

Article

Multi-Structure Fusion Conditional Domain Adaptation for Bearing Fault Diagnosis

Cairong Li and Jianan Wang *

School of Electronics Information Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China; s202315210523@stu.tyust.edu.cn

* Correspondence: jiananwang@tyust.edu.cn; Tel.: +86-18735189968

Abstract

To address the limited feature extraction capability of single-architecture deep models, pronounced distribution discrepancies in bearing vibration signals, and instability during cross-domain training under varying operating conditions, a multi-structure fusion conditional domain adaptation (MSFCDA) method is proposed for bearing fault diagnosis. First, a multi-structure feature extractor is constructed by integrating a multi-channel multi-scale convolutional neural network, a bidirectional long short-term memory network, and an efficient channel attention mechanism to comprehensively extract multi-scale features from vibration signals. On this basis, a class-conditional feature distribution alignment strategy is developed, in which class prediction information is coupled with the cross-domain alignment process to achieve feature alignment under class-conditional constraints. In addition, a dynamic adversarial weighting factor is introduced to adaptively regulate the strength of adversarial training, thereby balancing feature discriminability and domain invariance. Extensive cross-condition transfer experiments are conducted on the publicly available CWRU and PU bearing datasets. The results demonstrate that the proposed method consistently achieves high diagnostic accuracy under different operating conditions, validating its strong cross-domain transfer capability and training stability under variable operating conditions.

Keywords: rolling bearing; fault diagnosis; varying operating conditions; domain adaptation; transfer learning

1. Introduction

With the increasing scale of industrial systems and the growing complexity of operating environments, rotating machinery is often operated continuously under high loads and diverse operating conditions. The reliability of such equipment directly affects production efficiency and product quality. As a fundamental component of rotating machinery, bearings are widely used in industrial applications such as wind power equipment, aero-propulsion systems, and rail transportation, and their health condition is directly related to industrial safety and reliability [1,2]. If they cannot be identified and detected at an early stage, they may readily lead to performance degradation and excessive component wear and, in severe cases, result in major safety accidents and substantial economic losses [3]. Therefore, high-accuracy fault diagnosis of rolling bearings under variable operating conditions is of considerable theoretical and engineering significance [4,5].

In recent years, deep learning has been widely applied in mechanical fault diagnosis owing to its powerful feature extraction capability, eliminating the reliance on manually



Academic Editor: Ahmed Abu-Siada

Received: 4 September 2026

Revised: 9 October 2026

Accepted: 12 October 2026

Published: 14 October 2026

Copyright: © 2026 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

extracted features in traditional methods [6]. Yu et al. [7] fused variational mode decomposition, continuous wavelet transform, and CNN for intelligent bearing fault identification. Yan et al. [8] applied an ordered neurons LSTM neural network to long-term gear life prediction. Guo et al. [9] proposed an end-to-end attention CNN and BiLSTM network for fault diagnosis, improving model generalization. Sui et al. [10] used time-frequency images with an improved CNN to overcome manual feature extraction and insufficient temporal-channel deep feature learning in traditional methods. Liu et al. [11] developed a feature pyramid network with a convolutional recurrent neural network for diagnosing complex vibration signals under varying loads and operating conditions.

Although deep learning methods have achieved promising results in fault identification, they generally rely heavily on sufficient labeled samples. In real industrial scenarios, collecting and labeling fault samples is difficult and costly. Meanwhile, changes in operating conditions can cause shifts in data distributions, leading to a sharp degradation in diagnostic performance. Domain-adaptive transfer learning provides a feasible solution to distribution shifts by transferring fault knowledge learned from a source domain to an unlabeled target domain through cross-domain feature alignment and has therefore been widely applied to fault diagnosis under varying operating conditions. Li et al. [12] proposed a method combining adversarial training with correlation alignment to learn common features across different domains. Wan et al. [13] employed an improved deep residual network to extract informative features and combined domain adaptation with multi-kernel maximum mean discrepancy to achieve cross-domain fault diagnosis. Wang et al. [14] developed a multi-scale intra-class domain-adaptive deep transfer learning model to align cross-domain feature distributions corresponding to the same fault. Li et al. [15] proposed a deep convolutional domain adversarial model that adversarially trains on labeled source and unlabeled target domains to reduce domain discrepancy and boost diagnostic transfer. Han et al. [16] proposed an adversarial learning framework incorporating an additional discriminator and combining discrepancy measurement with adversarial training for cross-domain fault diagnosis. Jiao et al. [17] proposed a multilayer adversarial network for fault diagnosis that simultaneously employed a domain discriminator and discrepancy measurement, thereby improving fault identification accuracy.

Existing domain adaptive transfer diagnosis methods have made certain progress, yet several issues remain unresolved. (1) Most networks adopt single-scale feature extraction structure, making it difficult to simultaneously capture impulsive fault signatures and long-term temporal evolution in bearing vibration signals. (2) Many methods align only the global feature distribution without explicitly incorporating fault-class information, which can result in feature overlap between different fault categories. (3) Most domain adversarial training methods employ a fixed adversarial weight, excessively strong adversarial constraints in the early training stage can interfere with the learning of fault-discriminative features, causing convergence oscillations and training instability.

To address these issues, this study proposes a multi-structure fusion conditional domain adaptation (MSFCDA) method for bearing fault diagnosis. The main contributions of this study are summarized as follows:

- A multi-channel multi-scale parallel convolutional architecture were constructed to extract fault information from different frequency bands, while an efficient channel attention mechanism was introduced to enhance informative channel features. A bidirectional long short-term memory network is subsequently employed to further capture the temporal characteristics of vibration signals.
- To reduce the distribution discrepancy between source and target domains while preserving inter-class separability, we introduce a class-conditional feature distribution alignment strategy.

- A dynamic adversarial training mechanism with a gradient reversal layer was designed to adaptively regulate the strength of domain adversarial training, thereby balancing fault discriminability and domain invariance.

2. Related Work

2.1. Convolutional Neural Network

A convolutional neural network (CNN) is a deep feedforward neural network commonly used to process one-dimensional time-series signals or images [18]. Compared with conventional neural networks, CNN exhibits strong capability of automatic feature extraction and reduces the number of parameters through weight sharing and sparse connectivity, thereby alleviating overfitting during training. A CNN generally consists of an input layer, convolutional layers, pooling layers, fully connected layers, and an output layer, as illustrated in Figure 1.

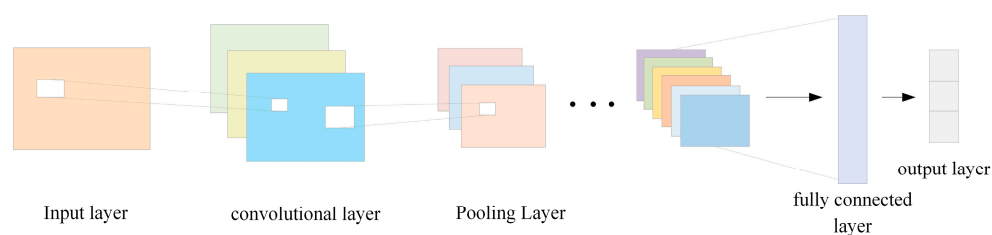


Figure 1. Convolutional neural network architecture diagram.

2.2. Bidirectional Long Short-Term Memory Network

A long short-term memory (LSTM) network consists of a forget gate f_t , an input gate i_t , an output gate O_t , and a memory cell C_t [19]. Through these three gating mechanisms, the LSTM selectively retains or discards time-dependent information, effectively mitigating long-term dependency issues. The LSTM architecture is shown in Figure 2.

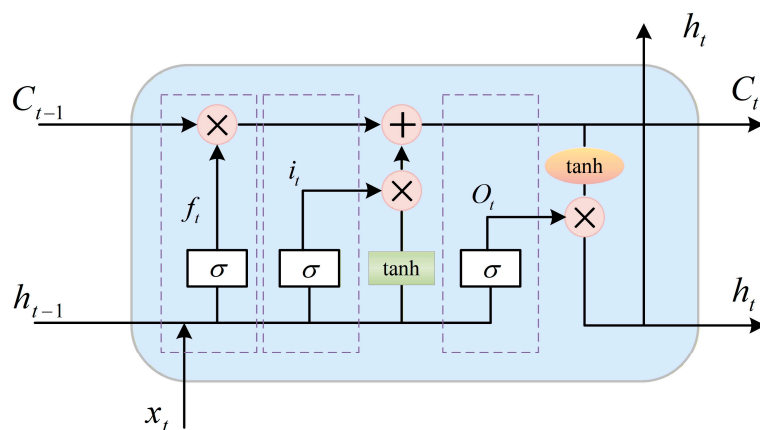


Figure 2. Long short-term memory network structure diagram.

LSTM processes information only in temporal order and therefore has difficulty capturing dependencies in both past and future directions. A bidirectional long short-term memory (BiLSTM) network employs two LSTMs operating in opposite directions to process forward and backward information in the input sequence simultaneously. It thus integrates temporal features from both past and future sequences and improves the overall learning capability of the network. The BiLSTM architecture is shown in Figure 3.

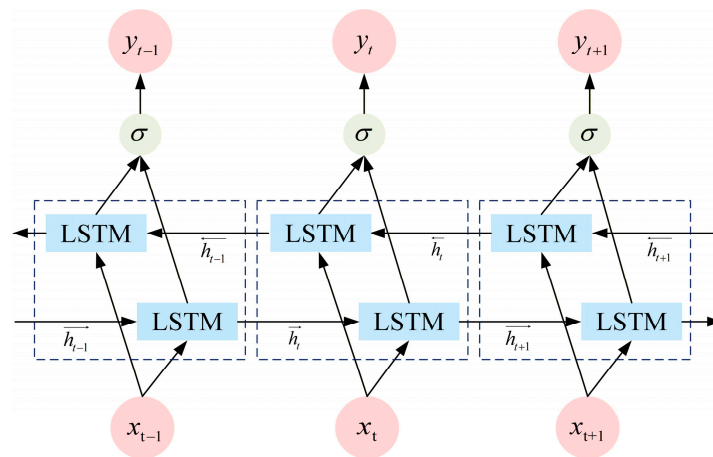


Figure 3. Bidirectional long short-term memory network structure diagram.

The BiLSTM working procedure is formulated as in Equations (1)–(3):

$$\vec{h}_{t+1} = LSTM \left[x_t, \vec{h}_t \right] \quad (1)$$

$$\overleftarrow{h}_{t+1} = LSTM \left[x_t, \overleftarrow{h}_t \right] \quad (2)$$

$$y_t = \sigma \left[W_Y \left[\vec{h}_{t+1}, \overleftarrow{h}_{t+1} \right] + b_Y \right] \quad (3)$$

where $\vec{h}_t$ denotes the hidden state at the forward time step, and $\overleftarrow{h}_t$ denotes the hidden state at the backward time step. W_Y and b_Y represent the weight matrix and the bias term, respectively.

2.3. Efficient Channel Attention Mechanism

During deep feature extraction, convolutional feature channels contain substantial redundant information, which can interfere with the model's learning of effective fault features. The efficient channel attention (ECA) mechanism adaptively learns channel weights through local cross-channel interactions without dimensionality reduction, enhancing informative feature channels with only a small number of additional parameters while suppressing interference from irrelevant channels [20], as illustrated in Figure 4.

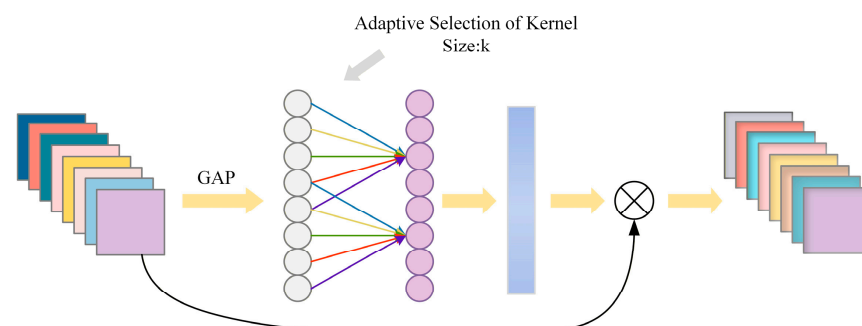


Figure 4. Efficient channel attention mechanism.

The adaptive function for kernel size k is given by:

$$k = \left\lceil \frac{\log_2 C}{\gamma} + \frac{b}{\gamma} \right\rceil \quad (4)$$

where C is the number of input feature channels, γ is the scale parameter controlling inter-channel relationships, and b is the scaling factor for the kernel size, with $\gamma = 1$ and $b = 2$.

2.4. Domain Adaptation

The central objective of domain adaptation is to learn a domain-invariant feature representation that preserves fault-discriminative information while reducing the feature distribution discrepancy between the source and target domains. The domain adaptation process is illustrated in Figure 5. Through feature distribution alignment, samples belonging to the same fault class are encouraged to become close to each other in the feature space, while samples from different fault classes remain separated. Distribution alignment reduces the source–target discrepancy via adversarial training or moment matching, whereas subspace calibration aligns the underlying subspace bases. In this paper, we adopt distribution alignment, which conditions samples on class posterior probabilities to preserve discriminability. Unlike subspace calibration, it does not explicitly align subspace bases but implicitly learns domain-invariant characteristics. Its limitation lies in reliance on the reliability of class posterior probabilities, especially in early training or under strong noise, and the lack of explicit subspace geometric consistency [21].

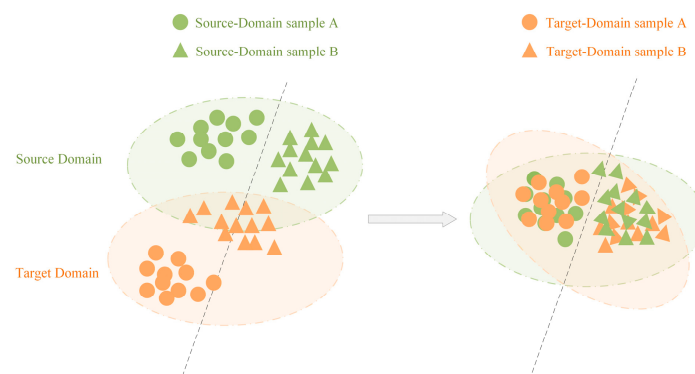


Figure 5. Domain adaptation process.

3. Methods

This study proposes an MSFCDA bearing fault diagnosis model based on multi-structure fusion conditional domain adaptation, as illustrated in Figure 6. The overall architecture consists of four components: a multi-structure feature extractor (multi-channel multi-scale convolutional neural network, BiLSTM and ECA, MMCBE), a fault classifier, a domain discriminator, and a gradient reversal layer (GRL). The source and target samples share all network parameters of the feature extractor.

Firstly, raw vibration signals from both the source and target domains are fed into the shared MMCBE feature extractor for hierarchical deep feature extraction. Labeled source-domain samples are used to supervise the training of the fault classifier, producing posterior probabilities for the fault classes. To alleviate the class-confusion problem caused by global distribution alignment, a class-conditional feature distribution alignment strategy is constructed by coupling the predicted class posterior probabilities with the extracted features through an outer product, thereby incorporating class-prior information into domain alignment. The domain discriminator determines the domain origin of each sample based on the resulting joint features, while the GRL establishes adversarial interaction between the feature extractor and the domain discriminator. A dynamic adversarial weighting factor driven by cosine annealing is further introduced to adaptively regulate the strength of adversarial training. The model therefore emphasizes learning fault-discriminative features

during the early training stage and progressively strengthens cross-domain distribution alignment at later stages, alleviating training oscillations and convergence instability under varying operating conditions. Finally, the model jointly optimizes the classification loss and dynamically weighted domain-adversarial loss to enable bearing fault identification in an unlabeled target domain.

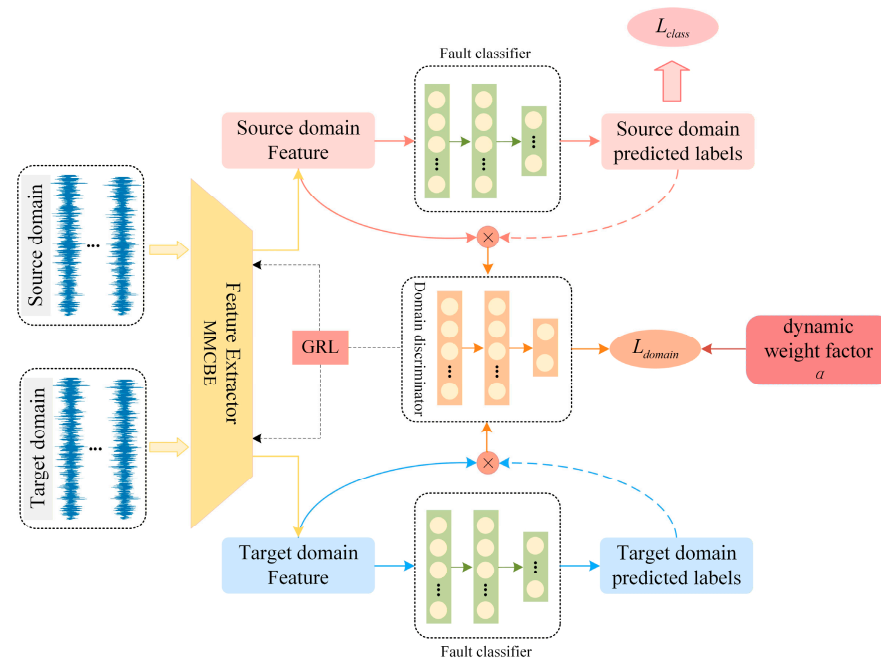


Figure 6. Fault diagnosis model based on MSFCDA.

3.1. Feature Extraction Module

The MMCBE multi-structure feature extraction module is developed by integrating a multi-channel multi-scale convolutional neural network, a BiLSTM, and an ECA mechanism; its architecture is depicted in Figure 7.

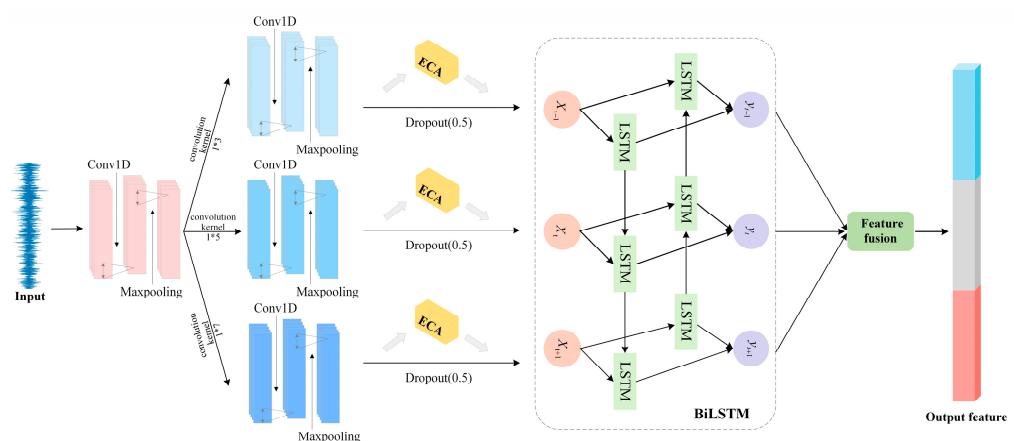


Figure 7. MMCBE structural diagram.

The module takes raw time-domain vibration signals of length 2048 as input. To preserve the overall trend and enlarge the initial receptive field at the input stage, a large 1×128 convolution kernel is applied for one-dimensional convolution, enabling the extraction of broader and more global features. The initially extracted features are then fed into the MMCNN for finer-grained analysis. Parallel small convolution kernels of 1×3 , 1×5 , and 1×7 are employed to model local features under different receptive fields. The smaller kernels focus more strongly on local and fine-grained features, thereby enriching

the diversity of the extracted representations. After the parallel convolution operations, the outputs sequentially pass through batch normalization to stabilize feature distributions, the ReLU activation function to alleviate gradient vanishing, and max pooling layer to further enhance translation invariance. A dropout layer is then applied within each branch. The convolutional features extracted by the different branches are subsequently fed into the BiLSTM, which captures the temporal characteristics of the fault evolution process from both forward and backward directions. To further suppress redundant information and emphasize informative features, an ECA attention mechanism is incorporated into each branch. Finally, the features extracted from the three branches with different scales are fused, and dropout is applied to randomly deactivate a subset of neurons, reducing excessive reliance on local features and mitigating overfitting.

3.2. Class-Conditioned Feature Distribution Alignment Strategy

Conventional domain-adversarial methods mainly perform global feature distribution alignment without explicit class constraints, which may cause features from different fault categories to overlap. This study introduces class-conditional constraints into the feature alignment process to construct a class-conditional feature distribution alignment strategy, thereby achieving both class discriminability and domain invariance in the learned representations. The formulation is defined as follows:

$$L_{class} = -\frac{1}{N_s} \sum_{i=1}^{N_s} \sum_{k=1}^K y_i^k \log(\hat{y}_i^k) \quad (5)$$

where N_s denotes the number of source-domain samples, K represents the number of fault categories (with $K = 10$ in this study), y_i^k denotes the ground-truth label of the i -th source sample for class k , which takes a value of 1 if the sample i belongs to class k and 0 otherwise, and $\hat{y}_i^k$ denotes the predicted probability by the fault classifier that the sample i belongs to class k .

The features extracted by the feature extractor and the class prediction probabilities output by the fault classifier are jointly modeled to construct a class-conditional joint representation, which is defined as follows:

$$z_i = f_i \otimes \hat{y}_i \quad (6)$$

where f_i denotes the feature vector extracted by the feature extractor for the i -th sample, and $\hat{y}_i$ is the class prediction probability vector output by the fault classifier. $\otimes$ denotes the outer product operation, which couples the sample's feature representation with class prediction information to capture high-order interactions between them, enabling the domain discriminator to distinguish the domain properties of samples under class-conditional constraints.

To align the source and target distributions under class-conditional constraints, the joint representations are fed into the domain discriminator for domain classification. The domain loss for source-domain samples is computed as follows:

$$L_{source_domain} = -\frac{1}{N_s} \sum_{i=1}^{N_s} \{\log[D(z_i^s)]\} \quad (7)$$

where $D(\cdot)$ denotes the probability that the domain discriminator outputs for a sample to originate from the source domain, and z_i^s represents the joint representation of the features and predicted probabilities for the i -th source-domain sample. The domain loss for

target-domain samples is calculated in a manner similar to that for source-domain loss; the domain discrimination loss for target-domain samples is then formulated as follows:

$$L_{target_domain} = -\frac{1}{N_t} \sum_{j=1}^{N_t} \left\{ \log \left[1 - D(z_j^t) \right] \right\} \quad (8)$$

The overall domain discrimination loss can be expressed as follows:

$$L_{domain} = -\frac{1}{N_s} \sum_{i=1}^{N_s} \log D(z_i^s) - \frac{1}{N_t} \sum_{j=1}^{N_t} \log \left(1 - D(z_j^t) \right) \quad (9)$$

The domain-discrimination loss is used to optimize the domain discriminator so that it can distinguish source-domain samples from target-domain samples as accurately as possible.

3.3. Dynamic Adversarial Training Mechanism with Gradient Reversal Layer

Under varying operating conditions, imposing excessively strong adversarial constraints at the early training stage can interfere with the establishment of the fault discrimination structure and affect convergence stability and diagnostic performance. To address this issue, a dynamic adversarial training mechanism is designed, which adaptively adjusts the domain-adversarial intensity to achieve a balanced optimization between fault classification learning and cross-domain alignment. The GRL is positioned between the feature extractor and the domain discriminator. During forward propagation, it passes features identically, whereas during backward propagation, it reverses the sign of the gradient from the domain discriminator. Consequently, the feature extractor is updated toward confusing the domain discriminator and consequently obtaining to a more domain-invariant feature representation. Through this adversarial process, the model maintains source-domain classification capability while effectively alleviating distribution shifts caused by operating condition variations.

To prevent overly strong adversarial constraints at the early training stage from disrupting the stable learning of fault-discriminative features, a dynamic adversarial weighting factor based on a cosine annealing strategy is introduced to adaptively adjust the adversarial intensity of the GRL during training, and it is defined as follows:

$$\lambda(e) = 0.5\alpha_{\max} \cdot \left(1 - \cos \left(\pi \cdot \left(\frac{e}{E} \right) \right) \right) \quad (10)$$

where e denotes the current training epoch; E is the total number of training epochs, $\lambda(e)$ is the adversarial intensity coefficient of the GRL at the e -th epoch, and $\alpha_{\max}$ is a prespecified maximum adversarial weight used to limit the upper bound of the adversarial gradient.

This strategy assigns a relatively small adversarial weight during the early training stage, enabling the network to prioritize the learning of discriminative feature structures. As training progresses, the consistency of feature distributions under varying operating conditions is gradually enhanced. Finally, under the joint action of the GRL and the dynamic adversarial factor, the domain discriminator and feature extractor undergo a progressive adversarial process from weak to strong, allowing the model to smoothly transition from fault-discriminative learning to cross-condition distribution alignment.

3.4. Overall Loss and Optimization Objective

A dynamically weighted loss function is designed to balance the optimization of fault classification and cross-domain alignment tasks. The total loss function consists of the fault

classification loss and the dynamically weighted domain-adversarial loss, and is formulated as follows:

$$L_{total} = L_{class} + \alpha L_{domain} \quad (11)$$

where α is the dynamic weighting factor. During training, the dynamic weighting factor is adjusted according to the cosine annealing strategy to balance classification accuracy and domain alignment. The relevant parameters of the model are updated as follows:

Feature extractor parameter update:

$$\theta_f \leftarrow \theta_f - \eta \left(\frac{\partial L_{class}}{\partial \theta_f} - \alpha \frac{\partial L_{domain}}{\partial \theta_f} \right) \quad (12)$$

Fault classifier parameter update:

$$\theta_c \leftarrow \theta_c - \eta \frac{\partial L_{class}}{\partial \theta_c} \quad (13)$$

Domain discriminator parameter update:

$$\theta_d \leftarrow \theta_d - \eta \frac{\partial L_{domain}}{\partial \theta_d} \quad (14)$$

where η is the learning rate, and θ_f , θ_c , and θ_d are the parameters of the feature extractor, the fault classifier, and the domain discriminator, respectively.

To facilitate effective knowledge transfer across different operating conditions, a learning rate decay strategy is introduced during model training to improve training stability. The learning rate is gradually decreased over the training progress; the expression is given as follows:

$$\eta_p = \frac{\eta_0}{1 + (\gamma \cdot p)^\beta} \quad (15)$$

where η_0 is the initial learning rate, p is the current relative training progress, computed as the ratio of the current iteration number to the total number of iterations, and γ and β are hyperparameters controlling the decay rate of the learning rate.

4. Results

4.1. Dataset Introduction and Settings of Transfer Tasks

4.1.1. Case Western Reserve University Dataset

(1) Dataset Introduction

The Case Western Reserve University (CWRU) bearing dataset is employed for validation. The experimental setup for the CWRU dataset mainly consists of an electric motor, a torque measurement device, a dynamometer, and a data acquisition system. A torque sensor is installed at the drive end of the motor to collect vibration signals from an SKF6205 bearing under different operating conditions. Data sampled at 12 kHz are used in this study. The motor loads are 0, 1, 2, and 3 hp, corresponding to motor speeds of approximately 1797, 1772, 1750, and 1730 rpm, respectively. The dataset is divided into four operating conditions, A0, A1, A2, and A3. The main bearing fault types include normal condition, ball fault, inner-race fault, and outer-race fault. Each fault type has three damage sizes of 0.007, 0.014, and 0.021 inch, resulting in 10 fault categories in total. Each category contains 500 samples, and each sample consists of 2048 data points. The dataset partition is provided in Table 1.

Table 1. CWRU bearing fault category information.

Operating Condition	Load	Label	Fault ID	Fault Types	Fault Size	Training/Test
A0/A1/A2/A3	0/1/2/3	0	NO	Normal	—	350/150
		1	IR007	Inner Race Fault	0.007	350/150
		2	B007	Ball Fault	0.007	350/150
		3	OR007	Outer Race Fault	0.007	350/150
		4	IR014	Inner Race Fault	0.014	350/150
		5	B014	Ball Fault	0.014	350/150
		6	OR014	Outer Race Fault	0.014	350/150
		7	IR021	Inner Race Fault	0.021	350/150
		8	B021	Ball Fault	0.021	350/150
		9	OR021	Outer Race Fault	0.021	350/150

(2) Settings of Transfer Tasks

The operating conditions A0, A1, A2 and A3 are regarded as four independent domains. In each transfer setting, only one operating condition is used as the labeled source domain for training, and the learned fault-discriminative knowledge is transferred to a different operating condition used as the unlabeled target domain for testing. Accordingly, twelve transfer tasks T1–T12 are constructed, including A0→A1, A0→A2, A0→A3, A1→A0, A1→A2, A1→A3, A2→A0, A2→A1, A2→A3, A3→A0, A3→A1, and A3→A2. The source domain is labeled, whereas the target domain is unlabeled.

4.1.2. Paderborn University Dataset

(1) Dataset Introduction

Another dataset used in the experiments is the Paderborn University (PU) bearing fault dataset from Germany. The experimental setup mainly consists of a drive motor, a torque meter shaft, a rolling bearing test module, a flywheel, and a load motor. Vibration signals from SKF6203 bearings under different operating conditions are collected. The faulty bearings include artificially damaged and naturally damaged bearings; the naturally damaged vibration signals are selected for this study. The vibration signals are sampled at 64 kHz, the mechanical parameters at 4 kHz, and temperature at 1 Hz. The dataset is divided into four operating conditions, B0, B1, B2, and B3, according to different rotational speeds, load torques, and radial forces. The parameters are listed in Table 2.

Table 2. PU dataset: four operating condition parameters.

Operating Condition	Rotational Speed/rpm	Radial Force/N	Load Torque/(N·m)	Name
B0	1500	1000	0.7	N15_M07_F10
B1	900	1000	0.7	N09_M07_F10
B2	1500	1000	0.1	N15_M01_F10
B3	1500	400	0.7	N15_M07_F04

The major bearing fault types under each operating condition mainly include normal condition, inner race fault, outer race fault, and compound fault. Each fault type corresponds to different damage types, including single-point damage, multiple damage, and repetitive damage, resulting in a total of 10 fault categories. Each category consists of 500 samples, with each sample containing 2048 data points. The dataset partitioning is summarized in Table 3.

Table 3. PU bearing fault category information.

Operating Condition	Label	Fault ID	Fault Types	Damage Severity	Damage Type	Training/Test
B0/B1/B2/B3	0	K004	Normal	—	—	350/150
	1	KA04	Outer-race fault 1	1	Single-point damage	350/150
	2	KA15	Outer-race fault 2	1	Single-point damage	350/150
	3	KA16	Outer-race fault 3	2	Repetitive damage	350/150
	4	KA22	Outer-race fault 4	1	Single-point damage	350/150
	5	KI16	Inner-race fault 1	3	Single-point damage	350/150
	6	KI17	Inner-race fault 2	1	Repetitive damage	350/150
	7	KI18	Inner-race fault 3	2	Single-point damage	350/150
	8	KI21	Inner-race fault 4	1	Single-point damage	350/150
	9	KB24	Compound fault	3	Multiple-point damage	350/150

(2) Setting of Transfer Tasks

The operating conditions B0, B1, B2, and B3 are regarded as four independent domains. In each transfer setting, only one operating condition is used as the source domain for training, and the learned fault-discriminative knowledge is transferred to another operating condition as the target domain for testing. Twelve transfer tasks, T1–T12, are constructed: B0→B1, B0→B2, B0→B3, B1→B0, B1→B2, B1→B3, B2→B0, B2→B1, B2→B3, B3→B0, B3→B1, and B3→B2. The source domain is labeled; the target domain is unlabeled.

4.2. Subsection Data Preprocessing and Experimental Setup

To increase the diversity of training and testing samples, a sliding-window segmentation strategy with an overlap ratio of 0.5 is applied to the vibration signals. The original continuous vibration signal of each fault class is first divided into a training segment and a testing segment in chronological order at a ratio of 7:3. The overlapping sliding-window segmentation is then performed independently within the training and testing segments. The training set contains source-domain and target-domain samples, whereas the test set contains only target-domain samples, thereby simulating the realistic scenario of unlabeled fault diagnosis under a new operating condition. During training, source-domain samples contribute to both the classification loss and domain-adversarial loss, whereas target-domain samples contribute only to the domain-adversarial loss and are not used for supervised classification training. To reduce the influence of random factors on the experimental results, the model is trained ten times independently under the same data partition. All reported accuracy values are expressed as the mean $\pm$ standard deviation over the ten independent runs. The total number of training epochs is 100, the learning rate is 0.0001, the weight decay is $1e-5$, and the Adam optimizer is employed.

The experimental environment configuration is shown in Table 4. Conducting experiments under a unified environment effectively avoids the influence of hardware and software discrepancies on the experimental results and ensures the controllability and reproducibility of model performance evaluation.

Table 4. Experimental environment configuration.

Experimental Environment	Specific Parameters
Operating system	Windows 11
CPU	Intel(R) Core (TM) i9-11900K @ 3.50 GHz
GPU	NVIDIA GeForce RTX 3080Ti
Programming language	Python 3.7.10
Development framework	TensorFlow 2.15.0
Development tool	PyCharm 2024.3

4.3. Experimental Results and Analysis on the CWRU Dataset

4.3.1. Ablation Experiments

To evaluate the contribution of each key module to fault diagnostic performance, five derivative models were constructed by sequentially removing or replacing core components while keeping the overall network architecture and training strategy unchanged.

M1: The multi-scale parallel convolution structure MMCNN in the proposed model is replaced by single-scale convolution kernels and channels; M2: the ECA mechanism embedded in each parallel convolution branch of MSFCDA is removed; M3: the BiLSTM module in the model is replaced with a unidirectional LSTM; M4: the class-conditional feature distribution alignment strategy of the proposed model is replaced with DANN; M5: the weight of adversarial loss is fixed, without introducing a dynamic factor to adjust the adversarial intensity. Under these ablation settings, the diagnostic accuracies of all models on different transfer tasks are reported in Table 5 and Figure 8.

Table 5. Comparison of fault diagnosis accuracy of each ablation model and the proposed model under different transfer tasks.

Transfer Task	M1/%	M2/%	M3/%	M4/%	M5/%	MSFCDA/%
T1	91.82 ± 0.54	93.74 ± 0.25	96.21 ± 0.17	97.35 ± 0.30	98.66 ± 0.74	99.85 ± 0.09
T2	91.25 ± 0.92	94.11 ± 0.43	94.87 ± 0.20	96.82 ± 0.23	98.35 ± 0.41	99.60 ± 0.17
T3	90.96 ± 1.84	93.42 ± 0.63	94.36 ± 0.43	97.21 ± 0.24	96.82 ± 1.14	100.00 ± 0.00
T4	92.17 ± 0.35	94.36 ± 1.22	95.72 ± 1.11	98.63 ± 1.45	97.74 ± 0.39	100.00 ± 0.00
T5	91.48 ± 0.63	92.88 ± 0.58	96.54 ± 0.59	95.72 ± 1.75	98.05 ± 0.77	99.52 ± 0.42
T6	91.06 ± 0.29	93.51 ± 0.40	95.34 ± 0.58	97.11 ± 0.59	97.48 ± 1.26	99.71 ± 0.12
T7	92.35 ± 0.85	94.43 ± 0.27	95.63 ± 0.59	96.88 ± 0.79	97.86 ± 0.18	99.48 ± 0.17
T8	89.74 ± 1.36	91.17 ± 0.61	94.14 ± 0.46	97.54 ± 0.37	98.24 ± 1.26	100.00 ± 0.00
T9	91.22 ± 1.03	93.69 ± 0.33	96.36 ± 0.28	96.61 ± 0.42	97.58 ± 0.70	100.00 ± 0.00
T10	92.64 ± 0.21	92.78 ± 0.16	94.71 ± 0.34	95.76 ± 1.60	95.22 ± 0.37	99.33 ± 0.25
T11	91.40 ± 0.56	94.28 ± 0.68	95.48 ± 0.51	97.47 ± 0.26	98.14 ± 0.22	99.68 ± 0.16
T12	92.55 ± 0.48	93.62 ± 0.44	95.53 ± 0.25	97.25 ± 0.42	97.67 ± 1.47	99.57 ± 0.21
Accuracy	91.55 ± 0.76	93.50 ± 0.50	95.41 ± 0.46	97.03 ± 0.70	97.65 ± 0.74	99.73 ± 0.13

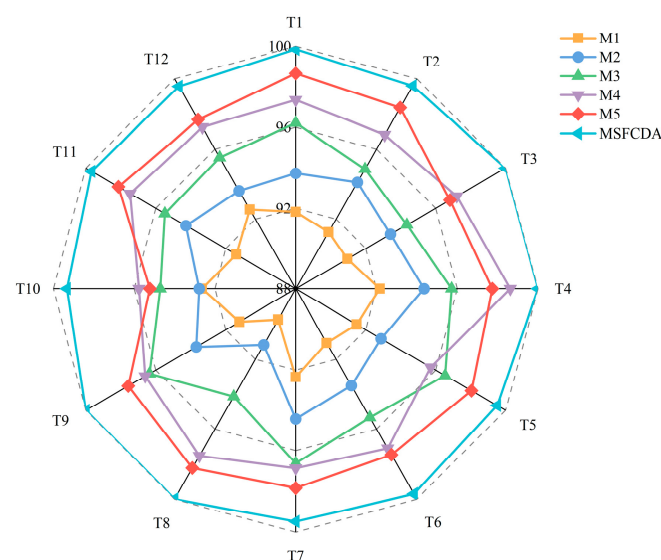


Figure 8. Radar chart of diagnostic accuracy of each model under different transfer tasks.

M1 exhibits the lowest average accuracy, indicating that single-scale features are insufficient to simultaneously characterize local impulsive information and global variation trends in fault signals, whereas multi-scale parallel convolution can extract richer features at different receptive fields. Compared with M3, M4, and M5, M2 exhibits a decrease in

diagnostic accuracy of approximately 1.9–4.2%. This indicates that removing the ECA mechanism prevents the model from distinguishing critical features from irrelevant information, resulting in degraded discriminative performance due to redundant features. M3 indicates that BiLSTM integrates temporal information from both historical and future contexts, producing more complete dynamic features and improving cross-condition identification capability. M4 also exhibits unsatisfactory diagnostic performance. Global feature distribution alignment is insufficient to fully resolve class-conditional distribution discrepancies, whereas class-conditional constraints enable more refined feature alignment and effectively reduce the influence of cross-domain distribution shifts. Compared with the other ablation models, MSFCDA achieves an average accuracy of 99.73%. This result indicates that the dynamic adversarial factor adaptively adjusts the adversarial intensity during training, progressively achieving more stable and refined cross-domain alignment. These results confirm that the key modules work synergistically and contribute substantially to diagnostic accuracy and generalization capability.

4.3.2. Comparative Experiments

To further evaluate the effectiveness and superiority of the proposed method for transfer fault diagnosis tasks, MSFCDA is compared with five representative methods: DCNN, DANN [22], MCD [23], JAN [24], and DCORAL [25]. The classification accuracy of different methods under various transfer tasks is presented in Table 6 and Figure 9.

Table 6. Comparison of accuracy of various methods under different transfer tasks.

Transfer Task	DCNN/%	DANN/%	MCD/%	JAN/%	DCORAL/%	MSFCDA/%
T1	82.14 ± 0.68	95.42 ± 0.43	96.34 ± 0.17	96.94 ± 0.30	96.40 ± 1.39	99.85 ± 0.09
T2	79.86 ± 1.04	95.94 ± 0.25	92.69 ± 1.11	98.11 ± 0.23	95.46 ± 0.74	99.60 ± 0.17
T3	75.42 ± 0.07	96.81 ± 0.63	96.64 ± 0.59	95.22 ± 0.24	92.67 ± 0.41	100.00 ± 0.00
T4	83.57 ± 0.59	96.19 ± 1.23	93.25 ± 0.46	97.40 ± 0.59	93.07 ± 0.16	100.00 ± 0.00
T5	80.87 ± 0.48	95.33 ± 0.53	94.47 ± 0.28	96.53 ± 0.42	94.86 ± 0.86	99.52 ± 0.42
T6	77.25 ± 0.85	94.62 ± 0.58	93.54 ± 0.34	96.31 ± 1.45	96.52 ± 0.51	99.71 ± 0.12
T7	81.96 ± 0.38	95.13 ± 0.68	91.92 ± 0.51	97.64 ± 0.37	95.38 ± 0.39	99.48 ± 0.17
T8	82.31 ± 0.49	94.79 ± 0.68	95.20 ± 0.58	95.87 ± 0.59	91.27 ± 0.64	100.00 ± 0.00
T9	78.62 ± 1.26	95.88 ± 0.40	96.33 ± 0.37	97.46 ± 1.00	96.73 ± 0.12	100.00 ± 0.00
T10	74.83 ± 0.45	94.45 ± 0.43	91.48 ± 0.41	95.78 ± 1.30	93.62 ± 1.47	99.33 ± 0.25
T11	76.15 ± 0.21	94.47 ± 0.61	92.86 ± 0.75	97.26 ± 0.49	96.19 ± 0.43	99.68 ± 0.16
T12	79.47 ± 0.55	95.20 ± 0.27	95.72 ± 1.48	95.83 ± 0.73	94.08 ± 1.02	99.57 ± 0.21
Accuracy	79.37 ± 0.59	95.35 ± 0.56	94.20 ± 0.59	96.70 ± 0.64	94.69 ± 0.68	99.73 ± 0.13

DCNN achieves an average accuracy of only 79.37%, this indicates that when the training and testing conditions exhibit distribution discrepancies, source-domain supervision alone is insufficient to learn transferable feature representations. By introducing a domain classifier and a gradient reversal mechanism, DANN enables the feature extraction network to learn domain-invariant representations and increases the average accuracy to 95.35%. Although DCORAL and MCD can alleviate domain shifts to some extent, DCORAL lacks direct constraints on class-discriminative boundaries, whereas MCD relies strongly on the quality of feature representations, and its alignment effectiveness remains limited. JAN jointly aligns feature distributions and classification-output information, substantially improving overall performance. However, its average accuracy remains approximately 3.03% lower than that of the proposed method, indicating that joint distribution matching alone is insufficient to maintain a stable advantage across all transfer tasks. In contrast, MSFCDA achieves consistently high classification accuracy across all transfer tasks, demonstrating more stable and superior diagnostic performance.

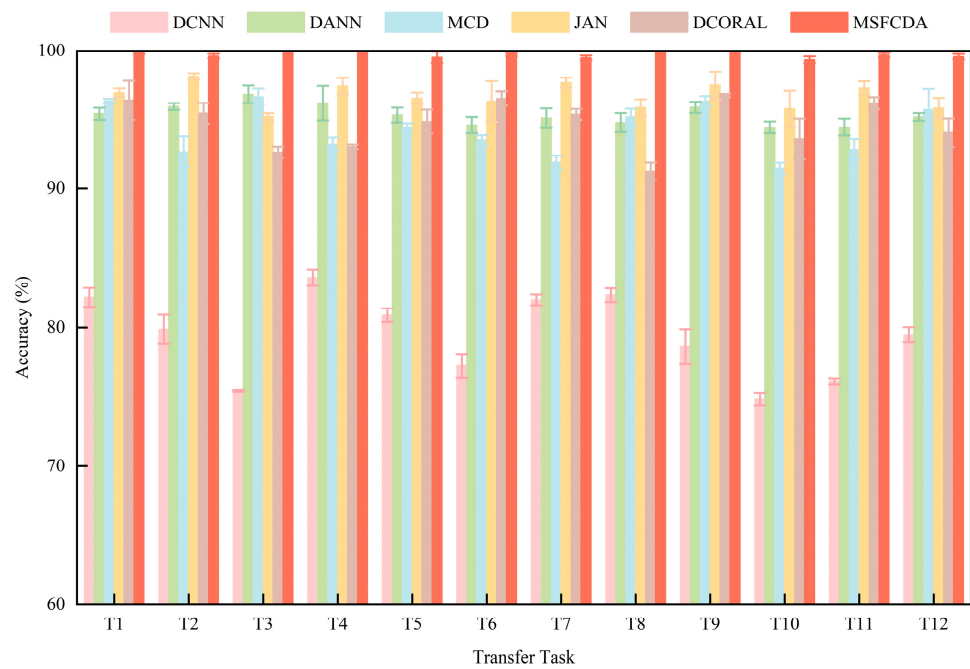


Figure 9. Diagnostic accuracy of each method under different transfer tasks.

To intuitively analyze misclassification among different fault categories, confusion matrices are used to examine the classification results for transfer task T5. The horizontal axis represents the predicted labels, and the vertical axis represents the true labels. Off-diagonal values represent the probability of misclassification between different categories, while darker colors indicate higher classification accuracy. The confusion-matrix results of each method for transfer task T5 are illustrated in Figure 10.

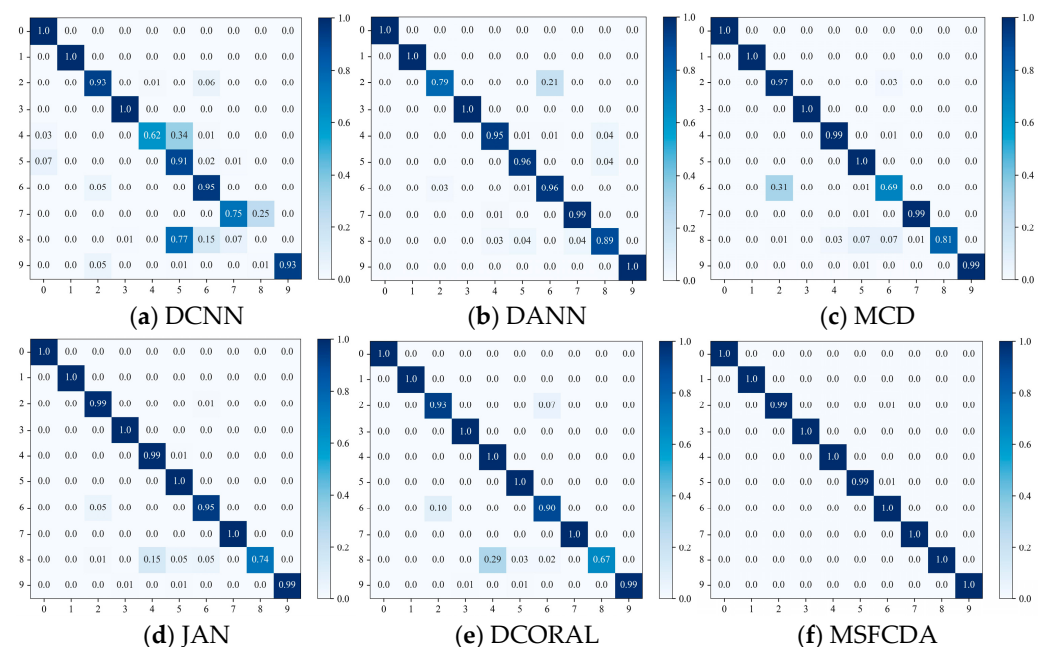


Figure 10. Confusion matrix result chart of the comparative experiments.

In Figure 10a, only classes 0, 1, and 3 are accurately identified, while the remaining fault types exhibit substantial confusion, particularly between inner-race and ball faults. Figure 10b shows a lower misclassification rate, indicating that the inter-domain distribution discrepancy is partially alleviated. Figure 10c,e demonstrate good classification

performance for classes 0–5, although substantial distribution shifts remain for classes 6 and 8. Figure 10d exhibits a relatively concentrated diagonal with only a small number of misclassified samples. In contrast, Figure 10f shows the best classification performance, with very few misclassifications, confirming that the proposed method can establish clearer class boundaries and effectively suppress fault-feature overlap.

To visualize the alignment effects of different domain-adaptation methods in the feature space, t-SNE is employed to reduce the dimensionality of the high-dimensional features extracted by the models. By preserving local structural characteristics, t-SNE maps high-dimensional features into a low-dimensional space, providing an intuitive view of the distribution of different classes and the degree of alignment between source- and target-domain features. The feature distributions obtained by the different comparison methods under a representative transfer task are shown in Figure 11.

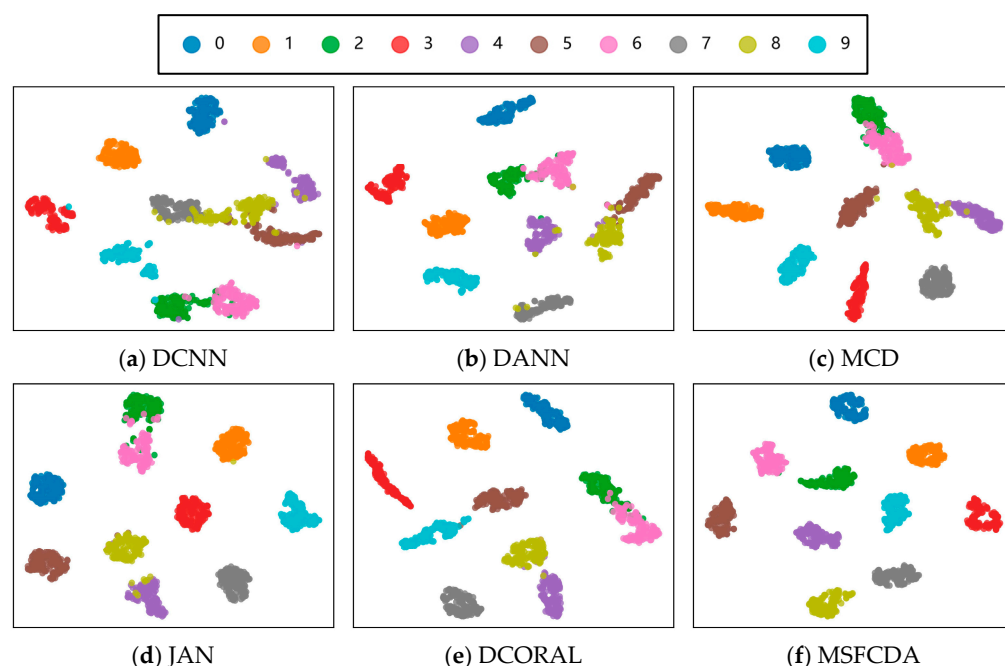


Figure 11. t-SNE visualization results of the comparative experiments.

The visualization results show that in Figure 11a, most samples fail to form clear cluster structures, and class distributions are highly mixed. Figure 11b,c,e, the DANN, MCD, and DCORAL models begin to form local clusters for different classes. Although domain shifts are alleviated to some extent, the clustering of class boundaries remains inadequate. Figure 11d achieves relatively accurate feature classification, but overlap remains among some samples. In contrast, Figure 11f shows ten highly compact and well-separated clusters, demonstrating that the proposed method maintains stable class-discriminative capability while performing cross-domain alignment and substantially improves the separability of fault types under varying operating conditions.

4.4. Experimental Results and Analysis on the PU Dataset

Comparative experimental results on the PU dataset are presented in Table 7 and Figure 12. The proposed model achieves the strongest overall performance among the 12 transfer tasks, with an average accuracy of 98.58%. DCNN performs poorly on most tasks, particularly T8, T10, and T11, where its accuracies are only 67.67%, 70.41%, and 69.85%, respectively. This indicates that without a domain-alignment mechanism, the model has difficulty eliminating distribution shifts among different operating conditions. The average accuracies of DANN and DCORAL increase to 91.76% and 93.19%, respectively,

demonstrating that domain-adversarial learning and statistical distribution alignment can improve transfer performance to some extent. However, pronounced fluctuations remain on tasks such as T10 and T11, indicating limited adaptability to complex class-conditional distribution discrepancies. MCD and JAN perform relatively well overall, achieving average accuracies of 94.40% and 96.56%, respectively. JAN maintains high accuracy on multiple tasks, suggesting that joint distribution alignment has certain advantages. Nevertheless, both methods exhibit varying degrees of performance gaps on T1, T4, T8, and T10 and fail to consistently achieve high diagnostic accuracy. In contrast, MSFCDA performs particularly well on T4 and T10, achieving accuracies of 99.60% and 99.08%, respectively. Overall, the proposed model achieves higher diagnostic accuracy and stronger training stability across different transfer tasks, indicating that it can more effectively learn discriminative cross-domain features and substantially improve fault diagnosis under varying operating conditions through more refined distribution alignment.

Table 7. Comparison of accuracy of various methods under different transfer tasks.

Transfer Task	DCNN/%	DANN/%	MCD/%	JAN/%	DCORAL/%	MSFCDA/%
T1	77.15 ± 0.56	93.65 ± 0.87	96.47 ± 0.28	97.88 ± 0.42	93.62 ± 1.15	99.27 ± 0.21
T2	71.87 ± 1.14	92.34 ± 0.45	94.62 ± 0.63	97.11 ± 0.31	92.71 ± 0.76	98.65 ± 0.58
T3	73.94 ± 0.32	87.68 ± 1.23	95.36 ± 0.47	96.71 ± 0.19	95.22 ± 0.92	98.87 ± 0.14
T4	74.21 ± 0.91	91.33 ± 0.69	93.85 ± 0.82	96.25 ± 0.54	93.73 ± 0.38	99.60 ± 0.08
T5	70.53 ± 0.68	94.72 ± 0.51	95.92 ± 0.24	94.84 ± 0.96	90.84 ± 1.34	98.54 ± 0.44
T6	72.44 ± 0.47	91.47 ± 0.79	89.66 ± 1.06	97.32 ± 0.37	93.48 ± 0.65	97.93 ± 0.73
T7	70.82 ± 0.83	93.26 ± 0.34	94.37 ± 0.58	96.54 ± 1.26	92.93 ± 0.49	98.45 ± 0.62
T8	67.67 ± 1.48	94.81 ± 0.62	92.22 ± 0.39	95.77 ± 0.74	94.35 ± 0.83	99.14 ± 0.17
T9	73.59 ± 0.24	95.58 ± 0.73	97.19 ± 0.55	94.39 ± 1.08	93.96 ± 0.41	97.42 ± 0.96
T10	70.41 ± 0.76	84.96 ± 1.37	95.56 ± 0.34	97.62 ± 0.61	89.57 ± 1.19	99.08 ± 0.27
T11	69.85 ± 1.43	89.73 ± 0.98	91.85 ± 0.71	96.84 ± 0.48	95.26 ± 0.56	96.79 ± 0.85
T12	75.32 ± 0.59	91.62 ± 0.82	95.72 ± 0.45	97.43 ± 0.68	92.64 ± 0.72	99.32 ± 0.13
Accuracy	72.32 ± 0.78	91.76 ± 0.78	94.40 ± 0.54	96.56 ± 0.65	93.19 ± 0.78	98.58 ± 0.43

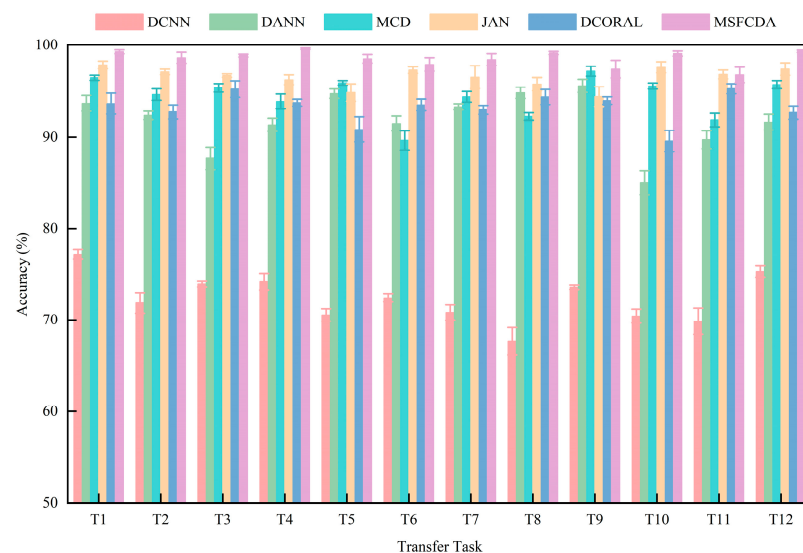


Figure 12. Diagnostic accuracy of each method under different transfer tasks.

To more intuitively demonstrate the discriminative ability and misclassification status of each method for different fault categories, transfer task T7 was selected. Confusion matrices corresponding to each model were plotted according to the fault-classification accuracy results, as shown in Figure 13. Figure 13a exhibits severe cross-misclassification among multiple fault classes, particularly for classes 2, 5, and 6, whose feature distributions are dispersed. The discriminative capability in Figure 13b is substantially improved,

although a considerable proportion of class 6 samples are still misclassified as class 4. The misclassification rates in Figure 13c,e are markedly reduced, indicating that global distribution alignment can partially reduce inter-domain discrepancies. Figure 13d shows relatively concentrated diagonal elements, though a slight shift still exists. The proposed method produces the closest approximation to the ideal classification pattern and better preserves discriminative boundaries among fault classes.

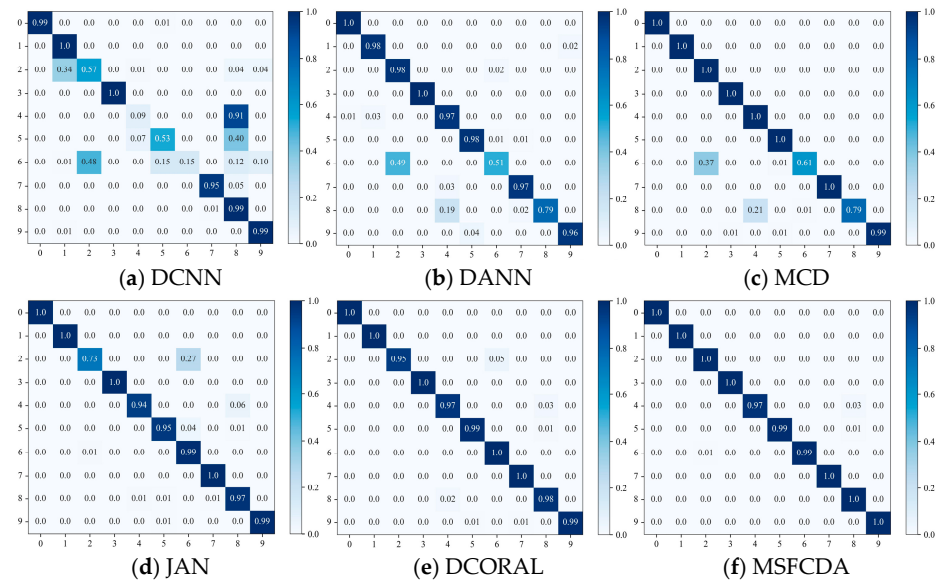


Figure 13. Confusion matrix result chart of the comparative experiments.

To further investigate the distribution structures learned by the different methods in the feature space, t-SNE is again employed to reduce the dimensionality of the extracted high-dimensional features, as illustrated in Figure 14. Comparing the clustering patterns produced by different methods provides an intuitive means of examining inter-class separability and intra-class compactness.

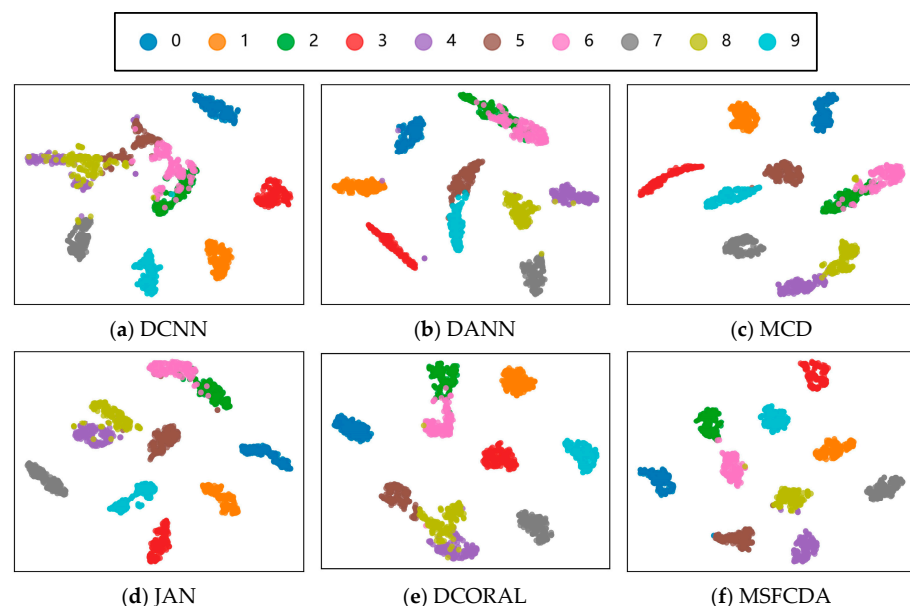


Figure 14. t-SNE visualization results of the comparative experiments.

The feature distribution in Figure 14a is relatively disordered, with blurred class boundaries and large inter-class dispersion. Figure 14b partially alleviates class overlap,

but substantial intersections remain among some classes. In Figure 14c,e, different classes gradually form relatively independent clusters, with more compact intra-class samples. Figure 14d further improves the feature structure and achieves relatively high accuracy. In contrast, MSFCDA exhibits the best feature visualization: the ten fault classes form clearly separated clusters with highly compact intra-class samples and substantially increased inter-class dispersion, without obvious class overlap.

4.5. Hyperparameter Sensitivity Analysis on the CWRU and PU Datasets

To investigate the influence of the optimal combination of the loss weight α and the learning-rate decay parameter β on model performance and determine a reasonable parameter range, a two-dimensional grid search is conducted to analyze the sensitivity of α and β . All hyperparameters were selected solely on the source-domain validation set, and the target-domain test set was used only once for the final evaluation after α and β had been fixed. The distributions of diagnostic accuracy under different parameter combinations on the transfer tasks of the CWRU and PU datasets are shown in Figure 15.

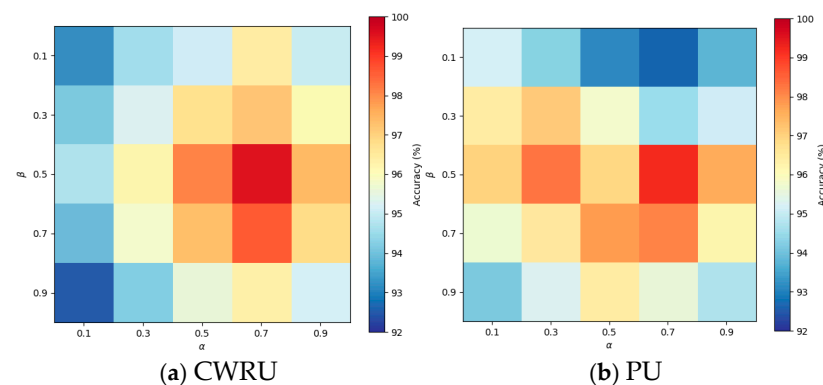


Figure 15. Heatmap of the 2D grid search over the loss weight α and the learning-rate decay parameter β on two datasets.

The loss weight α regulates the contribution of the domain-adversarial loss to the total loss, while the learning-rate decay parameter β controls the convergence behavior during training to balance multiple optimization objectives. Although the high-accuracy regions differ between the two heat maps. In Figure 15a, the high-accuracy region is relatively concentrated, and diagnostic accuracy decreases smoothly as the parameters deviate from the optimum, indicating that the model is relatively insensitive to small parameter perturbations and exhibits stable overall performance on this dataset. In contrast, Figure 15b shows a more pronounced performance gradient, with a larger decrease in accuracy when the parameter combination moves away from the optimal range, indicating that the PU dataset is more sensitive to changes in the loss weight and the learning-rate decay parameter. Overall, the optimal parameter combinations for both datasets are concentrated around $\alpha = 0.7$ and $\beta = 0.5$, achieving the best diagnostic performance on both datasets and providing a reasonable reference for hyperparameter selection in engineering applications.

5. Conclusions

To address the practical challenges of insufficient multi-level feature extraction from bearing vibration signals by single deep-learning architectures, substantial feature-distribution discrepancies among different operating conditions, and unstable convergence during cross-domain adversarial training, this study proposes a multi-structure fusion conditional domain adaptation (MSFCDA) method for rolling-bearing fault diagnosis. The

model is developed from three aspects: feature-extraction construction, class-conditional domain alignment, and dynamic regulation of adversarial strength. Its performance is comprehensively evaluated through ablation experiments, comparative experiments, feature visualization, and hyperparameter sensitivity analysis.

(1) An MMCBE multi-structure fusion feature extractor was constructed. Multi-channel multi-scale parallel convolution captures local impulsive features and global variation trends in vibration signals using different receptive fields. An efficient channel attention (ECA) mechanism was embedded into each convolutional branch to adaptively enhance fault-sensitive channels and suppress redundant channel information. BiLSTM was employed to model bidirectional temporal dependencies in vibration signals. The coordinated operation of these modules produces deep fault features incorporating multi-scale, temporal, and channel-wise information, providing a discriminative feature basis for subsequent cross-domain alignment.

(2) A class-conditional feature distribution alignment strategy and a dynamic adversarial training mechanism were developed. The class posterior probabilities produced by the classifier are coupled with deep features through an outer product to construct a class-conditional joint representation, thereby introducing class-prior information into domain discrimination and overcoming the lack of class constraints in conventional global domain alignment that can lead to class overlap. Meanwhile, a cosine-annealing-based dynamic adversarial weighting factor was introduced to alleviate gradient oscillations and convergence instability associated with fixed-weight adversarial training, enabling dynamic balancing between fault classification and domain-invariant feature learning.

(3) Multiple unlabeled cross-condition transfer tasks were established using the CWRU and PU bearing datasets. The proposed method demonstrates clear advantages in both classification accuracy and generalization performance, effectively alleviating distribution discrepancies caused by changes in operating conditions and achieving effective alignment of source- and target-domain feature spaces. Hyperparameter analysis identifies a favorable combination of loss trade-off parameters, providing a reference for parameter selection in engineering applications. The proposed method therefore provides a feasible technical solution for rolling-bearing fault diagnosis when target-domain data are unlabeled under varying operating conditions.

Author Contributions: Conceptualization, C.L.; methodology, C.L.; software, C.L.; validation, C.L. and J.W.; investigation, J.W.; data curation, C.L.; writing—original draft preparation, C.L.; writing—review and editing, C.L.; visualization, C.L.; supervision, J.W.; project administration, J.W.; funding acquisition, J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the Major Science and Technology Project of Shanxi Province under Grant 202301020101001.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author(s).

Acknowledgments: The authors thank the editors and reviewers for their valuable comments and suggestions to improve the quality of this manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhao, Y.; Yang, X.; Huang, J.; Gao, J.; Cui, J. Improved weighted extreme learning machine with adaptive cost-sensitive strategy for imbalanced fault diagnosis of rotating machinery. *Mech. Syst. Signal Process.* **2024**, *217*, 111526. [[CrossRef](#)]
2. Xiao, Y.M.; Shao, H.D.; Han, S.Y.; Huo, Z.; Wan, J. Novel joint transfer network for unsupervised bearing fault diagnosis from simulation domain to experimental domain. *IEEE/ASME Trans. Mechatron.* **2022**, *27*, 5254–5263. [[CrossRef](#)]

3. Xu, Y.; Li, Z.; Wang, S.; Li, W.; Sarkodie-Gyan, T.; Feng, S. A hybrid deep-learning model for fault diagnosis of rolling bearings. *Measurement* **2021**, *169*, 108502. [[CrossRef](#)]
4. Chen, Y.; Zhang, D.; Zhu, K.; Yan, R. An adaptive activation transfer learning approach for fault diagnosis. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 2645–2656. [[CrossRef](#)]
5. Zhu, Z.; Lei, Y.; Qi, G.; Chai, Y.; Mazur, N.; An, Y.; Huang, X. A review of the application of deep learning in intelligent fault diagnosis of rotating machinery. *Measurement* **2023**, *206*, 112346. [[CrossRef](#)]
6. Chen, H.; Miao, F.; Chen, Y.; Xiong, Y.; Chen, T. A hyperspectral image classification method using multi feature vectors and optimized KELM. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2781–2795. [[CrossRef](#)]
7. Yu, Z.F.; Xiong, B.S.; Xiong, T.Y.; Ou, Q.F.; Li, X.M. Helicopter bearing fault diagnosis based on VMD-CWT and improved CNN. *J. Aerosp. Power* **2021**, *36*, 948–958.
8. Yan, H.; Qin, Y.; Xiang, S.; Wang, Y.; Chen, H. Long-term gear life prediction based on ordered neurons LSTM neural networks. *Measurement* **2020**, *165*, 108205. [[CrossRef](#)]
9. Guo, Y.; Mao, J.; Zhao, M. Rolling bearing fault diagnosis method based on attention CNN and BiLSTM network. *Neural Process. Lett.* **2023**, *55*, 3377–3410. [[CrossRef](#)]
10. Sui, J.X.; Su, Y.; Qi, J.L. Rolling bearing fault diagnosis method based on time-frequency image and improved convolutional neural network. *J. Xi'an Technol. Univ.* **2026**, *46*, 1–11.
11. Liu, X.L.; Xu, X.L. Fault diagnosis method based on feature pyramid convolutional recurrent neural network. *J. Shanghai Jiao Tong Univ.* **2022**, *56*, 182–190.
12. Li, R.; Li, S.; Xu, K.; Li, X.; Lu, J.; Zeng, M.; Li, M.; Du, J. Adversarial domain adaptation of asymmetric mapping with CORAL alignment for intelligent fault diagnosis. *Meas. Sci. Technol.* **2022**, *33*, 055101. [[CrossRef](#)]
13. Wan, L.; Li, Y.; Chen, K.; Gong, K.; Li, C. A novel deep convolution multi-adversarial domain adaptation model for rolling bearing fault diagnosis. *Measurement* **2022**, *191*, 110752. [[CrossRef](#)]
14. Wang, X.; Shen, C.; Xia, M.; Wang, D.; Zhu, J.; Zhu, Z. Multi-scale deep intra-class transfer learning for bearing fault diagnosis. *Reliab. Eng. Syst. Saf.* **2020**, *202*, 107050. [[CrossRef](#)]
15. Li, F.; Tang, T.; Tang, B.; He, Q. Deep convolution domain-adversarial transfer learning for fault diagnosis of rolling bearings. *Measurement* **2021**, *169*, 108339. [[CrossRef](#)]
16. Han, T.; Liu, C.; Yang, W.; Jiang, D. A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults. *Knowl.-Based Syst.* **2019**, *165*, 474–487. [[CrossRef](#)]
17. Jiao, J.; Zhao, M.; Lin, J.; Liang, K. Residual joint adaptation adversarial network for intelligent transfer fault diagnosis. *Mech. Syst. Signal Process.* **2020**, *145*, 106962. [[CrossRef](#)]
18. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 6999–7019. [[CrossRef](#)] [[PubMed](#)]
19. Fu, G.; Wei, Q.; Yang, Y. Bearing fault diagnosis with parallel CNN and LSTM. *Math. Biosci. Eng.* **2024**, *21*, 2385. [[CrossRef](#)] [[PubMed](#)]
20. Zhang, S.; Deng, A.D.; Xu, S.; Ding, X. Rolling bearing fault diagnosis based on DFT and ECA. *J. Vib. Meas. Diagn.* **2024**, *44*, 754–760+830.
21. Qian, Q.; Yu, Y.; Qin, Y.; Zhang, J.; Cheng, Y.; Chen, K. Asynchronous joint distribution alignment: A new domain confusion mechanism for fault transfer diagnosis. *IEEE/ASME Trans. Mechatron.* **2026**. [[CrossRef](#)]
22. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
23. Wu, Z.; Jiang, H.; Lu, T.; Zhao, K. A deep transfer maximum classifier discrepancy method for rolling bearing fault diagnosis under few labeled data. *Knowl.-Based Syst.* **2020**, *196*, 105814. [[CrossRef](#)]
24. Zhao, K.; Jiang, H.; Wang, K.; Pei, Z. Joint distribution adaptation network with adversarial learning for rolling bearing fault diagnosis. *Knowl.-Based Syst.* **2021**, *222*, 106974. [[CrossRef](#)]
25. Li, X.; Jiang, H.; Liu, S.; Zhang, J.; Xu, J. A unified framework incorporating predictive generative denoising autoencoder and deep coral network for rolling bearing fault diagnosis with unbalanced data. *Measurement* **2021**, *178*, 109345. [[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.