Action Recognition Using Doubly Linked List

Muhammad Sajid Khan[1, 2], Andrew Ware[2], Usman Habib[3], Muhammad Junaid Khalid[4], Nisar Bahoo[5]

[1]*Assistant Professor, Department of Computer Science, MY University, Islamabad, Pakistan*
[2] *Professor, Wales Institute of Digital Information, University of South Wales, United Kingdom*
[3]*Assistant Professor, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan*
[4]*Student, Institute of Information Technology, PMAS Arid Agriculture University, Rawalpindi, Pakistan*
[5]*Student, University of Engineering and Technology, Taxila, Pakistan*

Received Date: 08 January 2022
Revised Date: 11 February 2022
Accepted Date: 23 February 2022

Abstract *- Human Action Recognition is a significant focus for research because of its many applications in robotics and automation. This paper demonstrates how doubly linked lists can be used to sequence the 3D human actions recorded as video clips in the NTU RGBD 60 dataset. The nodes and edges in the list represent the joints and bone structure in the human skeleton. Each node holds information about the joint's position within the skeleton and pointers to its parent and child nodes. The doubly link list is constructed by first utilising the nodes representing the torso joints and then adding the nodes for the limbs' joints. The chosen sequence of nodes preserves the structural shape of the skeleton. The linked lists for many known activities are used as the training set for a classifier capable of identifying subsequent human actions. The classifier is based on the displacement between consecutive nodes in the action sequence. This approach minimises the complexity of the tree structure and improves the accuracy of 3D action recognition.*

Keywords — *Skeleton based action recognition, Human Action recognition, Video Processing, Doubly linked list.*

I. INTRODUCTION

Human action recognition (HAR), a complex machine vision problem, has various human-computer interactions (HCI), automation, and computer vision applications [1][2].

A critical driving force in the development of HAR is to enable a computer to perceive the world and the human activity within it. HAR research has a long track record [3], but there are still significant issues to be addressed in realising its full potency.

This work was supported by Wales Institute of Digital Information, University of South Wales, United Kingdom.

Despite the diversity in its application, HAR involves several common steps, including detecting human movement using raw sensor data and extracting these movements' characteristics to classify the performed actions. Researchers have developed various algorithms to accomplish these tasks, with each algorithm having its strengths and weaknesses.

Some algorithms are based on neural networks, while others use acyclic graphs and trees in the classification process. Neural networks have comparatively higher training time but better accuracy [4].

This paper presents an algorithm that uses a series of linked lists. Each linked list in the series holds static and dynamic information about a human's position at a point during the performance of a given action. Each linked sequence is fed into a neural network that learns and identifies the human action performed. The use of linked lists minimises the complexity of the system design and reduces the processing overhead.

After training the neural network using the NTU RGBD 60 dataset, the system's recognition ability was tested using a series of linked lists generated from data obtained using a Microsoft Kinect.

II. PREVIOUS WORK

Conventional HAR techniques use handcrafted features that often provide insufficient accuracy [5], [6]. Moreover, while the latest deep learning approaches have improved accuracy and performance, they require long training. These deep learning approaches typically make use of one of three frameworks: these being, image-based, graph-based, and sequence-based. Techniques based on the image-based framework tend to result in low accuracy.

For example, in testing, the TSRJI algorithm [7], [8] which uses a tree representation of the skeleton as the input to a Convolutional Neural Network (CNN), produced results with just 67.9% accuracy. The graph-based methods have proven themselves to have excellent performance and accuracy. For example, the Directed Graph Neural Network (DGNN) [9] utilises the human body's natural structure in the form of a directed acyclic graph, as shown in Fig.1.

The referenced work uses four consecutive neural networks, with the output of the first providing the input for the second, etcetera. The paper reported a relatively quick training time and an accuracy of 96.1%.Sequence-based methods have also resulted in good accuracy and performance. For example, the Structured Tree Neural Network [10] builds the tree sothat the joint's natural properties are preserved, as shown in Fig.2.

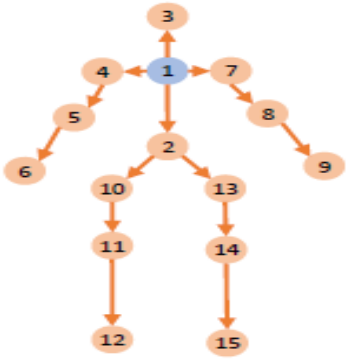

Fig. 1 Directed acyclic graph used in DGNN

This neural network has four different layers, where each layer's input is the output of the previous layer. This layered approach and simple tree structure help the neural networks learn the features quickly and accurately (96.3% were reported).

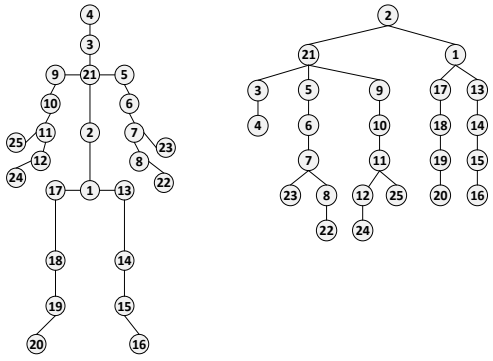

Fig. 2 Tree-based on skeletal information

III. PROPOSED METHOD

This paper proposes a sequence-based method that uses a Doubly Linked List to represent the skeleton. Initially, a hash table representing each element of the doubly linked list is constructed. The table comprises the central joints or skeleton, i.e., head, nose, neck, back and spine joints (note that the neck and spine joints appear twice). The linked list elements representing the arms are attached to the neck element. Similarly, the linked list elements representing the legs are connected to the spine joint element. The joints of the backbone are also interconnected. The node of the linked list represents the joints of the arms and legs. Each node (whether in the linked list or hash table) holds information about its position and connected nodes, as shown in Fig. 3. The node numbers and names for the skeletal structure are given in table 1.

Table 1. The node numbers and names for the skeletal structure of the human body

Node #	Node Name	Node #	Node Name
1	Spine	14	Right knee
2	Back	15	Right foot joint
3	Nose	16	Right toe
4	Head	17	Left thigh
5	Right shoulder	18	Left knee
6	Right elbow	19	Left foot joint
7	Righthand joint	20	Left toe
8	Righthand finger joint	21	Neck
9	Left shoulder	22	Righthand fingertip
10	Left elbow	23	Right thumb
11	Left hand joint	24	Lefthand fingertip
12	Lefthand finger joint	25	Left thumb

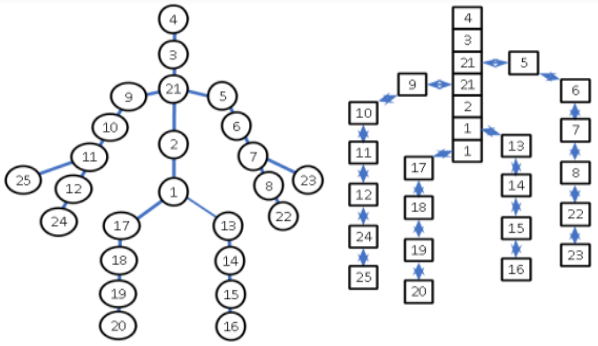

Fig. 3 Doubly linked list based on skeleton structure

The data in the nodes are used to store the spacio-temporal information about the skeleton. This spacio-temporal information is then fed to a global pooling layer for aggregation before the classification by the SoftMax layer.

The input to the classification process is the linked list representation of the skeleton. The block diagram, Fig.4, illustrates the flow of the classification process.

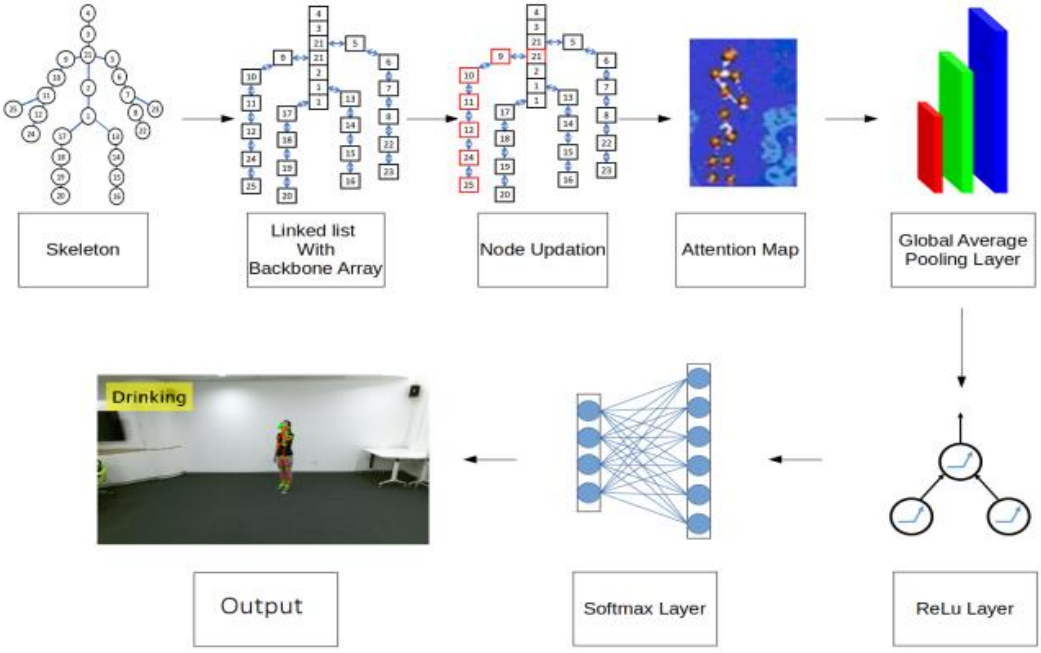

Fig. 4 Use of the doubly linked list for classification

A. Data sets and training features

The primary reason for using the doubly linked list is its simplicity and ability to preserve all the required information. The data used to build the model are from the NTU RGBD 60[5] dataset. The dataset contains 60 action classes and 56,880 video samples. The dataset comprises RGB videos, depth map sequences, 3D skeletal data, and infrared (IR) videos for each sample. The dataset was captured using three Kinect V2 cameras concurrently. The RGB videos' resolutions are 1920x1080, depth maps and IR videos are 512x424, and 3D skeletal data contains the 3D coordinates of 25 body joints at each frame. This large variety of samples helps the model achieve high accuracy. A full description and download of the dataset are freely available online.

B. Joint-Displacement detection

Joint-Displacement refers to the distance travelled by the joint between successive frames. This is calculated as:

$$Displacement = Position_{currentframe} - Position_{nextframe}$$

Knowing Joint-Displacement helps determine the action being performed.

C. Speed detection

Speed, a measure of how quickly joints have moved, is defined as the rate of change of displacement. This is calculated as:

$$Speed = \frac{Displacement}{Time}$$

Where "Time" is the inverse of the frame rate, for example, if the video operates at 30 fps, the time between the two frames is 0.03333333 seconds.

D. Change in angle

This feature helps determine the direction of movement of the joint. This knowledge is crucial because accurate recognition depends on the joint motion. Motion is calculated using the difference between one node and the next. Their positions are used to find the slope of the bones and then, using$\tan^{-1}$. The angle between the incoming and outgoing bone. The joint's direction in the subsequent frame is used to determine the change in the angle.

IV. NEURAL NETWORK

The Neural Network employed has four layers: an input layer, upper hidden layer, lower hidden layer, and output layer. The input layer's task is to make the Skelton Based Doubly Linked List (SBDLL) from the skeleton and update the node with new data about the movements. The input and upper hidden layers start working simultaneously, with the upper hidden layer responsible for extracting the movements' features. These layers save the values of the features for each node frame-by-frame. The lower hidden layer starts processing as soon as the upper hidden layer begins saving the feature values. The lower layer aggregates the features and filters the nodes with redundant features. This filtering of nodes helps reduce the processing time of the output layer.

The output layer begins processing after the lower hidden layer has finished. The function of the output layer is to determine and predict which action is being performed. This layer uses a convolution neural network, a ReLu and SoftMax classifier to do its job.

V. TESTING

The experiments were carried out on an i5 system with Windows 10 and 16GB of RAM, utilising almost 100GB of secondary memory. The NTU-RGBD 60 dataset was used to train and test the system. Training took 144 hours using 70% of the dataset (almost 40,000 videos) and tested the remaining 16,880 videos. The linked list methodology achieved a marked improvement over previous techniques, with an overall accuracy of 97.8% and a reduced training time.

Table 2. A COMPARISON WITH METHODOLOGIES

Algorithm	Accuracy CS	Accuracy CV
DGNN[9]	89.9%	96.1%
STNN[10]	90.2%	96.3%
SBDLL(Proposed Approach)	91.5%	97.8%

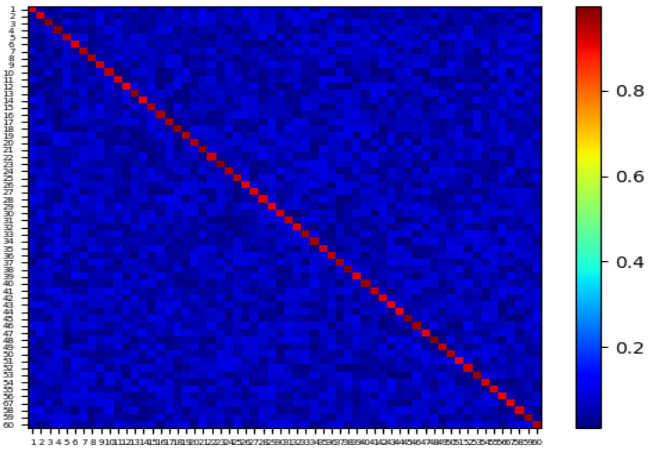

Fig. 5 Confusion matrix of observation and prediction

VI. CONCLUSION

This paper demonstrates how a Skeleton Based Doubly Linked List (SCDLL) can reduce the complexity and improve the performance of a neural network-based HAR system. The system described produces good results and reduced training time frames.

ACKNOWLEDGEMENT

Portions of this paper's research used the NTURGB+D Action Recognition Dataset made available by the ROSE Lab, Nanyang Technological University, Singapore.

REFERENCES

[1] Dasgupta, Poorna Banerjee. Compressed Representation of Color Information for Converting 2D Images Into 3D Models ., International Journal of Computer Trends and Technology, 68(11) (2020) 59-63.

[2] Snehal Shah, Kishan PS, Jitendra Jaiswal., Implementation of Python Packages For Image Recognition International Journal of Computer Trends and Technology, 69(11) (2021) 6-10.

[3] Y. Kuniyoshi, H. Inoue, and M. Inaba, Design and implementation of a system that generates assembly programs from visual recognition of human action sequences, in IEEE International Workshop on Intelligent Robots and Systems, Towards a New Frontier of Applications, 2 (1990) . 567–574 . doi: 10.1109/IROS.1990.262444.

[4] C. Robert, C. Guilpin, and A. Limoge, Comparison between conventional and neural network classifiers for rat sleep-wake stage discrimination, Neuropsychobiology, 35(4) (1997) 221–225. doi: 10.1159/000119348.

[5] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, (2020) 1010–1019. Accessed: Sep. 23, 2020. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2016/html/Shahroudy_NTU_RGBD_A_CVPR_2016_paper.html.

[6] 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 Conference on Computer Vision and Pattern Recognition, (2014) 588–595. Accessed: Sep. 23, 2020. [Online].Available:https://www.cvfoundation.org/openaccess/content_cvpr_2014/html/Vemulapalli_Human_Action_Recognition_2014_CVPR_paper.html.

[7] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars, Modeling Video Evolution for Action Recognition, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2015) 5378–5387. Accessed: Sep. 23, 2020. [Online].Available:https://www.cvfoundation.org/openaccess/content_cvpr_2015/html/Fernando_Modeling_Video_Evolution_2015_CVPR_paper.html.

[8] C. Caetano, F. Brémond, and W. R. Schwartz, Skeleton Image Representation for 3D Action Recognition Based on Tree Structure and Reference Joints, in 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), (2019) 16–23. doi: 10.1109/SIBGRAPI.2019.00011.

[9] L. Shi, Y. Zhang, J. Cheng, and H. Lu, Skeleton-Based Action Recognition with Directed Graph Neural Networks, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (2019) 7912–7921, Accessed: Sep. 23, 2020. [Online].Available:https://openaccess.thecvf.com/content_CVPR_2019/html/Shi_SkeletonBased_Action_Recognition_With_Directed_Graph_Neural_Networks_CVPR_2019_paper.html.

[10] M. S. Khan, A. Ware, M. Karim, N. Bahoo, and M. J. Khalid, Skeleton based Human Action Recognition using a Structured-Tree Neural Network, Eur. J. Eng. Res. Sci., 5(8) (2020) . doi: 10.24018/ejers.2020.5.8.2004.