

## Research Article

# Deep Transfer Learning Method Based on 1D-CNN for Bearing Fault Diagnosis

Jun He <sup>1</sup>, Xiang Li <sup>1</sup>, Yong Chen,<sup>1</sup> Danfeng Chen,<sup>1</sup> Jing Guo,<sup>1</sup> and Yan Zhou<sup>2</sup>

<sup>1</sup>College of Automation Engineering, Foshan University, Foshan City 528000, Guangdong Province, China

<sup>2</sup>College of Computer Science, Foshan University, Foshan City 528000, Guangdong Province, China

Correspondence should be addressed to Xiang Li; 1106546762@qq.com

Received 21 October 2020; Revised 21 April 2021; Accepted 10 May 2021; Published 19 May 2021

Academic Editor: Zengshun Chen

Copyright © 2021 Jun He 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 mechanical fault diagnosis, it is impossible to collect massive labeled samples with the same distribution in real industry. Transfer learning, a promising method, is usually used to address the critical problem. However, as the number of samples increases, the interdomain distribution discrepancy measurement of the existing method has a higher computational complexity, which may make the generalization ability of the method worse. To solve the problem, we propose a deep transfer learning method based on 1D-CNN for rolling bearing fault diagnosis. First, 1-dimension convolutional neural network (1D-CNN), as the basic framework, is used to extract features from vibration signal. The CORrelation ALignment (CORAL) is employed to minimize marginal distribution discrepancy between the source domain and target domain. Then, the cross-entropy loss function and Adam optimizer are used to minimize the classification errors and the second-order statistics of feature distance between the source domain and target domain, respectively. Finally, based on the bearing datasets of Case Western Reserve University and Jiangnan University, seven transfer fault diagnosis comparison experiments are carried out. The results show that our method has better performance.

## 1. Introduction

As an essential component of mechanical system, bearing was widely used in rotating machinery. Once the bearing fails, it will cause the mechanical system failure resulting in rotating machinery shutdown and even causing casualties. Therefore, bearing fault diagnosis is of great significance to the health and safe operation of machinery and has attracted more and more attention in scholars and manufacturing industries [1–3].

In the last decade, with the increasingly complex structure of mechanical equipment structure and rapid development of sensor technology, the acquisition of vibration signal has become easy and has brought new perspectives and challenges to the traditional intelligent fault diagnosis of rotating machinery [4, 5]. Zhang et al. presented a fault diagnosis and location method based on artificial neural networks (ANNs) [6], and support vector machine (SVM) method can obtain a higher accuracy rate in

classification for machine diagnosis used in [7]. Random forest (RF) classifier was used for roll bearing fault diagnosis published in [8], and Li et al. [9] used the variational mode decomposition (VMD) and kernel extreme machine learning (EML) for bearing fault diagnosis. Shi et al. proposed an intelligent fault diagnosis method based on deep learning and particle swarm optimization support vectors machine [10]. He et al. reported an intelligent bearing fault diagnosis method based on sparse autoencoder [11]. In [12], authors proposed a CNN model based on dislocation time series for fault diagnosis. ANN was used to model and identify fault signals [13]. Although these traditional intelligent fault diagnosis methods mentioned above can achieve good results, they are all based on the following two assumptions: (1) a large number of labeled fault information samples are available and (2) the training and testing samples are shared with the same probability marginal distribution. However, in actual engineering, it is a luxury to collect massive labeled fault information samples, and the

collected data from unknown operation condition are not drawn from the same probability marginal distribution [14].

In order to satisfy the actual needs of bearing fault diagnosis in practical application engineering, transfer learning, a novel classification method by using the learned knowledge from sources domain to unknown target domain, has attracted more and more attentions in fault diagnosis [15, 16]. Li et al. used the multilayer domain adaptation method to estimate the discrepancy of source domain and target domain for fault diagnosis [17]. A transfer learning method based on hierarchical deep domain adaptation was proposed for fault diagnosis [18]. An instance transfer learning method based on the long-term memory recurrent neural network model was proposed to solve the problem of difficulty in obtaining a large number of labeled fault data [19]. In order to overcome the problem of sparse feature space and few unlabeled fault data problem of some modes, in [20], Hao et al. proposed a multimodel transfer learning method for chemical process fault diagnosis. An and Ai proposed a fault diagnosis method based on end-to-end unsupervised domain adaptive Riemann CORrelation ALignment metric [21]. Xu et al. presented a transfer component analysis (TCA) method trying to learn some transfer components across domains aiming to alleviate insufficient data conditions [22].

Based on the abovementioned literature analysis, transfer learning has made a great breakthrough in the field of insufficient training data and data collected in varying condition. However, the existing transfer learning-based methods mainly focus on how to measure the interdomain feature marginal distribution discrepancy in domain adaptation. MMD (maximum mean discrepancy), a well-known domain adaptation method for distance metric, has been widely adopted in marginal distribution optimization and has achieved better performance [23–25]. Nevertheless, the limitation of MMD in the domain adaptation is that as the number of samples increases, the computation cost will increase exponentially, resulting in poor generalization ability. Thus, it is difficult to meet the requirements of real-time and generalization for fault diagnosis methods in real-world industries.

Motivated by the analysis abovementioned, this paper proposes an intelligent bearing fault diagnosis method based on deep transfer learning with CORAL loss metric, which is used to measure interdomain marginal distribution discrepancy. First, as a basic feature representation-learning framework, CNN is used to obtain the robust feature space from vibration signals. To estimate the marginal distribution discrepancy between source domain and target domain, the nonlinear transformed CORAL domain adaption is exploited to minimize marginal distribution discrepancy and at same time to constrain the CNN parameters aiming to obtain more robust feature representation learned by CNN also. Then, two objectives need to be optimized, respectively. One optimization objective is a conditional classifier based on CNN, using the cross-entropy loss function to minimize classification error. The other is the second-order statistics of features between the optimal features of source domain and target domain, which are optimized by the Adam method.

Finally, twelve comparative experiments based on Case Western Reserve University bearing dataset and six comparative experiments based on Jiangnan University bearing dataset are carried out to verify the effectiveness of our method. The main contribution of our method lies in the following aspects:

- (1) One-dimensional (1-D) CNN is build to extract representation features from the original vibration signal, and then the domain adaptation is performed only in the latter two layers unlike other CNN in latter three layers, with the purpose of reducing computation cost.
- (2) A differentiable loss function is constructed for extending CORAL metric domain adaptation to minimize the marginal distribution discrepancy from cross-domain representation feature covariance.
- (3) Two objectives are optimized by cross-entropy loss function and Adam optimizer, to minimize the classification error for CNN and second-order statistics feature of source domain and target domain, respectively.

The remainder of this paper is organized as follows. In Section 2, the theory of TL, CNN, and CORAL is briefly introduced. In Sections 3, our method is described in detail. The comparison experiments for verifying the performance of the proposed method based on bearing dataset are studied in Section 4. Conclusions are given in Section 5.

## 2. Theoretical Background

In this section, we will mainly introduce the model structure of TL and CNN, which is usually used to transfer learned knowledge from source domain to the target and to classify fault. In addition, we will introduce the relevant theoretical knowledge of CORAL.

*2.1. Transfer Learning.* TL, an important branch of machine learning, is usually used to tackle the problem of insufficient data and marginal distribution inconsistency by learning knowledge from training data to testing data. It has been widely used in fault diagnosis [26].  $D_s$  used to define the source domain has massive distinguishing knowledge which is the main object to be migrated.  $D_t$  denotes the target domain, which obtained the knowledge learned from  $D_s$ , and transfer learning tries to apply the knowledge distinguished learned previously from  $D_s$  to  $D_t$ .  $x_i$  and  $\mathcal{X}$  represent the feature of  $i$ th sample and the feature space of samples, respectively.  $s$  and  $t$  are used to identify the source domain and the target domain, respectively. The class space of the source domain and target domain can be expressed as  $Y_s$  and  $Y_t$ ; meanwhile,  $y_s$  and  $y_t$  represent the categories of the source domain and target domain, respectively.

The classification accuracy of traditional machine learning methods will drop sharply when source domain and target domain did not have the same marginal distribution. Thus, domain adaptation was used to weaken the influence

of marginal distribution inconsistency from the two domains [27]. As shown in Figure 1, the data marginal distribution of the target domain is quite different from that of the source domain before domain adaptation. After domain adaptive learning, the data marginal distribution difference between the two domains is reduced to achieve the migration from the source domain to the target domain.

Given a labeled source domain  $D_s = \{x_i^s, y_i^s\}_{i=1}^{n_s}$  and an unlabeled target domain  $D_t = \{x_i^t\}_{i=1}^{n_t}$ , the marginal distribution of the two domains is  $P(x^s)$  and  $P(x^t)$  and satisfied  $P(x^s) \neq P(x^t)$ , but their feature spaces meet  $\mathcal{X}_s = \mathcal{X}_t$ , and their category spaces are the same, that is,  $Y_s = Y_t$ .

MMD, a widely used distance in transfer learning for interdomain distribution discrepancy measure, was explored to construct as a new regularized item in loss function to make the distribution discrepancy of the two domains as small as possible. It can obtain the nonparametric distance from interdomain feature distribution without calculating the intermediate density. To measure the marginal distribution discrepancy by migrating data in the reproducing kernel Hilbert space (RKHS), the calculation formula of MMD is defined as follows:

$$\text{MMD}^2(X_s, X_t) = \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_i^s) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(x_j^t) \right\|_H^2, \quad (1)$$

where  $\phi(\bullet)$  is the nonlinear mapping from the original feature space to RKHS and  $H$  indicates that the distance is measured in RKHS. For a detailed introduction about MMD, please refer to [28].

**2.2. Convolutional Neural Network.** CNN, one of the most representative networks in the field of deep learning, was extensively used in civil structures, mechanical structures, and wind engineering [29, 30]. It has three layers such as convolution layer, pooling layer, and full connection layer.

Convolution layer is the core layer of CNN, which contains a set of trainable filters. Weight sharing is the most important characteristic of the convolution layer. It is used to optimize the network parameters to avoid over fitting caused by too many parameters and to relax the computer load, which is expressed as follows:

$$x_j^l = f \left( \sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l \right), \quad (2)$$

where  $x_j^l$  denotes the  $l$ th feature in the  $j$  layer,  $M$  and  $k$  indicate the set of input features and the convolution kernel, respectively, and  $f(\bullet)$  and  $b$  are the nonlinear activation function and the bias term, respectively. The commonly used activation function is ReLU (rectified linear unit), which is expressed as follows:

$$\text{ReLU}(x) = \max(0, x). \quad (3)$$

Generally speaking, the pooling layer (PL) performs the down sampling operation. The main purpose of PL is to reduce the parameters of the neural network while retaining the representative features and to prevent over fitting and

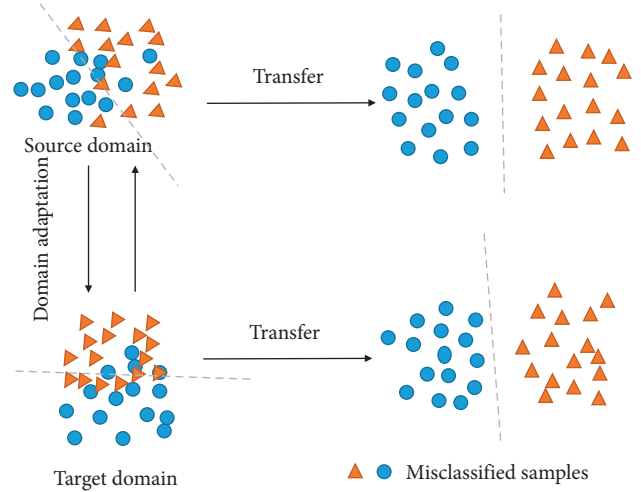


FIGURE 1: The framework of transfer learning.

improve the generalization ability of the model. The PL operation can be carried out as follows:

$$x_j^{l+1} = p(x_j^l), \quad (4)$$

where  $x_j^{l+1}$  is the  $j$ th feature of  $l+1$  layer and  $p(x_j^l)$  represents pooling operation, respectively.

The full connection layer plays the role of ‘‘Classifier’’ in the whole neural network. First, the output of the last pooling layer is expanded into a one-dimensional feature vector as the input of the fully connected layer. Then, the inputs and outputs are fully connected, and the activation function of the hidden layer is ReLU. Finally, the Softmax function is used to the output layer; the calculation of full connection layer is given as follows:

$$x^l = f(w^l x^{l-1} + b^l), \quad (5)$$

where  $w^l$  and  $b^l$  indicate the weight and bias of the full connection layer, respectively, and  $f(\bullet)$  denotes the nonlinear activation function. As the  $l$  layer was the hidden layer, the ReLU was used as activation function usually, and when the  $l+1$  layer was the output layer, the activation function was changed to Softmax and it was given by

$$h_\theta(x) = \begin{bmatrix} p(y^{(i)} = 1 | x^{(i)}; \theta) \\ p(y^{(i)} = 2 | x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k | x^{(i)}; \theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^k e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \vdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix}, \quad (6)$$

where  $p(y^{(i)} = k | x^{(i)}; \theta)$  is the probability denoted that input  $i$ th sample feature  $x^{(i)}$  belongs to category  $j$ ,  $\theta_1, \theta_2, \dots, \theta_k \in \mathcal{R}^{n+1}$  is the parameters of the model, and  $1/\sum_{j=1}^k e^{\theta_j^T x^{(i)}}$  is used to normalize the marginal probability distribution so that the sum of all the probabilities is equal to 1.

**2.3. Correlation Alignment Metric.** CORAL is an effective and simple unsupervised adaptive method which was first proposed by [31] and widely used to measure the discrepancy of source domain and target domain in model recognition such as it aligns the input feature distributions of the source and target domains by exploring their second-order statistics. Therefore, the only computations it needs are computing covariance statistics in each domain. When incorporated into a deep neural network, it can be summarized as follows:

$$\mathcal{L}_{\text{CORAL}} = \frac{1}{4d^2} \text{cov}(\mathbf{H}_s) - \text{cov}(\mathbf{H}_t)_F^2, \quad (7)$$

where  $\text{cov}(\mathbf{X}) = \mathbf{X}^\top \mathbf{C}_n \mathbf{X}$  denotes the covariance matrix and  $\mathbf{C}_n = \mathbf{I}_n - (1/n)\mathbf{1}_n \mathbf{1}_n^\top$  denotes that the centering matrix  $\mathbf{I}_n$  is a  $n$ -dimensional vector with all elements being one.

Compared with MMD, the difference is that MMD-based approaches usually apply the same transformation to both the source and target domain. [31] And asymmetric transformations are more flexible and often yield better performance for domain adaptation tasks [32]. Therefore, we use CORAL as a measure of the difference between the two domains instead of MMD to get better results.

The domain adaptation is achieved by minimizing the difference between the feature space of the source domain and the target domain, and the CORAL method is used. By taking the coral loss into the optimization objective, the similarity of the feature space learned in the source domain and the target domain is maximized so as to make up for the deficiency of CNN's insufficient learning of domain invariant feature space.

### 3. Proposed Method

This section is divided into subheadings. It should provide a concise and precise description of the experimental results, their interpretation as well as the experimental conclusions that can be drawn.

**3.1. Condition Classifier Based on CNN.** The condition classifier based on CNN consists of 10 layers of one-dimensional CNN, including one input layer, four convolution layers, two pooling layers, two fully connected layers, and one output layer. In 10 layers of CNN, the first seven layers are called feature extractors, which are used to extract the conditional representative feature from vibration signals. Meanwhile, the last layer is regarded as the condition classifiers for judging the condition of the test sample. The input layer is constructed by one-dimensional vibration signal with the length of 784. In the convolution layer, convolution kernels are used for the local region of the input signal and generate corresponding features as shown in Figure 2.

In order to reduce the dimension of convolution features and preserve representatively features as much as possible, a pooling layer is connected after the first convolution layer and last convolution layers. Through four convolution layers and two pooling layers operation, the input features will

become flat in the first fully connected layer  $F1$ . And then, in the two fully connected layers, the distribution discrepancy of interdomain is estimated by CORAL metric. Finally, the category of input sample is recognized by Softmax classifier. Feature extraction is mainly composed of one-dimensional CNN, and its structure and parameters are given in Table 1.

**3.2. Domain Adaptation Based on CORAL.** Domain adaptation is an important means to transfer knowledge from source domain to target domain when data marginal distribution is inconsistent between source domain and target domain, which determines the efficiency of knowledge transfer. The common MMD domain adaptation criteria have high computational complexity and low generalization ability with the increase in data volume. In order to effectively measure the data marginal distribution difference between the source domain and the target domain, we use a differentiable loss function to minimize the similarity comparison of the marginal distribution differences between the source domain and the target domain [31]. As shown in Figure 2, we introduce the domain adaptive learning module in FC1 and FC2 of the full connection layer to calculate the covariance distance between the features at FC1 and FC2 of the source and the target domains and define it as CORAL loss. The calculation formula is as follows:

$$\text{CORAL loss} = \frac{1}{4d^2} |C_S - C_T|_F^2, \quad (8)$$

where  $|\cdot|_F^2$  is the Frobenius norm of the matrix and  $C_S$  and  $C_T$  are the covariance matrices of the source domain and target domain. Their calculation formulas are as follows:

$$C_S = \frac{1}{n_S - 1} \left( F_S^\top F_S - \frac{1}{n_S} (\mathbf{1}^\top F_S)^\top (\mathbf{1}^\top F_S) \right), \quad (9)$$

$$C_T = \frac{1}{n_T - 1} \left( F_T^\top F_T - \frac{1}{n_T} (\mathbf{1}^\top F_T)^\top (\mathbf{1}^\top F_T) \right),$$

where  $\mathbf{1}$  is the column vector with elements of 1,  $F_S$  is the output data of the source domain passing through the full connection layer FC,  $F_T$  is the output data of the target domain passing through the full connection layer FC, and  $n_S$  and  $n_T$  are the number of samples of the source domain and the target domain, respectively. Their gradient calculation is as follows:

$$\frac{\partial \mathcal{L}_{\text{CORAL}}}{\partial F_S^{ij}} = \frac{1}{d^2 (n_S - 1)} \left( \left( F_S^\top - \frac{1}{n_S} (\mathbf{1}^\top F_S)^\top \mathbf{1}^\top \right)^\top (C_S - C_T) \right)^{ij},$$

$$\frac{\partial \mathcal{L}_{\text{CORAL}}}{\partial F_T^{ij}} = - \frac{1}{d^2 (n_T - 1)} \left( \left( F_T^\top - \frac{1}{n_T} (\mathbf{1}^\top F_T)^\top \mathbf{1}^\top \right)^\top (C_S - C_T) \right)^{ij}. \quad (10)$$

**3.3. Optimization Objective.** In this subsection, we will concern the optimization objectives of the proposed method in detail. There are two objectives need to be optimized: (1) minimize conditional classification errors on the source

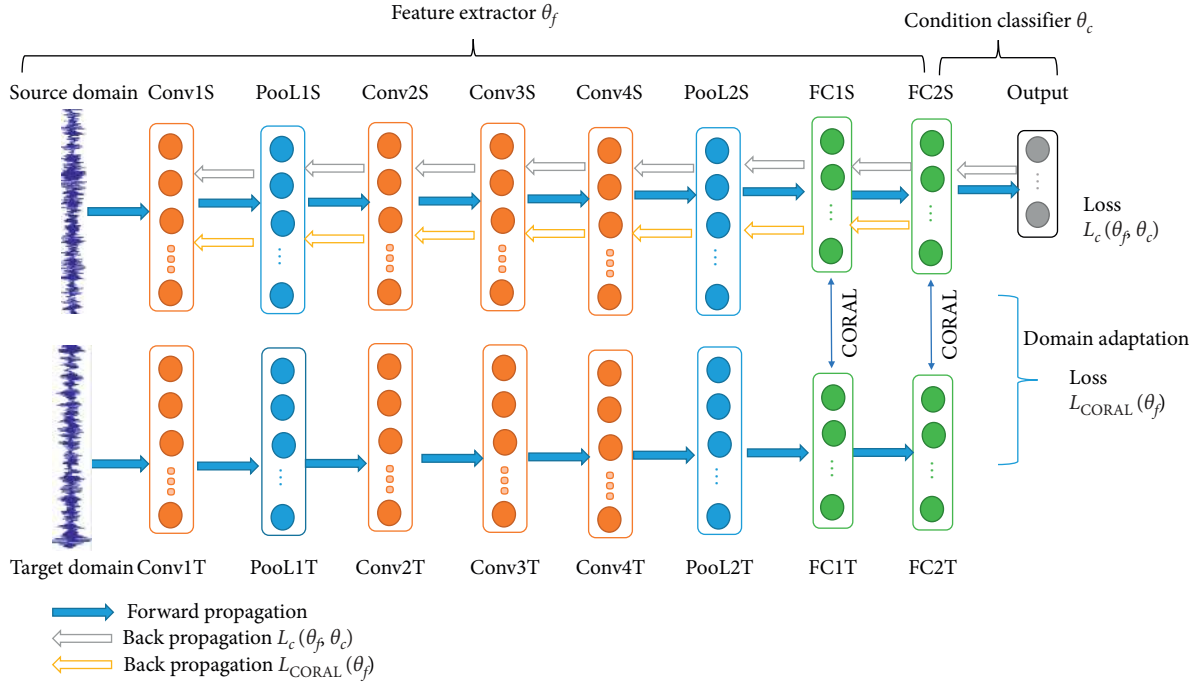


FIGURE 2: The condition classifier framework based on CNN.

TABLE 1: Structure and parameters of CNN.

Layers	Parameters	Activation function
Input	—	—
Conv1	Kernels: $1 \times 64 \times 16$ , stride: 16	ReLU
Pool1	Tride: 2, max pooling	—
Conv2	Kernels: $1 \times 3 \times 32$ , stride: 1	ReLU
Conv3	Kernels: $1 \times 5 \times 64$ , stride: 1	ReLU
Conv4	Kernels: $1 \times 5 \times 128$ , stride: 1	ReLU
Pool2	Tride:2, max pooling	—
FC1	Weights: 5000	ReLU
FC2	Weights: 1000	ReLU
Output	Weights: 10	Softmax

domain dataset given as in Figure 2 and (2) minimize the second-order statistics (covariance) of the source and target features between the source domain and the target domain. For the first optimization goal, we aim to minimize the condition classification errors of health condition category on the source domain data set by reducing the cross-entropy loss function. The specific calculation formula is expressed as follows:

$$L_c = -\frac{1}{m} \left[ \sum_{i=1}^m \sum_{j=1}^k I[y_i = j] \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^k e^{\theta_l^T x^{(i)}}} \right], \quad (11)$$

where  $m$  is the batch size of training samples,  $j$  is the fault category, and  $I[\bullet]$  is the index function. The second optimization object is the covariance distance between FC1S/FC1T and FC2S/FC2T in the fully connected layer. The covariance distance is written as follows:

$$L_{\text{CORAL}} = \lambda \sum_{l=1}^{l_2} \frac{1}{4d^2} |C_S - C_T|_F^2, \quad (12)$$

where  $l_1$  and  $l_2$  are 7-th layer and 8-th layer, respectively, which means that the network adaptation is carried out from layer 7 to layer 8, and the former layers are not employed domain adaptation. Therefore, the loss function of our method is constructed as follows:

$$L = L_c + \lambda L_{\text{CORAL}}. \quad (13)$$

Let  $\theta_f$  and  $\theta_c$  be defined as parameters of feature extractor and condition classifier, respectively. Therefore, the loss function is rewritten as follows:

$$L(\theta_f, \theta_c) = L_c(\theta_f, \theta_c) + \lambda L_{\text{CORAL}}(\theta_f). \quad (14)$$

Based on equation (14) and the stochastic gradient descent algorithm [33], the parameters  $\theta_f$  and  $\theta_c$  are updated as follows:

$$\begin{aligned} \theta_f &\leftarrow \theta_f - \eta \left( \frac{\partial L_c}{\partial \theta_f} + \lambda \frac{\partial L_{\text{CORAL}}}{\partial \theta_f} \right), \\ \theta_c &\leftarrow \theta_c - \eta \frac{\partial L_c}{\partial \theta_c}, \end{aligned} \quad (15)$$

where  $\eta$  is the learning rate; the marginal distribution discrepancy between the source domain and the target domain can be minimized by domain adaptation, and the unlabeled samples in the target domain can be classified correctly by the condition classifier.

**3.4. Overview of This Proposed Method.** In the training process of the proposed method, the Adam optimization algorithm is used for objective optimization [34], which can effectively accelerate the training process and solve the optimal problems for a large number of parameters. First, we divide the bearing data from different domains into training set (source domain) and testing set (target domain). In the feature extraction process, the forward propagation method was used to extend samples from source domain and target domain into the CNN to facilitate the feature extraction by the CNN. Then, in the domain adaptation process, the multilevel covariance distance of features between two domains is calculated to increase the similarity between the two domains as much as possible. Finally, we optimize the loss function by iterative to constrain the parameters of CNN and train the conditional classifiers in the optimization process until the end of the iterative. After the training, the obtained conditional classifier will be used to classify samples in the target domain. The specific training process is shown in Figure 3.

## 4. Experiment and Result Analysis

In order to test the performance of the proposed intelligent fault diagnosis method and verify its effectiveness, we conducted experiments using two bearing datasets. Comparative experiments are also carried out to compare the classification accuracy with existing methods including traditional CNN without transfer learning, TCA-based [22] method, handcrafted feature-based CORAL [31] method, Wasserstein distance-based deep transfer learning (WD-DTL) [35] method, DDC [36] method, and DAN [37] method.

### 4.1. Description of Dataset

**4.1.1. CWRU Bearing Dataset.** The data for Experiment 1 of this method come from the Bearing Data Center of Case Western Reserve University [38], and the original vibration signal for carrying out experiment collected from the standard platform is shown in Figure 4.

The vibration signal is collected by the bearing test platform which consists of a motor (left), a dynamometer (right), and a control circuit, and the experiment data are arranged on the bearing (SKF6205). We divide the experimental datasets into four categories. Each class has 10 groups data, including one category of general data and nine categories of fault data, namely, normal (N), inner-race fault (IF), outer-race fault (OF), and ball fault (BF). Each fault type has different degrees of fault severity (0.007 inch, 0.014 inch, and 0.021 inch fault diameters). So, there are 9 fault conditions and 1 health condition. More details are given in Table 2.

In our experiment process, we select the original vibration signal with the sampling being 12 kHz, and four different motor speeds (1797 rpm, 1772 rpm, 1750 rpm, and 1730 rpm) are applied to the bearing. We regard them as four different operating conditions (named A, B, C, and D), and each operating condition contains 1500 samples

(including 9 classes of fault, each class has 150 samples, respectively, and 150 samples with health condition). The waveforms of various categories of original vibration signal for experiment are illustrated in Figure 5. Assuming that the source domain is workload 0 and the target domain is workload 1, we treat the transfer task from 0 to 1 as  $A \rightarrow B$ .

**4.1.2. JNU Bearing Dataset.** The second bearing data come from the centrifugal fan system for rolling bearing fault diagnosis testbed of JiangNan University [39]. The bearings under test are single-row spherical roller bearings. The faults were artificially induced into bearings with a wire-cutting machine. Vibration signals of four categories of bearings include normal, outer-race defect, inner-race defect, and roller element defect. It contains three operating conditions, and the rotation speed is divided into 600, 800, and 1000 rpm, and the sampling frequency is 50 kHz. In the process of the experiment, we divided it into three different operating conditions: E, F, and G. Each operating condition contains 600 samples (including three types of faults, 150 samples, and 150 health samples, respectively). Figure 6 shows the waveforms of all kinds of original vibration signals used in the experiment.

### 4.2. Results Comparative Analysis

**4.2.1. Experimental Results of CWRU Bearing Dataset.** In order to present a comprehensive evaluation for our method, we first carry out the fault diagnosis experiment from source domain to source domain and from source domain transfer to target domain. Various condition classification accuracies of the proposed method are shown in Table 3. It can be found that our method achieves 100% classification accuracy for source domain to source domain and obtains 97.85% average test accuracy for source domain transfer to target domain. In addition, it can be seen from Table 3 that the classification accuracy of the target domain is slightly lower than that of the source domain due to the distribution inconsistency between different domains, but it is not very serious.

Furthermore, to analyze the classification accuracy of each category in more detail, the widely used confusion matrix is used to obtain the performance of compared methods. In this proposal, we choose the task  $A \rightarrow B$  to calculate the confusion matrix randomly. The detailed results are illustrated in Figure 7, with rows denoting actual health category and columns representing predicted health category.

From Figure 7, we can find that our method achieves average classification accuracy of 98.5% for each condition and the recognition differences for ten conditions are very small. However, the other comparison approaches (CNN, DDC, and DAN) have a lot of confusion across different fault conditions than the proposed, which illustrates the superiority of the proposed method.

In this proposed method, in order to conduct a detailed comparative experiment, we divide the comparison algorithm according to the feature extraction into four

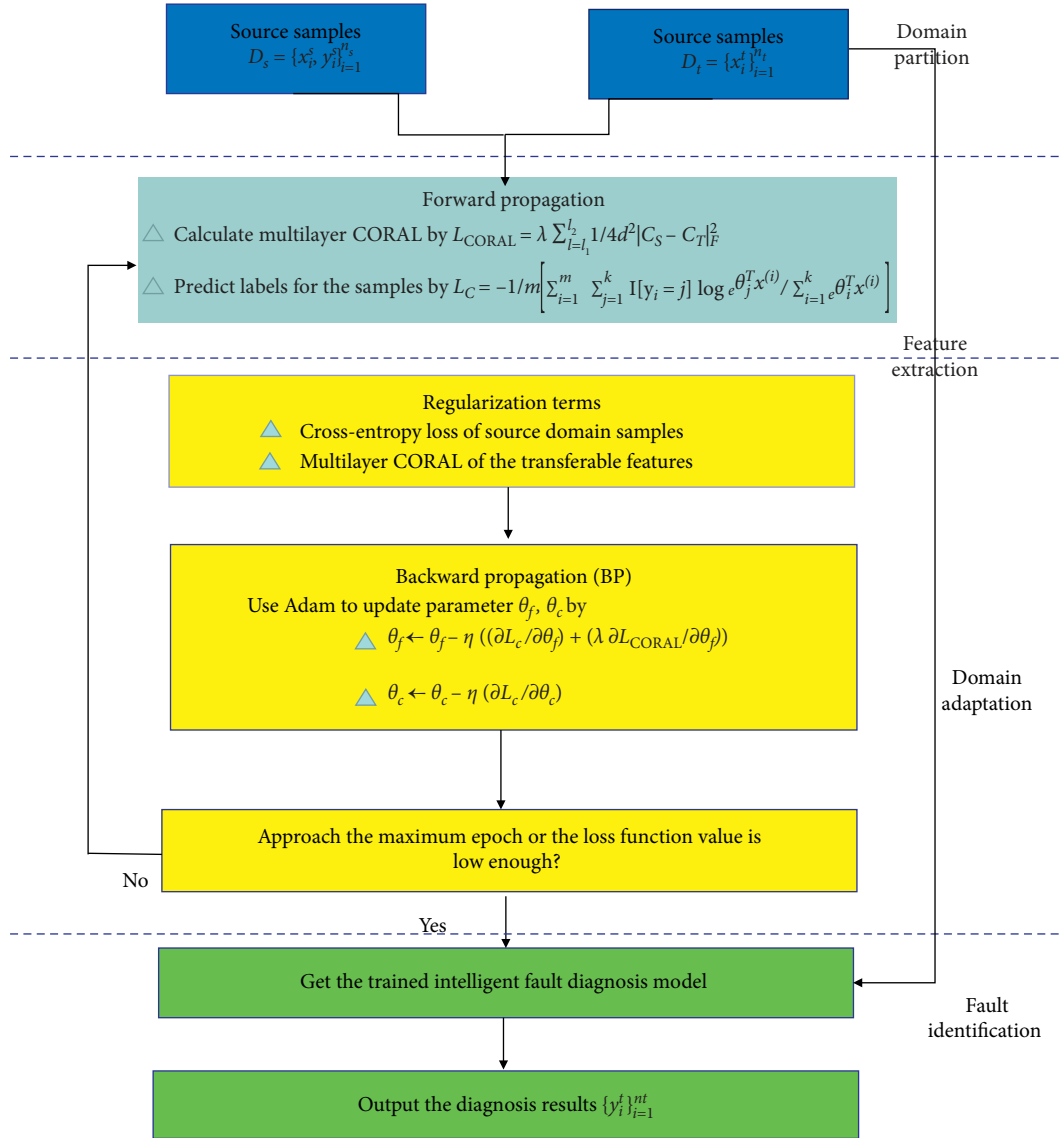


FIGURE 3: The flowchart of our method.



FIGURE 4: Bearing testbed used for experiment.

categories, when the feature extraction is the same, and then classify according to the transfer manner. The details of various methods are introduced in Table 4, and the comparative experiment by category is carried out as follows.

(1) *Comparison with CNN (without Domain Adaptation).* To be fairer, the framework of CNN used for comparison with our method is consistent with our feature extraction framework. The only difference is that the CNN method does not add a domain adaptation layer and is only trained by the source domain. It can be seen from Table 5 that the average classification accuracy of CNN in the target domain is 90.24%, which is better than that of the handcrafted feature method TCA and CORAL, respectively. However, the classification accuracy of our method is 7.61% higher than that of CNN on average. From the comparative experiments in the field of unlabeled fault diagnosis, it is necessary to perform domain adaptation and learned features to improve the classification accuracy.

(2) *Compared with TCA and CORAL.* Contrasted with the deep learning method, the main difference of traditional transfer learning is that they rely on the handcrafted features

TABLE 2: Experimental data description.

Dataset	Condition description											Samples	Operation conditions
A	Loc	N	IF	IF	IF	BF	BF	BF	OF	OF	OF	10 × 150	0HP (1797 rpm)
	Size	0	7	14	21	7	14	21	7	14	21		
B	Loc	N	IF	IF	IF	BF	BF	BF	OF	OF	OF	10 × 150	1HP (1772 rpm)
	Size	0	7	14	21	7	14	21	7	14	21		
C	Loc	N	IF	IF	IF	BF	BF	BF	OF	OF	OF	10 × 150	2HP (1750 rpm)
	Size	0	7	14	21	7	14	21	7	14	21		
D	Loc	N	IF	IF	IF	BF	BF	BF	OF	OF	OF	10 × 150	3HP (1730 rpm)
	Size	0	7	14	21	7	14	21	7	14	21		

directly used in the classifiers. In this paper, we use 21 features, including the mean, RMS (root mean square), maximum 2-order to 10-order central moment, and 9 peaks obtained by fast Fourier transform and combine with the KNN algorithm for fault diagnosis experiments.

In the TCA method, the MMD was used to calculate the distribution discrepancy of two domains, and then Gaussian kernel function was employed to minimize the MMD. The CORAL method aims to minimize the domain offset with aligning the second-order statistics (alignment of mean and covariance matrices) by linear transformation, respectively. The details of classification accuracy of various transfer tasks are shown in Table 5; the classification accuracy of TCA and CORAL is lower than that of our method by 25.06% and 30.25%, respectively. The main reason for this result is that the feature extraction manner of TCA and CORAL is handcrafted feature. We can know that it is a very helpful thing to replace the manual feature extract with deep feature learning.

(3) *Comparison with Transfer Learning Based on Wasserstein.* Wasserstein was proposed to measure the discrepancy of two probability distributions to solve the training instability of GANs (generative adversarial networks) and ensure the diversity of generated samples [40]. The advantage of Wasserstein distance over KL (Kullback–Leibler) divergence and JS (Jensen–Shannon, JS) divergence is that it can still calculate the discrepancy between the two distributions regardless of whether the two distributions are consistent. Although JS divergence is a constant in this case, KL divergence may be meaningless.

Cheng et al. proposed an intelligent fault diagnosis method based on Wasserstein distance deep transfer learning named WD-DTL [35]. Wasserstein was used to measure the distance from two domains and tried to reduce the discrepancy of two domains to achieve transfer learning fault diagnosis from different motor speed and location of sensors. From Table 4, the WD-DTL [35] method can achieve best classification accuracy in tasks B → C and D → C and 95.75% average. However, the average classification accuracy is 2.1% less than that of our method.

(4) *Comparison with Transfer Learning Based on MMD.* As a widely used discrepancy distribution measure in domain adaptation, MMD estimates the distance between two marginal distributions in Hilbert space. In this proposal, two transfer learning methods based on MMD are

employed to comparative experiments, such as DDC [36] and DAN [37]. In order to ensure the fairness of the experiment, the CNN framework that automatically extracts features in both methods is consistent with our method. In the DDC method, a single kernel function is used to reduce MMD of two domains, and the domain adaptation is only performed in one layer of CNN. However, multiple kernel functions are used to obtain MMD; meantime, the domain adaptation is executed in three layers of CNN. The classification accuracy of various mentioned algorithms in the proposed method is illustrated in Figure 8, and it can be seen that the proposed method achieves the best classification performance in 12 types of transfer tasks compared with other approaches.

4.2.2. *Experimental Results of JNU Bearing Dataset.* In order to verify the results obtained in the previous section of the experiment, we use another dataset to perform experiments on each method. The experimental results obtained are shown in Table 6, and classification accuracy of various mentioned algorithms in the proposed method is illustrated in Figure 9. As the difference between working conditions increases, the accurate progress of each method is compared with the previous experiment that has all declined, but our method is still significantly ahead of other methods. Therefore, the experimental conclusions obtained in the previous section are correct.

4.2.3. *Implementation Details.* In this subsection, we will give the detailed introduction about our experiments. The software framework used is python, and the GPU is NVIDIA GTX 1660ti. In each experiment, Adam optimizer with the learning rate of 0.001 is set, with batch size being set to 128. And penalty parameter lambda affects the performance of transfer fault diagnosis. By tuning this parameter from {0.1, 0.2, 0.5, 1, 10}, best classification accuracy is acquired. As an example, we take into account the transfer task A → B to show the training process of its loss function. Due to Adam’s fast characteristics, the loss value decreases rapidly in the first 30 iterations and tends to be stable at about 50 iterations and approaches to 0 at about 400 iterations; the detailed process of the iteration is illustrated in Figure 10.

4.3. *Feature Visualization.* In order to further demonstrate the knowledge transfer ability of our method at the feature level, t-SNE technology is also employed to visualize learned



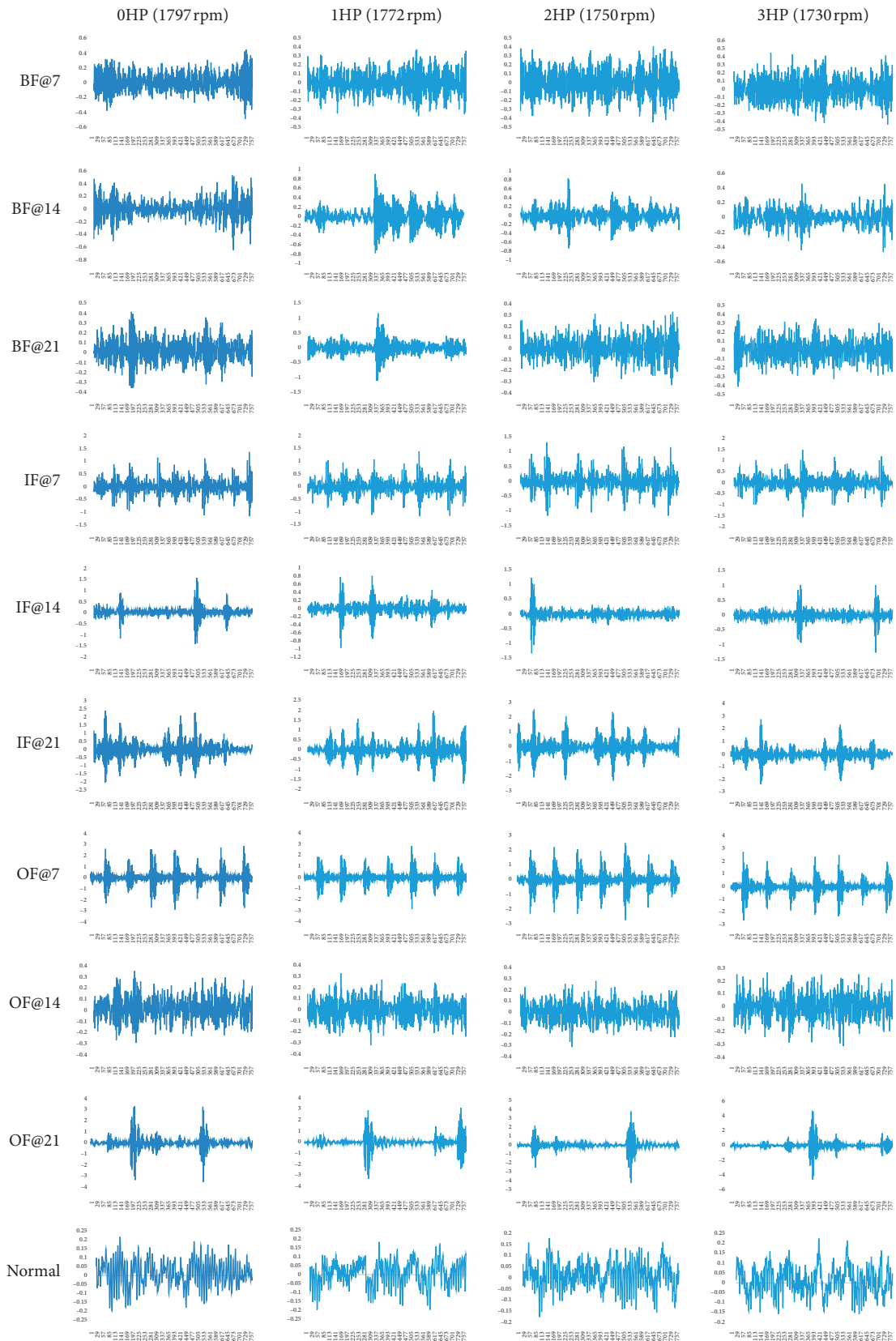


FIGURE 5: Original waveform of various working conditions.

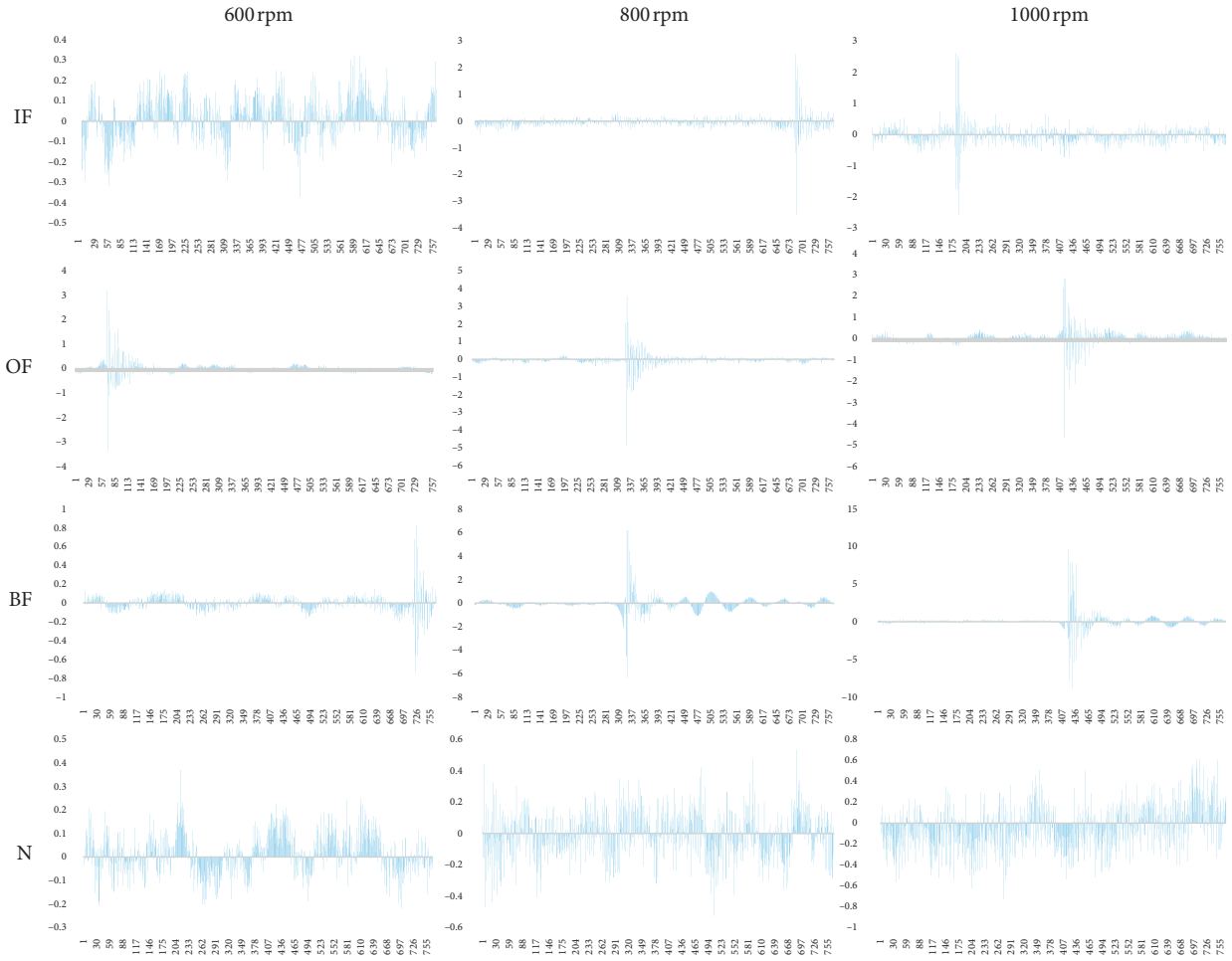


FIGURE 6: Original waveform of various working conditions.

TABLE 3: Classification accuracy (%) of various conditions.

Transfer condition	Source domain accuracy (%)	Target domain accuracy (%)
A $\rightarrow$ B	100	98.50
B $\rightarrow$ A	100	98.33
A $\rightarrow$ C	100	99.07
C $\rightarrow$ A	100	98.40
A $\rightarrow$ D	100	99.00
D $\rightarrow$ A	100	93.87
B $\rightarrow$ C	100	99.53
C $\rightarrow$ B	100	97.67
B $\rightarrow$ D	100	98.40
D $\rightarrow$ B	100	95.20
C $\rightarrow$ D	100	98.93
D $\rightarrow$ C	100	97.20
AVG	100	97.85

features for further classification accuracy analysis [41]. We randomly select the task A  $\rightarrow$  B to carry out the t-SNE feature visualization, and the obtained results are presented in Figure 10. It can be seen from Table 5 that the classification accuracy obtained by TCA and CORAL is relatively poor and only the visualization results of CNN, DCT, and DAN are realized.

Figure 11(a) shows the result of CNN without domain adaptation. Compared with the other three visualization results, the features drawn by CNN are more relaxed, which show that transfer learning can weaken the discrepancy from data marginal distribution.

Figures 11(b) and 11(c) show the visualization results obtained by DDC and DAN, respectively. The marginal

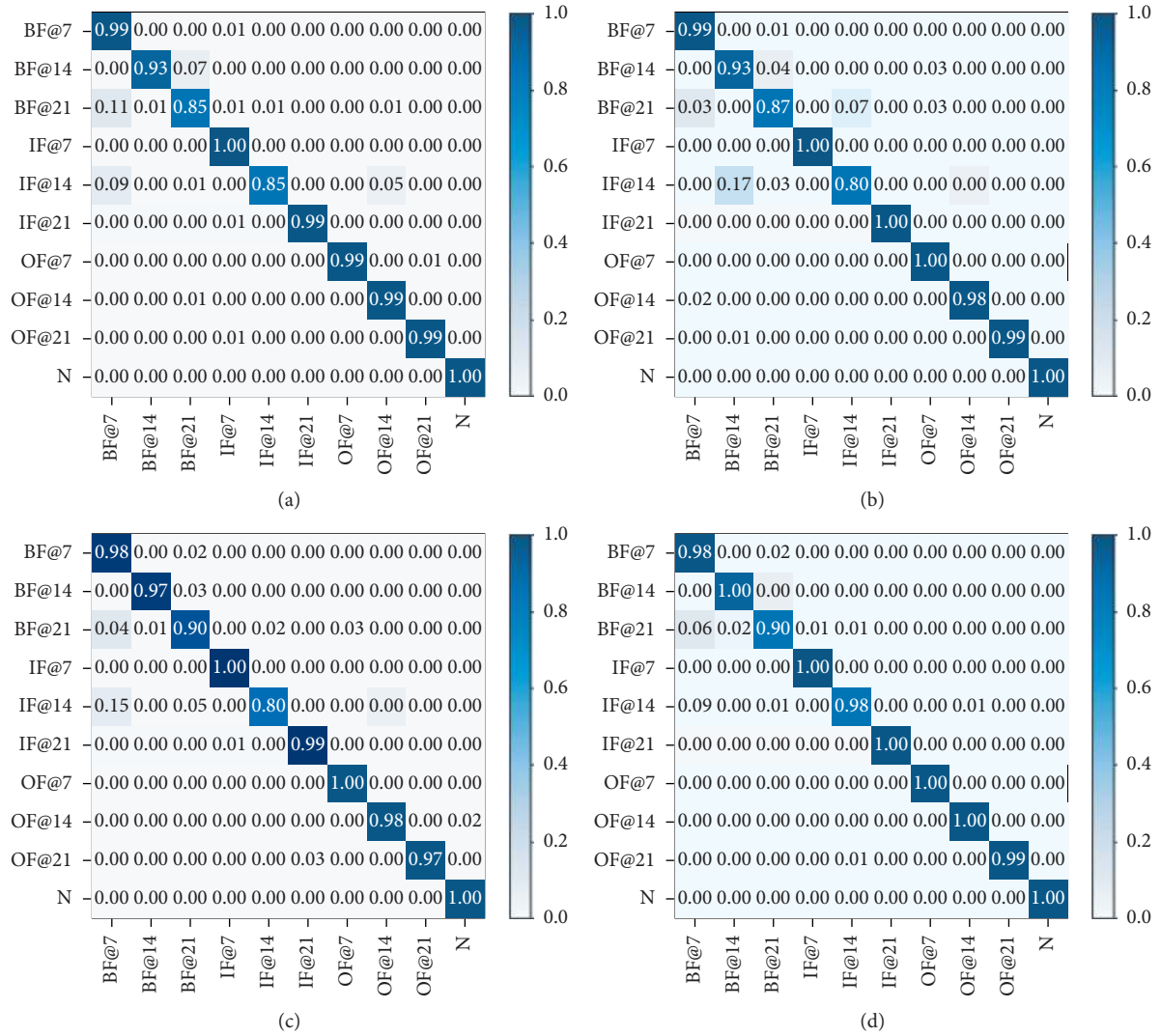


FIGURE 7: Confusion matrix of the prediction for task A → B: (a) CNN; (b) DDC; (c) DAN; (d) the proposed.

TABLE 4: Parameters and category of compared method.

Category	Method	Features	Transfer manner
1	CNN	Learned features	No transfer
2	TCA [22]	Handcrafted features	MMD
	CORAL [31]	Handcrafted features	Covariances
3	WD-DTL [35]	Learned features	Wasserstein distance
4	DDC [36]	Learned features	MMD
	DAN [37]	Learned features	MMD
5	Our method	Learned features	Covariances

distribution distances are calculated by MMD from source domain to target domain. Although comparing with CNN features, some fault features are still far apart, and a few number of features overlap together. The visualization

results of the proposed method are shown in Figure 11(d), and the feature aggregation is relatively concentrated. There are only a few outliers in the same fault condition, and there is almost no overlap in different feature domains. We can

TABLE 5: Classification accuracy of various methods.

Method	A → B	B → A	A → C	C → A	A → D	D → A	B → C	C → B	B → D	D → B	C → D	D → C	AVG
GNN	95.80 ± 0.1%	95.67 ± 0.43%	93.33 ± 1.26%	92.40 ± 2.13%	95.33 ± 0.17%	77.60 ± 1.17%	95.40 ± 0.28%	96.67 ± 0.26%	91.07 ± 0.47%	77.27 ± 5.35%	97.13 ± 0.12%	75.20 ± 4.22%	90.24
TCA [22]	79.13 ± 3.17%	77.33 ± 2.22%	72.93 ± 1.13%	63.00 ± 1.24%	66.67 ± 4.73%	74.47 ± 0.62%	82.80 ± 0.71%	78.27 ± 1.52%	75.27 ± 3.53%	74.13 ± 1.22%	61.07 ± 4.24%	68.44 ± 5.36%	72.79
CORAL [31]	75.55 ± 2.16%	72.33 ± 4.22%	69.80 ± 3.52%	66.86 ± 7.14%	52.73 ± 4.15%	54.00 ± 5.56%	72.53 ± 1.93%	70.47 ± 2.18%	80.86 ± 0.19%	76.93 ± 3.17%	72.27 ± 4.16%	67.60 ± 5.13%	67.60
WD-DTL [35]	97.52 ± 3.09%	96.80 ± 1.10%	94.43 ± 2.99%	92.16 ± 2.61%	95.05 ± 2.52%	89.82 ± 2.41%	99.69 ± 0.59%	96.03 ± 6.27%	95.51 ± 2.52%	95.16 ± 3.67%	97.56 ± 3.31%	99.62 ± 0.80%	95.75
DDC [36]	95.60 ± 1.25%	90.60 ± 2.43%	95.00 ± 2.32%	97.87 ± 0.12%	86.73 ± 2.71%	88.47 ± 2.04%	92.80 ± 1.58%	97.07 ± 0.21%	89.53 ± 4.19%	79.27 ± 0.23%	96.87 ± 0.18%	86.27 ± 2.26%	91.34
DAN [37]	95.90 ± 1.17%	93.33 ± 1.32%	96.67 ± 0.59%	96.00 ± 0.68%	88.93 ± 4.66%	88.00 ± 1.02%	94.40 ± 1.63%	98.00 ± 0.22%	90.80 ± 1.13%	78.33 ± 3.16%	97.13 ± 0.33%	81.93 ± 3.42%	91.63
The proposed	98.50 ± 0.11%	98.33 ± 0.06%	99.07 ± 0.12%	98.40 ± 0.46%	99.00 ± 0.21%	93.87 ± 1.46%	99.53 ± 0.14%	97.67 ± 0.32%	98.40 ± 0.18%	95.20 ± 2.24%	98.93 ± 0.13%	97.20 ± 1.02%	97.85

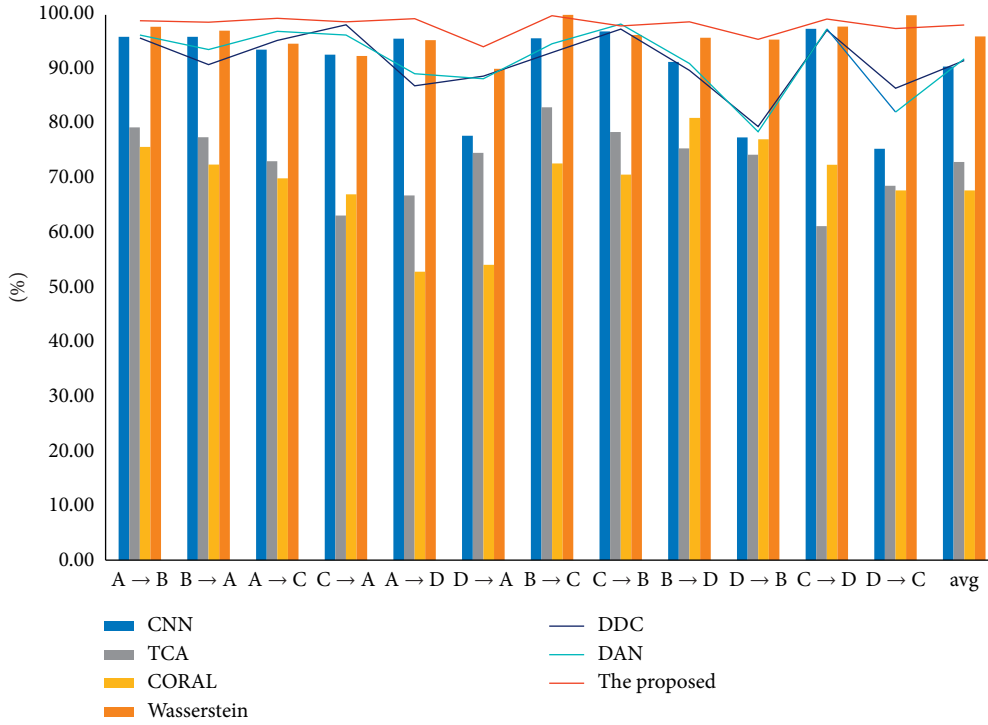


FIGURE 8: Classification accuracy of mentioned methods.

TABLE 6: Classification accuracy of various methods.

Method	E → F	F → E	E → G	G → E	F → G	G → F	AVG (%)
CNN	62.8 ± 0.1%	51.4 ± 0.43%	53.33 ± 1.26%	62.4 ± 2.13%	55.4 ± 0.28%	56.67 ± 0.26%	57.00
TCA [22]	39.13 ± 3.17%	37.33 ± 2.22%	32.93 ± 1.13%	33.25 ± 1.24%	42.80 ± 0.71%	38.27 ± 1.52%	37.29
CORAL [31]	41.55 ± 2.16%	36.33 ± 4.22%	35.80 ± 3.52%	36.86 ± 7.14%	37.53 ± 1.93%	40.47 ± 2.18%	38.09
WD-DTL [35]	77.52 ± 3.09%	74.80 ± 2.10%	64.69 ± 2.99%	72.86 ± 2.61%	75.35 ± 2.59%	68.74 ± 4.27%	72.23
DDC [36]	71.67 ± 4.25%	73.60 ± 4.43%	66.70 ± 2.32%	69.87 ± 4.12%	73.80 ± 2.58%	72.07 ± 2.21%	71.29
DAN [37]	72.90 ± 2.17%	76.33 ± 3.32%	67.67 ± 3.59%	66.40 ± 4.68%	74.40 ± 3.63%	72.30 ± 2.22%	71.67
The proposed	75.50 ± 3.11%	81.33 ± 2.06%	75.07 ± 3.52%	75.40 ± 0.46%	80.53 ± 3.14%	71.67 ± 4.32%	76.58

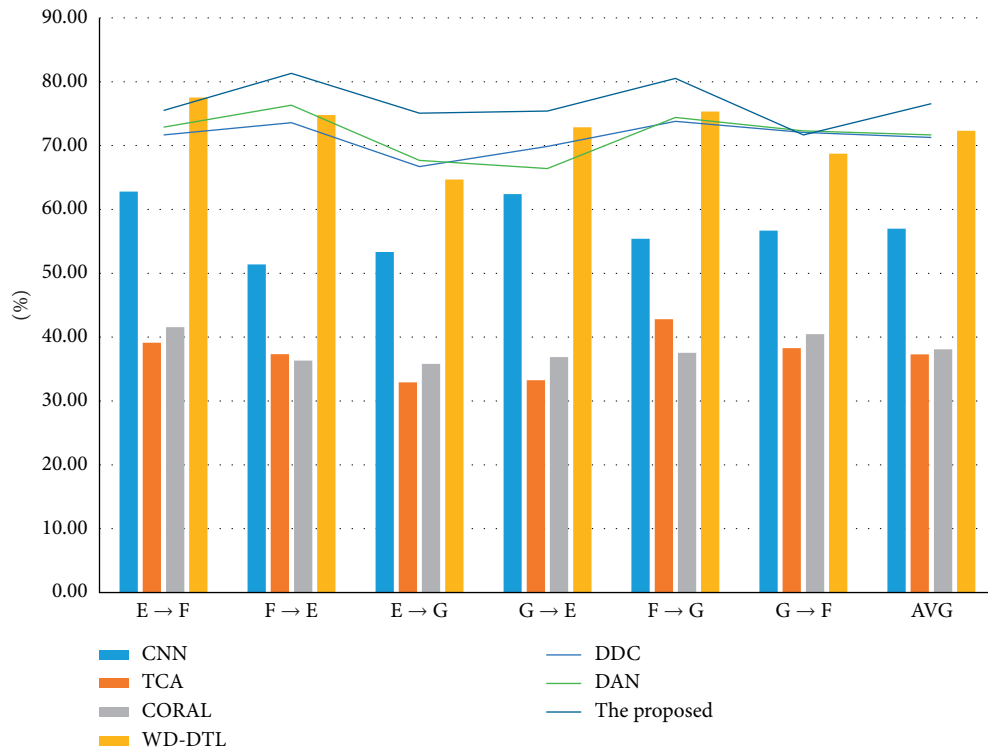


FIGURE 9: Classification accuracy of mentioned methods.

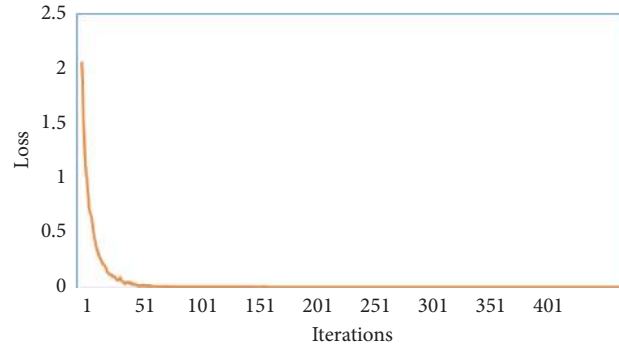


FIGURE 10: The loss of training process for the transfer task A  $\rightarrow$  B.

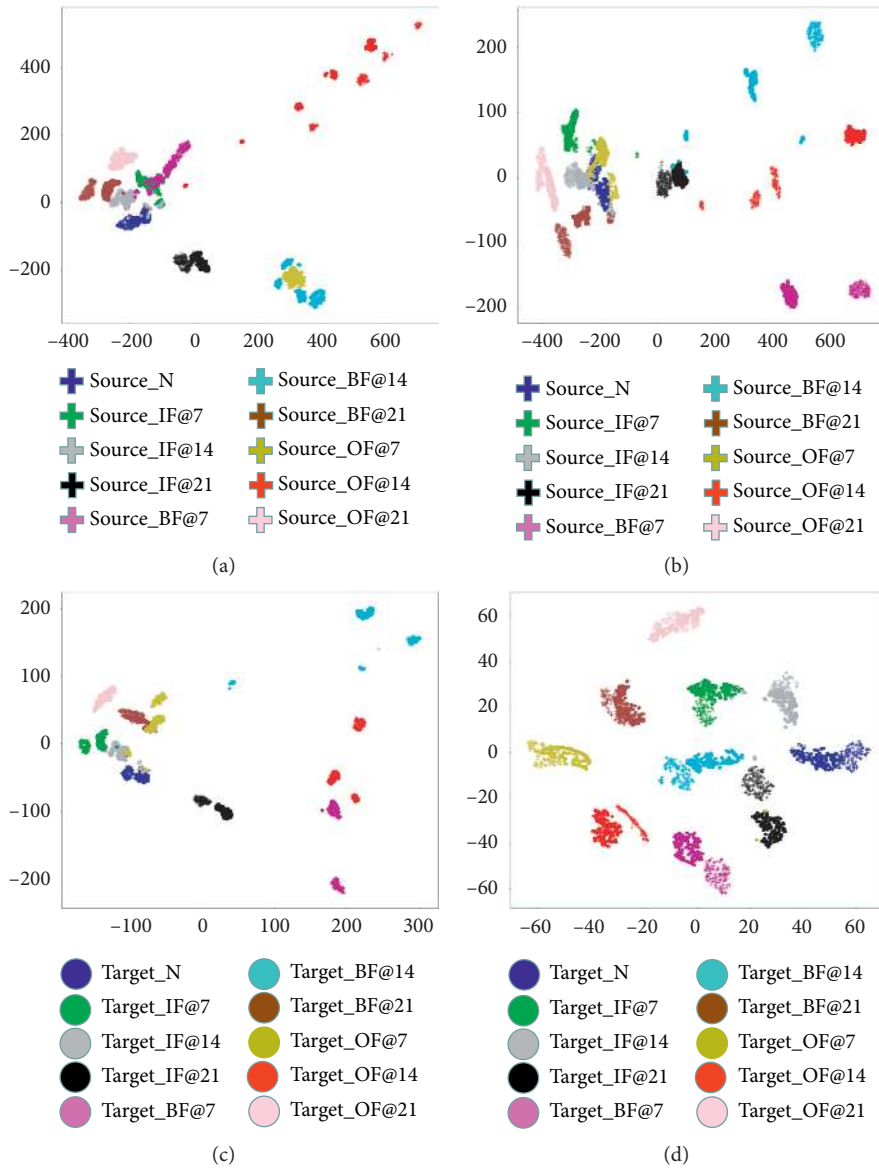


FIGURE 11: t-SNE visualization of features: (a) CNN; (b) DDC; (c) DAN; (d) the proposed.

find that the learned features in domain adaptation obtained by our method are not sensitive to domain variation and have strong fault identification ability.

## 5. Conclusion

In this paper, a deep transfer learning method based on CORAL metric for bearing fault diagnosis is proposed. The key idea of this proposed method is to employ the nonlinear transform-based CORAL loss function to estimate the discrepancy of interdomain. As a feature extractor and classifier, CNN is used to train the condition classifier model with its parameters contrasted by CORAL loss function. Eighteen types of fault of transfer tasks in two different dataset are carried out to verify classification performance of the proposed method, and five state-of-the-art architectures are used to compare with our method. These results illustrated above demonstrate that the proposed approach can achieve more satisfactory classification accuracy and domain adaptation capabilities. However, in the second experiment, the difference between the working conditions increased, and the results obtained by our method were not satisfactory. This may be due to the following two limitations of the CORAL: (1) aligning covariance with usual Euclidean metric is suboptimal and (2) second-order statistics have limited expression for the non-Gaussian distribution [42]. Therefore, in the next step, the authors will improve the proposed method and devote themselves to the unsupervised transfer learning fault diagnosis for different machines with greater differences between domains.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The research is supported by National Natural Science Foundation of China (Grant No. 61803087, 61972091), by Guangdong Natural Science Fund (Grant No. 2017A030310580, 2017A030313388).

## References

- [1] S. Osman and W. Wang, "A morphological Hilbert-Huang transform technique for bearing fault detection," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 11, pp. 2646–2656, 2016.
- [2] R. Abdelkader, A. Kaddour, A. Bendiabdellah, and Z. Derouiche, "Rolling bearing fault diagnosis based on an improved denoising method using the complete ensemble empirical mode decomposition and the optimized thresholding operation," *IEEE Sensors Journal*, vol. 18, pp. 7166–7172, 2018.
- [3] A. Moshrefzadeh, A. Fasana, and J. Antoni, "The spectral amplitude modulation: a nonlinear filtering process for diagnosis of rolling element bearings," *Mechanical Systems and Signal Processing*, vol. 132, pp. 253–276, 2019.
- [4] E. Naderi and K. Khorasani, "A data-driven approach to actuator and sensor fault detection, isolation and estimation in discrete-time linear systems," *Automatica*, vol. 85, pp. 165–178, 2017.
- [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] T. Zhang, L. X. Sun, J. C. Liu et al., "Fault diagnosis and location method for active distribution network based on artificial neural network," *Electric Power Components and Systems*, vol. 46, pp. 985–996, 2018.
- [7] A. Widodo and B.-S. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2560–2574, 2007.
- [8] Z. Wang, Q. Zhang, J. Xiong et al., "Fault diagnosis of a rolling bearing using wavelet packet denoising and random forests," *IEEE Sensors Journal*, vol. 17, pp. 5581–5588, 2017.
- [9] K. Li, L. Su, J. Wu, H. Wang, and P. Chen, "A rolling bearing fault diagnosis method based on variational mode decomposition and an improved kernel extreme learning machine," *Applied Sciences*, vol. 7, 2017.
- [10] P. Shi, K. Liang, D. Han, and Y. Zhang, "A novel intelligent fault diagnosis method of rotating machinery based on deep learning and PSO-SVM," *Journal of Vibroengineering*, vol. 19, no. 8, pp. 5932–5946, 2017.
- [11] J. He, M. Ouyang, C. Yong, D. Chen, J. Guo, and Y. Zhou, "A novel intelligent fault diagnosis method for rolling bearing based on integrated weight strategy features learning," *Sensors*, vol. 20, no. 6, p. 1774, 2020.
- [12] L. Ruonan, M. Guotao, Y. Boyuan, S. Chuang, and C. Xuefeng, "Dislocated time series convolutional neural architecture: an intelligent fault diagnosis approach for electric machine," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1310–1320, 2017.
- [13] R. S. Gunerkar, A. K. Jalan, and S. U. Belgamwar, "Fault diagnosis of rolling element bearing based on artificial neural network," *Journal of Mechanical Science and Technology*, vol. 33, no. 2, pp. 505–511, 2019.
- [14] L. Guo, Y. Lei, S. Xing, T. Yan, and N. Li, "Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabeled data," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 9, pp. 7316–7325, 2019.
- [15] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14–23, 2015.
- [16] S. M. Salaken, A. Khosravi, N. Thanh, and S. Nahavandi, "Extreme learning machine based transfer learning algorithms: a survey," *Neurocomputing*, vol. 267, pp. 516–524, 2017.
- [17] X. Li, W. Zhang, Q. Ding, and J.-Q. Sun, "Multi-Layer domain adaptation method for rolling bearing fault diagnosis," *Signal Processing*, vol. 157, pp. 180–197, 2019.
- [18] X. Wang, H. He, and L. Li, "A hierarchical deep domain adaptation approach for fault diagnosis of power plant thermal system," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5139–5148, 2019.
- [19] Z. Wu, H. Jiang, K. Zhao, and X. Li, "An adaptive deep transfer learning method for bearing fault diagnosis," *Measurement*, vol. 151, 2020.

- [20] H. Wu and J. Zhao, "Fault detection and diagnosis based on transfer learning for multimode chemical processes," *Computers & Chemical Engineering*, vol. 135, Article ID 106731, 2020.
- [21] J. An and P. Ai, "Deep domain adaptation model for bearing fault diagnosis with Riemann metric correlation alignment," *Mathematical Problems in Engineering*, vol. 2020, Article ID 4302184, 12 pages, 2020.
- [22] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.
- [23] X. Jia, M. Zhao, Y. Di, Q. Yang, and J. Lee, "Assessment of data suitability for machine prognosis using maximum mean discrepancy," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5872–5881, 2018.
- [24] W. W. Lin, M. W. Mak, and J. T. Chien, "Multisource I-vectors domain adaptation using maximum mean discrepancy based autoencoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 12, pp. 2412–2422, 2018.
- [25] Y. Pan, R. Hong, J. Chen, J. Feng, and W. Wu, "Performance degradation assessment of wind turbine gearbox based on maximum mean discrepancy and multi-sensor transfer learning," *Structural Health Monitoring-An International Journal*, vol. 20, no. 1, pp. 118–138.
- [26] Q. Yang and S. Jialin Pan, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, 2010.
- [27] M. Wang and W. Deng, "Deep visual domain adaptation: a survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.
- [28] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schoelkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, pp. 723–773, 2012.
- [29] Z. Chen, X. Fu, Y. Xu, C. Y. Li, B. Kim, and K. T. Tse, "A perspective on the aerodynamics and aeroelasticity of tapering: partial reattachment," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 212, Article ID 104590, 2021.
- [30] Z. Chen, K. T. Tse, K. C. S. Kwok, A. Kareem, and B. Kim, "Measurement of unsteady aerodynamic force on a galloping prism in a turbulent flow: a hybrid aeroelastic-pressure balance," *Journal of Fluids and Structures*, vol. 102, Article ID 103232, 2021.
- [31] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," 2015, <https://arxiv.org/abs/1511.05547>.
- [32] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proceedings of the International Conference on Computer Vision IEEE*, Columbus, OH, USA, June 2014.
- [33] A. Bordes, L. Bottou, and P. Gallinari, "SGD-QN: careful quasi-Newton stochastic gradient descent," *Journal of Machine Learning Research*, vol. 10, pp. 1737–1754, 2009.
- [34] S. B. Tahir, A. Jalal, and K. Kim, "Wearable inertial sensors for daily activity analysis based on Adam optimization and the maximum entropy markov model," *Entropy*, vol. 22, 2020.
- [35] C. Cheng, B. Zhou, G. Ma, D. Wu, and Y. Yuan, "Wasserstein distance based deep adversarial transfer learning for intelligent fault diagnosis," *Neurocomputing*, vol. 409, 2020.
- [36] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: maximizing for domain invariance," 2014, <https://arxiv.org/abs/1412.3474>.
- [37] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 3071–3085, 2019.
- [38] Case Western Reserve University Bearing Data Center, Digital Object Identifier, 2020, <http://csegroups.case.edu/bearingdatacenter/pages/welcomecase-western-reserve-university-bearing-data-center-website>.
- [39] Bearing Dataset of Jiangnan University, 2020, <http://madnet.org:8765/explore.html?id=9>.
- [40] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," 2017, <https://arxiv.org/abs/1701.07875>.
- [41] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [42] Z. Cheng, C. Chen, Z. Chen, K. Fang, and J. Xinju, "Robust and high-order correlation alignment for unsupervised domain adaptation," *Neural Computing and Applications*, no. 1, 2021.