IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 17, 2019, accepted May 5, 2019, date of publication May 14, 2019, date of current version June 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916935

A Deep Transfer Model With Wasserstein Distance Guided Multi-Adversarial Networks for Bearing Fault Diagnosis Under Different Working Conditions

MING ZHANG[1], DUO WANG[2], WEINING LU[3], JUN YANG[2], ZHIHENG LI[1], (Member, IEEE), AND BIN LIANG[2,4]

[1]Center for Artificial Intelligence and Robotics, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China
[2]Department of Automation, Tsinghua University, Beijing 100084, China
[3]School of Aerospace Engineering, Tsinghua University, Beijing 100084, China
[4]Research Institute, Tsinghua University, Shenzhen 518057, China

Corresponding authors: Weining Lu (luweining_thu@163.com) and Jun Yang (yangjun603@tsinghua.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB0602700, and in part by the Science and Technology Research Foundation of Shenzhen under Grant JCYJ20160301100921349 and Grant JCYJ20170817152701660.

ABSTRACT In recent years, intelligent fault diagnosis technology with the deep learning algorithm has been widely used in the manufacturing industry for substituting time-consuming human analysis method to enhance the efficiency of fault diagnosis. The rolling bearing as the connection between the rotor and support is the crucial component in rotating equipment. However, the working condition of the rolling bearing is under changing with complex operation demand, which will significantly degrade the performance of the intelligent fault diagnosis method. In this paper, a new deep transfer model based on Wasserstein distance guided multi-adversarial networks (WDMAN) is proposed to address this problem. The WDMAN model exploits complex feature space structures to enable the transfer of different data distributions based on multiple domain critic networks. The essence of our method is learning the shared feature representation by minimizing the Wasserstein distance between the source domain and target domain distribution in an adversarial training way. The experiment results demonstrate that our model outperforms the state-of-the-art methods on rolling bearing fault diagnosis under different working conditions. The t-distributed stochastic neighbor embedding (t-SNE) technology is used to visualize the learned domain invariant feature and investigate the transferability behind the great performance of our proposed model.

INDEX TERMS Transfer learning, fault diagnosis, convolutional neural network, multi-adversarial networks.

I. INTRODUCTION

Deep learning has brought impressive progress to the state-of-the-art across a variety of machine learning tasks, including image classification, natural language processing, object recognition and so on [1], [2]. In the manufacturing industry, failure may introduce unplanned downtime and loss of benefits. Fault diagnosis is a technology which aims at identifying the cause of failure and preventing equipment breakdown, and it plays an important role in modern manufacturing systems. As the connection between the rotor and support, rolling bearings have been widely applied to the rotating mechanical device. To prevent long-term breakdowns or sudden catastrophic failure, it is a critical matter that identifies the rolling bearings fault at its incipient stage [3]–[5]. Recently, deep learning has been applied in the field of fault diagnosis and achieved certain success [6]. Fault diagnosis with deep learning provides a new end-to-end solution. It has been investigated for detection of faults after the occurrence of certain failure and prediction of the future working

The associate editor coordinating the review of this manuscript and approving it for publication was Malik Jahan Khan.

conditions [7]–[11]. We can believe that deep learning will have a promising future in the fault diagnosis field.

However, we find that the deep learning method works well only when enough labeled training data is available, and the training and testing data are drawn from the same distribution. When these conditions can not be satisfied, the performance of the deep learning method may decline or even become invalid. This kind of problem can be referred to as transfer learning problem, which often occurs in fault diagnosis tasks of the real industry. Concretely, under the learning process of the classification model, we can only collect fault data with a finite state as training data. However, the data of rolling bearings need to be identified usually come from different working conditions in actual fault diagnosis application. Although the fault categories of rolling bearings are the same ones, the distribution of testing data from the target domain is different from the distribution of training data from the source domain. Therefore, model learning from training data is difficult to accurately identify the testing data. Even worse, it will cost too much or even be impossible to recollect and label the data that you need for rebuilding the classification model for the actual mission.

Transfer learning method aims at solving this kind of problem. It attempts to transfer a model for the target domain by utilizing the source domain when these domains draw from different distributions. In order to effectively transfer the classifier model between different domains, many different methods have been investigated for transfer learning. The early methods are mainly conducted instance transfer, which is reweighting the source domain data based on the shared information contained in the target domain data [12], [13]. Recently, the feature mapping method has achieved great success, which projects the data from different domains to a common feature space where the feature representations are domain invariant. The shared feature can be obtained by learning the feature representation to minimize the discrepancy of the different domain, which is determined by maximum mean discrepancy (MMD) [14]–[17] or other relative distances [18], [19]. The adversarial adaptation method has been developed over the last few years, which is becoming a powerful solution for reducing the domain discrepancy by an adversarial objective with respect to a domain critic [20]–[23]. However, these methods are suffering gradients vanishing and exploding problem during the learning procedure [24]. The previous methods mainly adjust the source and target distributions based on a single domain critic, without regarding the complex multiple feature space structures underlying the data distributions. When aligning the source and target domain only with a single domain critic, it may not work well for diverse transfer situation [25].

Some adaptive methods have been studied for fault diagnosis of the rolling bearing with the different working conditions in recent years [26]–[30]. It is obvious that these methods have made some achievements, and the input signal of the model is not the raw signal but the processed features. Other methods proposed in [31] and [32] implements the end-to-end deep model for the rolling bearing fault diagnosis, which is effective for the domain transfer task. However, these models do not utilize any transfer learning algorithm. Some transfer neural network models with maximum mean discrepancy (MMD) have achieved good results on the transfer task of rolling bearing fault diagnosis [33]–[36]. Our goal is to develop an end-to-end deep transfer model with multi-adversarial learning strategy under the raw signal as input, which can be very effective for diverse transfer tasks of bearing fault diagnosis under different working conditions.

In this paper, we proposed a novel deep model, named WDMAN: Wasserstein Distance Guided Multi-Adversarial Networks. This method is inspired by generative adversarial net (GAN) [37] which is aiming at learning a generator to produce the fack image which can not distinguish from the real images. To solve the gradient problem, we replace the original GAN distance with Wasserstein distance [24]. The convolutional neural network is constructed for standard identification task due to its powerful performance of feature representation and classification. Our approach is the first attempt for solving the transfer learning problem for fault diagnosis through Multi-Adversarial Networks with Wasserstein Distance. The main contributions of this literature are summarized as follows:

1) A new WDMAN model has been proposed for transfer learning in the bearing fault diagnosis with the different working conditions. The key to our model is learning the domain invariant feature cross the source and target domain to solve the transfer problem between different data distributions. The transfer procedure is training domain critic to estimate the distribution discrepancy between the source and target domain, then the domain invariant feature can be learned through adversarial training strategy.
2) In order to improve the transfer capacity over previous methods, we present a multilayer adversarial approach in our WDMAN model, which matches the complex feature space structures to adapt different domain distributions based on multiple domain critic networks, and the Wasserstein distance is used to measure the discrepancy between source and target domain distribution for avoiding the gradients vanishing and exploding problem in the adversarial training strategy.
3) The effectiveness of WDMAN model has been verified by implementing it to the CWRU dataset under different loads and MCP dataset under different speeds and powers. The results illustrate that our model is outperformed than tradition models and other deep models. Different penalty coefficients λ are tested for WDMAN model, which indicates that our method has very good robustness.
4) The t-SNE is used to visualize the fully connected layer between the source and target domain for all the deep models including Convolutional Neural Network (CNN), Deep Domain Confusion (DDC) [15], Deep Adaptation Networks (DAN) [16], Joint Adaptation

Networks (JAN) [19], and our WDMAN. The visualization results demonstrate the transferability for all deep models and verify the effectiveness of WDMAN model in promoting the domain transfer capability.

The rest of paper is organized as follows. In section II, the problem definition and assumption, preliminaries of CNN and WGAN are described. Section III details the proposed method. A series of discussion and analysis for the experiment are conducted in Section IV. Finally, conclusions are made in Section V.

II. PRELIMINARIES

A. PROBLEM DEFINITION AND ASSUMPTION

For the transfer learning problem, it is assumed that there is an labeled source dataset $X_s = \{(x_i^s, y_i^s)\}_{i=1}^{n^s}$ drawn from the source domain D_s which is enough to train a source domain distribution model, as well as an target dataset $X_t = \{x_j^t\}_{j=1}^{n^t}$ drawn from the target domain D_t, where there is no label apparently. The general goal of the problem is to learn an transfer classifier model $\eta : X \to Y$ which has a low target risk $R_{D_T}(\eta) = \underset{(x,y)\sim D_T}{Pr}(\eta(x) \neq y)$.

For the traditional fault diagnosis problem, the testing and training data are drawn from the same distribution. Therefore, the most critical matter is to learn an identification model from the training data can generalize well to the testing data. However, our concern is about the training and testing data from different distributions, which results in the fact that the training model can not directly classify the testing data. To solve this challenge of transfer learning, many researchers propose their methods concentrate on learning domain invariant representation by minimizing the discrepancy between the source and target domain. In this paper, transfer learning problem for rolling bearing fault diagnosis has been investigated by using adversarial transfer technique. Our goal is to learn a model that can accurately identify data from the target domain which has no labels in it.

B. CONVOLUTIONAL NEURAL NETWORK

Recently, the convolutional neural network (CNN) is extraordinary famous and have been tremendously successful in practical applications, particularly in the field of image classification [38]–[40]. The deep convolutional neural network is a kind of neural network structures and consists of hierarchically arranged trainable layers learning the efficient feature representation [1].

The typical convolutional layer for image classification contains the input image I and the kernel K, the two-dimensional convolution is defined as follows [41]:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \qquad (1)$$

Since the data to be processed is a one-dimensional vibration signal in this paper, we apply the one-dimensional convolution in each convolutional layer. From the above, the one-dimensional convolution could be obtained easily when m is equal to 1. Specifically, the one-dimensional convolution could be calculated by the following equation:

$$C_{ij}^l = \phi(k_{n\times 1}^j * x_{i:i+n}^i + b_{ij}) \qquad (2)$$

$k_{n\times 1}^j$is the jth kernel which belong to the kernels K_j^l size $n \times 1 \times j$ of the l-th convolution layer; $x_{i:i+n}^i$ is the ith input segment; b_{ij} is corresponding to the bias; ϕ is the activation function which can be Sigmoid, Tanh, Relu and leakyRelu; C_{ij}^l is the i-th feature point of the j-th kernel in the l-th convolution layer.

For the classification problem, some fully connected layers should be connected to the last convolution layer response to output classification result. The output of the fully connected is defined as follows:

$$y^l = \phi(W^l y^{l-1} + b^l) \qquad (3)$$

where W^l is the weight matrix between the upper layer and the current layer; y^{l-1} is the feature map of the upper layer; b^l is the bias for the current layer. All the parameters are updated by using the backpropagation method [42] with the objective to minimize the error between the actual output $\hat{y}$ and the desired output y. The loss function for CNN classification network is expressed as follows:

$$L = \frac{1}{2n}\|y - \hat{y}\|_F^2 \qquad (4)$$

where F is Frobenius norm.

C. WASSERSTEIN GAN

Generative Adversarial Networks (GAN) [37] are a powerful class of generative models that cast generative modeling as a game between two networks: a generator network produces synthetic data given some noise source and a discriminator network discriminates between the generator's output and true data. Formally, the game between the generator (G) and the discriminator (D) is the minimax objective:

$$\underset{G}{min}\,\underset{D}{max}\,\underset{x\sim P_r}{E}[log(D(x))] + \underset{\tilde{x}\sim P_g}{E}[log(1 - D(\tilde{x}))] \qquad (5)$$

where P_r is the real data distribution and P_g is the data distribution from generative model which defined by $\tilde{x} = G(z)$, $z \sim p(z)$. z is the sampled data from certain distribution, such as uniform distribution or Gaussian distribution. If the discriminator is trained to optimality before each generator parameter update, then minimizing the value function amounts to minimizing the Jensen-Shannon divergence between P_d and P_g [37], but doing so often leads to vanishing gradients as the discriminator saturates [24].

Arjovsky *et al.* [24] argues that the Jensen-Shannon divergence between P_d and P_g which GAN typically minimize often leads to vanishing gradients as the discriminator saturates since the divergences are potentially not continuous with respect to the generator's parameters. Therefore, they propose using the Wasserstein distance $W(q, p)$ instead of typical GAN divergence, which is informally defined as the minimum cost of transforming the distribution q into

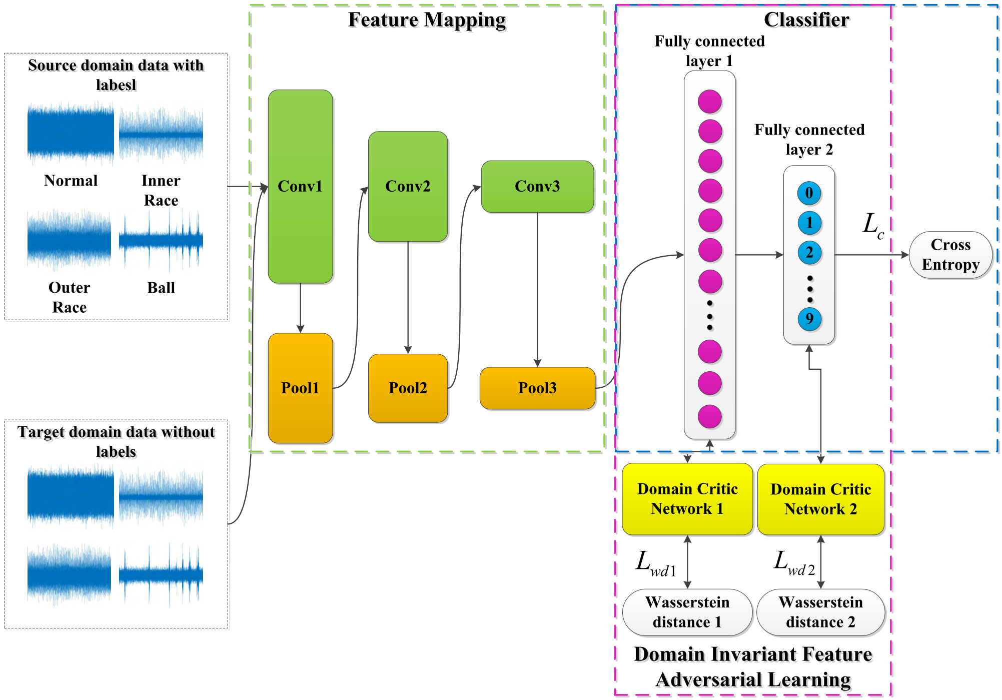


FIGURE 1. **An architechture of wasserstein distance guided multi-adversarial networks (WDMAN) model.**

the distribution p. The advantage of Wasserstein distance $W(q, p)$ is that it is continuous and differentiable almost everywhere. The WGAN objective function is constructed using the Kantorovich-Rubinstein duality [43] to obtain:

$$\underset{G}{min}\ \underset{D}{max}\ \underset{x\sim P_r}{E}[D(x)] - \underset{\tilde{x}\sim P_g}{E}[D(\tilde{x})] \tag{6}$$

where D is the set of 1-Lipschitz function and P_g is also the generative model distribution which defined by $\tilde{x} = G(z)$, $z \sim p(z)$.

To enforce the Lipschitz constraint on the domain discriminator, WGAN must clip the weights of the discriminator into a compact space $[-c, c]$, which will lead to optimization difficulties. To solve the optimization problem, Gulrajani *et al.* [44] propose an alternative way to improve the training of WGAN, which add a gradient penalty (GP) item to the original WGAN objective function, called WGAN-GP. The objective function of WGAN-GP is defined as follows:

$$\underset{G}{min}\ \underset{D}{max}\ \underbrace{\underset{x\sim P_r}{E}[D(x)] - \underset{\tilde{x}\sim P_g}{E}[D(\tilde{x})]}_{WD} - \lambda \underbrace{\underset{\hat{x}\sim P_{\hat{x}}}{E}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2]}_{GP} \tag{7}$$

$P_{\hat{x}}$ is the sampling uniformly along straight lines between pairs of points sampled from the real data distribution P_r and the generator distribution P_g; λ is the penalty coefficient. The results in [44] presents that WGAN-GP is a more reasonable method which avoids gradients vanishing and exploding. With the enforced gradient penalty term, adversarial networks would have stronger robustness and more complicated networks could be trained easily.

III. PROPOSED METHOD

In this section, we propose a novel WDMAN model to solve the transfer learning problem in rolling bearing fault diagnosis under different working conditions, called WDMAN: Wasserstein Distance guided Multi-Adversarial Networks. The architecture of our WDMAN model is presented in Figure 1. There are three components in our WDMAN model. Feature mapping M consist of many convolutional and Pooling layers aim at extracting essential feature representation, and fully connected layers construct the classifier C. The critical part is domain invariant feature adversarial learning which is made of multiple domain critic networks D.

A. WASSERSTEIN METRIC

Before introducing our approach, we start with a brief introduction to the Wasserstein metric which is investigated for the optimal transport problem to measure the cost form any different location. $\pi(x, y)$ is the transference policy from location x to location y. The Wasserstein metric is a distance measure between probability distributions on a given Polish metric space (M, ρ) and let $p \in [1, \infty)$. For any two probability measure $\mathbb{P}$ and $\mathbb{Q}$ on M, the Wasserstein distance of order p is defined as follows:

$$W_p(\mathbb{P}, \mathbb{Q}) = (\underset{\pi\in\prod(\mathbb{P},\mathbb{Q})}{inf} \int_M \rho(x, y)^p d\pi(x, y))^{\frac{1}{p}} \tag{8}$$

where $\rho(x, y)$ is a distance function; x and y is the samples from set M; $\prod(\mathbb{P}, \mathbb{Q})$ is the set of all probability measures on $M \times M$ with marginals $\mathbb{P}$ and $\mathbb{Q}$. Let $(\mu_k)_{k\in\mathbb{N}}$ be a sequence of probality samples in the Wasserstein space $P_p(M)$ and let μ be another sample of $P_p(M)$, then μ_k converges weakly in $P_p(M)$ to μ when the Wasserstein distance is close to zero $W_p(\mu_k, \mu) \to 0$.

The most useful exponents in the Wasserstein distances are $p = 1$ and $p = 2$. Comparing with the W_2 distance, the W_1 distance is more flexible and easier to bound and it has strong implications in functional analysis. The Kantorovich-Rubinstein duality [43] tell us that, the W_1 distance (Earth-Mover distance) could be expressed as follows:

$$W(\mathbb{P}, \mathbb{Q}) = \sup_{\|f\|\leq 1} \mathbb{E}_{x\sim\mathbb{P}}[f(x)] - \mathbb{E}_{x\sim\mathbb{Q}}[f(x)] \tag{9}$$

where the Lipschitz semi-norm is $\|f\| = \sup|f(x) - f(y)|/\rho(x, y)$. In this paper, we follow the suggestion in [24] and use W_1 distance to guide adversarial networks training procedure, since the Wasserstein distance own much better property than the typical GAN distance (JS divergence).

B. DOMAIN INVARIANT FEATURE LEARNING

The transfer learning problem in rolling bearings fault diagnosis primarily caused by the fact that source and target domain data draw from different distribution due to the varying conditions. Therefore, model learning from the labeled source domain data is hard to classify the data of the target domain. The highly bias between two different distributions may be an essential challenge for this situation. We propose an approach to obtain the invariant domain feature of different distributions to figure out this challenge. The learning process is to minimize the Wasserstein distance between source and target domain by utilizing the adversarial training strategy mimic GAN [37].

The convolutional neural network is applied for implementing feature mapper and domain critic in our invariant feature learning procedure. Feature mapper aims at acquiring the invariant feature between the source and target domain, which means that the discrepancy from both domains should be close to zero. Meanwhile, the domain critic proposed in [24] is supposed to estimate the Wasserstein distance between the source and target feature distribution, use for discriminating discrepancy in our approach. The Wasserstein distance between source feature distribution $\mathbb{P}_{f_s}$ and target feature distribution $\mathbb{P}_{f_t}$, where $f_s = F(x^s)$ and $f_t = F(x^t)$, can be approximated by maximizing the domain critic (D) loss L_{wd} with the respect to parameter θ_d:

$$L_{wd}(x^s, x^t) = \frac{1}{n^s} \sum_{x^s\in X^s} D(F(x^s)) - \frac{1}{n^t} \sum_{x^t\in X^t} D(F(x^t)) \tag{10}$$

where x^s and x^t are the data samples draw from source domain X^s and target domain X^t, respectively. In order to satisfy the Lipschitz constraint condition of Wasserstein distance, Gulrajani et al [44] propose a more rational method which enforces the constraint with a penalty $L_{gp}(\tilde{x})$ on the gradient norm for the domain critic parameter θ_d

$$L_{gp}(\tilde{x}) = (\| \nabla_{\tilde{x}\in\mathbb{P}_{\tilde{x}}} D(\tilde{x}) \|_2 - 1)^2 \tag{11}$$

where $\tilde{x}$ is the random samples from $\mathbb{P}_{\tilde{x}}$, which is sampling uniformly along straight lines between pairs of points sampled from the source and target feature distribution.

The domain invariant feature can be obtained through adversarial training strategy. There are two steps in this process, we first train the domain critic to maximize the Wasserstein distance on both domains, and then fix the parameter of domain critic to minimize the Wasserstein distance by tuning the feature mapper with respect to parameter θ_f. Thus the domain invariant feature learning strategy can be expressed as follows:

$$\underset{\theta_f}{min}\underset{\theta_d}{max}\{L_{wd} - \lambda L_{gp}\} \tag{12}$$

where λ is the penalty coefficient. Through the iterative learning algorithm, we can consider the feature mapper will own domain invariant feature when the Wasserstein distance converges to zero.

C. TRANSFER MODEL ARCHITECTURE

The purpose of our transfer model is trying to address the classification problem of target domain without labels. Our proposed model uses adversarial learning method with Wasserstein distance to achieve domain invariant feature between the source and target domain without labels. More specifically, we train the model for the source domain with labels by using the supervised learning method and then transfer the model to adapt the target domain without labels by adversarial learning the invariant feature between the source and target domain. The transfer adaption procedure in our method only requires source and target domain data, no labels needed, which mean the transfer process is under an unsupervised learning condition.

The objective of classification model for source domain data is to train the feature mapping M with parameter θ_M and the classifier C with parameter θ_c. The loss L_c is defined as the cross-entropy between the Softmax predicted probabilistic distribution and the one-hot encoding of the labels of the source domain data samplings:

$$L_c(x^s, y^s) = -\frac{1}{n^s} \sum_{i=1}^{n^s} \sum_{k=1}^{K} l(y_i^s = k) \cdot logC(M(x_i^s))_k \tag{13}$$

where $l(y_i^s = k)$ is the indicator function; $C(M(x_i^s))_k$ is the $k - th$ dimension value of the predicted distribution; K is the number of categories.

As mention above, the domain invariant feature can be learned through adversarial training strategy guided by Wasserstein distance. Our WDMAN model transfer the classifier C from the source to target domain by obtaining the domain invariant feature of fully connected layers Fc_j, where Fc_j is the j-th fully connected layer in classifier C. In order to minimize the Wasserstein distance between source and target feature distribution, we use multiple domain critic networks D_j to estimate the distribution discrepancy for fully connected layers Fc_j, respectively. During the transfer process, the domain critic networks D_j are optimized by maximizing the domain adversarial loss L_{adv_j} with the respect to parameter θ_{d_j}, the invariant feature is learning from fully connected layers Fc_j by minimizing the domain adversarial loss L_{adv_j} and classification loss L_c with the respect to parameter θ_{fc_j}. The domain adversarial loss L_{adv_j} of our model is defined as follows:

$$L_{adv_j}(x^s, x^t) = \frac{1}{n^s}\sum_{i=1}^{n^s} D(F_j(M(x_i^s))) - \frac{1}{n^t}\sum_{i=1}^{n^t} D(F_j(M(x_i^t))) - \lambda(\|\nabla_{\hat{x}} D(\tilde{x})\| - 1)^2 \tag{14}$$

where,

$$F_j = \begin{cases} Fc_1 & j = 1 \\ Fc_j(F_{j-1}) & j > 1; \end{cases} \tag{15}$$

$(\|\nabla_{\tilde{x}} D(\hat{x})\| - 1)^2$ is the gradient penalty of this optimization problem, and it can control the training process without gradient vanishing and exploding problems; λ is the penalty coefficient. The domain adversarial loss function is used to guide the distribution discrepancy reducing progressively, and the purpose of increasing the classification loss item is to ensure the classification effect.

D. ALGORITHM AND TRAINING STRATEGY

We present the algorithm and training strategy of the WDMAN model in this part. The algorithm of our method is summarized in Algorithm 1, and it can be trained by the standard back-propagation. Firstly, the deep network architecture is determined. In our work, the deep neural network is consist of three layers of one-dimensional convolution and two layers of fully connected. Then, we train the feature mapping M and classifier C in the deep model architecture with the labeled source domain data, the parameters θ_M and θ_c are updated by using the loss function in (13). In the transfer procedure of adversarial learning, the domain critic networks with relevant parameter θ_{d_j} is updated by maximizing the adversarial loss function in (14) and the parameter θ_{fc_j} in the fully connected layer is trained by minimizing the sum of loss functions in (13) and (14). The domain invariant feature is achieved in the fully connected layers until the end of the training process. The workflow of WDMAN model in detail shown in Figure 2

IV. EXPERIMENTS

In this section, we evaluate the efficacy of our approach on the CWRU rolling bearings dataset under different loads and MCP rolling bearings dataset under different rotating speeds and powers, the testbeds of CWRU and MCP are shown in Figure 3.

Algorithm 1 Wasserstein Distance Guilded Multi-Adversarial Network Learning Procedure

```
Require: source data $X^s$, target data $X^t$, mini-batch size $m$,
    feature mapping and classier training step $n_c$, transfer
    procedure train step $n_t$, number of fully connected layers
    in classifier $n_l$, domain critic networks training step $n_d$,
    learning rate $\alpha$.
Initialize the parameters of classifier $\theta_M$ and $\theta_c$.
for $t = 1, \ldots, n_c$ do
    Sample mini-batch $\{x_i^s, y_i^s\}_{i=1}^m$ from $X^s$
    $\theta_M \leftarrow \theta_M - \alpha\nabla_{\theta_M} L_c(x^s, y^s)$
    $\theta_c \leftarrow \theta_c - \alpha\nabla_{\theta_c} L_c(x^s, y^s)$
end for
Initialize the parameter of discriminator $\theta_{d_j}$
for $i = 1, \ldots, n_t$ do
    Sample mini-batch $\{x_i^s, y_i^s\}_{i=1}^m$, $\{x_i^t\}_{i=1}^m$ from source
    data $X^s$ and target data $X^t$.
    for $j = 1, \ldots, n_l$ do
        for $k = 1, \ldots, n_d$ do
            a random number $\varepsilon \sim U[0, 1]$
            $\tilde{x} \leftarrow \varepsilon F_j(M(x^s)) + (1 - \varepsilon)F_j(M(x^t))$
            $\theta_{d_j} \leftarrow \theta_{d_j} - \alpha\nabla_{\theta_{d_j}} L_{adv_j}(x^s, x^t)$
        end for
        $\theta_{fc_j} \leftarrow \theta_{fc_j} - \alpha\nabla_{\theta_{fc_j}}[L_{adv_j}(x^s, x^t) + L_c(x^s, y^s)]$
    end for
end for
```

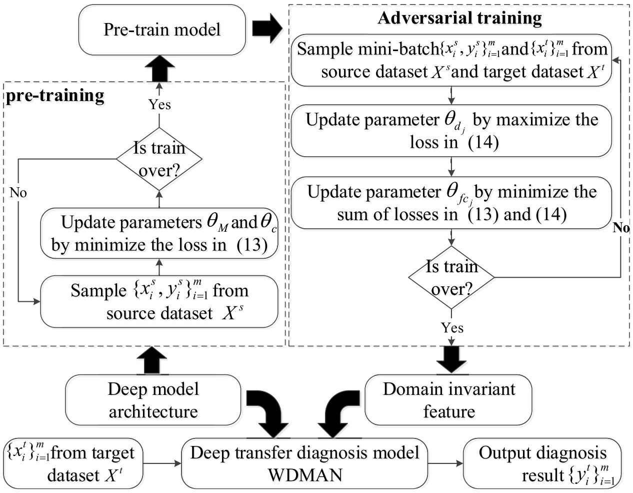

Pre-train model
Adversarial training
Sample mini-batch$\{x_i^s, y_i^s\}_{i=1}^m$and$\{x_i^t\}_{i=1}^m$ from source dataset $X^s$and target dataset X^t
pre-training
Yes
Is train over?
No
Update parameter θ_{d_j} by maximize the loss in (14)
Update parameter θ_{fc_j} by minimize the sum of losses in (13) and (14)
No
Update parameters θ_M and θ_c by minimize the loss in (13)
Is train over?
Sample $\{x_i^s, y_i^s\}_{i=1}^m$ from source dataset X^s
Yes
Deep model architecture
Domain invariant feature
$\{x_i^t\}_{i=1}^m$from target dataset X^t
Deep transfer diagnosis model WDMAN
Output diagnosis result $\{y_i^t\}_{i=1}^m$

FIGURE 2. Workflow of WDMAN model.

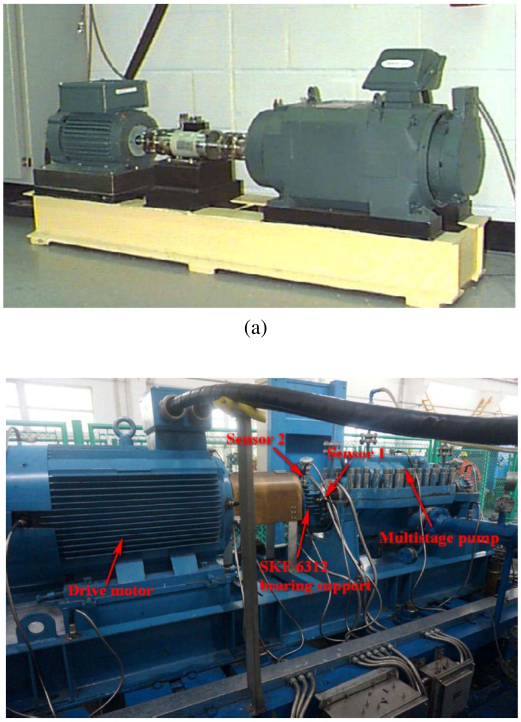
A. COMPARED METHODS

In order to verify the effectiveness of our WDMAN model, we mainly compare our proposed approach with multiclass Support Vector Machines (SVM) [45], Transfer Component Analysis (TCA) [14], Deep Domain Confusion (DDC) [15], Deep Adaptation Networks (DAN) [16], and Joint Adaptation Networks (JAN) [19].

For the traditional model SVM and TCA, we use four kinds of input samples, which are the raw signal, frequency spectrum, envelope signal, and envelope frequency spectrum,

TABLE 1. Detail of the feature mapping and classifier networks used in the experiments.

	Layer type	Kernal	stride	Channel	Padding	Activation function
Feature Mapping	Convolution 1	5×1	2×1	16	Yes	Relu
	Pooling 1	2×1	2×1	16	No	
	Convolution 2	3×1	2×1	32	Yes	Relu
	Pooling 2	2×1	2×1	32	No	
	Convolution 3	3×1	2×1	64	Yes	Relu
	Pooling 3	2×1	2×1	64	No	
Classifier	Fully connected 1	512		1		Tanh
	Fully connected 2	10 or 4		1		Softmax



(a)

(b)

FIGURE 3. (a) CWRU testbed; (b) MCP testbed.

respectively. The latter three kinds of input samples are the typical feature extraction methods for rolling bearing fault diagnosis [5]. For the deep model CNN, DDC, DAN, JAN, and WDMAN, only the raw signal as the input samples.

As a benchmark for the comparison of deep models, the CNN model trained by using the labeled source data, then directly test the target data. DDC, DAN, and JAN are domain adaptive frameworks based on CNN and learn the invariant representation by using MMD, MK-MMD, and JMMD to compute the domain transfer loss, respectively. Our model is a Wasserstein distance guided adversarial transfer approach based on CNN, which aims at minimizing the Wasserstein distance between different distribution to achieve domain invariant feature.

B. IMPLEMENTATION DETAILS

In order to ensure the fairness of comparison, we follow the evaluation criteria proposed in [18] and evaluate all compared methods by searching the parameter space for the optimal parameter settings, and report the best results of each model.

The classification model of the traditional method is trained by using the labeled source samples, and then the labels of target samples are directly predicted. Additionally, TCA takes advantage of data from both source and target domain to learn some transfer components across domains before turning into the classifier. We use the SVM package of scikit-learn to implement SVM [45], which is realized with the Gaussian kernel and tradeoff parameter is set to 1 for our work. Taking the advice in literature [14], TCA model utilizes a KNN classifier trained with labeled source samples to classify the target samples.

All deep models are implemented by using TensorFlow framework. We use a convolutional neural network (CNN) as the basic framework for classification. The detail of the CNN model architecture is shown in Table 1. The same network architecture is used for all deep model approaches in the experiment. For the DDC method, we follow the suggestions in [15] and use the MMD to reduce the distance of a fully connected layer in the CNN classification model. The parameters of DAN and JAN are respectively set according to the literature [16] and [19]. The MK-MMD of DAN and JMMD of JAN are utilized to minimize the distribution discrepancy of the fully connected layers in the classification model. For each approach, the transfer loss term is added to the classification loss with a coefficient for the trade-off. Our proposed WDMAN model can be implemented based on the algorithm procedures mentioned in Algorithm 1. The detail of domain critic networks is presented in Table 2. There are two fully connected layers in the classifier $n_l = 2$.

In our work, we first use classification accuracy on test data as the evaluation metric, which is widely used in many literatures [16]–[20].

$$Accuracy = \frac{|x : x \in D_t \wedge \hat{y}(x) = y(x)|}{|x : x \in D_t|} \tag{16}$$

where D_t is the set of test data; $y(x)$ is the label of x; $\hat{y}(x)$ is the predicted labels. Then, we introduce several performance metric tools, including precision, recall, ROC, and AUC which are widely used in machine learning, to evaluate our WDMAN and compare with other methods. For the classification problem, the combination of the actual and predicted categories in the samples can be divided into true positive (TP), false positive (FP), true negative (TN), false

TABLE 2. Detail of the domain critic networks used in the experiments.

	Layer type	Kernal	stride	Channel	Padding	Activation function
Domain critic 1	Convolution 1	5 × 1	2 × 1	64	Yes	leakyRelu
	Fully connected 1	512		1		leakyRelu
	Fully connected 2	1		1		linear
Domain critic 2	Convolution 1	5 × 1	2 × 1	32	Yes	leakyRelu
	Fully connected 1	512		1		leakyRelu
	Fully connected 2	1		1		linear

TABLE 3. Confusion matrix.

		Predicted	
		Positive	Negative
Actural	True	True Positive (TP)	False Negative (FN)
	False	False Positive (FP)	True Negative (TN)

negative (FN). The confusion matrix of classification results is shown in Table 3.

For our fault diagnosis problem, the precision and recall of each failure category f can be calculated as follows:

$$precision = \frac{TP}{TP + FP} \tag{17}$$

$$recall = \frac{TP}{TP + FN} \tag{18}$$

where TP is the number of the current fault which belongs to f correctly classified in f; FP is the number of the current fault which does not belong to f incorrectly classified in f; FN is the number of the current fault which belongs to f incorrectly classified in other categories. Receiver operating characteristic (ROC) curve is a powerful tool to measure the performance of models, which is a graph with Y-axis is true positive rate (TPR) and X-axis is false positive rate (FPR). The TPR and FPR can be calculated as follows:

$$TPR = \frac{TP}{TP + FN} \tag{19}$$

$$FPR = \frac{FP}{FP + TN} \tag{20}$$

where the TP, FN, FP, and TN are from the confusion matrix in Table 3. AUC is the area under the ROC curve to quantify the performance of the model.

C. CASE 1: RESULTS AND ANALYSIS ON CWRU DATASET

In this experiment, the performance of different methods has been evaluated with CWRU dataset under different loads. Since the sampling frequency is 12kHz and rotating speed is 1750 rpm, the feature-length of each sample in the dataset is equal 1200, which also equals three rotation period. All deep models are trained by using Adam method with hyperparameters batch size $m = 64$, learning rate is $\alpha = 0.0001$, and 20000 generator iterations. The hyperparameters as recommended in [44] ($\alpha = 0.0001$, $n_d = 5$, $\lambda = 10$) for our WDMAN model. The pre-train iterations $n_c = 1000$ and adversarial iterations $n_t = 20000$.

1) DATASET AND PREPROCESSING

The first dataset was acquired from a test stand built by the bearing data center of Case Western Reserve University (CWRU) [46]. The testbed is composed of a motor, a torque transducer, and a dynamometer. Vibration data was collected using accelerometers which were attached to the motor housing. The dataset consists of normal and faulty data. The faulty dataset was produced with defects in the inner race, the outer race and ball with different sizes (0.007, 0.014 and 0,021 in.) by using electro-discharge machining (EDM). The data were collected at different motor loads (0, 1, 2, and 3 hp) with the sampling frequency of 12 kHz. The samples drawn from four different working loads are called as domain A, B, C, and D, respectively. There are ten categories for each domain, which consist of nine kinds of defects and a normal condition, 500 samples are assigned in each category. The detail is shown in Table 4.

2) ACCURACY ACROSS DIFFERENT DOMAINS

As shown in Table 5, SVM, F-SVM, E-SVM, and EF-SVM have poor performance for domain transfer problem, whose accuracy average around 20%, 70%, 35%, and 60%, respectively. This results confirm that the classification model trained from the labeled source domain samples is hardly applied to the target domain samples due to that the different working loads make the data distribution of rolling bearing change. The average accuracy of TCA, F-TCA, E-TCA, and EF-TCA are about 30%, 75%, 70%, and 90%, respectively. These results of TCA are better than the results of SVM which indicates that the transfer method in TCA makes an effect. From all the results of the traditional model, these methods are greatly influenced by the transfer problem, the model with the envelope frequency spectrum is the best, and the raw signal is the hardest to transfer. Therefore, it is quite a challenge to implement model transfer different domains, when the input of the model is the raw signal of the rolling bearing.

For the deep models, the raw signal is treated as the only input to test the performance of all the deep transfer methods, the accuracies are presented in Table 6. As the benchmark of the deep model, the results of CNN verify that there must be a certain effect on data distribution between the source and target domain with the change of working loads. So, the CNN model trained by the labeled source samples does not work well on the unlabeled target samples. According to the average accuracies, we can find that DDC limited promote

TABLE 4. Description of CWRU dataset.

Fault location	None	Inner Race			Outer Race			Ball			Load
Fault Diameter(in.)	0	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	
Dataset A no.	500	500	500	500	500	500	500	500	500	500	0
Dataset B no.	500	500	500	500	500	500	500	500	500	500	1
Dataset C no.	500	500	500	500	500	500	500	500	500	500	2
Dataset D no.	500	500	500	500	500	500	500	500	500	500	3

TABLE 5. Tradition models accuracy (%) for CWRU dataset.

	SVM	F-SVM	E-SVM	EF-SVM	TCA	F-TCA	E-TCA	EF-TCA
A→B	22.87%	67.53%	39.00%	70.27%	31.80%	80.07%	74.20%	**95.33%**
A→C	20.73%	69.13%	35.47%	58.73%	27.27%	70.20%	70.93%	**92.20%**
A→D	23.27%	73.67%	37.40%	38.40%	36.20%	**87.47%**	27.26%	83.87%
B→A	15.60%	75.67%	32.67%	64.87%	25.00%	75.07%	78.60%	**92.27%**
B→C	20.20%	75.07%	37.53%	64.40%	78.07%	76.60%	76.33%	**96.40%**
B→D	22.20%	76.93%	34.20%	42.20%	45.73%	79.40%	67.27%	**79.67%**
C→A	15.80%	69.93%	30.67%	70.60%	34.00%	63.20%	73.27%	**92.00%**
C→B	23.20%	68.80%	39.20%	73.00%	27.27%	66.20%	74.73%	**96.33%**
C→D	23.53%	68.47%	33.53%	73.13%	25.80%	68.47%	72.80%	**94.87%**
D→A	12.47%	74.33%	32.80%	52.40%	29.80%	**82.67%**	43.87%	78.20%
D→B	23.60%	68.07%	36.47%	53.13%	34.47%	72.27%	70.93%	**84.47%**
D→C	21.87%	66.40%	36.13%	77.20%	25.87%	68.27%	74.53%	**96.13%**
AVG	20.45%	71.17%	35.42%	61.53%	30.94%	74.16%	67.09%	**90.15%**

TABLE 6. Deep models accuracy (%) for CWRU dataset.

	CNN	DDC	DAN	JAN	WDMAN
A→B	87.93%	91.53%	98.93%	95.60%	**99.73%**
A→C	89.00%	89.00%	99.00%	94.33%	**99.67%**
A→D	80.73%	83.73%	96.47%	89.93%	**100.00%**
B→A	97.47%	96.93%	98.53%	97.33%	**99.13%**
B→C	99.40%	99.53%	99.80%	**100.00%**	**100.00%**
B→D	89.00%	98.40%	98.73%	**100.00%**	99.93%
C→A	97.00%	97.60%	97.53%	95.60%	**98.53%**
C→B	97.20%	97.67%	98.13%	98.60%	**99.80%**
C→D	89.53%	98.00%	99.40%	**100.00%**	**100.00%**
D→A	90.20%	92.93%	93.53%	93.73%	**98.07%**
D→B	75.53%	88.13%	96.27%	**98.40%**	98.27%
D→C	79.26%	89.00%	99.07%	**100.00%**	99.53%
AVG	89.35%	93.54%	97.95%	96.96%	**99.39%**

the transfer accuracy, while DAN, JAN, and our WDMAN get a great improvement. However, DAN and JAN don't work very well on each domain transfer task, the performance of $D \rightarrow A$ in DAN and JAN, $A \rightarrow C$ and $A \rightarrow D$ in JAN display poorly. As we can see in all the results, our WDMAN outperforms other compared methods, and it performs great in all transfer tasks. Table 7 shows the accuracies of each category in $A \rightarrow B$, and it can be clearly observed that the category of defect on the ball is the hardest to classify.

3) PERFORMANCE EVALUATION TEST

The precision and recall of each category in all domain transfer tasks calculated by our WDMAN are shown in Table 8 and Table 9. In Table 8, 70% precisions are equal to 100%, which means that each sample belonging to f accurately classify into f. Two categories of ball fault with size 0.007 inch have inferior precision in the transfer task $D \rightarrow A$ and $D \rightarrow B$, which are 88.00% and 86.52%. These mean that about 10% fault alarms of category 7 are unreliable. In Table 9, 80% recalls are equal to 100%, which means there is no missing alarm, all samples belonging to f accurately classify into f. In the ball fault with 0.021 inch, the recall of four categories under 90%, half of them close to 80%, which indicates that lots of failures aren't detected in category 9 of $C \rightarrow A$, $C \rightarrow B$, $D \rightarrow A$, and $D \rightarrow B$.

The sensitivity (SN) and specificity (SP) are used when detecting the effect of the method. ROC curve is the tool can objectively reflect sensitivity and specificity, simultaneously. In order to further compare the performance of all the models, we take $D \rightarrow B$ as an example, the ROC curves and their AUC are presented in Figure 4. For a classification model, we always want to get high TPR (means SN) and low FPR (equal to 1-SP). Reflecting in the ROC curve, the curve is closer to the upper left corner, the model will show better performance. It is obvious that JAN and WDMAN are the two best models compare with other models in Figure 4, whose AUCs are equal to 99% and 98%, respectively. These results close to 100% illustrate that there almost no false alarm and no miss detection for JAN and WDMAN. From the accuracy in the previous section, DDC and DAN benefit the accuracy compare with CNN. Meanwhile, we can find that many false alarms have been put into these two models based on the ROC curve in Figure 4(b).

4) PARAMETER ANALYSIS

We study the influence of penalty coefficient λ on our WDMAN model in this part. The penalty coefficient λ is the balance factor between domain critic loss L_{wd} and penalty term L_{gp}. CWRU rolling bearing dataset is still chosen for analyzing the effect of different λ. To quantified analysis this proposed problem, we calculate the accuracies of all domain

TABLE 7. Accuracy (%) of each category in transfer task $A \rightarrow B$ for CWRU dataset.

Fault location	None	Inner Race			Outer Race			Ball		
Fault Diameter(in.)	0	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
SVM	78.57%	9.54%	0.0%	8.53%	2.33%	7.25%	28.40%	4.81%	57.14%	1.47%
F-SVM	100.00%	0.0%	100.00%	71.20%	13.33%	100.00%	100.00%	100.00%	100.00%	1.47%
H-SVM	100.00%	99.29%	8.0%	4.33%	44.33%	78.26%	94.14%	58.61%	19.73%	17.91%
HF-SVM	100.00%	100.00%	91.67%	100.00%	100.00%	100.00%	100.00%	6.45%	31.06%	42.85%
TCA	100.00%	23.19%	16.32%	23.97%	8.33%	8.74%	58.90%	25.81%	47.58%	5.15%
F-TCA	100.00%	28.66%	100.00%	98.63%	55.33%	100.00%	100.00%	95.91%	99.38%	22.79%
H-TCA	55.19%	100.00%	98.67%	93.74%	100.00%	77.36%	100.00%	15.48%	43.48%	60.29%
HF-TCA	100.00%	100.00%	100.00%	100.00%	100.00%	90.58%	100.00%	97.42%	81.48%	83.82%
CNN	100.00%	100.00%	0.0%	100.00%	100.00%	100.00%	100.00%	100.00%	97.89%	81.38%
DDC	100.00%	100.00%	43.33%	99.32%	100.00%	99.38%	99.42%	100.00%	91.93%	81.88%
DAN	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	98.84%	90.44%
JAN	100.00%	100.00%	94.42%	100.00%	100.00%	100.00%	100.00%	92.90%	81.27%	87.44%
WDMAN	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	98.81%	100.00%	97.89%

TABLE 8. Precision (%) of the proposed WDMAN for CWRU dataset.

Fault location	None	Inner Race			Outer Race			Ball		
Fault Diameter(in.)	0	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
A→B	100.00%	100.00%	100.00%	100.%	100.00%	100.00%	100.00%	99.36%	99.38%	100.00%
A→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
A→D	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→A	100.00%	100.00%	100.00%	100.00%	99.34%	100.00%	99.32%	96.23%	99.37%	98.48%
B→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→D	100.00%	100.00%	99.34%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
C→A	100.00%	100.00%	99.34%	100.00%	97.40%	100.00%	100.00%	91.62%	100.00%	96.80%
C→B	100.00%	99.39%	100.00%	100.00%	93.17%	100.00%	99.32%	92.81%	97.55%	100.00%
C→D	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
D→A	100.00%	100.00%	100.00%	100.00%	96.77%	100.00%	99.32%	88.00%	98.03%	98.33%
D→B	100.00%	99.39%	100.00%	100.00%	99.34%	98.57%	99.32%	86.52%	99.36%	99.12%
D→C	100.00%	100.00%	100.00%	100.00%	100.00%	95.17%	98.65%	97.48%	100.00%	99.23%

TABLE 9. Recall (%) of the proposed WDMAN for CWRU dataset.

Fault location	None	Inner Race			Outer Race			Ball		
Fault Diameter(in.)	0	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
A→B	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.38%	99.26%
A→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
A→D	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→A	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	98.71%	98.14%	95.59%
B→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→D	100.00%	100.00%	100.00%	100.00%	100.00%	99.28%	100.00%	100.00%	100.00%	100.00%
C→A	100.00%	100.00%	100.00%	100.00%	100.00%	99.28%	100.00%	98.71%	97.52%	88.97%
C→B	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.32%	100.00%	98.76%	80.88%
C→D	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
D→A	100.00%	100.00%	100.00%	100.00%	100.00%	99.28%	100.00%	99.35%	92.55%	86.76%
D→B	100.00%	100.00%	99.33%	100.00%	100.00%	100.00%	99.32%	99.35%	96.89%	83.09%
D→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	95.65%	94.85%

transfer problem under different penalty coefficient λ, the results are drawn in Figure 5. These results demonstrate that WDMAN model can achieve stable and desirable performance in a wide range of λ, which may benefit from more appropriate domain critic networks trained from the adversarial training strategy. However, the performance has a little decline in $D \rightarrow A$, when λ is 50 and 100. So, we still prefer to follow the suggestion by Gulrajani *et al.* [44], let $\lambda = 10$ for the fault diagnosis problem in this paper.

5) FEATURE VISUALIZATION

In order to demonstrate the transferability of all deep models and explain the reason why our proposed WDMAN outperforms than other methods on rolling bearing fault diagnosis under different working loads, we visualize the features of the fully connected layer before the output layer in this part. We take advantage of the data visualization technology called t-Distributed Stochastic Neighbor Embedding (t-SNE) [47] to reduce the high-dimensional features of the full-connection layer into a two-dimensional map for visualization.

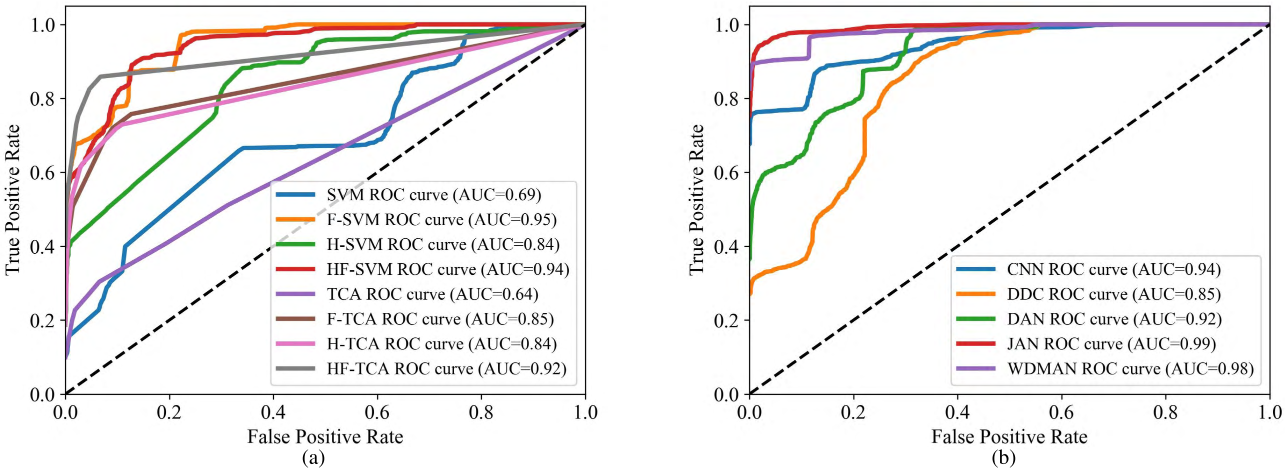
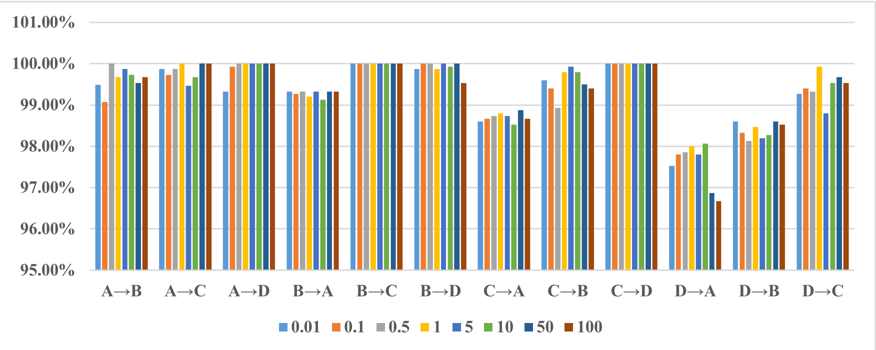


FIGURE 4. ROC curves and their AUCs of tradition models and deep models for CWRU dataset of tranfer task *D* → *B*. (a) Tradition model. (b) Deep model.



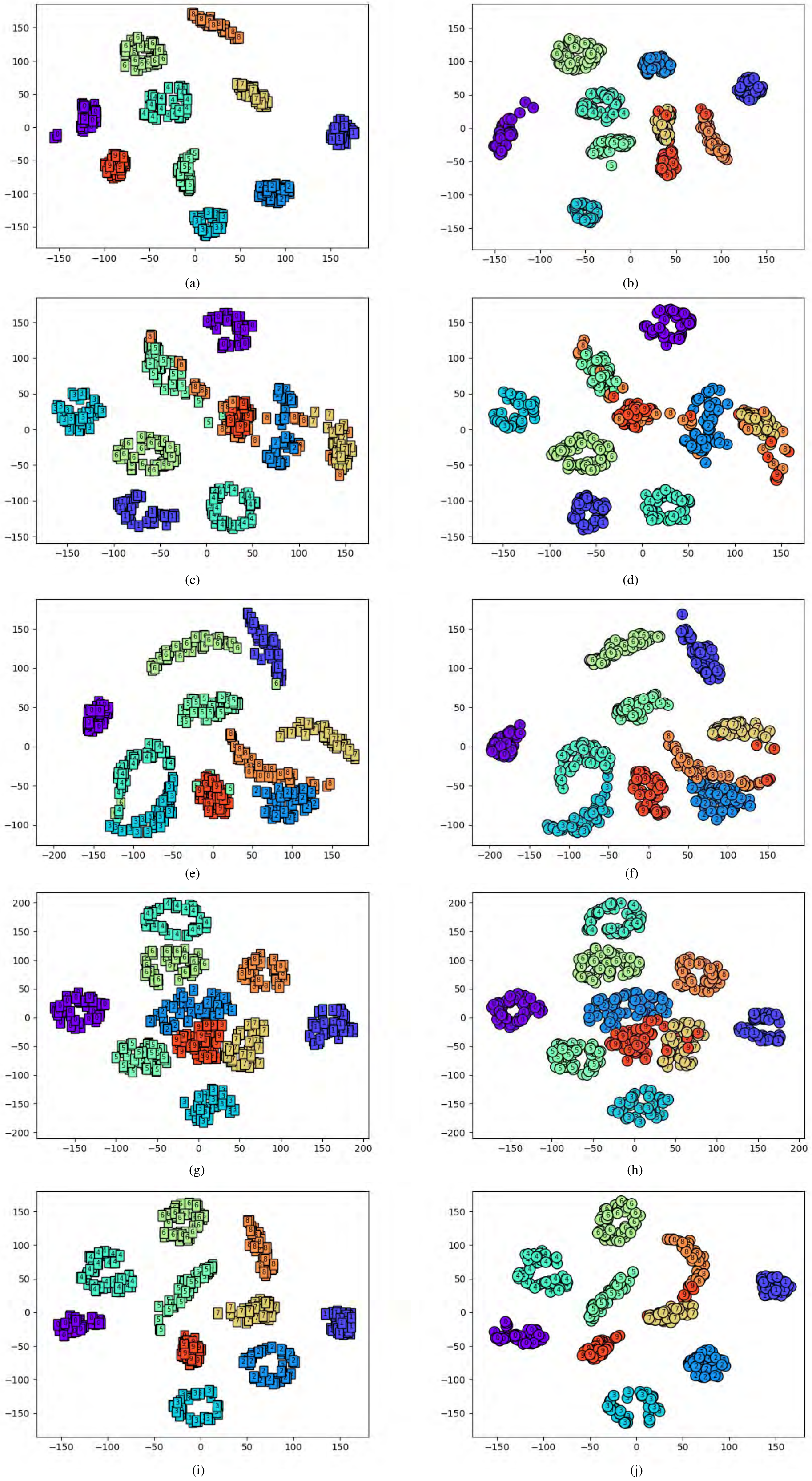
λ	0.01	0.1	0.5	1	5	10	50	100
A→B	99.49%	99.07%	100.00%	99.67%	99.87%	99.73%	99.53%	99.67%
A→C	99.87%	99.73%	99.87%	100.00%	99.47%	99.67%	100.00%	100.00%
A→D	99.33%	99.93%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→A	99.33%	99.27%	99.33%	99.20%	99.33%	99.13%	99.33%	99.33%
B→C	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
B→D	99.87%	100.00%	100.00%	99.87%	100.00%	99.93%	100.00%	99.53%
C→A	98.60%	98.67%	98.73%	98.80%	98.73%	98.53%	98.87%	98.67%
C→B	99.60%	99.40%	98.93%	99.80%	99.93%	99.80%	99.50%	99.40%
C→D	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
D→A	97.53%	97.80%	97.86%	98.00%	97.80%	98.07%	96.87%	96.67%
D→B	98.60%	98.33%	98.13%	98.47%	98.20%	98.27%	98.60%	98.53%
D→C	99.27%	99.40%	99.33%	99.93%	98.80%	99.53%	99.67%	99.53%

FIGURE 5. Parameter analysis for penalty coefficient λ.

We take the transfer task $D \rightarrow B$ as an example, the visualization results of five deep models are displayed in Figure 6. For the benchmark CNN model, the distribution of each category in source domain is very distinguishable, but the target domain distribution of category 2, 8, and 9 completely separate from the source domain, as shown in Figure 6(a) and (b). This is why the CNN model train with source samples is difficult to identify the target samples. Through the process of transfer learning, the distribution of each category between the source and target domain become consistent. However, there are lots of misclassification for DDC method, which explains why DDC has low accuracy when going on the task $D \rightarrow B$. The observation also shows that the distribution between the source and target domain in DAN, JAN, and WDMAN occupy great consistency, and there are few incorrectly classified cases. Nevertheless, the distance between the distribution of each category is further in WDMAN, which means the last classification layer is more easier to train.

D. CASE II: RESULTS AND ANALYSIS ON MCP DATASET

In order to further test the performance of our method, different deep models have been analyzed with the MCP dataset under different speeds and powers. For the fairer comparison with CWRU dataset, in this case, the MCP dataset is first


(a)

(b)

(c)

(d)

(e)

(f)

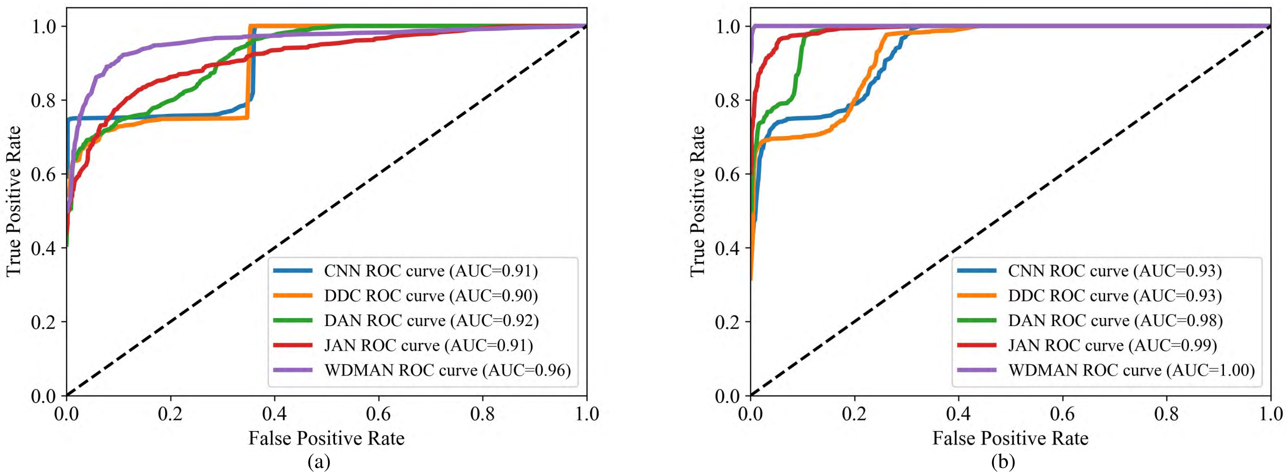
(g)

(h)

(i)

(j)

FIGURE 6. **Visualization of full-connection layer for CWRU dataset of tranfer task *D* → *B*. (a) CNN source. (b) CNN target. (c) DDC source. (d) DDC target. (e) DAN source. (f) DAN target. (g) JAN source. (h) JAN target. (i) WDMAN source. (j) WDMAN target.**



(a)

(b)

FIGURE 7. ROC curves and their AUCs of deep models for MCP dataset. (a) Transfer task $L \to H$. (b) Transfer task $H \to L$.

TABLE 10. Description of MCP dataset.

Fault location	None	Inner Race	Outer Race	Ball	Speed	Power
Category Labels	0	1	2	3		
Dataset L no.	500	500	500	500	1750 rpm	37 Kw
Dataset H no.	500	500	500	500	2960 rpm	94 Kw

TABLE 11. Deep models accuracy (%) for MCP dataset.

	L→H	H→L	AVG
CNN	54.83%	50.16%	52.50%
DDC	75.83%	73.00%	74.42%
DAN	86.67%	97.33%	92.00%
JAN	81.33%	93.00%	87.17%
WDMAN	**91.83%**	**100.00%**	**95.92%**

processed by downsampling from 25.6 kHz to 12 kHz, and the length of model input is also equal to 1200. The key parameter of WDMAN as follow: $\alpha = 0.0001$, $n_d = 2$, $\lambda = 10$, $n_c = 1000$, and $n_t = 20000$. The general hyperparameters for all models, in this case, is the same as Case I.

1) DATASET AND PREPROCESSING

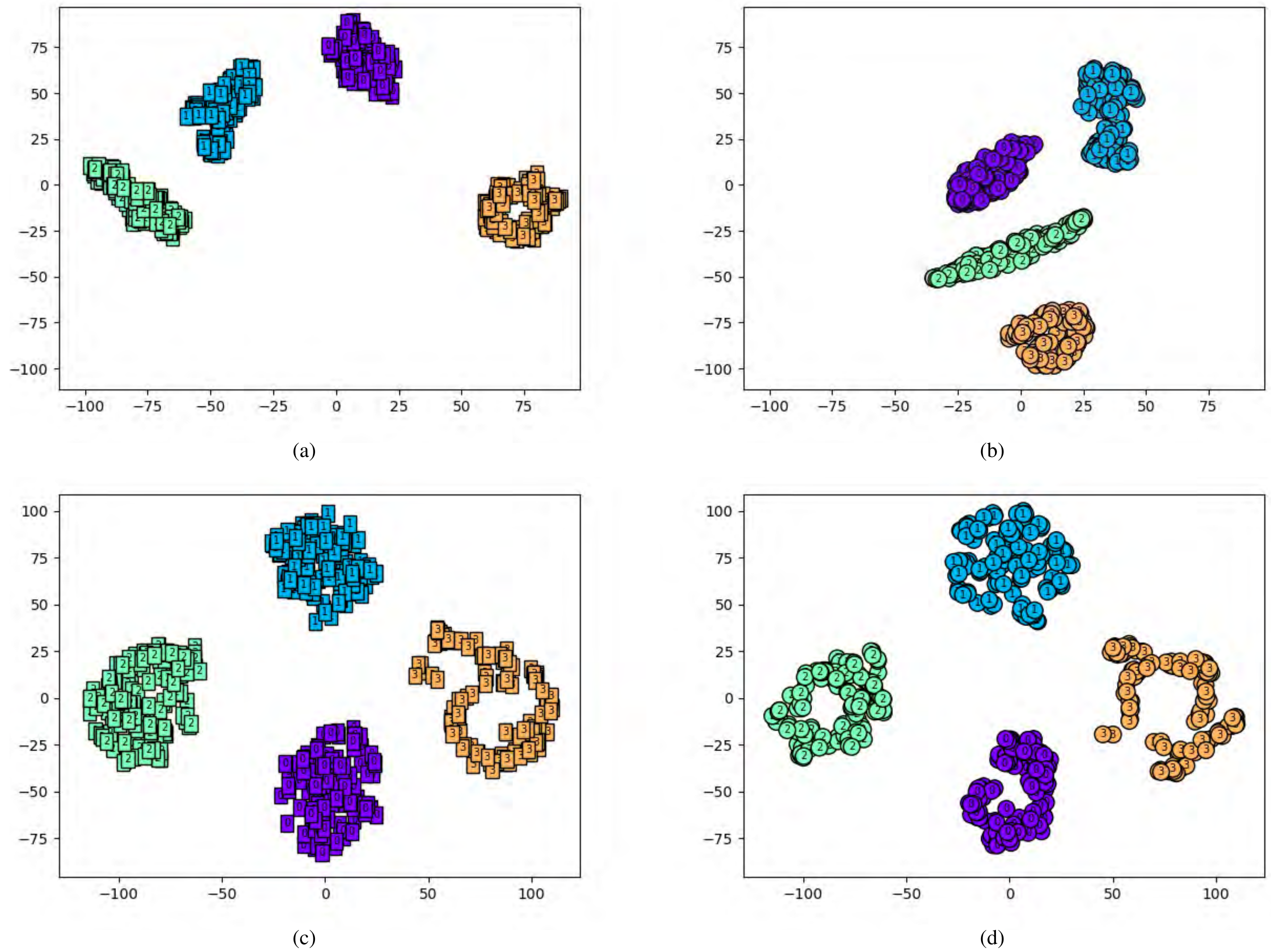
The second dataset was collected from rolling bearings of a real Multistage Centrifugal Pump (MCP) which consists of a drive motor, pump body and rolling bearing support [5]. The electrical-discharge machining (EDM) was used to make defects on a different location of rolling bearings including the inner race, outer race, and ball, respectively. The vibration data were acquired by the acceleration sensor which was mounted on the bearing house of the pump at different conditions with the sampling frequency of 25.6 kHz. For the real industrial centrifugal pump, the power changes with the rotating speed. In this dataset, the data samples are drawn from two different conditions called as domain L and H, the speeds are 1750 rpm and 2960 rpm, and powers are 37 kW and 96 kW. The detail of this dataset is shown in Table 10.

2) ACCURACY ACROSS DIFFERENT DOMAINS

The accuracies of the different experiments are calculated by (16). The results are shown in Table 11. The accuracies of the CNN model for these two transfer tasks are much lower, which demonstrates that different rotating speeds and powers significantly increase the distribution discrepancy between the source and target domain. The DDC model certain improves the accuracy for the target domain, but it is still not enough effective when compared with other deep models. The DAN is a little better than the JAN, and they are performing well in this case. It is obvious that our WDMAN model performs best for these two transfer tasks based on the results. There is a common phenomenon for the DAN, JAN, and WDMAN that the accuracy of task $L \to H$ is a little lower than $H \to L$. We think it may be incurred by the different speeds. Since the input sample length of our method remains unchanged, the high-speed data will provide more information than the data from low-speed, when the sampling frequency is the same. Therefore, the base model will learn more from the labeled source samples at high-speed, and it will make the transfer task $H \to L$ easier.

3) PERFORMANCE EVALUATION TEST

Table 12 displays the precision and recall of transfer tasks $L \to H$ and $H \to L$ for our WDMAN model. It is obvious that there is no false and missing alarm in the task $H \to L$ since all the precision and recall equal to 100%. However, the normal and inner race fault in task $L \to H$ present poor precision, which are 82.58% and 86.36%. This means that about 15% failures are inaccurately classified in categories 1 and 2. About 10% and 20% of fault categories 1 and 2 are undetected based on the recall of task $L \to H$. The ROC curves and their AUCs for all transfer tasks in MCP dataset are shown in Figure 7. The results confirm that there are no false and missing alarm for task $H \to L$ and some certain false and missing alarm occurs in task $L \to H$. Our WDMAN model performs best in the performance evaluation test.



(a) (b) (c) (d)

FIGURE 8. Visualization of full-connection layer for MCP dataset of tranfer task $H \to L$. (a) CNN source. (b) CNN target. (c) WDMAN source. (d) WDMAN target.

TABLE 12. Precision (%) and recall (%) of the proposed WDMAN for MCP dataset.

Fault location		None	Inner Race	Outer Race	Ball
Category Labels		0	1	2	3
Precision	L→H	82.58%	86.36%	100.00%	100.00%
	H→L	100.00%	100.00%	100.00%	100.00%
Recall	L→H	91.87%	78.62%	100.00%	98.31%
	H→L	100.00%	100.00%	100.00%	100.00%

4) FEATURE VISUALIZATION

The visualization results obtained by using t-SNE technology for transfer task $H \to L$ are shown in Figure 8. In the CNN model, the distributions of source and target domain are highly differentiated, which means that the feature layers are trained well and ready for classifying. However, the distribution discrepancy for each category between the source and target domain is quite large. This explains why the CNN model learned from source samples can not accurately classify the target samples. The results displayed in Figure 8(c) and (d) illustrate that after processed by our WDMAN method the distribution between the source and target domain becomes completely consistent and there is no misclassification. The observation results demonstrate that the WDMAN model is very effective in promoting the domain transfer capability and provides better robustness for the deep convolutional neural network model on the problem of rolling bearing fault diagnosis under different working conditions.

V. CONCLUSION

This paper presents a new model, WDMAN, to address transfer learning problem in the rolling bearing fault diagnosis under different working conditions through adversarial learning method. WDMAN model takes advantage of convolutional neural network (CNN) and generative adversarial network (GAN) to solve this transfer problem. The idea of our method is to learn domain invariant feature between the source and target domain through adversarial training strategy with Wasserstein Distance Guided Multi-Adversarial Networks. The experimental results on real-world datasets validate the superiority of our proposed model and demonstrate that WDMAN outperforms the state-of-the-art domain transfer learning methods.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[2] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[3] H. Qiu, J. Lee, J. Lin, and G. Yu, "Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics," *J. Sound Vib.*, vol. 289, nos. 4–5, pp. 1066–1090, 2006.

[4] K. Feng, Z. Jiang, W. He, and Q. Qin, "Rolling element bearing fault detection based on optimal antisymmetric real Laplace wavelet," *Measurement*, vol. 44, no. 9, pp. 1582–1591, 2011.

[5] M. Zhang, Z. Jiang, and K. Feng, "Research on variational mode decomposition in rolling bearings fault diagnosis of the multistage centrifugal pump," *Mech. Syst. Signal Process.*, vol. 93, pp. 460–493, Sep. 2017.

[6] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, Jan. 2019.

[7] 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," *Mech. Syst. Signal Process.*, vol. 72, pp. 303–315, May 2016.

[8] C. Li, R.-V. Sanchez, G. Zurita, M. Cerrada, D. Cabrera, and R. E. Vásquez, "Gearbox fault diagnosis based on deep random forest fusion of acoustic and vibratory signals," *Mech. Syst. Signal Process.*, vol. 76, pp. 283–293, Aug. 2016.

[9] P. Wang, R. Yan, and R. X. Gao, "Virtualization and deep recognition for system fault classification," *J. Manuf. Syst.*, vol. 44, pp. 310–316, Jul. 2017.

[10] G. Hu, H. Li, Y. Xia, and L. Luo, "A deep Boltzmann machine and multi-grained scanning forest ensemble collaborative method and its application to industrial fault diagnosis," *Comput. Ind.*, vol. 100, pp. 287–296, Sep. 2018.

[11] J. Xie, G. Du, C. Shen, N. Chen, L. Chen, and Z. Zhu, "An end-to-end model based on improved adaptive deep belief network and its application to bearing fault diagnosis," *IEEE Access*, vol. 6, pp. 63584–63596, 2018.

[12] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation with multiple sources," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1041–1048.

[13] W.-S. Chu, F. De la Torre, and J. F. Cohn, "Selective transfer machine for personalized facial action unit detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3515–3522.

[14] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[15] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. (2014). "Deep domain confusion: Maximizing for domain invariance." [Online]. Available: https://arxiv.org/abs/1412.3474

[16] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.

[17] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 136–144.

[18] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2200–2207.

[19] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2208–2217.

[20] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1647–1657.

[21] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4068–4076.

[22] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2015.

[23] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7167–7176.

[24] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[25] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 3934–3941.

[26] F. Shen, C. Chen, R. Yan, and R. X. Gao, "Bearing fault diagnosis based on svd feature extraction and transfer learning classification," in *Proc. Prognostics Syst. Health Manage. Conf. (PHM)*, Oct. 2015, pp. 1–6.

[27] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang, "Deep model based domain adaptation for fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2296–2305, Mar. 2017.

[28] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 1, pp. 136–144, Jan. 2019.

[29] W. Qian, S. Li, and J. Wang, "A new transfer learning method and its application on rotating machine fault diagnosis under variant working conditions," *IEEE Access*, vol. 6, pp. 69907–69917, 2018.

[30] Z. Tong, W. Li, B. Zhang, F. Jiang, and G. Zhou, "Bearing fault diagnosis under variable working conditions based on domain adaptation using feature transfer learning," *IEEE Access*, vol. 6, p. 76 187–76 197, 2018.

[31] 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.

[32] W. Zhang, C. Li, G. Peng, Y. Chen, and Z. Zhang, "A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load," *Mech. Syst. Signal Process.*, vol. 100, pp. 439–453, Feb. 2018.

[33] 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 Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7316–7325, Sep. 2018.

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

[35] X. Li, W. Zhang, and Q. Ding, "Cross-domain fault diagnosis of rolling element bearings using deep generative neural networks," *IEEE Trans. Ind. Electron.*, vol. 66, no. 7, pp. 5525–5534, Jul. 2019.

[36] X. Li, W. Zhang, and Q. Ding, "A robust intelligent fault diagnosis method for rolling element bearings based on deep distance metric learning," *Neurocomputing*, vol. 310, pp. 77–95, Oct. 2018.

[37] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[39] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[41] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[42] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[43] C. Villani, *Optimal Transport: Old and New*, vol. 338. Berlin, Germany: Springer, 2008.

[44] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5767–5777.

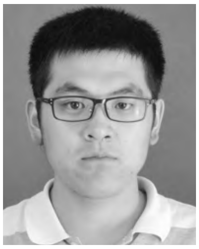
[45] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[46] K. Loparo, "Case western reserve University bearing data center," Case Western Reserve Univ. Bearing Data Center, Cleveland, OH, USA, Tech. Rep., 2012. [Online]. Available: http://csegroups.case.edu/bearingdatacenter/pages/download-data-file

[47] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

MING ZHANG received the B.S. and Ph.D. degrees in mechanical engineering from the Beijing University of Chemical Technology, Beijing, China, in 2011 and 2017, respectively.

He is currently a Postdoctoral Research Fellow with the Department of Information, Graduate School at Shenzhen, Tsinghua University. His research interests include deep learning and transfer learning, and their application in fault diagnosis and computer vision.



DUO WANG received the B.S. degree in automation from the Harbin Institute of Technology, China, in 2015. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, Beijing, China.

His current research interests include weakly-supervised deep learning/machine learning, few-shot learning, and their applications in computer vision and robotics vision.

WEINING LU received the B.S. degree from the Department of Physics, Fudan University, Shanghai, China, in 2011, and the Ph.D. degree from the Department of Automation, School of Information Science and Technology, Tsinghua University, Beijing, China, in 2017.

His current research interests include solving anomaly detection problems by using deep architecture networks, computer vision, and data mining.

JUN YANG received the B.S. degree in automation from Northwestern Polytechnical University, Xi'an, China, in 2004, the M.E. degree in automation from the Beijing University of Posts and Telecommunications, Beijing, China, in 2007, and the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2011.

He is currently a Lecturer with the Department of Automation, Tsinghua University.

ZHIHENG LI (M'05) received the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2009.

He is currently an Associate Professor with the Department of Automation, Tsinghua University, and with the Graduate School at Shenzhen, Tsinghua University, Shenzhen, China. His research interests include traffic operation, advanced traffic management systems, urban traffic planning, and intelligent transportation systems. He serves as an Associate Editor for the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS.

BIN LIANG received the B.S. and M.S. degrees in control engineering from Northwestern Polytechnical University, Xi'an, China, in 1991 and 1994, respectively, and the Ph.D. degree in precision instruments and mechanology from Tsinghua University, Beijing, China, in 1994.

He is currently a Professor with the Department of Automation, Tsinghua University. His current research interests include artificial intelligence, anomaly detection, space robotics, and fault-tolerant control.