On Applying Convolutional Neural Network to Bearing Fault Detection

Valentin Perminov, Dmitry Korzun
Petrozavodsk State University (PetrSU)
Petrozavodsk, Russia
{perminov, dkorzun}@cs.petrsu.ru

***Abstract*—In this paper, we explore the applicability of CNN for the classification of bearing defects. The approaches to the CNN evaluation are discussed and the one that brings the formulation of the diagnostic problem closer to real conditions is purposed. We demonstrate problems associated with the practical application of CNN in monitoring industrial equipment. We show that hyperparameter optimization is able to improve training process stability and consequently provide the reliability of the CNN-based diagnostic method. Our early experiments indicate a possibility of application CNN for failure diagnostics using a vibration signal provided that the training data contains a sufficient number of bearings with various types of faults, ensuring high representativeness of the dataset.**

I. Introduction

With the rising of the industrial Internet of things, big data, and machine learning, intelligent systems of mechanical equipment condition monitoring are increasingly being developed [1]. Early detection of defects and failure prediction allows preventing unscheduled downtime, economic loss, and accidents. The rolling bearing damages are the most common failures, therefore the problem of bearing fault detection is the main one that attracts the attention of many researchers [2].

The Convolution Neural Networks (CNN) is one of the most promising deep learning methods. Being a data-driven approach CNN enables end-to-end learning without requiring manual feature engineering. The application of CNN to bearing fault detection has been investigated by many researchers [3]–[9]. The impressive results are demonstrated, the reported accuracy of the proposed methods often achieves almost 100%. However, most of the works evaluate the accuracy of the model on a dataset that includes signal samples for the same bearings on which the training was carried out [3]–[6]. In the case of applying the bearing fault detection method in practice, we need to diagnose the bearing, which was not included in the training dataset. That is, testing the model on data from the same bearings on which it was trained is not correct, even if the training and test datasets contain data obtained under different conditions (rotation speed, load, time).

In the best case, we are able to collect a dataset containing examples of failures that have already happened on this equipment in order to train the model to diagnose or predict similar failures of this equipment in the future. Regarding the bearing diagnostics, to correctly evaluate the model, it is necessary to separate the training and test data by bearing instances, observing, as much as possible, an equal percentage of bearing condition examples in both datasets.

In this paper, we explore the applicability of CNN for the classification of bearing defects. To evaluate the CNN accuracy, we use a separate test dataset containing those bearings that were absent in the training and validation datasets. This brings the formulation of the diagnostic problem closer to real conditions and demonstrates the problems associated with the practical application of CNN in monitoring industrial equipment.

The rest of the paper is organized as follows. Section II introduces our methodology for using CNN. Section III discusses experiments of applying CNN to the classification of bearing defects. Section IV summarizes our early experimental results.

II. Methodology

We use the Paderborn university dataset [10], which contains 6 healthy bearings, 12 bearings with artificial damages, and 14 with natural damages, caused by accelerated life tests. The damage type is classified as inner race defect or outer race defect. Hence, we will refer to the following bearing classes: healthy (H), inner race defect (IR), and outer race defect (OR). The dataset contains the records of next signals:

- vibration (acceleration of the bearing housing),
- motor current,
- radial force,
- load torque,
- rotational speed,
- oil temperature in the bearing module.

The 20 measurements of 4 seconds each were done for 4 operating conditions. The operating conditions varied rotational speed, load torque, and radial force.

To simulate the deployment of a CNN-based fault diagnostic system, we chose four bearings of each class to the training dataset and one of each class to the test dataset. Among the faulty bearings, we chose only those with natural damage. The names of selected bearings are given in Table I. We randomly select 30% from the training dataset as the validation dataset.

As input data of CNN, we use spectrograms obtained by Short-time Fourier transform (STFT). The spectrograms are constructed from 256 one-sided spectra of vibration signal segments of length 512, smoothed by the Hann window function. No overlap between signal segments is applied.

TABLE I. DATASET CATEGORIZATION

Set No.	Healthy (H)	Outer ring damage (OR)	Inner ring damage (IR)
1	K001	KA04	KI04
2	K002	KA15	KI14
3	K003	KA16	KI16
4	K004	KA22	KI18
5	K005	KA30	KI21

Hence, obtained spectrograms have a resolution of 257 along the frequency axis and 256 along the time axis. Since the signal sampling rate is 64 kHz, each spectrogram covers a time interval of approximately 2 seconds. To match the size of the spectrogram with the input of the CNN we resize the spectrogram to 256x256 by bicubic interpolation and normalize to an interval from 0 to 1. The examples of input data (spectrograms) are shown in Fig. 1.

We use 2D CNN to classify bearing conditions by spectrograms of the vibration signal. The CNN was purposed by Yann LeCun et al. [11] for image classification and is widely used today. The CNN consists of multiple convolutions and pooling layers that perform feature extraction followed by a multilayer perceptron that performs classification.

The architecture of CNN used in our experiments is shown in Table II. We use CNN with 7 convolution layers (Conv2D) and 2 fully-connected layers (Dense). After each convolution layer, the max-pooling layer (MaxPool2D) is applied to reduce the spatial size of the feature map. And before each fully-connected layer, a dropout layer (Dropout) is applied to reduce overfitting. The number of filters and the kernel size of convolution layers are denoted as F and K respectively. The index n defines the CNN width. During architecture search we vary n from 0 to 4. The pooling size of max-pooling layers is designated as P. The number of neurons in fully-connected layers is denoted as N and the dropout rate of the dropout layer is D.

We use the ReLU activation function in all convolution layers and the first fully-connected layer. In the last fully-connected layer we use the softmax activation function. We apply L2 regularization to the weights of convolution layers with a default factor of 0.01. We use Adam optimizer [12] with a default learning rate of 0.0005 and categorical cross-entropy loss function. The number of training epochs was set to 100.

We use the next software stack. The Keras framework with TensorFlow backend is used to implement and train CNN. The Optuna framework is used for hyperparameter optimization. The SciPy and OpenCV libraries are applied for data preprocessing.

III. RESULTS AND DISCUSSION

We conduct a simple CNN architecture search by varying its width n from 0 to 4 (see Table II). We use set 1-4 for train and validation datasets and set 5 for testing, corresponding to Table I. The results of the CNN architecture search are shown in Table III.

TABLE II. THE CNN ARCHITECTURE

Layer	Output shape	Hyperparameters
Input	256x256x1	
Conv2D	256x256x2^n	F=2^n, K=3x3, ReLU
MaxPool2D	128x128x2^n	P=2x2
Conv2D	128x128x2^{n+1}	F=2^{n+1}, K=3x3, ReLU
MaxPool2D	64x64x2^{n+1}	P=2x2
Conv2D	64x64x2^{n+2}	F=2^{n+2}, K=3x3, ReLU
MaxPool2D	32x32x2^{n+2}	P=2x2
Conv2D	32x32x2^{n+3}	F=2^{n+3}, K=3x3, ReLU
MaxPool2D	16x16x2^{n+3}	P=2x2
Conv2D	16x16x2^{n+4}	F=2^{n+4}, K=3x3, ReLU
MaxPool2D	8x8x2^{n+4}	P=2x2
Conv2D	8x8x2^{n+5}	F=2^{n+5}, K=3x3, ReLU
MaxPool2D	4x4x2^{n+5}	P=2x2
Conv2D	4x4x2^{n+4}	F=2^{n+4}, K=3x3, ReLU
MaxPool2D	2x2x2^{n+4}	P=2x2
Dropout	2^{n+6}	D=0.2, ReLU
Dense	2^{n+6}	N=2^{n+6}
Dropout	2^{n+6}	D=0.2
Dense	3	N=3, softmax

TABLE III. THE CNN ACCURACY FOR DIFFERENT n

n	Train accuracy	Validation accuracy	Test accuracy
0	36%	26%	35%
1	34%	32%	28%
2	98%	97%	28%
3	98%	99.7%	74%
4	99.9%	99.7%	9%

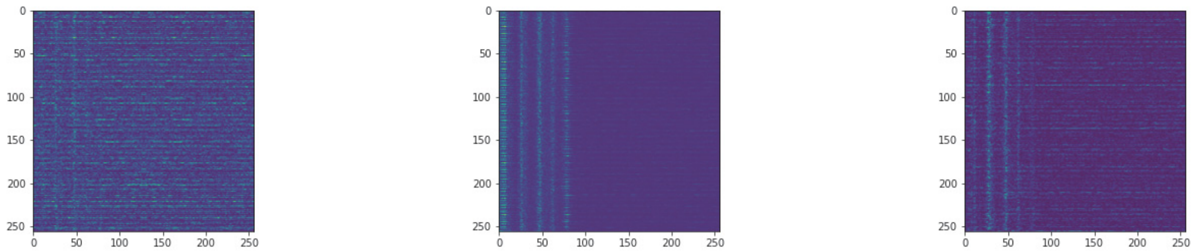
One can see, that when n is less than 2 the accuracy on all datasets is at the level of a random decision. This indicates that there is no convergence of the training process. When $n = 2$ the train and test accuracy exceeds 95%, but test accuracy is very low, which suggests that the CNN has poor generalization and is able to classify only those bearings on which it was trained. With $n = 3$, the results are satisfactory and indicate that CNN can be trained so that it is able to classify new bearings. When $n = 4$, overfitting is observed.

Hence, we can mark two milestones when developing a CNN for vibration diagnostics of bearings:

1) the CNN is able to classify data acquired from the same bearings that were in the training dataset;
2) the CNN is able to classify data acquired from bearings unseen before.

Only the second milestone would allow us to deploy CNN in the production diagnostic system. But the first one also remains necessary and may help steer the search in the right direction.

During experiments, we observe instability in test accuracy and loss between training trials. We show this effect for best-performed CNN (with $n = 3$) in Fig. 2 an 3. One can see that the test loss has a large magnitude and variance, and the test accuracy sometimes exceeds 70%, and sometimes it remains



(a) (b) (c)

Fig. 1. The spectrogram examples: healthy (a), outer race defect (b), and inner race defect (c)

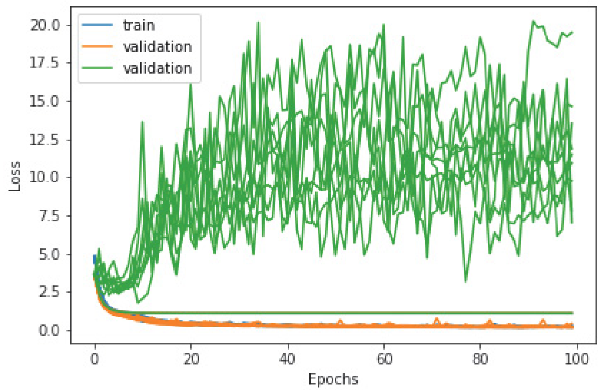


Fig. 2. The optimization learning curve of 10 training trials of the same CNN

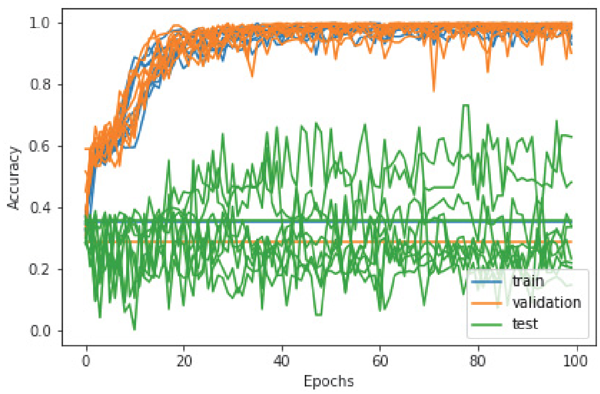


Fig. 3. The performance learning curve of 10 training trials of the same CNN

stable at the same level of about 30%. Such a variance in trends from one training trial to another may be due to the random initialization of CNN weights and the stochastic nature of the training process. This instability suggests that the high accuracy of the model is due to chance. To develop reliable diagnostic methods, it is necessary to reduce this instability as much as possible.

To reduce this variance, we have performed optimization of training hyperparameters, introducing standard deviation of test accuracy over 5 training trails as one of the objectives. Hence, we set the multiobjective optimization task, where we vary next hyperparameters:

- L2 regularization – from 0.001 to 0.1 (log scale);
- dropout (for two fully-connected layers) – from 0 to 0.7;
- learning rate – from 0.00001 to 0.001 (log scale).

We maximize two objectives:

1) mean validation accuracy,
2) mean test accuracy;

and minimize:

1) mean validation loss;
2) mean test loss;
3) standard deviation of test accuracy.

The objectives are calculated over 5 training trails of the same CNN configuration. The 93 evaluation trials have been done. The first 43 hyperparameters sets have been sampled randomly and the rest 50 have been sampled by the MOTPE algorithm [13].

Fig. 4 shows the resulting Pareto-front for the test accuracy mean and standard deviation. The optimal configuration can be considered as having the maximum mean accuracy, since the standard deviation is relatively small, in comparison with other configurations the mean accuracy of which is higher than 40%. Hyperparameters of this configuration:

- L2 regularization – 0.001697;
- dropout for the first layer – 0.2148;
- dropout for the second layer – 0.5096;
- learning rate – $5.302 \cdot 10^{-5}$.

Fig. 5 and 6 shows the learning curves for the selected optimal configuration. One can see that for all 5 trials similar trends are observed, the training process is more stable in comparison with the one that was observed before the optimization of the training hyperparameters (Fig. 2 and 3). Though the test error is still rising, which indicates the overfitting of the model, its maximum value is about two times less than before optimization. The obtained results demonstrate the applicability of optimization of training hyperparameters to increase the stability of the training process and the reliability of the model.

The confusion matrices for 5 training trials are shown in Fig. 7. It can be seen that in most cases the model classifies the health bearing (H) and inner ring defect (IR) with satisfactory

Fig. 4. The Pareto-front: test accuracy mean vs test accuracy standard deviation

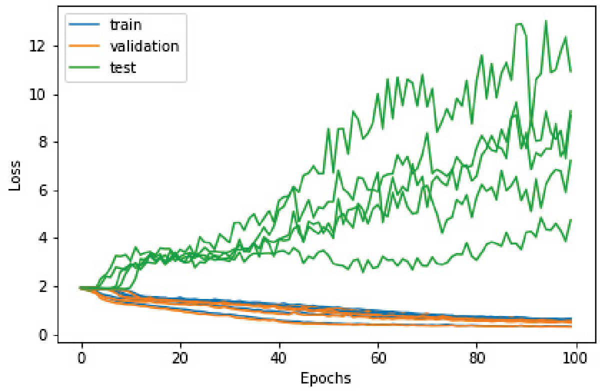


Fig. 5. The optimization learning curve of 5 training trials of the same CNN after hyperparameters optimization

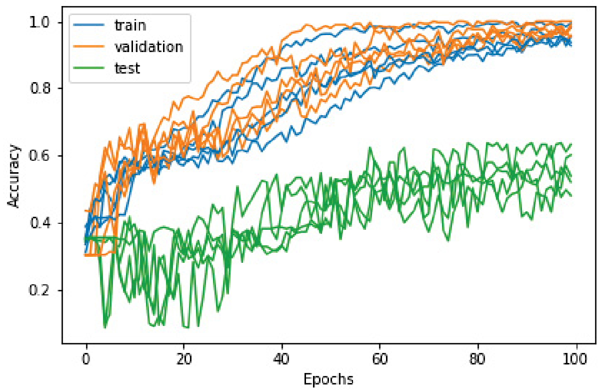


Fig. 6. The performance learning curve of 5 training trials of the same CNN after hyperparameters optimization

accuracy, but the accuracy of outer ring defect (OR) classification is almost zero. This may indicate that the vibration signal of the bearing KA30 (OR) defect is similar to IR defects from the training dataset. In this case, data collection from more bearings is required to improve the classification accuracy.

To check whether the selected architecture and training hyperparameters are optimal only for the selected division of the dataset into training and test sets, cross-validation was carried out. According to Table I, data is split into 5 folds (sets), 4 folds is used for training and validation datasets, and 1 for test dataset, resulting in 5 combinations. For each combination, training is repeated 5 times to evaluate training process stability. Fig. 8 and 9 show the learning curves for each combination. One can see that the general trends of the curves remain the same within a combination. This suggests that the chosen training hyperparameters ensure the stability of the training process. The question of the possibility of selecting more optimal parameters for each combination of training and testing data remains open. Fig. 10 shows confusion matrices for 5 combinations of training and testing data normalized over the true conditions and averaged over 5 training trials. The accuracy of classification is low and varies from one data combination to another. Similar results for cross-validation of CNN on the Paderborn university dataset were observed by Pandhare et.al. [7].

The third experiment (fold 3 as test set) outstands among others. The CNN accuracy on the test data exceeds 80%. This may indicate that the mode of the defects in the test dataset coincided with the mode of the defects in the training dataset. This, in turn, demonstrates the possibility of diagnosing bearing defects in practice, when a diagnosed bearing was not in the training dataset. The condition for such a possibility is the presence in the training dataset of bearings with a similar mode of defects. The likelihood of having such bearings in the training dataset increases with the number of bearings and the variety of their defects. Thus, in the presence of a sufficiently large data set, it is possible to apply CNN for failure diagnostics using a vibration signal. However, collecting enough data for training is difficult in the case of bearings. The task of diagnosing/classifying the state of units that often fail or wear out seems to be more feasible. In this case, it is possible to collect enough examples of faults and then, by introducing CNN-based diagnostics, perform fault detection in the early stages, as well as evaluate the remaining useful life, if there is historical data on the fault progression.

In summary, our experimental results demonstrated the possibility of application CNN to failure diagnostics using a vibration signal provided that the training data contains a sufficient number of bearings with various types of faults, ensuring a high representativeness of dataset. However, since in practice it is difficult to collect a sufficient amount of labeled data, transfer learning methods can be considered for applying CNN in such cases [14]–[16].

IV. Conclusion

In this paper, we focus on the problem of diagnosing bearing faults using CNN. We discuss the approaches to the evaluation of CNN accuracy in the context of bearing faults diagnosing. We argue that for real application the model quality should be evaluated on the bearing instances which was not included in train data. We show, that training process

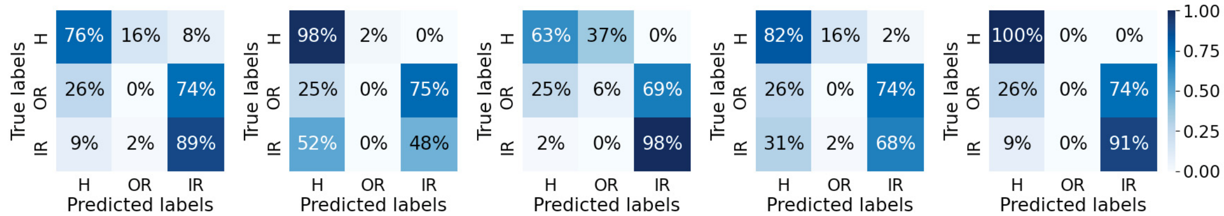


Fig. 7. The confusion matrices for 5 training trials normalized over the true conditions

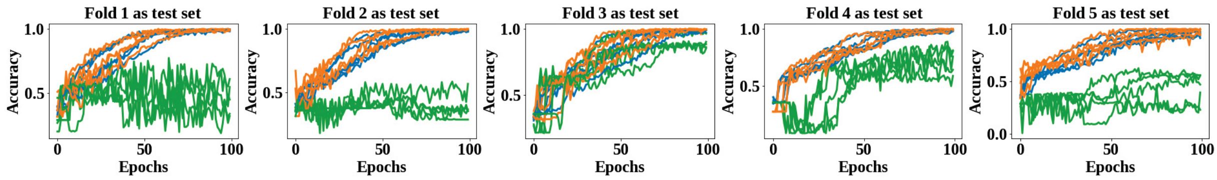


Fig. 8. The performance learning curve for 5 combinations of training and testing data

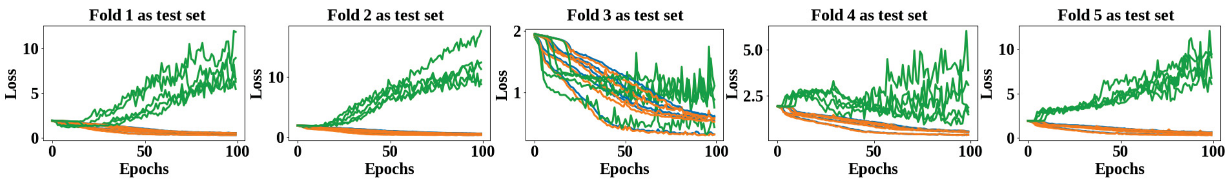


Fig. 9. The optimization learning curve for 5 combinations of training and testing data

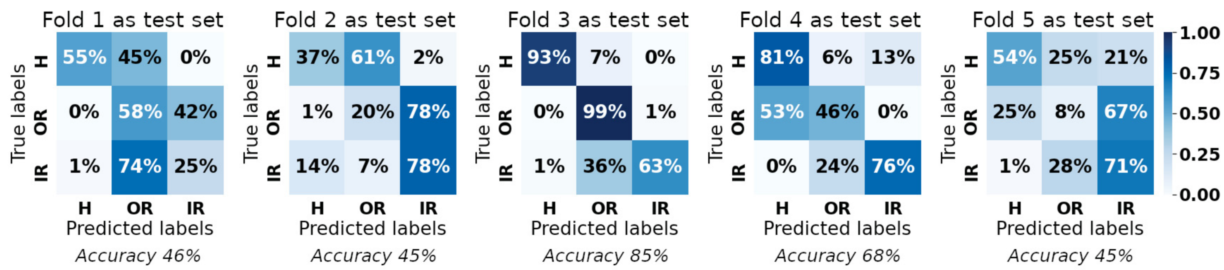


Fig. 10. The confusion matrices for 5 combinations of training and testing data normalized over the true conditions and averaged over 5 training trials

stability should be checked since the stochastic nature of the training process could lead to falsely successful results. Hence, the training process stability should be increased as much as possible to provide the reliability of the diagnostic method. We demonstrate that a hyperparameter optimization is an effective tool for increasing the stability of the neural network training process, which would be difficult to achieve by manual selection.

Acknowledgment

This development was implemented in Petrozavodsk State University (PetrSU) with financial support of the Ministry of Science and Higher Education of Russia within Agreement no. 075-11-2019-088 of 20.12.2019 on the topic "Creating the high-tech production of mobile microprocessor computing modules based on SiP and PoP technology for smart data collection, mining, and interaction with surrounding sources". This research study is supported by RFBR (research project # 19-07-01027) in part of edge-centric computing models for applying neural networks. The work is in collaboration with the Artificial Intelligence Center of PetrSU.

References

[1] V. Perminov, V. Ermakov, and D. Korzun, "Edge computing opportunities for vibration diagnostics of rotary machinery using neural network approach," *26th Conference of Open Innovations Association FRUCT*, vol. 26, no. 2, pp. 695–697, 2020.

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

[3] P. Liang, C. Deng, J. Wu, Z. Yang, and J. Zhu, "Intelligent fault diagnosis of rolling element bearing based on convolutional neural network and frequency spectrograms," in *2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, 2019, pp. 1–5.

[4] L. Wen, X. Li, and L. Gao, "A transfer convolutional neural network for fault diagnosis based on resnet-50," *Neural Computing and Applications*, pp. 1–14, 2019.

[5] Z. Zilong and Q. Wei, "Intelligent fault diagnosis of rolling bearing using one-dimensional multi-scale deep convolutional neural network based health state classification," in *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, 2018, pp. 1–6.

[6] S. Li, G. Liu, X. Tang, J. Lu, and J. Hu, "An ensemble deep convolutional neural network model with improved ds evidence fusion for bearing fault diagnosis," *Sensors*, vol. 17, no. 8, p. 1729, 2017.

[7] V. Pandhare, J. Singh, and J. Lee, "Convolutional neural network based rolling-element bearing fault diagnosis for naturally occurring and progressing defects using time-frequency domain features," in *2019 Prognostics and System Health Management Conference (PHM-Paris)*. IEEE, 2019, pp. 320–326.

[8] 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.

[9] V. Perminov, V. Ermakov, and D. Korzun, "Fault diagnosis for industrial rotary machinery based on edge computing and neural networking," in *UBICOMM 2020 : The Fourteenth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*. IARIA, 2020, pp. 1–6.

[10] C. Lessmeier, J. K. Kimotho, D. Zimmer, and W. Sextro, "Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification," in *Proceedings of the European conference of the prognostics and health management society*. Citeseer, 2016, pp. 05–08.

[11] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, pp. 1–15, 2014.

[13] Y. Ozaki, Y. Tanigaki, S. Watanabe, and M. Onishi, "Multiobjective tree-structured parzen estimator for computationally expensive optimization problems," in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, 2020, pp. 533–541.

[14] L. Guo, Y. Lei, S. Xing, T. Yan, and N. Li, "Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 9, pp. 7316–7325, 2018.

[15] B. Yang, Y. Lei, F. Jia, and S. Xing, "An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings," *Mechanical Systems and Signal Processing*, vol. 122, pp. 692–706, 2019.

[16] Z. Chen, K. Gryllias, and W. Li, "Intelligent fault diagnosis for rotary machinery using transferable convolutional neural network," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 339–349, 2019.