

## Research Article

# Deep Domain Adaptation Model for Bearing Fault Diagnosis with Domain Alignment and Discriminative Feature Learning

Jing An <sup>1,2</sup>, Ping Ai <sup>2</sup>, and Dakun Liu<sup>1</sup>

<sup>1</sup>School of Mechanical Engineer, Yancheng Institute of Technology, Yancheng 224051, China

<sup>2</sup>College of Computer and Information Engineering, Hohai University, Nanjing 211100, China

Correspondence should be addressed to Jing An; [anjing991982@163.com](mailto:anjing991982@163.com)

Received 18 November 2019; Accepted 17 February 2020; Published 20 March 2020

Academic Editor: Felix Albu

Copyright © 2020 Jing An 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.

Deep learning techniques have been widely used to achieve promising results for fault diagnosis. In many real-world fault diagnosis applications, labeled training data (source domain) and unlabeled test data (target domain) have different distributions due to the frequent changes of working conditions, leading to performance degradation. This study proposes an end-to-end unsupervised domain adaptation bearing fault diagnosis model that combines domain alignment and discriminative feature learning on the basis of a 1D convolutional neural network. Joint training with classification loss, center-based discriminative loss, and correlation alignment loss between the two domains can adapt learned representations in the source domain for application to the target domain. Such joint training can also guarantee domain-invariant features with good intraclass compactness and interclass separability. Meanwhile, the extracted features can efficiently improve the cross-domain testing performance. Experimental results on the Case Western Reserve University bearing datasets confirm the superiority of the proposed method over many existing methods.

## 1. Introduction

Traditional machine learning techniques, especially deep learning, have recently made great achievements in the data-driven fault diagnosis field [1–6]. Most machine learning methods assume that the training data (source domain) and test data (target domain) must be in the same working condition and have the same distribution and feature space. However, in many real-world working conditions, the distribution of source domain samples is different from that of target domain samples, resulting in performance degradation.

To address this challenge, the main research on domain adaptation techniques focuses on how a machine learning model built in a source domain can be adapted in a different but related target domain, which is necessary to avoid reconstruction efforts. In the field of knowledge engineering, many beneficial and promising examples with domain adaptation have been found, including image classification,

object recognition, natural language processing, and feature learning [7–10].

In recent years, considerable research has been conducted on domain adaptation on the basis of deep architectures. Most published deep domain adaptation works can be roughly divided into three categories [11]: (1) discrepancy-based, (2) adversarial adaptation, and (3) reconstruction-based methods.

Typical discrepancy-based methods are shown in [12, 13]. They are usually implemented by adding a loss to minimize the distribution discrepancy between the source and target domains in the shared feature space. For example, Tzeng et al. [12] applied a single linear kernel to one layer for minimizing the maximum mean discrepancy (MMD), whereas Long et al. [13, 14] minimized MMD by applying multiple kernels to multiple layers across domains. Another impressive work is Deep Coral [15], which extends CORAL [16] to deep architectures and aligns the second-order statistics of the source and target distributions.

Another increasingly popular work is adversarial domain adaptation methods, which include adversarial discriminative and generative methods. The former aims to encourage domain confusion through an adversarial objective with respect to a domain discriminator. Tzeng et al. [17] proposed a unified adversarial domain adaptation framework that combines discriminative modeling, untied weight sharing, and a Generative Adversarial Network (GAN) loss [18]. Among the discriminative models, Tzeng et al. proposed a model with confusion loss [19] and also considered an inverted label GAN loss [17], whereas Ganin et al. [20] proposed a model with minimax loss. The latter combines the discriminative model with a generative component on the basis of GANs. Liu and Tuzel [21] developed a Coupled Generative Adversarial Network (CoGAN) that adopts two GANs, each corresponding to one of the domains, and the CoGAN learns a joint distribution of multidomain samples and enforces a weight-sharing constraint to limit the network capacity. The method presented in [22] also adopts GANs to generate source domain images to appear as if drawn from the target domain.

Typical reconstruction-based methods can be seen in [23–25]. Data reconstruction can be viewed as an auxiliary task to support the adaptation of the label prediction. Ghifary et al. [23] combined the standard convolutional network for source label prediction with a deconvolutional network [26] for target data reconstruction. Bousmalis et al. [24] introduced the notion of private and shared subspaces for each domain. Meanwhile, a reconstruction loss is integrated in the model by using a shared decoder, which learns to reconstruct the input sample through domain-specific and share features. Tan et al. [25] presented a selective learning algorithm that uses the reconstruction error to select useful unlabeled data from intermediate domains.

Despite the success achieved by domain adaptation, limited research can be found with respect to its application on fault diagnosis. Zhang et al. [27] took raw vibration signals as inputs of a deep convolutional neural network with a wide first-layer kernel convolutional neural network (WDCNN) model. They also used adaptive batch normalization (AdaBN) as the algorithm of domain adaptation to realize fault diagnosis under different load conditions and noisy environments. Lu et al. [28] introduced a deep CNN model with domain adaptation for fault diagnosis, and this model integrates MMD as the regularization term into the loss function of the model to reduce the cross-domain distribution difference. Zhang et al. [29] developed an adversarial domain adaptation model, which comprises a source feature extractor, a target feature extractor, a domain discriminator, and a label classifier, for fault diagnosis. Jian et al. [30] proposed a fusion CNN model that combines 1DCNN and Dempster–Shafer evidence theory to enhance the cross-domain adaptive capability for fault diagnosis. Tong et al. [31] proposed a bearing fault diagnosis domain adaptation method to find transferable features across domains, which were obtained by reducing marginal and conditional distributions simultaneously based on MMD. Li et al. [32] presented a deep domain adaptation method for

bearing fault diagnosis on the basis of the multikernel maximum mean discrepancies between domains in multiple layers to learn representations from the source domain applied to the target domain. Furthermore, Han et al. [33] proposed a new intelligent fault diagnosis framework, which extends the marginal distribution adaptation to joint distribution adaptation and guarantees an accurate distribution adaptation.

Improving CNN performance by learning additional discriminative features has become a recent trend. For example, contrastive loss [34] and center loss [35] are presented to learn discriminative deep features for face verification and recognition. Furthermore, Liu et al. [36] proposed a large-margin softmax loss to extend the softmax loss to large margin softmax, which leads to a large angular separability between the learned features. Chen et al. [37] proposed two discriminative feature learning approaches, namely, instance-based and center-based discriminative losses, with joint domain alignment and discriminative feature learning.

Inspired by these methods, we propose a novel deep domain adaptation model for bearing fault diagnosis with Deep Coral and center-based discriminative feature learning. By combining domain alignment and discriminative feature learning, the domain-invariant features extracted by the model can be well clustered and separable, which can clearly contribute to domain adaptation and classification.

The main contributions of this work include the following:

- (1) An end-to-end method with domain adaptation for fault diagnosis is proposed, that is, CACD-1DCNN. This method directly works on raw temporal signals and does not require time-consuming denoising preprocessing and a separate feature extraction algorithm.
- (2) By combining domain alignment and discriminative feature learning, CACD-1DCNN aims to extract domain-invariant features with improved intraclass compactness and interclass separability and guarantees high classification performance in the two domains.
- (3) Extensive experiments on the Case Western Reserve University (CWRU) bearing datasets demonstrate that CACD-1DCNN achieves superior diagnosis performance over existing baseline methods.
- (4) Furthermore, network visualization and loss analysis provide an intuitive presentation of the adaptation results and verify the effectiveness of our method.

The remaining parts are organized as follows. In Section 2, the domain adaptation problem of fault diagnosis is formulated and the basic theories of Deep Correlation Alignment and Center-Based Discriminative Loss are introduced. In Section 3, the proposed intelligent fault diagnosis method, CACD-1DCNN, is presented. The comparison methods, the experiments, and discussion are given in Section 4. Finally, the conclusions are drawn in Section 5.

## 2. Theoretical Background

**2.1. Problem Formulation.** Traditional machinery fault diagnosis aims to identify the fault location and severity of the unknown fault set on the basis of a prior known fault set. Taking fault diagnosis as an example, the collected labeled raw vibration temporal signals are taken as the source domain samples and the collected unlabeled raw vibration temporal signals as the target domain samples. The assumption is that the distributions of the source and target domains are the same, and the learned fault patterns obtained from the labeled training data can be directly applied to the unlabeled test data. However, a difference inevitably exists between the source and target domains in practical tasks, thereby deteriorating the model generalization capability across domains. Therefore, the domain adaptation problem for fault diagnosis has attracted increasing attention. This study focuses on the problem of unsupervised domain adaptation, that is, the target domain data have no label.

Studies on the unsupervised domain adaptation problem of rolling bearing fault diagnosis are generally conducted under the following assumptions:

- (1) The source and target domains are related to each other but have different distributions
- (2) For different domains, the fault diagnosis task is the same, which is to share class labels
- (3) The labeled samples from the source domain are used for training, whereas the unlabeled samples from the target domain are available for training and testing

Formally, domain  $\mathcal{D}$  is composed of  $m$ -dimensional feature space  $\mathcal{X} \subset R^m$  with marginal probability distribution  $P(X)$ , where  $X = \{x_1, \dots, x_n\} \in \mathcal{X}$ . Task  $\mathcal{F}$  consists of two components: label space  $\mathcal{Y}$  and predictive function  $f(X)$  corresponding to the labels.  $f(X) = Q(Y|X)$  is also the conditional probability distribution, and  $Y \in \mathcal{Y}$ , where  $Y$  represents the possible machine health condition. The following is the formal definition of the unsupervised domain adaptation problem for fault diagnosis. Given labeled source domain dataset  $D^S = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$  and unlabeled target dataset  $D^T = \{(x_i^t)\}_{i=1}^{N_t}$ , where  $N_s$  and  $N_t$  are the numbers of samples of the source and target domains, respectively. The eigenspaces of  $D^S$  and  $D^T$  (that is,  $\mathcal{X}^s = \mathcal{X}^t$ ), the label space (that is,  $\mathcal{Y}^s = \mathcal{Y}^t$ ), and conditional probability distribution  $Q_s(y^s | x^s) = Q_t(y^t | x^t)$  are assumed to be the same. However, the marginal probability distributions of the two domains, that is,  $P_s(x^s) \neq P_t(x^t)$ , are different. Unsupervised domain adaptation aims to use labeled  $D^S$  to learn classifier  $f(x^t) \rightarrow y^t$  for predicting labels  $y^t$  of  $D^T$ , where  $y^t \in \mathcal{Y}^t$ .

**2.2. Deep Correlation Alignment.** To fill the gap between the domains, CORAL loss is adopted by aligning the second-order statistics of the source and target features. In the activations computed at a given layer,  $x_i^s$  and  $x_i^t$  are  $d$ -dimensional representations. The domain discrepancy loss measured by CORAL loss ( $L_{CA}$ ) [16], as shown below,

minimizes the distribution discrepancy between the second-order statistics (covariance) of the source and target features.

$$L_{CA} = \frac{1}{4d^2} \|C_s - C_T\|_F^2, \quad (1)$$

where  $\|\cdot\|_F^2$  denotes the squared matrix Frobenius norm and  $C_S$  and  $C_T$  are the covariance matrices of the source and target features, respectively. According to reference [16],  $C_S$  and  $C_T$  are, respectively, computed as follows:

$$\begin{aligned} C_S &= A_S J A_S^T, \\ C_T &= A_T J A_T^T, \end{aligned} \quad (2)$$

where  $J$  is the centering matrix [38]. Taking the source domain as an example,  $J$  is a matrix of  $N_S \times N_S$ , and it is derived as follows:

$$\begin{aligned} J_{ii} &= \frac{1}{N_S}, \\ J_{ij} &= \frac{1}{N_S(N_S - 1)}, \\ & \quad i \neq j. \end{aligned} \quad (3)$$

The training process is realized by mini-batch Stochastic Gradient Descent (SGD) in which only a batch of training samples is aligned in each iteration.

**2.3. Center-Based Discriminative Loss.** To make the deep features learned by the deep CNN model further discriminative, center-based discriminant loss  $L_{CD}$  is adopted [37], which is different from center loss [35]. The latter penalizes the distance of each sample to the center of the corresponding class, whereas the former not only has the characteristic of center loss but also enforces large margins among centers across different categories. Center-based discriminant loss is defined as follows:

$$\begin{aligned} L_{CD} &= \beta \sum_{i=1}^{n_s} \max\left(0, \|f_i^s - c_{y_i}\|_2^2 - m_1\right) \\ & \quad + \sum_{i,j=1, i \neq j}^c \max\left(0, m_2 - \|c_i - c_j\|_2^2\right). \end{aligned} \quad (4)$$

The loss is composed of two items. The first item is used to measure the intraclass compactness, whereas the second item is used to measure the interclass separability.  $\beta$  is the trade-off parameter, and  $m_1$  and  $m_2$  are the two constraint margins.  $f_i^s \in R^d$  represents the deep features of the  $i$ -th training sample in the fully connected layer, and  $n$  is the number of neurons of the fully connected layer.  $c_{y_j} \in R^d$  denotes the class center of the  $y_j$ -th sample corresponding to the deep features,  $y_j \in \{1, 2, \dots, c\}$ , where  $c$  is the number of classes.

Equation (4) shows that the center-based discriminative loss forces the distance between intraclass samples to be no more than  $m_1$  and the distance between the interclass samples to be no less than  $m_2$ . Obviously, this penalty item can make the deep features further discriminative.

Ideally, class center  $c_{y_j}$  should be calculated by averaging the deep features of all samples, which is evidently inefficient and unrealistic. In practical applications, we update the central point through the mini-batch training samples. In each iteration, we calculate the central point by averaging the features of the corresponding class. The updated formula in each iteration is presented as follows:

$$\Delta_{c_j} = \frac{\sum_{i=1}^b \delta(y_i = j)(c_j - f_i^s)}{1 + \sum_{i=1}^b \delta(y_i = j)}, \quad (5)$$

$$c_j^{t+1} = c_j^t - \lambda \Delta_{c_j}^t. \quad (6)$$

When the condition is true,  $\delta(\text{condition}) = 1$ , and 0 otherwise.  $b$  denotes the batch size, and  $\lambda$  is the learning rate. Every class center is initialized as the ‘‘batch class center’’ in the first iteration, and it is updated according to (5) and (6) for the next batch of samples in each iteration.

### 3. Fault Diagnosis Framework Based on CACD-1DCNN

**3.1. CACD-1DCNN Fault Diagnosis Model.** CACD-1DCNN is proposed to solve the cross-domain learning problem in the bearing fault diagnosis area. As illustrated in Figure 1, taking CNN as the main architecture, the two-stream CNN architecture with shared weights is adopted, and the model employs a domain adaptation layer with correlation alignment and center-based losses before the classifier.

As the input of the two-stream, the labeled source and unlabeled target data are fed into the CACD-1DCNN model during the training process. Subsequently, domain-invariant features with discriminative raw vibration signals are extracted through the multiple convolutional and pooling layers. The distribution discrepancy is minimized at the last fully connected layer. Theoretically, correlation alignment can be performed at multiple layers in parallel. Empirical evidence [15, 39] indicates that a solid performance is obtained even if this alignment is conducted only once. As a common practice, correlation alignment loss ( $L_{CA}$ ) is performed after the last fully connected layer. Similarly, center-based discriminative loss ( $L_{CD}$ ) is generally placed after the last fully connected layer. Therefore, the two kinds of loss functions in the proposed model are trained on the basis of the features extracted by the last fully connected layer. In addition to the conventional softmax loss function ( $L_s$ ) based on source domain, the loss function of the CACD-1DCNN model is defined as follows:

$$L = L_s + \alpha L_{CA} + \beta L_{CD}, \quad (7)$$

where  $\alpha$  and  $\beta$  are the trade-off parameters for balancing the contributions of the domain discrepancy and discriminative losses.

Only the source domain data are discriminated here. During the training process, the source features are discriminant learning and aligned with the target features. Joint training with classification, correlation alignment, and center-based discriminative losses between the two domains

in the last fully connected layer can adapt the learned representations in the source domain for application to the target domain. This joint training can also guarantee domain-invariant features with improved intraclass compactness and interclass separability. Meanwhile, the extracted features can efficiently improve the cross-domain testing performance.

**3.2. Architecture Designs of 1DCNN.** Considering that the bearing vibration signals collected by acceleration sensors are usually 1D, using a 1DCNN is reasonable for the processing of vibration signals. In this study, the 1DCNN is adopted to deal with bearing fault diagnosis. The network structure is composed of four convolution and pooling layers, two fully connected layers, and a softmax layer at the end. The first convolutional layer uses wide kernel for extracting feature and suppressing high-frequency noise. Small convolutional kernels in the following layers are used to deepen the network for multilayer nonlinear mapping and preventing overfitting [27]. The parameters of 1DCNN are presented in Table 1. The pooling type is max pooling, and the activation function is ReLU. To minimize the loss function, the Adam stochastic optimization algorithm is applied to train our model and the learning rate is set to  $1e-4$ . The experiments are conducted using the TensorFlow toolbox of Google.

**3.3. Data Augmentation.** Without sufficient training samples, the model can easily result in overfitting. Data augmentation techniques are commonly used in computer vision to increase the generalization of networks by adding the number of training samples. In fault diagnosis, the vibration signals collected by the acceleration sensor is 1D, and overlap sampling can easily obtain a large number of data by slicing the training samples with overlap. Figure 2 illustrates a vibration signal with 120,000 points. We can take 2,048 data points from this signal as a sample. We can also offset it by a certain amount to be the second sample.

## 4. Experimental Analysis of the Proposed CACD-1DCNN Model

**4.1. Data Description.** The bearing fault data used for experimental validation were obtained from the Bearing Data Center of CWRU [40]. The data were collected from a motor driving mechanical system under four different loads (0, 1, 2, and 3 hp) and three different locations (fan end, drive end, and base). The sampling frequency includes 48 and 12 kHz. The bearing has three fault types: outer race fault (OF), inner race fault (IF), and roller fault (RF). Each fault type contains fault diameters of 0.007, 0.014, and 0.021 inches, respectively; there are also normal condition (N), a total of 10 health states.

In this study, vibration signals of different fault locations and different health states with a sampling frequency of 12 kHz at the driving end of rolling bearing are selected for experimental research. The detailed description of the datasets is shown in Table 2. Three datasets are acquired

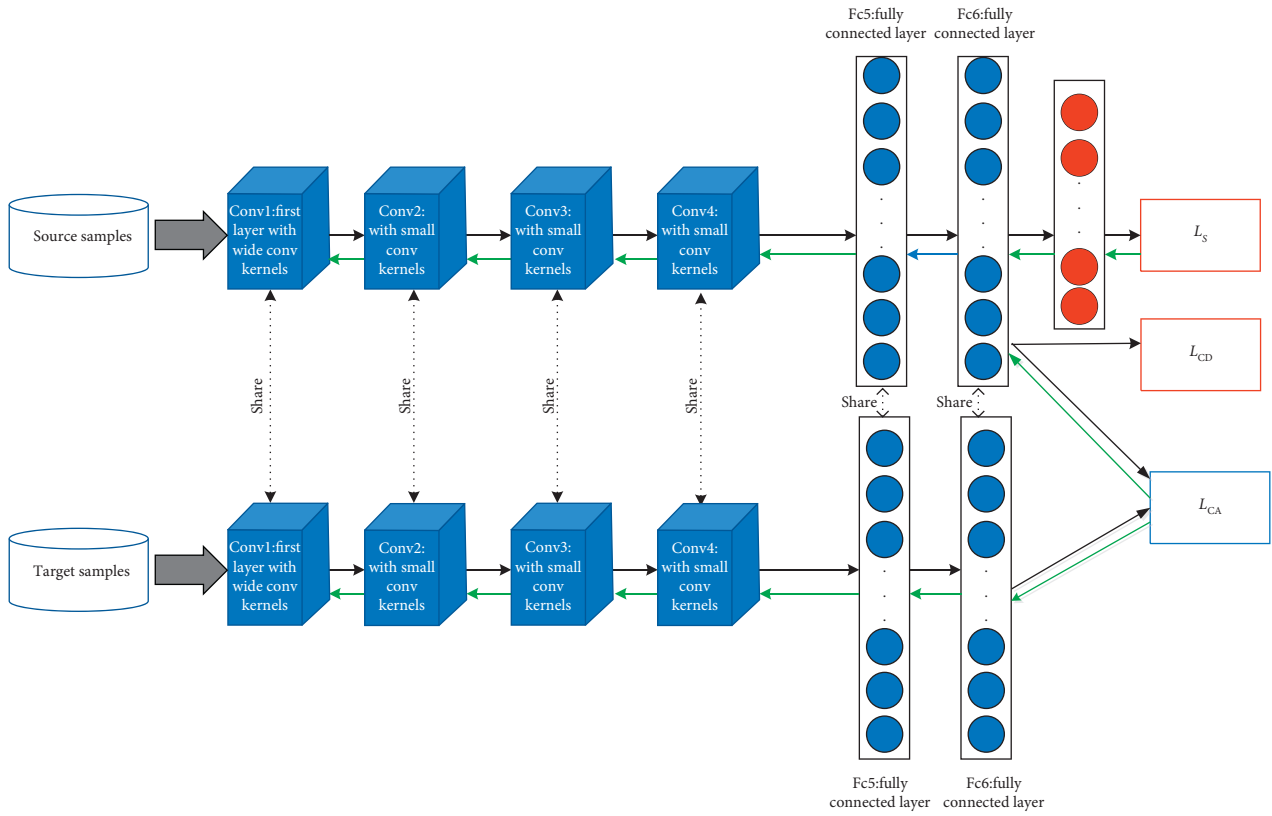


FIGURE 1: Architecture of CACD-1DCNN.

TABLE 1: Details of 1DCNN architecture used in experiments.

No.	Layer type	Kernel size	Stride	Kernel number	Output	Padding
1	Convolution1	$32 \times 1$	$8 \times 1$	32	$253 \times 32$	Yes
2	Pooling1	$2 \times 1$	$2 \times 1$	32	$126 \times 32$	No
3	Convolution2	$3 \times 1$	$2 \times 1$	64	$124 \times 64$	Yes
4	Pooling2	$2 \times 1$	$2 \times 1$	64	$62 \times 64$	No
5	Convolution3	$3 \times 1$	$2 \times 1$	64	$60 \times 64$	Yes
6	Pooling3	$2 \times 1$	$2 \times 1$	64	$30 \times 64$	No
7	Convolution4	$3 \times 1$	$1 \times 1$	64	$28 \times 64$	Yes
8	Pooling4	$2 \times 1$	$2 \times 1$	64	$14 \times 64$	No
9	Fully connected1	100		1	$100 \times 1$	
10	Fully connected2	64		1	$64 \times 1$	
11	Softmax	10		1	10	

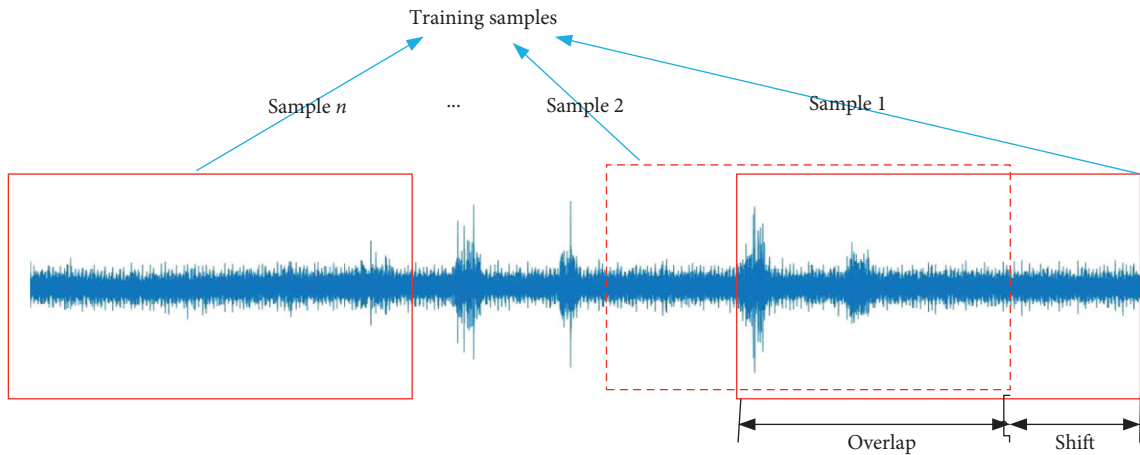


FIGURE 2: Schematic of overlap sampling.

TABLE 2: Description of 12 kHz drive-end bearing datasets.

Fault location		None	RF			IF			OF		
Category labels		0	1	2	3	4	5	6	7	8	9
Fault diameter (inch)		0	0.007	0.014	0.021	0.007	0.014	0.021	0.007	0.014	0.021
Dataset A (1 HP)	Train	660	660	660	660	660	660	660	660	660	660
	Test	25	25	25	25	25	25	25	25	25	25
Dataset B (2 HP)	Train	660	660	660	660	660	660	660	660	660	660
	Test	25	25	25	25	25	25	25	25	25	25
Dataset C (3 HP)	Train	660	660	660	660	660	660	660	660	660	660
	Test	25	25	25	25	25	25	25	25	25	25

under three loads of 1, 2, and 3 HP, respectively. Each large dataset contains training and testing samples, and each sample contains 2,048 data points. To increase the number of training samples, the overlap sampling technique is used. In this study, the training samples are overlapped to augment data. However, no overlapping is observed in the testing set. Therefore, each dataset is composed of 6,600 training samples and 250 test samples of 10 health states.

**4.2. Accuracy across Different Domains.** The source domain samples have labels, whereas the target domain samples have none. Owing to the three domains, the experiments are conducted in six domain transfer scenarios,  $A \rightarrow B$ ,  $A \rightarrow C$ ,  $B \rightarrow C$ ,  $C \rightarrow B$ ,  $C \rightarrow A$ , and  $B \rightarrow A$ . Taking  $A \rightarrow B$  as an example, dataset A is the source domain, whereas dataset B is the target domain.

**Comparison Methods.** The proposed method is compared with several successful machine learning methods to verify the effectiveness of the CACD-1DCNN model:

- (1) SVM
- (2) Multilayer perceptron (MLP)
- (3) Deep neural network (DNN) [1]
- (4) WDCNN proposed in [27] and the domain adaptation capacity from the AdaBN
- (5) OFNN-DE proposed in [30]
- (6) Adversarial adaptive model based on 1-D CNN (A2CNN) [29]

1–3 and 6 are methods that work with the data transformed through fast Fourier transform, whereas 4 and 5 are CNN-based methods that work with normalized raw signals. Notably, the OFNN in [30] is not used here because the OFNN uses the diagnostic result of data fusion between the drive-end and fan-end datasets, which are different from the datasets used in the experiments in this research. By contrast, the datasets used by OFNN-DE are the same as the datasets used in the experiments in this study. Furthermore, the diagnosis accuracy of the OFNN is slightly lower than that of the method used in this research.

For a fair comparison, we adopt the accuracy reported by other authors with the same setting or conduct experiments by using the source code provided by the authors.

A total of 10 experiments are conducted for each domain transfer scenario to reduce the influence of random factors.

The experimental results of the six domain scenarios are displayed in Figure 3. In the domain shift scenarios of  $A \rightarrow B$ ,  $A \rightarrow C$ ,  $B \rightarrow C$ , and  $C \rightarrow B$ , the test accuracy of each scenario reaches 100%. In the domain shift scenarios of  $C \rightarrow A$  and  $B \rightarrow A$ , the test accuracies exceed 97.6% and 97.2%, respectively. These results show that the domain adaptation performance of the proposed method is remarkable and stable.

The comparison with other approaches is shown in Figure 4. The average performance of the CACD-1DCNN is better than that of the A2CNN and six other baseline methods. The CACD-1DCNN also achieves the state-of-the-art average accuracy of domain adaptation in all domain transfer scenarios.

As illustrated in Figure 4, the performance of SVM, MLP, and DNN in domain adaptation is poor, with average accuracies of 66.63%, 80.40%, and 78.05% in the six scenarios, respectively. These results suggest that the sample distribution differs under varying conditions and the model trained in one working condition is unsuitable for fault diagnosis in another condition.

Compared with recent approaches, such as OFNN-DE and A2CNN, our method achieves an average accuracy of 99.47%, which is higher than those of OFNN-DE and A2CNN with average accuracies of 98.73% and 99.21%, respectively. This result shows that the features learned by the proposed method have better domain invariance and fault discrimination than those learned by other methods.

In five out of six shifts, that is,  $A \rightarrow B$ ,  $A \rightarrow C$ ,  $B \rightarrow C$ ,  $C \rightarrow B$ , and  $C \rightarrow A$ , the fault diagnosis accuracy of the proposed method achieves state-of-the-art domain adaptation performance and reaches up to 100% in the first four domain shifts. In the domain transfer scenario of  $B \rightarrow A$ , the accuracy of the proposed method is 98%, which is, respectively, 0.18% and 0.5% lower than those of the A2CNN and OFNN-DE methods and far better than the accuracies of the SVM, MLP, DNN, and WDCNN methods. On this basis, the CACD-1DCNN can well learn domain-invariant and fault-discriminate features and effectively solve the domain adaptation problem caused by different loads of bearing data.

Taking the domain shift scenarios of  $C \rightarrow B$ ,  $C \rightarrow A$ , and  $B \rightarrow A$  as examples, for the CACD-1DCNN model, we compare the test accuracy of the target domain under the four loss functions of  $L_s$ ,  $L_s + L_{CA}$ ,  $L_s + L_{CD}$ , and  $L_s + L_{CA} + L_{CD}$  (for simplicity, the coefficients of each loss function is omitted), as presented in Table 3. The results of other shifts

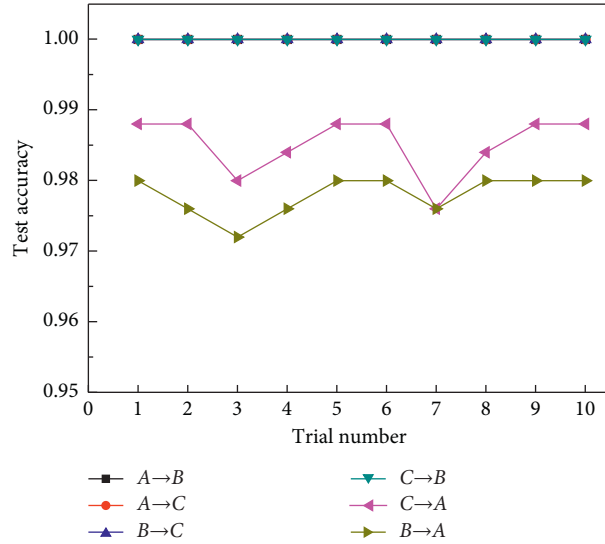


FIGURE 3: 10 diagnosis results of six domain-shift scenarios of the proposed method: (a)  $A \rightarrow B$ , (b)  $A \rightarrow C$ , (c)  $B \rightarrow C$ , (d)  $C \rightarrow B$ , (e)  $C \rightarrow A$ , and (f)  $B \rightarrow A$ .

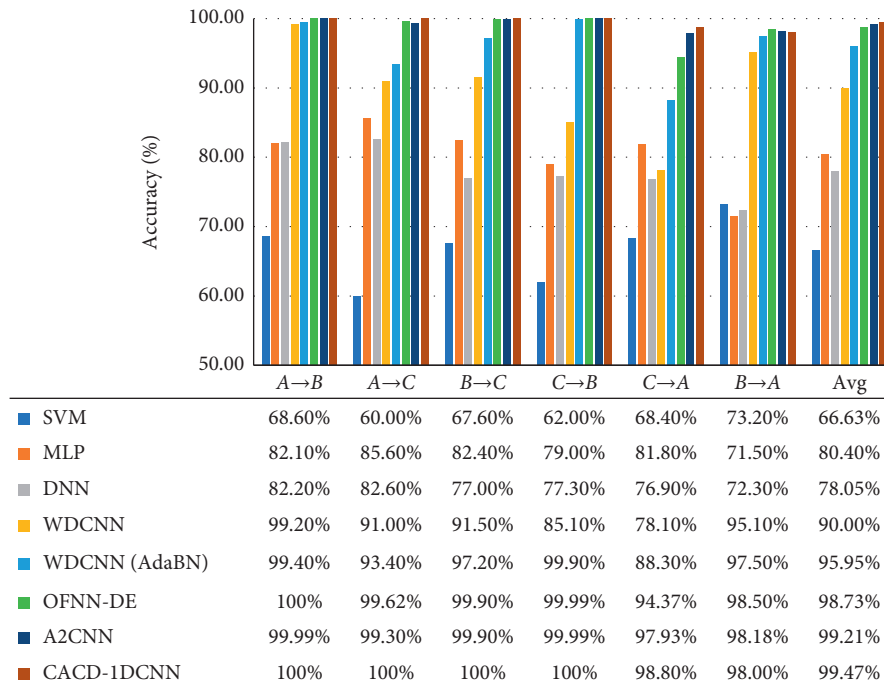


FIGURE 4: Accuracy on six domain shifts of the proposed and compared methods.

TABLE 3: Accuracy of the CACD-1DCNN model under four different loss functions.

Losses	Domain shifts			Average
	$C \rightarrow B$	$C \rightarrow A$	$B \rightarrow A$	
$L_s$	0.948	0.916	0.96	0.941
$L_s + L_{CA}$	1	0.972	0.976	0.983
$L_s + L_{CD}$	1	0.976	0.972	0.983
$L_s + L_{CA} + L_{CD}$	1	0.988	0.98	0.989

are similar. We observe that the target test accuracy is worst with the loss function of  $L_s$  because no adaptive strategy is adopted between the source and target domains. The target

test accuracy is in the middle level with the loss functions of  $L_s + L_{CA}$  and  $L_s + L_{CD}$ , and the target test accuracy is the highest with the loss function of  $L_s + L_{CA} + L_{CD}$ . Therefore,

in the case of  $L_s + L_{CA}$  and  $L_s + L_{CD}$ , the model classification performance is comparable, and only when jointly supervised by the three loss functions can the proposed model achieve the best performance.

The results confirm that (1) the CACD-1DCNN is effective in filling the domain gap and (2)  $L_{CD}$  is added on the basis of  $L_{CA}$ , that is, by combining domain alignment and discriminative feature learning, the proposed model guarantees that the domain-invariant features are extracted with improved intraclass compactness and interclass separability. The gap between the class clusters and the hyperplane is large, which is conducive to the correct classification of target samples near the edge or far from their corresponding class centers.

Furthermore, taking the domain shift scenario of  $C \rightarrow A$  as an example, the accuracy of the training and testing stages in the case of the joint loss of  $L_s + L_{CA} + L_{CD}$  is illustrated in Figure 5. This approach clearly helps us to achieve improved performance in the target domain while maintaining a strong classification accuracy in the source domain.

Taking the domain shift of  $C \rightarrow A$  as an example, the domain loss and intraloss are analyzed. Notably, the proposed model adopts batch values. Although batch values cannot fully represent the distance between the entire source and target domains, it is a practical and fast approximation method for classifying samples.

The domain loss under the four loss functions is illustrated in Figure 6. In the case of  $L_s$ , where only the source domain is trained, the feature representations obtained from the source domain are likely to be different from the target features because the sample features of the target domain are not considered at all during the model training. Therefore, overfitting occurs, and the domain loss is great. In the case of  $L_s + L_{CD}$ , without considering domain alignment and only considering discriminant learning, the domain loss is smaller than that in the case of  $L_s$ , within the range of 0–0.55. Only  $L_s + L_{CA}$  and  $L_s + L_{CA} + L_{CD}$  are compared in Figure 7 for clarity. The domain loss is minimal, and the domain loss curve is smooth in the case of  $L_s + L_{CA} + L_{CD}$ , indicating that the slight change of weight causes a slight change of distance between domains. A further stable and accurate domain adaptation model can be obtained by considering the domain alignment and discriminative feature learning.

The intraclass loss under the four loss functions is illustrated in Figure 8. Evidently, the intraclass loss is large in the case of  $L_s$  because the sample features of the target domain are not considered. The case of  $L_s + L_{CD}$  takes second place. The cases of  $L_s + L_{CA}$  and  $L_s + L_{CA} + L_{CD}$  are small, suggesting that discriminant learning can achieve a small intraclass loss only in the model training of joint domain alignment. The comparison of the intraclass losses of  $L_s + L_{CA}$  and  $L_s + L_{CA} + L_{CD}$  shows that the curve is smooth in the case of  $L_s + L_{CA} + L_{CD}$ , indicating that the model is stable and reasonable in this case.

**4.3. Sensitivity Analysis of Fault.** For each type of fault detection, we introduce three evaluation indexes, namely, Precision, Recall, and  $F$ -Measure, to further analyze the

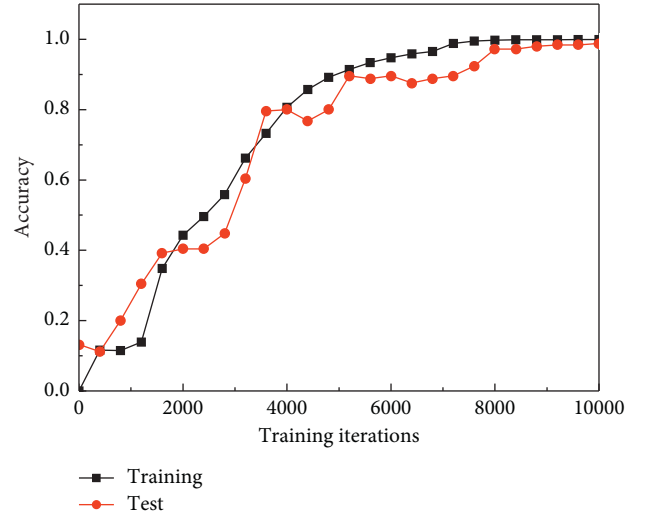


FIGURE 5: Training and test accuracy of domain transfer scenario of  $C \rightarrow A$  in the proposed model.

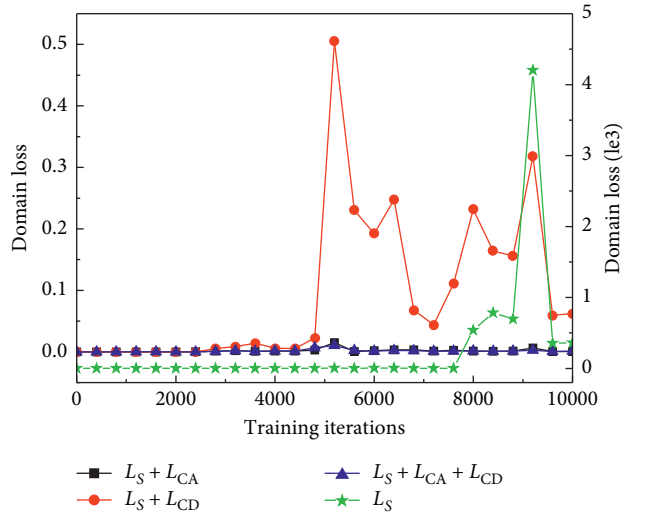


FIGURE 6: The domain loss of the proposed model with four loss functions.

sensitivity of the proposed CACD-1DCNN method. In the multiclassification problem of fault diagnosis, for each fault category  $f$ , Precision and Recall are defined as follows:

$$\text{Precision}(f) = \frac{TP}{TP + FP}, \quad (8)$$

$$\text{Recall}(f) = \frac{TP}{TP + FN},$$

where True Positive (TP) represents the number of faults correctly identified as fault category  $f$ , False Positive (FP) means the number of faults wrongly identified as fault category  $f$ , and False Negative (FN) represents the number of faults  $c$  incorrectly labeled as not belonging to  $f$ .

$F$ -Measure is defined as a reference for diagnosis analysis. The calculation method of  $F$ -Measure is as follows:



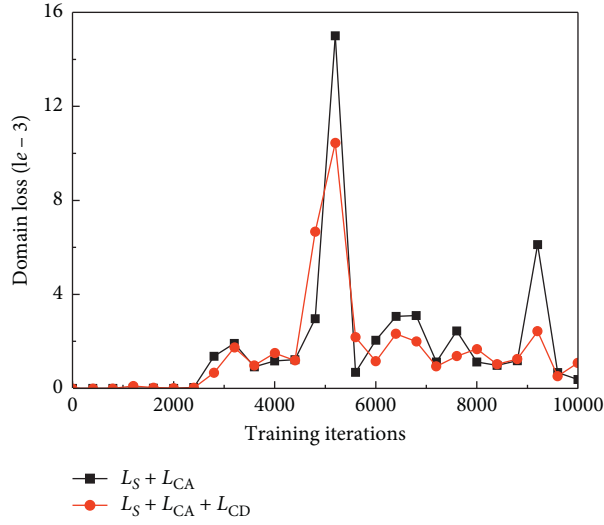


FIGURE 7: The domain loss of the proposed model with the two loss functions.

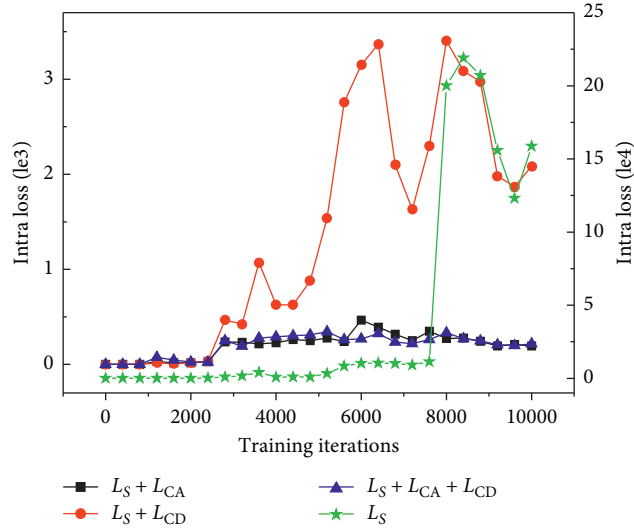


FIGURE 8: The interclass loss of the proposed model with four loss functions.

$$F = (1 + \alpha^2) \times \frac{\text{Precision} \times \text{Recall}}{\alpha^2 \times (\text{Precision} + \text{Recall})}. \quad (9)$$

$F$ -Measure denotes the geometric weighted average of Precision and Recall, with  $\alpha$  as the weight. Setting  $\alpha$  to 1 indicates that Precision is as important as Recall. When  $\alpha > 1$ , Precision is important; when  $\alpha < 1$ , Recall is important. In this study,  $\alpha$  is set to 1; the closer the  $F$ -Measure to 1, the better the fault diagnosis performance. This evaluation method considers the Precision and Recall. The highest  $F$ -Measure is 1. The Precision, Recall, and  $F$ -Measure of each health state in the comparison method A2CNN and the proposed method CACD-1DCNN are presented in Table 4, and the comparison results of other methods are similar.

For the first (rolling body) and fourth (inner raceway) fault types, both of which have a fault size of 0.007 inch,

the CACD-1DCNN method has low Precision values in the domain shift scenario  $B \rightarrow A$ , which are 89% and 93%, respectively. Thus, approximately 10% of these kinds of fault alerts in this domain shift are unreliable. For the first fault type, the Precision of the proposed method in the domain shift of  $C \rightarrow A$  is 89%, indicating that 11% of the samples are incorrectly classified as this fault category. In A2CNN, we can see that, in some domain shifts, based on the first, second, third, and fifth fault types and normal state, there are some fault alarms which are not reliable.

For the third fault type, the rolling body fault size is 0.021 inch. The Recall values of the CACD-1DCNN method in domain shift scenarios  $C \rightarrow A$  and  $B \rightarrow A$  are low, which are 88% and 80%, respectively. That is, 12% of these faults are undetected in the domain shift scenario of  $C \rightarrow A$ , whereas 20% are undetected in the domain shift  $B \rightarrow A$ . In A2CNN,

TABLE 4: Precision, Recall, and  $F$ -Measure of the proposed CACD-1DCNN and A2CNN on six domain shifts.

Fault location	$N$	RF			IF			OF		
Fault diameter (inch)		0.007	0.014	0.021	0.007	0.014	0.021	0.007	0.014	0.021
Category labels	0	1	2	3	4	5	6	7	8	9
Precision										
A2CNN										
$A \rightarrow B$	100	100%	100%	99.88%	100%	100%	100%	100%	100%	100%
$A \rightarrow C$	93.46	100%	100%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow C$	100	90.07%	92.59%	100%	100%	100%	100%	100%	100%	100%
$C \rightarrow B$	100	100%	100%	100%	100%	99.01%	100%	100%	100%	100%
$C \rightarrow A$	100	90.91%	90.40%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow A$	100	99.88%	100%	100%	100%	100%	100%	100%	100%	100%
CACD-1DCNN										
$A \rightarrow B$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$A \rightarrow C$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow C$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$C \rightarrow B$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$C \rightarrow A$	100%	89%	100%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow A$	100%	89%	100%	100%	93%	100%	100%	100%	100%	100%
Recall										
A2CNN										
$A \rightarrow B$	100	100%	99.88%	100%	100%	100%	100%	100%	100%	100%
$A \rightarrow C$	100	100%	93.00%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow C$	100	100%	100%	81.75%	100%	100%	100%	100%	100%	100%
$C \rightarrow B$	100	100%	100%	100%	100%	100%	100%	100%	99.00%	100%
$C \rightarrow A$	100	100%	100%	79.38%	100%	100%	100%	100%	100%	100%
$B \rightarrow A$	100	100%	100%	99.88%	100%	100%	100%	100%	100%	100%
CACD-1DCNN										
$A \rightarrow B$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$A \rightarrow C$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$B \rightarrow C$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$C \rightarrow B$	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
$C \rightarrow A$	100%	100%	100%	88%	100%	100%	100%	100%	100%	100%
$B \rightarrow A$	100%	100%	100%	80%	100%	100%	100%	100%	100%	100%
F-Measure										
A2CNN										
$A \rightarrow B$	1	1	0.9994	0.9994	1	1	1	1	1	1
$A \rightarrow C$	0.9662	1	0.9637	1	1	1	1	1	1	1
$B \rightarrow C$	1	0.9478	0.9615	0.8996	1	1	1	1	1	1
$C \rightarrow B$	1	1	1	1	1	1	1	1	0.9950	1
$C \rightarrow A$	1	0.9524	0.9496	0.8850	1	1	1	1	1	1
$B \rightarrow A$	1	0.9994	1	0.9994	1	1	1	1	1	1
CACD-1DCNN										
$A \rightarrow B$	1	1	1	1	1	1	1	1	1	1
$A \rightarrow C$	1	1	1	1	1	1	1	1	1	1
$B \rightarrow C$	1	1	1	1	1	1	1	1	1	1
$C \rightarrow B$	1	1	1	1	1	1	1	1	1	1
$C \rightarrow A$	1	0.9434	1	0.9362	1	1	1	1	1	1
$B \rightarrow A$	1	0.9434	1	0.8889	0.9615	1	1	1	1	1

we can see that, in some domain shifts, based on the second, third, and eighth fault types, there are some undetected faults.

Similarly, the  $F$ -Measure values in the domain shift scenarios of  $C \rightarrow A$  and  $B \rightarrow A$  for the first fault type are all 0.9434. For the third fault type, the  $F$ -Measure values in the domain shifts of  $C \rightarrow A$  and  $B \rightarrow A$  are 0.9362 and 0.8889, respectively. The  $F$ -Measure in the domain shift of  $B \rightarrow A$  for the fourth fault type is 0.9615. All the other  $F$ -Measure values of the fault classes are 1. In A2CNN, we can see that, in some domain shifts, based on the first, the second, the third, and the eighth fault types and normal state, the  $F$ -Measure values are less than 1.

In general, the Precision, Recall, and  $F$ -Measure of the CACD-1DCNN are higher than that of the A2CNN, which means that the CACD-1DCNN has less false alarms and missed alarms. Except for a few third fault types in domain shift  $B \rightarrow A$ , which are incorrectly classified into the first and fourth fault types, and a few third fault types in domain shift  $C \rightarrow A$ , which are incorrectly classified into the first fault type, the CACD-1DCNN method divides all categories into the correct classes. The results show that, after combining the domain alignment and discriminative feature learning, the classification performance of the proposed method achieves remarkable improvement.

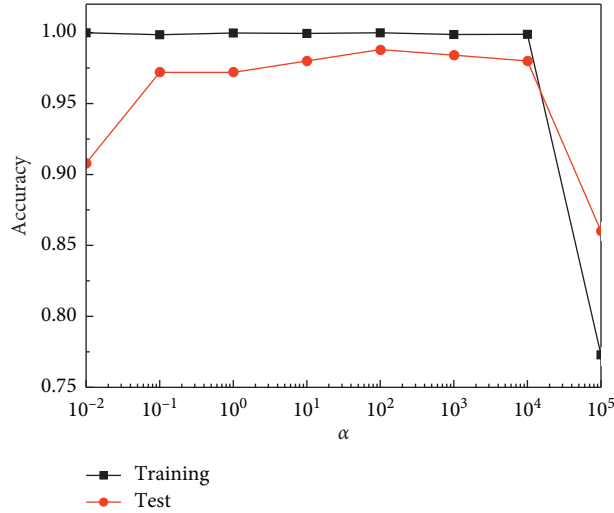


FIGURE 9: Accuracy corresponding to different  $\alpha$  in domain shift of  $C \rightarrow A$ .

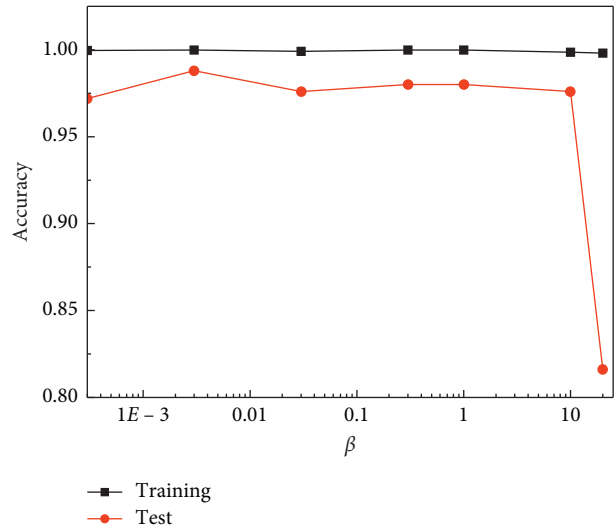


FIGURE 10: Accuracy corresponding to different  $\beta$  in domain shift of  $C \rightarrow A$ .

**4.4. Parameter Sensitivity.** In this section, we study hyperparameter  $\alpha$ , which is a critical coefficient for cross-validation. A high value of  $\alpha$  may force networks to learn oversimplified low-rank feature representations. Although a high value may lead to perfectly aligned covariances, it may not be useful for classification. Meanwhile, a small  $\alpha$  may be insufficient to bridge the domain shift.

Taking the domain shift of  $C \rightarrow A$  as an example, the results of  $\alpha$  with different values are illustrated in Figure 9.  $\beta$  is fixed at 0.003. Similar trends are observed in other domain transfer scenarios. A large range of  $\alpha$  ( $\alpha \in [10^{-1}, 10^4]$ ) can be selected to obtain better results than those of the best baseline methods. When the value of  $\alpha$  is larger than  $10^4$ , the accuracy rapidly decreases. The effectiveness and robustness of the proposed method are further verified.

With a fixed  $\alpha$  of 100, we consider the influence of parameter  $\beta$ , which balances the discriminant loss to increase the intraclass compactness and interclass

dispersion. A large  $\beta$  can produce deep discriminate features, whereas a small  $\beta$  is insufficient to improve the discrimination of features. Figure 10 shows the change in accuracy in the domain shift scenario of  $C \rightarrow A$  when  $\beta$  ( $\beta \in \{0.0003, 0.003, 0.03, 0.3, 1, 10, 20\}$ ) takes different values. When  $\beta$  is very small, the classification accuracy of the target domain is high. At this time, correlation alignment loss plays a role and the model classification performance is high. With the increase of  $\beta$ , when the domain alignment keeps up with the change of the source features under the influence of discriminant loss, a domain adaptation model with high accuracy is obtained. However, when  $\beta$  is larger than a certain interval, which is 10 in this case, the classification performance of the target domain is poor. The reason may be because the discriminant influence is too large to exceed the speed of domain alignment. In this case, we can conclude that when  $\beta$  is taken  $(0, 10]$ , the test accuracy remains high.

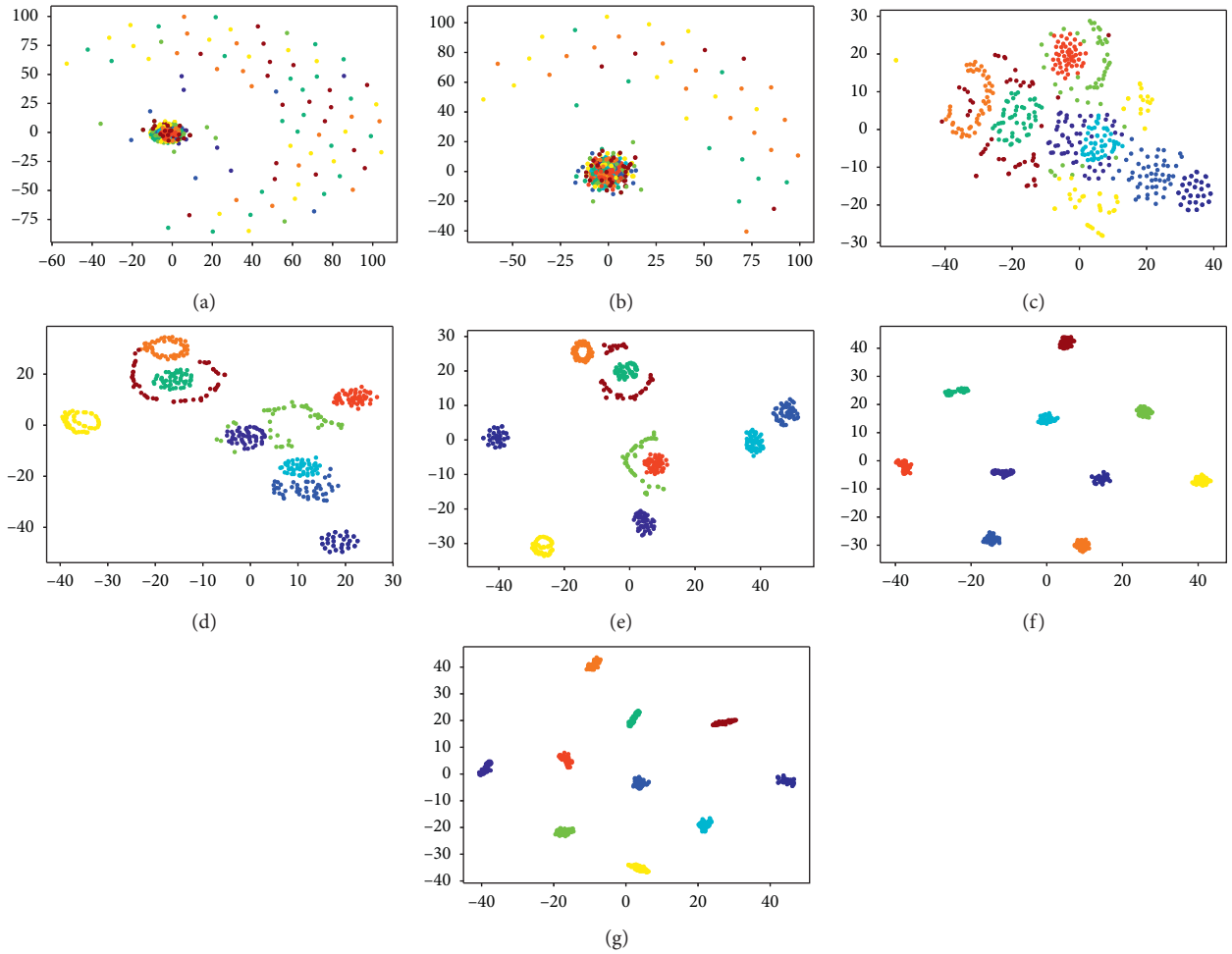


FIGURE 11: Feature visualization via t-SNE for domain shift of  $B \rightarrow C$ : feature representations for original test signals, four convolutional layers, and two fully connected layers, respectively. (a) Original signal. (b) Conv1. (c) Conv2. (d) Conv3. (e) Conv4. (f) FC5 (g) FC6.

The experimental results reveal that appropriate trade-off parameters between domain alignment and discriminant feature learning in the CACD-1DCNN model can improve the domain adaptation performance.

**4.5. Network Visualizations.** To further describe the effectiveness of the CACD-1DCNN, the t-SNE technology [41] is adopted for visualizing the feature representations of the proposed approach in all convolutional and fully connected layers. Domain scenario  $B \rightarrow C$  in Figure 11 is taken as an example. Three points are worth noting. (1) As the number of layers of the CACD-1DCNN model increases, the signals become increasingly separable at each layer, indicating the necessity of a deep structure. (2) In the third and fourth convolutional layers, the phenomenon of linear inseparability of feature representations occurs. In the fully connected layer, the feature representations of all faults are linearly separable. Therefore, the nonlinear expression ability of the model increases as the number of layers increases. (3) The feature representation of signals in the first convolutional layer is similar to that of the original signals and fails to show any separability. In the third and fourth

convolutional layers, the signal samples gradually show separability. At the fully connected layer, the faults can be well distinguished. The following describes the core idea of the proposed model to implement domain adaptation. The correlation of the data is initially removed, and then recorrelation operations are performed on the basis of the information of the target domain.

## 5. Conclusion

In this work, we propose a CACD-1DCNN model for the domain adaptation of bearing fault diagnosis by combining domain alignment and discriminative feature learning. The CACD-1DCNN aims to extract domain-invariant features with improved intraclass compactness and interclass separability and guarantees high classification performance in the two domains. The experimental results on the CWRU bearing datasets confirm the superiority of the proposed method over many existing methods. Future research may focus on two aspects: (1) applying correlation alignment at multiple layers between the two domains in parallel and (2) further reducing the domain shifts in the aligned feature space through other constraints.

## Data Availability

The data used in this paper are acquired from the Bearing Data Center of Case Western Reserve University (CWRU) and web page: <http://csegroups.case.edu/bearingdatacenter/home> (accessed October 2015).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Authors' Contributions

Jing An conceived the study, participated in the research design, conducted the experiments, and wrote the paper. Ping Ai designed the methodology and reviewed the manuscript critically for important intellectual content. Dakun Liu carried out more supplementary experiments, including the calculation of Precision, Recall, and  $F$ -Measure of the comparison methods and the effectiveness verification of the proposed method. All authors read and approved the final manuscript.

## Acknowledgments

This research was supported by the "Natural Science Foundation of the Jiangsu Higher Education Institutions of China" (Grant no. 18KJB520050), "Natural Science Foundation of China" (Grant no. 51805466), and "Natural Science Foundation of Jiangsu Province" (Grant no. BK20181055).

## References

- [1] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: a promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mechanical Systems and Signal Processing*, vol. 72-73, pp. 303-315, 2016.
- [2] Z. Chen, K. Gryllias, and W. Li, "Mechanical fault diagnosis using convolutional neural networks and extreme learning machine," *Mechanical Systems and Signal Processing*, vol. 133, p. 106272, 2019.
- [3] T. Han, C. Liu, W. Yang, and D. Jiang, "A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults," *Knowledge-Based Systems*, vol. 165, pp. 474-487, 2019.
- [4] M. M. M. Islam and J.-M. Kim, "Automated bearing fault diagnosis scheme using 2D representation of wavelet packet transform and deep convolutional neural network," *Computers in Industry*, vol. 106, pp. 142-153, 2019.
- [5] H. Shao, H. Jiang, H. Zhang, W. Duan, T. Liang, and S. Wu, "Rolling bearing fault feature learning using improved convolutional deep belief network with compressed sensing," *Mechanical Systems and Signal Processing*, vol. 100, pp. 743-765, 2018.
- [6] 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.
- [7] R. Liu, Y. Shi, C. Ji, and M. Jia, "A survey of sentiment analysis based on transfer learning," *IEEE Access*, vol. 7, pp. 85401-85412, 2019.
- [8] T. Kim, M. Jeong, S. Kim, S. Choi, and C. Kim, "Diversify and match: a domain adaptive representation learning paradigm for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12456-12465, Long Beach, CA, USA, June 2019.
- [9] L. Gautheron, I. Redko, and C. Lartizien, "Feature selection for unsupervised domain adaptation using optimal transport," in *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 759-776, Dublin, Ireland, September 2018.
- [10] A.-A. Liu, S. Xiang, W.-Z. Nie, and D. Song, "End-to-end visual domain adaptation network for cross-domain 3D CPS data retrieval," *IEEE Access*, vol. 7, pp. 118630-118638, 2019.
- [11] G. Csurka, "Domain adaptation for visual applications: a comprehensive survey," 2017, <https://arxiv.org/abs/1702.05374>.
- [12] 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>.
- [13] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," 2015, <https://arxiv.org/abs/1502.02791>.
- [14] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, JMLR. org, Sydney, Australia, pp. 2208-2217, August 2017.
- [15] B. Sun and K. Saenko, "Deep coral: correlation alignment for deep domain adaptation," in *Computer Vision—ECCV 2016 Workshops*, pp. 443-450, Springer, 2016, Springer, Berlin, Germany, Lecture Notes in Computer Science.
- [16] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, AZ, USA, February 2016.
- [17] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167-7176, Honolulu, HI, USA, July 2017.
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 2672-2680, Montreal, Canada, December 2014.
- [19] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4068-4076, Santiago, Chile, December 2015.
- [20] Y. Ganin, E. Ustinova, H. Ajakan et al., "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096-2030, 2016.
- [21] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 469-477, Barcelona, Spain, December 2016.
- [22] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3722-3731, Honolulu, HI, USA, July 2017.
- [23] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction-classification networks for unsupervised

- domain adaptation,” in *Computer Vision—ECCV 2016*, pp. 597–613, Springer, Berlin, Germany, 2016.
- [24] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, “Domain separation networks,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 343–351, Barcelona, Spain, December 2016.
- [25] B. Tan, Y. Zhang, S. J. Pan, and Q. Yang, “Distant domain transfer learning,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2017.
- [26] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, “Deconvolutional networks,” in *Proceedings of the 2010 IEEE Computer Society Conference on computer vision and pattern recognition*, pp. 2528–2535, San Francisco, CA, USA, June 2010.
- [27] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, “A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals,” *Sensors*, vol. 17, no. 2, p. 425, 2017.
- [28] 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.
- [29] B. Zhang, W. Li, J. Hao, X.-L. Li, and M. Zhang, “Adversarial adaptive 1-D convolutional neural networks for bearing fault diagnosis under varying working condition,” 2018, <https://arxiv.org/abs/1805.00778>.
- [30] X. Jian, W. Li, X. Guo, and R. Wang, “Fault diagnosis of motor bearings based on a one-dimensional fusion neural network,” *Sensors*, vol. 19, no. 1, p. 122, 2019.
- [31] Z. Tong, W. Li, B. Zhang, and M. Zhang, “Bearing fault diagnosis based on domain adaptation using transferable features under different working conditions,” *Shock and Vibration*, vol. 2018, Article ID 6714520, 12 pages, 2018.
- [32] 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.
- [33] T. Han, C. Liu, W. Yang, and D. Jiang, “Deep transfer network with joint distribution adaptation: a new intelligent fault diagnosis framework for industry application,” *ISA Transactions*, vol. 97, pp. 269–181, 2019.
- [34] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1988–1996, Montreal, Canada, December 2014.
- [35] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in *Computer Vision—ECCV 2016*, pp. 499–515, Springer, Berlin, Germany, 2016.
- [36] W. Liu, Y. Wen, Z. Yu, and M. Yang, “Large-margin softmax loss for convolutional neural networks,” in *Proceedings of the 2016 International Conference on Machine Learning (ICML)*, p. 7, New York City, NY, USA, June 2016.
- [37] C. Chen, Z. Chen, B. Jiang, and X. Jin, “Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3296–3303, 2019.
- [38] J. Cavazza, A. Zunino, M. San Biagio, and V. Murino, “Kernelized covariance for action recognition,” in *Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 408–413, Cancun, Mexico, December 2016.
- [39] P. Morerio, J. Cavazza, and V. Murino, “Minimal-entropy correlation alignment for unsupervised deep domain adaptation,” 2017, <https://arxiv.org/abs/1711.10288>.
- [40] Case Western Reserve University bearing data center website, <https://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website>.
- [41] L. V. D. Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.