

Article citation info:

Hei Z, Sun B, Wang G, Lou Y, Zhou Y. Multi-feature spatial distribution alignment enhanced domain adaptive method for tool condition monitoring, *Eksploracja i Niezawodność – Maintenance and Reliability* 2023; 25(4) <http://doi.org/10.17531/ein/171750>

## Multi-feature spatial distribution alignment enhanced domain adaptive method for tool condition monitoring

Indexed by:



Zhendong Hei<sup>a</sup>, Bintao Sun<sup>a</sup>, Gaonghai Wang<sup>b</sup>, Yongjian Lou<sup>b,\*</sup>, Yuqing Zhou<sup>a,b</sup>

<sup>a</sup> College of mechanical and electrical engineering, Wenzhou University, China

<sup>b</sup> College of mechanical and electrical engineering, Jiaying Nanhu University, China

### Highlights

- A domain adaptive method for aligning multi feature spatial distributions is proposed.
- A ResNet18\_BiLSTM feature extraction model is proposed to reduce signal fluctuations.
- A soft threshold technique based on attention mechanism is proposed for informativeness.

### Abstract

Transfer learning (TL) has been successfully implemented in tool condition monitoring (TCM) to address the lack of labeled data in real industrial scenarios. In current TL models, the domain offset in the joint distribution of input feature and output label still exists after the feature distribution of the two domains is aligned, resulting in performance degradation. A multiple feature spatial distribution alignment (MSDA) method is proposed, including Correlation alignment for deep domain adaptation (Deep CORAL) and Joint maximum mean difference (JMMD). Deep CORAL is employed to learn nonlinear transformations, align source and target domains at the feature level through the second-order statistical correlations. JMMD is applied to improve domain alignment by aligning the joint distribution of input features and output labels. ResNet18 combining with bidirectional short-term memory network and attention mechanism is developed to extract the invariant features. TCM experiments with four transfer tasks were conducted and demonstrated the effectiveness of the proposed method.

### Keywords

condition monitoring, Transfer learning, correlation alignment, joint maximum mean difference, feature extractor

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

### 1. Introduction

Recent years, with the continual development of the machining process, the machining complexity and accuracy of products have been greatly improved, and the condition of tools during processing directly affects the surface quality of products processed. In order to obtain high precision machining products, it is necessary to establish an effective tool condition monitoring (TCM) system [6,18,41]. Generally, a tool's condition is divided into three periods: grinding, steady, and failure. When a new tool starts to be used, it needs to go through a short break-in

phase between the tool and the machined workpiece firstly, followed by a slow increase in wear and a long period of steady wear. Failure is the final sharp wear stage of the tool until the end of its useful life [22]. As the deterioration of the tool wear increases, the surface quality of the workpiece decreases. Therefore, a great deal of studies have been conducted by many researchers on TCM in order to achieve high quality machined products [5,8,35,43]. The results show that 10% to 40% of the process downtime is caused by tool fault, which often results in

(\*) Corresponding author.  
E-mail addresses:

Z. Hei (ORCID: 0009-0002-4591-4968) 21461440022@stu.wzu.edu.cn, B. Sun (ORCID: 0000-0003-2688-6949) 381861064@qq.com, G. Wang (ORCID: 0000-0003-1581-6267) wang\_gonghai@jxnhu.edu.cn, Y. Lou 220043@jxnhu.edu.cn, Y. Zhou (ORCID: 0000-0002-8580-5427) zhoyuq@jxnhu.edu.cn

50% to 80% of the effective tool life being used [24,39]. Therefore, an effective TCM method is of great importance to improve productivity, surface quality of machined products and cost savings [20].

Tool condition is difficult to describe using precise mathematical models because it is nonlinear, time-varying and continuous in actual industrial scenarios. Since the 1980s, TCM has been extensively studied [1], and many effective models have been proposed, including statistics, physical, data-driven, and hybrid models [16]. Data-driven models have been shown significant benefits in dealing with monitoring the tool condition due to the independence of the complex physical model and the systematic a priori knowledge [11,40]. It can effectively extract wear feature information from time or frequency domain signals of tools without the need for empirical knowledge [28,36]. Guan et al. proposed a method based on Hilbert edge spectrum to analyze the wear signals for effective feature extraction and achieve accurate classification of tool wear conditions [11]. Yan et al. used ResNet18 network to fuse the collected signals in multiple channels, which effectively improved accuracy in tool wear monitoring [36]. Nawrocki et al. utilized vibroacoustic signals obtained from spindle bearings in mass production machines in the automotive industry to diagnose the spindle and detect wear symptoms [28]. Jamshidi et al. employed machine tool spindle current and multi-scale analysis for tool condition monitoring [17]. Kasim et al. proposed the Z-rotation method to calculate the milling tool wear progress index based on variance across signal components [19]. Rizal et al. developed an embedded multi-sensor system on a rotary dynamometer for real-time condition monitoring of milling tools [29]. Data-driven based condition monitoring methods require a large number of labeled training sets to learn the model [4,23], however, in actual machining process, machines are usually operating under different working conditions, and it is challenging to collect enough labeled samples for model training under each working condition [25].

Transfer learning (TL) tries to resolve this issue. Li et al. proposed an adaptive partial domain approach for implementing smart fault diagnosis [15,27,34]. Chen et al. proposed a method to calibrate data labels using a TL algorithm, which makes TL play a significant role for fault diagnostics of wind turbines [4]. Long et al. learned transfer network by aligning the joint

distribution of multiple domain specific layers across domains (JMMD) to make the source and target domain distributions more distinguishable [26]. Marei et al. proposed a TL based Convolutional Neural Network (CNN) for TCM [27]. Long et al. proposed deep adaptation networks (DAN) that apply multiple linear kernels to multiple layers of a neural network to minimize the MMD [26]. Ross et al. applied transfer learning to tool condition monitoring in sustainable machining of nickel alloy under variable working conditions [30]. Gudelek et al. introduced a long-short-term memory depth multilayer perceptron method based on wavelets for tool state monitoring, considering operational variability [12]. Unver et al. devised a novel transfer learning framework by combining analytical solutions with CNN and performed tool milling experiments on CNC vertical milling machines to validate the effectiveness of the approach [37]. These above literature have partially revealed that in the feature extraction process of deep learning, deep features can be reduced from cross domain distribution differences, but cannot be eliminated [13,31,33,42]. The deep features of CNN ultimately transition from general features to specific features along the network's layer, and the transferability of features and classifiers decreases with the increase of inter domain differences, although the source and target domain features are backward aligned in a specific feature space, However, deep networks with multiple layers of feature extraction still exhibit changes in the joint distribution of features and labels in the activation layer of higher-level networks.

For data-driven methods, TCM is a time series problem that captures the nonlinear mapping relationship between the time series of previous and future tool conditions during the machining process [38]. Although recurrent neural network (RNN) can retain the input short-term memory and establish the mapping relationship between short-term memory and target vector [3], it cannot solve the long-term prediction problem. Long Short-Term Memory (LSTM) and Gate Recursive Unit Network (GRU), as variants of RNN, can capture long-term dependencies between cutting force signals and tool states [9,21]. However, due to the complexity of the model, the difficulty of training increases sharply with the increase of layers, and the demand for training data increases [10]. Moreover, performance is also affected by the number of hidden

layers and units.

To solve the above problem, this paper proposes a novel multi-feature spatial distribution alignment (MSDA) network for TCM under variable working conditions, taking advantage of TL to reduce the requirement for feature distribution consistency between the training set and the test set, and reduces the dependence on labeled samples. A structure based on ResNet18 and BiLSTM is constructed for long-term and short-term prediction of tool conditions to improve the feature extraction capability. In order to retain valuable features, the extracted features are subjected to an attention mechanism and soft thresholding to minimize noise-related information. The contributions of this paper are as follows:

- (1) A domain adaptive method for aligning multi-feature spatial distributions is proposed to achieve feature alignment in the source and target domains, considering the joint distribution of features and labels in the deeper levels of the aligned neural network.
- (2) A ResNet18\_BiLSTM feature extraction model is proposed to extract features from spatial and temporal dimensions to reduce the effects of signal fluctuations, in which the gradient disappearance and information loss are avoided through the residual network and preserve the integrity of signal features.
- (3) A soft threshold technique based on attention mechanism is proposed to effectively improve the value of information.

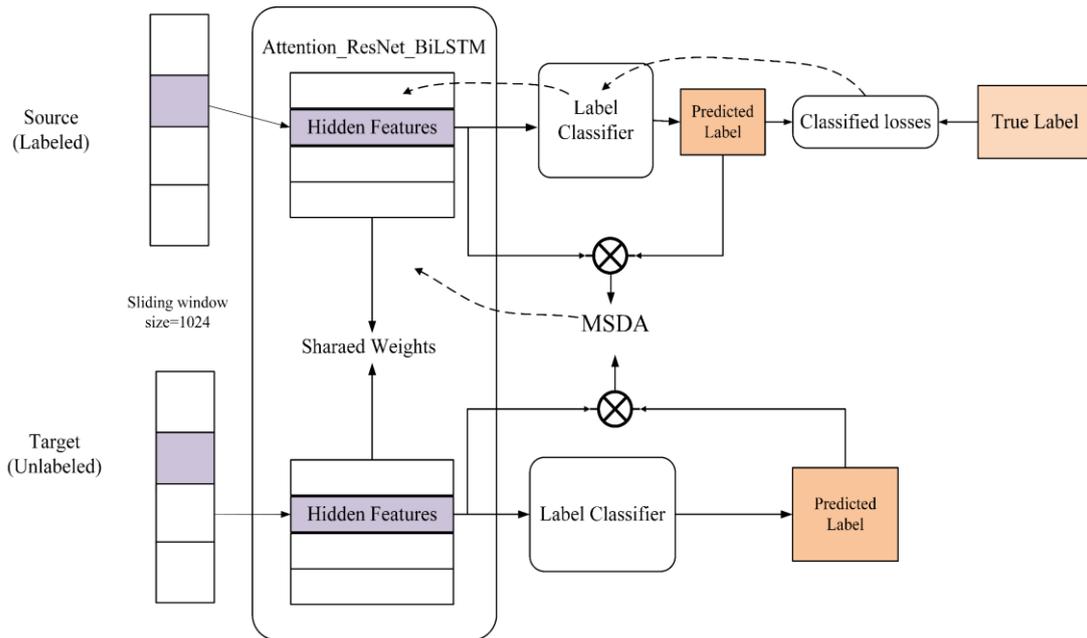


Fig. 1. The overall framework of the methodology proposed in this paper.

## 2. Theoretical background

When the datasets in the source and target domains have different feature distributions, traditional supervised learning algorithms are often unable to achieve effective classification, and domain adaptation is well suited for this situation.

Since the data in the target and source domains obey the probability distributions of  $P$  and  $Q$ , respectively, and for domain adaptation, our goal is to construct a deep neural network that classifies unlabeled data in the target domain through feature learning that is amenable to transfer, as follows:

$$\hat{y} = \beta(x) \quad (1)$$

where  $\beta(\cdot)$  denotes DNN and  $\hat{y}$  is the output of the model prediction; therefore, the purpose of domain adaptation is to minimize the target domain risk  $\varepsilon_t(\beta)$  with supervision of the source data **Błąd! Nie można odnaleźć źródła odwołania..**

$$\varepsilon_t(\beta) = \Pr_{(x,y) \sim Q} [\beta(x) \neq y] \quad (2)$$

We can write the total domain adaptation loss as:

$$\mathcal{L} = \mathcal{L}_c + \lambda \mathcal{L}_{TL} \quad (3)$$

where  $\mathcal{L}_c$  is the maximal cross-entropy loss,  $\lambda$  is the trade-off parameter,  $\mathcal{L}_{TL}$  denotes partial loss to reduce the difference in characteristics between the two domains:

$$\mathcal{L}_c = -\mathbb{E}_{(x_i^s, y_i^s) \in \mathcal{D}_s} \sum_{c=0}^{C-1} \mathbb{1}_{[y_i^s=c]} \log[\beta(x_i^s)] \quad (4)$$

where  $C$  is the count of all possible labels and  $\mathbb{1}$  is the indicator function.

### 3. Proposed method

The model proposed in this paper which uses several recently monitored sequential tool wear values to track the current condition of tool, while adding the proposed MSDA domain adaptive method, which can effectively perform monitoring of the condition of the tool under variable working conditions within the real machining process, which is important for achieving high precision part size machining and avoiding part scrap.

The tool's condition monitoring modules mainly include feature extraction, feature normalization, design of Attention\_ResNet\_BiLSTM deep learning model, design of MSDA domain adaptive network, etc. The flow chart of the tool condition monitoring model designed in this paper is shown in Figure 1.

#### 3.1 Multi-feature spatial distribution alignment (MSDA)

The approach of Deep CORAL is analogous to that of , DAN and Reverse Grad [14] methods, it adds another loss (CORAL loss) with the aim of minimizing the variance of the learned covariance matrix across domains, which is analogous to that of mini-mizing MMD in the case of a polynomial kernel, but it's more potent than DDC (which merely aligns the means of the sample) and more amenable to optimization than DAN, the ReverseGrad algorithm. It is easy to optimize, and its most salient feature is that it can be seamlessly integrated across different deep learning layers or architectures. For example, in the case of MMD and MK-MMD [7], solve the domain adaptation problem by considering the difference between the

edge distributions  $P(X_s)$  and  $Q(X_t)$  of the target and source domains, but do not take into account the joint distribution alignment of  $P(X_s, Y_s)$ ,  $Q(X_t, Y_t)$  of the deeper layers of the neural network. Many papers have also found that the distribution differences can be reduced, but do not disappear, means that the differences in joint distribution still exist in higher levels of the neural network despite the fact that the source and target domain data is fed and passed through a deep network with multi-layer feature extraction.

Our method takes advantage of the advantages of Deep CORAL and JMMD, and offers the MSDA method applicable to monitoring the condition of tools under varying working conditions, the principle of which is shown in Figure 2. Firstly, in order to realize the alignment of the second order statistics and to initially approximate the distribution of the source and target domain features in the feature space is shown, we take advantage of the powerful compatibility of Deep CORAL and insert it into the shallower layer of the neural network in order to learn some sort of non-linear source and target domain transformations in the initial model stage. We insert JMMD into a deeper layer of the neural network, using the feature vectors already aligned by Deep CORAL as the input of JMMD, and use embedding in Hibert space to measure the distance from the joint distribution  $P(X_s, Y_s)$ ,  $Q(X_t, Y_t)$ , but JMMD differs from MMD in that it imposes a uniform JMMD uses non-uniform weights, which captures the interaction between the different variables of the joint distribution  $P(X_s, Y_s)$ ,  $Q(X_t, Y_t)$ , and takes a step closer to the distance between the two domains in the feature space.

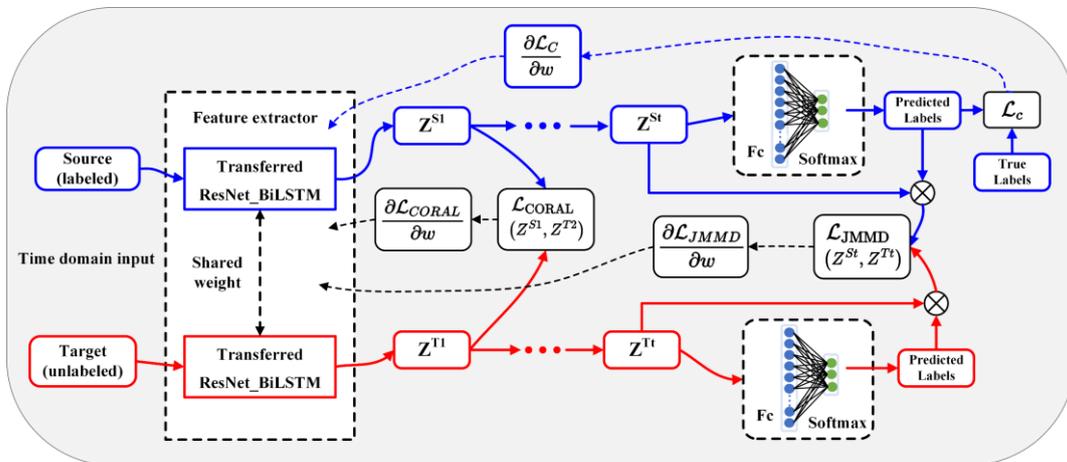


Fig. 2. Process of Multi-feature spatial distribution alignment.

### 3.1.1 Correlation Alignment for Deep Domain Adaptation

Methods like MK-MMD and other domain adaptation approaches often perform feature alignment at the last layer of a neural network, which is more suitable for aligning overall feature distributions. On the other hand, the Deep CORAL method allows for consideration of feature representations at different layers by measuring the CORAL distance at arbitrary layers, enabling multi-layer feature alignment. Therefore, in this case, we choose Deep CORAL to achieve multi-layer feature alignment during the domain adaptation process. Its advantage lies in the fact that deep neural networks can learn feature representations with different semantic information and abstraction levels at different layers. Through multi-layer alignment, Deep CORAL can better capture the relationships between the source and target domains and align features at different abstraction levels. This facilitates the extraction of more diverse and robust feature representations, thereby improving the performance of domain adaptation.

CORAL loss was first proposed by Sun et al. with the aim of align the second order statistics of source domain data and feature distributions of target domain data in order to achieve domain adaptation. Deep CORAL loss is defined as the distance between the second order statistics of the features in the source domain and those in the target domain, and Deep CORAL is expressed by the formula [32]:

$$\mathcal{L}_{\text{CORAL}} = \frac{1}{4d^2} \|C_S - C_T\|_F^2 \quad (5)$$

where  $C_S$  and  $C_T$  denote in terms of the two dataset domains covariance matrices, respectively,  $d$  is the dimension of each

sample,  $\|\cdot\|_F^2$  denotes the Frobenius norm between the covariance matrices.

$$\begin{aligned} C_S &= \frac{1}{n_s-1} \left( X_S^T X_S - \frac{1}{n_s} (\mathbf{1}^T X_S)^T (\mathbf{1}^T X_S) \right) \\ C_t &= \frac{1}{n_t-1} \left( X_t^T X_t - \frac{1}{n_t} (\mathbf{1}^T X_t)^T (\mathbf{1}^T X_t) \right) \end{aligned} \quad (6)$$

where  $\mathbf{1}$  denotes a column vector whose elements are all equal to 1.

### 3.1.2 Joint Maximum Mean Discrepancy

To introduce JMMD, we first introduce the concept of MMD, and many current methods implement domain adaptation by measuring the difference between the edge distributions  $P(X_S)$  and  $Q(X_t)$  of the source and target domains [2]. Maximum Mean difference (MMD), a two-core sample statistic, has been used extensively to measure the distribution of edges between  $P(X_S)$  and  $Q(X_t)$ . However, MMD does not resolve the offset arising from the joint distribution  $P(X_S, Y_S), Q(X_t, Y_t)$  between the input and output, so the Hilbert space embedding is used to measure the distance between the joint distributions  $P(X_S, Y_S)$  and  $Q(X_t, Y_t)$  between the source and target domains. The resulting distance metric becomes the joint maximum mean difference (JMMD) with the following equation:

$$\mathcal{L}_{\text{JMMD}}(P, Q) = \left\| \mathbb{E}_P \left( \bigotimes_{l=1}^{|L|} \phi^l(z^{sl}) \right) - \mathbb{E}_Q \left( \bigotimes_{l=1}^{|L|} \phi^l(z^{tl}) \right) \right\|_{\bigotimes_{l=1}^{|L|} \mathcal{H}^l}^2 \quad (7)$$

where  $L$  is the set of top-level networks,  $|L|$  is the number of layers of the matching set,  $z^{sl}$  denotes the  $l$ -th level activation generated in the source domain, and  $z^{tl}$  the  $l$ -th level activation generated in the target domain.

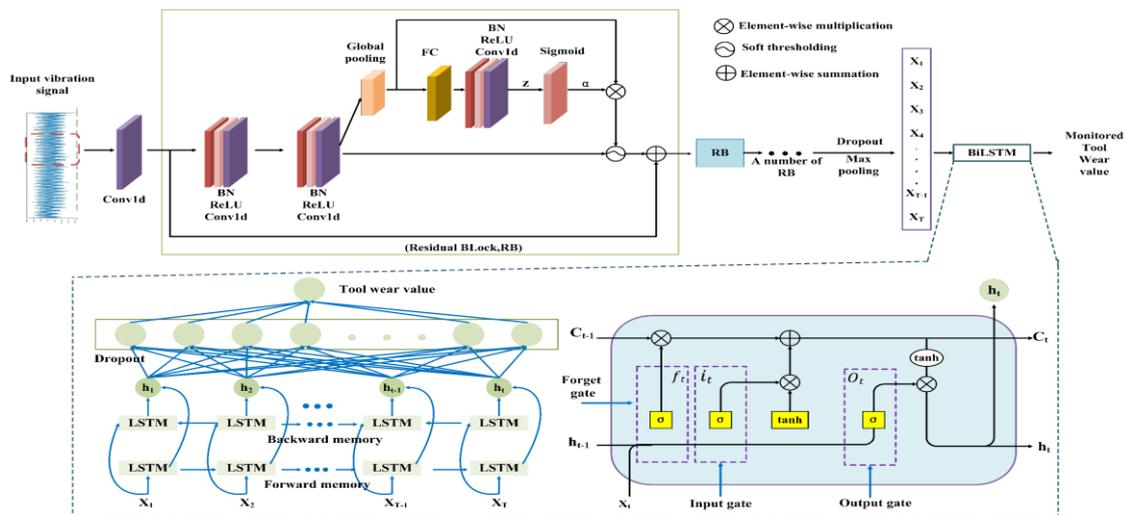


Fig. 3. Attention\_ResNet\_BiLSTM model.

### 3.1.3 MSDA loss definition

Inspired by Deep CORAL and JMMD alignment domain offsets, we designed the MSDA loss to first aligning the distributions of features in the two dataset domains via second order statistical alignment, and then aligns the joint distribution of features and labels in deeper layers of the neural network through the joint maximum average difference, with the MSDA loss function as follows:

$$\mathcal{L}_{\text{MSDA}} = \beta \frac{1}{4d^2} \|C_S - C_T\|_F^2 + \gamma \left( \left\| \mathbb{E}_P \left( \bigotimes_{l=1}^{L_l} \phi^l(z^{sl}) \right) - \mathbb{E}_Q \left( \bigotimes_{l=1}^{L_l} \phi^l(z^{tl}) \right) \right\|_{\bigotimes_{l=1}^{L_l} \mathcal{H}^l}^2 \right) = \beta \mathcal{L}_{\text{CORAL}} + \gamma \mathcal{L}_{\text{JMMD}} \quad (8)$$

We add it to the loss function to achieve domain adaptation for feature transfer between the target and source domains, as can be seen above, and the final loss function is set:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_{\text{MSDA}} = \mathcal{L}_c + \beta \mathcal{L}_{\text{CORAL}} + \gamma \mathcal{L}_{\text{JMMD}} \quad (9)$$

As can be seen from the MSDA loss formula, the training process contains two trade-off parameters  $\beta$  and  $\gamma$ . These two trade-off parameters have an important impact on the accuracy of MSDA, and we will determine the settings of these two trade-off parameters based on specific experimental data in Section 4.

### 3.2 Attention\_ResNet\_BiLSTM model

The key idea behind ResNet is adding directly connected channels to the network, the concept of identity short-cuts. The prior structure of the network is a non-linear transformation of the performance data, while identity shortcuts keep some percentage of the output from previous layers of the network that decrease the number of computations and parameters, and the gradient disappearance does not occur. BiLSTM employs forward and reverse bidirectional operations to improve learning of sequence features from known time series and reverse position sequences. Combining the advantages of ResNet and BiLSTM, a soft thresholding ResNet\_BiLSTM deep learning model is proposed to be used as a feature extractor for milling tool condition monitoring, the principle of which is shown in Figure 3. The residual block in ResNet\_BiLSTM differs from traditional residual network (Figure 4) in that it uses soft thresholding to remove noise-related features, and the attention-based mechanism of a soft thresholding layer is inserted in the form of a nonlinear into the the residual block, and the values of the thresholds can be automatically learned

during the training of the network, as described in the following sections. It is found that the computation time increases when the complexity of the residual blocks increases and under different task conditions, the robustness of the model decreases slightly.

Therefore, we have designed the ResNet\_BiLSTM model, where BiLSTM serves as the temporal encoder built upon the prior knowledge of ResNet. By leveraging the excellent feature extraction capabilities of ResNet and the temporal modeling abilities of BiLSTM, this model effectively explores the spatiotemporal features of the signal, thus improving the accuracy and robustness of vibration signal diagnosis. In this model, ResNet acts as the frontend feature extractor to capture local features from the vibration signal. The extracted feature sequence is then mapped into the feature representation of BiLSTM. Two fully connected layers serve as the input to BiLSTM, enabling the modeling and prediction of temporal relationships. This further enhances the accuracy and robustness of vibration signal diagnosis.

#### 3.2.1 Traditional residual network

ResNet is a deep-learning method that has received much attention in the past few years [14], and the residual block (RB) is its basic building block, as shown in Figure 4, the RB consists of two ReLu, two BN, two convolutional layers and a Shortcut Connection, the Shortcut Connection is what allows ResNet to outperform general ConvNets. In a general convolutional network, the cross-entropy error gradient is back-propagated layer by layer. By using identity shortcuts, the gradient can effectively fluxes to previous layers near the input layer, thus allowing for efficient parameter updates. Figure 4, shows the general structure of the ResNet, which is made up of one input layer, and convolution layers, many RB, a BN, a ReLu, a GAP, and outputs fully-connected layers, and serves as the basis for further improvements needed in this study.

#### 3.2.2 Soft thresholding based on attention mechanism

The cutting force signal collected in the experiment contains rich information about the change of tool condition, but there is inevitably some noise in the cutting force signal, when the model extracts these features from the signal, it is not beneficial to monitor the tool condition, in order to retain the valuable features to remove the redundancy, these unimportant features

may be noticed through the attention mechanism, by soft threshold painting to make them zero [7], thereby enhancing in terms of the neural network's ability to extract useful information from the cut-off force signal. The traditional soft thresholding operation often requires setting filters based on human experience, and the setting of filter thresholds requires a great deal of expertise. Deep learning changes this way of thinking, and instead of needing to think about setting thresholds, deep learning uses gradient descent to learn automatically, and the formula for soft thresholding is:

$$y = \begin{cases} x - \tau & x > \tau \\ 0 & -\tau \leq x \leq \tau \\ x + \tau & x < -\tau \end{cases} \quad (10)$$

where  $x$  is the input feature,  $y$  is the output feature, and  $\tau$  is the threshold (a positive parameter). Soft thresholding preserves useful negative features by setting the features close to zero in the ReLU activation layer to zero.

The soft threshold is inserted into the residual block as a nonlinear layer, as shown in Figure 5. The residual block is different from the classical residual block in Figure 4, and there is a special module dedicated to learning the threshold in Figure 5.

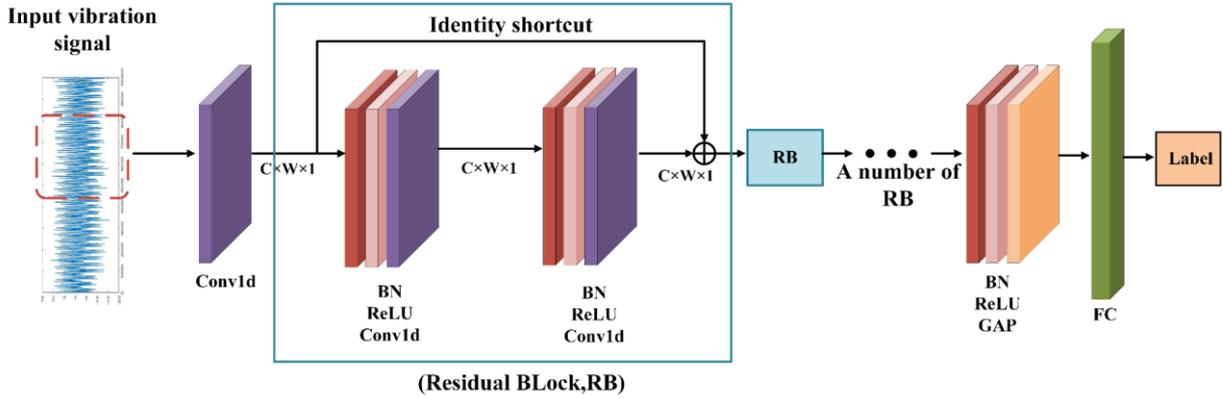


Fig. 4. Traditional residual network.

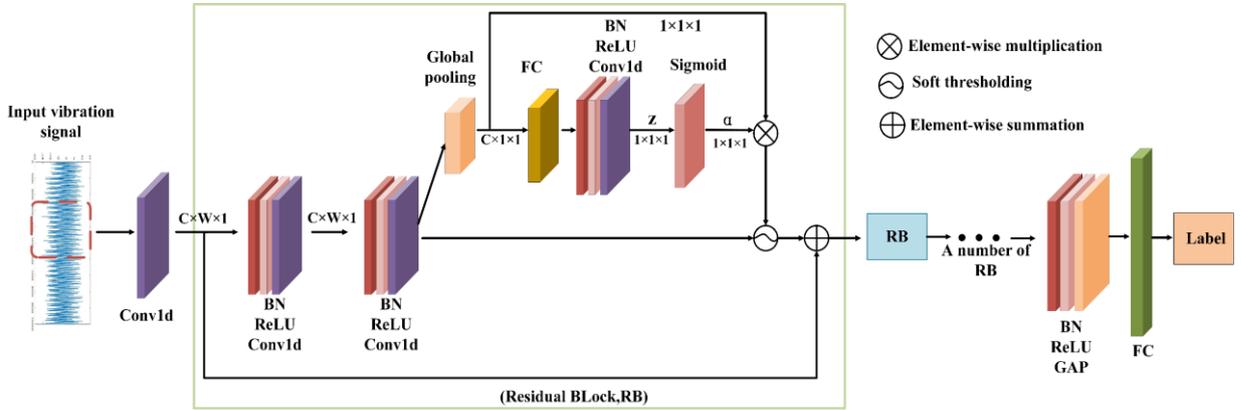


Fig. 5. Residual network with added attention mechanism.

In gradient backpropagation, the absolute value of the feature mapping  $X$  is first subjected to the GAP operation to obtain a one-dimensional vector, which is then passed to the FC layer to obtain a scaling parameter, and the end uses the Sigmoid function to its scaling to (0.1), the range is expressed as:

$$\alpha = \frac{1}{1+e^{-z}} \quad (11)$$

where  $\alpha$  is the scaling parameter and  $z$  is the output of the FC layer, multiply  $\alpha$  based on the mean of  $|x|$  in order to achieve the desired threshold value, and the threshold value is denoted as:

$$\tau = \alpha \cdot \text{average}_{i,j,c} |x_{i,j,c}| \quad (12)$$

where  $\tau$  is the threshold value,  $i, j, c$  are the width, height and index of the channel of the extracted feature mapping, respectively. The iterative process of deep learning can keep the threshold value in a reasonable range of values so that the soft thresholding not all outcomes will be zero.

### 3.2.3 Bi-directional long short term memory

LSTM is a kind of RNN, RNN was found to have the vanishing of gradients, exploding gradients and poor dependencies over

long distances in practical application development, so LSTM was created. The main body of LSTM is similar to LSTM, and the key enhancement is the addition of three gating units to the hidden layer for learning long-term memory. The structure of BiLSTM is two independent LSTMs with input sequences in forward and inverse order are input to the two LSTMs in the case of feature extraction, and both output vectors, i.e., together with the extracted feature vectors, are sewn together and used as the final feature expression for that network.

$$\begin{cases} c_t = f_t \otimes c_{t-1} \oplus i_t \\ \otimes (\tanh(w_c x_t + U_c h_{t-1} + b_c)) \\ h_t = o_t \otimes \tanh(c_t) \end{cases} \quad (13)$$

where,  $f_t$ ,  $i_t$ , and  $o_t$  mean the input gate forgetting gate, input gate, and output gate, respectively;  $c_t$ ,  $h_t$  and  $h_{t-1}$  mean the long-term memory, short-term memory, and the feature information of the previous time series, respectively;  $b$ ,  $w$ , and  $U$  mean the model parameters to be learned during the iterative process, respectively;  $\oplus$  and  $\otimes$  denote multiplication and addition, respectively.

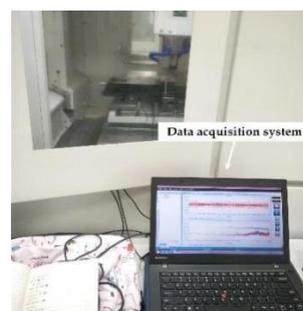
The single-layer BiLSTM is actually two LSTMs, one forward processing sequence and one reverse processing sequence. After processing, the outputs of the two LSTMs are stitched together as the output of the BiLSTM, so that the obtained feature vector at time  $t$  has information between the past and the future at the same time, with the following equation:

$$\begin{cases} h_t^R = f^R(w_1 x_t + w_2 h_{t-1}^R) \\ h_t^L = f^L(w_3 x_t + w_5 h_{t+1}^L) \\ h_t = f(w_4 h_t^R + w_6 h_t^L) \end{cases} \quad (14)$$

where  $h_t^R$ ,  $h_t^L$ , and  $h_t$  denote the moment  $t$ , the forward LSTM, the reverse LSTM, and the final output of the feature vector, respectively.  $w$  is the weight parameter of the BiLSTM to be learned. After the feature vectors are processed by the BiLSTM, two double tangent ( $\tanh$ ) functions are stacked as activation functions to monitor the current condition.



(a)



(b)

### 3.3 Training setup

To rationalize the testing process, we trained each model for 160 iterations, and during the training process, for the first 50 iterations, we only used model weight-sharing transfer to obtain the so-called pre-trained model, and then activated the MSDA domain adaptive strategy. The model training and testing process alternated, and during the training we used small batches of Adam for backpropagation, each batch size equal to 64, using a "stepwise" strategy as the learning rate decay method, with an initial learning rate of 0.001, decaying at 80 and 120, respectively, multiplied by 0.1.

#### 3.3.1 Normalization

Data normalization is a fundamental step in migration learning that ensures the input values are within a specific range. It plays a crucial role in reducing differences in data distribution between the source and target domains, enhancing the model's generalization capability, and improving its adaptability to target domain data. In this study, we employ Z-score normalization, which is calculated using the following equation:

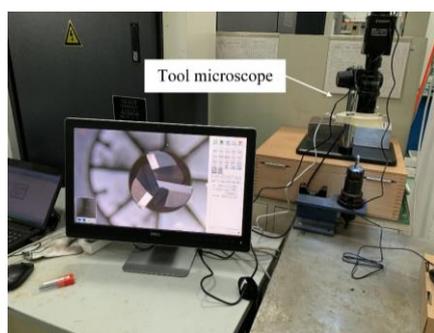
$$x_i^{\text{normalize}} = \frac{x_i - x_i^{\text{mean}}}{x_i^{\text{std}}}, i = 1, 2, \dots, N \quad (15)$$

where  $x_i$  is the input data,  $x_i^{\text{mean}}$  is the mean of  $x_i$ , and  $x_i^{\text{std}}$  is the standard deviation of  $x_i$ .

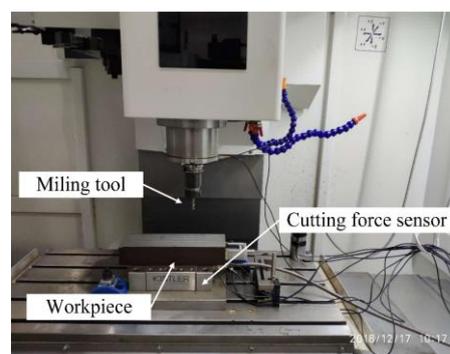
## 4. Experiment verification and research

### 4.1. Experimental setup

The TCM experiments were performed on a CNC (DMTG VDL850A) machining center as shown in Figure 6. The workpiece material was AISI 1045 steel with dimensions of L1300 mm  $\times$  W100 mm  $\times$  H80 mm. The tools used are three-flute uncoated carbide end milling cutters with a diameter of 10 mm and the chemical properties given in Table 1.



(c)



(d)

**Fig. 6.** (a) CNC machine (b) data acquisition system (c) tool microscope (d) experimental platform.

Tab. 1. Chemical properties of workpiece material.

Carbon (%)	Silicon (%)	Manganese (%)	Nickel (%)	Chromium (%)	Copper (%)
0.42~0.50	0.17~0.37	0.50~0.80	<0.30	<0.25	<0.25

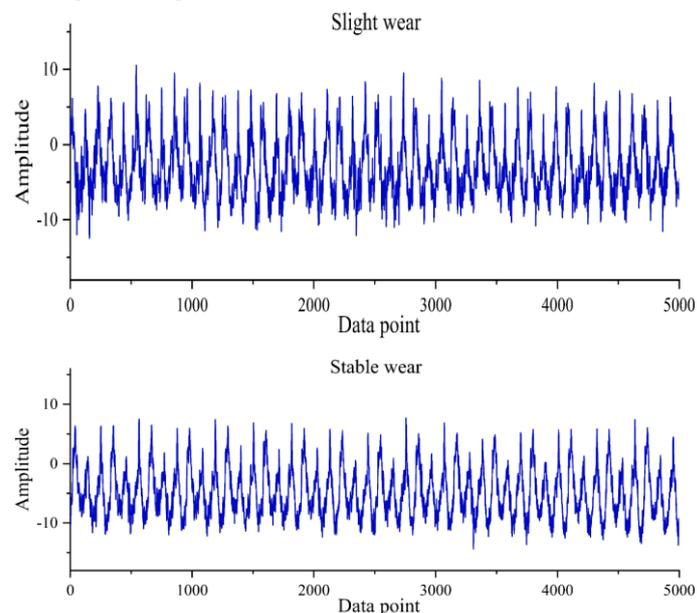
We conducted TCM experiments under four operating conditions, each parameter in the experiment can be changed, in the machining process of the tool, the cutting force is often the most sensitive, and contains a wealth of information about the working conditions, and the dynamic cutting characteristics of the tool are closely linked to the dynamics of the dynamometer, a single dynamometer sensor has the advantages over the dynamometer in that it is inexpensive, easy to install, does not affect normal processing. We used a Kistler 9139AA dynamometer to collect cutting force signals during tool machining, as shown in Figure 6, and installed it under the workpiece being machined to collect data with a sampling frequency set to 12000 Hz. We collected and organized the cutting force signals from four simulated experiments to generate four experimental data sets, D1, D2, D3, D4 as shown in Table 2.

Tab. 2. Experimental parameters.

Domain	Tool Number	Spindle Speed(rpm)	Feed speed(mm/min)	Axia Cut Depth(mm)
D1	1	2300	0.4	400
D2	2	2300	0.5	450
D3	3	2400	0.4	450
D4	4	2400	0.5	500

The tool and the workpiece in the experiment are dry cutting, and the wear and tear of the tool is relatively rapid, we will put the tool offline under the industrial microscope after each tool stroke to measure the distance of 1.5 meters for each stroke. ISO3685-1977 defines tool wear as the VB width of wear on the side of the tool, but in the actual experiment, our results indicate

that the change in tool side wear width is not sufficiently obvious for an effective measurement and that measurement error is large. In the experiment, we positioned the microscope vertically and measured the length of wear on each slide, as shown in Figure 8. We took the maximum wear length of the tool section as the tool wear standard,  $VB = \text{Max}(VB1, VB2, VB3)$ . In our experiments, we found that the tool tends to reach the end of life in the tenth stroke, so we performed 10 strokes of the tool for each working condition and made offline measurements each time, as shown in Figure 9 which shows the tool wear variation for the 10 offline measurements. We divided the data set into three conditions according to the tool wear length, slight wear ( $VB \leq 0.8 \text{ mm}$ ), stable wear ( $0.8 \text{ mm} < VB \leq 1.6 \text{ mm}$ ), and sharp wear ( $VB > 1.6 \text{ mm}$ ). Figure 7 shows the cutting force signals on the three conditions.



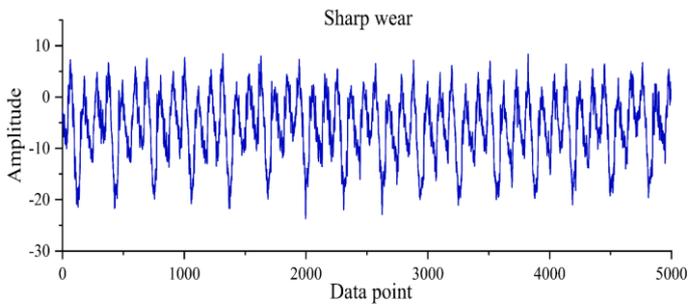
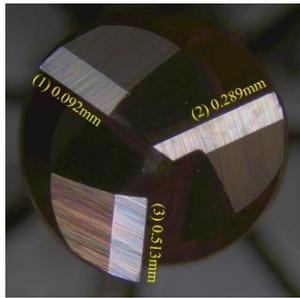
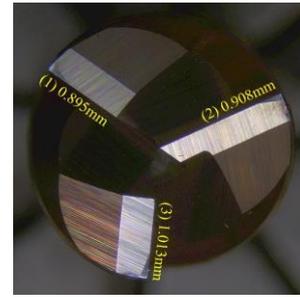


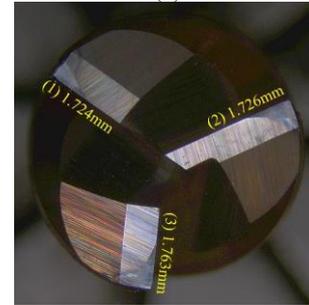
Fig. 7. Time domain signals under different wear conditions.



(a)



(b)



(c)

Fig. 8. Images of the three conditions of the tool: (a) Slight wear (b) Stable wear (c) Sharp wear

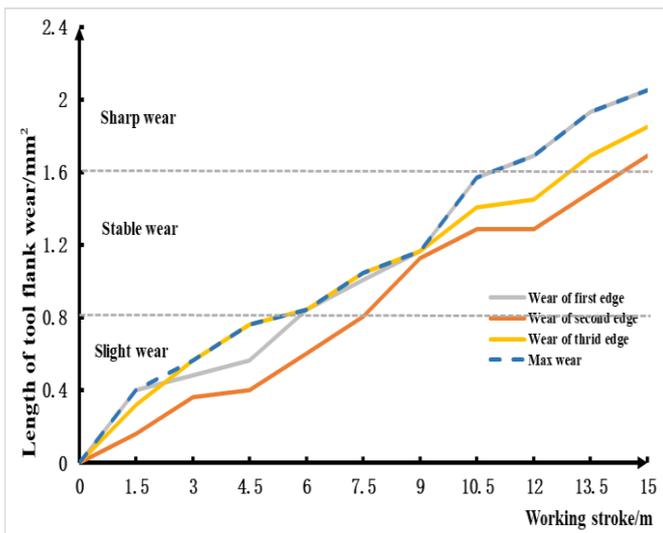


Fig. 9. Wear length variation of the three cutting edges during milling.

#### 4.2 Description of the data set

According to the experiments, we collected a total of four different cutting force signals of milling tools under different working conditions, and to test the efficacy of the method proposed in this study, we designed four transfer tasks, which are task (D1→D2), task (D2→D1), task (D3→D4), and task (D4→D3), and the left and right of "→" denote the source domain dataset and the target domain dataset, respectively, and when transfer learning is performed, only the source domain data are labeled and the target domain is unlabeled, but data

from the target domain also participates in training to achieve domain adaptation. In our experiments, we collect 1474560 time points for each working condition, and we directly use the measured cutting force signals as the input to the model, with each sample set to 1024 time series. When training, we use 80 percent of the total data set in both the source and target domains as the training set and the remaining 20 percent of the data set as the test set.

#### 4.2.1 Selection of cutting force

In the process of milling tool condition monitoring, we use radial cutting force as the input to the model. The radial force is one of the primary components of the cutting force and directly reflects the interaction between the tool and the workpiece during the cutting process. Furthermore, the relationship between radial force and tool condition or workpiece material is more apparent. Additionally, the variation range of radial force is typically larger than that of axial force or tangential force because the relative motion between the tool and the workpiece primarily occurs along the radial direction during the cutting process. Therefore, the radial force exhibits higher sensitivity and provides more information for tool condition detection and monitoring.

### 4.3 Model hyperparameter analysis

Grid search is a commonly used method for hyperparameter tuning. One of its advantages is its simplicity and intuitiveness. By exhaustively searching through predefined ranges of hyperparameters, grid search can cover all possible combinations in the hyperparameter space, avoiding the omission of potential optimal combinations. Another advantage of this method is its comprehensiveness, as it explores various scenarios in the hyperparameter space, including extreme values and boundary cases. This comprehensive search helps understand the impact of different hyperparameter values and find the best combination of hyperparameters. Additionally, the results of grid search provide a set of hyperparameter combinations that clearly demonstrate the influence of each hyperparameter on the model's performance. This offers interpretability and information to guide subsequent hyperparameter adjustments.

As can be seen from Equation 11, the total loss function of MSDA contains two trade-off parameters, and the setting of the trade-off parameters has an important impact on the transfer ability of MSDA. We conduct experiments with D1 by the source domain and D2 as the target domain. After conducting initial attempts, we take  $\beta \in [0.08, 1]$  and  $\gamma \in [0, 0.0000001]$  for domain adaptive experiments, respectively, and draw a three-dimensional plot of the accuracy change in the target domain, as shown in Figure 10. According to the three-dimensional plot of accuracy, we can see that the test set accuracy of the target domain reaches a maximum of 98.26% when  $\beta$  equals to 0.1 and  $\gamma$  equals to 0.00001, and the accuracy of the target domain is lower when  $\beta = 0.2$  and  $\gamma \in [0, 0.001]$ . According to the accuracy 3D plot we also found that when the value of  $\gamma$  is fixed and  $\beta$  is changed, the correct rate changes less, and when the value of  $\beta$  is fixed and  $\gamma$  is changed, the accuracy rate changes significantly, thus indicating that the influence of  $\gamma$  on the MSDA domain adaption is greater than that of  $\beta$ .

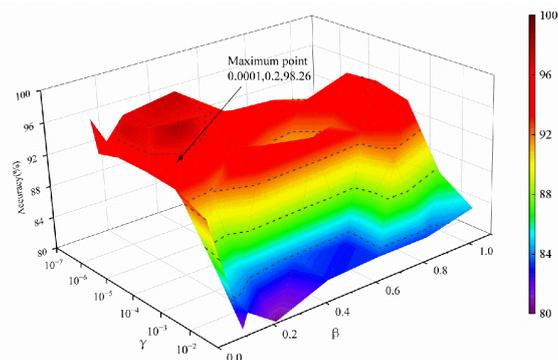


Fig. 10. The results of model hyperparameter analysis.

## 4.4 Results and Discussion

### 4.4.1 Attention\_ResNet\_BiLSTM

In order to test the efficacy and superiority of our proposed model, four transfer experiments were conducted to compare several other models, mainly including AlexNet, BiLSTM, ResNet18, LeNet, and Attention\_ResNet\_BiLSTM, using the MSDA domain adaptation framework proposed in. The prediction results of the transmission task of TCM are shown in The Figure 11 shows that the average classification accuracy of the four transmission tasks shows that the Attention\_ResNet\_BiLSTM model has more stable performance and can realize the monitoring of the condition of the tool under variable working conditions.

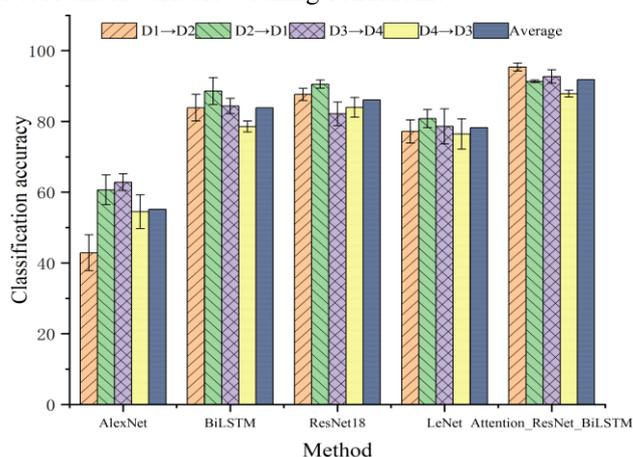


Fig. 11. Correctness of different feature extraction models in four transfer tasks.

The reasons are as follows: firstly, the deep feature extraction is performed by the residual network, which effectively prevents the gradient disappearance phenomenon, information useful to tool wear monitoring is then augmented by the attention mechanism and soft thresholds, and the redundant information is filtered and weakened, and finally the learning of sequence features is enhanced by BiLSTM based on the forward and

reverse position sequences of the time series, and more useful discriminative features are extracted to make the classification more accurate. It can be seen that the Attention\_ResNet\_BiLSTM achieves more than 85% correct rate in all four classification tasks, and the residual 18 network also shows comparable performance with the Attention\_ResNet\_BiLSTM in the D2→D1 transfer task, but in the other three transmission tasks, of the four experiments, The Attention\_ResNet\_BiLSTM performs significantly better than the other methods and has the lowest fluctuations in error.

#### 4.4.2 MSDA Performance Evaluation

Tables 3-6 show the performance of our proposed MSDA with the adaptive method and the benchmark model in four transfer tasks. To verify the effectiveness of MSDA, we all use

Attention\_ResNet\_BiLSTM as the feature extractor, and use Network-based DTL, Instanced-based DTL (Adabn), MK-MMD, JMMD, and DANN transfer learning methods for comparative analysis. From the results obtained in the table, the domain adaptive method MSDA for aligning multiple spatial domains proposed in this paper achieves the best results, especially on F1 SCORE for all four transfer tasks. Tables 3 and 5 show that our method outperforms other methods by about 5 percent in terms of correctness, and we find that the model is more likely to extract features from the vibration signals when the data set with lower feed rate is labeled source domain data, and the F1 SCORE of our method is about 10 percent lower than other methods in Tables 4 and 6, indicating that our method can effectively reduce interference from individual signals as well as efficiently close the distance between the source and target domains.

Table 3. Performance of six methods for the classification task D1→D2.

Method	Accuracy	Precision	Recall	F1 Score	AUC	Far
Network	0.8708±0.0228	0.5081±0.0254	0.8012±0.0334	0.6815±0.0516	0.6445±0.0214	0.0145±0.0195
Adabn	0.7850±0.0126	0.6912±0.0034	0.7433±0.0320	0.5986±0.0259	0.8732±0.0437	0.0851±0.0178
MK-MMD	0.9159±0.0035	0.6912±0.0293	0.6786±0.0254	0.7092±0.0322	0.7903±0.0257	0.0751±0.0154
JMMD	0.9085±0.0213	0.8415±0.0103	0.9515±0.0156	0.9124±0.0125	0.9541±0.0122	0.0145±0.0021
DANN	0.9242±0.0198	0.9343±0.0050	0.9735±0.0110	0.9140±0.0084	0.9701±0.0025	0.0115±0.0024
MSDA	0.9535±0.0111	0.9425±0.0145	0.9833±0.0096	0.9151±0.0067	0.9802±0.0017	0.0117±0.0005

Table 4. Performance of six methods for the classification task D2→D1.

Method	Accuracy	Precision	Recall	F1 Score	AUC	Far
Network	0.8346±0.0398	0.8638±0.0254	0.7500±0.0360	0.7992±0.0191	0.8665±0.0079	0.0285±0.0064
Adabn	0.6927±0.0450	0.7915±0.0126	0.6879±0.0245	0.7587±0.0231	0.6620±0.0048	0.0312±0.0150
MK-MMD	0.8951±0.0186	0.9049±0.0304	0.8745±0.0256	0.9539±0.0051	0.9393±0.0231	0.0170±0.0121
JMMD	0.8976±0.0200	0.9521±0.0103	0.8745±0.0122	0.9414±0.0191	0.9231±0.0021	0.145±0.0085
DANN	0.8775±0.0358	0.9112±0.0301	0.9058±0.0161	0.9075±0.0120	0.8701±0.0231	0.0125±0.0108
MSDA	0.9232±0.0170	0.9479±0.0159	0.8975±0.0301	0.9576±0.0085	0.9583±0.0156	0.0045±0.0144

Table 5. Performance of six methods for the classification task D3→D4.

Method	Accuracy	Precision	Recall	F1 Score	AUC	Far
Network	0.9123±0.0048	0.7583±0.0147	0.7922±0.0282	0.5501±0.0158	0.8550±0.0413	0.0808±0.0051
Adabn	0.8058±0.0335	0.5805±0.0337	0.8359±0.0362	0.5572±0.0373	0.7380±0.0251	0.1833±0.0188
MK-MMD	0.9114±0.0140	0.9051±0.0161	0.9108±0.0323	0.8787±0.0154	0.9065±0.0051	0.0252±0.0228
JMMD	0.9009±0.0101	0.9211±0.0054	0.8812±0.0145	0.9183±0.0121	0.8992±0.0097	0.0322±0.0085
DANN	0.9011±0.0082	0.8951±0.0252	0.8554±0.0026	0.8037±0.0158	0.9214±0.0045	0.0421±0.0252
MSDA	0.9271±0.0189	0.9024±0.0321	0.9402±0.0022	0.8965±0.0154	0.9045±0.0052	0.0145±0.0184

Table 6. Performance of six methods for the classification task D4→D3.

Method	Accuracy	Precision	Recall	F1 Score	AUC	Far
Network	0.8638±0.0089	0.8135±0.0251	0.8216±0.0182	0.7714±0.0230	0.8565±0.0339	0.0193±0.0110
Adabn	0.7986±0.0253	0.5481±0.0337	0.8016±0.0360	0.7296±0.0252	0.8232±0.0195	0.0225±0.0107
MK-MMD	0.8591±0.0186	0.7270±0.0288	0.7825±0.0214	0.8359±0.0104	0.7524±0.0245	0.0121±0.0085
JMMD	0.8752±0.0250	0.7735±0.0056	0.8835±0.0190	0.8154±0.0208	0.8013±0.0109	0.0202±0.0057
DANN	0.8481±0.0101	0.8018±0.0082	0.8182±0.0135	0.7573±0.0026	0.8454±0.0252	0.0185±0.0144
MSDA	0.8783±0.0094	0.8316±0.0150	0.8526±0.0154	0.8571±0.0104	0.8621±0.0025	0.0117±0.0078

To verify the ability of our method to classify each class in the tool milling process, we draw the confusion matrix of the

MSDA method for four transfer tasks (Figure 12), and it can be seen in the confusion matrix that our model can effectively classify the milling signals of the tools, but it is often difficult to achieve good results when the tool signals are in the frequency bands adjacent to the classes, which is because tool milling is a gradual process, when the wear signal is at the critical point between classes, the signal features are less different, and the classification model is often difficult to perform effective classification, which leads to the second category signal classification accuracy is often difficult to achieve satisfactory accuracy, but for the first and third category our model achieves more than 98% correct rate on all four classification tasks, with only a small number of critical values, resulting in misclassification.

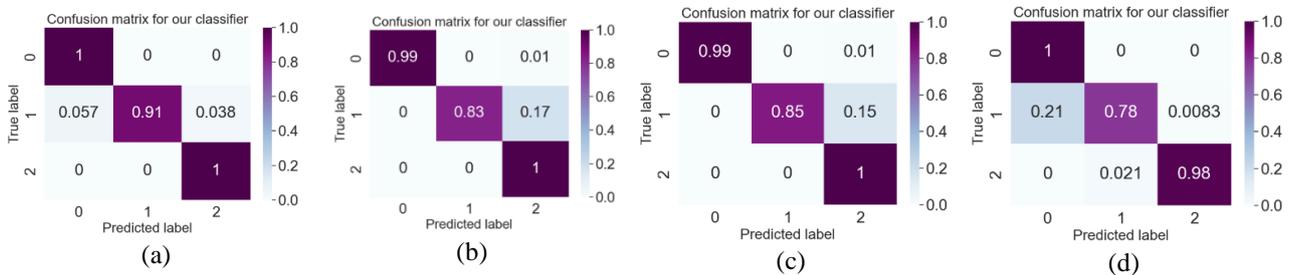


Fig. 12. Confusion matrix for four transfer tasks: (a) D1 → D2 (b) D2 → D1 (c) D3 → D4 (d) D4 → D3.

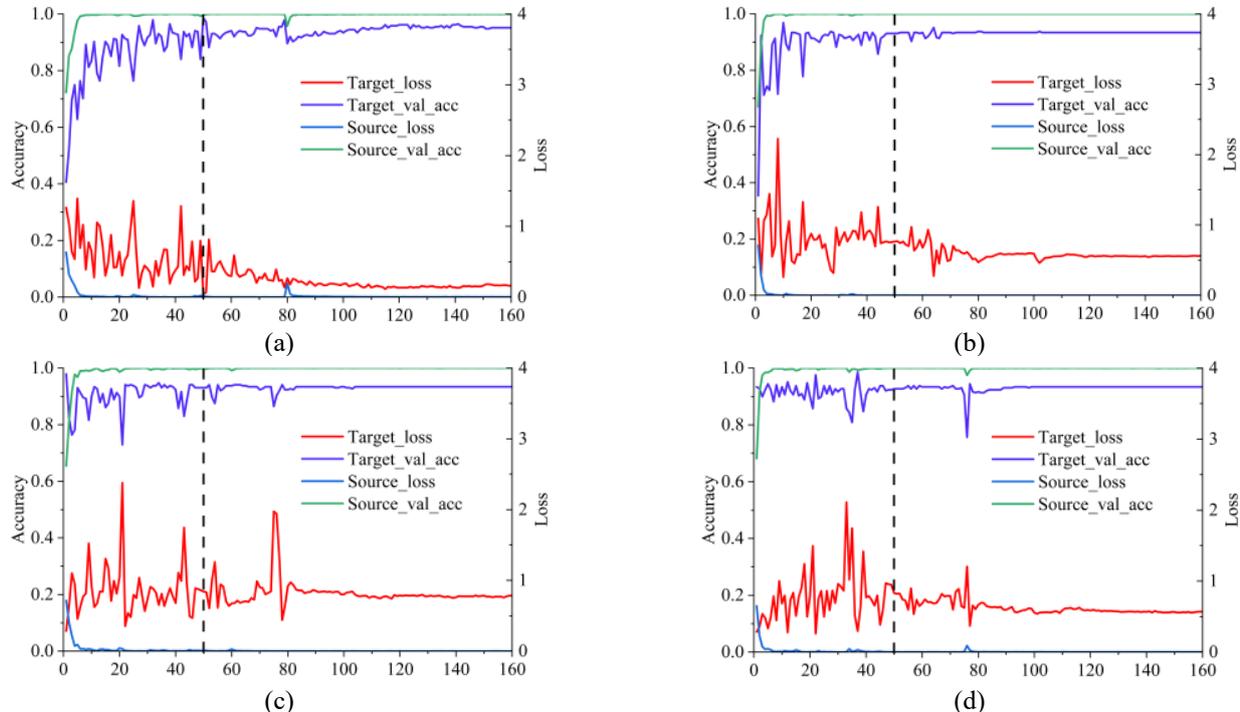


Fig. 13. Change in loss value for four transfer missions: (a) D1 → D2 (b) D2 → D1 (c) D3 → D4 (d) D4 → D3. The dashed line indicates the introduction of MSDA domain adaptation at the 51<sup>th</sup>.

In order to prevent the problem of small size of the training dataset, which leads to insufficient data volume to support the training and evaluation of the model, we expand the dataset by

In the training process we set the first 50 times to train only the transfer of model weights, the trained model as a pre-trained model, in the 51st time to introduce the MSDA domain adaptive way, in order to gain a clearer understanding of the training process, in all four transfer tasks, we plot the change curve of the source and target domain with the correct rate and the correct loss value, from the Figure 13, we can see that in the first 50 times of pre-training the correct rate and loss value float. The correct rate and loss value fluctuate greatly in the first 50 pretraining sessions, but after the introduction of the MSDA domain adaptive approach in the 51st session, the correct rate and loss value soon stabilize, which also shows the effectiveness of our method for the transfer task.

using data overlapping. Data overlapping refers to extracting the subsequences of a signal in such a way that the neighboring subsequences have a certain overlapping portion. A larger

percentage of overlap not only provides more training samples, but also the higher correlation between training samples increases the diversity of samples, which is conducive to increasing the learning ability and stability of the model. So we consider using 75% overlap ratio. From Fig. 13, it can be seen that the proposed framework has obvious accuracy and loss function jitter in the validation set during the migration task of D3→D4, and it requires many iterations to stabilize. So the D3→D4 task was targeted for data expansion experiments.

The experimental results are presented in Figure 14. By comparing Figure 14(a) with Figure 13(c), it is evident that with an increased amount of training data and under the same MSDA framework, the training process reaches stability in fewer iterations, and the fluctuations in loss and accuracy of the validation sets for both the source and target domains are significantly reduced. This indicates the critical impact of increasing the training data on the training process. After data augmentation, we further compared our approach with other methods as shown in Figure 14 and Table 5. The results

demonstrate that all four domain adaptation methods achieved improved accuracy after data augmentation. By examining the loss curves of the four methods in Figure 14, it is clear that the MSDA method exhibits smoother loss variation. Although JMMD gradually stabilizes with deepening training, it initially experiences significant fluctuations in loss. Despite the enlarged training dataset, the Adabn method still performs poorly, indicating its ineffectiveness in the task of milling tool condition monitoring. DANN, excluding MSDA, performs the best among the four methods. However, it can be observed that as training progresses, the loss curve of DANN shows an upward trend accompanied by a slight decrease in accuracy, indicating overfitting. This demonstrates that MSDA not only achieves higher accuracy but also effectively avoids issues like vanishing or exploding gradients, ensuring stable changes in the loss function. This further validates the superiority of the proposed method in this study.

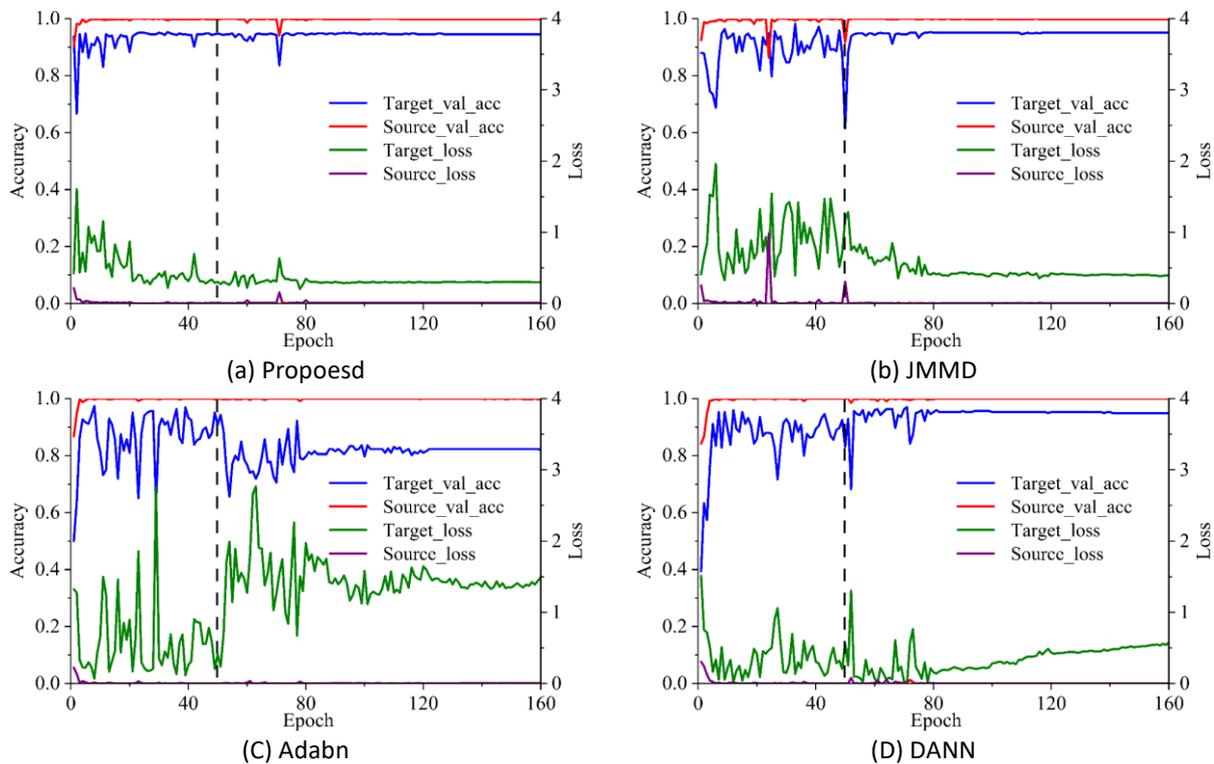


Fig. 14. Comparison chart of methods for data expansion experiments.

## 5. CONCLUSION

This paper proposes a novel MSDA domain Adaptation method based on multi feature space alignment, integrating attention

mechanism, BiLSTM and ResNet, to realize tool condition monitoring under variable working conditions. The experimental results show that the proposed

Attention\_ResNet\_BiLSTM method is superior to other four methods (AlexNet, BiLSTM, ResNet18, and LeNet) based on four transmission tasks. In terms of classification accuracy, the proposed MSDA domain adaptive method improves the average accuracy by about 5% compared to the other five classification methods. It has strong robustness to changes in variable working conditions and performs best in each classification task. Therefore, the proposed method is expected to be suitable for identifying tool wear in the actual machining process.

In addition, our TCM experiment only considered variations in cutting parameters (spindle speed, axial depth of cut, and feed rate). The focus of future research will be on implementing TL-based TCM for different tools and materials. There are limitations to consider and address in proposing the MSDA domain adaptation method: (1) If there are significant differences between the data or insufficient feature representations, it may result in poor alignment performance. (2) The method relies on a specific model architecture, and

although the model performs well on specific tasks and datasets, its adaptability may be limited on other tasks or datasets. (3) Additionally, the method utilizes a dataset from milling tool experiments for condition monitoring. However, this dataset may only represent specific operating conditions and experimental settings, which may not generalize well to other operating conditions or real industrial environments.

To further promote research on domain adaptation and milling tool condition monitoring, future studies can explore the following directions: First, it is possible to try combining other domain adaptation methods, such as MK-MMD or DANN, to further improve the effectiveness of feature alignment. Second, exploring more complex feature extractor model structures and attention mechanisms can be done to extract richer and more accurate feature representations. Additionally, considering the introduction of domain-specific prior knowledge or label information can further enhance the performance of condition monitoring.

## ACKNOWLEDGMENTS

This project was supported by the National Natural Science Foundation of China (No. 52275125), the Zhejiang Natural Science Foundation of China (No. LY23E050001), the General Project of the Department of Education of Zhejiang Province (No. Y202249011), the Science and Technology Plan Project of Jiaxing city (Grant. 2023AY11013), and the Science and Technology Plan Project of Wenzhou city (Grant. G20220006).

## REFERENCES

1. Ahmad M A F, Nuawi M Z, et al. Tool wear monitoring using macro fibre composite as a vibration sensor via I-kaz statistical signal analysis. *ARPN Journal of Engineering and Applied Sciences*, 2018, 13(11): p. 3607-3616.
2. Ben-David, S., et al., A theory of learning from different domains. *Machine learning*, 2010. 79: p. 151-175. <https://doi.org/10.1007/s10994-009-5152-4>
3. Chen, K., et al., Short-term load forecasting with deep residual networks. *IEEE Transactions on Smart Grid*, 2018. 10(4): p. 3943-3952. <https://doi.org/10.1109/TSG.2018.2844307>
4. Chen, W., et al., Diagnosis of wind turbine faults with transfer learning algorithms. *Renewable Energy*, 2021. 163: p. 2053-2067. <https://doi.org/10.1016/j.renene.2020.10.121>
5. Cheng, M., et al., Intelligent tool wear monitoring and multi-step prediction based on deep learning model. *Journal of Manufacturing Systems*, 2022. 62: p. 286-300. <https://doi.org/10.1016/j.jmsy.2021.12.002>
6. Dimla Sr, D. and P.M. Lister, On-line metal cutting tool condition monitoring.: I: force and vibration analyses. *International Journal of Machine Tools and Manufacture*, 2000. 40(5): p. 739-768. , [https://doi.org/10.1016/S0890-6955\(99\)00085-1](https://doi.org/10.1016/S0890-6955(99)00085-1)
7. Donoho, D.L., De-noising by soft-thresholding. *IEEE transactions on information theory*, 1995. 41(3): p. 613-627. <https://doi.org/10.1109/18.382009>
8. G. Vetrichelvan, S. Sundaram, S.S. Kumaran, P. Velmurugan, An investigation of tool wear using acoustic emission and genetic algorithm, *J. Vib. Control* 2014; 21(15): 3061-3066, <https://doi.org/10.1177/1077546314520835> .
9. Ganin, Y. and V. Lempitsky. Unsupervised domain adaptation by backpropagation. in *International conference on machine learning*. 2015. PMLR.

10. Gretton, A., et al., Optimal kernel choice for large-scale two-sample tests. *Advances in neural information processing systems*, 2012. 25.
11. Guan, S., W. Song, and H. Pang. Tool Wear Feature Extraction Based on Hilbert Marginal Spectrum. in *IOP Conference Series: Materials Science and Engineering*. 2017. IOP Publishing. <https://doi.org/10.1088/1757-899X/230/1/012049>
12. Gudelek M U, et al. An industrially viable wavelet long-short term memory-deep multilayer perceptron-based approach to tool condition monitoring considering operational variability. *Proceedings of the Institution of Mechanical Engineers. Journal of Process Mechanical Engineering*, 2022. <https://doi.org/10.1177/09544089221142161>
13. Hahn, T.V. and C.K. Mechefske, Self-supervised learning for tool wear monitoring with a disentangled-variational-autoencoder. *International Journal of Hydromechatronics*, 2021. 4(1): p. 69-98. <https://doi.org/10.1504/IJHM.2021.114174>
14. He, K., et al. Deep residual learning for image recognition. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. <https://doi.org/10.1109/CVPR.2016.90>
15. He, Z., et al., Ensemble transfer CNNs driven by multi-channel signals for fault diagnosis of rotating machinery cross working conditions. *Knowledge-Based Systems*, 2020. 207: p. 106396. <https://doi.org/10.1016/j.knsys.2020.106396>
16. Huang, J., et al., Geometric deep learning for shape correspondence in mass customization by three-dimensional printing. *Journal of Manufacturing Science and Engineering*, 2020. 142(6): p. 061003. <https://doi.org/10.1115/1.4046746>
17. Jamshidi M, Chatelain J F, et al. Tool condition monitoring using machine tool spindle electric current and multiscale analysis while milling steel alloy. *Journal of Manufacturing and Materials Processing*, 2022, 6(5): p. 115. <https://doi.org/10.3390/jmmp6050115>
18. Javed, K., et al., Tool wear monitoring and prognostics challenges: a comparison of connectionist methods toward an adaptive ensemble model. *Journal of Intelligent Manufacturing*, 2018. 29: p. 1873-1890. <https://doi.org/10.1007/s10845-016-1221-2>
19. Kasim N A, Nuawi M Z, et al. Cutting tool wear progression index via signal element variance. *Journal of mechanical engineering and sciences*, 2019. 13(1): p. 4596 – 4612. <https://doi.org/10.15282/jmes.13.1.2019.17.0387>
20. Kasim N A, Nuawi M Z, et al. Enhancing Clustering Algorithm with Initial Centroids in Tool Wear Region Recognition. *International Journal of Precision Engineering and Manufacturing*, 2021, 22: p. 843-863. <https://doi.org/10.1007/s12541-020-00450-5>
21. Kumar, A., et al., Sparse transfer learning for identifying rotor and gear defects in the mechanical machinery. *Measurement*, 2021. 179: p. 109494. <https://doi.org/10.1016/j.measurement.2021.109494>
22. Li, C., et al., Novel environmentally friendly manufacturing method for micro-textured cutting tools. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 2021. 8: p. 193-204, <https://doi.org/10.1007/s40684-020-00256-w>.
23. Li, X. and W. Zhang, Deep learning-based partial domain adaptation method on intelligent machinery fault diagnostics. *IEEE Transactions on Industrial Electronics*, 2020. 68(5): p. 4351-4361. <https://doi.org/10.1109/TIE.2020.2984968>
24. Li, Y., et al., A novel method for accurately monitoring and predicting tool wear under varying cutting conditions based on meta-learning. *CIRP annals* 2019; 68(1): 487-490. <https://doi.org/10.1016/j.cirp.2019.03.010>
25. Long, M., et al. Deep transfer learning with joint adaptation networks. in *International conference on machine learning*. 2017. PMLR.
26. Long, M., et al. Learning transferable features with deep adaptation networks. in *International conference on machine learning*. 2015. PMLR.
27. Marei M, El Zaatari, and W. Li. Transfer learning enabled convolutional neural networks for estimating health state of cutting tools. *Robotics and Computer-Integrated Manufacturing*, 2021, 71: p. 102145. <https://doi.org/10.1016/j.rcim.2021.102145>
28. Nawrocki W, Stryjski R, Kostrzewski M, et al. Application of the vibro-acoustic signal to evaluate wear in the spindle bearings of machining centres. *Journal of Manufacturing Processes*, 2023. 92: p. 165-178. <https://doi.org/10.1016/j.jmapro.2023.02.036>.
29. Rizal M, Ghani J A, Nuawi M Z, et al. An embedded multi-sensor system on the rotating dynamometer for real-time condition monitoring in milling. *The International Journal of Advanced Manufacturing Technology*, 2018. 95: p. 811 – 823. <https://doi.org/10.1007/s00170-017-1251-8>
30. Ross N S, Sheeba P T, et al. A novel approach of tool condition monitoring in sustainable machining of Ni alloy with transfer learning models. *Journal of Intelligent Manufacturing*, 2023: p. 1-19, <https://doi.org/10.1007/s10845-023-02074-8>
31. Schmidhuber, J., A local learning algorithm for dynamic feedforward and recurrent networks. *Connection Science*, 1989. 1(4): p. 403-412. <https://doi.org/10.1080/09540098908915650>
32. Sun, B., J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. in *Proceedings of the AAAI conference on artificial*

- intelligence. 2016. <https://doi.org/10.1609/aaai.v30i1.10306>
33. Sun, H., et al., In-process tool condition forecasting based on a deep learning method. *Robotics and Computer-Integrated Manufacturing*, 2020. 64: p. 101924. <https://doi.org/10.1016/j.rcim.2019.101924>
  34. Unver H O, Sener B. A novel transfer learning framework for chatter detection using convolutional neural networks. *Journal of Intelligent Manufacturing*, 2021: p. 1-20. <https://doi.org/10.1007/s10845-021-01839-3>
  35. Wang HC, Sun W, Sun WF, Ren Y. A novel tool condition monitoring based on Gramian angular field and comparative learning. *International Journal of Hydromechatronics* 2023; 6(2): 93-107, <https://doi.org/10.1504/IJHM.2022.10048957>.
  36. Yan, B., L. Zhu, and Y. Dun, Tool wear monitoring of TC4 titanium alloy milling process based on multi-channel signal and time-dependent properties by using deep learning. *Journal of Manufacturing Systems*, 2021. 61: p. 495-508. <https://doi.org/10.1016/j.jmsy.2021.09.017>
  37. Yosinski, J., et al., How transferable are features in deep neural networks? *Advances in neural information processing systems*, 2014. 27.
  38. Zheng G, Sun W, Zhang H, Zhou Y, Gao C. Tool wear condition monitoring in milling process based on data fusion enhanced long shortterm memory network under different cutting conditions. *Eksploracja i Niezawodność – Maintenance and Reliability* 2021; 23 (4): 612-618, <http://doi.org/10.17531/ein.2021.4.3>.
  39. Zhou, Y., B. Sun, and W. Sun, A tool condition monitoring method based on two-layer angle kernel extreme learning machine and binary differential evolution for milling. *Measurement*, 2020. 166: p. 108186. <https://doi.org/10.1016/j.measurement.2020.108186>
  40. Zhou, Y., Kumar, A., Parkash, C., et al. Development of entropy measure for selecting highly sensitive WPT band to identify defective components of an axial piston pump. *Applied Acoustics*; 2023; 203: 109225, <https://doi.org/10.1016/j.apacoust.2023.109225>.
  41. Zhou, Y., Zhi, G., Chen, W., et al., A new tool wear condition monitoring method based on deep learning under small samples. *Measurement*, 2022. 189: p. 110622. <https://doi.org/10.1016/j.measurement.2021.110622>
  42. Zhou YQ, Kumar A, Parkash C, et al. A novel entropy-based sparsity measure for prognosis of bearing defects and development of a sparsogram to select sensitive filtering band of an axial piston pump. *Measurement* 2022; 203: 111997, <https://doi.org/10.1016/j.measurement.2022.111997>.
  43. Zhu, Q., Sun, B., Zhou, Y., et al., Sample augmentation for intelligent milling tool wear condition monitoring using numerical simulation and generative adversarial network. *IEEE Transactions on Instrumentation and Measurement*, 2021. 70: p. 1-10. <https://doi.org/10.1109/TIM.2021.3077995>