Convolutional Neural Networks (CNN) Based Human Fall Detection on Body Sensor Networks (BSN) Sensor Data

Ali Haider Fakhrulddin
School of MAAE
Faculty of EEC
Coventry University
Coventry, UK
fakhrula@uni.coventry.ac.cuk

Xiang Fei, Hanchao Li
School of CEM
Faculty of EEC
Coventry University
Coventry, UK
x.fei@coventry.ac.uk, lih30@uni.coventry.ac.uk

Abstract— **According to the World Health Organization, around 28-35% of people aged 65 and older fall each year. This number increases to around 32-42% for people over 70 years old. For this reason, this research targets the exploration of the role of *Convolutional Neural Networks(CNN)* in human fall detection. There are a number of current solutions related to fall detection; however, remain low detection accuracy. Although CNN has proven a powerful technique for image recognition problems, and the CNN library in Matlab was designed to work with either images or matrices, this research explored how to apply CNN to streaming sensor data, collected from Body Sensor Networks (BSN), in order to improve the fall detection accuracy. The idea of this research is that given the stream data sets as input, we converted them into images before applying CNN. The final accuracy result achieved is, to the best of our knowledge, the highest compared to other proposed methods: 92.3%.**

Key words: Convolutional Neural Networks (CNN), Internet of Things (IoT), Body Sensor Networks (BSN), Telecommunication Systems Team (TST), Support Vector Machines (SVM), Hidden Markov Model (HMM), Radio Frequency Identification (RFID), Kinect Activity Recognition Dataset (KARD), Cornell Activity Dataset (CAD), short term Fourier transformation (STFT), Gramian Angular Fields (GAF), Markov Transition Fields (MTF), Stereotypical Motor Movement Detection (SMM)

I. INTRODUCTION

In the modern era, technological scenario such as *Internet of Things* (IoT) has taken over several tasks that are performed in scientific research and the advances in associated data mining and machine learning techniques have exhibited significant impacts on our society. One example, among others, is the human activity recognition, especially fall detection, for remote health monitoring (Igual et al., 2013). According to the World Health Organization, around 28-35% of people aged 65 and older fall each year. This number increases to around 32-42% for people over 70 years old (Igual et al., 2013). Currently there are mainly two sources of data for human activity recognitions. One is BSN sensors, such as accelerometers or other inertial devices; the other is RGB-D camera, namely the Microsoft Kinect (Gaglio et al., 2015). While RGB-D sensors are cost effective and vision-based human action recognition continues to advance, the recognition performance is subject to various challenges such as occlusion, camera position, subject variations in performing actions, background clutter, etc. In addition, vision-based approaches are applicable to a limited field of view or a constrained space defined by the camera position (Chen et al., 2015). In contrast, BSN sensor technologies, although being intrusive, sensor location sensitive and subject to sensor drift, enable much wider field of views and is insensitive to lighting conditions (Chen et al., 2015).

There have existed several BSN based solutions to human activity recognition including fall detection. However, as mentioned later in Section II, the detection accuracy of these mechanisms are not high enough (less than 90%). CNN is a deep learning model that is generically comprised of one or more convolutional layers followed by one or more fully connected layers (multilayer neural network). The convolutional layers extract features while the multilayer neural network part performs classification based on the learned features (UFLDL, n.d.). CNN has proven not only a powerful machine learning technique for a wide range of problems such as object recognition, speech processing, and affect recognition, but also easier to train as it has many fewer parameters than fully connected networks with the same number of hidden units (UFLDL, n.d.). Due to the distinct learning capabilities of CNN, it has been applied to vision based human activity recognition, i.e. the recognition is on Kinect RGB-D data. This research targets the exploration of the role of CNN in detecting falls, especially for the elderly generation, given BSN sensor data streams.

The research emphasizes the design of a fall detection system with the help of CNN in the dataset. Rich colour and texture of CNN is a remarkable trait that supports its use in detection of fall using dataset. Moreover, use of CNN and datasets has shown the ability to deliver good performances. In in-depth pictures, there is a powerful triggering of the border form. The use of CNN in the public dataset would be utilized in the research. The drawback of this system has also been reported, and that is sluggishness of fall detectors while computing and the same would be studied.

The remainder of this paper is structured as follow. The second section presents key literature in the area of fall

978-1-5386-1107-4/17/$31.00 ©2017 IEEE

detection. The third section explains the methodology followed, the datasets used, alongside the coding process and time series conversions to images. The fourth section discusses the results and presents the analysis and the overall comparison of CNN to other methods used by other researchers. The final section presents potential future works.

II. Literature Review

Hwang et al. (2017) proposed a deep learning approach in order to maximise the accuracy of fall detection systems. The proposed system makes use of depth cameras that will be put in place in houses and/or nursing homes that will film videos and send them to a local computer that will resize them and generates an alarm in case a fall detection is identified. To test this system, a fall detection dataset containing 264 actions from 8 different categories representing all daily activities from the *Telecommunication Systems Team (TST*) was used. The testing consisted of 5 random trials where 240 videos for training were extracted. The preliminary results have shown that the system's accuracy lies between 69.9% and 78.8% in all trials. Nonetheless, after data increase was applied, the performance drastically improved as it achieved 92.4 to 96.6% accuracy. Although the proposed model looks promising, these are only preliminary results and the system has not been tested thoroughly (e.g., for the effectiveness of data augmentation). Also, as it is based on depth camera, it shares the limitation of vision-based human action recognition mentioned in section I.

Gaglio et al. (2015) proposed system's main aim is to automatize the different activities-based on the postures. For that matter, the system was decomposed into three major components: the features detection, the posture analysis, and the activity recognition. Each of these components has a specific task. The features detection component is responsible for the different strategic points' detection that will be used later to determine the body posture. The posture analysis component is responsible for the classification of the postures-based on K-means and the *Support Vector Machines (SVM)*. Last but not least, the activity recognition component is carried out by using the *Hidden Markov Model (HMM)*. As a case study, a prototype of this aforementioned system was implements at the Networking & Distributed Systems laboratory at the University of Palermo, Italy. To run the test, *Radio Frequency Identification (RFID)* readers were installed at the doors. This was used to provide information about whether there is a human presence or not. Furthermore, software sensors, including Kinect, were used to detect the different activities. The collected data set was called *Kinect Activity Recognition Dataset (KARD)* and contained a number of diverse activities that were categorized-based on whether they are gestures or actions. On another note, to capture the Kinect stream, the system was implemented using Java. The results have proved that the system is fully functional and efficient as it was able to capture all activities. Moreover, by testing the model using the *Cornell Activity Dataset (CAD)*, a precision of 77.3% and a recall of 76.7%. Although this paper presents very good algorithms and techniques, some scenarios were not taken into account. The 18 activities considered in this study are not enough; more activities should have been considered. The precision is not high enough?

Gaglio et al. (2015) has also presented a method to recognize the human activities via using an unobtrusive motion section, more specifically the Microsoft Kinect, an RGB-D camera. The human activities studies in this research are thought of as a set of joints connecting body parts; for instance, the arms and the legs. The main objective behind this research is to identify the different human postures and recognize them using machine learning tools and techniques. Furthermore, another important objective of this paper is the three contributions it brings.

1. The activity recognition method; this should have an acceptable accuracy while maintaining a low power consumption. Also, it should make use of real time processing and provide real-time data as output.

2. Releasing the KARD to the general public. This includes a number of activities, 18 to be more specific, and the corresponding actions and gestures that it was broken down into.

3. Validating the proposed method.

Goodwin et al. (2014) looked at a number of wireless accelerometers and design recognition systems that sense SMM automatically in people that have autism, using fluctuating methods applied and outcomes attained. The system consists of three-axis accelerometers that sample at 100Hz, Westeyn and classmates realised 69 percent of hand flapping activities applying Hidden Markov Models; nonetheless, data were taken from fit people imitating behaviours. The research never observed people with autism. From four varied, distinct researches, Min and classmates report gathering an aggregate of forty hours of three-axis acceleration information tested at 50Hz from the wrists and torso of four people who suffered from autism. Applying a number of diverse characteristics and semi-administered organization methods, they attained highest recognition degrees of 86 percent for hand; nevertheless, correct positive and untrue positive degrees were never sufficiently defined, and autonomous duplications outside their trial of four were never recounted. Amongst the three researches, Goncalves and classmates established 5 people that suffered from autism for numerous ten minute times whereas wearing a commercially accessible 1GHz three-axis accelerometer on their right hand and accomplished 76 percent automatic recognitions of hand flapping activities with an agreeing 24 percent incorrect positive degree. Goodwin and classmates gathered three-axis acceleration content tested at 60Hz from the wrists and torso of 6 people who suffered from autism frequently experimented in both workshop and schoolroom situations. Applying a five time and occurrence domain characteristics and a C4.5 decision tree classifier; they attained an average automatic hand flapping and body shocking recognition degrees of 89.5 percent in the workroom and 88.6 percent in the schoolroom. Lastly, Ploetz and classmates used on body accelerometers and pattern recognition classifiers to sense automatically severe behaviours such as aggression, disruption, and self-injury in people with developing incapacities; nevertheless, information in this research was taken from skilled professionals acting out severe behaviour occurrences, and SMM were never included.

In terms of CNN based time series classification, there exist several mechanisms. They can be categorized into two groups. The first group takes the time series data as 1-D grid. For example, (Zheng et al., 2014; Zheng et al., 2016) separated multivariate time series into univariate ones and performed feature learning on each univariate series individually. The second group tries to convert 1-D time series data into 2-D representations. Some examples are (Abdul-Hamid et al., 2013) transformed audio signals to time-frequency domain before been fed into CNN. (Gong et al., 2016) calculated each data piece of inertial body sensor data to generate the multichannel *short term Fourier transformation (STFT)* spectrograms as the input of CNN. In addition, (Wang and Oates, 2015) propose a framework to encode time series data as different types of images, namely, *Gramian Angular Fields (GAF)* and *Markov Transition Fields (MTF)*. While GAF images are represented, in polar coordinates, as a Gramian matrix where each element is the trigonometric sum between different time intervals, MTF images represent the first order Markov transition probability along one dimension and temporal dependency along the other. Our proposal, as illustrated in section III, falls into the second group, but compared to existing mechanisms, is simple and straightforward.

III. METHODOLOGY

Primary and secondary research were both carried out for this project. The secondary research takes the form of literature review that has been presented in *section 2*. In this section, the researcher discusses the methodology used regarding the setups of the experiments carried out as primary research. This consists of the choice of deep learning mechanisms and the way content is pre-processed before training. Furthermore, this section describes the type of section proposal algorithms that have been designed as an option to the descending window technique. The alternative of method in this regard is a vital component for the presentation of the CNN. It may be perceived as a fragile classifier applied in a cascade with a CNN. Lastly, the researcher illustrates the prototype that have been used to train the CNN.

In this research, the initial data is the readings of two accelerometers: time series for each situation (falling and not falling) (Gasparrini et al., 2016). These time series represent the coordinates of the wearable device at each time point. CNN in Matlab are designed to work with images or matrices (Zhao et al., 2017). Therefore, the researcher needs to convert the time series into images. The conversion steps are as follows:

i. pre-process all the series so that they are of equal size

ii. Scaling of values in the interval 0..255 for 8-bit integer image format

The formula that have been used: $255 \times \frac{x_i - min}{max - min}$

iii. Getting matrices for each coordinate

iv. Convert matrices to grayscale images

v. Combine grayscale images to one RGB image

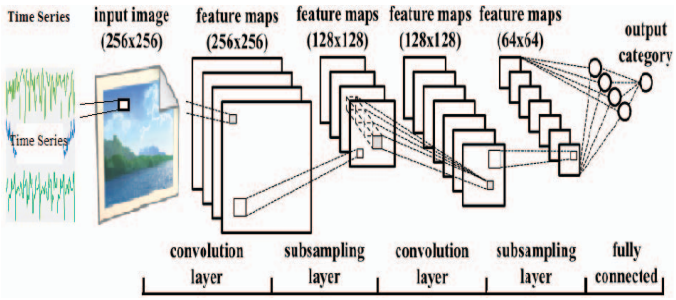


Figure 1: Sample of the System Design (Fakhrulddin, 2017; Cong and Xiao, 2014)

The images produced are shown below

Figure 2: Non-Fall Images

Figure 3: Fall Images

Therefore, the researcher gets the image as a representation of accelerometer data. The researcher has two accelerometers and gets two images of a same size for each of them. These images are combined to one image. Thus, the researcher gets image with sizes l and 2*l. the images are saved to folder "1" for "Fall" cases, and to folder "0" for "" Non-Fall" cases.

Then researcher creates a CNN layers. The researcher creates two convolution layers with 12 5x3 filters in first and 16 3x2 filters in second. In addition, the researcher creates one intermediate fully connected layer with 16 outputs. It should be noted that the researcher can change structure of the CNN for enhancing prediction accuracy (Shao et al., 2014).

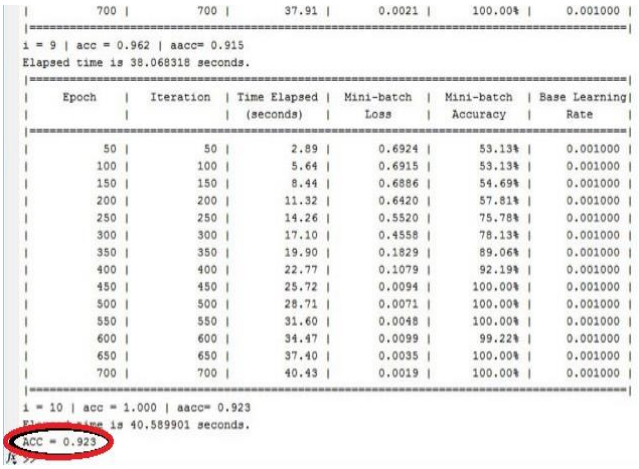
Moreover, the researcher splits the training dataset on training and test sets. The researcher can do it randomly in proportion 90% and 10% for each situation (fall and non-fall). Therefore, the researcher gets 238 training samples and 26 test samples. But one splitting is not significant especially when we have small size dataset, as in our case. The 264 is a very small dataset size for CNN good training. Therefore, I must repeat data splitting and training for many random splits and calculate mean of accuracy which is a more representative.

The system that the researcher proposes in this research follows three phases such as pre-processing, characteristics mining and sorting (Ordóñez and Roggen, 2016). Pre-processing is comprises of down sampling and window system which is characterised by feature mining comprising of autoregressive moving average, and finally, sorting is done using support vector machines and a pattern net neural network. The researcher uses CNN because they have

comparatively cheaper calculation involvedness and swiftness and because they have been proven in previous research works (Hemmatpour et al., 2016).

IV. Expermints and Analysis

In this section, the researcher discusses the fall detection using CNN with dataset testing strategy. The researcher also introduces the dataset and test results and compares it with existing methods. Likewise, the researcher carried out a full analysis and discussion of the obtained test results.



```
|         700 |         700 |       37.91 |      0.0021 |     100.00% |     0.001000 |
|=============================================================================================|
i = 9 | acc = 0.962 | aacc= 0.915
Elapsed time is 38.068318 seconds.
|=============================================================================================|
|     Epoch   |   Iteration  | Time Elapsed |  Mini-batch  |  Mini-batch  | Base Learning|
|             |              |   (seconds)  |     Loss     |   Accuracy   |     Rate     |
|=============================================================================================|
|          50 |          50 |         2.89 |      0.6924 |      53.13% |     0.001000 |
|         100 |         100 |         5.64 |      0.6915 |      53.13% |     0.001000 |
|         150 |         150 |         8.44 |      0.6886 |      54.69% |     0.001000 |
|         200 |         200 |        11.32 |      0.6420 |      57.81% |     0.001000 |
|         250 |         250 |        14.26 |      0.5520 |      75.78% |     0.001000 |
|         300 |         300 |        17.10 |      0.4558 |      78.13% |     0.001000 |
|         350 |         350 |        19.90 |      0.1829 |      89.06% |     0.001000 |
|         400 |         400 |        22.77 |      0.1079 |      92.19% |     0.001000 |
|         450 |         450 |        25.72 |      0.0094 |     100.00% |     0.001000 |
|         500 |         500 |        28.71 |      0.0071 |     100.00% |     0.001000 |
|         550 |         550 |        31.60 |      0.0048 |     100.00% |     0.001000 |
|         600 |         600 |        34.47 |      0.0099 |      99.22% |     0.001000 |
|         650 |         650 |        37.40 |      0.0035 |     100.00% |     0.001000 |
|         700 |         700 |        40.43 |      0.0019 |     100.00% |     0.001000 |
|=============================================================================================|
i = 10 | acc = 1.000 | aacc= 0.923
Elapsed time is 40.589901 seconds.
ACC = 0.923
```

Figure 4: Result

Figure 4 above shows the results of the testing that the researcher carried out. The first column shows the Epoch parameters that the system produced during testing. The second column similarly indicates the iterations that were made by the system during testing stage. Time elapsed is shown in the third column, and mini-batch loss and mini-batch accuracy is shown by the fourth and fifth column respectively. Finally, the last column indicates the base learning rate that the system generated.

There are 10 tests as shown in Figure 12 (i = 10), and there are 700 epochs, 10 x the time of the 700 epochs it is training process. The researcher split initial dataset on 2 datasets training and testing. The researcher training CNN on training dataset and check the accuracy on test datasets, but the researcher need to do many iterations for good approach of accuracy.

Regarding previous studies conducted by a large number of researchers, such as Hwang et al. (2017), Gasparrini et al. (2016), and Gaglio et al. (2015) who emphasised on the detection of human falling from the ground, it has been noticed that the output accuracy of these studies reached a good percentage range of 70 ~79%. The results of this study have shown a significant improvement of the detection percentage; this implies that there are still methods to be investigated to enhance the before-mentioned detection. However, the outcomes of this research in a comparison to the previous studies are-based on a different method; CNN, that provides the result of 92.3% of people detection.

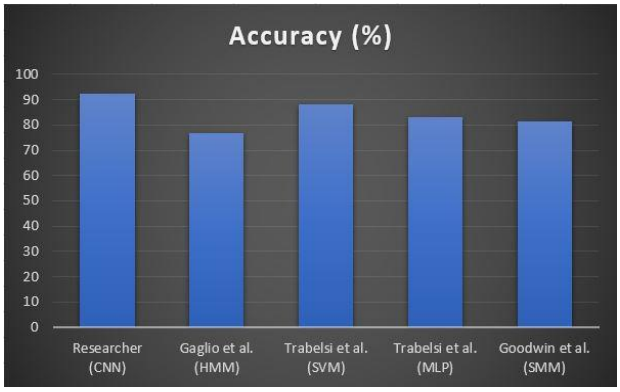


Figure 5: A Comparison between Researcher result and other Researchers

Elderly persons often fall and may lead to serious injuries and consequences. Fall detection systems are now a very common component to quickly detect any fall, allowing the family or health person to quickly assist the person that has fallen. Different methods have been presented to assist. In this research, the research proposes CNN. The researcher found it efficient and workable in fall detection. Other methods have been proposed by research including Artificial Neural Networks. This technique is expected to improve fall detection precision by evading the traditional threshold anchored fall detection approaches and inventing Artificial Neural Networks as an appropriate choice. The CNN has low computational costs, making it easy to execute on a moveable device and be comfortable to be wear.

V. Conclusion and Future Work

In this research, the researcher clearly indicated that CNN can be used to solve the pertaining fall detection problem. The dataset depiction that the researcher used varied in terms of frequencies which, given its little necessity for pre-processing, was taken as a low-level feature exemplification. The application of this approach for a convolutional design was reinforced by the ability of these prototypes to separately learn fundamental data from their training information, therefore projecting the entered data in higher-level feature spaces.

The obtained output is the highest compared to the other methods that have been used to detect falls. So, that the system robustness is evaluated, no fine tuning was carried out. The dataset that was used allowed the researcher to obtain an average score of 92.3%, which is an improvement on the previous system that were compared to (Camplani and Salgado, 2012). This result can potentially increase if no hardware limitations are encountered.

As for future work, other methods, other than CNN, are going to be used to test their efficiency in fall detection and check whether or the results gotten using CNN can be improved. In addition, other body postures, other than falls, will be closely studied in order to implement a powerful algorithm for detection purposes.

References

[1] Abdel-Hamid, O., Deng, L. and Yu, D. (2013). Exploring Convolutional Neural Network Structures and Optimization Techniques for Speech Recognition. INTERSPEECH, pp.3366-3370.

[2] Camplani, M., Salgado, L., 2012. Efficient spatio-temporal hole filling strategy for Kinect depth maps. p. 82900E–82900E–10. doi:10.1117/12.911909

[3] Chen, C., Jafari, R. and Kehtarnavaz, N. (2015). A survey of depth and inertial sensor fusion for human action recognition. Multimedia Tools and Applications, 76(3), pp.4405-4425.

[4] Cong, J. and Xiao, B. (2014). Minimizing Computation in Convolutional Neural Networks. Artificial Neural Networks and Machine Learning – ICANN 2014, pp.281-290.

[5] Gaglio, S., Re, G.L., Morana, M., 2015. Human Activity Recognition Process Using 3-D Posture Data. IEEE Trans. Hum.-Mach. Syst. 45, 586–597. doi:10.1109/THMS.2014.2377111

[6] Gasparrini, S., Cippitelli, E., Gambi, E., Spinsante, S., Wåhslén, J., Orhan, I., Lindh, T., 2016. Proposal and Experimental Evaluation of Fall Detection Solution Based on Wearable and Depth Data Fusion, in: ICT Innovations 2015, Advances in Intelligent Systems and Computing. Springer, Cham, pp. 99–108. doi:10.1007/978-3-319-25733-4_11

[7] Gong, Goldman and Lach (2016). Deepmotion: a deep convolutional neural network on inertial body sensors for gait assessment in multiple sclerosis*. 2016 IEEE Wireless Health (WH).

[8] Goodwin, M.S., Haghighi, M., Tang, Q., Akcakaya, M., Erdogmus, D., Intille, S., 2014. Moving Towards a Real-time System for Automatically Recognizing Stereotypical Motor Movements in Individuals on the Autism Spectrum Using Wireless Accelerometry, in: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '14. ACM, New York, NY, USA, pp. 861–872. doi:10.1145/2632048.2632096

[9] Hwang, S., Ahn, D., Park, H., Park, T., 2017. Maximizing Accuracy of Fall Detection and Alert Systems Based on 3D Convolutional Neural Network: Poster Abstract, in: Proceedings of the Second International Conference on Internet-of-Things Design and Implementation, IoTDI '17. ACM, New York, NY, USA, pp. 343–344. doi:10.1145/3054977.3057314

[10] Hemmatpour, M., Ferrero, R., Montrucchio, B., Rebaudengo, M., 2016. A Baseline Walking Dataset Exploiting Accelerometer and Gyroscope for Fall Prediction and Prevention Systems, in: Proceedings of the 11th EAI International Conference on Body Area Networks, BodyNets '16. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, pp. 81–85.

[11] Igual, R., Medrano, C., & Plaza, I. 2013. Challenges, issues and trends in fall detection systems. BioMedical Engineering OnLine, 12, 66. http://doi.org/10.1186/1475-925X-12-66

[12] Ordóñez, F.J., Roggen, D., 2016. Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. Sensors 16, 115. doi:10.3390/s16010115

[13] Trabelsi, D., Mohammed, S., Chamroukhi, F., Oukhellou, L., Amirat, Y., 2013. An Unsupervised Approach for Automatic Activity Recognition -based on Hidden Markov Model Regression. IEEE Trans. Autom. Sci. Eng. 10, 829–835. doi:10.1109/TASE.2013.2256349

[14] Ufldl.stanford.edu. (n.d.). Unsupervised Feature Learning and Deep Learning Tutorial. [online] Available at: http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/ [Accessed 2 Sep. 2017].

[15] Wang, Z. and Oates, T. (2015). Encoding Time Series as Images for Visual Inspection and Classification Using Tiled Convolutional Neural Networks. Association for the Advancement of Artificial Intelligence, pp.40-46.

[16] Zhao, R., Song, W., Zhang, W., Xing, T., Lin, J.-H., Srivastava, M., Gupta, R., Zhang, Z., 2017. Accelerating Binarized Convolutional Neural Networks with Software-Programmable FPGAs, in: Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, FPGA '17. ACM, New York, NY, USA, pp. 15–24. doi:10.1145/3020078.3021741

[17] Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L., 2014. Time Series Classification Using Multi-Channels Deep Convolutional Neural Networks, in: Web-Age Information Management, Lecture Notes in Computer Science. Presented at the International Conference on Web-Age Information Management, Springer, Cham, pp. 298–310. doi:10.1007/978-3-319-08010-9_33

[18] Zheng, Y., Liu, Q., Chen, E., Ge, Y., Zhao, J.L., 2016. Exploiting Multi-channels Deep Convolutional Neural Networks for Multivariate Time Series Classification. Front Comput Sci 10, 96–112. doi:10.1007/s11704-015-4478-2