DOI 10.31256/UKRAS19.24

Movement and Gesture Recognition Using Deep Learning Technology

Baao Xie , Baihua Li, and Andy Harland,

Loughborough University, Loughborough, UK LE11 3TU

***Abstract*—For several decades, the pattern recognition of movement and gesture shows promise for human-machine interaction in many areas. A remarkable application in this area is gesture recognition for upper limb amputees using surface electromyography (sEMG) to capture the muscle activation as electrical signals. Another well-known application of in this field is human activity recognition (HAR). Most HAR applications are based on raw sensor inputs such as accelerometer and gyroscope signals which show its ability in learning profound knowledge about movement recognition [1]. Within the field of signal-based gesture recognition, traditional machine learning (ML) approaches have been widely used [2]. ML models give a high accuracy with large amounts of hand-crafted, structured, and under controlled data. However, traditional ML models require lengthy offline and batch training which is not incremental or interactive for real time application. In addition, ML models always cost a long period of time to extract a set of reliable features especially for high-dimensional, complex and noisy data because of the various situations in practical applications. Besides the ML methodologies, in recent years, the use of deep learning (DL) algorithms has become increasingly more prominent for their tremendous ability to extract and learn features from large amounts of data [3]. Compared to ML models, DL models make it possible for artificial intelligence to train the networks without hand-craft feature extracting. The aim of this work is to develop DL based methods for human movement and gesture recognition from time-series signals such as obtained using sEMG and IMU signals. We would like to understand the performance of DL for time-series signal analysis and accuracy, as to our knowledge, this aspect is still understudied. A series of experiments have been conducted to achieve it with different datasets and signals. The DB1 is a HAR dataset from the UCI repository. The DB2 and DB3 are sub-datasets of Ninapro database contains the recordings of 17 gestures from subjects by collecting sEMG signal. There are 4 different DL models designed for the experiments to find out the optimum solution by performance comparison: a 1-D CNN, a LSTM model, a C-RNN and 3+3 C-RNN. This is an extended abstract of a poster for the conference. The details of datasets and models are described in the methodology section, followed with the result section to present the results of different DL models on datasets.**

I. Methodology

A. Models

In the experiment, 4 DL models were used for gesture and movement recognition. The first one is a 1-D CNN which was inspired by [4]. The model processes separable convolution operation on each channel of the data rather than do the convolution on the entire input matrix. There are 2 convolutional layers in the model with max pooling and activation function applied after each convolution layer. The output of several separable convolution layers is the feature maps of inputs from different channels. And a fully-connected layer will be applied on these nodes, following by the classifier to generate the result. The second competing model is a basic LSTM with the sequence length of 128 which equals to the sliding window size. There is a dropout layer after the LSTM layer with problem rate of 0.8 to overcome the overfitting problem. And a fully-connected layer will be applied on these nodes, following by the classifier to generate the result.

The third model and fourth model is hybrid model combined with CNN and RNN. Convolutional layers are applied as a feature extractor in the structure. The output of the convolutional layers is the feature map of the input signal which contains useful information for other layers. The LSTM layers focus on the influence from previous time point and generate a probability map for each input. In the early stage of the experiment, a basic C-RNN model was developed including 1 convolutional and 1 recurrent layer. After some literature reviews and modifications, another advanced C-RNN model was built with 3 convolutional layers and 3 recurrent layers. In this section, the structure of the 3+3 C-RNN is described with more details.

As shown in Figure 1, the input of the 3+3 C-RNN model should be a signal piece fixed by a sliding window. The width of the input is in the time domain, and the size equals to the window size *w*. The height of the signal should always be 1. *C* represents the number of input channels while the signal from each channel will be fed into 1-D convolutional layers separately. For one round of training, a batch of such signal piece will be fed into the network where the batch size equals to *i*. As Figure 2 shows, the number of filters of the 1st Conv layer is designed as *C**2, with the filter size of 2 and stride size of 1. The zero-padding approach is applied after each Conv layer to generate a feature map (FM) in the same width. The output format of 1st Conv layer should be [*B(i), w, C*2*]. The 2nd and 3rd Conv layers are designed to have the same filter size and stride size but twice of the number of filters. The output of the 3rd Conv layer should be [*B(i), w, C*8*]. It is worth mentioning that, the parameters in the layers are controllable for a better performance. And different from traditional CNN, there is no max pooling layer after convolution which aims to keep the integrality of data and ensure the fixed length of the sequence to feed into LSTM layers.

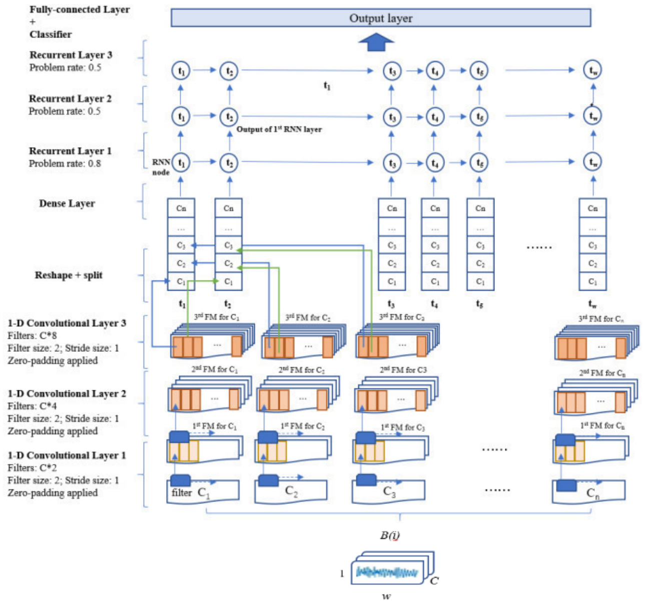


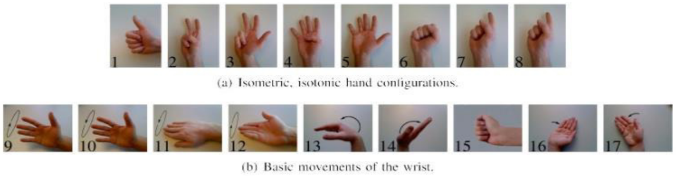
Figure 1. Structure of 3+3 C-RNN.

As shown in Figure 1, the output of convolutional layers are sequences of the feature map. These feature sequences will be reshaped into a node for recurrent layers. The width of the sequence is treated as the time period of recurrent layers which equals to *w*. Then, a dense layer will transform these nodes and feed them into the LSTM cells, each with the dimension LSTM size (*Ls*). This size parameter is designed to be 3 times larger than the number of channels, which is the similar way in the embedding layers in text applications where words are embedded as vectors from a given vocabulary. Then the sequence with the length of window size will be feed into three LSTM layers continuously. The input of each layer is the output from the previous layer. The dropout function is applied with problem rate of 0.8 for 1st and 2nd layers. And for the 3rd recurrent layer, the problem rate will be 0.5. In addition, the gradient clipping approach is added to improve training by preventing exploding gradients during back propagation. Only the last member of the sequence at the last LSTM layer is used as the final result, which will be feed into the fully-connected layers and a Softmax layer for classification.

B. Datasets

Database 1: The DB1 used in the experiment is the HAR dataset from the UCI repository. The dataset is taken from with 30 subjects within an age range of 19-48 years. Each volunteer was asked to perform six movements (walking, walking upstairs, walking downstairs, sitting, standing and laying) wearing a smartphone on the waist. The accelerometers, gyroscope, and body accelerometer signals were recorded at a sampling rate of 50 Hz. The dataset was separated into two parts randomly where 70% of the set was selected as training set and 30% as the testing set. In the pre-processing step, noise filters were applied to the signals. The signals sampled in the fixed-width sliding window of 2.56 sec with 50% overlapping [5].

Dataset 2: The DB2 and DB3 used in the experiment are sub-datasets of Ninapro database which provides a repository of sEMG data. sEMG measures the electrical activity when muscles are moving and exercising. It is an important attribute of the nervous systems aimed at collecting more muscular force or compensating for force losses. The purpose of the Ninapro project is to aid research on advanced hand myoelectric prosthetics with public datasets [6]. Currently, there are 7 databases available, each containing results from a series of movements where volunteers performed sets of hand, wrist and finger movements in controlled laboratory situations. The DB2 is the sub-dataset 5 of Ninapro database which contains data acquisitions of 10 subjects. The sEMG signals in the set were collected using two Thalmic Myo armbands with 16 electrodes, providing the upsampled sEMG signal at 200 Hz. The armbands were fixed close to the elbow according to the Ninapro standards. Each subject repeats 17 different hand movements for 6 times. Each movement lasts for 5 seconds and following by 3 seconds of rest as shown in Figure 3.



(a) Isometric, isotonic hand configurations.

(b) Basic movements of the wrist.

Figure 2. 17 gestures in Ninapro databases.

The subject 1-7 were treated as training set and subject 8,9,10 were selected as the testing set.

Dataset 3: The DB3 is the sub-dataset 2 of Ninapro database, which contains data acquisitions of 40 subjects. The sEMG signals in the set were collected using 12 electrodes from a Delsys Trigno Wireless System, providing the raw sEMG signal at 2 kHz. The type of movements of DB3 is same as DB2. The dataset was separated into two parts randomly where 70% of the set was selected as training set and 30% as the testing set. More details and attributes information of DB2 and DB3 are available at http://ninapro.hevs.ch/node/7.

II. Result

For each model, the learning rate is set at 0.0001 and the epoch size is set as 1000. The batch size is designed as 600. The training and testing are implemented on a computer with GPU of GTX 1080ti and CPU of Intel(R) Core(TM) i7-7700k @ 4.20Ghz. The programming platform is Tensorflow with python. Table 1 and Figure 3 show the average accuracy of different models when applied on datasets. It is obvious that 3+3 C-RNN gives the best performance on three datasets, which are 90.29%, 83.61% and 63.74%. For the Ninapro datasets (DB2 and DB3), 1-D CNN produces aresult of 53.17% when compared to other models. It is clear that for these 2 datasets, the models containing LSTM layer give a better accuracy, which means the relationships between different time points have more influence on the sEMG signal recognition.

TABLE I. PERFORMANCE OF DL MODELS

Models	DB1	DB2	DB3
1-D CNN	88%	72.49%	52.17%
LSTM	86.8%	78.13%	55.3%
C-RNN	87.62%	82.1%	59.31%
3+3 C-RNN	90.29%	83.61%	63.74%

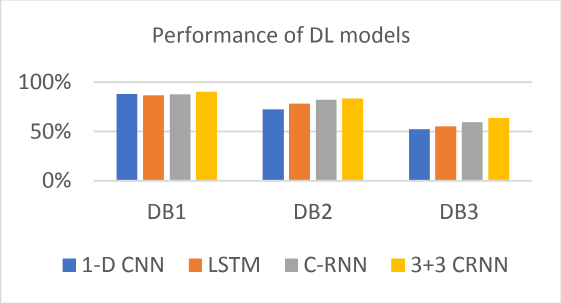


Figure 3. Experiment results of DL models on different dataset

However, for the HAR dataset, 4 models produce a high accuracy (above 85%). The one reason is the HAR database has fewer classes (6) than the Ninapro database (18) which makes it easier to classify. In addition, the class of ‘rest’ in Ninapro datasets seems to cause a decrease of the accuracy.

It is also worthy to mention that a large number of subjects (40 for DB3) with insufficient sample data cause a confusion for DL models and lead to a lower accuracy. Theoretically, this situation should be ameliorated if more sample data are fed to the networks. For the future experiments, we plan to compete the models with other existing researches using traditional ML or DL models. In addition, we will have a series of trials on different hyperparameters such as window sizes, filter sizes and batch sizes. We should also improve the structure of 3+3 C-RNN based on the experiments mentioned above to get a better performance in the future.

REFERENCES

[1] Wang, J., Chen, Y., Hao, S., Peng, X., and Hu, L. 2018. Deep Learning for Sensor-based Activity Recognition: A Survey. *Pattern Recognition Letters* 103,1 (Feb. 2018), 1-9.

[2] Lara, O.D., Labrador, M.A. 2012. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys & Tutorials* 15, 3 (Nov. 2012) 1192-1209.

[3] LeCun, Y., Bengio, Y., Hinton, G. 2015. Deep learning. *Nature* 521,1 (May. 2015), 436-444.

[4] Saeed, A., 2016. Implementing a CNN for Human Activity Recognition in Tensorflow. Retrieved from https://aqibsaeed.github.io/2016-11-04-human-activity-recognition-cnn/

[5] Anguita D, Ghio A, Oneto L, et al. 2013. A public domain dataset for human activity recognition using smartphones. In *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*. (Bruges, Belgium).

[6] Pizzolato, S., Tagliapietra, L., Cognolato, M., Reggiani, M., Müller, H., & Atzori, M. 2017. Comparison of six electromyography acquisition setups on hand movement classification tasks. PloS one, 12, 10 (Oct. 2017), e0186132.