Proceedings of 9th IEEE International Conference
on CYBER Technology in Automation, Control, and Intelligent Systems
July 29- August 2, 2019, Suzhou, China

An Incremental Learning Framework for Skeletal-based Hand Gesture Recognition with Leap Motion

Jie Li, Junpei Zhong, Fei Chen, Chenguang Yang

***Abstract*—Hand gesture recognition has become the focus of researchers lately because of its manifold applications in various fields. Leap Motion (LM) is a device to obtain useful and accurate information of the hand action, which is suitable for collecting the three-dimensional (3D) human hand gesture. In this paper, a novel framework which consists of an incremental learning (IL) algorithm without deep structure is proposed and applied to hand gestures classification that explicitly aimed to the LM data. The same datasets are used to train the proposed framework and the conventional Long Short Term Memory Recurrent Neural Network (LSTM-RNN). Due to the structural advantage of the proposed model, the recognition performance is improved distinctly in robustness and training time than the LSTM network. Moreover, convincing experiment results are given to illustrate that the solution is more efficient in static gesture classification.**

I. INTRODUCTION

As one of the most dexterous and sophisticated limbs of human, hands are essential to implement of comprehensible, intuitive, expressive and abundant activities. Hand gesture recognition can be regarded as a way of that computer interpret human body language. In the last few years, interest in human behaviors and gesture recognition has attracted growing attention from many researchers. This is due to its widely range of applications in many different domains to some extent. It has brought a revolution in different fields for this technology, such as language/communication, entertainment, education, security, health care, abundant human-computer interaction scenarios and so on [1] [2].

Gesture is a sequence included the temporal and spatial information. And the hand gesture is a static pose or a dynamic moving trajectory of our hands. Gesture recognition has been widely investigated for several decades in the domains of computer vision [3] [4]. A substantial achievement for this task has been obtained by the researchers two decades ago. Traditional studies about gesture classification are generally based on the two-dimensional (2D) color images to extract a set of correlative features. Many approaches are applied to tackle 2D gesture recognition, including the the Hidden Markov Model (HMM) [5], particle filtering [6], Support Vector Machine (SVM) [7], etc. [8] [9]. Since it is easy to affected by the background disturbance, light and other factors when human hands are detected from the two-dimensional RGB images. To improve the gestures recognition performance, a lot of low-cost depth cameras for instance Microsoft Kinect have provided several possible ways to a number of different structures that make use of these devices to acquire the depth information. These approaches based on deep-learning recently become a common method in gesture recognition, achieving outstanding performance. The authors in [10] proposed a two-stream ConvNet structure which consists of a network of spatial and temporal dimensions. The training input of the ConvNet composed by the multi-frame optical flow is able to achieve good results, although the training data is limited. Eleni Tsironi et al. proposed a Convolutional LSTM-RNN (CNNLSTM) to deal with dynamic gesture recognition [11].

Another vision-based 3D acquisition device is LM which provides the 3D data of the frames scene for human hands. It is more professional in hand gesture recognition than Kinect. The hand orientation and position of the finger are supported directly without contacting or touching. For the Hidden Conditional Neural Field (HCNF), the authors [12] presented an unusual feature data which includes single-finger features and double-finger features to classify the dynamic hand gestures with a Leap Motion controller and a better recognition accuracy is achieved. The first features address the issue of labeling errors caused by performing dynamic gestures at different locations and the various kinds of interactions between contiguous fingertips can be distinguished between by the second features. Some other researchers use LM to collect 3D hand features such as coordinates, palm position to train the LSTM-RNN network which is able to recognize two kind of basic hand gestures [13].

Deep structural neural networks have achieved great success in large-scale data processing and diversifies fields. Although deep structured networks are very powerful, most are plagued by extremely time-consuming training processes. IL network as an effective incremental learning algorithm without the requirement for deep structure is proposed in 2017 [14]. Nowadays, IL network has been employed in classification problem of several fields [15] [16] [17]. The IL network performed better results in classification than the deep structure based network.

In this paper, we proposed an unfrequent framework aiming at recognizing the different hand postures collected by LM. A set of features concerning different static hand gestures are fed into the IL network to recognize various gestures. The

This work was partially supported by National Nature Science Foundation (NSFC) under Grants 61861136009 and 61811530281.

J. Li and C. Yang are with the Key Laboratory of Autonomous Systems and Networked Control, School of Automation Science and Engineering, South China University of Technology, Guangzhou, 510640, China.

J. Zhong is with School of Science and Technology, Nottingham Trent University, Nottingham, NG11 8NS, UK.

F. Chen is with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy.

C. Yang is with Bristol Robotics Laboratory, University of the West of England, Bristol, BS16 1QY, UK.

Corresponding author is C. Yang. Email: cyang@ieee.org.

same datasets from LM have also been sent to the LSTM-RNN network. And we also compare the performance of the two methods in gesture classification in this paper. Moreover, convincing experiment results are given to illustrate that the IL network outperform the LSTM-RNN network in recognition performance.

This paper is structured as follows. In Section II, we introduce the general architecture of the IL network and give a brief overview of LSTM-RNN network. Section III presents the details with respect to data collection and our experimental setup. In Section IV, we provide the experiment results and analysis about those results. Finally, we conclude with some discussion and conclusions.

II. PRELIMINARIES

In this section, two approaches are introduced: 1) RNN classifier with LSTM [18] and 2) the IL network in order to address the issue of gesture recognition accuracy and efficiency.

A. LSTM

Compared with the HMM, recurrent neural network does not require that the data is probabilistic, it is more easier to train by simple stochastic gradient descent (SGD) algorithm. However, due to the activation functions used in the hidden layers of the RNN, gradients vanishing or exploding phenomenon occurs during back-propagation through time, which makes the model difficult to remember the long-term dependencies of the time-series. As a special kind of RNN, the LSTM network are capable of learning long-time dependencies. It [19] was introduced by Hochreiter and Schmidhuber and designed to overcome the modeling shortcomings of traditional RNN.

The architecture of LSTM network is shown in Fig. 1. The key to LSTM network is the special units called memory blocks in the form of gates within the RNN framework in the hidden layer. Each memory block of a traditional LSTM network is composed by a memory cell, a forget gate, an input gate and an output gate. The memory cell records values depending on the random time intervals. The forget, input, and output gates control the information flow to enter and out from the cell. The three separated gates work in tandem to ensure the information is retained in the cell long enough. The actual structure is shown in Fig. 2.

In the following equations, the notation of $\odot$ is used to represent the element-wise product between vectors, $\boldsymbol{h}_t$ and $\boldsymbol{x}_t$ are applied to present the output and input of the LSTM layers. The subscripts of f, i and o are related to the parameters of forget, input and output gates. There are three activation functions which expressed by the sigmoid function $sigma(.)$, the hyperbolic activation function $tanh(.)$, and the above-mentioned element-wise multiplication $\odot(.)$, while $tanh$ denotes the output activation function of LSTM. Employing the memory gates, the LSTM network updates equations iteratively from $t = 1$ to $t = T$.

$$\boldsymbol{i}_t = sigma(\boldsymbol{W}_{ix}\boldsymbol{x}_t + \boldsymbol{U}_{ih}\boldsymbol{h}_{t-1} + \boldsymbol{V}_{ic}\boldsymbol{c}_{t-1} + \boldsymbol{b}_i), \quad (1)$$

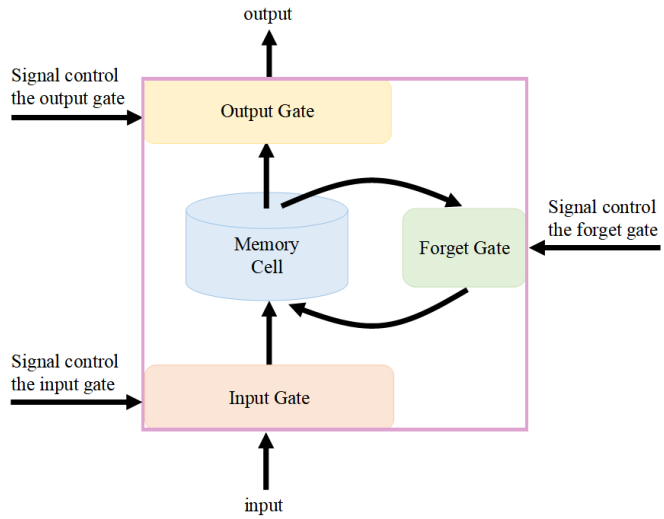


Fig. 1. The architecture of LSTM network

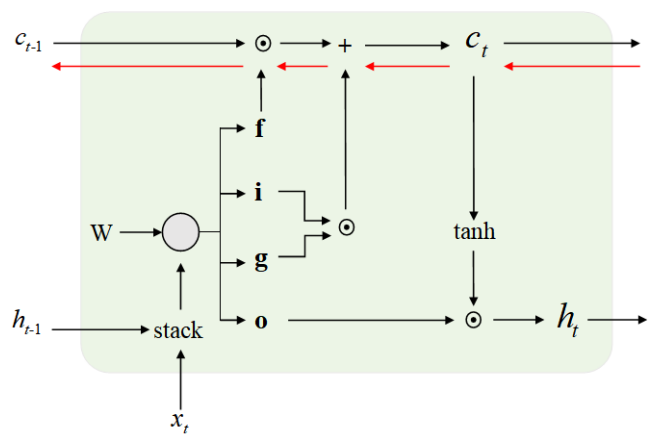


Fig. 2. The actual structure of LSTM network

$$\boldsymbol{f}_t = sigma(\boldsymbol{W}_{fx}\boldsymbol{x}_t + \boldsymbol{U}_{fh}\boldsymbol{h}_{t-1} + \boldsymbol{V}_{fc}\boldsymbol{c}\boldsymbol{x}_{t-1} + \boldsymbol{b}_i), \quad (2)$$

$$\boldsymbol{o}_t = sigma(\boldsymbol{W}_{ox}\boldsymbol{x}_t + \boldsymbol{U}_{oh}\boldsymbol{h}\boldsymbol{x}_{t-1} + \boldsymbol{V}_{oc}\boldsymbol{c}_{t-1} + \boldsymbol{b}\boldsymbol{x}_o), \quad (3)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \tilde{c}_t, \quad (4)$$

$$\tilde{c}_t = \boldsymbol{W}_{cx}\boldsymbol{x}_t + \boldsymbol{U}_{ch}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c, \quad (5)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot tanh(\boldsymbol{c}_t), \quad (6)$$

where $\boldsymbol{i}_t$ is the input gate variable at time t. Similarly, $\boldsymbol{o}_t$ is output gate, $\boldsymbol{f}_t$ is forget gate, $\boldsymbol{c}_t$ is the memory cell state and $\tilde{c}_t$ is employed to express the memory gates at the current time t. $\boldsymbol{b}_f$ $\boldsymbol{b}_c$, $\boldsymbol{b}_i$ and $\boldsymbol{b}_o$ are respectively the forget gate memory cell biases vectors, input gate and output gate of the LSTM network. The weight between input and the forget gate is denoted as $\boldsymbol{W}_{fx}$. $\boldsymbol{W}_{ox}$ and $\boldsymbol{W}_{ix}$ represent the weights between the input and the output gate, the input gate respectively. The weights between the hidden layer and input gate, forget gate, and output gate in term of memory are $\boldsymbol{U}_{ih}$, $\boldsymbol{U}_{fh}$ and $\boldsymbol{U}_{oh}$, while $\boldsymbol{V}_{iC}$, $\boldsymbol{V}_{fC}$ and $\boldsymbol{V}_{oC}$ are the weights between cell state, forget gate, input gate and output gate.

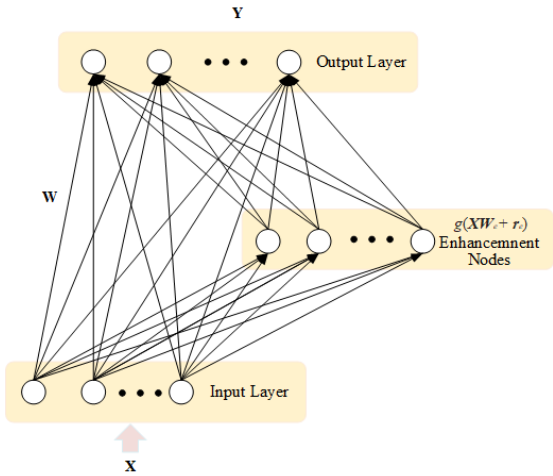


Fig. 3. The structure of FLNN

B. IL Network

Deep architecture networks have successfully used in numerous fields. Nevertheless, the time-consuming training process, amounts of hyperparameters and complicated structures turned the deep structure network into a challenging problem. Moreover, these complex problems are pivotal sides with regard to train a mass of parameters with deep-structured networks. Although several neural networks, such as random vector functional-link neural network (RVFLNN), are proposed to eliminate the shortcomings of the long training time and the lack of generalization ability in regard to function approximation [20] [21]. However, the RVFLNN is not very effective in the case of remodeling large capacity and time varying data. Recently, a new learning system called IL network is proposed to address the aforementioned problems, which is inspired by the RVFLNN.

The IL network is essentially a RVFLNN with flat functional-link architecture due to its fast training speed and generalization ability. The structure of a typical FLNN is presented in Fig. 3 [22]. The biggest difference of the structures between the two networks is the way of connections between input and output. In the IL network, we convert the input data sets into feature nodes by the given functions. Moreover, we can also develop the network by the enhancement nodes in the way of flatting it. The structure of the IL network is presented in Fig. 4 [14]. And the particular descriptions about the IL network are presented below.

Assume that the data (feeding into the network which includes N samples with M variables) is denoted by $\boldsymbol{X}$. The output data is represented as $\boldsymbol{O}$. Namely, the output of IL network is $\boldsymbol{O}$ and $\boldsymbol{O}$ belongs to $R^{N\times C}$. The expression of $g_i(\boldsymbol{X}\boldsymbol{W}_{bi}+r_{bi}), i=1,2,...,n$ is adopted to describe the ith mapping feature which is denoted by $\boldsymbol{M}_i$, and g is the transformation corresponding to the activation function in the deep network. $\boldsymbol{W}_{bi}$ represents the random weight of feature maps and decides the dimensions of these mapped features. In addition, $\boldsymbol{M}^i\equiv[\boldsymbol{M}_1,...,\boldsymbol{M}_i]$ is used to denote the first i groups in regards to the mapped features. Likewise, the first j groups with respect to enhancement nodes $\boldsymbol{E}^j\equiv[\boldsymbol{E}_1,...,\boldsymbol{E}_j]$ are generated by $q_j(\boldsymbol{M}^i\boldsymbol{W}_{ej}+r_{ej})$. The weights $\boldsymbol{W}_{ej}$ have relations with the enhancement nodes. Both $\boldsymbol{W}_{ej}$ and $\boldsymbol{W}_{bi}$ are sampled from the distribution density $l(w)$. In particular, we can choose different values with respect to the subscripts of g_i and q_j in terms of the complexity the model. Namely, the functions g_i and g_k are not equal for $i\neq k$. Similarly, q_j and q_t are also different for $j\neq t$. In the undermentioned formulas, the subscripts of g_i and q_j are omitted to avoid confusing.

The detailed formulations of the IL network are given below. From the previous statement, the input data for IL network is $\boldsymbol{X}\in R^{N\times M}$, and the output matrix $\boldsymbol{O}\in R^{N\times C}$. Each feature mapping can generate k nodes. As for the formula of the n feature mappings is represented as

$$\boldsymbol{M}_i=g(\boldsymbol{X}\boldsymbol{W}_{bi}+r_{bi}), i=1,2,...,n. \tag{7}$$

for $\boldsymbol{M}^n\equiv[\boldsymbol{M}_1,...,\boldsymbol{M}_n]$, the first mth sets with respect to the enhancement nodes are computed by

$$\boldsymbol{E}_m=q(\boldsymbol{M}^i\boldsymbol{W}_{em}+r_{em}). \tag{8}$$

From the above, the IL network can be formulated by

$$\begin{aligned}\boldsymbol{O}&=[\boldsymbol{M}_1,...,\boldsymbol{M}_n|g(\boldsymbol{M}^n\boldsymbol{W}_{e1}+r_{e1}),\\ &...,q(\boldsymbol{M}^n\boldsymbol{W}_{em}+r_{em})]\boldsymbol{W}^m\\ &=[\boldsymbol{M}_1,...,\boldsymbol{M}_n|\boldsymbol{E}_1,...,\boldsymbol{E}_m]\boldsymbol{W}^m\\ &=[\boldsymbol{M}_n|\boldsymbol{E}_m]\boldsymbol{W}^m,\end{aligned} \tag{9}$$

and the weight $\boldsymbol{W}^m=[\boldsymbol{M}_n|\boldsymbol{E}_m]^+\boldsymbol{O}$. $\boldsymbol{W}^m$ is computed by $[\boldsymbol{M}_n|\boldsymbol{E}_m]^+$ with extremely rapid speed. The computation of the pseudo-inverse of $[\boldsymbol{M}_n|\boldsymbol{E}_m]$ is deduced by the following equation

$$[\boldsymbol{M}_n|\boldsymbol{E}_m]^+=\lim_{\lambda\to 0}(\lambda\boldsymbol{I}+[\boldsymbol{M}_n|\boldsymbol{E}_m][\boldsymbol{M}_n|\boldsymbol{E}_m]^T)^{-1}[\boldsymbol{M}_n|\boldsymbol{E}_m]^T. \tag{10}$$

where λ represents the l_2-norm regularization. IL network is much faster than current deep learning networks in both the complexity of structure and computation formula.

III. EXPERIMENTS DESIGN

In this part, two neural networks are employed to training the hand gestures data, and a comprison analysis is done in recognizing performance. Before that, we firstly present the feature descriptor collected by LM.

A. Feature Extract from Leap Motion

LM, as a sophisticated hardware device, is used to collect hand gesture data. The device is specifically designed to detect and track human hands motion, and to generate a wealth of data about various features from the same hand gestures. It is consisted of a binocular camera and three infrared LEDs. The LM is capable to accurately capture and extract essential information such as skeletal joint angle, palm positions, and so on. LM follows the right-handed coordinate system. And the coordinates are shown in Fig. 5. As to the coordinate reference system about data frame, the LM device is the coordinate center. The original point of the coordinate is in the middle of the top of LM. The XZ axis forms a horizontal plane, the

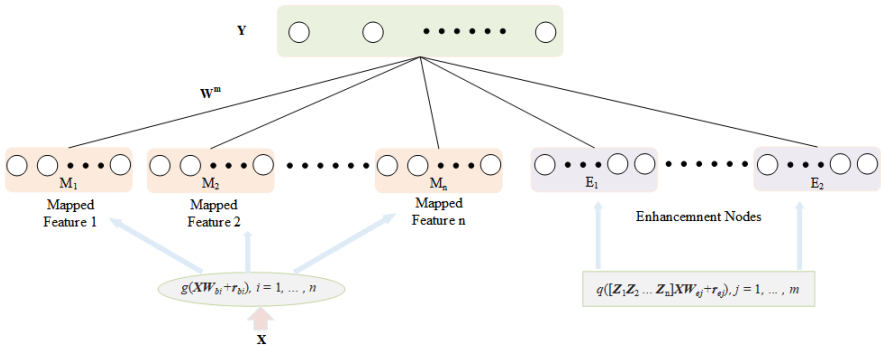


Fig. 4. The architecture of IL network

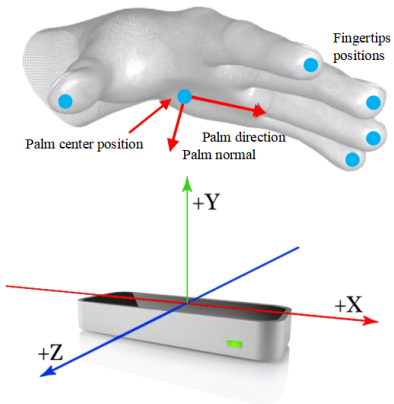


Fig. 5. The coordinates of Leap Motion

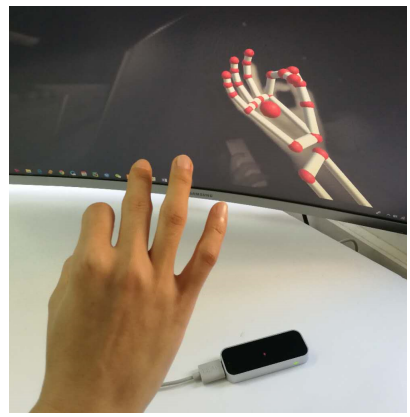


Fig. 6. The skeleton of human hand captured by Leap Motion

Y axis is perpendicular, the X axis is in the direction of the long side of the LM. For Y axis, the up direction is positive. The Z axis is positive in the screen direction.

As shown in Fig. 5, when human hand palm is facing the Leap Motion with five fingers, it is capable to accurately capture and extract essential information such as skeletal joint angle, palm positions, palm velocity, palm normal and so on. Relevant research [23] shows that the error about the 3D coordinate of fingertips is about 200 μm. In our experiment, ten features from all fingers on one hand are chosen as the input of our training network. To distinguish different hand gestures, the skeletal joint angle from our right hand for each fingers are computed depending on the position fingers. The skeleton of human hand captured by LM is shown in Fig. 6. It is well known that there are fourteen joints on a single hand for all fingers. Except the thumb, the rest fingers all have coupled joint angles. Therefore, only the critical features are extracted.

In order to acquire more accurate data, the sampling frequency of collecting hand gesture data is 20 Hz. To avoid the invalid data and improve the robust performance, several steps are included in the process of data pre-processing. First of all, We delete the invalid gesture data. Then, we select the same number of frames for different gestures. The skeletal joint angles are not influenced by the hand location and therefore we capture several sets of skeletal joint angles for the same gesture in different hand location. Finally, these data are merged into one CSV file with various labels.

B. Gesture Recognition

To highlight the framework, we proposed for recognizition, the LSTM network is evaluated for a comparison. The LSTM network is structured with 7 neurons to deal with the gestures sequences. More neurons will bring the problem of overfitting. The output layer is structured with 8 neurons which is a fully-connected layer. we choose 8 static gestures to classify. These gestures are presented in Fig. 7. The activation function with respect to output layer is a sigmoid function.

A range of experiments for gestures recognition are considered to prove the efficient IL network. The familiar gestures are composed of 320 gestures to train the network and 80 gestures to test it. Since the training is performed for many times, the proportion between the test sets and training sets is changing for each network. The training results in above-mentioned scenario are presented in Fig. 8 and Fig. 9.

In the IL network, the weights such as $\boldsymbol{W}_{bi}$ and r_{bi}, $i = 1, 2, ..., n$ are randomly generated. $\boldsymbol{W}_{ej}$ and r_{ej}, $i = 1, 2, ..., m$ are both set between -1 and 1, and the values of this two parameters follow a normal distribution. Different mapping

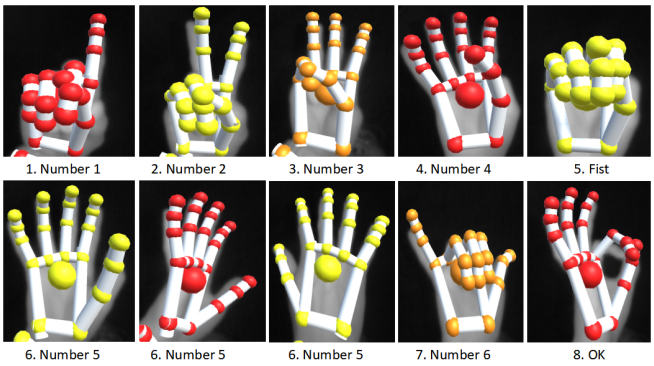


Fig. 7. The hand gestures in our data

nodes and enhancement nodes decide the test accuracy of the network. The detailed results are shown in the Tab. I and Tab. II.

IV. EXPERIMENTAL RESULTS ANALYSIS

We consider that the ratio of test data to training data is expressed as β. Fig. 8 presents the accuracy, the error and the performed time of the LSTM network with the constant β. In Fig. 8, the pink and red lines respectively denote the training accuracy and training error, while the dotted lines with blue and green colors show the test accuracy and test error. It is widely agreed that, the training time will become longer with the increasing epochs for common neural network. Therefore, Fig. 8 implies that the accuracy could be improved in a certain range of epochs. In Fig. 9, four different values of β are trained where the test accuracy is exactly 100%. Obviously, with the same accuracy the smaller β is, the smaller epoch is. Looking closely at the c and d sub-figures of Fig. 9, it is not difficult to find that although the number of network training is small with the 100% accuracy of the c graph, but the accuracy curves of the training and test sets are more unstable. For the LSTM network, we can conclude that when the network parameter β is 0.05 and the epochs is 28, the network reaches the best recognition performance. The corresponding training time is 13.883 seconds. It should be noted that we mainly takes the recognition accuracy and training time into account to evaluate the recognition performance.

Table I reports the recognition accuracy and training time on gesture data set with different mapping nodes and enhancement nodes. Similar to LSTM, the IL network spends less time when the mapping nodes and enhancement nodes reduce. Nevertheless, it is worth noting that the executing time of IL network is much less than LSTM network. According to the Tab. II, we can see that initially the executing time of IL network decreases when β becomes small. Once β is greater than 0.1, the executing time begins to increase. These results show that the best recognition performance of the IL network appears when feature nodes and enhancement nodes both are 10 and β=0.1.

Compared all the results of LSTM and IL network, it is confirmed that the IL network is well-suited for the certain hand gesture recognition tasks. The recognition performance is better and the proposed method is more robust. This is mainly because the IL network connects the input and the output in a special way. In addition, the single structure of hidden network also makes a contribution to the results.

TABLE I
THE RECOGNITION ACCURACY AND TRAINING TIME ON GESTURE DATA SET WITH DIFFERENT FEATURE NODES AND ENHANCEMENT NODES

β	Number of Mapping Nodes	Number of Enhancement Nodes	Test Accuracy	Training Time (seconds)
0.2	10	10	100%	0.00698
0.2	20	20	100%	0.01097
0.2	50	50	100%	0.02693
0.2	80	80	100%	0.04089
0.2	100	100	100%	0.05084

TABLE II
THE RECOGNITION ACCURACY AND TRAINING TIME ON GESTURE DATA SET WITH CHANGING β

β	Number of Mapping Nodes	Number of Enhancement Nodes	Test Accuracy	Training Time (seconds)
0.2	10	10	100%	0.0069813
0.1	10	10	100%	0.005983
0.05	10	10	100%	0.0069811
0.03	10	10	100%	0.0059847

V. CONCLUSION AND FUTURE WORK

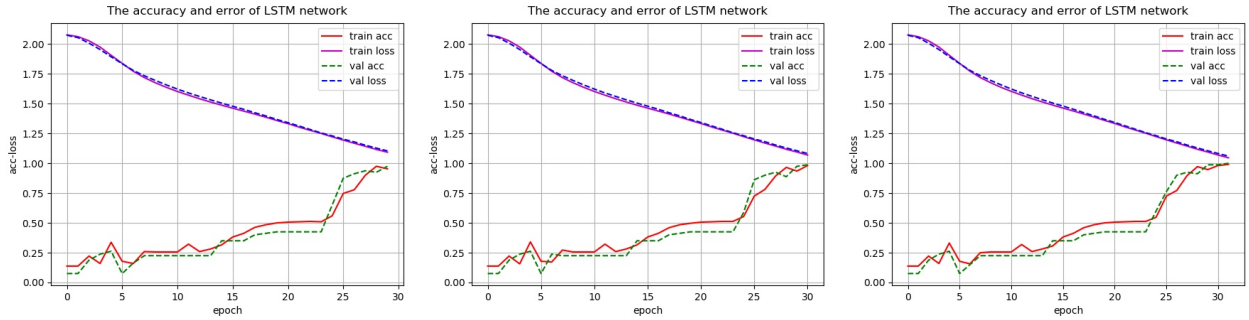
In order to overcome the problems that deep network is difficult to alleviate the extremely time-consuming training process, we propose a novel gesture recognition framework applied with Leap Motion. Two different structure neural network are compared mainly in recognition accuracy and training time. One is the proposed IL network with single network, and the other is LSTM-RNN with deep structure of multiple hidden layers. The same feature sets and training environment have been applied to train the different network. The results obtained from numerical experiments show that our proposed framework performs more effectively than the other ones.

We have presented an expandable network in width which is suitable to recognize static hand gestures with Leap Motion. Since the limition of the IL network, only one-dimension row or column feature vector can be well classified. Future work will focus on addressing the dynamic hand gesture recognition with higher dimension. The main challenge is to explore a more fast and efficient modified IL algorithm that can perfectly classify various dynamic gestures.

REFERENCES

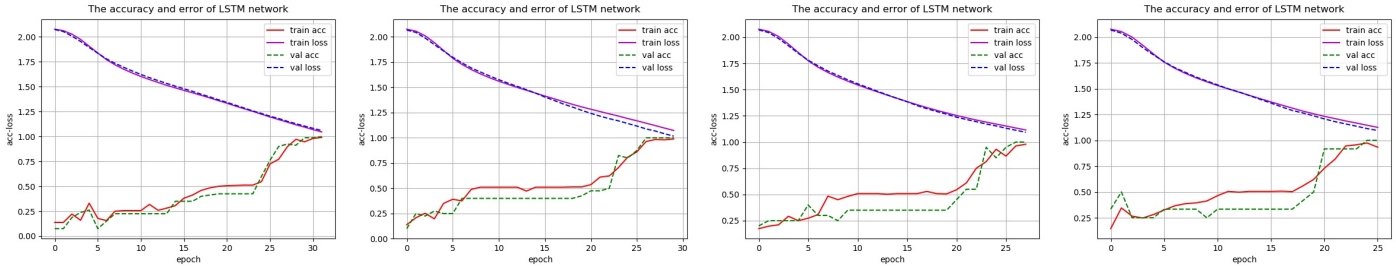
[1] Y. Xue, Z. Ju, K. Xiang, C. Jing, and H. Liu, "Multimodal human hand motion sensing and analysis -a review," *IEEE Transactions on Cognitive and Developmental Systems*, vol. PP, no. 99, pp. 1–1, 2018.

[2] M. Asadi-Aghbolaghi, A. Claps, M. Bellantonio, H. J. Escalante, V. Ponce-Lpez, X. Bar, I. Guyon, S. Kasaei, and S. Escalera, "Deep learning for action and gesture recognition in image sequences: A survey," 2017.



(a) epoch=30, accuracy=0.975, t =14.151s (b) epoch=31, accuracy=0.9785, t =14.615s (c) epoch=32, accuracy=1.0, t =15.065s

Fig. 8. The results of LSTM with β = 0.2



(a) epoch=32, β=0.2, accuracy=1.0 (b) epoch=31, β=0.1, accuracy=1.0 (c) epoch=28, β=0.05, accuracy=1.0 (d) epoch=26, β=0.03, accuracy=1.0

Fig. 9. The results of LSTM with changing β

[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 90 Towards A New Frontier of Applications*, 1990.

[4] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," in *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pp. 379–385, IEEE, 1992.

[5] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden markov models," in *International Symposium on Computer Vision*, 2002.

[6] L. Bretzner, I. Laptev, and T. Lindeberg, "Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2002.

[7] N. H. Dardas and N. D. Georganas, "Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 11, pp. 3592–3607, 2011.

[8] Y. Wu and T. S. Huang, "Vision-based gesture recognition: A review," in *International Gesture Workshop*, pp. 103–115, Springer, 1999.

[9] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, pp. 311–324, 2007.

[10] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," 2014.

[11] E. Tsironi, P. V. A. Barros, and S. Wermter, "Gesture recognition with a convolutional long short-term memory recurrent neural network," in *ESANN*, 2016.

[12] W. Lu, Z. Tong, and J. Chu, "Dynamic hand gesture recognition with leap motion controller," *IEEE Signal Processing Letters*, vol. 23, no. 9, pp. 1188–1192, 2016.

[13] *https://github.com/niharikabalachandra/Gesture-Recognition-Using-RNN*.

[14] C. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 10–24, 2018.

[15] L. Shen, J. Shi, B. Gong, Y. Zhang, Y. Dong, Q. Zhang, and H. An, "Multiple empirical kernel mapping based broad learning system for classification of parkinsons disease with transcranial sonography," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3132–3135, IEEE, 2018.

[16] H. Min, F. Shoubo, C. C. L. Philip, X. Meiling, and Q. Tie, "Structured manifold broad learning system: A manifold perspective for large-scale chaotic time series analysis and prediction," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1.

[17] Z. Liu, Z. Jin, and C. L. P. Chen, "Broad learning system: Feature extraction based on k-means clustering algorithm," in *International Conference on Information*, 2017.

[18] A. Samadani, "Gated recurrent neural networks for emg-based hand gesture classification. a comparative study," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1–4, IEEE, 2018.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, pp. 1735–1780, Nov. 1997.

[20] Y. H. Pao, G. H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163–180, 1994.

[21] K. S. Narendra and . Parthasarathy, K., "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 1, no. 1, pp. 4–27, 1990.

[22] C. P. Chen and J. Z. Wan, "A rapid learning and dynamic stepwise updating algorithm for flat neural networks and the application to time-series prediction," *IEEE Transactions on Systems Man and Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man and Cybernetics Society*, vol. 29, no. 1, pp. 62–72, 1999.

[23] W. Frank, B. Daniel, B. Rudak, and F. Denis, "Analysis of the accuracy and robustness of the leap motion controller," *Sensors*, vol. 13, no. 5, pp. 6380–6393, 2013.