

Noise-Boosted Convolutional Neural Network for Edge-based Motor Fault Diagnosis with Limited Samples

Lv Chen, Kang An, Dali Huang, Xiaoxian Wang, *Member, IEEE*, Min Xia, *Senior Member, IEEE*,
and Siliang Lu, *Senior Member, IEEE*

Abstract—Convolutional neural networks (CNNs) have been widely applied in motor fault diagnosis. However, to obtain high recognition accuracy, massive training data are typically required and transmitted to the cloud/local server for training, which may suffer from security and privacy problems. In this study, a noise-boosted CNN (NBCNN) model is developed to achieve accelerated training and improved recognition accuracy with limited training samples. First, the NBCNN model with a noise-injection fully connected layer is established. Then, a strategy for noise selection and injection is proposed to obtain an optimal matching among the data, model, and noise. Finally, the optimal injected noise accelerates the convergence of model training and improves the accuracy of motor fault diagnosis. Compared with the conventional CNN without noise injection and the state-of-the-art models, the effectiveness and superiority of the proposed NBCNN model are validated by two benchmark datasets. In addition, the algorithm is deployed onto an edge device and the results show that the training speed of the developed NBCNN can reach nine times faster than the conventional CNN. The proposed method shows remarkable potential for distributed model training, federal learning, and real-time motor fault diagnosis.

Index Terms—CNN, noise injection, motor fault diagnosis, limited samples, edge device

I. INTRODUCTION

Motors provide mechanical kinetic energy and have been generally used and play an indispensable role in household and industrial equipment [1, 2]. Motors that work in extreme environments are prone to failure after a long period of operational wear and environmental erosion. These failures can accelerate equipment damage and cause

safety hazards, or directly lead to severe safety accidents, resulting in huge losses of life and property [3, 4]. Therefore, to maintain the safe and reliable operating environment of motor equipment, the study of motor fault detection and diagnosis technologies is particularly important.

The rapid development of computing power has led to a new wave of deep learning (DL), which has shown great application potential and superiority in natural language processing, speech recognition, and image recognition [5-9]. Given the shortcomings of traditional fault diagnosis that heavily rely on expert knowledge and human experience, the application of DL to mechanical fault diagnosis has become a hot topic in current research. For example, Ma et al. used enhanced stacked auto-encoder for fault determination of gears and bearings [10]. Du et al. used the adaptive multifractal detrended fluctuation analysis and doubly iterative empirical mode decomposition to examine the fault diagnosis of bearings, gears, and piston pumps [11]. Song et al. investigated the accurate detection of motor faults based on envelope and time domain energy analyses [12]. Zhang et al. used stacked sparse autoencoders for interturn short-circuit fault in permanent magnet motors [13]. Liu et al. proposed a deep adversarial domain adaptation model to address the issue where the distribution of the source and target domains is inconsistent [14]. Wang et al. investigated a new fault diagnosis method using time-frequency representation and deep reinforcement learning in varying working conditions [15]. These studies show the capability of DL and further promote its application in fault diagnosis.

One of the most commonly used models for machine fault diagnosis is the convolutional neural network (CNN). Many CNN-based machine fault diagnosis methods can extract representative features from the raw input and have achieved outstanding classification results. For instance, Husari et al. proposed a two-stage network based on CNN and support vector machine, and this method can not only identify the fault category of the induction motor, but also the severity of failure [16]. Xie et al. used CNN to implement intelligent fault diagnosis of multi-sensor data [17]. Zhang et al. directly extracted features from the original time domain model of data through CNN model with two dropout layers and two fully connected layers [18]. Ren et al. combined CNN and long-short term memory network to design a residual life prediction model for lithium batteries [19]. Based on an improved CNN with transfer learning, Shao et al. designed a new framework for fault diagnosis of rotor bearing systems under different operating conditions [20]. Peng et al. proposed a multibranch

Manuscript received Oct. 2, 2022; accepted Dec. 07, 2022. Paper no. TII-22-4132. This work was supported in part by the National Natural Science Foundation of China under Grants 52075002 and 62203010, and the Project of the Outstanding Young Talents in Colleges and Universities of Anhui Province under Grant gxyqZD2022006. (*Corresponding authors: Xiaoxian Wang; Min Xia; Siliang Lu*)

Lv Chen, Kang An, and Siliang Lu are with the College of Electrical Engineering and Automation, Anhui University, Hefei 230601, China. (e-mail: lusiliang@mail.ustc.edu.cn)

Dali Huang is with the Beijing Research Institute of Mechanical & Electrical Technology, Beijing 102299, China.

Xiaoxian Wang is with the College of Electronics and Information Engineering, Anhui University, Hefei, 230601, China, and also with the Department of Precision Machinery and Precision Instrumentation, University of Science and Technology of China, Hefei 230027, China. (e-mail: xiaoxian@ahu.edu.cn)

Min Xia is with the Department of Engineering, Lancaster University, Lancaster LA1 4YW, U.K. (e-mail: m.xia3@lancaster.ac.uk)

and multiscale CNN for multiple signal components and time scale fault information [21]. The method proposed by Tran et al. achieved higher fault diagnosis accuracy for induction motor diagnosis by combining continuous wavelet transform with convolutional attention neural network (CANN) [22]. The new progressive cascading CNN model designed by Wang et al. achieved better performance of fault diagnosis under nonstationary conditions [23]. Xiong et al. developed a fault diagnosis data preprocessing method based on mutual dimensionless and similar Gram matrix [24].

With the rapid growth of data and its transmission, data security and privacy are becoming more concerning [25-27]. Thus, the leakage during the transmission and processing of data caused by malicious or unintentional attacks needs to be prevented. To mitigate the risks, federated learning and distributed machine learning are developing fast in many applications. Data no longer need a uniform transfer to the central processor, but rather partially or fully calculated locally. Given the reduction in data transmission, the possibility of data leakage is also considerably reduced. However, distributed devices such as miniature instruments or handheld machines are often limited in computational resources and power supply due to their compact size and remote location. Therefore, optimizing the existing CNN model for distributed deployment is necessary to ensure satisfactory performance under resource-constrained conditions.

Another critical issue that prevents the practical adoption of the above approaches is the large amount of data needed for training. In real-world engineering applications, equipment operates in a healthy or normal state most of the time, which causes an extremely unbalance between data collected from fault conditions and normal condition. Unbalanced data with limited fault condition data brings significant difficulties to train the conventional CNN [28]. According to literature [29], backpropagation algorithms are a special case of generalized expectation maximization, and injecting appropriate noise can speed up the relevant training of CNN. In the Mixed National Institute of Standards and Technology database (MNIST), CNN with noise injection can effectively reduce the convergence speed of the first few trainings in handwritten digit recognition. Simulation results show that the maximum noise gain can be obtained in a small dataset.

CNN methods with noise injection show potential in improving the training efficiency. However, noise injection parameters have not been effectively optimized such that the noise gain is positive only in the first few rounds of training. For different samples, the noise yield hyperplane also needs to change accordingly, which hinders the generalizability of the model. To address these issues, we propose a novel noise-boosted convolutional neural network (NBCNN) model to achieve fault diagnosis of motor faults with limited data samples. Besides significantly accelerating the training convergence of CNN, the proposed method achieves higher prediction accuracy than other conventional models. The developed method achieves superior classification accuracy and is suitable for edge solutions where computational resources are limited and massive fault samples are difficult to be obtained. The main contributions of this paper are summarized in the following:

1) This study indicates that noise is beneficial to improve the

CNN model performance. To our knowledge, this topic has rarely been studied in the field of CNN-based motor fault diagnosis.

- 2) The proposed NBCNN method can not only effectively improve the convergence speed of model training, but also improve the prediction accuracy. The average accuracy of the NBCNN model can reach 99.72%.
- 3) The effectiveness and superiority of the proposed method have been verified in comparison with several state-of-the-art models. The flexibility and robustness are also validated by two kinds of datasets including the Case West Reserve University (CWRU) motor bearing dataset and the motor dataset from our laboratory.
- 4) The designed NBCNN has a small model size with fewer parameters, and can achieve high accuracy and fast fault prediction on resource-constrained edge devices such as NVIDIA Jetson TX1.

The remainder of the paper is organized as follows. Section II introduces the theoretical model of NBCNN. Section III describes the experimental setup and datasets. The performance of the NBCNN model is validated in Section IV in comparison with the state-of-the-art models. Section V verifies the deployment of the NBCNN algorithm onto an edge device. Section VI presents the conclusions and future work.

II. PROPOSED METHOD

This section introduces a new motor fault recognition method for limited samples based on NBCNN. The flowchart of the method is shown in Fig. 1. The raw vibration signal acquired from the motor is pre-processed to enhance the fault characteristics. Subsequently, the one-dimensional (1D) signal is stacked into a two-dimensional (2D) grayscale image after removing half of the repetitive information. Finally, the converted image is processed by the NBCNN model and the fault type of the motor can be determined.

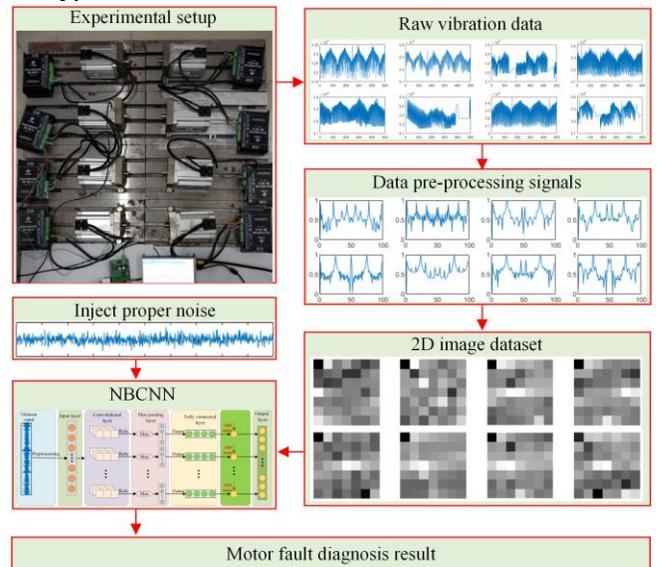


Fig. 1. The framework of the proposed NBCNN model for fault prediction.

A. Data Preprocessing

CNN can efficiently extract feature information from a 2D image. The typical machine condition monitoring data (e.g., vibration signals) are 1D data and hence they need to be

preprocessed to accommodate the CNN input. Converting a 1D vibration signal into a 2D image signal can be done in various methods, such as time–frequency conversion and 1D stacking method [30, 31]. In this study, a two-step data preprocessing method is adopted. The first step is a segmented sampling of the original vibration data, and the second is to use two logarithmic spectra (B2LS) [32] to process the segmented signal. This approach has been proven effective in extracting distinct features of fault data in Ref. [27]. Fig. 2 illustrates the workflow of the signal preprocessing. The final result is a set of low-resolution grayscale atlases, which corresponds to the NBCNN input resolution.

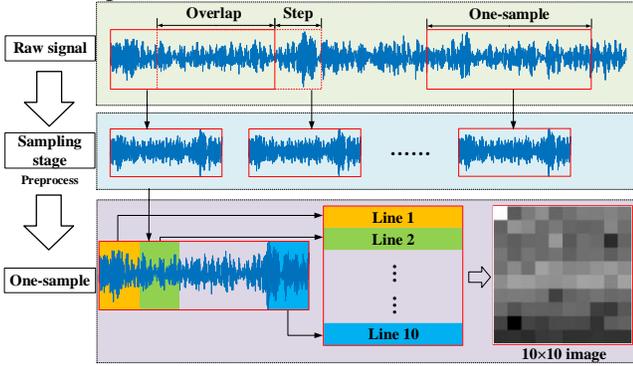


Fig. 2. Conversion of raw vibration data to 2D grayscale images.

In the segmented collection of raw data, augmentation is used to enhance the characterization ability of the small training samples [33]. Suppose that the original vibration signal length is L , each sampled segment has a length of M , and the sample coincident part length is O . The final sample size S that can be obtained is

$$[S] = \frac{L - O}{M - O}, \quad (1)$$

where $[\cdot]$ means rounding down.

The B2LS method can balance the difference in frequency components on the amplitude scale in the original data, weaken the significant influence of larger frequency components on the classification results, and enable CNN to extract comprehensive features. For the sampling segment $P[s]$, $s=1, 2, \dots, S$ mentioned above, this step can be described as

$$Q[i] = \log_2 \left(\frac{2 \times |FFT(P[s])|}{M} \right), i=1, 2, \dots, S, \quad (2)$$

where $FFT(\cdot)$ denotes the fast Fourier transform, and $Q[i]$ is the calculated symmetric sequence. Normalization to obtain the desired input grayscale image X of m row n column leads to

$$X = 255 \times \frac{\begin{bmatrix} Q[1] & Q[2] & \dots & Q[n] \\ Q[n+1] & Q[n+2] & \dots & Q[2n] \\ \dots & \dots & \dots & \dots \\ Q[(m-1)n+1] & \dots & \dots & Q[mn] \end{bmatrix}}{\max(Q[i])} - \min(Q[i]), \quad (3)$$

where $\max(Q[i])$ and $\min(Q[i])$ represent the maximum and minimum values of sequence $Q[i]$, respectively. This step ensures that all points of different amplitudes in the original vibration signal are fully mapped to pixels of varying brightness in the image.

B. CNN

The designed CNN architecture consists of one layer each for input, convolution, max-pooling, full connection, noise injection, and output.

When the gray-scale image dataset is fed into the input layer, the convolution layer extracts the information in the graph, described as

$$C = \sum_{j=1}^J (X * W_j) + B, \quad (4)$$

where X and C denote the input and feature matrices of a single grayscale map in the convolutional step, respectively; $j=1, 2, \dots, J$, is the index of the filter W_j ; and the symbol $*$ represents the 2D discrete convolution between the input X and the j th filter W_j . Then, the bias matrix B is added to the summation. Later, the rectified linear unit acts on the feature matrices to prevent the gradient from disappearing. This step can be described as

$$c' = \max(0, c), \quad (5)$$

where c and c' are the before and after elements of the output matrix C of Equation (4), respectively.

The max-pooling layer can down-sample the feature matrices, helping to reduce the model parameters, remove redundant information, and avoid overfitting. The max-pooling layer formula is

$$g'_{x,y} = \max(g_{x,y} : x' \leq x \leq x'+m, y' \leq y \leq y'+n), \quad (6)$$

where m and n are the width and height of the pooling window, respectively; and $g_{x,y}$ and $g'_{x,y}$ are the eigenmatrix elements before and after the update, respectively.

The fully connected layer acts as a linear classifier for the features extracted from the upper layer. Then noise is added to speed up the training and improve prediction performance.

The CNN is trained using a small batch gradient descent algorithm. The weight parameters are updated by calculating and minimizing the cross-entropy loss of the predicted values, calculated by a logarithmic softmax operation as

$$s_i = -\log h(x_i), \quad (7)$$

$$h(x_i) = \frac{e^{x_i}}{\sum_k e^{x_k}}, \quad (8)$$

where x_i is the value of the i th output neuron and s_i is the corresponding result. Equation (8) shows the calculation of $h(\cdot)$, which is the softmax operator.

C. NBCNN

Conditional noise injection has been applied to CNNs [29] and validated on handwritten digit recognition in MNIST datasets. However, noise in hypercubes corresponding to different sample sets is not always beneficial. The noise gains also diminish as the training progresses. Experiments on conditional noise injection [34, 35] show that the number of noise injections account for approximately half of the total training times in each dataset trained alone. Clearly, the potential of the noise injected CNN remains to be further explored, and its gain in classification accuracy can be enhanced.

In this study, the cross-entropy function is used to measure the difference between the real and predicted samples. The convergence and the weight matrix update faster in fault

classification training than using mean squared error. The cross-entropy $E(\Theta)$ is calculated as

$$E(\Theta) = -\sum_{i=1}^I y_i \ln(z_i), \quad (9)$$

where z_i represents the result of the output value calculated by softmax and $z_i = h(x_i)$; and y_i is the true value corresponding to z_i . The label vector y is a binary one-hot encoding of the fault type, meaning $y_i = 1$ only if i is the correct label classification and 0 otherwise. I is the total fault type number. Thus, Equation (9) can also be written as

$$E(\Theta) = -\ln z_i. \quad (10)$$

Equation (10) is the deformation of the y_i term with a value of 0 discarded in Equation (9). The likelihood function $L(\Theta)$ of the network can be written as

$$L(\Theta) = \ln p(y|z, \Theta). \quad (11)$$

The conditional probability density function $p(y|z, \Theta)$ is the probability of observation y of input x at the output layer in a CNN with parameters Θ . Continue deriving Equation (11) to obtain

$$L(\Theta) = \ln \prod_{i=1}^I p(y_i | z_i, \Theta), \quad (12)$$

$$= \ln \prod_{i=1}^I z_i^{y_i}, \quad (13)$$

$$= \sum_{i=1}^I y_i \ln z_i, \quad (14)$$

$$= -E(\Theta). \quad (15)$$

The likelihood function of the network $L(\Theta)$ is then equal to the value of negative cross-entropy $-E(\Theta)$, such as

$$L(\Theta) = -E(\Theta). \quad (16)$$

As seen from Equation (16), minimizing $E(\Theta)$ can maximize the likelihood $L(\Theta)$. To explore the noise component that can reduce cross-entropy $E(\Theta)$, we check that the cross-entropy of NBCNN meets the conditions

$$E_n(\Theta) - E(\Theta) \leq 0, \quad (17)$$

where $E_n(\Theta)$ represents the cross-entropy of NBCNN. Substitute Equation (10) into Equation (17) to obtain

$$\sum_{i=1}^I y_i \ln(z_i) - \sum_{i=1}^I y_i \ln(z'_i) \leq 0, \quad (18)$$

where z'_i represents the predicted value of NBCNN. This case involves the mapping of noise added in the fully connected layer at the output layer, which is expanded in detail as

$$z'_i = h\left(\sum_{j=1}^J U_{ij} \times (\omega_j + n_j) + b_i\right), \quad (19)$$

where U is the output layer neuron matrix, ω_j is the output value of the previous layer, n_j is the added noise, b_i is biased number, and $h(\cdot)$ is the softmax operator. Substitute Equation (19) into Equation (18) to obtain

$$\sum_{i=1}^I y_i \ln\left(h\left(\sum_{j=1}^J U_{ij} \times \omega_j + b_i\right)\right) - \sum_{i=1}^I y_i \ln\left(h\left(\sum_{j=1}^J U_{ij} \times (\omega_j + n_j) + b_i\right)\right) \leq 0. \quad (20)$$

Simplified, the conditions with positive noise gain are

$$h(u) \leq h(u + n), \quad (21)$$

where

$$u = \left(\sum_{j=1}^J U_{ij} \times \omega_j + b_i\right), \quad (22)$$

$$n = \left(\sum_{j=1}^J U_{ij} \times n_j\right). \quad (23)$$

As shown in Equations (22) and (23), respectively, u is the sample component and n is the noise component.

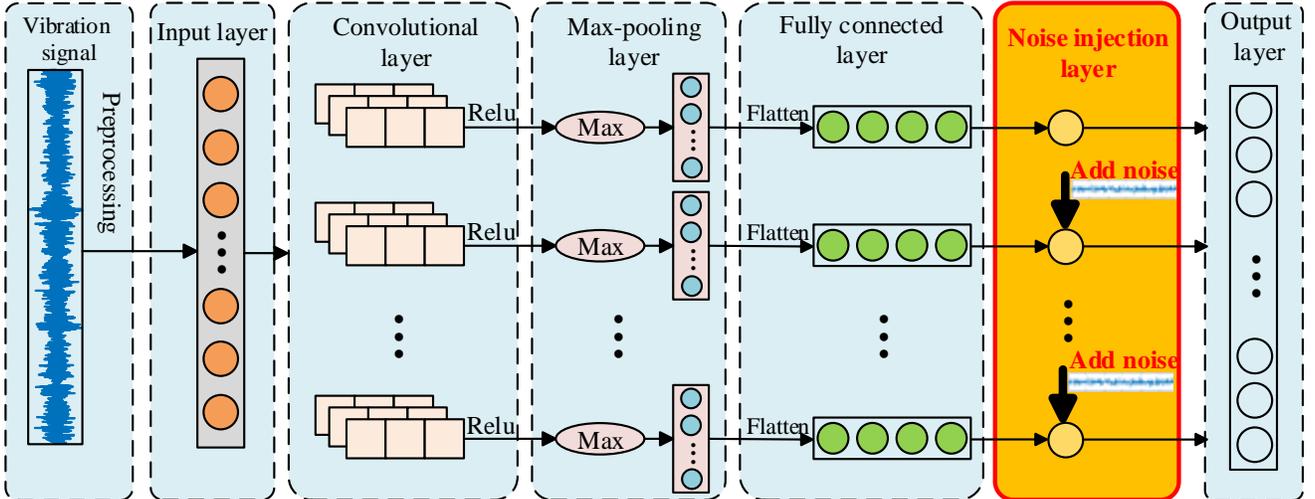


Fig. 3. Schematic diagram of NBCNN.

Noise can be used to improve CNN and save resource waste when judging such conditions. When choosing the noise type, random uniform noise can be avoided and the noise that is biased and is relative to the input can be selected. On this basis, and verified by experimentation, a parametric Gaussian noise associated with the input is applied to the present method and can be expressed as

$$n_i = \frac{-\omega_i^2}{\sqrt{2\pi\sigma^2}} e^{\frac{-\omega_i^2}{2\sigma^2}}, \quad (24)$$

where ω_i is the influencing factor of the noise component and the output of the last fully connected layer; σ is the noise parameter and set to 0.1, and can adjust the noise amplitude and affect the performance of the CNN; and n_i is the resulting noise

to be injected before the output layer. This noise is unconditionally injected into the CNN and affects the parameter update at each iteration. The noise injection method is given in Algorithm 1. Fig. 3 shows the schematic diagram.

Algorithm 1. Fault diagnosis based on NBCNN

Result: Trained CNN weight matrices
Initialize network weights Ω

For: epoch i : $1 \rightarrow I$ do
For: iteration j : $1 \rightarrow J$ do
Determine a batch of K samples $\{x^k, y^k\}_{k=1}^K$
Forward propagation
Convolution of the input matrix using (4)
Nonlinear rectification using (5) and (6)
Down-sample by max-pooling layer
A 1D vector a is obtained by a fully connected layer
Inject noise
Produce noise vector n using (24)
Inject noise
Backpropagation calculation
Calculate the cross-entropy error $E(\theta)$
Calculate the error gradient $\nabla_{\theta}^{(i)} E(\theta)$
Update the parameter set Ω

End
End

In this study, we set the learning rate, minimum batch size, and maximum epoch number of all CNNs as 0.01, 20 and 1000, respectively. The noise parameter of NBCNN is set to 0.1 by default.

III. EXPERIMENTAL SETUP AND DATA DESCRIPTION

Two datasets are used to verify the proposed method; one is the CWRU bearing data [36] and the other is collected from experiments on our test bench, as shown in Fig. 4. For the convenience of distinction, these two are called datasets 1 and 2, respectively.

A. Description of Dataset 1

The CWRU experimental setup includes a motor, a torque transducer/decoder, a dynamometer, and an electronic controller. The bearings that support the motor hinge are monitored (the drive end bearing is SKF6205 and fan end bearing is SKF6203). Single point damage is created by electrical discharge machining for different fault conditions. Table I shows the details of dataset 1.

TABLE I

FAULT TYPE AND LABEL OF DATASET 1

Fault label	Fault type	Fault depth(inch)	Fault name
1	B1	0.007	Ball fault
2	B3	0.014	Ball fault
3	B5	0.021	Ball fault
4	IR1	0.007	Inner race
5	IR3	0.014	Inner race
6	IR5	0.021	Inner race
7	OR1	0.007	Outer race
8	OR3	0.014	Outer race
9	OR5	0.021	Outer race
10	Normal	0	Normal

B. Description of Dataset 2

Fig. 4 shows the experimental setup for motor fault data collection. The test rig consists of 8 brushless direct current motors (BLDCMs) with different healthy or fault states.

Specifically, 2 kinds of bearing inner ring faults (BIRF) and 3 kinds of bearing outer ring faults (BORF) are generated by electrical discharge machining. The left bottom and right bottom subfigures in Fig. 4 show the BIRF and BORF with fault width of 1 mm, respectively. The rotor unbalance fault (RUF) is set by attaching an unbalanced mass to the rotor. The Hall sensor fault (HSF) is set by disconnecting one of the Hall sensor wires from the motor controller. Dataset 2 is collected from 7 faulty motors and one healthy motor, as shown in Table II. The motor is driven by the motor driver with a rated voltage of 48 VDC. The motor speed is set to around 1500 rpm. The motor vibration signal is collected by an accelerometer mounted on the motor housing, and amplified by an operational amplifier, and finally sampled by an analog-to-digital converter (ADC). The sampling frequency and duration are set as 20 kHz and 120 seconds, respectively. The raw vibration signal is transmitted from a micro controller unit (MCU) to a computer for storage and analysis. The detailed specifications of the instrument systems are summarized in Table III.

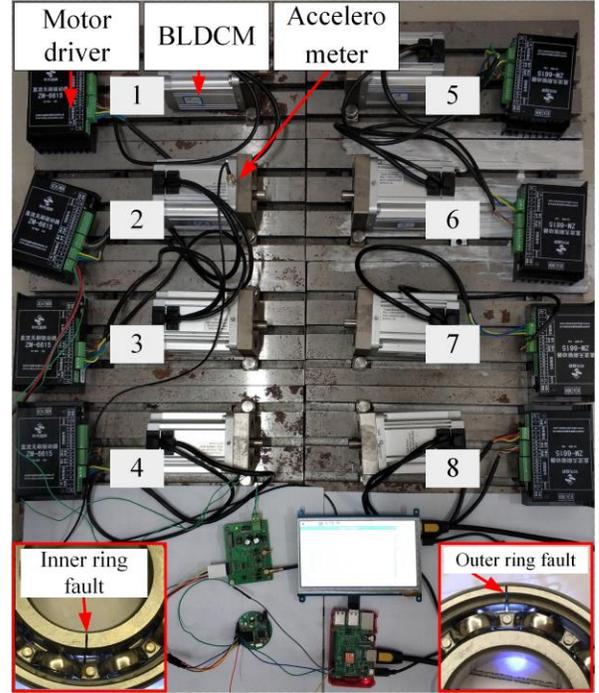


Fig. 4. Experimental setup.

TABLE II

FAULT TYPE AND LABEL OF DATASET 2

Fault Type	BIRF (1mm)	BIRF (2mm)	BORF (0.3mm)	BORF (1mm)
Label	1	2	3	4
Fault Type	BORF (2mm)	RUF	Healthy	HSF
Label	5	6	7	8

TABLE III

HARDWARE PARAMETERS OF BLDCM TEST RIG

Device	Model	Manufacturer
BLDCM	80BL110S50	SDCQ Inc.
Accelerometer	CA-YD-1182	SINOCERA Inc.
Operational amplifier	MAX9632	Maxim Inc.
ADC	MAX1300	Maxim Inc.
MCU	STM32F407	STMicroelectronics Inc.

When the training sample size is comparatively small, the sensitivity of the sample size to the final performance can be more obvious. To investigate the effects of the different sample sizes on the NBCNN, we selected different numbers of samples from each dataset to form 10 subsets, namely, subsets A–E for dataset 1 and subsets F–J for dataset 2. Table IV shows the details of these subsets. Each subset is used for two tests using NBCNN and CNN. Although sufficient raw data can be easily obtained in the laboratory, the fault data are usually difficult to be accessed in practical applications. Given this, this study considers the extreme conditions where only a small volume of data can be used for model training. To evaluate the effect of dataset size on the model training and prediction, the subsets with different sizes are generated.

TABLE IV
10 GROUPS OF SUBSETS

Subset	Superset	Training samples	Test samples	Classification number
A	Dataset 1	200	40	10
B	Dataset 1	500	100	10
C	Dataset 1	1000	200	10
D	Dataset 1	2000	400	10
E	Dataset 1	5000	1000	10
F	Dataset 2	160	16	8
G	Dataset 2	400	40	8
H	Dataset 2	800	80	8
I	Dataset 2	1600	160	8
J	Dataset 2	4000	400	8

IV. EXPERIMENTAL RESULT

To reduce the occasionality of experiments, 10 times of the comparative experiments for the subsets are repeated and the average accuracy is calculated. The algorithm is run on the

PyCharm Community Edition 2021 software and implemented on a computer with GPU of RTX 3060 with 12GB memory, and CPU of AMD 5600X with 6-Cores.

A. Results of Dataset 1

1) NBCNN accelerates Model Training

The five subsets of dataset 1 are inputted into both the NBCNN and the conventional CNN without noise injection. Fig. 5 shows the training accuracy and cross-entropy loss. In this experiment, differences in the first 500 epochs of training are highlighted.

Fig. 5 shows the training accuracy and cross-entropy loss w.r.t. the epoch of subsets A, B, C, D, and E, respectively. It can be seen that the performance of NBCNN is superior to that of conventional CNN without noise injection in both accuracy and cross-entropy loss. Meanwhile, with the increase in sample size (from subset A to E), the performance of the two CNNs improves accordingly. In this study, the number of epochs needed for a stable accuracy of over 90% is used to measure the convergence speed of CNN. Table V records the number of epochs needed for each subset of dataset 1 to reach 90% training accuracy. The results show that with the same subset, the convergence speed of NBCNN is 6–9 times that of CNN. The performance boosted by noise increases as the sample size decreases, which demonstrates the strong capability of the developed method in applications with a small-size dataset. Meanwhile, the convergence speed and cross-entropy loss of subsets A and B by NBCNN are similar to those of subsets D and E in CNN, respectively. Subsets A and B have only one-tenth samples as subsets D and E. The benefit of noise for CNN with small samples is similar to that with ten times samples, which is highly advantageous to industrial fault diagnosis where sufficient fault data can hardly be obtained.

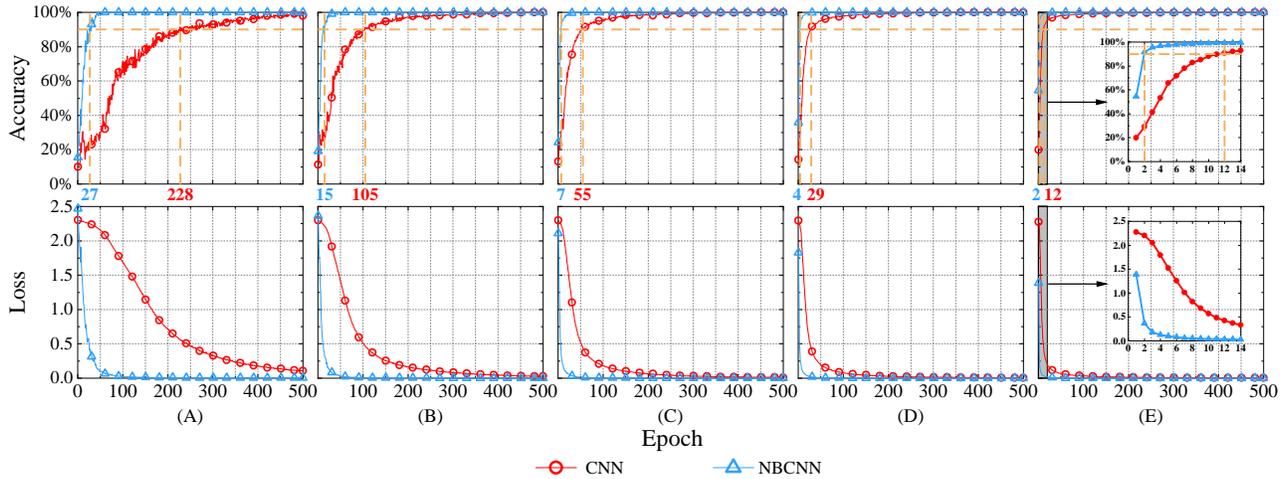


Fig. 5. Training accuracy and cross-entropy loss of subsets A, B, C, D, and E.

TABLE V
DATASET 1: NUMBER OF TRAINING EPOCHS TO REACH 90% ACCURACY

	Subset A	Subset B	Subset C	Subset D	Subset E
CNN	228	105	55	29	12
NBCNN	27	15	7	4	2

2) NBCNN Improves Fault Diagnosis Accuracy

It can be seen that the proposed NBCNN method effectively accelerates the model training procedure. This subsection

validates the fault diagnosis performance of the NBCNN model in comparison with three conventional methods, i.e., CNN without noise injection, 1D-CNN [37], and CANN [22]. In the 1D-CNN method, the raw vibration signal is directly inputted into the model. In the CANN method, the 1D vibration signal is transformed to a 2D image using continuous wavelet transform, and then the 2D image is inputted into the model. The key hyperparameters of the 1D-CNN and CANN are configured according to that in Refs. [22, 37], respectively.

The fault diagnosis accuracies under different numbers of training samples are evaluated and compared, and the results are shown in Fig. 6. It can be seen that when the number of training samples is 5000, the accuracy of the proposed method is 99.68%, which is higher than that of other compared models. When the number of training samples decreased to 200, the accuracy of all the models decreases accordingly. Nevertheless, the NBCNN still maintains the highest accuracy (92.75%), and the accuracy of the CNN, CANN, and 1D-CNN model is 85.25%, 73.00%, and 50.75%, respectively.

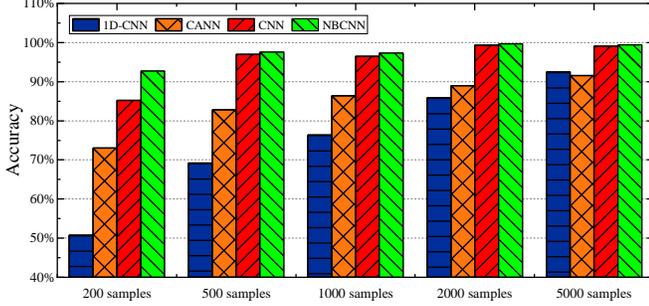


Fig. 6. Prediction accuracy of the four models for dataset 1.

When the number of samples is limited, the capacity of feature learning will affect the fault diagnosis accuracy. The proposed NBCNN model firstly pre-processes the original signal to make the signal features easier to be discriminated, and then injects approximate noise in the fully connected layer to avoid the over dependency on certain features. The randomness enhances the generalization ability and robustness of the model to some extent. In contrast, the traditional DL models require sufficient training samples to extract the fault features. When the number of samples is limited, the insufficient information embedding in the sample prevents an effective extraction of features. In this case, the model is difficult to converge during training. Subsequently, the performance of the DL models will degenerate significantly, which finally leads to a decrease in fault diagnosis accuracy. The distinct feature of the NBCNN model, i.e., requires fewer samples for training, makes it suitable for practical applications where the labeled data is hard to be obtained.

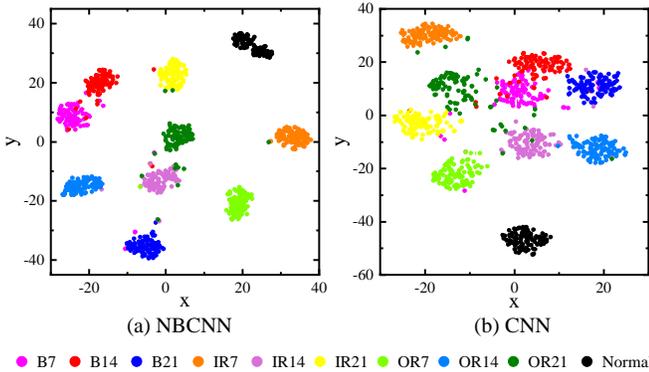


Fig. 7. T-SNE visualization of subset A: (a) NBCNN, and (b) CNN.

To further compare the effects of NBCNN and CNN on the accuracy of fault classification, the output features of subset A are visualized as a 2D distribution via t-SNE. Fig. 7 shows that the features extracted by NBCNN can effectively distinguish different types of faults. Different faults also have apparent

decomposition bands. For CNN, except for normal features, samples with the same fault are clustered loosely, and more overlaps are observed between different faults, such as B1 and B3 in Fig. 7(b).

B. Results of Dataset 2

In this subsection, dataset 2 is processed to further verify the robustness and generalization ability of the proposed NBCNN model. Dataset 1 from CWRU is composed of the motor bearing data with different fault types. Dataset 2 is composed of both the mechanical and electrical faults of the test motors.

The vibration data is collected from the experimental apparatus as shown in Fig. 4. Table VI shows the epochs required to reach training accuracy of 90% of dataset 2. In subset F, noise improves the training speed by up to 10 times. The convergence speeds of subsets F and G using NBCNN are comparable to subsets I and J using CNN, which shows significant improvement in convergence speed of NBCNN.

TABLE VI
DATASET 2: NUMBER OF TRAINING EPOCHS TO REACH 90% ACCURACY

	Subset F	Subset G	Subset H	Subset I	Subset J
CNN	210	88	47	25	11
NBCNN	21	13	7	4	3

Fig. 8 shows the prediction accuracy of different models including NBCNN, CNN, 1-D CNN, and CANN. The proposed NBCNN still shows considerable improvement when trained with a small number of samples. For instance, when the number of samples decreases to 160, the accuracy of NBCNN, CNN, 1D-CNN, and CANN is 97.50%, 93.75%, 93.75%, and 86.25%, respectively. The comparative results of the four models on dataset 1 and dataset 2 confirm the effectiveness and robustness of the proposed method. The NBCNN model can effectively diagnose both the mechanical and electrical faults in a motor, and also shows potential to be extended and applied to other datasets and fault diagnosis applications.

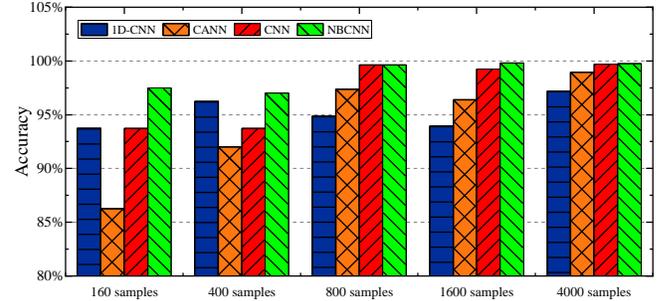


Fig. 8. Prediction accuracy of the four models in dataset 2.

C. Parameter Evaluation

The performance of the DL model is affected by both the data and model structure. The methods and strategies for determination of the proper CNN parameters have been widely investigated in the literature. For instance, Liao et al. investigated the relationship between CNN parameter distribution and CNN discrimination performance [38]. Zhang et al. proposed a high-precision CNN model with successive small kernels, and the parameters of the dropout and fully connected layers were evaluated and discussed [18]. Wen et al. firstly investigated the effect of learning rate on classification accuracy, and then proposed a new learning rate adjustment

strategy for CNN-based fault classification [39]. The NBCNN is an improved version of the classical CNN, and hence the optimization of the conventional parameters of the CNN can be referred to the references. The most distinct feature of the NBCNN is that a proper noise is injected into the specific layer of the model. Given this, the effects of the injected noise on the prediction accuracy are evaluated and discussed in this study.

The above experiments show that adding suitable noise can effectively improve the training and prediction performance of CNN. To investigate the influence of noise parameter σ on training results, we inject different noises into CNN. Equation (24) shows the sign of σ does not affect the magnitude of the noise, and thus only the case where $\sigma > 0$ needs discussion. The gain varies with the change of noise parameter on CNN training. Fig. 5 shows that the effect of noise on CNN training is mainly concentrated in the first 500 epochs of training, thus this gain can be measured with Equation (25)

$$DIF = \frac{\sum_{i=1}^{i=500} (B_i - A_i)}{500}, \quad (25)$$

where A_i represents the accuracy of the noiseless CNN at i th training; B_i indicates the NBCNN accuracy at i th training; and DIF measures the degree of influence of noise on the training. Fig. 9 shows the relationship between DIF and σ in subset A.

The dots in the upper half of Fig. 9 indicate that the NBCNN performs better than the CNN, while those in the lower half indicate that the noise reduces the convergence speed of CNN. From the results of the multiple tests in subset A, it can be seen that NBCNN can achieve the best performance when the noise parameter σ is set to 0.1. When σ exceeds 0.1, DIF decreases and converges to 0. This result is reasonable because as σ increases, the noise amplitude also approaches 0. This is equivalent to a noiseless CNN, and thus the accuracy difference DIF is 0. When σ is less than 0.1 and greater than 0.05, the noise has positive feedback on CNN. As σ continues to decrease and approach 0, the noise amplitude continues to increase and impairs the performance of CNN.

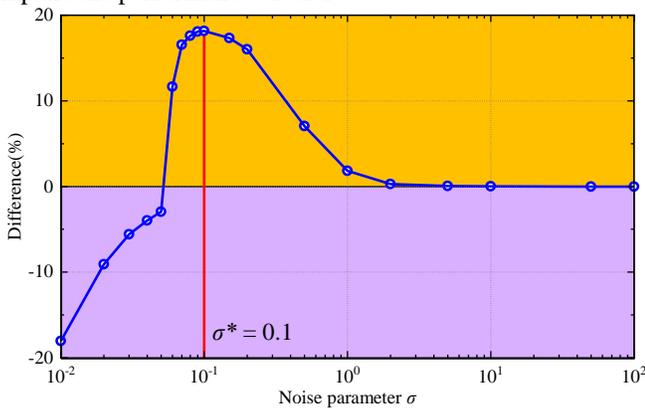


Fig. 9. Effect of different parameters σ on noise gain on subset A.

In addition, in all of the above experiments, although the noise waveforms are different in each test, the NBCNN can still yield the optimal result when $\sigma = 0.1$. For both dataset 1 and dataset 2, the relationship between the performance and noise parameter in each subset follows the pattern shown in Fig. 9. When the σ value is set around 0.1, the performance of NBCNN exceeds that of CNN, which means that CNN is sensitive to

noise with a certain intensity. When the noise intensity is too large, the original signal features would be masked which leads to a decrease in prediction accuracy. Besides the noise intensity, different types of noises such as colored noise, broadband noise, and narrow band noise, could also affect the performance of NBCNN. Additionally, the noises injected into different model layers can also affect the performance of the NBCNN method. These topics remain a further study in the future.

V. PERFORMANCE ON EDGE COMPUTING PLATFORM

The above experimental results indicate that the NBCNN has remarkable performance in both training and prediction procedures. Besides, the NBCNN model also has a small size and can be easily deployed onto resource-constrained edge devices for online and real-time motor fault recognition. For example, the total number of model parameters trained from subset A is less than 40,000 and the model size is less than 200 Kbytes, and this model is implemented onto the edge platform to evaluate its actual performance.

The effectiveness of the proposed method on edge computing platforms is verified by 10 independent comparative experiments of subset A run on an NVIDIA Jetson TX1 edge device with 4G memory. Fig. 10 shows the experimental setup. Apart from the comparisons on training and prediction performance, the actual time the algorithm runs on edge computing platforms is also evaluated.

Fig. 11 plots the accuracy and cross-entropy loss variation of the CNN and NBCNN training in Jetson TX1. The epochs required for the accuracies of model training to reach 90% are 255 and 25, respectively. This result is consistent with the conclusion of the convergence speed gap of up to 10 times in the previous subsections on experiments on the desktop GPU. In addition, the overall accuracy and cross-entropy loss of NBCNN are better than those of CNN.

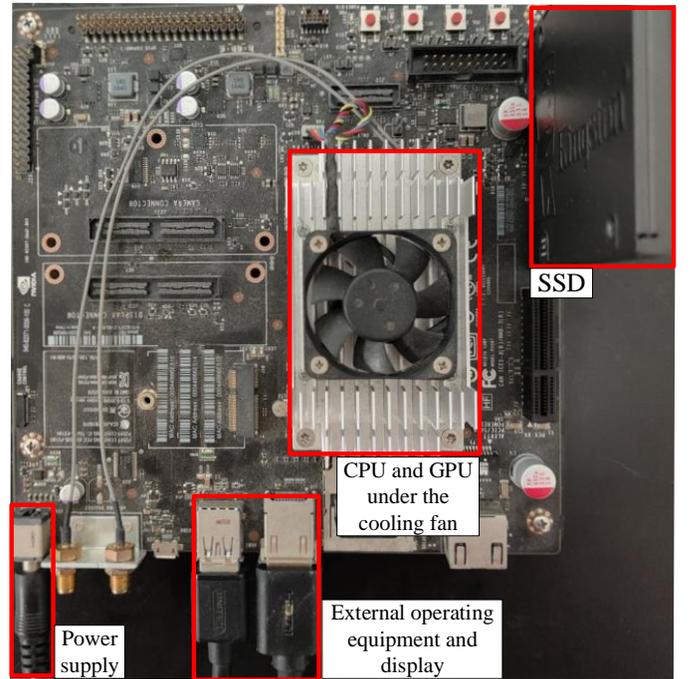


Fig. 10. Edge computing platform.

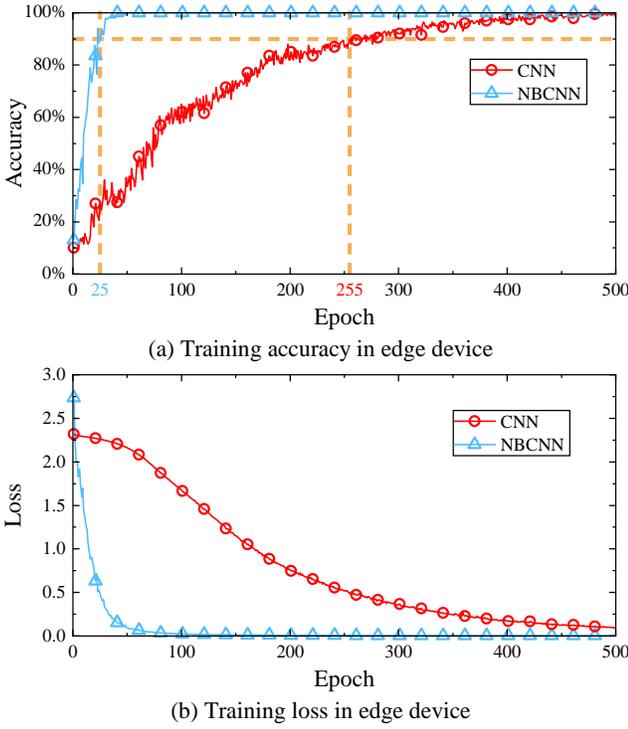


Fig. 11. Model trained on edge devices of subset A: (a) accuracy and (b) cross-entropy loss.

Deploying CNN in resource-constrained edge devices must also consider algorithm execution time in addition to basic performance requirements. Fig. 12(a) shows the time spent for each epoch of CNN and NBCNN training. NBCNN has a higher training time cost than CNN in each epoch, a reasonable result because NBCNN is realized by adding noise on the basis of the CNN model. However, to consider only the time spent on each epoch of training has no practical significance. The actual training time to reach 90% accuracy can be calculated by considering the number of training epochs, and the results are shown in Fig. 12(b). Both 6.6 s on edge device and 1.5 s on the computer for NBCNN are more than nine times faster than CNN's 60.5 s and 14.5 s respective results.

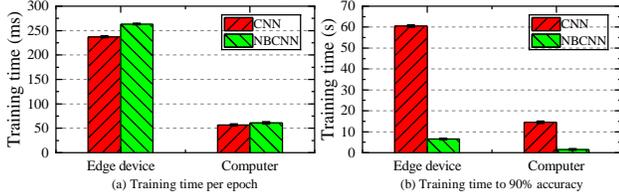


Fig. 12. Training time of CNN and NBCNN of subset A on edge devices: (a) per epoch, and (b) to 90% accuracy.

The comparison in fault prediction performance between the two models is also tested 10 times and the results are shown in Fig. 13. The inference time of the NBCNN model is slightly higher than that of the CNN method due to the fact that external noise is injected into the model, as shown in Fig. 13(a). Besides, the inference time on the edge device is also higher than that on the desktop GPU, as the edge device has lower GPU frequency and memory. Nevertheless, the average time for processing each frame of signal is only 36.5 ms. Such a time consumption is fast enough for real-time fault diagnosis.

The prediction accuracy of NBCNN is significantly higher than that of CNN, with an average of 93.25% for NBCNN compared with only 84.75% for CNN, and their difference is 6.5%. It should be noticed that Subset A has only 200 training samples, which is the smallest sample size among the subsets of dataset 1. Based on the previous experimental results, the prediction results of this model can represent the worst performance of NBCNN. Even in this extreme situation, the NBCNN can still obtain a considerable accuracy, which demonstrates its robustness and reliability.

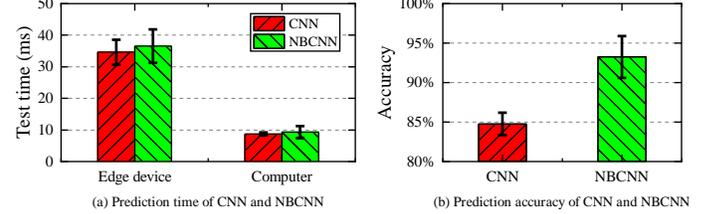


Fig. 13 Prediction performance comparison of CNN and NBCNN on edge devices: (a) prediction time and (b) prediction accuracy.

In addition, the TX1 has much fewer computational resources in comparison with the typical DL device such as a desktop GPU. Under this circumstance, the above experimental results indicate that the NBCNN model can still easily execute the computation on this edge device. In the future, the NBCNN can be tested on cheaper devices with less computational resources, which could further improve the scalability and flexibility for real-time motor fault diagnosis.

VI. CONCLUSIONS

This paper designs an improved CNN model to solve the problem of limited training samples in industrial applications. Adding the proper amount of noise to the CNN can speed up training and improve inference accuracy. Experiments on two datasets show that NBCNN could significantly improve the convergence speed compared with the conventional CNN without noise injection. In addition, the prediction accuracy is also significantly improved with NBCNN in comparison with several state-of-the-art models. Experiments in resource-constrained edge platforms also show that the proposed method achieves better performance. The proposed method is widely applicable to machine fault diagnosis where fault data is typically difficult to be collected.

REFERENCES

- [1] C. Ding, M. Zhao, J. Lin, J. Jiao, and K. Liang, "Sparsity-Based Algorithm for Condition Assessment of Rotating Machinery Using Internal Encoder Data," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 9, pp. 7982-7993, 2020.
- [2] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric Locomotive Bearing Fault Diagnosis Using a Novel Convolutional Deep Belief Network," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2727-2736, 2018.
- [3] Q. Guo, Y. Li, Y. Song, D. Wang, and W. Chen, "Intelligent Fault Diagnosis Method Based on Full 1-D Convolutional Generative Adversarial Network," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2044-2053, 2020.
- [4] Z. Long, X. Zhang, M. He, S. Huang, G. Qin *et al.*, "Motor Fault Diagnosis Based on Scale Invariant Image Features," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1605-1617, 2022.
- [5] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of CNN and RNN for natural language processing," *arXiv preprint arXiv:1702.01923*, 2017.

- [6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017.
- [7] Z. Liu, Z. Wu, T. Li, J. Li, and C. Shen, "GMM and CNN Hybrid Method for Short Utterance Speaker Recognition," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3244-3252, 2018.
- [8] R. Socher, Y. Bengio, and C. D. Manning, "Deep learning for NLP (without magic)," in *Tutorial Abstracts of ACL 2012*, 2012, pp. 5-5.
- [9] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82-97, 2012.
- [10] M. Ma, C. Sun, and X. Chen, "Deep Coupling Autoencoder for Fault Diagnosis With Multimodal Sensory Data," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 3, pp. 1137-1145, 2018.
- [11] W. Du, M. Kang, and M. Pecht, "Fault Diagnosis Using Adaptive Multifractal Detrended Fluctuation Analysis," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 3, pp. 2272-2282, 2020.
- [12] J. Song, J. Zhao, X. Zhang, F. Dong, J. Zhao *et al.*, "Accurate Demagnetization Faults Detection of Dual-Sided Permanent Magnet Linear Motor Using Enveloping and Time-Domain Energy Analysis," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6334-6346, 2020.
- [13] J. Zhang, Y. Wang, K. Zhu, Y. Zhang, and Y. Li, "Diagnosis of Interturn Short-Circuit Faults in Permanent Magnet Synchronous Motors Based on Few-Shot Learning Under a Federated Learning Framework," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8495-8504, 2021.
- [14] Z.-H. Liu, B.-L. Lu, H.-L. Wei, L. Chen, X.-H. Li *et al.*, "Deep Adversarial Domain Adaptation Model for Bearing Fault Diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 7, pp. 4217-4226, 2021.
- [15] H. Wang, J. Xu, C. Sun, R. Yan, and X. Chen, "Intelligent Fault Diagnosis for Planetary Gearbox Using Time-Frequency Representation and Deep Reinforcement Learning," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 2, pp. 985-998, 2022.
- [16] F. Husari and J. Seshadriath, "Incipient Interturn Fault Detection and Severity Evaluation in Electric Drive System Using Hybrid HCNN-SVM Based Model," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1823-1832, 2022.
- [17] T. Xie, X. Huang, and S.-K. Choi, "Intelligent Mechanical Fault Diagnosis Using Multisensor Fusion and Convolution Neural Network," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3213-3223, 2022.
- [18] J. Zhang, Y. Sun, L. Guo, H. Gao, X. Hong *et al.*, "A new bearing fault diagnosis method based on modified convolutional neural networks," *Chinese Journal of Aeronautics*, vol. 33, no. 2, pp. 439-447, 2020.
- [19] L. Ren, J. Dong, X. Wang, Z. Meng, L. Zhao *et al.*, "A Data-Driven Auto-CNN-LSTM Prediction Model for Lithium-Ion Battery Remaining Useful Life," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3478-3487, 2021.
- [20] H. Shao, M. Xia, G. Han, Y. Zhang, and J. Wan, "Intelligent Fault Diagnosis of Rotor-Bearing System Under Varying Working Conditions With Modified Transfer Convolutional Neural Network and Thermal Images," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3488-3496, 2021.
- [21] D. Peng, H. Wang, Z. Liu, W. Zhang, M. J. Zuo *et al.*, "Multibranch and Multiscale CNN for Fault Diagnosis of Wheelset Bearings Under Strong Noise and Variable Load Condition," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4949-4960, 2020.
- [22] M.-Q. Tran, M.-K. Liu, Q.-V. Tran, and T.-K. Nguyen, "Effective Fault Diagnosis Based on Wavelet and Convolutional Attention Neural Network for Induction Motors," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-13, 2022.
- [23] F. Wang, R. Liu, Q. Hu, and X. Chen, "Cascade Convolutional Neural Network With Progressive Optimization for Motor Fault Diagnosis Under Nonstationary Conditions," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2511-2521, 2021.
- [24] J. Xiong, C. Li, C.-D. Wang, J. Cen, Q. Wang *et al.*, "Application of Convolutional Neural Network and Data Preprocessing by Mutual Dimensionless and Similar Gram Matrix in Fault Diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1061-1071, 2022.
- [25] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, Opportunities, and Directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724-4734, 2018.
- [26] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4177-4186, 2020.
- [27] C. Yin, J. Xi, R. Sun, and J. Wang, "Location Privacy Protection Based on Differential Privacy Strategy for Big Data in Industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3628-3636, 2018.
- [28] M. Zhao, S. Zhong, X. Fu, B. Tang, and M. Pecht, "Deep Residual Shrinkage Networks for Fault Diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4681-4690, 2020.
- [29] K. Audhkhasi, O. Osoba, and B. Kosko, "Noise-enhanced convolutional neural networks," *Neural Netw.*, vol. 78, pp. 15-23, Jun 2016.
- [30] Y. M. Yeap, N. Geddada, K. Satpathi, and A. Ukil, "Time- and Frequency-Domain Fault Detection in a VSC-Interfaced Experimental DC Test System," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4353-4364, 2018.
- [31] J. Yu and X. Zhou, "One-Dimensional Residual Convolutional Autoencoder Based Feature Learning for Gearbox Fault Diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6347-6358, 2020.
- [32] S. Lu, G. Qian, Q. He, F. Liu, Y. Liu *et al.*, "In Situ Motor Fault Diagnosis Using Enhanced Convolutional Neural Network in an Embedded System," *IEEE Sensors Journal*, vol. 20, no. 15, pp. 8287-8296, 2020.
- [33] P. Peng, J. Lu, T. Xie, S. Tao, H. Wang *et al.*, "Open-Set Fault Diagnosis via Supervised Contrastive Learning with Negative Out-of-Distribution Data Augmentation," *IEEE Transactions on Industrial Informatics*, doi:10.1109/tii.2022.3149935, 2022.
- [34] L. Xiao, J. X. Tang, X. H. Zhang, E. Bechhoefer, and S. Y. Ding, "Remaining useful life prediction based on intentional noise injection and feature reconstruction," *Reliability Engineering & System Safety*, vol. 215, Nov 2021, Art no. 107871.
- [35] J. Tang, N. Wu, L. Xiao, and J. Zhao, "The Improvement of Remaining Useful Life Prediction for Aero-engines by Injecting Noise," in *2020 Global Reliability and Prognostics and Health Management (PHM-Shanghai)*, 2020: IEEE, pp. 1-6.
- [36] W. A. Smith and R. B. Randall, "Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study," *Mechanical systems and signal processing*, vol. 64, pp. 100-131, 2015.
- [37] C. Wu, P. Jiang, C. Ding, F. Feng, and T. Chen, "Intelligent fault diagnosis of rotating machinery based on one-dimensional convolutional neural network," *Computers in Industry*, vol. 108, pp. 53-61, 2019.
- [38] L. Liao, Y. Zhao, S. Wei, Y. Wei, and J. Wang, "Parameter Distribution Balanced CNNs," *IEEE Trans Neural Netw Learn Syst*, vol. 31, no. 11, pp. 4600-4609, Nov 2020.
- [39] L. Wen, X. Li, and L. Gao, "A New Reinforcement Learning Based Learning Rate Scheduler for Convolutional Neural Network in Fault Classification," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 12, pp. 12890-12900, 2021.

Lv Chen received the B.S. degree in electrical engineering from Anhui University, Hefei, China, in 2020.

He is currently working toward the M.S. degree in control engineering with the College of Electrical Engineering and Automation, Anhui University, Hefei, China. His research interest includes signal processing and deep learning-based machine fault diagnosis.



Kang An received the B.S. degree in mechanical design manufacture and automation from Anhui University, Hefei, China, in 2020.

He is currently working toward the M.S. degree in mechanical engineering with the College of Electrical Engineering and Automation, Anhui University, Hefei, China. His research interest includes signal processing-based machine fault diagnosis.





Dali Huang received the M.S. degree in Materials Processing Engineering from Beijing Research Institute of Mechanical & Electrical Technology, Beijing, China, in 2021.

He is currently an engineer with the Beijing Research Institute of Mechanical & Electrical Technology, Beijing, China. His research interest includes plastic forming equipment fault diagnosis.



Xiaoxian Wang (Member, IEEE) received the B.S. degree from the Shandong University of Science and Technology, Qingdao, China, in 2010, and the M.S. degree in 2013 from the University of Science and Technology of China, Hefei, China, where she is currently working toward the Ph.D. degree with the Department of Precision Machinery and Precision Instrumentation, all in mechanical engineering.

She is currently an Engineer with the College of Electronics and Information Engineering, Anhui University, Hefei. Her research interests include innovative design and intelligent maintenance of

electromechanical system.



Min Xia (Senior Member, IEEE) received the B.S. degree in industrial engineering from Southeast University, Jiangsu, China, in 2009, the M.S. degree in precision machinery and precision instrumentation from the University of Science and Technology of China, Anhui, China, in 2012, and the Ph.D. degree in mechanical engineering from the University of British Columbia, Vancouver, BC, Canada, in 2017.

He is currently a Lecturer with the Department of Engineering, Lancaster University, Lancaster, U.K. He served as an Associate Editor for *IEEE Transactions on Instrumentation and Measurement*.

His research interests include smart manufacturing, machine diagnosis and prognosis, deep neural networks, autonomous vehicle, and sensor fusion.



Siliang Lu (Senior Member, IEEE) received the B.S. and Ph.D. degrees in mechanical engineering from the University of Science and Technology of China, Hefei, China, in 2010 and 2015, respectively.

He is currently an Associate Professor with the College of Electrical Engineering and Automation, Anhui University, Hefei. He served as an Associate Editor for *IEEE Transactions on Instrumentation and Measurement*, and an editorial board member for *Journal of Dynamics, Monitoring and Diagnostics*.

His research interests include machinery-condition-based monitoring and fault diagnosis, signal processing, IoT and edge computing, and robotics.