ICIC Express Letters
Part B: Applications
Volume **11**, Number **3**, March **2020**

ICIC International ©2020 ISSN 2185-2766
pp. **229–236**

ROBUST NOISE FOR HUMAN ACTIVITY RECOGNITION USING CONVOLUTIONAL NEURAL NETWORK

Sujittra Sarakon[1], Thanathorn Phoka[1] and Kreangsak Tamee[1,2]

[1]Department of Computer Science and Information Technology
[2]Research Center for Academic Excellence in Nonlinear Analysis and Optimization
Faculty of Science
Naresuan University
Phitsanulok 65000, Thailand
sujittras61@email.nu.ac.th; { thanathornp; kreangsakt }@nu.ac.th

Received September 2019; accepted December 2019

Abstract. *Data from sensory devices such as an accelerometer changes continuously. The characteristics of fluctuating data are interesting, especially if we attempt to predict the results from this data, which may be corrupted or otherwise disrupted. Thus the quality of the data is uncertain. The uncertain quality of the data may be caused by data loss and other environmental problems. Researchers study the structure adjustments of the hyper-parameters of the Convolutional Neural Network (CNN) for human activity classification and test the persistence against noise data. In this study, raw smartphone data from WISDM LAB was used. Noise was removed from the data and the assessed model to an accuracy of 97.95%. The model was tested using a mixed noise trail ratio from 10%-50%. The results indicate that the accuracy of the model dropped only slightly.*
Keywords: Convolutional neural network, Noisy data, Time-series classification

1. **Introduction.** Over the past several years, papers concerning deep learning have been published. The current deep learning processes have become more active in various domain recognition tasks and can design and create more comprehensive recognition structures in each branch. In the same way, the importance of the information obtained is very important in creating a suitable learning method. In general, we tend to use large data to find hidden patterns within the data to create recognition patterns to solve problems. The most popular method of deep learning is Convolutional Neural Network (CNN or ConvNet), which is a type of neural network. It has been used in many types of classification techniques, especially image recognition. Starting from LeNet, the recognition architecture with CNN was conceived in 1998 [1] and later began to have an architecture built as well. CNN can be intended for specific applications such as in the medical field [2]. In each architecture, the structure optimized for time-series data. Signal data can be used to recognize and classify signals [3]. The structure or architecture of CNN is essential to the creation of effective learning due to the problem of reconstructing the CNN architecture. It is important to consider three main points: sparse interaction, parameter sharing, and equivalent representations.

There are many ways in which we can improve CNN architecture to be more efficient and suitable for that data, which causes the least errors and provides the highest accuracy, sometimes requiring experiments from relevant research [4] as a guideline for further architectural improvements. The general strategy of deep learning takes a long time to learn and experience to get an effective example. For example, we are finding additional information because deep learning requires learning from large data to be effective. Deep

DOI: 10.24507/icicelb.11.03.229

learning is similar to human learning in that it requires a lot of experience to work correctly and reduce errors.

In this study, we wanted to examine the depth of learning behaviour and the persistence of CNN's internal architecture network against noise data, because the data will be disrupted and corrupted at all times, especially as the data from the sensor is intended to recognize signs of human activity. In our work, a large raw data set was used to find the optimal parameter. The number of layers and activation functions are such a parameters, which were adjusted to find the minimum of processing time and resources. The paper is organized as follows. Section 2 reviews the related work. After that, Section 3 illustrates datasets. Then, Section 4 explains the experiment and demonstrate the result. Finally, the last section is to discuss the conclusion.

2. **Related Work.** Convolutional Neural Network (CNN) is another artificial neural network similar to general neural network. It simulates the behavior of the human brain with multiple neurons connected by neurotransmitters for learning the information and classification. That can be performed efficiently. In [5] Lecun and Bengio described a CNN process it is consisting of a collection of layers convolutional layers. There is an extraction important feature in an image, such as edges or lines in the first layer of the network, then the deeper layers will involve the simple features into more complex features.

Research by [6] studied time series classification using Convolutional Neural Networks. Multivariate is a group of data collected at the same time. The results of the study show the importance of the multi-channel deep CNN. The model is mostly constant, although the large data training set shows that the effects of sliding time steps found at 8. The model of 8 time steps can clearly represent the optimal to feed in process within the CNN structure and more accurate than a step higher. Because they split the data set so, the data are different in each time step. The convolutional neural networks consider many hyper-parameters to create a structure for better learning [7]. They designed a network for time-series data and proposed two approaches to increase the size of the training sets. Moreover, they can improve classification performance. It has been shown that the standard back-propagation process used to learn model. They chose the Rectifield Linear Unit (ReLu) as the activation function to avoid the gradient vanishing situation.

Human activity recognition has been tested widely in many kinds of research. From research [8] three public domain datasets based on the wearable sensor were studied in the detection of human activities. They designed the CNN and restructuring of the existing structure which includes two convolutional layers and two subsampling layers. They claim that architecture reaches the best performance, which means high accuracy and less time to compute. They explain that, if the network is too flat, high-level features are not able to be learned. On the other hand, if the network is too deep, valuable features may be filtered out in the convolutional and subsampling processes.

Ronao and Cho [9], studied deep learning with one-dimensional CNN using smartphone sensor data. They propose tuning the hyper-parameter of CNN. They initially numbered the filters from 10 to 200. Their model obtained initial accuracy values rising from 90% and higher, respectively, with CNN in two and three layers. When increasing the number of layers, the initial accuracy was higher than 95%. From these experiments, not only the number of network layers and the pooling size affected feature extraction to find variations in each activity to enhance an activity classification. More complicated features derived from every additional layer were found. Nevertheless, variations in the level of complexity in the next layers decrease as the data moves up to the top network layers.

In this regard, they [10] proposed the importance of dividing sensor data into the smaller data units from the beginning and the end point of each activity and sliding windows or windows times step. This research represents the large window sizes do not necessarily convert into a better recognition performance. Research [11] has presented an understanding of dropout that has been known to help large models and avoid overfitting, which is illustrated by avoiding co-adaptation. In their work, they raise an explanation that dropout works as an effective optimization process to generate more gradient information flowing through in layers. It depresses the network towards saturation areas of the nonlinear activation function, e.g., tanh and relu. Therefore, a study from previous research in this section conducted to fine-tune the hyper-parameter of the networks to the most suitable model. The next section will explain the research described in this article.

3. **Material and Datasets.** In this paper, the researchers wanted to improve the efficiency of deep learning by using Human Activity Recognition data (HAR) from WISDM Lab [12,13]. The data has been collected from acceleration sensors (Accelerometer) and stored on smartphones. The data consists of 3 axes (x, y, and z axes), consisting of six main activities: sitting, jogging, walking, upstairs, downstairs, and standing. This gave a total of 1,098,203 items of data which were collected from 36 people and classified. It was used to predict these six activities. That is a large data source as shown a percentage ratio for the six activities as shown in Figure 1.

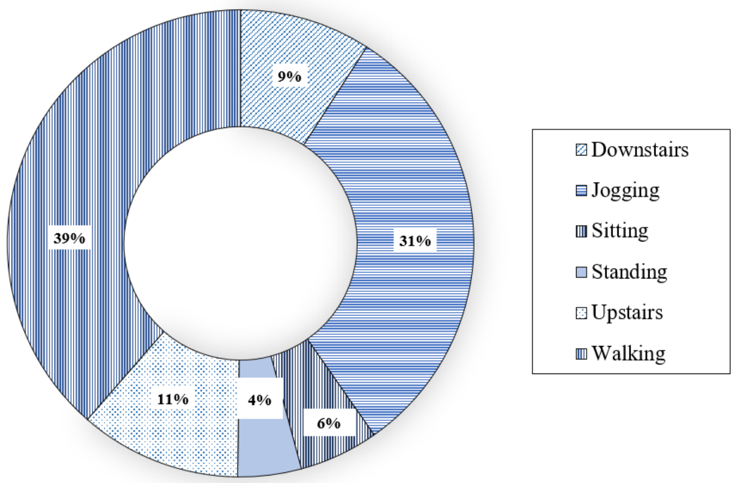

Figure 1. The percentage of data set by activity

In all six activities, we plotted the data on 3 axes (x, y, z axes) to see the trend of each activity. The data collected from a sample rate of 20 Hz (1 sample every 50 milliseconds). Therefore, we considered 10 seconds for each activity, as illustrated in Figures 2(a)-2(f). The z-axis will move forward in any direction, the y-axis will move vertically, and the x-axis will be horizontal movement. For the walking activity, the acceleration of the data is slower than for jogging. We can see that in Figures 2(c)-2(d) the y-axis and z-axis of the jogging movement is stronger than for walking. Figures 2(a)-2(b) show the activity of walking upstairs and downstairs. The y and z-axes of the upstairs movement indicate a slower movement than for the downstairs movement. There is a period up to and down from the top edge to the lower edge. Each period is longer than for the downstairs movement. In Figures 2(e)-2(f), sitting and standing are activities that do not change the acceleration rate when compared to other activities.

4. **Experiment Setup and the Main Result.** In this section, we present our experimental planning in CNN by tuning the hyper-parameter as in Table 1. In this work, we choose an activation function for all models as Rectified Linear Unit (ReLu). The batch

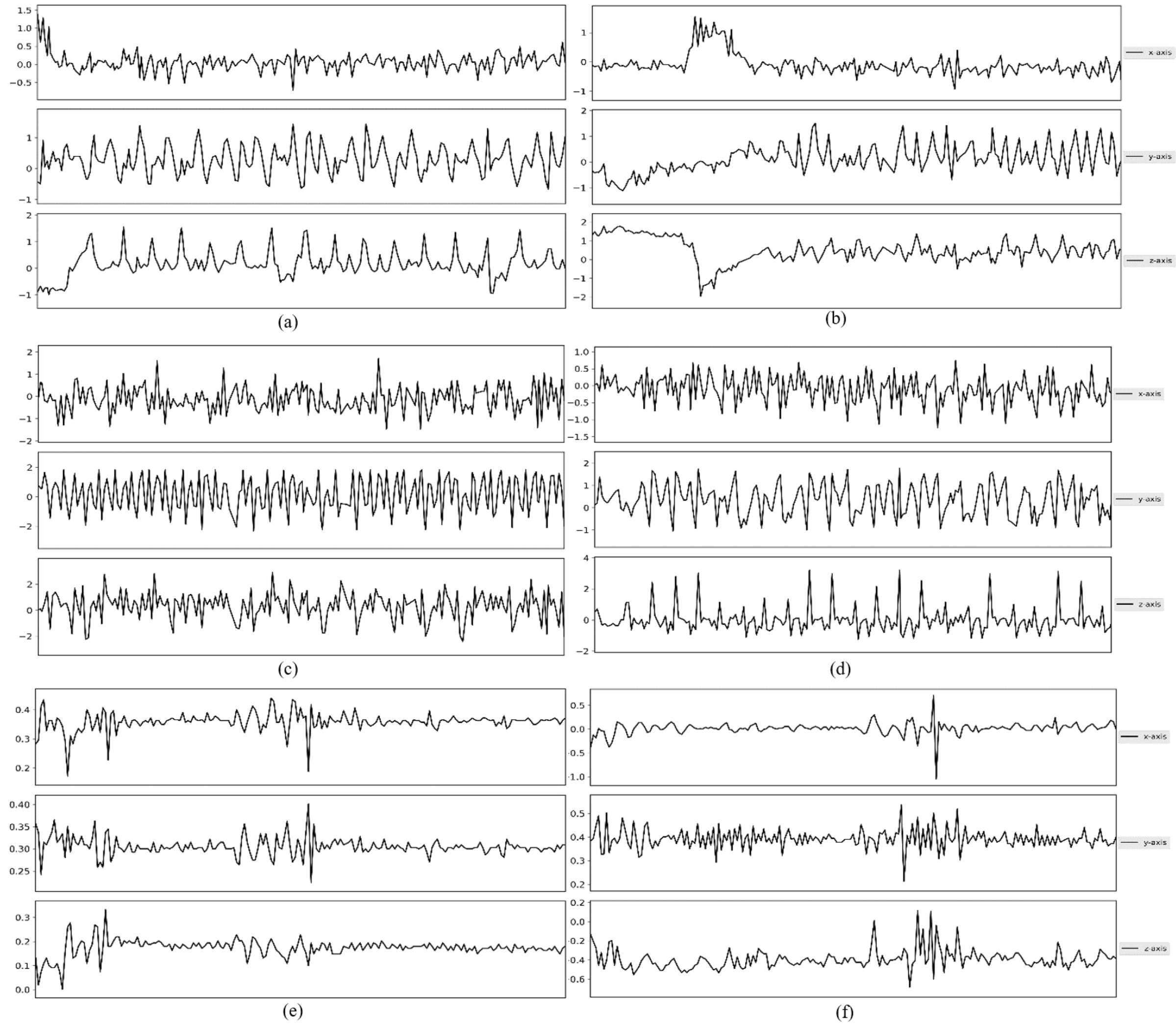

FIGURE 2. Graphs of the tri-axes for upstairs (a), downstairs (b), jogging (c), walking (d), sitting (e) and standing (f)

TABLE 1. Hyper-parameter tuning

Parameter	**Value**	**Parameter**	**Value**
CNN filter layer1	10	Dropout	0.5
CNN filter layer2	10	Max-pooling size	3
Kernel size	10	Flatten layer	TRUE
Batch size	8-128	Nodes of fully connected layer	100
Times step	8-128	Activation function	ReLu

size adjustment affects the convergence speed [14]. Because we use Intel I3-8100 4-cores for the CPU and 8 GB for the memory in this experiment, if the batch size increases, the computing times were increased and memory usage increased. Therefore, we tried to adjust the batch size equal to the time step. All the hyper-parameters in Table 1 can affect CNN and therefore human recognition. The input data is the value of x, y and z from the accelerometer and the output or classes of the label are the six activities of this experiment.

We start from the size of the window of the input data as 256 and training with 50 epochs for all models. The model in Figure 3 of one-dimension convolution layers followed by a max-pooling layer, followed by a second convolution layer and a second max-pooling

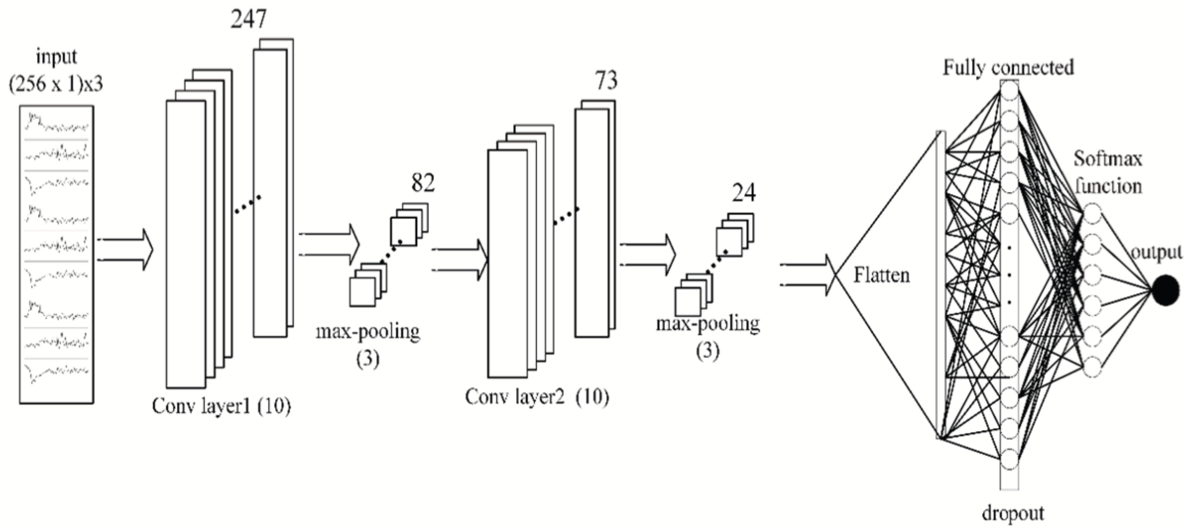

FIGURE 3. The CNN architecture for classification of the six-activities

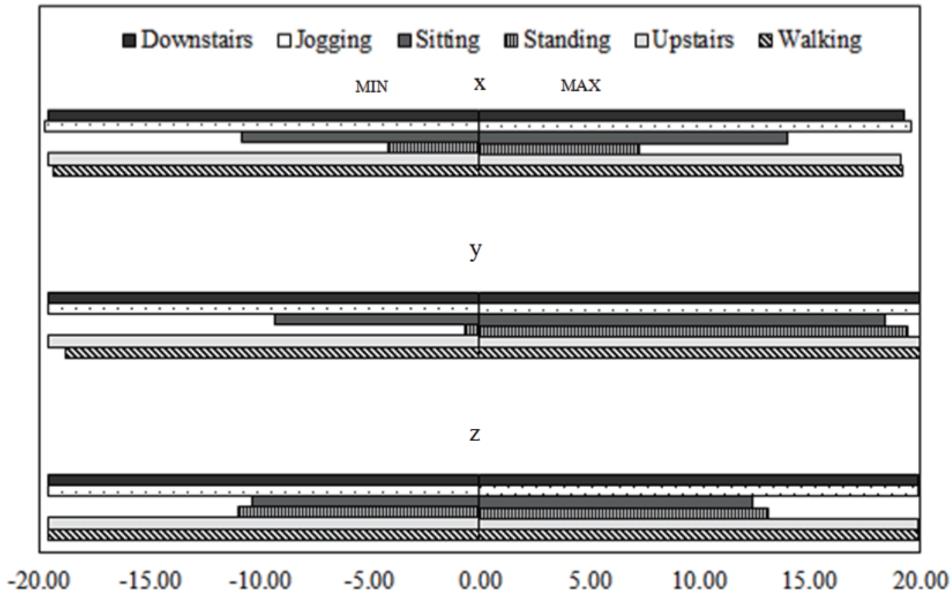

FIGURE 4. Graphs of data range along the x, y and z axes of the six-activities

layer. This is followed by a fully connected layer at the side of the dropout at 50% and a softmax function. The Adaptive Moment Estimation (ADAM) was chosen as the optimizer [15].

After the hyper-parameter tuning is complete, we found that the performance model on the accuracy rate of 98.74% on tested data, at the time step 8 and the batch size of 8. The accuracy of the result was usable. That is close to the higher batch size and found that the minimum training time used to test the noise in the next step. When the best model as obtained was chosen, the performance of the CNN structure on the data was studied. This performance was disturbed by the raw data to suffer noise with the two methods. First of all, we put the value 0 into the data of three coordinate axes by randomly entering the data set without a missing value. Secondly, we randomly entered the numeric in the range of all three coordinate axes of the main six activities with uniform random filled with random floats sampled, as shown in Figure 4. It can be seen that the activity has a slightly different data range in the walking, jogging, upstairs, and downstairs activities. On the y-axis the vertical of standing and sitting is different. From previous research where [16-18], experimented with image data to test the network structure stability, they founded that if the structure is appropriate for the data set to be resistant to noise.

Our interest is in experimenting with adding noise data using two methods starting from 10 to 50 percent of all datasets. Noise data was placed into the x, y and z-axes, with an increase of 10 percent at random from all data sets. The result is shown in Figure 5, where it was found that the accuracy in the first method is the random input of the value 0 at model 10 percent from graph Figure 5(a). It was seen that efficiency could be maintained at an accuracy of more than 90% and gradually adjusting to each epoch. It is better than all the other models, even though the error rate has highly oscillated, but the trends show the convergence to the right fit model. Figure 5(b) in a 50-percentage model has the highest error rate compared to other models since it has a higher noise in the model. In the second method, random numbers in the data range for all three axes of each activity are assigned and entered into the computer program and compared with the first method, above. When considering the correctness of Figure 5(c) in the 10 percentage model and the 20 percentage model, the accuracy rate is high at more than 95%. If increasing the number of epochs, which are different from the 50 percent model it is not as accurate as it has the most anomalies in the data. If considered in Figure 5(d), it will be seen that there is the highest error rate in comparison with model 10 and 20 percent as consistent with reality. The results of this experiment are evaluated with a test set of both methods for all models, as shown in Table 2. An accuracy rate of the based model (without missing value) is 97.95%. When compared with the base model, we founded the noise data of the two methods on the accuracy rate in the 10 percent model gradually decreases to 50 percent models with the least accuracy rates.

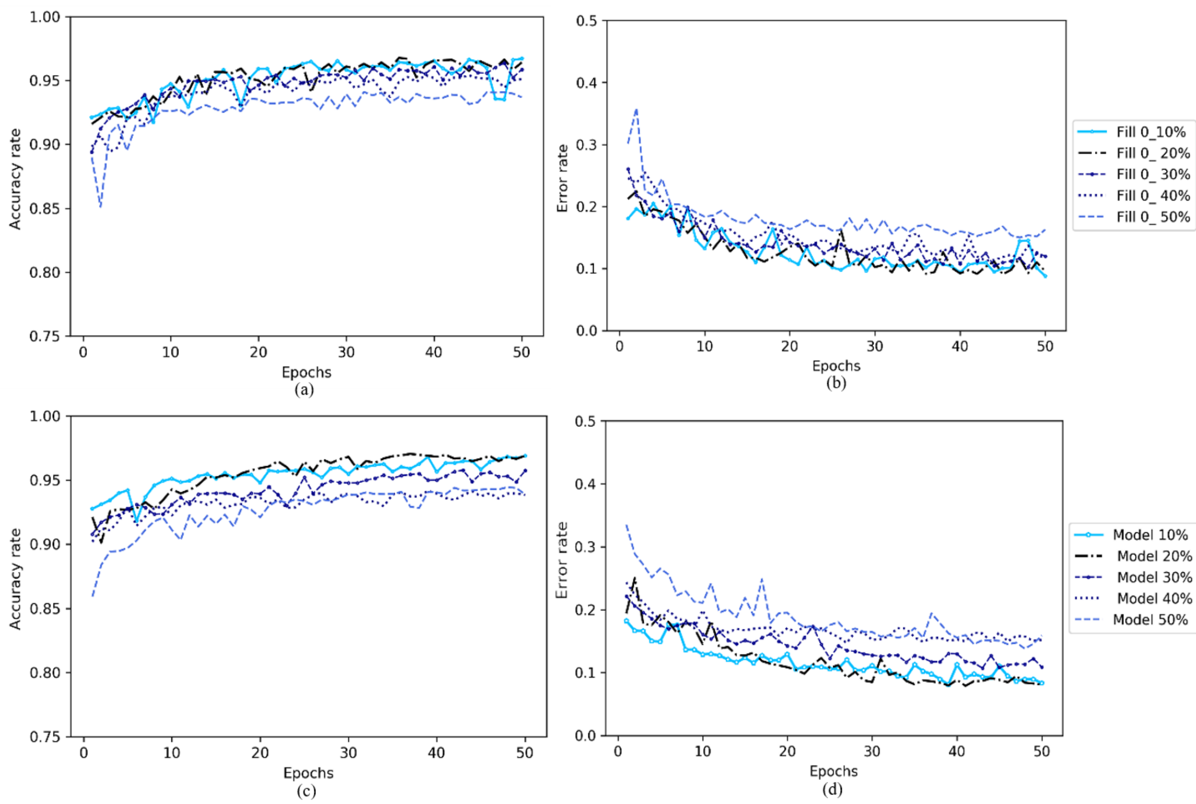

FIGURE 5. A graph of adding noise data model for accuracy rates (a) and error rates (b) of the first method, while the accuracy rates of (c) and the error rates (d) of the second method

Table 2. The accuracy rate of full zero value and a random numeric model on the test set compared with the base model

Model	**Random numeric**	**Full zero**	**Base model (no missing value)**
	Accuracy		
10%	96.35%	96.67%	97.95%
20%	96.88%	96.60%	
30%	96.03%	95.68%	
40%	94.15%	95.45%	
50%	94.10%	94.10%	

5. **Conclusion.** In this paper, a study of the CNN structure for the persistence of noise in the datasets is discussed. We use the data from WISDM for the classification of six human activities to tune the hyper-parameters of the optimal CNN structure. After this, an experiment was set up to study two types of noise, namely the addition of missing random zero and random numbers in the range of the activity, which has a filling ratio from 10% to 50%. This is compared to the model that has no missing value. We found that it is still resistant to disturbing data with more than 90% accuracy with the worst case at approximately 95%. In reality, the sensor data is faulty, even if it has been cleaned before entering the model. It should be reorganized to accommodate the wrong information, which has a high chance of occurrence. Time-series data requires high levels of accuracy and prediction. In future studies, the activation function, loss function, optimizer, the time step can be adjusted to reduce complexity. Moreover, batch size is an adjustable hyper-parameter. If we have a greater environment such as GPU memory [19], we can try larger batch sizes for all models.

Acknowledgment. This research was partially supported by The Royal Golden Jubilee (RGJ) Ph.D. Programme. Data for this research come from WISDM Lab by Department of Computer and Information Science, Fordham University, Bronx, NY. Especially, we appreciate Mr. Kevin Mark Roebl of the Division of International Affairs and Language Development for their checking and editing of English grammar and expression in this article.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, Gradient-based learning applied to document recognition, *Proc. of the IEEE*, vol.86, no.11, pp.2278-2324, 1998.
[2] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan and H. Adeli, Deep convolutional neural network for the automated detection and diagnosis of seizure using EEG signals, *Comput. Biol. Med.*, vol.100, pp.270-278, 2018.
[3] A. Subasi, M. Yilmaz and H. R. Ozcalik, Classification of EMG signals using wavelet neural network, *J. Neurosci. Methods*, vol.156, nos.1-2, pp.360-367, 2006.
[4] H.-C. Shin et al., Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning, *IEEE Trans. Med. Imaging*, vol.35, no.5, pp.1285-1298, 2016.
[5] Y. LeCun and Y. Bengio, Convolutional networks for images, speech, and time-series, *Handb. Brain Theory Neural Netw.*, vol.3361, no.10, 1995.
[6] Y. Zheng, Q. Liu, E. Chen, Y. Ge and J. L. Zhao, Time series classification using multi-channels deep convolutional neural networks, *International Conference on Web-Age Information Management*, pp.298-310, 2014.
[7] A. L. Guennec, S. Malinowski and R. Tavenard, Data augmentation for time series classification using convolutional neural networks, *Adv. Anal. Learn. Temporal Data*, vol.30, 2016.
[8] W. Jiang and Z. Yin, Human activity recognition using wearable sensors by deep convolutional neural networks, *2015 ACM Multimedia Conference (MM'15)*, Brisbane, Australia, pp.1307-1310, 2015.

[9] C. A. Ronao and S.-B. Cho, Human activity recognition with smartphone sensors using deep learning neural networks, *Expert Syst. Appl.*, vol.59, pp.235-244, 2016.

[10] O. Banos, J.-M. Galvez, M. Damas, H. Pomares and I. Rojas, Window size impact in human activity recognition, *Sensors*, vol.14, no.4, pp.6474-6499, 2014.

[11] S. Hahn and H. Choi, Understanding dropout as an optimization trick, *arXiv.org*, arXiv:1806.09783, 2018.

[12] J. R. Kwapisz, G. M. Weiss and S. A. Moore, Activity recognition using cell phone accelerometers, *SIGKDD Explorations*, vol.12, no.2, pp.74-82, 2010.

[13] J. W. Lockhart, G. M. Weiss, J. C. Xue, S. T. Gallagher, A. B. Grosner and T. T. Pulickal, Design considerations for the WISDM smart phone-based sensor mining architecture, *Proc. of the 5th International Workshop on Knowledge Discovery from Sensor Data*, pp.25-33, 2011.

[14] S. L. Smith, P.-J. Kindermans, C. Ying and Q. V. Le, Don't decay the learning rate, increase the batch size, *ArXiv Prepr.*, ArXiv171100489, 2017.

[15] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, *arXiv.org*, arXiv:1412.6980, 2014.

[16] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev and R. Fergus, Training convolutional networks with noisy labels, *arXiv.org*, arXiv:1406.2080, 2014.

[17] O. Yalcinkaya, *Training CNNs with Noisy Data*, https://ozgeyalcinkaya.com/wp-content/uploads/2017/11/Training-CNNs-with-Noisy-Data.pdf?189db0&189db0, Accessed on 04-Jun-2019.

[18] A. Vahdat, Toward robustness against label noise in training deep discriminative neural networks, *arXiv.org*, arXiv:1706.00038v1, 2017.

[19] J. Fu, Y. Huang, J. Xu and H. Wu, Optimization of distributed convolutional neural network for image labeling on asynchronous GPU model, *International Journal of Innovative Computing, Information and Control*, vol.15, no.3, pp.1145-1156, 2019.