Convolutional Neural Network Based Two-Layer Transfer Learning for Bearing Fault Diagnosis

WANPENG ZHANG, PENG ZHANG, XIAOHAN HE, AND DAILIN ZHANG
School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
Corresponding author: Dailin Zhang (mnizhang@hust.edu.cn)

ABSTRACT
Rolling bearing fault diagnosis is one of crucial tasks in mechanical equipment fault diagnosis. Currently, artificial intelligence and machine learning-driven fault diagnosis methods are extensively utilized for rolling bearing. When compared to traditional techniques, the diagnostic accuracy has significantly improved. These methods, however, need a substantial amount of labelled training data, which is difficult to obtain in actual failures. In order to resolve this problem, Transfer Learning (TL) was created to learn in the target domain by accessing knowledge from the pertinent labelled source domain. Inspired by Maximum Mean Discrepancy, this paper puts forward a Convolutional Neural Network (CNN) based Two-layer Transfer Learning (CTTL) method for fault diagnosis. In the first layer, the fault features are automatically extracted by CNN and a term called Feature Weighted Maximum Mean Discrepancy (WMMD) is considered to minimise the difference between source and target domains. In the second layer, the Third Dataset, which is based on the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) method was designed on the basis of the predicted labels from the first layer. The Calinski-Harabaz (CH) index of the Target Dataset controls the iteration times of CTTL. CTTL change the process of Transfer Learning method from learning the distribution of domains to learning the distribution of fault types in more detail, which will get higher accuracy. Proposed CTTL is tested by the bearing datasets of Case Western Reserve University (CWRU) and XJTU-SY. The experimental findings reveal that CTTL is capable of achieving a high diagnosis accuracy across different load domains. In the majority of experiments, CTTL outperformed other algorithms, including Deep Neural Network, Support Vector Machine, and several other methods.

INDEX TERMS
Rolling bearing, fault diagnosis, transfer learning, DBSCAN, CH index.

I. INTRODUCTION
As a result of the rapid development of machine learning technology, data-driven fault diagnosis methods are widely employed. At the same time, deep learning has the ability to automatically extract features, and its performance and efficiency are often better in comparison to traditional methods. Among the completed works, Deep Belief Networks (DBN) [1], Sparse Autoencoder (SAE) [2], Support Vector Machine (SVM) [3], in particular convolutional neural network (CNN) [4], [5], have demonstrated good performance in fault diagnosis. Liu et al. [6] used a 1-D convolutional neural network for the detection and diagnosis of a sensor. Wavelet neural network has been employed by. Pham et al. [7] proposed a rolling bearing fault diagnosis based on improved GAN and two-dimensional representation of acoustic emission signals. Wu et al. [8] for bearing fault diagnosis. Li [9] proposed an attention mechanism (AM) algorithm weighted long short-term memory (LSTM) neural network for the purpose of getting better fault diagnosis results in Tennessee Eastman process. He et al. [10] suggested a fault diagnosis method based on wavelet packet transform and convolutional neural network. Nauyen et al. [11] proposed a new DNN-based vibration signal-based bearing fault diagnosis method. A multi-channel deep convolutional neural network (MC-DCNN) fault diagnosis model based on multi-source was proposed by Gong et al. [12]. Despite the fact that these diagnostic methods have achieved good results, there are
some problems with these deep learning data-driven methods that prevent their application in real fault case. Reason 1: A large amount of labelled data is needed as training samples. It is obviously very difficult or uneconomical for a defined fault diagnosis task to obtain such a substantial amount of labelled data. Reason 2: While obtaining a number of labelled samples as a training set, in order to ensure the performance of the algorithm, we also must make sure that these training data have the same distribution as well as and external conditions as the data to be detected. For this problem, transfer learning is a viable solution. It learns information from the source domain and applies them to the target domain, while performing the transfer task on two similar datasets.

Domain adaptation [13], [14], [15], [16] is a popular paradigm for transfer learning. A number of scholars have carried out research on fault diagnosis. Zhang et al. [17] proposed an intelligent machine fault diagnosis method using convolutional neural networks and transfer learning. A transfer learning approach based on deep neural networks was presented by Zhang et al. [18], which can fine-tune the network classification layer through a small amount of labelled target data for rolling bearing fault diagnosis. Wang et al. [19] proposed an adversarial network coupled with an attention mechanism for fault transfer tasks of bearings. Li et al. [20] suggested the transfer learning method based on weighted adversarial networks. Zang et al. [21] proposed an Extreme Learning Machine with output weight alignment for knowledge transfer. Cheng et al. [22] make use of the 1-D CNN as feature extractor, and Wasserstein distance as domain loss for bearing fault diagnosis. A domain adaption method for bearing fault diagnosis was proposed by Wen et al. [23]. They extracted the features using a three-layer sparse autoencoder network and built the transfer model using MMD as the domain loss, and achieved good results. However, they did not take into account the different contributions of multidimensional data, in this regard, Wang et al. [24] proposed linear discriminant analysis (LDA) weighted MMD loss as domain loss, which obtained good results than MMD. Ma et al. [25] proposed a double weighted domain adaptation method that further improved the MMD adaptation function. Although the research of these scholars has greatly improved the performance of bearing fault diagnosis, they ignored the further improvement that the high-performance results of the diagnosis can bring to the algorithm. Since the target data is unlabeled, and the disordered data of two domain can only learn the distribution of domains in traditional methods while the distribution of individual fault classes in the two domain data are ignored. This situation is a great waste for the predicted high accuracy labels. This also inspired further work in this paper, and then CTTL for Bearing Fault Diagnosis was proposed.

The first layer of CTTL is the transfer layer. The goal of this layer is consistent with that of Wen and Liu et al., which are to obtain good diagnostic results by improving and adapting transfer learning. In this layer, we adopt the LDA-weighted mmd as the domain adaptation function, and output the final fault diagnosis result by training two networks with shared parameters. The second layer of CTTL is the iteration layer. We treat diagnostic labels as maximum likelihood estimates for every samples. The samples with the highest confidence under each diagnostic label are selected and trained in the same batch with source domain data under the same label. This process enables the transfer network to minimize the distribution difference under the same fault class rather than the two whole domains. In this process, different to the source domain and target domain, these samples with high confidence are made into a third dataset by the binary DBSCAN. To control the number of iterations, we compute the Calinski-Harabasz (CH) index of the Target Domain data (ratio of between-class divergence to within-class divergence), and stop the iteration when CH index is not rising. The following is the contribution of this paper.

1) The WMMD domain adaptation function is considered. In this paper, the first layer of CTTL adopts the convolution layer to extract fault features, and uses the WMMD domain adaptation function to perform migration learning fault diagnosis on bearing fault data under different working conditions, and the effect exceeds the traditional deep learning and migration learning diagnosis methods.

2) The information of diagnostic labels is reused. To better reuse the predicted labels, the false positive samples (FPs) in each predicted class need to be removed. In this paper, we propose a DBSCAN-based method for the generation of the Third Dataset. This method can effectively avoid the interference of outlier samples and obtain a cleaner Third Dataset. This is a better representation of the distributional expectation of the target domain data under that label.

3) The adaptive threshold of the number of iterations is set. We consider the possible negative contribution caused by the introduction of the Third Dataset, which may lead to a decrease of the final diagnosis accuracy. We use the CH index of the Target Domain data as the post-controller for the number of CTTL iterations. When there is no increase for CH index, the iterative process will be stopped and the prediction result of the previous round is taken as the final fault diagnosis result. This ensures that CTTL can obtain the highest accuracy of diagnostic results regardless of whether the predicted label is accurate or not. The rest of this manuscript is organized as follows. We describe the Materials and Methods of this manuscript in Section II. We describe the dataset used for the experiments in this manuscript in Section III. We describe the results of the proposed method in this manuscript compared to other methods in Section IV. The discussion is given in Section V. The conclusion and future research are given in Section VI.

II. PROPOSED METHODOLOGY

The framework of whole CTTL is shown in Figure 1. Bearing failure data under different loads are used as labeled source
domain and unlabeled target domain data. First, the frequency domain features are extracted from the vibration signal in the time domain using the FFT, and then the Source and Target domains are input into the first layer of the CTTL in a random order. In this layer, the modified domain adaptation function WMMD will minimize the differences between the two domain features and output the labels of the Target Domain data.

In the end, the second layer of CTTL is used to obtain the final fault diagnosis results. Specifically, the most clustered samples in the estimated class are obtained by the DBSCAN method to produce the Third Dataset, which is trained with the ordered source domain data. The results are optimized by multiple iterations to improve the accuracy, and the number of algorithm iterations is controlled by CH index.

A. THE FIRST LAYER OF THE PROPOSED CTTL

The first layer of CTTL is shown as Figure 2. where the source and target domains share parameters in the feature extraction network and the fully connected layer. Since the target domain data $x^t$ are unlabelled, the data inputs $(x^s, x^t)$ should be unordered. The datasets of these two domains are fed into two network models, and the fault features are retained to the maximum extent by convolutional feature extractor. Then the input is fed into the fully connected (FC) layer, and the difference between the two-domain distributions can be accurately, minimized by the feature weighted WMMD domain loss which is using the features $(\xi_s, \xi_t)$ outputted before the classification layer. Combining the WMMD loss ($l_{dom}$) of the two domain features and the cross-entropy classification loss ($l_{cla}$) of the source domain data, a network with domain adaptation capability and strong classification ability can be trained with high prediction precision, as (1).

$$l_{total} = l_{cla} + \lambda l_{dom}$$

Table 1 shows the parameters of the proposed CTTL network, where the Size term of the Conv layer is the input dimension * output dimension * kernel size * stride in padding mode. FC1 is the fully connected layer with 512 nodes, 2048 dimensions of input and 512 dimensions of output. FC2 is the fully connected layer with 128 nodes, 512 dimensions of input and 128 dimensions of output.

1) DOMAIN LOSS BASED ON FEATURE WEIGHTING MAXIMUM MEAN DISCREPANCY

The purpose of domain adaptation is to adapt the model, which was learned from the source domain to the target domain. The biggest challenge of domain adaptation is the lack of target domain labels, which increases the difficulty of knowing target domain distribution. For this problem, we choose to minimize the disparity of the domain distribution to achieve domain adaptation. The Maximum Mean Discrepancy (MMD) [26], has achieved good results in domain adaptation. Assuming that the source domain data is $x^s_i$ and the target domain data is $x^t_i$, they are mapped to the same space by kernel function $\phi$. The expression is as:

$$MMD(X^s, X^t) = \left\| \frac{1}{N_s} \sum_{i=1}^{N_s} \phi(x^s_i) - \frac{1}{N_t} \sum_{i=1}^{N_t} \phi(x^t_i) \right\|_F$$

The DDC algorithm [27] is proposed by combining MMD with neural network. In order to embed the MMD into the network framework and avoid the heavy computation of kernel
function, the neural network was used to replace the kernel function, and the MMD is rewritten as:

\[ l_{\text{MMD}} = \frac{1}{N_S} \sum_{i=1}^{N_S} \xi_i^s - \frac{1}{N_T} \sum_{i=1}^{N_T} \xi_i^t \]  \quad (3)

where \( \xi_i^s \) represents the characteristic data output through network, \( \xi_i^t \) represents the characteristic data of target domain data output through network, \( N_S \) represents the quantity of source domain data and \( N_T \) represents the quantity of target domain data. For multi-dimensional data, different dimensions have different contributions to the transfer tasks. Therefore, it is not appropriate to assign the same weight to each characteristic variable of the data when calculating the transfer domain loss. Instead, what we need is to find a new projection, which can maximize the covariance of the source domain and the target domain, thus the category information can be fully preserved. It should also be able to minimize the distribution difference of the source domain and the target domain. In the meantime, in order to get the feature weight in the network structure, we need to find a linear projection to minimize the covariance of the two domains and maximize the center distance of the two domains. Thus, the suggested projection overall objective function can be written as:

\[
max J = \frac{\| w^T \mu_0 - w^T \mu_1 \|^2}{w^T \Sigma_0 w + w^T \Sigma_1 w} \\
= \frac{\| w^T (\mu_0 - \mu_1) (\mu_0 - \mu_1)^T w \|}{w^T (\Sigma_0 + \Sigma_1) w} 
\]  \quad (4)

where \( \mu_d, \Sigma_d \) is the mean vector and covariance matrix of samples from the d-Domain dataset (d = Source or Target). If all the samples are projected onto a Hyperplane \( w \), then the centers of the classes can be written as \( w^T \mu_0 \) and \( w^T \mu_1 \) respectively. The covariance matrices of the two classes are \( w^T \Sigma_0 w \) and \( w^T \Sigma_1 w \).

\( S_w \) is defined as the within-class scatter matrix

\[
S_w = \Sigma_0 + \Sigma_1 \\
= \sum_{x \in X_0} (x - \mu_0)(x - \mu_0)^T + \sum_{x \in X_1} (x - \mu_1)(x - \mu_1)^T 
\]  \quad (5)

where \( X_d \) is the input matrix. \( S_b \) is defined as the between-class scatter matrix

\[
S_b = (\mu_0 - \mu_1)(\mu_0 - \mu_1)^T 
\]  \quad (6)

Equation (4) is expressed as the maximum generalized Rayleigh quotient as:

\[
J = \frac{w^T S_b w}{w^T S_w w} 
\]  \quad (7)

Let \( w^T S_w w = 1 \), (7) can be changed as:

\[
min -w^T S_b w \\
s.t. w^T S_w w = 1 
\]  \quad (8)

By Lagrange multiplier method, (8) is changed as:

\[
S_{bw} = \lambda S_w w 
\]  \quad (9)

Let \( S_{bw} = (\mu_0 - \mu_1) \), (9) is changed as:

\[
S_{bw} = \lambda (\mu_0 - \mu_1) 
\]  \quad (10)
The projection matrix $w$ can be expressed as:

$$w = S_w^{-1} (\mu_0 - \mu_1)$$

(11)

To ensure the stability of the solution, the Singular Value Decomposition (SVD) is used. This allows to find the pseudo-inverse avoiding that the lack of inverse matrix makes the calculation wrong. We have done the Singular Value Decomposition on $S_w$:

$$S_w = U \Sigma V^T$$

(12)

Then, the $S_w^{-1}$ in 11 can be calculated by

$$S_w^{-1} = V \Sigma^{-1} U^T$$

(13)

The feature weights can be expressed as:

$$z_i = |w_i| / \sum_{i=1}^{m} |w_i|$$

(14)

$$Z = \text{diag}([z_1, z_2, \ldots, z_m])$$

(15)

$|w_i|$ is the absolute value of the $i$-th element of $w$. Therefore, $|w_i|$ reflects contribution of the $i$-th dimensional feature, and diagonal matrix $Z$ can be combined with their weights $z_i$. WMMD can be expressed as:

$$WMMD\left(X^S, X^T\right) = \left\| \frac{1}{N_S} \sum_{i=1}^{N_S} \xi_i^S Z - \frac{1}{N_T} \sum_{i=1}^{N_T} \xi_i^T Z \right\|_F$$

(16)

After the domain adaptation process of WMMD as domain loss, we can get a higher precision fault diagnosis label for the second layer of CTTL.

B. THE SECOND LAYER OF THE PROPOSED CTTL

The second layer of CTTL as the whole is a circular iterative process, which is composed of the following four modules: namely Testing Network, Training Networks, Label Reuse Module and Adaptive Iterative Controller. Testing Networks outputs prediction labels, where the network parameters are shared from Training Networks. There are two networks in the Training Network, corresponding to the Source dataset and the third dataset, and the parameters of the two networks are shared. Adaptive Iterative Controller adaptively controls the number of iterations by CH index. Label Reuse Module uses the DBSCAN method for the third dataset. In the first layer of CTTL, the data of two domains are randomly ordered, and only the distribution of two domains can be learned. In contrast, the second layer can get the distribution under specific fault types due to the higher accuracy of the prediction labels, and change from learning the distribution of domains to learning the distribution of fault types in more detail, which will get higher accuracy. The structure is shown in Figure 3.

1) DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE

The Third Dataset is the training data of the second layer CTTL, and has an important role in the subsequent iterations process. It will definitely lead to a decrease in diagnostic accuracy when the labels that are used to generate the Third Dataset contain a lot of FPs. In order to obtain the cleaner Third Dataset, we used the DBSCAN algorithm in this paper to take the samples with the highest confidence under each fault class of the prediction labels. The traditional
distance-based method first calculates the mean of all samples, and then removes outliers based on the distance of each sample from the sample mean. However, this method introduces the error of FP samples. The center sample in DBSCAN is a real sample selected by constant attempts, avoiding the interference of FP samples. Table 2 shows the percentage of TP samples in the Third Dataset obtained by DBSCAN and averaging. There were eight groups of experiments, with the number of TPs in each group being 60% of the total. The TP and FP columns in the table are given in {label: numbers} pairs. The experiments show that the percentage of TP samples in the Third Dataset obtained by DBSCAN is 20.66% higher than that of the averaging method, allowing for a cleaner Third Dataset to be generated for the second layer of CTTL.

DBSCAN [28] is a density-based spatial clustering algorithm. As Figure 4 shown, the algorithm is able to divide regions with sufficient density into clusters.

In order to obtain a reliable third dataset, the misdetection samples of each class should be removed from its central data after the prediction labels have been generated. As shown in Figure 5, if a class exists for the predicted labels, the samples corresponding to this class need to be subjected to the DBSCAN binary classification (TPs, FPs) method to remove outliers with large deviations. Otherwise, if a class does not exist for the predicted label, the samples for this fault class in the Third Dataset are replaced with the mean of the target domain samples. If the number of samples processed by DBSCAN is less than the number of samples in the training dataset, the number of samples is expanded to 800 by selecting the samples that are closer to the sample center according to the distance of each sample from the sample center. If this number is greater than 800, the samples that are further away from the sample center are eliminated. The size of the entire Third Dataset is $8000^*1024$, which is equivalent to the source domain dataset.

There are two most important parameters in the algorithm, one is the distance (Eps) from the core points to the farthest border points within the same cluster, and the other is the minimum number (MinPts) of points within the same cluster.

Under the predicted labels of the first layer CTTL, 60% of the total number of each class is taken as MinPts. To avoid the interference of multi-class labels, the Eps is taken as a smaller value (0.05) and gradually increased until the number of cluster points reaches the MinPts. The implementation process of the Third Dataset production is shown in Algorithm 1.

**Algorithm 1** Approach for Creating a Third Dataset for Iteration $k$

Input: Source dataset; Target dataset; Prediction labels for target dataset.

**STEP1**: The target domain data is divided according to fault classes according to the prediction labels.

**STEP2**: Determine if the fault class $N$ exists. IF false: go to **STEP3**; ELSE go to next fault class.

**STEP3**: RETURN the mean value of the Source Domain samples corresponding to the fault class $N$. Then, go to next iteration.

**STEP4**: Set the parameter MinPts to 60% of the total number of this fault class and Eps to the initial value of 0.05.

**STEP5**: Calculate DBSCAN to determine whether clusters exist. IF true: RETURN clusters samples, then, go to next iteration. ELSE: Eps += 0.05, go to **STEP5**.

**STEP6**: Calculate Calinski-Harabaz index $CH_k$. IF $CH_k > CH_{k-1}$ RETURN Third Dataset, sequential fault labels for next second layer of CTTL. ELSE End the CTTL algorithm. RETURN prediction labels of last time for target dataset as fault diagnosis result.

2) **CALINSKI-HARABAZ INDEX**

The Calinski-Harabasz index [29] is obtained by calculating the ratio of between-class scatter matrix and within-class
scatter matrix in a multi-classification problem. A higher CH index means the tighter sample within a class and a greater separation of samples between classes, that is, a better classification result. The CH index is an internal evaluation index that does not need data labels. When the CH index of the generated Third Dataset is incremented, it can be considered that the current iteration of prediction is better than the previous round. CTTL iteration is performed. Otherwise, the training is stopped and the prediction labels of the previous round are used as the diagnosis results. The equation for CH is shown in (17).

\[
CH = \frac{\text{tr} (S_b) / k - 1}{\text{tr} (S_W) / N - k}
\]  

(17)

where \(S_b\) is the between-class scatter matrix; \(S_W\) is the within-class scatter matrix; \(\text{tr}()\) is the trace of the matrix. \(k\) represents the number of fault classes, which is 10 in this paper. \(N\) is the total number of samples. The within-class scatter matrix is expanded on (5) as (18).

\[
S_w = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q) (x - c_q)^T
\]  

(18)

where \(x\) is the sample in the fault class \(q\), \(c_q\) is the mean of the samples in that class.

Equation (6) is the equation of \(S_b\) for the two-classification problem, expanding to the multi-classification problem as shown in (19).

\[
S_b = \sum_{q=1}^{k} n_q (c_q - c_E) (c_q - c_E)^T
\]  

(19)

where \(n_q\) is the number of samples under the one fault class; \(c_E\) is the mean of the overall samples.

### III. DATA DESCRIPTION

#### A. CWRU BEARING DATASET

In this paper, we use the CWRU [30] bearing dataset, experimental conditions shown in Figure 6. The CWRU bearing dataset provides vibration acceleration data for multiple operating conditions, loads, fault classes and fault points. To better represent the performance of the migration learning algorithm, this paper uses drive-side, 48KHz sampling frequency fault data. Four different loads from 0 HP-4 HP are used as the fault migration domain data. Load in horsepower (HP), 1 imperial horsepower (HP) = 0.7456999 kilowatts (kW).

In the dataset, normal condition (Normal) and three types of faults including outer fault (OF), inner fault (IF) and roller fault (B). Each fault type has three fault levels of 0.007, 0.014, and 0.021 inches. In the case of rolling bearings, the deterioration in the degree of fault is a full life cycle process. Functional breakdown usually occurs at severe levels of fault [31]. Fault severity identification is helpful for determining the urgency of repairs. An effective fault diagnosis method would include both fault pattern and fault severity. In addition, data are collected at different motor loads (0, 1, 2, and 3). ‘a-b’ indicates the transfer from source domain load of ‘a’ to target domain load of ‘b’.

As shown in Figure 7 (a), the sample length in the raw fault signal is 2048. In this paper, the sliding window operation is used for data enhancement considering the relationship between the amount of available data as well as the sample data size. The sampling window has a sliding length of 360 points (condition 0HP IR 0.014 is 70) per time. After obtaining the time-domain sample, the fast Fourier transform (FFT) is used to get a frequency-domain signal with a size of \((1^*1024)\) as Figure 7 (b). As shown in Table 3, we obtain 800 samples under each fault class, for a total size is \((800^*1024)\) of each dataset. The source or target domain is obtained under different loads (0HP, 1HP, 2HP, 3HP).

### TABLE 2. Table of comparison between two methods to remove outlier samples.

| Group | Dataset | TP (Label: Num) | FP (Label: Num) | DBSCAN (%) | Mean method (%) |
|-------|---------|-----------------|----------------|------------|----------------|
| G1    | 0HP     | 6:480 1:320     | 5:36 6:39 7:25 9:20 | 98.33      | 82.71          |
| G2    | 0HP     | 8:480 0:39 1:39 2:32 3:46 4:44 |               | 96.47      | 71.25          |
| G3    | 1HP     | 2:480 6:320     |               | 99.38      | 73.33          |
| G4    | 1HP     | 6:480 1:77 2:89 3:84 4:70 |               | 99.38      | 77.08          |
| G5    | 2HP     | 3:480 9:320     |               | 98.75      | 75.83          |
| G6    | 2HP     | 5:480 0:50 1:47 2:50 3:53 7:58; 8:61 |         | 99.17      | 81.04          |
| G7    | 3HP     | 1:480 7:320     |               | 97.30      | 78.54          |
| G8    | 3HP     | 7:480 0:84 1:76 2:78 3:82 |         | 96.47      | 80.21          |
| Average | /       | /               |               | 98.16      | 77.50          |
TABLE 3. Operating conditions for CWRU dataset.

| Fault type | Motor speed (rpm) | training label set/ test set number |
|------------|------------------|-----------------------------------|
|            | 0HP | 1HP | 2HP | 3HP |                   |
| B          | 0.007 | 0 | 800 |
|           | 0.014 | 1 | 800 |
|           | 0.021 | 2 | 800 |
|           | 0.007 | 3 | 800 |
| IR         | 0.014 | 4 | 800 |
|           | 0.021 | 5 | 800 |
|           | 0.007 | 6 | 800 |
| OR (60' clock) | 0.014 | 7 | 800 |
|           | 0.021 | 8 | 800 |
| N          | / | 9 | 800 |

B. XJTU-SY BEARING DATASET

Although the CWRU dataset is capable of providing experimental data for multiple loads and fault classes, it was obtained in an ideal environment and may not demonstrate its limitations in some cases. Therefore, in Section IV, we consider the full life-cycle XJTU-SY [32] dataset. The conditions to collect the XJTU-SY dataset are shown in Figure 8.

A total of 3 different operating conditions were set, and 5 bearings were tested under each operating condition. The operating conditions are shown in Table 4. The sampling frequency is set to 25.6 kHz. A total of 32768 data points are recorded for each sampling, and the sampling period is equal to 1 min. The third column in Table 4, “Number of file”, represents how many files were collected using the current bearing, i.e., how many groups of data were collected, each of which contains 32768 data points. There are three fault types N (Normal), OR (Out Race), and IR (Inner Race) in conditions 2 and 3.

Figure 9 shows the full life cycle of the bearing from the start of the experiment to damage, the horizontal coordinates are the acquisition file number (minutes). The red box is the sample selection range, as shown by the Sample selection range keyword in Table 4. In order to obtain more stable fault data, the sampling window has a sliding length of 1024 points per time, each data file can contribute 30 samples. The vibration data was selected for 10 consecutive minutes and 300 samples were obtained for each type of fault (OR, IR), combined with the normal samples for a total of 900 samples, which were formed into a dataset after FFT.

We use operating condition 2 and condition 3 as transfer task.

IV. RESULT

In this section, the performance of the proposed CTTL model is tested for transfer learning under different working conditions on the CWRU bearing dataset and XJTU-SY bearing datasets.

1) COMPARISON BETWEEN FIRST LAYER OF CTTL AND DDC

In this section, first layer of CTTL is compared with DDC. All models in this paper are built on the Pytorch platform, and all experiments are implemented on a computing platform.
configured with an NVIDIA GTX 1660Ti GPU and 16GB of RAM.

DDC used the same network framework as CTTL and used MMD to minimize the distribution of domains. This experiment contains 12 cases as Table 5.

As Table 5 shows that the prediction accuracy of first layer of CTTL of most cases is higher than DDC. The prediction accuracy of 0-3 is 89.78%, improves 8.65%, 1-0 is 88.76%, improves 6.7%. Especially the transfer task ‘a’ to ‘b’ is different from the transfer task ‘b’ to ‘a’. Such as the accuracy of 0-1 is 85.05% but 1-0 is 88.76%. Although in some transfer tasks, the accuracy of first layer of CTTL is less than DDC, such as 1-2, 2-1 etc., the average accuracy is 91.43%, improves 1.62%.

To further illustrate the improvement of the initial CTTL on the diagnosis results, the Confusion Matrix Figure 10 and Figure 11 are plotted. Figure 10 shows the confusion matrix for the DDC of 0-3 transfer task, where true label 4
W. Zhang et al.: Convolutional Neural Network Based Two-Layer Transfer Learning for Bearing Fault Diagnosis

TABLE 4. Operating conditions for XJTU-SY dataset.

| Conditions                 | Bearing Dataset | Number of file | Fault element | Sample selection range | Sample size |
|----------------------------|-----------------|----------------|---------------|------------------------|-------------|
| Condition 1:               |                 |                |               |                        |             |
| 2100 rpm (35 Hz)           | Bearing 1.1     | 123            | Outer race    | /                      | /           |
| 12 kN                      | Bearing 1.2     | 161            | Outer race    | /                      | /           |
|                            | Bearing 1.3     | 158            | Outer race    | /                      | /           |
|                            | Bearing 1.4     | 122            | Cage          | /                      | /           |
|                            | Bearing 1.5     | 52             | Inner/Outer race | /                      | /           |
| Condition 2:               |                 |                |               |                        |             |
| 2250 rpm (37.5 Hz)         | Bearing 2.1     | 491            | Inner race    | IR:475-484             | 300         |
| 11 kN                      | Bearing 2.2     | 161            | Outer race    | OR:140-149             | 300         |
|                            | Bearing 2.3     | 533            | Cage          | /                      | /           |
|                            | Bearing 2.4     | 42             | Outer race    | /                      | /           |
|                            | Bearing 2.5     | 339            | Outer race    | N:40-49                | 300         |
| Condition 3:               |                 |                |               |                        |             |
| 2400 rpm (40 Hz)           | Bearing 3.1     | 2538           | Outer race    | N:200-209              | 300         |
| 10 kN                      | Bearing 3.2     | 2496           | Inner/Ball/Cage/Outer race | /                      | /           |
|                            | Bearing 3.3     | 371            | Inner race    | IR:350-359             | 300         |
|                            | Bearing 3.4     | 1515           | Inner race    | /                      | /           |
|                            | Bearing 3.5     | 114            | Outer race    | OR:100-109             | 300         |

FIGURE 10. Confusion matrix of the task 0-3 of DDC.

FIGURE 11. Confusion matrix of the task 0-3 of first layer of CTTL.

(AIR 0.014) is predicted to label 2 (B 0.021) with the highest probability for 66% misclassification. True label 7 (OR 0.014) is predicted to label 0 (B 0.007) and 4 with the highest probability for 100% misclassification. Figure 11 shows the confusion matrix of the proposed initial CTTL, where the misclassification rate of true label 4 is reduced to 5%, true label 7 is reduced to 93%. This shows that using WMMD as the domain adaptation function in most cases of this paper can be more reasonable for fault transfer tasks.

A. COMPARISON WITH OTHER METHODS

To get a better comparison, we further compare CTTL with other methods under 12 cases transfer tasks. The comparison results are shown in Table 6.

In order to do justice to the comparative approach, the CNN, DDC and D-CORAL are based on the same network framework found in this paper. The SVM uses the RBF kernel function with parameters $C = 100$ and gamma = 0.01.

A common deep learning method is CNN, which uses source domain datasets for training and target domain datasets for testing. It is crucial to note that although CNN outperforms SVM in terms of performance, its accuracy is still lower than these transfer methods such as Deep CORAL and DDC.
Deep CORAL [34] is a deep transfer method that adapted the second-order statistical features (covariance) [36] of the source and target domain distributions utilising a nonlinear transformation of the deep network. Another method that many academics have focused on to improve fault diagnosis performance is the improved preprocessing method of converting the raw signal into an image. The DFCNN [33] converts the sampled 784 points into a 28$\times$28 grayscale image for training, which can learn deeper fault features, with an accuracy improvement of 2.85% compared to CNN. DTLCNN [35], converts the sampled 25600 points into a 160$\times$160 grayscale image and performs Multi-kernel MMD (MK-MMD) function in the last two FC layers (100 nodes) for domain adaptation, with an accuracy higher than DDC by 3.18%.

The proposed CTTL method improves the MMD domain adaptation function in the DDC method and achieves a more excellent accuracy by reusing the prediction labels via the DBSCAN method and controlling the times of iterations by CH index. In the first layer of CTTL, a larger Batchsize (200) can be employed for the learning distribution of the target domain. In the second layer of CTTL, due to the introduction of a third dataset, a small Batchsize (50) is used to learn more detailed knowledge of fault classes rather than the whole target domain. The average accuracy of the proposed CTTL method is 94.29%, which is 4.24% better than DDC.

To make a better comparison of the diagnosis performance of transfer methods, Figure 12 shows a histogram of the percentage improvement in accuracy of the three transfer methods, D-CORAL, DTLCNN and CTTL, over the DDC which considered as the baseline.

Figure 12 shows that CTTL outperforms DDC diagnostics in all cases and was better than Deep-CORAL and DTLCNN in most cases. In the transfer task 0-3 and 2-0 cases, CTTL iterates only once and has worse diagnostics accuracy than DTLCNN, which indicates that converting signals to pictures for training has better feature learning capabilities in some cases. In the other cases, CTTL has better accuracy than DTLCNN, which indicates that the iterative controller of CTTL has a positive effect, which will better serve the transfer tasks of fault diagnosis.

In Table 6, Times represents the number of iterations of CTTL. To clearly demonstrate this process, this experiment using transfer task 0-2 and plots the t-distributed Stochastic Neighbour Embedding (t-SNE) [37] visualization for each round time, as shown in Figure 13. In Figure 13, (a) is the visualization of the first iteration, (b) is the visualization of the second iteration (c) is the visualization of the third iteration, (d) is the visualization of the correct labels. As shown in the plots (a)-(c), class 5 (IR 0.014) separates progressively better, class 7 (OR 0.014) predicts incorrectly, and the other classes do not change much. The accuracy rate and CH index during this iteration are shown in Figure 14.

As shown in Figure 14, the prediction accuracy of first-time is 82.41%, and the CH index of the Third Dataset generated by the DBSCAN algorithm is 3683.34. The accuracy of second-time is 85.50% and the CH index is 3733.44.

**TABLE 6.** Full transfer accuracy (%) table comparison with other transfer methods.

| Tasks (HP) | SVM | DFCNN | D-CORAL | DDC | DTLCNN | CTTL | Times |
|-----------|-----|-------|---------|-----|--------|------|-------|
| 0-1       | 45.26 | 80.58 | 83.35   | 88.63 | 83.96 | 86.66 | 89.75 | 2    |
| 0-2       | 44.37 | 76.21 | 79.92   | 86.85 | 81.31 | 86.37 | 87.75 | 4    |
| 0-3       | 39.62 | 72.33 | 75.15   | 86.53 | 81.13 | 89.79 | 89.78 | 1    |
| 1-0       | 62.31 | 76.68 | 76.33   | 89.42 | 82.03 | 88.57 | 89.38 | 3    |
| 1-2       | 76.24 | 97.32 | 97.61   | 99.02 | 97.02 | 99.56 | 99.61 | 2    |
| 1-3       | 76.41 | 96.32 | 97.88   | 98.55 | 97.51 | 98.91 | 99.31 | 2    |
| 2-0       | 64.39 | 71.25 | 70.08   | 87.75 | 85.67 | 87.82 | 87.7 | 1    |
| 2-1       | 76.58 | 94.28 | 96.85   | 99.16 | 97.91 | 99.25 | 99.91 | 2    |
| 2-2       | 80.76 | 91.69 | 97.72   | 99.43 | 96.28 | 98.73 | 99.57 | 2    |
| 2-3       | 64.32 | 71.01 | 71.26   | 89.91 | 82.85 | 85.74 | 89.28 | 3    |
| 2-4       | 71.19 | 81.51 | 93.47   | 99.72 | 96.61 | 97.43 | 99.57 | 2    |
| 3-2       | 71.32 | 94.21 | 97.96   | 99.01 | 98.37 | 99.87 | 99.91 | 2    |
| Ave       | 65.48 | 83.62 | 86.47   | 93.67 | 90.05 | 93.23 | 94.29 | 2    |

**FIGURE 12.** Bar chart of accuracy (%) of proposed CTTL comparison with other methods.
The accuracy of third-time is 87.75% and the CH index is 3797.42. The accuracy of forth-time is 85.66% and CH index is 3768.99, which is less than 3797.42, to end the CTTL iteration process. The two curves in Figure 14 are the accuracy and CH index, which are positively correlated, and the iterative process of the proposed CTTL algorithm has a positive effect of increasing the diagnostic effect.

B. EXPERIMENTS UNDER IMBALANCE OF DATA
In order to show the performance of the CTTL method with unbalanced data and missing fault classes, six cases of experiments are proposed in this section, and the details are shown in Table 7.

In Table 7, the Source domain data (0, 3) represents the dataset mixture of load 0 and load 3 (total 8000∗2 samples). The Target domain data (2) represents the dataset load 2 is selected, and the fault classes indicate the class selected in the Target domain data, e.g., (4, 7) indicates that only two fault classes, IR 0.014 and OR 0.014, are extracted from the Target domain data (total 800∗2 samples). The results of the experiments are represented in Table 8.

As shown in Table 8, the accuracy of CTTL was maintained highest in all six cases compared to other two transfer
W. Zhang et al.: Convolutional Neural Network Based Two-Layer Transfer Learning for Bearing Fault Diagnosis

TABLE 7. Setting of experimental condition.

| Group | Source domain | Target domain | Fault classes |
|-------|---------------|---------------|---------------|
| A     | 0,2,3         | 1             | all           |
| B     | 1,2,3         | 0             | all           |
| C     | 0,2,3         | 1             | 1             |
| D     | 0,3           | 2             | 4,7           |
| E     | 0,1,3         | 2             | 0,3,6         |
| F     | 1,2           | 3             | 2,5,8,9       |

TABLE 8. The accuracy (%) table of data unbalance experiment.

| Method   | A   | B   | C   | D   | E   | F   |
|----------|-----|-----|-----|-----|-----|-----|
| CTTL     | 99.62 | 89.33 | 98.67 | 98.56 | 98.85 | 99.71 |
| DDC [27] | 97.46 | 88.08 | 96.25 | 98.50 | 95.29 | 98.62 |
| D-CORAL [34] | 99.33 | 89.13 | 97.25 | 76.62 | 68.95 | 67.93 |

methods. However, it is noteworthy that Deep CORAL achieves a lower accuracy when only a few fault classes are involved, because the domain adaptation function CORAL adjusts the covariance of the two domains, and the imbalance of the classes affects it greatly. Mosaic plots of the prediction results of Deep CORAL and CTTL are shown in Figure 15 and 16.

In Figure 15 and 16, the horizontal axis represents the fault classes in the Target domain sample, the color blocks on the vertical axis represent the predicted classes, and the mosaic plot determines the area of the color blocks based on the percentage of predicted accuracy. The mosaic plot provides a more intuitive picture of the fault prediction [38], as in Figure 15, fault class 2 is most likely to confuse faults class 4 and 0, fault class 5 is most likely to confuse fault class 1, etc.

C. EXPERIMENT IN GENERALIZABILITY

XJTU-SY dataset is a full life-cycle dataset with more noise contained in each type of fault data. Condition2 dataset is visualized in Figure 17 by blue box, where the inner race (IR) is relatively dispersed. Condition3 dataset is visualized in Figure 17 by green box, where both OR and IR are relatively dispersed. At the same time, it can be seen that there is a large separation of the same label data between the two domains, which means that it is difficult to use one domain data to train another domain data for testing, because of the different data distributions. This condition provides a greater challenge for transfer tasks.

The results of the generalization experiments are shown in Table 9, the quality of the dataset is described in this
TABLE 9. Comparison table of generalization experiments.

| Cohesion | 1HP | 2HP | Condition 2 | Condition 3 |
|----------|-----|-----|-------------|-------------|
| Wasserstein distance | 0.0004 | 0.03 |
| CNN (%) | 1HP-2HP: 97.32 | Condition 2-3: 70.33 |
| DFCNN (%) | 1HP-2HP: 97.61 | Condition 2-3: 71.87 |
| DDC (%) | 1HP-2HP: 96.85 | Condition 2-3: 73.26 |
| DTLCNN (%) | 1HP-2HP: 99.56 | Condition 2-3: 96.51 |
| CTL (%) | 1HP-2HP: 99.91 | Condition 2-3: 99.33 |

section by Cohesion and Wasserstein instance. The equation of Cohesion is shown as (20).

\[
\text{cohesion} = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{N_{C_i}} \sum_{x \in C_i} \text{dist}(x, c_i)
\]  

(20)

where \(k\) is the number of fault classes in the dataset, \(N_{C_i}\) is the number of samples under the class \(i\), \(x\) is a sample of class \(i\), and \(C_i\) is the mean value of the sample. Cohesion denotes the degree of noise content of the sample, the higher the value, the higher the noise intensity, as shown in Table 9, the noise degree of the XJTU-SY dataset is much higher than the CWRU dataset.

To measure the difficulty of the transfer tasks for the CWRU and XJTU-SY datasets, we used the Wasserstein distance to measure the difference in the distribution of the source and target domains data. The larger the Wasserstein distance, the greater the difficulty of transfer learning.

We selected 1HP and 2HP dataset, which performed better in the CWRU dataset, as control experiments. In Table 9, the accuracy of the CNN and DFCNN in the XJTU-SY dataset is much lower than the CNN in Condition 3-2, which indicates that a deeper feature learning capability might lead to poorer diagnostic performance when the difference between the two domains is too large. Due to the large Wasserstein distance between the source and target domains in XJTU, which should be learned by transfer learning to better apply the source domain knowledge to the target domain. Therefore, DDC and DTLCNN can substantially enhance the diagnostic performance. While CTL further improves the accuracy on the XJTU-SY dataset by reusing the predicted labels, which demonstrating the generalization and robustness of CTL on noisier datasets.

V. DISCUSSION

Among the existing bearing fault diagnosis algorithms, many methods can obtain a high accuracy rate, such as MS-DCNN, with the accuracy of 99.27% [39], CNN of LeNet-5 with the accuracy of 99.79% [40], AOCNN with the accuracy of 99.19% [41], and NSAE-LCN with the accuracy of 99.92% [42]. These algorithms assumed that the training data has the identically distribution with the testing data, that is, the data are proportionally sliced into the training and testing sets in the same operating condition. However, the accuracy decreases rapidly when the training data and testing data are from different working conditions. In Table 6, the accuracy of DFCNN is 86.47%, which is much lower than the 99.8% described by Zhang et al. [33] in the literature. It can be seen that transfer learning is able to learn deeper domain invariant features and has a positive effect on industrial fault diagnosis. Therefore, fault diagnosis models trained by CTL can be better deployed to industrial scenarios when there are a large number of labelled samples with different working conditions and a few unlabelled samples with the same working conditions.

At the same time, in order to compare the online deployment capabilities of each algorithm, the data such as Pre-time, Macs and Params are discussed in the Table 10. At a sampling frequency of 48kHz, the DTLCNN has a pre-processing time (pre-time) of 426ms, an inference time (infer-time) of 3.4ms, a sampling time of 163.33ms and a duty cycle of 262.9%, which does not meet the real-time requirements.

\[
\text{duty cycle} = \frac{\text{pre time} + \text{infer time}}{\text{sampling time}}
\]  

(21)

If the image size is further increased, although the duty cycle decreases, it also leads to an increase in Macs, such as DFCNN, with a duty cycle of 8.4% and an increase in Macs of 30.9 M. CTL, however, uses a simple FFT as the pre-processing method, has lower Macs and duty cycle, and can be deployed to poorer hardware environments and higher sampling frequencies. In conclusion, CTL has better online deployment capabilities.

VI. CONCLUSION AND FUTURE RESEARCH

In this paper, a Convolutional Neural Network-based Two-layer Transfer Learning (CTTL) is proposed for bearing fault diagnosis. The first layer of CTL, which is based on CNN feature extractor and feature weighted MMD can enhance the prediction accuracy. The second layer of CTL, which is based on the Third dataset of DBSCAN and CH index reuses the prediction results and obtains the final fault diagnosis with

TABLE 10. Model statistical parameters.

| Method | Macs (M) | Params (M) | length | Pre-time (ms) | Infer-time (ms) | Duty Cycle (%) |
|--------|----------|------------|--------|---------------|----------------|----------------|
| CTTL   | DDC      | 2.82       | 2.16   | 2048          | 10             | 4.3            | 3.35           |
| D-CORAL| DFCNN    | 0.053      | 0.053  | 2560          | 443            | 4.8            | 8.40           |
| DTLCNN |          | 1.17       | 0.065  | 784           | 426            | 3.4            | 262.90         |
high accuracy. For feature visualisation, the t-SNE algorithm is utilised, which can provide a more intuitive feeling of the data distribution and the function of the algorithm. It has been demonstrated that the proposed CTTL has a higher accuracy under the experiments with the CWRU dataset. In the cases of unbalanced fault Target data, CTTL is still able to maintain a good diagnosis performance. Finally, the experiments on the XJTU-SY bearing dataset demonstrate the robustness and generalization of CTTL.

Future research work can be defined in terms of improving the fault diagnosis capabilities and the online deployment adaptability of CTTL.

Firstly, integrated learning [43] can combine the performance advantages of individual models and data pre-processing methods. What can be expected is that we will use more advanced signal to image processing methods and integrate them with frequency signal (FFT) methods to make the algorithms perform even better.

Secondly, the experiments in this paper are trained on existing bearing failure dataset only, but in industrial applications, vibration signals collected by sensors often include signal disturbances from shafts, couplings and gears. This is something that has not been considered in much of the literature [44], [45], [46]. In the next step of our research, we will improve the online deployment of the model by modelling faults such as shafts and gears together with bearing faults; or setting thresholds to determine whether the model is disturbed by faults of other components when compared with the probability values in the final SoftMax layer of the model. These two methods can improve the deployment ability of the algorithm.

### Nomenclature

- $x^t$: Target Dataset.
- $x^s$: Source Dataset.
- $x^{3rd}$: Third Dataset.
- $x_i^t$: i-th target domain data.
- $x_i^s$: i-th source domain data.
- $l_{dom}$: Domain Adaptation loss.
- $l_{cla}$: Classification loss.
- $\xi_s$: Source Feature.
- $\xi_t$: Target Feature.
- $N_S$: Number of source domain data.
- $N_T$: Number of target domain data.
- $S_b$: Between-class scatter matrix.
- $S_w$: Within-class scatter matrix.

### Abbreviations

- TL: Transfer Learning.
- CNN: Convolutional Neural Network.
- SVM: Support Vector Machine.
- MMD: Maximum Mean Discrepancy.
- WMMD: Feature-Weighted Maximum Mean Discrepancy.
- DTL: Deep Transfer Learning.
- DBSCAN: Density-Based Spatial Clustering of Applications with Noise.
- CH: Calinski-Harabaz.
- CWRU: Case Western Reserve University.
- LDA: linear discriminant analysis.
- FC: fully connected.
- SVD: Singular Value Decomposition.
- FFT: fast Fourier transform.
- DDC: Deep Domain Confusion.
- CORAL: CORrelation ALignment.
- D-CORAL: Deep-CORrelation Aalignment.

### References

[1] P. Tamilselvan and P. F. Wang, “Failure diagnosis using deep belief learning based health state classification,” Rel. Eng., Syst. Saf., vol. 115, pp. 124–135, Jul. 2013.
[2] Z. Chen and W. Li, “Multisensor feature fusion for bearing fault diagnosis using sparse autoencoder and deep belief network,” IEEE Trans. Instrum. Meas., vol. 66, no. 7, pp. 1693–1702, Jul. 2017.
[3] X. Li, Y. Li, Y. Zhang, F. Liu, and Y. Fang, “Fault diagnosis of belt conveyor based on support vector machine and grey wolf optimization,” Math. Problems Eng., vol. 2020, pp. 1–10, Sep. 2020.
[4] H. Hu, B. Tang, X. Gong, W. Wei, and H. Wang, “Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks,” IEEE Trans. Ind. Informat., vol. 13, no. 4, pp. 2106–2116, Aug. 2017.
[5] M. He and D. He, “Deep learning based approach for bearing fault diagnosis,” IEEE Trans. Ind. Appl., vol. 53, no. 3, pp. 3057–3065, May/Jul. 2017.
[6] J. Liu, M. Zhang, H. Wang, W. Zhao, and Y. Liu, “Sensor fault detection and diagnosis method for AHU using 1-D CNN and clustering analysis,” Comput. Intell. Neurosci., vol. 2019, pp. 1–20, Sep. 2019.
[7] M. T. Pham, J.-M. Kim, and C. H. Kim, “Rolling bearing fault diagnosis based on improved GAN and 2-D representation of acoustic emission signals,” IEEE Access, vol. 10, pp. 78056–78069, 2022.
[8] L. Wu, B. Yao, Z. Peng, and Y. Guan, “Fault diagnosis of roller bearings based on a wavelet neural network and manifold learning,” Appl. Sci., vol. 7, no. 2, p. 158, Feb. 2017.
[9] Y. Li, “A fault prediction and cause identification approach in complex industrial processes based on deep learning,” Comput. Intell. Neurosci., vol. 2021, pp. 1–13, Mar. 2021.
[10] F. He and Q. Ye, “A bearing fault diagnosis method based on wavelet packet transform and convolutional neural network optimized by simulated annealing algorithm,” Sensors, vol. 22, no. 4, p. 1410, Jun. 2022.
[11] V.-C. Nguyen, D.-T. Hoang, X.-T. Tran, M. Van, and H.-J. Kang, “A bearing fault diagnosis method using multi-branch deep neural network,” Machines, vol. 9, no. 12, p. 345, Dec. 2021.
[12] X. Gong, Z. Zhi, K. Feng, W. Du, and T. Wang, “Improved DCNN based on multi-source signals for motor compound fault diagnosis,” Machines, vol. 10, no. 4, p. 277, Apr. 2022.
[13] M. Azamfar, X. Li, and J. Lee, “Intelligent ball screw fault diagnosis using a deep domain adaptation methodology,” Mech. Mach. Theory, vol. 151, Sep. 2020, Art. no. 103932.
[14] J. Wu, T. Tang, M. Chen, Y. Wang, and K. Wang, “A study on adaptation lightweight architecture based deep learning models for bearing fault diagnosis under varying working conditions,” Expert Syst. Appl., vol. 160, Dec. 2020, Art. no. 113710.
[15] Z. Fan, Q. Xu, C. Jiang, and S. X. Ding, “Weighted quantile discrepancy-based deep domain adaptation network for intelligent fault diagnosis,” Knowl.-Based Syst., vol. 240, Mar. 2022, Art. no. 108149.
[16] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, “Universal domain adaptation in fault diagnostics with hybrid weighted deep adversarial learning,” IEEE Trans. Ind. Informat., vol. 17, no. 12, pp. 7975–7986, Dec. 2021.
[17] W. Zhang, T. Zhang, G. Cui, and Y. Pan, “Intelligent machine fault diagnosis using convolutional neural networks and transfer learning,” IEEE Access, vol. 10, pp. 50959–50973, 2022.
[18] R. Zhang, H. Tao, L. Wu, and Y. Guan, “Transfer learning with neural networks for bearing fault diagnosis in changing working conditions,” IEEE Access, vol. 5, pp. 14347–14357, 2017.
[19] Y. Wang, S. Li, F. Jia, and J. Shen, “Multi-domain weighted transfer adversarial network for the cross-domain intelligent fault diagnosis of bearings,” Machines, vol. 10, no. 5, p. 326, Apr. 2022.
[20] X. Li, W. Zhang, H. Ma, Z. Luo, and X. Li, “Partial transfer learning in machinery cross-domain fault diagnostics using class-weighted adversarial networks,” *Neural Netw.*, vol. 129, pp. 313–322, Sep. 2020.

[21] S. Zang, Y. Cheng, X. Wang, and Y. Yan, “Transfer extreme learning machine with output weight alignment,” *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–14, Feb. 2021.

[22] C. Cheng, B. Zhou, G. Ma, D. Wu, and Y. Yuan, “Wasserstein distance based deep adversarial transfer learning for intelligent fault diagnosis with unlabeled or insufficient labeled data,” *Neurocomputing*, vol. 409, pp. 35–45, Oct. 2020.

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

[24] Y. Wang, D. Wu, and X. Yuan, “LDA-based deep transfer learning for fault diagnosis in industrial chemical processes,” *Comput. Chem. Eng.*, vol. 140, Sep. 2020, Art. no. 106964.

[25] C. Ma, Q. Tian, and H. Sun, “Unsupervised domain adaptation based on double-weighed deviation modeling,” *Comput. Sci.*, vol. 48, pp. 217–223, Dec. 2021.

[26] A. Gretton, K. Borgwardt, and M. Rasch, “A kernel method for the two-sample-problem,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, 2007, pp. 1–10.

[27] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, “Deep domain confusion: Maximizing for domain invariance,” 2014, arXiv:1412.3474.

[28] A. Ram, S. Jalal, A. S. Jalal, and M. Kumar, “A density based algorithm for discovering density varied clusters in large spatial databases,” *Int. J. Comput. Appl.*, vol. 3, no. 6, pp. 1–4, Jun. 2010.

[29] J. Wang, W. Zhang, T. Hua, and T.-C. Wei, “Unsupervised learning of topological phase transitions using the Calinski–Harabaz index,” *Phys. Rev. Res.*, vol. 3, no. 1, Jan. 2021, Art. no. 013074.

[30] W. A. Smith and R. B. Randall, “Rolling element bearing diagnostics using neural network architecture search and sharing weights,” *IEEE Trans. Rel.*, vol. 69, no. 1, pp. 401–412, Mar. 2020.

[31] A. Ram, S. Jalal, A. S. Jalal, and M. Kumar, “A density based algorithm for discovering density varied clusters in large spatial databases,” *Int. J. Comput. Appl.*, vol. 3, no. 6, pp. 1–4, Jun. 2010.

[32] J. Zhu, N. Chen, and C. Shen, “A new deep transfer learning method for bearing fault diagnosis under different working conditions,” *IEEE Sensors J.*, vol. 20, no. 15, pp. 8394–8402, Aug. 2019.

[33] B. Sun, J. Feng, and K. Saenko, “Return of frustratingly easy domain adaptation,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, no. 1, 2016, pp. 1–10.

[34] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, p. 11, Nov. 2008.

[35] J. Raymaekers, P. J. Rousseeuw, and M. Hubert, “Class maps for visualizing classification results,” *Technometrics*, vol. 64, no. 2, pp. 151–165, Apr. 2022.

[36] X. Li, W. Zhang, H. Ma, Z. Luo, and X. Li, “Partial transfer learning in machinery cross-domain fault diagnostics using class-weighted adversarial networks,” *Neural Netw.*, vol. 129, pp. 313–322, Sep. 2020.

[37] S. Zang, Y. Cheng, X. Wang, and Y. Yan, “Transfer extreme learning machine with output weight alignment,” *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–14, Feb. 2021.

[38] C. Cheng, B. Zhou, G. Ma, D. Wu, and Y. Yuan, “Wasserstein distance based deep adversarial transfer learning for intelligent fault diagnosis with unlabeled or insufficient labeled data,” *Neurocomputing*, vol. 409, pp. 35–45, Oct. 2020.

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

[40] Y. Wang, D. Wu, and X. Yuan, “LDA-based deep transfer learning for fault diagnosis in industrial chemical processes,” *Comput. Chem. Eng.*, vol. 140, Sep. 2020, Art. no. 106964.

[41] C. Ma, Q. Tian, and H. Sun, “Unsupervised domain adaptation based on double-weighed deviation modeling,” *Comput. Sci.*, vol. 48, pp. 217–223, Dec. 2021.

[42] A. Gretton, K. Borgwardt, and M. Rasch, “A kernel method for the two-sample-problem,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, 2007, pp. 1–10.

[43] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, “Deep domain confusion: Maximizing for domain invariance,” 2014, arXiv:1412.3474.

[44] A. Ram, S. Jalal, A. S. Jalal, and M. Kumar, “A density based algorithm for discovering density varied clusters in large spatial databases,” *Int. J. Comput. Appl.*, vol. 3, no. 6, pp. 1–4, Jun. 2010.

[45] J. Wang, W. Zhang, T. Hua, and T.-C. Wei, “Unsupervised learning of topological phase transitions using the Calinski–Harabaz index,” *Phys. Rev. Res.*, vol. 3, no. 1, Jan. 2021, Art. no. 013074.

[46] W. A. Smith and R. B. Randall, “Rolling element bearing diagnostics using the case western reserve university data: A benchmark study,” *Mech. Syst. Signal Process.*, vols. 64–65, pp. 100–131, Dec. 2015.

[47] A. Heng, Z. Zhang, A. C. C. Tan, and J. Mathew, “Rotating machinery prognostics: State of the art, challenges and opportunities,” *Mech. Syst. Signal Process.*, vol. 23, no. 3, pp. 724–739, 2009.

[48] B. Wang, Y. Lei, N. Li, and N. Li, “A hybrid prognostics approach for estimating remaining useful life of rolling element bearings,” *IEEE Trans. Rel.*, vol. 69, no. 1, pp. 401–412, Mar. 2020.

[49] J. Zhu, N. Chen, and C. Shen, “A new deep transfer learning method for bearing fault diagnosis under different working conditions,” *IEEE Sensors J.*, vol. 20, no. 15, pp. 8394–8402, Aug. 2019.

[50] B. Sun, J. Feng, and K. Saenko, “Return of frustratingly easy domain adaptation,” in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, no. 1, 2016, pp. 1–10.

[51] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, p. 11, Nov. 2008.

[52] J. Raymaekers, P. J. Rousseeuw, and M. Hubert, “Class maps for visualizing classification results,” *Technometrics*, vol. 64, no. 2, pp. 151–165, Apr. 2022.

[53] Z. Zilong and Q. Wei, “Intelligent fault diagnosis of rolling bearing using one-dimensional multi-scale deep convolutional neural network based health state classification,” in *Proc. IEEE 15th Int. Conf. Netw., Sens. Control (ICNSC)*, Mar. 2018, pp. 1–5.

[54] L. Wen, X. Li, L. Gao, and Y. Zhang, “A new convolutional neural network-based data-driven fault diagnosis method,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5990–5998, Jul. 2018.

[55] W. Qian, S. Li, J. Wang, Z. An, and X. Jiang, “An intelligent fault diagnosis framework for raw vibration signals: Adaptive overlapping convolutional neural network,” *Meas. Sci. Technol.*, vol. 29, no. 9, Sep. 2018, Art. no. 095009.

[56] F. Jia, Y. Lei, L. Guo, J. Lin, and S. Xing, “A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines,” *Neurocomputing*, vol. 272, pp. 619–628, Sep. 2018.

[57] C. Che, H. Wang, Q. Fu, and X. Ni, “Deep transfer learning for rolling bearing fault diagnosis under variable operating conditions,” *Adv. Mech. Eng.*, vol. 11, no. 12, Dec. 2019, Art. no. 1687814019897212.