

Article

Failure Mechanism Information-Assisted Multi-Domain Adversarial Transfer Fault Diagnosis Model for Rolling Bearings under Variable Operating Conditions

Zhidan Zhong ^{1,*} , Zhihui Zhang ¹, Yunhao Cui ¹, Xinghui Xie ² and Wenlu Hao ³

¹ School of Mechanical and Electrical Engineering, Henan University of Science and Technology, Luoyang 471023, China; zzh@stu.haust.edu.cn (Z.Z.); 9906412@haust.edu.cn (Y.C.)

² State Key Laboratory of Aerospace Precision Bearings, Luoyang 471000, China

³ Luoyang Xinqianglian Slewing Bearing Co., Ltd., Luoyang 471000, China

* Correspondence: zzd@haust.edu.cn

Abstract: Deep transfer learning tackles the challenge of fault diagnosis in rolling bearings across variable operating conditions, which is pivotal for intelligent bearing health management. Traditional transfer learning may not be able to adapt to the specific characteristics of the target domain, especially in the case of variable working conditions or lack of annotated data for the target domain. This may lead to unstable training results or negative transfer of the neural network. This paper proposes a new method for enhancing unsupervised domain adaptation in bearing fault diagnosis, aimed at providing robust fault diagnosis for rolling bearings under varying operating conditions. It incorporates bearing fault finite element simulation data into the domain adversarial network, guiding adversarial training using fault evolution mechanisms. The algorithm establishes global and subdomain classifiers, with simulation signals replacing label predictions for target data in the subdomain, ensuring minimal information transfer. By reconstructing the loss function, we can extract the common features of the same type bearing under different conditions and enhance the domain antagonism robustness. The proposed method is validated using two sets of testbed data as target domains. The results demonstrate that the method can adequately adapt the deep feature distributions of the model and experimental domains, thereby improving the accuracy of fault diagnosis in unsupervised cross-domain scenarios.

Keywords: bearing fault diagnosis; finite element; loss function; simulation domain; transfer learning; unsupervised fault diagnosis



Citation: Zhong, Z.; Zhang, Z.; Cui, Y.; Xie, X.; Hao, W. Failure Mechanism Information-Assisted Multi-Domain Adversarial Transfer Fault Diagnosis Model for Rolling Bearings under Variable Operating Conditions.

Electronics **2024**, *13*, 2133. <https://doi.org/10.3390/electronics13112133>

Academic Editor: Davide Astolfi

Received: 10 May 2024

Revised: 27 May 2024

Accepted: 28 May 2024

Published: 30 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Rolling bearings are widely used in industrial equipment. Their health is closely related to machinery and equipment failure prediction and health management (PHM). For this reason, the condition monitoring of rolling bearings has received attention from users at different stages. Condition monitoring for bearings generates a large amount of data. Deep learning facilitates the creation of end-to-end rolling bearing fault diagnosis models, effectively processing this large amount of data. The application of extensive monitoring data [1] is anticipated to enhance machinery and equipment's fault prediction and health management (PHM).

Neural network methods have become a hot topic in the field of rolling bearing fault diagnosis [2–4]. The advancement of rolling bearing fault diagnosis parallels the rapid development of machine learning theories and techniques [5]. Early researchers primarily focused on utilizing signal processing methods to extract features containing fault information. For instance, in the early stages of bearing failure, there may be normal temperature, slightly increased noise, and slightly elevated total vibration velocity and acceleration. A significant rise in vibration spike energy can be observed, and empirical modal decomposition (EMD) or signal kurtosis can be used to extract fault characteristics.

These extracted features can then be input into a shallow neural network, for fault diagnosis, such as an artificial neural network (ANN). Currently, the deep learning methods focus on constructing end-to-end fault diagnosis models, including Convolutional Neural Networks (CNNs), Deep Belief Networks (DBNs), and Auto-Encoders (AEs). The earliest CNN models were proposed by LeCun et al. [6], followed by the development of the VGG Net [7] and the 152-layer ResNet [8], which illustrate the evolution of deep networks. Lei et al. [5] introduced classical deep learning into the field of fault diagnosis of rotating machinery to the extent of bearings. Lei et al. led the craze of using deep neural networks to complete bearing fault diagnosis.

However, the remarkable success of deep learning in bearing fault diagnosis relies on a key assumption. This assumption is that the dataset used to train the neural network (source domain) follows the same distribution as the dataset in the target application scenario (target domain). In practice, this assumption often does not hold true in engineering applications. Transfer learning is anticipated to address this challenge. The core issue in transfer learning is handling the distributional differences between the source and target domain data. Essentially, this involves reducing the discrepancies between the marginal and conditional distributions of the source and target domain data. To enhance transfer performance, Gretton et al. [9] improved the Maximum Mean Discrepancy (MMD) to Multiple Kernel Maximum Mean Discrepancy (MK-MMD), which better characterizes the differences between the source and target domains. Li et al. [10] employed an optimal integrated deep transfer network to automate the classification of different faults in rolling bearings. Yu et al. [11] designed a generalized convolutional neural network (BCNN) with incremental learning capability for fault detection in industrial processes [12].

The methods proposed in the above literature have been well relocated in unsupervised domain adaptation (UDA); however, there are still some challenging issues. (1) In the absence of sufficient real data support, the significant difference in bearing fault diagnosis information under different operating conditions leads to the problem of the insufficient robustness of traditional methods in cross-condition diagnosis. (2) The introduction of simulated data provides an opportunity to consider the effective integration of real and simulated data, as well as the establishment of an effective transfer learning mechanism to deal with fine-grained alignment between multiple domains. (3) In the environment of bearing fault diagnosis under variable operating conditions, traditional methods are prone to negative migration due to the lack of a clear supervisory method to guide model learning, which leads to poor model performance in the target domain.

We aim to build a model that can reflect the characteristics of the bearing itself, serving as a guide for troubleshooting. The domain that contains certain knowledge of the mechanism of the transferred object is called the model domain. Representing the source domain as D_s and the target domain as D_t , we use D_m to denote the model domain. Generally, the source domain, such as the bearing failure data collected in the laboratory, contains many fully labeled data. In most scenarios, the target domain is the opposite; it typically has either fully unlabeled data (UDA) or a few labeled samples (Semi-Supervised Domain Adaptation) [13]. Deep learning models trained on the source domain often suffer performance degradation when applied to the target domain. This degradation is due to differences between the source and target domains. Transfer learning applied to the field of bearing fault diagnosis helps to build diagnostic models with generalization capabilities. These models can adapt to variations in operating conditions, loads, and components [14].

Based on the preceding discussion, transfer learning can be guided by analyzing the target object under diagnosis, such as a bearing, to acquire model domain data or probability functions. Taking rolling bearings as an example, a common approach involves analyzing the damage mechanism and creating analytical models of the bearings. McFadden et al. [15] identified shock vibrations generated by a rolling element passing through a single point of failure in the inner ring as an impulse sequence function. Some newer studies [16–19] have shown that the fault excitation in a faulty bearing consists of a cyclic impulse force of equal amplitude with a smaller random variable, i.e., a random slipping phenomenon.

The above is to further refine the difference between the simulated and actual signals by establishing a differential equation set method, which considers factors such as the random slipping of rolling elements, lubricant film stiffness, damping, the high-frequency resonance of the bearing housing, etc. However, the smaller the difference, the more complicated the differential equation set, making it difficult to apply to transfer learning. Based on ABAQUS display dynamics, Liu et al. [19] analyzed the changes in contact stresses and contact stress distribution of rings, rollers, and cages during operation.

Computer simulation software methods can reveal new phenomena that are difficult to obtain through theoretical analysis or experimental observation. They can also provide the intuitive bearing dynamic response and faulty bearing vibration signals, which are directly applicable to bearing fault diagnosis. This paper's contributions are summarized as follows:

- (1) The proposal is to develop a domain-adversarial transfer learning network that fuses bearing simulation data. This network should be able to cope with significant differences in bearing fault signal features under different operating conditions. Furthermore, it should improve the robustness and reliability of the model for fault diagnosis under variable operating conditions.
- (2) A novel transfer learning mechanism for bearing simulation signals is proposed, comprising a global domain classifier and a subdomain classifier based on simulation signals. This mechanism supports the precise alignment of multiple domains, thereby promoting positive migration, mitigating negative migration, and enhancing the model's generalization capability under different operating conditions.
- (3) The optimal supervision method for simulation signals is determined. The reconstruction of the loss function and the design of the domain adversarial network serves to effectively guide the model in learning the target domain's fault characteristics; this can improve the robustness and reliability of the model and resolve the issue of a tendency toward negative transfer.

2. Theoretical Foundation

Some researchers have carried out related studies and made certain progress in UDA.

The domain adversarial neural network (DANN) [20] aims to integrate domain adaptation and deep feature learning into a single training process. The goal is to include domain adaptation in representation learning. This ensures that classification relies on invariant features. These features should have similar distributions in both the source and target domains. The system's structure is illustrated in Figure 1, primarily comprising Feature Extractor G_f , Domain Classifier G_d , and Label Predictor G_y .

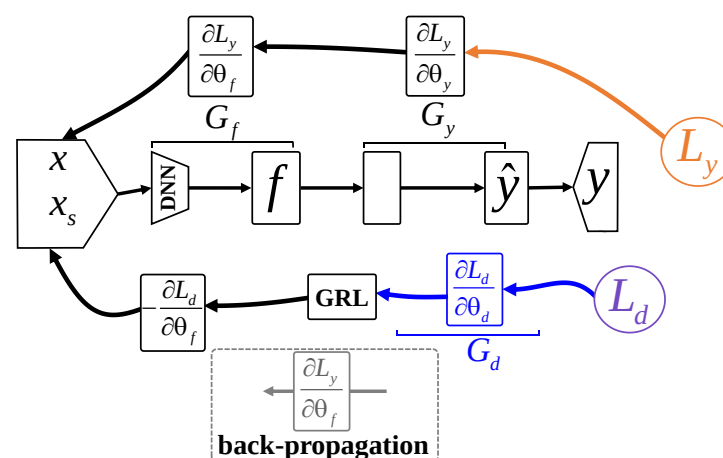


Figure 1. DANN basic architecture.

Given a labeled sample set $D_s = \{x_i^s, y_i^s\}$ in the source domain and an unlabeled sample set $D_t = \{x_i^t\}$ in the target domain, the loss of its label predictor is

$$\mathcal{L}_y(G_y(G_f(x_i)), y_i) = \log \frac{1}{G_y(G_f(x))_{y_i}}. \quad (1)$$

The optimization objective for the source domain is as follows:

$$\min_{\mathbf{W}, \mathbf{b}, \mathbf{V}, \mathbf{c}} \left[\frac{1}{n} \sum_{i=1}^n \mathcal{L}_y^i(G_y(G_f(x_i; \mathbf{W}, \mathbf{b}); \mathbf{V}, \mathbf{c}), y_i) + \lambda \cdot R(\mathbf{W}, \mathbf{b}) \right], \quad (2)$$

In this equation, $\mathcal{L}_y^i$ represents the label prediction loss for the i th sample and $\lambda \cdot R(\mathbf{W}, \mathbf{b})$ is an additional term used to prevent overfitting of the neural network. The regularizer $R(\mathbf{W}, \mathbf{b})$ is optional and its weight is determined by hyperparameter λ .

The loss of the domain classifier $G_d(\cdot)$ is as follows:

$$\begin{aligned} \mathcal{L}_d(G_d(G_f(\mathbf{x}_i)), d_i) &= d_i \log \frac{1}{G_d(G_f(\mathbf{x}_i))} \\ &+ (1 - d_i) \log \frac{1}{1 - G_d(G_f(\mathbf{x}_i))}, \end{aligned} \quad (3)$$

The loss of an adversarial transfer network is composed of two parts: the label predictor loss (training loss of the network) and the domain discrimination loss. The DANN total objective function is

$$\begin{aligned} E(\mathbf{W}, \mathbf{V}, \mathbf{b}, \mathbf{c}, \mathbf{u}, z) &= \frac{1}{n} \sum_{i=1}^n \mathcal{L}_y^i(\mathbf{W}, \mathbf{b}, \mathbf{V}, \mathbf{c}) \\ &- l \left(\frac{1}{n} \sum_{i=1}^n \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) + \frac{1}{n_i} \sum_{i=n+1}^N \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) \right), \end{aligned} \quad (4)$$

In this, the parameters of the label predictor are updated by minimizing objective function $\mathcal{L}_y^i(\mathbf{W}, \mathbf{b}, \mathbf{V}, \mathbf{c})$ and maximizing objective function $\mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z)$ to update the parameters of the domain classifier. According to Ganin et al. [20], Equations (1)–(4) can be summarized.

3. Proposed Architecture

3.1. Model Domain Construction Based on Kinetic Finite Element Models

The set $D_m = \{x_i^m, y_i^m\}$ of faulty bearings obtained by the modeling approach defines the model domain sample set, while the model domain sample set contains the main factors that influence the fault signal. This simplification of the bearing system is reasonable. The construction of the model domain is based on kinetic finite element models.

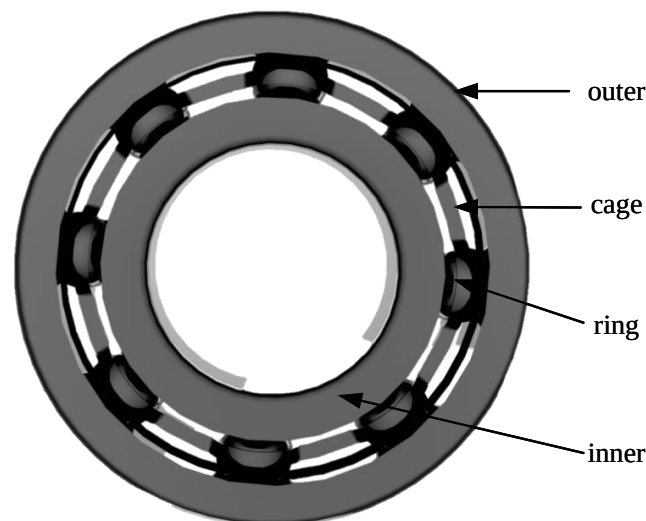
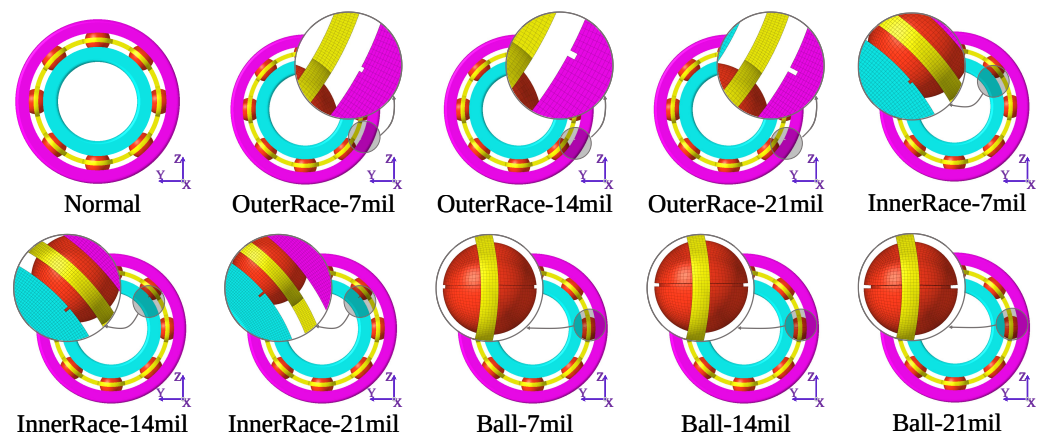
We establish a three-dimensional solid finite element model of a healthy rolling bearing based on the explicit algorithm. The analysis considers the conditions of frictional contact, velocity, and load to examine the variation in vibration acceleration in the vertical direction of the bearing. Using the healthy rolling bearing model, we simulated local spalling faults in the outer ring, inner ring, and rolling element by setting unit defects to construct the model domain dataset.

The paper models 6205 and 6203 rolling bearings for subsequent fault diagnosis. Table 1 shows the main dimensional parameters of these bearing models, which are also used in conjunction with the test cases.

Table 1. The 6205 and 6203 bearing part geometry.

Bearing Type	Parameter	Size/mm	Bearing Type	Parameter	Size/mm
6205	Outside diameter	52	6203	Outside diameter	40
	Bore diameter	25		Bore diameter	17
	Width	15		Width	12
	Ball diameter	8		Ball diameter	6.75
	Pitch diameter	39		Pitch diameter	28.5
	Pocket clearance	0.05		Pocket clearance	0.03
	Number of bearing balls	9		Number of bearing balls	8

The modeling process for the ten types of faulty bearings is the same as that for the three types of faulty bearings. Using the Case Western Reserve University bearing dataset drive end bearing decile failure data as an example, Figure 2 shows the three-dimensional model of the bearing dynamics. The model consists of the inner ring, outer ring, rolling element, and cage. Figure 3 shows the fault parts after meshing, assuming constant stiffness and damping of the main components, and with the fault parts being the inner ring, outer ring, and rolling body. For normal, inner-ring, outer-ring, and rolling body failures, the fault sizes are 7 mils, 14 mils, and 21 mils, respectively. There are a total of nine types of failure and one normal condition, which are represented by bar finite element meshes of varying sizes.

**Figure 2.** Three-dimensional modeling of bearing dynamics.**Figure 3.** Nine types of failure and one normal condition of 6205 bearing.

The experimental bearing failure manifested itself as grooves produced by electrical discharge machining (EDM) and this simulation mimics the same artificial damage as the experiment. The finite element model was meshed using Hypermesh software (version:2021.2) so that each mesh width was 7 mil. Bearing failures of different sizes exhibited grooves with a circumferential length of 7 mil and an axial depth of 7–21 mil.

The delineated mesh was submitted to ANSYS (Version:2020 R2) for finite element calculations. The equation of motion for a rolling bearing, taking into account the damping effect, is provided.

$$M\ddot{x}(t) = P(t) - F(t) + H(t) - C\dot{x}(t). \quad (5)$$

where M is the system mass matrix; and $x(t)$, $\dot{x}(t)$, and $\ddot{x}(t)$ are the position coordinate vector, velocity vector, and acceleration vector of the node, respectively. $P(t)$, $F(t)$, $H(t)$, and C , are the load vector, the internal force vector, the hourglass resistance vector, and the damping matrix, respectively. The explicit center difference method is used to solve the time integrals of the system equations.

3.2. Subdomain Classifier and Global Domain Classifier Design

Current domain adversarial neural networks feature a domain classifier. This classifier aligns the overall distribution of the source and target domains. However, it does not consider the alignment of the corresponding categories. This led to confusion between the data from the source and target domains and affected the discriminant component, resulting in misclassification and misalignment. This paper aims to tackle the issue by introducing simulation data and constructing a multimodal structure. This structure is expected to achieve the fine-grained alignment of multiple domain classifiers corresponding to different data distributions. This alignment will be at the category level.

When faced with the task of transferring multiple categorical labels, it is crucial to consider the direction of transfer. This means ensuring that the features of samples from the same category are aligned. Existing studies can be summarized into two main categories [21]: (a) instance reweighting, which reuses source domain samples using a weighting technique; and (b) feature matching, which implements subspace learning through subspace geometries. To achieve the fine-grained alignment of different data distributions based on multiple domain classifiers, Pei et al. [22] proposed a local domain classifier for the number of categorical categories. This classifier is used to handle domain adaptation for each category and optimize the conditional probability distribution. Yu et al. [21] proposed a deep adversarial network model that can dynamically adjust the relationship between edge and conditional distributions by introducing a conditional domain discriminant block and an integrated dynamic adjustment factor. Zhu et al. [23] defined subdomains based on class labels and grouped the same class into a subdomain.

The current method for fine-grained alignment in unsupervised domain adaptation depends on predicting labels for the target domain data. However, this approach does not entirely diminish the reliance on the source domain data during the transfer process, despite recognizing the phenomenon of negative transfer. Therefore, to achieve fine-grained alignment while avoiding negative transfer, it is necessary to introduce new data with label predictions that are independent of the source and target domain data. It is important to note that the model domain can provide labeled information that reveals a multimodal structure. By using modeling, we can acquire a labeled sample set of model domains that share the same working conditions as the target domain. We can then substitute the gradient inversion results, which are predicted by the labels of the target domain data in the subdomain classifier, with the labeled model domains. Please refer to Figure 4 for a visual representation. After label matching, the model domain and source domain's subdomain classifier computes the domain classification loss for each batch. Similarly, the global domain classifier computes the domain classification loss between the source domain and the target domain.

The loss of the global domain classifier G_d is

$$\mathcal{L}_d(G_d(G_f(\mathbf{x}_i)), d_i) = d_i \log \frac{1}{G_d(G_f(\mathbf{x}_i))} + (1 - d_i) \log \frac{1}{1 - G_d(G_f(\mathbf{x}_i))}. \quad (6)$$

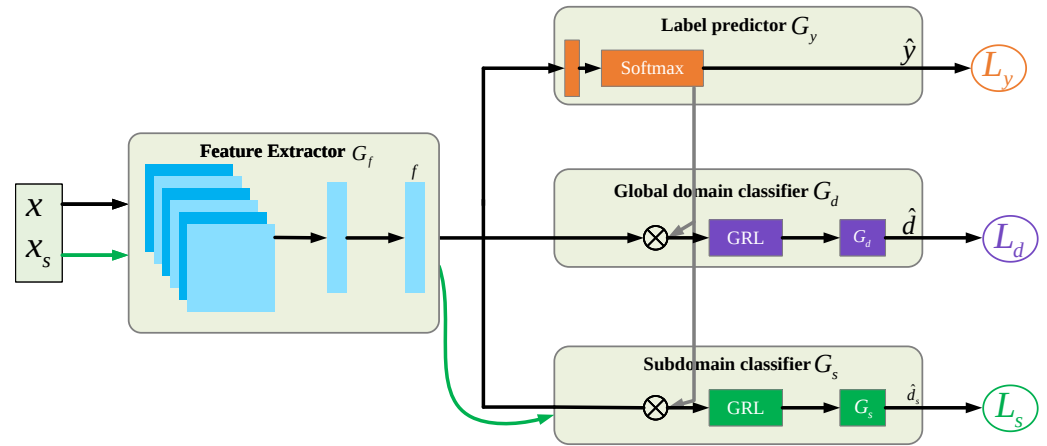


Figure 4. Network structure diagram of the proposed method.

A binary variable of the domain label is denoted by d_i . The loss of subdomain classifier G_s is defined as

$$\mathcal{L}_s(G_s(G_f(\mathbf{x}_i)), d_i) = d_i \log \frac{1}{G_s(G_f(\mathbf{x}_i))} + (1 - d_i) \log \frac{1}{1 - G_s(G_f(\mathbf{x}_i))}, \quad (7)$$

In this case, the domain label denoted by d_i is no longer the source or model domain. However, the calculation can still consider d_i as a binary label to determine whether it is from the source domain.

Referring to Equation (3) gives the following regularizer

$$R(\mathbf{W}, \mathbf{b}) = \max_{\mathbf{u}, \mathbf{z}} \begin{bmatrix} -\frac{1}{n} \sum_{i=1}^n \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) - \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) \\ -\frac{1}{n} \sum_{i=1}^n \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) - \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) \end{bmatrix}, \quad (8)$$

where

$$\mathcal{L}_d^i(\mathbf{w}, \mathbf{b}, \mathbf{u}, \mathbf{z}) = \mathcal{L}_d(G_d(G_f(\mathbf{x}_i; \mathbf{w}, \mathbf{b}); \mathbf{u}, \mathbf{z}), d_i), \quad (9)$$

$$\mathcal{L}_s^i(\mathbf{w}, \mathbf{b}, \mathbf{u}, \mathbf{z}) = \mathcal{L}_s(G_s(G_f(\mathbf{x}_i; \mathbf{w}, \mathbf{b}); \mathbf{u}, \mathbf{z}), d_i). \quad (10)$$

The optimization objective of the domain classifier is to combine the subdomain classifiers and the global domain classifiers.

$$\begin{aligned} E(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) = & -\lambda \left(\frac{1}{n} \sum_{i=1}^n \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) + \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) \right) \\ & - \lambda_s \left(\frac{1}{n} \sum_{i=1}^n \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) + \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, \mathbf{z}) \right), \end{aligned} \quad (11)$$

The weights of the global domain classifier and subdomain classifier are adjusted using hyperparameters λ and λ_s .

The DANN model includes only the global domain classifier G_d . Fine-grained alignment methods, such as Multi-Adversarial Domain Adaptation (MADA) and Dynamic Adversarial Adaptation Network (DAAN), rely on subdomain classifier samples that are the model's labeled predictions of the target domain. Our proposed methodology, however, is not entirely reliant on labeled predictions; hence, it effectively mitigates negative transfers to a greater degree.

The neural network structure responsible for domain classification is shown in Figure 5. The global domain classifier and subdomain classifier use the same network structure to distinguish whether the inputs belong to the same domain or not. The domain classifier contains Convolutional Layers, a Self-Attention Module, and Fully Connected Layers. Its output is compared with the real domain label to calculate the binary cross-entropy loss. The binary cross-entropy loss is calculated by comparing the output of the domain classifier with the real domain labels. $\mathcal{L}_d$ or $\mathcal{L}_s$ is outputted by either the global domain classifier G_d or subdomain classifier G_s .

The Self-Attention Module is added to distinguish between different domains. The Self-Attention Module enhances the ability of the domain classifier to focus on important features, and the domain classifier is better able to capture complex dependencies in the input data, thus improving the efficiency of the domain adaptation task. The dual classifier framework ensures a comprehensive and detailed domain adaptation process. The same architecture simplifies design and implementation while maintaining high performance.

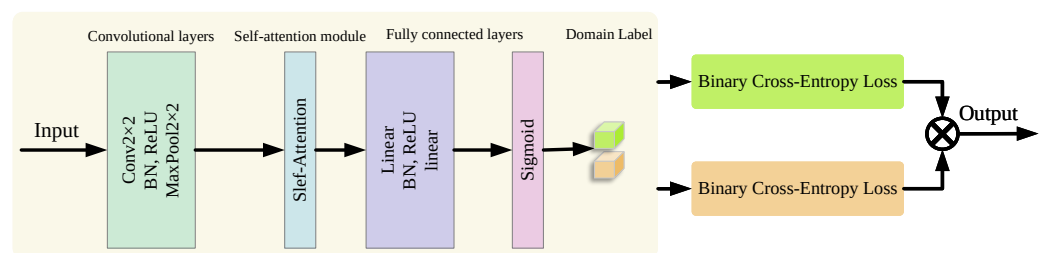


Figure 5. Structure and calculation methods for domain classifier and adversarial loss.

3.3. Improved Loss Function Design for Embedded Model Domains

The subdomain classifier indicates the similarity between the real data and the model domain for each fault. The global domain classifier reflects the similarity between the source and target domains. The model domain data is based on simulations conducted under the same operating conditions as the target domain. It reflects the degree of similarity between the source and target domains in terms of categories to some extent. The subdomain classifier aligns the conditional distribution, while the global domain classifier aligns the marginal distribution. To enhance the feature representation of the model and better express differences in data distribution in high-dimensional space, we have opted for MK-MMD instead of the conventional MMD metric.

One improvement over previous methods is the ability to simultaneously promote positive transfers of relevant data and mitigate negative transfers of irrelevant data. This is achieved by introducing a model domain sample set A instead of the network's labeled predictions of target domain samples. Therefore, the complete optimization objective of Equation (4) can be rewritten as follows:

$$\begin{aligned}
E(\mathbf{W}, \mathbf{V}, \mathbf{b}, \mathbf{c}, \mathbf{u}, z) &= \frac{1}{n} \sum_{i=1}^n \mathcal{L}_y^i(\mathbf{W}, \mathbf{b}, \mathbf{V}, \mathbf{c}) \\
&- \lambda \left(\frac{1}{n} \sum_{i=1}^n \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) + \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_d^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) \right) \\
&- \lambda_s \left(\frac{1}{n} \sum_{i=1}^n \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) + \frac{1}{n'} \sum_{i=n+1}^N \mathcal{L}_s^i(\mathbf{W}, \mathbf{b}, \mathbf{u}, z) \right),
\end{aligned} \quad (12)$$

The subdomain classifier against loss is utilized to achieve conditional distribution alignment from the source domain to the target domain, while the global domain classifier against loss is employed to achieve edge distribution alignment from the source domain to the target domain.

3.4. Model Structure and Optimization Methods

Figure 6 illustrates the fundamental architecture of the network model proposed in this study. The model comprises three modules: a feature extractor, a classifier, and a domain classifier.

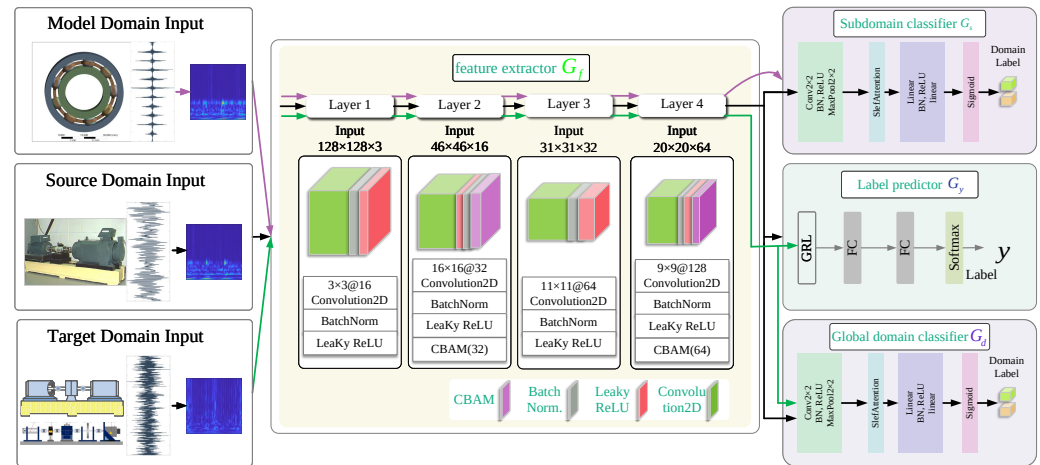


Figure 6. Network structure diagram.

In each epoch, the feature extractor G_f processes the source domain samples. Then, the label predictor's loss $\mathcal{L}_y$, the global domain classifier's loss $\mathcal{L}_d$ in comparison with the source domain samples, and the subdomain classifier's loss $\mathcal{L}_s$ in comparison with the model domain samples are computed.

To extract domain-invariant features f , we aim to minimize the label predictor loss $\mathcal{L}_y$ while maximizing the domain classifier losses $\mathcal{L}_d$ and $\mathcal{L}_s$. The feature extractor G_f 's parameters θ_f are learned by maximizing $\mathcal{L}_d$ and $\mathcal{L}_s$ to ensure domain invariance. The domain classifiers G_g and G_s 's parameters are learned by minimizing the domain classifier loss. To improve the network's feature representation, we have introduced CBAM [24] to the feature extractor.

Due to the large amount of data, an iterative algorithm is required to reduce the computational cost and accelerate the convergence. We chose the SGD algorithm as the optimizer for parameter updating to improve the training efficiency and performance of the model.

The optimization problem is to find the parameters $\hat{\theta}_f$, $\hat{\theta}_y$, $\hat{\theta}_d$, and $\hat{\theta}_s$ that jointly satisfy

$$\left(\hat{\theta}_f, \hat{\theta}_y \right) = \underset{\theta_f, \theta_y}{\operatorname{argmin}} E\left(\theta_f, \theta_y, \hat{\theta}_d \right), \quad (13)$$

$$\hat{\theta}_d = \operatorname{argmax}_{\theta_d} E(\hat{\theta}_f, \hat{\theta}_y, \theta_d, \hat{\theta}_s), \quad (14)$$

$$\hat{\theta}_s = \operatorname{argmax}_{\theta_s} E(\hat{\theta}_f, \hat{\theta}_y, \hat{\theta}_d, \theta_s). \quad (15)$$

The global domain classifier adversarial loss was used to reduce the difference between the source and target domains in the feature space. Still, it may misalign different kinds of bearing health state features, resulting in negative migration. The subdomain classifier adversarial loss is used to align the distribution of samples of the same type from different domains, updating parameters θ_f , θ_y , θ_d , and θ_s as

$$\begin{aligned} \theta_f &\leftarrow \theta_f - \varepsilon \left(\frac{\partial \mathcal{L}_y}{\partial \theta_f} - \lambda \frac{\partial \mathcal{L}_d}{\partial \theta_f} + \mu \frac{\partial D}{\partial \theta_f} \right) \\ \theta_y &\leftarrow \theta_y - \varepsilon \left(\frac{\partial \mathcal{L}_y}{\partial \theta_y} + \mu \frac{\partial D}{\partial \theta_y} \right) \\ \theta_d &\leftarrow \theta_d - \varepsilon \frac{\partial \mathcal{L}_d}{\partial \theta_d} \\ \theta_s &\leftarrow \theta_s - \varepsilon \frac{\partial \mathcal{L}_s}{\partial \theta} \end{aligned} \quad (16)$$

where ε is the learning rate. In the forward propagation, the label predictor G_y calculates $\mathcal{L}_y$ through Equation (1). The global domain classifier G_d compares the source and target domain feature vectors and calculates $\mathcal{L}_d$ through Equation (6). The subdomain classifier G_s compares the feature vectors of the model and the source domains and calculates $\mathcal{L}_s$ through Equation (7). In the backpropagation, the SGD optimizer is based on Equation (16) to update θ_f , θ_y , θ_d , and θ_s .

The work described above has achieved three objectives:

- (1) The accuracy of predictions is maximized.
- (2) Alignment of edge distribution from the source domain to the target domain.
- (3) The approach utilizes network-independent model domain data instead of predicting labels for the target domain. Each subdomain classifier matches corresponding kinds of source and model domain data to achieve conditional distributional alignment from the source domain to the target domain.

4. Experiments and Analysis

4.1. Description of Datasets

The method's effectiveness is demonstrated through two cross-domain bearing fault diagnosis cases.

(1) Case 1

Case 1 is an experimental bearing dataset for condition monitoring (CM) based on vibration and motor current signals provided by the University of Paderborn, Germany [25]. Figure 7 displays the test rig, which comprises several modules arranged from left to right: motor, torque measuring shaft, rolling bearing test module, flywheel, and load motor. The test bearing model is a 6203 rolling bearing and the vibration signals are sampled at 64 KHz. The signals contain three health conditions: normal, internal fault, and external fault. Table 2 provides further details on the three domains.

Table 2. Case 1: details of domain datasets.

Datasets	Bearing Types	Health States			Load (hp)
Source domain	6203	Normal	Inner race fault	Outer race fault	W_0
Target domain	6203	Normal	Inner race fault	Outer race fault	W_3
Model domain	6203	Normal	Inner race fault	Outer race fault	W_3

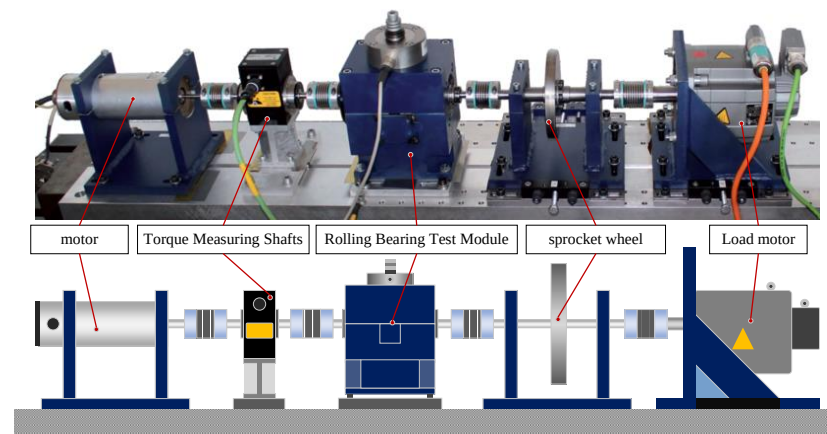


Figure 7. Test rig and measuring equipment.

(2) Case 2

The CWRU bearing dataset came from Case Western Reserve University (CW-RU) ElectrotechnicsLab [26]. Figure 8 shows the recorded vibration signals of the bearing's inner ring, outer ring, ball failure, and normal bearing at 0, 1, 2, and 3 hp loads. The vibration signals of the drive end (DE) were used in this paper with a sampling frequency of 12 KHz. Table 3 provides detailed information on the three domains.

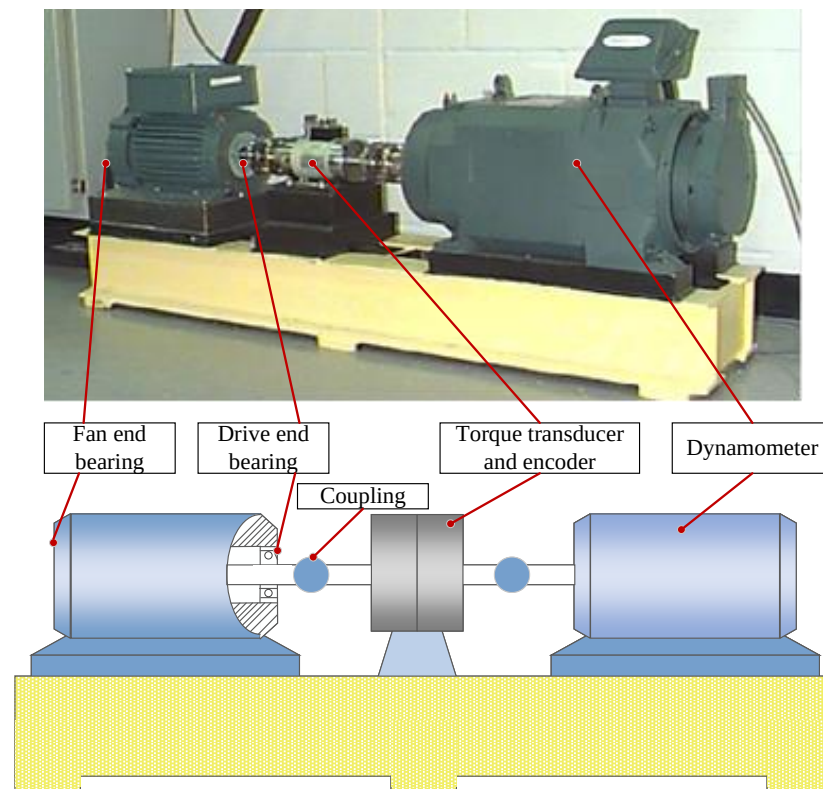


Figure 8. CW-RU ElectrotechnicsLab.

Table 3. Case 2: details of domain datasets.

Datasets	Bearing Types		Health States		Load (hp)
Source domain	6203	Normal	Inner race fault (7, 14, 21 mil)	Outer race fault (7, 14, 21 mil)	0
Target domain	6203	Normal	Inner race fault (7, 14, 21 mil)	Outer race fault (7, 14, 21 mil)	3
Model domain	6203	Normal	Inner race fault (7, 14, 21 mil)	Outer race fault (7, 14, 21 mil)	3

4.2. Model Domain Dataset

The data in the model domain consist of bearing vibration signals from four states obtained through the finite element simulation method. The simulation method is illustrated in Figure 9 and the simulation parameters are consistent with those of the target domain. The bearing failures in the experiments were manifested as grooves of different depths produced by the EDM and the finite element simulation modeled the same deepening bearing failures along the radial direction as in the experiments, i.e., the types of failures in the experiments were the same as those in the EDM. The experiments were the same as those in the simulations. The signal processing method used in the model domain is consistent with that of the source and target domains. Using the rolling bearing simulation data for the 14 mil bearing with a failed outer ring as an example, Figure 10 displays the envelope signals of the vibration acceleration signals and their spectra collected when the rotary axis has an angular velocity of 1796 rad/min and a radial force of 2000 N.

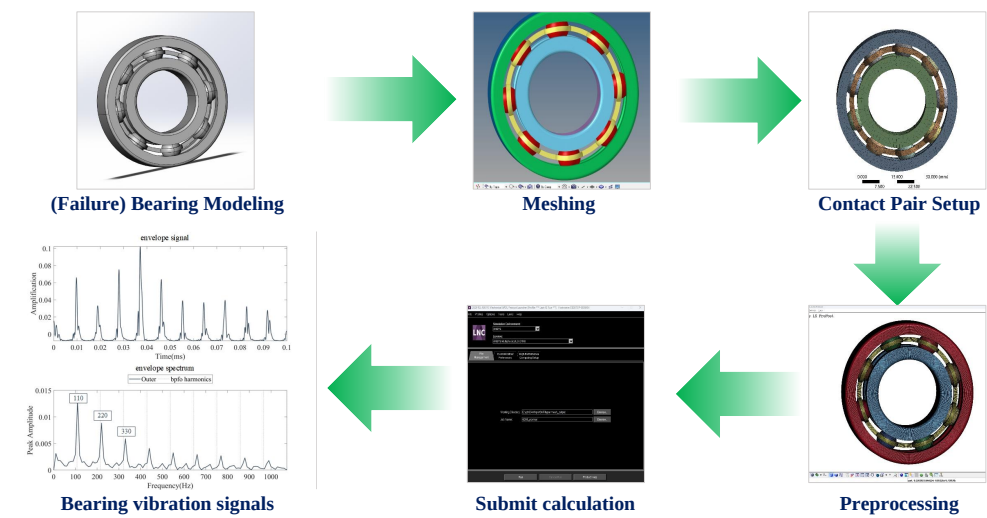


Figure 9. Simulation methods for modeling domains.

During the uniform acceleration and uniform loading stage, the rolling bearing experiences a radial load of 2000 N while rotating at a fixed speed of 1796 rad/min. After 0.005 s, the rotational speed is fixed and the time-domain waveform presents a series of equal-amplitude pulse shapes. When the bearing surface produces local defects, an impact excitation force is generated each time the defective part contacts other parts during rotation. This impact has a clear periodicity, resulting in sharp peaks in the spectrogram. The ball pass frequency outer race (*BPFO*) is calculated as

$$BPFO = \frac{N}{2}n \left[1 - \left(\frac{d \cos \alpha}{D} \right) \right] \quad (17)$$

where d represents the diameter of the rolling body, D represents the diameter at the center of the rolling body, α represents the contact angle in the radial direction, n represents the number of rolling bodies, and N represents the rotational speed of the shaft.

Note: The rolling bearing operates without sliding, and its geometry remains constant. Additionally, the outer ring of the bearing is fixed and does not rotate.

Using the 6205 deep groove ball bearing as an example, the simulated spectrum shows obvious spikes around 107.305 Hz and octave frequency, indicating the model's effectiveness in simulating the fault state of rolling bearings.

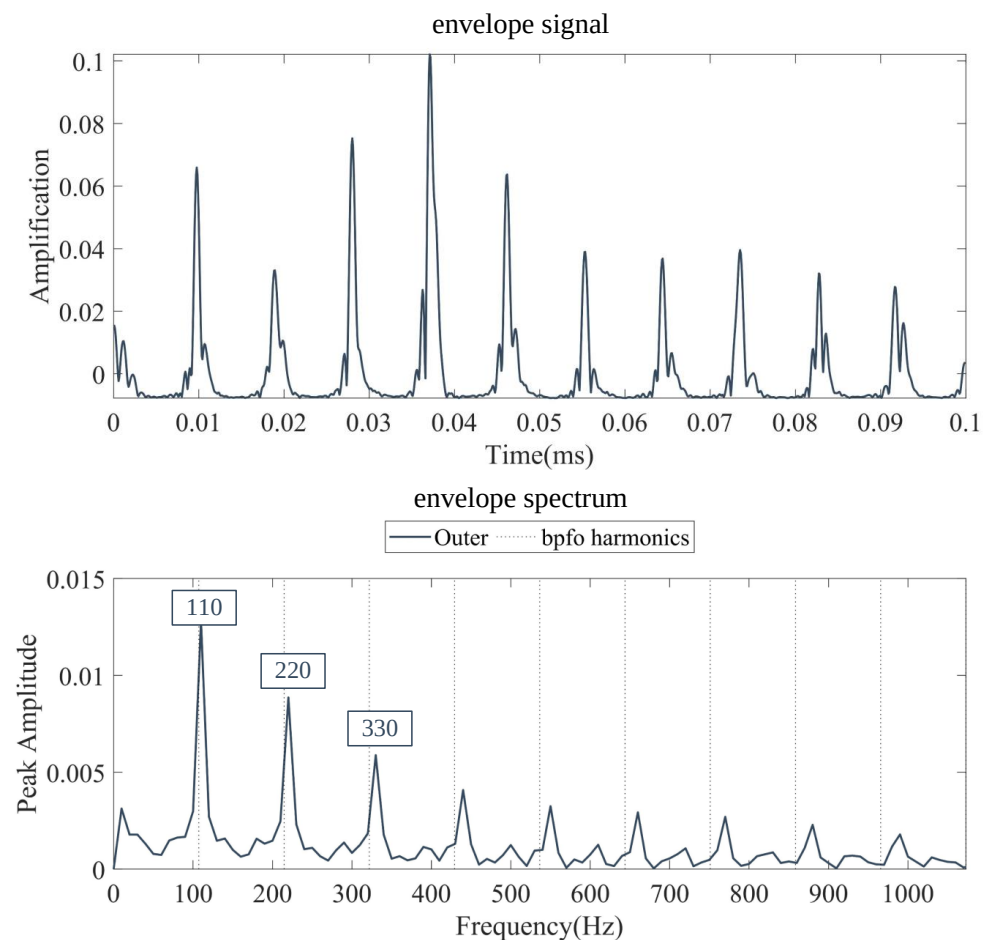


Figure 10. Envelope signals and their spectra of simulation data for rolling bearing outer-ring faults.

4.3. Experimental Setting

To determine the effectiveness of the proposed method, it was compared with seven alternative methods.

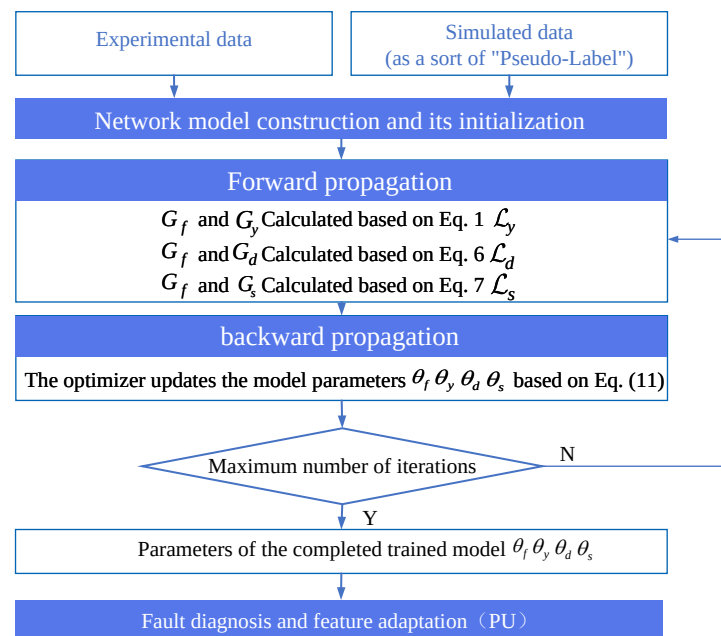
- (1) To verify the necessity of the transfer strategy, the proposed method (denoted as Method 1) is compared with a standard 2DCNN (denoted as Method 2).
- (2) In order to reflect the superiority of the proposed method, it is compared with the classical UDA methods, including DANN, deep adaptation network (DAN), and Multi-Adversarial Domain Adaptation (MADA), which are denoted as Method 3 to Method 5.
- (3) To demonstrate the respective effectiveness of the model domain sample participation with the improved neural network, comparisons were made with the DANN with only the model domain added and with the DANN with only the improved network structure, denoted as Methods 6 and 7, respectively.
- (4) Industrial processes face noise interference [27,28], and to reflect the anti-interference capability of the proposed method, the training results are compared with the data with Gaussian noise added, denoted as Method 8.

To reduce the effect of random initialization, 10 replications of the experiment were performed in each case. In two cases, the main parameters of the proposed method are set as follows: The learning rate is 0.0001, the batch size is 16, the iteration number N is 150, and the optimizer is SGD. A grid search method is employed to identify the optimal parameter combinations, which are then used for parameter tuning. The model parameter settings for the proposed and compared methods are presented in Table 4.

Table 4. Main parameters of the proposed and comparative methods.

No.	Method	Learning Rate	Epoch	Batch Size	Optimizer	λ	λ_s
1	Proposed Method	0.0001	150	16	SGD	15	0.02
2	2DCNN	0.0001	150	16	SGD	N/A	N/A
3	DANN	0.0001	150	16	SGD	15	N/A
4	DAN	0.0001	150	16	SGD	N/A	N/A
5	MADA	0.0001	150	16	SGD	N/A	0.02
6	Model Domain DANN	0.0001	150	16	SGD	15	0.02
7	Improved DANN	0.0001	150	16	SGD	15	N/A
8	Add Noise	0.0001	150	16	SGD	15	0.02

The proposed method's specific training and testing flow is illustrated in Figure 11. During the training process, the global domain classifier aims to produce domain-invariant features by achieving domain alignment through adversarial training. The simulation data serves as a kind of “pseudo-label” to assist the subdomain classifiers in achieving more precise feature alignment and enhancing generalization performance.

**Figure 11.** Training and testing process of the proposed method.

4.4. Analysis of the Comparison Results

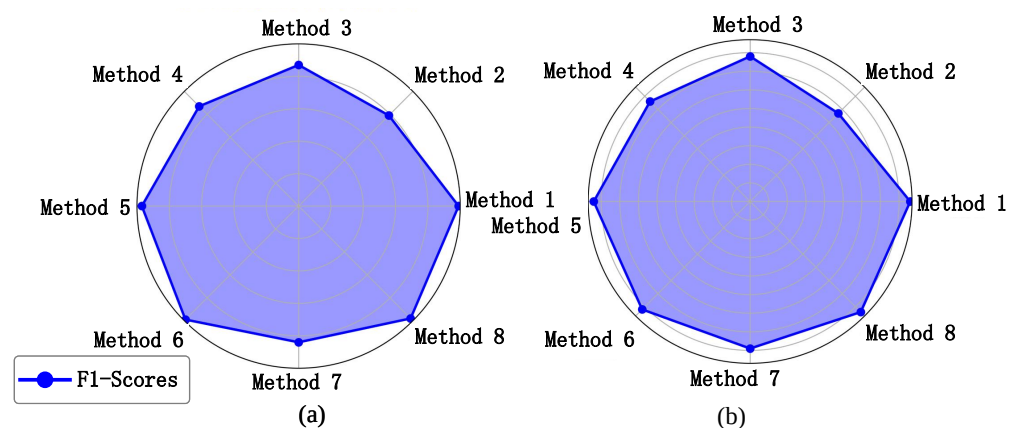
To determine the effectiveness of the proposed method, it was compared with seven alternative methods.

In this paper, we compare the diagnostic effectiveness of each method on the target domain test set using average accuracy (Table 5), average F1 score (Figure 12), accuracy over iterations for different methods (Figure 13), and confusion matrix (Figures 14 and 15).

In the confusion matrix, normal, slight (7 mil), moderate (14 mil), and severe (21 mil) inner-ring faults, ball faults, and outer-ring faults in Case 1 are set to 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9, respectively. In Case 2, the labels for normal, inner-ring faults, and outer-ring fault states are set to 0, 1, and 2, respectively. The abscissa is the predicted label and the ordinate is the real label.

Table 5. Diagnostic results of eight methods.

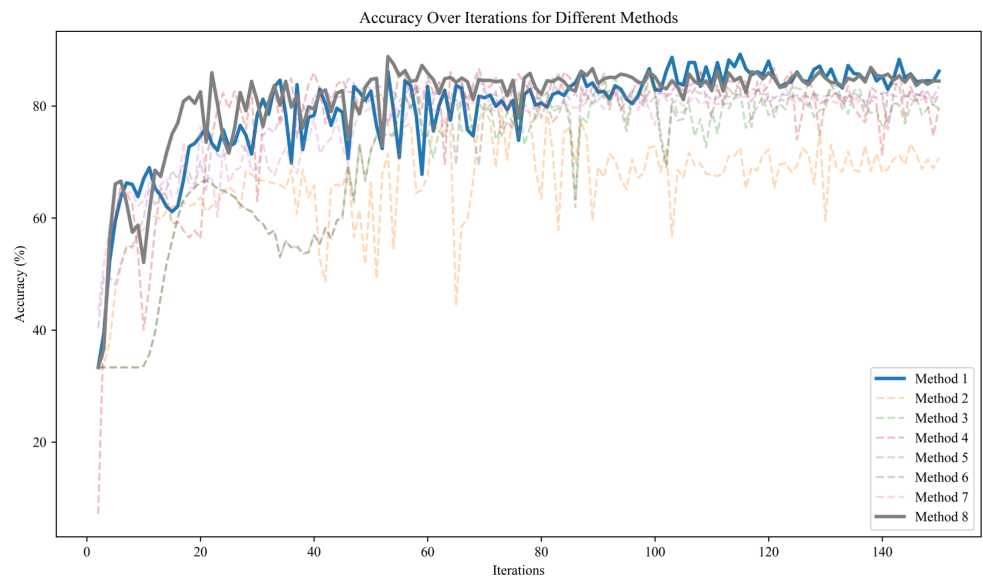
No.	Method	Case 1 (%)	Case 2 (%)
1	Proposed Method	99.435 $\pm$ 0.454	86.667 $\pm$ 0.880
2	2DCNN	77.125 $\pm$ 1.468	67.330 $\pm$ 7.142
3	DANN	86.239 $\pm$ 5.100	79.167 $\pm$ 6.767
4	DAN	86.239 $\pm$ 1.600	80.533 $\pm$ 4.933
5	MADA	98.039 $\pm$ 1.200	83.406 $\pm$ 4.056
6	Model Domain DANN	99.040 $\pm$ 0.820	82.503 $\pm$ 3.267
7	Improved DANN	86.360 $\pm$ 4.030	79.870 $\pm$ 7.535
8	Add Noise	98.827 $\pm$ 1.080	84.334 $\pm$ 2.824

**Figure 12.** Average F1 score for each method in different cases. (a) Case 1. (b) Case 2.

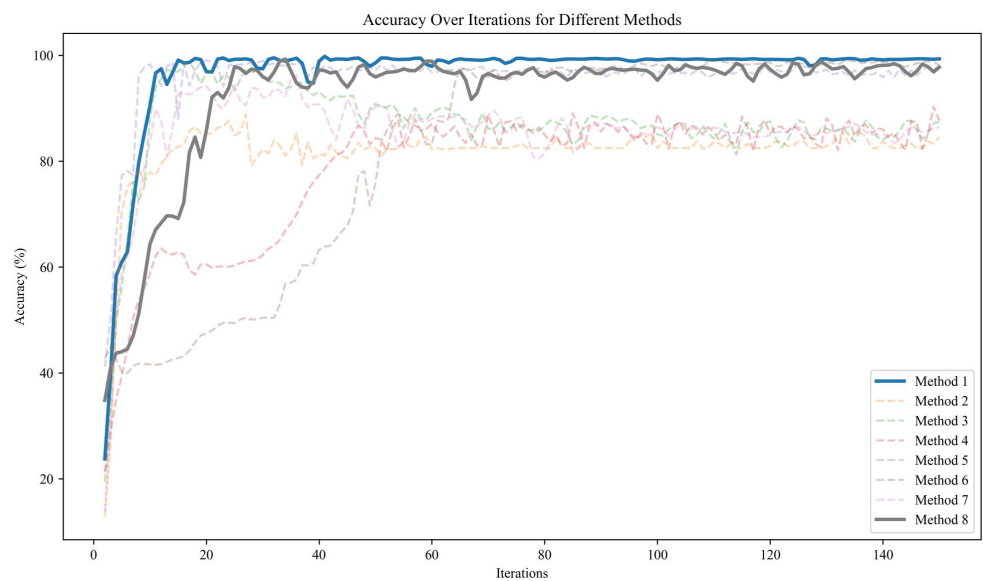
The following main conclusions can be drawn:

- (1) When comparing Method 1 with Methods 2–5, it is evident that Method 1 has a higher average accuracy and a lower standard error of the mean. In unsupervised scenarios, the method of learning demonstrates stronger generalization capabilities. Methods 2 and 3 can be considered models trained on the source domain, directly predicting the target domain. These methods show better post-transfer results in terms of accuracy, F1 score, and other dimensions, reflecting the inhibitory nature of the proposed method on negative transfer. Method 1 achieves an average accuracy of 99.435% and 86.667% in the two scenarios, respectively, while the remaining methods achieve an average accuracy of 98.039% and 83.406%.
- (2) Comparison of Method 1 and Methods 3, 5, 6, and 7 shows that the diagnostic accuracy of the fault categories is improved by the subdomain classifier and the global domain classifier. This is achieved by optimizing the marginal and conditional probability distributions, respectively, and avoiding negative transfers. When comparing Method 7 and Method 3, the accuracy increased from 79.167% to 79.870% in Experiment 1 and from 86.239% to 86.360% in Experiment 2 with the introduction of the MK-MMD metric. This improvement enhances the ability to discriminate inter-domain variability. Method 1 had diagnostic accuracies of 86.667% and 99.435% in both cases, while the highest diagnostic accuracies of the other methods were 83.406% and 99.040%.
- (3) By comparing the unsupervised cross-domain diagnostic iteration accuracies of the eight methods in Cases 1 and 2, it is evident that Method 1 and Method 8, which involves training a neural network with added Gaussian noise, exhibit faster convergence, higher diagnostic accuracy, and greater stability after convergence. The simulation data may be inaccurate for real-world scenarios. However, it is supervised data with a lower-bound guarantee. Adapting samples from the source domain that are too different from the target domain may lead to negative transfers. The proposed

method maximally suppresses negative transfers and facilitates positive transfers by characterizing the common features of the source, target, and simulation domains.



(a)



(b)

Figure 13. Accuracy over iterations for different methods. (a) Cross-domain diagnostic accuracies for the eight methods in one instance of Case 1. (b) Cross-domain diagnostic accuracies for the eight methods in one instance of Case 2.

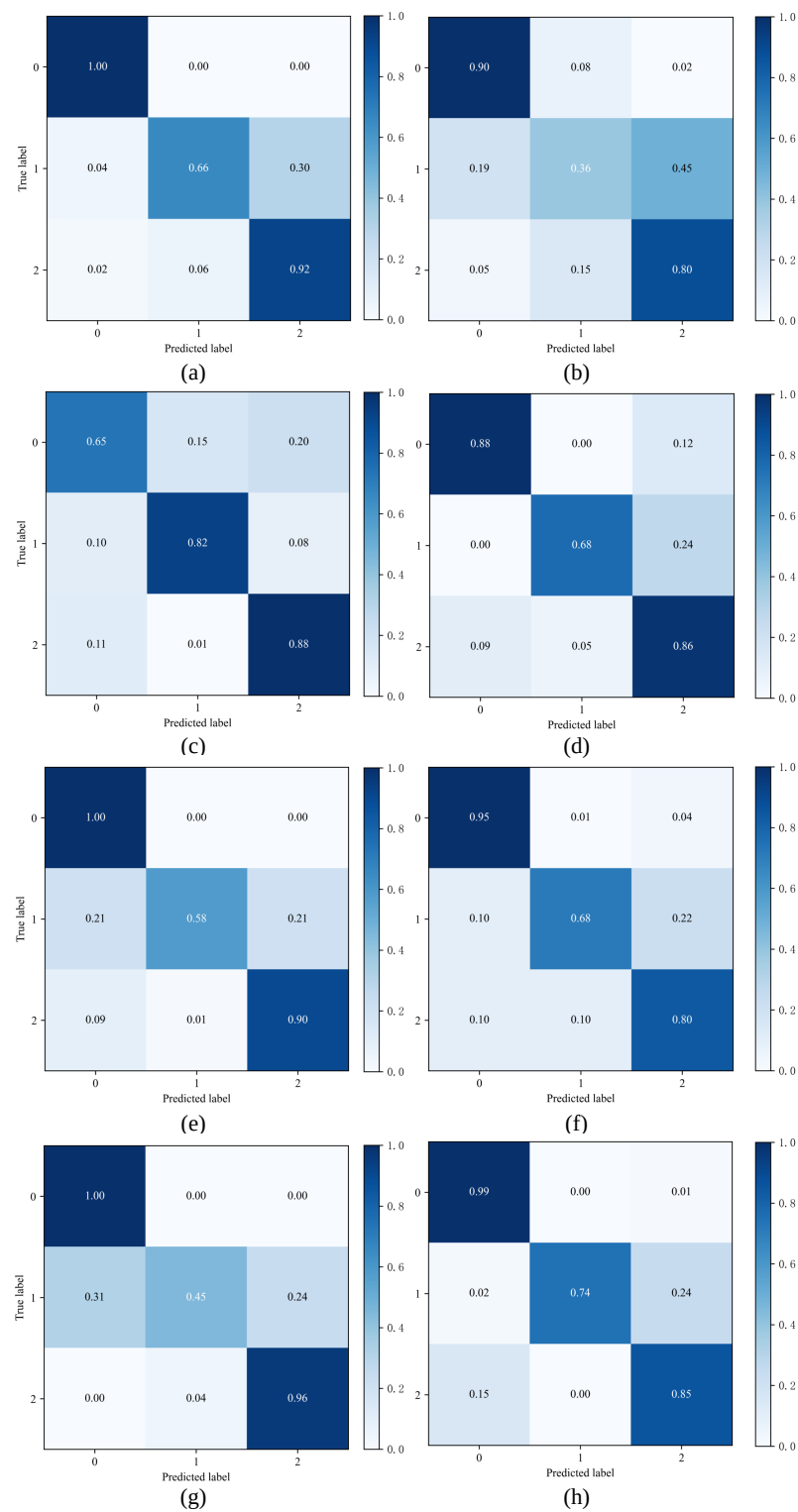


Figure 14. Confusion matrixes for the third trial of each method in Case 1. (a) Method 1. (b) Method 2. (c) Method 3. (d) Method 4. (e) Method 5. (f) Method 6. (g) Method 7. (h) Method 8.

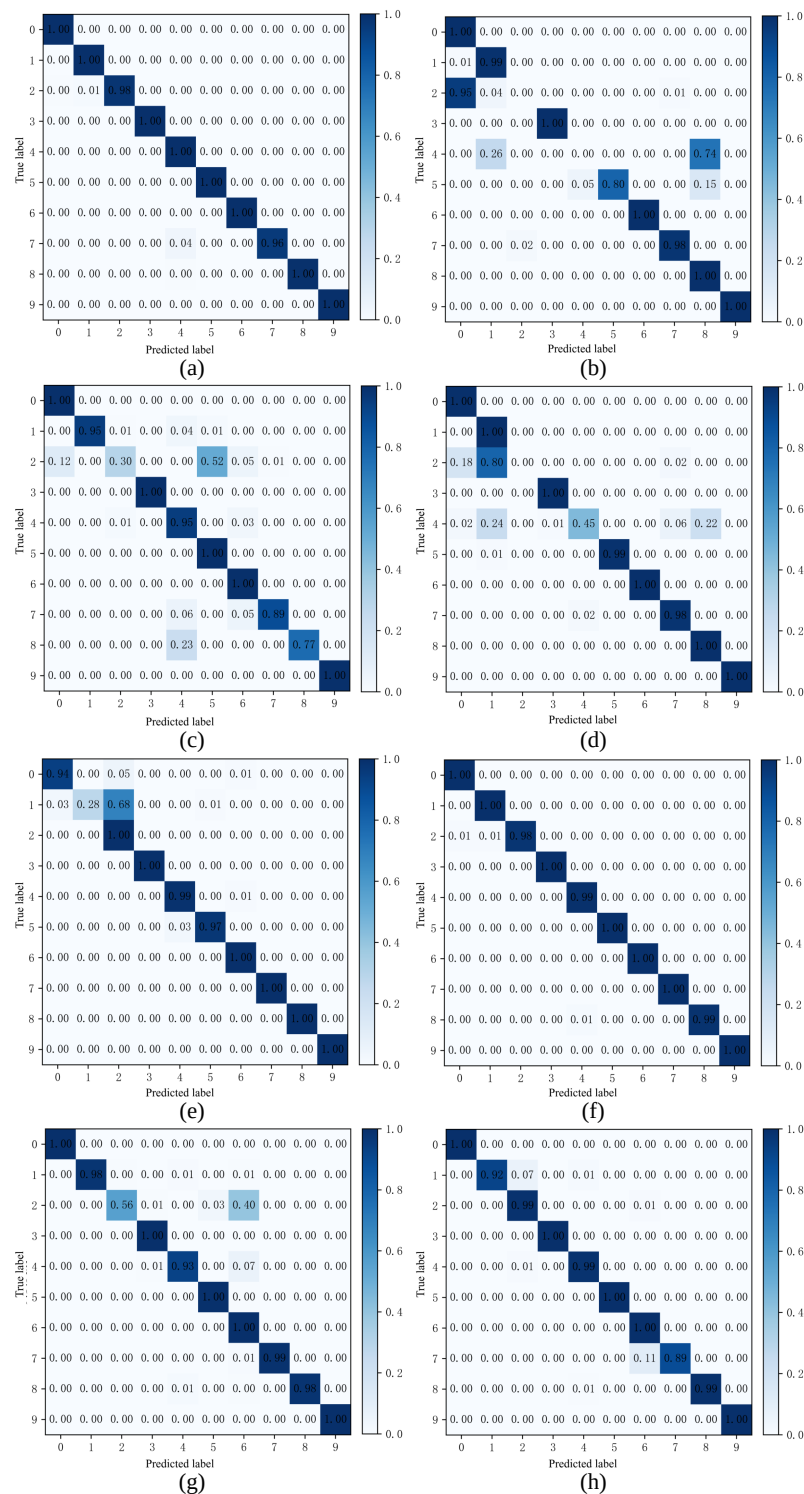


Figure 15. Confusion matrixes for the third trial of each method in Case 2. (a) Method 1. (b) Method 2. (c) Method 3. (d) Method 4. (e) Method 5. (f) Method 6. (g) Method 7. (h) Method 8.

The dimensionality of the output feature vector is reduced using t-distributed stochastic neighborhood embedding (t-SNE) to reflect the feature adaptation performance of the proposed method.

The feature adaptation results of the eight methods for two cases in one experiment are shown in Figures 16 and 17. The label annotation Outer denotes bearing outer-ring

failure, Inner denotes bearing inner-ring failure, Norm denotes normal bearing, and Ball denotes bearing ball failure.

For Case 1, the feature adaptation results for the source and target domains in one experiment of eight methods are shown in Figure 16, where the label prefix S denotes the source domain and the prefix T denotes the target domain.

For Case 2, the faulty bearings in the case are classified into three classes, 1, 2 and 3, according to the size of the faults, and Figure 17 shows the feature adaptation results of the test set in the target domain for one experiment of the eight methods.

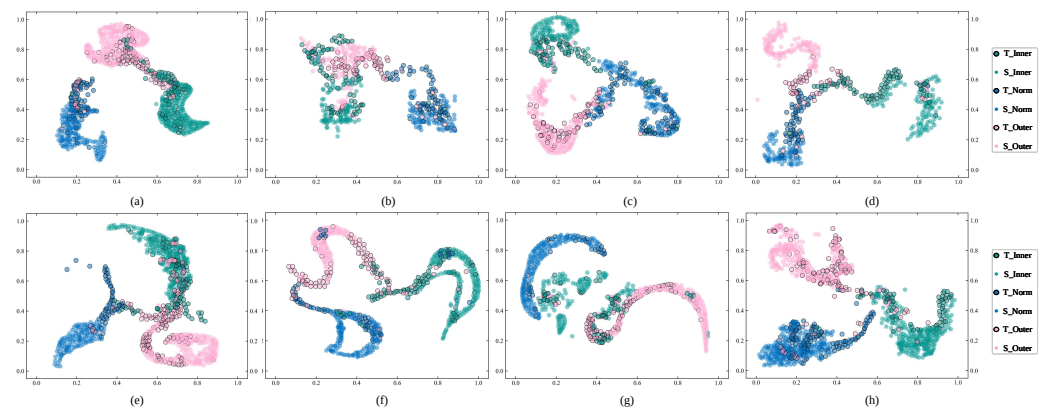


Figure 16. Visual feature adaptation results for each method using t-SNE in Case 1. (a) Method 1. (b) Method 2. (c) Method 3. (d) Method 4. (e) Method 5. (f) Method 6. (g) Method 7. (h) Method 8.

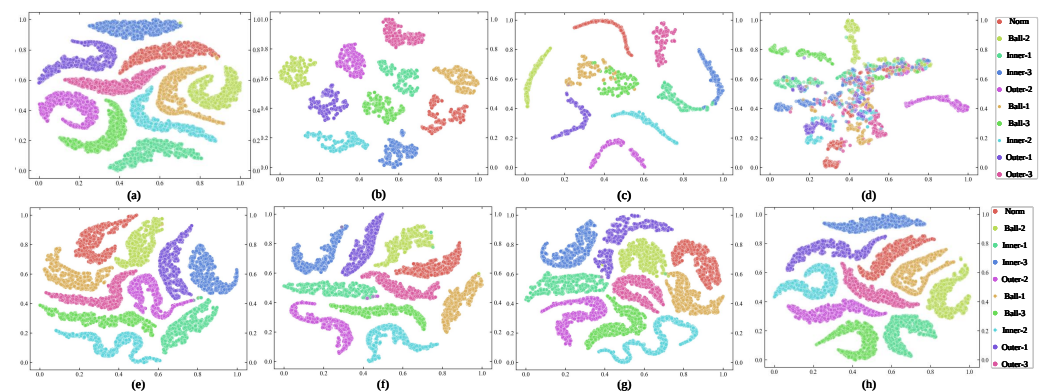


Figure 17. Visual feature adaptation results for each method using t-SNE in Case 2. (a) Method 1. (b) Method 2. (c) Method 3. (d) Method 4. (e) Method 5. (f) Method 6. (g) Method 7. (h) Method 8.

Based on the comparison results, it can be seen that the proposed method shows better results in deep feature extraction common to both source and target domains. This indicates that it has better domain-invariant properties. This is due to the subdomain classifier involved in the simulation data, which achieves features of the same class of samples to be aligned, i.e., facilitates the conditional distribution alignment, not just the source and target domains' indistinguishable feature alignment. Meanwhile, comparing with MADA, the introduction of simulation data avoids relying on the label prediction of the target domain data, which further suppresses the negative transfer.

5. Conclusions

This paper proposes a fault mechanism-assisted multi-domain adversarial transfer fault diagnosis model for rolling bearing fault diagnosis across operating conditions. The aim is to inhibit negative transfer and promote positive transfer to improve the performance of unsupervised domain adaptation rolling bearing fault diagnosis. The method utilizes bearing simulation signals as a benchmark for the domain classifier. It optimizes the align-

ment of conditional and marginal distributions under unsupervised conditions through the subdomain classifier and global domain classifier. This method achieves alignment on fault categories and avoids reliance on labels predicted by the target domain. Through experimental validation, our proposed method achieves better results in the stability of accuracy over multiple training sessions. This proves that our method promotes positive transfer, suppresses negative transfer, and improves the algorithm's generalization ability.

The current need for composite fault diagnosis of bearings is pressing. The next research direction involves simulating the simultaneous occurrence of multiple faults in the simulation data to provide a more comprehensive description of the target domain's characteristics. This approach will facilitate the achievement of composite fault diagnosis.

Author Contributions: Writing—original draft, Z.Z. (Zhidan Zhong) and Z.Z. (Zhihui Zhang); Writing—review and editing, Z.Z. (Zhidan Zhong); Visualization, Y.C.; Project administration, X.X.; Data curation, W.H.; Funding acquisition, Z.Z. (Zhidan Zhong). All authors have read and agreed to the published version of the manuscript.

Funding: This project was supported by the Major Science and Technology Project of Henan Province under Grant 231111222900, the Joint Fund of Science and Technology Research and Development Plan of Henan Province under Grant 232103810038, and the Key Research Projects of Higher Education Institutions of Henan Province under Grant 24A460009.

Data Availability Statement: Experimental data can be downloaded from: <https://engineering.case.edu/bearingdatacenter/apparatus-and-procedures> (accessed on 27 May 2024) and <https://mb.uni-paderborn.de/kat/forschung/datacenter/bearing-datacenter/> (accessed on 27 May 2024).

Conflicts of Interest: Author Wenlu Hao was employed by the company Luoyang Xinqianglian Slewing Bearing Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Lei, Y.; Yang, B.; Jiang, X.; Jia, F.; Li, N.; Nandi, A.K. Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mech. Syst. Signal Process.* **2020**, *138*, 106587. [\[CrossRef\]](#)
2. Ali, J.B.; Fnaiech, N.; Saidi, L.; Chebel-Morello, B.; Fnaiech, F. Application of empirical mode decomposition and artificial neural network for automatic bearing fault diagnosis based on vibration signals. *Appl. Acoust.* **2015**, *89*, 16–27.
3. Lei, Y. *Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery*; Butterworth-Heinemann: Oxford, UK, 2016.
4. Khan, S.; Yairi, T. A review on the application of deep learning in system health management. *Mech. Syst. Signal Process.* **2018**, *107*, 241–265. [\[CrossRef\]](#)
5. Jia, F.; Lei, Y.; Lin, J.; Zhou, X.; Lu, N. Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mech. Syst. Signal Process.* **2016**, *72*, 303–315. [\[CrossRef\]](#)
6. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1989**, *1*, 541–551. [\[CrossRef\]](#)
7. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
8. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
9. Gretton, A.; Sejdinovic, D.; Strathmann, H.; Balakrishnan, S.; Pontil, M.; Fukumizu, K.; Sriperumbudur, B.K. Optimal kernel choice for large-scale two-sample tests. In Proceedings of the Advances in Neural Information Processing Systems 25 (NIPS 2012), Lake Tahoe, NV, USA, 3–6 December 2012; Volume 25.
10. Li, X.; Jiang, H.; Wang, R.; Niu, M. Rolling bearing fault diagnosis using optimal ensemble deep transfer network. *Knowl.-Based Syst.* **2021**, *213*, 106695. [\[CrossRef\]](#)
11. Yu, W.; Zhao, C. Broad convolutional neural network based industrial process fault diagnosis with incremental learning capability. *IEEE Trans. Control Syst. Technol.* **2019**, *28*, 1083–1091. [\[CrossRef\]](#)
12. Yu, W.; Zhao, C. Robust monitoring and fault isolation of nonlinear industrial processes using denoising autoencoder and elastic net. *IEEE Trans. Ind. Electron.* **2019**, *67*, 5081–5091. [\[CrossRef\]](#)
13. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [\[CrossRef\]](#)
14. Kiral, Z.; Karagülle, H. Vibration analysis of rolling element bearings with various defects under the action of an unbalanced force. *Mech. Syst. Signal Process.* **2006**, *20*, 1967–1991. [\[CrossRef\]](#)

15. McFadden, P.; Smith, J. Model for the vibration produced by a single point defect in a rolling element bearing. *J. Sound Vib.* **1984**, *96*, 69–82. [\[CrossRef\]](#)
16. Sassi, S.; Badri, B.; Thomas, M. A numerical model to predict damaged bearing vibrations. *J. Vib. Control* **2007**, *13*, 1603–1628. [\[CrossRef\]](#)
17. Brie, D. Modelling of the spalled rolling element bearing vibration signal: An overview and some new results. *Mech. Syst. Signal Process.* **2000**, *14*, 353–369. [\[CrossRef\]](#)
18. Wang, C.; Tian, J.; Zhang, F.L.; Ai, Y.T.; Wang, Z. Dynamic modeling and simulation analysis of inter-shaft bearing fault of a dual-rotor system. *Mech. Syst. Signal Process.* **2023**, *193*, 110260. [\[CrossRef\]](#)
19. Liu, H.B.; Zhang, L.; Shi, Y.S. Dynamic finite element analysis for tapered roller bearings. *Appl. Mech. Mater.* **2014**, *533*, 21–26. [\[CrossRef\]](#)
20. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
21. Yu, C.; Wang, J.; Chen, Y.; Huang, M. Transfer learning with dynamic adversarial adaptation network. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 778–786.
22. Pei, Z.; Cao, Z.; Long, M.; Wang, J. Multi-adversarial domain adaptation. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
23. Zhu, Y.; Zhuang, F.; Wang, J.; Ke, G.; Chen, J.; Bian, J.; Xiong, H.; He, Q. Deep subdomain adaptation network for image classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 1713–1722. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
25. Lessmeier, C.; Kimotho, J.K.; Zimmer, D.; Sextro, W. Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification. *PHM Soc. Eur. Conf.* **2016**, *3*, 1–17. [\[CrossRef\]](#)
26. These Data Comes from Case Western Reserve University Bearing Data Center Website. Available online: <http://www.eecs.cwru.edu/laboratory/bearings/> (accessed on 27 May 2024).
27. Yu, W.; Zhao, C.; Huang, B.; Xie, M. An Unsupervised Fault Detection and Diagnosis With Distribution Dissimilarity and Lasso Penalty. *IEEE Trans. Control Syst. Technol.* **2023**, *32*, 767–779. [\[CrossRef\]](#)
28. Yu, W.; Zhao, C.; Huang, B. MoniNet with concurrent analytics of temporal and spatial information for fault detection in industrial processes. *IEEE Trans. Cybern.* **2021**, *52*, 8340–8351. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.