

Research Article

DC-NNMN: Across Components Fault Diagnosis Based on Deep Few-Shot Learning

Juan Xu,¹ Pengfei Xu,¹ Zhenchun Wei,¹ Xu Ding,² and Lei Shi¹ 

¹School of Computer and Information Science, Hefei University of Technology, Hefei 230009, China

²Institute of Industry and Equipment Technology, Hefei University of Technology, Hefei 230009, China

Correspondence should be addressed to Lei Shi; shilei@hfut.edu.cn

Received 21 January 2020; Revised 16 September 2020; Accepted 9 November 2020; Published 17 December 2020

Academic Editor: Nuno M. Maia

Copyright © 2020 Juan Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, deep learning has become a popular topic in the intelligent fault diagnosis of industrial equipment. In practical working conditions, how to realize intelligent fault diagnosis in the case of the different mechanical components with a tiny labeled sample is a challenging problem. That means training with one component sample but testing with another component sample has not been resolved. In this paper, we propose a deep convolutional nearest neighbor matching network (DC-NNMN) based on few-shot learning. The 1D convolution embedding network is constructed to extract the high-dimensional fault feature. The cosine distance is merged into the K -Nearest Neighbor method to model the distance distribution between the unlabeled sample from the query set and labeled sample from the support set in high-dimensional fault features. The multiple few-shot learning fault diagnosis tasks as the testing dataset are constructed, and then the network parameters are optimized through training in multiple tasks. Thus, a robust network model is obtained to classify the unknown fault categories in different components with tiny labeled fault samples. We use the CWRU bearing vibration dataset, the bearing vibration data selected from the Lab-built experimental platform, and another gearing vibration dataset for across components experiment to prove the proposed method. Experimental results show that the proposed method can achieve fault diagnosis accuracy of 82.19% for gearing and 82.63% for bearings with only one sample of each fault category. The proposed DC-NNMN model provides a new approach to solve the across components fault diagnosis in few-shot learning.

1. Introduction

In complex industrial systems, fault diagnosis is an important issue to ensure the safety of equipment and personnel [1, 2]. In recent years, the ability of deep neural network models to learn fault features of a large number of samples has been well known and widely used in the field of fault diagnosis [3, 4]. However, the success of deep learning-based fault diagnosis depends on the following two conditions: (1) massive amounts of labeled fault data; (2) training data and testing data which have the same category space and consistent distribution [5–7].

At present, many scholars focus on the fault diagnosis with limited labeled samples. The method of transfer learning has been introduced in recent years, which uses existing knowledge in the source domain to solve fault

classification in the different target domains. Lu et al. [8] proposed a deep neural network model with domain adaption to realize fault diagnosis under different loads. Wen et al. [9] proposed a Deep Transfer Learning method of rolling bearing fault diagnosis with unlabeled target domain data, which minimizes the loss of difference between features of training and test data using maximum mean discrepancy. Hang et al. [10] proposed a Principal Component Analysis (PCA) method based on the improved SMOTE algorithm and applied PCA to the field of high-dimensional imbalance fault data. In order to increase the size of the sample set.

Many scholars have used the idea of GAN to realize the generation of vibration samples for fault diagnosis. Cabrera et al. [11] used the GANs model to evaluate the data distribution of each minority failure mode. Zhao et al. [12] proposed a switchable normalized semisupervised

generative fault diagnosis method network, by generating samples to assist the model training. Then, the problem of insufficient label of fault samples under test conditions can be solved.

The above studies can solve the problem of fault diagnosis with insufficient labeled data when the training set and testing set have the same category space in deep networks. The model trained by the labeled data of one component cannot be able to classify other component fault categories, because even though the labeled data can be obtained from some other components, the fault category space and data distribution of different components are different; we call it across components fault diagnosis.

Few-shot learning is committed to understanding new categories from a few examples, and it is a very popular topic in the field of image classification. Some implementation approaches include model-based, metric-based, and optimization-based methods. The model-based methods aim to quickly update the parameters with a small number of samples through the design of the model structure and directly establish a mapping function of the input x and the predicted value p , such as memory enhancement methods [13] and Meta Network [14].

The metric-based method completes the classification by measuring the distance between the samples in the batch set and the samples in the support set. The typical methods based on metrics are the Siamese Network [15], the Match Network [16], the Prototype Network [17], and so on.

The optimization-based methods are represented by Finn et al. [18], who proposed that ordinary gradient descent methods were difficult to fit in few-shot scenarios. The idea of optimization-based methods completes the task of few-shot classification by adjusting the optimization method, so the methods are not limited to the size of the parameters and the model architecture.

However, because of the difference distribution between image data and vibration data, the existing few-shot learning models cannot be well adapted in the field of fault diagnosis. Thus, this paper proposes a across components few-shot learning fault diagnosis method based on matching network, and the model is verified through a series of experiments. The main insights and contributions of this study are summarized as follows:

- (1) We propose an intelligent fault diagnosis method based on deep convolutional nearest neighbor matching networks (DC-NNMN). A four-layer convolutional network is designed to extract high-dimensional fault features. The cosine distance is merged into the K -Nearest Neighbor method to model the distance distribution between the unlabeled sample from query set and labeled sample from support set in high-dimensional fault features, so that the fault samples of the same category are close to each other and the samples of different categories are far away. The query set and support set samples of one component are decomposed into different meta tasks to learn the generalization ability of the model when the fault category changes; then, the unknown

fault category of another component can be classified without changing the network model.

- (2) We use the Case Western Reserve University (CWRU) bearing vibration datasets as the training set and the bearing vibration data selected from Lab-built experimental platform and another gearing vibration dataset, respectively, as the testing set for our experiment to prove the feasibility of the proposed method. Experimental results prove that the model trained by bearing fault data has achieved accurate fault classification on the new fault category of both bearing and gearing. The proposed method implements across components fault diagnosis with tiny fault samples.

The rest of the paper is organized as follows. Section 2 introduces the preliminaries of DC-NNMN. Section 3 details the proposed deep convolution nearest neighbor matching network model (DC-NNMN), including problem description, model structure, and optimization objectives. In Section 4, experimental verification and corresponding analysis are conducted. The conclusions are drawn in Section 5.

2. Preliminaries

2.1. Few-Shot Learning. The main challenge of few-shot learning is how to understand new categories from a few examples. Specifically, the training set of few-shot learning contains many categories, and each category has multiple samples. In the training phase, c categories are randomly selected in the training set, and each category selects k samples (a total of $c * k$ samples) as the support set S . Then selecting k' from the remained data in the c categories samples serves as the query set Q for the model. The goal of the model is to minimize the prediction loss on the query set Q , by giving the support set S as input. That is, the model is required to learn how to distinguish these c classes from the $c * k$ samples in the support set. Such a task is called a C -way k -shot problem. In few-shot learning, k is usually less than 20. S and Q can be expressed as follows:

$$\begin{aligned} S &= \{(X_i, Y_i)\}_{i=1}^{c*k}, \\ Q &= \{(X'_i, Y'_i)\}_{i=1}^{k'}. \end{aligned} \quad (1)$$

2.2. K-Nearest Neighbor. K -Nearest Neighbor (KNN) was originally proposed by Cover and Hart in 1968 [19]. It is a relatively mature nonparametric statistical method for classification and regression. The core idea is that if most of the K -Nearest Neighbors of a sample in the feature space belong to a certain category, the sample also belongs to this category. Take a set of data with known labels $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, where x_i is the feature vector of the sample i . and y_i is its label, $y_i = c_1, c_2, \dots, c_k$. For the training sample (x, y) , the KNN algorithm searches for the K instances that are closest to x based on the given distance metric, denoted as $x'_i, i = 1, 2, \dots, K$. Then calculate the label of the sample x to be tested based on the decision rule:

$$y = \arg \max_{x_i \in \mathcal{X}_i} \sum I(y_i = c_j), \quad i = 1, 2, \dots, n; j = 1, 2, \dots, k, \quad (2)$$

where I is the distance for measuring similarity. Therefore, after the distance metric is determined, the K -Nearest Neighbor algorithm has only one parameter of K . How to choose an optimal K value depends on the dataset itself. As shown in Figure 1, a red circle is the test sample, if $K=3$, it is classified as a green square, and if $K=5$, it is classified as a yellow triangle. It has the advantages of simplicity, easy to understand, easy to implement, no need to estimate parameters, and no training. It is especially suitable for multiclassification problems.

3. Proposed Method

3.1. Problem Description. In this paper, the idea of few-shot learning based on Match Network is used to the fault diagnosis across category spaces. We define across components few-shot learning fault diagnosis problem as follows:

- (1) The mechanical component A (MCA) and mechanical component B (MCB) are different components with different fault categories.
- (2) The training set $T = \{X_i, Y_i\}_{i=1}^n$ of MCA contains many categories of labeled fault samples, X_i is the vibration data, Y_i is the corresponding fault labels, and n is the number of data.
- (3) Given a support set of MCB $S = \{(X_i, Y_i)\}_{i=1}^{c*k}$, which contains c different fault categories, each category contains k samples, $k \leq 20$. Given a query set $Q = \{(X'_i, Y'_i)\}_{i=1}^k$, the data in query set Q has the same categories with support set S .
- (4) T and S have different feature spaces χ and category spaces Y :

$$\begin{aligned} \chi^S &\neq \chi^T, \\ Y^S &\neq Y^T. \end{aligned} \quad (3)$$

- (5) A support set $T_s = \{(X_i, Y_i)\}_{i=1}^{c*k}$ and a query set $T_q = \{(X'_i, Y'_i)\}_{i=1}^k$ randomly selected from the training set T . Among them, T_s is the same as S and T_q is the same as Q . During training, each task randomly selects T_s and T_q to train the fault diagnosis model and repeats the task many times to achieve model training at the metalevel.

Therefore, our goal is to train the model using T_s and T_q of fault MCA vibration sample to classify each new class in Q according to the set S of fault MCB. The main idea of the problem description of across components is shown in Figure 2.

3.2. Deep Convolution Nearest Neighbor Matching Network. This paper proposes a deep convolutional neighbor matching network (DC-NNMN) to learn a support set S

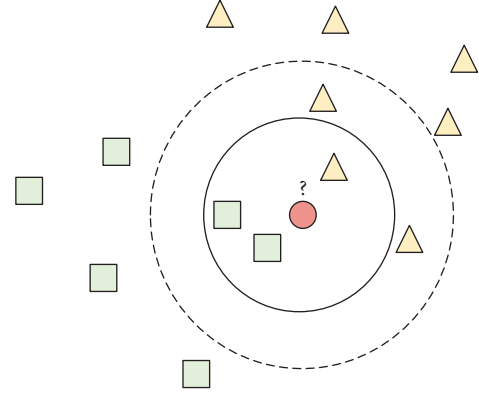


FIGURE 1: An example of K-NN.

with labeled fault samples and then classify the fault samples in query dataset Q .

As shown in Figure 3, the model proposed in this paper contains two parts: the embedding module f_φ and the matching module g_φ . In the embedding module, we use a convolution network to complete the map from the input space of the sample to the feature space, using the K -Nearest Neighbor algorithm to complete the matching from the feature space to the category space, so as to achieve the fault classification task.

$$\begin{aligned} f: x &\longrightarrow z, \\ g: z &\longrightarrow y. \end{aligned} \quad (4)$$

The features in the time-domain vibration sample have translation invariance; that is, a certain statistical feature in the sample may appear at any time. Convolutional neural networks have the characteristics of local connections and weights sharing, so convolution operations are particularly suitable for processing time-domain vibration samples. As shown in Table 1, we adopt a neural network with a four-layer convolution operation as our embedded module to extract the feature information of each fault sample. Because the number of samples is too small, in order to prevent overfitting, the fully connected layer after the traditional convolution operation is canceled, reducing the parameters that the network model needs to train. The first layer is the input layer, and the size of the input fault sample is $L \times 1$. Each subsequent convolution operation includes a convolution and a batch regularization. The size of the convolution kernel is s and the number of convolution kernels is n . The activation function is the Leaky ReLU activation function. In addition, the first and second layers add an additional max-pooling layer after the convolution operation. The convolution operations of the first two layers are as follows:

$$\begin{aligned} m^d &= \text{Max_pool}(W^d * x + b^d), \\ h^d &= \text{LReLU}(\text{BN}(m^d)), \end{aligned} \quad (5)$$

where $*$ represents convolution operation, W^d and b^d represent convolution kernel and bias, h^d is the result of convolution operation, and d represents d -th layer of network. The last two layers are

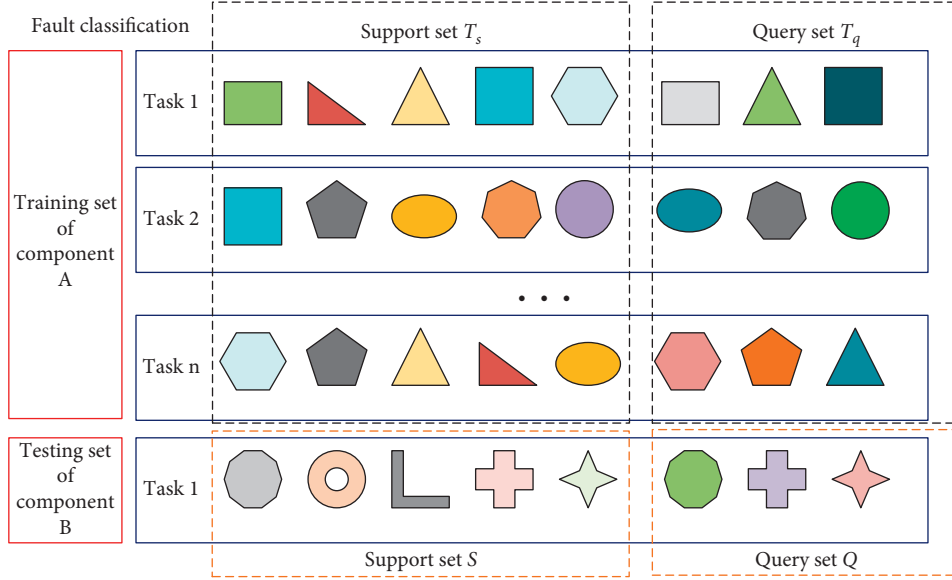


FIGURE 2: Problem description of cross components fault diagnosis.

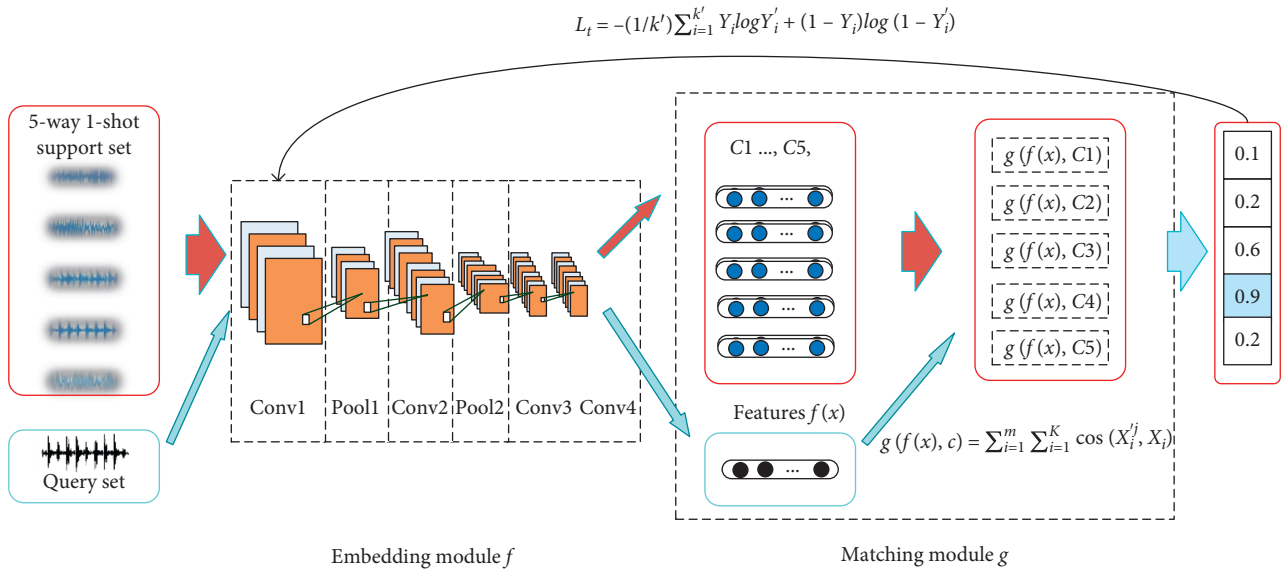


FIGURE 3: The structure of the proposed model.

$$h^d = \text{LReLU}(\text{BN}(W^d * h^d + b^d)). \quad (6)$$

In this way, after a four-layer convolution operation, a feature vector with a size of $L/4 \times 1 \times n$ is obtained, which can be expressed as

$$f(x) = [x_1, \dots, x_m] \in R^{n \times m}. \quad (7)$$

The matching module is mainly to use the deep feature descriptions of all fault samples in a category to construct the local feature space for fault classification. If we directly use a limited amount of data to train a classifier on a few-shot learning task, the model will almost certainly overfitting. There are tens of thousands of parameters in the neural network classifier which need to be optimized. Instead,

many nonparametric methods are more suitable. Considering the discreteness of the fault vibration sample, the KNN algorithm is used to verify the spatial distance between the samples of the query set and this category in the support set, as shown in Figure 2.

Specifically, each sample q from the query set Q is processed by the embedding module to obtain $f(q) = [x_1, \dots, x_m] \in R^{k \times m}$. K -Nearest Neighbors in a category c for x_i in turn are found and get x_i^j , ($j = 1, \dots, K$). Then, we calculate the distance between each of the nearest neighbors x_i^j and x_i and finally add the distances of m local features to their K -Nearest Neighbors to get a similarity of sample q on the query set Q that matches the category c :

TABLE 1: Architecture of CNN.

Layer	Symbol	Operate	Parameter size
1	Input	Input samples	864
2	Conv1	Convolution	$3 \times 1 \times 64$
		Batch normalization	—
3	Pool1	Leaky ReLU	Leaky
		Max pooling	2×1
4	Conv2	Convolution	$3 \times 1 \times 64$
		Batch normalization	—
5	Pool2	Leaky ReLU	Leaky
		Max pooling	2×1
6	Conv3	Convolution	$3 \times 1 \times 64$
		Batch normalization	—
7	Conv4	Leaky ReLU	Leaky
		Max pooling	2×1
8	Output	Output features	$216 \times 1 \times 64$

$$g(f(q), c) = \sum_{i=1}^m \sum_{j=1}^K \cos(x_i^j, x_i). \quad (8)$$

The cosine of the angle between the two vectors is used to measure the correlation between them. The cosine distance can reduce the sensitivity to absolute values, which is suitable for measuring the distance between discrete data. The cosine similarity of the vectors x_i^j and x_i is

$$\cos(x_i^j, x_i) = \frac{x_i^j \cdot x_i}{|x_i^j| \cdot |x_i|}. \quad (9)$$

3.3. Optimization Objective. In this paper, the number of labeled fault samples known as MCB is less than 20. If we train the limited number of labeled samples directly, the model will inevitably fall into overfitting and fail to accurately classify faults.

The episodic training mechanism [16] has been demonstrated as an effective approach to learn the transferable knowledge from the training dataset. Specifically, in each iteration, we use the constructed training set to construct a data structure similar to that in the test set. So, the network is trained through N tasks. For each task, there has two inputs, namely, support set S and query set Q . The feature information of each sample is obtained through the processing of the embedded module, and they are matched with the correct category according to the matching module. For the model, we hope that, at each task, the network can try to have a good classification effect on the samples in Q ; that is, $g(f(q), c)$ can match the correct category. The output of the network is considered as a value of 0 to 1.0 which means very dissimilar, while 1 means completely similar. In this way, for each sample in Q , a predicted value for the real category is got. This predicted value can be used to build a cross-entropy loss function for a single task; that is,

$$L_t = -\frac{1}{k'} \sum_{i=1}^{k'} Y_i \log Y_i' + (1 - Y_i) \log(1 - Y_i'), \quad (10)$$

where t represents the t -th training task, Y_i represents the true label of the i -th sample, and Y_i' represents the predicted label obtained through the network. For N tasks, the total loss function is

$$L = \sum_{t=1}^N L_t. \quad (11)$$

During the training process, the loss function L is minimized through backward transfer and gradient descent. We adopt the adaptive moment estimation method to update the parameters of the model in this paper. The algorithm can calculate the adaptive learning rate of each parameter, and the convergence speed is fast. At the same time, it can correct problems in other optimization techniques, such as the disappearance of the learning rate, slow convergence, or the large variance of the loss functions caused by the update of high variance parameters. The parameter update rules are as follows:

$$\begin{aligned} m_t' &= \frac{m_t}{1 - \beta_1^t}, \\ v_t' &= \frac{v_t}{1 - \beta_2^t}, \\ \phi_t' &= \phi_t - \frac{\mu}{\sqrt{v_t' + \varepsilon}} m_t'. \end{aligned} \quad (12)$$

In the above equation, φ is the characterization of the convolutional network parameters, m_t is the average value of the first moment of the gradient, and v_t is the noncenter variance value of the second moment of the gradient. The variance at the two moments, μ , is the learning rate of the model, ε is an infinitesimal small amount 10^{-8} , and β_1 and β_2 are two parameters of the Adam optimizer. The pseudocode of the algorithm is shown in Table 2.

3.4. Fault Diagnosis Based on DC-NNMN. The flowchart of the proposed fault diagnosis method is shown in Figure 4. It mainly includes three steps: the construction of the training dataset, the model building and training, and the testing of the fault samples.

- (1) In the step of training dataset construction, many different categories of labeled vibration samples of fault bearing need to be used. According to the setting form of few-shot learning dataset, support set T_s and query set T_q of C -way k -shot are randomly selected.
- (2) In the step of model building and training, the model as shown in Figure 2 is built firstly. Then, we send the dataset extracted each time to the network for training and record it as a task. After N times of training, the parameters involved in our model are fixed.

TABLE 2: Algorithm pseudocode.

Algorithm: deep convolution near neighbor matching networks

$\alpha = 0.005, \beta_1 = 0.5, \beta_2 = 0.9, k'$ is the number of query set

Require: $p(t)$: Distribution over tasks

Require: $\mu, \beta_1, \beta_2, \varepsilon = 10^{-8}$: step size hyperparameters

(1) randomly initialize φ

(2) **while** not done **do**

(3) Sample batch of tasks $t \sim p(t)$

(4) **for all** t **do**

(5) Evaluate $\nabla_{\varphi} L_t(f_{\varphi})$ with respect to k' examples

(6) Compute adapted parameters with Adam:

$$m'_t = m_t / (1 - \beta_1^t)$$

$$v'_t = v_t / (1 - \beta_2^t)$$

$$\phi'_t = \phi_t - (\mu / (\sqrt{v'_t} + \varepsilon)) m'_t$$

(7) **end for**

(8) Update:

$$\nabla_{\phi} L \leftarrow \nabla_{\phi} \sum_{i=1}^N (-1/k') \sum_{i=1}^{k'} Y_i \log Y_i + (1 - Y_i) \log(1 - Y_i)$$

$$\varphi \leftarrow \text{Adam}(\nabla_{\phi} L, \mu, \beta_1, \beta_2, \varepsilon = 10^{-8})$$

(9) **end while**

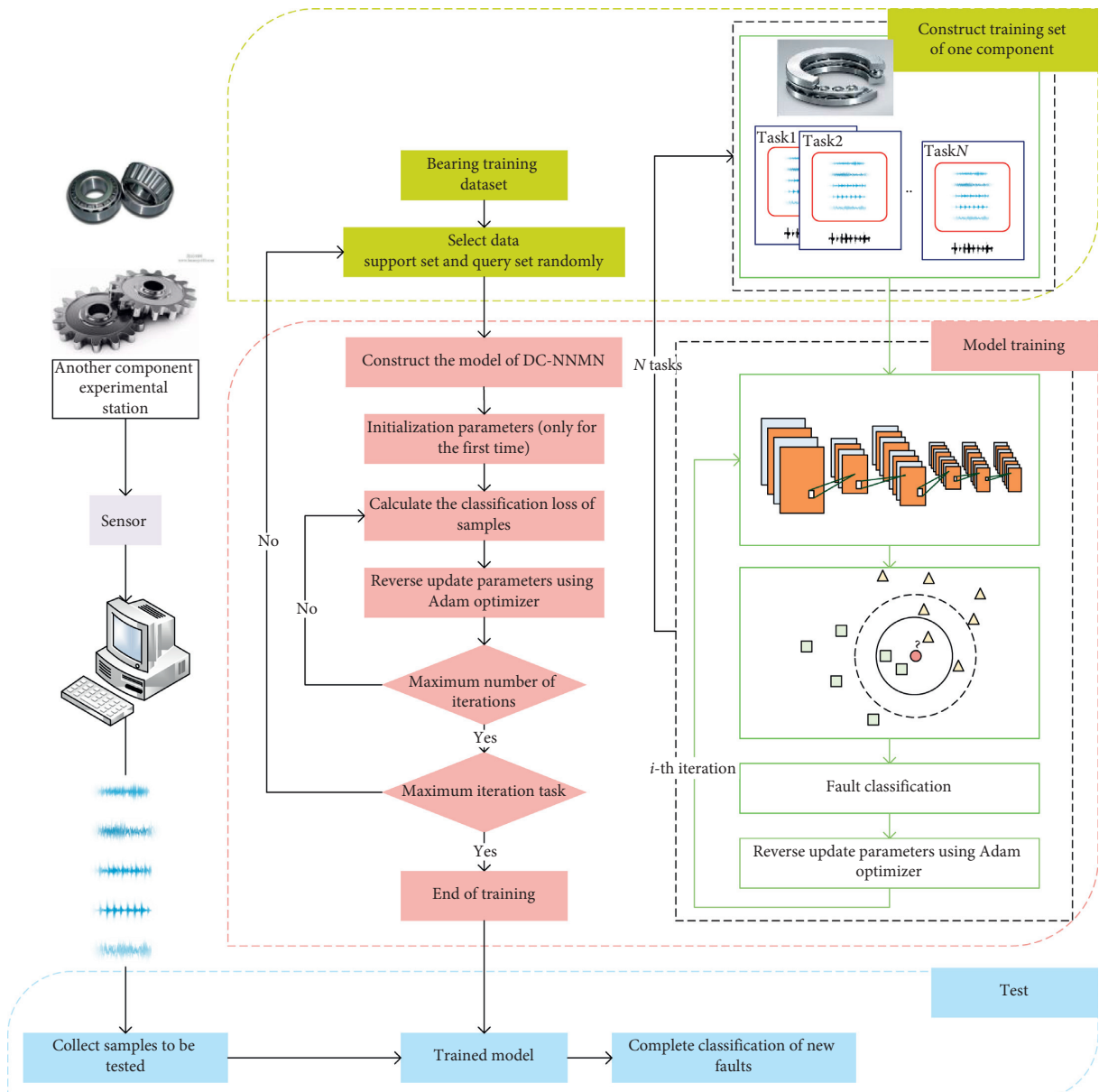


FIGURE 4: Flowchart of the proposed method.

- (3) Input the support set S and query set Q to the network, and the terminal of the network will give the classification results. It is worth mentioning that, in this stage, the parameters of the model will not be updated anymore; that is to say, through the training of multiple tasks and the parameters optimization, the model has already possessed the ability to classify completely different C -type fault samples on the C -way k -shot sample set.

4. Case Study

In this section, we use the Case Western Reserve University (CWRU) bearing datasets [20], the bearing vibration data selected from Lab-built experimental platform, and another gearing dataset [21] for our experiment to prove the feasibility of the proposed method.

4.1. Data Setting. As shown in Figure 5, the CWRU bearing experimental platform includes a 2-horsepower motor (left), a torque sensor (middle), a power meter (right), and electronic control equipment. This dataset is one of the most commonly used benchmark datasets in the field of fault diagnosis. Single point pitting faults are arranged on the bearings using EDM technology. The fault categories include IF (inner ring faults), OF (outer ring faults), and BF (rolling body faults). At the same time, the location of the faulty bearing is also different, which is located at the drive end and the fan end, respectively. It can be clearly seen in Figure 6 that the type of bearing fault, load, and fault size will cause significant differences in the collected signals.

Based on the above, we select the vibration samples under the conditions of different bearings on two positions, 2 kinds of load conditions, 5 kinds of fault categories, and 4 kinds of fault sizes. We set up the training set for the model with 80 fault categories of CWRU bearing and 90 samples of each category. The specific description is shown in Table 3, where bearing position contains FE (fan end) and DE (drive end) and fault categories contain BF (ball fault), IF (inner ring fault), N (normal), and OF@3 which means that the fault point is at 3 o'clock in the outer ring of the bearing; both OF@6 and OF@9 are the same.

During the test step, two different vibration datasets from different mechanical components will be verified on the model. One is the bearing fault data, which is collected by our self-built bearing experimental platform, as shown in Figure 7. The other is the gearing fault data [21]. The specific data settings are shown in Table 4.

The bearing fault categories we selected are normal (N), ball fault (BF), outer ring fault (OF), inner ring fault (IF), and a compound fault consisting of ball fault and outer ring fault (B and OF). The gear fault categories we selected are crack, health, missing, spall, and chip5a, where 5a means wear degree.

4.2. Experimental Results and Analysis

4.2.1. Part 1 Fault Classification Experiment Results on the C-Way K-Shot Problem. All the experiments in this section revolve around the classification task of the C -way K -shot problem. During the training phase, we extracted 5 different categories of fault data for each task, each fault category contains 1, 3, or 5 samples. In each task, each category of fault sample provides 15 verification data for query set. In other words, for each 5-way 1-shot task, it contains 5 support samples and 75 query samples.

In the test phase, both the bearing dataset and the gearing dataset are verified. The experimental results are the average of multiple experimental results, as shown in Figure 8. When the model is tested on the Lab-built bearing dataset, the fault classification accuracy of 5-way 1-shot, 5-way 3-shot, and 5-way 5-shot is, respectively, 82.63%, 92.60%, and 94.79%. That is to say, we only need tiny labeled data for each category for the model training; the across components fault diagnostic model has a satisfactory generalization performance when the testing set has the same category space and different probability distribution.

Moreover, we also can see that when the model is tested on the gearing dataset, the fault classification accuracy of 5-way 1-shot, 5-way 3-shot, and 5-way 5-shot is, respectively, 82.19%, 91.28%, and 93.00%. Considering the testing set of the three across components fault diagnostic experiments are with the different category space and different probability distribution, although the classification accuracy is lower than that of Lab-built bearing dataset, the results is also reasonable and favorable.

4.2.2. Part 2 Fault Classification Results of Different Models.

In this section, we compare the performance of the proposed method in this paper with several most commonly used models in the bearing fault diagnosis with fewer known labels samples, reflecting the superiority of the proposed method. The compared models include WDCNN, CNN_SVM, SAE, and SS-GAN. We give 5, 50, or 100 labeled fault samples for training the model and testing on query set. All of the samples are selected from Lab-built experimental platform and then we obtain the fault classification results through multiple experiments, as shown in Table 5.

It can be seen that the proposed method has the highest accuracy on three training sets. In the case of only five knowns fault samples, the best performance of the traditional model is the SAE model, but its fault classification accuracy is only 58.07%, while the fault classification accuracy of the proposed method on the gearing dataset is 82.19% and the accuracy on the bearing data is 82.63%.

As we know, for most neural networks, it is necessary to train with a large amount of labeled data to have a good classification accuracy. Therefore, when there is only a small amount of labeled data, it is inappropriate to directly use the traditional model. The proposed method in this paper

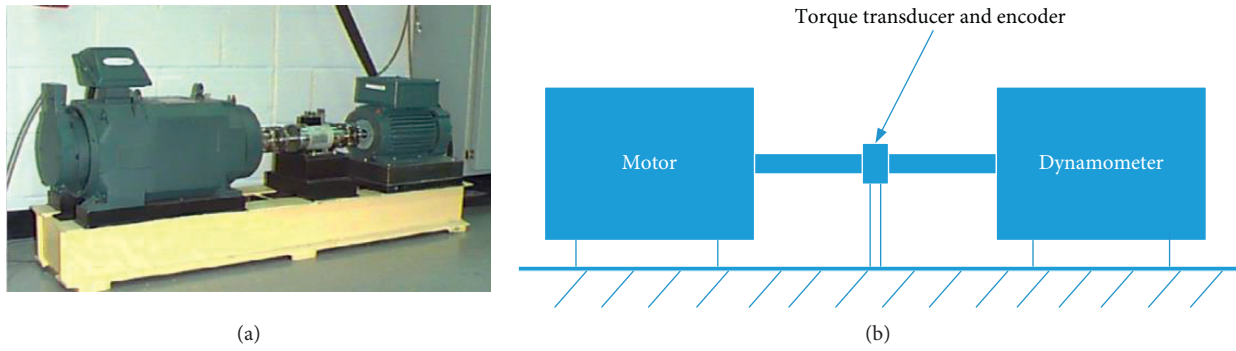


FIGURE 5: CWRU experimental platform: (a) the physical and (b) the schematic diagram.

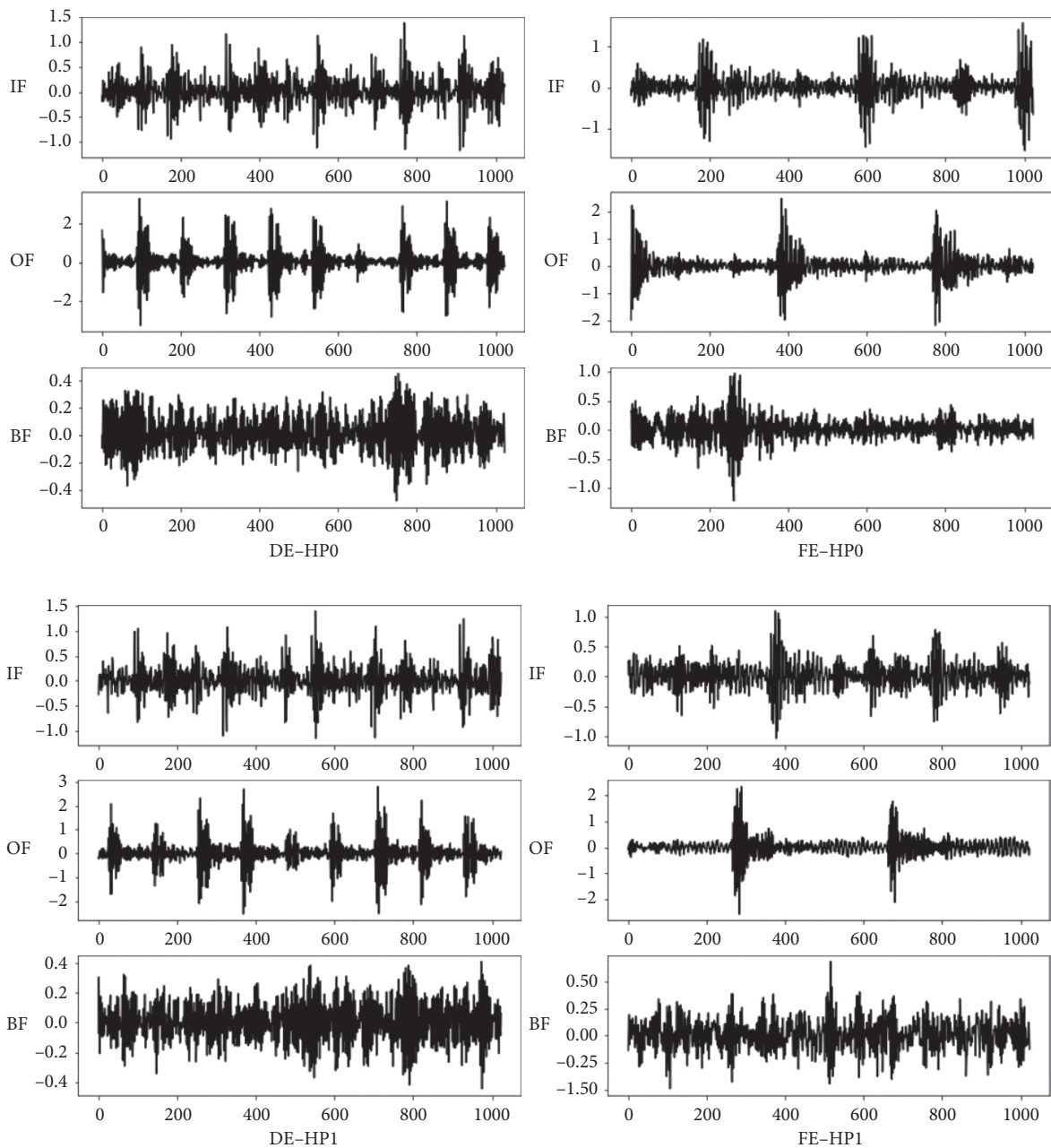


FIGURE 6: Time-domain vibration samples for three categories of faults under four working conditions.

TABLE 3: Description of the training datasets.

Frequency (kHz)	Position	Fault categories (total numbers)	Fault size (mils)	Load (hp)	Number
12	FE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	0	90
12	FE	BF, IF, OF@3, and OF@6 (4)	0.014	0	90
12	FE	BF, IF, and OF@6 (3)	0.021	0	90
12	FE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	3	90
12	FE	BF, IF, and OF@3 (3)	0.014	3	90
12	FE	BF, IF, and OF@3 (3)	0.021	3	90
12	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	0	90
12	DE	BF, IF, and OF@6 (3)	0.014	0	90
12	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.021	0	90
12	DE	BF and IF (2)	0.028	0	90
12	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	3	90
12	DE	BF, IF, and OF@6 (3)	0.014	3	90
12	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.021	3	90
12	DE	BF and IF (2)	0.028	3	90
48	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	0	90
48	DE	BF, IF, and OF@6 (3)	0.014	0	90
48	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.021	0	90
48	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.007	3	90
48	DE	BF, IF, and OF@6 (3)	0.014	3	90
48	DE	BF, IF, OF@3, OF@6, and OF@12 (5)	0.021	3	90
12	DE	N (1)	—	0	90

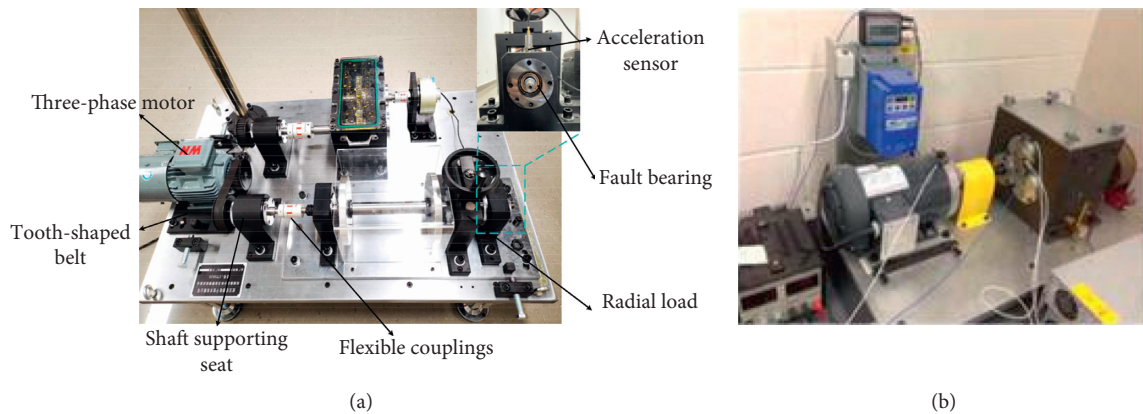


FIGURE 7: (a) Lab-built experimental platform and (b) gearbox experimental platform.

TABLE 4: Description of the testing datasets.

Component	Fault categories	Length	Number per type
Bearing	N, BF, OF, IF, B, and OF	864	90
Gear	Chip5a, crack, health, missing, and spall	864	90

performs well on tiny labeled samples. As the number of labeled samples increases, the fault classification accuracy is also improved. When using 100 labeled samples, that is, 5-way 20-shot, the fault classification accuracy of gearing can reach to 99.62%.

4.2.3. Part 3 The Effect of Different K on Experimental Results. Because the matching module of the model proposed in this paper does not need to adjust parameters, for network training, only the nearest neighbor number K can affect the

classification accuracy. Therefore, in this section, the impact of K on the classification results is discussed. Different $K = 1, 3, 5, 10$ are selected for comparison respectively. The classification results obtained through experiments are shown in Table 6.

From Table 6, we can see that, for the nearest neighbor algorithm, it is not that the larger the value of K is, the better the classification accuracy is. Relative to different datasets, the optimal value of K varies. In this paper, when $K = 5$ on the gear dataset, the model achieves the best result of 93.63%; when $K = 10$ on the bearing dataset, the model obtains the

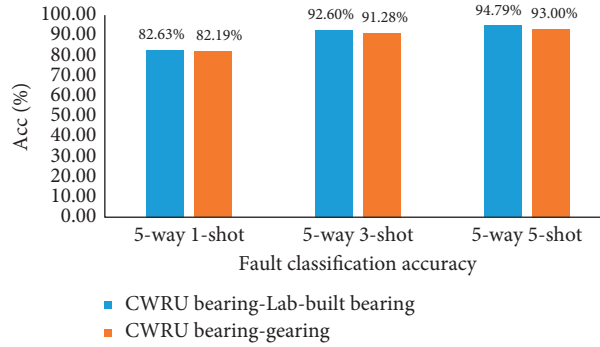


FIGURE 8: The fault classification accuracy using the trained model under the new fault category.

TABLE 5: Fault classification results of different models.

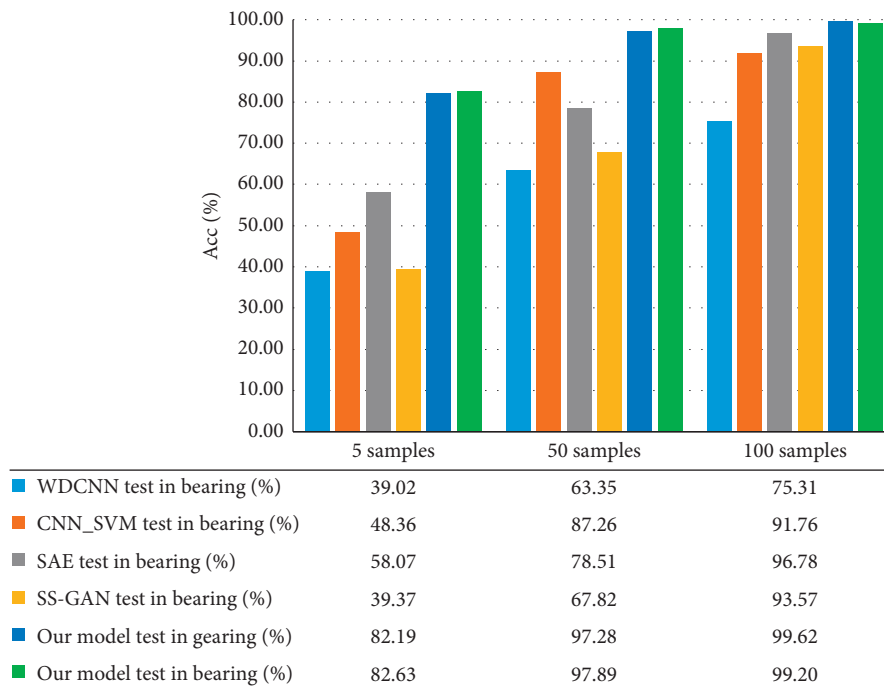


TABLE 6: Results of experiment with different K .

Model	5-way 5-shot			
	$K=1$	$K=3$	$K=5$	$K=10$
Accuracy (gear)	90.82	93.00	94.48	93.63
Accuracy (bear)	92.27	94.79	95.00	95.51

best result of 95.51%. This is because the gearing data has higher discreteness, the data distribution is relatively more dispersed, and a higher K value will reduce the accuracy of the classification results. While the distribution of the bearing data is more compact, it can promote classification accuracy with the increasing K .

5. Conclusion

In this study, a deep convolutional neighbor matching network based on few-shot learning is proposed, which

can solve the across components fault diagnosis with tiny labeled samples. The convolutional network is used to extract fault features from a small sample dataset. Then the K -Nearest Neighbor algorithm is adopted to match the samples of the unknown label with the dataset to achieve the fault classification of the new categories. We have proved the superiority of the proposed method by using three datasets of different components and comparing with four popular network models. The method in this paper provides a good idea for solving the problem of across components fault diagnosis with tiny labeled samples.

Data Availability

The experimental data of this article are from the Case Western Bearing Data Center; the bearing vibration data selected from Lab-built experimental platform and another gearing dataset are specified in the article.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

Pengfei Xu contributed to the writing of this manuscript, references, analysis of experimental results, and data interpretation. Juan Xu contributed to the research ideas, research directions, research acquisition, and research design. Lei Shi contributed to document retrieval, research content, algorithm flow, data analysis, algorithm analysis, and manuscript writing. Zhenchun Wei contributed to the experimental design, algorithm implementation, experimental analysis, and manuscript review of this manuscript. Xu Ding contributed to the experimental algorithm.

Acknowledgments

This research was funded by the National Key Research and Development Plan of China (no. 2018YFB2000505), Nation Nature Science Foundation of China (no. 61806067), and the Major Science and Technology Project of Anhui Province (no. 17030901047).

References

- [1] S. Gahlot, S. R. N. Reddy, and D. Kumar, "Review of smart health monitoring approaches with survey analysis and proposed framework," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2116–2127, 2019.
- [2] W. Huang, J. Cheng, and Y. Yang, "Rolling bearing performance degradation assessment based on convolutional sparse combination learning," *IEEE Access*, vol. 7, pp. 17834–17846, 2019.
- [3] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: fault diagnosis with model-based and signal-based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3757–3767, 2015.
- [4] Y. Lei, J. Lin, M. J. Zuo, and Z. He, "Condition monitoring and fault diagnosis of planetary gearboxes: a review," *Measurement*, vol. 48, pp. 292–305, 2014.
- [5] R. Liu, B. Yang, E. Zio, and X. Chen, "Artificial intelligence for fault diagnosis of rotating machinery: a review," *Mechanical Systems and Signal Processing*, vol. 108, pp. 33–47, 2018.
- [6] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.
- [7] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Information Fusion*, vol. 42, pp. 146–157, 2018.
- [8] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang, "Deep model based domain adaptation for fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2296–2305, 2017.
- [9] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 136–144, 2019.
- [10] Q. Hang, J. Yang, and L. Xing, "Diagnosis of rolling bearing based on classification for high dimensional unbalanced data," *IEEE Access*, vol. 7, pp. 79159–79172, 2019.
- [11] D. Cabrera, F. Sancho, J. Long et al., "Generative adversarial networks selection approach for extremely imbalanced fault diagnosis of reciprocating machinery," *IEEE Access*, vol. 7, pp. 70643–70653, 2019.
- [12] D. Zhao, F. Liu, and H. Meng, "Bearing fault diagnosis based on the switchable normalization SSGAN with 1-D representation of vibration signals as input," *Sensors*, vol. 19, no. 9, p. 2000, 2019.
- [13] A. Santoro, S. Bartunov, and M. Botvinick, "One-shot learning with memory-augmented neural networks," *Advances in Neural Information Processing Systems*, <https://arxiv.org/abs/1605.06065>, 2016.
- [14] T. Munkhdalai and Y. Hong, "Meta networks," in *Proceedings of the 34th International Conference on Machine Learning*, pp. 2554–2563, Sydney, Australia, August 2017.
- [15] K. Gregory, Z. Richard, and S. Ruslan, "Siamese neural networks for one-shot image recognition," in *Proceedings of the 32nd International Conference on Machine Learning*, Lille, France, July 2015.
- [16] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," *Advances in Neural Information Processing Systems*, <https://arxiv.org/abs/1606.04080>, 2017.
- [17] S. Jake, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in Neural Information Processing Systems*, <https://arxiv.org/abs/1703.05175>, 2017.
- [18] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, August 2017.
- [19] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.
- [20] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 5, pp. 3137–3147, 2016.
- [21] P. Cao, S. Zhang, and J. Tang, "Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning," *IEEE Access*, vol. 6, pp. 26241–26253, 2017.