IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 30, 2019, accepted May 12, 2019, date of publication June 6, 2019, date of current version July 3, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2921480*

# Domain Adaptive Motor Fault Diagnosis Using Deep Transfer Learning

**DENGYU XIAO, YIXIANG HUANG, LUJIE ZHAO, CHENGJIN QIN, HAOTIAN SHI, AND CHENGLIANG LIU**

State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 201100, China

Corresponding author: Yixiang Huang (huang.yixiang@sjtu.edu.cn)

This work was supported in part by the National Key Technology Research and Development Program of China under Grant 2017YFB1302004, and in part by the National Natural Science Foundation of China Project under Grant 51305258.

**ABSTRACT** Motor fault diagnosis based on deep learning frameworks has gained much attention from academic research and industry to guarantee motor reliability. Those methods are commonly under two default assumptions: 1) massive labeled training samples and 2) the training and test data share a similar distribution under unvarying working conditions. Unfortunately, these assumptions are nearly invalid in a real-world scenario, where the signals are unlabeled and the working condition changes constantly, resulting in the diagnosis models of the previous studies that always fail in classifying the unlabeled data in real applications. To deal with those issues, in this paper, we propose a novel feature adaptive motor fault diagnosis using deep transfer learning to improve the performance by transferring the knowledge learned from labeled data under invariant working conditions to the unlabeled data under constantly changing working conditions. A convolutional neural network (CNN) is adopted as the base framework to extract multi-level features from raw vibration signals. Then, the regularization term of maximum mean discrepancy (MMD) is incorporated in the training process to impose constraints on the CNN parameters to reduce the distribution mismatch between the features in the source and target domains. To verify the effectiveness of our proposal, data from the motor tests of European driving cycle (NEDC) for simulating the real working scenario and the motor tests under invariant working conditions are, respectively, conducted as the target domain and the source domain. The results show that the proposal presents higher diagnosis accuracy for the unlabeled target data than other methods, and it is of applicability to bridge the discrepancy between different domains.

**INDEX TERMS** Motor fault diagnosis, transfer learning, domain adaptation, convolutional neural network (CNN).

## I. INTRODUCTION

Motor is the key component of many mechanical systems as the power source, in which a huge demand exists for higher reliability, security, and availability [1]. Motor fault diagnosis and health prognostics plays an important role in reducing unnecessary maintenance operations, attaining the enhancement of security and improving the reliability of motors [2]. In recent years, many effective attempts have been made to design novel algorithms to achieve superior diagnosis performance. Owing to the significant development of sensor technology and computing ability, deep learning, which is thought as an effective tool for the high-level features learning based on the nonlinear transformations through multiple layers, has been widely and successfully applied in some motor fault diagnosis applications [3]. Several deep learning techniques such as sparse deep stacking network (SDSN) [4], Long Short-Term Memory (LSTM) network [5], Convolutional Neural Network (CNN) [6], etc. are successfully used to identify the motor faults.

However, the use of the diagnosis approaches in the previous studies is limited in bench tests and hard to extend to real-world scenario. It can be interpreted by two reasons. 1) The previous studies are mostly trained and validated by massive labeled faulty data, which is only available in bench tests and inadequate in real applications. That is because motors in real working scenario are mostly operated under a healthy state and even though fault occurs, the faulty motor will be shut down immediately, which leads to the lack of labeled samples. 2) In the previous studies, datasets are acquired under relatively stable motor working conditions, where the load and speed are unvarying or

The associate editor coordinating the review of this manuscript and approving it for publication was Chuan Li.

linearly increase\ decrease at most. While in real applications, the working condition constantly changes over time. In summary, the good performances of the traditional deep learning methods in motor fault diagnosis owes to the supervised learning with massive labeled data under invariant working conditions. It still remains as a challenge to detect the motor faults with unlabeled data under constantly changing working conditions.

Therefore, transfer learning, which possesses the capacity to leverage the knowledge learnt from source domain to target domain [7], provides a promising solution to address this issue. As an important branch of machine learning, transfer learning focuses on applying the knowledge which has been researched before in one domain to a new domain [8]. The dataset in the target domain shares similar knowledges but dissimilar probability distribution with that in the source domain. The core of transfer learning is to find the similarities between different domains and improve the performance of the classification or regression model for the target domain with the help of the knowledges learned in the source domain [9]. Transfer learning has been widely researched and gained some attainments in the fields including natural language processing [10], [11] and computer vision [12]–[14]. According to the discrepancy of learning methods, transfer learning can be classified into four major categories: instance based, feature based, model based, and relationship based transfer learning. Recently transfer learning has emerged as a novel effective method for machinery fault diagnosis under different operating conditions [15]. First application of deep transfer learning (DTL) on fault diagnosis was proposed by Lu *et al.* [16], which was based on maximum mean discrepancy (MMD) and weight regularization. Wen et al. proposed a DTL method for bearing fault diagnosis. A sparse auto-encoder of three layers was used to scrutinize the raw data and extract the features, and the MMD term was applied to minimize the discrepancy penalty between the source data and the target data [17]. Shao et al. used pre-trained network to extract lower level features and labeled images from wavelet transformation to fine-tune the higher-levels to conduct the transfer tasks [18]. Zhang et al. used a model-based transfer learning method for a small sample problem based on multiple fully-connected neural networks [19]. Guo et al. designed a deep convolutional transfer learning network (DCTLN) to transfer the features extracted from labeled data of one machine to the unlabeled data of other machines [20]. Yang et al. developed feature-based transfer neural network named FTNN to improve the diagnosis accuracy for real-case machines by extracting the transferable features from laboratory machines [21].

However, those above-mentioned researches encounter with two shortcomings. 1) Those transfer learning applications in machinery fault diagnosis mainly focus on the transfer tasks between different working conditions or different machines. The used target and source data are both under unvarying working conditions, where the load and motor speed remain unchanged during the data acquisition process. They devoted little consideration to the real-world scenario and the problem about how to use transfer learning to adapt the labeled data under invariant working conditions to the unlabeled data under constantly changing working conditions is still not addressed well. 2) The regularization terms of multi-layer MMDs used in previous studies are endowed with a unique weight in back propagation. It requires a deeper investigation about the selection of different weights for MMDs and its effect to diagnosis performances.

In order to overcome the shortcomings above, in this paper, we proposed a deep transfer learning framework based on CNN and the regularization term of MMD to identify the health states of motors under constantly changing working conditions, with the help of the labeled data from motors under unvarying working conditions. In the proposed method, a CNN structure is adopted to simultaneously extract the multi-layer features of the raw vibration data from both source domain and target domain. MMD acts as a regularization term embedded in the optimizer object in the mini-batch supervised back-propagation training process. It aims to reduce the distributions discrepancy between two latent feature spaces drawn from different domains. The major contributions of this paper are summarized as follows:

1) To correctly classify the unlabeled faulty motor data under changing working conditions, a novel deep transfer learning framework with CNN and MMD is proposed, aiming to bridge the distribution discrepancy of multi-level features and transfer the knowledges learned from the labeled motor data under unvarying working conditions to the unlabeled data under constantly changing working conditions.
2) To acquire the data as close as possible to the real-world scenario, faulty motor tests of European Driving Cycle (NEDC) are conducted to collect the data under changing working conditions as the target data.
3) To find the relationship between the diagnosis performance and the MMD weights at different feature levels, a comparative evaluation of the accuracy with different MMD weight rates is performed.

The remainder of this paper is organized as follows. In Section II, some important notions and principles are presented. In Section III, the proposed framework and the training process are illustrated in details. In Section IV, the verification tests of six induction motors with different fault types under different operating conditions on the diagnostics simulator platform are introduced. In Section V, the classification results of the proposal are illustrated and discussed. Several other state-of-the-art related works are tested and contrasted. Finally, Section VI provides concluding remarks.

## II. BASIC THEORIES OF SOME CONCEPTS

### A. TRANSFER LEARNING

Some basic notations in transfer learning are introduced first. Let $D_S$ and $D_T$ denotes the source and target domain

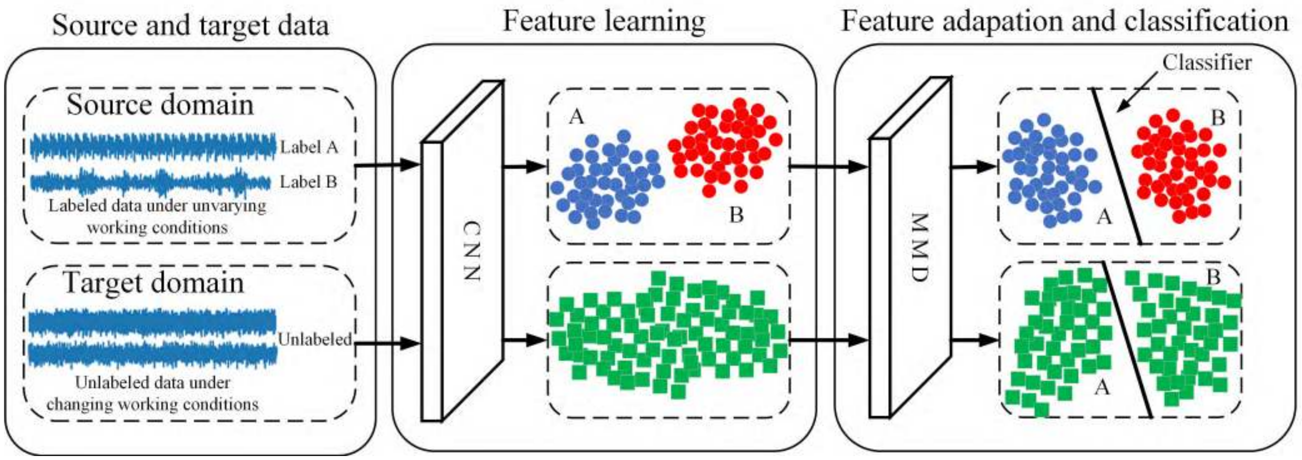

Source and target data
Feature learning
Feature adapation and classification
Source domain
Label A
Label B
Labeled data under unvarying working conditions
Target domain
Unlabeled
Unlabeled data under changing working conditions
CNN
A
B
MMD
Classifier
A
B
A
B

**FIGURE 1.** The overview of the transfer learning procedures using CNN and MMD.

respectively. A domain consists of a sample set and a probability distribution, thus $\{D_S = X_S, P(x_S)\}$, $\{D_T = X_T, P(x_T)\}$, where $X_S$ and $X_T$ respectively represents the source samples and target samples, $x_S \in X_S$, $x_T \in X_T$. $P(x_S)$ and $P(x_T)$ denotes the marginal probability distributions the source samples and the target samples are subject to. When $P(x_S) = P(x_T)$, the predictive models can be built by traditional deep learning methods. When $P(x_S) \neq P(x_T)$, transfer learning emerges as an promising tool to facilitate constructing the qualified target predictive function $F_T(X)$ in $D_T$ by using the related knowledges or information in $D_S$.

In this paper, $D_S$ is the labeled data from unvarying working conditions, and $D_T$ is the unlabeled data from changing working conditions. Thus $X_S = \{x_S^i, y_S^i\}_{i=1}^{ns}$ with $n_s$ labeled samples, and $X_T = \left\{x_T^j\right\}_{j=1}^{nt}$ with $n_t$ unlabeled samples. The unlabeled samples in $D_T$ are expected to be correctly classified by using both the $n_s$ labeled samples in $D_S$ and the $n_t$ unlabeled samples in $D_T$. To achieve this goal, $X_S$ and $X_T$ are simultaneously fed into a CNN to extract different-levels features in different layers, and MMD is adopt to adapt the feature distributions $P(F_S)$ in $D_S$ to the feature distributions $P(F_T)$ in $D_T$. The overview of the transfer process can be seen in Fig.1.

## B. MAXIMUM MEAN DISCREPANCY

The Maximum Mean Discrepancy (MMD) is a kind of distance metric for probability distributions between two datasets. It is a valid criterion that can measure the distribution difference without considering the initial density functions [22]. Given the nonlinear mapping function $H(\cdot)$ in a reproducing Kernel Hilbert space (RKHS) $\mathscr{H}$, MMD between $X_S$ and $X_T$ can be defined as

$$MMD_{\mathcal{H}|}(X_S, X_T) = sup[\frac{1}{n_S}\sum_{i=1}^{ns} H\left(x_S^i\right) - \frac{1}{n_t}\sum_{j=1}^{n_t}\left(x_T^j\right)] \tag{1}$$

where $sup(\cdot)$ represents the supremum of the aggregate no matter what $H(\cdot)$ is selected. $n_s$ is the source sample number and $n_t$ is the target sample number. To simplify the function above, some kernels such as Gaussian kernels are introduced.

The MMD can be rewritten with embedding kernels as follows to empirically estimate.

$$\begin{aligned} &MMD_{\mathrm{e}}(X_S, X_T) \\ &= \left(\frac{1}{n_s^2}\sum_{i=1}^{n_s}\sum_{j=1}^{n_s} K\left(x_S^i, x_S^j\right) + \frac{1}{n_t^2}\sum_{i=1}^{n_t}\sum_{j=1}^{n_t} K\left(x_T^i, x_T^j\right)\right. \\ &\quad \left. - \frac{1}{n_s n_t}\sum_{i=1}^{n_s}\sum_{j=1}^{n_t} K\left(x_S^i, x_T^j\right)\right)^{1/2} \end{aligned} \tag{2}$$

where $K(\cdot)$ represents the kernel function, $x_S^i \in X_S$, and $x_T^i \in X_T$.

## C. CONVOLUTIONAL NEURAL NETWORK

CNN, as a type of most state-of-the-art deep learning model, has been effectively used for a variety of fault diagnosis and health monitoring problems [23]. It is further effective for reducing computational burden by the use of weight sharing and local receptive field strategies [24]. Typically, a CNN consists of three types of layers including convolutional layers, pooling layers and fully-connected layers. The convolutional layer is the core building block. It contains a set of trainable filters named as kernels. Generally the kernel length should be smaller than the input data length. Let $X_n$ denotes the $n$th input data point of the convolutional layer, and $N$ be the number of such data points. The $n$th segment is denoted as $I_n^{j:j+l_k}$, where $l_k$ is the kernel length, and $j$ represents the $j$th data point. The general form of the convolution process is as expressed as followed:

$$Y_k^j = f(\sum_{n=1}^{N} X_n^{j:j+l_k} * w_{k,n} + b_n) \tag{3}$$

where $*$ denotes a one dimensional convolution operation, $w_{k,n}$ is the convolution kernel connecting the $n$th group to the $k$th group, $b_n$ is the bias vector, $f(\cdot)$ is the activation function, and $Y_k^j$ is the output of the convolutional operation. A pooling layer is added after the convolutional layer in order to eliminate the dimensionality curse of the convolutional layer. In a pooling layer, first the input data is segmented into a set of sub-region. Then, the ideal output of each sub-region is calculated through the pooling function. In this approach, we utilize the Max-pooling function to return the maximum value of each sub-region. The general form of the process is expressed as follows:

$$P_n^j = \max\{Y_n^{j*l_p:(j+1)*l_p}\} \tag{4}$$

where $l_p$ is the length of sub-region and $P_n^j$ is the output value of the $j$th group from the $n$ data points. For each channel of training data, the learned shift invariant multi-dimensional features are extracted through the convolution and pooling processing.

A flatten layer is added after the pooling layers to convert the multiple dimensions of features to one dimension. After that, fully-connected layers are added and the neurons at different layers are all connected to each other. The activation functions of ReLU at the first few layers and the activation function of Softmax at the last layer are selected.

## III. THE PROPOSED METHOD

In this paper, a feature-based deep transfer learning method is proposed based on a CNN architecture embedded with MMD. The core of the transfer task is to encourage the multi-level representations in different hidden layers induced by the source samples to be close enough to those induced by the target samples. It is mainly achieved by incorporating the multi-layer MMD terms between two domains of features into the loss function during the optimization process. Generally, the proposal contains two steps: feature learning and domain adaptation. For feature learning, a CNN is adopted to extract multi-level representations in different hidden layers induced by both source and target domains. Note that, the source samples and target samples are normalized to the range of [0,1] and then bound together as the input with sample length. For domain adaptation, MMD is used as the nonparametric distance metric to measure the distribution discrepancy of the multi-layer features. Then, the measured MMDs with various weights are regarded as the regularization terms in the loss function to propagate backward to train the CNN architecture, which aims to minimize the distribution mismatch and align the cross-domain features. After that, the unlabeled target samples can be classified by the trained CNN with higher accuracy, because the distributions of the learned multi-level features from the source and target domains are more and more similar, and thus the CNN of higher robustness and adaptability can deal with the data under different working conditions. The flow diagram of the procedures can be seen in Fig. 2.

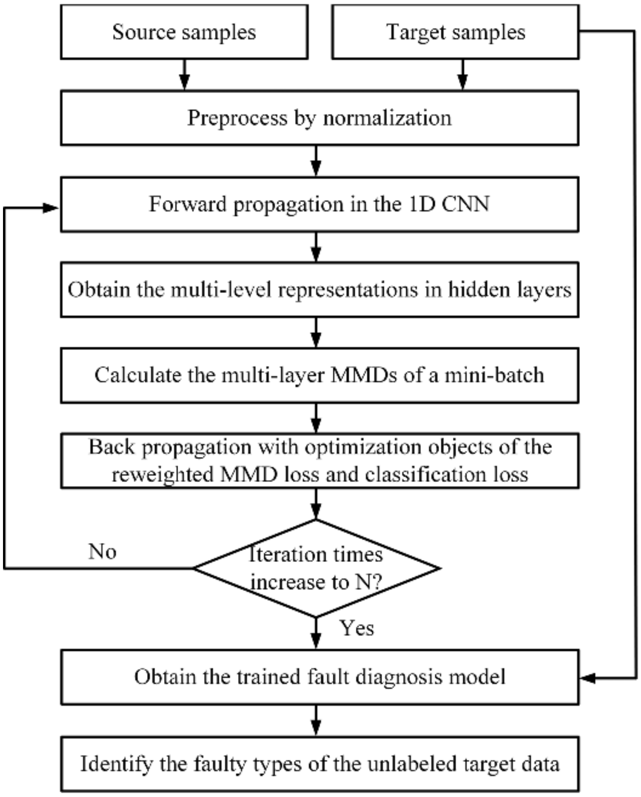

Source samples
Target samples
Preprocess by normalization
Forward propagation in the 1D CNN
Obtain the multi-level representations in hidden layers
Calculate the multi-layer MMDs of a mini-batch
Back propagation with optimization objects of the reweighted MMD loss and classification loss
No
Iteration times increase to N?
Yes
Obtain the trained fault diagnosis model
Identify the faulty types of the unlabeled target data

FIGURE 2. The flow diagram of the proposed framework.

TABLE 1. The architecture of the 1D CNN.

| Layer | Parameters | Activation function | Output size |
|---|---|---|---|
| Input | / | / | 1024*1 |
| C1 | kernel size=64; channels=64; stride=16; padding= 0; | ReLU | 61*64 |
| P1 | max-pooling size=2 | / | 30*64 |
| C2 | kernel size=3; channels=32; stride=1; padding= 2; | ReLU | 32*32 |
| C3 | kernel size=3; channels=32; stride=1; padding= 1; | ReLU | 32*32 |
| C4 | kernel size=3; channels=32; stride=1; padding= 1; | ReLU | 32*32 |
| F0 | / | / | 1024*1 |
| F1 | weights=1024*128; Bias= 128 | ReLU | 128*1 |
| F2 | weights=128*64; Bias= 64 | ReLU | 64*1 |
| F3 | weights=64*6; Bias= 6 | Softmax | 6*1 |

### A. FEATURE LEARNING

Typically, a CNN consists of three types of layers including convolutional layers, pooling layers and fully-connected layers. In this paper, the base learner with 1D-CNN architecture embraces the similar philosophy of the CNN structure in [32], employing wide kernels for the first convolutional layer C1 and small kernels for the next few convolutional layers (C2, C3, C4). Besides, one max-pooling layer (P1), one flatten layer (F0), and three fully-connected layers (F1, F2, F3) are adopted. The details of the 1D-CNN architecture are illustrated in Table. 1.

The forward propagation process is displayed in Fig. 3. We define the feature spaces in different hidden layers induced by the source data is $F_S$. Thus $F_S = \{F_S^{C1}, F_S^{C2}, F_S^{C3}, F_S^{C4} F_S^{F1}, F_S^{F2}\}$, where $F_S^{Ci}$ and $F_S^{Fi}$

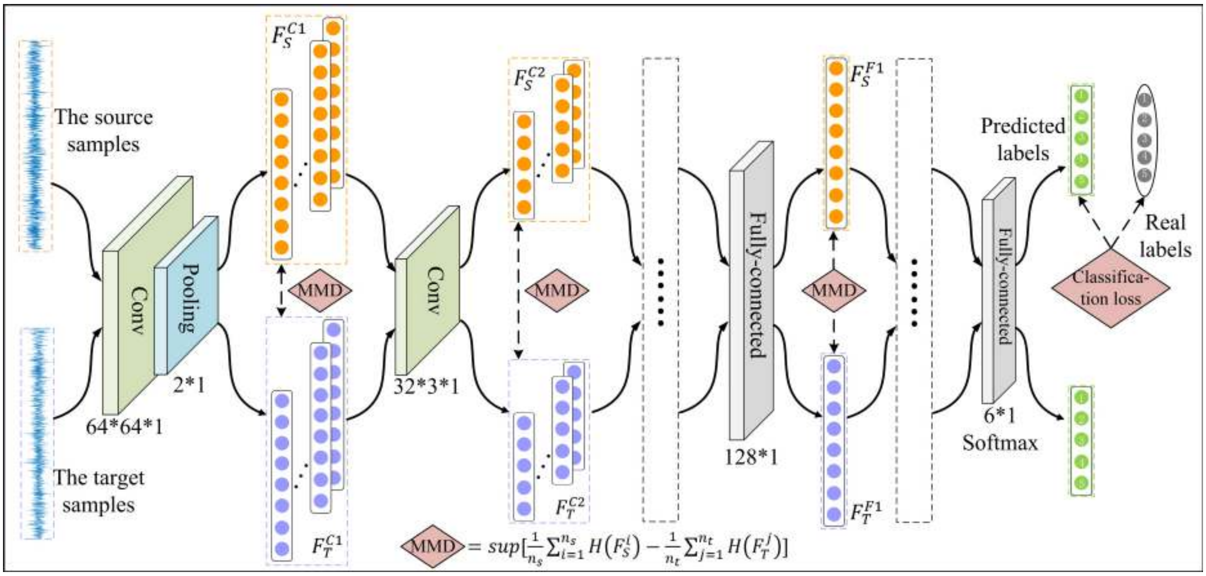


**FIGURE 3.** The extraction and adaptation of the multi-level features in the proposed CNN architecture.

respectively means the representations after the C$i$ layer and F$i$ layer. Similarly, the feature spaces induced by the target data is $F_T$, and $F_T = \{F_T^{C1}, F_T^{C2}, F_T^{C3}, F_T^{C4}, F_T^{F1}, F_T^{F2}\}$. From $F^{C1}$ to $F^{F2}$, the feature space shifts from shallow-level representations to high-level. Note that, each feature space is generated by a mini-batch of source and target samples instead of one sample, which can be described as

$$F^L = \{F^L(i)\}_{i=1}^{n} \tag{5}$$

where $n$ is the batch size, $L$ represents the layer, and $F^L(i)$ denotes the feature space after $L$ layer induced by the $i$ th sample in the mini-batch. The use of batch size can be interpreted by two reasons. 1) The intrinsic advantages of itself, including improving memory utilization through parallelization, correcting the gradient descent direction, and decreasing the amplitude of training vibration [25]. 2) Adapt to the input size of MMD, which requires two datasets each containing multiple samples for distance measurement.

### B. DOMAIN ADAPTATION

In order to obtain the desirable diagnosis performance in the different-distribution target domain, the proposal should be able to learn domain invariant features. Domain invariant features should be subject to the similar distribution, no matter the features are learned from source data or target data. If the learned features from different domains can be automatically aligned to become domain invariant features, the fault diagnosis classifier trained by the source domain data is able to effectively classify the target domain data. However, the multi-level representations in different hidden layers learned from the two domains suffer from the distribution mismatch. As displayed in Fig. 3, to reduce the discrepancy and encourage the feature spaces induced by the source samples to be close enough to those induced by the target samples, a new optimization objective through minimizing the reweighted MMDs of the multi-level representations is added to the loss function. Note that, in this paper, the source and target samples are assumed to share the same label space and same size. In other words, the label order of the source and target domains should be the same, and the source and target samples only differ in the data probability distributions.

As the feature level is different, the importance and contributions of different-layer MMDs for domain adaptation may be not same. To obtain the best transfer performance and assess the contributions of the MMD of each layer, the multi-layer MMDs are endowed with different weights in the loss function. Thus the regularization term about the multi-layer MMDs can be defined as follows.

$$L_{MMD} = \boldsymbol{W_\theta} \cdot MMD_e\left(\boldsymbol{F_S}, \boldsymbol{F_T}\right) \tag{6}$$

where $\boldsymbol{W_\theta}$ is the 6-dimension weight vector with weight rate $\theta = (\boldsymbol{W_\theta}(i+1))/(\boldsymbol{W_\theta}(i))$, MMD vector $MMD_{\mathrm{e}}\left(\boldsymbol{F_S}, \boldsymbol{F_T}\right) = \{MMD_{\mathrm{e}}\left(F_S^l, F_T^l\right)\}_{l=C1}^{l\in\{C1,C2,C3,C4,F1,F2\}}$. The empirical estimate of MMD, $MMD_{\mathrm{e}}$ is calculated by Eq. (2). In this paper, the Gaussian radial basis kernels [26] are taken as the kernel functions as follows.

$$K\left(X_1, X_2\right) = \exp\left(\frac{-\left\|X_1 - X_2\right\|^2}{2\sigma^2}\right) \tag{7}$$

where $\sigma$ is the kernel bandwidth. When $\sigma \to 0$ or $\sigma \to \infty$, $MMD_e$ will both decrease to zero.

### C. TRAINING PROCESS

The proposal is trained through minimizing the loss function which contains two regularization terms: 1) the classification loss $L_C$ between the real labels of the source samples and the predicted labels; 2) the reweighted MMD loss $L_{MMD}$ between the multi-level representations of two domains. The form of

**TABLE 2. 6 faulty motor types in the tests.**

| Fault condition | Abbr. | Class | Description |
|---|---|---|---|
| Healthy | HS | 1 | Healthy state |
| Built-in bowed rotor | BBR | 2 | Consists of a centrally intentionally bent rotor |
| Faulted bearings | FB | 3 | Consists of one inner race faulted bearing and one outer race faulted bearing |
| Broken rotor bars | BRB | 4 | Fitted with an intentionally broken rotor bar |
| Built-in rotor Misalignment | BRM | 5 | Caused by custom machined end bells with asymmetric structure |
| Built-in rotor unbalance | BRU | 6 | Intentionally removing one of the balanced rotors from induction motor and destroying the inner balance |

the loss function is

$$Loss = L_C + \gamma L_{MMD} \tag{8}$$

where $\gamma$ is the tradeoff parameter. The cross-entropy loss function [27] is used for the classification loss.

$$L_C = -\frac{1}{n}\sum\nolimits_x \left\|\hat{y}lny + (1-\hat{y})\ln(1-y)\right\| \tag{9}$$

where $y$ is the predicted label, and $\hat{y}$ is the real label in the source domain.

The batch gradient descent optimization is used to minimize the loss function. The parameters of the CNN architecture are updated with the supervised back propagation algorithm. Let the $\boldsymbol{U_1}$ as the parameter matrices of weight, $\boldsymbol{U_2}$ as the parameter matrices of bias. The back propagation procedures are described as follows.

1) Initial $\boldsymbol{U_1}$and $\boldsymbol{U_2}$with random values.
2) Update $\boldsymbol{U_1}$by minimizing $Loss$ in Eq. (8) with mini-batched gradient descent.

$$\begin{aligned} U_1^{t+1} &= U_1^t - \alpha(\frac{\partial Loss}{\partial U_1}) \\ &= U_1^t - \alpha(\frac{\partial L_C}{\partial U_1} + \gamma\frac{\partial L_{MMD}}{\partial U_1}) \end{aligned} \tag{10}$$

3) Update $\boldsymbol{U_2}$by only minimizing $L_C$ in Eq. (9) with mini-batched gradient Descent.

$$U_2^{t+1} = U_2^t - \alpha(\frac{\partial L_C}{\partial U_2}) \tag{11}$$

where $\alpha$ is the learning rate.

4) Repeat the step 2) and 3) until the epoch time reaches $N$.

Note that, the training experiments are implemented using Pytorch deep learning toolbox with NVIDIA Titan X GPU acceleration. In this paper, the learning rate is set to 0.02, the Gaussian kernel bandwidth $\sigma$ is set to $10^3$, the epoch time is 1000.

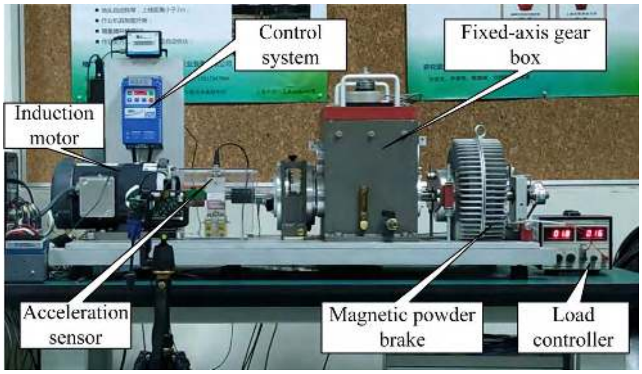


**FIGURE 4. The test rig of drivetrain diagnostic simulator.**

## IV. VERIFICATION TESTS

The proposal aims to transfer the learned knowledges from ideal working conditions to the real-world practice to identify the health states of motors, by adapting the feature spaces learned from labeled data under invariant working conditions to those from the unlabeled data under constantly changing working conditions. In order to acquire the corresponding motor faulty data and experimentally verify the effectiveness of the proposed framework, tests of six three-phase induction motors with different fault types under three different operating modes were carried out in the Drivetrain Diagnostic Simulator (DDS) platform, which is shown in Fig. 4, this system is mainly composed of 3 components: 1) a replaceable three-phase induction motor as the power mechanism, 2) a two-stage fixed-axis gearbox and a two-stage planetary gearbox as the transmission mechanism, 3) a magnetic brake as the load controller. This platform meets all the requirements for vibration analysts and provides a reliable test environment for faulty motor diagnosis.

Six motors of various healthy states (1 healthy and 5 faulty) were used to obtain the faulty samples. Data from the healthy motor is regarded as a benchmark for comparison with the test data from other faulty motors. The five faulty types include built-in bowed rotor (BBR), faulted bearings (FB), broken rotor bars (BRB), built-in rotor misalignment (BRM),

**TABLE 3.** Descriptions of those datasets.

| Dataset name | Number of samples | Controller current (A) | Load (N·m) | Motor frequency |
|---|---|---|---|---|
| A1 | 950×6 | 0 | 0 | 15Hz |
| A2 | 950×6 | 0 | 0 | 30Hz |
| A3 | 950×6 | 0 | 0 | 45Hz |
| N1 | 2800×6 | 0 | 0 | NEDC |
| B1 | 950×6 | 1.42 | 80 | 15Hz |
| B2 | 950×6 | 1.42 | 80 | 30Hz |
| B3 | 950×6 | 1.42 | 80 | 45Hz |
| N2 | 2800×6 | 1.42 | 80 | NEDC |

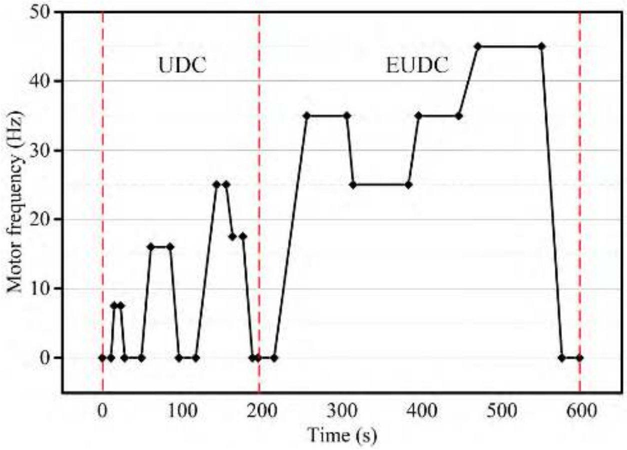


**FIGURE 5.** The modified NEDC test in dataset N1 and dataset N2.

built-in rotor unbalance (BRU). The faulty types and the corresponding causes are listed in Table 2. Three different operating modes are respectively applied on the six motors to gain eight datasets named Dataset A1-A3, Dataset B1-B3, Dataset N1-N2. The details of those datasets are described in Table. 3.

NEDC, New European Driving Cycle [28], was first designed as the driving rules for the passenger vehicles to assess the emission levels of vehicle engines and fuel economy. Now it is widely used as the driving rules for the electric cars in the tests of battery loss, motor performance, transmission efficiency, etc. [29]. It is composed of 4 ECE-15 urban driving cycles (UDC) and one Urban driving cycle (EUDC). For Dataset N1 and N2, a modified NEDC containing one UDC and one EUDC was adopted as the motor running rules to simulate the constantly working conditions of real-world scenario, which is described in Fig. 5. It aims to acquire the data as close as possible to the real-world practice as the target data.

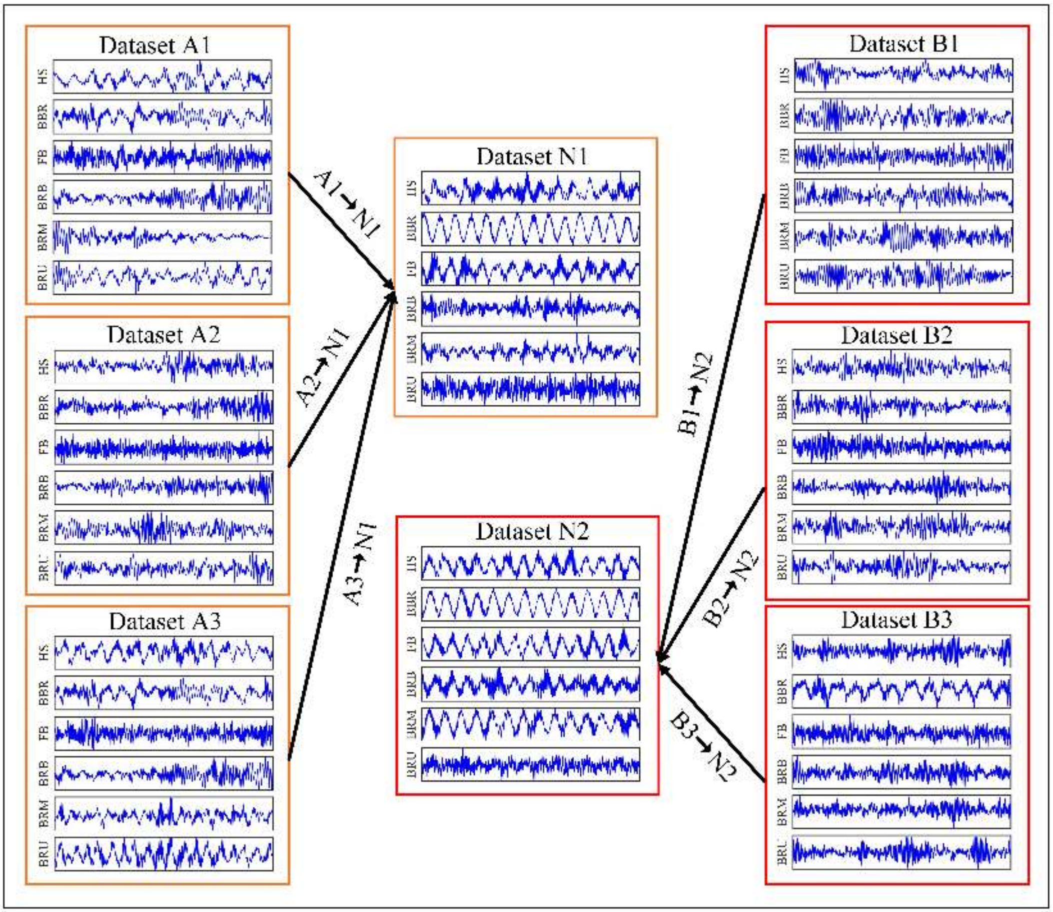
The acceleration sensor BW-BJ14530 was used in the tests and installed at the shell of motors to gain the acceleration signals in radial direction, where the vibrations are strongest. The vibration signals were collected by the data acquisition card named NI-9234 with sampling frequency of 5.12 KHz. The raw signals were equally segmented into several non-overlapping pieces and every 0.2 second signals which contain 1024 data points represent a sample. As the motor frequency is 15Hz, 30Hz, and 45Hz, one sample contains several complete cycle of motor periodic rotation signals. Each faulty motor test last about 3 mins to collect about $10^6$ data points under one unvarying working condition, and thus dataset A1-B3 each contains 950 samples of each faulty type. The NEDC test last about 5mins to collect around $2.8\times10^6$ data points, and thus dataset N1-N2 each contains 2800 samples of each faulty type. In this paper, it is assumed that the number of the source samples and target samples in the input layer should be the same. Hence, 950 samples of each faulty type were randomly selected from Dataset N1-N2 for each transfer task. Note that, there existed some pauses in the NEDC tests, and thus the zero sequences in Dataset N1-N2 should be eliminated before. Fig. 6 respectively displays a set of time waveforms of some samples in the eight datasets. It can be seen that the distributions and amplitudes vary a lot among different datasets. To verify our proposal, Dataset A1-B3 is respectively set as the source domain, and correspondingly dataset N1-N2 is respectively set as the target domain. Those transfer tasks are illustrated in Fig. 6.

## V. THE RESULTS AND DISCUSSION

### A. RESULTS OF THE PROPOSED FRAMEWORK

As shown in Fig. 6, six transfer tasks A$i$ $\rightarrow$N1 and B$i$ $\rightarrow$N2 are conducted according to the proposed framework. To guarantee the validity of the results, each task repeats 5 times and the corresponding results are displayed in Table. 4. In this table, $Acc_1$ denotes the testing accuracy of Dataset N$i$ in the CNN model independently trained by Dataset A$i$ or B$i$. It is used for comparison as the results without transfer learning. The proposed CNN architecture can achieve more than 99% training classification accuracy when the training and testing data are from same dataset in Dataset A1-B3. $Acc_2$ denotes the testing accuracy of dataset N$i$ in the transfer task. To visualize the results, the classification accuracies of all transfer tasks are displayed in Fig. 7. "with transfer learning" denotes the proposed framework. "without transfer learning" means the CNN model is independently built by the source data. It can be seen in Table. 4 and Fig. 7 that the classification accuracies for the unlabeled target data are elevated evidently with transfer learning. It proves the effectiveness of the proposed transfer learning approach in multi-level feature adaptation between two different domains.

Fig. 8 uses the confusion matrix to illustrate the specific classification results of every faulty type. It can be seen that FB is the most detectable faulty type, and the samples of BRM and BRU are more likely to be misclassified. Fig. 9 visualizes the training process in transfer task B3→N2. It can be seen that significant convergence trend occurs in the accuracy curves of both source and target datasets. The accuracy curves ascend slowly in the first 200 epochs and then tends to fluctuate in a small range. The final accuracy of the source data reaches nearly 100%, while the target data less than 80%.



| No. | Transfer tasks<br>(*i*=1,2,3) | Training data | Test data |
|---|---|---|---|
| 1-3 | Dataset<br>A*i*→N1 | Labeled Dataset A*i*: 950*6 samples<br>+ Unlabeled Dataset N1: 950*6 samples | Labeled Dataset N1:<br>950*6 samples |
| 4-6 | Dataset<br>B*i*→N2 | Labeled Dataset B*i*: 950*6 samples<br>+ Unlabeled Dataset N2: 950*6 samples | Labeled Dataset N2:<br>950*6 samples |

**FIGURE 6.** **The waveforms of the raw signals in the 8 datasets and the 6 transfer fault diagnosis tasks.**

**TABLE 4.** **Classification accuracy and improvement.**

| Transfer task | *Acc* 1[a] | *Acc*2[b] | Improvement |
|---|---|---|---|
| A1→N1 | 55.1 ± 2.1 | 59.9 ± 3.4 | + 4.8 |
| A2→N1 | 59.8 ± 2.7 | 65.4 ± 2.3 | + 5.6 |
| A3→N1 | 59.5 ± 1.7 | 69.3 ± 3.6 | + 9.8 |
| B1→N2 | 62.4 ± 0.9 | 68.7 ± 5.8 | + 6.3 |
| B2→N2 | 68.6 ± 0.8 | 75.3 ± 2.5 | + 6.7 |
| B3→N2 | 65.1 ± 1.3 | 75.6 ± 2.2 | + 10.5 |

[a] The testing accuracy of dataset N*i* in the CNN model independently trained by dataset A*i* or B*i*.
[b] The testing accuracy of dataset N*i* in the transfer task A*i* or B*i*→N1.

Besides, there are much bigger fluctuations in the accuracy curves of the target data compared with the source data, especially at the ascending stage.

In order to obtain the best performance and identify the contributions of different-layer MMDs, different weight rates $\theta$ of multi-layer MMDs are used to make a comparison.

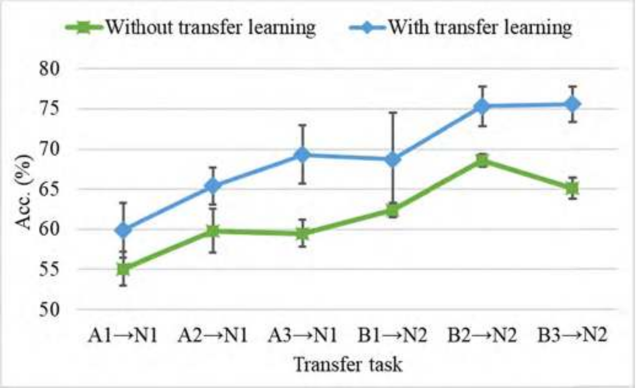


**FIGURE 7.** The testing accuracies of different transfer tasks.

The testing accuracies with different $\theta = \{0.5, 0.75, 1, 1.5, 2\}$ of six transfer tasks are visualized in Fig. 10. It can be observed that it achieves the best accuracy when the weight rate $\theta$ is selected as 2, although the accuracy disparity is little. It illustrates that the contributions of multi-layer MMDs increase as the layer goes deeper and the level of features gets higher. Besides the weight rate, several other key parameters in the training process need to be determined, such as batch size and tradeoff factor $\gamma$. The quantification of these parameters is mainly based on a comparative evaluation of the performances of various optional values. The accuracies at different batch size and tradeoff parameters are shown in Fig. 11. It can be observed that as the batch size increases from 16 to 256 and $\gamma$ increases from 0.25 to 1, the accuracy tends to get lower from more than 70% to 55%. Consequently, the optimal batch size is selected as 16 and the tradeoff parameter $\gamma$ is 0.25.

### B. COMPARISON WITH OTHER METHODS

To demonstrate the superiority of the proposed transfer learning approach, several other state-of-art transfer learning methods as follows are employed for comparison. 1) and 2) belong to the traditional unsupervised transfer learning framework using shallow manual features. 3)-6) belong to the deep transfer learning frameworks with various learners such as MLP, CNN, etc.

1) TCA [30]: Transfer Component Analysis for feature domain adaptation;
2) JDA [31]: Joint Distribution Adaptation for feature domain adaptation;
3) DaNN [13]: Multi-layer Neural Networks with MMD regularization term;
4) TICNN [32]: Convolution Neural Networks with kernel dropout interference;
5) DCTLN [20]: convolution Neural Networks with MMD used in fully-connected layers;
6) FTNN [21]: convolution Neural Networks with MMD and pseudo label learning.

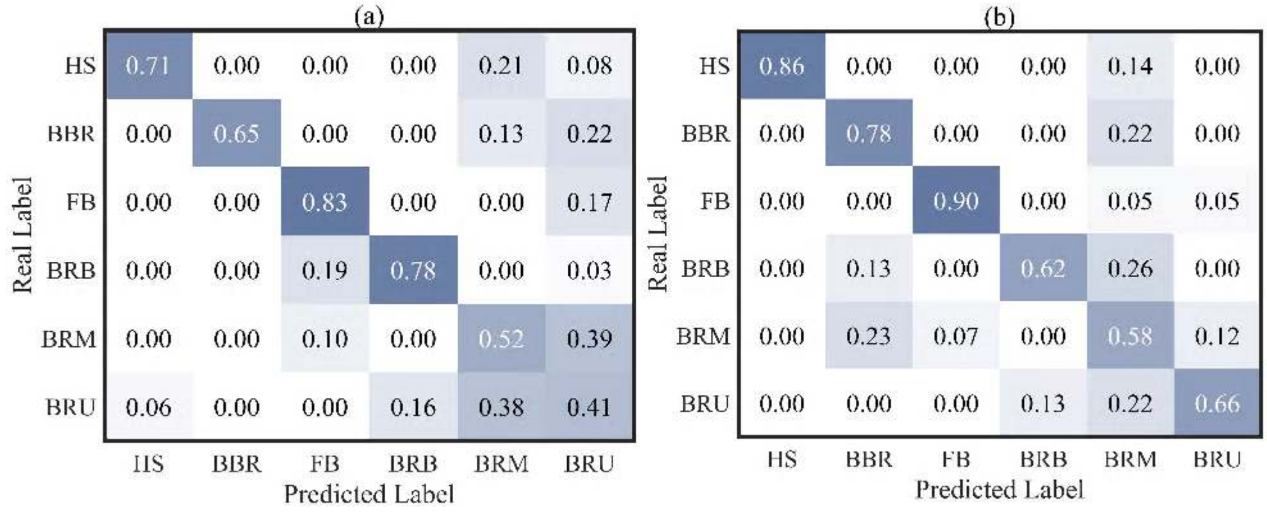
In 1) and 2), TCA and JDA are two classic transfer learning methods. TCA is used to learn some transfer components to reproduce Hilbert space with MMD. In the subspace spanned by these transfer components, data distributions in different domains are similar and data properties are preserved. Once the subspace is found, we train a KNN for subsequent classification with the source domain, and obtain the accuracy of the target domain. JDA, as a modification of TCA, can reduce the difference between domains by jointly adapting both the marginal distribution and conditional distribution in a procedure of dimensionality reduction. It can construct a new feature subspace that is robust and effective for substantially different distribution. The training and test procedures are same with TCA. The two methods, both incapable of addressing sequential data, depend on manual feature engineering, thus 29 statistic features [33], including mean, root mean square, kurtosis, variance, skewness, etc., are extracted as the two domains. The kernel type is both selected as*rbf* and the dimension after adaptation is set to 16. In 3) MLP with layers sizes $\{512 - 256 - 128 - 64 - 128 - 64 - 6\}$ is adopted. MMD of the feature spaces after 6th layer is used as the regularization term. In 4), 0.5 kernel dropout is added to the first convolution layer as the interference term. In 4) and 5), the iterative pseudo labels are tagged on the target samples every epoch the CNN model is built The classification loss between the predicted target labels and the pseudo labels is used as a new regularization term in back propagation.

The comparison results of the testing accuracy are visualized in Fig. 12. Contrasted with the other methods, the proposed method attains to the highest accuracies in the six transfer tasks. More specifically, it respectively yields 12.7%, 10.5%, 4.6%, 17.2%, 1.8%, and 4.7% average performance improvement compared with TCA, JDA, DaNN, TICNN, DCTLN, and FTNN.

### C. DISCUSSION

From Table. 4 it can be seen that 1) the classification accuracies for the unlabeled target data are elevated evidently with transfer learning. It proves the effectiveness of the proposed transfer learning approach in bridging the distribution gap between two different domains; 2) when batch size is determined to 16, the accuracy improvement gets the highest. It can be explained by the gradient descent mechanism. Too large batch size will result in a big cumulative descent for the parameter update, especially when MMD loss is incorporated into to the optimization process.

Fig. 10 illustrates the accuracy gap does exist when different weight rates are selected. Generally, the tendency is that the larger the weights of the last few layers are, the better the performance is. It makes sense to enlarge the MMD weights of the high-level features and lower those of the shallow-level features. It can be interpreted by the intrinsic property of deep learning. Previous studies have proved that for a trained deep learning model for image classification, the use of high-level features is more in tune with how objects are classified in real world, and low-level features are mostly concerned with dealing with pixel intensities or colors, finding lines or edges, and investigating corresponding points between images [34]. In general, the low-level representations are general features,



(a)

| Real Label | HS | BBR | FB | BRB | BRM | BRU |
|---|---|---|---|---|---|---|
| HS | 0.71 | 0.00 | 0.00 | 0.00 | 0.21 | 0.08 |
| BBR | 0.00 | 0.65 | 0.00 | 0.00 | 0.13 | 0.22 |
| FB | 0.00 | 0.00 | 0.83 | 0.00 | 0.00 | 0.17 |
| BRB | 0.00 | 0.00 | 0.19 | 0.78 | 0.00 | 0.03 |
| BRM | 0.00 | 0.00 | 0.10 | 0.00 | 0.52 | 0.39 |
| BRU | 0.06 | 0.00 | 0.00 | 0.16 | 0.38 | 0.41 |

Predicted Label

(b)

| Real Label | HS | BBR | FB | BRB | BRM | BRU |
|---|---|---|---|---|---|---|
| HS | 0.86 | 0.00 | 0.00 | 0.00 | 0.14 | 0.00 |
| BBR | 0.00 | 0.78 | 0.00 | 0.00 | 0.22 | 0.00 |
| FB | 0.00 | 0.00 | 0.90 | 0.00 | 0.05 | 0.05 |
| BRB | 0.00 | 0.13 | 0.00 | 0.62 | 0.26 | 0.00 |
| BRM | 0.00 | 0.23 | 0.07 | 0.00 | 0.58 | 0.12 |
| BRU | 0.00 | 0.00 | 0.00 | 0.13 | 0.22 | 0.66 |

Predicted Label

**FIGURE 8. The confusion matrix for the transfer results of each faulty type: (a) the average accuracy in transfer task *Ai* →N1, (b) the average accuracy in transfer task B*i* →N2.**

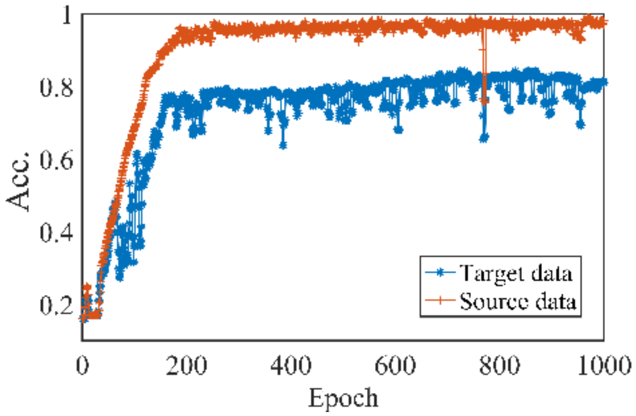
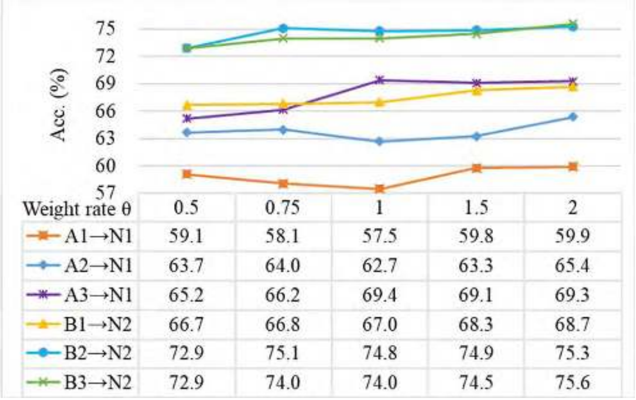


**FIGURE 9. The classification accuracy trends of the source and target data during 1000 epochs training in the transfer task B3→N2.**



| Weight rate θ | 0.5 | 0.75 | 1 | 1.5 | 2 |
|---|---|---|---|---|---|
| A1→N1 | 59.1 | 58.1 | 57.5 | 59.8 | 59.9 |
| A2→N1 | 63.7 | 64.0 | 62.7 | 63.3 | 65.4 |
| A3→N1 | 65.2 | 66.2 | 69.4 | 69.1 | 69.3 |
| B1→N2 | 66.7 | 66.8 | 67.0 | 68.3 | 68.7 |
| B2→N2 | 72.9 | 75.1 | 74.8 | 74.9 | 75.3 |
| B3→N2 | 72.9 | 74.0 | 74.0 | 74.5 | 75.6 |

**FIGURE 10. The testing accuracies of six transfer tasks with different weight rate $\theta$ from 0.5 to 2.**

and high-level representations are specific features. Therefore, the feature domain adaptation in high-level features is of more importance and should be endowed with higher weight.

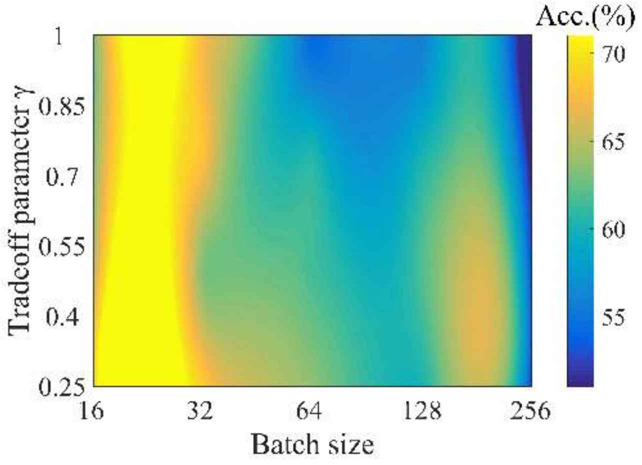
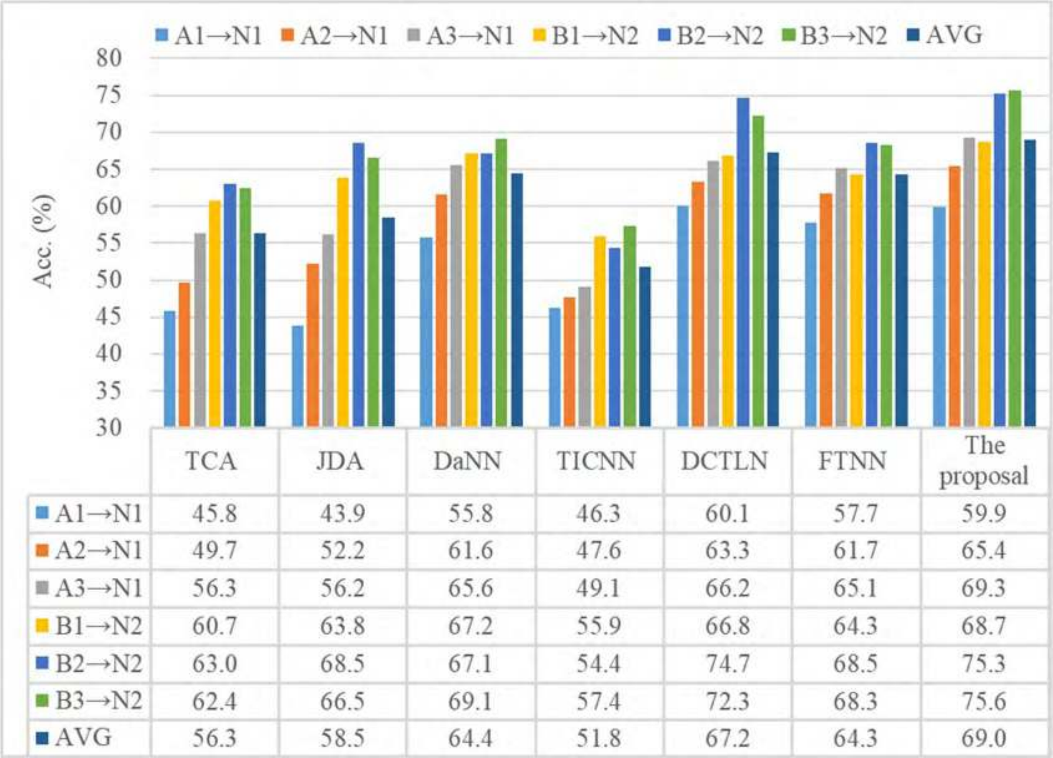


**FIGURE 11. The test accuracies at different batch size and tradeoff parameters.**

Fig. 12. illustrates the proposed method attains to the highest accuracies in both transfer tasks. More specifically, 1) the performance of TICNN is worst. The reason may be that it is the only comparative method that doesn't use any idea about transfer learning. The only use of interference terms of kernel dropout and batch size in a CNN is not able to train a robust enough model for the unlabeled target data. 2) TCA and JDA performs only slightly better than TICNN. It can be explained by fact that feature learning of the two methods heavily depends on manual feature engineering. The deep learning methods, with stronger feature learning capacity, always present a superior performance to the methods that demand manual feature engineering, no matter in traditional diagnosis framework or transfer learning framework [35]. 3) DaNN, DCTLN, and FTNN performs slightly worse than the proposal. The reason of the good performances may be the used similar transfer learning framework with deep learning architecture and the regularization term of MMD.



| | TCA | JDA | DaNN | TICNN | DCTLN | FTNN | The proposal |
|---|---|---|---|---|---|---|---|
| A1→N1 | 45.8 | 43.9 | 55.8 | 46.3 | 60.1 | 57.7 | 59.9 |
| A2→N1 | 49.7 | 52.2 | 61.6 | 47.6 | 63.3 | 61.7 | 65.4 |
| A3→N1 | 56.3 | 56.2 | 65.6 | 49.1 | 66.2 | 65.1 | 69.3 |
| B1→N2 | 60.7 | 63.8 | 67.2 | 55.9 | 66.8 | 64.3 | 68.7 |
| B2→N2 | 63.0 | 68.5 | 67.1 | 54.4 | 74.7 | 68.5 | 75.3 |
| B3→N2 | 62.4 | 66.5 | 69.1 | 57.4 | 72.3 | 68.3 | 75.6 |
| AVG | 56.3 | 58.5 | 64.4 | 51.8 | 67.2 | 64.3 | 69.0 |

**FIGURE 12.** The performances of all comparative methods in six transfer tasks. The proposal achieves the highest diagnostic accuracy.

The accuracy disparity between DaNN and the proposal can be explained by the fact that CNN can extract more meaningful representations based on the more complicated structure compared with DaNN. The accuracy disparity between DCTLN or FTNN and the proposal can be interpreted by the use of pseudo label learning. It can be seen that the test accuracies are generally lower than 80%, which is not high enough, and thus the use of pseudo labels may lead to the phenomenon of double misclassification and cause worse performance.

## VI. CONCLUSION

As the high difficulty in the collection of high quality data in real scenario, it remains as a challenge in the real-world motor fault diagnosis applications to improve the diagnosis accuracy for the unlabeled data under constantly changing working conditions, with the help of the labeled data under invariant working conditions. To address those issues, in this paper, a novel domain adaptive motor fault diagnosis framework using deep transfer learning is proposed. It possess the capability to transfer the knowledges learned from labeled data under invariant working conditions to the unlabeled data under constantly changing working conditions. The core of the proposal is to bridge the gap between the multi-level representations of the source domain and the target domain using the regularization term of MMD.

The proposal is verified by the data from motor tests under NEDC condition and invariant working conditions. The obvious classification accuracy improvement for the target domain in two transfer tasks demonstrates its effectiveness in bridging the discrepancy between different domains. A comparative evaluation of performances with various MMD weights distinguishes the importance or contribution of the low-level features and high-level features. The comparisons with other popular transfer learning or domain adaptive methods prove the proposed transfer learning framework can achieve the highest diagnosis accuracy.

There still exist some shortcomings in the proposal, such as the long training time and the too complex architecture. In future research, our effort will be devoted to two aspects. 1) make attempt to design new transfer learning frameworks with higher diagnosis accuracy. 2) Collect the faulty motor data from the real-world vehicle tests to realize motor fault diagnosis in real practice.

## REFERENCES

[1] M. El H. Benbouzid, “A review of induction motors signature analysis as a medium for faults detection,” *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 984–993, Oct. 2000.

[2] S. Yin, X. Li, H. Gao, and O. Kaynak, “Data-based techniques focused on modern industry: An overview,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 657–667, Jan. 2015.

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

[4] C. Sun, M. Ma, Z. B. Zhao, and X. Chen, “Sparse deep stacking network for fault diagnosis of motor,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3261–3270, Jul. 2018.

[5] D. Xiao, Y. Huang, C. Qin, H. Shi, and Y. Li, "Fault diagnosis of induction motors using recurrence quantification analysis and LSTM with weighted BN," *Shock Vib.*, vol. 2019, pp. 1–14, 2019.

[6] L. Guo, Y. Lei, N. Li, T. Yan, and N. Li, "Machinery health indicator construction based on convolutional neural networks considering trend burr," *Neurocomputing*, vol. 292, no. 1, pp. 142–150, 2018.

[7] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[8] Y. Wei, Y. Zhang, and Q. Yang, "Learning to transfer," Aug. 2017, *arXiv:1708.05629*. [Online]. Available: https://arxiv.org/abs/1708.05629

[9] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 1, p. 9, Dec. 2016.

[10] R. Zhang, H. Lee, and D. Radev "Dependency sensitive convolutional neural networks for modeling sentences and documents," 2016, *arXiv:1611.02361*. [Online]. Available: https://arxiv.org/abs/1611.02361

[11] R. Johnson and T. Zhang, "Semi-supervised convolutional neural networks for text categorization via region embedding," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2015, pp. 919–927.

[12] G.-S. Xie, X.-Y. Zhang, S. Yan, and C.-L. Liu, "Hybrid CNN and dictionary-based models for scene recognition and domain adaptation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 6, pp. 1263–1274, Jun. 2017.

[13] M. Ghifary, W. B. Kleijn, and M. Zhang, "Domain adaptive neural networks for object recognition," Sep. 2014, *arXiv:1409.6041*. [Online]. Available: https://arxiv.org/abs/1409.6041

[14] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.

[15] X. Li, W. Mao, and W. Jiang, "Extreme learning machine based transfer learning for data classification," *Neurocomputing*, vol. 174, no. 1, pp. 203–210, 2016.

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

[17] 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.

[18] S. Shao, S. McAleer, R. Yan, and P. Baldi, "Highly accurate machine fault diagnosis using deep transfer learning," *IEEE Trans. Ind. Inform.*, vol. 15, no. 4, pp. 2446–2455, Apr. 2019. doi: 10.1109/TII.2018.2864759.

[19] 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.

[20] L. Guo, Y. Lei, S. Xing, T. Yan, and N. Li, "Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data," *IEEE Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7316–7325, Sep. 2018.

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

[22] K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf, "Kernel measures of conditional dependence," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, Canada, 2008, pp. 489–496,

[23] R. Liu, G. Meng, B. Yang, C. Sun, and X. Chen, "Dislocated time series convolutional neural architecture: An intelligent fault diagnosis approach for electric machine," *IEEE Trans. Ind. Inf.*, vol. 13, no. 3, pp. 1310–1320, Jun. 2017.

[24] M. Zhao, M. Kang, B. Tang, and M. Pecht, "Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes," *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4290–4300, May 2018.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," Mar. 2015, *arXiv:1502.03167*. [Online]. Available: https://arxiv.org/abs/1502.03167

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

[27] P. Zhou and J. Austin, "Learning criteria for training neural network classifiers," *Neural Comput. Appl.*, vol. 7, no. 4, pp. 334–342, Dec. 1998.

[28] Parker, Peter, "The concept of NEDC," *RSA J.*, vol. 142, no. 72, 1994.

[29] L. Chen, F. Zhu, M. Zhang, Y. Huo, C. Yin, and H. Peng, "Design and analysis of an electrical variable transmission for a series–parallel hybrid electric vehicle," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2354–2363, Jun. 2011.

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

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

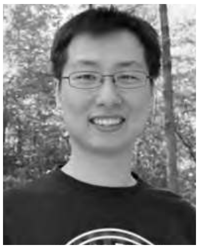
[32] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, "A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals," *Sensors*, vol. 17, no. 2, p. 425, 2017.

[33] Z. Shen, X. Chen, X. Zhang, and Z. He, "A novel intelligent gear fault diagnosis model based on EMD and multi-class TSVM," *Measurement*, vol. 45, no. 1, pp. 30–40, Jan. 2012.

[34] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," Nov. 2015, *arXiv:1511.06434*. [Online]. Available: https://arxiv.org/abs/1511.06434

[35] T. Han, C. Liu, W. Yang, and D. Jiang, "Deep transfer network with joint distribution adaptation: A new intelligent fault diagnosis framework for industry application," Apr. 2018, *arXiv:1804.0726*. [Online]. Available: https://arxiv.org/abs/1804.07265

**DENGYU XIAO** received the B.S. degree in mechanical engineering from the Harbin Institute of Technology, in 2015. He is currently pursuing the Ph.D. degree with the School of Mechanical Engineering, Shanghai Jiao Tong University. His research interests include industrial big data analysis, machinery health assessment and fault diagnosis, signal processing, and data-driven modeling.



**YIXIANG HUANG** received the Ph.D. degree in mechanical engineering from Shanghai Jiao Tong University, in 2010, where he is currently an Assistant Researcher. His main research interests include intelligent monitoring and maintenance of mechanical equipment, and industrial big data analysis.

**LUJIE ZHAO** received the B.S. degree in mechanical engineering from Shanghai Jiao Tong University, in 2017, where he is currently pursuing the M.S. degree with the School of Mechanical Engineering. His research interests include industrial big data analysis, machinery health assessment and fault diagnosis, and data-driven modeling.

**CHENGJIN QIN** received the B.Eng. degree in mechanical engineering from Southwest Jiao-tong University, Chengdu, China, in 2012, and the Ph.D. degree in mechanical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2018, where he holds a postdoctoral posi-tion in mechanical engineering. His research interests include machining stability analysis, robotic machining, machine condition monitoring, PHM, artificial intelligence, and time–frequency analysis.

**HAOTIAN SHI** received the B.S. degree in mechanical engineering from Shandong University, in 2015. He is currently pursuing the Ph.D. degree with the School of Mechanical Engineering, Shanghai Jiao Tong University. His research interests include data-driven modeling, big data analysis, intelligent monitoring, and maintenance of heart.

**CHENGLIANG LIU** received the Ph.D. degree in mechanical engineering from Southeast University, in 1999. He is currently a Professor with the School of Mechanical Engineering, Shanghai Jiao Tong University. His main research interests include intelligent monitoring and maintenance of mechanical equipment, and industrial big data analysis.