A Multi-Scale Deep Learning Attention-based Feature Method for Rolling Elements Bearing Fault Detection in Industrial Motor Drives

Yannis L. Karnavas*, Spyridon Plakias†, Ioannis D. Chasiotis‡
Electrical Machines Laboratory
Department of Electrical & Computer Engineering
Democritus University of Thrace, 671 00, Xanthi, Greece
*karnavas@ee.duth.gr, †splakias@ee.duth.gr, ‡ichasiot@ee.duth.gr

***Abstract*—In the last decade, convolutional neural networks have achieved great success in the automated fault diagnosis of rotating equipment in electrical machines. However, the application of convolutional models encounters some challenges to deal with such as (i) the requirement of a vast amount of training data and (ii) the selection of the neural architecture, and particularly the sizes of the convolutional kernels that effectively extract features from the raw input signal. To alleviate the above challenges, we propose a deep learning network consisting of multiple independent densely connected convolutional streams with different sizes of kernels and of a simple attention mechanism that fuses the extracted features, producing a feature mapping with generalization and discrimination power. Simulation cases with a widely used bearing fault detection benchmark show the effectiveness of the proposed approach, especially in cases of a restricted amount of training samples.**

***Index Terms*—Attention mechanism, convolutional neural networks, electrical machines, fault detection, rolling elements bearing.**

I. Introduction

Rolling bearings are core elements of rotating electrical machines. Scientific studies state that bearing failures are the most common in modern complex systems, making the early fault detection (FD) of rolling bearings imperative. Indeed, a fast and accurate FD of rolling bearings can prevent disastrous equipment destruction, unexpected and sudden shutdowns, and economic losses.

Over the years, many methods have been applied successfully for bearing FD such as signal processing techniques and machine learning classification algorithm. Most approaches use vibration measurements from sensors attached to the bearings because of their discrimination potential and progress in sensor technology. Recently, Deep Learning (DL) based approaches [1] have restated the accuracy of bearing FD, stacking multiple nonlinear neural processing layers, and achieving the auto-extraction of features [2]. Drawbacks of DL methods are the vast amount of raw data, needed to build effective discrimination models and their intensive training in terms of computational complexity.

Convolutional Neural Networks (CNNs) and their variants are the most successful among DL models in bearing FD [3], [4]. The main reasons behind the impressive effectiveness of CNNs are (i) the extraction of location invariant features with the direct processing of raw data and (ii) the convolution operation as core process of CNNs uses weight sharing, resulting in fewer unknown parameters and so better generalization capability and robustness to over-fitting.

The number of training weights depends on the kernel size of the convolutional layer. So, a straightforward solution is to use small kernel sizes, reducing the number of unknown parameters and the computation complexity of the process. On the other hand, the kernel size of a convolutional layer determines the effective receptive field that is applied in the input signal and so plays a vital role in the effectiveness of the extracted features. So, the sizes of kernel filters are important hyper-parameters that need to be tuned, and their optimal estimation is difficult.

In the current research study, we apply multi-scale convolutional layers with different sizes of kernel filters. More particularly, we adopt three independent densely connected convolutional blocks with various kernel sizes. In that way, the local receptive field varies in each convolutional stream and so different parts of the input signal are processed. Moreover, we employ the densely connected architecture of CNNs [5] to enhance features diversity and reuse, ease the vanishing gradient problem, and substantially decrease the number of learning parameters.

In the sequel, the question that arises is the handling of the feature mappings, provided from the densely convolutional streams. The concatenation of the feature vectors increases the dimension of the final mapping, raising the number of learning parameters and causing over-fitting. Their addition results in a reduction of information and consequently in cut of performance. So, to merge effectively the feature mappings of the convolutional streams, we apply a simple attention mechanism, another successful architecture of DL [6], [7]. The latter exports the most valuable information of each mapping, processing and estimating the most informative feature keys. Attention mechanism has been employed before on rolling bearings FD [8], [9] but to analyze the temporal coherence of subsequent segments of the input vibration signal. To the best of our knowledge, this is the first time where multi-scale

978-1-6654-1847-8/21/$31.00 ©2021 IEEE

convolutional blocks with various kernel sizes are used in a FD task and the merging of the produced feature vector mappings are achieved with an attention mechanism.

The organization of the paper follows. Section II describes the proposed architecture with its components. The following section presents the simulation cases and analyses the results. Finally, we complete with conclusions and possible future work.

II. Architecture of the Proposed DL model

The proposed DL architecture adopts multi-scale convolutional processing of the raw vibration signal using three identical independent densely connected convolutional blocks (Fig. 1). The difference of the convolutional streams lies in the first convolutional layer and more particularly to its kernel size. So, the kernel sizes of the independent neural streams are 8, 16 and 32 accordingly. With the process of diverse receptive fields of the input, we achieve to extract multi-scale knowledge of the raw signal, capturing spatial dependencies of various resolutions and so enhancing the representation ability of the model.

Additionally, the use of multi-scale convolutional blocks is combined with less kernel filters in each stream. In that way, we achieve the increase of the representation ability of the model without a burden in the number of parameters and in the computation complexity.

By the use of the densely connected framework, we reinforce the reuse of features, accomplishing to derive features that belong to different levels of representation. As shown in Fig. 1, Batch Normalization is used to stabilize the learning process [10]. Also, we notice that in the forward processing of the densely connected convolutional stream, the ReLU activation function is used to deal with the vanishing gradient problem. However, the applied non-linearity in the final feature mapping is the hyperbolic tangent activation function (Tanh) to produce responses even for negative input values. In that way, we exploit the advantages of both activation functions.

Continuing, the handling strategy of the produced multi-scale feature mappings follows. By the fusing of the feature vectors in a concatenation mode, we retain information but we increase the dimension of the concatenated vector and so the number of the learning parameters, causing possibly over-fitting and finally performance reduction. On the other hand, by the element-wise addition of the feature vectors we lose information since we don't check the importance of each feature mapping in the discrimination procedure.

In the current study, a simple attention mechanism processes the multi-scale features f_i and outputs their weighted mean considering their discrimination capability (Fig. 2) [11]. The attention mechanism is made of a simple feed-forward network with one layer and one output neuron. The applied non-linearity in the output of the neural model is the Tanh. The network process each multi-scale feature mapping f_i and estimates the reward value e_i that represents the significance of the corresponding feature. A following softmax layer normalizes the reward values into a probability distribution.

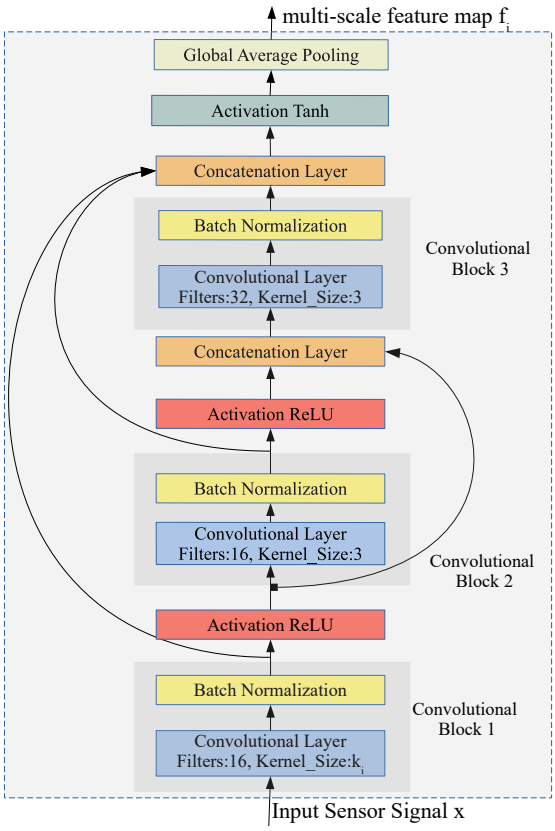

Fig. 1. Densely Connected Convolutional Stream (DCCS$_i$).

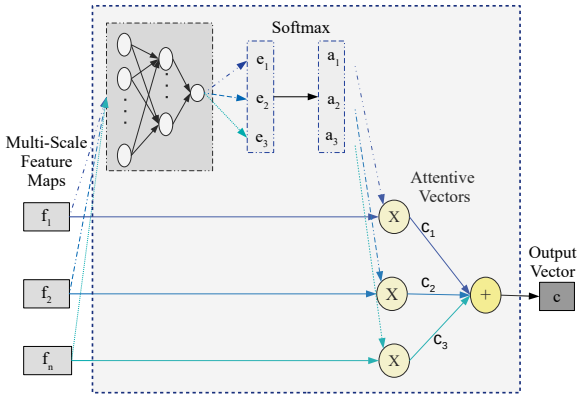

Fig. 2. Attention Mechanism block based on feed-forward NN.

Subsequently, the attentive feature vectors c_i are estimated taking into account the normalized importance a_i of each feature map f_i ($c_i = a_i * f_i$) and finally we sum up to estimate the attentive vector c. Alternatively, the self-attention framework [12] could also be used. However, the employment of self-attention will increase the training parameters and the complexity of network and thus this is avoided here.

Figure 3 presents an illustration of the proposed Multi-Scale Attention DL model. We notice that the final classification block consists of a dense neural network and batch normal-

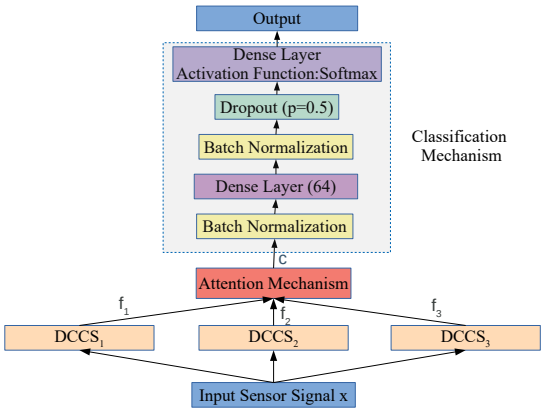

Fig. 3. Multi-Scale Attention Model.

ization layers. Also, we observe the use of dropout layer [13] in the classification block to further avoid over-fitting and enhance the regularization power of the model.

III. Experimental simulations

To show the effectiveness of the proposed DL model, we simulate a FD task with the use of a well-established dataset benchmark: the Case Western Reserve University (CWRU) dataset [14]. Using the bearing signals under the sampling rate of 12kHz and under 4 loading conditions, we build the training and testing datasets of a fault recognition problem. In more details, there are 5 types of defects: ball, inner race and three classes of outer race failures. Furthermore, we consider the severities of the faults that correspond to the diameter of the failures and recognize faults of 0.007, 0.014, 0.021, and 0.028 inches, ending up with the identification of 16 classes.

In order to compare against in the FD task, another CNN model has been built. The latter has 3 convolutional blocks in a similar way of the proposed model. The kernel sizes of each convolutional layer are 32, 5 and 3 while the number of filters are 32, 48 and 64. The final classification block is identical with the one of the proposed model except from the dense layer that has 128 output neurons. Both models under comparison have about the same number of parameters. In more details, the learning parameters of the proposed and of the competing simple CNN model are 27,024 and 29,536 accordingly and so we consider that have the same learning capacity. It is worth mentioning that the simple CNN model achieves accuracy close to state-of-the-art when the number of training samples is sufficient.

Moreover, both models employ the categorical cross-entropy loss function and the stochastic gradient descent optimizer with Nesterov momentum decay during training. The learning rate is set to 10^{-2} and the parameters of momentum are equal to 0.0 and 10^{-3}, respectively. Finally, the number of training epochs and the selected batch size both are both set equal to 200.

To explore the feasibility of the models under comparison, we create training datasets with different numbers of samples. Specifically, the amount of training samples varies from 960 (60 samples per class) to 6400 (400 per class) while the testing dataset is made of 6400 samples for all cases. Also, we set the input of the vibration signal to 400, which corresponds to the amount of sample points per revolution. During the formation of the samples, we apply an overlap of 20 points between successive ones.

TABLE I
Perfomance of Models under comparison using the CWRU bearings dataset

# Simulation Case	# training samples	Accuracy of CNN model	Accuracy of Proposed Model
1	960	94.71(±1.11)	97.51(±0.98)
2	1600	97.80(±0.40)	98.84(±0.28)
3	3200	98.65(±0.37)	99.20(±0.39)
4	4800	99.14(±0.26)	99.39(±0.20)
5	6400	99.48(±0.14)	99.57(±0.17)

In the simulation's development procedure, the programming language Python 2.7 in combination with the DL framework "keras" are adopted. To bypass the effect of initial random initialization of the neural weights, we train and estimate the FD accuracy of both models for 10 times. So, Table I shows the mean accuracy and the standard deviation for each model and simulation case.

Observing Table I, we notice that the proposed model achieves better performance from the competing CNN in all simulation cases. We see that for simulation cases #1, #2 and #3 where the number of training samples is limited, the multi-scale convolutional model performs much better than the CNN. Especially in simulation case #1, where the number of training samples is very small (60 samples per class), the proposed model overcomes the CNN one with a difference in performance of about 3 percentage points. So, the proposed model is robust to over-fitting and identifies the bearing failure patterns with effectiveness since achieves impressive results either with less or more training samples.

Furthermore, to obtain intuition of the arrangement of the produced feature vector mappings c and to illustrate their discrimination power, we display their visualization via t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm. In more details, we train the proposed model and in the sequel we estimate and visualize the output feature vectors for each testing input. The t-SNE algorithm is a widely used visualization method since it transforms each high-dimensional vector to a low dimensional space in such a way that related or similar objects are placed nearby while dissimilar ones are placed in distance. Figure 4 shows that the representation vectors that belong to the same type of fault are modeled by nearby points and are divided clearly by dissimilar ones, aiming the classification block to further distinguish the failure classes.

Also, to strengthen the knowledge behind the attention mechanism, we plot the histograms of the attentive values $\alpha_i, i \in \{1, 2, 3\}$ belonging to the normalized importance vectors α (Fig. 5). In a similar way, we train the model and subsequently gather the normalized attention vectors α

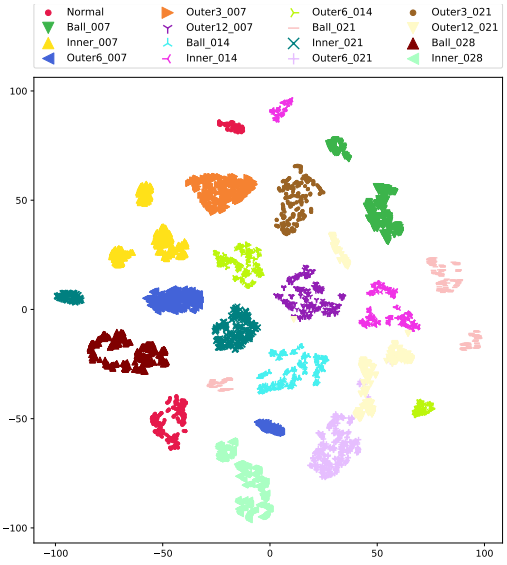

Fig. 4. Visualization via t-SNE of the extracted feature mapping.

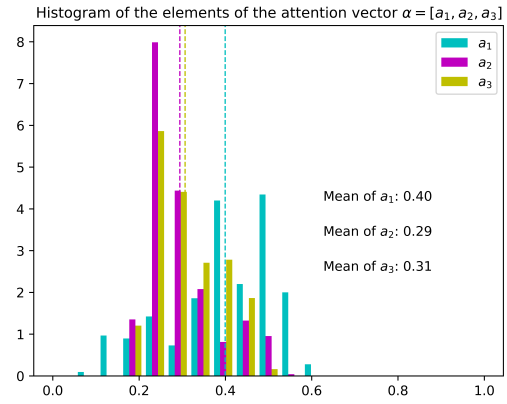

Fig. 5. Histogram of the attentive values $\alpha_i, i \in \{1, 2, 3\}$ that belong to the normalized importance vectors α.

corresponding to each testing sample. Each attention value a_i indicate the importance of the independent dense multi-scaling feature in the discrimination process. So, by the observation of the plot, we extract knowledge about the significance of each independent stream. We notice that the stream which corresponds to the convolutional layer with kernel size of 8 has the greater mean value while the other two have about the same means. Thus, it is the most important in the classification mechanism concluding that smaller kernel sizes in the 1D convolutional operation are more efficient since it process the input raw signals with greater resolution.

IV. Conclusions & Future work

In the current research paper, the use of multi-scale dense convolutional blocks, trained in parallel, is introduced aiming to rolling bearings fault detection in industrial motor drives. The produced training feature vectors, corresponding to each multi-scale neural stream, are fused with the aim of a simple attention mechanism. In that way, we merge the features achieving to keep information without increasing the dimension of the vector and thus avoiding over-fitting. The impressive simulation results, especially in cases of limited training datasets show the generalization and discrimination superiority of the proposed scheme. Future work can explore the application of more complex multi-scale architectures with a simultaneous recognition of the temporal coherence of the input vibration signal via attention mechanisms.

Acknowledgements

This research is co-financed by Greece and the European Union (European Social Fund- ESF) through the Operational Programme «Resources Development, Education and Lifelong Learning 2014-2020» in the context of the project "*Investigation and Development of an Intelligent System for Fault Detection, Diagnosing and Prognosing in Industrial Induction Motors*" (MIS 5050019).

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[2] S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep learning algorithms for bearing fault diagnostics—a comprehensive review," *IEEE Access*, vol. 8, pp. 29 857–29 881, 2020.

[3] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, and S. Van Hoecke, "Convolutional neural network based fault detection for rotating machinery," *Journal of Sound and Vibration*, vol. 377, pp. 331–345, 2016.

[4] S. Guo, T. Yang, W. Gao, and C. Zhang, "A novel fault diagnosis method for rotating machinery based on a convolutional neural network," *Sensors*, vol. 18, no. 5, 2018.

[5] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of the 31st Intl. Conf. on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[7] Y. L. Karnavas, S. Plakias, and I. D. Chasiotis, "Extracting spatially global and local attentive features for rolling bearing fault diagnosis in electrical machines using attention stream networks," *IET Electric Power Applications*, 2021. Early Access,. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/elp2.12063, doi: 10.1049/elp2.12063.

[8] S. Plakias and Y. S. Boutalis, "Fault detection and identification of rolling element bearings with attentive dense cnn," *Neurocomputing*, vol. 405, pp. 208–217, 2020.

[9] X. Li, W. Zhang, and Q. Ding, "Understanding and improving deep learning-based rolling bearing fault diagnosis with attention mechanism," *Signal Processing*, vol. 161, pp. 136–154, 2019.

[10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, p. 448–456.

[11] C. Raffel and D. P. W. Ellis, "Feed-forward networks with attention can solve some long-term memory problems," 2016.

[12] E. Kim, S. Cho, B. Lee, and M. Cho, "Fault detection and diagnosis using self-attentive convolutional neural networks for variable-length sensor data in semiconductor manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 3, pp. 302–309, 2019.

[13] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[14] D. Neupane and J. Seok, "Bearing fault detection and diagnosis using case western reserve university dataset with deep learning approaches: A review," *IEEE Access*, vol. 8, pp. 93 155–93 178, 2020.