

Article citation info:

Li C, Li G., Xing P, Cui D., Zhang H, A fault diagnosis method for marine diesel engine system components based on adversarial neural network, *Eksploracja i Niezawodność – Maintenance and Reliability* 2026: 28(2) <http://10.17531/ein/215704>

A fault diagnosis method for marine diesel engine system components based on adversarial neural network

Indexed by:



Congyue Li^a, Guobin Li^{a,*}, Pengfei Xing^a, Dexin Cui^b, Hongpeng Zhang^a

^a Marine Engineering Colloge, Dalian Maritime University, China

^b Ningbo Ocean Shipping CO., LTD, China

Highlights

- A more accurate class-level matching strategy is developed.
- A shallow independent-deep shared strategy is designed.
- A progressive adversarial training diagnostic framework is proposed.

Abstract

Diesel engine is the main power source of ships, and its failure seriously affects safe navigation. Transfer learning is an effective method to solve cross-domain diagnosis of marine diesel engine. However, when facing large domain shifts, relying solely on cross-domain consistent semantic information cannot achieve high-precision diagnosis of the target task. A diagnostic method for diesel engine has been proposed. The shallow independent - deep sharing mechanism is first designed to fully consider the specificity and commonality of different domains. Then, the reliable alignment of different classes of source and target domains is achieved based on domain alignment theory and pseudo-label strategy. The performance of the framework is rigorously evaluated by two real-world cases and its applicability across subsystems is verified. The results demonstrate that the average accuracy of the framework is over 93%, outperforming other methods by more than 1.55%. Even in tasks with large domain distribution discrepancies, the framework also enables robust feature transfer.

Keywords

adversarial training, class-level matching, fault diagnosis, marine diesel engine, pseudo-label

This is an open access article under the CC BY license (<https://creativecommons.org/licenses/by/4.0/>)

1. Introduction

Maritime industry is vital to the development of the global economy and accounts for about 90% of global trade [1], [2]. Meanwhile, vessels safety risks are also of great concern. Based on the Lloyd's List Intelligence accident database [3], Fig. 1 shows the main accident types for vessels of 3000 gross tonnage and above in 2023. More than 50% of the ship accidents were caused by mechanical (e.g., main diesel engine and auxiliary machinery) failures. The report [4] stated that effective condition monitoring and fault diagnostic techniques can

prevent worsening machinery failures and reduce ship accidents. Therefore, intelligent fault diagnosis technology plays a pivotal role in enhancing the level of intelligent operation and maintenance for vessels, as well as ensuring navigational safety. Lloyd's Register [5] reports that the shipping industry is undergoing major changes driven by artificial intelligence technologies. With the improvement of ship digitalization and automation, intelligent classification and fault-tolerant control based on deep learning have been widely used in the field of

(*) Corresponding author.

E-mail addresses:

C. Li (ORCID: 0000-0002-9290-5119) licongyue0126@163.com, G. Li (ORCID: 0000-0003-1307-9507) guobinli88@163.com, P. Xing (ORCID: 0000-0002-5911-1544) wlyxpf@dlnu.edu.cn, D. Cui (ORCID: 0009-0008-1430-0385) cuidx@nbosco.com, H. Zhang (ORCID: 0000-0003-3172-600X) zhppeter@dlnu.edu.cn

marine machinery [6-7]. Dong et al. [8] achieved health condition assessment of diesel engine subsystems through

a particle swarm optimization-based BP neural network.

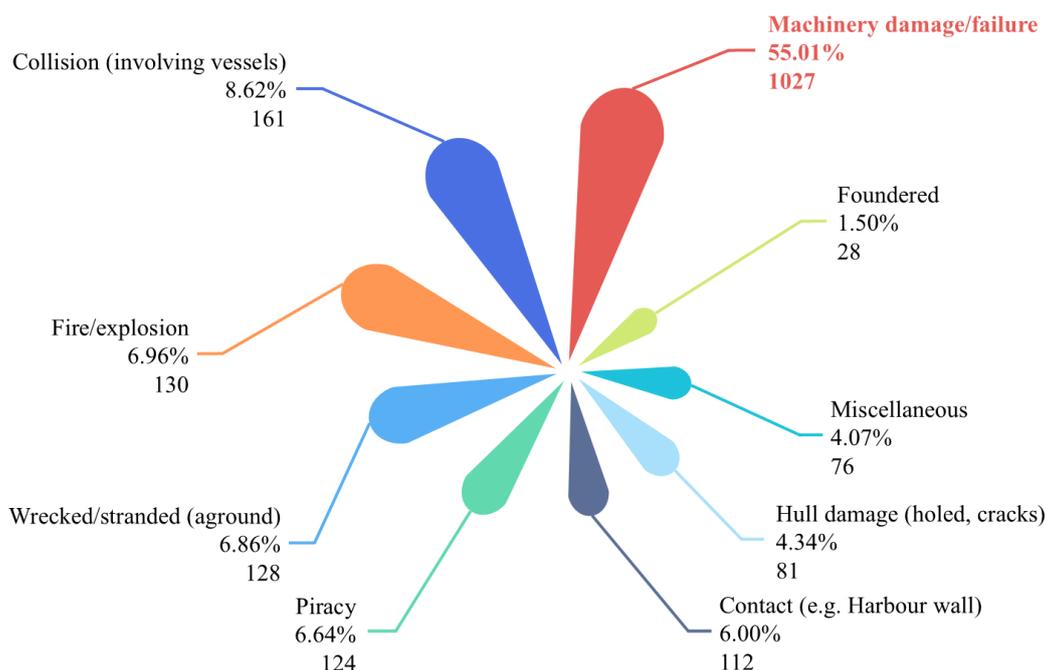


Fig. 1. Main accident types for ships of 3000 gross tonnage and above.

The model had the generalization ability while achieving accurate prediction results. Liu et al. [9] built a multi-scale convolutional neural network (CNN) model and successfully applied it to marine pipeline valve leakage monitoring. Shahid et al. [10] achieved real-time monitoring and classification of multiple abnormal states of diesel engine by using CNN. Shi et al. [11] used entropy to weight the dataset and combined it with a dual-channel CNN to achieve hydraulic directional valve fault diagnosis. Coraddu et al. [12] proposed a fault diagnosis framework integrating digital twins with weakly supervised learning, effectively reducing reliance on large amounts of labeled data. This provides a more feasible solution for condition monitoring of marine dual-fuel engines. Qin et al. [13] optimized the parameters of a back propagation neural network to improve the diagnostic accuracy of an electric rudder system. Weak faults are difficult to detect when two different degrees of faults occur simultaneously in rotating machinery. To solve this problem, Dibaj et al. [14] combined variational modal decomposition and CNN to achieve the classification of compound faults. Currently, popular deep learning methods have their own characteristics. Xu et al. [15] designed the CNN containing multi-attention mechanism and residual structure to achieve diesel engine fault diagnosis. For different diagnostic

objects, various methods have been applied to different degrees. However, deep learning methods still have some shortcomings. (1) When there are distributional differences between the training and test sets, the generalization ability of the method will be decreased [16]. (2) The effective implementation of the method relies on a large number of labeled datasets [17].

The development of transfer learning provides an effective solution to the problem of cross-domain intelligent diagnosis of marine machinery [18-19]. A “domain” is defined by the data distribution under a specific working condition (e.g., load). The “cross-domain” fault diagnosis refers to transferring knowledge from a source domain (labeled data under one condition) to a target domain (unlabeled data under a different condition) to achieve accurate diagnosis. Fault diagnosis methods based on parameter transfer were the first to be developed and utilized [20]. This method focuses on sharing common information between different domains at the model level. Jiang et al. [21] applied the pre-trained CNN model to a new diagnostic task and fine-tuned the parameters to achieve mine hoist main bearings fault diagnosis. Zhang and Yang [22] used a transfer strategy of pre-trained networks to extract diagnostic knowledge from the source domain for transformer fault diagnosis. Cui et al. [23] utilized the Generative Adversarial Network (GAN) to perform

data augmentation on the raw data, which was then fed into a CNN for further training. The real fault data was input into a pretrained network, enabling fault diagnosis of the fuel injection system.

Recently, feature-based transfer methods are widely used in the field of fault diagnosis [24]. The goal of this approach is to find a suitable mapping function. The features of the different domains are mapped to a common feature space to obtain domain invariance of different domains. Tang et al. [25] proposed a fault diagnosis method combining matrix classifiers with maximum mean discrepancy (MMD) theory, enabling the diagnosis of rolling bearing faults. Zhang et al. [26] combined stacked autoencoder and MMD to achieve fault early warning of centrifugal blower. Li et al. [27] proposed a pseudo label-guided class-level matching method based on the MMD approach, enabling cross-domain diagnosis of ship propulsion shaft systems. Huang et al. [28] constructed a deep transfer learning model and aligned conditional probability distributions using local MMD (LMMD). Hu et al. [29] adopted margin disparity discrepancy (MDD) to achieve variable condition fault diagnosis of marine machinery. Hu et al. [30] constructed a deep transfer learning network model, utilizing multi-kernel MMD (MK-MMD) to reduce distribution differences between domains, thereby achieving cross-domain diagnostics for turbocharging systems. Pei et al. [31] constructed an encoding network using raw signals and employed a turn-based adversarial method to enhance the network's generalization capability, successfully achieving cross-domain diagnosis of diesel engine. Li et al. [32] proposed a transfer learning method based on maximum classifier discrepancy (MCD) and verified the generalizability of the method through rolling bearing fault experiments. The adversarial training approach fully takes into account the impact of classification boundary differences due to task-specific transfer, effectively reducing the risk of generating uncertain features.

The research mentioned has promoted the application and development of deep-learning diagnostic methods in the field of marine machinery. However, most existing cross-domain approaches still have notable limitations. Firstly, the majority of methods rely solely on aligning feature distributions through identical transformation mechanisms across domains. While this helps in learning domain-invariant features, it overlooks the

transferability disparity of semantic information. Specifically, certain semantic features are domain-specific and should not be forcibly aligned, as doing so may result in the loss of critical diagnostic information or lead to negative transfer. This issue becomes particularly pronounced in complex marine engineering scenarios with significant domain shifts, such as diagnosing the same fault across highly divergent operating conditions (e.g., from low load to high load), where inflexible feature alignment can actually hinder diagnostic performance. Secondly, the effective implementation of class-level matching-essential for fine-grained domain alignment-heavily depends on the availability of reliable pseudo-labels. Under large domain distribution discrepancies, the feature distributions of target-domain samples often become complex and heterogeneous. This increases the uncertainty of pseudo-labels generated by pre-trained source models. Relying solely on these error-prone pseudo-labels for subclass alignment (e.g., in LMMD or conditional distribution alignment) can introduce noise into the training process, resulting in error propagation and degraded model performance. This is a critical and widely observed limitation in current adversarial and discrepancy-based domain adaptation methods.

To solve the aforementioned problems, a two-stage adversarial network-based diagnostic framework is proposed. This diagnostic framework not only effectively aligns transferable features across domains, but also preserves domain-specific semantic attributes, while simultaneously enhancing the reliability of pseudo-labels for robust subclass alignment under large domain shifts. The main contributions and innovations of this study is as follows:

- (1) To resolve the negative transfer issue caused by overlooking domain-specific features under large distribution shifts, a shallow independent - deep shared mechanism is designed. The proposed network employs dual-path parallel processing in shallow layers to explicitly capture domain-specific features, while sharing deep layers to learn domain-invariant representations. This hierarchical design significantly enhances the network's adaptability and robustness when confronting substantial domain differences.

- (2) To mitigate the critical challenge of unreliable pseudo-labels, a novel dual pseudo-label guided class-level matching strategy is proposed. This strategy innovatively fuses the

insights from both the classifier's predictions (statistical perspective) and the cosine similarity between target features and source class centroids (geometric perspective). This synergy generates more accurate and robust pseudo-labels, thereby enabling finely-grained subdomain alignment and effectively minimizing intra-class variance across domains.

(3) The adaptability of the proposed framework in multi-subsystem diagnostic scenarios is verified. The framework shows excellent diagnostic performance in both auxiliary blower and fuel injector cross-domain fault diagnosis tasks, demonstrating its robustness and applicability in real-world scenarios.

2. Proposed transfer learning method

2.1. Diagnosis framework

The developed framework consists of three main parts: data preprocessing, network training, and network testing, as shown in Fig. 2. The workflow is described below:

- (1) Data from the blower and injector under various operational conditions (including different loads and rotational speeds) were collected for different health states. Specifically, the data refers to the acceleration signals recorded during the blower experiments, as well as the acceleration and pressure signals obtained during the injector experiments.
- (2) Construct the enhanced transfer CNN and initialize the network model parameters. Batch size is set to 64. The initial learning rate is 0.001. The maxepoch is 1000. The Adam optimizer is selected.
- (3) Input training data to minimize the source domain loss and achieve global domain matching between the source and target domains. The loss can be expressed as $L_1 = L_{cls}(X_s, Y_s) + \lambda L_{MMD}$. Here, L_1 refers to the total training loss for this stage. $L_{cls}(X_s, Y_s)$ denotes the source domain cross-entropy loss, while L_{MMD} represents the Maximum Mean Discrepancy (MMD) loss. λ represents the trade-off factor. (X_s, Y_s) represents the source domain dataset and its corresponding labels.
- (4) The feature extractor parameters are fixed. Maximize the output discrepancy between the two

classifiers. The objective loss can be expressed as

$$\min_{c_1, c_2} L_{cls}(X_s, Y_s) - L_{dis}(X_t). L_{dis}(X_t)$$

represents the discrepancy loss. X_t is the target domain dataset.

- (5) Fixed classifiers parameters. The loss $\min_F L_{dis}(X_t)$ is calculated to minimize output discrepancy.
- (6) Repeat steps 3~5 to make the different domains match as much as possible. When the first set number of iterations (260) is reached, the model will enter the next stage of training. This value is determined based on preliminary experiments and ensures that the model has fully converged on global features before entering the more refined class alignment stage.
- (7) The feature extractor is reconfigured into the shallow independent-deep shared architecture, as shown in Fig. 2 (the stage 2 step 1). The two light blue structures represent two independent branches, one processing source domain data and the other processing target domain data. Each branch contains its own learnable parameters for capturing domain-specific features in shallow layers. The outputs from these two branches are then fed into deeper layers (dark blue). Parameters in the deeper layers remain shared across domains to extract domain-invariant representations. The loss function $L_1 = L_{cls}(X_s, Y_s) + \lambda L_{MMD}$ is replaced with $L_2 = L_{cls}(X_s, Y_s) + \lambda L_{CL-MMD} + L_t$, aiming to minimize the source domain loss while achieving class-level matching. L_2 represents the total loss for this stage. L_{CL-MMD} is the class-level matching loss. L_t denotes the cross-entropy loss of the target domain.
- (8) Repeat steps 3~5 to achieve class-level matching for different domains.
- (9) The target domain test set is input into the trained model. The average of the two classifiers' outputs serves as the final diagnostic accuracy.

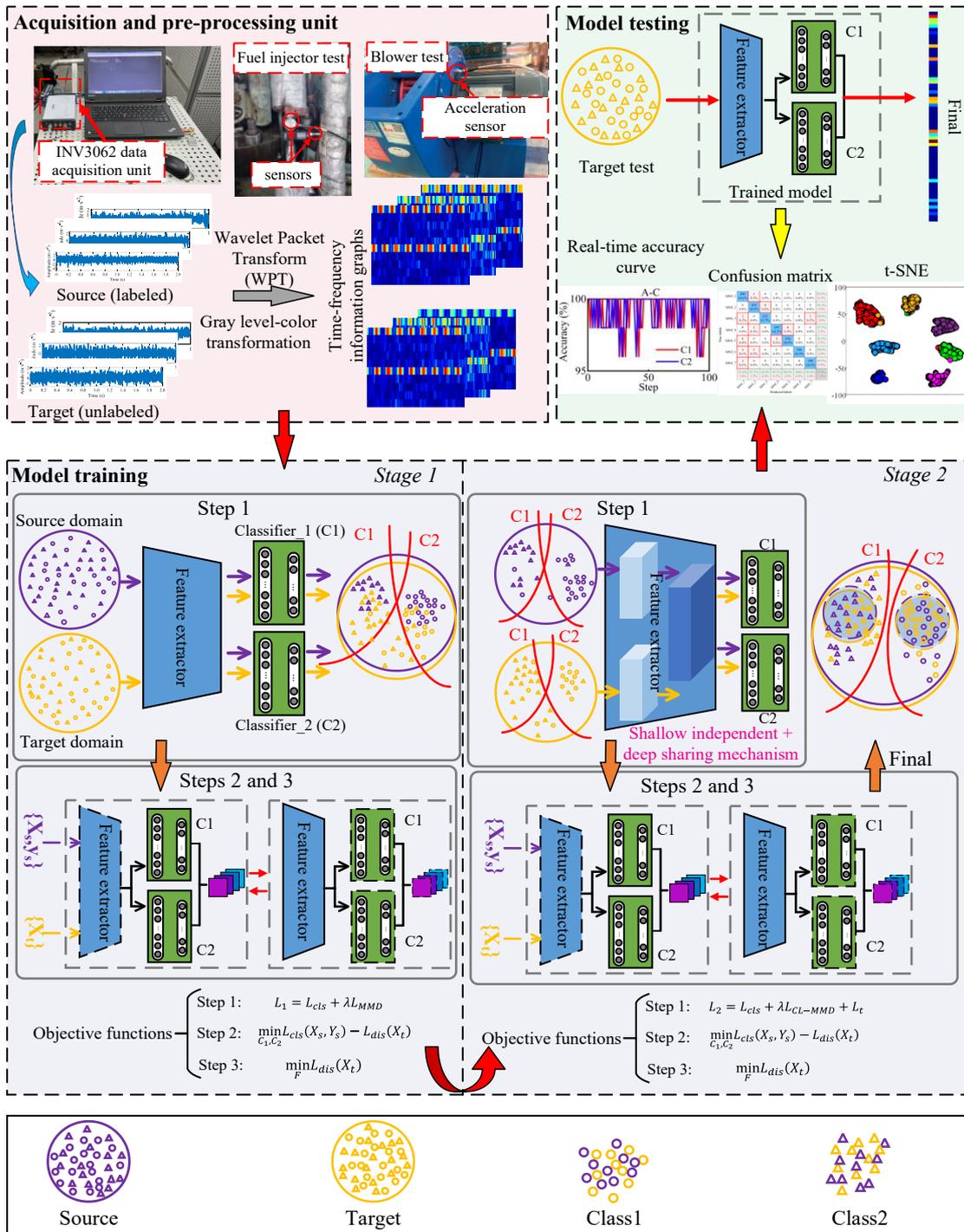


Fig. 2. The developed framework.

2.2. Maximum Classifier Discrepancy (MCD)

Existing intelligent diagnostic methods based on transfer learning usually perform feature alignment of the source and target domains to reduce distributional differences. However, these methods do not consider the effect of source domain category decision boundary on feature matching in the target domain. The MCD [33] approach overcomes the above problem. The training process of MCD is shown in Fig. 3. It should be

noted that Fig. 3 depicts the training phase (where source domain and target domain training data are input jointly). Under ideal conditions, the difference between the outputs of the two classifiers is zero. It introduces a feature extractor and two classifiers. The adversarial training technique is used to measure the features at the decision boundaries while reducing the distributional differences between different domains. The MCD learning process and the corresponding loss function are as

follows:

Step 1: The source domain dataset X_s is fed into the network for training. The source domain cross-entropy loss is expressed as follows:

$$L_{cls}(X_s, Y_s) = -E_{(x_i^s, y_i^s) \sim (X_s, Y_s)} \sum_{c=1}^C f[y_i^s = c] \log p(y_i^s | x_i^s) \quad (1)$$

where Y_s denotes the source domain label. E is Expectation. (x_i^s, y_i^s) is a specific sample-label pair. The joint distribution of the entire source domain dataset and its corresponding labels is represented as (X_s, Y_s) . \sim indicates sampling from the source domain dataset. f represents the indicator function. $p(y_i^s | x_i^s)$ represents the classifier's probability output for source domain samples. x_i^s represents the source domain sample, and y_i^s represents the output label. i denotes the sample number, s represents the source domain. C is the total number of categories. c represents the category index currently being processed. Train the whole network using labeled source domain data to minimize the source domain cross-entropy loss. The objective is as follows:

$$\min_{F, C1, C2} L_{cls}(X_s, Y_s) \quad (2)$$

Where F denotes the feature extractor. $C1$ and $C2$ represent two

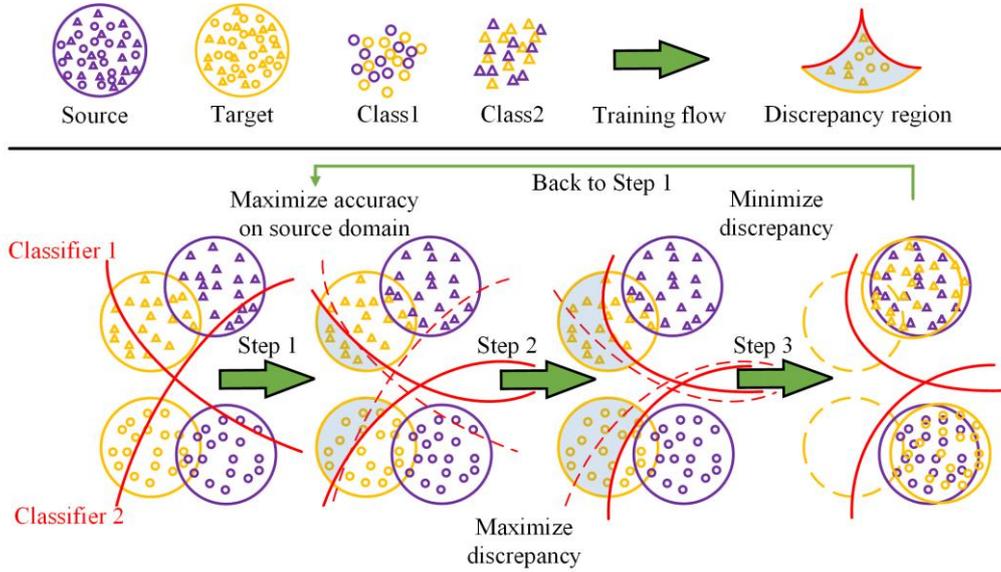


Fig. 3. MCD training process.

2.3. Pseudo-label guided class-level matching strategy

A two-stage progressive domain alignment strategy is developed to give the network more freedom in class-level matching. The global domain alignment is achieved using the MMD [25] method. The MMD equation is as follows:

$$L_{MMD} = \|E_P(\phi(x^s)) - E_Q(\phi(x^t))\|_{H_k}^2 \quad (6)$$

classifiers.

Step 2: The F is fixed. Maximizing the classification discrepancies of the classifiers. The classification discrepancy loss is shown in Eq. (3). Update the classifiers parameters according to Eq. (4).

$$\begin{cases} d(p_1, p_2) = \frac{1}{C} \sum_{c=1}^C |p_{1c} - p_{2c}| \\ L_{dis} = E_{x_j^t \in X_t} [d(p_1(y|x_j^t), p_2(y|x_j^t))] \end{cases} \quad (3)$$

$$\min_{C1, C2} L_{cls}(X_s, Y_s) - L_{dis}(X_t) \quad (4)$$

Where $d(p_1, p_2)$ measures the difference between the outputs of two classifiers. x_j^t represents the j th sample in target domain. p_{1c} and p_{2c} are the probability outputs for class c from two classifiers. $p_1(y|x_j^t)$ and $p_2(y|x_j^t)$ represent the probability outputs of classifiers $C1$ and $C2$, respectively. X_t is the target domain dataset.

Step 3: The classifiers are fixed. Minimizing the output difference between two classifiers. The calculation formula is as follows:

$$\min_F L_{dis}(X_t) \quad (5)$$

where E denotes the mathematical expectation, H_k denotes the regenerated Hilbert space. x^s is an input sample from the source domain. $\phi(x^s)$ represents mapping the input sample x^s to a high-dimensional feature space. $E_P(\phi(x^s))$ represents the expectation of the feature map $\phi(x^s)$ under the probability distribution P (source domain). Similarly, $E_Q(\phi(x^t))$ can be derived.

Based on the class-level matching theory, LMMD [34] is calculated as follows:

$$L_{LMMD} = \frac{1}{m} \sum_m^M \left\| \sum_{x_i^s \in D_s} \omega_i^{sm} \phi(x_i^s) - \sum_{x_j^t \in D_t} \omega_j^{tm} \phi(x_j^t) \right\|_{H_k}^2 \quad (7)$$

Where m represents the number of fault types. x_i^s and x_j^t represent samples from the source domain and target domain, respectively. ω denotes the weight of the sample class. ω is calculated as follows:

$$\omega_i^m = \frac{y_{im}}{\sum_{x_i, y_i \in D} y_{im}} \quad (8)$$

Where y_{im} represents the probability that vector x_i belongs to class m .

When calculating the weights for each class of samples in the target domain, the pseudo label probability distribution is computed through the model. Eq. (7) can be rewritten as:

$$L_{LMMD} = \frac{1}{C} \sum_{c=1}^C \left[\sum_{i=1}^{n_s} \sum_{j=1}^{n_s} w_i^{sc} w_j^{sc} K(z_i^{sl}, z_j^{sl}) + \sum_{i=1}^{n_t} \sum_{j=1}^{n_t} w_i^{tc} w_j^{tc} K(z_i^{tl}, z_j^{tl}) - 2 \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} w_i^{sc} w_j^{tc} K(z_i^{sl}, z_j^{tl}) \right] \quad (9)$$

Where z_i denotes the feature vector extracted by the network at a specific location.

When pseudo labels are used to guide the network to achieve class-level matching, it is necessary to obtain the feature vectors of different domains and the corresponding label probability distribution. For the source domain, it can be computed directly from the feature vectors with the corresponding true labels. Due to the lack of true labels in the target domain, classifiers pre-trained from the source domain can be used to calculate the probability distribution of pseudo labels in the target domain. This method can avoid a negative impact on network performance to a certain extent. However, the performance of the classifiers is poor at the beginning of network training. If too many labels are misclassified, the network convergence may be adversely affected. Therefore, on this basis, the cosine similarity between different domains is used as the probability distribution of pseudo labels. Two different pseudo-label probability distributions are used to jointly calculate the LMMD. Thus, more accurate class-level matching of different domains is achieved.

A pre-trained classifier is used to calculate the pseudo-label probability distribution for the target domain, which is defined as:

$$\hat{y}_1^t = f\{D[F(x_j^t)]\} \quad (10)$$

Where $F(x_j^t)$ denotes the features extracted from target domain sample x_j^t by the feature extractor F . $D[F(x_j^t)]$ represents the flattening of features extracted from the sample x_j^t by the feature extractor F . f represents the classifier, which outputs prediction results based on the flattened features. j denotes the sample index. D denotes the fully connected layer of the first classifier. L_{LMMD-1} can be calculated according to Eq. (9).

When calculating cosine similarity, it is first necessary to calculate the class center of each class of samples in the source domain. The inputs to the network model are random and mini-batch. As the mini-batch does not necessarily cover all source domain categories, the class center of each sample class is calculated during the last iteration according to Eq. (11).

$$G_s^d = \frac{1}{T_s^d} \sum_{(x_j^s, y_j^s) \in D_s^d} F(x_j^s), d = \{1, 2, \dots, m\} \quad (11)$$

Where D_s^d indicates the set of samples with label d in the source domain. T_s^d denotes the number of samples labeled d . In each iteration, the local class center is calculated for a random mini-batch of samples. The calculation formula is as follows:

$$L_s^d = \frac{1}{b} \sum_{(x_j^s, y_j^s) \in D_s^d} F(x_i^s) \quad (12)$$

Where b indicates the batch size. Considering that each small batch may not contain all classes, set the balance parameter α to update the class centers of the source domain. The balance parameter α is expressed as follows:

$$\alpha = \text{COSS}(G_s^d, L_s^d) \quad (13)$$

Where $\text{COSS}(G_s^d, L_s^d)$ denotes the cosine similarity between G_s^d and L_s^d .

The source domain class center is as follows:

$$G_s^{d+1} = \alpha G_s^d + (1 - \alpha) L_s^d \quad (14)$$

Where d represents a specific class within the source domain samples. The target domain features extracted by feature extractor F cluster around the class centers of each source domain class in the latent feature space. The feature vector corresponding to the j th unlabeled sample in the target domain is x_j^t . The distance of this feature vector from the center of the source domain category is calculated. The equation is expressed as follows:

$$\sigma(x_j^t)_d = \text{COSS}[G_s^d, F(x_j^t)] \quad (15)$$

Where x_j^t is the feature of the j th data in the target domain. By calculating the cosine similarity between feature x_j^t and the

center of each class in the source domain, the pseudo-label probability distribution can be obtained. It can be expressed as:

$$\hat{y}_2^t = (\sigma_1, \sigma_2, \dots, \sigma_m) \quad (16)$$

Based on Eq. (9), L_{LMMD-2} can be calculated. In summary, two class-matching formulas can be obtained. The weight allocation of the two class-matching loss functions directly affects the network performance. In general, the value of the trade-off parameter is set based on expert experience. However, in the absence of expert experience, conducting tests one by one would increase the workload and computational burden. Therefore, we adopt a simple and efficient adaptive method. The trade-off parameters are adaptively adjusted by Eq. (17).

$$\beta = \frac{1}{1+e^{-0.1 \times iteration}} - 0.1 \quad (17)$$

Thus, the pseudo-label guided class-level matching loss function is shown in Eq. (18).

$$L_{CL-MMD} = \beta L_{LMMD-1} + (1 - \beta) L_{LMMD-2} \quad (18)$$

Additionally, in the stage of performing class-level matching, pseudo label reinforcement loss is introduced to further enhance the learning of the target domain by the network, as shown in Eq. (19).

$$L_t = -E_{(x_j^t, \hat{y}_j^t) \in D_t} \sum_{c=1}^C oh_{[\hat{y}_j^t=c]} \log F(x_j^t) \quad (19)$$

Where $oh(\sim)$ denotes the one-hot function. y_j^t denotes the pseudo label corresponding to feature x_j^t .

2.4. Network structure and parameters

The network contains three convolutional layers, three pooling layers, and two classifiers. We set the first two layers of convolution and pooling as one module. It is convenient for the subsequent shallow convolution to have different weights and parameters when learning common features in different domains. Through several debugging, the structure and parameters of the established network are shown in Table 1. “Valid” indicates that the edge is not expanded. “Same” means expanding the edge to prevent the image from getting smaller and smaller after the convolution operation. As an example, we demonstrate the process of hyperparameter selection using the learning rate. In this regard, we refer to previous network models with superior performance. The possible values for the learning rate were 0.0001, 0.001, and 0.01. The impact of different learning rates on the network's diagnostic performance is shown in Fig. 4. When the learning rate is 0.001, the network achieves optimal diagnostic performance. Therefore, we ultimately set the learning rate to 0.001.

Table 1. Details of the network.

Name	Layer	Sub-layer	Parameters	Feature map (output)	
Input	Input layer	-	64×64	64×64	
		Conv_1	2×2 , Stride=1, “Valid”	$67 \times 67 \times 32$	
		BN_1	Batch standardization layer	$67 \times 67 \times 32$	
		Maxpool_1	3×3 , Stride=2, “Valid”	$33 \times 33 \times 32$	
		ReLu_1	-	$33 \times 33 \times 32$	
		Conv_2	3×3 , Stride=1, “Same”	$35 \times 35 \times 32$	
		BN_2	Batch standardization layer	$35 \times 35 \times 32$	
		Maxpool_2	3×3 , Stride=2, “Valid”	$17 \times 17 \times 32$	
		ReLu_2	-	$17 \times 17 \times 32$	
		Feature extractor	Module	Conv_3/BN_3	3×3 , Stride=1, “Same”
Maxpool_3/ ReLu_3	3×3 , Stride=2, “Valid”			$4 \times 4 \times 32$	
Classifier_1 (C1)	FC_1/ BN_4	-	4096 (neurons)	$1 \times 1 \times 4096$	
		FC_2/ BN_5	-	3072 (neurons)	$1 \times 1 \times 2048$
		FC_3	-	categories	$1 \times 1 \times \text{categories}$
Classifier_1 (C1)	FC_2/ BN_5	-	4096 (neurons)	$1 \times 1 \times 4096$	
		FC_3	-	categories	$1 \times 1 \times \text{categories}$

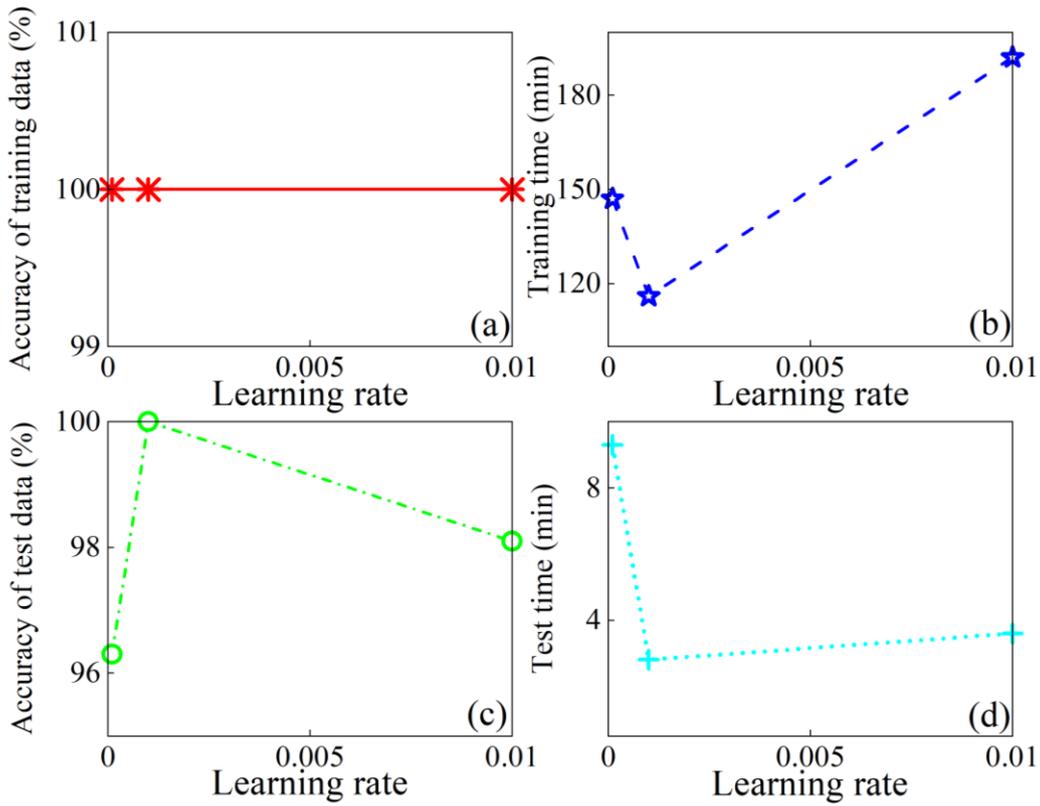


Fig. 4. The impact of learning rate on network performance: (a) training accuracy; (b) training time; (c) test accuracy; (d) test time.

2.5. Shallow independent - deep sharing strategy

When there is a relatively large distributional discrepancy between different domains, sharing the same weight parameters across domains reduces the adaptability of the network to the target domain, thereby impairing the transfer performance of the network. Inspired by multi-scale convolutional neural networks [35], [36], we design the first two convolutional layers of the network as a dual-path parallel convolutional structure. For different domains, the shallow structure of the feature extractor has different weight parameters. In other words, the weight parameters of the shallow structure are made more compatible with the properties of each domain. The branching structure takes into account the specific semantic information present in different domains. This information provides a key reference for the decision-making process of the network.

The shallow dual-path parallel architecture is shown in the second stage of Fig. 4. The network model is trained with source domain and target domain datasets. After reaching a predetermined number of iterations, the structure and weights of the module are replicated. Two identical modules are connected to the third convolutional layer in parallel. Two identical modules are fed into the source and target domain

datasets, respectively. It is important to note that the replicated module has to be pre-trained with source domain diagnostic knowledge, otherwise the transferability of the network model will be reduced.

2.6. Training procedure

When combined with MCD, network model training can be divided into two stages. The replacement process corresponds to the network model training process. Eq. (1) is replaced with Eqs. (20) and (21), Which are expressed as follows:

$$L_1 = \min_{F,C1,C2} L_{cls}(X_s, Y_s) + \lambda L_{MMD}, \quad (20)$$

$$L_2 = \min_{F,C1,C2} L_{cls}(X_s, Y_s) + \lambda L_{CL-MMD} + L_t \quad (21)$$

In Eq. (20), the feature distributions of the source and target domains are globally measured and reduced while ensuring the correct classification of the source domain. In Eq. (21), the target domain reinforcement loss and class-level matching loss are combined. Class-level matching of different domains can be facilitated while improving the confidence of pseudo labels in the target domain. To reduce the computational complexity and improve diagnostic efficiency, L_{MMD} , L_{CL-MMD} and L_t are calculated only in the first fully connected layer of the

classifier 1. The enhanced transfer CNN training process is shown in Fig. 5. The abbreviation “FC” in Fig. 5 denotes a Fully Connected layer, which is responsible for integrating high-level features from the preceding layers. Through the designed shallow dual-path parallel convolutional architecture, the

feature extractor has different weight parameters when extracting shallow features of different domains. As the depth of the network increases, different domains will share the weight parameters of the high-dimensional feature vectors.

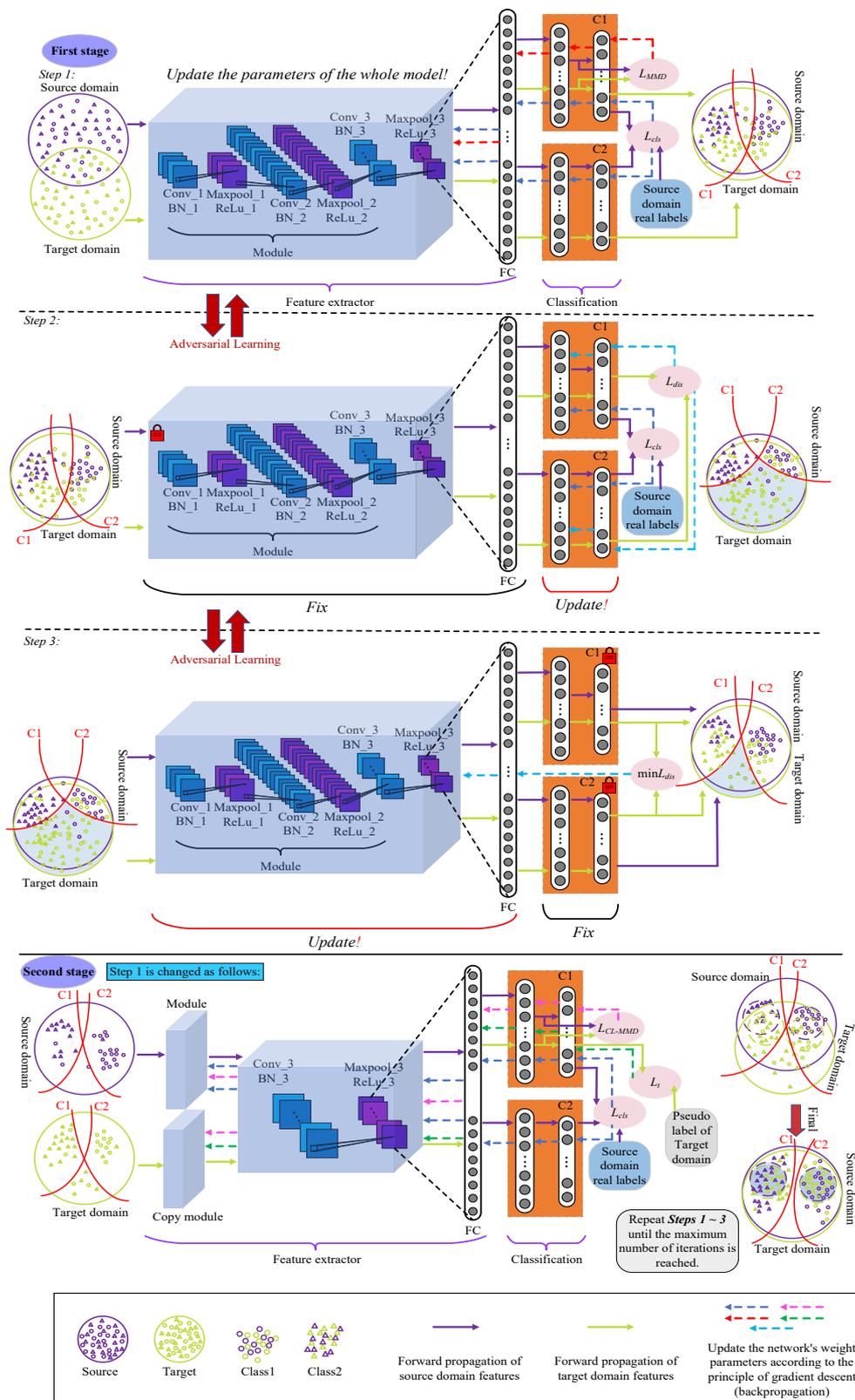


Fig. 5. Enhanced transfer CNN training process.

3. Case studies

3.1. Marine auxiliary blower experiment

3.1.1. Experimental equipment

The test object was an auxiliary blower driven by a three-phase asynchronous motor. The relevant parameters of the considered auxiliary blower are shown in Table 2.

Table 2. The main parameters of the marine auxiliary blower.

Rotation speed	1430 r/min
Rated voltage	380 V
Rated current	4.6 A
Rated power	1.1 kW
Total pressure	367~312 Pa
Power factor	0.89
Capacity	1500~2200 m ³ /h
Efficiency	59%

3.1.2. Experimental details

The vibration sensor is an INV9822A ICP accelerometer. The vibration sensor was installed on the front bearing housing of the three-phase asynchronous motor. The sampling frequency was 10.24 kHz. The sampling time was 5 seconds. This selection aims to balance data granularity and processing efficiency, ensuring sufficient resolution to capture relevant fault characteristics while minimizing computational complexity. The 5-second duration collects sufficient data

points within each operating cycle to identify key features associated with the fault state of marine auxiliary blowers.

A total of 7 different health states were set, namely normal condition (label_1), loose fan base bolts (label_2), loose motor base bolts (label_3), fan blade rotation imbalance - serious failure (label_4), fan blade rotation imbalance - minor fault (label_5), fan blocking (label_6), and compound fault (label_3 and label_5) (label_7). The auxiliary blower was operated at 25% load, 50% load, 75% load and 100% load. They were named as Domain A, Domain B, Domain C and Domain D, respectively. The different health states and domain information are shown in Table 3. Each health status in every domain contains 1,000 samples. The load of the auxiliary blower was changed by the motor frequency conversion unit. Minor fault condition (label_5) was created by removing one of the fixing bolts. Serious failure (label_4) was implemented by increasing the weight of the fixing bolt on the opposite side of the removed bolt. The foreign matter blocked the air outlet to simulate the label_6. The experimental platform is shown in Fig. 6. There are 12 cross-domain fault diagnosis tasks, the details of which are shown in Table 4. Fig. 7 shows the time domain waveforms of label_1~label_4 under domain A, domain B, domain C, and domain D.



Fig. 6. Auxiliary blower test platform and data acquisition.

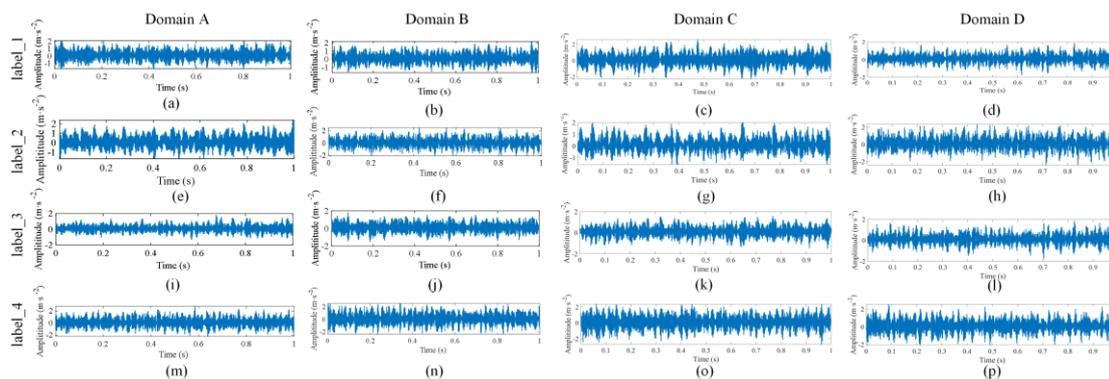


Fig. 7. Time-domain waveforms of label_1~label_4: (a) A_1; (b) B_1; (c) C_1; (d) D_1; (e) A_2; (f) B_2; (g) C_2; (h) D_2; (i) A_3; (j) B_3; (k) C_3; (l) D_3; (m) A_4; (n) B_4; (o) C_4; (p) D_4.

Table 3. Domain information for auxiliary blower fault diagnosis.

Fault type	Normal	Loose fan base bolts	Loose motor base bolts	Fan blade rotation imbalance (serious failure)	Fan blade rotation imbalance (minor fault)	Fan blocking	Compound fault (label 3 and label 5)
Label	1	2	3	4	5	6	7
Domain A	1000	1000	1000	1000	1000	1000	1000
Domain B	1000	1000	1000	1000	1000	1000	1000
Domain C	1000	1000	1000	1000	1000	1000	1000
Domain D	1000	1000	1000	1000	1000	1000	1000

Table 4. Auxiliary blower cross-domain test task types.

Source domain (labeled)	Target domain (unlabeled)	Target
A	B, C, D	Diagnosing unlabeled signals in the target domain
B	A, C, D	
C	A, B, D	
D	A, B, C	

3.1.3. Visual feature

The acquired vibration signals contain random interference

signals and other periodic high-frequency interference signals. To enhance the expression ability of signal feature information, we transformed 1-D features into 2-D image features. A 4-layer decomposition of the vibration signal is performed using the wavelet packet transform. The basis function is the db 4 wavelet. The rearrangement was performed according to the center frequency. The first five frequency bands (0~3200 Hz) were selected for analysis.

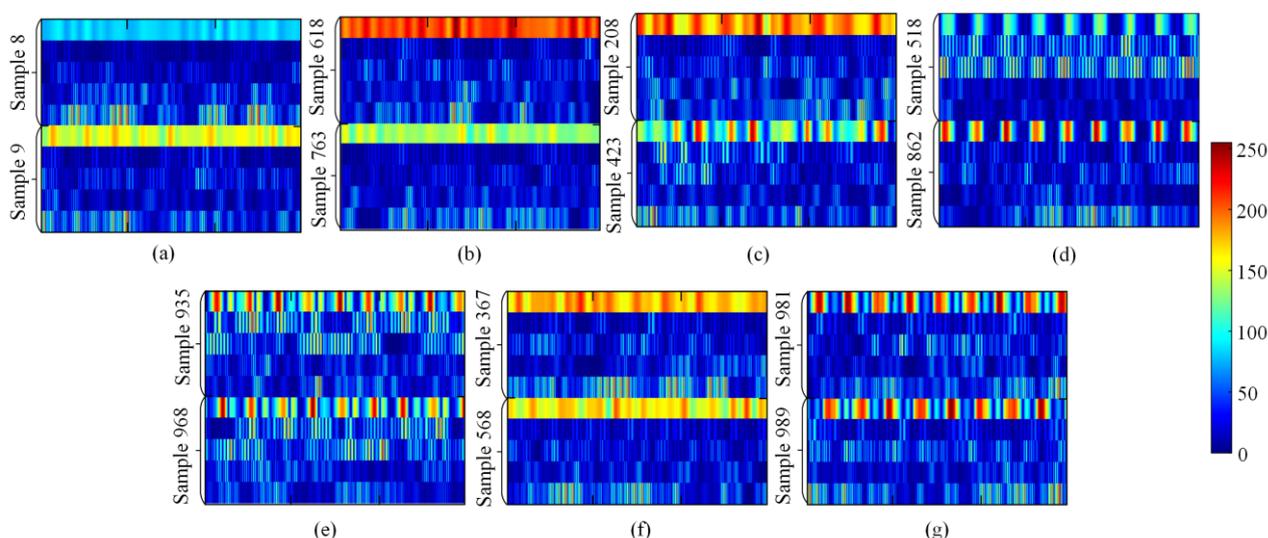


Fig. 8. Time-frequency images of vibration signals under Domain B condition: (a)~(g): label_1~label_7.

The time-frequency matrix was transformed into the time-frequency pseudo-color infographic by the wcodemat function and pseudo-color coding. Fig. 8 shows the images of time-frequency information for seven different health states under Domain B. For two-dimensional time-frequency images under other operating conditions, please refer to the appendix (Figures A1, A2 and A3). Features such as texture parameters and color distribution of the image will be presented in different forms when different faults occur. These features are closely related to the information about the operating conditions of the auxiliary blower, facilitating the network in mining referential and discriminative semantic information.

3.1.4. Network model performance

The hardware configuration for deep learning is AMD Ryzen 7 5800H CPU@3.2GHz, NVIDIA GeForce RTX 3060 Laptop GPU. The training set was input into the network. For 12 cross-domain diagnostic tasks, the training loss is shown in Fig. 9. The network was iterated 1000 times. The training loss of both classifiers gradually decreases and converges to a stable interval (0.001~0.0012).

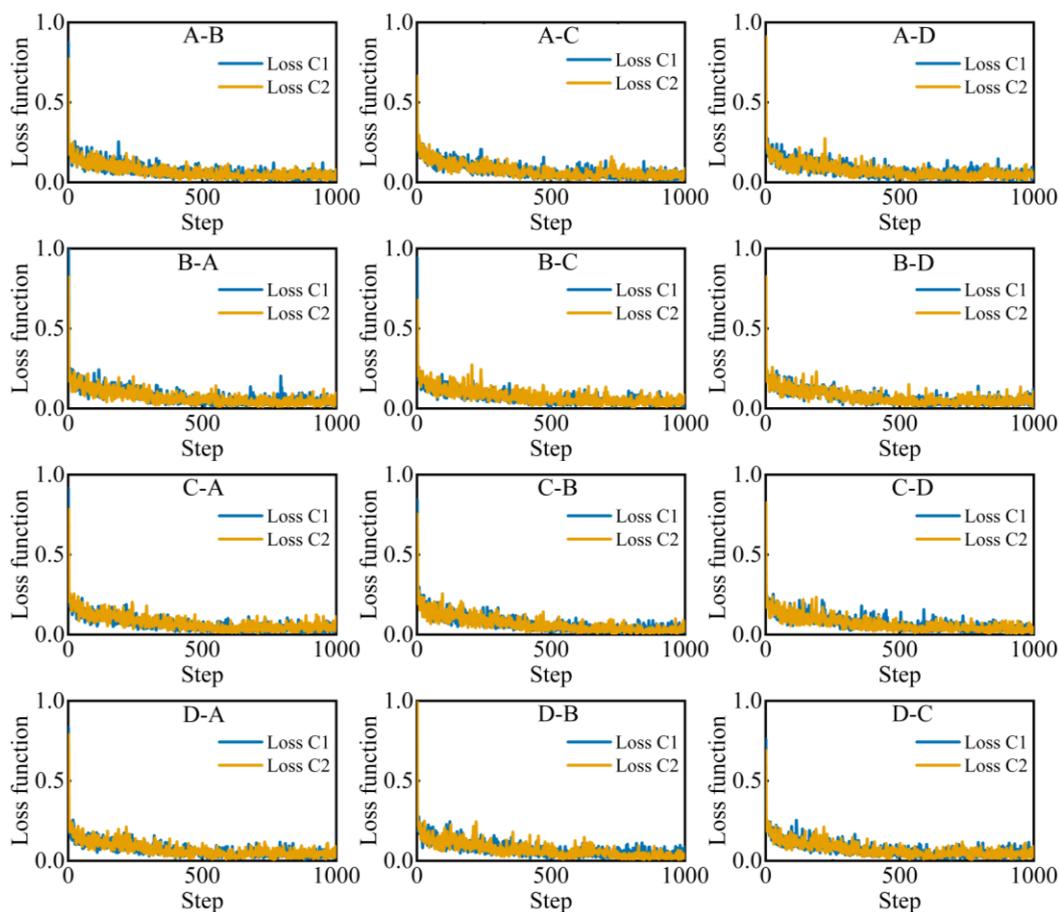


Fig. 9. Training loss for classifiers (C1 and C2).

Fig. 10 shows the discrepancy loss for the 12 transfer diagnostic tasks. With increasing training time, the discrepancy loss decreases continuously and converges to a stable interval. This is consistent with the goal of Step 3 of the training process.

Fig. 11 shows the training accuracy curve of the network model for the 12 transfer diagnostic tasks. The training accuracy of classifiers C1 and C2 is increasing and the final diagnostic accuracy can reach 100%. The diagnostic accuracy of the two classifiers for different transfer diagnostic tasks is shown in Fig. 12. The network was iterated 100 times. The proposed method performs well. Most cross-domain diagnostic tasks can be tested with over 97% accuracy. Only the test accuracy curves of D-A, C-A, and D-B have large oscillation amplitude compared to A-D, A-C and B-D. It may be due to the fact that the network can learn some more generalized features when low load is used as the source domain. These features show better transfer capability when high load is the target domain. Conversely, high-load datasets usually involve more dynamically varying information, such as nonlinear features, which poses a challenge for the network to learn domain invariants.

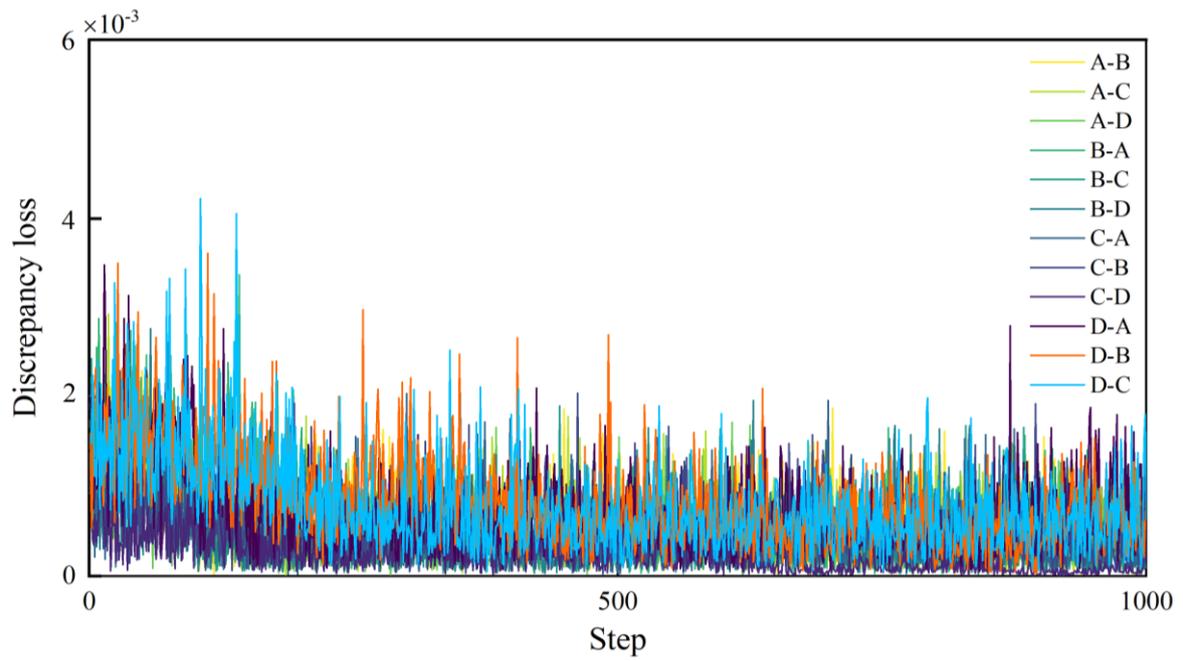


Fig. 10. Discrepancy loss of transfer diagnostic tasks.

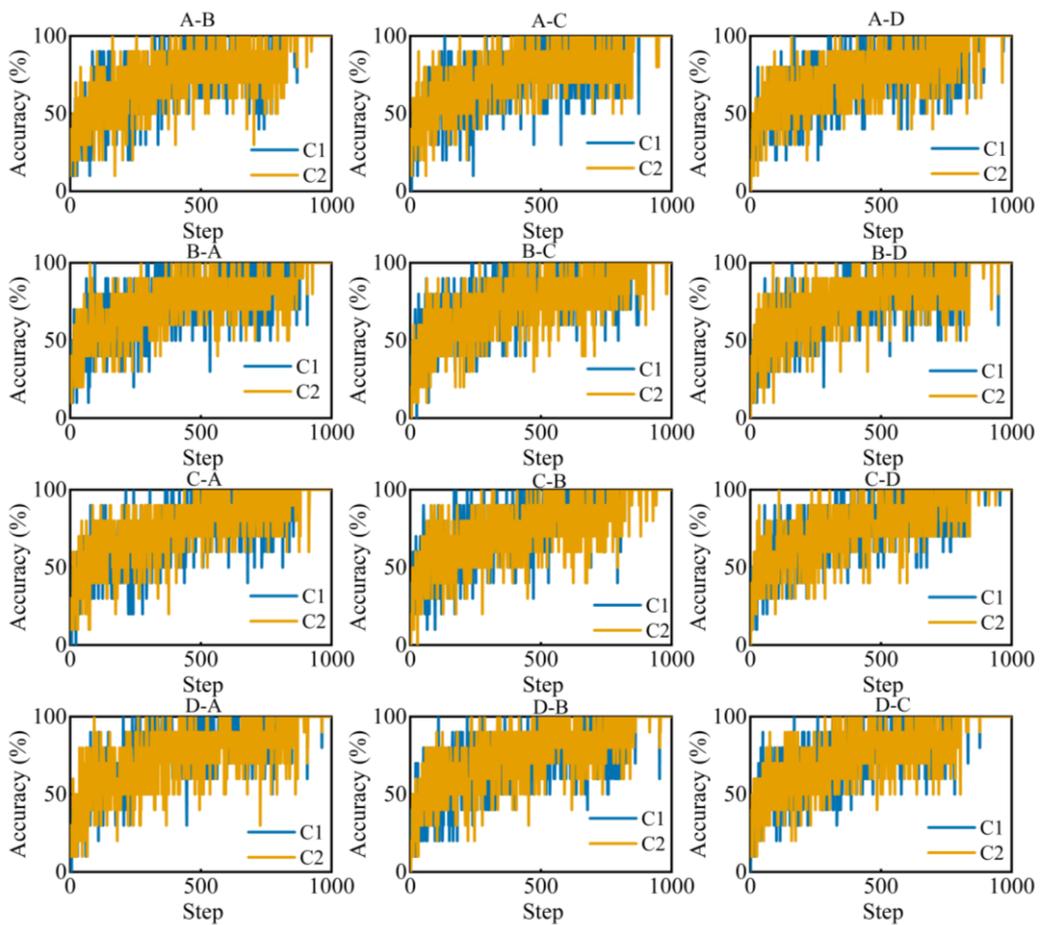


Fig. 11. Training accuracy curves for 12 transfer diagnostic tasks.

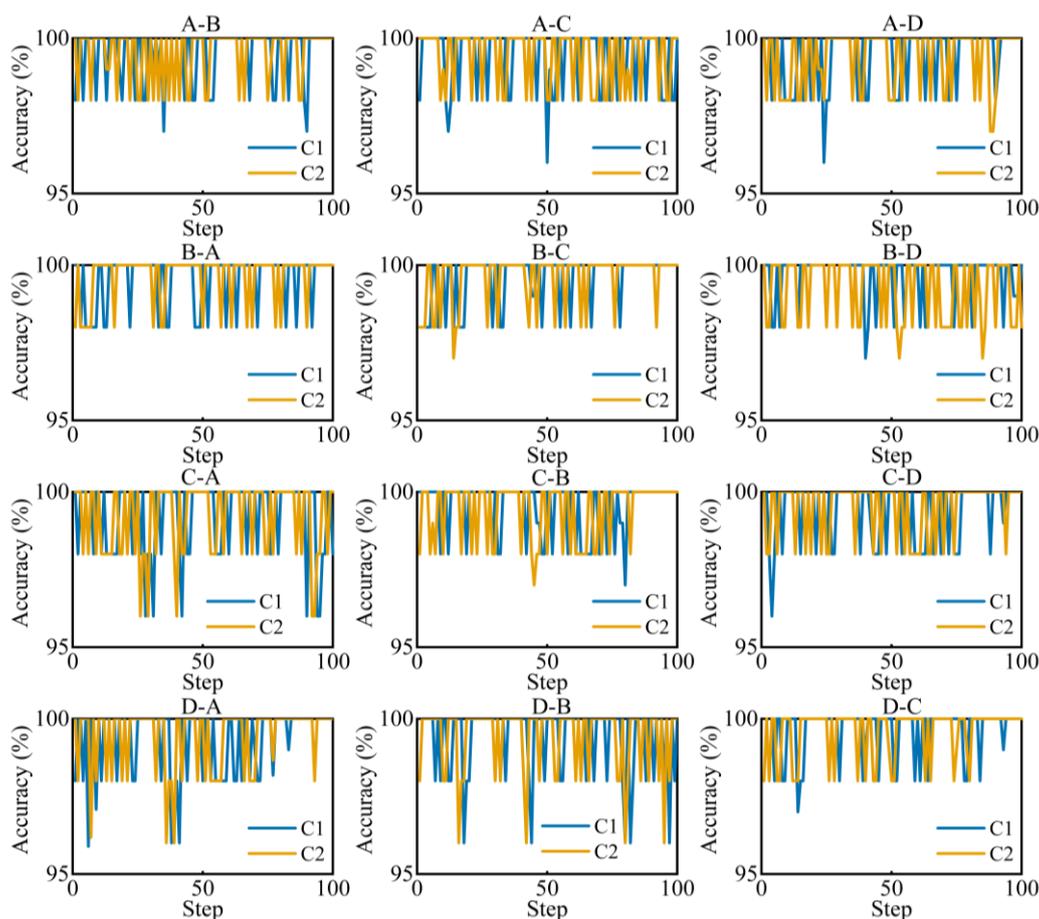


Fig. 12. Test accuracy curves for different transfer fault diagnosis tasks.

To further validate the transferability of the network, four diagnostic tasks are analyzed as examples and the confusion matrix is shown in Fig. 13. The horizontal direction of the confusion matrix is the predicted labels and the vertical direction represents the true labels. The accuracy for a specific fault type is calculated as the ratio of correctly predicted samples to the total samples of that type. The accuracy of a given diagnostic task is calculated as the proportion of correctly predicted samples to the total samples. In the four diagnostic tasks, the network was able to achieve 100% diagnostic accuracy for label_6. Additionally, a small number of label_3 samples are misclassified as label_2 in all four confusion matrices. The faults corresponding to Label 3 and Label 2 are both characterized by structural looseness, which induces similar dynamic behaviors. These two fault types exhibit analogous semantic features in the feature space, making it challenging for the network to effectively distinguish them during the classification process. The similarity in these feature patterns results in a high degree of overlap in the decision boundary, which is particularly pronounced under conditions of

domain shift and noise. From an engineering perspective, this specific form of confusion can have significant implications, potentially leading to misalignment in maintenance priorities. The risk of a loose motor base is of particular concern, as it may affect the alignment of the drive shaft, and if not promptly addressed, could result in secondary damage to the motor or fan system. To address this issue, future engineering efforts could focus on improving the feature extraction process to enhance the network's ability to distinguish between these subtle but critical fault types. For instance, more advanced domain adaptation techniques or additional sensor data could be incorporated to provide additional discriminative power for these faults.

The activation of neurons in the convolutional layer was visualized and analyzed using label_6 in the A-B diagnostic task. The results are shown in Fig. 14. It can be seen that most neurons can be activated in all three convolutional layers. There are several red bands in the Conv_1. This indicates that feature information such as texture features and color distribution of the image has been effectively extracted. Conv_2 and 3 further

extract high-dimensional features and filter them. Inactivated neurons are filtered out to highlight important features. Finally, the classifiers output the classification results as one-hot coding.

This further validates that the network can be effectively applied to the target diagnosis tasks.

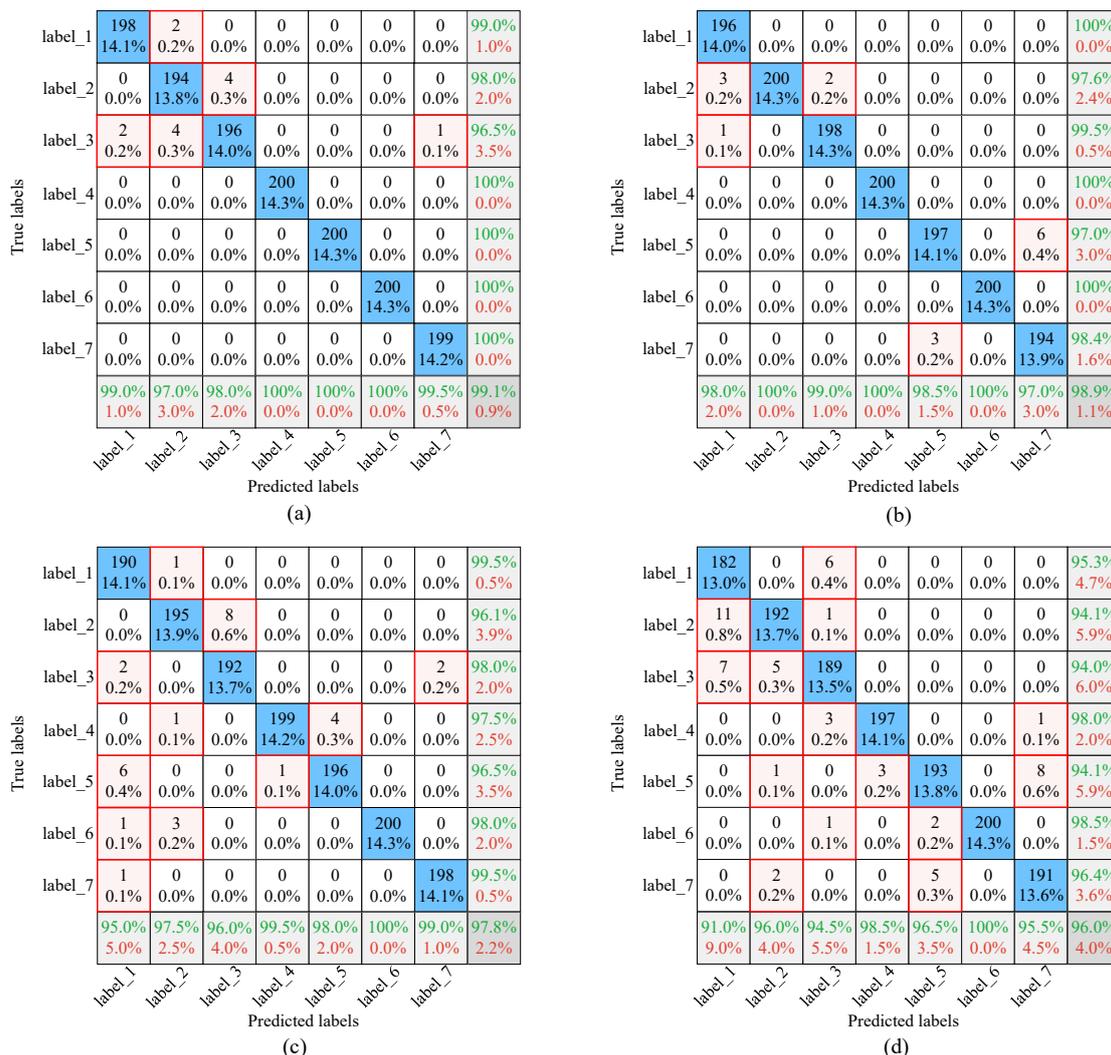


Fig. 13. Confusion matrix for four diagnostic tasks: (a) A-B; (b) A-D; (c) A-C; (d) D-B.

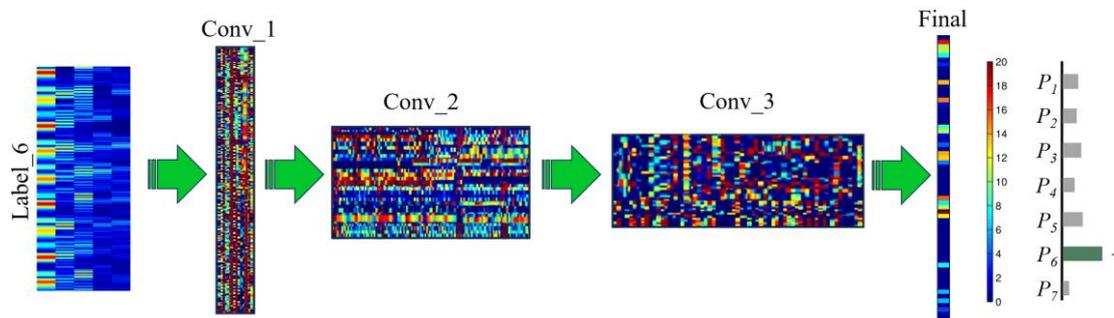


Fig. 14. Visualization of neuron activation states in the convolutional layer.

3.1.5. Ablation studies

The A-B transfer diagnostic task was taken as an example to test the effects of class-level matching, dual-path parallel architecture, and target domain correction loss on the network model. The real-time diagnostic accuracy of the network is

shown in Fig. 15. It can be observed that the absence of class-level matching causes the largest degradation in performance, with the accuracy curve dropping rapidly and remaining at a lower level. This confirms that class-level alignment is the most critical component, as it explicitly reduces the intra-class

discrepancy between source and target domains and prevents feature overlap across categories. Without this module, the extracted target features are less discriminative, leading to significant misclassification. When the dual-path parallel architecture is removed, the diagnostic accuracy not only decreases but also exhibits stronger fluctuations during training. This indicates that domain-specific shallow feature extraction is essential for stabilizing the adaptation process. The lack of parallel shallow branches forces the model to share identical parameters across domains, which limits its ability to capture domain-specific semantics and reduces transferability. The target domain correction loss also plays an important role in maintaining robustness. Without the target domain correction loss (L_t), the initial diagnostic accuracy would rise very quickly. However, after 596 iterations, the diagnostic performance of the network seems to have entered a bottleneck. Although this is not

the case in every experiment, it can increase the instability of diagnostic results. This suggests that correction loss enhances the reliability of pseudo labels and avoids error accumulation during iterative training, thereby supporting stable performance in the target domain. Through the synergistic collaboration of various mechanisms, the network's diagnostic performance ultimately achieves optimal results. It is important to note that the authors do not emphasize that more network improvements or additional constraints necessarily lead to better performance. Excessive constraints can limit the network's flexibility, potentially causing negative transfer or performance degradation. Finding the optimal balance is crucial. In summary, the proposed method can assign different transfer weight parameters to different domains and can effectively match the global and subclass feature information.

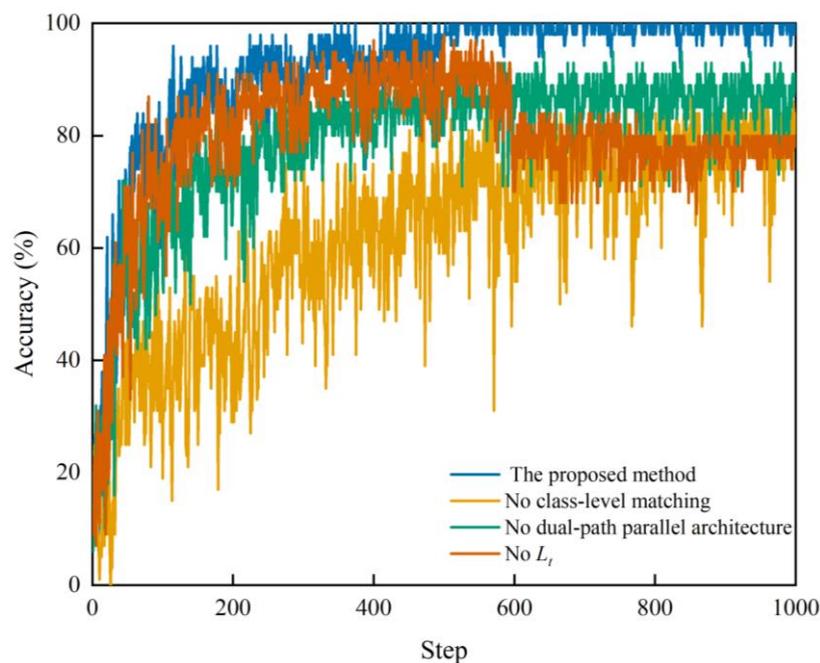


Fig. 15. Real-time diagnostic accuracy of ablation experiments.

3.2. Fuel injector experiment

3.2.1. Experimental design

The technical parameters of the marine diesel engine are shown in Table 5. In the injector fault experiment, two types of signals were collected: (1) pressure signals; (2) vibration signals from the inner wall of the high-pressure fuel injection pipe. The vibration sensor is an INV9828 ICP-type accelerometer. The pressure sensor type is Kistler 4067E3000DS. The pressure

measurement range is 0~3000 bar. The piezoresistive single-channel amplifier type is 4624 AK. The installation adaptor type is 6533.

Table 5. Technical parameters of marine diesel engine.

Item	Parameter
Engine type	MAN B&W 6S35ME-B9
Firing order	1-5-3-6-2-4
Rated speed	142 r/min
Rated power	3570 kW
Compression ratio	21

The sampling frequency was 2048 Hz. The high-pressure fuel line features a double-layer structure. In the experiment, the outer protective layer of the high-pressure fuel line was stripped away, leaving only the inner wall intact. The inner wall was then drilled to accommodate the installation of a pressure sensor. To ensure measurement accuracy and stability, an installation

adaptor was used to securely fasten the pressure sensor in the drilled location. The signal acquisition platform and sensor installation positions are shown in Fig. 16. Experiments conducted on the fuel injector of Cylinder 1. The fuel injector test was conducted on the diesel engine.

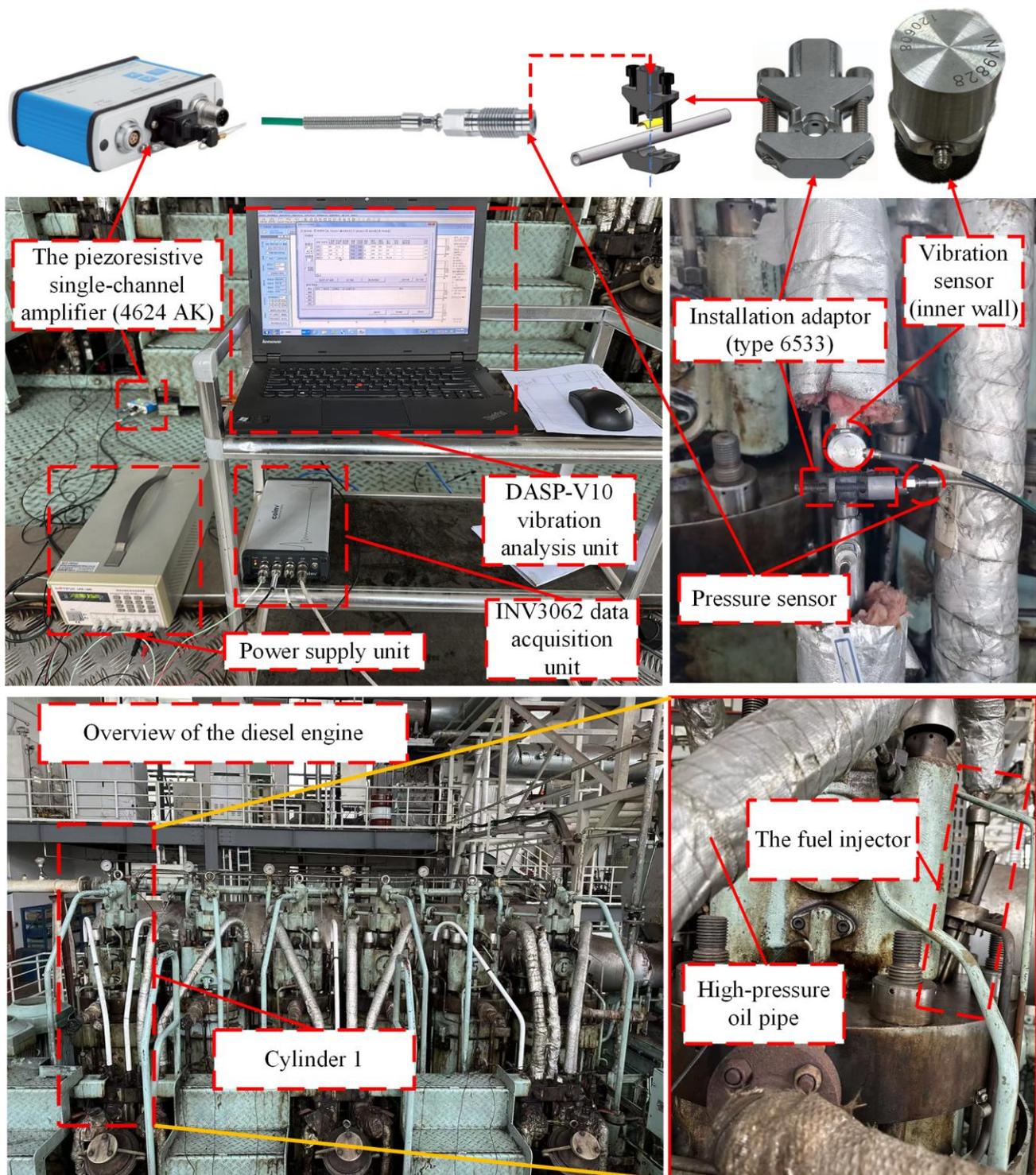


Fig. 16. Fuel injector failure experiment platform.

In the experiment, two fault signals of the fuel injector were collected under low, medium, and high loads. Each load

contains two fault states and one normal state. The fuel injector experimental data set is described in Table 6. Each health status

across different domains contains 1200 samples. The diesel engine speed was 90 r/min. The diesel engine load was changed by the hydraulic dynamometer. The normal opening pressure is 350 bar. The valve opening pressure was reduced by the pressure adjustment shims. The low valve opening pressure was

set to 300 bar. It should be noted that the above pressure values were measured using an injector test bench. One fuel injection hole was welded by argon arc welding to simulate nozzle blockage. The information of the experiment is shown in Fig. 17.

Table 6 Fuel injector data category information.

Fault type	Normal	Low fuel injector valve opening pressure	Clogged nozzle	Load
Label	1	2	3	
Dataset A	1200	1200	1200	30%
Dataset B	1200	1200	1200	50%
Dataset C	1200	1200	1200	70%

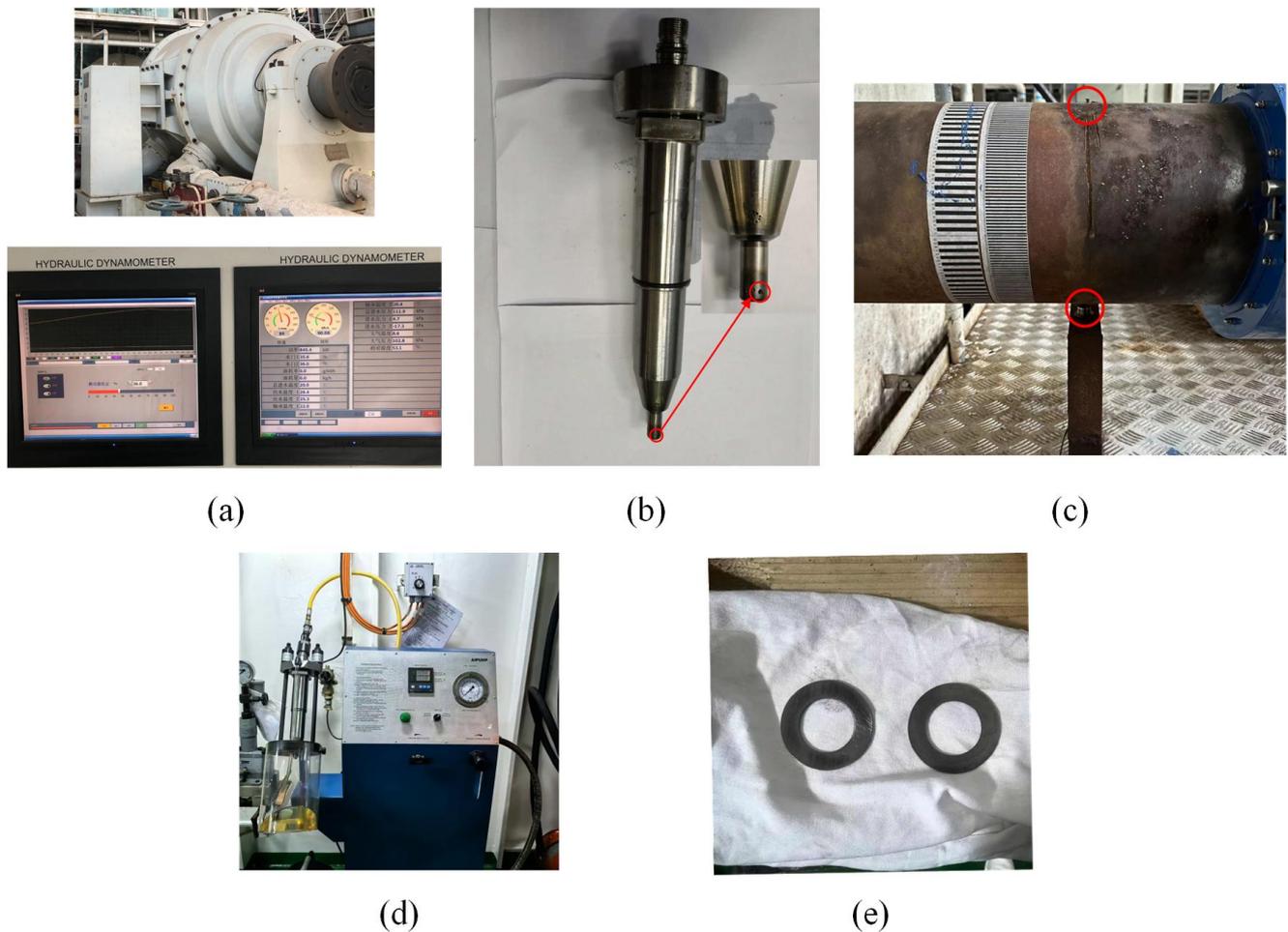


Fig. 17. Detailed information: (a) hydraulic dynamometer; (b) fuel injection hole clogging; (c) TDC sensor; (d) fuel injector test bench; (e) adjusting shims.

Fig. 18 shows the pressure signals, vibration signals, and TDC signals for dataset C (70% load). Signals for Datasets A and B can be found in the appendix (Figures A4 and A5). When the opening pressure is low, the amplitude of the pressure signal and the vibration signal are slightly reduced at the moment of fuel injection. The fuel injection duration increases. When

nozzle blockage occurs, the amplitude of the vibration and pressure signals is elevated compared to normal. During the injection cycle, it causes marine fuel oil to generate a strong pressure wave in the high-pressure pipe.

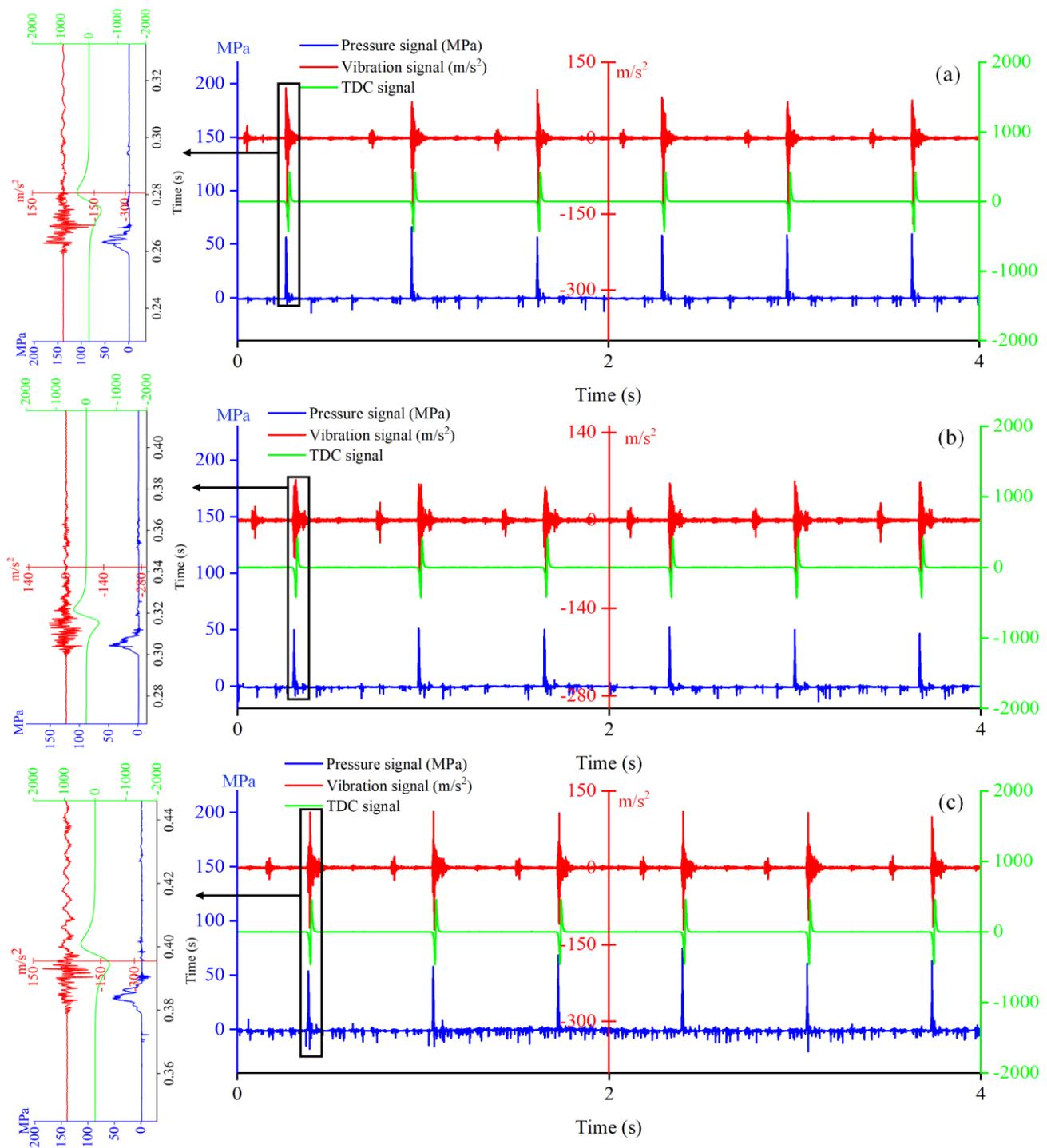


Fig. 18. Signals of different health states: (a) normal state; (b) low valve opening pressure; (c) nozzle blockage.

3.2.2. Multi-data time-frequency infographics

To integrate the rich feature information, the multi-source data are fused into a time-frequency infographic. Four-layer wavelet packet decomposition was performed on the vibration and pressure signals. The first five sub-bands were reconstructed and used as the data matrix for fault analysis. The sub-bands of each signal were arranged from top to bottom in the order of vibration signals and pressure signals. Taking

dataset A (Load 30%) as an example for analysis, Fig. 19 shows the time-frequency infographics of the signal (one cycle). The time-frequency information plots for datasets B and C are shown in the appendix (Figures A6 and A7). Features such as texture parameters and color distribution of the image will be presented in different forms when different faults occur. These features are closely related to fuel injector health status information.

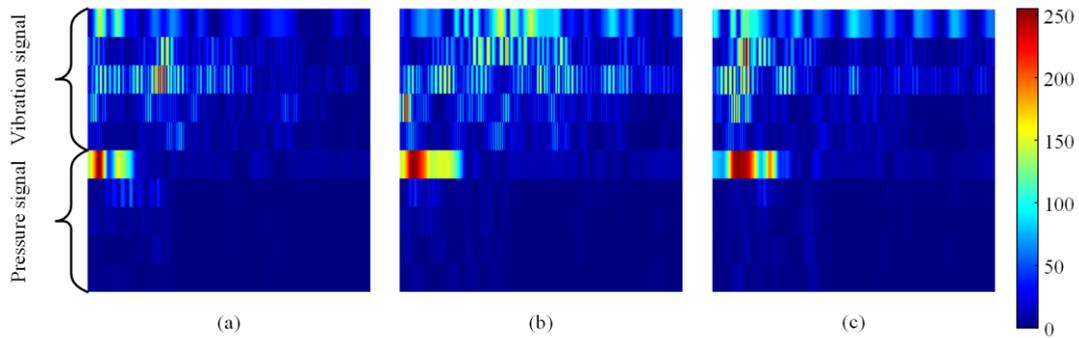


Fig. 19. Time-frequency information graph: (a) normal state; (b) low valve opening pressure; (c) clogged nozzle.

3.2.3. Transfer tasks diagnostic results

To visualize the diagnostic accuracy of the network for different health states, Fig. 20 shows the test accuracy for different transfer diagnostic tasks. The network model has high diagnostic accuracy for both faults. The average diagnostic accuracy can reach 93.37%. In task A-B, 100% diagnostic accuracy is achieved for the normal state. A greater proportion of misclassification occurs in normal condition and low valve opening pressure failure.

The visualization results for tasks A-B, B-A, and C-B are

given in Fig. 21. It can be seen that the proposed method reduces the distribution difference between source and target domains by extracting transferable features. With the introduction of class-level matching, the alignment of source and target domain features of the same category works well. The feature clusters are compact and well-differentiated. Only a small number of samples are misclassified. The reason for this may be the presence of similar feature components between the samples. This places higher demands on the diagnostic framework in terms of extracting distinguishable features.

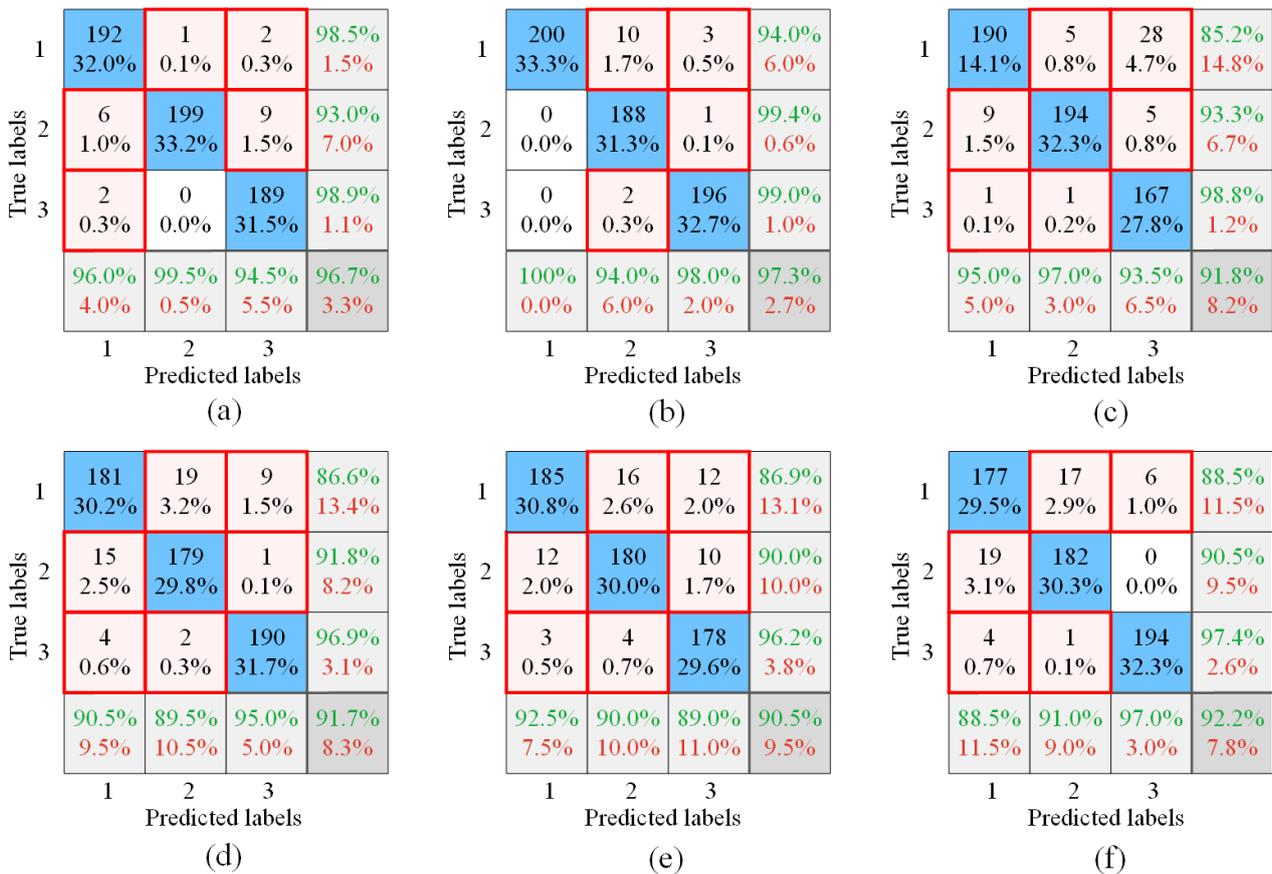


Fig. 20. Confusion matrix for the six test tasks: (a) A-B; (b) A-C; (c) B-A; (d) B-C; (e) C-A; (f) C-B.

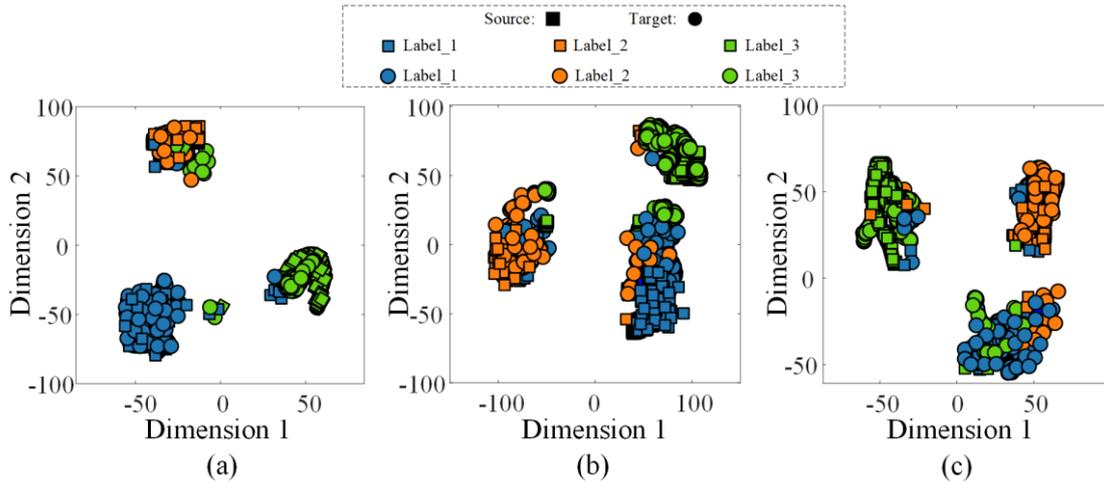


Fig. 21. Visualization results for the three tasks: (a) A-B; (b) B-A; (c) C-B.

3.3. Diagnostic performance comparison

In an effort to further validate the performance of the network model, the proposed diagnostic framework was analyzed in comparison with other methods. The comparison methods mainly include 2D-CNN, Deep Adaptation Network (DAN) [25], MDD [29], Deep Coral (DC) [37], and MCD [38]. This 2D-CNN is fully consistent with the developed network framework, but contains only one classifier and one feature

extractor. Specific parameters are detailed in Table 1. For a complete description of the methods and detailed hyperparameter settings, please refer to the relevant references. The network architecture of the different methods is shown in Fig. 22. The inputs to the DAN, MDD, and DC methods are the 1-D signals. The 2D-CNN has only one classifier. The inputs for other methods are consistent with the inputs for the developed network.

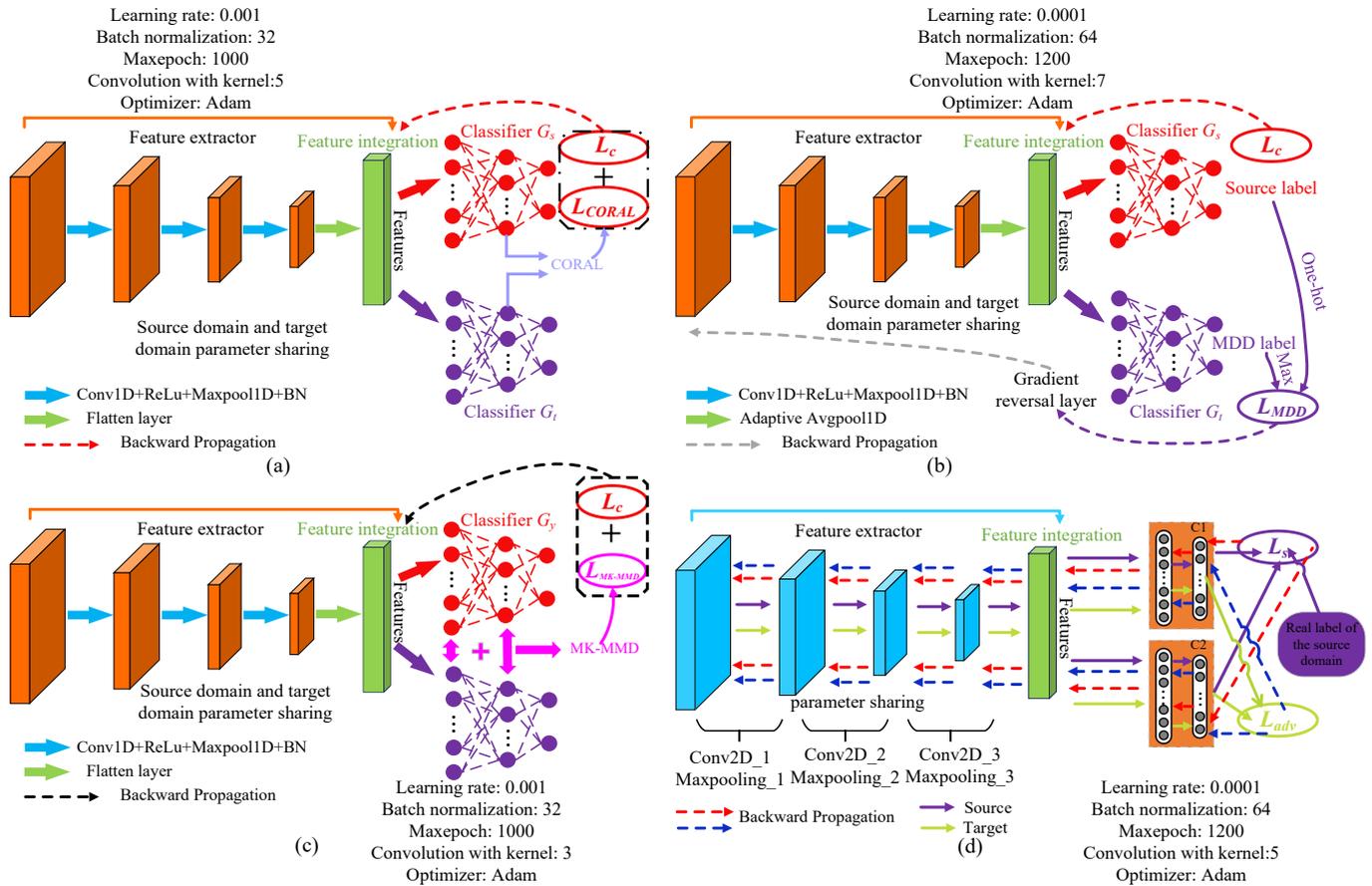


Fig. 22. The network structure of the different methods: (a) DC; (b) MDD; (c) DAN; (d) MCD.

3.3.1. Comparative analysis of auxiliary blower experiment

Each method was run ten times independently for each diagnostic task, and the average result was taken as the final diagnosis. The diagnostic results of the different methods on the marine auxiliary blower dataset are shown in Table 7. In diagnostic scenarios with significant domain distribution discrepancies, the proposed framework exhibits better adaptability in the target domain than other methods. The diagnostic accuracy of the proposed method is superior to other methods in most transfer tasks. The average diagnostic accuracy for the 12 transferred tasks exceeded 98.00%, which is 1.55% and 2.07% higher than the MDD and MCD methods, respectively. The core of the MDD method is to minimize the differences in classification boundaries of different domains,

but it ignores the effect of classification boundaries on feature matching in the target domain. Especially in diagnostic tasks with large domain shifts (C-A, D-B, D-A), the MDD method may not be able to effectively align the decision boundaries of different domains. There is no L_t and shallow dual-path parallel architecture in the MCD method. The L_t and dual-path parallel architecture can guide the model in real-time to extract the common features of different domains and assign different transfer weight parameters to different domains. Thus, the adaptability of the network model is improved. Compared with 2D-CNN, the proposed method has better transfer performance in tasks A-C and D-B. This demonstrates that introducing bi-classifier and adversarial learning mechanisms can be more effective in reducing the distributional differences between different domains.

Table 7. The transfer diagnostic results of auxiliary blower experiments.

Task	2D-CNN	DAN	DC	MDD	MCD	Our method
A-B	88.63 ± 3.23	91.42 ± 4.61	100.00	98.82 ± 0.83	100.00	99.10 ± 0.27
A-C	77.92 ± 2.24	89.95 ± 2.90	85.26 ± 1.43	97.23 ± 1.32	90.20 ± 4.76	97.80 ± 1.28
A-D	90.21 ± 1.39	95.62 ± 4.18	96.50 ± 1.12	98.96 ± 0.34	100.00	98.90 ± 0.51
B-A	89.95 ± 0.86	93.78 ± 2.83	98.49 ± 1.62	99.85 ± 0.13	98.53 ± 1.08	100.00
B-C	92.38 ± 1.52	91.10 ± 3.14	89.77 ± 2.33	96.33 ± 2.12	99.01 ± 0.36	99.59 ± 0.06
B-D	83.50 ± 0.81	89.82 ± 2.79	90.15 ± 1.13	96.46 ± 2.36	97.45 ± 3.51	100.00
C-A	81.43 ± 1.08	95.29 ± 3.27	87.35 ± 1.27	92.89 ± 5.74	93.21 ± 2.59	95.35 ± 1.87
C-B	93.21 ± 1.69	88.26 ± 2.16	91.01 ± 1.87	99.02 ± 0.28	98.20 ± 1.00	100.00
C-D	86.29 ± 2.73	100.00	90.87 ± 1.94	97.09 ± 1.43	100.00	98.90 ± 0.47
D-A	92.18 ± 3.01	94.95 ± 2.21	95.06 ± 4.02	96.02 ± 2.68	95.60 ± 3.67	98.87 ± 1.03
D-B	85.30 ± 3.22	80.03 ± 1.80	88.61 ± 1.56	93.53 ± 5.03	89.90 ± 5.86	96.00 ± 2.62
D-C	87.02 ± 3.62	95.47 ± 1.13	90.12 ± 2.63	99.69 ± 0.19	97.60 ± 1.01	100.00
Average	87.33	92.14	91.93	97.16	96.64	98.71

The output layers of the different methods were analyzed for t-SNE visualization (taking task A-B as an example). The results are shown in Fig. 23. The proposed method can cluster the features of each class. Compared with other methods, the proposed framework is able to fully utilize the fine-grained feature information of source and target domains. The intra-class interval is small, and the inter-class interval is large. Label_1 has a small number of samples misclassified into label_3. Label_2 has a small number of samples misclassified

into labels_1 and 3. It may be due to the correlation of features from different labels, which can expose the network to the effects of multicollinearity. The decision-making of the network may be biased towards certain features, leading to misclassification. Other methods have varying degrees of misclassification. The feature boundaries overlapped, and discriminative class-level boundaries were not obtained. This further validates that the network model has strong transferability.

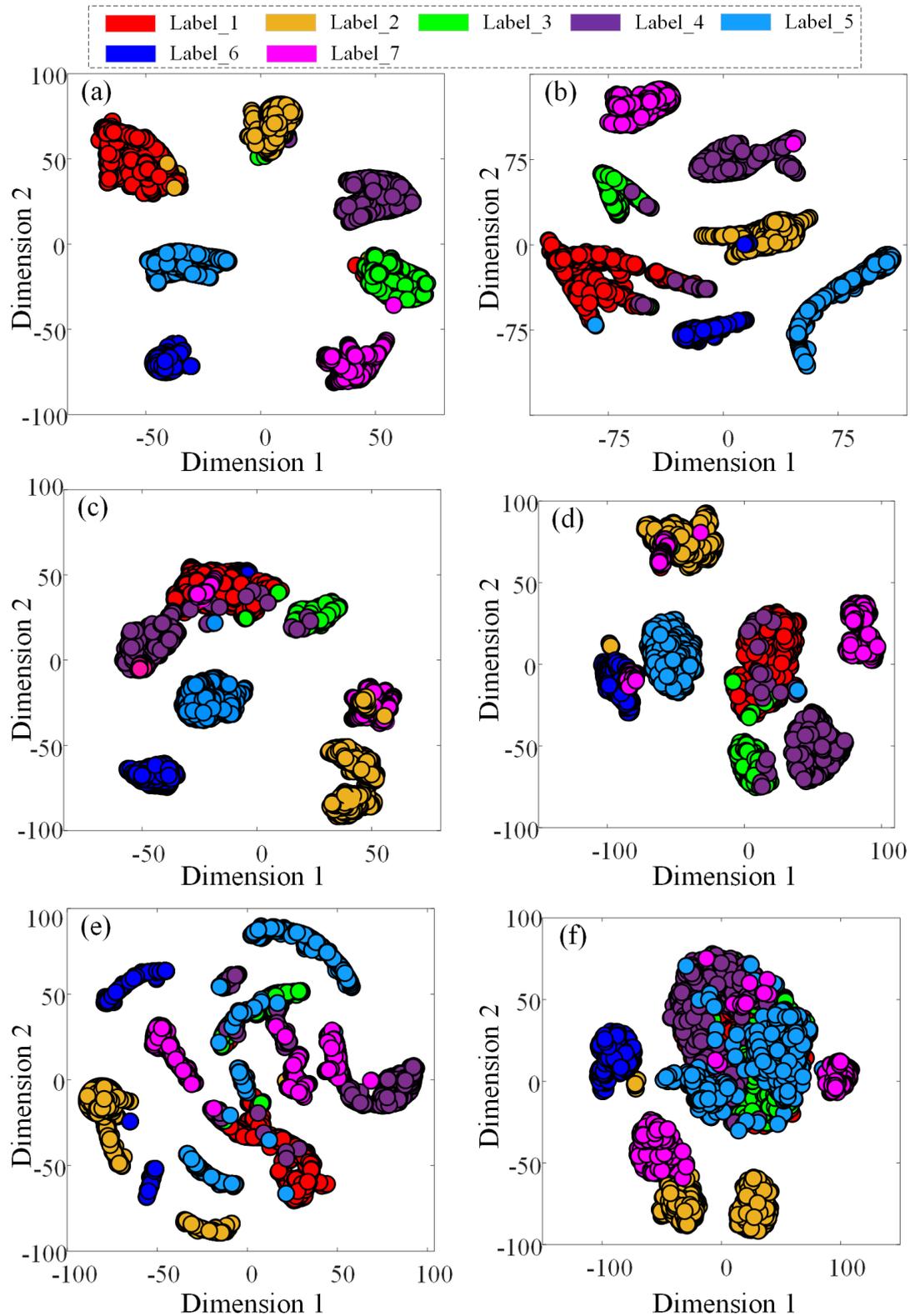


Fig. 23. Visualization results of different methods at the output layer: (a) the proposed framework; (b) MDD; (c) MCD; (d) DAN; (e) DC; (f) 2D-CNN.

To further validate whether the performance improvement of the proposed method is statistically significant, we conducted paired t-tests between the proposed framework and each comparison method. The results are shown in Figure 24. The

highest p-value (0.0392) was observed between the proposed method and the MDD method. All p-values were below 0.05, indicating that the proposed method exhibits significant differences compared to the other methods.

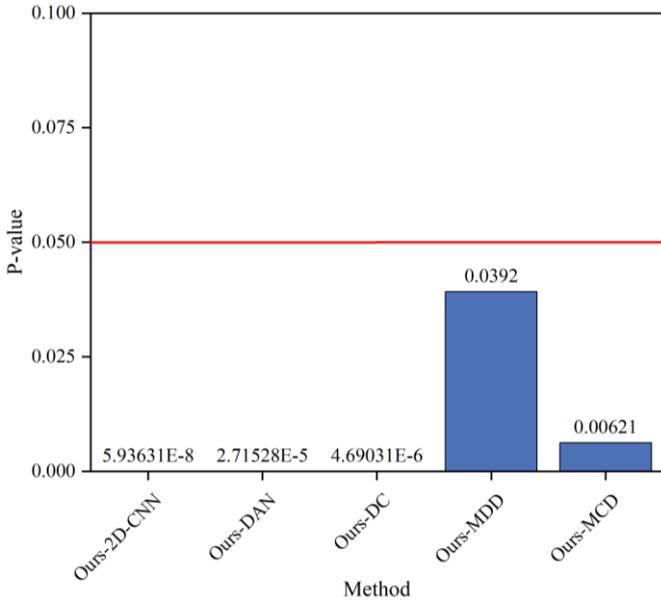


Fig. 24. Results of paired t-tests between the proposed method and the comparative methods.

3.3.2. Robustness evaluation

To directly validate the robustness of the proposed diagnostic framework under extreme environmental variations, this section simulates scenarios such as sensor interference or increased channel noise. We constructed test sets with varying Signal-to-Noise Ratios (SNR) by adding Gaussian white noise to the original vibration signals, enabling quantitative evaluation of each diagnostic method's interference resistance. Taking the auxiliary fan cross-domain task $C \rightarrow A$ as an example, we selected the test set from the target domain (Domain A) as the baseline data for noise addition. Based on this, five noisy test sets were generated with SNR values of -20 dB, -10 dB, 0 dB, 10 dB, and 20 dB. Subsequently, models trained on clean data (including the proposed method, DAN, DC, MDD, and MCD) were tested on these noisy datasets. Each experiment was repeated three times, and the average diagnostic accuracy was

recorded, as shown in Fig. 25.

The results indicate that the diagnostic accuracy of all methods decreases as the signal-to-noise ratio (SNR) is reduced. Nevertheless, the proposed framework consistently achieves the best performance, reaching 92.8% accuracy at 0 dB and 83.12% at -20 dB, which is markedly higher than MCD, MDD, DAN, and DC. These findings confirm that the shallow-independent-deep-shared architecture and the class-level matching strategy effectively alleviate distribution discrepancies induced by environmental variations, thereby ensuring robust cross-domain diagnostic capability even under extreme conditions.

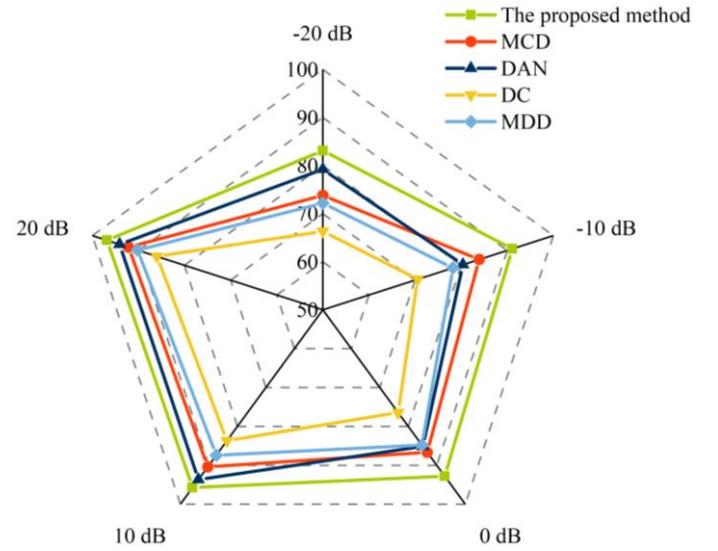


Fig. 25. Diagnostic accuracy comparison of different methods for the auxiliary fan $C \rightarrow A$ task under varying SNRs.

3.3.3. Computational efficiency analysis

To comprehensively evaluate the performance of different methods, a comparative analysis of computational complexity and efficiency has been added. Taking diagnostic task $A \rightarrow B$ as an example, the detailed results are shown in Table 8.

Table 8. Comparative analysis of computational efficiency and model complexity.

Method	Params (M)	FLOPs (G)	Training time (min)	GPU Memory (GB)
The Proposed method	2.43	5.81	118.3	3.2
2D-CNN	2.12	4.98	98.7	2.8
DAN	2.43	5.76	121.6	3.3
DC	2.40	5.70	100.1	3.2
MDD	4.91	8.21	140.6	5.1
MCD	4.24	9.96	128.6	4.9

The proposed method achieves a good balance between

performance and computational cost. It maintains a moderate

model size (2.43M parameters) and low computational complexity (5.81 G FLOPs), with a training time of 118.3 minutes and GPU memory usage of only 3.2 GB. Compared to advanced domain adaptation methods such as MDD and MCD, which require significantly more parameters and resources, the proposed framework reduces overhead by over 35% while delivering superior diagnostic accuracy. This highlights its efficiency, scalability, and suitability for practical applications in cross-domain fault diagnosis.

3.3.4. Comparative analysis of fuel injector experiment

Different diagnostic tasks are analyzed as an example, and Fig. 26 shows the real-time accuracy curves of the different methods. The stability of the proposed method is better than other transfer learning methods. In addition, the accuracy curves of 2D-CNN, DC, and DAN fluctuate more in the four transfer tasks (A-B, B-A, B-C, C-B). Since the DC method only uses second-order statistics to measure differences. It leads to poor transfer performance. The DAN approach to cross-domain feature alignment based on MMD requires pseudo-labels to measure the MMD distance. During the training process, if the network model overfits the source domain data, it will lead to mislabeling of pseudo-labels and generate negative transfer.

The diagnostic accuracy of the different methods for the six transfer tasks is shown in Fig. 27. Compared with 2D-CNN, DAN, DC, MCD, and MDD methods, the proposed method achieved average diagnostic accuracy improvements of 9.56%, 5.69%, 7.20%, 2.40%, and 1.91%, respectively. It can be noted that the diagnostic accuracy of the “smaller load - larger load” transfer task is higher than that of the “larger load - smaller load”. The probable reason is that the datasets acquired at low-speed and low-load, which contain more foundational domain-invariant semantic information. Even at high-speed and high-load scenarios, these common features still remain effective. Conversely, datasets collected under high-speed and high-load conditions exhibit greater complexity and specificity, which may limit their applicability in transfer diagnosis. The comparative analysis shows that the proposed method has higher testing accuracy than other methods in most diagnostic tasks. On the one hand, the proposed method can better align the class-level space of source and target domains under the condition of utilizing class-level fine-grained information. On the other hand, target domain correction loss function and dual-path parallel architecture help to improve the learning ability and adaptability of the network to unlabeled target domain.

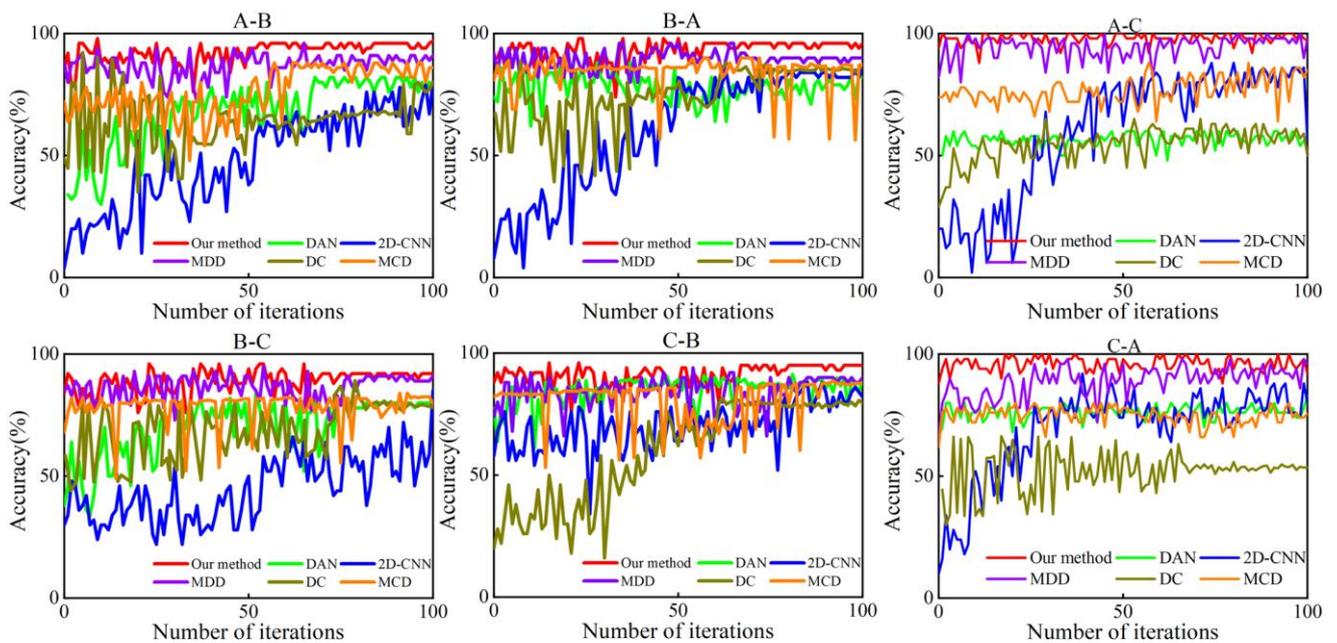


Fig. 26. Real-time diagnostic accuracy curves of different methods.

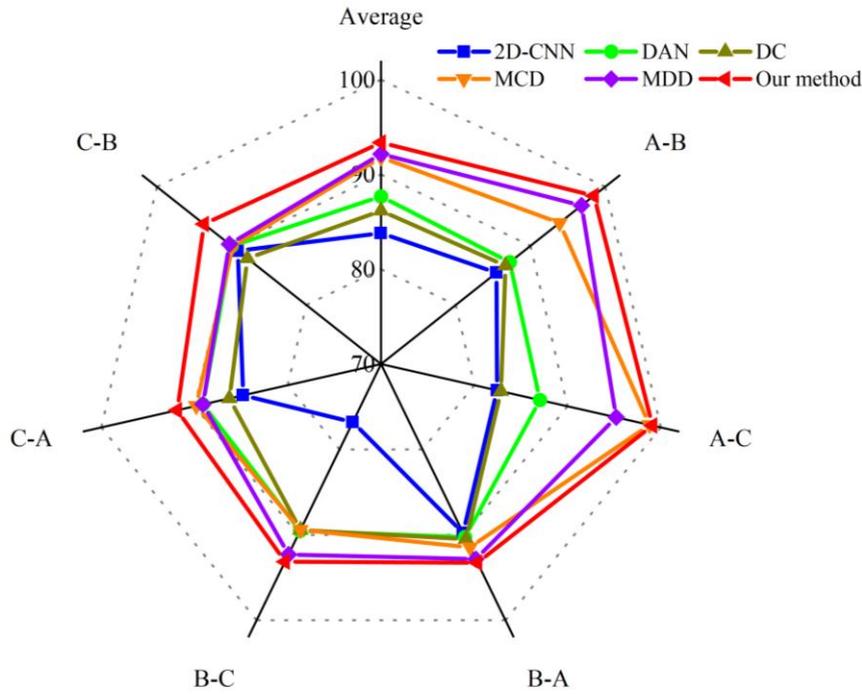


Fig. 27. Diagnostic results of different methods.

4. Conclusions

This study proposes a fault diagnosis framework based on two-stage progressive adversarial training. The two stages implement global domain alignment and local subspace alignment, respectively. In terms of the loss function, different pseudo label probability distributions are utilized to jointly compute the LMMD to reduce the deviation of the feature distribution in the same subspace of different domains, so that more fine-grained feature information can be applied to the diagnosis. Additionally, an adaptive weighting strategy is introduced in the joint computational LMMD for dynamically optimizing the weight allocation of each loss in the network training. The target domain reinforcement loss is designed to improve the learning ability of the network on target domain features. In terms of network architecture, a hierarchical adaptation mechanism is proposed. At the shallow level of the network, the source and target domains have independent learnable parameters, allowing the network to be more flexible in capturing semantic information of their respective domains. At the deeper layers of the network, the network is able to learn shared high-level, abstract feature representations from different domains. Through this method, the risk of negative transfer due to distribution divergence between source and target domains is reduced, contributing to the positive transfer of feature information. Furthermore, the network convergence

is more stable.

The proposed method has been thoroughly validated through blower experiments and fuel injector experiments. In the blower experiments, the average diagnostic accuracy across 12 transfer diagnosis tasks reached 98.71%, outperforming other methods by 1.55% to 11.38%. In the fuel injector experiments, the proposed method achieved an average diagnostic accuracy of 93.37%, which was 9.56%, 5.69%, 7.20%, 2.40%, and 1.91% higher than the 2D-CNN, DAN, DC, MCD, and MDD methods, respectively. This further highlights the superior scalability and adaptability of the proposed method.

For real-time application in ship operations, however, several challenges must be addressed. One of the primary challenges is ensuring that the system can operate efficiently and accurately in dynamic, real-time environments with varying engine loads, operational states, and environmental factors. Real-time processing requires low-latency data handling, which will necessitate further optimizations in terms of computational efficiency. Sensor noise, which is common in maritime machinery, also poses a significant challenge. While the proposed method is robust to some degree of noise, the addition of noise-resilient mechanisms, such as noise filtering or enhanced feature extraction techniques, will further improve the system's performance in real-world scenarios. The integration of multimodal data sources (e.g., vibration, temperature, and

acoustics) presents an exciting opportunity for future work. By fusing these multiple data streams, the diagnostic system could potentially capture a broader range of fault characteristics and improve diagnostic accuracy. Moreover, future studies could investigate the application of self-supervised learning

approaches, which would help reduce the reliance on labeled data and enhance the scalability and flexibility of the system, especially in environments where labeled data are scarce. We will also incorporate real-vessel data to conduct a broader evaluation of the proposed method.

Acknowledgements

This study is sponsored by the National Natural Science Foundation of China (Grant No. 52471311), the China Scholarship Council scholarship program (Grant No. 202506570017), the Fundamental Research Funds for the Central Universities (Grant No. 3132025611).

References

1. Guo Y, Zhang J, Sun B, Wang Y. A universal fault diagnosis framework for marine machinery based on domain adaptation. *Ocean Eng* 2024; 302: 117729, <https://doi.org/10.1016/j.oceaneng.2024.117729>.
2. Liang Q, Knutsen K, Vanem E, Aesoy V, Zhang H. A review of maritime equipment prognostics health management from a classification society perspective. *Ocean Eng* 2024; 301: 117619, <https://doi.org/10.1016/j.oceaneng.2024.117619>.
3. Lloyd's List Intelligence, Ship Accident Data 2023, London, United Kingdom, Nov. 2023. [Lloyd's List Intelligence \(lloydslistintelligence.com\)](https://lloydslistintelligence.com).
4. American Bureau of Shipping, Guidance Notes on Failure Mode and Effects Analysis (FMEA) for Classification, Houston, The USA, Jan. 2018.
5. Lloyd's List, Outlook 2021, Shipping Accelerates towards an Uncertain Future, London, United Kingdom, Jan. 2021.
6. Yang N, Liu J, Ma H, Zhao W, Gao Y. Fault diagnosis of gear based on URP-CVAE-MGAN under imbalanced and small sample conditions. *Eksploatacja i Niezawodność - Maintenance and Reliability* 2025; 27(3): 199417, <https://doi.org/10.17531/ein/199417>.
7. Ge Y, Zhang F, Ren Y. A Federated Transfer Fault Diagnosis Method for Cross-Domain and Incomplete Data. *Eksploatacja i Niezawodność - Maintenance and Reliability* 2025; 27(2): 194182, <https://doi.org/10.17531/ein/194182>.
8. Dong X, Zeng H, Li C, Ye K, Lu Z. Health condition assessment of marine diesel engine sub-system based on PSO-BP neural network. *Ships and Offshore Structures* 2025; 1–17. <https://doi.org/10.1080/17445302.2025.2486222>.
9. Liu Z, Yang X, Xie Y, Wu M, Li Z, Mu W, Liu G. Multisensor cross-domain fault diagnosis method for leakage of ship pipeline valves. *Ocean Eng* 2024; 299: 117211, <https://doi.org/10.1016/j.oceaneng.2024.117211>.
10. Shahid S, Ko S, Kwon S. Real-time abnormality detection and classification in diesel engine operations with convolutional neural network. *Expert Systems with Applications* 2022; 192: 116233, <https://doi.org/10.1016/j.eswa.2021.116233>.
11. Shi J, Ren Y, Tang H, Xiang J. Hydraulic directional valve fault diagnosis using a weighted adaptive fusion of multi-dimensional features of a multi-sensor. *J. Zhejiang Univ. Sci. A* 2022; 23(4): 257–271. <https://doi.org/10.1631/jzus.A2100394>.
12. Coraddu A, Oneto L, Ilardi D, Stoumpos S, Theotokatos G. Marine dual fuel engines monitoring in the wild through weakly supervised data analytics. *Eng. Appl. Artif. Intell* 2021; 100: 104179, <https://doi.org/10.1016/j.engappai.2021.104179>.
13. Qin H, Yang R, Guo C, Wang W. Fault diagnosis of electric rudder system using PSOFOA-BP neural network. *Measurement* 2021; 186: 110058, <https://doi.org/10.1016/j.measurement.2021.110058>.
14. Dibaj A, Etefagh M, Hassannejad R, Ehghaghi M. A hybrid fine-tuned VMD and CNN scheme for untrained compound fault diagnosis of rotating machinery with unequal-severity faults. *Expert Systems with Applications* 2021; 167: 114094, <https://doi.org/10.1016/j.eswa.2020.114094>.
15. Xu X, Lin Y, Ye C. Fault diagnosis of marine machinery via an intelligent data-driven framework. *Ocean Eng* 2023; 289: 116302, <https://doi.org/10.1016/j.oceaneng.2023.116302>.
16. Xu Z, Li C, Yang Y. Fault diagnosis of rolling bearing of wind turbines based on the variational mode decomposition and deep convolutional neural networks. *Appl. Soft. Comput* 2020; 95: 106515, <https://doi.org/10.1016/j.asoc.2020.106515>.
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, <https://doi.org/10.1016/j.ymsp.2020.106962>.

18. Zhu J, Li O, Chen M, Hu B, Ma E. Rolling bearing fault diagnosis method based on adaptive signal diagnosis network and its application. *Eksploatacja i Niezawodność – Maintenance and Reliability* 2025; 27(2): 194673; <https://doi.org/10.17531/ein/194673>.
19. Gao Y, Li X. Fault Diagnosis of Centrifugal fan Bearings Based on I-CNN and JMMD in the Context of Sample Imbalance. *Eksploatacja i Niezawodność – Maintenance and Reliability* 2024; 26(4):191459, <https://doi.org/10.17531/ein/191459>.
20. Zhao W, Lv Y, Lee K, Li W. An intelligent data-driven adaptive health state assessment approach for rolling bearings under single and multiple working conditions. *Computers & Industrial Engineering* 2025; 203: 110988, <https://doi.org/10.1016/j.cie.2025.110988>.
21. Jiang F, Song H, Shen X. A novel fault transfer diagnosis method using CBAM-MobileNetV2 and its application to mine hoist main bearings under variable working conditions. *J Mech Sci Technol.* 2025; 39: 3091–3104, <https://doi.org/10.1007/s12206-025-0509-6>.
22. Zhang X, Yang K. Transformer fault diagnosis method based on MTF and GhostNet. *Measurement*, 2025; 249: 117056, <https://doi.org/10.1016/j.measurement.2025.117056>.
23. Cui Z, Lu Y, Yan X, Cui S. Compound fault diagnosis of diesel engines by combining generative adversarial networks and transfer learning. *Expert Systems with Applications* 2024; 251: 123969, <https://doi.org/10.1016/j.eswa.2024.123969>.
24. Li C, Li G, Xing P, Cui D, Sui Y, Zhang H. A progressive domain separation network incorporating iris time-frequency maps for open-set diagnosis of ship propulsion shafting. *Ocean Eng* 2026; 343: 123219, <https://doi.org/10.1016/j.oceaneng.2025.123219>.
25. Tang A, Wang M, Pan H. Angle-based transfer support matrix machine for roller bearing fault diagnosis under limited labeled data. *Structural Health Monitoring* 2024, [doi:10.1177/14759217241298394](https://doi.org/10.1177/14759217241298394)
26. Zhang Y, Li C, Tang Y, Zhang X, Zhou F. A novel fault early warning method for centrifugal blowers based on stacked denoising autoencoder and transfer learning. *Journal of Manufacturing Systems* 2024; 76: 443-456, <https://doi.org/10.1016/j.jmsy.2024.08.013>.
27. Li C, Li G, Xing P, Cui D, Sui Y, Zhang H. A semi-supervised learning framework fusing multi-sensor visual features for ship propulsion shafting fault diagnosis. *Structural Health Monitoring* 2025. <https://doi.org/10.1177/14759217251359898>.
28. Huang M, Yin J, Yan S, Xue P. A fault diagnosis method of bearings based on deep transfer learning. *Simulation Modelling Practice and Theory* 2024; 122: 102659, <https://doi.org/10.1016/j.simpat.2022.102659>.
29. Hu J, Yu Y, Yang J, Jia H. Research on the generalisation method of diesel engine exhaust valve leakage fault diagnosis based on acoustic emission. *Measurement*. 2023; 210: 112560, <https://doi.org/10.1016/j.measurement.2023.112560>.
30. Hu L, Liu L, Yang J, Hu H, Zheng C, Yu Y. Fault diagnosis based on deep transfer learning for marine turbocharger. *International Journal of Mechanical Sciences* 2025; 300: 110444, <https://doi.org/10.1016/j.ijmecsci.2025.110444>.
31. Pei G, Zhu X, Wang X. A turn-based domain adversarial network for cross-domain fault diagnosis of diesel engine. *J Mech Sci Technol.* 2025; 39: 3079–3090, <https://doi.org/10.1007/s12206-025-0508-7>.
32. Li J, Deng W, Dang X, Zhao H. Cross-domain Adaptation fault diagnosis with Maximum classifier discrepancy and deep feature alignment under variable working conditions. *IEEE Transactions on Reliability*, <https://doi.org/10.1109/TR.2025.3551155>.
33. Saito K, Watanabe K, Ushiku Y. Maximum classifier discrepancy for unsupervised domain adaptation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2018; 201(4): 3723-3732, <https://doi.org/10.1109/CVPR.2018.00392>.
34. Ghorvei M, Kavianpour M, Beheshti M, Ramezani A. Spatial graph convolutional neural network via structured subdomain adaptation and domain adversarial learning for bearing fault diagnosis. *Neurocomputing* 2023; 517: 44-61, <https://doi.org/10.1016/j.neucom.2022.10.057>.
35. Fu S, Zou L, Wang Y, Lin L, Lu Y, Zhao M, Guo F, Zhong S. DCSIAN: A novel deep cross-scale interactive attention network for fault diagnosis of aviation hydraulic pumps and generalizable applications. *Reliability Engineering & System Safety* 2024; 249: 110246, <https://doi.org/10.1016/j.ress.2024.110246>.
36. Gao C, Wang Z, Guo Y, Wang H, Yi H. MPINet: Multiscale physics-informed network for bearing fault diagnosis with small samples. *IEEE Transactions on Industrial Informatics.* 2024; 20(12): 14371-14380, <https://doi.org/10.1109/TII.2024.3452174>.
37. 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, <https://doi.org/10.1016/j.measurement.2021.109345>.
38. Jiang Q, Lin X, Lu X, Shen Y, Zhu Q, Zhang Q. Self-supervised learning-based dual-classifier domain adaptation model for rolling bearings cross-domain fault diagnosis. *Knowledge-Based Systems* 2024; 284: 111229, <https://doi.org/10.1016/j.knosys.2023.111229>.

Appendix:

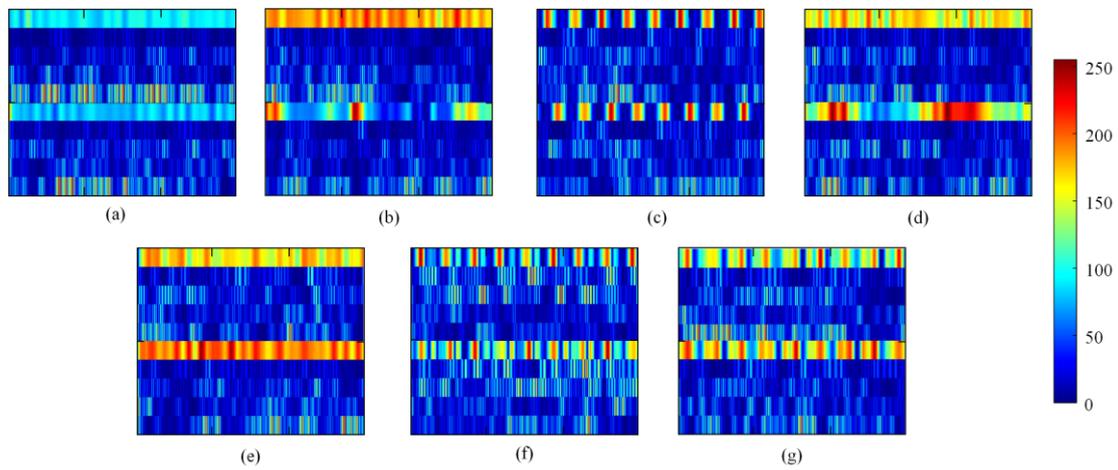


Figure A1: Time-frequency images of vibration signals under Domain A condition: (a)~(g): label_1~label_7.

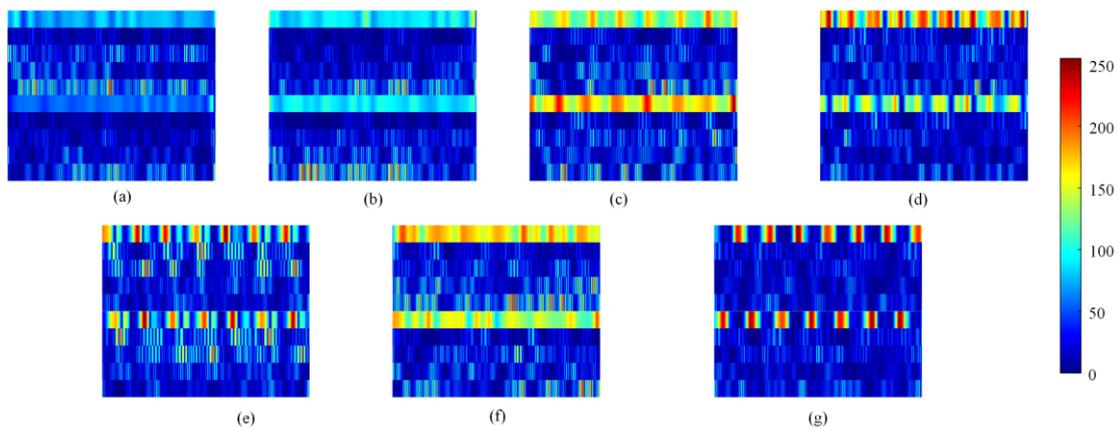


Figure A2: Time-frequency images of vibration signals under Domain C condition: (a)~(g): label_1~label_7.

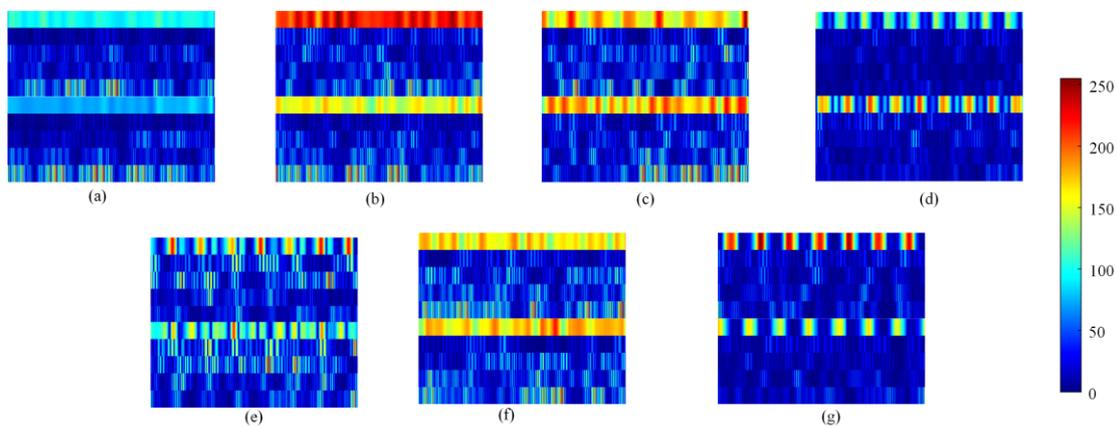


Figure A3: Time-frequency images of vibration signals under Domain D condition: (a)~(g): label_1~label_7.

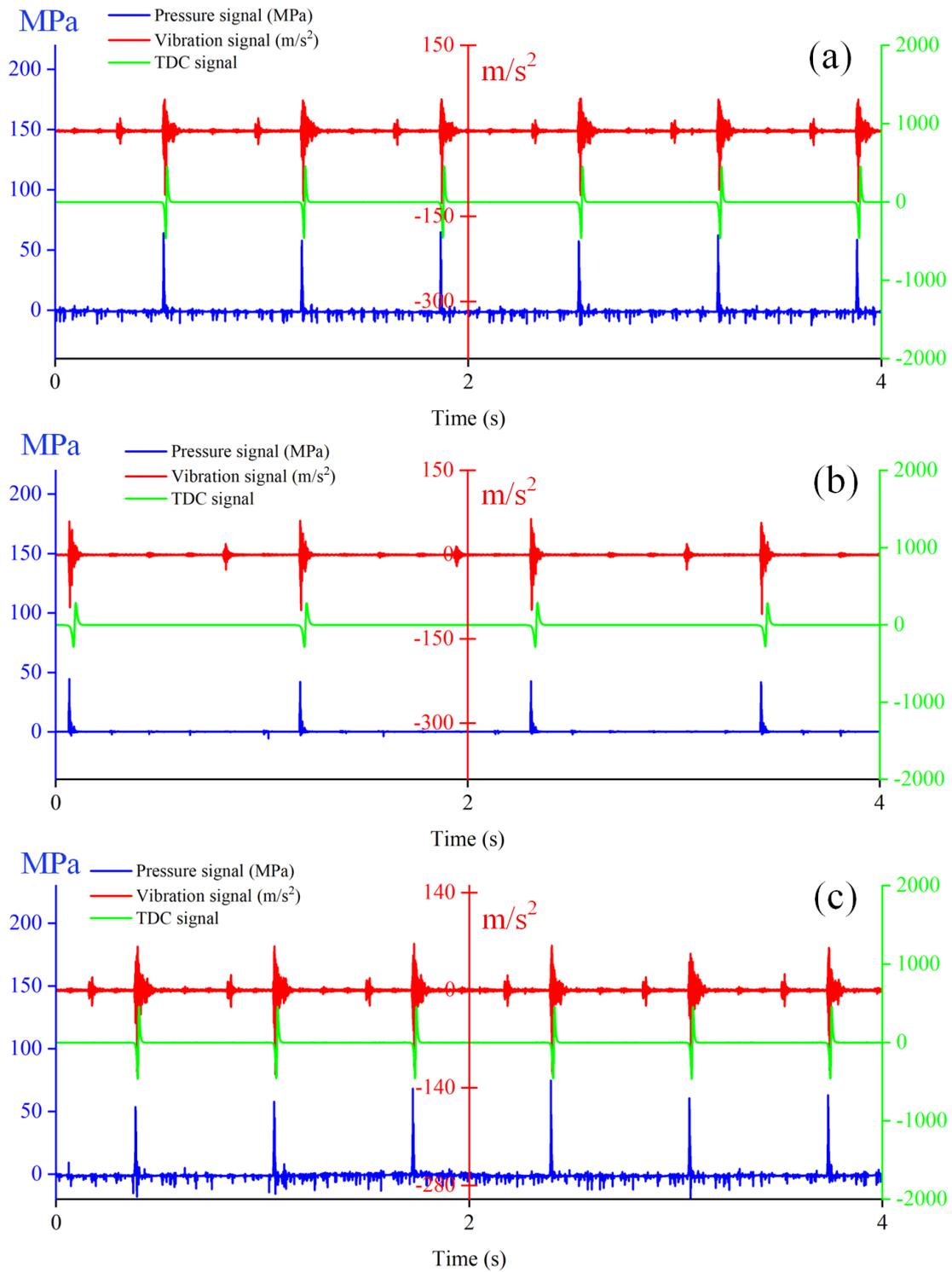


Figure A4: Signals of different health states (Dataset A): (a) normal state; (b) low valve opening pressure; (c) nozzle blockage.

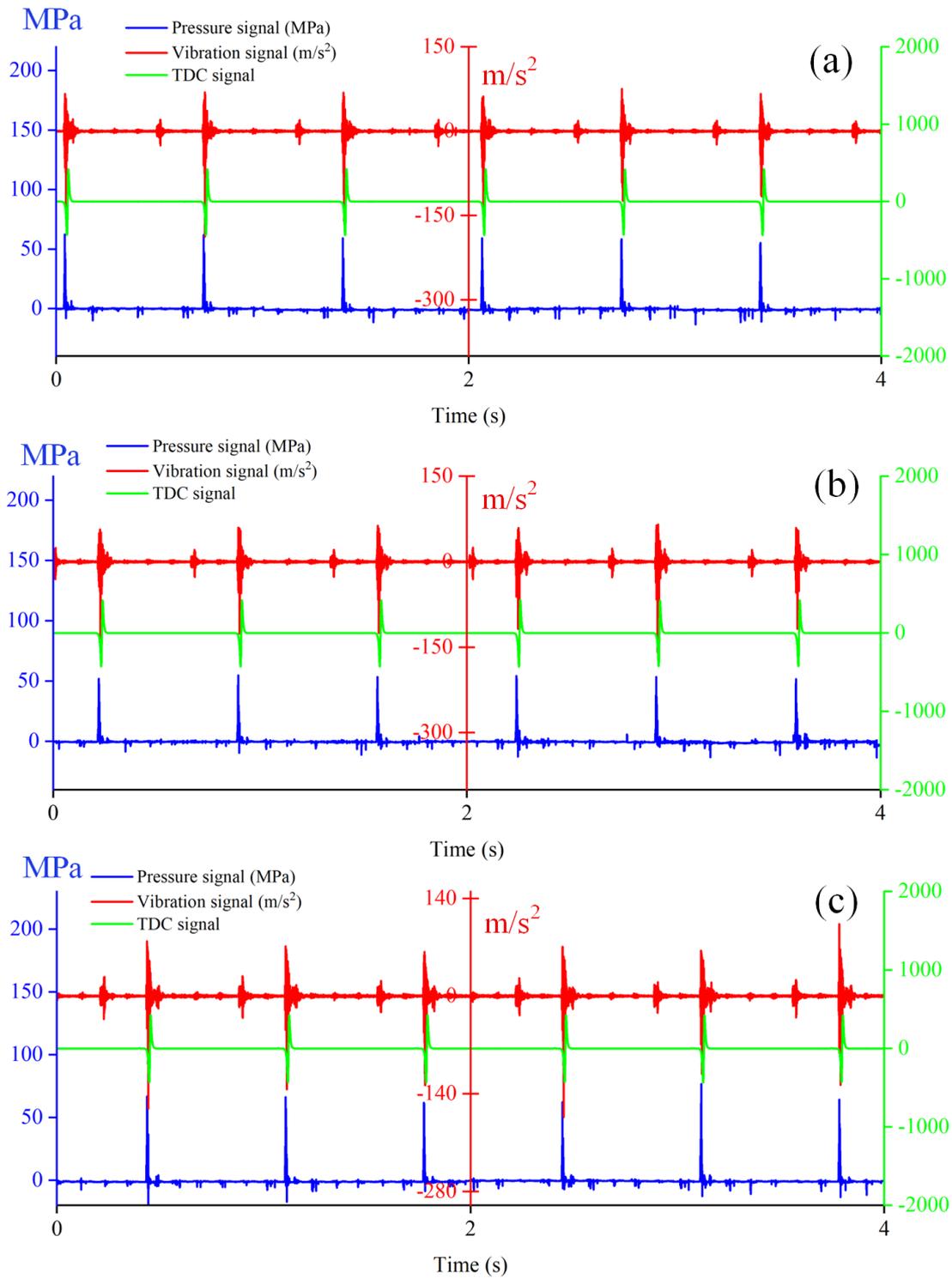


Figure A5: Signals of different health states (Dataset B): (a) normal state; (b) low valve opening pressure; (c) nozzle blockage.

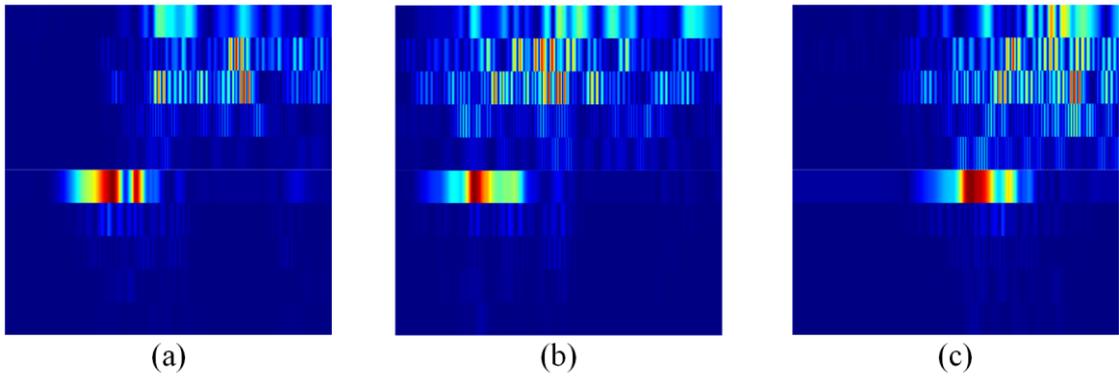


Figure A6: Time-frequency information graph (Dataset B): (a) normal state; (b) low valve opening pressure; (c) clogged nozzle.

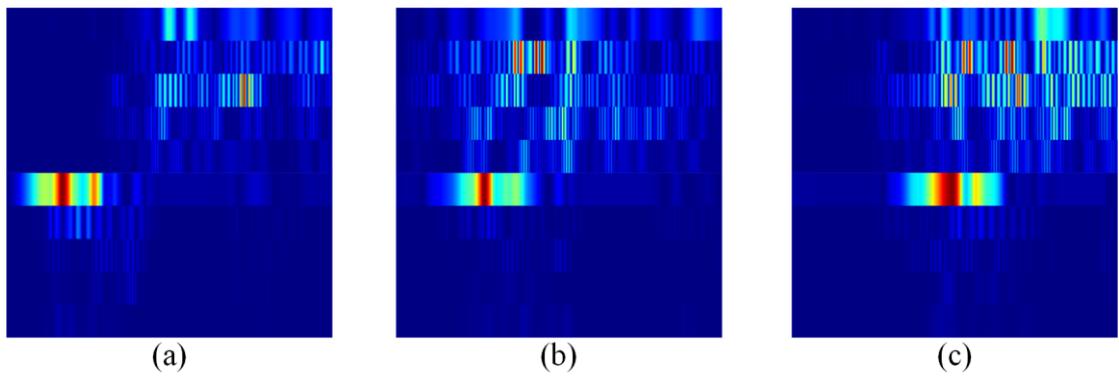


Figure A7: Time-frequency information graph (Dataset C): (a) normal state; (b) low valve opening pressure; (c) clogged nozzle.