Deriving Human motion recognition Using LSTM RNN

Mihir Shah
Lakehead University - Computer Science
email: mshah15@lakeheadu.ca

Abstract— **Quantum computing breakthroughs have changed the world into a place where objects can be discovered, the surrounding can be memorized, and future scenarios can be predicted. It is conceivable to anticipate the introduction of embedded technology has revolutionized the concept of as a result, the cost of detection systems has reduced. The enormous development and modernizing in the field of computer vision (CV), deep learning (DL), machine learning (ML), and artificial intelligence (AI) for recognizing moving objects. The most significant achievements in this area are human experimentation and study, that were at the heart of modern current research. This paper presented by use of a deep neural network in collaboration with a long short-term memory recurrent neural network for video processing. The convolution analytic technique the input stimuli into useful spatial objects. The extracted features were placed in the long short-term module to provide temporal characteristics. The feature maps including its long short-term memory component were incorporated into the framework. There were several methodologies and investigations developed in this field of the motion recognition. This current research will cover the LSTM and RNN-LSTM techniques for detecting human motion using Human Activity Recognition datasets for study of human activities recognition and, eventually, comparing the different algorithms for activity recognition.**

Index Terms— **onvolution Neural Network (CNN), Motion recognition, Recurrent Neural Network (RNN), Artificial intelligence (AI), Long Short-Term Memory (LSTM)onvolution Neural Network (CNN), Motion recognition, Recurrent Neural Network (RNN), Artificial intelligence (AI), Long Short-Term Memory (LSTM)C**

I. Introduction

Development and increase in technology also leads to the detection and the observation of any small item.Also in offices and for the safety of the children it is beneficial in the way of surveillance. For that Human activity recognition is becoming increasingly important in people's daily lives as technology gains more advanced knowledge about human behaviours from raw sensor data. With the rise of human-computer interaction applications, HAR technology has become a popular study topic both in the United States and abroad. So for this in this paper I had constructed one model for to analysis the six activities of the human for the detection with the help of dataset of HAR from the internet. An Establishment of method for analyzing data intelligence using pattern or non-model-based approaches, then track the picture using single or multi-perspective perspectives, then compare the image to the original. The goal is to predict one of the six activities(walking,walking upstairs, walking downstairs, laying down, sitting down and standing up). For precise measurements and behavior categorization, the LSTM and RNN LSTM methods were used to study this dataset. Traditional Machine Learning models will be used on these 561 sized domain expert specified features [2]. This research involves a 2D pose time series dataset and an LSTM RNN to classify behavioral patterns, with the goal of demonstrating that a sequence of 2D poses, rather than 3D poses or raw 2D pictures, may provide an accurate assessment of a person's or animal's activities.

This is an approach toward developing a system for classifying and predicting an animal's present action state. The feature maps of the long short-term memory component subsequently supplied to the demand software element. It maintains the absolutely essential information features in the video frame. The activities are detected using the soft - max module based on informational attributes. The feature maps of the long short-term memory component subsequently supplied to the demand software element. It maintains the absolutely essential information features in the video frame. The activities are detected using the soft - max module based on informational attributes.

The results of the experiments showed that the proposed model performed better in terms of accuracy. One of the numerous methods of activity identification is the use of mobile sensors. In these techniques, data from a variety of behaviors is captured utilizing a collection of particular body-worn motion sensors, such as accelerators, gyroscopes, and magnetic sensors.Acceleration and angular velocity data would be useful. This is in contrast to fixed sensor-based approaches [3]. Margarito et al. [4] connected accelerates to people's wrists to collect acceleration data, then used a template matching algorithm to identify eight common sporting activities. [5] developed a smart life support system (SAIL) for the aged and disabled.Motion recorded data (mocap) is one asset that has piqued the creative industry's curiosity by Menache, 2011, 2015 Delbridge. Consider the following scenario: instead of having the motion capture crew record a new sequence in the mocap lab, an animator has to animate a sequence where a figure walks limping from its leftleg due to an injury. The goal of this research is to figure out how to automate the motion capture tagging procedure. To tag a piece of material means to categorise it according to a certain ontology.

After stabilizing the model, the accuracy results of all methods were compared on both datasets using the LSTM-RNN approach for the activities. In previous study, LSTM was performed. The goal of this study is to deploy a RNN-LSTM to identify activities in this dataset and increase accuracy.

A. Related work

- **Gan et al. [6] proposed a sparse representation based classifier (SRC) based on the sparse representation theory.the basic idea is to transform the pattern recognition problem into a signal sparse representation problem without considering other types of sample data**
- **Yan et al. [7] developed a sparse representation classification with random projection (SRC-RP), which attempted to conduct sparse representation classification on compressed sample data from random projections to minimize sensor node energy costs and increase action recognition rate**
- **Sharma et al. [9]suggested a soft attention model based on LSTM for action recognition. The attention model learns**

which elements of the frames are important to the work at hand and links them to the work at hand greater priority for them. Previously, just the video level category was used as supervision in attention based approaches.This technique may lack precise and dynamical guidance, limiting their ability to innovate video modelling of complicated moves

- Klaser et al. [10] have developed a 3D HOG feature to explain human activity by extending the histogram of gradients (HOG) feature of an image. to LSTM training
- Zhang et al. [11] suggested a self-regulated view adaptation scheme that dynamically re-positions observation views, and they incorporated the suggested view adaptation scheme into an end-to-end LSTM network that automatically selects the "optimal observation perspectives during recognition.
- Feichtenhoer et al. [12] developed a unique strategy for spatio-temporal information fusion that involved shifting the classification layer to the middle of the network, which was demonstrated to enhance accuracy.
- Luo et al. [14] recommended using an encoder-decoder system based on RNNs. to forecast a sequence of basic motions expressed as atomic 3D flows in order to learn a video representation To recognize actions, the learnt representation is retrieved from the produced model
- Rahmani et al. [14] suggested an end-to-end learning approach for action recognition based on skeletal and depth data. In an end-to-end learning framework, the suggested model learnt to fuse characteristics from depth and skeletal data, capture interactions between bodyparts and/or interactions with surrounding objects, and simulate the temporal structure of human activities.

II. Methodology

In this proposed system, the involvement of RNN-LSTM is taken for consideration as this model is giving more than 90 percent of accuracy in the evaluation. The LSTM is consist of the different convolution layer with average max pooling layers in the structure of model. The main aim of the project is to detect the a 2D position can be used in activity monitoring with the same efficiency as a 3D shape. Instead of RGBD or a massive virtual reality dataset, this would facilitate for the use of RGB only cameras for people and animals posture prediction. Moreover, aim is to applying 2D shape for activity recognition is as efficient as using raw RGB photographs. This is based on the concept that restricting the input feature vector might aid in managing with a limited data, which is prevalent in animal face detection. To test the concept in anticipation of future work including overall behavior from motion in two-dimensional visuals.

I'll train an LSTM on the data to recognize the sort of activity the user is doing (as a smartphone worn around the waist). The data-set's description is as follows:

- The sensor data (accelerometer and gyroscope) were pre-processed using noise filters before being sampled in 2.56 sec fixed-width sliding windows with 50 % overlap (128 observations). A Butter-worth low-pass filter was used to separate the gravitational and body motion components of the sensor acceleration data into body acceleration and gravity. Because it is expected that the gravitational force has only low frequency components, a filter with a cutoff frequency of 0.3 Hz was employed.
- employ virtually raw data: only the gravity impact has been taken out of the accelerometer as a prepossessing step for another 3D feature to be used as an input to aid learning. You may fork my code on utilising a Butter-worth Low-Pass Filter (LPF) in Python and change it to have the correct cutoff frequency of 0.3 Hz, which is a nice frequency for activity recurrence.

A. what is RNN?

In this paper, an RNN received a total of input vectors and filters them before inventing different vectors. It may be loosely seen as in the figure below, with each rectangle having a bitmap graphics depth and other specific hidden eccentricities. The "many to one" paradigm is deployed in our instance. These models carry out the mapping in a predetermined series of phases. Recurrent networks are more appealing given those who allow us to operate on vector patterns.

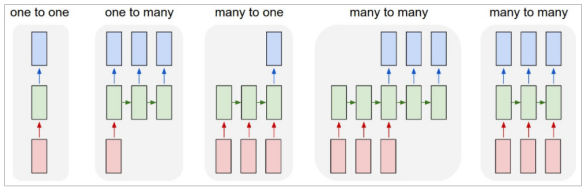

Fig. 1. Model of RNN

Each rectangle represents a vector, while the arrows denote functions (e.g. matrix multiply). The input vectors are red, the output vectors are blue, and the RNN's state is held by green vectors (more on this soon). From the top left to the bottom right: (1) Non-RNN, vanilla manner of processing, from fixed-sized input to fixed-sized output (e.g. image classification). (2) Output sequence (for example, image captioning takes a picture and outputs a caption).(3) Input sequence (e.g. sentiment analysis where a given sentence is classified as expressing positive or negative sentiment). (4) Input and output of sequences (e.g. Machine Translation: an RNN reads a sentence in English and then outputs a sentence in French). (5) Input and output sequences that are in synchronous[4].

B. What is LSTM?

Long short-term memory (LSTM) is one of Deep Learning's more difficult domains. Getting your mind around LSTM is a difficult process. It is concerned with algorithms that attempt to replicate the human brain's operation in order to find the underlying correlations in sequential data. Deep Learning is a field where this is applied. It's a class of recurrent neural networks (RNNs) that can learn long-term dependencies, which is useful for solving sequence prediction issues. Apart from single data points like pictures, LSTM contains feedback connections, which means it can process the full sequence of data. This is useful in speech recognition, machine translation, and other areas. An LSTM model's primary function is played by a memory cell known as a 'cell state', which retains its state throughout time [2]. The horizontal line that goes through the top of the figure below represents the cell state. It may be compared to a conveyor belt on which data just flows. LSTMs are built to solve these issues. All data passing via the network should pass through three gates first. These gates are activation functions that were created specifically to operate with data.

LSTM neural networks can solve a variety of problems that earlier learning algorithms, such as RNNs, couldn't. Long-term temporal relationships can be successfully represented by LSTM without a lot of optimization work. This is used to deal with high-end issues.

The sigmoid layer outputs values between 0 and 1, with 0 indicating that "nothing should be let through" and 1 indicating that "everything should be let through."

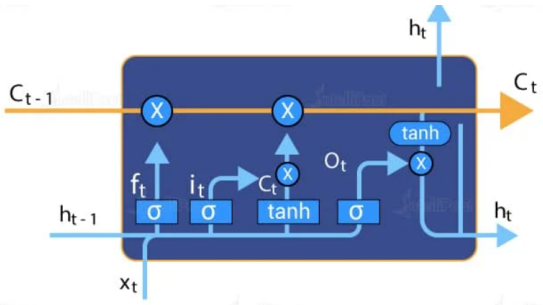

Fig. 2. Model of RNN

III. Dataset overview

The dataset consists of pose estimations, made using the software OpenPose [1] on a subset of the Berkeley Multimodal Human Action Database (MHAD) dataset [2]. There are 1438 videos in all (two were lost) with 211200 unique frames. The picture below shows an example of the four camera perspectives for subject 1 during the 'boxing' motion.

This dataset is comprised of 12 subjects doing the following 6 actions for 5 repetitions, filmed from 4 angles, repeated 5 times each.

- **JUMPING,**
- **JUMPING-JACKS,**
- **BOXING,**
- **WAVING-2HANDS,**
- **WAVING-1HAND,**
- **CLAPPING-HANDS.**

The 2D location of 18 joints throughout a time-series of frames numbered n steps (window-width) is sent into the LSTM, along with a class label for the frame series.The input for a single frame (where j is a joint) is recorded as:

$$\begin{bmatrix} j0-x & j0-y & j1-x & j1-y, \\ j2-x, & j2-y, & j3-x, & j3-y \\ j4-x, & j4-y, & j5-x, & j5-y, \\ j6-x, & j6-y, & j7-x, & j7-y, \\ j8-x, & j8-y, j9=y & & ,...,j17-y \end{bmatrix}$$

The dataset has only been lightly preprocessed for the following experiment. Experts believed outputs JSON of 18 joint x and y position keypoints and accuracies per frame JSONs converted to txt format, preserving just the x and y locations of each frame, the action done per frame, and the order of frames This is used to build a database with the related activity class number and a sequence of joint 2D positions. Multiple persons were recognised in certain frames, however only the initial detection was utilized in such situations.The data has not been normalized for subject position in the frame, mobility across frame (if any), subject size, action speed, or other factors. It's the raw 2D location of each joint as seen from a stationary camera. Sometimes individual joints were not found at position [0,0,0,0]. There is no overlap between the test and train sets, which were completely separated by activity repetition until the 26 of 32 frame overlap was created.

IV. Representation of model

In this project I had implemented RNN-LSTM model for the execution of the dataset. TO analysis the result of six different position of the human activities includes Walking, One hand clapping, Sitting, Laying and other for the analysis in the matrix format. From that LSTM is used to Search and memories the data then by normalizing data and parsing the data into the model and by fitting the model with help of Adam optimizer for the data accuracy. This experiment of the data training and testing will be done for about 300 epochs for the result analysis. In this experiment I had use different CNN layer and pre train LSTM model with merging of RNN network for the better result. About 96% of accuracy of testing is getting fro RNN-LSTM model. The result displayed for various position of human activities by creating confusion matrix.

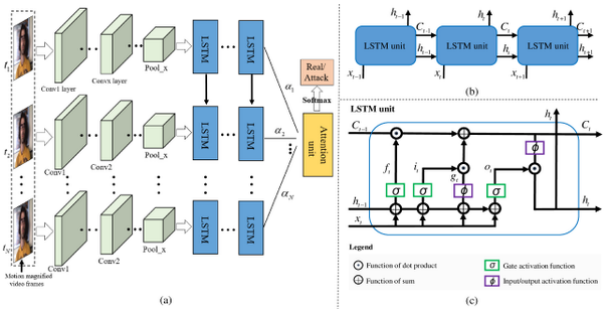

Fig. 3. Model architecture of LSTM

V. Problem statement

the data from the Smartphone's accelerometer and gyroscope is collected by domain specialists in the field of signal processing. They divide the data into 128 readings in a 2.56-second time frame with 50 percent overlap from each 2.56-second time interval, they produced 561 features. By combining human-engineered 561 feature data with raw 128 reading features. Our objective is to guess which of the six tasks a HAR Dataset for to doing in the 2.56 second time-frame. On these 561 sized domain expert designed characteristics, I used traditional Machine Learning models. I applied LSTM of Recurrent Neural Networks on 128 sized raw values collected from accelerometer and gyroscope signals since we know LSTM works well with time-series data.

VI. Discussion

To fill the gap and acquired the desire result for the data analysis with long short term memory I developed multiple layer of the RNN and LSTM for the data parsing and testing purpose. In the dataset about six categories are there for the various output of the human activities recognition so for that I had used Adam optimizer for the categorical analysis of the data. The dataset having the matrix form position of the body posture by converting the the image of human into 2D posture then by framing and getting points in form of matrix I parsed that HAR dataset into the RNN-LSTM model for the training and testing result.

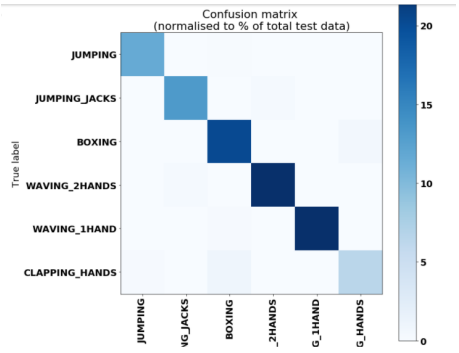

Fig. 4. Confusion Matrix

The above matrix created is explaining the various position of the human from the data had been used for training and testing purpose and getting the desired output by analysing using RNN. By analysing the result for 300 epochs I am getting the accuracy of the testing result is about more than 90% for both training and validating accuracy.

VII. Result

This project is done very accurate with the satisfied accuracy obtained after the testing result that is about 96,77% of testing accuracy and 97.5% of training accuracy for the precision of the analysis for the six different position for 2D pose detection of human activities. the Screen shot and the result detail of the model has been displayed below for the consideration. Moreover, confusion matrix is also placed for to see which position has been detected fast and more from the dataset for human recognition using RNN-LSTM. The above figure

```
PERFORMANCE ON TEST SET:             Batch Loss = 0.343032926321, Accuracy = 0.96487569809
Iter #6717440:  Learning rate = 0.000324:   Batch Loss = 0.262473, Accuracy = 0.986328125
PERFORMANCE ON TEST SET:             Batch Loss = 0.33838430047, Accuracy = 0.969396650791
Iter #6721536:  Learning rate = 0.000324:   Batch Loss = 0.284260, Accuracy = 0.978515625
PERFORMANCE ON TEST SET:             Batch Loss = 0.332896441221, Accuracy = 0.968701124191
Iter #6725632:  Learning rate = 0.000324:   Batch Loss = 0.286884, Accuracy = 0.98828125
PERFORMANCE ON TEST SET:             Batch Loss = 0.339195340872, Accuracy = 0.968527197838
Iter #6729728:  Learning rate = 0.000324:   Batch Loss = 0.262328, Accuracy = 0.982421875
PERFORMANCE ON TEST SET:             Batch Loss = 0.347805023193, Accuracy = 0.96887499094
Iter #6733824:  Learning rate = 0.000324:   Batch Loss = 0.276478, Accuracy = 0.978515625
PERFORMANCE ON TEST SET:             Batch Loss = 0.352110385895, Accuracy = 0.963658511639
Iter #6737920:  Learning rate = 0.000324:   Batch Loss = 0.263462, Accuracy = 0.990234375
PERFORMANCE ON TEST SET:             Batch Loss = 0.33824428916, Accuracy = 0.96748393774
Iter #6742016:  Learning rate = 0.000324:   Batch Loss = 0.269702, Accuracy = 0.982421875
PERFORMANCE ON TEST SET:             Batch Loss = 0.335421383381, Accuracy = 0.969918251038
Iter #6746112:  Learning rate = 0.000324:   Batch Loss = 0.267992, Accuracy = 0.984375
PERFORMANCE ON TEST SET:             Batch Loss = 0.336616426706, Accuracy = 0.96887499094
Iter #6750208:  Learning rate = 0.000324:   Batch Loss = 0.264508, Accuracy = 0.986328125
PERFORMANCE ON TEST SET:             Batch Loss = 0.337918400764, Accuracy = 0.966614484787
Iter #6754304:  Learning rate = 0.000324:   Batch Loss = 0.269014, Accuracy = 0.990234375
PERFORMANCE ON TEST SET:             Batch Loss = 0.344363123178, Accuracy = 0.966614484787
Iter #6758400:  Learning rate = 0.000324:   Batch Loss = 0.265195, Accuracy = 0.990234375
PERFORMANCE ON TEST SET:             Batch Loss = 0.317148417234, Accuracy = 0.972526490688
Iter #6762496:  Learning rate = 0.000324:   Batch Loss = 0.265389, Accuracy = 0.986328125
PERFORMANCE ON TEST SET:             Batch Loss = 0.347729802132, Accuracy = 0.964527904987
Iter #6766592:  Learning rate = 0.000324:   Batch Loss = 0.269586, Accuracy = 0.984375
PERFORMANCE ON TEST SET:             Batch Loss = 0.338627696037, Accuracy = 0.969396650791
Iter #6770688:  Learning rate = 0.000324:   Batch Loss = 0.261722, Accuracy = 0.982421875
PERFORMANCE ON TEST SET:             Batch Loss = 0.333800077438, Accuracy = 0.972178757191
Iter #6774784:  Learning rate = 0.000324:   Batch Loss = 0.260909, Accuracy = 0.9921875
PERFORMANCE ON TEST SET:             Batch Loss = 0.336783587933, Accuracy = 0.966440618038
Iter #6778880:  Learning rate = 0.000324:   Batch Loss = 0.288617, Accuracy = 0.9765625
PERFORMANCE ON TEST SET:             Batch Loss = 0.343452721834, Accuracy = 0.967136144638
Iter #6782976:  Learning rate = 0.000324:   Batch Loss = 0.283340, Accuracy = 0.98046875
PERFORMANCE ON TEST SET:             Batch Loss = 0.334667980671, Accuracy = 0.970613777637
Iter #6787072:  Learning rate = 0.000324:   Batch Loss = 0.279256, Accuracy = 0.98046875
PERFORMANCE ON TEST SET:             Batch Loss = 0.339634418488, Accuracy = 0.966440618038
Optimization Finished!
FINAL RESULT: Batch Loss = 0.339634418488, Accuracy = 0.966440618038
TOTAL TIME:  3154.37634516
```

Fig. 5. Accuracy result

represents the confusion matrix of the different positions of the human activities with how many percent training set and testing set had taken for the implementation of the project from the data set

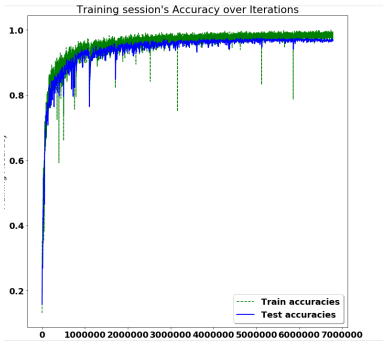

Fig. 6. Graph

This graph showing the result of the model iteration and accuracy obtained by running the model with HAR dataset parsing to this model approximately about more than 90% accuracy is obtained after thousand iterations.

VIII. Conclusion

Given that training takes around 7 minutes, a final accuracy of greater than 90% is rather impressive. There is reasonable confusion between the activities of Clapping Hands and Boxing, as well as Jumping Jacks and Waving Two Hands. In terms of its application to a larger dataset, I believe it could be used for any activity where the training includes images from various angles to be evaluated on. It would be fascinating to see whether it could be applied to camera angles other than the four in this dataset without having to train on them. Overall, this study supports the concept that 2D posture may be utilized for at least human activity identification, as well as providing evidence that 2D pose can be used for behaviour estimation in both humans and animal.

IX. Future Scope

Based on a baseline of normal motion, additional study will be done into the usage of more nuanced activity classes, such as walking versus running, agitated movement against calm movement, and perhaps normal versus aberrant behaviour. In addition, I will modified functions and the upgrade libraries that are written below:

- **A pipeline for qualitative results**
- **A validation of dataset and will try new dataset in this model**
- **upgrade the Momentum**
- **Normalise input data at each parsing of the model**
- **Adding Dropout**
- **Comparison of effect of changing batch size**

ACKNOWLEDGMENT

The preferred implementation has been done with the use of RNN-LSTM model on HAR(Human activities recognition) dataset for the evaluation of six categories of the human position that is Walking, Clapping, Laying and other. This project getting the accuracy of 96.77% for the testing results. I had used various libraries such as Numpy, placeholder, Sklearn, Matplotlib.

References

[1] **https://github.com/CMU-Perceptual-Computing-Lab/openpose's**
[2] **http://tele-immersion.citris-uc.org/berkeley-mhad.**
[3] **R. Poppe, "A survey on vision-based human action recognition," Image and Vision Computing, vol. 28, no. 6 pp. 976–990, 2010**
[4] **R. Poppe, "A survey on vision-based human action recognition," Image and Vision Computing, vol. 28, no. 6 pp. 976–990, 2010**
[5] **E. H. Miller, A note on reflector arrays (Periodical style Accepted for publication), IEEE Trans. Antennas Propagat., to be publised.**
[6] **Chen, L., Wei, H., Ferryman, J., 2013. A survey of human motion analysis using depth imagery.Pattern Recognition Letters 34, 1995–20**
[7] **M.T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation." [Online]. Available: https://arxiv.org/abs/1508.04025,2015**
[8] **E. P. Ijjina and K. M. Chalavadi, "Human action recognition in RGB-D videos using motion sequence information and deep learning," Pattern Recognition, vol. 72, pp. 504–516,2017.**
[9] **Chen, L., Wei, H., Ferryman, J., 2013. A survey of human motion analysis using depth imagery.Pattern Recognition Letters 34, 1995–2006**
[10] **Ye, M., Zhang, Q., Wang, L., Zhu, J., Yang, R., Gall, J., 2013. A survey on human motion analysis from depth data, in: Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications, pp. 149–187.**
[11] **Aggarwal, J.K., Xia, L., 2014. Human activity recognition from 3D data: A review. Pattern Recognition Letters 48, 70–80.**
[12] **Escalera, S., Athitsos, V., Guyon, I., 2016. Challenges in multimodal gesture recognition. Journal of Machine Learning Research 17, 1–54.**